Index

A
Ability to verify (ATV) probability, 982–983
abort method, 203
Abstract Factory pattern, 589
Abstract objects, 402
Abstraction layers, 801
Access control, 403, 496
   Assertion Builder pattern, 763
   broken, 14–15
   Business tier patterns, 693, 695
   DMTF, 403–404
   EPAL, 405–408
   IETF Policy Management Working Group, 404
   J2EE, 252
   management services, 28, 38–39
   Parlay Group, 405
   physical and logical, 962–963
   for smart cards, 995
   Web services, 408–410, 744
Access control lists (ACLs), 52
   J2EE, 227
   JMS, 271–272
Access points in case study, 932
AccessController class, 106
Accountability, checklist for, 500
Accounts. See User account provisioning
Accuracy of biometric verification, 981–982
ACLs (access control lists), 52
   J2EE, 227
   JMS, 271–272
Actions in Parlay, 405
Active RFID tags, 35
Activities in Secure UP, 445–446
Actors in use cases, 911–912
Add-on, security as, 5
Add operation in SPML, 834
   -addcert option, 135
   -addjarsig option, 135–136
   addListener method, 648
   AddResponse message, 834
Administration
   in biometric systems, 996
   reality checks for, 508–510
   in Web tier patterns, 616
Administrator privileges, 619
Advanced Encryption Standard (AES), 58,
   170–171, 317
Advice in SAML assertions, 371
Advisory policies, 497
Agent-based and agentless architecture for user
   account provisioning, 832
Agent-based authentication, 233, 247–248
Agent-based policy enforcement, 693
Aggregation, service, 685
Alchemy of security design, 439–440
   conclusion, 530–531
   framework adoption, 525–526
   rationale, 440–441
   reality checks. See Reality checks
   refactoring, 529
Index

Alchemy of security design, continued references, 531–532
Secure UP, 444–452
   artifacts in, 449–452
   risk analysis, 452–454
   trade-off analysis, 455
   security patterns. See Security patterns
service continuity and recovery, 530
testing, 523–525
ALE (Annual Loss Expectancy), 453, 922
Alerts
   SSL, 70
   Web services patterns, 750
AlgorithmParameter class, 155
AlgorithmParameterGenerator class, 155
Alteration attacks
   SAML, 385
   Secure Logger pattern, 583, 586
Annual Loss Expectancy (ALE), 453, 922
Anonymous EJB resources, 262–263
AOP (Aspect Oriented Programming) techniques, 634
Apache Struts
   in form validation XML, 565
   with SecureBaseAction, 565–566
   with SimpleFormAction, 566–567
   in Web data validation, 564
APDUs (Application Protocol Data Units), 138–139
APIs
   BioAPI, 967–968
   CertPath, 179–181
   JAAS, 200
   Java, 525
   Java Card, 139
   JCA, 155–159
   JCE, 163
   JSSE, 186
   SAAJ, 277, 724, 913
   SASL, 216–217
   Vendor-specific, 751
Applets, 109–115
   Java Card, 140–142
   signed, 112–115
   for smart cards, 995
Appletviewers, 109–110
Appliances
   firewall, 746
   strategies for, 593
   XML-aware, 352
Application-based authentication, 234
Application Controller, 550
Application data messages in SSL, 69
<application-desc> element, 118
Application Protocol Data Units (APDUs), 138–139
Application Requests, 372
Application security assessment model, 499–500
Application Security Providers, 527
Applications and application security, 8–10
   access control, 14–15
   audit and logging, 15
   authentication, 14
   buffer overflow, 11
   in case study, 913–914
   CLDC, 129
   coding problems, 17
   configuration data, 14
   cross-site scripting, 11
   data injection flaws, 11
   data transit and storage, 12
   deployment problems, 16–17
   DOS and DDOS attacks, 15–16
   encryption, 13
   error handling, 12
   input validation failures, 10
   Intercepting Web Agent pattern, 606–607
   J2EE, 227–228
   JSSE, 594
   man-in-the-middle attacks, 16
   multiple sign-ons, 16
   output sanitation, 10
   password exploits, 13
   policies, 15
   Secure Pipe pattern, 590
   security provisioning patterns, 892–893
   security tokens, 12–13
   servers
      for biometrics, 987
      for smart cards, 971
      in use cases, 906
      session identifiers, 12
      session theft, 13–14
      as weakest links, 8
   Web tier patterns, 617–618
Applying security patterns, 478–488
Architecture
   in case study, 913–914, 915–918
   inefficiencies, 5–6
   J2EE, 227–228
   J2ME, 126–127
   Java, 94–97
   Liberty Alliance, 391–392
   patterns-driven security design, 488
personal identification systems, 995
biometrics, 979–983
smart cards, 970–975
risk analysis, 452
SAML, 368–369, 371–372
Secure UP, 444, 449
in security patterns
Authentication Enforcer, 537
Business tier, 693–694
Intercepting Validator, 562
Intercepting Web Agent, 609
Secure Base Action, 571–572
Secure Service Proxy, 605
user account provisioning
centralized model vs. decentralized, 823–825
components of, 828
logical, 826–828
Web services, 284–285
XACML, 420
Artifact Resolution Profile, 367
Artifacts in Secure UP, 449–452
Aspect Oriented Programming (AOP) techniques, 634
Assemblers, J2EE, 228
Assertion Builder pattern, 477, 486
consequences, 763–764
forces, 757
participants and responsibilities, 760–762
problem, 756–757
reality check, 775
related patterns, 775
sample code, 764–774
security factors and risks, 774–775
in service provisioning, 475
in single sign-on, 763
and Single Sign-on Delegator pattern, 782–784, 801
solution, 757–759
strategies, 762–763
structure, 757–759
Assertion class, 760
Assertion Query/Request profile, 368
AssertionContext class, 758
AssertionContextImpl class, 760, 770–774
Assertions
Java System Access Manager, 402
SAML, 369
attribute, 378–380
authentication, 376–378, 387
authorization, 380–382
WS-Policy, 411
WS-Security, 342–343
assertRequest method, 671
Assessment checklists, 499–500
Asset valuation, 453
Asymmetric ciphers, 58–62
Attachments in SOAP messages, 287
Attack trees, 82–83
AttributeQuery class, 431
Attributes
J2EE, 229
SAML
assertion, 369, 378–380
authority, 369, 371
mapping, 368
profile, 368
repository, 369
Secure Service Facade pattern, 681
XACML, 415, 430
AttributeStatement class, 430–432, 760
<AttributeValue> element, 417
ATV (ability to verify) probability, 982–983
Audit Interceptor pattern, 473, 624–625, 696
in case study, 924, 927, 929, 936
consequences, 629–633
forces, 625
and Message Inspector pattern, 732
participants and responsibilities, 627
problem, 624
reality check, 634
related patterns, 634–635
sample code, 630–631
security factors and risks, 633–634
solution, 625
strategies, 627–629
structure, 625–626
audit method, 633
AuditClient.java file, 632–633
Auditing
Assertion Builder pattern, 763
Audit Interceptor pattern. See Audit Interceptor pattern
biometrics, 996
Business tier patterns, 624, 633, 695–696
Dynamic Service Management pattern, 658
failures in, 15
identity management, 28–29, 361, 813
Secure Service Facade pattern, 685
Secure UP, 447, 452
Security Wheel, 442
Single Sign-on Delegator pattern, 801
Web services, 294–295, 749
Web tier patterns, 619, 620
AuditLog class, 627, 630–631
Index

AuditLogJdbcDAO class, 630
AuditRequestMessageBean.java file, 630–632
Authentication, 51–52
   assessment checklists, 499
   biometrics, 981, 986, 989–991, 996
   broken, 14, 763, 850
   in case study, 932
   J2EE, 229, 232–234
      agent-based, 234, 247–248
      application-based, 234
      container-based, 233–234
      declarative, 235–236
      programmatic, 236
   Web tier, 240–247
JAAS, 200
   classes for, 199
   in clients, 206–210
   LoginModule for, 200–206, 541–544
   web-tier, 245–246
Java code, 144
JMS, 271
JSSE, 190–191
Liberty Alliance sessions, 398
multi-factor, 993–994
personal identification, 29–30
SAML, 369, 371
   assertions in, 369, 376–378, 387
   third-party, 383
in security patterns
   Assertion Builder, 761, 773
   Authentication Enforcer. See Authentication Enforcer pattern
   Business tier, 693–694
   Dynamic Service Management, 656
   Interception Web Agent, 612
   Password Synchronizer, 850
   Policy Delegate, 674
   Secure Base Action, 574
   Secure Service Facade, 683
   Secure Session Object, 690
   Web tier, 616
Security services, 527
   Security Wheel, 442
   smart cards, 969, 972–975, 991
   in trust model, 493
   Web services, 294
Authentication Enforcer pattern, 467
   in case study, 916, 922, 925, 927, 934
   consequences, 544–545
   forces, 536
   participants and responsibilities, 537
   problem, 535–536
   reality checks in, 548
   related patterns, 548
      Container Managed Security, 642
      Secure Base Action, 575
      sample code, 545–547
      security factors and risk in, 546–548
      solution, 536
      strategies in, 538–544
      structure, 536–537
Authentication provider-based strategy
   Authentication Enforcer pattern, 539–540
   JAAS Login Module, 544
Authentication Request protocol, 366
<AuthenticationContextStatement> message, 399
AuthenticationEnforcer class, 573
AuthenticationInstant class, 762–763
AuthenticationProvider class, 544
AuthenticationStatement class, 768–770
<AuthnRequest> message, 386, 398
<AuthnResponse> message, 386, 398
Authoritative Source of Data pattern, 463
Authorization, 52–53
   classes for, 198–199
   J2EE, 229, 234–236, 553
      declarative, 259–261
      programmatic, 261–262
   Web tier, 251–255
JAAS
   implementing, 210–215
   strategy, 556–559
SAML, 369, 380–382, 430
in security patterns
   Dynamic Service Management, 658
   Interception Web Agent, 614
   Policy Delegate, 676
   Secure Base Action, 576
   Secure Service Object, 692
Security services, 527
   Security Wheel, 441–442
   trust model, 495
   Web services, 294
   XACML 2.0, 430
Authorization and Access Control service, 500
Authorization Enforcer pattern, 468
   consequences, 559
   forces, 548
   participants and responsibilities, 550
   problem, 548–549
   reality check, 559
   related patterns, 559
   security factors and risks, 559
   solution, 549
strategies, 550–558
structure, 549
Authorization providers, 550–552
<AuthorizationDecisionStatement> feature, 382, 760
AuthorizationEnforcer class, 571
AuthPermission class, 200
Automated back-out strategy, 849–851
Automated password retry, 851
Availability
in case study, 915
identity management patterns, 813
J2EE network topology, 277
Message Interceptor Gateway pattern, 714
Secure Message Router pattern, 743
security provisioning patterns, 891
Security Wheel, 441
in use cases, 907
Web services, 295–296

B
B2B (Business-to-Business) applications
identity management in, 358
Liberty Alliance transaction support in, 391
Back-out password strategy, 849–851
Basic authentication
J2EE, 240, 243
in web.xml, 545
Basic Information, 371
Basic Profile, 366
Basics of security, 49–50
cryptography. See Encryption and cryptography
identity management, 83–89
LDAP. See LDAP (Lightweight Directory Access Protocol)
references, 89–91
requirements and goals, 50–53
summary, 89
threat modeling, 81–83
beginHandshake method, 194
Bertillon, Alphonse, 34
Best practices, 20
Business tier patterns
architecture, 694–695
infrastructure, 693–694
policy, 694–695
Identity management patterns, 812–813
personal identification
biometrics, 996–997
smart cards, 995
security provisioning patterns
application design, 891
quality of service, 891–892
risk mitigation, 894
server sizing, 893
Web services patterns
communication and message security, 747–749
infrastructure, 745–746
testing and deployment, 750–751
Web tier patterns, 615–616
applications, 618–620
communication, 617–618
infrastructure, 615–617
Binary security tokens, 343
BinaryToken class, 805
BIND requests, 219
BioAPI standard, 33, 967–968
Biometric identification and authentication, 32–34, 966, 979–980
accuracy, 981–982
architecture and implementation, 983–987
best practices, 996–997
in multi-factor authentication, 993–995
operational models, 987–991
SSO strategy, 991–993
verification process, 980–981
Biometric service providers (BSPs), 967
Black box testing, 524
in case study, 952–953
Secure UP, 446–447, 448, 451
Blanket MIDlets, 133
Block ciphers, 165–166
Block encryption algorithms, 317
Bodies in SOAP messages, 287
Broken access control risk, 763
Broken authentication, 14
Assertion Builder pattern, 763
Password Synchronizer pattern, 850
Browser plug-ins
for biometrics, 986
for smart cards, 974
Brute force attacks, 55
BSPs (biometric service providers), 968
Buffer overflow, 11
Build portion in patterns-driven security design, 488
Build vs. buy decisions
Assertion Builder pattern, 775
Business tier pattern, 696
Intercepting Web Agent pattern, 615
Password Synchronizer pattern, 890
Built-in Java security model, 97
Index

Business and business challenges, 5–7
in case study, 920
processing logic, 624
service provisioning, 818–822
tasks auditing, 620
Business Delegate pattern, 482
and Delegate pattern, 677
and Single Sign-on Delegator pattern, 777
and Synchronizer pattern, 891
Business tier
in case study, 929–930, 933, 936–937
in J2EE. See EJB tier in J2EE
reality checks for, 521–522
security patterns, 460–462, 623–624
Audit Interceptor, 624–635
best practices, 693–696
Container Managed Security, 635–645
Dynamic Service Management, 645–659
factor analysis, 491
Obfuscated Transfer Object, 659–668
overview, 473–475, 480–485
pitfalls, 696
Policy Delegate, 668–677
references, 697
Secure Service Facade, 677–686
Secure Session Object, 686–693
Business-to-Business (B2B) applications
identity management in, 358
Liberty Alliance transaction support in, 391
Bytecode verifiers, 108
C
CA (connector architecture), 266–270
CA SiteMinder WebAgent, 612–614
Caching in Single Sign-on Delegator pattern, 800–801
CADs (card acceptance devices), 30–31
Caesar ciphers, 57
California, notice of security breach requirements, 26
CallbackHandler class, 199–200, 207–209, 214, 541, 544
Callback in J2EE, 256–257
Canadian Public Accounting Board, 26
Canonical transformations, 364
Canonicalization algorithms, 304, 319–320
<CanonicalizationMethod> element, 301
CAP (Converted Applet) files, 141–142
Capacitance-based scanners, 985
Capstone project, 58
Card acceptance devices (CADs), 30–31, 972
Card Unique Identifiers (CUIDs), 971–973
Cards, smart cards. See Smart cards
CAs (Certificates of Authority), 64–66
for certificates
issuing, 66–68
revoking, 68
CertPath, 179
J2EE, 244
for signed applets, 112
Case study, 901
architecture, 913–915, 917–919, 927–930
assumptions, 904–905
challenges, 902–904
conceptual security model, 915–917
conclusion, 956–957
deployment, 953–955
design, 930
Business tier, 933, 936–937
classes in, 939–940
data modeling and objects, 939–940
factor analysis, 931–932
Identity tier, 937
infrastructure, 934–935
policy, 930–931
services in, 940–951
threat profiling, 937
tier analysis, 935
trust model, 937
Web Services tier, 932–933, 937
Web tier, 932–933, 936
development, 951–952
lessons learned, 955–956
overview, 902
pitfalls, 956
references, 957
risk analysis and mitigation, 920–922
security patterns, 924–927, 939
summary, 955
trade-off analysis, 923
use cases. See Use cases
Caslon Analytics Privacy Guide, 26
Catalog service, 911, 943–946
Catalogs, security pattern, 467
CBEFF (Common Biometric Exchange File Format), 34
CDC (Connected Device Configuration), 127–128
CDSSO (cross-domain SSO)
in identity management, 85–86
in Liberty Alliance, 400
Centralization
auditing, 629
authentication, 545
Authorization Enforcer pattern, 559
Index

