



**National Security Agency
Information Assurance
Directorate**



**(U) Global Information Grid
Information Assurance
Capability/Technology Roadmap**

Version 1.0 (Final Draft)

26 October 2004

Prepared by:
IA Architecture Office (I11)

33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50

(U) This page intentionally left blank

(U) TABLE OF CONTENTS

51			
52	Section	Title	Page
53	(U) Revision History		xv
54	(U) Executive Summary		I
55	1 (U) Introduction		1-1
56	1.1 (U) Purpose		1-1
57	1.2 (U) Scope		1-2
58	1.3 (U) Approach.....		1-2
59	2 (U) IA System Enablers and their Technologies		2-1
60	2.1 (U) Identification and Authentication.....		2.1-1
61	2.1.1 (U) GIG Benefits due to I&A		2.1-2
62	2.1.2 (U) I&A: Description.....		2.1-2
63	2.1.3 (U) I & A: Technologies		2.1-4
64	2.1.3.1 (U) Authentication Tokens.....		2.1-5
65	2.1.3.2 (U) Biometrics		2.1-15
66	2.1.3.3 (U) Device/Service Authentication		2.1-25
67	2.1.3.4 (U) Authentication Protocols.....		2.1-35
68	2.1.3.5 (U) Authentication Confidence		2.1-43
69	2.1.3.6 (U) Single Sign-On.....		2.1-46
70	2.1.4 (U) I&A Gap Analysis		2.1-59
71	2.1.5 (U) Identification and Authentication: Recommendations and Timelines		2.1-62
72	2.2 (U) Policy-Based Access Control		2.2-1
73	2.2.1 (U) GIG Benefits due to Policy-Based Access Control.....		2.2-2
74	2.2.2 (U) Policy-Based Access Control: Description		2.2-2
75	2.2.2.1 (U) Core RAdAC Functions.....		2.2-2
76	2.2.2.2 (U) Assured Metadata and Data Describing Enterprise Elements		2.2-4
77	2.2.2.3 (U) Digital Access Control Policy.....		2.2-5

78	2.2.2.4 (U) IA Enabler Dependencies	2.2-6
79	2.2.3 (U) Policy-Based Access Control: Technologies	2.2-6
80	2.2.3.1 (U) Core RAdAC.....	2.2-7
81	2.2.3.2 (U) Assured Metadata	2.2-15
82	2.2.3.3 (U) Digital Access Control Policy.....	2.2-40
83	2.2.4 (U) Distributed Policy Based Access Control: Gap Analysis.....	2.2-44
84	2.2.4.1 (U) Core RAdAC: Gap Analysis.....	2.2-44
85	2.2.4.2 (U) Assured Metadata: Gap Analysis.....	2.2-46
86	2.2.4.3 (U) Digital Access Control Policy: Gap Analysis.....	2.2-49
87	2.2.5 (U) Policy Based Access Control: Recommendations and Timelines.....	2.2-50
88	2.3 (U) Protection of User Information	2.3-1
89	2.3.1 (U) GIG Benefits Due to Protection of User Information	2.3-2
90	2.3.2 (U) Protection of User Information: Description.....	2.3-3
91	2.3.3 (U) Protection of User Information: Technologies.....	2.3-7
92	2.3.3.1 (U) Technologies for Protecting Data-at-Rest.....	2.3-8
93	2.3.3.2 (U) Technologies for Protecting Data-in-Transit	2.3-10
94	2.3.3.3 (U) Trusted Platforms.....	2.3-87
95	2.3.3.4 (U) Trusted Applications.....	2.3-98
96	2.3.3.5 (U) Cross Domain Solution Technologies	2.3-106
97	2.3.3.6 (U) Non-Repudiation.....	2.3-116
98	2.3.4 (U) Protection of User Information: Gap Analysis.....	2.3-126
99	2.3.5 (U) Protection of User Information: Recommendations and Technology	
100	Timelines.....	2.3-130
101	2.3.5.1 (U) Data-in-Transit.....	2.3-130
102	2.3.5.2 (U) Cross Domain Solutions	2.3-132
103	2.4 (U) Dynamic Policy Management.....	2.4-1
104	2.4.1 (U) GIG Benefits due to Dynamic Policy Management.....	2.4-2

105 2.4.2 (U) Dynamic Policy Management: Description 2.4-2

106 2.4.3 (U) Dynamic Policy Management: Technologies..... 2.4-6

107 2.4.3.1 (U//FOUO) Development of Policies..... 2.4-6

108 2.4.3.2 (U) Distribution of Policies 2.4-22

109 2.4.3.3 (U) Policy Management Architectures..... 2.4-29

110 2.4.4 (U) Dynamic Policy Management: Gap Analysis 2.4-31

111 2.4.5 (U) Dynamic Policy Management: Recommendations and Timelines..... 2.4-33

112 2.4.5.1 (U) Standards..... 2.4-33

113 2.4.5.2 (U) Technology 2.4-34

114 2.4.5.3 (U) Infrastructure..... 2.4-34

115 **2.5 (U) Assured Resource Allocation..... 2.5-1**

116 2.5.1 (U) GIG Benefits of Assured Resource Allocation 2.5-3

117 2.5.2 (U) Assured Resource Allocation: Description 2.5-3

118 2.5.3 (U) Technologies 2.5-5

119 2.5.3.1 (U//FOUO) IA Policy-Based Routing..... 2.5-6

120 2.5.3.2 (U//FOUO) Operational-Based Resource Allocation..... 2.5-17

121 2.5.3.3 (U//FOUO) Integrity of Network Fault Monitoring/Recovery and

122 Integrity of Network Management & Control..... 2.5-26

123 2.5.4 (U) Assured Resource Allocation: Gap Analysis 2.5-38

124 2.5.5 (U) Assured Resource Allocation: Recommendations and Technology

125 Timelines..... 2.5-40

126 **2.6 (U) Network Defense and Situational Awareness 2.6-1**

127 2.6.1 (U) GIG Benefits due to Network Defense and Situational Awareness..... 2.6-2

128 2.6.2 (U) Network Defense and Situational Awareness: Description 2.6-3

129 2.6.3 (U) Network Defense and Situational Awareness: Technologies..... 2.6-8

130 2.6.3.1 (U) Protect Technologies..... 2.6-9

131 2.6.3.2 (U) Deception Technologies 2.6-14

132	2.6.3.3	(U) Situational Awareness.....	2.6-21
133	2.6.3.4	(U) Network Mapping	2.6-26
134	2.6.3.5	(U) Intrusion Detection Systems	2.6-30
135	2.6.3.6	(U) Intrusion Prevention Systems (IPSs)	2.6-37
136	2.6.3.7	(U) User Activity Profiling.....	2.6-39
137	2.6.3.8	(U) Cyber Attack Attribution	2.6-44
138	2.6.3.9	(U) Correlation Technologies.....	2.6-49
139	2.6.3.10	(U) CND Response Actions	2.6-54
140	2.6.3.11	(U) Automated IAVA Patch Management	2.6-58
141	2.6.4	(U) Network Defense and Situational Awareness: Gap Analysis	2.6-63
142	2.6.5	(U) Network Defense and Situational Awareness: Recommendations and	
143		Timelines.....	2.6-66
144	2.6.5.1	(U) Standards.....	2.6-66
145	2.6.5.2	(U) Technology	2.6-66
146	2.6.5.3	(U) Infrastructure.....	2.6-69
147	2.6.5.4	(U) Technology Timelines	2.6-69
148	2.7	(U) Management of IA Mechanisms and Assets	2.7-1
149	2.7.1	(U) GIG Benefits due to Management of IA Mechanisms and Assets.....	2.7-1
150	2.7.2	(U) Management of IA Mechanisms and Assets: Description	2.7-1
151	2.7.2.1	(U) Identity Management	2.7-2
152	2.7.2.2	(U) Privilege Management	2.7-5
153	2.7.2.3	(U) Key Management.....	2.7-9
154	2.7.2.4	(U) Certificate Management.....	2.7-11
155	2.7.2.5	(U) Configuration Management of IA Devices and Software	2.7-14
156	2.7.2.6	(U) Inventory Management	2.7-16
157	2.7.2.7	(U) Compromise Management of IA Devices	2.7-16
158	2.7.2.8	(U) Audit Management	2.7-17

159	2.7.3 (U) Management of IA Mechanisms & Assets: Technologies	2.7-18
160	2.7.3.1 (U) Identity Management	2.7-18
161	2.7.3.2 (U) Privilege Management	2.7-26
162	2.7.3.3 (U) Key Management	2.7-33
163	2.7.3.4 (U) Certificate Management.....	2.7-49
164	2.7.3.5 (U) Configuration Management of IA Devices and Software	2.7-59
165	2.7.3.6 (U) Inventory Management	2.7-68
166	2.7.3.7 (U) Compromise Management of IA Devices	2.7-76
167	2.7.3.8 (U) Audit Management	2.7-82
168	2.7.4 (U) Management of IA Mechanisms & Assets: Gap Analysis.....	2.7-93
169	2.7.4.1 (U) Identity Management	2.7-96
170	2.7.4.2 (U) Privilege Management	2.7-96
171	2.7.4.3 (U) Key Management	2.7-97
172	2.7.4.4 (U) Certificate Management.....	2.7-97
173	2.7.4.5 (U) Configuration Management of IA Devices and Software	2.7-98
174	2.7.4.6 (U) Inventory Management	2.7-99
175	2.7.4.7 (U) Compromise Management of IA Devices	2.7-99
176	2.7.4.8 (U) Audit Management	2.7-99
177	2.7.5 (U) Management of IA Mechanisms and Assets: Recommendations and	
178	Timelines.....	2.7-100
179	2.7.5.1 (U) Standards.....	2.7-100
180	2.7.5.2 (U) Technology	2.7-101
181	2.7.5.3 (U) Infrastructure.....	2.7-101
182	3 (U) Summary	3-1
183	3.1 (U//FOUO) Assured information Sharing Summary	3.1-3
184	3.1.1 (U) Identification and Authentication Technologies	3.1-3
185	3.1.2 (U) Access Control and Data Labeling Technologies	3.1-5

186 3.1.3 (U) Cross-Domain Technologies 3.1-7

187 3.1.4 (U) Trusted Platform Technologies 3.1-9

188 **3.2 (U) Highly Available Enterprise Summary 3.2-11**

189 3.2.1 (U//FOUO) IA Policy-based Routing for Mobile/Tactical Environments
190 Technologies 3.2-11

191 3.2.2 (U) End-to-End Resource Allocation Technologies 3.2-12

192 3.2.3 (U//FOUO) Edge-to-Edge Boundary Protection Technologies 3.2-13

193 3.2.4 (U) Secure Voice Technologies 3.2-13

194 3.2.5 (U) Enforcement of QoP in Transit Technologies 3.2-14

195 3.2.6 (U//FOUO) Protection of High Risk Link Technologies..... 3.2-14

196 **3.3 (U) Assured Enterprise Management and Control Summary..... 3.3-15**

197 3.3.1 (U) Identity Management Technologies 3.3-16

198 3.3.2 (U) Inventory Management Technologies 3.3-16

199 3.3.3 (U) Privilege Management Technologies 3.3-17

200 3.3.4 (U) Key Management Technologies 3.3-18

201 3.3.5 (U) Certificate Management Technologies..... 3.3-19

202 3.3.6 (U) Configuration Management Technologies 3.3-20

203 3.3.7 (U) Policy Management Technologies 3.3-20

204 3.3.8 (U) Audit Management Technologies 3.3-22

205 3.3.9 (U) Confidentiality & Integrity of Network Management & Control
206 Technologies 3.3-23

207 **3.4 (U) Cyber Situational Awareness and Network Defense Summary..... 3.4-24**

208 3.4.1 (U) Protection Technologies 3.4-25

209 3.4.2 (U) Monitoring Technologies 3.4-25

210 3.4.3 (U) Detection Technologies 3.4-26

211 3.4.4 (U) Analysis Technologies 3.4-28

212 3.4.5 (U) Response Technologies 3.4-29

213 **4 (U) Acronyms and Abbreviations..... 4-1**

214

215

Appendices

216 **(U//FOUO) Appendix A: Mapping of technologies to IA System Enablers A-2**

217 **(U//FOUO) Appendix B: TV-1 for IA A-6**

218 **(U//FOUO) Appendix C: TV-2 for IA..... A-23**

219

220	(U) LIST OF FIGURES		
221	Figure	Title	Page
222	Figure 1.3-1: (U)	GIG Mission Concepts, IA Cornerstones, and IA System Enablers	1-3
223	Figure 1.3-2: (U)	Iterative Development of the GIG IA Capability/Technology Roadmap	1-5
224	Figure 2.1-1: (U)	Examples of time-driven hardware tokens	2.1-6
225	Figure 2.1-2: (U)	DoD Common Access Card	2.1-10
226	Figure 2.1-3: (U)	Example of a Hybrid Device	2.1-14
227	Figure 2.1-4: (U)	Biometric System Block Diagram.....	2.1-15
228	Figure 2.1-5: (U)	Network Authentication Framework.....	2.1-37
229	Figure 2.1-6: (U)	Device Authentication Framework.....	2.1-37
230	Figure 2.1-7: (U)	Centralized Architecture for Single Sign-On	2.1-48
231	Figure 2.1-8: (U)	Federated KEBEROS Based Single Sign-On.....	2.1-50
232	Figure 2.1-9: (U)	Federated PKI-based Single Sign-on.....	2.1-51
233	Figure 2.1-10: (U)	Federated SAML-Based Single Sign-On	2.1-52
234	Figure 2.2-1: (U)	RAAdAC Functional Model	2.2-3
235	Figure 2.2-2: (U)	Codifying the Net-Centric Data Strategy	2.2-22
236	Figure 2.2-3: (U)	Encapsulation Notional Diagram	2.2-38
237	Figure 2.2-4: (U)	Policy-Based Access Control Gap Closure Timelines	2.2-51
238	Figure 2.3-1: (U)	Context of Non Real-Time Application Security.....	2.3-11
239	Figure 2.3-2:(U)	Layered Protocol Wrapping Concept.....	2.3-12
240	Figure 2.3-3:(U)	CMS Supports S/MIME and Other Secure Applications.....	2.3-16
241	Figure 2.3-4: (U)	TLS Handshake Protocol.....	2.3-23
242	Figure 2.3-5:(U)	Model for Web Services Security	2.3-30
243	Figure 2.3-6: (U)	FNBDT Location in Network Protocol Stack	2.3-33
244	Figure 2.3-7: (U)	Packet Jitter Mitigation Process	2.3-35
245	Figure 2.3-8: (U//FOUO)	FNBDT Frame Structure for Signaling Reliability and Reliable	

246	Transport Data Mode	2.3-36
247	Figure 2.3-9: (U//FOUO) FNBDT 2400 bps MELP Blank and Burst Superframe	
248	Structure	2.3-37
249	Figure 2.3-10: (U//FOUO) FNBDT 7200 bps G.729D Superframe Structure	2.3-37
250	Figure 2.3-11: (U) Media Gateway Protocol Stack Illustration.....	2.3-41
251	Figure 2.3-12: (U//FOUO) Secure Voice Gateway Functionality	2.3-42
252	Figure 2.3-13: (U) Real-Time Protocol	2.3-46
253	Figure 2.3-14: (U) RTCP Sender Report Format- Sender Report (SR).....	2.3-48
254	Figure 2.3-15: (U) SRTP Format	2.3-50
255	Figure 2.3-16: (U) SRTCP Format	2.3-50
256	Figure 2.3-17: (U//FOUO) FNBDT over V.150.1 Modem Relay	2.3-52
257	Figure 2.3-18: (U) V.150.1 Simple Packet Relay Transport for IP networks	2.3-52
258	Figure 2.3-19: (U//FOUO) State Variable Stepping.....	2.3-62
259	Figure 2.3-20: (U) SIP Architecture	2.3-78
260	Figure 2.3-21: (U) H.323 Network Elements	2.3-80
261	Figure 2.3-22 (U) Legacy Manifestation of Cross-Domain Solutions	2.3-106
262	Figure 2.3-23: (U) Controlled Interface Example.....	2.3-107
263	Figure 2.3-24: (U) Two MSL Architectures.....	2.3-108
264	Figure 2.3-25: (U) Technology Timeline for Protection of User Information: Date in	
265	Transit	2.3-132
266	Figure 2.3-26: (U) Technology Timeline for Protection of User Information: Cross	
267	Domain Solutions.....	2.3-133
268	Figure 2.4-1: (U) Notional Architectural Framework for Dynamic Policy Management	2.4-3
269	Figure 2.4-2: (U) Berners-Lee’s Seven Layer Model of the Semantic Web	2.4-19
270	Figure 2.4-3: (U) Technology Timeline for Dynamic Policy Management	2.4-35
271	Figure 2.5-1: (U//FOUO) The Role and Components of Assured Resource Allocation	2.5-2
272	Figure 2.5-2: (U//FOUO) IA Policy-Based Routing.....	2.5-6

273	Figure 2.5-3: (U//FOUO) Security-Aware ad-hoc Routing (SAR) in Tactical Wireless	
274	Application.....	2.5-10
275	Figure 2.5-4: (U) OSPF Implemented With (QoS) IA Policy-Based Routing Extensions.....	2.5-14
276	Figure 2.5-5: (U) DeSiDeRaTa Architecture for Operational-Based Resource Allocation	2.5-18
277	Figure 2.5-6: (U) Joint Resource Allocation Across GIG Networks.....	2.5-21
278	Figure 2.5-7: (U) Basic Elements of SNMP Operation	2.5-26
279	Figure 2.5-8: (U) SNMPv3 Security Capabilities.....	2.5-27
280	Figure 2.5-9: (U) SNMPv3 Message Format & Security Components	2.5-28
281	Figure 2.5-10: (U) SNMPv3 View-based Access Control Model (VACM) Logic	2.5-29
282	Figure 2.5-11: (U) Technology Timeline for Assured Resource Allocation	2.5-41
283	Figure 2.6-1: (U) Representative Sensor Configuration.....	2.6-4
284	Figure 2.6-2: (U) Representative Flow of Situational Awareness Data	2.6-5
285	Figure 2.6-3: (U) Vulnerabilities Reported from CERT.....	2.6-59
286	Figure 2.6-4: (U) Technology Timeline for Network Defense and Situational Awareness	2.6-69
287	Figure 2.7-1: (U//FOUO) Assured Sharing Context Diagram Emphasizing Privileges.....	2.7-6
288	Figure 2.7-2: (U) Example of Multiple Identities Assigned to a Single User	2.7-18
289	Figure 2.7-3: (U//FOUO) ECU and Technology Evolution	2.7-33
290	Figure 2.7-4: (U) Current Key Management Infrastructures	2.7-33
291	Figure 2.7-5: (U//FOUO) KMI – Envisioned Infrastructure	2.7-35
292	Figure 2.7-6: (U//FOUO) KMI Protected Channel Layers.....	2.7-39
293	Figure 2.7-7: (U) XKMS Environment.....	2.7-40
294	Figure 2.7-8: (U) DoD and Commercial Certificate-Managed Infrastructures	2.7-49
295	Figure 2.7-9: (U) PKI Technology Model	2.7-50
296	Figure 2.7-10: (U) RFID Operation	2.7-69
297	Figure 2.7-11: (U) Audit Life Cycle.....	2.7-82
298	Figure 2.7-12: (U) Audit Trail Information Flow	2.7-84

299 Figure 2.7-13: (U) Audit Logs – Protection..... 2.7-86

300 Figure 2.7-14: (U) Aggregation and Normalization 2.7-88

301 Figure 2.7-15: (U) Interfaces - Agents and Pipes between Log Devices and the
302 Collection/Monitoring Processes 2.7-89

303 Figure 2.7-16: (U) Technology Timeline for Assured Resource Allocation 2.7-102

304 Figure 3.1-1: (U//FOUO) Technology Timeline for Assured Information Sharing 3.1-3

305 Figure 3.2-1: (U//FOUO) Technology Timeline for Highly Available Enterprise..... 3.2-11

306 Figure 3.3-1: (U//FOUO) Technology Timeline for Assure Enterprise Management and
307 Control 3.3-15

308 Figure 3.4-1: (U//FOUO) Technology Timeline for Cyber Situational Awareness and
309 Network Defense 3.4-24

(U) LIST OF TABLES

310	(U) LIST OF TABLES		
311	Table	Title	Page
312	Table 1.3-2: (U) Example of a Technology Adequacy Matrix.....		2-4
313	Table 2.1-1: (U) Hardware Token Standards.....		2.1-9
314	Table 2.1-2: (U) Biometric Standards.....		2.1-21
315	Table 2.1-3: (U) Technology Adequacy for Tokens and Biometrics		2.1-60
316	Table 2.1-4: (U) Technology Adequacy for Single Sign-On and Authentication		2.1-61
317	Table 2.2-1: (U) Access Control Standards		2.2-11
318	Table 2.2-2: (U) Technologies Supporting Access Control.....		2.2-12
319	Table 2.2-3: (U) Minimum Set of IA Attributes for Access Control Decisions.....		2.2-18
320	Table 2.2-4: (U) IC and CES Metadata Working Groups Attribute Comparison		2.2-20
321	Table 2.2-5: (U) Metadata Standards.....		2.2-24
322	Table 2.2-6: (U) Metadata Gap Analysis.....		2.2-27
323	Table 2.2-7: (U) Metadata Tool Standards		2.2-35
324	Table 2.2-8: (U) Standards on Cryptographic Binding.....		2.2-39
325	Table 2.2-9: (U) Technology Adequacy for Access Control.....		2.2-46
326	Table 2.2-10: (U) Technology Adequacy for Metadata.....		2.2-48
327	Table 2.3-1: (U) Traditional Layered Application Security Standards		2.3-19
328	Table 2.3-2: (U) Session Security Standards.....		2.3-26
329	Table 2.3-3: (U) Web Services Security Standards.....		2.3-31
330	Table 2.3-4: (U) FNBDT Standards.....		2.3-38
331	Table 2.3-5: (U) Secure Voice over IP Standards.....		2.3-57
332	Table 2.3-6: (U//FOUO) HAIPE ESP Tunnel Mode Encapsulation		2.3-60
333	Table 2.3-7: (U//FOUO) HAIPE State Variable Content		2.3-61
334	Table 2.3-8: (U//FOUO) IP Security Technology Readiness Levels		2.3-64
335	Table 2.3-9: (U//FOUO) Standards Applicable to IP Security Technology		2.3-66
336	Table 2.3-10: (U//FOUO) Standards Applicable to VPN Technology.....		2.3-73

337 Table 2.3-11: (U) Secure VoIP Call Control Standards 2.3-84

338 Table 2.3-12: (U) CDS Standards..... 2.3-114

339 Table 2.3-13: (U) Non-Repudiation Standards..... 2.3-124

340 Table 2.3-14: (U//FOUO) Secure Voice Technology Gap Analysis 2.3-126

341 Table 2.3-15: (U//FOUO) Gap Analysis for Non-real-time Application Layer
 342 Technologies 2.3-128

343 Table 2.3-16: (U//FOUO) CDS Technology Gap Assessment..... 2.3-129

344 Table 2.4-1: (U) Access Control Standards 2.4-10

345 Table 2.4-2: (U) Trust Anchor Standards 2.4-13

346 Table 2.4-3: (U) Policy Language Standards..... 2.4-20

347 Table 2.4-4: (U) Distribution Standards 2.4-24

348 Table 2.4-5: (U) Distribution Security Standards 2.4-28

349 Table 2.4-6: (U) Directory Standards 2.4-31

350 Table 2.4-7: (U) Technology Adequacy for Dynamic Policy Management..... 2.4-33

351 Table 2.5-1: (U) Technology Adequacy for Assured Resource Allocation..... 2.5-39

352 Table 2.6-1: (U) Standards for Intrusion Detection Systems..... 2.6-33

353 Table 2.6-2:(U) Network Defense & Situational Awareness Technology Gap Assessment.... 2.6-64

354 Table 2.6-3: (U//FOUO) Summary of Technology Gaps 2.6-67

355 Table 2.7-1 (U) Identity Management Standards 2.7-24

356 Table 2.7-2: (U) Comparisons of PKI and PMI..... 2.7-28

357 Table 2.7-3: (U) Privilege Management Standards 2.7-31

358 Table 2.7-4: (U) Key Management Standards 2.7-47

359 Table 2.7-5 (U) Key Management and Certificate Management Standards..... 2.7-54

360 Table 2.7-6: (U) Configuration Management Standards 2.7-64

361 Table 2.7-7: (U) Frequency Ranges for RFID Systems..... 2.7-70

362 Table 2.7-8: (U) Inventory Management RFID Standards 2.7-72

363 Table 2.7-9: (U) Compromise Management Standards 2.7-79

364 Table 2.7-10: (U) Audit Management Standards..... 2.7-91

365 Table 2.7-11: (U) Technology Adequacy for Management of IA Mechanisms and Assets..... 2.7-94

366 Table A-1: (U//FOUO) Mapping of Technologies to IA System Enablers A-2

367 Table B-1: (U//FOUO) TV-1 for IA A-6

368 Table C-1: (U//FOUO) TV-2 for IA A-23

(U) REVISION HISTORY

This Table is (U//FOUO)		
Revision Number	Description	Date
Revision 1.0 Initial Draft	Initial baseline document that describes the Capability Technology Roadmap. Primary focus is on Identification and Authentication technologies	30 June 2004
Revision 1.0 Final Draft	<p>General</p> <p>Revised Summary and Executive Summary based on latest technology research and ability to meet Transition Strategy for each Cornerstone</p> <p>Reorganized introduction and eliminated Global Information Grid (GIG) Mission Concept description</p> <p>Added introduction to Section 2 that explains application of TRLs, adequacy levels, and technology timelines in subsequent sections</p> <p>Deleted Appendix on IA Pillars, added appendix with mapping of technologies to section where described, and updated TV-1 and TV-2</p> <p>2.1 Identification and Authentication</p> <p>Refined Enabler description, pulling out Identity Management since it is covered in Enabler 7, Management of IA Mechanisms and Assets</p> <p>Reorganized technologies and add material on Authentication Protocols. Clarified other text</p> <p>Revised gap analysis to reflect adequacy of current technologies to meet 2008 needs</p> <p>Revised recommendations to reflect results of gap analysis</p> <p>2.2 Policy-Based Access Control</p> <p>Editorial Clean-up</p> <p>Expanded Functionality Description</p> <p>Technologies content added and subsection structure changed to reflect results of Technology Analysis Results (Major Technology Subsections: Core RAdAC, Assured Metadata, Digital Access Control Policy)</p> <p>Technologies content added and subsection structure changed to reflect results of Technology Gap Analysis Results (Major Technology Subsections: Core RAdAC, Assured Metadata, Digital Access Control Policy)</p> <p>Section revised to reflect roll-up of Gap Closure recommendation from the major Access Control technology subsections. Also eliminated "Standards," "Technology," and "Infrastructure" subsections as this information subsumed into each major technology subsection</p> <p>2.3 Protection of User Information</p> <p>Added material on Trusted Platforms. Combined (previously empty) sections on Trusted Platforms and Trusted Operating Systems</p> <p>Added material on Trusted Applications. Section was previously empty</p> <p>Added material on Web Security and Application Layer Security</p> <p>Added material on FNBDT and VoIP technologies</p> <p>2.4 Dynamic Policy Management</p>	15 Oct 2004

Updated description based upon revised Notional Architecture which is better aligned with RFCs
Technologies content added and subsection structure changed to reflect results of Technology Analysis Results (Major Technology Subsections: Development of Policies, Distribution of Policies, Policy Architectures)
Technology gap information added to reflect results of Technology Gap Analysis Results (Major Technology Gaps: Expanded policy languages, policy modeling and simulation tools, policy deconfliction tools, and tools or compilers to translate policy language into a device interpretable language)
Section revised to reflect roll-up of Gap Closure recommendation from the major Policy Management technology subsections. Also updated timeline for technologies
2.5 Assured Resource Allocations
Technologies content added (Major Technology Subsections: IA Policy-Based Routing, Operational-Based Resource Allocation, Integrity of Network Fault Monitoring/Recovery and Integrity of Network Management & Control)
Technologies content added along with Technology Adequacy matrix
Section revised to include Recommendations list and edited Technology Timeline figure
2.6 Network Defense and Situational Awareness
Protect Technologies content added
Situational Awareness: Maturity content added
Technologies content added and subsection structure changed to reflect results of Technology Gap Analysis Results (Major Technology Subsections: Core RAdAC, Assured Metadata, Digital Access Control Policy)
Intrusion Detection Systems content added
Intrusion Prevention Systems content added
Cyber Attack Attribution – Editorial clean-up based on comments from John Lowry
Correlation Technologies: Technical Detail content added
CND Response Actions content added
2.7 Management of IA Mechanisms and Assets
Key Management - Elaborated on the Maturity Section and the Technology Readiness level (TRL)
Audit Management: Modified based on comments and feedback. These had to do with Maturity subsection through Complementary Technologies
Added material on Configuration Management of IA Assets, Compromise Management and Inventory Management
Modifications and additions were made in order to better represent the Technology Gap Analysis Results
Standards, Technology, Infrastructure, and Timelines were all enhanced with additional inputs. The tables were reconfigured, values assigned, and summarized
This Table is (U//FOUO)

371 **(U) EXECUTIVE SUMMARY**

372 (U//FOUO) The Office Secretary of Defense/ Networks and Information Integration
373 (OSD/NII) tasked the National Security Agency (NSA) to develop the Information
374 Assurance (IA) component of the Global Information Grid (GIG). This GIG IA
375 Capability/Technology Roadmap document, together with several other documents—
376 including the GIG IA Reference Capability Document (RCD)—describe the IA
377 component of the GIG.

378 (U) The GIG IA Capability/Technology Roadmap identifies the technologies needed to
379 implement the GIG IA Vision, and it provides a partial evaluation of current and in-
380 development technologies that can or will be able to support GIG needs. As such, the
381 objectives of this document are to:

- 382 • (U) Establish, within the context of the GIG IA engineering process, an effective
383 methodology to discover and examine relevant technologies for the purpose of
384 providing guidance to GIG program decision makers and GIG research sponsors
- 385 • (U) Provide an assessment of the maturity and suitability of relevant IA
386 technologies to meet GIG IA-required capabilities, focusing in particular on the
387 2008 milestones of the transition strategy outlined in the GIG IA RCD
- 388 • (U) Identify gaps in standards and technologies that will prevent attainment of
389 GIG IA capabilities, and recommend specific actions to take in closing those gaps
- 390 • (U) Serve as a means for members of the GIG community and stake holders to
391 gain visibility into the technology roadmap process and provide feedback on
392 appropriate topics, such as standards or significant technologies overlooked
393 during the study to date

394 (U) In meeting these objectives, this document provides decision makers with the
395 information needed to write new or revise existing standards and policies, develop
396 implementation guidance, make research funding decisions, and devise technology
397 development strategies.

398 **(U) Scope**

399 (U) The GIG IA Capability/Technology Roadmap document presents a fairly complete
400 view of all the technologies that can or should be used to implement IA in the GIG.
401 Those that can support the GIG IA vision are examined in detail. Results are presented to
402 describe the ability of the most promising technologies to fulfill needed GIG IA
403 capabilities in terms of technical capability, maturity, development schedule, and
404 availability. Interdependencies between needed capabilities, technology timelines, and
405 gaps between capability needs and technology availability are also described.

406 (U//FOUO) In developing the roadmap, the team compared the state, trends, and
407 forecasts of commercial and government technologies available today against the needed
408 capabilities defined in the RCD. Three main categories of information were used. The
409 first is documentation and analyses performed by the NSA as part of development of the
410 IA component of the GIG architecture. This information includes the GIG Mission
411 Concepts, the *As Is* state of GIG programs, and the GIG risk analysis. The second
412 category of information includes current IA standards, technology trends and forecasts
413 available from commercial sources such as Gartner, IDC, etc. and Government trends and
414 forecasts. The third type of information—to be used in subsequent versions of the GIG
415 IA Capability/Technology Roadmap document—is previously-determined technology
416 gaps.

417 (U) Results

418 (U//FOUO) The analyses were carried out in the context of the capabilities outlined in the
419 RCD and the Transition Strategy. In particular, the team assessed the maturity and
420 adequacy of the technologies in meeting the 2008 Vision milestones (Increment I).

421 (U) The results show that nearly all the Increment I milestones can be achieved if actions
422 are taken immediately to address identified technology or capability gaps. The roadmap
423 provides over 75 specific recommendations to address these gaps. Recommendations
424 range from monitoring ongoing technologies and standards development efforts to ensure
425 compliance with GIG IA needs, to initiating new technology research to support post-
426 2008 milestones) and standards development efforts. We believe that most milestones can
427 be achieved if immediate action is taken on these recommendations.

428 (U) In our estimation, five milestones defined in the Transition Strategy are unachievable
429 by the specified dates:

- 430 • (U//FOUO) *Limited support for end-to-end resource allocation* milestone.
431 Operationally-based resource allocation technologies are very immature,
432 especially considering the constraints and limitations of a secure Black Core.
433 Since there is much research remaining to be done in this area, it is our opinion
434 that a limited capability for allocating resources end-to-end will not be available
435 until 2012—four years after the objective date. The operational impact is a delay
436 in moving from today's *best effort* service for all to efficient resource allocation
437 schemes that ensure priority users receive needed services based on mission
438 criticality to efficient resource allocation schemes.

- 439 • (U//FOUO) *All human users identified in accordance with GIG ID standard*
440 milestone. Currently, standards neither exist nor are under development for
441 establishing and maintaining unique, persistent, and non-forgable identities as
442 will be needed for the GIG. Because of the coordination that will be required
443 across multiple communities, such standards will not likely be in place to support
444 subsequent technology development in time to meet a 2008 objective; however,
445 2010 is an achievable date for this milestone. No impact on 2008 operational
446 objectives are expected, but delays in meeting 2012 operational objectives is
447 likely. These include: 1) achieving closer collaboration within Communities of

- 448 Interest (COI),
449 2) implementing a global sign-on capability, and 3) achieving Risk-Adaptive
450 Access Control (RAdAC).
- 451 • (U//FOUO) *Over-the-network keying for wired and wireless devices* milestone.
452 Efforts are planned for developing the needed security technologies. However an
453 initial capability will not be fielded until 2010, two years after the deadline. Low
454 bandwidth devices, such as wireless nodes, will not be supported until 2012, and
455 coalition networks will not be addressed until 2016. The operational impact is a
456 continued dependence on manual re-keying, which 1) requires greater manpower
457 and costs for handling and safeguarding key material, which will become more
458 troublesome as additional IP encryptors are deployed as the GIG matures, and 2)
459 causes slower response to key compromises, risking more widespread damage.
 - 460 • (U//FOUO) *Configuration management standards ratified* milestone. Remote
461 configuration products abound, but standards do not yet exist for the secure
462 management of IA-enabled devices. Due to the time needed to draft, coordinate,
463 and achieve consensus among the engineering community, such standards will
464 likely not be ratified before 2008, two years later than the milestone called out in
465 the Transition Strategy. The operational impact is a delay in achieving a
466 consolidated network view. This reduces the overall security posture of the GIG
467 and prevents policy-based network management.
 - 468 • (U//FOUO) *Audit format and exchange standard ratified* milestone. Auditing
469 products are available today, but the absence of standards, holds-up product
470 integration into the GIG. Developing the needed audit standards and achieving
471 industry acceptance is not likely to be achieved until 2008, two years after the
472 milestone. This will delay the ability to carry-out forensic analysis of attacks and
473 thus hamper computer network defense.
- 474 (U) Section 3 provides a summary of the identified gaps and recommendations.
- 475 (U) While this version of the document provides the first comprehensive coverage of the
476 technologies, technology assessment is an iterative process: As additional capability
477 needs are identified and IA technologies mature, subsequent analyses will provide
478 recommendations to re-direct current development efforts and initiate new research as
479 needed to meet the GIG visions. These analyses will be documented in subsequent
480 versions of this document, which will be issued on an annual basis.

481 **1 (U) INTRODUCTION**

482 **1.1 (U) PURPOSE**

483 (U//FOUO) The GIG IA Capability/Technology Roadmap document is part of the November
484 2004 deliverables of the Information Assurance (IA) Component of Global Information Grid
485 (GIG) Architecture. Office Secretary of Defense/Networks and Information Integration
486 (OSD/NII) tasked development of the IA component of the GIG architecture to the National
487 Security Agency (NSA).

488 (U//FOUO) Since the tasking by OSD/NII, the NSA has translated the GIG Vision into derived
489 GIG capabilities and associated IA capabilities. The GIG IA Reference Capability Document
490 (RCD) details the IA derived capabilities by describing the general attributes of each capability.
491 Thresholds and objectives are then defined for each attribute. The thresholds are considered near-
492 term GIG IA requirements to meet the 2008 Vision while the objectives are the capabilities
493 required to meet the GIG 2020 Vision.

494 (U//FOUO) The GIG IA Capability/Technology Roadmap identifies the current technology
495 trends in IA and compares the trends against the thresholds and objectives identified in the RCD.
496 The result is an availability timeline of anticipated technologies required to support the GIG
497 2020 Vision.

498 (U//FOUO) The GIG IA Capability/Technology Roadmap document analyzes the technology
499 trends and technology forecasts (both commercial and government) available today against the
500 capabilities defined in the RCD. The results of the analysis are:

- 501 • (U) Capability inter-dependencies
- 502 • (U) Technology timelines
- 503 • (U) Gaps between capability needs and technology availability

504 (U//FOUO) The GIG IA Capability/Technology Roadmap document also provides background
505 information and analysis to support decision makers with regard to:

- 506 • (U) New/Updated standards
- 507 • (U) Infrastructure guidance
- 508 • (U) Technology research to fund
- 509 • (U) Technology strategy development

510 **1.2 (U) SCOPE**

511 (U//FOUO) Section 2, Information Assurance (IA) System Enabler and Their Technologies, is
512 divided into seven subsections based on the Fundamental System Enablers. Each subsection
513 describes the IA System Enabler, covers the GIG implications of the System Enabler, and
514 describes its related technologies. The related technologies define research areas for technology
515 trends and forecasts to support the development of the technology timelines and the
516 capability/technology gap analysis.

517 (U) Section 3, Summary, contains a discussion of the technology improvement
518 recommendations needed to meet the Transition Strategy, defined in the RCD, for each
519 Cornerstone. When technologies are missing or unable to meet the strategy, the discussion
520 highlights the impacted operational capability. The four Cornerstones, defined in the GIG IA
521 Operational Concepts Overview document, are:

- 522 • (U) Assured Information Sharing
- 523 • (U) Highly Available Enterprise
- 524 • (U) Assured Enterprise Management and Control
- 525 • (U) Cyber Situational Awareness and Network Defense

526 (U) Section 4 lists acronyms and abbreviations.

527 (U//FOUO) Appendix A provides a mapping of technologies to IA System Enablers.

528 (U//FOUO) Appendix B: Technical View (TV)-1 for IA, contains standards that exist today that
529 had not previously been identified as needed to satisfy capabilities listed in the RCD.

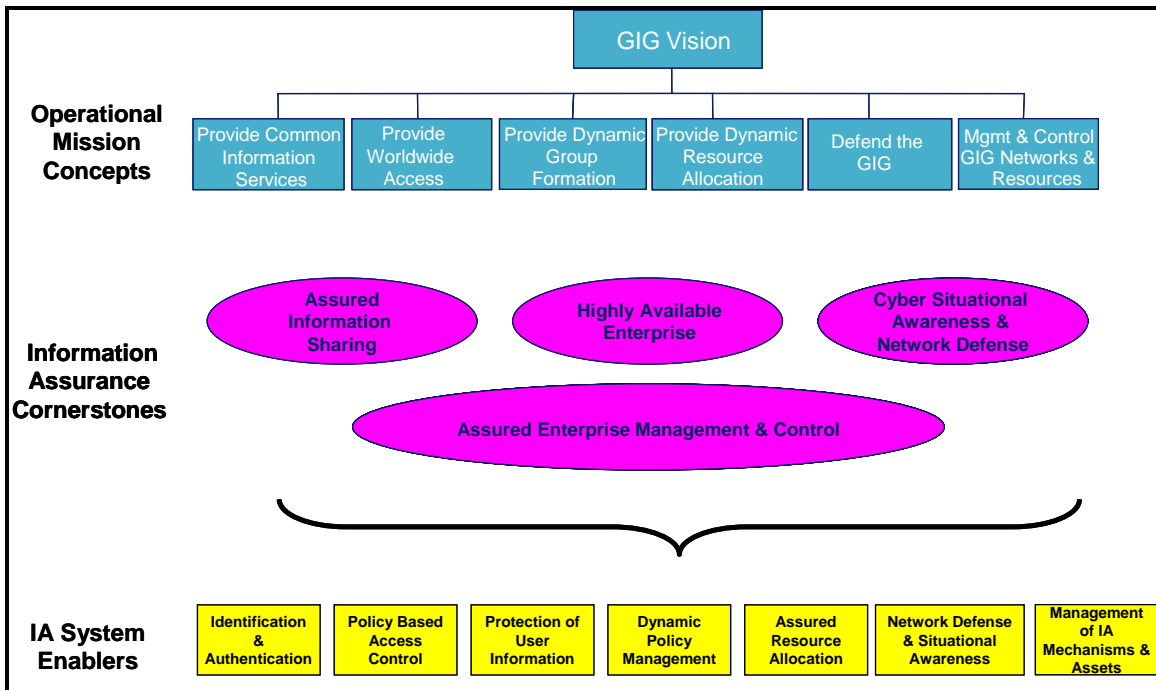
530 (U//FOUO) Appendix C: TV-2 for IA, contains standards that have been identified as needed to
531 satisfy capabilities listed in the RCD but that do not exist today.

532 **1.3 (U) APPROACH**

533 (U//FOUO) The primary guiding principle is to achieve the Objective Goals described in the
534 RCD. This means identifying the necessary technology evolution to fill the gaps between today's
535 IA technology and what is needed for the GIG 2020 Vision's objective system. The IA Risk
536 Assessment helps prioritize—based on security risks—which gaps need to be filled sooner than
537 others. The gap analysis must consider the GIG capability timeline to identify the criticality of
538 each gap.

539 (U//FOUO) The GIG IA Capability/Technology Roadmap document is built upon three main
 540 categories of information. The first category is documentation and analysis performed by the
 541 NSA while developing the IA component of the GIG architecture. This information includes the
 542 GIG Mission Concepts, *As Is* state of GIG programs, and GIG threats as identified by the GIG
 543 Risk Assessment activities. The GIG Mission Concepts capture the NSA’s understanding of the
 544 capabilities required by the GIG, based on the *To Be* GIG vision as defined by the GIG 2020
 545 architecture and documentation. The *As Is* input captures the IA capabilities currently planned by
 546 ongoing GIG programs such as Net-Centric Enterprise Services (NCES), GIG Bandwidth
 547 Expansion (GIG-BE), Transformational Satellite (TSAT) Communications, and the Joint
 548 Tactical Radio System (JTRS). The GIG Risk Assessment identifies threats to the GIG that must
 549 be countered. These threats could be countered in a number of ways, including technology,
 550 standards, and policies.

551 (U//FOUO) The primary document used in development of the GIG IA Capability/Technology
 552 Roadmap is the RCD. This includes a description of the GIG Mission Concepts and identifies a
 553 set of IA Cornerstones which define the high level IA capabilities required to support the GIG
 554 Mission Concepts. This document describes the technologies needed to support the GIG Mission
 555 Concepts and IA Cornerstones, but organizes these around the IA System Enablers. The
 556 technologies are organized by IA System Enablers because most technologies map to a single
 557 Enabler while they are associated with multiple IA Cornerstones. The Summary of this document
 558 describes which technologies are needed to support the system capabilities described in the
 559 Transition Strategy for each IA Cornerstone. Figure 1.3-1 depicts the GIG Mission Concepts, IA
 560 Cornerstones, and IA System Enablers.



561

562

Figure 1.3-1: (U) GIG Mission Concepts, IA Cornerstones, and IA System Enablers

563 (U//FOUO) The second category of information includes the IA standards in place today,
564 technology trends and forecasts available from commercial sources (i.e., Gartner, IDC), and
565 government trends and forecasts.

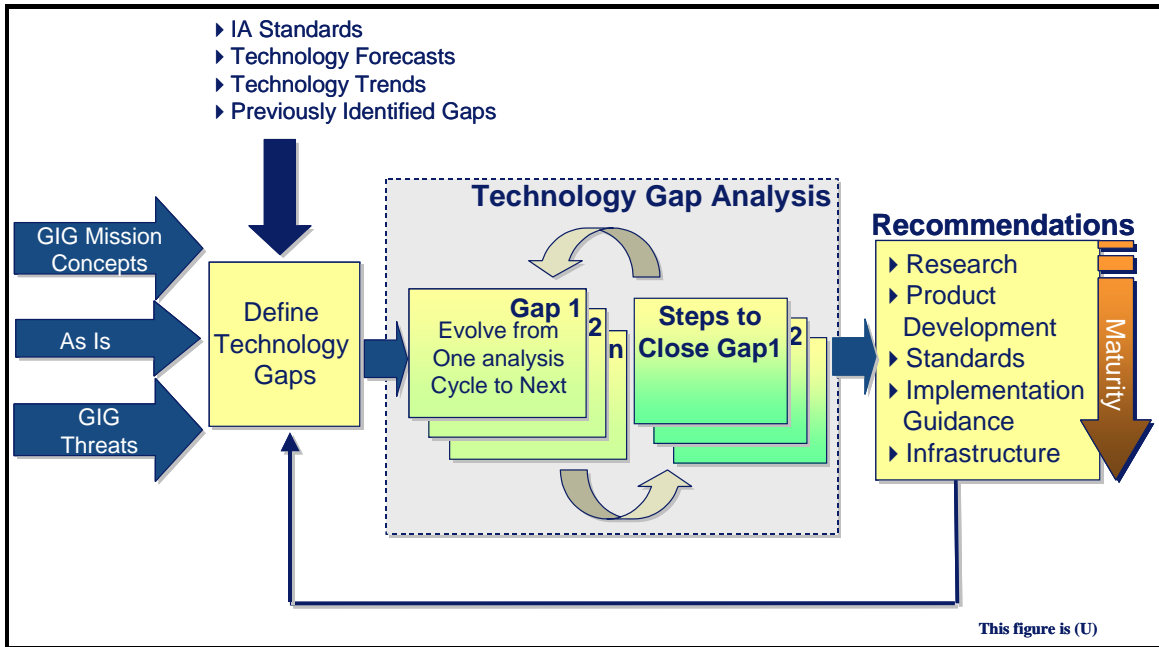
566 (U//FOUO) The third category of information consists of already defined gaps. The expectation
567 is that the process of developing the GIG IA Capability/Technology Roadmap document is an
568 iterative one. And the gaps identified today will drive various activities to close the gaps. These
569 activities could take the form of research, product development, standards implementation,
570 implementation guidance, and policy guidance. The technology development cycle to satisfy a
571 capability could encompass all the previously mentioned forms.

572 (U//FOUO) The document summary contains an indication of the technology improvement
573 needed to meet the transition strategy—defined in the RCD—for each Cornerstone. When
574 technologies are missing or unable to meet the strategy, the description highlights the impacted
575 operational capability.

576 (U//FOUO) Any recommendations could be in the form of the need for research, product
577 enhancements, new or enhanced standards, or new or enhanced infrastructure. These
578 recommendations, together with other background information and analysis in this document, are
579 intended to provide decision makers with the information needed for the following decisions:

- 580 • (U) Revise or write new standards and policies
- 581 • (U) Develop implementation guidance
- 582 • (U) Determine direction of research funding
- 583 • (U) Devise technology development strategies
- 584 • (U) Develop technology implementation plans

585 (U//FOUO) Figure 1.3-2 graphically represents the iterative development of the GIG IA
586 Capability/Technology Roadmap.



587

588 **Figure 1.3-2: (U) Iterative Development of the GIG IA Capability/Technology Roadmap**

589 (U) As a technology progresses through the development cycle, the current state of the
 590 development is input into the analysis process. The result of the analysis could be the closing of a
 591 gap or the identification of additional gaps between capabilities and the technology.

592 (U) The work of this document is an iterative process that will require re-analyses as additional
 593 capability needs are identified and technologies mature. Future releases of this document will be
 594 issued on an annual basis.

595 **2 (U) IA SYSTEM ENABLERS AND THEIR TECHNOLOGIES**

596 (U//FOUO) Information assurance (IA) is essential to meet the six GIG Mission Concepts.
597 Without IA, the GIG would not only fail to provide the right information, at the right time, at the
598 right place, in support of warfighting, business, enterprise information environment and national
599 intelligence, but the GIG could also be a haven for other nation states, cyber terrorists, hackers,
600 and malicious insiders to further disrupt operations in support of national objectives.

601 (U//FOUO) While a large body of technologies exist to at least partially provide the IA
602 capabilities stipulated by the GIG IA Vision, evolution of most existing technologies is needed,
603 and new technologies must be invented. This section of the document identifies the needed IA
604 technologies—both existing and to be developed. Preliminary assessments of technology
605 maturity and identified gaps are presented, which should aid decision makers in guiding existing
606 and starting new technology development efforts.

607 (U) For convenience of analysis and organization, the IA technologies are categorized and
608 presented in the context of the IA Enablers¹ they support. This “binning” into IA Enablers
609 ensures minimal technology overlap and complete coverage of the needed IA capabilities to
610 better support technical gap analysis. The IA Enablers used to organize the subsequent
611 subsections are:

- 612 • (U) Identification and Authentication
- 613 • (U) Policy-Based Access Control
- 614 • (U) Protection of User Information
- 615 • (U) Dynamic Policy Management
- 616 • (U) Assured Resource Allocation
- 617 • (U) Network Defense and Situational Awareness
- 618 • (U) Management of IA Mechanisms and Assets

619 (U//FOUO) Each subsection presents all aspects and benefits of the associated IA Enabler. The
620 IA Enabler itself is described, and key features of the IA Enabler are defined. An overview of the
621 supporting types of technologies is presented, organized around the IA Enabler. Each technology
622 is described in sufficient detail to support gap analysis. Finally, results of the technology gap
623 analyses are presented, and technology development timelines and recommendations for the IA
624 Enabler are provided. The technology timelines, showing the date that each technology will be
625 available for integration into the GIG, are optimistic; they are based on ideal circumstances, e.g.,
626 adequate funding and appropriate technical manpower are available to begin and execute the
627 recommended research, or existing developments continue as currently planned. Adverse
628 budgetary decisions will obviously delay the availability of the technologies for use.

¹ (U) The seven IA Enablers are core constructs that, together, provide the IA component of the GIG. These serve as architectural building blocks for enabling the GIG Mission Concepts.

629 (U//FOUO) A fairly comprehensive description is presented for each technology, but only to the
 630 extent needed to support recommendations for subsequent development efforts. Details of
 631 specific product implementations are avoided where possible to avoid conferring vendor
 632 endorsement, which distracts from the purpose of this analysis. Rather, numerous technology
 633 implementations were researched and considered, and only one or a small number were selected
 634 for inclusion in the roadmap, according to authors' opinions of how well these implementations
 635 represent the state of practice. In this context, specific items included for each technology are as
 636 follows:

- 637 • (U) Technical details: Description of the technology in terms of technical characteristics,
 638 features, and theory of operation. Consistent with the goals of the roadmap, the
 639 description may cover the superset of capabilities represented by the combination of a
 640 few related implementations or products.
- 641 • (U) Usage considerations: Discussion of potential implementation issues peculiar to the
 642 technology and anticipated operating environments, advantages of the technologies, and
 643 risks—in terms of potential threats and attacks—that might be incurred in employing the
 644 technology.
- 645 • (U) Maturity: Description of the current state relative to the goal capability of the
 646 technology itself. (This is not to be confused with the GIG IA capability that the
 647 technology would support.) While it is desirable to specify maturity of every technology
 648 in terms of Technology Readiness Level² (TRL), the roadmap does not attempt to do so,
 649 because either a TRL could not be found and there is insufficient information on which to
 650 base a specific estimate, or the analysis is based on multiple products/implementations
 651 that are each at different stages of development. Instead, then, the overall development
 652 stage of each technology is assessed and described by one of three maturity level terms:
 - 653 • (U) *Early* refers to technologies that are in the research or analysis phase
 654 (corresponding to TRL range 1-3).
 - 655 • (U) *Emerging* refers to those in the initial prototyping and lab demonstration phase
 656 (TRL range 4-6).
 - 657 • (U) *Mature* refers to technologies that are undergoing operational demonstration,
 658 production, and deployed operations (TRL range 7-9).

659 (U) In addition, specific TRLs are provided where they could be determined with a fair degree of
 660 certainty.

- 661 • (U) Standards: Discussion of standards that are pertinent to the technology. Included are
 662 existing standards, or those that will need to be developed in order to support the
 663 technology.
- 664 • (U) Costs/limitations: Discussion of the costs and limitations the technology would pose
 665 on the GIG architecture and connected systems when they are significant. Examples are

² (U) There are nine TRLs defined in Appendix 6 of DoD Instruction 5000.2, ranging from basic principals observed and reported (level 1) to actual system proven through successful mission operations (level 9).

666 where the technology would impose significant operational manpower burden (amount,
 667 caliber, and training), extraordinary procurement costs, undue complexity, unusual
 668 integration difficulties, adverse or significant impact on warfighting operations, or
 669 significant communications bandwidth or processing overhead.

- 670 • (U) Dependencies: List of related items, such as other technologies and data, upon which
 671 the technology must depend in order to provide the described capability.
- 672 • (U) Alternatives: List of possible alternative technologies or techniques that could
 673 support the IA Enabler, either for early adoption to provide an interim capability, or as a
 674 substitute if the described technology does not mature.
- 675 • (U) Complementary techniques: List of additional technologies or techniques that
 676 improve or enhance the described technology.

677 (U//FOUO) To facilitate discussions of the gap analyses, one or more matrices are provided for
 678 each IA Enabler. These are intended to summarize the explanations and show, at a glance, how
 679 adequately the analyzed technologies meet the capabilities defined by the IA Enabler for the
 680 2008 GIG implementation (Increment 1). Technologies are combined into categories for
 681 simplification. The adequacy level, determined by how well the sum of the assessed technologies
 682 in each category addresses each IA Enabler attribute, is described in Table 1.3-1. Capability
 683 attributes from the RCD are included in the matrices for reference.

684 **Table 1.3-1: (U) Definitions of Technology Adequacy Levels**

This Table is (U)		
Adequacy Level	Indication	Definition
Not Applicable	N/A	There is no expectation that the technology category could support the IA Enabler attribute.
Unknown	White	Technology investigation not completed, e.g., no result presented
Completely uncovered	Light gray	No technology is available, and no research is underway to develop the needed technology(ies), to address the IA Enabler attribute
Some coverage, but insufficient	Light black/white grid	R&D is underway that should lead eventually to at least partially covering the IA Enabler attribute, and anticipate that the resulting technology will be available in time to meet GIG IA milestone dates, <i>OR</i> A technology exists in the category that partially meets the needs of the IA Enabler attribute now, but additional technology R&D is needed to either enhance it or add to it in order to fully satisfy the attribute, <i>OR</i> Taken together, a combination of existing products or technologies in the group could satisfy the IA Enabler attribute now, but additional work is needed to combine the technologies in order to fully satisfy the attribute.
Fully adequate	Solid black	Technology, or a compatible combination of technologies, is available now that fully meets all aspects of the IA Enabler attribute, <i>OR</i> Technology development is underway and on schedule to fully satisfy the attribute at the time needed.
This Table is (U)		

685

686 (U//FOUO) Table 1.3-2 shows an example technology adequacy matrix for the Policy Based
 687 Access Control Enabler. Here, Digital Rights technologies are needed only to support the Object
 688 Life Cycle and Protection Profile attributes. This is indicated in the table by the black grid and
 689 gray shading under Digital Rights technologies under the Object Life Cycle and Protection
 690 Profile IA attributes and N/As under the Digital Rights technologies for all other IA attributes.
 691 Access Control Policy technologies are needed to support all IA Attributes, as shown by the
 692 black grid and gray shading. The white intersection of Access Control Policy technology and the
 693 Protection Profiles attribute indicates technologies are neither available nor research underway to
 694 satisfy the Protection Profiles attribute.

695 (U//FOUO) A matrix filled with black and “n/a” entries would reflect the ideal situation where
 696 all IA attributes are satisfied with technologies.

697 **Table 1.3-2: (U) Example of a Technology Adequacy Matrix**

This Table is (U)					
		Technology Categories			Required Capability (RCD attribute)
		Core Access Control	Digital Rights	Access Control Policy	
IA Attributes	Risk & Need Determination		N/A		IAAC4
	Math model		N/A		IAAC4
	Decision logic		N/A		IAAC1, IAAC4, IAAC7
	Ontology	N/A	N/A		IAAC4
	Exception handling		N/A		IAAC5
	Conflict resolution		N/A		
	Object Lifecycle				IAAC8
	Protection Profile				IAAC9
This Table is (U)					

698 2.1 (U) IDENTIFICATION AND AUTHENTICATION

699 (U//FOUO) I&A mechanisms provide critical IA foundations toward achieving the GIG Vision
700 of assured information sharing. In the assured sharing model, information is exchanged among
701 entities (e.g., individuals, devices) on the enterprise infrastructure. Similarly, services are shared
702 among entities on the enterprise infrastructure.

703 (U//FOUO) Access to information or services is based upon several factors including entity
704 properties, their authentication mechanism, properties of the objects to be accessed, the IT
705 components, the environment in which the entities exist, and the access control policy
706 implemented. All of this is based on the ability to uniquely identify the entities participating in
707 the exchange and the authentication mechanisms used by the entities participating in the
708 transaction. The ultimate goal is to support a SSO process independent of the many roles and
709 privileges of the entities involved.

710 (U//FOUO) The Identity and Authentication (I&A) Enabler is the sum of the mechanisms and
711 processes that result in a composite level of trust of an entity that can be used in all access
712 control decisions on the GIG during a given service request or login session. Entities that need to
713 be identified and authenticated include human users, workstations, networks, services, and other
714 resources.

715 (U//FOUO) The level of trust of an entity is referred to as its I&A Strength of Mechanism (SoM)
716 Score. Each service request is examined to determine how resistant the authentication of that
717 request is to impersonation or forgery. The likelihood that a service request was forged depends
718 on both the difficulty of forging the request, as measured by the I&A SoM, and the motivation
719 and ability of the adversary.

720 (U//FOUO) To support I&A SoM scoring on the GIG it is necessary to develop the following:

- 721 • (U//FOUO) Standards and technical requirements for assigning assurance levels for each
722 of the following factors that affect I&A strength and for deriving the I&A SoM score
723 from those factors:
 - 724 • (U//FOUO) Strength (resistance to compromise) of identity proofing during user
725 registration
 - 726 • (U//FOUO) Strength of the user's authentication token
 - 727 • (U//FOUO) Strength of the protocols used to authenticate service requests
 - 728 • (U//FOUO) Strength of the user's operating environment (e.g. clients, IT components,
729 and network).
- 730 • (U//FOUO) Mechanisms for conveying to services the assurance level of a specific
731 service request and of the IT components used to generate and process the request.
- 732 • (U//FOUO) Policies that make use of I&A SoM scores and other assurance measures in
733 the decision to grant or deny access to particular resources

734 **2.1.1 (U) GIG Benefits due to I&A**

735 (U//FOUO) The Information Assurance constructs used to support I&A provide the following
736 services to the GIG:

- 737 • (U//FOUO) Provides assurance that every entity participating in a GIG transaction is who
738 he/she/it claims
- 739 • (U//FOUO) Enables accountability for all GIG actions
- 740 • (U//FOUO) Accommodates varying trust levels for users and IT components by
741 identifying how much an entity can be trusted
- 742 • (U//FOUO) Enables capability for single sign-on (SSO) once the identity is recognized
743 and trusted throughout the GIG

744 **2.1.2 (U) I&A: Description**

745 (U//FOUO) Unique identity and identity proofing are fundamental to the I&A process. Unique
746 IDs are created for all entities (e.g. individuals, devices, services). Identity proofing refers to the
747 methods used to prove an individual's or devices identity before issuing a Unique ID. Identity
748 proofing mechanisms for individuals could range from providing no proof of identity presented
749 to requiring multiple picture IDs be presented in person by the individual receiving the Unique
750 ID. The identity registered for an individual is unique and remains constant despite changes of
751 that individual's name or other attributes. More information on Identity Management is provided
752 in Section 2.7, Management of IA Mechanisms and Assets

753 (U//FOUO) The authentication mechanism used in conjunction with this ID is also critical to
754 granting access to shared data and resources. The strength of the authentication mechanism
755 measures resistance to attempts to guess, sniff, extract, or otherwise compromise the entity's
756 authentication material. Current authentication mechanisms for human individuals include:

- 757 • (U) User ID and password
- 758 • (U) Use of software PKI certificates to provide a verifiable identity
- 759 • (U) Use of a Hardware Token that contains an entities PKI certificate and on-board
760 mechanisms to verify an entity's identity
- 761 • (U) Biometrics to unlock a token that protects the non-forgable PKI certificate
762 containing the identity that is shared in a protected manner during authentication
763 exchanges

764 (U//FOUO) The User Profile is a logical collection of information associated with a user, but it is
765 not necessarily stored in a single location. The identity management system must store a basic
766 user record containing the unique ID and the core identifying information that was verified (e.g.,
767 birth certificate information, driver's license number) or collected (biometrics) during identity
768 proofing. Other information that is logically part of the user profile must be strongly bound to
769 the user's unique ID but may be stored separately. For example, public key certificates used to
770 authenticate the user may be stored in a hardware token or certificate repository, role information
771 may be stored as signed attributes in a privilege server, contact information may be stored in a
772 user directory, and subscription information may be stored in a discovery server.

773 (U//FOUO) After registration, a user may log into a GIG asset using the authentication token
774 issued to that user. At the conclusion of the login process, an authenticated session would exist,
775 which has an associated I&A SoM session score. Authentication information for service requests
776 can either be derived from the user's login session or generated by the user's token for each
777 request. When the service provider can directly authenticate the user's original request, the
778 request assurance score can be determined directly based on the user and client assurance. But in
779 architectures where requests are passed through multiple providers, each of which can
780 authenticate only the preceding requestor, the originator's assurance score is decreased at each
781 intermediate processing point. In either case, the final request assurance score is determined
782 based upon the following factors:

- 783 • (U//FOUO) Identity Proofing method used to register the user
- 784 • (U//FOUO) Token used to authenticate the user's identity (e.g., password, software
785 certificate, hardware certificate, biometrics)
- 786 • (U//FOUO) Authentication mechanism used for the request or session (e.g., unbound
787 identity assertion, Secure Session Layer (SSL) session, signed request)
- 788 • (U//FOUO) The properties of the device used to logon (based upon their configuration
789 and management of the devices some devices may not be as trusted as others) and each
790 device in a trust chain between the originator and the provider
- 791 • (U//FOUO) The user's location or operating environment (e.g., highly trusted network,
792 remote access via a computer on the Internet)

793 (U//FOUO) Some participants in the GIG may require the entity and session to be periodically
794 re-authenticated. For example, if the data being retrieved is critical mission data, then the data
795 sharer may want to re-validate that the parameters of the original session login are still valid.
796 This may entail a requirement for the data requestor to provide their biometric data periodically
797 to ensure they are still present.

798 **2.1.3 (U) I & A: Technologies**

799 (U//FOUO) The following technology areas support the Identification and Authentication
800 Enabler:

- 801 • (U) Authentication Tokens
- 802 • (U) Biometrics
- 803 • (U) Device/Service Authentication
- 804 • (U) Authentication Protocols
- 805 • (U) Authentication Confidence
- 806 • (U) Single Sign-On (SSO)

807 (U) The three basic means of user authentication (and what they are based upon) are:

- 808 • (U) Authentication by knowledge (what a user knows, e.g., a fixed memorized password)
- 809 • (U) Authentication by characteristic (what a user is, e.g., a biometric)
- 810 • (U) Authentication by ownership (what a user has, e.g., a token)

811 (U) The main disadvantage of fixed passwords is that they are vulnerable to various attacks,
812 including social engineering, sniffing (e.g., network and/or electromagnetic emanations),
813 dictionary attacks, maliciously planted Trojan-horse software, etc. Once a user's fixed password
814 is compromised, it is impossible to detect subsequent system accesses by malicious parties.

815 (U) Section 2.1.3.1 discusses the token technologies that support authentication by ownership.
816 Biometric technologies are discussed in Section 2.1.3.2.

817

818 **2.1.3.1 (U) Authentication Tokens**

819 **2.1.3.1.1 (U) Technical Detail**

820 (U) Authentication tokens provide a means for a user to dynamically generate a single-use one-
821 time password (OTP) every time a remote secure system is accessed. This thus avoids fixed
822 password vulnerabilities. Tokens may be implemented in either hardware or software.

823 (U) A hardware token is a device, which the user in some manner employs to interface (either
824 physically or indirectly through user interaction) with a local client processor (e.g., a PC),
825 requiring secure access to a remote server processor or system. This hardware token contains the
826 critical security parameters for the authentication process.

827 (U) A software token is implemented within the local client processor itself and thus depends
828 upon the security and trustworthiness of the client's operating system. Examples of standard
829 OTP authentication protocols that are functional equivalents of software tokens include S/Key,
830 OPIE (One-Time Passwords in Everything, <http://inner.net/opie>), and SSH (Secure Shell).

831 (U) Most implementations of authentication tokens require the user to enter a PIN (personal
832 identification number) to locally unlock the token functionality (and thus are not subject to
833 network sniffing attacks). A PIN can be viewed as a primitive fixed and memorized password. A
834 biometric also can be used to unlock token functionality. This combination of independent
835 authentication factors provides a stronger authentication mechanism and prevents system
836 compromise if a hardware token is lost or stolen.

837 (U) Tokens function by using either Symmetric Key Authentication (a single shared secret key
838 known at both the client and server) or Public Key Authentication (where the client knows only
839 the private key, and the server knows the public key). All authentication tokens work by
840 producing dynamic single-use OTPs based upon credentials unique to the issued user and upon a
841 cryptographic algorithm or hash function. Symmetric Key Authentication and Public Key
842 Authentication are further explained in Section 2.1.3.4 which describes authentication protocols
843 in general.

844 (U) There are several basic token authentication modes under symmetric key, grouped within the
845 categories of Asynchronous and Synchronous.

846 **2.1.3.1.1.1 (U) Asynchronous Token Authentication Mode**

847 (U) Challenge-Response: In this mode, the user sends his username to the server in order to
848 identify the shared secret key. The server generates a random challenge and sends it back to the
849 user. The user keys the challenge into the token. This challenge is then cryptographically
850 processed with the secret key in order to generate a response. The response is then entered onto
851 the client and sent back to the server. The server independently does the same process and
852 compares results.

853 (U) This mode is 'asynchronous' because there is no (time-based) requirement that the response
854 arrive at the server within a prescribed and limited amount of time, nor is the response a function
855 of any underlying event counter.

856 **2.1.3.1.1.2 (U) Synchronous Token Authentication Mode**

857 2.1.3.1.1.2.1 (U) Time-driven

858 (U) In this mode, both user token and server generate an OTP based upon the shared secret key
 859 and an internal (network-synchronized) clock value. In order to permit network transmission
 860 time variations, the clock value resolution may be on the order of 60 seconds or less (to allow for
 861 clock drift). An example of this token type is SecurID by RSA. The user reads the varying time-
 862 based OTP from the LCD display of the hardware token (See Figure 2.1-1—Note the option for
 863 a 10-digit numeric keypad for entry of PIN to enable the token). The user then inputs this number
 864 onto the client processor, and it is sent to the server where it is compared with the server's
 865 expected value. A match yields successful authentication.



866

867 **Figure 2.1-1: (U) Examples of time-driven hardware tokens**

868 2.1.3.1.1.2.2 (U) Event-driven

869 (U) In this mode, instead of using time to create an OTP, an authentication event counter value is
 870 used with the shared secret key to generate the one-time password.

871 (U) Time and event driven modes are examples of response-only authentication schemes since
 872 the process only requires one-way transmission from user to server. A third version of response-
 873 only authentication is accomplished by both the user token and server. This creates a response
 874 from a hidden random challenge (rather than a mere time or counter value, which could be
 875 viewed as more predictable and certainly as monotonically increasing). This challenge is derived
 876 from the previous challenge, which is ultimately derived from some random seed value created
 877 at token initialization and known to both token and server.

878 (U) Authentication modes could be combined (e.g., challenge-response + time-driven + event-
 879 driven) to provide stronger authentication, just as stronger authentication is achieved by
 880 combinations of independent authentication factors (password/PIN + one or more biometrics +
 881 token), the.

882 2.1.3.1.2 (U) Usage Considerations**883 2.1.3.1.2.1 (U) Implementation Issues**

884 (U) Hardware tokens are implemented in a variety of physical form factors. Those that require
885 only indirect user-interaction (i.e., user observation of displays on the token, followed by manual
886 entry of data onto the local client processor) include pocket-style calculators and key fobs.
887 Hardware tokens that connect directly to the client processor include *smart* cards and Universal
888 Serial Bus (USB) tokens. An example of smart card authentication tokens is the DoD Common
889 Access Card (CAC), which can also be used as a photo ID card and for physical access control.

890 2.1.3.1.2.2 (U) Advantages

891 (U) In general, authentication tokens have the basic advantage that they can be used over public
892 networks and are not subject to compromise by hostile network sniffing, since the authentication
893 information is cryptographically based and unpredictable (i.e., not subject to standard replay
894 attacks).

895 (U) Hardware tokens are inherently resistant to social engineering attacks, since it is very
896 unlikely that an innocent user would provide an attacker with both the token and its enabling
897 PIN. Another obvious distinct advantage of hardware tokens is their portability, which enhances
898 the user's mobility and capability to authenticate remotely by home PCs, laptops, or personal
899 digital assistants (PDA).

900 (U) Smart cards (and USB tokens), since they interface directly with a user client, offer the
901 advantages that they can be used as a safe repository for sensitive personal data, such as PKI
902 credentials, passwords, and various account numbers. Smart cards have onboard processors,
903 which can do critical authentication processing (e.g., generating a cryptographic digital
904 signature) without being observed by a potential attacker (as opposed to the alternative of doing
905 the processing on a client processor, which may have been compromised by Trojan horse
906 software). Protection of sensitive information on the smart card when it is not being used is
907 accomplished by tamper-resistant—both physical and electronic—encryption of any stored data
908 and the required use of an enabling PIN.

909 2.1.3.1.2.3 (U) Risks/Threats/Attacks

910 (U) A basic disadvantage or risk of hardware tokens is that they can be lost or stolen. However,
911 in the case of smart cards such as the DoD CAC, the privileges of a lost card can be revoked or
912 canceled by the centralized PKI infrastructure authority. In addition, unless the enabling PIN is
913 also known by the malicious possessor of a lost/stolen token, that token can not be used in a
914 compromising manner.

915 (U) In the deployment of any authentication token system, especially in the case of an
916 organization like the DoD with large numbers of geographically dispersed users, secure token
917 distribution requires a robust proof of delivery (POD) mechanism (e.g., by manual signature for
918 non-repudiation).

919 (U) Public key authentication systems also suffer from potential risks if they have weak public
920 key management or certification. These systems rely on the clear and verifiable binding between
921 a user identity and the associated public key by the public key certificate. Only a trustworthy and
922 reliable certification/registration authority can assure that the certificate database is valid and up
923 to date.

924 (U) Another potential risk is that a hardware token can be left enabled at a client workstation,
925 which could allow a malicious intruder to masquerade as the valid user. A potential solution to
926 this might be to require periodic biometric checking/re-authentication.

927 (U) Besides the risk of potential attack where a hardware authentication token is physically taken
928 from the valid user—through loss, theft, or misplacement—there are further potential attacks on
929 the authentication process *at a distance* from the actual token itself. The classic attack would be
930 the *man-in-the-middle* attack against the collaborative process between the remote user and the
931 centralized authentication server. In this case, the attacker would have access to the
932 communications path somewhere on the network between the communicating parties. A man-in-
933 the-middle could inject, delete, or alter data that is sent in either direction. However, due to the
934 unpredictable cryptographically-based nature of the authentication responses sent by the user to a
935 server, it is unlikely that a man-in-the-middle could predict a future authentication value
936 response and thus could not gain access to the system.

937 (U) Another attack that could be mounted *at a distance* from the hardware token would be
938 planting malicious attacker Trojan horse modifications in the client workstation. This could be
939 partially avoided by having the authentication process done primarily within the token itself and
940 not allowing the shared symmetric secret key, or private key, to ever be transmitted off the token.
941 Finally, a guaranty that this secret key is safe can be made if some form of physical tamper
942 resistance is built into the token itself. Such tamper resistance would also prevent alterations to
943 any software that operates on the token itself.

944 (U) Of course, a token and its associated authentication function can be assumed to be secure
945 only if the main system authentication server is non-hostile and has not been compromised in
946 any manner. Thus, since the server is potentially the worst location for single point failure, the
947 most effort should be expended in safeguarding this resource.

948 **2.1.3.1.3 (U) Maturity**

949 (U) Authentication token technology has matured significantly, especially when each sub-
950 technology is viewed as an independent component. Current and future work needs to be done in
951 integrating the sub-technologies, along with the complementary authentication enhancing
952 technologies such as biometrics. An example of this is the DoD CAC with added biometric
953 functionality. In summary, the Technology Readiness Level of tokens can be thought of as
954 Mature (TRL 7 -9).

955 **2.1.3.1.4 (U) Standards**

956 (U) There are a variety of standards arenas—both formalized and actual—which play a role in
957 the development and evolution of authentication tokens and their underlying protocols and
958 algorithms.

959 **2.1.3.1.4.1 (U) Hardware Token Standards:**

960 (U) Organizations and arenas responsible for developing standards related to smart card
 961 technology and other tokens include RSA Labs Public-Key Cryptography Standards (PKCS),
 962 Microsoft Crypto API (CAPI), Personal Computer/Smart Card (PC/SC), and the ISO
 963 International Organization for Standardization. These standards are listed in Table 2.1-1

964

965

Table 2.1-1: (U) Hardware Token Standards

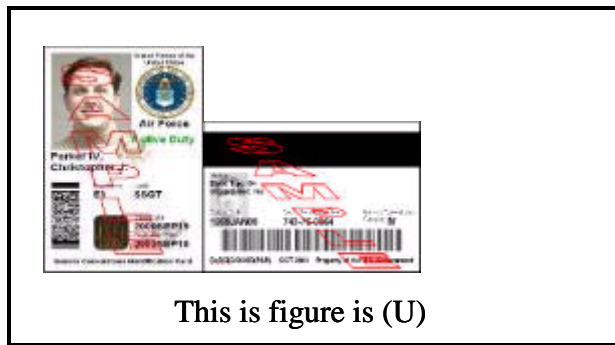
This Table is (U)	
Name	Description
RSA Labs PKCS Standards	
PKCS #11	Cryptographic Token Interface (<i>cryptoki</i>) Standard (specification of an application programming interface API for cryptographic token devices)
PKCS #12	Personal Information Exchange Syntax (specifies transfer syntax for personal identity information such as private keys and certificates, etc.)
PKCS #15	Cryptographic Token Information Format Standard (ensures interoperability of multiple vendor implementations)
Microsoft API Standards	
CAPI	Cryptographic Application Programming Interface standards
PC/SC Standards	
PC/SC Workgroup Specifications 1.0	Interoperability Specs for Smart Cards and PCs (platform and OS independent)
PC/SC Workgroup Specifications 2.0	Updated enhancements, including contactless (wireless RF) cards
ISO Standards	
ISO/IEC 7810	Identification Cards – physical characteristics
ISO/IEC 7811	ID Cards – Recording techniques
ISO/IEC 7812	ID Cards – Identification of issuers
ISO/IEC 7813	Financial transaction cards
ISO/IEC 7816	ID Cards with contacts
ISO/IEC 10373	ID Cards – Test Methods
ISO/IEC 10536	Contactless ID Cards – Close Coupled
ISO/IEC 14443	Contactless ID Cards – Proximity (<i>Mifare</i> cards) - 1-inch range
ISO/IEC 15693	Contactless ID Cards – Vicinity (<i>I.CODE</i> cards) - 5-inch range
This Table is (U)	

966

967 (U) The RSA PKCS specifications originated in the early 1990s from RSA Labs and, though
968 from a single company (versus a collaborative standards body), have been subsumed into and
969 adopted by numerous de facto and formalized standards. The PC/SC specifications are both PC
970 platform and PC operating system independent, while also specifying low-level device
971 interfaces. The updated version is addressing contactless/wireless smart card specifications. The
972 ISO smart card standards are derived from the basic ISO identification card standards. Similar to
973 the RSA PKCS #11, Microsoft's CAPI for Windows defines application programming interface
974 (API) for accessing tokens and letting vendors integrate security products into the OS—without
975 token developers having to write separate drivers for each application.

976 **2.1.3.1.4.2 (U) DoD Common Access Card**

977 (U) An emerging *standardized* authentication token within the Department of Defense is the
978 DoD Common Access Card (CAC), an example of which is shown in Figure 2.1-2:



979

980 **Figure 2.1-2: (U) DoD Common Access Card**

981 (U) The characteristics of the DoD CAC will define a de facto smart card standard, merely by
982 virtue of the vast number of CACs that will eventually be issued (to reserve and active military,
983 DoD civilian employees, and DoD contractors on DoD networks such as the emerging GIG).

984 (U) The main directed requirements of the CAC were that it provide for encryption of secure
985 messages, digital signature for non-repudiation, hardware token capability for storage of
986 cryptographic keys for use on unclassified networks, and flexible smart card technology to
987 support the efficient evolution of DoD identity-based business processes. Additional
988 requirements were that a CAC work as a photographic identification card, provide for facility
989 physical access control, provide logical access (with strong Identification authentication) to
990 unclassified DoD networks, and take advantage of the existing card-issuance infrastructure.
991 Hardware token smart cards are a solution that satisfies all of these requirements.

992 (U) In order to support legacy applications, the CAC includes both bar codes and magnetic
993 stripes. Current work is being done to include contactless/wireless versions of the CAC (for easy
994 facility/physical access applications).

995 (U) The CAC is a credit-card sized smart card that conforms to ISO/IEC standards 7810 and
996 7816. The basic processor CPU on the current version card is an 8-bit microcontroller (with
997 newer versions having up to 32-bit RISC reduced instruction set processors), the memory is 32K
998 EEPROM (plans for 64K), and the operating system is a Java API application programming
999 interface (to allow for a multiple vendor open architecture). Crypto processing is done by an
1000 1100-bit, advanced crypto engine (and a 112-bit/192-bit DDES-ECC crypto accelerator). Double
1001 DES (DDES) is used instead of single DES, which was originally used in many commercial
1002 token implementations. CAC vendors include Gemplus, Axalto, and Oberthur. The DoD
1003 implanted 3 Java applets on the card—the PIN security applet, the PKI applet, and a generic
1004 personal information management applet. DoD is currently looking at adding an enhanced
1005 biometric (e.g., fingerprint/thumbprint) capability directly onto the CAC.

1006 (U) Issuance of a new CAC to a DoD employee requires a user fingerprint template collection
1007 and verification and self-selection of an enabling PIN. At this time, about 82 percent of the over
1008 4.4 million potential CAC recipients have been issued their cards (with cards being issued at up
1009 to 12,000 a day). The CAC issuance infrastructure includes about 1,600 stations at more than 900
1010 sites around the world.

1011 (U) The DoD also plans to develop a central issuance facility. In order to complement the
1012 issuance of CACs, the DoD has already purchased more than 2 million stand-alone card readers
1013 for use with existing PC computers (with new PCs being purchased with embedded card
1014 readers).

1015 (U) Due to the extremely large number of DoD CACs being distributed and due to their
1016 application in sensitive areas and operations, much thought is going into the development and
1017 evolution of the CAC. Thus it can be viewed as a robust de facto implementation standard of a
1018 smart card token. It and its descendants will be important tokens in the future GIG.

1019 **2.1.3.1.5 (U) Cost/Limitations**

1020 (U) The cost and limitations of authentication tokens is based on both the token functionality
1021 itself and upon the required supporting infrastructure—both local (as in the case of requiring
1022 peripheral card readers for smart card tokens) and centralized (as in the case of a PKI Public Key
1023 Infrastructure with its associated Certificate Authority [CA]).

1024 (U) The concept of a PKI is very straightforward, but in order to implement a PKI that adaptively
1025 scales to support a large user population, large investments must be made and complexities
1026 overcome. One cost advantage of the DoD CAC smart card is that, by serving multiple legacy
1027 functions, it will enable the DoD to eliminate and phase out many legacy identity cards and
1028 thereby provide a larger than might be expected return on investment.

1029 (U) Symmetric single-key software tokens implemented on a user client PC are significantly
1030 cheaper than the equivalent hardware token implementation, since there is no hardware cost
1031 beyond the already existing PC. An imputed lower mental cost to the user is that much of the
1032 authentication process is hidden from the user. Another cost of software tokens is the lack of
1033 operating system independence.

1034 (U) Depending on how complex (and inherently costly) one wishes to make either smart cards or
1035 USB tokens when doing public key authentication, one can tradeoff the processing demands
1036 placed upon the token device and the host client. In the cheapest and simplest token, it can
1037 simply act as the repository of the private key that it can export to the client, which would then
1038 do any required cryptographic processing. The low monetary cost of this approach however
1039 incurs potentially high security risks (and costs) since the client PC must now be fully trusted to
1040 be impervious to malicious attacks (i.e., Trojan horses).

1041 (U) The alternate to this approach is to do cryptographic processing on the token itself (as on the
1042 DoD CAC). This may be done by either first generating the needed private key on a client
1043 workstation and then storing it on the token (Off-Token Key Generation) or by generating the
1044 private key only on the token itself (On-Token Key Generation).

1045 (U) The cost/limitation of Off-Token Key Generation is that it may temporarily expose the
1046 private key to potential hacking attacks on the client (although another advantage is that the user
1047 can make a backup copy of the key for disaster recovery purposes). The potential cost or
1048 limitation of On-Token private key generation is that if the token suffers a hardware failure, the
1049 private key may be lost forever. An example of a vendor product that does on-token key
1050 generation is the cryptographic smart card by Datakey Inc.

1051 (U) Despite their ease of security and convenience in carrying on one's key-chain, the fact that
1052 they can do onboard processing and storage, and the prevalence of USB ports on client
1053 computers, there are several inherent limitations and costs to USB authentication key-fob tokens.
1054 USB ports are often very inconveniently located on PCs (i.e., on the back panel of a PC tower).
1055 USB ports may not be physically robust enough to avoid being damaged by the repeated daily
1056 (or more often) interface with a USB token. And finally, a USB token is not large enough to
1057 easily incorporate a photo ID.

1058 (U) The DoD CAC has several current limitations. It was developed originally for use on the
1059 NIPRNet and not on systems that require higher assurance. For example, the CAC is not NIAP
1060 evaluated (specifically, a High Assurance Protection Profile does not currently exist), and it
1061 contains foreign COTS hardware and software (e.g., one of the vendors is Gemplus, a French
1062 manufacturer).

1063 (U) The GIG will require higher assurance tokens that provide a way to present identity
1064 credentials and authentication for access to classified information, which is an option not
1065 currently supported by the CAC. The three primary Java security applets (access control/PIN
1066 security, PKI support, generic information container management) need to undergo full high
1067 assurance security evaluation.

1068 (U) Plans are also being made to utilize asymmetric (public key) cryptography for the purpose of
1069 transport of keys and integrate this capability into the CAC issuing system by December, 2006.
1070 The January 2008 goal, to deliver a new DoD CAC compliant high assurance token that is
1071 manufactured only in the U.S with only U.S.-developed software, will provide a CAC that
1072 delivers high assurance Identification Management capabilities for the full suite of GIG
1073 customers including DoD, the Intelligence Community, and International Partners. This high
1074 assurance token will be able to carry classified information and Type I keying material.

1075 **2.1.3.1.6 (U) Dependencies**

1076 (U) Further evolution of the DoD CAC to include full biometric integration and
1077 contactless/wireless RF capability will rely on the full developing and maturing of the PC/SC
1078 Workgroup Specifications 2.0. Biometric integration also depends upon acceptance of a
1079 biometric technique (e.g., fingerprint).

1080 **2.1.3.1.7 (U) Alternatives**

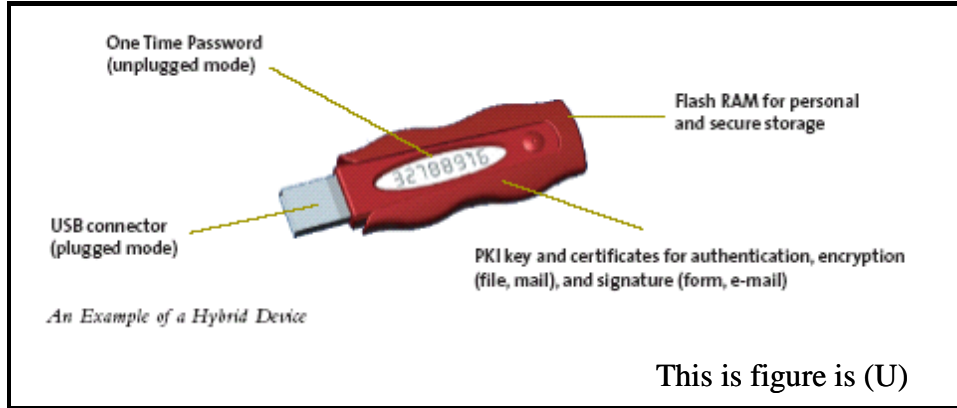
1081 (U) The basic stand-alone alternatives to tokens (what you have) are biometrics (what you are)
1082 and simple fixed passwords (what you know).

1083 **2.1.3.1.8 (U) Complementary Techniques**

1084 (U) Though biometrics and simple passwords (or PINs) can be viewed as mere alternatives to
1085 authentication tokens, they are better viewed as adjuncts or complementary techniques that when
1086 combined together have a multiplicative effect on an overall system security. Biometric data
1087 templates can be stored securely on a smart card token rather than on the client workstation.
1088 There is even the possibility of integrating an actual biometric fingerprint/thumbprint reader onto
1089 the surface of a smart card, thus eliminating the need for additional peripheral hardware.

1090 (U) The concept of an all in-one security device is an example where complementary techniques
1091 are combined. Security devices can embed many, if not all, base authentication methods. The
1092 intent is to create highly flexible and versatile security devices, such as for authentication,
1093 encryption, signing, secure storage, and physical access. Comprehensive functionality and
1094 personalization (e.g., personal storage) are essential to encourage users to embrace security
1095 devices such as a token on a key chain or a smart card in a wallet. By supporting multiple strong
1096 authentication methods, the same device becomes capable of interacting with a wide range of
1097 networks and applications.

1098 (U) The remote access scenario shows the benefit of integrating multiple authentication methods
1099 into one single security device. Figure 2.1-3 shows a USB token with either a PKI-enabled SIM
1100 chip inside or a smart card, with a display integrated within the reader to display the OTP. With
1101 this hybrid device, a user roams over a Wi-Fi network using SIM-based authentication. Once on
1102 the public network, the user can initiate a virtual private network (VPN) connection to a gateway
1103 using the RSA private key and certificate, which are stored in the token. Once the VPN tunnel is
1104 established, the user can log on to a portal to access the user's account through a Web
1105 interface—using the One Time Password generated by the token.



1106

1107

Figure 2.1-3: (U) Example of a Hybrid Device

1108

1109

1110

1111

1112

1113

1114

1115

1116

1117

1118

(U) An additional complementary technique would be mere physical access controls to secure facilities. Though this serves more as a member of an authorized group identification authentication than as an individual identification authentication, it is an important first barrier that must be overcome before a malicious intruder can get anywhere close to sensitive IT equipment. Indeed, the DoD CAC serves the dual purposes of both facility access (with both the photo ID feature and legacy magnetic stripe for card swipe-controlled physical accesses) and the follow-on required identification authentication for use of sensitive IT network resources. Other physical access control technologies are being researched that use facial scans to enable access to computer resources. An advantage to this approach is that it is a continual authentication, so each time the user leaves the computer locks the user's screen. When the user returns to the computer, the facial recognition authentication is repeated to re-authenticate access.

1119

2.1.3.1.9 (U) References

1120

(U) "One Time Passwords in Everything", <http://inner.net/opie>, by Craig Metz.

1121

1122

(U) "PKCS Public-Key Cryptography Standards", <http://www.rsasecurity.com/rsalabs/node.asp?id=2124> (RSA Labs).

1123

(U) PCSC Workgroup, <http://www.pcscworkgroup.com/>

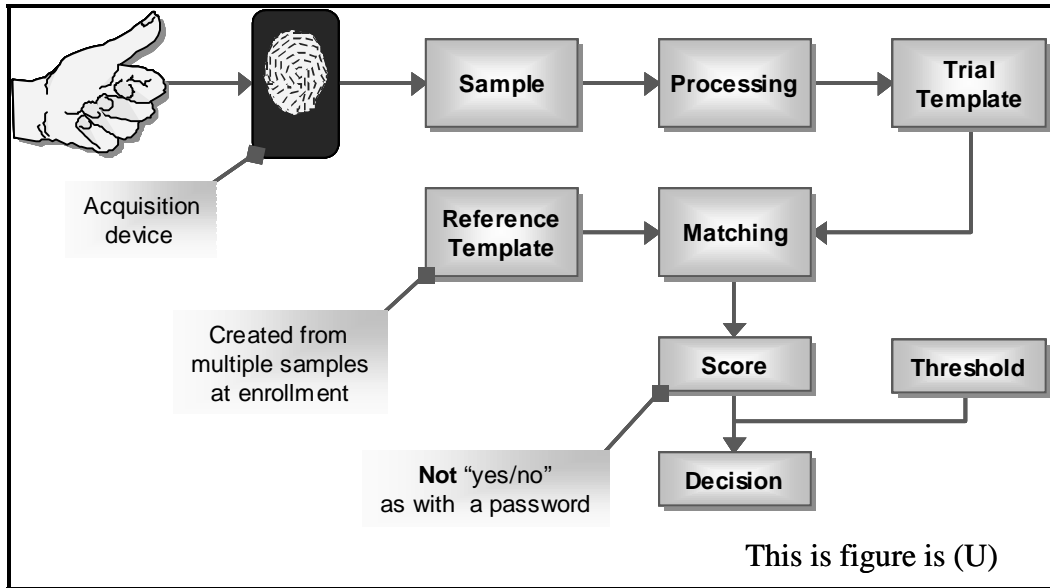
1124

1125

(U) "Multi-Biometric Verification Demonstration (Category: Secure Access to Physical Systems, Devices and Spaces)", Vijayakumar Bhagavatula, Dept of ECE, Carnegie Mellon.

1126 **2.1.3.2 (U) Biometrics**1127 **2.1.3.2.1 (U) Technical Detail**

1128 (U) A biometric is a measurable, physical characteristic or personal behavioral trait that can be
 1129 used to recognize the identity—or verify the claimed identity—of an enrollee.



1130

1131 **Figure 2.1-4: (U) Biometric System Block Diagram**

1132 (U) Two processes are necessary for any biometrics system: enrollment and verification.
 1133 Enrollment involves recording the user's biometric and storing it in the system as a template.
 1134 Verification is the comparison of a user's biometric against the reference template to verify a
 1135 user's identity. The enrollment process typically happens during system initialization or when a
 1136 new user is added to the system.

1137 (U) Biometric systems all perform the same basic process for verification, as illustrated in Figure
 1138 2.1-4. First a biometric acquisition device reads the user's biometric and creates a trial template.
 1139 A template is data that represents the biometric measurement of an enrollee used by a biometric
 1140 system for comparison against previously or subsequently submitted biometric samples. The trial
 1141 template is then compared against a reference template, previously stored during the enrollment
 1142 process.

1143 (U) If biometrics is used with other authentication factors, the reference template for the user's
 1144 claimed identity can be retrieved and compared against the trial template to verify the user's
 1145 identity; this is referred to as an authentication mode. If a biometric is the only authentication
 1146 factor, the trial template must be compared against all reference templates in the database until a
 1147 match is found; this is referred to as a recognition mode. The matching process is based on a
 1148 scoring system. The system must judge whether there is a close enough match between the trial
 1149 and reference templates.

1150 (U) The accuracy of a system is measured by its False Match Rate (FMR) and False Non-Match
1151 Rate (FNMR). The FMR is the probability that the biometric system will incorrectly identify an
1152 individual or will fail to reject an impostor. The FNMR is the probability that the biometric
1153 system will fail to identify an enrollee or verify the legitimate claimed identity of an enrollee.
1154 Generally the lower the FNMR the easier the system is to use while the lower the FMR, the
1155 better the security of the system. These characteristics are typically configurable by an
1156 administrator. For a biometric system that uses just one template matching attempt to decide
1157 acceptance, FMR is the same as False Acceptance Rate (FAR). When multiple attempts are
1158 combined in some manner to decide acceptance, FAR is more meaningful at the system level
1159 than FMR. For a biometric system that uses just one attempt to decide acceptance, FNMR is the
1160 same as False Rejection Rate (FRR). When multiple attempts are combined in some manner to
1161 decide acceptance, FRR is more meaningful at the system level than FNMR.

1162 (U) During enrollment some biometric systems perform multiple scans of the same biometric to
1163 create the reference template. This can create a more accurate reference template and help reduce
1164 the FMR and FNMR.

1165 (U) Accuracy is also driven by the amount of data collected or the number of data points
1166 collected in the reference sample. This also contributes to storage requirements: more data points
1167 means more storage capacity is required, which translates into more cost.

1168 (U) Reliability is affected by aging and environmental conditions. Injuries and background noise
1169 could affect the accuracy of the devices and increase the FNMR.

1170 (U) There are many biometric factors that can be used. They are generally broken down into two
1171 categories: physiological and behavioral. Physiological biometrics is usually derived from a
1172 person's anatomy and are difficult to alter. Examples include fingerprints, iris, and hand print.
1173 Behavioral biometrics are derived from an action performed by an individual. Behavioral
1174 biometrics are usually easier to alter but can be perceived as less intrusive by the user. Examples
1175 of behavioral biometrics include signature, voice recognition, and gait.

1176 **2.1.3.2.1.1 (U) Physiological Biometrics**

1177 2.1.3.2.1.1.1 (U) Fingerprint Recognition

1178 (U) The patterns of friction ridges and valleys on an individual's fingertips are unique to that
1179 individual. For decades, law enforcement has been classifying and determining identity by
1180 matching key points of ridge endings and bifurcations. Fingerprints are unique for each finger of
1181 a person including identical twins.

1182 (U) Fingerprint recognition is the most widely available biometric technology. Fingerprint
1183 recognition devices for desktop and laptop access are now widely available at a low cost from
1184 many different vendors. With these devices, users no longer need to type passwords—instead;
1185 only a touch provides instant access. Fingerprint systems can also be used in identification mode.
1186 Several states check fingerprints for new applicants to social services benefits to ensure
1187 recipients do not fraudulently obtain benefits under fake names.

1188 2.1.3.2.1.1.2 (U) Face Recognition

1189 (U) The identification of a person by the facial image can be done in a number of different ways.
1190 It can be done by capturing an image of the face in the visible spectrum, using an inexpensive
1191 camera, or by using the infrared patterns of facial heat emission. Facial recognition in visible
1192 light typically models key features from the central portion of a facial image. Using a wide
1193 assortment of cameras, the visible light systems extract features from the captured image(s) that
1194 do not change over time while avoiding superficial features such as facial expressions or hair.
1195 Several approaches to modeling facial images in the visible spectrum are Principal Component
1196 Analysis, Local Feature Analysis, neural networks, elastic graph theory, and multi-resolution
1197 analysis. The major benefits of facial recognition are that it is non-intrusive, hands-free,
1198 continuous, and is acceptable to most users.

1199 (U) Some of the challenges of facial recognition in the visual spectrum include reducing the
1200 impact of variable lighting and detecting a mask or photograph. Some facial recognition systems
1201 may require a stationary or posed user in order to capture the image, although many systems use
1202 a real-time process to detect a person's head and locate the face automatically.

1203 2.1.3.2.1.1.3 (U) Iris Recognition

1204 (U) The iris of the eye is the colored area that surrounds the pupil. Iris patterns are unique. The
1205 iris patterns are obtained through a video-based image acquisition system. Iris scanning devices
1206 have been used in personal authentication applications for several years. Systems based on iris
1207 recognition have substantially decreased in price, and this trend is expected to continue.

1208 (U) The technology works well in both verification and identification modes (in systems
1209 performing one-to-many searches in a database). Current systems can be used even in the
1210 presence of eyeglasses and contact lenses. The technology is not intrusive. It does not require
1211 physical contact with a scanner. Iris recognition has been demonstrated to work with individuals
1212 from different ethnic groups and nationalities.

1213 2.1.3.2.1.1.4 (U) Hand and Finger Geometry

1214 (U) These methods of personal authentication are well established. Hand recognition has been
1215 available for over twenty years. To achieve personal authentication, a system might measure
1216 physical characteristics of either the fingers or the hands. These include length, width, thickness,
1217 and surface area of the hand. One interesting characteristic is that some systems require only a
1218 small biometric sample (a few bytes).

1219 (U) Hand geometry has gained acceptance in a range of applications. It can frequently be found
1220 in physical access control in commercial and residential applications, in time and attendance
1221 systems, and in general personal authentication applications.

1222 **2.1.3.2.1.2 (U) Behavioral Biometrics**

1223 2.1.3.2.1.2.1 (U) Signature Verification

1224 (U) The technology is based on measuring the speed, pressure, and angle used by the person
1225 when a signature is produced. One focus for this technology has been e-business applications and
1226 other applications where signature is an accepted method of personal authentication.

1227 2.1.3.2.1.2.2 (U) Speaker Recognition

1228 (U) Speaker recognition has a history dating back some four decades, where the output of several
1229 analog filters was averaged over time for voice matching. Speaker recognition uses the acoustic
1230 features of speech that have been found to differ between individuals. These acoustic patterns
1231 reflect both anatomy (e.g., size and shape of the throat and mouth) and learned behavioral
1232 patterns (e.g., voice pitch, speaking style). This incorporation of learned patterns into the voice
1233 templates (the latter called voiceprints) has earned speaker recognition its classification as a
1234 behavioral biometric.

1235 (U) Ambient noise levels can impede collection of the initial and subsequent voice samples.
1236 Performance degradation can result from changes in behavioral attributes of the voice and from
1237 enrollment using one telephone and verification on another telephone. Voice changes due to
1238 aging also need to be addressed by recognition systems. Many companies market speaker
1239 recognition engines, often as part of large voice processing, control, and switching systems.
1240 Capture of this biometric is seen as non-invasive. By using existing microphones and voice-
1241 transmission technology to allow recognition over long distances by ordinary telephones (wire
1242 line or wireless), this technology needs little additional hardware.

1243 **2.1.3.2.2 (U) Usage Considerations**

1244 (U) There are two typical implementations for deploying a biometric system: using a centralized
1245 database for storing user reference biometric templates (recognition mode) or storing the
1246 biometric value directly on a token the user possesses (authentication mode).

1247 **2.1.3.2.2.1 (U) Implementation Issues**

1248 (U) Recognition mode uses a centralized database containing all enrolled users' reference
1249 templates. A user presents himself/herself at the biometric reader for authentication. The reader
1250 collects the biometric, digitizes it, and sends it over the network from the client (directly
1251 connected to the reader) to a Biometric Authentication Database. The comparison and
1252 acceptance/rejection of the fingerprint/face/etc. is made there, and the acceptance or rejection
1253 notice is sent back to the client. If a match is verified, the user is allowed to access the various
1254 resources on the network.

1255 (U) Authentication mode typically stores the biometric value directly on the user's token. In this
1256 case there is no central database. Rather, the user feeds a hardware token into the reader, and
1257 then presents the fingerprint, face, etc., for reading. The reader is a trusted device that compares
1258 the measured biometric directly with the value stored on the presented token.

1259 (U) Biometrics may not be suitable for every environment. For example, users in tactical
1260 environments may have difficulty using a fingerprint reader since their fingers might get dirty or
1261 cut or their protective clothing may preclude access to the biometrics reader. Carrying a large
1262 biometric reader with a handheld device may limit the device's mobility. Hence, use of a
1263 particular biometric must be weighed against its operational environment. The authentication
1264 confidence associated with biometrics must consider the applicability of the authentication
1265 mechanism for the environment in question.

1266 2.1.3.2.2.2 (U) Advantages

1267 (U) The time required to perform a match in authentication mode is much less than in
1268 recognition mode because the trial template must only be matched against a single reference
1269 template. The time necessary to perform the recognition process is driven by the size of the
1270 template database and the size of the template. The more users enrolled in a system the longer it
1271 will take to perform a match in the database. Also the larger the template the longer a positive
1272 match will take. Using biometrics as one of several authentication factors increases the strength
1273 of the authentication, and because the biometric system can be used in authentication mode
1274 versus recognition mode, it should not impact system performance.

1275 (U) To access some information in the GIG, multifactor authentication may be required.
1276 Biometrics can play an important role in providing a higher authentication score than a simple
1277 user name and password. They can also be used to unlock a user's privileges or other
1278 authentication information. Biometrics also assist in providing an audit function as they can
1279 uniquely identify a user and enable the system to tie the user to performed actions.

1280 2.1.3.2.2.3 (U) Risks/Threats/Attacks

1281 (U) With the recognition mode implementation, an adversary does not need to attack the reader,
1282 but rather the network or the biometric database. The biometric template must be secure as it
1283 crosses the network. If the template can be captured, an adversary can present it to the biometric
1284 database and impersonate an authorized user. This can be avoided by securing the connection
1285 between client and database by using protocols such as IPsec or TLS, which includes replay
1286 protection.

1287 (U) The database itself also is a target for attack. If the database can be compromised, all
1288 reference templates stored on it are also compromised. The database is likely to be riding on an
1289 OS that can be exploited through a variety of methods, much like attackers on the Internet
1290 capture credit card databases today. Alternatively, an attacker can use the weakness to replace
1291 the stored value with his own value, thus granting him access while completely eliminating the
1292 legitimate user from the system.

1293 (U) The difference between this biometric attack and credit card attacks is that biometric
1294 templates are very difficult to revoke. If an attacker captures a set of credit card numbers, those
1295 cards can be revoked and new cards issued. Or if an attacker captures a set of private encryption
1296 keys from a PKI, the certificates corresponding to those keys can be revoked and new
1297 keys/certificates issued. While there is some pain and expense in the revocation operation, the
1298 procedures and methods are known.

1299 (U) Contrast this with an attack that captures the digital fingerprints of the user base. The
1300 attacker now has the digitized fingerprints and can inject them into the system as needed to
1301 impersonate a user. It is not practical to have users get new fingerprints; the only option is to
1302 throw out the existing biometric solution and replace it with a new one (e.g., a new method of
1303 digitizing fingerprints that bears no relation to the other one and cannot be derived from it or
1304 switch to using face recognition instead of fingerprints).

1305 (U) To defend against these attacks, a number of steps must be taken:

- 1306 • (U) The digitized image must be some transform of the actual biometric that cannot
1307 easily be reversed. For example, the value sent, stored and compared would be a SHA-1
1308 hash of the digitized fingerprint. If this were to be captured, it would be replaced with a
1309 SHA-2 hash of the face, etc.
- 1310 • (U) Each use of the biometric should include some unique value (e.g., time stamp)
1311 hashed in with the actual value to protect against replay attacks. This is a trade-off, as the
1312 goal would be to use a biometric value for an entire session (e.g., only capture the
1313 fingerprint once, then let the user work for a few hours), and replay attacks can
1314 potentially be done whenever the time is still within the legal window of use of the
1315 biometric.
- 1316 • (U) As indicated above, the communication between the computer connected to the
1317 biometric reader and the central database must be secured, for example, using TLS,
1318 IPsec, or equivalent security.
- 1319 • (U) The computer on which the Central Database resides must be secured to the
1320 maximum extent possible.
- 1321 • (U) Protected Resources must also be secured. They must be able to authenticate all
1322 accesses by users—they should be able to tell from where an access arrives in case it is
1323 attempted by an attacker who has compromised the system.

1324 (U) The authentication mode implementation avoids the network and operating system-based
1325 vulnerabilities described above; however, it presents a number of its own potential
1326 vulnerabilities. Chiefly, these relate to the tamper resistance of the hardware token—if an
1327 attacker can acquire the token and replace the stored value with his own value, he will be
1328 approved by the system.

1329 (U) Other vulnerabilities with this approach relate to how the biometric reader communicates
1330 successful matching to the system. If an attacker can simply forge a successful match message
1331 from the reader to the protected resources, the attacker is in the system again.

1332 **2.1.3.2.3 (U) Maturity**

1333 (U) The Gartner Hype Cycle lists two to five years to reach the plateau/adoption. The plateau is
1334 defined as “the real-world benefits of the technology are demonstrated and accepted.” Gartner
1335 lists several factors, which determine the maturity level. User acceptance is one of the primary
1336 factors along with ease of use, accuracy, reliability, resistance to attack, and cost.

1337 (U) User acceptance is a concern with iris and retina scanning, because of a general fear people
1338 have about instruments close to their eye. The accuracy of iris and retina scanning is reasonably
1339 good, but the cost is high for scanning equipment. Voice and signature recognition are neither as
1340 intrusive as iris and retina scanning nor as expensive, but are not as accurate and require more
1341 effort to use. Fingerprint, face, and hand recognition fall in between in terms of intrusiveness,
1342 accuracy, and expense.

1343 (U) IDC lists three main challenges to adoption of biometrics authentication: convenience,
 1344 installation, and portability. Convenience translates into ease of use in Gartner's terms while
 1345 installation is really a cost factor, which includes time and money. Portability is something
 1346 Gartner does not discuss.

1347 (U) IDC describes portability as how easy is the biometric device to carry around. If the
 1348 biometric device is cumbersome to carry, people will refuse to use it.

1349 (U) Gartner lists the following obstacles to biometrics technology:

- 1350 • (U) Biometric equipment is expensive to buy and install
- 1351 • (U) Applications have to be changed
- 1352 • (U) None of the biometrics devices are fool proof
- 1353 • (U) Accuracy can be affected by aging, injury, or environmental conditions

1354 (U) There are several initiatives that may accelerate the biometric development market. For
 1355 example, a trusted traveler program is being lobbied for to move people through airports quickly
 1356 and to improve security. One of the fundamental pieces to a trusted traveler program is
 1357 biometrics. Travelers must be authenticated as they move through the transportation system.
 1358 While a trusted traveler program is still being debated in Congress, a pilot program is underway.
 1359 Developments related to the trusted traveler program could accelerate the biometrics market.

1360 (U) When it comes to the core algorithms and mechanisms involved, the Technology Readiness
 1361 Level of biometric technologies in general can be thought of as nearing the Mature level (TRL7-
 1362 9).

1363 **2.1.3.2.4 (U) Standards**

1364 (U) Standards applicable to biometrics are listed in Table 2.1-2.

1365 **Table 2.1-2: (U) Biometric Standards**

This Table is (U)	
Standard	Description
Common Biometric Exchange Formats Framework (CBEFF)	CBEFF originally stood for Common Biometric Exchange File Format and was originally developed by the Biometric Consortium (BC). It was published by NIST as NISTR 6529. CBEFF defines a standard method for identifying and carrying biometric data. It describes a framework for defining data formats that facilitate the communication of biometric data. CBEFF does not specify the actual encoding of data (e.g., bits on a wire) but provides rules and requirements and the structure for defining those explicit data format specifications.
This Table is (U)	

This Table is (U)	
Standard	Description
BioAPI	<p>The BioAPI standard defines an Application Program Interface (API) and a Service Provider Interface (SPI) for standardizing the interaction between biometric-enabled applications and biometric sensor devices. The API provides a common method for applications to access biometric authentication technology without requiring application developers to have biometric expertise. The SPI allows the production of multiple BSPs (Biometric Service Providers) that may be used by an application without modification of that application, regardless of biometric technology.</p> <p>The BioAPI Consortium originally developed the BioAPI specification. The BioAPI Consortium is a group of over 50 organizations focused solely on furthering a standard biometric API. M1 has taken the resulting specification from the consortium and standardized it nationally as ANSI INCITS 358-2002. M1 has also contributed ANSI INCITS 358-2002 to SC 37 where it is currently a draft international standard.</p>
Data Interchange Formats	<p>A data interchange format specifies the low-level format for storing, recording, and transmitting biometric information. This biometric information may be unique to each biometric characteristic (e.g., fingerprint, iris, signature) and/or to each method of capture (e.g., photograph, capacitive sensor). In some technologies, this biometric information is called a <i>template</i>. M1.3 is currently working on projects dedicated to standards for the following formats.</p>
Biometric Profiles	<p>A biometric profile identifies a set of base biometric standards that apply to a single application or scenario. The profile then identifies the appropriate configurations, parameters, and choices for options provided within those specifications. The goal is to provide interoperability and consistent functionality and security across a defined environment.</p> <p>M1.4 is engaged in the following projects:</p> <ul style="list-style-type: none"> • Interoperability and Data Interchange—Biometric Based Verification and Identification of Transportation Workers • Interoperability, Data Interchange and Data Integrity—Biometric Based Personal Identification for Border Management • Point-of-Sale Biometric Verification/Identification <p>SC 37 has defined a functional architecture that serves as part one of a multi-part standard. SC 37 is also working on the first profile of the standard titled <i>Biometric Profile for Employees</i>.</p>
Biometric Evaluation Methodology	<p>The Biometric Evaluation Methodology (BEM), Version 1.0, was designed to aid security evaluators who were attempting to evaluate biometric products against the Common Criteria (CC). The Common Evaluation Methodology (CEM) used in CC evaluations does not address the environmental, user population, and other issues that have an impact on a biometric implementation. The BEM specifically addresses these issues as they apply to biometric technology evaluations under the CC.</p> <p>Evaluators, certifiers and developers from Canada, U.K., GERMANY, U.S., Italy, Sweden, and others developed the BEM. Version 1.0 of BEM was released in August of 2002.</p>
This Table is (U)	

This Table is (U)	
Standard	Description
Biometrics Protection Profile	<p>The CC is an effort of the US, Canada, and European countries to establish a common set of security criteria by which to evaluate IT products. This effort has resulted in an international standard (ISO/IEC 15408-1) for evaluating IT security products. The document that establishes the implementation-independent security requirements for a given category of product is called a Protection Profile. Currently, the DoD Biometrics Management Office (BMO) and the National Security Agency (NSA) are developing four Protection Profiles for biometrics products:</p> <ul style="list-style-type: none"> • Robustness Biometric PP for Verification Mode • Basic Robustness Biometric PP for Verification Mode • Medium Robustness Biometric PP for Identification Mode • Basic Robustness Biometric PP for Identification Mode
Biometric API for JavaCard	<p>The JavaCard Forum was established in 1997 to promote Java as the preferred programming language for multiple-application smart cards. A subset of the Java programming language was proposed for these cards and resulted in a standard for a JavaCard API. The JavaCard Forum has extended the JavaCard API to enroll and manage biometric data securely and facilitate a match on card capability with the Biometric API for JavaCard. The Biometric API manages templates, which are stored only in the card. During a match process, no sensitive information is sent off the card.</p>
Common Data Security Architecture (CDSA), Human Recognition Services Module	<p>The Human Recognition Services Module (HRS) is an extension of the Open Group's Common Data Security Architecture (CDSA). CDSA is a set of layered security services and a cryptographic framework that provides the infrastructure for creating cross-platform, interoperable, security-enabled applications for client-server environments. The biometric component of the CDSA's HRS is used in conjunction with other security modules (i.e., cryptographic, digital certificates, and data libraries) and is compatible with the BioAPI specification and CBEFF.</p>
This Table is (U)	

1366 **2.1.3.2.5 (U) Cost/Limitations**

1367 (U) Biometrics can provide an enhanced authentication capability but they have several costs
 1368 associated with them. First, biometric readers must be deployed on the system. This may be a
 1369 substantial cost depending on the cost per reader and the number of readers required. In the GIG,
 1370 it is envisioned that many systems will require biometric authentication, and therefore a large
 1371 number of readers will be required.

1372 (U) There are several processes that require administration in a biometric system and therefore
 1373 add to the maintenance cost of the system. One of these processes is enrollment, which incurs a
 1374 cost both upon the central administrator and upon the user.

1375 (U) Another limitation of biometrics is the user's acceptance. This is influenced by the perceived
 1376 intrusiveness of the biometric. For example, signatures are widely accepted today, and a user
 1377 would be far less likely to mind a signature biometric than an iris or retina scan that requires
 1378 them to put their eye close to the biometric reader. If the users will not accept the use of the
 1379 particular biometric technology, it cannot be expected to be successful.

1380 **2.1.3.2.6 (U) Alternatives**

1381 (U) Alternatives for biometrics include any information that can be used to verify a user's
1382 identity. For example, Government issued photo identification may be substituted for a biometric
1383 for applications such as physical access to a building. However, it alone is not adequate to
1384 authenticate access to an information system.

1385 (U) Another alternative to biometrics is to require more information that the user knows. For
1386 example, if a biometric is not available, inputting several passwords may be sufficient to
1387 authenticate the user.

1388 **2.1.3.2.7 (U) Complementary Techniques**

1389 (U) Hardware tokens are complementary to biometric implementations using the authentication
1390 mode.

1391 **2.1.3.2.8 (U) References**

1392 (U) Biometric Authentication Perspective (Gartner)

1393 (U) Hype Cycle for Information Security, 2003 (Gartner)(U) "Reduced Complexity Face
1394 Recognition using Advanced Correlation Filters and Fourier Subspace Methods for Biometric
1395 Applications", by M. Savvides, PhD Thesis, May 2004, Electrical & Computer Eng, Carnegie
1396 Mellon University

1397 2.1.3.3 (U) Device/Service Authentication

1398 (U//FOUO) Security and trust in any network is a function of all the elements that make up a
1399 network. This includes end-point (client and server) devices that can impersonate users and
1400 organizations. As network devices proliferate (e.g., mobile phones, PDAs, portable digital music
1401 players, set-top boxes, and laptops), the ability to distinguish between trusted and rogue devices
1402 becomes a fundamental security requirement.

1403 (U) Since an authenticated device can act as the root of trust, it can also provide the security
1404 foundation for a new breed of applications, such as identity based anti-virus solutions and digital
1405 information rights management software. From this standpoint, device and service authentication
1406 is a core requirement of any strong identification management strategy.

1407 (U) There are a variety of initiatives and incentives/motivations that are driving industry towards
1408 robust device authentication, including the following:

- 1409 • (U) Transform today's mobile devices (e.g., cell phones, PDAs, laptops) into strong
1410 authentication devices
- 1411 • (U) Propagate device credentials, strong authentication algorithms, and authentication
1412 client software across many network end points (e.g., desktop computers, servers,
1413 switches, Wi-Fi access points, set-top boxes)
- 1414 • (U//FOUO) Enhance device credentialing management schemes for improving SSO in
1415 the GIG, or at least to help reduce Sign-On problems
- 1416 • (U) Build around well-established infrastructure components such as directory and
1417 RADIUS servers
- 1418 • (U) Proliferate low-cost, multi-function authentication devices (e.g., tokens, smart cards)
- 1419 • (U) Facilitate native support (e.g., platform connectors) for strong device and user
1420 authentication in application development and identification management platforms
- 1421 • (U) Leverage federated identity protocols as a powerful propagation and integration
1422 mechanism
- 1423 • (U) Enable best-of-breed solutions through interoperable components
- 1424 • (U) Credentials and Security Devices

1425 **2.1.3.3.1 (U) Technical Detail**

1426 **2.1.3.3.1.1 (U) Universal Strong Authentication for Devices**

1427 (U) The strength—the trustworthiness—of an identity depends on multiple factors. The initial
1428 authentication process (i.e., identity verification), the type of credential being issued (i.e.,
1429 security token), and the depth of the relationship between the authenticator and the authenticated
1430 entity all contribute to the strength of an identity. Beyond the authentication process, the security
1431 policies enforced by the authentication authority and its operation best practices have a direct
1432 impact as well.

1433 (U) Strong identification management must take into account technology, policy, and operational
1434 issues. Strong authentication is the first level of trusted networks where identities can be securely
1435 shared and trusted across independent partners. It is the foundation for a more secure network,
1436 one in which all people and all devices are strongly authenticated in an open, interoperable, and
1437 federated environment.

1438 (U) Three methods specify the core types of authentication credentials—SIM secret and X.509
1439 certificate. Each of these methods has a specific use in an interoperable environment:

- 1440 • (U) SIM-based authentication – SIM (Subscriber Identity Module). This authentication
1441 method predominates in telecommunications. It also is emerging as an important
1442 authentication method in public Wi-Fi networks (authentication and roaming across
1443 Global System for Mobile Communications/General Packet Radio Service and 802.11
1444 networks).
- 1445 • (U) PKI-based authentication – PKI is a fundamental security component of all major
1446 Internet protocols for authentication and communication (e.g., Transport Layer Security
1447 [TLS], WS-Security, IPsec IKE, 802.1x, Session Initiation Protocol [SIP]). The choice of
1448 X.509v3 certificates as strong credentials is also consistent with deployment trends in
1449 enterprise and government markets. Furthermore, certificates offer additional security
1450 functionality beyond authentication, for example for electronic form and e-mail signing
1451 and file encryption. It should also be noted that there are ongoing developments within
1452 PKI/KMI to specify not just devices in the Directory Information Tree, but also services,
1453 servers and roles.

1454 **2.1.3.3.2 (U) Usage Considerations**

1455 (U) When describing authenticating a device, it is important to consider to what the device is
1456 authenticating. In the case of 802.1x, the device is being authenticated at the link layer. In the
1457 case of a call setup on a mobile phone network, the authentication occurs at an application level.
1458 Sometimes authentication will need to be done on a per connection basis (such as on a point-to-
1459 point link). Other times, authentication will need to be done at an enterprise level for auditability
1460 and scalability purposes.

1461 (U) Each of these different scenarios implies a different mechanism to perform device
1462 authentication. This can lead to many overlapping (and potentially conflicting) protocols and
1463 processes.

1464 2.1.3.3.2.1 (U) Advantages

1465 (U) Secure device authentication enables many other security goals of GIG-related technologies.
1466 By also authenticating a device that a user is interacting with, the entire system has a higher
1467 degree of confidence in the authenticated session. By authenticating a device in a data center
1468 communicating with another unmanned device, services such as web services can use the
1469 identity of a device as a foundation for trust in the end-to-end system. Device authentication
1470 permits secure access to networks, applications, and any other GIG-connected resources.

1471 2.1.3.3.2.2 (U) Risks/Threats/Attacks

1472 (U) Device authentication mechanisms have many potential points of vulnerability. The protocol
1473 used to relay authentication across the network may be a point of attack. Dr. Arbaugh from the
1474 University of Maryland has already found several weaknesses in the 802.1x protocol. These
1475 vulnerabilities allow 802.1x to be attacked over the network. These attacks may allow an attacker
1476 to either hijack a session from an authenticated device or prevent a legitimate device from using
1477 the network.

1478 (U) Furthermore, device authentication may be relying on the physical security of the device
1479 itself. This security may come in the form of guards, guns, and dogs (standard physical security)
1480 or may be the result of the use of tamperproof/tamper evident devices such as a smart card. The
1481 guards, guns, and dogs model of physical security can be overcome by physical force.
1482 Tamperproof/tamper evident protections might be overcome by sophisticated technical attacks.
1483 Ross Anderson has published many papers on the topics of subverting tamper resistant/ proof
1484 devices.

1485 (U) However the device authentication mechanism is subverted, the end result is generally the
1486 same; lack of trust in the actual identity of the end device. When designing or deploying device
1487 authentication systems, great care must be exercised to determine the real security limitations of
1488 the protocols and products involved.

1489 2.1.3.3.3 (U) Maturity

1490 (U) Device authentication is an emerging technology. Until recently, there has been little
1491 perceived value in authenticating a device. Enterprises have been more worried about the identity
1492 of the user and have not focused their attention on the device itself. However, as devices become
1493 more mobile and disposable, device authentication is rapidly gaining visibility.

1494 (U) Unfortunately, few standards exist and even fewer products. This area of device
1495 authentication still requires a great deal of research and standards development before
1496 widespread market adoption will occur.

1497 (U) In summary, the Technology Readiness Level of device authentication can be viewed as
1498 Emerging (TRL 4 – 6).

1499 **2.1.3.3.4 (U) Standards**1500 **2.1.3.3.4.1 (U) 802.1x**

1501 (U) The Institute of Electrical and Electronics Engineers (IEEE) approved the standard 802.1x on
1502 June 14, 2001. This standard is based on the physical characteristics and identification of the
1503 device, port, or wireless station that is requesting the connection. The standard provides a
1504 mechanism for restricting access to a local area network (LAN) or a virtual local area network
1505 (VLAN). Generally, it is described as providing port-based access control.

1506 (U) The 802.1x authentication architecture consists of a supplicant—a user or entity representing
1507 the endpoint requesting a network connection; an authenticator—a network device or entity that
1508 is facilitating the authentication of the supplicant; and an authentication server or service—
1509 responsible for validating the supplicant’s credentials and determining whether to authorize the
1510 authenticator to grant access to the requested services.

1511 (U) 802.1x specifies how to carry link-level authentication information using Extensible
1512 Authentication Protocol (EAP). (See the next section.) While 802.1x does not require the use of a
1513 separate authentication service, it is often deployed in combination with a RADIUS server.

1514 **2.1.3.3.4.2 (U) EAP**

1515 (U) EAP, or Internet Engineering Task Force (IETF) RFC 2284, is an authentication framework
1516 that defines a way to encapsulate different authentication methods. EAP can be used in
1517 combination with point-to-point protocol (PPP) (IETF RFC 1661) or IEEE 802.1x. A recent
1518 Internet draft updates the original EAP specification.

1519 (U) A range of methods have emerged that build on EAP, including:

- 1520 • (U) EAP-Transport Layer Security (TLS), for encrypted communication between
1521 endpoints identified by public key infrastructure (PKI) certificates
- 1522 • (U) EAP-message digest 5 (MD5), for password authentication using a challenge-
1523 response approach
- 1524 • (U) EAP-Generic Token Card (GTC), for use with one-time password tokens
- 1525 • (U) EAP-Microsoft Challenge Handshake Authentication Protocol (MS-CHAPv2)

1526 (U) Cisco, Microsoft, and RSA collaborated in proposing Protected EAP (PEAP) to the IETF.
1527 PEAP has security improvements that extend Cisco’s Lightweight EAP (LEAP). LEAP uses a
1528 stronger password-hashing authentication approach than EAP-MD5, but is also susceptible to
1529 offline dictionary attacks against the password. PEAP is supported by Microsoft, Cisco, Funk
1530 Software, and Meetinghouse Communications, but is not recognized as an industry-wide
1531 standard. Typically, PEAP is used in combination with TLS for secure communication between
1532 endpoints that are authenticated using a method other than PKI.

1533 **2.1.3.3.4.3 (U) RADIUS**

1534 (U) RADIUS, most recently specified by IETF RFC 2865, was originally designed as a protocol
1535 mechanism for authenticating remote users. It is still typically used today to authenticate remote
1536 users connecting to a dial-in modem pool or an Internet-accessible, virtual private network
1537 (VPN), gateway device.

1538 (U) The typical architecture for RADIUS involves the VPN gateway or access server acting as
1539 the client, requesting authentication of a user connection; and a RADIUS server, performing the
1540 authentication and passing back appropriate configuration information to the requesting service.
1541 In addition, RADIUS servers can act as proxies for other RADIUS servers or authentication
1542 services. This is often required when users are roaming between service providers or interfacing
1543 between a service provider and an internal network's identification management infrastructure.

1544 (U) While RADIUS is independent of 802.1x, many network access devices are expected to
1545 implement both the 802.1x authenticator role and the RADIUS client role. However, 802.1x is
1546 unable to support the challenge-response mechanisms of RADIUS. Where a port ID is not
1547 available, such as in wireless situations, an association ID will be used.

1548 (U) The IETF informational RFC 3580 defines specific mappings and special considerations
1549 when using both 802.1x and RADIUS. In particular, it defines how to authorize access to a
1550 VLAN by leveraging the tunnel attributes of RADIUS. It also discusses specific known
1551 vulnerabilities with RADIUS and EAP and provides approaches to mitigate them.

1552 (U) IETF informational RFC 3579 specifies how a RADIUS client, or a network access server,
1553 encapsulates EAP packets to forward to the RADIUS server, where method-specific code can
1554 interpret and process the requests. This characteristic enables the network access server to be
1555 neutral as to which authentication method is being used and to be unaffected by the introduction
1556 of new authentication methods.

1557 **2.1.3.3.4.4 (U) PANA**

1558 (U) A more recent standards initiative is underway in the IETF work on a Protocol for carrying
1559 Authentication for Network Access (PANA). This work is still in a draft status, with additional
1560 deliverables planned for 2004 to define the interactions between PANA and 802.1x and to
1561 specify a Management Information Base (MIB) for the protocol.

1562 (U) Goals for the PANA effort include support for roaming devices, dynamic choice of service
1563 providers, and multiple authentication methods—all based on IP protocols. PANA is designed to
1564 work with EAP as a network-layer transport, carrying EAP payloads independently from the
1565 choice of link-layer protocol and avoiding potential roundtrip delays during connection
1566 establishment. Note, however, that the primary focus of this effort is to authenticate devices at
1567 Layer 3 or above before granting use of network services. A typical usage scenario involves a
1568 client system authenticating to a server to gain network access.

1569 (U) While mechanisms such as 802.1x and PPP already support specific link-layer support for
1570 EAP, other application-layer authentication approaches are considered to be ad hoc and
1571 vulnerable.

1572 (U) The work on PANA is still at an early stage and is being driven mostly by vendors,
1573 providing wireless network services, and mobile clients.

1574 **2.1.3.3.4.5 (U) Platform-Based Key Storage**

1575 (U) Hardware key storage is becoming built directly into personal computing devices. The
1576 Trusted Computing Group (TCG) and Next Generation Secure Computing Base (NGSCB) allow
1577 PKI keys and certificates to be stored on chips, which are manufactured into PC and PDA
1578 motherboards. In essence, the personal computing device contains a *built-in* smart card.

1579 (U) Although only a small number of vendors (e.g., IBM and HP) offer such products today
1580 TCG and NGSCB will play important roles in digital rights management and platform security in
1581 the next few years.

1582 **2.1.3.3.4.6 (U) XML and PKI [XKMS]**

1583 (U) As mentioned previously, the appeal of XML has reached PKI in the form of XKMS, a
1584 lighter-weight approach for clients and servers to deal with some of the complexities of
1585 traditional PKI processing, such as certificate path-checking and validation. While XKMS
1586 capability is being introduced into newer versions of PKI products, it has not yet had a major
1587 impact on the industry.

1588 (U) The World Wide Web Consortium (W3C) has published requirements for Version 2 of
1589 XKMS, which intends to improve the XKMS interactions with Simple Object Access Protocol
1590 (SOAP), XML Schema, and Web Services Description Language (WSDL).

1591 (U) XML Signature and XML Encryption standards have been formalized by the W3C and
1592 promise to be a prevalent part of future application development. The ability to encrypt and sign
1593 individual components of XML documents will require robust key management capabilities, a
1594 role potentially filled by PKI.

1595 (U) The Organization for the Advancement of Structured Information Standards (OASIS) has
1596 initiated a standards process for the XML-based Digital Signature Services (DSS). To date, a
1597 draft exists only for requirements and use cases, but DSS intends to provide an overarching set of
1598 XML techniques for the processing of digital signatures, including verification, time stamping,
1599 and signature creation.

1600 (U) Although ITU X.509 and the IETF PKIX group use ASN.1 as the basis of encoding for PKI
1601 certificates, there is interest in creating a general-purpose standard for XML certificate encoding.
1602 Discussions in the IETF and W3C have resulted in some initial drafts, but nothing has emerged
1603 as a clear standards candidate at this point. Due to the concerns about ASN.1 development and
1604 processing complexity, however, it is likely that continued effort in this area will result in the
1605 creation of a standards-based XML digital certificate format.

1606 **2.1.3.3.4.7 (U) IPsec VPNs**

1607 (U) Two headers form the basis of IPsec: the Authentication Header (AH) protocol and the
1608 Encapsulating Security Payload (ESP) protocol. AH, as the name implies, is used for
1609 authenticating packets from a host or network device. The ESP header can be used for both
1610 authentication and encryption.

1611 (U) Each of these protocols can operate in one of two modes: the transport mode or the tunnel
1612 mode. In transport mode, the protocol operates primarily on the payload of the original datagram.
1613 In tunnel mode, the protocol encapsulates the original datagram in a new datagram, creating a
1614 new IP header and treating the original datagram as the data payload.

1615 (U) The design of the AH and ESP headers is modular, which allows different cryptographic
1616 algorithms to be used as needed. As new algorithms are developed, such as elliptic curve
1617 algorithms and the Advanced Encryption Standard (AES), the parameters for their use can be
1618 standardized within IPsec's architecture and then used in conjunction with AH or ESP.

1619 (U) Although the AH and ESP protocols do not specify a particular automated encryption key-
1620 management system, IPsec implementations are designed to support both preshared keys and the
1621 automated key management system called Internet Key Exchange (IKE), which is defined in
1622 IEEE RFC 2401.

1623 **2.1.3.3.4.8 (U) SSL VPNs**

1624 (U) Using SSL version 3.0 to implement secure network connections is different than using
1625 IPsec, because connections focus on individual users and sessions rather than on multiplexed
1626 communications between sites. Thus, SSL-secured networks are similar to remote access VPNs,
1627 although most implementations of SSL-secured networks connect a user to a server (or server
1628 farm) and not to all the resources at a site.

1629 (U) One of the most appealing features of using SSL for a secure network is the deployment
1630 simplicity. The minimum requirements for an SSL-secured network are a Web server with an
1631 appropriate digital certificate and a Web browser on each user's computer. Note that this setup is
1632 mostly used for Web-based access. File Transfer Protocol (FTP), Simple Mail Transfer Protocol
1633 (SMTP), and Network News Transport Protocol (NNTP) can use SSL if the appropriate SSL-
1634 enabled versions of those products are used.

1635 (U) As commonly deployed, only the servers require digital certificates to initiate SSL sessions.
1636 This considerably reduces the number of certificates to be managed and distributed. That may
1637 suit some enterprises. However, organizations looking to authenticate external users, such as for
1638 an extranet, must employ some form of client authentication. This adds the requirement for a
1639 PKI system if authentication is to be performed within the SSL protocol.

1640 **2.1.3.3.5 (U) Costs/Limitations**

1641 (U) Device authentication technologies and protocols, while existing in some form today, are
1642 still considered emerging technologies. This can be seen in the Standards section, while noting
1643 the number of Working Groups (IETF and others) that are still working towards enhancing the
1644 authentication and security of these standards.

1645 (U) From a pragmatic GIG enterprise services viewpoint, the type technology selected depends
1646 on the particular situation and its mission needs of the authentication strength. For example, for
1647 situations that do not require the strictest authentication and secure levels, combinations of Wi-Fi
1648 Protected Access (WPA) on a wireless local area network (WLAN) using RADIUS and LDAP
1649 servers should meet most needs.

1650 2.1.3.3.6 (U) Dependencies

1651 (U) Microsoft provides built-in support for 802.1x in Windows XP. Windows 2000 users
1652 running Service Pack 3 can download the Microsoft 802.1x Authentication Client for Windows
1653 2000. Microsoft also supplies versions of this client software to Windows 98 and NT users with
1654 a Premier support agreement.

1655 (U) Apple has built-in support for 802.1x in Mac OS X (v10.3), which can be configured to
1656 access either an AirPort wireless connection or a secure Ethernet port. Mac OS X v10.3 also
1657 supports WPA for WLANs without the need for 802.1x or a RADIUS server, which is ideal for
1658 home users without a RADIUS server.

1659 (U) Linux systems require client software that performs the 802.1x supplicant function. This
1660 software is available from the Open Source Implementation of 802.1x site used in combination
1661 with OpenSSL (Secure Sockets Layer) and FreeRADIUS.

1662 (U) In addition, developer kits and 802.1x drivers for various operating environments are
1663 available from software vendors, such as Meetinghouse Data Communications with its AEGIS
1664 product line. The AEGIS client is available for Windows 98, ME, NT, 2000, and XP; Pocket PC
1665 and Palm products; Mac OS X; and Linux. Funk Software offers its Odyssey client for Windows
1666 98, ME, NT, 200, and XP; Pocket PC; and Windows Mobile.

1667 (U) A growing class of products that assess the status of client systems for conformance to
1668 security policies are embracing 802.1x authentication to integrate with network switching
1669 systems. Access to the network is only granted once policy conformance has been established.
1670 Both Zone Labs' Integrity 5.0 and Sygate's Secure Enterprise support this feature. Zone Labs
1671 (acquired by Checkpoint Software) certifies its 802.1x feature to work with products from
1672 Aruba, Cisco, Enterasys, Funk Software, and Microsoft. Sygate announced support for
1673 interoperability with products from Cisco, Enterasys, Extreme, HP, and Nortel. One of the
1674 features of Sygate's solution is to quarantine any client systems, which are not running policy-
1675 checking agent software, to a guest VLAN.

1676 (U) Other third-party software products inherit support for 802.1x simply by working with
1677 existing 802.1x-aware client software, such as the support built in to Windows XP. For example,
1678 RSA provides support for SecurID authentication to WLANs through its Advanced Computing
1679 Environment (ACE)/Agent for Windows and the Windows XP wireless LAN client.

1680 (U) Fiberlink, GRIC, and iPass are implementing similar capabilities for their VPN clients.
1681 These companies provide remote access management and VPN capabilities. Their clients check
1682 the mobile device infrastructure to make sure—before allowing connection—that the firewall is
1683 running, the virus scanner is running and up to date, and the VPN is active.

1684 **2.1.3.3.7 (U) Alternatives**1685 **2.1.3.3.7.1 (U) MAC/IP address**

1686 (U) An alternative is to use the older simpler methods of device identification such as the media
1687 access control (MAC) address or IP address of the device the user is using at the time. Enterasys'
1688 User Personalized Network (UPN) is such an example. Once identity is established, the switch
1689 can determine whether to grant access to devices associated with a restricted VLAN. One of the
1690 main strengths of the UPN is its ability to provision network services and applications based on
1691 user identity. The Enterasys solution relies on existing enterprise investments in directories—
1692 such as Microsoft's Active Directory or Novell's eDirectory—to authenticate user identity and
1693 establish an association with the user's location.

1694 (U) Within the scope of device authentication, there exist a number of alternatives and
1695 combinations. Most of these are related to specific vendors and platforms. These are described
1696 below.

- 1697 • (U) Alcatel implements an approach to Layer 2 authentication within its OmniSwitch
1698 product line. Alcatel's authenticated VLAN (AVLAN) feature does not rely on operating
1699 system support for EAP and 802.1x, but requires an Alcatel-supplied client application:
1700 AV-Client for Windows 9x, NT, 2000, and XP. This client combines the Windows login
1701 with a network login, so a user enters an identity and credential only once. A successful
1702 authentication connects the user to the VLAN and its resources.
- 1703 • (U) Cisco has a framework for identity-based networking services that is supported
1704 across several product lines, including Catalyst switches (6500, 4500, 3550, and 2950),
1705 Aironet wireless access points, and Cisco's Secure Access Control Server v3.2 (ACS).
1706 The various network switch products implement 802.1x. They perform the role of an
1707 authenticator or intermediary between the supplicant at the client and the RADIUS
1708 authentication service. Cisco's RADIUS server product is ACS.
- 1709 • (U) Cisco extends 802.1x to enable dynamic assignment of VLANs to ports (based on
1710 identity), guest VLAN support, mapping of access control lists (ACLs) to a port based on
1711 the user's 802.1x identity, and synchronization of port security status in case of failover.
1712 Also, Cisco IP phones can be automatically mapped to a voice VLAN when detected.
1713 Computers connected to IP phones will need to authenticate to get access to the network.
- 1714 • (U) Cisco also announced its Network Admission Control (NAC) program, a
1715 collaboration with industry partners focused on limiting damage from security threats
1716 originating at client systems that have been compromised by a virus or worm. In its initial
1717 phase, NAC enables Cisco routers to enforce access privileges when an endpoint device
1718 attempts to connect to a network. This decision can be based on information about the
1719 endpoint device, such as its current antivirus state and operating system patch level. NAC
1720 allows noncompliant devices to be denied access, placed in a quarantined area, or given
1721 restricted access to computing resources.
- 1722 • (U) Nortel has supported 802.1x in its BayStack switches since 2001. Recent extensions
1723 to its BayStack operating system Switching Software (BoSS) v3.0 for BayStack 420 and

1724 425 switches, improve its support for EAP and 802.1x. Access to network services
1725 requires a login to a RADIUS authentication server. Also, its Wireless LAN 2200 series
1726 includes support for Virtual Port-based Authentication (VPA) based on EAP and 802.1x
1727 back to a RADIUS server (both in its WLAN Access Points and in the optional WLAN
1728 Security Switch 2250 unit). Other products, such as Passport 8600, support VLANs for a
1729 variety of network separation requirements. Nortel partners with Sygate to leverage
1730 802.1x to quarantine systems that are out of compliance with local security configuration
1731 policies.

1732 **2.1.3.3.7.2 (U) VPN-based Authentication**

1733 (U) IPsec-based VPN: Due to its original development for site-to-site VPNs, IPsec focuses on
1734 machine authentication rather than user authentication, and this has caused problems in creating
1735 interoperable dial-in clients. To improve the usability and interoperability of IPsec-based VPN
1736 dial-in clients, the IETF's IPsec Remote Access (IPSRA) working group has been trying to settle
1737 on a single protocol that it will propose as a standard to the IETF. After almost two years' work
1738 on four (or more) different proposals, the working group has settled on the Pre-IKE Credential
1739 Provisioning Protocol, or PIC, which is slowly making its way into commercial products.

1740 (U) SSL-based VPN: Though the SSL standard does not support client authentication methods
1741 other than digital certificates, it is possible to use other authentication methods in conjunction
1742 with SSL. The simplest approach is username and password, but it is also possible to use
1743 stronger authentication methods, such as security tokens or smart cards.

1744 **2.1.3.3.8 (U) References**

1745 (U) An Initial Security Analysis of the IEEE 802.1x Protocol -

1746 (U) <http://www.cs.umd.edu/~waa/1x.pdf>.

1747 (U) Ross Anderson's Home Page - <http://www.cl.cam.ac.uk/users/rja14/#Reliability>.

1748 **2.1.3.4 (U) Authentication Protocols**1749 **2.1.3.4.1 (U) Technical Detail**

1750 (U) There are two major traditional authentication protocol techniques – Symmetric Key
1751 Authentication and Public Key Authentication.

1752 (U) Symmetric Key Authentication:

1753 (U) In symmetric key authentication, the shared secret key is used at the client to create an OTP
1754 that is then transmitted to the server. The same process is done at the server, and if a match
1755 exists, the user is authenticated.

1756 (U) Many commercial schemes use public-domain hash functions based upon ANSI X9.9, which
1757 relies on Data Encryption Standard Message Authentication Code (DES MAC), which is a
1758 cipher-block, chained checksum. Some vendors use proprietary algorithms, such as RSA
1759 Security. It should be noted that X9.9 (based on 56-bit single DES) was withdrawn by ANSI in
1760 1999 in favor of the stronger Triple DES algorithm.

1761 (U) Another often used public domain hash function is the SHA-1 or Secure Hash Algorithm,
1762 which comes from NIST in the federal government. For greater security, some tokens actually
1763 recalculate a new-shared secret key after each authentication process, which requires that the
1764 server do likewise in order to keep in step.

1765 (U) A common symmetric key authentication scheme is the Kerberos protocol. Kerberos is a
1766 network authentication protocol. Kerberos is designed to provide strong authentication for
1767 client/server applications by using secret-key cryptography. This is accomplished without relying
1768 on authentication by the host operating system, without basing trust on host addresses, without
1769 requiring physical security of all the hosts on the network, and under the assumption that packets
1770 traveling along the network can be read, modified, and inserted at will. Kerberos performs
1771 authentication under these conditions as a trusted third-party authentication service by using
1772 conventional cryptography, i.e., shared secret key. The authentication process proceeds as
1773 follows:

- 1774 1. (U) A client sends a request to the authentication server (AS) requesting "credentials" for
1775 a given server.
- 1776 2. (U) The AS responds with these credentials, encrypted in the client's key. The credentials
1777 consist of 1) a "ticket" for the server and 2) a temporary encryption key (often called a
1778 "session key").
- 1779 3. (U) The client transmits the ticket (which contains the client's identity and a copy of the
1780 session key, all encrypted in the server's key) to the server.
- 1781 4. (U) The session key (now shared by the client and server) is used to authenticate the
1782 client, and may optionally be used to authenticate the server. It may also be used to
1783 encrypt further communication between the two parties or to exchange a separate sub-
1784 session key to be used to encrypt further communication.

1785 (U) Another symmetric key authentication protocol is CHAP or the Challenge Handshake
1786 Authentication Protocol (defined in [RFC 1994](#)) verifies the identity of the peer using a three-way
1787 handshake. The following general steps are performed in CHAP.

- 1788 1. (U) After the link esTablishment phase is complete, the authenticator sends a challenge
1789 message to the peer.
- 1790 2. (U) The peer responds with a value calculated using a one-way hash function (Message
1791 Digest 5 [MD5]).
- 1792 3. (U) The authenticator checks the response against its own calculation of the expected
1793 hash value. If the values match, the authentication is successful. Otherwise, the
1794 connection is terminated.

1795 (U) Public Key Authentication:

1796 (U) Unlike symmetric key authentication which relies on a single shared secret key, public key
1797 authentication employs a related pair of keys: one public (known to the server) and one private
1798 (known only to the client token and computationally unlikely to be derived from its public key
1799 counterpart). In the authentication process, the token employs its private key in a cryptographic
1800 function related to that which is executed by the server with the public key. The token function is
1801 typically implemented as a software token on the local client host, usually in a challenge-
1802 response mode.

1803 (U) Effective management of public and private key pairs across a large population of users
1804 requires a PKI. A public key certificate (or digital certificate) binds a user identity with its
1805 associated public key, and a trusted central agent or certification authority (CA) serves to verify
1806 the validity of issued certificates.

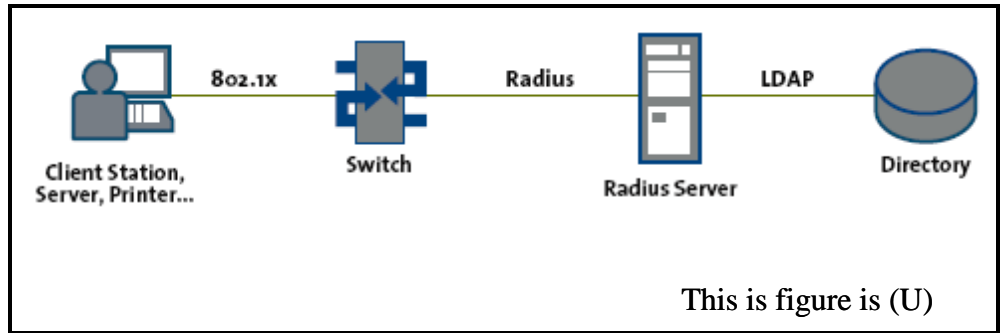
1807 (U) In a challenge-response authentication process, the server would send a random challenge to
1808 the client. The client then uses its private key to digitally sign the challenge, which is then
1809 returned as a response to the server along with its public key certificate (which could
1810 alternatively be retrieved by the server from the CA). If the certificate is shown to be valid, the
1811 server verifies the digital signature through application of the client's public key.

1812 (U) Currently deployed examples of public key certificate-based software token authentication
1813 include Microsoft's Windows 2000 server operating system (using PKINIT or Public Key
1814 Initialization Authentication) and commercial versions of Secure Shell (SSH).

1815 (U) Authentication mechanisms often depend upon the environments in which they are to
1816 operate, along with other considerations. The following sections describe various aspects of
1817 emerging authentication technology.

1818 **2.1.3.4.1.1 (U) 802.1x for network applications**

1819 (U) For network access applications, 802.1x can serve as the authentication protocol framework.
 1820 This is true both for wired and wireless networks. The authenticator is the access point for
 1821 wireless networks; it is the layer-two switch for wired networks. Figure 2.1-5 shows a network
 1822 authentication framework. A natural candidate is 802.1x because it already defines EAP methods
 1823 for each of the proposed base authentication methods (e.g., EAP-SIM for SIM-based
 1824 authentication, EAP-TLS for PKI-based authentication, and EAP-PEAP for OTP-based
 1825 authentication).

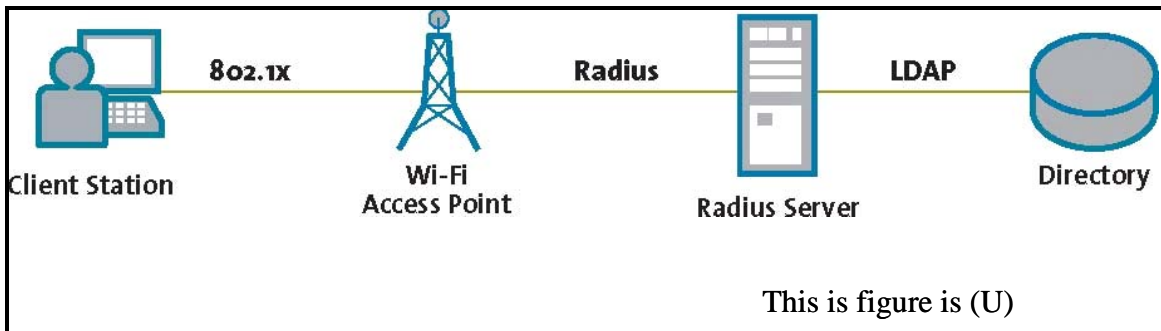


1826

1827 **Figure 2.1-5: (U) Network Authentication Framework**

1828 **2.1.3.4.1.2 (U) 802.1x for device authentication**

1829 (U) The 802.1x framework is crucial to promote a consistent deployment profile for device
 1830 authentication across manufacturers and OS vendors. Embedded 802.1x clients can be deployed
 1831 to enable these devices (e.g., VoIP phones, access points, switches, servers) to transparently
 1832 authenticate to the network, before being handed an IP address and being granted access to the
 1833 network. Figure 2.1-6 shows this.



1834

1835 **Figure 2.1-6: (U) Device Authentication Framework**

1836 2.1.3.4.1.3 (U) Manufacturing-time device credentials

1837 (U) Device certificates can be combined with emerging secure computing technologies such as
1838 the Trusted Platform Module (TPM) and the 802.1x authentication protocol framework. This
1839 convergence will foster a common technology stack and deployment profile to allow device
1840 manufacturers to enable turnkey-strong device authentication solutions. In fact, using the
1841 established profile, manufacturers and OEMs will be able to rapidly collaborate to embed the
1842 necessary hardware credentials and client software at manufacturing time.

1843 2.1.3.4.1.4 (U) Web service protocol for business-application integration

1844 (U) Universal strong authentication must address the protocol dichotomy between network
1845 access applications (e.g., dial-up, VPN, Wi-Fi) and business applications, such as Web or
1846 enterprise portals, Web applications, ERP systems, and Web services. The 802.1x framework is
1847 particularly well suited to the former, but not to the latter. A Web service interface is better
1848 adapted to today's business applications.

1849 (U) Because the authentication protocols constitute the primary mechanism for integration into
1850 applications, open authentication requires a palette of protocols that can support both types of
1851 applications. This requirement leads to the definition of a Web service API alongside the 802.1x
1852 EAP methods already covered. The Simple Object Access Protocol (SOAP) API can leverage the
1853 WS-Security specification as the primary mechanism for encoding the base security tokens
1854 (OTP, X509 certificate). It can also define a challenge-response mechanism for SIM-based
1855 authentication.

1856 2.1.3.4.1.5 (U) Application connectors and authentication clients

1857 (U) The main motivation for standardizing an authentication protocol and promoting the
1858 development of authentication clients is to foster the creation of application *connectors*.
1859 Application connectors, or agents, are the client libraries of strong authentication. They must be
1860 portable across major operating systems and offer APIs across popular languages. Such
1861 flexibility would make it easier for application developers to integrate strong authentication
1862 within custom applications (e.g., link, compile, and run). This is mainly true for the EAP
1863 protocols—EAP-SIM, EAP-TLS, EAP-PEAP—because the Web service can immediately
1864 leverage the Web services stack that exists in all major development platforms.

1865 2.1.3.4.1.6 (U) Credential Provisioning and Validation

1866 (U) Since universal strong authentication is a key objective, the blueprint needs a method to
1867 harmonize credential issuance and other life cycle management functions across all types of
1868 secrets, symmetric keys or RSA key pairs. The SIM and OTP secrets become subordinate to an
1869 RSA key pair (a device certificate key pair). The shared secrets are encrypted and embedded as
1870 attributes within the certificate. The certificate acts as a private store for the shared secrets, and
1871 the security device acts as a secure hardware vault for the *root* credential.

1872 (U) This approach allows manufacturers and customers to leverage the breadth of secret
1873 management capabilities and security practices (e.g., key escrow, secure roaming, and directory
1874 services) from existing PKI platforms. The method applies both to secure device personalization
1875 (shared secret and device certificates embedded at manufacturing time) and secure provisioning
1876 of user credentials. This unified credential life cycle management framework will leverage
1877 existing public key cryptography standards and modern protocols such as XML Key
1878 Management Specification (XKMS).

1879 (U) Validation profiles will be defined by the choice of authentication protocols, as described
1880 earlier. In addition, validation services will be able to validate X.509 certificates using certificate
1881 revocation lists (CRLs) and industry standards such as Online Certificate Status Protocol (OCSP)
1882 or XKMS.

1883 (U) Validation servers in a strong authentication environment have the same characteristics as
1884 RADIUS servers. This is a conscious choice, as RADIUS servers are already a key component of
1885 an ISP or enterprise network infrastructure. Furthermore, high-quality RADIUS servers are
1886 widely available from vendors and open-source projects. The complexity and cost overhead for
1887 deploying strong authentication can be reduced by leveraging the large, existing, installed base
1888 of RADIUS servers.

1889 (U) For applications that require a Web service interface, the validation server will be required to
1890 implement the SOAP validation protocol discussed earlier. In the network world, the strong
1891 authentication validation server is congruent to a RADIUS server; while in a service-oriented
1892 architecture, the validation server is an instance of a Web service. Because credential validation
1893 is highly complementary to credential mapping and exchange, it makes sense to consolidate Web
1894 services with the architectural concept of Security Token Service (STS), as defined by Web
1895 Services Trust Language (WS-Trust).

1896 (U) An important architecture goal for universal authentication is to enforce the separation
1897 between validation and identity stores. All identities (user or device identities, as well as device-
1898 to-user bindings) should be maintained outside the validation server. This separation is important
1899 from an integration and cost-control standpoint. It promotes a distributed architecture that favors
1900 the reuse of an enterprise's existing infrastructure (e.g., corporate directories). In such an
1901 architecture, the validation server is a minimal front end.

1902 **2.1.3.4.2 (U) Usage Considerations**

1903 (U) In many cases, the specific application dictates the authentication protocol. For example, in
1904 a Web application, TLS will often be the primary protocol. In the VPN case, IPsec IKE is the
1905 standard, and for wireless Wi-Fi (802.1x), Extensible Authentication Protocol (EAP) methods
1906 such as EAP-TLS or EAP-PEAP are the norm.

1907 (U/FOUO) A major disadvantage of symmetric key authentication is that it does not scale well to
1908 large and global user populations, due to the logistical difficulties of distributing the shared
1909 secret keys. This disadvantage affects the use of the following protocols:

- 1910 • (U) Kerberos

- 1911 • (U) CHAP
- 1912 • (U) 802.11 wireless
- 1913 • (U) EAP-PEAP for OTP (one-time-password) authentication

1914 **2.1.3.4.2.1 (U) Advantages**

1915 (U/FOUO) A distinct advantage of public-key authentication is that it easily scales to very large
1916 networks (such as the GIG), whereas symmetric key or shared-secret authentication is generally
1917 limited to specific communities of interest in which the key management process will not be
1918 unduly burdensome.

1919 **2.1.3.4.2.2 (U) Risks/Threats/Attacks**

1920 (U/FOUO) A common risk/threat/attack that has to be anticipated and dealt with appropriately
1921 by any proposed authentication scheme is the classic man in the middle (MITM) attack in which
1922 a malicious adversary will intercept the communications between a client and its authentication
1923 server, and then modify the message protocol contents so as to defeat, hijack, or otherwise
1924 maliciously alter the proper authentication protocol. It is essential that all critical authentication
1925 messaging be suitably encrypted so as to prevent this.

1926 **2.1.3.4.3 (U) Maturity**

1927 (U) Due to the strong desire across both the government and industry (particularly the financial
1928 industry) for secure authentication of parties conducting electronic communications and
1929 transactions, authentication protocols have developed over the years into a fairly mature state.
1930 Thus, the Technology Readiness Level of authentication protocols would be grouped into the
1931 Mature category (TRL 7 – 9).

1932 **2.1.3.4.4 (U) Standards**

1933 (U) There are a variety of formalized international and American standards covering the
1934 technology of authentication protocols.

1935 **2.1.3.4.4.1 (U) International Standards:**

1936 (U) The international standards bodies that are responsible for developing authentication
1937 protocols include:

- 1938 • (U) IETF Internet Engineering Task Force (<http://www.ietf.org>)
- 1939 • (U) ISO International Organization for Standardization (<http://www.iso.ch>)
- 1940 • (U) ITU-T International Telecommunication Union Telecommunication Standardization
1941 Sector (<http://www.itu.int/ITU-T>)
- 1942 • (U) IEEE Institute of Electrical and Electronics Engineers
1943 (<http://grouper.ieee.org/groups/1363/>)
- 1944 • (U) Industrial consortiums such as OASIS (Organization for the Advancement of
1945 Structured Information Standards, <http://www.oasis-open.org/committees/wss>), which develops

1946 security standards for web services

1947 (U) IETF standards that are relevant to authentication tokens include Internet Drafts from the
1948 Secure Shell working group, and RFCs 2289 and 1760 that describe the S/Key One-Time-
1949 Password System.

1950 (U) Relevant ISO standards include ISO 8731 (algorithms for banking message authentication),
1951 ISO/IEC 9797 (MACs via block cipher and hash function), ISO/IEC 9798 (entity authentication
1952 by symmetric, digital signature, and cryptographic check), and ISO/IEC 19092.

1953 (U) Relevant ITU-T standards include those describing directory certificates for authentication
1954 such as X.509 (issued 08/97, authentication framework), and X.509 (issued 03/00, public key
1955 and attribute certificate frameworks).

1956 (U) IEEE standards include P1363 (specifications for public key cryptography).

1957 (U) OASIS standards include WSS (Web Services Security) Version 1.0 (April 2004). WSS
1958 handles confidentiality/integrity for SOAP (Simple Object Access Protocol) messages, providing
1959 a mechanism for associating security tokens with message content. WSS is extensible and
1960 supports multiple security token formats. It builds upon existing security technologies such as
1961 Extensible Markup Language (XML) Digital Signature, XML Encryption, and X.509
1962 Certificates to deliver a standard for securing Web Services message exchanges. Providing a
1963 framework where authentication and authorization take place, WSS lets users apply existing
1964 security technology in a Web Services environment.

1965 (U) Founded in 1993, OASIS has members in 100 countries and 600+ organizations (including
1966 Entrust, HP, Hitachi, IBM, Microsoft, Nokia, RSA Security, Sun Microsystems, and Verisign).

1967 **2.1.3.4.4.2 (U) American Standards:**

1968 (U) Organizations in the United States that are responsible for developing and promulgating
1969 authentication protocol standards include ANSI American National Standards Institute
1970 (<http://www.ansi.org>), and NIST National Institute of Standards and Technology
1971 (<http://www.itl.nist.gov/fipspubs>, repository of the Federal Information Processing Standards or
1972 FIPS).

1973 (U) Relevant ANSI standards include X9.9 (message authentication codes for symmetric token
1974 authentication, withdrawn in 1999 due to attacks demonstrated against single DES 56-bit key, in
1975 favor of double or triple DES), X9.30 (public key cryptography, digital signature algorithm
1976 DSA, secure hash algorithm SHA-1, DSA certificate management), X9.31 (reversible public key
1977 cryptography for digital signatures rDSA), X9.45 (management controls using digital signatures
1978 and attribute certificates), X9.52 (triple DES modes of operations), X9.63 (key agreement and
1979 transport using elliptic curve cryptography ECC), X9.71 (keyed hash for message
1980 authentication), and X9.72 (peer entity authentication using public keys).

1981 (U) Relevant NIST FIPS PUB standards include FIPS 180 (secure hash algorithm SHA-1), FIPS
1982 186-2 (digital signature standard DSS, same as ANSI X9.30), FIPS 190 (guideline for use of
1983 advanced authentication technology alternatives), FIPS 196 (entity authentication using public
1984 key cryptography, same as ANSI X9.72), and FIPS 197 (advanced encryption standard AES). An
1985 informative new NIST draft document on authentication mechanisms is Special Publication 800-
1986 63 (*Recommendation for Electronic Authentication*, January 2004, which can be found at
1987 <http://csrc.nist.gov/publications/drafts/draft-sp800-63.pdf>).

1988 (U) The purpose of this section is not to explain all of the various algorithms used by
1989 authentication tokens but to note that tokens—hardware or software—can use a variety of
1990 cryptographic algorithms to produce the desired OTP (algorithms such as DES, Triple-DES,
1991 DSA, SHA, ECC, and the new AES Advanced Encryption Standard). However, as algorithms
1992 are improved and attacks discovered against the weaker algorithms, some standards are
1993 superseded or withdrawn.

1994 **2.1.3.4.5 (U) Cost/Limitations**

1995 (U) An authentication protocol that is based upon symmetric or secret key cryptography has in it
1996 a very costly and limiting characteristic in that the associated secret keys must be delivered a
1997 priori to all parties. This is a severe limitation in the context of the GIG.

1998 (U) Whereas both symmetric and public key authentication can be done at the application layer,
1999 only public key authentication can be done automatically at the transport layer.

2000 **2.1.3.4.6 (U) Dependencies**

2001 (U) One dependency of public key encryption-based authentication protocols is the existence of
2002 a well-developed and robust PKI.

2003 **2.1.3.4.7 (U) Alternatives**

2004 (U) The alternatives to use of an authentication protocol are few and undesirable. One
2005 alternative is simply to forgo authentication, but this is not thinkable in the context of the GIG.
2006 Another alternative would be within the context of a closed system where all
2007 communicating/participating parties are talking securely to each other over link-encrypted lines
2008 and are thus inherently trusted to each other.

2009 **2.1.3.4.8 (U) Complementary Techniques**

2010 (U) Certainly tokens (both hardware and software) are a complementary technology to that of
2011 authentication protocols. It is within the client-retained token that much of the authentication
2012 algorithm is either stored and/or executed in the field during a given authentication attempt.

2013 **2.1.3.4.9 (U) References**

2014 (U) RFC 1994, “PPP Challenge Handshake Authentication Protocol (CHAP)”,
2015 <http://www.ietf.org/rfc/rfc1994.txt> , by W. Simpson, 1996.

2016 (U) NIST Special Publication 800-63, “Recommendation for Electronic Authentication”,
2017 <http://csrc.nist.gov/publications/drafts/draft-sp800-63.pdf>, January 2004.

2018 **2.1.3.5 (U) Authentication Confidence**

2019 (U) Authentication confidence refers to developing a system that determines the probability that
 2020 a user or other device is who he/she/it claims to be. It takes into account such factors as:

- 2021 • (U) The authentication mechanism (e.g., static password, public-key cryptography,
 2022 software token, hardware token, biometrics)
- 2023 • (U) The authentication protocol used: e.g., a protocol that is known to be secure against
 2024 man-in-the-middle attacks or one that is based on strong cryptographic operations
- 2025 • (U) The location of the entity being authenticated: e.g., a secure office, CONUS or
 2026 OCONUS, a public kiosk or Internet cafe, a tactical battlefield
- 2027 • (U) Characteristics of the device used to authenticate: e.g., a COTS computer owned and
 2028 controlled by the US Government; a publicly-accessible COTS computer; a dedicated,
 2029 tamper-resistant device
- 2030 • (U) The communications path between the entity being authenticated, and the server
 2031 providing authentication and/or access decisions: e.g., a secure, U.S. Government-owned
 2032 or leased network; a wireless network on a battlefield; commercially-provided
 2033 telecommunications lines; a coalition partner's network

2034 (U//FOUO) The goal of authentication confidence is to quantify the risk that a user or entity
 2035 attempting to access the system is not the purported user or entity. This risk can then be provided
 2036 to an access control service to grant or restrict access to system resources.

2037 (U//FOUO) The simplest example of authentication confidence is a user logging into the system
 2038 over an insecure network, from a public kiosk, using a static password based authentication
 2039 system. For example, someone purporting to be Joe logs into the system and provides the correct
 2040 password. However, from tracing IP addresses and using known information, the authentication
 2041 server determines that Joe is coming in over a public Internet Service Provider's network from a
 2042 public kiosk in a coffee shop and is not using a strong authentication protocol. How confident is
 2043 the authentication server that this is really Joe, when there are numerous opportunities for the
 2044 password to have been compromised? It could have been acquired previously through a
 2045 dictionary attack or by someone finding a slip of paper with Joe's password. It could have been
 2046 captured on this use, via a keystroke logging function on the public terminal, or at some point
 2047 over the network. Thus, even though some entity has provided a valid user identifier and the
 2048 correct password, the system may still want to limit or even prevent access to resources, for fear
 2049 that the entity at the other end of the connection is not really Joe. This may be the case for future
 2050 login sessions as well, as Joe's password now is very likely to have been compromised upon this
 2051 use.

2052 (U//FOUO) Note that authentication confidence is related to but distinct from policy-based
2053 access control decisions. In the scenario described in the previous paragraph, the result of a weak
2054 level of confidence in Joe's authentication was that Joe was restricted from or prevented from
2055 accessing certain resources. This is because authentication confidence is one of a number of
2056 inputs to the access control mechanism. However, other inputs to that mechanism could have
2057 also resulted in access being restricted. For example, even if there was perfect confidence that
2058 Joe was really the user accessing the system, and that there was no chance that Joe's
2059 authentication data was compromised for future uses, Joe's access might still be restricted
2060 because of his location or communications path (e.g., sensitive or classified information would
2061 not be sent to a location with insufficient physical security).

2062 **2.1.3.5.1 (U) Technical Detail**

2063 (U) Authentication confidence at this time is a research area. While some work has been done,
2064 and the general requirement is understood, there are significant details to be worked out and
2065 major questions to be resolved. Among the issues to be addressed are:

- 2066 • (U) Authentication metrics: It is generally accepted that static passwords are weaker than
2067 one-time passwords, and that a hardware token with a PIN is generally better than a
2068 software token. However, there is no quantitative metric that compares different types of
2069 biometric authentication with each other or that compares biometric authentication with
2070 hardware token-based authentication or public-key cryptography-based authentication. In
2071 order for authentication confidence to have any meaning, there must be a way to measure
2072 and determine the relative (if not absolute) strength of each given authentication method.

- 2073 • (U) Reliable communication of user location: One of the factors normally considered to
2074 be part of authentication confidence is the location of the user, e.g., within a secure area
2075 or in public. In order for authentication confidence to be used, there must be a way for the
2076 authentication server to reliably know this information. The information must be
2077 conveyed to the server, and it must not be possible for an attacker to spoof this. For
2078 example, it must not be possible for a public terminal in a kiosk to convince the
2079 authentication server that it is in a secure location; and it must not be possible for a
2080 device that is on a battlefield in Southwest Asia to convince an authentication server that
2081 it is in a headquarters building in CONUS.

- 2082 • (U) Reliable communication of device characteristics: Another factor of authentication
2083 confidence is the characteristics of the device being used by the user (e.g., a public COTS
2084 computer system, a COTS computer system controlled by the Government organization,
2085 or a special-purpose device with strong tamper resistance and strong cryptography). The
2086 device must be capable of communicating this information to the authentication server,
2087 and it must not be capable of being spoofed. One of the initial research areas is
2088 determining precisely which set of characteristics is important in which situations.

- 2089 • (U) Corrections/modifications for error cases: For every type of authentication system
2090 used, there are two possible types of errors: false positives, in which the wrong entity is
2091 authenticated as being the correct one; and false negatives, in which the correct entity is
2092 rejected. Each authentication technique has different false positives and false negatives.
2093 For a password-based system, a false positive occurs when an attacker knows the correct

2094 password; a false negative occurs when the legitimate user fails to enter the correct
2095 password (because he has forgotten it or mistyped it). For a biometric-based system,
2096 false positives occur when an attacker's measurement is close enough to the legitimate
2097 value to allow authentication. For a false negative to occur the legitimate user's value is
2098 rejected as not matching the stored value. In traditional authentication systems, these
2099 differences can be taken into account by policy, but the bottom line is that a user is
2100 authenticated or not as a binary state. A user who is deemed to match gets access; one
2101 who is deemed not to match is rejected. There is no partial authentication or reflection of
2102 potential errors. One of the potential benefits of an authentication confidence system is
2103 that it allows for partial access, based on a partial match. That is, the authentication server
2104 could decide that a fingerprint is close enough to the correct value to allow some access,
2105 but there is enough doubt (i.e., through possibly smudged lenses, scraped-off
2106 fingerprints) that access to the most sensitive information and resources will be withheld.
2107 This results in allowing legitimate users some use of the system so that they are not
2108 completely shut out, while restricting the amount of damage that an attacker can cause.

2109 **2.1.3.5.2 (U) Maturity**

2110 (U) As this is a research area at the present time, there are no significant usage considerations to
2111 document. As the area matures, usage will be a major factor in the development and deployment
2112 of authentication confidence mechanisms and solutions.

2113 (U) At this point, authentication confidence is in its infancy, and thus is assigned to the lowest
2114 Technology Readiness Group: Early (TRL 1 – 3).

2115 **2.1.3.5.3 (U) Standards**

2116 (U) A major step necessary for acceptance of authentication confidence metrics will be standards
2117 for those metrics. Without standards, users and organizations will not be able to assign
2118 meaningful values and make appropriate decisions about allowing access. In particular,
2119 standards will need to address:

- 2120 • (U) Authentication metrics. In addition to standards for the individual authentication
2121 mechanisms (e.g., passwords, biometrics, and authentication tokens), standards will be
2122 needed to map the metrics to one another
- 2123 • (U) Error indications: Standards will be required for assessing “how close” a presented
2124 authenticator is to the “correct” one; e.g., a biometric value was deemed to be incorrect,
2125 but it was off by some small value; or a password presented was not the correct one, but
2126 it differed from the correct one by some characteristic which could easily be explained by
2127 a typing error or line noise.

2.1.3.6 (U) Single Sign-On

2128 (U) Single Sign-On (SSO) has traditionally been limited to cases covering the one-time sign-on
2129 process for access to all services of a single organization, whereas Global Sign-On has applied to
2130 multiple participating organizations that had reached an a priori collaborative agreement to avail
2131 users with a common sign-on process. In the GIG Vision SSO is expanded to enable a user to
2132 login or sign-on only once to a global authentication server thus allowing an entity to
2133 simultaneously access the GIG information and resources without any requirement for additional
2134 identification and authentication. With this definition, SSO and Global Sign-On become one and
2135 the same. Some communities view Global Sign-on as including the issues related to mobile
2136 users, while SSO does not. In this document fixed versus mobile issues are both treated under
2137 SSO.
2138

2139 (U) The goal of an ideal SSO system is to enable a user to login or sign-on only once to a global
2140 authentication server. This approach eliminates the need to enter different passwords to login to a
2141 workstation, to each service, database, etc. and replaces this with an automatic sign-on or re-
2142 authentication of an entity, making sign-on transparent. SSO must not sign an entity on with all
2143 of their privileges or escalate an entity's privileges without the entity's consent. This would be
2144 equivalent to signing on as a system administrator/super user to read personal email. SSO should
2145 also include a way to lower (or release) privileges once the activity that required increased
2146 privileges is complete.

2147 (U/FOUO) The initial sign-on process must be very robust and secure and based upon the
2148 ancillary enabling technologies of biometrics, multi-factor authentication, tokens, one-time
2149 passwords, and/or strong session establishment protocols. Once the server is certain as to the
2150 entity's identity, that entity's global credentials and/or roles would be provided back to the entity
2151 (e.g., as a ticket, certificate, or SAML assertion), thus enabling follow-on transparent login to all
2152 network resources and applications that are allowed.

2153 (U) Since the credentials/roles are critical, if and when they are sent to the local user client end,
2154 they should be managed and processed only by trusted hardware (e.g., a hardware token or smart
2155 card) that would be immune to malicious sniffing, viruses, or Trojan horses. Transmission of
2156 credential information should be done encrypted so as to protect it while it is in transit.

2157 (U/FOUO) All of the above merely emphasize that SSO technology has the unavoidable effect of
2158 concentrating much potential, aggregated risk in a small number of processes and information
2159 repositories. Nevertheless, the convenience and utility of SSO to the average user is such that the
2160 GIG is certain to feature SSO capabilities. As such, a successful SSO architecture fruition within
2161 the context of the GIG will demand very strong and mature identification and authentication
2162 technologies at the front end along with a robust privilege management infrastructure at the back
2163 end.

2.1.3.6.1 (U) Technical Detail

2164 (U) SSO capabilities have been evolving over a number of years in commercial applications.
2165 SSO has been enabled by a number of technical advances, including strong authentication
2166 techniques, biometrics, and tokens (which allow one-time passwords).
2167

2168 **2.1.3.6.1.1 (U) Early SSO Techniques**

2169 (U) A number of methods have been used over the years by organizations in order to implement
2170 techniques that in limited ways approximate the functionality of SSO. These include login
2171 scripting, password synchronization, and Lightweight Directory Access Protocol (LDAP)
2172 directories, as described below.

2173 **2.1.3.6.1.1.1 (U) Scripting**

2174 (U) Initial commercial techniques developed for SSO included scripting, whose primary goal is
2175 the simple automation of the login procedure, rather than the security enhancement of application
2176 access. In scripting, a user conducts a primary authentication to a SSO authentication server. In
2177 subsequent accesses to various target systems, the client intercepts the standard login dialogue
2178 and then retrieves the appropriate login script from a repository. The client software then merely
2179 forwards the *credentials* (which may merely be an instance of a user ID and password) to the
2180 target system via the login dialogue, achieving a transparent automation of the login procedure
2181 on behalf of the user. The login script repository may reside within the SSO server or may be
2182 downloaded to the client and cached locally.

2183 **2.1.3.6.1.1.2 (U) Password Synchronization**

2184 (U) As can be seen from the above description, scripting is merely a forced automation of the
2185 login procedure across various target systems—each of which may have unique User IDs and/or
2186 passwords associated with a specific user. An evolution of this technique is the concept of
2187 Password Synchronization, in which a password is shared across various systems and can be
2188 updated in a synchronous fashion across all the target systems.

2189 (U) Automatic password synchronization ensures that when a user modifies the password, that
2190 new password is routed network-wide to other target systems. Applying password
2191 synchronization and self-service password reset technologies reduces the number of unique
2192 passwords that a user needs to remember. However, while password policies could be
2193 strengthened for passwords that would be reused to access multiple applications and resources
2194 (with resulting risk aggregation), there is often still a need for the user to respond to each
2195 application's unique login prompt.

2196 **2.1.3.6.1.1.3 (U) LDAP directories**

2197 (U) Other technologies have also contributed to reducing the number of unique sign-ons that are
2198 needed. Fewer application-specific login prompts are required as applications are upgraded to
2199 new software that offers integrated support for authentication to a shared Lightweight Directory
2200 Access Protocol (LDAP) directory. LDAP directory-based authentication generally involves
2201 storing only the cryptographic hash of the user's password, and it may not provide the contextual
2202 credential information about password policies and expiration dates.

2203 (U) Each application would require its own logic to support authentication based on the LDAP
2204 and the credentials maintained in the directory. Through the enabling of LDAP authentication for
2205 target systems, user password information could be made retrievable from any LDAP-supporting
2206 network directory. Each user then has only one password—the LDAP password—to gain access
2207 to all LDAP-enabled target systems.

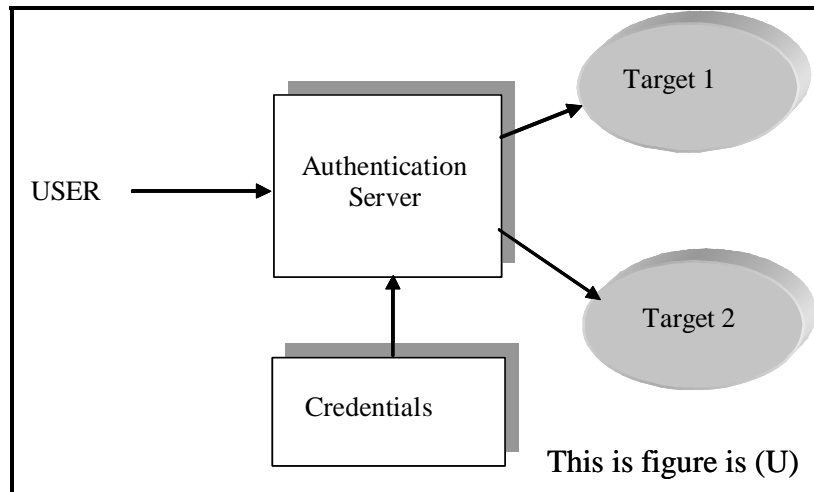
2208 (U) LDAP authentication employs the Simple Authentication and Security Layer (SASL)
2209 protocol implemented between client systems and the directory server. IETF RFCs, which
2210 discuss SASL, include RFC 2222 (*Simple Authentication and Security Layer*,
2211 <http://www.ietf.org/rfc/rfc2222.txt>, by J. Myers, 1997) and RFC 2244 (*The One-Time Password SASL*
2212 *Mechanism*, by C. Newman, 1998). In reality, LDAP authentication only provides for
2213 consolidated sign-on rather than true SSO. The user must authenticate separately on each target
2214 system. Functionality and benefits similar to password synchronization are provided by LDAP
2215 authentication. A potential limitation is that each possible target system must support the LDAP
2216 protocol. Nevertheless, LDAP can still effectively reduce the complexity of password
2217 management within an enterprise.

2218 (U) The advent of strong multi-factor authentication techniques (leveraged upon the enabling
2219 technologies of biometrics, tokens, and one-time passwords) has made it possible to evolve more
2220 fully integrated SSO systems that rely upon the initial very robust authentication to an
2221 authentication server. Then, the as-needed propagation of (encrypted) authorizing credentials and
2222 one-time passwords is sent to each target system as it is encountered. This can follow either a
2223 centralized or a federated architecture model.

2224 **2.1.3.6.1.2 (U) SSO Architectures**

2225 2.1.3.6.1.2.1 (U) Centralized Model

2226 (U) A totally centralized architecture for SSO implementation (as exemplified by the original
2227 Microsoft Passport system) is shown in Figure 2.1-7 below.



2228
2229

Figure 2.1-7: (U) Centralized Architecture for Single Sign-On

2230 (U) In the centralized model, the user signs on to the centralized gate-keeping authentication
2231 server and, if successful, is then automatically signed on to further participating services and/or
2232 applications to which the user is entitled—based on the user’s credentials.

2233 (U) There are several problems with this model. The user must fully trust the authentication
2234 server, which may be problematic if the authentication server is managed by a second party, such
2235 as Microsoft. There also is the potential problem of basic security in that the authentication
2236 server is a single point of failure or central point of attack. Finally, there may be a privacy
2237 problem in that personal information could be collected as part of the authentication information.

2238 (U) Note also that if the centralized authentication server were to be temporarily unavailable, a
2239 user would be precluded from accessing any additional target system during this period.

