Web Services Security Update

Hal Lockhart

BEA Systems

Architecture and Standards Group

bea
Think liquid.

Hal Lockhart

- Principal Technologist, BEA Systems
- Co-chair XACML TC, SAML TC
- Co-chair OASIS TAB
- Vice Chair WS-I Basic Security Profile
- Also Member:
  - Provisioning TC, Digital Signature Services TC, Web Services Secure Exchange (WS-SX), WS-I Reliable Secure Profile WG
- OASIS Coordinator for WSS Interop Demo

Topics

- OASIS WS-Security
  - History & Content
  - WSS 1.1
  - Issues
- WS-I Basic Security Profile
- OASIS WS-Secure Exchange (WS-SX)
  - WS-Trust
  - WS-Secure Conversation
  - WS-Security Policy
- Experience from the field

Web Services Security History

- Submitted to OASIS September 2002
- Interoperability testing began Summer 2003
- OASIS Standard in April 2004
  - Core Specification + Username and X.509 Profiles
- OASIS Standard December 2004
  - SAML and REL Token Profiles
- WSS 1.1 – OASIS Standard February 2006
  - Includes Attachments & Kerberos
- WS-I Basic Security Profile
  - Profiling OASIS specs as completed

Features in WSS - 1

- Security Header
  - Can contain mustUnderstand
  - Can be addressed to Role
- Tokens
  - Associated with signature or encryption or otherwise used to identify party to message exchange
  - Binary Token - encapsulates binary object
    - X.509 certificate – defined by ITU/IETF
    - Kerberos ticket – defined by IETF/Microsoft
  - XML Token – inserted as is
    - Username Token – defined by OASIS WSS TC
    - SAML Assertion – defined by OASIS SS TC
    - XrML License – defined by ContentGuard

Features in WSS - 2

- Security Token Reference
  - Points to or encapsulates a token
  - Four types
    - Direct – URI or URI fragment
    - Key Identifier – specific to token type – identifies key, certificate, ticket, assertion, etc.
    - Key Name – identifies token by content, e.g. SubjectName
    - Embedded – encapsulates token, allows association of additional information with token
- Signature element
  - New transform - STR Dereference Transform
- Encryption ReferenceList or EncryptedKey elements
- Timestamp element
  - Only applies to security mechanisms
  - Created and/or Expires

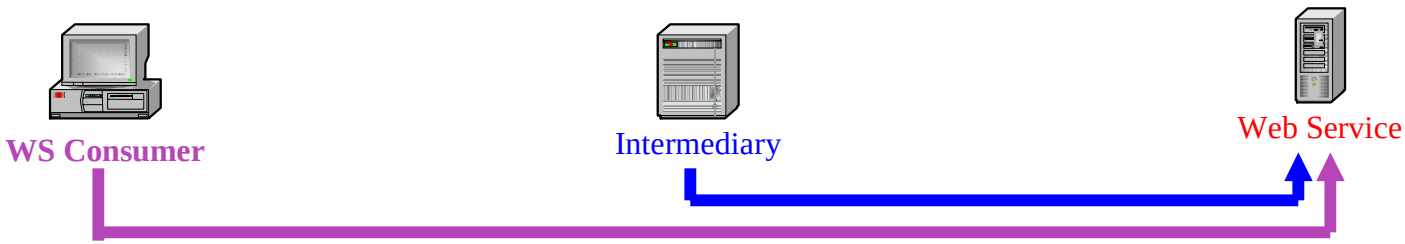
OASIS WSS 1.1

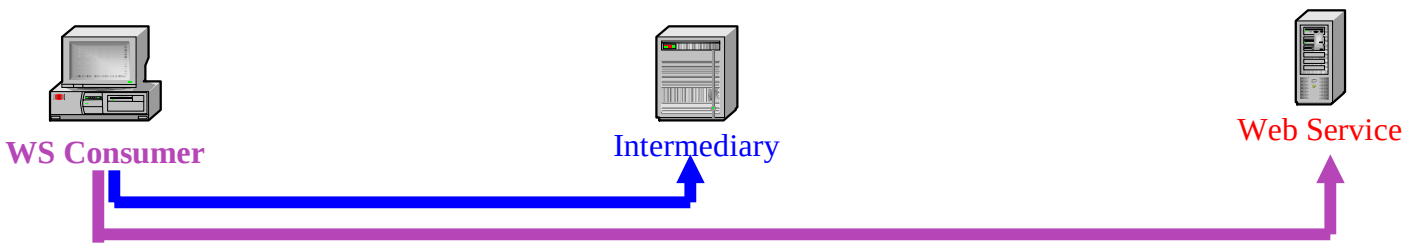
- Upwardly compatible with WSS 1.0
- New Features
  - Encrypted SOAP Header
  - Token Reference to Encrypted Key
  - Signature Confirmation
  - Password-based Key Derivation
  - Thumbprint References
  - Errata and Clarifications
- SOAP Attachment Security Profile
- Kerberos Token Profile

