Sicherheit in Web Services

Seminar
Service-orientierte Software Architekturen

Melanie Storm

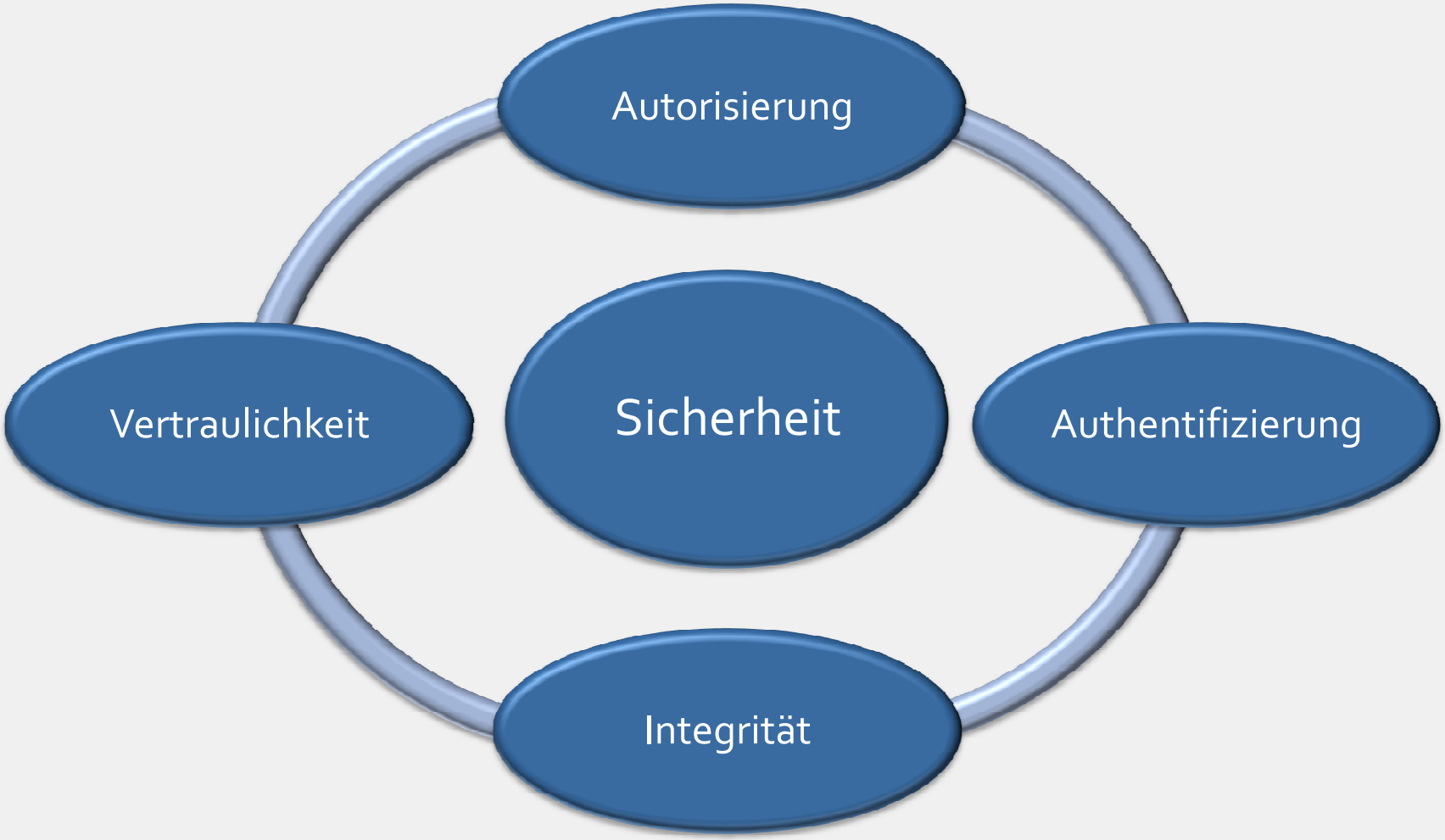
Agenda

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Motivation**
- **Fallbeispiel**
- **WS-Security**
  - XML Encryption
  - XML Signature
- **WS-Policy**
  - WS-SecurityPolicy
- **WS-Trust**
- **WS-SecureConversation**
- **WS-***
- **Fazit**