Adaptation, 230
authentication, 184–186
Authorization Enforcer, 549–550
Assertion Builder, 758–759
Audit Interceptor, 625–626
Authentication Enforcer, 536–537
Authorization patterns, 326–328
Authorization patterns, 326–328
Basic (flat), 326
Centralized model in user account provisioning, 823–826
Certificate revocation lists (CRLs), 68
Certificate Signing Requests (CSRs), 66, 124
CertificateFactory class, 154, 180
Certificates and certificate keys, 64–66, 78–81
for applets, 114–115
CA role, 66–68
certificate chains, 180–182
importing, 115, 122–123
for JAD files, 135
in JSSE mutual authentication, 190–191
for keytool, 119
PKI, 326
printing, 123
revocation, 68, 78
Secure Pipe pattern, 598
security pattern factor analysis, 493
for SSL, 67
tokens, 805, 812
Web tier patterns, 618
Certificates of Authority (CAs), 64–66
for certificates
issuing, 66–68
revoking, 68
CertPath, 179
J2EE, 244
for signed applets, 112
CertPath
for certificate chains, 180–182
classes and interfaces in, 179–180
CertPathBuilder class, 180
CertPathValidator class, 180
-certrq option, 124
CertStore class, 180
CGI in Web tier patterns, 617
Challenge-response protocol authentication, 976–980
Change management request (CMR) system, 451
ChangeCipherSpec messages, 69, 71
Check Point patterns, 458
checkPermission method, 104
checkRead method, 105
Child nodes in attack trees, 82
Children’s Online Privacy Protection Act (COPPA), 25
CIM (Common Information Model), 404–405
Cipher class
in JCE, 162, 164
in Secure Logger pattern, 582
<CipherData> element, 315–316
CipherInputStream class, 167
CipherOutputStream class, 167
Ciphers, 51
asymmetric, 58–62
JCE
block, 165–167
stream, 167–168
symmetric, 56–58
CipherState messages, 70
Circles of trust
in Liberty specifications, 388–389
in Single Sign-on Delegator pattern, 801
Claims in WS-Security, 338
Class diagrams for security patterns
Assertion Builder, 758–759
Audit Interceptor, 625–626
Authentication Enforcer, 536–537
Authorization Enforcer, 549–550
Container Managed Security, 636
Credential Tokenizer, 803–805
Dynamic Service Management, 646
Intercepting Validator, 562–563
Intercepting Web Agent, 608
JAAS Login Module strategy, 541–542
MBean strategy, 649–650
Message Inspector, 719
Message Interceptor Gateway, 709
Obfuscated Transfer Object, 660
Password Synchronizer, 843–844
Policy Delegate, 670
Secure Base Action, 571
Secure Logger, 578–579
Secure Message Router, 735–736
Secure Pipe, 591–592
Secure Service Facade, 678–679
Secure Service Proxy, 600
Secure Session Object, 687–688
Single Sign-on Delegator, 778–779
Classes
in case study, 939–940
CertPath, 179–180
JAAS, 198–200
Java, 95–96
JCA, 153–155
JCE, 162
JSSE, 184–186
}

encryption, 666
logging, 577, 584
Message Interceptor Gateway pattern, 713
policies, 421–422, 682
routings, 742–743
transaction management, 683
validations, 568–569
Web services patterns, 750
Centralized model in user account provisioning, 823–826
CER (Crossover Error Rate) probability, 980
Certificate revocation lists (CRLs), 68
Certificate Signing Requests (CSRs), 66, 124
CertificateFactory class, 154, 180
Certificates and certificate keys, 64–66, 78–81
for applets, 114–115
CA role, 66–68
certificate chains, 180–182
importing, 115, 122–123
for JAD files, 135
in JSSE mutual authentication, 190–191
for keytool, 119
PKI, 326
printing, 123
revocation, 68, 78
Secure Pipe pattern, 598
security pattern factor analysis, 493
for SSL, 67
tokens, 805, 812
Web tier patterns, 618
Certificates of Authority (CAs), 64–66
for certificates
issuing, 66–68
revoking, 68
CertPath, 179
J2EE, 244
for signed applets, 112
CertPath
for certificate chains, 180–182
classes and interfaces in, 179–180
CertPathBuilder class, 180
CertPathValidator class, 180
-certrq option, 124
CertStore class, 180
CGI in Web tier patterns, 617
Challenge-response protocol authentication, 976–980
Change management request (CMR) system, 451
ChangeCipherSpec messages, 69, 71
Check Point patterns, 458
checkPermission method, 104
checkRead method, 105
Child nodes in attack trees, 82
Children’s Online Privacy Protection Act (COPPA), 25
CIM (Common Information Model), 404–405
Cipher class
in JCE, 162, 164
in Secure Logger pattern, 582
<CipherData> element, 315–316
CipherInputStream class, 167
CipherOutputStream class, 167
Ciphers, 51
asymmetric, 58–62
JCE
block, 165–167
stream, 167–168
symmetric, 56–58
CipherState messages, 70
Circles of trust
in Liberty specifications, 388–389
in Single Sign-on Delegator pattern, 801
Claims in WS-Security, 338
Class diagrams for security patterns
Assertion Builder, 758–759
Audit Interceptor, 625–626
Authentication Enforcer, 536–537
Authorization Enforcer, 549–550
Container Managed Security, 636
Credential Tokenizer, 803–805
Dynamic Service Management, 646
Intercepting Validator, 562–563
Intercepting Web Agent, 608
JAAS Login Module strategy, 541–542
MBean strategy, 649–650
Message Inspector, 719
Message Interceptor Gateway, 709
Obfuscated Transfer Object, 660
Password Synchronizer, 843–844
Policy Delegate, 670
Secure Base Action, 571
Secure Logger, 578–579
Secure Message Router, 735–736
Secure Pipe, 591–592
Secure Service Facade, 678–679
Secure Service Proxy, 600
Secure Session Object, 687–688
Single Sign-on Delegator, 778–779
Classes
in case study, 939–940
CertPath, 179–180
JAAS, 198–200
Java, 95–96
JCA, 153–155
JCE, 162
JSSE, 184–186
Index

Classification of security patterns, 497–498
ClassLoader, 108–109
CLDC (Connected Limited Device Configuration), 128–130
Client-certificate authentication
Authentication Enforcer pattern, 548
J2EE, 244
in web.xml, 546
Client Device tier, reality checks for, 511
ClientHello messages, 70
ClientKeyExchange messages, 70
Clients
in case study, 933
Identity Provider Agent strategy, 729
J2EE, 226–227, 233, 255–258
JAAS authentication for, 206–210
JSSE, 164–188
Liberty specifications, 389
SAML, 875–840
SASL, 217–218, 220
in security patterns
Assertion Builder, 760
Audit Interceptor, 627
Authentication Enforcer, 537
Container Managed Security, 637
Credential Tokenizer, 805
Intercepting Validator, 562
Intercepting Web Agent, 608
Message Inspector, 719
Message Interceptor Gateway, 709
Obfuscated Transfer Object, 661
Password Synchronizer, 888
Policy Delegate, 670, 677
Secure Base Action, 573
Secure Logger, 580
Secure Message Router, 736
Secure Pipe, 592
Secure Service Proxy, 601
Secure Session Object, 688
Single Sign-on Delegator, 780, 784–785
server connections, 908, 916
use cases, 910–911
closeService method, 788, 791
closeSSOConnection method, 791
Clustered PEP pattern, 464
CMR (change management request) system, 451
code, Java, 143
obfuscation, 144–145
reverse engineering, 143–144
Codebase in Java 2, 106–107
CodeSource in Java 2, 105
Coding problems, 17
Cold Standby pattern, 464
Collisions in one-way hash functions, 55
<CombinerParameters> element, 417
Command APDUs, 138
commit method
LoginModule, 203
SAML, 385
commitTransactions method, 405
Common Biometric Exchange File Format (CBEFF), 34
Common classes in JAAS, 199
Common Information Model (CIM), 404–405
Common Open Policy Service (COPS), 404
Common SAML functions, 402
Communication
biometrics, 997
JGSS, 215
Liberty Alliance, 400
Web services, 289–290, 747–750
Web tier patterns, 617–619
Compact Virtual Machine (CVM), 128
Comparator-checked Fault Tolerant System pattern, 465
Compatibility
in proprietary systems, 6
in Secure Pipe pattern, 598
Compiling applets, 113
Component-managed sign-on, 268–270
Component security
Business tier patterns, 694
J2EE, 238
authentication, 240–248
authorization, 251–255
context propagation, 265–266
HTTP session tracking, 248–251
users, groups, roles, and realms, 238–239
Web tier, 239–240
Composability issues, 749
Computer Security Institute survey, 3–4
Conceptual security model, 915–917
Concurrency
  Message Inspector pattern, 732
  Secure Session Object pattern, 693
<condition> element
  EPAL, 406
  XACML, 413
Conditions
  Parlay, 405
  policy design, 496–497
  SAML assertions, 371
Confidentiality, 50–51, 498
  breaches, 357
in security patterns
  Assertion Builder, 774–775
  Authentication Enforcer, 548
  Dynamic Service Management, 658
  Message Inspector, 731
  Obfuscated Transfer Object, 666–667
  Policy Delegate, 677
  Secure Logger, 589
  Secure Pipe, 595
  Security Wheel, 442
  Web services, 292, 295
Configuration, 6
  Assertion Builder pattern, 774
  in case study, 954
  insecure, 14
  J2ME, 127–130
  Web services patterns, 751
Configuration class, 199
Conformance requirements, 389
Connected Device Configuration (CDC), 127–128
Connected Limited Device Configuration (CLDC), 129–130
Connections
  in case study, 916
  client-server, 908, 916
  SSL, 69–70, 186–190
  in use cases, 908
Connector architecture (CA), 266–270
Connector Factory, 828
Consequences in security patterns, 466–467
  Assertion Builder, 763–764
  Audit Interceptor, 629–630
  Authentication Enforcer, 544–545
  Authorization Enforcer, 559
  Container Managed Security, 640
  Credential Tokenizer, 806–807
  Dynamic Service Management, 650–652
  Intercepting Validator, 568–569
  Intercepting Web Agent, 610–612
  Message Inspector, 729–731
  Message Interceptor Gateway, 713–714
  Obfuscated Transfer Object, 666
  Password Synchronizer, 850–851
  Policy Delegate, 672–673
  Secure Base Action, 574–575
  Secure Logger, 584, 587
  Secure Message Router, 742–743
  Secure Pipe, 595
  Secure Service Facade, 682–683
  Secure Service Proxy, 605–606
  Secure Session Object, 691–692
  Single Sign-on Delegator, 786–787
Consequences in Java System Access Manager, 402
Constraints
  authorization, 252
  in use cases, 909
Contact cards, 30–31
Container authenticated strategy, 539
Container-based security, 230–232
  authentication, 230–232
  authorization, 233–234
  declarative, 231–232
  JACC, 236
  programmatic, 232
  protection domains in, 234–235
  sign-ons in, 268–269
  TLS, 237
Container Managed Security pattern, 473
  consequences, 640
  forces, 635
  participants and responsibilities, 636–638
  problem, 635
  reality check, 644
  related patterns, 644–645
  sample code, 640–643
  security factors and risks, 643–644
  solution, 636
  strategies, 638–639
  structure, 636
Content encryption in Web services patterns, 751
Content-specific policies, 496
Context
  in J2EE, 252
  propagation of, 265–266
  in XACML, 414–416, 419–420
Context Object related patterns
  Authentication Enforcer, 548
Index

Context Object related patterns, continued
  Authorization Enforcer, 560
  Secure Base Action, 577
Continuity, 530
  in case study, 917
  strategies, 23
  in use cases, 909
Control transformations, 145
Converted Applet (CAP) files, 141–142
Cookies
  HTTP session tracking, 248–251
  Liberty Alliance, 394
COPPA (Children’s Online Privacy Protection Act), 27
COPS (Common Open Policy Service), 404
CORBA-based clients, 266
Core Web services standards, 286
  SOAP, 286–288
  UDDI, 289
  WSDL, 288
  XML, 286
Corporations, identity management in, 358
Correlation
  in fingerprint matching, 981
  in Web services patterns, 748
Countermeasures, 384
CRC (cyclic-redundancy check) algorithms, 51, 54
Create, read, update, and delete (CRUD) form data, 560
create method
  AddUser, 838–839
  AuthenticationStatement, 768–770
createAssertionReply method, 773
createAssertionStatement method, 764
createAuthenticationStatement method, 764, 768
createCondition method, 405
createMBean method, 656–657
createObjectName method, 657
createPasswordRequest method, 867
createRule method, 405
createServerSocket method, 597
createService, 788
createSocket method, 597
createSPMLRequest method, 860, 863–867
createSSLEngine method, 194
createSSOConnection method, 789–790
createSSOConnection method, 780
createSSOToken method
  AssertionContextImpl, 771
  SSODelegatorFactoryImpl, 793–74
createToken method, 809
Credential Collector, 369–370
Credential Tokenizer pattern, 477, 486
  consequences, 806–807
  forces, 802
  participants and responsibilities, 805
  problem, 802
  reality check, 812
  related patterns, 812
  sample code, 807–811
  security factors and risks, 811–812
  and Single Sign-on Delegator pattern, 801–802
  solution, 802–803
  strategies, 806
  structure, 803–805
Credentials
  delegation of, 215
  J2EE, 229
  Liberty Alliance, 399
CRLs (certificate revocation lists), 68, 78
Cross-domain federations, 88–89
Cross-domain SSO (CDSSO),
  in identity management, 85–86
  in Liberty Alliance, 400
Cross-site scripting (XSS), 11
Crossover Error Rate (CER) probability, 980
CRUD (create, read, update, and delete) form data, 560
Cryptographic Service Providers, 153, 160–161
Cryptographic token interface standard, 964
Cryptography. See Encryption and cryptography
CSRs (Certificate Signing Requests), 66, 124
CUIDs (Card Unique Identifiers), 969–971
Custom error handling, 836–837
Custom tag libraries for permissions, 554–556
Customer IT security requirements, 412
CVM (Compact Virtual Machine), 128
Cyclic-redundancy check (CRC) algorithms, 51, 54

D
DAP (Directory Access Protocol), 74
Data Encryption Standard (DES), 58, 163
Data flow in XACML, 418–419
Data injection flaws, 11
Data Protection Directive, 25–26
Data Transfer HashMap pattern, 668
Data transformations, 144–145
Database communication, 694
DCE PAC Profile, 368
DDOS (distributed DOS) attacks, 15–16, 920
Debuggers in white box testing, 524
Decentralized model in user account provisioning, 823–826
<Decision> element, 428
Declarative auditing, 629
Declarative authorization, 235–236, 259–261
Declarative security
  Container Managed Security pattern, 635, 640
  EJBs, 638
  J2EE, 231–232, 252
Decompiling Java code, 143–144
Decoupling
  in Audit Interceptor pattern, 630
  in Intercepting Web Agent pattern, 607
  validations from presentation logic, 569
Decryption. See Encryption and cryptography
Default settings in Identity management patterns, 814
Defensive strategies, 20
Delegates and delegations
  administration, 361
  EJB tier, 263–264
  Policy Delegate pattern. See Policy Delegate pattern
  Web services, 296
Delete operation in SPML, 834
Deleted data detection, 582
DeleteRequest message, 834
Deleting keystores, 124
Demilitarized Zones (DMZs), 274, 617
Denial of Service (DOS) attacks, 15–16
  in case study, 920
  Intercepting Web Agent pattern, 611
  SAML, 384–385
  Web services, 291
  Web tier patterns, 616
  XKMS, 337
Dependencies in WS-Policy, 411
Deployed component restrictions, 695
Deployment and deployment descriptors
  in case study, 953–955
  J2EE, 245, 248, 251, 254, 260, 264
  patterns-driven security design, 489
  problems in, 16–17
  Web services patterns, 750–751
DES (Data Encryption Standard), 58, 163
DescriptorStore class, 648
Design
  alchemy of. See Alchemy of security design
  in case study. See Case study policy, 496–497
Design patterns, 19
Destinations in JMS, 271
DestinationSite class, 760, 762
destroy method, 674
Detached signatures, 299, 309–310
Detecting data deletion, 582
Developers in J2EE, 228
Development in case study, 951–952
Devices
  in case study, 933
  in security pattern factor analysis, 493
Differentiators, 832–833
Diffie-Hellman (DH) key agreement, 61–62, 173–175
Digest authentication, 244–245
digest method, 156, 588
Digester class, 588
<DigestMethod> element, 302
Digests
  JCA, 155–156
  XML signatures, 311
<DigestValue> element, 302
Digital certificates. See Certificates and certificate keys
Digital Signature Algorithm (DSA)
  Cryptographic Service Providers, 153
  XML signatures, 304
Digital signatures. See Signatures
  Direct access in Web tier patterns, 619
  Directive, Data Protection, 25–26
  Directories in LDAP, 74
  Directory Access Protocol (DAP), 74
  Directory services, 28, 704–705
  Directory Services Markup Language (DSML), 833–834
Disassembling Java code, 143–144
Discovery
  service, 672
  in user account provisioning, 829
Distributed DOS (DDOS) attacks, 15–16, 920
Distributed Management Task Force (DMTF), 404–405
Distributed policy stores, 421
Distributed security, 633–634
DLLs (dynamically linked libraries), 967
DMTF (Distributed Management Task Force), 404–405
DMZs (Demilitarized Zones), 274, 617
doAs method, 212
doAsPrivileged method, 212
Document style web services, 290
doFinal method, 164
Domain models, 370–371
domains, protection
  J2EE, 234–235
  Java 2, 98–99
doPost method
  for new sessions, 249–250
  SingleProxyEndpoint, 603–604
Index

