About the Panelists

• Chris Steel, CISSP
  Chief Architect, Fortmoon Consulting
  csteel@fortmoonconsulting.com

• Ramesh Nagappan
  Technology Architect, Sun Microsystems
  Ramesh.Nagappan@sun.com

• Ray Lai
  Principal Engineer, Sun Microsystems
  Ray.Lai@sun.com
This session is based on the findings of a dedicated research and industry work experience on applying proactive approaches, patterns-driven and best practices solutions for implementing trustworthy applications, Web services and Identity Management.

- Chris Steel, Ramesh Nagappan, Ray Lai
authors@coresecuritypatterns.com
www.coresecuritypatterns.com
Objectives

• The convergence of existing applications and Web services based technologies introduces a myriad of newer security challenges.
  — Many of us realize them only as an afterthought!

• This session brings those afterthoughts to the fore by addressing:
  — Proactive and prescriptive guidance to Web services security
  — Patterns-driven Web services security development and deployment
  — Adopting best practices and reality checks

• This session provides a product-agnostic, vendor-neutral, platform/language independent view of implementing security.
Agenda for the Panel Session

• A quick look at XML Web Services
• Web services security issues
• Web services security requirements
• Web services – industry efforts and standards
• Security Wheel and Secure UP
• Applied Web services security patterns
• Best practices
Question

What are XML Web Services?

What are the associated security challenges?
A Quick look at XML Web services

• Based on XML standards, standards-based protocols and Service-oriented architecture (SOA) concepts,
  — XML, SOAP, WSDL, UDDI
  — HTTP, SMTP, etc

• Delivers a composable architecture using standards-based technologies
  — Standards-based components allows defining and describing services, discovering and subscribing services, communication, aggregation, collaboration, provisioning and so on.
  — Ensures interoperability at all levels
Web Services – Core Security issues

Web services is not an exception…..

Schema
Replay-attacks
Man-in-the-Middle
DOS/DDOS/XML DOS
Data Privacy
Message interception and manipulation
WSDL disclosure
Phishing and Spoofing
Fault
Validation issues
Compliance
Virus attachments
Auditing/Logging failure
Multiple Sign-on
Weak Security tokens

…and it is not threat-free or future-proof …
What are the core security requirements for XML Web services?
Web Services – Core Security Requirements

- Authentication
- Authorization
- Auditing and Traceability
- Message Integrity
- Message Confidentiality
- Non-repudiation
- Availability and Service continuity
- Identity and Policy
- Security Interoperability
What are the relevant industry standards for securing Web Services?
Web Services Security – Industry Standards / Specifications

- XML Digital Signature (XML DSIG)
- XML Encryption (XML-ENC)
- Web Services Security (WSS)
- XML Key Management (XKMS)
- Secure Assertions Markup Language (SAML)
- XML Access Control Markup Language (XACML)
- ...And a growing list...
From your research, what is your advice for ensuring end-to-end security for applications and web services?
What is the Weakest Link in IT Ecosystems?

• The Network
• The Operating System
• The Application
Think Proactive - Security Wheel
Think Predictive – Secure UP

- Based on Unified Process
- Prescribes a Security Discipline to ensure security from ground-up
- Layered defense strategy by applying “Security Wheel”
- Mitigates potential security risks and vulnerabilities at the early design phase
- Adopts a patterns-driven security architecture
- Enforces best practices and reality checks
- Delivers “Security By Default”

RSA Conference 2005
Actors
• Analyst
• Architect
• Designer
• Developer
• Tester
• Operations staff
• Auditor

Think Process – Secure UP Workflow

Copyright ©2005 Steel, Nagappan, Lai
Can you elaborate on the patterns catalog?
Describe a real-world scenario to illustrate those approaches and how to adopt a patterns-driven security architecture.
Let’s consider a common Web Services scenario involving:

- Hospitals
- Doctors
- Pharmacies
- Laboratories
- Insurance providers

What are the business problems related to security?

And what solution do you recommend?
Applied Web Services Security Patterns

• Message Interceptor Gateway (*Web services*)
• Message Inspector (*Web services*)
• Secure Message Router (*Web services*)
• Secure Logger (*Web-tier*)
• Audit Interceptor (*business-tier*)
• *Assertion Builder* (*Identity-tier*)
Business Problems Related to Security

Communication Scenarios
- Insurance Provider ⇔ Patient info ⇔ Health provider
- Physician ⇔ Patient info ⇔ Pharmacy
- Health provider ⇔ Patient info ⇔ Laboratories
- Health provider ⇔ Patient info ⇔ Insurance provider
- Pharmacy ⇔ Patient info ⇔ Insurance provider

Key Security Challenges
- Message integrity and confidentiality
- Message Verification and Validation
- Message security processing (Authentication, Authorization)
- Message logging and auditing
- Message/Element level security
- Message routing to multiple endpoints
- Message origin verification
- Message Compliance and interoperability
Message Interceptor Gateway

Problem

*You want to use a single entry point and centralize security enforcement for all incoming and outgoing messages.*

