Web Application Security

Rajendra Kachhwaha
rajendra1983@gmail.com

November 3, 2015

Outline

Basics:

1. A web service is a self-contained software component that performs specific functions and publishes information about its capabilities to other components over a network.
2. Web services offer a coherent mechanism for alleviating the task of integrating multiple web applications, coordinating standards to pass data, protocols, platforms, and so on.
3. Web services provide a means for different organizations to connect their applications with one another to conduct dynamic e-business across a network, no matter what their application, design, or run-time environment.
4. It is XML-based information exchange systems that use the Internet for direct application-to-application interaction. These systems can include programs, objects, messages, or documents.

Basics:

Web services are based on:

1. A set of Internet standards, including the Web Services Definition Language (WSDL), an XML format for describing the connection points exported by a service;
2. The Universal Description, Discovery, and Integration (UDDI) specification, a set of XML protocols and an infrastructure for the description and discovery of web services;
3. The Simple Object Access Protocol (SOAP), an XML based protocol for messaging and RPC-style communication between web services.
4. Web services can describe their own functionality and search out and dynamically interact with other web services via WSDL, UDDI, and SOAP.

What distinguishes web services from plain old web sites?

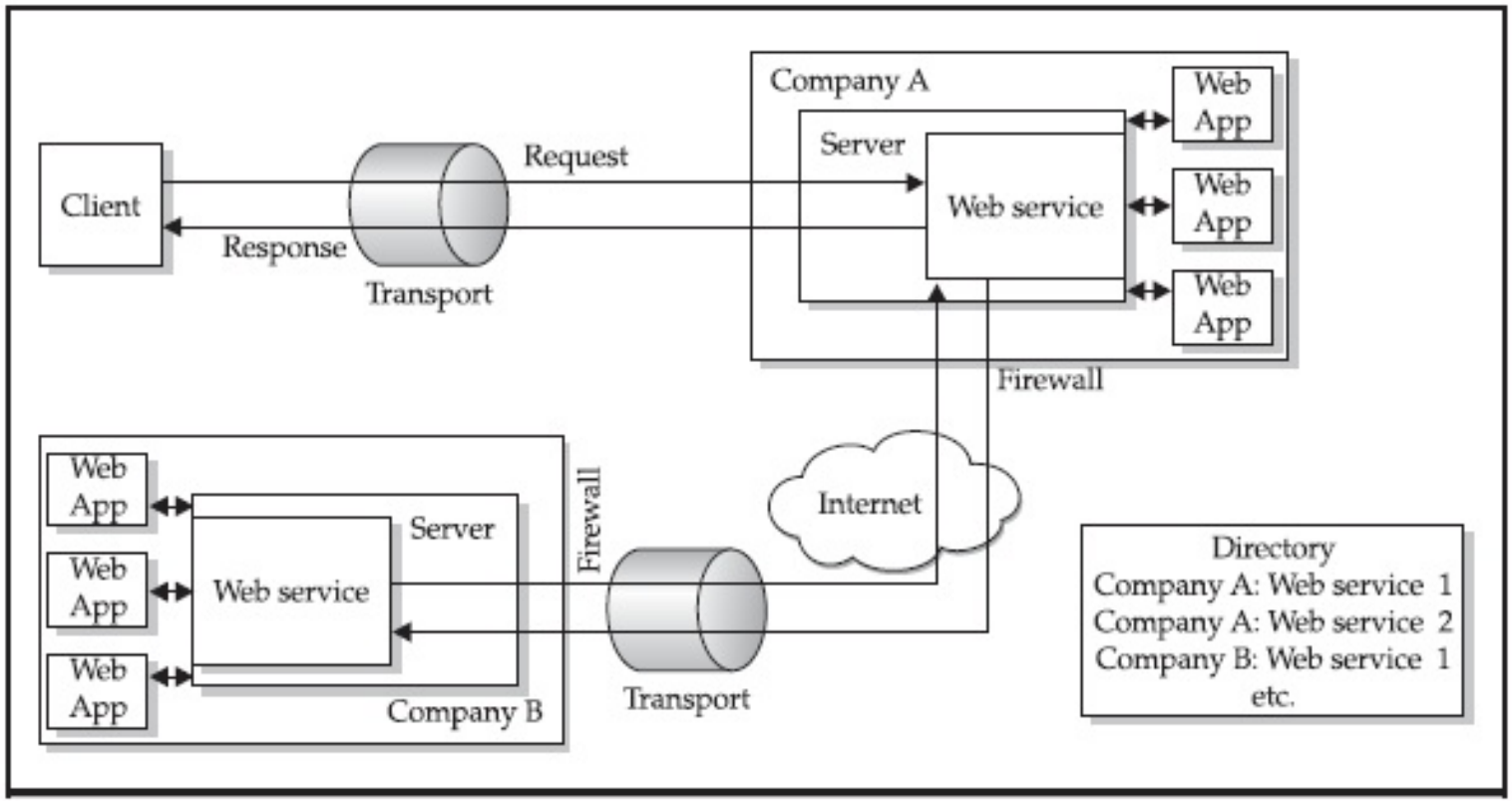
What distinguishes web services from plain old web sites?

