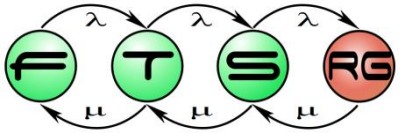
(Web)Szolgáltatások (WS, WS-*)

Szolgáltatásintegráció 2014.



Elosztott rendszerek

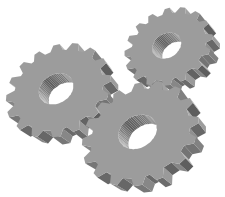
- Elosztott rendszerek
  - Egy *hálózat*on lévő számítógépek
    - Tipikus példa: Internet
  - Üzenet alapú kommunikáció
- Motiváció
  - Erőforrás megosztás
  - Skálázhatóság
  - Modularizáció
  - Együttműködés
- Követelmények
  - Kommunikációs réteg

XML webszolgáltatások

- Alkalmazások közötti adatcserére szolgáló protokollok és szabványok gyűjteménye
- Szabvány: XML alapúak
  - Strukturált szöveges állomány
  - Kötött formátum (séma)
- Nem csak nyílt Web-es környezetben
- Lazán csatolt alkalmazások
- Főbb fejlesztők
  - Apache, IBM, HP, SUN & Microsoft (.NET)
  - http://www.webservices.org/

Példák

E-mail küldés
Bemenet: levél adatai
Válasz: nyugta



Új alkalmazott felvétele
Bemenet: személyes adatok
Válasz: ID

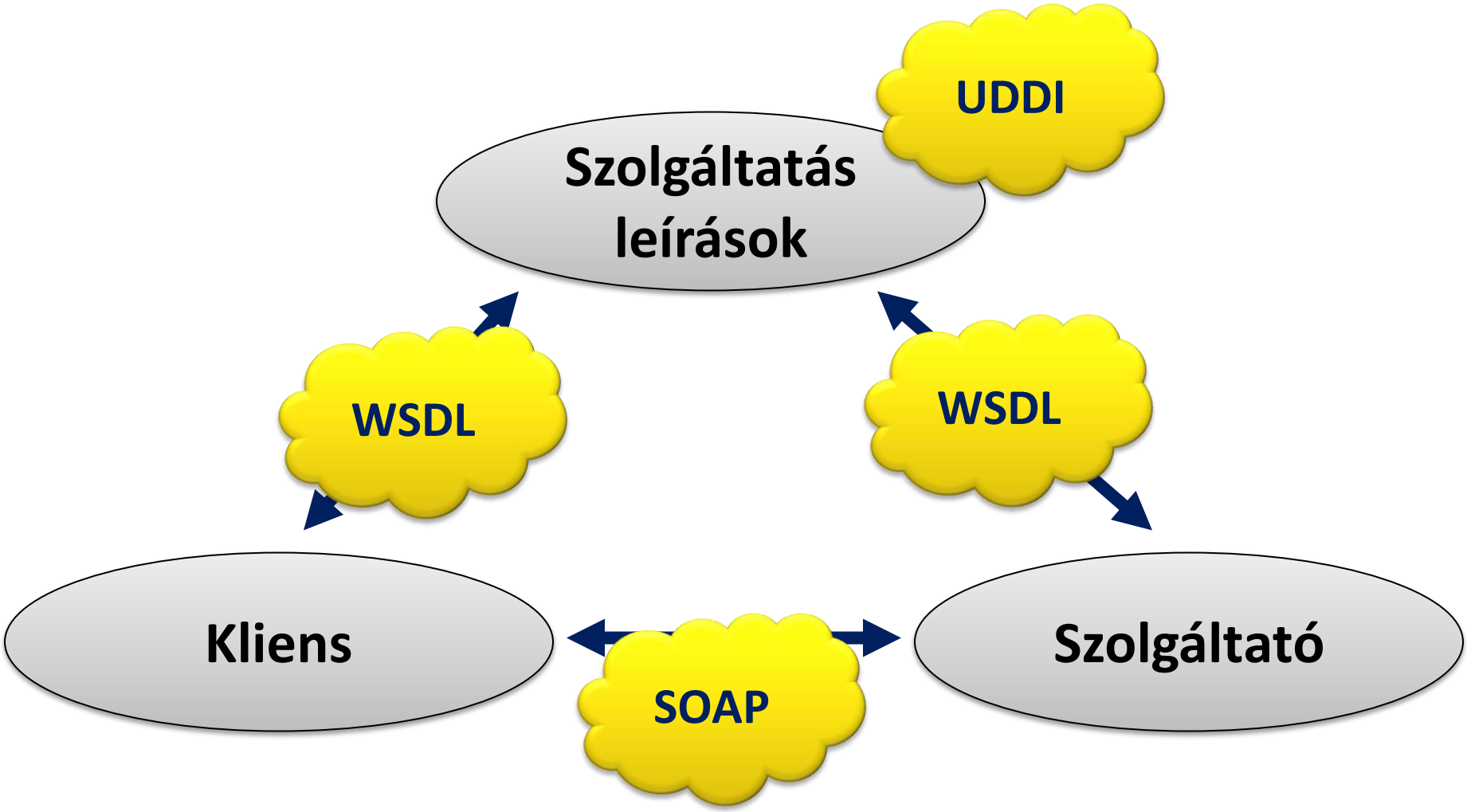
Vállalatirányítási rendszer elérése
Bemenet: beszállítók lekérdezése
Válasz: beszállítók listája, preferenciák

Infrastruktúra elérése
Bemenet: váratlan események lekérdezése
Válasz: riasztáslista

Lekérdező műveletek
Pl. időjárás előrejelzés, keresés,
repülőjegy árának lekérdezése, …
Bemenet: intervallum, hely
Válasz: csapadék, hőmérséklet, …

Szenzorok lekérdezése
Bemenet: épületrész állapota
Válasz: hőmérséklet, páratartalom,…

