WEB SERVICES CYBER-SECURITY ISSUES*

D. Quock#, ANL, Argonne, IL 60439, U.S.A.

Abstract

The Web's potential for distributed programming has been proven not only in the business realm, but also in the accelerator controls domain. Web services describes clients and servers that communicate over the Internet's Hypertext Transfer Protocol (HTTP) using predefined Internet-based Application Programming Interfaces (APIs). It is the uniqueness of Web services transactions such as cloud computing, data sharing, and data archiving that give rise to the security concerns of Web services (authentication, data integrity, non-repudiation, and privacy). At Argonne National Laboratory's Advanced Photon Source, Simple Object Access Protocol (SOAP)-based Web services were implemented into the Integrated Relational Model of Installed Systems (IRMIS) as the application interface to Oracle's Content Server document management software. This report reviews the basics of Web services, cyber-security issues that are inherent for Web services, current Web services security implementation practices, and future directions of Web service security development efforts where the overriding goal of Web services security is to focus on managing risk and protecting data.

BASICS OF WEB SERVICES

In simplest terms, Web services are distributed Internet applications that have standard-based interfaces. Web services are typically thought of as being divided into two main technologies:

1. Big Web Services: This technology uses Extensible Markup Language (XML) messages that follow the Simple Object Access Protocol (SOAP) standard.

2. RESTful Web Services: The representational state transfer (REST) software architecture uses PUT, GET, DELETE and POST HTTP methods to integrate Web browsers with underlying client/server software applications.

Service-Oriented Architecture (SOA) is model-based software that is typically constructed from loosely coupled Web services. SOA can be broken down into the three layers: business workflow, Web services, and communication [1]. Table 1 demonstrates that SOA adds three layers on top of the standard client-server architecture and shows the associated Web services standards that are used at each layer.

Table 1: SOA Architectural Layers

Architectural Layer	Web Services Standards
Business Workflow	BPEL
	WSCI
Web Services	WSDL
	UDDI
Communications	
SOAP	XML
Client-Server Transports	
HTTP	
SSL	
TCP/IP	

Web Services Standards

At the highest level of SOA, Business Process Execution Language (BPEL) is used to describe and execute the business processes. An alternative to BPEL is the World Wide Web Consortium (W3C)'s standard Web Service Choreography Interface (WSCI). These two standards are currently diverging as industry is divided in its support of either business workflow standard. The role that BPEL (or WSCI) plays in SOA is orchestrating the overall business workflow by providing mapping between the services and business processes through documents.

At the next level of SOA, the standard Web Service Description Language (WSDL) provides static interface definitions for the software components that are accessible to clients. The Universal Description, Discovery and Integration (UDDI) is a specification for repositories where organizations can publish services that they provide and describe the interfaces to their services via WSDLs.

At the communications layer of SOA, messages are transmitted through SOAP, which is an envelope containing a header and body. The services that are communicating with each other can be identified through their unique name contained in SOAP messages.

ADVANCED PHOTON SOURCE WEB SERVICES

The benefits of SOA to organizations is the flexibility of implementing business processes on top of Web services and the ability to compose and re-compose systems frequently. SOA provides a peer-to-peer style of architecture with a general statelessness of services. One example of how Web services technology was implemented at Argonne National Laboratory's Advanced Photon Source is in the interaction between the in-house built IRMIS accelerator controls relational database

*Work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-06CH11357.

#quock@aps.anl.gov

software application and Oracle Content Server document management software.

Figure 1 shows an IRMIS Applications Organizing Index display that has retrieved general document information for documents stored in Oracle Content Server (ICMS). The search for documents in Content Server was done in IRMIS using a simple PHP SOAP client utility that looks similar to:

```
$client = new SoapClient($wsdl,

        array('login'=>$user_name,'password'=>$user_password));
```

where $wsdl is the https Internet address of the location where the SOA WSDL file can be obtained. In this example, the WSDL file defines the interface to a document search software component for Oracle Content Server. The search results are retrieved and stored locally by using the PHP statements:

```
$param = array("queryText"=>$searchString,"sortField"=>"dInDate");
$retVal = $client->AdvancedSearch($param);
$search_results = $retVal->AdvancedSearchResult->SearchResults;
```