1. Web services are targeted at unintelligent agents rather than end users.
2. As Microsoft puts it, "In contrast to web sites, browser-based interactions, or platform-dependent technologies, web services are services offered computer-to-computer, via defined formats and protocols, in a platform independent and language-neutral manner."

How web services integrate into the typical web application architecture:

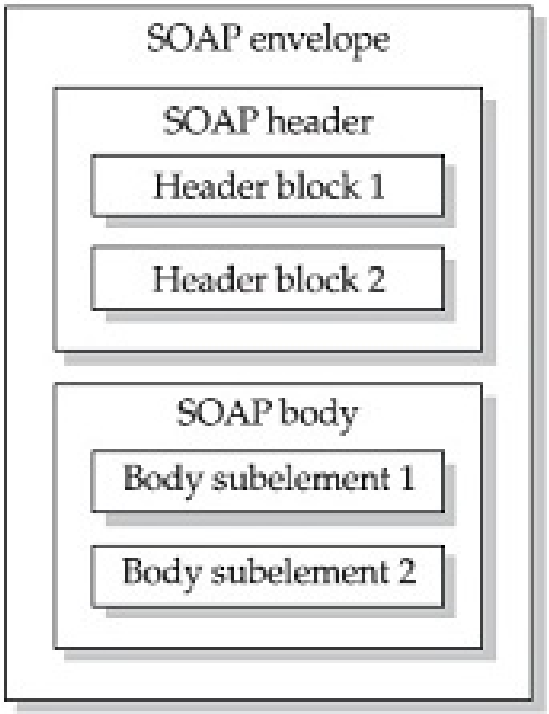
1. We take a web service at hypothetical Company A that publishes information about Company A's applications to other companies (hypothetical Company B) and Internet clients.
2. We have wrapped our web services inside of a generic "Server" that mediates communication with web services.
3. SOAP is encapsulated in whatever transport is used-the most common example is SOAP over HTTP.
4. SOAP is the messaging protocol used for communication with a web service.
5. SOAP provides the definition of an XML document, which can be used for exchanging structured and typed information between peers in a decentralized, distributed environment.

How web services integrate into the typical web application architecture:



How web services integrate into the typical web application architecture:

SOAP messages are comprised of three parts: an envelope, a header, and a body.



