Interested in learning
more about security?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Web Services Security - An Overview

Many information technology visionaries say that the Internet is primed for the next phase of its evolution. The first phase, the physical infrastructure build out, has been completed, and it is now time to make use of the new communications and processing capacity to produce value. One strategy used to improve productivity is to increase the speed and quality of information flow. Another strategy is to make it easier for producers and consumers of information to locate each other and exchange value. One tactic that wi...

Copyright SANS Institute
Author Retains Full Rights

AD
CounterTack
Counter Tack Active Monitoring for In-Progress Attacks
GET THE WHITE PAPER NOW >>>

Web Services Security – An Overview

Practical Assignment: GSEC 1.2f
Scott Burns
November 20th, 2001

Introduction

Many information technology visionaries say that the Internet is primed for the next phase of its evolution. The first phase, the physical infrastructure build out, has been completed, and it is now time to make use of the new communications and processing capacity to produce value. One strategy used to improve productivity is to increase the speed and quality of information flow. Another strategy is to make it easier for producers and consumers of information to locate each other and exchange value. One tactic that will be used to facilitate these exchanges is the adoption of a new approach to application construction known as "web services".

An example of a web service is the stock price-updating feature in the Quicken personal finance software package. When the user requests a price update the software queries servers provided by Intuit, Quicken's maker, and they return current prices for the stock symbols the user is interested in.

This paper will use the definition of a web service as found on O'Reilly xml.com's "Web Services Primer", that is, "…component services that others might use to build bigger services…"[ORA-Primer]. At a basic level these services are designed to replace many commonly used middleware protocols (CORBA, DCOM, etc) with a vendor and language neutral services architecture that operates over HTTP. A basic component interaction involves a client process sending an XML document via HTTP to a service and receiving an XML document in return. The web services paradigm also provides the means to advertise the availability of component services and define rules for their use.

Web services components themselves can query other components as part of the services they provide. Complex applications will be constructed using multiple components from different vendors. System architects will assemble them by simply subscribing to the functionality required by the application.

The use of the following three XML vocabularies is generally accepted as a requirement to implement a web services component architecture:

1. The Simple Object Access Protocol (SOAP) defines the format of messages used when communicating between web services components. It consists of a message envelope, which indicates the contents of the message and how it should be handled, encoding rules for different data types enclosed, and how to use them to request data or an operation from the remote service [W3C-SOAP]. At a very basic level a SOAP message can be viewed as web form submission, the envelope is analogous to the HTTP POST method, the remove procedure call result is analogous to the HTTP response, and the data type is analogous to the CGI "querystring".

© SANS Institute 2001, As part of the Information Security Reading Room. Author retains full rights.

2. The Web Services Description Language (WSDL) is "…an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information…"[W3C-WSDL]. It is used to create a "contract" for a web service. This means that if you send a message in the format specified to the URL/URI specified, you will receive a response of a specified format. Web service consumers use this description to format the messages that will be sent to the web service.

3. Universal Description, Discovery and Integration (UDDI) provides both a way to describe what web services an organization offers and an online repository to hold that information [UDDI]. The UDDI repository is analogous to an online phone book that computers may search to find organizations and services. It stores contact information, industrial categorizations, and technical information about services make available by organizations listed.

Web services protocols will be used over TCP/IP networks using the HTTP/HTTPS protocol. Its REQUEST/RESPONSE model, and ubiquity, makes it the most useful for invoking remote procedures. Other protocols such as SMTP and proprietary messaging protocols may also be used, but will not be covered in this paper.

Web Service Security Challenges

The evolution of the web services paradigm brings new security challenges to organizations that use the Internet to conduct business. Unprotected web services are vulnerable to the following types of attacks:

- Reconnaissance – One of the attractive features of web services, the ability to advertise and document the services an organization offers, also provides attackers with useful information. In addition to gleaning information from WHOIS databases and DNS servers, potential attackers will be able to query UDDI registries for lists of organizations offering services and the URLs for accessing those services. The WSDL documents for those services, available from the repository, will provide detailed information on how to query a service and what kind of output to expect.