Megvalósítás

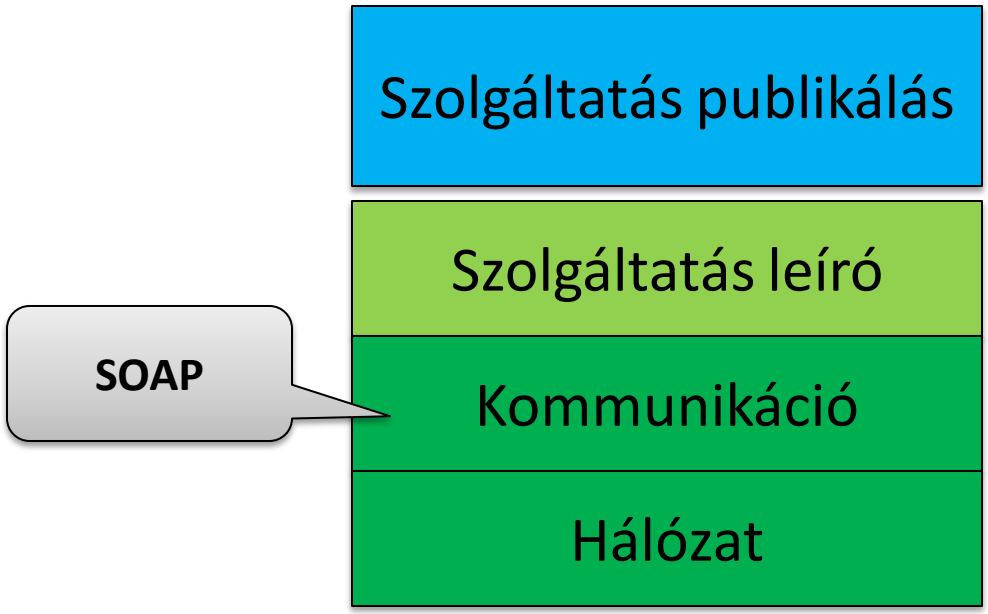


Web service stack

Alacsony szintű protokollok

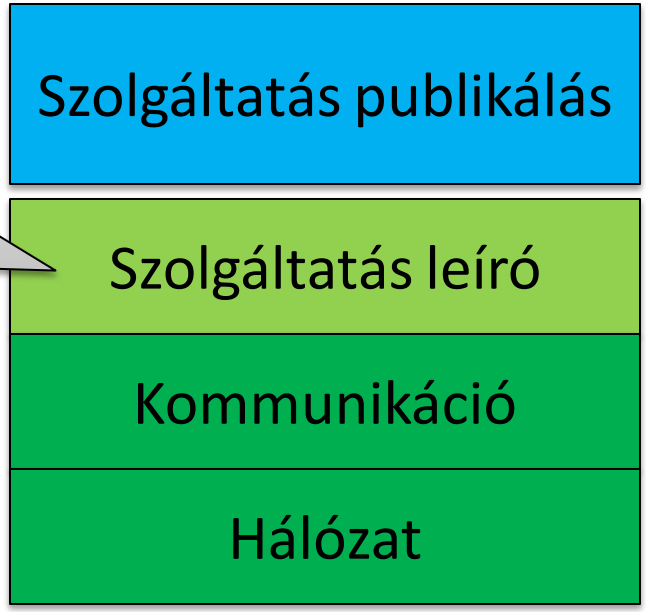
- **URI**, **HTTP**, **FTP**, etc.

Web service stack



Web service stack

Szintakszis HOGYAN?
(portok, műveletek, üzenetek)
- **WSDL**



Web service stack

regisztráció
HOL?
(„Arany oldalak” / hirdetők)
•**UDDI**, **WSIL**

SOAP

- Simple Object Access Protocol
  - 1.1 verzió: 2000 óta
  - Ma: leginkább 1.2
- Általános kommunikációs protokoll
- XML „Boríték”
- Fejléc
  - Címzett/feladó
  - Formátum
  - Kiegészítő információk
    - titkosítás, time-to-live, stb.
- Törzs
- http://www.w3schools.com/webservices/ws_soap_example.asp

MTOM

- Message Transmission Optimization Mechanism
- Bináris adatok átküldésének optimalizálása
- Különböző platformfüggő megoldások
  - Bináris adat külön elküldve
    - hogyan kapcsoljuk össze?
  - XML-binary Optimized Packaging használata (MIME)
    - Mi fölött küldjük át?
  - HTTP SOAP küldés specifikációja
    - Hogyan használjuk ehhez a HTTP-t?
- SOAP node szinten megadva

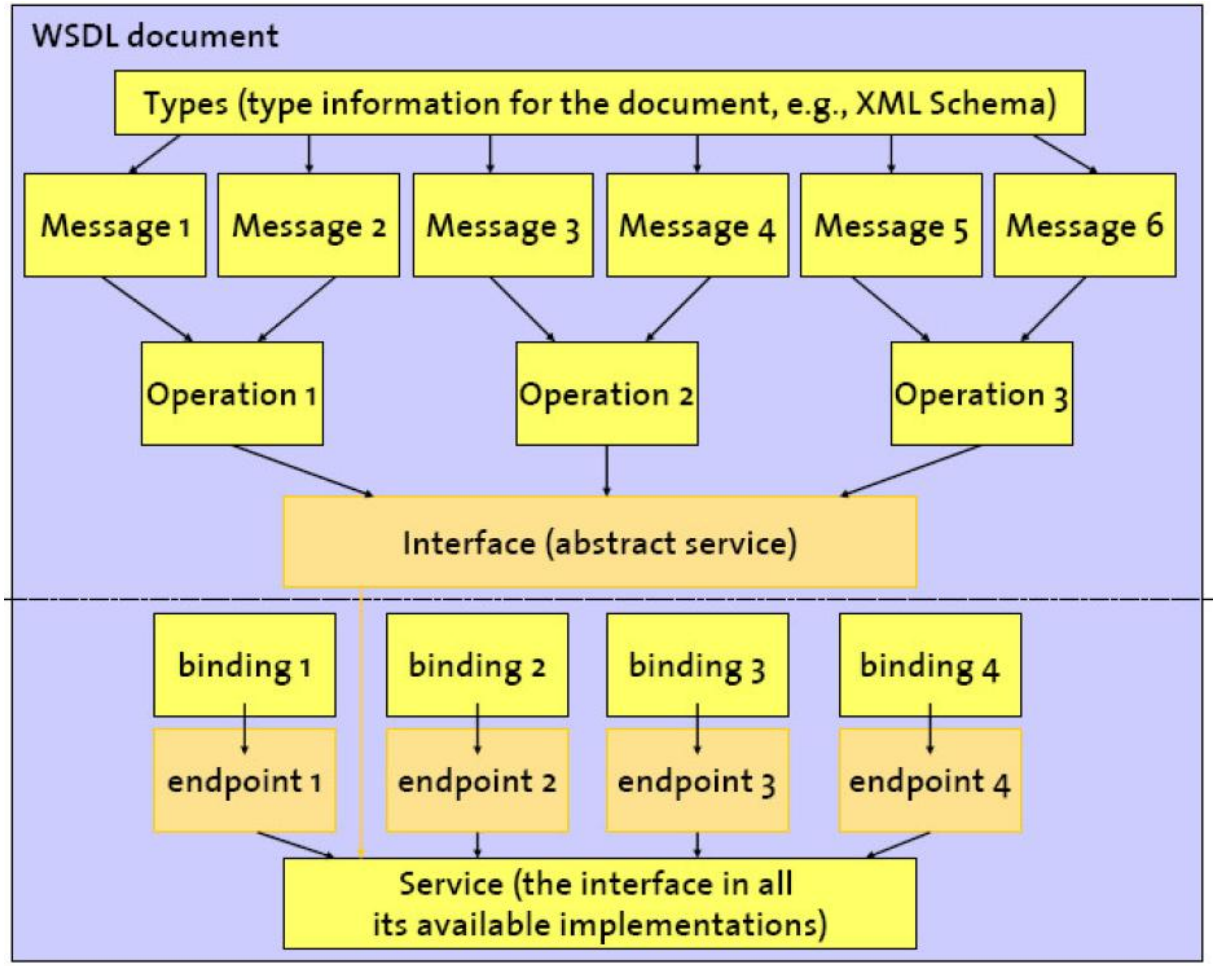
Web Services Description Language

- Interfészdefiníció
- Kérdés-válasz párok leírása
- „Port” típusok megadása
- Generálható az osztály publikus metódusai alapján
- Kliens (proxy) generálásához szükséges
- Lekérdezhető szolgáltatás tárakból

