ACTION 441: Impact of Basic Security Profile on XML DSIG v1.1

1) namespace prefixes

BSP:

- **xsd** - "http://www.w3.org/2001/XMLSchema"
- ds - "http://www.w3.org/2000/09/xmldsig#""
- c14n - "http://www.w3.org/2001/10/xml-exc-c14n#"

DSIG:

http://www.w3.org/2009/xmldsig11# dsig11:<!ENTITY dsig11 "http://www.w3.org/2009/xmldsig11#">

<CanonicalizationMethod Algorithm="http://www.w3.org/2006/12/xml-c14n11"/>

Comment: Should the BSP also have the dsig11 and x14n11 entries? Should DSIG have the xsd entry?

2) Comment: These definitions are not the same. They need to be consistent between documents.

a. Canonicalization Method

BSP: CANONICALIZATION_METHOD an element named ds:CanonicalizationMethod, included as a child of a SIGNED_INFO or a wsse:TransformationParameters child of a SIG_TRANSFORM

DSIG: CanonicalizationMethod is the algorithm that is used to canonicalize the SignedInfo element before it is digested as part of the signature operation. <element name="CanonicalizationMethod" type="ds:CanonicalizationMethodType"/>

b) Digest Method

BSP: DIGEST_METHOD is an element named ds:DigestMethod, included as a child of a SIG_REFERENCE.

DSIG: DigestMethod is a required element that identifies the digest algorithm to be applied to the signed object. This element uses the general structure here for algorithms specified in Algorithm Identifiers and Implementation Requirements. <element name="DigestMethod" type="ds:DigestMethodType"/>

c) Key Info

BSP: ENC_KEY_INFO - an element named ds:KeyInfo, included as a child of an ENCRYPTED_KEY or ENCRYPTED_DATA.

DSIG: KeyInfo is an optional element that enables the recipient(s) to obtain the key needed to validate the signature. KeyInfo may contain keys, names, certificates and other public key management information, such as in-band key distribution or key agreement data. <KeyInfo>

d) Reference

BSP: SIG_REFERENCE - an element named ds:Reference, included as a child of a SIGNED_INFO

DSIG: Reference is an element that may occur one or more times. It specifies a digest algorithm and digest value, and optionally an identifier of the object being signed, the type of the object, and/or a list of transforms to be applied prior to digesting.

<element name="Reference" type="ds:ReferenceType"/>

e) Transforms

BSP: SIG_TRANSFORMS - an element named ds:Transforms, included as a child of a SIG_REFERENCE.

BSP: SIG_TRANSFORM - an element named ds:Transform, included as a child of a SIG_TRANSFORMS.

DSIG: The optional Transforms element contains an ordered list of Transform elements; these describe how the signer obtained the data object that was digested. The output of each Transform serves as input to the next Transform. <element name="Transforms" type="ds:TransformsType"/>

f) Signature

BSP: SIGNATURE - an element named ds:Signature, included as a child of a SECURITY_HEADER.

DSIG: XML Signatures are applied to arbitrary digital content (data objects) via an indirection. Data objects are digested, the resulting value is placed in an element (with other information) and that element is then digested and cryptographically signed. </Signature>

g) Signature Method

BSP: SIGNATURE_METHOD - an element named ds:SignatureMethod, included as a child of a SIGNED_INFO.

DSIG: SignatureMethod is a required element that specifies the algorithm used for signature generation and validation. This algorithm identifies all cryptographic functions involved in the signature operation (e.g. hashing, public key algorithms, MACs, padding, etc.). <element name="SignatureMethod" type="ds:SignatureMethodType"/>

h) Signed Info

BSP: SIGNED_INFO - an element named ds:SignedInfo, included as a child of a SIGNATURE.

DSIG: The required SignedInfo element is the information that is actually signed. Core validation of SignedInfo consists of two mandatory processes: validation of the signature over SignedInfo and validation of each Reference digest within SignedInfo. <SignedInfo>

i) IssuerSerial

BSP: STR_ISSUER_SERIAL - an element named ds:X509IssuerSerial, included as a child of a child element named ds:X509Data of a SECURITY_TOKEN_REFERENCE.

DSIG: The X509IssuerSerial element, which contains an X.509 issuer distinguished name/serial number pair.

j) Key Name

BSP: STR_KEY_NAME - an element named ds:KeyName, included as a child of a SECURITY_TOKEN_REFERENCE.