Motivation
Ziele der Sicherheit

fhwedel
UNIVERSITY OF APPLIED SCIENCES



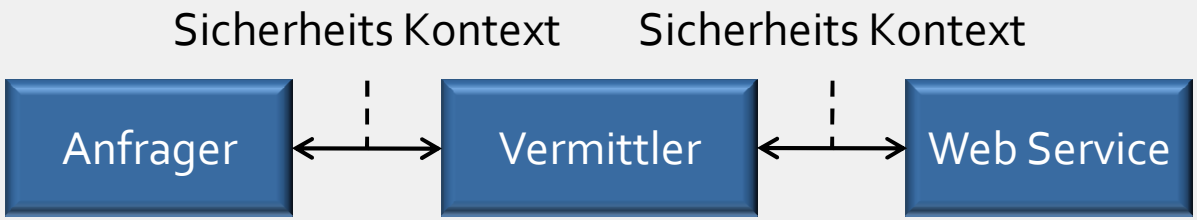
Motivation

SSL/TLS reicht nicht mehr aus

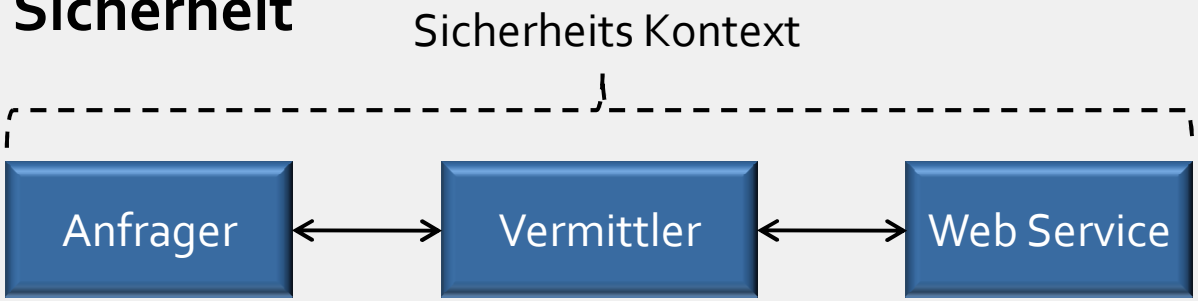
fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Reine Transportsicherheit, nicht informationsgebunden**
- **Verschlüsselt gesamte Nachricht**