- Denial of Service – There are several possibilities for DoS attacks to negatively impact web services. Identity and advertising of services depends on being able to find them in the UDDI repository. If the repository machines become inaccessible it will make it impossible for new web service consumers to locate providers. Individual web services themselves may also be vulnerable to denial of service attacks. The UDDI repository will conveniently list URLs that operate as the end-points of these services and they can be flooded with invalid requests.

- Integrity Attacks – Since our definition of a web service is that of a component that can be combined with other components, if one component can be hijacked and spoofed it will taint the data and operation of the rest of the application. If done in a sophisticated manner the service's consumers may not realize that there is a problem.

© SANS Institute 2001, As part of the Information Security Reading Room. Author retains full rights.

- Bypassing of Firewalls – as outlined in Bruce Schneier's June 2000 Crypto-Gram [Crypto-Gram], one of the web services benefits touted by vendors is that complex queries can be made through corporate firewalls because they pass through service ports that are often held open. Sloppily implemented services can be exploited to compromise systems inside the firewall.

- Unintended software interactions – Web services are complex in nature. They receive complex queries and reply with complex results. A body of important lessons learned does not yet exist to help define industry best practices.

- Immaturity of the Platform – as noted previously, web services are brand new and the standards for enabling security are still being formed. Using them while they are still in the design phase is a risk to system stability.

Counter Measures

The following counter measures should be employed to protect web services:

- Enforce Trust Relationships – Each side of a web services transaction should be trusted and accountable to the other. Enforcing this requirement is difficult because the web services paradigm puts value into being able to create new relationships on the fly, but there is currently no vendor neutral platform standard for exchanging authentication and authorization information. New standards like SAML and XACML, noted below, are on the way to solve this problem, and industry leaders like Microsoft and Sun are working on systems to provide a federated identity.

- Encrypt Transport Links – Requiring traffic to travel over encrypted links can prevent transaction snooping. SSL/TLS encryption is already widely used to encrypt HTTP traffic. SSL/TLS can be used to also enforce point-to-point trust relationships because each side certifies the keys used to encrypt data. The SAML specification also provides for encrypting the SOAP message content itself when being transported over unencrypted links.

- Engineer Secure Components – The input expected by a web service will usually arrive as an XML document. The component will operate programmatically on this data to generate a response back to the client. The component should be able to receive data of an unrecognized format and handle it correctly to prevent buffer overflows. A proper error response should be returned to the requestor, as well. The faster an attacker can determine that a component has been hardened, the faster she or he will move to another target.

- Perform Regular Tests on Components – It's a good idea to run regular unit tests on web service components. Unit tests were likely created when the component was engineered and the component should still pass after being deployed. If components start failing tests it could be a sign that they've been replaced by malware versions.

© SANS Institute 2001, As part of the Information Security Reading Room. Author retains full rights.

- Reconcile WDSL Specifications with Actual Operation – The WSDL specifications placed in UDDI repositories will change as components are added, updated, and removed. Regular reconciliation should be performed to make sure that the specifications in the repositories match the actual component deployments. Old components should be removed as soon as they are removed from the directory. If the component has fallen out of favor due to a security problem, it is possible that an attacker someplace knows this. He or she may come back and attempt additional compromises. The published list of services should also be matched with the log files on component servers to identify any rogue components not otherwise accounted for.

- Use HTTP Proxy Filters – The use of HTTP proxies with XML parsing capabilities can reduce the risk of exposure through open HTTP ports on firewalls. The proxy's parser can validate each request and drop any that do not conform to the published service descriptions.

- Configuration Management – Configuration management practices should be employed when managing information in public UDDI repositories. That information resides on servers outside organizational control and needs to be protected. If an attacker is able to change the public data for a service it could compromise the integrity of other services that rely on it. Configuration management practices should also be applied to component server configurations as a way to prevent internal attackers from placing surreptitious components on servers that can be used for back door access.

This list is certainly not exhaustive. Many more will be developed as compromises occur, are investigated, and handled by incident response teams.

Technology Solutions

The following technical solutions have been developed, or are under development, to help implement some of the counter measures outlined above. Unfortunately many of them have not been finalized as standards and are subject to change.

- Security Assertion Markup Language (SAML) – an evolving standard under development by OASIS, it defines a format for transferring security assertions between components. A security assertion is a "statement of fact" that can be tested. In this case the fact is the identity of the sender. It provides a "single sign-on" capability to clients of web services [OASIS-SAML].