SOAP Request:

```
POST /StockTrader HTTP/1.1
Host: www.stocktrader.edu
Content-Type: text/xml; charset "utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"
<SOAP-ENV:Envelope
   xmlns:SOAP-ENV "http://schemas.xmlsoap.org/soap/envelope/"
   SOAP-ENV:encodingStyle "http://schemas.xmlsoap.org/soap/encoding/">
     <SOAP-ENV:Header>
       <m:quote xmlns:m "http://www.stocktrader.edu/quote"
           env:actor "http://www.w3.org/2001/12/soap-envelope/actor/next"
           env:mustUnderstand "true">
       <m:reference>uuid:9oe4567w-q345-739r-ba5d-pqff98fe8j7d</reference>
       <m:dateAndTime>2010-03-28T09:34:00.000-06:00</m:dateAndTime>
     </m:quote>
   <SOAP-ENV:Body>
        <m:GetQuote xmlns:m "Some-URI">
           <symbol>MSFT</symbol>
        </m:GetQuote>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP Response:

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset "utf-8"
Content-Length: nnnn
<SOAP-ENV:Envelope
   xmlns:SOAP-ENV "http://schemas.xmlsoap.org/soap/envelope/"
   SOAP-ENV:encodingStyle "http://schemas.xmlsoap.org/soap/encoding/"/>
    <SOAP-ENV:Body>
       <m:GetQuoteResponse xmlns:m "Some-URI">
            <Price>67.5</Price>
       </m:GetQuoteResponse>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP Hacking Tools:

1. WebService Studio
2. SoapUI
3. WSDigger
4. WSFuzzer
5. SoapClient.com

WSDL:

1. WSDL is central to the concept of web services.
2. It is a core component of the web service itself, the mechanism by which the service publishes or exports information about its interfaces and capabilities.
3. WSDL is typically implemented via one or more pages that can be accessed on the server where the web service resides (typically, these carry .wsdl and .xsd file extensions).
4. WSDL is an XML grammar for describing network services as collections of communication endpoints capable of exchanging messages.
5. In essence, this means a WSDL document describes what functions ("operations") a web service exports and how to connect ("bind") to them.

WSDL Example:

Here is a sample WSDL definition for a simple web service that provides stock-trading functionality. Note that our example contains the following key pieces of information about the service:

1. The types and message elements define the format of the messages that can be passed.
2. The portType element defines the semantics of the message passing.
3. The binding element specifies various encoding over a specified transport such as HTTP, HTTPS, or SMTP.
4. The service element defines the endpoint for the service (a URL).

UDDI:

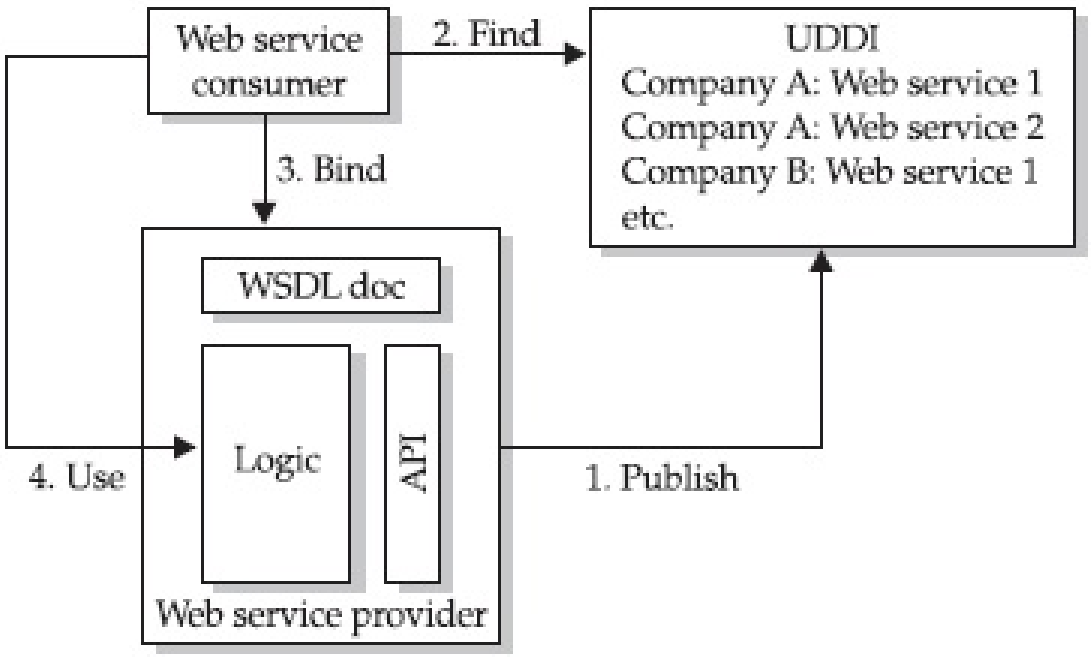
1. UDDI is a specification for distributed web-based information registries of web services.
2. UDDI is also a publicly accessible set of implementations of the specification that allow businesses to register information about the web services they offer so that other businesses can find them.
3. UDDI directories fall into two categories: public and private. A public UDDI is what companies would use in order to offer their web services to the public. An example of a public UDDI directory is xmethods.net.
4. Private UDDI directories are usually implemented in large corporations for internal or B2B use. These directories are hosted internally at the company and are usually only accessible to the employees or partners of the organization.

How UDDI fits into the overall framework of web services:

1. First, a web service provider publishes information about its service using the appropriate API.
2. Then, web services consumers can look up this particular service in the UDDI directory, which will point the consumer toward the appropriate WSDL document(s) housed within the web service provider.
3. WSDL specifies how to connect to and use the web service, which finally unites the consumer with the specific functionality he or she was seeking.

cont.

How UDDI fits into the overall framework of web services:



The "publish, find, bind" interaction among UDDI, WSDL, and web services. All arrows represent SOAP communications.