Ende zu Ende Sicherheit

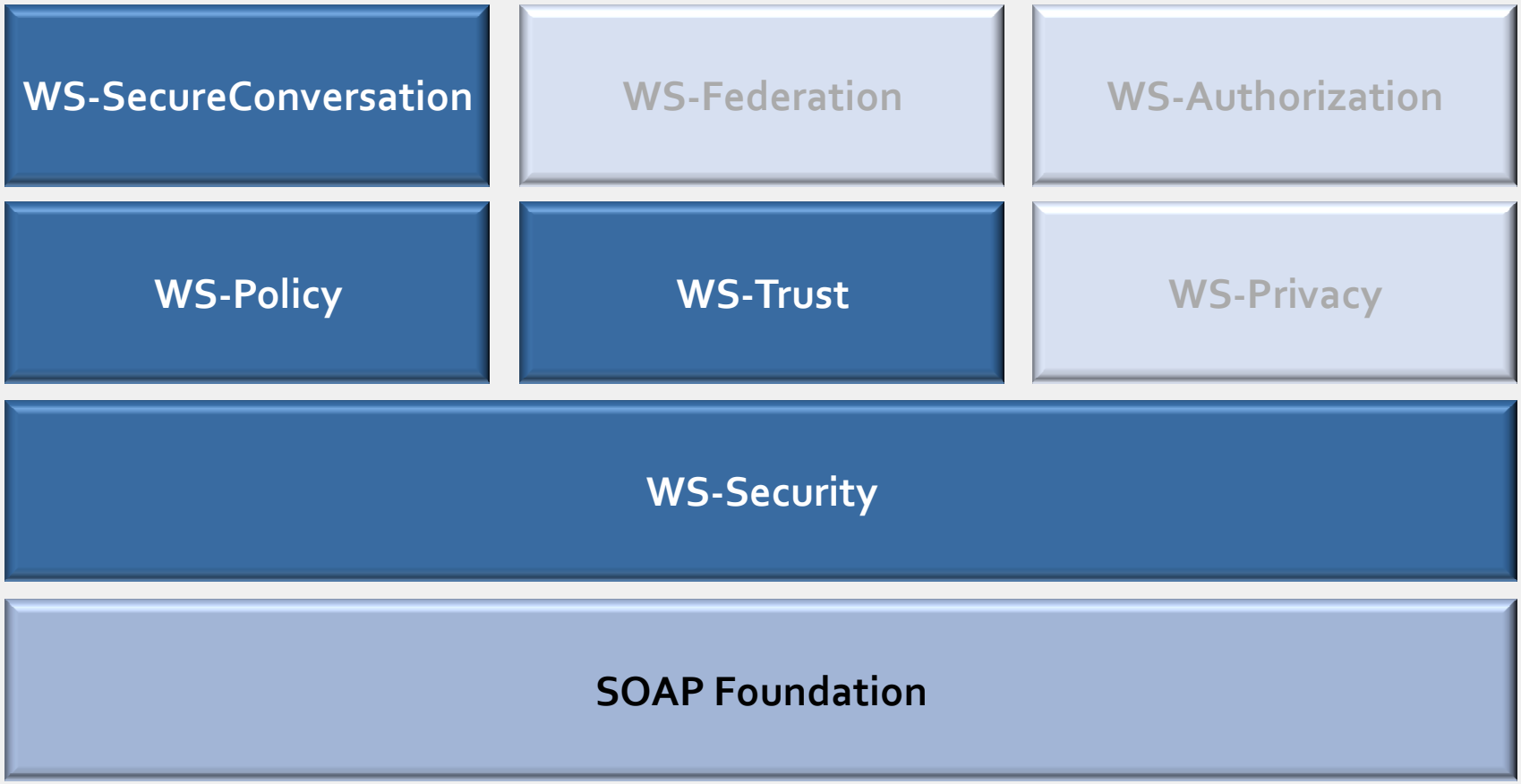


Punkt zu Punkt Sicherheit



Security Roadmap

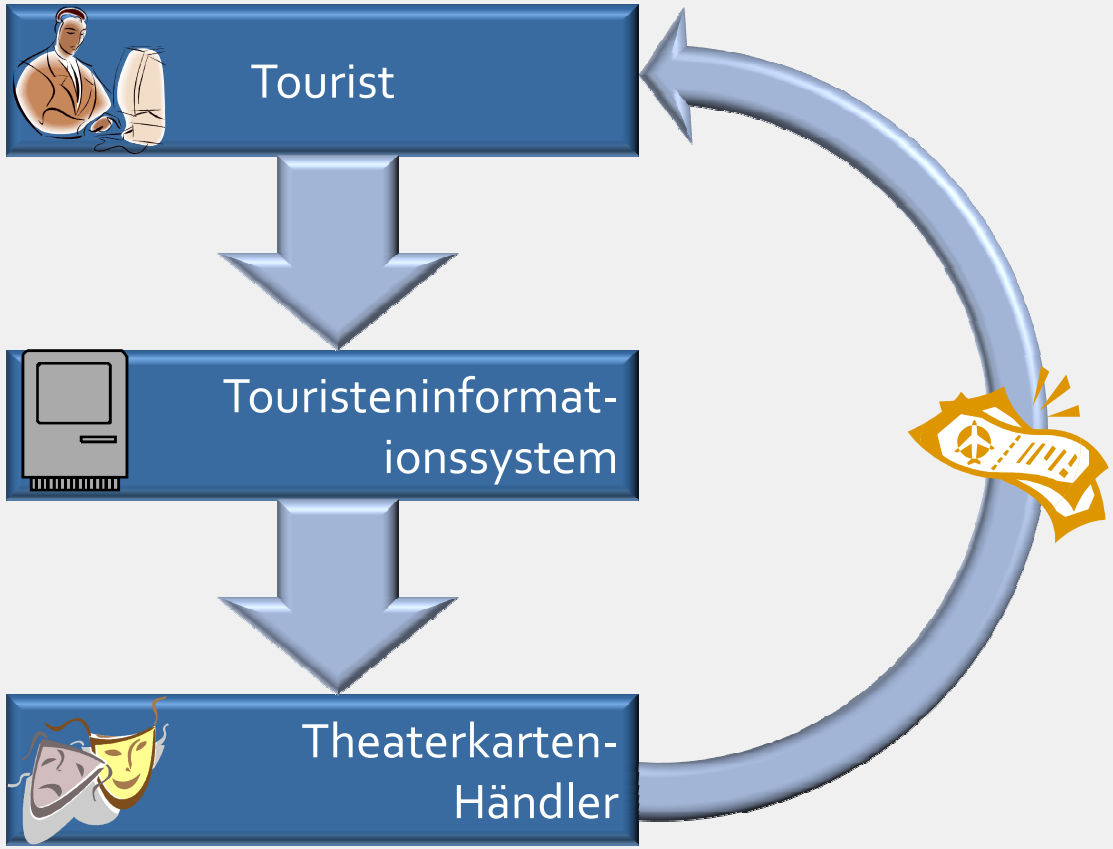
fhwedel
UNIVERSITY OF APPLIED SCIENCES



Fallbeispiel

Tourist kauft Theaterkarten

fhwedel
UNIVERSITY OF APPLIED SCIENCES



Fallbeispiel
SOAP-Nachricht

fhwedel
UNIVERSITY OF APPLIED SCIENCES

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
   xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <SOAP-ENV:Body>
      <vorstellung>
         …
      </vorstellung>
      <kontoverbindung>
         <inhaber type="xsd:string">Max Mustermann</inhaber>
         <blz type="xsd:string">20050550</blz>
         <ktonr type="xsd:string">1379345876</ktonr>
      </kontoverbindung>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

WS-Security

XML Encryption - Einleitung

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Möglichkeit XML-Dokumente ver- und entschlüsseln zu können**
- **Beliebige Dokumententeile**
- **Beliebig häufig, aber nicht verschachtelbar**

WS-Security
XML Encryption - Aufbau

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **EncryptedData**
  - Wurzelelement
- **EncryptionMethod**
  - Gibt den Verschlüsselungsalgorithmus an
  - Optional
- **CipherValue**
  - Verschlüsselte Daten