DOS attacks. See Denial of Service (DOS) attacks
Drivers, JDBC, 272
<ds:KeyInfo> element, 314
<ds:Signature> element, 345–346
DSA (Digital Signature Algorithm)
   Cryptographic Service Providers, 153
   XML signatures, 304
DSIG profiles, 417
DSML (Directory Services Markup Language),
   833–834
dumpHeaderContents method, 872–873
Duplication in Secure Service Facade pattern, 686
Dynamic Service Framework, 681
Dynamic Service Management pattern, 473, 482, 484
   consequences, 650–652
   forces, 645–646
   participants and responsibilities, 646, 648
   problem, 645
   reality check, 658–659
   related patterns, 659
   sample code, 652–658
   security factors and risks, 658
   solution, 646
   strategies, 649–650
   structure, 646
Dynamically linked libraries (DLLs), 967

E
EBJContext interface, 262
EJBX registry, 422
ECP (Enhanced Client and Proxy) profile, 386
EEPROM in smart cards, 137–138
EER (Equal Error Rate) probability, 982
Effect Matrix, 455
EIS (Enterprise-information system) tier, 227, 266
   connector architecture in, 266–270
   JDBC in, 272
   JMS in, 271–272
Ejb-jar.xml deployment descriptor, 642–643
EJB tier in J2EE, 259
   anonymous and unprotected resources,
   262–263
   context propagation from web-tier to, 265–266
   declarative authorization, 259–261
   principal delegations, 263–264
   programmatic authorization, 261–262
   run-as identity, 264–265
ejbCreate method
   AuditRequestMessageBean, 631
   SecureSessionFacadeSessionBean, 684
ejbRemove method, 631
EJBs. See Enterprise Java Beans (EJBs)
Electronic Product Codes (EPCs), 35
Element-level encryption, 748
Elgamal cipher, 61
Embedded SQL commands, 560
Enabling technologies for personal identification, 964
   BioAPI, 967–968
   GINA, 969–970
   Global Platform, 965–966
   JAAS, 970
   Java Card API, 964–965
   OpenCard Framework, 966–967
   OpenSC framework, 967
   PAM, 968–969
   PC/SC framework, 966
Encapsulation
   Assertion Builder pattern, 764
   Credential Tokenizer pattern, 807, 812
   Java, 96
   Secure Base Action pattern, 577
   encrypt method, 589
   EncryptDecryptWithAES class, 171–172
   EncryptDecryptWithBlowfish.java program,
   164–165
   <EncryptedData> element, 313–314
   <EncryptedKey> element, 316
Encryption and cryptography, 53
   asymmetric ciphers, 58–62
   in authentication, 52
   in case study, 923
   certificates in. See Certificates and certificate
      keys
   challenges, 77
   hardware-based, 594
   HTTP-POST, 363
   Java, 124–125, 144
   JCA, 151–152
   JCE, 163–168, 178
      AES, 170–172
      PBE, 169–170
   JGSS, 216
   Obfuscated Transfer Object pattern, 664–667
   one-way hash function algorithms, 54–56
   Secure Logger pattern, 578
   Secure Pipe pattern, 598
   signatures in. See Signatures
   SSL in, 69–71
   symmetric ciphers, 56–58
   TLS in, 71–73
   weak, 13
   Web services patterns, 746, 748, 751
   Web tier patterns, 618–619
WS-Security, 337, 346–348
XML. See XML (Extensible Markup Language)
EncryptionHelper class, 582
<EncryptionMethod> element, 314
<EncryptionProperties> element, 316
Encryptor class, 588–589
End-to-end transport layer security, 745
Endpoints in case study, 933
Engine classes
JCA, 154–155
JCE, 162
Enhanced Client and Proxy (ECP) profile, 386
Enrollment systems
biometrics, 986–987, 996
smart card, 973, 975–977
Enterprise-information system (EIS) tier, 227, 266
connector architecture in, 266–270
JDBC in, 272
JMS in, 271–272
Enterprise Java Beans (EJBs), 76
in case study, 913
Container Managed Security pattern, 637
declarative security for, 638
helper classes in, 639
for programmatic security, 643
Enterprise Privacy Authorization Language (EPAL), 405–408
EnterpriseService class, 601
Entitlement in Web services, 294
Entity management, 296
Enveloped Signature transform algorithms, 305
Enveloped signatures, 298–299
Envelopes in SOAP messages, 287
Enveloping signatures
examples, 306–309
XML, 299
<Environment> element, 418
Environment setup in Secure UP, 451
<EnvironmentMatch> element, 417
EPAL (Enterprise Privacy Authorization Language), 405–408
EPCGlobal standards, 36
EPCs (Electronic Product Codes), 35
Equal Error Rate (EER) probability, 982
ERewards Membership Service. See Case study
Errors and error handling
improper, 12
reporting, 694
SPML, 836–837
translation, 672
EventCatalog class, 627
<ExactlyOne> element, 410
Exclusive canonicalization encryption, 320
execute method
Policy Delegate pattern, 675
PolicyDelegateInterface, 673
Secure Base Action pattern, 575
SecureSessionFacadeSessionBean, 684
executeAsPrivileged method, 558
Expertise
Message Interceptor Gateway pattern, 714
problems in, 6
-export option, 122
Exporting
keystore certificates, 122
policies for, 619
Exposure risk factor, 920
Extended SPML operations, 835
Extensibility
Message Inspector pattern, 731
Message Interceptor Gateway pattern, 714
Secure Logger pattern, 587
Secure Message Router pattern, 743
SPML, 836
user account provisioning, 827, 833
Extensible Access Control Markup Language. See XACML (Extensible Access Control Markup Language)
Extensible Markup Language. See XML (Extensible Markup Language)
Extensible Rights Markup Language (XrML), 340
External policy server strategy, 610–611
Extract Adapter pattern, 607
F
Facial recognition, 32
Factor analysis
in case study, 931
for security patterns, 490–493
Factory pattern, 669
Failover
J2EE network topology, 273
service, 672
Failure to Enroll (FTE) probability, 982
False Acceptance Rate (FAR) probability, 982, 997
False Match Rate (FMR) probability, 982
False Non-Match Rate (FNMR) probability, 982
False Reject Rate (FRR) probability, 982, 995
FAR (False Acceptance Rate) probability, 982, 997
Fault handling, 749
Fault tolerance
J2EE network topology, 273
Fault tolerance, continued
Message Interceptor Gateway pattern, 714
Secure Message Router pattern, 743
Web services patterns, 751
FBI survey, 9
Federal regulations, 357, 359
Federal Trade Commission survey, 9
Federated affiliates, 383
Federated data exchange, 391
Federated identity, 360–362, 389
Federated SSO, 86–88, 400
Federation management in Liberty Alliance, 396–397
Federated services, 28
Federation termination protocol, 398
Federations, cross-domain, 88–89
<FederationTerminationNotification> message, 397
fileChanged method, 657–658
Final classes in Java, 96
Financial losses
from confidentiality breaches, 357
reported, 3–4
Financial Privacy Rule, 23
Financial Services Modernization Act, 23–24
findApplicationId method, 877–878
Fine-grained security, 643
Fingerprint matching, 32
approaches to, 981
logical architecture, 983–985
Fingerprints, key, 64
Finished messages, 70
Firewalls
DMZs for, 274
for Java Card applets, 142
Secure Service Proxy pattern, 605
Web services patterns, 746
Web tier patterns, 616
XML, 351, 711–712
Flag values in JAAS, 205–206, 541
Flexibility in security patterns
Intercepting Web Agent, 611
Secure Base Action, 576
Secure Service Proxy, 605
Single Sign-on Delegator, 786
FMR (False Match Rate) probability, 982
FNMR (False Non-Match Rate) probability, 982
Forces in security pattern templates, 466
Assertion Builder, 757
Audit Interceptor, 625
Authentication Enforcer, 536
Authorization Enforcer, 549
Container Managed Security, 635
Credential Tokenizer, 802
Dynamic Service Management, 645–646
Intercepting Validator, 561
Intercepting Web Agent, 607
Message Inspector, 716–717
Message Interceptor Gateway, 706
Obfuscated Transfer Object, 679–680
Password Synchronizer, 841
Policy Delegate, 668–669
Secure Base Action, 560
Secure Logger, 578
Secure Message Router, 733–734
Secure Pipe, 591
Secure Service Facade, 678
Secure Service Proxy, 599
Secure Session Object, 687
Single Sign-on Delegator, 777
Form-based authentication
J2EE, 233, 241–243
in web.xml, 545–546
Form-POST-based redirection, 393
Form validation
Web tier patterns, 619
in XML using Apache Struts, 565
Foundstone Enterprise testing tool, 953
Fowler, Martin, 529
Frameworks, security
adopting, 525–529
in Secure Service Facade pattern, 681
Front Controller pattern, 478, 573, 577
FRR (False Reject Rate) probability, 981, 997
FTE (Failure to Enroll) probability, 981
Full View with Errors pattern, 461
G
Gambling casino, 9
Gang of Four (GoF) design patterns, 456
Gartner Group report, 8–9
Gateways
Message Interceptor Gateway pattern. See Message Interceptor Gateway pattern
Parlay, 405
generateKey method, 163
Generic products, XACML for, 412
-genkey option, 119
genKeyPair method, 157
genPrivateKey method, 157
genPublicKey method, 157
gerAction method, 405
gerAlgorithm method, 156
gerAllConfigContext method, 860
gerApplicationBufferSize method, 196
gerAssertionReply method, 773
gerAuthenticationMethod method, 771–773
getCallerPrincipal method, 262, 639
getCallersIdentity method, 262
getConfigFile method, 857
getConfigProperties method, 797
getConnection method, 269
getContents method, 873–874
getContext method, 859
data method, 667
getEncoded method, 156
getInstance method
Cipher, 164
KeyAgreement, 173
KeyGenerator, 163
KeyPairGenerator, 157
MBeanFactory, 656
MBeanManager, 652–653
MessageDigest, 156
Signature, 158
getMaxInactiveInterval method, 251
getPacketBufferSize method, 196
getPermissions method, 106
getPrincipal method, 810
getProtectionDomain method, 106
getProtocolBinding method
AssertionContextImpl, 774
SSOContextImpl, 798
TokenContextImpl, 810
getRegistryFileName method, 657
getRemoteUser method, 253, 539
getSecurityInfo method, 258
getSecurityManager method, 104
getServiceName method, 798
getServiceStatus method, 795
getSession method, 249–250
getSessionInfo method, 797
getSSODelegator method, 793
getSSOTokenMap method, 794
getStatus method, 800
getSubject method, 541
getToken method
TokenContextImpl, 810
UsernameToken, 811
getUserPrincipal method, 254–255, 539, 639
GINA (Graphical Identification and Authentication), 967–968
GINA Module
for biometrics, 987
for smart cards, 973
GLB (Gramm-Leach-Bliley) Act, 23–24
Global logout
in identity management, 361

Liberty Alliance, 400–401
SAML, 383–384
Single Sign-on Delegator pattern, 784–786
Global Platform technology, 963–964
Goals, security, 50–53
GoF (Gang of Four) design patterns, 456
Gramm-Leach-Bliley (GLB) Act, 23–24
Grant statement, 210–211
Granularity
Container Managed Security pattern, 644
Intercepting Web Agent pattern, 615
Graphical Identification and Authentication (GINA), 969–970
Groups
identity management, 360
J2EE, 238
GSS-API, 544

H
HA. See High availability (HA)
Hand geometry, 32
handle method, 208–209
Handlers, 527
Handshake messages, 69–71
Hardening, 8
in Web services patterns, 745
in Web tier security, 616
Hardware acceleration, 175
Hardware cryptographic devices
Web services patterns, 746
Web tier patterns, 594
Headers in SOAP messages, 287
Health Insurance Privacy and Portability Act (HIPPA) of 1996, 24–25
Helper classes
AuditInterceptor, 632–633
EJB, 639
Hierarchical resource profiles, 417
High availability (HA)
in case study, 917
Identity management patterns, 813
J2EE network topology, 277
Message Interceptor Gateway pattern, 714
Secure Message Router pattern, 743
in use cases, 910
HIPPA (Health Insurance Privacy and Portability Act) of 1996, 24–25
HMAC-MD5 algorithm, 172–173
Honeypots, 746
Horizontal scaling
Business tier patterns, 480
J2EE network topology, 274–275
Index

Host operating systems, 8
Host security
  in case study, 915
  using JSSE, 193–194
  in Web services patterns, 746
HostnameVerifier class, 186
Hot Standby pattern, 465
HTTP
  basic authentication, 233, 240, 243
  digest authentication, 233
  POST messages
    identity management, 363
    SAML, 385, 387
    Web tier patterns, 618–619
  redirection, 393
  sessions
    cookies and URL rewriting in, 248–251
    in Web tier patterns, 620
HTTPS (HTTP over SSL)
  J2EE, 191–193, 243–244, 255–256, 258
  for Web services, 278
HTTPSClientUsingProxyTunnel.java program, 192–193
HttpServletRequest objects, 253–254, 539
HttpsURLConnection class, 185
Hub in Security Wheel, 442

I
I/O, non-blocking, 194–197
ID-FF (Identity Federated Framework) version 1.2, 390
ID-SIS (Identity Service Interface Specification) version, 390
ID-WSF (Identity Web Services Framework) version, 390
IDEA symmetric cipher, 79
Identification processes, 6
Identity Federated Framework (ID-FF) version 1.2, 390
Identity federation, 28, 360–362, 389
cross-domain, 88–89
Liberty Alliance, 396–397
SAML, 365
Identity management, 27, 83–84, 357, 755–756
  access control. See Access control
  access management services, 28
  auditing, 28–29, 361, 813
  in case study, 915–916, 937
  core issues, 358–360
  data synchronization services, 27
  directory services, 28
  importance, 362
  justifying, 38–39
Liberty Alliance Project. See Liberty Alliance Project
network identity, 360–362
personal identification. See Personal identification
point-to-point interfaces for, 822
policies, 496
provisioning services, 27
references, 89–91, 432–436
reporting services, 28–29
SAML. See SAML (Security Assertion Markup Language)
security patterns for, 466, 477, 486–488
  Assertion Builder, 756–775
  best practices, 813–814
  Credential Tokenizer, 802–813
  factor analysis, 492
  Password Synchronizer, 890
  pitfalls, 813–814
  references, 814–815
  Single Sign-on Delegator, 776–802
  Web services, 749–750
service provisioning relationship, 819–820
services for, 528
single sign-on. See Single sign-on (SSO)
  mechanisms
  summary, 89, 432–433
  in use cases, 908
XACML. See XACML (Extensible Access Control Markup Language)
Identity Provider Agent strategy, 728–730
Identity Provider Discovery Profile, 367
Identity providers, 86–88
  in case study, 916
  Liberty Alliance, 391
  Liberty specifications, 388
  Message Inspector pattern, 721
  Message Interceptor Gateway pattern, 711
  Secure Message Router pattern, 738
  Single Sign-on Delegator pattern, 782, 784
  user account provisioning, 830–832
  Web services, 704
Identity Service Interface Specification (ID-SIS) version, 390
Identity termination strategy, 786
Identity theft and spoofing, 9
  Assertion Builder pattern, 774
  Web services, 292
Identity Web Services Framework (ID-WSF) version, 390
IDSs (Intrusion Detection Systems), 954
IETF Policy Management Working Group, 404
Index

