When To Use SOAP And When REST

Marek Potociar

Oracle

426

Comparing SOAP vs. REST

> SOAP

- Technology specification
  - Message format
  - Protocol bindings
  - Service Description
  - …

> REST

- Architectural style
  - Set of architectural constraints
  - Leveraging Web Standards
  - Defines expected system properties

When To Use SOAP And When REST Services

Web Service Use Cases, Interoperability And Programming Models

Marek Potociar

Oracle

JAZOON
INTERNATIONAL CONFERENCE ON THE
MODERN ART OF SOFTWARE, 21-23 JUNE 2011, ZURICH

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Service Discovery and Automation

Programming Models and Interoperability

> SOAP

- Interoperability: WSDL - W3C Standard
- Tooling: Code ↔ WSDL generation support
- Automation: Orchestrate with BPEL (OASIS Standard)
- Discovery: UDDI registries (OASIS Standard)

> REST

- Interoperability: WADL - W3C Submission
  - not standard, limited vendor support
  - Most RESTful services documented in human-readable text
- Tooling: WADL generation supported, no WADL consumption
- Automation: Resource orchestration using uniform interface (HTTP)
- Discovery: No Standard Available

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Public Cloud APIs

SOAP and REST

- **Scalability**
  - **REST** - Leverages Internet intermediary caching (HTTP Proxies)

    Uniform interface, Idempotent methods
  - **SOAP** - No automated caching possible

    HTTP POST used for tunneling all requests
- **Stateless communication**
  - **REST** – Stateless communication (by definition)
  - **SOAP** – SOAP over HTTP WS-I BasicProfile is stateless

    WS-* specifications often involve state management:

    WS-ReliableMessaging, WS-SecureConversation, WS-AtomicTransactions …

Public Cloud APIs

SOAP (JAX-WS) Amazon e-commerce client

```
wsimport http://ecs.amazonaws.com/AWSECommerceService/AWSECommerceService.wsdl
```

```
<service name="AWSECommerceService”>
  <port name="AWSECommerceServicePort” binding=“AWSECommerceServiceBinding">
    <soap:address location=
         “https://ecs.amazonaws.com/onca/soap?Service=AWSECommerceService” />
  </port>
</service>
```

```
AWSECommerceService service = new AWSECommerceService();
AWSECommerceServicePortType port = service.getAWSECommerceServicePort();
```

Public Cloud APIs

SOAP (JAX-WS) Amazon e-commerce client

```
<xs:complexType name="ItemSearchRequest">
  <xs:sequence>
    ...
    <xs:element name="Keywords" type="xs:string” minOccurs="0"/>
    <xs:element name="SearchIndex" type="xs:string” minOccurs="0"/>
    ...
  </xs:sequence>
</xs:complexType>
```

```
ItemSearchRequest request = new ItemSearchRequest();
request.setSearchIndex("Books");
request.setKeywords("Web Services");
```

Public Cloud APIs

SOAP (JAX-WS) Amazon e-commerce client

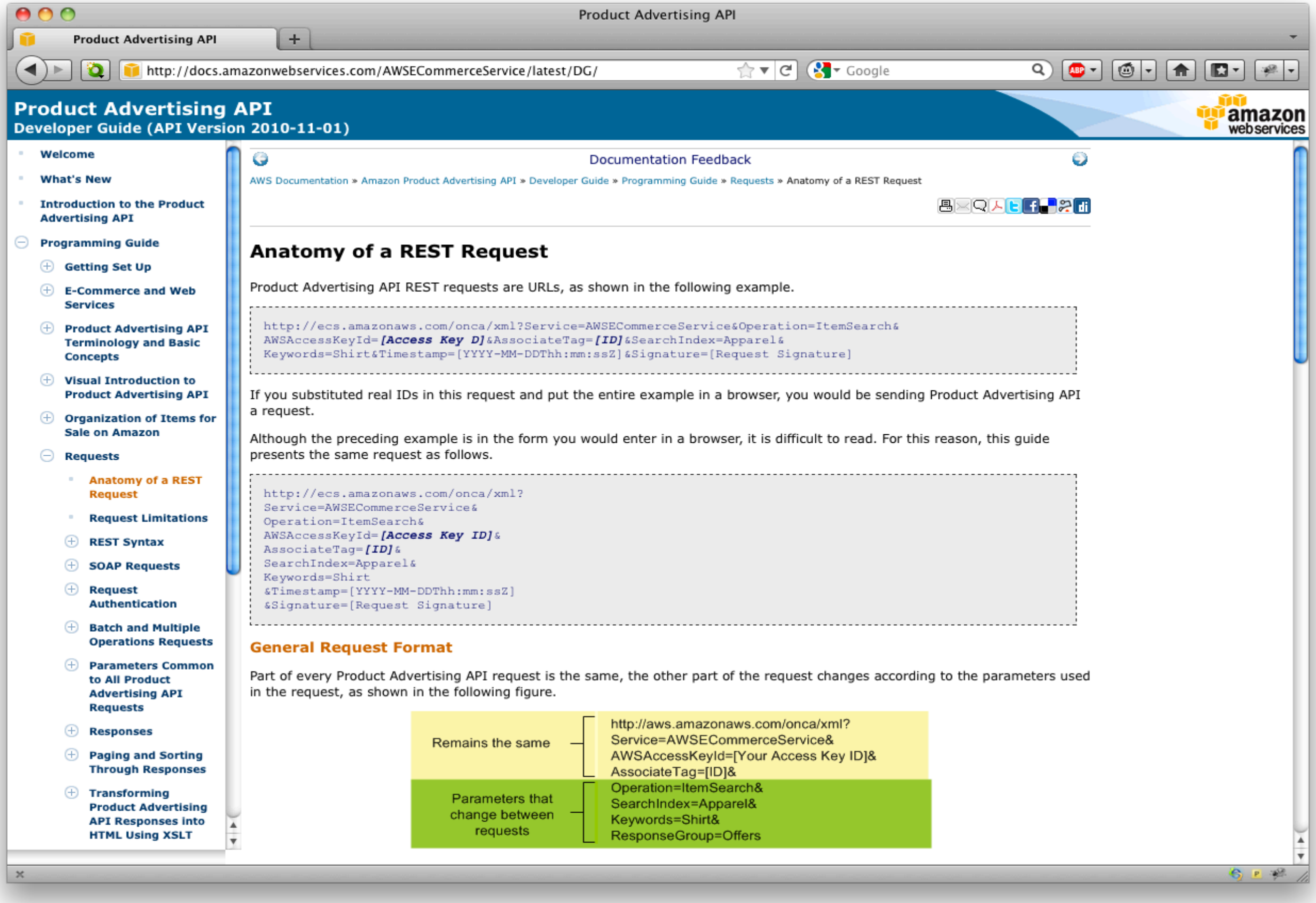
```
<portType name="AWSECommerceServicePortType">
  <operation name="ItemSearch">
    <input message="tns:ItemSearchRequestMsg"/>
    <output message="tns:ItemSearchResponseMsg"/>
  </operation>
</portType>
```

```
request.setSearchIndex("Books");
request.setKeywords("Web Services");

ItemSearchResponse result = port.itemSearch(request);
```

