SOAP-Enabled Web Service

Friday, February 5, 2021 3:54 PM

1. What is a web service?
2. How does the architecture of a web service differ from a web application

Web service: self-describing web enabled component
Web application: does not follow SOA (service published in an internet registry, can be accessed by any client and invoked)
Evolved from component technology (intranet(EJB, DCOM) and internet(web service) enabled component

Comparing protocols:
Additional differences in terms of security (web service has flexible security, web app has brute force security)
EJB + wrapper code = web service that responds to SOA protocol or HTTP protocol

Net Beans or Eclipse used to build web services today

```
POST/Stockquote HTTP/1.1
Host: www.stockquote.com
<!--(web service is hosted here, mapped to ip address) -->
Content_type: SOAP/XML
<?xml>
<envelope>
<body>
<getStockPrice>
<!--method name, user defined -->
<stockname>IBM</stockname>
</getStockPrice>
</body>
</envelope>
</xml>
```

If you want to invoke a web service, client must create a request like this.

Web service hosted on an application server

Client -> Soap request -> SOAP webservice
<-SOAP response<-

When the server receives the above XML, it will

1. Intercept soap request
2. Parse XML
3. Extract method name and parameters (getStockPrice("IBM") implemented in the web service)

SOAP Response has response headers

```
HTTP/1.1 200/OK
Content-Type: application/soap
Content-Length: 126

<?xml>
    <envelope>
        <body>
            <getstockpriceresponse>
                <price>34.5</price>
            </getstockpriceresponse>
        </body>
```

```
    </envelope>
</xml>
```

If response HTTP -> not OK -> request failed

HTTP can host more than one method
HTTP headers + XML = SOAP request/response

Firewall: port number 80
Request permitted

UTF-8: english characters
UTF-16: chinese characters

If we want to keep the stock name private, we can encrypt only required information. Only client and web service can read the message.

- **Selective XML encryption**

Security flexibility only for SOAP-enabled web services. Web applications do not have flexible security (brute force security only). Encryption takes a considerable amount of time, selective encryption reduces unnecessary encryption.

Web application protocol:

Browser asks for information (enter stock name). Once submitted (POST HTTP request), invokes the application (HTTP response from web application)

SOAP Request(through web app):

```
POST/Stockquote.php HTTP/1.1
Host: www.stockquote.com
<!-- blank line -->
Stock Name = "IBM"
```

Only parameter name passed (no method name==> **only one method is hosted by web app**)
Hide the stock name IBM
(HTTP security encrypts everything)

SOAP Response(through web app):

```
POST HTTP/1.1 200 OK
Content-type:

<HTML>
<P>IBM Price: 40</p>
</HTML>
```

Conclusion:

1. SOAP-enabled web service has flexible security compared to simple web application
2. Web service follows service oriented architecture

SOAP uses HTTP as a transport protocol (over SSL - brute force security)

In the case of web application, we can encrypt the whole thing.

SOAP:

1. RPC: exclusively specify method name and parameter of web service

2. Document-centric: no need to specify method name of web service