IMAP (Internet Message Access protocol), 216
Impact risk factor, 920
Implementation
Assertion Builder pattern, 764–768
AssertionContextImpl class, 770–774
biometrics, 981–987
JAAS authorization, 210–215
LoginModule class, 200–204
Policy Delegate pattern, 673–675
Secure UP, 446, 450–451
smart cards, 968–969
SPML, 837–840
UserNameToken class, 811
implies method, 553
-import option, 122–123
Importing certificates, 115, 122–123
Inclusive canonicalization encryption, 319–320
Information aggregators, 358
<information> element, 117
Informative policies, 497
Infrastructure
Application Security Provider, 527
in case study, 915, 931–932, 934–935
J2EE, 229–230
policies, 496
in security patterns, 462–465
Business tier, 693–694
factor analysis, 491
Intercepting Web Agent, 614
Password Synchronizer, 888
Secure Pipe, 597–598
Web services, 702–705, 745–746
Web tier, 616–617
Security Services, 527–528
init method
AuditClient, 632
Cipher, 164
HTTPProxy, 606
MBeanFactory, 656
MBeanManager, 653
PasswordSyncLedger, 882–883
PasswordSyncListener, 871–872
Policy Delegate pattern, 674
SimpleSOAPServiceSecurePolicy, 604
TakeAction, 879–880
WriteFileApplet, 110
initConfig method
ServiceConfig, 857
SSODelegatorFactoryImpl, 792
initialize method
KeyPairGenerator, 157
LoginModule, 201
initSign method, 158
initVerify method, 158
Injection flaws, 11
Input validation failures, 10
Insider attacks, 9
Integration and Integration tier
in case study, 951–952
in identity management, 359
J2EE, 227
in patterns-driven design, 488
reality checks, 520–523
rule-based, 683
in security patterns, 462–463
Assertion Builder, 764
Intercepting Web Agent, 611–612, 614
Password Synchronizer, 889
Secure Service Facade, 683
Secure Service Proxy, 605
user account provisioning, 829–833
Integrity
Secure Pipe pattern, 595, 597–598
as security goal, 51
in Security Wheel, 442
Web services, 295
Intellectual property, 960
Intercepting Filter pattern
and Audit Interceptor pattern, 634
and Authentication Enforcer pattern, 548
and Intercepting Validator pattern, 561
Intercepting Validator pattern, 468
in case study, 924, 926, 929, 936, 953
consequences, 568–569
forces, 561
participants and responsibilities, 562, 564
problem, 560–561
reality check, 569
related patterns, 569
and Secure Base Action pattern, 573, 577
security factors and risks, 569
solution, 561
strategies, 564–568
structure, 562–563
Intercepting Web Agent pattern, 472
in case study, 926
consequences, 611–612
forces, 607
participants and responsibilities, 608–610
problem, 607
reality check, 615
related patterns, 615
Container Managed Security, 645
Secure Service Proxy, 606
Intercepting Web Agent pattern, continued
  sample code, 612–614
  security factors and risks, 614
  solution, 607–608
  strategies, 610–611
  structure, 608

Intercepting Web Agent strategy, 712–713

Interceptor strategy, 690–691

Interfaces
  CertPath, 179–180
  JAAS, 198–200
  JCA, 153–155
  JCE, 162
  JSSE, 184–186
  Password Synchronizer pattern, 850–851
  PKCS#11 and PKCS#15 standards, 964
  Policy Delegate pattern, 673
  Secure Service Facade pattern, 683–684

Intermediary infrastructure, 734

Internet Message Access protocol (IMAP), 216

Interoperability
  Liberty Phase 1, 389
  Secure Message Router pattern, 744
  Secure Pipe pattern, 595
  security provisioning patterns, 892
  user account provisioning, 827
  Web services, 296

Intrusion Detection Systems (IDSs), 954

Invalid data. See Intercepting Validator pattern

Invalidating HTTP sessions, 250

Invocation, rule-based, 683

invoke method, 727

IP address capture, 775

IP filtering, 745

Iris verification, 34

isAuthorized method, 552

isCallerInRole method
  EJBContext, 639
  J2EE authorization, 553

<<IssuesInstant>> element, 418

Issuing authority in SAML, 377

isUserInRole method, 254, 539, 639

isValidStatement method, 771

Iterative development in Secure UP, 452

ITS4 testing tool, 953

J

J2EE (Java 2 Enterprise Edition) platform, 225–226
  architecture and logical tiers, 226–228
  authorization, 229, 235–236, 553
  declarative, 259–261
  programmatic, 261–262
  web-tier, 251–255
  for biometrics, 985, 987
  clients, 226–227, 233, 255–258
  component security. See Component security
  container-based security. See Container-based security
definitions in, 228–229

EIS tier, 227, 266
canector architecture in, 266–270

JDBC in, 272

JMS in, 271–272

EJB tier. See EJB tier in J2EE infrastructure, 229–230

LDAP in, 76

network topology, 273–274
  horizontal scalability, 274–275
  vertical scalability, 276–277

references, 280–281

SAML in, 386–387

security patterns for, 466

for smart cards, 972

summary, 279–280

for Web services, 257–259, 351

J2ME (Java 2 Micro Edition)
  architecture, 126–127
  configurations, 127–130

MIDlets in, 131–134

profiles, 130–132

J2SE (Java 2 Standard Edition)
  J2SE 5.0, 124
  in Obfuscated Transfer Object pattern, 664

JAAS (Java Authentication and Authorization Service), 968

authentication, 200
  in clients, 206–210

LoginModule, 200–206, 541–544, 972

Web tier, 245–246

authorization, 210
  implementing, 210–215

strategy, 556–559

for biometrics, 987

classes and interfaces, 198–200

client-side callbacks, 256–257

vs. JGSS, 215–216

JAAS Authorization policy file, 557

JAAS Module, 556

JACC (Java Authorization Contract for Containers), 236, 553

JAD (Java application descriptor) files, 131

JADTool utility, 135–136
JAR (Java archive format) files, 97
   for signed applets, 112–114
   signing, 125
   verifying, 126
   in Web tier patterns, 620
Jarsigner tool, 125–126
   for signed applets, 113–114
   for smart cards, 178
Java 2 Enterprise Edition (J2EE) platform. See J2EE (Java 2 Enterprise Edition) platform
Java 2 Micro Edition (J2ME)
   architecture, 126–127
   configurations, 127–130
   MIDlets in, 131–134
   profiles, 130–132
Java 2 platform security, 93–94, 525
   applet security, 109–112
   Java Card, 140–142
   signed, 112–115
   for smart cards, 995
biometrics, 987
CertPath, 179–182
code, 143
   obfuscation, 144–145
   reverse engineering, 143–144
   extensible, 149–151
   importance, 36–37
J2EE. See J2EE (Java 2 Enterprise Edition) platform
J2ME. See Java 2 Micro Edition (J2ME)
JAAS. See JAAS (Java Authentication and Authorization Service)
Java Card technology, 30–31, 136
   API framework, 962–963
   applets in, 140–142
   development kit, 142–143
   model for, 140
   smart cards, 136–139
JCA. See JCA (Java Cryptography Architecture)
JCE. See JCE (Java Cryptographic Extensions)
JGSS, 215–216
JSSE. See JSSE (Java Secure Socket Extension)
JVM, 94–95
JWS, 116–118
language, 95–97
MIDlets
   components of, 131–132
   signed, 134–136
   trusted, 132–134
   references, 146
   reusable components, 649
SASL, 216–217
   clients, 217–218, 230
   installing, 219–220
   servers, 218–219
security model, 97–98
   AccessController, 106
   bytecode verifiers, 108
   ClassLoader, 108–109
   codebase, 106–107
   CodeSource, 107
   permissions, 101–103
   policies, 103–104
   protection domains, 98–99
   SecurityManager, 104–105
summary, 145–146
tools, 118
   jarsigner, 125–126
   keystores, 118, 124–125
   keytool, 118–124
   policytool, 125
Web services, 350–353
Java 2 Standard Edition (J2SE)
   J2SE 5.0, 124
   in Obfuscated Transfer Object pattern, 664
Java Access Manager, 391
Java API for XML Registry (JAXR), 277
Java application descriptor (JAD) files, 131
Java archive format (JAR) files, 97
   for signed applets, 112–114
   signing, 125
   verifying, 126
   in Web tier patterns, 620
Java Authentication and Authorization Service. See JAAS (Java Authentication and Authorization Service)
Java Authorization Contract for Containers (JACC), 236, 553
Java Card runtime environment (JCRE), 138–139
Java Card technology, 30–31, 136
   API framework, 962–963
   applets in, 140–142
   development kit, 142–143
   model for, 140
   smart cards, 136–139
Java Card Workstation Development Environment (JCWDE), 142–143
Java Certification Path, 179–182
Java Cryptographic Extension Keystores (JCEKS), 119
Java Cryptographic Extensions. See JCE (Java Cryptographic Extensions)
Index

Java Cryptography Architecture. See JCA (Java Cryptography Architecture)
Java Data Objects (JDO), 915, 940
Java Database Connectivity (JDBC), 266–267, 272
Java Development Kit (JDK), 97
Java Generic Secure Services (JGSS), 215–216
Java GSS-API, 544
Java Management Extension (JMX) technology, 652
Java Message Service (JMS)
  Audit Interceptor pattern, 633–634
  EIS tier, 266, 271–272
Java Naming and Directory Interface (JNDI), 76
Java native code (JNI), 128
Java Network Launch protocol (JNLP), 116–117
Java Secure Socket Extension (JSSE). See JSSE (Java Secure Socket Extension)
Java.security file, 204–205
Java System Access Manager, 352, 401–402
Java system web server, 612–614
Java Virtual Machine (JVM), 94–95
Java Web Services Developer Pack (JWSDP), 350–351
Java Web Start (JWS) security, 116–118
Javac command, 113
Javax.net.* package, 184
Javax.net.ssl.* package, 184
Javax.security.auth package, 198
Javax.security.cert.* package, 184
JAX-RPC API
  in case study, 913
  in Message Inspector pattern, 724
  for Web services, 279
JAXR (Java API for XML Registry), 277
JCA (Java Cryptography Architecture), 151–152
API, 155–159
classes and interfaces, 153–155
cryptographic services, 152–153
digital signature generation, 158–159
key pair generation, 156–157
message digests, 155–156
JCE (Java Cryptographic Extensions), 159–160
Advanced Encryption Standard, 170–172
API, 163
classes and interfaces, 162
Cryptographic Service Providers, 160–161
encryption and decryption, 163–168, 178
hardware acceleration, 175
key agreement protocols, 173–175
MAC objects, 172–173
Password-Based Encryption, 169–170
sealed objects, 168–169
smart card support, 176–178
strong vs. unlimited strength cryptography, 178
JCEKS (Java Cryptographic Extension Keystores), 119
JCRE (Java Card runtime environment), 138–139
JCWDE (Java Card Workstation Development Environment), 142–143
JDBC (Java Database Connectivity), 266–267, 272
JDK (Java Development Kit), 97
JDO (Java Data Objects), 915, 940
JGSS (Java Generic Secure Services), 215–216
JiffyXACML, 416
JKS (Java keystores), 119
JMS (Java Message Service)
  in Audit Interceptor pattern, 633–634
  in EIS tier, 266, 271–272
JMX (Java Management Extension) technology, 652
JNDI (Java Naming and Directory Interface), 76
JNI (Java native code), 128
JNLP (Java Network Launch protocol), 116–117
Journaled Component pattern, 465
JSPs, 554–555
JSSE (Java Secure Socket Extension), 182–183
  API programming in, 186
  application layer using, 594–595
  classes and interfaces, 184–186
  client-side communication, 186–188
  host name verification, 193–194
  HTTP over SSL, 191–193
  vs. JGSS, 215–216
  mutual authentication, 190–191
  providers, 183–184
  secure socket connections using SSL, 186–190
  server-side communication, 188–190
  SSL Engine and non-blocking I/O, 194–197
Justifications, 37–38
  compliance, 42
  identity and access management, 38–39
  proactive security approaches, 39–41
JVM (Java Virtual Machine), 94–95
JVMDI debugger, 128
JVMPI profiling code, 128
JWS (Java Web Start) security, 116–118
JWSDP (Java Web Services Developer Pack), 350–351

K
K Virtual Machine (KVM), 129–130
Kerberos
  in JAAS Login Module strategy, 544
  in JGSS, 215–216
Kerievsky, Joshua, 529
Index

Key agreement encryption
- JCE, 173–175
- XML, 318

KeyAgreement class, 162, 173–175
- -keyalg option, 120

KeyFactory class, 154

KeyGenerator class, 162
- <KeyInfo> element, 302

KeyPairGenerator class, 154
- -keypass option, 120

Keys and key management
- in assessment checklists, 500
- asymmetric ciphers, 58–62
- in certificates. See Certificates and certificate keys

Credential Tokenizer patterns, 812

KeyStore class, 154
- -keystore option, 120

KeyStore property, 191

Keystore tool, 109, 118

Keystores, 109
- cryptographic devices based, 124–125
- databases, 119
- deleting, 124
- exporting certificates from, 122
- importing certificates to, 122–123
- listing entries in, 121–122
- passwords for, 124

Keytool tool, 109, 118–119
- for CSRs, 124
- for deleting keystores, 124
- for exporting keystore certificates, 122
- for importing keystore certificates, 122–123
- for keystore databases, 119
- for keystore entry lists, 121–122
- for keystore passwords, 124
- for printing certificate information, 123
- for private/public key pairs, 119–121
- for signed applets, 113–114
- for smart cards, 176–178

Ktoolbar tool, 136

KVM (K Virtual Machine), 129–130

L

Labeling
- in security patterns, 498
- in Security Wheel, 443

Layered Security pattern, 464

LDAP (Lightweight Directory Access Protocol), 73–76
- certificate revocation issues, 78
- cryptography challenges, 77
- J2EE, 76
- key management, 77–78
- random number generation, 77
- SASL, 216, 219
- trust models, 79–81

Leaf nodes in attack trees, 82–83

Ledger, 842, 853

Legacy systems
- Intercepting Web Agent pattern, 611–612, 614
- Password Synchronizer pattern, 889
- Secure Service Proxy pattern, 605

Lessons learned in case study, 955–956

Liberty Alliance consortium, 361

Liberty Alliance Project, 387–389
- architecture, 391–392
- Liberty Phase 1, 389–390
- Liberty Phase 2, 390–391
- meta-data and schemas, 394
- relationships, 392–393
- for SAML, 364
- security mechanisms, 394–395
- SSO strategy, 740–742
- usage scenarios, 395–396