Public Cloud APIs

REST (Jersey) Amazon e-commerce client

http://docs.amazonwebservices.com/AWSECommerceService/latest/DG/


Product Advertising API
Product Advertising API
http://docs.amazonwebservices.com/AWSECommerceService/latest/DG/
Google
Product Advertising API
Developer Guide (API Version 2010-11-01)
amazon webservices
Welcome
What's New
Introduction to the Product Advertising API
Programming Guide
Getting Set Up
E-Commerce and Web Services
Product Advertising API Terminology and Basic Concepts
Visual Introduction to Product Advertising API
Organization of Items for Sale on Amazon
Requests
Anatomy of a REST Request
Request Limitations
REST Syntax
SOAP Requests
Request Authentication
Batch and Multiple Operations Requests
Parameters Common to All Product Advertising API Requests
Responses
Paging and Sorting Through Responses
Transforming Product Advertising API Responses into HTML Using XSLT
Documentation Feedback
AWS Documentation » Amazon Product Advertising API » Developer Guide » Programming Guide » Requests » Anatomy of a REST Request

Anatomy of a REST Request

Product Advertising API REST requests are URLs, as shown in the following example.

```
http://ecs.amazonaws.com/onca/xml?Service=AWSECommerceService&Operation=ItemSearch&
AWSAccessKeyId=[Access Key D]&AssociateTag=[ID]&SearchIndex=Apparel&
Keywords=Shirt&Timestamp=[YYYY-MM-DDThh:mm:ssZ]&Signature=[Request Signature]
```

If you substituted real IDs in this request and put the entire example in a browser, you would be sending Product Advertising API a request.

Although the preceding example is in the form you would enter in a browser, it is difficult to read. For this reason, this guide presents the same request as follows.

```
http://ecs.amazonaws.com/onca/xml?
Service=AWSECommerceService&
Operation=ItemSearch&
AWSAccessKeyId=[Access Key ID]&
AssociateTag=[ID]&
SearchIndex=Apparel&
Keywords=Shirt
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

General Request Format

Part of every Product Advertising API request is the same, the other part of the request changes according to the parameters used in the request, as shown in the following figure.

Remains the same

```
http://aws.amazonaws.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[Your Access Key ID]&
AssociateTag=[ID]&
```

Parameters that change between requests

```
Operation=ItemSearch&
SearchIndex=Apparel&
Keywords=Shirt&
ResponseGroup=Offers
```

Public Cloud APIs

REST (Jersey) Amazon e-commerce client

```
xjc -wsdl -d src/main/java http://ecs.amazonaws.com/AWSECommerceService/
   AWSECommerceService.wsdl
```

```
Client client = Client.create();
WebResource wr = client.resource("http://ecs.amazonaws.com/onca/xml");
wr.addFilter(new AWSCommerceClientFilter(awsAccessKey, awsSecureAccessKey));
ItemSearchResponse result = wr
      .queryParam("Service", "AWSECommerceService")
      .queryParam("Operation", "ItemSearch")
      .queryParam("SearchIndex", "Books")
      .queryParam("Keywords", "Web Services")
      .accept(MediaType.APPLICATION_XML).get(ItemSearchResponse.class);
```

Public Cloud APIs

REST (Jersey) Amazon e-commerce client

```
xjc -wsdl -d src/main/java http://ecs.amazonaws.com/AWSECommerceService/
   AWSECommerceService.wsdl
```

```
Client client = Client.create();
WebResource wr = client.resource("http://ecs.amazonaws.com/onca/xml");
wr.addFilter(new AWSCommerceClientFilter(awsAccessKey, awsSecureAccessKey));
ItemSearchResponse result = wr
      .queryParam("Service", "AWSECommerceService")
      .queryParam("Operation", "ItemSearch")
      .queryParam("SearchIndex", "Books")
      .queryParam("Keywords", "Web Services")
      .accept(MediaType.APPLICATION_XML).get(ItemSearchResponse.class);
```

Public Cloud APIs

Consuming Amazon e-commerce service response

```
for (Items items : result.getItems())
    for (Item i : items.getItem())
         System.out.println(i.getItemAttributes().getTitle());
```

```
Java Web Services: Up and Running
Restful Web Services
Sams Teach Yourself Web Services in 24 Hours
RESTful Web Services Cookbook: Solutions for Improving Scalability and
   Simplicity
Web Services Essentials (O'Reilly XML)
Programming .NET Web Services
...
```

Public Cloud APIs

SOAP (JAX-WS) bank account service

```
@WebService
public class Account {
  @WebMethod(operationName = "deposit")
  public String deposit(@WebParam(name = "amount”) final double amount) {
    ...
  }

  @WebMethod
  public Double withdraw(final double amount) throws OverBalanceException {
    ...
  }
}
```

Public Cloud APIs

REST (JAX-RS) bank account balance resource

```
@Path("accounts/{id}")
public class Account {
    @Path(“balance”) @GET @Produces(MediaType.APPLICATION_XML)
    public BalanceBean getBalance(@PathParam(“id”) String accountId) {
       return Accounts.get(accountId).getBalance();
    }

    @Path(“balance”) @PUT @Consumes(MediaType.APPLICATION_XML)
    public void putBalance(@PathParam(“id”) String accountId, BalanceBean b) {
       Accounts.get(accountId).setBalance(b);
    }
    ...
}
```

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Small Device Support

SOAP and REST

- Limited Memory
  - **REST** – Jersey server-side 1MB, client-side ~500kB
  - **SOAP** - Metro server+client ~10MB

- Limited Bandwidth & Performance
  - **REST** – Message media type Content-negotiation (client-controlled)

    Accept: image/jpeg; q=0.8, image/*; q=0.5
  - **SOAP** – MTOM (FastInfoset, XML-binary Optimized Packaging)

    Metro supports client-side initiated FastInfoset

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Service Mash-ups

Composing content from multiple sources

> REST

- Natural support for Hypermedia and Hyperlinking
- Client-driven media types
- JavaScript clients
- Easy to integrate resource queries into HTML

