ebXML CPPA Technology

Dale Moberg, Cyclone Commerce

Chair, OASIS ebXML TC

Dmoberg@cyclonecommerce.com

Collaboration Protocol Profiles and Agreements: Export Use Case

- In a nutshell, the security, packaging, transport, and XML messaging "infrastructure" *capabilities* for engaging in specific business processes are announced. Export either CPPs or CPA templates
- Can be discovered using Registries and Repositories, made available through services, or obtained directly as files. (JAXR API publish support)

OASIS

Implementation Use Case: Import

- Once the Profiles are available, draft CPAs can be exchanged and agreed upon. The CPA is an agreement on which of the capabilities will be used for XML messaging.
- Messaging and other middleware software MAY import the CPA information for automated configuration. The ROI is in increased automation of configuration lifecycle management.

DESIGN
DEVELOP
DEPLOY

OASIS

Other Use Cases (JSR 157)

- Given BPSS, create CPP or CPA template for middleware systems.
- Given 2 CPPs, create draft CPA or CPA template for negotiation .
- Given CPA, check validity and acceptablity.
- Given CPA, sign CPA.
- Given CPA, update for new configuration.

DESIGN
DEVELOP
DEPLOY

Structure of CPPs

- Defines associations between:
- Party, Role, Service, Action, Packaging and Delivery Channels
- Preferences on BusinessTransactionCharacteristics
- And
- MessagingCharacteristics

Party,Role,Service,Action

- PartyInfo identified by partyName attribute, 1 or more PartyId and PartyRef.
- CollaborationRole/Role: “buyer” “seller”
- Service: “RosettaNet” “cXML”
- Action: “PurchaseOrder” “ReceiptAcknowledgment”

DeliveryChannels Focus

- Combinations of DocExchange and Transport that document the capabilities for using different transports (SMTP, FTP, HTTP) with security (SSL3, TLS) and capabilities for digital enveloping, digital signature, reliable messaging, and XML extensions supported (indicated by namespace URI)

DocExchange Focus

- Contains the ebXMLSender (Receiver)Binding elements
- Can be generalized to other messaging approaches by defining different subordinate elements.
- Contains IDREFs to PKI alignment details for digital envelope and digital signature (non-repudiation) resolving to SecurityDetails/TrustAnchor and Certificate elements. Uses XMLDsig KeyInfo elements to contain or refer to certificates. How to check for alignment. Checking requires XMLDsig aware software.

PKI Alignment DocExchange Example

- Within CanReceive context, an IDREF to DeliveryChannel and then an IDREF to DocExchange, permits finding Certificates used for digital signatures or envelopes.
- URL for ebXML CPPA schema: `http://www.oasis-open.org/committees/ebxml-cppa/schema/cpa-example-2_0b.xml`

Digital Envelope PKI Alignment SenderSide

- <element name="SenderDigitalEnvelope">
- <complexType><sequence><element name="DigitalEnvelopeProtocol" type="tns:protocol.type"/>
- <element ref="tns:EncryptionAlgorithm" maxOccurs="unbounded"/>
- <element name="EncryptionSecurityDetailsRef" type="tns:SecurityDetailsRef.type" minOccurs="0"/>
- </sequence></complexType></element>

Digital Envelope ReceiverSide

- <element name="ReceiverDigitalEnvelope">
- <complexType><sequence><element name="DigitalEnvelopeProtocol" type="tns:protocol.type"/>
- <element ref="tns:EncryptionAlgorithm" maxOccurs="unbounded"/>
- <element name="EncryptionCertificateRef" type="tns:CertificateRef.type"/>
- </sequence></complexType></element>

SecurityDetails

- <complexType>
- <sequence>
- <element name="SecurityDetails">
- <element ref="tns:TrustAnchors" minOccurs="0"/>
- <element ref="tns:SecurityPolicy" minOccurs="0"/>
- </sequence>
- <attribute name="securityId" type="ID“ use="required"/>
- </complexType> </element>

Trust Anchors

- <element name="TrustAnchors">
- <complexType><sequence>
- <element name="AnchorCertificateRef" type="tns:CertificateRef.type" maxOccurs="unbounded"/>
- </sequence></complexType>
- </element>

Certificate

- <element name="Certificate">
- <complexType><sequence>
- <element ref="ds:KeyInfo"/>
- </sequence>
- <attribute name="certId" type="ID" use="required"/>
- </complexType>
- </element>

EncryptionAlgorithm

- element name="EncryptionAlgorithm">
- <complexType> <simpleContent>
- <extension base="tns:non-empty-string">
- <attribute name="minimumStrength" type="integer"/>
- <attribute name="oid" type="tns:non-empty-string"/>
- <attribute name="w3c" type="tns:non-empty-string"/>
- <attribute name="enumerationType" type="tns:non-empty-string"/>
- </extension></simpleContent></complexTyp

Other Examples

- Digital Signature
- Transport (SSL or TLS)
- ApplicationLevelSecurity
- AccessAuthentication
- For each security “module,” both PKI alignment and algorithm details can be announced, negotiated, and agreed upon.