2240 2.1.3.6.1.2.2 (U) Federated Model

2241 (U) In general, as target systems become more numerous and as networks of systems become
2242 more complex, a centralized architecture becomes too complicated to manage efficiently. In this
2243 case, a federated architecture becomes more desirable. With federated authorization, credentials
2244 are propagated in a less centrally-controlled method than the original Microsoft Passport model.
2245 In addition, as the number of target systems (and even operating systems) proliferates, it is
2246 desirable that the SSO methodology be standards-based. There are currently three standards-
2247 based SSO techniques: Kerberos (via Tickets), PKI (via Certificates), and Security Assertion
2248 Markup Language (SAML) (via Assertions).(U) Since the GIG will have a broad geographic
2249 sweep in addition to a large number of interrelated participating organizations/partners, it is
2250 logical for the GIG to adopt a federated model for Single Sign-On implementation. The three
2251 candidates are described as follows:

2252 2.1.3.6.1.2.2.1 (U) KERBEROS (Tickets)

2253 (U) Kerberos is a password-based authentication protocol/mechanism that is based upon
2254 symmetric cryptography. A user's password does not pass unprotected through a network subject
2255 to potential sniffing attacks by adversaries. Single sign-on can be implemented using Kerberos in
2256 the following manner as shown in Figure 2.1-8.

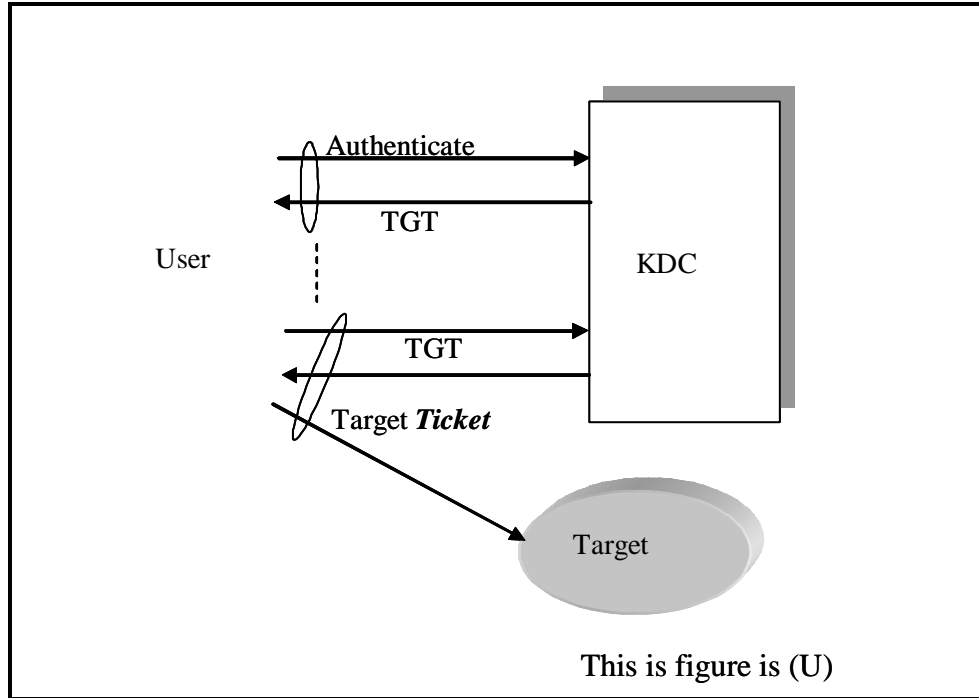


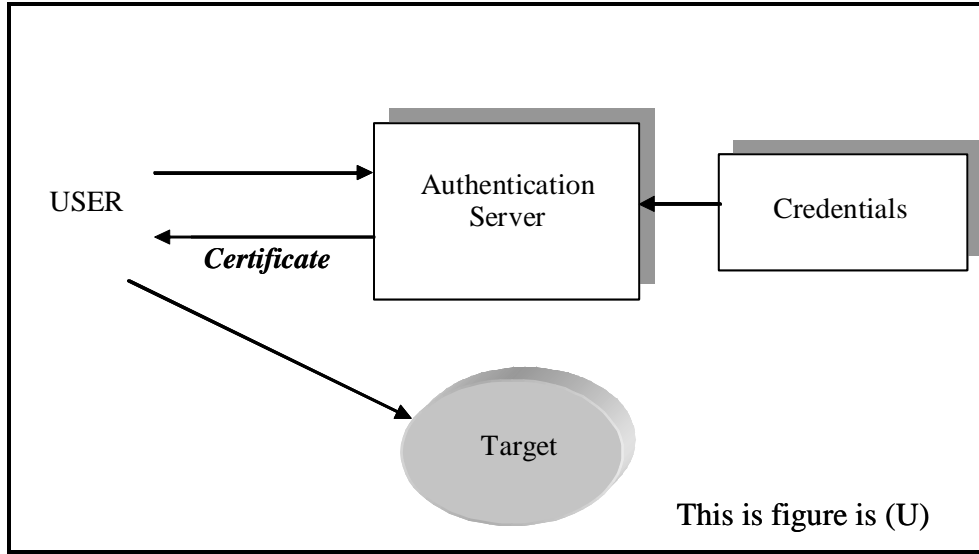
Figure 2.1-8: (U) Federated KERBEROS Based Single Sign-On

2257
2258

2259 (U) Initially, a user would authenticate to a Key Distribution Center (KDC), which would in turn
 2260 issue the user an encrypted Ticket Granting Ticket (TGT). For the lifetime of the TGT (typically
 2261 several hours), the user is authorized to access a given target system by presenting the TGT back
 2262 to the KDC. The KDC in turn then issues an enabling ticket that the user can present to the
 2263 desired target system (without need for further authentication). Kerberos can be used across
 2264 *Kerberized* platforms and/or applications. It is the standard inter-domain authentication protocol
 2265 in Microsoft Windows .NET Server OSs and Windows 2000. Microsoft is updating its original
 2266 basic Passport system using this model (Federated Microsoft Passport). One improvement is that
 2267 a user can acquire a collection of target tickets and subsequently access a variety of target
 2268 systems (within the ticket lifetimes), even if the KDC was to become unavailable due to an
 2269 intervening system failure or KDC communication problems.

2270 2.1.3.6.1.2.2.2 (U) PKI Certificates

2271 (U) A SSO system based upon credential attributes, following the syntax defined by PKI X.509
 2272 certificates, is shown in Figure 2.1-9.



2273

2274

Figure 2.1-9: (U) Federated PKI-based Single Sign-on

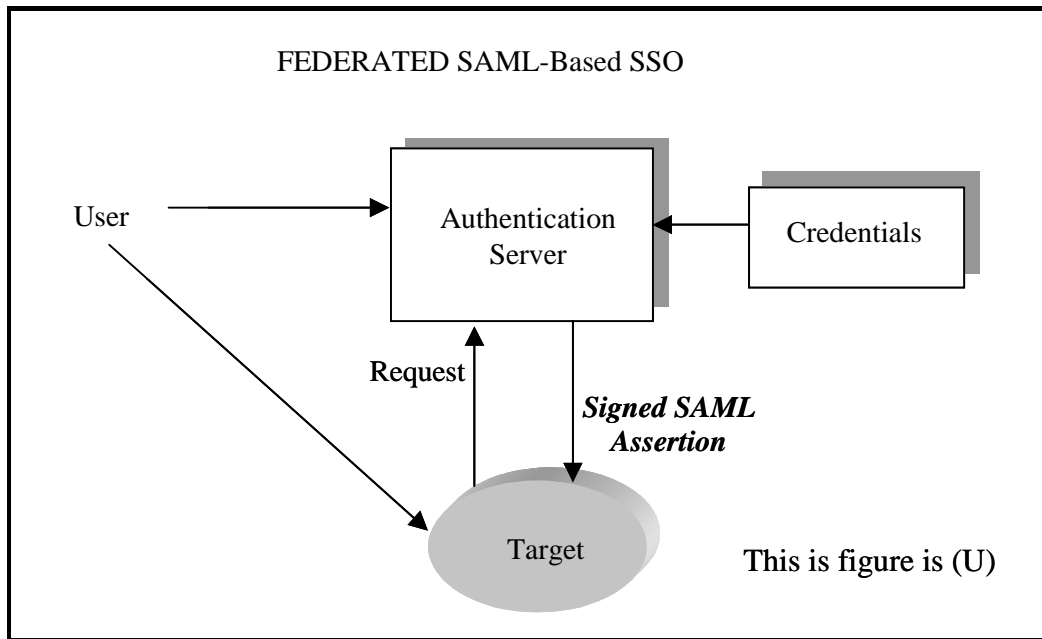
2275 (U) This model is federated in the sense that all the potential target systems are treated as equals
 2276 in that they would each have assigned credential attributes defined within the SSO-enabling
 2277 certificate, and the user may request access at any time to a pre-defined, included target system.
 2278 When the user attempts to login to a candidate target system, it would forward its authorizing
 2279 credentials held within an encrypted version of its attribute certificate. This certificate would
 2280 have been signed by the original authorizing trust authority (using the private key of that
 2281 authority), and it could be thus verified by the target system as authentic through use of the
 2282 originating trust authority's public key. This application of digital signature technology thus
 2283 enables the user to subsequently and transparently login to as many candidate target systems as
 2284 are defined and allowed by the user's credential certificate.

2285 (U) In addition to the certificate being digitally signed by the originating trust authority, it would
 2286 be forwarded to candidate target systems in an encrypted format by using the public key of the
 2287 target system. Any target system could then easily decrypt the *password* attributes through
 2288 application of its own private key. As far as the user is concerned, all of the processing and
 2289 transference of the attribute certificates would be done transparently in the background with the
 2290 user simply accessing the target system and requesting use of available resources.

2291 (U) Use of PKI-based asymmetric key technology could mesh nicely with the maturing DoD PKI
 2292 and its supporting CAC smart card technology, which would retain the private key of each
 2293 respective user.

2294 2.1.3.6.1.2.2.3 (U) SAML (Assertions)

2295 (U) Finally, an alternative SSO implementation may be based upon SAML as shown in Figure
 2296 2.1-10.



2297

2298

Figure 2.1-10: (U) Federated SAML-Based Single Sign-On

2299 (U) Within a SAML-based SSO, the authentication server and all relevant target systems form a
 2300 Circle of Trust to which a user may exercise SSO privileges. It is federated in the sense that the
 2301 circle of trust is a predefined collection of target systems to which the user may potentially wish
 2302 to apply the SSO mechanism. Each of the federated target systems is aware of the existence of
 2303 the authentication server and knows how to request the signed SAML assertion when needed.

2304 (U//FOUO) There are several examples of SAML being applied in projects in the DoD. One of
 2305 these is the U.S. Navy's Space and Naval Warfare Systems Command (SPAWAR) Navy
 2306 Enterprise Portal program, in which SSO capabilities based upon SAML are being introduced in
 2307 order to tie together an estimated 200,000 applications on the Navy-Marine Corps Intranet
 2308 (reached by 720,000 users distributed among active service members, civilian Navy employees,
 2309 and contractors). In an initial demonstration, SAML-enabled SSO was provided to 5,500 users
 2310 aboard the aircraft carrier USS Teddy Roosevelt.

2311 (U//FOUO) Another example of SAML being used in DoD programs is the DISA/DIA (Defense
 2312 Intelligence Agency) Virtual Knowledge Base (VKB) program. As is normally done with SAML
 2313 implementations of SSO, this program uses the XML signature of the SAML assertions to
 2314 provide for the non-repudiation of authentication/authorization credentials. In a prototype
 2315 demonstration, the computation and processing burden of applying digital XML signatures was
 2316 quite manageable and shown to be able to scale well to large user populations. This program also
 2317 looked into the option of employing XML encryption of the SAML assertions in order to provide
 2318 for confidentiality during transport. Unlike the XML signature experience, the XML encryption
 2319 took much more computation time and was shown to not be amenable to scaling well to large
 2320 populations. An alternative to using XML encryption would be to use the SAML implementation
 2321 within established SSL/TLS (Secure Sockets Layer / Transport Layer Security) encrypted
 2322 connections, since SSL is a proven and efficient protocol.

2323 **2.1.3.6.2 (U) Usage Considerations**2324 **2.1.3.6.2.1 (U) Advantages**

2325 (U) There are many clear advantages to SSO. For the individual user the benefits are highlighted
2326 by the convenience of not having to authenticate into each service that is accessed over the web
2327 (and having to remember a large number of passwords).

2328 (U) In turn, SSO serves as a driver to the required supporting technologies of robust, multifactor-
2329 secure authentication (with biometrics, smart cards, etc.) by serving as the gatekeeper at the front
2330 end. It also provides a robustly implemented privilege management infrastructure, which keeps
2331 straight those net resources that a user can access through SSO.

2332 **2.1.3.6.2.2 (U) Risks/Threats/Attacks**

2333 (U) There were some disadvantages associated with the early versions of SSO technologies. For
2334 example, concerning password synchronization: while having a password synchronized across
2335 many applications may be more convenient for the user, it also results in a point of vulnerability.
2336 If a single password can be compromised, this compromises all applications linked to that
2337 password. This *risk aggregation* problem (clearly unacceptable in the GIG) is one of the key
2338 reasons why an earlier generation of so-called enterprise SSO products was not broadly adopted.
2339 Other factors that limited early adoption were the complexity and cost of deployment.

2340 (U//FOUO) As the various SSO standards have been developed and deployed, a number of
2341 additional weaknesses were uncovered. These have led to revising and strengthening the
2342 underlying standard protocols. In 2000, D. Kormann and A. Rubin of AT&T Labs described
2343 weaknesses of the Microsoft Passport SSO protocol in their paper, "Risks of the Passport Single
2344 Sign-On Protocol" (See <http://avirubin.com/passport.html>). They identified three attacks on
2345 Passport: (1) Bogus Merchant Attack (where a user accesses a web site controlled by a malicious
2346 attacker who then proceeds to steal the user's valuable authentication information), (2) Active
2347 Rewrite Attack, and (3) DNS (Domain Name System) Attacks. Requiring SSL security for all
2348 Passport exchanges would protect against the active rewrite attack. Similarly, adoption of
2349 DNSSEC enhancements (See <http://www.dnssec.net/>) would help to protect against DNS attacks.

2350 (U) In 2003 SAML attacks were uncovered by T. Gross of IBM in "Security Analysis of the
2351 SAML Single Sign-On Browser/Artifact Profile" (See
2352 <http://www.acsac.org/2003/papers/73.pdf>). The attacks that were uncovered included Connection
2353 Hijacking / Replay Attack, Man-in-the-Middle Attack (by DNS spoofing), and HTTP Referrer
2354 Attack. Recommended solutions include use of secure channels such as SSL 3.0 or TLS 1.0 with
2355 unilateral authentication for all SAML-related message transfers. Clearly, as the various
2356 competing SSO protocols (Kerberos-based, PKI-based, or SAML-based) are implemented and
2357 studied, additional weaknesses and vulnerabilities may be discovered. This should only lead to
2358 strengthening the protocols as they are revised.

2359 **2.1.3.6.3 (U) Maturity**

2360 (U) Due to the increasing demands for enterprise-wide SSO capabilities, SSO technology has
2361 been maturing at a rapid pace over the past decade—pushed by the competitive pressures of the
2362 commercial marketplace. This has led to a variety of incompatible proprietary implementations,
2363 which has in turn led towards the desirable evolution of standards-based SSO architectures and
2364 protocols. Unfortunately, several distinct and incompatible islands of SSO standards have
2365 emerged (e.g., Kerberos, PKI, SAML), but there also has been a movement towards the
2366 interoperable merger of these standards so that truly universal and cross-platform SSO
2367 capabilities can emerge. In general, these individual technologies can be described as Mature
2368 (TRL 7 – 9).

2369 **2.1.3.6.4 (U) Standards**

2370 (U) The development of the various SSO architectures has been conducted in a number of
2371 formalized standards organizations and industrial vendor alliances. These are discussed below.

2372 (U) There has been some movement towards the interoperability-enabling convergence of the
2373 various SSO standards protocols and their associated camps of supporting vendors. This is
2374 potentially advantageous to the evolution of the GIG, which should not be hindered by the
2375 adoption of security mechanisms that may eventually lose in the standards arena. One example
2376 of this convergence is work on defining SAML assertions in X.509-syntax attribute certificates.
2377 (See the privilege and role management infrastructure standards site at <http://www.permis.org>,
2378 and the NSF Middleware Initiative site at <http://www.nsf-middleware.org/NMIR5/>.) Another
2379 example of similarities between the PKI and Kerberos standards is that X.509 sign-on privilege
2380 attributes can be pre-defined with a validity period of hours or days, just like the Federated
2381 Kerberos-Based SSO architecture with its fixed lifetime tickets. This eliminates the need for the
2382 formalized revocation of X.509 attributes (as compared against the usually infrequent occurrence
2383 of revoking crypto keys in PKI X.509 public key certificates).

2384 (U) It is also interesting to note that the Kerberos V5 version implements extensions to the
2385 original Kerberos protocol to permit initial SSO server authentication using public keys on smart
2386 cards. The original Kerberos protocol relied on symmetric secret key algorithms.

2387 (U) Due to the continued success of each of the standards in its respective application domains, a
2388 mutual convergence of interoperability is preferable to conflict. For example, Kerberos is well
2389 known for certain applications and is supported by modern operating systems, whereas PKI
2390 certificate systems are widely spread (e.g., DoD PKI) and can provide portability across
2391 platforms.

2392 (U) Large and influential vendors such as Microsoft, which has a history of supporting the WS-
2393 Federation, Kerberos-based SSO methodology, have introduced the concept of protocol
2394 transition. This is supposed to be a feature of Microsoft's Windows .NET Server and should
2395 allow a user to gain access to .NET Server-based resources by any one of a number of
2396 authentication mechanisms: Kerberos, PKI X.509 digital attribute certificate, SAML, etc. The
2397 target Windows .NET Server would then transition the sign-on token into a Kerberos ticket for
2398 use in the backend. This is an example of how, if provided with enough appropriate Inter
2399 Working Functions, a conglomeration of SSO standards can be made to interoperate successfully
2400 and securely.

2401 **2.1.3.6.4.1 (U) WS-Federation (Microsoft, IBM)**

2402 (U) The Kerberos-based SSO architecture has been championed primarily by Microsoft and its
2403 WS-Federation standard (promulgated jointly with IBM. See [http://www-
2404 106.ibm.com/developerworks/webservices/library/ws-fed/](http://www-106.ibm.com/developerworks/webservices/library/ws-fed/)). It is based upon the original IETF
2405 RFC 1510, "The Kerberos Network Authentication Service" by J. Kohl and C. Neuman
2406 (September, 1993), found at <http://www.ietf.org/rfc/rfc1510.txt>.

2407 (U) Kerberos, developed at the Massachusetts Institute of Technology, is a system that depends
2408 on passwords and Data Encryption Standard (DES) symmetric cryptography in order to
2409 implement ticket-based, peer entity authentication service, and SSO access control service
2410 distributed in a client/server network environment. Kerberos came out of Project Athena and is
2411 named for the mythical three-headed dog guarding Hades.

2412 (U) The overall Web Services Security Specification roadmap entitled "Security in a Web
2413 Services World: A Proposed Architecture and Roadmap" was promulgated by Microsoft and
2414 IBM in April, 2002. The base layer is called WS-Security, on top of which lie the layers of WS-
2415 Policy, WS-Trust, WS-Privacy, WS-SecureConversation, WS-Authorization, and WS-Federation
2416 (enabling SSO single sign-on). After development of these specifications, they were turned over
2417 to the non-profit OASIS standards body (See below).

2418 **2.1.3.6.4.2 (U) ITU**

2419 (U) The United Nations ITU-T standards organization (<http://www.itu.int/home/>) based in
2420 Geneva, Switzerland has been evolving its PKI-enabling X.509 standard into a standard that will
2421 support SSO-enabling attribute certificates.

2422 **2.1.3.6.4.3 (U) SAML (OASIS)**

2423 (U) The SAML v1.1 standard was approved and promulgated in September, 2003 by the
2424 Organization for the Advancement of Structured Information Standards (OASIS, at
2425 <http://www.oasis-open.org>). Webopedia defines SAML as "an XML (Extensible Markup
2426 Language)-based framework for ensuring that transmitted communications are secure. SAML
2427 defines mechanisms to exchange authentication, authorization and non-repudiation information,
2428 allowing SSO capabilities for web services." This allows organizations to create contractual
2429 federations and enables browsing end-users to reach services using a SSO with appropriate
2430 authentication/authorization information. SAML technology does not define any new
2431 authentication techniques itself, but rather merely enables the existing technology in XML.
2432 SAML is also targeted as a security services implementation to support Internet2.

2433 (U) In order to foster the use of SAML as open source software, OpenSAML
2434 (<http://www.opensaml.org/>) has been developed. It is a set of open source Java and C++ libraries that
2435 are fully consistent with the formal SAML standard specifications. The OpenSAML toolkit may
2436 be licensed royalty-free from RSA.

2437 **2.1.3.6.4.4 (U) Liberty Alliance**

2438 (U) The Liberty Alliance “Project Liberty” (<http://www.projectliberty.org/>) was organized and
2439 introduced in 2001. It is a joint effort by 38 different companies, with Sun Microsystems as the
2440 motivating force. Also involved are staunch supporters of open source software such as the
2441 Apache Software Foundation and O’Reilly & Associates. Other involved technology companies
2442 include Verisign, RealNetworks, and Cisco.

2443 (U) Liberty Alliance is adopting the SAML SSO architecture and protocols. Due to Sun
2444 Microsystems support of SAML, it is being applied in the Java sphere. The related Java
2445 technology API (Application Programming Interface) standard for SAML is covered by Java
2446 Specification Request JSR-155. (See <http://www.jcp.org/>.)

2447 **2.1.3.6.5 (U) Cost/Limitations**

2448 (U) While there are initial costs to implementing a robust and wide-reaching SSO capability, the
2449 eventual return on investment can be huge, and the realization of this is one of the prime drivers
2450 in persuading organizations to adopt SSO technology. When an automated and secure standards-
2451 based SSO system replaces a myriad of existing and disjoint independent traditional sign-on
2452 mechanisms, a tremendous administrative burden is lifted from the shoulders of both the
2453 individual user and the system administrator (e.g., help desks). A broadly adopted standards-
2454 based approach also allows for clearly defined evolution paths for SSO implementation.

2455 **2.1.3.6.6 (U) Dependencies**

2456 (U) Certainly one of the most important dependencies of a robustly secure SSO system is that a
2457 SSO architecture relies greatly on a very strong and secure multifactor initial user authentication,
2458 since if a malicious attacker were to successfully accomplish an invalid initial SSO login, they
2459 would effectively be given the keys to the kingdom of the violated authentic user (or one-stop
2460 shopping for hackers).

2461 (U) The GIG thus is sure to benefit from a robustly developed and standards-based methodology
2462 of SSO. Fortunately, the evolution of SSO technologies is being driven by a number of strong
2463 commercial market forces. Specifically, there are three legislative processes that are requiring
2464 effective SSO capabilities in future commercial IT systems, particularly those dealing with
2465 sensitive—either personal or corporate proprietary—information.

- 2466 • (U) Within the domain of corporate governance, the Sarbanes-Oxley Rule 404 requires
2467 public companies to centralize the reporting of who has access to what and who uses what.
2468 Moreover, business governance and privacy laws in many countries impose similar
2469 requirements.
- 2470 • (U) Similarly, in the financial services market, the Gramm-Leach-Bliley Act specifies the
2471 need for stronger audit and separation of duties, in order to control who, how, and when users

2472 access information and systems.

- 2473 • (U) Finally, the healthcare market is a primary revenue-driving segment for many SSO
2474 vendors. The Health Insurance Portability and Accountability Act (HIPAA) requirement for
2475 an audit trail that associates information access to individual identities becomes mandatory in
2476 April, 2005. Healthcare typically involves the deployment of workstations that need to be
2477 accessed by many healthcare workers, who must frequently and quickly log in and out of
2478 these systems. A robust and secure SSO technique will be very beneficial to this requirement.

2479 **2.1.3.6.7 (U) Alternatives**

2480 (U) The alternative to implementation of an integrated SSO infrastructure within the GIG is to
2481 continue the operation of disparate and independently maintained and administered SSO
2482 mechanisms for each application or resource that GIG users will want to use. A partial solution,
2483 which could be application sensitivity-based in that SSO capability, could be developed for most
2484 of the GIG-spanning resources. However, certain very sensitive (e.g., command and control-
2485 oriented) applications may require independent and rigorously assured authorization and
2486 authentication every time they are accessed. As the GIG-wide SSO solution and supporting
2487 privilege delegation infrastructure matures, the scope of its applicability may indeed expand.

2488 **2.1.3.6.8 (U) References**

2489 (U) Simple Authentication and Security Layer, <http://asg.web.cmu.edu/sasl/>.

2490 (U) “Single Sign-On and the System Administrator”, by M. Grubb and R. Carter (Duke
2491 University), LISA’98 conference,
2492 http://www.usenix.org/publications/library/proceedings/lisa98/full_papers/grubb/grubb.pdf , 6-11 December
2493 1998.

2494 (U) “Single Sign-On Architectures”, presentation by Jan De Clercq (HP),
2495 <http://www.esat.kuleuven.ac.be/cosic/seminars/slides/SSO.pdf>, 2002.

2496 (U) “Identity Management: A Technical Perspective”, presentation by Richard Cisse, e,
2497 [http://www.calt.insead.edu/fidis/workshop/workshop-wp2-
2498 december2003/presentation%5CTUB%5CTUB_fidis_wp2_workshop_dec2003.ppt](http://www.calt.insead.edu/fidis/workshop/workshop-wp2-december2003/presentation%5CTUB%5CTUB_fidis_wp2_workshop_dec2003.ppt), December 2003.

2499 (U) “Single Sign-On Across Web Services”, presentation by Ernest Artiaga, CERN OpenLab
2500 Security Workshop, [http://openlab-mu-internal.web.cern.ch/openlab-mu-
2501 internal/Documents/Presentations/Slides/2004/04-09_EA_Security_Wokshop-SingleSignOn.ppt](http://openlab-mu-internal.web.cern.ch/openlab-mu-internal/Documents/Presentations/Slides/2004/04-09_EA_Security_Wokshop-SingleSignOn.ppt) , April 2004.

2502 (U) “Navy Deploying Its Battle Plan: SAML”, by Anne Chen,
2503 <http://www.eweek.com/article2/0.1759.1502403.00.asp>, 20 October 2003.

2504 (U) “Lessons Learned in a Department of Defense Program (The Virtual Knowledge Base
2505 VKB)”, presentation by Kevin Smith,
2506 http://www.omg.org/news/meetings/workshops/Web%20Services%20USA%20Manual/02-3_K_Smith.pdf.

- 2507 (U) “Web Services Security”, presentation by Sang Shin (Sun Microsystems),
2508 <http://www.javapassion.com/webservices/WebServicesSecurity4.pdf>, January 2004.
- 2509 (U) “SAML Basics”, presentation by Eve Maler, <http://www2002.org/presentations/maler.pdf>, 2002.
- 2510 (U) “Survey of the Status of Security and Emerging Security Innovations for Key Technological
2511 Protocols, Recommendations, Specifications and Standards Used in E-commerce”, by Angela
2512 Mozart, http://www.giac.org/practical/GSEC/Angela_Mozart_GSEC.pdf, November 2003.
- 2513 (U) “Gartner report: Password Management, Single Sign-On, and Authentication Management
2514 Infrastructure Products: Perspective”, by Ant Allan, 7 January 2003.

2515 **2.1.4 (U) I&A Gap Analysis**

2516 (U//FOUO) Gap analysis for the Identification and Authentication Enabler indicates that the
 2517 main areas of required future development are as follows:

- 2518 • (U//FOUO) Complete the development of Protection Profiles for Medium and High
 2519 Assurance authentication technologies (e.g., biometrics).
- 2520 • (U//FOUO) Develop an authentication framework standard that includes SoM levels,
 2521 authentication session scoring, and a SoM forwarding structure.
- 2522 • (U//FOUO) Develop a standard for the methods/protocol of remote access point retrieval
 2523 of authentication privileges.
- 2524 • (U//FOUO) Develop a token with onboard biometric and liveness test (to assure that
 2525 automated logon is not taking place), or offboard biometrics (communicated to token).
 2526 Candidate offboard biometrics are iris scan, retinal scan, face recognition, hand
 2527 geometry, voice recognition, etc. Based on current technology, only a
 2528 thumbprint/fingerprint reader could be integrated directly onto a smart card token.
- 2529 • (U//FOUO) Develop a high assurance DoD PKI Class 5 token w/Type I cryptography
 2530 (where definition of Class 5 token is for use with classified information + hardware token
 2531 + using Type I cryptography + having assurance/trust in security critical functionality
 2532 throughout its lifecycle, including design, development, production, fielding, and
 2533 maintenance).
- 2534 • (U//FOUO) Develop a scalable re-authentication scheduling algorithm, adjustable per
 2535 sensitivity of application, access location, and user profile.
- 2536 • (U//FOUO) Develop a scalable authentication server that is able to interpret and use I&A
 2537 session scores and comply with the GIG authentication standards. The server function
 2538 will need to be secure, efficient, accurate, and transparent in terms of performance
 2539 impact. In addition, it should operate in multiple architectural constructs (e.g., in-line,
 2540 embedded, co-processor, remote).
- 2541 • (U//FOUO) Develop an Identification Registration/Management Infrastructure that can
 2542 support all GIG customers (DoD, IC, and all temporary/permanent partners).
- 2543 • (U//FOUO) Develop a common GIG-wide Single Sign On mechanism, protocol, and
 2544 architecture.
- 2545 • (U//FOUO) Develop a GIG standard for authentication confidence metrics.

2546 (U//FOUO) In addition, the following gaps must be satisfied under other IA System Enablers
 2547 that directly support this IA System Enabler

- 2548 • (U//FOUO) Develop converged standards for Partner Identity Proofing, enabling identity
 2549 interoperability with future GIG partners (e.g., allies, coalition partners, civil government,
 2550 DHS). (See Section 2.7, Management of IA Mechanisms and Assets)

2551 • (U//FOUO) Develop a common identification management and ID proofing standard for
 2552 all future GIG entities (human users, devices, processes). (See Section 2.7, Management
 2553 of IA Mechanisms and Assets)

2554 • (U//FOUO) Ensure metadata standard includes the capability for binding authenticated
 2555 sources to GIG information. (See Section 2.2, Policy-Based Access Control)

2556 (U//FOUO) Technology adequacy is a means of evaluating the technologies as they currently
 2557 stand. This data can be used as a gap assessment between a technology's current maturity and
 2558 the maturity needed for successful inclusion in the GIG in 2008.

2559 (U//FOUO) The following two tables list the adequacy of the Identification and Authentication
 2560 technologies with respect to the enabler attributes discussed in the RCD. Not shown in the tables
 2561 below are entries for Authentication Protocols which are in general quite adequate, in so far as
 2562 their strength and flexibility is concerned.

2563 **Table 2.1-3: (U) Technology Adequacy for Tokens and Biometrics**

This Table is (U)				
		Technology Category		Required Capability (attribute from RCD)
		Tokens	Biometrics	
Enabler Attribute	Standard			IAAU3, IAIR2, IAIR4
	Secure Solution			IAAU1, IAAU3, IAAU8, IAAU9, IAAU18, IAAU19, IAAU20, IAIR1, IAIR6
	Scalable Solution		N/A	IAAU10, IAAU23, IAIR2, IAIR5, IAIR6
	Protection Profile			IAAU1
	High Assurance			IAAU2, IAAU24
	Distributed/Global Reach		N/A	IAAU1, IAAU6, IAAU17, IAAU21, IAIR2
	Verifiable Solution			IAAU1, IAAU12-IAAU15, IAIR1, IAIR3, IAIR4, IAIR5
This Table is (U)				

2564

Table 2.1-4: (U) Technology Adequacy for Single Sign-On and Authentication

This Table is (U)					
		Technology Category			Required Capability (attribute from RCD)
		Single Sign On	Authentication Confidence	Device Authentication	
Enabler Attribute	Standard				IAAU4, IAAU5, IAIR1, IAIR7
	Secure Solution		N/A		IAAU8, IAAU22, IAIR6
	Scalable Solution		N/A		IAAU23, IAIR6, IAIR7
	Protection Profile	N/A	N/A		
	High Assurance		N/A		IAAU22
	Distributed/Global Reach				IAAU6, IAAU25, IAAU23, IAAU21, IAAU17, IAIR7
	Verifiable Solution		N/A		IAIR1
This Table is (U)					

2565

2.1.5 (U) Identification and Authentication: Recommendations and Timelines

(U) The following is a list of preliminary recommendations for advancing the technologies required for the successful implementation of this GIG enabler:

- (U//FOUO) Define a converged Partner Identity Proofing standard that has been vetted and accepted by partner communities.
- (U//FOUO) Develop a common GIG-wide device/service authentication techniques and standards, due to the relative immaturity of this technology area.
- (U//FOUO) Rapidly advance research into the relatively new area of authentication confidence metrics.
- (U//FOUO) Develop a scalable, robust, and distributed authentication server capability.
- (U//FOUO) Develop an accepted high assurance biometric authentication technique.
- (U//FOUO) Assure ongoing and future developments of the DoD CAC Common Access Card will support all future GIG requirements (including Class 5 token).
- (U//FOUO) Advance the selection of a GIG-wide architecture for Single Sign-On (from the candidates described in this document, such as SAML-based or PKI-based). Include in this process the complete analysis of the proposed NCES single sign-on architecture.

(U//FOUO) Figure 2.1-11 contains preliminary technology timelines for this IA System Enabler. These are the result of research completed to date on these technologies. As the Reference Capability Document and the research of technologies related to these capabilities continue, these timelines are expected to evolve.

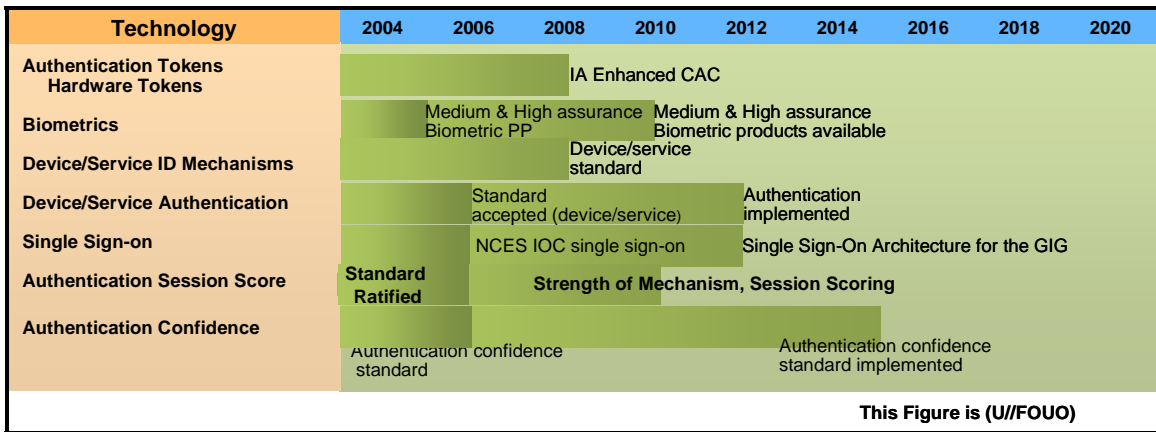


Figure 2.1-11: (U) Technology Timeline for Identification & Authentication

2588 **2.2 (U) POLICY-BASED ACCESS CONTROL**

2589 (U//FOUO) Policy-Based Access Control is the use of flexible, hierarchical rules to
2590 determine whether to grant or deny access to GIG assets at points throughout the GIG.
2591 This policy-based access control capability is also distributed. It provides common GIG
2592 access control services across the enterprise, supports an enterprise wide digital access
2593 policy, and provides decision processing location transparency to the user to improve
2594 availability and load sharing capability. GIG assets include all resources within the
2595 enterprise, such as hardware (e.g., routers, servers, workstations, security components),
2596 software (e.g., services, applications, processes), firmware, bandwidth, information, and
2597 connectivity.

2598 (U//FOUO) From a context prospective, today's information sharing capabilities are not
2599 sufficient to support the net-centric operations vision. Current information sharing is far
2600 too constrained through:

- 2601 • (U//FOUO) A culture that fosters not sharing
- 2602 • (U//FOUO) Physically separate, system-high environments
- 2603 • (U//FOUO) Limitations of information assurance (IA) technology to safely
2604 support assured information sharing

2605 (U//FOUO) Our no-risk culture allows access to classified information only to recipients
2606 who have the proper clearance and a need-to-know. But this accessibility culture must
2607 change to support the vision of information sharing functionality that empowers users
2608 through easy access to information, anytime, anyplace, and anywhere in support of
2609 operational requirements with attendant security.

2610 (U//FOUO) The GIG information sharing philosophy is fundamentally different as it is a
2611 sharing centric security philosophy. The user is presented with information consistent
2612 with such factors as his security clearance, operational situation, privilege and policy,
2613 then decides what information is needed and pulls that information. This differs from the
2614 need-to-show paradigm in which the data originator decides to whom to provide the data
2615 (i.e., no one else knows the data exists).

2616 (U//FOUO) Policy-Based Access Control supports this need to share paradigm and
2617 represents a transformation of historical mandatory and discretionary access control. It
2618 considers security risk and operational need as part of each access control decision. It
2619 thus recognizes that situational conditions (e.g., peacetime, war, terror threat levels,
2620 location of people) will drive the relative weight of operational need and security risk in
2621 determining access.

2622 (U//FOUO) The access control decisions can adapt to varying situational conditions in
2623 accordance with an access control policy. Each policy prescribes the criteria for
2624 determining operational need, the acceptable security risk, and the weighting between the
2625 two under various conditions. Thus the model can support extremely restrictive policies
2626 and also those that provide the widest sharing under specific conditions with added risk.
2627 This new access control model has been named Risk Adaptable Access Control
2628 (RAdAC).

2629 **2.2.1 (U) GIG Benefits due to Policy-Based Access Control**

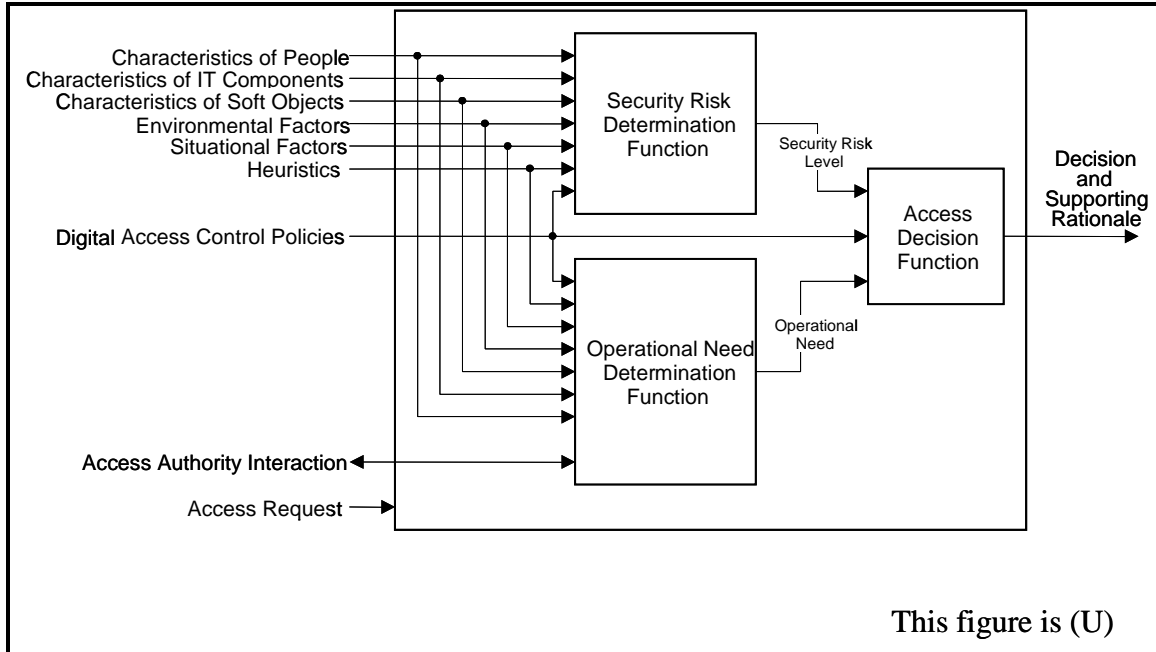
2630 (U//FOUO) The Information Assurance constructs used to support Policy-Based Access
2631 Control provide the following services to the GIG.

- 2632 • (U//FOUO) Provides standardized access control behavior for information,
2633 communications, and services throughout the GIG
- 2634 • (U) Provides fine-grained access control based on the labeled value and life cycle
2635 constraints of the information
- 2636 • (U) Provides fine-grained access control based on the privileges and priority of
2637 the user (user is defined as a human user, entity, or service)
- 2638 • (U//FOUO) Provides ability to segregate multiple communities sharing the GIG to
2639 increase availability while providing dynamic connectivity as needed
- 2640 • (U//FOUO) Supports Single Sign-on (SSO) because an authorization granted is
2641 then recognized throughout the GIG
- 2642 • (U) Allows flexibility to tailor aspects of enterprise policies by region, COIs, C2
2643 Node, etc.
- 2644 • (U) Supports data owner information life cycle policy to track and control object
2645 creation, dissemination, use, and destruction

2646 **2.2.2 (U) Policy-Based Access Control: Description**

2647 **2.2.2.1 (U) Core RAdAC Functions**

2648 (U//FOUO) Policy-Based Access Control is a critical enabler for sharing information and
2649 services within the GIG. Access Control checks will no longer follow the traditional
2650 check for an exact match of mandatory (e.g., credentials) and discretionary (e.g.,
2651 privileges) checks. Instead, the RAdAC Model will be employed. RAdAC is a rule-based
2652 access control policy, based on real-time assessment of the operational need for access
2653 and the security risk associated with granting access. Figure 2.2-1 depicts the RAdAC
2654 model. There are two core functions within RAdAC, Security Risk Determination and
2655 Operational Need Determination.



2656

2657

Figure 2.2-1: (U) RAdAC Functional Model

2658 (U//FOUO) Security Risk Determination provides a real-time, situational, and
 2659 probabilistic determination of the security risk associated with granting the requested
 2660 access. The challenge here is to come up with ways to quantitatively express risk. The
 2661 security risk for granting the access will be determined for at least three different areas:

- 2662 • (U//FOUO) The person receiving the information
- 2663 • (U//FOUO) The IT components the person is using
- 2664 • (U//FOUO) Those that will otherwise be involved in sharing the information

2665 (U//FOUO) Operational Need Determination assesses the operational need of a requestor
 2666 to access some information. A person's membership in some COI or organization, their
 2667 rank or role in an organization, their location, or a supervisor's approval might all be
 2668 contributing factors to establishing their need to know information, but ultimately access
 2669 control policy will specify how to use these factors to determine operational need.

2670 (U//FOUO) An important attribute of Operational Need Determination is the capability of
 2671 allowing an exception to an access control decision. The access control policy would
 2672 specify who is entitled to approve an exception. For example, a commander may
 2673 determine particular data is critical to his mission and grant access to data to which his
 2674 forces would normally not have access. However, the policy must grant the commander
 2675 this right.

2676 **2.2.2.2 (U) Assured Metadata and Data Describing Enterprise Elements**

2677 (U//FOUO) Assured metadata and data describing enterprise elements such as users, IT
2678 components, environment, and situation serve as inputs to the RAdAC functional model.
2679 Not all inputs may be required to make a specific access decision. Digital access control
2680 policy will dictate the minimum decision criteria and how limited input affects the access
2681 control decision.

2682 • (U//FOUO) Characteristics of people who create and consume information will be
2683 used to measure their risk and to determine their operational need. These
2684 characteristics might include identifier, citizenship, security clearance level, and
2685 source of clearance, organization, COI membership, military rank, length of
2686 service, current operational assignment, job title, GIG system privileges—and any
2687 other characteristics that might be usable in determining their security risk and
2688 operational need. Characteristics of the authentication process that granted a
2689 person access to the system would also be included here since multiple proofs of
2690 identity increase how certain the system is concerning the true identity of a
2691 requester.

2692 • (U//FOUO) Characteristics of IT components that create information and enable
2693 users to create, share, and use information will be used to determine security risk.
2694 Determining the robustness of the components is the primary consideration.
2695 Therefore, such things as identifier, operating system, hardware platform features,
2696 current configuration conformance to certified configuration, third-party
2697 robustness evaluation, owning organization, system administrator characteristics,
2698 connectivity to unprotected networks, and software distribution protection might
2699 be characteristics considered when determining the risk associated with IT
2700 components. Furthermore, the operation of these components as a system must be
2701 considered.

2702 • (U//FOUO) Characteristics of Soft Objects contribute to the access decision,
2703 affecting both the security risk measurement and the determination of operational
2704 need. Soft objects include data, applications, and services.

2705 • (U//FOUO) The important characteristics of an object being accessed might
2706 include its identifier, source/originator or controlling entity (including COIs), a
2707 description of the type of data and its value, a description of the data source and
2708 its pedigree, intended roles and expected uses of this object, object life cycle
2709 properties, and traditional labeling information. Object life cycle properties
2710 include object-level attributes that constrain use, dissemination, and disposition
2711 after use.

2712 • (U//FOUO) Traditional labeling information would include such data as
2713 classification level, releasability, and caveat handling. The metadata will be
2714 cryptographically bound to the data to which it applies, so the requestor can
2715 validate the authenticity of the data.

2716 • (U//FOUO) Environmental factors apply to people, IT components, and objects,

2717 and can be used in determining both security risk and operational need.
2718 Environmental factors include such things as a physical location and any
2719 adversarial threat associated with that location. The adversarial threat should be
2720 tied to the GIG operational threat model and risk assessment. It might indicate for
2721 a particular location—or class of locations—the probability that a specific threat
2722 or attack could happen. Location might also be a factor in determining operational
2723 need. All GIG users in a particular location, such as Iraq, might have a need to
2724 access some specific class of information.

2725 • (U//FOUO) Situational factors are national, enterprise-wide, or local indicators of
2726 some situational condition that might affect access control decisions. The terrorist
2727 threat level, for example, might be used to change criteria for determining
2728 operational need. For example, an indication that the enterprise is under cyber
2729 attack or nuclear attack might be other such situational indicators that could affect
2730 access.

2731 • (U//FOUO) Heuristics are intended to represent the knowledge of the information
2732 sharing system that it has acquired from past information sharing and access
2733 control decisions. User-based heuristics might capture previously granted object
2734 access and can be used to help assess current risk and weigh operational need for
2735 future similar access requests. System-based heuristics may capture knowledge of
2736 compromises that have resulted under various access conditions in order to refine
2737 policy to avoid similar future compromises. A policy must specify the degree to
2738 which heuristics should be considered in each access decision.

2739 **2.2.2.3 (U) Digital Access Control Policy**

2740 (U//FOUO) Digital access control policies will be the key to making the RAdAC model
2741 successful. They must be capable of specifying the policy for each step of the access
2742 control process. They must also be capable of expressing rules for various types of access
2743 such as discovery, retrieval, modification, and execution rights. In other words, the
2744 requestor may be able to discover the object/service, but may not have rights to access the
2745 data without verification of need to know.

2746 (U//FOUO) A policy would also be conditional in nature. It could stipulate different rules
2747 of access depending on the current operational condition or mission need. An example
2748 condition might be the current DEFCON level. Under one condition, access might be
2749 limited to those within a COI, while under another condition those with special
2750 operational needs might be given access. Policy flexibility is crucial.

2751 (U//FOUO) Another aspect of digital access control policies is that multiple policies will
2752 exist in the GIG. There will be enterprise level policies and local policies (e.g., COI
2753 policies). The composite set of policies that apply to the object/service will be enforced
2754 during access control checks.

2755 (U//FOUO) For access control to meet the information sharing needs of the GIG, digital
2756 access control policy must extend beyond the initial RAdAC decision through the
2757 inclusion of object life cycle attributes that accompany the soft object. For example, these
2758 attributes will specify whether the entity can save or print or forward the object, whether
2759 it is provided as read only, when the object's lifetime will expire, and what methods are
2760 acceptable for secure disposal of an object.

2761 **2.2.2.4 (U) IA Enabler Dependencies**

2762 (U//FOUO) Identification & Authentication. The authenticity of requester can be
2763 measured through the robustness and number of authenticators used to validate the
2764 requester's identity. Periodic re-authentication may be necessary for a I&A Strength of
2765 Mechanism (SoM) score to be considered viable by the RAdAC model.

2766 (U//FOUO) Protection of User Information. This environment will be a significant factor
2767 in calculating security risk since it is a major portion of the Characteristics of IT
2768 Components input.

2769 (U//FOUO) Dynamic Policy Management. Digital access control policy will be a subset
2770 of the policies managed dynamically in the GIG. The distributed RAdAC function will
2771 require the distribution, synchronization, and revocation capabilities offered by the
2772 Dynamic Policy Management environment.

2773 (U//FOUO) Network Defense and Situational Awareness. RAdAC policy depends upon
2774 the enterprise's Information Condition (INFOCON) and threat levels on suspected or
2775 actual Information Warfare attack as a subset of its Situational Factors input.

2776 (U//FOUO) Management of IA Mechanisms and Assets. RAdAC will depend upon this
2777 enabler to assure use of specific routes that guarantee Quality of Protection, management
2778 enforcement of IT Components with their approved uses and configurations, and
2779 certification & accreditation of enterprise domains as a risk input.

2780 **2.2.3 (U) Policy-Based Access Control: Technologies**

2781 (U//FOUO) For simplicity, the discussion of technologies for Policy-Based Access
2782 Control is divided into three sections:

- 2783 1. (U//FOUO) Core RAdAC that addresses the internal computation of risk and
2784 operational need
- 2785 2. (U//FOUO) Assured Metadata that supports RAdAC decision-making and
2786 enforcement
- 2787 3. (U//FOUO) Dynamic Policy that influences RAdAC decision-making and
2788 enforcement

2789 2.2.3.1 (U) Core RAdAC

2790 (U//FOUO) The core RAdAC functions of security risk and operational need
2791 determination are very new ideas in the access control sphere, both in industry and
2792 Government. Traditionally, both these functions have been handled as administrative
2793 procedures that are then implemented and enforced through a combination of physical
2794 access controls (e.g., locked or guarded facilities) and static, but modifiable, logical
2795 access control business rules (e.g., traditional discretionary access controls in mainstream
2796 operating systems and mandatory access controls in multilevel environments). These
2797 static business rules can be correctly referred to as access control policy, but the
2798 underlying technology essentially assesses a request against a list of authorized actions
2799 and provides a binary allow/disallow decision to an enforcement mechanism.

2800 2.2.3.1.1.1 (U) Technical details

2801 (U//FOUO) IT security risk has historically been a calculation (either qualitative or
2802 quantitative) of the loss expected due to an attack being carried out against a valuable
2803 asset with a specific vulnerability. The exposure of the asset through the vulnerability and
2804 the probability the attack will occur are significant inputs for the final calculation. While
2805 technologies exist to guide a security professional in performing this type of risk
2806 assessment for a business or system, applying this technique to the access control domain
2807 is a very new idea.

2808 (U//FOUO) In the access control domain, soft objects are the information assets that can
2809 be exposed to threats in the environment within a specific situation (including users)
2810 through vulnerabilities in the IT Components themselves. This relationship indicates that
2811 most of the RAdAC inputs affect security risk determination in one way or another—as
2812 described below. A high-level analysis of these RAdAC inputs shows that most will be
2813 textual in nature.

- 2814 • (U//FOUO) Availability and integrity risk - Characteristics of IT components
2815 influence whether these tenets of IA would be placed at risk if access is
2816 authorized. For example, authorizing the release of a 40GB imagery file through a
2817 28kbps tactical circuit would effectively cause a sustained denial of service for all
2818 users of that tactical circuit.
- 2819 • (U//FOUO) Aggregation - Situational Factors should include details of what
2820 information is already available at a user's IT platform to assess the risk of
2821 aggregation (multiple Unclassified documents being combined to learn Classified
2822 information). As multiple services are subscribed to by a single user, the risk of
2823 aggregation (multiple unclassified inputs = classified information) increases.
- 2824 • (U//FOUO) User information and platform context - Consideration of the
2825 classification of current information on the user's IT platform should be
2826 considered alongside the capabilities and assurances of the user's platform. For
2827 example, if a cleared user is subscribed to all FOUO services and requests a
2828 classified document, the risk of disclosure increases greatly if the platform cannot
2829 support MLS or MILS processing.

- 2830 • (U//FOUO) Identity factors - Clearance and formal access approvals of the user,
2831 assurance of the user's identity, and assurance of bindings to roles and COIs are
2832 critical factors to determining risk.
- 2833 • (U//FOUO) Classification lifetime - Classified lifetime of a Soft Object is an
2834 important consideration for risk. If declassification is expected within hours
2835 versus years and the specific operation that the information pertains to is already
2836 underway, the risk of disclosure to an uncleared soldier is much lower than it
2837 ordinarily would be.
- 2838 • (U//FOUO) Violation of traditional access models - All things considered equal,
2839 any access that violates the Bell-LaPadula properties³ should sharply raise the risk
2840 value.
- 2841 • (U//FOUO) Probability of overrun - Risk of disclosure should increase due to the
2842 proximity of enemy forces and probability of overrun. This should be captured in
2843 the Environment Factor.
- 2844 • (U//FOUO) Unavailable input parameters - Lack of input parameters (e.g., no
2845 value for IT environment) or low reliability of input parameters (e.g., non-
2846 authoritative source provides input for an IT environment segment) should
2847 increase the resulting risk due to unknowns.
- 2848 • (U//FOUO) Heuristics - Heuristics from previously authorized similar requests
2849 (proximity with respect to time or content) should result in a reduced security risk.
- 2850 • (U//FOUO) Transitivity - There are transitive security risks to consider in a
2851 highly-connected environment when authorization exceptions are permitted.
2852 Authorizing a classified document to one member of a COI operating at an
2853 unclassified level has implications that reach beyond that individual User making
2854 the access request.
- 2855 • (U//FOUO) External connections - Since policy negotiations between security
2856 domains is a desirable dynamic policy feature, there is a potentially higher risk
2857 that all information released to an external domain should carry. Domain
2858 interconnection only begins to scratch the surface of risks associated with
2859 interconnections within the GIG.
- 2860 • (U//FOUO) Enterprise C&A - GIG risks associated with IT Components within
2861 the enterprise must be considered in RAdAC risk determination. With the
2862 direction DIACAP is heading, near real-time knowledge of GIG system's risks,
2863 countermeasures applied to them, and residual risk that is accepted by a cognizant
2864 approval authority will be available through the eMASS system. The RAdAC
2865 model should interface to the eMASS services to understand residual risk in
2866 systems involved in the access path. This data should be presented to RAdAC via

³ (U) The **Bell-Lapadula Model** of protection systems deals with the control of information flow. It is a linear non-discretionary model.

- 2867 the “Characteristics of IT Components” and “Environment Factors” inputs.
- 2868 • (U//FOUO) Identity Strength of Mechanism - A higher authentication robustness
2869 (e.g., a 3-factor authentication versus 2-factor) should yield a lower risk score.
 - 2870 • (U//FOUO) Soft Object Life Cycle Characteristics - Soft Object characteristics
2871 that limit or preclude widespread dissemination should raise the risk score, and
2872 imposed life cycle characteristics on a specific instance of information such as
2873 “do not copy, do not print, do not further disseminate” may reduce the risk of
2874 disclosure.

2875 (U//FOUO) The other major function of the core RAdAC model is operational need
2876 determination, a function somewhat understood in the administrative domain and much
2877 less understood technologically. Outside of workflow technology that retrieves a
2878 manager’s approval for need-to-know, no technology exists to perform this function.
2879 Characteristics of IT Components will have little to no impact to this function, and
2880 Situational Factors and Heuristics will probably have the most impact.

2881 **2.2.3.1.1.2 (U) Usage considerations**

2882 (U//FOUO) The successful usage of core RAdAC as the GIG access control model will
2883 require substantial proof of correctness, a highly robust distributed design, low-latency
2884 performance, life cycle information management, and significant buy-in from the various
2885 GIG user communities. The shift to a need-to-share philosophy is essential but largely
2886 depends on the assurances that the technology can mitigate risks associated with doing
2887 so.

2888 (U//FOUO) For RAdAC to be successfully deployed and used throughout the GIG, the
2889 existence of any alternate access control mechanisms is problematic. Part of the RAdAC
2890 environment description must address how RAdAC is always invoked and non-
2891 bypassable within the enterprise. This description contributes to the proof of correctness
2892 needed to gain customer acceptance of the technology.

2893 **2.2.3.1.1.2.1 (U) Implementation Issues**

2894 (U//FOUO) Since most of these inputs are textual, RAdAC risk determination should be
2895 performed using technology that can parse, understand meaning, and reason about
2896 relationships under an imposed policy. Otherwise, the performance impact of translation
2897 between text and numeric scores will prove very costly, and RAdAC risks being
2898 inflexible in accommodating more than one ontology.

2899 (U//FOUO) The ontology problem for textual inputs is very significant. In a trivial case,
2900 consider the existing U.S. Air Force and U.S. Navy ontologies used daily. A user
2901 identified with the rank of Captain in the Air Force is an O-3, who is a junior officer
2902 compared to a Navy Captain, who is a senior O-6 typically assigned to commander roles.
2903 Operational Need determination should weigh an Air Force Captain’s verification of an
2904 E-5’s need to know as less than a Navy Captain’s verification of an E-5’s need to know.
2905 A technology that doesn’t understand more than one ontology cannot understand these
2906 distinctions that can be critical in determining access control risk and operational need.

2907 (U//FOUO) To comply with national laws that strictly prohibit disclosure of classified
2908 information to users without appropriate clearances, any immediate implementations of
2909 RAdAC must implement a mathematic model to prove correctness for handling classified
2910 information. To comply with the national law in the near term, this model should map to
2911 traditional Discretionary and Mandatory Access Control (DAC and MAC) models, and it
2912 must never violate the properties established by the Bell-LaPadula confidentiality model.

2913 (U//FOUO) Commercial access control technology is not heading in the direction of risk
2914 calculation. Rather, industry understands the traditional access control models of DAC
2915 and MAC. Role-based Access Control (RBAC) has recently reached maturity, and
2916 Attribute-based Access Control (ABAC) is just beginning to mature. Because of this, the
2917 scope of RAdAC may be more suited to the service-oriented architecture domain rather
2918 than the operating system domain so that both sets of access control models can coexist.

2919 (U//FOUO) RAdAC must be able to offer performance guarantees despite the complexity
2920 of its calculations and the varied inputs required to make a decision. RAdAC must also be
2921 deterministic and produce an access control decision for every request.

2922 (U//FOUO) RAdAC must provide decision rationale to support appeals for operational
2923 need-to-know, audit, and heuristics-based learning.

2924 (U//FOUO) Heuristics implementation can take the form of either user-based or system-
2925 based knowledge of past actions, and most likely both are needed. In either form, the
2926 heuristics data must be verifiably system-recorded (not spoofed or modified) and rapidly
2927 available to the RAdAC decision service. Heuristics is a desirable RAdAC feature that is
2928 not as crucial as other features and can be delayed until later increments.

2929 (U//FOUO) RAdAC's distributed model must be able to support the dismounted soldier
2930 with intermittent connectivity in addition to the CONUS-based desk user and the
2931 enterprise service tier. This distribution model should be able to synchronize updates to
2932 access control policy and information needed to make decisions to support operations in
2933 an offline mode.

2934 (U//FOUO) RAdAC requires assured metadata about Soft Objects and assured data for its
2935 other inputs to make an informed decision and protect itself against well-known security
2936 threats. This assured metadata must be tightly bound to the information it describes and
2937 must itself have verifiable integrity.

2938 (U//FOUO) RAdAC must provide state management to detect and consider repeated
2939 failed access attempts. This state management needs to be extremely lightweight to scale
2940 well in order to support thousands of users.

2941 2.2.3.1.1.2.2 (U) Advantages

2942 (U//FOUO) The RAdAC concept offers the following significant advantages relative to
2943 traditional access control schemes.

- 2944 • (U//FOUO) Supports GIG need-to-share vision through dynamic access control

2945 decision-making that weighs security risk and operational need versus traditional
2946 hard-coded access control

2947 • (U//FOUO) Allows broader scope of inputs that contribute to access control
2948 decision-making, including operational need and situational urgency

2949 • (U//FOUO) Provides fine-grained access decisions (not just “allow” or
2950 “disallow”) that specify required transport path or object life cycle attributes to
2951 secure the risk of granting access

2952 2.2.3.1.1.2.3 (U) Risks/Threats/Attacks

2953 (U) The primary risks to RAdAC are:

2954 • (U//FOUO) Spoofed or altered RAdAC inputs which can allow unauthorized
2955 access

2956 • (U//FOUO) Access DoS attacks (counter detailed in CND RCD section) which
2957 prevent authorized access by legitimate users

2958 • (U//FOUO) RAdAC bypass (direct object access)

2959 • (U//FOUO) Distributed environment synchronization attacks

2960 2.2.3.1.1.3 (U) Maturity

2961 (U//FOUO) Both security risk and operational need determination technologies are in the
2962 conceptual stage. Basic principles have been observed and reported in the Assured
2963 Information Sharing Model white paper, and practical applications are being explored
2964 through a separate study. Technology maturity is rated as Early (TRL 1-3).

2965 2.2.3.1.1.4 (U) Standards

2966 (U//FOUO) Potential standards that loosely apply include:

2967 **Table 2.2-1: (U) Access Control Standards**

This Table is (U)	
Standard	Description
Role-Based Access Control (ANSI INCITS 359-2004)	Describes Role Based Access Control (RBAC) features that have achieved acceptance in the commercial marketplace. It includes a reference model and functional specifications for the RBAC features defined in the reference model
Validated Common Criteria protection profiles	For access control, including Controlled Access Protection Profile, Labeled Security Protection Profile, Role-Based Access Control Protection Profile
Multinational Information Sharing Environment Protection Profile v.1.0	Contains functional and security requirements for sharing information up to Secret among multinational partners

2968

2969 (U//FOUO) Potential supporting commercial technologies include:

2970 **Table 2.2-2: (U) Technologies Supporting Access Control**

This Table is (U)	
Standard	Description
Security Assertion Markup Language (SAML) v2.0	The Security Assertion Markup Language (SAML) is "an XML-based framework for exchanging security information. This security information is expressed in the form of assertions about subjects, where a subject is an entity (either human or computer) that has an identity in some security domain. (W3C standards organization)
eXtensible Access Control Markup Language (XACML) v1.0	OASIS Extensible Access Control Markup Language (XACML) defines a core schema and corresponding namespace for the expression of authorization policies in XML against objects that are themselves identified in XML. (OASIS standard 6 Feb 2003; a working draft of v2.0 is available)
DARPA Agent Markup Language (DAML)	Provides constructs to create ontologies and metadata markup information for machine readability
Web Ontology Language (OWL) v2.0	Provides a language that can be used to describe the classes and relations between them that are inherent in Web documents and applications.
Web DAV Access Control Protocol (RFC3744)	WebDAV stands for "Web-based Distributed Authoring and Versioning". It is a set of extensions to the HTTP protocol which allows users to collaboratively edit and manage files on remote web servers. (IETF standards organization)
Content Based Information Security	Joint Forces Command-sponsored advanced technology concept demonstration that supports the notion of abstracting the complexity of label development from the operator through the use of roles. It also supports the notion of a hierarchy of policies to control sharing.
This Table is (U)	

2971 **2.2.3.1.1.5 (U) Costs/limitations**

2972 (U//FOUO) A large monetary cost will be incurred to design, develop, test, and field
2973 RAdAC into the GIG enterprise since there is no similar commercial technology.

2974 (U//FOUO) A significant performance cost will be associated with access control
2975 decision-making due to the quantity of RAdAC model inputs and the amount of detail
2976 required for these inputs. Current access control technologies compare a request against a
2977 user's identity—and an associated list of authorizations—and then produce a binary
2978 access decision. The complexity of RAdAC will most likely increase the computation
2979 needs for each decision by an order of magnitude.

2980 (U//FOUO) There will also be significant network bandwidth cost due to the transfer of
2981 RAdAC inputs and outputs and the distribution of RAdAC heuristics, although the
2982 distributed design can be optimized to reduce the bandwidth cost.

2983 **2.2.3.1.1.6 (U) Dependencies**

2984 (U//FOUO) Implementation of the RAdAC concept relies on several technologies
2985 covered by other IA System Enablers:

- 2986 • (U//FOUO) Access Control Policy language and associated standards
- 2987 • (U//FOUO) Assured Metadata with integrity verification and reliable binding to
2988 source object
- 2989 • (U//FOUO) Availability of enterprise situation, environment, and IT Component
2990 data with integrity verification features
- 2991 • (U//FOUO) Enterprise Management information regarding domain Certification
2992 & Accreditation and its associated configuration, risks, and threat levels
- 2993 • (U//FOUO) Dynamic Policy Management to push access control policy updates to
2994 distributed RAdAC decision points
- 2995 • (U//FOUO) Requester identity and associated Strength of Mechanism data
- 2996 • (U//FOUO) Assured user profiles for storing user-based access control heuristics
- 2997 • (U//FOUO) Discovery process interface for RAdAC to decide about service
2998 subscriptions (authorization to use a service) and service disclosure (authorization
2999 to know about a service's existence)

3000 **2.2.3.1.1.7 (U) Alternatives**

3001 (U//FOUO) Attribute-based Access Control (ABAC) offers a more dynamic access
3002 control environment than traditional hard-coded access control models since it is based
3003 on attribute-value pairs. Because of its similarity to RAdAC with respect to attribute-
3004 based inputs, this approach offers a significant advantage in the near term while the
3005 harder technical problems of risk determination can be matured through research and
3006 development. ABAC can leverage advances in object metadata and enterprise data (both
3007 in the form of attribute-value pairs) and can be used as a prototype to address some
3008 aspects of operational need determination without requiring the implementation of
3009 security risk determination.

3010 (U//FOUO) In ABAC, the digital access control policy would be simpler than in RAdAC
3011 since it is essentially rules about required attribute-value pairs for access to a Soft Object,
3012 but it does offer dynamic update capabilities through its typical directory-based structure.
3013 This approach can also be paired with the complementary Digital Rights Management
3014 technology (potentially implemented as additional lists of attribute-value pairs) to address
3015 object life-cycle needs. In the long run, this approach will not meet the GIG capabilities
3016 required to fully implement the need-to-share enterprise, but it can be used as an
3017 alternative technology during early increments.

3018 (U//FOUO) Content-Based Information Security uses encryption and key management
3019 techniques to control access to information objects. This approach addresses security risk
3020 during the decision to present an access “key” to a given user based on his or her
3021 clearance, formal access approvals, and need-to-know. The technological burden in this
3022 approach is in the key management rather than on security risk determination or dynamic
3023 policy.

3024 **2.2.3.1.1.8 (U) Complementary techniques**

3025 (U//FOUO) Digital Rights Management, an access control and usage control technology
3026 that uses a combination of metadata-based capabilities, cryptographic techniques, and key
3027 management. The xRML proposed standard offers significant capability to express digital
3028 rights for objects as a set of well-defined attributes.

3029 **2.2.3.1.1.9 (U) References**

3030 (U//FOUO) Role-Based Access Control (ANSI INCITS 359-2004)

3031 (U//FOUO) Validated Common Criteria protection profiles for access control, including
3032 Controlled Access Protection Profile, Labeled Security Protection Profile, Role-Based
3033 Access Control Protection Profile

3034 (U//FOUO) Multinational Information Sharing Environment Protection Profile v.1.0

3035 (U//FOUO) Security Assertion Markup Language (SAML) v2.0 (W3C standards
3036 organization)

3037 (U//FOUO) eXtensible Access Control Markup Language (XACML) v1.0, OASIS
3038 standard 6 Feb 2003; a working draft of v2.0 is available

3039 (U//FOUO) DARPA Agent Mark-up Language (DAML)

3040 (U//FOUO) Web Ontology Language (OWL) v2.0

3041 (U//FOUO) Web DAV Access Control Protocol (IETF standards organization)

3042 (U//FOUO) Content Based Information Security

3043 (U//FOUO) XML Rights Markup Language v2.0

3044 (U//FOUO) Attribute Based Access Control research

- 3045 • (U//FOUO) SPAWAR:
3046 <http://www.networkassociates.com/us/tier0/nailabs/media/documents/atn.pdf>
- 3047 • (U//FOUO) Mitre: <http://portal.acm.org/citation.cfm?id=510781#CIT>

3048 **2.2.3.2 (U) Assured Metadata**

3049 (U//FOUO) GIG Policy-Based Access Control as implemented via RAdAC capabilities
3050 relies on certain information conveyed as inputs to its control decision in a consistent and
3051 known format. A portion of this control decision input is based on the attributes of the
3052 information objects or services that are being requested. These object attributes, including
3053 IA related information are relayed by Metadata. To ensure integrity of objects and
3054 metadata linkage, this metadata is cryptographically bound to the source (information or
3055 service object). Metadata also serves a related function, by providing filterable
3056 information supporting discovery and advertisement of data or service object availability
3057 for access by qualified GIG users.

3058 (U//FOUO) Specific metadata content and labeling for GIG information and service
3059 object is dependant on the object's type. For example, a server-stored information (file)
3060 object may have a far different set of metadata attributes than a real-time session object.
3061 GIG metadata standards will specify and define these required IA attributes per object
3062 type relationships by.

3063 (U//FOUO) The IA related technologies and capability investments that will be required
3064 to enable the GIG vision of Policy-Based Access Control in the metadata area include:
3065 GIG wide language standardization for IA attributes, trusted metadata creation tools,
3066 cryptographic binding of metadata to its source object as well as the ability to reflect and
3067 convey metadata for GIG services.

3068 **2.2.3.2.1 (U) Metadata Language and Standards**

3069 (U//FOUO) Supporting the transition from a GIG need-to-know to a need-to-share
3070 information exchange paradigm will require reliable and trusted mechanisms to
3071 characterize the IA aspects of information or service objects requested by GIG entities.
3072 To provide a reliable supporting mechanism to the GIG Access Control Decision Point
3073 process, metadata language/usage must be standardized regarding syntax, semantics, and
3074 ontology of IA related information. This standardization provides both the owner
3075 (creating organization) and access policy authors with the ability to unambiguously and
3076 consistently communicate attributes regarding data about the information or service
3077 object, as well as define the attributes of the entities that will support access control
3078 decisions for the object instance. This metadata also supports the user information
3079 discovery process by providing filterable information content about GIG publicly
3080 available objects to authorized users—via GIG search applications.

3081 **2.2.3.2.1.1 (U) Technical details**

3082 (U//FOUO) GIG Data owners must have the ability to provide granular expression of the
3083 value of their information through new fields in the metadata tags. These fields will point
3084 to information access policies that define the users, roles, or COIs authorized to access a
3085 specific data asset.

3086 (U//FOUO) The IA Component of the GIG will also implement a notion of Quality of
3087 Protection (QoP) for data assets. As part of tagging a data asset, a set of security-related
3088 properties necessary for protecting the asset would be associated with the asset.
3089 Properties can include how to protect the object as it travels across the network, how the
3090 data object can be routed, or how the data object must be protected while at rest.

3091 (U//FOUO) The purpose of QoP metadata elements differs from the metadata elements
3092 used to describe the contents of an asset. Content-description metadata elements are
3093 designed to enable data discovery and sharing. QoP metadata tags define how the data
3094 object is to be protected while at rest and in transit. This concept, for instance, will allow
3095 the GIG to require routing of highly classified or sensitive information through a more
3096 trusted (i.e., better protected) portion of the GIG or require that a user's client support
3097 encryption of the information in storage before granting access to the information.
3098 Clearly one of the technical issues surrounding these metadata QoP designations are the
3099 mechanisms of transformation—especially for transport from metadata to routing
3100 request/selection information.

3101 (U//FOUO) Another important aspect when considering metadata usage within the GIG
3102 is to consider the types (classes) of objects being requested for access and the potential
3103 action context of these object classes. Objects in this context are any information, service,
3104 session, application, streaming media, metadata or other resource to which access will be
3105 controlled in the GIG. Objects are described as being active or passive with respect to the
3106 access control decision process. An object is considered active if it is the cause of the
3107 access control decision (i.e., an active object is one that is requesting access to some other
3108 object/entity). An object is considered passive if it is the entity that will be shared as a
3109 result of the access control decision (i.e., a passive object is the one that is being
3110 requested by some other object/entity). There are many classes of objects that will exist
3111 in the GIG and be involved in access control decisions. Some possible classes include:

- 3112 • (U//FOUO) Information objects include any data file, report, document,
3113 photograph, database element, or similar types of data object. It might also
3114 include metadata that describes other objects. Information objects are arguably the
3115 core objects as they typically are what is being shared. They represent all the
3116 information that will be resident in the GIG or made available to the GIG from
3117 information originators/creators (e.g., Intelligence Community). They are usually
3118 passive (that which is being acted upon), and thus their IA attributes often define
3119 the minimal requirements for access to the object.

- 3120 • (U//FOUO) Service objects are executable applications that provide some
3121 function for the GIG. They are the services in a service-oriented architecture.
3122 Service objects can be both active and passive objects of an access control
3123 decision. Services will provide portals to useful information and computational
3124 resources on the GIG. People will need to be granted access to services to use
3125 them and thus services can also be the passive object of an access control
3126 decision. In addition, services can be expected to make independent requests of
3127 other services and information objects, or may make requests on behalf of people.
3128 When making requests on behalf of a person, services might be expected to
3129 provide their own IA attributes to the access control decision process along with
3130 those of the person. When independently accessing other services (e.g., service to
3131 service interactions), service objects are active objects in the access control
3132 decision process.

 - 3133 • (U//FOUO) Session objects are objects that are created as a result of a real-time
3134 collaboration between two or more people. A telephone call, a video
3135 teleconference, or an online virtual meeting, are examples of collaborative
3136 sessions that produce session objects. Session objects are in essence a
3137 representation of the collaborative session. They have attributes that describe key
3138 characteristics of the session. Session objects will generally be passive objects in
3139 an access control decision, and thus the IA attributes of the session will be used to
3140 grant or deny access to the session. There may be cases where a session object is
3141 also an active object as it might request content be added to the session, such as a
3142 data file (e.g., PowerPoint presentation).

 - 3143 • (U//FOUO) Real-time objects are a special class of information objects. Examples
3144 of real-time objects are live streaming video and voice, as well as real-time
3145 network management/control traffic exchanges. What makes real-time objects
3146 special is the temporal aspect of the objects (saving samples to disk turns real-
3147 time objects into normal information objects, i.e., these real-time objects are not
3148 retained to persistent storage media). Attributes that describe real-time objects
3149 must be assigned a priori and thus must be generalized to what the real-time
3150 object is expected to be. For IA attributes, this means that the security relevant
3151 features of the streaming information must be anticipated. Once IA attributes are
3152 established, they will live through the duration of the real-time object.
- 3153 (U//FOUO) Metadata IA attributes are the foundation of making access control decisions
3154 in the GIG. There needs to be a universal agreed-upon set of IA attributes across the GIG.
3155 These attributes, in effect, provide a vocabulary for describing security actions. Without a
3156 common vocabulary, it is quite difficult, if not impossible, to make meaningful decisions
3157 about sharing information. Table 2.2-3 shows the minimum set of IA attributes needed to
3158 support policy based access control decision-making via the RAdAC information-sharing
3159 model, based on the class of the object.

Table 2.2-3: (U) Minimum Set of IA Attributes for Access Control Decisions

This Table is (U)	
Category	IA Attribute Description/Requirement
Passive object	Identifier: Provide the GIG unique designation for the object
Passive object	Sensitivity Level: Provide a standards-based designation of object classification and perishability timeframe (**include Operational Need Modifier structure)
Passive object	Data Owner Community of Interest: GIG standards-based COI designator for the organization/activity responsible for creation of the object
Passive object	Access Control Information List/Policy (Direct Data or Pointer): GIG Standards-based Pairing of entities that are allowed access to an object (COI, individual, individual w/ Role/Privilege or groups) and the operations the entity is allowed to perform (read, write, execute, etc.) on the requested object. (**include Operational Need Modifier structure)
Passive object	Time to Live: Length of time an object can be used before it is destroyed automatically by the system as part of an automated life cycle management capability
Passive object	Originator: GIG unique and authenticated identifier linked to the person, organization, or entity that created the object
Passive object	Releaseability: Standards-based designator of countries or GIG external organizations with whom the object may be shared (**include Operational Need Modifier structure)
Passive object	Sanitization Supported: Identifies if real-time sanitization of the object is supported.
Passive object	Security Policy Index: GIG standards-based policy language specifies the various procedures for the object with flexibility/structure to include access protection policy (entity authentication, platform, environment and operational factor scoring) and QoP (**include Operational Need Modifier structure)
Passive object	QoP object life cycle attributes (view only, printable, no-forward, destroy after view, digital rights, etc.) (**include Operational Need Modifier structure)
Passive object	Location: GIG Standards-based designation of virtual path to the object's storage location
Passive object	Timestamp: Time/date information when the object was created or copied.
Passive object	Integrity mechanism: Insure that unauthorized changes to the information object and its IA attributes can be detected
Passive object	Cryptobinding: Cryptographic binding and metadata (supporting access control decision making) to the source object. (Supports prevention of direct access to object w/o metadata based access control decision processing)
Passive object	Split or IA capable filtering of Metadata: Support for both discovery and access control processes
Passive object	Classification/releasability of descriptive metadata itself (not the source object)
Session object	Member IA Attributes: GIG Standards-based listing (pointers) of mandatory privilege/identity IA attribute and value pairings
Session object	Access Control List: List of GIG unique identifier for people allowed to join session paired with GIG unique identifier for approval authority
Session object	Security Level: GIG standards-based parameter indicating how the security level of the session is to be controlled (fixed/float)

This Table is (U)	
Category	IA Attribute Description/Requirement
Session object	Session Archive Control: GIG standards-based parameters indicating archive/recording and classification marking required
Session object	Owner/Moderator ID: GIG unique identifier of session owner/moderator
Session object	Session Members: GIG unique identifier of current/past session members
Session object	Session Identifier: Standards-based unique identifier for the session.
Service object	For Access Requests coming from a service object (acting as proxy for the source entity) this structure must address GIG unique ID of service object, as well as GIG unique ID of requesting source <i>EDITOR'S NOTE: REMAINING SPECIFIC IA ATTRIBUTES FOR SERVICE OBJECT TYPES ARE CURRENTLY UNDER INVESTIGATION</i>
Real-time object	<i>EDITOR'S NOTE: SPECIFIC IA ATTRIBUTES FOR REAL-TIME OBJECT TYPES ARE CURRENTLY UNDER INVESTIGATION</i>
This Table is (U)	

3161 (U//FOUO) **The RAdAC model describes an approach to access control whereby
 3162 operational necessity can override security risk. In this context, IA attributes might have
 3163 'modifiers' in addition to values. Specifically, each designated IA Attribute might have a
 3164 modifier that describes which, if any, exceptions/overrides to normal policy might be
 3165 permitted relative to that attribute. Thus, when an access control process is making a
 3166 decision whether to permit or deny access and encounters a mismatch on a particular IA
 3167 Attribute, it may use the modifiers in an effort to reach a decision that supports sharing.

3168 **2.2.3.2.1.2 (U) Usage considerations**

3169 (U//FOUO) The successful usage of a standardized metadata language supporting access
 3170 control decisions will require a clearly defined and consistently implemented set of IA
 3171 Attributes and supporting infrastructure/tools capabilities. This set of IA related attributes
 3172 (labels); their syntax, semantics, and taxonomy form a critical link in the GIG automated
 3173 access control and discovery processes. The usage and meaning of these IA Attributes
 3174 must be understood and/or supported via user assisting infrastructure especially for the
 3175 roles of information owner, access control policy author, and access privilege (operation
 3176 override) authority. Incorrect usage of these IA Attributes (labels) could result inability to
 3177 discover or access information by GIG users with the correct operation need and
 3178 clearance. On the other side of the scale, incorrect IA Attribute usage could result in
 3179 unintended or unauthorized disclosure of information to a compromised GIG user or
 3180 service entity.

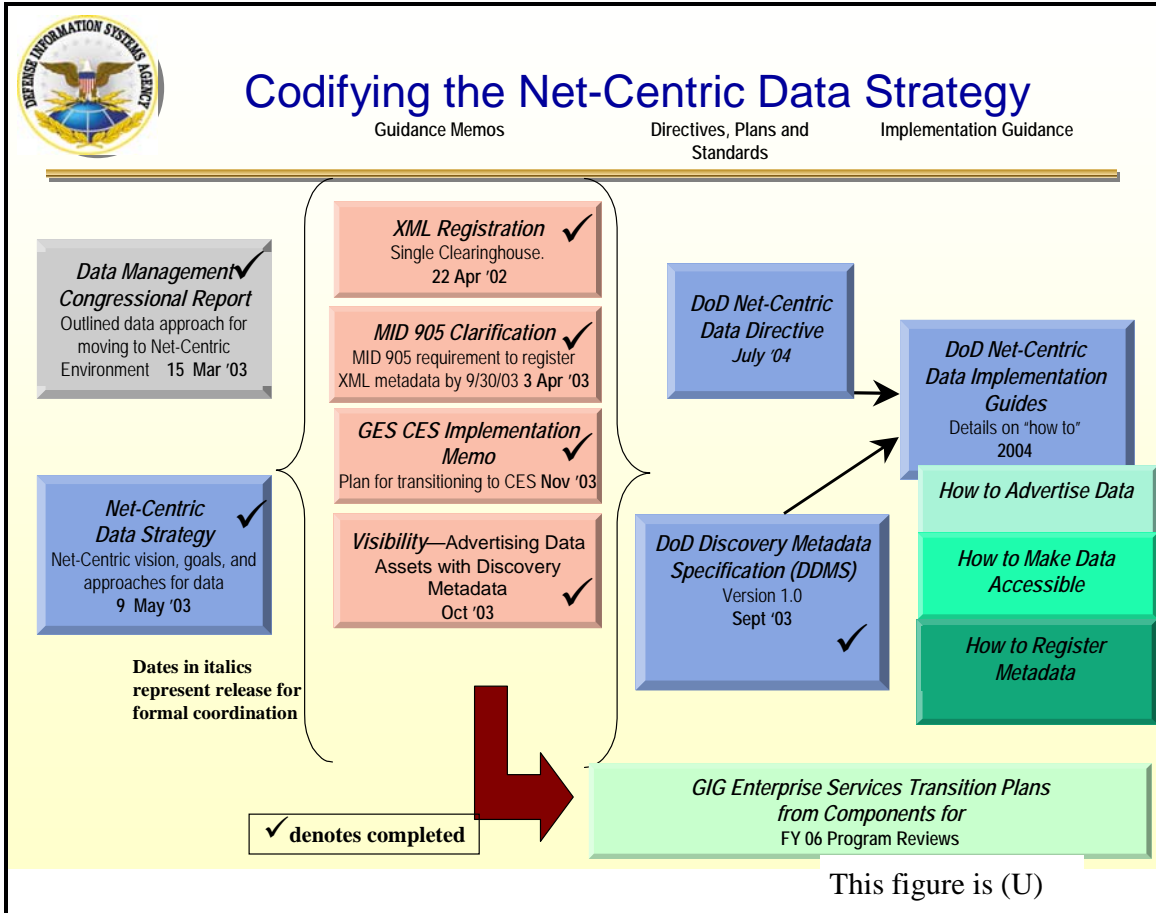
3181 (U//FOUO) Currently there are two known standards bodies working within the GIG to
 3182 define metadata language principles for use by their communities. The primary purpose
 3183 of each group’s products are different, and neither standard provides the entire IA
 3184 Attribute suite needed to support the Policy-Based Access Control Enabler as envisioned
 3185 in the RAdAC model (See Table 2.2-3 for detailed analysis). However, the Core
 3186 Enterprise Services (CES) Metadata Working Group, now led by DISA, is attempting to
 3187 ensure commonality between itself and the IC Metadata Working Group (see attribute
 3188 comparison Table 2.2-4). Further, discussions have been initiated with both standards
 3189 groups to investigate and integrate the required IA Attribute and supporting language
 3190 semantics/syntax into these implementation documents and infrastructures.

3191 **Table 2.2-4: (U) IC and CES Metadata Working Groups Attribute Comparison**

This Table is (U)		
Core Layer Category Set	DDMS Attributes	IC MSP Attributes
The Security elements enable the description of security classification and related fields	Security: Classification Dissemination Controls Releasable To	Security: Classification Dissemination Controls Releasable To
Resource elements enable the descriptors of maintenance and administration information	Title Identifier Creator Publisher Contributor Date Rights Language Type Source	Title Identifier List AuthorInfo Publisher Co-authorInfo Date Rights Language IntelType Source
The Summary Content elements enable the description of concepts and topics	Subject Geospatial Coverage Temporal Coverage Virtual Coverage Description	Subject Geospatial Temporal Virtual Description
The Format elements enable the description of physical attributes to the asset	Format	Media Format
This Table is (U)		

3192 (U//FOUO) The IC Metadata Working Group has developed an XML-based standard and
3193 schema that supports containers for security marking as prescribed by the CAPCO
3194 standard. IC MSP is an implementation of the World Wide Web Consortium's (W3C)
3195 specification of the Extensible Markup Language (XML). It consists of a set of XML
3196 attributes that may be used to associate security-related metadata with XML elements in
3197 documents, web service transactions, or data streams. It is distributed as both an XML
3198 entity set and a W3C XML Schema (WXS) so that the XML attributes defined in the
3199 standard can be incorporated into any XML document type definition (DTD) or schema.
3200 The IC ISM entity set and WXS are controlled vocabularies of terms that are used as the
3201 sources for the values of the IC ISM attributes. The IC MSP schemas incorporate the
3202 classification and controls attributes defined by the IC Metadata Standard for Information
3203 Security Markings (IC ISM). The IC ISM provides the IC with a standard method for
3204 tagging CAPCO authorized markings and abbreviations on XML-based information. The
3205 standard provides flexibility for each agency to implement their security policy and
3206 granularity with respect to security marking.

3207 (U//FOUO) The DoD's Discovery Metadata Specification (DDMS) and supporting XML
3208 schema produced by the Core Enterprise Services (CES) Metadata Working Group
3209 defines discovery metadata elements for resources posted to community and
3210 organizational shared spaces. "Discovery" is the ability to locate data assets through a
3211 consistent and flexible search. The DDMS specifies a set of information fields that are to
3212 be used to describe any data or service asset that is made known to the enterprise. This
3213 CES document serves as a reference guide by laying a foundation for Discovery Services.
3214 The document describes the DDMS elements and their logical groupings. It does not
3215 provide an interchange specification or substantive implementation guidance. However,
3216 there is a roadmap for the development of implementation guides in line with this and
3217 higher-level GIG directive documents (see Figure 2.2-2).



3218

3219

Figure 2.2-2: (U) Codifying the Net-Centric Data Strategy

3220

2.2.3.2.1.2.1 (U) Implementation Issues

3221

(U//FOUO) Some IA metadata attributes sets for information objects may change over time and due to the impact scale. The IA metadata standards/language and supporting infrastructure must support the ability to point (index) to a trusted secondary source for current version attribute information. For instance, if a departmental access policy were hard coded into metadata for all of that department's products, potentially large numbers of information objects must be modified to new hard-coded values if a change policy change occurs over time in this area

3228

(U//FOUO) The metadata language standard must include fields (IA Attributes) within the metadata tag that allow access control decisions to be made on the metadata itself. For example, in some instances, security code words or compartment names are classified themselves

3229

3230

3231

3232

(U//FOUO) It is also paramount, given the critical nature of the metadata tags, that appropriate integrity, data origination and in some cases traffic flow security measures are applied and that the metadata label be securely bound to the object

3233

3234

- 3235 (U//FOUO) The use of IA attribute modifiers (as described above) will add significant
3236 complexity to the IA metadata standards definition.
- 3237 (U//FOUO) IA metadata attributes will be needed to support both GIG Access Control
3238 and Discovery processes. If implementation decisions drive segregation of IA Attributes
3239 to differing location (virtual or physical) synchronization of new or changed IA attributes
3240 must be addressed
- 3241 (U//FOUO) Ontology of metadata (referring to input factors for RAdAC computation) is
3242 extremely important so that computation logic correctly assesses the risk and the
3243 operational need (e.g., is this a Navy “Captain” endorsing operational need or an Air
3244 Force “Captain”)
- 3245 (U//FOUO) It is unclear what implementation method can support the transport-related
3246 Quality of Protection (QoP) IA metadata attributes into the transport infrastructure to
3247 support routing decisions. For the data at rest portion of IA QoP attributes, commercial-
3248 based Digital Right Management capability may provide acceptable and compatible
3249 methods give further investigation.
- 3250 2.2.3.2.1.2.2 (U) Advantages
- 3251 (U//FOUO) Supports GIG need-to-share vision though discovery process and movement
3252 away from “determine at the time of creation” access control lists. (Creator of
3253 information may not know who has need of information produced)
- 3254 (U//FOUO) Supports finer granularity in access control decision making logic
- 3255 (U//FOUO) Support policy based vs. hard coded, access control decision making that
3256 enables rapid changes in GIG situational and environmental factors as well as operational
3257 need
- 3258 2.2.3.2.1.2.3 (U) Risks/Threats/Attacks
- 3259 (U//FOUO) Attempts to access information object directly on server location by passing
3260 metadata/RAdAC Access Control processes
- 3261 (U//FOUO) Confidentiality of some portions of metadata itself
- 3262 (U//FOUO) Discovery DOS attacks
- 3263 (U//FOUO) Access DOS attacks
- 3264 (U//FOUO) Metadata tags that include compromised identity of the original source of the
3265 information and of any entities (e.g., processes) that have modified it prior to posting in
3266 its current form
- 3267 (U//FOUO) Compromised metadata is presented to discovery users (e.g., metadata is
3268 maliciously hidden, out of date metadata is maliciously presented)

3269

2.2.3.2.1.3 (U) Maturity

3270

(U//FOUO) As described above, the two GIG standards organizations (CES Metadata Working Group and IC Metadata Working Group) are in the process of defining metadata standards and implementation schemas. These standards are being designed for implementation, using mature and tested commercial standards for internet communication including XML and OWL. Further, GIG usage of XML to support metadata is being configuration managed and standardized via the DOD Metadata Registry and Clearing house (<http://diides.ncr.disa.mil/mdregHomePage/mdregHome.portal>). Therefore, technical readiness level has been assessed in the Early range (2-3).

3271

3272

3273

3274

3275

3276

3277

3278

2.2.3.2.1.4 (U) Standards

3279

Table 2.2-5: (U) Metadata Standards

This Table is (U)	
Standard	Description
Department of Defense Discovery Metadata Specification (DDMS) Version 1.1	Defines discovery metadata elements for resources posted to community and organizational shared spaces. "Discovery" is the ability to locate data assets through a consistent and flexible search.
Intelligence Community Metadata Standards for Information Assurance, Information Security Markings Implementation Guide, Release 2.0	An implementation of the World Wide Web Consortium's specification of the Extensible Markup Language (XML). It consists of a set of XML attributes that may be used to associate security-related metadata with XML elements in documents, webservice transactions, or data streams.
Intelligence Community Metadata Standard for Publications, Implementation Guide, Release 2.0	A set of XML document models that may be used to apply metadata to analytical data to produce publications. IC MSP prescribes element models and associated attributes for use in marking up document-style products for posting on Intelink and other domain servers.
Federal Information Processing Standard FIPS PUB 10-4, April, 1995, Countries, Dependencies, Areas of Special Sovereignty, and Their Principal Administrative Divisions	Provides a list of the basic geopolitical entities in the world, together with the principal administrative divisions that comprise each entity.
Extensible Markup Language (XML) 1.0 (Second Edition) W3C Recommendation, 6 October 2000	Describes a class of data objects called XML documents and partially describes the behavior of computer programs which process them.
Web Ontology Language (OWL) Guide Version 1.0, W3C Working Draft 4 November 2002	Provides a language that can be used to describe the classes and relations between them that are inherent in Web documents and applications.
This Table is (U)	

3280 **2.2.3.2.1.5 (U) Costs/limitations**

3281 (U//FOUO) More resources and time will be required to develop, produce, and maintain
3282 these IA related metadata attributes than today's basic security markings and MAC/DAC
3283 characteristics. This cost can be off set to some degree by the use of automated metadata
3284 creation tools

3285 (U//FOUO) Legacy DoD information and service objects that currently exist or will be
3286 produced before metadata standards and infrastructure are available may need to be
3287 retrofitted with standard IA Attributes to support RAdAC access control and Discovery
3288 process

3289 (U//FOUO) The use of trusted metadata type information tagging for real-time and
3290 session object types will increase the GIG's transport and network traffic overhead.
3291 Performance impacts should also be investigated early in the design process

3292 (U//FOUO) To avoid the need to retrofit metadata for very large quantities of information
3293 objects, IA metadata attributes syntax and semantics must remain "stable" or remain
3294 backwards compatible.

3295 **2.2.3.2.1.6 (U) Dependencies**

3296 (U//FOUO) Access Control Policy Language Standards

3297 (U//FOUO) Metadata Creation Tools

3298 (U//FOUO) Identity and Privilege Management Capacities

3299 **2.2.3.2.1.7 (U) Alternatives**

3300 (U//FOUO) Depending on the final fidelity/functionality and transition sequence of
3301 RAdAC, functionality-less IA Attribute could be included in the metadata language and
3302 standards. However later additions could result in metadata, large-scale retrofit impacts.

3303 **2.2.3.2.1.8 (U) Complementary techniques**

3304 (U//FOUO) Digital Rights Management

3305 **2.2.3.2.1.9 (U) References**

3306 (U//FOUO) Department of Defense Discovery Metadata Specification (DDMS) Version
3307 1.1

3308 (U//FOUO) Intelligence Community Metadata Standards for Information Assurance,
3309 Information Security Markings Implementation Guide, Release 2.0

3310 (U//FOUO) Intelligence Community Metadata Standard for Publications, Implementation
3311 Guide, Release 2.0

3312 (U//FOUO) Federal Information Processing Standard FIPS PUB 10-4, April, 1995,
3313 Countries, Dependencies, Areas of Special Sovereignty, and Their Principal
3314 Administrative Divisions.

3315 **2.2.3.2.1.10 (U) Technology/Standards Analysis**

3316 **Table 2.2-6: (U) Metadata Gap Analysis**

This Table is (U)					
Category	IA Attribute Description/Requirement	Req. Source	Existing Standards Coverage (Y/N) Identifier	Gap Description/Recommendation	Recommendations and/or Remarks
Passive object/MD Creator Entry	Identifier: Provide GIG unique designation for the object	Tiger Team Report 5/26/2004	Y (IC MSP) Y (DDMS)		IC MSP requires a Universal Unique ID, Identifier List, and a Public Document No.
Passive object/MD Creator Entry	Sensitivity Level: Provide a standards based designation of object classification and perishability timeframe (**include Operational Need Modifier structure)	Tiger Team Report 5/26/2004	Y (IC MSP) Y (IC ISM) Y (DDMS)	Recommend DDMS implement (by reference) the IC ISM markings	IC ISM allows all CAPCO classification markings and dissemination constraints including declassification instructions IC MSP employs IC ISM markings on all block object element types and in the descriptive metadata for the source data DDMS only implements DoD 5200.1-R and does not currently express foreign, SCI, or non-standard classification or declassification

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U)					
Category	IA Attribute Description/Requirement	Req. Source	Existing Standards Coverage (Y/N) Identifier	Gap Description/Recommendation	Recommendations and/or Remarks
Passive object/MD Creator Entry	Data Owner Community of Interest: GIG standards based COI designator for the organization/activity responsible for creation of the object	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	IC MSP: Make Affiliation a required field and ensure it aligns to GIG COI designator IC MSP and DDMS: Make UserID a required field and ensure it maps to globally unique GIG UserID DDMS: Make Organization a required field	IC MSP allows specification of 1+ POC's information in PersonalProfileGroup, but Affiliation is optional and COI is missing
Passive object/MD Creator Entry	Access Control Information List/Policy (Direct Data or Pointer): GIG Standards-based Pairing of entities that are allowed access to an object (COI, individual, individual w/ Role/Privilege or groups) and the operations the entity is allowed to perform (read, write, execute, etc.) on the requested object. (**include Operational Need Modifier structure)	Tiger Team Report 5/26/2004	N	See comments/questions	For Access Requests coming from a service object (acting as proxy for the source entity), this structure must address GIG unique ID of service object, as well as GIG unique ID of requesting source
Passive object/MD, Creator Entry	Time to Live: Length of time an object can be used before it is destroyed automatically by the system as part of an automated life cycle management capability	Tiger Team Report 5/26/2004	Y (IC MSP) Y (DDMS)		Supports information cutoff and information "death" dates in the DateList element (IC MSP) and Date element (DDMS)

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U)					
Category	IA Attribute Description/Requirement	Req. Source	Existing Standards Coverage (Y/N) Identifier	Gap Description/Recommendation	Recommendations and/or Remarks
Passive object/MD, Creator Entry	Originator: GIG unique and authenticated identifier linked to the person, organization, or entity that created the object	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	IC MSP: Make Affiliation a required field and ensure it aligns to GIG COI designator IC MSP and DDMS: Make UserID a required field and ensure it maps to globally unique GIG UserID DDMS: Make Organization a required field	IC MSP allows specification of 1+ POC's information in PersonalProfileGroup, but Affiliation is optional and COI is missing
Passive object/MD, Creator Entry	Releaseability: Standards-based designator of countries or GIG external organizations with whom the object may be shared (**include Operational Need Modifier structure)	Tiger Team Report 5/26/2004	Y (IC MSP) Y (IC ISM) Y (DDMS)		Support CAPCO and DoD 5200.1-R compliant releasability markings
Passive object/MD, Creator Entry	Sanitization Supported: Identifies if real-time sanitization of the object is supported.	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Add optional element containing URI to metadata for alternate source or acceptable sanitization service. The URI to alternate metadata should contain a security classification.	

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U)					
Category	IA Attribute Description/Requirement	Req. Source	Existing Standards Coverage (Y/N) Identifier	Gap Description/Recommendation	Recommendations and/or Remarks
Passive object/MD, Creator Entry/View	Security Policy Index: GIG standards based policy language specifies the various procedures for the object w/ flexibility/structure to include access protection policy (entity authentication, platform, environment and operational factor scoring) (**include Operational Need Modifier structure)	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Add mandatory element that indexes the organization security policy that governs access to the information. Index can take the form of a URI or a UUID. Intent is that though the policy may change over time, this reference to it won't need to.	
Passive object/MD, Creator Entry/View	Object lifecycle attributes (view only, printable, no-forward, destroy after view, etc.) (**include Operational Need Modifier structure)	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Need to add Digital Rights Management (or equivalent attribute fidelity) capability to specify read, modify, forward, copy, destroy, print types of constraints	There's a primitive structure in place for rights management in the Rights element, but it only supports copyright and Privacy Act flags
Passive object/MD, Creator Entry/View	Location: GIG Standards-based designation of virtual path to the object's storage location	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Add SourceURI field to AdministrativeMetadata element	
Passive object/MD, Creator View	Timestamp: Time/date information when the object was created or copied.	Tiger Team Report 5/26/2004	Y (IC MSP) Y (DDMS)		IC MSP: DateList element contains DatePosted, DatePublished, DateReviewed, DateRevised fields DDMS: Date element contains DateCreated

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U)					
Category	IA Attribute Description/Requirement	Req. Source	Existing Standards Coverage (Y/N) Identifier	Gap Description/Recommendation	Recommendations and/or Remarks
Passive object/MD, Creator No Entry/View	Integrity mechanism: Insure that unauthorized changes to the information object and its IA attributes can be detected	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Append an Integrity element with MetadataIntegrity field and SourceIntegrity field that include name/type/URI of integrity mechanism used	
Passive object/Infrastructure	Cryptobinding: Cryptographic binding and metadata (supporting access control decision making) to the source object. (Supports prevention of direct access to object w/o metadata based access control decision processing)	GIG IA Arch Docs	N (IC MSP) N (DDMS)	Add a Security element with crypto algorithm designator/URI and a portion of the key needed to access the source	
Passive object/Infrastructure	Split or IA capable filtering of Metadata: Support for both discovery and access control processes	GIG IA Arch Docs	Y (IC MSP) Y (DDMS)		IC MSP splits Administrative Metadata from Descriptive Metadata DDMS is designed to enable discovery and access control filtering on a single "metacard"
Passive object/Infrastructure	Classification/releasability of descriptive metadata itself (not the source object)	GIG IA Arch Docs	N (IC MSP) N (DDMS)	Descriptive Metadata only describes the security classification of the source object	
Session object/Owner Entry/View	Member IA Attributes: GIG Standards based listing (pointers) of mandatory privilege/identity IA attribute and value pairings	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Need to add Element structure that specifies the common qualities of People who can participate in the session	

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U)					
Category	IA Attribute Description/Requirement	Req. Source	Existing Standards Coverage (Y/N) Identifier	Gap Description/Recommendation	Recommendations and/or Remarks
Session object/Owner Entry/View	Access Control List: List of GIG unique identifier for people allowed to join session paired with GIG unique identifier for approval authority	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)		
Session object/Owner Entry/View	Security Level: GIG standards based parameter indicating how the security level of the session is to be controlled (fixed/float)	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Add an binary "Fixed" field in the Security element of session DescriptiveMetadata	
Session object/Owner Entry/View	Session Archive Control: GIG standards-based parameters indicating archive/recording and classification marking required	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Need to add archive/recording fields (Y/N) and URI for archive/recording	Covers classification markings only
Session object/Owner Entry/View	Owner/Moderator ID: GIG unique identifier of owner/moderator of the session	Tiger Team Report 5/26/2004	N (IC MSP) N (DDMS)	Can be adapted using Elements from the PersonalProfileGroup	Assumption: This person is responsible for granting access to the session and is responsible for allowing information objects to be shared in the session
Session object/Owner /View	Session Members: GIG unique identifier of current/past session members	Tiger Team Report 5/26/2004	Y (IC MSP) Y (DDMS)		UUID should suffice for this purpose
Session object/Owner /View	Session Identifier: Standards based unique identifier for the session.	Tiger Team Report 5/26/2004	Y (IC MSP) Y (DDMS)		UUID should suffice for this purpose
This Table is (U)					

UNCLASSIFIED//FOR OFFICIAL USE ONLY

3317 **2.2.3.2.2 (U) Trusted Metadata Creation Tools**

3318 (U//FOUO) The IA metadata attributes are a key element of access control decisions that are at
 3319 the heart of assured information sharing. Given the pivotal role of these attributes, the policies
 3320 and supporting creation tools/infrastructure used to generate them can be leveraged to help
 3321 encourage—or even enforce—the appropriate level of data sharing across the enterprise.

3322 (U//FOUO) It is envisioned that automated process/tools can be developed to support the
 3323 business processes of the GIG community and can translate these business processes into sharing
 3324 policies that assist in the application of IA metadata attributes for both sharing and required
 3325 information security. While such a robust translation capability is beyond the ability of current
 3326 technologies, the general notion of turning business processes and natural language statements
 3327 about organization's processes into a machine-readable metadata, supporting policy, tools and
 3328 infrastructure is supported by current technology—the issue is one of robustness and
 3329 sophistication. If such a robust capability can be created, it will allow automated processes to
 3330 facilitate appropriate levels of sharing and security by assisting in the creation of the object
 3331 metadata IA Attributes.

3332 (U//FOUO) It should be noted that IA Attributes will form only a portion of the overall metadata
 3333 for GIG information objects. However, due to the critical nature of these elements, a significant
 3334 amount of complexity and added processing interfaces will be needed to support this metadata
 3335 subset.

3336 **2.2.3.2.2.1 (U) Technical details**

3337 (U//FOUO) The following listing provides a brief inventory of capabilities and interfaces that
 3338 will be required of trusted metadata creation tools and IA attributes for the GIG.

- 3339 • (U//FOUO) Identity and Privilege management interface: Ensure that the entity
 3340 (user/process) is authenticated and has the correct privilege to create/validate this
 3341 metadata for the data owning organization.
- 3342 • (U//FOUO) Object Identifier CM interface: Assign GIG unique object identifier
- 3343 • (U//FOUO) Access Control Policy Interface: Allow user to link the correct access
 3344 control policy or access control list (based on information owner organization's business
 3345 rules) as well as directive/pointers to related transport QoP and life cycle QoP policy
- 3346 • (U//FOUO) Operational Need Entry supporting structure (IA attributes 'modifiers' in
 3347 addition to values. IA attributes might have a modifier that describes which, if any,
 3348 exceptions to normal policy might be permitted relative to that attribute)
- 3349 • (U//FOUO) Metadata Integrity Mechanism Interface: Ensure unauthorized changes to
 3350 metadata are detected
- 3351 • (U//FOUO) Discovery metadata filtering structure/policy, allows portions of metadata to
 3352 be filtered from search results unless the user possess required clearance level

3353 • (U//FOUO) Cryptographic Binding Interface: Supports trusted binding of the metadata
3354 to the source information object when metadata has been successfully created (syntax
3355 check complete)

3356 • (U//FOUO) Trusted transport interface, required for assure pull and push of information
3357 related to metadata creation process

3358 (U//FOUO) The following listing provides an inventory of capabilities may be common to the
3359 overall metadata creation process (non IA attribute unique)

3360 • (U//FOUO) Context Sensitive User Help Capabilities

3361 • (U//FOUO) Syntax Checker Capability (Note: This may be present for standard metadata
3362 requirements; the unique IA attribute will likely add significant code size and interface
3363 complexity)

3364 **2.2.3.2.2.2 (U) Usage considerations**

3365 (U//FOUO) With the advent of XML-based metadata that supports web document publishing and
3366 search application, creation tools and templates have been developed to assist users and
3367 document owners with the generation/maintenance of supporting metadata. From the prospective
3368 of commercial standards, most of these supporting tools are based on the Dublin Core Initiative.

3369 (U//FOUO) The Dublin Core Metadata Initiative is an open forum engaged in the development
3370 of interoperable online metadata standards to support a broad range of purposes and business
3371 models. The Dublin Core supports standard schemas in both XML and RDFS. Customized
3372 metadata creation and maintenance tools—based on the Dublin Core schema—are then
3373 developed that reflect the required metadata purpose and business model/policies. These
3374 metadata creation/maintenance tools are designed and implemented either by the information
3375 owning organization or through customization of commercially available products.

3376 (U//FOUO) However, the IA metadata attributes will require additional capability to ensure trust,
3377 security, and unique GIG environment requirement are met for both discovery and access control
3378 processes. These IA-related characteristics of metadata support/generation tools were defined in
3379 section 2.2.3.2.2.1.

3380 **2.2.3.2.2.2.1 (U) Implementation Issues**

3381 (U//FOUO) Metadata creation tools may have to support GIG minimum standards related to both
3382 discovery and access control as well as providing the user with information related to specific
3383 organizational policy. As discussed above, the criticality of the IA Attributes form an access
3384 control prospective and will probably make these tools complex. Finally, these tools must be
3385 widely distributed, available to a user in a timely manner, be intuitive to a human in their use,
3386 and support greater levels of automation during final program timeframes.

3387 2.2.3.2.2.2 (U) Advantages
 3388 (U//FOUO) Metadata creation tools and supporting infrastructure provide the user or
 3389 organization entity responsible for creation of data improvements with accuracy and aid with
 3390 population of the correct metadata information (especially IA Attribute) vs. manual (template
 3391 only) methods.

3392 2.2.3.2.2.2.3 (U) Risks/Threats/Attacks
 3393 (U//FOUO) Unauthorized user “checks in” malicious file/metadata to info storage
 3394 (U//FOUO) Unauthorized user attempt to change IA attribute of metadata to gain information
 3395 object access

3396 (U//FOUO) Authorized user changes metadata

3397 (U//FOUO) Metadata creation tool DOS Attack

3398 (U//FOUO) Compromised metadata creation tool source software

3399 **2.2.3.2.2.3 (U) Maturity**

3400 (U//FOUO) Clearly, there have been successful implementations and commercial products that
 3401 provide metadata creation tools based on the Dublin Core metadata standard. As such, the overall
 3402 technology would receive a TRL score of 7-8. However, the IA related capabilities and interfaces
 3403 as defined in section 2.2.3.2.2.1 are new, complex and unique to this GIG implementation.
 3404 Further, one of the key required predecessors needed is a stable metadata standard for IA
 3405 Attributes. As discussed in the Metadata Language and Standards section, this activity is in the
 3406 early stages of technology development. Therefore, we assess the overall TRL score for this
 3407 technology in the Early range (1-2).

3408 **2.2.3.2.2.4 (U) Standards**

3409 **Table 2.2-7: (U) Metadata Tool Standards**

This Table is (U)	
Standard	Description
Department of Defense Discovery Metadata Specification (DDMS) Version 1.1	Defines discovery metadata elements for resources posted to community and organizational shared spaces. “Discovery” is the ability to locate data assets through a consistent and flexible search.
Intelligence Community Metadata Standards for Information Assurance, Information Security Markings Implementation Guide, Release 2.0	An implementation of the World Wide Web Consortium’s specification of the Extensible Markup Language (XML). It consists of a set of XML attributes that may be used to associate security-related metadata with XML elements in documents, webservice transactions, or data streams.
Intelligence Community Metadata Standard for Publications, Implementation Guide, Release 2.0	A set of XML document models that may be used to apply metadata to analytical data to produce publications. IC MSP prescribes element models and associated attributes for use in marking up document-style products for posting on Intelink and other domain servers
Dublin Core Metadata For Resource Discovery, (RFC 2413 IETF)	Defines interoperable metadata standards and specialized metadata vocabularies for describing resources that enable more intelligent

This Table is (U)	
Standard	Description
	information discovery systems.
This Table is (U)	

3410 **2.2.3.2.2.5 (U) Costs/limitations**

3411 **2.2.3.2.2.6 (U) Dependencies**

- 3412 • (U) Access Control Policy Service
- 3413 • (U) GIG Information Object Identity Assignment Service
- 3414 • (U) Identity and Privilege Service
- 3415 • (U) Cryptographic Binding Service

3416 **2.2.3.2.2.7 (U) Alternatives**

3417 (U//FOUO) Manual (template-based metadata entry) forms with limited syntax checking and
 3418 external interfaces may be sufficient for the early stages of Dynamic Access (RAdAC based)
 3419 Control

3420 **2.2.3.2.2.8 (U) References**

3421 (U//FOUO) Department of Defense Discovery Metadata Specification (DDMS) Version 1.1

3422 (U//FOUO) Intelligence Community Metadata Standards for Information Assurance, Information
 3423 Security Markings Implementation Guide, Release 2.0

3424 (U//FOUO) Intelligence Community Metadata Standard for Publications, Implementation Guide,
 3425 Release 2.0

3426 (U) Dublin Core Metadata For Resource Discovery, RFC 2413 IETF

3427 **2.2.3.2.3 (U) Crypto-Binding of Metadata to Source Information Object**

3428 (U//FOUO) Cryptographically, binding the metadata describing an information object to its
 3429 source object provides a critical access control integrity mechanism. Crypto-binding ensures at
 3430 the time of creation or authorized modification that a trusted linkage is established between the
 3431 two components of an information object (source info and metadata). This capability becomes
 3432 important to GIG’s implementation of Policy Based Access Control via RAdAC because
 3433 metadata is one of the primary, determining information inputs for access control decisions.
 3434 Without crypto-binding, the metadata could be altered or maliciously pointed to an invalid
 3435 metadata tag in order to gain unauthorized access to a source information element.

3436 **2.2.3.2.3.1 (U) Technical details**

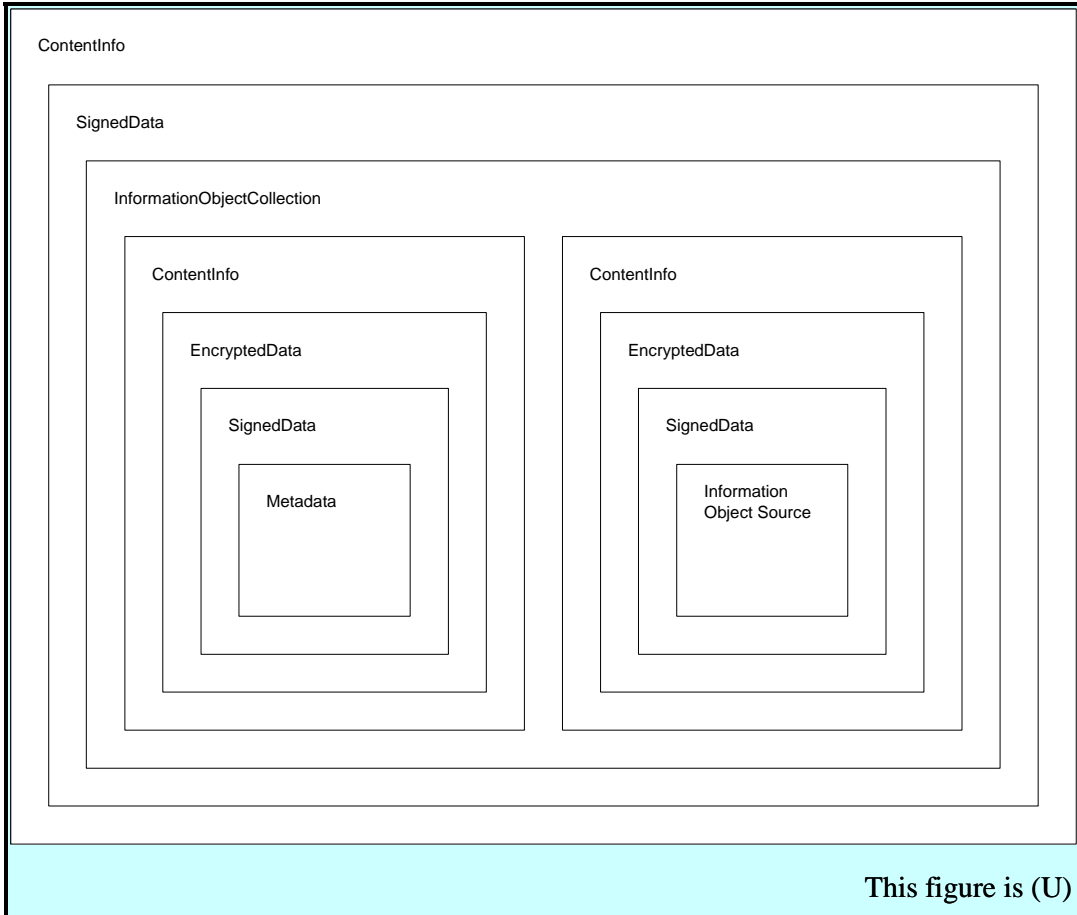
3437 (U//FOUO) The following list provides a brief inventory of capabilities and interfaces that will
 3438 be required of crypto-binding of metadata to its source information object for the GIG.

- 3439 • (U//FOUO) Interface capability to GIG metadata creation tools/services
- 3440 • (U//FOUO) Interfaces to accept and process key and/or digital signature information (as
3441 required)
- 3442 • (U//FOUO) Provide up to type 1 assurance of binding and digest functions for metadata
3443 and its source information object
- 3444 • (U//FOUO) Ability support for rapid decryption of digest (hash file) and return original
3445 component files upon receipt of properly authorized command

3446 **2.2.3.2.3.2 (U) Usage considerations**

3447 (U//FOUO) Research to date indicates that the best standards technology available today to meet
3448 the required capabilities of the Cryptographic Binding function are best implemented via the
3449 Cryptographic Message Syntax, (RFC 2630) standard. The Cryptographic Message Syntax
3450 (CMS) was derived from PKCS #7 version 1.5 as specified in RFC 2315 [PKCS#7].

3451 (U//FOUO) CMS is a data protection encapsulation syntax that employs ASN.1 [X.208-88,
3452 X.209-88]. This syntax is used to digitally sign, digest, authenticate, or encrypt arbitrary
3453 messages. It supports digital signatures, message authentication codes, and encryption. The
3454 syntax allows multiple encapsulations, so one encapsulation envelope can be nested inside
3455 another. This capability aligns well with the needs defined for the cryptographic binding
3456 functionality (see Figure 2.2-3 Encapsulation Notional diagram).



3457

3458

Figure 2.2-3: (U) Encapsulation Notional Diagram

3459 (U//FOUO) CMS implementations must include the SHA-1 message digest algorithm (defined in
3460 FIPS Pub 180-10). CMS implementations should include the MD5 message digest algorithm
3461 (defined in [RFC 1321](#)) as well. CMS implementations must include DSA signature algorithm
3462 (defined in FIPS Pub 186). CMS implementations may also include RSA signature algorithm
3463 (defined in [RFC 2347](#) for use with SHA-1 and MD5).

3464 2.2.3.2.3.2.1 (U) Implementation Issues

3465 (U//FOUO) The decision as to whether this crypto-binding and decrypt function is a central GIG
3466 service or a local plug in to affected applications may affect overall performance, network
3467 overhead, and user perception

3468 2.2.3.2.3.2.2 (U) Advantages

3469 (U//FOUO) CMS is flexible and nesting levels are expandable to meet program needs

3470 (U//FOUO) CMS has been successfully implemented in commercial and government network
3471 environments

3472 (U//FOUO) CMS provides flexibility to program selection of message digest and signature
 3473 algorithms. Further, as new encryption and signature algorithms the CMS syntax structure can be
 3474 expanded to accommodate movement in the technology state of the art.

3475 2.2.3.2.3.2.3 (U) Risks/Threats/Attacks

3476 (U//FOUO) Decryption Analysis/Attack

3477 (U//FOUO) Compromised digital signatures

3478 **2.2.3.2.3.3 (U) Maturity**

3479 (U//FOUO) Elements of the CMS have a successful lineage from PKCS#7 and a wide variety of
 3480 successful implementation examples in both commercial and DoD environments for the base
 3481 encryption, binding, and linkage function. However, the interfaces to other GIG
 3482 applications/services and potential distributed nature of this function will drive a small to
 3483 moderate level of new development. As such, we judge the overall TRL level of this technology
 3484 to be in the Early to Emerging range (3-4).

3485 **2.2.3.2.3.4 (U) Standards**

3486 **Table 2.2-8: (U) Standards on Cryptographic Binding**

This Table is (U)	
Standard	Description
Cryptographic Message Syntax, IETF (RFC 2630)	This syntax is used to digitally sign, digest, authenticate, or encrypt arbitrary messages.
PKCS #7 version 1.5 9 IETF (RFC 2315)	Describes a general syntax for data that may have cryptography applied to it, such as digital signatures and digital envelopes. The syntax admits recursion. It also allows arbitrary attributes, such as signing time, to be authenticated along with the content of a message, and provides for other attributes such as countersignatures to be associated with a signature.
SHA-1 (FIPS Pub 180-10)	Standard specifies a Secure Hash Algorithm, SHA-1, for computing a condensed representation of a message or a data file
MD5 IETF (RFC 1321)	Standard describes the MD5 message-digest algorithm. The algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input.
Hashed Message Authentication Codes (FIPS PUB 198)	Standard describes a keyed-hash message authentication code (HMAC), a mechanism for message authentication using cryptographic hash functions. HMAC can be used with any iterative Approved cryptographic hash function, in combination with a shared secret key.
This Table is (U)	

3487 **2.2.3.2.3.5 (U) Dependencies**

3488 (U) Key management infrastructure

3489 (U) Metadata standards/infrastructure

3490 **2.2.3.2.3.6 (U) Alternatives**

3491 (U//FOUO) SHA-1 in concert with RSA signature service could be implemented and used
 3492 without standardized syntax (CMS). Syntax relation processing and infrastructure would need to
 3493 be maintained in entirety by DoD/GIG.

3494 **2.2.3.2.3.7 (U) Complementary techniques**

3495 (U) Described in section 2.2.3.2.3.2.

3496 **2.2.3.2.3.8 (U) References**

3497 (U) Cryptographic Message Syntax, IETF (RFC 2630)

3498 (U) PKCS #7 version 1.5 9 IETF (RFC 2315)

3499 (U) SHA-1 (FIPS Pub 180-10)

3500 (U) MD5 IETF ([RFC 1321](#))

3501 (U) RSA IETF ([RFC 2347](#))

3502 (U) Hashed Message Authentication Codes (RFC 2401)

3503 **2.2.3.3 (U) Digital Access Control Policy**

3504 (U//FOUO) Influencing all aspects of the RAdAC model is the digital access control policy
 3505 (DACP). It serves as an input to the Core RAdAC functions and as the deciding factor for
 3506 allowing or denying access. Although RAdAC will need specific capabilities in its DACP, these
 3507 policy needs should fold into the larger GIG dynamic policy effort. Some potential technologies
 3508 being examined for that enabler are WS-Policy, Standard Deontic Logic, and artificial
 3509 intelligence constructs. The scope of this section addresses only the RAdAC-specific needs for
 3510 DACP and assumes that the dynamic policy enabler provides the necessary distributed
 3511 functionality (e.g., secure update, revocation, currency validation, and caching for off-line use).

3512 (U//FOUO) The RAdAC model depicts DACP as influencing all aspects of internal RAdAC
 3513 behavior. In this role, DACP must be expressive enough to address the following:

- 3514 • (U//FOUO) Minimum number of required inputs to calculate risk and operational need
- 3515 • (U//FOUO) Relative weighting of the various inputs for risk and operational need
- 3516 • (U//FOUO) Relative weighting of risk versus operational need for the final decision
- 3517 • (U//FOUO) Ability to express stateful access control rules (e.g., successive failed access
 3518 attempts)
- 3519 • (U//FOUO) Ability to express policy according to enterprise and COI roles
- 3520 • (U//FOUO) Ability to negotiate two or more conflicting access control rules
- 3521 • (U//FOUO) Ability to negotiate access control policy with neighboring security domains

- 3522 in order to define an access control boundary interface that is agreeable to both sides
- 3523 • (U//FOUO) Ability to express and automatically select between multiple policies based
 - 3524 on nationality or security domain
 - 3525 • (U//FOUO) Ability to express more granular or more restrictive access control policies at
 - 3526 each successive echelon down the chain of command
 - 3527 • (U//FOUO) Ability to dynamically tighten or loosen access control policy based on
 - 3528 situation (INFOCON, proximity to enemy forces, etc.).
 - 3529 • (U//FOUO) Ability to reach a decision deterministically within bounded time

3530 (U//FOUO) DACP also requires expressiveness to support RAdAC output. For example, the
 3531 policy engine may recognize a specific request as having a compelling operational need but
 3532 having too risky an IT Component to release the information to. In this case, policy should be
 3533 expressive enough to conclude that an alternate path (alternate Course of Action, or COA) for
 3534 this LIMFAC should be examined. For this role, DACP expressiveness must address:

- 3535 • (U//FOUO) Ability to understand and specify in human- and machine-readable terms the
- 3536 limiting factors (LIMFACs) that contributed to a failed access attempt
- 3537 • (U//FOUO) Ability to reason whether an alternate COA could sway the decision (e.g., an
- 3538 uncleared user attempting to access Top Secret information could never be allowed—
- 3539 regardless of the QoP offered by a specific route—because of national policy).
- 3540 • (U//FOUO) Integrity and timestamp features to avert malicious attacks
- 3541 • (U//FOUO) Ability to select and reason about various enterprise alternatives (e.g.,
- 3542 alternate routing for higher QoP, imposed digital rights to limit risk, automatic
- 3543 sanitization options, nearby neighbors with sufficient access) via comparison and “what
- 3544 if” scenarios

3545 (U//FOUO) Finally, extraordinary operations support requires that DACP be able to handle
 3546 policy exceptions that are able to authorize normally disallowed actions due to an extremely
 3547 urgent operational need. Most likely, such an authorization would be tightly constrained by time
 3548 controls (very limited access period) and additional access/distribution controls (very minimal
 3549 set of well-defined actions) to limit risk.

3550 **2.2.3.3.1.1 (U) Technical details**

3551 (U//FOUO) DACP must be able to reach a decision based on risk computation, operational need
 3552 computation, and policy input. Final decision logic uses digital policy to compare risk and need
 3553 computations against acceptable thresholds, specify a decision, and generate a corresponding
 3554 access token of some sort, generate a decision rationale, and generate an audit record.

3555 (U//FOUO) The DACP language features must support conflict detection and resolution,
 3556 negotiation across RAdAC domains/COIs, dynamic update, ontology specification, and human
 3557 readability. Policy must be able to be securely updated, revoked, and enforced within acceptable
 3558 performance margins to ensure currency with dynamic enterprise policy.

3559 (U//FOUO) RAdAC must have a grammar that can succinctly express decision rationale that is
3560 unequivocally tied to the input received, including the ability to list limiting factors (LIMFACs)
3561 in both a machine-understandable and human-readable format.

3562 (U//FOUO) While the ability to discover and select an alternate COA is a highly desirable
3563 feature of a RAdAC-enabled system, embedment of this capability within the core RAdAC
3564 model would severely impact the performance of the access decision process. Rather than embed
3565 this functionality within RAdAC, the preferred approach is to have an offload capability to a
3566 separate service to perform this analysis and make recommendations. Similar digital access
3567 control policy can be used by this ACOA service to reason about alternatives it considers, and
3568 this ACOA service may optionally provide a user interface for the User to select between
3569 [possibly less desirable] alternate COAs.

3570 (U//FOUO) The ability to handle temporal exceptions for extraordinary operations via
3571 dispensations or work flows is a critical DACP feature to enable RAdAC dynamic operations
3572 support. Certain deontic languages provide this capability in the form of “dispensations” that
3573 augment the DACP based on a compelling temporal need. Other approaches include work flows
3574 to address the specific LIMFACs identified in access control decisions. Regardless of the
3575 technical approach, great care must be taken to constrain where dispensations are allowed and
3576 not allowed within the policy language due to national law or immutable operational policy. For
3577 example, dispensations may be allowed for dissemination of a classified document to a cleared
3578 User, without formal access approval, given compelling operational need but may never be
3579 allowed for an uncleared User. Dispensations may be the most appropriate way for digital policy
3580 to annotate and reason about a commander or supervisor’s consent for a User’s operational need
3581 to know a particular piece of information.

3582 (U//FOUO) The policy must be robust enough offer a low error rate (i.e., meet extremely
3583 stringent false negative and false positive rates). Since RAdAC would be replacing the traditional
3584 Mandatory Access Control model objectively, false positives in particular cannot be tolerated for
3585 risk of information disclosure. Dispensations for exception handling must be constrained in such
3586 a way that guarantees select portions of digital access control policy will comply with national
3587 law.

3588 **2.2.3.3.1.2 (U) Usage considerations**

3589 (U//FOUO) Since DACP forms the primary underpinning of the RAdAC model, its
3590 implementation will require significant analysis and community vetting. It will also require
3591 protections against a wide range of security threats since it will be a likely target of IW attack.

3592 **2.2.3.3.1.2.1 (U) Implementation Issues**

3593 (U//FOUO) Conflicting laws and policies - Established laws and organization security policies
3594 require sufficient clearance, formal access approval, and need to know to establish authorization
3595 for classified information. We need to do an assessment of how RAdAC maps to these
3596 requirements (e.g., does operational need equate to need to know) to determine which laws and
3597 organization policies require amendment

3598 (U//FOUO) Human understandable access control policy - Enterprise managers and certifiers
3599 will want a human-readable format to the Access Control Policy to examine and evaluate its
3600 specifications, but RAdAC will need fast machine-readable versions of the same policy to meet
3601 performance needs

3602 (U//FOUO) Supporting decision rationale - A format/grammar must be developed to express the
3603 rationale for an access control decision and any associated LIMFACs and deciding factors. This
3604 grammar may need to be purposefully limited, though, to avoid disclosing too much information
3605 about the current DACP and how to influence its decisions

3606 (U//FOUO) Minimal acceptable input parameters - Need to do research in defining the minimum
3607 quorum and pedigree of input parameters necessary to make an access decision with bounded
3608 risk. Does this minimal set vary based on the Environment (CONUS versus tactical) or Situation
3609 (exercise versus active engagement)? Are heuristics employed only if the access is not decidable
3610 given the other input parameters, or is it always part of the decision process?

3611 (U//FOUO) IT Component integration - DACP and RAdAC's decision output must be tightly
3612 integrated with the policies that affect the management of the IT Components. This avoids
3613 situations where RAdAC allows access through a given enterprise route but then the enterprise
3614 routes the information over a different path because of other decision metrics. Digital rights
3615 policy enforcement must be tightly integrated with the end user equipment portion of IT
3616 Components so that the rights embedded with the information object are strictly enforced

3617 2.2.3.3.1.2.2 (U) Advantages

3618 (U//FOUO) Supports dynamic operations through update and reasoning about operational need
3619 and security risk for access control decisions

3620 (U//FOUO) Facilitates expression in human understandable format for analysis and update

3621 (U//FOUO) Supports exception handling for extraordinary operations with compelling
3622 operational need

3623 (U//FOUO) Extends beyond the access decision to address soft object life cycle and distribution
3624 controls

3625 2.2.3.3.1.2.3 (U) Risks/Threats/Attacks

3626 (U//FOUO) Spoofing or man-in-the-middle unauthorized modification of policy updates

3627 (U//FOUO) Replay of access control requests or decisions to cause a denial of service

3628 (U//FOUO) Unintentional misconfiguration of DACP can introduce access denial or
3629 confidentiality breaches

3630 (U//FOUO) Exception handling could potentially be misused by insiders to gain access to
3631 unauthorized soft objects (e.g., exaggerating operational need)

3632 **2.2.3.3.1.3 (U) Technology/Standards Analysis**

3633 (U) Specific technologies and standards for digital policy are analyzed in Section 2.4. This
 3634 subsection applies that analysis specifically to the digital policy needs for Policy-Based Access
 3635 Control.

3636 (U//FOUO) XML Access Control Markup Language (XACML) has been pushed within the web
 3637 services and DoD network-centric initiatives and has reached significant maturity as a result, but
 3638 it has some serious limitations for digital access control policy. The largest limitations are its
 3639 present inability to understand ontology and to resolve conflicting policy assertions. A third
 3640 limitation is in the area of dispensations, since they can only be approximated through a policy
 3641 update and policy revocation after a specified period.

3642 (U//FOUO) The standards being developed under the W3C semantic web initiative appear to
 3643 meet the wide range of needs for digital access control policy. They address ontology (via OWL)
 3644 and use deontic logic to capture, reason through, and apply business rules according to
 3645 underlying mathematics. Certain deontic logic technologies such as Rei and KaOS offer the
 3646 ability to create and apply dynamic dispensation rules as well. Though the expressiveness of the
 3647 standards appear sufficient to cover the needs for digital access control policy, further analysis
 3648 needs to be done to extend the deontic logic math model to address specific access control needs,
 3649 verify performance of the technologies, and verify scalability to an enterprise level.

3650 **2.2.4 (U) Distributed Policy Based Access Control: Gap Analysis**

3651 **2.2.4.1 (U) Core RAdAC: Gap Analysis**

3652 (U//FOUO) The Core RAdAC functions are in their infancy with respect to concept formulation,
 3653 standards development, and technology implementation, as shown from a summary level in
 3654 Table 2.2-9. Industry really will not benefit from RAdAC as the Government will, so it is not
 3655 surprising to see little research and development in this area. Industry is showing interest in role-
 3656 based access control and now attribute-based access control, but RAdAC's unique features put it
 3657 on a complementary but dissimilar technology path.

3658 (U//FOUO) The following technology gaps exist for RAdAC:

- 3659 • (U//FOUO) Attribute Based Access Control standard - Although there is research and
 3660 even initial product offerings for ABAC-based products, there is no IETF or Government
 3661 standard. Cisco and Maxware have proprietary products, and Network Associates is
 3662 doing research funded by SPAWAR, but none meets all of the attribute requirements for
 3663 RAdAC. Since we are looking at ABAC as an interim implementation of RAdAC, we
 3664 could employ a proprietary solution while RAdAC is being explored and developed in
 3665 parallel. But we would do so at the potential risk of it becoming the GIG standard if
 3666 RAdAC is not realizable for a presently unknown technical or political reason. Prudence
 3667 dictates that we have an alternate fallback standard in place, given the current immaturity
 3668 of RAdAC and its critical role in the enterprise.

- 3669 • (U//FOUO) Protection Profiles - There are no current or planned protection profiles that
3670 address RAdAC or attribute-based access control. Existing protection profiles are limited
3671 to Orange Book approximations. These protection profiles are necessary to establish the
3672 minimum security protections required for any implementation of RAdAC.

- 3673 • (U//FOUO) RAdAC standard - Since industry is not moving in the RAdAC direction,
3674 there are no formal representations of architecture, interface definitions, performance
3675 requirements, or protocol requirements.

- 3676 • (U//FOUO) RAdAC math model - RAdAC needs an underlying math model to meet
3677 medium and high assurance implementation requirements and to assist in the
3678 transformation from a DAC and MAC access control culture. This math model needs to
3679 include the digital access control policy since the two are so tightly integrated. Further
3680 extensions to the deontic logic math model need to be accomplished to apply it
3681 specifically to the access control domain and prove mappings of certain policy constructs
3682 to traditional DAC and MAC access control models.

- 3683 • (U//FOUO) Input parameter ontology - All attributes that feed the RAdAC model need to
3684 have an ontology that is accessible and standardized. This applies to attributes of IT
3685 Components, Environment, Situation, Soft Objects (metadata), and People.

3686

3687

Table 2.2-9: (U) Technology Adequacy for Access Control

This Table is (U)					
		CoreAccess Control	Digital Rights	Access Control Policy	Required Capability (RCD attribute)
Enabler Attributes	Risk & Need Determination		N/A		IAAC4
	Math model		N/A		IAAC4
	Decision logic		N/A		IAAC1, IAAC4, IAAC7
	Ontology	N/A	N/A		IAAC4
	Exception handling		N/A		IAAC5
	Conflict resolution		N/A		IAAC7
	Object Lifecycle				IAAC8
	Protection Profile				IAAC9
This Table is (U)					

3688

2.2.4.2 (U) Assured Metadata: Gap Analysis

3689

(U//FOUO) From an overall prospective, as shown in Table 2.2-10: (U) Technology Adequacy for Metadata, the technology and functionality gaps in the assured metadata area will not require the same levels of technology leaps, or major innovations in comparison to the RAdAC portion of this enabler or other technologies needed in the GIG.

3690

3691

3692

3693 (U//FOUO) For the metadata standards area, both the IC and DoD are working on the definition
3694 of standards to support discovery and marking of information that will be part of the GIG. Both
3695 groups have built their standards implementation/schemas based on a widely proven and
3696 available commercial language/technology (XML and OWL). Further, an initial gap analysis has
3697 been completed which compares the capabilities (IA attributes) needed to support RAdAC style
3698 access control decision making and discovery (See Table 2.2-6: (U) Metadata Gap Analysis)
3699 with these standards. The process of coordinating between these two organizations has been
3700 started to ensure that these required IA attributes are integrated into these implementation
3701 standards. Stability or backward compatibility of these IA attributes from a syntax and semantics
3702 prospective will be critical. If not well planned, changes after final approval will likely ripple to
3703 changes in supporting tools and infrastructure, or could affect large quantities of previously
3704 populated information object metadata records. Finally, prior to stabilizing the metadata
3705 standards and IA attributes, it is strongly recommended that further studies be conducted
3706 examining the impact and potential for optimization regarding the increased of metadata IA
3707 granularity and its potential GIG impact to network traffic/overhead, especially for real-time and
3708 session object types.

3709 (U//FOUO) The development of trusted metadata creation tools can parallel the metadata
3710 standards in initial design. However, final development, integration, and testing will be
3711 dependant on a stable and accepted metadata standard(s) with required IA attributes. There have
3712 been successful implementations and commercial products that provide metadata creation tools
3713 based on the XML web publishing, Dublin Core metadata standard, and have been applied to
3714 their communities' metadata standard and creation needs in the commercial environment.
3715 However, the IA related capabilities and interfaces as defined in section 2.2.3.2.2.1 are new,
3716 complex, and unique to this GIG implementation.

3717 (U//FOUO) In the area of cryptographic binding of metadata to its source information,
3718 Cryptographic Message Syntax (RFC2630) is the recommended technology standard. This
3719 syntax standard provides the capability to support selectable digest and signature algorithms. It is
3720 also expandable to support the potential inclusion of other algorithms/standards as technology
3721 progresses. However, like the metadata creation tools, the GIG and IA interface aspects required
3722 of this capability will remain a technical challenge.

3723 (U//FOUO) NOTE: The metadata IA attributes analysis is currently focused on the various
3724 forms of GIG information objects. IA metadata attributes unique to service objects and their
3725 supporting tools are currently in work and will be addressed in the next release of the technology
3726 roadmap.

3727

Table 2.2-10: (U) Technology Adequacy for Metadata

This Table is (U)					
		Metadata Standards/ Language	Metadata Creation Tools	Metadata Cryptographic Binding	Required Capability (RCD attribute)
Enabler Attributes	Commercial Standards Based				
	GIG or IA assurance (unique) interfaces provided				IAIL1, IAIL2, IAIL3, IAIL4, IAIL6, IAIL13, IAIL16
	GIG Governance (Standards) Bodies in Place			N/A	IAIL5, IAIL10, IAIL12, IAIL16, IAIL17
	Need to Share/Control Granularity supported			N/A	IAIL9, IAIL10, IAIL12, IAIL14
	RAdAC value and Modifier Construct supported			N/A	IAIL9
	IA Attribute Internal Consistency (syntax Checking)			N/A	IAIL9, IAIL10, IAIL20
	Cryptographic Performance up to Type 1 Assurance	N/A	N/A		IAIL15
This Table is (U)					

3728

3729 **2.2.4.3 (U) Digital Access Control Policy: Gap Analysis**

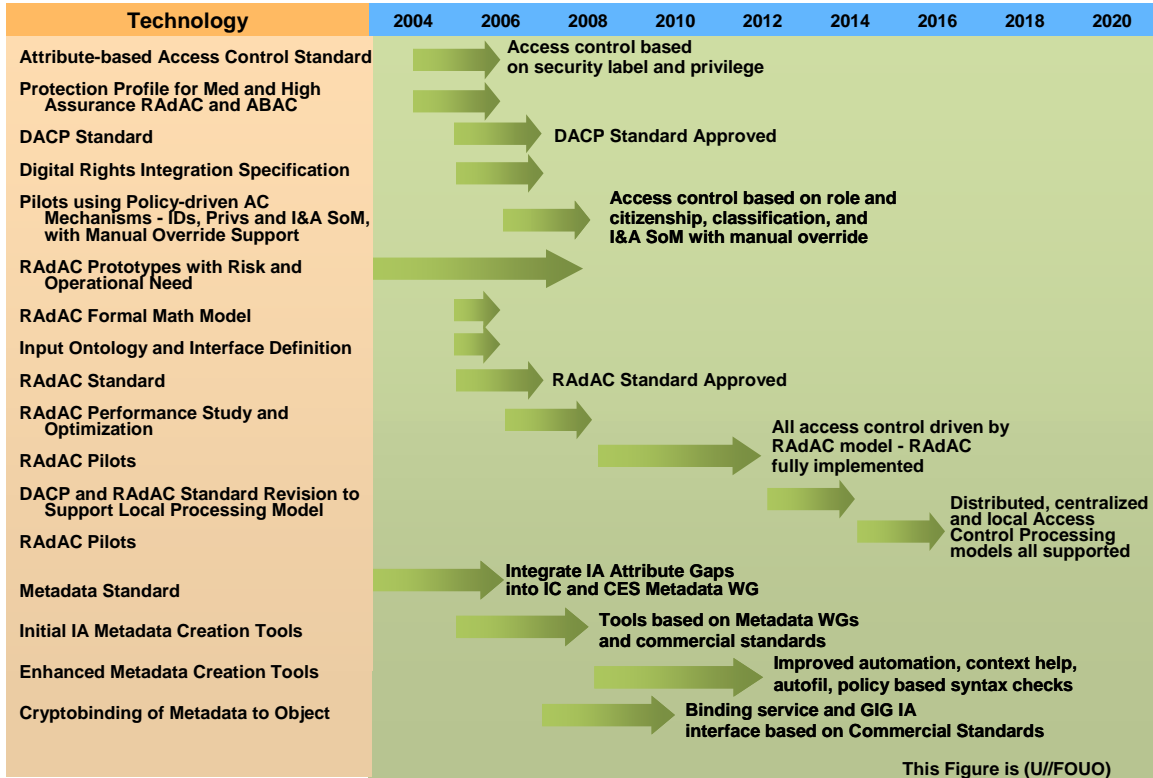
3730 (U//FOUO) The proposed OWL v2.0 standard for ontology and the deontic language
3731 implementations Rei and KaOS appear to meet the expressiveness required for digital access
3732 control policy, but there is significant work needed to realize a complete implementation that
3733 will meet GIG information-sharing requirements. The following list describes the major gaps.

- 3734 • (U) DACP standard. A digital access control policy standard that uses ontology and
3735 deontic languages needs to be developed based on the underlying math model. This
3736 standard will address the access control policy grammar, exception handling, business
3737 rules about allowable and disallowable policy constructs, and business rules for policy
3738 negotiation and deconfliction.

- 3739 • (U) Digital Rights Management integration specification. Digital Rights can be viewed as
3740 a static projection of digital access control policy onto a particular soft object. There is
3741 currently ongoing research in the Digital Rights realm and proposed standards, but none
3742 of them specify a relationship to digital access control policy. An analysis of their
3743 relationships, digital rights implementation (XrML or otherwise), and Policy
3744 Enforcement Point interface is necessary to complete the end-to-end access control of
3745 GIG information and support the transition to a “need-to-share” culture.

- 3746 **2.2.5 (U) Policy Based Access Control: Recommendations and Timelines**
 3747 (U//FOUO) The following is a list of prioritized distributed policy-based access control gap
 3748 closure recommendations or actions. They are listed from highest to lowest priority.
- 3749 • (U//FOUO) Develop Attribute-Based Access Control standard
 - 3750 • (U//FOUO) Develop ABAC and RAdAC Protection Profiles
 - 3751 • (U//FOUO) Develop RAdAC standard
 - 3752 • (U//FOUO) Develop RAdAC math model
 - 3753 • (U//FOUO) Conduct RAdAC prototyping for requirements discovery. This activity feeds
 3754 input ontology development, RAdAC standard development, DACP standard
 3755 development, and Digital Rights integration specification
 - 3756 • (U//FOUO) Work with IC and CES Metadata working groups to integrated IA attributes
 3757 into a standard in accordance with detailed analysis, or (preferred) support the merge of
 3758 these standards, and ensure IA RAdAC required attributes are included
 - 3759 • (U//FOUO) Begin early design of metadata creation tools in parallel with metadata
 3760 standards definition to ensure IA specific attributes and authorization interface needs are
 3761 addressed
 - 3762 • (U//FOUO) Develop input parameter ontology
 - 3763 • (U//FOUO) Conduct study on RAdAC performance and optimization techniques
 - 3764 • (U//FOUO) Conduct RAdAC pilot program to test fielding and operational issues
 - 3765 • (U//FOUO) Develop DACP standard with associated business rules
 - 3766 • (U//FOUO) Develop Digital Rights integration specification
 - 3767 • (U//FOUO) Conduct study on impact and potential for optimization of metadata IA
 3768 granularity related to GIG network traffic/overhead
 - 3769 • (U//FOUO) Continue work of defining the GIG services metadata tagging capabilities
 3770 potential technologies

3771 (U//FOUO) Figure 2.2-1 summarizes timeframes for these closure recommendations.



3772

3773

Figure 2.2-4: (U) Policy-Based Access Control Gap Closure Timelines

3774 **2.3 (U) PROTECTION OF USER INFORMATION**

3775 ((U//FOUO) Protection of user information provides the protection of data-at-rest and data-in-
3776 transit from end-entity to end-entity. For applications based on the client-server model common
3777 to much of today's networks, this GIG vision would provide integrity, confidentiality, and other
3778 required security services in both directions between the originating client and the responding
3779 server. For peer-to-peer-based applications, this provides those same services between the
3780 corresponding peers. For applications-based on other models, appropriate security services will
3781 be applied.

3782 (U//FOUO) End-to-end protection of user information does not always mean security services
3783 are provided between the true endpoints of communication. There is always a trade-off to be
3784 made. For example, if end-to-end confidentiality is provided, that implies that the information is
3785 encrypted between the requesting client and the responding server. That means that GIG-
3786 provided or organization-provided infrastructure devices such as intrusion detection systems and
3787 firewalls cannot examine the data as it passes. This makes it difficult to detect and stop malicious
3788 code such as viruses or worms, it makes it difficult to perform content-based filtering (e.g., Spam
3789 checking), and it makes it more difficult to detect and stop intrusions. In this scenario, the client
3790 node itself must provide all security. This may not be feasible for commercial operating systems
3791 and products—even in the 2020 time frame—and it may make it very difficult to detect attacks
3792 from authorized GIG insiders.

3793 (U//FOUO) Even within single devices, end-to-end protection of user information may have
3794 different meanings depending on the specific application or organization. For example, multiple
3795 users or user identifiers may share a single end-point (e.g., multiple users may share a client
3796 node, and multiple services may share a single server). End-to-end communications security in
3797 this context may mean client-to-server security or it may mean end-user-to-server-identifier
3798 security.

3799 (U//FOUO) Thus, depending on the enterprise and U.S. Government policy, different
3800 applications may have end-to-end security between clients and servers or communicating peers;
3801 or they may have end-to-end security between organizational enclaves; or between other points.
3802 These situations are entirely consistent with the GIG Vision.

3803 (U//FOUO) However, there is much work to be done before this vision can be accomplished.
3804 The current environment includes many systems operating in different domains and at different
3805 security levels. Communication and interoperation among these domains and across these
3806 different security levels is not always possible. True end-to-end secure communications cannot
3807 be provided in the current or near-term GIG.

3808 (U//FOUO) For the current and near-term GIG implementations, Cross-Domain Solutions
3809 provides the necessary secure interoperation. Applications and communications must be secured
3810 within a single security level—within a domain. Then, interactions between domains are allowed
3811 by using cross-domain solutions (e.g., guards, gateways and firewalls, and specific routing
3812 techniques).

3813 (U//FOUO) As the GIG evolves from the current capability set to the vision system, this will
3814 gradually change. As core systems are fielded that can allow the merger of domains and
3815 supporting multiple classification levels in a system, less emphasis will be placed on such cross-
3816 domain solutions as guards and content filters. More emphasis will be placed on security at the
3817 end-points, whether those end-points are enclave boundaries or client nodes themselves.

3818 **2.3.1 (U) GIG Benefits Due to Protection of User Information**

3819 (U//FOUO) The Information Assurance constructs used to support Protection of User
3820 Information provide the following services to the GIG:

- 3821 • (U//FOUO) Protects information in accordance with enterprise-wide policy and the data
3822 owner's specified Quality of Protection (QoP)
- 3823 • (U//FOUO) Allows multiple users to use a single workstation so a user can walk up to a
3824 client and access their information
- 3825 • (U//FOUO) Allows access to multiple levels of information on the same platform without
3826 compromising that information (i.e., trusted hardware/software platforms)
- 3827 • (U//FOUO) Protects against the analysis of network protocol information, traffic volume,
3828 and covert channels
- 3829 • (U//FOUO) Provides user-to-user protection of secure voice traffic from speaker to
3830 listener.

3831 **2.3.2 (U) Protection of User Information: Description**

3832 (U//FOUO) Protection of user information provides the protection of data objects at rest and the
3833 protection of data-in-transit. Data-at-rest protection is the protection of data objects while they
3834 are stored in repositories across the GIG and within a client's local environment. Data-in-transit
3835 protection is the protection of information flows as they move across the GIG within all levels of
3836 the transmission protocol stack, including application, network, and link level.

3837 (U//FOUO) Protection of User Information also includes the concept of the GIG Black Core. The
3838 Black Core is the packet-based portion of the GIG, where packet level protections are provided
3839 between end entities. Over time, end entities providing packet level protections move from the
3840 network boundaries to the enclave boundaries to the end clients. Circuits within the GIG will be
3841 protected with circuit encryption. In addition, some high risk links may need additional
3842 protections if the risk of traffic analysis or other threat is exceptionally high. Possible solutions
3843 include link encryption and TRANSEC.

3844 (U//FOUO) Classified information will be protected using high assurance (Type 1) mechanisms,
3845 while unclassified information will be protected using evaluated commercial mechanisms. To
3846 support the end state capability to enable users to access the proper information, encryption
3847 boundaries must be able to support both Type 1 and commercial mechanisms. Encryption
3848 products must also have access to the proper key material to protect all classifications of
3849 information.

3850 (U//FOUO) The protection of user information must support large numbers of dynamic
3851 communities of interest (COI). Support for COIs does not necessarily imply encrypted tunnels
3852 between COI members. COIs can also be accomplished through other mechanisms, such as
3853 filtering (e.g., Access Control Lists [ACLs]), or logical separation (e.g., Multi Protocol Label
3854 Switching [MPLS] Labeled Switch Path [LSPs]). Sufficient auditing mechanisms are necessary
3855 to track the establishment and termination of COIs.

3856 (U//FOUO) To support connectivity between GIG networks and coalition networks, mechanisms
3857 are necessary that allow information flows to pass between coalition partners and the GIG. Each
3858 coalition network will be different and require different security mechanisms and procedures.
3859 Some coalition networks will be owned and operated by the U.S. with partners using resources.
3860 Other will be owned and operated by allies with U.S. users. Still others will be owned and
3861 managed by a number of different allies all intended to seamlessly interconnect. These
3862 mechanisms are enforced in a construct referred to as a trust manager.

3863 (U//FOUO) Trust managers enforce policy for connections to coalition partners and allow or
3864 disallow individual connections between GIG users and coalition partners. Trust managers can
3865 filter traffic types, allow or disallow specific users, monitor information flows, or enforce any
3866 other policy required for coalition connections.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

3867 (U//FOUO) Whenever GIG systems interact with coalition or an ally's resources, both sides of
3868 the connection will have security mechanisms in place. While the GIG will be able to control
3869 policy on the GIG side of the connection, the coalition partner will set policy for the other side of
3870 the connection. This compounds the problem of information sharing with coalition partners. This
3871 is similar to the issues with sharing information across GIG systems, as both policies must be
3872 coordinated. Information shared with coalition partners could include all types of data objects
3873 and data formats, including data files, messaging, video, streaming video, voice, and web traffic.

3874 (U//FOUO) The GIG will require that clients and computing platforms (i.e., hardware and
3875 software) have more inherent trust than they do today. Devices directly accessible by users—
3876 running a variety of user applications and connected to untrusted networks—tend to be the least
3877 trustworthy devices in the network. They are ripe targets for malicious code attacks and mis-
3878 configuration. However, the GIG will rely on clients to do a variety of security-critical functions
3879 (e.g., maintain domain separation when accessing information at various levels of sensitivity,
3880 support authentication of a user to the infrastructure, support authentication of a client to the
3881 infrastructure, properly label data, enforce local security policy, properly encrypt data).

3882 (U//FOUO) In today's system high environments (i.e., JWICS, SIPRNet), less trust in clients is
3883 required since all users within an environment have an equivalent level of trust. While placing
3884 trust in clients today may seem unreasonable, the GIG Vision requires that procedures and
3885 mechanisms be in place to allow clients to perform critical security functions. A higher level of
3886 trust within clients is especially important as coalition users and networks are connected to the
3887 GIG and as today's system high boundaries are eliminated. A higher level of trust is required for
3888 all devices in the GIG— not just end user clients. All devices in the GIG will be required to
3889 perform security related functions, and there must be a sufficient degree of trust in these devices
3890 for them to reasonably execute their functions.

3891 (U//FOUO) The GIG, however, will consist of IT devices (i.e., routers, servers, clients) with
3892 varying levels of trust. The GIG will use a concept referred to as Quality of Protection for data
3893 objects. As part of the data labeling, an object will be associated with security properties and
3894 policies necessary for protecting the object. Properties can include:

- 3895 • (U//FOUO) How to protect the object as it travels across the network (e.g., commercial
3896 grade vs. Type 1 protections, object and/or packet or link level data-in-transit protection
3897 requirements)
- 3898 • (U//FOUO) How the data object can be routed (e.g., must be contained within the GIG,
3899 can flow to or through networks external to the GIG, such as coalition networks or the
3900 Internet)
- 3901 • (U//FOUO) How the data object must be protected while at rest. QoP is different from
3902 metadata that describes the contents of the object.

3903 (U//FOUO) Metadata is designed to enable discovery and data sharing. QoP defines how a data
3904 object is protected while it is at rest and in transit. When QoP is defined, it should not reveal
3905 attributes related to the data originator or client. Policy-Based Access Control will provide the
3906 enforcement mechanisms to assure the specified QoP is provided.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

3907 (U//FOUO) Data-at-rest protection will be required for some types of data (e.g., for extremely
3908 sensitive information) and for certain environments (e.g., information stored on a local client
3909 within a hostile environment). The requirements for data-at-rest protection will be identified
3910 through a protection policy, such as within a data object or client's protection policy. For shared
3911 information, data-at-rest protection must be provided at the object level, where an object is
3912 defined as a file or pieces of a file, such as paragraphs. This leads to a large range of object
3913 types.

3914 (U//FOUO) All data objects must be protected properly. GIG users must be able to discover and
3915 access objects. This will require a key management infrastructure that can dynamically deliver
3916 the key material to access objects requested by the user. Data-at-rest for local clients can be
3917 provided in a number of ways, including media encryption mechanisms.

3918 (U//FOUO) Protection of data-in-transit consists of the ability to provide confidentiality,
3919 integrity, and authentication services to information as it is transmitted within the GIG. The QoP
3920 information will describe the services needed for any specific data object.

3921 (U//FOUO) Protection of data-in-transit includes providing traffic flow security (TFS). TFS
3922 should be provided for all high-risk links in the GIG but could also be provided for medium or
3923 low-risk links. In general, TFS protections include mechanisms that protect against network
3924 mapping and traffic analysis. In general, the lower in the protocol stack confidentiality is applied,
3925 the greater the TFS benefit. For circuits, end-to-end circuit encryption provides traffic flow
3926 security. For IP networks a variety of mechanisms can be used. For IP environments where the
3927 communications links are circuit based and the routers are protected one option could be hop-by-
3928 hop link encryption applied to the communications links to provide traffic flow security for
3929 encrypted packet traffic. TFS mechanisms, however, have a performance impact and should be
3930 carefully matched against the risk for the information flow.

3931 (U//FOUO) Protection of data-in-transit also includes the ability to prevent unauthorized
3932 transmission of data within the GIG. A covert channel is an unauthorized information flow that is
3933 precluded by the network's security policies. Covert channels must be eliminated to permit
3934 global access of information required within the GIG.

3935 (U//FOUO) Network layer data-in-transit security is the protection of IP packets as they flow
3936 across the GIG. Protection could be from enclave to enclave to enclave, or from host to host.
3937 High Assurance Internet Protocol Encryptor (HAIZE)-compliant devices will be used to provide
3938 Type 1 data-in-transit network layer security for the GIG. At a minimum, Unclassified data will
3939 be protected using medium robustness Type 3 solutions.

3940 (U//FOUO) Speech traffic (Voice over IP [VoIP]) within the GIG can be protected at the
3941 Network Layer. Currently, HAIZE can only provide enclave-to-enclave protection. In the future,
3942 when HAIZE is integrated into end-systems, the protection can be migrated from the enclave
3943 level back to the user level. This functionality will require the development of a new mode
3944 within the HAIZE standard to meet the real-time performance requirements of a Voice over
3945 Secure IP (VoSIP) terminal.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

3946 (U//FOUO) Media gateways can also be defined to extend speech capability beyond the GIG to
3947 legacy circuit-based systems, although Network Layer security is not effective beyond such a
3948 gateway. Therefore, using HAIPE to protect speech traffic would require Red gateways to legacy
3949 circuit-switched networks. The appropriate security (e.g. Future Narrow Band Digital Terminal
3950 [FNBDT]) would have to be applied on the circuit-switched side of the gateway to protect the
3951 speech traffic over a legacy network.

3952 (U//FOUO) Application Layer data-in-transit security is the protection of information as it flows
3953 from one end user terminal to another, where the end user terminals apply the protection
3954 mechanisms and the protected information is not accessible at any point during transmission.

3955 (U//FOUO) Within the GIG, most speech traffic is carried across circuit switched networks.
3956 Speech traffic in circuit switched networks is protected at the application layer using Secure
3957 Terminal Unit-Third Generation (STU-III) or Secure Terminal Equipment (STE) products. STU
3958 and STE products provide application layer speaker to listener security. Future secure voice
3959 products and architectures must consider interoperability with existing secure voice products
3960 (e.g., secure voice products used by NATO, tactical secure voice products.)

3961 (U//FOUO) Application layer protection of speech traffic (VoIP) within the GIG could also be
3962 accomplished through development of secure VoIP terminals. Interoperability of secure VoIP
3963 terminals will require a common implementation of FNBDT over IP. Secure VoIP terminals will
3964 provide end-to-end, Multiple Single Levels of security across the Black Core. That is, although
3965 only one session is permitted on each end terminal at a time, subsequent sessions can be
3966 established at different security levels.

3967 (U//FOUO) Secure VoIP terminals can be placed on the Black Core to provide end-to-end,
3968 Application Layer security across the Black Core. VoIP gateways can also be developed to
3969 provide interoperability with legacy FNBDT products on the Public Switched Telephone
3970 Network (PSTN). Such a gateway requires access to the IP network on one side and access to
3971 appropriate circuit-based networks on the other. The gateway then provides interworking
3972 between the IP protocol stack and a circuit-based modem. There are some issues (e.g.,
3973 transcoding in the gateway needs to be disabled), but these issues can be resolved to provide a
3974 Black gateway solution for the FNBDT Application Layer security approach.

3975 (U//FOUO) Secure VoIP terminals can also be placed in Red enclaves to provide user-to-user
3976 security, whereas HAIPEs fronting the enclaves only provide enclave-to-enclave security.
3977 FNBDT can be overlaid on HAIPE to provide this user-to-user level of security.

3978 (U//FOUO) Overlaying FNBDT on top of HAIPE provides several benefits. First, it provides
3979 confidentiality of user voice traffic within the enclave. Second, it allows the security level of the
3980 voice session to be based on the clearances of the users rather than the security level of the Red
3981 enclave. Finally, it enables interoperability between phones attached to networks at different
3982 security levels (cross-domain solutions).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

3983 (U//FOUO) Communications between two secure VoIP terminals in different enclaves, where
3984 the two enclaves are in the same security domain, is relatively straightforward. The HAIPE
3985 fronting the two enclaves perform network level encryption between the enclaves, and the Secure
3986 VoIP phones attached to the Red enclave networks perform FNBDT application level encryption
3987 between the two users. In this scenario, the users are not restricted to a conversation at the
3988 security level of the Red enclave networks. For example, two users with Top Secret clearances
3989 could hold a Top Secret conversation on phones attached to secret level enclave networks. Note
3990 this scenario utilizes Red enclave call control (call control in the security domain of the Red
3991 enclaves).

3992 (U//FOUO) From the above examples, it can be seen that there are potentially multiple domains
3993 of call control. A single user and associated secure VoIP terminal could potentially use multiple
3994 call control domains. The call control domain used for an instance of communications would be
3995 based on the security domains of the networks where the local and remote users' secure VoIP
3996 terminals are attached.

3997 (U//FOUO) Data-in-transit protection can also be applied to the GIG at protocol stack layers
3998 other than Network and Application. This protection may be in place of or in addition to security
3999 at other layers. Specifically, many individual links within the GIG may require protection
4000 appropriate for the Physical Layer, such as transmission security (TRANSEC). Security at this
4001 layer provides protection that cannot be obtained at other layers, including:

- 4002 • (U//FOUO) Anti-Jam (A/J)
- 4003 • (U//FOUO) Low Probability of Interception/Detection (LPI/LPD)
- 4004 • (U//FOUO) Traffic Flow Security (TFS) and Traffic Analysis Protection
- 4005 • (U//FOUO) Signals Analysis Protection
- 4006 • (U//FOUO) Protocol and Header Cover/Packet Masking
- 4007 • (U//FOUO) TRANSEC Isolation for Major Sets of Users

4008 **2.3.3 (U) Protection of User Information: Technologies**

4009 (U//FOUO) The technologies in this enabler are organized into technologies that provide data-at-
4010 rest protection, data-in-transit protection, trusted platforms, trusted applications, Cross Domain
4011 Solutions, and non-repudiation. The data-in-transit protection technologies are further organized
4012 by protocols layers. Non-repudiation and Cross Domain Solutions are broken out separately
4013 because they do not fit cleanly into either data-at-rest or data-in-transit.

4014 **2.3.3.1 (U) Technologies for Protecting Data-at-Rest**

4015 *(U) EDITOR'S NOTE: MATERIAL ON PROTECTING DATA-AT-REST WILL BE ADDED IN A FUTURE*
4016 *RELEASE. SECTIONS ARE PROVIDED BELOW THAT REFLECT THE TYPE OF CONTENT PLANNED.*

4017 **2.3.3.1.1 (U) Cryptography**

4018 (U) There are several applications of cryptography for protecting data at rest including
4019 encryption, signing/authentication, binding, and integrity checking. Cryptographic capability
4020 may reside in dedicated security devices or be provided within the host itself.

4021 **2.3.3.1.1.1 (U) Storage Networks and Networked Storage Operations**

4022 (U) There is an increasing trend towards the use of storage networks to share storage resources
4023 (data and/or capacity) or to provide geographic distribution of storage assets for increased
4024 availability and survivability. Network Attached Storage (NAS) and Storage Area Networks
4025 (SAN) are the two primary approaches. SANs introduce or exacerbate security problems due to
4026 the following:

- 4027 • (U) A very large amount of information may be contained within one system
- 4028 • (U) Storage resources may need to be shared between domains or enclaves
- 4029 • (U) Storage assets may be directly accessible from the network including the WAN
- 4030 • (U) The storage network management infrastructure needs protection
- 4031 • (U) Access enforcement is remote from data owners/producers and data users
- 4032 • (U) Possible distribution of storage elements over large distances.

4033 (U) A NAS provides file storage using Network File System (NFS) or Common Internet File
4034 System (CIFS) over TCP/IP. A SAN provides virtual disk volume storage using a Small
4035 Computer Systems Interface (SCSI) family protocol. IP-based storage protocols are being
4036 developed and implemented. Elements of a storage network include storage arrays, switches,
4037 host bus adapters/hosts, and security devices.

4038 (U) Whether storage networks are used or not, there are also existing storage operations across
4039 the GIG networks that have similar security concerns. These include replication of data among
4040 distributed sites, distributed data stores, backup and restorable operations between sites, and
4041 archives of data to remote sites.

4042 (U//FOUO) In general, security standards and specifications for network storage are less mature
4043 than those for communications security. There are no common definitions of security services
4044 across vendors. Across security services vendors, common definitions are lacking and a
4045 corresponding shortfall in security products and security features for storage devices exist.

4046 **2.3.3.1.2 (U) Data Backup & Archive**

4047 **2.3.3.1.3 (U) Data Destruction**

4048 **2.3.3.1.4 (U) Labeling**

4049 **2.3.3.1.5 (U) Periods Processing**

4050 **2.3.3.1.6 (U) Physical Controls**

4051 **2.3.3.1.7 (U) Quality of Protection**

4052 (U//FOUO) Quality of Protection is a ranked set of end-to-end protection properties of a system
4053 that collectively describe how resources will be protected within that system. These properties
4054 may include network infrastructure characteristics, client IT characteristics and the cryptographic
4055 capabilities of the IT and network components. A resource will not be made available to a user
4056 unless the resource protection requirements can be met by the QoP level of the system or another
4057 policy supersedes these requirements. The QoP of the system is not typically one fixed level but
4058 is instead a range of available capabilities that can be utilized by the component enforcing the
4059 resource protection requirements. For example, in routing a packet, a path that meets the
4060 packet's resource protection requirements is utilized if available and if in accordance with QoS
4061 and other applicable policy. For data-at-rest, the QoP includes such topics as controls for
4062 copying the data, moving the data, and printing.

4063 **2.3.3.2 (U) Technologies for Protecting Data-in-Transit**

4064 **2.3.3.2.1 (U) Application Layer Technologies**

4065 (U) Application Layer security technologies typically secure primary user data and may also
4066 secure aspects of the application protocols themselves. Application Layer security can provide
4067 protection of user data while in transit, and in some case while stored. These security
4068 technologies do not generally provide protection against traffic analysis, or attacks on lower
4069 layer protocols (e.g., IP).

4070 (U) Application security technologies are characteristically different for real-time applications
4071 than for non-real-time applications. Real-time applications include technologies such as
4072 streaming audio and video. Non-real-time applications include such technologies as email, web
4073 browsing and web services.

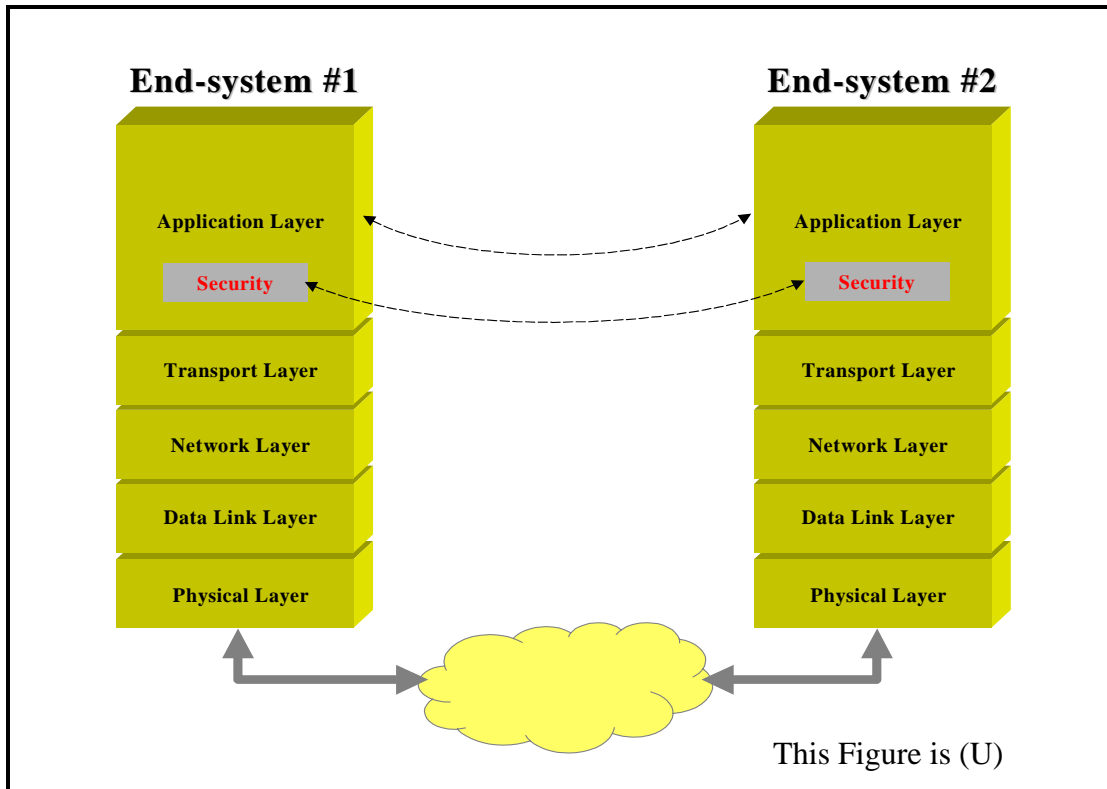
4074 **2.3.3.2.1.1 (U) Non-Real-Time Data Technologies**

4075 (U) Three basic classes of technology are used to provide non-real-time application security.
4076 These consist of the following:

- 4077 • (U) Traditional Layered Application Security Technologies
- 4078 • (U) Session Security Technologies
- 4079 • (U) Web Services Security Technologies

4080 (U) Security technologies for applications that operate in non real-time apply a wide spectrum of
4081 techniques to the problem of securing primary user data end-to-end. Such technologies generally
4082 provide a generic framework for using basic security mechanisms—such as cryptography, one-
4083 way functions, and security protocols—to potentially provide abstract security services within
4084 the context of a particular type of information exchange between cooperating applications.
4085 Figure 2.3-1 shows this relationship.

4086 (U) Nearly all non-real-time applications interface to the layer below them in a connection-
4087 oriented manner, making the dialog between the applications subject to security concerns.
4088 Generally, security will be provided by a sub-layer that operates below the application and
4089 applies security mechanisms to the communication. In the Application Layer, such security
4090 functionality is usually modeled as a discrete functional object rather than a sub-layer because
4091 same security mechanisms might be applied in different ways to different applications, leading to
4092 layering inconsistencies. Some security objects are generic, and can offer service to multiple
4093 applications. Others are tightly coupled to or embedded in the applications that they serve. Like
4094 the applications themselves, security objects exchange protocols with peer objects.



4095

4096

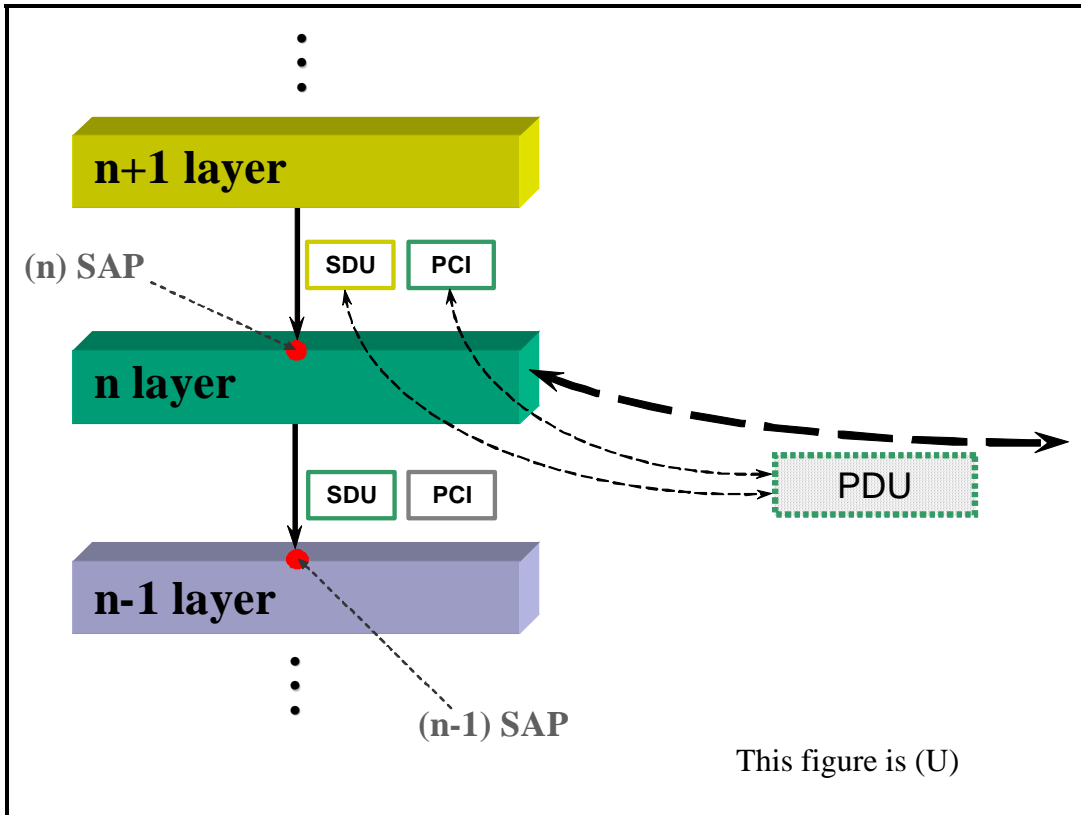
Figure 2.3-1: (U) Context of Non Real-Time Application Security⁴

4097 (U) Application security tailors the application of security techniques to the specific needs of the
 4098 application. This means that the security object can selectively apply security techniques
 4099 differently to discrete fields or messages exchanged with the peer. Security mechanisms can be
 4100 applied selectively to specific fields, using different keys for different fields, to achieve different
 4101 services. This is superior in many regards, such as better accommodating Cross Domain
 4102 Solutions (CDS) by selectively leaving parts of the application data readable—or even
 4103 unprotected—for use by CDS boundary protection devices.

4104 (U) The concept of layered communications entails each layer operating semi-autonomously and
 4105 adding its own additional protocol wrapper or control information to the data of the layer above
 4106 it. Figure 2.3-2 illustrates this concept. The layer in question (termed the “n” layer) provides
 4107 service to the layer above (termed the “n+1” layer since it is one layer higher), and receives
 4108 service from the layer below (termed the “n-1” layer). Service is provided at the Service Access
 4109 Point (SAP) for layer “n” also termed the (n)SAP. To request service from the “n” layer, the
 4110 “n+1” layer conceptually submits a request at the (n)SAP along with an Interface Data Unit
 4111 (IDU) to support the request. The IDU consists of a Service Data Unit (SDU) (i.e., payload data
 4112 from the “n+1” layer) and Protocol Control Information (PCI) associated with the requested “n”
 4113 layer services.

⁴ (U) Note that this figure uses the more commonly used OSI terminology for the layers, but omits the Presentation and Session layers as in the Internet model because comparatively few applications in use today employ these layers.

4114 (U) A concrete example of this is an Application Programming Interface, which typically
 4115 consists of a calling address (analogous to the (n)SAP) and a convention for passing parameters
 4116 (analogous to the SDU and PCI). The SDU and PCI passed to the “n” layer are used to formulate
 4117 the SDU that is passed to the “n-1” layer, and thus the virtual Protocol Data Unit (PDU)
 4118 exchanged with the “n” layer of a communicating peer end-system. Security sub-layers or
 4119 objects continue to follow this layered communication model. Security objects provide service to
 4120 the application above, encapsulate the incoming SDU from the “n+1” layer as part of the SDU
 4121 that is passed down to the “n-1” layer, and incorporate some of the supplied PCI in the SDU and
 4122 PCI that are passed down.



4123

4124

Figure 2.3-2: (U) Layered Protocol Wrapping Concept

4125 (U) Engineering application security entails working through trade-offs among different choices
4126 of mechanisms used to provide the desired protection. Application security usually contains
4127 embedded use of cryptography, one-way functions, and security protocols. Cryptography is used
4128 to render selected portions of the application data unreadable to any entities not possessing the
4129 proper key material. One-way functions are a class of mathematical operations that are
4130 elementary to perform, but prohibitively difficult to reverse. They are often used to embed
4131 irreversibility in application security operations. Security protocols are the backbone of
4132 application security. They define the data structures (i.e., what to send) and dialogs (i.e., when to
4133 send it) used to exchange information between the application security peer entities. Protocols
4134 may resemble a simple one-way exchange or a complex conversation replete with security
4135 handshakes. Security protocol design is crucial because most abstract security services (e.g.,
4136 integrity, authentication) are not possible except in a specific protocol context. Design of the
4137 application security usually relies equally on all three of these types of mechanisms as part of an
4138 overall open system security solution. Cryptography alone is not enough as a bad security
4139 protocol can hamper or compromise good cryptography.

4140 2.3.3.2.1.1.1 (U) Traditional Application Security Technologies

4141 (U) Most development to date of application security has focused on so-called traditional layered
4142 technologies. These are characterized by implementation of a standardized security element in
4143 the application layer with a strong relationship to and binding with the target application. Such
4144 technology has been applied to many applications including message handling or electronic mail,
4145 web hypertext, and file transfer. Development of security elements in this manner represents the
4146 old school of application security because doing so can require many years of standardization,
4147 implementation, and testing to realize workable secure solutions. However, considerable
4148 development of traditional application security has already taken place, and it can be leveraged
4149 by the GIG.

4150 2.3.3.2.1.1.1.1 (U) Technical Detail

4151 2.3.3.2.1.1.1.1.1 (U) Secure Messaging

4152 (U) Secure messaging is a good example of the evolutionary development of traditional layered
4153 application security technology. Early messaging was based on Simple Mail Transfer Protocol
4154 (SMTP) and (MSGFMT). It offered ASCII-only messages without attachments, security, or other
4155 advanced features. Many implementations of these messaging standards were created including,
4156 most notably, the SENDMAIL implementation which was bundled free with most UNIX
4157 implementations.

4158 (U) The International Telegraph and Telephone Consultative Committee (CCITT)⁵ entered the
4159 scene by developing its X.400 series of recommendations. X.400 aimed to provide a full-
4160 function messaging system. However, the initial version of the X.400 released in 1984 contained
4161 no provision for security features.