```
<IMG SRC=”http://maps.google.com/maps/api/staticmap?center=Moscone
    +Center,San
    +Francisco,CA&zoom=16&size=512x512&maptype=roadmap&sensor=false”/>
```

> SOAP

- No advantage

Service Mash-Ups

REST (JAX-RS) Example

```
@Path("events/{id}")
public class Event {
    @GET @Produces(“application/event+json”)
    public EventBean getEvent(@PathParam(“id”) String eventId) {
        return events.lookup(eventId);
    }

    @Path(“map")
    @GET @Produces(“text/html”)
    public String getMapHtml(@PathParam(“id”) String eventId) {
        EventBean e = events.lookup(eventId);
        return GoogleMapService.getGoogleMap(e.getAddress(), zoom);
    }
}
```

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Large Messages

Medical imaging, Multimedia content …

> **SOAP**

- MTOM/XOP – W3C Standard
- FastInfoset – ITU-T and ISO Standard
- Message streaming support (Metro)

  …without any intermediate data buffering

> **REST**

- Media types, HTTP Content-negotiation – Internet standard
- Natural support for binary media types in HTTP

  …MTOM not needed
- XML transfer optimization using FastInfoset (Jersey)

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Modeling Application State

Sessions, Shopping carts…

- SOAP
  - Metro – Stateful Web Services

    HTTP Cookies
  - WS-RM, WS-AT, WS-SC – provide stateful sessions
  - WS-Addressing – Sender/Recipient identification

- REST
  - Properly defined application URI spaces

    /carts

    /carts/marek.potociar

    /carts/marek.potociar/1

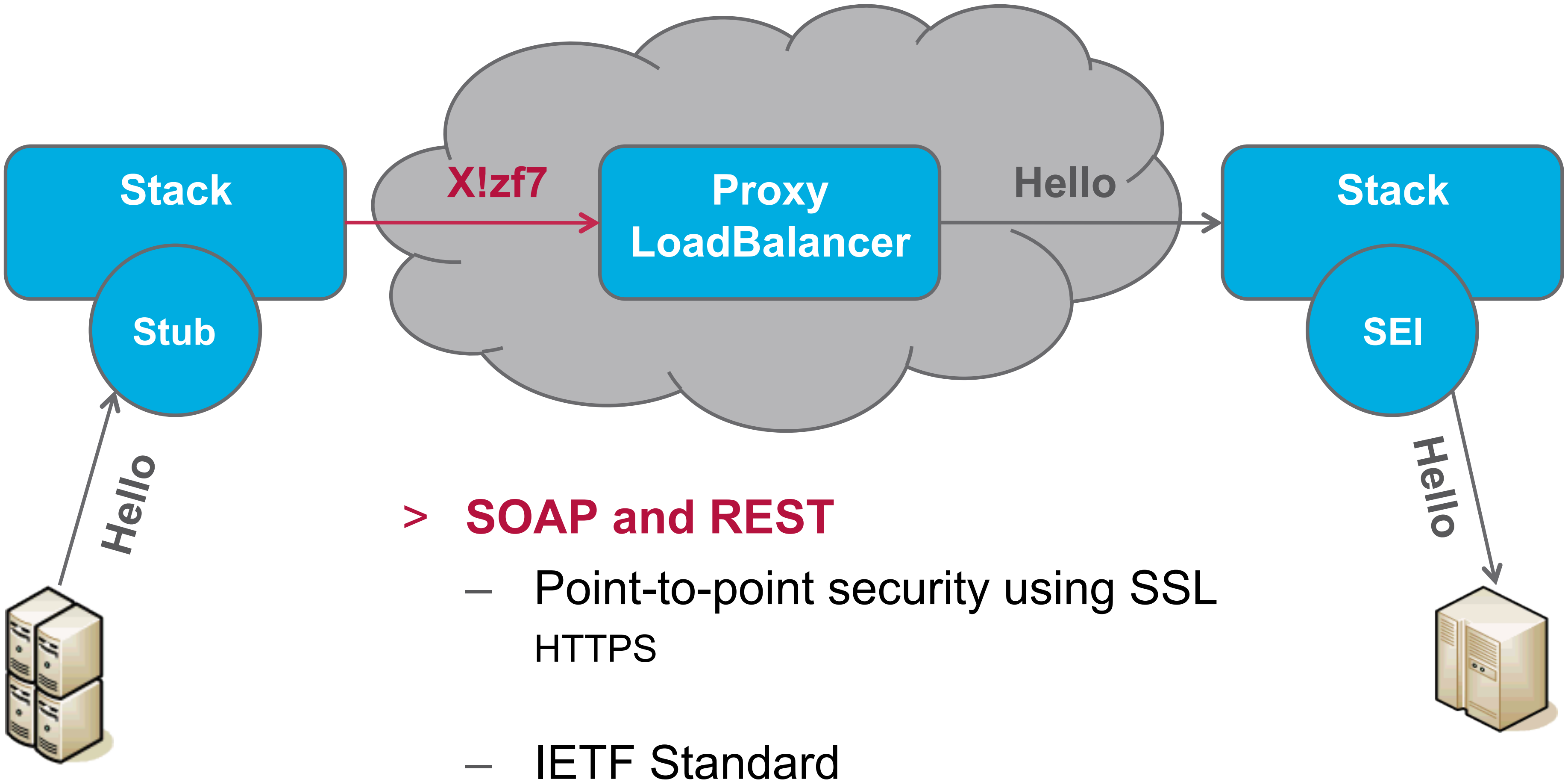
    /carts/marek.potociar/1/items

    /carts/marek.potociar/1/total-price

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

Point-to-point security



> **SOAP and REST**

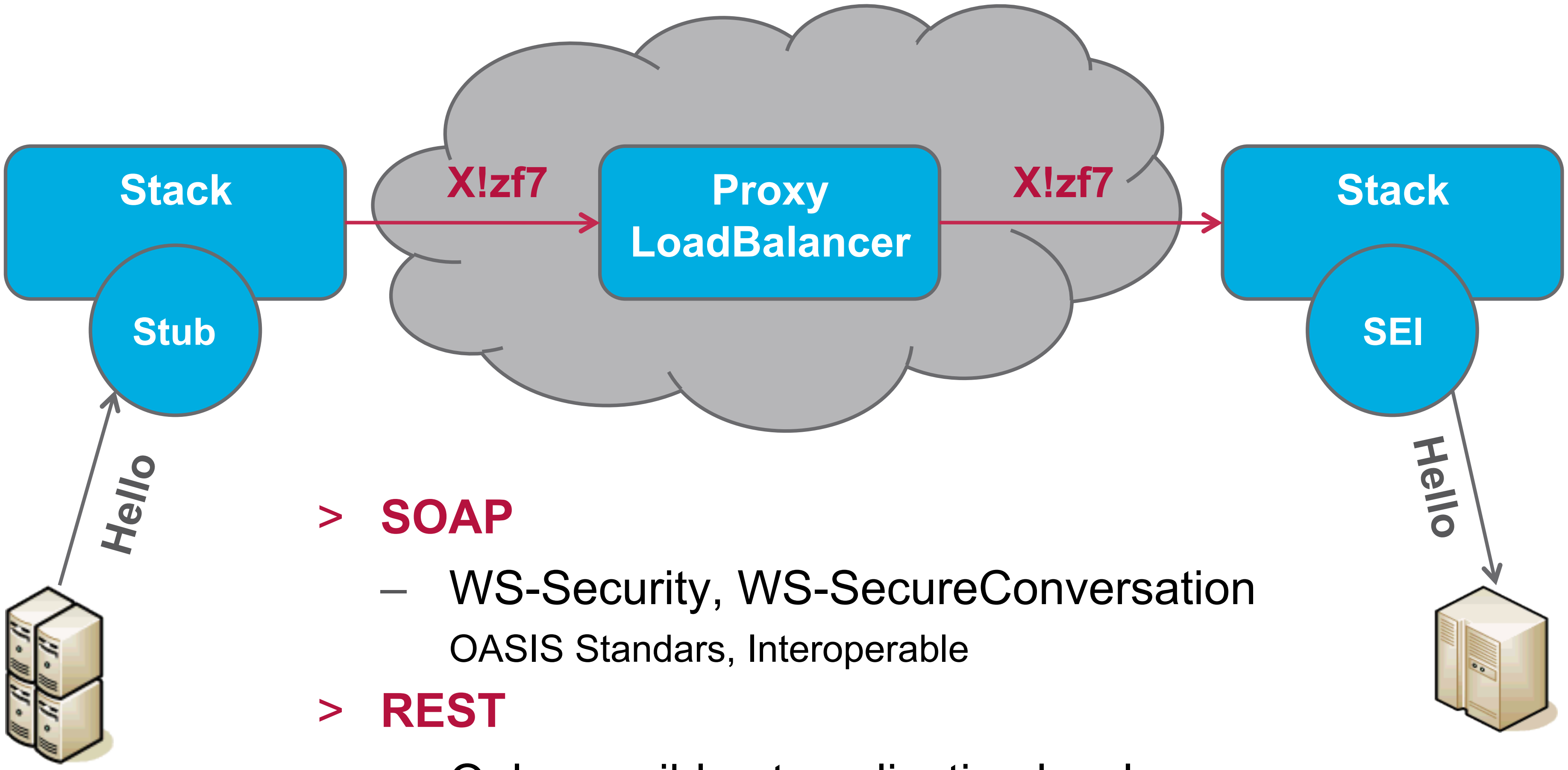
– Point-to-point security using SSL

HTTPS

– IETF Standard

Interoperability ensured

End-to-end security



> **SOAP**

– WS-Security, WS-SecureConversation

OASIS Standars, Interoperable

> **REST**

– Only possible at application level

No Standards, Limited interoperability

End-to-end security
SOAP interoperability