Liberty-enabled clients, 389

Liberty-enabled proxies, 389

Libraries
- DLL, 967
- tag, 554–556
### Index

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Licenses in WS-Security</td>
<td>343–344</td>
</tr>
<tr>
<td>Limited View pattern</td>
<td>461</td>
</tr>
<tr>
<td>-list option</td>
<td>121–122</td>
</tr>
<tr>
<td>Load-balancing in case study</td>
<td>923</td>
</tr>
<tr>
<td>Load Balancing PEP pattern</td>
<td>463</td>
</tr>
<tr>
<td>loadRegistry method</td>
<td>656</td>
</tr>
<tr>
<td>Locate service in X-KISS</td>
<td>327–329</td>
</tr>
<tr>
<td>LogFactory class</td>
<td>580</td>
</tr>
<tr>
<td>Logging</td>
<td>577–590</td>
</tr>
<tr>
<td>alteration detection for biometrics</td>
<td>587, 590</td>
</tr>
<tr>
<td>in biometrics</td>
<td>997</td>
</tr>
<tr>
<td>in case study</td>
<td>917</td>
</tr>
<tr>
<td>failures</td>
<td>15</td>
</tr>
<tr>
<td>in security patterns</td>
<td></td>
</tr>
<tr>
<td>Identity management</td>
<td>813</td>
</tr>
<tr>
<td>Password Synchronizer</td>
<td>888–889</td>
</tr>
<tr>
<td>Policy Delegate</td>
<td>676</td>
</tr>
<tr>
<td>Secure Base Action</td>
<td>576</td>
</tr>
<tr>
<td>Secure Logger. See Secure Logger pattern</td>
<td></td>
</tr>
<tr>
<td>Single Sign-on Delegator</td>
<td>801</td>
</tr>
<tr>
<td>Web services</td>
<td>749</td>
</tr>
<tr>
<td>in Security Wheel</td>
<td>443</td>
</tr>
<tr>
<td>Logging out in JAAS</td>
<td>209–210</td>
</tr>
<tr>
<td>LoggingHandler.java file</td>
<td>726–728</td>
</tr>
<tr>
<td>Logical access control</td>
<td>960–962</td>
</tr>
<tr>
<td>Logical architecture</td>
<td></td>
</tr>
<tr>
<td>biometric systems</td>
<td>981–985</td>
</tr>
<tr>
<td>smart cards</td>
<td>969–973</td>
</tr>
<tr>
<td>user account provisioning</td>
<td>826–828</td>
</tr>
<tr>
<td>Logical tiers in J2EE</td>
<td>226–228</td>
</tr>
<tr>
<td>Logical views in use cases</td>
<td>912</td>
</tr>
<tr>
<td>Login attempts in biometrics</td>
<td>997</td>
</tr>
<tr>
<td>login method</td>
<td></td>
</tr>
<tr>
<td>Authentication Enforcer</td>
<td>546–547</td>
</tr>
<tr>
<td>LoginContext</td>
<td>206–208, 214, 541</td>
</tr>
<tr>
<td>LoginModule</td>
<td>201–202</td>
</tr>
<tr>
<td>Login service in case study</td>
<td>941–943</td>
</tr>
<tr>
<td>LoginContext class</td>
<td></td>
</tr>
<tr>
<td>JAAS authentication</td>
<td>199–200, 206–209</td>
</tr>
<tr>
<td>JAAS Login Module strategy</td>
<td>541, 544</td>
</tr>
<tr>
<td>LoginModule class</td>
<td>199</td>
</tr>
<tr>
<td>Authentication Enforcer pattern</td>
<td>541</td>
</tr>
<tr>
<td>biometrics</td>
<td>985</td>
</tr>
<tr>
<td>implementing</td>
<td>200–204</td>
</tr>
<tr>
<td>providers for</td>
<td>204–206</td>
</tr>
<tr>
<td>smart cards</td>
<td>972</td>
</tr>
<tr>
<td>LogManager class</td>
<td>580</td>
</tr>
<tr>
<td>logout method</td>
<td></td>
</tr>
<tr>
<td>LoginContext</td>
<td>209</td>
</tr>
<tr>
<td>LoginModule</td>
<td>204</td>
</tr>
<tr>
<td>Logout requests in SAML</td>
<td>366</td>
</tr>
<tr>
<td>Loosely coupled architecture</td>
<td></td>
</tr>
<tr>
<td>Intercepting Web Agent pattern</td>
<td>611</td>
</tr>
<tr>
<td>Secure Service Proxy pattern</td>
<td>605</td>
</tr>
<tr>
<td>Lost smart cards</td>
<td>997</td>
</tr>
<tr>
<td>Low-level KVM security</td>
<td>129–130</td>
</tr>
<tr>
<td>MAC (message authentication code)</td>
<td>159, 172–173</td>
</tr>
<tr>
<td>Mac class</td>
<td>162</td>
</tr>
<tr>
<td>Magnus.conf file</td>
<td>613–614</td>
</tr>
<tr>
<td>Maintainability in security patterns</td>
<td></td>
</tr>
<tr>
<td>Audit Interceptor</td>
<td>630</td>
</tr>
<tr>
<td>Intercepting Web Agent pattern</td>
<td>608</td>
</tr>
<tr>
<td>Message Inspector</td>
<td>731</td>
</tr>
<tr>
<td>Message Interceptor Gateway</td>
<td>713</td>
</tr>
<tr>
<td>Secure Message Router</td>
<td>743</td>
</tr>
<tr>
<td>Malicious code prevention. See Intercepting Validator pattern</td>
<td></td>
</tr>
<tr>
<td>Man-in-the-middle (MITM) attacks</td>
<td>16</td>
</tr>
<tr>
<td>in case study</td>
<td>921</td>
</tr>
<tr>
<td>in SAML, 385–386</td>
<td></td>
</tr>
<tr>
<td>in Web services</td>
<td>291</td>
</tr>
<tr>
<td>Manageability</td>
<td></td>
</tr>
<tr>
<td>J2EE network topology</td>
<td>274</td>
</tr>
<tr>
<td>Secure Base Action pattern</td>
<td>574</td>
</tr>
<tr>
<td>Secure Logger pattern</td>
<td>587</td>
</tr>
<tr>
<td>&lt;Manifest&gt; element</td>
<td>302</td>
</tr>
<tr>
<td>Manifest files</td>
<td>131</td>
</tr>
<tr>
<td>Manipulation attacks</td>
<td>291–292</td>
</tr>
<tr>
<td>Mapping</td>
<td></td>
</tr>
<tr>
<td>in Container Managed Security pattern</td>
<td>640</td>
</tr>
<tr>
<td>SAML attributes</td>
<td>368</td>
</tr>
<tr>
<td>user account</td>
<td>822</td>
</tr>
<tr>
<td>Masked list strategy</td>
<td>662–664</td>
</tr>
<tr>
<td>Match-off-the-card strategy</td>
<td>967, 994</td>
</tr>
<tr>
<td>Match-on-the-card strategy</td>
<td>967, 994, 997</td>
</tr>
<tr>
<td>Matrix, Effect</td>
<td>455</td>
</tr>
<tr>
<td>MBean strategy</td>
<td>649–650</td>
</tr>
<tr>
<td>MBeanFactory class</td>
<td>648</td>
</tr>
<tr>
<td>MBeanFactory.java file</td>
<td>655–658</td>
</tr>
<tr>
<td>MBeanManager.java file</td>
<td>652–655</td>
</tr>
<tr>
<td>MBeanServer class</td>
<td>648</td>
</tr>
<tr>
<td>MD5 cryptography</td>
<td>54–56</td>
</tr>
<tr>
<td>Cryptographic Service Providers</td>
<td>153</td>
</tr>
<tr>
<td>JCA message digests</td>
<td>155–156</td>
</tr>
<tr>
<td>Media in security pattern factor analysis</td>
<td>493</td>
</tr>
<tr>
<td>Memory</td>
<td></td>
</tr>
<tr>
<td>for Secure Session Object pattern</td>
<td>691</td>
</tr>
<tr>
<td>in smart cards</td>
<td>137–138</td>
</tr>
<tr>
<td>Memory cards</td>
<td>30</td>
</tr>
<tr>
<td>Message authentication code (MAC)</td>
<td>159, 172–173</td>
</tr>
</tbody>
</table>
Message authentication encryption, 319
Message Configurators, 736, 738
Message digests
  encryption algorithms for, 319
  JCA, 155–156
Message-handler chain strategy, 722–728
Message injection attacks, 291–292
Message Inspector pattern, 475, 486
  in case study, 918, 926, 930, 933
  consequences, 729–731
  forces, 716–717
  participants and responsibilities, 719–721
  problem, 715–716
  reality checks, 731–732
  related patterns, 732
    Intercepting Validator, 569
    Message Interceptor Gateway, 715
    Secure Message Router, 744
  security factors and risks, 731
  solution, 717–719
  strategies, 721–730
  structure, 719
Message Interceptor Gateway pattern, 476, 486
  in case study, 918, 926, 933
  consequences, 713–714
  forces, 706
  participants and responsibilities, 709–711
  problem, 705
  reality check, 715
  related patterns, 715
    Audit Interceptor, 635
    Intercepting Web Agent, 615
    Message Inspector, 721, 732
    Secure Message Router, 746
  security factors and risks, 714–715
  solution, 707–709
  strategies, 711–713
  structure, 709
Message replay
  SAML, 385–387
  security provisioning patterns, 894
Message Routers, 736
MessageDigest class
  JCA, 154
  Secure Logger pattern, 582
Messages and message-level security
  in case study, 917, 921
  encryption, 748
  PasswordSyncListener, 886–887
  SAML, 374–376
    for password authentication, 376–378
    protocols for, 366
  translation, 672
  in use cases, 908
  validation abuses, 293
  Web services, 279, 702, 747–748
Meta-data and schemas, 389, 394
Methodology choices in use cases, 905–906
Methods, Java, 96
Microprocessor cards, 30
MIDlets, 258
  components of, 131–132
  signed, 134–136
  trusted, 132–134
MidP (Mobile Information Device Profile), 130–132
Migration
  in Message Interceptor Gateway pattern, 714
  SAML 1.1 to SAML 2.0, 775
Mimic scanner attacks, 997
Minimization and hardening in Web services patterns, 745
Minutiae-based fingerprint matching, 981
MITM (man-in-the-middle) attacks, 16
  in case study, 921
  in SAML, 385–386
  in Web services, 291
Mobile Information Device Profile (MIDP), 130–132
Model MBean strategy, 649–650
Model-View-Controller (MVC) architecture
  Authentication Enforcer pattern, 539
  Intercepting Validator pattern, 564
  Secure Base Action pattern, 573–574
Models
  biometrics, 985–989
  conceptual, 915–927
  data, 939–940
  domain, 370–371
  JWS security, 116–117
  smart cards, 968–969
  threat, 81–83
  trust, 79–81, 495, 937
  user account provisioning, 823–826, 832–833
  Web services, 285–286
Modification attacks
  SAML, 385
  Secure Logger pattern, 582, 587
  Modify operations in SPML, 834
  ModifyResponse message, 834
Modularity
  Message Inspector pattern, 731
  Message Interceptor Gateway pattern, 713
  Secure Message Router pattern, 743
Index

Monitoring
biometrics, 997
Business tier patterns, 696
in case study, 954
Secure UP, 447, 452
Security Services, 527
user account provisioning, 828
Web services patterns, 750
Web tier patterns, 617
Multi-factor and multi-tiered authentication
Liberty Alliance sessions, 399
using smart cards and biometrics, 993–997
Multi service controller, 602–603
Multiple circles of trust, 801
Multiple login attempts, 997
Multiple resource profiles, 416–417
Multiple sign-ons, 16, 786
Multithreading strategy, 849
Mutual authentication
J2EE, 244
JSSE, 190–191
Web tier patterns, 618
MVC (Model-View-Controller) architecture
Authentication Enforcer pattern, 539
Intercepting Validator pattern, 564
Secure Base Action pattern, 573–574
MyJAASaux.policy file, 557–558
MyPrivilegedAction.java file, 559
MySSLClient.java program, 187–188
MySSLServer.java program, 188–190
MyTestAction.java program, 213
MyTestLoginModule class, 200–204

N
Name Identifier Management Profile, 367
Name-value (NV) pairs, 664
<brnameidentifiermappingrequest> message, 396–397
Names. See Identity management
Namespaces in WS-Security, 342
Naming service restrictions, 694
Native code, Java, 128
Negotiations in WS-Policy, 411
Network appliance strategy, 594
Network identity, 360–362
Network-Layer security, 701
Network perimeter security
in case study, 915, 932
Web Services, 702–703, 745
Network policies, 496
Network responsiveness, 714
Network services, 7–8
Network topology in J2EE, 273–274
horizontal scalability, 274–275
vertical scalability, 276–277
Nodes, attack tree, 82–83
Non-blocking I/O, 194–197
Non-HTTP packets in Web tier patterns, 616
Non-repudiation, 53, 295
Notice of security breach requirements, 26
Notification messages, 887
NV (name-value) pairs, 664

O
OASIS standards
in identity management, 361
OASIS Web services. See WS-Security
XCBF, 33–34
Obfuscated Transfer Object pattern, 474
in case study, 918, 924, 929, 936–937
consequences, 666
forces, 659–660
participants and responsibilities, 660–662
problem, 659
reality check, 667–668
related patterns, 668
sample code, 666–667
security factors and risks, 667
solution, 660
strategies, 662–666
structure, 660
Obfuscation
Business tier patterns, 695
in case study, 924, 933
Java code, 144–145
Web tier patterns, 620
Obj.conf file, 612–613
<object> element, 302
Object Name Service (ONS), 35
Objects in case study, 939–940
OCF (OpenCard Framework), 964–965
OCSP (Online Certificate Status Protocol), 68–69,
179, 976–977
ODRL (Open Digital Rights Language), 340
One-to-many/one-to-one Policy Delegate, 672
One-way hash function algorithms, 54–56
Oneshot MIDlets, 133
onFault method, 728
Online Certificate Status Protocol (OCSP), 68–69,
179, 976
Online portals, 911–912, 914
onMessage method
AuditRequestMessageBean, 631
PasswordSyncLedger, 883–884
PasswordSyncListener, 874–876
ONS (Object Name Service), 35
Open Content model, 836
Open Digital Rights Language (ODRL), 340
OpenCard Framework (OCF), 964–965
OpenSC framework, 965
Operating systems, 8, 616
Operational models
biometrics, 985–989
smart cards, 973–977
Web services, 285–286
Operational practices, 6
Operations
Secure UP, 449
SPML, 834–835
Optical scanners, 985
Optimization, 20
Optional flag, 206
Orders in case study
fulfillment, 911, 948–951
management, 946–948
placement, 911
Origin host verification, 746
OS (operating systems), 8, 616
Output sanitation, 10
Over-The-Air (OTA) provisioning techniques, 136
Overflow, buffer, 11

P
Padding in JCE block ciphers, 166
paint method, 111
PAM (Pluggable Authentication Module), 197, 966–967
for biometrics, 985
for smart cards, 973
PAFs (Policy Administration Points), 369, 373
ParamValidator class, 562
Parlay Group, 403, 405
Partial content of XML documents, accessing, 412
Participants and responsibilities in security patterns
Assertion Builder, 760–762
Audit Interceptor, 627
Authentication Enforcer, 537–538
Authorization Enforcer, 550–551
Container Managed Security, 636–638
Credential Tokenizer, 805–806
Dynamic Service Management, 646, 648
Intercepting Validator, 562, 564
Intercepting Web Agent, 608–610
Message Inspector, 719–721
Message Interceptor Gateway, 709–711
Obfuscated Transfer Object, 660–662
Password Synchronizer, 845–848
Policy Delegate, 670–671
Secure Base Action, 571, 573
Secure Logger, 579–580
Secure Message Router, 736–738
Secure Service Facade, 679, 681
Secure Service Proxy, 600–601
Secure Session Object, 688–689
Single Sign-on Delegator, 780–782
Partitioning in network topology, 273
Passive RFID tags, 35
Password-Based Encryption (PRE), 169–170
Password Manager, 828
Password Synchronizer Manager, 842–848
Password Synchronizer pattern, 477, 486
consequences, 850–851
forces, 841
participants and responsibilities, 845–848
problem, 840–841
reality check, 890–891
related patterns, 891
sample code, 851–887
security factors and risks, 888–890
solution, 842–843
strategies, 849–850
structure, 843–844
Passwords
in authentication, 51–52
Credential Tokenizer patterns, 812
exploits, 13
Identity management, 84, 814
JAAS authorization, 214
keystore, 124
SAML, 376
smart cards, 31
synchronization, 822, 829. See also Password Synchronizer pattern
vendor products for, 897–898
Web tier patterns, 617
PasswordSyncLedger class, 845
notification messages from, 887
sample code, 853, 881–885
PasswordSyncListener class, 845
sample code, 853, 869–887
screen display messages from, 886–887
PasswordSyncManager class, 844, 851, 853–855
PasswordSyncRequest class, 844, 853, 860–868
Patches
problems from, 6
in Secure UP, 451
Patterns, security. See Security patterns
PBEWithMD5AndDES algorithm, 169–170
PC/SC framework, 964
### Index

- **PKCS providers**, 124
- **PDPs (policy decision points)**
  - SAMM, 369
  - sample programs, 425–429
  - XACML, 413, 419–420
  - XACML 2.0 with SAMM 2.0, 432
- **Penetration tests**, 750
- **PEPs (Policy Enforcement Points)**
  - SAMM, 369–371, 373
  - XACML, 413
  - XACML 2.0 with SAMM 2.0, 430
- **Performance**
  - helper classes for, 639
  - J2EE network topology, 274
  - in security patterns
    - Audit Interceptor, 630, 634
    - Business tier, 696
    - Intercepting Validator, 569
    - Message Interceptor Gateway, 715
    - Obfuscated Transfer Object, 666
    - Policy Delegate, 672
    - Secure Logger, 587, 589
    - Secure Pipe, 595, 598
- **Permission class**, 99–101, 552
- **PermissionCollection class**, 550, 553
- **Permissions**
  - J2EE, 553
  - Java 2, 99–101
  - JNLP, 117
  - MIDlets, 132–134
  - tag library for, 554–556
  - Web tier patterns, 616
- **PermissionsCollection class**, 552–553
- **Persistent mode**, 892
- **Personal Data Ordinance**, 26
- **Personal Health Information (PHI)**, 24–25
- **Personal identification**, 29–30, 959–960
  - authentication, 29–30
  - best practices, 993–995
  - biometric. See Biometric identification and authentication
  - enabling technologies. See Enabling technologies for personal identification
  - physical and logical access control, 960–962
  - pitfalls, 997
  - references, 999
  - RFID-based, 35–36
  - smart cards. See Smart cards
- **PGP (Pretty Good Privacy)**
  - in PKI, 326
  - in trust models, 79–81
- **PHI (Personal Health Information)**, 24–25
- **Phishing**, 29
- **Physical access control**, 960–962, 977
- **PINs for smart cards**, 31, 995
- **Pipes**
  - Secure Pipe pattern. See Secure Pipe pattern
  - Web tier patterns, 617–618
  - Pipes and Filters pattern, 634
- **Pitfalls**
  - in case study, 956
  - in personal identification, 997
  - in security patterns
    - Business tier, 696
    - Identity management, 813–814
    - security provisioning, 891–894
    - Web services, 751
- **PKCS1 algorithm**, 304
- **PKCS#11 interface standard**, 175, 964
- **PKCS#15 interface standard**, 964
- **PKI (Public Key Infrastructure)**
  - limitations, 78
  - in Security Wheel, 443
  - Web tier patterns, 618
  - XML, 325
- **PKITS (Public Key Interoperability Test Suite)**, 179
- **PKIX**, 326
- **PKIXParameters class**, 180
- **Platforms in case study**, 933
- **Plug-ins**
  - for biometrics, 984
  - in Java System Access Manager, 402
  - for smart cards, 972
- **Pluggable Authentication Module (PAM)**, 197, 966–967
  - for biometrics, 985
  - for smart cards, 973
- **Point-to-Point Channel pattern**, 598
- **Point-to-point interfaces**, 822
- **Pointers in Java**, 96
- **POJO business objects**, 650, 652
  - policies, 6
  - in case study, 930–931
  - failures, 17
  - J2EE domains for, 228
  - JAAS authorization, 211–212
  - Java 2, 103–104
  - management, 403–404
    - DMTF, 404–405
    - EPAL, 405–408
    - IETF Policy Management Working Group, 404
    - Parlay Group, 405
  - services for, 528–529
Index

