Web Service Security – Vulnerabilities and Threats in the Context of WS-Security

Jesper Holgersson
Eva Söderström
University of Skoevde, Sweden

Outline of presentation

- Research objectives
- Web Services
- Basic requirements for achieving information security
- Threats and challenges related to security in Web Services
- WS-Security basics
- WS-Security vs Threats
- Summary
Objective

- Security concerns are the main issues preventing organizations from adopting public Web Services (Greenspan, 2003)
- Security standards for Web Services are emerging, WS-Security is considered to be the most profound one yet
- How does WS-Security address known threats and weaknesses within Web Services?
Web Services

- “a technology for publishing, identifying and calling services in a network of interacting computer nodes” (Henkel & Wiktorin, 2005)

- Provider: The holder of the implemented service
- Requestor: The node that wants to use the service
- Registry: Is searched by the requestor and updated by the provider
Information security

Of particular interest for publicly exposed WS since failure in security might result in access to the WS-providers back end systems connected to the WS.

- Confidentiality
- Integrity
- Non-repudiation
- Authentication
- Authorization
- Availability

(Boncella, 2004)
Threats and challenges related to security in Web Services

- Maintaining security while routing between multiple Web Services
  - Confidentiality, Integrity, Authentication, Non-repudiation
- Unauthorized access
  - Authentication, Authorization
- Parameter manipulation/Malicious input
  - Availability, Integrity
- Network eavesdropping and message replay
  - Confidentiality, Integrity, Authentication, Non-repudiation
- Denial of Service
  - Availability
- Bypassing of firewalls
  - Confidentiality, Integrity, Authentication
Maintaining security while routing between multiple Web Services

- Traditional security techniques, such as SSL, are designed to protect communication between two points, i.e. security context 1
- Traditional security techniques can not handle end-to-end security, i.e. security context 2
- Traditional security techniques work at the session layer while SOAP works at the application layer
- A SOAP message has to be decrypted at the intermediary, thereby threatening confidentiality, integrity and authentication which all are related to authorization and non-repudiation

From O'Neill, 2002
Threats and challenges related to security in Web Services

- Maintaining security while routing between multiple Web Services
  - Confidentiality, Integrity, Authentication, Non-repudiation
- Unauthorized access
  - Authentication, Authorization
- Parameter manipulation/Malicious input
  - Availability, Integrity
- Network eavesdropping and message replay
  - Confidentiality, Integrity, Authentication, Non-repudiation
- Denial of Service
  - Availability
- Bypassing of firewalls
  - Confidentiality, Integrity, Authentication
WS-Security

- Set as a standard by OASIS in April, 2004
- Developed by IBM and Microsoft
- Works as an add-on to SOAP, offering a common format for security in SOAP messages
- Consists of three main elements
  - XML Encryption (W3C)
  - XML Signature (W3C)
  - Security tokens
WS-Security: an example

1. The requestor sends an order request
   1. A binary security token (X.509) is used for authentication
   2. The message is signed with one signature
   3. Customer information is encrypted with two different keys

2. Receiver 1 and 2 check whether the sender is legitimate, check the signature and decrypts those parts of the message that can be decrypted

3. Receiver 1 and 2 send a response in the same manner back to the requestor
WS-Security roadmap

- **WS-Policy**: Policy details about security issues
- **WS-Trust**: Establishing of trust between nodes
- **WS-Privacy**: Policies regarding privacy issues
- **WS-Secure Conversation**: Session spanning
- **WS-Federation**: Brokering of security related data
- **WS-Authorization**: How express and manage rules regarding access rights?
### Result

<table>
<thead>
<tr>
<th>Threat</th>
<th>Security requirements affected</th>
<th>Solved by WS-Security?</th>
<th>If so, by what?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintaining security while routing between multiple Web Services</td>
<td>Confidentiality, Integrity, Authentication, Non-repudiation</td>
<td>Yes</td>
<td>XML Encryption, XML Signature, Tokens</td>
</tr>
<tr>
<td>Unauthorized access</td>
<td>Authentication, Authorization</td>
<td>Yes</td>
<td>Tokens, XML Signature</td>
</tr>
<tr>
<td>Parameter manipulation and Malicious input</td>
<td>Availability, Integrity</td>
<td>Yes</td>
<td>XML Signature</td>
</tr>
<tr>
<td>Network eavesdropping and Message Replay</td>
<td>Confidentiality, Integrity, Authentication, Non-repudiation</td>
<td>Yes</td>
<td>Tokens, XML Encryption, XML Signature</td>
</tr>
<tr>
<td>Denial of Service</td>
<td>Availability</td>
<td>No</td>
<td>-</td>
</tr>
<tr>
<td>Bypassing of firewalls</td>
<td>Integrity, Authentication, Confidentiality</td>
<td>Indirectly</td>
<td>XML Encryption, XML Signature</td>
</tr>
</tbody>
</table>
Conclusions

- WS-Security handles the most urgent issues, i.e. secure transmission via intermediaries, thereby eliminating a number of related threats
- Much remains to be done
- WS-Security is still a young standard with little real life testing
- More mature technologies, such as SSL, has an immediate advantage as long as no intermediaries are involved
The end

- Questions?