Guide to XML Web Services Security

XML Application Firewalls for
Securing and Monitoring
XML Web Services

XML Web Services Are Revolutionizing the IT Industry

XML and Web Services are simple, but powerful standards that enable applications to more efficiently communicate with each other. Similar to how Internet-based technologies enable person-to-person (email) and person-application communication (web pages), XML Web Services is revolutionizing application-to-application communication. When you send an email, you need not worry about what technologies are being used to send and receive the email. As long as various simple standards are followed, you are ensured the message will be readable by the receiving party regardless of whether they are using Outlook, Unixmail or Yahoo Mail on any platform.

XML Web Services enables applications to communicate more effectively without having to work out the underlying mechanics of the communication. However, Web Services standards do not completely address security for XML Web Services. This guide provides you with a quick overview of the security issues related to XML Web Services, what standards are currently in the works and how you can secure communication today.

Why are Web Services Potentially Dangerous?

The use of XML and Web Services can pose serious risk if security is not properly addressed from the beginning. Current security schemes must be updated to handle the new class of communications that Web Services enables. Current technologies can be used to secure highly controlled Web Service networks but are not able to scale to mission critical environments.

Web Services Expose New Security Risks	
Network firewalls do not provide protection	Web Services are **designed to tunnel through firewalls**, evading their usefulness at the application layer. **Firewalls are intended for network level** security anyway**, and not application-level security**
Web Service interfaces expose more functionality	Web Service interfaces are application interfaces and can **expose much more functionality** to both external and internal threats
Web Service interfaces are complex and heterogeneous	The many types of Web Service interfaces enable **new types of security breaches.** Mainframes, desktop applications, .NET, SunOne, packaged apps can all be communicating via Web Services simultaneously
Web Service networks are peer-to-peer with decentralized control	Web Service networks are often peer-based with decentralized administration, **hampering standardized and comprehensive visibility and control**
Technologies and standards are fast moving and ever-changing	**Standards are evolving and ever-changing** with little unification**. Multiple standards are often in use simultaneously.** Technology managers don't want to be forced to "bet the farm" on one particular standard.
Web Service networks are dynamic	Web Service networks are dynamic and grow organically which **increases the cost of management and administration.**

Copyright © 2003 Westbridge Technology

Standard Security for Web Applications Apply to Web Services

The four basics security requirements for web-based communication: authentication, authorization, confidentiality and signature support are also fundamental to the secure communication of XML Web Services. The existing infrastructure for web security can be leveraged for XML Web Services to provide security for XML messages.

Basic Security Is Needed For Web Services	
Authentication	Authentication is verifying the identity of the sender or receiver. Credentials are embedded in either the headers or body of the SOAP message. Standard Web technologies using passwords, X.509 certificates, Kerberos, LDAP and Active Directory can be used to authenticate service requestors. Both service requestors and providers should be authenticated for sensitive communication. Even WSDL file delivery should be authenticated as WSDL files can be spoofed.
Authorization	Authorization is critically important because Web Services can introduce complex levels of access. In addition to authorizing what information users/applications have access to, there also needs to be authorization of which operations an application or user has access rights to perform.
Data Privacy, Confidentiality	Standard SSL encryption using HTTPS allows point-to-point data privacy between service requestors and service providers. However, in many cases, the service provider may not be the ultimate destination for the message. A service provider may act as a service requestor, sending pieces of information to multiple services. The XML Encryption standard permits encryption of portions of the message allowing header and other information to be clear text while leaving the sensitive payload encrypted. Sensitive information can then be left encrypted to the ultimate destination, allowing true end-to-end data privacy.
Data Integrity	Digital signatures can be used to verify if a message has been tampered with. A service requestor can sign a document with the sender's private key and send it along with the payload of the message. The service provider can then verify the signature with the sender's public key to see if any portion of the document has been compromised. Thus systems can ensure data integrity when communicating with each other. The XML Signature standard provides a means for signing parts of XML documents, providing end-to-end data integrity across multiple systems.

Who should care?

Chief Information Officer

Web Services enables initiatives such as the real-time enterprise, straight through processing and migration to services-oriented architectures (SOA). The CIO must ensure security, reliability and interoperability with minimal cost and maximum flexibility to support current and future standards.

Chief Security Officer

Web Services encourages decentralized development and yet there must be minimum security throughout the entire organization and extended network. Centralized and standardized application-level security and enterprise-wide enforcement of security policies with minimal cost and disruption are required

Application Architect/Developer

Security needs to be built in from the start and not added in as an afterthought. Architects need to focus on building business applications and not on mechanisms to implement security policies.

