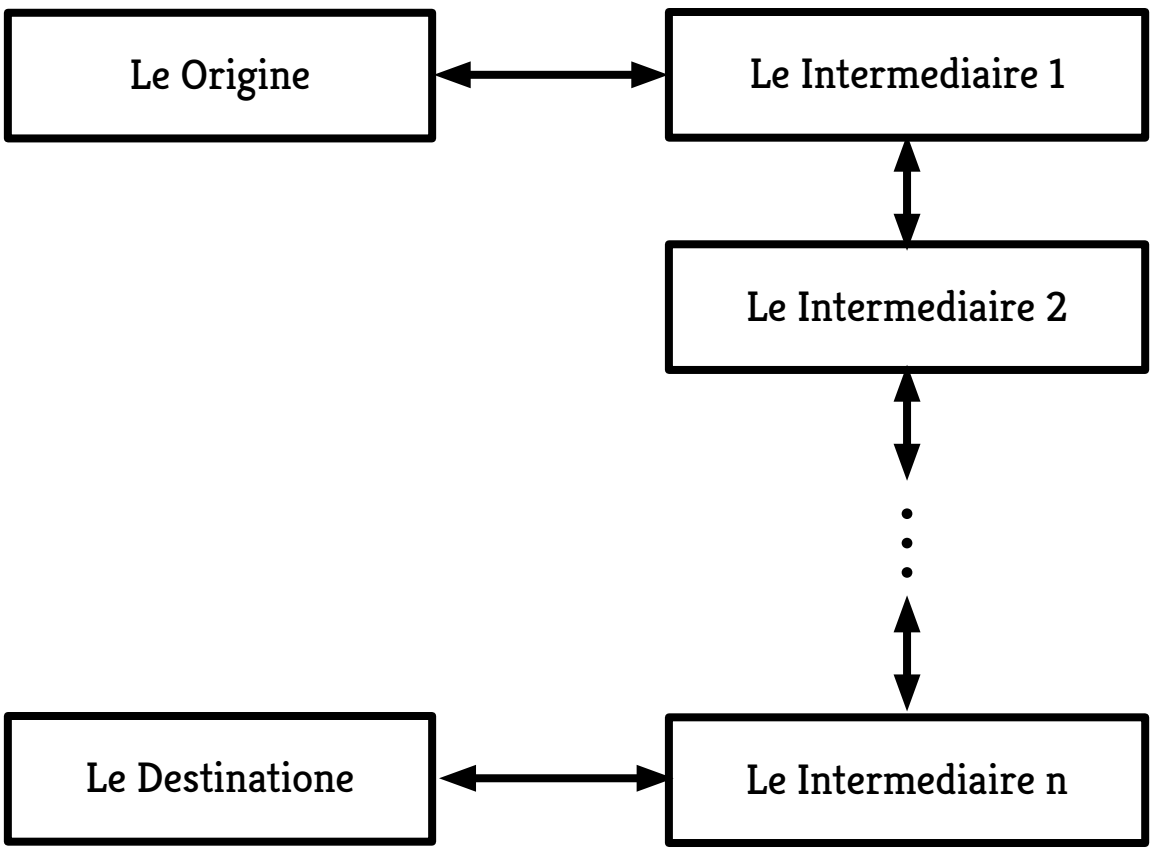

End-to-End
Web Services
Security

What is End-to-End Security?

Security from the origin to the destination.

A Pictorial Representation*

* (In pseudo-French for silliness)



Achieving End-to-End Security

Secure the transport or message layer


Message Level Security
e.g. WS-Security
ATLANTA
POLICE
Transport Level Security
e.g. SSL/TLS

SSL/TLS

- TLS = Transport Layer Security
- SSL = Secure Sockets Layer
- Cryptographic protocols that provide security over the Internet
- Uses a Certifiying Authority (CA) and (typically) symmetric key encryption
- Usually doesn't not support mutual authentication

WS-Security

- Member of the WS-* spec family
- Provides security extensions to SOAP
- Specifies how to enforce confidentiality and integrity on SOAP messages
- Allows for various security token formats to be attached (e.g. X.509 certificates, Kerberos tickets)