PRE (Password-Based Encryption), 169–170
Pre-process audit handling, 629–630
Prerequisites in case study, 915
Presentation tier
J2EE, 227, 239–240
reality checks for, 512
Pretexting Provisions, 23
Pretty Good Privacy (PGP)
in PKI, 326
in trust models, 79–81
Preventive transformations, 145
Primitives in Java, 96
Principal-based policy files, 210–211
Principal class, 199
Principals
Authorization Enforcer pattern, 552
delegation of, 263–264
J2EE, 228
JAAS authorization, 210–211
JAAS Login Module Strategy, 544
Liberty specifications, 388
propagation of, 695
resource, 270
-printcert option, 123
Printing certificate information, 123
Priorities, 6
Privacy
Secure Pipe pattern, 597–598
security provisioning patterns, 894
Security Services, 527
XACML, 416
Privacy-rule administrators, 405
Private keys, 325
Private/public key pairs, 119–121
PrivateCredentialsPermission class, 200
PrivateKey interface, 157
PrivilegedAction, 556
Proactive assessment, 20
Proactive security, 21, 39–41
Probability risk factors, 920
Problem in security pattern templates, 466
Assertion Builder, 756–757
Audit Interceptor, 624
Authentication Enforcer, 535–536
Authorization Enforcer, 548–549
Container Managed Security, 635
Credential Tokenizer, 802
Dynamic Service Management, 645
Intercepting Validator, 560–561
Intercepting Web Agent, 607
Message Inspector, 715–716
Message Interceptor Gateway, 705

in Web services, 296, 408–411, 421–422, 749–750
reality checks for, 501–507
in security patterns, 496–497
Business tier, 693, 695–696
Identity management, 813
Intercepting Web Agent, 610–611
Secure Service Facade, 682
Web tier, 619
in Security Wheel, 442
XACML, 413, 420–421, 425–429
Policy Administration Points (PAPs), 369, 373
Policy class, 199
Policy Decision Point Authority, 371
Policy decision points (PDPs)
SAML, 369
sample programs, 425–429
XACML, 413, 419–420
XACML 2.0 with SAML 2.0, 432
Policy Delegate pattern, 474, 482
consequences, 672–673
forces, 668–669
participants and responsibilities, 670–671
problem, 668
reality check, 677
related patterns, 677
sample code, 673–676
security factors and risks, 676–677
solution, 669
strategies, 672
structure, 670
Policy Enforcement Points (PEPs)
SAML, 369–371, 373
XACML, 413
XACML 2.0 with SAML 2.0, 430
Policy repository
SAML, 369
XACML, 420
Policy sets, 413
Policy stores, 421
<PolicySetIdReference> element, 418
Policytool tool, 104, 113, 125
Portals
SSO through, 85
in use cases, 911–912, 914
in user account provisioning, 829–830
Possibility risk factor
in case study, 920
in risk analysis, 453
Post-issuance applet downloads, 995
Post-process audit handling, 629–630
Post-synchronization event strategy, 849
Problem in security pattern templates, continued
- Obfuscated Transfer Object, 659
- Password Synchronizer, 840–841
- Policy Delegate, 668
- Secure Base Action, 570
- Secure Logger, 577–578
- Secure Message Router, 732–733
- Secure Pipe, 590–591
- Secure Service Facade, 677–678
- Secure Service Proxy, 599
- Secure Session Object, 686
- Single Sign-on Delegator, 776

Process method, 605
processPasswordSyncRequests method, 854–855

Profiles, 20
- in case study, 938
- J2ME, 130–132
- SAML, 367–368, 385–386, 416
- XACML, 416–417

Programmatic security
- authentication, 546–547
- authorization
  - Authorization Enforcer pattern, 553–556
  - J2EE, 232, 236, 253–255, 261–262
  - Container Managed Security pattern, 639
  - EJB method using, 643
  - Password Synchronizer pattern, 850–851
  - validation logic, 567–568

Proprietary solutions, 6

Protected resources, 637

Protection domains
- J2EE, 234–235
- Java 2, 98–99

ProtectionDomain class, 98–99

Protocol Binding strategy
- Assertion Builder pattern, 762
- Credential Tokenizer patterns, 806
- Password Synchronizer pattern, 850

Protocols
- Business tier patterns, 695
- Java System Access Manager, 402
- SAML, 366
- Security Services, 527

Protocols stack, 700–701

Provider classes
- JCA, 153
- JCE, 162

Providers
- authorization, 550, 552–553
- J2EE, 228–229
- JMS, 271

JSSE, 183–184
- Liberty specifications, 388, 391
- LoginModule, 204–206
- PCKS, 124
- Secure Message Router pattern, 743–744
- session state maintenance, 398
- Web services, 286, 350

Provisioning Service Points, 833, 845–847

Provisioning Service Targets, 833

Provisioning services. See User account provisioning

Proxies
- in Liberty specifications, 389
- Secure Service Proxy pattern. See Secure Service Proxy pattern

Proxy pattern, 606, 615
- Proxy tunneling, 192–193
- PSTID-ID mapping tables, 842–844
- Public Accounting Board, 26
- Public credential set, 553
- Public Key Infrastructure (PKI)
  - limitations, 78
  - in Security Wheel, 443
  - Web tier patterns, 618
  - XML, 325
  - Public Key Interoperability Test Suite (PKITS), 179
- Public keys
  - in assessment checklists, 500
  - LDAP, 74, 78
- PublicKey interface, 157
- publishPasswordSyncResult method, 880–881

Q

Qualitative risk analysis, 453

Quality of services
- reality checks, 510–511
- security patterns, 462–465
- factor analysis, 492
- security provisioning, 892–893

Quantitative risk analysis, 453–454

R

<r:license> element, 343–344

RA (risk analysis), 452–454

Radio Frequency Identification (RFID), 35–36

RAM
- for Secure Session Object pattern, 691
- in smart cards, 137–138

Random number generation, 77

Rationale in security design, 440–443

RBAC profiles, 416

RC6 algorithm, 58
Reactive security, 21
Readers
  RFID, 35
  smart card, 30–31, 971
Reality checks, 20, 501
  for administration, 508–510
  Business tier, 519–520
  Client Device tier, 511
  Integration tier, 520–523
  for policies, 501–507
  Presentation tier, 512
  for quality of services, 510–511
in security pattern templates, 467
  Assertion Builder, 775
  Audit Interceptor, 634
  Authentication Enforcer, 548
  Authorization Enforcer, 550
  Container Managed Security, 644
  Credential Tokenizer, 812
  Dynamic Service Management, 658–659
  Intercepting Validator, 569
  Intercepting Web Agent, 615
  Message Inspector, 731–732
  Message Interceptor Gateway, 715
  Obfuscated Transfer Object, 667–668
  Password Synchronizer, 890–891
  Policy Delegate, 677
  Secure Base Action, 576–577
  Secure Logger, 589–590
  Secure Message Router, 744
  Secure Pipe, 598
  Secure Service Facade, 685–686
  Secure Service Proxy, 606
  Secure Session Object, 693
  Single Sign-on Delegator, 801
Web tier, 513–519
Realms
  J2EE, 228, 239
  JAAS, 968
  for smart cards, 972
Reconciliation in user account provisioning, 829
Recovery, 21, 530
  in case study, 917
  key, 333
  service, 672
  in use cases, 909
  in XKMS, 337
Redirection, web, 393–394
Redundancy in Policy Delegate pattern, 677
Refactoring security design, 529
<Reference> element, 301
Reference templates for biometrics, 986
registerObject method, 653–654
Registration
  identity, 397–398
  UDDI, 747
Registries
  Dynamic Service Management pattern, 648
  UDDI, 289
  Web services, 286
  XACML, 422
RegistryMonitor class, 648
Regulatory policies, 496
Reissue service, key, 333
REL (Rights Expression Language), 340
Related patterns in security pattern templates, 467,
  478–479
  Assertion Builder, 775
  Audit Interceptor, 634–635
  Authentication Enforcer, 548
  Authorization Enforcer, 560
  Container Managed Security, 644–645
  Credential Tokenizer, 812–813
  Dynamic Service Management, 659
  Intercepting Validator, 569
  Intercepting Web Agent, 615
  Message Inspector, 732
  Message Interceptor Gateway, 715
  Obfuscated Transfer Object, 668
  Password Synchronizer, 891
  Policy Delegate, 677
  Secure Base Action, 577
  Secure Logger, 590
  Secure Message Router, 746
  Secure Pipe, 598
  Secure Service Facade, 686
  Secure Service Proxy, 606–607
  Secure Session Object, 693
  Single Sign-on Delegator, 801–802
Relationships in Liberty Alliance, 392–393
Reliability
  Assertion Builder pattern, 775
  Secure Message Router pattern, 743
reloadMBeans method, 654
Remote interface and services
  Secure Service Facade pattern, 683–684
  Single Sign-on Delegator pattern, 787
removeAssertionReply method, 774
removeAssertionStatement method, 799
removeCompRef method, 797
removeSessionInfo method, 797
removeSSOTokenMap method, 794
Index

Replay attacks
  Intercepting Web Agent pattern, 611
  SAML, 385
  Web services, 292
  XKMS, 337

Reporting practices
  Gramm-Leach-Bliley Act, 23–24
  Sarbanes-Oxley Act, 22–23

Reporting services in identity management, 28–29

Repository
  for biometric information, 997
  SAML, 369
  XACML, 420
  &lt;Request&gt; element, 418

Request messages
  Message Inspector pattern, 719, 721
  Secure Message Router pattern, 738

Request-reply model
  SAML, 373–376
  attribute assertion, 378–380
  authentication assertions, 376–378
  SPML, 835–836

RequestContext class, 536
  Authentication Enforcer pattern, 537–538
  Authorization Enforcer pattern, 550
  JAAS Login Module Strategy, 541

Requesters for Web services, 286

Requesting Authority, 833

RequestMessage class, 709

Requests, XACML, 423–425

Required flag, 205–206

Requirements
  Secure UP, 445, 449–450
  security basics, 50–53
  in use cases, 906–909

Requisite flag, 205–206

Resource principals, 270

Resource profiles, 416–417

&lt;resources&gt; element, 117

Resources tier, 227

respond method, 605

Response APDUs, 138

Response Message
  Message Inspector pattern, 721
  Message Interceptor Gateway pattern, 711
  &lt;Result&gt; element, 418

Retinal analysis, 32

Reusability
  Java components, 649
  password, 814
  in security patterns
  Authorization Enforcer, 559

Message Inspector, 731
  Message Interceptor Gateway, 714
  Secure Base Action, 574
  Secure Message Router, 743

Reverse engineering Java code, 143–144

Revocation issues and services, 68
  LDAP, 78
  Single Sign-on Delegator pattern, 786
  X-KRSS, 333

Revoked smart cards, 997

RFID (Radio Frequency Identification), 35–36

Rich-client authentication, 695

Rights Expression Language (REL), 340

Rijndael algorithm, 58

RIPEMD-160 encryption algorithms, 319

Risk analysis (RA), 452–454

Risk Analyzer, 828

Risks
  in case study, 917, 920–922
  in patterns-driven security design, 489–490
  in security patterns
    Assertion Builder, 774–775
    Audit Interceptor, 633–634
    Authentication Enforcer, 547–548
    Authorization Enforcer, 559
    Container Managed Security, 643–644
    Credential Tokenizer, 810–812
    Dynamic Service Management, 658
    Intercepting Validator, 569
    Intercepting Web Agent, 614
    Message Inspector, 731
    Message Interceptor Gateway, 714–715
    Obfuscated Transfer Object, 667
    Password Synchronizer, 888–890
    Policy Delegate, 676–677
    Secure Base Action, 576
    Secure Logger, 589
    Secure Message Router, 743–744
    Secure Pipe, 597–598
    Secure Service Facade, 685
    Secure Service Proxy, 606
    Secure Session Object, 692
    Single Sign-on Delegator, 800–801
  in security provisioning patterns, 894
  in trust model, 495
  in use cases, 909
  in Web services, 290–293

RMI/IIOP-based clients, 266

RMI socket factories
  client, 597
  server, 595–597

ROI study, 39–41
Roles

Business tier, 460, 694
Container Managed Security pattern, 644
identity management, 360
J2EE, 239, 553
ROM in smart cards, 137–138
Root certificates, 66–67
Root nodes in attack trees, 82–83
Rotate ciphers, 57
Routers. See Secure Message Router pattern
RPC style web services, 289
RSA BSAFE Secure-WS toolkit, 352–353
RSA cipher, 61–62
RSA-SHA1 algorithm, 304
Rule-based service integration and invocation, 683
<rule> element, 406
Rule Engine, 828
<RuleCombinerparameters> element, 417
Rules

EPAL, 405
in policy design, 496–497
XACML, 413
Run-as identity, 264–265
run method, 212–213

S

SAAJ API

in case study, 913
in Message Inspector pattern, 724
for Web services, 277
Safeguards Rule, 23
SAML (Security Assertion Markup Language), 85, 340, 362
for access control, 403
architecture, 368–369, 371–373
assertions, 363, 369, 376–382, 387
attribute, 378–380
authentication, 376–378, 387
authorization, 380–382
domain model, 370–371
Identity management patterns, 813–814
J2EE-based applications and web services, 386–387
Java System Access Manager with, 401–402
migration in, 775
motivation, 362
Policy Administration Point, 373
Policy Enforcement Point, 373
profiles, 365–366, 385–386, 416
request-reply model, 373–376
SAML 1.0, 364, 775
SAML 1.1, 364
SAML 2.0, 364–367, 430–432, 775
SSO in, 363, 813–814
usage scenarios, 383
DOS attacks, 384–385
global logout, 383–384
man-in-the-middle attacks, 385–386
message replay and message modification, 385
third-party authentication and authorization, 383
in XACML, 416, 422, 430–432
XML signatures in, 382
<saml:Assertion> element, 343–344
SAML Token profile, 368
Sample code for security patterns
Assertion Builder, 764–774
Audit Interceptor, 630–633
Authentication Enforcer, 545–547
Container Managed Security, 640–643
Credential Tokenizer pattern, 807–811
Dynamic Service Management, 652–658
Intercepting Web Agent, 612–614
Obfuscated Transfer Object, 666–667
Password Synchronizer, 851–887
Policy Delegate, 673–676
Secure Base Action, 575–576
Secure Logger, 587–589
Secure Pipe, 595–597
Secure Service Facade, 683–685
Secure Service Proxy, 603–605
Secure Session Object, 692
Single Sign-on Delegator, 787–800
SampleAuthorizationEnforcer.java file, 558
Sarbanes-Oxley Act (SOX), 22–23
identity protection in, 357, 359
in security provisioning patterns, 893
SASL (Simple Authentication and Security Layer)
API, 216–217
clients, 217–218, 220
installing, 219–220
servers, 218–219
SATAN (Security Administrator Tool for Analyzing Networks) tool, 953
SBU (Sensitive But Unclassified) information, 498
Scalability
J2EE network topology, 274
horizontal, 274–275
vertical, 276–277
Index

