Security Token Service

Valéry Tschopp - SWITCH

JRA1 All-Hands Meeting
NIKHEF, Amsterdam, 20-22 February 2008

www.eu-egee.org

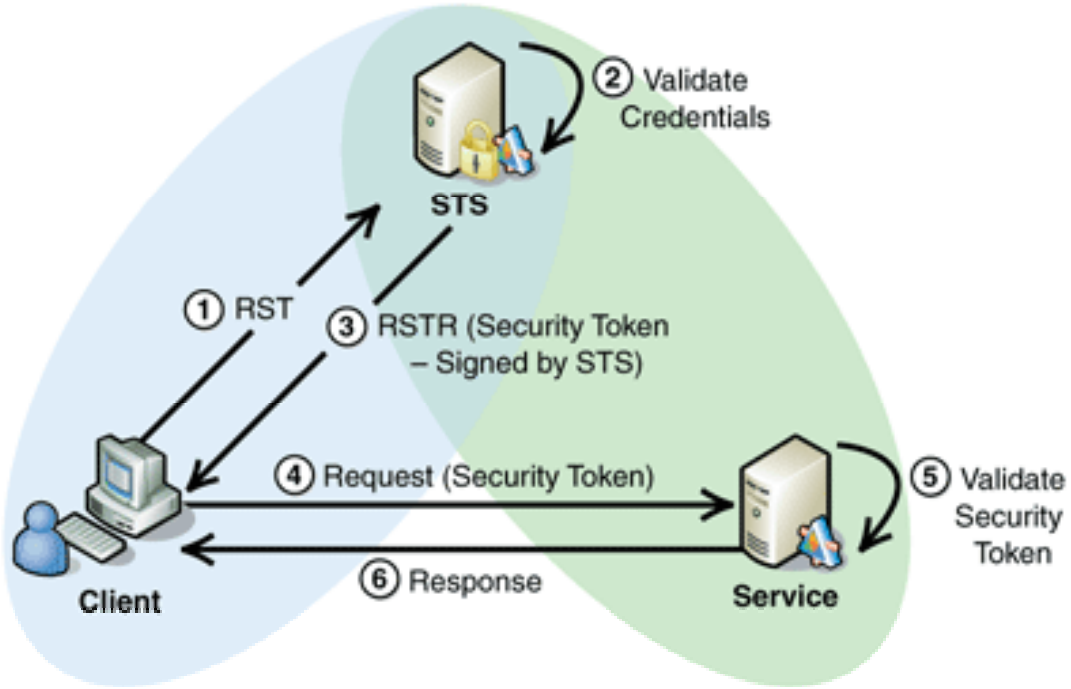
Information Society and Media

Web Service Standards

- **XML, SOAP, WSDL, HTTPS (SSL/TLS)**
- **WS-Security**
  - Security Tokens
    - Username token
    - X.509 Certificate
    - SAML
    - Kerberos
  - XML Encryption
    - Encrypt message content
  - XML Signature
    - Digital signature of message
- **WS-Trust**
  - OASIS Standard (WS-Trust 1.3)
  - Framework for requesting and issuing Security Tokens
  - Broker for trust relationship
  - It' also a point for enforcing token related policies (i.e. this token can be delegated by service A to service B)

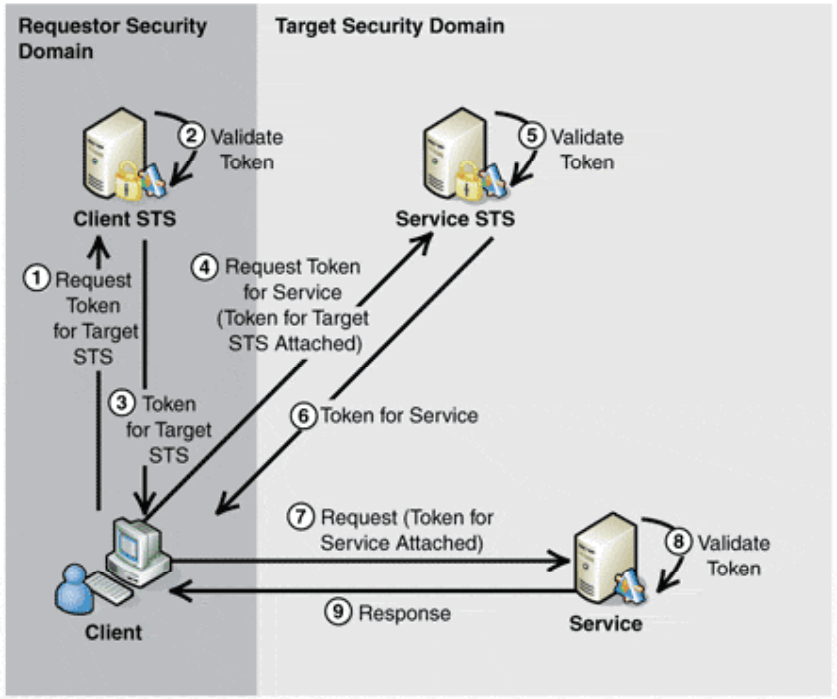
Security Token Service

- **WS-Trust defines mechanisms for brokering trust to an authority called Security Token Service (STS)**
- **The Security Token Service have a trust relationship with both the client and the service.**