Figure 2 shows the home Web page for Oracle Content Server where search options are provided to the user for performing a manual search on its document database. The same search string can be entered into the Content Server's Comment field to obtain the same set of document search results. The benefit of using SOAP Web services provided by Oracle Content Server in the IRMIS PHP application is that the IRMIS user only needs to interface to one software application (IRMIS) to get information provided from two separate database applications. The IRMIS display provides an Internet link to directly launch each document in its native format, thus providing even more ease of use and efficiency to the user. The implementation shown here for Web services in IRMIS is the very simplistic case of client software interacting with a server behind the same firewall in the same organization.

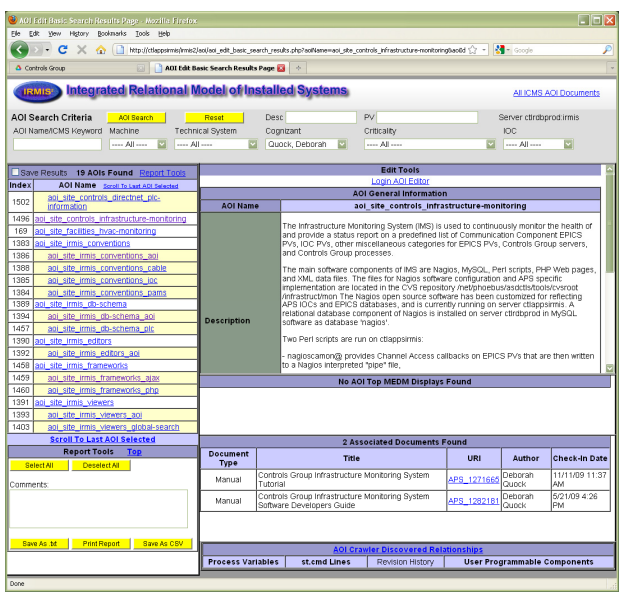

Figure 1: IRMIS display with ICMS search results.

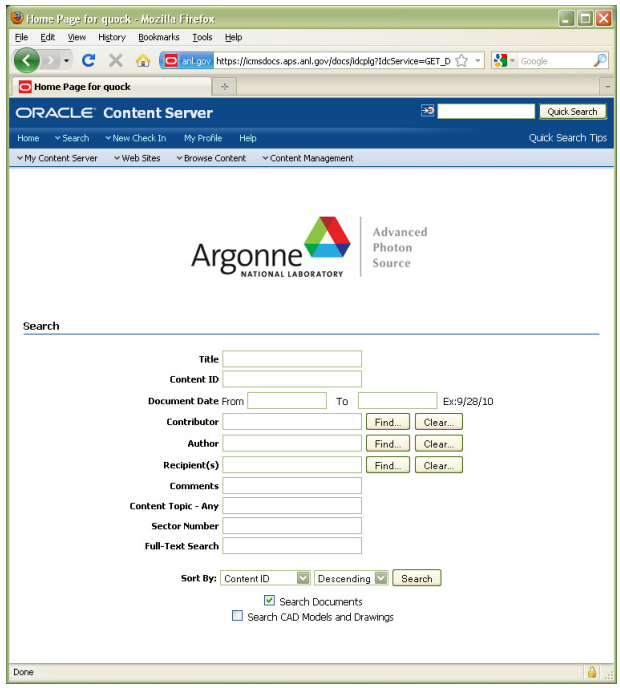

Figure 2: ICMS search home page.

WEB SERVICES CYBER-SECURITY ISSUES

The trade-off for having software components that are defined by their interfaces and can be accessed on the Internet is the increased complexity of the systems where they are used. This increase in complexity in SOA applications is due to several factors including:

- The software used to manage the Web services is complex;
- The boundaries of communication may extend outside of an organization's intranet;
- Dynamic reconfiguration of a client application can be easily obtained through both combination and reuse of individual Web services.