```
<wsp:Policy wsu:Id="SPortBindingPolicy">
  <sp:SymmetricBinding>
    <wsp:Policy>
      <sp:AlgorithmSuite>
        <wsp:Policy>
          <sp:TripleDesSha256Rsa15 />
        </wsp:Policy>
      </sp:AlgorithmSuite>
      <sp:EncryptBeforeSigning />
      <sp:EncryptSignature />
      <sp:Layout>
        <wsp:Policy>
          <sp:Strict />
        </wsp:Policy>
      </sp:Layout>
      ...
<wsp:Policy wsu:Id="SPortBinding_echo_Input_Policy">
  <sp:EncryptedParts>
    <sp:Body />
    ...
<binding name="SPortBinding" type="tns:S">
  <wsp:PolicyReference URI="#SPortBindingPolicy" />
  <operation name="echo">
    <input>
      <wsp:PolicyReference URI="#SPortBinding_echo_Input_Policy" />
    ...
```

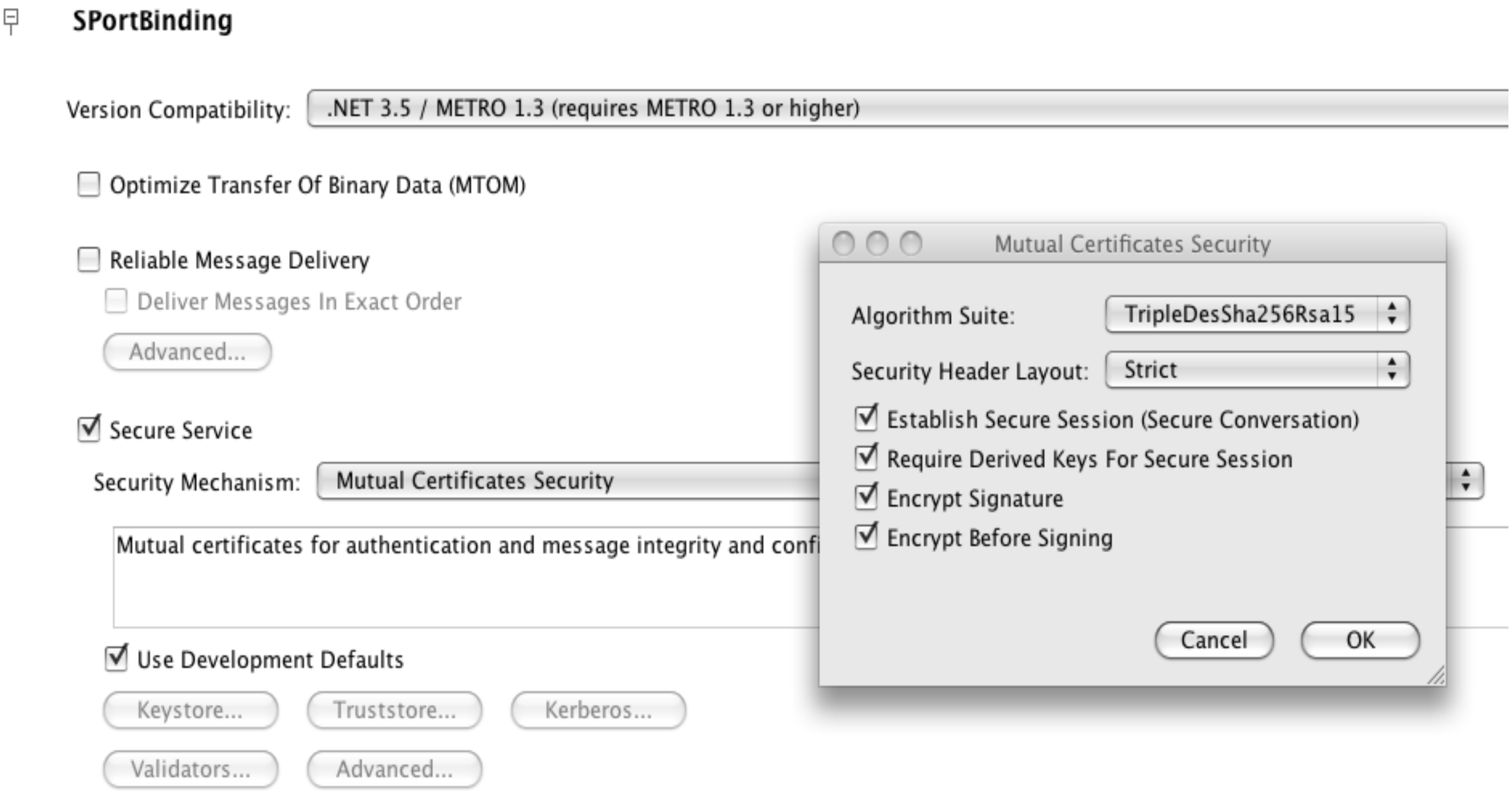
> **WS-Policy**

- Common policy language
- W3C Standard

> **WS-SecurityPolicy**

- Expressing security domain requirements
- OASIS Standard

End-to-end security
Metro (SOAP) tooling support



SPortBinding

Version Compatibility: .NET 3.5 / METRO 1.3 (requires METRO 1.3 or higher)

□ Optimize Transfer Of Binary Data (MTOM)