Web Services Security Issues

- Goal of specs is deliberately limited
  - Many options allowed
  - Impact on Interoperability and Security
- No description mechanism included
  - WS-Security Policy submitted to OASIS in 2005
- Use of intermediaries not well understood
  - Therefore neither are security implications
  - Necessary usage patterns my require private agreements
- No completely satisfactory C14N algorithm
  - Essential for digital signatures

SOAP Intermediaries





Topics

- OASIS WS-Security
  - History & Content
  - WSS 1.1
  - Issues
- WS-I Basic Security Profile
- OASIS WS-Secure Exchange (WS-SX)
  - WS-Trust
  - WS-Secure Conversation
  - WS-Security Policy
- Experience from the field

WS-I Basic Security Profile WG

- Chartered March 2003
- Deliverables:
  - Security Scenarios (and supporting Usecases if nec.)
  - Extension Profile of BP (the first ever)
    - HTTP over TLS
    - SOAP with attachments
    - WSS Core + Profiles
- Security Scenarios complete October 2004
  - Renamed “Security Challenges, Threats and Countermeasures”

Work on Profile Document

- General Process
  - Reviewed underlying documents
    - IETF RFCs covering TLS
    - XML Signature, XML Encryption
  - Reviewed WSS Documents – Committee Draft
  - Identified candidate profiling points
  - Defined constraint or dropped each
  - Restated important constraints to force testing

Related WS-I Activities

- Testing Tools
  - Driven by Profile
  - Test Assertion for each Profile Requirement
  - Special considerations for encryption
- Sample Applications
  - Value Chain Application
  - Mix and match pieces from vendors
  - Public demonstrations
  - Security added to current design
  - Test tools used to verify compliance

Scope - What BSP is Not

- BSP is *not* developing a cohesive set of security use cases
  - These are expressed through sample apps
- BSP is *not* developing a recipe for securing WS communications
  - BSP is about interoperability, not describing how to secure messages
- BSP does *not* dictate best practices
  - Although it does draw attention to some security considerations
- BSP does *not* profile dependent technologies outside of the constraints already imposed by a profiled specification
  - E.g. BSP does not constrain W3C XML Digital Signature outside of what is relevant for OASIS WSS
- BSP does *not* try to dictate security behavior outside of the message/transport level

Standards to be Profiled

BSP 1.0

- WSS 1.0 (Mar 2004) + Oct 2004 errata
- Username Token Profile 1.0 (Mar 2004) + Oct 2004 errata
- X.509 Token Profile 1.0 (Mar 2004) + Oct 2004 errata
- SAML Token Profile1.0 (Dec 1, 2004)
- REL Token Profile 1.0 (Dec 19, 2004)
- SOAP with Attachments 1.1 (OASIS Public Review Draft 28 June 2005)
- Kerberos Token Profile 1.1 (OASIS Public Review Draft 28 June 2005)

BSP 1.1

- WSS 1.1
- Username Token Profile 1.1
- X.509 Token Profile 1.1
- SAML Token Profile1.1
- REL Token Profile 1.1
- SOAP with Attachments 1.1
- Kerberos Token Profile 1.1

Principles

- No guarantee of interoperability
- Focus profiling effort
- Application semantics
- Testability
- Strength of requirements
- Restriction vs. relaxation
- Multiple mechanisms
- Future compatibility

- Compatibility with deployed services
- Focus on interoperability
- Conformance targets
- Lower-layer interoperability
- Do no harm
- Best Practices

Conformance Claims

- Each requirement statement contains exactly one requirement level keyword (e.g., "MUST") and one conformance target keyword (e.g., "MESSAGE").
  - R9999 *WIDGETs **SHOULD** be round in shape.*

- Targets defined in BSP

SECURE_ENVELOPE	**ENCRYPTION_REFERENCE_LIST**
SECURE_MESSAGE	**ENCRYPTED_KEY_REFERENCE_LIST**
SENDER	**ENCRYPTED_DATA**
RECEIVER	**SECURITY_TOKEN_REFERENCE**
INSTANCE	**INTERNAL_SECURITY_TOKEN**
SECURITY_HEADER	**EXTERNAL_SECURITY_TOKEN**
REFERENCE	**SECURITY_TOKEN**
SIGNATURE	**TIMESTAMP**
ENCRYPTED_KEY	**MIME_PART**