WS-Security

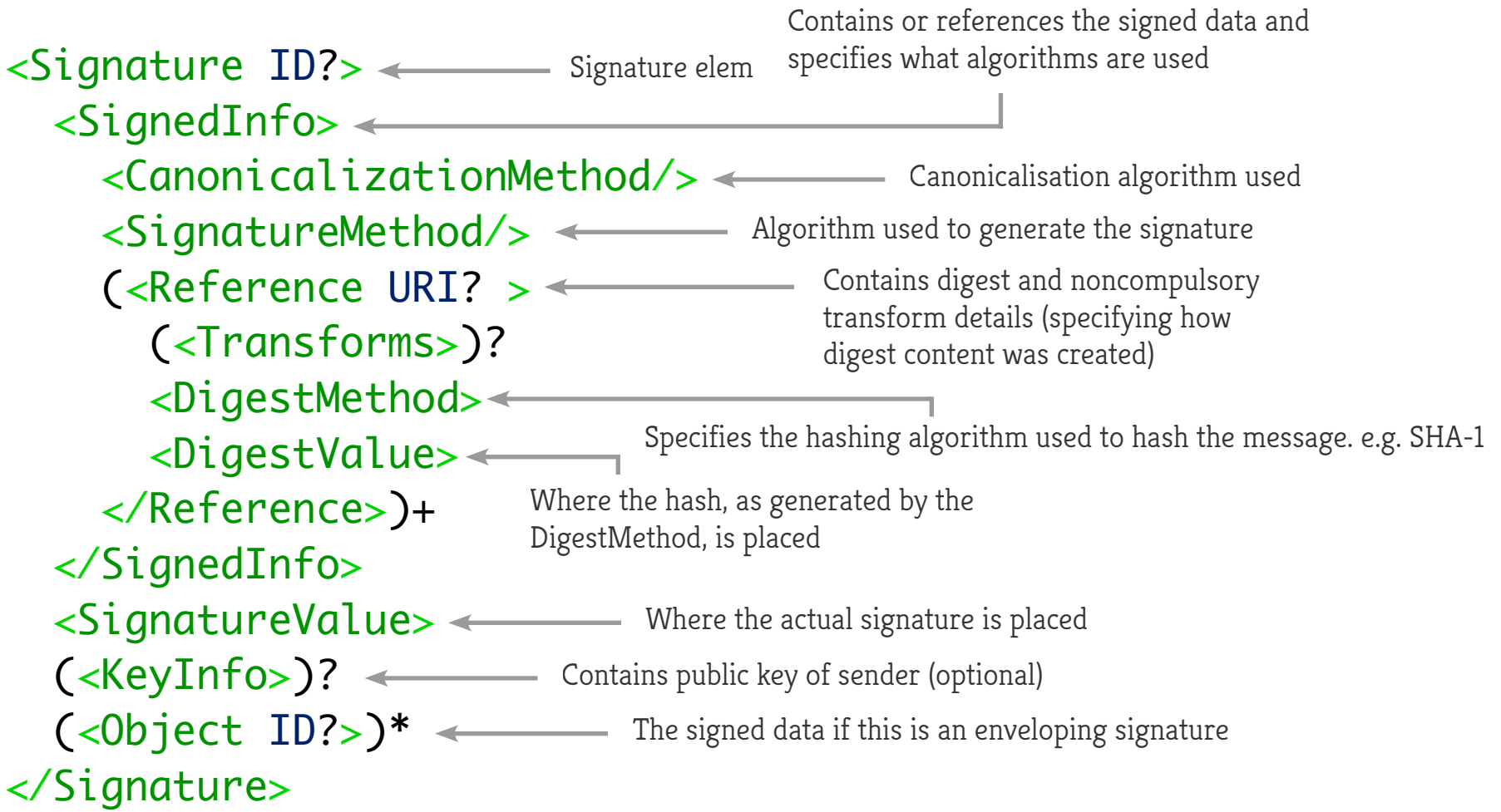
- Confidentiality → XML-Encryption
- Integrity → XML-Signature
- Authentication → Attach security token
- Authorisation → Attach security token
- Non-Repudiation → Audit trail + XML-Sig


XML-Signature

XML-Signature

- Used to digitally sign resources (of any type, but typically XML documents)
- Detached signature → signing a resource outside the XML document
- Enveloped signature → signing some part of XML document
- Enveloping signature → XML document contains signed data

XML-Signature Structure




XML-Encryption

XML-Encryption

- Defines how to encrypt XML elements
- Secures portions (including headers) of SOAP messages
- Contains a manifest of encrypted elements in a `ReferenceList`
- Encrypted data components are placed in `EncryptedData` elements

XML-Encryption Structure

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

XML-Encryption Example

```
<soap:Header
  xmlns:wsse="http://schemas.xmlsoap.org/
ws/2002/07/secext">
  <wsse:Security soap:mustUnderstand="1" >
     <xenc:ReferenceList>
       <xenc:DataReference URI="#bodyID-1"
/>
     </xenc:ReferenceList>
  </wsse:Security>
</soap:Header>
```

XML-Encryption Example

```
<soap:Body>
  <xenc:EncryptedData Id="bodyID-1"
    Type="http://www.w3.org/2001/04/xmlenc#Content">
    <xenc:EncryptionMethod Algorithm=
      "http://www.w3.org/2001/04/xmlenc#tripledes-cbc" />
    <KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">
      <KeyName>Our Cool Symmetric Key</KeyName>
    </KeyInfo>
    <xenc:CipherData>
      <xenc:CipherValue>
        InmSSXV5UiTzzzzzzITEC833i5Aw3S0M3...zzzY7RVZQMg==
      </xenc:CipherValue>
    </xenc:CipherData>
  </xenc:EncryptedData>
</soap:Body>
```

Do we really need WS-Security?

YES

YES

- End-to-end security is sometimes necessary, especially when hops over untrusted intermediaries are involved
- More flexibility (1) Can selectively encrypt parts of messages (2) Can use different encryption methods for different parts of the message

NO

NO

- Not all implementations require it
- End-to-end security is expensive
- Complex to implement
- Support is not widespread across all programming languages


Any Questions?

Photo Credits

- Title Slide by pshutterbug
  http://www.flickr.com/photos/pshan427/815348106/
- President's Car by Getty Images
  http://autoworld.wordpress.com/2009/01/22/barack-obamas-new-cadillac-presidential-limousine/
- Police Motorcade by TimothyJ
  http://www.flickr.com/photos/tjc/413204218/
- XML-Signature Slide by bizmac
  http://www.flickr.com/photos/bizmac/2492390738/
- XML-Encryption Slide by Ryan Somma
  http://www.flickr.com/photos/ideonexus/5175892723/