Scalability, continued
  in security patterns
    Intercepting Web Agent, 612
    Secure Message Router, 744
    Secure Service Proxy, 606
    security provisioning, 892
    Single Sign-on Delegator, 782
  sCallerInRole method, 262
Scanners
  in biometrics, 997
  fingerprint, 981–982
  Scanning data protection. See Intercepting Validator pattern
Scope
  Liberty Alliance, 388
  service provisioning, 818–819
Screen display messages, 886–887
Scripting, cross-site, 11
seal method, 667
Sealed objects, 168–169
SealedObject class, 162
Search operations in SPML, 834–835
SearchRequest message, 834–835
Secret data, 498
Secret keys in assessment checklists, 500
SecretKeyFactory class, 162
Secure Association patterns, 457
Secure Base Action pattern, 469, 480
  in case study, 918, 924, 927, 929
  consequences, 574–575
  forces, 570
  participants and responsibilities, 571, 573
  and Policy Delegate pattern, 677
  problem, 570
  reality checks, 576–577
  related patterns, 577
  sample code, 575–576
  security factors and risk, 576
  solution, 570–571
  strategies, 573–574
  structure, 571
Secure Communication patterns, 457
Secure data logger strategy, 580–584
Secure log store strategy, 584–585
Secure Logger pattern, 469, 480
  in case study, 924, 926, 929
  consequences, 584, 587
  forces, 578
  participants and responsibilities, 579–580
  problem, 577–578
  reality check, 589–590
  related patterns, 590
    Message Inspector, 732
    Secure Base Action, 573
    sample code, 587–589
    security factors and risks, 589
    solution, 578
    strategies, 580–586
    structure, 578–579
Secure Message Interceptor pattern, 927
Secure Message Router pattern, 476
  in case study, 918, 924, 929–930, 937
  consequences, 742–743
  forces, 733–734
  participants and responsibilities, 734–736
  problem, 730–731
  reality check, 744
  related patterns, 744
    Message Interceptor Gateway, 715
    Secure Service Proxy, 606
    security factors and risks, 743–744
    solution, 734–735
    strategies, 738–742
    structure, 735–736
Secure Pipe pattern, 470, 480, 486
  in case study, 924, 926, 929, 936
  consequences, 595
  forces, 591
  participants and responsibilities, 592
  problem, 590–591
  reality check, 598
  related patterns, 598
    Authentication Enforcer, 548
    Credential Tokenizer, 813
    Dynamic Service Management, 659
    Secure Logger, 590
    sample code, 595–597
    in secure log store strategy, 584, 586
    security factors and risks, 597–598
    solution, 591
    strategies, 593–595
    structure, 591–592
Secure Service Facade pattern, 474, 482, 484
  in case study, 924
  consequences, 682–683
  forces, 678
  participants and responsibilities, 679, 681
  problem, 677–678
  reality check, 685–686
  related patterns, 686
  sample code, 683–685
  security factors and risks, 685
solution, 678
strategies, 681–682
structure, 678–679
Secure Service Proxy pattern, 471
consequences, 605–606
forces, 599
participants and responsibilities, 600–601
problem, 599
reality check, 606
related patterns, 606–607
   Container Managed Security, 645
   Intercepting Web Agent, 615
   Secure Service Facade, 686
sample code, 603–605
security factors and risks, 606
solution, 599–600
strategies, 602–603
structure, 600
Secure service proxy single service strategy, 603
Secure Session Facade pattern, 918, 927
Secure Session Manager, 472, 480
Secure Session Object pattern, 475
consequences, 691–692
forces, 687
participants and responsibilities, 688–689
problem, 686
reality check, 693
related patterns, 693
sample code, 692
security factors and risks, 692
solution, 687
strategies, 689–691
structure, 687–688
Secure Session pattern, 812
Secure Socket Layer. See SSL (Secure Socket Layer)
Secure UP, 444–449
   artifacts in, 449–452
   risk analysis, 452–454
   trade-off analysis, 455
SecureBaseAction class
   with Apache Struts, 566–567
   Authentication Enforcer pattern, 537
   Authorization Enforcer pattern, 550, 552
   Intercepting Validator pattern, 562
   JAAS Login Module strategy, 541
SecureClassLoader class, 108
SecureID, 52
SecureRandom class, 154
SecureServiceFacade class, 670
SecureSessionFactorySessionBean.java file, 684–685
Security Administrator Tool for Analyzing Networks (SATAN) tool, 953
Security Assertion Markup Language. See SAML
   (Security Assertion Markup Language)
Security by default, 3–5, 442
   application security, 8–10
   business challenges, 5–7
   compliance. See Compliance
   flaws and exploits, 10–17
   four W's, 17–19
   identity management, 27–29
   Java technology, 36–37
   justifications, 37–42
   personal identification, 29–36
   proactive and reactive, 21
   references, 43–47
   strategies, 19–21
   summary, 42–43
   weakest links, 7–8
Security class, 154
Security Context pattern, 461
<security> element, 117–118
Security Event Logging pattern, 462
Security factors in security patterns, 467
   Assertion Builder, 774–775
   AuditInterceptor, 633–634
   Authentication Enforcer, 547–548
   Authorization Enforcer, 559
   Container Managed Security, 643–644
   Credential Tokenizer, 811–812
   Dynamic Service Management, 658
   Intercepting Validator, 569
   Intercepting Web Agent, 614
   Message Inspector, 731
   Message Interceptor Gateway, 714–715
   Obfuscated Transfer Object, 667
   Password Synchronizer, 888–890
   Policy Delegate, 676–677
   Secure Base Action, 576
   Secure Logger, 589
   Secure Message Router, 743–744
   Secure Pipe, 597–598
   Secure Service Facade, 685
   Secure Service Proxy, 606
   Secure Session Object, 692
   Single Sign-on Delegator, 800–801
Security levels in J2EE network topology, 273
Security patterns, 466
   application security assessment model, 499–500
   applying, 478–488
   Business tier, 460–462, 623–624
Index

Security patterns, continued
Audit Interceptor, 624–635
best practices, 693–696
Container Managed Security, 635–645
Dynamic Service Management, 645–659
factor analysis, 491
Obfuscated Transfer Object, 659–668
overview, 473–475, 480–485
pitfalls, 696
Policy Delegate, 668–677
references, 697
Secure Service Facade, 677–686
Secure Session Object, 686–693
in case study, 924–927, 939
classification, 497–498
existing, 456
factor analysis, 490–493
Identity management, 466, 477, 486–488
Assertion Builder, 756–775
best practices, 813–814
Credential Tokenizer pattern, 802–813
pitfalls, 813–814
references, 814–815
Single Sign-on Delegator pattern, 776–802
infrastructure and quality of services, 462–465
Integration tier, 462–463
labeling in, 498
in patterns-driven security design, 488–490
policy design in, 496–497
references, 531–532
relationships, 478–479
security provisioning
best practices and pitfalls, 891–894
Password Synchronizer, 890–891
threat profiling, 494
tier analysis, 493–494
trust model, 495
Web services, 475–476, 484–487
best practices, 745–751
Message Inspector, 715–732
Message Interceptor Gateway, 705–715
pitfalls, 751
references, 752
Secure Message Router, 732–744
Web tier, 456–459, 467–472, 478–481
Authentication Enforcer, 535–548
Authorization Enforcer, 548–560
best practices, 615–620
Intercepting Validator, 560–569
Intercepting Web Agent, 607–615
references, 620–621
Secure Base Action, 570–577
Secure Logger, 577–590
Secure Pipe, 590–598
Secure Service Proxy, 599–607
Security principles, references for, 531
Security Provider patterns, 459
Security provisioning
references, 895–898
security patterns
best practices and pitfalls, 891–894
Password Synchronizer, 840–891
summary, 894–895
Security realms
J2EE, 228, 239
JAAS, 968
for smart cards, 974
Security requirements and goals, 50
authentication, 51–52
authorization, 52–53
confidentiality, 50–51
integrity, 51
non-repudiation, 53
Security Services, 527–528
Security tokens. See Tokens
Security Wheel, 441–442
hub, 442
spokes, 442–443
wheel edge, 443
SecurityException class, 111
SecurityManager class, 104–105
SecurityProtocolHandler class, 601
SecurityToken class, 803, 805
Self-healing in Web services patterns, 750–751
Sensitive information
in case study, 922
Secure Logger pattern, 582
Secure Session Object pattern, 691
Web tier patterns, 618
Sensitive But Unclassified (SBU) information, 498
Separation of responsibility, 559
Sequence diagrams
identity provider agent strategy, 729
JAAS Login Module strategy, 541–543
in security patterns, 465, 480–486
Assertion Builder, 760–761
Audit Interceptor, 626
Authentication Enforcer, 537–538
Authorization Enforcer, 551–552
Container Managed Security, 636–637
Credential Tokenizer, 805–806
Dynamic Service Management, 646–647
Intercepting Validator, 562, 564
Intercepting Web Agent, 608–609
Message Inspector, 719–720
Message Interceptor Gateway, 709–710
Obfuscated Transfer Object, 660–661
Password Synchronizer, 845–846
Policy Delegate, 670–671
Secure Base Action, 571–572
Secure Logger, 578–579
Secure Message Router, 736–737
Secure Pipe, 592–593
Secure Service Facade, 679–680
Secure Service Proxy, 600–601
Secure Session Object, 688–689
Single Sign-on Delegator, 780–781
Sequence numbers for deletion detection, 582, 584
Server Gated Cryptography (SGC), 619
Server mutual authentication, 233
Server-side communication, 188–190
Server-side SSL example, 495
Server-to-server connections
  in case study, 916
  in use cases, 908
Web tier patterns, 618
ServerHello messages, 70
Servers
  for biometrics, 985
  DMZ, 617
  Password Synchronizer pattern, 888
  in provisioning, 828, 893–894
  SASL, 218–219
  Secure Pipe pattern, 598
  sizing, 893–894
  for smart cards, 971–972
  in use cases, 908
Service-level agreements (SLAs), 452, 907
Service Locator pattern
  and Secure Service Facade pattern, 681
  and Single Sign-on Delegator pattern related, 802
Service-Oriented Architecture (SOA), 277, 284, 599
Service provider interfaces (SPIs)
  in Credential Tokenizer patterns, 806
  in PAMs, 966
Service providers
  in Liberty specifications, 388, 391
  Single Sign-on Delegator pattern, 782, 784
  for Web services, 286
Service provisioning, 817–818
  business challenges, 818–823
  identity management relationship, 819–820
  scope, 818–819
  security patterns for, 466, 477
  in Security Services, 528
  user account. See User account provisioning
Service Provisioning Markup Language (SPML),
  819, 823, 833–834
  features, 835–837
  implementation, 837–840
  operations, 834–835
Service registry, 286
Service requesters, 286, 480, 486
ServiceConfig class, 844, 855–860
ServiceConfigContext class, 844
ServiceLocator service, 780
ServiceManager class, 646, 648
Services
  aggregation of, 685
  in case study, 940–941
  catalog, 943–946
  order fulfillment, 948–951
  order management, 946–948
  user login, 941–943
  continuity and recovery, 530
  strategies, 21
  in use cases, 909
  in Web services, 295–296
  directory, 28, 704–705
  penetration tests, 750
  as weakest links, 8
  Web. See Web services tier
Servlets, 913
Session Authority entity, 369
Session pattern, 459
Session Facade pattern
  and Secure Service Facade pattern, 686
  and Secure Session Object pattern, 693
Session facade strategy, 628–629
Sessions
  MIDlet, 133
  states
    Liberty Alliance, 398
    SSL, 69
    theft, 13–14
    Single Sign-on Delegator pattern, 786
    Web services, 292
timeouts in, 620
  tracking
    cookies and URL rewriting in, 248–251
    Web tier patterns, 620
  weak identifiers, 12
setActionList method, 405
setAssertionType method, 764, 771
setAuthenticationMethod method, 764, 771
setComponentsConfig method, 857–858
setConfigProperties method, 798
setConfRef method, 799
setData method, 667
setLoginContext method, 576
setMaxInactiveInterval method, 251
setMessageDrivenContext method, 631
setProtocolBinding method
AssertionContextImpl, 774
PasswordSyncRequest, 868–869
SSOContextImpl, 798
setRegistryFileName method, 657
setSecureTransferObject method, 692
setSecurityManager method, 104
setServiceName method, 799
setSessionInfo method, 780, 796
setSSOTokenMap method, 794
setStatus method, 800
setTokenType method, 809
Setup IDS, 617
setupDefaultUserProfile method, 854
SGC (Server Gated Cryptography), 619
SHA-1 cryptography, 54–56
in Cryptographic Service Providers, 153
for JCA message digests, 155–156
SHA1 encryption, 319
SHA256 encryption, 319
SHA512 encryption, 319
-showcert option, 136
sign method
Signature, 158
Signer, 587–588
Sign-ons
EIS tier, 268–270
multiple, 16, 786
single. See Single sign-on (SSO) mechanisms
Signature class
JCA, 152
Secure Logger pattern, 582
<Signature> element, 300–301
<SignatureMethod> element, 301
<SignatureProperties> element, 303
Signatures, 62–64
Assertion Builder pattern, 775
for JAD files, 135–136
in Java System Access Manager, 402
JCA, 158–159
verification, 33, 311
Web services patterns, 748
WS-Security, 339, 345–346
XML. See XML (Extensible Markup Language)
<Signaturevalue> element, 301
Signed applets, 112–115
Signed MIDlets, 134–136
Signed security tokens, 339
<SignedInfo> element, 301
Signer class, 587–588
Signing jar files
jarsigner for, 125
in Web tier patterns, 620
Simple Authentication and Security Layer (SASL), 216–217
clients, 217–218, 220
installing, 219–220
servers, 218–219
Simple Object Access Protocol. See SOAP (Simple Object Access Protocol) and SOAP messages
SimpleFormAction class, 567–568
Single Access Point patterns, 458
Single Logout Profile, 367
Single Loss Expectancy (SLE), 453
Single service secure service proxy strategy, 602–605
Single Sign-on Delegator pattern, 477, 488, 813–815
consequences, 788–789
forces, 777
participants and responsibilities, 780–782
problem, 776
reality check, 801
related patterns, 801–802
Assertion Builder, 775
Password Synchronizer, 891
sample code, 787–800
security factors and risks, 800–801
solution, 777–778
strategies, 780–784
structure, 778–780
Single sign-on (SSO) mechanisms, 16, 28, 84
Assertion Builder pattern, 763
biometrics, 990–993
in case study, 917
Credential Tokenizer patterns, 806–807
cross-domain, 85–86, 400
federated, 86–88
identity management, 361
J2EE authentication, 246–247
JAAS authorization, 213
JGSS, 215
Liberty Alliance, 391, 398, 400, 740–742
Password Synchronizer pattern, 890
Index

through portals, 85
SAML in, 361
in use cases, 908–909
user account provisioning, 822, 831
Web services, 296
SLAs (service-level agreements), 452, 907
SLE (Single Loss Expectancy), 453
Smart cards, 30–32, 136–137, 961–963
architecture and implementation model, 968–969
best practices, 995
components, 137–138
Java Card technology, 138–139
as Java key stores, 176–177
in Java security, 124–125
in JCE, 176–178
logical architecture, 969–973
in multi-factor authentication, 992–994
operational model, 973–977
for physical access control, 977
snoop method, 876–877
SOA (Service-Oriented Architecture), 277, 284, 599
SOAP (Simple Object Access Protocol) and SOAP messages, 286–288
SAML, 385–387
in security patterns, 277–278, 484
Message Inspector, 717, 722, 724
Password Synchronizer, 889–890
Secure Message Router, 740
Secure Service Proxy, 602
SPML, 834
WS-Policy, 408–409
Socket factories, 595–597
SocketFactory class, 185
Solution in security patterns, 466
Assertion Builder, 757–758
Audit Interceptor, 625
Authentication Enforcer, 536
Authorization Enforcer, 549
Container Managed Security, 636
Credential Tokenizer, 802–803
Dynamic Service Management, 646
Intercepting Validator, 561
Intercepting Web Agent, 607–608
Message Inspector, 717–719
Message Interceptor Gateway, 707–709
Obfuscated Transfer Object, 660
Password Synchronizer, 842–843
Policy Delegate, 669
Secure Base Action, 570–571
Secure Logger, 578
Secure Message Router, 734–735
Secure Pipe, 591
Secure Service Facade, 678
Secure Service Proxy, 599–600
Secure Session Object, 687
Single Sign-on Delegator, 777–778
SOP (Standard Operating Procedure) documents, 451
Source code scanners, 524
SourceBaseAction class, 538
SourceSite class, 760, 762
SOX (Sarbanes-Oxley Act), 22–23
identity protection in, 357, 359
in security provisioning patterns, 893
SPFs (service provider interfaces)
in Credential Tokenizer patterns, 806
in PAMs, 962
SPKI, 326
SPML (Service Provisioning Markup Language), 819, 823, 833–834
features, 835–837
implementation, 837–840
operations, 834–835
Spokes in Security Wheel, 442–443
Spoofing
and client-side validations, 561
in Web services, 292
SQL
embedded commands, 560
injection vulnerability, 11
SQLValidator, 564
SSL (Secure Socket Layer), 67
accelerators, 594, 618
in case study, 923
issues, 72–73
J2EE, 243–244, 258, 278
JSSE, 182–185
HTTP over SSL, 191–193
for secure socket connections, 186–190
for RMI socket factories, 595–597
role of, 69–71
vs. TLS, 71–72
Web-server-based, 594
Web services, 276, 618
WS-Security, 338
XML encryption, 312
SSL_NULL_WITH_NULL_NULL, 73
SSLContext class, 194–197
SSLSession class, 185
SSO. See Single sign-on (SSO) mechanisms
SSOContext class, 778, 784
SSOImpl class, 795–800
SSOServerDelegate class, 780, 782, 785–786
SSOServerDelegateException class, 776
Index