⁵ (U) CCITT has since reorganized into the International Telecommunications Union (ITU) Telecommunications Standardization Sector (ITU-T).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

4162 (U) As the U.S. government was becoming interested in X.400, as part of the developing Open
4163 Systems Interconnection (OSI) protocol stack, NSA began development of the Message Security
4164 Protocol (MSP) as part of the Secure Data Network System (SDNS). MSP provided security to
4165 either X.400 or SMTP through the addition of a connectionless security protocol wrapper around
4166 the message content. MSP evolved further as part of the Multilevel Information Systems Security
4167 Initiative (MISSI), and was eventually offered to the Allies as Allied Communications
4168 Publication (ACP) 120 [CSP].

4169 (U) ACP 120 is used in the presently deployed Defense Message System (DMS). The DMS
4170 implementation of ACP 120 works with the FORTEZZA card and the FORTEZZA Certificate
4171 Management Infrastructure (CMI) to provide encryption and digital signature for formal military
4172 messages. When properly used, ACP 120 is capable of providing the following security services:

- 4173 • (U) Proof of Content Origin
- 4174 • (U) Proof of Content Receipt
- 4175 • (U) Content Confidentiality
- 4176 • (U) Content Integrity
- 4177 • (U) Common Security Protocol (CSP) Integrity
- 4178 • (U) Security Labeling
- 4179 • (U) Rules-based Access Control
- 4180 • (U) Secure Mail List Support.

4181 (U) While MSP was being developed and deployed, CCITT was working on their security
4182 solution for X.400. This solution is today primarily described in X.400, X.402, and X.411. The
4183 X.400 security solution potentially offered all of the same services as CSP, but offered too much
4184 flexibility and insufficient definition of necessary embedded security objects to suffice without
4185 additional profiling. With the demise of OSI, X.400 security has never achieved widespread
4186 implementation or deployment and is no longer a major factor in the evolution of secure
4187 messaging.

4188 (U) With the wholesale abandonment of OSI and X.400, emphasis returned to providing security
4189 for SMTP and the recently standardized Multipurpose Internet Mail Extensions (MIME). Work
4190 began on the Privacy Enhanced Mail ([PEM) project within the IETF.

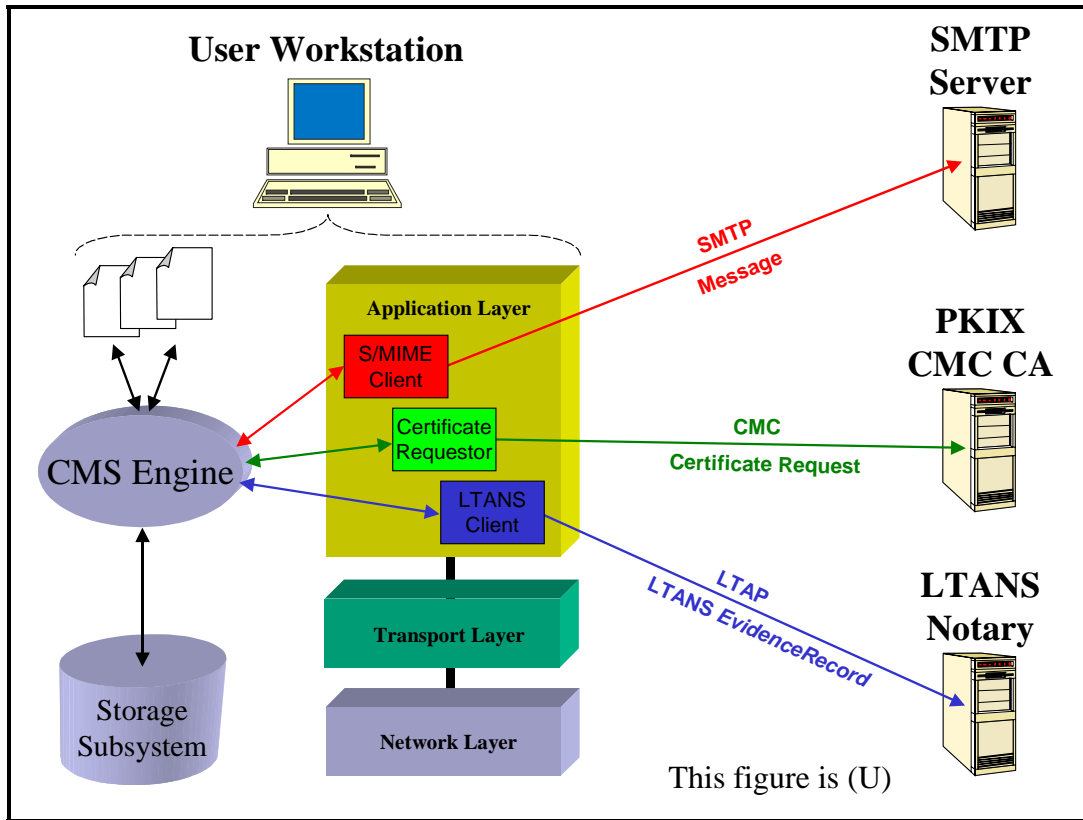
UNCLASSIFIED//FOR OFFICIAL USE ONLY

4191 (U) While PEM ultimately failed⁶, it led to the private development of the Public Key
4192 Cryptographic Standard (PKCS) #7 PKCS7 and the Secure MIME (S/MIME) development by
4193 RSA Data Security Inc. An industry desire to expand the available choices of S/MIME
4194 cryptography and achieve compatibility with MSP led to the development of S/MIME v3 in the
4195 IETF. The S/MIME working group of the IETF has produced several proposed standards of note,
4196 including CMS, MSG, CERT, and ESS. Like MSP, S/MIME v3 provides a wide range of
4197 security services including:

- 4198 • (U) Proof of Content Origin
- 4199 • (U) Proof of Content Receipt
- 4200 • (U) Content Confidentiality
- 4201 • (U) Content Integrity
- 4202 • (U) S/MIME Protocol Integrity
- 4203 • (U) Security Labeling
- 4204 • (U) Secure Mail List Support.

4205 (U) Unlike some application security mechanisms, the specification of the CMS is inherently
4206 designed to be a flexible and reusable module in the S/MIME design. It thereby has the potential
4207 to support other communications or non-communications applications. This arrangement is
4208 illustrated in Figure 2.3-3. This situation already demonstrably exists in that the IETF Public Key
4209 Infrastructure X.509 (PKIX) working group has used CMS as the foundation for its successful
4210 Certificate Management Messages over CMS (CMC) protocol. The IETF Long-Term Archive
4211 and Notary Services (LTANS) working group is planning to similarly use CMS as a foundation
4212 for their EvidenceRecord format. CMS can (and is) similarly used locally for file encryption
4213 outside of the communication stack. The inherent flexibility of this modular style of application
4214 security development has the potential to lead to expedited development of traditional layered
4215 application security elements in the future.

⁶ (U) The failure of PEM had much to do with the conflicting requirements of the changing messaging environment at the time of its development. Interested readers should also see the MIME Object Security Services [MOSS] enhancement of PEM.



This figure is (U)

4216

4217

Figure 2.3-3: (U) CMS Supports S/MIME and Other Secure Applications

4218 (U) Another parallel track of secure messaging evolution is that of the Pretty Good Privacy
 4219 (PGP) development. PGP began as a piece of freeware code for file encryption with public keys.
 4220 Through the introduction of the PGP-MIME specification, it also began to provide application
 4221 security for SMTP/MIME messaging. The OpenPGP working group of the IETF is continuing to
 4222 develop and advance PGP format and PGP-MIME as Internet standards. PGP is not believed to
 4223 be in use within DoD. While not as capable as S/MIME, OpenPGP nevertheless remains a
 4224 competitor in the marketplace. OpenPGP is capable of providing the following security services:

- 4225 • (U) Proof of Content Origin
- 4226 • (U) Content Confidentiality
- 4227 • (U) Content Integrity.

4228 (U) The development and evolution of application security for message handling is a long story
 4229 that is continuing to be written. The widespread use of secure messaging, both in DoD and
 4230 industry will make it an important factor for the GIG for many years.

4231 2.3.3.2.1.1.1.1.2(U) Web Security

4232 (U) Traditional layered application security technology has been applied to provide security for
4233 web browsing with the HyperText Transfer Protocol (HTTP), but have yielded only limited
4234 success. This is not to be confused with Secure Sockets Layer (SSL) which is a different
4235 technology covered in Section 2.3.3.2.1.1.2.1.1. Salient examples of application security for web
4236 browsing include Secure HTTP (S-HTTP) and the IETF Web Distributed Authoring and
4237 Versioning (WebDAV) effort.

4238 (U) The S-HTTP protocol extended the basic HTTP/1.1 protocol to provide mechanisms that can
4239 deliver strong authentication, integrity, and confidentiality. While HTTPAuth provided a means
4240 for password and digest-based authentication and integrity for HTTP, it failed to provide strong
4241 authentication or confidentiality. S-HTTP defines its own URL protocol designator, namely
4242 shttp.⁷ When a S-HTTP aware client or server detects a shttp URL, it individually secures HTTP
4243 requests and responses while preserving the transaction model and implementation
4244 characteristics of HTTP. The S-HTTP protocol provides flexibility in choice of cryptographic
4245 algorithms, key management mechanisms, and security policy by negotiating each option
4246 between the client and server. Key exchange mechanisms include a password-style keying,
4247 manually shared secret keys, and public key. The protocol has the capacity to use a variety of
4248 cryptographic message formats, including CMS and MOSS. While effective, S-HTTP was never
4249 very successful as a technology. S-HTTP is seldom used today.

4250 (U) The IETF WebDAV working group is now taking another look at developing traditional
4251 application security for web transactions that are not well served by the simple SSL treatment of
4252 the application layer. Web authoring, as opposed to browsing, has a strong emphasis on
4253 authentication, access control, and privileges. The Distributed Authoring Protocol (WebDAV)
4254 built a framework for distributed authoring by standardizing HTTP extensions to support
4255 overwrite prevention (locking), metadata management (properties), and namespace management
4256 (copy, move, collections). The Access Control Protocol (WebDAV-AC) builds upon this to
4257 provide the means for a web client to read and modify access control lists (ACLs) that instruct a
4258 server whether to allow or deny operations upon a resource. As implementation of List Based
4259 Access Control (LBAC) fundamentally requires authentication, WebDAV-AC relies on existing
4260 authentication mechanisms defined for use with HTTP. WebDAV-AC particularly specifies that
4261 if the basic authentication in HTTPAuth is used, it must be performed over secure transport such
4262 as TLS. WebDAV is still a relatively young developing standard, and its support level in
4263 industry is still relatively low.

4264 (U) On the whole, traditional application security has not been very competitive for web security.
4265 The ubiquitous support for SSL in web browsers has made a lot of past web security efforts
4266 irrelevant. However, the WebDAV effort appears to recognize the limits of SSL technology, and
4267 is exploring richer application security features.

⁷ (U) This should not be confused with “https,” which signifies SSL/TLS security.

4268 2.3.3.2.1.1.1.3(U) Strong Client Authentication

4269 (U) Several applications are known to employ traditional application security elements as part of
4270 their authentication design. Some employ the reusable module philosophy already demonstrated
4271 for S/MIME. Applications known to do this include the Simple Authentication and Security
4272 Layer (SASL), the Post Office Protocol (POP3), the Internet Message Access Protocol (IMAP),
4273 the Application Configuration Access Protocol (ACAP), and security extensions to the File
4274 Transfer Protocol (FTP).

4275 (U) Addition of strong client authentication has been a success from a standardization
4276 perspective. However, from an implementation and deployment standpoint the track record is
4277 spotty. IMAP products commonly incorporate strong authentication. However, POP products
4278 still commonly rely on plaintext passwords. ACAP products have been very slow to emerge
4279 overall, but incorporate strong security where they exist. FTP products incorporating strong
4280 authentication exist, but are seldom used today.

4281 2.3.3.2.1.1.1.4(U) Summary

4282 (U) As their widespread use demonstrates, traditional layered application security technologies
4283 have a large footprint in industry and represent a mature, stable development path. However,
4284 their maturity is offset by the long lead time associated with their evolution. It is noteworthy,
4285 though, that a lot of this lead time is not profitless in that it allows interest and enthusiasm for the
4286 standard to build in the vendor community before the standard is finalized. This can lead to
4287 improved standards and more widespread support among vendors. Many application security
4288 protocols now also embrace a modular design philosophy, such as employed by S/MIME, CMC,
4289 and others, which promises to shorten future development cycles.

4290 2.3.3.2.1.1.1.2 (U) Usage Considerations

4291 (U) Application security is generally highly tailored to the needs of the application in question.
4292 Since the applications that will make up the GIG are necessarily a moving target, it is difficult to
4293 provide a comprehensive overview of specific application security technologies that are of
4294 potential interest to the GIG community. That type of analysis is best conducted within the
4295 framework of a particular project (e.g., DMS, GDS).

4296 2.3.3.2.1.1.1.3 (U) Maturity

4297 (U) Overall, the traditional application security technology represents a mature foundation for
4298 GIG development. Many application security standards have been developed. Some have
4299 succeeded while others have failed. Products for most widely-used applications offer at least
4300 some form of embedded application security today. Secured application products are generally
4301 available, functional, reasonably secure, interoperable and well tested. However, the maturity of
4302 specific application security varies dramatically. S/MIME security is widely available in mail
4303 clients. Embedded strong IMAP authentication is likewise mature and dependable. Toolkits are
4304 available to facilitate rapid integration of many technologies into existing or new product
4305 developments. However, other more negative examples, such as strong POP3 authentication and
4306 S-HTTP, also exist. Thus the maturity of the different individual technologies must be assessed
4307 individually.

4308 2.3.3.2.1.1.1.4 (U) Standards
 4309 (U) Table 2.3-1 summarizes pertinent application and traditional application security standards
 4310 discussed in this section.

4311 **Table 2.3-1: (U) Traditional Layered Application Security Standards**

This table is (U)				
Reference	Forum	Standards	Date	Maturity
[SMTP]	IETF	RFC 821: Simple Mail Transfer Protocol	August 1982	Standard
[MSGFMT]	IETF	RFC 822: Standard for the Format of ARPA Internet Text Messages	August 1982	Standard
[PEM]	IETF	RFC 1421: Privacy Enhancement for Internet Electronic Mail: Part I: Message Encryption and Authentication Procedures	February 1993	Proposed Standard
	IETF	RFC 1422: Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management	February 1993	Proposed Standard
	IETF	RFC 1423: Privacy Enhancement for Internet Electronic Mail: Part III: Algorithms, Modes, and Identifiers	February 1993	Proposed Standard
	IETF	RFC 1424: Privacy Enhancement for Internet Electronic Mail: Part IV: Key Certification and Related Services	February 1993	Proposed Standard
[MOSS]	IETF	RFC 1848: MIME Object Security Services	October 1995	Proposed Standard
[CMS]	IETF	RFC 3852: Cryptographic Message Syntax (CMS)	July 2004	Proposed Standard
[MSG]	IETF	RFC 3851: S/MIME v3.1 Message Specification	July 2004	Proposed Standard
[CERT]	IETF	RFC 3850: S/MIME v3.1 Certificate Handling	July 2004	Proposed Standard
[ESS]	IETF	RFC 2634: Enhanced Security Services for S/MIME	June 1999	Proposed Standard
	IETF	RFC 3854: Securing X.400 Content with S/MIME	July 2004	Proposed Standard
	IETF	RFC 3855: Transporting S/MIME Objects in X.400	July 2004	Proposed Standard
	IETF	RFC 3370: CMS Algorithms	August 2002	Proposed Standard
[CMC]	IETF	RFC 2797: Certificate Management Messages over CMS	April 2000	Proposed Standard
[HTTP]	IETF	RFC 2616: Hypertext Transfer Protocol - HTTP/1.1	June 1999	Draft Standard
[HTTPAuth]	IETF	RFC 2617: HTTP Authentication: Basic and Digest Access Authentication	June 1999	Draft Standard
[S-HTTP]	IETF	RFC 2660: The Secure HyperText Transfer Protocol	August 1999	Experimental

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U)				
Reference	Forum	Standards	Date	Maturity
[WebDAV}	IETF	RFC 2518: HTTP Extensions for Distributed Authoring -- WEBDAV	February 1999	Proposed Standard
[WebDAV-AC]	IETF	RFC 3744: WebDAV Access Control Protocol	May 2004	Proposed Standard
[SASL]	IETF	RFC 2222: Simple Authentication and Security Layer (SASL)	October 1997	Proposed Standard
	IETF	RFC 2444: The One-Time-Password SASL Mechanism	October 1998	Proposed Standard
	IETF	RFC 2554: SMTP Service Extension for Authentication	March 1999	Proposed Standard
[POP3]	IETF	RFC 1939: Post Office Protocol - Version 3	May 1996	Standard
	IETF	RFC 2449: POP3 Extension Mechanism	November 1998	Proposed Standard
	IETF	RFC 1734: POP3 AUTHentication command	December 1994	Proposed Standard
	IETF	RFC 3206: The SYS and AUTH POP Response Codes	February 2002	Proposed Standard
[IMAP4]	IETF	RFC 3501: Internet Message Access Protocol (IMAP) - Version 4rev1	March 2003	Proposed Standard
	IETF	RFC 2195: IMAP/POP AUTHorize Extension for Simple Challenge/Response	September 1997	Proposed Standard
	IETF	RFC 1731: IMAP4 Authentication Mechanisms	December 1994	Proposed Standard
	IETF	RFC 2086: IMAP4 ACL extension	January 1997	Proposed Standard
	IETF	RFC 2228: FTP Security Extensions	October 1997	Proposed Standard
[ACAP]	IETF	RFC 2244: Application Configuration Access Protocol	November 1997	Proposed Standard
[X.400]	ITU-T	X.400: Information Technology – Message Handling Systems (MHS) – Message Handling System and Service Overview	June 1999	Final Recomm.
[X.402]	ITU-T	X.402: Information Technology – Message Handling Systems (MHS) – Overall Architecture	June 1999	Final Recomm.
[X.411]	ITU-T	X.411: Information Technology – Message Handling Systems (MHS) – Message transfer system: Abstract Service Definition and Procedures	June 1999	Final Recomm.
[MSP]	NSA	SDN.701: Message Security Protocol	June 1996	v4.0
[CSP]	CCEB	ACP 120: Common Security Protocol (CSP)	June 1998	Base Edition

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U)				
Reference	Forum	Standards	Date	Maturity
[PKCS7]	RSA	PKCS #7: Cryptographic Message Syntax Standard	November 1993	v1.5
This table is (U)				

4312 2.3.3.2.1.1.1.5 (U) Dependencies

4313 (U) Traditional application security technologies rely extensively on cryptographic technologies
 4314 to provide encryption, digital signature, hash, and key exchange algorithms.

4315 (U) Protocol development is a key enabling technology for traditional application security. At
 4316 present, the dominant techniques rely on the following technologies:

- 4317 • (U) Object-oriented design based on modeling in Abstract Syntax Notation One (ASN.1)
- 4318 • (U) Syntactic description using Augmented Backus-Naur Format (ABNF)
- 4319 • (U) Description of eXtensible Markup Language (XML) syntax using XML-Schema
 4320 techniques
- 4321 • (U) Formal system state analysis and modeling
- 4322 • (U) Other formal techniques.

4323 (U) These combined with a liberal application of English descriptive and ad-hoc techniques form
 4324 a necessary part of application security development.

4325 2.3.3.2.1.1.2 (U) Session Security Technologies

4326 (U) As an alternative to developing application security, many applications choose to rely on
4327 session security technology. Session security technology protects all user data passed over a
4328 virtual connection between peer applications. Implementation of session security technology
4329 varies, and can be modeled variously as part of the application layer, or part of the transport
4330 layer. Session security technologies afford many of the same protections to user information, but
4331 with reduced flexibility and perhaps often with less permanence. Session security technologies
4332 are, however, vastly simpler than traditional layered application security and frequently offer
4333 rapid integration via exposed APIs.

4334 2.3.3.2.1.1.2.1 (U) Technical Detail

4335 2.3.3.2.1.1.2.1.1 (U) Secure Sockets Layer & Transport Layer Security

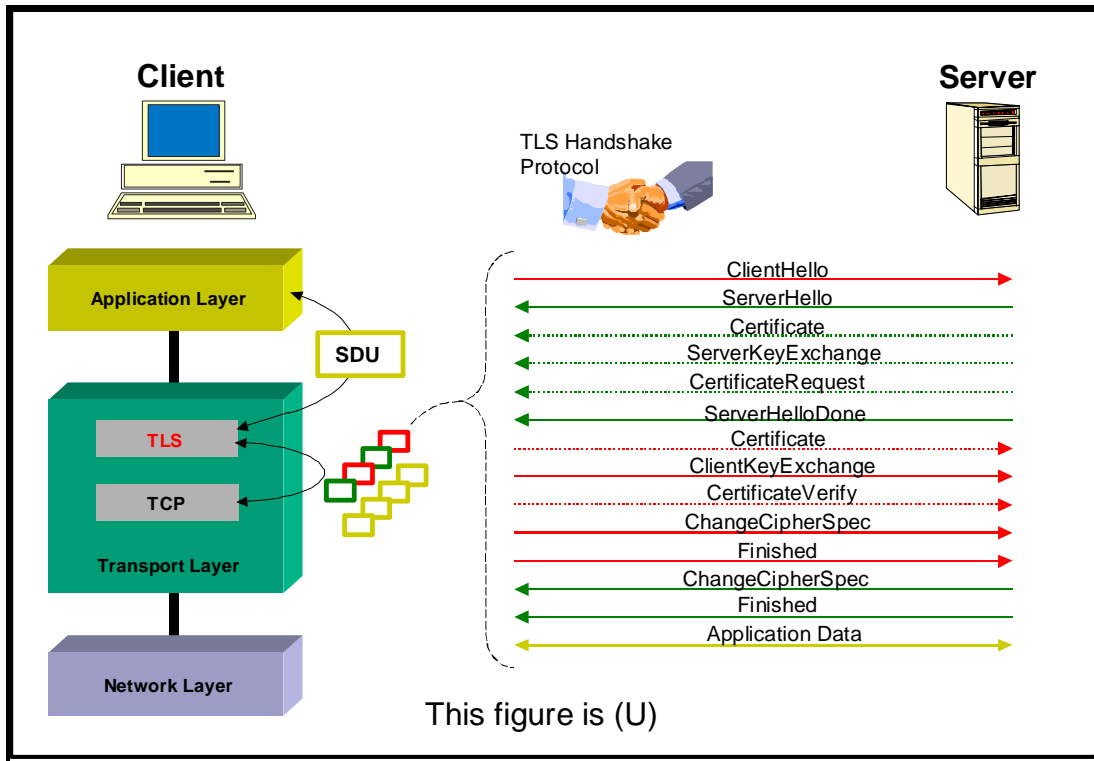
4336 (U) The Secure Sockets Layer began as a proprietary technology developed by Netscape. SSL
4337 provided an extension to the popular Berkeley Sockets and Windows Sockets API to allow
4338 applications to invoke security services provided by a common encapsulation protocol. Initially,
4339 SSL was developed to service HTTP exclusively. Eventually it began to be used by broader
4340 range of applications.

4341 (U) As SSL use became widespread, an effort was made to open the protocol and API definition
4342 to industry. This led to the development of the Transport Layer Security (TLS) standard in the
4343 IETF. TLS v1.0 is based on the SSL v3.0, but the two protocols do not interoperate. TLS
4344 implementations can, however, fall back to SSL 3.0 during negotiation. TLS v1.0 offers more
4345 flexibility in features and cryptography than SSL v3.0 and is expected to be the platform for all
4346 future evolution and development of the technology.

4347 (U) TLS works by using the TLS Record Protocol to fragment data into manageable blocks.
4348 Each block has a MAC code applied, is (optionally) encrypted, and the resulting block is
4349 transmitted via TCP. Record Protocol might also compress the fragmented data—depending on
4350 the specific implementation. TLS uses the Record Protocol as a foundation for different types of
4351 protocol exchanges. The basic TLS specification defines four record types. Additional record
4352 types are supported as an extension mechanism. The following types are defined:

- 4353 • (U) Handshake Protocol – Enables mutual establishment of identity between the client
4354 and server, and for negotiation of TLS options
- 4355 • (U) Alert Protocol – Conveys information about important events in the communication
4356 such as normal closure of the association and errors
- 4357 • (U) Change Cipher Spec Protocol – Enables the client and server to signal and mutually
4358 acknowledge transitions in ciphering strategies
- 4359 • (U) Application Data Protocol – Conveys the fragmented, compressed, and encrypted
4360 application data. Messages are treated as transparent data.

4361 (U) Although three of these protocols are quite simple, the TLS Handshake Protocol uses several
4362 staged exchanges. Figure 2.3-4 illustrates the context and operation of TLS and the Handshake
4363 Protocol.



4364

4365

Figure 2.3-4: (U) TLS Handshake Protocol

4366

(U) The TLS Handshake Protocol involves the following steps:

4367

- (U) Exchange hello messages to agree on algorithms, exchange random values, and check for session resumption

4368

4369

- (U) Exchange the necessary cryptographic parameters to allow the client and server to agree on a pre-master secret

4370

4371

- (U) Exchange certificates and cryptographic information to allow the client and server to authenticate themselves

4372

4373

- (U) Generate a master secret from the pre-master secret and exchanged random values

4374

- (U) Provide security parameters to the record layer

4375

- (U) Allow the client and server to verify that their peer has calculated the same security parameters and that the handshake occurred without tampering.

4376

4377 (U) While the average user experience with TLS has mainly to do with confidentiality and
4378 integrity, the protocol is capable of strong mutual authentication. Authentication is only as strong
4379 as the Public Key Infrastructure (PKI) underlying the certificates issued to the client and server.
4380 While TLS-enabled servers commonly have certificates issued for their domains, most web
4381 browser implementations using TLS do not. Such browsers commonly establish an anonymous,
4382 but encrypted association with the TLS server and then perform basic authentication within that
4383 virtual circuit in accordance with HTTPAuth. When properly provisioned with certificates, TLS
4384 is capable of providing the following security services to an application:

- 4385 • (U) Authentication of Server Identity
- 4386 • (U) Authentication of Client Identity
- 4387 • (U) Data Confidentiality
- 4388 • (U) Data Integrity.

4389 (U) TLS has been successfully applied to several different applications including:

- 4390 • (U) HTTP (see HTTPTLS)
- 4391 • (U) LDAPv3 (see LDAPAuth and LDAPTLS)
- 4392 • (U) POP (see RFC 2595)
- 4393 • (U) IMAP (see RFC 2595)
- 4394 • (U) ACAP (see RFC 2595)
- 4395 • (U) SMTP (see SMTPTLS).

4396 2.3.3.2.1.1.2.1.2(U) Generic Upper Layer Security

4397 (U) In the early 1990s, the International Organization for Standardization (ISO) and International
4398 Telecommunication Union Telecommunication Standardization Sector (ITU-T) began to
4399 recognize a gap between the requirements for applications security set forth in CCITT X.800 |
4400 ISO/IEC 7498-2 (see OSISecArc]) and ITU-T X.803 | ISO/IEC 10745 (see ULSecMode]) and
4401 the practice of building security into individual applications from scratch. This realization led
4402 eventually to the development of the Generic Upper Layer Security (GULS) standards. GULS
4403 provided a set of standardized ASN.1 conventions to facilitate development of secure application
4404 syntaxes. It also defined a Security Exchange Service Element (SESE), which would establish
4405 and maintain a secure association over which application data could be exchanged securely. The
4406 SESE would function somewhat similarly to TLS. Unlike TLS, GULS was unambiguously
4407 modeled in the application layer and was distinct from OSI transport layer security standards.

4408 (U) Unfortunately, GULS was of little value to existing OSI applications (e.g., X.400 and X.500)
4409 without modification. Also, since GULS was unambiguously wedded to ASN.1 and the OSI
4410 application layer structures, it was only of value to OSI applications. The total collapse of
4411 interest in OSI development in the mid-1990s virtually eliminated any work on new OSI
4412 applications or updates to existing applications. These factors have combined to make GULS
4413 virtually irrelevant today.

4414 2.3.3.2.1.1.2.1.3(U) Summary

4415 (U) Session security technologies provide a very simple and potent solution for securing
4416 application communication. The development of TLS has proven extremely effective on
4417 widespread deployments and has been applied to a variety of applications. However, there are a
4418 number of limitations and security concerns on use of TLS for application security. GULS is of
4419 little present-day interest, but the similarity of evolution between GULS and TLS is noteworthy
4420 from the perspective of examining session security technologies as a whole.

4421 2.3.3.2.1.1.2.2 (U) Usage Considerations

4422 (U) Caution should be exercised in employing session security technologies, such as TLS, for
4423 application security purposes. The suitability of TLS depends heavily on it functioning in an
4424 overall security architecture. For example, TLS can be subjected to man-in-the-middle attacks.
4425 So care must be taken that strong 2-way authentication is applied during the Handshake Protocol,
4426 and that certificates or other credentials are validated and recognized. This is true even if
4427 subsequent access control based on [HTTPAuth] will be used within the TLS association. TLS
4428 is also vulnerable to compromise of its feature negotiation mechanisms. So care must be taken to
4429 ensure that the implementation minimum acceptable security measures reflect the security policy
4430 in force. TLS is also not suitable for application architectures that require secure multipoint
4431 communications, multiple different application entities or architectures that require persistent
4432 security that endures through a relaying application entity.

4433 2.3.3.2.1.1.2.3 (U) Maturity

4434 (U) Session security technologies are Mature (TRLs 7 – 9), and TLS in particular is a Mature,
4435 widely implemented, and well deployed solution. It is worth noting that most TLS client
4436 implementations operate without certificates or public keys by default. Most are not easily
4437 configurable to employ a per-application certificate much less a per-user certificate. Therefore it
4438 seems likely that more product improvement must take place for TLS to expand beyond web
4439 browsing and properly provide security to multiple applications.

4440 2.3.3.2.1.1.2.4 (U) Standards

4441 (U) Table 2.3-2 summarizes pertinent session security standards discussed in this section.

4442

Table 2.3-2: (U) Session Security Standards

This table is (U)				
Reference	Forum	Standards	Date	Maturity
[TLS]	IETF	RFC 2246: The TLS Protocol v1.0	January 1999	Proposed Standard
[HTTPTLS]	IETF	RFC 2817: Upgrading to TLS Within HTTP/1.1	May 2000	Proposed Standard
	IETF	RFC 2818: HTTP Over TLS	May 2000	Informational
[TLSEXT]	IETF	RFC 3546: TLS Extensions	June 2003	Proposed Standard
[AESTLS]	IETF	RFC 3268: AES Ciphersuites for TLS	June 2002	Proposed Standard
[LDAPAuth]	IETF	RFC 2829: Authentication Methods for LDAP	May 2000	Proposed Standard
[LDAPTLS]	IETF	RFC 2830: LDAPv3 Extension for TLS	May 2000	Proposed Standard
[LDAPv3]	IETF	RFC 3377: LDAP v3 Technical Specification	September 2002	Proposed Standard
[RFC2595]	IETF	RFC 2595: Using TLS with IMAP, POP3 and ACAP	June 1999	Proposed Standard
[SMTPTLS]	IETF	RFC 3207: SMTP Service Extension for Secure SMTP over TLS	February 2002	Proposed Standard
[GULS]	ISO	ISO/IEC 11586-1: Information technology -- Open Systems Interconnection -- Generic upper layers security: Overview, models and notation	1996	International Standard
	ISO	ISO/IEC 11586-2: Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) service definition	1996	International Standard
	ISO	ISO/IEC 11586-3: Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) protocol specification	1996	International Standard
	ISO	ISO/IEC 11586-4: Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax specification	1996	International Standard

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U)				
Reference	Forum	Standards	Date	Maturity
	ISO	ISO/IEC 11586-5: Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) Protocol Implementation Conformance Statement (PICS) proforma	1997	International Standard
	ISO	ISO/IEC 11586-6: Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax Protocol Implementation Conformance Statement (PICS) proforma	1997	International Standard
[OSISecArch]	ISO	ISO/IEC 7498-2: Data Communication Networks – Open Systems Interconnection (OSI) – Security, Structure and Applications – Security Architecture for Open Systems Interconnection for CCITT Applications	1989	International Standard
[ULSecModel]	ISO	ISO/IEC 10745: Information Technology – Open Systems Interconnection – Upper Layers Security Model	July 1994	International Standard
[OSISecArch]	ITU-T	CCITT X.800: Data Communication Networks – Open Systems Interconnection (OSI) – Security, Structure and Applications – Security Architecture for Open Systems Interconnection for CCITT Applications	1991	Final Recomm.
[ULSecModel]	ITU-T	ITU-T X.803: Information Technology – Open Systems Interconnection – Upper Layers Security Model	July 1994	Final Recomm.
[GULS]	ITU-T	ITU-T X.830: Information technology -- Open Systems Interconnection -- Generic upper layers security: Overview, models and notation	April 1995	Final Recomm.
	ITU-T	ITU-T X.831: Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) service definition	April 1995	Final Recomm.
	ITU-T	ITU-T X.832: Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) protocol specification	April 1995	Final Recomm.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U)				
Reference	Forum	Standards	Date	Maturity
	ITU-T	ITU-T X.833: Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax specification	April 1995	Final Recomm.
	ITU-T	ITU-T X.834: Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) Protocol Implementation Conformance Statement (PICS) proforma	October 1996	Final Recomm.
	ITU-T	ITU-T X.835: Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax Protocol Implementation Conformance Statement (PICS) proforma	October 1996	Final Recomm.
This table is (U)				

4443 2.3.3.2.1.1.2.5 (U) Dependencies

4444 (U) Neither cryptography nor security protocol development are discussed in detail in this
 4445 section. However, session security technologies have a similar dependency on them.

4446 2.3.3.2.1.1.3 (U) Web Services Security Technologies

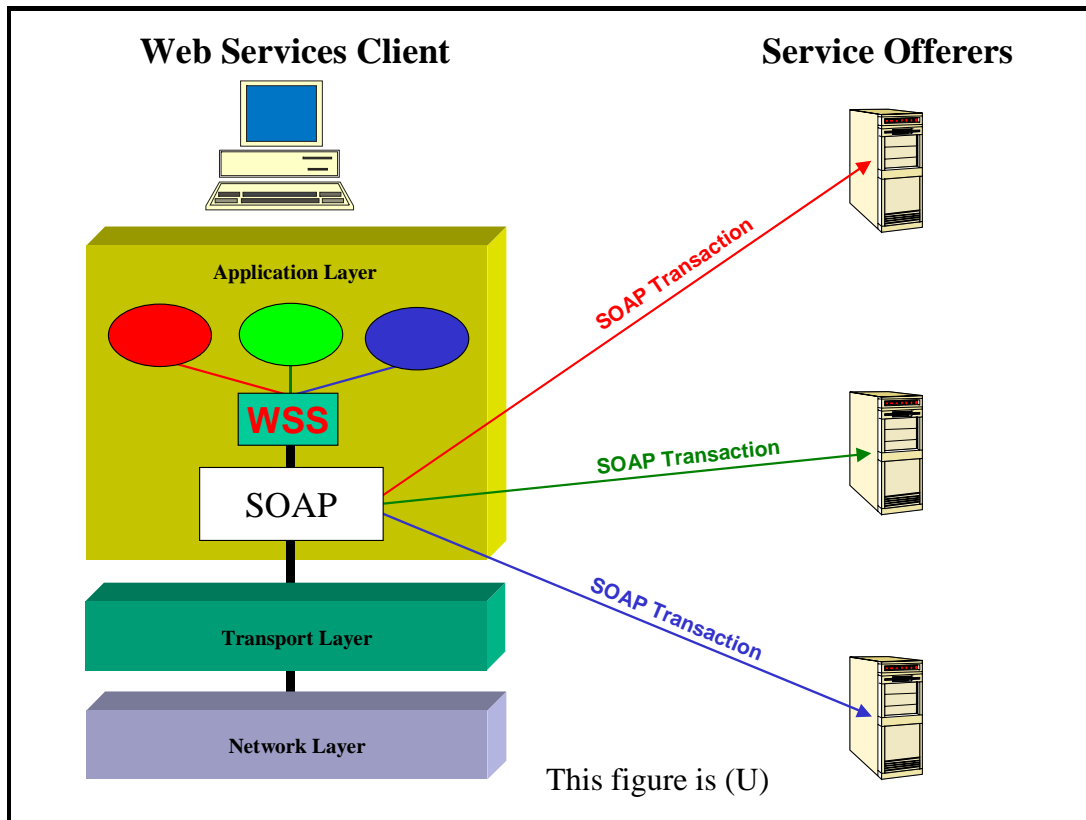
4447 (U) The future of application development for the GIG is expected to take a different direction
4448 from past application layer development. The emphasis for GIG applications is expected to be
4449 service-oriented architectures. And the primary focus for service-oriented application
4450 development is the technology known as Web Services. Unfortunately, development security
4451 technology for Web Services is still in its infancy.

4452 2.3.3.2.1.1.3.1 (U) Technical Detail

4453 (U) With the tremendous success of web browsing as the Internet's second killer application,
4454 pressure grew to leverage the success and ubiquity of the web for other purposes. Recognition
4455 also dawned that while HTTP servers and dynamically generated HTML documents were
4456 sufficient to allow humans users basic access to databases, they were not sufficient to enable
4457 automated systems to access information in those same databases. This was a function of HTML
4458 being optimized for specifying presentation rather than semantics. This led to the development of
4459 the XML, which was optimized instead for identifying the semantics of data.

4460 (U) In the late 1990s, the Simple Object Access Protocol (SOAP) was developed as a means to
4461 allow XML objects to be requested and transferred over HTTP or a variety of other protocols.
4462 SOAP provides an XML envelope consisting of a heading and body. The specification in SOAP
4463 provides bindings between SOAP and HTTP so that SOAP transactions can take advantage of
4464 the existing, ubiquitous HTTP infrastructure. Other bindings, such as to SMTP or other existing
4465 protocols, are also possible but seldom seen. Services built to request and delivery specific data
4466 using XML and SOAP have come to be known as Web Services.

4467 (U) Developing security services as a common add-on to the web services framework offer
4468 significant benefits over traditional layered application security development. Figure 2.3-5
4469 contrasts the web services model with that shown previously for CMS and S/MIME. In the web
4470 services framework a variety of service offerings can be provided through SOAP and HTTP.
4471 Each service would benefit from the same security elements applied to the common SOAP
4472 envelope. This form of security is called Web Services Security (WSS). Conceptually, WSS has
4473 much in common with a reusable module, such as CMS, or session security services, such as
4474 TLS. However, WSS has the potential to combine the best elements of both.



4475

4476

Figure 2.3-5: (U) Model for Web Services Security

4477 (U) Different organizations are involved in developing standards and specifications for WSS.
 4478 The World Wide Web Consortium (W3C), the organization responsible for the original
 4479 development of both XML and SOAP, has contributed to the development of WSS by
 4480 introducing the XML Digital Signature (XML_DSIG) and XML Encryption (XML_ENC)
 4481 standards. These have the potential to become foundation standards for more advanced WSS
 4482 development. In competition with the W3C standards is the work of the American National
 4483 Standards Institute (ANSI). ANSI has developed an XML Cryptographic Message Syntax
 4484 (XCMS) which provides functions similar to XML_DSIG and XML_ENC, but does so by
 4485 applying a relatively simple XML wrapper to the existing IETF CMS wrappers. It is unclear at
 4486 this point which approach will dominate.

4487 (U) The Organization for the Advancement of Structured Information Standards (OASIS) is
 4488 developing several standards that have the promise to contribute to WSS. These include SAML,
 4489 XACML, and WSS.

4490 (U) Another significant WSS development is under way at the Web Services Interoperability
 4491 (WS-I) Organization. WS-I is engaged in an effort to achieve commonality and interoperability
 4492 among web service components. WS-I has already released the WS-I Basic Profile for web
 4493 services and is continuing work on a draft Basic Security Profile for WSS.

4494 (U) Another contender in the WSS area is Liberty Alliance. They are focused on solving the
 4495 problem of cooperation between federated web services to provide secure operation where all of
 4496 the participants may not be part of the same organization or necessarily share a common security
 4497 policy. It is unclear how the Liberty Alliance work will ultimately affect the overall WSS effort.
 4498 Liberty Alliance has released three sets of standards that promise to have an impact on WSS.

- 4499 • (U) The Identity Federation Framework (ID-FF) offers an approach for establishing a
 4500 standardized, multi-vendor, web-based SSO with federated identities based on commonly
 4501 deployed technologies
- 4502 • (U) The Identity Web Services Framework (ID-WSF) is a set of specifications for
 4503 creating, using, and updating various aspects of identities
- 4504 • (U) The Identity Services Interface Specifications (ID-SIS) define profiles for commonly
 4505 useful services, including a personal profile service (ID-SIS-PP) that provides basic
 4506 profile information such as contact information and an employee profile service (ID-SIS-
 4507 EP) that provides Employee's basic profile information.

4508 2.3.3.2.1.1.3.2 (U) Maturity

4509 (U) WSS standards are Emerging (TRLs 4 – 6). They are still under development and are not
 4510 ready for full scale deployment. Further, there are different standards competing for many of the
 4511 same functional requirements. It is not clear at this point which standards will succeed and in
 4512 what market segments. It is possible that some security standards will prove to be suited to
 4513 certain types of web service while others will better support different forms of web service. So
 4514 there is considerable risk in early adoption of any of these immature solutions.

4515 2.3.3.2.1.1.3.3 (U) Standards

4516 (U) Table 2.3-3 summarizes pertinent web services security standards discussed in this section.

4517 **Table 2.3-3: (U) Web Services Security Standards**

This table is (U)

Reference	Forum	Standards	Date	Maturity
[XML]	W3C	XML		Final
		XML Schema		Stable
[XML-DSIG]		XML-DSIG		Final
[XML-ENC]		XML-ENC		Final
		XKMS		Revision
[SOAP]		SOAP		Revision
		WSDL		Revision
[SAML]	OASIS	SAML		Stable
[XACML]		XACML		Revision
		UDDI		Revision
		SPML		Stable
		XCBF		Final

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U)				
Reference	Forum	Standards	Date	Maturity
		XCBF Token Profile		Final
[WSS]		Web Services Security (WSS)		Revision
		WSS UsernameToken Profile		Revision
		WSS X.509 Certificate Token Profile		Revision
		Web Services Reliable Messaging		Draft
		ebXML Registry		
		ebSOA		
		WSDM		
		XrML (eXtensible Rights Management Language)		Draft
		Web Application Security		
		Digital Signature Services		
		Security Services		
		Web Services Distributed Management		
[WSI-SEC]	WS-I	Basic Security Profile Security Scenarios		Draft
		Basic Profile		Revision
	ANSI	ANSI X9.84 (XCBF)		Final
[XCMS]		ANSI X9.96 (XCMS)		
		ANSI X9.73 (CMS)		
	ITU-T	ITU-T X.509		
	ISO	ISO 19092 (biometric formats)		Draft
[ID-FF]	Liberty Alliance	ID-FF		Stable
[ID-SIS]		ID-SIS		Revision
[ID-WSF]		ID-WSF		Revision
		draft-lib-arch-soap-authn		Draft
This table is (U)				

4518 2.3.3.2.1.1.3.4 (U) Dependencies

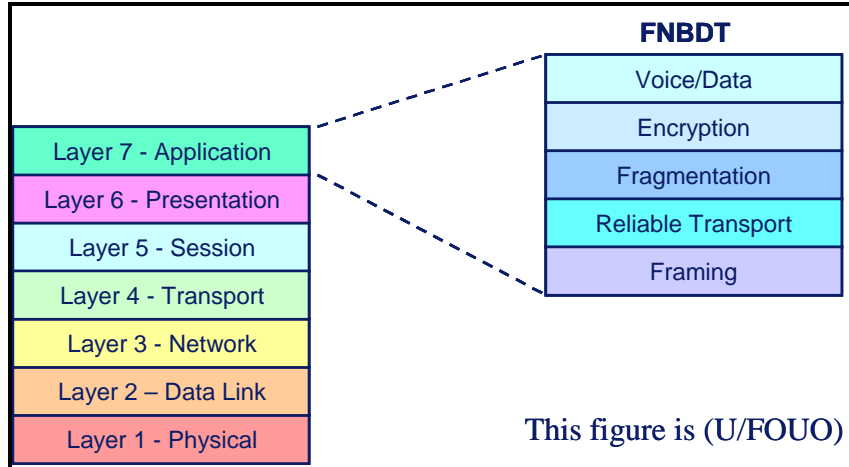
4519 (U) Neither cryptography nor security protocol development are discussed in detail in this
 4520 section. However, web services security technologies have a similar dependency on them. It
 4521 should be noted that web services' exclusive focus on SOAP and XML narrow the range of
 4522 techniques used in security protocol development.

4523 **2.3.3.2.1.2 (U) Real-Time Data Technologies**

4524 2.3.3.2.1.2.1 (U) FNBDT

4525 2.3.3.2.1.2.1.1 (U) Technical Detail

4526 (U) Future Narrowband Digital Terminal (FNBDT) is a group of signaling and cryptography
 4527 specifications designed to allow end-to-end secure communications using commercial
 4528 communications channels. FNBDT operates at the Application Layer (see Figure 2.3-6) and is
 4529 designed to operate over whatever transport method is available.



4530

4531 **Figure 2.3-6: (U) FNBDT Location in Network Protocol Stack**

4532 (U//FOUO) FNBDT specifications define the following aspects of secure voice and data
 4533 communication:

- 4534 • (U//FOUO) The signaling required to establish and maintain secure calls independent of
 4535 the transport network
- 4536 • (U//FOUO) A Minimum Essential Requirement (MER) mode which guarantees
 4537 interoperability between FNBDT-compliant devices
- 4538 • (U//FOUO) Key management for generating and maintaining compatible encryption keys
- 4539 • (U) Encryption algorithms
- 4540 • (U//FOUO) MELP (2400 bps) and G.729D (6400 bps) voice coders
- 4541 • (U//FOUO) Cryptographic synchronization management functionality
- 4542 • (U//FOUO) An escape mechanism enabling vendors to implement proprietary modes.

4543 (U//FOUO) Currently the FNBDT specifications specify only Type 1 encryption methods,
 4544 although the signaling is directly applicable to vendor-defined non-Type 1 applications. Multiple
 4545 vendors have introduced Type 1 and non-Type 1 products based on the FNBDT specifications.

4546 (U//FOUO) FNBDT provides the ability for products to operate in high Bit Error Rate (BER)
4547 environments. Establishing an FNBDT channel involves an initial negotiation of capabilities
4548 between endpoints, with the ability to select vendor proprietary modes if both endpoints have
4549 compatible capabilities. Compatible operational modes, encryption algorithms, and key sets are
4550 also selected during this initial exchange.

4551 2.3.3.2.1.2.1.2 (U) Usage Considerations

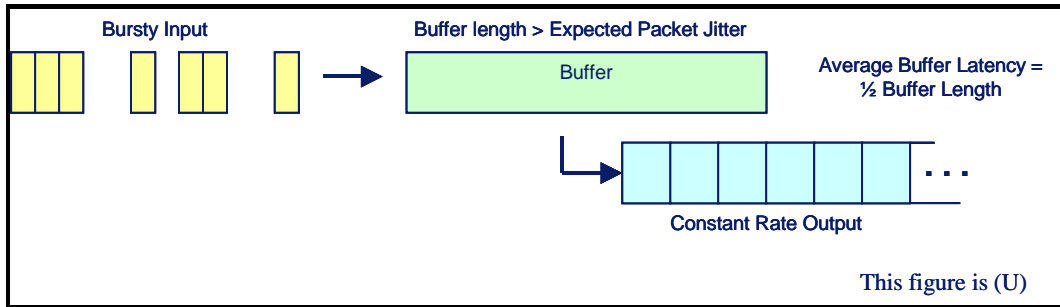
4552 2.3.3.2.1.2.1.2.1 (U) Implementation Issues

4553 (U//FOUO) The FNBDT signaling protocol at the Application Layer has proved to be a
4554 successful method of providing security for voice systems. While the FNBDT protocol is not
4555 oriented toward a packet-based system, it does not inherently prohibit operating with such a
4556 system. FNBDT is a streaming protocol that defines a constant-rate bitstream. For voice
4557 applications, this bitstream is either 2400 bps or 7200 bps. As long as the receiving end of the
4558 communication link can receive the bits and reformat them into the same constant-rate bitstream
4559 that was presented to the network at the transmit end of the link, the FNBDT signaling protocol
4560 will be adequate for secure voice applications.

4561 (U) Packet-based transport systems present unique challenges for streaming protocols such as
4562 FNBDT. The following list identifies several sources of degradation introduced by packet-based
4563 systems and evaluates the tolerance of the FNBDT signaling protocol to these degradation
4564 sources.

4565 (U//FOUO) Packet latency. This refers to the network delay in transporting bits from one end to
4566 the other. Two-way real-time applications such as voice conversations are negatively affected by
4567 total delay times that are perceptible to the user, typically in the 0.5 sec range. Because there are
4568 other sources of delay in the system besides packet transport time, the delay introduced by packet
4569 transport must be significantly less than this. The FNBDT protocol is not inherently affected by
4570 increased packet latency, although of course the regenerated speech at the receiving end of the
4571 link will be delayed accordingly.

4572 (U//FOUO) Packet jitter. Packet jitter refers to the difference in time required to transport
4573 packets, as opposed to the absolute delay (packet latency). Streaming protocols such as FNBDT
4574 are required to maintain a constant-rate output even when the network transport mechanism
4575 results in packets arriving at different times. This is typically resolved by buffering at the receive
4576 end of the link. Packets are fed into the buffer at varying times as they arrive, but are read out of
4577 the buffer at the constant (streaming) rate required by the application, a process shown in Figure
4578 2.3-7. The buffer must be able to accommodate the largest potential jitter, and therefore the net
4579 result of this arrangement is that the received signal is delayed enough to account for the largest
4580 potential jitter. This delay is in addition to the delay introduced by packet latency. As with packet
4581 latency, the FNBDT protocol is not inherently affected by increased jitter.



4582

4583

Figure 2.3-7: (U) Packet Jitter Mitigation Process

4584 (U) Packet loss. Packets may be lost during the transport process, resulting in missing data at the
 4585 receiver. In the case of secure applications, missing data invariably leads to loss of cryptographic
 4586 synchronization. Any subsequent data received and decrypted will be garbled until cryptographic
 4587 synchronization is re-established. This potentially devastating situation is mitigated by the
 4588 FNBDT signaling protocol, which includes embedded cryptographic synchronization
 4589 information periodically in the transmitted bitstream. This cryptographic re-synchronization
 4590 information occurs every 320 msec for G.729D speech and every 540 msec for MELP speech.
 4591 The potential impact of individual lost packets is therefore a short (0-500 msec) section of
 4592 garbled speech during a conversation. Periods of sequential lost packets will result in
 4593 appropriately long periods of missing or garbled speech, with a 0-500 msec period for re-
 4594 establishing cryptographic synchronization when the packets begin arriving again.

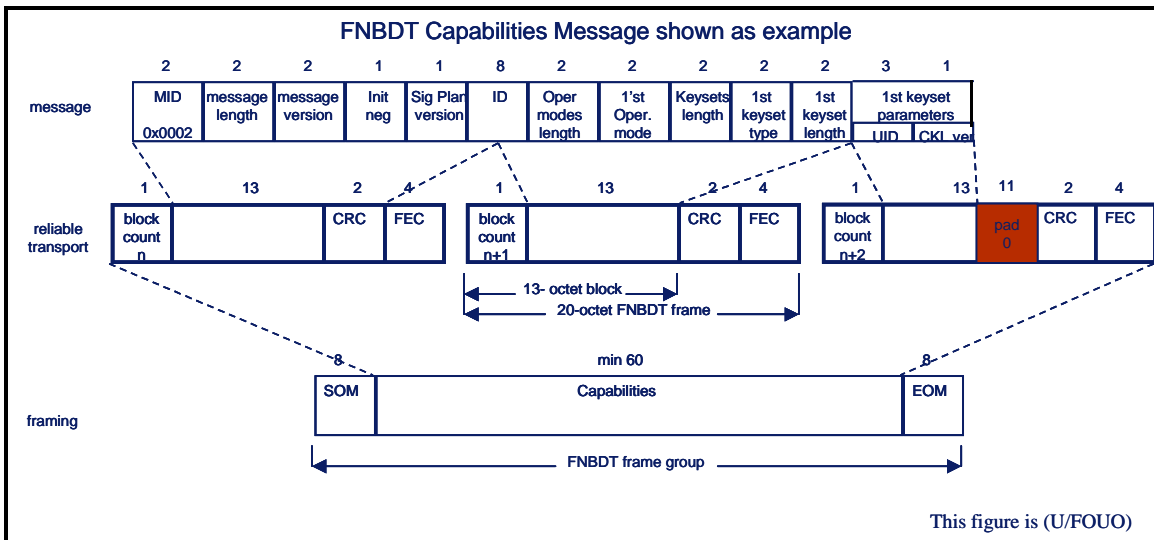
4595 (U) Packet re-ordering. Some packet transport systems have the capability to use different paths
 4596 for transporting packets, resulting in the potential for packets to arrive out of order. Often the
 4597 transport system has the capability to rearrange the received packets to the correct order before
 4598 presenting them to the upper layers, resulting in packet jitter rather than actual ordering errors in
 4599 the bitstream. If, however, information is presented to an FNBDT receiver with segments out of
 4600 order, the out-of-order segments will result in random (garbled) information. The length of any
 4601 such garbled data will depend on the packet size. Since speech applications will likely keep
 4602 packets small in order to reduce latency, the period of speech degradation will likewise be small.
 4603 Packet re-ordering issues lead to cryptosync loss with appropriate recovery periods as described
 4604 in the previous paragraph.

4605 (U//FOUO) Packet bit-errors. Uncorrected bit errors within transmitted packets will have the
 4606 same effect as bit errors in a circuit switched network. The FNBDT protocol was designed for
 4607 relatively high bit-error rate environments (~2%) and includes automatic retransmission
 4608 capabilities for those portions of the signaling which must arrive error-free. Once a secure call is
 4609 established, the speech algorithms themselves are extremely tolerant to random bit errors.
 4610 Individual bit errors seldom result in noticeable degradation to the received speech. FNBDT
 4611 traffic modes use crypto methods that do not result in bit error extension, meaning that single bit
 4612 errors in the received ciphertext do not extend to multiple bit errors in the decrypted plaintext.

4613 2.3.3.2.1.2.1.2.2(U) Advantages

4614 (U//FOUO) FNBDT is an end-user to end-user protocol. Information is encrypted at the
 4615 transmitting end-user where traffic is generated and is never decrypted until it arrives at the
 4616 receiving end-user where the traffic is consumed. User data is protected through whatever
 4617 network and across whatever communication channels might be traversed. Where gateways are
 4618 required to deliver bits from one protocol stack to another (e.g., VoIP to PSTN) user data
 4619 remains encrypted as it traverses the gateway.

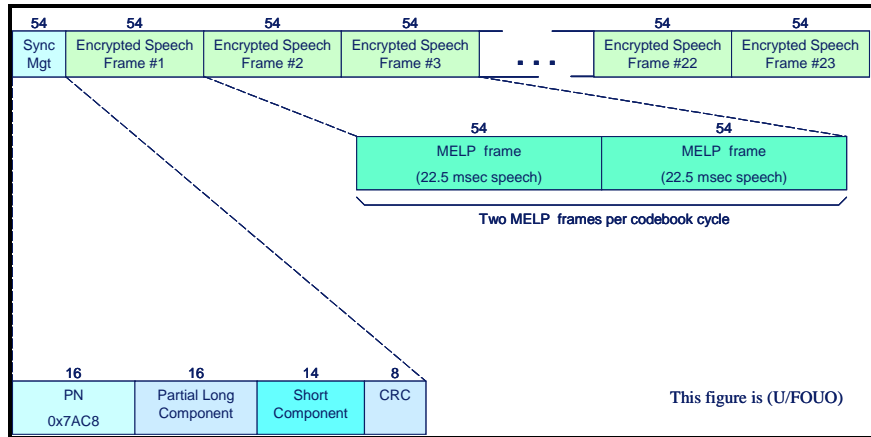
4620 (U//FOUO) The FNBDT protocol provides inherent transport reliability (Ack/Nak with
 4621 retransmission) for signaling messages. Voice modes operate without any underlying
 4622 retransmission protocol to reduce latency. Data modes are defined with and without
 4623 retransmission to allow increased throughput (Guaranteed Throughput mode) or increased
 4624 reliability (Reliable Transport mode) as required for specific applications. The frame structure
 4625 for signaling transport reliability and Reliable Transport data mode is shown in Figure 2.3-8.



4626
 4627 **Figure 2.3-8: (U//FOUO) FNBDT Frame Structure for Signaling Reliability and Reliable**
 4628 **Transport Data Mode**

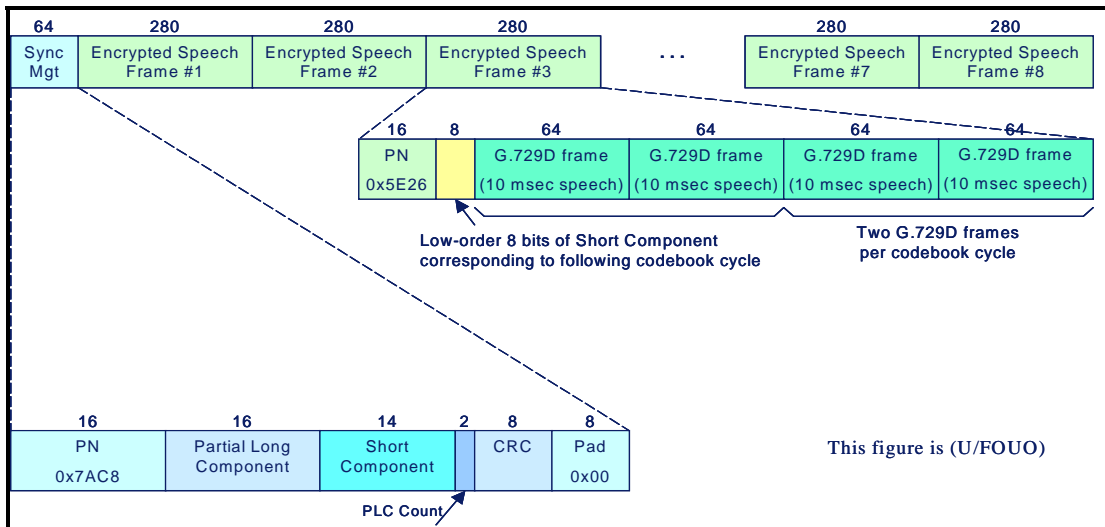
4629 (U//FOUO) An inherent strength of the FNBDT protocol is its ability to maintain cryptographic
 4630 synchronization for secure voice applications throughout signal fading and high BER
 4631 environments. Without this ability, the application data would continually need to be interrupted
 4632 to resynchronize the cryptography as data is lost or corrupted, leading to annoying gaps and
 4633 artifacts in encrypted speech.

4634 (U//FOUO) Synchronization is accomplished by periodically embedding information in the
 4635 transmitted bitstream. This allows the receiver to resynchronize the cryptography without using
 4636 channel resources other than the periodic embedded information. When the MELP vocoder has
 4637 been selected, the FNBDT specifications define both a Blank and Burst mode where
 4638 cryptographic resynchronization information replaces every 24th vocoder frame as indicated in
 4639 Figure 2.3-9, and a Blank without Burst mode where all vocoder frames are transmitted. The
 4640 Blank and Burst mode results in no additional overhead for the embedded resync information,
 4641 which occur every 540 msec and results in a composite bitstream of 2400 bps.



4642
 4643 **Figure 2.3-9: (U//FOUO) FNBDT 2400 bps MELP Blank and Burst Superframe Structure**

4644 (U//FOUO) When the G.729D vocoder has been selected, cryptographic resynchronization
 4645 information is inserted every 8th vocoder frame as shown by Figure 2.3-10. This allows the
 4646 cryptography to resynchronize every 320 msec and results in a composite bitstream of 7200 bps.



4647
 4648 **Figure 2.3-10: (U//FOUO) FNBDT 7200 bps G.729D Superframe Structure**

4649 (U//FOUO) FNBDT is particularly useful in high BER environments where channels are likely
 4650 to fade and where low latency, real-time encrypted speech and data applications are required.

4651 (U//FOUO) The FNBDT protocol is transport-independent in that it is designed to operate over
 4652 whatever lower-layer protocols might be available. Within constraints applicable to specific
 4653 applications (timeouts, speech delay, etc.) the FNBDT protocol can operate over any channel
 4654 capable of transporting bits between two end-user terminals. FNBDT terminal vendors have
 4655 implemented products utilizing PSTN, ISDN, GSM, CDMA, Iridium satellite, digital radio, and
 4656 other channel types for data transport.

4657 2.3.3.2.1.2.1.2.3 (U//FOUO) Risks/Threats/Attacks

4658 (U//FOUO) Since the FNBDT protocol operates at the Application Layer in the network protocol
 4659 stack, risks associated with lower protocol layers are not addressed. Issues such as Traffic Flow
 4660 analysis, LPI/LPD, and DoS must be dealt with outside the bounds of the FNBDT protocols.

4661 2.3.3.2.1.2.1.3 (U//FOUO) Maturity

4662 (U//FOUO) The FNBDT protocol is Mature, in the sense that products have been implemented
 4663 and deployed for several years. Users have real-world experience with FNBDT products—both
 4664 wired and wireless. Additional modes and features will continue to be added to the
 4665 specifications, but the basic interoperable FNBDT modes are mature and will continue to exist
 4666 for some time into the future. The TRL of the basic FNBDT bitstream definition is 9 (Mature -
 4667 products deployed in operational mission conditions).

4668 (U//FOUO) Application of the FNBDT protocol to IP-based transport is less mature. Although
 4669 different vendors are working to apply FNBDT technology to IP networks, there are currently no
 4670 interoperable standards for this specific application. The TRL for using FNBDT over IP
 4671 networks is currently estimated at 4 (Emerging - breadboard validation in lab environment).

4672 2.3.3.2.1.2.1.4 (U) Standards

4673 (U//FOUO) The FNBDT protocols are defined by the standards listed in Table 2.3-4:

4674

Table 2.3-4: (U) FNBDT Standards

This table is (U//FOUO)	
Name	Description
FNBDT-210 (Signaling Plan)	This unclassified specification defines the signaling requirements for FNBDT operational modes. A secure overlay capable of interoperation with FNBDT compatible equipment on various similar or disparate networks is defined. Since the various networks will often have different lower-layer communications protocols, the FNBDT secure overlay specification specifies the higher-layer end-to-end protocols only. Appendices to this specification define operation using specific networks.
FNBDT-230 (Cryptography Specification)	This classified specification outlines details of the cryptography defined for FNBDT. Issues such as key generation, traffic encryption, and compromise recovery are specified in sufficient detail to allow interoperable implementation.
Proprietary extensions	The FNBDT signaling and cryptography specifications define interoperable branch points allowing vendors to implement proprietary modes. This allows vendors to take advantage of the basic FNBDT structure to add modes fulfilling specific needs. Legacy FNBDT implementations have used these branch points to implement custom cryptographic modes. Details of such modes are contained in vendor proprietary specifications.

This table is (U//FOUO)	
Name	Description
Other specifications	Other interoperable FNBDT specifications have been suggested and are currently under consideration by the FNBDT Working Group. These additional documents would provide interoperable ways of implementing additional features such as non-Type 1 operation and key management.
This table is (U//FOUO)	

4675 2.3.3.2.1.2.1.5 (U) Cost/Limitations

4676 (U//FOUO) Although the FNBDT protocol is a good choice for solving many speech-related
 4677 security issues, there are limitations with this protocol as well. Potential limiting factors that
 4678 must be considered when evaluating FNBDT as a candidate protocol for solving security
 4679 problems include:

- 4680 • (U//FOUO) Point-to-point operation. The current definition of FNBDT includes point-to-
 4681 point operation only. There are no provisions in place for multi-user conferencing or net
 4682 broadcast capabilities. The FNBDT Working Group is currently active in defining net
 4683 broadcast modes and Pre-Placed Key (PPK) methods allowing multiple users to decrypt a
 4684 common encrypted bitstream.
- 4685 • (U//FOUO) Voice coders. The FNBDT specifications currently define two voice coders;
 4686 2400 bps MELP and 6400 bps G.729D. FNBDT-compatible speech equipment must
 4687 include one of these vocoders in order to interoperate with FNBDT equipment provided
 4688 by other vendors.
- 4689 • (U//FOUO) Legacy interoperability. FNBDT equipment is not compatible with other
 4690 types of secure voice equipment. Specifically, the older generation STU-III devices that
 4691 have been widely deployed throughout the world during the past 20 years are not
 4692 compatible with the cryptography, speech coders, or wireline modems used by FNBDT
 4693 equipment.
- 4694 • (U//FOUO) Establishing a channel. FNBDT is defined as an application layer technology
 4695 that provides the encrypted bitstream to transfer between two endpoints. The details
 4696 regarding how the digital channel is established between these two endpoints is left
 4697 outside the scope of the FNBDT specifications. As a result, potential users must be aware
 4698 of channel establishment procedures to make sure this process is successful outside the
 4699 bounds of FNBDT.
- 4700 • (U//FOUO) Trusted platform requirement. Application Layer security methods are not
 4701 suitable for operation using general purpose computing equipment. FNBDT and other
 4702 Application Layer security approaches require trusted hardware to support separation
 4703 requirements.

4704 2.3.3.2.1.2.1.6 (U) Dependencies

4705 (U//FOUO) FNBDT cryptography specifications depend on terminals containing appropriate key
 4706 material. The necessary key material is supplied by the Government's Electronic Key
 4707 Management System (EKMS).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

4708 (U//FOUO) The call control process (call establishment, maintenance, teardown, etc.) is not
4709 defined by the FNBDT protocol. These processes, which are a necessary part of a successful
4710 FNBDT voice or data call, must occur outside the scope of the FNBDT specifications.

4711 2.3.3.2.1.2.1.7 (U) Alternatives

4712 (U//FOUO) The most widespread alternative to FNBDT secure speech systems continues to be
4713 the STU-III terminals. These devices, which are based on approximately 20-year old technology,
4714 are no longer produced but are so pervasive throughout the Government that they continue to be
4715 a factor in secure speech system decisions. The Government expects to continue producing key
4716 material to support these terminals through the GIG 2008 Vision timeframe.

4717 (U//FOUO) Other tactical and strategic secure voice system terminals exist in lower quantities.
4718 Systems such as Advanced Narrowband Digital Voice Terminal (ANDVT), MSE, etc. are also
4719 relatively dated but continue to provide acceptable quality encrypted speech communications for
4720 certain specific applications.

4721 (U//FOUO) Depending on specific operational requirements, a speech channel could be
4722 protected at the IP layer (e.g., HAIPE) rather than the Application Layer. This approach, referred
4723 to as Voice over Secure IP rather than Secure Voice over IP, provides an alternative to the
4724 FNBDT Application Layer protection approach for user environments where separation within
4725 an enclave is not a consideration.

4726 2.3.3.2.1.2.1.8 (U) Complementary Techniques

4727 (U//FOUO) Any given user situation may require a combination of technologies in order to meet
4728 all operational requirements. For example, the FNBDT protocol may provide confidentiality at
4729 the Application Layer, but does nothing toward meeting any potential Traffic Flow Security or
4730 TRANSEC requirements at the lower layers. Additional technologies will often need to be used
4731 in combination with the FNBDT protocol in order to meet all applicable security requirements.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

4732 2.3.3.2.1.2.2 (U) Interoperability/Gateways

4733 2.3.3.2.1.2.2.1 (U) Technical Detail

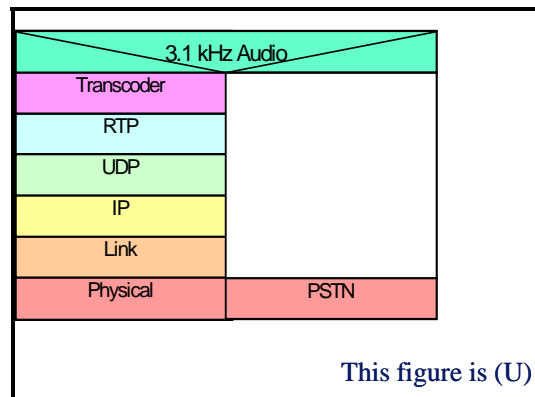
4734 (U//FOUO) Interoperability is an important GIG consideration, both from enclave to enclave
 4735 within the GIG and from GIG resources to infrastructure external to the GIG. Gateways provide
 4736 the necessary interworking and protocol stack adaptation to provide this interoperability.

4737 (U) Gateways adapt the communication needs of different networks such that user data can be
 4738 sent from one to another. Gateways can be described as relay devices; that is, they relay user
 4739 traffic from one protocol stack to another.

4740 (U) Gateways can be grouped according to the specific functions they perform. Some are
 4741 signaling gateways that adapt the call control and other signaling needs of a particular network to
 4742 the signaling needs of a different network. Some are media gateways that adapt user speech from
 4743 one form to another. Signaling and media functions can be combined such that a common device
 4744 provides both functions.

4745 (U//FOUO) Within the GIG architecture, gateways will be necessary both for providing
 4746 interoperability between different vendors VoIP implementations and for providing
 4747 interoperability between packet-switched and circuit-switched networks.

4748 (U) Figure 2.3-11 illustrates the protocol stacks associated with a typical Media Gateway (MG)
 4749 included with a VoIP system for interoperation with legacy PSTN networks. This MG provides a
 4750 termination point for the IP, UDP, and RTP layers, as well as providing a transcoder function.
 4751 The result is audio speech that can be routed to the PSTN.



4752

4753 **Figure 2.3-11: (U) Media Gateway Protocol Stack Illustration**

4754 (U//FOUO) Tactical networks within the GIG may also include gateway functionality to allow
 4755 interoperation with other systems. Like the commercial VoIP gateways, these tactical versions
 4756 contain a protocol stack appropriate to the specific tactical network on one side and a protocol
 4757 stack appropriate to the target network on the other.

4758 (U//FOUO) Although gateways will remain a necessary part of the infrastructure, it is important
 4759 that secure system architectures are designed so that gateways remain Black. This means that
 4760 although the gateways may remove or adapt network protocol stack layers, they must not be
 4761 expected to decrypt user traffic. User traffic must remain encrypted as it traverses the gateway—
 4762 resulting in true end-user to end-user encryption.

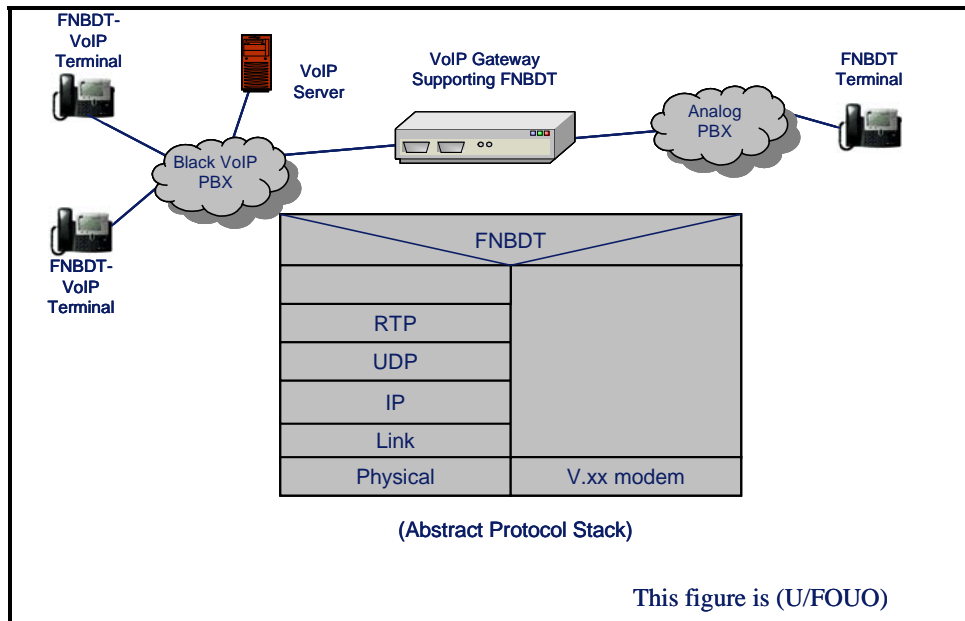
4763 2.3.3.2.1.2.2.2 (U) Usage Considerations

4764 2.3.3.2.1.2.2.2.1 (U) Implementation Issues

4765 (U) Legacy PSTN-based secure voice systems transport bits using a commercial wireline modem
 4766 to modulate digital traffic over the analog PSTN. In order for secure VoIP terminals to
 4767 interoperate with these legacy systems, the gateway must provide a compatible modem function
 4768 on the PSTN side. Although commercial VoIP systems today have recognized the need for
 4769 PSTN Interworking and have included the Media Gateway functionality, there is no commonly
 4770 recognized need to include the modem function in this gateway.

4771 (U//FOUO) Therefore, non-standard gateways are required to allow interworking between secure
 4772 VoIP systems and legacy secure PSTN-based systems. Although gateways containing this
 4773 functionality have not been identified as a requirement in commercial VoIP systems, it is
 4774 important to point out that implementation and maintenance of such a gateway does not
 4775 necessarily need to be carried out by the same vendor that supplies the basic VoIP system. A
 4776 system integrator having access to the IP network on one side and the PSTN on the other could
 4777 insert the required gateway independent of the other infrastructure.

4778 (U//FOUO) The functionality associated with a secure VoIP gateway is shown in Figure 2.3-12.



4779

4780

Figure 2.3-12: (U//FOUO) Secure Voice Gateway Functionality

4781 2.3.3.2.1.2.2.2(U) Advantages

4782 (U) Gateway technology allows interoperation in at least two areas that would not be possible
4783 without gateways:

- 4784 • (U) Operation with legacy equipment on circuit-switched networks
- 4785 • (U) Operation with different technologies within the same user environment

4786 2.3.3.2.1.2.2.2.3(U) Risks/Threats/Attacks

4787 (U//FOUO) The basic risk associated with Black gateway technology is DoS. If an adversary can
4788 gain access to the control mechanisms of the gateway, traffic channels can be blocked such that
4789 users can be kept from using them. There is no additional risk associated with the confidentiality
4790 of the user data since it is not decrypted at the gateway.

4791 2.3.3.2.1.2.2.3 (U) Maturity

4792 (U) Commercial signaling and media gateways are Mature (TRLs 7 – 9) and exist for solving
4793 specific problems within specific bounds. For instance, the gateway technology associated with
4794 interoperating standard non-secure calls between VoIP systems and the PSTN is well understood
4795 and has been implemented in many forms.

4796 (U//FOUO) However, the non-standard variations required for secure voice systems are
4797 Emerging (TRLs 4 – 6). Commercial vendors have not seen a business case for defining and
4798 implementing gateways containing modem functionality as will be required for secure voice
4799 interoperation.

4800 (U//FOUO) Commercial VoIP systems on system-high networks are Mature (TRL 9 - successful
4801 mission operations). Secure Voice variants are Emerging (TRL 5 -breadboard evaluation in
4802 relevant environment).

4803 2.3.3.2.1.2.2.4 (U) Standards

4804 (U) The following standards are used for gateway control in VoIP systems:

- 4805 • (U) MEGACO, also referenced as Gateway Control Protocol (GCP). RFC 3525, formerly
4806 RFC 3015. Also published by the ITU-T as Recommendation H.248.1
- 4807 • (U) Media Gateway Control Protocol (MGCP), RFC 3435.

4808 2.3.3.2.1.2.2.5 (U) Cost/Limitations

4809 (U//FOUO) Use of commercial media gateways is a cost-effective approach for VoIP systems
4810 that provide security by residing on system-high networks. For VoIP systems that require
4811 FNBDT security there are at least two options to provide the necessary gateways:

- 4812 • (U//FOUO) Dedicated special-purpose gateway that leaves out the transcoder function
4813 and includes the modem function
- 4814 • (U//FOUO) Modifications to commercial gateways to allow a client to bypass the
4815 transcoder in the gateway and route the information through a modem instead

4816 (U//FOUO) Either of these options will result in additional complexity and associated cost.

4817 2.3.3.2.1.2.2.6 (U) Dependencies

4818 (U//FOUO) Gateway technology is highly dependent on the specific systems a particular
4819 gateway is providing interoperability between. A gateway is designed to be completely
4820 compatible with a particular system on each side. If a third system is introduced into the
4821 architecture, it is highly likely that a separate gateway will be required.

4822 2.3.3.2.1.2.3 Secure Voice over IP

4823 (U//FOUO) Secure VoIP technologies described here secure the voice bearer, or user voice
4824 packets. Secure VoIP call or session control used to establish calls is addressed in section
4825 2.3.3.2.2.2.1.

4826 2.3.3.2.1.2.3.1 (U) Technical Detail

4827 (U//FOUO) Security technologies considered for VoIP voice packets include:

- 4828 • (U) Secure Real-Time Protocol (SRTP)
- 4829 • (U//FOUO) FNBDT over RTP
- 4830 • (U//FOUO) Secured voice, such as FNBDT, over V.150 Modem Relay Simple Packet
4831 Relay Transport protocol (SPRT).

4832 (U//FOUO) A brief introduction to the VoIP technology is presented before a description of
4833 potential security technologies. (VoIP call control is described in section 2.3.3.2.2.2.1.) VoIP
4834 architectures typically include control planes to set up VoIP calls and execute network services.
4835 They also include bearer planes used to transfer voice packets between users after the call has
4836 been established. H.323 and SIP are the leading protocol systems used for VoIP call control.
4837 Other notable VoIP protocols, specifically MGCP and GCP/H.248/MEGACO reside between
4838 control and bearer planes. They are used when VoIP-PSTN Gateways and Multimedia
4839 conference units are decomposed into Media Gateway Controllers (MGCs) and Media Gateways
4840 (MGs). MGCs use protocols such as MGCP to control the bearer path through MGWs.

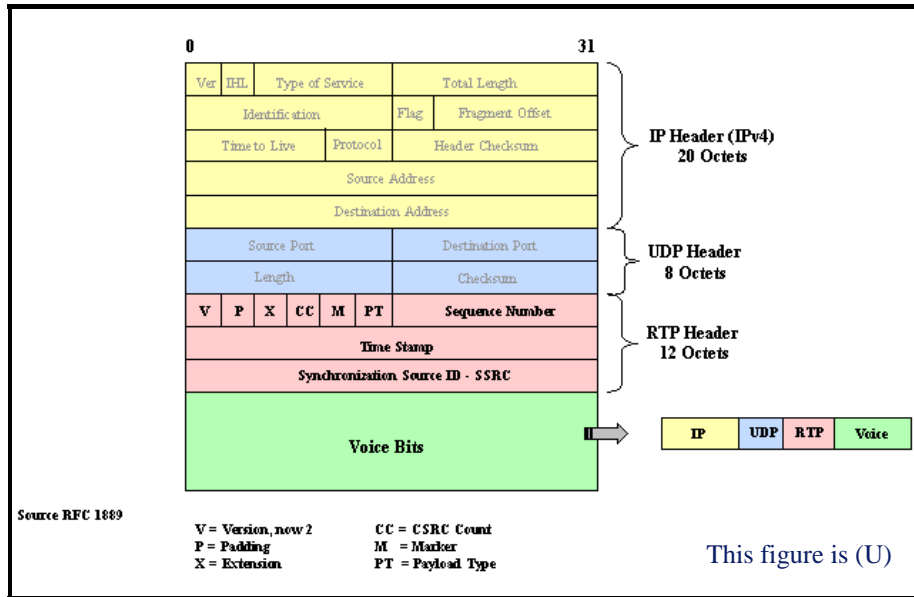
4841 (U//FOUO) QoS protocols and systems, such as RSVP and DiffServ, are complementary
4842 technologies needed to support VoIP services, but are not call control protocols themselves. QoS
4843 should be established or negotiated outside of the voice bearer plane as part of the overall call set
4844 up process, and subsequently applied to the actual voice stream packets. Security mechanisms
4845 are needed to protect QoS mechanisms, but such are outside the scope of this section.

4846 (U) RTP is used in all common VoIP systems to transport voice packets between users. A closer
4847 look at RTP follows.

4848 2.3.3.2.1.2.3.1.1 (U) RTP and RCTP Overview

4849 (U) Real-Time Transport Protocol is designed to transport real-time applications over IP
4850 networks. RTP runs in conjunction with RTCP (RealTime Control Protocol), which provides
4851 feedback to applications about the quality of the media transmission.

4852 (U) RTP provides time-stamping and Sequence Numbering of the Multimedia packets to enable
4853 synchronization of a received media stream. As shown in Figure 2.3-13, RTP, along with RTCP,
4854 reside on UDP. Reliability mechanisms such as re-transmits are not included since latency and
4855 jitter are more important to voice quality than bit errors or occasional voice packet losses. A
4856 description of the fields within the RTP header follows the figure.



4857

4858

Figure 2.3-13: (U) Real-Time Protocol

4859 (U) Version (V): 2 bits - This field identifies the version of RTP. The current version is two (2).

4860 (U) Padding (P): 1 bit - If the padding bit is set, the packet contains one or more additional
4861 padding octets at the end that are not part of the payload. Padding may be needed by some
4862 encryption algorithms with fixed block sizes or for carrying several RTP packets in a lower-layer
4863 protocol data unit.

4864 (U) Extension (X): 1 bit - If the extension bit is set, the fixed header is followed by exactly one
4865 header extension.

4866 (U) CSRC count (CC): 4 bits - The (CSRC) count contains the number of CSRC identifiers that
4867 follow the fixed header. CSRCs are media contributors that reside behind a conference unit.

4868 (U) Marker (M): 1 bit - The interpretation of the marker is defined by a profile. It is intended to
4869 allow significant events such as frame boundaries to be marked in the packet stream. A profile
4870 may define additional marker bits or specify that there is no marker bit by changing the number
4871 of bits in the payload type field.

4872 (U) Payload type (PT): 7 bits This field identifies the format of the RTP payload and determines
4873 its interpretation by the application. A profile specifies a default static mapping of payload type
4874 codes to payload formats. An RTP sender emits a single RTP payload type at any given time;
4875 this field is not intended for multiplexing separate media streams.

4876 (U) Sequence number: 16 bits - The sequence number increments by one for each RTP data
4877 packet sent. This number can be used by the receiver to detect packet loss and to restore packet
4878 sequence. The initial value of the sequence number is random (unpredictable) to make known-
4879 plaintext attacks on encryption more difficult, even if the source itself does not encrypt, because
4880 the packets may flow through a translator that does.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

4881 (U) Time-stamp: 32 bits - The time-stamp reflects the sampling instant of the first octet in the
4882 RTP data packet. The sampling instant must be derived from a clock that increments
4883 monotonically and linearly in time to allow synchronization and jitter calculations.

4884 (U) SSRC: 32 bits - The Synchronization Source Real-time Content (SSRC) field identifies the
4885 synchronization source. This identifier is chosen randomly, with the intent that no two
4886 synchronization sources within the same RTP session will have the same SSRC identifier.

4887 (U) CSRC list: 0 to 15 items, 32 bits each - The Contributing Source Real-time Content (CSRC)
4888 list identifies the contributing sources for the payload contained in this packet.

4889 (U) RTCP runs in conjunction with RTP. Receiving participants send periodic information, using
4890 RTCP, back to the originating source to convey quality information about the received data.
4891 RTCP provides the following services:

4892 • (U) Quality Monitoring and Congestion Control – This is the primary function of RTCP,
4893 and it provides feedback to senders about the quality of the connection. The sender can
4894 use this information to adjust transmission. Also, third party monitors can use the
4895 information to access network operation.

4896 • (U) Source Identification – The source field in the RTP header is a 32 bit identifier,
4897 randomly generated, and not user friendly. RTCP provides a more user friendly
4898 identification of the 32 bit RTP source field by providing a global identification of
4899 session participants. This information is typically, user name, telephone number, email
4900 address, etc.

4901 • (U) Inter-Media Synchronization – RTCP sends information that can be used to
4902 synchronize audio and video packets.

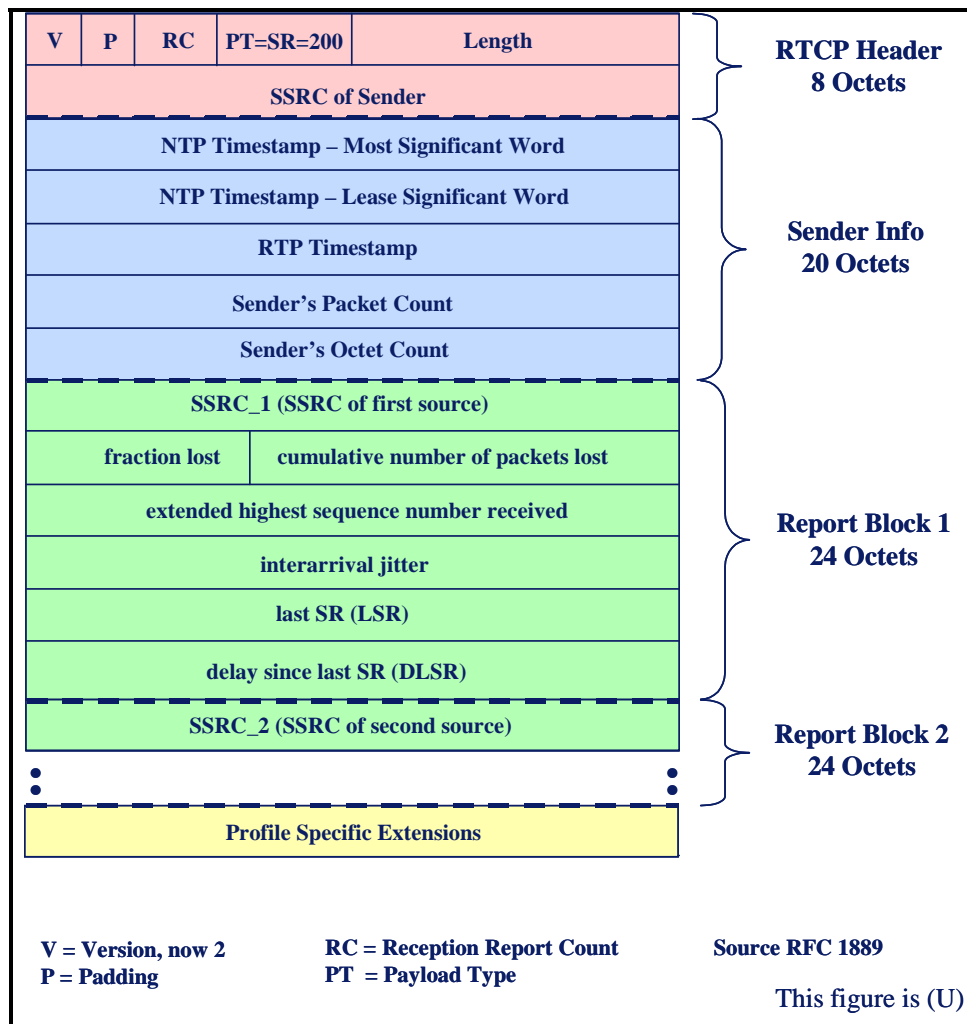
4903 • (U) Control Information Scaling – As the number of participants increase, the amount of
4904 control information must be scaled down to allow sufficient bandwidth for the RTP
4905 channels. This is done by the RTP protocol by adjusting the RTCP generation rate.
4906 Typically, the control bandwidth is limited to 5% of the RTP traffic.

4907 (U//FOUO) Each RTCP packet begins with a fixed part similar to that of RTP data packets,
4908 followed by structured elements that may be of variable length according to the packet type. The
4909 alignment requirement and a length field in the fixed part are included to make RTCP packets
4910 stackable. Multiple RTCP packets may be concatenated without any intervening separators to
4911 form a compound RTCP packet that is sent in a single packet of the lower layer protocol, such as
4912 UDP. There is no explicit count of individual RTCP packets in the compound packet since the
4913 lower layer protocols are expected to provide an overall length to determine the end of the
4914 compound packet.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

4915 (U//FOUO) Figure 2.3-14 displays the format of one of the five RTCP messages—the RTCP
 4916 Send Report. RTP receivers provide reception quality feedback using RTCP report packets
 4917 which may take one of two forms depending upon whether or not the receiver is also a sender.
 4918 The only difference between the sender report (SR) and receiver report (RR) forms, besides the
 4919 packet type code, is the sender report includes a 20-byte sender information section active
 4920 senders can use.

4921 (U//FOUO) It is expected that reception quality feedback will be useful not only for the sender
 4922 but also for other receivers and third-party monitors. The sender might modify its transmissions
 4923 based on the feedback. Receivers can determine whether problems are local, regional or global.
 4924 Network managers can use profile-independent monitors that receive only the RTCP packets and
 4925 not the corresponding RTP data packets to evaluate the performance of their networks for
 4926 multicast distribution.



4927

4928

Figure 2.3-14: (U) RTCP Sender Report Format- Sender Report (SR)

4929 **(U) Secure RTP (SRTP) and Secure RTCP (SRTCP) per RFC 3711**

4930 (U//FOUO) SRTP/SRTCP are used to protect the RTP/RTCP protocols. SRTP supports both IP
4931 unicast and multicast communications. SRTP is a commercial security system and no type 1
4932 versions are available. Therefore, SRTP can be used within a security domain, but without
4933 further development is not advised to use to secure voice traffic across separate security
4934 environments. Therefore, another lower layer protocol, such as IPsec, should be used to transport
4935 secure voice across security domains.

4936 (U//FOUO) Secure RTP is used to authenticate and protect RTP headers and payloads. It is
4937 defined as an extension to the RTP Audio/Video profile per RFC 3551. Each SRTP stream is
4938 organized around cryptographic contexts that senders and receivers use to maintain
4939 cryptographic state information. The cryptographic context is uniquely mapped to the
4940 combination of:

- 4941 • (U) The destination IP address
- 4942 • (U) The destination port
- 4943 • (U) The SSRC (as seen in the RTP header).

4944 (U//FOUO) SRTP session keys are cryptographically derived per the RFC from master keys and
4945 salt keys that are initialized via key management. The salt keys are updated per the RFC for use
4946 in subsequent session key derivations. The session keys are then applied to the voice media
4947 stream to provide encrypted voice.

4948 (U//FOUO) SRTP does not define or mandate a specific key management protocol. However, it
4949 does place requirements and considerations on the key management system. These
4950 considerations are described in the dependencies section below.

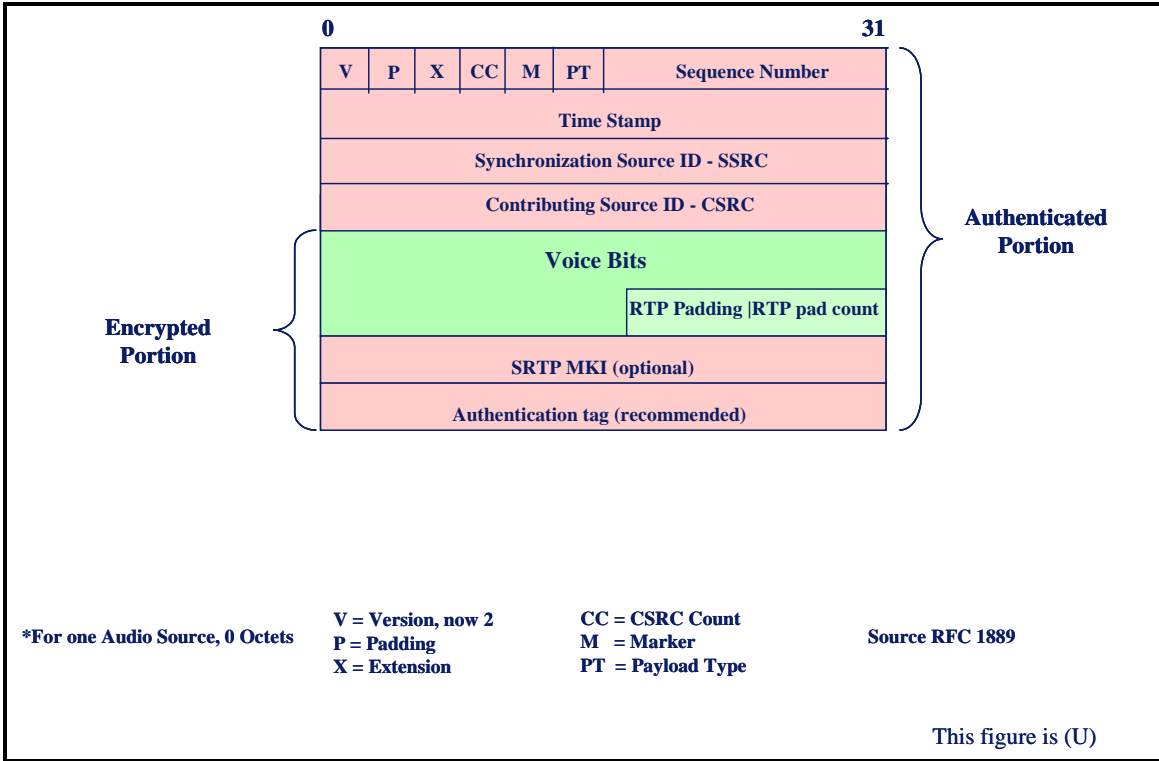
4951 (U) The SRTP Protocol

4952 (U//FOUO) Figure 2.3-15 illustrates the format of SRTP.

4953 (U//FOUO) As can be seen from Figure 2.3-15, the SRTP format uses the RTP header and RTP
4954 payload, followed by the SRTP MKI and Authentication tag fields. The entire SRTP packet is
4955 authenticated while the RTP payload and SRTP MKI and authentication tags are encrypted.

4956 (U//FOUO) The optional SRTP MKI (Master Key Identifier) field identifies the master key used
4957 in the session. It does not indicate cryptographic context. The MKI is defined and distributed by
4958 the key management system.

4959 (U//FOUO) The SRTP authentication tag is used to carry message authentication data. The tag
4960 provides authentication of the RTP header and payload and indirectly provides replay protection
4961 by authenticating the RTP sequence number. The tag is recommended for use by the RFC.



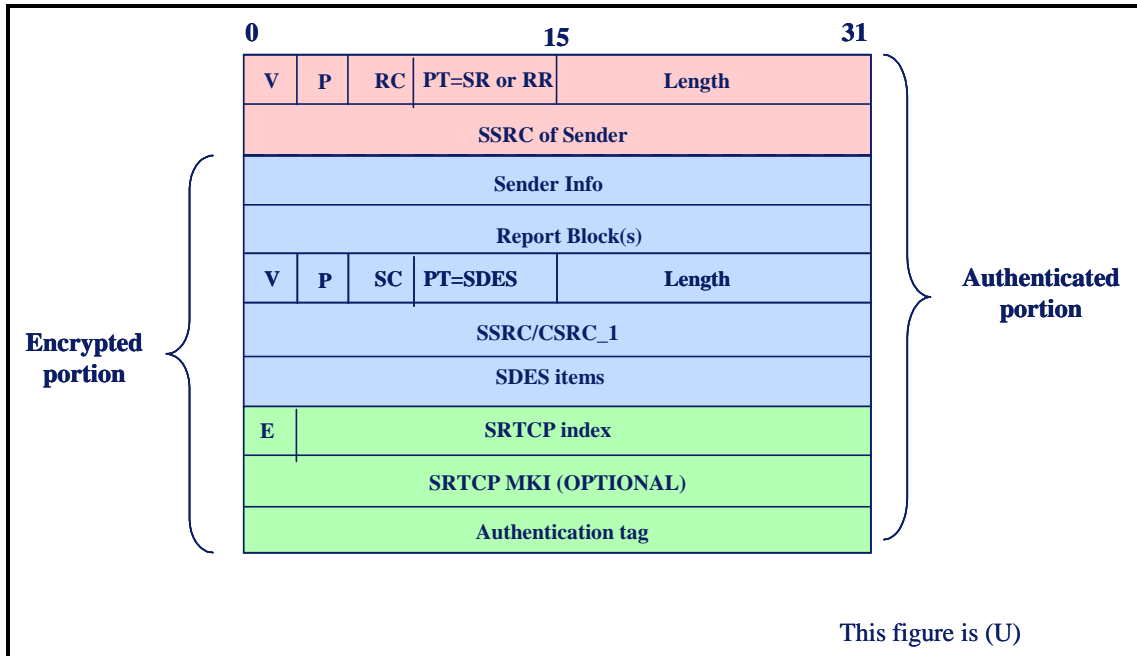
4962

4963

Figure 2.3-15: (U) SRTCP Format

4964 (U) The SRTCP protocol:

4965 (U//FOUO) Figure 2.3-16 illustrates the format of SRTCP:



4966

4967

Figure 2.3-16: (U) SRTCP Format

UNCLASSIFIED//FOR OFFICIAL USE ONLY

4968 (U//FOUO) As can be seen from Figure 2.3-16, the SRTCP format uses the RTCP header and
4969 RTCP payload reports, followed by the SRTP 'E', SRTCP index, SRTCP MKI, and
4970 Authentication tag fields. The entire SRTCP packet is authenticated while the RTCP payload is
4971 encrypted along with SRTCP specific fields. (Note that Figure 2.3-16 shows a generalized
4972 representation of the RTCP reports, but this does not impact the discussion of SRTCP fields.)

4973 (U//FOUO) The 'E' field is a one-bit flag that indicates if the current RTCP packet is encrypted.

4974 (U//FOUO) The SRTCP index is a 31-bit counter for the SRTCP packet. It is initially set to zero
4975 before the first SRTCP is sent and incremented by one after each SRTCP packet. It is not reset to
4976 zero after a rekey event to help provide replay protection.

4977 (U//FOUO) The optional SRTCP MKI field indicates the master key used to derive the session
4978 key for the RTCP context.

4979 (U//FOUO) The SRTCP authentication tag is used to carry message authentication data. The tag
4980 provides authentication of the RTCP header and payload. The tag is recommended for use by the
4981 RFC.

4982 (U) Encryption algorithms

4983 (U//FOUO) SRTP calls out AES in counter mode for encryption and HMAC-SHA1 for message
4984 authentication and integrity. The RFC explicitly states that any transforms added to SRTP must
4985 be added with a companion standard track RFC that exactly defines how the transform is used
4986 with SRTP.

4987 (U) FNBDT over RTP

4988 (U//FOUO) The second technology considered for secure voice is to create an RTP payload type
4989 for FNBDT type 1 secured voice. Please refer to section 2.3.3.2.1.2.1 for a description of
4990 FNBDT. The new RTP profile is defined to support both FNBDT signaling and data. This
4991 FNBDT media type needs to be identified and negotiated between clients within the GIG VoIP
4992 call control architecture. The RTP protocol field described above contains a GIG unique payload
4993 value indicating FNBDT to GIG users. In such a scenario, a client could start a clear voice call
4994 using an Internet Assigned Numbers Authority (IANA) standard RTP payload type (voice codec)
4995 and then use call control signaling to transition to the FNBDT profile.

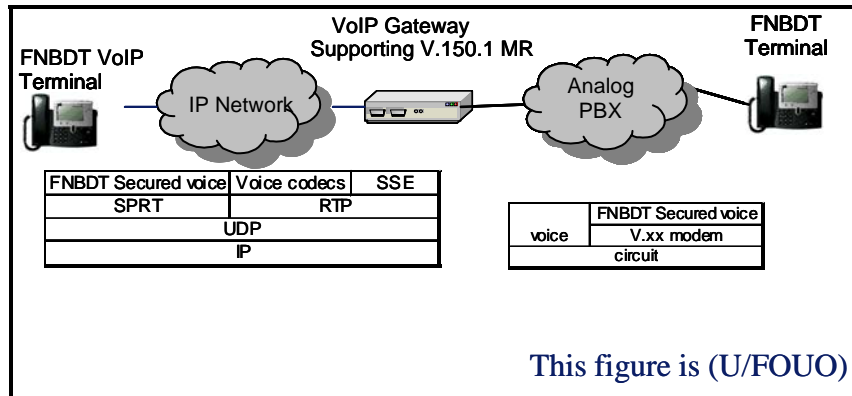
4996 (U//FOUO) Note that certain FNBDT modes add overhead to the clear voice codec approaches
4997 in order to maintain crypto-synchronization. Differences in network resource requirements when
4998 transitioning between clear and secured FNBDT voice would need to be accounted for in the
4999 GIG QoS architecture.

5000 (U) Secured voice, such as FNBDT encryption, over V.150.1 Modem Relay

5001 (U//FOUO) Secure voice, such as FNBDT encryption, over V.150.1 modem relay applies type 1
5002 secured voice over a commercial standard real-time transport mechanism for data. Please refer to
5003 section 2.3.3.2.1.2.1 for a description of FNBDT. The following is a brief overview of V.150.1
5004 modem relay.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5005 (U//FOUO) ITU specification V.150.1 Modem Relay, hereafter referred to V.150.1 MR, is a
 5006 VoIP-PSTN gateway feature. It is designed to allow the successful transfer of data from a
 5007 modem on a circuit network, through a PSTN-VoIP GW and across an IP network, through a
 5008 second VoIP-PSTN GW and back onto a second modem over circuit network. FNBDT secure
 5009 voice over V.150.1 MR exploits the V.150.1 SPRT protocol as illustrated in Figure 2.3-17
 5010 below.

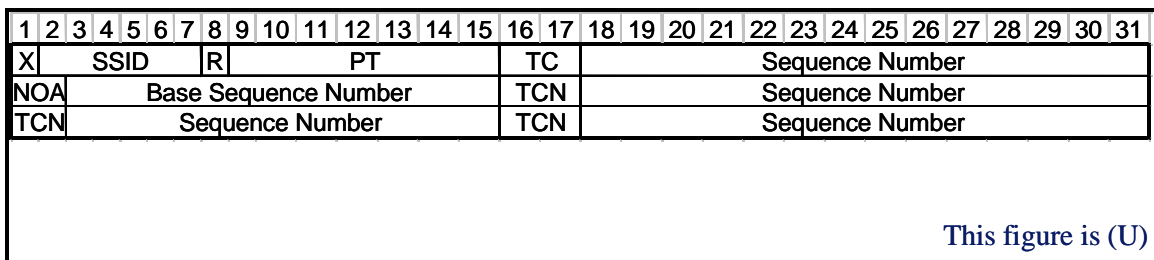


5011

5012 **Figure 2.3-17: (U//FOUO) FNBDT over V.150.1 Modem Relay**

5013 (U//FOUO) V.150.1 supports several modes that allow a user to multiplex between voice and
 5014 data transport. As can be seen from Figure 2.3-17, V.150.1 enables clear voice and the V.150.1
 5015 defined State Signaling Protocol to be carried over RTP. SSE is used to transition between voice
 5016 and modem data transport modes at a V.150.1 PSTN-VoIP GW. As such, State Signaling Event
 5017 (SSE) instructs the GW to initiates a V.xx modem on the circuit network in preparation of
 5018 making a voice to data transition. Data bearer, however, is carried over the IP network with the
 5019 V.150.1 Simple Packet Relay Transport protocol, SPRT. SPRT is placed on UDP and includes
 5020 both reliable and transparent sequenced modes. FNBDT secured voice is carried over the SPRT
 5021 transparent sequenced mode, which is designed to support real-time data. SPRT includes
 5022 sequence numbers to facilitate proper packet ordering at receivers. As such, V.150.1 MR is able
 5023 to transport type 1 secured voice between VoIP and circuit networks using a V.150.1 commercial
 5024 standard VoIP GW.

5025 (U) Figure 2.3-18 below illustrates the SPRT format:



5026

5027 **Figure 2.3-18: (U) V.150.1 Simple Packet Relay Transport for IP networks**

5028 (U) The SPRT header fields are summarized as follows:

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 5029 (U) X = Header Extension Bit, currently reserved by ITU
- 5030 (U) SSID = SubSession ID, used to identify a media stream
- 5031 (U) R = reserved by ITU
- 5032 (U) PT = payload type. The payload is set to the value assigned by the media stream by call
5033 control signaling. Note that the R and PT field together are consistent with the same fields seen
5034 in the RTP header such that clients and GWs can transition between voice/RTP and data/SPRT
5035 over the same UDP port.
- 5036 (U//FOUO) TC – Transport Channel number. FNBDT uses transport channel 3, designed for
5037 real-time data without acknowledgements.
- 5038 (U//FOUO) The sequence number field is incremented with each SPRT packet, similar to the
5039 sequence number in RTP.
- 5040 (U//FOUO) NOA – Number of Acknowledgments. Acknowledgments are set to zero for FNBDT
5041 and other real-time data services.
- 5042 (U//FOUO) The Base Sequence number field is used to manage re-transmits. It is set to zero for
5043 TC= 3, the channel used by FNBDT.
- 5044 (U//FOUO) TCN and subsequent sequence number fields are used for re-transmits. These fields
5045 are not used with FNBDT over V.150.1 MR.
- 5046 (U//FOUO) In summary, an FNBDT over V.150.1 client can first establish a clear voice call and
5047 send clear voice via RTP. It can then use SSE to transition to data mode. Once in data mode, the
5048 client then used SPRT to transport FNBDT signaling and secured voice.
- 5049 2.3.3.2.1.2.3.2 (U) Usage Considerations
- 5050 2.3.3.2.1.2.3.2.1 (U) Implementation Issues
- 5051 (U) SRTP
- 5052 (U//FOUO) Each media stream in a multimedia session requires its own SRTP session key. This
5053 multiplies the potential number of security contexts initiated per user. This is a concern for
5054 mobile multimedia services with limited battery and processing power. More security contexts
5055 could also multiply the amount of key management traffic.
- 5056 (U//FOUO) Forward error correction and interleaving, if required by the RTP media type, need
5057 to be completed before application of SRTP.
- 5058 (U//FOUO) SRTP cannot span into non-IP networks, such as the PSTN or DSN. Therefore, a
5059 VoIP-PSTN GW would need to terminate SRTP and invoke another security mechanism for the
5060 PSTN side of a VoIP to PSTN call. This requires a complex, Red GW function.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5061 (U//FOUO) Note that SRTP can be used for end-to-end security in half duplex voice conferences
5062 using multicast. But full duplex conferences require a Red conference unit, either at each client
5063 or in a central server.

5064 (U) FNBDT over RTP

5065 (U//FOUO) The custom RTP profile developed for RTP complicates the use of existing VoIP
5066 call control mechanisms, which will need to be extended for this unique media type. It should
5067 also be noted that the RTP header time stamp is not used as originally intended since the RTP
5068 payload contains a mixture of FNBDT security signaling besides ciphered voice. FNBDT over
5069 RTP does not fit within conventional VoIP-circuit GWs. A custom GW would be required for
5070 FNBDT over RTP. See section 2.3.3.2.1.2.2 for further discussion of GW topics.

5071 (U) FNBDT over V.150.1 Modem Relay

5072 (U//FOUO) V.150.1 MR is defined for PSTN-IP-PSTN scenarios and as such is a GW
5073 architectural element. Therefore, this GW function would need to be collapsed into a voice client
5074 for application in an all IP environment. This may be considered complex for a mobile user
5075 device. Since V.150.1 MR is not a widely used transport mechanism in IP networks, it
5076 potentially introduces a new network transport mechanism in the GIG specifically for secure
5077 voice.

5078 2.3.3.2.1.2.3.2.2(U) Advantages

5079 (U//FOUO) SRTP fits well within conventional VoIP architectures and promises to become a
5080 widely known commercial standard. It is less complex than other approaches when used in an all
5081 IP network of a single-security domain.

5082 (U//FOUO) FNBDT is a proven type 1 protocol. The use of RTP fits within conventional VoIP
5083 architectures, although it is extended for the non-standard FNBDT media type. But FNBDT over
5084 RTP would require custom black circuit-VoIP GWs.

5085 (U//FOUO) V.150.1 MR can be used within an all IP network as well across black V.150.1
5086 PSTN GWs to legacy FNBDT devices. As such, it offers the potential to provide type 1 security
5087 from a VoIP to a PSTN-based terminal as it leverages an established type 1 security protocol.

5088 2.3.3.2.1.2.3.2.3(U) Risks/Threats/Attacks

5089 (U) SRTP

5090 (U//FOUO) SRTP does support type 1 algorithms without extending the protocol with a new
5091 standards track RFC. As such, it should not be used to transport secure voice across security
5092 domain boundaries.

5093 (U//FOUO) RTP uses the 16-bit RTP header sequence number to help set the KG state. Use of
5094 the RTP sequence number to set KG state may not be sufficient for type 1 algorithms. The RTP
5095 header used is subject to manipulation, although the SHA-1 authentication mechanism provides
5096 at least partial protection.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5097 (U//FOUO) SRTP would also require custom, red VoIP- circuit GWs. Since SRTP requires Red
5098 GWs to reach circuit networks, it may not be the leading security protocol candidate for secure
5099 voice.

5100 (U) FNBDT over RTP

5101 (U//FOUO) There are no risks, threats, or attacks unique to FNBDT over RTP identified that are
5102 not common to any type 1 application transferred over an RTP/UDP/IP stack. Specifically, the
5103 IP, UDP, and RTP headers are not protected nor authenticated.

5104 (U) FNBDT over V.150

5105 (U//FOUO) There are no risks, threats or attacks unique to FNBDT over V.150.1 identified that
5106 are not common to any type 1 application transferred over a SPRT/UDP/IP stack. Specifically,
5107 the IP, UDP and SPRT headers are not protected nor authenticated.

5108 2.3.3.2.1.2.3.3 (U) Maturity

5109 (U) SRTP

5110 (U//FOUO) SRTP is Emerging with an estimated TRL of 4 since it is well specified and released
5111 as an RFC, dated March 2004. It is assumed that portions of the function have been demonstrated
5112 with experimental code by SRTP within the IETF community. SPRT products are widely
5113 available in commercial products.

5114 (U//FOUO) Areas of further study and development are recommended before SRTP can be used
5115 within the GIG—specifically:

- 5116 • (U//FOUO) A Key management system that incorporates SRTP requirements needs to be
5117 defined and developed
- 5118 • (U//FOUO) A concept of operations that describes how SRTP is supported within the
5119 GIG call/session control architecture needs to be developed
- 5120 • (U//FOUO) Methods to transition between clear and secure voice within a single session
5121 using SRTP need to be defined and developed
- 5122 • (U//FOUO) An evaluation of how SRTP might evolve to support type 1 security might be
5123 considered
- 5124 • (U//FOUO) Performance evaluation and prediction of SRTP within mobile environments
5125 should be addressed.

5126

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5127 (U) FNBDT over RTP

5128 (U//FOUO) FNBDT and RTP are both Mature with a TRL of 9, since they have been proven in
5129 multiple product deployments. But use of an FNBDT RTP profile or media type is new and has
5130 progressed little past laboratory demonstration. As such, we consider the combination of FNBDT
5131 and RTP to be Emerging with a TRL of 4.

5132 (U//FOUO) Areas of further study and development are recommended before FNBDT over RTP
5133 can be used within the GIG—specifically:

- 5134 • (U//FOUO) FNBDT rekey over IP needs to be developed
- 5135 • (U//FOUO) A concept of operations that describes how FNBDT over RTP is supported
5136 within the GIG call/session control architecture needs to be developed
- 5137 • (U//FOUO) Methods to transition between clear and secure voice within a single session
5138 using FNBDT over RTP need to be defined and developed
- 5139 • (U//FOUO) FNBDT over RTP is currently defined for point-to-point communications.
5140 An evaluation of how FNBDT and RTP constructs could be extended to support
5141 multicast communications is advised
- 5142 • (U//FOUO) Performance evaluation and prediction of FNBDT over RTP within mobile
5143 environments should be addressed

5144 (U) FNBDT over V.150.1

5145 (U//FOUO) FNBDT is a Mature secure technology as merits a TRL of 9.

5146 (U//FOUO) V.150.1 MR is a new, immature transport technology that may not be widely used or
5147 supported in the commercial world. Several sections in the ITU are labeled, ‘to be defined,’ and
5148 standard evolution is likely. Even so, a commercial manufacturer has demonstrated V.150.1 MR
5149 capability and is likely to offer the function in commercial products. For this reason, FNBDT
5150 over V.150.1 is Emerging (TRL 4).

5151 (U//FOUO) Areas of further study and development are recommended before V.150.1 MR can
5152 be used within the GIG—specifically:

- 5153 • (U//FOUO) FNBDT rekey over IP needs to be defined and developed
- 5154 • (U//FOUO) A concept of operations that describes how V.150.1 media type is supported
5155 within the GIG call/session control architecture needs to be developed
- 5156 • (U//FOUO) Methods to transition between clear and secure voice within a single session
5157 using RTP – V.150.1 (SPRT) session need to be defined and developed
- 5158 • (U//FOUO) FNBDT over V.150.1 is currently defined for point-to-point
5159 communications. An evaluation is advised on how FNBDT and V.150.1 constructs could
5160 be extended to support multicast communications

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- (U//FOUO) Performance evaluation and prediction of V.150.1 within mobile environments should be addressed.

2.3.3.2.1.2.3.4 (U) Standards

Table 2.3-5: (U) Secure Voice over IP Standards

This table is (U//FOUO)	
Name	Description
FNBBDT-210	Signaling Plan Revision 2.0
ITU V.150	Procedures for the end-to-end connection of V-series DCEs over and IP network
RFC 3550	RTP: A Transport Protocol for Real-Time Applications
RFC 3711	The Secure Real-time Transport Protocol (SRTP)
This table is (U//FOUO)	

2.3.3.2.1.2.3.5 (U) Cost/Limitations

(U//FOUO) SRTP cannot be used with COTS PSTN GWs to reach secure voice devices on the PSTN or DSN. SRTP must be terminated at the GW. This lack of PSTN interoperability: (1) can complicate migration plans, (2) might restrict mobile GIG user communications in less developed countries and (3) can restrict secure voice with less developed coalition partners. A custom Red PSTN GW is required.

(U//FOUO) FNBBDT over RTP has the same restriction as SRTP. It cannot be used with COTS PSTN GWs to reach secure voice devices on the PSTN or DSN. A custom Black PSTN GW is required. Furthermore, FNBBDT currently does not support multicast or groups call. FNBBDT standards development is required for group calling.

(U//FOUO) V.150.1 may not be widely used in the commercial market. It might be used exclusively for secure voice within the GIG. As such, it is a network transport mechanism that may not enjoy economies of scale as other approaches might. Furthermore, V.150.1 was not defined for multicast groups, so a concept of operations for secure group calls utilizing V.150.1 needs to be developed.

(U//FOUO) FNBBDT currently does not support multicast, or group calls. Further FNBBDT standards development is required.

2.3.3.2.1.2.3.6 (U) Dependencies

(U) SRTP

(U) Key Management Dependencies and interaction

(U//FOUO) SRTP places a number of dependencies upon key management. Section 8.2 of the RFC details a list of parameters the key management system provides including:

- (U//FOUO) Master key parameters for an SSRC
- (U//FOUO) Salt keys parameters for an SSRC

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 5189 • (U//FOUO) Initial RTP sequence number and other crypto context index parameters
5190 (optional)

5191 (U//FOUO) Clients will need to account for the amount of traffic protected with a single master
5192 key and request a rekey from the key management system based on specific usage criteria. The
5193 key management system can, of course, push keys to SRTP clients.

5194 (U) QoS management

5195 (U//FOUO) Network QoS mechanisms suitable for VoIP and RTP should be sufficient for SRTP.

5196 (U) FNBDT over RTP and FNBDT over V.150.1 MR

5197 (U//FOUO) FNBDT has specific key management requirements and specifications and currently
5198 supported with deployed facilities. These facilities may need to be upgraded to meet GIG needs.

5199 (U) QoS management

5200 (U//FOUO) Network QoS mechanisms need to be developed that take into account the resource
5201 utilization of FNBDT, particularly between clear and secure voice transitions.

5202 2.3.3.2.1.2.3.7 (U) Alternatives

5203 (U//FOUO) IP layer security could be used as an alternative to SRTP and FNBDT over RTP.

5204 2.3.3.2.1.2.3.8 (U) Complementary Techniques

5205 (U//FOUO) Type 1 IPsec is needed to secure SRTP protected voice traffic across security
5206 domains.

5207 **2.3.3.2.2 (U) Transport & Network Layer Technologies**

5208 **2.3.3.2.2.1 (U) Non-Real-Time Data Technologies**

5209 2.3.3.2.2.1.1 (U) IP Layer Security

5210 (U//FOUO) IP layer security enables the Black Core concept allowing IP layer routing and sub-
5211 network layer switching to occur on the Black Core, whereas traditional link security required all
5212 routers and switches to be Red.

5213 2.3.3.2.2.1.1.1 (U) Technical Detail

5214 (U//FOUO) Commercial IPsec can be used to protect SBU traffic, and HAIPE can be used to
5215 protect classified traffic. HAIPE was originally based on the existing commercial IPsec
5216 standards, but has shifted from these standards to provide the higher level of security necessary
5217 in a Type-1 application.

5218 (U//FOUO) Commercial IPsec defines separate protocols for confidentiality and authentication,
5219 but the confidentiality protocol (ESP) provides an optional authentication mechanism. HAIPE
5220 requires authentication as well as confidentiality.

5221 (U//FOUO) Commercial IPsec has defined both a transport mode and a tunnel mode and is
5222 intended to support both end system and intermediate system implementations. HAIPE has
5223 defined only a tunnel mode and is intended to support only intermediate system (INE)
5224 implementations. The GIG vision is to migrate HAIPE back to end system implementations, and
5225 consequently some modifications will be required to HAIPE to achieve this vision.

5226 (U//FOUO) HAIPE v1 supported IPv4 only, and HAIPE v2 is intended to support both IPv4 and
5227 IPv6. The GIG vision is to migrate to IPv6.

5228 (U//FOUO) HAIPE supports a Security Policy Database (SPD) to control the flow of IP
5229 datagrams. HAIPE supports selectors such as source/destination addresses (IPv4 and IPv6) to
5230 map IP datagram traffic to policy in the SPD. Each entry specifies the relevant selectors and
5231 whether data should be tunnel-mode encrypted or discarded. If an SPD entry cannot be found for
5232 an IP datagram, the IP datagram is discarded. Entries in the SPD are ordered. The SPD can be
5233 managed locally by the administrator/operator HMI or remotely from the SMW.

5234 (U//FOUO) HAIPE also supports a Security Association Database (SAD). The SPD is consulted
5235 in formation of SA entries in the SAD during an Internet Key Exchange (IKE). Separate distinct
5236 SAs are used for inbound and outbound traffic. The two SAs use the same Traffic Encryption
5237 Key (TEK), but have different SPI values. Entries in the SAD are not ordered. The SAD is
5238 consulted in the processing of all traffic including non-IPsec traffic (i.e., bypassed/regenerated
5239 traffic as well as traffic encrypted in tunnel mode).

5240 (U//FOUO) HAIPE utilizes the ESP tunnel mode to provide data integrity, anti-replay protection,
 5241 confidentiality, and authentication. The original Red IP datagram is encapsulated with the
 5242 HAIPE ESP protocol and then a Black IP protocol, as shown in Table 2.3-6. Table 2.3-6
 5243 indicates a total overhead of 83 octets (or 664 bits) for each Red datagram (assuming Black IP is
 5244 v4). The HAIPE trailer padding includes both crypto padding and TFS padding. Crypto padding
 5245 varies from 0-47 octets (HAIPE supports crypto block sizes of 4, 8, and 48 octets), and an
 5246 average value of 23 octets is assumed for the overhead calculation in Table 2.3-6. No TFS
 5247 padding is assumed in the overhead calculation in Table 2.3-6. Of course, the addition of TFS
 5248 padding would increase the overhead.

Table 2.3-6: (U//FOUO) HAIPE ESP Tunnel Mode Encapsulation

This table is (U//FOUO)					
Field		Authenticated	Encrypted	Overhead	
				Bits	Octets
Black IP Header				160	20
HAIPE ESP Header	SPI	X		32	4
	ESP Sequence Number			32	4
	State Variable			128	16
	Payload Sequence Number	X	X	64	8
Red IP Datagram	Red IP Header	X	X	-	-
	Red IP Payload	X	X	-	-
HAIPE ESP Trailer	Padding (Crypto + TFS)	X	X	184	23
	Dummy	X	X	8	1
	Pad Length	X	X	16	2
	Next Header	X	X	8	1
	Authentication Data		X	32	4
Total				664	83
This table is (U//FOUO)					

5250

5251 (U//FOUO) Note that HAIPE provides authentication (anti-spoof protection) of the Red IP
 5252 datagram and parts of the HAIPE header and trailer as indicated in the Authenticated column in
 5253 Table 2.3-6. PDUs that fail the authentication check are discarded. This may be undesirable for
 5254 voice and video data where a few bit errors are tolerable. HAIPE provides confidentiality
 5255 (encryption) of the Red IP datagram and parts of the HAIPE header and trailer as indicated in the
 5256 Encrypted column in Table 2.3-6.

5257 (U//FOUO) The 32-bit SPI identifies the security association to the receiving HAIPE device. The
 5258 SPI is either calculated from key material and peer information (for PPKs) or developed during
 5259 the IKE exchange (for automatic TEK generation).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5260 (U//FOUO) HAIPE uses the payload Payload Sequence Number (PSEQN) for anti-replay
 5261 service. Therefore even though transmitting HAIPE devices initially set the ESP SEQN value to
 5262 a random number and increment for each packet set, receiving HAIPE devices ignore the ESP
 5263 SEQN value.

5264 (U//FOUO) The state variable is used to synchronize the crypto state of the transmitting and
 5265 receiving HAIPE device, and does not repeat for any given TEK. The state variable is
 5266 transmitted with each PDU so that the receiving HAIPE device can independently decrypt each
 5267 PDU. Table 2.3-7 shows contents of the state variable.

Table 2.3-7: (U//FOUO) HAIPE State Variable Content

This table is (U//FOUO)			
Field	Bits	Value On Wire	Encryption/Decryption Value
Update Count	16	Indicates daily update count of TEK	
Unique	69	Unique	
LRS	36	Unique	Stepped when SEG# has value 0
SEQ#	4		All zeros
SEG#	3		Stepped from 0-3 for WEASLE Mode
Total	128	-	-
This table is (U//FOUO)			

5269

5270 (U//FOUO) The update count field represents the number of daily updates performed on the
 5271 TEK. The receiver uses to determine which update version of the TEK was used by the
 5272 transmitter to encrypt the PDU.

5273 (U//FOUO) The Unique, LRS, SEQ#, and SEG# fields are unique on the wire for each PDU.

5274 (U//FOUO) The initial value for the Linear Recursive Sequence (LRS) is transmitted on the wire
 5275 and is uniquely generated for each PDU. During encryption or decryption processing of crypto
 5276 blocks of the same PDU, the LRS is stepped each time the SEG# has the value zero illustrated in
 5277 Figure 2.3-19. The polynomial for the LRS is $1+x^{11}+x^{36}$.

5278 (U//FOUO) The SEQ# field is unique on the wire, but all zeros for encryption and decryption.

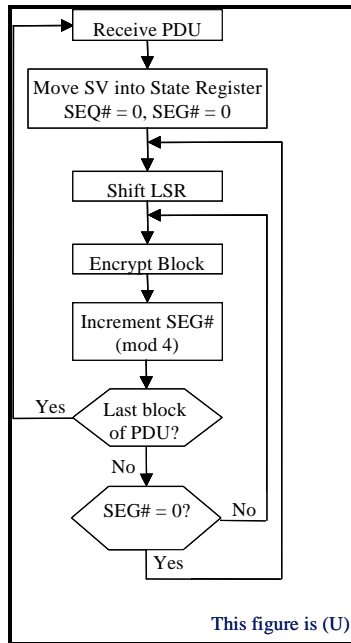
5279 (U//FOUO) The SEG# is unique on the wire. During encryption or decryption processing of
 5280 crypto blocks of the same PDU, the SEG# is stepped from 0 to 3 in WEASEL mode as illustrated
 5281 in Figure 2.3-19.

5282 (U//FOUO) The PSEQN value provides anti-replay services for HAIPE. The PSEQN value is
 5283 both authenticated and encrypted. The PSEQN value is initialized to zero by the transmitting
 5284 HAIPE upon SA setup and incremented for each PDU sent for the duration of the TEK. The
 5285 receiving HAIPE uses the PSEQN value to detect and discard PDUs that are replayed.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5286 (U//FOUO) Inner IP header fields are coded in accordance with RFC 2401.

5287 (U//FOUO) Padding ensures the encrypted PDU is an integer multiple of the encryption block
 5288 size, which may be negotiated during the IKE. Padding is also used to provide TFS protection.
 5289 The ESP padding is added by the transmitting HAIPE and removed by the receiving HAIPE.



5290

Figure 2.3-19: (U//FOUO) State Variable Stepping

5291

5292 (U//FOUO) The ESP dummy field is used to support TFS protection. A value of all 0s indicates a
 5293 dummy PDU, whereas a value of all 1s indicates a Valid PDU.

5294 (U//FOUO) The ESP pad length field is used by the receiving HAIPE to determine the amount of
 5295 padding added by the transmitting HAIPE, so the receiver can remove this padding before
 5296 forwarding the decrypted PDU to the receiving host.

5297 (U//FOUO) The ESP next header field indicates the encapsulated protocol. In the case of
 5298 tunneled user traffic, this field will indicate IPv4 (or IPv6).

5299 (U//FOUO) HAIPE supports both the BIP-32 and SHA-1 algorithms for authentication. In either
 5300 case, the authentication value is encrypted under the negotiated cryptographic algorithm
 5301 operating in WEASEL mode.

5302 (U//FOUO) The BIP-32 algorithm is a 32-bit exclusive-or function of each of the 32-bit words of
 5303 data to be authenticated and a 32-bit word of hexadecimal "A"s (0xAAAAAAAA).

5304 2.3.3.2.2.1.1.2 (U) Usage Considerations

5305 2.3.3.2.2.1.1.2.1 (U) Implementation Issues

5306 (U//FOUO) The application of IPsec to protect Unclassified traffic in the GIG introduces key
5307 management issues. In order to provide sufficient protection secure Type 3 key material must be
5308 generated, distributed, stored, updated and destroyed. The current KMI is only designed for
5309 Type 1 key material. Given the volume of Unclassified data in the GIG, the number of IPsec
5310 devices that must be keyed will also be significant (residing on every client in the GIG Vision).
5311 Without a key management infrastructure for Type 3 keys, deployment of IPsec to protect
5312 Unclassified traffic may be unmanageable.

5313 (U//FOUO) HAIPE does not support dynamic routing in a multi-homed environment (i.e. an
5314 enclave fronted by more than one HAIPE. This limitation may be overcome by placing external
5315 routers behind HAIPEs, and using IP tunneling (e.g. see RFC-2784 on Generic Routing
5316 Encapsulation) between the routers to disguise the ultimate destination from the INE. This
5317 approach requires an extra IP header, and therefore increases bit overhead across the Black Core.

5318 (U//FOUO) An alternative is to integrate a router into the Red side of the INE and to select the
5319 SA based on the next hop instead of the ultimate destination address. This approach has less bit
5320 overhead than the external router approach, but couples the routing function with the HAIPE
5321 security function.

5322 (U//FOUO) HAIPE does not fully support QoS mechanisms for real-time traffic like voice.
5323 HAIPE does support bypass of IPv4 Type of Service (ToS) field and IPv6 Traffic Class field, but
5324 does not support reservation protocols. For example, TSAT has proposed modifications to
5325 HAIPE to provide an RSVP proxy service where a HAIPE INE would aggregate multiple Red
5326 side RSVP requests into a single Black side RSVP request.

5327 (U//FOUO) HAIPE does not provide true end-to-end security. Currently HAIPE is designed to
5328 support INE implementations with multiple end-systems behind an INE. Even when migrated to
5329 end-systems, HAIPE will not provide true end-to-end security for some applications. For
5330 example, e-mail is a store-and-forward application with multiple IP end-systems in the path.
5331 Likewise, secure voice through a gateway will have IP end-systems as intermediate nodes.
5332 Additional security protocols may be overlaid on top of HAIPE to provide true end-to-end
5333 security (e.g. SMIME v3 for secure e-mail and FNBDT for secure voice).

5334 (U//FOUO) HAIPE is currently not designed for end system implementation. HAIPE version 1
5335 and version 2 only support tunnel mode and does not support transport mode. HAIPE version 3
5336 will support both tunnel and transport modes. Anti-tamper and TEMPEST are also significant
5337 issues for a Type-1 end-system implementation.

5338 (U//FOUO) HAIPE discovery does not support dynamic Black-side IP addresses. Dynamic
5339 Black-side IP addresses are common in a mobile IPv4 environment. The migration to IPv6 in the
5340 future will help to resolve this issue to some extent.

5341 (U//FOUO) HAIPE has significant complexity, and was not intended for implementation in a
 5342 resource-constrained environment (e.g., memory and processing power). A profile of HAIPE
 5343 may be desirable for implementation in hand-held mobile devices.

5344 (U//FOUO) HAIPE was not intended for low bandwidth and/or high BER environments. HAIPE
 5345 has significantly more bit overhead than FNBDT for protecting secure voice traffic. There have
 5346 been several HAIPE-lite proposals to address this issue. These proposals do reduce the bit
 5347 overhead associated with HAIPE, but are still not as efficient as FNBDT for secure voice
 5348 applications. HAIPE is also not tolerant to bit errors. HAIPE provides cryptographic error
 5349 extension and also implements an integrity check on every packet. A single bit error will cause
 5350 the packet to be discarded. This is not necessarily desirable for applications like secure voice that
 5351 can tolerate a few bit errors.

5352 2.3.3.2.2.1.1.2.2(U) Advantages

5353 (U//FOUO) IP layer security supports a black core concept allowing switches and routers to exist
 5354 on the black side of the crypto.

5355 2.3.3.2.2.1.1.2.3(U) Risks/Threats/Attacks

5356 (U//FOUO) IP layer security is somewhat susceptible to Traffic Flow Analysis. Moving IP layer
 5357 security (HAIPE) back to end systems will likely increase susceptibility to Traffic Flow
 5358 Analysis. Link layer security may be used to provide Traffic Flow Security (TFS) for traffic
 5359 encrypted at the IP layer.

5360 2.3.3.2.2.1.1.3 (U) Maturity

5361 (U//FOUO) IPsec is a Mature technology in the commercial world, but continues to evolve. Type
 5362 1 IP security standards (SDNS and HAIPE) have been around for quite some time, but HAIPE
 5363 continues to evolve, and the current standard is not adequate to support the long-term GIG
 5364 vision. The TRLs for the IP security technology are illustrated in Table 2.3-8 below. Table 2.3-8
 5365 is based on the HAIPE Roadmap presentation dated May 12, 2004.

5366 **Table 2.3-8: (U//FOUO) IP Security Technology Readiness Levels**

This table is (U//FOUO)			
Specification	Core/Extension	Features	TRL
IPsec (November 1998)			9
IPsec (March 2004)			2
HAIPE v1.3.5	Core	BATON and FIREFLY, IPv4	9
HAIPE v2.0.0	Core	MEDLEY and Enhanced FIREFLY IPv6, QoS, Multicast Over the Network Management	2
	Extension	Interim Routing Enclave Prefix Discovery Server Foreign Interoperability	

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U//FOUO)			
Specification	Core/Extension	Features	TRL
HAIPE v3 & Beyond	Core	Bandwidth Efficiency (v3) OTNK (Beyond v3) Over the Network Management Enhancements (Beyond v3)	1
	Extension	Standard HAIPE MIB Scalable & Efficient Routing End-to-End QoS Voice over Secure IP (VoSIP)	
This table is (U//FOUO)			

5367

5368 2.3.3.2.2.1.1.4 (U) Standards
 5369 (U//FOUO) The standards applicable to IP security technology are identified in Table 2.3-9
 5370 below.

5371 **Table 2.3-9: (U//FOUO) Standards Applicable to IP Security Technology**

This table is (U)			
Number	Title	Version	Date
	Interoperability Specification For High Assurance Internet Protocol Encryptor (HAIPE) Devices	1.3.5	May 2004
	Interoperability Specification For High Assurance Internet Protocol Encryptor (HAIPE) Devices	2.0.0	May 2004
RFC-2401	Security Architecture for the Internet Protocol http://www.ietf.org/rfc/rfc2401.txt		November 1998
	Security Architecture for the Internet Protocol http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2401bis-02.txt		April 2004
RFC-2402	IP Authentication Header http://www.ietf.org/rfc/rfc2402.txt		November 1998
	IP Authentication Header http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2402bis-07.txt		March 2004
RFC-2406	IP Encapsulating Security Payload (ESP) http://www.ietf.org/rfc/rfc2406.txt		November 1998
	IP Encapsulating Security Payload (ESP) http://www.ietf.org/internet-drafts/draft-ietf-ipsec-esp-v3-08.txt		March 2004
This table is (U)			

5372 2.3.3.2.2.1.1.5 (U) Cost/Limitations
 5373 (U//FOUO) Moving HAIPE back to end systems may not be as economical as fronting multiple
 5374 end systems with a single HAIPE INE.

5375 2.3.3.2.2.1.1.6 (U) Dependencies
 5376 (U//FOUO) Key management is needed to support commercial IPsec and HAIPE
 5377 implementations. HAIPE supports Pre-Placed Keys (PPKs) as well as auto-generated Traffic
 5378 Encryption Keys (TEKs). Auto-generation includes FIREFLY and Enhanced FIREFLY.

5379 (U//FOUO) HAIPE also depends on remote security management via a Security Management
 5380 Workstation (SMW).

5381 2.3.3.2.2.1.1.7 (U) Alternatives
 5382 (U//FOUO) FNBDT application layer security can be used to provide end-to-end protection for
 5383 secure voice traffic.

5384 (U//FOUO) Sub-network layer security can be used to protect information as it crosses a sub-
5385 network. Sub-network layer security allows black side switches, but still requires all IP routers to
5386 be red. For example, the CANEWARE Front End had a mode where it encrypted the payload of
5387 X.25 packets. A more modern example is FASTLANE, which provides security of the payload
5388 of ATM cells.

5389 (U//FOUO) It is also possible to tunnel red side sub-network, link and physical layers over a
5390 black IP network. For example, BLACKER provided the ability to map red side X.25 addresses
5391 to black side IP addresses creating a Red Virtual Network which spanned a black side internet.
5392 Additional examples include the NES and Sectera INEs, which provide the ability to map red
5393 side MAC addresses to black side IP addresses essentially bridging a black side internet.

5394 (U//FOUO) Security is also possible over SONET using the KG-189 to provide security of the
5395 SONET payload.

5396 2.3.3.2.2.1.1.8 (U) Complementary Techniques

5397 (U//FOUO) Link and physical layer mechanisms provide additional security. TRANSEC
5398 mechanisms support LPI/LPD. HAIPE has some TFS mechanisms, but link security can be used
5399 to enhance Traffic Flow Security (TFS). Higher layer mechanisms (e.g., S/MIME v3 for secure
5400 e-mail and FNBDT for secure voice) can be used to provide true end-to-end security and
5401 confidentiality within a domain.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5402 2.3.3.2.2.1.2 (U) Traffic Flow Security (TFS)

5403 (U) *EDITOR'S NOTE: MATERIAL ON TFS WILL BE INCLUDED IN A FUTURE RELEASE*

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5404 2.3.3.2.2.1.3 (U) Virtual Private Networks (VPN)

5405 (U//FOUO) A Virtual Private Network (VPN) generally connects two private networks over a
5406 publicly accessible network (e.g., the Internet). Most VPNs are IP implementations that can be
5407 handled by a company's existing Internet technology. A VPN can provide a secure connection
5408 between remote sites without additional expenses for leased lines, ISDN, or frame-relay and
5409 Asynchronous Transfer Mode (ATM) technologies.

5410 2.3.3.2.2.1.3.1 (U) Technical Detail

5411 (U//FOUO) VPNs provide authentication, integrity, and confidentiality security services across a
5412 network, usually a publicly accessible network. Most VPN products use IPsec to carry out these
5413 security features, but other protocols (e.g., SSL) are also used in some products. For non-IP
5414 networks, (e.g., Internetwork Packet Exchange [IPX] or AppleTalk) Layer 2 Tunneling Protocol
5415 (L2TP) is more suitable.

5416 (U) IPsec VPNs are a network layer technology. This means they operate independent of the
5417 applications that may use them. Tunnel mode IPsec encapsulates the IP data packet, hiding the
5418 application protocol information. Once the IPsec tunnel is created, various connection types
5419 (e.g., web, email, VoIP, FTP) can flow through the tunnel, each destined for different
5420 destinations on the other side of the VPN gateway.

5421 (U) SSL VPNs are a remote access solution because they do not require IT departments to
5422 upgrade and manage client software. All a user needs is a Web browser.

5423 (U//FOUO) VPN products can be grouped into three categories:

- 5424 • (U) Hardware-based systems
- 5425 • (U) Firewall-based systems
- 5426 • (U) Stand-alone application packages.

5427 (U//FOUO) Hardware-based VPNs typically use encryption routers providing IP services, such
5428 as IPsec tunneling. This is a common deployment strategy in a corporate network infrastructure
5429 to securely connect remote networks. Another hardware implementation involves VPN gateways
5430 used as IPsec tunnel endpoints. The VPN gateways provide firewalls and routing, as well as
5431 authentication, encryption, and key management capabilities.

5432 (U//FOUO) Firewall VPNs take advantage of a firewall's authentication and access control
5433 features adding a tunneling capability and encryption functionality.

5434 (U//FOUO) Stand-alone application VPNs use software to perform the access control,
5435 authentication and encryption needed for the VPN. The software VPN solution is the least
5436 expensive but generally has the worst performance. Software VPNs are adaptive to technology
5437 changes because no hardware changes are involved. The software VPN is ideal for company
5438 employees working on travel or from home.

5439 2.3.3.2.2.1.3.2 (U) Usage Considerations

5440 (U//FOUO) When setting up a VPN, you must consider the following options:

- 5441 • (U) Security protocols supported
- 5442 • (U) Cryptographic algorithms supported
- 5443 • (U) Key management system used
- 5444 • (U) User authentication used
- 5445 • (U) Server platforms that run the product
- 5446 • (U) Client platforms supported
- 5447 • (U) Accreditation or approval
- 5448 • (U) Price and maintenance costs
- 5449 • (U) Number of users or connections supported

5450 2.3.3.2.2.1.3.2.1 (U) Implementation Issues

5451 (U//FOUO) Commercial vendors have VPN capabilities built into firewall, gateway, or router
5452 products. There are currently dozens of different COTS VPN products available today. These
5453 products range from supporting small business connections to supporting large organizations
5454 requiring thousands of connections. Many vendors have a family of VPN products that support
5455 the different ranges of user needs. Some products support modular upgrades and have integrated
5456 hardware VPN acceleration capabilities, delivering highly scalable, high-performance VPN
5457 services.

5458 (U//FOUO) As the VPN products have advanced, their configuration and administration has
5459 become easier. Configuration and management tools have been created to make the
5460 establishment, administration and monitoring of VPN clients and networks easier to perform.
5461 Some products advertise a One-click VPN, where VPNs can be created with a single operation
5462 by using VPN communities. As new members are added to the community, they automatically
5463 inherit the appropriate properties and can immediately establish secure IPsec/IKE sessions with
5464 the rest of the VPN community.

5465 (U//FOUO) IPsec is still the most popular protocol for performing VPN security, but SSL has
5466 been gaining support in the last few years. Many VPN vendors now support both protocols in
5467 either the same product or as a separate item. One advantage of SSL over IPsec is that SSL does
5468 not require special VPN client software on remote PCs, which reduces administrative costs.

5469 (U) VPN Products

5470 (U) There are many COTS VPN products. The following is a list of some of the VPN products
5471 available today:

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 5472 • (U) Cryptek's DiamondTEK has been evaluated and validated in accordance with the
5473 provisions of the NIAP Common Criteria Evaluation and Validation Scheme and the
5474 Common Criteria Recognition Arrangement as a EAL4 product -
5475 <http://www.cryptek.com/SecureNetworks/>
- 5476 • (U) Blue Ridge Networks' VPN CryptoServer is also on the Common Criteria validated
5477 products list (EAL2) - <http://www.blueridgenetworks.com>
- 5478 • (U) Check Point's VPN-1 Pro product line is an integrated VPN and firewall gateway,
5479 which offers management capability, attack protection and traffic shaping technology. -
5480 <http://www.checkpoint.com/products/index.html>
- 5481 • (U) Nortel Networks Contivity is a line of VPN switches and gateways with supporting
5482 configuration and management tools. -
5483 <http://www.nortelnetworks.com/products/family/contivity.html>
- 5484 • (U) Cisco PIX Security Appliances support hardware and software VPN clients, as well
5485 as PPTP and L2TP clients. -
5486 <http://www.cisco.com/en/US/products/hw/vpndevc/index.html>
- 5487 • (U) SafeNet's HighAssurance™ Gateway product lines provide IPsec VPN solutions for
5488 small to large customers. SSL VPN also supported. - <http://www.cylink.com>
- 5489 • (U) Avaya Secure Gateway products have specialized support for voice-over-IP (VoIP)
5490 applications - http://www1.avaya.com/enterprise/vpn/sg203_sg208/
- 5491 • (U) Symantic Gateway supports both IPsec and SSL based VPN products -
5492 <http://enterprisesecurity.symantec.com/content/productlink.cfm?EID=0>
- 5493 • (U) SonicWall has firewall and gateway products that feature IPsec VPN security -
5494 <http://www.sonicwall.com/products/vpnapp.html>
- 5495 • (U) ADTRAN's NetVanta 2000 Series is a family of VPN/firewall appliances -
5496 <http://www.adtran.com>
- 5497 • (U) ArrayNetworks Array SP family of appliances offers SSL VPNs -
5498 <http://www.arraynetworks.net/globalaccess.htm>
- 5499 • (U) Celestix's RAS3000 supports SSL VPN for Microsoft Exchange Server 2003 -
5500 <http://www.celestix.com/products/ras/ras3000/sslvpnforexchange.htm>
- 5501 • (U) Lucent Technologies Access Point® supports routing, secure VPN, QoS, firewall
5502 security, and policy management - <http://www.lucent.com/solutions/>
- 5503 • (U) Juniper Networks Netscreen SSL VPNs provide a broad range of SSL VPN
5504 appliances - <http://www.juniper.net/products/ssl/>
- 5505 • (U) V-ONE produces both IPsec and SSL VPN products - <http://www.v-one.com>

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5506 2.3.3.2.2.1.3.2.2(U) Advantages
5507 (U//FOUO) VPNs provide economical and secure solutions for remote access users (mobile
5508 users and telecommuters), intranets (site-to-site connections within a company or organization),
5509 and extranets (organization to organization network connections to suppliers, customers, or
5510 partners).

5511 2.3.3.2.2.1.3.2.3(U) Risks/Threats/Attacks
5512 (U//FOUO) VPN clients should not be able to access your private network and the Internet at the
5513 same time. Doing so can be a security risk if the VPN client can become a gateway between the
5514 Internet and the private network.

5515 (U//FOUO) PPTP authentication dependence on Microsoft Challenge Handshake Authentication
5516 Protocol (MSCHAP) makes it vulnerable to attacks using a hacker tool called l0phtcrack.

5517 (U//FOUO) Nearly all computer equipment is susceptible to Distributed Denial of Service
5518 (DDoS) attacks. The Corrent Corporation's S3500 Turbocard Firewall/VPN accelerator is one
5519 VPN product that can withstand a massive DDoS attack, while keeping valid network traffic
5520 flowing at a high rate. The new Corrent® S3500 Turbocard is able to sustain 50,000 TCP
5521 sessions per second and deliver 648 Megabits per second in throughput in the face of a
5522 concentrated attack.

5523 2.3.3.2.2.1.3.3 (U) Maturity
5524 (U//FOUO) VPNs are a mature technology in the commercial world and continue to evolve.
5525 Products continue to support additional protocols and algorithms and run on more and more
5526 different platforms.

5527 (U//FOUO) Interoperability between different manufacturers has seen significant improvements
5528 over the last few years but interoperability issues still exist.

5529 (U//FOUO) VPNs are Mature (TRLs 7 – 9). Interoperability between different manufacturers
5530 and platforms should continue to move forward.

5531 2.3.3.2.2.1.3.4 (U) Standards
5532 (U//FOUO) Table 2.3-10 identifies the standards applicable to VPN technology.

5533

Table 2.3-10: (U//FOUO) Standards Applicable to VPN Technology

This table is (U)		
Number	Title	Date
RFC-2401	Security Architecture for the Internet Protocol http://www.ietf.org/rfc/rfc2401.txt	November 1998
	http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2401bis-02.txt	April 2004
RFC-2402	IP Authentication Header http://www.ietf.org/rfc/rfc2402.txt	November 1998
	http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2402bis-07.txt	March 2004
RFC-2406	IP Encapsulating Security Payload (ESP) http://www.ietf.org/rfc/rfc2406.txt	November 1998
	http://www.ietf.org/internet-drafts/draft-ietf-ipsec-esp-v3-08.txt	March 2004
	The SSL Protocol, Version 3.0 http://wp.netscape.com/eng/ssl3/ssl-toc.html	November 1996
RFC 3031	Multiprotocol Label Switching Architecture http://www.ietf.org/rfc/rfc3031.txt	January 2001
RFC 2661	Layer Two Tunneling Protocol (L2TP) http://www.ietf.org/rfc/rfc2661.txt	August 1999
RFC 2637	Point-to-Point Tunneling Protocol (PPTP) http://www.ietf.org/rfc/rfc2637.txt	July 1999
	VPN Protection Profile for Protecting Sensitive Information http://www.iatf.net/protection_profiles/file_serve.cfm?chapter=vpn_pp.pdf	February 2000
This table is (U)		

5534 2.3.3.2.2.1.3.5 (U) Cost/Limitations

5535 (U//FOUO) Most VPN products have a maximum connection number. So before purchasing a
5536 VPN product you must determine the maximum number of VPN connections you expect to have.

5537 2.3.3.2.2.1.3.6 (U) Alternatives

5538 (U//FOUO) There are several alternatives to VPN security of information over an untrusted
5539 network.

- 5540 • (U//FOUO) FNBDT application layer security can be used to provide end-to-end
5541 protection for secure voice traffic
- 5542 • (U//FOUO) Sub-network layer security can be used to protect information as it crosses a
5543 sub-network. Sub-network layer security allows black side switches, but still requires all
5544 IP routers to be red. For example, the CANEWARE Front End had a mode where it
5545 encrypted the payload of X.25 packets. A more modern example is FASTLANE, which
5546 provides payload security for ATM cells

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 5547 • (U//FOUO) Security is also possible over SONET using the KG-189 to provide security
5548 of the SONET payload.

5549 2.3.3.2.2.1.3.7 (U) References

5550 (U) <http://www.cryptek.com/SecureNetworks/>

5551 (U) <http://www.blueridgenetworks.com>

5552 (U) <http://www.checkpoint.com/products/index.html>

5553 (U) <http://www.nortelnetworks.com/products/family/contivity.html>

5554 (U) <http://www.cisco.com/en/US/products/hw/vpndevc/index.html>

5555 (U) <http://www.cylink.com>

5556 (U) http://www1.avaya.com/enterprise/vpn/sg203_sg208/

5557 (U) <http://enterprisesecurity.symantec.com/content/productlink.cfm?EID=0>

5558 (U) <http://www.sonicwall.com/products/vpnapp.html>

5559 (U) <http://www.adtran.com>

5560 (U) <http://www.arraynetworks.net/globalaccess.htm>

5561 (U) <http://www.celestix.com/products/ras/ras3000/sslvpnforexchange.htm>

5562 (U) <http://www.lucent.com/solutions/>

5563 (U) <http://www.juniper.net/products/ssl/>

5564 (U) <http://www.v-one.com>

5565 (U) Security Architecture for the Internet Protocol - <http://www.ietf.org/rfc/rfc2401.txt> and
5566 <http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2401bis-02.txt>

5567 (U) IP Authentication Header - <http://www.ietf.org/rfc/rfc2402.txt> and <http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2402bis-07.txt>

5569 (U) IP Encapsulating Security Payload (ESP) - <http://www.ietf.org/rfc/rfc2406.txt> and
5570 <http://www.ietf.org/internet-drafts/draft-ietf-ipsec-esp-v3-08.txt>

5571 (U) The SSL Protocol, Version 3.0 - <http://wp.netscape.com/eng/ssl3/ssl-toc.html>

5572 (U) Multiprotocol Label Switching Architecture - <http://www.ietf.org/rfc/rfc3031.txt>

5573 (U) Layer Two Tunneling Protocol (L2TP) - <http://www.ietf.org/rfc/rfc2661.txt>

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5574 (U) Point-to-Point Tunneling Protocol (PPTP) - <http://www.ietf.org/rfc/rfc2637.txt>

5575 (U) VPN Protection Profile for Protecting Sensitive information -
5576 http://www.iatf.net/protection_profiles/file_serve.cfm?chapter=vpn_pp.pdf

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5577 **2.3.3.2.2.2 (U) Real-Time Data Technologies**

5578 2.3.3.2.2.2.1 (U) Secure VoIP Call Control

5579 (U//FOUO) Secure VoIP Call Control addresses technologies used to protect the signaling plane
5580 or VoIP call establishment protocols. Secure VoIP technologies that focus on the voice packets
5581 are described in the previous secure VoIP section.

5582 (U//FOUO) Note that Secure VoIP call control mechanisms may be considered part of network
5583 management and control technologies within the GIG. Further information on VoIP call controls
5584 from the view of network security may be addressed in the GIG network control technology
5585 section. This section concerns itself with the protection of user information that is potentially
5586 vulnerable when using VoIP call control. This section also complements the secure VoIP section
5587 that describes methods to secure user voice.

5588 2.3.3.2.2.2.1.1 (U) Technical Detail

5589 (U//FOUO) A brief introduction to VoIP call control is presented followed by a description of
5590 security mechanisms that can be used to secure call control.

5591 (U) The most common call control protocols used in VoIP system today include:

- 5592 • (U) Session Initiation Protocol (SIP)
- 5593 • (U) H.323
- 5594 • (U) Media Gateway Control Protocol (MGCP)
- 5595 • (U) Gateway Control Protocol (GCP), formerly MEGACO, and H.248

5596 (U) In the spirit of distributed call control rather than centralized, integrated call managers, the
5597 IETF has decomposed gateways into Media Gateway Controllers and Media Gateways. As such,
5598 MGCP and GCP are protocols that can be used when PSTN-VoIP Gateways (PSTN-VoIP GWs)
5599 and multi-media conference units are decomposed between control and media processing units.
5600 This document does not address MGCP and GCP security mechanisms. It is assumed that GW
5601 control and processing units are integrated or collocated within a single security domain such
5602 that security mechanisms do not need to be applied to MGCP or GCP. Commercial IPsec or TLS
5603 could be used to secure these protocols within a single security domain if needed.

5604 (U//FOUO) VoIP networks also require a QoS architecture designed to support the voice service.
5605 As such, secure mechanisms for the GIG QoS architecture needs to be developed as a
5606 complementary technology for secured voice. GIG secure VoIP call control and secure QoS
5607 mechanisms may likely need to work with each other, possibly through a network interface.
5608 Secure QoS security mechanisms are beyond the scope of this section.

5609 (U//FOUO) Priority of Service, PoS, is another important voice feature. This feature allows users
5610 to pre-empt other voice calls or be placed in a higher priority queue for call processing. The GIG
5611 secure VoIP call control will need to request or invoke priority of service and, therefore, is likely
5612 to interact with GIG PoS security mechanisms. PoS security mechanisms are beyond the scope
5613 of this section.

5614 (U//FOUO) Also note that the GIG VoIP call control architecture will need to include a dialing
5615 plan that not only includes SIP and H.323-based user identities, but also users on non-GIG
5616 networks expected to conform to E.164 numbering plans. This implies directory service
5617 techniques such as Electronic Numbering (ENUM). As such, GIG directory services that provide
5618 VoIP calling plans will need to be secured. The VoIP call control security mechanisms will need
5619 to interact with the security mechanisms of these directory services. GIG directory service
5620 security technologies are beyond the scope of this section.

5621 (U) Therefore, this section focuses on SIP and H.323 as addressed below.

5622 (U) SIP

5623 (U) SIP is a text based client/server protocol that can establish, modify, and terminate
5624 multimedia sessions (conferences) or Internet telephony calls. SIP can invite participants to
5625 unicast and multicast sessions. An initiator does not necessarily have to be a member of the
5626 session to which it is inviting, and new media streams and participants can be added to an
5627 existing multi-media session. It can establish, modify, and terminate multimedia sessions or
5628 calls, such as conferences, Internet telephony and similar applications. SIP enables VoIP
5629 gateways, client end points, Private Branch Exchanges (PBX), and other communications
5630 systems and devices to communicate with each other.

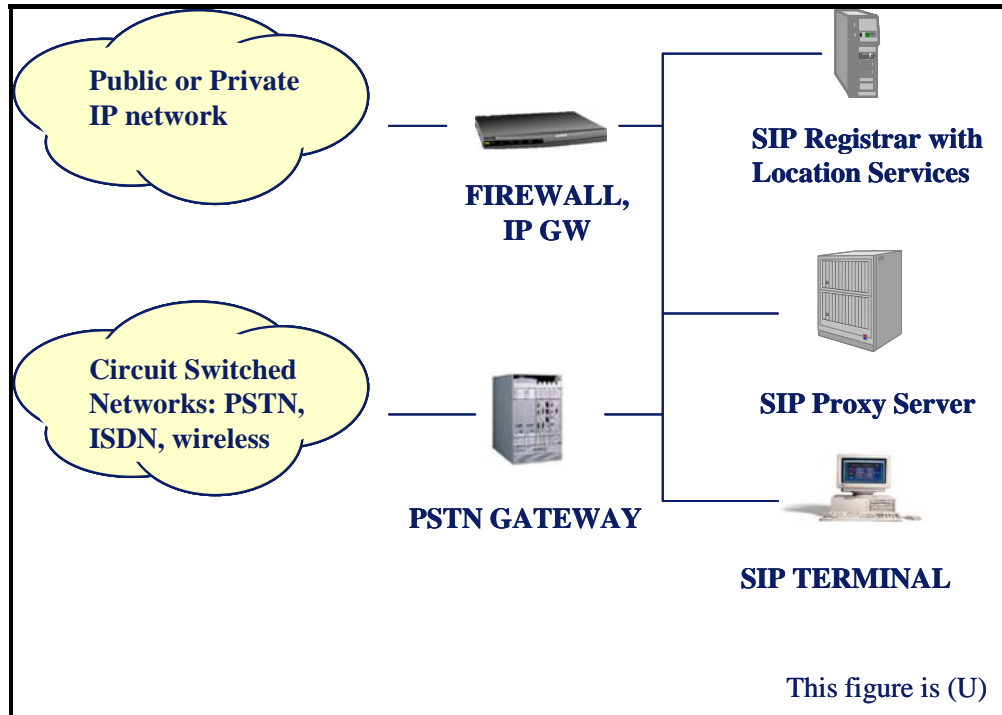
5631 (U) SIP transparently supports name mapping and redirection services, allowing the
5632 implementation of Intelligent Network telephony subscriber services. These facilities also
5633 include mobility that allows the network to identify end users as they move.

5634 (U) SIP supports five facets of establishing and terminating multimedia communications:

- 5635 • (U) User location: determination of the end system to be used for communication
- 5636 • (U) User capabilities: determination of the media and media parameters to be used
- 5637 • (U) User availability: determination of the willingness of the called party to engage in
5638 communications
- 5639 • (U) Call setup: ringing, establishment of call parameters at both called and calling party
- 5640 • (U) Call handling: including transfer and termination of calls.

5641 (U) SIP does not offer conference control services such as floor control or voting and does not
5642 prescribe how a conference is to be managed, but SIP can be used to introduce conference
5643 control protocols. SIP does not allocate multicast addresses and does not reserve network
5644 resources.

5645 (U) SIP network elements are shown in Figure 2.3-20 and described below.



5646

5647

Figure 2.3-20: (U) SIP Architecture

5648 (U) **USER AGENT:** The user agent, shown here as a client, accepts requests from a user and
 5649 provides the appropriate SIP messages, or receives SIP messages and provides appropriate
 5650 responses to the user. It is the SIP end point in the network. Examples for such a user agent
 5651 might be an SIP-enabled PC or a SIP-enabled UMTS mobile device. Gateways can also act as
 5652 SIP user agents, for example a VoIP SIP phone calling a POTS line will connect to the PSTN via
 5653 a VoIP GW. The VoIP GW, then provides the SIP endpoint, or user agent, for the POTS line.

5654 (U) **REGISTRAR:** The SIP Registrar is a server that receives registration requests from **USER**
 5655 **AGENTS** in order to keep a current list of all SIP users and their location which are active within
 5656 its domain/network. A registrar is typically collocated with a proxy or redirect server.

5657 (U) **LOCATION SERVICES:** Location services find the location of a requested party in support
 5658 of SIP based mobility. For example, when a SIP agent places a session or call request to another
 5659 network user, the SIP location server will find the domain in which the second party was last
 5660 registered. When found, the SIP request (SIP INVITE to the session) will be forwarded to the
 5661 appropriate domain.

5662 (U) **PROXY SERVER:** The proxy server is an intermediate device that receives SIP requests
 5663 from a user agent and then forwards the requests on the client's behalf. The proxy server can stay
 5664 in a signaling path for the duration of the session. Proxy servers can also provide functions such
 5665 as authentication, authorization, network access control, routing, reliable request retransmission,
 5666 and security. Equally important, the proxy server can be used by the network to execute a range
 5667 of supplementary services. Soft switches, for example, may use the SIP proxy as a way to
 5668 interface the call or session model to the user agent. In the VoIP world, call forwarding on busy
 5669 can be implemented and invoked in the proxy server.

5670 (U) RE-DIRECT SERVER: The re-direct server provides the client with information about the
5671 next hop or hops that a message should take, and then the client contacts the next hop server or
5672 user agent directly. Unlike the use of a proxy server, the re-direct server simply serves to direct
5673 on-going communications at session initiation, but does not stay in the signaling path for the
5674 duration of the data session.

5675 **(U) SIP Security Mechanisms**

5676 (U) The SIP RFCs describe a number of security mechanisms that can be used within a SIP
5677 system. Message authentication and encryption of SIP messages are supported. Since SIP uses
5678 proxy servers and registrars in support of service execution on behalf of the user, SIP assumes
5679 security associations between the SIP client and various SIP infrastructure elements, rather than
5680 secured end-to-end call control security between voice users. This, of course, means that all SIP
5681 servers need to be trusted network elements. As such, all SIP call control is assumed to be
5682 located within a single security domain. IPsec can be used to tunnel SIP call control from SIP
5683 clients to SIP servers across backhaul networks that may be outside the SIP security domain.
5684 Note that SIP is a text-based protocol, borrowing many elements from other text based protocols
5685 such as HTTP. As such, SIP security reuses HTTP and MIME security mechanisms as explained
5686 below. (Note that PGP is not longer recommended in the latest SIP RFC.)

5687 (U) The following encryption scenarios are identified in the SIP RFCs:

- 5688 • (U) A network can use lower layer security protocols, such as IPsec or TLS between the
5689 SIP UA and SIP server. Although many implementations transport SIP with UDP, SIP
5690 can also be transported with TCP so that TLS can be used
- 5691 • (U) TLS can be used between servers
- 5692 • (U) S/MIME techniques can be used to encrypt SIP bodies for end-to-end security of the
5693 SIP message payload, while leaving SIP headers in the clear for server support. S/MIME
5694 also provides for integrity and supports mutual authentication. Note that this method does
5695 offer protection of user network address or URI information. Furthermore, specific
5696 applications of SIP servers can offer the user a number of network enabled services. The
5697 set of services offered by these kinds of SIP servers may be restricted if the SIP message
5698 body is opaque to the SIP server

5699 (U//FOUO) SIP includes basic password authentication mechanisms as well as digest based
5700 mechanisms. The SIP protocol includes messages that facilitate authentication. For example, SIP
5701 protocol specific authentication challenge messages Response 401 (Unauthorized) or Response
5702 407 (Proxy Authentication Required) can be used in conjunction with a cryptographic
5703 mechanism. The Digest authentication mechanisms called out by the SIP RFCs are based upon
5704 HTTP authentication.

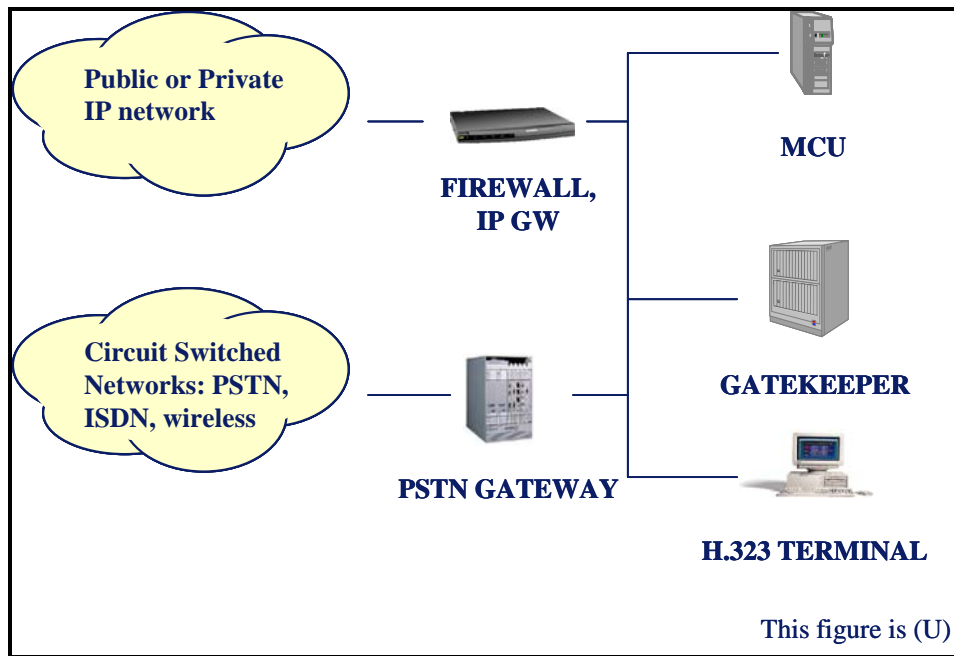
5705 (U) SIP also defines option mechanisms to convey user privacy requirements. Finally, SIP
5706 extensions are identified for media and QoS authorization as a means of protecting against DoS
5707 attacks.

5708 **(U) A Brief Overview of H.323**

5709 (U) H.323 from the ITU is binary based (ASN.1) and provides for both signaling plane
 5710 messaging as well as signaling to bearer control. Because it was initially designed to support
 5711 video packets, H.323 has considerable overhead, which is a disadvantage for IP telephony
 5712 applications.

5713 (U) Note that H.323 is an umbrella specification, covering several protocols related to call setup
 5714 and signaling. Chief among these are H.225, which defines the call signaling channel, H.245, or
 5715 the call control channel, and RAS - registration, admission, status. Underlying these are the Real
 5716 Time Transport Protocol (RTP) and/or the Real Time Control Protocol (RTCP), which define the
 5717 basic requirements for transporting real time data over a packet network.

5718 (U) The H.323 architecture and components are introduced in Figure 2.3-21 and summarized
 5719 below.



5720

5721 **Figure 2.3-21: (U) H.323 Network Elements**

5721

5722 (U) Gatekeeper:

- 5723 • (U) Manages an H.323 zone or network (collection of H.323 devices)
- 5724 • (U) Supports address translation, access and admissions control, bandwidth control, and
 5725 allocation
- 5726 • (U) Optional functionality includes call authorization, supplementary services, directory
 5727 services, call management services

5728 (U) Gateway

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 5729 • (U) The Gateway provides interoperability between different networks by converting
5730 signaling and bearer between, as an example, IP and circuit-based networks
- 5731 • (U) H.323 Terminal
- 5732 • (U) The Terminal is the H.323 signaling endpoint/client on an IP network
- 5733 • (U) Supports real-time, 2-way communications with another H.323 entity. Must support
5734 voice (audio codecs) and signaling (Q.931, H.245, RAS). Optionally supports video and
5735 data, e.g., PC phone or videophone, Ethernet phone
- 5736 (U) Multipoint Control Unit (MCU)
- 5737 • (U) The MCU supports conferences between 3 or more endpoints
- 5738 • (U) The MCU must contain multi-point controller (MC) for signaling and may contain
5739 multi-point processor (MP) for media stream processing. The MCU can be stand-alone or
5740 integrated into gateway, gatekeeper or terminal
- 5741 **(U) H.323 Security Framework defined in H.235**
- 5742 (U) The H.235 standard defines a security framework to be used within H.323 systems. H.323
5743 supports a menu of encryption, and authentication options are supported. Note that H.235 defines
5744 security mechanisms between H.323 clients and H.323 call control servers (Gatekeepers, MCUs,
5745 Gateways). End-to-end call control security between clients is not provided. Therefore, H.235
5746 requires all H.323 infrastructure elements to be trusted servers. The H.245 signaling protocol
5747 used within H.323 systems includes methods to negotiate the security algorithms and keys used
5748 for a secure call control connection.
- 5749 (U) H.235 supports DES, 3DES and AES encryption algorithms. H.235 allows for TLS or IPsec
5750 to be used amongst H.323 clients and H.323 call control servers.
- 5751 (U) A variety of authentication options are identified, which include HMAC-SMA1-96. Public
5752 certificates as well as subscription-based authentication mechanisms can be used. Three options
5753 for subscription-based authentication are identified, specifically:
- 5754 • (U) Password-based with symmetric encryption (shared secret)
- 5755 • (U) Password-based with hashing
- 5756 • (U) Certificate based subscriptions with digital signatures.
- 5757 (U) Key management is incorporated into the H.323 family of signaling specifications. H.235
5758 describes the use of H.245 signaling protocol messages for key management. The use of H.323
5759 RAS protocol for key management has been identified in the specification, but is not completely
5760 defined. Third party key escrow schemes are described, and Diffie-Hellman can be use for key
5761 agreements.
- 5762 (U) This menu of security options is organized within three security profiles as listed below:

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5763 • (U) The Basic security profile employs user passwords as part of the authentication
5764 approach

5765 • (U) An optional Digital Signature profile utilizing certificates

5766 • (U) A Hybrid profile that combines elements of Basic and Digital Signature

5767 (U) H.235 also describes secure call control consideration in the presence of firewalls and
5768 Network Address Translation devices.

5769 (U) Finally, H.235 also references H.510 and H.530, which together describe security
5770 mechanisms for mobility across H.323 systems. These specifications describe a generic security
5771 concept for mobility among domains. Hop-by-hop security with shared secrets is employed to
5772 protect call control.

5773 2.3.3.2.2.1.2 (U) Usage Considerations

5774 2.3.3.2.2.1.2.1 (U) Implementation Issues

5775 (U) Call Control Environment

5776 (U//FOUO) Since both SIP and H.323 security measures rely upon trusted call control network
5777 elements to complete call processing, the VoIP call control will need to reside within a single
5778 security domain. Furthermore, SIP and H.323 security mechanisms are not type 1 and do not
5779 lend themselves to existing type 1 solutions. Therefore, other security mechanisms, such as
5780 HAIPEs, are required to transport call control across security domain boundaries. For example,
5781 Secure SIP mechanisms at the session layer can be used amongst devices within the security
5782 domain. Clients roamed onto other security domains can use HAIPEs to tunnel back into the
5783 security domain to join VoIP calls using the SIP security. As such, the client would apply SIP
5784 security over HAIPEs security for the call control.

5785 (U) Clear – Secure Transitions

5786 (U//FOUO) Note that the ability to transition between clear and secure voice is an important
5787 security feature for voice services. As such, the GIG VoIP architecture needs to allow for a
5788 transition between clear and secure voice media types within a single call.

5789 (U) Multiple call control systems and user mobility

5790 (U//FOUO) Since SIP and H.323 need to reside in a single security domain, it is conceivable that
5791 a SIP client may need to operate with a number of call control managers, depending upon the
5792 security domain of other call participants. For example, a user may need to register on the GIG
5793 VoIP call control manager for communications with on-GIG users and another call control
5794 manager to communicate with a non-GIG or coalition user. This means that user mobility may
5795 need to be tracked in multiple call control planes, based upon security domain.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5796 (U) Security technologies within the call control environments

5797 **(U) SIP security**

5798 (U//FOUO) The SIP security RFCs call out a menu of security options. Therefore, the GIG
5799 security architecture will need to standardize on a specific SIP security profile to ensure
5800 interoperability. This is especially important since call control needs to pass through a chain of
5801 trusted clients and trusted call control network elements (servers) in order to place a VoIP call.
5802 Furthermore, non-GIG networks may use other SIP security profiles, requiring GIG clients to
5803 support additional security mechanisms when communicating with non-GIG users.

5804 (U//FOUO) SIP interaction with IPv6 firewalls is not clearly defined and needs to be studied.

5805 (U//FOUO) Note that many VoIP networks place SIP on top of UDP. But TLS forces SIP to be
5806 placed on TCP, reducing the efficiency of the protocol in comparison with UDP-based
5807 approaches. IPsec may be more efficient alternative to TLS in this case.

5808 **(U) H.323 security**

5809 (U//FOUO) The implementation of H.323 security shares much of the same concerns as SIP
5810 security. H.323 offers a wide variety of security mechanisms within three profile types. The GIG
5811 security architecture will need to standardize on a specific set of security functions for
5812 interoperability. Although H.235 does address interaction with IPv4 firewalls, IPv6 firewall
5813 interaction requires further study.

5814 (U) Unlike SIP, H.323 was originally designed for TCP, so the use of TLS amongst clients and
5815 servers fits well within the H.323 system.

5816 2.3.3.2.2.1.2.2(U) Advantages

5817 (U) SIP security mechanisms fit well within a VoIP architecture and reuse many techniques from
5818 HTTP and S/MIME security.

5819 (U) H.323 security is flexible and, like SIP, is intended to fit well within VoIP architectures.

5820 2.3.3.2.2.1.2.3(U) Risks/Threats/Attacks

5821 (U//FOUO) The SIP RFCs declare SIP to be difficult to secure. SIP security requires trusted SIP
5822 infrastructure (proxies, registrars, etc) and hop-by-hop security mechanisms such as TLS to
5823 avoid security risks. Even so, SIP security features are not type 1 and would need to be extended
5824 to support type 1 techniques. This limits SIP to reside within a single security domain.

5825 (U//FOUO) As with SIP, H.323 requires trusted call control infrastructure and hop-by-hop
5826 security to avoid security risks. Although H.235 describes a number of possible non-repudiation
5827 techniques, the overall topic of non-repudiation is listed for further study in the specification.
5828 Further security evolution is likely. As with SIP, H.323 is not defined to support type 1 security
5829 and would need to be extended to do so. This limits H.323 to reside within a single security
5830 domain.

5831 2.3.3.2.2.1.3 (U) Maturity
 5832 (U//FOUO) Although some of the security techniques SIP leverages are well known and have
 5833 deployed in commercial networks, the overall SIP security is immature and not widely used.
 5834 Therefore, SIP Security as defined in the RFCs is Emerging with an estimated TRL of 5.

5835 (U//FOUO) Similar to SIP, many of the H.323 security mechanism are used in deployed
 5836 networks. But the overall H.235 security framework is not widely used in VoIP networks. As
 5837 such, H.235 is also Emerging with an estimated TRL of 5.

5838 2.3.3.2.2.1.4 (U) Standards
 5839 (U) Table 2.3-11 summarizes the standards relevant to secure VoIP call control.

Table 2.3-11: (U) Secure VoIP Call Control Standards

This table is (U)	
Name	Description
H.235	Security and encryption for H-series multimedia terminals
H.245	Call Control Protocol for multimedia communication: Series H
H.323	Packet-based multimedia communications: Series H
H.510	Mobility for H.323 multimedia systems and services
H.530	Symmetric security procedures for H.323 mobility in H.510
RFC 3262	SIP: Session Initiation Protocol
RFC 3310	HTTP Digest Authentication Using Authentication and Key Agreement (AKA)
RFC 3313	Private SIP Extensions for Media Authorization
RFC 3323	A Privacy Mechanism for the SIP
RFC 3325	Private Extensions to the SIP for Asserted Identity within Trusted Networks
RFC 3329	Security Mechanism Agreement for the SIP
RFC 3435	Media Gateway Control Protocol
RFC 3525	Gateway Control Protocol
RFC 3761	The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)
RFC 3762	Telephone Number Mapping (ENUM) Service Registration for H.323
RFC 3853	S/MIME Advanced Encryption Standard (AES) Requirement for the Session Initiation Protocol (SIP)
This table is (U)	

5841 2.3.3.2.2.1.5 (U) Cost/Limitations
 5842 (U//FOUO) SIP and H.323 security mechanisms require trusted call control network elements,
 5843 restricting application of SIP and H.323 security to within a single security domain. Lower layer
 5844 security, such as HAIPE, is required to tunnel SIP across security domain boundaries. These
 5845 multiple layers of security add complexity to client call control processing and may help increase
 5846 costs.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5847 (U//FOUO) Note that there is no method defined to provide protection of circuit-based signaling.
5848 Therefore any protection offered SIP or H.323 would terminate at a VoIP-circuit signaling
5849 gateway.

5850 2.3.3.2.2.1.6 (U) Dependencies

5851 (U//FOUO) The level of trust of any proxy within the call control chain for either SIP or H.323
5852 call control may impact RAdAC and policy enforcement.

5853 (U//FOUO) The GIG key management architecture will need to take into account SIP and H.323
5854 key management requirements. Although SIP does not specify key management systems, H.245
5855 does include key manage messages.

5856 (U//FOUO) The security mechanism of a VoIP architecture will likely need to interact with the
5857 security mechanisms applied to the GIG QoS and PoS architectures.

5858 (U//FOUO) Also note that GIG directory services will need to accommodate GIG and off GIG
5859 'dialing plans' to support VoIP call control. The protection of these directory services and VoIP
5860 call control security will need to interact and be coordinated within the GIG architecture.

5861 2.3.3.2.2.1.7 (U) Alternatives

5862 (U//FOUO) HAIPE could be applied not only to tunnel SIP across security domains, but could
5863 also be used to protect call control within a single domain. A HAIPE tunnel could be applied
5864 between clients and servers and between servers.

5865 2.3.3.2.2.1.8 (U) Complementary Techniques

5866 (U//FOUO) Protection of QoS, protection of PoS, protection of directory services, and the use of
5867 HAIPEs to span security domains are complementary technologies to secure VoIP call control.

5868 (U//FOUO) A secure key management system that accommodates VoIP call control security
5869 mechanisms is also a complementary technology.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5870 **2.3.3.2.3 (U) Link & Physical Layer Technologies**

5871 **2.3.3.2.3.1 (U) Anti-Jam**

5872 *(U) EDITOR'S NOTE: MATERIAL ON ANTI-JAM WILL BE ADDED IN A FUTURE RELEASE.*

5873 **2.3.3.2.3.2 (U) Link Encryption**

5874 *(U) EDITOR'S NOTE: MATERIAL ON LINK ENCRYPTION WILL BE ADDED IN A FUTURE RELEASE. IT WILL*
5875 *ADDRESS IMPLICATIONS OF EXPANDING LINK ENCRYPTIONS CAPABILITIES TO MEDIUM AND LOW*
5876 *ASSURANCE LINKS.*

5877 **2.3.3.2.3.3 (U) TRANSEC**

5878 *(U) EDITOR'S NOTE: MATERIAL ON TRANSEC WILL BE ADDED IN A FUTURE RELEASE.*

5879 **2.3.3.3 (U) Trusted Platforms**

5880 **2.3.3.3.1 (U) Technical Detail**

5881 **2.3.3.3.1.1 (U) Definition**

5882 (U) A trusted platform is a GIG component that is relied upon to enforce its security policy. That
5883 is, it has been assigned a set of security rules to enforce (its policy) and is relied upon to enforce
5884 those rules. No other GIG component will prevent a violation of the security policy if the trusted
5885 platform is subverted, successfully attacked, or otherwise fails to act appropriately.

5886 (U) By contrast, an untrusted platform is not relied upon to enforce any specific policy. It is
5887 prevented from harming the GIG or its users by other trusted platforms.

