

SOA SECURITY

Ramarao Kanneganti
Prasad Chodavarapu

 MANNING



contents

preface xvii
acknowledgments xix
about this book xxi

PART I SOA BASICS 1

- 1** ***SOA requires new approaches to security*** 3
- 1.1 SOA lowers long-standing barriers 5
 - Basic tenets of SOA* 6 ▪ *Idea of a service* 7
 - 1.2 Lowering of barriers forces us to rethink security 10
 - 1.3 Functional aspects of security: With and without SOA 13
 - Authentication* 14 ▪ *Authorization* 16 ▪ *Data confidentiality* 17 ▪ *Data integrity and nonrepudiation* 19
 - Protection against attacks* 20 ▪ *Privacy protection* 21
 - 1.4 Nonfunctional aspects of security 22
 - Interoperability* 22 ▪ *Manageability* 23
 - Ease of development* 24
 - 1.5 New security approaches for SOA 25
 - Message-level security* 25 ▪ *Security as a service* 26
 - Policy-driven security* 28

- 1.6 Current SOA security implementation choices 28
- 1.7 Summary 30
 - Suggestions for further reading 31

2

Getting started with web services 33

- 2.1 Setting up tools and environment 34
 - Choosing a platform and a toolkit 35* ▪ *Getting started with Apache Axis 36*
- 2.2 XML basics 39
 - XML data format 41* ▪ *XML namespaces 43*
 - XML schema 45* ▪ *Processing XML 49* ▪ *XPath 51*
- 2.3 SOAP basics 55
 - SOAP message exchange model 55* ▪ *Anatomy of a SOAP message 56* ▪ *RPC with SOAP 57*
 - Document exchange with SOAP 60* ▪ *SOAP Fault 61*
- 2.4 WSDL basics 64
 - Describing a service with WSDL 65* ▪ *Understanding ports and port types 65* ▪ *Understanding bindings 66*
- 2.5 Web services in action with Apache Axis 68
 - Creating a web service 68* ▪ *Consuming a web service 71* ▪ *Using a web service from .NET 75*
- 2.6 Choices in service design 77
 - Wrap existing interfaces or design from scratch? 77* ▪ *To use SOAP or not? 78* ▪ *Start with WSDL or generate it? 79*
 - Should security context be part of the interface? 79* ▪ *RPC or document exchange? 80*
- 2.7 Related technologies: UDDI 80
- 2.8 Summary 81
 - Suggestions for further reading 82

3

Extending SOAP for security 84

- 3.1 Finding the right approach for security in SOAP 86
 - Lessons from web authentication schemes 86* ▪ *Authentication at the HTTP layer 87* ▪ *Choices for security implementation in SOAP 89*

- 3.2 Extending SOAP with headers 92
 - Anatomy of a SOAP header* 93
 - *Standard header entry attributes* 94
- 3.3 WS-Security: The standard extension for security 97
 - Introduction to WS-Security* 97
 - *Example: Identifying a brokerage service user* 100
- 3.4 Processing SOAP extensions using handlers 103
 - How handlers work* 103
 - *Outline of the solution* 105
 - *Implementing a server-side JAX-RPC handler* 106
 - *Implementing a client-side JAX-RPC handler* 110
 - *Handler chains* 112
 - *Configuring handlers and handler chains* 114
- 3.5 Processing SOAP extensions using intermediaries 118
 - Preserving the endpoint information: WS-Addressing* 119
 - SOAP processing rules for intermediaries* 121
- 3.6 SOAP Extensions FAQ 124
 - What should go into the headers?* 124
 - *How do we standardize on headers?* 125
 - *How many handlers?* 125
 - *How do we support handlers?* 126
- 3.7 Summary 126
 - Suggestions for further reading* 127

PART II BUILDING BLOCKS OF SOA SECURITY 129

4 **Claiming and verifying identity with passwords** 131

- 4.1 Authentication with username and password 133
 - Example: Username and password in WS-Security* 133
 - Implementing username/password scheme: client-side* 137
 - JAAS: A generic framework for authentication* 138
 - Implementing username/password scheme: server-side validation* 148
- 4.2 Using password digest for authentication 151
 - How password digest authentication works* 152
 - *Password digest authentication in action* 153
 - *Implementing password digests: client-side* 156
 - *Implementing password digests: server-side validation* 161

- 4.3 Is password authentication the right solution for you? 168
 - Why is the digest scheme secure?* 168
 - *Problems with digest authentication* 169
 - *Limitations of password-based schemes* 170
- 4.4 Summary 171
 - Suggestions for further reading 172

5

Secure authentication with Kerberos 173

- 5.1 Authentication requirements in SOA 175
- 5.2 Introduction to Kerberos 177
 - Basic ideas behind Kerberos* 178
 - *Authentication sequence* 184
 - *Beyond client authentication* 186
 - Roadmap for the rest of the chapter* 187
- 5.3 Implementing Kerberos with JAAS and GSS APIs 189
 - Client-side implementation* 189
 - *Service-side implementation* 194
- 5.4 Using Kerberos with WS-Security 196
 - Running the Kerberos example* 196
 - *Adding a Kerberos ticket to a WS-Security header* 199
 - *Using a Kerberos ticket for authentication* 200
 - *Adding a Kerberos ticket on the client-side* 201
 - *Processing a Kerberos ticket on the service-side* 202
- 5.5 What authentication scheme to use? 205
- 5.6 Summary 207
 - Suggestions for further reading 207