Multiple Security Domains

- **A client may need to communicate with services that operate across trust boundaries**
  - Shibboleth SAML vs Grid X.509
- **Multiple STS can be used in a trust chain across security domains (delegated trust)**



Functionalities

- **Authenticates and authorizes users based on security tokens**
- **Transforms a security token (the claim) into another security token suitable for the requested service**
  - Username token into SAML token
  - SAML token into X.509 token
- **Aggregates required information from external Attribute Authorities**
- **Establishes a trust relation between different application domains**
  - Shibboleth domain vs Grid domain
- **Web Service (SOAP) based protocol**
  - Platform independent
  - Language independent (C, Java, Python, …)

Security Mechanisms

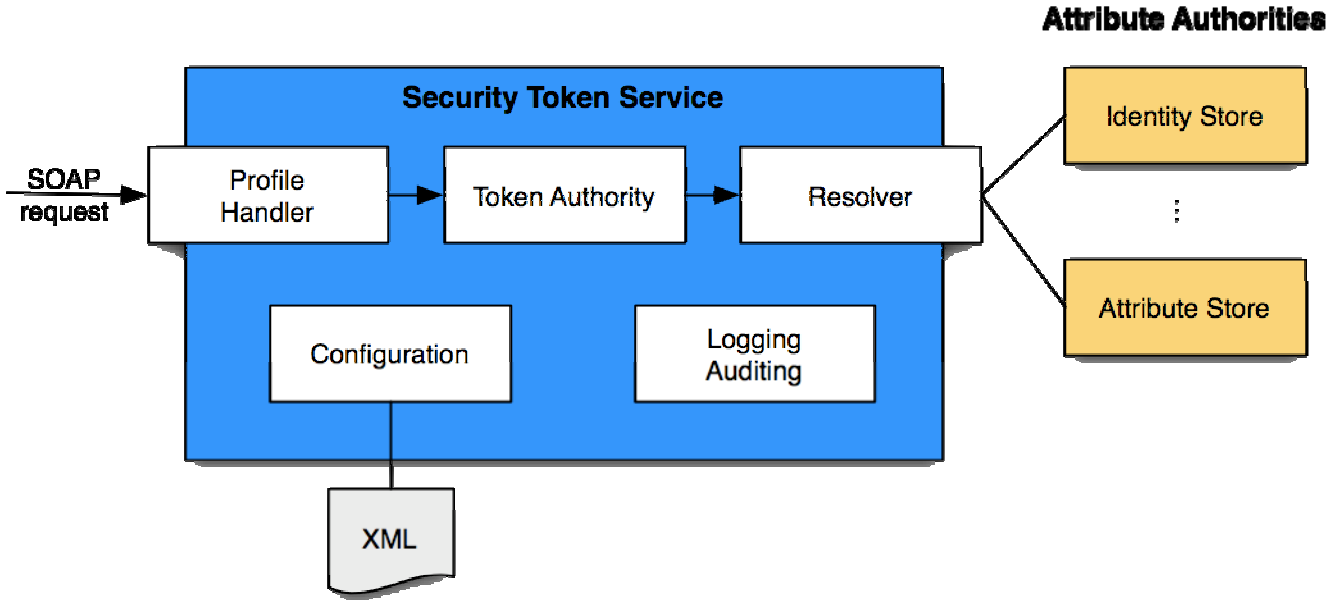
- **Authentication**
  - WS-Security
  - Transport provided mechanisms
- **Authorization**
  - WS-Policy
  - XACML
  - Application specific (ACL, …)
- **Message Integrity**
  - Transport SSL/TLS
  - XML Signature
- **Message Confidentiality**
  - Transport SSL/TLS
  - XML Encryption

