International Civil Aviation Organization

AERONAUTICAL TELECOMMUNICATION NETWORK IMPLEMENTATION COORDINATION GROUP – EIGHTH WORKING GROUP MEETING (ATNICG WG/8)

Christchurch New Zealand
28 September – 1 October 2010

Agenda Item 3: Planning the use of XML and SWIM related standards in AMHS environment

WEB SERVICE SECURITY STANDARDS

(Presented by USA)

SUMMARY

This Working Paper provides an introduction to the web services security standards.

1. Introduction

This paper provides a general introduction to web services security standards.

2. Discussion

2.1 Web Services Standards Overview

Figure 2-1 from the NIST Guide to Secure Web Services depicts a notional reference model that maps Web services security standards to different functional layers of a Web service implementation.

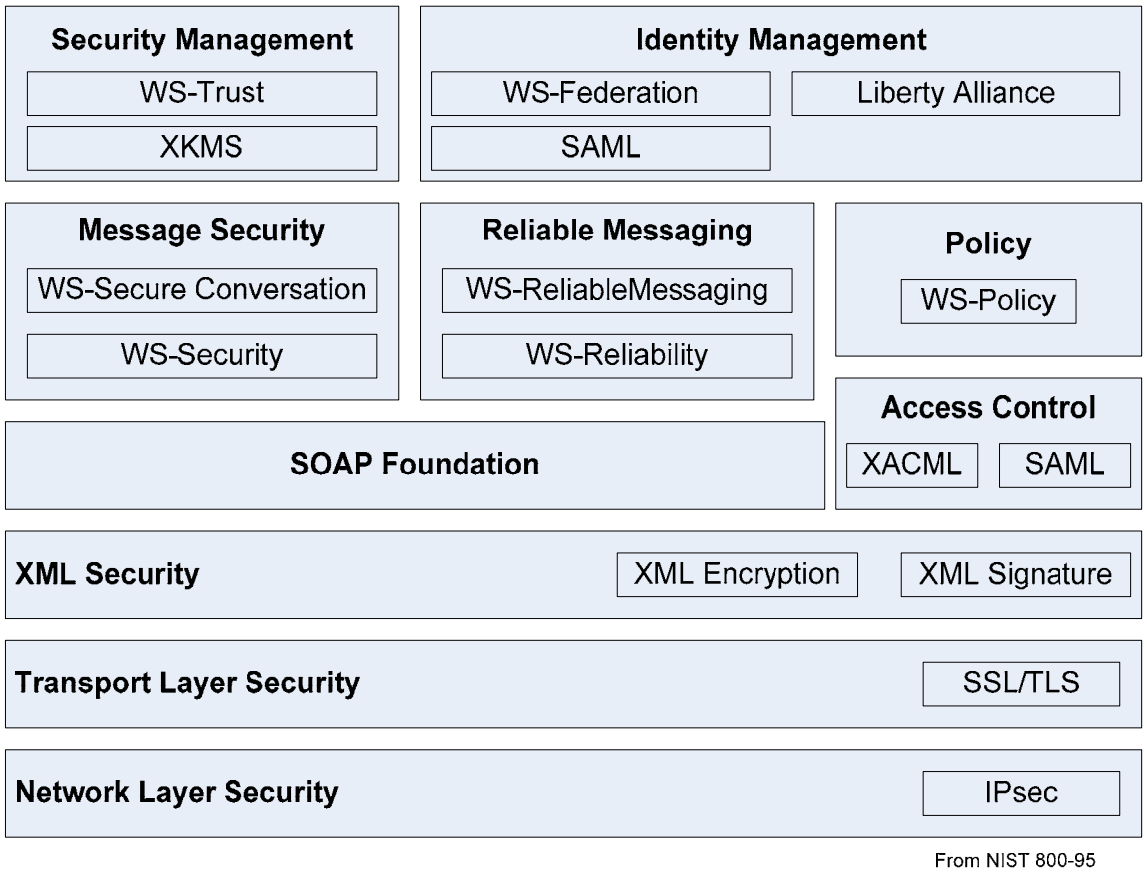

Figure 2-1: Web Services Security Standards

2.2 Network Layer Security

2.2.1 Internet Protocol Security (IPsec)

Internet Protocol Security (IPsec) is a set of protocols for securing IP communications. The Internet Key Exchange (IKE or IKEv2) is used to establish keys and set up a security association to be used by either the Authentication Header (AH) or the Encapsulating Security Payload (ESP) protocol. AH provides integrity and authentication at the IP layer. ESP provides integrity and authentication and optionally confidentiality at the IP layer.

2.3 Transport Layer Security

2.3.1 Secure Socket Layers/Transport Layer Security (SSL/TLS)

Transport Layer Security (TLS) [RFC 5246], which is derived from the Secure Sockets Layer (SSL) protocol developed by Netscape, is a cryptographic protocol which provides security at the transport layer. It is widely used in web browser to server communications to provide unilateral authentication, i.e., where only the server is authenticated. TLS also provides an option for mutual authentication where both communicating peers authenticate one another generally using Public Key Certificates.

