Service Oriented Architectures (SOA): Distributed Systems over the Web
Uses of Web-based SOAs

My Server

My Web Service

HTTP
Uses of Web-based SOAs (cont.)

Data Center (Google, Oracle, etc.)

Customer

HTTP

Customer

Customer

Customer
Uses of Web-based SOAs (cont.)

Cloud Provider (Amazon, Salesforce, etc.)

Server

My Startup Service

HTTP

User

User

User

User
Advantages/Disadvantages of Web-based SOAs

- **Advantages**
  - Platform and language independence
  - Built on widely-adopted standards (HTTP, XML, …)
  - HTTP traffic passes through firewalls untouched

- **Disadvantages**
  - Relatively high overhead due to use of XML
SOA Alternatives - SOAP and REST

- **SOAP**
  - Full distributed object system with IDL (WSDL)
  - Arbitrary method calls
  - Stateful services with stateful interactions
  - Support for advanced features: security, transactions, etc.
  - Tightly coupled distributed applications: Google services, enterprise applications

- **REST**
  - REpresentational State Transfer
  - No IDL
  - Simplified stateless interactions (self-describing messages)
  - Only HTTP get, head, put, post, delete methods
  - State maintained on clients and resources, accessible by other services
  - Loosely coupled distributed applications: twitter, flickr, …
Web Services with SOAP

- HTTP is simply a transport layer for WS-SOAP
- SOAP messages are tunneled through HTTP
- There is one URI, which identifies the service
Web Services with SOAP (cont.)

- All messages use HTTP posts and the unique service URI
- Service maintains state ("order" object maintained by service is created in one message exchange and operated on in subsequent message exchanges)
- WSDL interface description used to generate client stubs
Web Services with SOAP: Server Side

- e.g. Apache HTTP Server
- e.g. Apache Tomcat Server
- e.g. Apache Axis Server

sometimes combined

your Web service
Web Services with SOAP: WS-Security

- Provides for:
  - Signing of SOAP messages
  - Encryption of SOAP messages
  - Attachment of security tokens to SOAP messages

- Security token options:
  - X.509 certificates
  - Kerberos tickets
  - User ID, password
  - SAML assertions
  - ...

Web Services with SOAP: SAML

- An XML-based open standard data format for exchanging authentication and authorization data between parties

- Assertion A was issued at time t by issuer R regarding subject S provided conditions C are valid

- Assertion types:
  - *Authentication statement*: subject S authenticated with the issuer using xx authentication mechanism
  - *Attribute statement*: subject S has attributes x, y, and z
  - *Authorization statement*: subject S is authorized to perform action A on resource R (limited expressibility ⇒ XACML)
Web Services with SOAP: Other Features

- **WS-AtomicTransaction**
  - Supports two-phase commit functionality for transactions

- **WS-ReliableMessaging**
  - End-to-end message reliability (ACKs)
  - Message retry
Web Services with REST

- HTTP is the application layer for WS-REST
- REST messages, for a given service, can operate on multiple resources identified by their respective URIs
Web Services with REST (cont.)

- Operations are carried out using different HTTP methods operating on resources with their own URIs
- Two resources: “books” and “orders”
- Server-side state pushed into resources, which can be accessed concurrently by different services
Web Services with REST (cont.)

- Communication is stateless: each client request to the server must contain all information needed to understand the request, without referring to any stored context on the server
- Application state is pushed to edges: clients and resources
- Client state can be maintained using cookies
- Server-side state pushed into resources, which can be accessed concurrently by different clients and different services
Web Services with REST: Principles

1. Identify *all* resources through URIs
2. Uniform and simple interface: HTTP get, head, put, post, delete
   - 1. and 2. ⇒ “small set of verbs applied to a large set of nouns”
3. Self-describing messages
4. Hypermedia driving application state: applications “navigate” interconnected set of resources
5. Stateless interactions
SOAP vs. Rest: State Handling

SOAP: Shopping cart is state maintained by service, available only to clients of that service that know how to access it.

REST: Shopping cart is resource stored persistently on server, accessible via its URI to any client and any service.
Web Services with REST: Security

- REST relies solely on HTTPS for security
  - HTTPS = HTTP + TLS/SSL
  - Provides secure (encrypted) communication channel
  - Does *not* provide signatures/authentication
  - XML inside REST messages can be signed and encrypted
  - HTTP supports basic authentication (userID, password only)
Web-based SOAs: Summary

- Web-based SOAs have many advantages for building distributed applications over the Internet
  - Platform/language independence
  - Standards
  - Firewall-resilient HTTP

- SOAP and REST are two Web SOA alternatives
  - SOAP is more complex but has more features, including better security support
  - REST is simpler and allows easy integration of services via shared resources (mash-ups, etc.)