This increase in complexity of a SOA application results in a wider variety of cyber security threats. Examples of such security threats are message alteration, message reading, man-in-the-middle attacks, principal spoofing, forged claims, message replay, and denial of service.

WEB SERVICES CYBER-SECURITY STANDARDS AND ORGANIZATIONS

Web Services Organizations

To effectively deal with cyber security threats that are specific to SOA technology, cooperation and coordination among the vast number of businesses and other institutions using and providing Web service is crucial. The Organization for the Advancement of Structured Information Standards (OASIS) is a not-for-profit consortium that drives the development, convergence and

adoption of open standards for the global information society. The consortium produces more Web services standards than any other organization along with standards for security, e-business, and standardization efforts in the public sector, and for application-specific markets [2]. OASIS sponsors ebXML (Electronic Business using eXtensible Markup Language), a modular suite of specifications. The World Wide Web Consortium is an international community where member organizations, a full-time staff, and the public work together to develop Web standards. Among the many standards developed and supported by W3C are XML Encryption and XML Signature [3]. Corporations such as IBM, Microsoft, and Oracle also contribute to the development of Web services standards.

Web Services Cyber-Security Standards

In addition to the Web services standards list in Table 1, there are several standards specific to addressing cyber-security issues.

- Security Assertion Markup Language (SAML) defines authentication and authorization assertions. SAML assertions can be included in the header or in the payload of a SOAP message.
- WS-* is a general nomenclature used to refer to a family of Web services standards supported by various organizations. Common WS- standards include:
    - WS-Security
      Defines security tokens that can be used for claims of authentication or proof of some right.
    - WS-ReliableMessaging
      Describes a protocol that allows SOAP messages to be reliably delivered between distributed applications in the presence of software component, system, or network failures.

WEB SERVICES DESIGN PRACTICES

For new Web services applications, security considerations should be applied to every aspect of the software development life cycle:

- Security requirements
- Security architecture
- Web services security standards
- Certification (show that software complies with security requirements and security standards)
- Run-time security monitoring
- Penetration testing

The possibilities for implementing software security practices range from the very simple and well-known coding techniques to extremely analytical and involved source code analysis methodologies. A rule of thumb for designing user interfaces is that simple interfaces with few options are easy to test and audit. “Giant APIs require giant security measures” [4]. Another commonly used practice is SSL (Transport Layer Security) for encrypting and verifying the integrity of every client request. Source code analysis tools for identifying security weaknesses include:

- Vulnerability databases that are published to the general public (e.g., Microsoft publishes one);
- Pointer and reflection analysis that constructs a call graph that allows input data to be traced along function calls [1].

There are many Web services security analysis software products available on the market. The functionalities and standards that they typically examine include conformance validation, integrity checks, XML schema validation, XML encryption, XML signature, WS-Security, user authentication, audit, alert, Web services access control, and content inspection. IBM has developed a service-oriented analysis and design process for modeling, analyzing, designing, and producing a SOA application that is based on Java and IBM WebSphere software development tools. Microsoft, Oracle, Sun, and various other companies have also developed Web services design and Web services manager software tools.

CONCLUSION

The complex nature of SOA applications calls for governance of SOA and its underlying Web services technology. Cooperation among industry and Web services standards organizations is crucial to ensure reliable Internet-based business and government processes, and safeguarding of intellectual property and high-security-level government data. Web services standards organizations are well established and have received widespread support and contributions from major computer and IT corporations. Smaller institutions benefit from readily available Web services standards and Web services security products. Theoretical research in Web services security technology is active at many universities and continues to advance the software security design and monitoring tools available to the general public [1]. At Argonne National Laboratory, Web services applications have been deployed successfully as they make efficient use of disparate software applications.

REFERENCES

[1] C. Gutierrez, ‘Web Services Security Development and Architecture: Theoretical and Practical Issues,’ (IGI Global, Hershey, PA: 2010).
[2] OASIS, http://www.oasis-open.org/who/, 2010.
[3] W3C, http://www.w3.org/standards/webofservices/, 2010.
[4] C. Snyder and M. Southwell, ‘Pro PHP Security,’ (Apress, Berkeley, CA: 2005).