WS-Security
XML Encryption - Beispiel

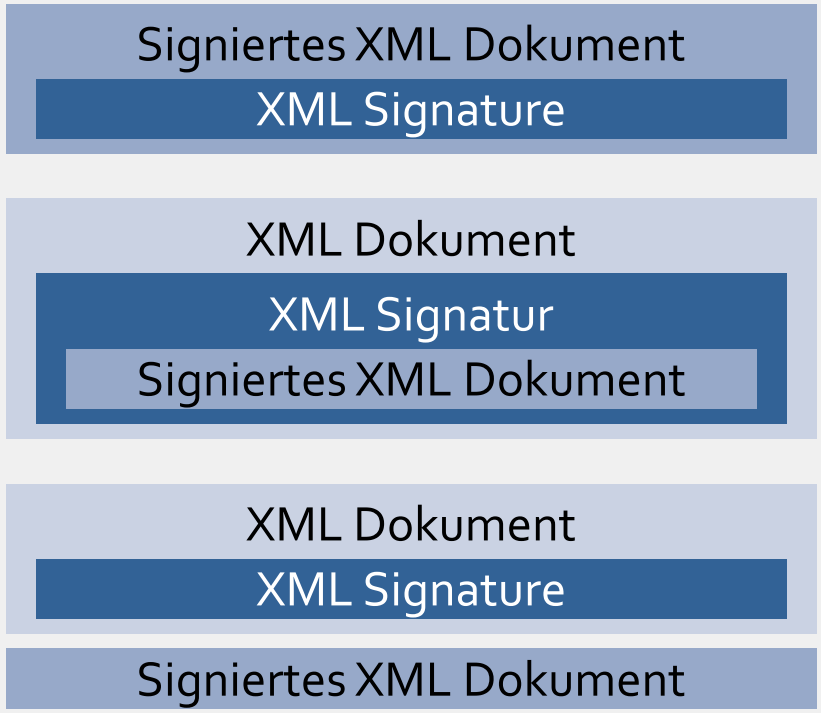
fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Was soll verschlüsselt werden? (ReferenceList)**
- **Zu verschlüsselndes Element durch EncryptedData ersetzen**
- **Womit soll verschlüsselt werden?**

WS-Security

XML Signature - Einführung

- **Möglichkeit XML-Dokumente zu signieren**
- **Beliebige Dokumentteile**
- **Beliebige Verschachtelungstiefe**
- **3 Signatur Typen**
  - Enveloped
  - Enveloping
  - Detached



WS-Security
XML Signature - Aufbau

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Signature**
  - Wurzelelement
- **SignedInfo**
  - Reference
    - Zu signierende Daten
  - DigestMethod
    - Algorithmus zur Hashwertberechnung
  - Digest Value
    - Hashwert der jeweiligen Ressource
- **SignatureValue**
  - Eigentliche Signatur

