Transitioning to message-level security on the SP–IdP back channel

Pros and cons

SWITCH
Serving Swiss Universities

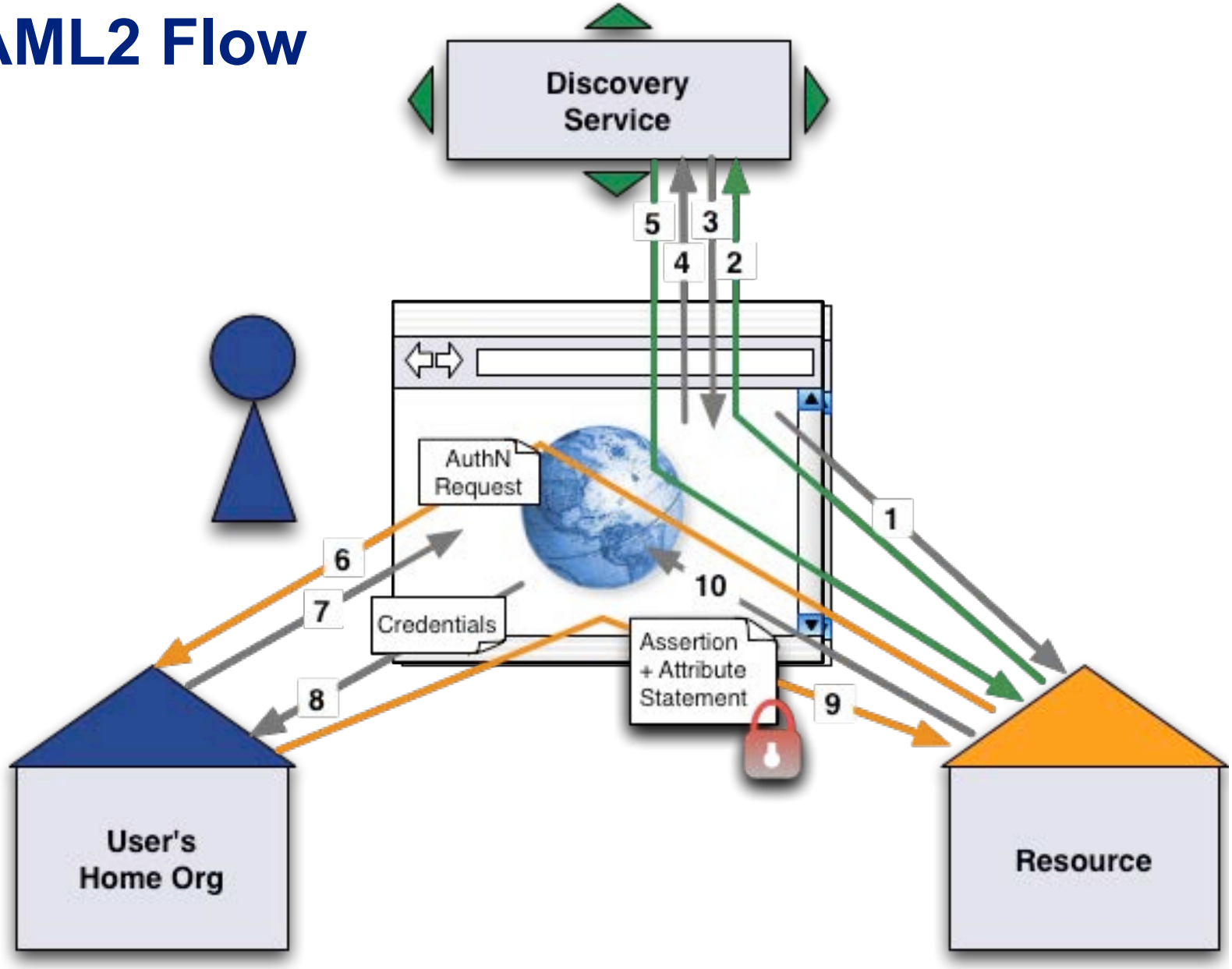
Kaspar Brand
kaspar.brand@switch.ch

Berne, 24 May 2011

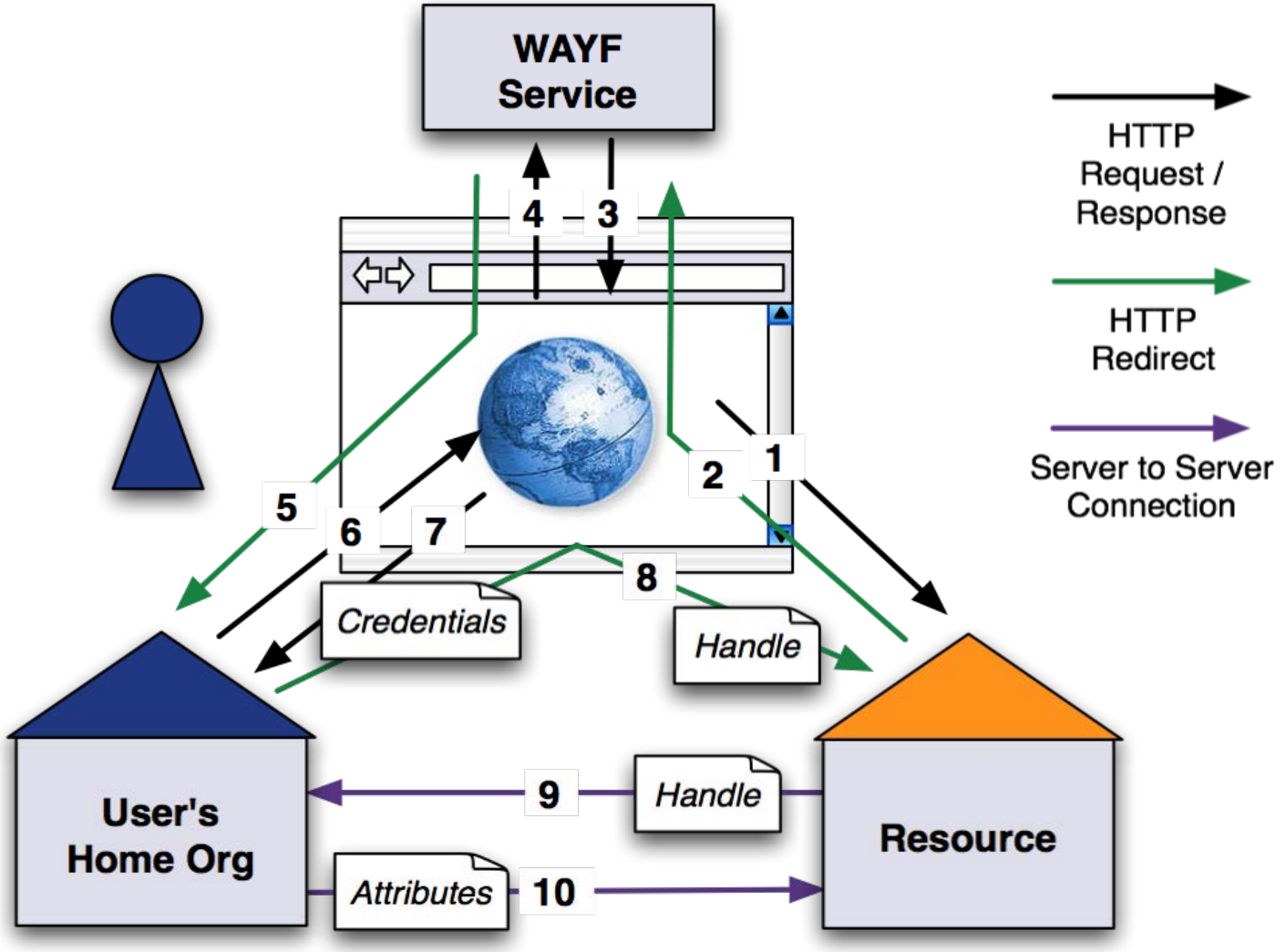
Transport layer vs. message-level security

- Transport layer security
  - in TCP/IP-based networks, typically provided by the SSL/TLS protocol (RFCs 2246/4346/5246)
  - adds confidentiality and data integrity to socket-based connections (“secures the pipe”)
- Message level security
  - in the SAML world, provided by the XML Signature and XML Encryption W3C recommendations
  - secures specific messages in a protocol
- Shibboleth favors transport layer security, in general
  - puts some burden on admins to properly configure SP–IdP back channel communication, though