WSDL

- Adattípusok leírása
  - Séma (XSD) hivatkozás
- Portok leírása
  - Összetartozó műveletek halmaza
  - Pl. ÁrfolyamLekérdezőPort
- Műveletek (operations) leírása
  - Pl. UtolsóHónapÁtlagosÁrfolyama
- Üzenetek (típusának) deklarálása
  - Pl. ÁrfolyamKérdés
- Binding
  - Konkrét protokollhoz kötés, leggyakrabban SOAP

WSDL struktúra



Forrás: http://www.iks.inf.ethz.ch

Pl. http://tomi.vanek.sk/index.php?page=wsdl-viewer http://www.soapui.org/

Webszolgáltatás példák

- Példa
  - Google API
  - http://www.mnb.hu/arfolyamok.asmx?WSDL
- Pl. Amazon Elastic Compute Cloud
  - Maga a cloud management is webszolgáltatás alapon történik
  - http://aws.amazon.com/ec2/
  - https://s3.amazonaws.com/ec2-downloads/2013-02-01.ec2.wsdl
  - „Do NOT try to read or edit this file ” (SDK számtalan nyelvhez)

Webszolgáltatás példák 2

- Publikus webszolgáltatások
  - http://www.webservicex.net/WS/wscatlist.aspx
  - http://www.service-repository.com/
  - http://www.xmethods.net/ve2/Directory.po
  - https://www.thedacs.com/

Gyakorlati kérdések

- Hibakezelés
  - SOAP Fault
- Technológiai kötések
  - Java (Apache CXF, JAX-WS) → Metro stack
  - C# (.NET) (Windows Communication Framework)
  - PHP
  - COBOL
  - Python
  - R
  - Android kliens fejlesztés (pl. KSOAP2)

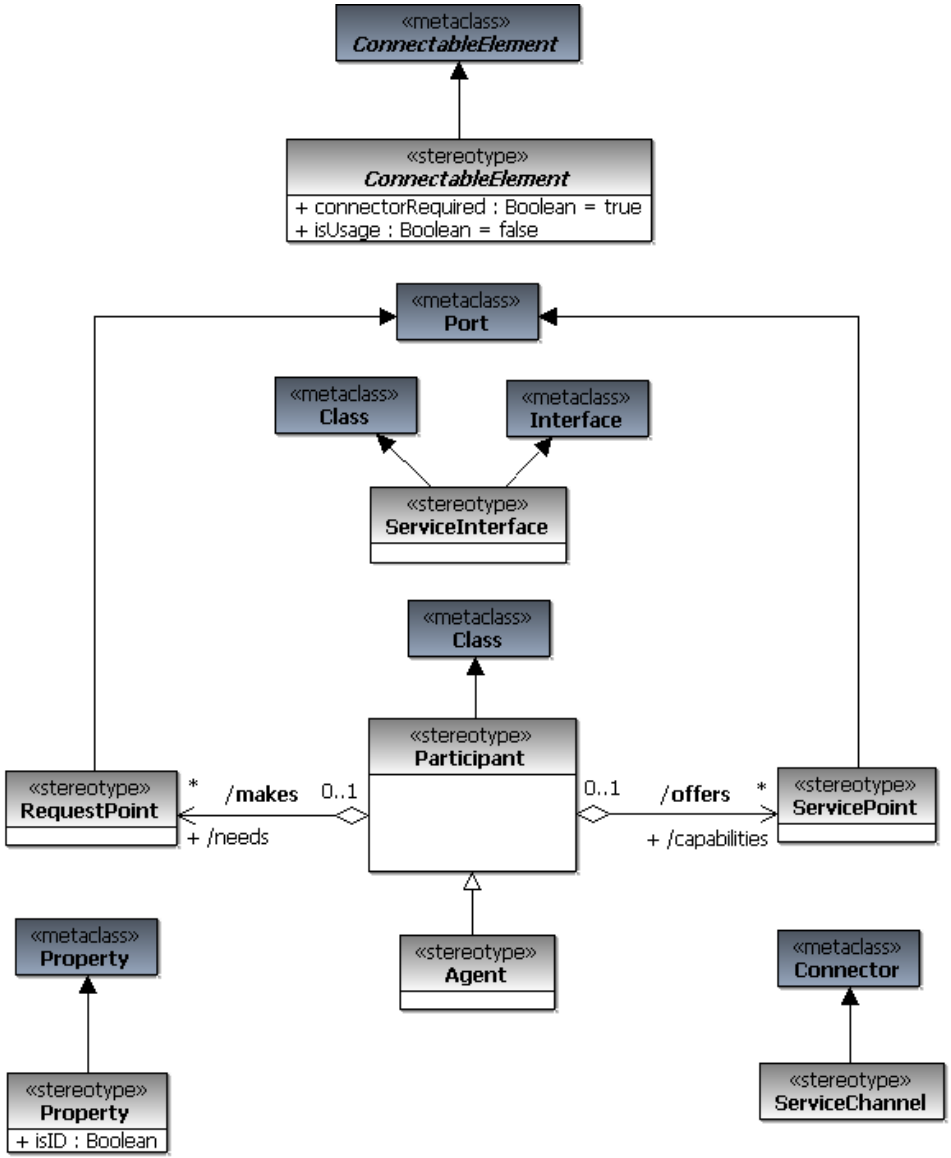
WS fejlesztés tipikus lépései

- Szolgáltatás interfészek, adatstruktúrák tervezése
  - WSDL, XSD
- Implementáció/integráció
  - Ezek alapján generálható a WSDL
- Kliens/szerver oldali csonkok előállítása
  - API hívással
  - XML konfiguráció alapján
- Futási időben (middleware)
  - SOAP üzenetek előállítása
  - SOAP boríték elküldése
  - SOAP üzenet transzformálása a szolgáltatás bemeneti formátumára