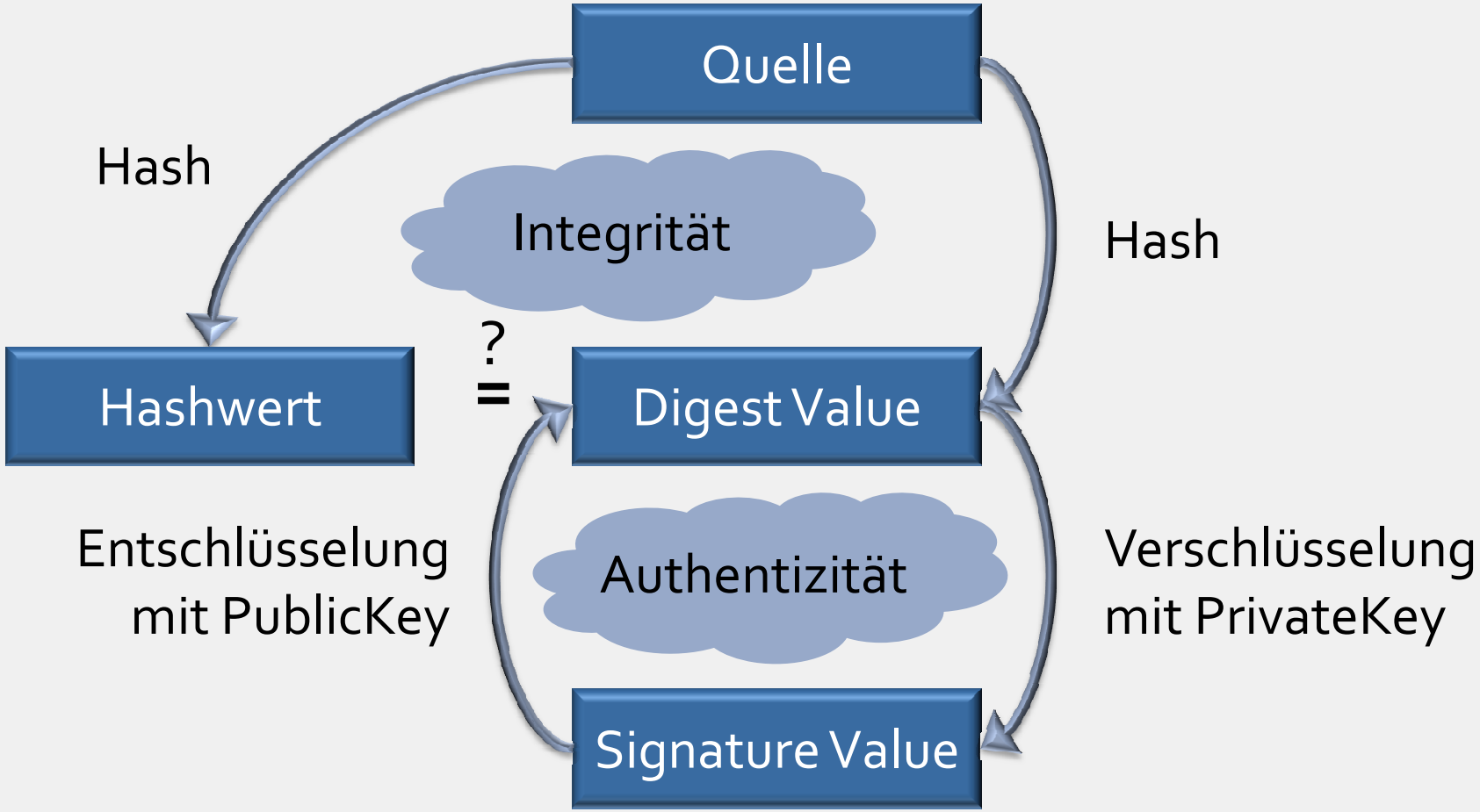
WS-Security
XML Signature - Beispiel

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Was soll signiert werden? (Reference)**
- **Hashwert berechnen (DigestValue)**
- **SignedInfo**
  - Normalisierungsmethode
  - Signierungsmethode
  - Reference
- **Hashwert der SignedInfo signieren (SignatureValue)**
- **Evtl. Schlüssel speichern (KeyInfo)**

WS-Security

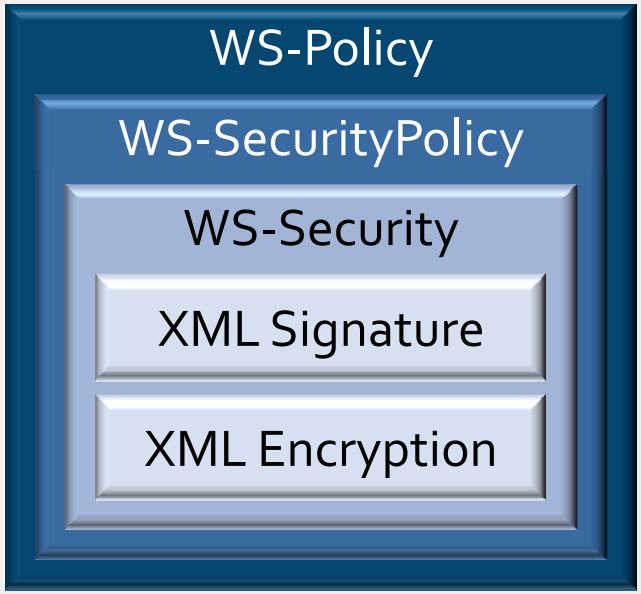
XML Signature - Ablauf



WS-Policy
eine kurze Wiederholung

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **WS-Policy**
  - Rahmenwerk zur Strukturierung und Darstellung von Anforderungen
- **WS-SecurityPolicy**
  - Semantik für WS-Policy konforme Inhalte zur Darstellung von Sicherheitsanforderungen
- **Ziel**
  - Vereinbarung von Bedingungen für das Zusammenspiel von zwei Web Services



WS-Policy
einführendes Beispiel

fhwedel
UNIVERSITY OF APPLIED SCIENCES

```
<wsp:Policy>
  <wsp:ExactlyOne>
    <wsp:All>
      <wsse:SecurityToken>
        <wsse:TokenType>wsse:Kerberosv5TGT</wsse:TokenType>
      </wsse:SecurityToken>
    </wsp:All>
    <wsp:All>
      <wsse:SecurityToken>
        <wsse:TokenType>wsse:X509v3</wsse:TokenType>
      </wsse:SecurityToken>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
```

WS-Policy
WS-SecurityPolicy

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **WS-SecurityPolicy**
  - SecurityTokenAssertion
    - Spezifiziert geforderte Tokens
  - Confidentiality Assertion
    - Teil der Nachricht muss verschlüsselt werden
  - Integrity Assertion
    - Teil der Nachricht muss signiert werden
  - Visibility Assertion
    - Teil der Nachricht muss sichtbar sein
  - Security Header Assertion
    - Verpflichtet z.B. zur Nutzung von References
  - MessageAge Assertion
    - Höchstalter der Nachricht