IT Operations

Decentralized applications with Web Services increase the complexity and load of IT Operations management and greatly affect the overall reliability of the Web Service network.

Auditor / Compliance Officer

As in any secure environment, provability is critical for security management and governmental compliance. Complex, decentralized data networks pose an auditing and reporting challenge that requires sophisticated content inspection, data collection and tracing of transactions across multiple services.

However, Web Services Introduce New Security Issues

While XML Web Services utilizes the same transport mechanisms as most Web-based traffic, a typical XML network requires additional security to ensure minimum protection levels.

Interoperability	Because Web Services enables much easier integration with 3rd parties, including suppliers, customers and partners, authentication and access rights must be tightly controlled and kept up-to-date. However, because multiple parties are involved, it is often difficult or impossible to standardize on one authentication and access control scheme. In particular, B2B exchanges have the extra challenge of managing multiple formats. In the extreme case, every service would need a separate credential for each service accessed. Single sign on and credential mapping solutions can help make these environments easier to administrate and easier for participants to use.
Scalability	Any security solution must be able to scale from a performance standpoint. With new standards such as WS-Security requiring processor intensive functions such as signature verification and decryption built into the network, bottlenecks can occur. Specialized hardware for performing cryptographic functions can be used to offset the load and can be located at each Web Service or at a proxy to share processing across multiple services. An additional scaling issue is management and administration scalability. Being able to effectively manage security policies across a number of combinations of service requestors and Web Service interfaces is an often forgotten requirement that should be built in early.
Centralized Management	Single sign-on plays an important role in Web Services environments. Diverse systems need to communicate with each other and it is impractical for each system to maintain each other's authentication rights and access control lists. One solution is to give everybody the same credential, however, that presents a serious problem when one member becomes untrusted. New credentials must be sent to all remaining valid members. Requiring a separate credential for each service is difficult to administer when a user needs to be revoked. Each system the user needs to be revoked from, can have different authentication and authorization implementations. In addition, it is difficult to fully ensure that the user no longer has access to all of the systems. Single sign-on solutions help solve this problem by allowing credential mapping among many diverse systems. Each Web Service may then deal with the credential system that they are accustomed to. This can lower administration cost and help ensure data protection. The
Malicious Attack	SOAP interfaces are software API's and can expose much more functionality. A packaged application for instance may have hundreds or thousands of critical operations exposed, all accessible through port 80. In addition, an attacker has more information available to them. WSDL files and UDDI entries can provide detailed information that enable a hacker to gain entry. The message format is in XML format, which is self-describing and clearly show the data elements. While attacks on Web Services will become more sophisticated, more information is available to security systems to detect and deter problems

Some Possible Malicious Attacks Against Web Services

In addition to the standard forms of attacks that network firewalls address, Web Services introduces new forms and flavors of attacks that in general flow through Port 80 of the firewall.

Denial of Service	Web site technologies are well understood for detecting standard denial of service activities. Web Service interfaces are much more heterogeneous and require more knowledge to protect. For instance, a Web Service that provides simple information may be able to comfortably handle 1000 requests per second however, a loan approval application may only be able to handle at most 5 requests per hour. Sending a loan approval interface 10 requests per hour may constitute a denial of service attack but be undetectable by normal means, such as firewalls. Understanding and collecting data should help provide profile information on each service so that they can be protected from denial of service attacks.
Replay Attack	Similar to Denial of Service, replay attacks involve copying valid messages and repeatedly sending them to a service. Similar techniques for detecting and handling Denial of Service can be applied towards replay attacks. In some ways, replay attacks are easier to detect with Web Services because payload information is more readily available. With the right tools, patterns can be detected more easily even if the same or similar payload is being sent across multiple mediums like HTTP, HTTPS, SMTP or across different interfaces.
Message of Application Buffer Overflow	With XML Web Services, information about data parameters are exposed. In addition, much more data is likely to be sent between systems, creating the opportunity for buffer overflow attacks. For example, an attacker can send a parameter that is longer than the program can handle, causing the service to crash or for the system to execute undesired code supplied by the attacker. A typical method of attack is to send an overly long request, for instance, a password with many more characters than expected. Many legacy systems that will be Web Service-enabled are designed for controlled, well behaving requests and may not be prepared to handle unusual requests. Similar to buffer overflow attacks, hackers often send malformed content to produce a similar effect. Sending in strings such as quotes, open parentheses and wildcards can often confuse a Web Service interface.
Dictionary Attack	Many systems have weak password protection and Web Service interfaces are no different. However, unlike portals, XML Web Service interfaces are heterogeneous in nature with each system having its own authentication system and methods for deterring undesired behavior. Dictionary attacks are common where a hacker may either manually or programmatically attempt common passwords to gain entry into a system or multiple systems. Administrators should ensure that passwords are difficult to guess and are changed often. Unlike standard user credentials, application credentials are determined by the administrator. Password strengthening rules that are desirable for users should also apply to administrators of Web Service interfaces.
SQL Injection	SQL Injection Attacks involve adding special characters or terms to cause SQL statements to return unintended data. For example, strings that may end up in a WHERE clause of a SQL statement may be tricked into including more information. For instance a parameter value of: X' OR 1=1 may cause the whole table to be returned because 1=1 is always TRUE.
Cross-Site Scripting	Cross-site scripting involves inserting code into a field or URL that gets executed and hands over control or sensitive data to the attacker.
Virus Detection	One challenge with encryption is virus detection. Web Service traffic may include attachments. When content is encrypted, viruses that may be a part of the message are also encrypted. This makes it difficult for a virus checking program to detect malware. When using encrypted data, virus checking can be performed at the destination or by an intermediary that can decrypt the data to be virus scanned and reencrypted for transport.