TLS at a high level involves three phases. In the first phase, the communicating peers negotiate cipher suites which identify which cryptographic algorithms are to be used. In the second phase keys are exchanged and authentication is performed. In the third phase symmetric message authentication and optionally encryption is performed. TLS is used in Web Services in conjunction with HTTP, where it is denoted HTTPS, to provide point-to-point security between a client and service provider.

2.4 XML Security

2.4.1 XML Signature

The World Wide Web Consortium (W3C) XML Signature specification specifies an XML compliant syntax used for representing the signature of web resources and portions of XML documents and protocol messages. XML Signature also specifies procedures for computing and verifying signatures. An XML signature can be used to sign data outside of the XML document in which it appears. In this case it is called a detached signature. If it is used to sign a part of its containing document it is called an enveloped signature. If it contains the signed data within its own structure, it is called an enveloping signature.

The general XML signature syntax is depicted below.

```
<Signature ID>
  <SignedInfo>
    <SignatureMethod />
    <CanonicalizationMethod />
    <Reference URI >
       (<Transforms>)
       <DigestMethod>
       <DigestValue>
    </Reference>
    (<Reference />)
  </SignedInfo>
  <SignatureValue />
  (<KeyInfo>)
    <KeyName>
    <KeyValue>
    <RetrievalMethod>
    <X509Data>
    <PGPData>
    <SPKIData>
   </KeyInfo>
  <Object ID>
  </Object>
</Signature>
```

The <Signature> element is the root element of an XML signature. The <Signature> element has the child elements <SignedInfo>, <SignatureValue>, <KeyInfo>, and <Object ID>. <SignedInfo> contains references to signed data and specifies what algorithms are used. <SignatureValue> contains the signature generated encoded in base64 format. <KeyInfo> permits the signer to provide the verifier with the key to validate the signature. The key may be included in the structure in the form of a public key certificate or the RetrievalMethod element may be used to reference KeyInfo information outside of the structure. XML

Signature is used in both SOAP and REST implementations for integrity and non-repudiation.

2.4.2 XML Encryption

The W3C XML Encryption specification provides requirements for XML syntax and processing for encrypting digital content, including portions of XML documents and protocol messages.

The general XML encryption syntax is depicted below.

```
<EncryptedData ID Type MimeType Encoding>
    <EncryptionMethod />
    <ds:KeyInfo>
    <EncryptedKey>
    <AgreementMethod>
    <ds:KeyName>
    <ds:RetrievalMethod>
    <ds:*>
   </KeyInfo>
   <CipherData>
    <CipherValue>
    <CipherReference URI>
   </CipherData>
   <EncryptionProperties>
</EncryptedData>
```

The <EncryptedData> element is the root element generated with XML encryption is applied. The <EncryptedData> element has the child elements <EncryptionMethod>, <KeyInfo>, <CipherData> , and <EncryptionProperties>. <EncryptionMethod> specifies which algorithm was used to encrypt the data. XML data is encrypted using symmetric techniques, therefore, the recipient needs to be able to obtain the symmetric key. The key may be transported by enciphering and wrapping the key in the EncryptedKey element. Alternatively key agreement may be performed as specified by the AgreementMethod. Information about the transported key or key derived by key agreement using other child elements under <KeyInfo>. These elements (prefixed by ds:) use the same syntax as was described for XML Signature. The <CipherData> element contains the actual encrypted data, either as base64-encoded data in CipherData or as a reference to an external location in CipherReference. <EncryptionProperties> contains additional information such as a date/time stamp or serial number of a cryptographic hardware element. XML Encryption is used in both SOAP and REST implementations for confidentiality.

2.5 SOAP Foundation

2.5.1 SOAP Version 1.2 Part 1

The most recent W3C standard for SOAP messaging is specified in SOAP Version 1.2 Part 1
http://www.w2.org/TR/soap12-part1

2.6 Message Security

2.6.1 WS-Security

The WS-Security standard specifies how to attach signature and encryption headers and security tokens to SOAP messages. The placement of WS-Security elements in a SOAP message is depicted below.

```
<env:Envelope>
 <env:Header>
  <wsse:Security>
   <wsse:Usernametoken, BinarySecuritytoken or SecurityTokenReference>
   <ds:Signature>
   <xenc:Encrypted>
   <wsu:Timestamp>
  </wsse:Security>
 </env:Header>
 <env:Body>
   <xenc:EncryptedData>
 </env:Body>
</env: Envelope>
```

The security header parent element, <wsse:Security>, provides a mechanism for attaching security related information to a SOAP message. The security header parent element has child elements for including a Security Token, an XML Signature, an XML Encryption element. Security Tokens are defined in WS-Security as access mechanisms and methods used for authentication and authorization. WS-Security supports the following security token types:

- Username/password
- OASIS SAML Assertion
- IETF X.509 Certificate
- IETF Kerberos token
- ISO Rights Expression Language

When a Username/password is included in plaintext, it is expected that the connection will provide encryption using either HTTPS or IPsec. An alternative is to use a digest of the password, a nonce, and the password using a cryptographic hash function.
The <ds:Signature> element can be an XML Signature as described in section 2.4.1. The <xenc: Encrypted> element can be an XML Encryption element as described in section 2.4.2. The <wsu:Timestamp> element is a Security Utility element which allows for including a time stamp. When the item to be encrypted is in the body of the message, the <xenc:EncryptedData> element contains the encrypted information.