Forces

- Block and prevent all direct access to the exposed service endpoints.
- Intercept all XML traffic and inspect the complete XML message/attachments.
- Verify the message integrity and confidentiality for eavesdropping and tampering.
- Enforce transport-layer security using Two-way SSL/TLS (Mutual authentication)
- Protect the exposed WSDL descriptions from public access and revealing operations.
- Apply message inspection and filter mechanisms for content, payload size and message representation.
- Monitor and identify replay and DOS attacks by tracking and verifying the IPs, hostnames, message timestamps and other message sender information.
Message Interceptor Gateway

Solution

- Message Interceptor Gateway pattern is a proxy to the Web services infrastructure and provides a centralized entry point encapsulating access to all target service endpoints.
- It works as a security enforcement point.
- It secures the incoming and outgoing XML traffic by securing the communication channels.

Strategies

- XML Firewall Strategy
- Web Proxy Infrastructure Strategy
Problem

You want to verify and validate the quality of message-level security mechanisms applied to XML Web services.

Forces

• Examine message-level security for structure and content, verifying their uniqueness, confidentiality, integrity and validity.

• Identify the applied security-tokens and assertions representing the identity and policies.

• Monitor and identify XML based DOS and REPLAY attacks by tracking and verifying security tokens, signatures, message correlation, message expiry or timestamps.

• Verify messages for interoperability and standards compliance.

• Enforce a centralized logging based on the security actions and decisions

• Reusable API mechanisms for managing and processing the message-level security.
Solution

• Message Inspector is a modular or a pluggable component that integrates with infrastructure components.

• Executes a chain of tasks as a preprocessing or post processing intermediary for all incoming and outgoing messages.

• It works as a security decision point

Strategies

• Message Handler strategy

• Identity provider agent strategy
Secure Message Router

Problem

You want to securely communicate with multiple partner endpoints applying message-level security and identity federation mechanisms.

Forces

• A security intermediary for Web services based workflow or multiple service endpoints.
• Configure element-level security and access control
• Revealing only the required portions of a protected message to a target recipient.
• Enable SSO (Ex: generating SAML assertions and XACML based ACLs).
• Global logout notification.
• Notification of identity registration and termination
• Dynamically apply security criterion through message transformations, canonicalizations before forwarding it to its intended recipient.
Secure Message Router

Solution

• Secure Message Router is security intermediary infrastructure that aggregates access to multiple endpoints.

• Acts on the incoming messages and dynamically applies security logic for routing messages to multiple end-point destinations.

• Makes use of a security configuration utility to apply end-point specific security decisions and mechanisms.

Strategies

• Messaging Provider strategy

• Liberty SSO strategy
Problem

Certain requests must be securely logged for security auditing and debugging purposes. This can lead to redundant code and complex logic.

Forces

• You need to log sensitive information that should not be accessible to unauthorized users.

• You need to ensure the integrity of the data logged.

• You want to capture output at one level for normal operations, but at all levels prior to an exception.

• You want to centralize control of logging in the system.
Secure Logger

Solution

• Use a Secure Logger to log messages in a secure manner such that they can not be easily altered or deleted, and events can not be lost.

Strategies

• Secure Data strategy
• Secure Log strategy
Problem

You want to intercept and audit requests and responses to and from the business tier.

Forces

- You want centralized, declarative auditing of service requests.
- You want auditing of service decoupled from the services themselves.
- You want pre- and post-process audit handling of service requests, including auditing of exceptions.
**Solution**

- **Use an Audit Interceptor to centralize auditing functionality and define audit events declaratively, independent of the business tier services.**

**Strategies**

- Interception Session Façade strategy
Can you describe some common-sense best practices for Web services security?
Best Practices in Web Services Security

- Adopt a standards-based security infrastructure
- Protect your network perimeter
- Enforce end-to-end transport layer security
- Choose an XML-aware security infrastructure at the perimeter level
- Restrict all direct access to service endpoints, WSDL and XML Schemas
- Verify and validate all messages before processing
- Inspect messages for all content-level threats and vulnerabilities
- Use timestamps and correlation to determine the validity of a message
- Apply Message element-level encryption and signature mechanisms
- Adopt secure logging, monitoring and auditing mechanisms
- Adopt XML-aware appliances for performance
- Apply service penetration tests and hardening

..........................and more
For More Information

“From the ground up, this Java platform was designed for security. Read this book to learn how to apply patterns and proven technologies to secure your J2EE applications and beyond.”

—James Gosling, Father of the Java programming language

core
SECURITY
PATTERNS
Best Practices and Strategies for J2EE™, Web Services, and Identity Management

- Patterns catalog includes 23 new patterns for building end-to-end security
- Security design methodology, patterns, best practices, ready checks, proactive security assessments, defensive strategies and checklists
- Applied techniques for Web services security, Identity Management, and Service Provisioning
- Comprehensive security guide using J2SE™, J2EE™, JAAS™, and Java Card™

CHRIS STEEL • RAMESH NAGAPPAN • RAY LAI

www.coresecuritypatterns.com

RSA Conference 2005