WS-Trust
Einleitung

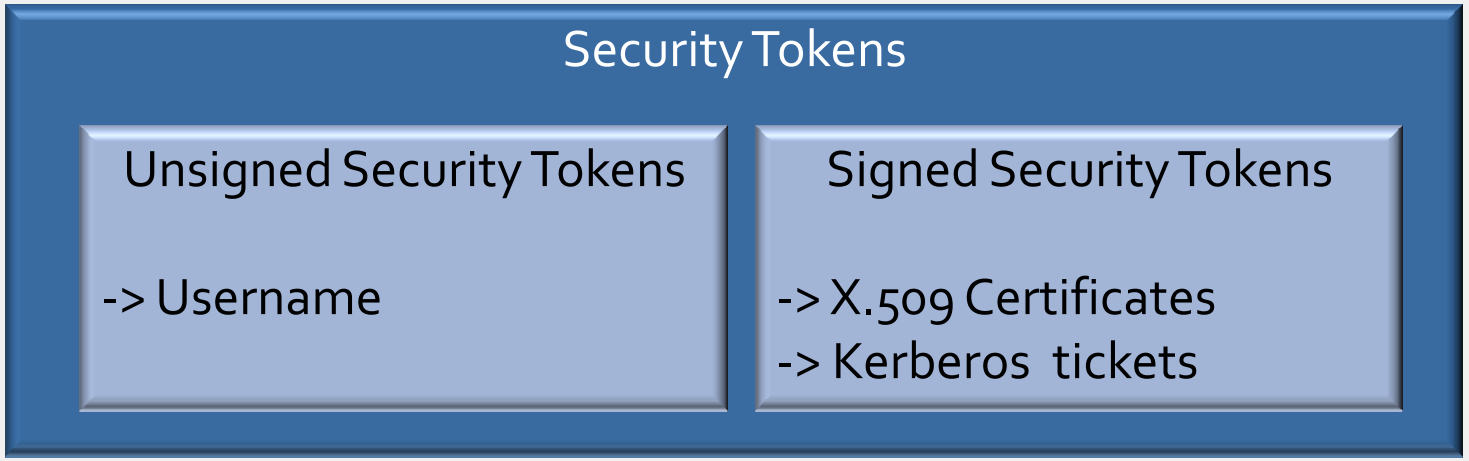
fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Problem**
  - Ein Requester kann evtl. die in der WS-Policy des Webservice genannten Sicherheitsnachweise nicht erbringen
- **Lösung**
  - Ein dritter Webservice verifiziert die Security Claims des Requesters und stellt Security Tokens aus, mit denen der Requester dann die Anforderungen des Webservice erfüllen kann.

WS-Trust
Security Tokens

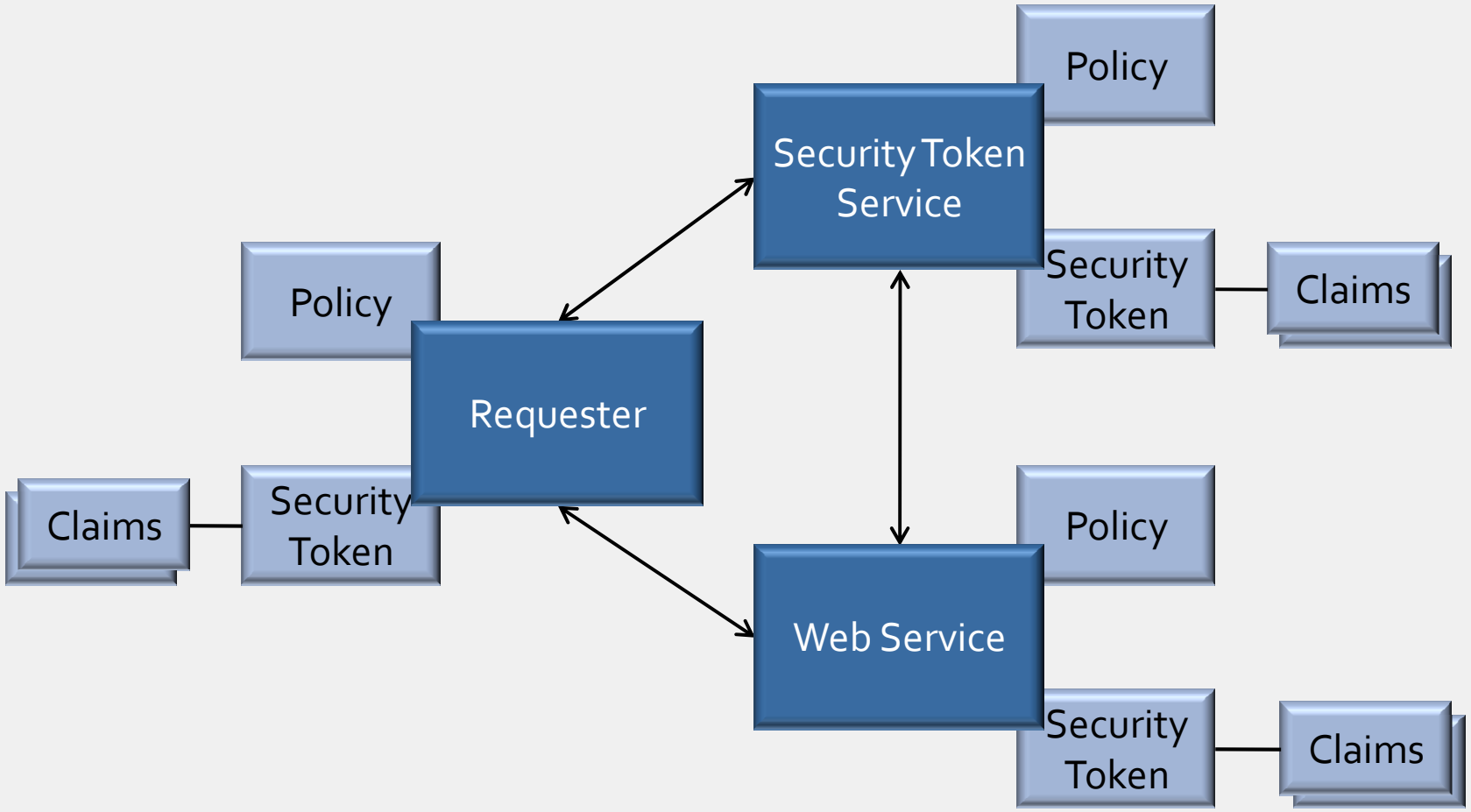
fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **Anforderung, Ausgabe, Austausch und Validierung von Security Tokens**
  - Username/Password Token
  - Binary Security Token
- **WS-Security trifft nur Aussagen zur Einbindung**