6

Protecting confidentiality of messages using encryption 209

- 6.1 Encryption in action: an example 211
- 6.2 The basics of encryption 214
 - Types of encryption algorithms* 214
 - *PKI: A framework for encryption* 222
- 6.3 Programming with digital certificates 228
 - Creating digital certificates* 228
 - *Point to point encryption with digital certificates (SSL/TLS)* 231
 - *Java APIs for encryption* 235

- 6.4 Encrypting SOAP messages 237
 - Example: Sending user credentials with selective encryption* 238
 - *Encrypting-side implementation* 244
 - *Decrypting-side implementation* 253
- 6.5 Practical issues with encryption 256
- 6.6 Summary 258
 - Suggestions for further reading 259

7 **Using digital signatures** 260

- 7.1 The basics of XML signatures 264
 - Challenges in signing XML* 264
 - *XML canonicalization* 266
- 7.2 Signing SOAP messages 275
 - Example: Signing order creation request* 276
 - Sender-side implementation* 284
 - *Receiver-side implementation* 294
- 7.3 Practical issues with signatures 302
 - Three rules of signatures* 302
 - *Mixing encryption and signatures* 303
 - *Which canonicalization scheme?* 303
- 7.4 Summary 304
 - Suggestions for further reading 305

PART III ENTERPRISE SOA SECURITY 307

8 **Implementing security as a service** 309

- 8.1 Security as a service 311
 - Is a security service technically feasible?* 315
 - *Standards for implementing security as a service* 316
- 8.2 Analyzing possible uses of a security service 316
 - Use case 1: Destination endpoint invokes security service out-of-band* 317
 - *Use case 2: Source endpoint invokes security service out-of-band* 319
 - *Use case 3: Both endpoints invoke security service out-of-band* 320
 - *Use case 4: Security service as an explicit intermediary* 322
 - *Use case 5: Security service as an implicit intermediary* 323

- 8.3 Conveying the findings of a security service: SAML 325
 - SAML assertion basics* 326
 - *AuthenticationStatement: Asserting authentication results* 327
 - *AttributeStatement: Asserting user attributes* 328
 - *AuthorizationDecisionStatement: Asserting authorization decisions* 329
- 8.4 Example implementation using OpenSAML 331
 - Client-side implementation* 332
 - *Security service implementation* 334
 - *Server-side implementation* 341
- 8.5 Standards for security service interfaces 343
 - WS-Trust* 344
 - *SAML protocol* 352
- 8.6 Summary 354
 - Suggestions for further reading 355

9

Codifying security policies 356

- 9.1 Introducing declarative security 358
 - Policy consolidation for planning and consistent enforcement* 359
 - Use at design time to ensure interoperability* 361
 - *Use at runtime to ensure interoperability* 363
- 9.2 Interoperability challenges in SOA security 365
 - Sources of incompatibility* 365
 - *WS-I basic security profile* 368
- 9.3 Web services policy framework 369
 - What is a policy?* 370
 - *WS-Policy* 372
 - *Standards for fetching policy: WS-MetadataExchange and WS-PolicyAttachment* 374
- 9.4 WS-SecurityPolicy 379
 - Security assertions for endpoints* 381
 - *Security assertions for messages* 392
 - *Security assertions for operations* 393
 - *Limitations of WS-SecurityPolicy* 394
- 9.5 Summary 394
 - Suggestions for further reading 395

10	<i>Designing SOA security for a real-world enterprise</i>	397
10.1	Meeting the demands of enterprise IT environments 399	
	<i>Large and diverse user base</i> 400 ▪ <i>Long life cycle</i> 402	
	<i>Robustness</i> 402 ▪ <i>Manageability</i> 403 ▪ <i>Integration with diverse legacy applications</i> 404	
10.2	Securing diverse services 404	
	<i>Services developed from scratch</i> 405 ▪ <i>Services wrapping legacy applications</i> 406 ▪ <i>Services composed of other services</i> 413	
10.3	Choosing a deployment architecture 414	
	<i>For securing services in the intranet</i> 417 ▪ <i>For securing services offered to the public</i> 422 ▪ <i>For securing services offered to/by partners</i> 427	
10.4	Making the solution industrial-strength 429	
	<i>Performance</i> 429 ▪ <i>Scalability</i> 431 ▪ <i>Availability</i> 433	
10.5	Vulnerability management 433	
	<i>Common vulnerabilities</i> 434 ▪ <i>XML-specific vulnerabilities</i> 438 ▪ <i>Vulnerability remediation workflow</i> 440	
10.6	Summary 442	
	Suggestions for further reading 443	
	<i>appendix A: Limitations of Apache Axis</i> 445	
	<i>appendix B: WS-SecureConversation</i> 449	
	<i>appendix C: Attaching and securing binary data in SOAP</i> 453	
	<i>appendix D: Securing SAML assertions</i> 461	
	<i>appendix E: Application-Oriented Networking (AON)</i> 472	
	<i>index</i> 477	