SAML2 Flow



SAML1 Flow



Back channel still needs to be supported…

- for attribute queries
  - SAML1 service providers rely on back-channel attribute queries (no support for encryption on the front channel)
  - can occur with SAML2 as well, depending on the specific setup
- for artifact resolution requests
  - for both SAML1 and SAML2 peers
- possibly for single logout (IdP 3)

Transport layer security on the back channel

- IdP endpoint must support TLS client authentication
  - requires a separate port (or IP address), otherwise it would interfere with users' browser requests
  - is achieved either through the use of the **`SSLVerifyClient optional_no_ca`** kludge with mod_ssl or by adding a custom SSL connector to the Servlet container (**`tomcat6-dta-ssl`** or similar)
  - when using Apache in front of the IdP, assumes that you're running **`httpd`** and the Servlet container on the same system (otherwise say bye-bye to end-to-end security)
- using ports other than 443 are a possible causes for issues with packet filters / firewalls
- SP must support (and be configured for) TLS client authentication, too

Switching to message-level security

- for most communication flows, the default configurations of the current IdP and SP releases already support message-level security as an alternative to transport layer security
- changes required on the SP
  - configure the SP to always sign its back-channel requests
- changes required on the IdP
  - enforce that it unconditionally signs replies to attribute queries
  - configure attribute query / artifact resolution locations (URIs) to use the “browser-facing” TLS endpoint
- SP modifications have been added to the deployment guide in early April, IdP guide will be adapted with 2.3

SP changes in detail

- in **`shibboleth2.xml`**:
  adapt **`ApplicationDefaults`** (or **`ApplicationOverrides`**)

```
<ApplicationDefaults entityID="https://sp.example.org/shibboleth"
                     ...
                     signing="back" requireTransportAuth="false">
...
```

- best solution would be to add **`encryption="back"`** as well, but the IdP doesn't yet support encrypted IDs (should be in version 3.0, cf. https://issues.shibboleth.net/jira/browse/IDP-74)
- for Shibboleth 1.x service providers (which should be upgraded ASAP, anyway): in **`shibboleth.xml`**, adapt **`CredentialUse`**

```
<CredentialUse TLS="switchaai" Signing="switchaai"
               signRequest="true" />
```

IdP changes in detail

- in **`relying-party.xml`**: adapt profile configuration

```
<rp:DefaultRelyingParty provider="https://aai-logon.example.org/idp/shibboleth" ...>
    ...
    <rp:ProfileConfiguration
        xsi:type="saml:SAML2AttributeQueryProfile"
        signResponses="always" />
    ...
</rp:DefaultRelyingParty>
```

- in the metadata / resource registry: change artifact resolution and attribute service locations

```
<ArtifactResolutionService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
    Location="https://aai-logon.example.org/idp/profile/SAML2/SOAP/ArtifactResolution"
    index="2" />
...
<AttributeService Binding="urn:oasis:names:tc:SAML:2.0:bindings:SOAP"
    Location="https://aai-logon.example.org/idp/profile/SAML2/SOAP/AttributeQuery" />
```

[similarly for SAML1]

Changing an existing IdP deployment?

- aai-logon.switch.ch has been running its SOAP endpoints on port 443 since early April – no noteworthy issues observed so far
- analyze your Web server / Servlet container logs to determine how feasible a migration is… for the Common Log Format (CLF), something like

```
grep -hE 'AttributeQuery|ArtifactResolution' /path/to/your/log.files.* |
awk '{ print $7 }' | sort | uniq -c | sort -rn
```

  should give you a rough idea:

```
2859 /idp/profile/SAML1/SOAP/AttributeQuery
 871 /idp/profile/SAML2/SOAP/ArtifactResolution
 640 /idp/profile/SAML2/SOAP/AttributeQuery
   7 /idp/profile/SAML1/SOAP/ArtifactResolution
```

- based on that, figure out what SPs would be affected by the change (use `{ print $1, $7 }`) and talk to their admins

Summary: pros and cons

- one TLS endpoint at the IdP for everything
  - less hassle with Web server / Servlet container configuration
  - more straightforward setup wrt packet filters or firewalls
- puts message verification to its proper place
  - the application should decide, not the Servlet container or some TLS frontend
- XML Signature and XML Encryption are complex beasts
  - have become part of standard SAML2 message flows, meanwhile
  - can hopefully be considered mature by now
- might have performance impacts (on the IdP, in particular)
  - verification of signed requests, signing of attribute statements
- migrating an existing IdP deployment can be delicate