SSODelegatorFactory class, 778, 780, 788–795
SSOServiceProvider class, 778, 780
Standard Operating Procedure (SOP) documents, 451
Standards
   Authentication Enforcer pattern, 547
   smart cards, 995
   Web services, 286, 297–303, 745
start method
   PasswordSyncLedger, 884–885
   PasswordSyncRequest, 864–885
State maintenance in Liberty Alliance sessions, 398
Stateful firewalls, 616
Stateful transactions, 273
Stateless/stateful Policy Delegate, 672
Stateless transactions, 273
Static conformance requirements, 389
Static mappings, 640
<Status> element, 418
Stolen smart cards, 995
Storage, insecure, 12
-storepass option, 120
-storepasswd option, 124
Strategies in security patterns, 19–21, 466
   Assertion Builder, 762–763
   Audit Interceptor, 627–629
   Authentication Enforcer, 538–544
   Authorization Enforcer, 552–559
   Container Managed Security, 638–639
   Credential Tokenizer, 803–805
   Dynamic Service Management, 646
   Intercepting Validator, 562–563
   Intercepting Web Agent, 608
   Message Inspector, 719
   Message Interceptor Gateway, 709
   Obfuscated Transfer Object, 660
   Password Synchronizer, 843–845
   Policy Delegate, 670
   Secure Base Action, 571
   Secure Logger, 578–579
   Secure Message Router, 735–736
   Secure Pipe, 591–592
   Secure Service Facade, 678–679
   Secure Service Proxy, 600
   Secure Session Object, 687–688
   Single Sign-on Delegator, 778–780
Subject class
   Authentication Enforcer pattern, 537
   Authorization Enforcer pattern, 550
   JAAS authorization, 193, 556
Subject Descriptor pattern, 460
Subjects in JAAS
   authorization, 210, 212
   Login Module strategy, 544
   Sufficient flag, 205–206
   Summaries of security factors, 493
   SunJCE provider, 161
   SunJSSE provider, 183–184
   SunPKS11 provider, 124
   Super encryption, 324–325
   Support strategy in security provisioning patterns, 893
Symmetric ciphers, 56–58
Symmetric keys
   Obfuscated Transfer Object pattern, 664
   Secure Logger pattern, 582
   XML, 318
Synchronization
   identity management, 27
   passwords
      Password Synchronizer pattern. See Password Synchronizer pattern
      user account provisioning, 822, 829
System constraints, 909
System Entry Point, 370
System environment in use cases, 913

T
Tags
libraries for, 554–556
RFID, 35
TakeAction class, 878–881
Tamper-proofing transformations, 145
<Target> element, 413
Targets
in case study, 931–932
in security patterns
Audit Interceptor, 627
Intercepting Validator, 564
Obfuscated Transfer Object, 661–662, 665
Password Synchronizer, 888
Secure Session Object, 688
SPML, 833
XACML, 413–414
Technology differentiators, 832–833
Technology elements in case study, 913, 915
Templates
biometrics, 986
Java System Access Manager, 402
security pattern. See Security patterns
Termination
biometrics, 985–987
identity, 397–398
smart card, 973–974
Testability
Message Interceptor Gateway pattern, 714
Secure Message Router pattern, 743
Testing, 523
black box, 524
in case study, 951–953
Web services patterns, 750–751
white box, 524–525
Theft
identity, 9
session, 13–14
in Web services, 292
Third-party authentication and authorization, 383
Third-Party Communication pattern, 463
Threat modeling, 81–83
Threat profiling
in case study, 938
for security patterns, 494
Threats to Web services, 290–293
Three-factor authentication, 992
Tier matrices, 494
Tiers
in case study, 935
in J2EE, 226–228
in risk analysis, 453
in security patterns, 493–494
Time checking strategy, 762
Timeouts
HTTP sessions, 250–251
URLConnections, 191–192
Web tier patterns, 620
Timestamps
Web services patterns, 748
WS-Security, 348–349
TLS. See Transport Layer Security (TLS)
TOA (trade-off analysis)
in case study, 923
in Secure UP, 455
TokenContext class, 803, 805
TokenContextImpl class, 805
Tokens
biometrics, 991–992
Credential Tokenizer. See Credential Tokenizer pattern
HTTP-POST, 363
Identity management patterns, 814
policy, 410–411
RSA cryptographic, 964
smart card, 178
SSO, 85, 991–992
unified credential token solution, 961
weak, 12–13
WS-Policy, 410–411
WS-Security, 338–340, 343, 345
Tolerance level of potential security threats, 530
Top Secret data, 498
Traceability in Web services, 294–295
Trade-off analysis (TOA)
in case study, 923
in Secure UP, 455
Trading partners in use cases, 912
Training, 6
Transactions
in case study, 916
J2EE network topology, 273
Liberty Alliance, 391
Secure Pipe pattern. See Secure Pipe pattern
Secure Service Facade pattern, 683
in use cases, 908
transfer method, 643
Index

Transfer object member strategy, 689–690, 692
Transfer Object pattern
   and Obfuscated Transfer Object pattern, 659, 668
   and Secure Session Object pattern, 693
Transform algorithms, 305
Transformation, code, 144–145
<Transforms> element, 302
Transparency
   Assertion Builder pattern, 764
   Credential Tokenizer patterns, 807
Transport Layer Security (TLS), 71–72
   issues in, 72–73
   J2EE, 237, 258, 278
   JMS, 272
   JSSE, 182–185
   Web services, 701–702
   WS-Security, 339
   XML encryption, 312
TRIPLEDES encryption algorithm, 317–318
Trust models
   in case study, 937
   LDAP, 79–81
   for security patterns, 495
TrustAnchor class, 180
Trusted certificates
   for applets, 115
   importing, 122–123
Trusted MIDlets, 132–134
TrustManager class, 185
TrustManagerFactory class, 185
Trusts in WS-Security, 339
TrustStore property, 191
TSIK services, 352
Tunneling, proxy, 192–193
Twofish algorithm, 58
Types, Java, 96

U
UDDI (Universal Description, Discovery, and Integration)
   attacks on, 293
   and Secure Logger pattern, 590
   for Web services, 289, 747
UIDGenerator class, 582
Unclassified data, 498
Unified credential tokens, 961, 995
Unified Process (UP), 19
   references for, 531
   secure. See Secure UP
Unit testing, 951–952
Universal Description, Discovery, and Integration (UDDI)
   attacks on, 293
   and Secure Logger pattern, 590
   for Web services, 289, 747
Unlimited strength cryptography, 178
unloadMBeans method, 654–655
Unprotected resources, 262–263
unregisterObject method, 654
unwrap method, 194–196
UP (Unified Process), 19
   references for, 531
   secure. See Secure UP
update method, 156, 158
Upgrades, 6
URL rewriting, 248–251
URLClassLoader class, 108–109
URLConnection timeouts, 191–192
Use cases, 905
   actors, 911–912
   diagrams, 910–911
   methodology choices, 905–906
   online portals, 911–912, 914
   requirements identification, 906–909
   in Secure UP, 450
   system constraints, 909
   system environment, 913
User account provisioning, 27–28, 361
   approaches to, 822–823
   architecture
      centralized model vs. decentralized, 823–826
      components of, 828
      logical, 826–828
      differentiators for, 832–833
      identity management, 360
      identity provider infrastructure integration, 830–832
      portal integration, 829–830
      scenario, 820–821
      services, 828–829
User agents, 391
User login
   biometrics, 996
   in case study, 941–943
   use case, 911
Usernames
   JAAS authorization, 214
   WS-Security, 343
UsernameToken class, 805, 811–812
Users in J2EE, 238
UserStore class, 537
UUID pattern and profile
  SAML, 368
  Secure Logger pattern, 584
V
  validate method, 566–567
  Validate service, 329–330
  validateSecurityContext method, 604–605
  validateSecurityToken method, 789
Validation
  in case study, 953
  certificate chains, 181–182
  failures, 10
  Liberty Phase 1, 389
  in security patterns
    Business tier, 695
    Intercepting Validator. See Intercepting Validator pattern
    Policy Delegate, 677
    Secure Base Action, 576
    Secure Service Facade, 682
    Web services, 747
    Web tier, 619
    X-KISS, 329–330
    XML signatures, 311
  <VariableDefinition> element, 418
  <VariableReference> element, 417–418
Variables in Java, 96
Vendor-independent security, 529
Vendor-neutral credential handler, 807
Vendor-specific security
  session management, 691–692
  Web services APIs, 751
Vendors
  password management, 897–898
  service provisioning, 896–897
Verification, 6
  biometric, 978–981
  certificate chains, 65
  host name, 193–194
  jar files, 126
  signatures, 33, 311
  Web tier patterns, 619
verify method, 158
VeriSign CA, 78, 352
<Version> element, 417
Version numbers in XACML, 418
Vertical scalability in J2EE network topology, 276–277
viewResult method, 885
virtual machines
  CVM, 128
  JVM, 94–95
  KVM, 129–130
VLANs, 932
Voice verification, 32
VPN access, 746
Vulnerabilities, Web services, 290–293
W
  Watermarking Java code, 145
  Weakest links, 7–8
Web
  application load-balancing, 923
  authentication, 547–548
  redirection, 393–394
  servers
    in SSL, 594
    in use cases, 908
  Web tier patterns, 616–617
  services. See Web services tier
  validation, 565
  Web-based transactions. See Secure Pipe pattern
  Web tier patterns. See Web tier
Web browser SSO Profile, 367
Web of trust models, 79
Web Services Definition Language (WSDL), 277, 288, 293, 747
Web Services Interoperability Organization (WS-I), 349–350
Web Services Policy Framework (WS-Policy), 408–411
Web services policy language (WSPL), 408–411
Web services tier, 283–284, 527, 699–700
  architecture and building blocks, 284–285
in case study, 915, 930, 933, 937
  communication styles, 289–290
  core issues, 290–293
  infrastructure, 702–705
  in J2EE, 257–259
  Java-based providers, 350–353
  in Liberty Alliance, 391, 393
  message-layer security, 702
  network-layer security, 701
  operational model, 285–286
  policies, 408–411, 421–422, 749–750
  protocols stack, 700–701
  references, 355
  requirements, 293–296
Index

Web services tier, continued
SAML in, 386–387
security patterns, 466, 474–476, 484–487
best practices, 745–751
factor analysis, 491
Message Inspector, 715–732
Message Interceptor Gateway, 705–715
pitfalls, 751
references, 752
Secure Message Router, 732–744
standards, 286–290, 745
WS-Security. See WS-Security
XML. See XML (Extensible Markup Language)
summary, 354
Transport-Layer Security, 701–702
WS-I security profile, 349–350
Web tier
in case study, 929, 932–933, 936
container managed security strategy, 638
in J2EE, 227, 239–240
authentication, 240–248
authorization, 251–255
default propagation from, 265–266
HTTP session tracking, 248–251
reality checks for, 513–519
security patterns, 456–459, 467–472, 478–481
Authentication Enforcer, 535–548
Authorization Enforcer, 548–560
best practices, 615–620
factor analysis, 491
Intercepting Validator, 560–569
Intercepting Web Agent, 607–615
references, 620–621
Secure Base Action, 570–577
Secure Logger, 577–590
Secure Pipe, 590–598
Secure Service Proxy, 599–607
Web.xml file
basic HTTP authentication entry, 545
certificates based authentication entry, 546
deployment descriptor, 640–641
form based authentication entry, 545–546
WebAgent class
in case study, 933
Single Sign-on Delegator pattern, 782, 784
Wheel edge in Security Wheel, 443
Where in security, 18
Which in security, 18–19
White box testing, 524–525
in case study, 952
Secure UP, 446, 449, 451
Who in security, 18–19
Why in security, 18–19
Wireless Toolkit (WTK), 131, 135
Wireless Transport Layer Security (WTLS), 258
Workflow Engine, 828
WorkflowRecipient class, 738
wrap method, 194, 196
WriteAppletPolicy.policy file, 112
WriteFileApplet.html file, 111–112
WriteFileApplet.java file, 110–111, 113
WS-I (Web Services Interoperability Organization), 349–350
WS-I Security profiles, 296
WS-Policy (Web Services Policy Framework), 408–411
WS-Security, 337
definitions, 338–339
encryption, 339, 346–348
in JWSDP, 351
motivation, 338
namespaces, 342
SAML and REL in, 340
signatures, 339
SOAP messages, 340–349
tokens, 339–340
WSDL (Web Services Definition Language), 277, 288, 293, 747
WSPL (Web services policy language), 408–411
<wsse:BinarySecurityToken> element, 343
<wsse:Security> element, 342
<wsse:SecurityTokenReference> element, 345
<wsse:UsernameToken> element, 343
<wsu:TimeStamp> element, 348–349
WTK (Wireless Toolkit), 131, 135
WTLS (Wireless Transport Layer Security), 258
X
X.500/LDAP Profile, 368
X.509 certificates. See Certificates and certificate keys
X-BULK, 334–336
X-KISS (XML key information services), 326–327
locate service, 327–329
validate service, 329–330
X-KRSS (XML key registration service), 330
recovery, 333
registration, 331–333
reissue, 333
revocation, 333
X509v3CertToken, 805
XACML (Extensible Access Control Markup Language), 411–414
access control and policy management, 403–404
architecture, 420

data flow, 418–419

vs. EPAL, 405–408

usage scenarios, 421

ebXML registry, 422

policy centralization, 421–422

policy stores, 421

SAML collaboration, 422

sample policy, 425–429

sample requests, 423–425

sample scenario, 423

XACML 2.0 with SAML 2.0, 430–432

XACML Kit, 422–423

WSPL from, 409

XACML 2.0, 416–418

XML schema definition, 414–416

XACML kit, 416, 422–422

XACML Profile, 368

XCBF (XML Common Biometric Format), 33–34

<xenc:EncryptedData> element, 346–347

<xenc:EncryptedKey> element, 347–348

XKMS (XML key management system), 325, 748–749

issues, 336–337

motivation, 325–326

specification overview, 326

Verisign services, 352

X-BULK, 334–336

X-KISS, 326–327

locate service, 327–329

validate service, 329–330

X-KRSS. See X-KRSS (XML key registration service)

XLST transform algorithms, 305

XML (Extensible Markup Language), 286

encryption, 311–312

algorithms, 316–320

anatomy, 312–316

arbitrary content, 324

element level, 321–324

example scenarios, 320–325

motivation, 312

super encryption, 324–325

firewalls, 353

for performance, 746

for Web Services, 704

in J2EE, 279

in Message Inspector pattern, 716–717

in Message Interceptor Gateway pattern, 705

signatures, 297–298

algorithms, 303–305

anatomy, 298–303

Assertion Builder pattern, 775

creating, 310–311

examples, 305–310

motivation, 298

in SAML, 382

verifying and validating, 311

XACML for, 412

XML-aware security, 353

infrastructure, 745–746

Message Inspector pattern, 721–723

Message Interceptor Gateway pattern, 711–712

XML Common Biometric Format (XCBF), 33–34

XML Denial of Service (XML-DOS) attacks, 291

XML Digital Signature algorithm, 319

XML key information services (X-KISS), 326–327

locate service, 327–329

validate service, 329–330

XML key management system. See XKMS (XML key management system)

XML key registration service (X-KRSS), 330

recovery, 333

registration, 331–333

reissue, 333

revocation, 333

XML messaging provider strategy, 738–740

XML Schemas, 293, 748

in Message Inspector pattern, 722

in Message Interceptor Gateway pattern, 707–708

for XACML, 414–416

Xpath transform algorithms, 305

XrML (Extensible Rights Markup Language), 340

XSS (cross-site scripting), 11

Xverify command, 108

Y

Yarrow random number generator, 77

Z

Zero knowledge testing, 524

Zimmerman, Phil, 79