□ Reliable Message Delivery

□ Deliver Messages In Exact Order

☑ Secure Service

Security Mechanism: Mutual Certificates Security

Mutual certificates for authentication and message integrity and confi

☑ Use Development Defaults

Mutual Certificates Security

Algorithm Suite: TripleDesSha256Rsa15

Security Header Layout: Strict

☑ Establish Secure Session (Secure Conversation)

☑ Require Derived Keys For Secure Session

☑ Encrypt Signature

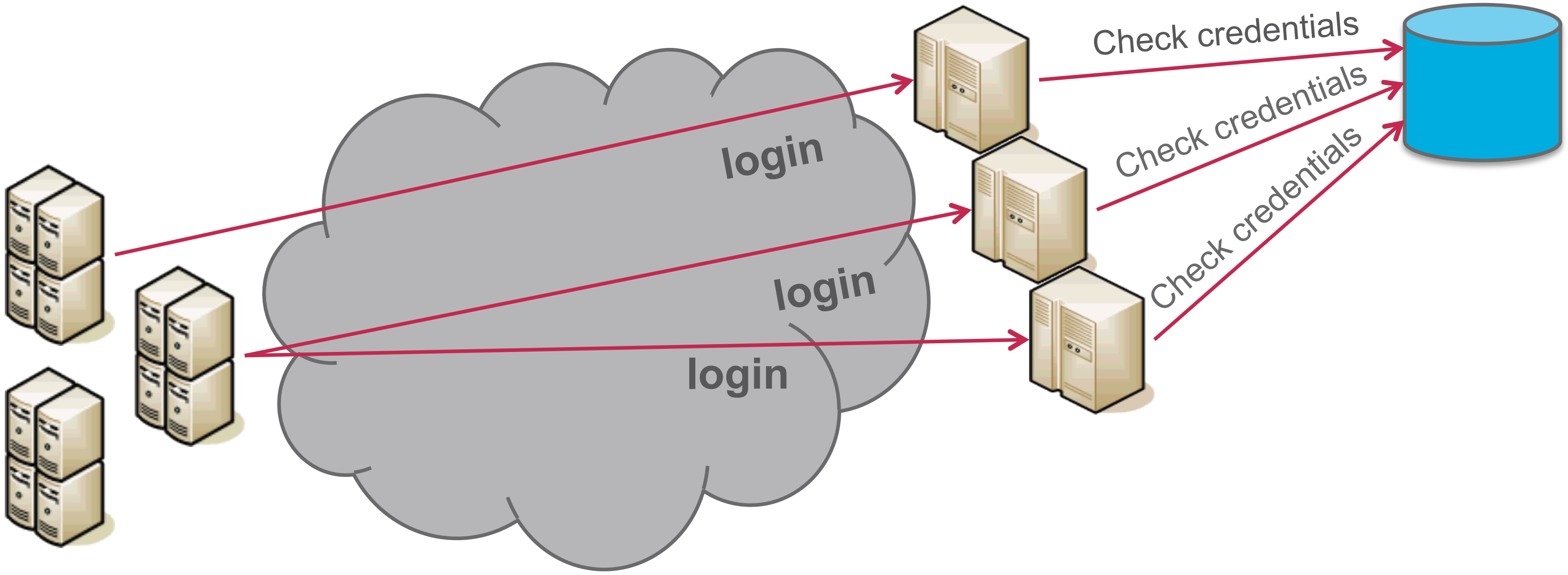
☑ Encrypt Before Signing

AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

3rd Party Identity

Authenticating users SOAP & REST

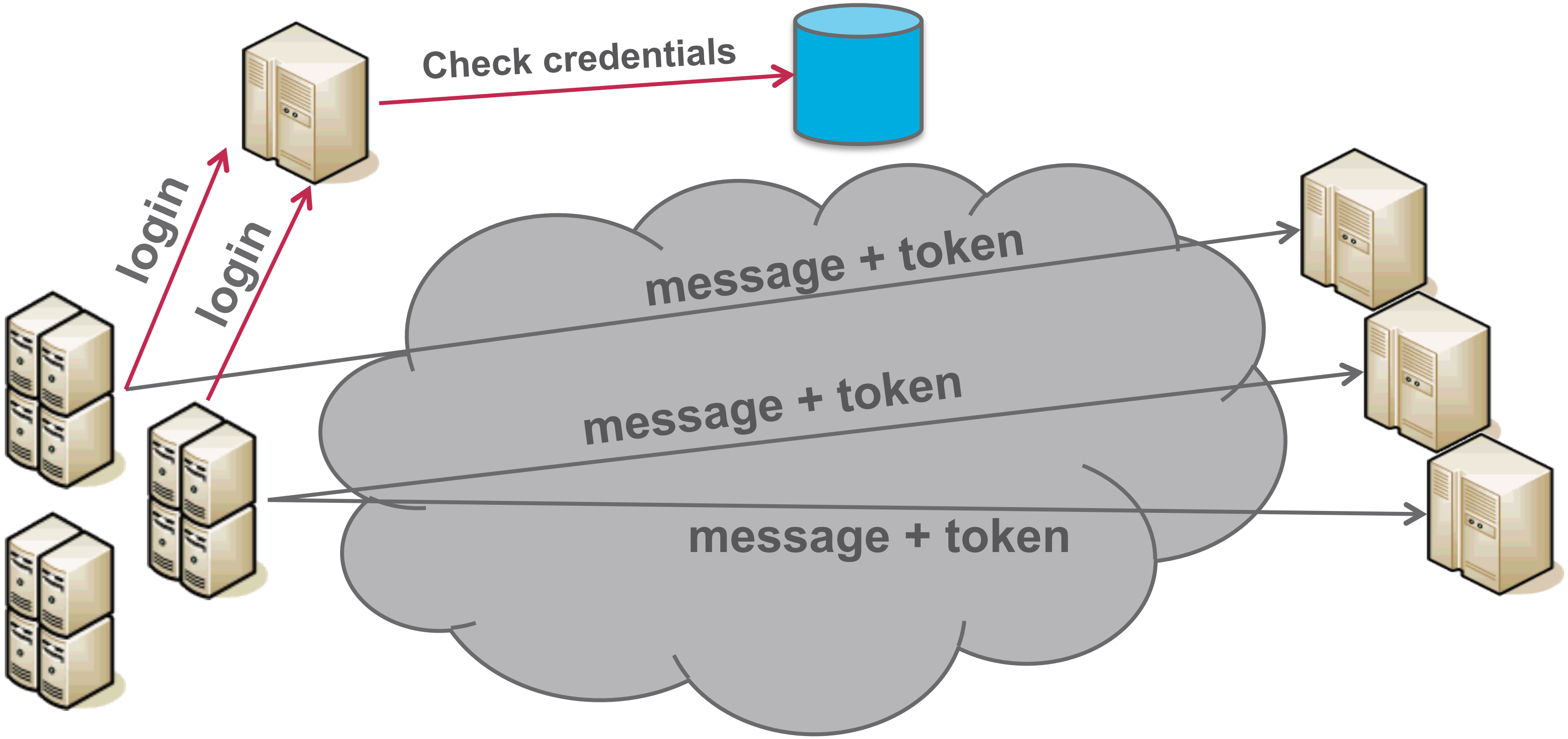


3rd Party Identity

SOAP WS-Trust (OASIS Standard) – Authentication delegation

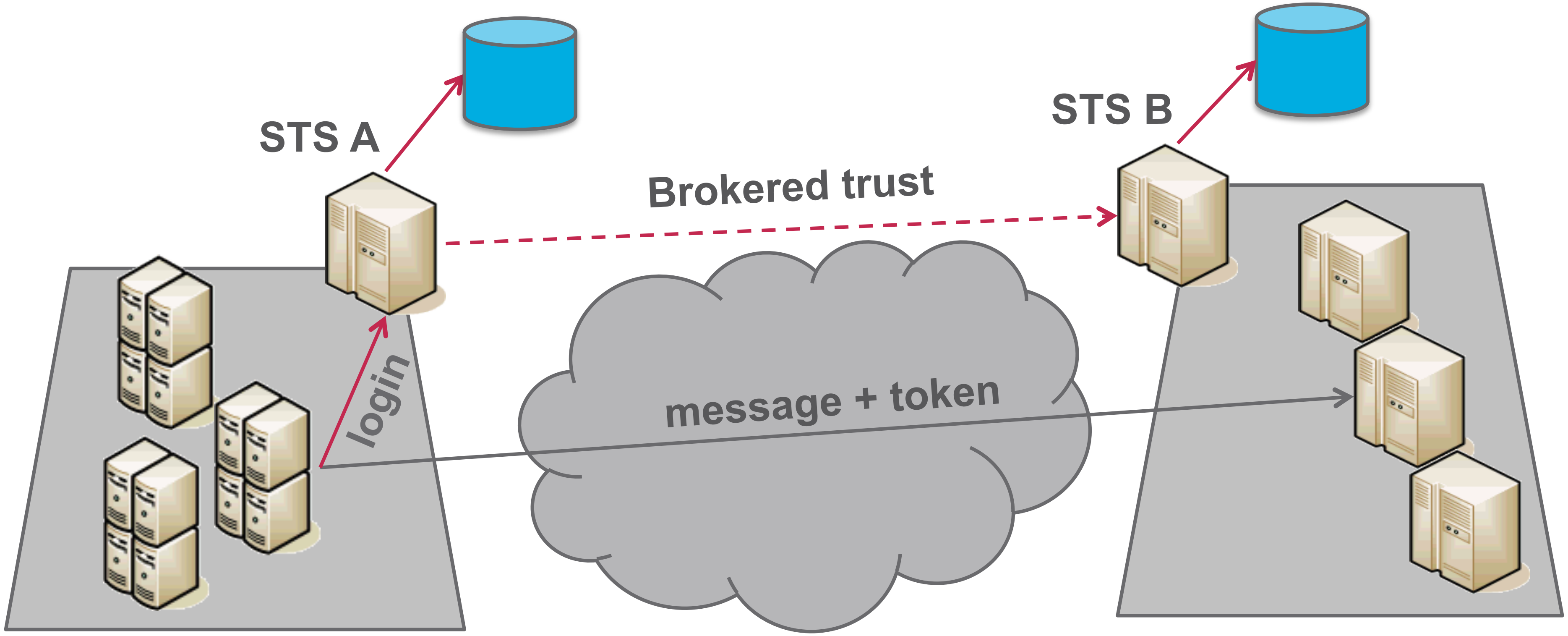
Secure Token Service

> SAML (OASIS), Kerberos (IETF) etc. tokens



3rd Party Identity

SOAP Brokered Trust (between domains or organizations)



AGENDA: Exploring Use Cases

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

3rd Party Access

OAuth Overview (Social applications, Private resources mash-ups …)