5888 (U) The security policy enforced by a given trusted platform can vary. In the 1980s, a trusted
5889 platform was considered to be one that could enforce a multilevel security policy. (See [TCSEC]
5890 for technical details.) That is, one could have information at different classification levels (e.g.,
5891 SECRET information and TOP SECRET information) on the system at the same time, and
5892 possibly have users with different clearances accessing the system at the same time. And, one
5893 could have some appropriate level of confidence that a confidentiality policy would be correctly
5894 enforced. Examples: that TOP SECRET information would not be disclosed directly or indirectly
5895 to a user with only a SECRET clearance or that TOP SECRET and SECRET information would
5896 not be co-mingled in a file labeled SECRET. There might also be an integrity policy of some sort
5897 enforced, e.g., it might be prohibited for a SECRET user to modify or delete a TOP SECRET
5898 file.

5899 (U//FOUO) With the GIG's Task Post Process Use (TPPU) model and different
5900 operational/networking scenarios, the definition of a trusted platform has been expanded from
5901 this previous meaning. It now has the more generic meaning given above; that is, a trusted
5902 platform enforces whatever security policy it has been assigned. It does not have to be a
5903 traditional multilevel security policy. Some trusted platforms enforce MILS policies. These
5904 policies allow the platform to be used for different levels of security at different times—while
5905 restricting use to one level at any given time. For example, a device could be used to connect to
5906 an unclassified network and process unclassified information and then later be used to connect to
5907 a classified command and control network and process SECRET information.

5908 (U//FOUO) The concept of MILS is similar to the traditional periods processing operations of
5909 processing one security level of information, wiping the system of any information (e.g., by
5910 removing disks, tapes, etc.), and then reloading it to process another level of information.
5911 However, it is not acceptable to have to physically change a GIG component (e.g., to remove a
5912 SECRET hard drive and replace it with an UNCLASSIFIED hard drive) given the need for
5913 network connectivity and communications. Thus, a MILS system must enforce a security policy
5914 that separates information and grants access appropriately, while not requiring significant
5915 reconfiguration.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5916 (U) Trusted platforms have two types of mechanisms: functions and assurances. Functions are
5917 the things that the platform actually does to enforce its security policy. Typical security functions
5918 in a trusted platform include identification and authentication of users, access controls, and
5919 auditing of security relevant events.

5920 (U) Assurance mechanisms are things used during the development and operation of the trusted
5921 platform to gain confidence that it actually will work correctly in its intended environment, and
5922 that it will not have hidden, undocumented, or unintended features that will allow the security
5923 functions to be subverted. Assurance mechanisms that can be used for trusted platforms include
5924 things like mapping levels of specification (to determine consistency in the development of the
5925 platform), adherence to software development standards and practices, and testing.

5926 (U) The earliest work in trusted platforms was carried out from the late 1960's to the early
5927 1980's. It led to the DoD Trusted System Evaluation Criteria (TCSEC), which was the DoD
5928 standard from 1985 until the late 1990s. Work from other organizations (such as NIST) and other
5929 countries (such as Canada, which published its own Canadian Trusted Computing Platform
5930 Evaluation Criteria, and the United Kingdom, Germany, France and the Netherlands, which
5931 jointly published the Information Technology Security Evaluation Criteria) led to the
5932 development during the 1990s of the harmonized Common Criteria (encapsulated in ISO
5933 Standard 15408, volumes 1-3). The Common Criteria (CC) are recognized by at least 20
5934 countries, including the U.S., and evaluation of a product against the Common Criteria is
5935 required now for use in most DoD programs.

5936 (U) The CC intend for an organization (for example, an industry standards group or an
5937 organization interested in acquiring trusted platforms) to publish a Protection Profile. A
5938 Protection Profile is a set of security functions, drawn from ISO 15408 volume 2—combined
5939 with a set of required assurance mechanisms, drawn from ISO 15408 volume 3. It represents the
5940 set of requirements that a category of IT products must meet to be useful and secure. For
5941 example, there is a Protection Profile covering Multilevel Operating Systems in Environments
5942 Requiring Medium Robustness. This would be for any operating system that is to be used in
5943 multilevel secure operations, in environments where threats are non-trivial but not severe, must
5944 meet the requirements contained in that protection profile. Customers attempting to deploy
5945 systems in those environments should not use systems that have not been successfully evaluated
5946 against that Protection Profile.

5947 ((U) The CC evaluation scheme does allow for the evaluation of products even when there is no
5948 established Protection Profile. The vendor publishes a Security Target, which is a description of
5949 the security properties and capabilities of the product, and the product is evaluated against that
5950 Security Target. Potential customers can then review the Security Target to determine if the
5951 product is useful in their solutions.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

5952 (U) The functional and assurance mechanisms covered in ISO 15408 are largely independent.
5953 However, some dependencies are known to exist. Some of these exist because it is not possible
5954 to implement one function without another one (e.g., mandatory access controls cannot be
5955 enforced unless data objects are appropriately tagged/labeled). Others exist because the
5956 approximately 30 years of experience in this area indicates that they provide equivalent and
5957 compatible security. In ISO 15408-2, dependencies among the functional requirements are
5958 identified, and Protection Profiles that include a given requirement must also include all
5959 requirements on which the initial one depends.

5960 (U) For the convenience of users, the assurance mechanisms in ISO 15408-3 have been grouped
5961 in a set of seven levels, referred to as Evaluated Assurance Level (EAL)—1 (the lowest) through
5962 EAL-7 (the highest). The intent is that each EAL-value describes a system that is usable in a
5963 specific type of environment. EAL-1 products tend to be appropriate in environments in which
5964 there is little to no threat; EAL-7 products are designed to stand up to the most rigorous threat
5965 environments known.

5966 **2.3.3.3.1.2 (U) Components**

5967 (U//FOUO) A trusted platform consists of hardware, software, and the guidance and procedures
5968 that go into using it. A trusted platform may be a COTS product (as most GIG components are
5969 expected to be) or it may be custom-designed device.

5970 (U//FOUO) Security policy enforcement can be apportioned among the components of a trusted
5971 platform in any way the developer wishes. Typically, in a commercial trusted operating system,
5972 little to no enforcement is assigned to the hardware; all of the explicit enforcement functions are
5973 put into software. The hardware is merely relied upon to operate correctly; i.e., to operate in
5974 accordance with its specifications without having any ways in which the software policy
5975 enforcement can be avoided or prevented. For other solutions—including most Government-
5976 provided Information Assurance assets—the hardware is explicitly assigned security policy
5977 enforcement responsibilities, such as cryptography, tamper resistance, etc.

5978 (U//FOUO) Trusted platforms must also include guidance/procedures for their assumed
5979 environments. For example, most commercial products do not offer strong tamper resistance.
5980 They are assumed to operate in environments in which modification or replacement of the
5981 hardware is prevented by physical and procedural security means. Operating a trusted platform
5982 outside of its assumed environment tends to negate the basis for trust in the system. Thus,
5983 guidance/procedures must be clear.

5984 **2.3.3.3.1.3 (U) Minimal Requirements**

5985 (U//FOUO) The requirements for specific trusted platforms to be used in the GIG will vary
5986 according the specific functions, roles and security policies of those platforms. However, there is
5987 a set of functions that must be supported in all cases for a platform to be considered trusted. This
5988 set includes:

- 5989 • (U//FOUO) Identification and authentication of users, subjects, and objects. Different
5990 users of the system must be identified and authenticated. Other parts of the system—e.g.,
5991 processes running as subjects; information objects contained within it—must be

5992 identified and, if appropriate, authenticated. It is important that the identification and
5993 authentication mechanisms cannot be subverted or bypassed by attackers. The strength of
5994 authentication mechanisms used for a specific platform will vary according to its uses For
5995 example, for some platforms, user ids and robust passwords will be sufficient; for others,
5996 biometrics or hardware tokens will be required. The strength of authentication
5997 mechanism required will generally be specified in the Protection Profile. If there is no
5998 Protection Profile covering this platform, the system engineering or requirements group
5999 will have to determine an appropriate level

6000 • (U//FOUO) An ability to securely initiate (i.e., boot) the trusted platform. It must be
6001 possible to boot the system into a known secure state. This includes mechanisms that will
6002 verify the integrity of the system and its components during the boot process to detect
6003 modification/tampering. For example, the trusted platform may need to verify serial
6004 numbers or private keys assigned to hardware modules to ensure that they have not been
6005 removed or replaced (although strategies must exist to replace defective modules with
6006 new replacement or upgraded modules). Similarly, it may be appropriate to digitally sign
6007 boot code such as Basic Input-Output System (BIOS) code to ensure that it has not been
6008 modified. When using modification-detection routines as part of the secure initialization
6009 process, it is necessary to ensure that the detection routines themselves cannot easily be
6010 defeated. As noted, though, it is still also necessary to allow for required upgrades as well
6011 as the replacement of failed modules

6012 • (U//FOUO) Partitioning. The trusted platform must have the ability to support different
6013 processes with different privileges. Those processes and the resources they access must
6014 be physically or logically partitioned, so that one process cannot interfere with or learn
6015 information about another process in violation of the security policy. Partitioning of the
6016 system supports MLS and/or MILS operation. It can be implemented using a virtual
6017 machine architecture, in which processes operating at one level are given access to virtual
6018 resources rather than the platform's physical resources (such as disk space, network
6019 interfaces, etc.). Partitioning usually requires that the platform have the ability to save
6020 process state, and to sanitize internal memory. It also requires that communications
6021 among processes operating at different security levels be controlled—whether
6022 simultaneously or at different times, since covert channels to leak information often exist
6023 in the ways processes interact

6024 • (U//FOUO) Access control. The trusted platform must have the ability to support an
6025 access control policy. This policy determines what subjects may access what objects, in
6026 what contexts, and in what modes. In traditional MLS operation, this policy is label-based
6027 and follows the lattice model of security originally described by Bell and LaPadula. In
6028 MILS operation, typically all subjects can access all objects in the virtual machine that
6029 represents a single security level, but the virtual machine manager tightly controls
6030 accesses that cross or go beyond a virtual machine. In other operations, the policy can be
6031 based on integrity models, non-interference models, or other models. In addition,
6032 discretionary access control policies, in which object owners determine access rights, can
6033 also be enforced

- 6034 • (U//FOUO) Auditing. The trusted platform must have the ability to track security relevant
6035 events that occur on the platform. These events will depend on the specific platform,
6036 environment, applications, and security policy to be enforced. See Section 2.7 of the
6037 Technology Roadmap for a description of Audit Management and Section 2.6 of the
6038 Technology Roadmap for a description of Computer Network Defense, which is closely
6039 related to audit and which all trusted platforms must support

6040 **2.3.3.3.1.4 (U) Implementing Trusted Platforms**

6041 (U//FOUO) There are a number of techniques that can be used to implement the functions of
6042 trusted platforms and to provide the assurance levels necessary to achieve the confidence that a
6043 trusted platform cannot be subverted. These techniques run from stronger policy enforcement
6044 mechanisms to implementation and testing techniques that increase confidence that bugs have
6045 been found. We will address these techniques in this section.

6046 2.3.3.3.1.4.1 (U) Functions

6047 (U//FOUO) Trusted platforms must identify and authenticate all users and subjects that access
6048 the system before allowing them to take any other security-relevant actions. The identification
6049 and authentication mechanisms chosen must be of appropriate strength—given the security
6050 requirements for the platform and the security mechanisms and assurances implemented for the
6051 rest of the system. Identification and authentication is discussed in detail in Section 2.1 of this
6052 document.

6053 (U//FOUO) Trusted platforms must generally implement some form of access control. This
6054 could include one or more of: Rule-based or mandatory access control; discretionary access
6055 control; role-based access control, or other forms. In the future, it may be Risk-Adaptable Access
6056 Control (RAAdAC), discussed in Section 2.2.

6057 (U) Rule-based or mandatory access control is based on labels or metadata tags associated with
6058 subjects and objects. It is non-discretionary, in that access to an object can only be granted to a
6059 subject if the tags associated with the subject and object match according to the established rules.
6060 Data owners (originators) cannot change this. Typically, these access controls are used to
6061 implement classification-based access controls, e.g., to ensure that TOP SECRET data objects
6062 are only accessed by those with TOP SECRET clearances.

6063 (U//FOUO) Discretionary access controls allow data object owners to make decisions about
6064 whether access is granted to a subject. Discretionary access controls provide a flexible
6065 mechanism, but they are vulnerable to attacks such as Trojan horses, in which a program acting
6066 as the data owner grant access without the owner's knowledge or approval.

6067 (U//FOUO) Access controls are discussed in detail in Section 2.2 of the GIG IA
6068 Capability/Technology Roadmap.

6069 2.3.3.3.1.4.2 (U) Assurance

6070 (U//FOUO) More difficult than implementing functional mechanisms is achieving some level of
6071 assurance, that is, some level of confidence that the trusted platform will actually enforce its
6072 security policy and not be vulnerable to attack. A number of mechanisms can be used to
6073 accomplish assurance, some of which are more effective than others.

6074 (U) In the Common Criteria (ISO 15408-3), assurance mechanisms are divided into seven
6075 classes:

- 6076 • (U) Class ACM: Configuration management
- 6077 • (U) Class ADO: Delivery and operation
- 6078 • (U) Class ADV: Development
- 6079 • (U) Class AGD: Guidance documents
- 6080 • (U) Class ALC: Life cycle support
- 6081 • (U) Class ATE: Tests
- 6082 • (U) Class AVA: Vulnerability assessment

6083 (U) Each of these classes has a number of mechanisms within it. Configuration management
6084 includes configuration automation, scope, and management. Delivery and operation includes
6085 requirements for secure delivery, installation, day-to-day operation, recovery, and platform life
6086 cycle support. Platform developers are required to provide guidance documents for users and
6087 operators/administrators that describe the intended environment and how to use, configure, and
6088 manage a trusted platform to meet its goals.

6089 (U//FOUO) Life cycle support mechanisms include requirements for development environment
6090 security and for flaw remediation. Security of the development environment deals with the
6091 likelihood of malicious code or hardware being inserted into a trusted platform during its design
6092 and implementation. Flaw remediation deals with how a developer addresses security flaws once
6093 they are known. This includes reporting the flaw to customers; developing fixes for it; testing
6094 those fixes to ensure that they do fix the problem but do not cause additional security issues
6095 themselves; and distributing the fixes to customers.

6096 (U) Testing includes both functional testing (e.g., making sure that a trusted platform meets all of
6097 its requirements) and independent penetration testing in which a Red Team attempts to attack a
6098 platform in an operational-type environment to look for security flaws that the development team
6099 did not contemplate. This independent testing is expanded in a vulnerability assessment, in
6100 which a developer and/or an independent Red Team analyze a trusted platform and attempt to
6101 identify all remaining flaws and vulnerabilities, so that informed choices can be made about
6102 whether to accept the vulnerabilities or to modify the system or environment to mask them.

6103 (U//FOUO) That leaves the largest and most complex class of assurance mechanisms, which are
6104 the development mechanisms. These are the mechanisms that address how the system is to be
6105 designed and implemented. They include:

- 6106 • (U) The functional specification of the trusted platform and its interfaces. This can be
6107 done in an informal style (e.g., written in natural language), a formal, mathematical way,
6108 or some combination
- 6109 • (U) The high-level design of the trusted platform. This describes the platform in terms of
6110 major subsystems
- 6111 • (U) The completeness of the implementation representation (that is, how accurately the
6112 completed trusted platform is represented by its documentation)
- 6113 • (U//FOUO) The structure and correctness of the internals of the trusted platform. This
6114 includes requirements for structure and modularity to minimize the number and size of
6115 the components that must actually be trusted and allow for full analysis of them. There
6116 are also requirements for reducing or eliminating circular dependencies, and for
6117 minimizing non-security-critical code and hardware in security-critical modules. The goal
6118 is to make the trusted parts of the trusted component as small and simple as possible. This
6119 leads to components that will be less likely to have buffer overflows, incomplete
6120 implementations, etc.
- 6121 • (U) The low-level design, which describes the trusted platform in terms of its component
6122 modules, their interfaces and dependencies. At this level, the design of the trusted
6123 platform is much more complete and much more representative of what will actually be
6124 fielded than is the high-level design described above
- 6125 • (U) A demonstration of correspondence between the different levels of documentation
6126 (e.g., showing that the high-level design and low-level design are consistent with one
6127 another, and no gaps or major new areas exist in either document)
- 6128 • (U) Security policy modeling. The purpose of this mechanism is to develop a model of
6129 the security policy to be enforced by the trusted platform and then to demonstrate that
6130 that model is actually enforced by the platform specification

6131 (U//FOUO) Together, these mechanisms can be used to show that a trusted platform has been
6132 built with an acceptable level of confidence that it does not contain security vulnerabilities such
6133 as buffer overflows, incomplete or ineffective implementations, or undocumented features that
6134 can be used to defeat security.

6135 **2.3.3.3.2 (U) Usage Considerations**

6136 (U//FOUO) Trusted platforms have been around for more than 20 years. However, they have
6137 enjoyed essentially no success in the general computing field and very little success in special-
6138 purpose processing. Largely, the reasons for this have been the significant cost of these platforms
6139 and their user-unfriendliness.

6140 (U//FOUO) It costs a tremendous amount of money to develop a strongly-secure trusted
6141 platform—typically many millions of dollars. And there is not now nor has there ever been a
6142 significant market for them. The market has typically been expressed in terms of thousands of
6143 units, at best. Thus, amortizing development costs has resulted in the charge to customers for
6144 these devices being many thousands of dollars per copy. Faced with this, most potential
6145 customers have chosen to try to find alternative solutions, and new customers have stayed away
6146 entirely.

6147 (U//FOUO) Also, because of security restrictions and other design choices, trusted platforms
6148 generally present markedly different interfaces to users than do more general purpose systems.
6149 Users quickly become frustrated at slow interfaces, limited networking, and being unable to do
6150 what they are used to do on their home or office computers. Even though U.S. Government
6151 policy required the use of a low-level trusted platform for all computers purchased after 1992,
6152 they have never caught on, and user unhappiness is a major reason why.

6153 (U//FOUO) With advances in research results and the move away from pure MLS to MILS and
6154 other simpler solutions, there is a better chance for trusted platforms now than there has been in
6155 the past. However, it will be a challenge for some time.

6156 **2.3.3.3.2.1 (U) Implementation Issues**

6157 (U//FOUO) The biggest implementation issues with trusted platforms are the cost to build and
6158 evaluate them, and the difficulty in providing an acceptable user interface. When used in many
6159 environments, trusted platforms prevent users from doing things in the way they have always
6160 done them (often for sound security reasons), and thus users will look for ways to circumvent the
6161 trusted platform or will choose other platforms entirely.

6162 **2.3.3.3.2.2 (U) Advantages**

6163 (U) The advantage of a trusted platform is that it allows a single device to be used to handle a
6164 variety of different processing needs across security domains. As the GIG causes security
6165 domains to be brought together into COIs, it will be important for some components to have a
6166 high level of trust. With a trusted component, a user can connect to unclassified sites and
6167 SECRET sites and TOP SECRET sites with the same device. This will lead to better information
6168 sharing and a clearer picture of the situation. With MLS capability, this information can then be
6169 shared across users having the same device. With a MILS solution, this sharing will be more
6170 limited, but it will still be a substantial improvement over what exists today.

6171 **2.3.3.3.2.3 (U) Risks/Threats/Attacks**

6172 (U//FOUO) No trusted platform is perfect, because we as a community do not have the
6173 technology to implement systems without bugs. All trusted platforms are subject to some
6174 security attacks. Some will exploit bugs in the design or implementation; others will go around
6175 the security policy (i.e., exploit system features that the trusted component was explicitly
6176 designed not to protect).

6177 (U//FOUO) The biggest risk in the GIG is that a trusted platform will be relied upon too much.
6178 GIG security will always require defense-in-depth. Trusted platforms will have their roles, and
6179 they can provide great advantages. But they should never be used in environments outside those
6180 described in their documentation, and they should not be relied upon to provide perfect
6181 protection against attacks.

6182 **2.3.3.3.3 (U) Maturity Level**

6183 (U//FOUO) As noted above, trusted platforms have been around for more than 20 years. For
6184 some categories of products (e.g., firewalls, gateways, special purpose IA components), they are
6185 mature technology and can be used in the 2006-2008 GIG. For other categories of products (e.g.,
6186 devices to connect to multiple levels of wireless networks; general purpose desktop computers),
6187 significant research is needed in the areas of software engineering, high-assurance computing,
6188 network security, and system evaluation. In addition, much work is needed for all types of
6189 platforms in the areas of system performance, user friendliness, and cost-effective security.

6190 (U//FOUO) For those categories of products for which the technology is well adapted (e.g.,
6191 firewalls and gateways), trusted platforms are Mature (TRLs 7 - 9). There are products which
6192 can be purchased and used today. In fact there are products that are being used within the DoD
6193 today.

6194 (U//FOUO) For other types of products, significant research is needed. The technology readiness
6195 group is near the boundary of Emerging (TRLs 4- 6) and Early (TRLs 1 - 3).

6196 **2.3.3.3.4 (U) Standards**

6197 (U) Validated non-U.S. Government Protection Profiles on trusted platforms (per
6198 <http://niap.nist.gov/cc-scheme/pp/index.html>)

6199 • (U) Trusted Computing Group (TCG) Personal Computer (PC) Specific Trusted Building
6200 Block (TBB) Protection Profile and TCG PC Specific TBB with Maintenance Protection
6201 Profile

6202 • (U) Trusted Computing Platform Alliance Trusted Platform Module PP

6203 (U) The primary standard for Trusted Platforms is the Common Criteria, ISO 15408, volumes 1-
6204 3. These documents are used as the basis for evaluation in the U.S. and approximately 20 other
6205 countries.

6206 (U//FOUO) For other, non-COTS devices, there are specific standards internal to NSA that are
6207 applied to high-assurance devices.

6208 **2.3.3.3.5 (U) Cost/Limitations**

6209 (U//FOUO) As noted above, trusted platforms tend to be very costly to develop, and costly to
6210 use. Development costs are attributed to the fact that it (at least initially) costs a lot of money to
6211 build security into a product. Typical commercial best practices cannot be used, so the
6212 development system has to be changed. In addition, evaluation of a product costs money and
6213 time. Given that the market for these trusted platforms has traditionally been very small, the
6214 development costs have to be amortized over this relatively small base, and thus the cost to users
6215 tends to be high.

6216 (U//FOUO) There is a perceived cost to the users in running a trusted platform in the GIG
6217 environment, as well. This cost is due to the fact that for security reasons the trusted platform
6218 often does not allow the users to do things the same way they've always done them.

6219 **2.3.3.3.6 (U) Dependencies**

6220 (U) As noted above, trusted platforms are dependent on a number of the other enablers:
6221 Identification and Authentication; Policy-Based Access Control; Network Defense and
6222 Situational Awareness; and Management of IA Mechanisms and Assets.

6223 **2.3.3.3.7 (U) Alternatives**

6224 (U//FOUO) The alternative to using trusted platforms is to restrict the GIG to having each device
6225 be used for a single classification level or domain of information and users. However, this
6226 defeats the GIG's vision of information sharing; it prevents RAdAC from being implemented; it
6227 drives up costs; and in general it will prevent the GIG from meeting its mission.

6228 (U//FOUO) Within the field of trusted platforms, there are a variety of possible alternatives—
6229 traditional Multi-level security; Multiple Independent Levels of Security; various other security
6230 policies to be supported. It is likely that each of these alternatives will be useful for some set of
6231 scenarios, and thus that there will be a wide variety of trusted platforms deployed in the GIG in
6232 the future.

6233 **2.3.3.3.8 (U) Complementary Techniques**

6234 (U//FOUO) Trusted platforms can be deployed along with cross-domain solutions such as a
6235 firewall and gateways to provide cost-effective solution for the GIG. Trusted platforms can also
6236 be used with other IA components to strengthen security, e.g., a trusted platform along with a
6237 QoP-capable router provides better resource allocation for the GIG that resists attacks better than
6238 routers alone.

6239 **2.3.3.3.9 (U) References**

6240 (U) ISO 15408-1: Information technology — Security techniques — Evaluation criteria for IT
6241 security — Part 1: Introduction and general model, 1999.

6242 (U) ISO 15408-2: Information technology -- Security techniques -- Evaluation criteria for IT
6243 security -- Part 2: Security functional requirements, 1999.

6244 (U) ISO 15408-3: Information technology -- Security techniques -- Evaluation criteria for IT
6245 security -- Part 3: Security assurance requirements, 1999.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 6246 (U) Johnson, R. and D. Wagner, Finding User/Kernel Pointer Bugs with Type Inference, Proc.
6247 13th USENIX Security Symposium, pp. 119-133, San Diego, CA, August 2004.
- 6248 (U) “Static Disassembly of Obfuscated Binaries,” by Kruegel, C., W. Robertson, F. Valeur and
6249 G. Vigna, Proc. 13th USENIX Security Symposium, pp. 255 – 270, San Diego, CA, August
6250 2004.
- 6251 (U) “Protection Profile for Multilevel Operating System in Environments Requiring Medium
6252 Robustness,” Version 1.22, NIAP, May 23, 2001. Available at [http://niap.nist.gov/cc-](http://niap.nist.gov/cc-scheme/pp/PP_MLOSPP-MR_V1.22.html)
6253 [scheme/pp/PP_MLOSPP-MR_V1.22.html](http://niap.nist.gov/cc-scheme/pp/PP_MLOSPP-MR_V1.22.html)
- 6254 (U) “Protection Profile for Single-level Operating Systems in Environments Requiring Medium
6255 Robustness,” Version 1.22, NIAP, May 23, 2001. Available at [http://niap.nist.gov/cc-](http://niap.nist.gov/cc-scheme/pp/PP_SLOSPP-MR_V1.22.html)
6256 [scheme/pp/PP_SLOSPP-MR_V1.22.html](http://niap.nist.gov/cc-scheme/pp/PP_SLOSPP-MR_V1.22.html)
- 6257 (U) “Controlled Access Protection Profile,” Version 1.d, NIAP, October 8, 1999. Available at
6258 http://niap.nist.gov/cc-scheme/pp/PP_CAPP_V1.d.html
- 6259 (U) “Labeled Security Protection Profile,” Version 1.d, NIAP, October 8, 1999. Available at
6260 http://niap.nist.gov/cc-scheme/pp/PP_LSPP_V1.b.html
- 6261 (U) “Copilot – a Coprocessor-based Kernel Runtime Integrity Monitor,” Petroni, N., T. Fraser, J.
6262 Molina, and W. Arbaugh, Proc. 13th USENIX Security Symposium, pp. 179 – 193, San Diego,
6263 CA, August 2004.
- 6264 (U) “Design and Implementation of a TCG-based Integrity Measurement Architecture,” Sailer,
6265 R., X. Zhang, T. Jaeger, and L. van Doorn, in Proc. 13th USENIX Security Symposium, pp. 223
6266 – 238, San Diego, CA, August 2004.
- 6267 (U) TCSEC –Department of Defense Trusted Computer System Evaluation Criteria, DoD
6268 Standard 5200.28-STD (obsolete), December 1985. Available at
6269 <http://www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html>

UNCLASSIFIED//FOR OFFICIAL USE ONLY

6270 **2.3.3.4 (U) Trusted Applications**

6271 **2.3.3.4.1 (U) Technical Detail**

6272 **2.3.3.4.1.1 (U) Definition**

6273 (U) For the purposes of this Technology Roadmap, a trusted application is an application that is
6274 relied upon while performing its functions to enforce a specific security policy. The security
6275 policy may be as simple as not being malicious or it may involve simultaneously processing and
6276 separating information from several domains (e.g., at multiple classification levels).

6277 (U) An application is simply a program with a specific task or use. That is, a database
6278 management system may be an application, or a web server, or a browser, or an e-mail program.
6279 An operating system, or other generic program without a specific task to perform, is not an
6280 application. (Trusted operating systems are addressed in section 2.3.3.2 of the GIG IA
6281 Capability/Technology Roadmap.)

6282 (U) At a minimum, a trusted application is not and cannot be subverted by malicious logic, and it
6283 does not harm other components of the GIG—including the platform(s) on which it is hosted.
6284 Depending on its specific security policy, a trusted application may also have other
6285 responsibilities, such as the separation of classified information.

6286 (U) There are different definitions of trusted applications available in the literature. Some of
6287 them vary only in syntax, others in semantics. For example, some definitions assume that a
6288 trusted application is one that simultaneously processes information at multiple level of security;
6289 i.e., the trust is conferred because of the way the application was designed and operates. Other
6290 definitions assume that a trusted application is one that a user has accepted and agrees to let run
6291 on a computer without restrictions; i.e., the trust is conferred because the user has accepted the
6292 application based on whatever evidence exists.

6293 (U) Yet another definition involves the ability of a trusted application to do harm. That is, a
6294 trusted application is one that, by its design and implementation, could subvert the security of the
6295 GIG if it unintentionally or maliciously attempts to violate the security policy of its host
6296 platform. That is, the application is trusted because it has to be; it isn't confined by a platform
6297 (e.g., an operating system) in such a way that the damage it can cause is constrained. An
6298 untrusted application, by contrast, is one that is constrained by a trusted platform (Section
6299 2.3.3.2 of this document) so that it cannot directly impact the security of the GIG by either
6300 malicious use by an attacker, or simply by error in its own implementation and operation.⁸

6301 (U) The resolution of these conflicting ideas is to use the definition cited above. That is, a trusted
6302 application is one that is assigned a security policy and is relied upon to enforce that security
6303 policy.

⁸ (U//FOUO) Note that, where availability is concerned, this means that a trusted application cannot prevent other applications or the system from meeting its security requirements. Any application can fail to provide availability for itself, simply by failing to work.

6304 **2.3.3.4.1.2 (U) Security Policies**

6305 (U) As noted above, a trusted application may have any of a variety of security policies. In the
6306 most extreme cases, a trusted application such as a trusted database management system (see
6307 [TDI] for more details) may have a significant security policy such as maintaining the separation
6308 of information at different classification levels, while at the same time allowing accesses by
6309 users with different clearance levels. At the other end of the spectrum, a trusted application may
6310 have a security policy of doing no harm. That is, the application is trusted to execute and perform
6311 its task, without carrying or being vulnerable to viruses, worms, or other malicious logic; and
6312 without being able to cause denial of service attacks on its host component or on other GIG
6313 components.⁹

6314 **2.3.3.4.1.3 (U) Host Platform**

6315 (U//FOUO) A trusted application runs on top of a host platform. This host platform consists of
6316 hardware and software (e.g., an operating system) that provides support for the application.
6317 Enforcement of the application's security policy depends directly on the host platform. The host
6318 platform provides basic facilities (e.g., storage, processing, access to printers and networks) that
6319 the application requires to enforce its security policy. The host platform can also be used as a
6320 vector for attacking a trusted application; e.g., an attacker can modify the processor to ensure that
6321 encryption routines are not called when requested, or are called with pre-determined keys and
6322 initialization vectors to prevent the proper encryption of data.

6323 (U//FOUO) Because of this, there must be an appropriate match between the security policy
6324 assigned to the trusted application and the capabilities of the host platform on which the
6325 application executes. For example, it is generally NOT acceptable to run an application
6326 providing multilevel security on a host platform that does not support strong process separation
6327 and access control—it is too easy for the application to be subverted.

6328 **2.3.3.4.1.4 (U) Requirements for Trusted Applications**

6329 (U) The specific requirements that are to be imposed on any trusted application depend on both
6330 what the application is supposed to accomplish and what its security policy is. Regardless of
6331 those factors, though, a set of minimum requirements can be established for all trusted
6332 applications. These requirements provide a basis for establishing some level of confidence that
6333 the application will enforce its security policy.

6334 (U) These requirements are levied upon the execution environment in which the application runs
6335 (e.g., the operating system and hardware on which a program runs), as well as on the application
6336 itself. Peinado, et al. [PEINADO] list the following properties that must be possessed by the
6337 application and its execution environment.

- 6338 • (U) No interference: The execution environment must provide a program that executes in
6339 it with the same underlying machine interface every time the program executes. The
6340 program must be isolated from external interference. A necessary condition is that a

⁶ (U//FOUO) Note that a trusted application is not "trusted" to work correctly. Program correctness – working correctly without error, under all input conditions – is not a function of Information Assurance as addressed here. A trusted application may still hang, or fail to complete; it may still calculate an incorrect answer or provide incorrect outputs for specific data.

6341 deterministic sequential program that does not access devices or persistent state should
6342 always reach the same result, irrespective of other programs that might have executed
6343 earlier or at the same time on the same machine

6344 • (U) No observation: The computations and data of a program should not be observable by
6345 other entities, except for data the program chooses to reveal (e.g., through IPC)

6346 • (U) Trusted paths: A program should be able to receive data from a local input device
6347 (e.g., keyboard, mouse), such that only the program and the user of the input device share
6348 the data. Data integrity must be assured. A similar requirement applies to local output
6349 devices, such as video.

6350 • (U) Persistent storage: A program should be able to store data (e.g., cryptographic keys)
6351 persistently, so that the integrity and the confidentiality of the data are ensured

6352 • (U) Communication: A program should be able to exchange data with another program,
6353 such that the integrity and the confidentiality of the data are ensured

6354 • (U) Local authentication: A local user should be able to determine the identity of a
6355 program

6356 • (U) Remote authentication: A program should be able to authenticate itself to a remote
6357 entity. For example, a corporate network administrator should be able to verify that all
6358 machines on his network are running the latest security patches and virus checker files

6359 **2.3.3.4.1.5 (U) Implementing Trusted Applications**

6360 (U) There are a number of factors that go into the implementation of a trusted application. These
6361 include how the application fits into the overall architecture of the system and how it is
6362 supported; the development process for the trusted application; and evaluation of the trusted
6363 application.

6364 **2.3.3.4.1.6 (U) Architecture**

6365 (U//FOUO) As noted above, trusted applications must rely on their host platforms to provide
6366 some level of security. At a minimum, the application must rely on the host platform to prevent a
6367 direct attack on and subversion of the application's security policy. Therefore, implementers of
6368 trusted applications must first decide on what platforms the application is to be hosted. Then,
6369 they must decide how to use the security features and security policy enforcement provided by
6370 the host platform.

6371 (U//FOUO) Most application developers want their software to be able run on a variety of
6372 hardware platforms. That maximizes the return on the investment of application development.
6373 However, from a security standpoint, that creates a complex situation, since the application
6374 developer must decide which characteristics of the family of host platforms are to be supported
6375 and clearly state these.

6376 (U//FOUO) For example, an application developer writing a trusted application to run on a
6377 server operating system can require that the server and its environment provide identification and
6378 authentication of users, tamper-resistance, and a certain level of functionality. Then, any server
6379 and environment that provide those features would be an acceptable host for the application.

6380 (U//FOUO) Or, the developer can write the application to require its own identification and
6381 authentication service and not rely on an underlying server operating system to provide that
6382 feature. This can mean that the application could run on a wider variety of platforms, but at the
6383 potential negative of being less friendly to the user (because it would require a separate log-in
6384 step).

6385 (U//FOUO) Some applications will be written for use on special purpose devices, such as NSA-
6386 certified Type 1 encryption devices. These applications can be tailored for the capabilities of the
6387 device to make use of all the features provided by that host platform.

6388 (U//FOUO) Once a platform (or set of platforms) has been selected, the developer must next
6389 decide on what features of the platform—and specifically, what security policy features—to rely
6390 on to protect the application. For example, an application running on an Multi-Level Security
6391 (MLS) or Multiple Independent Levels of Security (MILS) platform may operate at one level or
6392 it may support multiple security levels itself. As another example, an application may make use
6393 of a platform’s strong identification and authentication service by accepting the user identity
6394 preferred by the host platform, or it may choose to require its own identification and
6395 authentication process.

6396 (U//FOUO) Once all these decisions are made, the developer of the trusted application will know
6397 what is the security policy of the application, what parts of the host platform will be used, and
6398 therefore what the requirements are for the application itself. The developer can then build to
6399 those requirements.

6400 **2.3.3.4.1.7 (U) Development**

6401 (U//FOUO) A trusted application implements a security policy, even if that security policy is as
6402 simple as one that does not contain malicious code. The application must therefore be developed
6403 in such a way as to provide some level of assurance that the application does indeed enforce its
6404 security policy.

6405 (U//FOUO) The specific development environment and developmental requirements chosen by
6406 the developer will be a function of the target assurance level selected by the developer.
6407 Typically, the assurance level will be one of the seven Evaluated Assurance Levels (EAL),
6408 defined in Volume 3 of the Common Criteria [ISO15408].¹⁰ In this case, the development
6409 process must be consistent with the requirements defined for that EAL.

¹⁰ (U//FOUO) This is not required; the assurance level of an application can be whatever level is selected as appropriate for the target environment. However, the U.S. Government approved standard for assurance levels consists of the seven EALs defined in Volume 3 of ISO 15408.

6410 **2.3.3.4.1.8 (U) Evaluation**

6411 (U//FOUO) Many commercial products support the notion of a trusted application. However, in
6412 most cases, a trusted application is simply one that the user decides to trust. Sometimes, the user
6413 is told that there is an organization that digitally signed the application, so there is some level of
6414 confidence that it came from a particular commercially-reliable organization¹¹. There are known
6415 weaknesses in systems that rely on digitally signing code to assure its benign properties, and this
6416 mechanism by itself is insufficient for the GIG.

6417 (U//FOUO) In short, for GIG users, there is generally no reason for the user to believe that any
6418 particular application will enforce its security policy or will not contain malicious logic. The way
6419 to improve this situation is to have security-critical applications be evaluated. The Common
6420 Evaluation Methodology (CEM), defined under the NIAP, should be used as the baseline.
6421 Protection profiles should be defined for any common applications, in the way that [Dprof] and
6422 [Wprof] are being defined for database management systems and web servers, respectively. If an
6423 application is sufficiently unique to make a protection profile not worthwhile, then an equivalent
6424 evaluation should be done based on a security target established for that application.

6425 **2.3.3.4.2 (U) Usage Considerations**

6426 **2.3.3.4.2.1 (U) Implementation Issues**

6427 (U//FOUO) The major implementation issues have been described above. They are:

- 6428 • (U//FOUO) Determining the security policy to be enforced by the application
- 6429 • (U//FOUO) Determining the set of host platforms on which the application is to be
6430 executed
- 6431 • (U//FOUO) Determining the security policies/properties enforced by those platforms and
6432 deciding which of them will be used by the applications
- 6433 • (U//FOUO) Finally, determining the requirements to be met by the application itself to
6434 enforce the selected security policy in the assumed environment

6435 (U//FOUO) The development environment must reflect the chosen assurance level for the
6436 application. If required, an independent evaluation of the implemented application must be
6437 performed to achieve the required confidence that it enforces its security policy.

6438 **2.3.3.4.2.2 (U) Advantages**

6439 (U//FOUO) The advantage of a trusted application over a generic, untrusted application is that
6440 GIG users and designers have an established level of confidence (the assurance level) that the
6441 trusted application enforces its security policy in its presumed environment. This allows users
6442 and security officers to better understand the total risks involved in operating the GIG and also
6443 improves the security of the GIG.

¹¹ (U) The “worth” of such a signature, if any, in the commercial environment is that there is some organization that can be sued if the application turns out to damage the user’s environment.

6444 (U//FOUO) A trusted application that enforces a complex security policy, such as support for
6445 MLS or MILS, can allow enhanced operations for certain GIG users. For example, an MLS e-
6446 mail system can allow a user to communicate via e-mail with multiple communities operating at
6447 different classification levels simultaneously, eliminating the need for multiple e-mail devices
6448 and connections.

6449 **2.3.3.4.2.3 (U) Risks/Threats/Attacks**

6450 (U//FOUO) Attacks against trusted applications can come either directly against the application
6451 itself or through the underlying host platform. This is why the application developer must
6452 consider the intended host platform and the totality of the system in designing and implementing
6453 the application.

6454 (U//FOUO) An example of an attack against the application itself is feeding the application bad
6455 data. Early web servers did not filter data passed to them by clients, and it was often possible to
6456 provide a shell script program in the data field of a web form, and thus have the program
6457 executed on the web server. Thus it was necessary for web server implementers to filter all data
6458 provided by a client to ensure that no malicious logic got executed on the server's host computer.
6459 Application developers must consider similar attacks against their own applications and defend
6460 against them appropriately.

6461 (U//FOUO) An example of an attack against an application through the host platform is one in
6462 which an attacker places a keystroke logger on a computer to capture all keystrokes typed into an
6463 application. This attack can potentially recover passwords, PINs needed to unlock and use
6464 private keys, and other sensitive material without the application itself ever being aware of this.
6465 Application developers must be aware of the types of host platforms on which their applications
6466 will run and the potential attacks that can occur through those host platforms. They must make
6467 decisions about which types of attacks to defend against, and which types of attacks to simply
6468 accept. The application's documentation must make clear the decisions made by developers and
6469 the resultant risks to application users.

6470 **2.3.3.4.3 (U) Maturity Level**

6471 (U) The maturity level of trusted applications varies according to the type of application and host
6472 platform. Trusted applications that enforce simple security policies (e.g., within a single security
6473 level) have existed for several years. Standards and guidelines for developing trusted
6474 applications such as web servers, multi-level e-mail, and multi-level database management
6475 systems (DBMS) exist now or are in development. Some implementations of applications that
6476 comply with those standards and guidelines are in various stages of development.

6477 (U//FOUO) However, there is much work to do in the general field of trusted applications.
6478 Applications that work across security domains or levels; that enforce dynamic access control
6479 policies such as RAdAC; and that work across a range of general-purpose host platforms while
6480 communicating across a variety of networks, require significant amounts of research and
6481 development.

6482 (U//FOUO) In terms of timelines, the set of trusted applications that are available will increase
6483 gradually over time. Simple trusted applications exist today, and there will be a few more by
6484 2008. Self-protecting trusted applications and support for some more complex security policies
6485 will exist by 2012. Full support for complex security policies on a variety of host platforms will
6486 not exist until the 2016–2020 timeframe.

6487 (U//FOUO) The technology readiness level group assigned for trusted applications is Emerging
6488 (TRLs 4 – 6). This is an accurate reflection of the overall status of the area. As noted above,
6489 some parts of this field are already well understood, and trusted applications exist. Other areas
6490 require significant research and development.

6491 **2.3.3.4.4 (U) Standards**

6492 (U) Applicable standards for trusted applications are Protection Profiles developed against ISO
6493 15408, the Common Criteria. Because of the different security requirements, security policies,
6494 and functional requirements of different applications, it is not possible to have a generic Trusted
6495 Application Protection Profile. Rather, a different Protection Profile will need to be developed
6496 for each type of application. It may be necessary to have multiple Protection Profiles for some
6497 types of applications, depending on the possible security policies that can be assigned for that
6498 application.

6499 (U) At the time of this writing, there were no U.S. Government validated Protection Profiles for
6500 trusted applications. There are two draft profiles currently being validated, one for database
6501 management systems [DBProf] and one for web servers [WSProf].

6502 **2.3.3.4.5 (U) Cost/Limitations**

6503 (U//FOUO) The costs of trusted applications are associated with the processes that must be
6504 followed, particularly the development and evaluation processes. Trusted applications have the
6505 potential of being very expensive, particularly if custom development processes must be
6506 followed in order to achieve acceptable assurance levels. A goal is to improve the software
6507 development process so that standard commercial best practices are sufficient to develop high
6508 assurance trusted applications.

6509 (U//FOUO) If trusted applications must be evaluated, the costs of the evaluation are also a
6510 concern. A robust, efficient evaluation process must be developed.

6511 **2.3.3.4.6 (U) Dependencies**

6512 (U) Successful development of robust trusted applications with complex security policies
6513 depends on the completion or establishment of:

- 6514 • (U//FOUO) Dynamic access control policies
- 6515 • (U//FOUO) Standards for application development and evaluation
- 6516 • (U//FOUO) Understanding of the relationship between host platform security policies and
6517 trusted application security policies
- 6518 • (U//FOUO) Establishment of techniques and uniform requirements for self-protecting

6519 applications

6520 **2.3.3.4.7 (U) Alternatives**

6521 (U//FOUO) The only real alternative to trusted applications is to regard all applications as
6522 untrusted and rely upon the host platform to provide protection. Depending on the need to share
6523 information within a COI, untrusted applications constrained by host platforms may not provide
6524 sufficient functionality to accomplish a mission.

6525 **2.3.3.4.8 (U) Complementary Techniques**

6526 (U//FOUO) Trusted applications work more efficiently with trusted platforms, as that enhances
6527 the uses for trusted applications (e.g., MLS e-mail programs on MLS or MILS platforms provide
6528 greater functionality than MLS e-mail programs on single-level platforms). In addition, there is
6529 greater overall confidence in the security provided by a trusted application running on a trusted
6530 platform than there is in a trusted application running on an untrusted platform, because there is
6531 less likelihood of an attack on the application coming through the host platform.

6532 **2.3.3.4.9 (U) References**

6533 (U) [DBProf] – National Information Assurance Partnership, U.S. Government Protection Profile
6534 Database Management Systems for Basic Robustness Environments, Version 0.24, 15 December
6535 2003. Available at http://www.niap.nist.gov/pp/draft_pps/pp_draft_dbms_br_v0.24.pdf

6536 (U) [ISO 15408] – a three volume set, consisting of:

6537 (U) ISO 15408-1: Information technology -- Security techniques -- Evaluation criteria for IT
6538 security -- Part 1: Introduction and general model, 1999.

6539 (U) ISO 15408-2: Information technology -- Security techniques -- Evaluation criteria for IT
6540 security -- Part 2: Security functional requirements, 1999.

6541 (U) ISO 15408-3: Information technology -- Security techniques -- Evaluation criteria for IT
6542 security -- Part 3: Security assurance requirements, 1999.

6543 (U) [PEINADO] - Peinado, Marcus, Yuqun Chen, Paul England, and John Manferdelli, NGSCB:
6544 A Trusted Open System, Microsoft White Paper, Microsoft Corporation, Redmond, WA,
6545 available at <http://research.microsoft.com/~yuqunc/papers/ngscb.pdf>

6546 (U) Shirey, R., Internet Security Glossary, RFC 2828, May 2000. Available at
6547 <http://www.ietf.org/rfc/rfc2828.txt>

6548 (U) [TDI] – National Computer Security Center, Trusted Database Management System
6549 Interpretation of the Trusted Computer System Evaluation Criteria, NCSC-TG-021, April 1991.
6550 Available at <http://www.radium.ncsc.mil/tpep/library/rainbow/NCSC-TG-021.html>

6551 (U) [WSProf] - National Information Assurance Partnership, U.S. Government Protection Profile
6552 Web Server for Basic Robustness Environments, Version 0.41, 1 August 2003. Available at
6553 http://www.niap.nist.gov/pp/draft_pps/pp_draft_websrv_br_v0.41.pdf

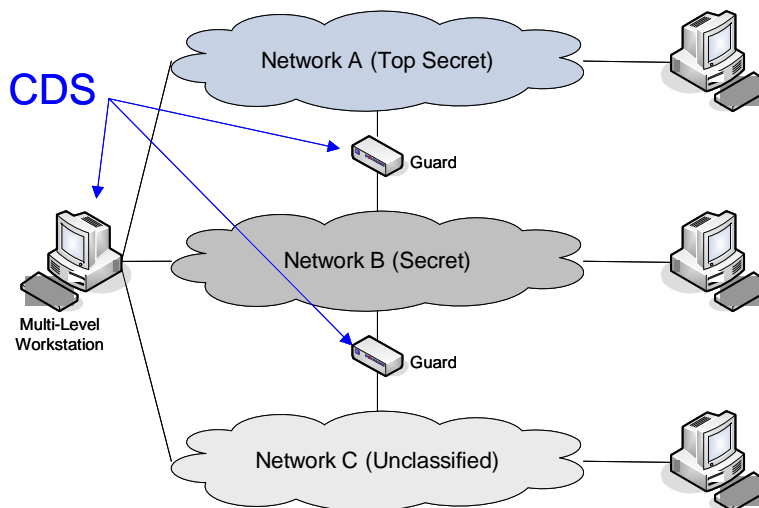
6554 **2.3.3.5 (U) Cross Domain Solution Technologies**

6555 (U//FOUO) The demands of modern warfare require that deployed forces must tightly
6556 synchronize activities in real-time within Joint and international Combined Force environments.
6557 Such synchronization necessitates an assured sharing environment where information travels
6558 seamlessly across space, time, security, and releasability domains so that the right information
6559 gets to the right warfighters in a time and place that maximizes operational effectiveness. The
6560 operational need for cross-domain information flow has been recognized for decades. However,
6561 the advent of high-speed information systems, their supporting networks, and the subsequent
6562 reliance of U.S. and multinational forces on these standardized, high-performance technologies
6563 emphasizes an urgent need for assured cross-domain solutions if organizations are to keep pace
6564 with doctrinal and operational shifts toward network-centric warfare.

6565 (U//FOUO) A cross-domain solution (CDS) is an information assurance solution that provides
6566 the ability to manually or automatically access or transfer data between two or more differing
6567 security domains (CJCSI 6211.01b). These solutions should enable the secure transfer of
6568 information across differing security domains to sustain and maximize operational effectiveness
6569 while supporting GIG security objectives. This document recognizes that while the
6570 interconnection of information systems of different security domains within and at the periphery
6571 of the GIG may be necessary to meet essential mission requirements, such connections pose
6572 significant security concerns and shall be used only to meet compelling operational
6573 requirements, not operational convenience (DoD Instruction 8540.aa (DRAFT)).

6574 **2.3.3.5.1 (U) Technical Detail**

6575 (U//FOUO) The broad definition of CDS given in CJCSI 6211.01b manifests itself in legacy
6576 systems as two distinct sets of technologies as shown in Figure 2.3-22.



This figure is (U//FOUO)

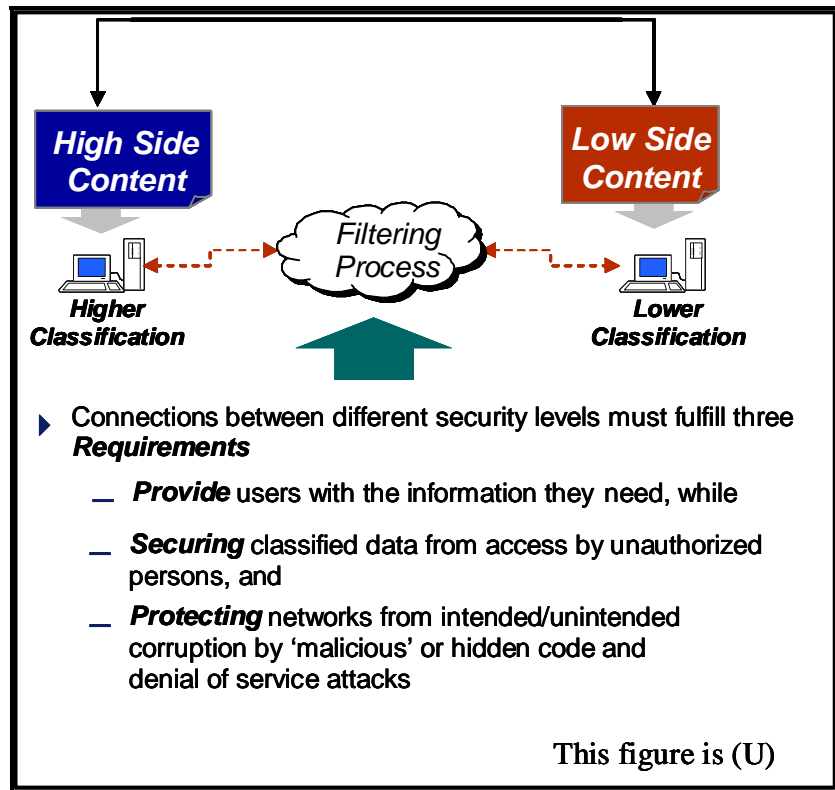
6577

6578

Figure 2.3-22 (U) Legacy Manifestation of Cross-Domain Solutions

6579 (U//FOUO) The first set of technologies falls within the category of controlled interfaces or
 6580 guards. These technologies enable the flow of information between security domains. The
 6581 second set of technologies deal with accessing multiple domains from a single node, workstation,
 6582 or server. As technology matures within each GIG IA increment, the distinction between these
 6583 two sets of technologies will blur substantially.

6584 (U//FOUO) A CDS that is used for the transfer of information could be a device or a group of
 6585 devices that mediate controlled transfers of information across security boundaries (e.g., between
 6586 security domain A and security domain B). In this usage, a CDS is trusted to allow sharing of
 6587 data across boundaries and enforces a defined security policy. CDS take into account the
 6588 following characteristics: type of data flow, direction of data flow (e.g., high to low, low to
 6589 high), human or automated review, connection protocol, number of connections) as shown in
 6590 Figure 2.3-23. Some services of a CDS include filtering, dirty word search, integrity checks,
 6591 sanitization, downgrading, and virus scanning.

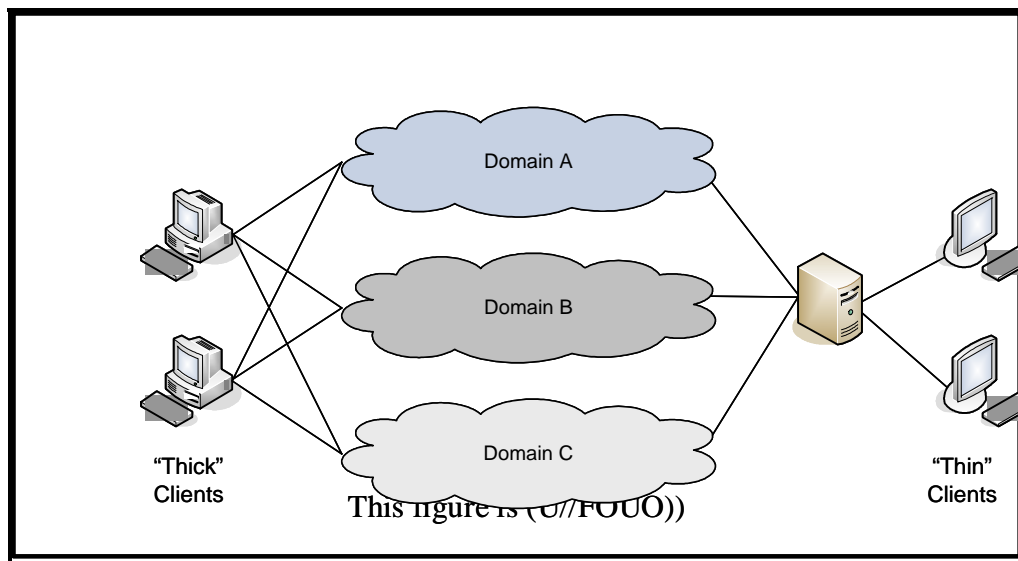


6592

6593

Figure 2.3-23: (U) Controlled Interface Example

6594 (U//FOUO) Current CDS technologies that deal with simultaneously accessing information from
 6595 multiple domains from a single location typically fall within two categories based on
 6596 functionality. The first category includes information systems that internally separate multiple
 6597 single levels (MSL) of security. Two dominant architectures supporting MSL technologies
 6598 include systems where separation is maintained locally within an edge platform (e.g., thick client
 6599 architectures such as NetTop), and systems where separation is performed remotely at a server
 6600 and clients lacking local storage access to each domain as allowed by the server (e.g., thin client
 6601 architectures such as Multi-Level Thin Client (MLTC). These two MSL architectures, shown in
 6602 Figure 2.3-24, are complementary and address information access requirements of different
 6603 operational environments.



6604

6605 **Figure 2.3-24: (U) Two MSL Architectures**

6606 (U//FOUO) The second category of information-access CDS includes information systems that
 6607 can manage multiple levels of security (MLS) simultaneously allowing, for example, cut-and-
 6608 paste between windows of a MLS workstation. An MLS database, for example, could allow
 6609 users in different security domains to access information in the same database up to their
 6610 respective clearance levels (versus accessing information in two distinct replicated databases,
 6611 one database image within each security domain).

6612 2.3.3.5.2 (U) Usage Considerations

6613 (U//FOUO) Traditional MLS architectures meet many of the requirements of CDS and have been
 6614 used successfully in limited contexts in the past. The main factors limiting broader acceptance
 6615 and deployment of MLS solutions historically have been cost and certification and accreditation
 6616 (C&A) issues with respect to mainstream operating systems and COTS applications in
 6617 widespread use throughout the DoD (e.g., Windows NT, Microsoft Office). While certain
 6618 systems (e.g., Wang XTS-300 Trusted Computer System) have been approved by NSA for MLS
 6619 processing in particular contexts, such systems have historically not supported a broad enough
 6620 variety of COTS applications and DoD-specific applications to form a viable basis for
 6621 developing a complete CDS capability. An example of an MLS solution deployed today is OSIS
 6622 Evolutionary Development (OED).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

6623 (U//FOUO) OED has an impressive accreditation and usage record in the DoD. However, there
6624 still remain issues that affect its applicability as a general MLS capability. In brief, the issues
6625 relate to vendor support for the underlying operating system, support for commercial off-the-
6626 shelf (COTS) applications (e.g., MS Office), the ability of serial links used to scale in large
6627 environments, the need to operate the system in a Sensitive Compartmented Information Facility
6628 (SCIF) with only TS-cleared personnel, and the need to run customized versions of command
6629 and control applications.

6630 (U//FOUO) The absence of fully general MLS solutions deployable on a mass scale has led to
6631 the proliferation of MSL technologies. With the exception of guard components, MSL solutions
6632 are more technically tractable and are often more straightforward to certify and accredit than
6633 general purpose MLS technologies. Examples of such solutions deployed today include Coalition
6634 Operational Wide Area Networks (COWANs) and, more recently, Combined Enterprise
6635 Regional Information Exchange System (CENTRIXS). Additional examples of systems using a
6636 MSL architecture are MLTC and Network on a Desktop (NetTop).

6637 (U//FOUO) Experience has shown that MSL architectures have proven unsatisfactory in many
6638 settings. MSL often reinforces the dependence on duplicated infrastructures—one for each
6639 security level. Duplicate infrastructure exacerbates the multiple sign-on problem (the warfighter
6640 must logon to each infrastructure separately rather than using an SSO login) and often leads to
6641 increased Space Weight and Power (SWaP). SWaP is particularly acute in many constrained,
6642 warfighting environments (e.g., ships, submarines, aircraft, ground vehicles, as well as the
6643 hauling capacity of individual troops). While certain approaches (e.g., MLTC and Keyboard,
6644 Video, Mouse [KVM] switches) have partially ameliorated the duplicate hardware issue, a
6645 number of additional drawbacks remain in MSL.

6646 (U//FOUO) At the most basic level, MSL tends to impede the efficient and timely dissemination
6647 of information. The warfighter is hampered with sometimes awkward, frequently insufficient,
6648 and at times even inappropriate workarounds. The warfighter must manually switch between
6649 disparate security domains and their associated separate databases and networks in order to
6650 accomplish the assigned mission. Correlating information between domains is challenging and
6651 may result in a warfighter having to resort to various methods, such as rekeying of data, sneaker
6652 net, and hard drive swapping, to ferry information between segregated networks. Such
6653 procedures introduce risks and delays.

6654 (U//FOUO) Such inefficiencies and risks manifest themselves in many ways from the merely
6655 inconvenient to the potentially serious. The result is that MSL drawbacks extend across many
6656 areas. Operational shortcomings include situational awareness timeliness and accuracy,
6657 situational awareness confusion, data under-classification, data over-classification, information
6658 inaccessibility, online searching difficulties, and timeliness of setup for ad-doc coalitions. MSL
6659 has a number of technical shortcomings, including guard proliferation breaking end-to-end
6660 security assumptions, lack of need-to-know enforcement, identity maintenance and correlation
6661 difficulties, collaboration difficulties, timely setup for ad-hoc coalitions, and proliferation of
6662 network interconnections.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

6663 **2.3.3.5.2.1 (U) Implementation Issues**

6664 (U//FOUO) CDSs are software/hardware implementations of networked IT. The variability and
6665 complexity of IT systems, for practical purposes, result in solutions that have an infinite number
6666 of possible configurations. Resources at all levels do not reasonably exist to adequately test and
6667 evaluate all possible configurations of CDSs and their component devices. For this reason,
6668 configuration of CDSs must be strictly controlled and enforced. Any configuration change
6669 outside of that specified could dramatically affect one or more of the three assessment areas
6670 mentioned earlier. In addition, CDSs are designed for specific purposes and to handle specific
6671 data requirements. Any changes in operational concept or data encoding will most often defeat
6672 the security provided by the CDS.

6673 (U//FOUO) Beyond configuration, control, and maintenance, there are fundamental issues that
6674 must be addressed when considering the use of CDS within an environment that aims for end-to-
6675 end security. Of principle concern is the following. A primary function of CDS is to examine
6676 content, and this instantiates numerous conflicts with technologies aimed at protecting
6677 confidentiality and integrity (e.g., FNBDT, SSL/TLS, IPsec, and HAIPE).

6678 **2.3.3.5.2.2 (U) Advantages**

6679 (U//FOUO) Until the GIG 2020 Vision is achieved, multiple security domains will exist. CDS is
6680 essential to allow information sharing among GIG entities during this transition period. CDS will
6681 remain a necessary component of interoperability within multinational forces whose technology
6682 procurement schedules are not dictated by the GIG acquisition timelines.

6683 **2.3.3.5.2.3 (U) Risks/Threats/Attacks**

6684 (U//FOUO) Any use of a CDS entails an acceptance of risk. Risks exist for both the inadvertent
6685 release of restricted data as well as the risk of malicious attack. For community-operated
6686 networks, the risk assumed by a CDS is imposed on all network operations and is not restricted
6687 to the specific system requiring the CDS. All CDSs represent some level of risk, and a CDS
6688 should not be contemplated except under compelling operational requirements. In considering a
6689 CDS for use, the specific and community risks must both be assessed before any accreditation
6690 decision is made.

6691 (U//FOUO) The risk of a CDS is comprised of more than just the connection technology. It must
6692 encompass the data/application environment and risk posture of the connecting enclave as well.
6693 The three assessment areas required for any CDS are:

- 6694 • (U//FOUO) Connection Confidence – An assessment of how confident the solution will
6695 behave as specified and is resistant to exploitation.
- 6696 • (U//FOUO) Data Potential – An assessment of the volatility of the data formats/types
6697 allowed by the connection and their potential to cause harm in the operational
6698 environment.
- 6699 • (U//FOUO) Partner Type – An assessment of how likely the connection system or its
6700 administrator would support/sponsor an attack.

6701 (U//FOUO) It is important that cross-domain solutions be understood to be holes intentionally
6702 placed within a more strict security environment for the purpose of improved information
6703 sharing. Current CDS technologies provide no ability to mitigate filtering or disclosure errors.

6704 **2.3.3.5.3 (U) Maturity**

6705 (U//FOUO) This section describes CDS that currently exist, new cross-domain solutions being
6706 considered for near term development, and systems that will require research and the use of
6707 future technologies.

6708 **2.3.3.5.3.1 (U) Current Technologies and Solutions**

6709 (U//FOUO) Currently, most operational cross-domain solutions fall into one of four technology
6710 areas. These areas are electronic mail (e-mail), fixed formatted data, file transfer, and desktop
6711 reduction. The lack of maturity of underlying IA controls causes these technologies to be
6712 considered Emerging (TRLs 5 - 7), even though many of these technologies have been
6713 demonstrated in an operational environment.

6714 (U//FOUO) E-mail cross-domain solutions scan ASCII-formatted e-mail for dirty words as
6715 messages traverse from high-side (e.g., SIPRNet) Mail Transfer Agents (MTAs) to low-side
6716 (e.g., NIPRNet) MTAs, helping prevent the disclosure of classified information. For low-to-high
6717 data flows, e-mail solutions check e-mail for malicious code. As an additional mitigation,
6718 senders and recipients can be restricted to a list of those permitted to pass messages through a
6719 particular e-mail solution. E-mail attachments are allowed to traverse security boundaries in
6720 some cases, but the file types are limited. The majority of e-mail solutions consist of the Defense
6721 Information Infrastructure (DII) Guard.

6722 (U//FOUO) Fixed format cross domain solutions transmit ASCII data that conforms to pre-
6723 defined format requirements (such as field length, allowed characters, numerical ranges). The
6724 strict formatting requirements applied to data submitted for traversal of the security boundary act
6725 as a mitigation of the concerns with unintended release and malicious content. The two main
6726 solutions that meet the needs of this category are Defense Information Systems Agency's (DISA)
6727 Command and Control Guard (C2G) and the Radiant Mercury (RM).

6728 (U//FOUO) File transfer solutions allow data files to be transmitted across the boundaries of their
6729 original security level. The files allowed to pass through the solution currently are only those
6730 considered low-risk data. This is due to the complexities of many file formats. Therefore, most
6731 solutions only allow the passage of plain ASCII text documents and image files. Additionally,
6732 high-to-low flows require human review prior to release to the low side. Currently the Imagery
6733 Support Server Environment (ISSE) Guard and the Trusted Gateway Solution (TGS) are the two
6734 solutions used most frequently for this type of data transfer.

6735 (U//FOUO) Desktop Reduction is a valid concern in business today. The need to have access to
6736 multiple networks of different security domains in one location is a necessity in many
6737 environments. The problem faced here is the user now has multiple desktop computers using up
6738 his/her desk space. In the cases of small office space or aboard a ship, space is at a premium. The
6739 idea of desktop reduction is to free physical workspace and decrease the footprint of the
6740 computers. In the example of a KVM switch, the user only requires one monitor, one keyboard,
6741 and one mouse. In other solutions presented, the user may only require one desktop computer
6742 and one monitor to access these multiple networks.

6743 **2.3.3.5.3.2 (U) New Cross-Domain Solutions Being Considered for Near-Term**
6744 **Development**

6745 (U//FOUO) Many technologies fit into this category including chat, file transfer (high risk data),
6746 Browse Down, and Content-Based Information Security (CBIS). These technologies are
6747 considered Early/Emerging (TRLs 3 - 4).

6748 (U//FOUO) Chat is a technology most people are now familiar with. Many commercial Instant
6749 Messengers can be downloaded free today from the Internet. Chat gives the user the ability to
6750 communicate with co-workers and friends in real-time by sending text. In the cross-domain
6751 world, Chat would allow a user to send text across security domains in near-real-time (allowing
6752 some latency for filtering).

6753 (U//FOUO) In the world of Cross-Domain, file transfer has always been a big issue. Although
6754 accredited solutions exist to transfer fixed file formats, there are many files prohibited from
6755 being passed through these solutions (e.g., executable files, documents with macros). The
6756 technology exists currently to filter some of these higher risk data types. One of the larger pieces
6757 of the puzzle lies in filtering Microsoft Office documents since they are so widely used.
6758 Microsoft Office documents are considered to be high risk due to all of the hidden information
6759 and executable contents (macros) which can be stored in them. With the right solution in place,
6760 business could carry on relatively seamlessly between security domains.

6761 (U//FOUO) Browse Down is a technology used to browse a lower security domain from a higher
6762 security domain network. One example of this would be to surf the Internet while attached to
6763 your classified network at the office. This would alleviate the need to purchase more hardware
6764 for the user's workspace and pull a network feed to his/her desk.

6765 (U//FOUO) CBIS is the direction many in the Cross-Domain community are going. CBIS can
6766 provide controlled access to assets based on the attributes associated with them. These attributes
6767 will include a security classification as well as a need-to-know attribute. CBIS is policy driven,
6768 which dictates a specific role to a user. After using strong Identification and Authentication
6769 (I&A) mechanisms to help enforce the access control, a user is only permitted to access files,
6770 which his/her role allows them to see. Although some of this technology exists today, it is
6771 relatively in its infancy. When this technology has matured, central repositories will be able to
6772 hold information from multiple security domains and allow CBIS to drive the policy. It is
6773 anticipated that the key Assured Information Sharing technologies developed by CBIS will be
6774 incorporated into other cross domain solutions.

6775 **2.3.3.5.3.3 (U) Future Technology and Research Needed**

6776 (U//FOUO) In general, for any mission-essential IT services within a system (security domain) a
6777 requirement will exist for that IT service to be supported across systems. Today, key IT services
6778 are e-mail, sharing files, collaboration, and web browsing. In the future key IT services are
6779 expected to include XML-based Web Services, VoIP, and others. In addition, the 2020 Vision is
6780 for a single system that can support as many security domains as needed. Given the diversity of
6781 these requirements for CDSs, to date research and development has provided solutions to only a
6782 small portion of the requirements, and for those requirements that can be satisfied with CDSs
6783 today the overall administration of the CDSs is very labor intensive. Several areas for research
6784 and development exist that would target making existing CDSs more enterprise-enabled and net-
6785 centric. The objective is to have near-term return on investment by enhancing the collaboration
6786 capabilities supported by CDS, bringing existing CDSs into compliance with standards and
6787 necessary assurance levels, and making their administration less labor intensive.

6788 (U//FOUO) Research and development is also needed to address cross-domain security issues for
6789 particular capabilities operating within an environment supporting end-to-end security. For
6790 example, research and development is needed to address the cross-domain security issues with
6791 VoIP within an environment supported by HAIPE. Likewise this applies to Web Services, and
6792 for collaboration capabilities such as virtual white boarding, shared applications, remote desktop
6793 control, audio/video conferencing, etc. This research and development will address gaps in our
6794 knowledge of how to architect cross-domain capabilities into the GIG vision.

6795 (U//FOUO) As the GIG evolves towards the 2020 Vision, CDSs as we often see them today
6796 (devices at the system boundary) will continue to evolve and exist, primarily to control the flow
6797 of information between the GIG and non-GIG systems, such as the Internet, coalition networks
6798 owned by multiple nations, national networks owned by another nation, and possibly other U.S.
6799 Government agencies. Of course, CDSs controlling information passing into and from the GIG
6800 will need to be GIG-compliant and net-centric themselves.

6801 (U//FOUO) Achieving the 2020 Vision of a single system capable of handling all types of DoD
6802 information will require that virtually all components within the GIG incorporate, to some
6803 extent, the techniques and technologies first developed and deployed at the boundaries in CDSs.

6804 (U//FOUO) For example, as we pursue research and development to establish a capability to
6805 examine Microsoft Office files for executable and hidden content, that capability will likely first
6806 appear in a CDS. Initial capabilities such as this are often complex and costly, making them
6807 unwieldy for initial deployment on every desktop. Instead, these complex and costly capabilities
6808 will appear in centralized locations that are already—basically—complex and costly, where
6809 application of the capability/checking can be assured, and where the benefit of the capability has
6810 the largest payoff. As the capability matures, it would migrate from the CDSs to the desktops
6811 themselves. To achieve the 2020 Vision, a capability such as in this example, will likely be a key
6812 requirement for protecting the GIG's confidentiality, integrity, and availability.

6813 (U//FOUO) Another example would be the ability to discern the meaning of a document from its
 6814 content. While this capability is costly and complex, it will likely be used in a CDS to detect and
 6815 prevent inappropriate information from being released/disclosed. As the capability matures, it
 6816 can migrate to the desktop so that in 2020 a person preparing a document for a given recipient
 6817 will receive a flag/notice if the tool determines from the content of the document that it would be
 6818 inappropriate for that intended recipient. These are examples of how techniques and technologies
 6819 originally implemented in CDSs can mature and then migrate from the boundary of the GIG into
 6820 components within the GIG, and thus are critical enablers to achieving the GIG 2020 Vision.

6821 **2.3.3.5.4 (U) Standards**

6822 (U//FOUO) Standards for addressing Cross Domain requirements listed in Table 2.3-12.

6823 **Table 2.3-12: (U) CDS Standards**

This table is (U//FOUO)		
Name	Description	Applicability
CJCSI 6211.02B	Defense Information System Network (DISN): Policy, Responsibilities, and Procedures	This Instruction applies to the Joint Staff, Combatant Commands, Services, Defense Agencies, DoD Field Activities, and Joint Activities. It addresses Cross Domain requirements and the policy, responsibilities, and procedures for resolving Cross Domain issues. Cross Domain connections shall be in accordance with DoD Directive 8500.1, Information Assurance, and DoD Instruction 8500.2, Information Assurance Implementation. Procedures within DoD Instruction 5200.40, DoD Information Technology Security Certification and Accreditation Process, including a risk assessment by the Cross Domain Technical Advisory Board and approval by the DISN Security Accreditation Working Group, will be followed.
DCID6/3	Protecting Sensitive Compartmented Information within Information Systems	This is a mandate for the Intelligence Community (IC). It is not applicable within the DoD unless a DoD system is connected to an IC system. The Policy portion of this Directive establishes security policy and procedures for storing, processing, and communicating classified intelligence information in information systems (ISs). It lists policies plus roles and responsibilities. The Manual portion of this Directive provides policy, guidance, and requirements for ensuring adequate protection of intelligence information. It includes a section on Controlled Interfaces, which are used for interconnected ISs, including those of different security domains.

This table is (U//FOUO)

Name	Description	Applicability
(DRAFT) DODI 8540.aa		Interconnection and Data Transfer between Security Domains. This Instruction will establish the DoD policy, responsibilities, and procedures for Cross Domain interconnections and the engineering, installation, certification, accreditation, and maintenance of such interconnections. Upon publication, it will apply to the Office of the Secretary of Defense, Military Departments, Chairman Joint Chiefs of Staff, Combatant Commands, Inspector General of the DoD, Defense Agencies, and DoD Field Activities. It applies to all DoD information systems. It includes Cross Domain Connection Request Procedures, a Cross Domain Data Transfer Generic Framework and Scenario, and Controlled Interface Characteristics.

This table is (U//FOUO)

6824 **2.3.3.5.5 (U) Cost/Limitations**

6825 (U//FOUO) Cross domain solutions are Government Off-The-Shelf (GOTS) products because
 6826 they require higher assurance levels than available commercially.

6827 **2.3.3.5.6 (U) Dependencies**

6828 (U//FOUO) Advancement of CDS technologies is heavily dependent upon the development and
 6829 management of trusted platforms and trusted applications. The success of CDS technology in
 6830 enhancing operational effectiveness depends substantially on the involvement of Programs of
 6831 Record in developing collaboration tools as well as command and control applications that are
 6832 CDS aware. The ability of CDS to enhance force protection capabilities, avoidance of blue-on-
 6833 blue engagements, and rapid dissemination of blue force indications and warnings depends
 6834 heavily on our ability to put CDS-aware capabilities directly in the hands, cockpits, and
 6835 workspaces of our warfighters.

6836 **2.3.3.6 (U) Non-Repudiation**

6837 **2.3.3.6.1 (U) Technical Detail**

6838 **2.3.3.6.1.1 (U) What is Non-Repudiation**

6839 (U//FOUO) Non-repudiation is a service used to protect against an entity that attempts to falsely
6840 deny, such as falsely denying generating a message or falsely denying receipt of information.
6841 Strictly speaking, technical non-repudiation mechanisms cannot actually prevent an entity from
6842 denying participating in an action or communication. Instead, they provide evidence that can be
6843 used to refute the repudiation claim. That is, the goal of a non-repudiation service is to provide a
6844 presumption that the entity performed the action in question and force the entity to provide
6845 strong evidence that it did not.

6846 (U) An example is in order. Suppose that Alice and Bob are in business. Bob presents a purchase
6847 request purported to be from Alice and asserts that Alice thus owes him money. However, Bob
6848 could well be forging the request himself, or it could have come from another entity entirely. It
6849 could have actually come from Alice, who now wants to disclaim it, as she does not wish to pay
6850 the money owed. A non-repudiation service would allow Bob to go to a neutral third party—such
6851 as a court—and convince it that Alice really did send the purchase request. Conversely, it would
6852 provide Alice strong proof that she did not send the purchase request.

6853 (U) As defined in the International Standards Organization's Open Systems Interconnection
6854 Reference Model (ISO/OSI 7498 part 2), there are two basic types of non-repudiation service:

- 6855 • (U) Non-repudiation with proof of origin: A security service that provides the recipient
6856 of data with evidence that can be retained and that proves the origin of the data, and thus
6857 protects the recipient against any subsequent attempt by the originator to falsely deny
6858 sending the data. This service can be viewed as a stronger version of a data origin
6859 authentication service, because it can verify identity to a third party
- 6860 • (U) Non-repudiation with proof of receipt: A security service that provides the originator
6861 of data with evidence that can be retained and that proves the data was received as
6862 addressed. This thus protects the originator against a subsequent attempt by the recipient
6863 to falsely deny receiving the data.

6864 (U//FOUO) These two services both deal with network communications, that is, the sending and
6865 receipt of a message. In the GIG, the concept of non-repudiation must be generalized to address a
6866 variety of other actions, such as over-riding security policies, granting access to classified
6867 information to entities without appropriate clearance, etc.

6868 (U) Non-repudiation has both technical and non-technical components. The technical measures
6869 involved in a non-repudiation service include:

- 6870 • (U) Authentication of the identity or identities associated with a transaction or
6871 transmission. The authentication MUST be such that, with a very high degree of
6872 confidence, only one entity can provide the correct authentication information. Typically,
6873 this is done by the use of a PKI, where each entity is assigned a private key to use for
6874 authentication/digital signature, and this key is not determinable by any attacker—given

6875 assumed efforts

6876 • (U) Integrity of the information. Once an entity has taken some action—sent or received
6877 a message; taken part in a transaction—it must not be possible for any attacker to modify
6878 the contents/records of that transaction. Typically, this is accomplished using digital
6879 signatures—the entity signs the message/transaction/ record, and any modification to that
6880 signature or record is detectable

6881 • (U) Time Stamping. One of the problems with signature-based systems is that back-
6882 dating of records/events is possible. Suppose that Alice has a private key used for digital
6883 signatures. If Alice’s key is compromised for whatever reason (e.g., she loses the token
6884 on which it is stored, along with the PIN to that token), an attacker (Mal) who now knows
6885 the private key can create various records and assign to them whatever time Mal desires.
6886 Mal can create signed records that are dated before the compromise occurred—even
6887 years earlier, if desired. To protect against this, a third-party time stamping service can be
6888 used, to indicate that a record did exist no later than a given time. Any records presented
6889 without time stamps are not considered to be protected by the non-repudiation service

6890 (U) Notarization. Even stronger than a time-stamping service is a digital notarization service.
6891 With this service, an entire transaction is certified and recorded by a neutral third party. This
6892 provides a stronger chain of evidence for the transaction.

6893 (U) As noted, there are both technical and non-technical components of a non-repudiation
6894 service, and no technical service can ever prevent an entity from attempting to deny, or
6895 repudiate, an action. Some of the grounds for denial or repudiation could include:

6896 • (U) Compromise of the key. If the authentication service is provided by means of a PKI,
6897 Alice can claim that her key was compromised (e.g., stolen), and she did not know it.
6898 Thus, she is not responsible for the transaction

6899 • (U) Weakness of the PKI. Alice can attempt to claim that her private key was known to
6900 attackers due to procedural or technical weaknesses in the PKI itself. For example, the
6901 cryptography was not strong enough, and thus an attacker figured out her private key; or
6902 the key purportedly issued to her was actually given to another entity, etc.

6903 • (U) Intent. Alice can claim that the transaction in question was not the one in which she
6904 intended to participate. For example, a worm program modified the data; what she saw on
6905 her computer screen is not the same as what is in the message. Or, an attacker broke into
6906 her computer and used her private key to sign a message without her knowledge. Or, that
6907 she did not understand the nature/content of the transaction; she merely clicked OK when
6908 presented with a confusing license agreement on her screen

6909 (U) All of these are within the legal scope of non-repudiation, but are outside the technical scope.
6910 To date, there is essentially no case law that exists to guide system designers/evaluators in
6911 determining what would happen in each of these situations, and what they should do to defend
6912 against them. Thus, any non-repudiation service deployed in the GIG should be regarded as
6913 providing technical non-repudiation only and not regarded as providing any basis for the
6914 resolution of a legal dispute.

6915 **2.3.3.6.1.2 (U) Providing Non-Repudiation**

6916 (U//FOUO) In the GIG, non-repudiation is required in conjunction with the TPPU model. The
6917 non-repudiation service will be applicable to the source and receipt of posted data.

6918 (U//FOUO) Trust of GIG data is associated with the source of the data, particularly since a large
6919 number of users may post data of varying confidence. Thus, any user of the data must reliably
6920 know the source of the data in order to be able to use it effectively. Where proof of source may
6921 be needed, non-repudiation should be applied to the data.

6922 (U//FOUO) Traditional application level non-repudiation services should also be available
6923 outside the scope of the TPPU model. Certain security critical events will require authorization
6924 by a third party. Non-repudiation evidence of the source or the authorizer of the events will be
6925 useful for the investigation of security incidents. GIG security policy will identify certain events
6926 as security critical. For example, an access that violates traditional mandatory access control may
6927 be identified as a security critical event that requires authorization by a third party.

6928 (U//FOUO) There are three steps in the non-repudiation service: (1) a request for the service
6929 (either implicit or explicit), (2) the creation and storage of the non-repudiation evidence, and (3)
6930 the retrieval of the evidence, either to assess its acceptability for future non-repudiation or to
6931 actually refute a repudiation claim. Requests for service are typically handled in the specific
6932 application requiring non-repudiation.

6933 (U//FOUO) We will now address the technology requirements for the components involved in
6934 creation and storage of non-repudiation evidence.

6935 **2.3.3.6.1.2.1 (U) Authentication**

6936 (U) A fundamental requirement for non-repudiation is to be able to authenticate the entity
6937 involved in the transaction. Authentication helps to ensure that the entity involved is the one
6938 expected to be involved.

6939 (U//FOUO) As noted above, a major requirement to achieve non-repudiation is that the
6940 authentication process is very strong. If an attacker can successfully authenticate as another
6941 entity, then non-repudiation cannot be provided. For this reason, non-repudiation services are
6942 typically based on public-key infrastructures. Authentication is based on possession of a token
6943 containing a private key, as well as knowledge of the PIN associated with that token, or with one
6944 or more specific biometric properties used to unlock the token. Authentication using passwords
6945 is not acceptable for non-repudiation systems, as they are too weak and easily defeated. For
6946 example, if Bob wishes to repudiate a transaction, he could simply post his password in a public
6947 location, and thus show a strong possibility that it was not he involved in the transaction.

6948 (U//FOUO) For the threshold (2008) GIG instantiation, any application requiring a non-
6949 repudiation service must require authentication based on a token, such as the DoD Common
6950 Access Card (CAC) and with a PIN or biometric property required in association with the token.
6951 As this is already available, the threshold GIG should be able to meet its authentication
6952 requirements for non-repudiation.

6953 (U//FOUO) Future iterations of the GIG will require stronger versions of the token. For example,
6954 the DoD should advance to a Class 5 PKI for tokens to be used for non-repudiation in the
6955 objective GIG. See section 2.7 for a description of the issues related to the DoD PKI.

6956 2.3.3.6.1.2.2 (U) Integrity

6957 (U//FOUO) Once a transaction has occurred, or a message has been sent or received, the record
6958 of that transaction or message must be preserved. In order for a non-repudiation service to be
6959 provided, it must not be possible to modify the transaction from the time it is created, without
6960 that modification being detectable. For example, if Alice creates a message saying, “Pay Bob
6961 \$100.00”, it must not be possible for anyone, including Alice or Bob, to change the message to
6962 “Pay Bob \$1.0000” or “Pay Bob \$10000” or “Pay Fred \$100.00” without the change being
6963 detectable.

6964 (U//FOUO) This protection against undetected modification is referred to as an integrity service.
6965 Integrity is a mandatory requirement for non-repudiation to be provided.

6966 (U//FOUO) Integrity can be provided through a number of different mechanisms. One common
6967 mechanism is through a digital signature. The record (transaction, message) is hashed, and then
6968 the hash is digitally signed. Anyone, using only publicly available information (e.g., the public
6969 signing key, and the hashing/signature algorithms used) can verify that the purported record has
6970 not been changed by validating the digital signature on it. If the hash value is different, the
6971 record has been changed and must not be regarded as valid. If the digital signature cannot be
6972 verified, the association of the record with the purported sender must not be regarded as valid.

6973 (U//FOUO) A second way to provide integrity is to use Message Authentication Codes (MACs)
6974 and specifically, Hashed Message Authentication Codes (HMACs). In an HMAC, a shared
6975 secret, such as an AES symmetric key, is known by both Alice and Bob, but no one else. The
6976 shared secret is added to the record, and then the entire quantity is hashed. The integrity of the
6977 message can be validated by anyone who knows the shared secret, simply re-calculate the hash
6978 given the purported record. If it validates, the record was created by someone who knows the
6979 shared secret; if not, the record has been modified and must not be regarded as valid.

6980 (U//FOUO) Both of these methods have potential weaknesses. In the digital signature method, if
6981 the private signature key is compromised, anyone can create a new record saying whatever the
6982 attacker wants, hash, and sign it, and it will be accepted as legitimate by anyone validating the
6983 record. Possession of a signed record in that case indicates that the record has not been changed
6984 since it was generated, but it does not prove anything about who generated the record, or when,
6985 nor indeed show that Alice or Bob had anything to do with the record.

6986 (U//FOUO) The HMAC method is vulnerable to compromise of the shared secret (i.e., the
6987 symmetric key). If Mal knows the shared AES key used by Alice and Bob, Mal can create
6988 whatever records he wants. This prevents anyone from making valid statements about whether
6989 Alice or Bob is responsible for a record.

6990 (U//FOUO) In addition, both methods are vulnerable to weaknesses in or attacks against the hash
6991 algorithm used. If it is possible to invert a hash (i.e., given a hash value, find a valid message that
6992 results in that hash value) then an attacker could create or modify records undetectably.

6993 2.3.3.6.1.2.3 (U) Time-Stamping

6994 (U//FOUO) As noted above, non-repudiation services are vulnerable to after-the-fact attacks,
6995 such as the compromise of a private signature key. An attacker, Mal, who learns Alice's private
6996 signature key can create records and then back-date them. Mal can, for example, create records
6997 indicating Alice promised to pay him money several years ago.

6998 (U//FOUO) This attack is particularly worrisome in the context of non-repudiation, in situations
6999 in which Alice may want to repudiate a record. That is, Alice may promise to pay Bob a large
7000 sum of money, but then want to back out of the obligation. Alice may even try to deliberately
7001 disclose her private signature key. By showing that the key was compromised, she can then cast
7002 doubt as to whether she was the real originator of the record, and thus may be able to avoid her
7003 obligation.

7004 (U//FOUO) To combat this attack, a system can employ trusted time-stamp and notary services.
7005 These services support the non-repudiation service by supplying proof of when information was
7006 signed. In a trusted time-stamp service, a neutral but trusted third party creates a record of when
7007 some specific record existed. That is, the time-stamping service certifies (e.g., through a digital
7008 signature of its own) that a record, R, of Alice's actions existed at time T. Later on, if there is a
7009 question about the validity of some action, this record can be consulted. For example, if Alice's
7010 private key is compromised, and a record R' is produced that is dated before time T, but there is
7011 no time-stamp of R' from the time-stamping service, R' will be rejected as invalid. However, if
7012 Alice tries to repudiate her original record R by showing that her private key was compromised,
7013 but the time-stamped record existed before the compromise, then the validity of the record would
7014 be upheld.

7015 (U//FOUO) In order to provide a time-stamping service, a number of items are needed. First, the
7016 time-stamping service needs access to a clock whose accuracy is accepted by all parties. (That is,
7017 it should not be possible to manipulate the clock to deliberately set the time ahead or back.
7018 Similarly, the clock drift must be acceptably small. The acceptable drift will depend on specific
7019 applications—in some cases, millisecond accuracy will be required; in others, it will acceptable
7020 if only the day is correct.)

7021 (U//FOUO) Second, the time-stamping service must have a very strong digital signature
7022 capability. Typically, this would be a more secure digital signature capability (e.g., longer private
7023 key length; more tamper-resistant signing module; higher-assurance procedures) than regular
7024 system users.

7025 (U//FOUO) Third, there must be a way to securely store and retrieve time-stamped records. Even
7026 if the records cannot be manipulated without detection, no useful service has been provided if
7027 they cannot be found and used when needed.

7028 (U) Some work on time-stamping standards and requirements has been done. For example, the
7029 IETF has developed RFC 3161, Internet X.509 Public Key Infrastructure Time-Stamp Protocol
7030 (TSP). The European Technical Standards Institute (ETSI) has also developed a number of
7031 standards relating to time-stamping; for example see ETSI ES 201 733, "Electronic Signature
7032 Formats".

7033 (U//FOUO) The basic technical requirements of time-stamping can be met with current
7034 technology, such as PKI-based digital signatures. Improving the service requires a stronger PKI,
7035 such as a future, higher-assurance version of the DoD PKI. Other improvements in time-
7036 stamping all rely on stronger procedures and personnel security.

7037 2.3.3.6.1.2.4 (U) Notarization

7038 (U) Time-stamping provides third party evidence that a particular record existed at a specific
7039 time. A stronger service is digital notarization. Notarization adds to the time-stamping service by
7040 generating and preserving authenticating evidence, such as digital signatures, associated X.509
7041 certificates, and related certificate validation data (e.g., a validation path or an On-Line
7042 Certificate Status Protocol transcript). The authentication evidence for a record may itself be
7043 signed, time-stamped, and stored for future use.

7044 (U//FOUO) Notarization, thus, shows the complete state of the system—to the extent that it was
7045 knowable—when a specific record was generated. Notarization not only shows that the record
7046 existed at time T, but also that at time T, the certificates used were not compromised or revoked,
7047 and that the purported user had been successfully authenticated. Any other relevant system state
7048 can also be captured.

7049 (U//FOUO) As with time-stamping, a number of items are needed for a notarization service to be
7050 provided. First, it requires time-stamping. Second, the notarization service must have a very
7051 strong digital signature capability. Typically, this would be a more secure digital signature
7052 capability (e.g., longer private key length; more tamper-resistant signing module; higher-
7053 assurance procedures) than regular system users. Third, the notarization service needs reliable
7054 access to current system information (e.g., certificates; Certificate Revocation Lists or OCSP
7055 responses; authentication information). Finally, there must be a way to securely store and
7056 retrieve notarized records.

7057 2.3.3.6.2 (U) Usage Considerations

7058 (U//FOUO) The decision to deploy a non-repudiation service, and what type of service to deploy,
7059 will be influenced by a number of factors. These include the costs of service deployment
7060 (including system and connectivity costs, as well as costs in terms of the number of people
7061 required to install and maintain the service, and the performance costs involved in the actual
7062 operations), and the benefits gained by deploying the non-repudiation service.

7063 2.3.3.6.2.1 (U) Implementation Issues

7064 (U//FOUO) There are a number of implementation issues that must be considered when
7065 deploying a non-repudiation service. These directly affect the cost, staffing levels, and level of
7066 security required.

7067 (U//FOUO) First, the appropriate level of authentication must be selected. A non-repudiation
7068 service depends completely on the correct authentication of each entity (e.g., each user, group,
7069 process). If the authentication system selected is weak (e.g., user-identifier and passwords), then
7070 it will be relatively easy to defeat the non-repudiation service. An attacker can simply guess a
7071 user's password, or a user attempting to repudiate an action can simply post his password on a
7072 public repository. Stronger authentication systems, such as those based on one-time passwords,
7073 hardware tokens, or biometrics, provide better security for a non-repudiation system but are more
7074 costly to implement. Authentication systems are described in detail in Section 2.1 of this
7075 document.

7076 (U//FOUO) Another issue impacting non-repudiation is key management. Whether symmetric
7077 cryptography or public-key approaches are chosen, there must be some way to securely generate
7078 the keys/shared secrets, distribute them to the proper users, revoke them when necessary, and in
7079 general manage these important data items. Key Management is described in detail in Section 2.7
7080 of the Roadmap.

7081 (U//FOUO) Appropriate time-stamp and notarization services must be deployed, if required.
7082 Access to sufficiently accurate clocks must be secured, and servers providing the time stamp and
7083 notarization functions must be deployed. Sufficient access (e.g., 24/7 uptime with a minimum
7084 response time of X; or whatever other metric is required) to these services must be provided.
7085 This will create support, configuration, and maintenance issues.

7086 (U//FOUO) Records storage and retrieval must be provided. The purpose of a non-repudiation
7087 service is to be able to prove to a third party, if required, that an entity did or did not participate
7088 in some event. Depending on exactly what parameters are chosen, the records must be stored for
7089 some period of time, with access available within a given level of time when required, and strong
7090 security to protect the records from modification or deletion.

7091 (U//FOUO) The decisions made for each of these issues have implications in the number of
7092 people needed to operate the system; the trust and skill level that are required by those people;
7093 the degree of access and backup required for the systems that implement the function; and other
7094 management aspects. All of these impact the cost of implementing a non-repudiation service, and
7095 the strength that that service provides.

7096 **2.3.3.6.2.2 (U) Advantages**

7097 (U) The biggest single advantage to a non-repudiation service is that, if implemented properly, it
7098 can provide a strong level of accountability for individual actions. It will be extremely difficult
7099 for an entity to falsely deny participation in some event (e.g., there will be strong records that
7100 Bob did access particular data, or sent a message, or received a message).

7101 **2.3.3.6.2.3 (U) Risks/Threats/Attacks**

7102 (U//FOUO) There are two primary failure modes of a non-repudiation service. One is that an
7103 entity can successfully repudiate participation in an event in which that entity really did
7104 participate. The other is that an entity can be wrongly blamed for participating in an even in
7105 which that entity did not participate.

7106 (U//FOUO) The risks to the non-repudiation service that would allow either of these failure
7107 modes to occur have largely been discussed above. They include:

- 7108 • (U//FOUO) Compromise of a private key or shared secret, allowing attackers to forge or
7109 modify records
- 7110 • (U//FOUO) Failure of authentication mechanisms, allowing an attacker to successfully
7111 assume an identity
- 7112 • (U//FOUO) Failure of the integrity mechanism, allowing undetected modifications to
7113 records after they have been created
- 7114 • (U//FOUO) Failure of the personnel or procedural security mechanisms, allowing
7115 attackers access to the system or causing records to not be available for examination
7116 when required
- 7117 • (U//FOUO) Insufficient recording, time-stamping, or notarization services, allowing an
7118 entity to successfully repudiate an action by, for example, deliberately compromising a
7119 private key or shared secret.

7120 (U//FOUO) The biggest risk to a non-repudiation service at this time is that it will be deemed not
7121 sufficient by legal authorities when it is required. This can only be solved by working through a
7122 number of cases, and developing a body of case law that shows clearly what is sufficient and
7123 what is not sufficient for a true non-repudiation service.

7124 **2.3.3.6.3 (U) Maturity Level**

7125 (U//FOUO) As noted above, the technical requirements for a robust non-repudiation service can
7126 be met today. The issues involved in setting up such a service are mostly legal. There is no legal
7127 precedent for what is minimally required or acceptable, and very little indication from the legal
7128 community as to what is acceptable. For example, the American Bar Association's Information
7129 Security Committee has declined to set standards or make recommendations on what is
7130 acceptable under U.S. laws for non-repudiation systems. Technical people, such as system and
7131 application developers, are making their best guesses as to requirements. However, under U.S.
7132 laws, any entity can always attempt to deny or repudiate any action, and then it becomes a matter
7133 for the courts to determine whether the technical measures provided were adequate to prevent a
7134 successful false denial. Once a body of case law has been established, it may well be possible to
7135 set more concrete technical standards.

7136 (U//FOUO) Non-repudiation technology is considered to be Mature (TRLs 7 – 9). As noted
7137 above, the technical solutions are known, although individual technical protections could be
7138 strengthened. The major developments needed are in the process and legal arenas.

7139 **2.3.3.6.4 (U) Standards**

7140 (U) The standards that address the technical measures required to provide a non-repudiation
7141 service include the ISO's 3-part standard 13888 and the European Technical Standards Institute's
7142 "Electronic Signature Formats" work. Specific references are listed in Table 2.3-13.

7143

Table 2.3-13: (U) Non-Repudiation Standards

This table is (U)	
Name	Description
ETSI ES 201 733	European Technical Standards Institute, "Electronic Signature Formats", 2000. Available at http://webapp.etsi.org/exchange/folder/es_201733v010103p.pdf
ISO 13888-1	International Standards Organization, "IT security techniques -- Non-repudiation -- Part 1: General", 2004
ISO 13888-2	International Standards Organization, "Information technology -- Security techniques -- Non-repudiation -- Part 2: Mechanisms using symmetric techniques", 1998
ISO 13888-3	International Standards Organization, "Information technology -- Security techniques -- Non-repudiation -- Part 3: Mechanisms using asymmetric techniques", 1997.
This table is (U)	

7144 2.3.3.6.5 (U) Cost/Limitations

7145 (U//FOUO) As noted in the Implementation Issues section, above, the costs of a non-repudiation
 7146 system are largely driven by the choices made in how strong the system is to be. Costs can be
 7147 quite large, if real-time access to stored records from several years ago is required and if
 7148 solutions are chosen that require highly-trusted system operators with a very high skill level.
 7149 Other cost factors include the strength of the authentication system and the key management
 7150 solution required.

7151 (U//FOUO) The single biggest limitation of a non-repudiation system is that an entity can always
 7152 attempt to deny having done something, and the legal system may or may not accept the
 7153 evidence provided by the non-repudiation system.

7154 2.3.3.6.6 (U) Dependencies

7155 (U) As noted above, a non-repudiation service depends on the proper implementation of a user
 7156 authentication service, a data integrity service, and a time-stamping or digital notary service. In
 7157 addition, non-repudiation depends on system access controls and system integrity, and on the
 7158 proper enforcement of system procedures and processes to prevent modification or deletion of
 7159 records.

7160 2.3.3.6.7 (U) Alternatives

7161 (U) There are some alternatives into how a non-repudiation service can be provided. It can be
 7162 based on digital signatures from a PKI. It can make use of MACs and HMACs. It can use time-
 7163 stamping, or digital notary services. The strength and robustness of the service needed will
 7164 determine which choices are needed.

7165 (U) If what is desired is a way of proving to a neutral third party that one or more record is valid,
 7166 or that an entity did or did not participate in a transaction, there is no alternative to a non-
 7167 repudiation service.

7168 **2.3.3.6.8 (U) Complementary Techniques**

7169 (U//FOUO) Non-repudiation systems can be used in combination with existing authentication
7170 and accountability systems to provide a stronger level of user accountability. Where the technical
7171 measures provided by a non-repudiation service are deemed to be insufficient, they can be
7172 combined with stronger procedural requirements of personnel security requirements to provide a
7173 system of the necessary strength.

7174 **2.3.3.6.9 (U) References**

7175 (U) ETSI ES 201 733: European Technical Standards Institute, Electronic Signature Formats,
7176 2000. Available at http://webapp.etsi.org/exchangefolder/es_201733v010103p.pdf

7177 (U) ISO/OSI 7498-2: International Standards Organization, Information processing systems --
7178 Open Systems Interconnection -- Basic Reference Model -- Part 2: Security Architecture, 1989.

7179 (U) ISO 13888-1: International Standards Organization, IT security techniques -- Non-
7180 repudiation -- Part 1: General, 2004.

7181 (U) ISO 13888-2: International Standards Organization, Information technology -- Security
7182 techniques -- Non-repudiation -- Part 2: Mechanisms using symmetric techniques, 1998.

7183 (U) ISO 13888-3: International Standards Organization, Information technology -- Security
7184 techniques -- Non-repudiation -- Part 3: Mechanisms using asymmetric techniques, 1997.

7185 (U) RFC 2104: Krawczyk, H., M. Bellare and R. Canetti, HMAC: Keyed-Hashing for Message
7186 Authentication, February 1997. Available at <http://www.ietf.org/rfc/rfc2104.txt>

7187 (U) RFC 3126: Pinkas, D.; J. Ross and N. Pope, Electronic Signature Formats for long term
7188 electronic signatures, September 2001. Available at <http://www.ietf.org/rfc/rfc3126.txt>

7189 (U) RFC 3161: Adams, C., P. Cain, D. Pinkas and R. Zuccherato, Internet X.509 Public Key
7190 Infrastructure Time-Stamp Protocol (TSP), August 2001. Available at
7191 <http://www.ietf.org/rfc/rfc3161.txt>

7192 **2.3.4 (U) Protection of User Information: Gap Analysis**

7193 (U//FOUO) Table 2.3-14 is a matrix listing basic requirements for secure voice compared with
 7194 the secure voice-related technologies described in previous sections. Their adequacy of the
 7195 technologies to meet the 2008 attributes is shown. Some of the IA attributes do not have RCD
 7196 capabilities mapped to them because they are below the detail specified in the RCD.

7197 **Table 2.3-14: (U//FOUO) Secure Voice Technology Gap Analysis**

This Table is (U//FOUO)							
		Technology Category					Required Capability (attribute from RCD)
		FNBBDT	Interop / Gateways	FNBBDT Voice over IP	VoIP Call Control	IP Encryption	
IA Attributes	Type 1 End-user to End-user Confidentiality		N/A		N/A		IAAU3, IAAU4, IACNF1-IACNF5, IACNF7, IACNF17, IANCM1, IANCM11, IANCM12
	Authentication		N/A				IAAU25, IANCM8, IANCM9, IANCM14
	Data Integrity		N/A				IAINT1, IAINT3, IANCM3, IANCM7, IANCM13
	Anti-replay protection		N/A				
	Bit-error Tolerance						
	Traffic Flow Security						IACNF8, IANCM2
	Dynamic Routing	N/A	N/A	N/A	N/A		
	QoS/PoS Support	N/A	N/A	N/A	N/A		IAAV1, IAAV2, IANCM4, IANCM5, IARC01-IARC03, IARC05
	Dynamic IP Addresses	N/A	N/A	N/A	N/A		
	Resource-Constrained Implementation						
	Black Media Gateway Capability				N/A		

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)							
		Technology Category					Required Capability (attribute from RCD)
		FNBDT	Interop / Gateways	FNBDT Voice over IP	VoIP Call Control	IP Encryption	
IA Attribute	Crypto Sync Maintenance		N/A	N/A	N/A		
	Denial of Service Protection						
	Multipoint Operation					N/A	
	Key Management		N/A				IAKCM1, IAKCM3- IAKCM6, IAKCM15, IAKCM16, IAKCM18, IAKCM23, IAKCM32, IAKCM35, IAKCM38, IAKCM39, IAKCM41, IAKCM43, IAKCM45, IAKCM47, IAKCM48, IAKCM50, IAKCM53
	Clear-to-Secure Transition					N/A	
	Mobile Environment Support						
	Electronic Rekey		N/A				IAKCM44
	This Table is (U//FOUO)						

UNCLASSIFIED//FOR OFFICIAL USE ONLY

7198 Table 2.3-15 reflects the gap analysis for the non-real-time application layer technologies (i.e.,
 7199 traditional layered application security , session security, and web services security. The
 7200 mapping of RCD attributes to the IA Attributes will be provided in a future release.

7201 **Table 2.3-15: (U//FOUO) Gap Analysis for Non-real-time Application Layer Technologies**

This Table is (U//FOUO)					
		Technology Categories			Required Capability (attribute from RCD)
		Traditional Layered Application Security	Session Security	Web Services Security	
IA Attributes	Confidentiality				
	Integrity				
	Authentication				
	Labeling				
	Access Control				
	Persistent Security				
	Standards Mature				
	Products Available				
	Technology Deployed				
	This Table is (U//FOUO)				

7202

7203 The gaps identified in Table 2.3-16 are based upon an investigation of warfighter requirements.
 7204 The assumption is that CDS technologies are to be used to meet compelling operational
 7205 requirements. These requirements are categorized according to warfighter objectives, warfighter
 7206 protection, and environment (security, physical, operational, etc.). The technological capabilities
 7207 available to meet these requirements were categorized by interdomain transfer (i.e., guards),
 7208 multiple domain access via clients and servers, and software applications (voice, collaboration,
 7209 command and control, situational awareness, etc.) with multiple-domain capabilities. Supporting
 7210 technologies (e.g., trusted platforms) not specifically applied to CDS will be discussed in their
 7211 respective enabler descriptions.

Table 2.3-16: (U//FOUO) CDS Technology Gap Assessment

This Table is (U//FOUO)					
		Technologies			RCD Capability
		Multiple-Domain Servers and Clients	Guards and Controlled Interfaces	CDS-Aware Applications	
Warfighter Objectives	Coordination				IAAV4, IAAV8, IACNF16
	Planning				IAAV4, IAAV8, IACNF16
	Task Dissemination				IAAV4, IAAV8, IACNF16
	Intel Assessment				IAAV4, IAAV8, IACNF16
Warfighter Protection	Indications and Warnings				IAAM8, IAAV4, IAAV8, IACNF16
	Combat ID				IAAM8, IAAV4, IAAV8, IACNF16
Warfighter Environment	Constrained Resources				IAAUD9, IAAV4, IAAV15, IAAV17
	Cognitive Workload				IACND8, IACND20, IACM11, IAIAC11, IAIAC12, IAPOL1
	Dynamic Participation				IAAM6, IAAM7, IAAM8, IACND9, IACNF15, IAIAC12, IAKCM15, IAKCM29, IAKCM33, IAKCM34, IAKCM53, IAPOL1, IARC05
	Security Environment				IAAC4, IAAC5, IAAC6, IAAM4, IAAM11, IAAM12, IAAU12, IAAUD1, IAAUD2, IAAUD3, IAAUD7, IAAUD9, IACM1, IACM5, IACM11, IACND10, IACND12, IACNF1, IACNF2, IACNF3, IACNF4, IACNF5, IACNF7, IACNF11, IACNF12, IACNF13, IACNF16, IACNF17, IAFM1, IAFM2, IAFM3, IAFM4, IAIAC3, IAIAC7, IAIL1, IAIL3, IAIL4, IAIL6, IAIL13, IAIL15, IAIL19, IAIL20, IAIN1, IAIN2, IAIN4, IAIR1, IAIR3, IAKCM29, IAKCM36 IAKCM30, IANMP5, IANRP1, IANRP2, IANRP3, IAPOL1, IAPOL3, IARC02, IARC03, IARC04, IAUAM8
	Remote Support				IAAV17, IAPOL1
This Table is (U//FOUO)					

7213 **2.3.5 (U) Protection of User Information: Recommendations and Technology Timelines**

7214 (U//FOUO) The following gaps have been identified in the Protection of User Information
7215 Enabler. Without these, the benefits to be gained by this Enabler cannot be fully satisfied.

7216 **2.3.5.1 (U) Data-in-Transit**

7217 (U) The technology gaps for Data-in-Transit have been categorized as the following types—
7218 Standards, Technology, and Infrastructure.

7219 **2.3.5.1.1 (U) Standards**

7220 (U) The following gap areas have been identified in standards associated with Secure Voice
7221 applications:

- 7222 • (U) Standards for providing end-to-end QoS for IP systems, specifically mechanisms for
7223 allowing QoS information to traverse red/black boundaries
- 7224 • (U//FOUO) HAIPE standard updates to support dynamic routing in a multi-homed
7225 environment, QoS, dynamic black IP addresses, mobility, end-system implementations,
7226 resource-constrained implementations, and low-bandwidth high BER environments
- 7227 • (U) Standards for providing interoperability between Secure Voice over IP systems and
7228 Voice over Secure IP systems
- 7229 • (U//FOUO) Standards defining a common interoperable implementation of FNBDT over
7230 IP networks, including call control, gateway operation, and user media details
- 7231 • (U//FOUO) Standards defining FNBDT multipoint operation (conferencing, net
7232 broadcast applications)
- 7233 • (U//FOUO) Standards defining additional voice coders for FNBDT systems on specific
7234 GIG sub-networks
- 7235 • (U//FOUO) Standards defining implementation and enforcement methods for applying
7236 Quality of Protection mechanisms to secure voice data
- 7237 • (U//FOUO) Standards allowing Priority Service for authorized voice users

7238 **2.3.5.1.2 (U) Technology**

7239 (U//FOUO) The following gap areas have been identified in technologies associated with Secure
7240 Voice applications:

- 7241 • (U//FOUO) Secure VoIP-enabled gateways
- 7242 • (U//FOUO) Secure multipoint voice operation (conferencing, net broadcast applications)

7243 (U//FOUO) Specific areas related to trusted applications requiring research include:

- 7244 • (U//FOUO) Linkage between a security policy enforced by the trusted application and the
7245 security policy enforced by the host platform. This is the composition problem which has

7246 been researched off and on without satisfactory results for at least 20 years. A side issue
7247 to be examined is what happens when the trusted application is implemented on a variety
7248 of host platforms and those platforms must communicate and interoperate

- 7249 • (U//FOUO) Support for complex security policies, such as dynamic access control
7250 policies like RAdAC
- 7251 • (U//FOUO) Construction of self-protecting applications that can guard themselves
7252 against attacks coming through the host platform, e.g., against attacks using disk storage
7253 or input devices.
- 7254 • (U//FOUO) Work is needed for all types of trusted platforms in the areas of system
7255 performance, user friendliness, and cost-effective security.

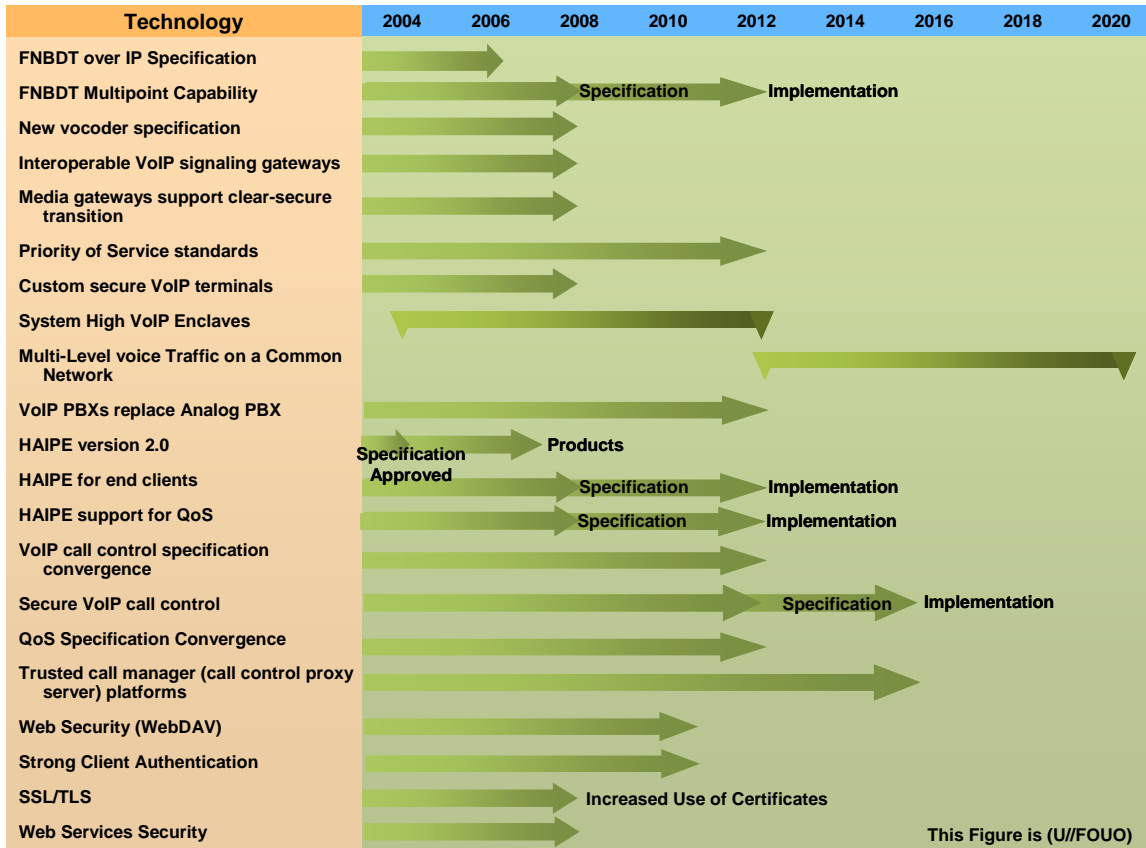
7256 (U//FOUO) In terms of strengthening the non-repudiation service, some technical steps can be
7257 taken. As noted above, potential technical vulnerabilities include compromise of private
7258 signature keys or shared secrets; inversion of hashing algorithms; and inability to securely store
7259 and/or retrieve records. These vulnerabilities can be narrowed through use of a stronger PKI,
7260 such as a higher-assurance DoD PKI. They can be narrowed through more controls on shared
7261 secrets; and more robust storage/retrieval systems. Time-stamping and notarization systems can
7262 be made more secure against attack (e.g., through the use of trusted operating systems and/or
7263 firewalls).

7264 (U//FOUO) Stronger proof of intent is a research area. As noted above, Alice can claim that she
7265 was not adequately presented with all the material she signed, or that the information she was
7266 presented on her screen did not match what was signed, or that her private key was used without
7267 her knowledge and consent (e.g., by a Trojan horse program operating on her computer).
7268 Defending against these claims will require much stronger computer systems. These systems
7269 must be secure against Trojan horses and other malware being present on the system. Software
7270 must be more reliable and secure to prevent modifications being made between presenting the
7271 material to Alice on her screen and it actually being signed within the system. Determining
7272 reliably that Alice was presented with the proper material, and did understand it, requires
7273 significant research breakthroughs in the area of computer-human interfaces.

7274 **2.3.5.1.3 (U) Infrastructure**

7275 (U//FOUO) The following gap areas have been identified in infrastructure associated with Secure
7276 Voice applications:

- 7277 • (U//FOUO) Secure VoIP-enabled gateways



7278

7279 **Figure 2.3-25: (U) Technology Timeline for Protection of User Information: Date in Transit**

7280 **2.3.5.2 (U) Cross Domain Solutions**

7281 (U) Recommendations for CDS technologies are as follows.

- 7282 • (U//FOUO) Develop trusted platforms that enable users who are not cleared to the
- 7283 highest level of data contained in the platform to use the platform to the level that they
- 7284 are cleared for.

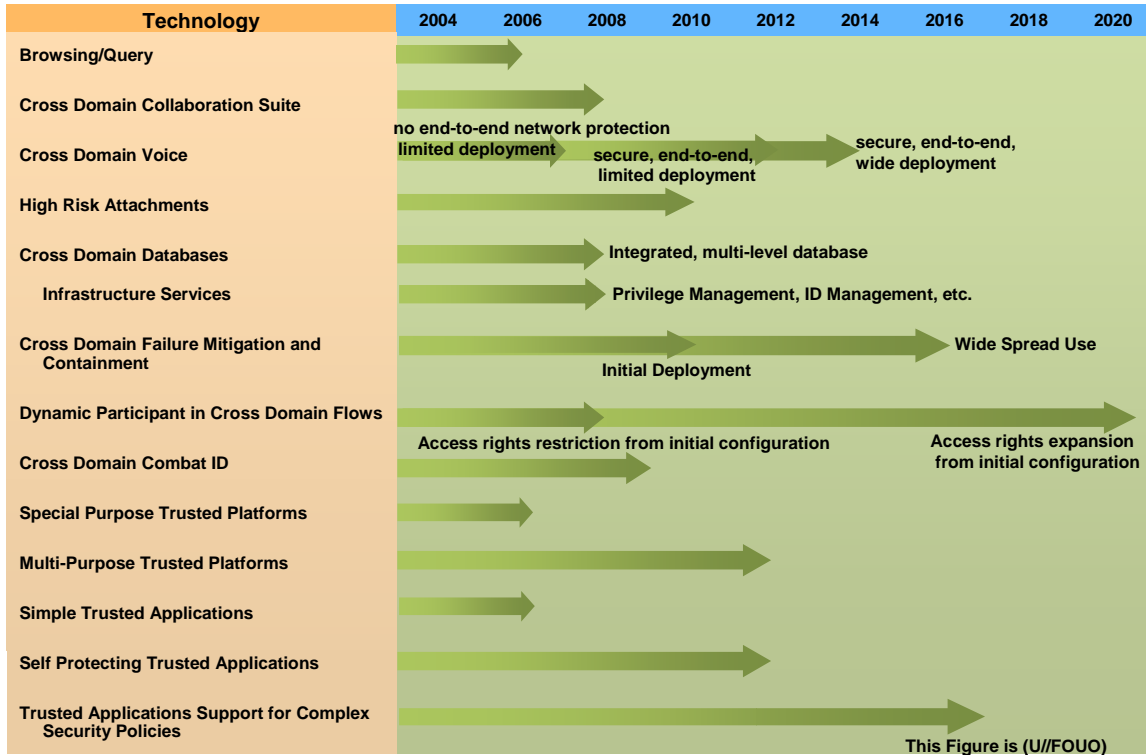
- 7285 • (U//FOUO) Develop trusted CDS workstations that allow warfighters to use applications
- 7286 they are accustomed to, e.g., for word processing, collaboration, situational awareness,
- 7287 and planning.

- 7288 • (U//FOUO) Develop trusted platforms allowing multiple domain access that can function
- 7289 under the resource constraints of the warfighters (e.g., space, weight, and power
- 7290 constraints of infantry) while supporting critical functionalities (e.g., combat ID, secure
- 7291 voice).

- 7292 • (U//FOUO) Enhance the functionality of data protection technologies to support
- 7293 information flows between security domains.

- 7294 • (U//FOUO) Immediately developed technologies to support cross-domain real-time
- 7295 flows, such as voice communications and collaboration, among coalition partners

- 7296 • (U//FOUO) Created standards for cross-domain technologies that focus on the reality of
7297 jointness of warfighter operations.
- 7298 • (U//FOUO) Develop common CDS capabilities, adequately deploy these Joint solutions,
7299 and sufficiently train warfighters in the use of these technologies in realistic
7300 environments.



7301
7302 **Figure 2.3-26: (U) Technology Timeline for Protection of User Information: Cross Domain**
7303 **Solutions**

7304

7305

7306 2.4 (U) DYNAMIC POLICY MANAGEMENT

7307 (U//FOUO) Dynamic Policy Management is the establishment of digital policies for enforcing
7308 how GIG assets are managed, utilized, and protected. This includes policies for access control,
7309 Quality of Protection (QoP), Quality of Service (QoS), transport, audit, computer network
7310 defense, and policies covering the hardware and software associated with GIG assets. GIG assets
7311 include all resources within the enterprise, including physical devices (e.g., routers, servers,
7312 workstations, security components), software (e.g., services, applications, processes), firmware,
7313 bandwidth, information, and connectivity. As this list of assets shows, Policy Management is
7314 more than just information access. It also includes performance management (both transport and
7315 network management and control), enforcement of QoP, QoS, resource allocation, connectivity,
7316 and prioritization within the transport, and enforcement of access to enterprise services which are
7317 all critical to GIG availability and end-to-end data-in-transit protections.

7318 (U//FOUO) Digital policy is the set of rules with which all actions of these assets must comply.
7319 To achieve enterprise wide (end-to-end) GIG policy management, the policies defining the rules
7320 for use of information, communications (transport), management and control functions, and
7321 service access must be integrated into a cohesive global policy. A full range of delegation of
7322 authority for policy creation and management, including intermediary policies (e.g.,
7323 departmental, domain) and local (e.g., mission, COI), will still reside below the global level.

7324 (U//FOUO) In addition, the GIG must be able to support the policies of non-GIG partners (e.g.,
7325 Intelligence Community, Industry, Department of Homeland Security, State Department, Allied
7326 nations, NATO) who have GIG access. This would include establishment of an agreed approach
7327 through perhaps an assured information sharing policy for risk measurement, risk management,
7328 and risk acceptance. The policy with non-GIG entities would specify things such as U.S. access
7329 and handling rights for allied-restricted data. Reciprocally, GIG policies must address the access
7330 to GIG assets by these partners.

7331 (U//FOUO) The dynamic aspect of policy management allows for the rapid adjustment of these
7332 rules in response to crisis situations that require either a reduction of privileges and accesses or
7333 increased latitude. These adjustments will change the criteria used to determine how resources
7334 are allocated to users and how access control decisions are made.

7335 (U//FOUO) The GIG must not only support adjustments to policy but also conditional policies.
7336 The policy for accessing a GIG asset will specify different access rules based upon the
7337 conditions that apply to this set of information. For example, the conditions for Warfighter
7338 information may be based upon DEFCON levels and the location of the user, while the
7339 conditions for business-to-business processes may be based upon the conditions of contracting
7340 (e.g., pre-request for proposal [RFP] coordination, RFP release, contract award). Under each
7341 condition a different set of access rules would apply. A policy accounts for the various
7342 conditions that affect access to that GIG asset. The set of conditions are not expected to be
7343 global; instead policies will specify behavior for the conditions that apply.

7344 (U//FOUO) Dynamic Policy Management allows the flexibility needed for the right data, at the
7345 right place, at the right time.

7346 **2.4.1 (U) GIG Benefits due to Dynamic Policy Management**

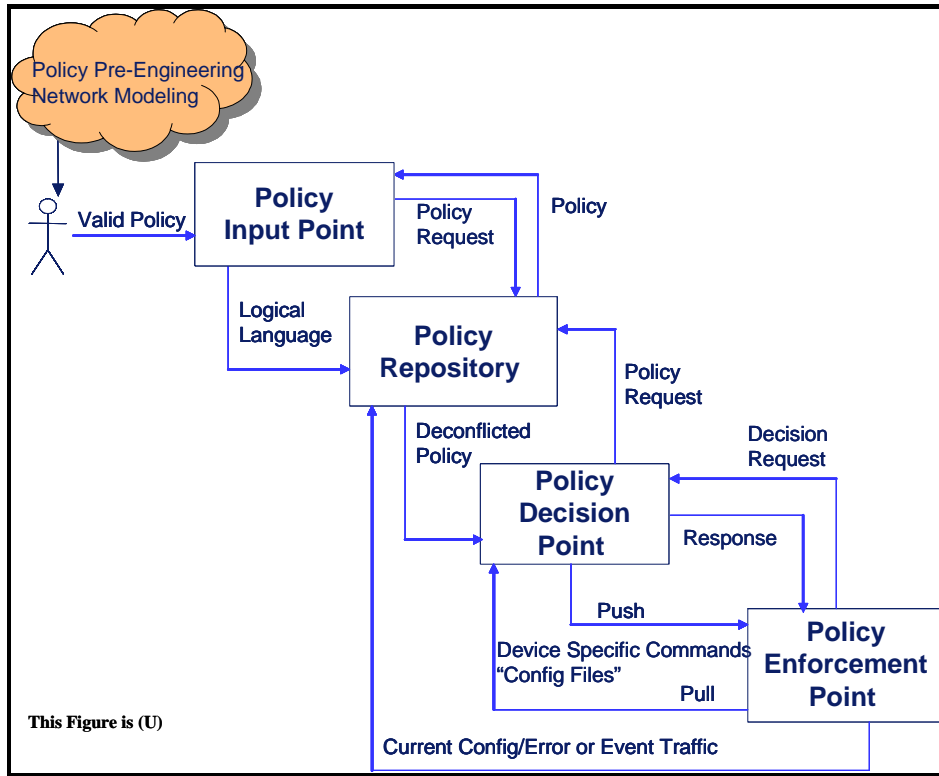
7347 (U//FOUO) The IA constructs used to support Dynamic Policy Management provide the
7348 following services to the GIG:

- 7349 • (U//FOUO) Create and manage the set of rules that govern all GIG actions
- 7350 • (U//FOUO) Provide synchronization among enterprise-wide and local policies
- 7351 • (U//FOUO) Translate and distribute (push or pull) the digital policy to devices enforcing
7352 policy
- 7353 • (U//FOUO) React to situational awareness conditions by changing the behavior of
7354 devices

7355 **2.4.2 (U) Dynamic Policy Management: Description**

7356 (U//FOUO) Dynamic Policy Management requires a framework to address policy management
7357 from the point of policy creation to policy installation in end devices. Included in this framework
7358 must also be the ability to dynamically update the policy in response to changing enterprise
7359 conditions. Figure 2.4-1 provides an architectural framework for discussing the functions and
7360 data flows required to perform dynamic policy management at the enterprise-level within the
7361 GIG.

7362 (U//FOUO) Dynamic Policy Management begins with a pre-engineering phase in which the
7363 enterprise security policy is validated before entering into the enterprise. Pre-engineering of the
7364 policy is critical to ensure that policy changes do not have an adverse effect on enterprise
7365 performance or security. Typically, predictive planning through network modeling and
7366 simulation tools is used to assess the impact of candidate policy changes on operational risk,
7367 network loads, and network/application interactions and to ensure security requirements for asset
7368 usage are not violated. Local, mission-specific policies will undergo similar pre-engineering
7369 activities. Prior to deployment, these candidate policy changes should be advertised to and
7370 negotiated with the appropriate approval body. The approval body will verify that no additional
7371 issues outside of those tested in this phase are applicable to the new policy.



7372

7373 **Figure 2.4-1: (U) Notional Architectural Framework for Dynamic Policy Management**

7374 (U//FOUO) Validated and approved global and local security policies enter the GIG enterprise
 7375 at a policy input point. The entity entering the policy must be identified and authenticated at the
 7376 input point. The input point must also determine if the entity has the proper authorizations
 7377 (privileges) to enter policy. Procedures will define how an entity is granted the right to
 7378 create/enter/modify policy. These privileges to enter/modify policy will be tightly controlled to
 7379 ensure that false policies cannot enter the GIG enterprise. The identity of the entity that
 7380 entered/modified the policy will be cryptographically bound to the policy so source and pedigree
 7381 authentication can be performed. The entered policies are coded in a logical language for transfer
 7382 to a policy repository. The policy is also sent to the policy repository in a human readable format
 7383 so that users can read the policy and better understand its impacts.

7384 (U//FOUO) The policy repository performs the main policy configuration management functions
 7385 in the GIG. All GIG policies are securely stored at the policy repository. It also performs policy
 7386 deconfliction to resolve any conflicts between the enterprise-wide policy, local, mission-
 7387 specific/COI policies, and the policies of non-GIG entities (e.g., coalition partners, allies, civil,
 7388 Homeland Security [HLS]) that have access to the GIG. There are specific functions performed
 7389 or responses provided at a given GIG asset that may be controlled by local users and their
 7390 mission-specific policies (i.e., COI policies). All other functions must be performed in
 7391 accordance with the rules dictated by enterprise-wide policy. This hierarchy of policies is
 7392 enforced at the policy repository.

7393 (U//FOUO) Policy deconfliction includes the identification of policy conflicts and a resolution
7394 capability that supports automated or human adjudication between multiple policies targeted for
7395 the same device suite. These deconfliction and synchronization steps are essential to avoid
7396 vulnerabilities that could be introduced by incompatible policies. The policy repository will
7397 generate a log of detected policy conflicts and the resolution outcome so that the policy input
7398 point operator can see in English how the new deconflicted policy differs from the original.

7399 (U//FOUO) The deconflicted policy is provided to a policy decision point (PDP). The policy
7400 decision point is a logical entity that has a centralized role in making policy decisions for itself or
7401 for other network elements that request such decisions. The PDP performs the following
7402 functions (which are further described in the following paragraphs):

- 7403 • (U//FOUO) Translates policy into device specific configuration commands
- 7404 • (U//FOUO) Distributes/Synchronizes policy configuration commands to affected policy
7405 enforcement points
- 7406 • (U//FOUO) Services policy requests from the policy enforcement points

7407 (U//FOUO) The PDP takes the policy rules stored in the policy repository and translates from the
7408 device-independent schema to device-specific configuration commands for the specific network
7409 devices to which the policy applies. These configuration commands program the network device
7410 to recognize the policy conditions, and when met, perform the policy action.

7411 (U//FOUO) The PDP also services policy requests from the policy enforcement points. If a
7412 policy enforcement point does not know what to do when presented with a particular situation or
7413 set of conditions, it will make a policy request to the PDP asking for guidance. The PDP can then
7414 either make a decision or send the request further up the policy chain for resolution.

7415 (U//FOUO) Policy distribution may take place as a result of the creation of a new policy or may
7416 be the result of a change in policy. The goal is to minimize changes in policies by defining
7417 different behaviors based upon different operational or mission conditions within a single policy.
7418 As the conditions change different behaviors are enforced. However, dynamic changes must still
7419 be supported for situations that require new behavior not anticipated in the original digital policy.

7420 (U//FOUO) Before the policy can be pushed to the end devices, the policy's base logic must be
7421 interpreted and transformed into the specific commands understood by each targeted recipient. It
7422 is envisioned that this process be automated for the GIG. These commands must have the right
7423 level of policy enforcement granularity for the targeted recipients policy enforcement function
7424 (i.e., the policy controlling user information access may require a more dynamic and finer
7425 grained policy than a policy controlling connectivity within the Black Core).

7426 (U//FOUO) This translation function supports the use of commercially available products such as
7427 Policy Enforcement Points (PEP). Usually, these commands take the form of configuration files.
7428 After the files are created and validated, the policy is distributed using a push or pull model. The
7429 push model would be used for policy changes that must take effect immediately because new
7430 behavior is needed under a particular condition. The pull model can be used in cases in which a
7431 policy change is scheduled to take effect at a particular time but is not critical to current
7432 operations. The targeted device pulls the updated policy from the policy decision point. Ensuring
7433 the devices receive or retrieve the updated policies in a synchronized manner is a critical aspect
7434 of policy distribution.

7435 (U//FOUO) A PEP is a GIG asset with the responsibility of conforming to and enforcing the GIG
7436 rules (e.g., which entities can access which resources, what functions can entities perform). PEPs
7437 will be able to react and implement one or more policy rules, based on an input trigger that
7438 denotes a change in condition. These conditions will signify operational conditions or mission
7439 environment changes. Because the digital policies encode different behavior under different
7440 conditions, the PEP will implement the new rules without requiring redistribution of policy
7441 configuration information from a central source. The trigger could be automated or manual (such
7442 as an operator command). If the policy rules to implement are ambiguous (e.g., multiple
7443 conditions exist concurrently), intervention may be necessary to resolve the ambiguity.

7444 (U//FOUO) The actual enforcement function is addressed in the Policy-Based Access Control IA
7445 System Enabler (Section 2.2). The policy management functions performed at the PEP includes
7446 policy receipt (by push or pull), policy storage, and policy error or event handling. When errors
7447 or events are detected, the PEP identifies these conditions to the policy repository for resolution.
7448 Examples of errors or events are: receipt of a configuration file that a device does not know how
7449 to use, receipt of a corrupted configuration file, or inability to pull a policy from a decision point
7450 at the specified time or under the specified condition.

7451 (U//FOUO) Throughout the dynamic policy management architectural framework is the need for
7452 security services and mechanisms to protect the policy throughout its life cycle. From the point
7453 of creation to installation in the policy enforcement point, every GIG entity handling digital
7454 security policies must maintain the integrity of policy information for policy-at-rest and policy-
7455 in-transit throughout the management infrastructure. In addition, GIG assets must maintain
7456 integrity of the source of origin for policy throughout the management infrastructure.
7457 Confidentiality protection must be provided if the policy resident at the GIG asset requires it.

7458 (U//FOUO) Security Services must be applied to actions within Dynamic Policy Management.
7459 Every entity sending or receiving policy information must be identified and authenticated. In
7460 addition, their privileges to send, receive, and modify policy as well as to send error or event
7461 messages to the policy repository must be validated. The integrity of the policies being
7462 promulgated must also be validated each time they are distributed and used. Other pervasive
7463 security services include the logging of all policy management transactions and the assured
7464 availability of the management infrastructure. As a critical aspect to maintain the security posture
7465 of the GIG, the availability of policy input, repository, decision, and enforcement points is vital
7466 to nearly all GIG functions.

7467 (U//FOUO) In summary, the policy life cycle includes:

- 7468 • (U//FOUO) A pre-engineering phase in which the security policy is validated before
7469 being used
- 7470 • (U//FOUO) A policy creation phase, where policies enter the GIG enterprise
- 7471 • (U//FOUO) Policy deconfliction to resolve the conflicts between all the policies
- 7472 • (U//FOUO) Policy distribution, targeting which GIG assets should receive the digital
7473 policy and translating the base logic of the policy into device specific commands
- 7474 • (U//FOUO) An installation phase in which policy is installed or replaces existing policy
7475 in end devices
- 7476 • (U//FOUO) Security services and mechanisms are used to authenticate and protect the
7477 integrity, availability, and confidentiality of the policy throughout its life cycle

7478 **2.4.3 (U) Dynamic Policy Management: Technologies**

7479 (U//FOUO) The following technology areas support the Dynamic Policy Management Enabler:

- 7480 • (U) Development of Policies
 - 7481 • (U) Centralized vs. Distributed
 - 7482 • (U) Elements of the policies
 - 7483 • (U) Access Control
 - 7484 • (U//FOUO) Trust Anchors
 - 7485 • (U//FOUO) Policy Languages
- 7486 • (U) Distribution of Policies
 - 7487 • (U) Standard Protocols
 - 7488 • (U) Security Issues
- 7489 • (U) Policy Architectures
- 7490 • (U) Policy Directories

7491 **2.4.3.1 (U//FOUO) Development of Policies**

7492 (U) The development of policy includes the following three sub-sections:

- 7493 • (U) Centralized vs. distributed
- 7494 • (U) Elements of the policies
- 7495 • (U) Policy languages

7496 **2.4.3.1.1 (U) Centralized vs. Distributed**

7497 **2.4.3.1.1.1 (U) Technical Detail**

7498 (U) Centralized Policy Control: Several commercial products perform centralized policy
7499 management. This technology provides a centralized control of network configuration, including
7500 policy creation, maintenance, and protection. A server is used to define and store the network
7501 policies and then distribute the policies out to the remote policy enforcement points with little or
7502 no user intervention.

7503 (U) Distributed Policy Control: Distributed policy control focuses on large dynamic networks
7504 with no central administrative control. These are independent Internet domains with dynamic
7505 topology and state information. In a multi-domain network, a number of individuals or service
7506 providers interact in a collaborative environment to provide certain services, organized according
7507 to a set of rules and policies that define how their resources can be shared among them. A
7508 distributed policy system has no centrally controlled enforcement of the policies. Consequently,
7509 there is no guarantee that policies will be followed as they are prescribed: members of a network
7510 may fail to—or choose not to—comply with the rules. If there is no way of practical (physical)
7511 enforcement of policies, then it would be useful to have a normative control mechanism for their
7512 soft enforcement (sanctions or penalties).

7513 **2.4.3.1.1.2 (U) Usage Considerations**

7514 **2.4.3.1.1.2.1 (U) Implementation Issues**

7515 (U//FOUO) Current centralized policy management products are mostly product specific. For
7516 example, the Network-1 Security Solutions CyberwallPLUS product is used to configure
7517 firewalls. The McAfee® ePolicy Orchestrator® (ePO™) product is used to define policies for
7518 virus activity, desktop firewall policy, and spam and content-filtering policies. The Pedestal
7519 Software's SecurityExpressions product is used to configure Microsoft application policies.

7520 (U//FOUO) For decentralized policy management, there are implementation issues with the
7521 synchronization of a common GIG policy amongst independent network administration systems.
7522 How do you enforce that all distributed network systems are working from the current GIG
7523 policy?

7524 **2.4.3.1.1.2.2 (U) Advantages**

7525 (U//FOUO) Centralized policy controlled systems can be configured so that local users cannot
7526 change the policy configurations at the end network devices. They can also verify current policy
7527 usage through compliance reports. This insures that the network is using the correct policy. This
7528 synchronization of policy is very important to GIG stability and overall security.

7529 **2.4.3.1.1.2.3 (U) Risks/Threats/Attacks**

7530 (U//FOUO) Centralized policy management requires strong identification, authentication, and
7531 confidentiality protection at the policy server. An attack at the centralized policy server could
7532 effect all policy enforcement points in the system.

7533 **2.4.3.1.1.3 (U) Maturity**

7534 (U) Examples of centralized policy control products includes the following:

- 7535 • (U) Network-1 Security Solutions CyberwallPLUS firewall software
- 7536 • (U) McAfee® ePolicy Orchestrator® (ePO™)
- 7537 • (U) Pedestal Software's SecurityExpressions

7538 (U//FOUO) The various sub-technologies of the centralized vs. distributed policy control
 7539 technology area can be generally assigned Technology Readiness Level groups of Early,
 7540 Emerging, or Mature.

- 7541 • (U//FOUO) Centralized Policy Management—Emerging (TRLs 4- 6)
- 7542 • (U//FOUO) Distributed Policy Management—Early (possibly low Emerging) (TRLs 2 –
 7543 4)

7544 **2.4.3.1.1.4 (U) Cost/Limitations**

7545 (U) When comparing centralized vs. distributed policy management, the centralized approach
 7546 has less overhead cost. Performing the policy creation, verification, and distribution at a
 7547 centralized site requires less personnel than a distributed approach where there could be multiple
 7548 groups of people performing similar tasks.

7549 **2.4.3.1.1.5 (U) References**7550 (U) <http://www.esecurityplanet.com/prodser/article.php/1431251>7551 (U) http://www.networkassociates.com/us/products/mcafee/mgmt_solutions/epo.htm7552 (U) <http://infosecuritymag.techtarget.com/2002/feb/testcenter.shtml>7553 (U) <http://trantor.imit.kth.se/vinnova/DPBM.html>7554 (U) http://www.cs.wisc.edu/condor/doc/ncoleman_tr1481.pdf7555 **2.4.3.1.2 (U) Elements of the Policies**

7556 (U) Two technologies are discussed in this section: access control and trust anchors.

7557 **2.4.3.1.2.1 (U) Access Control**7558 **2.4.3.1.2.1.1 (U) Technical Detail**

7559 (U//FOUO) Access control policies consist of a set of rules imposed on all users and devices in
 7560 the network. These rules generally rely on a comparison of the sensitivity of a resource and the
 7561 possession of corresponding attributes for users or devices attempting to access the resource.
 7562 These rule-based policies can be used by the GIG to enforce access control and other policies.

7563 (U//FOUO) Some Public Key Infrastructure (PKI) programs such as Defense Message System
7564 (DMS) use rule-based policies, mostly for access control. The GIG includes policies for access
7565 control, quality of protection, quality of service, transport, audit, computer network defense, and
7566 policies covering the hardware and software associated with GIG assets. As these policies grow
7567 in complexity, so do the number of rules and the deconfliction of these rules.

7568 (U//FOUO) These rule-based policies are first entered at the Policy Input Point (PIP) in an easily
7569 recognizable, human readable format. The PIP serves as a console for an authorized user to
7570 create new policies and edit existing policies. After the policy is created or updated, the PIP
7571 performs a translation to a base logic format that is sent to the Policy Repository. See section
7572 2.4.2 for more details on PIP.

7573 2.4.3.1.2.1.2 (U) Usage Considerations

7574 2.4.3.1.2.1.2.1 (U) Implementation Issues

7575 (U//FOUO) The GIG will have many rule-based policies. There will be enterprise-wide policies,
7576 local, mission-specific/COI policies, and the policies of non-GIG entities (e.g., coalition partners,
7577 allies, civil, HLS). Deconfliction of all these policies must take place before a policy is posted
7578 for distribution. And these rule-based policies will need to be deconflicted each time a new or
7579 update policy is introduced.

7580 (U//FOUO) There are few deconfliction tools available today to perform this task. The KAoS
7581 policy service and Rei product have some policy confliction resolution capabilities, but these
7582 tools will need to be further developed for the GIG program. Initial versions of deconfliction
7583 tools may require operator intervention to settle conflicts between policies. As the deconfliction
7584 tools mature, this process will become more automated.

7585 2.4.3.1.2.1.2.2 (U) Advantages

7586 (U//FOUO) Rule-based policies can be easily expanded to define additional policy by adding
7587 new rules. This does cause more complicated attributes but with mapping each attribute category
7588 to a bit value, a very detailed user attribute can be stored in a small package.

7589 2.4.3.1.2.1.2.3 (U) Risks/Threats/Attacks

7590 (U//FOUO) One risk to rule-based policies is in keeping a new policy synchronized amongst the
7591 users and devices. When new policies are created with additional rules, the existing user/device
7592 attributes may not cover these rules properly and they will need to be updated. Automated
7593 distribution of policy and electronic updates of users and device attributes are required to keep
7594 rule-based policy information synchronized and working properly.

7595 2.4.3.1.2.1.3 (U) Maturity

7596 (U//FOUO) Basic rule-based policies are very mature in the PKI world. The DMS has been using
7597 the Security Policy Information File (SPIF) with v3 X.509 certificates for five years.

7598 (U//FOUO) DMS uses the SPIF as a configurable access control mechanism. SPIFs contain the
 7599 information needed to create and interpret security labels. Each v3 signature certificate
 7600 references the SPIF, defining the security policy under which the certificate is issued. The SPIF
 7601 is used to interpret Partition Rule Based Access Control (PRBAC) parameters contained in the
 7602 X.509 certificate and the object security label. The SPIF is directly linked to a security policy.
 7603 When a security policy is changed (i.e., the classifications or security categories are redefined),
 7604 the SPIF associated with that policy must also be changed.

7605 (U//FOUO) SPIFs are generated and signed by a root authority (i.e., trust anchor) and pushed to
 7606 sub-authorities by a physical distribution path. The sub-authorities re-sign the SPIF and post the
 7607 signed SPIF to the directory. The SPIF is also distributed to lower level authorities within the
 7608 sub-authority’s domain. Local policy dictates whether end users receive SPIFs through
 7609 distribution or retrieve them from the directory.

7610 (U//FOUO) There is enough flexibility in the SPIF to create a fairly complex implementation.
 7611 Since there are no syntactic constraints on the uniqueness of displayable strings for security
 7612 classifications and security categories, it is possible for independent classifications or categories
 7613 to be assigned the same representation. To limit this complexity, SPIF implementers shall ensure
 7614 that all human readable (displayable) or external representations of security classifications and
 7615 security categories are unique within a SPIF implementation.

7616 (U//FOUO) When two security policy domains cross-certify, there is the possibility that two or
 7617 more external policy sensitivities might be mapped to a single local policy sensitivity. This
 7618 many-to-one sensitivity mapping must be carefully managed to prevent unwanted changes in
 7619 sensitivities when sending data across policy domain boundaries. For example, a security policy
 7620 in Domain 2 may be implemented so that both Sensitivity A and Sensitivity B originating in
 7621 Domain 1 will be mapped to Sensitivity X in Domain 2. The possibility of sensitivities changing
 7622 when mapped between policy domains must be carefully considered when the two Security
 7623 Policy Authorities develop equivalencies between their respective security policies.

7624 (U//FOUO) The various sub-technologies of the access control technology area can be generally
 7625 assigned Technology Readiness Level groups of Early, Emerging, and Mature.

- 7626 • (U//FOUO) Rule based access control—Mature (TRLs 6 – 9)
- 7627 • (U//FOUO) Adaptive access control—Early (TRLs 1 – 3)
- 7628 • (U//FOUO) Deconfliction of policy—Early (TRLs 1 – 3).

7629 2.4.3.1.2.1.4 (U) Standards

7630 **Table 2.4-1: (U) Access Control Standards**

This Table is (U)	
Standard	Description
SDN.801	SDN.801 addresses concepts, tools and mechanisms for implementation of access control (AC). SDN.801 should be used to gain both a global understanding of MISSI access control, and as a guide for implementing access control features in MISSI-compliant components. SDN.801 is designed to advance from general concepts that introduce access control to more

This Table is (U)	
Standard	Description
	detailed information on access control tools, mechanisms, and processes as they apply to real-world communication systems.
ANSI INCITS 359-2004	This standard describes Role Based Access Control (RBAC) features that have achieved acceptance in the commercial marketplace. It includes a reference model and functional specifications for the RBAC features defined in the reference model. RBAC has become the predominant model for advanced access control because it reduces the complexity and cost of security administration in large networked applications. Many information technology vendors have incorporated RBAC into their product line, and the technology is finding applications in areas ranging from health care to defense, in addition to the mainstream commerce systems for which it was designed. The National Institute of Standards and Technology (NIST) initiated the development of the standard via the INCITS fast track process.
XACML 1.0	XACML is an XML-based language, or schema, designed specifically for creating policies and automating their use to control access to disparate devices and applications on a network.
This Table is (U)	

7631 2.4.3.1.2.1.5 (U) Cost/Limitations

7632 (U) GIG dynamic policy management performs policy deconfliction to resolve the conflicts
7633 between the enterprise-wide policy, local, mission-specific/COI policies, and the policies of non-
7634 GIG entities. There are limitations on how well current access control methods can support this
7635 deconfliction process.

7636 2.4.3.1.2.1.6 (U) Alternatives

7637 (U) XACML is an OASIS standard (Organization for the Advancement of Structured
7638 Information Standards) that describes both a policy language and an access control decision
7639 request/response language (both written in XML). The policy language is used to describe
7640 general access control requirements, and has standard extension points for defining new
7641 functions, data types, combining logic, etc. The request/response language lets you form a query
7642 to ask whether or not a given action should be allowed, and then interpret the result. This
7643 resulting response always includes an answer about whether the request should be allowed, using
7644 one of four values: Permit, Deny, Indeterminate (an error occurred or some required value was
7645 missing, so a decision cannot be made) or Not Applicable (the request can't be answered by this
7646 service).

7647 2.4.3.1.2.1.7 (U) References

7648 (U) FORTEZZA[®] Security Management Infrastructure (SMI) Concept of Operation CONOP) for
7649 CipherNET[®] 3000 CAW 5.0

7650 (U) SDN.801: ACCESS CONTROL CONCEPT AND MECHANISMS

7651 (U) http://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html

7652 **2.4.3.1.2.2 (U//FOUO) Trust Anchors**

7653 2.4.3.1.2.2.1 (U) Technical Detail

7654 (U//FOUO) The purpose of a trust anchor is to serve as a baseline for the validation of some
7655 entity/action. The trust anchor is something that has been accepted through out-of-band means as
7656 being valid and reliable. For example, it can be a public key or certificate corresponding to a
7657 private key. Without this baseline, there is no sound way of validating anything else.

7658 (U//FOUO) In some systems, the trusted anchor is called a trusted root or root authority. The
7659 trusted root or root authority is the point at which trust begins in a PKI system. The root
7660 authority is the certification authority that certifies the existence and quality of other certification
7661 authorities in the particular PKI that you wish to use. The business and Internet communities are
7662 not waiting for some over-arching system to be put into place by governments or agencies. They
7663 are seizing opportunities as they arise—putting in place systems that they trust and selecting their
7664 own root authorities.

7665 (U//FOUO) The initial loading of a trust anchor in the system **MUST** be by a trusted out of band
7666 means. If you receive a trust anchor over the network—how do you know it's good? You have
7667 no trust anchor to use to validate the new one, so you either take a chance that you're being
7668 spoofed and accept it (and open yourself up to lots of attacks), or you refuse to accept it because
7669 you can't validate it. That is why it is so important that the initial loading of a trust anchor comes
7670 from a highly trusted source.

7671 (U//FOUO) With respect to dynamic policy management, how does the policy input point know
7672 to trust the person requesting to create or edit GIG policies? How does the policy enforcement
7673 point verify the policy configuration file received from the policy decision point? The answer to
7674 these questions starts with the trust anchor.

7675 (U//FOUO) Once the initial loading of a trust anchor has been accomplished, it can be updated or
7676 transferred securely over the net. See RFC 3157 for details of the Securely Available Credentials
7677 (SACRED) protocol, which can be used to securely transfer credentials.

7678 (U//FOUO) The trust anchor and the personnel managing the trust anchor are the heart of the
7679 trust in PKI and other authentication-based systems. The consequences of compromise to a trust
7680 anchor by malicious intent, inadvertent errors, or system failures can be severe. Hence, this trust
7681 anchor must be diligently protected. Such protection can be provided by placing all
7682 cryptographic key management and encryption/decryption functions into a trusted/tamper-proof
7683 hardware device rather than residing in software on a host computer.

7684 (U//FOUO) Trust anchors operate under a set of rules or policies that describe both the physical
7685 and electronic protection of the trust anchor information. Failing to follow these rules and
7686 policies could cause the revocation or compromise of the trust anchor, affecting all authorities,
7687 users, and devices whose authentication path is based on that trust anchor.

7688 2.4.3.1.2.2.2 (U) Usage Considerations

7689 2.4.3.1.2.2.2.1 (U) Implementation Issues

7690 (U//FOUO) The main implementation issue with trust anchors is the initial delivery of the trust
 7691 anchor information. If you receive a trust anchor over the network—how do you know it’s good?
 7692 It is very important that the initial loading of a trust anchor comes from a highly trusted source.

7693 2.4.3.1.2.2.2.2 (U) Advantages

7694 (U//FOUO) Trust anchors provide a fairly simple and straightforward method of verifying
 7695 authentication paths for users, devices, and organizations. With the help of compromise lists and
 7696 revocation lists, the trust anchor provides the information needed to determine if a message or
 7697 data is from a valid source.

7698 2.4.3.1.2.2.2.3 (U) Risks/Threats/Attacks

7699 (U//FOUO) Trust anchors must be protected from both physical and electronic attacks due to the
 7700 implications of a revocation or compromise. Trust anchors should be stored in well-protected
 7701 locked areas. Multi-person access to the physical location will reduce the risk of attacks. Multi-
 7702 person access to the workstation or system containing the trust anchor would further reduce
 7703 attacks. Personnel operating the trust anchors should be highly trusted individuals.

7704 2.4.3.1.2.2.3 (U) Maturity

7705 (U//FOUO) PKI systems have been using trust anchors for over ten years. The trust anchor in a
 7706 PKI system is usually called the root authority. Some PKI systems also support cross certificates,
 7707 which allow certificate path validation between users under different trust anchors.

7708 (U//FOUO) The various sub-technologies of the trust anchor technology area can be generally
 7709 assigned Technology Readiness Level groups of Early, Emerging, and Mature.

- 7710 • (U//FOUO) PKI root authority—Mature (TRLs 6 – 9)
- 7711 • (U//FOUO) Cross registration between trust anchors—Emerging (TRLs 4 – 6)
- 7712 • (U//FOUO) Trust anchor initial load and updates—Emerging (TRLs 4 – 6).

7713 2.4.3.1.2.2.4 (U) Standards

7714 **Table 2.4-2: (U) Trust Anchor Standards**

This Table is (U)	
Standard	Description
RFC 3157	This document identifies a set of requirements for credential mobility. Using SACRED protocols, users will be able to securely move their credentials between different locations, different Internet devices, and different storage media as needed.
This Table is (U)	

7715 2.4.3.1.2.2.5 (U) References

7716 (U) FORTEZZA[®] Security Management Infrastructure (SMI) Concept of Operation CONOP) for
7717 CipherNET[®] 3000 CAW 5.0

7718 **2.4.3.1.3 (U) Policy Languages**

7719 **2.4.3.1.3.1 (U) Technical Detail**

7720 (U//FOUO) Policy Languages are used to define policy statements that can be used by
7721 networking hardware such as routers, firewalls, and guards. These policy statements can be used
7722 for routing, access control, and QoS purposes.

7723 (U//FOUO) Several policy languages exist which may be appropriate for application in the GIG:
7724 Routing Policy Specification Language, Path-based Policy Language, Security Policy
7725 Specification Language, KeyNote, and Extensible Access Control Markup Language (XACML).
7726 But most of these languages were designed for one thing, such as generate routing tables, QoS
7727 using differentiated service code points, access control using access control lists (ACLs), etc.

7728 (U//FOUO) GIG requires dynamic policy management that handles all the required GIG policies,
7729 including: access control, RAdAC, QoP, QoS, transport, audit, computer network defense, and
7730 policies covering the hardware and software associated with GIG assets. To do that, either
7731 multiple policy languages will be needed to create the overall GIG policy or a more robust policy
7732 language needs to be developed that will support all the GIG policies. Some existing policy
7733 languages such as Ponder, KAoS, Rei, and XACML are flexible in that they allow you to define
7734 new policy within the language. GIG should further research these flexible policy languages to
7735 see which would be best suited for the GIG policies.

7736 **2.4.3.1.3.2 (U) Usage Considerations**

7737 (U//FOUO) RAdAC will need specific capabilities in its access control policy but should fold
7738 into the larger GIG dynamic policy effort. Some potential technologies that could support access
7739 control policy include WS-Policy, Standard Deontic Logic such as that implemented in Rei or
7740 Ponder, and artificial intelligence constructs in PROLOG, decision trees, or fuzzy logic. This
7741 section assumes that the distributed functionality (e.g., secure update, revocation, currency
7742 validation, and caching for off-line use) is provided by the dynamic policy enabler and thus
7743 focuses only on RAdAC-specific digital policy needs.

7744 (U//FOUO) Dynamic Access Control Policy serves as an input to the RAdAC model in order to
7745 control its behavior. In this usage, the policy must be expressive enough to dictate some or all of
7746 the following access control characteristics:

- 7747 • (U//FOUO) Minimum number of required inputs to calculate risk and operational need
- 7748 • (U//FOUO) Relative weighting of the various inputs for risk and operational need
- 7749 • (U//FOUO) Relative weighting of risk versus operational need for the final decision
- 7750 • (U//FOUO) Ability to understand (in human readable terms) the limiting factors
7751 (LIMFAC) that contributed to a failed access attempt

- 7752 • (U//FOUO) Ability to express stateful access control rules (e.g., successive failed access
7753 attempts)
- 7754 • (U//FOUO) Ability to express policy according to enterprise and COI roles
- 7755 • (U//FOUO) Ability to negotiate two or more conflicting access control rules
- 7756 • (U//FOUO) Ability to negotiate access control policy with neighboring security domains
7757 in order to define an access control boundary interface that is agreeable to both sides
- 7758 • (U//FOUO) Ability to express and automatically select between multiple policies based
7759 on nationality or security domain
- 7760 • (U//FOUO) Ability to express more granular or more restrictive access control policies at
7761 each successive echelon down the chain of command
- 7762 • (U//FOUO) Ability to dynamically tighten or loosen access control policy based on
7763 situation (INFOCON, proximity to enemy forces, etc.).
- 7764 • (U//FOUO) Ability to do all of this very quickly so as not to become the system
7765 bottleneck

7766 (U//FOUO) In this first role influencing RAdAC behavior, the policy must somehow be able to
7767 handle policy exceptions (termed “dispensations” in some deontic languages) that are able to
7768 authorize otherwise disallowed actions—but only for a limited time period and only for a well-
7769 defined set of actions.

7770 (U//FOUO) Due to national law or immutable operational policy, care has to be taken to
7771 constrain where dispensations themselves are allowed and not allowed within the policy
7772 language. For example, dispensations may be allowed for dissemination of a classified document
7773 to a cleared User without formal access approval, given compelling operational need but may
7774 never be allowed for an uncleared User. Dispensations may be the most appropriate way for
7775 digital policy to annotate and reason about a commander’s or supervisor’s consent for a User’s
7776 operational need to know a particular piece of information.

7777 (U//FOUO) Dynamic Access Control Policy also requires expressiveness for RAdAC output. For
7778 instance, the policy engine may recognize a specific request as having a compelling operational
7779 need but having too risky an IT Component to release the information to. In this case, policy
7780 should be expressive enough to conclude that an alternate path (alternate Course Of Action, or
7781 COA) for this LIMFAC should be examined before arriving at a final access decision. In this
7782 role, policy must be expressive enough to dictate the following alternate COA determinations:

- 7783 • (U//FOUO) Alternate enterprise routing evaluation to obtain higher QoP from end to end
- 7784 • (U//FOUO) Digital rights restrictions to limit the risk of disclosure or further
7785 dissemination
- 7786 • (U//FOUO) Automatic sanitization through a guard (or originator) process prior to
7787 release

- 7788 • (U//FOUO) Evaluation of nearby neighbors or superiors who might have more robust IT
7789 Components for handling the data as-is

7790 (U//FOUO) In this second role influencing RAdAC output, the policy must be tightly integrated
7791 with the policies that affect management of the IT Components. This avoids situations where
7792 RAdAC allows access through a given enterprise route but then the enterprise routes the
7793 information over a different path because of other decision metrics. Digital rights policy
7794 enforcement must be tightly integrated with the end user equipment portion of IT Components so
7795 that the rights embedded with the information object are strictly enforced.

7796 (U//FOUO) Finally, the policy must be robust enough to meet extremely stringent false negative
7797 and false positive rates. Since RAdAC would be replacing the traditional Mandatory Access
7798 Control model objectively, false positives in particular cannot be tolerated for risk of information
7799 disclosure. Dispensations for exception handling must be constrained in such a way that
7800 guarantees select portions of digital access control policy will comply with national law.

7801 2.4.3.1.3.2.1 (U) Implementation Issues

7802 (U//FOUO) Current policy management products are mostly vendor specific. There are many
7803 forms of policy languages for covering routing, QoS, or access control.

- 7804 • (U) Routing Policy Specification Language (RPSL) was developed by the IETF Routing
7805 Policy System Working Group (RFC 2622 and RFC 2725). RPSL allows a network
7806 operator to specify routing policies at various levels in the Internet hierarchy; for example
7807 at the autonomous system level. At the same time, policies can be specified with
7808 sufficient detail in RPSL so that low-level router configurations can be generated from
7809 them. RPSL is extensible and new routing protocols and new protocol features can be
7810 introduced at any time.

- 7811 • (U) Tier 1 ISP in Australia designed and built Connect's RPSL-based system to
7812 manage routing policy and configure routers. Problem: Policy can easily get very
7813 complex and result in very complex router configuration.

- 7814 • (U) Ponder Policy Specification Language: Ponder is a declarative, object-oriented
7815 language for specifying management and security policies for distributed systems. It is a
7816 role-based access control. Ponder is a product of the Imperial College of Science,
7817 Technology, and Medicine in London, England. It has been developed as part of ongoing
7818 research being carried out by the group into the use of policy in distributed systems
7819 management. Ponder is a general-purpose management language for specifying what
7820 actions are performed and how to allocate resources when specific events occur.

- 7821 • (U) The Ponder toolkit includes the following:

- 7822 • (U) Ponder Compiler: A Compiler framework for the Ponder policy specification
7823 language. It supports the main features of the Ponder grammar. It consists of a
7824 Syntax Analyzer, a two-pass Semantic Analyzer, and the default Java Code
7825 Generator for Obligation and Refrain Policies, and XML code generator.

- 7826
- 7827
- 7828
- 7829
- 7830
- (U) Ponder Policy Editor: A customizable text editor for the Ponder language, written in Java. It has all the basic features of a text editor and includes features that make text editing Ponder Policies easy.
 - (U) Ponder Management Toolkit: A Management Toolkit Framework, designed to allow for the addition of tools to be managed from a central management console.
- 7831
- 7832
- (U) Ponder also has built-in tools for performing both runtime checking of policy rules and offline checking of policy rules.
- 7833
- (U) The Security Assertion Markup Language (SAML) is a planned standard for interoperability among Web services security products. SAML is developed and maintained by the Organization for the Advancement of Structures Information Standards (OASIS) organization's XML-Based Security Services Technical Committee (SSTC). SAML defines a common XML framework for exchanging security assertions between entities for the purpose of exchanging authentication and authorization information.
- 7834
- 7835
- 7836
- 7837
- 7838
- (U) Extensible Access Control markup Language (XACML). XACML is an OASIS standard that describes both a policy language and an access control decision request/response language (both encoded in XML). The policy language is used to describe general access control requirements. It has standard extension points for defining new functions, data types, combining logic, etc. The request/response language lets you form a query to ask whether or not a given action should be allowed and then interpret the result.
- 7839
- 7840
- 7841
- 7842
- 7843
- 7844
- 7845
- (U) Parthenon Software has produced a suite of Policy products based on XACML. It identifies an XML-based language that is used to describe access control requirements for online resources. The intent is to allow for efficient machine parsing of arbitrarily complex security policies.
- 7846
- 7847
- 7848
- 7849
- (U) Sun's XACML was developed in Sun Microsystems Laboratories, part of Sun Microsystems, Inc., as an open source implementation of the OASIS XACML standard, and was written in the Java™ programming language. This product provides complete support for all the mandatory features of XACML as well as a number of optional features. Specifically, there is full support for parsing both policy and request/response documents, determining applicability of policies, and evaluating requests against policies. All of the standard attribute types, functions, and combining algorithms are supported, and there are interfaces for adding new functionality as needed. Sun is looking at adding features to connect XACML and things like SAML or Lightweight Directory Access Protocol (LDAP), and strong tools support.
- 7850
- 7851
- 7852
- 7853
- 7854
- 7855
- 7856
- 7857
- 7858
- 7859
- (U) Lagash Systems XACML.NET is an implementation of the XACML specification released by OASIS in purely .Net code (C#) that can be used by anyone in the .Net developer community. XACML.NET is under the Mozilla public license (MPL) 1.1 so any software under a license compatible with MPL can use this code.
- 7860
- 7861
- 7862
- 7863
- (U) KeyNote is a flexible trust-management system designed to work well for a variety of large- and small-scale Internet-based applications. KeyNote was designed and developed in 1997 by representatives from AT&T Labs, Yale University, and the University of
- 7864
- 7865
- 7866

7867 Pennsylvania. It provides a single, unified language for both local policies and
 7868 credentials. KeyNote policies and credentials, called assertions, contain predicates that
 7869 describe the trusted actions permitted by the holders of specific public keys. KeyNote
 7870 assertions are essentially small, highly structured programs. A signed assertion, which
 7871 can be sent over an untrusted network, is also called a credential assertion. Credential
 7872 assertions, which also serve the role of certificates, have the same syntax as policy
 7873 assertions but are also signed by the principal delegating the trust.

- 7874 • (U) KeyNote is described in RFC-2704. It has no restrictions on its use and
 7875 distribution. The KeyNote Toolkit is a C language open-source reference
 7876 implementation and can be obtained at <http://www.crypto.com/trustmgt/kn.html>

- 7877 • (U) Rei was developed by the eBiquity Group, a research organization that consists of
 7878 faculty and students from the Department of Computer Science and Electrical
 7879 Engineering (CSEE) of UMBC. Rei is a policy language based on OWL-Lite (Web
 7880 Ontology Language with a restricted vocabulary) that allows policies to be specified as
 7881 constraints over allowable and obligated actions on resources in the environment. Rei
 7882 also includes logic-like variables, which give it the flexibility to specify relations like role
 7883 value maps that are not directly possible in OWL. Rei includes meta policy specifications
 7884 for conflict resolution, speech acts for remote policy management, and policy analysis
 7885 specifications like what-if analysis and use-case management—making it a suitable
 7886 candidate for adaptable security in the environments under consideration. The Rei engine,
 7887 developed in XSB (extended Prolog), reasons over Rei policies and domain knowledge in
 7888 Resource Description Framework (RDF) and OWL to provide answers about the current
 7889 permissions and obligations of an entity, which are used to guide the entity's behavior.

- 7890 • (U) The Web Services Policy Framework (WS-Policy) was developed by BEA Systems
 7891 Inc., IBM Corporation, Microsoft Corporation, and SAP AG. The WS-Policy
 7892 specification provides a general-purpose model and corresponding syntax to describe and
 7893 communicate the policies of a Web service. The goal of WS-Policy is to provide the
 7894 mechanisms needed to enable Web Services applications to specify policy information.
 7895 WS-Policy by itself does not provide a negotiation solution for Web Services. WS-Policy
 7896 is a building block that is used in conjunction with other Web Service and application-
 7897 specific protocols to accommodate a wide variety of policy exchange models.

- 7898 • (U) Knowledgeable Agent-oriented System (KAoS) is a collection of component agent
 7899 services compatible with several popular agent frameworks, including Nomads, the
 7900 DARPA CoABS Grid, the DARPA ALP/Ultra*Log Cougaar framework, CORBA, and
 7901 Voyager. The adaptability of KAoS is due in large part to its pluggable infrastructure
 7902 based on Sun's Java Agent Services (JAS). KAoS policy services is developed by The
 7903 Institute for the Interdisciplinary Study of Human & Machine Cognition (IHMC) under
 7904 NASA and DARPA sponsorship.

- 7905 • (U) KAoS policy services allow for the specification, management, conflict
 7906 resolution, and enforcement of policies within domains. Policies are represented in
 7907 DAML (DARPA Agent Markup Language) as ontologies. The KAoS Policy
 7908 Ontologies (KPO) distinguish between authorizations (i.e., constraints that permit or

7909
7910
7911
7912
7913
7914

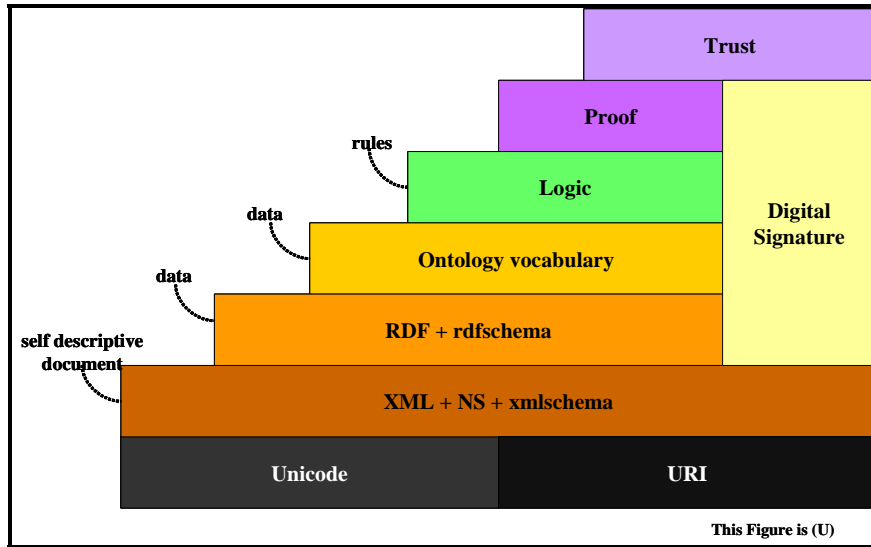
forbid some action) and obligations (i.e., constraints that require some action to be performed—or else serve to waive such a requirement). Through various property restrictions in the action type, a given policy can be variously scoped, for example, either to individual agents, to agents of a given class, to agents belonging to a particular group, or to agents running in a given physical place or computational environment (e.g., host, VM).

7915
7916
7917
7918
7919

- (U) KAoS framework supports dynamic runtime policy changes and is extensible to a variety of execution platforms that might be simultaneously running with different enforcement mechanisms. Currently KAoS supports agent platforms implemented in Java and Aroma, but could be adapted to work with other platforms for which policy enforcement mechanisms are written.

7920
7921
7922
7923
7924
7925

- (U) Semantic Web Rule Language (SWRL) was produced as part of the DARPA DAML Program. SWRL is built on top of the W3C Ontology layer (OWL DL and OWL lite and a subset of RuleML, a Rule Markup Language). As such SWRL implements Frame Logic that unfortunately omits the Deontic Modal Operators, (i.e., 'P' "it is permitted that", 'O' "it is obligatory that", and 'F' "it is forbidden that"). SWRL can be used as the logic layer in Berners-Lee's seven-layer model of the Semantic Web. See Figure 2.4-2 below.



7926

Figure 2.4-2: (U) Berners-Lee's Seven Layer Model of the Semantic Web

7927

7928 2.4.3.1.3.2.2 (U) Advantages

7929
7930
7931
7932

(U//FOUO) Having one policy language would make it easier for the person managing the GIG policy to understand. A single common policy language would also greatly simplify the GIG policy management components (i.e., Policy Input Point, Policy Repository, and Policy Decision Points).

7933
7934

(U//FOUO) A single policy language would also simplify the translation to device specific configuration files needed at the policy enforcement points.

7935 2.4.3.1.3.2.3 (U) Risks/Threats/Attacks
 7936 (U//FOUO) Need to verify that what is put in the language actually gets translated into the device
 7937 configuration files correctly. This will require verification testing prior to a new policy entering
 7938 the GIG.

7939 (U//FOUO) Also need authentication and integrity protection on the messages to prevent
 7940 spoofing and possibly confidentiality to protect sensitive policy data. This can be either
 7941 implemented directly in the policy protocol—or implemented in a lower layer protocol, like
 7942 IPsec or transport layer security (TLS).

7943 **2.4.3.1.3.3 (U) Maturity**

7944 (U//FOUO) Several policy languages are being used by commercial products today:

- 7945 • (U) Sun’s xacml: <http://sunxacml.sourceforge.net/>
- 7946 • (U) Ponder toolkit: <http://www-dse.doc.ic.ac.uk/Research/policies/ponder.shtml>
- 7947 • (U) KeyNote toolkit: [http://lists.netfilter.org/pipermail/netfilter/1999-](http://lists.netfilter.org/pipermail/netfilter/1999-October/002634.html)
 7948 [October/002634.html](http://lists.netfilter.org/pipermail/netfilter/1999-October/002634.html)
- 7949 • (U) KAoS toolkit: <http://www.ihmc.us/research/projects/KAoS/>
- 7950 • (U) Cisco’s QoS Policy Management:
 7951 <http://www.cisco.com/warp/public/cc/pd/wr2k/qoppmn/>
- 7952 • (U) Nortel’s Optivity Suite:
 7953 <http://www.nortelnetworks.com/products/01/optivity/policy/index.html>

7954 (U//FOUO) The various sub-technologies of the policy language technology area can be
 7955 generally assigned Technology Readiness Level groups of Early, Emerging, and Mature.

- 7956 • (U//FOUO) Routing and access control languages—Mature (TRLs 7 –9)
- 7957 • (U//FOUO) Extensible policy languages—Emerging (TRLs 4 – 6)
- 7958 • (U//FOUO) Security incorporated into policy languages—Early (TRLs 1 – 3)
- 7959 • (U//FOUO) Verification/test of policy languages—Early (TRLs 1 – 3)
- 7960 • (U//FOUO) Handling policy conflicts—Early (TRLs 1 – 3).

7961 **2.4.3.1.3.4 (U) Standards**

7962 **Table 2.4-3: (U) Policy Language Standards**

This Table is (U)	
Standard	Description
Extensible Access Control markup Language	XACML provides fine-grained control of authorized activities, the effect of characteristics of the access requestor, the protocol over which the request is made, authorization based on classes of activities, and content introspection.

This Table is (U)	
Standard	Description
(XACML)	
Routing Policy Specification Language (RPSL)	RPSL allows a network operator to be able to specify routing policies at various levels in the Internet hierarchy. Policies can be specified with sufficient detail in RPSL so that low-level router configurations can be generated from them. RPSL is extensible; new routing protocols and new protocol features can be introduced at any time.
Rei	A declarative policy language for describing policies over actions. It is possible to write Rei policies over ontologies in other semantic web languages.
KeyNote	KeyNote provides a simple language for describing and implementing security policies, trust relationships, and digitally signed credentials.
SDN.801	SDN.801 provides guidance for implementing access control concepts using both public key certificates and attribute certificates.
Security Assertion Markup Language (SAML)	SAML is an XML framework for exchanging authentication and authorization information.
Ponder	Ponder is a language for specifying management and security policies for distributed systems.
KAoS	KAoS policy services allow for the specification, management, conflict resolution, and enforcement of policies within domains.
This Table is (U)	

7963 **2.4.3.1.3.5 (U) Cost/Limitations**

7964 (U//FOUO) The policy language used by GIG will need to cover all GIG policies. This includes
7965 policies for access control, QoP, QoS, transport, audit, computer network defense, and policies
7966 covering the hardware and software associated with GIG assets.

7967 **2.4.3.1.3.6 (U) Dependencies**

7968 (U//FOUO) Need compilers to translate the policy language into configuration files that are used
7969 by the policy enforcement points. These configuration files are mostly vendor specific so a
7970 compiler would need to output many different formats.

7971 (U//FOUO) Also need testing and verification tools to test the policy language statements prior
7972 to distribution to the operational environment.

7973 **2.4.3.1.3.7 (U) Alternatives**

7974 (U) Generate new policy language to securely cover all the GIG policy management needs. This
7975 would be an expensive and time-consuming task.

7976 **2.4.3.1.3.8 (U) References**

7977 (U) http://www.parlay.org/about/policy_management/index.asp

7978 (U) www-106.ibm.com/developerworks/library/ws-secpol/

7979 (U) <http://sunxacml.sourceforge.net/guide.html>

- 7980 (U) RFC 2622
- 7981 (U) <http://www.comsoc.org/ni/private/2001/jan/stone.html>
- 7982 (U) <http://www.doc.ic.ac.uk/~mss/Papers/Ponder-Policy01V5.pdf>
- 7983 (U) <http://www.cis.upenn.edu/~keynote/>
- 7984 (U) <http://rei.umbc.edu/>
- 7985 (U) http://www.ihmc.us/research/projects/KAoS/FinalIHMC_DEIS.pdf
- 7986 (U) <http://www.parthenoncomputing.com>
- 7987 (U) <http://sunxacml.sourceforge.net/>
- 7988 (U) <http://mypos.sourceforge.net/>
- 7989 (U) <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnglobspec/html/ws-policy.asp>
- 7990 (U) http://www.wiwiss.fu-berlin.de/suhl/bizer/SWTSGuide/KAoS/KAoS_Policy_03.pdf
- 7991 (U) <http://www.ihmc.us/research/projects/KAoS/>
- 7992 (U) <http://www.daml.org/2003/11/swrl/rules-all.html>

7993 **2.4.3.2 (U) Distribution of Policies**

7994 **2.4.3.2.1 (U) Standard Protocols**

7995 **2.4.3.2.1.1 (U) Technical Detail**

7996 (U//FOUO) Distribution of dynamic material is required to configure the policy enforcement
7997 points, through the use of GIG policy files. After the files are created and validated, the policy is
7998 distributed using a push or pull model. The push model would be used for policy changes that
7999 must take effect immediately because new behavior is needed in reaction to the current
8000 condition. The pull model can be used in cases in which a policy change is scheduled to take
8001 effect at a particular time and is not critical to current operations.

8002 (U//FOUO) Policy distribution extends from the policy input point to the Policy Enforcement
8003 Points (PEP). PEPs are those GIG assets that enforce the GIG rules. (See section 2.4.2 for more
8004 information on PEP.) PEPs include routers, firewalls, guards, and other networking equipment
8005 that require configuration files to enforce policy. Most PEP equipment is currently configured
8006 manually by network support personnel. But some policy management products are using
8007 directories to store policy configuration information and Light-weight Directory Access Protocol
8008 (LDAP) to distribute the configuration files.

8009 (U//FOUO) These policy enforcement configuration files are generally vendor specific and only
8010 support routing and access control policy decisions. The policy distribution point will need to
8011 know the type of PEP when distributing new policy so that the policy can be in the correct
8012 configuration format for the specific PEP.

8013 (U//FOUO) It is highly critical that the GIG program work with PEP vendors to expand PEP
8014 capabilities and possibly standardize policy enforcement configuration files to reduce policy
8015 management overhead. Common Open Policy Service (COPS) protocol and Command Line
8016 Interface (CLI) commands are two enforcement configuration formats currently being used.

8017 (U//FOUO) COPS is a query and response protocol that the PDP and PEP can use to exchange
8018 policy information. COPS uses the Transmission Control Protocol (TCP) to transfer the
8019 messages.

8020 (U//FOUO) There are other options for distributing the policy updates. Administrators can send
8021 users an email with a URL where users can download the update, or use a facility such as
8022 Microsoft's System Management Server (SMS) to automatically push the updates out to
8023 distributed end points.

8024 (U//FOUO) Another alternative is to use the CyberwallPLUS policy pull feature. Each time a
8025 user logs on to the network, the software checks a central policy database to ensure the user has
8026 the most current policy configuration.

8027 **2.4.3.2.1.2 (U) Usage Considerations**

8028 2.4.3.2.1.2.1 (U) Implementation Issues

8029 (U) Current policy management products are mostly vendor specific. Policy distribution formats
8030 must be agreeable with the network products receiving the policy information.

8031 2.4.3.2.1.2.2 (U) Advantages

8032 (U//FOUO) Automating the distribution of policy information would be a significant savings
8033 over the current manual configuration of PEPs. To fully take advantage of this automated
8034 distribution, the integrity and authentication of the delivery must be verifiable to insure that the
8035 policy was received unchanged from a trusted source.

8036 (U) Having a common distribution protocol would greatly simplify the distribution process to the
8037 network components.

8038 2.4.3.2.1.2.3 (U) Risks/Threats/Attacks

8039 (U//FOUO) Policy data must be protected from the time the policy is created at the policy input
8040 point to the time the policy reaches the policy enforcement points. This requires identification
8041 and authentication of the person creating new policies. It also requires authentication, integrity,
8042 and confidentiality of the policy data as it passes through the GIG policy management system.