Webszolgáltatások vs RPC

- Adatmodell
  - Nem objektumorientált
- Adatátadás
  - Csak értékalapú (nem referencia)
  - Nagy adatmennyiségnél probléma
- Kommunikáció
  - Nincs köztes komponens QoS garanciákkal
  - Tűzfalproblémák könnyebben megoldhatóak
- „Beszélgetések” megvalósítása
  - Állapotmentes szerver

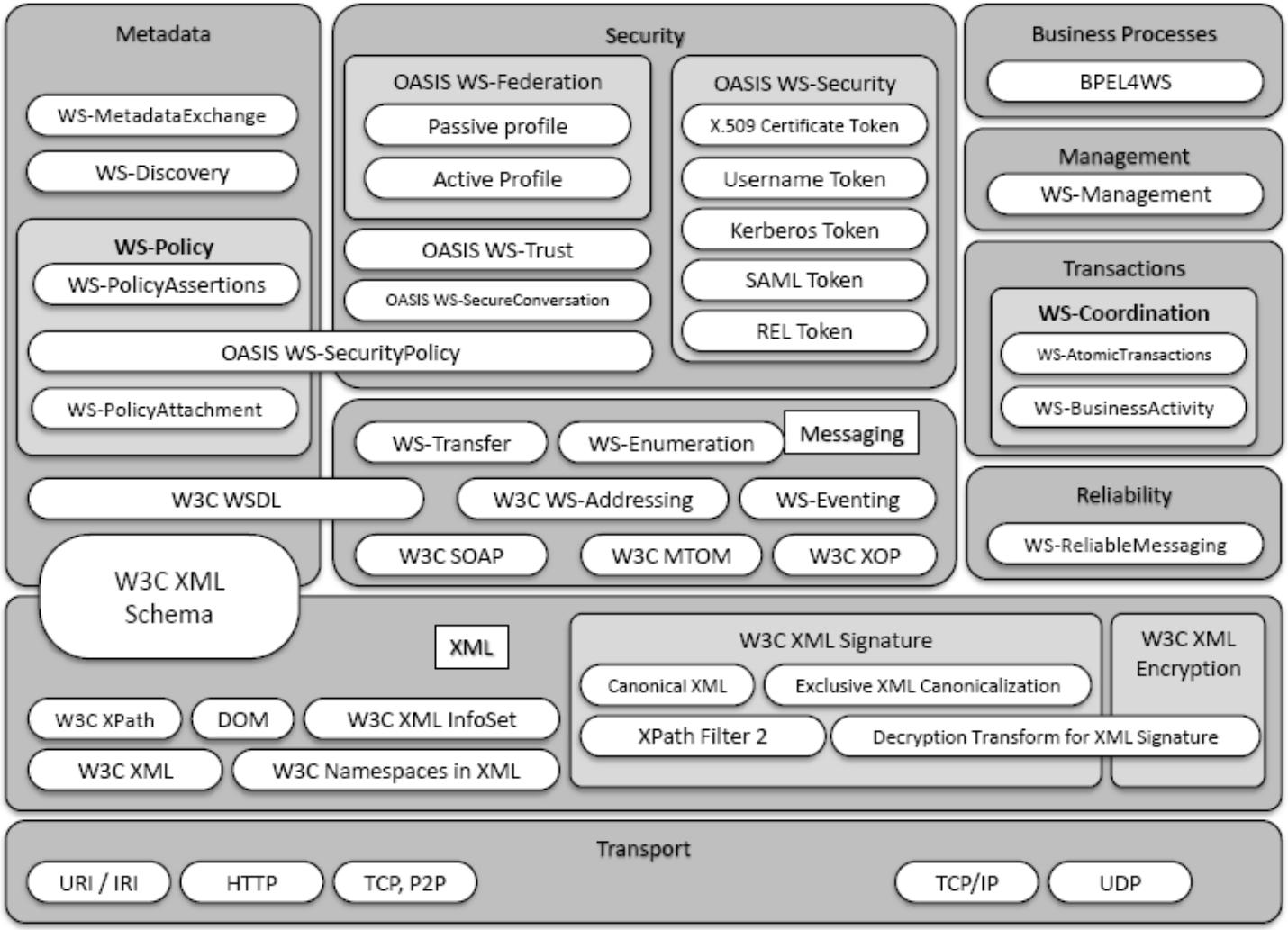
Modellezés: OMG: SoaML



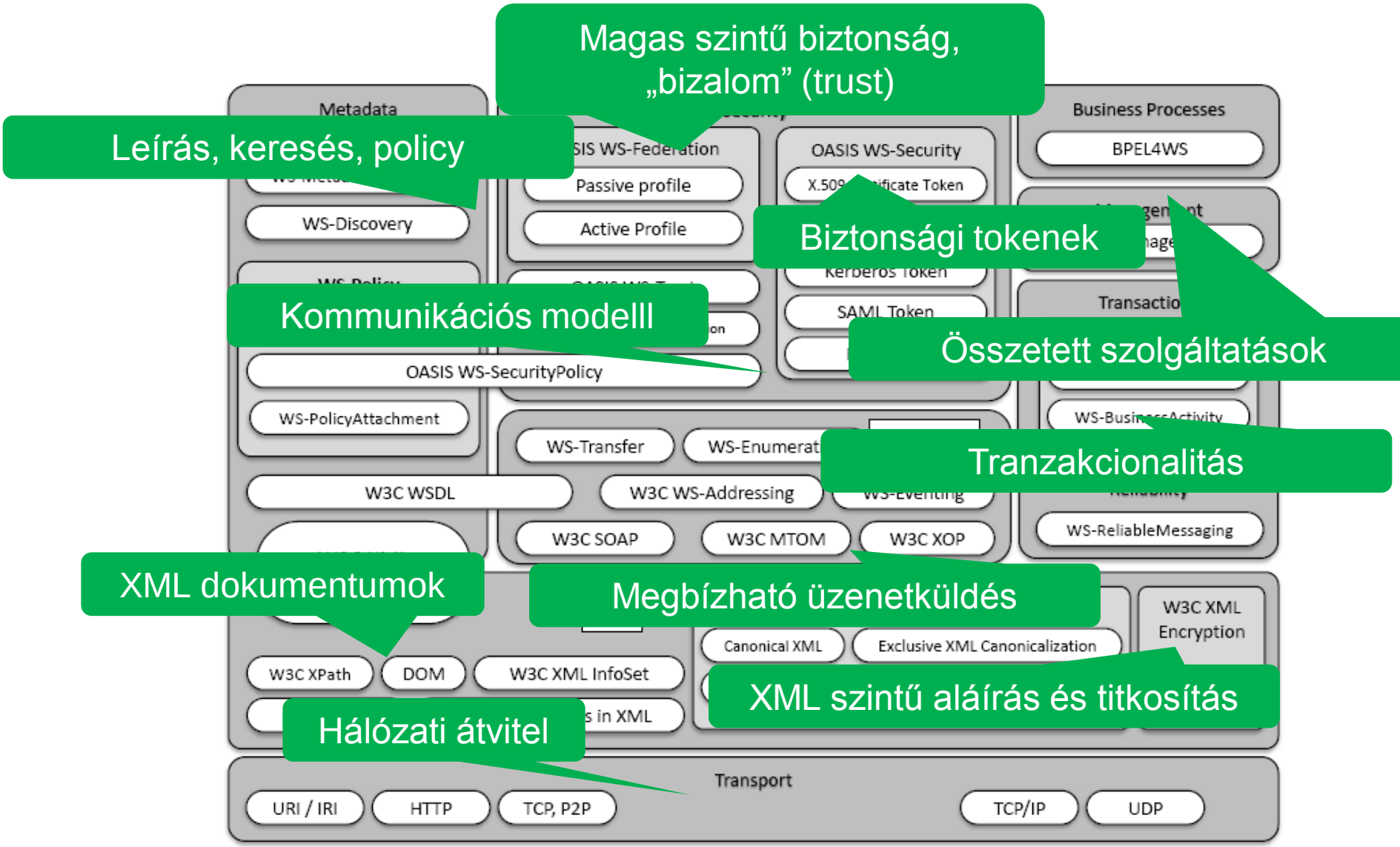
WS-*

- Hogyan adjunk alkalmazás szint garanciákat az integrációra?
  - Ne függjön a hálózattól
  - Ne függjön az implementációtól
  - Az alkalmazás logika határozza meg
- Feladatok
  - Tranzakciókezelés
  - Session kezelés
  - Biztonság
  - Titkosítás
  - Szolgáltatások kombinálása
  - Szolgáltatások szemantikája
  - Loggolás

Szabványok


Metadata
WS-MetadataExchange
WS-Discovery
WS-Policy
WS-PolicyAssertions
OASIS WS-SecurityPolicy
WS-PolicyAttachment
Security
OASIS WS-Federation
Passive profile
Active Profile
OASIS WS-Trust
OASIS WS-SecureConversation
OASIS WS-Security
X.509 Certificate Token
Username Token
Kerberos Token
SAML Token
REL Token
Business Processes
BPEL4WS
Management
WS-Management
Transactions