Example RST Message

```
<wst:RequestSecurityToken xmlns:wst="..." xmlns:wsse="..." xmlns:wsu="..."
    wst:Context="Context-1202218308889">
    <wst:RequestType>
        http://docs.oasis-open.org/ws-sx/ws-trust/200512/Issue
    </wst:RequestType>
    <wst:TokenType>
        http://docs.oasis-open.org/wss/oasis-wss-saml-token-profile-1.1#SAMLV2.0
    </wst:TokenType>
    <wsu:Timestamp>
        <wsu:Created>2008-02-05T14:31:48.890Z</wsu:Created>
    </wsu:Timestamp>
    <wst:Claims>
        <wsse:UsernameToken wsu:Id="UsernameToken-1202218308889">
            <wsse:Username>myusername</wsse:Username>
            <wsse:Password wsse:Type="http://docs.oasis-open.org/wss/2004/01/oasis-
    200401-wss-wssecurity-secext-1.0.xsd#PasswordText">
                mypassword
            </wsse:Password>
            <wsu:Created>2008-02-05T14:31:48.889Z</wsu:Created>
        </wsse:UsernameToken>
    </wst:Claims>
</wst:RequestSecurityToken>
```

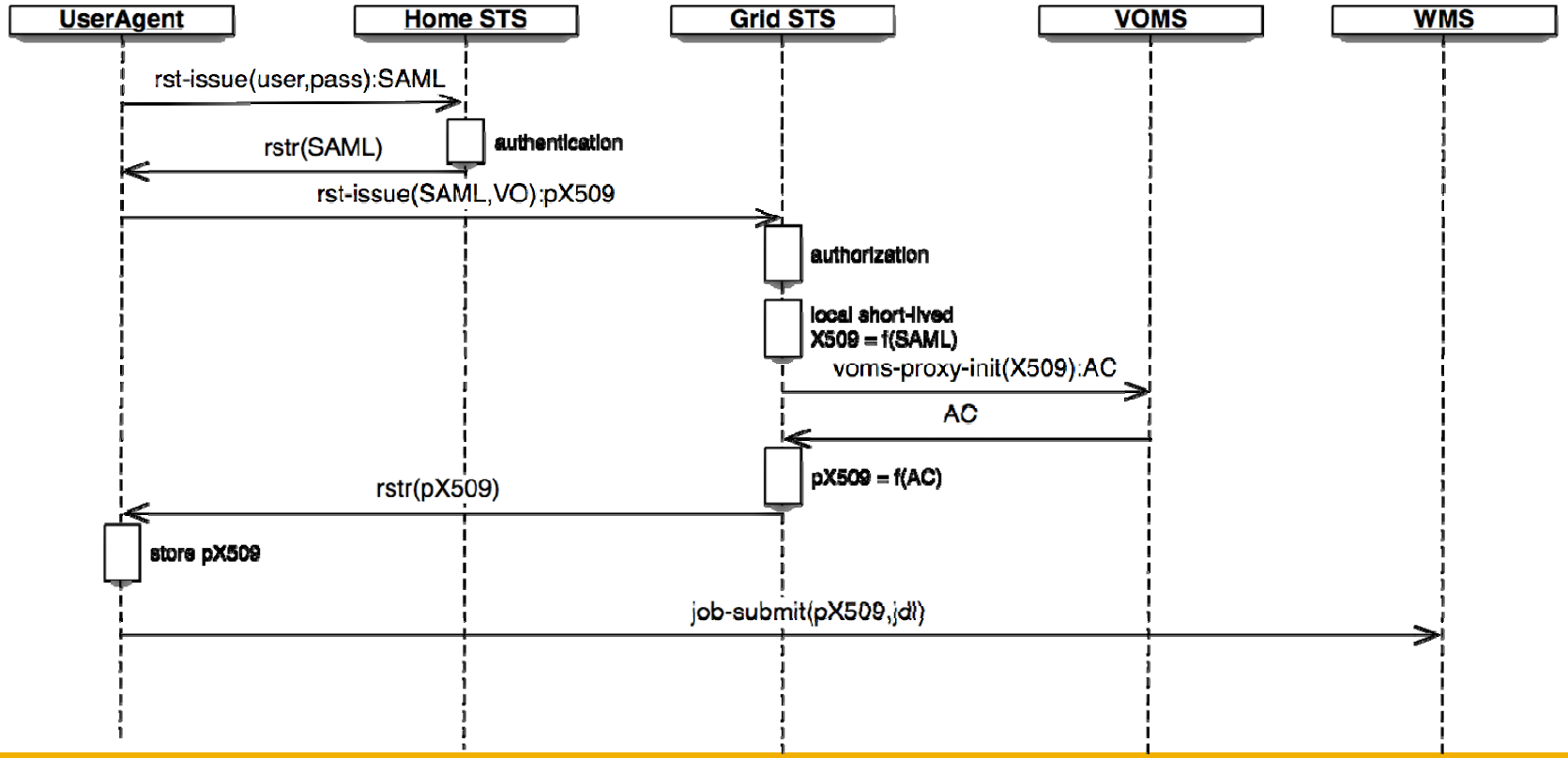
Architecture

- Profile Handler implements the WS-Trust profile
- Token Authority manages the security tokens
- Resolver retrieves information, attributes from external authorities (LDAP, DB, Online CA, VOMS, …)