8043 **2.4.3.2.1.3 (U) Maturity**

8044 2.4.3.2.1.3.1 (U) DMS Example

8045 (U//FOUO) DMS has a trusted policy distribution system with both manual and automated
 8046 procedures. With DMS, the rule-based access control policy is held in the SPIF. An External
 8047 Source, such as a policy making body, generates the security policies used by DMS. This
 8048 information is delivered to a root authority in an unsigned SPIF format on a trusted physical
 8049 path. The root authority reviews and approves the security policy before signing the SPIF. After
 8050 signing an SPIF, the root authority distributes it to the subordinate authorities that support the
 8051 security policy defined in the SPIF. The root authority can maintain multiple SPIFs, but the
 8052 subordinate authorities only need to receive the SPIFs for the security policy(s) they support.

8053 (U//FOUO) The sub-authority verifies the received SPIF has been signed by the root authority
 8054 and is valid. Next, the sub-authority removes the root authority signature, updates the issuer and
 8055 date information, and re-signs the SPIF. The sub-authority then posts the SPIF to the directory
 8056 and distributes the SPIF to the rest of the authority hierarchy.

8057 (U//FOUO) User applications and devices using the SPIF will periodically retrieve the SPIF
 8058 from the directory, verify the signature of the SPIF, and use the SPIF for access control
 8059 decisions.

8060 2.4.3.2.1.3.2 (U) Vendor Distribution Example

8061 (U//FOUO) Most network component vendors (e.g., Cisco, Juniper, Ciena, and Nortel) have
 8062 configuration formats and distribution methods that are specific to their equipment. Distribution
 8063 methods include LDAP, File Transfer Protocol (FTP), Telnet, and Secure Server Protocol (SSP).

8064 (U//FOUO) The various sub-technologies of the policy distribution technology area can be
 8065 generally assigned Technology Readiness Level as follows.

- 8066 • (U//FOUO) Distribution protocols—Mature (TRLs 7 – 9)
- 8067 • (U//FOUO) PEP configuration file standard—Early (TRLs 1 – 3)

8068 **2.4.3.2.1.4 (U) Standards**

8069 **Table 2.4-4: (U) Distribution Standards**

This Table is (U)	
Standard	Description
LDAP	LDAP is an Internet protocol used to look up information from a LDAP server or directory. LDAP servers index all the data in their entries, and "filters" may be used to select just the information you want. "Permissions" and "authentications" can be set by the administrator to allow only certain people to access the LDAP database, and optionally keep certain data private. Reference http://www.ldap-directory.org/rfc-ldap for a list of LDAP RFCs.
File Transfer Protocol (FTP)	File Transfer Protocol (FTP), a standard Internet protocol, is the simplest way to exchange files between computers on the Internet. FTP is an application protocol that uses the Internet's TCP/IP protocols.

This Table is (U)	
Standard	Description
	Reference RFC959: http://www.w3.org/Protocols/rfc959/
Common Open Policy Service (COPS)	The Common Open Policy Service (COPS) protocol is a simple query and response protocol that can be used to exchange policy information between a policy server (PDP) and its clients (PEPs). Reference http://www.networksorcery.com/enp/protocol/cops.htm for a list of COPS related RFCs
Microsoft's SMS	SMS provides a solution for change and configuration management for the Microsoft platform, enabling organizations to provide relevant software and updates to users quickly and cost effectively.
Telnet	The Telnet program allows you to connect your PC to a server on the network using a username and password. You can then enter commands through the Telnet program, and they will be executed as if you were entering them directly on the server console.
This Table is (U)	

8070 **2.4.3.2.1.5 (U) Dependencies**

8071 (U//FOUO) PEP configuration formats are mostly vendor specific. Creating a standard for this
8072 configuration format would require support from many network component vendors.

8073 **2.4.3.2.1.6 (U) Alternatives**

8074 (U//FOUO) For policy distribution, there are many existing protocols that can be used to safely
8075 distribute the GIG policy throughout the system.

8076 (U//FOUO) GIG-developed common protocol for format of all GIG policy enforcement points.

8077 **2.4.3.2.1.7 (U) Complementary Techniques**

8078 (U//FOUO) Security features can also be applied to policy distribution if required by the GIG
8079 program. Directories can be configured to limit write access to the policy information so only
8080 authorized persons can create and update GIG policy information stored in the directory.

8081 (U//FOUO) Authentication and confidentiality can also be applied to the policy distribution by
8082 adding additional levels of protection to the policy data. A protocol such as Secure Sockets Layer
8083 (SSL) allows the server and client to authenticate each other and to negotiate an encryption
8084 algorithm and cryptographic keys before the application protocol transmits or receives its first
8085 byte of data. One advantage of SSL is that it is application-protocol independent.

8086 **2.4.3.2.1.8 (U) References**

8087 (U) FORTEZZA[®] Security Management Infrastructure (SMI) Concept of Operation CONOP) for
8088 CipherNET[®] 3000 CAW 5.0

8089 (U) <http://wp.netscape.com/eng/ssl3/ssl-toc.html>

8090 (U) <http://www.nortelnetworks.com/products/01/optivity/policy/index.html>

8091 (U) http://www.parlay.org/about/policy_management/index.asp

8092 **2.4.3.2.2 (U) Security Issues**

8093 **2.4.3.2.2.1 (U) Technical Detail**

8094 (U//FOUO) Policy data must be protected from the time the policy is created at the policy input
8095 point to the time the policy reaches the policy enforcement points. This requires identification
8096 and authentication of the person creating new policies. It also requires authentication and
8097 integrity of the policy data as it passes through the GIG policy management system.

8098 (U//FOUO) Policy data can provide great value to an attacker to know exactly what rules the
8099 infrastructure is enforcing. Confidentiality may also be required if the policy data contains
8100 sensitive data. Having a common configuration file format would also make it easier for an
8101 attacker to understand policy changes when they are sent to the PEPs. This is another reason
8102 confidentiality should be applied to this enforcement configuration file so outside sources cannot
8103 change or see the PEP's configuration.

8104 (U//FOUO) Policy repository directories can be configured to limit the read and write access to
8105 policy information so only authorized persons can read and update GIG policy information
8106 stored in the directory.

8107 (U//FOUO) Authentication and confidentiality can also be applied to the policy distribution by
8108 adding more levels of protection to the policy data. A protocol such as SSL allows the server and
8109 client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys
8110 before the application protocol transmits or receives its first byte of data. One advantage of SSL
8111 is that it is application-protocol independent.

8112 **2.4.3.2.2.2 (U) Usage Considerations**

8113 2.4.3.2.2.2.1 (U) Implementation Issues

8114 (U//FOUO) Currently, none of the policy languages incorporate the security features required for
8115 secure GIG dynamic policy distribution. So either a new GIG-defined protocol could be
8116 developed that includes the security features or existing security protocols (e.g., SSL, IPsec, or
8117 TLS) can be added to the policy distribution procedures.

8118 2.4.3.2.2.2.2 (U) Advantages

8119 (U//FOUO) Using a COTS solution for policy distribution security provides an immediate cost
8120 and schedule advantage over a new secure policy language or policy distribution protocol.

8121 2.4.3.2.2.2.3 (U) Risks/Threats/Attacks

8122 (U//FOUO) Having a secure policy distribution path will greatly reduce the risk of threats or
8123 attacks on the dynamic policy management system.

8124 **2.4.3.2.2.3 (U) Maturity**

8125 (U//FOUO) Current COTS solutions (e.g., SSL-TLS or IPsec) are very well defined and
8126 available. The following products are commercially available today and are candidates for GIG
8127 secure policy distribution:

- 8128 • (U) SSL-TLS Products:
 - 8129 • (U) F5 Networks Inc., Firepass
 - 8130 • (U) RSA Security Inc., RSA BSAFE® SSL-J
 - 8131 • (U) Thawte Consulting (Pty) Ltd , Thawte SSL Web Server Certificate
 - 8132 • (U) GeoTrust, Inc., QuickSSL® Premium
 - 8133 • (U) Canfone.com Web Services, eSecure 128-bit SSL Hosting
 - 8134 • (U) OpenConnect Systems, Incorporated, Secure ClientConnect
 - 8135 • (U) Citrix Systems, Inc., Citrix MetaFrame Access Suite: Secure Gateway
 - 8136 • (U) Entrust, Inc., Entrust Authority™ Toolkits
 - 8137 • (U) Ingrian Networks, Inc., Ingrian i225 - Secure Transaction Platforms
 - 8138 • (U) VeriSign, Inc., Managed PKI for SSL Certificate
 - 8139 • (U) Valicert, Inc., Valicert SecureTransport™
- 8140 • (U) IPsec Products:
 - 8141 • (U) Check Point Software Technologies Ltd., Checkpoint Secure Platform AI R55
 - 8142 • (U) DrayTek, Vigor 3300 Version
 - 8143 • (U) Enterasys Networks, XSR 3000 Series
 - 8144 • (U) Intoto Inc., iGateway
 - 8145 • (U) NetScreen Technologies, Inc., NetScreen Security Gateway Product Group
 - 8146 • (U) Novell, Novell BorderManager
 - 8147 • (U) Secure Computing Sidewinder G2 Firewall
 - 8148 • (U) Cisco Systems, Inc., Cisco VPN Client
 - 8149 • (U) CentricVoice, CentricVoice's IPsec VPN
 - 8150 • (U) Entrust, Inc., Entrust Authority™ Toolkits.

8151 (U//FOUO) The various sub-technologies of the distribution security technology area can be
 8152 generally assigned Technology Readiness Level groups of Early, Emerging, and Mature.

- 8153 • (U//FOUO) COTS SSL-TLS and IPsec products—Mature (TRLs 7 – 9)
- 8154 • (U//FOUO) Security embedded into policy languages—Early (TRLs 1 –3).

8155 **2.4.3.2.2.4 (U) Standards**

8156 **Table 2.4-5: (U) Distribution Security Standards**

This Table is (U)	
Standard	Description
SSL	SSL is designed to make use of TCP as a communication layer to provide a reliable end-to-end secure and authenticated connection between two points over a network.
TLS	RFC2246: The primary goal of the Transport Layer Security (TLS) Protocol is to provide privacy and data integrity between two communicating applications. The protocol is composed of two layers: the TLS Record Protocol and the TLS Handshake Protocol. At the lowest level, layered on top of some reliable transport protocol (e.g., TCP), is the TLS Record Protocol. The TLS Record Protocol provides connection security that provides confidentiality and integrity. TLS is designed as a successor to SSL and is sometimes called SSL V3.0.
IPsec	RFC 2401: Internet Protocol Security (generally shortened to IPsec) is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers at the IP layer. IPsec can be used to protect one or more data flows between IPsec peers.
This Table is (U)	

8157 **2.4.3.2.2.5 (U) Cost/Limitations**

8158 (U//FOUO) A limitation with a COTS solution is how DoD PKI (or other GIG key credentials)
8159 would be integrated into COTS products. This assumes that GIG policy distribution would
8160 require the use of GIG keys.

8161 **2.4.3.2.2.6 (U) Alternatives**

8162 (U) The alternative to using COTS security solution for policy distribution would be to develop a
8163 secure policy distribution protocol for the GIG system.

8164 **2.4.3.2.2.7 (U) References**

8165 (U) <http://www.faqs.org/rfcs/rfc2246.html>

8166 (U) <http://www.faqs.org/rfcs/rfc2401.html>

8167 (U) <http://www.bitpipe.com/plist/SSL.html>

8168 (U) <http://www.bitpipe.com/plist/IPSec.html>

8169 (U) [http://www.icsalabs.com/html/communities/ipsec/certification/certified_products/1.0Dindex.s](http://www.icsalabs.com/html/communities/ipsec/certification/certified_products/1.0Dindex.shtml)
8170 [html](http://www.icsalabs.com/html/communities/ipsec/certification/certified_products/1.0Dindex.shtml)

8171 **2.4.3.3 (U) Policy Management Architectures**

8172 (U//FOUO) One example of a policy management architecture is described in the paper titled
 8173 “Distributed Multi-National Network Operation Centres” by Scott Shyne (AFRL), David Kidson
 8174 (CRC), and Peter George (DSTO). This paper describes a coalition network management
 8175 architecture to use between Australia, Canada, and the U.S. A. This policy-based network
 8176 management system was developed to manage the coalition domain’s network quality of service
 8177 configuration. The system consists of a domain policy integration manager, policy distribution
 8178 points, policy enforcement points, and policy delivery protocol. The high level XML policy
 8179 statements are used to constitute a defined course of action for coalition domains. Each domain
 8180 must break down the policy into configuration files for use by the network entities for policy
 8181 enforcement. Local policy is introduced at this level to further define domain operations.

8182 (U//FOUO) Another example is the commercial product SecureSpan, by Layer 7 Technologies.
 8183 SecureSpan addresses web service security, trust establishment, enterprise policy management,
 8184 and dynamic policy from the Transport layer through the Application layer. SecureSpan is made
 8185 up of three major components: SecureSpan Manager, SecureSpan Gateway, and SecureSpan
 8186 Agent. See <http://www.layer7tech.com/products/>

- 8187 • (U) The SecureSpan Manager is a GUI-based application that enables administrators to
 8188 centrally define, provision, monitor, and audit security and integration policies for Web
 8189 services.
- 8190 • (U) The SecureSpan Gateway is a rack-mountable, high-performance network appliance
 8191 enforces policy on every Web service provisioned through the SecureSpan Manager. The
 8192 Gateway identifies and processes each message under the policy created for the service. It
 8193 shields access to internal services, ensuring that only those messages that meet all
 8194 security and integration policy requirements are forwarded to the destination service.
- 8195 • (U) The SecureSpan Agent interfaces with client-side applications and automatically
 8196 negotiates policy-specific security, routing, and transaction preferences with the
 8197 SecureSpan Gateway.

8198 (U//FOUO) The policy management architecture described in Section 2.4.2 above includes a
 8199 policy input point, policy repository, policy decision point, and policy enforcement point. A
 8200 technology that supports the policy repository is a policy directory, as described below.

8201 **2.4.3.3.1 (U) Policy Directories**

8202 **2.4.3.3.1.1 (U) Technical Detail**

8203 (U//FOUO) A policy directory can be used as a repository for policies, as well as device
 8204 information and administrative information needed for policy distribution, deconfliction,
 8205 synchronization, and promulgation.

8206 (U//FOUO) A directory has several beneficial features that can be used in policy management:

- 8207 • (U//FOUO) Directories can provide distributed policy management. As the GIG network
 8208 expands, additional directories can be added to handle new or expanded domains.

- 8209 • (U//FOUO) Directories also have the ability to shadow or replicate the policy information
8210 between policy directories. This capability greatly simplifies the maintenance and
8211 management of policy information as policies change or as the network grows.
- 8212 • (U//FOUO) Directories can also be partitioned to limit access to sensitive data stored in
8213 the directory. Partitioning can be configured so that only certain users can have write
8214 access to the policy information stored in the directory. Partitioning can also be used to
8215 limit read access to only the policies that apply to a specific user or device.

8216 **2.4.3.3.1.2 (U) Usage Considerations**

8217 2.4.3.3.1.2.1 (U) Implementation Issues

8218 (U) Nortel Networks Optivity Policy Services (OPS) is a software application designed to
8219 manage network QoS and network access security. The Nortel OPS product uses a directory as
8220 the policy repository. This directory is used to store policies, device information, and related
8221 administrative information required by OPS.

8222 (U) Netegrity's SiteMinder product and DMS also use a directory to store critical policy
8223 information used in making access control decisions.

8224 2.4.3.3.1.2.2 (U) Advantages

8225 (U//FOUO) The main advantages to using a directory to store GIG policy information are:

- 8226 • (U//FOUO) Directories have flexible storage schemas to store all types of policy
8227 information
- 8228 • (U//FOUO) Directories have defined interface protocols for access to the data
- 8229 • (U//FOUO) Directories can limit read and write access to the data
- 8230 • (U//FOUO) Directories have chaining capabilities that can keep information
8231 synchronized between different directories

8232 2.4.3.3.1.2.3 (U) Risks/Threats/Attacks

8233 (U//FOUO) A policy directory would need to be well-protected against improper access to the
8234 data stored in the directory. Directories have a binding process where they determine if a person
8235 requesting access is who they are and if they should be granted access information stored in the
8236 directory.

8237 **2.4.3.3.1.3 (U) Maturity**

8238 (U//FOUO) Using directories for storing network and system information is very mature. Strong
8239 binds and SSL tunnels to directories to make more secure interfaces to the directory data are also
8240 in use. There may be additional work needed in the directory access security, depending on the
8241 required level of authentication for the GIG program.

8242 (U//FOUO) The various sub-technologies of the policy directories technology area can be
8243 generally assigned Technology Readiness Level groups of Early, Emerging, and Mature.

- 8244 • (U//FOUO) Directory standards—Mature (TRLs 7 – 9)
- 8245 • (U//FOUO) Directory security—Emerging (TRLs 4 – 6).

8246 **2.4.3.3.1.4 (U) Standards**

8247 **Table 2.4-6: (U) Directory Standards**

This Table is (U)	
Standard	Description
X.500	X.500 is a CCITT protocol that is designed to build a distributed, global directory. It offers decentralized maintenance, searching capabilities, single global namespace, structured information framework, and a standards-based directory.
Finger, whois, domain name	These are very simple directory formats that are also in use.
This Table is (U)	

8248 **2.4.3.3.1.5 (U) Alternatives**

8249 (U//FOUO) Using a database for the policy repository is an alternative to the directory approach.
 8250 The database could store all policy information, and a secure interface could be written to control
 8251 access to the data.

8252 **2.4.3.3.1.6 (U) References**

8253 (U) <http://www.nortelnetworks.com/products/01/optivity/policy/index.html>

8254 (U) “The Directory: Overview of Concepts, Models and Service,” CCITT Recommendation
 8255 X.500, 1988.

8256 (U) <http://www.netegrity.com/products/products.cfm?page=productsoverview>

8257 **2.4.4 (U) Dynamic Policy Management: Gap Analysis**

8258 (U) Gap analysis for the Dynamic Policy Management Enabler indicates that the main areas of
 8259 future development are as follows:

- 8260 • (U//FOUO) Need to further expand the extensible policy languages to cover the complete
 8261 set of GIG policies. Some existing policy languages such as Ponder, KAoS, Rei, and
 8262 XACML are flexible in that they allow you to define new policy within the language.
 8263 GIG should further research these flexible policy languages to see which would be best
 8264 suited for GIG policies.
- 8265 • (U//FOUO) Need to develop/refine network modeling and simulation tools used to assess
 8266 the impact of candidate global and local policy configuration changes on operational risk,
 8267 network loads and network/application interactions. These policy management testing
 8268 tools must ensure security requirements for asset usage are not violated. The Ponder
 8269 toolkit has some capabilities in this gap area.
- 8270 • (U//FOUO) Need to develop automated policy deconfliction tools. The KAoS policy
 8271 service and Rei product have some policy confliction resolution capabilities, but these

8272 tools will need to be further developed for the GIG program. Initial versions of this tool
8273 may require operator intervention to settle conflicts between policies. As the
8274 deconfliction tools mature, this process will become more automated.

- 8275 • (U//FOUO) Need to develop tools or compilers to translate policy language into a device
8276 interpretable language such as a router configuration file. These configuration files are
8277 generally vendor specific. Standardizing the end network device configuration formats
8278 would greatly simplify this task.

8279 (U//FOUO) Technology adequacy is a means of evaluating the technologies as they currently
8280 stand. This data can be used as a gap assessment between a technology's current maturity and the
8281 maturity needed for successful inclusion.

8282 (U//FOUO) The Table 2.4-7 lists the adequacy of the dynamic policy management technologies
8283 with respect to the enabler attributes discussed in the RCD. Gray entries currently have no
8284 technology available, and no research is underway to develop the needed technology. The gray
8285 grid entries represent insufficient technology. Solid black entries are adequate today.

8286

Table 2.4-7: (U) Technology Adequacy for Dynamic Policy Management

This Table is (U)						
		Technology categories				Required Capability (attribute from RCD)
		Policy Distribution	Policy languages	Trust Anchor	Policy Enforcement Configuration	
Enabler Attributes	Secure solution					IACNF6, IACNF12, IAIN1, IAPOL6, IAIAC8, IAIAC6, IAIAC9, IACM11, IAAV20
	Standard format					IAPOL8, IAPOL9, IAIAC1, IAAUD7
	Verifiable solution					IACNF15, IAPOL5, IAPOL7, IACM2, IACM4, IACM5
	Policy synchronization and deconfliction					IAAV4, IAPOL1, IAPOL3, IAPOL4, IACM9, IARC08, IARC09
This Table is (U)						

8287 **2.4.5 (U) Dynamic Policy Management: Recommendations and Timelines**

8288 (U//FOUO) The following gaps have been identified in the Dynamic Policy Management
 8289 Enabler. Without these, this Enabler cannot be fully satisfied. The technology gaps can be of the
 8290 following types—Standards, Technology, and Infrastructure.

8291 **2.4.5.1 (U) Standards**

- 8292 • (U//FOUO) Standards for specifying policy. The policy language needs to cover all GIG
 8293 policies: access control, quality of protection, quality of service, transport, audit,
 8294 computer network defense, and policies covering the hardware and software associated
 8295 with GIG assets. Candidate policy languages include:

- 8296 • (U) XACML
- 8297 • (U) Ponder
- 8298 • (U) K AoS
- 8299 • (U) Security Assertion Markup Language (SAML)
- 8300 • (U) Rei
- 8301 • (U//FOUO) Policy deconfliction standard for how to handle policy conflicts
- 8302 • (U//FOUO) Policy Distribution Standard (push and pull), including protection of policies
- 8303 at rest and in transit, policy validation, distribution error and exception handling
- 8304 • (U//FOUO) Standard for managing authorities that can promulgate policy and delegate
- 8305 their authority

8306 **2.4.5.2 (U) Technology**

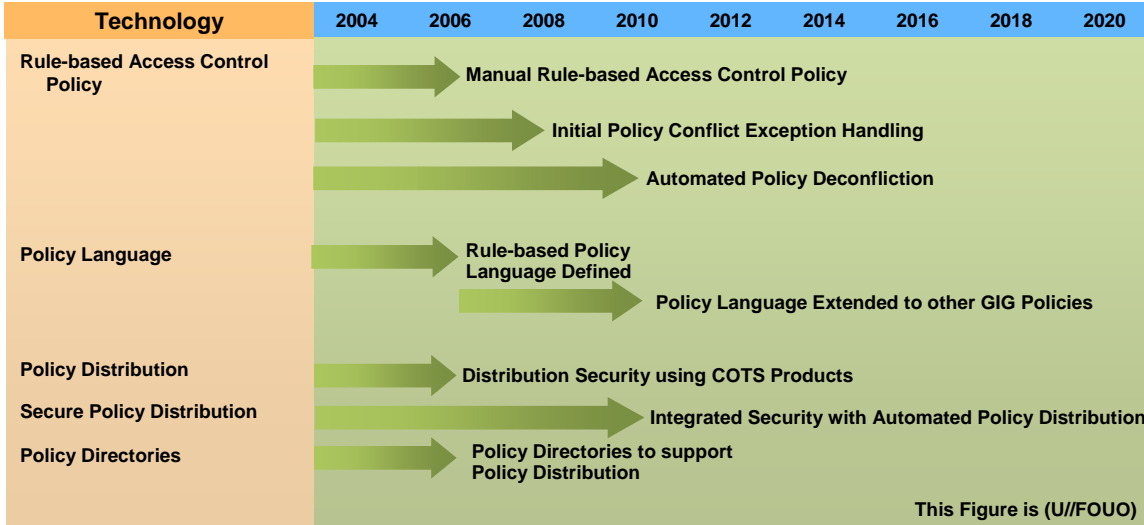
- 8307 • (U//FOUO) Mechanisms and performance analysis of policy specification languages and
- 8308 translation to device interpretable language
- 8309 • (U//FOUO) Performance analysis of various methods of distributing policies (pull and
- 8310 push approaches) to support Policy Distribution Standard
- 8311 • (U//FOUO) Methods for performing policy synchronization
- 8312 • (U//FOUO) Tools for analyzing affects of policy and multiple policy objects on overall
- 8313 system
- 8314 • (U//FOUO) Life cycle model for policy objects
- 8315 • (U//FOUO) Application of artificial intelligence, heuristics, learning systems, etc., to
- 8316 policy management

8317 **2.4.5.3 (U) Infrastructure**

8318 (U) Policy management infrastructure that provides:

- 8319 • (U) Single Graphical User Interface (GUI) for managing multiple classes of assets
- 8320 • (U//FOUO) Tools for translating automated human language policies into policy base logic
- 8321 • (U//FOUO) Tools for policy deconfliction
- 8322 • (U//FOUO) Integrity protection for all policy storage and transfer
- 8323 • (U//FOUO) Authentication services on all policy exchanges
- 8324 • (U//FOUO) Logging all policy management transactions

- 8325 • (U//FOUO) Signed receipts in response to received policy information
- 8326 (U//FOUO) Figure 2.4-3 contains technology timelines for the Dynamic Policy Management
- 8327 Enabler. These are the results of research completed to date on these technologies. These
- 8328 timelines are expected to evolve as the Reference Capability Document and the research of
- 8329 technologies related to these capabilities continues. The timelines reflect when the technologies
- 8330 could be available given an optimum set of conditions (e.g., commercial community evolution
- 8331 starts immediately, GOTS funding is obtained, staffing is available). Technology topics with
- 8332 missing timelines indicate areas where further work is needed to identify the milestones.



8333 **Figure 2.4-3: (U) Technology Timeline for Dynamic Policy Management**

8334

8335

8336 **2.5 (U) ASSURED RESOURCE ALLOCATION**

8337 (U//FOUO) Assured Resource Allocation Enabler maintains the integrity and availability of all
8338 enterprise resources (e.g., communication, computing, and core services) and ensures those
8339 resources are available to GIG entities—based on operational needs. GIG resources include
8340 bandwidth, QoS and priority, processing cycles, access to GIG services, the network
8341 management system, routes, and similar assets. Management and allocation of these resources
8342 are required for the GIG to meet its operational requirements to provide services to users.

8343 (U//FOUO) This Enabler does not cover the topic of initially designing and implementing the
8344 GIG to provide sufficient resources for any end user to accomplish a mission. That is more
8345 properly the responsibility of systems engineering and design.

8346 (U//FOUO) This Enabler also does not assume that all GIG users will require resource
8347 management services. It assumes the capability needs to exist to deconflict shared resources and
8348 to support better-than-best effort service for users that require greater QoS or priority to meet
8349 their mission needs.

8350 (U//FOUO) Assured management and allocation includes protecting these management and
8351 allocation functions from failures or attacks. It also includes ensuring that no attack or failure can
8352 put the GIG into a state where customers cannot get resources to at least the level defined in
8353 service level agreements (SLA).

8354 (U//FOUO) Assured Resource Allocation must ensure the availability of computing and
8355 communications resources to both GIG infrastructure components and end users. GIG and non-
8356 GIG users, processes, and services must not be able to exceed their authorizations and thereby
8357 deny or degrade or co-opt services of other GIG users.

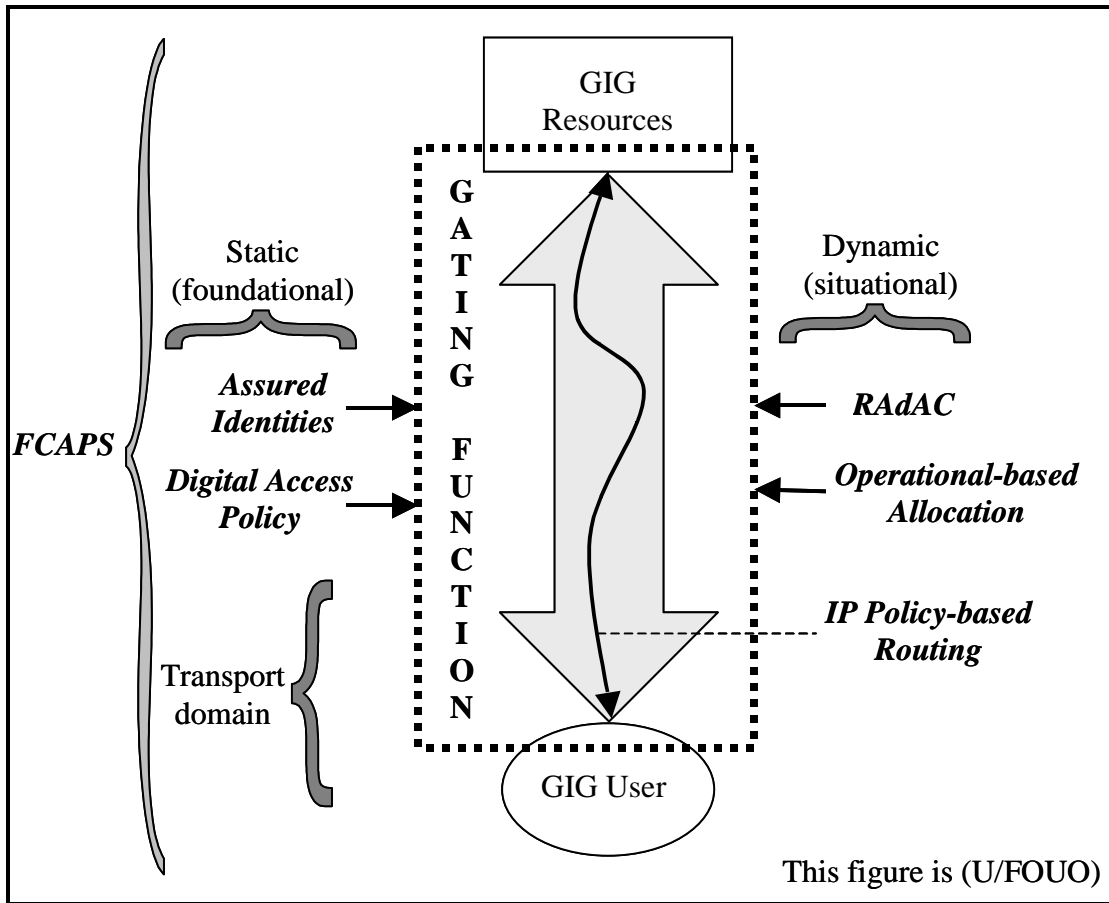
8358 (U//FOUO) To meet the GIG 2020 Vision, the GIG architecture must support a number of
8359 features. The essential features include:

- 8360 • (U) Assured Identities
- 8361 • (U) Digital Access Policy
- 8362 • (U) IA Policy-based Routing
- 8363 • (U) Operational-Based Resource Allocation
- 8364 • (U) RAdAC
- 8365 • (U) Fault Management, Configuration Management, Accounting Management,
8366 Performance Management, and Security Management (FCAPS).

8367 (U//FOUO) These six features are the components of assured management and control of
 8368 network resources. They combine to provide assurance to the GIG user that requested GIG
 8369 resources will be available in a securely and equitably managed manner that considers both the
 8370 nominal/normal privilege status of that user in addition to when the GIG user demand privileges
 8371 are increased (or decreased) by unique mission or environmental conditions. Their notional
 8372 interactions may be visualized in Figure 2.5-1.

8373 (U//FOUO) In Figure 2.5-1, the Assured Resource Allocation Enabler acts as a gating function
 8374 between GIG resources and GIG users. Four of the six components—RAdAC, assured identities,
 8375 digital access policies, and operational-based resource allocation—act as gate modulators.

8376 (U//FOUO) IA Policy-based routing is a selected or controlled path within the overall path
 8377 availability to the user. FCAPS has a universal scope of applicability, which means that it
 8378 impacts all the other five architectural requirements.



8379

8380 **Figure 2.5-1: (U//FOUO) The Role and Components of Assured Resource Allocation**

8381 **2.5.1 (U) GIG Benefits of Assured Resource Allocation**

8382 (U//FOUO) The Assured Resource Allocation Enabler supports continued operation of the
8383 system in the face of design failures and hostile attacks. This IA system enabler ensures that
8384 there are adequate resources to manage and control the GIG and its attached systems. This
8385 enabler applies when:

- 8386 • (U//FOUO) All data passes through only GIG-controlled systems
- 8387 • (U//FOUO) Data is transmitted from one portion of the GIG to another through non-GIG
8388 controlled systems. GIG management data must also move between portions of the GIG
8389 to properly manage resources
- 8390 • (U//FOUO) User data passes from the GIG to end user systems through non-GIG
8391 controlled systems. GIG management data must flow between the GIG and the end
8392 system to ensure proper resource management.

8393 (U//FOUO) Assured Resource Allocation provides the following additional benefits to the GIG:

- 8394 • (U//FOUO) Ensures allocation of GIG resources to meet operational needs (e.g., priority
8395 and preemption)
- 8396 • (U//FOUO) Routes information based upon the specified IA policy, which must account
8397 for factors such as Quality of Protection (QoP) for the information, QoS, and priority for
8398 the information
- 8399 • (U//FOUO) Provides enforcement of QoP, QoS, and priority to ensure GIG entities do
8400 not exceed their authorizations to deny/degrade service of other GIG users
- 8401 • (U//FOUO) Provides network control across multiple disparate networks both within the
8402 GIG and across both GIG and non-GIG networks
- 8403 • (U//FOUO) Prevents unauthorized entities from accessing management and control data
8404 of the network and network assets

8405 **2.5.2 (U) Assured Resource Allocation: Description**

8406 (U//FOUO) The GIG core will have a management and allocation system consisting of two
8407 major components: the routing and allocation component and the management and control
8408 component. Each of the constituent transport programs of the GIG (e.g., Global Information
8409 Grid-Bandwidth Expansion [GIG-BE], Transformational Satellite (TSAT), and Joint Tactical
8410 Radio System [JTRS]) contains these two components. This fundamental system enabler
8411 addresses the IA aspects of these components.

8412 (U//FOUO) The management and control component of the GIG is responsible for monitoring
8413 the state of each of the GIG infrastructure components (e.g., communication, computing, and
8414 core services) and systems. This component also reacts to changes in the state (e.g., detecting an
8415 attack and reacting to it; detecting that a device has failed and taking steps to restart it or route
8416 around it).

8417 (U//FOUO) In order to achieve the provisioning of assured management of GIG resources, the
8418 following functions must be provided by the GIG:

- 8419 • (U//FOUO) Transfer of network control (i.e., performance, configuration) across multiple
8420 disparate networks (e.g., TSAT, GIG-BE, JTRS) and security domains to support
8421 Operational-Based Resource Allocation
- 8422 • (U//FOUO) QoS/CoS integrity and authorization and priority enforcement mechanisms to
8423 ensure that prioritization and precedence requirements are met and to defend against
8424 attacks that would allow attackers to hijack or monopolize resources by improperly
8425 claiming high priority traffic privileges
- 8426 • (U//FOUO) Threat-based Traffic Flow Security for network management data to prevent
8427 attackers from gaining information about the topology of the network in violation of a
8428 system security policy.

8429 (U//FOUO) GIG management and control must function properly for the GIG resource allocation
8430 capabilities to be provided. This enabler focuses on assured management that provides protection
8431 against attacks on the management and control system.

8432 (U//FOUO) These attacks could take the form of an attacker masquerading as a legitimate
8433 management node/user and then modifying a component through the management interface, for
8434 example, shutting it down remotely. To prevent this, there must be controlled management and
8435 control interfaces. Also, only authenticated components and users should be able to modify a
8436 component or the system.

8437 (U//FOUO) In addition, management and control communications should be protected from
8438 disclosure to unauthorized individuals. Disclosure of this type of information reveals substantial
8439 details about the network topology and capabilities and could provide an attacker a roadmap for
8440 a successful attack.

8441 (U//FOUO) The routing and allocation component is responsible for establishing and updating
8442 information routing paths as necessary. This includes the initial route establishment, monitoring
8443 of the actual flow of data, and the ongoing operation of the routing algorithm to modify paths for
8444 changing network conditions (e.g., congestion, failure, attack).

8445 (U//FOUO) IA policy-based routing is essential. The digital policy will stipulate the Quality of
8446 Protection required to assure the appropriate security protection is maintained while the data
8447 traverses the GIG. This differs from standard commercial networks that use metrics based
8448 primarily on cost in their routing algorithms. Routes are chosen to minimize the cost to the
8449 service provider and to the end customer of moving bits across the network. Other factors, such
8450 as latency or who owns the network or components, are less frequently used.

8451 (U//FOUO) The intent of policy-based routing is to guarantee a minimum level of service to
8452 users. This is generally measured in terms of bandwidth (i.e., they will be able to ship X bits per
8453 second), latency (i.e., data will take no more than Y seconds to transit from point A to point B),
8454 or similar measures. However, GIG routing will also have to factor in the security protection
8455 provided by the route and whether this protection is adequate for the QoP required by the data.

8456 (U//FOUO) For security reasons, a low-cost route through a network owned by a coalition
 8457 partner will often be rejected in favor of a higher cost route through a network owned by the U.S.
 8458 Government. To meet application requirements, a route with lower latency will sometimes be
 8459 selected over a lower-cost route with higher latency (e.g., a terrestrial network will be chosen
 8460 over a satellite connection).

8461 (U//FOUO) Routing decisions of this type constitute IA policy-based routing. The GIG must
 8462 support this feature. Further, the policy must be changeable for dynamic responses to changing
 8463 conditions, and the policy must be protected to ensure an adversary cannot substitute or modify a
 8464 policy to change operation of the GIG.

8465 (U//FOUO) QoS/CoS encompasses designing and implementing a network and its routing
 8466 infrastructure so that different types (classes) of data are treated differently. Typically, data
 8467 associated with applications that require real-time delivery with low latency and high likelihood
 8468 of error-free delivery can be assigned to a class that is forwarded or delivered faster than other
 8469 traffic, which can be delivered with classic Internet Protocol (IP) best efforts service. Examples
 8470 of data service applications which require low latency (near real-time), low error rates, and high
 8471 availability include streaming live video, and real-time collaboration tool services (combining
 8472 live interactive voice, video, and whiteboarding capabilities), in addition to high quality voice
 8473 transmissions over IP (VoIP) using high rate voice coders (32 kbps and above). An example of
 8474 an application that can be delivered with only classic IP best-effort service is e-mail, which can
 8475 be delivered whenever extra resources are available (typically, any time within the next 5 days).
 8476 An intermediate data service application that does not require low error rates, but only needs for
 8477 the low latency and high availability specifications to be met, is secure voice over the FNBBDT
 8478 protocol, whose 2.4 kbps Mixed Excitation Linear Prediction (MELP) vocoder can provide good
 8479 quality at up to 1% error rates. Thus, depending upon the specific application requirements, a
 8480 tailored QoS/CoS should be available that meets the desired performance specifications.

8481 (U//FOUO) In order to meet these requirements, the GIG must support certain QoS/CoS
 8482 mechanisms. However, there is often a clash between QoS/CoS and security requirements. For
 8483 example, QoS/CoS is often implemented by having the originator indicate to the infrastructure
 8484 the type of data being sent, so that the core routers can treat it appropriately. However, doing so
 8485 can result in a leak of potentially sensitive data around an encryption service and provide an
 8486 excellent covert channel for attackers to use as they wish. Thus, research must be done to
 8487 develop ways to have the GIG support QoS/CoS and at the same time meet its security
 8488 requirements.

8489 **2.5.3 (U) Technologies**

8490 (U//FOUO) The following technology areas support the Assured Resource Allocation Enabler:

- 8491 • (U//FOUO) IA Policy-Based Routing
- 8492 • (U//FOUO) Operational-Based Resource Allocation
- 8493 • (U//FOUO) Integrity of Network Fault Monitoring/Recovery
- 8494 • (U//FOUO) Integrity of Network Management & Control

8495 (U//FOUO) Since the last two technology areas (Integrity of Network Fault
 8496 Monitoring/Recovery and Integrity of Network Management & Control) are functionally similar
 8497 and likely to depend upon the same underlying infrastructures and secure signaling protocols,
 8498 they will be addressed within the same section.

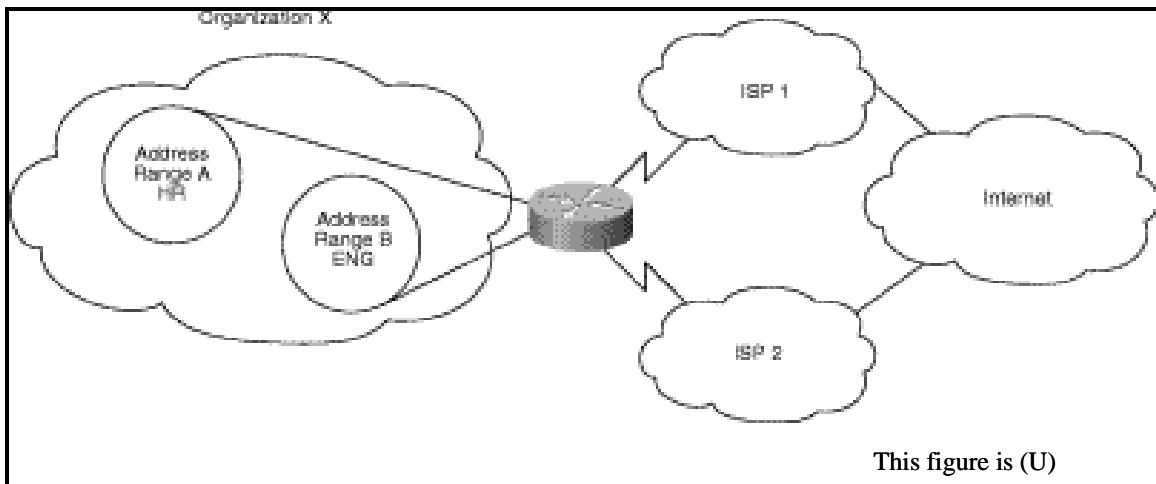
8499 **2.5.3.1 (U//FOUO) IA Policy-Based Routing**

8500 **2.5.3.1.1 (U) Technical Detail**

8501 (U//FOUO) Since varying levels of data sensitivity will be traversing the future GIG network
 8502 routing infrastructure—from unclassified up to and beyond Top Secret—the GIG would benefit
 8503 from a capability for Information Assurance policy-based routing. To a certain degree, Multi-
 8504 Protocol Label Switching (MPLS) can provide this attribute. However, MPLS is a static
 8505 technique that is not amenable to adaptation and dynamic operation in order to react to changing
 8506 network conditions. Should the network topology change or degrade due to router malfunctions
 8507 or adversarial denial of service attacks on specific routers, certain predetermined MPLS-Labeled
 8508 Switch Paths (LSPs) may become similarly broken (if they traverse the affected routers).

8509 (U//FOUO) Any IA policy-based routing scheme should ideally be adaptive and intelligent
 8510 enough to dynamically react to and compensate for network element outages. In general, IA
 8511 policy-based routing can be viewed as a subset of QoS-based routing, where the quality being
 8512 used as a metric happens to be that of information assurance.

8513 (U//FOUO) In very simplistic terms, the Figure 2.5-2 shows an elementary aspect of how IA
 8514 policy-based routing can be realized:



8515

8516

Figure 2.5-2: (U//FOUO) IA Policy-Based Routing

8517 (U//FOUO) As an example, suppose an organization wants to have a subset of its data traffic
8518 (traffic of its HR human relations group from address range A) go through Internet Service
8519 Provider (ISP) 1 and another subset of traffic (of its Engineering group, from address range B)
8520 go through ISP2. It uses different ISPs due to the different sensitivity levels of the two traffic
8521 flows and to the commensurate trust put in each of the ISPs. This is an example of Source-Based
8522 Transit Provider Selection—Internet service providers and other organizations can use policy-
8523 based routing to route traffic originating from different sets of users through different Internet
8524 connections across the policy routers.

8525 (U//FOUO) In general terms, Policy-Based Routing (PBR) provides a mechanism for expressing
8526 and implementing the forwarding or routing of data packets based on the policies defined by the
8527 network policy administrators. It provides a more flexible mechanism for routing packets
8528 through routers, complementing the existing mechanism provided by routing protocols. Routers
8529 forward packets to the destination addresses based on information from static routes or dynamic
8530 routing protocols—such as Routing Information Protocol (RIP), Open Shortest Path First
8531 (OSPF), or Enhanced Interior Gateway Routing Protocol (Enhanced IGRP[®]).

8532 (U//FOUO) Instead of routing by the destination address, policy-based routing allows network
8533 administrators to determine and implement routing policies to allow or deny paths based on the
8534 following:

- 8535 • (U) Identity of a particular end system
- 8536 • (U) Application
- 8537 • (U) Protocol
- 8538 • (U) Size of packets
- 8539 • (U) Security/classification level of traffic data packets
- 8540 • (U) Security/assurance of links/router nodes

8541 (U//FOUO) Policies can be defined as simply as "My network will not carry traffic from the
8542 engineering department." or as complex as "Traffic originating within my network with the
8543 following characteristics will take path A, while other traffic will take path B."

8544 (U//FOUO) One of the hallmarks or characteristics of a routing protocol, which enables taking
8545 into account the IA aspects of both the routing environment and the data packets that are being
8546 routed, is that the protocol must be flexible. This flexibility means different applications can use
8547 different paths between the same two points. A mechanism that provides for this capability
8548 would include the ability to modify at runtime the routing algorithms and property metrics used
8549 to generate forwarding tables. This would essentially result in routers having more than one
8550 forwarding table from which to make forwarding decisions, with packets being filtered in order
8551 to decide which forwarding table to employ. A routing protocol that utilizes this paradigm is the
8552 Flexible Intra-AS Routing Environment protocol (FIRE), developed under the auspices of
8553 Defense Advanced Research Projects Agency (DARPA) in 2000. FIRE is an interior gateway-
8554 routing protocol that allows traffic to be routed based on a set of routing algorithms rather than
8555 one algorithm—such as shortest path first.

8556 (U//FOUO) Today's routing protocols create a single forwarding table for routing decisions.
8557 These routing decisions are based on a single configured metric (generally determined by the
8558 specifier of the routing protocol, with some modest ability for operators to adjust the metrics).
8559 The least cost or shortest path based on that metric is usually what is chosen as the best route.

8560 (U//FOUO) The routing protocols are a closed system—access to routing information is
8561 permitted only for participating routers. This is not conducive to modern network architectures
8562 where adaptive or active networks provide applications greater freedom to specify the routing
8563 services needed. Current routing protocols do not permit applications to actively participate in
8564 the routing of their data and make it difficult for researchers and, more importantly, network
8565 operators to devise and deploy new metrics such as those they might require for QoS routing.

8566 (U//FOUO) FIRE addresses these problems by substantially enhancing the flexibility of a routing
8567 system within an autonomous system. FIRE is a link-state routing protocol, like Open Shortest
8568 Path First (OSPF), but rather than advertising a single metric as OSPF does, a FIRE router will
8569 advertise a series of property values such as security, cost, and bandwidth. Properties can be
8570 configured by an operator, or they can be a value determined at run time. Multiple forwarding
8571 tables can then be generated from these properties.

8572 (U//FOUO) In addition, FIRE may use path-generation algorithms other than SPF. For instance,
8573 a best path based on highest bandwidth is found by comparing the lowest bandwidth link of all
8574 possible paths. Of the lowest bandwidth links, whichever one has the highest bandwidth belongs
8575 to the highest bandwidth path. Similar computations would be done if security of specific links
8576 were the deciding factor, which would be the case in an IA policy-based routing environment.

8577 (U//FOUO) FIRE separates the routing algorithms from the environment within which these
8578 algorithms create forwarding tables. Consequently, the algorithms are treated as applets that are
8579 easily installed and replaced. In this respect, FIRE has an Active Networks component for
8580 expandability. In general, FIRE would employ a property repository or database for the
8581 links/nodes in a subject autonomous system (AS). It would use various routing algorithms,
8582 especially tailored to security attributes, to produce forwarding tables. Filters would then be
8583 applied to incoming packets to determine which table is appropriate to make a forwarding
8584 decision (where various criteria determine the path).

8585 (U//FOUO) A protocol such as FIRE can be implemented because many of the traditional
8586 baseline routing protocols have extension capabilities. For example, OSPF and IS-IS allow
8587 definition of new state advertisement messages. Thus, FIRE can be viewed as a evolution of the
8588 OSPF baseline capabilities.

8589 **2.5.3.1.2 (U) Usage Considerations**

8590 (U//FOUO) Certain portions of the GIG are likely to require baseline capabilities in support of
8591 IA policy-based routing early in the development of the GIG. High Assurance Internet Protocol
8592 Encryptor (HAIZE) program products should provide support for routing and QoS by the 2008
8593 timeframe. In addition, the JTRS Wideband Networking Waveform (WNW) program should
8594 provide for improved support for route selection, also in the 2008 timeframe.

8595 (U//FOUO) The application of IA policy-based routing techniques may be different depending
8596 upon whether the subject portion of the GIG network is wireless (as in JTRS) or wired (as in the
8597 GIG-BE core network). Wireless networks naturally are more topologically dynamic than wired
8598 networks and, as such, will require more agile IA policy-based routing implementations.

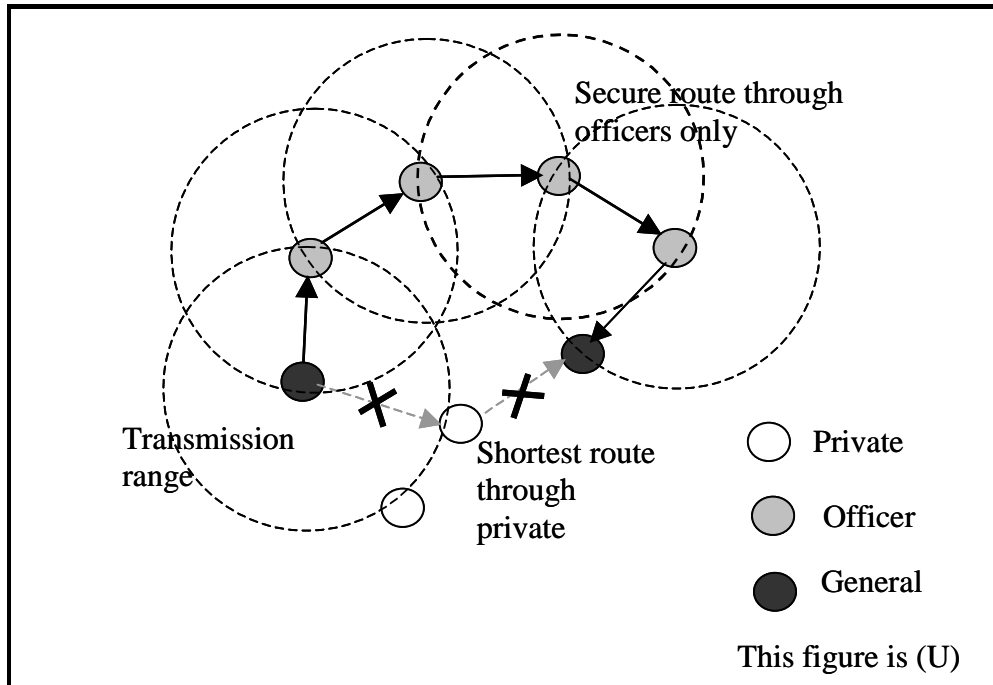
8599 (U) Wireless Applications:

8600 (U//FOUO) There has been some research in the area of IA policy-based routing in tactical
8601 wireless communications (as exemplified by mobile ad hoc networks or MANETs). One such
8602 study area is Security Aware Ad-hoc Routing (SAR—work done by Yi, Naldurg, and Kravets at
8603 the University of Illinois).

8604 (U//FOUO) The SAR protocol operates as follows: When a route of a particular security level is
8605 desired, a Route REQuest (RREQ) message is sent out. The RREQ header is encrypted with a
8606 group key (known only to those nodes in the network at the same trust level who can handle the
8607 desired data security level). The RREQ packet includes a field indicating the overall required
8608 route security level. Those intermediate nodes which can decrypt the RREQ then reply with a
8609 Route REPLY (RREP) message, indicating that they are capable of providing a security guarantee
8610 for the path through that node. Thus, eventually, a suitably secure end-to-end path is attained. An
8611 advantage of the SAR protocol is that it also provides security to the flow of routing protocol
8612 messages themselves.

8613 (U//FOUO) SAR can also be easily incorporated into generic ad hoc routing protocols. In
8614 general, SAR enables the automatic discovery of secure routes in a mobile ad hoc environment.
8615 Though not optimal, the routes that are discovered by SAR come with quality of protection
8616 guarantees. SAR's integrated security metrics allow applications to explicitly capture and
8617 enforce cooperative trust relationships. SAR can be built upon a base routing protocol, such as
8618 Ad hoc On demand Distance Vector (AODV), in which case it is known as SAODV.

8619 (U//FOUO) A notional scenario of how this SAR algorithm would operate in tactical
8620 applications (using JTRS and/or WIN-T technologies) is depicted in Figure 2.5-3:



8621

Figure 2.5-3: (U//FOUO) Security-Aware ad-hoc Routing (SAR) in Tactical Wireless Application

8622
8623

8624 (U//FOUO) In the above scenario, even though the second General is reachable most quickly by
8625 a path through a Private, the more secure path may be deemed to be only through those with
8626 officer rank. The SAR protocol implemented on a tactical Mobile Ad-hoc Networks (MANET)
8627 would allow the discovery of the desired path with an appropriate overall integrated end-to-end
8628 security metric. Future GIG wireless networks such as JTRS and WIN-T will require similar
8629 capabilities so that security attributes can be factored into routing decisions.

8630 2.5.3.1.2.1 (U) Implementation Issues

8631 (U//FOUO) Depending upon the restrictions which are to be imposed upon the core GIG router
8632 network, capabilities for full IA policy-based routing may be similarly restricted. For example, in
8633 the GIG-BE during its initial implementation phases, there will be no allowance for unprotected
8634 information such as QoS levels/specifications to pass from the Red side of the network to the
8635 Black side. This potentially limits routing options to static ones, other than routing around any
8636 immediately local router node failures that might occur within the Black Core.

8637 (U//FOUO) Fortunately, the HAIPE specification (as written) does make allowance for HAIPE
8638 encryption devices to be configured so as to bypass certain information fields (such as QoS bits,
8639 IPv6 flow labels, etc) around the encryption process from the Red to the Black domain.
8640 However, though many HAIPE encryptors will have this inherent capability, current IA policies
8641 tend to prohibit its use due to potential covert channel vulnerabilities. This restriction on an
8642 otherwise supported feature is both a GIG-wide implementation issue and a possible limitation to
8643 fully dynamic and responsive IA policy-based routing protocols.

8644 2.5.3.1.2.2 (U) Advantages

8645 (U//FOUO) Certainly one of the advantages of a dynamic and flexible IA policy-based routing
8646 protocol (as could be implemented within the constructs of the previously described FIRE
8647 routing environment) is that it can be automatically adaptive to changing network conditions and
8648 topologies. This is compared with static, MPLS path configurations which would not be as
8649 survivable or as forgiving to network topology modifications, especially those that would be seen
8650 in instances of denial of service attacks. This is due to the fact that MPLS is defined and set up
8651 beforehand, by the manual configuration of essentially hard-wired network paths for specific
8652 traffic classes. Indeed, the MPLS solution is merely an emulation of a static circuit-switched
8653 network solution within the environment of a potentially much more robust and adaptively
8654 dynamic packet-switched network fabric.

8655 2.5.3.1.2.3 (U) Risks/Threats/Attacks

8656 (U//FOUO) One of the risks or threats that any network faces is Denial of Service (DoS) or
8657 Distributed Denial of Service (DDoS) attacks from adversaries. A good defense of such attacks
8658 would include having a routing protocol or mechanism that is dynamic and proactive, in that it
8659 would be tied into and integrated with the CND Computer Network Defense/Situational
8660 Awareness infrastructure of the subject network. There has been some research into this idea,
8661 including some recent work at the University of Arizona (“Impact Analysis of Faults and Attacks
8662 in Large-Scale Networks,” by Hariri et al,
8663 [http://dslab.csie.ncu.edu.tw/92html/paper/pdf/Impact%20analysis%20of%20faults%20and%20attacks%20in%20lar
8664 ge-scale%20networks.pdf](http://dslab.csie.ncu.edu.tw/92html/paper/pdf/Impact%20analysis%20of%20faults%20and%20attacks%20in%20large-scale%20networks.pdf)).

8665 (U//FOUO) There is little value in an IA policy-based routing protocol if it only looks at the
8666 nominal or normal-condition status of link and nodal security attributes (along with the security
8667 characteristics of traffic data packets), without also having means to compensate for either
8668 already occurred or impending partial network router fabric failure due to aggressive denial of
8669 service attacks. The work at Arizona develops a series of needed metrics, including the
8670 Vulnerability Index (VI), Component Impact Factor (CIF), and System Impact Factor (SIF).
8671 Using these defined metrics, it then develops a dynamic proactive QoP routing protocol, capable
8672 of responding in real time to DDoS router attacks. The primary goal is to maintain availability so
8673 that essential network traffic is not denied paths to required end destinations. This is achieved
8674 through close observation and analysis of various router operational metrics, such as router
8675 buffer utilization, number of flows, and router request-processing rates.

8676 2.5.3.1.3 (U) Maturity

8677 (U//FOUO) Most current routing protocols are based on the policy of finding the shortest path
8678 (by application of cheapest cost algorithms) through the given network, for purposes of overall
8679 network efficiency and reduction of messaging latency. The extension of routing protocol
8680 algorithms to include the aspect or metric of path assurance/security is relatively recent and thus
8681 not nearly as mature. Some work in this area has been done for mobile ad hoc networks, due to
8682 the obvious potential vulnerabilities of wireless networks as compared with more secure wired
8683 network infrastructures. However, some of the ad hoc wireless research results have been
8684 extended to the wired domain due to the realization that IA policy-based routing can benefit all
8685 networks (wired or wireless).

8686 (U//FOUO) The various sub-technologies of the Integrity of Network
 8687 Management/Control/Monitoring/Recovery technology area can be generally assigned
 8688 Technology Readiness Level groups of Early, Emerging, and Mature.

- 8689 • (U//FOUO) Wireless domain flexible assured routing (SAR, etc.)—Early (TRLs 1 – 3)
- 8690 • (U//FOUO) Security-driven routing protocols (FIRE, etc.)—Early to low Emerging
 8691 (TRLs 1 - 4)
- 8692 • (U//FOUO) Basic MPLS-based (fixed) security routing—Mature (TRLs 7 – 9).

8693 2.5.3.1.4 (U) Standards

8694 (U) Draft U.S. Government Protection Profile on “Switches and Routers” ([http://niap.nist.gov/cc-](http://niap.nist.gov/cc-scheme/index.html)
 8695 [scheme/index.html](http://niap.nist.gov/cc-scheme/index.html)).

8696 (U) Routing Policy Specification Language (RPSL).

8697 (U//FOUO) There are not many current standards specific to the area of policy-based routing, let
 8698 alone standards that are devoted to the more specific and delineated area of IA policy-based
 8699 routing. One standard under development within the IETF is the Routing Policy Specification
 8700 Language (RPSL). The text of the RPSL specification, as described in IETF RFC 2622, can be
 8701 found at <http://www.ietf.org/rfc/rfc2622.txt> (C. Alaettinoglu et. al., 1999).

8702 (U//FOUO) RPSL is merely a language for expressing and conveying routing policies. The
 8703 language defines a maintainer class (mntner class) object, which is the entity that controls or
 8704 maintains the objects stored in a database expressed by RPSL. Requests from maintainers can be
 8705 authenticated with various techniques as defined by the *auth* attribute of the maintainer object.
 8706 The exact protocols used to communicate RPSL objects is beyond the scope of RPSL as
 8707 described by RFC 2622, but it is envisioned that several techniques may be used, ranging from
 8708 interactive query/update protocols to store and forward protocols (similar to email). Regardless
 8709 of which protocols are used, it is expected that appropriate security techniques, such as IPsec,
 8710 TLS, or PGP/MIME would be used.

8711 (U) Routing Policy Specification Language next generation (RPSLNg):

8712 (U//FOUO) The Internet Engineering Steering Group (IESG) of the IETF has recently initiated
 8713 work on RPSLNg (Routing Policy Specification Language next generation) to add a new set of
 8714 extensions to RPSL, thus enabling the language to implement routing policies for the IPv6 and
 8715 multicast address families that are currently used in the Internet. Since the GIG will operate
 8716 within IPv6 environments (by mandate as of 2008), it is advantageous that RPSL is undergoing
 8717 this timely updating process. The text of the RPSLNg draft can be found at
 8718 <http://www.radb.net/rpslng.txt> (L. Blunk et.al., 2004).

8719 (U//FOUO) While the extensions described by RPSLng introduce no additional security threats,
8720 it should be noted that the original RFC 2622 describing the RPSL standard included several
8721 weak or vulnerable authentication mechanisms. For example, among RPSL-defined mechanisms
8722 and constructs, the “MAIL-FROM” scheme can be easily defeated by source email address
8723 spoofing. Secondly, the “CRYPT-PW” scheme is subject to dictionary attacks and password
8724 sniffing if RPSL objects are submitted by unencrypted channels, such as email. And finally, the
8725 “NONE” mechanism option offers no protection for objects.

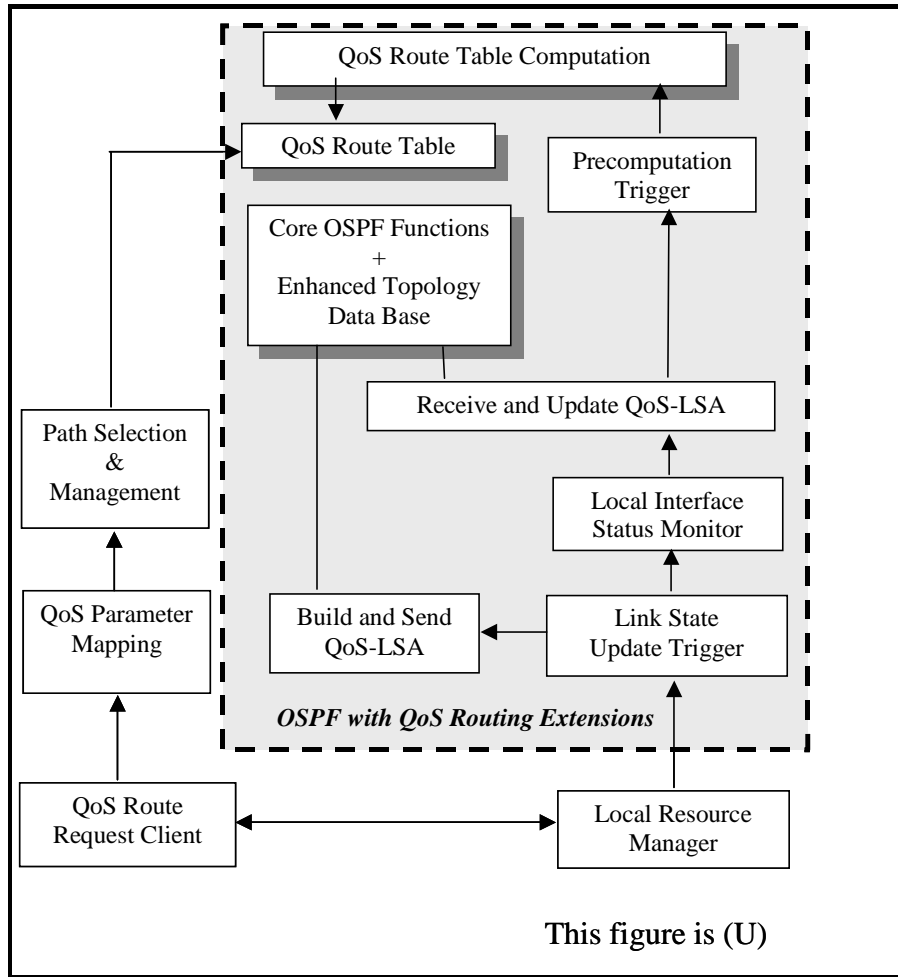
8726 (U) Related QoS Routing Standards:

8727 (U//FOUO) There are currently several existing IETF RFCs devoted to the description of QoS-
8728 based routing mechanisms. IA policy-based routing is merely a specialized subset of QoS-based
8729 routing, where the governing QoS is transport security. RFC 2386 “A Framework for QoS-based
8730 Routing in the Internet” (Crawley et al, 1998) describes a framework for extending the current
8731 Internet routing model of intra and interdomain routing to support QoS.

8732 (U//FOUO) Another relevant IETF standard document is RFC 2676 “QoS Routing Mechanisms
8733 and OSPF Extensions” (Apostolopoulos et al, 1999). The GIG is expected to use routing
8734 protocols such as OSPF or the related Intermediate System to Intermediate System (IS-IS)
8735 protocol.

8736 (U//FOUO) As can be deduced from its name, OSPF normally in its default mode would simply
8737 opt to select the shortest path route through a network, without taking into consideration any
8738 other metrics such as the security or IA attributes of encountered nodes and links. Fortunately,
8739 both OSPF and IS-IS allow modifications of their default operation by the use of extensions,
8740 such as the provision to enable definition of new LSA link state advertisement messages (for
8741 updating routing tables). As noted in an earlier section, an example of a routing implementation
8742 environment that could allow for IA policy-based routing is BBN’s FIRE (Flexible Intra-AS
8743 Routing Environment) which takes advantage of the extension provisions within OSPF to enable
8744 dynamic and adaptive routing capabilities. Figure 2.5-4 shows how QoS policy-based routing
8745 can be implemented within the OSPF core environment:

8746



8747

8748

Figure 2.5-4: (U) OSPF Implemented With (QoS) IA Policy-Based Routing Extensions

8749

(U//FOUO) Observation of the above figure shows that such an adaptive and dynamic routing protocol can manage path selection based not only upon metrics such as perceived nominal security of any given links or nodes, but can also factor in such qualities as availability or congestion (based upon the residual bandwidth of network links). Future users of the GIG will not only demand routing based upon assurance but also upon optimized availability.

8753

8754

2.5.3.1.5 (U) Cost/Limitations

8755

(U//FOUO) Any IA policy-based routing methodology will have inherent costs and limitations when implemented in the GIG. Certainly, the installed router software would be more expensive in order to support all of the options presented to a router in so far as assurance-evaluated selectable network paths. Other implied costs would reside in a potential multiplicity of forwarding tables within each router, rather than a single forwarding table per router. Each router would select the relevant forwarding table based upon the IA policy required by the data packet in transit—with more sensitive data choosing the table that yields higher resultant end-to-end assurance levels.

8762

8763 **2.5.3.1.6 (U) Dependencies**

8764 (U//FOUO) One dependency of the potential evolution and development of a robust, enhanced
 8765 IA policy-based routing protocol is that it be built upon the foundation of an extensible baseline
 8766 protocol. One such protocol which allows for extensibility is the OSPF protocol, which is related
 8767 to the IS-IS protocol, both of which the GIG is likely to use.

8768 (U//FOUO) Another dependency of the development of a robust IA policy-based routing
 8769 protocol for the future GIG network is that of the required foundation of a GIG standard for
 8770 Quality of Protection (QoP). Given a QoP definition, whereby specific data entities or packets
 8771 are to be tagged with information (metadata) that marks the packets for handling and routing
 8772 tailored to the sensitivity of the data contents, an IA policy-based routing protocol can then use
 8773 the QoP metadata to optimize the overall network security of the various traffic flow elements.

8774 **2.5.3.1.7 (U) Alternatives**

8775 (U//FOUO) As has been already noted, an alternative to a fully implemented IA policy-based
 8776 routing protocol is the use of static MPLS routing. Although this is not as effective and flexible
 8777 (or fine-grained) a solution as a dynamic policy-based one, it is better than having no provision
 8778 at all for the protection of sensitive data classes.

8779 **2.5.3.1.8 (U) Complementary Techniques**

8780 (U//FOUO) In addition to being seen as an alternative solution, there is no reason why MPLS
 8781 cannot be used in conjunction with (or within the context of) a larger framework of an IA policy-
 8782 based routing methodology.

8783 **2.5.3.1.9 (U) References**

8784 (U) "Routing Policy Specification Language (RPSL)," by Alaettinoglu et al,
 8785 <http://www.ietf.org/rfc/rfc2622.txt> , or <http://www.faqs.org/rfcs/rfc2622.html> , 1999.

8786 (U) "QoS Routing Mechanisms and OSPF Extensions," by Apostolopoulos et al,
 8787 <http://www.ietf.org/rfc/rfc2676.txt> , or <http://www.faqs.org/rfcs/rfc2676.html> , 1999.

8788 (U) "A Framework for QoS-based Routing in the Internet," by Crawley et al,
 8789 <http://www.ietf.org/rfc/rfc2386.txt> , or <http://www.faqs.org/rfcs/rfc2386.html> , 1998.

8790 (U) "Integrating Quality of Protection into Ad Hoc Routing Protocols," by Yi, Naldurg, &
 8791 Kravets, <http://mobius.cs.uiuc.edu/publications/sci02.pdf> , 2002.

8792 (U) "Security-Aware Ad-Hoc Routing For Wireless Networks," by Yi, Naldurg, & Kravets,
 8793 <http://www.csee.umbc.edu/courses/graduate/CMSC628/spring2002/ppt/poonam.ppt> , 2002 (talk at UMBC).

8794 (U) "Security Aware Ad-hoc Routing (SAR)," by Yi, Naldurg, & Kravets,
 8795 <http://www.cs.fsu.edu/~yasinsac/group/slides/carter5.pdf> , 2002.

8796 (U) "Routing with Confidence: Supporting Discretionary Routing Requirements in Policy Based
 8797 Networks," by Kapadia, Naldurg, & Campbell,
 8798 http://choices.cs.uiuc.edu/~akapadia/papers/sec_routing.pdf.

- 8799 (U) "FIRE: Flexible Intra-AS Routing Environment," by Partridge et al,
8800 <http://www.cs.ndsu.nodak.edu/~yizhang/Presentations/FIRE.ppt>, 2000.
- 8801 (U) <http://www.ir.bbn.com/projects/fire/architecture/architecture.html>
- 8802 (U) <http://www.ir.bbn.com/documents/techmemos/>
- 8803 (U) <http://www.ir.bbn.com/documents/techmemos/TM1245.pdf>
- 8804 (U) <http://www.ir.bbn.com/documents/techmemos/TM1244.pdf>
- 8805 (U) <http://www.ir.bbn.com/documents/techmemos/TM1265.pdf>
- 8806 (U) <http://www.ir.bbn.com/documents/articles/FIREjsac-3-01.pdf>
- 8807 (U) "Impact Analysis of Faults and Attacks in Large-Scale Networks," by Hariri et al,
8808 <http://dslab.csie.ncu.edu.tw/92html/paper/pdf/Impact%20analysis%20of%20faults%20and%20attacks%20in%20large-scale%20networks.pdf> (also see
8809 <http://dslab.csie.ncu.edu.tw/92html/paper/ppt/Impact%20analysis%20of%20faults%20and%20attacks%20in%20large-scale%20networks.ppt>), 2003.
- 8810
8811
- 8812 (U) "Intelligent Active Routing For Supporting QoS Demands," by Tasir et al,
8813 http://www.cntr.salford.ac.uk/telecoms_research/research_activity/abdul_rahman_mohamed_tasir/pgnet2002.htm ,
8814 2002.
- 8815 (U) "Security Cost Routing," by Eli Winjum,
8816 http://web.unik.no/users/mobkom/presentasjoner/Eli_SecurityCost_250803.ppt, 2003.
- 8817 (U) "Policy-Based Routing," http://www.cisco.com/warp/public/cc/pd/iosw/tech/policy_wp.htm, 2002.
- 8818 (U) "Configuring Policy-Based Routing,"
8819 http://www.cisco.com/univercd/cc/td/doc/product/lan/cat4000/12_1_19/config/pbroute.pdf
- 8820 (U) "QoS/Policy/Constraint-Based Routing," by Wei Sun, http://www.cse.ohio-state.edu/~jain/cis788-99/ftp/qos_routing/, 2000.
8821

8822 **2.5.3.2 (U//FOUO) Operational-Based Resource Allocation**

8823 **2.5.3.2.1 (U) Technical Detail**

8824 (U//FOUO) The technical area of operational-based resource allocation is predominantly one of
8825 the pure research realm, with fairly few examples of fielded systems that employ this capability
8826 (in an automated sense). There are very few commercial efforts in this area—with the common
8827 response to the assurance of adequate resources being that of initial over-provisioning of
8828 computation and/or transport assets, so that all potential customers will be adequately served.
8829 However, in the defense/military field, there has been some research efforts dedicated, most
8830 recently sponsored by a variety of DARPA programs. Future customers of the GIG will expect
8831 and demand certain levels of network transport, database access, and computational services.
8832 Each customer will have a dynamic/changeable user profile that will describe the privileges that
8833 are given to that customer. The future GIG Privilege Management Infrastructure (PMI) will
8834 necessarily work very closely with a resource allocation system tailored to customer-centric
8835 operational demands.

8836 (U//FOUO) A traditional example of operational-based resource allocation is the Multi Level
8837 Precedence and Preemption (MLPP) mechanism that has been used for years in the context of
8838 the DoD voice telecommunications system. It is desirable to have the MLPP paradigm, which is
8839 nominally only for voice communications control/allocation purposes, extended to the packet-
8840 switching and enterprise services-based GIG environment. This extends the MLPP paradigm to
8841 coverage of far more system functionality.

8842 (U//FOUO) As is implied in the MLPP acronym, this paradigm allows for an a priori allocation
8843 through the precedence route of the (limited) resource of a telecommunications link to a
8844 customer whose rank or privileges exceed those of other potential service customers. Precedence
8845 decisions are made before the link is fully established. Thus this is a somewhat static and non-
8846 adaptive process. In addition, however, the preemption process of MLPP enables an already
8847 allocated resource of a telecommunications link to be taken away from the initial customer, or
8848 preempted, and to be given to a customer with higher privileges and immediate requirements.
8849 Hence, the preemption process is more dynamic and agile than precedence. Both of these
8850 capabilities—precedence and preemption—would be useful within the context of the GIG in
8851 terms of allocating data transport, data storage, computation, and enterprise service capabilities.

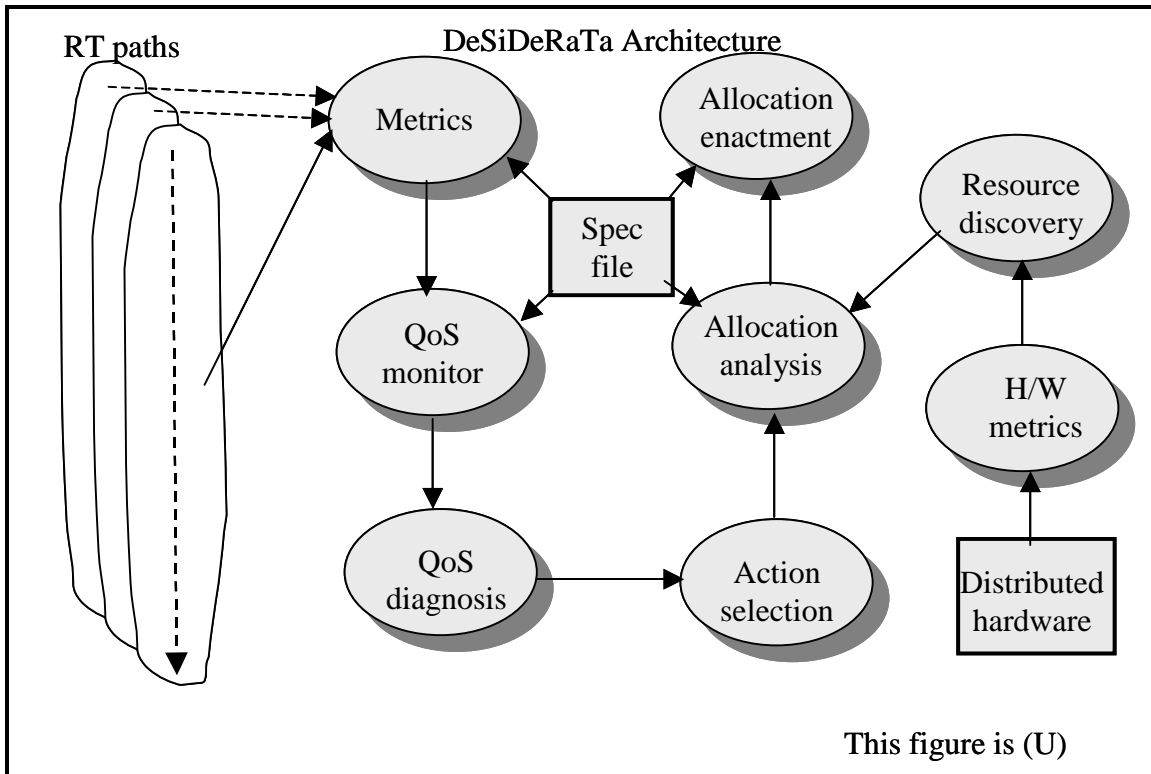
8852 (U//FOUO) Current DoD Information Resources Management (IRM) is fairly inconsistent in its
8853 mechanisms for the allocation or re-allocation of communications and other services. Each
8854 separate network service or layer has its own mechanism: circuit switched (voice) uses the
8855 MLPP protocol, satellite circuits are allocated according to the priorities defined in Chairman of
8856 the Joint Chiefs of Staff Instruction (CJCSI) 6250.01, and the current common user data
8857 networks have little or no priority-based assignment capabilities.

8858 (U//FOUO) Rather than have a similarly disjoint solution in the future GIG environment—where
 8859 the resources of Transformational Satellite (TSAT), GIG-BE routers, JTRS nodes/links, and
 8860 NCES services will all be interacting with each other—a common and integrated resource
 8861 allocation solution is required. This solution will be required to span across the boundaries of the
 8862 various GIG systems. Until now, however, many DoD and Intelligence Community (IC)
 8863 networks have avoided the implementation of automatic allocation and re-allocation mechanisms
 8864 by implementing community of interest (COI) networks that are small enough to allow for
 8865 effective manual arbitration. The efficiency-driven use of a common GIG infrastructure will
 8866 force the DoD and IC to address this issue of enterprise-wide automatic resource allocation.

8867 (U//FOUO) Several programs under the auspices of DARPA have studied the area of dynamic
 8868 and operational-based resource allocation over five years. These include the following:

- 8869 • (U) QUORUM Project
- 8870 • (U) Agile Information Control Environment (AICE) Program
- 8871 • (U) Battlefield Awareness and Data Dissemination (BADD) Program.

8872 (U//FOUO) One methodology for the automation of dynamic operational-based resource
 8873 allocation, developed under DARPA auspices, is that of Dynamic Scalable Dependable Real-
 8874 Time Systems (DeSiDeRaTa). Figure 2.5-5 shows the basic ideas behind DeSiDeRaTa:



8875

8876 **Figure 2.5-5: (U) DeSiDeRaTa Architecture for Operational-Based Resource**
 8877 **Allocation**(U//FOUO) From the above figure, it can be seen that DeSiDeRaTa is divided into
 8878 three vertical groups of functions. The left group deals with QoS measurement and analysis, the

8879 central deals with allocation analysis and actions, and the right deals with resource analysis (or
 8880 resource situational awareness). This model could be applicable to the GIG where resource
 8881 allocation and re-allocation decisions would be made by adjudication of resource requests
 8882 against the applicable customer privilege profiles (managed within the GIG's PMI. Overall
 8883 control of the DeSiDeRaTa mechanisms would be managed by using a nominal specification
 8884 file, which would consist of the desired (and allowed) customer QoS and the translatable and
 8885 relevant, required resources. These resources would consist of GIG transport, computation, data
 8886 storage, and enterprise services access.

8887 (U//FOUO) The next generation of computing and networking is leaning heavily towards the
 8888 paradigm of distributed computing and networking. As distributed real-time systems—such as
 8889 those that will be found within the GIG—become increasingly popular, there is an increasing
 8890 need of technology that can handle the resource allocation problems presented by distributed
 8891 computing and networking. It is from this basis that DeSiDeRaTa Resource Management has
 8892 found a grasp in the research community. The DeSiDeRaTa project has the goal of producing a
 8893 Resource Manager that provides the following features:

- 8894 • (U) Specification Language for Hardware Systems, including computing resources and
 8895 networks
- 8896 • (U) Specification Language for Software Systems, including methods of specifying QoS
 8897 requirements such as real-time, scalability, and dependability QoS constraints
- 8898 • (U) QoS Management for instrumentation, assessment, prediction, negotiation, and
 8899 allocation of resources for real-time systems.

8900 (U//FOUO) DeSiDeRaTa technology will employ the dynamic path paradigm, which is a
 8901 convenient abstraction for expressing end-to-end QoS objectives of systems and for performing
 8902 QoS management. The DeSiDeRaTa project provides an adaptive resource management
 8903 approach that is appropriate for systems (such as the GIG) that expect to experience large
 8904 variations in workload. A distributed collection of computing resources is managed by
 8905 continuously computing and assessing QoS metrics and resource utilization metrics that are
 8906 determined a posteriori.

8907 (U//FOUO) The DeSiDeRaTa project's specification language describes the environment-
 8908 dependent (and operationally-driven) features of dynamic real-time systems. Also provided is an
 8909 abstract model that is constructed (statically) from the specifications, and is augmented
 8910 dynamically with the state of operational environment-dependent features. The model is being
 8911 used to develop algorithms for QoS monitoring, QoS diagnosis, and resource allocation analysis.
 8912 Experimental results show the effectiveness of the approach for specification of real-time QoS,
 8913 detection and diagnosis of QoS failures, and restoration of acceptable QoS by re-allocation of
 8914 distributed computer and network resources.

8915 (U//FOUO) Future GIG customers who are given temporary privileges for access to certain GIG
 8916 resources due to operational exigencies would benefit from the dynamic real-time checking that
 8917 this protocol potentially affords, so that quality of service levels would be maintained and
 8918 adjusted to satisfy operational requirements. In this sense, DeSiDeRaTa can be viewed as being
 8919 simultaneously Proactive and Reactive in its methodology for the allocation and re-allocation of

8920 resources (see <http://www.atl.external.lmco.com/overview/papers/1117.pdf>). The DARPA Quorum project
8921 analyzed the applicability of DeSiDeRaTa for proactive and reactive resource allocation.

8922 (U//FOUO) Besides the potentially relevant DeSiDeRaTa protocol, there have been other
8923 projects done under DARPA auspices in the area of dynamic requirements-driven resource
8924 allocation. However, as already noted, this is a relatively new field with few fully mature
8925 implementations. Most instantiations of resource allocation to date are manually configured, as
8926 opposed to policy-driven automatic implementations, which is the desired end-state of the GIG.

8927 (U//FOUO) Research done during 2001 by a team at Colorado State University (CSU) (under the
8928 auspices and sponsorship of the DARPA AICE and BADD programs) concentrated on
8929 operational-based dynamic resource allocation for classes of prioritized session and data requests
8930 in preemptive heterogeneous networks (<http://www.engr.colostate.edu/~echong/pubs/conf/pdpta01.pdf>).
8931 The GIG can be viewed as such a large heterogeneous network, and certain classes of data within
8932 it will be prioritized—based upon the operation of the GIG standard for precedence and
8933 preemption.

8934 (U//FOUO) The work done at CSU could potentially be relevant to the internal specifics of this
8935 GIG foundational standard. CSU defined network transactions (or communication requests) as
8936 one of either two types: Data or Session (session being defined as bandwidth access over a
8937 certain timespan). Furthermore, network requests are assigned to a Class and a Priority level
8938 (within the class). For purposes of precedence analysis, the request ‘worth’ is computed as a
8939 weighted priority that is a function of the situation (war time, peace time, etc.). The CSU
8940 methodology then devises a scheduling heuristic that reorders customer service requests by
8941 maximizing the sum of weighted priorities of the highest class and then works down the class
8942 hierarchy.

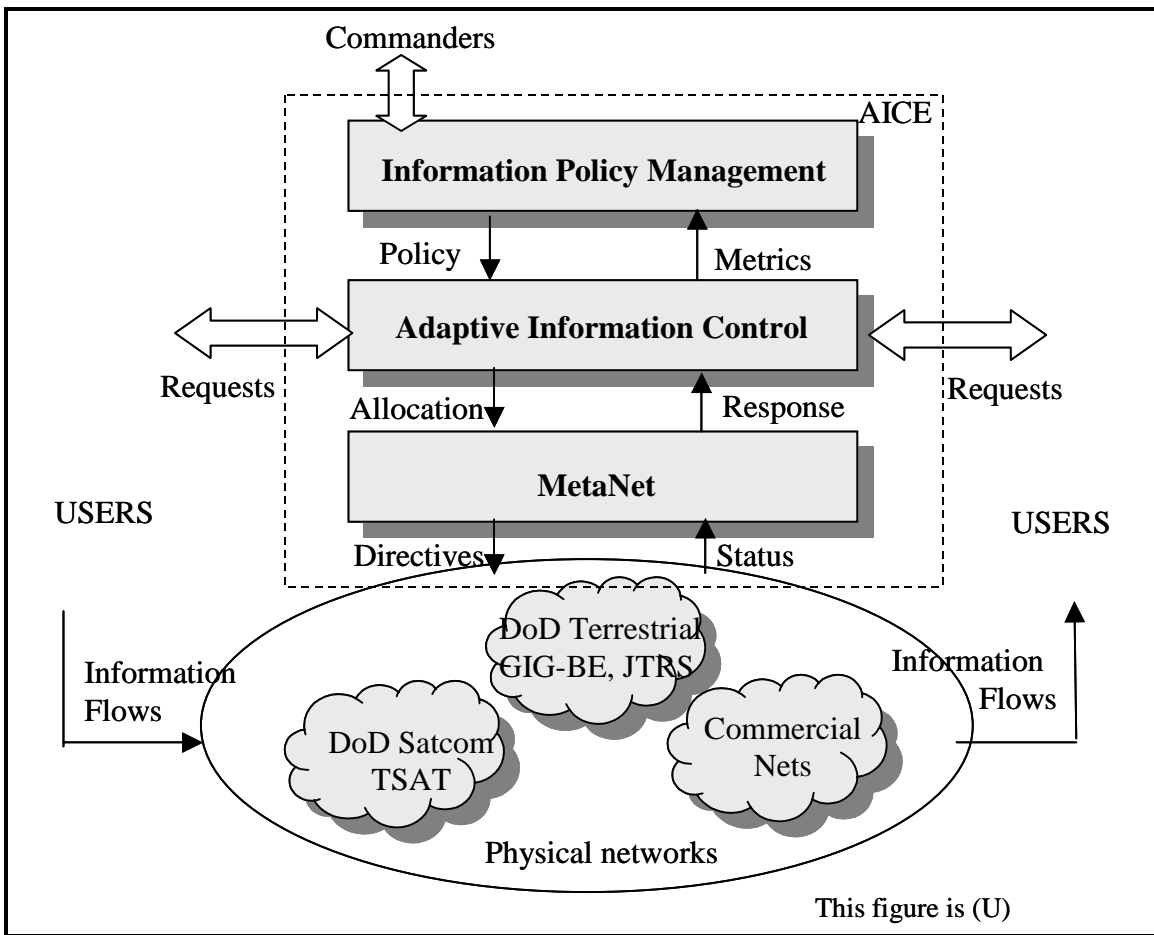
8943 (U//FOUO) An important issue raised by the CSU researchers is the need for a post-preemption
8944 scheduler so that any transaction request which is preempted is not lost but is rationally
8945 rescheduled in a logically prioritized sense. This rescheduling mechanism can be relevant to the
8946 development of a GIG Precedence and Preemption standard.

8947 **2.5.3.2.2 (U) Usage Considerations**

8948 **2.5.3.2.2.1 (U) Implementation Issues**

8949 (U//FOUO) Any operational-based resource allocation system in the future GIG infrastructure
8950 must have the capability for dynamic modification of customer privilege profiles within the PMI.
8951 Future military commanders will not always require privileges that consistently and persistently
8952 put them at the head of the line when it comes to getting requested resources before others. Only
8953 at times when unique and specific operations are underway will it be necessary for participating
8954 individuals to have their privilege status elevated. When the subject operation is completed,
8955 participating GIG customers shall in all likelihood have their privilege status relegated and re-
8956 baselined back to their normal levels. Since this implies a dynamic privilege management
8957 infrastructure, it is important that the PMI be robust and secure, and that the necessary policy
8958 adjudication entities be present to authorize any temporary modifications or elevations of
8959 privileges.

8960 (U//FOUO) Operational-based resource allocation can be viewed as an exercise in adaptive
 8961 information control across a distributed landscape. As such, the DARPA AICE Program has
 8962 conducted a number of relevant studies. The GIG landscape consists of a number of
 8963 interconnected disparate networks (TSAT, terrestrial wired GIG-BE, wireless JTRS, and WIN-T,
 8964 etc.) over which resources will be allocated. The transport networks themselves are also
 8965 allocated resources (for the transport of user communications, sensor data, database query
 8966 results, and enterprise services, etc). There is a need for the study of the global/overall control
 8967 and allocation of these disparate network resources so that the integrated services provided to the
 8968 subject customer base are maximized and optimized. The following figure (based upon work for
 8969 DARPA by S. Jones and I. Wang of Johns Hopkins Applied Physics Lab)
 8970 (<http://www.engr.colostate.edu/~echong/pubs/conf/00985799.pdf>) illustrates a partitioning of the required
 8971 signaling to achieve joint resource allocation across disparate networks.



8972
 8973 **Figure 2.5-6: (U) Joint Resource Allocation Across GIG Networks(U//FOUO)** The above
 8974 illustration separates the operational-based resource allocation functions into four different but
 8975 interconnected layers:

- 8976 • (U//FOUO) Physical Network Layer
- 8977 • (U//FOUO) This layer consists of the independent tactical (JTRS), terrestrial (GIG-BE),

8978 satellite (TSAT), and wireless, and commercial (Internet) networks that will together
8979 comprise the end-to-end user GIG fabric. They will provide packet routing services and
8980 unique QoS capabilities.

8981 • (U//FOUO) MetaNet Layer

8982 • (U//FOUO) This layer is the system that facilitates the QoS-based routing through the
8983 integrated collection of networks. Four aspects of the MetaNet layer include: inserting
8984 QoS-like capabilities into existing tactical networks to enable dynamic (and operational-
8985 based) re-allocation of network resources, negotiating service requests as an intermediary
8986 between the user and individual networks, providing end-to-end QoS solutions within a
8987 time-constraint, and maintaining negotiated end-to-end QoS by dynamically re-routing or
8988 renegotiating service.

8989 • (U//FOUO) Adaptive Information Control (AIC) Layer

8990 (U//FOUO) This layer provides global content-aware dynamic information flow control,
8991 employing the services of the MetaNet layer to do so. AIC layer features include:
8992 partitioning of information flows among available logical channels, globally optimizing
8993 allocation to achieve military users' information flow priorities (precedence and
8994 preemption), and re-allocating resources when necessary due to network QoS
8995 degradation.

8996 • (U//FOUO) Information Policy Management (IPM) Layer

8997 (U//FOUO) This layer has three primary functions: providing users the capability to
8998 visualize the impacts of their information control policies, relating information policy
8999 management to military operations, and aiding in the synthesis of effective information
9000 control policies. It is from this layer that relevant and temporary modifications to GIG
9001 customer privilege profiles will be made (within the GIG privilege management
9002 infrastructure), whereby users are allocated the resources sufficient to successfully
9003 conduct military operations.

9004 (U//FOUO) The ultimate objective of the DARPA AICE program is to realize information
9005 control and resource allocation in a way that is faster, more efficient, and more precise than is
9006 currently realized—(and in an automatic fashion as opposed to manually).

9007 **2.5.3.2.2.2 (U) Advantages**

9008 (U//FOUO) Certainly one of the advantages of a well constructed operational-based resource
9009 allocation system within the future GIG environment is that the overall operation and congestion
9010 of the GIG can be optimized to service the most important needs at any given time and in any
9011 given theatre of operations. This implies that an alternative over-provisioning solution need not
9012 be required. This thus yields savings in the fielded network infrastructure (transport, storage, and
9013 computational) equipment. This can especially be true in the case of wireless segments of the
9014 GIG (such as mobile ad hoc networks within the JTRS and WIN-T networks), where the network
9015 'mesh' is topologically dynamic and potentially sparse.

9016 **2.5.3.2.2.3 (U) Risks/Threats/Attacks**

9017 (U//FOUO) Since resource allocation will be based upon the privileges of requesting GIG
 9018 customers, it is essential that both the specific resource requests and the customer privileges be
 9019 secure, trusted, and not subject to tampering or modification by adversaries.

9020 **2.5.3.2.3 (U) Maturity**

9021 (U//FOUO) Maturity of operational-based resource allocation technology is fairly low level,
 9022 especially resource allocation that is automatic as opposed to manual (and human operator
 9023 intensive). Resource allocation traditionally has been limited to the scope of small geographic
 9024 areas, as opposed to the world-wide reach of the GIG network.

9025 (U//FOUO) Future warfighters in ‘hot-spots’ who require and deserve unique privileges to
 9026 resource access will need to have special consideration in the allocation of GIG transport,
 9027 computation, storage, and database access capabilities. All of these GIG resources will be
 9028 distributed. It is the coordination and timely delivery of these resource capabilities that will need
 9029 research and study before this technology area can be said to be in any stage of maturity.

9030 (U//FOUO) The various sub-technologies of the Integrity of Network
 9031 Management/Control/Monitoring/Recovery technology area can be generally assigned
 9032 Technology Readiness Level groups of Early, Emerging, and Mature.

- 9033 • (U//FOUO) MLPP in Defense Information System Network (DISN) voice
 9034 telecommunications—Mature (TRLs 7 – 9)
- 9035 • (U//FOUO) Adaptive/Dynamic distributed resource allocation (like DeSiDeRaTa)—
 9036 Emerging (TRLs 4- 6)
- 9037 • (U//FOUO) Operational resource allocation tied to secured/adaptive PMI—Early (TRLs
 9038 1 – 3).

9039 **2.5.3.2.4 (U) Standards**

9040 (U//FOUO) Since there are few commercial or industrial efforts in this technology area (such as
 9041 by the IETF), there are not any real standards relevant to operational-based resource allocation.
 9042 As the technology is developed, standards (within the GIG community) should be
 9043 commensurately developed, so as to assure that all participants within the GIG would be using
 9044 the same protocols. As a corollary to the implementation of standards for the actual mechanics of
 9045 resource allocation or re-allocation, a parallel, supporting GIG standard will be needed for
 9046 Precedence and Preemption (as a subset of the overall GIG privilege management infrastructure).

9047 **2.5.3.2.5 (U) Cost/Limitations**

9048 (U//FOUO) Any operational-based resource allocation system for the future GIG will have to be
9049 cognizant of the possibility that instantaneous local demands in any potential future theatre of
9050 operations may exceed the possible delivery capacity (in terms of transport throughput, etc.). As
9051 such, methodologies and technologies that are developed must have built-in mechanisms for
9052 intelligent resource trimming and notification and also for intelligent policy-driven arbitration in
9053 cases of simultaneous demands by disparate customers for the access to the same common
9054 resources.

9055 **2.5.3.2.6 (U) Dependencies**

9056 (U//FOUO) Successful implementation of operation-based resource allocation within the GIG
9057 will be dependent upon a number of other developments, especially that of the development of a
9058 GIG-wide standard for priority and preemption capability. This standard would be required to
9059 clearly define the priority status levels and classes in which all GIG customers will be assignable,
9060 in addition to the mechanisms for modifications (and reversions to nominal levels) of user
9061 privileges.

9062 **2.5.3.2.7 (U) Alternatives**

9063 (U//FOUO) An alternative to the necessity of developing an operational-based resource
9064 allocation capability within the future GIG is merely to have over-provisioning of required assets
9065 (computational, storage, and transport) across the future GIG. While this may be a potential
9066 solution when viewing across the GIG as a whole, it will probably not succeed when specific
9067 local assets are exceeded by temporarily excessive local demands (as could be the case in a
9068 theatre of war). As such, the GIG will require an allocation system that will provide priority
9069 claims on assets for those with the highest adjudicated, locally-valid privileges.

9070 **2.5.3.2.8 (U) Complementary Techniques**

9071 (U//FOUO) Complementary, or subsidiary, techniques for the operational-based allocation of
9072 resources include those of the traditional MLPP techniques currently used in the circuit-switched
9073 DoD DISN voice network. There are current efforts under pursuit by DISA to fully implement
9074 MLPP capabilities within the future GIG-BE router mesh fabric, where voice will no longer be
9075 circuit-switched but will instead be VoIP. This IP version of MLPP capabilities should be
9076 viewed as part of the future integrated overall resource allocation/re-allocation infrastructure of
9077 the GIG—all driven by an underlying dynamic and secure privilege management infrastructure.

9078 **2.5.3.2.9 (U) References**

9079 (U) "Proactive and Reactive Resource Allocation," by Cross & Lardieri,
9080 <http://www.atl.external.lmco.com/overview/papers/1117.pdf>, 2000.

9081 (U) "A MetaNet Architecture For End-To-End Quality Of Service (QoS) Over Disparate
9082 Networks," by Jones & Wang, <http://www.engr.colostate.edu/~echong/pubs/conf/00985799.pdf>, 2001.

9083 (U) "A QoS Performance Measure Framework for Distributed Heterogeneous Networks," by
9084 Kim et al, http://www.cs.nps.navy.mil/people/faculty/irvine/publications/2000/empdp00_QoSMSHN.pdf, 2000.

- 9085 (U) “Applying Intent-Sensitive Policy To Automated Resource Allocation: Command,
9086 Communication and Most Importantly, Control,” by Funk et al,
9087 http://www.sift.info/English/publications/HICS_AICE-00.pdf, 2000.
- 9088 (U) “Dynamic Resource Allocation for Classes of Prioritized Session and Data Requests in
9089 Preemptive Heterogeneous Networks,” by Naik et al,
9090 <http://www.engr.colostate.edu/~echong/pubs/conf/pdpta01.pdf>, 2001.
- 9091 (U) “Information Value based Information Resource Management of the Defense Information
9092 Systems Network,” by Devens & Pitt,
9093 http://www.argreenhouse.com/society/TacCom/papers99/08_5.pdf , 1999.
- 9094 (U) “Secure-RM: Security and Resource Management for Dynamic Real-Time Systems,” bu
9095 Tjaden et al (Ohio University), <http://jarok.cs.ohiou.edu/papers/scc2000.pdf>, 2000.

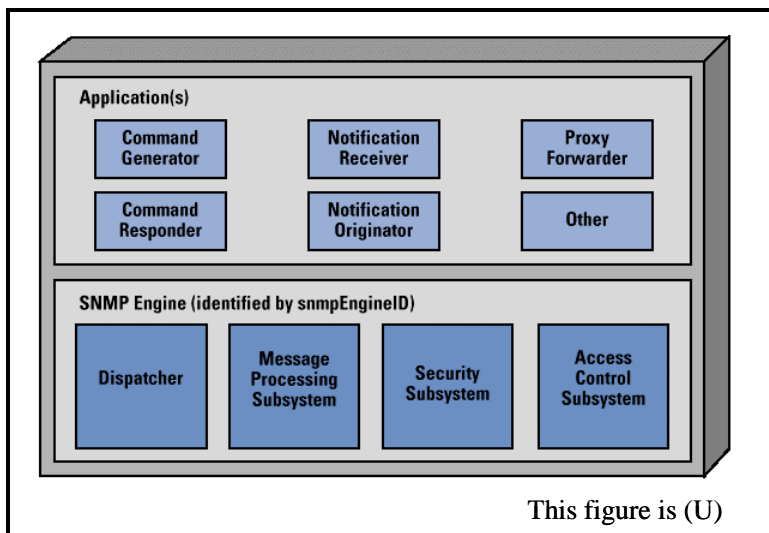
9096 **2.5.3.3 (U//FOUO) Integrity of Network Fault Monitoring/Recovery and Integrity of**
 9097 **Network Management & Control**

9098 **2.5.3.3.1 (U) Technical Detail**

9099 (U//FOUO) One of the most important IA aspects of the future GIG will be that of securely
 9100 managing and controlling—both locally and remotely—the various and many network elements.
 9101 On top of this, should portions of the GIG infrastructure become impaired due to an external
 9102 attack, component failure, or malfunction, there would be a need for robust and distributed,
 9103 network fault monitoring and recovery. Since all of these functions rely upon a well-defined set
 9104 of common sensing (incoming) and command (outgoing) message constructs, a standardized
 9105 protocol such as the IETF Simple Network Management Protocol (SNMP) would provide the
 9106 required capabilities. SNMP is a default standard methodology for network management and has
 9107 survived numerous competing standard entrants.

9108 (U//FOUO) What is really required for successful network management, control, and monitoring,
 9109 is an entire framework built around three foundation components: a data definition language as
 9110 defined by an Internet-standard Structure of Management Information (SMI), a set of definitions
 9111 of management information as delineated by an Internet-standard Management Information Base
 9112 (MIB), and a common protocol definition (SNMP). The MIB database resides generally at the
 9113 managed client/agent, and its variables define the scope, range and limitations of control features
 9114 which may be executed. The SNMP protocol is used to convey information and commands
 9115 between network managers and managed objects (or agents).

9116 (U//FOUO) There are four basic operations or commands that may be executed within the SNMP
 9117 protocol. These are **Get**, **GetNext**, **Set**, and **Trap**. The first three commands are initiated by the
 9118 manager, and they act upon MIB variables at the client agent of interest. The **Trap** message is
 9119 initiated by a client agent when an error or fault occurs, and it is used in order to notify the
 9120 central manager that something unexpected has gone wrong. The basic elements of SNMP
 9121 operation are shown in Figure 2.5-7.

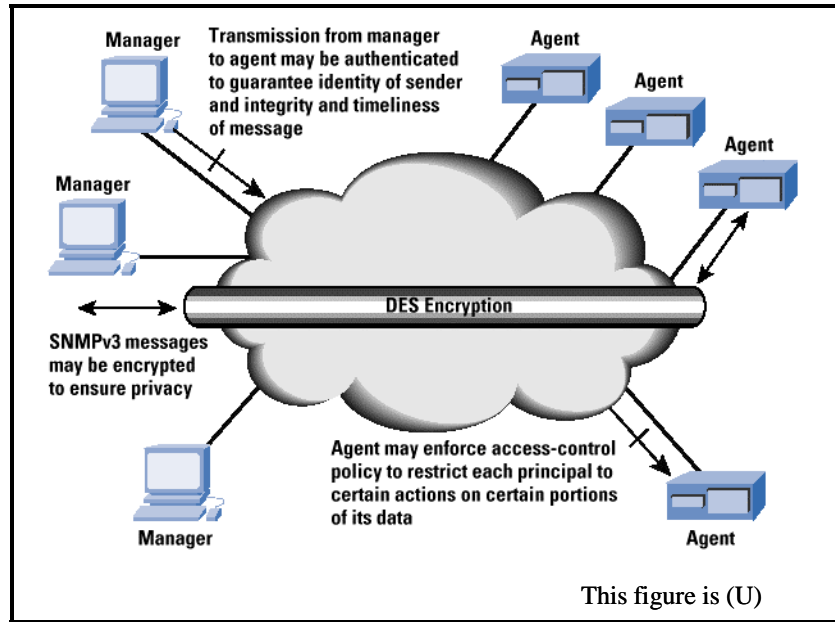


9122

9123 **Figure 2.5-7: (U) Basic Elements of SNMP Operation**

9124 (U//FOUO) Note the security-relevant components of the Security Subsystem and Access
 9125 Control Subsystem in the above figure. It is these component elements that have evolved
 9126 considerably during the evolution of SNMP through its SNMPv1, SNMPv2, and SNMPv3
 9127 versions.

9128 (U//FOUO) The first two versions of SNMP had no real security functionality. Security was
 9129 primarily introduced in the SNMPv3 implementation. Both authentication and privacy
 9130 capabilities were introduced by SNMPv3, as shown in Figure 2.5-8.

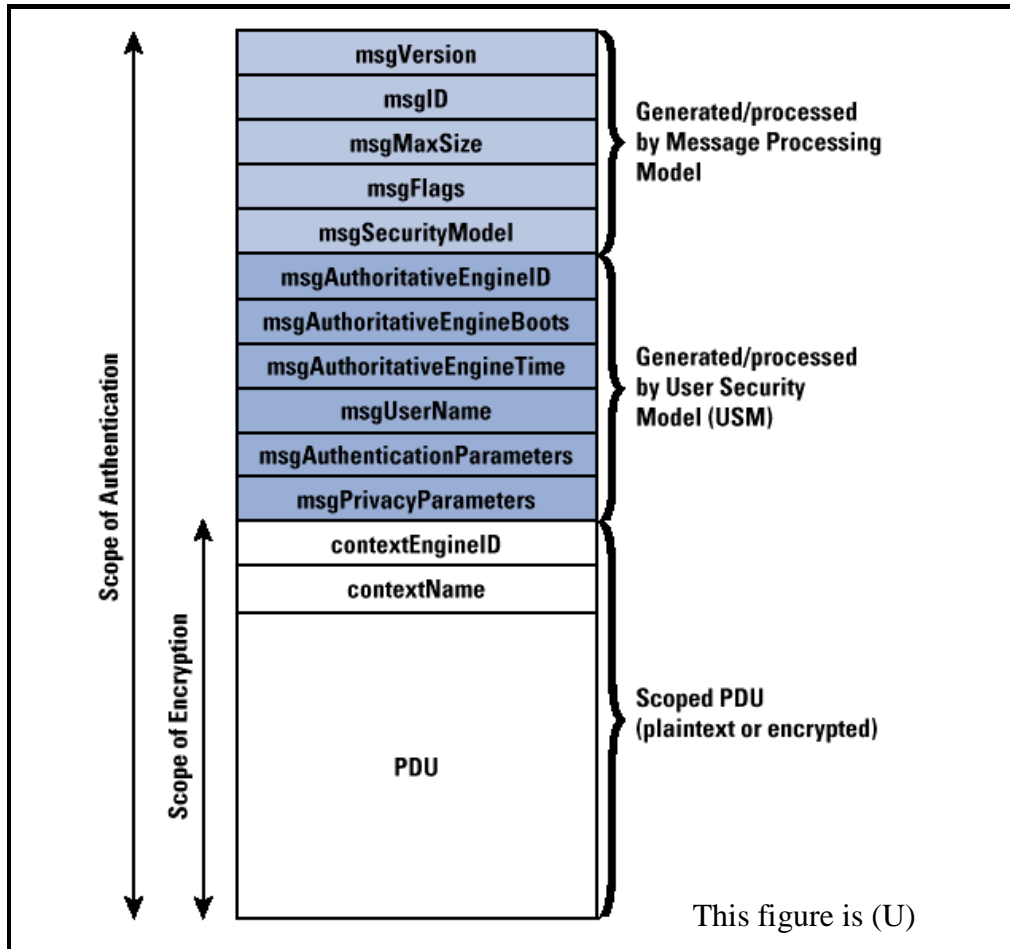


9131

Figure 2.5-8: (U) SNMPv3 Security Capabilities

9132

9133 (U//FOUO) The User Security Model (USM) describes operations of the security functions
 9134 within SNMPv3. In the basic model, cryptographic keys are assumed to be symmetric or private
 9135 keys. Authentication is accomplished by using Hashed Message Authentication Code-Message
 9136 Digest Algorithm 5 (HMAC-MD5) or alternatively HMAC- Secure Hash Algorithm 1 (SHA-1).
 9137 Encryption or message privacy is accomplished using the Digital Encryption Standard (DES) in
 9138 the Cipher Block Chaining (CBC) mode. The SNMPv3 message format, as implemented with
 9139 USM, along with the application scopes of authentication and encryption, is shown in Figure
 9140 2.5-9.



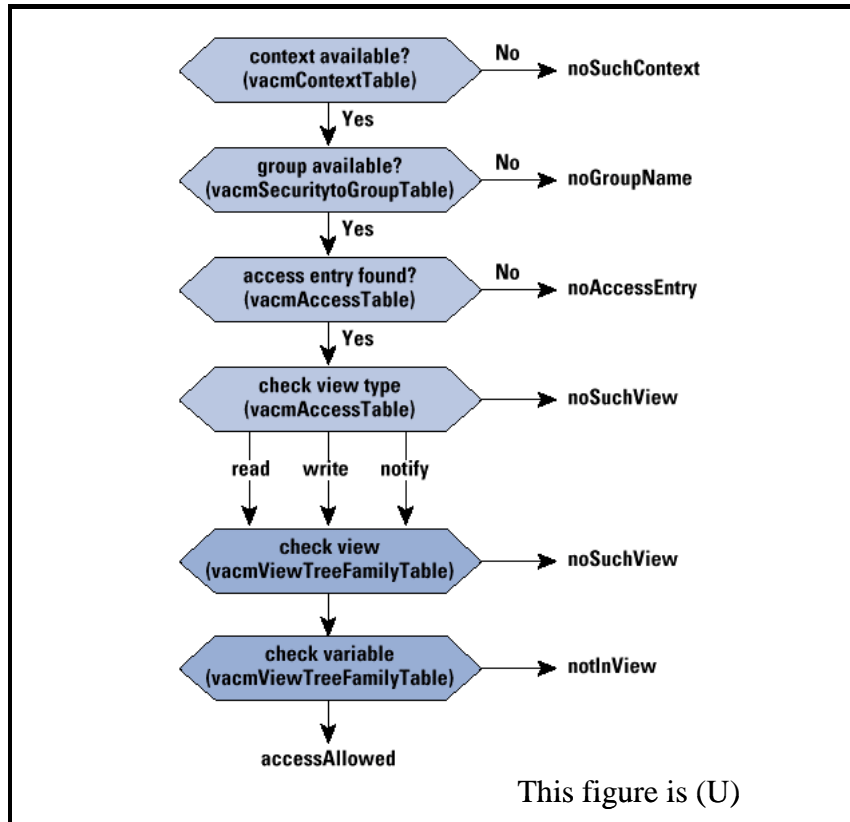
9141

9142

Figure 2.5-9: (U) SNMPv3 Message Format & Security Components

9143 (U//FOUO) The MD5 message digest algorithm (or optional SHA1) indirectly provides for data
 9144 origin authentication, and it directly defends against data modification attacks.

9145 (U//FOUO) One of the important security features of SNMPv3 is the View-based Access
 9146 Control Model (VACM) that it employs. VACM determines whether access to a managed object
 9147 or agent should be allowed. To do this, VACM makes use of an MIB that defines the access
 9148 control policy for the subject agent—thus enabling remote configuration capabilities. VACM is
 9149 flexible in that its logic provides for access to be decided by a series of relevant questions
 9150 concerning the access request: "Who ? + Where ? + How ? + Why ? + What ? + Which ?". Based
 9151 on the answers to these questions, in conjunction with the contents of policy-based access tables,
 9152 access is either allowed or disallowed. Figure 2.5-10 shows the access control logic employed by
 9153 VACM:



9154

9155

Figure 2.5-10: (U) SNMPv3 View-based Access Control Model (VACM) Logic

9156

9157

9158

(U//FOUO) The addition of the VACM capability within SNMPv3 should enable future GIG applications to conduct policy-based and fully access-controlled remote and distributed network management and monitoring functions. As such, it is a powerful construct.

9159

2.5.3.3.2 (U) Usage Considerations

9160

9161

9162

9163

9164

9165

9166

9167

9168

9169

(U//FOUO) Some components of the future GIG have already proposed using SNMPv3 in order to enable the IA of management control and monitoring functions. For example, the TSAT program proposes the use of SNMPv3 for network monitoring (as mentioned on slide 23 of the briefing “TCM IA Architecture Overview”, 30 June 2004, by NSA’s IAD TSAT IA Integrated Program Team [IPT]). Network management and control of the TSAT network will be required to be at the MAC I level (Mission Assurance Category), the highest of the three defined MAC levels (for a system requiring high integrity and high availability). Similarly, TSAT network management and control will require a confidentiality level of “Sensitive” (or the medium of 3 possible confidentiality levels). The SNMPv3 protocol is deemed adequate in satisfying network monitoring requirements.

9170

9171

9172

9173

9174

9175

(U//FOUO) Many experts (for example, computer science professor Dr. Richard Stanley of Worcester Polytechnic University) have said that SNMPv3 is the “clear long-term choice” for secure network management. Unfortunately, SNMPv3 is still a work-in-progress even within the IETF standardization process. SNMPv1 still holds 95% of the commercial market, with even the intermediate SNMPv2 not yet widely deployed. Upgrading to SNMPv3 is difficult and costly. However, it promises to provide for many GIG network management security requirements.

9176 (U//FOUO) There are actually disadvantages of SNMPv2 versus SNMPv1 in that version 2
9177 makes matters potentially worse from a security viewpoint. This is due to the fact that while both
9178 versions do not have security written into them, SNMPv2 introduces the concept of distributed
9179 management, which opens the management process to additional potential vulnerabilities. GIG
9180 implementations should only consider SNMPv3-compliant or equivalent systems.

9181 **2.5.3.3.2.1 (U) Implementation Issues**

9182 (U//FOUO) The addition of the security functions and their associated mechanisms to the
9183 SNMPv3 standard version has resulted in the fact that SNMPv3 is more compute-intensive than
9184 the earlier versions. This has led some in the research community to compare the efficiency of
9185 full SNMPv3 implementations with SNMPv2 running over Transport Layer Security TLS/TCP
9186 secure connections or, alternatively, over IPsec. These two options effectively separate out
9187 encryption protection from within the SNMP standard itself and bring it to a wrapping transport
9188 function. This only addresses the encryption/privacy aspects of SNMPv3 and does not
9189 implement any of the VACM access control functionality, which SNMPv3 provides us.

9190 (U//FOUO) The Office of Naval Research (ONR) funded Midkiff and Hia of Virginia Tech in
9191 2001 to look at the IPsec security option to SNMPv3 encryption across backbone networks. They
9192 showed that SNMPv3 could consume as much as 24% more network capacity than SNMPv2
9193 over IPsec. The disadvantage of the IPsec method is that it does not provide for fine-grained
9194 access control. The advantage shown by the SNMPv2-over-IPsec solution was shown to
9195 deteriorate as the size of the application-layer payload increased. Much of the inefficiency of the
9196 SNMPv3 solution is due to the Basic Encoding Rules (BER) used to encode SNMP application
9197 data.

9198 (U//FOUO) The NSA/ Laboratory for Telecommunications Science (LTS) funded Du and
9199 Shayman of the University of Maryland to investigate the performance comparisons of SNMPv1
9200 over a TLS/TCP base with full SNMPv3 security. One issue of SNMPv1/TLS/TCP is the
9201 nontrivial overhead associated with setting up a session, as compared against SNMPv3 over
9202 UDP (sessionless). However, for a long session the costs of setting up the session are amortized
9203 over a large number of messages, and therefore the overhead per message decreases. The final
9204 experimental results showed that SNMPv3 (with full USM security functionality) session times
9205 were much larger (from 163% up to 433% of) than the comparable SNMPv1/TLS/TCP session
9206 times. Thus, for situations of lower data rate environments, this aspect of SNMPv3 may perhaps
9207 need to be considered.

9208 **2.5.3.3.2.2 (U) Advantages**

9209 (U//FOUO) SNMPv3 builds upon the general overall advantages of SNMP in that it solves many
9210 of the security problems of the earlier SNMPv1 and SNMPv2 versions. One of the basic appeals
9211 of SNMP has been its simplicity, because SNMP provides a bare-bones set of functions and thus
9212 is easy to implement, install, and use. If applied sensibly it won't place an undue burden on the
9213 network. Moreover, due to its simplicity, interoperability can be achieved in a relatively
9214 straightforward manner—SNMP modules from various vendors can be made to work together
9215 with minimal effort.

9216 **2.5.3.3.2.3 (U) Risks/Threats/Attacks**

9217 (U//FOUO) The messages which will be needed to provide for assured GIG network
 9218 management control and monitoring will be subject to a variety of potential adversarial threats or
 9219 attacks. Hence, the security constructs of an enabling protocol such as SNMPv3 must be
 9220 adequate to protect against these potential malicious actions. The SNMPv3 protocol's User-
 9221 based Security Model (USM) improved upon the earlier versions of SNMP so as to protect
 9222 against the following four threats:

- 9223 • (U//FOUO) Modification of Information—Attempt by an unauthorized entity to alter an
 9224 SNMP message in-transit (issued on behalf of an authorized principal)
- 9225 • (U//FOUO) Masquerade—Attempt by an unauthorized entity to perform an operation by
 9226 assuming the identity of an authorized entity
- 9227 • (U//FOUO) Message Stream Modification—Delay or replay of messages to an extent
 9228 greater than can occur in natural conditions of network service
- 9229 • (U//FOUO) Disclosure—Attempt by an unauthorized entity to see the contents of SNMP
 9230 message/data exchanges

9231 (U//FOUO) SNMPv2 has been shown to be vulnerable to replay attacks (and resultant message
 9232 stream modification) due to the possibility of clock time drift between network manager and
 9233 remote agent. This is solved by SNMPv3—it supposedly would also be ameliorated by the
 9234 adoption of a truly secure and robust Network Time Protocol (NTP) across the GIG. Though the
 9235 SNMPv3 protocol provides for protection against the above 4 threats, it was decided during the
 9236 development of SNMPv3 to not provide for defense against the following two threats:

- 9237 • (U//FOUO) Traffic Analysis (TA)
- 9238 • (U//FOUO) Denial of Service (DoS)

9239 (U//FOUO) At the time of SNMPv3 definition it was deemed that these two threats either
 9240 required defenses that were nearly impossible to achieve or were not as significant as the others.

9241 (U//FOUO) While subject to various malicious threats or attacks—or merely to innocent network
 9242 component failures—the GIG infrastructure will be subject to the potential risk that network
 9243 management and control messages will be unable to reach their desired destinations. This is
 9244 especially true in the case of an Internet IP protocol such as SNMP that provides all its signaling
 9245 in-band (IB) on the same IP routing infrastructure upon which normal traffic travels. For
 9246 example, in order to conduct management and control of a particular network router, the paths to
 9247 that router will be necessarily operational or else the control function will not be possible. This
 9248 quandary has led some industry proponents to propose that perhaps backup out-of-band (OOB),
 9249 perhaps dial-up, control paths be maintained to at least the critical network elements.

9250 (U//FOUO) While perhaps not as essential in the area of everyday network management and
9251 control, these OOB techniques may become most valuable during times of network fault
9252 monitoring and recovery. The possible segregation of SNMP traffic onto a physically separate
9253 management network would potentially require an entirely parallel architecture redesign (e.g.,
9254 VLANs, routing, BGP/OSPF domains, new IP addresses, for configuring managers and remote
9255 agents). It would also require a transition plan to ensure continued management during
9256 migration. Carriers and other network service providers have used OOB for years because their
9257 businesses depend on the continuous availability of their network infrastructure. The degree to
9258 which the GIG should adopt this philosophy is yet to be determined.

9259 (U//FOUO) The vulnerabilities of the original SNMPv1 protocol, with virtually no provision for
9260 security functionality, are such that many organizations purposely limit the use and application
9261 of SNMP. The newer SNMPv3, when and if fully deployed as specified, should go far to remove
9262 these concerns.

9263 (U//FOUO) Meanwhile, however, the vulnerabilities of deployed SNMP systems continue to be
9264 exposed. An example of this is the work done in Finland during 2002 by the Oulu University
9265 Secure Programming Group (OUSPG). In this study more than four dozen vulnerabilities to
9266 SNMPv1 were demonstrated on commercial system implementations (e.g., Cisco). Examples of
9267 vulnerabilities include cases of seemingly innocent poor error handling when the SNMP
9268 primitive messages of **Get**, **Set**, or **Trap** were transmitted with invalid encodings or illegal
9269 internal values. The results of these simple non-malicious mistakes could lead to network
9270 elements crashing, locking up, rebooting, overwriting critical data values, or even enabling
9271 unauthorized access. Other uncovered vulnerabilities of SNMPv1 include the possibility of
9272 bounce attacks whereby malicious attackers could bounce their attacks off a trusted node.

9273 (U//FOUO) Risks and vulnerabilities of SNMP have been well-documented by the US.
9274 Computer Emergency Readiness Team (CERT) and the CERT Coordination Center at Carnegie
9275 Mellon University's Software Engineering Institute (CMU SEI). Useful documentation available
9276 from them includes an SNMP Vulnerability FAQ (frequently asked questions—at
9277 http://www.cert.org/tech_tips/snmp_faq.html), which accompanies the illustrative "CERT Advisory CA-
9278 2002-03" on SNMP vulnerabilities (<http://www.cert.org/advisories/CA-2002-03.html>). CERT
9279 acknowledges the foundation work of OUSPG in the uncovering of many examples of
9280 vulnerable commercial SNMP deployed implementations.

9281 (U//FOUO) Finally, even with the assumption of a finalized and robustly secure SNMPv3
9282 standard, if the Request For Comments (RFC) are not fully and carefully implemented by the
9283 various vendors, there may still be residual vulnerabilities such as those to buffer overflow
9284 exploits. However, this can also be true of other network management standards.

9285 **2.5.3.3.3 (U) Maturity**

9286 (U//FOUO) SNMP has a fairly long history since its debut in the late 1980s. As such, it has had
 9287 time to mature, certainly as proved by the development of the later versions through SNMPv3 in
 9288 the late 1990s. This maturing process has been beneficial by solving many of the security issues
 9289 left unresolved by the first version. The marketplace is populated by many implementations of
 9290 SNMPv1, with marketplace adoption of SNMPv2 and SNMPv3 lagging due to business inertia
 9291 reasons, while the standards process proceeds to improve upon SNMPv3. With the
 9292 vulnerabilities of SNMPv1 having become well known, pressure will mount for retrofit with
 9293 SNMPv3-compliant network management systems.

9294 (U//FOUO) There are many commercial implementations of SNMP. These include systems built
 9295 by HP, IBM, Novell, Sun, Microsoft, Compaq, Empire Technologies, Gordian, and SimpleSoft.
 9296 In addition, there are at least 18 commercial or academic implementations of the more advanced
 9297 SNMPv3, including those by AdventNet, BMC Software, Cisco, Halcyon, IBM, Multiport
 9298 Corporation, SimpleSoft, SNMP Research, UC Davis, and University of Quebec. Thus,
 9299 considering both the ongoing commercial work and the standards work within the IETF,
 9300 SNMPv3 should continue to evolve and improve.

9301 (U//FOUO) The various sub-technologies of the Integrity of Network
 9302 Management/Control/Monitoring/Recovery technology area can be generally assigned
 9303 Technology Readiness Level groups of Early, Emerging, and Mature.

- 9304 • (U//FOUO) Basic SNMPv3 implementations—Mature (7 – 9)
- 9305 • (U//FOUO) Key management enhancements for SNMPv3—Early (1 – 3)
- 9306 • (U//FOUO) Efficient SNMPv3 with security by IPsec or SSL/TLS (rather than native
 9307 SNMPv3 encryption)—Emerging (4 – 6)

9308 **2.5.3.3.4 (U) Standards**

9309 (U) U.S. Government Protection Profile on “Network Management” ([http://niap.nist.gov/cc-](http://niap.nist.gov/cc-scheme/pp/index.html)
 9310 [scheme/pp/index.html](http://niap.nist.gov/cc-scheme/pp/index.html)).

9311 (U//FOUO) As far as the definition of the SNMP protocols is concerned, there are a number of
 9312 IETF RFCs that explain the relevant security-enabling aspects of SNMPv3. These include the
 9313 following:

- 9314 • (U) RFC 3414, “User-based Security Model (USM) for version 3 of the Simple Network
 9315 Management Protocol (SNMPv3)”
- 9316 • (U) RFC 3415, “View-based Access Control Model (VACM) for the Simple Network
 9317 Management Protocol (SNMP)”

9318 (U//FOUO) In addition to the IETF arena, a number of different standards groups have been
9319 developing competing or alternate frameworks for network management control and monitoring.
9320 These include the International Standards Organization (ISO) and the Open Software Foundation
9321 (OSF). However, these alternate approaches have for various reasons not yet been successful in
9322 the commercial marketplace. As such, reviewing these will be delayed until the upcoming
9323 “Alternatives” section.

9324 **2.5.3.3.5 (U) Cost/Limitations**

9325 (U//FOUO) There are several limitations currently to the broad implementation of SNMPv3.
9326 One of these is in the area of key management. The official SNMPv3 standard generically calls
9327 for initial OOB distribution of secret keys among manager and agent elements, without
9328 specifying a technique. Thus there is no accepted, standardized initial key distribution
9329 mechanism—only an experimental Diffie-Hellman approach. There is also no integration with
9330 centralized key management and authorization, such as RADIUS. One approach exists for
9331 Kerberos, but that has been labeled experimental, and Kerberos does not seem to be in wide
9332 commercial use. Finally, there has been only some initial work to standardize the widely desired
9333 Advanced Encryption Standard AES support (as described in a 2002 IETF draft,
9334 <http://www.snmp.com/eso/draft-blumenthal-aes-usm-04.txt>).

9335 (U//FOUO) On a more positive note, however, very recent work during 2003-2004 has been
9336 undertaken on the SBSM (Session-Based Security Model) for SNMPv3. This would employ
9337 public key based I&A (between manager and agent elements), using the SIGMA key exchange
9338 protocol (using Diffie-Hellman). SIGMA has several advantages including its simplicity and
9339 efficiency, that it has had extensive review and is used for IKE (Internet Key Exchange), and it
9340 protects the identity of the session initiator.

9341 (U//FOUO) The SBSM protocol itself has a number of advantageous characteristics and features:

- 9342 • (U//FOUO) It uses existing security infrastructures for identity authentication
- 9343 • (U//FOUO) Both ends of message exchanges are authenticated
- 9344 • (U//FOUO) The responder agent reveals its identity and authenticates before the initiator
9345 manager
- 9346 • (U//FOUO) Separate mechanisms are used for identity authentication as compared with
9347 message authentication or encryption,
- 9348 • (U//FOUO) It has limited life time keys for encryption

9349 (U//FOUO) The consequences of these features are that there is a low cost to creating new
9350 identities, changing, or deleting their authentication credentials. Also, saved encrypted messages
9351 can not be decrypted after an identity key is compromised. However, SBSM is a work in
9352 progress, and overall SNMPv3 key management will require some maturation and standards
9353 adoption.

9354 **2.5.3.3.6 (U) Dependencies**

9355 (U//FOUO) The future success of SNMP-based network management systems will depend upon
9356 their full adoption of SNMPv3 security functionality and the full marketplace adoption of
9357 SNMPv3 implementations in lieu of SNMPv1 systems. Finally, use of SNMP within the GIG
9358 will depend upon the demonstrated robust and correct implementations by vendors of SNMPv3,
9359 so as to minimize any residual vulnerabilities.

9360 **2.5.3.3.7 (U) Alternatives**

9361 (U//FOUO) Since SNMPv1 was originally proposed in the late 1980s, several competing
9362 standards alternatives have been proposed. Nonetheless, for a variety of reasons, SNMP
9363 continues to evolve and improve, whereas the competitors have often come and gone. SNMP-
9364 based network management and its associated security mechanisms continue to grow, expand its
9365 scope, and mature. Four examples of competing alternative architecture schemes are described
9366 below:

- 9367 • (U) Common Management Information Protocol (CMIP) comes out of the ISO. The main
9368 problem with this protocol is that it is overly complex and perhaps overly ambitious. Due
9369 to this complexity, it can require up to 10 times the CPU power of an SNMP
9370 implementation. Few commercial implementations of CMIP can be found. CMIP
9371 originally was supposed to be the protocol that replaced SNMP in the late 1980s. It was
9372 funded by governments and large corporations, which caused many to believe that it
9373 would inevitably succeed. However, implementation problems delayed its widespread
9374 availability. CMIP had security advantages over SNMPv1 in that it included
9375 authentication and security log mechanisms. However, SNMPv3 solves the security holes
9376 of SNMP. Because of the fact that SNMP came out first and was much simpler to
9377 implement, CMIP is used today primarily in management of public telephone networks,
9378 while SNMP dominates most of the network management field.
- 9379 • (U) Distributed Management Environment (DME) comes from the Open Software
9380 Foundation (OSF), originating during the 1991 timeframe (from proposals submitted by
9381 25 organizations, including IBM, HP, Tivoli Systems, etc.). It is a framework meant for
9382 tackling the problem of managing distributed network devices. Unfortunately, it is not
9383 much used commercially. DME is an object-oriented environment (like CMIP). The main
9384 problem with DME is that it seems to over-generalize the framework. This causes a
9385 problem for the business interests of competing vendors (if SunNet Manager, HP
9386 OpenView, and IBM Netview all have the same GUI, protocols, etc., these platforms may
9387 lose bargaining position based on unique capabilities).
- 9388 • (U) Hierarchical Network Management System (HNMS) comes out of the Network
9389 Attached Storage (NAS) domain. Its goal is to provide the capability to manage and
9390 monitor a very large Internet Protocol network. It relies on four types of modules: a
9391 server, a database, IO input/output modules, and UI user interface modules. All inter-
9392 module communication is done by the Hierarchical Network Management Protocol
9393 (HNMP). HNMP (like SNMP) is built on top of UDP/IP. Finally, four types of services
9394 are provided by HNMS: system parameters setting, data exchange, device discovery, and
9395 object management. In general, HNMS is more complex than SNMP and thus not as

9396 successful in the marketplace.

9397 • (U) Hypermedia Management Architecture (HMMA) comes out of the Web-Based
9398 Enterprise Management (WBEM) initiative of the Distributed Management Task Force
9399 (DMTF) whose URL can be found at <http://www.dmtf.org/standards/wbem/>. It is a result of a
9400 movement to combine network management with system and desktop management.
9401 WBEM is supported by Microsoft, Compaq, Cisco, Intel, HP, etc. The idea is to integrate
9402 existing standards into a framework, combining Desktop Management Interface
9403 (DMI/RPC) for desktops/servers with SNMP for network management, and doing all
9404 related Internet communication through Hypertext Transfer Protocol (HTML/HTTP).
9405 This aggregated architecture can then be managed using any Web browser, which is an
9406 advantage over plain SNMP.

9407
9408 (U) However, HMMA can be viewed not as a SNMP competitor but, rather as the long-
9409 awaited HTTP version of SNMP. The HMMP Protocol has been submitted to the IETF
9410 forum, and the HMMS Schema has been submitted to the DMTF forum. Of all the
9411 competitors to SNMP, HMMA perhaps has some chance of succeeding.

9412 (U//FOUO) If a choice has been made to employ SNMP-based network management techniques,
9413 then an alternative to full SNMPv3 implementation would be to use non-native encryption
9414 (outside of the SNMPv3 specified techniques), such as IPsec or TCP/TLS (Transport Layer
9415 Security). This alternative encryption choice may prove to be more efficient in terms of
9416 computation burden, as compared with full SNMPv3 operation. Finally, as in the prior
9417 evaluations concerning out-of-band versus in-band network management, the ultimate alternative
9418 to in-band SNMPv3 would be to build a dedicated (physical) or dial-up backup network for
9419 network management purposes. And when it comes to the issue of fault management,
9420 consideration of HTTP over SSL has the problem of connection-orientation which would rule it
9421 out (as compared with SNMPv3).

9422 **2.5.3.3.8 (U) Complementary Techniques**

9423 (U//FOUO) As has already been shown, a complementary (or alternative) technique to the full
9424 implementation of SNMPv3 would be to implement SNMPv1 over IPsec or over TLS/TCP, due
9425 to the fact that SNMPv3 messages can require greater network capacity (mainly an issue only on
9426 lower data rate networks).

9427 **2.5.3.3.9 (U) References**

9428 (U) RFC 3414, "User-based Security Model (USM) for version 3 of the Simple Network
9429 Management Protocol (SNMPv3)," <http://www.ietf.org/rfc/rfc3414.txt> , or
9430 <http://www.faqs.org/rfcs/rfc3414.html>, by Wijnen et al, December 2002.

9431 (U) RFC 3415, "View-based Access Control Model (VACM) for the Simple Network
9432 Management Protocol (SNMP)," <http://www.ietf.org/rfc/rfc3415.txt> , or
9433 <http://www.faqs.org/rfcs/rfc3415.html>, by Wijnen et al, December 2002.

9434 (U) "TCM IA Architecture Overview" briefing, 30 June 2004, by NSA's IAD TSAT IA IPT.

- 9435 (U) "Understanding the SNMP Security Vulnerability,"
9436 http://www.ins.com/downloads/seminars/SNMP_Vulnerabilities_13mar02.ppt, by Nicastro et al, March 2002.
- 9437 (U) "Understanding the Risks of SNMP Vulnerabilities,"
9438 http://www.lucent.com/livelink/255868_Whitepaper.pdf, by Davis et al, 2002.
- 9439 (U) "SNMP's Real Vulnerability,"
9440 <http://www.networkmagazine.com/shared/printableArticle.jhtml;jsessionid=OHQF54CDNR3ISQSNDBCSKHQ?articleID=8703341>, May 2002.
- 9442 (U) "Researchers Reveal Major SNMP Holes," <http://www.nwfusion.com/news/2002/0218snmp.html>,
9443 February 2002.
- 9444 (U) "Security in Network Management,"
9445 <http://opensores.thebunker.net/pub/mirrors/blackhat/presentations/bh-usa-97/Jeremy-snmpp.ppt>.
- 9446 (U) "PROTOS Test-Suite: c06-snmppv1," <http://www.ee.oulu.fi/research/ouspg/protos/testing/c06/snmppv1/>,
9447 October 2002.
- 9448 (U) "Security Comes to SNMP: The New SNMPv3 Proposed Internet Standards,"
9449 http://www.cisco.com/warp/public/759/ipj_1-3/ipj_1-3_snmppv3.html, by William Stallings, The Internet
9450 Protocol Journal, December 1998.
- 9451 (U) "CERT Advisory CA-2002-03 Multiple Vulnerabilities in Many Implementations of the
9452 Simple Network Management Protocol (SNMP)," <http://www.cert.org/advisories/CA-2002-03.html>,
9453 February 2002.
- 9454 (U) "EE579T Network Security, 11: Firewalls and SNMP," Spring 2004 lecture by Dr. Richard
9455 Stanley of WPI Worcester Polytechnic Institute, [http://ece.wpi.edu/courses/ee579t/EE579T-
9456 Class%2011C.ppt](http://ece.wpi.edu/courses/ee579t/EE579T-Class%2011C.ppt).
- 9457 (U) "The AES Cipher Algorithm in the SNMP's User-based Security Model," IETF Internet
9458 Draft by Blumenthal et al, <http://www.snmp.com/eso/draft-blumenthal-aes-usm-04.txt>, October 2002.
- 9459 (U) "Implementation and Performance Analysis of SNMP on a TLS/TCP Base," by Du et al,
9460 <http://www.umiacs.umd.edu/docs/Du.ppt>.
- 9461 (U) "Securing SNMP Across Backbone Networks," by Hia et al,
9462 <http://fiddle.visc.vt.edu/courses/ece5566/lectures/21securesnmp.pdf>, 2001.
- 9463 (U) "Deploying Secure SNMP in Low Data Rate Networks," by Hia et al,
9464 http://www.irean.vt.edu/navciiti/reports/secure_snmp_nov_2000.pdf.
- 9465 (U) "Session-based Security Model for SNMPv3 (SNMPv3/SBSM)," [http://net-
9466 snmpp.sourceforge.net/sbsm/SBSM.ppt](http://net-snmpp.sourceforge.net/sbsm/SBSM.ppt).
- 9467 (U) <http://www.nanog.org/mtg-0405/pdf/hardaker.pdf>

9468 (U) <http://www.net-snmp.org/sbsm/SBSM-bof-wes.ppt>

9469 (U) <http://www.net-snmp.org/sbsm/>

9470 (U) IETF drafts during 2004:

9471 (U) <http://www.net-snmp.org/sbsm/draft-perkins-snmpv3-overview-00.txt>

9472 (U) <http://www.net-snmp.org/sbsm/draft-hardaker-snmp-session-sm-01.txt>

9473 (U) “SNMP Architecture Alternatives,”

9474 (U) <http://www2.rad.com/networks/1999/snmp/index.htm>.

9475 (U) “SNMP Vulnerabilities Frequently Asked Questions (FAQ),”

9476 (U) http://www.cert.org/tech_tips/snmp_faq.html.

9477 **2.5.4 (U) Assured Resource Allocation: Gap Analysis**

9478 (U) Gap analysis for the Assured Resource Allocation Enabler indicates that the main areas of
9479 future development are as follows:

- 9480 • (U//FOUO) Need to develop SAR (Security Aware ad-hoc Routing) protocol capability
9481 that will work in tactical wireless GIG contexts.

- 9482 • (U//FOUO) More generally, need to verify that flexible and security-cognizant routing
9483 protocols such as FIRE (Flexible Intra-AS Routing Environment) can be implemented
9484 across the GIG and that the needed security QoS parameters (and associated routing table
9485 information) can be passed to GIG routers across any intervening red/black boundaries.

- 9486 • (U//FOUO) Need to develop a GIG Quality of Protection standard that will be a
9487 foundational element of the IA Policy-based Routing capability.

- 9488 • (U//FOUO) Need to develop a robust MLPP precedence and preemption standard for the
9489 GIG that will be well-integrated with the required foundation of a GIG PMI Privilege
9490 Management Infrastructure. Operational-based resource allocation/deallocation actions
9491 will demand that the associated privileges be consistently valid and universally
9492 distributed to needed policy enforcement points.

- 9493 • (U//FOUO) Need to flesh out the capabilities of SNMPv3, if this protocol is decided as
9494 the way to go for network signaling security (as this document suggests). SNMPv3 is
9495 fairly mature, except for key management aspects. Also need to validate that SNMPv3
9496 will be efficient enough when widely applied throughout the GIG.

- 9497 • (U//FOUO) Need to develop a Protection Profile for Network Management.

9498 (U//FOUO) Technology adequacy is a means of evaluating the technologies as they currently
9499 stand. This data can be used as a gap assessment between a technology's current maturity and
9500 the maturity needed for successful inclusion in the GIG.

9501 (U//FOUO) Table 2.5-1 lists the adequacy of the Assured Resource Allocation technologies with
 9502 respect to the IA attributes discussed in the RCD.

9503 **Table 2.5-1: (U) Technology Adequacy for Assured Resource Allocation**

This Table is (U)					
		Technology Category			Required Capability (attribute from RCD)
		IA Policy-based Routing	Operational-based Resource Allocation	Integrity of Network Management /Control /Monitoring /Recovery	
Enabler Attribute	Standard				IAAV1-IAAV4, IACNF6, IANMA2, IANMP1-IANMP5, IAAV21, IARC01 – IARC12, IAMP02
	Secure Solution				IACNF6, IACNF12, IANMA3, IANMP4, IANMP5, IARC01 – IARC12
	Scalable Solution				IAAV1-IAAV4, IAAV15, IAFM1, IANMA2
	Protection Profile	N/A	N/A		
	High Assurance				IACNF6, IAFM1, IANMP1-IANMP5, IAAV20
	Distributed/Global Reach				IAAV1-IAAV4, IAAV15, IAFM1, IAFM3, IAFM4, IANMA3, IANMP1-IANMP5
	Verifiable Solution				IAAV15
This Table is (U)					

9504

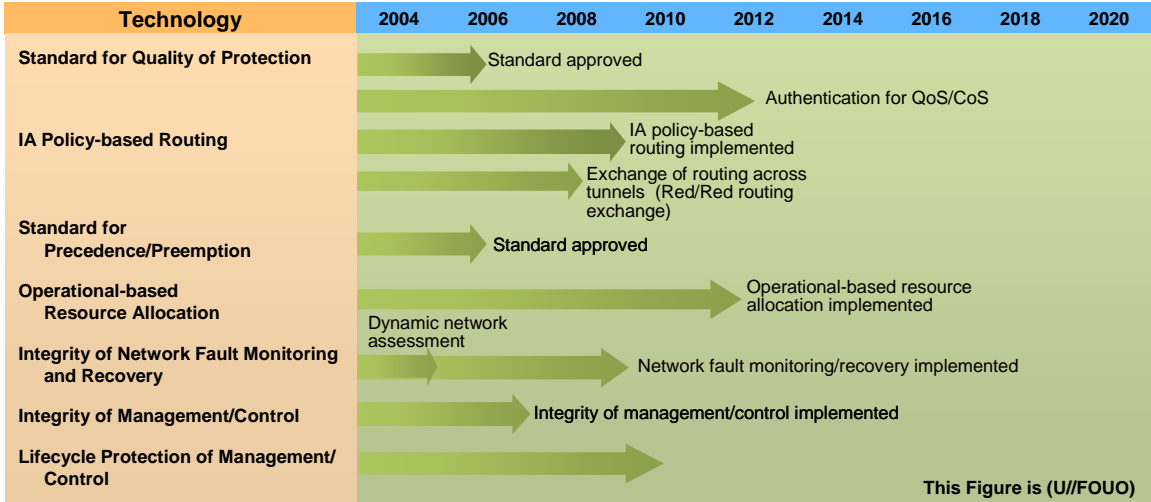
9505 (U//FOUO) In summary, the SNMPv3 standard is fairly mature (accounting for the black cell in
 9506 the matrix). At the current time, there is only provision for a Protection Profile dedicated to
 9507 Network Management. It is noted that there is a Protection Profile for “Switches and Routers” in
 9508 general (see <http://niap.nist.gov/cc-scheme/pp/index.html>). It is generally viewed that technology for
 9509 operational-based resource allocation is less mature and therefore less adequate for the GIG than
 9510 the available IA-based routing and network management technologies. Various sub-technologies
 9511 are available for the latter two areas but need to be integrated together and operationally
 9512 validated.

9513 **2.5.5 (U) Assured Resource Allocation: Recommendations and Technology Timelines**

9514 (U) The following is a list of recommendations for advancing the technologies required for the
9515 successful implementation of this GIG enabler:

- 9516 • (U//FOUO) Encourage the further development of adaptive security-driven (i.e., IA
9517 policy-based), wireless routing algorithms (such as SAR) for inclusion in JTRS and
9518 WIN-T
- 9519 • (U//FOUO) Advance the standards evolution and demonstration/implementation of
9520 extensible routing protocols (such as OSPF and IS-IS) so that IA metrics can be fully
9521 employed in routing decisions
- 9522 • (U//FOUO) Encourage the development of a GIG Precedence and Preemption standard
9523 that is closely tied with the required corollary GIG Privilege Management Infrastructure.
9524 The overall GIG Precedence/Preemption standard should ideally include the new GIG-
9525 BE-based VoIP-evolved DISN MLPP protocol as a subset capability
- 9526 • (U//FOUO) Advance, as an inclusion to the GIG Precedence and Preemption standard,
9527 the capability for rational post-preemption rescheduling so as to not leave GIG customers
9528 without requested services
- 9529 • (U//FOUO) Support developments that will ensure that an operational-based resource
9530 allocation infrastructure will have GIG-wide (i.e., worldwide) reach in its customer
9531 adjudication process (especially in the case of multiple requests and possible GIG
9532 congestion)
- 9533 • (U//FOUO) Push for the development of effective and scalable key management
9534 mechanisms for SNMPv3 messaging
- 9535 • (U//FOUO) Continue to follow the efficiency issue/impact of SNMPv3 native encryption
9536 (as being about 20%+ slower than SNMPv2 over IPsec or SSL)
- 9537 • (U//FOUO) Continue to track potential competing technologies/standards to SNMPv3 for
9538 network management/control/monitoring (even though various competitors have come
9539 and gone)

9540 (U//FOUO) Figure 2.5-11 contains preliminary technology timelines for this IA System Enabler.
9541 These are the result of research completed to date on these technologies. As the Reference
9542 Capability Document and the research of technologies related to these capabilities continues,
9543 these timelines are expected to evolve. The timelines reflect when the technologies could be
9544 available—given an optimum set of conditions (e.g., commercial community evolution starts
9545 immediately, GOTS funding is obtained, staffing is available). Technology topics with missing
9546 timelines (if any) indicate areas where further work is needed to identify the milestones.



9547

Figure 2.5-11: (U) Technology Timeline for Assured Resource Allocation

9548

9549

9550

9551 **2.6 (U) NETWORK DEFENSE AND SITUATIONAL AWARENESS**

9552 (U//FOUO) Network Defense and Situational Awareness is the IA System Enabler that allows
9553 the GIG to achieve the IA Mission Concept of Defend the GIG. It consists of enterprise-wide
9554 protection, monitoring, detection, and analysis that provide input into situational awareness of
9555 the operational mission(s) being carried out and result in response actions. The collection and
9556 analysis of sensor information—coupled with intelligence data, operational priorities, and other
9557 inputs—enables the creation of user-defined operational pictures of the assurance of GIG
9558 resources. The analysis of this information supports the development of situational awareness
9559 and the identification and characterization of potentially hostile activity.

9560 (U//FOUO) Situational awareness enables an adaptive and rapid adjustment of enterprise
9561 resources in response to unauthorized network activity and sub-optimal GIG resource
9562 configurations to support and achieve the six GIG IA Mission Concepts.

9563 (U//FOUO) This IA System Enabler consists of the following major functions as defined in the
9564 Joint Concept of Operations for Global Information Grid NetOps, 20 April 2004:

- 9565 • (U//FOUO) Protection: Prior actions taken to counter vulnerabilities in GIG information
9566 transport, processing, storage, service providers, and operational uses. Protection
9567 activities include emission security (EMSEC), communications security (COMSEC),
9568 computer security (COMPUSEC), and information security (INFOSEC)—all
9569 incorporating access control, cryptography, network guards, and firewall systems
- 9570 • (U//FOUO) Monitor: The monitoring of information systems to sense and assess
9571 abnormalities, the use of anomaly and intrusion detection systems. (Monitoring also
9572 includes receiving input from network monitoring as well as from a wide variety of real-
9573 time and status reporting)
- 9574 • (U//FOUO) Detection: Timely detection, identification, and location of abnormalities—to
9575 include attack, damage, or unauthorized modification—is key to initiating system
9576 response and restoration actions. [Detection also includes actions taken in anticipation of
9577 an attack (i.e., configuration adjustment)]
- 9578 • (U//FOUO) Analyze: Assessing pertinent information to [achieve] situational awareness,
9579 evaluate system status, identifying root cause, defining courses of action, prioritizing
9580 response and recovery actions, and conducting necessary reconfiguration of GIG assets as
9581 needed
- 9582 • (U//FOUO) Response: Directed actions taken to mitigate the operational impact of an
9583 attack, damage, or other incapacitation of an information system. Response also includes
9584 restoration—the prioritized return of essential information systems, elements of systems,
9585 or services to pre-event capability. (Coupled with restoration is the ability to undo a
9586 response)

9587 **2.6.1 (U) GIG Benefits due to Network Defense and Situational Awareness**

9588 (U//FOUO) The Network Defense and Situational Awareness System Enabler provides the
9589 following benefits to the GIG:

- 9590 • (U//FOUO) Dynamic protection of GIG network and computing resources from attack
9591 (attack being defined here as a sequence of one or more exploits or other actions taken by
9592 an adversary that lead to success of the adversary's mission); updated defensive posture
9593 based on near-real-time detection, intelligence, and operational and network information
9594 to enable a rapid response

- 9595 • (U//FOUO) Continuous, assured (e.g., availability, confidentiality, integrity) discovery,
9596 collection, processing, correlation, storage, and dissemination of intrusion detection and
9597 audit data. IA services applied to the sensor and audit resources ensure the availability,
9598 integrity, and confidentiality of the information received and also enable the
9599 authentication of the source

- 9600 • (U//FOUO) Detection and sharing of events and anomalies at multiple tiers (i.e., local,
9601 regional, global) within the GIG. User-defined operational pictures (UDOP) of the
9602 situational awareness information will enable analysis at all tiers and response to events
9603 as they occur

- 9604 • (U//FOUO) Trusted, real-time, user-defined operational picture of the IA/security posture
9605 of the GIG at any tier. Building upon the assured discovery, collection,
9606 processing/analysis, storage and dissemination of intrusion detection information, and
9607 audit and network management data, authorized users will be able to customize their
9608 view into the GIG as required to meet operational needs and also continuously monitor
9609 GIG network activity

- 9610 • (U//FOUO) Rapid analysis and response alternatives developed and modeled. Collection
9611 of sensor information, audit data, and network management data is only one step in the
9612 process. Being able to rapidly analyze that information requires greatly enhanced
9613 correlation, analysis, and modeling tools over what is currently available today in order to
9614 determine if an attack is occurring or imminent, and what the impact of such an attack
9615 might be if not countered

- 9616 • (U//FOUO) Enterprise-wide tools will enable the capability to rapidly monitor, analyze,
9617 and respond to system, computing, and network attacks, degradations, outages, misuse of
9618 resources, and events such as changes in operational priorities

- 9619 • (U//FOUO) Automatic and global intelligent (self-learning) defensive action enforcement
9620 to contain, recover, restore and undo, and reconstitute the GIG. Having determined an
9621 attack is underway—or imminent—and with likely resulting damage, alternative
9622 defensive countermeasures can be postulated and modeled/evaluated before
9623 implementation throughout the GIG

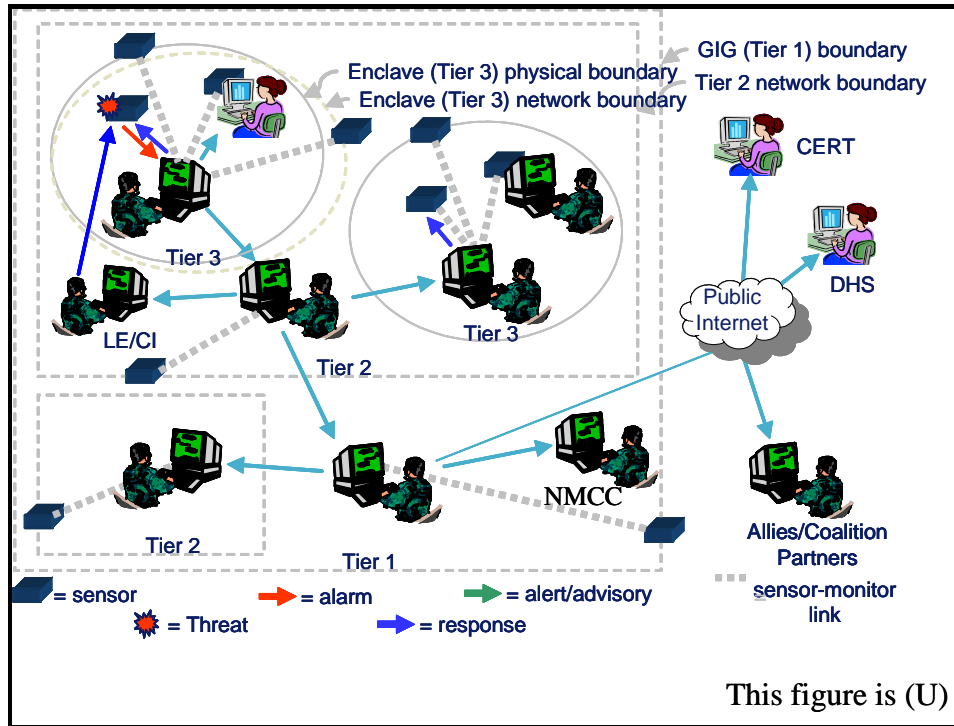
- 9624 • (U//FOUO) Governance of response actions. There are potential legal ramifications to
9625 employing defensive countermeasures to an attack. The analysis and modeling that will
9626 be available will strengthen the legal position that all due diligence was taken to analyze
9627 alternatives before deploying any response
- 9628 • (U//FOUO) Automatic prediction of attack strategies, objectives, and targets based on
9629 intrusion detection data, network data, and attack patterns. Automated tools performing
9630 trend analysis of sensor data and log files will provide the GIG with the capability to
9631 predict when and where identified attacks may appear elsewhere on the network

9632 **2.6.2 (U) Network Defense and Situational Awareness: Description**

9633 (U//FOUO) Network Defense and Situational Awareness is a critical enabler to provide the
9634 protection and support needed to achieve the GIG Mission Concept of Defend the GIG. This
9635 enabler defines actions taken to protect against, monitor, detect, analyze, and respond to potential
9636 and actual unauthorized network activities as well as unintentional non-malicious user error that
9637 could potentially harm the GIG. A concerted effort is required to find solutions to current
9638 technology issues related to an accurate view of organic system strengths and weaknesses for an
9639 enterprise of information on the scale of DoD's to be secure, available, and responsive to
9640 operational requirements.

9641 (U//FOUO) As a measure of effectiveness, DoD-wide system administration is highly dependent
9642 upon an accurate, real-time understanding of the configuration and situational awareness of DoD
9643 networks. Adversaries may periodically identify a weakness in a system, exploit that weakness,
9644 and then return the system to its original state. In addition, multiple attacks and exploitations can
9645 occur simultaneously and affect multiple missions. The planning of appropriate Courses of
9646 Action (COAs) will require constant awareness of the system and network configuration/state,
9647 which can lead to an overwhelming amount of data that needs to be analyzed. As a result, the
9648 task for administrators and analysts to understand how disparate attacks on a network affect an
9649 ongoing mission(s) and subsequently determine effective countermeasures becomes even more
9650 difficult.

9651 (U//FOUO) Distributed sharing of information is an important capability and begins with the
9652 monitoring and collecting of sensor information across the GIG. Referring to
9653 Figure 2.6-1, it can be seen that sensor information will be gathered from various locations and at
9654 all levels to include local (Tier 1), regional (Tier 2), and global (Tier 3) tiers. Information will be
9655 shared across all tiers, to include both peer-to-peer but also vertically within the organization.
9656 Further, while there might be a loss of information as it traverses horizontally and vertically, it is
9657 critical to have the ability for higher functions in the vertical space to be able to drill-down into
9658 specifics of a lower tier's data.



This figure is (U)

9659

9660

Figure 2.6-1: (U) Representative Sensor Configuration

9661 (U//FOUO) Today, sensors are primarily distinct special purpose devices (e.g., Intrusion
 9662 Detection Systems [IDSs], Intrusion Prevention Systems [IPSS], Firewalls, Guards)—providing
 9663 the information that is monitored and analyzed. In the future, every node and CND device on the
 9664 network will provide sensor information from its unique perspective and that will be coupled
 9665 with intelligence information, mission priorities, and audit logs to create a much broader view of
 9666 the operational picture. Sensors can be grouped in zones that are defined by geography, function,
 9667 and security. Zone/node sensors that can operate on the concept of reporting status changes to
 9668 their nearest neighbor will also be integrated into the GIG.

9669 (U//FOUO) A major goal of the GIG is to provide a Black Core for the data sent across it to
 9670 transit. The term Black in this sense means that the data traversing the GIG is encrypted, and if
 9671 necessary, also integrity-protected. Performance/situational monitoring and analysis of mixed
 9672 mode Black Core will require a change to sensor strategy. Sensors that require access to
 9673 encrypted information will need to be located before encryption. This introduces a host of new
 9674 challenges, including management and control of distributed sensors and sensor collection and
 9675 processing across multiple classification boundaries.

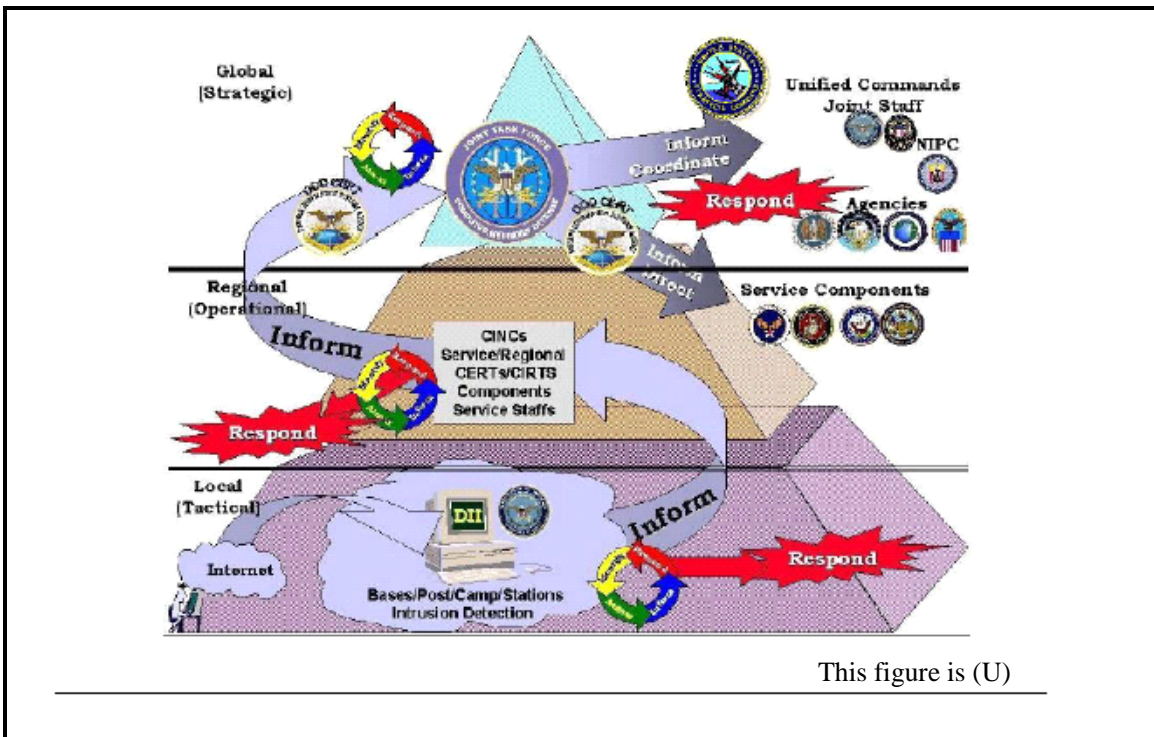
9676 (U//FOUO) While this notion of a Black Core provides significant confidentiality and data
 9677 integrity protection, it can also limit the ability of the GIG core itself to detect attacks. First, if all
 9678 data packets are encrypted at the IP level (e.g., by HAIPes or commercial IPsec
 9679 implementations), the GIG cannot detect the contents of the packets, and thus cannot detect
 9680 viruses, worms, or other malicious logic. As a result, the source of an attack may be hidden.
 9681 Information from the red IP header will need to be made available to the black IP header.

9682 (U//FOUO) Similarly, if the IP traffic is being tunneled (that is, there is a black IP header
9683 wrapped around the actual traffic), the GIG core may not even be able to tell where data packets
9684 are originating. At best, the GIG core can only tell that there is an unusual amount of traffic (e.g.,
9685 either much larger amounts of traffic than is normal or a usually-busy link goes quiescent). The
9686 GIG cannot directly tell that an attack is under way; nor can it launch a response to that attack. In
9687 this case, the only place where attacks can be detected, and the only place from which a response
9688 can be effected, is the application-layer code at the end system.

9689 (U//FOUO) Based on the size and complexity of the GIG, CND capabilities will need to be
9690 available for high volume, high speed connections to a variety of services (i.e., provider services,
9691 coalition services, and cross-domain services). Monitoring and collection of sensor information
9692 from coalition users and devices connected to the GIG is a serious concern.

9693 (U//FOUO) A non-DoD entity interface specification is needed to identify what minimum sensor
9694 information is required and how it is to be provided. This specification must also address how
9695 defensive actions will be promulgated to coalition partners. Correlating sensor information
9696 received from various networks will introduce additional challenges.

9697 (U//FOUO) Detection of anomalous behavior, detection of attack, quality of service, deviations
9698 from expected communication patterns, and all sorts of detailed monitoring provide the
9699 capability to ensure the integrity of individual GIG services and the enterprise-wide assurance of
9700 all managed information systems. Referring to Figure 2.6-2, it can be seen that if anomalous or
9701 attack activity is detected then the appropriate response will be performed at each tier.



9702

9703

Figure 2.6-2: (U) Representative Flow of Situational Awareness Data

UNCLASSIFIED//FOR OFFICIAL USE ONLY

9704 (U//FOUO) In addition, information is passed to the next higher tier for activities requiring a
9705 wider view, analysis, and response. An assumption can be made that anomalies and attacks will
9706 be detected at the lowest tiers. While this may be true today, more sophisticated attacks and
9707 anomalies can only be detected if a wider view is obtained from the outset. Consequently, while
9708 real-time data passed to higher tiers consists of only that which is necessary for real-time
9709 response, detailed data should also be made available periodically for off-line analysis to identify
9710 trends and to apply algorithms for low-intensity attacks, intrusions, and exploitation.

9711 (U//FOUO) Authorized users must be guaranteed ready access to all information contributing to
9712 situational awareness. ad Authorized users also must be able to verify the integrity and source of
9713 origin of the information—and in some cases ensure the confidentiality of the information. To do
9714 this, many different IA capabilities must be enabled. To determine that a user is authorized
9715 requires that there be mechanisms to provide assured identities to all users and mechanisms to
9716 derive an authenticated session score to confirm that a user identity has been authenticated to
9717 some level of assurance. A digital access policy will specify how access control to sensor
9718 information and any operational displays will be enforced and under what conditions exceptions
9719 to the policy will be managed. Management of sensor resources will fall under Section 3.5, the
9720 Assured Resource Allocation Enabler.

9721 (U//FOUO) Managers and users of the GIG need near real-time awareness of current threats,
9722 configuration, status, and performance of the GIG and its components. A trusted UDOP is a
9723 tailored view of an operational cyberspace picture. The GIG will provide relevant situational
9724 views of GIG operations at any level, with aggregation and event correlation to the higher levels
9725 and from peer-to-peer. Automated situational views will be enabled through:

- 9726 • (U//FOUO) Continuous monitoring of GIG configuration, status, and performance
- 9727 • (U//FOUO) Posting of situational awareness information (raw and processed)
- 9728 • (U//FOUO) Assembly of situational awareness information (monitored data plus threat
9729 and operational priorities)
- 9730 • (U//FOUO) Storage of situational awareness information. In addition to intrusion
9731 detection information, situational awareness will encompass network management data,
9732 intelligence findings, operational missions, operational mission requirements and
9733 priorities, and IA service status

9734 (U//FOUO) The CND component of the GIG will also provide the capability to take appropriate
9735 action on processed situational awareness data:

- 9736 • (U//FOUO) Automated display modifiable to suit each level of GIG management
- 9737 • (U//FOUO) Enterprise-wide mapping of services/applications to identify and mitigate
9738 vulnerabilities of all DoD hosts and associated services and applications
- 9739 • (U//FOUO) Enterprise-wide tools to rapidly evaluate, analyze, and respond to system and
9740 network attacks, degradations, outages, and events

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 9741 • (U//FOUO) Ability to rapidly adjust the GIG configuration based on different cyber
9742 Information Operations Condition (INFOCON) levels to best respond to identified
9743 situations

9744 (U//FOUO) The GIG will have the capability to immediately identify, detect, and respond
9745 appropriately to anomalies, attacks, or disruptions from external threats, internal threats, and
9746 natural causes. Once the event has occurred, the GIG will have the capability to implement
9747 mission impact analysis/battle damage assessment. The GIG will have the automated response
9748 capability to globally enforce intelligent (self-learning) defensive actions that contain, recover,
9749 restore, and reconstitute the GIG (e.g., automatically block DoS attacks traffic to vulnerable DoD
9750 hosts, and counter attack). Response actions will be coordinated across a broad range of
9751 operational elements, including Enterprise Service Management for configuration management
9752 and restoration of disrupted or degraded capabilities.

9753 (U//FOUO) Cyber attack attribution will play an essential role in identifying attackers and
9754 deterring further attacks. These capabilities will provide attacker/attack profiles and
9755 fingerprinting, trace to true country of origin, as well as provide complete trace-back and
9756 geolocation attackers. Forensic data will be captured and shared with Law Enforcement and
9757 Counter-Intelligence to investigate and if warranted prosecute perpetrators of unauthorized
9758 activities.

9759 (U//FOUO) As a complementary mechanism, a network capability will collect and assess
9760 network data to provide warnings of compromise to CND command and control elements, and
9761 information will be further disseminated to subordinate CND organizations. It will provide CND
9762 analysis of network data to detect if a severe compromise calls into question the integrity of the
9763 GIG.

9764 **2.6.3 (U) Network Defense and Situational Awareness: Technologies**

9765 (U//FOUO) The following technology areas support the Network Defense and Situational
9766 Awareness IA System Enabler:

9767 (U) Note: For convenience of analysis and organization, the technologies have been grouped
9768 together by the major function it is most designed to effect. This is not meant to suggest that the
9769 following technologies can only support one function, as many span multiple functions.

9770 (U) Protection

- 9771 • (U) Protect Technologies
- 9772 • (U) Firewalls
- 9773 • (U) Filters/Guards
- 9774 • (U) Anti-Virus, Anti-SPAM
- 9775 • (U) Disk and File Encryption
- 9776 • (U) Deception Technologies
- 9777 • (U) Honeypot
- 9778 • (U) Honeynet

9779 (U) Monitor

- 9780 • (U) Situational Awareness
- 9781 • (U) User-Defined Operational Picture (UDOP)
- 9782 • (U) Network Operations (NETOPS)
- 9783 • (U) Network Mapping
- 9784 • (U) Vulnerability Scanning

9785 (U) Detection

- 9786 • (U) Intrusion Detection Systems (IDS)
- 9787 • (U) Host-Based IDS, Network-Based IDS
- 9788 • (U) Misuse Detection, Anomaly Detection
- 9789 • (U) Intrusion Prevention Systems (IPS)
- 9790 • (U) Host-Based IPS, Network-Based IPS
- 9791 • (U) User Activity Profiling

9792 (U) Analyze

9793 • (U) Cyber Attack Attribution

9794 • (U) Traceback

9795 • (U) Correlation Technologies

9796 (U) Response

9797 • (U) CND Response Actions

9798 • (U) Courses of Action (COAs)

9799 • (U) Automated IA Vulnerability Alert (IAVA) Patch Management

9800 **2.6.3.1 (U) Protect Technologies**

9801 **2.6.3.1.1 (U) Technical Detail**

9802 (U) The ability to protect GIG network assets from computer network attack is a key cornerstone
9803 of computer network defense (CND) capabilities. A robust CND architecture includes both
9804 defense-in-depth and defense-in-breadth:

9805 • (U) Defense-in-depth - multiple layers of protection through the network against a
9806 particular attack type

9807 • (U) Defense-in-breadth - protection against various attack types through and across the
9808 network

9809 (U) Protection capabilities tend to be the first line of defense against network attacks as well as
9810 the propagation of potentially harmful non-malicious user activity. Less sophisticated adversaries
9811 can often be deterred by the sheer existence of protect technologies in today's network
9812 architectures. The most straightforward example of this is the placement of stateful firewalls at
9813 network perimeters that serve to deflect automated scanning and probing activity.

9814 (U) Current protect technologies are for the most part limited to static defenses against known
9815 attack types. They include, but are not limited to, the following technology areas:

9816 • (U) Network-Based Firewalls - The most common current implementation of protect
9817 technologies is network firewalls situated at perimeter boundaries to restrict data
9818 communications to and from one of the connected networks [RFC 2828]. These firewalls
9819 often provide the division between intranets and the Internet and come in both stateful
9820 and non-stateful varieties

9821 • (U) Host-Based Firewalls - Includes software application firewalls such as those that
9822 come pre-packaged with operating systems as well as independent commercial software
9823 firewalls, and hardware-based firewalls resident on the network interface card. Current
9824 hardware-based firewalls are highly resistant to attacks that successfully gain user access
9825 to a host

- 9826 • (U) Network Filtering Devices - A means of restricting data communications between
9827 connected networks—often implemented on network routers. These filtering devices can
9828 act as primitive non-stateful firewall devices
- 9829 • (U) Application Filters - A means of restricting data communications at the application
9830 layer (e.g., wrappers)
- 9831 • (U) Virus Protection - Software designed to search hard drives and disks for known
9832 viruses and then quarantine any found
- 9833 • (U) Disk and File Encryption - Software designed to encrypt portions of a disk to protect
9834 data while not in use
- 9835 • (U) Guards - Guards are generally used to prevent unauthorized data transfer between
9836 security domains. Hence, guard technology is discussed in Section 2.3.

9837 **2.6.3.1.2 (U) Usage Considerations**

9838 **2.6.3.1.2.1 (U) Implementation Issues**

9839 (U//FOUO) Many protection mechanisms are currently implemented and managed on a device-
9840 by-device or application-by-application basis. This presents significant challenges in a
9841 distributed advanced system such as the GIG where implementation and management is designed
9842 with a tiered approach for all levels. For a network of this scale, it will be necessary to deploy
9843 technologies with advanced, centralized management capabilities.

9844 (U//FOUO) The current trend in patch management also presents significant issues within the
9845 GIG. Many commercial operating systems and applications, including virus protection software,
9846 rely heavily on regular updates and patches to maintain up-to-date protection capabilities. This
9847 approach is rudimentary at best, since it requires secure web portals, accurate and trusted update
9848 code without inadvertent consequences, and valuable bandwidth.

9849 **2.6.3.1.2.2 (U) Advantages**

9850 (U) There is a clear advantage to preventing malicious activity before it reaches its intended
9851 target. Preventing an attack is far more desirable than detecting an attack—then responding to
9852 and recovering from it. The better we do the former, the easier it will be to do the latter. We
9853 cannot assume, however, that all attacks can be prevented, and therefore we must rely on a full
9854 breadth of CND capabilities to defend the network.

9855 (U) Network-based protection systems such as perimeter firewalls and network filtering devices
9856 offer the advantage of protecting entire enclaves from many types of attack at the gateway
9857 between the Internet and an intranet.

9858 (U) Host-based protection systems, on the other hand, push the protection capabilities to the
9859 network endpoints. Adversaries frequently consider these endpoints, often user workstations, to
9860 be attractive and more vulnerable targets to attack. By placing resilient host-based firewalls on
9861 the individual workstations, the defensive posture is increased significantly and makes them less-
9862 attractive targets. An additional advantage is that even if one workstation is compromised, the
9863 adversary still does not have open access to other workstations on the intranet. By pairing host-
9864 based firewalls with network-based perimeter firewalls, an additional layer is added to the
9865 defense-in-depth architecture.

9866 (U) The application filters technology area, including virus protection, provides the advantage of
9867 another defense-in-depth layer against cyber attack, this time at the application layer. A variety
9868 of wrappers have been developed to intercept system calls intended to exploit an application,
9869 operating system, or host access. Commercial filters exist to scan email for malicious
9870 attachments. This approach to protect workstations can stop attacks such as worms from
9871 propagating past the infected host or further infecting the host.

9872 (U) Disk and file encryption, a current COTS technology area, provides the advantage of
9873 encrypting file data stored on hard drives. This increases the work factor required by the
9874 adversary to access the file (the higher the number of encryption bits the longer it will take to
9875 crack).

9876 **2.6.3.1.2.3 (U) Risks/Threats/Attacks**

9877 (U//FOUO) Details of the GIG IA Risk Assessment, including detailed risks, threats, and attacks,
9878 are provided under separate cover. A fair amount is known about today's adversaries, and their
9879 goals and techniques. Unfortunately, very little can be said about the 2020 adversaries; thus
9880 making protecting against them a significant challenge.

9881 (U//FOUO) Results of the risk assessment indicate that protect technologies can in some cases
9882 provide a control surface for the adversary to launch an attack against the GIG. This is a risk that
9883 must be carefully considered when both designing and integrating protection mechanisms. The
9884 CND architecture must be designed so that protect technologies do not introduce vulnerable
9885 choke points. One approach to addressing this is to push the protection capabilities to the end
9886 point workstations rather than at network perimeters. Another approach in use today is
9887 redundancy in the architecture.

9888 **2.6.3.1.3 (U) Maturity**

9889 (U) There is much room for improvement in protect technologies.

- 9890 • (U) Current technologies are vulnerable to network attack and must be designed for
9891 robustness
- 9892 • (U) Protection must be designed throughout the network, not just at the perimeters.
9893 Adversaries often target the weaker, less protected network endpoints such as
9894 workstations
- 9895 • (U) Protection must be designed into all network components, not band-aids placed over
9896 weak Commercial-off-the-Shelf (COTS) and Government-off-the-Shelf (GOTS) devices

- 9897 • (U) Must be designed to be effective in encrypted network environments
 - 9898 • (U) Must be able to prevent attacks as close to the attack source as possible. This requires
9899 the ability to first detect where the source is at the onset of the attack
 - 9900 • (U) Must do a better job of protecting against an adversary with insider network access
- 9901 (U//FOUO) Because COTS products are widely available and have been so for years, protect
9902 technology is rated as Mature (TRL 7-9).

9903 **2.6.3.1.4 (U) Standards**

9904 (U) There are no current standards for protect technologies. Any standards should be closely tied
9905 to those for intrusion detection as a whole, in particular if the protect technology reports unusual
9906 behavior to a centralized monitoring or analysis engine.

9907 (U) The following Protection Profiles have been evaluated and certified with NIAP
9908 (<http://niap.nist.gov/cc-scheme/pp/index.html>):

- 9909 • (U) U.S. Government Firewall Protection Profile for Medium Robustness Environments,
9910 Version 1.0
- 9911 • (U) Application-Level Firewall for Basic Robustness Environments Protection Profile,
9912 Version 1.0
- 9913 • (U) Application-Level Firewall for Medium Robustness Environments Protection Profile,
9914 Version 1.0
- 9915 • (U) Traffic Filter Firewall Protection Profile for Medium Robustness Environments,
9916 Version 1.4
- 9917 • (U) Traffic Filter Firewall Protection Profile for Low Risk Environments,
9918 Version 1.1

9919 **2.6.3.1.5 (U) Cost/Limitations**

9920 (U) Protect technologies range from inexpensive, such as host-based virus filters, to moderately
9921 expensive, such as perimeter firewalls.

9922 (U) The requirement to continuously update today's protect technologies with security patches
9923 and new signature downloads is a significant limitation to their usefulness and survivability.
9924 Current industry practice is to issue a constant stream of patches that must be evaluated and
9925 implemented—requiring significant management overhead and annual licensing agreements.
9926 Without the most recent updates, these systems remain vulnerable to a variety of attacks, many
9927 of which are readily downloaded from the Internet.

9928 (U) A disadvantage of network-based protection systems is that once an attack pierces the edge
9929 device, it can cause widespread harm within the intranet. This is especially the case if little or no
9930 internal protection systems are in place.

9931 (U) A disadvantage of host-based protection systems is that a greater number of protection
9932 systems must be managed. Centralized management capabilities will be critical to this
9933 architectural approach.

9934 (U) The disadvantages of application filters include scalability with technologies that do not have
9935 centralized management systems, complexity associated with customization per user behavior,
9936 and any reliance upon signatures that must be updated on a regular basis.

9937 (U) The disadvantage of disk and file encryption is that when a file or encrypted partition is
9938 being accessed, it is decrypted and vulnerable. This is an inexpensive technology available today.

9939 **2.6.3.1.6 (U) Dependencies**

9940 (U) The ability to adequately protect a network relies heavily on maintaining control of the GIG
9941 assets as well as enforcing strong policies and procedures which GIG users are bound to follow.

9942 **2.6.3.1.7 (U) Alternatives**

9943 (U) The alternative to wide deployment of protect technologies is the incorporation of a strong
9944 IA architecture within the GIG.

9945 **2.6.3.1.8 (U) Complementary Techniques**

9946 (U) A resilient GIG network with a strong IA architecture goes a long way to provide protection
9947 against cyber attack and can therefore be considered a complementary technique. For example,
9948 encrypted network segments, use of strong authentication, and well written software immune
9949 from buffer overflow attacks can all serve to prevent network attacks. There will be holes,
9950 however, that protect technologies will serve to plug.

9951 **2.6.3.1.9 (U) References**

9952 (U) "A Public Key Infrastructure for the Secure Border Gateway Protocol (S-BGP),"
9953 by K. Seo, C. Lynn, and S. Kent, DARPA Information Survivability Conference and Exposition
9954 II, Volume 1, June 2001, pp. 239-253.

9955 (U) "Document Integrity through Mediated Interfaces," by M. Tallis and R. Balzer, DARPA
9956 Information Survivability Conference and Exposition II, Volume II, June 2001, pp. 263-272.

9957 (U) "Dynamic VPN Communities: Implementation and Experience," by D. Kindred and D.
9958 Sterne, DARPA Information Survivability Conference and Exposition II, Volume 1,
9959 June 2001, pp. 254-263.

9960 (U) Internet Security Glossary, Version 2, 20 August 2004 [replaces RFC2828].

9961 (U) "Preventing Denial of Service Attacks on Quality of Service," by E. Fulp, Z. Fu,
9962 D. Reeves, S. Wu, and X. Zhang, DARPA Information Survivability Conference and Exposition
9963 II, Volume II, June 2001, pp. 159-174.

9964 (U) "Security at the Network Edge: A Distributed Firewall Architecture," by T. Markham, and
9965 C. Payne, DARPA Information Survivability Conference and Exposition II, Volume 1, June
9966 2001, pp. 279-286.