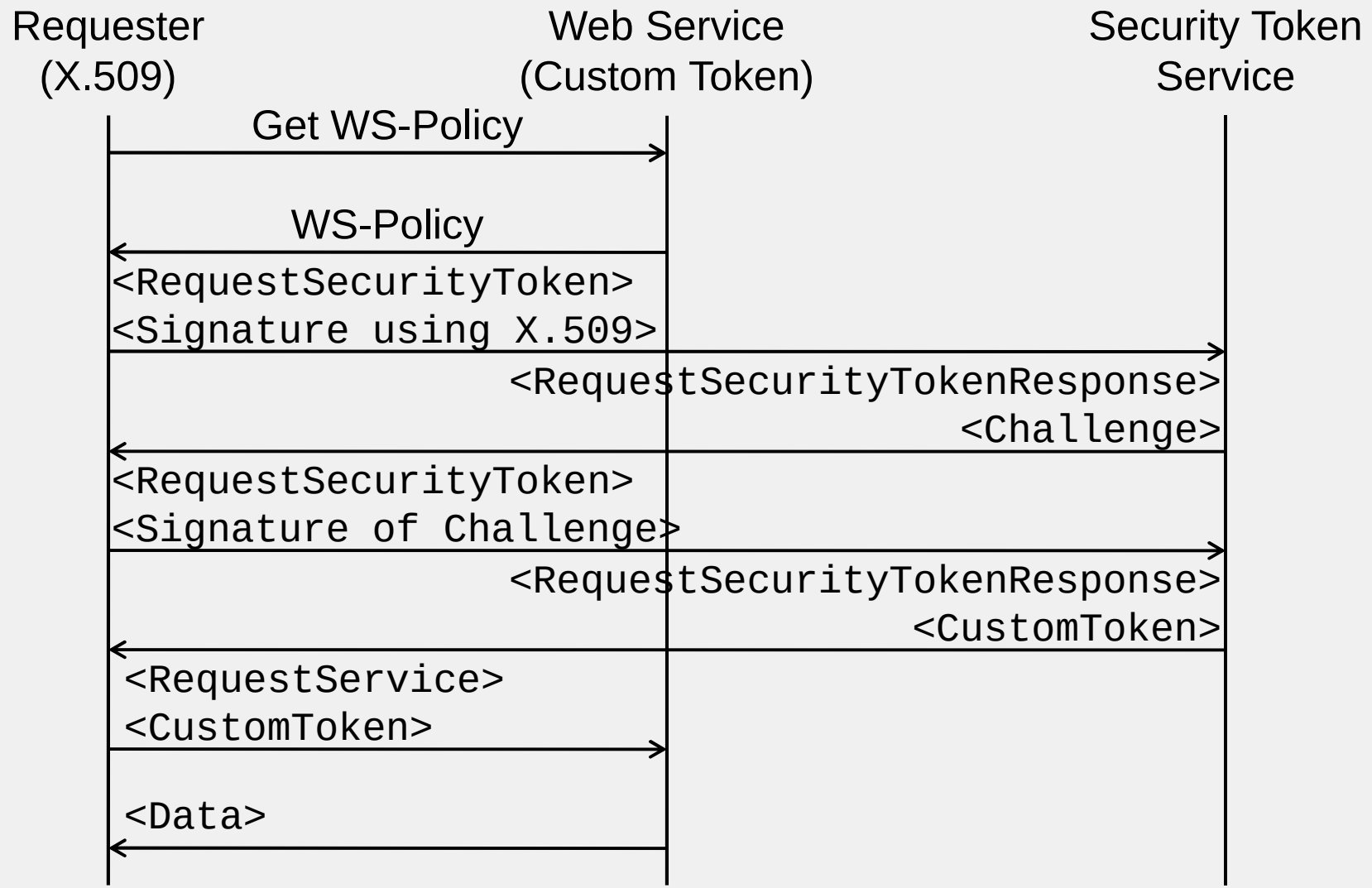
WS-Trust
Komponenten

fhwedel
UNIVERSITY OF APPLIED SCIENCES



WS-Trust

beispielhafter Ablauf



WS-SecureConversation

- **Definiert <SecurityContextToken> zum Speichern von Security Contexten**
  - Reduziert den Aufwand, der durch das Sichern einzelner Nachrichten entstehen kann