WS-Coordination
WS-AtomicTransactions
WS-BusinessActivity
Reliability
WS-ReliableMessaging
WS-Transfer
WS-Enumeration
Messaging
W3C WSDL
W3C WS-Addressing
WS-Eventing
W3C SOAP
W3C MTOM
W3C XOP
W3C XML Schema
XML
W3C XML Signature
Canonical XML
Exclusive XML Canonicalization
XPath Filter 2
Decryption Transform for XML Signature
W3C XML Encryption
W3C XPath
DOM
W3C XML InfoSet
W3C XML
W3C Namespaces in XML
Transport
URI / IRI
HTTP
TCP, P2P
TCP/IP
UDP

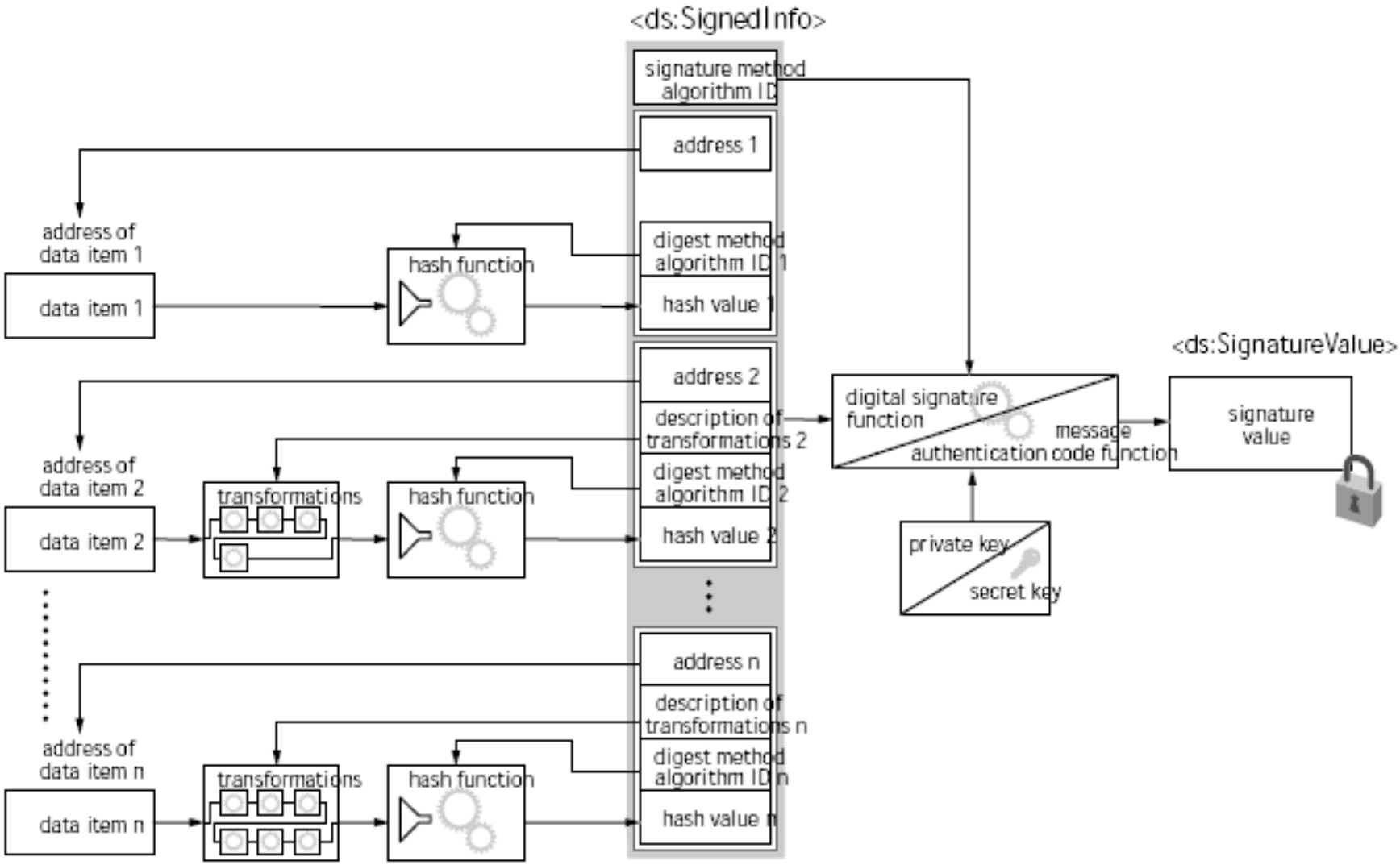
Szabványok


Magas szintű biztonság, „bizalom” (trust)
Leírás, keresés, policy
Biztonsági tokenek
Kommunikációs modelll
Összetett szolgáltatások
Tranzakcionalitás
XML dokumentumok
Megbízható üzenetkültdés
XML szintű aláírás és titkosítás
Hálózati átvitel
Metadata
WS-Discovery
OASIS WS-Security
Passive profile
Active Profile
Kerberos Token
SAML Token
OASIS WS-SecurityPolicy
WS-PolicyAttachment
Business Processes
BPEL4WS
WS-Transfer
W3C WSDL
W3C WS-Addressing
W3C SOAP
W3C MTOM
W3C XOP
WS-ReliableMessaging
W3C XML Encryption
Canonical XML
Exclusive XML Canonicalization
W3C XPath
DOM
W3C XML InfoSet
Transport
URI / IRI
HTTP
TCP, P2P
TCP/IP
UDP