- eXtensible Access Control Markup Language (XACML) – a proposed standard by OASIS, will integrate access control policies into SAML messages. This means that an assertion of rights can be delivered with the assertion of identity [OASIS-XACML].

- XML Signature – a W3C Proposed Recommendation, it provides a format to digitally "sign" the content of web services messages, guaranteeing their authenticity [W3C-XMLDSIG].

© SANS Institute 2001, As part of the Information Security Reading Room. Author retains full rights.

- XML Key Management Specification (XKMS) – a W3C Note, offered by Verisign, Microsoft and webMethods, "specifies protocols for the distribution and registration of public encryption keys"[W3C-XKMS]. It is to be used to link the XML Signature standard to an existing public key infrastructure.

- Kerberos – an IETF standard originally developed at MIT, it provides single sign-on capability for network users and services. It does this by providing a "ticket" to the user when they first sign on. This ticket is then used to get service tickets to use to access different network services. Unlike many of the other standards listed here, it has been in widespread use for years [MIT-Kerberos]. Microsoft operating systems, starting with Windows 2000, use the Kerberos protocols for authentication [MS-Kerberos].

- Lightweight Directory Access Protocol (LDAP) – an open standard derived from the OSI Directory Access Protocol, it defines an API for querying directories [OpenLDAP]. Directories are hierarchically organized databases that contain profile information for users, computing resources, and access control lists. LDAP is the central configuration repository for systems from Microsoft, Novell and Netscape/IPlanet.

Going Forward

Software architects and system administrators are faced with the difficult job of preparing for the coming era of web services. Current security practices must be maintained, as web services operate on top of the existing network infrastructure. They will not supplant existing services, web and email, for the foreseeable future, if ever.

It is recommended that architects and administrators keep up with the evolving standards, as well. The pace of change in many organizations, even in the current economic climate, is increasing. The best way to deal with change is to prepare for it. The research and standardization activities of the following organizations should be monitored regularly:

- The World Wide Web Consortium (http://www.w3c.org) - The W3C is responsible for web standards. Its activities are organized into five domains. The domains of most interest to security professionals are: Architecture, and Technology and Society.

- The Internet Engineering Task Force (http://www.ietf.org) - The IETF is responsible for the architecture of the Internet. It maintains working groups in eight areas. The area of most interest to security professionals is the Security area, though some familiarity with all areas is a good idea. Each working group has a description and a list of goals and milestones that can be reviewed to get a quick overview of activities.

- Organization for the Advancement of Structured Information Standards (http://www.oasis-open.org) - OASIS is an industry consortium that develops and promotes interoperability standards.

Organizations should also monitor the evolution of vendor "enhancements" to standards that may affect them. Microsoft Corporation, in particular, has a history of extending standards to enable new functionality in their products.

© SANS Institute 2001, As part of the Information Security Reading Room. Author retains full rights.

References

[ORA-Primer]

Vasudevan, Venu. "A Web Services Primer." O'Reilly XML.COM. 4 Apr. 2001.
URL: http://www.xml.com/pub/a/2001/04/04/webservices/index.html (10 Nov. 2001).

[W3C-SOAP]

Box, Don, et al. "Simple Object Access Protocol (SOAP) 1.1." W3C Note.
8 May 2000.
URL: http://www.w3.org/TR/SOAP/ (10 Nov. 2001)

[W3C-WSDL]

Christensen, Erik, et al. "Web Services Description Language (WSDL) 1.1." W3C Note.
15 Mar. 2001
URL: http://www.w3.org/TR/wsdl (10 Nov. 2001)

[UDDI]

"UDDI Executive White Paper." uddi.org. 6 Sep. 2000
URL: http://www.uddi.org/pubs/UDDI_Executive_White_Paper.PDF (10 Nov. 2001)

[Crypto-Gram]

Schneier, Bruce. "SOAP", Crypto-Gram Newsletter. 15 Jun. 2000
URL: http://www.counterpane.com/crypto-gram-0006.html (15 Nov. 2001)

[OASIS-SAML]