WS-Security is the baseline standard for use in any SOAP-based SOA.

2.6.2 WS-SecureConversation

WS-SecureConversation is an OASIS standard specifies mechanisms for establishing and security contexts and for deriving session keys for these contexts. It is useful when there are multiple messages exchanged for a long-lived session (“conversation”) as opposed to a simple request/response exchange. Essentially it operates in a fashion similar to IKE for IPsec or TLS for HTTPS where a “shared secret” can be agreed to and used to derive a session key that secures multiple messages in the security context.

2.7 Reliable Messaging

2.7.1 WS-Reliability and WS-ReliableMessaging

The WS-Reliability and WS-ReliableMessaging standards are competing standards for reliable delivery of messages. The standards define mechanisms that provide guaranteed message delivery. Their semantics include provisions to ensure that: (1) a message will be delivered, (2) a duplicate message will not be delivered, or (3) a message will be delivered without duplication. These standards essentially provide a type of transport protocol at the SOAP messaging level.

2.8 Access Control

2.8.1 Security Assertion Markup Language (SAML)

SAML, developed by the Security Services Technical Committee of the Organization for the Advancement of Structured Information Standards (OASIS), is an XML-based framework for communicating user authentication, entitlement, and attribute information.
SAML is based on the concept of an "assertion" which is a declaration about a subject. The assertion may be a declaration about a subject's authentication, a list of a subject's authorization credentials, or generally as an expression of an authorization decision by an asserting party, i.e., a SAML Issuing Authority, which grants a subject access to a particular resource. In addition to defining the syntax for assertions, SAML also defines a request/response protocol for requesting assertions from an Issuing Authority. As described in section 2.6.1 SAML may be used as a Security Token in SOAP messaging in accordance with WS-Security. SAML is not limited to SOAP messaging. It may also be used in REST based web services. SAML is generally used as the basic building block for Single Sign On (SSO).

2.8.2 eXtensible Access Control Markup Language (XACML)

XACML is an OASIS XML-based standard that describes:

- a policy language and
- an access control decision request/response language

The policy language is used to describe general access control requirements and the request/response language lets you form a query to ask whether or not a given action should be allowed. The response includes one of the following four answers: Permit,
Deny, Indeterminate (a decision cannot be made due to an error or missing information in the request), or Not Applicable (the request can't be answered by this service).

In a typical configuration the Web service provider contains an entity called a Policy Enforcement Point (PEP). The PEP will form a request based on the consumer's attributes and the service requested and will then send this request to a Policy Decision Point (PDP), which will look at the request and the applicable policy and determine the appropriate response. If access is permitted the service request may be executed.

2.9 Policy

2.9.1 WS-Policy

WS-Policy is an XML language to represent the capabilities, constraints, and requirements of Web services. WS-SecurityPolicy is a subset of WS-Policy which provides security policy assertions describing specific security policies used with WS-Security, WS- Trust, and WS-Secure

Conversation. For example, if a web service requires a WS-Security header containing an X.509 Certificate Token Profile, WS-SecurityPolicy could be used to express that policy.

2.10 Security Management

2.10.1 XKMS

The XML Key Management Specification (XKMS) is used for key management and addresses PKI and key management network services in XML. XKMS is composed of two subparts: The XML Key Information Service Specification (X-KISS) and the XML Key Registration Service Specification (X-KRSS). X-KISS provides functions that support locating public keys given identifier information and binding of keys to identifier information. These functions are called the locate service and the validate service. X-KRSS permits an XML application to register its public-key pair with associated binding information to an XKMS trust service provider.

The XKMS standards are generally not used. As noted in NIST SP 800-96 [NIST 1], "while SOAP-compliant services exist to interact with a PKI (e.g., XKMS), most installed PKIs user older non-XML-based protocols"

2.10.2 WS-Trust

The WS-Trust standard is an OASIS specification that extends WS-Security and describes a Security Token Service (STS) model and a protocol for requesting and issuing security tokens. Under the STS model a service requestor will obtain a security token from an external STS. It may then present the token to a service provider which also trusts the STS and therefore grants access to the requested resource.

2.11 Identity Management

2.11.1 WS-Federation

WS-Federation is an OASIS specification that extends the WS-Trust STS model for federating identity across organizational boundaries.

3. Action Required by the Meeting

The meeting is invited to:

a) Note the information on web services security

b) Note that if the region moves forward with using web services to carry MET and AIM data, then the appropriate security measures must also be specified.

– – – – – – – – – – – –