XML encryption

- W3C szabvány
- Szimetrikus kulcsú (pl. 3DES)
- Aszimetrikus kulcs (RSA)
- Fő elemek
  - SignedInfo
  - EncryptedData
  - EncryptedKey
- Tipikusan XML signature-el együtt használják

XML Signature



WS-Security

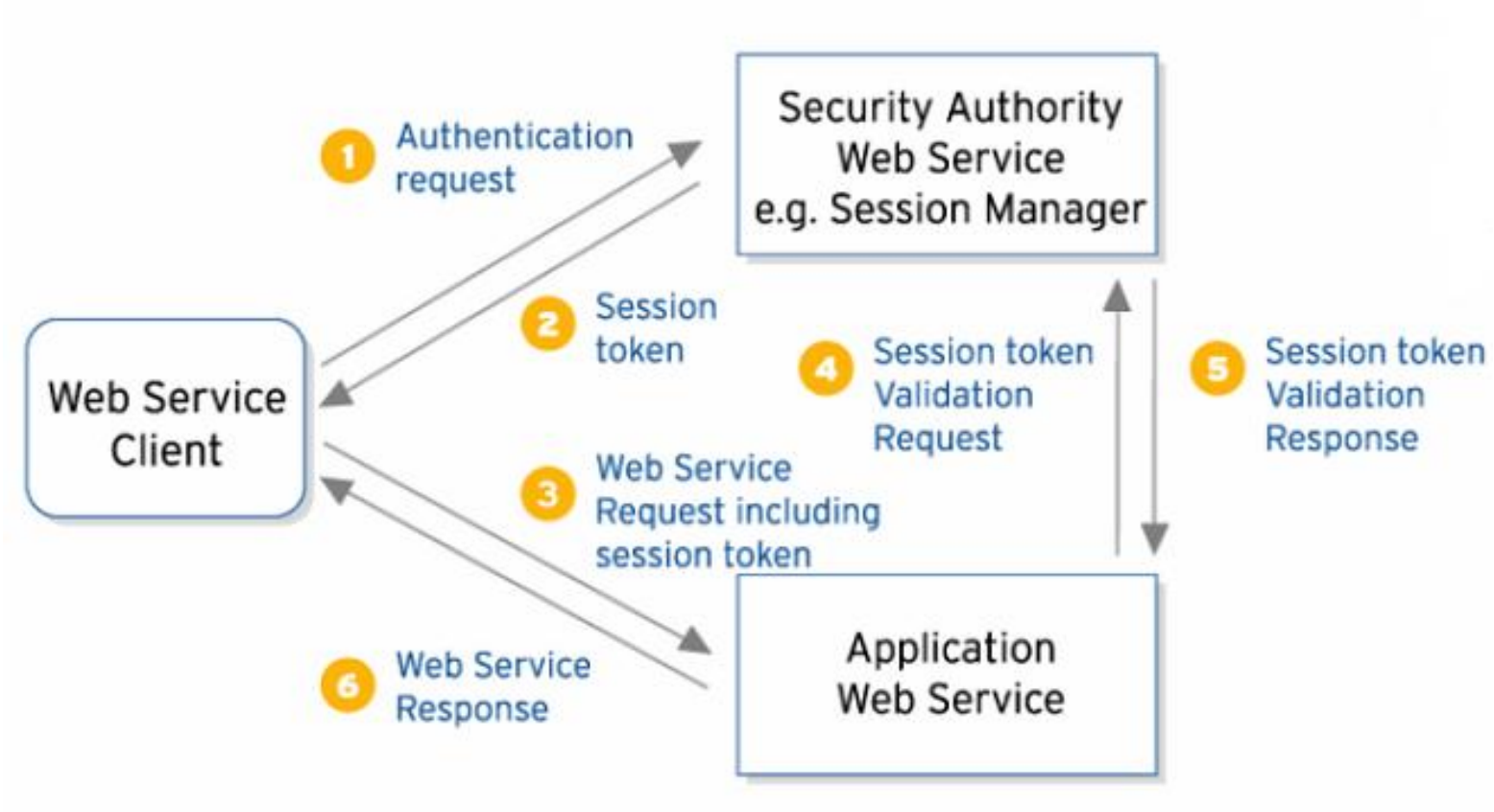
- Önmagában nem az üzenetet védi
  - „Hogyan igazoljuk, hogy védett az üzenet”
  - Használ más szabványokat
- Azonosítás és autentikáció
  - Milyen tokeneket használunk az üzenetben
- Integritás
  - XML signature
  - Időzítési védelem az újrajátszás ellen
- Titkosítás
  - WS-Encrpytion

Biztonsági tokenek

- Alkalmazás-specifikus
  - Usernév-jelszó
  - „unsigned”
- Aláírt biztonsági tokenek (bináris)
  - X.509 certificate
  - Kerberos
- XML tokenek
  - Pl. SAML
  - Általában „self-signed”