DSIG: Typically, KeyName contains an identifier related to the key pair used to sign the message, but it may contain other protocol-related information that indirectly identifies a key pair. <element ref="ds:KeyName"/>

3) **Transport Layer Mechanisms**

BSP: RFC 2246: The TLS Protocol Version 1.0
Comment: Updated by RFC 4336: The Transport Layer Security (TLS) Protocol Version 1.1

4) Mandatory and Recommended Ciphersuites

BSP: Section 4.2.1 and 4.2.2

R5701 Any TLS-capable INSTANCE that is not FIPS compliant MUST support TLS_RSA_WITH_3DES_EDE_CBC_SHA

R5702 Any SSL-capable INSTANCE that is not FIPS compliant MUST support SSL_RSA_WITH_3DES_EDE_CBC_SHA

R5703 Any TLS-capable INSTANCE that is FIPS compliant MUST support TLS_RSA_FIPS_WITH_3DES_EDE_CBC_SHA

R5704 Any SSL-capable INSTANCE that is FIPS compliant MUST support SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA

TLS_RSA_WITH_AES_128_CBC_SHA or the FIPS equivalent

SSL_RSA_WITH_AES_128_CBC_SHA or the FIPS equivalent

DSIG: Section 6, Algorithms

Comment: This should be consistent with FIPS PUB 180-3 Secure Hash Standard, FIPS PUB 186-3: Digital Signature Standard (DSS), and the DSIG document section 6. The digest needs to include the bit string length with proper identifiers "http://www.w3.org/2000/09/xmldsig#dsa-sha1"

5) Discouraged and Prohibited Ciphersuites

BSP: Section 4.2.3 and 4.2.4

SSL_RSA_WITH_NULL_SHA

TLS_RSA_WITH_NULL_SHA

SSL_RSA_WITH_NULL_MD5

TLS_RSA_WITH_NULL_MD5

DSIG: Section 6, Algorithms

Comment: The digest needs to include the bit string length with proper identifiers "http://www.w3.org/2000/09/xmldsig#dsa-sha1". The BSP does not discuss the deprecation of SHA1.

6) Processing Order

BSP: section 6.1.1 In Order of Appearance

With signature before encryption…

With encryption before signature…

DSIG: ??

Comment: I thought there was text somewhere that discussed order, but I didn't find it in the document. This needs to be documents somewhere (BCP?).

7) Timestamps

BSP: section 7, Timestamps

DSIG: section 2.2 Extended Example

<timestamp xmlns="http://www.ietf.org/rfcXXXX.txt">

Comment: Timestamps are not discussed in the DSIG document, perhaps some text should be added. Also, should the RFC be identified in the example?

8) KeyName

BSP: section 8.4 Key Name References

<ds:KeyName>CN=Security WG, OU=BSP, O=WS-I, C=US</ds:KeyName>

DSIG: section 4.5.1 The KeyName Element

<element name="KeyName" type="string"/>

Comment: The BSP example is incorrect, however there is no correct example. One should be provided.

9) XML DSIG Reference

BSP: section 9. XML-Signature- link reference is to XML Signature Syntax and Processing (Second Edition), W3C Recommendation 10 June 2008

Comment: Should this reference XML Signature Syntax and Processing Version 1.1, W3C Editor's Draft 05 January 2010? Perhaps it cannot reference a draft.

10) Enveloping Signatures

BSP: section 9.1.1 Enveloping Signatures Prohibited- R3102 A SIGNATURE MUST NOT be an Enveloping Signature as defined by the XML Signature specification.

DSIG: Introduction- An XML Signature may be applied to the content of one or more resources. Enveloped or enveloping signatures are over data within the same XML document as the signature; detached signatures are over data external to the signature element.

Comment: Do we need to add text in the DSIG specification to point out this acception?

11) XPath

BSP: section 9.2.5 XPath References Where Necessary- <!-- This example is incorrect because it uses the http://www.w3.org/TR/1999/REC-xpath-19991116 transform instead of the http://www.w3.org/2002/06/xmldsig-filter2 transform -->

DSIG: section 6.1 Algorithm Identifiers and Implementation Requirements

Recommended

1. XPath

http://www.w3.org/TR/1999/REC-xpath-19991116

2. XPath Filter 2.0

http://www.w3.org/2002/06/xmldsig-filter2

Comment: This seems inconsistent. Why isn't REC-xpath-19991116 supported in BSP?