9967 (U) <http://www.securecomputing.com/>

9968 (U) <http://www.symantec.com>

9969 **2.6.3.2 (U) Deception Technologies**

9970 **2.6.3.2.1 (U) Technical Detail**

9971 (U) Information systems have seen a growth in size and complexity over the past several years.
9972 Unfortunately, the ability to defend these systems has not evolved as quickly as the growth in the
9973 sophistication, tools, and techniques of attackers. Attackers are constantly developing new
9974 avenues for exploitation. Fortunately, research activities over the past several years have
9975 produced new technologies that will support a more advanced and layered approach to security.

9976 **2.6.3.2.1.1 (U) Honeypots**

9977 (U) “A honeypot is an information system resource whose value lies in unauthorized
9978 or illicit use of that resource.” - Lance Spitzner

9979 (U) Honeypots, also known as deception-based mechanisms or decoy-based intrusion protection,
9980 are specifically designed to attract an attacker’s attention away from an operational system into
9981 an environment where the attacker can be observed and monitored—ideally without the
9982 attacker’s knowledge.

9983 (U) The intention of honeypots is not to capture an attacker or to thwart an attack, but rather to
9984 allow an attack to proceed in a controlled manner as a means to monitor and gather information
9985 about new techniques and methods used to compromise systems. This must be done while
9986 carefully balancing the benefit of learning the attacker’s methods against the risk that a
9987 compromised system will be used as a launching point to attack real operational systems or other
9988 systems on the network.

9989 (U) In general, there are two ways that honeypots are implemented:

- 9990 • (U) Production – primarily used by companies or corporations to protect against an
9991 attack, easy to use, but capture limited amounts of information
- 9992 • (U) Research – primarily used by research, military, or government organizations,
9993 complex to deploy and maintain, but capture extensive amounts of information

9994 (U) The two general types of honeypots are:

- 9995 • (U) Low-Interaction Honeypots – requires less monitoring, limited interaction, normally
9996 work by emulating services and operating systems
- 9997 • (U) High-Interaction Honeypots – requires more monitoring, more complex, normally
9998 involve real operating systems and applications

9999 (U) Low-Interaction Honeyd - Low-interaction honeypots, such as Honeyd, work on the
10000 concept of monitoring unused IP space. Once an attack is attempted, the connection is
10001 intercepted and redirected to an emulated service. The honeypot is then able to detect and log the
10002 activity, as well as capture all of the attacker's interaction with the emulated service. In some
10003 honeypots, actual operating systems can also be emulated.

10004 (U) High-Interaction Honeyd - High-interaction honeypots, such as honeynets, offer an
10005 attacker an entire network of computers that are designed to be attacked. Within this highly
10006 controlled network, nothing is emulated or assumed. The idea here is to allow the attacker to
10007 find, attack, and break into these systems while controlling and capturing every activity.

10008 **2.6.3.2.1.2 (U) Honeynets**

10009 (U) As previously discussed, honeypots are deception devices within an operational network to
10010 learn an attacker's behavior and techniques. Honeynets, on the other hand, are an entire network
10011 of deception devices and are considered a combination of high-interaction honeypots. Their
10012 purpose is not focused on a specific operational environment, but rather to research an attacker's
10013 behavior in general. Also, honeynets are excellent tools for learning how to set up and manage
10014 all aspects of operational systems including traffic analysis, intrusion detection systems, system
10015 log and audit capabilities, system hardening, and risk management. Honeynets can be set up to
10016 model an entire operational network in order to research security risks and vulnerabilities of the
10017 network architecture.

10018 (U) The Honeynet Project is an ongoing research effort that is conducted on a volunteer basis by
10019 a non-profit research organization of security professionals. The organization is dedicated to
10020 learning the tools, tactics, and motives of the blackhat community and sharing the lessons
10021 learned to benefit both its members and the security community. Founded in October 1999, the
10022 Honeynet Project is now in its fourth phase, which is to create a centralized system that can
10023 collect and correlate data from distributed honeynets.

10024 **2.6.3.2.2 (U) Usage Considerations**

10025 **2.6.3.2.2.1 (U) Implementation Issues**

10026 (U) The two legal issues that need to be addressed when deploying deception technologies are
10027 entrapment and privacy. Although some attackers would like to argue that their activity was
10028 induced or persuaded, this is not the case. Attackers target honeypots/honeynets on their own
10029 initiative. Therefore, entrapment is most likely not an issue.

10030 (U) When deploying deception technologies in the U.S., three legal issues must be considered:

- 10031 • (U) Ensure compliance with laws restricting your right to monitor activities of users on
10032 your system
- 10033 • (U) Recognize and address the risk that the honeypot may be misused by attackers to
10034 commit crimes, or store and distribute contraband
- 10035 • (U) Consider the possibility that the honeypot can be used to attack other systems and
10036 result in potential liability for damages

10037 (U) At the federal level, the two main statutes concerning communications privacy are the
10038 Electronic Communication Privacy Act (18 USC 2701-11) and the federal Wiretap Statute (Title
10039 III, 18 USC 2510-22). Outside of the U.S., the applicable laws of jurisdiction may be different
10040 and should be investigated further.

10041 (U//FOUO) Honeypot and honeynet implementations can be complex and will vary depending
10042 upon the specific goals and objectives. Honeypots should be placed behind the firewall
10043 protecting the operational systems in order to mitigate risk. By doing so, the firewall will be able
10044 to log all traffic going through it and can provide some initial alerting capability. Review of the
10045 firewalls logs, assuming the firewall is not compromised, will assist in determining how the
10046 attack was initiated. Any packets sent to the honeypot are most likely probes from an attacker as
10047 no one should be communicating with it. Any traffic from a honeypot is indication that the
10048 device has been compromised. This is where it is critical to have the honeypot behind a
10049 firewall—to strictly control traffic to and from the honeypot.

10050 (U//FOUO) The system logs of the honeypot must be protected. An attacker will attempt to
10051 delete or modify system logs to cover their trail. In addition to normal system logs for the benefit
10052 of the attacker, provision must be made to export the real system logs (the ones tracking the
10053 attacker's moves) to a protected system for analysis. This has to be done in such a manner that a
10054 sniffer used by the attacker would not detect the log files were being sent. Different protocols
10055 and mechanisms can be used to achieve this.

10056 (U//FOUO) A sniffer, running on the firewall, can be used to capture keystrokes and screen shots
10057 so that there is documentation of everything the attacker enters and sees. To prevent the hacker
10058 from using encryption to hide activities, all services such as Secure Shell (SSH) should be
10059 disabled.

10060 **2.6.3.2.2.2 (U) Advantages**

10061 (U) The simple concept of honeypots and honeynets give way to some powerful strengths and
10062 advantages:

- 10063 • (U) Intrusion detection capability: Honeypots provide detection of new types of attacks
10064 (also known as “zero-day” attacks) that were undetected by other security mechanisms
- 10065 • (U) No false positives: Honeypots, by nature, do not conduct authorized activity.
10066 Therefore, any activity captured by a honeypot is considered suspect
- 10067 • (U) Small data sets of high value: Honeypots collect only small amounts of valuable
10068 information (i.e., what the attacker is doing and how the operational systems can be better
10069 protected), thus reducing the noise that needs to be analyzed
- 10070 • (U) Divert and control: Attackers probing a network will encounter honeypots that divert
10071 activity away from operational systems during some percentage of the time. The time an
10072 attacker spends investigating a honeypot will delay an attack on a real system
- 10073 • (U) Encryption or IPv6: Unlike most other security technologies, honeypots are
10074 unaffected by encrypted or IPv6 environments

10075 (U) Low-interaction honeypots have the advantage of simplicity. These honeypots are typically
10076 easier to deploy and maintain with minimal risk. A plug-and-play approach that involves
10077 installing software, and selecting the operating systems and services to be emulated and
10078 monitored makes deployment easy for most organizations. In addition, by containing the
10079 attacker's activity by emulated services, the risk is mitigated by never allowing access to an
10080 operating system to attack or do harm. Low-interaction systems work well because any access is
10081 anomalous.

10082 (U) High-interaction honeypots give the advantage of providing attackers with actual—not
10083 emulated—operating systems and services to interact with. This allows extensive amounts of
10084 information to be captured and as a result, a greater opportunity to learn the full extent of the
10085 attacker's behavior. Another advantage of high-interaction honeypots is that no assumptions on
10086 how an attacker will behave are made. Since the environment is open and all activity is captured,
10087 these honeypots are able to learn behavior beyond what is expected.

10088 **2.6.3.2.2.3 (U) Risks/Threats/Attacks**

10089 (U//FOUO) There are both security and liability risks involved with deploying honeypots. These
10090 devices will be compromised and could be used as launching points for other attacks. Given the
10091 fact that there are ways to fingerprint many honeypot implementations, it is safe to assume that
10092 an attacker will indeed determine that the device is a honeypot. Therefore, one must consider the
10093 threat that an attacker might retaliate in some way after being duped.

10094 (U//FOUO) All honeypots can and will be detected by an attacker who lingers long enough.
10095 Some honeypots provide signatures that can be easily fingerprinted warning attackers to move
10096 on. The firewalls providing some of the analysis data and protecting the operational systems can
10097 and will be compromised by determined attackers. The honeypots themselves will eventually be
10098 compromised by attackers who gain root access to the systems. The primary risk is that an
10099 attacker takes control of the honeypot, or honeynet, and uses it against the remaining operational
10100 systems or uses it as a launch point to other systems.

10101 (U//FOUO) Finally, there is a risk of overdependence on honeypots/honeynets. Although a
10102 honeypot/honeynet may be able to catch an attacker who is blindly groping a system, the same
10103 success will not be shared by a more sophisticated attacker with a focused mission. Therefore,
10104 implementing a honeypot/honeynet system may provide a false sense of security.

10105 **2.6.3.2.3 (U) Maturity**

10106 (U//FOUO) Honeypot technology has been around for many years and both commercial and
10107 Government-developed solutions are available. The current thrust in honeypot technology is to
10108 develop scalable solutions that more fully recreate a full operating system appearance to the
10109 attacker. In this regard, virtual machines have become highly useful to honeypot developers.
10110 Overall, maturity of honeypot technology is rated as Emerging (TRL 4-6), while honeynet
10111 technology is rated as Early (TRL 1-3).

10112 **2.6.3.2.4 (U) Standards**

10113 (U) There are no standards for honeypots and honeynets per se. However, there are standards
10114 that apply to data capture (what data should be captured at each honeynet and in what format)
10115 and data collection (what data should be sent to a central collection site and in what format).

10116 (U) Data Capture Standards

10117 • (U) All network activity (packets and full packet payload) must be captured in tcpdump
10118 binary format (OpenBSB libpcap standards) and rotated/compressed (gzip) on a daily
10119 basis

10120 • (U) Firewall logs must be converted to ASCII format to allow uploading into a
10121 centralized database

10122 • (U) An attacker's activity must be captured on the system itself. In the past, sniffing
10123 connections to capture keystrokes off the wire would suffice. However, attackers today
10124 are likely to adopt some form of encryption to communicate. The Honeynet Project has
10125 developed Sebek2, a kernel module that is capable of logging an attacker's keystrokes
10126 and capturing files uploaded via secure copy (scp)

10127 (U) Data Collection Standards

10128 • (U) Tcpdump binary logs – each honeynet can forward daily tcpdump binary log captures

10129 • (U) Firewall logs – every inbound and outbound connection logged by the firewall can be
10130 sent in ASCII text format on a daily basis

10131 **2.6.3.2.5 (U) Cost/Limitations**

10132 (U) Deception-based technologies are not necessarily expensive to deploy. The cost is dependent
10133 upon the size of the operational system in which they are being placed and the maintenance and
10134 support cost to operate and manage them. First and foremost, it is important to consider the
10135 nature and cost of containment and control. Measures should be taken to mitigate the risk of
10136 having a honeypot system deployed in a network. If a product does not support any native
10137 containment and control, the cost and complexity of implementation should be seriously
10138 examined.

10139 (U) Analysis of the data is another cost that must be factored. Some products provide integrated
10140 analysis, reporting, and alerting. However, other products require involvement by an
10141 administrator, which could have a significant impact on the cost of using such a system. Ongoing
10142 administrative costs include maintenance of content and restoration of the honeypot. Periodic
10143 updates to the content will be essential to maintain the appearance of a valid and live system.
10144 Also important is the need to periodically restore the system to a clean and controlled state. Once
10145 again, automated capabilities for restoration can greatly reduce administrative costs.

10146 (U) Honeypots, like any other technology, have limitations. Honeypots have a limited view in
10147 that only activity with direct interaction can be tracked and captured. Therefore, attacks made
10148 against other systems will not be captured. Also, chances are that an attacker will eventually
10149 learn that a device is a honeypot and either leave after cleaning up as much as possible, or worse,
10150 take punitive action against the operational systems. Even a successful honeypot will provide
10151 valuable data on the steps taken by an attacker, which has to be delivered to another system
10152 without the attacker's knowledge and then undergo extensive analysis before it can prove to be
10153 useful.

10154 **2.6.3.2.6 (U) Dependencies**

10155 (U) As an information gather tool, a honeynet can employ Methodology Fingerprinting to
10156 determine the patterns of behavior of a particular attack or attacker, as well as be used to
10157 discover the unknown.

10158 (U) A honeynet can perform these tasks by controlling, capturing, and analyzing data. Data
10159 control involves such activities as restricting inbound and outbound traffic from a compromised
10160 honeynet. Such tools as a Honeywall, firewalls such as OpenBSD firewall and Snort_inline (a
10161 modified version of Snort that is used in the Honeynet Project to drop or modify packets) would
10162 handle data control.

10163 (U) Capturing data allows the honeynet analysts to observe intruders, even in encrypted
10164 environments and without being noticed by the intruder. The honeynet analysts can also monitor
10165 all attacker activity. Data can be captured via keylogging, firewall logs, packet sniffer logs and
10166 Honeyd logs to name a few. Snort, Sebek, and Termlog are a few tools that may be used to
10167 capture data within a honeynet. The data can then be exported to a server for analysis.

10168 (U) Data analysis includes traffic analysis (IP addresses and ports, traffic frequency and volume),
10169 fingerprinting (flags and options indicate platform personality), content analysis, granularity,
10170 confidentiality issues, encryption, and digest analysis. Examples of tools used to analyze the data
10171 include HoneyInspector, which enables real-time analysis, PrivMsg, which extracts IRC
10172 conversations from tcmdump binary log files (eliminates noise), and Sleuthkit, a forensic toolset
10173 for analyzing hacked systems.

10174 (U) Most of the technologies (workstations, servers, firewalls, etc.) used to create Honeynets are
10175 not new. However, many of the tools used to control, capture, and analyze the data are new.
10176 Tools such as Sebek have become available within the last two years. In the future, developers
10177 are planning to include capabilities to automatically filter large volumes of data, correlate IDS
10178 data with other network data, and provide a unified view of the event or attack.

10179 **2.6.3.2.7 (U) Alternatives**

10180 (U) Other capabilities exist that could track the activities of attackers, but none in so controlled
10181 an environment.

10182 **2.6.3.2.8 (U) Complementary Techniques**

10183 (U) Honeypots complement network-based and host-based Intrusion Detection Systems (IDSs).
10184 Although closely related, honeypots do not require the capability to discriminate between
10185 operational traffic and attacker traffic nor share the likelihood of many false positives.

10186 **2.6.3.2.9 (U) References**

10187 (U) "The Evolution of Deception Technologies as a Means for Network Defense,"
10188 Recourse Technologies, <http://www.sans.org/rr/papers/30/recourse.php>.

10189 (U) "Honeypot Definitions, Requirements, Standards," version 1.5.3, 22 October 2003,
10190 <http://www.honeynet.org/alliance/requirements.html>.

10191 (U) "Honeypots - Definitions and Value of Honeypots," by Lance Spitzner,
10192 <http://www.tracking-hackers.com/papers/honeypots.html>, 29 May 2003.

10193 (U) "Honeypots - Definitions and Value of Honeypots," by Lance Spitzner,
10194 http://www.secinf.net/honeypots/Honeypots_Definitions_and_Value_of_Honeypots.html, 10 December 2002.

10195 (U) "Honeypots, the Hottest Thing in Intrusion Detection," by John Harrison,
10196 <http://channelzone.ziffdavis.com/article2/0%2C1759%2C1516562%2C00.asp>,
10197 4 November 2003.

10198 (U) "The Honeynet Project," <http://www.honeynet.org/misc/project.html>.

10199 (U) "Honeynet Project: What a Honeynet Is,"
10200 <http://www.awprofessional.com/articles/printerfriendly.asp?p=23948>.

10201 (U) "Know Your Enemy: GenII Honeynets,"
10202 <http://www.securesynergy.com/library/articles/051-2003.php>.

10203 (U) "Know Your Enemy: Honeynets," Honeynet Project,
10204 <http://www.honeynet.org/papers/honeynet/index.html>, 12 November 2003.

10205 (U) "Know Your Enemy," Chapter 8: Legal Issues, by Richard Salgado,
10206 <http://www.honeynet.org/book/Chp8.pdf>, 29 April 2004.

10207 **2.6.3.3 (U) Situational Awareness**

10208 **2.6.3.3.1 (U) Technical Detail**

10209 **2.6.3.3.1.1 (U) UDOP**

10210 (U//FOUO) Network situational awareness capabilities include monitoring tools (network health,
10211 bandwidth utilization, and key servers and processes) on-hand as percentage of those required for
10212 fixed and deployed forces. The CND UDOP provides situational awareness of CND activities,
10213 operations, and their impact, collaboration, and decision support to all levels of the GIG. The
10214 CND UDOP is the integration of a comprehensive data presentation interface and data storage
10215 coupled with intelligent data acquisition. The resulting solution is robust and flexible and
10216 provides situational awareness information across the DoD to support the Warfighter.

10217 (U//FOUO) These basic requirements define what will be included in the CND UDOP. The CND
10218 UDOP is defined as that portion of the IA and Network Operations (NETOPS) operational
10219 picture that provides local, intermediate, and DoD-wide situational awareness of CND activities,
10220 operations, and their impact, collaboration, and decision support. The emerging CND UDOP
10221 leverages common data, views, and mechanisms for data sharing and displays all information
10222 necessary for the defense of DoD networks.

10223 **2.6.3.3.1.2 (U) NETOPS**

10224 (U//FOUO) The CND UDOP is expected to receive the majority of its data from sources that will
10225 also feed the larger NETOPS picture.

10226 (U//FOUO) Information used to support the UDOP consists of both raw data inputs and
10227 processed and correlated alert information. Flow data is currently used for a number of analytical
10228 techniques—namely scan and application detection. Many analysis methods are available, and
10229 many others are under development.

10230 (U//FOUO) Core routers form the backbone of the existing monitor network. Attack detection
10231 and prevention systems installed at the core routers have the potential to detect and block attacks
10232 before they reach the enclaves. Sensors at these locations provide the analyst with a high-level
10233 view of attacks launched against large numbers of hosts located at different physical locations
10234 (provided that the data from the sensors is aggregated at some point). Also, a small number of
10235 sensors are required to detect and block attacks against a large number of hosts.

10236 (U//FOUO) Analysis is conducted on both raw and processed data whether acquired from the
10237 existing sensor grid or from other sources. The analysis uses both automated and manual means
10238 to correlate sensor grid data, alarms, and event detections. An alarm management interface
10239 provides operators the ability to acknowledge alarms and perform COAs on those alarms. When
10240 launched from the main dashboard level, the interface can show all of the alarms for the
10241 operator's sphere of responsibility.

10242 **2.6.3.3.2 (U) Usage Considerations**

10243 **2.6.3.3.2.1 (U) Implementation Issues**

10244 (U//FOUO) Usage considerations are complex and varied. The following list identifies
10245 significant requirements the system must deliver to the user.

- 10246 • (U//FOUO) The system must accomplish information sharing and information/data
10247 transmission in an appropriately controlled and secure environment, ensuring the
10248 appropriate security classification level for each level of user
- 10249 • (U//FOUO) In disseminating technical information to users, the system must provide the
10250 capability to evaluate, integrate, and synchronize proposed CND options with overall
10251 battle and security plans
- 10252 • (U//FOUO) The system must disseminate information on defensive strategies to the CND
10253 community
- 10254 • (U//FOUO) The system must provide information sharing and collaboration capabilities
10255 for near real-time tactical warning between the operations and intelligence communities
- 10256 • (U//FOUO) The system must provide a capability for distributed collaboration to
10257 coordinate mitigation and response in execution of the CND mission
- 10258 • (U//FOUO) The system must effectively enable controlled, releasable, and discloseable
10259 information sharing among authorized users within the DoD, other U.S. Government
10260 departments and agencies, law enforcement and other emergency response agencies,
10261 selected non-government and private sector entities, and organizations across a global
10262 architecture
- 10263 • (U//FOUO) The system must be scalable and adaptable to dynamic user requirements and
10264 have the reserve capacity to support surge loading and multiple military operations

10265 (U//FOUO) A sensor needs to be placed at every entrance and exit point to or from the network
10266 being protected. If the network in question has no gateways (LAN), an assessment must be made
10267 as to what the best collection points on the network are.

10268 (U//FOUO) The use of multiple and diverse sensor products compounds the analytical task of the
10269 network analysts. Each sensor has its own unique method for analyzing network packets and
10270 network sessions and for determining what constitutes an alert. To reduce the number of alerts
10271 sent to the analysts from different sensors, an automated approach to data correlation and
10272 summarization is needed.

10273 (U//FOUO) Data correlation needs to occur across commercial and government products,
10274 bridging the gap between network sensors, host-based information, and audit logs. Flow data is
10275 centralized and used to detect patterns. Mechanisms for centralizing data (dedicated circuits)
10276 must be in place to transport this data. The automated analysis of attacks should include
10277 indications of severity levels, damage assessment, and recommended COAs.

10278 **2.6.3.3.2.2 (U) Advantages**

10279 (U//FOUO) Effective CND requires an operational view of the networked environment to
10280 provide situational awareness of potential threats, attacks, network status, and other critical
10281 information to support a mission commanders' decision-making and prevent, stop, or reverse
10282 degradation of network resources due to unauthorized activities. The criticality of enhancing
10283 CND situational awareness is due to the increasingly information-centric operations conducted
10284 by DoD and its allies. Specifically:

- 10285 • (U//FOUO) Commanders and their forces are dependent upon accurate, complete,
10286 reliable, timely, and secure information to conduct their missions

- 10287 • (U//FOUO) Commanders and their forces are dependent on the GIG and other assets, and
10288 need to know when situations exist that can affect the information systems and networks
10289 supporting their critical Warfighting processes

- 10290 • (U//FOUO) DoD must protect, monitor, detect, analyze, and respond to unauthorized
10291 activity within DoD information systems and global networks to ensure continuity of
10292 operations throughout the spectrum of conflict (i.e., CND)

- 10293 • (U//FOUO) Commanders need the capability to quickly comprehend the status and
10294 reliability of their information and information systems to successfully engage in network
10295 centric operations

- 10296 • (U//FOUO) Network operators need the capability to develop user defined operational
10297 pictures (UDOP) for tailored or filtered views to meet the specific needs of Commanders
10298 and deployed Warfighters

- 10299 • (U//FOUO) Network operators require insight into the networked environment that will
10300 permit real-time decisions supporting security, continued availability, and restoration of
10301 DoD networks

- 10302 • (U//FOUO) Network operators need the capability to quickly share information
10303 concerning the status of allied/coalition nations' Command, Control, Communications,
10304 Computers (C4) systems

10305 (U//FOUO) Another advantage of the envisioned centralized structure is the use of flow analysis.
10306 Flow analysis requires much less data than content analysis, which eases computing, data
10307 transfer, and data storage requirements, resulting in significant performance benefits on a global
10308 network such as the GIG.

10309 **2.6.3.3.2.3 (U) Risks/Threats/Attacks**

10310 (U//FOUO) The CND UDOP is expected to get the majority of its data from sources deployed in
10311 the ESG. However, additional data sources (Joint CERT Database, Indications & Warnings, etc.)
10312 will need to be used to complete the CND UDOP picture. The correlation of information from
10313 many distributed sources represents both a risk and a challenge for DoD.

10314 (U//FOUO) The collaboration engine and tools are the hooks into the DoD collaboration
10315 software that among other capabilities allows users of the UDOP to collaborate with other users.
10316 The collaboration interface will include the capability for users of the UDOP to share and discuss
10317 incidents, reports, and alarms. Any failure of this collaboration capability could significantly
10318 impact mission success.

10319 **2.6.3.3.3 (U) Maturity**

10320 (U//FOUO) Automated Indications and Warning (I&W) is a proactive process that involves
10321 collecting, assembling, and analyzing large amounts of intelligence data from a variety of
10322 sources. Current collection, correlation, and visualization capabilities exist that support a
10323 NETOPS. The CENTAUR flow data analysis system has been operational since 2000.

10324 (U//FOUO) The rate of increase in network bandwidth is currently greater than the rate of
10325 increase in processing speeds and the rate of increase of memory sizes and speeds. As a result,
10326 automated I&W components built in software (i.e., IDSs, Traffic Normalizers [TNs], and
10327 Intrusion Prevention Systems [IPSs]) face significant difficulty being able to handle traffic at full
10328 line rate. Unfortunately, creating custom hardware such as Application-Specific Integrated
10329 Circuits (ASICs) requires a significant investment in manpower and in planning. In response to
10330 this need, various vendors have created programmable embedded systems that can process
10331 packets at full line rate in Gigabit, or higher, networks. Such technologies are broadly referred to
10332 as Network Processors (NPs).

10333 (U//FOUO) Overall, the maturity levels of both UDOP and NETOPS technologies are rated
10334 Emerging (TRL 4-6).

10335 **2.6.3.3.4 (U) Standards**

10336 (U//FOUO) Other than DoD Information Technology Security Certification and Accreditation
10337 Process (DITSCAP)-related certification and accreditation requirements, there are no standards
10338 directly applicable to this technology area. However, a requirements document, Computer
10339 Network Defense User Defined Operational Picture (CND UDOP) Requirements List, 23 March
10340 2004, has been released. This requirements list is expected to influence emerging standards by
10341 providing recommendations on a vendor-neutral, sensor information exchange format and
10342 interface standard.

10343 **2.6.3.3.5 (U) Cost/Limitations**

10344 (U//FOUO) Analysis data centers are affordable. Cluster technology, which combines
10345 independent computers into a unified system (or cluster) through software and networking,
10346 makes this analysis extremely scalable. Clusters are typically used for high availability to
10347 provide greater reliability or high performance computing to provide greater computational
10348 power than a single computer can provide. Beowulf clusters are an example.

10349 (U) Limitations to integrating a complete UDOP include the implementation of an enhanced
10350 sensor grid across the DoD enterprise, developing technologies scalable to the GIG, and creating
10351 detection tools that work with the IPv6 protocol. Sensor development is ongoing and will be
10352 deployed at some level in the near future. However, to achieve the full vision of the UDOP, more
10353 robust sensors will be necessary. Implementation of the sensor grid and continued research into
10354 the gap areas will further extend the current UDOP capabilities to meet UDOP needs.

10355 **2.6.3.3.6 (U) Dependencies**

10356 (U//FOUO) Visualization and correlation engine capabilities are dependent on both the ability to
10357 collect data from many sources at high data rates, and the ability to analyze this data in near real
10358 time. This technology requires a source of flow data, bandwidth to centralize data, and sufficient
10359 disk storage to store and process data. The ESG of 2008 is envisioned as a grid of sensors, each
10360 fully capable of collecting sufficient data in near real time to meet the needs of the UDOP. Fully
10361 implementing the sensor grid and maintaining sufficient centralized storage capacity are critical
10362 collection capabilities.

10363 **2.6.3.3.7 (U) Complementary Techniques**

10364 (U//FOUO) A complementary technique exists on the DoD enterprise at this time. CENTAUR is
10365 a metadata collection, storage, and analysis system that accepts Netflow data as its primary input.
10366 It enables analysis of traffic flow data produced by routers to determine the presence of
10367 malicious activity. The information is used to correlate/collaborate both reported incidents, as
10368 well as to detect anomalous activity including blatant and stealthy activities. Operational
10369 incidents are reported, and correlation of various network data is performed, reported, and
10370 distributed accordingly.

10371 **2.6.3.3.8 (U) References**

10372 (U) "Assessment of IA/CND Focus Areas," Mitre Corporation, June 2004.

10373 (U) "Beowulf Project Overview," <http://www.beowulf.org/overview/index.html>.

10374 (U) "Computer Network Defense User Defined Operational Picture (CND UDOP) Requirements
10375 List," DISA, 23 March 2004.

10376 **2.6.3.4 (U) Network Mapping**

10377 **2.6.3.4.1 (U) Technical Detail**

10378 (U//FOUO) Vulnerability scanning tools can discover and store topology and status information
10379 about transport-layer optical devices to data routers, switches, and IP addresses. They also have
10380 the capability to conduct a basic mapping of applications to their underlying systems and servers.
10381 These tools provide a graphical view of the environment and provide indicators of the presence
10382 of new devices that have appeared since the last scan.

10383 (U//FOUO) Vulnerability management tools find, evaluate, and optionally, eliminate
10384 vulnerabilities on systems before attackers take advantage of them. Efficient and comprehensive,
10385 near-real-time discovery tools are needed for accurate analysis of DoD's physical and virtual
10386 networks, as well as to identify applications running on the network and manifestations of cyber
10387 situational awareness. These tools are also needed to ensure that users adhere to security policies
10388 and to deter users from introducing vulnerabilities. This desired capability must also be scalable
10389 to very large networks.

10390 (U//FOUO) In some applications, mappers are combined with vulnerability databases and other
10391 correlation tools to identify potential weaknesses or routes of attack for various components. In
10392 such cases, these posture discovery tools enhance:

- 10393 • (U) Security Monitoring/Management
- 10394 • (U) Network Security
- 10395 • (U) Problem Management

10396 **2.6.3.4.2 (U) Usage Considerations**

10397 (U//FOUO) Usage considerations relate to the legal concerns associated with either passive
10398 listening or active vulnerability identification. Passive discovery tools have virtually no impact
10399 on normal operations as they represent one-way listening devices on the network. Active
10400 discovery tools impact bandwidth availability and can cause intrusion detection alarm conditions.

10401 **2.6.3.4.2.1 (U) Implementation Issues**

10402 (U//FOUO) In order to work most effectively, vulnerability management tools should have
10403 unimpeded access to the systems to be tested. Therefore, the vulnerability management tools
10404 must be inside of any site firewalls. In cases where multiple subnets protected by separate
10405 firewalls exist within an enclave, multiple vulnerability management tools will be needed. This
10406 increases the number of tools required—increasing cost and management difficulties.

10407 **2.6.3.4.2.2 (U) Risks/Threats/Attacks**

10408 (U//FOUO) The very feature of the GIG that makes it most beneficial, the ubiquitous access to
10409 the system resources enterprise-wide, is also what makes the GIG an attractive target for
10410 adversaries. There are over 90 countries with affirmed nation state computer network
10411 exploitation efforts at the tactical and strategic levels, most collecting critical information against
10412 the United States and her allies. Nearly 20 countries have confirmed, dedicated computer
10413 network attack programs. Their mere existence suggests success and satisfaction with the returns
10414 on their investments.

10415 (U//FOUO) In the GIG vision, all systems will be interoperable and information of all types
10416 reachable from anywhere. As a result there will be many insiders with potential access to
10417 information that was not available to them before. In addition, the connection of temporary
10418 coalition partners to the GIG will widen system access beyond our immediate control. Without
10419 accurate vulnerability detection and control, an adversary can use this increased capability
10420 against us and/or deny us the use of these capabilities at the point when we have become most
10421 reliant upon them.

10422 **2.6.3.4.3 (U) Maturity**

10423 (U//FOUO) Current network mapping and vulnerability discovery approaches are used for
10424 configuration management, vulnerability reduction, and for resource identification in large-scale
10425 enterprise networks. While both active and passive mapping solutions are currently installed, the
10426 usual means of accurate network discovery is to actively perform port mapping and open port
10427 investigations on network components to determine the following:

- 10428 • (U) The current host operating system
- 10429 • (U) The primary use of the network component
- 10430 • (U) Services and applications running on the system
- 10431 • (U) The physical connections and topology of the network
- 10432 • (U) If current platforms have unpatched vulnerabilities or are running unsecured services

10433 (U//FOUO) Together, the maturity of the various technologies of the Network Mapping
10434 technology area is rated as Emerging (TRL 4-6).

10435 **2.6.3.4.4 (U) Standards**

10436 (U//FOUO) Standards for mapping and vulnerability discovery are not applicable. In the GIG
10437 environment, near continuous monitoring will be essential because the environment will be so
10438 dynamic (ad hoc creation of expedient COIs and the associated vulnerabilities, etc.). Alert on
10439 Change could more appropriately be called Alert and Change since automated decision-making
10440 and response, based on GIG-wide situational awareness, will be vital to provide Active Network
10441 Defense for the GIG. This will require the use of agents and a GIG-wide hierarchy of agent
10442 functional stacks for correlating and fusing the data from homogeneous and heterogeneous
10443 sensor agents spread throughout the GIG. In this regard, standards should be developed to reflect
10444 automated agent-based change mechanisms.

10445 (U) Common Vulnerabilities and Exposures (CVE) is a list or dictionary that provides common
10446 names for publicly known vulnerabilities and information security exposures. CVE standardized
10447 the names for all publicly known vulnerabilities and security exposures.

10448 (U) Open Vulnerability Assessment Language (OVAL) is the common language for security
10449 experts to discuss and agree upon technical details about how to check for the presence of
10450 vulnerabilities on computer systems. OVAL queries are based primarily on the known
10451 vulnerabilities identified in CVE.

10452 **2.6.3.4.5 (U) Cost/Limitations**

10453 (U//FOUO) Mapping and vulnerability tools themselves represent minor costs. However, the
10454 implementation of an agent-based discovery technology, including the capability for automated
10455 vulnerability repair, represents both costs and limitations based on acceptance of allowable
10456 command functions. Further, although agent-based solutions require greater deployment labor,
10457 they ultimately reduce operating cost by enabling near-continuous monitoring and potential Alert
10458 and Change.

10459 **2.6.3.4.6 (U) Dependencies**

10460 (U//FOUO) To be effective in a large and ever changing environment, the preferred discovery
10461 approach needs to be both fast and focused. It should interpret the initial conditions and
10462 transpose these conditions through a scalable interface to the system administrator. Discovery
10463 tools should also have the inherent capability to interface with other tools and agent-based
10464 architectures at local host and hierarchal levels so more advanced discoveries or controls can
10465 take place. The ultimate goal would be to identify and then correct the identified deficiencies.
10466 Such response tools are dependent on both accurate discovery and automated decision making.

10467 (U//FOUO) The GIG's Net-Centric Operations Warfare (NCOW) development problem related
10468 to vulnerability discovery is best viewed not just as communications, networking, information
10469 assurance, and knowledge dissemination problem. Rather, due to size and complexity, it should
10470 be viewed primarily as an artificial intelligence (AI) problem. Here, artificial intelligence is
10471 defined as:

10472 (U) A collection of algorithms and their attendant infrastructure
10473 organized to automate a decision-making process.

10474 (U//FOUO) The GIG will need a ubiquitous AI architecture to address the discovery problem.
10475 The key expected AI building block, an Agent Functional Stack (AFS), is a collection of
10476 specialized intelligent agents organized into layers, thus providing services to each other. These
10477 should be used for managing IA services at the enclave initially and in the future at a COI.

10478 **2.6.3.4.7 (U) Alternatives**

10479 (U//FOUO) Many network mapping programs are designed to automatically discover a local
10480 network. They use SNMP or pings to identify network devices and work out how they are
10481 physically connected together. The network is then presented as a topology diagram with simple
10482 integrated monitoring. Changes in the network are reflected in the diagram that continuously
10483 updates and are usually customizable to provide various views of the network map. Some of
10484 these tools identify the characteristics of the links as well as the various devices, including their
10485 operating system, primary use, and some even what ports they have open. While current
10486 graphical network monitoring can be a useful management tool for system administrators, active
10487 monitors can become bandwidth intensive when used in enterprise-level applications.

10488 (U//FOUO) A probe usually implements a portion of the device's protocol; if the device does not
10489 answer properly, the mapper will indicate the device as down, or in the alarm or warning state.
10490 Devices on the network are usually shown as various figures on a map. In general, each figure
10491 represents a piece of network equipment (such as a router, switch, or hub), a workstation, a
10492 database, or a server.

10493 **2.6.3.4.8 (U) Complementary Techniques**

10494 **2.6.3.4.9 (U) References**

10495 (U) "Introduction to Vulnerability Scanning," by Tony Bradley, Internet/Networking Security,
10496 <http://netsecurity.about.com/cs/hackertools/a/aa030404.htm>.

10497 (U) <http://www.cve.mitre.org/>

10498 (U) <http://www.us-cert.gov/oval/about.html>

10499 **2.6.3.5 (U) Intrusion Detection Systems**

10500 **2.6.3.5.1 (U) Technical Detail**

10501 (U) Due to the ubiquitous nature of the Internet and local networks today, organizations have
10502 seen an increase in the number of systems being implemented that can monitor against the
10503 growing number of intrusion events and security breaches. While IDSs are primarily concerned
10504 with the detection of hostile actions, they can, in some cases, even initiate a series of actions in
10505 response to an intrusion or attack.

10506 (U) The two primary types of IDSs are:

- 10507 • (U) Host-Based IDS – derives its source of information from a single host or system
- 10508 • (U) Network-Based IDS – derives its source of information from a whole segment of a
10509 local network

10510 **2.6.3.5.1.1 (U) Host-Based IDS (HIDS)**

10511 (U) A HIDS resides on a single computer and provides protection for that specific computer
10512 system. This allows HIDS to analyze activities with great reliability and very fine granularity.
10513 HIDS are essential in monitoring systems that reside on high-speed networks and in networks in
10514 which encryption is used. As HIDS are used in more critical locations, their ability to both detect
10515 and withstand attacks must also increase. By implementing a HIDS within the kernel layer,
10516 detection can be placed closer to the root operation that compromises a system. This improves
10517 the ability of a HIDS to detect both known and unknown attacks. Implementation within the
10518 kernel can also help maintain the robustness of the HIDS itself by having it run within protected
10519 space where it cannot be easily modified or subverted.

10520 (U) The current state of technology for kernel-layer HIDS shows an increasing emergence of
10521 COTS products. Most of these products are agent-based solutions. They intercept system calls
10522 between applications and the kernel, but do not run within the context of the kernel. This would
10523 make them more vulnerable to mimicry attacks and attacks against their availability.

10524 **2.6.3.5.1.2 (U) Network-Based IDS (NIDS)**

10525 (U) The majority of commercial intrusion detection systems are network-based. These IDSs
10526 detect attacks by capturing and analyzing network packets. By listening on a network segment or
10527 switch, one NIDS can monitor the network traffic affecting multiple hosts that are connected to
10528 the network segment, thereby protecting those hosts. The need to work online with encrypted
10529 networks destined to a single host has seen the introduction of what some consider a separate
10530 class of intrusion detection systems, known as Network Node IDS (NNIDS). NNIDS are a blend
10531 of HIDS and NIDS, with agents deployed on every host within the network being protected
10532 (typical NIDS uses network agents to monitor whole LAN segments). Most of the large intrusion
10533 detection systems that are commercially offered today merge the strengths of HIDS and NIDS
10534 into a unique concept.

10535 (U) There are generally two different analysis approaches of IDSs: misuse detection and
10536 anomaly detection.

- 10537 • (U) Misuse Detection - Misuse detection techniques attempt to model attacks on a system
10538 as specific patterns, and then systematically scan the system for occurrences of these
10539 patterns. This process involves a specific encoding of previous behaviors and actions that
10540 were deemed intrusive or malicious. Misuse detectors analyze system activity, looking
10541 for events or sets of events that match a predefined pattern of events that describe a
10542 known attack. As the patterns corresponding to known attacks are called signatures,
10543 misuse detection is sometimes called signature-based detection. The most common form
10544 of misuse detection used in commercial products specifies each pattern of events
10545 corresponding to an attack as a separate signature. However, there are more sophisticated
10546 approaches called state-based analysis techniques that can leverage a single signature to
10547 detect groups of attacks. After revamping the commercial IDSs with signatures which
10548 reflect generic attack classes, we seem to be in a very good position to detect incoming
10549 attacks through content examination.
- 10550 • (U) Anomaly Detection - Anomaly detection approaches attempt to detect intrusions by
10551 noting significant departures from normal behavior. Anomaly detectors identify abnormal
10552 unusual behavior (anomalies) on a host or network. They function on the assumption that
10553 attacks are different from normal (legitimate) activity and can therefore be detected by
10554 systems that identify these differences. Anomaly detection, while initially attractive, has
10555 yet to show any promise due to the large number of false alarms that are created.
10556 Although some commercial IDSs include limited forms of anomaly detection, few, if any,
10557 rely solely on this technology. The anomaly detection that exists in commercial systems
10558 usually revolves around detecting network or port scanning. However, anomaly detection
10559 remains an active intrusion detection research area and may play a greater part in future
10560 IDSs.

10561 **2.6.3.5.2 (U) Usage Considerations**

10562 **2.6.3.5.2.1 (U) Implementation Issues**

10563 (U) Current systems require full content examination. However, metadata-based detectors have
10564 shown promise in handling scans and large-scale worm activities. This means that detection is
10565 still very much content-oriented and that future detectors must continue to be able to handle full
10566 content examination.

10567 (U) Physical limitations dominate. Sufficient cooling, power, and rack space requirements are
10568 the driving factors. Load balancing is also a must as detectors have fixed bandwidth limitations.

10569 (U) There are serious concerns about deployment of NIDS and HIDS in the GIG. The current
10570 architectures used by DISA and the services NIDS may not be compatible with the Black Core
10571 concept and may not scale well. Further, there is little protection at the enclave level today.
10572 HIDS are rarely used, if at all, and planning and deployment in the GIG requires significant
10573 architectural work. Subsequently, the integration of many NIDS and HIDS into the tiers
10574 envisioned for the GIG will require significant architectural work, standards development, and
10575 technology development.

10576 **2.6.3.5.2.2 (U) Advantages**

10577 (U) Host-based IDSs, with their ability to monitor events local to a host, have the advantage of
10578 being able to detect attacks that cannot be seen by a network-based IDS. Also, host-based IDSs
10579 can often operate in an environment in which network traffic is encrypted by gathering
10580 information before data is encrypted or after the data is decrypted at the destination host.

10581 (U) The advantages of misuse/signature detection methods are:

- 10582 • (U) Lower false alarm rates
- 10583 • (U) Simple algorithms
- 10584 • (U) Easy creation of attack signature databases
- 10585 • (U) Easy implementation
- 10586 • (U) Typically minimal system resource usage

10587 (U) The advantages of anomaly detection methods are:

- 10588 • (U) Possibility of detection of novel attacks
- 10589 • (U) Anomalies are recognized without specific knowledge of details
- 10590 • (U) Ability to detect abuse of user privileges
- 10591 • (U) Ability to produce information that can in turn be used to define signatures for
10592 misuse detectors

10593 **2.6.3.5.2.3 (U) Risks/Threats/Attacks**

10594 (U) A risk exists in the fact that current commercial IDSs do not detect novel attacks, nor do they
10595 catch most novel variations of attacks. This is a significant technology gap in the IDS technology
10596 area and calls for more research and development.

10597 (U) Furthermore, an important distinction needs to be made between the terms: detection and
10598 recognition. For signature-based systems, there is very little difference between the two—
10599 detection means that an attack is recognized, at least for those systems with very low false-
10600 positives. However, the same is not true for anomaly-based systems. Here detection data that has
10601 been recorded may not result in a report or recognition, but still be analyzed more deeply at a
10602 later time. Misuse detectors do not report or record near misses and so the only time the detection
10603 data is available is when an attack is recognized.

10604 **2.6.3.5.3 (U) Maturity**

10605 (U) While much IDS research is underway, commercial IDSs are still in their formative years.
10606 The negative publicity of some commercial IDSs can be attributed to the following:

- 10607 • (U) Large number of false alarms
- 10608 • (U) Awkward control and reporting interfaces

- 10609 • (U) Overwhelming number of attack reports
 - 10610 • (U) Lack of scalability
 - 10611 • (U) Lack of integration with enterprise network management
- 10612 (U) However, there is a strong commercial demand for IDSs that will increase the likelihood of
 10613 these problems being addressed in the near future.

10614 (U//FOUO) The various sub-technologies of the Intrusion Detection System technology area can
 10615 be generally assigned Technology Readiness Level group of Emerging (TRL 4-6).

10616 **2.6.3.5.4 (U) Standards**

10617 (U) Standardization is problematic as we are still dependent on vendor hardware that changes
 10618 from time to time. Still, sensor outputs are standardizing with almost everyone supporting Packet
 10619 Capture (PCAP). The PCAP library provides a high level interface to packet capture systems and
 10620 allows access to all packets on the network.

10621 (U) There are no mature IDS standards at this point in time. The Internet Engineering Task Force
 10622 (IETF) has one working group, the Intrusion Detection Working Group (IDWG), which is tasked
 10623 with defining “formats and exchange procedures for sharing information of interest to intrusion
 10624 detection and response system, and to management systems which may need to interact with
 10625 them.” The standards are listed in Table 2.6-1

Table 2.6-1: (U) Standards for Intrusion Detection Systems

This table is (U)	
IETF Intrusion Detection Working Group (drafts)	
Name	Description
Intrusion Detection Message Exchange Requirements (October 22, 2002)	This Internet-Draft describes the high-level requirements for sharing IDS information. http://www.ietf.org/internet-drafts/draft-ietf-idwg-requirements-10.txt
The Intrusion Detection Message Exchange Format IDMEF (January 8, 2004)	Describes a data model to represent information exported by intrusion detection systems, and explains the rationale for using this model. http://www.ietf.org/internet-drafts/draft-ietf-idwg-idmef-xml-12.txt
The Intrusion Detection Exchange Protocol (IDXP) (October 22, 2002)	Describes the Intrusion Detection Exchange Protocol (IDXP), an application-level protocol for exchanging data between intrusion detection entities. http://www.ietf.org/internet-drafts/draft-ietf-idwg-beep-idxp-07.txt
IETF Incident Handling Working Group (working drafts)	
Distributed Denial of Service Incident Handling: Real-Time Inter-Network Defense (February 28, 2004)	This proposal integrates current incident detection and tracing practices for network traffic, which could be extended for security incident handling. Policy guidelines for handling incidents are recommended and can be agreed upon by a consortium using the defined protocol and extended to each NP's clients. http://www.ietf.org/internet-drafts/draft-moriarty-ddos-rid-06.txt
The Incident Object Description Exchange	Provides implementation guidelines for Computer Security Incident Response Teams (CSIRT) adopting the Incident Object Description Exchange Format

This table is (U)	
Format (IODEF) Implementation Guide (March 9, 2004)	(IODEF). http://www.ietf.org/internet-drafts/draft-ietf-inch-implement-00.txt
This Table is (U)	

10627 (U) Preliminary implementation work is probably possible, though implementations would have
 10628 to change as the standard is finalized. The design involves sending XML-based alerts over an
 10629 HTTP-like communications format. A lot of attention has been paid to the needs of IDS analysis
 10630 and to making the protocol work through firewalls in a straightforward way.

10631 (U) There is also an effort by the ISO's T4 committee to develop an Intrusion Detection
 10632 Framework. The status of that effort is presently unknown, and attempts to gather further
 10633 information have been unsuccessful.

10634 (U) The following Protection Profiles have been evaluated and certified with NIAP
 10635 (<http://niap.nist.gov/cc-scheme/pp/index.html>):

- 10636 • (U) Intrusion Detection System (System) Protection Profile, Version 1.4
- 10637 • (U) Intrusion Detection System (Analyzer) Protection Profile, Version 1.1
- 10638 • (U) Intrusion Detection System (Sensor) Protection Profile, Version 1.1
- 10639 • (U) Intrusion Detection System (Scanner) Protection Profile, Version 1.1

10640 **2.6.3.5.5 (U) Cost/Limitations**

10641 (U) The acquisition of IDS software is not the actual cost of ownership. Additional costs include
 10642 acquisition of a system to run the software, specialized assistance in installing and configuring
 10643 the system, personnel training, and maintenance costs. Personnel to manage the system are the
 10644 largest cost.

10645 (U) Most host-based systems implement common architectures in which a host system works as
 10646 a host agent reporting to a central console. The associated costs of HIDS deployments can vary
 10647 depending on vendor and software versions.

10648 (U) Network-based systems can be deployed as stand-alone hosts with a possible management
 10649 interface or distributed sensors and management console. The cost of commercially available
 10650 sensors varies depending on vendor, bandwidth, and functional capabilities. Management
 10651 consoles can be free or can cost several thousand dollars—also depending on the vendor.
 10652 Additional costs include hardware or back-end databases.

10653 (U) Intrusion detection technology is still evolving. Limitations of IDSs include the following:

- 10654 • (U) Certain types of attacks are still possible which preclude detection at present
- 10655 • (U) Bandwidth is a serious limitation on most hardware

10656 **2.6.3.5.6 (U) Dependencies**

10657 (U) We are still largely dependent on commercial vendors for hardware/basic software
10658 development. Other trends in computing that is believed will affect the form and function of IDS
10659 products include the move to appliance-based IDSs. It is also likely that certain IDS pattern-
10660 matching capabilities will move to hardware in order to handle increased bandwidth.

10661 **2.6.3.5.7 (U) Alternatives**

10662 (U) Government-developed IDSs may be better suited for generic attack class detection.
10663 Currently, systems rely on a commercially developed base which has been optimized to detect
10664 singular attacks.

10665 (U) Current anomaly detection methods have proven inadequate, and therefore prompt new
10666 methods to be researched and tried. Additionally, very little research has been done in the area of
10667 parallel processing of content via Beowulf clusters even though this area shows much promise.
10668 Beowulf clusters are scalable performance clusters that are based on commodity hardware, on a
10669 private system network and with open source software (Linux) infrastructure. Each cluster
10670 consists of PCs or workstations that are dedicated to running high performance computing tasks,
10671 with improved performance being proportional to added machines.

10672 (U) Traffic normalizers, commonly referred to as protocol scrubbers, are inline network devices
10673 that remove protocol ambiguities from network traffic. The primary objective of the traffic
10674 normalizer is to provide a security enhancement that aids in preventing insertion, DoS, and
10675 evasion attempts against IDSs, thereby eliminating a weakness of many IDSs.

10676 **2.6.3.5.8 (U) Complementary Techniques**

10677 (U) As stated earlier, metadata-based detection can take some of the load off content-based
10678 examination. However, in the end, some content must be made available to validate any attack.

10679 (U) When used in conjunction with firewalls and other access control devices, IDSs can bolster
10680 an organization's ability to detect, prevent, and respond to unauthorized access and intrusion
10681 attacks. Firewalls, if positioned within the enclave, can decrease the amount that an IDS is
10682 required to examine. In addition, any number of policy enforcement mechanisms (i.e., guards,
10683 OS/application wrappers, and anti-virus) can become complements to an IDS.

10684 (U) Several other tools exist that are often labeled as intrusion detection products by vendors.
10685 These tools include vulnerability analysis/assessment systems, file integrity checkers, and attack
10686 isolation.

10687 **2.6.3.5.9 (U) References**

10688 (U) "Algorithm-Based Approaches to Intrusion Detection and Response," by Alexis Cort,
10689 16 March 2004.

10690 (U) Beowulf Project Overview, <http://www.beowulf.org/overview/index.html>.

10691 (U) "Defending Yourself: The Role of Intrusion Detection Systems," by John McHugh, Alan
10692 Christie, and Julia Allen, IEEE Software – September/October 2000.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 10693 (U) “FAQ: Network Intrusion Detection Systems, by Robert Graham,”
10694 accessed on 10 March 2004,
10695 <http://www.robertgraham.com/pubs/network-intrusion-detection.html#2.1>.
- 10696 (U) “Intrusion Detection FAQ: What open standards exist for Intrusion Detection?,”
10697 by Stuart Staniford-Chen, 8 April 2000, http://www.sans.org/resources/idfaq/id_standards.php.
- 10698 (U) “Intrusion Detection: Implementation and Operational Issues, by Julia Allen,”
10699 Alan Christie, and John McHugh, IEEE Crosstalk, January 2001.
- 10700 (U) “Intrusion Detection Systems (IDS) Part 2 – Classification; Methods; Techniques,” by P.
10701 Kazienko and P. Dorosz, 15 June 2004, [http://www.windowsecurity.com/articles/IDS-Part2-Classification-](http://www.windowsecurity.com/articles/IDS-Part2-Classification-methods-techniques.html)
10702 [methods-techniques.html](http://www.windowsecurity.com/articles/IDS-Part2-Classification-methods-techniques.html).
- 10703 (U) “Justifying the Expense of IDS, Part One: An Overview of ROIs for IDS,”
10704 by David Kinn and Kevin Timm, Security Focus (Infocus), 18 July 2002.
- 10705 (U) “NIST Special Publication on Intrusion Detection Systems,” by Rebecca Bace and
10706 Peter Mell.
- 10707 (U) “Why is a firewall alone not enough? What are IDSeS and why are they worth having?,” by
10708 Wojciech Dworakowski, 23 July 2004,
10709 [http://www.windowsecurity.com/articles/Why_is_a_firewall_alone_not_enough_What_are_IDSeS](http://www.windowsecurity.com/articles/Why_is_a_firewall_alone_not_enough_What_are_IDSeS_and_why_are_they_worth_having.html)
10710 [es_and_why_are_they_worth_having.html](http://www.windowsecurity.com/articles/Why_is_a_firewall_alone_not_enough_What_are_IDSeS_and_why_are_they_worth_having.html).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

10711 **2.6.3.6 (U) Intrusion Prevention Systems (IPSs)**

10712 **2.6.3.6.1 (U) Technical Detail**

10713 (U) A new category of CND technologies has recently emerged in the COTS environment.
10714 Called Intrusion Prevention Systems (IPSs), these technologies represent the merger of protect
10715 capabilities with intrusion detection capabilities. As technology advanced, it became clear that if
10716 a device knowingly prevented an attack, then it also detected the attack and could alert the
10717 operator in some useful way while also blocking it. Likewise, advanced technology made it
10718 possible to first detect an attack and then protect against it in either a static or dynamic fashion.
10719 One can imagine that the five core CND capabilities of protect, monitor, detect, analyze and
10720 respond could all exist together on one platform as technology continues to mature.

10721 (U) IPSs are considered as the convergence of the fourth generation of firewall and IDS
10722 technologies. IPSs can monitor traffic and decide whether to drop or allow traffic based on
10723 expert analysis. These devices normally work at different areas in the network and can
10724 proactively monitor suspicious activity that may otherwise have bypassed the firewall. A
10725 complete network IPS solution has the ability to enforce traditional static firewall rules and
10726 administrator-defined whitelists and blacklists.

10727 (U) The two main types of IPSs are:

- 10728 • (U) Host-Based IPS – runs software directly on workstations and servers, and can detect
10729 and prevent threats aimed at the local host
- 10730 • (U) Network-Based IPS – monitors from a network segment level, and can detect and
10731 prevent both internal and external attacks

10732 (U) Host-Based IPS (HIPS) - As with HIDS, HIPS relies on agents that are installed directly on
10733 the system being protected and are closely bound to the operating system kernel and services.
10734 This allows system calls to the kernel or APIs to be monitored and intercepted in order to prevent
10735 and log attacks. In addition, data streams and the environment that are specific to a particular
10736 application may be monitored in order to protect against generic attacks for which no signature
10737 exists.

10738 (U) Network-Based IPS (NIPS) - NIPS combines features of a standard IDS, an IPS, and a
10739 firewall. Packets appear at either the internal or external interface and are passed to the detection
10740 engine to determine if the packet poses a threat. Upon detection of a malicious packet, an alert is
10741 raised, the packet is discarded, and the flow is marked as bad. This results in the remaining
10742 packets of that particular TCP session arriving at the IPS device and immediately being
10743 discarded.

10744 (U) In both types of IPSs, attack recognition is usually accomplished by known-attack detection
10745 or anomalous behavior detection.

10746 **2.6.3.6.2 (U) Usage Considerations**

10747 **2.6.3.6.2.1 (U) Implementation Issues**

10748 (U) A number of challenges to implementing an IPS device stem from the inherent nature of
10749 being designed to work in-line, thus presenting a potential choke point and single point of
10750 failure. Performance of the network can be seriously impacted. Increased latency and reduced
10751 throughput could become problematic as IPS devices struggle to keep pace with high speed
10752 networks. A NIPS device must perform much like a network switch and meet stringent network
10753 performance and reliability requirements.

10754 (U) Another potential problem deals with false positives. Although not as critical for an IDS
10755 device, false positives can be far more serious and far reaching for an in-line IPS device. The
10756 result can be a self-inflicted DoS condition.

10757 **2.6.3.6.2.2 (U) Advantages**

10758 (U) The basic advantage of an IPS in comparison to an IDS is the ability to not only detect
10759 attacks, but also to block them. An IPS acts to combine previous single-point security solutions
10760 (i.e., firewalls for access control and IDS for hackers) into a solitary architecture that is capable
10761 of blocking network attacks, intrusions, viruses, malicious code, and spam. For zero-day attacks
10762 where the virus is previously unknown, current IPS technologies can utilize databases of known
10763 protocol weaknesses and anomalous behavior techniques (also known as heuristics) to identify
10764 malicious traffic.

10765 (U) The benefits of deterministic intrusion prevention can be summarized as:

- 10766 • (U) Proactive protection from the network security infrastructure
- 10767 • (U) Operational efficiencies due to reduced need to react to event logs for protection
- 10768 • (U) Increased coverage against packet attacks and zero-day attacks

10769 **2.6.3.6.2.3 (U) Risks/Threats/Attacks**

10770 (U) Few barometers exist to provide an indication as to how much software or tools are needed
10771 to protect an organization's systems. Another risk is the false sense of confidence within an
10772 organization once IPS is deployed. Without adequate training of personnel and proper
10773 implementation and maintenance by service providers, the IPS remains at risk.

10774 **2.6.3.6.3 (U) Maturity**

10775 (U) As stated previously, the IPS technology is new and emerging. While IPSs represent a
10776 significant advancement over its predecessors, the IPS technology is just beginning to evolve and
10777 gain acceptance in industry. A recent trend of consolidation within the IPS industry has been
10778 observed, and shows no signs of slowing. The aim of these mergers is to acquire capabilities that
10779 can be re-branded into a resultant technology that is marketable as a new form of IPS.

10780 (U//FOUO) The maturity of the various sub-technologies of the Intrusion Prevention System
10781 technology area is rated Early (TRL 1-3).

10782 **2.6.3.6.4 (U) Standards**

10783 (U) Due to the sensitive nature of most security events, the information will have to be
10784 sufficiently abstracted and shared via a standardized protocol. The current best candidate is the
10785 Intrusion Detection Message Exchange Format (IDMEF). Participating organizations would be
10786 able to exchange security-related information (i.e., new protocol anomaly patterns and worm
10787 outbreaks). Although a handful of collections have currently embarked on this endeavor,
10788 adoption in production systems is sparse thus far.

10789 **2.6.3.6.5 (U) Cost/Limitations**

10790 (U) As with any new emerging technology, IPS can be widely categorized as relatively
10791 expensive. Costs include the initial investment in IPS devices, as well as continual costs of IPS
10792 upgrades, maintenance, training, and management.

10793 (U) One potential limitation of HIPS is that given the tight integration with the host operating
10794 system, future OS upgrades could cause problems.

10795 **2.6.3.6.6 (U) References**

10796 (U) "Intrusion Prevention: A White Paper," www.nitroguard.com.

10797 (U) "Intrusion Prevention Systems," by Mike Barkett, July 2004,
10798 <http://www.nfr.com/resource/downloads/SentivistIPS-WP.pdf>.

10799 (U) "Intrusion Prevention Systems (IPS)," January 2004,
10800 http://www.nss.co.uk/WhitePapers/intrusion_prevention_systems.htm.

10801 (U) "Intrusion Prevention Systems (IPS): Next Generation Firewalls,
10802 A Spire Research Report – March 2004," by Pete Lindstrom,
10803 http://www.toplayer.com/generic/Spire_IPS_Whitepaper.pdf.

10804 **2.6.3.7 (U) User Activity Profiling**

10805 **2.6.3.7.1 (U) Technical Detail**

10806 (U//FOUO) There are many challenges in detecting and responding to insider misuse, including
10807 analyzing how insider misuse differs from penetrations and other forms of misuse by outsiders.
10808 The differences among users themselves vary by responsibility and involve either physical
10809 presence and/or logical presence. For example, there may be logical insiders who are physically
10810 outside and physical insiders who are logically outside. Technology solutions for user profiling
10811 are primarily focused on activity monitoring those insiders who are both logical and physical
10812 users.

10813 (U//FOUO) User activity profiling attempts to learn normal behavior patterns with respect to key
10814 observed or derived features (i.e., resource usage and temporal patterns). There are different
10815 degrees of logical insiders related to the nature of the systems and networks involved, the extent
10816 to which authentication is enforced, and the exact environment in which a user is operating at the
10817 moment.

10818 (U//FOUO) Profiling the behavior of not only individuals, but also the programs they operate can
10819 be a useful reference for detecting potential intrusions against systems. In general, anomaly
10820 detection techniques for profiling program behavior evolve from memorization to generalization
10821 using both host-base and network-based agent structures. These techniques often employ
10822 machine-learning techniques that can generalize from past observed behavior to the problem of
10823 intrusion detection.

10824 **2.6.3.7.2 (U) Usage Considerations**

10825 **2.6.3.7.2.1 (U) Implementation Issues**

10826 (U//FOUO) Detecting insider misuse must rely heavily on user profiling of expected normal
10827 behavior as well as application-specific rules. The goal of monitoring programs is to be able to
10828 detect potential insider misuse by noting irregularities in user activities or program behavior and
10829 without extensive false alarms. These monitors often start from the development of a simple
10830 equality matching algorithm over time, and evolve to a feed-forward back-propagation neural
10831 network for learning program behavior, and finally to approaches for recognizing recurrent
10832 features in activity execution. In order to detect future malicious activities against systems,
10833 intrusion detection systems must be able to generalize from past observable behavior.

10834 (U//FOUO) The validation of profiling systems is problematic, and usually relies on some
10835 variant of cross profiling, wherein fine-grained system measurements for one subject are played
10836 through the trained profile of another. Measurements can include network traffic activity,
10837 identity of current programs being executed, user typing speed, time of day, etc. Typical
10838 measures of detection effectiveness include time to detection and probability of detection for, say
10839 a user typing in a different way than normal or a window of unusual commands being issued.
10840 Unfortunately, this approach cannot be used to make strong claims about effectiveness against
10841 malicious use, but rather about discrimination between examples of use that are, to the best of the
10842 analyst's knowledge, legitimate.

10843 (U//FOUO) For large enterprise environments in which monitoring key strokes are not
10844 considered practical, some effort has been made to use triggers to initiate monitoring, plus
10845 monitor key-stroke dynamics. Triggers are most useful when closely monitored for false alarm
10846 control. Keystroke dynamics tend to be much less reliable in general, particularly when the
10847 differences in a typist's frame of mind or the time of day must be considered.

10848 (U//FOUO) Among the issues in implementing an activity monitor solution are providing a real-
10849 time database relating to physical whereabouts and extending statistical profiling to
10850 accommodate subtle computer usage variants. Further considerations should also take into
10851 account personal behavior such as intellectual and psychological attributes.

10852 (U//FOUO) As an example of an intellectual attribute, consider writing styles. There are already
10853 a few tools for analyzing natural-language writing styles. Profiles of individual-specific
10854 'misspellings,' the frequency of obscenities and the choice of explicit expletives, the relative use
10855 of obscure words, and measures of obfuscation proclivities and meanderings are also useful.

10856 **2.6.3.7.2.2 (U) Advantages**

10857 (U//FOUO) User profiling has long been used with some success to detect masquerader attacks
10858 and insider abuse. In general, profiling can be applied to any process under observation, such as
10859 the system call stream from programs and invites analogy to process control. The basic paradigm
10860 is to alert when the process under observation exhibits behavior that is extremely unusual with
10861 respect to learned norms.

10862 (U//FOUO) As previously discussed, there are generally two types of intrusion detection
10863 systems: misuse detection and anomaly detection. The most significant advantage of misuse
10864 detection approaches is that known attacks can be detected fairly reliably and with a low false
10865 positive rate.

10866 **2.6.3.7.2.3 (U) Risks/Threats/Attacks**

10867 (U//FOUO) Masquerader and insider abuse pose fundamentally different problems than
10868 traditional detection solutions were intended to resolve. The masquerader may be detected by
10869 stylistic differences, while the insider can train his profile so that the eventual exploit appears
10870 normal. The difficulty is exacerbated by the problem of a hit and run attack, where the exploit is
10871 one event in an otherwise normal stream.

10872 (U//FOUO) Another difficult to resolve problem is the false alarm. Even with fine-grained user
10873 profiles, user job functions mature and profiles change over time. There is a serious risk that a
10874 tool's alarm generation capability will be greatly limited to reduce the number of false alarms
10875 being generated.

10876 (U//FOUO) Many Government organizations strongly endorse the use of proprietary COTS IDS-
10877 like software that are unsecure, unreliable, and nonsurvivable. There are few emerging intrusion
10878 detection systems that are completely reliable at detecting hitherto unrecognized insider misuse.
10879 The reality that COTS intrusion-detection tools are not oriented toward insider attacks, unknown
10880 forms of misuse, intelligent results interpretation, and long-term evolution presents a very
10881 significant reason for closer evaluation of GOTS solutions.

10882 **2.6.3.7.3 (U) Maturity**

10883 (U//FOUO) Efforts to date have concentrated on relatively straightforward statistical measures,
10884 thresholds, weightings, and statistical aging of the profiles, independent of particular users. The
10885 basic problem with tools that automatically learn user models from things like what applications
10886 the person uses (order important) and the associated timing information is scalability and
10887 computation time. For this reason, current solutions are limited to enclave-level networks.

10888 (U//FOUO) The maturity of the various sub-technologies of the User Activity Profiling
10889 technology area is rated Early (TRL 1-3).

10890 **2.6.3.7.4 (U) Standards**

10891 (U//FOUO) There are no standards that address this technology need. However, USSTRATCOM
10892 has published a CND Insider Threat Requirements document that addresses basic objectives and
10893 needs for insider threat technology solutions.

10894 **2.6.3.7.5 (U) Cost/Limitations**

10895 (U//FOUO) Simple activity monitor technologies that trigger more large-scale monitoring are not
10896 expensive to deploy. The cost is dependent upon the size of the operational system in which they
10897 are being placed and the maintenance and support cost to operate and manage them.

10898 **2.6.3.7.6 (U) Dependencies**

10899 (U//FOUO) Adequate controls of insider misuse suggest that better system security is necessary
10900 as one part of the solution. There is a fundamental need for better differential access controls
10901 (access control lists, compartmentalized protection, fine-grain roles, etc.). There is also a need
10902 for better user authentication to prevent intruders from gaining insider access and to provide
10903 positive identification of insiders that might diminish their ability to masquerade as other insiders
10904 and to otherwise hide their identities.

10905 **2.6.3.7.7 (U) Alternatives**

10906 (U//FOUO) Personal on-line behavior can also be profiled statistically by extending the analysis
10907 information that is recorded, such as with whom an individual tends to exchange e-mail, which
10908 Web sites are visited regularly, and even what level of sophistication the user appears to exhibit.
10909 This is only a minor extension of what can be done with monitor tools available today.

10910 **2.6.3.7.8 (U) Complementary Techniques**

10911 (U//FOUO) There are a few relative differences in detecting insider misuse compared with
10912 outsider-initiated misuse, but these differences do not seem to be intrinsic. Instead, the
10913 differences might involve the following:

- 10914 • (U) Information to be gathered
- 10915 • (U) Rules given to an expert system
- 10916 • (U) Parameters used to tune the profile-based analysis
- 10917 • (U) Priorities associated with different modes of misuse
- 10918 • (U) Urgency accorded to various responses

10919 (U//FOUO) Some new inference tools might be useful, but they could also be developed
10920 generally enough to be applicable to outsider misuse as well.

10921 **2.6.3.7.9 (U) References**

10922 (U) "CND Insider Threat Requirements, V0.43," USSTRATCOM, August 2004.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

10923 (U) "The Challenges of Insider Misuse," by Peter G. Neumann, Sri Computer Science Lab, Post-
10924 Workshop Version, 23 August 1999, Prepared For The Workshop On Preventing, Detecting,
10925 And Responding To Malicious Insider Misuse, 16-18 August 1999, At Rand, Santa Monica,
10926 California.

10927 (U) "Evaluating Software Sensors for Actively Profiling Windows 2000 Computer Users," by
10928 Jude Shavlik, University of Wisconsin-Madison, Mark Shavlik, Michael Fahland, Shavlik
10929 Technologies, St. Paul, Minnesota.

10930 (U) "Learning Program Behavior Profiles for Intrusion Detection," by Anup K. Ghosh, Aaron
10931 Schwartzbard & Michael Schatz, Reliable Software Technologies Corporation.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

10932 **2.6.3.8 (U) Cyber Attack Attribution**

10933 **2.6.3.8.1 (U) Technical Detail**

10934 (U//FOUO) Approaches in defending network-based intrusions are categorized as intrusion
10935 prevention, intrusion detection, intrusion tolerance, and intrusion response. Response
10936 mechanisms usually take two approaches: localizing the source of the attack using traceback
10937 techniques or reducing the intensity of the attack by blocking attack packets.

10938 (U//FOUO) To hide their identity, network-based intruders seldom attack directly from their own
10939 hosts, but rather from hosts acting as intermediate stepping-stones or zombies. Spoofing the
10940 return address in a one-way communications is also a common practice. In order to identify the
10941 intruder behind these stepping-stones, it is necessary to be able to trace through each
10942 intermediate host and construct the correct intrusion connection chain. Traceback is the term
10943 used to describe the technology for reconstructing the connection chain to the original IP host.

10944 (U//FOUO) There are several different approaches of tracking and tracing attacks via route-based
10945 distributed packet filtering, some of which include:

- 10946 • (U) Hop-by-Hop Traceback
- 10947 • (U) Backscatter Traceback
- 10948 • (U) CenterTrack
- 10949 • (U) ICMP Traceback or iTrace
- 10950 • (U) Hash-Based IP Traceback

10951 **2.6.3.8.1.1 (U) Hop-by-Hop Traceback**

10952 (U) The most common and basic in use today, hop-by-hop traceback traces large, continuous
10953 packet flows that are currently in progress and that originate from spoofed source addresses (i.e.,
10954 DoS packet flood attack). Starting with the Internet Service Provider's (ISP's) router closest to
10955 the victim, an ISP administrator uses the diagnostic, debugging, or logging features of the router
10956 to characterize the nature of the traffic and determine the input link of the attack. The
10957 administrator then moves to the upstream router where the attack packets are coming from. This
10958 diagnostic procedure and trace backwards is repeated—hop-by-hop—until the source of the
10959 attack is ultimately found.

10960 **2.6.3.8.1.2 (U) Backscatter Traceback**

10961 (U) The backscatter traceback technique makes clever use of the large number of invalid source
10962 addresses that are characteristic of contemporary distributed denial-of-service (DDoS) attacks.
10963 Once a DDoS attack has been identified, routers are configured by the ISP to reject all packets
10964 destined for the victim. This will result in a slew of destination unreachable error message
10965 packets or backscatter that can be routed for capture. This technique makes use of the fact that
10966 the Internet Address Naming Authority (IANA) has not allocated several large blocks of IP
10967 addresses for global routing.

10968 **2.6.3.8.1.3 (U) CenterTrack**

10969 (U) The CenterTrack approach improves traceability by adding an overlay network, or auxiliary
10970 network formed from the joining of new physical/logical connections on top of the existing one.
10971 The overlay network is optimized for hop-by-hop tracing and analysis because of having only a
10972 small number of hops between edge routers. Intended DoS flood attack packets can be diverted
10973 to the overlay network which is equipped with special-purpose tracking routers.

10974 **2.6.3.8.1.4 (U) ICMP Traceback or iTrace**

10975 (U) The fundamental concept is that about once in every 20,000 packets, a router sends an ICMP
10976 traceback message (called an iTrace packet) to the same destination address as the sampled
10977 packet (or to an outboard monitor). The destination (or monitor) collects and correlates the
10978 tracking information to successfully trace the attack.

10979 **2.6.3.8.1.5 (U) Hash-Based IP Traceback**

10980 (U) All of the traceback methods described so far have limited capability because each of these
10981 techniques requires a large amount of attack traffic to support tracking. Arguably the most
10982 promising new research approach, Hash-Based IP Traceback (also known as Single-Packet IP
10983 Traceback) offers the possibility of making feasible the traceback of single IP packets. The
10984 fundamental idea is to store highly compact representations of each packet rather than the full
10985 packets themselves. These compact representations are called packet digests and are created
10986 using mathematical functions called hash functions. Hash-based IP traceback uses a system
10987 known as Source Path Isolation Engine (SPIE).

10988 (U//FOUO) Using a timing and marking approach, current research has been able to develop a
10989 partial solution to the traceback problem. The ARDA Footfall Project at North Carolina State
10990 University is currently evaluating a new method of embedding that works in real-time and
10991 spreads the delay across all the packet pairs selected. The method is based on actively
10992 watermarking the traffic timing so that traffic can be correlated across stepping stones, or
10993 intermediate hosts, and through the network. The basic idea is to manipulate the timing in such a
10994 way that the traffic is uniquely recognizable by an analysis program. Watermarking techniques
10995 create a method of traceback that is almost arbitrarily robust to attempts by attackers to perturb
10996 traffic timing to avoid traceability. It is expected that the approach developed through the
10997 Footfall Project will be the first deployable partial solution on DoD networks. This technology
10998 transition should take place by the end of 2005. Therefore, the integration of a partial traceback
10999 solution on the DoD network will take place before GIG Technology increment 1.

11000 **2.6.3.8.2 (U) Usage Considerations**

11001 **2.6.3.8.2.1 (U) Implementation Issues**

11002 (U) Route-based traceback is a very labor-intensive, technical process and often requires
11003 cooperation among bordering ISPs to complete the trace. Routers at each hop will need sufficient
11004 diagnostic capabilities to follow the trace. In addition, as in tracing a phone call by the police, the
11005 attack must remain in progress in order for the trace to be completed back to its origin.

11006 (U//FOUO) There are also policy implications that need to be considered. Careful coordination
11007 needs to be in place as attacks can flow across administrative, jurisdictional, and national
11008 boundaries. Unlike passive defense techniques, active traceback can involve privacy laws. These
11009 laws directly impact automated systems that perform investigations for law enforcement. While
11010 the legal issues prevent Government use of some available commercial systems, private firms
11011 use them to gather information or actively react to network intrusions.

11012 (U//FOUO) The three U.S. Laws that dictate legal considerations are the Electronic
11013 Communications Privacy Act, ECPA (18USC2701), the Wiretap Act (18USC2511), and the
11014 Trap and Trace Act (18USC3121). Since these laws were not written directly to protect against
11015 computer network crime, additional case law and interpretation is necessary to determine their
11016 exact relationship to traceback.

11017 (U//FOUO) There are also some less defined areas within the statutes themselves. Specifically,
11018 techniques that involve some form of content monitoring or fingerprinting may violate privacy
11019 issues. Privacy protects the original packet contents, not a digested metric of the packet itself.
11020 Additionally, there are distinctions made between collecting and disclosing to others, voluntary
11021 versus non-voluntary collection, and stored access versus real-time access. The bottom line is
11022 that a traceback technology solution could violate the law under some conditions.

11023 (U//FOUO) Unlike actual adversaries, legal restrictions and the rights of U.S. citizens limit the
11024 capabilities of Defensive Information Operations (DIO) services. For example, a Red Team
11025 cannot target public domain servers being used as avenues to place malicious code on DoD
11026 hosts. However, real adversaries do target and exploit public domain servers at will. Also, even if
11027 all legal restraints were lifted, robust tools were developed, and additional defensive resources
11028 were available, the ability to respond to attacks would still be challenged by political
11029 considerations based on adversarial relationships.

11030 **2.6.3.8.2.2 (U) Advantages**

11031 (U) Backscatter traceback is a fast and efficient method of countering DDoS flood attacks.

11032 (U) An advantage of the CenterTrack approach is that special-purpose tracking and analysis
11033 features are not needed on all routers, but only on the edge routers and those for special-purpose
11034 tracking.

11035 (U) All of the probabilistic traceback approaches depend on auditing very sparse samples of
11036 large packet flows and thus are well suited for attacks that generate massive packet flows, such
11037 as DDoS floods.

11038 **2.6.3.8.2.3 (U) Risks/Threats/Attacks**

11039 (U) Hop-by-hop traceback of DDoS attacks can be adversely affected due to resource
11040 consumption of bandwidth and processing power in the network by the DDoS attack itself.

11041 (U) Backscatter traceback is heavily dependent upon specific characteristics of DDoS attacks it
11042 was defined to defeat. Like many other approaches designed to work against DDoS flood attacks,
11043 its success depends on a large number of attack packets being directed to a victim and is
11044 therefore, not as effective to subtle attacks. As attack methodology continues to advance (i.e.,
11045 DDoS attack tool that uses randomly selected IP address from valid IANA allocation), the
11046 backscatter traceback technique will eventually be defeated. In addition, attacks that do not forge
11047 zombie source addresses would also be able to defeat this technique.

11048 (U) The CenterTrack approach fails when an attack originates inside an ISP's network. In
11049 addition, high scalability is uncertain for DDoS attacks with many entry points into the ISP's
11050 network.

11051 (U) The iTrace approach can be defeated or disrupted by sending spoofed iTrace packets.
11052 Therefore, iTrace packets must include an authentication field.

11053 **2.6.3.8.3 (U) Maturity**

11054 (U//FOUO) DoD organizations investigating attacks currently use manual techniques. There is
11055 no current automated solution to traceback. Existing approaches have focused on identifying the
11056 set of correlated connections in the connection chain. These approaches have overlooked the
11057 serialization of those correlated connections, thus providing an incomplete solution (Wang,
11058 March 2004).

11059 (U//FOUO) The maturity of the various sub-technologies of the Cyber Attack Attribution
11060 technology area is rated Early (TRL 1-3).

11061 **2.6.3.8.4 (U) Standards**

11062 (U//FOUO) One emerging standard that will help—but not solve the traceback problem—is
11063 implementation of the IPv6 protocol. Another standard that could significantly reduce the
11064 problem would be requiring all routers to place their own unique ID in the protocol of each
11065 packet they receive. The drawback to this approach is that the routing overhead would increase
11066 greatly, and all existing hardware would need to be replaced. One technique that uses a query
11067 approach between routers employs the Intrusion Detection and Isolation Protocol (IDIP)
11068 developed through a Defense Advanced Research Projects Agency (DARPA) project. At this
11069 time no standard is being promoted for resolving the traceback gap area.

11070 **2.6.3.8.5 (U) Cost/Limitations**

11071 (U//FOUO) Route-based distributed packet filtering for attack prevention and traceback, has
11072 been widely studied. Tracing IP packets with forged source addresses requires complex and often
11073 expensive techniques to observe the traffic at routers and reconstruct a packet's real path
11074 traveled. Tracing becomes ineffective when the volume of attack traffic is small or the attack is
11075 distributed.

11076 (U//FOUO) Currently available Traceback tools that can be used by DoD are primarily
11077 Government-off-the-Shelf (GOTS). Additionally, there are a limited number of authorized
11078 organizations that can use these tools.

11079 (U) Policy implications can limit the tracing of attacks that go beyond administrative,
11080 jurisdictional, and international boundaries and will most likely depend upon the trustworthiness,
11081 cooperation, and skill of other ISPs.

11082 (U) For the CenterTrack approach, an increase in the overall complexity can result in operational
11083 errors (i.e., routing updates). Also, the overhead inherent in creating IP tunnels could amplify a
11084 DoS flood's negative effects on the network.

11085 **2.6.3.8.6 (U) Dependencies**

11086 (U) International agreements will need to be established in order to formalize the cooperation
11087 needed to make the techniques effective. This may need to include agreements to share traceback
11088 technology if the overall level of skill needed to complete a trace is not sufficient.

11089 **2.6.3.8.7 (U) Alternatives**

11090 (U//FOUO) Within DoD, the alternatives to traceback using traditional techniques form the basis
11091 of the currently deployed Defense-in-Depth approach. Until deployable automated traceback can
11092 be developed, only defensive approaches and manual techniques are available.

11093 **2.6.3.8.8 (U) Complementary Techniques**

11094 (U) Other research works such as various intrusion detection models, data mining-based models,
11095 and IDSs are complementary to the aforementioned traceback techniques.

11096 **2.6.3.8.9 (U) References**

11097 (U) "Advanced and Authenticated Marking Schemes for IP Traceback,"
11098 by Dawn Song and Adrian Perrig, University of California Berkeley,
11099 DARPA Research Project N6601-99-28913.

11100 (U) The Footfall Project, <http://footfall.csc.ncsu.edu/index.htm>.

11101 (U) "A Little Background on Trace Back," CSC 774 Network Security, Spring 2003,
11102 <http://discovery.csc.ncsu.edu/~pning/Courses/csc774/on-trace-back.pdf>.

11103 (U) "The Loop Fallacy and Serialization in Tracing Intrusion Connections through Stepping
11104 Stones," by Xinyuan Wang, North Carolina State University,
11105 SAC' 04, March 14-17, 2004, Nicosia, Cypress.

11106 (U) "Technical, Legal, and Societal Challenges to Automated Attack Traceback,"
11107 by Susan Lee and Clay Shields, Technical, ITPro, May/June 2002.

11108 (U) "Tracking and Tracing Cyber Attacks: Technical Challenges and Global Policy Issues," by
11109 Howard F. Lipson, CMU/SEI-2002-SR-009, November 2002,
11110 <http://www.cert.org/archive/pdf/02sr009.pdf>.

11111 **2.6.3.9 (U) Correlation Technologies**

11112 **2.6.3.9.1 (U) Technical Detail**

11113 (U//FOUO) Correlation technologies are tools that provide the capabilities to perform data
11114 aggregation, correlation, reduction, and analysis. With the widespread integration of security
11115 solutions such as intrusion detection and protection/prevention systems into the global networked
11116 environment, comes an increased need to implement tools that provide for the management of
11117 the data collected by these systems.

11118 (U//FOUO) Many security solutions generate enormous quantities of data. It has become
11119 necessary to use applications to perform the comprehensive analysis necessary to correlate
11120 security event data in a timely (real-time/near real-time) manner. The analysis of this data allows
11121 for the identification of the anomalies and trends that are buried within the data. These events
11122 must then be displayed and reported in the most comprehensive method possible in order to
11123 respond immediately to an event.

11124 (U//FOUO) The GIG architecture calls for a significant increase in network bandwidth
11125 throughout the entire system from the core to the remote wireless endpoints. As network
11126 bandwidth increases, the job of CND becomes more challenging. Both the volume of packets
11127 inspected by CND technologies and the number of alerts generated by the CND tools increase
11128 tremendously. For this reason alert correlation becomes increasingly important through each of
11129 the GIG IA increments.

11130 (U) As stated by Haines et al:

- 11131 • (U) [Correlation] systems take as input the output produced by low-level sensors such as
11132 intrusion detection systems, firewalls, and integrity checkers. Correlators issue reports
11133 that group together related alerts and events to provide an improved understanding of a
11134 suspected cyber attack and to help analysts identify and dismiss false alarms. Human
11135 administrators use these reports to understand the state of their network and select an
11136 appropriate response
- 11137 • (U) The goal of correlation is to provide high-level reasoning beyond low-level sensor
11138 capabilities

11139 (U//FOUO) As a data analysis tool, the correlation tool pulls attack, reconnaissance, and log data
11140 from a number of sources (e.g., network and computer sensors, NIDS, HIDS, firewalls, packet
11141 filtering routers, and vulnerability assessment tools). It also normalizes data from stovepipe
11142 systems, correlates, prioritizes, and reduces that data. Using the normalized data, the tool
11143 generates graphical representations of data and generates reports. The normalized data can then
11144 be used later for forensics analysis. The data presented in the reports would trigger the active
11145 response capability to provide immediate mitigation to a highly destructive event.

11146 (U//FOUO) In the current state of the art, security vulnerability analysis tools consider individual
11147 vulnerabilities independent of one another. Moreover, they analyze single machines only, in
11148 isolation from other machines in the network. However, the interdependency of vulnerabilities
11149 and the connectivity of a network make such analysis incomplete. While a single vulnerability
11150 itself may not pose a significant direct threat to a system, a combination of vulnerabilities may.
11151 Thus even well administered networks are vulnerable to attacks, because of the security
11152 ramifications of offering a variety of combined services. That is, services that are secure when
11153 offered in isolation nonetheless render the network insecure when offered simultaneously.

11154 (U//FOUO) Many current tools address vulnerabilities in isolation and in the context of a single
11155 host only. This can be extended by searching for sequences of interdependent vulnerabilities,
11156 distributed among the various hosts in a network. This approach is called Topological
11157 Vulnerability Analysis (TVA).

11158 (U) Correlation tools include components to perform data capture (agent), data collection and
11159 storage (manager), organization and tagging (database), and a user interface (console or web-
11160 based). The data being manipulated by the system internally should be encrypted.

11161 **2.6.3.9.2 (U) Usage Considerations**

11162 **2.6.3.9.2.1 (U) Implementation Issues**

11163 (U) As correlation technologies are currently in the research and development stage,
11164 implementation issues have not yet been fully explored. It is expected, however, that there will
11165 be some significant obstacles that must be addressed. For example, some correlation approaches
11166 rely on the sensor's ability to learn what normal network traffic is, and thus develop the ability to
11167 identify and correlate unusual events. If the correlation engine requires knowledge of typical
11168 adversary behavior, this too must be analyzed, tailored to the specific network segment, and
11169 incorporated into the system. If the correlation engine requires knowledge of the network
11170 architecture or vulnerabilities, the capability to readily include this information, preferably in a
11171 mostly automated manner, must be integrated.

11172 (U) Intrusion detection on an encrypted network in itself presents significant challenges that
11173 must be addressed before the next step of correlation can be taken.

11174 (U) Implementing a collective set of correlation technologies, rather than a single one, to further
11175 enhance analysis capabilities has significant cost, integration, maintenance, and management
11176 implications.

11177 **2.6.3.9.2.2 (U) Advantages**

11178 (U) The advantage to correlating alert information, as opposed to having teams of analysts
11179 digging through voluminous near-raw alert data, is significant as the bandwidth of the GIG
11180 increases. It will not be practical to rely on pure human analysis of this data in the future. It is
11181 critical to CND to have the ability to reduce the overall volume of alert information, as well as
11182 correlate similar alerts, disparate alerts, alerts detected by a variety of sensor systems, and alerts
11183 collected on a variety of different network systems. It will be important to be able to correlate
11184 alerts across different tiers within the GIG architecture. It will be critical to have this information
11185 available to the key decision makers at all levels within the GIG in near-real time. And,
11186 eventually, the ability to include mission priorities in the correlation process will put the CND
11187 analyst in a position to be proactive about protecting the mission rather than reactive.

11188 (U) With the assistance of correlation technologies, the analyst is better able to quickly assess a
11189 current status of the network by focusing on manageable information sets. With the assistance of
11190 advanced visualization tools, this process is further enhanced. From this information, decisions
11191 on response actions can be made and implemented. For future iterations of correlation
11192 capabilities, it is desirable to overlay mission priorities on the correlation analysis to see if the
11193 mission is targeted or impacted as a result of a malicious network event, or if response actions
11194 will impact the mission in an undesirable manner.

11195 **2.6.3.9.2.3 (U) Risks/Threats/Attacks**

11196 (U) There are three key risks:

- 11197 • (U) The first is the user's ability to trust that the data has been correlated accurately
- 11198 • (U) The second is the ability to trust that the correlation process has not dropped key
11199 alerts
- 11200 • (U) The third to the ability to trust that the correlation process has not developed false
11201 positives

11202 (U) The only way to address these risks is to continue to invest in correlation research and
11203 development to improve these systems.

11204 (U) It is conceivable that an adversary could try to distract a correlation system by intentionally
11205 triggering alerts and hiding the real attack traffic in the subsequent smokescreen. This is
11206 something to be addressed by the research community.

11207 **2.6.3.9.3 (U) Maturity**

11208 (U) As previously stated, correlation technologies are currently in the research prototype stage.
11209 There are no advanced correlation technologies available off-the-shelf today. While some COTS
11210 sensors have limited data reduction capabilities, such as reducing the individual alerts due to a
11211 scan, true analytical correlation with disparate alerts is not commercially available. However,
11212 alert correlation has been the subject of recent research with proof of concept technologies
11213 currently being explored showing promising results. Several of these are referenced below.

11214 (U) Research has shown that the combination of different correlation technologies, rather than a
11215 single technology, can yield even better results. This allows the disparate systems to focus on
11216 their strengths, and compensate for one another's weaknesses.

11217 (U//FOUO) The maturity of the various sub-technologies of the Correlation technology area is
11218 rated Early (TRL 1-3).

11219 **2.6.3.9.4 (U) Standards**

11220 (U) There are no correlation standards at this time.

11221 **2.6.3.9.5 (U) Cost/Limitations**

11222 (U) Correlation technology cost is unknown at this time. However, one can assume that it would
11223 cost in the range of an advanced IDS. Some advanced IDSs will include correlation capabilities.
11224 Costs associated with the manpower to monitor the systems can be a limitation depending on the
11225 number of sensors being managed/monitored per analyst and the volume of data collected.

11226 **2.6.3.9.6 (U) Dependencies**

11227 (U) Correlation systems rely in total on the alert information that it can access. It is absolutely
11228 essential for advanced accurate sensors to precede the implementation of correlation
11229 technologies. These sensors must be effective in detecting malicious activity on encrypted
11230 network segments.

11231 (U) Correlation systems also depend on the ability to display the correlated results. While some
11232 systems can generate reports or visual aids, much work can be done to improve current
11233 prototypes. Ideally, correlation results would be fed into a complete situational awareness picture
11234 for further analysis.

11235 **2.6.3.9.7 (U) Alternatives**

11236 (U) The alternative to correlating alert information is to simply increase the overall assurance of
11237 a network and prevent attacks from the outset. Since this is clearly unrealistic, the remaining
11238 alternative is to rely on human analysis to draw the correlation relationships. This would be a
11239 significant challenge with every increasing bandwidth, and the sheer volume of network
11240 components that must be monitored.

11241 **2.6.3.9.8 (U) Complementary Techniques**

11242 (U) Again, human analysts can correlate information manually to some degree. However, these
11243 capabilities can be improved upon significantly with the proper use of computing, mathematical,
11244 and modeling power.

11245 **2.6.3.9.9 (U) References**

11246 (U) "Adaptive, Model-Based Monitoring for Cyber Attack Detection," by A. Valdes, K Skinner,
11247 Recent Advances in Intrusion Detection (RAID 2000), pp. 80-92.

11248 (U) "A Mission-Impact-Based Approach to INFOSEC Alarm Correlation," by P. Porras,
11249 M. Fong, A. Valdes, Proceedings Recent Advances in Intrusion Detection, Zurich, Switzerland,
11250 October 2002, pp. 95-114.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 11251 (U) "Probabilistic Alert Correlation," by A. Valdes, K. Skinner,
11252 Recent Advances in Intrusion Detection (RAID 2001).
- 11253 (U) "Scyllarus Intrusion Detection Report Correlator and Analyzer," by W. Heimerdinger,
11254 DARPA Information Survivability Conference and Exposition, Volume 2, April 2003, pp. 24-26.
- 11255 (U) "The STAT Toolsuite," by G. Vigna, M. Eckmann, R.A. Kemmerer, DARPA Information
11256 Survivability Conference and Exposition, Volume 2, April 2003, pp. 46-55.
- 11257 (U) "Validation of Sensor Alert Correlators," by J. Haines, D. Ryder, L. Tinnel, S. Taylor, IEEE
11258 Security and Privacy, January/February 2003, pp. 46-56.
- 11259 (U) <http://www.sdl.sri.com/programs/intrusion/>.
- 11260 (U) <http://www.cs.ucsb.edu/~rsg/STAT/>.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

11261 **2.6.3.10 (U) CND Response Actions**

11262 **2.6.3.10.1 (U) Technical Detail**

11263 (U//FOUO) U.S. Strategic Command (USSTRATCOM) defines CND Response Actions (CND
11264 RAs) as deliberate, authorized defensive measures or activities that protect and defend DoD
11265 computer systems and networks under attack or targeted for attack by adversary computer
11266 systems/networks. CND RAs extend DoD's layered defense-in-depth capabilities and increase
11267 DoD's ability to withstand adversary attacks.

11268 (U) Response actions are taken as a result of a detected intrusion and can be either automated or
11269 manual—requiring a human in the loop to activate the response. Response actions are
11270 implemented to stop ongoing attacks, such as a denial-of-service, or to plug already exploited
11271 vulnerabilities from future network attack.

11272 (U) Response actions can be construed as counter attack when the action reaches beyond the
11273 GIG controlled assets to target the source of the attack. There are currently notable legal
11274 limitations on such actions.

11275 (U) Response actions can be proactive in nature, updating a security posture based on external
11276 intelligence or other sources or to prioritize mission critical asset protections prior to executing
11277 an operations plan. By the same token, proactive response actions can be targeted against
11278 adversary assets in support of an operation. This action generally falls under the computer
11279 network attack category and will not be discussed further herein.

11280 **2.6.3.10.2 (U) Usage Considerations**

11281 **2.6.3.10.2.1(U) Implementation Issues**

11282 (U) Clearly the ramifications of response actions can be far reaching, especially if the response
11283 does not take into consideration mission priorities. The actions must be well considered, and if
11284 there is time and opportunity, modeling the response in advance of implementing it can be
11285 advantageous. In cases where an active attack must be stopped, it will not be practical to take the
11286 time to do any modeling. In such an instance, an immediate short-term response can be taken,
11287 followed by a well-considered longer-term solution that has undergone analysis, and in some
11288 cases modeling.

11289 (U) While automated response capabilities do exist in a limited capacity in some COTS and
11290 research prototype technologies, automated response is not currently a widely accepted practice.
11291 DoD policies and procedures limit or prohibit an automated response in most cases, and lack of
11292 experience and in-depth knowledge of CND capabilities makes the leadership chain hesitant to
11293 fully trust and use automated engines.

11294 (U) When the technology becomes available, response actions need to be global solutions
11295 coordinated across multiple network enclaves, rather than localized implementations. There are
11296 bound to be significant conflicts resulting in temporary loss of mission critical assets otherwise.

11297 (U) There is discussion between the CND community and the network management community
11298 as to who will actually implement the response actions, whether it is a CND analyst or a network
11299 management operator. As the GIG progresses, the lines between the two groups will continue to
11300 blur, and it will be absolutely critical for both to work hand-in-hand continuously. In many cases
11301 the technologies used to implement the responses will often be the same technologies that either
11302 detected or prevented some portion of the attack. It is impractical to think that a clean hand-off to
11303 the network management group will be possible. Response is also frequently an iterative process
11304 requiring a series of detected and analyzed intrusion detection alerts, followed by more and more
11305 refined response actions.

11306 **2.6.3.10.2.2 (U) Advantages**

11307 (U) Responding to a network attack provides the opportunity for the defenders to stop malicious
11308 network events and prevent the adversary from reaching its goals. Without implementing some
11309 sort of a responsive action, an adversary that has gained unauthorized access will have the luxury
11310 of time to collect further intelligence about the GIG network assets and see a wealth of sensitive
11311 data.

11312 (U) The advantage of automated response is that malicious packets can be stopped within
11313 seconds of being detected. This packet race can be critical in blocking the adversary before more
11314 lethal network attacks are launched. It is not a perfect solution as the adversary will still be at
11315 least one packet ahead of the defenders, and this is particularly critical with the most
11316 sophisticated adversaries that have the one packet, one kill mentality. It is far better to prevent
11317 the attack in the first place than to have to monitor, detect, analyze, and then respond to
11318 unauthorized activity. The shorter the time window between detection and response, the closer
11319 one reaches prevention.

11320 (U) The disadvantage to an automated response, however, is that the impact of the initial
11321 response may not be fully analyzed. This is why the two-tiered response approach provides
11322 additional value. Automated response must be resilient to adversary techniques intended to
11323 trigger it.

11324 (U) Manual response, on the other hand, requires analysis time and human intervention, which
11325 can be slow and sometimes inaccurate. It does, however, allow for manual consideration of the
11326 mission impact and consultation with the appropriate chain of command.

11327 **2.6.3.10.2.3 (U) Risks/Threats/Attacks**

11328 (U) The risk of this technology is that response actions that are not well considered can have a
11329 detrimental impact on mission-critical GIG network functionality. Loss of functionality can be
11330 far reaching and result in significant down time to the user community. A negative impact of this
11331 sort could cause the user community and the chain of command to lack trust, and therefore not
11332 use the response technology, which would leave the networks vulnerable once again.

11333 (U) If the adversary were able to trigger the response technology in some way to also make it
11334 untrustworthy, or to cause an analyst to disable the capability, there would be a negative impact
11335 on the GIG. In this case the technology would actually provide an additional control surface for
11336 the adversary to exploit—something which has been a point of interest in the risk assessment.

11337 **2.6.3.10.3 (U) Maturity**

11338 (U) There are available today a handful of CND technologies with integrated response
11339 capabilities. For example, a commercial DoS discovery technology is able to monitor, and
11340 analyze packets, and once a threshold has been crossed, alert the operator that a DoS has been
11341 detected. The technology then recommends a course of action to block the attack, which can be
11342 implemented either manually or automatically in a neighboring perimeter router.

11343 (U) Response capabilities have been the subject of much research within the DoD, as noted in
11344 the references section below. Research prototypes have been developed, and they show much
11345 promise, especially when paired with sophisticated correlation systems. The science of launching
11346 sophisticated response actions would also benefit tremendously from the capability to include
11347 mission-critical network assets, plans, alternatives, and the notion of timing. Research into
11348 advancing response technologies beyond their present state should yield capabilities and
11349 technologies far greater than what is available today, for both the DoD and its adversaries.

11350 (U//FOUO) The maturity of the various sub-technologies of the CND Response Actions
11351 technology area is rated Early (TRL 1-3).

11352 **2.6.3.10.4 (U) Standards**

11353 (U) There are no current standards for response actions. Any standards for response should be
11354 closely tied to those for intrusion detection.

11355 **2.6.3.10.5 (U) Cost/Limitations**

11356 (U) Response technologies may be integral to other CND technologies, so the cost of the
11357 technology product should be explored as a unit and is expected to be similar to that of other
11358 CND technologies. Research costs to develop response technologies, however, are expected to
11359 be significant.

11360 (U) The usefulness of response technologies will be limited by the ability to centrally manage a
11361 set of devices, and the number of deployed DoD experts available to operate the systems and
11362 make critical and timely decisions involving response actions.

11363 **2.6.3.10.6 (U) Dependencies**

11364 (U) Response technologies are highly dependent on reliable intrusion detection data, which is in
11365 turn dependent upon monitoring and analysis capabilities. Without these, coordinated,
11366 sophisticated response actions will be unattainable. In addition, it will be important for the CND
11367 analyst to have access to reliable and comprehensive situational awareness data in near real time
11368 to make decisions and monitor the effects of response actions. This situational awareness data
11369 should include operational plans and prioritization of mission-critical assets in a time-based
11370 schedule.

11371 **2.6.3.10.7 (U) Alternatives**

11372 (U) The alternative to response technologies is a manual process of analyzing intrusion detection
11373 information and manually updating the security posture based on engineering judgment. This
11374 rudimentary approach will give the adversary the advantage of time. Equally important, the CND
11375 analyst responsible for reviewing the intrusion detection data will be more likely to experience
11376 fatigue, miss critical events, or make mistakes recommending and implementing response
11377 actions.

11378 **2.6.3.10.8 (U) Complementary Techniques**

11379 (U) The only complementary technique to response actions is to constantly evaluate and update
11380 the security posture of the GIG network devices as a result of perceived or known network
11381 threats.

11382 **2.6.3.10.9 (U) References**

11383 (U) "Autonomic Response to Distributed Denial of Service Attacks," by D. Sterne,
11384 K. Djahandari, B. Wilson, B. Babson, D. Schnackenberg, H. Holliday, and T. Reid, Proceedings
11385 of the 4th International Symposium on Recent Advances in Intrusion Detection (RAID), October
11386 2001, pp. 134-149.

11387 (U) "Cooperative Intrusion Traceback and Response Architecture (CITRA),"
11388 by D. Schnackenberg, H. Holliday, R. Smith, K. Djahandari, and D. Sterne, DARPA Information
11389 Survivability Conference and Exposition II, Volume 1, June 2001, pp. 56-68.

11390 (U) "Electronic Quarantine: An Automated Intruder Response Tool," by P. Brutch,
11391 T. Brutch, and U. Pooch, Proceedings of the 1998 IEEE Information Survivability Workshop
11392 (ISW'98), October 1998.

11393 (U) "SARA: Survivable Autonomic Response Architecture," by S. Lewandowski, D. Van Hook,
11394 G. O'Leary, J. Haines, and L. Rossey, DARPA Information Survivability Conference and
11395 Exposition II, Volume 1, June 2001, pp. 77-88.

11396 **2.6.3.11 (U) Automated IAVA Patch Management**

11397 **2.6.3.11.1 (U) Technical Detail**

11398 (U//FOUO) Until recently, patch management had always been a labor and time-intensive ordeal
11399 with little or no support tools. Patch management tools are now available that automate much of
11400 the process, including discovery of reported vulnerabilities and patches, scanning systems for
11401 vulnerabilities and configuration status, assisting in the analysis and decision making process to
11402 decide which patches to deploy and when, testing proposed patches in controlled environments,
11403 deploying patches to systems, and verifying successful patch deployments.

11404 (U//FOUO) Since patch management only addresses software defects that lead to vulnerabilities,
11405 management tools are being integrated into security and vulnerability management tools that can
11406 provide a more complete system management capability. These newer tools reduce the amount
11407 of human intervention now required with current solutions.

11408 **2.6.3.11.2 (U) Usage Considerations**

11409 **2.6.3.11.2.1(U) Implementation Issues**

11410 (U//FOUO) Best practices in patch management indicate that a thorough analysis of proposed
11411 patches must be conducted to assess whether the patch even applies, and if so, to what systems
11412 within the production environment. The potential impacts to those systems must be clearly
11413 understood and evaluated and a priority assigned to mitigating the vulnerability.

11414 (U//FOUO) Vulnerabilities in widely used applications, such as Microsoft's Internet Explorer
11415 (IE), would have high priority because of the number of users, the pervasive use of IE by other
11416 applications, and the severity of the attacks that could be mounted against it. IE is one of those
11417 applications where extensive testing must be performed to understand the impact of the patch in
11418 the production environment. Fixing one security vulnerability problem could cause others to
11419 arise or could cause some functions of IE to stop working.

11420 (U//FOUO) Patches must be implemented quickly to thwart attacks using discovered
11421 vulnerabilities. However, deploying untested patches in a production environment may prove
11422 more costly than the attack. All patches should be thoroughly tested before deployment on as
11423 many of the release configurations as possible. A patch is just that—a quick fix to correct a
11424 functional bug or to counter a security vulnerability. It is not uncommon for a patch that corrects
11425 one problem to cause one or more other problems.

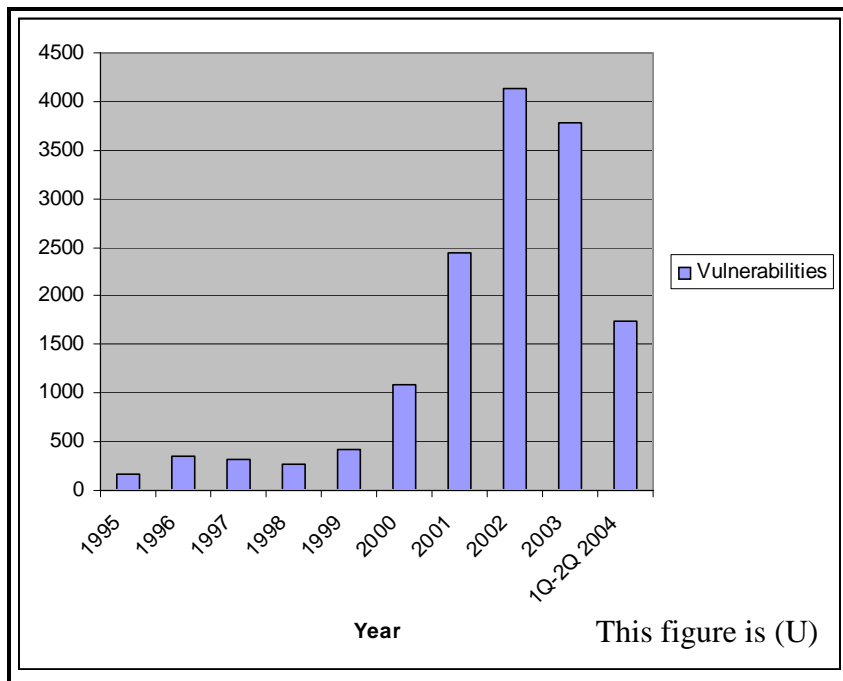
11426 (U//FOUO) Standard software releases should be periodically re-baselined to avoid patches
11427 colliding with each other and to simplify maintaining patch and configuration status.

11428 **2.6.3.11.2.2(U) Advantages**

11429 (U//FOUO) Patch management technologies enable the automation of much of the labor-
 11430 intensive aspects of identifying, analyzing, and deploying patches. As the complexity of network
 11431 systems continues to grow, manually-based patch management techniques quickly demonstrated
 11432 their inability to scale with it. Stand-alone patch management products answer the immediate
 11433 need of businesses to provide some relief in mitigating vulnerabilities. Patch management
 11434 capabilities are being integrated into vulnerability management and system management tools to
 11435 provide security and administration personnel even more automated capabilities.

11436 **2.6.3.11.2.3(U) Risks/Threats/Attacks**

11437 (U//FOUO) Patch management by itself is not a complete security solution. It only addresses
 11438 software defects. It needs to be integrated into a system management capability that includes
 11439 asset inventory, vulnerability, configuration, and policy management. According to the
 11440 vulnerabilities reported from CERT (<http://www.cert.org/stats/>), the number of vulnerabilities that
 11441 must be addressed by the patch management task has steadily increased through 2002 and is only
 11442 slightly tapering off as indicated in Figure 2.6-3: (U) Vulnerabilities Reported from CERT. The
 11443 total vulnerabilities reported (1995-2Q 2004): 14,686.



11444
 11445 **Figure 2.6-3: (U) Vulnerabilities Reported from CERT**

11446 **2.6.3.11.3 (U) Maturity**

11447 (U//FOUO) The maturity of the various sub-technologies of the Automated IAVA Patch
 11448 Management technology area is rated as Emerging (TRL 4-6).

11449 (U) The maturity of patch management systems can be seen in the wide variety of products that
 11450 are currently available. The following are examples of point solution products:

- 11451 • (U) BigFix - BigFix Enterprise – <http://www.bigfix.com>
- 11452 • (U) Ecora - Ecora Patch Manager – <http://www.ecora.com>
- 11453 • (U) PatchLink Corporation - PatchLink Update – <http://www.patchlink.com>
- 11454 • (U) SecurityProfiling - SysUpdate – <http://www.securityprofiling.com/>
- 11455 • (U) Shavlik - Shavlik HFNetChkPro – <http://www.shavlik.com>
- 11456 • (U) St. Bernard Software - UpdateEXPERT – <http://www.stbernard.com>
- 11457 • (U) Microsoft – Software Update Services – <http://www.microsoft.com>

11458 (U) The following are examples of security management products:

- 11459 • (U) Citadel Security Software – <http://www.citadel.com/>
- 11460 • (U) Configuresoft – Enterprise Configuration Manager – <http://www.configuresoft.com>

11461 (U) The following are examples of security configuration management products:

- 11462 • (U) Altiris – Client Management Suite – <http://www.altiris.com/products/clientmgmt/>
- 11463 • (U) LANDesk Software – LANDesk Management Suite – <http://www.landesk.com>
- 11464 • (U) ManageSoft – Security Patch Management –
- 11465 <http://www.managesoft.com/solution/patchmanagement/index.xml>
- 11466 • (U) HP – Novadigm – <http://www.novadigm.com>
- 11467 • (U) Novell (partner with PatchLink) – ZENworks Patch Management –
- 11468 <http://www.novell.com/products/zenworks/patchmanagement/>
- 11469 • (U) Symantec/ON Technology (partner with Shavlik) – iPatch and iCommand –
- 11470 <http://www.on.com>

11471 (U) The following is an example of a vulnerability management product:

- 11472 • (U) Harris Corporation – STAT Scanner – <http://www.stat.harris.com/index.asp>

11473 **2.6.3.11.4 (U) Standards**

11474 (U//FOUO) There are no standards on patch management. Generally, all of the products offer
11475 similar capabilities following a de-facto industry best practice.

11476 **2.6.3.11.5 (U) Cost/Limitations**

11477 (U//FOUO) A variety of options exist for acquiring patch management products and services.
11478 Generally, there is a per seat price with break points at various quantities or an option to acquire
11479 an enterprise-wide license. Most vendors also offer a managed service capability.

11480 **2.6.3.11.6 (U) Dependencies**

11481 (U//FOUO) Patch management systems receive vulnerability and patch information from a
11482 number of industry and Government sources. Continued on-line access to these systems is
11483 required in order to maintain the most current information about patches.

11484 (U//FOUO) An asset inventory of PCs and servers must be established and maintained that
11485 includes an up to date listing of operating system and applications with current patch and service
11486 pack status. The patch management system must periodically scan the PCs and servers to
11487 determine if there have been any changes to the status of the information on file. This status
11488 information is used during the analysis of a newly discovered patch or security vulnerability to
11489 determine which system may be vulnerable, what the likely impact will be to the enterprise, and
11490 what priority should be given to the mitigation of the vulnerability.

11491 **2.6.3.11.7 (U) Alternatives**

11492 (U) Basically, there are two types of patch management architectures available:

- 11493 • (U) Agent-less: Agent-less based approach does not require any special software on the
11494 target machines. This approach typically uses RPC calls to scan machines for status and
11495 to deliver patches. This approach may result in some machines that cannot use such IT
11496 management tools to be patched manually.
- 11497 • (U) Agent-based: Agent-based approach use special software delivered to each target
11498 system to enable communication with the patch server and to perform operations locally
11499 on the targeted machine. This approach typically uses TCP/IP to communicate with the
11500 server and could enable security features such as encryption that may not otherwise be
11501 available. Devices with limited bandwidth may require the use of agent-based software.
11502 Fortunately, vendors are making applications that support both capabilities.

11503 (U//FOUO) Patch management systems are evolving to become an integral part of system
11504 management and vulnerability management applications. A separate patch management
11505 capability may not be needed in the near future.

11506 **2.6.3.11.8 (U) Complementary Techniques**

11507 (U//FOUO) Patch management is not a new concept. It is the evolution from a manual discovery
11508 and mitigation process to partially automated steps, and from discrete patch management tools to
11509 integrated security management tools. These tools include asset management, vulnerability
11510 assessment and management, policy compliance, configuration management, and patch
11511 management.

11512 **2.6.3.11.9 (U) References**

11513 (U) "Get Ready to Patch," by Foley and Hulme, InformationWeek, 30 August 2004,
11514 <http://www.informationweek.com/story/showArticle.jhtml?articleID=45400083>.

11515 (U) "Patch Management is a Fast Growing Market," by Schroder, Colville, and Nicolett, Gartner,
11516 30 May 2003, [http://download.microsoft.com/download/a/2/6/a2625228-9394-4388-8dcf-
11517 de876ccfa88c/Gartner_patch_mgt_fast_growing.pdf](http://download.microsoft.com/download/a/2/6/a2625228-9394-4388-8dcf-de876ccfa88c/Gartner_patch_mgt_fast_growing.pdf).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 11518 (U) "Patch Management Vendor Overview," by Mark Nicolett, Colville, and Silver, Gartner, 27
11519 May 2004.
- 11520 (U) "Patching Things Up, Emerging Technology," CIO Magazine, 1 August 2003,
11521 http://www.cio.com/archive/080103/et_article.html.
- 11522 (U) "Practical Patch Management," NetworkWorldFusion, 21 October 2002,
11523 <http://www.nwfusion.com/supp/security2/patch.html>.
- 11524 (U) "Robust Patch Management Requires Specific Capabilities," by Nicolett and Colville,
11525 Gartner, 18 March 2004, http://www.knowledgestorm.com/sol_summary_63613.asp.
- 11526 (U) "Security Patches: Plugging the Leaks in the Dike," by Karen Krebsbach,
11527 Bank Technology News, August 2004,
11528 <http://www.banktechnews.com/article.html?id=20040802B1GHC6AT>.
- 11529 (U) "SQL Slammer Lesson: Patch Management Is Not Enough," by Mark Nicolett and
11530 John Pescatore, Tech Republic, 2 July 2003,
11531 <http://techrepublic.com.com/5102-6264-5054273.html>.
- 11532 (U) "Taking Control of Vulnerabilities, Citadel Security Software Interview with John
11533 Pescatore," Gartner Research, Interview conducted 27 April 2004,
11534 <http://mediaproducts.gartner.com/gc/webletter/citadel/issue3/gartner1.html>.
- 11535 (U) The Need for Patch Management, Symantec, June 2004,
11536 <http://sea.symantec.com/content/displaypdf.cfm?pdfid=29>.
- 11537 (U) "The Power of Optional Agent Architecture: Advantages of Managing Patches Remotely with
11538 UpdateEXPERT," St. Bernard Software, Inc., 28 July 2003,
11539 <http://www.stbernard.com/products/docs/OptionalAgent.pdf>.
- 11540 (U) "Vulnerability and IT Security Management Are Converging," by Mark Nicolett, Gartner, 10
11541 February 2004.
- 11542 (U) "Vulnerability Management Technology Landscape," by Mark Nicolett, Gartner,
11543 11 September 2003.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

11544 **2.6.4 (U) Network Defense and Situational Awareness: Gap Analysis**
11545 (U//FOUO Table 2.6-2 is a matrix of Network Defense and Situational Awareness technologies
11546 described in previous sections. The adequacy matrix is based upon 2008 capabilities.

**Table 2.6-2:(U) Network Defense & Situational Awareness
Technology Gap Assessment**

11547
11548

This table is (U)																			
Technology Categories																			
			Firewalls (Host)	Firewalls & Filtering Routers (Network)	Virus Protection	Automated Patch Management	Honeypots & Honeynets	Situational Awareness	Vulnerability Scanning	Host-Based IDS	Network-Based IDS	Host-Based IPS	Network-Based IPS	User Activity Profiling	Attack Attribution	Alert Correlation	CND Response	Required Capability (RCD attribute)	
IA Attributes	Protect	Vulnerability Assessment & Reporting	N/A	N/A	N/A	N/A	N/A	N/A		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	IACND6, IACND7	
		Unauthorized/Malicious Activity Prevention					N/A	N/A	N/A	N/A	N/A			N/A	N/A	N/A	N/A		
		Configuration Change					N/A	N/A	N/A	N/A	N/A			N/A	N/A	N/A	N/A	IACND8, IACND9	
	Monitor	Information Monitoring				N/A												N/A	IACND10
		Information Presentation				N/A			N/A						N/A			N/A	IACND11
	Detect	Unauthorized/Malicious Activity Identification				N/A			N/A										IACND12
		Unauthorized/Malicious Activity Reporting				N/A			N/A									N/A	IACND13
	Analyze	Data Reduction & Correlation				N/A	N/A		N/A										IACND14

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This table is (U)																		
Technology Categories																		
			Firewalls (Host)	Firewalls & Filtering Routers (Network)	Virus Protection	Automated Patch Management	Honeypots & Honeynets	Situational Awareness	Vulnerability Scanning	Host-Based IDS	Network-Based IDS	Host-Based IPS	Network-Based IPS	User Activity Profiling	Attack Attribution	Alert Correlation	CND Response	Required Capability (RCD attribute)
		Unauthorized/Malicious Activity Analysis	N/A	N/A		N/A	N/A		N/A									IACND15
		Information Visualization & Sharing	N/A	N/A		N/A	N/A		N/A									IACND17
		Development & Coordination of COAs	N/A	N/A	N/A	N/A	N/A		N/A									IACND16, IACND18, IACND20
		Modeling & Simulation of COAs	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A			IACND19
	Respond	Response Actions			N/A		N/A		N/A				N/A					IACND21, IACND23
		Recovery Actions	N/A	N/A	N/A		N/A		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	IACND22, IACND24
This table is (U)																		

UNCLASSIFIED//FOR OFFICIAL USE ONLY

11550 **2.6.5 (U) Network Defense and Situational Awareness: Recommendations and Timelines**
11551 (U//FOUO) The following recommendations have been identified in the Network Defense and
11552 Situational Awareness Enabler. Without these, the GIG Vision cannot be fully satisfied. The
11553 recommendations are organized in the following categories: Standards, Technology, and
11554 Infrastructure.

11555 **2.6.5.1 (U) Standards**

11556 (U) One or more standards on sensor data are needed to address:

- 11557 • (U) Format of sensor data
- 11558 • (U) Semantics of sensor data

11559 **2.6.5.2 (U) Technology**

11560 (U) It is unlikely that today's protect technologies alone can stop sophisticated stealthy attacks.
11561 In order to raise the bar on the sophisticated risk-averse adversary, tomorrow's protect
11562 technologies must include capabilities such as:

- 11563 • (U) Dynamic protection mechanisms capable of modifying the defensive structure either
11564 on-the-fly as a result of an adverse event or in a proactive organized defensive manner
- 11565 • (U) Adaptive, self-learning capabilities that do not rely on previously known attack
11566 signatures
- 11567 • (U) Ability to successfully protect encrypted network segments. As current protect
11568 technologies are not designed to operate on encrypted network segments, additional
11569 research and development is needed to develop new capabilities and technologies
11570 designed for such an environment.

11571 (U//FOUO) In general, the Situational Awareness technologies represented by the current
11572 capabilities are not scalable to the needs of the GIG. More robust tools are needed to
11573 automatically collect and correlate a variety of information sources and to augment many of the
11574 I&W tasks that are now extremely manpower intensive. Additional processing requiring
11575 automation is the assessment of changes in an adversary's posture and perceived threat intent for
11576 all three levels of the Defense-in-Depth security strategy: computing environment, enclave, and
11577 network.

11578 (U//FOUO) The scalability issue with current correlation tools, the need for collection
11579 capabilities, both at the packet level and from metadata sources on a very large enterprise, and
11580 the need to integrate some form of risk analysis based on current conditions has created several
11581 technology gap areas. These technology areas are currently being researched, and solutions are
11582 expected within the GIG Increment 1 time period.

11583 (U//FOUO) Table 2.6-3 summarizes needs, gaps, and areas for exploration for Situational
11584 Awareness.

11585

Table 2.6-3: (U//FOUO) Summary of Technology Gaps

This table is (U//FOUO)		
Need	Gaps	Areas for Exploration
Develop and present the situation (via GUI). (The GUI supports all of the other needs listed below.)	3-D scientific data visualization tools for this application need enhancing.	Ways to effectively present to the user the security configuration and status of the enterprise.
	Interactive GUI tools and forms need developing specific to this application	The GUI needs to allow the user to respond to events. Management events would include changes in the network and new requirements; Operational events would include alerts, problems, and failures.
Application (high-level) security management and operations	Application-specific software tools need to be written for this DoD problem domain.	Managing changes in software to support changes in policy, developing CND COAs, deploying new CND services, and upgrading IAVM processes.
		Operationally performing the IAVM process, setting INFOCON levels dynamically, coordinating cyber awareness and reactions with other organizations.
Infrastructure (medium-level) security management and operations	Research products (e.g. Outpost, Network Policy Product) from Federally Funded Research and Development Centers (FFRDCs) need to be extended.	Managing security of web portals and servers, access lists in routers, database servers, modem pools, and policy settings in proxy servers.
		Operationally changing routing paths, accessibility to domain name servers, and the accessibility status of a modem port.
Security device (low-level) management and operations	Many COTS Intrusion Detection Systems (IDSs) exist. Applying them to large-scale DoD enterprise systems is a challenge.	Managing external-threat intrusion detectors, internal-threat sensors, and policy settings in firewalls.
		Operationally analyzing firewall logs, monitoring connections to the proxy server, and analyzing intrusion detection alerts.
This table is (U//FOUO)		

11586

11587 (U//FOUO) In the area of enterprise-wide mapping of services/applications, advanced
 11588 infrastructures require the mapper to manage, process, and interpret the volumes of data required
 11589 to protect an information infrastructure. This includes strategies for discovery, data storage and
 11590 retrieval, and visualization techniques to identify both network components and the defense
 11591 posture they represent.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

11592 (U//FOUO) With the current passive mapping solution implemented on a portion of the DoD
11593 enterprise to meet the above needs, further implementation of the technology across the entire
11594 enterprise would provide a comprehensive solution. However, the focus of new research and tool
11595 development for enterprise-wide network monitoring and vulnerability assessments should take
11596 into account advances in intelligent agents that can potentially solve the problems faced with
11597 large-scale network situational awareness and defense posture discovery. The following gap
11598 areas need further research:

- 11599 • (U//FOUO) Validate configuration management compliance of all network resources
- 11600 • (U//FOUO) Validate INFOCON implementation conditions by combining with
11601 visualization and risk-based predictive tools
- 11602 • (U//FOUO) Verify Ports and protocol adjudication and adherence
- 11603 • (U//FOUO) Produce SA analysis and assessment tools using agent-based approaches that
11604 will allow the combination of mapping technologies

11605 (U//FOUO) There is a basic gap in host systems and networks between what kinds of system
11606 uses are intended and what uses are actually specified or allowable based on installed
11607 applications. Application-based anomaly detection work has been effective at detecting novel
11608 threats against Internet servers. Anomaly detection approaches detect changes in the normal
11609 behavioral profile of the process and flag warnings of possibly corrupted processes. Anomaly
11610 detection systems trained to look at inside activity are now being viewed as having potential
11611 application to the insider threat technology solution. However, greater emphasis needs to be
11612 focused on detecting unknown modes of misuse, rather than just focusing so heavily on detecting
11613 known attacks. The existing statistical paradigms must be pursued and refined.

11614 (U//FOUO) Reporting extremely unusual activity is important, but it is not enough. In addition,
11615 one promising approach is to describe classes of misuse probabilistically, so that much of the
11616 generalization potential of anomaly detection is retained but with improved sensitivity and
11617 specificity. Finally, signature detection is required for attacks manifest in single events or buried
11618 in a mostly normal stream (so that signal integration will not make it stand out sufficiently). We
11619 propose an innovative approach based on hybrid systems integrating anomaly detection (model-
11620 free inference) and Bayes (probabilistic, model-based).

- 11621 • (U//FOUO) Use of zone/node sensors that operate on the concept of reporting status
11622 changes to their nearest neighbor
- 11623 • (U) Geolocation of attacks

11624 (U) A number of different automated approaches to the IP traceback problem have been
11625 suggested. However, no current method is completely effective in large-scale networks. This is
11626 known as the IP traceback technology gap.

- 11627 • (U//FOUO) Performance/situational monitoring in the Black Core
- 11628 • (U//FOUO) CND for high speed, high volume coalition services

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 11629 • (U//FOUO) CND for high speed, high volume cross domain services
- 11630 • (U) Automatically blocking DoD attacks

2.6.5.3 (U) Infrastructure

11632 (U//FOUO) The creation and enterprise-wide implementation of an Enterprise Wide Sensor Grid
 11633 (ESG) is essential to meet the needs of the UDOP. An ESG will provide collection capabilities
 11634 for correlation and analysis of CND events and activities from single or multiple sensor
 11635 categories (i.e., combine attack data with inventory, vulnerabilities, and network status data). It
 11636 provides information to the CND Analyst community that facilitates the execution of selected
 11637 COAs to mitigate and respond to attacks directed at the GIG. The ESG will collect, process, and
 11638 store sensing environment information (raw, processed, correlated, alert, etc.) and make that
 11639 information available for use to the CND UDOP.

2.6.5.4 (U) Technology Timelines

11641 (U//FOUO) Figure 2.6-4 contains preliminary technology timelines for this IA System Enabler.
 11642 These are the results of research completed to date on these technologies. These timelines are
 11643 expected to evolve as the RCD and the research of technologies related to these capabilities
 11644 continues.

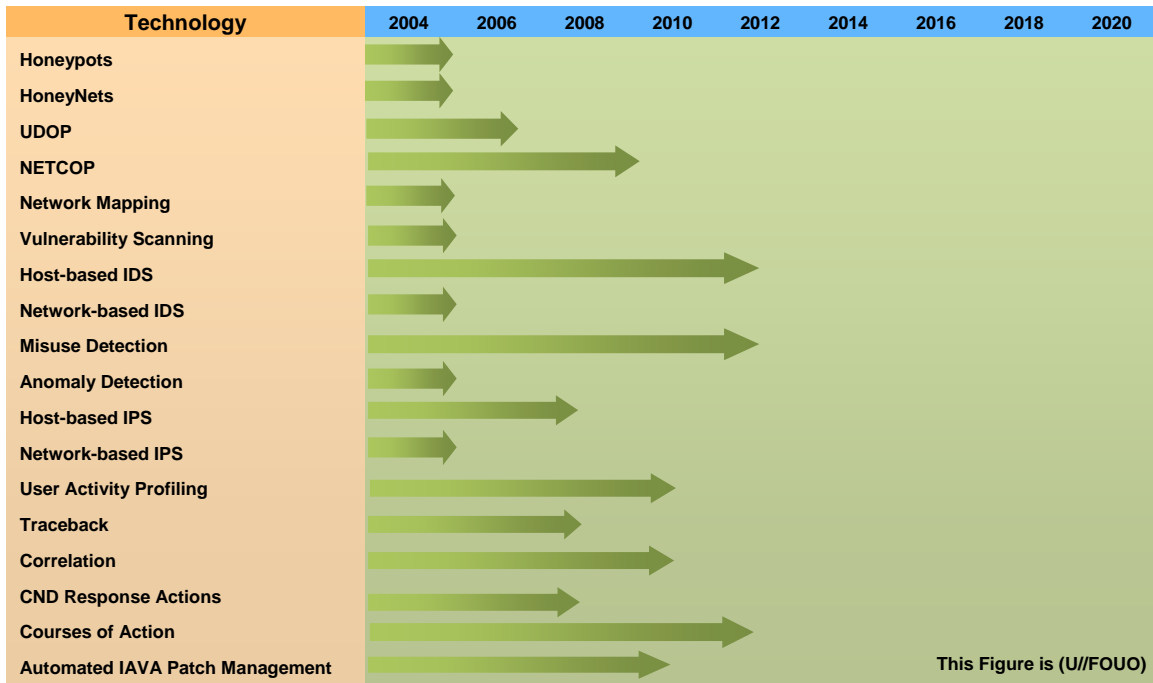


Figure 2.6-4: (U) Technology Timeline for Network Defense and Situational Awareness

11648 **2.7 (U) MANAGEMENT OF IA MECHANISMS AND ASSETS**

11649 (U//FOUO) Management of IA Mechanisms and Assets encompasses the policies, procedures,
11650 protocols, standards, and infrastructure elements required to reliably support initialization and
11651 full lifecycle management of IA mechanisms and assets. IA Mechanisms are persistent data
11652 constructs that support key IA services including identity, privilege, keys, and certificates. IA
11653 Assets are devices/software that perform an IA function. Some IA assets are:

- 11654 • (U//FOUO) Cryptographic Devices (including devices providing data in transit/data-at-
11655 rest protection and protection of management and control information)
- 11656 • (U//FOUO) Cross-Domain Solutions, Firewalls, Guards
- 11657 • (U//FOUO) Call Trace/lawful Intercept Systems
- 11658 • (U//FOUO) Intrusion Detection/Prevention Systems
- 11659 • (U//FOUO) Audit Management Systems
- 11660 • (U//FOUO) Virus Protection Software
- 11661 • (U//FOUO) Key Generation/Management Systems
- 11662 • (U//FOUO) Policy Enforcement Points (including devices that control access to
11663 information).

11664 **2.7.1 (U) GIG Benefits due to Management of IA Mechanisms and Assets**

11665 (U//FOUO) The Information Assurance constructs used to support Management of IA
11666 Mechanisms and Assets provide the following benefits to the GIG:

- 11667 • (U//FOUO) Secure management of persistent IA constructs (e.g., identity, privilege,
11668 policy, key/certificate)
- 11669 • (U//FOUO) Secure management of devices/software that performs an IA function
- 11670 • (U//FOUO) Prevention of establishment of false identities, rogue Communities of Interest
11671 (COI)s, etc.
- 11672 • (U//FOUO) Elimination of manual keying, configuration, and inventorying of IA assets
- 11673 • (U//FOUO) Support for compromise recovery of IA mechanisms and assets
- 11674 • (U//FOUO) Standardized protocols and common data packaging formats to address the
11675 complications of managing numerous IA-enabled enterprise entities.

11676 **2.7.2 (U) Management of IA Mechanisms and Assets: Description**

11677 (U//FOUO) Management of IA Mechanisms and Assets focuses on providing management and
11678 control of security data, processes, and resources. The security of management and control data,
11679 process and resources is the focus of the Assured Resource Allocation enabler.

11680 (U//FOUO) The Security Management infrastructure is comprised of components, services, and
11681 products provided by external systems and within the system. Examples of products provided by
11682 a Security Management Infrastructure (SMI) include:

- 11683 • (U) Unique identities for all GIG entities and COIs
- 11684 • (U) Symmetric keys
- 11685 • (U) Public keys
- 11686 • (U) X.509 certificates
- 11687 • (U) New or updated software-based cryptographic algorithms, operating systems,
11688 application software updates and patches
- 11689 • (U) Virus update files.

11690 Examples of services that must be provided by the GIG SMI include:

- 11691 • (U) [Identity Management](#)
- 11692 • (U) [Privilege Management](#)
- 11693 • (U) [Key Management](#)
- 11694 • (U) [Certificate Management](#)
- 11695 • (U) [Configuration Management of IA Devices and Software](#)
- 11696 • (U) [Inventory Management](#) of IA Devices
- 11697 • (U) [Compromise Management of IA Devices](#)
- 11698 • (U) [Audit Management](#).

11699 **2.7.2.1 (U) Identity Management**

11700 (U//FOUO) Identity management is the capability to unambiguously associate unique assured
11701 digital identities with individuals (a.k.a., human), named groups (e.g. Organizational Domains,
11702 Operational Domains, COIs), devices, and services. Assured identities are made available to
11703 processes and functions that create, modify, or enforce policy and privileges and, therefore, must
11704 be guaranteed to represent the real GIG entity. Due to the criticality of the assured digital
11705 identity, the infrastructure that provides identity management must ensure the confidentiality,
11706 integrity, and availability of the identity registration processes, equipment, configurations,
11707 registries, and databases that it uses to operate.

11708 (U//FOUO) The scope of identity management includes the entire lifecycle of an identity from
11709 creation, maintenance of information associated with an identity, revocation, and retiring of the
11710 identity. For named groups, identity management also includes updating the mapping of
11711 individual identities to the group. Identities must be persistent in the GIG; they cannot expire, be
11712 overwritten, or reset by events in the GIG. In fact, the identity registered for an individual is
11713 unique and remains constant despite changes of that individual's name or other attributes.

11714 **2.7.2.1.1 (U) Identity Creation**

11715 (U//FOUO) The process of creating an assured digital identity is called registration. Human
11716 registration includes the process of performing identity proofing, establishing a unique ID and
11717 initial user profile, and creating an authentication token. The authentication token may be a
11718 personal token or device management key that will later be used to authenticate that identity. At
11719 a minimum, the digital identity consists of an identifier (e.g., serial number or user name) and an
11720 associated set of attributes (for a human user, attributes may include password, PIN,
11721 public/private key pair, fingerprint, and retinal scan.) that can be used to authenticate the identity
11722 when access is requested. Assured identities must be nonforgeable to prevent masquerades.

11723 (U//FOUO) Registration of individuals establishes and maintains a user profile that refers
11724 unambiguously to an identified entity. The identification information verified (e.g., passport,
11725 birth certificate) or collected (e.g., biometrics) during the identity proofing is maintained as
11726 identity data in the user profile. The identity proofing method used to register the individual is
11727 also maintained in the user profile and used as a factor in an access control decision. Identity
11728 proofing mechanisms for individuals could range from no proof of ID presented during
11729 registration to presenting multiple picture IDs in person. Identity proofing for devices and
11730 services will require different standards and processes than those for users.

11731 (U//FOUO) In addition to GIG users, all managed GIG devices and services will have an assured
11732 identity. Currently devices have a serial number or a Media Access Control (MAC) address
11733 associated with them, based on their Network Interface Card (NIC). This will evolve to a
11734 nonforgeable identity in the future so that individual devices can be identified with their
11735 configurations, software, hardware, and firmware. Unique identities for managed devices will
11736 also enable the management infrastructure to more accurately keep track of GIG resources and
11737 more effectively manage devices.

11738 (U//FOUO) Identity proofing of devices and processes will differ from that for individuals. For
11739 example, proofing of a device may require examination by a competent authority to determine
11740 whether it is a National Security Agency (NSA)-certified Type-1 device, a FIPS-level 1 device,
11741 or an uncertified device. A check of the device serial number, manufacturer's equipment number,
11742 etc., before putting the device into the GIG may also be appropriate. The result would be
11743 included in the registration profile of the device. In addition, the registration process may have to
11744 verify the pedigree of the device or service to avoid connecting potentially compromised devices
11745 or services to the GIG.

11746 (U//FOUO) Registration requires a heterogeneous system based on open standards for identity
11747 management that focus on non-proprietary mechanisms and procedures. Methods will be
11748 required for real-time enrollment and authorization of entities in the GIG as well as archiving,
11749 binding, and auditing their identities and credentials.

11750 **2.7.2.1.2 (U//FOUO) Identity Maintenance**

11751 Information associated with an identity must be maintained as events occur that change the
11752 attributes of the entity the identity represents. For example, an individual may change their name.
11753 The user profile for the individual must then be updated to reflect the new name for the
11754 individual. Other events that may require user profiles to be updated include:

- 11755 • (U//FOUO) A new authentication token is received by the individual
- 11756 • (U//FOUO) An authentication token is compromised
- 11757 • (U//FOUO) An individual is added to or removed from a named group.

11758 **2.7.2.1.3 (U//FOUO) Retiring of the identity**

11759 (U//FOUO) Identities could become obsolete for a variety of reason including:

- 11760 • (U//FOUO) An individual no longer will be operating on the GIG
- 11761 • (U//FOUO) A named group is no longer needed
- 11762 • (U//FOUO) A device is destroyed.

11763 (U//FOUO) Under any of these conditions the identity would be retired, but not deleted.
11764 Identities would be archived to allow the continued analysis of historical transactions involving
11765 that identity. As a result, the Identity Management Infrastructure must be able to archive and
11766 restore identities.

11767 **2.7.2.2 (U) Privilege Management**

11768 (U//FOUO) The GIG model is based upon massively distributed resources and services that are
 11769 to be dynamically and selectively drawn from (e.g., information Pull) and utilized by a large and
 11770 diverse user population. In addition, this same user population will be given the capability to
 11771 influence and modify (e.g., information Post or Push) the GIG-resident databases. Due to these
 11772 inherent capabilities of the GIG, a globally robust and secure way is required to manage the
 11773 privileges assigned to a GIG entity. The synchronization of privileges across the GIG is essential
 11774 to support collaborative sessions that do not overstep policy-mandated sharing boundaries. The
 11775 potentially vast GIG user base combined with the tremendous range of sensitivity/classification
 11776 of future GIG-resident resources makes the privilege management function of utmost
 11777 importance.

11778 (U//FOUO) The GIG's Privilege Management Infrastructure (PMI) needs to be an evolution of
 11779 and improvement upon traditional techniques. In general, utilization of some computer-based
 11780 resources or applications has always required both the authentication (verification of identity)
 11781 and authorization (verification of privilege) of a potential user. Traditionally, authorization
 11782 employed an Access Control List (ACL) that was held internal to and controlled by the
 11783 application itself. The most recent concepts for privilege management enable the
 11784 authorization/privilege verification process to be drawn outside of individual applications. This
 11785 paradigm is essential for the robust and efficient operation of the future GIG.

11786 (U//FOUO) Privilege management in the GIG must be scaleable. Privileges will be needed in a
 11787 timely fashion and consistent with their valid and authorized requirements. Potential conflicts
 11788 and inconsistencies between the various sources of authority will require the development of
 11789 GIG-wide arbitration entities so as to arrive at universally acceptable privilege attributes before
 11790 multiple users or entities enter into any joint missions. It is anticipated that many groups (e.g.,
 11791 COIs) will manage their own privileges.

11792 (U//FOUO) In all cases, the base mechanism for communicating privileges needs to be
 11793 consistent. However, the set of privileges granted will vary from entity to entity. As a result, the
 11794 assured identities of an entity will be associated (cryptographically bound) with one or more sets
 11795 of privileges, likely a separate set for each role and COI to which the entity belongs or supports.
 11796 The group of bound privileges to an assured identity would be part of a User Profile.

11797 (U//FOUO) Privilege management must support the following operational concepts and
 11798 environmental conditions:

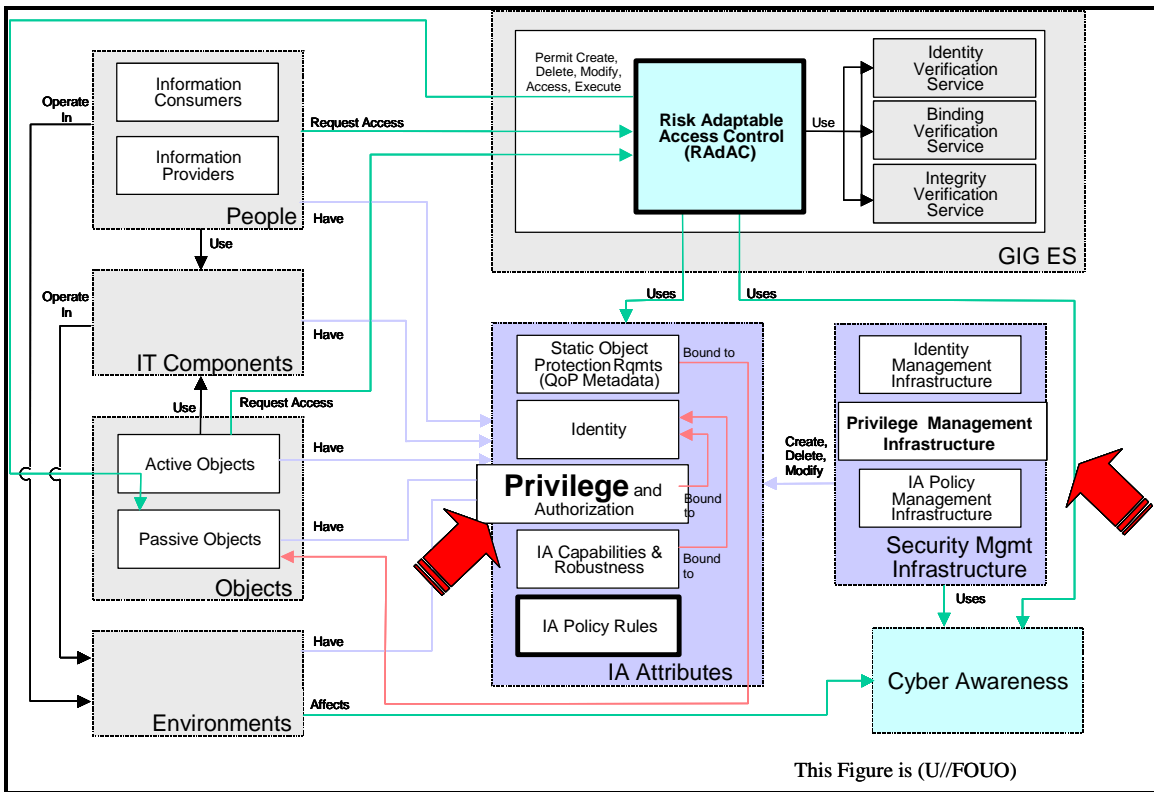
- 11799 • (U//FOUO) RAdAC Model
- 11800 • (U//FOUO) Multiple Security Domains
- 11801 • (U//FOUO) Temporary Mission Needs
- 11802 • (U//FOUO) Dynamic COIs
- 11803 • (U//FOUO) Operation within GIG Network of Networks Context
- 11804 • (U//FOUO) Trusted Transport/Distribution/Synchronization

- (U//FOUO) Role-based Privileges.

(U) The roles of privilege management in supporting each of these are described in the following sections.

2.7.2.2.1 (U//FOUO) Privilege Management Role in RAdAC Assured Sharing Model

(U//FOUO) One of the core concepts of the GIG, essential to enabling on-the-fly and situational-agile access-privilege control, is the RAdAC model, shown in Figure 2.7-1.



This Figure is (U//FOUO)

11811

Figure 2.7-1: (U//FOUO) Assured Sharing Context Diagram Emphasizing Privileges

11812

(U//FOUO) As shown, the Privilege IA attribute and the PMI that manages it are key elements in the notional flow of the RAdAC process. Moreover, not only do users have privilege authorization; so do the active objects they access (e.g., applications and services) and the IT components that they use (e.g., routers, servers, clients). As discussed in Section 2.2, Policy-Based Access Control, the privileges of all entities involved in a transaction are evaluated before granting access. For example, the user may have the right privileges to access information, but the client through which the user is accessing the GIG may be in a less secure environment or may not have the required set of IA capabilities or security robustness to permit access. In this case, access would be denied.

(U//FOUO) The privileges that any future GIG PMI must manage will include privileges to not only gain knowledge of distributed GIG resources, but also to act upon those resources, e.g., read, write, modify, delete, and share various information entities, be they data, software, or policy. Thus the PMI needs to be multidimensional in this sense.

11822
11823
11824
11825

11826 2.7.2.2.2 (U//FOUO) Accommodations of Multiple Classification Levels

11827 (U//FOUO) One of the basic features that will drive the function of a GIG PMI will be the need
11828 to accommodate multiple levels of classification. This applies both to the situation in which a
11829 single user is operating within the context of a single session on the GIG (in which case that lone
11830 user's clearance level shall dictate the classification level up to which the user may gain access)
11831 and also to the more likely scenario in which multiple users of potentially different clearance
11832 levels must collaborate in order to accomplish a joint mission. Collaboration requires joint
11833 situational awareness based on the lowest common denominator of clearance-based privileges so
11834 as to not violate or overstep any classification-limited sharing boundaries.

11835 (U//FOUO) An example of how this might work is a Multi-Level Security (MLS) system with
11836 the following type of Mandatory Access Control scheme. Each piece of information is given a
11837 security label (as metadata), which includes classification level (e.g., unmarked, unclassified,
11838 FOUO, NATO-restricted, confidential, secret, top secret, compartmented), and each subject user
11839 has a clearance which specifies the classification level the user is permitted to access.

11840 (U//FOUO) A potential security policy (i.e., privilege) designed to stop information leakage
11841 while maximizing sharing would permit formation of collaborative sessions among a group of
11842 users at a level equal to or lower than the lowest common set of privileges. Users with MLS
11843 devices could form multiple concurrent sessions at different levels, and they could shift between
11844 levels based on the current access policy (e.g., read down/write up). Users with single-level
11845 devices would have to either end one session to access information at a different level (as
11846 determined by RAdAC), transfer the information through a cross-domain solution (assuming the
11847 information was at an appropriate level as determined by RAdAC), or request information only
11848 at or below their current level. This would allow users to read targets with lower classifications
11849 than their own clearance and to write to targets with higher classifications. Thus, effective
11850 collaboration within a coalition of users with varying clearance levels is accommodated.

11851 2.7.2.2.3 (U//FOUO) Adaptation to Temporary Mission Needs

11852 (U//FOUO) Exception handling to support temporary mission needs would be supported by a
11853 policy that designates when exceptions are allowed (given human intervention) for access to GIG
11854 resources not normally available based upon an entity's current privileges. In this case, it may be
11855 necessary for the GIG privilege management infrastructure to enable temporary (time-limited)
11856 alterations to individual privileges to support the special mission. In this example, an entity
11857 would be temporarily provided the privilege to assert precedence or priority for access to certain
11858 GIG resources during a specific mission. This will require that the PMI provide for globally-
11859 available notification of this increase in privilege and that it be automatically validated system-
11860 wide.

11861 2.7.2.2.4 (U//FOUO) Support of Dynamic COIs

11862 (U//FOUO) Future COIs that operate within the context of the GIG are likely to be not only
11863 diverse but dynamic from day to day as single coalition partners arrive and depart from
11864 participation in collaborative sessions. This may require an adaptive and agile scheme to assign
11865 and modify individual and coalition-wide privileges to meet needs.

11866 **2.7.2.2.5 (U//FOUO) Operation within the GIG Network of Networks Context**

11867 (U//FOUO) The GIG will evolve as a collection of networks that are tied together, each with its
11868 own Network Operations Center (NOC). These networks include the Transformational Satellite
11869 (TSAT) network, the Global Information Grid – Bandwidth Expansion (GIG-BE) network, the
11870 mobile/wireless JTRS Joint Tactical Radio System (JTRS), and Net Centric Enterprise Services
11871 (NCES). These first three are the fundamental transport networks over which GIG services (such
11872 as NCES) will be accessed.

11873 (U//FOUO) Each of the transport networks and enterprise services will have its own defined
11874 populations of users and operational entities, all of whom will require managed sets of privilege
11875 attributes. Privilege management can be thought of as occurring at three basic levels (from
11876 lowest to highest)—local administration, Service/Transport network operations and
11877 administration (as described above), and GIG-wide operations and administration.

11878 (U//FOUO) Control of the various networks will be done through action of each relevant NOC,
11879 with an envisioned GIG-wide NOC eventually coming into being (though not entirely
11880 supplanting the intermediate NOC roles). Division of network control among these requires a
11881 commensurate PMI functionality across these networks. This mandates the tying together and
11882 cross-awareness of the various PMI level actions so that privileges are jointly adjudicated.

11883 **2.7.2.2.6 (U//FOUO) Trusted Transport/Distribution/Synchronization**

11884 (U//FOUO) In support of essentially static COIs, the GIG PMI will need to have the ability to
11885 securely transport (with integrity and confidentiality) and distribute privileges to all necessary
11886 parties before collaborative sessions can start. If a coalition membership becomes dynamic with
11887 resultant modification of joint privileges, then there will be a need for timely and synchronous
11888 distribution across the GIG of sharing privilege modifications.

11889 **2.7.2.2.7 (U//FOUO) Support of Role Based Privileges**

11890 (U//FOUO) In addition to individual-based privilege management, there will likely be the need
11891 for role-based privileges in the GIG. A role is defined by a specific set of tasks that require a set
11892 of privileges in order to be performed. Typical roles in the GIG would be IA security manager
11893 with policy-setting privileges, network administrator with NOC control privileges, and mission-
11894 specific roles.

11895 2.7.2.3 (U) Key Management

11896 (U//FOUO) Cryptography is one of the fundamental IA mechanisms used to protect the GIG, and
11897 cryptography cannot be implemented correctly without key management. Key management is
11898 one of the fundamental aspects of Information Assurance. The full lifecycle of key management
11899 includes:

- 11900 • (U) Key Management Practice Statement
- 11901 • (U) Key Ordering
- 11902 • (U) Key Generation and Labeling
- 11903 • (U) Key Packaging and Distribution
- 11904 • (U) Storage, Backup and Recovery
- 11905 • (U) Revocation and Destruction.

11906 2.7.2.3.1 (U) Key Management Practice Statement

11907 (U//FOUO) A Key Practice Statement is a document that describes the process of handling and
11908 controlling cryptographic keys and related material (such as initialization values) according to
11909 key policy. It details key management functions and parameters available to authorized users.

11910 (U//FOUO) Key Management Plans are written for systems that use keys. Such plans need to be
11911 compatible with the Key Practice Statement. However since the GIG is not being built or
11912 operated as a single, consolidated system, it is not reasonable to expect that there will be a single
11913 GIG Key Management Plan. Rather, each constituent component of the GIG (e.g., GIG-BE,
11914 TSAT, JTRS, and end user systems connecting to the GIG) must have a key management plan.
11915 Component key management plans will adhere to established key management standards and
11916 approved architecture. Appropriate authorities for completeness and consistency with other
11917 component key management plans must review these plans, and any discrepancies must be
11918 resolved prior to operation. For example, if the GIG-BE key management plan makes
11919 assumptions about the duty and ability of End Cryptographic Units (ECUs) to protect keys, then
11920 no ECU should be connected to the GIG-BE unless its key management plan clearly states how it
11921 protects those keys sufficiently to meet GIG-BE assumptions.

11922 2.7.2.3.2 (U) Key Ordering, Generation, Labeling, Packing, and Distribution

11923 (U//FOUO) The first phase of a key's life supports the request and delivery of key material to the
11924 intended recipient. This begins by ordering of the key material by a user who is authorized to
11925 request keys. Once an order is verified to come from a valid requestor, an authorized key source
11926 can generate the key material, label the key and its attributes, package the key in a manner
11927 compatible with delivery protocol, and distribute the key to the specified recipient.

11928 (U//FOUO) Distribution may be either physically or electronically. Electronic delivery includes
11929 the use of NSA-approved benign techniques for encrypted, over-the-network (OTNK) key
11930 distribution by a direct network connection between the keying source and the intended receiving
11931 device.

11932 (U//FOUO) Keys and algorithms used by GIG components must be only those approved by
11933 authorized key sources (e.g., NSA). Keys can be either locally-generated or provided by a central
11934 authority. Keys provided by a central authority must be validated before being used. Locally-
11935 generated keys must be generated only through approved processes and equipment and must be
11936 used only within defined constraints.

11937 (U//FOUO) Any cryptographic algorithms used in the GIG must be approved by authorized
11938 sources. No ECU shall use an algorithm unless it can be validated as approved by an authorized
11939 source and not be modified in an unapproved way.

11940 **2.7.2.3.3 (U) Storage, Backup and Recovery**

11941 (U//FOUO) Key storage is performed at the authorized key source and at the receiving device.
11942 Keys must be stored securely on an ECU. Even if stored in software rather than on a dedicated
11943 hardware device, the key must be stored so that it can neither be extracted easily by an attacker
11944 (including attackers' software agents), nor modified without detection in an unauthorized way.

11945 (U//FOUO) At the trusted key source, the key must be backed up to support the following:

- 11946 • (U) Decryption of stored enciphered information
- 11947 • (U//FOUO) Continuity of operation when the key is not readily available due to
11948 conditions such as crypto period expiration, key corruption, or permanent departure of the
11949 key owner
- 11950 • (U//FOUO) Key recovery.

11951 (U//FOUO) The key management infrastructure must be able to identify all ECUs impacted by a
11952 key compromise and ensure the rapid recovery of operations by supporting key compromise
11953 recovery mechanisms with the affected ECUs.

11954 **2.7.2.3.4 (U) Revocation and Destruction**

11955 (U//FOUO) At times it is necessary to revoke a key before its expiration. This may occur
11956 because its use is no longer needed, or the key may have been compromised. Revocation of a key
11957 that has not been compromised does not require its destruction, but the key management
11958 infrastructure must support a mechanism for notifying GIG entities that the key can no longer to
11959 be used.

11960 (U//FOUO) All GIG components must have a way of destroying keys when circumstances
11961 require it. When a key is destroyed, it must not be possible for an adversary with physical
11962 possession of the hardware on which the key resided to recover any parts of the key. Key
11963 destruction mechanisms must be designed in such a way as to minimize the chance of unintended
11964 or accidental destruction.

11965 **2.7.2.4 (U) Certificate Management**

11966 The following main phases define the certificate life cycle management process;

- 11967 • (U) Adherence to CPS (Certificate Practice Statement)
- 11968 • (U) Registration/Enrollment
- 11969 • (U) Certificate Creation
- 11970 • (U) Certificate Distribution
- 11971 • (U) Certificate Retrieval
- 11972 • (U) Certificate Expiration
- 11973 • (U) Certificate Revocation.

11974 **2.7.2.4.1 (U) Adherence to CPS**

11975 (U) The Certificate Practice Statement lists the services supported and practices used throughout
11976 the Certificate Life Cycle. These services include registration, creation, distribution, storage,
11977 retrieval, revocation, and other supporting sub-services. This process is used to govern the
11978 operating principles at the various levels – which include individual components, enclaves,
11979 enterprise or the entire infrastructure (e.g., Public Key Infrastructure [PKI]). The adherence to
11980 the CPS should be auditable, and the appropriate measures should be place to account for
11981 activities related to Certificate Management phases.

11982 **2.7.2.4.2 (U) Registration**

11983 (U) Registration process starts when an end-entity requests a Registration Authority (RA) to
11984 issue a certificate. Depending on the Certificate Practice Statement, Certificate Policy, and
11985 privileges associated with the requested certificate, the identity verification may require a
11986 physical appearance or submission of appropriate authorization documentation. The same is true
11987 for registering devices, except that devices do not make appearances, but rather have a
11988 representative to act on their behalf.

11989 (U) RAs are a critical element within the infrastructure. The assurance level attained within the
11990 infrastructure is dependent on the accuracy of their actions and their adherence to established
11991 policies. The higher the level of assurance required within the infrastructure, the more stringent
11992 the identification process. The RA provides the new User's information to the Certificate
11993 Authority (CA) which then creates a key pair and a Certificate.

11994 (U) Clearly, the Registration Authority plays a very critical role in the overall security and
11995 integrity of the infrastructure. If RAs do not adhere to established procedures and properly verify
11996 identify or accurately enter other personal information, they put the entire infrastructure at risk.

11997 2.7.2.4.3 (U) Certificate Creation

11998 (U) The CA has responsibility for certificate creation, regardless of where the key is generated.
11999 A certificate binds an entity's unique distinguished name (DN) and other additional attributes
12000 that identifies an entity with a public key associated with its corresponding private key. The
12001 entity DN can be an individual, an organization or organizational unit, or a resource (web-
12002 server/site). Appropriate certificate policies govern creation and issuance of certificates. The
12003 public key needs to be transmitted securely to the CA in case it was generated elsewhere by a
12004 party other than the CA. Certificates can be used to verify a digital signature or for encryption
12005 purposes.

12006 (U) There are several groups working on the standards for a specific application area, and hence
12007 there exist a number of certificate profiles or formats for different requirements. SPKI, PGP, and
12008 SET formats are popular versions. Most of them derive from the X.509 Version 3.0 specification.
12009 A typical X.509 Certificate contains several standard fields and additional policy-related
12010 extension fields.

12011 (U) Though certificates enable the PKI, there are several privacy issues surrounding an
12012 individual's certificate usage [2]. Requests and subsequent distribution of keys and certificates
12013 require secure transmission modes. The IETF PKIX working group has defined management and
12014 request message format protocols (CMP/CRMF) specifically for this purpose. Alternatives such
12015 as Public Key Cryptography Standards (PKCS) also exist.

12016 2.7.2.4.4 (U) Certificate Distribution

12017 (U) Certificate Distribution involves securely and easily making the certificate information
12018 available to a requestor. This can be done through several techniques, including out-of-band and
12019 in-band distribution, publication, centralized repositories with controlled access, etc. Each has its
12020 own benefits and drawbacks.

12021 (U) Depending on the client-side software, certificate usage, privacy and operational
12022 considerations, the information requirements and distribution methods vary. Several protocols
12023 are available that facilitate secure distribution of certificates and revocation information. For
12024 example, enterprise domains widely use LDAP repositories with appropriate security controls
12025 along with in-band distribution through S/MIME based e-mail. This hybrid approach maximizes
12026 the benefits. Even within the repository model several configurations like direct-access, inter-
12027 domain replication, guard mechanism, border, and shared repositories are possible and often
12028 used.

12029 2.7.2.4.5 (U) Certificate Retrieval

12030 (U) Certificate Retrieval involves access to certificates for general signature verification and for
12031 encryption purposes. Retrieval is necessary as part of the normal encryption process for key
12032 management between the sender and the receiver. It is also necessary for verification, as a
12033 reference where the certificate containing the public key of a signed private key is retrieved and
12034 sent along with the signature or is made available on demand.

12035 (U) It is imperative to have an easy and simple mechanism to retrieve certificates. Otherwise the
12036 whole infrastructure will introduce unacceptable inefficiency. Validation is performed to ensure
12037 a certificate has been issued by a trusted CA in accordance with appropriate policy restrictions
12038 and to verify its integrity and validity (whether expired/revoked) before its actual use. In most
12039 cases all this is achieved transparently by the client-software before cryptographic operations
12040 using the certificate are carried out.

12041 **2.7.2.4.6 (U) Certificate Expiration**

12042 (U) Certificate Expiration occurs when the validity period of a certificate expires. Every
12043 certificate has a fixed lifetime and expiration is a normal occurrence. A certificate can be
12044 renewed provided the keys are still valid and remain uncompromised. When renewed, a new
12045 certificate is generated with a new validity period. In this case, the same public key is placed into
12046 the new certificate. Alternatively, a certificate update can also be done to create essentially a new
12047 certificate, with a new key pair and new validity period. Certificate update, like key update must
12048 take place before the certificate expires. In this case, the policy restrictions may remain the same
12049 as that of the expiring certificate.

12050 **2.7.2.4.7 (U) Certificate Revocation**

12051 (U) Certificate Revocation is the cancellation of a certificate before its natural expiration. Several
12052 situations warrant revocation. For instance, it could be due to privilege changes for the certificate
12053 owner, key loss due to hardware failure, private key compromise, etc. Cancellation per se is an
12054 easier process when compared to properly notifying and maintaining the revocation information.
12055 The delay associated with the revocation requirement and subsequent notification is called
12056 revocation delay. This is clearly defined in the Certificate Policy, because it determines how
12057 frequently or quickly the information is broadcast and used for verification.

12058 (U) When there is a subscriber compromise, all subscribers within the entire infrastructure can be
12059 exploited until the compromise is detected. Therefore, compromises of individual subscribers
12060 must be dealt with quickly and efficiently, with new keys generated as appropriate. Concurrently,
12061 the Compromised Key List (CKL) would need to be updated. Should the CA itself be
12062 compromised, all CA subscribers would need to be rekeyed and new Certificates created.

12063 **2.7.2.5 (U) Configuration Management of IA Devices and Software**

12064 (U//FOUO) Configuration Management (CM) of IA devices and software provides the ability to
 12065 manage and control the IA equipment and software components that provide the framework for
 12066 the IA infrastructure or provides IA services within the GIG. Examples of these components
 12067 include ECUs, trusted platforms, trusted software, and software elements that provide or support
 12068 IA functionality (e.g., anti-virus updates). An ECU is a device, normally a component of a larger
 12069 system, which contains cryptographic functionality, provides security services to the larger
 12070 system, and from the viewpoint of a supporting management infrastructure, is the identifiable
 12071 component with which a desired management transaction can be conducted. Management
 12072 transactions can also be conducted with IA software elements, which include either embedded or
 12073 stand-alone software functionality that supports GIG IA services.

12074 (U//FOUO) Configuration Management activities involve the distribution, handling, and storage
 12075 of software, data packages, and policy used by the IA devices or software to control dynamic
 12076 mission parameters needed to establish their various operational configurations.

12077 (U//FOUO) The types of configuration changes considered to be part of IA CM, as compared to
 12078 the CM performed as part of traditional network management, include:

- 12079 • (U//FOUO) Cryptographic algorithm updates
- 12080 • (U//FOUO) IA device feature updates
- 12081 • (U//FOUO) Virus (malware) detection/prevention updates.

12082 (U//FOUO) Cryptographic algorithm updates are needed to support the GIG 2020 Vision in
 12083 which ECUs must be able to change algorithms to meet new interoperability or mission
 12084 requirements. This change—adding support for new algorithms; ceasing support for outdated
 12085 algorithms; switching algorithm modes—must happen only under authorized conditions. That is,
 12086 the units must have a way to recognize that an authorized entity is telling it to change algorithm
 12087 support, and the unit must then be capable of acting on that request. Unauthorized attempts to
 12088 change algorithm support must be rejected.

12089 (U//FOUO) Coalition interoperability is one example in which the ability to upload different
 12090 cryptographic algorithms is beneficial. Currently, coalition interoperability is generally
 12091 accomplished by providing U.S. systems to partners. However, this has some negative side
 12092 effects; notably, the coalition partner has direct access to U.S. hardware and software. It also
 12093 requires the logistics step of physically transporting that hardware to the coalition partner's
 12094 location and training coalition partners on equipment operation. In the 2020 system, the GIG
 12095 must be capable of interoperating with coalition partners' existing systems. By uploading
 12096 algorithms in the U.S. equipment that are compatible with the coalition partners' equipment,
 12097 there would be no need to share U.S. equipment, because our equipment would interoperate with
 12098 the coalition equipment. The GIG must interoperate with coalition partners, while
 12099 simultaneously providing a high assurance U.S.-only capability. The ability to communicate on
 12100 one channel of the equipment using the coalition partner's algorithms and on another channel
 12101 with U.S. algorithms satisfies warfighter needs.

12102 (U//FOUO) U.S. Policy sometimes requires a reduced set of features in IA enabled devices used
12103 overseas. The CM characteristic that supports device feature updates enables the capabilities of
12104 the device to be tailored to the feature set appropriate for the operating environment.

12105 (U//FOUO) Today, the control and management of virus (malware) detection/prevention
12106 capability is currently performed locally at a virus detection server. These server activities
12107 include application update and configuration per policy, virus signature pull operations from the
12108 external source to the parent server, and configuring the update (push) and scan policy for clients
12109 connected to this parent server. The parent virus detection server can also gather statistics and
12110 scan results based on CND policy settings.

12111 (U//FOUO) In the future, as the GIG migrates from edge-to-edge encrypted network to a
12112 converged Black Core (end-to-end) network, it will become more critical that trusted, and up to
12113 date virus detection applications be resident on GIG clients. This client application-based
12114 malware code defense will form a last critical barrier in this type of encrypted core architecture
12115 where IPv6 tunneled packets are not decrypted and checked at the traditional DMZ firewall
12116 network boundary. This type implementation will make scalability, distribution of updates, and
12117 synchronization important between the parent virus detection server and the large number of
12118 GIG client that could be affected by this type of malware attack.

12119 (U//FOUO) CM operations are accomplished by information exchange between GIG
12120 management systems (local or remote) and target devices and software components. The
12121 following paragraphs highlight a number of the critical aspects associated with security
12122 management of the GIG's IA devices and software.

12123 (U//FOUO) The management infrastructure is responsible for the packaging, delivery, and
12124 control of software/firmware packages/dynamic policy parameters. A software/firmware/anti-
12125 virus update package must have been developed, tested, and evaluated and validated before
12126 distribution. Distribution of validated packages could be operator initiated or automated as a
12127 result of configuration changes determined by CND operations.

12128 (U//FOUO) The CM infrastructure will verify the signature and will assume authority for the
12129 management and distribution of the package or policy. It will be responsible for commanding
12130 and performing any required preprocessing (e.g., common data formatting). As part of
12131 distribution to the target IA devices/software, the management infrastructure signs and encrypts,
12132 as required, the configuration information.

12133 (U//FOUO) Once the targeted IA devices/software receives a configuration package from the
12134 management system, it must validate the source of the package and verify the package's data
12135 integrity. This implies that the proper trust anchors have been installed. (Trust anchors and
12136 management authority are established as part of the initialization process.) Handling and storage
12137 of configuration information at a device also requires an ability to read and act upon version
12138 information contained in the package. Finally, the element must also provide feedback status
12139 information to its directing management system.

12140 (U//FOUO) The CM infrastructure must monitor and maintain compliance of the IA
 12141 devices/software configurations with the current security and configuration policies. If
 12142 discrepancies are found, distribution of the current configuration packages would be initiated.
 12143 The target IA devices/software provides version/status information in response to query traffic
 12144 from its validated and authorized management system. In summary, in order to enable the GIG
 12145 IA Vision, existing configuration management functionality must be enhanced in the areas of
 12146 source authentication support, transfer confidentiality/integrity, and version management.

12147 **2.7.2.6 (U) Inventory Management**

12148 (U//FOUO) Inventory Management provides the ability to exchange machine identification,
 12149 status, version, and network topology information between the target IA devices and the
 12150 management infrastructure. During the manufacturing or initialization process, GIG devices will
 12151 be given a Unique Identifier that conforms to an Identity Management Standard. When queried
 12152 by an authorized management system, the device will output its identification information and
 12153 configuration information. The device configuration information being queried often is sensitive.
 12154 Therefore, confidentiality of the inventory information must be maintained in this process. In
 12155 addition, queries and requests will need to be authenticated by the device before processing.

12156 (U//FOUO) The Inventory Management infrastructure uses the status information to support
 12157 higher level accounting, tracking and network location system as well as providing information
 12158 and data support to network visualization tools, cyber situational awareness, and Computer
 12159 Network Defense (CND) systems. Use of this information is described in Section 2.6.

12160 **2.7.2.7 (U) Compromise Management of IA Devices**

12161 (U//FOUO) The GIG infrastructure must support Compromise Management of IA Enabled
 12162 Equipment. IA Enabled Equipment is considered compromised when its integrity or
 12163 confidentiality is no longer assured. This might occur through such mechanisms as exploitation
 12164 of a vulnerability by a worm, or through physical loss of equipment possession. Compromise
 12165 Management includes:

- 12166 • (U//FOUO) Detection – The determination, through audit or network sensors, that a
 12167 compromise may have occurred. This is described in Section 2.6.
- 12168 • (U//FOUO) Investigation – Confirmation of the status of IA Enabled Equipment. This
 12169 involves communicating with the equipment and confirming its configuration and state.
- 12170 • (U//FOUO) Isolation – The active steps taken to ensure that the compromised equipment
 12171 is isolated from the rest of the GIG that compromised keys are invalidated, and that the
 12172 equipment is cleared of all sensitive information and rendered benign.
- 12173 • (U//FOUO) Restoration - The initialization, reconfiguration, and reconnection to the GIG
 12174 of equipment which is suspect, due either to loss of control or known compromise.

12175 **2.7.2.8 (U) Audit Management**

12176 (U//FOUO) The GIG infrastructure must also support audit management. Audit management
12177 processes include:

- 12178 • (U//FOUO) Ability to configure IA audit data gathering per policy
- 12179 • (U//FOUO) Local collection of auditable events that identify the source of the audit data
- 12180 • (U//FOUO) Secure storage and transfer of the logged information from the device to the
12181 management infrastructure
- 12182 • (U//FOUO) Ability to analyze the audit data to identify significant events.

12183 (U//FOUO) All these process must be supported with integrity services to assure accuracy of the
12184 audit data. Audit data provides a means to detect any events that resulted in a security breach of
12185 the GIG system. Once audit data is gathered from IA components and correlated, the audit
12186 management infrastructure provides Security Operations/Administrations personnel the
12187 following:

- 12188 • (U//FOUO) A means of independent review and examination of records to determine the
12189 adequacy of system controls to ensure compliance with established policies and
12190 operational procedures
- 12191 • (U//FOUO) Information needed to alter the use of resources to improve system
12192 performance
- 12193 • (U//FOUO) A source of data that can be used to identify an individual, process, or event
12194 associated with any security-violating event.

12195 (U//FOUO) In summary, to enable the GIG IA Vision, existing audit management functionality
12196 must be enhanced by incorporating unique identifiers for authenticated individuals/devices into
12197 audit event records, resource utilization recording, trusted time tagging, secure storage, transfer
12198 integrity, and risk-based access to audit records.

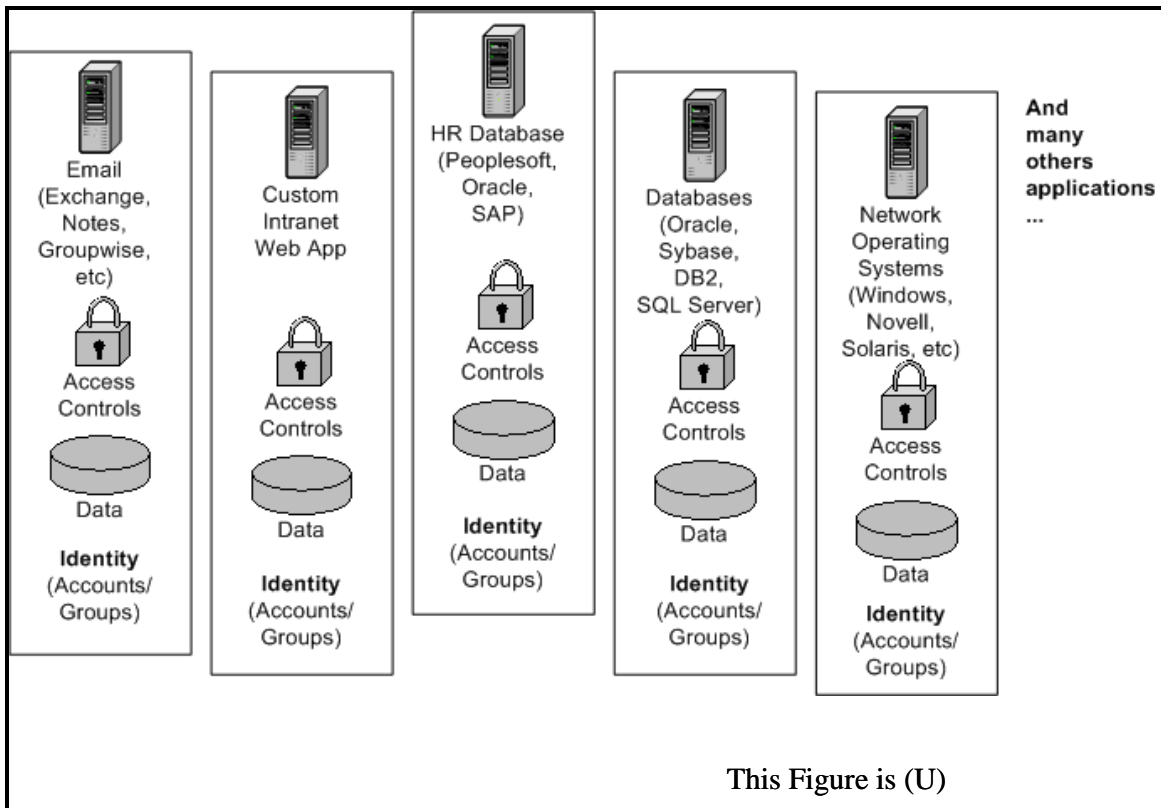
12199 **2.7.3 (U) Management of IA Mechanisms & Assets: Technologies**

12200 **2.7.3.1 (U) Identity Management**

12201 **2.7.3.1.1 (U) Technical Detail**

12202 (U) Identity management is essentially the process of creating and maintaining entity accounts
 12203 and credentials. Throughout an enterprise, an entity may have many identities; a user account on
 12204 a UNIX system, another account on the mail server, digital certificates for Secure Socket layer
 12205 (SSL) access, and a smart card for building access. Each identity is used for access control
 12206 decisions on each independent system. For example a UNIX account name will not mean
 12207 anything to the SSL-enabled web server.

12208 (U) Historically not only are the identities not related to each other, they are managed
 12209 independently as well. Independent management can lead to many problems for users and
 12210 enterprises. An entity will have to be enrolled and provisioned in each system to which it
 12211 requires access, a time consuming activity. Further, when an entity's access to the GIG has been
 12212 revoked (i.e., a person quits their job, a device is destroyed, etc.), each system needs to terminate
 12213 the entity's account. Account termination may be a very low priority activity, causing inactive
 12214 accounts to exist for some time after the user has left the enterprise. An example of the multiple
 12215 identities a user might have is shown in Figure 2.7-2.



12216

12217

Figure 2.7-2: (U) Example of Multiple Identities Assigned to a Single User

12218 (U) Identity Management provides a means to unify disparate identity data stores. By controlling
12219 identity information in a single location, user accounts need to be created only once. Users also
12220 have an easier time administering their own profiles and other account information as they only
12221 have one location to make modifications. Also, non-human entities that require identity
12222 information can be controlled in a more consistent and automated fashion. Further, access
12223 control (i.e., privilege management, Section 2.7.2.2) information can be bound to the identity,
12224 whether stored locally or remotely.

12225 (U) The scope of identity management includes the entire life cycle of an identity from creation,
12226 maintenance of information associated with an identity, revocation, and retiring of the identity.
12227 For named groups, identity management also includes updating the mapping of individual
12228 identities to the group. Identities must be persistent in the GIG and not expire, or be overwritten
12229 or reset by events in the GIG. In fact, the identity registered for an individual is unique and
12230 remains constant despite changes of that individual's name or other attributes.

12231 (U) Identity Management systems typically only handle identities within an enterprise. However,
12232 there may be times (such as when dealing with different programs) when identity information
12233 needs to be exchanged outside of the native enterprise boundaries. Federated management is the
12234 concept of a user being allowed to use the same identity across multiple enterprise identity
12235 management systems. For instance, a warfighter could sign into an external resource with the
12236 same identity information and credentials as he/she would normally use for their native
12237 resources.

12238 **2.7.3.1.2 (U) Usage Considerations**

12239 (U) Seamless integration of identity management comes at a cost. The enterprises that form a
12240 federation must trust each other. Effectively, one partner must trust the other in order to vouch
12241 for the validity of a given user. This type of trust may be a bit much for programs to bear in the
12242 initial years of integrated Identity Management use. As time goes on and Identity Management
12243 practices and standards evolve, there will likely be greater trust in the technologies and programs
12244 allowing Federated Identity Management to take hold.

12245 (U) Federated identity management is one of the biggest concerns when implementing identity
12246 management in the GIG. In order to make identity management grow to something larger than an
12247 enclave-level service, federations will need to be formed between programs, services, and
12248 agencies. Unfortunately, it is unclear where and how federations should be created. Should
12249 coalition partners be part of the GIG federation? Will multiple federations exist? How will these
12250 federations interoperate? These are important questions that will need to be answered as GIG
12251 programs integrate identity management.

12252 (U) The DoD has invested much money in the Common Access Card (CAC) system. CAC cards
12253 are a smartcard based systems for providing a unique identity for any entity within the DoD. The
12254 CAC platform provides most of the DoD with a common identity system that can be leveraged
12255 for GIG-wide identity systems.

12256 (U//FOUO) While not a perfect solution, the CAC system is a good first start. Presently CAC
12257 cards are not fully used as electronic identity tokens. They are still generally used as physical
12258 badges, since many of the back-office systems that could take advantage of CAC cards are still
12259 being developed. However, a program, started in summer 2004, called Federated Identity Cross -
12260 credentialing System/Defense Cross-credentialing Identification System (FiXs/DCIS) is
12261 attempting a large leap forward. FiX/DCIS, is a pilot program designed to use CAC tokens to test
12262 federated access between DoD programs and DoD contractors. The program, co-sponsored by
12263 the Defense Manpower Data Center and the Office of the Secretary of Defense, will provide
12264 practical insight into using CAC cards in an Identity Management system. More info can be
12265 found at <http://www.fegc.org/>.

12266 (U) The standards for identity management are still emerging. Identity management promises to
12267 be an important technology in the next decade. As such, many big industry and government
12268 organizations have stepped up to assist in standards development. However, as with any high-
12269 profile standards process, some vendors disagree on the technical details and end up creating
12270 separate and competing standards. This will continue to be a problem until the industry matures
12271 further.

12272 (U) SAML – Security Assertion Markup Language

12273 (U) From the SAML Technical Overview on oasis-open.org

12274 (U) “The Security Assertion Markup Language (SAML) standard defines a
12275 framework for exchanging security information between online business partners.

12276 (U) More precisely, SAML defines a common XML framework for exchanging
12277 security assertions between entities. As stated in the SSTC charter, the purpose of the
12278 Technical Committee is:

12279 (U) ...to define, enhance, and maintain a standard XML-based framework for
12280 creating and exchanging authentication and authorization information.

12281 (U) SAML is different from other security systems due to its approach of expressing
12282 assertions about a subject that other applications within a network can trust. What
12283 does this mean? To understand the answer, you need to know the following two
12284 concepts used within SAML.