> **OAuth Protocol (http://oauth.net)**

- IETF RFC 5849
- **Not** an official standard

  Wikipedia: “OAuth (Open Authorization) is an open standard for authorization.”

> **Use Case**

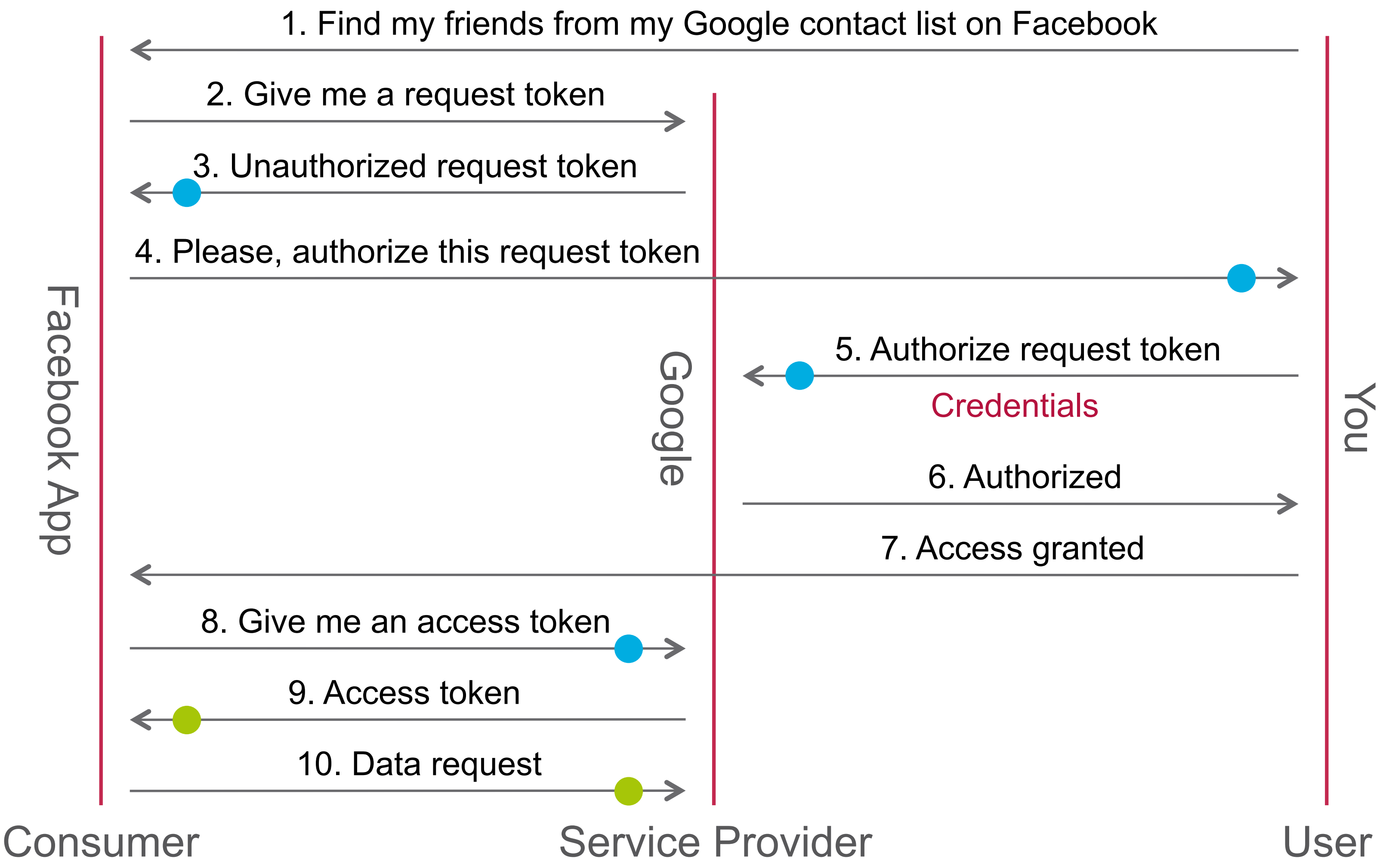
- Grant 3rd part (a temporary) access to your protected resources…
- …without a need to share your credentials

  e.g. user name / password

> **Involved Parties**

- Original Service User – owns the resources, holds credentials
- Service Provider – maintains resources, verifies credentials, controls access
- 3rd Party Consumer – accesses resources on User’s behalf

3rd Party Access
OAuth Operation



1. Find my friends from my Google contact list on Facebook

2. Give me a request token

3. Unauthorized request token

4. Please, authorize this request token

5. Authorize request token

Credentials

6. Authorized

7. Access granted

8. Give me an access token

9. Access token

10. Data request

Facebook App

Google

You

Consumer

Service Provider

User

OAuth / REST Example

> DEMO

3rd Party Access – OAuth / REST Example

Initialization