Platt, Darren. "Oasis Security Services Use Cases and Requirements." Oasis SSTC. 30 May 2001.
URL: http://www.oasis-open.org/committees/security/docs/draft-sstc-saml-reqs-01.pdf (12 Nov. 2001)

[OASIS-XACML]

Lockhart, Hal. "OASIS XACML: Online Application Server Use Cases." Oasis XAMLC TC. 27 Nov. 2001.
URL: http://www.oasis-open.org/committees/xacml/docs/OnlineServerUseCases.doc (13 Nov. 2001)

[W3C-XMLDSIG]

Bartel, Mark, et al. "XML-Signature Syntax and Processing." W3C Proposed Recommendation. 20 Aug. 2001
URL: http://www.w3.org/TR/xmldsig-core/ (10 Nov. 2001)

[W3C-XKMS]

Ford, Warwick, et al. "XML Key Management Specification (XKMS)." W3C Note. 30 Mar. 2001
URL: http://www.w3.org/TR/xkms/ (10 Nov. 2001)

[MIT-Kerberos]

© SANS Institute 2001, As part of the Information Security Reading Room. Author retains full rights.

Tung, Brian. "The Moron's Guide to Kerberos, Version 1.2.2." 19 Dec. 1996.
URL: http://www.isi.edu/gost/brian/security/kerberos.html (12 Nov. 2001)

[MS-Kerberos]

"Single Sign-On in Windows 2000 Networks." Microsoft Technet Whitepaper. May 1999.
URL: http://www.microsoft.com/TechNet/prodtechnol/windows2000serv/evaluate/featfunc/nt2ksso.asp (15 Nov. 2001)

[OpenLDAP]

"OpenLDAP 2.0 Administrator's Guide." The OpenLDAP Project. 15 Sep. 2000
URL: http://www.openldap.org/doc/admin/index.html (11 Nov. 2001)

© SANS Institute 2001, Author retains full rights

Last Updated: May 19th, 2013

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS Malaysia @ MCMC 2013	**Cyberjaya, MY**	**Jun 03, 2013 - Jun 08, 2013**	**Live Event**
SANS Pen Test Berlin 2013	**Berlin, DE**	**Jun 03, 2013 - Jun 08, 2013**	**Live Event**
Industial Control Systems Security Training - Houston	**Houston, TXUS**	**Jun 10, 2013 - Jun 15, 2013**	**Live Event**
Security Impact of IPv6 Summit 2013	**Washington, DCUS**	**Jun 14, 2013 - Jun 16, 2013**	**Live Event**
SANSFIRE 2013	**Washington, DCUS**	**Jun 14, 2013 - Jun 22, 2013**	**Live Event**
SANS Canberra 2013	**Canberra, AU**	**Jul 01, 2013 - Jul 13, 2013**	**Live Event**
SANS London Summer 2013	**London, GB**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
Digital Forensics & Incident Response Summit 2013	**Austin, TXUS**	**Jul 09, 2013 - Jul 16, 2013**	**Live Event**
SANS Rocky Mountain 2013	**Denver, COUS**	**Jul 14, 2013 - Jul 20, 2013**	**Live Event**
SANS Mumbai 2013	**Mumbai, IN**	**Jul 22, 2013 - Jul 27, 2013**	**Live Event**
SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification	**Washington, DCUS**	**Jul 22, 2013 - Jul 26, 2013**	**Live Event**
SANS San Francisco 2013	**San Francisco, CAUS**	**Jul 29, 2013 - Aug 03, 2013**	**Live Event**
SANS SEC 560: Network Penetration Testing @ Bangalore 2013	**Bangalore, IN**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
SANS Boston 2013	**Boston, MAUS**	**Aug 05, 2013 - Aug 10, 2013**	**Live Event**
Critical Security Controls Summit	**Washington, DCUS**	**Aug 12, 2013 - Aug 18, 2013**	**Live Event**
Industrial Control Systems Security Training - DC	**Washington, DCUS**	**Aug 12, 2013 - Aug 16, 2013**	**Live Event**
Mobile Device Security Summit 2013	**OnlineCAUS**	**May 30, 2013 - Jun 06, 2013**	**Live Event**
SANS OnDemand	**Books & MP3s OnlyUS**	**Anytime**	**Self Paced**