WS-*

Überblick restliche Spezifikationen

fhwedel
UNIVERSITY OF APPLIED SCIENCES

- **WS-Federation**
  - Vereinigung von Sicherheitsdomänen
- **WS-Privacy**
  - Syntax und Semantik zur Einbindung von Datenschutz Policies
- **WS-Authorization**
  - Zugriffsberechtigungen werden geprüft

Fazit

Wie sinnvoll sind diese Spezifikationen

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Potential vorhanden

Fragen

fhwedel
UNIVERSITY OF APPLIED SCIENCES

Vielen Dank für die Aufmerksamkeit!

Beispiel XML-Signature

```
<SOAP-ENV:Envelope
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2003/06/secext"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:wsu="http://schemas.xmlsoap.org/ws/2003/06/utility">
  <SOAP-ENV:Header>
    <wsse:Security>
      <ds:Signature>
        <ds:SignedInfo>
          <ds:CanonicalizationMethod
            Algorithm=
            "http://www.w3.org/2001/10/xml-exc-c14n#"/>
          <ds:SignatureMethod
            Algorithm=
            "http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
          <ds:Reference URI="#MessageBody">
            <ds:DigestMethod
              Algorithm=
              "http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue>
              JwFsd3eQc0iXlJm5PkLh7...
            </ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>
          BSxlJbSiFdm5Plhk...
        </ds:SignatureValue>
        <ds:KeyValue>
          <ds:DSAKeyValue>
            <ds:P>…</ds:P> <ds:Q>…</ds:Q>
            <ds:G>…</ds:G> <ds:Y>…</ds:Y>
          </ds:DSAKeyValue>
        </ds:KeyValue>
      </ds:Signature>
    </wsse:Security>
  <SOAP-ENV:Body wsu:Id="MessageBody">
    …
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Beispiel XML-Encryption

```
<SOAP-ENV:Envelope
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2003/06/secext"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#"
  xmlns:wsu="http://schemas.xmlsoap.org/ws/2003/06/utility">
  <SOAP-ENV:Header>
    <wsse:Security>
      <xenc:ReferenceList>
        <xenc:DataReference URI="#MessageBody"/>
      </xenc:ReferenceList>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <vorstellung>
      …
    </vorstellung>
    <xenc:EncryptedData Id="MessageBody"
      Type="http://www.w3.org/2001/04/xmlenc#Element">
      <xenc:EncryptionMethod
        Algorithm=
        "http://www.w3.org/2001/04/xmlenc#rsa-1_5/">
      <xenc:CipherData>
        <xenc:CipherValue>
          sdjGhfHhsky…
        </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Beispiel WS-SecurityPolicy

```
<wsp:Policy
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/12/secext"
  xmlns:wsp="http://schemas.xmlsoap.org/ws/2002/12/policy">
  <wsp:SpecVersion   wsp:Usage="wsp:Required"
    URI="http://schemas.xmlsoap.org/ws/2002/07/secext"/>
  <wsse:Confidentiality    wsp:Usage="wsp:Required">
    <wsse:Algorithm   Type="wsse:AlgEncryption"
      URI="http://www.w3.org/2001/04/xmlenc#3des-cbc"/>
    <MessageParts>
      wsp:GetInfosetForNode(wsp:GetBody(.))
    </MessageParts>
  </wsse:Confidentiality>
  <wsse:Integrity    wsp:Usage="wsp:Required">
    <wsse:Algorithm    Type="wsse:AlgCanonicalization"
      URI=
      "http://www.w3.org/Signature/Drafts/xml-exc-c14n"/>
    <wsse:AlgorithmType="wsse:AlgSignature"
      URI="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
    <wsse:SecurityToken>
      <wsse:TokenType>wsse:X509v3</wsse:TokenType>
    </wsse:SecurityToken>
    <MessageParts
      Dialect=
      "http://schemas.xmlsoap.org/2002/12/wsse#soap">
      S:Body
    </MessageParts>
  </wsse:Integrity>
  <wsse:Visibility wsp:Usage="wsp:Required">
    <MessageParts>
      wsp:GetInfosetForNode(wsp:GetBody(.))
    </MessageParts>
  </wsse:Visibility>
  <wsse:SecurityHeader  wsp:Usage="wsp:Required"
              Must Prepend="true"
              MustManifestEncryption="true"/>
  <wse:MessageAge wsse:Usage="wsp:Required" Age=3600"/>
</wsp:Policy>
```