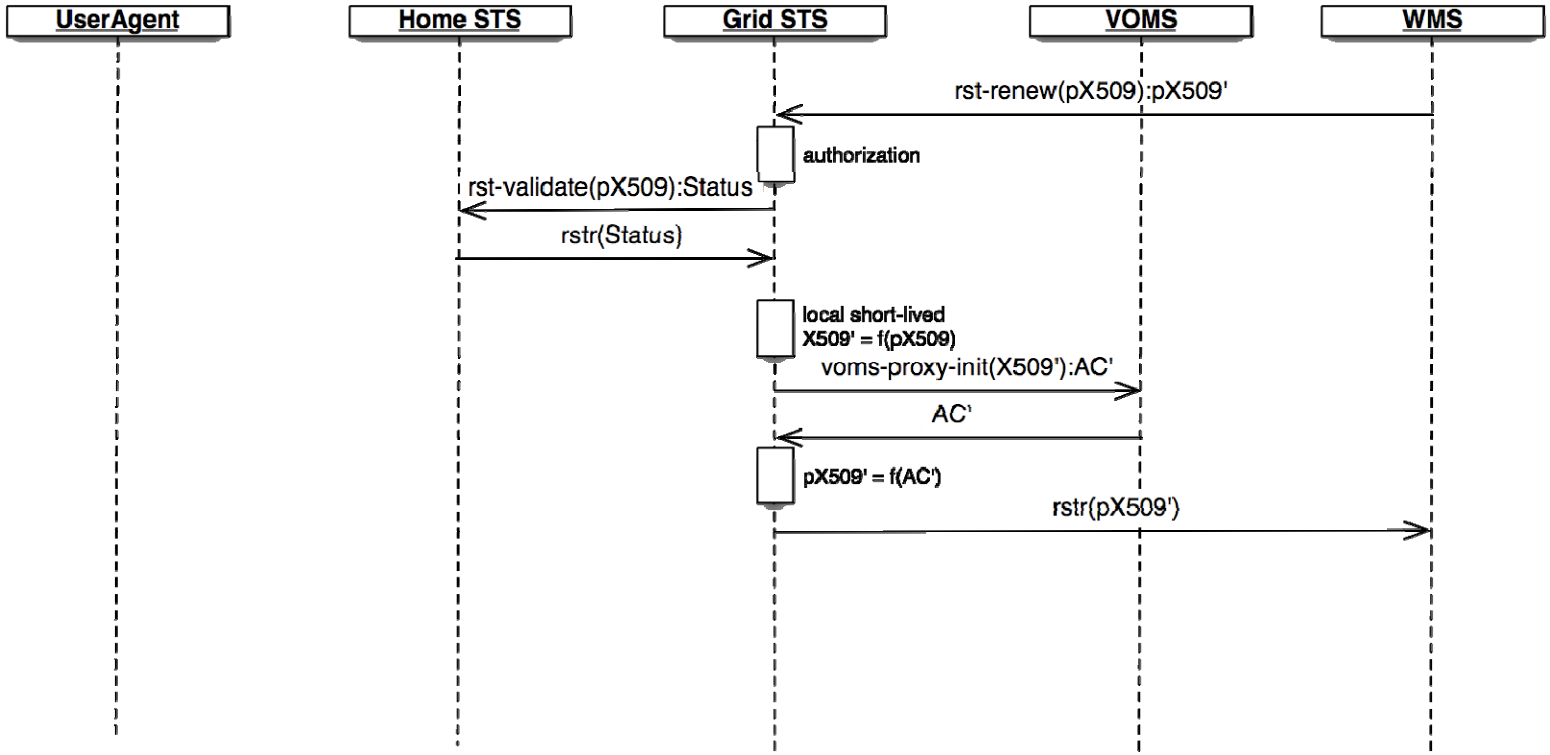
Scenario: Issue a proxy X.509

- User authenticates with his credential to a Shibboleth IdP STS and receives a SAML security token
- User requests a proxy X.509 from a Grid STS using the SAML token



Scenario: Renew a proxy X.509

- **At proxy X.509 expiration the WMS requests a proxy X.509 renewal from the Grid STS**



Status

- **WS-Trust 1.3 Profile document finalized**
- **The Security Token Service (STS) will be implemented within the Internet2 framework**
  - WS-Security implementation finished
  - WS-Trust implementation finished
  - STS profile handlers to be implemented
    - Transforms username token into SAML token
    - Transforms SAML token into X.509 token
- **Prototype should be available by the middle of March**

Comments and Questions

- **Comments ?**
- **Questions ?**