12) Transforms

BSP: section 9.3.3 Transform Algorithms

R5423 Any SIG_TRANSFORM Algorithm attribute MUST have a value of "http://www.w3.org/2001/10/xml-exc-c14n#" or "http://www.w3.org/2002/06/xmldsig-filter2" or "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#STR-Transform" or "http://www.w3.org/2000/09/xmldsig#enveloped-signature" or "http://docs.oasis-open.org/wss/oasis-wss-SwAProfile-1.1#Attachment-Content-Signature-Transform" or http://docs.oasis-open.org/wss/oasis-wss-SwAProfile-1.1#Attachment-Complete-Signature-Transform

Comment: If Enveloping Signatures is prohibitied, why is it one of the attributes?

13) Canonicalization Methods

BSP: section 9.4 Canonicalization Methods- R5404 Any CANONICALIZATION_METHOD Algorithm attribute MUST have a value of "http://www.w3.org/2001/10/xml-exc-c14n#" indicating that it uses Exclusive C14N without comments for canonicalization.

DSIG: section 3.1.1 Reference Generation- . In cases in which inclusive canonicalization is desired, we RECOMMEND that Canonical XML 1.1 [XML-C14N11] be used.

Comment: The Canonicalization Method is inconsistent.

14) SHA-1

BSP: 9.5.2 Whitespace in PrefixList- <ds:SignatureMethod Algorithm='http://www.w3.org/2000/09/xmldsig#rsa-sha1' />

BSP: 9.6.1 SHA-1 Preferred

BSP: 9.7.1 Algorithms

R5421 Any SIGNATURE_METHOD Algorithm attribute SHOULD have a value of "http://www.w3.org/2000/09/xmldsig#hmac-sha1" or "http://www.w3.org/2000/09/xmldsig#rsa-sha1".

DSIG: SHA-1 is deprecated.

Comment: This are inconsistent recommendations.

15) Signature Encryption

BSP: section 9.10.1 Encrypt Only Entire Signature

> If the value of a ds:DigestValue element in a SIGNATURE needs to be encrypted the entire parent ds:Signature element MUST be encrypted.
>
> R5440 A SIGNATURE MUST NOT have any xenc:EncryptedData elements amongst its descendants.

DSIG: No requirement for this.

Comment: Does it make sense to include this requirement in the DSIG specification, perhaps as an example?

16) X509IssuerSerial

BSP: 13.2.5 KeyIdentifier or X509IssuerSerial for External References- see correct example

DSIG: 4.5.4 The X509Data Element- see first example

Comment: the DSIG example begins with <KeyInfo>, however the BSP does not, this should be consistent.

17) X509IssuerSerial Value

BSP: section 13.2.8 X509IssuerSerial Value- DNames are encoded as follows…

DSIG: section 4.5.4.1 Distinguished Name Encoding Rules

Comment: The BSP has additional rules, should they be included in the DSIG (or BCP)?

18) Digest Values

BSP: section 18.2.4 Digest Values

R6104 Any SIG_REFERENCE to a MIME_PART not containing XML MUST include a ds:DigestValue calculated using MIME canonicalization according to the requirements of the MIME Type.

R6108 Any SIG_REFERENCE to a MIME_PART containing the "application/text" content type MUST include a ds:DigestValue which is calculated after the text is canonicalized according to MIME canonicalization algorithm for text.

R6109 Any SIG_REFERENCE to a MIME_PART containing an XML content type MUST include a ds:DigestValue which is calculated after the XML is canonicalized according to Exclusive XML Canonicalization without comments, as specified by the URI "http://www.w3.org/2001/10/xml-exc-c14n#".

DSIG: 4.4.3.6 The DigestValue Element

Comment: The BSP restrictions are not in the DSIG specification, should they be (or in BCP)?

19) Security Token Substitution

BSP: section 19.3 Security Token Substitution

19.3.1 Security Token Substitution

19.3.2 Security Token Reference in Subsequent Messages

19.4 Protecting against removal and modification of XML Elements

19.5 Only What is Signed is Protected

19.6 Use of SHA1

DSIG: 2.2 Extended Example (Object and SignatureProperty)

Comments: The BSP restrictions are not in the DSIG specification, should they be (or in BCP)?