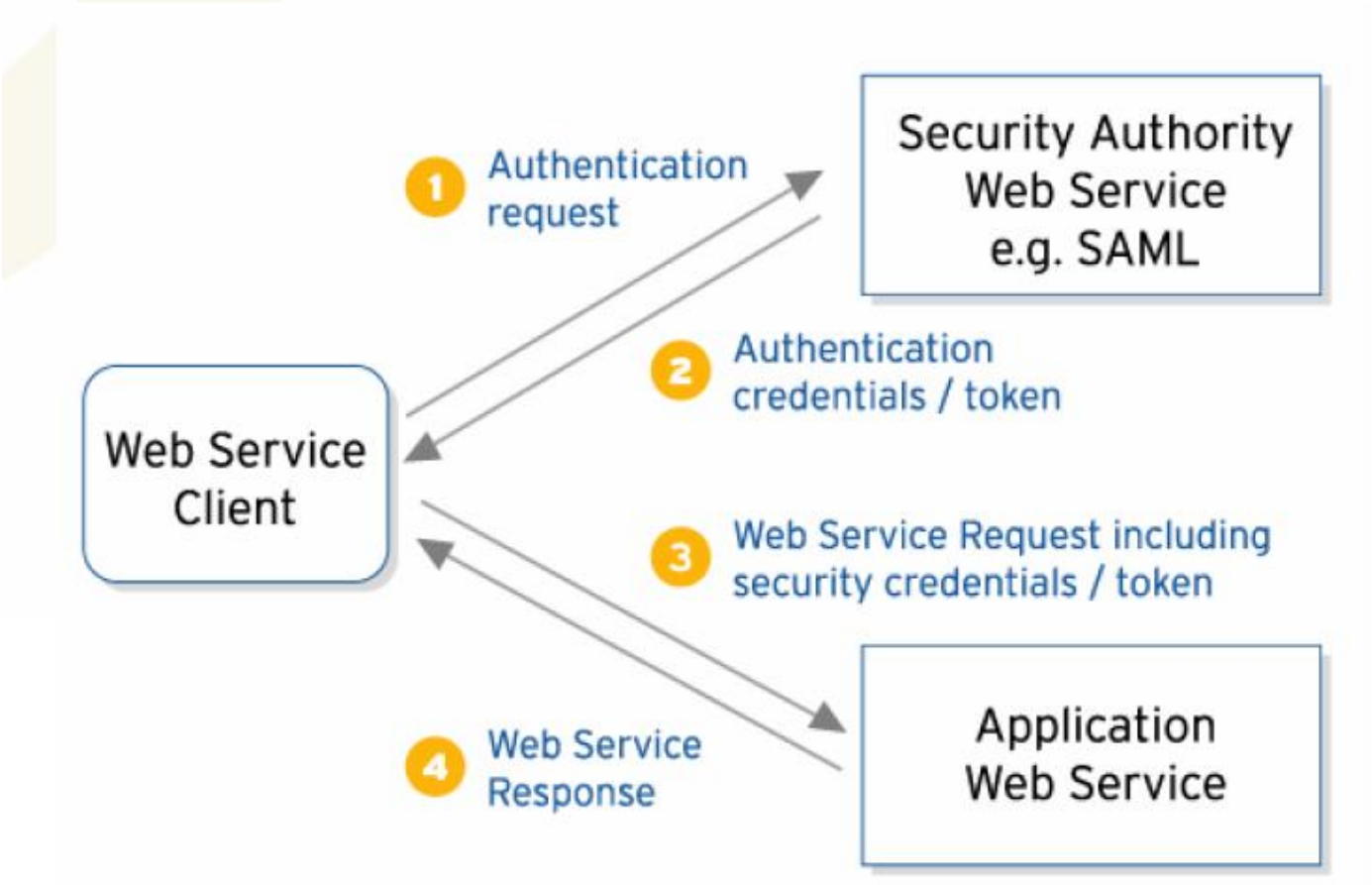
XML security párbeszéd

- Nem önhitelesítő tanúsítványok esetén (non self-validating credentials)



Szcenárió 2 – „self-validating credentials”

- Önhitelesítő tanúsítványok esetén (self-validating credentials)



WS-Security fejléc (SOAP)

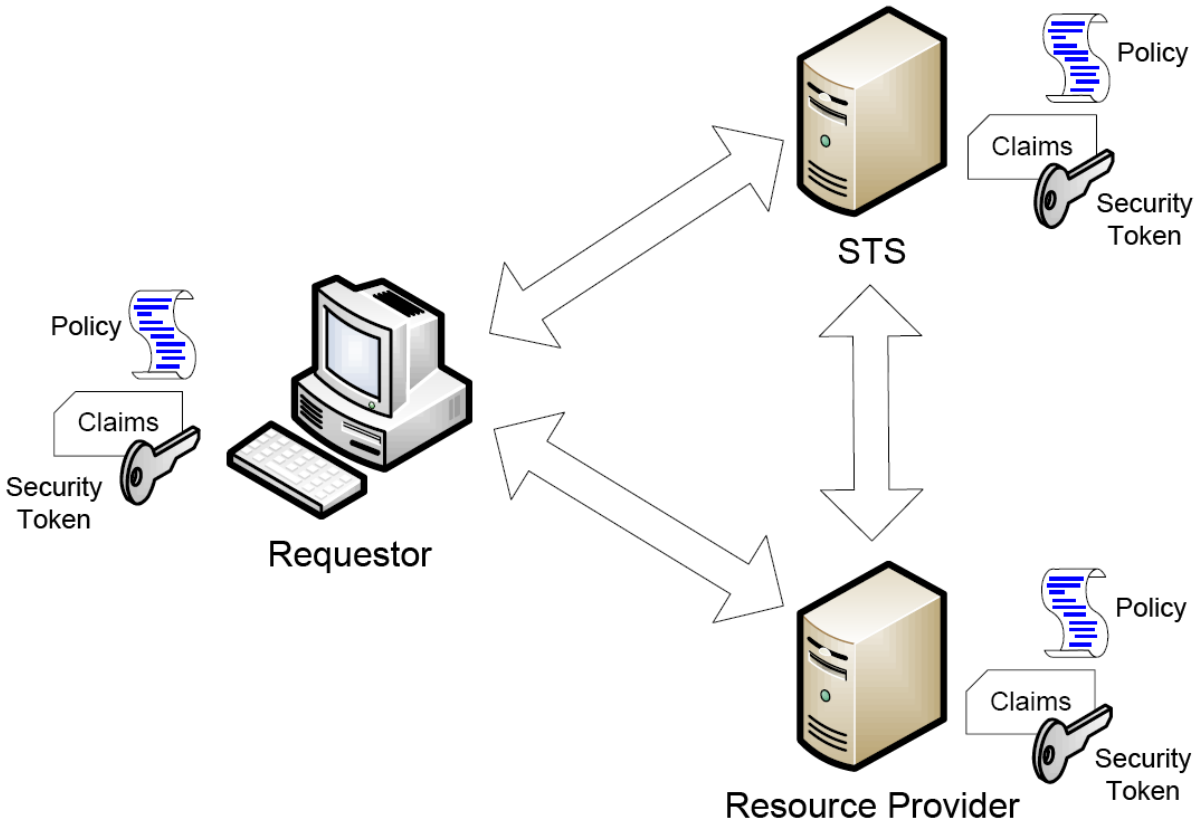
```
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope xmlns:soapenv=
"http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
<soapenv:Header>
        <wsse:Security xmlns:wsse="..." soapenv:mustUnderstand="1">
        <xenc:EncryptedKey Id="EncKeyId-229902">
                <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5" />
                <ds:KeyInfo xmlns:ds="...">
                        <wsse:SecurityTokenReference>…</wsse:SecurityTokenReference>
                        </ds:KeyInfo>
                        <xenc:CipherData>
                        <xenc:CipherValue>PpAOXj5P0W8ukm...</xenc:CipherValue>
                        </xenc:CipherData>
                <xenc:ReferenceList>
                <xenc:DataReference URI="#EncDataId-30957433" />
        </xenc:ReferenceList>
        </xenc:EncryptedKey>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"Id="Signature-17764792">
        <ds:SignedInfo>
                <ds:CanonicalizationMethod Algorithm=…/>
                <ds:SignatureMethod Algorithm=… />
                <ds:Transforms>…/ds:Transforms>
                <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
                <ds:DigestValue>+ECkM6R4GQ7AQ=...</ds:DigestValue>
        /ds:SignedInfo>
</ds:Signature>
<wsu:Timestamp../>
<soapenv:Body xmlns:wsu="..." wsu:Id="id-30957433">
        <xenc:EncryptedData Id="EncDataId-30957433”….>
</sopaenv:Body>
```