12285 (U) Asserting party

12286 (U) The system, or administrative domain, that asserts information about a subject.
12287 For instance, the asserting party asserts that this user has been authenticated and has
12288 given associated attributes. For example: This user is John Doe, he has an email
12289 address of john.doe@acompany.com, and he was authenticated into this system using
12290 a password mechanism. In SAML, asserting parties are also known as SAML
12291 authorities.

12292 (U) Relying party

12293 (U) The system, or administrative domain, that relies on information supplied to it by
12294 the asserting party. It is up to the relying party as to whether it trusts the assertions
12295 provided to it. SAML defines a number of mechanisms that enable the relying party
12296 to trust the assertions provided to it. It should be noted that although a relying party

12297 can trust the assertions provided to it, local access policy defines whether the subject
12298 may access local resources. Therefore, although the relying party trusts that I'm John
12299 Doe – it doesn't mean I'm given carte blanche access to all resources.”

12300 (U) Available from <http://www.oasis-open.org/>

12301 (U) SPML – Service Provisioning Markup Language

12302 (U) SPML is intended to facilitate the creation, modification, activation, suspension,
12303 and deletion of data on managed Provision Service Targets (PSTs). It is the only real
12304 standard of import that deals explicitly with the act of provisioning. Provisioning is a
12305 core component of Identity Management, but unfortunately most of the standards
12306 work has been in the direction of privilege management.

12307 (U) Available from <http://www.oasis-open.org/>

12308 (U) XACML – eXtensible Access Control Markup Language

12309 (U) From [http://www.oasis-](http://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html)
12310 [open.org/committees/download.php/2713/Brief Introduction to XACML.html](http://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html)

12311 (U) “XACML is an *OASIS* standard that describes both a policy language and an
12312 access control decision request/response language (both written in XML). The policy
12313 language is used to describe general access control requirements, and has standard
12314 extension points for defining new functions, data types, combining logic, etc. The
12315 request/response language lets you form a query to ask whether or not a given action
12316 should be allowed, and interpret the result. The response always includes an answer
12317 about whether the request should be allowed using one of four values: Permit, Deny,
12318 Indeterminate (an error occurred or some required value was missing, so a decision
12319 cannot be made) or Not Applicable (the request can't be answered by this service).”

12320 (U) Available from <http://www.oasis-open.org/>

12321 (U) Liberty Alliance

12322 (U) The Liberty Alliance is an industry-created standards setting body. Project
12323 Liberty is largely concerned with Federated Identity Management. Their standards
12324 include ID-FF (the Identity Federation Framework), ID-WSF (Identity Web Service
12325 Framework), and ID-SIS (a collection of Identity Services Interface Specifications).

12326 (U) Available from <http://www.projectliberty.org/>

12327 **2.7.3.1.2.1 (U) Implementation Issues**

12328 (U//FOUO) Creation of GIG-wide Identity Management Schema – When implementing an
12329 identity management system, a schema describing users, their properties, and profiles must be
12330 created. This schema can vary significantly from enterprise to enterprise. For the GIG, a schema
12331 should be developed that encompasses DoD-wide needs. Further, systems need to be designed to
12332 handle potential future schema modifications. Whatever identity management schema that is
12333 developed in the near term will likely need revision after a few years of deployed use.

12334 (U) Evolving Standards – The standards revolving around identity management are still
12335 evolving. While the standards settings bodies are (generally) attempting to maintain backward
12336 compatibility, it is still critical to design systems that can adapt to the changes. From a software
12337 engineering standpoint, this underscores the need to write modular code that abstracts the user
12338 from the underlying standards. However, with the use of web services (a direction for most
12339 identity management systems) modularity is already a core construct, so changing standards
12340 should have less impact.

12341 (U//FOUO) Integration of Privilege Management – Identity management and privilege
12342 management go hand in hand (sometimes they are even referred to as the same concept). Due to
12343 the complexity of the GIG, these concepts get separate treatment since they could be managed at
12344 different levels. For example, GIG-wide identities may require a centrally controlled construct.
12345 However, privileges may be managed at a local level to support COIs. In any case, in order to be
12346 fully functional, an Identity Management system must integrate seamlessly with the Privilege
12347 Management system.

12348 (U//FOUO) Supporting Directory Infrastructure – An Identity Management system will need to
12349 have a supporting directory structure to store the identity information. Many programs already
12350 have existing installed directories, whether it be LDAP, Active Directory, NDS, or some other
12351 system. GIG-compliant programs may chose to either implement a new directory system from
12352 scratch or leverage existing infrastructures. Any directory system, however, must comply with
12353 the concept of least privilege for identity information stored in the directory store. That is, unlike
12354 the general concept of an open directory, Identity Management directories will contain
12355 information that is sensitive or classified in nature and must be protected as any other data store
12356 would be.

12357 (U//FOUO) Delegated and Dynamic Management – For an Identity Management system as large
12358 as will be required for the GIG, delegated management is an important yet difficult requirement.
12359 Depending on the situation (wartime vs. peacetime), location (in the Pentagon vs. in the field),
12360 and other factors, the scope and speed of changes to the identity management system by an actor
12361 may vary significantly. The identity management must reflect the chain of command in a service,
12362 allowing those in control of a warfighter or GIG entity to make changes (add privilege, add
12363 profiles, etc) to the identity information of that entity. Further, when there are large state changes
12364 (such as going to war), the Identity Management system will have to automatically and securely
12365 update privileges of entities to wartime privileges.

12366 **2.7.3.1.2.2 (U) Advantages**

12367 (U) Identity Management provides two major advantages to an enterprise. The first advantage is
12368 cost savings. Rather than create a user account in many systems, a user can be enrolled in a
12369 central identity management system. This cuts down the man-time it takes to get a user up and
12370 running in an enterprise. Further, the self-service aspect of an identity management system can
12371 allow users to manage their own profiles and credentials. This can reduce help desk calls for
12372 things like lost passwords and name changes.

12373 (U) The second advantage is security. By managing all users and entities through a common
12374 mechanism, policies can be applied uniformly across all actors. Accounts can be terminated in a
12375 timely manner. Access can be granted from a central location, enabling auditable policy
12376 enforcement. In general, Identity Management can get rid of the mish-mash of accounts and roles
12377 in an organization.

12378 **2.7.3.1.2.3 (U) Risks/Threats/Attacks**

12379 (U//FOUO) As with any system that tries to unify data storage and allow for distributed access,
12380 the movement to provide Identity Management in the GIG creates a new risk for the GIG.

12381 (U//FOUO) Identity Theft – By unifying identity information, even at a enclave level, identity
12382 management system can become a central location for hijacking an entity's identity. This is
12383 commonly called identity theft. However, in the context of the GIG, the ramifications are more
12384 severe for the enterprise than to an individual. An attacker who could subvert the identity
12385 management system could take on the identity of a trusted entity. This would allow the attacker
12386 to operate with the privileges of the subverted account.

12387 (U//FOUO) Denial of Service – Another concern as identity management becomes more
12388 pervasive is denial of service attacks. Many identity management architectures rely on a central
12389 host(s) to be available to either a) validate the identity, b) obtain a list of attributes or c) check to
12390 see if the identity token has been revoked. If the central hosts can be disabled, the identity
12391 management may cease to work. Generally speaking, this will cause problems throughout the
12392 system that is relying on the identity management system. Disabling the identity management
12393 system may affect a large number of systems. This makes the identity management infrastructure
12394 a weak point in the enterprise and should be protected as such.

12395 **2.7.3.1.3 (U) Maturity**

12396 (U) Currently, Identity Management is assessed a maturity level of Early (TRLs 1 - 3). While
12397 standards exist and there have been limited programs adopting the technology, it is not ready for
12398 deployment. The vast majority of what is dubbed Identity Management is actually privilege
12399 management. Privilege management is a more robust technology with many more vendors
12400 providing solutions today. However, strict identity management is immature.

12401 **2.7.3.1.4 (U) Standards**

12402 (U) The standards for identity management (Table 2.7-1) are still emerging. It promises to be an
12403 important technology in the next decade. As such, many big industry and government
12404 organizations have stepped up to assist in standards development. However, as with any high-
12405 profile standards process, some vendors disagree on the technical details and end up creating
12406 separate and competing standards. This will continue to be a problem until the industry matures
12407 further.

12408

Table 2.7-1 (U) Identity Management Standards

This Table is (U)	
Name	Description
OASIS Standards	
SAML Core	E. Maler et al. <i>Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML)</i> . OASIS, September 2003. Document ID oasis-sstc-saml-core-1.1. http://www.oasis-open.org/committees/security/ .
SAML Gloss	E. Maler et al. <i>Glossary for the OASIS Security Assertion Markup Language (SAML)</i> . OASIS, September 2003. Document ID oasis-sstc-saml-glossary-1.1. http://www.oasis-open.org/committees/security/ .
SAMLSec	E. Maler et al. <i>Security Considerations for the OASIS Security Assertion Markup Language (SAML)</i> , OASIS, September 2003, Document ID oasis-sstc-saml-sec-consider-1.1. http://www.oasis-open.org/committees/security/
SAMLReqs	Darren Platt et al., <i>SAML Requirements and Use Cases</i> , OASIS, April 2002, http://www.oasis-open.org/committees/security/ .
SAMLBind	E. Maler et al. <i>Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML)</i> . OASIS, September 2003. Document ID oasis-sstc-saml-bindings-1.1. http://www.oasis-open.org/committees/security/ .
SPML – Service Provisioning Markup Language	SPML is intended to facilitate the creation, modification, activation, suspension, and deletion of data on managed Provision Service Targets (PSTs). It is the only real standard of import that deals explicitly with the act of provisioning. Provisioning is a core component of Identity Management, but unfortunately most of the standards work has been in the direction of privilege management. http://www.oasis-open.org/committees/documents.php
SPML-Bind	OASIS Provisioning Services Technical Committee., <i>SPML V1.0 Protocol Bindings</i> , http://www.oasis-open.org/apps/org/workgroup/provision/download.php/1816/draft-pstc-bindings-03.doc , OASIS PS-
XACML – eXtensible Access Control Markup Language	From http://www.oasis-open.org/committees/download.php/2713/Brief Introduction to XACML.html
Liberty Alliance	
ID-FF	Identity Federation Framework Available from http://www.projectliberty.org/
ID WSF	Identity Web Service Framework Available from http://www.projectliberty.org/
ID SIS	Identity Services Interface Specifications Available from http://www.projectliberty.org/
This Table is (U)	

12409 2.7.3.1.5 (U) Cost/Limitations

12410 (U) Deployment of Identity Management systems can have high up-front costs. The initial
 12411 planning on how identities will be standardized within an enterprise will require coordination
 12412 from any party that will be affected by the transition. Applications need to be either retooled to
 12413 use the identity management system, or middleware needs to be used to interface legacy systems
 12414 with the Identity Management infrastructure.

12415 **2.7.3.1.6 (U) Dependencies**

12416 (U) In a vacuum, identity management is only marginally useful. It really becomes useful when
12417 coupled with Privilege Management to provide authorization information to applications. Any
12418 Identity Management system will rely on a Privilege Management system to provide real value.

12419 **2.7.3.1.7 (U) Alternatives**

12420 (U) The alternative to Identity Management is the status quo with respect to user and entity
12421 information. Some programs may choose to not integrate into a common identity management
12422 framework. While these programs may be able to avoid integration in the near-term based on
12423 cost and security considerations, eventually user requirements will drive the need for full
12424 integration as identity management matures. Users will expect to have a single interface for all
12425 account management. Further, GIG-wide CND mechanisms will be integrated into GIG Identity
12426 Management systems to enable centralized attack sensing and defense. Systems not leveraging
12427 the GIG identity management system may lose these protections.

12428 **2.7.3.1.8 (U) Complementary Techniques**

12429 (U) Managing identity is only part of the entire access control domain. Privilege must also be
12430 managed in order to complete the picture. It is not enough to simply prove who an entity is; a
12431 system must be able to provide authorization for that entity to perform actions. Managing this
12432 authorization is the core of privilege management and integral with any systems Identity
12433 Management architecture.

12434 **2.7.3.1.9 (U) References**

12435 (U) [Reed] The Definitive Guide to Identity Management; Reed, Archie;
12436 Realtimepublishers.com; 2002-2004.

12437 **2.7.3.2 (U) Privilege Management**

12438 **2.7.3.2.1 (U) Technical Detail**

12439 (U) The goal of privilege management is to allow fluid access to legitimate resources. Privilege
12440 management technologies had not evolved significantly until only recently. That is because
12441 collaborative networks and computing environments have grown very large and complex within
12442 the past decade, consequently requiring improved solutions towards identification,
12443 authentication, and authorization solutions.

12444 (U) In the early computing environment days and even up to the present day, privileges were
12445 implemented via ACLs that were commonly found on the major operating systems. They were
12446 used to control access to files, directories, and services on either local hosts or mainframes. As
12447 the computing environments evolved, there was a progression towards utilizing scripts to
12448 automate rules towards managing a user's privileges to resources. More recently, during the past
12449 decade in particular, we have seen an even more sophisticated drive towards concepts of Role-
12450 based Access Control (RBAC) and even infrastructure-based concepts such as PMI. These
12451 significant concepts as well as the standards and technologies that surround them are described
12452 below.

12453 **2.7.3.2.1.1 (U) Rules-Based Authorization Schemes**

12454 (U) Rules are provided as run-time processes that dynamically determine outcome based on
12455 privileges. Rules can include complex Boolean operations, using an interpretive language or
12456 scripting language to define rules.

12457 (U) Instead of aggregating all permissions within predefined roles (roles are detailed in the next
12458 sub-section), some enterprises have chosen to take advantage of rule-based processing
12459 capabilities of provisioning systems and WAM (Web Access Management) products. Rule-
12460 processing engines examine and evaluate user attributes and privileges, and make outcome
12461 decisions on the fly. This functionality permits more dynamic actions to be taken during
12462 processing instead of relying on the ability to map out every possibility in advance. Rule-based
12463 processing may be more dynamic than roles, but at the same time requires that business
12464 processes be accurately understood.

12465 **2.7.3.2.1.2 (U) Roles-Based Authorization Schemes**

12466 (U) More recently, research has focused on RBAC. In the basic RBAC model, a number of roles
12467 are defined. These roles typically represent organizational roles such as secretary, manager,
12468 employee, etc. In the authorization policy, each role is given a set of permissions, i.e., the ability
12469 to perform certain actions on certain targets. Each user is then assigned to one or more roles.
12470 When accessing a target, a user presents his role(s), and the target reads the policy to see if this
12471 role is allowed to perform the action.

12472 (U) There are several fairly new standards to choose from; however, there are minimal
12473 implementations or compatibility with the standards to really make them useful. The National
12474 Institute of Standards and Technology (NIST) offers an RBAC reference model and The
12475 Organization for the Advancement of Structured Information Standards (OASIS) offers
12476 eXtensible Access Control Markup Language (XACML)—an XML specification for expressing
12477 policies for information access over the Internet.

12478 (U) The NIST core RBAC offers a good overview of what is desired in an RBAC solution at
12479 <http://csrc.nist.gov/publications/nistbul/csl95-12.txt>.

12480 (U) The NIST component defines five basic data elements:

- 12481 • (U) Users – An entity that uses the system
- 12482 • (U) Roles – A job function within the context of an organization
- 12483 • (U) Permissions – Approval to perform an operation on one or more objects
- 12484 • (U) Objects – Can be many things; for example, an entry in a target system (such as an
12485 account), a network resource (a printer), an application (a procurement), a policy
12486 (password policies), and so on
- 12487 • (U) Operations – Various and unbounded but including customer-defined workflow
12488 processes such as a password reset, the addition, modification, or removal (deletion) of
12489 user accounts, and specific data about those accounts; importantly, it should be possible
12490 to delegate these operations to other users

12491 (U) Hierarchical RBAC

12492 (U) Hierarchical RBAC requires the support of role hierarchies, whereby senior roles acquire the
12493 permissions of their juniors, and junior roles acquire the user membership of their seniors.

12494 (U) The NIST standard recognizes two types of role hierarchies.

- 12495 • (U) General Hierarchical RBAC—Arbitrary orders and relationships between roles serve
12496 as the role hierarchy.
- 12497 • (U) Limited Hierarchical RBAC—Restrictions are placed on the role hierarchy. Typically
12498 hierarchies are limited to simple structures such as trees or inverted trees.

12499 (U) Although General Hierarchical RBAC introduces potential problems of hierarchy loop
 12500 detection and prevention, it is seen as the most useful. In an RBAC solution, consider that
 12501 occupants of the same roles at different locations in an organization will need access to different
 12502 underlying systems. This allows the same role (say, development engineers) to be given access to
 12503 different systems based on differing values in the role occupant’s profile. So while all
 12504 development engineers need access to source control, it is likely that those in one office or
 12505 working on one product may need access to a different source control system from those in
 12506 another office or working on a different project. To solve this problem, a parameterized
 12507 permission object can be used. A single permission Source Control Access might be used.
 12508 However the mappings from that object into the connected systems (that is, source control
 12509 systems) would vary based on a user’s location attribute or on the project attribute.

12510 **2.7.3.2.1.3 (U) Privilege Management Infrastructure (PMI)**

- 12511 • (U) PMI is the information security infrastructure that assigns privilege attribute
 12512 information, such as privilege, capability, and role, to users and issues. One options is to
 12513 manages manage privileges by using the X.509 Attribute Certificate (AC). The function
 12514 of the PMI is to specify the policy for the attribute certificate issuance and management.
 12515 Then, the PMI carries out the AC-related management functions, such as issuing,
 12516 updating, and revoking an attribute certificate based on a specified policy.

12517 (U) Although Attribute Certificates were first defined in X.509(97), it was not until the fourth
 12518 edition of X.509 (ISO 9594-8:2001) that a full PMI for the use of attribute certificates was
 12519 defined. A PMI enables privileges to be allocated, delegated, revoked, and withdrawn
 12520 electronically. A PMI is to authorization what a PKI is to authentication. Table 2.7-2 summarizes
 12521 these relationships.

12522 **Table 2.7-2: (U) Comparisons of PKI and PMI**

This Table is (U)		
Concept	PKI Entity	PMI Entity
Certificate	Public Key Certificate (PKC)	Attribute Certificate (AC)
Certificate Issuer	Certificate Authority (CA)	Attribute Authority (AA)
Certificate User	Subject	Holder
Certificate Binding	Subject’s Name to Public Key	Holder’s Name to Privilege Attribute(s)
Revocation	Certificate Revocation List (CRL)	Attribute Certificate Revocation List (ACRL)
Root of Trust	Root Certification Authority or Trust Anchor	Source of Authority (SOA)
Subordinate Authority	Subordinate Certification Authority	Attribute Authority (AA)
This Table is (U)		

12523 (U) A public key certificate (PKC) is used for authentication and maintains a strong binding
 12524 between a user's name and his public key, while an attribute certificate (AC) is used for
 12525 authorization and maintains a strong binding between a user's name and one or more privilege
 12526 attributes. The entity that digitally signs a public key certificate is called a CA, while the entity
 12527 that digitally signs an attribute certificate is called an Attribute Authority (AA). The root of trust
 12528 of a PKI is sometimes called the root CA while the root of trust of the PMI is called the Source
 12529 of Authority (SOA). CAs may have subordinate CAs that they trust, and to which they delegate
 12530 powers of authentication and certification. Similarly, SOAs may delegate their powers of
 12531 authorization to subordinate AAs. If a user needs to have his signing key revoked, a CA will
 12532 issue a Certificate Revocation List (CRL). Similarly if a user needs to have his authorization
 12533 permissions revoked, an AA will issue an attribute certificate revocation list (ACRL).

12534 (U) PMI systems need to provide the following functionalities:

- 12535 • (U) Assigning attributes to a user
- 12536 • (U) Creating an X.509 attribute certificate
- 12537 • (U) Issuing, updating, revoking, searching, and publishing attribute certificate
- 12538 • (U) Validating an attribute certificate and making an access control decision
- 12539 • (U) Supports ID/Password or PKC authentication method
- 12540 • (U) Applying RBAC model to access control framework
- 12541 • (U) Supports push/pull model in an attribute certificate usage
- 12542 • (U) Supports flexible system architecture by providing independent DMS.

12543 **2.7.3.2.2 (U) Usage Considerations**

12544 **2.7.3.2.2.1 (U) Implementation Issues**

12545 (U) There are various implementations between Authority Management, Policy Management,
 12546 and other components within the PMI.

12547 (U) X.509 supports simple RBAC by defining role specification attribute certificates that hold
 12548 the permissions granted to each role, and role assignment attribute certificates that assign various
 12549 roles to the users. In the former case, the AC holder is the role, and the privilege attributes are
 12550 permissions granted to the role. In the latter case the AC holder is the user, and the privilege
 12551 attributes are the roles assigned to the user.

12552 (U) Another extension to basic RBAC is constrained RBAC. This allows various constraints to
 12553 be applied to the role and permission assignments. One common constraint is that certain roles
 12554 are declared to be mutually exclusive, meaning that the same person cannot simultaneously hold
 12555 more than one role from the mutually exclusive set. Another constraint might be placed on the
 12556 number of roles a person can hold or on the number of people who can hold a particular role.

12557 (U) X.509 only has a limited number of ways of supporting constrained RBAC. Time constraints
12558 can be placed on the validity period of a role assignment attribute certificate. Constraints can be
12559 placed on the targets at which a permission can be used and on the policies under which an
12560 attribute certificate can confer privileges. Constraints can also be placed on the delegation of
12561 roles. However many of the constraints (e.g., the mutual exclusivity of roles) have to be enforced
12562 by mechanisms outside the attribute certificate construct (i.e., within the privilege management
12563 policy enforcement function).

12564 **2.7.3.2.2.2 (U) Advantages**

12565 (U)The challenges of role-based access control will continue to be the contention between strong
12566 security and easier administration. For stronger security, it is better for each role to be more
12567 granular—thus to have multiple roles per user. For easier administration, it is better to have
12568 fewer roles to manage.

12569 (U) The creation of rules and security policies is also a complex process. Depending on the
12570 situation within the enterprise, there will be a need to strike an appropriate balance between the
12571 two.

12572 (U) PMI-based solutions have the advantage of existing or emerging infrastructures such as PKI.
12573 But on the other hand, management schemes for PMI and the attribute certificates are considered
12574 to be complex and more challenging compared to the other privilege management schemes.

12575 **2.7.3.2.2.3 (U) Risks/Threats/Attacks**

12576 (U//FOUO) PMI will need to provide complete and accurate audit of authorization activity. It
12577 will be necessary to have both attributable receipts of the transaction, and of the contributing
12578 activities, such as the authorization decision.

12579 (U//FOUO) XML is a good prospect for creating these receipts, digitally signing them, and
12580 submitting them to a notarization service to enhance their non-repudiation ability. Vendors are
12581 currently providing capabilities for creating, storing, and managing non-repudiated records via
12582 XML encryption and digital signature.

12583 **2.7.3.2.3 (U) Maturity**

12584 (U) Privileges, and specifically Attribute Certificates, have some basis as an X.509 extended
12585 standard. However, the management of rules and roles-based access specifications are still left to
12586 proprietary implementations. There are enabling technologies such as SAML and XACML to
12587 assist in the development of RBAC as well as rules-based applications, and some COTS vendors
12588 have in fact implemented solutions based on these underlying technologies. Given the
12589 availability of prototypes that prove working concepts, this technology is assessed as Emerging
12590 (TRLs 4 - 6).

12591 **2.7.3.2.4 (U) Standards**

12592 The Privilege Management Standards are listed in Table 2.7-3.

12593

12594

Table 2.7-3: (U) Privilege Management Standards

This Table is (U)	
Name	Description
IETF Standards	
RFC3281	S. Farrell, R. Housley, “An Internet Attribute Certificate Profile for Authorization“, IETF RFC, April 2002
ISO Standards	
ISO/IEC 9594-8	ITU-T Rec. X.509 (2000) ISO/IEC 9594-8 The Directory: Authentication Framework
This Table is (U)	

12595

2.7.3.2.5 (U) Dependencies

12596

2.7.3.2.5.1 (U) Relationship of Authorization to Identity

12597

(U) Authorization policies are rules for determining which subjects are allowed to access resources. In some cases, privacy considerations may require that some form of anonymous or pseudonymous access be supported. In most cases, however, users must first be identified in order to receive authorization to access resources.

12598

12599

12600

12601

(U) An identity system is therefore critical to establishing users’ identities as the basis for authorizing access to resources. The identity infrastructure binds a unique name or identifier to a user. It also maintains a set of attributes (often in a general-purpose directory service) that supports the authentication and authorization processes. These attributes could include not only credentials, such as hashed passwords or X.509 certificates, but also information about the user, which could be referenced in an access rule.

12602

12603

12604

12605

12606

12607

2.7.3.2.5.2 (U) Standards Development

12608

(U) Three classes of standards specifications can improve the interoperability of policy-based management systems:

12609

12610

- (U) Schemas: provide standardization in the way groups, roles, rules, and resources are described in directories and other repositories

12611

12612

- (U) Protocols: enable interoperability of policy decision requests and policy distribution across products from multiple vendors

12613

12614

- (U) Languages: provide means of codifying policy rules.

12615

(U) The lack of policy standards has been a significant barrier to building interoperable authorization systems, but now a number of standards are being developed. In fact, several standards have recently emerged—such as SAML, XACML, Web Services Policy (WS-Policy), and XrML—creating the risk that standards will overlap.

12616

12617

12618

12619 **2.7.3.2.5.3 (U) SAML – Enabling Technology**

12620 (U) SAML is intended to provide a session-based security solution for authentication and
12621 authorization across disparate systems and organizations through the use of XML SAML, which
12622 was also described earlier with Identity Management. SAML, in addition to authentication
12623 assertion, also provides Authorization Assertion, which implies the system can assert that a
12624 subject is authorized to access the object. The SAML specification also enables protocols to send
12625 and receive messages, as well as specify bindings that define how SAML message exchanges are
12626 mapped to SOAP exchanges. SAML can use multiple protocols, including HTTP, Simple Mail
12627 Transfer Protocol (SMTP), File Transfer Protocol (FTP), and SOAP.

12628 **2.7.3.2.5.4 (U) XACML – Enabling Technology**

12629 (U) SAML enables PEP-to-PDP or PDP-to-PDP communication of requests and responses for
12630 authentication, attributes, and authorization. However, SAML does not define detailed semantics
12631 for the data it carries. Roles in SAML, for example, are only text strings. OASIS left it to
12632 XACML to provide the details for attribute information or authorization information. XACML
12633 fulfills SAML's needs by providing richer semantic constructs for authorization information.
12634 Among other things, it enables use of common LDAP attributes in XML-based security
12635 protocols. But XACML does much more than this.

12636 **2.7.3.2.6 (U) Complementary Technologies**

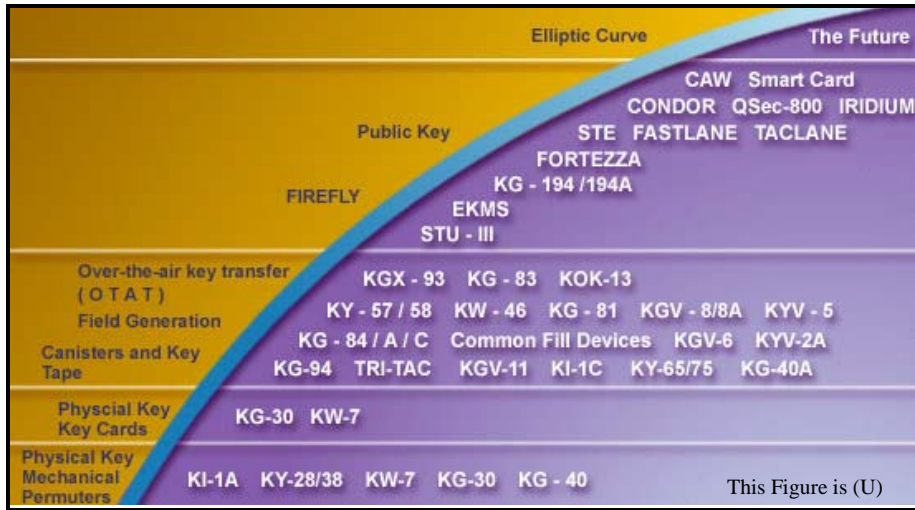
12637 (U) Policy rules form the basis for authorization. As an extension of policy-based security,
12638 management, and networking within a single organization, standards groups (such as, the
12639 Distributed Management Task Force [DMTF], the TeleManagement Forum [TMF], the Open
12640 Group, OASIS, and a group of vendors standardizing Web services) have all been working on
12641 the definition of standardized policy objects that can be used to create or negotiate agreements,
12642 make decisions, or carry out obligations.

12643 **2.7.3.3 (U) Key Management**

12644 **2.7.3.3.1 (U) Technical Detail**

12645 **2.7.3.3.1.1 (U) Evolution of Key-based Equipment Technology**

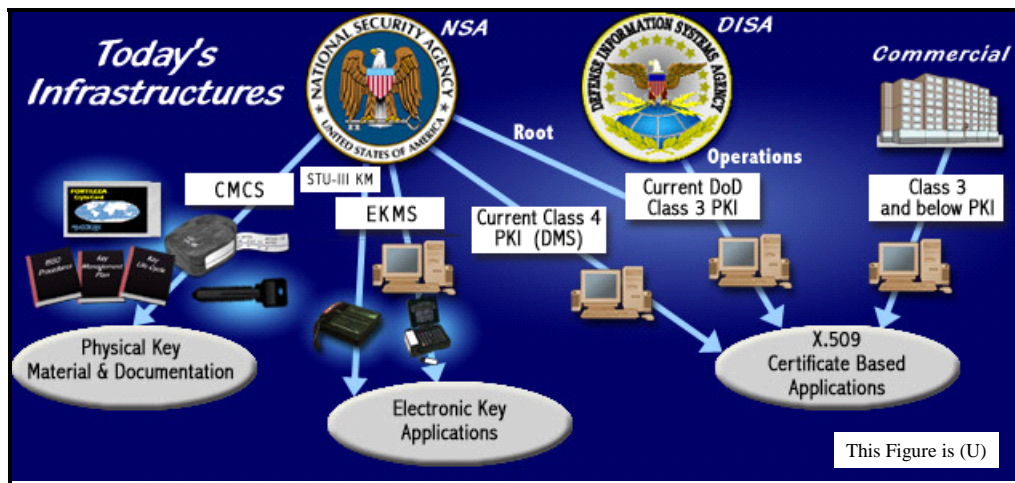
12646 (U) The following are supported ECUs and their associated technologies that have evolved over
 12647 the past few decades. This can be seen in Figure 2.7-3.



12648

12649 **Figure 2.7-3: (U//FOUO) ECU and Technology Evolution**

12650 (U//FOUO) There is a growing need for the DoD and government enterprises to reexamine
 12651 existing approaches that provision cryptographic key products and services for military,
 12652 intelligence, governments, allied, contracting and business customers. It is no longer feasible or
 12653 cost effective to design, develop, and field unique, independent key and certificate management
 12654 systems to support the various classes of cryptographic products, as seen in Figure 2.7-4.



12655

12656 **Figure 2.7-4: (U) Current Key Management Infrastructures**

12657 (U//FOUO) Advances in information technology are giving the customer a wider array of
12658 communication choices, while simultaneously necessitating wide ranges of IA solutions that are
12659 equal to the task. The current key management environment is made up of separate and
12660 independent infrastructures that provide and manage their own set of security products. These
12661 systems will become increasing cumbersome and costly as new technology and their attendant
12662 security solutions continue to advance and the resources needed to operate them decline. This
12663 key management environment, shown in Figure 2.7-4, is comprised of several unique solutions
12664 built for specific product lines. While the solutions satisfy unique security needs, they each
12665 require different tools and training in order to obtain their respective products and services,
12666 imposing an unwarranted strain on resources.

12667 (U//FOUO) Adding a new key management capability has frequently meant creating a new,
12668 independent system to support it. The most recent example is in the public key certificate arena
12669 where independent infrastructures are being deployed to meet the demand created by the use of
12670 PKI-based security products. Continuing this approach will increasingly tax resources
12671 throughout the community.

12672 (U) Several of the systems in Figure 2.7-4 have been in existence for a number of years and are
12673 in need of upgrade to take advantage of recent advances in communication technology. This
12674 technology area has advanced significantly in recent years, providing the market place with
12675 many new and worthwhile, applicable techniques that would greatly improve efficiency and
12676 performance.

12677 (U//FOUO) Although created independently, the existing systems contain many common threads
12678 (e.g., registration, ordering, and distribution) that could logically be combined and offered as a
12679 unified set of processes. Not only has the key management community recognized this fact, so
12680 has the DoD Joint Staff. They have identified a unified Key Management Infrastructure (KMI) as
12681 a critical infrastructure needed to support key and certificate management approaches for mission
12682 critical, logistic, and administrative systems.

12683 (U//FOUO) Given the critical importance of key management and the state of the current key
12684 management systems, the focus should be on developing a singular approach, using sound IA
12685 principles and modern technology.

12686 **2.7.3.3.1.2 (U) Vision of the KMI**

12687 (U//FOUO) Consequently, the NSA has launched the KMI Strategic and Architectural Planning
12688 initiative, supported by Service, Joint Staff, and contractor personnel. The KMI initiative will
12689 focus on unifying the disparate key management systems within a single, modern architecture—
12690 one that is modular, flexible, and extensible. Unification will eliminate redundant resources
12691 associated with operation, maintenance, and training that will result in substantial cost savings.

12692 (U//FOUO) The KMI will be the primary means to support the many current and future
 12693 cryptographic products and services needed to conduct secure electronic transactions. Security
 12694 services such as identification and authentication, access control, integrity, non-repudiation, and
 12695 confidentiality become increasingly critical as the government transitions to an electronic
 12696 environment. The KMI provides a means for the secure creation, distribution and management of
 12697 the cryptographic products that enable these services for a wide variety of missions as seen in
 12698 Figure 2.7-5.

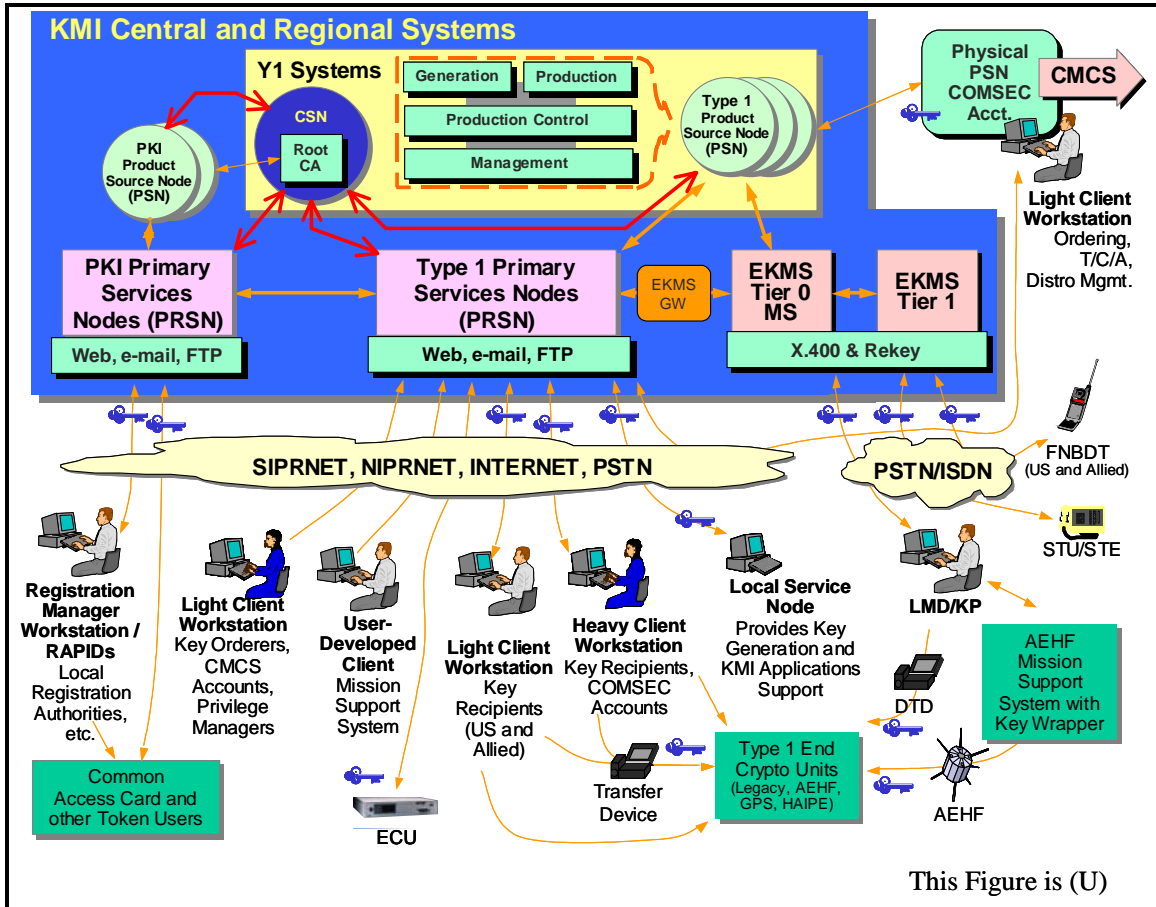


Figure 2.7-5: (U//FOUO) KMI – Envisioned Infrastructure

2.7.3.3.1.3 (U) Scope of KMI

(U//FOUO) The current key management systems service a wide variety of Departments, Services, Agencies, and Organizations within the U.S. Government and those of its allies. The common characteristic of these customers is their need to protect classified or mission critical SBU information or to inter-operate with U.S. components in doing so. As the KMI initiative evolves, its architecture will be designed, at a minimum, to continue the support of these customers’ traditional requirements, as well as their growing information assurance needs for the less sensitive, but important, unclassified operational information. This means providing everything from Type 1 netted key or Class 5 certificates for classified applications to commercial Class 3 or 2 certificates for lesser needs.

12711 (U//FOUO) Support different infrastructures such as:

- 12712 • (U//FOUO) COMSEC Material Control System (CMCS)
- 12713 • (U//FOUO) Electronic Key Management System (EKMS)
- 12714 • (U) PKI
- 12715 • (U) Government - Class 4
- 12716 • (U) Government - Class 3
- 12717 • (U) Commercial
- 12718 • (U//FOUO) STU-III Infrastructure.

12719 (U//FOUO) It is anticipated that requirements for support of classified applications will continue
12720 to grow as new Type 1 solutions, such as secure wireless and Global Positioning System
12721 modernization, are implemented. It is the intent of the KMI to enhance the DoD's capability to
12722 support these mission-critical requirements.

12723 (U//FOUO) It is projected that there will be a significant increase within the DoD in the use of
12724 cryptographic applications for the conduct of unclassified and sensitive but unclassified (SBU)
12725 business transactions. Many of these applications will be obtained from commercial sources,
12726 with their keying and management services being supplied by the evolving DoD PKI or
12727 commercial service providers.

12728 (U//FOUO) Today, the DoD is fielding two independent PKI systems supporting different
12729 assurance levels. The MISSI, or High Assurance PKI (HAPKI), supports security applications
12730 that handle medium to high value information in any environment, and the DoD. The Medium
12731 Assurance PKI (MAPKI) supports security applications that handle medium value information in
12732 a low to medium-risk environment.

12733 **2.7.3.3.1.4 (U) Key Components of a KMI**

12734 2.7.3.3.1.4.1 (U) Central Oversight Authority

12735 (U//FOUO) The central oversight authority is the entity that provides overall key and data
12736 synchronization, as well as system security oversight for an organization or set of organizations.
12737 The central oversight authority: 1) coordinates protection policy and practices (procedures)
12738 documentation, 2) might function as a holder of data provided by service agents, and 3) serves as
12739 the source for common and system level information required by service agents (e.g., keying
12740 material and registration information, directory data, system policy specifications, and system
12741 wide key compromise and certificate revocation information). As required by survivability or
12742 continuity of operations policies, central oversight facilities may be replicated at an appropriate
12743 remote site to function as a system back up.

12744 2.7.3.3.1.4.2 (U//FOUO) Key Processing Facilities

12745 (U//FOUO) Key processing services typically include the following services:

- 12746 • (U) Acquisition or generation of public key certificates (where applicable)
- 12747 • (U//FOUO) Initial generation and distribution of keying material
- 12748 • (U) Maintenance of a database that maps user entities to an organization's certificate/key
12749 structure
- 12750 • (U//FOUO) Maintenance and distribution of nodal CKLs and/or CRLs
- 12751 • (U) Generation of audit requests and the processing of audit responses as necessary for
12752 the prevention of undetected compromises.

12753 (U//FOUO) An organization may use more than one key processing facility to provide these
12754 services (e.g., for purposes of inter-organizational interoperation). Key processing facilities can
12755 be added to meet new requirements or deleted when no longer needed and may support both
12756 public key and symmetric key establishment techniques.

12757 (U) Where public key cryptography is employed, the organization operating the key processing
12758 facility will generally perform most PKI registration authority, repository, and archive functions.
12759 The organization also performs at least some PKI certification authority functions. Actual X.509
12760 public key certificates may be obtained from a government source (certification authorities
12761 generating identification, attribute, or encryption certificates) or a commercial external
12762 certification authority (usually a commercial infrastructure/CA that supplies/sells X.509
12763 certificates). Commercial external certification authority certificates should be cross-certified by
12764 a government root CA.

12765 2.7.3.3.1.4.3 (U) Service Agents

12766 (U//FOUO) Service agents support organizations' KMIs as single points of access for other KMI
12767 nodes. All transactions initiated by client nodes are either processed by a service agent or
12768 forwarded to other nodes for processing. Service agents:

- 12769 • (U//FOUO) Direct service requests from client nodes to key processing facilities, and
12770 when services are required from multiple processing facilities, coordinate services among
12771 the processing facilities to which they are connected
- 12772 • (U//FOUO) Are employed by users to order keying material and services, retrieve keying
12773 material and services, and manage cryptographic material and public key certificates
- 12774 • (U//FOUO) Might provide cryptographic material and certificates by using specific key
12775 processing facilities for key and certificate generation
- 12776 • (U//FOUO) Might provide registration, directory, and support for data recovery services
12777 (i.e. key recovery), as well as provide access to relevant documentation, such as policy
12778 statements and infrastructure devices
- 12779 • (U//FOUO) Might process requests for keying material (e.g., user identification
12780 credentials), and assign and manage KMI user roles and privileges
- 12781 • (U//FOUO) Might also provide interactive help desk services as required

12782 (U//FOUO) A service agent who supports a major organizational unit or geographic region may
12783 either access a central or inter-organizational key processing facility or use local, dedicated
12784 processing facilities—as required—to support survivability, performance, or availability,
12785 requirements (e.g., a commercial external Certificate Authority).

12786 2.7.3.3.1.4.4 (U) Client Nodes

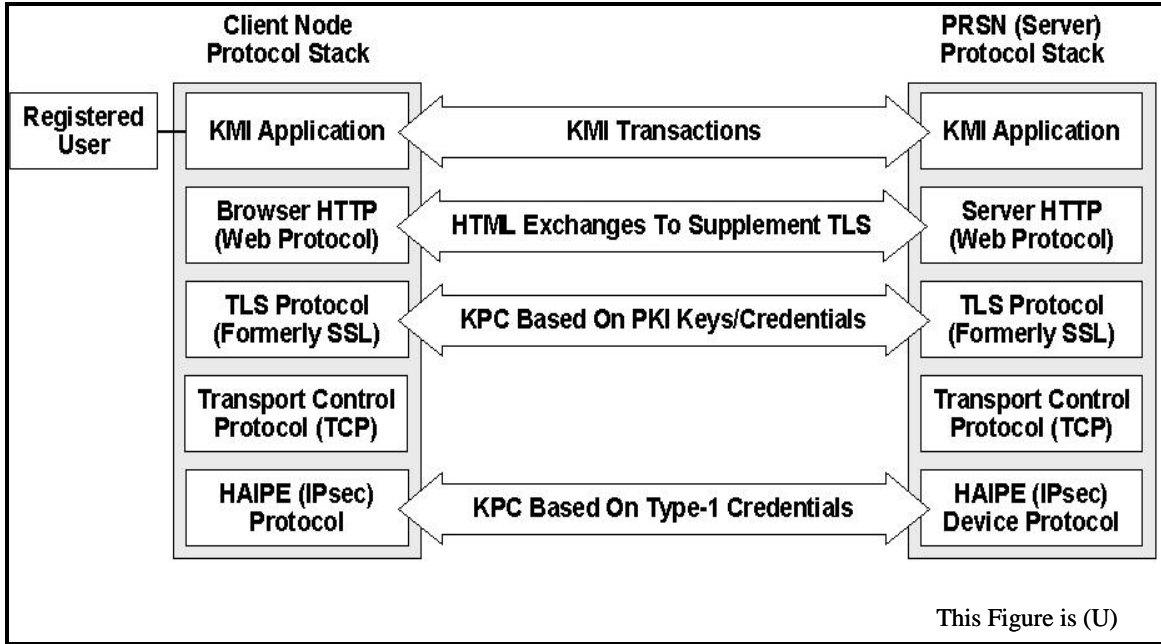
12787 (U//FOUO) Client nodes are interfaces for managers, devices, and applications to access KMI
12788 functions, including the requesting of certificates and other keying material. They may include
12789 cryptographic modules, software, and procedures necessary to provide user access to the KMI.
12790 Client nodes:

- 12791 • (U//FOUO) Interact with service agents to obtain cryptographic key services
- 12792 • (U//FOUO) Provide interfaces to end user entities (e.g., encryption devices) for the
12793 distribution of keying material, for the generation of requests for keying material, for the
12794 receipt and forwarding (as appropriate) of CKLs and CRLs for the receipt of audit
12795 requests, and for the delivery of audit responses
- 12796 • (U//FOUO) Typically initiate requests for keying material in order to synchronize new or
12797 existing user entities with the current key structure, and receive encrypted keying
12798 material for distribution to end-user cryptographic devices (in which the content—the
12799 unencrypted keying material—is not usually accessible to human users or user- node
12800 interface processes).
- 12801 • (U//FOUO) Can be a FIPS 140-2 compliant workstation executing KMI security software
12802 or a FIPS 140-2 compliant special purpose device.

12803 (U//FOUO) Actual interactions between a client node and a service agent depend on whether the
12804 client node is a device, a manager, or a functional security application.

12805 (U) Protection in KM Layers

12806 (U//FOUO) Key Management (KM) layers that correspond to the Open Systems Interconnection
12807 (OSI) model, require assurance while exchanging data on the network between client systems
12808 and the server. One model with the KMI initiative, seen in Figure 2.7-6, depicts using several
12809 standardized protocols for security such as IPsec, TLS, and HTTP-S.



12810

12811

Figure 2.7-6: (U//FOUO) KMI Protected Channel Layers

12812

2.7.3.3.1.5 (U) XML Key Management Services

12813

(U) Online key registration, issuance, distribution, validation, and revocation services are a core feature of any network trust environment.

12814

12815

(U) Under the XKMS initiative (draft specification available at <http://www.w3.org/TR/xkms/>), the PKI industry is defining a set of XML-based services, protocols, and formats for distributing and registering public keys to support various cryptographic services—including authentication, authorization, digital signatures, content encryption, and session encryption. Principally defined by VeriSign, Microsoft, and webMethods, XKMS has already been endorsed by leading PKI providers, including VeriSign, Baltimore, Entrust, and RSA. The specification was submitted on March 30, 2001, as a Technical Note to the W3C, which has not yet created a standards-track working group to develop the specification (although creation of a formal W3C working group is likely by the end of 2004).

12816

12817

12818

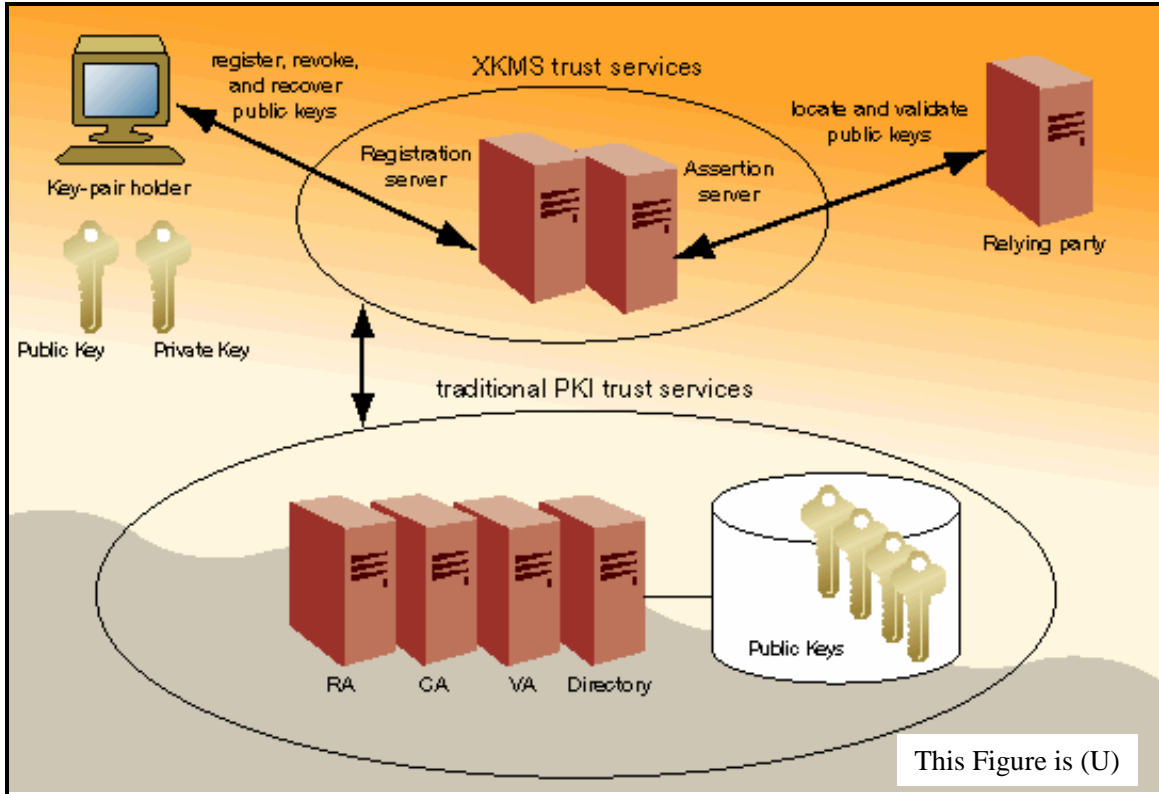
12819

12820

12821

12822

12823



12824

12825

Figure 2.7-7: (U) XKMS Environment

12826

12827

12828

12829

12830

(U) The XKMS framework consists of two services: the XML Key Registration Service Specification (X-KRSS) and the XML Key Information Service Specification (X-KISS). Registration Servers are at the heart of X-KRSS, while Assertion Servers are the hub of the X-KISS environment. Figure 2.7-7 shows a high-level functional topology of an XKMS environment that supports both the X-KRSS and X-KISS services.

12831

12832

12833

12834

12835

12836

12837

(U) XKMS defines a SOAP/XML-messaging-based alternative to traditional PKI, though in many ways XKMS is designed to complement, rather than replace, established PKI standards. At the client level, XKMS defines mechanisms under which applications delegate the retrieval, parsing, and validation of X.509 digital certificates to trusted servers, thereby streamlining the configuration of client-side, trust-service business logic. XKMS requires retrofitting of today's clients and applications to support—at a minimum—such standards as SOAP, XML-DSig, XML Schemas, XML Namespaces, WSDL, and XML Encryption.

12838

12839

12840

12841

12842

12843

12844

(U) The Registration Servers and Assertion Servers support all traditional PKI functions, but do so through the exchange of standardized, digitally signed XML-based messages with PKI-enabled clients. XKMS servers and clients digitally sign every message they exchange with each other via formats and mechanisms defined under XML-DSig. XKMS clients are set up to explicitly trust specific Registration and Assertion Servers, and will accept trust assertions (such as messages containing registered public keys) only if they contain valid digital signatures from those trusted servers.

12845 (U) When deployed into a traditional X.509 PKI environment, the XKMS-enabled servers would
 12846 integrate with traditional infrastructure services—including Registration Authorities, Certificate
 12847 Authorities, and Validation Authorities—through established PKI X.509 (PKIX) protocols.
 12848 However, the XKMS framework does not specify the need to interoperate with any external PKI.
 12849 It can in fact interoperate with PKIX, PGP, or Simple PKI (SPKI) environments for such
 12850 services as registering, issuing, validating, and revoking digital certificates.

12851 **2.7.3.3.1.6 (U) Constructive Key Management (CKM)**

12852 (U) CKM technology is a standards-based (X9.69 and X9.73) and patented cryptographic key
 12853 management technology that resolves critical information security and information management.
 12854 As more information is being created, transmitted, and stored in digital format, there is a higher
 12855 percentage of information that needs to be secured. Further, the need has never been greater to
 12856 identify authorized users, protect and control sensitive information assets, and restrict access to
 12857 information in compliance with privacy statutes and regulations.

12858 (U) CKM is also an authorization management system that provides logical access to individual
 12859 objects. This access is enforced through encryption in a manner that efficiently supports a variety
 12860 of applications, such as:

- 12861 • (U) Dynamic, Assured Information Sharing
- 12862 • (U) Collaboration among Communities of Interest
- 12863 • (U) Digital Rights Management
- 12864 • (U) Critical Infrastructure Protection
- 12865 • (U) Liability Mitigation through Assured Enforcement
- 12866 • (U) Data Separation
- 12867 • (U) Defined Access Control to Information by Content

12868 (U) CKM provides Cryptographically Enforced Management of Keys, Objects, and Access.
 12869 CKM's Object Level Access Control (OLAC) techniques allow users to control anything that can
 12870 be named, from a character, page, image, or sound in a document to a field in a database. In
 12871 addition, CKM's RBAC techniques cryptographically enforce who should be able to see which
 12872 piece of data or information. The approach of differentially encrypting data based on the need-to-
 12873 know or need-to-share principle allows secure communication among groups of individuals with
 12874 a variety of roles. Those individuals who have a legitimate need to view information have access
 12875 to it, while others do not.

12876 (U) When encrypting with CKM, users label information with Credentials that define the rights
 12877 required to access the information. Users holding matching Credentials will be able to decrypt
 12878 the information while those who do not will be unable to view the information. For example, a
 12879 document may be labeled Proprietary or Sensitive, and it may be labeled to require certain other
 12880 Credentials. Users' Credentials are stored on Smart Tokens, which can be soft tokens or hard
 12881 tokens (such as smart cards or key fobs).

12882 (U) Behind the scenes, each Credential is associated with a public and private key pair. The
12883 public key provides encryption (writing) capabilities. The private key provides decryption
12884 (reading) capabilities. When encrypting, each of these assigned Credentials (public key values) is
12885 combined with other values and random information to construct a key. This key is used with
12886 any number of cryptographic algorithms to encrypt the information and is then destroyed. The
12887 same key will never be used again to encrypt other information.

12888 (U) Once encrypted, the information is unreadable until it is decrypted using the same set of
12889 Credentials (private key values) and the same algorithm. Since CKM immediately destroys the
12890 key, it must later reconstruct it to decrypt the information. It does this by using a header that it
12891 attaches to the encrypted information, along with other cryptographic data retrieved from the
12892 user's Member Profile. In the header, CKM includes identifiers to the Credentials applied, but
12893 not the actual values. When decrypting, CKM attempts to retrieve the values needed to build the
12894 key from the receiver's set of Credentials. If the receiver holds the appropriate Credentials, CKM
12895 will be able to construct the key needed to decrypt the information. If not, the information will
12896 remain unreadable. This process is transparent and requires no instructions or intervention from
12897 the user.

12898 (U) CKM technology cryptographically binds different access elements together. These elements
12899 can uniquely represent users (identity components), application processes, information, media,
12900 business rules, and scope. When these various elements are uniquely combined and
12901 mathematically proven through cryptography, the goals of content-based, role-based access and
12902 distributed information security can be achieved.

12903 **2.7.3.3.1.7 (U) IKE and ISAKMP**

12904 (U) Internet Key Exchange (IKE) is the key management protocol used with IPsec—automating
12905 the process of negotiating keys, changing keys, and determining when to change keys. IKE
12906 implements a security protocol called Internet Security Association and Key Management
12907 Protocol (ISAKMP), which uses a two-Phase process for establishing an IPsec tunnel. During
12908 Phase 1, two gateways establish a secure, authenticated channel for communication. Phase 2
12909 involves an exchange of keys to determine how to encrypt data between the two entities.

12910 (U) Details on IKE can be found in IETF (RFC 2409).

12911 **2.7.3.3.1.8 (U) HSM (Hardware Security Module)**

12912 (U) An HSM is a physically secure, tamper-resistant security server that provides cryptographic
12913 functions to secure transactions in applications. Acting as a peripheral to a host computer, the
12914 HSM provides the cryptographic facilities needed to implement a wide range of data security
12915 tasks. HSMs perform cryptographic operations, protected by hardware. These operations may
12916 include:

- 12917 • (U) Random number generation
- 12918 • (U) Key generation (asymmetric and symmetric)
- 12919 • (U) Asymmetric private key storage while providing protection (security) from attack
12920 (i.e., no unencrypted private keys in software or memory)

- 12921 • (U) Private keys used for signing and decryption
- 12922 • (U) Private keys used in PKI for storing Root Keys
- 12923 • (U) Stored value card issuing and processing
- 12924 • (U) Chip card issuing and processing
- 12925 • (U) Message authentication
- 12926 • (U) PIN encryption and verification.

12927 (U) HSMs offer a higher level of security than software. They are normally evaluated by third
 12928 parties, such as “National Institute of Standards and Technology” (NIST), or through the Federal
 12929 Information Processing Standards Publication (FIPS PUB 140-2). This level of security is
 12930 required by some highly secured web applications, PKIs, and CAs.

12931 **2.7.3.3.2 (U) Usage Considerations**

12932 **2.7.3.3.2.1 (U) Implementation Issues**

12933 2.7.3.3.2.1.1 (U) Key Management Policy (KMP)

12934 (U) The KMP is a high-level statement of organizational key management policies that includes
 12935 authorization and protection objectives, and constraints that apply to the generation, distribution,
 12936 accounting, storage, use, and destruction of cryptographic keying material. The policy
 12937 document—or documents that comprise the KMP—will include high-level key management
 12938 structure and responsibilities, governing standards and guidelines, organizational dependencies
 12939 and other relationships, and security objectives. [Note that in a purely PKI environment, the
 12940 KMP is usually a stand-alone document known as a Certificate Policy (CP).] The scope of a
 12941 KMP may be limited to the operation of a single PKI CA and its supporting components or to a
 12942 symmetric point-to-point or single key center environment. Alternatively, the scope of a KMP
 12943 may be the operations of a hierarchical PKI, bridged PKI, or multiple center symmetric key
 12944 environments.

12945 (U) The KMP is used for a number of different purposes. The KMP is used to guide the
 12946 development of KMPSs for each PKI CA or symmetric key management group that operates
 12947 under its provisions. CAs from other organizations’ PKIs may review the KMP before cross-
 12948 certification, and managers of symmetric key KM infrastructures may review the KMP before
 12949 joining new or existing multiple center groups. Auditors and accreditors will use the KMP as the
 12950 basis for their reviews of PKI CA and symmetric key KMI operations. Application owners that
 12951 are considering a PKI certificate source should review a KMP/CP to determine whether its
 12952 certificates are appropriate for their applications.

12953 2.7.3.3.2.1.2 (U//FOUO) Key Packaging

12954 (U//FOUO) In the past, different key packaging structures and delivery protocols were developed
12955 for interaction between elements of different hierarchical tiers in the KMI. This proved to be an
12956 inflexible approach in that it constrained the spectrum of possible interactions by requiring
12957 specialized interface functionality for each communicating entity. The goal is to now provide a
12958 single packaging scheme that supports interactions between entities regardless of their placement
12959 in the key management hierarchy

12960 (U//FOUO) The existing secure key and data packaging techniques evaluated and analyzed are:

- 12961 • (U//FOUO) EKMS BET (Bulk Encrypted Transaction)
- 12962 • (U//FOUO) KMS Benign Techniques Transactions
- 12963 • (U) S/MIME format.

12964 (U//FOUO) The Key Packaging design must support implementation of the security mechanisms
12965 required by key transport/delivery.

12966 2.7.3.3.2.1.3 (U//FOUO) Key Delivery

12967 (U//FOUO) Key Delivery is a separate and distinct method from Key Packaging. The Key
12968 Delivery method addresses situations where keys (i.e., variables in the form they are to be used
12969 in a cryptographic algorithm) are moved from one entity to another. Entities may be ECUs,
12970 elements of the KMI, or Mission Support and Management Systems (MS&MSs).

12971 (U//FOUO) The purpose of the delivery method is to provide a common key transport standard
12972 regardless of whether the key is being distributed from a PSN to a PRSN, a DTD to an ECU, a
12973 PRSN to a workstation, etc. The method is used to provide an initial key or set of keys to an
12974 ECU or to replace keys already in use to sustain a security service. It is an application layer KM
12975 method that is independent of the underlying communications protocols and is totally self
12976 supportive with respect to protecting the key.

12977 (U//FOUO) The Key Delivery method requires certain security services. It is structured to
12978 incorporate the best of EKMS and industry standard security mechanisms. The sender and
12979 receiver in a key delivery interaction expect the following security properties to be maintained:

- 12980 • (U//FOUO) Confidentiality - The key value must not be released in transit between
12981 authorized entities, that is, the key value is only known to authorized entities.
- 12982 • (U//FOUO) Source Authentication - The key is received from an authorized and
12983 verifiable source.
- 12984 • (U//FOUO) Integrity - The key value is accurate, that is, the received and generated keys
12985 are identical.

12986 (U//FOUO) Key Distribution Over-the-Air Distribution (OTAD) encompasses two processes:

- 12987 • (U//FOUO) Over-the-Air Rekey (OTAR) – a cryptographic equipment takes unencrypted

12988 key from a fill device, encrypts that key, and sends it to a receiving cryptographic
12989 equipment for use in that equipment.

- 12990 • (U//FOUO) Over-the-Air Transfer (OTAT) – a cryptographic equipment takes
12991 unencrypted key from a fill device, encrypts that key, and sends it to a receiving
12992 cryptographic equipment that transfers that key to a fill device. The key is then loaded
12993 into a cryptographic equipment that is not on the same network.

12994 (U//FOUO) COMSEC equipment, such as the KYX-15, is an example of a Type-1 key
12995 distribution system. The KYX-15 is the Net Control Device for a key being used within the
12996 communications net. It enables the operator to generate a key and electronically send it to any
12997 member of the net. Since the KYX-15 is the Net Controller, it has a copy of all the keys being
12998 used in the net. Each KY-57 has a unique Key Encryption Key (KEK)¹² and at least one Traffic
12999 Encryption Key (TEK). To do an Over-the-Air Rekey (OTAR), the Net Controller generates and
13000 electronically sends a new TEK (TEK 2) encrypted in the individual user's unique KEK. All of
13001 this is managed by the KYX-15 over the net communications. This process is also known as in-
13002 band rekeying.

13003 (U//FOUO) A key that is transferred by OTAT is also sometimes available in an unencrypted
13004 form before and after distribution. Even electronic key that is distributed via EKMS is sometimes
13005 available in an unencrypted form when loading most cryptographic equipment.

13006 (U//FOUO) Benign Techniques are used for distributing and loading key material into
13007 cryptographic equipment that do not allow exposure of the material to any entity other than the
13008 equipment which will be consuming the material. EKMS uses benign keying techniques to
13009 support all of its own internal functions. Examples of benign technique being used today can be
13010 seen with Benign Fill FIREFLY Keys, which are used by End Cryptographic Units, Local
13011 Management Devices/Key Processor, and the Central Facility to implement benign fill. The fill
13012 is the actual process by which operational keys are to be generated, distributed, and loaded into
13013 compatible cryptographic end equipment—without human exposure. This includes the loading of
13014 all cryptographic key material into the end equipment.

13015 (U//FOUO) EKMS is currently being modified to support a broad range of benign keying
13016 techniques while interacting with and supporting new equipment. Over time, older equipment
13017 will be replaced with newer equipment using these techniques. This is being planned and
13018 coordinated under the NSA Crypto Modernization Plan.

¹² (U) A Key Encryption Key is a key that is used to encrypt or decrypt another key that is to be transmitted or stored.

13019 2.7.3.3.2.1.4 (U)

13020 (U) A Federal Information Processing Standard (FIPS) or NIST Recommendation will be
13021 developed to define the acceptable key establishment schemes. The standard or recommendation
13022 will select Diffie-Hellman (D-H) and MQV key agreement schemes from ANSI X9.42, RSA key
13023 agreement and key transport schemes from ANSI X9.44, and Elliptic Curve key agreement and
13024 key transport schemes from ANSI X9.63. All three ANSI documents are currently in a draft
13025 form, but are expected to be adopted by ANSI in the near future. NIST intends to select a subset
13026 of the schemes specified in the draft ANSI standards. The scheme definition document will also
13027 include a specification for a key wrapping technique, whereby a symmetric key is encrypted
13028 using another symmetric key (e.g., an AES key is encrypted by an AES key).

13029 **2.7.3.3.2.2 (U) Advantages**13030 **2.7.3.3.2.3 (U) Risks/Threats/Attacks**

13031 (U//FOUO) Risks in key management occur from the moment a cryptographic key is generated.
13032 Manual processes that handle distribution pose the single biggest threat. In fact, studies have
13033 shown that HUMINT (Human Intelligence) is the greatest threat factor. Key storage and facilities
13034 are also areas where keys can be compromised.

13035 **2.7.3.3.3 (U) Maturity**

13036 (U//FOUO) Some of the technologies in Key Management, such as generation, initial key load
13037 and rekeying are quite mature, and have been adopted under various classified (EKMS) and
13038 unclassified (PKI) infrastructures. However, the management and distribution of crypto-material
13039 still remains in some cases a very manually intensive process. Technologies such as high
13040 assurance OTNK and similar key distribution methods are only just emerging. Many of the
13041 issues that surround technological issues of high assurance with key management practices are
13042 being addressed by the KMI initiative.

13043 (U//FOUO) Another gap area is the lack of standards for unified key labeling, packaging, and
13044 distribution formats. The only area where some semblance of standards exist here are in the PKI
13045 (public, asymmetric keys). But none exists beyond PKI. Moreover, PKI has its own limitations
13046 with keys, such as re-keying, since PKI is certificate driven and not so much key-driven. In the
13047 Type-1 Classified arena, the key packaging and distribution processes are mainly manual
13048 processes. While they follow individual and situational-based policy, there are no standards to
13049 unify these to eliminate or reduce manual error-prone and human access vulnerabilities towards
13050 threats. Standards and technologies should include the incorporation of MLS systems and data
13051 stores to close these gaps.

13052 (U) Based on the above, the overall maturity of Key Management is assessed overall as
13053 Emerging (TRL 4-6). Prototypes and implementations for generation and sometimes-automated
13054 distribution exist, as seen in EKMS. But standardization at the enterprise-wide level, such as
13055 fulfilling GIG-wide requirements, is yet to be developed and adopted. These, as well as
13056 infrastructure issues are key concerns that the KMI effort needs to address.

13057

2.7.3.3.4 (U) Standards

13058

Table 2.7-4: (U) Key Management Standards

This table is (U//FOUO)	
Name	Description
IETF Standards	
S/MIME	Ramsdell, B., "S/MIME Version 3 Message Specification", RFC 2633, June 1999
MIME	Freed, N., Borenstein, N., "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996
CMS	Housley, R., "Cryptographic Message Syntax", RFC 3369, June 1999
ANSI Standards	
X9.69	Framework for Key Management Extensions. This standard defines specific key management methods for controlling and handling keys
X9.73	Cryptographic Message Syntax (CMS) The Constructive Key Management technique (CKM), described in ANS X9.69, is used to encrypt objects. It may be used with CMS to encrypt a message (as the object) to a set of users sharing a common set of values (known as key components)
X9.42	Key Agreement of Symmetric Keys using Discrete Logarithm Cryptography
X9.44	Key Establishment Using Factoring-Based Public Key Cryptography
NIST Standards	
FIPS PUB 140-2 ANNEX D	Security Requirements for Cryptographic Modules Annex D: Approved Key Establishment Techniques Annex D provides a list of the FIPS Approved key establishment techniques applicable to FIPS PUB 140-2.
XKMS	XML Key Management Specification (XKMS) http://csrc.nist.gov/cryptval/140-2.htm
FIPS 171	Symmetric Key Establishment Techniques National Institute of Standards and Technology, Key Management using ANSI X9.17, Federal Information Processing Standards Publication 171, April 27, 1992 http://csrc.nist.gov/publications/fips/fips171/fips171.txt
NSA Standards	
EKMS 208	EKMS Key Distribution Functional Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 215	EKMS Communications Requirements Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 301	EKMS Types Dictionary Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 302	EKMS Key Distribution Data Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 311	EKMS ACCORDION 1.3 Length Indicator and Binding Code Specification. National Security Agency, Director, National Security Agency, Ft. George G. Meade MD. 20755-6734

This table is (U//FOUO)	
Name	Description
EKMS 603	Interface Specification for the Data Transfer Device AN/CYZ-10. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734
W3C Standards	
XAdES	J.C. Cruellas, G. Karlinger, K. Sankar XML Advanced Electronic Signatures; W3C Note 20 February, 2003 http://www.w3.org/TR/XAdES/
XML	Bray, T., Paoli, J., Sperberg-McQueen, C. M., Maler, E., "Extensible Markup Language (XML) 1.0 (Second Edition)," W3C Recommendation 6 October, 2000
XMLENC	Eastlake, D., Reagle, J., Imamura, T., Dillaway, B., Simon, E., "XML Encryption Syntax and Processing," W3C Recommendation 10 December, 2002
XMLSIG	Eastlake, D., Reagle, J., Solo D., "(Extensible Markup Language) XML-Signature Syntax and Processing," RFC 3075, March, 2002
XMLSEC	Mactaggart, M., "Enabling XML Security: An introduction to XML encryption and XML signature," http://www-106.ibm.com/developerworks/xml/library/s-xmlsec.html/index.html
KMI-2200	July, 2004
This table is (U//FOUO)	

13059 **2.7.3.3.5 (U) Dependencies**

13060 (U//FOUO) The success of KM technologies depends on the successful specification and
 13061 completion of the new and improved emerging infrastructures, mainly KMI.

13062 **2.7.3.3.6 (U) Complementary Technologies**

13063 (U//FOUO) KMI attempts to encompass a number of complementary (but disparate)
 13064 technologies found in PKI, CMCS (COMSEC Material Control System), and EKMS.

13065 (U) CKM complements PKIs by adding the Authorization component and works with all the
 13066 leading PKI technologies.

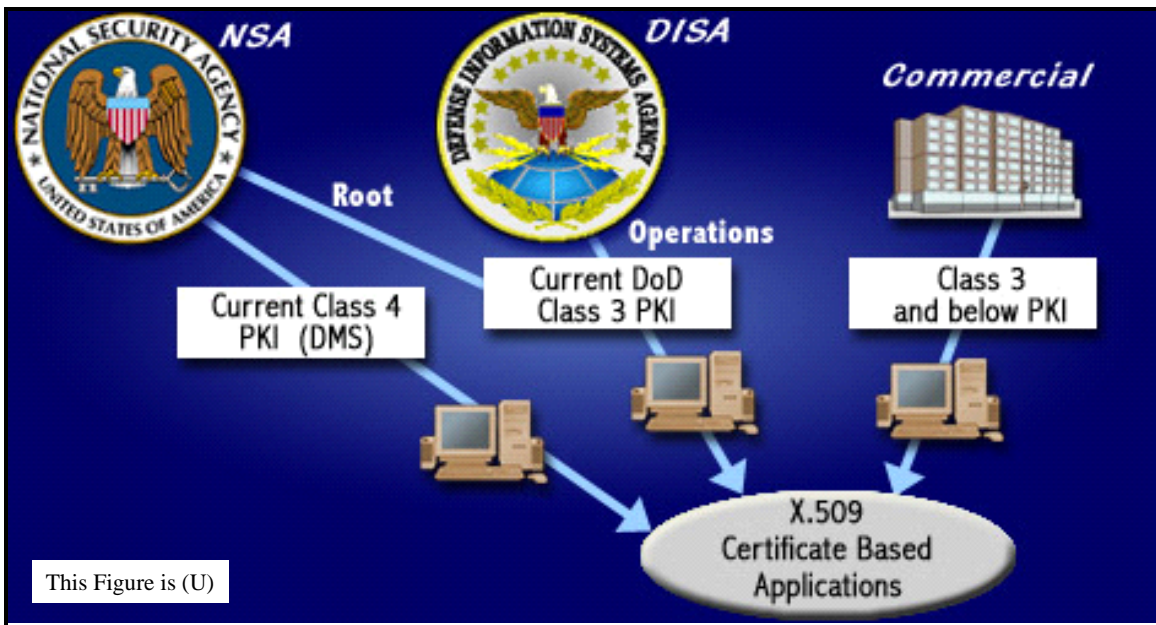
13067 **2.7.3.4 (U) Certificate Management**

13068 **2.7.3.4.1 (U) Technical Detail**

13069 **2.7.3.4.1.1 (U) Certificate-Managed Infrastructures**

13070 (U) Certificate Management ties closely with Key Management. Certificates are widely used
 13071 today within the PKI, based on the ANSI X.509v3 standards. Certificates use asymmetric keys,
 13072 which are public/private key pairs.

13073 (U) There are three independent PKI infrastructures that manage certificates today. These
 13074 certificate management infrastructures are also addressed in the KMI Architecture and vision, as
 13075 detailed in the section on Key Management. Figure 2.7-8 shows the three independent certificate
 13076 management infrastructures that exist today in the government and commercial arenas.



13077
 13078 **Figure 2.7-8: (U) DoD and Commercial Certificate-Managed Infrastructures**

13079 **2.7.3.4.1.2 (U) Certificate Assurance levels**

13080 (U) The DoD specifies assurance levels for certificates, and technologies have been built to these
 13081 Certificate level specifications. Overall, there are three Certificate Assurance levels in the DoD
 13082 PKI. These are Class 5, Class 4 and Class 3 certificates. Class 5 is still under development, but
 13083 technologies based on Classes 4 and 3 are operational today in various forms, and a brief
 13084 description of these classes of certificates follow.

13085 (U) Class 4 (Operational — based on NSA Technology)
 13086 (U//FOUO) The Class 4 PKI serves to protect Sensitive But Unclassified (SBU) information for
 13087 the Defense Message System (DMS). It uses the FORTEZZA card as the user token for the
 13088 storage of the Private key. It is designed to manage SBU and Secret information. It contains an
 13089 individual's private key and the cryptographic algorithms for encryption and digital signature.
 13090 FORTEZZA Plus card is primarily used with the STE. It is designed to protect information up to
 13091 and including Top Secret.

13092 (U) The Certificate Authority Workstation generates and manages certificates within the
 13093 Government Class 4 PKI.

13094 (U) Class 3 (Operational - Based on Commercial technology)
 13095 DoD Class 3 PKI: This PKI serves to protect mission critical information, and provides mission
 13096 and administrative support. NSA serves as the Root CA and the Defense Information Systems
 13097 Agency (DISA) manages operations. Private keys are stored on software tokens, such as floppy
 13098 disks.

13099 (U) The last PKI used by the Government is Commercial based. It is not controlled or operated
 13100 by the Government. Certificates and keys are entirely generated and managed within the private
 13101 sector. Private keys are generally on a software token. A private company serves as the Root CA.
 13102 The PKI structure enables the Government to participate in E-Commerce activities.

2.7.3.4.1.3 (U) PKI Technology Model

13104 (U) The PKI Technology Model illustrated in Figure 2.7-9 divides the PKI landscape into five
 13105 layers and expresses requirements in terms of this model.

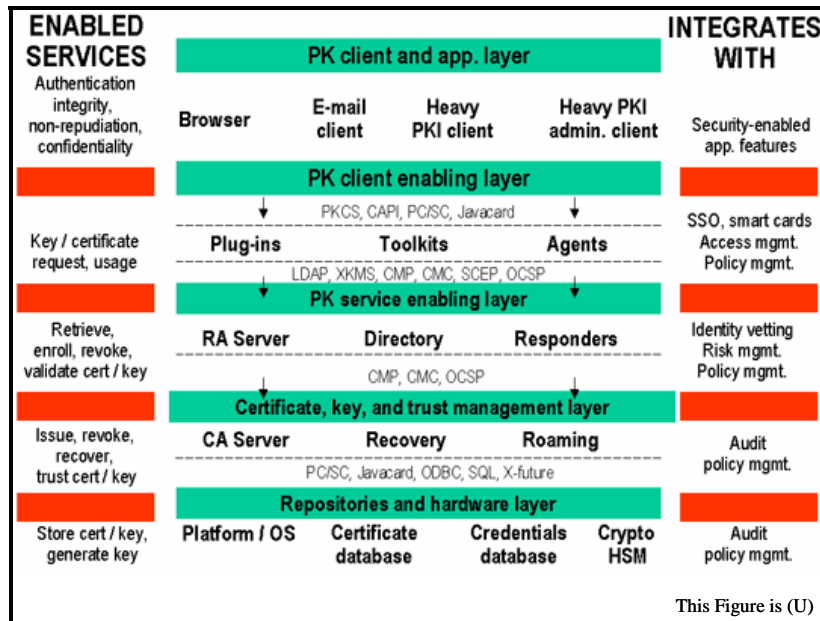


Figure 2.7-9: (U) PKI Technology Model

13108 2.7.3.4.1.3.1 (U) Client and Application Layer

13109 (U) Native PK-enabled browser, micro-browser, e-mail, and VPN clients in most cases are able
13110 to work with RAs and other enrollment gateways from the PKI products to obtain, check, and
13111 use certificates. There is the need for plug-ins to enhance security functionality in commodity
13112 browsers and e-mail programs. In some cases, there is a need for full desktop security clients, to
13113 enable functions such as file encryption, smart card support, secure e-mail with key history, dual
13114 encryption and signature key pair support, and CRL/OCSP checking usually accomplished by
13115 replacing an entire browser-based trust model with a more scalable and policy-managed PKI
13116 client implementation. The need for heavy PKI clients is declining, however, with improved
13117 CRL checking, dual key, smart card support and other features in the latest Internet Explorer (IE)
13118 and Communicator browsers and higher, as well as native file encryption in Windows XP. In
13119 fact, recent updates to IE have activated certificate validation, causing VeriSign to implement
13120 additional servers to handle the extra requests.

13121 (U) Also, the client is the focus for administration of PKI systems. Some vendors implement a
13122 Windows or UNIX-based management client to administer their core RA and CA infrastructures,
13123 while others provide browser-based management, and some do both.

13124 2.7.3.4.1.3.2 (U) Client Enabling Layer

13125 (U) Whenever an application does not natively support PKI, vendors might provide plug-ins,
13126 toolkits, applets, or agents to help. These client enablers should preferably operate in conjunction
13127 with platform-resident security APIs such as Microsoft's Crypto API or in a stand-alone manner.
13128 Java applets or servlets let vendors immaculately transplant PK functionality into an
13129 application—no code installation required. Plug-ins are most effective in enhancing PK functions
13130 of mass market browsers and other clients with existing security hooks, but do leave a code
13131 footprint behind. Toolkits and APIs are required for custom integration of more obscure
13132 applications. Agents can enable certificate-equipped clients to sign onto other security domains
13133 through SSO or portal systems.

13134 2.7.3.4.1.3.3 (U) Service Enabling Layer

13135 (U) Enabling services include RAs, directories, and PK responders that a field client requests for
13136 enrollment, retrieval, recovery, validation, roaming, and other services. Ideally, they operate
13137 through standard protocols such as CMP, OCSP, SCEP, and XKMS. Directories are essential, as
13138 they allow clients to retrieve certificates, check policies, and check CRLs using LDAP. RAs take
13139 enrollment, recovery and revocation requests, vet them, and pass them on to the certificate
13140 infrastructure. Sometimes the RA function should be interactive or in other cases automated,
13141 particularly for enrollment. RAs may receive many batched user enrollment requests, and issue
13142 or deny certificates to those users based on rules in an automated identity vetting system.

13143 (U) RAs may serve as gateways implementing protocols, such as SCEP, for automated pickup of
13144 certificates by machines or application services. For example, Windows XP and Server 2003
13145 enable auto enrollment of machines and users.

13146 (U) There is a push for the service-enabling layer to help thin out the PKI client in order to
13147 reduce the burden on application developers. PK responders implementing OCSP V2 and XKMS
13148 must supplement and replace today's simple OCSP V1 responders. Validation must become
13149 more sophisticated, linking with policy management and automated risk management systems
13150 provided by credit bureaus and other businesses.

13151 2.7.3.4.1.3.4 (U) Certificate, Key, and Trust Management Layer

13152 (U) Certificate servers (or CAs) must obtain their root credentials, enter into trust relationships
13153 by signing cross certificates or certificates of subordinate CAs, and issue and revoke client
13154 certificates. Recovery servers allow clients to obtain backup copies of private keys and
13155 certificates. Roaming servers provide securely stored credentials on demand to properly
13156 authenticated users. Note that PKI vendors implement and package certificate, recovery, and
13157 roaming servers in different ways.

13158 2.7.3.4.1.3.5 (U) Repositories and Hardware Layer

13159 (U) No PKI system would be whole without rock-solid underlying computer platforms,
13160 databases, and hardware security modules (HSMs) provided by best-in-class vendors. Customers
13161 must locate CAs, recovery, and roaming servers on UNIX, Windows 2000, Windows Server
13162 2003, or other OS/hardware platforms as securely as possible. The PKI servers must also
13163 leverage robust databases with strong performance, backup, and audit features. Finally, it should
13164 be possible to store CA root keys and archived private keys in HSMs. In fact, CAs may depend
13165 on HSMs to achieve the FIPS 140-1 or ITSEC compliance levels needed for government
13166 certification or certification from private organizations like Identrus. HSMs can also accelerate
13167 signing, signature checking, and encryption processing performance. HSMs, CAs, and
13168 applications must implement common asymmetric, symmetric, and message digest crypto
13169 algorithms.

13170 2.7.3.4.1.3.6 (U) Wireless Considerations

13171 (U) Wireless PKI functionality is similar to wired PKI functionally, though requiring support for
13172 different product components such as wireless software toolkits, PKI Portal RAs, and CAs
13173 capable of issuing WTLS certificates. PKI systems must support short-lived certificates where
13174 micro-browsers cannot validate certificates online or store root keys. PKI vendors offering
13175 outsourced services need to get their root certificates implanted in devices just as they have done
13176 in browsers.

13177 **2.7.3.4.2 (U) Usage Considerations**

13178 **2.7.3.4.2.1 (U) Implementation Issues**

13179 (U) Interoperability of components among multi-vendors is a critically important issue for
13180 infrastructures that support key and certificate management. Interoperability is used to describe
13181 the ability for one application to communicate seamlessly with another. Other aspects of
13182 interoperability include the ability to mix and match various PKI components from various
13183 vendors. Interoperability can also refer to the interaction between one enterprise domain and
13184 another (e.g., in order to conduct secure business-to-business transactions). Interoperability
13185 would allow greater flexibility and freedom of choice between vendor solutions and lowers the
13186 risk of deploying a PKI-based solution.

13187 (U) The lack of interoperability is perceived as the leading barrier to wide-scale deployment of
13188 PKIs. Indeed, one of the fundamental reasons for the formation of the PKI Forum in December,
13189 1999, was to identify and resolve existing barriers to multi-vendor interoperability.

13190 (U) The PKI Forum (<http://www.pkiforum.org/pdfs/PKIInteroperabilityFramework.pdf>) has
13191 identified three major interoperability areas that require enhancements:

- 13192 • (U) Component-Level Interoperability
- 13193 • (U) Application-Level Interoperability
- 13194 • (U) Inter-Domain Interoperability

13195 **2.7.3.4.2.2 (U) Advantages**

13196 (U) The PKI infrastructure and public certificates have been around for many years. One of its
13197 advantages is longevity. There have been many improvements along the way, but there are still
13198 challenges ahead.

13199 **2.7.3.4.2.3 (U) Risks/Threats/Attacks**

13200 (U) There are two primary entities vulnerable to attack are the subscriber and the CA. When
13201 there is a subscriber compromise, all subscribers within the entire infrastructure can be exploited
13202 until the compromise is detected. If there are no subordinate CAs, and the Root CA was
13203 compromised, the entire PKI could be compromised with devastating results. New keys and
13204 certificates would be required through the entire infrastructure. If a subordinate CA is
13205 compromised, only that CA and its subscribers must initiate actions to recover. This would still
13206 be an enormous amount of work, which is the reason extreme measures are required to protect
13207 the CAs.

13208 (U) The CA must therefore maintain the integrity of its operations. The policies and procedures
13209 for its operation must be strictly adhered to at all times. Compromises of individual subscribers
13210 must be quickly and efficiently remedied and new keys generated, as appropriate. Concurrently,
13211 the Compromised Key List would need to be updated. Should the CA itself be compromised, all
13212 CA subscribers would need to be rekeyed and new Certificates created.

13213 **2.7.3.4.3 (U) Maturity**

13214 (U) The Infrastructure of public certificates, i.e., the PKI, has been around for many years, and as
13215 such has undergone significant growth and maturity. The maturity of this technology is rated as
13216 Emerging (TRLs 4 - 6). However due to the lack of interoperability standards for technologies
13217 within the infrastructure and the lack of security policy mandates, there is still reluctance for
13218 enterprises with need for high assurance to adopt the PKI standard. The maturity of Certificate
13219 Management is also rated as Emerging (TRLs 4 - 6).

13220 **2.7.3.4.4 (U) Standards**

13221 (U) Table 2.7-5 highlights some of the components and the standards with which PKI products
13222 comply.

Table 2.7-5 (U) Key Management and Certificate Management Standards

This Table is (U)	
Name	Description
Symmetric Encryption Algorithms	
DES	U.S. Data Encryption Standard (DES) in accordance with U.S. FIPS PUB 46-2 and ANSI X3.92
AES	U.S. Advanced Encryption Standard (AES) in accordance with U.S. FIPS PUB 197 (256-bit keys supported)
CAST block cipher	CAST block cipher in accordance with RFC 2144 (64-bit, 80-bit, and 128-bit variations are supported)
Triple-DES	Triple-DES in accordance with ANSI X9.52 (3-key variant for an effective key size of 168-bits is supported)
RC2®	RC2® in accordance with RFC 2268 (40-bit and 128-bit variations are supported)
IDEA	IDEA as listed in the ISO/IEC 9979 Register of Cryptographic Algorithms (128-bit supported)
Note: DES, CAST, Triple-DES, RC2 and IDEA encryption all use CBC mode of operation in accordance with U.S. FIPS PUB 81, ANSI X3.106 and ISO/IEC 10116	
Digital Signature Algorithms	
RSA	RSA in accordance with Public Key Cryptographic Standards (PKCS) specification PKCS#1 Version 2.0, ANSI X9.31, IEEE 1363, ISO/IEC 14888-3 and U.S. FIPS PUB 186-2 (1024-bit, 2048-bit, 4096-bit and 6144-bit supported)
DSA	DSA in accordance with the Digital Signature Standard, U.S. FIPS PUB 186-2, ANSI X9.30 Part 1, IEEE P1363 and ISO/IEC 14888-3 (1024-bit supported)
ECDSA	ECDSA in accordance with ANSI X9.62, IEEE P1363, ISO/IEC 14888-3 and U.S. FIPS PUB 186-2 (192-bit default)
One-Way Hash Functions	
SHA-1, SHA-256, SHA-384 and SHA-512	SHA-1, SHA-256, SHA-384 and SHA-512 in accordance to U.S. FIPS PUB 180-2 and ANSI X9.30 Part 2
MD5 Message-Digest algorithm	MD5 Message-Digest algorithm in accordance with RFC 1321
MD2 Message-Digest algorithm	MD2 Message-Digest algorithm in accordance with RFC 1319
RIPEDM-160	RIPEDM-160 in accordance with ISO/IEC 10118-3:1998
Key Exchange Algorithms	
RSA key transfer	RSA key transfer in accordance with RFC 1421 and RFC 1423 (PEM), PKCS#1 Version 2.0, IEEE P1363

This Table is (U)	
Name	Description
Diffie-Hellman key agreement	Diffie-Hellman key agreement in accordance with PKCS#3
Simple Public-Key GSS-API Mechanism (SPKM) authentication and key	Simple Public-Key GSS-API Mechanism (SPKM) authentication and key agreement in accordance with RFC 2025, ISO/IEC 9798-3 and U.S. FIPS PUB 196
SSL v3 and TLS v1	SSL v3 and TLS v1 in accordance with RFC 2246
Symmetric Integrity Techniques	
MAC	MAC in accordance with U.S. FIPS PUB 113 (for DES-MAC) and X9.19
HMAC	HMAC in accordance with RFC 2104
Pseudo-Random Number Generator	
Pseudo random number generator	Pseudo random number generator in accordance with ANSI X9.17 (Appendix C) and FIPS 186-2
Certificates and Certificate Revocation Lists (CRLs)	
Version 3 public-key certificates and Version 2 CRLs	Version 3 public-key certificates and Version 2 CRLs in accordance with ITU-T X.509 Recommendation and ISO/IEC 9594-8 (4th edition, 2000 as well as earlier editions)
Version 3 public-key certificate and Version 2 CRL extensions	Version 3 public-key certificate and Version 2 CRL extensions in accordance with RFC 2459 and RFC 3280
Version 3 public-key certificate and Version 2 CRL extensions in accordance with U.S. FPKI X.509 Certificate and CRL Extensions Profile	Version 3 public-key certificate and Version 2 CRL extensions in accordance with U.S. FPKI X.509 Certificate and CRL Extensions Profile
Version 3 public-key certificate and Version 2 CRL extensions in accordance with NIST X.509 Certificate and CRL Extensions Profile for the Common Policy	Version 3 public-key certificate and Version 2 CRL extensions in accordance with NIST X.509 Certificate and CRL Extensions Profile for the Common Policy
Version 3 "Qualified" certificates in accordance with RFC 3039 and ETSI TS 101 862	Version 3 "Qualified" certificates in accordance with RFC 3039 and ETSI TS 101 862
Version 3 public-key certificates and Version 2 CRLs in accordance with de-facto standards for Web browsers and servers	Version 3 public-key certificates and Version 2 CRLs in accordance with de-facto standards for Web browsers and servers
WTLS Certificate support in accordance with WAP WTLS Version 1.1. (certificate issuance)	WTLS Certificate support in accordance with WAP WTLS Version 1.1. (certificate issuance)
RSA algorithm identifiers and public key formats in accordance with RFC 1422 and 1423 (PEM) and PKCS#1	RSA algorithm identifiers and public key formats in accordance with RFC 1422 and 1423 (PEM) and PKCS#1

This Table is (U)	
Name	Description
Online Certificate Status Protocol, version 2. Working document of the Internet Engineering Task Force (IETF) RFC 2560.	Online Certificate Status Protocol, version 2. Working document of the Internet Engineering Task Force (IETF) RFC 2560.
File Envelope Formats	
Standard file envelope format based on Internet RFC 1421 (PEM)	Standard file envelope format based on Internet RFC 1421 (PEM)
PKCS#7 Version 1.5 based on RFC 2315 and Cryptographic Message Syntax (CMS) based on RFC 3369 and 3370	PKCS#7 Version 1.5 based on RFC 2315 and Cryptographic Message Syntax (CMS) based on RFC 3369 and 3370
S/MIME Version 2 based on RFC 2311	S/MIME Version 2 based on RFC 2311
Secure Session Formats	
On-line GSS-API public key implementation mechanism using SPKM in accordance with Internet RFC 2025 and SPKM entity authentication in accordance with FIPS 196	On-line GSS-API public key implementation mechanism using SPKM in accordance with Internet RFC 2025 and SPKM entity authentication in accordance with FIPS 196
SSL v3 and TLS v1 in accordance with RFC 2246	SSL v3 and TLS v1 in accordance with RFC 2246
Repositories	
LDAP Version 2	LDAP Version 2 in accordance with RFC 1777 and RFC 2559
LDAP Version 3	LDAP Version 3 in accordance with RFC 2251-2256
Private Key Storage	
Private key storage	Private key storage in accordance with PKCS#5 and PKCS#8
Certificate Management	
Secure Exchange Protocol (SEP)	Secure Exchange Protocol (SEP), built using Generic Upper Layers Security (GULS) standards ITU-T Recs. X.830, X.831, X.832 and ISO/IEC 11586-1, 11586-2, 11586-3 (SEP continues to be supported for backward compatibility only)
PKIX-CMP	PKIX-CMP in accordance with RFC 2510 and PKIX-CRMF in accordance with RFC 2511
PKCS 7/10	PKCS 7/10 (for Web based clients and VPN solutions)
Cisco Certificate Enrollment Protocol (CEP)	Cisco Certificate Enrollment Protocol (CEP) (for VPN solutions)

13224

This Table is (U)	
Name	Description
Application Programming Interfaces (APIs)	
Hardware cryptographic interface	Hardware cryptographic interface in accordance with PKCS#11
Generic Security Services API (GSS-API)	Generic Security Services API (GSS-API) in accordance with RFC 1508 and 1509
IDUP-GSS-API	IDUP-GSS-API in accordance with Internet Draft draft-ietf-cat-idup-gss-08.txt
This Table is (U)	

13225 **2.7.3.4.5 (U) Dependencies**

13226 (U) There needs to be component interoperability, application interoperability, and inter-
 13227 organization interoperability. While PKI may never just disappear into the infrastructure, it
 13228 should be reduced to a set of simpler, better-understood services and decisions. Furthermore,
 13229 there is a need for integration with the OS, wireless, and smart card platforms as well as
 13230 platform-neutral Java and XML functionality. Emerging Web Services, SAML, and other XML
 13231 security specifications will benefit if PKI can be easily integrated. Both infrastructure and
 13232 application vendors should give high priority to XKMS development (once the standard
 13233 stabilizes and is ratified by W3C) to increase interoperability by thinning out the client layer of
 13234 PKI and support WS-Sec, SAML, Liberty Alliance, XML Access Control Markup Language
 13235 (XACML), Extensible Rights Markup Language (XrML), and other XML security
 13236 specifications.

13237 (U) Infrastructure vendors are preparing for a gradual evolution from PKIX toward XML-based
 13238 PKI, shedding the ASN.1 heritage of OSI in favor of a universal text encoding. But this
 13239 presumes that PKIX will eventually re-map X.509v3 certificates to XML encoding, and until
 13240 then, there will be an ongoing need to preserve PKIX interoperability by implementing XKMS,
 13241 CMP, CMC, and OCSP V2 (once it stabilizes) to achieve the broadest functionality and
 13242 component interoperability.

13243 (U) XML security standards are still quite immature and will require several more years before a
 13244 broad suite is available for deployment in commercial products. But architects and planners can
 13245 target a model architecture to leverage WS-Security. And federated identity will be ready to
 13246 move as vendor software is available. In the meantime, vendors are developing WS-Sec and
 13247 federated identity best practices that easily integrate PKI.

13248 (U) Liberty Alliance circles of trust may provide a driver for enterprises to cross certify. Vendors
 13249 must continue to engage these organizations. But no one consortium or trust network will unlock
 13250 the real potential of PKI unless it helps users meet the need for mutual certificate acceptance by
 13251 cross-certifying with others.

13252 (U) Cross-certifying requires simpler, more compatible policies, for it is the policy that cleaves
13253 CAs apart by limiting which certificates users and organizations can trust. Industry consortiums
13254 and vendors alike should invest significant effort in projects such as the Federal Public Key
13255 Infrastructure (FPKI) Group's Bridge CA program, which pushes the boundaries of PKI with its
13256 effort to extend inter-organization interoperability by refining path processing, policy mapping,
13257 cross certification, and directory services between agencies and commercial organizations. Sites
13258 must be able to leverage pre-existing certificates.

13259 **2.7.3.4.6 (U) Alternatives**

13260 (U) There are no real alternatives to Certificate Management technologies. As indicated earlier,
13261 there is a drive to establish interoperability standards, such that components from various
13262 certificate management providers can interoperate.

13263 **2.7.3.4.7 (U) Complementary Technologies**

13264 (U) CKM and Key Management technologies—especially asymmetric key methodologies—
13265 complement the incorporation of Certificate Management infrastructures and technologies.

13266 (U) XKMS defines a SOAP/XML-messaging-based alternative to traditional PKI, though in
13267 many ways XKMS is designed to complement, rather than replace, established PKI standards. At
13268 the client level, XKMS defines mechanisms under which applications delegate the retrieval,
13269 parsing, and validation of X.509 digital certificates to trusted servers, thereby streamlining the
13270 configuration of client-side trust-service business logic. XKMS requires retrofitting today's
13271 clients and applications to support, at a minimum, such standards as SOAP, XML-DSig, XML
13272 Schemas, XML Namespaces, WSDL, and XML Encryption.

13273 **2.7.3.5 (U) Configuration Management of IA Devices and Software**13274 **2.7.3.5.1 (U) Technical Detail**

13275 (U) The purpose of configuration management is to establish and maintain the integrity of IA
 13276 components—hardware, firmware, and software throughout their life cycle. Configuration
 13277 management involves identifying the configuration, controlling configuration changes, and
 13278 maintaining the integrity and traceability of the configuration throughout the component's life
 13279 cycle. Given the assured, dynamic, decentralized nature of the GIG, configuration establishment
 13280 and control must be assured—only authorized authorities should be able to modify
 13281 configurations. It must be remotely accessible, since GIG assets may be literally anywhere and
 13282 configuration updates must be possible in the field without local manual intervention. It must
 13283 also be auditable—there must be a mechanism for verifying a configuration is still valid.

13284 (U) Configuration Items that must be securely managed within the GIG include such items as:

- 13285 • (U) Operating System Software, particularly for trusted or high-assurance components
- 13286 • (U) Router tables
- 13287 • (U) Firewall configurations
- 13288 • (U) VPN configurations
- 13289 • (U) NDS configurations
- 13290 • (U) Host-based IDS/IPS agent configurations
- 13291 • (U) Malware detection and prevention agents, software and signature configuration files
- 13292 • (U) CDS configurations
- 13293 • (U//FOUO) Cryptographic modules and algorithms (hardware and software)
- 13294 • (U) Keys and Certificates
- 13295 • (U) Trusted applications.

13296 (U) Some of these may be represented in hardware, firmware, or software. Many will require
 13297 constant, regular updates to accommodate dynamic changes in the GIG and to fix discovered
 13298 vulnerabilities or defects. Some, such as keys and certificates, require that strict accountability be
 13299 maintained for their possession and distribution. Such items require packaging for distribution,
 13300 receipts, and auditable tracking of any transactions. Management operations that must be
 13301 performed include:

- 13302 • (U) Maintaining the set of authorized configuration baselines
- 13303 • (U) Installing a software configuration baseline
- 13304 • (U) Provisioning a system—installing optional or additional software components
 13305 according to the mission requirements for the target system.

- 13306 • (U) Verifying the completeness and integrity of a software configuration in IA
13307 components against a baseline
- 13308 • (U) Determining if upgrades or patches are necessary for an IA component
- 13309 • (U) Upgrading software or installing patches
- 13310 • (U) Installing and Upgrading third-party software applications
- 13311 • (U//FOUO) Transferring, receipting and installing data packages
- 13312 • (U) Reporting on the version and status of any IA component firmware or software
13313 including OS, system software, application software, and versioned data
- 13314 (U) Such tasks as determining if upgrades or patches are necessary overlap with tasks such as
13315 vulnerability assessment, discussed in Section 2.6, Network Defense and Situational Awareness.
- 13316 (U) A number of CM problems within the GIG already have point solutions, which are discussed
13317 below.
- 13318 **2.7.3.5.1.1 (U) Systems Management Applications**
- 13319 (U) Systems Management Consoles are centralized, dedicated systems that can manage other
13320 systems within an enterprise. They interact with the managed systems or clients through an
13321 installed agent. They can perform a variety of configuration management tasks using a
13322 proprietary communications protocol, which is highly extensible to allow development of
13323 additional operations on the clients. Actions that such servers can perform are:
 - 13324 • (U) Installation of the operating system remotely on a bare metal system for supported
13325 clients
 - 13326 • (U) Installation of data and applications or provisioning of client systems
 - 13327 • (U) Distribution and installation of software updates or patches and tracking of which
13328 machines did and did not receive updates
 - 13329 • (U) Forced remote execution of software on clients to perform such actions as malware
13330 detection updates
 - 13331 • (U) Verification and auditing of client system software configuration and versions
 - 13332 • (U) Asset tracking.

13333 (U) System Management applications interact with an agent residing on the client machine to
13334 perform their operations. Additional applications can be added via scripts and the API, but this
13335 can be a complex programming task with attendant development, testing, and deployment issues.
13336 These applications generally support common desktop and server operating systems with some
13337 supporting models of PDA. APIs and custom software development can extend high-end
13338 management frameworks to handle operations beyond that originally envisioned—or client
13339 targets. In cases where there is a large market, such as popular routers, third parties such as the
13340 router vendors have written plug-ins or interfaces to their proprietary management applications
13341 that connect to the large management applications.

13342 **2.7.3.5.1.2 (U) Network Boot Applications**

13343 (U) A wide variety of desktop and server computer systems are capable of booting an un-
13344 configured machine from a network server that is discovered at boot time. For Intel processor-
13345 based computers, the Intel Preboot eXecution Environment (PXE) [INTEL] specification defines
13346 an interface for booting from the network. Most RISC-based processors also have network boot
13347 capability by default. They depend upon such standard protocols as the Dynamic Host Control
13348 Protocol (DHCP) [DHCP97], Trivial File Transfer Protocol (TFTP), and the Boot Protocol
13349 (BOOTP). They can be used to dynamically boot a diskless client off a central server or as an
13350 initialization step that then loads a bootstrap kernel to load a complete system onto a local disk
13351 for subsequent use. Servers or systems management consoles can be configured to supply
13352 standardized OS images to booting PCs.

13353 (U) However, the underlying protocols are unauthenticated and depend upon network broadcast
13354 and are suitable only for a trusted, benign LAN environment. Since any server can respond, and
13355 the clients cannot authenticate to a server, the security vulnerabilities have proven so great that
13356 this mechanism is only used in special cases. The Intel PXE specification includes a Boot
13357 Integrity Services (BIS) API, but this is not widely available, and for high-assurance
13358 requirements requires making modifications to the Boot ROM of a system.

13359 **2.7.3.5.1.3 (U) Malware Management**

13360 (U) Virus detection is one of the more mature areas of IA. Viruses were one of the earliest
13361 attacks on computer systems, emerging shortly after the initial widespread adoption of personal
13362 computers. Because most virus detection software was signature-based, update mechanisms were
13363 developed early and have evolved with communication technologies. Current malware detection
13364 agents can automatically update themselves securely from central servers—both signatures and
13365 the application software itself. A number of virus vendors have enterprise management servers,
13366 which will manage the client malware detection agents in a local enterprise. These managers can
13367 generally perform the following:

- 13368 • (U) Signature (data) file or application update download (pull) from the vendor per policy
- 13369 • (U) Signature and application update to clients (push) per policy
- 13370 • (U) Configuration of scan and update policy
- 13371 • (U) Tracking of client update status (last contact, last version)

13372 • (U) Tracking of enrolled clients (machines with and without malware detection agents)

13373 • (U) Reporting statistics and consolidation of alerts.

13374 (U) With current products, only the malware detection agent vendor can provide the associated
 13375 management solutions. The format and structure of signature files and updates are proprietary, as
 13376 are the protocols used to perform the updates. As a consequence, no malware management
 13377 system can manage third party agents, and if general enterprise security console applications are
 13378 able to monitor the agents on a network, they cannot perform the configuration updates on those
 13379 agents.

13380 **2.7.3.5.1.4 (U) ECU Update**

13381 (U) Recent models of cryptographic hardware such as the KG 235 and KG-240 can be securely
 13382 managed by a manager device over the network. The manager is capable of performing the
 13383 following tasks remotely on a KG-240:

13384 • (U) Updating the system software

13385 • (U) Updating cryptographic algorithms

13386 • (U) Updating keys

13387 • (U) Updating security policies.

13388 (U) The protocol for managing the devices is proprietary and unique to the KG-240. Devices
 13389 such as the KG-235 do provide SNMP interfaces on both the red and black sides, but they are
 13390 limited to standard SNMP operations and do not provide configuration management capabilities.

13391 **2.7.3.5.1.5 (U) Patch Management Systems**

13392 (U) Patch Management Systems are software applications that are specifically designed to
 13393 centralize the distribution of operating system and specific application patches within an
 13394 enterprise. Some are agent-based, with small agent servers installed on monitored clients. Others
 13395 do not require an agent on the client targets. They use only the built-in capabilities of the resident
 13396 OS to provide the hook into the target system. Although a number of patch management
 13397 solutions operate on multiple architectures and operating systems, all investigated products
 13398 currently target only desktop and server systems and smaller devices that run Microsoft
 13399 Windows CE. None handle embedded systems or arbitrary client architectures.

13400 **2.7.3.5.2 (U) Usage Considerations**

13401 **2.7.3.5.2.1 (U) Implementation Issues**

13402 (U) Systems Management Applications are very large and complex systems. They require a
 13403 large, full time staff to use and maintain. Although very flexible and extensible, it comes at the
 13404 cost of software development with its associated development, testing, and deployment issues.

13405 (U) Agentless Patch Management systems suffer from significant network traffic from server to
 13406 target machines. In contrast, Agent-based patch management applications can use the on-device
 13407 agent to locally scan the machine for individual file version and configuration information.

13408 **2.7.3.5.2.2 (U) Advantages**

13409 (U) All these tools centralize one or more aspects of configuration management. For large
13410 systems management applications, they can centrally control many common aspects of
13411 configuration management.

13412 **2.7.3.5.2.3 (U) Risks/Threats/Attacks**

13413 No applications provide mechanisms to validate version numbers or system configurations using
13414 techniques as MD5 hashes to verify that critical files are unchanged. Although mechanisms exist
13415 to authenticate management servers, clients are not authenticated and the transactions are not
13416 generally protected, so they are unsuitable for high assurance applications.

13417 (U) Agent-based CM applications

13418 (U) For all configuration management systems that do not use secure communications, the threat
13419 is that an adversary could spoof the management console and take control or install arbitrary
13420 software on a client. If the client is not authenticated, then an adversary can spoof the client and
13421 receive keys or other cryptographic material and possibly assume the identity of the spoofed
13422 client. The issue of a spoofed client identity is discussed further in Section 2.7.2.1, Identity
13423 Management.

13424 (U) Agentless CM applications

13425 (U) For configuration management tools such as patch managers that are agentless, they use
13426 alternate means of accessing information, such as Microsoft NetBIOS file sharing and
13427 administrator login. Typically these services cannot be available on a machine except in the most
13428 benign environments due to extreme vulnerability, so such applications cannot be used at all
13429 outside the local enclave.

13430 **2.7.3.5.3 (U) Maturity**

13431 (U) The maturity is high for individual point solutions for various parts of configuration
13432 management. All of the various technologies have examples of successfully deployed product
13433 solutions in commercial environments. So, the maturity of CM technology is rated as Mature
13434 (TRLs 7 - 9). However, none of the technologies meets GIG requirements such as the high
13435 assurance required to securely manage Information Assurance Components (IAC) across a
13436 lower-assurance network.

13437 **2.7.3.5.4 (U) Standards**

13438 (U) Standards related to Configuration Management are included in Table 2.7-6.

Table 2.7-6: (U) Configuration Management Standards

This Table is (U)	
(U) Name	(U) Description
IETF Standards	
SNMPv3	The Simple Network Management Protocol, version 3 is the latest version of the IETF standard for managing network devices. Version 3 includes authentication and authorization, so is considered much more secure than previous versions. SNMP is widely implemented, but has some significant restrictions because of its very simple structure.
TFTP	The Trivial File Transfer Protocol (TFTP), as defined by IETF RFC 1350, is a very simple file transfer protocol that can be implemented in very small systems, such as firmware. It implements no authentication whatsoever and consequently is usable only in the most benign, protected environments.
DHCP	The Dynamic Host Control Protocol (DHCP) is defined by IETF RFC 2131 and modified by a host of other RFCs. It allows a machine, which at network initialization time does not know its own IP address, to request allocation of an IP address from a server and receive network configuration data sufficient to communicate on an IP network.
The Open Group Standards	
SM Spec	Signed Manifest Specification, The Open Group SM Spec Signed Manifest Specification, The Open Group, 1997. http://www.opengroup.org/pubs/catalog/c707.htm
DMTF Standards	
CIM	<p>The Distributed Management Task Force (DMTF) originally developed the Common Information Model (CIM) to provide a data model for integrating management across SNMP, the Desktop Management Interface (DMI) (another part of WBEM), Common Management Information Protocol (CMIP or ISO 9596) (for telecom devices) and private applications. CIM is part of the DMTF's overall Web-based Enterprise Management (WBEM) initiative. WBEM includes CIM as the data definition, XML as the transport/encoding method, and HTTP as the access mechanism.</p> <p>CIM is an object-oriented data model for describing managed elements across the enterprise, including systems, networks, and applications. The CIM schema provides definitions for servers, desktops, peripherals, operating systems, applications, network components, users, and others along with details of each. One of the main functions CIM offers is the ability to define the associations between components. CIM's object-oriented approach makes it easier to track the relationships and interdependencies between managed objects. WBEM/CIM proponents promote this as a key advantage over SNMP.</p>
WBEM	The Web-Based Enterprise Management (WBEM) standard is an initiative by the DMTF to develop a broader enterprise management structure than SNMP. The DMTF is an industry coalition that is developing an enterprise management framework for computer systems that is richer than SNMP

This Table is (U)	
(U) Name	(U) Description
SMBIOS	The System Management Basic I/O System (SMBIOS) is a DMTF standard for making firmware-level information available via a CIM model on computer systems.
Vendor Standards	
Intel PXE specification	The Intel-developed Preboot eXecution Environment (PXE) specification defines an OS-independent firmware-level mechanism for booting from a variety of media, including the network, using standard protocols. ftp://download.intel.com//labs/manage/wfm/download/pxespec.pd
Intel PXE BIS specification	The Intel PXE Boot Integrity Services is an extension to the Intel PXE specification that provides for PKI-based authentication of the server to the booting client. ftp://download.intel.com//labs/manage/wfm/download/bisspec.zip
This Table is (U)	

13440 **2.7.3.5.5 (U) Cost/Limitations**

13441 (U) Systems management applications can provide full management of client systems, but are
13442 extremely expensive—reaching \$1000 per client system or more in annual licensing costs.

13443 (U) All systems have no support for non-standard target machines. Although the general systems
13444 management applications can be extended to cover embedded systems or appliances, it is a
13445 custom software development. Specialized hardware such as IDS appliances, HAIPEs, or
13446 specialized military hardware with IA components are unsupported by any commercial
13447 implementation. Some applications can be extended to included non-standard clients, but this is
13448 only with custom software development.

13449 **2.7.3.5.6 (U) Dependencies**

13450 (U) Many of the current products assume a native patch management mechanism exists for the
13451 target machine such as the Microsoft Installer (MSI) for Microsoft Windows clients or
13452 something like Redhat Package Manager (RPM) for Linux clients, and either use it directly or
13453 develop a common proprietary packaging scheme that unpacks on the target machine into a
13454 native format. All of the configuration management tools depend upon the OS-native application
13455 version and configuration data to be correct and valid. None of the current products provide an
13456 independent server-based record of a client installation for comparison to the current
13457 configuration or validation of the contents of files.

13458 **2.7.3.5.7 (U) Complementary Techniques**

13459 (U) The determination of the optimal configuration of an IA device is intimately related to the
13460 vulnerabilities of that device and its associated software, so many configuration assessment tools
13461 are integrated with a general vulnerability assessment scanner, or they derive their configuration
13462 definition from a vulnerability assessment tool.

13463 **2.7.3.5.8 (U) References**

- 13464 (U) [SNMP02a] "An Architecture for Describing Simple Network Management Protocol
13465 (SNMP) Management Frameworks," RFC 3411; D. Harrington, R. Presuhn, B. Wijnen,
13466 December, 2002. <http://www.ietf.org/rfc/rfc3411.txt>
- 13467 (U) [SNMP02b] "Message Processing and Dispatching for the Simple Network Management
13468 Protocol (SNMP)," RFC 3412, J.D. Case, D. Harrington, R. Presuhn, B. Wijnen, December,
13469 2002. <http://www.ietf.org/rfc/rfc3412.txt>
- 13470 (U) [SNMP02c] "Simple Network Management Protocol (SNMP) Applications," RFC 3413. D.
13471 Levi, P. Meyer, B. Stewart. December, 2002. <http://www.ietf.org/rfc/rfc3413.txt>
- 13472 (U) [SNMP02d] "User-based Security Model (USM) for version 3 of the Simple Network
13473 Management Protocol (SNMPv3)," RFC 3414, U. Blumenthal, B. Wijnen. December, 2002.
13474 <http://www.ietf.org/rfc/rfc3414.txt>
- 13475 (U) [SNMP02e] "Management Information Base (MIB) for the Simple Network Management
13476 Protocol (SNMP)," RFC 3418, R. Presuhn, Ed.. December, 2002.
13477 <http://www.ietf.org/rfc/rfc3418.txt>
- 13478 (U) [CIM99] "Common Information Management (CIM) Specification, DSP004, The
13479 Distributed Management Task Force, Inc. Version 2.2," June 14, 1999.
13480 <http://www.dmtf.org/standards/documents/CIM/DSP0004.pdf>
- 13481 (U) [SMBIOS02] "System Management BIOS Specification, The Distributed Management Task
13482 Force, Inc. v2.3.4," DSP0134, December 6, 2002.
13483 <http://www.dmtf.org/standards/documents/SMBIOS/DSP0134.pdf>
- 13484 (U) [TFTP] "The TFTP Protocol," RFC 1350, Sollins, K. July, 1992. [ftp://ftp.rfc-editor.org/in-
13485 notes/rfc1350.txt](ftp://ftp.rfc-editor.org/in-notes/rfc1350.txt)
- 13486 (U) [DHCP97] "Dynamic Host Configuration Protocol," RFC 2131, R. Droms, March, 1997.
13487 <http://www.ietf.org/rfc/rfc2131.txt>
- 13488 (U) [WBEM] "WBEM Discovery using SLP, DSP0205," The Distributed Management Task
13489 Force, Inc. Version 1.0.0, Jan. 27, 2004. <http://www.dmtf.org/standards/wbem/DSP0205.pdf>
- 13490 (U) [INTEL] Intel-developed Preboot eXecution Environment (PXE) specification.
13491 <ftp://download.intel.com/labs/manage/wfm/download/pxespec.pdf>
- 13492 (U) Intel PXE BIS specification
13493 (U) <ftp://download.intel.com/labs/manage/wfm/download/bisspec.zip>
- 13494 (U) [CMS02] "Cryptographic Message Syntax," RFC3369, R. Housley, August, 2002,
13495 <http://www.ietf.org/rfc/rfc3369.txt>.
- 13496 (U) "Symantec Enterprise Security Architecture (SESA™)," Symantec Corporation, 2002.
13497 itpapers.zdnet.com/abstract.aspx?scid=284&tag=tu.sc.ont.dir3&x=80&docid=87493

13498 (U) "Virus and Vulnerability Classification Schemes: Standards and Integration," S. Gordon,
13499 Symantec. 2003. [http:// securityresponse.symantec.com/](http://securityresponse.symantec.com/)
13500 avcenter/reference/virus.and.vulnerability.pdf

13501 **2.7.3.6 (U) Inventory Management**13502 **2.7.3.6.1 (U) Technical Detail**

13503 (U) A key element to managing GIG assets is an ability to dynamically create and maintain an
 13504 accurate inventory of IA assets. There are three components to an automated inventory
 13505 management system—the data entry mechanism, central database, and reporting system. An
 13506 emerging technology to support data entry and collection is Radio-Frequency Identification
 13507 (RFID). RFID is a technology that offers the ability to add small radio transponders to objects
 13508 that respond to an RF signal with a small amount of information. With advances in
 13509 manufacturing technology, RFID tags are now small enough to be embedded in banknotes and
 13510 are rapidly become sufficiently inexpensive to attach to relatively inexpensive items, which
 13511 enables a large number of widespread inventory, supply-chain, and tracking and identification
 13512 applications. For a number of logistics applications the DoD is currently piloting RFID tags, and
 13513 USD/ATL has issued a policy memorandum [DOD04] specifying use of RFID tags for large
 13514 classes of logistics applications by January 1, 2005. They have significant advantages over other
 13515 approaches for inventory tagging:

- 13516 • (U) No physical contact or line of sight is required, only proximity—removal from
 13517 packaging is not a requirement
- 13518 • (U) They are relatively immune to dirt, chemicals, or temperature variations
- 13519 • (U) Many RFID tags can be read virtually simultaneously. This yields scan rates much
 13520 higher than barcodes that require manual scanning of each individual barcodes

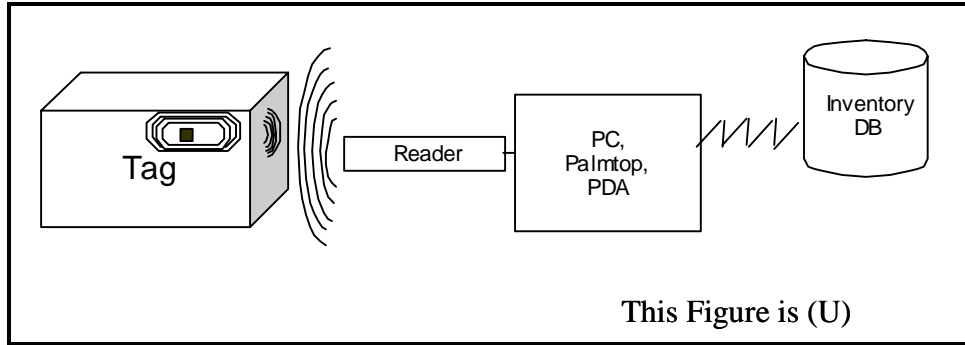
13521 (U) An RFID system is composed of three components:

- 13522 • (U) Tag
- 13523 • (U) Antenna
- 13524 • (U) Reader

13525 (U) The tag is a small electrical device that is—at its simplest— silicon chip connected to an
 13526 antenna. Other forms include a smart label or a rectangular case.

13527 (U) The reader is a device that reads RFID tags. There are many varieties, from small hand-held
 13528 devices to fixed readers for smart shelves or warehouse doorways. They may have integral
 13529 antennas or separately attached antennas. Readers are placed at key locations where they can
 13530 track tags as they pass automatically, such as in warehouse doorways, loading docks, and
 13531 inspection points. Emerging applications are smart shelves that can report their contents
 13532 automatically and readers on forklifts that automatically identify when the correct pallet is being
 13533 lifted or moved.

13534 (U) An example application is shown in Figure 2.7-10. The central inventory application is what
 13535 stores and processes the data from the reader. It can reside anywhere on the GIG, but it must be
 13536 accessible by the reader hardware and software.



13537

13538

Figure 2.7-10: (U) RFID Operation

13539

(U) RFID tags come in three major varieties:

13540

- (U) Active RFID tags consist of an antenna, transponder chip, and a power source such as a battery. They have a lifetime limited by the battery charge

13541

13542

- (U) Passive RFID tags have no power source, but are powered solely by the RF energy of the external reader

13543

13544

- (U) Semi-passive RFID tags have a power source to improve performance, but the power is used solely to power the internal circuitry during operation and is not used to generate RF signals. They are intermediate in cost and capability.

13545

13546

13547

(U) RFID tags can be manufactured in a variety of form factors. With printed or etched antennas and single-chip transponders, they can be manufactured as adhesive labels that can be read without physical contact. Range and data rate performance of RFID tags varies widely depending upon the type of tag and the environment. The minimum for passive tag ranges are a few inches and a capacity of 64 or 96 bits. Active tags can reach up to 100 ft with up to 2Kb data. Emerging UHF-band passive tags have longer ranges.

13548

13549

13550

13551

13552

13553

(U) The most inexpensive tags are read-only, set at manufacture. Other tags can be programmed once with an ID code. Read-write tags can store mutable information in addition to a fixed serial number.

13554

13555

13556

2.7.3.6.2 (U) Usage Considerations

13557

2.7.3.6.2.1 (U) Implementation Issues

13558

(U) RFID tags have some significant physical limitations, primarily in range. Passive tags must have close physical proximity to the reader to receive a strong enough signal to energize the circuit enough to send a detectable response (the signal strength varies as the fourth power of the distance). For close range devices, this is accomplished by making the reader a hand-held scanner, which then uses a conventional wireless communications technology, such as 802.11b, for communications with the network and central database server.

13559

13560

13561

13562

13563

13564 (U) As RF devices, RFID tags are affected by the same environmental considerations common to
 13565 all RF devices. Metal or conductive objects block RF signals. The antenna must be outside and
 13566 physically separated from a metal enclosure. Otherwise it acts a Faraday cage¹³—blocking all
 13567 signals. RFID tags are subject to interference from other RF sources, electrical equipment, or
 13568 motors. In addition, in many environments such as loading docks, readers may interfere with
 13569 each other, reading tags in other docks, requiring additional signal processing to avoid ambiguity
 13570 or errors.

13571 (U) Although RFID tags individually have low bandwidth requirements, when processing large
 13572 numbers of tags, the readers may use significant bandwidth to communicate with the host
 13573 application in real time. Hence, large amounts of equipment processing in low-bandwidth tactical
 13574 communications environments might use a significant amount of bandwidth.

13575 (U) Currently, RFID tags operate in regions of the frequency spectrum reserved for Industrial,
 13576 Scientific, and Medical (ISM) Applications, as detailed in Table 2.7-7.

13577 **Table 2.7-7: (U) Frequency Ranges for RFID Systems**

This Table is (U)		
Frequency Range	Comment	Allowed Field Strength / Transmission Power
< 135 kHz	Low frequency, inductive coupling	72 dBμA/m
6.765 - 6.795 MHz	Medium frequency (ISM), inductive coupling	42 dBμA/m
7.400 - 8.800 MHz	Medium frequency, used for EAS (electronic article surveillance) only	9 dBμA/m
13.553 - 13.567 MHz	Medium frequency (13.56 MHz, ISM), inductive coupling, wide spread usage for contactless smartcards (ISO 14443, MIFARE, LEGIC,), smartlabels (ISO 15693, Tag-It, I-Code,) and item management (ISO 18000-3).	42 dBμA/m
26.957 - 27.283 MHz	Medium frequency (ISM), inductive coupling, special applications only	42 dBμA/m
433 MHz	UHF (ISM), backscatter coupling, rarely used for RFID	10 .. 100 mW
868 - 870 MHz	UHF (SRD), backscatter coupling, new frequency, systems under development	500 mW, Europe only
902 - 928 MHz	UHF (SRD), backscatter coupling, several systems	4 W - spread spectrum, USA/Canada only
950 - 956 MHz	UHF (SRD), backscatter coupling, new frequency	Power TBD, Japan only
2.400 - 2.483 GHz	SHF (ISM), backscatter coupling, several systems, (vehicle identification: 2.446 .. 2.454 GHz)	4 W - spread spectrum, USA/Canada only, 500 mW, Europe
5.725 - 5.875 GHz	SHF (ISM), backscatter coupling, rarely used for RFID	4 W USA/Canada, 500 mW Europe
This table is (U//FOUO)		

¹³ (U) A Faraday cage is any conductive surface which surrounds an antenna. Any electromagnetic field is canceled inside a conductor, so no RF can ever pass through.

13578 (U) As shown, the emerging, UHF RFID spectrum is different for the United States, Europe, and
13579 Japan. This makes a common worldwide solution more challenging.

13580 **2.7.3.6.2.2 (U) Advantages**

13581 (U) RFID tags offer the ability to reliably process large numbers of IA components just by
13582 physical proximity. The ability to track pallet loads of devices automatically, merely by moving
13583 them through the warehouse door with no data entry error, represents a significant improvement
13584 in tracking. Smart shelves that know what items are stored on them and that can communicate to
13585 an inventory application can revolutionize inventory management.

13586 **2.7.3.6.2.3 (U) Risks/Threats/Attacks**

13587 (U) RFID tags are RF transponders that respond whenever they are probed. With an absolute
13588 minimum of circuitry for power and cost reasons, they contain no circuitry capable of supporting
13589 complex encryption, decryption, or authentication operations. Passive smart label RFID chips
13590 contain only enough circuitry to broadcast a 64- or 96-bit serial number. Although the RFID tag
13591 information itself would rarely be classified, to be useful, it must be connected to status and
13592 descriptive information for the component, which may be classified.

13593 (U) The third component of any RFID system is the host application. For IA component
13594 inventory and tracking applications, this will certainly involve sensitive or classified information
13595 such as current keysets and algorithms. As a result, either the database must operate in a multiple
13596 security domain configuration or the reader and all communications links must be capable of
13597 operating at the required assurance and confidentiality levels. Commercial hand-held RFID
13598 readers which use 802.11b/g for communications with the host application do not support the
13599 level of protection required for such information. Many offer applications which display the
13600 status of any component scanned on a local screen. When the inventory item is a high-value
13601 sensitive IA component or an element of a larger such component, communication with the
13602 centralized database becomes sensitive or classified.

13603 (U) In addition, there are a number of attacks that are possible with RFID systems:

- 13604 • (U//FOUO) Attack – Unauthorized Read Tag. An attacker can determine the inventory of
13605 sensitive equipment simply by using a commercial RFID reader, perhaps with an
13606 extended-range antenna to query the RFID tags in the same manner as an authorized user.
13607 This could present a very significant vulnerability in a battlefield or tactical environment
13608 where every tag represents the equivalent of a IFF transponder broadcasting a location.
13609 Tags are currently being developed which can be “deactivated” upon command, but they
13610 are primarily being developed in response to consumer privacy concerns, not
13611 authentication concerns, so the potential deactivation operations are permanent and non-
13612 reversible. More complex tags that allow soft deactivation and reactivation are being
13613 developed, but the cost will be significantly higher, and they will not have any
13614 authentication features.
- 13615 • (U//FOUO) Attack – Remove tag or cover tag – Tags which are mounted externally for
13616 shielding and range also become vulnerable to removal from the equipment, which in an
13617 automated environment would cause it to disappear from inventory and tracking. A
13618 similar result can be achieved with foil or a wire mesh covering the antenna.

- (U//FOUO) Attack – Replace tag ID information – More sophisticated RFID tags that have read-write capability will rewrite their data on any command from any RFID reader. No authentication is available. A handheld reader can transform a high-value sensitive piece of equipment into an innocent, low-value item for easy removal from the warehouse.

2.7.3.6.3 (U) Maturity

(U) Although RFID tags have existed since 1974, only within the last few years has the price of tags dropped to the level that makes them feasible for wide-scale deployment within the supply chain infrastructure. The DoD has issued and updated an RFID policy mandating the use of RFID tags for certain shipping containers and large pallet-sized shipments by Jan 1, 2005, with further expansion of use over the next few years. UHF tags, which appear to have the greatest promise for low-cost, long-range usage—ideal for inventory applications—are just now being developed by manufacturers and are not in widespread use. No readers currently operate at all three (U.S., European, and Japanese) UHF bands. The current drive is to reduce tag manufacturing costs, so security enhanced tag systems may be some time in coming.

(U//FOUO) A key element of RFID for GIG inventory management is that the RFID tags must be secure. Many IA assets will be used in combat, and inadvertent or adversary-triggered RF transmissions from RFID tags would be a serious vulnerability. A key enhancement would be the ability to activate and deactivate tags before and after missions. A greater issue is that current RFID tags have no authentication or authorization capability at all. Any reader can interrogate a tag, and any reader can write or rewrite writeable tags. With extremely limited on-board processing capacity, the capacity to restrict functions to authenticated, authorized readers is a number of years away.

(U) The maturity of tag technology for general inventory management is rated as Emerging (TRL 4-6). There are large-scale DoD and commercial pilot programs underway, such as those initiated by Walmart and Gillette. However, current pilot programs are not addressing secure RFID tags for assured inventory management, and significant vulnerabilities of conventional tags have not been addressed. Accordingly, maturity of RFID technology that would meet the security requirements of the GIG is rated as Early (TRL 1-3).

2.7.3.6.4 (U) Standards

(U) Table 2.7-8 lists the RFID standards applicable to Inventory Management

Table 2.7-8: (U) Inventory Management RFID Standards

This Table is (U)	
Name	Description
EPC Global Network Standards	
EPC Tag Data Specification Version 1.1	Identifies the specific encoding schemes for a serialized version of the EAN.UCC Global Trade Item Number (GTIN®), the EAN.UCC Serial Shipping Container Code (SSCC®), the EAN.UCC Global Location Number (GLN®), the EAN.UCC Global Returnable Asset Identifier (GRAI®), the EAN.UCC Global Individual Asset Identifier (GIAI®), and a General Identifier (GID)

This Table is (U)	
Name	Description
900 MHz Class 0 Radio Frequency (RF) Identification Tag Specification.	This document specifies the communications interface and protocol for 900 MHz Class 0 operation. It includes the RF and tag requirements and provides operational algorithms to enable communications in this band.
13.56 MHz ISM Band Class 1 Radio Frequency (RF) Identification Tag Interface Specification.	This specification defines the communications interface and protocol for 13.56 MHz Class 1 operation. It also includes the RF and tag requirements to enable communications in this band.
860MHz -- 930 MHz Class 1 Radio Frequency (RF) Identification Tag Radio Frequency & Logical Communication Interface Specification	This document specifies the communications interface and protocol for 860 – 930 MHz Class 1 operation. It includes the RF and tag requirements to enable communications in this band.
Physical Markup Language (PML)	The PML Core specification establishes a common vocabulary set to be used within the EPC global Network. It provides a standardized format for data captured by readers. This specification also includes XML Schema and Instance files for your reference.
ISO Standards	
ISO/IEC 15963:2004	Information technology -- Radio frequency identification for item management -- Unique identification for RF tags
ISO/IEC 18000-4:2004	Information technology -- Radio frequency identification for item management -- Part 4: Parameters for air interface communications at 2.45 GHz
ISO/IEC 18000-6:2004	Information technology -- Radio frequency identification for item management -- Part 6: Parameters for air interface communications at 860 MHz to 960 MHz
ISO/IEC 18000-7:2004	Information technology -- Radio frequency identification for item management -- Part 7: Parameters for active air interface communications at 433 MHz
This Table is (U)	

13651 **2.7.3.6.5 (U) Cost/Limitations**

13652 (U) Tags vary significantly in cost, depending upon their frequency range, application, and
 13653 whether they are active, semi-active, or passive. Current industry efforts are working to reach the
 13654 goal of \$0.05 for a passive smart-label tag, at which point a host of applications become
 13655 economically feasible. Current tags range from \$100 for complex, long-range active tags, to
 13656 approximately \$.50 to \$1.00 per tag in very high-volume applications. The major limitation for
 13657 GIG IA applications will be the cost of tags which can support the encryption and authentication
 13658 required to securely deactivate and reactivate RFID tags.

13659 (U) Readers vary in cost depending upon the type and range requirements. Fixed installation
 13660 systems with separate antennas can cost several thousand dollars. RFID readers in a PC Card
 13661 (PCMCIA) format are currently available for \$150.

13662 **2.7.3.6.6 (U) Alternatives**

13663 (U) Standard optical bar codes are an alternative to RFID tags, but they carry serious limitations.
 13664 Bar codes require line of sight to read so they must be external to packaging, unobstructed, and
 13665 facing the reader. This may require manual orientation of the scanner or the scanned item. Bar
 13666 codes can only be read one at a time by a scanner. Because they are exposed, printed barcodes
 13667 are susceptible to wear, dirt, marks, and water, or chemical damage, becoming unreadable. In
 13668 contrast, RFID tags can be sealed inside a relatively impervious container.

13669 **2.7.3.6.7 (U) Complementary Techniques**

13670 (U) RFID tagging systems only provide value when tied to updates of a centralized, real-time
 13671 asset management application. The application provides visibility into the inventory status, and
 13672 the RFID system provides real-time, highly accurate updates to the inventory.

13673 **2.7.3.6.8 (U) References**

13674 (U) [Chung] “Low Cost and Reliable RFID Tags for All Frequencies,” by Kevin Chung, [http://](http://itpapers.zdnet.com/abstract.aspx?kw=%20RFID&dtid=1&docid=89816)
 13675 itpapers.zdnet.com/abstract.aspx?kw=%20RFID&dtid=1&docid=89816

13676 (U) [DOD04] “Radio Frequency Identification (RFID) Policy,” Undersecretary of Defense for
 13677 Acquisition, Technology and Logistics (USD/ATL) Memorandum, July 30, 2004.
 13678 [http://www.acq.osd.mil/log/logistics_materiel_readiness/organizations/sci/rfid/assets/Policy/RFI](http://www.acq.osd.mil/log/logistics_materiel_readiness/organizations/sci/rfid/assets/Policy/RFID%20Policy%2007-30-2004.pdf)
 13679 [D%20Policy%2007-30-2004.pdf](http://www.acq.osd.mil/log/logistics_materiel_readiness/organizations/sci/rfid/assets/Policy/RFID%20Policy%2007-30-2004.pdf)

13680 (U) [Finkenzeller03] “Frequencies for RFID-systems,” by Klaus Finkenzeller, from RFID
 13681 Handbook, 2ed, tr. Rachel Waddington, Wiley & Sons, Ltd, April 2003.

13682 (U) [Hodges03] “Demystifying RFID: Principles and Practicalities,” by Steve Hodges, Mark
 13683 Harrison, October 1, 2003. www.autoidlabs.org/whitepapers/CAM-AUTOID-WH024.pdf

13684 (U) [Juels03a] “Minimalist Cryptography for Low-Cost RFID Tags,” by Ari Juels, 2003.
 13685 [http://www.eicar.org/.../11%20-](http://www.eicar.org/.../11%20-%20Minimalist%20Cryptography%20for%20RFID%20Tags.pdf)
 13686 [%20Minimalist%20Cryptography%20for%20RFID%20Tags.pdf](http://www.eicar.org/.../11%20-%20Minimalist%20Cryptography%20for%20RFID%20Tags.pdf)

13687 (U) [Juels03b] “The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy,” by
 13688 Ari Juels, Ronald L. Rivest, Michael Szydlo. In V. Atluri, ed. 8th ACM Conference on Computer
 13689 and Communications Security, pp. 103-111. ACM Press. 2003.
 13690 <http://www.rsasecurity.com/rsalabs/node.asp?id=2060>

13691 (U) [Ohkubo03] “Cryptographic Approach to “Privacy-Friendly” Tags,” by Miyako Ohkubo,
 13692 Koutarou Suzuki, Shingo Kinoshita, November 2003.

13693 (U) <http://www.rfidprivacy.org/papers/ohkubo.pdf>

13694 (U) [Rivas03] “RFID – its Applications and Benefits,” Mario Rivas, RFID Privacy Workshop @
 13695 MIT: November 15, 2003. <http://www.rfidprivacy.org/papers/rivas/rivas.pdf>

13696 (U) [Sarma03] "RFID Systems and Security and Privacy Implications," by Sanjay E. Sarma,
13697 Stephen A. Weis, Daniel W. Engel, [http://www.eicar.org/rfid/kickoffcd/04%20-](http://www.eicar.org/rfid/kickoffcd/04%20-%20Hintergrundinformationen/09%20-%20RFID%20Systems%20and%20Security%20and%20Privacy%20Implications.pdf)
13698 [%20Hintergrundinformationen/09%20-](http://www.eicar.org/rfid/kickoffcd/04%20-%20Hintergrundinformationen/09%20-%20RFID%20Systems%20and%20Security%20and%20Privacy%20Implications.pdf)
13699 [%20RFID%20Systems%20and%20Security%20and%20Privacy%20Implications.pdf](http://www.eicar.org/rfid/kickoffcd/04%20-%20Hintergrundinformationen/09%20-%20RFID%20Systems%20and%20Security%20and%20Privacy%20Implications.pdf)

13700 **2.7.3.7 (U) Compromise Management of IA Devices**13701 **2.7.3.7.1 (U) Technical Detail**

13702 (U//FOUO) Compromise Management is the management and actions required to respond to the
 13703 potential compromise of IA Devices. A device is compromised when the integrity and
 13704 confidentiality of data on that device cannot be assured or determined. Many IA Devices will be
 13705 operating in unprotected, partially protected or tactical environments where they may fall into the
 13706 hands of an adversary. At that point the capability to use the equipment to communicate on the
 13707 GIG must be removed.

13708 (U) Compromise Management consists of the following components:

- 13709 • (U) Compromise Detection
- 13710 • (U) Compromise Investigation
- 13711 • (U) Compromise Isolation
- 13712 • (U) Compromise Recovery.

13713 (U//FOUO) Compromise Detection is the ability to determine that an IA component has been
 13714 tampered with, either physically or logically. Many components have mechanisms to indicate
 13715 when tampering has occurred. Mechanisms that may indicate the physical integrity of a
 13716 component include:

- 13717 • (U) Physical labels that tear easily
- 13718 • (U) Tamper detection hardware, included in the component as part of the design
- 13719 • (U) Audit logs or alarms also form a component of compromise detection. These are
 13720 discussed further in Section 2.7.3.8, Audit Management
- 13721 • (U) Explicit regular external communication to check the status of the component. This
 13722 may be in the form of a SNMP status check or keep-alive timers on a physical link
- 13723 • (U) In the GIG environment, IA devices will spend more and more time in less and less
 13724 protected environments, and security will be dependent upon the internal IA device
 13725 protection or the network's ability to detect device or system compromise.

13726 (U) Compromise Detection – Tamper Mechanisms. The first key technology supporting
 13727 compromise detection is tamper resistance and detection. Tamper resistance is the use of
 13728 physical packaging to restrict the ability to physically alter or connect to components of a device.
 13729 Tamper detection is the addition of elements to the component to provide an active indication to
 13730 the system that a compromise is taking place. In many situations today, tamper detection is done
 13731 through physical means, such as seals. Seals can be applied to any physical enclosure or opening
 13732 to determine if an attempt has been made to open it. However, such mechanisms require physical
 13733 inspection by a knowledgeable person to determine if tampering may have occurred. Instead,
 13734 active measures must be incorporated into IA components to detect attempts to tamper with them
 13735 or compromise their integrity.

13736 (U) All high-assurance cryptographic modules must provide a means to detect tampering. NIST
 13737 FIPS 140-2 specifies federal requirements for cryptographic modules. For security level 2
 13738 modules, they must provide coatings or seals that will make tampering evident. It specifies that
 13739 for security level 3, components must zeroize any keys or sensitive parameters whenever the
 13740 device is opened. For level 4, components must have a high probability that any attempt to
 13741 tamper with the device or bypass the physical protection will result in device zeroization. A wide
 13742 variety of techniques are used to detect tampering, such as:

- 13743 • (U) Switches on access panels or lids
- 13744 • (U) Temperature sensors to detect attempts to manipulate the device by operating it
 13745 outside normal temperature parameters
- 13746 • (U) X-ray sensors to detect attempts to image the interior circuitry
- 13747 • (U) Ion-beam sensors to detect attempts to probe specific integrated circuit gates
- 13748 • (U) Voltage sensors to detect attempts to operate the device outside its normal voltage
 13749 parameters to force lockups or processing vulnerabilities
- 13750 • (U) Wire or optical fiber meshes assembled over components and sealed inside sealing
 13751 compounds that are wired to detect holes 50 um or larger.

13752 (U) In high-assurance components, a permanent battery powers these sensors for the life cycle of
 13753 the component, so that they are active even when the device is powered down or being shipped.
 13754 The standard response is that any keys or security parameters are zeroized or cleared. Due to
 13755 issues with standard static RAM remnants, this operation is considerably more complex than
 13756 simply removing power to SRAM memory. It generally involves at least writing multiple times
 13757 to each location to overwrite data.

13758 (U) Compromise Detection – Keep Alive Protocol. The current technology for external keep-
 13759 alive testing is the SNMP. Currently this is widely implemented as part of network management
 13760 products and is used for network status reporting, covered at length in Section 2.6.

13761 (U) Compromise Investigation is the ability to determine with a high assurance that a component
 13762 is either operating within its parameters or that it cannot be determined. Since many compromise
 13763 detection approaches are indirect, and only provide evidence of tampering, further investigation
 13764 may be required. This is a verification of the configuration of an IA device. This is described in
 13765 Section 2.7.2.5.

13766 (U//FOUO) Compromise Isolation is the ability to isolate a component that is no longer trusted
 13767 from the rest of the GIG. There are two components of this. The first is the reliable removal of
 13768 any keys from the IA component, or zeroization. The second is the notification of all other GIG
 13769 entities that may communicate with or use a component that it is not trustworthy. This is
 13770 accomplished through such mechanisms as CRLs or the Online Certificate Status Protocol
 13771 (OCSP). This is described in Section 2.7.2.4. For IA devices that do not use the PKI
 13772 infrastructure, key replacement is described in Section 2.7.2.3.

13773 (U) Compromise Recovery is the ability to restore a device to operation after its integrity has
 13774 been restored. In many cases, the compromise of a device may be temporary or in error—in
 13775 which case the device must be restored to service. There are two facets of Compromise recovery.
 13776 First, the configuration of the component must be restored. This means the software, data, and
 13777 firmware must be restored to a known, assured state, either by verification of the existing
 13778 configuration of the component or by reinitializing it and restoring the configuration. This is
 13779 described in Section 2.7.2.5, Configuration Management. Second, the trustworthiness of the
 13780 device must be communicated to its peers. These are certificate and key management issues,
 13781 which are discussed in Sections 2.7.2.4, Certificate Management and 2.7.2.3, Key Management,
 13782 respectively.

13783 **2.7.3.7.2 (U) Usage Considerations**

13784 **2.7.3.7.2.1 (U) Advantages**

13785 (U) These mechanisms are required for high-assurance devices such as INEs or HAIPes that
 13786 protect Secret or above data. FIPS 140-2 requires them for Level 4 devices used for high-
 13787 assurance unclassified operations.

13788 **2.7.3.7.2.2 (U) Risks/Threats/Attacks**

13789 (U/FOUO) The number and types of possible physical tampering attacks against IA devices
 13790 number in the hundreds [Weingart00]. We describe some of the broad characteristics of attacks
 13791 that must be considered.

13792 (U) Physical threats to IA enabled equipment have been characterized by three classes of
 13793 attackers:

- 13794 • (U) Class 1 - clever outsiders – It is assumed the attacker has limited knowledge of the
 13795 system, but can take advantage of known weaknesses. This typically characterizes
 13796 hackers.
- 13797 • (U) Class II -knowledgeable insiders – They have substantial specialized technical
 13798 experience and highly sophisticated tools and instruments. They include professional
 13799 researchers and academics.
- 13800 • (U) Class III funded organizations – Specialists backed by large funding sources, capable
 13801 of in-depth analysis, sophisticated attacks, and extremely advanced analysis tools. These
 13802 include criminal organizations and foreign governments.

13803 (U) The attacks can be characterized as well by the goal of the attacker:

- 13804 • (U) Steal keys – The attacker wants to extract unencrypted keys or cryptographic
 13805 parameters protected by a device for loading into another device
- 13806 • (U) Use equipment to continue communication – The attacker wants to control the device
 13807 and use it to continue communications for intelligence or further attacks
- 13808 • (U) Reverse engineering – The attacker wants to copy the device

- (U) Backdoor the device – The attacker wants to modify the device with a backdoor or Trojan without detection and allow its continued use while stealing data or further compromising the network.

(U) Each of these goals affects the type of attack from relatively simple non-invasive, non-destructive, attacks to invasive attacks which modify or destroy the device under attack.

2.7.3.7.3 (U) Maturity

(U) The mechanisms of tamper detection are understood, and current commercial products are available that incorporate them. However, in many cases the tamper response is limited to zeroizing the sensitive contents of the IA device. Currently only a few type 1 cryptographic devices, (e.g., HAIPE-compliant products) support SNMP management and so are physically capable of network alerts of tampering. However, current security policy is that tamper detection results in an immediate, non-interruptible response of zeroizing all communications keys, making it impossible for a device to securely send any communications such as a tamper indication to a central manager. Most commercial cryptographic modules only incorporate passive tamper resistance, only one device, the IBM 4578 cryptographic processor was evaluated to FIPS 140-1 Level 4 which mandates tamper detection. The Dallas Semiconductor DS5240 and DS5250 processors incorporate tamper detection but have not been FIPS evaluated. SNMP management of network devices is standard, and as additional commercial implementations of the specification emerge, network notification of tamper will become commercially available.

(U) The maturity of compromise management technology is assessed as Emerging (TRLs 4 - 6). Commercial products with limited capabilities are available. However, they are expensive and are not widely used or supported. Current GOTS equipment routinely incorporates zeroizing as a compromise response, but current designs do not define any possible mechanism by which communications with a management entity can occur after a zeroization. External compromise detection by keep-alive or heartbeat protocols can be implemented by current standard protocols, but no provision for explicit compromise signaling or detection exists.

2.7.3.7.4 (U) Standards

Table 2.7-9: (U) Compromise Management Standards

This Table is (U)	
Name	Description
NIST Standards	
FIPS 140-2	Security Requirements for Cryptographic Modules
IETF Standards	
SNMPv3	The Simple Network Management Protocol, version 3 is the latest version of the IETF standard for managing network devices. Version 3 includes authentication and authorization, so it is considered much more secure than previous versions. SNMP is widely implemented, but has some significant restrictions because of its very simple structure.
ISO Standards	
ISO/IEC 15408-1:1999	Information technology -- Security techniques -- Evaluation criteria for IT security -- Part 1: Introduction and general model

This Table is (U)	
Name	Description
ISO/IEC 15408-2:1999	Information technology -- Security techniques -- Evaluation criteria for IT security -- Part 2: Security functional requirements
ISO/IEC 15408-3:1999	Information technology -- Security techniques -- Evaluation criteria for IT security -- Part 3: Security assurance requirements
This Table is (U)	

13837 **2.7.3.7.5 (U) Cost/Limitations**

13838 (U) Cost is the major limitation on the use of tamper mechanisms. As a result, tamper
 13839 mechanisms are only implemented on high-assurance cryptographic equipment, either FIPS 140-
 13840 2 Level 4 or Common Criteria EAL 4, or above. The manufacturing complexity and limited
 13841 production of such components has meant that components incorporating tamper mechanisms
 13842 are extremely expensive relative to components certified to lower assurance levels.

13843 **2.7.3.7.6 (U) Complementary Techniques**

13844 (U) The primary complement to tamper mechanisms is an external approach using a keep-alive
 13845 protocol between the IA Component and an external source such as the Network Operations
 13846 Center. Common protocols such as ICMP were designed for testing a connection or the response
 13847 from a server. However continuing issues with using ICMP for DoS attacks has meant that it is
 13848 often turned off and certainly restricted to within an enclave.

13849 (U) The TCP includes the notion of a keep-alive packet that essentially checks at regular
 13850 intervals to see if the connection has been dropped on an otherwise idle TCP connection. It is a
 13851 null packet that serves only to generate a TCP disconnect if it does not go through. The negative
 13852 is that it only indicates that the connection failed, which can be due to transient network
 13853 conditions, and does not reflect the state of the connection endpoint host. However, a TCP
 13854 connection does consume network resources on both ends, so it does not scale well to large
 13855 numbers of systems.

13856 **2.7.3.7.7 (U) References**

13857 (U) [Anderson96] "Tamper Resistance – a Cautionary Note, in Second USENIX Workshop on
 13858 Electronic Commerce Proceedings," by Ross Anderson, Markus Kuhn; Oakland, CA. 1996.
 13859 <http://www.cl.cam.ac.uk/users/rja14/tamper.html>

13860 (U) [ATMEL04] "AT97SC3201 The Atmel Trusted Platform Module," Atmel Corporation,
 13861 www.atmel.com/dyn/resources/prod_documents/doc5010.pdf

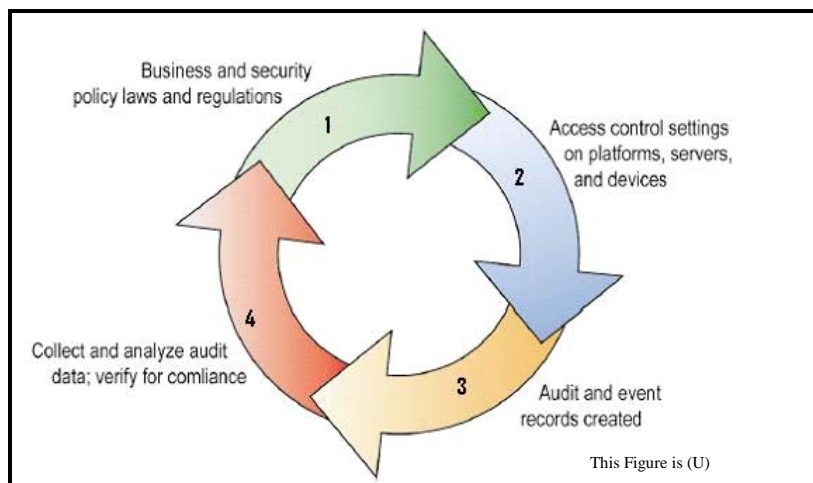
13862 (U) [Auer00] "Tamper Resistant Smartcards – Attacks and Countermeasures," by Auer, Eric;
 13863 <http://www-krypt.cs.uni-sb.de/teaching/seminars/ss2000/auer.pdf>

13864 (U) [Bajikar02] "Trusted Platform Module (TPM) based Security on Notebook PCs – White
 13865 Paper," by Sundeep Bajikar, June 20, 2002.
 13866 [developer.intel.com/design/mobile/platform/downloads/Trusted_Platform_Module_White_Paper](http://developer.intel.com/design/mobile/platform/downloads/Trusted_Platform_Module_White_Paper.pdf)
 13867 [.pdf](http://developer.intel.com/design/mobile/platform/downloads/Trusted_Platform_Module_White_Paper.pdf)

- 13868 (U) [CSV04] FIPS 140-1 and FIPS 140-2 Vendor List, NIST Cryptographic Standards and
13869 Validation Program, 2004. <http://csrc.nist.gov/cryptval/140-1/1401val-all.htm>
- 13870 (U) [Dallas03] "DS5250 High-Speed Secure Microcontroller," Dallas Semiconductor, July 18,
13871 2003. <http://pdfserv.maxim-ic.com/en/ds/DS5250-DS5250F.pdf>
- 13872 (U) [Johnston97] "Vulnerability Assessment of Security Seals," by R. G. Johnston and A. R. E.
13873 Garcia, Journal of Security Administration, 20, 15 (1997, <http://lib-www.lanl.gov/la-pubs/00418796.pdf>.
- 13874 (U) [Weingart99] "The IBM 4758 Secure Cryptographic Coprocessor Hardware Architecture
13875 and Physical Security," by S. H. Weingart, IBM Corporation, 1999.
13876 <http://www.cl.cam.ac.uk/Research/Security/seminars/1999/materials/weingart-19990222b.pdf>
- 13877 (U) [Weingart00] "Physical Security Devices for Computer Subsystems: A Survey of Attacks
13878 and Defenses," S. H. Weingart, Workshop on Cryptographic Hardware and Embedded Systems,
13879 2000.

13880 **2.7.3.8 (U) Audit Management**13881 **2.7.3.8.1 (U) Technical Detail**13882 **2.7.3.8.1.1 (U) Audit Life Cycle**

13883 (U) The typical lifecycle of an Audit process can be seen in Figure 2.7-11.



13884

13885

Figure 2.7-11: (U) Audit Life Cycle

13886 (U) Business and security policies form the first step in an audit life cycle. Policies are then
 13887 implemented via access controls that are put in place to enforce the rules of the policies. Access
 13888 controls mandate how users are authenticated and granted access to system resources. As users
 13889 conduct their business functions, Identity Management and other system components generate
 13890 audit events that are stored locally in log files or forwarded to event log databases. Finally, audit
 13891 and event data is collected and analyzed to verify that the intent of the business and its security
 13892 policies has been carried out.

13893 **2.7.3.8.1.2 (U) Auditing – Objectives**

13894 (U) Policy Compliance: Enterprises, such as the GIG, use systems and services that will need to
 13895 comply with business, security, legal, and regulatory mandates, such as SOX (Sarbanes-Oxley),
 13896 HIPAA, FISMA, DCID 6/3, and NISPOM. Thus, audit and event records need to be recorded
 13897 and monitored in order to provide the evidence that the GIG will use to demonstrate compliance.

13898 (U) Detecting Intrusions: Auditing is the ability to provide the means of detecting events that
 13899 result in a security breach of the GIG system. As such, the audit management of event logs
 13900 works closely with the collection services of the IDS and IPS systems. It is the latter's objective
 13901 to collect, analyze, detect, and react to the event log data for intrusions.

13902 (U) Determining Performance: Auditing also provides a means of independent review and
 13903 examination of records to determine the adequacy of system controls that ensure compliance
 13904 with established policies and operational procedures. This information serves as a resource for
 13905 the recommendation of necessary changes in controls, policies, and procedures. Auditing of
 13906 system resources should provide the information needed to reconfigure these resources to
 13907 improve system performance.

13908 (U) Accountability: Auditing will be used to identify an individual, process, or event associated
13909 with any security-violating event. In order to provide a complete audit picture, data must be
13910 collected and classified according to one of a number of areas of concern. This multi-
13911 dimensional approach would include an audit recording based on a subject's attributes, a time
13912 tagged object, and the state of a system resource. The subject will be tied to an audit event record
13913 via the individual's identification data, if that is tagged appropriately.

13914 (U) Access to the GIG will require authentication of the individual attempting to log into the
13915 system. The user login event will be recorded in the audit log along with any security-related
13916 audit events associated with the individual user. An identifier that will uniquely identify the user
13917 will be logged for these events. Object-based auditing identifies an audit event by an identifier of
13918 a modifiable security related data item such as a file on a storage medium. This identifier must
13919 include the name of the file and the storage volume identifier. An audit event would be generated
13920 whenever a security related object, such as a configuration file, was modified. The resource
13921 identifier is used in the auditing of system resources, such as network throughputs or the
13922 percentage of idle time during specified intervals or periods.

13923 (U) Robustness: Audit logs and the data contained in them represent valuable information,
13924 especially to adversaries who are attempting without detection to intrude and compromise a
13925 system. Such undetected activities of intruders could wreak significant havoc, such as the
13926 unleashing of malware, denial of service attacks, espionage, and other harm. Consequently, audit
13927 data and services must be strongly secured, employing the most robust access control standards
13928 possible for each situation.

13929 (U) Log Analysis: Logs and event records created by infrastructure systems are part of the
13930 evidence trail of what happens during the course of business for an enterprise. By examining
13931 audit logs, GIG systems can determine whether security components are properly enforcing
13932 policies and regulations to provide accountability in the event that non-compliance occurs. Audit
13933 log analysis can also reveal valuable information on patterns and exceptions. Long-term trends or
13934 usage patterns can help system planners adjust to customer habits; support forensic analysis for
13935 investigations into fraudulent activity; and harden targeted servers on sensitive systems that are
13936 experiencing attacks. Monitoring audit and event data in real time can enable enterprises to react
13937 to attacks in progress or new threats as they emerge.

13938 **2.7.3.8.1.3 (U) Audit Trails – Flow, Formats and Storage**

13939 (U) Figure 2.7-12 shows the typical information flows associated with audit trails. Audit records
13940 are generated at sources that include network devices, operating systems, and applications. The
13941 records are transported either internally within systems using inter-process communications or
13942 over networks using network protocols to storage media. Stored audit information along with
13943 other information from the system is either analyzed within the system under examination or by
13944 using separate analysis stations.

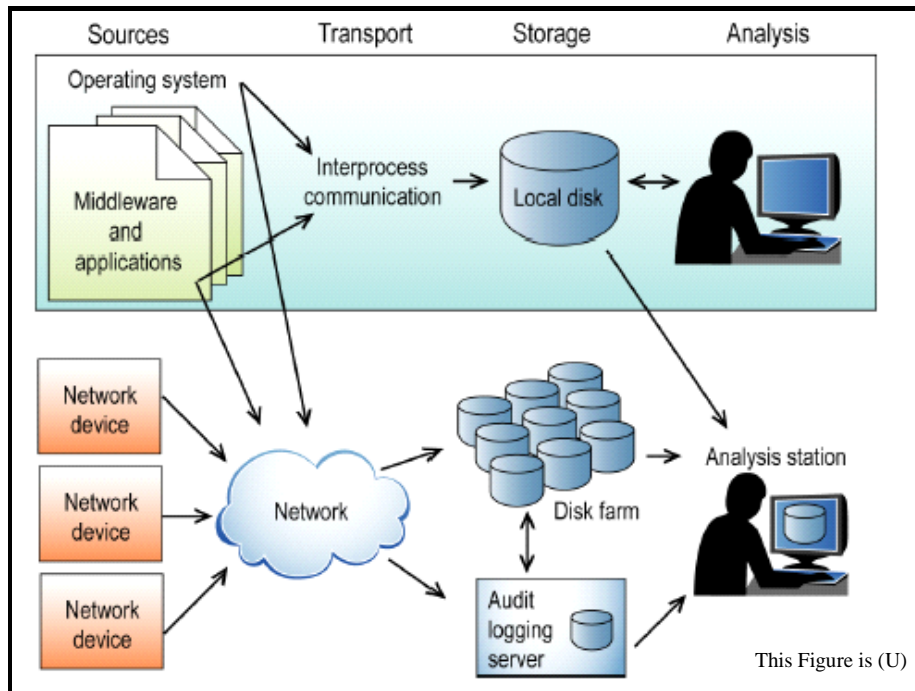


Figure 2.7-12: (U) Audit Trail Information Flow

13945

13946

13947 (U) Audit sources generate audit records in a wide variety of formats and transmit and store them
 13948 using a range of different techniques. Communication within a single system via inter-process
 13949 communication is usually effective at retaining integrity, but storage within the system under
 13950 examination makes these records subject to attack by anyone circumventing system security.
 13951 Analysis of audit trails and the systems they are supposed to reflect can be quite complex and
 13952 time consuming depending on the audit's objectives. Analysis within the system under
 13953 examination creates audit integrity and other related problems. With the exception of low
 13954 assurance casual audits, audits within trusted systems, and special cases where there are no other
 13955 options, information flows that remain entirely within a single system should be avoided.

13956 **2.7.3.8.1.4 (U) Providing Reports.**

13957 (U) An important aspect of Audit management is the ability to provide Conformance and
 13958 Compliance Reports to show that user and system activities are indeed complying with the
 13959 governing policies. These compliance reports should be generated automatically, periodically,
 13960 and on demand.

13961 (U) Compliance reports are used by auditors and review management. The reporting technology
 13962 should provide many types of views to help management visualize the findings and take
 13963 appropriate action based on the assessments. Higher levels of reviewing typically involve
 13964 Visualization and UI (User Interface) reporting tools that visually depict details or summaries in
 13965 multiple dimensions (3D), indicate weak points or failures, provide overviews of the operational
 13966 security health of the infrastructure, as well as indicate conformance to policy, compliance, or
 13967 lack thereof. These reports can be useful in conducting further risk analysis, as well as for
 13968 improving the process and resource provisioning of the system.

13969 **2.7.3.8.2 (U) Usage Considerations**

13970 **2.7.3.8.2.1 (U) Implementation Issues**

13971 2.7.3.8.2.1.1 (U) Monitoring and Verification of Compliance to Policy

13972 (U) A number of policy categories are required to be supported:

- 13973 • (U) Regulatory policies: FISMA, DCID 6/3, NISPOM, SOX (Sarbanes-Oxley), etc.
- 13974 • (U) Intrusion Detection (IDS, IPS) based policies
- 13975 • (U) Configuration Management policies, such as software and hardware upgrade policies.
13976 These include IAC CM policies, such as the updates and patches applied to application
13977 software, virus detection software, etc.

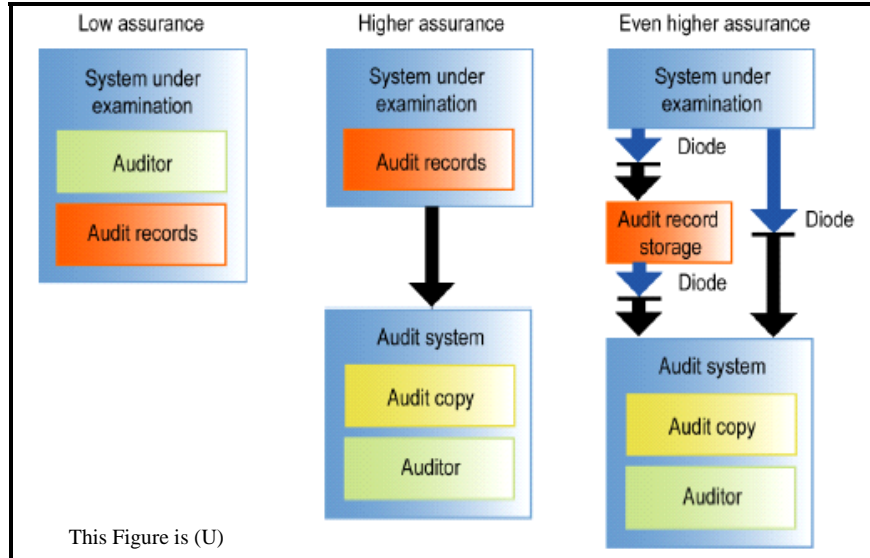
13978 (U) There is technology currently available that aids in capturing and applying policy statements
13979 via software tools and then using the stipulated policy rules to monitor and verify that the system
13980 or enclave activities are taking place within the rules. However, the main issue of concern is that
13981 today's solutions are mainly point solutions. Each vendor's software is proprietary in nature,
13982 differs from the others, and as such best of breed components from among different vendor
13983 choices cannot be selectively mixed. The major reason for this is the lack of standards that would
13984 allow policy rules to be specified and monitored in a uniform and normalized manner. Hence,
13985 there is little interoperability between vendor products.

13986 2.7.3.8.2.1.2 (U) Tamper Resistance of Logs

13987 (U) The following assertions and discussion are based on Figure 2.7-13:

13988 (U) Low Assurance Architectures are Usually Inadequate:

13989 (U) A low assurance architecture has an auditor logged into the system while it is operating. This
13990 presents the potential for the auditor to alter and affect the system, for the auditor to be fooled by
13991 the system under examination, for those under audit to detect the presence of the auditor, for the
13992 auditor to damage the system under examination, for audit trail loss or damage, or for the
13993 revelation of audit records in unauthorized ways. Such audit architectures should only be used in
13994 low-risk situations (low threat and low consequence), involving audits that are not related to
13995 regulatory compliance and where any of these consequences from the audit are acceptable.



13996

13997

Figure 2.7-13: (U) Audit Logs – Protection

13998

(U) Higher Assurance Architectures are Advised for Medium Risks:

13999

(U) A higher assurance architecture separates the auditor from the system under examination. If properly implemented, no information flows from the audit system to the system under examination, and the audit system includes a copy of all audit records as well as a forensically sound copy of the contents of the system under examination. In this example, audit records can be attacked within the system under examination, but the auditor can have no effect on that system or the audit records. It is impossible for the users of the system to know from the system whether an audit is underway, and the auditor can operate without concern about harm to the system under examination or subversion by the system under examination. This architecture is acceptable in most medium-risk situations (medium or lower threats and medium or lower consequences) and is normally acceptable for regulatory compliance audits in cases when loss or subversion of audit records is acceptable. In cases where audit records are required, such as under Gramm Leach Bliley regulations, this approach is inadequate, because the original audit records can be subverted.

14000

14001

14002

14003

14004

14005

14006

14007

14008

14009

14010

14011

14012 (U) Even Higher Assurance Architectures are Advised for High Risk:

14013 (U) An even higher assurance architecture adds independent audit trail storage and higher
14014 assurance separation of the audit trails and auditor from the system under examination. The use
14015 of digital diodes (systems that enforce one-directional information flows) provides high
14016 assurance against backflows of information, while the use of an external audit record storage
14017 device separates the audit records from those who might seek to subvert the audit trail. The
14018 auditor is protected against subversion, the system is protected from the auditor, and the audit
14019 records are protected from attackers. For additional assurance, redundant copies of audit trails
14020 can be generated and stored, additional coding can be used to verify records in transmission and
14021 storage, records can be generated from multiple sources associated with the system under
14022 examination, and higher assurance components can be used. There are audit servers on the
14023 market designed to implement the audit record storage requirements of this architecture, and
14024 most system audit mechanisms provide the means to transmit audit records as they are generated
14025 to remote systems over a network. Some audit servers also provide reasonable assurance against
14026 information backflows, forming different assurance levels of diode protection. This network
14027 audit architecture should be used for situations in which threats or consequences are high and
14028 regulatory compliance mandates effective auditing.

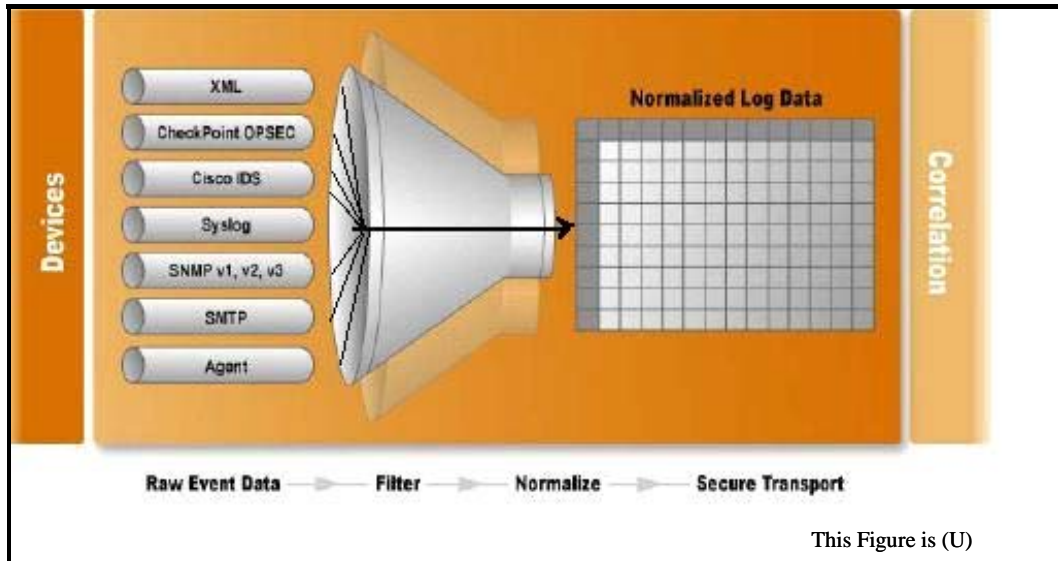
14029 2.7.3.8.2.1.3 (U) Log Formats and Event Records

14030 (U) A lack of standard message formats and exchange protocols intensifies the problem of
14031 coping with the huge data volume. Operating systems, firewalls, application servers, intrusion
14032 detection systems, and other network components create proprietary record formats that must be
14033 normalized before additional correlation analysis can be performed. Auditing systems, including
14034 directory, access management, and provisioning servers, contribute to the chaos with their
14035 mostly inadequate auditing features that require manual handling of nonstandard records, and
14036 often with no unified audit view within their product boundary—and certainly none beyond it.
14037 Without standard exchange protocols, software vendors have little reason to do more than write
14038 their own proprietary log records, and audit tools vendors are forced to write platform-specific
14039 agents, parse diverse log file formats, or rely on sparse protocols like the SNMP to transmit data
14040 and event information to central servers.

14041 (U) A partial list of supported devices, each with their own vendor-specific log formats:

- 14042 • (U) Firewall products
- 14043 • (U) Antivirus products
- 14044 • (U) Intrusion detection products
- 14045 • (U) Routers and switches
- 14046 • (U) SYSLOG

14047 (U) Various devices that record and log events are shown in Figure 2.7-14.



14048

14049

Figure 2.7-14: (U) Aggregation and Normalization

14050 (U) Audit log data, indicated as raw event data in Figure 2.7-14, is collected from various
 14051 devices and sensors on the network. This raw data are either typically pulled or monitored from a
 14052 central monitoring facility, or are pushed out the devices via agent technologies. Regardless of
 14053 the manner in which the data is collected, the data then undergoes filtration, aggregation, and
 14054 normalization into a unified format before it is further transported (securely) to analytical and
 14055 correlation engines for intrusion or anomaly detection. Current technologies for normalization
 14056 are manifested in the form of custom middleware that performs the normalization into formats
 14057 only understood by the custom vendor provider. This is because standards that provide
 14058 normalized formats do not currently exist. As such, normalization is subject to vendor
 14059 interpretation and consequential errors.

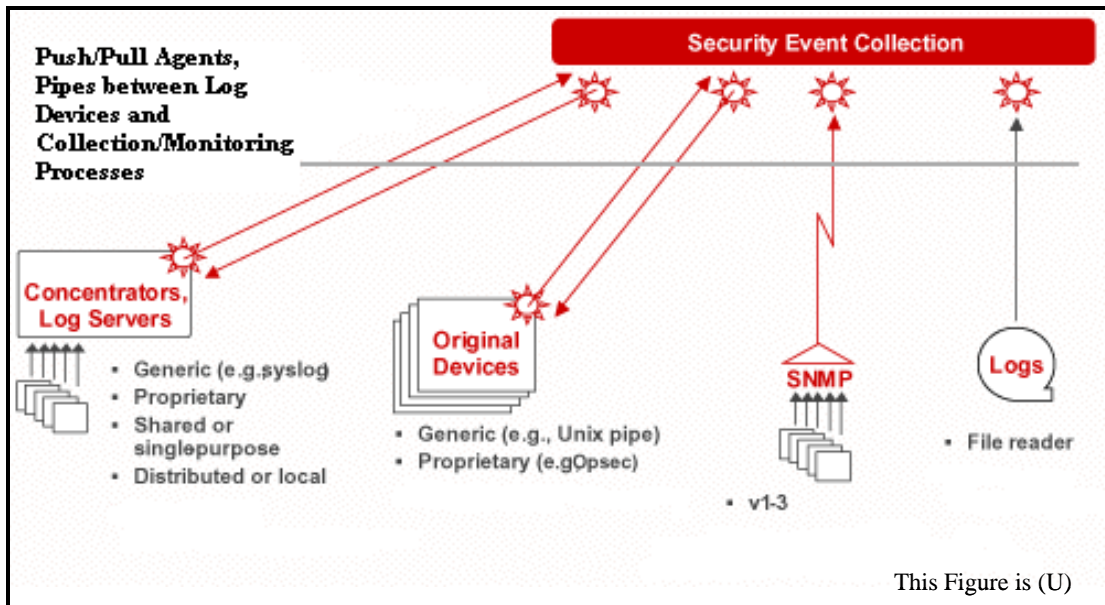
14060 (U) The type of attributes surrounding auditable events also vary among the various devices,
 14061 sensors, operating systems, and platforms. Due to lack of standards or policies, not all
 14062 implementations of log events capture the following essential attributes:

- 14063 • (U) Subject – The person accessing the object
- 14064 • (U) Object – The target object that is accessed by the subject
- 14065 • (U) Resource – Monitor items like throughput and idling time, used for performance and
 14066 utilization measurements
- 14067 • (U) Time Stamps – When the activities occurred
- 14068 • (U) Event Status – Success/Failure with appropriate codes.

14069 2.7.3.8.2.1.4 (U) Collection Services

14070 (U) Push versus Pull Agents: Some software vendors provide agents to forward logs to a
 14071 Security Operations Center (SOC); this is considered a push model, since the host or target
 14072 device pushes data out to the collection side via a custom host agent. There are also central
 14073 monitoring services that are agent-less; that is, they do not require forwarding agents at host
 14074 sites, but instead use technology to pull device logs from a central facility (e.g., SOC). These
 14075 central monitoring services poll the distributed network of hosts and remote devices' logs at
 14076 specified intervals. These collection modes can be seen in Figure 2.7-15.

14077 (U) There are currently no standards-based specifications that prescribe interfaces between
 14078 central and host agents—or specify how to achieve interoperability.



14079

14080 **Figure 2.7-15: (U) Interfaces - Agents and Pipes between Log Devices and the**
 14081 **Collection/Monitoring Processes**

14082 2.7.3.8.2.1.5 (U) Log Reduction and Archiving (Log Retention)

14083 (U) Event reduction and full logging are opposing methodologies. The motivation for event
 14084 reduction is to reduce the event data set in order to quickly detect deviations from the operational
 14085 norm for IDS and IPS purposes; this is achieved by filtering out the noise or non-threat event
 14086 information. Whereas, full logging and complete archives capture and log every event. This is
 14087 done for computer forensics needs that include criminal investigations, where the complete audit
 14088 trail archive is a requirement for the chain of evidence. The extent of log archives, reduction and
 14089 retention will depend on the situational system policy or policies being established. Policies that
 14090 require both Event Reduction (e.g., for IDS and IPS needs) and full logging (Forensic needs)
 14091 need to be supported.

14092 **2.7.3.8.2.2 (U) Advantages**

14093 (U) Management can be implemented with or without agents.

14094 (U) Agent technologies are more suited for host-based logging and monitoring. They have the
14095 advantage of being able to log events even when the connection to the network goes down or is
14096 unavailable for any reason. Agent technologies tend to push data out of the logs, periodically, to
14097 a central monitoring service on the network. They do require periodic configuration and
14098 maintenance however.

14099 (U) Agentless technologies are built into centralized monitors and have the distinct advantage of
14100 eliminating the need for host-based agents. This eliminates the maintenance that would otherwise
14101 be required on a host system. But the central monitors do have a few disadvantages. They depend
14102 on the availability of the network. Central monitoring is also more complex; it requires keeping
14103 an up-to-date list of target hosts and routers whose logs need monitoring.

14104 (U) Management can also include all logs, or reduced logs.

14105 (U) Managing Full-logs (i.e., no reduction) has the advantage of being simpler to implement. But
14106 this also imposes higher stress levels on network bandwidth and storage.

14107 (U) Log reduction management is just the opposite. It works well with comparatively modest
14108 bandwidth and storage requirements, but requires the maintenance of complex analytical
14109 software that can accurately filter out non-threat noise from the real threat related events.

14110 **2.7.3.8.2.3 (U) Risks/Threats/Attacks**

14111 (U) Audit logs run the risk of being a target for attack due to the valuable information contained
14112 in them. Thus, managing the audit data requires high assurance and tamper resistance. Assurance
14113 implies the confidentiality, integrity and continuous availability of the audit trails and logs data.

14114 (U) Audit data represents valuable policing information and is thus highly desirable as a target
14115 for attack, stealing, or modification. Consequently audit data should be protected from
14116 unauthorized access or compromise and needs to be secured at every step whether the audit data
14117 is at rest in a latent log file or in motion (transported over the network for analysis and post
14118 processing). Appropriate access-controls and hardening principles need to be in place to ensure
14119 the integrity and proper authorized access to the audit event data.

14120 (U) Audit data should also be made available, on demand, for urgent or immediate processing
14121 needs. Thus, provisions for continuous availability of the data are required for consideration.
14122 This would include backup and fault tolerant audit databases.

14123 (U) Audit technologies are also affected by the dynamic nature of policy changes. Dynamic
14124 Policy Management states that policies and their rules can change dynamically based on
14125 situational and directive changes. Consequently, auditing mechanisms are then at the risk of
14126 being outdated quickly, and if the technologies do not permit the adaptability of auditing
14127 processes to new policies and rules, then false positive or false negative reporting can occur as a
14128 result—thereby defeating the auditing mission.

14129 **2.7.3.8.3 (U) Maturity**

14130 (U) The Audit management market appears to be somewhat mature today, but products exist
 14131 only as point solutions. As indicated earlier, vendor solutions can be found in the SEM market
 14132 today. The SEM vendors provide all the middleware that tie together the various steps of audit
 14133 monitoring, collection, filtering, and normalization. But their solutions are proprietary in nature,
 14134 and there is little or no interoperability between the various facets of secure event management
 14135 and auditing capabilities.

14136 (U) The efforts of groups like IDMEF, CIDF, and CERIAS are still largely unknown. They have
 14137 yet to emerge with concrete standards and are outlined in next section on Standards.