Conformance Claims *(cont.)*

- Conformance mechanism detailed in:
  - *WSI Conformance Claim Attachment Mechanisms Version 1.0*
- URI for conformance:
  - *http://ws-i.org/profiles/basic-security/core/1.0*
  - Exceptions:
    - Transport Layer Security
      - http://ws-i.org/profiles/basic-security/transport/1.0
    - Username Token Profile
      - http://ws-i.org/profiles/basic-security/username-token/1.0
    - X.509 Certificate Token Profile
      - http://ws-i.org/profiles/basic-security/x.509-certificate-token/1.0
    - Attachment Security
      - http://ws-i.org/profiles/basic-security/swa/1.0

Topics

- OASIS WS-Security
  - History & Content
  - WSS 1.1
  - Issues
- WS-I Basic Security Profile
- OASIS WS-Secure Exchange (WS-SX)
  - WS-Trust
  - WS-Secure Conversation
  - WS-Security Policy
- Experience from the field

WS-Secure Exchange TC (WS-SX)

- New TC formed December 2005
- Three Specs
  - WS-SecureConversation
  - WS-Trust
  - WS-SecurityPolicy
- Secure Conversation is relatively straightforward
- Some of WS-Trust required for SC, not clear how much more to implement
- Interops began summer of 2006
- WS-SC & WS-Trust currently in Public Review
- Work continues on WS-SecurityPolicy

WS-Trust

- Mechanisms to issue tokens and associated keys
- Essential to bridging Security Domains
  - Allow client to authenticate and obtain token for foreign domain
- Builds on WS-Security
- Security Token Service (STS)
  - Trusted Service – Brokers between Domains
- Request/Response Protocols
  - Issuance, Renewal, Validation, Cancellation, Challenges/Negotiations
- Multiple Key Agreement alternatives
- Like WSS it is a toolkit, not a solution

WS-Secure Conversation

- Builds on WS-Security and WS-Trust
- Allows establishment of secure session
- More efficient and secure than using long term secrets directly
- Like SSL/TLS except at SOAP layer
- Useful in conjunction with reliable messaging
- Adds two new Token types
  - Security Context Token (holds session info, including keys)
  - Derived Key Token (enables key derivation)
- Two party and three party flows
- Also a toolkit, but less so

WS-Security Policy

- Allows Web Service to express Security Policies
  - What needs to be protected
  - What tokens to use
  - Algorithms, reference types, etc.
- Builds on WS-Policy
  - Uses nested policy to provide scope
- Defines various groups of policy assertions
  - Correspond to features of WSS, Secure Conversation, Trust, etc.
- Expressed in WSDL per WS-PolicyAttachment
- Constrains content and layout of wsse:Securityheader
- Defines a number of Assertion types

WS-Security Policy Assertion Types

- Protection assertions
  - What parts of msgs need to be protected – Confidentiality, Integrity
- Token assertions
  - Types of tokens, inband or out of band
- Binding assertions
  - Transport, Symmetric, Asymmetric Bindings
  - Can apply to response as well as request
- Supporting Token assertions
  - Additional signatures, e.g. Endorsements
- Protocol assertions
  - Other properties, e.g. Algorithms, Timestamps, Reference types

Topics

- OASIS WS-Security
  - History & Content
  - WSS 1.1
  - Issues
- WS-I Basic Security Profile
- OASIS WS-Secure Exchange (WS-SX)
  - WS-Trust
  - WS-Secure Conversation
  - WS-Security Policy
- Experience from the field

Representative Applications

- Protecting communications between car rental company and insurance companies
- Privacy protection of consumer data in car loan applications
- Online government services pilot project
  - Authentication based on X.509
  - Requests passed to external access control policy engine
- Large telephone company requires signatures on all web service transactions

- Integration usually a factor
  - Legacy systems, multiple organizations, existing security technology
- Many requests for custom authentication tokens
  - Generally use existing, proprietary format

Challenges

- WS-Security provides may options – many orgs lack existing policies for message protection
- Performance of message processing
  - Signatures – Canonicalization
- Many moving parts
  - Hard to understand
  - Tricky to configure
- Impedance mismatch between programming styles
- Closed world vs. Open world
- Trying to find the right tradeoff between ease of development and flexibility
  - Model policies

Summary

- WSS 1.0 is widely implemented in products
- WSS 1.1 is complete and will be available soon
- Products are providing interoperability and effective security in simple, common situations today
- The WS-I BSP will provide useful guidance for Interoperability
- Expect WS-Secure Conversation in products within 6 months
- Other uses of WS-Trust are speculative
- WS-Security Policy will be widely implemented, but there is little operational experience with problem space
- Users will see benefits not available in SSL/TLS
- No way around some technology restrictions like C14N
- Complex scenarios will likely have security holes initially
- Parts of WS are not well enough understood to secure

Questions?

bea
Think liquid.