Titkosítatlan válasz

```
<?xml version='1.0' encoding='UTF-8'?>
   <soapenv:Envelope xmlns:soapenv=
   "http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Header />
<soapenv:Body>
   <resp:numberOfArcticles xmlns:resp=
   "http://daily-moon.com/cms/" xmlns:tns=
   "http://ws.apache.org/axis2">
   42</resp:numberOfArcticles>
</soapenv:Body>
</soapenv:Envelope>
```

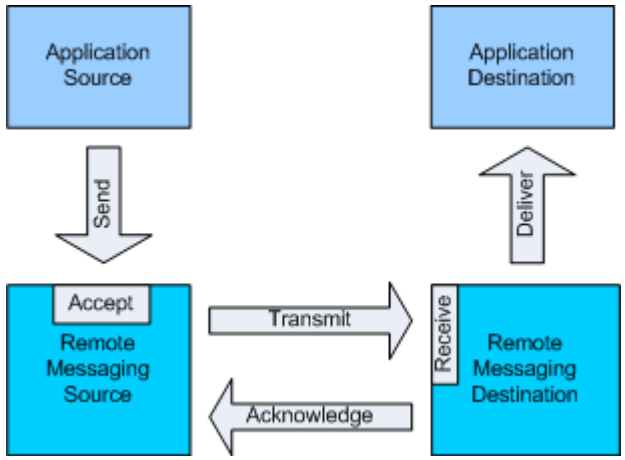
WS-Trust

- Szolgáltatók közti „bizalmasság”
- Biztonsági tokenek
  - Kibocsátása
  - Megújítása
  - Ellenőrzése
- Secure Token Service (STS)



Letagadathatatlanság: WS-RM

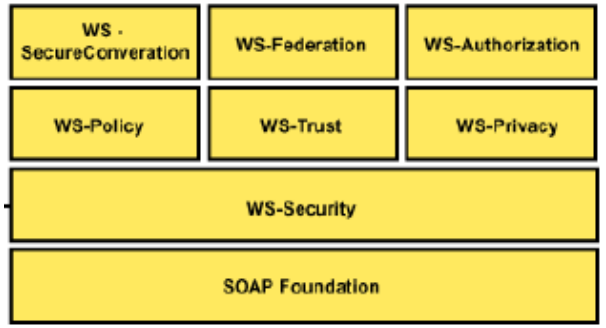
- „TCP réteg” webszolgáltatásoknak
- Megbízható üzenetküldés
  - Nyugtázás
  - Üzenetsorrendezés
  - Duplikátumok szűrése
  - Garantált kézbesítés



- Több szabvány egyesítése (MS, IBM)
- Implementációk
  - RAMP (IBM WebSphere Application Server)
  - Apache Sandesha (Axis2)
  - Microsoft Windows Communication Foundation
  - Bea WebLogic (→Oracle), Sun Glassfish (→ Oracle)

További specifikációk

- WS-SecurityPolicy
  - WS-policy alapon
  - Leírja a követelményeket
  - Milyen elemeket kell használni a többi nyelvből



- WS-SecureConversation
  - Hogyan történik a kulcsok igénylése, generálása, stb.
- WS-Policy
  - „assertion”: magas szintű állítás
  - „Legyen nyugtázott üzenetküldés”

WS-Interoperability

- WS-Interoperability
  - Különböző megvalósítások/eszközök közt
  - Oracle és MS kölcsönösen tesztelik egymás platformját
- Verziók pontos megadása
- Pl. WS-I Basic Profile 2.0 (2010)
  - SOAP 1.2, WSDL 1.1, UDDI 2.0, WS-Addressing,MTOM
- Létezik WSDL 2.0
  - Egyszerűsítések (pl. message nincs kiemelve)
  - Nem terjedt el
  - 4 részre osztva (types + interface / binding + service)

Milyen plusz feladatokat jelent?

- Magas szintű policy értelmezés
  - Pl. UM4SOA
- Pl. Apache Axis2 konfiguráció esetén
  - Szerver oldalon
    - Rampart, Sandesha modulok engedélyezése
    - services.xml konfiguráció beállítása
    - WSDL újragenerálása
    - Apache WSS4J
  - Kliens oldalon
    - Apache WSS4J
- Mögöttes infrastruktúra
  - Pl. Keystore, üzenetsorok
- Mi történik, ha a kliens nem tud bármit kezelni?
  - Pl. BPEL, Android… (erőforrások, konfiguráció)

WS Eszköztámogatás (példák)

- Apache: Apache Web Services Project
- IBM: WebSphere Application Server
- Microsoft: Windows Communication Foundation
- Oracle: METRO stack
- Eclipse: SOA Tools Platform
- Altova XML Tools
- Gyártók saját környezetei
  - Speciális célú, pl. adatbázis elérés
    - DB2
    - Oracle
    - MSSQL, stb.

Biztonsági analízis

- Szisztematikus támadás WSDL alapján
  - Publikus információ felhasználása
- „Brute force” támadás (XML parsing)
  - Túlterhelés: a parse-olás a szűk keresztmetszet
- „XML injection”
  - Magának a feldolgozási folyamatnak a megváltoztatása
    - Pl. XPath, XSLT, XQuery használatával
- Külső referencia támadás
  - Dokumentum linkelése
- SOAP protokol szintű támadás
- Szállító réteg támadása

Források

- http://ws.apache.org/axis2/modules/rampart/1_0/security-module.html
- http://ws.apache.org/sandesha/
- Sopera.de
- Security in a Web Services World: A Proposed Architecture and Roadmap (IBM & Microsoft whitepaper)
- Web Services Security Tutorial, Jorgen Thelin, CapeClear Software
- Standards and Practices in Operational Security, Yuri Demchenko, AIRG
- Understanding Web services Specifications –Part IV: security, Nicholas Chase (IBM whitepaper)
- http://www.slideshare.net/rmaclean/json-and-rest
- http://www.slideshare.net/PeterREgli/soap-wsdl-uddi#