```
secrets = new OAuthSecrets().consumerSecret(MySecret);
params = new OAuthParameters().consumerKey(MyKey)
                              .signatureMethod("HMAC-SHA1")
                              .version("1.0");


oAuthFilter = new OAuthClientFilter(
                      client.getProviders(),
                      params,
                      secrets);
client.addFilter(oAuthFilter);
```

3rd Party Access – OAuth / REST Example

Obtaining The Request Token

```
resource = client.resource(“https://www.google.com/”);
requestToken = resource.path("accounts/OAuthGetRequestToken")
        .queryParam("scope", "https://www.google.com/m8/feeds/")
        .get(RequestToken.class);

authorizationUri = UriBuilder.fromUri(“https://www.google.com/”)
        .path("accounts/OAuthAuthorizeToken")
        .queryParam("oauth_token", requestToken.token)
        .build();

Desktop.getDesktop().browse(authorizationUri);
```

3rd Party Access – OAuth / REST Example

Obtaining The Access Token, 1st Data Request

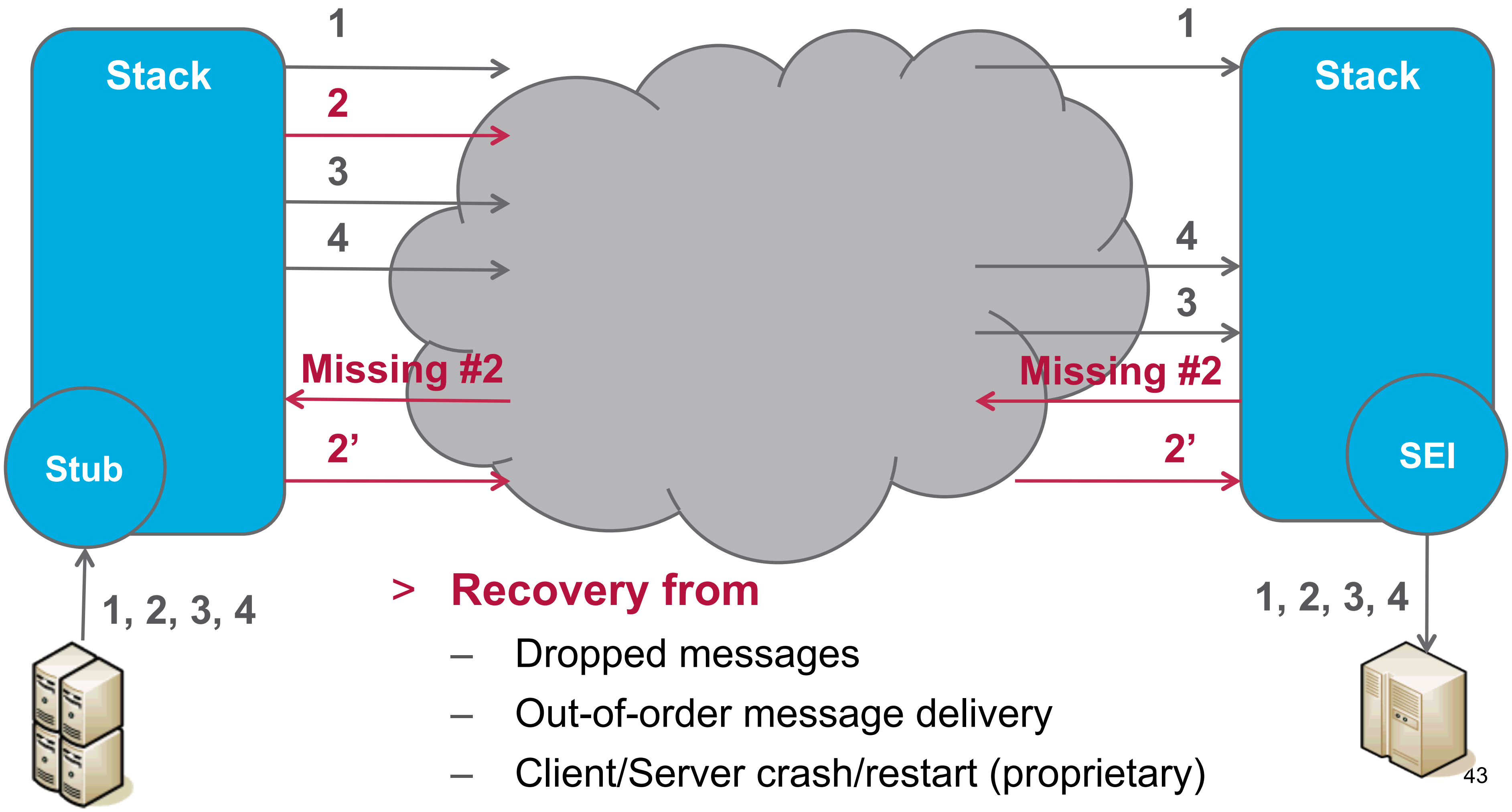
```
secrets.tokenSecret(requestToken.secret);
params.token(requestToken.token);
accessToken = resource.path("accounts/OAuthGetAccessToken”)
                      .get(AccessToken.class);


secrets.tokenSecret(accessToken.secret);
params.token(accessToken.token);
Feed contacts = resource.path("m8/feeds/contacts")
                        .path(”marek.potociar@gmail.com")
                        .path("full")
                        .queryParam("max-results", "15")
                        .get(Feed.class);
```

AGENDA: Exploring Use Cases

> Service Discovery and Automation
> Public Cloud APIs
> Small Device Support
> Service Mash-ups
> Large Messages
> Modeling Application State
> Security (point-to-point, end-to-end)
> 3rd Party Identity
> 3rd Party Access
> Intermittent Connectivity

Intermittent Connectivity

SOAP WS-ReliableMessaging (OASIS Standard)



> **Recovery from**
> – Dropped messages
> – Out-of-order message delivery
> – Client/Server crash/restart (proprietary)

AGENDA: Exploring Use Cases

	SOAP	REST
> Service Discovery and Automation	[✔]	[ ]
> Public Cloud APIs	[✔]	[✔]
> Small Device Support	[ ]	[✔]
> Service Mash-ups	[ ]	[✔]
> Large Messages	[✔]	[✔]
> Modeling Application State	[✔]	[✔]
> Security (point-to-point, end-to-end)	[✔]	[✔]
> 3rd Party Identity	[✔]	[ ]
> 3rd Party Access	[ ]	[✔]
> Intermittent Connectivity	[✔]	[ ]

- Service Discovery and Automation
- Public Cloud APIs
- Small Device Support
- Service Mash-ups
- Large Messages
- Modeling Application State
- Security (point-to-point, end-to-end)
- 3rd Party Identity
- 3rd Party Access
- Intermittent Connectivity

> THANK YOU.

Marek Potociar
Oracle

http://marek.potociar.net/
marek.potociar@oracle.com

Oracle @Jazoon, Booth 1

- **Participate in Prize Draw**
  - Daily. [21.06., 6:45 pm; 22.06. and 23.06. End of afternoon coffee break]
- **View demonstrations**
  - Java Card, Java SE Embedded, Java SE ARM, Java EE/GlassFish, Java FX, ADF, SOA Suite
- **Speak to Oracle Experts / Speakers** (English/German/French)
- **Provide Feedback on Java to Oracle team** and collect a giveaway
- **Meet/Have your picture taken with Duke**
- **Learn more about our Java 7 Launch webcast and Launch events**