14138 (U) Audit Management today exhibits a lack of maturity in standards-based solutions. This
 14139 makes componentization and interoperability in the different phases of audit management very
 14140 difficult. Standards are needed to prescribe log formats, normalized records, interfaces with
 14141 collection processes, and policies directed towards secure storage as well as secure transport
 14142 mechanisms to and from hosts and collection/analytical agencies.

14143 (U) Audit management technologies are assigned an overall maturity level of Emerging (TRLs 4
 14144 - 6). This is based on the middleware technologies (point solutions) available in the commercial
 14145 SEM market. However, standards for GIG-wide audit log formats, aggregation and
 14146 normalization of records, and interfacing to audit analysis processes that include IDS and IPS
 14147 systems, need to be devised and adopted.

14148 **2.7.3.8.4 (U) Standards**

14149 (U) A general lack of standards is one of the main challenges to the collection and correlation of
 14150 security events from heterogeneous systems. The few existing standards (Table 2.7-10) are still
 14151 in development, have not gained significant acceptance in the industry, or are narrowly focused
 14152 on a particular technology area. Some vendors have started using eXtensible Markup Language
 14153 (XML) to describe the event records in their repositories, but the formats are still proprietary.

14154 **Table 2.7-10: (U) Audit Management Standards**

This Table is (U)	
Name	Description
IETF Standards	
CLF	Common Log Format. Typically, the information is presented in plain ASCII without special delimiters to separate the different fields. See http://www.ietf.org
ELF	Extended Log Format
IDMEF	Intrusion Detection Message Exchange Format. The IETF's Intrusion Detection Working Group (IDWG) is developing message formats and procedures for sharing messages between intrusion detection systems and the SEM systems that manage them. The IDMEF requirements were posted as an Internet Draft in October, 2002, along with a draft of the Intrusion Detection Exchange Protocol (IDXP). In January, 2003, an Internet Draft was submitted for IDMEF that included an XML implementation. This initiative is still in development and it's future is uncertain.
RFC 1155,	Structure of Management Information

This Table is (U)	
Name	Description
RFC 1156	Management Information Base (MIB-I)
RFC 1157	SNMP
RFC 1187	Bulk table retrieval
RFC 1212	Concise MIB definitions
RFC 1213	Management Information Base (MIB-II)
RFC 1215	Traps
RFC 1227	SNMP Multiplex (SMUX)
RFC 1228	SNMP-DPI
RFC 1229	Generic-interface MIB extensions
RFC 1239	Reassignment of MIBs
RFC 1243	AppleTalk MIB
RFC 1248	OSPF MIB
IEEE Standards	
1230 IEEE 802.4	Token Bus MIB
1231 IEEE 802.5	Token Ring MIB
ISO Standards	
ISO 8824-1	Abstract Syntax Notation One (ASN.1): Specification of basic notation
ISO 8824-2	Abstract Syntax Notation One (ASN.1): Information object specification
ISO 8824-3	Abstract Syntax Notation One (ASN.1): Constraint specification
ISO 8824-4	Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications
ISO 8825-1	ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
ISO 8825-4	ASN.1 encoding rules: XML Encoding Rules (XER)
Other Standards Efforts	
Common Intrusion Detection Framework	<p>http://gost.isi.edu/cidf/</p> <p>Taken from above website: "The Common Intrusion Detection Framework (CIDF) is an effort funded by DARPA to develop protocols and application programming interfaces so that intrusion detection research projects can share information and resources and so that intrusion detection components can be reused in other systems."</p> <p>It appears that the CIDF initiative started in 1997, but has yet to materialize into an accepted standard. The work is still under development.</p>
Open Security Exchange	<p>The Open Security Exchange in April, 2003, announced specifications to enable more effective and interoperable security management across physical and IT security systems. A focal point of the specifications is to improve the auditability of systems. The Open Security Exchange (www.opensecurityexchange.com), founded by Computer Associates, HID Corporation, Gemplus, and Tyco, was created to address today's lack of integration between various components of security infrastructures.</p> <p>See: www.opensecurityexchange.com</p>
This table is (U//FOUO)	

14155 2.7.3.8.5 (U) Costs/Limitations

14156 (U) The development of standards needed to provide a GIG-wide common log format, and
14157 aggregation and normalization scheme—as well as interoperable standards—might prove to be
14158 difficult and costly. Industry working groups such as the CIDF and IDMEF mentioned earlier
14159 have been stymied in the process of unifying and standardizing formats and information
14160 exchanges among disparate systems.

14161 (U) The progress of these groups and initiatives is still tentative. The difficulty likely arises from
14162 political battles that affect current SEM vendors who have captured niche markets based on their
14163 point solutions and custom middleware. Standardization in the recording, storage, collection,
14164 analysis, monitoring and reporting phases will increase competition among these SEM vendors
14165 for each of these phases. Thus, there is little incentive for existing vendors to conform to
14166 component and application-based standards that would result in sacrificing niches in the SEM
14167 market to competition. However, as pointed out earlier, these limitations have to be overcome or
14168 at least reduced in order to provide a GIG-wide automated auditing solution.

14169 2.7.3.8.6 (U) Dependencies

14170 (U) A GIG-wide unified and automated audit technology solution will strongly depend on
14171 overcoming the limitations described earlier, and the advancement and adoption of standards-
14172 based recording, collecting, and monitoring solutions.

14173 2.7.3.8.7 (U) Alternatives

14174 (U) The alternative to utilizing automation with audit management is to use manual methods –
14175 which is not a viable solution. Manual methods and paper trails and have proven to be tedious,
14176 inefficient and unreliable. With the advent of smarter and faster-acting attacks, there is the need
14177 for immediately detecting deviations from normal operations. This includes especially the
14178 detection of zero-day attacks. Automating the four phases of audit management lifecycle appears
14179 to be the prudent approach.

14180 (U) The alternative to adopting a unified standards-based GIG technological solution is to select
14181 individual SEM and middleware solutions for various needs. In fact, this is the modus operandi
14182 in today's commercial enterprises. The obvious disadvantage with this solution is the dependency
14183 reliance on the vendor to provide a holistic solution.

14184 2.7.3.8.8 (U) Complementary Technologies

14185 (U) Collection and Analysis-based technology standards at the back end, such as those found
14186 commonly in IDS (Intrusion Detection Systems) and IPS (Intrusion Prevention Systems) in
14187 particular, should complement the development of audit-analysis and audit-collection technology
14188 standards. Also, dynamic policy technology standards at the front-end should complement audit-
14189 recording technology development.

14190 2.7.4 (U) Management of IA Mechanisms & Assets: Gap Analysis

14191 (U//FOUO) Table 2.7-11 summarizes the adequacy of the technologies to meet the needs of this
14192 IA Enabler.

14193

Table 2.7-11: (U) Technology Adequacy for Management of IA Mechanisms and Assets

This table is (U//FOUO)										
		Technology Categories								RCD Attributes
		Identity Management	Privilege Management	Key Management	Certificate Management	CM of IA Devices and Software	Inventory Management	Compromise Management	Audit Management	
IA Attributes	GIG Identity Management		N/A	N/A	N/A	N/A	N/A	N/A	N/A	IAIR1, IAIR2, IAIR3, IAIR4, IAIR5, IAIR6, IAKCM40, IAUAM1-IAUAM3
	GIG Authorization and Privilege Management	N/A		N/A	N/A	N/A	N/A	N/A	N/A	IAAM1, IAAM2, IAAM3, IAAM4, IAAM7, IAAM8, IAAM9, IAAM11, IAKCM41
	Policy based Access Control	N/A	N/A	N/A		N/A	N/A	N/A	N/A	IAAM12
	GIG Remote IA Asset Management	N/A	N/A	N/A		N/A	N/A	N/A	N/A	IANMA1
	OTN Benign Fill	N/A	N/A			N/A	N/A	N/A	N/A	IAKCM7, IAKCM9
	IA Asset Inventory Management	N/A	N/A	N/A	N/A			N/A	N/A	IAPS2, IAPS3, IANMA1, IANMA2
	Assured IA Asset Configuration Management	N/A	N/A		N/A		N/A	N/A	N/A	
	IA Asset Compromise Management	N/A	N/A	N/A	N/A	N/A	N/A		N/A	IAKCM35
	IA Asset High Robustness	N/A	N/A	N/A	N/A				N/A	

This table is (U//FOUO)										
		Technology Categories								RCD Attributes
		Identity Management	Privilege Management	Key Management	Certificate Management	CM of IA Devices and Software	Inventory Management	Compromise Management	Audit Management	
	GIG Key Management	N/A	N/A		N/A	N/A	N/A	N/A	N/A	IAKCM1-IAKCM9, IAKCM12-IAKCM17, IAKCM24-IAKCM27, IAKCM33, IAKCM34, IAKCM36-IAKCM38
	GIG Cert Management	N/A	N/A			N/A	N/A	N/A	N/A	IAKCM18, IAKCM19, IAKCM23, IAKCM24, IAKCM27, IAKCM39, IAKCM40, IAKCM43-IAKCM52, IANRP6
	GIG Coalition Key Management	N/A	N/A			N/A	N/A	N/A	N/A	IAKCM29, IAKCM30, IAKCM53
	GIG Package Management									IAKCM32
	GIG Management Auditing	N/A		N/A	N/A	N/A	N/A	N/A		IAIR5, IAAM11, IAKCM28, IAEM23, IANMP4
	GIG Audit Logging and Analysis	N/A	N/A	N/A	N/A	N/A	N/A	N/A		IAIAC7, IAAUD1-IAAUD10
	GIG CM Management	N/A	N/A	N/A	N/A			N/A	N/A	IACM1-IACM5, IASA05
This Table is (U//FOUO)										

14194 **2.7.4.1 (U) Identity Management**

14195 (U//FOUO) Provisioning and Maintenance Standards – Currently SPML is the only standard for
14196 provisioning, and there are no major standards for ongoing maintenance operations. SPML is a
14197 relatively new standard that needs wider adoption before it can stabilize. A maintenance standard
14198 needs to be developed to allow disparate aspects of an identity management enterprise to manage
14199 existing identities. These standards are required for an identity management deployment at the
14200 DoD level to support GIG activities. Once developed, these standards need to be integrated into
14201 new and existing identity management products and services.

14202 (U//FOUO) Federated Identity – While there are some commercial early adopters of Federated
14203 Identity systems, this technology is still immature. Creating federated identity systems require a
14204 great deal of trust, coordination, and development between two federated partners. The current
14205 commercial model will likely not meet the DoD’s requirements for security and dynamic
14206 administration. DoD-specific standards and guidelines need to be developed to support Federated
14207 Identity Management within the GIG.

14208 (U//FOUO) GIG-Specific Identity Management Schema – When implementing an identity
14209 management system, a schema describing users, their properties, and profiles must be created.
14210 This schema can vary dramatically from enterprise to enterprise. For the GIG, a schema should
14211 be developed that encompasses DoD-wide needs. Further, systems need to be designed to handle
14212 potential future schema modifications. Whatever identity management schema is developed in
14213 the near term will likely need revision after a few years of deployed use.

14214 **2.7.4.2 (U) Privilege Management**

14215 (U//FOUO) There is a standard that defines an Attribute Certificates to bind privileges to an
14216 Identity Certificate. This standard is an extension of the X.509 standard and has been adopted
14217 widely by PKI. Today, PMI works within the PKI infrastructure. Scalable alternatives to
14218 Attribute Certificates need to be explored.

14219 (U//FOUO) There are limitations in the capabilities currently provided by PMI.

14220 (U//FOUO) There are no standard mechanisms that specify how privileges are to be managed in
14221 a RAdAC Model that is required by the GIG.

14222 (U//FOUO) Another gap is the lack of technologies and standards that accommodate MLS
14223 classifications. This is likely a policy gap as well.

14224 (U//FOUO) Furthermore, while privileges for individuals are accounted for within existing PMI,
14225 standards and formats that address dynamically changing communities (COI) or Role-based
14226 privileges need to be developed and standardized across the GIG enterprise.

14227 (U//FOUO) Finally, policies on the trusted transportation and distribution need to be developed
14228 GIG-wide as well.

14229 2.7.4.3 (U) Key Management

14230 (U//FOUO) Automated solutions for managing the life cycle of keys do not currently exist.
14231 Human intervention is required in many aspects of key management, including registration,
14232 distribution, revocation, re-keying, and destruction. These human access points are vulnerable to
14233 threats and errors. To mitigate these vulnerabilities, there needs to be a strong drive towards
14234 standards for automation that provide and control the management of the life cycle keys. One
14235 such identified initiative that is driving requirements and standards is the KMI effort. The
14236 outcome of the KMI effort is expected to produce standards and policy.

14237 (U//FOUO) The identified gap areas within the individual aspects of key management include
14238 both policy gaps and technological gaps.

14239 (U//FOUO) One major gap area is the weakness or non-existence of tying policy controls
14240 (including dynamic policy changes) to various aspects of the key management cycle in an
14241 automated fashion. Standards need to exist so that automation can be built into promulgating
14242 dynamic policy changes into the necessary rules and regulations with which key registration,
14243 packaging, distribution, re-keying, revocation, and destruction work seamlessly and in an up-to-
14244 date, situational, manner.

14245 (U//FOUO) Another gap area is the lack of standards for unified key labeling, packaging, and
14246 distribution formats. The only area where some semblance of standards exist here are in the PKI
14247 (public, asymmetric keys) infrastructure. But none exists beyond PKI. Moreover, PKI has its
14248 own limitations with keys—such as re-keying—since PKI is certificate driven and not so much
14249 key-driven. In the Type-1 Classified arena, the key packaging and distribution processes are
14250 mainly manual processes. While they follow individual and situational-based policy, there are no
14251 standards to unify these in order to eliminate or reduce manual error-prone and human access
14252 vulnerabilities towards threats. Standards and technologies should include the incorporation of
14253 MLS systems and data stores to close these gaps.

14254 (U//FOUO) The management of symmetric keys needs to be included and evolved as well. For
14255 example, while there are individually controlled escrows and distribution of symmetric keys,
14256 there are no identified standards for the unified distribution of keys that would be required in the
14257 GIG-wide enterprise.

14258 2.7.4.4 (U) Certificate Management

14259 (U//FOUO) The only existing Certificate Management standard that exists today is found in the
14260 PKI arena. However, PKI has interoperability limitations at the application and component
14261 levels. There are no identified interoperability standards or technologies that specify the
14262 interfaces for certificate and data exchange between CAs. There are secure transports currently in
14263 use for certificates, but as such, there is no GIG-wide enterprise policy that governs what these
14264 access control restrictions should be.

14265 (U//FOUO) There are standards that are supposedly emerging for enhancing certificate attributes
14266 that aim to capture additional significant information such as subject privileges, trust anchor
14267 information and other necessary identity, trust, distribution, and access control information.
14268 There are also initiatives that are attempting to specify and collate various levels and
14269 classifications of certificates such as the Class 3, Class 4, and Class 5 Government and
14270 commercial type certificates. Until that happens, there is a gap here.

14271 (U//FOUO) There are standards and technology gaps in the manner that the GIG would require
14272 cryptographically binding, public keying material to information and attributes associated with a
14273 particular user/entity using a trust anchor in order to certify that the private key corresponding to
14274 the public key in the certificate is held by the same user/entity. There is a need to have binding
14275 strength increase with the strength of the cryptographic algorithm and key length used. No such
14276 standards or policy exist today.

14277 **2.7.4.5 (U) Configuration Management of IA Devices and Software**

14278 (U) Commercial products do not currently address a number of GIG requirements:

- 14279 • (U//FOUO) Although some configuration management tools authenticate the server, few
14280 use encrypted communications channels
- 14281 • (U//FOUO) Authentication of client machines is nonexistent
- 14282 • (U//FOUO) Only one identified product provides any support for modeling configuration
14283 changes before deployment
- 14284 • (U//FOUO) Although many products provided support for test deployments before a
14285 patch or upgrade deployment, none provide support for testing the configuration
- 14286 • (U//FOUO) No product provides support for authenticated or cryptographic verification
14287 of configurations, all assumed the device configuration information could be trusted
- 14288 • (U//FOUO) No product provides support for sensitive material distribution, such as keys,
14289 which require protection, receipts, and auditable tracking of delivery
- 14290 • (U//FOUO) No product supports remote update of firmware.
- 14291 • (U//FOUO) No general standard exists for communications between a configuration
14292 manager and its agent, or the target machines, although Microsoft has implemented the
14293 WBEM standard for its operating systems.

14294 **2.7.4.6 (U) Inventory Management**

14295 (U) The technology gap for Inventory Management lies in the area of RFID technology.
 14296 Although the technology has been available since 1974, standardized, interoperable tags and
 14297 readers are a recent innovation. Although the field is rapidly developing on its own, the focus is
 14298 on developing low-cost, passive UHF RFID tags that reach the critical \$.05 per unit goal that can
 14299 be used for consumer supply chain applications. Consumer privacy issues have raised the
 14300 concern that RFID tags are still active after leaving the retail sales point and could be used to
 14301 track individuals, so some work is being done to develop RFID tags that can be killed—rendered
 14302 permanently inert or unresponsive to a reader interrogation, or rendered temporarily inert.
 14303 However, no apparent work is being done to develop secure RFID tags which would respond
 14304 only to interrogation and commands from an authenticated reader. Even DES encryption is
 14305 considered significantly more expensive than can be handled by an RFID tag.

14306 (U) The greater gap for Inventory Management is that it requires development of an Inventory
 14307 Management infrastructure that tracks and manages.

14308 **2.7.4.7 (U) Compromise Management of IA Devices**

14309 (U) Tamper detection mechanisms are well understood, although tamper resistant mechanisms
 14310 such as seals can always be defeated. However, current systems limit their tamper response to
 14311 zeroizing their internal data and do not include the concept of network-aware reporting of
 14312 alerts—secure or otherwise—as part of their tamper processing.

14313 (U) External compromise monitoring mechanisms such as an IAC status and monitoring protocol
 14314 or IAC Keep Alive protocol do not currently exist, and must be developed. It could become part
 14315 of a SNMP Management Information Base or part of an IA device management protocol. A
 14316 secure device management protocol is a requirement brought by secure configuration
 14317 management requirements.

14318 **2.7.4.8 (U) Audit Management**

14319 (U//FOUO) Audit management exists today in a very non-standard manner. There are a
 14320 multitude of SEM vendors that provide some type of audit capability built into their proprietary
 14321 solutions. Without standards in technology, interoperability (within components, log formats,
 14322 audit analyses, etc.), and policies, there is a big gaping hole in the unified audit management
 14323 scheme that the GIG enterprise requires.

14324 (U) Standards in technology, interfaces, interoperability, and policies need to be developed and
 14325 defined in the areas of:

- 14326 • (U//FOUO) Log and event formats – to capture and record normalized GIG-wide
 14327 activities and system performance
- 14328 • (U//FOUO) Standardized Securing of audit data into one-way (diode) stores
- 14329 • (U//FOUO) Standardizing Agents and Agentless components for interoperability and
 14330 security (assurance)
- 14331 • (U//FOUO) Standards for tools that monitor system resources

- 14332 • (U//FOUO) Adhering to the DoDI 8500.2 standards for audit data record capture. This
14333 includes provisioning for attributes, such as the ECAR-1, 2 and 3, that correspond to the
14334 various classification levels
- 14335 • (U//FOUO) Central monitoring and interfacing standards, from a NOC or SOC
- 14336 • (U//FOUO) Standards for correlation, analysis and alerting services that subscribe to
14337 audit data publishing
- 14338 • (U//FOUO) Secure transport standards.

14339 **2.7.5 (U) Management of IA Mechanisms and Assets: Recommendations and Timelines**

14340 (U) The management of network assets itself is relatively mature, however, this is true only for
14341 low-threat environments. In a medium to high-threat environment, a significant gap exists. For
14342 the high-assurance management of cryptographic components, there are only limited proprietary
14343 solutions. No solutions exist which provide configuration management of high assurance IA
14344 devices.

14345 **2.7.5.1 (U) Standards**

14346 (U) Standards that need to be developed to support the management of GIG Assets and
14347 Mechanisms include:

- 14348 • (U//FOUO) Standard for maintenance, communication, and management of existing
14349 identities across federated authorities
- 14350 • (U//FOUO) DoD-specific standards and protection profiles for federated identity
14351 management, including a DoD-wide identity management schema
- 14352 • (U) Secure device identification standards, which use cryptographic authentication of the
14353 identity of a device.
- 14354 • (U) Standards for dynamic establishing and disestablishing COIs and COI membership
- 14355 • (U) Standards for role-based privilege management across federated organizations
- 14356 • (U) Standards for wholly automated life cycle for key material
- 14357 • (U) Standards for key labeling, packaging, and distribution, particularly symmetric keys
- 14358 • (U) Standards for interoperability among certificate management infrastructure
14359 components
- 14360 • (U//FOUO) Standard protocols for the secure management of IA-enabled devices,
14361 including initialization, software load, configuration, verification of a configuration, and
14362 update
- 14363 • (U//FOUO) A standard for secure boot and remote initialization of a device, including
14364 device authentication; especially a cryptographic device across a black network

- 14365 • (U) Standards for secure remote data delivery including receipting
- 14366 • (U) Secure RFID standards
- 14367 • (U//FOUO) Standard IAC keep-alive protocol
- 14368 • (U//FOUO) Standard compromise notification protocol, particularly in the case of
- 14369 notification across a black network
- 14370 • (U) Widely adopted audit log standard
- 14371 • (U) Audit aggregation and analysis data standard.

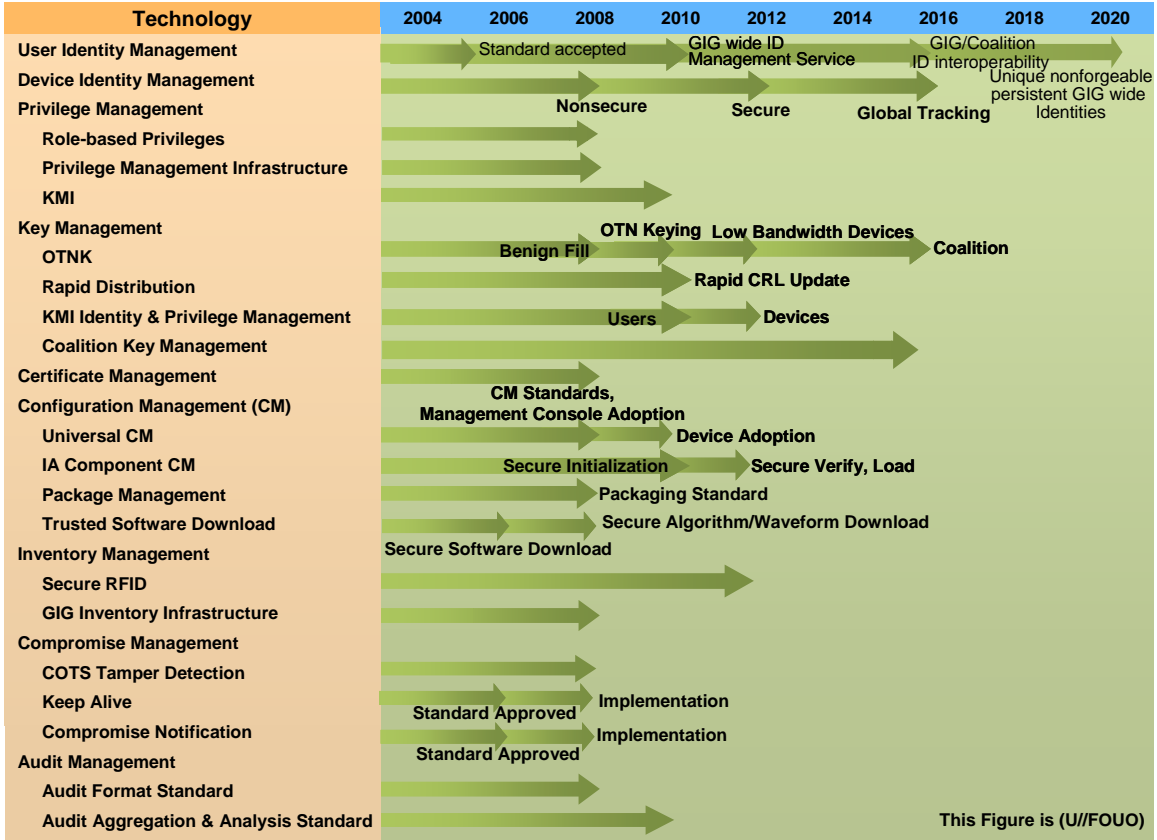
14372 **2.7.5.2 (U) Technology**

- 14373 • (U) Secure, authenticated network boot devices
- 14374 • (U) Secure RFID, including authentication of the reader to the RFID tag
- 14375 • (U) Tamper detection and network manager notification
- 14376 • (U//FOUO) Multi-level PKI certificate authorities for a single identity and certificate
- 14377 across the GIG.

14378 **2.7.5.3 (U) Infrastructure**

- 14379 • (U) Device identification and tracking
- 14380 • (U) IA device inventory and configuration management
- 14381 • (U) Key Management Infrastructure.

14382



14383

14384

14385

Figure 2.7-16: (U) Technology Timeline for Assured Resource Allocation

14386 **3 (U) SUMMARY**

14387 (U//FOUO) The Global Information Grid (GIG) Information Assurance (IA)
14388 Capability/Technology Roadmap compares the commercial and Government technology trends
14389 and technology forecasts available today against the needed capabilities defined in the Transition
14390 Strategy in the GIG IA Reference Capability Document (RCD). The results of these analyses
14391 include descriptions of interdependencies between needed capabilities, technology timelines, and
14392 gaps between capability needs and technology availability. These results, together with other
14393 background information and analysis in this document, are intended to provide decision makers
14394 with the information needed to revise or write new standards and policies, develop
14395 implementation guidelines, make research funding decisions, devise strategies for needed
14396 technology development, and develop technology implementation plans.

14397 (U//FOUO) This section summarizes the most significant impressions and conclusions arising
14398 from the investigations and analyses of the candidate IA technologies. Results are organized
14399 around the four IA cornerstones defined in the GIG IA RCD and presented in the context of the
14400 Transition Strategy. The four IA cornerstones are:

- 14401 • (U) Assured Information Sharing
- 14402 • (U) Highly Available Enterprise
- 14403 • (U) Assured Enterprise Management and Control
- 14404 • (U) Cyber Situational Awareness and Network Defense

14405 (U) Some of the technologies support more than one cornerstone. Therefore, results for any
14406 particular technology may appear to be duplicated in two or more cornerstones. However, there
14407 are generally slight differences in the gaps and recommendations, reflecting the different aspects
14408 of the cornerstone that the technology supports.

14409 (U//FOUO) For each IA cornerstone, a summarizing timeline is shown that illustrates the
14410 primary technology categories described in the Transition Strategy and needed to meet 2008 GIG
14411 IA capabilities. Gaps and recommendations are then described for the technology areas and
14412 component technologies, where appropriate. In the timelines, milestones for specific imperatives
14413 are shown as colored diamonds, where:

- 14414 • (U) Green indicates that the milestone will be achieved under current development plans,
14415 schedules, and funding of the component technologies supporting that milestone
- 14416 • (U) Yellow indicates that the milestone will not be achieved if development of the
14417 supporting technologies proceeds as planned—but the milestone could be achieved by
14418 accelerating current development efforts or starting new development efforts
- 14419 • (U) Red indicates that the milestone cannot be achieved as currently defined by the
14420 Transition Strategy

UNCLASSIFIED//FOR OFFICIAL USE ONLY

14421 (U) The milestone color-coding is largely based on isolated examinations of the supporting
14422 component technologies. In practice, some technology development efforts will be
14423 interdependent. For example, one technology development effort may be delayed because it must
14424 rely on an intermediate result from another technology development. Such interdependencies
14425 were not fully considered, so some of the technology development timelines and the color-
14426 coding of the affected milestones may be slightly optimistic. Further investigation will be needed
14427 to refine these timeline estimates.

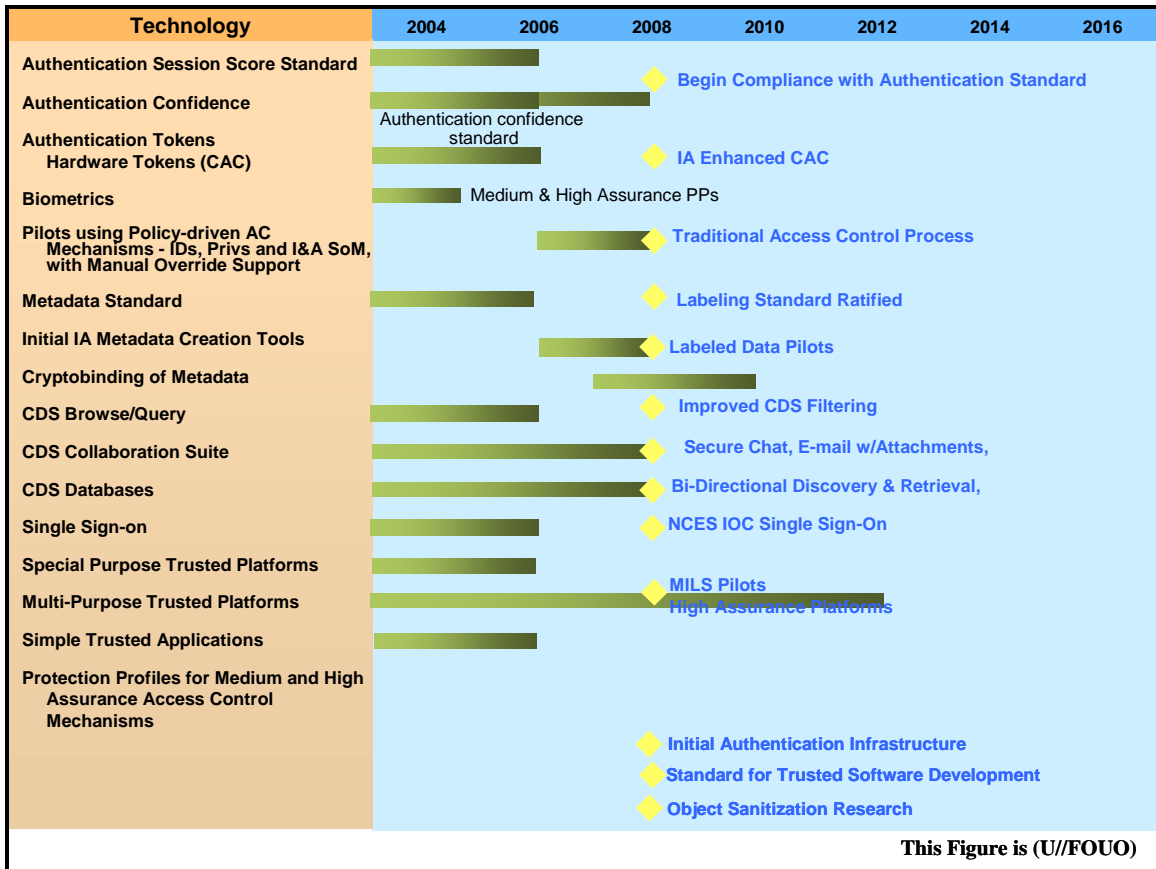
14428 (U) With only minor exceptions, the gaps and recommendations are described for technologies
14429 needed to meet the 2008 GIG IA objectives as described in the Transition Strategy. The
14430 description is further limited to technologies that are deemed risky, either because no work is
14431 currently going on, or because ongoing development effort will probably not be completed in
14432 time to deploy for 2008. In some cases, gaps and recommendations are summarized for
14433 technologies needed for 2012 and beyond, but only in cases where technology development
14434 efforts must begin now in order to meet those technology milestone dates.

14435 (U) These results give a fairly complete picture. Subsequent effort on this document will focus
14436 on updating and refining the status of the technologies.

14437 **3.1 (U//FOUO) ASSURED INFORMATION SHARING SUMMARY**

14438 (U//FOUO) Technologies supporting this cornerstone are organized into five general categories:
 14439 Identification, Authentication, Access Control, Data Labeling, and Cross-Domain Security.

14440 (U) Figure 3.1-1 provides an overview of the technologies and how they support the IA
 14441 imperatives listed in the Transition Strategy. As shown, while none of the technologies will be
 14442 completed in time to meet the 2008 IA imperatives, the milestones are achievable if current
 14443 efforts are accelerated. Some imperatives have no supporting technologies identified in this
 14444 release of the document. These are discussed in the gaps below.



14446 **Figure 3.1-1: (U//FOUO) Technology Timeline for Assured Information Sharing**

14447 **3.1.1 (U) Identification and Authentication Technologies**

14448 (U//FOUO) As the technology development efforts currently stand, none of the I&A-related
 14449 milestones shown in Figure 3.1-1 will be met. Gaps that will prevent deploying an initial
 14450 authentication infrastructure that conforms to a common authentication standard are listed
 14451 below—along with recommended corrective actions.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 14452 • (U//FOUO) Gap: An authentication framework standard does not yet exist. Such a
14453 standard (or set of standards) must address SoM levels, authentication session scoring, a
14454 SoM forwarding structure, and authentication confidence metrics. Until such standards
14455 exist, a global authentication infrastructure and associated technologies cannot be
14456 deployed.
 - 14457 (U//FOUO) Recommendations: 1) Develop a common GIG-wide device/service
14458 authentication techniques and standards.
 - 14459 2) (U//FOUO) Rapidly advance research into the relatively new area of authentication
14460 confidence metrics.
- 14461 • (U//FOUO) Gap: Protection Profiles are needed for Medium and High Assurance
14462 authentication technologies, including biometrics technologies.
 - 14463 (U//FOUO) Recommendations: Develop protection profiles to facilitate
14464 authentication standards and architecture development.
- 14465 • (U//FOUO) Gap: A common GIG-wide Single Sign-On (SSO) mechanism, protocol, and
14466 architecture have not yet been selected.
 - 14467 (U//FOUO) Recommendation: Study and select a GIG-wide architecture for SSO
14468 using the candidate approaches described in Section 2.1. Include in this study a
14469 complete analysis of the proposed NCES SSO architecture.
- 14470 • (U//FOUO) Gap: A scalable authentication server that is able to interpret and use I&A
14471 session scores and comply with the GIG authentication standards does not exist.
 - 14472 (U//FOUO) Recommendation: Begin development of a scalable, robust, and
14473 distributed authentication server capability whose components can operate in multiple
14474 architectural constructs (e.g., in-line, embedded, coprocessor, remote).
- 14475 (U//FOUO) In addition to the technologies listed above, other gaps have been identified that will
14476 prevent meeting 2012 (and later) imperatives. Those listed below require that recommendations
14477 be acted on soon in order to ensure sufficient development time to meet the affected milestones.
- 14478 • (U//FOUO) Gap: A high assurance DoD PKI Class 5 token with Type I cryptography will
14479 eventually be needed. Development of the DoD CAC is proceeding in the needed
14480 direction, but it is not yet available. A Class 5 token will be needed for assured access to
14481 classified information. Such a token will use Type I cryptography, and its security-critical
14482 functionality will be assured throughout its life cycle, including design, development,
14483 production, fielding, and maintenance.
 - 14484 (U//FOUO) Recommendation: Monitor ongoing and future developments of the DoD
14485 CAC to ensure support of all future GIG requirements (including the Class 5 token).
- 14486 • (U//FOUO) Gap: Common standards for Partner Identity Proofing and a common
14487 Identification Registration/Management Infrastructure will be needed to ensure identity
14488 interoperability among all current and future GIG partners (e.g., DoD, IC, civil
14489 Government, Department of Homeland Security (DHS), allies, coalition partners).

UNCLASSIFIED//FOR OFFICIAL USE ONLY

14490 (U//FOUO) Recommendation: Begin formation of future partner community; then
14491 start development of a common Partner Identity Proofing standard.

14492 **3.1.2 (U) Access Control and Data Labeling Technologies**

14493 (U//FOUO) The Risk Adaptable Access Control (RAdAC) functions central to GIG access
14494 control are in their infancy with respect to concept formulation, standards development, policy
14495 implications, and technology implementation. While industry has shown interest in role-based
14496 access control, and now attribute-based access control, the unique features of RAdAC require
14497 additional technologies.

14498 (U//FOUO) Moreover, industry is not likely to sponsor the needed research and development in
14499 this area, since no commercial market is anticipated for such a capability. Therefore, there are
14500 numerous technology gaps that the Government will need to address. Only the first gap listed
14501 below is called out in the 2008 Transition Strategy imperatives. The remainder can and should be
14502 closed by 2008 in order to meet the imperatives of subsequent increments.

- 14503 • (U//FOUO) Gap: Protection Profiles. There are no current or planned protection profiles
14504 that address RAdAC or attribute-based access control. These protection profiles are
14505 necessary to establish the minimum security protections required for any implementation
14506 of RAdAC.

14507 (U//FOUO) Recommendation: Develop Attribute-Based Access Control (ABAC) and
14508 RAdAC Protection Profiles.

- 14509 • (U//FOUO) Gap: RAdAC standard. Since industry is not moving in the RAdAC
14510 direction, there are no formal representations of architecture, interface definitions,
14511 performance requirements, or protocol requirements.

14512 (U//FOUO) Recommendations: Develop a RAdAC standard. Also, begin RAdAC
14513 prototyping to support standards development. This activity will also be valuable for
14514 other related RAdAC development activities, including requirements discovery, input
14515 ontology development, Digital Access Control Policy (DACP) standard development,
14516 and Digital Rights integration specification development.

- 14517 • (U//FOUO) Gap: ABAC standard. Given the current immaturity and criticality of
14518 RAdAC, it would be prudent to have an alternative to RAdAC. ABAC should be
14519 considered as an interim solution while RAdAC is being developed. However, even
14520 though there is research and even commercial ABAC-based products, there are no
14521 commercial or government standards.

14522 (U) Recommendation: The Government should initiate development of a commercial
14523 or government ABAC standard.

- 14524 • (U//FOUO) Gap: RAdAC mathematical model: An underlying mathematical model is
14525 needed to meet Medium and High assurance implementation requirements and to assist in
14526 the transformation from a Discretionary Access Control (DAC) and Mandatory Access
14527 Control (MAC) access control culture. This model needs to include the digital access
14528 control policy since the two are so tightly integrated.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

14529
14530
14531
14532
14533
14534
14535
14536
14537
14538
14539
14540
14541
14542
14543
14544
14545
14546
14547
14548
14549
14550
14551
14552
14553
14554
14555
14556
14557
14558
14559
14560
14561
14562
14563
14564
14565
14566
14567

(U//FOUO) Recommendation: Develop RAdAC mathematical model.

- (U//FOUO) Gap: Input parameter ontology. All attributes that feed the RAdAC model need to have an ontology that is accessible and standardized. This applies to attributes of IT Components, Environment, Situation, Soft Objects (metadata), and people.

(U//FOUO) Recommendation: Develop input parameter ontology.

(U//FOUO) There is at least one ontology standard and language that meet some of the basic requirements for DACP. However, significant work is needed to realize a complete implementation that will meet GIG information-sharing requirements.

- (U) Gap: DACP standard. Based on the underlying math model, a DACP standard that uses ontology and deontic languages needs to be developed. This standard will address the access control policy grammar, exception handling, business rules about allowable and disallowable policy constructs, and business rules for policy negotiation and de-confliction.

(U) Recommendation: Develop DACP standard with associated business rules.

- (U) Gap: Digital Rights Management integration specification. Digital Rights can be viewed as a static projection of digital access control policy onto a particular soft object. There is currently ongoing research in the Digital Rights realm and proposed standards, but none of this work is aimed at specifying a relationship between digital rights and digital access control policy. An analysis of these relationships, digital rights implementation, and Policy Enforcement Point interface is necessary to complete the end-to-end access control of GIG information and support the transition to a need-to-share culture.

(U) Recommendations: 1) Develop Digital Rights integration specification

(U) 2) Work with commercial standards groups to integrate needed aspects into the appropriate commercial standards.

(U//FOUO) The RAdAC core technologies present the most technical risk for access control, but gaps in metadata technologies are also of concern because of the centrality of metadata to assured information sharing. These gaps can and should be closed by 2008 in order to meet the imperatives of subsequent increments.

(U//FOUO) Each data object will be associated with a Quality of Protection (QoP) that specifies how that object is to be protected while at rest, and how it is to be protected throughout its lifetime. This impacts the technology employed and design of virtually every entity in the GIG that handles data.

- (U//FOUO) Gap: The definition, implementation, and enforcement of QoP at the data object level.

(U//FOUO) Recommendations: A QoP standard must first be developed that defines the privileges that can be assigned to each data object. Analytical and modeling-based studies will be needed to develop appropriate policies, standards, and specifications for all affected entities.

14568 • (U//FOUO) Gap: Standards. Both the Intelligence Community (IC) and Department of
14569 Defense (DoD) are developing metadata standards, and they are coordinating their work
14570 to ensure that IA attributes associated with RAdAC style access control decision-making
14571 and discovery are addressed in these standards. However, standards development
14572 activities must be closely coordinated with ongoing research and development efforts, in
14573 order to avoid incompatibilities in technology standards that would eventually require
14574 changes to supporting tools, infrastructure, and large quantities of existing metadata
14575 records.

14576 (U//FOUO) Recommendations: 1) The GIG community should work with IC and
14577 Core Enterprise Service (CES) Metadata working groups to ensure IA RAdAC
14578 required attributes are adequately addressed, and to either guide the integration of IA
14579 attributes into the metadata standards according to detailed analysis, or (preferred)
14580 support the merger of these standards.

14581 2) (U//FOUO) Before stabilizing the metadata standards and IA attributes, conduct
14582 further studies to examine the impact of metadata on network traffic/overhead
14583 (especially for real-time and session object types) and potential for trading metadata
14584 IA granularity with transmission overhead.

14585 • (U//FOUO) Gap: Metadata creation tools. Commercial metadata creation tools are
14586 available. However, they do not have the needed GIG IA-related capabilities and
14587 interfaces, which are new, complex, and unique to the GIG.

14588 (U//FOUO) Recommendation: Begin now early design of metadata creation tools in
14589 parallel with the metadata standards definition to ensure IA specific attributes,
14590 cryptographic binding of metadata and the source object, and authorization interface
14591 needs are addressed.

14592 3.1.3 (U) Cross-Domain Technologies

14593 (U//FOUO) Despite the large number and variety of Cross Domain Solution (CDS) development
14594 efforts underway for many years and moderate number of accredited products available,
14595 significant work remains in order to meet the CDS-related 2008 GIG IA requirements of the
14596 Transition Strategy.

14597 • (U//FOUO) Gap: Cross-domain file transfer. Although accredited solutions exist to
14598 transfer fixed-file formats, there are many files prohibited from being passed through
14599 these solutions. Most notably these include executable files and documents with
14600 macros—Microsoft Office files in particular.

14601 (U//FOUO) Recommendations: 1) Research and develop advanced capabilities for
14602 safely transferring files across security domains, initially targeting the examination of
14603 files generated by Microsoft Office and other common warfighter applications for
14604 executable and hidden malicious content.

14605 2) (U//FOUO) Develop clear and consistent policies for dealing with discovered
14606 malicious content, such as automatic deletion of content, imposition of execution
14607 constraints, manual security review, etc.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 14608 3) (U//FOUO) Develop mechanisms to execute the malicious content discovery
14609 policy. 4) Investigate alternatives to commonly-used products known to contain
14610 security weaknesses in this area.
- 14611 • (U//FOUO) Gap: Trusted workstations, needed to push multiple domain access out to
14612 users in the field and support warfighter applications in the operational environments, are
14613 not available.
14614 (U//FOUO) Recommendations: Accelerate research to develop trusted CDS platforms
14615 that are:
14616 1) (U//FOUO) certified to allow users who are not cleared for the highest levels of
14617 information on the workstation to use the platform at the level for which they are
14618 cleared;
14619 2) (U//FOUO) allow warfighters to use applications to which they are accustomed,
14620 e.g., for word processing, collaboration, situational awareness, and planning;
14621 3) (U//FOUO) can function under the resource constraints of the warfighters (e.g.,
14622 space, weight, and power constraints of infantry) while supporting critical
14623 functionalities (e.g., combat ID, secure voice).
 - 14624 • (U//FOUO) Gap: Information protection technologies (e.g., High Assurance Internet
14625 Protocol Encryptor [HAIPE]) supporting the GIG Black Core concept are currently single
14626 security domain devices and prevent traditional CDS from examining information flow
14627 content.
14628 (U//FOUO) Recommendation: Enhance functionality of data protection technologies
14629 to support information flows between security domains. Tighter integration between
14630 the content review and filtration system (e.g., the high assurance guard), and the
14631 protection system (e.g., the HAIPE) is required.
 - 14632 • (U//FOUO) Gap: Current cross-domain solutions are designed to examine static blocks of
14633 information containing entire message sets (e.g., files, email), and no ability currently
14634 exists to support critical real-time information flows (e.g., secure voice, video
14635 teleconferencing).
14636 (U//FOUO) Recommendation: Efforts for developing technologies to support cross-
14637 domain real-time flows—such as voice communications and collaboration among
14638 coalition partners—should begin immediately.
 - 14639 • (U//FOUO) Gap: Current maturity of IA controls has resulted in cross-domain solutions
14640 with strict management and configuration properties that do not facilitate flexible
14641 management, configuration, and adaptation of the CDS to insure proper operation in a
14642 changing environment (e.g., INFOCON transitions, dynamic multinational agreements,
14643 etc.).
14644 (U//FOUO) Recommendation: Develop standards, techniques, and procedures that
14645 can be certified to insure that CDS initialization, management, configuration and
14646 support shall not be impaired by use in remote warfighting environments among Joint
14647 and Multinational participants with dynamic agreements.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 14648
- 14649
- 14650
- 14651
- (U//FOUO) Gap: Insufficient training and inadequate deployment of a Joint cross-domain solution leads to ineffective use of existing Service-owned CDS capabilities, restricts the flow of vital information, and complicates the correlation of information from multiple security domains.

14652

14653

14654

14655

14656

(U//FOUO) Recommendation: Develop standards for cross-domain technologies that are based on current Joint and Multinational operational doctrine and practices. These standards apply to the entire lifecycle of CDS technologies and include the development of common, Joint CDS capabilities, adequate deployment of Joint solutions; and sufficient training for the warfighters who will use these solutions.

14657

3.1.4 (U) Trusted Platform Technologies

14658

14659

14660

14661

14662

14663

14664

(U//FOUO) Trusted platforms have been around for more than 20 years in one form or another. For special purpose IA components, such as firewalls and gateways, the technologies are mature and will meet the 2008 GIG IA imperatives. For workstations and other devices that must connect to multiple security domains, significant research and development in the areas of software engineering, high-assurance computing, network security, and system evaluation will be required before needed GIG IA capabilities can be met. The primary gap and the action needed to meet 2008 GIG IA imperatives are:

- 14665
- 14666
- 14667
- 14668
- 14669
- 14670
- 14671
- 14672
- 14673
- (U//FOUO) Gap: Software development for trusted applications. No universally-accepted methodologies—much less standards—have been devised for development of software to be used in applications requiring high assurance. This problem has been recognized, and Office Secretary of Defense (OSD) Networks and Information Integration (NII) and DHS are co-sponsoring an effort to investigate the problem of high-assurance software, with the goal of establishing partnerships between Government, academia, and industry to develop solutions that span the software development process, evaluation, and training. However, it is not clear if the current efforts will result in the publishing of standards for trusted software development by 2008.

14674

14675

14676

(U//FOUO) Recommendation: Given the importance of high-assurance software to GIG components, DoD should accelerate its current study efforts and focus on devising trusted-software development processes and standards.

- 14677
- 14678
- 14679
- 14680
- 14681
- 14682
- (U//FOUO) Gap: A linkage between a security policy enforced by the trusted application and the security policy enforced by the host platform needs to be developed. This is the composition problem that has been researched off and on with unsatisfactory results for at least 20 years. A side issue to be examined is what happens when the trusted application is implemented on a variety of host platforms, and those platforms must communicate and interoperate.

14683

14684

(U//FOUO) Recommendation: Conduct research aimed at coordinating application security policy and hardware security policy.

- 14685
- 14686
- 14687
- (U//FOUO) Gap: Construction of self-protecting applications that can guard themselves against attacks coming through the host platform, such as against attacks using disk storage or input devices.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

14688 (U//FOUO) Recommendation: Conduct research into trusted applications that can
14689 guard themselves against attacks coming through the host platform (hardware or
14690 software).

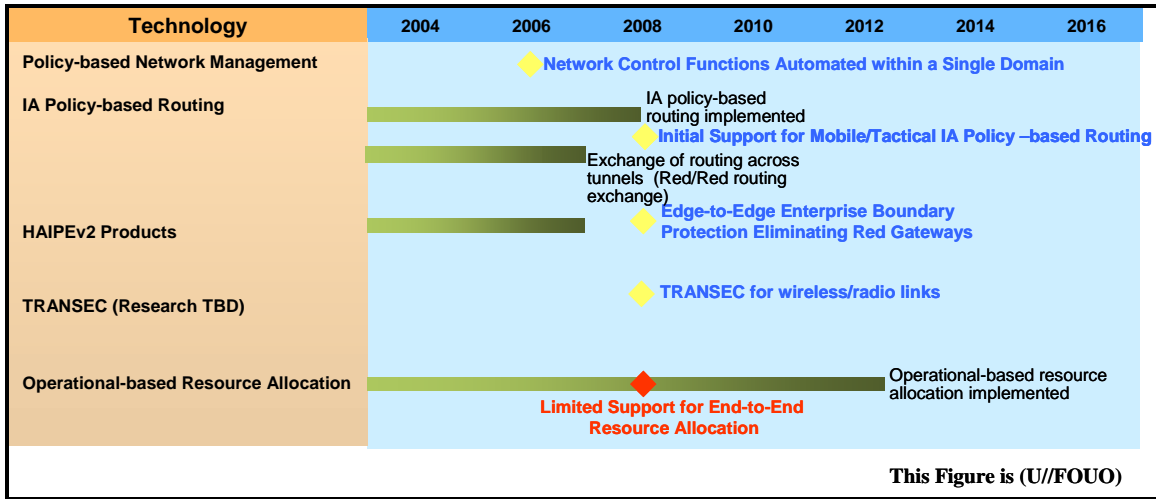
- 14691 • (U//FOUO) Gap: Support for complex security policies within trusted platforms, such as
14692 dynamic access control policies like RAdAC.

14693 (U//FOUO) Recommendation: Conduct research aimed at defining and enforcing
14694 complex security policies with trusted platforms. Include research into developing
14695 and enforcing RAdAC policies.

14696 **3.2 (U) HIGHLY AVAILABLE ENTERPRISE SUMMARY**

14697 (U//FOUO) Technologies supporting this cornerstone are organized into five general categories:
 14698 IA Policy-based Routing, End-to-end Resource Allocation, Edge-to-Edge Boundary Protection
 14699 in the Black Core, Secure Voice, and Quality of Protection.

14700 (U) Figure 3.2-1 provides an overview of the technologies and how they support the IA
 14701 imperatives listed in the Transition Strategy. As shown, while none of the technologies will be
 14702 completed in time to meet the 2008 IA imperatives, if current efforts are accelerated, the
 14703 milestones are achievable.



14704

14705 **Figure 3.2-1: (U//FOUO) Technology Timeline for Highly Available Enterprise**

14706 **3.2.1 (U//FOUO) IA Policy-based Routing for Mobile/Tactical Environments**
 14707 **Technologies**

14708 (U//FOUO) As described in Section 2.5, all routing is policy-based, but a policy usually enforces
 14709 the shortest path or least cost rather than considering the IA properties of the links. The extension
 14710 of routing protocol algorithms to include the aspect or metric of path assurance/security is
 14711 relatively recent and thus not nearly as mature. Some work in this area has been done in the area
 14712 of IA policy-based routing for mobile ad hoc networks, due to the obvious potential
 14713 vulnerabilities of wireless networks as compared with more secure wired network infrastructures.
 14714 Research for IA policy-based routing needs to be continued.

- 14715 • (U//FOUO) Gap: Lack of IA metrics in wired and wireless routing protocols.

14716 (U//FOUO) Recommendations: Further research and development of adaptive
 14717 security-driven (i.e., IA policy-based), wireless routing algorithms is required for
 14718 inclusion in mobile and tactical programs (e.g., Joint Tactical Radio System [JTRS]
 14719 and Warfighter Information Network-Tactical [WIN-T]). Research should be
 14720 extended into the wired domain so that IA policy-based routing can benefit all
 14721 networks (wired or wireless). Findings must be used to advance the standards
 14722 evolution and demonstration/implementation of extensible routing protocols (such as

14723 Open Shortest Path First [OSPF] and Intermediate System-to-Intermediate System
14724 [IS-IS]) so that IA metrics can be fully employed in routing decisions.

14725 **3.2.2 (U) End-to-End Resource Allocation Technologies**

14726 (U//FOUO) Resource allocation traditionally has been limited to the scope of small geographic
14727 areas as opposed to the world-wide reach of the GIG. The GIG must be able to control and
14728 modify the amount of resources (e.g., bandwidth, processor cycles) allocated to any given user,
14729 based on current operational requirements. For example, the GIG should be able to cut back on
14730 the amount of resources available to sustain operations on portions of the network in order to
14731 increase the resources available to a unit currently engaged in battle.

- 14732 • (U//FOUO) Gap: At this point, there is insufficient research—much less technology—to
14733 support all of the GIG requirements for dynamic resource allocation. Dynamic
14734 reconfiguration of resources is a difficult problem that has only some limited solutions
14735 available now.

14736 (U//FOUO) Recommendation: Initiate research into permitting dynamic
14737 reconfiguration within a Black Core where all traffic is encrypted while at the same
14738 time defending against attacks as needed in the GIG (e.g., ensuring that a requested
14739 change in resources comes from an authorized entity and is legitimate and appropriate
14740 given the current operational situation).

14741 (U//FOUO) Part of resource allocation involves the deployment of Quality of Service (QoS)
14742 mechanisms across the GIG. While there has been a significant amount of work done by
14743 commercial industries related to QoS, implementing and enforcing QoS mechanisms has proven
14744 difficult. Commercial products are evolving to support QoS, and the GIG must keep abreast of
14745 new developments and integrate them where appropriate.

- 14746 • (U//FOUO) Gap: An area of QoS that has not being given much attention by commercial
14747 industry is security mechanisms. QoS parameters need to be applied to packets and flows
14748 across the GIG by devices that do not abuse the features of QoS to use more than their
14749 share of resources or create Denial of Service conditions.

14750 (U//FOUO) Recommendation: Define the IA aspects of QoS and socialize them
14751 across the GIG community. Define procedures and mechanisms for end-to-end QoS
14752 and resource allocation across crypto boundaries. Define security mechanisms and
14753 solution for supporting end-to-end QoS in the GIG. Solutions need to be developed to
14754 support the end-to-end QoS GIG requirements.

14755 (U//FOUO) QoS solutions are currently being deployed within the GIG. Although the Transition
14756 Strategy does not specify end-to-end QoS enforcement until 2012, research and development
14757 must continue in order to mitigate the risk of non-interoperable QoS islands within the GIG.

- 14758 • (U//FOUO) Gap: An additional capability related to resource allocation that is not being
14759 considered by commercial industry is precedence and preemption in the Black Core. The
14760 GIG has requirements (particularly with regards to voice) to assign priority (different
14761 from QoS) to packets, and in times of congestion, higher priority packets can preempt
14762 lower priority packets.

14763 (U//FOUO) Recommendation: Development of a GIG Precedence and Preemption
14764 standard to provide the capability for rational post-preemption rescheduling should
14765 continue so as to not leave GIG customers without requested services.

14766 **3.2.3 (U//FOUO) Edge-to-Edge Boundary Protection Technologies**

14767 (U//FOUO) GIG programs need to provide boundary protections without the use of red
14768 gateways. Within the Black Core, traffic will be encrypted at the boundary of the originating
14769 network and remain encrypted across the GIG transport programs until it is decrypted at the
14770 ingress to the recipient's network.

14771 (U//FOUO) Gap: Traditional firewalling, content filtering, intrusion detection, and other IA
14772 capabilities will not function in the Black Core as they need to do today. The GIG community
14773 still has a need for these IA capabilities in the Black Core.

14774 (U//FOUO) Recommendations: Resolving these issues will require research and
14775 testing as well as significant community socialization to ensure that solutions are
14776 consistently applied across the GIG and end-to-end services can be supported.
14777 Specifically the following areas need to be addressed:

- 14778 1. (U//FOUO) Evolution of the HAIPE protocol is required to support dynamic
14779 routing in a multi-homed environment, red-to-red routing exchanges, QoS,
14780 dynamic black IP addresses, mobility, end-system implementations, resource-
14781 constrained implementations, and low-bandwidth, high bit error rate environments
- 14782 2. (U//FOUO) Research is necessary to enable filtering on source, destination, and
14783 payload in the Black Core in order to monitor for unauthorized traffic before it
14784 crosses a GIG network
- 14785 3. (U//FOUO) Research is necessary to provide admission control and priority
14786 handling of encrypted packets
- 14787 4. (U//FOUO) Research is necessary to develop effective intrusion detection
14788 capabilities on encrypted segments.

14789 **3.2.4 (U) Secure Voice Technologies**

14790 (U//FOUO) Based on the 2008 Transition Strategy, Voice over IP (VoIP) solutions will be
14791 deployed within system high networks. While this is achievable with today's technology using a
14792 single vendor's solution, much work is required to move towards interoperable secure voice over
14793 secure IP solutions required by the GIG 2020 Vision.

- 14794 • (U//FOUO) Gap: Lack of interoperable secure voice over secure IP solutions.

14795 (U//FOUO) Recommendations: Activities that must be started to achieve the GIG
14796 2020 Vision related to voice include:

- 14797 1) (U) Standards for providing interoperability between Secure Voice over IP systems
14798 and Voice over Secure IP systems
- 14799 2) (U//FOUO) Standards defining a common interoperable implementation of Future
14800 Narrow Band Digital Terminal (FNBDT) over IP networks, including call control,
14801 gateway operation, and user media details

14802 3) (U//FOUO) Standards defining FNBDT multipoint operation (conferencing, net
14803 broadcast, and multicast applications)

14804 4) (U//FOUO) Standards defining additional voice coders for FNBDT systems on
14805 specific GIG sub-networks

14806 5) (U//FOUO) Interoperability between secure voice products in circuit switched
14807 networks and secure voice products in packet switched networks.

14808 **3.2.5 (U) Enforcement of QoP in Transit Technologies**

14809 (U//FOUO) Each data object will be associated with a QoP that specifies how that object is to be
14810 protected and routed across the GIG. This impacts the technology employed and design of
14811 virtually every entity in the GIG that handles data.

- 14812 • (U//FOUO) Gap: Devices must be able to understand and enforce the QoP for a data
14813 object while it is in transit.

14814 (U//FOUO) Recommendations: Enforcement mechanisms must be designed into GIG
14815 components that can recognize the QoP parameters and provide the appropriate
14816 enforcements. Research must be started immediately to lead to the development of
14817 automated solutions, end-to-end QoP enforcement, and standardization of those
14818 solutions, to support the GIG 2020 Vision.

14819 **3.2.6 (U//FOUO) Protection of High Risk Link Technologies**

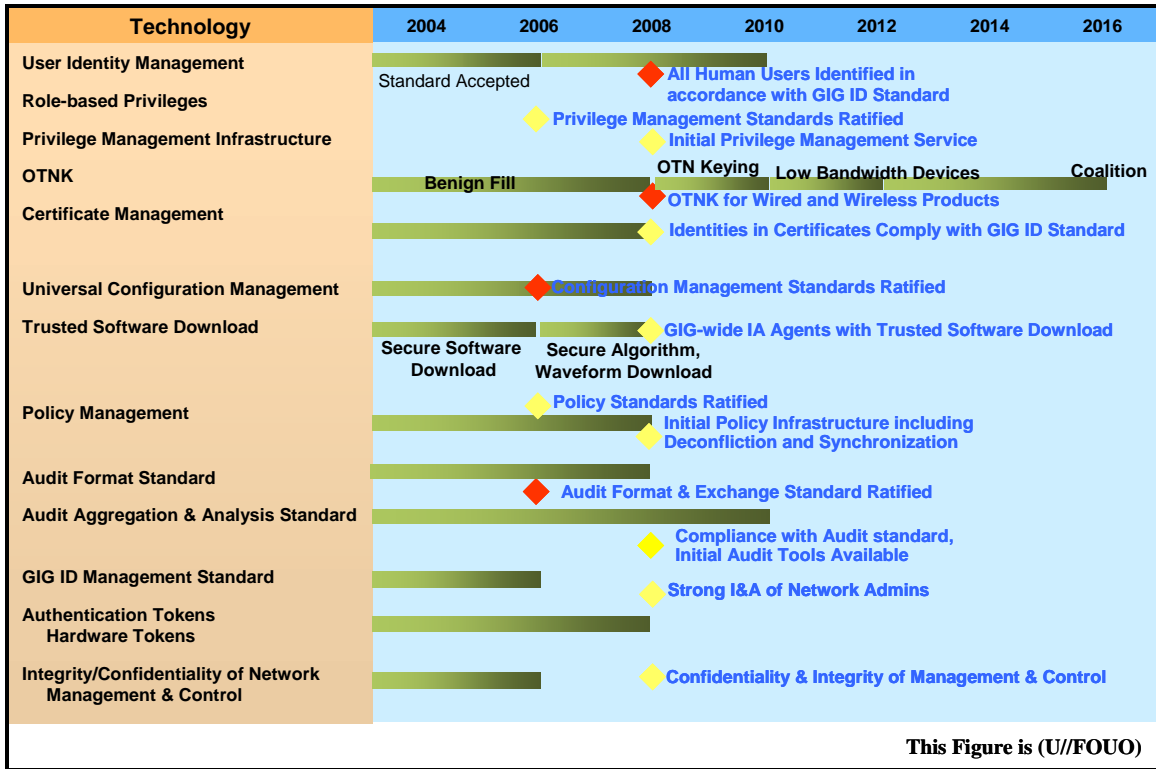
14820 (U//FOUO) Within the Black Core, packets are protected at the network layer. Network layer
14821 protection inherently has traffic analysis, network mapping, and covert channel issues. The risk
14822 varies on a link-by-link basis across the Black Core as each link can be characterized as high,
14823 medium, or low risk. The definition of these links can be found in The Configuration Guidance
14824 for HAIPE Protected Networks, version 2.0.

- 14825 • (U//FOUO) Gap: Within the Black Core certain links traverse high-risk environments
14826 with higher threat of traffic analysis, network mapping, and exfiltration. Cost-effective
14827 solutions are required to protect individual links that are characterized as high risk.

14828 (U//FOUO) Recommendations: Develop a strategy for developing a low-cost
14829 protection capability that can be deployed to protect high-risk links. The solution
14830 must protect against traffic analysis, network mapping, and prevent an exfiltration
14831 path across the link. Solutions must also be easily manageable so that they are not
14832 impracticable or are prohibitively costly to use.

14833 **3.3 (U) ASSURED ENTERPRISE MANAGEMENT AND CONTROL SUMMARY**
 14834 (U//FOUO) The technologies that support this cornerstone are organized into seven general
 14835 categories: Identity Management, Privilege Management, Key Management, Certificate
 14836 Management, Configuration Management, Policy Management, and Audit Management.

14837 (U) Figure 3.3-1 provides an overview of the technologies and how they support the IA
 14838 imperatives listed in the Transition Strategy. While none of the technologies will be completed in
 14839 time to meet the 2008 IA imperatives, the milestones are achievable if current efforts are
 14840 accelerated.



14841
 14842 **Figure 3.3-1: (U//FOUO) Technology Timeline for Assure Enterprise Management and**
 14843 **Control**

14844 **3.3.1 (U) Identity Management Technologies**

14845 (U//FOUO) To meet the 2008 Transition Strategy, a GIG-wide Identity Management standard
14846 must be created to describe users, their properties, and profiles. Identity management
14847 technologies and standards have traditionally varied dramatically from enterprise to enterprise.
14848 For the GIG, a standard should be developed that provides unique, persistent, nonforgeable
14849 identities. Since being able to uniquely identify each user is perhaps the most relied on capability
14850 in the GIG, developing this standard needs to start immediately.

14851 (U//FOUO) While the 2008 Transition Strategy only requires that Identities be defined for users,
14852 to support the 2012 Transition Strategy, Identities also need to be defined for GIG devices and
14853 services. Development of an Identity standard for devices and services also needs to start.

- 14854 • (U//FOUO) Gap: Lack of a unique, persistent, nonforgeable GIG-wide Identity for
14855 human users.

14856 (U//FOUO) Recommendation: Immediately begin development of a GIG Identity
14857 standard for human users.

- 14858 • (U//FOUO) Gap: Lack of a unique, persistent, nonforgeable GIG-wide standard for
14859 devices and services.

14860 (U//FOUO) Recommendation: Begin development of a GIG Identity standard for
14861 devices and services.

14862 **3.3.2 (U) Inventory Management Technologies**

14863 (U) The commercial world is embracing RFID as a technology for inventory management.
14864 Although the technology has been available since 1974, standardized, interoperable tags and
14865 readers are a recent innovation. Although the field is rapidly developing on its own, the focus is
14866 on developing low-cost, passive RFID tags. Consumer privacy issues have raised the concern
14867 that RFID tags are still active after leaving the retail sales point and could be used to track
14868 individuals, so some work is being done to develop RFID tags that can be killed—rendered
14869 permanently inert or unresponsive to a reader interrogation, or rendered temporarily inert.
14870 However, no apparent work is being done to develop secure RFID tags which would respond
14871 only to interrogation and commands from an authenticated reader.

- 14872 • (U) Gap: Lack of security for RFID. Lack of ability for RFID tags to only respond to
14873 commands from an authorized reader. Lack of an ability to disable RFID tags so they
14874 may not be used as tracking devices.

14875 (U//FOUO) Recommendation: Develop a security architecture and security
14876 mechanisms for RFID tags.

- 14877 • (U) Gap: Lack of a GIG Inventory Management Infrastructure.

14878 (U//FOUO) Recommendation: Develop a Inventory Management Infrastructure that
14879 tracks and manages GIG assets.

14880 **3.3.3 (U) Privilege Management Technologies**

14881 (U//FOUO) There is an existing standard for privileges (Attribute Certificates) that stems from
14882 an extension of the X.509 standard and has been adopted widely by the Public Key Infrastructure
14883 (PKI). Attribute Certificates effectively bind privileges to a certificate. Other privilege
14884 management approaches also exist that address role-based privileges.

- 14885 • (U//FOUO) Gap: Lack of definition of privileges necessary to support the GIG.

14886 (U//FOUO)) Recommendations: Research should be initiated that defines the
14887 necessary privilege set and privileges for the GIG. How privileges are stored,
14888 retrieved, and managed within the GIG must also be defined so it is scalable to the
14889 GIG enterprise. Rule-based privileges need to be defined for human users, devices,
14890 services, and COIs. Role-based privileges also need to be defined so that a GIG entity
14891 can dynamically switch between roles and still receive the appropriate privileges.
14892 Privileges also need to be defined in the context of the RAdAC model.

- 14893 • (U//FOUO) Gap: Lack of sufficient support for privileges, trust anchors, and other access
14894 control information required by the GIG.

14895 (U) Recommendations:

14896 1) (U//FOUO) Develop GIG requirements for privileges, trust anchors, and other
14897 access control information required by the GIG. Devise an efficient and scalable
14898 approach and supporting standard for managing this information.

14899 2) (U//FOUO) Evaluate exiting privilege technologies for meeting the GIG Privilege
14900 Management requirements.

14901 (U//FOUO) To meet the 2008 Transition Strategy, the above issues need to be standardized, and
14902 initial products conforming to the standards be made available to provide an initial privilege
14903 management infrastructure. It is recommended that the standardization activity begin
14904 immediately in order to meet this timeline.

14905 **3.3.4 (U) Key Management Technologies**

14906 (U//FOUO) Some of the technologies in Key Management, such as generation, initial key load
14907 and rekeying are quite mature and have been adopted under various classified (e.g., Electronic
14908 Key Management System [EKMS]) and unclassified (e.g., DoD Public Key Infrastructure [PKI])
14909 infrastructures. However, the management and distribution of crypto-material still remains a very
14910 manually intensive process in some cases. Technologies that reduce the distribution burden, such
14911 as Over the Air Distribution (OTAD), are available on a relatively small number of devices. The
14912 future Over the Network Keying (OTNK) initiative is expected to further reduce the
14913 management burden of key material. Many of the issues that surround technological issues of
14914 high assurance with key management practices are being addressed by the Key Management
14915 Infrastructure (KMI) initiative.

- 14916 • (U//FOUO) Gap: Weakness or non-existence of associating policy controls (including
14917 dynamic policy changes) in an automated fashion to various aspects of the key
14918 management cycle.

14919 (U//FOUO) Recommendation: Develop standards so that automation can be built
14920 into promulgating dynamic policy changes into the necessary rules and regulations so
14921 that key registration, packaging, distribution, re-keying, revocation, and destruction
14922 work seamlessly and in an up-to-date, situational, manner.

- 14923 • (U//FOUO) Gap: Lack of sufficient automated key distribution and management
14924 techniques.

14925 Recommendations: Continue to develop the OTNK infrastructure to provide
14926 automated key distribution. Either revise OTNK or develop additional automated
14927 procedures to meet the needs for tactical and special needs users.

- 14928 • (U//FOUO) Gap: Lack of standards for unified key labeling, packaging, and distribution
14929 formats.

14930 (U//FOUO) Recommendation: Develop standards to unify key packaging and
14931 distribution in order to eliminate or reduce manual error-prone and human access
14932 vulnerabilities towards threats. Standards and technologies should include the
14933 incorporation of Multi-Level systems and data stores.

- 14934 • (U//FOUO) Gap: Lack of EKMS support for symmetric and Type 3 keys.

14935 (U//FOUO) Recommendation: The management of symmetric keys and Type 3 keys
14936 needs to be included in the evolution of the KMI.

14937 **3.3.5 (U) Certificate Management Technologies**

14938 (U//FOUO) The only existing Certificate Management standard is found in the PKI arena.
14939 Although PKI has been around for many years, PKI has interoperability limitations at the
14940 application and component levels. There are no identified interoperability standards or
14941 technologies that specify the interfaces for certificate and data exchange between certificate
14942 authorities. There are secure transports currently in use for certificates, but as such, there is no
14943 GIG-wide enterprise policy that governs what these access control restrictions should be.

14944 (U//FOUO) There currently is ongoing work to enhance certificate attributes that aim to capture
14945 additional significant information such as subject privileges, trust anchor information, and other
14946 necessary identity, trust, distribution, and access control information. There are also initiatives
14947 that are attempting to specify and collate various levels and classifications of certificates such as
14948 the Class 3, Class 4, and Class 5 Government and commercial-type certificates. These efforts
14949 should continue since they are necessary for the GIG.

- 14950 • (U//FOUO) Gap: Lack of definition and infrastructure support for Class 5 certificates.

14951 (U//FOUO) Recommendations: Develop a standard and the necessary infrastructure
14952 for Class 5 certificates.

- 14953 • (U//FOUO) Gap: Lack of support for the GIG Identity standard in the PKI.

14954 (U//FOUO) Recommendation: Once the GIG Identity management standard has been
14955 approved, PKI must evolve to support the newly defined identities.

- 14956 • (U//FOUO) Gap: Lack of cryptographic binding of a user/entity's information and
14957 attributes of its public key material and the associated trust anchor. The binding is needed
14958 to certify that the private key corresponding to the public key in the certificate is held by
14959 the same user/entity. There is a need to have binding strength increase with the strength
14960 of the cryptographic algorithm and key length used.

14961 (U//FOUO) Recommendation: Develop a standard to address the appropriate
14962 cryptographic binding of attributes to GIG entity.

14963 **3.3.6 (U) Configuration Management Technologies**

14964 (U//FOUO) Individual point solutions for various parts of configuration management are mature.
14965 There are examples of successfully deployed products in commercial environments. However,
14966 none of the technologies meets GIG requirements for the high assurance required to securely
14967 manage Information Assurance assets across a lower assurance network.

14968 • (U//FOUO) Gap: Lack of product support for:

- 14969 • (U//FOUO) Protected communication paths between configuration management
14970 server and managed device
- 14971 • (U//FOUO) Authentication of client machines and authentication of configuration
14972 management servers
- 14973 • (U//FOUO) Ability to model configuration changes before deployment
- 14974 • (U//FOUO) Testing configuration. Many products support test deployments
14975 before a patch or upgrade deployment, but it is not industry wide
- 14976 • (U//FOUO) Authenticated or cryptographic verification of configurations. Current
14977 products assumed the device configuration information could be trusted
- 14978 • (U//FOUO) Sensitive material distribution, such as keys, which require
14979 protection, receipts, and auditable tracking of delivery
- 14980 • (U//FOUO) Remote update of firmware

14981 (U//FOUO) Recommendations: Develop interoperable solutions to the above list of
14982 configuration management product gaps to support deployment of GIG-wide
14983 Configuration Management agents.

14984 • (U//FOUO) Gap: Lack of Trusted download capability for software, algorithms, and
14985 waveforms.

14986 (U//FOUO) Recommendation: Continue the development of a trusted software
14987 download capability.

14988 **3.3.7 (U) Policy Management Technologies**

14989 (U//FOUO) There are several vendor-specific policy management products available today, but
14990 they do not incorporate security attributes required by the GIG into their products. In order to
14991 meet the 2008 Transition Strategy standard for policy definition, deconfliction and
14992 synchronization need to be developed and ratified. Initial products complying with these
14993 standards are also required in order to begin deploying a policy management infrastructure.

14994 • (U//FOUO) Gap: Lack of standards for specifying policy. The policy language needs to
14995 cover all GIG policies: access control, quality of protection, quality of service, transport,
14996 audit, computer network defense, and policies covering the hardware and software
14997 associated with GIG assets.

14998 (U//FOUO) Recommendations: There are several initiatives to define policy
14999 languages. These initiatives must be examined to determine their suitability for the
15000 GIG. Security attributes must be inserted into the appropriate policy languages to
15001 ensure that the GIG IA policy can be managed.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

- 15002 • (U//FOUO) Gap: Lack of research in policy deconfliction. Existing research has shown
15003 that there can be different types of conflicts between policies. In cases where one policy
15004 requires a particular action and an overlapping policy does not address that action, the
15005 conflicts can generally be resolved. However, when one policy requires an action and an
15006 overlapping policy explicitly prohibits it, the conflict can not be resolved by an
15007 automated system.

- 15008 • (U//FOUO) Recommendation: Build on existing research in policy deconfliction to
15009 establish general rules and procedures, to automate the deconfliction process to the
15010 maximum extent possible.

- 15011 • (U//FOUO) Gap: Lack of tools that provide policy deconfliction. Current tools require
15012 human intervention for policy deconfliction.
15013 (U//FOUO) Recommendations: Develop technology standards for how to handle IA
15014 policy conflicts.

- 15015 • (U//FOUO) Gap: Lack of standard approaches (push/pull) for policy distribution,
15016 including protection of policy at rest and in transit, policy validation, distribution error,
15017 and exception handling.
15018 (U//FOUO) Recommendations: Develop standard approaches to provide policy
15019 distribution. Both push and pull policy distribution will be used in the GIG. Standard
15020 approaches need to address multiple policy distribution techniques. Develop
15021 standards for policy validation, error and exception handling.

- 15022 • (U//FOUO) Gap: Lack of methods for performing policy synchronization. It is not
15023 feasible to assume that policy changes will be implemented instantaneously across the
15024 GIG or even across an enterprise. Methods and procedures must be in place to allow a
15025 policy to be propagated at a reasonable pace across multiple components.
15026 (U//FOUO) Recommendations: Develop standards that allow policy changes to be
15027 propagated at a reasonable rate across an enterprise. Policy propagation should not
15028 create a window of vulnerability during the transition and should not create a denial
15029 of service condition.

- 15030 • (U//FOUO) Gap: Lack of tools for analyzing the affects of policy and multiple policy
15031 objects on the GIG. New policy can create undesired conditions through incorrect policy
15032 translation or incorrectly formulated policy.
15033 (U//FOUO) Recommendations: Develop tools for modeling new policy on multiple
15034 classes of objects and testing their implementation to verify that policy is being
15035 enforced as intended and the new policy is performing the desired changes.

- 15036 • (U) Gap: Lack of a consistent user interface for managing policy on multiple classes of
15037 assets.
15038 (U//FOUO) Recommendations: Develop tools that can manage multiple classes of
15039 assets, including devices from multiple vendors.

- 15040 • (U//FOUO) Gap: Lack of tools for translating natural language policies into policy base
15041 logic.

15042 (U//FOUO) Recommendations: Develop tools that translate a human understandable
15043 (i.e., natural language) policy statement into a configuration file that can be used by a
15044 device in the GIG. Translation must also be done in reverse; that is, from a
15045 configuration file into natural language policy statements.

15046 **3.3.8 (U) Audit Management Technologies**

15047 (U//FOUO) Audit Management today exhibits a lack of maturity in standards-based solutions.
15048 There are vendors that provide some type of audit capability built into their proprietary solutions.
15049 Without standards in technology, interoperability (within components, log formats, audit
15050 analyses, etc.), and policies, there is a gap in the unified audit management scheme required by
15051 the GIG enterprise.

- 15052 • (U//FOUO) Gap: Lack of standards in technology, interfaces, interoperability, and
15053 policies needed to support Audit Management for the 2008 Transition Strategy.

15054 (U//FOUO) Recommendations: To ensure that 2008 Transition Strategy for Audit
15055 Management can be met, work to define the following set of standards must begin
15056 immediately and be completed by 2006:

- 15057 1) (U//FOUO) Standard Log and event formats to capture and record normalized
15058 GIG-wide activities and system performance
- 15059 2) (U//FOUO) Standards for correlation, analysis, and alerting services that subscribe
15060 to audit data publishing
- 15061 3) (U//FOUO) Standardized Securing of audit data into one-way stores
- 15062 4) (U//FOUO) Standardizing Agents and Agentless components for interoperability
15063 and security assurance
- 15064 5) (U//FOUO) Standards for tools that monitor system resources
- 15065 6) (U//FOUO) Central monitoring and interfacing standards
- 15066 7) (U//FOUO) Policies on what events are to be audited under what circumstances.
15067 This must include: (a) what actions to take when the audit log becomes full (e.g., stop
15068 auditing new events; overwrite the oldest existing records; shut down the system); (b)
15069 whether auditing can change in an automated manner in response to system events
15070 (e.g., if processing load becomes too high, scale back auditing to allocate more
15071 resources to production work); (c) what privileges are required to change audit
15072 parameters; and (d) deletion of audit records.

- 15073 • (U//FOUO) Gap: Lack of Audit analysis tools.

15074 (U//FOUO) Recommendations: Develop Audit Management products and tools that
15075 comply with the above list of standards.

15076 **3.3.9 (U) Confidentiality & Integrity of Network Management & Control Technologies**
15077 (U//FOUO) Solutions currently exist for providing confidentiality and integrity of network
15078 management flows. However, they are not widely deployed across the GIG. An effort by GIG
15079 programs must be made to provide secure network management solutions.

- 15080 • (U//FOUO) Gap: Lack of interoperable solutions for providing confidentiality and
15081 integrity of network control flows. When solutions exist they are usually specific for a
15082 particular protocol. Implementing several protocol unique solutions can impose a heavy
15083 management burden on a system without much benefit due to incomplete security
15084 solutions.

15085 (U//FOUO) Recommendations: An approach to providing confidentiality and
15086 integrity for all network control protocols needs to be defined such that a secure
15087 solution is provided that does not unnecessarily burden the operation of the GIG.
15088 Research should be started to develop and socialize this solution with the GIG and
15089 commercial industry. Any potential solution must be embraced by commercial
15090 industry to have the interoperable implementations required by the GIG.

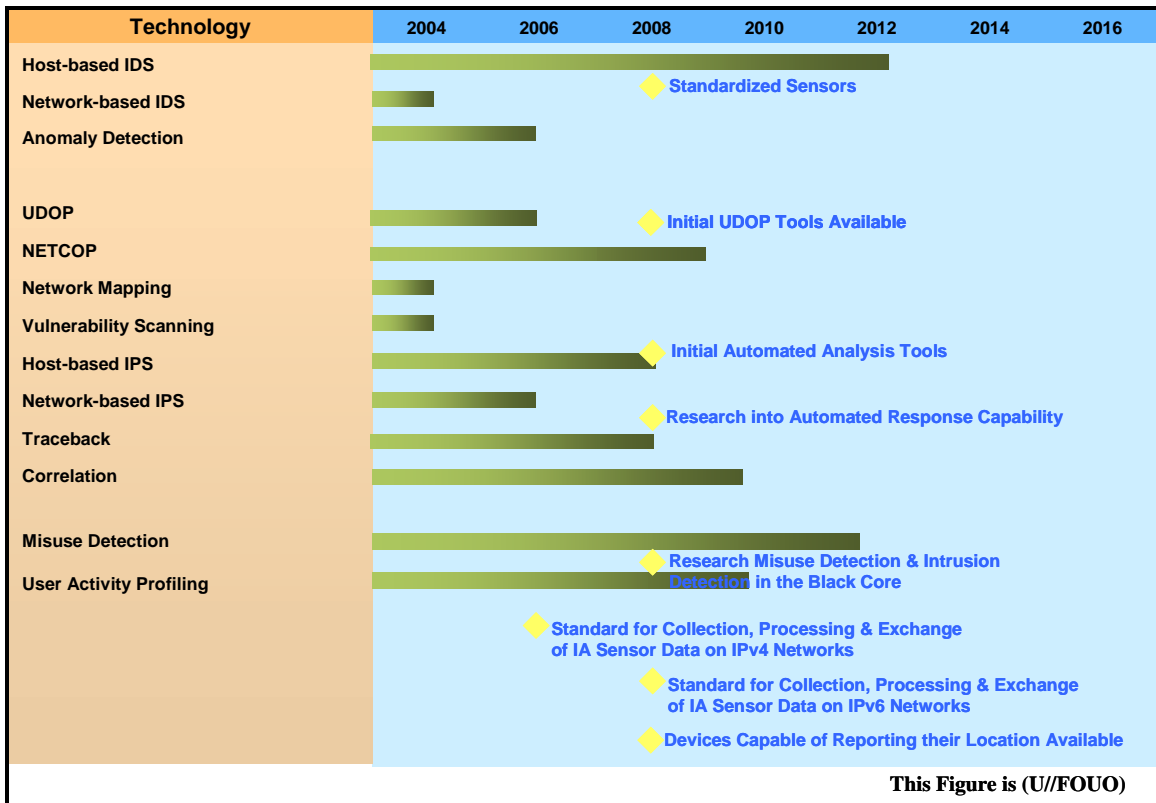
15091
15092
15093
15094

3.4 (U) CYBER SITUATIONAL AWARENESS AND NETWORK DEFENSE SUMMARY

(U//FOUO) Technologies that support this cornerstone are organized into five general categories: Protection, Monitoring, Detection, Analysis, and Response.

15095
15096
15097
15098
15099

(U) Figure 3.4-1 provides an overview of the technologies and how they support the IA imperatives listed in the Transition Strategy. As shown, while none of the technologies will be completed in time to meet the 2008 IA imperatives, the milestones are achievable if current efforts are accelerated. Some imperatives have no supporting technologies identified in this release of the document. These are discussed in the gaps below.



15100
15101
15102

Figure 3.4-1: (U//FOUO) Technology Timeline for Cyber Situational Awareness and Network Defense

15103 **3.4.1 (U) Protection Technologies**

15104 (U//FOUO) Protection technologies relevant to the GIG have been developed by the commercial
15105 market and are—for the most part—mature. However, additional work is needed to adapt these
15106 solutions in order to meet the unique requirements imposed by the GIG's environments.

- 15107 • (U//FOUO) Gap: Protection technologies are static in their operation and require manual
15108 configuration.

15109 (U//FOUO) Recommendation: Research and develop dynamic protection mechanisms
15110 capable of modifying device settings (e.g., ports and protocols on network and host-
15111 based firewalls) according to current network conditions and published Information
15112 Operations Condition (INFOCON).

- 15113 • (U//FOUO) Gap: Honeynet and Honeypot technologies are well developed, but current
15114 implementations do not provide semi-automated operation or support the data volume
15115 needed for useful implementation in the GIG.

15116 (U//FOUO) Recommendation: Existing tools used to capture, control, and analyze
15117 data must be enhanced to support automatic filtering of larger amounts of data,
15118 correlate with other network operations information (e.g., Intrusion Detection System
15119 (IDS) activity), and provide a unified view of ongoing attacks.

15120 **3.4.2 (U) Monitoring Technologies**

15121 (U//FOUO) State-of-practice approaches for Computer Network Defense (CND) monitoring rely
15122 on access to unencrypted traffic and often employ small numbers of sensors located on high-
15123 speed backbones. Such approaches are incompatible with the GIG. As the Black Core evolves,
15124 unencrypted traffic will be limited—eventually eliminated, so network monitoring must adapt.
15125 Moreover, the sheer size and complexity of the GIG mandates use of a large and distributed
15126 network of sensors located on lower data bandwidth links. These sensors will provide
15127 information to a central correlation function to provide an integrated picture of GIG network
15128 state.

15129 (U) Specific gaps and recommendations for monitoring technologies are listed below:

- 15130 • (U//FOUO) Gap: Current sensor data collection, manipulation, and storage capabilities
15131 are insufficient to support the GIG's aggregate bandwidth.

15132 (U//FOUO) Recommendation: Inexpensive data collection technologies that work at
15133 line speed must be developed, along with a scaleable architecture to employ those
15134 technologies and sensors.

- 15135 • (U//FOUO) Gap: Architecture for a sensor grid, capable of supporting the real time data
15136 needs of the User Defined Operational Picture (UDOP), has not been defined. This grid
15137 would require centralized data storage capabilities sufficient to meet GIG needs.

15138 (U//FOUO) Recommendation: Devise a technology development plan for deploying a
15139 distributed sensor grid across the GIG. The sensor grid must report collected data to a
15140 central location. Data storage requirements for the centralized store must be defined.

- 15141 • (U//FOUO) Gap: The lack of standards for sensor data prevents the design and
15142 implementation of a global sensor grid.
15143 (U//FOUO) Recommendation: Develop a standard for the collection of sensor data.
15144 The standard must be integrated into new products to support the 2008 Transition
15145 Strategy.
- 15146 • (U//FOUO) Gap: Current sensors are unable to monitor encrypted packets in the Black
15147 Core.
15148 (U//FOUO) Recommendation: Conduct research aimed at providing CND monitoring
15149 capabilities on encrypted traffic in the Black Core.
- 15150 • (U//FOUO) Gap: Lack of monitoring capabilities for Internet Protocol version 6 (IPv6)
15151 networks.
15152 (U//FOUO) Recommendation: Conduct research aimed at defining monitoring
15153 capabilities for IPv6 networks. Incorporate IPv6 monitoring capabilities into GIG
15154 situational awareness and monitoring capabilities.
- 15155 • (U//FOUO) Gap: Current network mapping and discovery tools do not provide the
15156 collaborative capabilities across a hierarchical architecture that will be needed to support
15157 sophisticated monitoring and controls across the large, complex, and dynamic GIG.
15158 (U//FOUO) Recommendation: An agent-based collaborative network discovery
15159 architecture must be devised and the associated tool technologies developed.

15160 3.4.3 (U) Detection Technologies

15161 (U//FOUO) Detection technologies consist of Intrusion Detection Systems (IDS), Intrusion
15162 Protection Systems (IPS), and user profiling. IDSs have been developed and used for years in the
15163 commercial market to detect network intrusions and attacks. However, they have not been used
15164 on such an expansive network as the GIG. IPSs are a relatively new technology, combining
15165 detection and response capabilities in one package. As a detection capability, however, they offer
15166 nothing new relative to IDSs. User-profiling is used to detect insider misuse, but these
15167 technologies are fairly new and immature.

15168 (U) Specific gaps and recommendations for detection technologies are listed below:

- 15169 • (U//FOUO) Gap: The current Network-based Intrusion Detection System (NIDS)
15170 architectures used by Defense Information Systems Agency (DISA) and the Services are
15171 not compatible with the Black Core concept and may not scale well.
15172 (U//FOUO) Recommendation: Initiate architecture, technology, and standards
15173 development efforts to integrate NIDS and Host-based Intrusion Detection System
15174 (HIDS) into the global and tiered architecture envisioned for the GIG.
- 15175 • (U//FOUO) Gap: Anomaly detection offers several significant potential benefits, most
15176 notably the ability to detect zero-day attacks. However, current implementations are
15177 plagued with high, false-alarm rates that make this technology unusable for GIG
15178 applications.

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15179 (U//FOUO) Recommendation: Accelerate and guide ongoing research to develop
15180 robust anomaly detection capabilities with low false-alarm rates.

- 15181 • (U//FOUO) Gap: Current intrusion detection capabilities rely on unencrypted packet
15182 headers and payloads to detect anomalous activity.

15183 (U//FOUO) Recommendation: Initiate research to develop advanced intrusion
15184 detection systems capable of interoperating on encrypted segments.

- 15185 • (U//FOUO) Gap: Standards for IDS data and communication are needed to implement a
15186 comprehensive and distributed intrusion detection scheme for the GIG. An IETF working
15187 group, the Intrusion Detection Working Group (IDWG), is developing standards that will
15188 formalize data formats and exchange processes, but this work is not yet complete.

15189 (U//FOUO) Recommendation: Through participation on the IDWG, the DoD should
15190 influence the IDS standards currently under development to meet GIG needs.

- 15191 • (U//FOUO) Gap: The IDS data exchange requirements for a global network, such as the
15192 GIG, could present an undue bandwidth burden on the network being protected. It is not
15193 known what these data bandwidth requirements are or what is the best approach for
15194 architecting, connecting, and controlling the IDSs across the GIG.

15195 (U//FOUO) Recommendation: Devise and study architecture alternatives for
15196 integrating and controlling GIG IDSs. This study should include trade-offs of key
15197 characteristics, such as expected performance, complexity, operational costs (in terms
15198 of manpower), and inter-IDS communications bandwidth.

- 15199 • (U//FOUO) Gap: Detecting insider misuse must rely heavily on user profiling of expected
15200 normal behavior as well as on application-specific rules. However, there are significant
15201 limitations to this approach, including detectability of slow profile changes and high false
15202 alarm rates. Some Government Off-The-Shelf (GOTS) and Commercial Off-The-Shelf
15203 (COTS) user profiling tools are available, but much more work is needed to bring their
15204 capabilities to maturity.

15205 (U//FOUO) Recommendation: Development work should be undertaken to determine
15206 additional user observables (e.g., websites frequently visited and other individuals
15207 with whom the user exchanges e-mail) and to refine existing tools to incorporate this
15208 additional information.

15209 **3.4.4 (U) Analysis Technologies**

15210 (U//FOUO) The breadth, depth, and dynamic nature of the GIG present huge challenges for the
15211 analysis component of CND. Correlation processes will have to deal with a large, distributed
15212 sensor grid that generates enormous amounts of data from a variety of sources. The manual
15213 attack attribution techniques currently used will be inadequate for the expected large volume of
15214 network traffic on the GIG.

15215 (U) Specific gaps and recommendations for analysis technologies are listed below:

- 15216 • (U//FOUO) Gap: There are several different trace-back techniques that have been used
15217 with varying success to identify the source of an attack. However, they often feature
15218 manual operation and operate on unencrypted packets. These are serious limitations for
15219 the GIG.

15220 (U//FOUO) Recommendation: Continue research and development to advance current
15221 techniques for use in the Black Core.

- 15222 • (U//FOUO) Gap: Existing trace-back approaches are based on correlating similar
15223 transactions along a connection path. Many attacks use remote hosts to launch attacks,
15224 which are effective at circumventing these trace-back techniques.

15225 (U//FOUO) Recommendation: Research new techniques that deduce correlation of
15226 intent along connection paths, perhaps through a combination of transaction
15227 correlation and signature analysis of packet content. Such techniques would be
15228 cognizant of attack strategies (i.e., attacks launched through one or more
15229 intermediaries) and look for correlations of the resulting packet sequences among
15230 hosts across the networks.

- 15231 • (U//FOUO) Gap: Vulnerability analysis tools consider individual vulnerabilities
15232 independent of one another and in the context of a single host. The vulnerability of an
15233 enclave or network, however, is determined—in part—by the aggregation of host
15234 vulnerabilities. Current tools do not determine aggregate vulnerability.

15235 (U//FOUO) Recommendation: Extend the capabilities of current vulnerability
15236 analysis tools to include Topological Vulnerability Analysis across groups of hosts in
15237 networks.

- 15238 • (U//FOUO) Gap: The large number of sensors disbursed across multiple hierarchical
15239 levels of the GIG represents huge challenges for analysis. Correlation technology must be
15240 able to handle large volumes of intrusion detection data in real time, fuse heterogeneous
15241 data from disparate levels in the global network hierarchy, and accommodate other
15242 operational factors, such as typical adversary behavior, normal network activity, and
15243 mission critical components and applications. No technologies exist to provide this
15244 analysis capability for such a large network.

15245 (U//FOUO) Recommendation: Research must be undertaken to devise a unified
15246 correlation and analysis approach for the GIG CND effort. In addition to correlation
15247 of intrusion detection data, focus should include key performance measures critical
15248 for a GIG-sized network, such as dropped-alert rate, false alarm rate, bandwidth for

15249 communication between distributed processing nodes, and processing latency. Part of
15250 this work must be an analysis-of-alternatives (AoA) study to determine sensor grid
15251 architecture limitations imposed by the analysis approach.

15252 **3.4.5 (U) Response Technologies**

15253 (U//FOUO) Today, responses to computer network attacks are largely manual, because available
15254 tools are limited in their capabilities, and uncertainties exist on the impact of automated
15255 responses on the mission of the enterprise. However, due to its size and criticality, the GIG will
15256 require an automated or at least semi-automated response to network attacks.

15257 (U) Specific gaps and recommendations for response technologies are listed below:

- 15258 • (U//FOUO) Gap: CND analysts and warfighters must understand, a priori, the operational
15259 implications of shutting down or restricting capabilities in response to network attack.
15260 Until combatants are able to fully understand the implications of network response
15261 actions to attacks, automated response capabilities will not be adopted. Some research has
15262 been done in developing tools for assessing operational impact of attack responses.
15263 However, these tools have not evolved to the point where they can provide the user an
15264 estimate of impact on specific missions.

15265 (U//FOUO) Recommendation: Accelerate and guide research for modeling the impact
15266 of attack response on warfighting operations in context of the GIG.

- 15267 • (U//FOUO) Gap: A semi-automated approach for responding to attacks is needed. A
15268 fully manual process permits deliberate consideration of response options and more
15269 complete attack analysis, but it is labor intensive and takes more time (especially for the
15270 GIG), so attack damage could be more widespread. Automated responses can quickly
15271 contain an attack, but the impact on the network can be unpredictable and unnecessarily
15272 restrict warfighting operations. For maximum response effectiveness in the GIG, a
15273 balance between manual and automated response is needed, but no work has yet been
15274 done to determine how this could be done.

15275 (U//FOUO) Recommendation: Continue research to determine the best approach for a
15276 semi-automated response to network attack, taking into consideration effectiveness of
15277 attribution activities, impact of response on warfighting operations, and manpower
15278 required.

- 15279 • (U//FOUO) Gap: Current response capabilities are limited to simplistic point solutions,
15280 such as blocking a port and IP address pair at the network boundary, which will become
15281 ineffective against sophisticated attacks.

15282 (U//FOUO) Recommendation: Continue research into sophisticated response capabilities applied
15283 to distributed network components.

15284	4 (U) ACRONYMS AND ABBREVIATIONS	
15285	AA	Attribute Authority
15286	ABAC	Attribute-Based Access Control
15287	ABNF	Augmented Backus-Naur Format
15288	AC	Attribute Certificate
15289		Access Control
15290	ACAP	Application Configuration Access Protocol
15291	ACE	Advanced Computing Environment
15292	ACL	Access Control List
15293	ACOA	Alternate Course Of Action
15294	ACP	Allied Communications Publication
15295	ACRL	Attribute Certificate Revocation List
15296	ACS	Access Control Server
15297	AEHF	Advanced Extremely High Frequency
15298	AES	Advanced Encryption Standard
15299	AFS	Agent Functional Stack
15300	AH	Authentication Header
15301	AIC	Adaptive Information Control
15302	AICE	Agile Information Control Environment
15303	A/J	Anti-Jam
15304	AKA	Authentication and Key Agreement
15305	a.k.a	Also known as
15306	AKP	Advanced Key Processor
15307	ANDVT	Advanced Narrowband Digital Voice Terminal
15308	ANSI	American National Standards Institute

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15309	AODV	Ad-hoc On-Demand Distant Vector
15310	API	Application Programming Interface
15311	AS	Autonomous System
15312	ASD	Assistant Secretary of Defense
15313	ASIC	Application-Specific Integrated Circuit
15314	ASN1	Abstract Syntax Notation One
15315	AS&W	Attack, Sensing & Warning
15316	ASCII	American Standard Code for Information Interchange
15317	ASN.1	Abstract Syntax Notation
15318	ATM	Asynchronous Transfer Mode
15319	AVLAN	Authenticated Virtual Local Area Network
15320	BAAD	Battlefield Awareness and Data Dissemination
15321	BC	Biometric Consortium
15322	BEM	Biometric Evaluation Methodology
15323	BER	Basic Encoding Rules
15324		Bit Error Rate
15325	BET	Bulk Encrypted Transaction
15326	BGP	Border Gateway Protocol
15327	BIOS	Basic Input-Output System
15328	BIS	Boot Integrity Services
15329	BMO	Biometric Management Office
15330	BOOTP	Boot Protocol
15331	BoSS	Baystack Operating System Switching Software
15332	BSP	Biometric Service Providers
15333	C2	Command and Control

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15334	C2G	Command and Control Guard
15335	C3I	Command, Control, Communications and Intelligence
15336	CA	Certification Authority
15337	CAC	Common Access Card
15338	CAPCO	Controlled Access Program Coordinator Office
15339	CAPI	Crypto API
15340	CAW	Certification Authority Workstation
15341	CBC	Cipher Block Chaining
15342	CBEFF	Common Biometric Exchange Formats Framework
15343	CBIS	Content-Based Information Security
15344	CC	Common Criteria
15345	CCITT	Consultative Committee on International Telegraphy and Telephony
15346	CDMA	Code Division Multiple Access
15347	CDS	Cross-Domain Solutions
15348	CDSA	Common Data Security Architecture
15349	CEM	Common Evaluation Methodology
15350		Constructive Key Management
15351	CENTRIXS	Combined Enterprise Regional Information Exchange System
15352	CEP	Certificate Enrollment Protocol
15353	CER	Canonical Encoding Rules
15354	CERIAS	Center for Education and Research in Information Assurance and Security
15355	CERT	Computer Emergency Readiness Team
15356	CES	Core Enterprise Service
15357	CHAP	Challenge Handshake Authentication Protocol
15358	CIDF	Common Intrusion Detection Framework

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15359	CIF	Component Impact Factor
15360	CIFS	Common Internet File System
15361	CIM	Common Information Model
15362	CJCSI	Chairman of the Joint Chiefs of Staff Instruction
15363	CKL	Compromised Key List
15364	CKM	Constructive Key Management
15365	CLF	Common Log Format
15366	CLI	Command Line Interface
15367	CM	Configuration Management
15368	CMC	COMSEC Material Control
15369	CMCS	COMSEC Material Control System
15370	CMI	Certificate Management Infrastructure
15371	CMMF	Certificate Management Message Format
15372	CMP	Certificate Management Protocol
15373	CMS	Cryptographic Message Syntax
15374	CND	Computer Network Defense
15375	COA	Course of Action
15376	COI	Community of Interest
15377	COMPUSEC	Computer Security
15378	COMSEC	Communications Security
15379	CONOP	Concept of Operation
15380	CONUS	Continental United States
15381	COP	Common Operating Picture
15382	CoP	Coalition Partner
15383	COPS	Common Open Policy Service

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15384	CORBA	Common Object Request Broker Architecture
15385	CoS	Class of Service
15386	COTS	Commercial-off-the-Shelf
15387	COWANS	Coalition Operational Wide Area Networks
15388	CPS	Certification Practice Statement
15389	CPU	Central Processing Unit
15390	CRD	Capstone Requirements Document
15391	CRL	Certificate Revocation List
15392	CRMF	Certificate Request Message Format
15393	CSEE	Computer Science and Electrical Engineering
15394	CSIRT	Computer Security Incident Response Team
15395	CSP	Common Security Protocol
15396	CSRC	Contributing Source Real-time Content
15397	CVE	Common Vulnerabilities and Exposures
15398	DAC	Discretionary Access Control
15399	DACP	Digital Access Control Policy
15400	DAML	DARPA Agent Markup Language
15401	DARPA	Defense Advanced Research Projects Agency
15402	DAV	Distributed Authoring & Versioning
15403	DBMS	Database Management System
15404	DCID	Director of Central Intelligence Directive
15405	DCIS	Defense Cross-credentialing Identification System
15406	DDDS	Dynamic Delegation Discovery System
15407	DDES	Double Data Encryption Standard
15408	DDMS	DoD's Discovery Metadata Specification

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15409	DDoS	Distributed Denial of Service
15410	DEERS	Defense Enrollment Eligibility Reporting System
15411	DEFCON	Defense Condition
15412	DER	Distinguished Encoding Rules
15413	DES	Data Encryption Standard
15414	DeSiDeRaTa	Dynamic Scalable Dependable Real-Time systems
15415	DH-CHAP	Diffie-Hellman augmented CHAP
15416	DHCP	Dynamic Host Control Protocol
15417	DHS	Department of Homeland Security
15418	DIA	Defense Intelligence Agency
15419	DIACAP	DoD Information Assurance Policy for IA Certification and Accreditation
15420	DII	Defense Information Infrastructure
15421	DIO	Defensive Information Operations
15422	DISA	Defense Information Systems Agency
15423	DISN	Defense Information Systems Network
15424	DITSCAP	DoD Information Technology Security Certification and Accreditation
15425		Process
15426	DMDC	Defense Manpower Data Center
15427	DME	Distributed Management Environment
15428	DMI	Desktop Management Interface
15429	DMS	Defense Message System
15430	DMTF	Distributed Management Task Force
15431	DN	Distinguished Name
15432	DNS	Domain Name System
15433	DoD	Department of Defense

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15434	DoDI	Department of Defense Instruction
15435	DoS	Denial of Service
15436	DPM	Digital Policy Management
15437	DR/COOP	Disaster Recovery and Continuous Operations
15438	DSA	Digital Signature Algorithm
15439	DSS	Digital Signature Service
15440	DTD	Document Type Definition
15441		Data Transfer Device
15442	dB μ A	decibels, micro-amps per meter
15443	EAL	Evaluation Assurance Level
15444	EAN.UCC	European Article Number, Uniform Code Council
15445	EAP	Extensible Authentication Protocol
15446	EAS	Electronic Article Surveillance
15447	EIAU	End Information Assurance Unit
15448	ECC	Elliptic Curve Cryptography
15449		Error Correcting Code
15450	ECDSA	Elliptic Curve Digital Signature Algorithm
15451	ECU	End Cryptographic Unit
15452	EIAU	End Information Assurance Unit
15453	EIGRP	Enhanced Interior Gateway Routing Protocol
15454	EKMS	Electronic Key Management System
15455	ELF	Extended Log Format
15456	eMASS	Enterprise Mission Assurance Support System
15457	EMSEC	Emission Security
15458	ENUM	Electronic Numbering
15459	EOTN	Encrypted Optical Transport Network

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15460	EPC	Electronic Product Code
15461	ESG	Enterprise Wide Sensor Grid
15462	ESM/NM	Enterprise Service Management/Network Management
15463	ESP	Encapsulating Security Payload
15464	ESS	Enhanced Security Services
15465	ETSI	European Technical Standards Institute
15466	FAQ	Frequently Asked Questions
15467	FAR	False Acceptance Rate
15468	FC	Fibre Channel
15469	FCAPS	Fault, Configuration, Accounting, Performance, and Security
15470	FC-GS-3	Fibre Channel–Generic Services–3
15471	FCIP	Fibre Channel over TCP/IP (RFC 3821)
15472	FCsec	Fibre Channel Security
15473	FC-SP	Fibre Channel–Security Protocol
15474	FFRDC	Federally Funded Research and Development Center
15475	FIPS	Federal Information Processing Standards
15476	FIRE	Flexible Intra-AS Routing Environment
15477	FISMA	Federal Information Security Management Act
15478	FiXs	Federated Identity Cross-credentialing System
15479	FMR	False Match Rate
15480	FNBDT	Future Narrow Band Digital Terminal
15481	FNMR	False Non-Match Rate
15482	FOUO	For Official Use Only
15483	FPKI	Federal Public Key Infrastructure
15484	FRR	False Rejection Rate

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15485	FTP	File Transfer Protocol
15486	Gbps	Giga bits per second
15487	GCCS	Global Command and Control System
15488	GCP	Gateway Control Protocol
15489	GDS	Global Directory Services
15490	GES	Global Enterprise Service
15491	GIAI	Global Individual Asset Identifier
15492	GID	General Identifier
15493	GIG	Global Information Grid
15494	GIG-BE	Global Information Grid–Bandwidth Expansion
15495	GIG ES	Global Information Grid Enterprise Services
15496	GLN	Global Location Number
15497	GOTS	Government Off-The-Shelf
15498	GRAI	Global Returnable Asset Identifier
15499	GSM	Global System for Mobile (communication)
15500	GSS	Generic Security Services
15501	GTC	Generic Token Card
15502	GTIN	Global Trade Item Number
15503	GUI	Graphical User Interface
15504	GULS	Generic Upper Layer Security
15505	GW	GateWay
15506	HAIPE	High Assurance Internet Protocol Encryptor
15507	HAIPIIS	High Assurance Internet Protocol Interoperability Specification
15508	HAPKI	High Assurance Public Key Infrastructure
15509	HBA	Host Bus Adapter

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15510	HI	Horizontal Integration
15511	HIDS	Host-Based Intrusion Detection System
15512	HIPPA	Health Information Protection and Privacy Act
15513	HIPS	Host-Based Intrusion Prevention System
15514	HLS	Home Land Security
15515	HMAC	Keyed-Hashing for Message Authentication
15516	HMAC-MD5	Hashed Message Authentication Code-Message Digest Algorithm 5
15517	HMMA	Hypermedia Management Architecture
15518	HNMP	Hierarchical Network Management Protocol
15519	HNMS	Hierarchical Network Management System
15520	HRS	Human Recognition Services
15521	HSM	Hardware Security Module
15522	HTML	HyperText Markup Language
15523	HTTP	Hypertext Transfer Protocol
15524	I&A	Identification and Authentication
15525	IA	Information Assurance
15526	IAA SPO	Information Assurance Architecture Special Program Office
15527	IAC	Information Assurance Component
15528	IAD	Information Assurance Directorate
15529	IANA	Internet Assigned Numbers Authority
15530	IATF	Information Assurance Task Force
15531	IAVA	Information Assurance Vulnerability Alert
15532	IAVM	Information Assurance Vulnerability Management
15533	I&W	Indications and Warnings
15534	IB	In Band

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15535	IC	Intelligence Community
15536	ICMP	Internet Control Message Protocol
15537	IDC	International Data Corporation
15538	ICSIS	Intelligence Community System for Information Sharing
15539	IDIP	Intrusion Detection and Isolation Protocol
15540	IDMEF	Intrusion Detection Message Exchange Format
15541	IDS	Intrusion Detection System
15542	IDU	Interface Data Unit
15543	IDUP	Independent Data Unit Protection
15544	IDWG	Intrusion Detection Working Group
15545	IDXP	Intrusion Detection eXchange Protocol
15546	IdM	Identification Management
15547	IEC	International Electrotechnical Commission
15548	IEEE	Institute of Electrical and Electronics Engineers
15549	IESG	Internet Engineering Steering Group
15550	IETF	Internet Engineering Task Force
15551	IFF	Identification Friend-or-Foe
15552	IHMC	Interdisciplinary Study of Human & Machine Cognition
15553	IIA SPO	Information Assurance Architecture Special Program Office
15554	IKE	Internet Key Exchange
15555	IMAP	Internet Message Access Protocol
15556	INE	In-line Network Encryptor
15557	INFOCON	Information Operations Condition
15558	INFOSEC	Information Security
15559	INDEF	Incident Object Description Exchange Format

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15560	IOC	Initial Operational Capability
15561	IP	Internet Protocol
15562	IPM	Information Policy Management
15563	IPS	Intrusion Prevention System
15564	IPsec	Internet Protocol Security (IP Security)
15565	IPSRA	Internet Protocol Security Remote Access
15566	IPT	Integrated Product Team
15567	IPv4	IP version 4
15568	IPv6	IP version 6
15569	IPX	Internetwork Packet Exchange
15570	IRM	Information Resources Management
15571	ISAKMP	Internet Security Association and Key Management Protocol
15572	ISDN	Integrated Services Digital Network
15573	IS-IS	Intermediate System-to-Intermediate System
15574	ISM	Information Security Markings
15575	ISSE	Imagery Support Server Environment
15576	ISO	International Organization for Standardization
15577	ISP	Internet Service Provider
15578	IT	Information Technology
15579	ITSEC	Information Technology Security Evaluation Criteria
15580	ITU	International Telecommunications Union
15581	ITU-T	International Telecommunication Union Telecommunication
15582		Standardization Sector
15583	IV	Initialization Vector
15584	iFCP	Internet Fiber Channel Protocol (Internet Draft)

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15585	iSCSI	Internet SCSI (RFC 3720)
15586	iSNS	Internet Storage Name Service
15587	JAS	Java Agent Services
15588	JTF	Joint Task Force
15589	JTRS	Joint Tactical Radio System
15590	JV2020	Joint Vision 2020
15591	JWICS	Joint Worldwide Intelligence Communication System
15592	KAoS	Knowledgeable Agent-oriented System
15593	KDC	Key Distribution Center
15594	KEK	Key Encryption Key
15595	KMI	Key Management Infrastructure
15596	KVM	Keyboard, Video, Mouse
15597	KMP	Key Management Policy
15598	KMPS	Key Management Policy Server
15599	KMS	Key Management System
15600	KRSS	Key Registration Service Specification
15601	L2TP	Layer 2 Tunneling Protocol
15602	LAN	Local Area Network
15603	LBAC	List Based Access Control
15604	LCD	Liquid Crystal Display
15605	LDAP	Lightweight Directory Access Protocol
15606	LEAP	Lightweight Extensible Authentication Protocol
15607	LIMFAC	List Limiting Factors
15608	LPD	Low Probability of Detection
15609	LPI	Low Probability of Interception

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15610	LRS	Linear Recursive Sequence
15611	LSP	Labeled Switch Path
15612	LTANS	Long-Term Archive and Notary Services
15613	LTS	Laboratory for Telecommunications Science
15614	M&C	Management and Control
15615	MAC	Mandatory Access Control
15616		Media Access Control
15617		Message Authentication Code
15618		Mission Assurance Category
15619	MANET	Mobile Ad hoc Network
15620	MAPKI	Medium Assurance Public Key Infrastructure
15621	MC	Multipoint Controller
15622	MCU	Multipoint Control Unit
15623	MD5	Message Digest Algorithm 5
15624	MEGACO	MEDIA GAteway COntrol protocol
15625	MELP	Mixed Excitation Linear Prediction
15626	MER	Minimum Essential Requirement
15627	MG	Media Gateway
15628	MGCP	Media Gateway Control Protocol
15629	MIB	Management Information Base
15630	MID	Message Identifier
15631	MIB	Management Information Base
15632	MILS	Multiple Independent Levels of Security
15633	MIME	Multipurpose Internet Mail Extensions
15634	MISSI	Multilevel Information Systems Security Initiative
15635	MITM	Man in the Middle

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15636	MKI	Master Key Identifier
15637	MLPP	Multi-Level Precedence and Preemption
15638	MLS	Multi-Level Secure
15639		Multiple Levels of Security
15640	MLTC	Multi-Level Thin Client
15641	MNIS	Multi-National Information Sharing
15642	MOSS	MIME Object Security Services
15643	MP	Multipoint Processor
15644	MPL	Mozilla Public License
15645	MPLS	Multi-Protocol Label Switching
15646	MQV	Menezes-Qu-Vanstone
15647	MR	Modem Relay
15648	MS-CHAP	Microsoft Challenge Handshake Authentication Protocol
15649	MS&MS	Mission Support and Management Systems
15650	MSE	Mobile Subscriber Equipment
15651	MSGFMT	Message Format
15652	MSI	Microsoft Installer
15653	MSL	Multiple Single Levels
15654	MSLS	Multiple Single Levels of Security
15655	MSP	Message Security Protocol
15656		Metadata Standard for Publication
15657	MTA	Mail Transfer Agent
15658	MW	milli-Watt
15659	Mbps	Megabits per second
15660	NAC	Network Admission Control
15661	NAS	Network Attached Storage

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15662	NASA	National Aeronautics and Space Administration
15663	NATO	North Atlantic Treaty Organization
15664	NCES	Network-Centric Enterprise Services
15665	NCOW	Network-Centric Operations and Warfare
15666	NCOW-RM	Net-Centric Operations and Warfare-Reference Model
15667	NCW	Net-Centric Warfare
15668	NDA	National Distribution Authorities
15669	NDS	Novell Directory Service
15670	NETOPS	Network Operations
15671	NFS	Network File System
15672	NGSCB	Next Generation Secure Computing Base
15673	NIAP	National Information Assurance Partnership
15674	NIDS	Network-Based Intrusion Detection System
15675	NII	Networks and Information Integration
15676	NIPRNet	Non-secure Internet Protocol Router Network
15677	NIPS	Network-Based Intrusion Protection System
15678	NISPOM	National Industrial Security Program Operating Manual
15679	NIST	National Institute of Standards and Technology
15680	NMCC	National Military Command Center
15681	NNIDS	Network Node Intrusion Detection System
15682	NNTP	Network News Transport Protocol
15683	NOC	Network Operations Center
15684	NP	Network Processor
15685	NSA	National Security Agency
15686	NTP	Network Time Protocol

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15687	OASIS	Organization for the Advancement of Structured Information Standards
15688	OCSP	Online Certificate Status Protocol
15689	OED	OSIS Evolutionary Development
15690	OEM	Original Equipment Manufacturer
15691	OLAC	Object Level Access Control
15692	ONR	Office of Naval Research
15693	OOB	Out of Band
15694	OPIE	One-time Password in Everything
15695	OPS	Optivity Policy Services
15696	OPSEC	Operations Security
15697	OS	Operating System
15698	OSD	Office Secretary of Defense
15699	OSF	Open Software Foundation
15700	OSI	Open Systems Interconnection
15701	OSIS	Open System Information System
15702	OSPF	Open Shortest Path First
15703	OTAD	Over-The-Air Distribution
15704	OTAR	Over-The-Air Rekey
15705	OTAT	Over-The-Air Transfer
15706	OTNK	Over the Network Keying
15707	OTP	One Time Password
15708	OUSPG	Oulu University Secure Programming Group
15709	OVAL	Open Vulnerability Assessment Language
15710	OWL	Web Ontology Language
15711	PAC	Positive Access Control

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15712	PANA	Protocol for carrying Authentication for Network Access
15713	PBR	Policy-Based Routing
15714	PBX	Private Branch Exchange
15715	PC	Personal Computer
15716	PCAP	Packet Capture
15717	PC/SC	Personal Computer/Smart Card
15718	PCI	Protocol Control Information
15719	PDA	Personal Digital Assistant
15720	PDP	Policy Decision Point
15721	PDU	Protocol Data Unit
15722	PEAP	Protected Extensible Authentication Protocol
15723	PEM	Privacy Enhanced Mail
15724	PEP	Policy Enforcement Point
15725	PGP	Pretty Good Privacy
15726	PIC	Pre-IKE Credential Provisioning
15727	PIN	Personal Identification Number
15728	PIP	Policy Input Point
15729	PK	Public Key
15730	PKC	Public Key Certificate
15731	PKCS	Public Key Cryptographic Standard
15732	PKI	Public Key Infrastructure
15733	PKINIT	Public Key Initialization Authentication
15734	PKIX	Public Key Infrastructure X.509
15735	PKCS	Public Key Cryptography Standards
15736	PMI	Privilege Management Infrastructure

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15737	POC	Point of Contact
15738	POD	Proof of Delivery
15739	POM	Program Objective Memorandum
15740	POP	Point of Presence
15741	POP3	Post Office Protocol
15742	PoS	Priority of Service
15743	POTS	Plain Old Telephone System
15744	PP	Protection Profiles
15745	PPK	Pre-Placed Key
15746	PPP	Point-to-Point Protocol
15747	PRBAC	Partition Rule-Based Access Control
15748	PRSN	Primary Services Node
15749	PSEQN	Payload Sequence Number
15750	PSN	Product Services Node
15751	PST	Provision Service Target
15752	PSTN	Public Switched Telephone Network
15753	PXE	Preboot eXecution Environment
15754	QoP	Quality of Protection
15755	QoS	Quality of Service
15756	RA	Registration Authority
15757		Response Action
15758	RAdAC	Risk Adaptable Access Control
15759	RADIUS	Remote Access Dial In User Service
15760	RAID	Recent Advances in Intrusion Detection
15761	RBAC	Role-Based Access Control

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15762	RCD	Reference Capability Document
15763	RDEP	Remote Data Exchange Protocol
15764	RDF	Resource Description Framework
15765	RDFS	Resource Description Framework Schema
15766	RF	Radio Frequency
15767	RFC	Request for Comments
15768	RFID	Radio Frequency Identification
15769	RFP	Request for Proposal
15770	RIP	Routing Information Protocol
15771	RM	Radiant Mercury
15772	ROI	Return on Investment
15773	RPC	Remote Procedure Call
15774	RPSLNg	Routing Policy Specification Language Next Generation
15775	RR	Receiver Report
15776	RREP	Route Reply
15777	RREQ	Route REQuest
15778	RSA	Rivest Shamir Adelman (public key encryption algorithm)
15779	RSVP	ReSerVation Protocol
15780	RTCP	Real Time Control Protocol
15781	RTP	Real Time Protocol
15782	RVN	Red Virtual Network
15783	rDSA	Reversible Public Key Cryptography for Digital Signatures
15784	S/Key	Shared Key
15785	SA	Security Association
15786		Situational Awareness

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15787	SACRED	Securely Available Credentials
15788	SAD	Security Association Database
15789	SAMI	Source and Method Information
15790	SAML	Security Assertion Markup Language
15791	SAN	Storage Area Network
15792	SAODV	Secure Ad hoc On Demand Distance Vector
15793	SAP	Service Access Point
15794	SAR	Security Aware ad-hoc Routing
15795	SASL	Simple Authentication Security Layer
15796	SBSM	Session-Based Security Model
15797	SBU	Sensitive But Unclassified
15798	SCA	Subordinate Certificate Authority
15799	SCEP	Simple Certificate Enrollment Protocol
15800	SCI	Sensitive Compartmented Information
15801	SCIF	Sensitive Compartmented Information Facility
15802	SCP	Secure Copy
15803	SCSI	Small Computer System Interface
15804	SDNS	Secure Data Network System
15805	SEI	Software Engineering Institute
15806	SEM	Security Event Management
15807	SEQN	Sequence Number
15808	SESA	Symantec Enterprise Security Architecture
15809	SESE	Security Exchange Service Element
15810	SHA	Secure Hash Algorithm
15811	SHF	Super High Frequency

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15812	S-HTTP	Secure HTTP
15813	SIF	System Impact Factor
15814	SIGINT	Signals Intelligence
15815	SIM	Subscriber Identity Module
15816	SIP	Session Initiation Protocol
15817	SIPRNet	Secret Internet Protocol Router Network
15818	SISWG	Security in Storage Working Group (IEEE)
15819	SLA	Service Level Agreement
15820	SLP	Service Location Protocol (RFC 2608)
15821	SMBIOS	Systems Management Basic Input/Output System
15822	SMI	Security Management Infrastructure
15823		Security Management Information
15824	SMI-S	Storage Management Initiative - Specification
15825	S/MIME	Secure Multipurpose Internet Mail Extensions
15826	SMS	System Management Server
15827	SMTP	Simple Mail Transfer Protocol
15828	SMUX	SNMP Multiplex
15829	SMW	Security Management Workstation
15830	SNIA	Storage Network Industry Association
15831	SNMP	Simple Network Management Protocol
15832	SOA	Source of Authority
15833	SOAP	Simple Object Access Protocol
15834	SOC	Security Operations Center
15835	SoM	Strength of Mechanism
15836	SONET	Synchronous Optical NETwork

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15837	SPAWAR	Space and Naval Warfare Systems Command
15838	SPD	Security Policy Database
15839	SPF	Shortest Path First
15840	SPI	Service Provider Interface
15841		Security Parameters Index
15842	SPIE	Source Path Isolation Engine
15843	SPIF	Security Policy Information File
15844	SPKI	Simple Public Key Infrastructure
15845	SPKM	Simple Public-Key GSS-API Mechanism
15846	SPML	Services Provisioning Markup Language
15847	SPRT	Simple Packet Relay Transport
15848	SR	Sender Report
15849	SRD	Short Range Device
15850	SRTCP	Secure Real Time Control Protocol
15851	S RTP	Secure Real Time Protocol
15852	SSCC	Serial Shipping Container Code
15853	SSE	State Signaling Events
15854	SSH	Secure Shell
15855	SSL	Secure Session Layer
15856		Secure Sockets Layer
15857	SSO	Single Sign-On
15858	SSP	Secure Server Protocol
15859	SSRC	Synchronization Source Real-time Content
15860	SSTC	Security Services Technical Committee
15861	STE	Secure Teleconferencing Equipment
15862	STS	Security Token Service

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15863	STU	Secure Telecommunications Unit
15864	SVoIP	Secure Voice over Internet Protocol
15865	SWRL	Semantic Web Rule Language
15866	TA	Traffic Analysis
15867	TAMP	Trust Anchor Management Protocol
15868	TC	Transformational Communications
15869	TCG	Trusted Computing Group
15870	TCM	Transformational Communications MILSATCOM
15871	TCP	Transmission Control Protocol
15872	TCP/IP	Transmission Control Protocol/Internet Protocol
15873	TCSEC	Trusted System Evaluation Criteria
15874	TEK	Traffic Encryption Key
15875	TFS	Traffic Flow Security
15876	TFTP	Trivial File Transfer Protocol
15877	TGS	Trusted Gateway Solution
15878	TGT	Ticket Granting Ticket
15879	TLS	Transport Layer Security
15880	TMF	TeleManagement Forum
15881	TN	Traffic Normalizer
15882	TPED	Task, Process, Exploit, and Disseminate
15883	TPM	Trusted Platform Module
15884	TPPU	Task, Post, Process, and Use
15885	TRANSEC	Transmission Security
15886	TRL	Technology Readiness Level
15887	TSAT	Transformational Satellite

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15888	TSP	Time-Stamp Protocol
15889	TVA	Topological Vulnerability Analysis
15890	TV-1	Technical View current state
15891	TV-2	Technical View future state
15892	U	Unclassified
15893	UDOP	User Defined Operational Picture
15894	UDP	Universal Datagram Protocol
15895	UMBC	University of Maryland Baltimore County
15896	UMTS	Universal Mobile Telecommunications System
15897	UPN	User Personalized Network
15898	URI	Universal Resource Identifier
15899	USB	Universal Serial Bus
15900	USD/ATL	Undersecretary of Defense for Acquisitions, Technology and Logistics
15901	USM	User-based Security Model
15902	USSTRATCOM	U.S. Strategic Command
15903	UUID	Universal Unique ID
15904	VACM	View-based Access Control Model
15905	VI	Vulnerability Index
15906	VKB	Virtual Knowledge Base
15907	VLAN	Virtual Local Area Network
15908	VM	Virtual Machine
15909	VoIP	Voice over IP
15910	VoSIP	Voice over Secure IP
15911	VPA	Virtual Port-based Authentication
15912	VPN	Virtual Private Network

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15913	W3C	World Wide Web Consortium
15914	WAM	Web Access Management
15915	WAN	Wide Area Network
15916	WAP	Wireless Application Protocol
15917	WBEM	Web-Based Enterprise Management
15918	WebDAV	Web Distributed Authoring and Versioning
15919	WebDAV-AC	Web Distributed Authoring and Versioning-Access Control
15920	WG	Working Group
15921	WIN-T	Warfighter Information Network-Tactical
15922	WLAN	Wireless Local Area Network
15923	WNW	Wideband Networking Waveform
15924	WPA	Wi-Fi Protected Access
15925	WS	Web Services
15926	WSDL	Web Services Description Language
15927	WSF	Web Services Framework
15928	WS-I	Web Services Interoperability
15929	WSS	Web Services Security
15930	WS-Trust	Web Services–Trust Language
15931	WTLS	Wireless Transport Layer Security
15932	WXS	W3C XML Schema
15933	XACML	eXtensible Access Control Markup Language
15934	XCMS	XML Cryptographic Message Syntax
15935	XER	XML Encoding Rules
15936	X-KISS	XML Key Information Service Specification
15937	XKMS	XML Key Management Specification

UNCLASSIFIED//FOR OFFICIAL USE ONLY

15938	X-KRSS	XML Key Registration Service Specification
15939	XML	Extensible Mark-up Language
15940	XML_DSIG	XML Digital Signature
15941	XML_ENC	XML Encryption
15942	XrML	eXtensible Rights Markup Language

15943

15944

15945

15946

15947

15948

15949

15950

15951

15952

15953

15954

15955

15956

(U) Appendices

15957
15958
15959
15960
15961
15962

15963

(U//FOUO) APPENDIX A: MAPPING OF TECHNOLOGIES TO IA SYSTEM ENABLERS

(U) The following Table lists the detailed technologies explored in this document. Each is mapped to the Technology Category under which it is discussed and the IA System Enabler section where it can be located in the document. The goal is for this to help readers locate the technologies in which they are interested.

Table A-1: (U//FOUO) Mapping of Technologies to IA System Enablers

This Table is (U//FOUO)		
Technology Category	Detailed Technology	IA System Enabler
Authentication Tokens	Asynchronous Synchronous Time-driven Event-driven	2.1 Identification and Authentication
Biometrics	Physiological Fingerprint Face Recognition Iris Recognition Hand and Finger Geometry Behavioral Signature Verification Speech Recognition	2.1 Identification and Authentication
Device/Service authentication	Strong Authentication for Devices	2.1 Identification and Authentication
Authentication protocols	802.1x for Network Applications 802.1x for Device Authentication Manufacturing Time Device Credentials Web Service Protocol for Business-Application Integration Application Connectors and Authentication Clients Credential Provisioning and Validation	2.1 Identification and Authentication
Single Sign-On	Early SSO Techniques Scripting Password Synchronization LDAP Directories SSO Architectures Centralized Model Federated Model Kerberos PKI Certificates SAML	2.1 Identification and Authentication

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)		
Technology Category	Detailed Technology	IA System Enabler
Authentication Confidence	Authentication Confidence	2.1 Identification and Authentication
Core RAdAC	Core RAdAC	2.2 Policy-Based Access Control
Assured Metadata	Metadata Language & Standards Trusted Metadata Creation Tools Crypto-binding of Metadata to Source Information Objects	2.2 Policy-Based Access Control
Digital Access Control Policy	Digital Access Control Policy	2.2 Policy-Based Access Control
Protecting Data-at-Rest	Cryptography Data Backup & Archive Data Destruction Labeling Periods Processing Physical Controls Quality of Protection	2.3 Protection of User Information
Protecting Data-in-Transit	Application Layer Technologies Non-Real-Time Data Technologies Traditional Application Security Session Security SSL/TLS GULS Web Services Security Real-Time Data Technologies FNBDT Interoperability/Gateways Secure VoIP RTP and RTCP Transport & Network Layer Technologies Non-Real-Time Data Technologies IP Layer Security TFS VPN Real-Time Data Technologies Secure VoIP Call Control Link & Physical Layer Technologies Anti-Jam Link Encryption TRANSEC	2.3 Protection of User Information

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)		
Technology Category	Detailed Technology	IA System Enabler
Trusted Platforms	Trusted Platforms	2.3 Protection of User Information
Trusted Applications	Trusted Applications	2.3 Protection of User Information
Cross Domain Solutions	Cross Domain Solutions	2.3 Protection of User Information
Non-Repudiation	Non-Repudiation	2.3 Protection of User Information
Development of Policies	Centralized vs. Distributed Elements of Policies Access Control Trust Anchors Policy Languages	2.4 Dynamic Policy Management
Distribution of Policies	Standard Protocols Security Issues	2.4 Dynamic Policy Management
Policy Management Architectures	Policy Directories	2.4 Dynamic Policy Management
IA Policy-based Routing	IA Policy-based Routing	2.5 Assured Resource Allocation
Operational-based Resource Allocation	Operational-based Resource Allocation	2.5 Assured Resource Allocation
Integrity of Network Fault Monitoring/Recovery and Integrity of Network Management & Control	Integrity of Network Fault Monitoring/Recovery and Integrity of Network Management & Control	2.5 Assured Resource Allocation
Protect Technologies	Protect Technologies	2.6 Network Defense and Situational Awareness
Deception Technologies	Honeypots Honeynets	2.6 Network Defense and Situational Awareness
Situational Awareness	UDOP NETOPS	2.6 Network Defense and Situational Awareness
Network Mapping	Network Mapping	2.6 Network Defense and Situational Awareness
IDS	Host-based IDS Network-based IDS	2.6 Network Defense and Situational Awareness
IPS	IPS	2.6 Network Defense and Situational Awareness
User Activity Profiling	User Activity Profiling	2.6 Network Defense and Situational Awareness
Cyber Attack Attribution	Hop-by-Hop Traceback Backscatter Traceback CenterTrack ICMP Traceback or iTrace Hash-based IP Traceback	2.6 Network Defense and Situational Awareness

This Table is (U//FOUO)		
Technology Category	Detailed Technology	IA System Enabler
Correlation Techniques	Correlation Techniques	2.6 Network Defense and Situational Awareness
CND Response Actions	CND Response Actions	2.6 Network Defense and Situational Awareness
Automated IAVA Patch Management	Automated IAVA Patch Management	2.6 Network Defense and Situational Awareness
Identity Management	Identity Management	2.7 Management of IA Mechanisms and Assets
Privilege Management	Rules-based Authorization Schemes Roles-based Authorization Schemes PMI	2.7 Management of IA Mechanisms and Assets
Key Management	Evolution of Key-based Equipment Technology KMI XML Key Management Services Constructive Key Management IKE and ISAKMP Hardware Security Module	2.7 Management of IA Mechanisms and Assets
Certificate Management	Certificate Management	2.7 Management of IA Mechanisms and Assets
Configuration Management of IA Devices and Software	Systems Management Applications Network Boot Applications Malware Management ECU Update Patch Management Systems	2.7 Management of IA Mechanisms and Assets
Inventory Management	Inventory Management	2.7 Management of IA Mechanisms and Assets
Compromise Management of IA Devices	Compromise Management of IA Devices	2.7 Management of IA Mechanisms and Assets
Audit Management	Audit Management	2.7 Management of IA Mechanisms and Assets
The Table is (U//FOUO)		

15964

15965
15966
15967
15968
15969
15970
15971
15972
15973
15974
15975
15976
15977

(U//FOUO) APPENDIX B: TV-1 FOR IA

(U) The DoD Architecture Framework (DoDAF) provides a convenient repository for describing some of the content from the Roadmap, but there is no “system” for which a system architecture is being described. From DoDAF: “The TV includes a collection of the technical standards, implementation conventions, standards options, rules, and criteria organized into profile(s) that govern systems and system elements for a given architecture.” The set of standards approved to support the existing capability (as-is) includes those standards listed in the DoD IT Standards Registry (DISR) Baseline Release 04-1.0. Table B-1: Technical Standards Profile identifies standards that apply to systems view elements. The standards in this table are a summary of the standards identified in the Section 3—existing standards identified as needed to satisfy capabilities listed in the GIG IA Reference Capabilities Document (RCD).

Table B-1: (U//FOUO) TV-1 for IA

This Table is (U//FOUO)	
Name	Description
PKCS #11	Cryptographic Token Interface (cryptoki) Standard (specification of an application programming interface API for cryptographic token devices)
PKCS #12	Personal Information Exchange Syntax (specifies transfer syntax for personal identity information such as private keys and certificates, etc.)
PKCS #15	Cryptographic Token Information Format Standard (ensures interoperability of multiple vendor implementations)
CAPI	Cryptographic Application Programming Interface standards
PC/SC Workgroup Specifications 1.0	Interoperability Specs for Smart Cards and PCs (platform and OS independent)
PC/SC Workgroup Specifications 2.0	Updated enhancements, including contactless (wireless RF) cards
ISO/IEC 7810	Identification Cards – physical characteristics
ISO/IEC 7811	ID Cards – Recording techniques
ISO/IEC 7812	ID Cards – Identification of issuers
ISO/IEC 7813	Financial transaction cards
ISO/IEC 7816	ID Cards with contacts
ISO/IEC 10373	ID Cards – Test Methods
ISO/IEC 10536	Contactless ID Cards – Close Coupled
ISO/IEC 14443	Contactless ID Cards – Proximity (Mifare cards) - 1-inch range
ISO/IEC 15693	Contactless ID Cards – Vicinity (ICODE cards) - 5-inch range
Common Biometric Exchange Formats Framework (CBEFF)	CBEFF originally stood for Common Biometric Exchange File Format and was originally developed by the Biometric Consortium (BC). It was published by NIST as NISTR 6529. CBEFF defines a standard method for identifying and carrying biometric data. It describes a framework for defining data formats that facilitate the communication of biometric data. CBEFF does not specify the actual encoding of data (e.g., bits on a wire) but provides rules and requirements and the structure for defining those explicit data format specifications.
BioAPI	The BioAPI standard defines an Application Program Interface (API) and a Service Provider Interface (SPI) for standardizing the interaction between biometric-enabled applications and biometric sensor devices. The API provides a common method for applications to access biometric authentication technology without requiring application

This Table is (U//FOUO)	
Name	Description
	<p>developers to have biometric expertise. The SPI allows the production of multiple BSPs (Biometric Service Providers) that may be used by an application without modification of that application, regardless of biometric technology.</p> <p>The BioAPI Consortium originally developed the BioAPI specification. The BioAPI Consortium is a group of over 50 organizations focused solely on furthering a standard biometric API. M1 has taken the resulting specification from the consortium and standardized it nationally as ANSI INCITS 358-2002. M1 has also contributed ANSI INCITS 358-2002 to SC 37 where it is currently a draft international standard.</p>
Data Interchange Formats	<p>A data interchange format specifies the low-level format for storing, recording, and transmitting biometric information. This biometric information may be unique to each biometric characteristic (e.g., fingerprint, iris, signature) and/or to each method of capture (e.g., photograph, capacitive sensor). In some technologies, this biometric information is called a template. M1.3 is currently working on projects dedicated to standards for the following formats.</p>
Biometric Profiles	<p>A biometric profile identifies a set of base biometric standards that apply to a single application or scenario. The profile then identifies the appropriate configurations, parameters, and choices for options provided within those specifications. The goal is to provide interoperability and consistent functionality and security across a defined environment.</p> <p>M1.4 is engaged in the following projects:</p> <p>Interoperability and Data Interchange—Biometric Based Verification and Identification of Transportation Workers</p> <p>Interoperability, Data Interchange and Data Integrity—Biometric Based Personal Identification for Border Management</p> <p>Point-of-Sale Biometric Verification/Identification</p> <p>SC 37 has defined a functional architecture that serves as part one of a multi-part standard. SC 37 is also working on the first profile of the standard titled Biometric Profile for Employees.</p>
Biometric Evaluation Methodology	<p>The Biometric Evaluation Methodology (BEM), Version 1.0, was designed to aid security evaluators who were attempting to evaluate biometric products against the Common Criteria (CC). The Common Evaluation Methodology (CEM) used in CC evaluations does not address the environmental, user population, and other issues that have an impact on a biometric implementation. The BEM specifically addresses these issues as they apply to biometric technology evaluations under the CC.</p> <p>Evaluators, certifiers and developers from Canada, U.K., GERMANY, U.S., Italy, Sweden, and others developed the BEM. Version 1.0 of BEM was released in August of 2002.</p>
Biometrics Protection Profile	<p>The CC is an effort of the US, Canada, and European countries to establish a common set of security criteria by which to evaluate IT products. This effort has resulted in an international standard (ISO/IEC 15408-1) for evaluating IT security products. The document that establishes the implementation-independent security requirements for a given category of product is called a Protection Profile. Currently, the DoD Biometrics Management Office (BMO) and the National Security Agency (NSA) are developing four Protection Profiles for biometrics products:</p> <p>Robustness Biometric PP for Verification Mode</p> <p>Basic Robustness Biometric PP for Verification Mode</p> <p>Medium Robustness Biometric PP for Identification Mode</p> <p>Basic Robustness Biometric PP for Identification Mode</p>

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
Biometric API for JavaCard	The JavaCard Forum was established in 1997 to promote Java as the preferred programming language for multiple-application smart cards. A subset of the Java programming language was proposed for these cards and resulted in a standard for a JavaCard API. The JavaCard Forum has extended the JavaCard API to enroll and manage biometric data securely and facilitate a match on card capability with the Biometric API for JavaCard. The Biometric API manages templates, which are stored only in the card. During a match process, no sensitive information is sent off the card.
Common Data Security Architecture (CDSA), Human Recognition Services Module	The Human Recognition Services Module (HRS) is an extension of the Open Group's Common Data Security Architecture (CDSA). CDSA is a set of layered security services and a cryptographic framework that provides the infrastructure for creating cross-platform, interoperable, security-enabled applications for client-server environments. The biometric component of the CDSA's HRS is used in conjunction with other security modules (i.e., cryptographic, digital certificates, and data libraries) and is compatible with the BioAPI specification and CBEFF.
RFC 2413	Dublin Core Metadata For Resource Discovery
RFC 821	Simple Mail Transfer Protocol
RFC 822	Standard for the Format of ARPA Internet Text Messages
RFC 1421	Privacy Enhancement for Internet Electronic Mail: Part I: Message Encryption and Authentication Procedures
RFC 1422	Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management
RFC 1423	Privacy Enhancement for Internet Electronic Mail: Part III: Algorithms, Modes, and Identifiers
RFC 1424	Privacy Enhancement for Internet Electronic Mail: Part IV: Key Certification and Related Services
RFC 1848	MIME Object Security Services
RFC 3852	Cryptographic Message Syntax (CMS)
RFC 3851	S/MIME v3.1 Message Specification
RFC 3850	S/MIME v3.1 Certificate Handling
RFC 2634	Enhanced Security Services for S/MIME
RFC 3854	Securing X.400 Content with S/MIME
RFC 3855	Transporting S/MIME Objects in X.400
RFC 3370	CMS Algorithms
RFC 2797	Certificate Management Messages over CMS
RFC 2616	Hypertext Transfer Protocol -- HTTP/1.1
RFC 2617	HTTP Authentication: Basic and Digest Access Authentication
RFC 2660	The Secure HyperText Transfer Protocol
RFC 2518	HTTP Extensions for Distributed Authoring -- WEBDAV
RFC 3744	WebDAV Access Control Protocol
RFC 2222	Simple Authentication and Security Layer (SASL)
RFC 2444	The One-Time-Password SASL Mechanism
RFC 2554	SMTP Service Extension for Authentication
RFC 1939	Post Office Protocol - Version 3
RFC 2449	POP3 Extension Mechanism

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
RFC 1734	POP3 AUTHentication command
RFC 3206	The SYS and AUTH POP Response Codes
RFC 3501	Internet Message Access Protocol (IMAP) - Version 4rev1
RFC 2195	IMAP/POP AUTHorize Extension for Simple Challenge/Response
RFC 1731	IMAP4 Authentication Mechanisms
RFC 2086	IMAP4 ACL extension
RFC 2228	FTP Security Extensions
RFC 2244	Application Configuration Access Protocol
X.400	Information Technology – Message Handling Systems (MHS) – Message Handling System and Service Overview
X.402	Information Technology – Message Handling Systems (MHS) – Overall Architecture
X.411	Information Technology – Message Handling Systems (MHS) – Message transfer system: Abstract Service Definition and Procedures
SDN.701	Message Security Protocol
ACP 120	Common Security Protocol (CSP)
PKCS #7	Cryptographic Message Syntax Standard
RFC 2246	The TLS Protocol v1.0
RFC 2817	Upgrading to TLS Within HTTP/1.1
RFC 2818	HTTP Over TLS
RFC 3546	TLS Extensions
RFC 3268	AES Ciphersuites for TLS
RFC 2829	Authentication Methods for LDAP
RFC 2830	LDAPv3 Extension for TLS
RFC 3377	LDAP v3 Technical Specification
RFC 2595	Using TLS with IMAP, POP3 and ACAP
RFC 3207	SMTP Service Extension for Secure SMTP over TLS
ISO/IEC 11586-1	Information technology -- Open Systems Interconnection -- Generic upper layers security: Overview, models and notation
ISO/IEC 11586-2	Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) service definition
ISO/IEC 11586-3	Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) protocol specification
ISO/IEC 11586-4	Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax specification
ISO/IEC 11586-5	Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) Protocol Implementation Conformance Statement (PICS) proforma
ISO/IEC 11586-6	Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax Protocol Implementation Conformance Statement (PICS) proforma
ISO/IEC 7498-2	Data Communication Networks – Open Systems Interconnection (OSI) – Security, Structure and Applications – Security Architecture for Open Systems Interconnection for CCITT Applications

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
ISO/IEC 10745	Information Technology – Open Systems Interconnection – Upper Layers Security Model
CCITT X.800	Data Communication Networks – Open Systems Interconnection (OSI) – Security, Structure and Applications – Security Architecture for Open Systems Interconnection for CCITT Applications
ITU-T X.803	Information Technology – Open Systems Interconnection – Upper Layers Security Model
ITU-T X.830	Information technology -- Open Systems Interconnection -- Generic upper layers security: Overview, models and notation
ITU-T X.831	Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) service definition
ITU-T X.832	Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) protocol specification
ITU-T X.833	Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax specification
ITU-T X.834	Information technology -- Open Systems Interconnection -- Generic upper layers security: Security Exchange Service Element (SESE) Protocol Implementation Conformance Statement (PICS) proforma
ITU-T X.835	Information technology -- Open Systems Interconnection -- Generic upper layers security: Protecting transfer syntax Protocol Implementation Conformance Statement (PICS) proforma
[XML]	XML
	XML Schema
[XML-DSIG]	XML-DSIG
[XML-ENC]	XML-ENC
	XKMS
[SOAP]	SOAP
	WSDL
[SAML]	SAML
[XACML]	XACML
	UDDI
	SPML
	XCBF
	XCBF Token Profile
[WSS]	Web Services Security (WSS)
	WSS UsernameToken Profile
	WSS X.509 Certificate Token Profile
	Web Services Reliable Messaging
	ebXML Registry
	ebSOA
	WSDM
	XrML (eXtensible Rights Management Language)
	Web Application Security
	Digital Signature Services
	Security Services

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
	Web Services Distributed Management
[WSI-SEC]	Basic Security Profile Security Scenarios
	Basic Profile
	ANSI X9.84 (XCBF)
[XCMS]	ANSI X9.96 (XCMS)
	ANSI X9.73 (CMS)
	ITU-T X.509
	ISO 19092 (biometric formats)
[ID-FF]	ID-FF
[ID-SIS]	ID-SIS
[ID-WSF]	ID-WSF
	draft-lib-arch-soap-authn
[XML]	XML
	XML Schema
[XML-DSIG]	XML-DSIG
[XML-ENC]	XML-ENC
	XKMS
[SOAP]	SOAP
	WSDL
[SAML]	SAML
[XACML]	XACML
	UDDI
	SPML
	XCBF
	XCBF Token Profile
[WSS]	Web Services Security (WSS)
	WSS UsernameToken Profile
	WSS X.509 Certificate Token Profile
	Web Services Reliable Messaging
	ebXML Registry
	ebSOA
	WSDM
	XrML (eXtensible Rights Management Language)
	Web Application Security
	Digital Signature Services
	Security Services
	Web Services Distributed Management
[WSI-SEC]	Basic Security Profile Security Scenarios
	Basic Profile
	ANSI X9.84 (XCBF)

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
[XCMS]	ANSI X9.96 (XCMS)
	ANSI X9.73 (CMS)
	ITU-T X.509
	ISO 19092 (biometric formats)
[ID-FF]	ID-FF
[ID-SIS]	ID-SIS
[ID-WSF]	ID-WSF
	draft-lib-arch-soap-authn
FNBDT-210 (Signaling Plan)	This unclassified specification defines the signaling requirements for FNBDT operational modes. A secure overlay capable of interoperation with FNBDT compatible equipment on various similar or disparate networks is defined. Since the various networks will often have different lower-layer communications protocols, the FNBDT secure overlay specification specifies the higher-layer end-to-end protocols only. Appendices to this specification define operation using specific networks.
FNBDT-230 (Cryptography Specification)	This classified specification outlines details of the cryptography defined for FNBDT. Issues such as key generation, traffic encryption, and compromise recovery are specified in sufficient detail to allow interoperable implementation.
Proprietary extensions	The FNBDT signaling and cryptography specifications define interoperable branch points allowing vendors to implement proprietary modes. This allows vendors to take advantage of the basic FNBDT structure to add modes fulfilling specific needs. Legacy FNBDT implementations have used these branch points to implement custom cryptographic modes. Details of such modes are contained in vendor proprietary specifications.
Other specifications	Other interoperable FNBDT specifications have been suggested and are currently under consideration by the FNBDT Working Group. These additional documents would provide interoperable ways of implementing additional features such as non-Type 1 operation and key management.
FNBDT-210	Signaling Plan Revision 2.0
ITU V.150	Procedures for the end-to-end connection of V-series DCEs over and IP network
RFC 3550	RTP: A Transport Protocol for Real-Time Applications
RFC 3711	The Secure Real-time Transport Protocol (SRTP)
	Interoperability Specification For High Assurance Internet Protocol Encryptor (HAIPE) Devices
	Interoperability Specification For High Assurance Internet Protocol Encryptor (HAIPE) Devices
RFC-2401	Security Architecture for the Internet Protocol http://www.ietf.org/rfc/rfc2401.txt
	Security Architecture for the Internet Protocol http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2401bis-02.txt
RFC-2402	IP Authentication Header http://www.ietf.org/rfc/rfc2402.txt
	IP Authentication Header http://www.ietf.org/internet-drafts/draft-ietf-ipsec-rfc2402bis-07.txt
RFC-2406	IP Encapsulating Security Payload (ESP) http://www.ietf.org/rfc/rfc2406.txt

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
	IP Encapsulating Security Payload (ESP) http://www.ietf.org/internet-drafts/draft-ietf-ipsec-esp-v3-08.txt
H.235	Security and encryption for H-series multimedia terminals
H.245	Call Control Protocol for multimedia communication: Series H
H.323	Packet-based multimedia communications: Series H
H.510	Mobility for H.323 multimedia systems and services
H.530	Symmetric security procedures for H.323 mobility in H.510
RFC 3262	SIP: Session Initiation Protocol
RFC 3310	Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)
RFC 3313	Private Session Initiation Protocol (SIP) Extensions for Media Authorization
RFC 3323	A Privacy Mechanism for the Session Initiation Protocol (SIP)
RFC 3325	Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks
RFC 3329	Security Mechanism Agreement for the Session Initiation Protocol (SIP)
RFC 3435	Media Gateway Control Protocol
RFC 3525	Gateway Control Protocol
RFC 3761	The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)
RFC 3762	Telephone Number Mapping (ENUM) Service Registration for H.323
RFC 3853	S/MIME Advanced Encryption Standard (AES) Requirement for the Session Initiation Protocol (SIP)
ETSI ES 201 733	European Technical Standards Institute, "Electronic Signature Formats", 2000. Available at http://webapp.etsi.org/exchangefolder/es_201733v010103p.pdf
ISO 13888-1	International Standards Organization, "IT security techniques -- Non-repudiation -- Part 1: General", 2004
ISO 13888-2	International Standards Organization, "Information technology -- Security techniques -- Non-repudiation -- Part 2: Mechanisms using symmetric techniques", 1998
ISO 13888-3	International Standards Organization, "Information technology -- Security techniques -- Non-repudiation -- Part 3: Mechanisms using asymmetric techniques", 1997.
SDN.801	SDN.801 addresses concepts, tools and mechanisms for implementation of access control (AC). SDN.801 should be used to gain both a global understanding of MISSI access control, and as a guide for implementing access control features in MISSI-compliant components. SDN.801 is designed to advance from general concepts that introduce access control to more detailed information on access control tools, mechanisms, and processes as they apply to real-world communication systems.
ANSI INCITS 359-2004	This standard describes Role Based Access Control (RBAC) features that have achieved acceptance in the commercial marketplace. It includes a reference model and functional specifications for the RBAC features defined in the reference model. RBAC has become the predominant model for advanced access control because it reduces the complexity and cost of security administration in large networked applications. Many information technology vendors have incorporated RBAC into their product line, and the technology is finding applications in areas ranging from health care to defense, in addition to the mainstream commerce systems for which it was designed. The National Institute of Standards and Technology (NIST) initiated the development of the standard via the

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
	INCITS fast track process.
XACML 1.0	XACML is an XML-based language, or schema, designed specifically for creating policies and automating their use to control access to disparate devices and applications on a network.
RFC 3157	This document identifies a set of requirements for credential mobility. Using SACRED protocols, users will be able to securely move their credentials between different locations, different Internet devices, and different storage media as needed.
Extensible Access Control markup Language (XACML)	XACML provides fine-grained control of authorized activities, the effect of characteristics of the access requestor, the protocol over which the request is made, authorization based on classes of activities, and content introspection.
Routing Policy Specification Language (RPSL)	RPSL allows a network operator to be able to specify routing policies at various levels in the Internet hierarchy. Policies can be specified with sufficient detail in RPSL so that low-level router configurations can be generated from them. RPSL is extensible; new routing protocols and new protocol features can be introduced at any time.
Rei	A declarative policy language for describing policies over actions. It is possible to write Rei policies over ontologies in other semantic web languages.
KeyNote	KeyNote provides a simple language for describing and implementing security policies, trust relationships, and digitally signed credentials.
SDN.801	SDN.801 provides guidance for implementing access control concepts using both public key certificates and attribute certificates.
Security Assertion Markup Language (SAML)	SAML is an XML framework for exchanging authentication and authorization information.
Ponder	Ponder is a language for specifying management and security policies for distributed systems.
KAoS	KAoS policy services allow for the specification, management, conflict resolution, and enforcement of policies within domains.
LDAP	LDAP is an Internet protocol used to look up information from a LDAP server or directory. LDAP servers index all the data in their entries, and "filters" may be used to select just the information you want. "Permissions" and "authentications" can be set by the administrator to allow only certain people to access the LDAP database, and optionally keep certain data private. Reference http://www.ldap-directory.org/rfc-ldap for a list of LDAP RFCs.
File Transfer Protocol (FTP)	File Transfer Protocol (FTP), a standard Internet protocol, is the simplest way to exchange files between computers on the Internet. FTP is an application protocol that uses the Internet's TCP/IP protocols. Reference RFC959: http://www.w3.org/Protocols/rfc959/
Common Open Policy Service (COPS)	The Common Open Policy Service (COPS) protocol is a simple query and response protocol that can be used to exchange policy information between a policy server (PDP) and its clients (PEPs). Reference http://www.networksorcery.com/enp/protocol/cops.htm for a list of COPS related RFCs
Microsoft's SMS	SMS provides a solution for change and configuration management for the Microsoft platform, enabling organizations to provide relevant software and updates to users quickly and cost effectively.
Telnet	The Telnet program allows you to connect your PC to a server on the network using a

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
	username and password. You can then enter commands through the Telnet program, and they will be executed as if you were entering them directly on the server console.
SSL	SSL is designed to make use of TCP as a communication layer to provide a reliable end-to-end secure and authenticated connection between two points over a network.
TLS	RFC2246: The primary goal of the Transport Layer Security (TLS) Protocol is to provide privacy and data integrity between two communicating applications. The protocol is composed of two layers: the TLS Record Protocol and the TLS Handshake Protocol. At the lowest level, layered on top of some reliable transport protocol (e.g., TCP), is the TLS Record Protocol. The TLS Record Protocol provides connection security that provides confidentiality and integrity. TLS is designed as a successor to SSL and is sometimes called SSL V3.0.
IPsec	RFC 2401: Internet Protocol Security (generally shortened to IPsec) is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers at the IP layer. IPsec can be used to protect one or more data flows between IPsec peers.
X.500	X.500 is a CCITT protocol that is designed to build a distributed, global directory. It offers decentralized maintenance, searching capabilities, single global namespace, structured information framework, and a standards-based directory.
Finger, whois, domain name	These are very simple directory formats that are also in use.
RFC 2386	A Framework for QoS-Based Routing in the Internet
RFC 2676	QoS Routing Mechanisms and OSPF Extensions
SAML Core	E. Maler et al. Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML). OASIS, September 2003. Document ID oasis-sstc-saml-core-1.1. http://www.oasis-open.org/committees/security/ .
SAML Gloss	E. Maler et al. Glossary for the OASIS Security Assertion Markup Language (SAML). OASIS, September 2003. Document ID oasis-sstc-saml-glossary-1.1. http://www.oasis-open.org/committees/security/ .
SAMLSec	E. Maler et al. Security Considerations for the OASIS Security Assertion Markup Language (SAML), OASIS, September 2003, Document ID oasis-sstc-saml-sec-consider-1.1. http://www.oasis-open.org/committees/security/
SAMLReqs	Darren Platt et al., SAML Requirements and Use Cases, OASIS, April 2002, http://www.oasis-open.org/committees/security/ .
SAMLBind	E. Maler et al. Bindings and Profiles for the OASIS Security Assertion Markup Language (SAML). OASIS, September 2003. Document ID oasis-sstc-saml-bindings-1.1. http://www.oasis-open.org/committees/security/ .
SPML – Service Provisioning Markup Language	SPML is intended to facilitate the creation, modification, activation, suspension, and deletion of data on managed Provision Service Targets (PSTs). It is the only real standard of import that deals explicitly with the act of provisioning. Provisioning is a core component of Identity Management, but unfortunately most of the standards work has been in the direction of privilege management. http://www.oasis-open.org/committees/documents.php
SPML-Bind	OASIS Provisioning Services Technical Committee., SPML V1.0 Protocol Bindings, http://www.oasis-open.org/apps/org/workgroup/provision/download.php/1816/draft-pstc-bindings-03.doc , OASIS PS-
XACML – eXtensible Access Control Markup	From http://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
Language	
ID-FF	Identity Federation Framework Available from http://www.projectliberty.org/
ID WSF	Identity Web Service Framework Available from http://www.projectliberty.org/
ID SIS	Identity Services Interface Specifications Available from http://www.projectliberty.org/
RFC3281	S. Farrell, R. Housley, "An Internet Attribute Certificate Profile for Authorization", IETF RFC, April 2002
ISO/IEC 9594-8	ITU-T Rec. X.509 (2000) ISO/IEC 9594-8 The Directory: Authentication Framework
S/MIME	Ramsdell, B., "S/MIME Version 3 Message Specification", RFC2633, June 1999
MIME	Freed, N., Borenstein, N., "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.
CMS	Housley, R., "Cryptographic Message Syntax", RFC 3369, June 1999.
X9.69	Framework for Key Management Extensions. This standard defines specific key management methods for controlling and handling keys.
X9.73	Cryptographic Message Syntax (CMS) The Constructive Key Management technique (CKM), described in ANS X9.69, is used to encrypt objects. It may be used with CMS to encrypt a message (as the object) to a set of users sharing a common set of values (known as key components).
X9.42	Key Agreement of Symmetric Keys using Discrete Logarithm Cryptography.
X9.44	Key Establishment Using Factoring-Based Public Key Cryptography.
FIPS PUB 140-2 ANNEX D	Security Requirements for Cryptographic Modules Annex D: Approved Key Establishment Techniques Annex D provides a list of the FIPS Approved key establishment techniques applicable to FIPS PUB 140-2.
XKMS	XML Key Management Specification (XKMS) http://csrc.nist.gov/cryptval/140-2.htm
FIPS 171	Symmetric Key Establishment Techniques National Institute of Standards and Technology, <u>Key Management using ANSI X9.17</u> , Federal Information Processing Standards Publication 171, April 27, 1992. http://csrc.nist.gov/publications/fips/fips171/fips171.txt
EKMS 208	EKMS Key Distribution Functional Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 215	EKMS Communications Requirements Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 301	EKMS Types Dictionary Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 302	EKMS Key Distribution Data Standard. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
EKMS 311	EKMS ACCORDION 1.3 Length Indicator and Binding Code Specification. National Security Agency, Director, National Security Agency, Ft. George G. Meade MD. 20755-

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
	6734.
EKMS 603	Interface Specification for the Data Transfer Device AN/CYZ-10. National Security Agency, Director, National Security Agency, Ft. George G. Meade, MD. 20755-6734.
	[XAdES] J.C. Cruellas, G. Karlinger, K. Sankar XML Advanced Electronic Signatures; W3C Note 20 February, 2003 http://www.w3.org/TR/XAdES/
XML	Bray, T., Paoli, J., Sperberg-McQueen, C. M., Maler, E., "Extensible Markup Language (XML) 1.0 (Second Edition)," W3C Recommendation 6 October, 2000.
XMLENC	Eastlake, D., Reagle, J., Imamura, T., Dillaway, B., Simon, E., "XML Encryption Syntax and Processing," W3C Recommendation 10 December, 2002.
XMLSIG	Eastlake, D., Reagle, J., Solo D., "(Extensible Markup Language) XML-Signature Syntax and Processing," RFC 3075, March, 2002.
XMLSEC	Mactaggart, M., "Enabling XML Security: An introduction to XML encryption and XML signature," http://www-106.ibm.com/developerworks/xml/library/s-xmlsec.html/index.html .
	KMI-2200, dated July, 2004
DES	U.S. Data Encryption Standard (DES) in accordance with U.S. FIPS PUB 46-2 and ANSI X3.92
AES	U.S. Advanced Encryption Standard (AES) in accordance with U.S. FIPS PUB 197 (256-bit keys supported)
CAST block cipher	CAST block cipher in accordance with RFC 2144 (64-bit, 80-bit, and 128-bit variations are supported)
Triple-DES	Triple-DES in accordance with ANSI X9.52 (3-key variant for an effective key size of 168-bits is supported)
RC2®	RC2® in accordance with RFC 2268 (40-bit and 128-bit variations are supported)
IDEA	IDEA as listed in the ISO/IEC 9979 Register of Cryptographic Algorithms (128-bit supported)
RSA	RSA in accordance with Public Key Cryptographic Standards (PKCS) specification PKCS#1 Version 2.0, ANSI X9.31, IEEE 1363, ISO/IEC 14888-3 and U.S. FIPS PUB 186-2 (1024-bit, 2048-bit, 4096-bit and 6144-bit supported)
DSA	DSA in accordance with the Digital Signature Standard, U.S. FIPS PUB 186-2, ANSI X9.30 Part 1, IEEE P1363 and ISO/IEC 14888-3 (1024-bit supported)
ECDSA	ECDSA in accordance with ANSI X9.62, IEEE P1363, ISO/IEC 14888-3 and U.S. FIPS PUB 186-2 (192-bit default)
SHA-1, SHA-256, SHA-384, and SHA-512	SHA-1, SHA-256, SHA-384 and SHA-512 in accordance to U.S. FIPS PUB 180-2 and ANSI X9.30 Part 2
MD5 Message-Digest algorithm	MD5 Message-Digest algorithm in accordance with RFC 1321
MD2 Message-Digest algorithm	MD2 Message-Digest algorithm in accordance with RFC 1319
RIPEMD-160	RIPEMD-160 in accordance with ISO/IEC 10118-3:1998
RSA key transfer	RSA key transfer in accordance with RFC 1421 and RFC 1423 (PEM), PKCS#1 Version 2.0, IEEE P1363
Diffie-Hellman key agreement	Diffie-Hellman key agreement in accordance with PKCS#3
Simple Public-Key	Simple Public-Key GSS-API Mechanism (SPKM) authentication and key agreement in

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
GSS-API Mechanism (SPKM) authentication and key	accordance with RFC 2025, ISO/IEC 9798-3 and U.S. FIPS PUB 196
SSL v3 and TLS v1	SSL v3 and TLS v1 in accordance with RFC 2246
MAC	MAC in accordance with U.S. FIPS PUB 113 (for DES-MAC) and X9.19
HMAC	HMAC in accordance with RFC 2104
Pseudo random number generator	Pseudo random number generator in accordance with ANSI X9.17 (Appendix C) and FIPS 186-2
Version 3 public-key certificates and Version 2 CRLs	Version 3 public-key certificates and Version 2 CRLs in accordance with ITU-T X.509 Recommendation and ISO/IEC 9594-8 (4th edition, 2000 as well as earlier editions)
Version 3 public-key certificate and Version 2 CRL extensions	Version 3 public-key certificate and Version 2 CRL extensions in accordance with RFC 2459 and RFC 3280
Version 3 public-key certificate and Version 2 CRL extensions	Version 3 public-key certificate and Version 2 CRL extensions in accordance with U.S. FPKI X.509 Certificate and CRL Extensions Profile
Version 3 public-key certificate and Version 2 CRL extensions	Version 3 public-key certificate and Version 2 CRL extensions in accordance with NIST X.509 Certificate and CRL Extensions Profile for the Common Policy
Version 3 "Qualified" certificates	Version 3 "Qualified" certificates in accordance with RFC 3039 and ETSI TS 101 862
Version 3 public-key certificates and Version 2 CRLs	Version 3 public-key certificates and Version 2 CRLs in accordance with de-facto standards for Web browsers and servers
WTLS Certificate support in accordance with WAP WTLS Version 1.1.	WTLS Certificate support in accordance with WAP WTLS Version 1.1. (certificate issuance)
RSA algorithm identifiers and public key formats	RSA algorithm identifiers and public key formats in accordance with RFC 1422 and 1423 (PEM) and PKCS#1
Online Certificate Status Protocol, version 2. Working document of the IETF	Online Certificate Status Protocol, version 2. Working document of the IETF RFC 2560.
Standard file envelope format	Standard file envelope format based on Internet RFC 1421 (PEM)
PKCS#7 Version 1.5 based on RFC	PKCS#7 Version 1.5 based on RFC 2315 and Cryptographic Message Syntax (CMS) based on RFC 3369 and 3370

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
2315 and Cryptographic Message Syntax (CMS)	
S/MIME Version 2	S/MIME Version 2 based on RFC 2311
On-line GSS-API public key implementation mechanism using SPKM	On-line GSS-API public key implementation mechanism using SPKM in accordance with Internet RFC 2025 and SPKM entity authentication in accordance with FIPS 196
SSL v3 and TLS v1	SSL v3 and TLS v1 in accordance with RFC 2246
LDAP Version 2	LDAP Version 2 in accordance with RFC 1777 and RFC 2559
LDAP Version 3	LDAP Version 3 in accordance with RFC 2251-2256
Private key storage	Private key storage in accordance with PKCS#5 and PKCS#8
Secure Exchange Protocol (SEP)	Secure Exchange Protocol (SEP), built using Generic Upper Layers Security (GULS) standards ITU-T Recs. X.830, X.831, X.832 and ISO/IEC 11586-1, 11586-2, 11586-3 (SEP continues to be supported for backward compatibility only)
PKIX-CMP	PKIX-CMP in accordance with RFC 2510 and PKIX-CRMF in accordance with RFC 2511
PKCS 7/10	PKCS 7/10 (for Web based clients and VPN solutions)
Cisco Certificate Enrollment Protocol (CEP)	Cisco Certificate Enrollment Protocol (CEP) (for VPN solutions)
Hardware cryptographic interface	Hardware cryptographic interface in accordance with PKCS#11
Generic Security Services API (GSS-API)	Generic Security Services API (GSS-API) in accordance with RFC 1508 and 1509
IDUP-GSS-API	IDUP-GSS-API in accordance with Internet Draft draft-ietf-cat-idup-gss-08.txt
SNMPv3	The Simple Network Management Protocol, version 3 is the latest version of the IETF standard for managing network devices. Version 3 includes authentication and authorization, so is considered much more secure than previous versions. SNMP is widely implemented, but has some significant restrictions because of its very simple structure.
TFTP	The Trivial File Transfer Protocol (TFTP), as defined by IETF RFC 1350, is a very simple file transfer protocol that can be implemented in very small systems, such as firmware. It implements no authentication whatsoever and consequently is usable only in the most benign, protected environments.
DHCP	The Dynamic Host Control Protocol (DHCP) is defined by IETF RFC 2131 and modified by a host of other RFCs. It allows a machine, which at network initialization time does not know its own IP address, to request allocation of an IP address from a server and receive network configuration data sufficient to communicate on an IP network.
SM Spec	Signed Manifest Specification, The Open Group SM Spec Signed Manifest Specification, The Open Group, 1997. http://www.opengroup.org/pubs/catalog/c707.htm
CIM	The Distributed Management Task Force (DMTF) originally developed the Common Information Model (CIM) to provide a data model for integrating management across SNMP, the Desktop Management Interface (DMI) (another part of WBEM), Common

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
	<p>Management Information Protocol (CMIP or ISO 9596) (for telecom devices) and private applications. CIM is part of the DMTF's overall Web-based Enterprise Management (WBEM) initiative. WBEM includes CIM as the data definition, XML as the transport/encoding method, and HTTP as the access mechanism.</p> <p>CIM is an object-oriented data model for describing managed elements across the enterprise, including systems, networks, and applications. The CIM schema provides definitions for servers, desktops, peripherals, operating systems, applications, network components, users, and others along with details of each. One of the main functions CIM offers is the ability to define the associations between components. CIM's object-oriented approach makes it easier to track the relationships and interdependencies between managed objects. WBEM/CIM proponents promote this as a key advantage over SNMP.</p>
WBEM	The Web-Based Enterprise Management (WBEM) standard is an initiative by the DMTF to develop a broader enterprise management structure than SNMP. The DMTF is an industry coalition that is developing an enterprise management framework for computer systems that is richer than SNMP, the WBEM standards.
SMBIOS	The System Management Basic I/O System (SMBIOS) is a DMTF standard for making firmware-level information available via a CIM model on computer systems.
Intel PXE Specification	<p>The Intel-developed Preboot eXecution Environment (PXE) specification defines an OS-independent firmware-level mechanism for booting from a variety of media, including the network, using standard protocols.</p> <p>ftp://download.intel.com/labs/manage/wfm/download/pxespec.pd</p>
Intel PXE BIS Specification	<p>The Intel PXE Boot Integrity Services is an extension to the Intel PXE specification that provides for PKI-based authentication of the server to the booting client.</p> <p>ftp://download.intel.com/labs/manage/wfm/download/bisspec.zip</p>
EPC Tag Data Specification Version 1.1	Identifies the specific encoding schemes for a serialized version of the EAN.UCC Global Trade Item Number (GTIN®), the EAN.UCC Serial Shipping Container Code (SSCC®), the EAN.UCC Global Location Number (GLN®), the EAN.UCC Global Returnable Asset Identifier (GRAI®), the EAN.UCC Global Individual Asset Identifier (GIAI®), and a General Identifier (GID)
<u>900 MHz Class 0 Radio Frequency (RF) Identification Tag Specification.</u>	This document specifies the communications interface and protocol for 900 MHz Class 0 operation. It includes the RF and tag requirements and provides operational algorithms to enable communications in this band.
<u>13.56 MHz ISM Band Class 1 Radio Frequency (RF) Identification Tag Interface Specification.</u>	This specification defines the communications interface and protocol for 13.56 MHz Class 1 operation. It also includes the RF and tag requirements to enable communications in this band.
<u>860MHz – 930 MHz Class 1 Radio Frequency (RF) Identification Tag Radio Frequency & Logical Communication Interface Specification</u>	This document specifies the communications interface and protocol for 860 – 930 MHz Class 1 operation. It includes the RF and tag requirements to enable communications in this band.
Physical Markup	The PML Core specification establishes a common vocabulary set to be used within the

UNCLASSIFIED//FOR OFFICIAL USE ONLY

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
Language (PML)	EPCglobal Network. It provides a standardized format for data captured by readers. This specification also includes XML Schema and Instance files for your reference.
<u>ISO/IEC 15963:2004</u>	Information technology – Radio frequency identification for item management – Unique identification for RF tags
ISO/IEC 18000-4:2004	Information technology – Radio frequency identification for item management – Part 4: Parameters for air interface communications at 2.45 GHz
<u>ISO/IEC 18000-6:2004</u>	Information technology – Radio frequency identification for item management – Part 6: Parameters for air interface communications at 860 MHz to 960 MHz
<u>ISO/IEC 18000-7:2004</u>	Information technology – Radio frequency identification for item management – Part 7: Parameters for active air interface communications at 433 MHz
FIPS 140-2	Security Requirements for Cryptographic Modules
SNMPv3	The Simple Network Management Protocol, version 3 is the latest version of the IETF standard for managing network devices. Version 3 includes authentication and authorization, so it is considered much more secure than previous versions. SNMP is widely implemented, but has some significant restrictions because of its very simple structure.
<u>ISO/IEC 15408-1:1999</u>	Information technology – Security techniques – Evaluation criteria for IT security – Part 1: Introduction and general model
<u>ISO/IEC 15408-2:1999</u>	Information technology – Security techniques – Evaluation criteria for IT security – Part 2: Security functional requirements
<u>ISO/IEC 15408-3:1999</u>	Information technology – Security techniques – Evaluation criteria for IT security – Part 3: Security assurance requirements
CLF	Common Log Format. Typically, the information is presented in plain ASCII without special delimiters to separate the different fields. See http://www.ietf.org
ELF	Extended Log Format
IDMEF	Intrusion Detection Message Exchange Format ietf.org/html.charters/idwg-charter.html The IETF's Intrusion Detection Working Group (IDWG) is developing message formats and procedures for sharing messages between intrusion detection systems and the SEM systems that manage them. The IDMEF requirements were posted in an Internet Draft in October, 2002, along with a draft of the Intrusion Detection Exchange Protocol (IDXP). In January, 2003, an Internet Draft was submitted for IDMEF that included an XML implementation. This initiative is still in development and it's future is not determined
RFC 1155,	Structure of Management Information
RFC 1156	Management Information Base (MIB-I)
RFC 1157	SNMP
RFC 1187	Bulk table retrieval
RFC 1212	Concise MIB definitions
RFC 1213	Management Information Base (MIB-II)
RFC 1215	Traps
RFC 1227	SNMP Multiplex (SMUX)
RFC 1228	SNMP-DPI
RFC 1229	Generic-interface MIB extensions
RFC 1239	Reassignment of MIBs

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)	
Name	Description
RFC 1243	AppleTalk MIB
RFC 1248	OSPF MIB
1230 IEEE 802.4	Token Bus MIB
1231 IEEE 802.5	Token Ring MIB
ISO 8824-1	Abstract Syntax Notation One (ASN.1): Specification of basic notation
ISO 8824-2	Abstract Syntax Notation One (ASN.1): Information object specification
ISO 8824-3	Abstract Syntax Notation One (ASN.1): Constraint specification
ISO 8824-4	Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications
ISO 8825-1	ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
ISO 8825-4	ASN.1 encoding rules: XML Encoding Rules (XER)
The Table is (U//FOUO)	

15978 **(U//FOUO) APPENDIX C: TV-2 FOR IA**

15979 (U) Current technologies do not provide sufficient capabilities to satisfactorily enable required
 15980 GIG IA capabilities. Therefore, new technologies and standards will need to be developed. Table
 15981 C-1 provides an initial view of the required technologies and when they are expected to mature.
 15982 This table is a summary of the figures shown in Section 3 and the same cautions apply. Only the
 15983 gaps and recommendations are discussed for technologies needed to meet the 2008 GIG IA
 15984 objectives as discussed in the Transition Strategy (RCD Volume I). The discussion is further
 15985 limited to technologies that are deemed risky, either because no work is currently ongoing, or
 15986 because ongoing development effort will not be completed in time to deploy for 2008. In some
 15987 cases, gaps and recommendations are summarized for technologies needed for 2012 and beyond,
 15988 but only in cases where technology development efforts must begin now in order to meet those
 15989 milestone dates.

15990

Table C-1: (U//FOUO) TV-2 for IA

This Table is (U//FOUO)			
Technology/Standard	Short term 2006 (or earlier)	Mid term 2008	Long term 2010+
Assured Information Sharing			
Authentication Session Score Standard	Standard defined	Begin Compliance with Authentication Standard	
Authentication Confidence	Standard defined	Begin Compliance with Authentication Standard	
Authentication Tokens Hardware Tokens (CAC)	Standard defined	IA Enhanced CAC	
Biometrics	Medium and High Assurance PP's		
Pilots using Policy-driven AC Mechanisms - IDs, Privs and I&A SoM, with Manual Override Support	Pilots begin	Traditional Access Control Process	
Metadata Standard	Standard defined	Labeling Standard Ratified	
Initial IA Metadata Creation Tools	Pilots begin	Labeled data pilots	
Cryptobinding of Metadata			Standard defined
CDS Browse/Query	Standard defined	Improved CDS filtering	
CDS Collaboration Suite		Secure chat, e-mail w/ attachments	
CDS Databases		Bi-directional discovery and retrieval	
Single Sign-on	Standard defined	NCES IOC Single Sign-on	
Special Purpose Trusted Platforms	Standard defined	MILS pilots	

This Table is (U//FOUO)			
Technology/Standard	Short term 2006 (or earlier)	Mid term 2008	Long term 2010+
Multi-Purpose Trusted Platforms		Standard defined	High assurance platforms
Simple Trusted Applications	Standard defined		
Protection Profiles for Medium and High Assurance Access Control Mechanisms		Initial Authentication Infrastructure Standard for trusted software development Object sanitization research	
Highly Available Enterprise			
Policy-based Network Management	Network control functions automated within a single domain		
IA Policy-based Routing	Exchange of routing across tunnels (red/red routing exchange)	IA policy based routing implemented Initial support for mobile/tactical IA policy-based routing	
HAIPEv2 Products		Edge-to-edge enterprise boundary protection eliminating red gateways	
TRANSEC (Research TBD)		TRANSEC for wireless/radio links	
GIG ID Management Standard	Standard defined	Strong I&A of network admins	
Authentication Tokens Hardware Tokens		Standard defined	
Integrity/Confidentiality of Network Management & Control	Standard defined	Confidentiality and integrity of management & control	
Operational-based Resource Allocation		Limited support for end-to-end resource allocation	Operational-based resource allocation implemented
Cyber Situational Awareness and Network Defense			
Host-based IDS			Standard defined
Network-based IDS	Standard defined	Standardized sensors	
Anomaly Detection	Standard defined		
UDOP	Standard defined	Initial UDOP tools available	
NETCOP		Standard defined	
Network Mapping	Standard defined		
Vulnerability Scanning	Standard defined		

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)			
Technology/Standard	Short term 2006 (or earlier)	Mid term 2008	Long term 2010+
Host-based IPS		Initial automated analysis tools	
Network-based IPS	Standard defined	Research into automated response capability	
Traceback		Standard defined	
Correlation			Standard defined
Misuse Detection		Research misuse detection and intrusion detection in the Black Core	Standard defined
User Activity Profiling	Standard for collection, processing, & exchange of IA sensor data on IPv4 networks	Standard for collection, processing, & exchange of IA sensor data on IPv6 networks Devices capable of reporting their location available	
Assured Enterprise Management and Control			
User Identity Management	Standard accepted	All human users identified in accordance with GIG ID standard	Full implementation
Role-based Privileges	Privilege management standard ratified		
Privilege Management Infrastructure		Initial privilege management service	
OTNK	Benign fill	OTNK for wired and wireless products	ONTK for low bandwidth devices ONTK for coalition forces
Certificate Management		Identities in certificates comply with GIG ID standard	
Universal Configuration Management	Configuration management standards ratified		
Trusted Software Download	Secure software download Policy standards ratified	GIG-wide IA agents with trusted software download Initial policy infrastructure including deconfliction and synchronization	
Audit Format Standard	Audit format and exchange standard ratified		

UNCLASSIFIED//FOR OFFICIAL USE ONLY

This Table is (U//FOUO)			
Technology/Standard	Short term 2006 (or earlier)	Mid term 2008	Long term 2010+
Audit Aggregation & Analysis Standard		Compliance with audit standard, initial audit tools available	
This Table is (U//FOUO)			

15991