Copyright © 2003 Westbridge Technology

Emerging Web Services Standards Represent Part of the Picture

Of course existing standards such as LDAP, PKI and SSL play an important role in securing Web Services. To handle the special needs of security for Web Services, numerous additional standards are being introduced, some of which are covered here.

Some Promising Web Services Security Standards	
SAML	Security Assertion Markup Language is being developed by the W3C and is a protocol for asserting authentication and authorization information. SAML compliant servers can be accessed for authentication and authorization data in order to enable single-sign on.
XKMS	XML Key Management Specification is a protocol developed by the W3C which describes the distribution and registration of public keys. Services can access an XKMS compliant server in order to receive updated key information for encryption and authentication.
XML Encryption	XML Encryption is a protocol developed by the W3C which describes the encryption of digital content. The XML Encryption standard includes protocols for encrypting sections of XML documents. XML Encryption enables end-to-end encryption because the actual message is encrypted. Session-level encryption can only provide data privacy between two servers.
XML Signature	XML Signature is a protocol developed by the W3C which describes the signing of digital content. The XML Signature standard includes protocols for signing sections of XML documents. XML Signature enables such capabilities as message integrity.
Some Upcoming Standards	
WS-Security	WS-Security is being developed by Microsoft and IBM in order to provide core facilities for protecting the integrity and confidentiality of a message as well as mechanisms for associating security-related claims with the message. Currently, WS-Security describes how to attach signature, encryption and security tokens to SOAP messages
XACML	Standard used to describe access rights policies and is being driven by OASIS. XACML represents authorization and entitlement information.
Liberty Alliance Project	An alliance formed to deliver and support a federated network identity solution for the Internet that enables single sign-on for consumers as well as business users in an open, federated way. The Liberty Alliance is led by Sun and other industry players.
HTTP-R	Standard for reliable messaging being developed by IBM.
XrML	eXtensible rights Markup Language is a universal method for securely specifying and managing rights and conditions associated with resources including digital content and services.
.NET Passport	A Microsoft-based initiative that provides a suite of Web-based services that help make Internet and purchasing online easier and faster. .NET Passport provides users with single sign-in (SSI).

Security today, Security tomorrow

Web Services communications can be secured today using legacy technologies. Dedicated links or SSL can be used to ensure data privacy of communication and Web Service networks. Existing authentication directories such as LDAP or X.509 certificates can be used for authentication and access control. However, these solutions only work in a controlled and simple environment and do not scale from a performance or manageability standpoint. Web Services is all about piecing together different systems, even across different enterprises. Security interoperability is expensive to maintain using legacy technologies. To provide true end-to-end security for these fluid and dynamic environments will require a mix of technology and standards use. Building in scalable security from the beginning is critical. Building in security after the fact is similar to the range of problems of building quality in later.

Where Do I Build the Security?

Enterprises often have to choose to build the security in an shared services layer or to build it within each application node (such as at each application server). Each approach has its own advantages and disadvantages.

	Advantages	Disadvantages
Security at each Web Service	❖ Low initial barrier to entry ❖ Application servers will build in some security ❖ Existing legacy tools are available to use	❖ Difficult to add to legacy code and packaged applications. ❖ Difficult to implement and maintain consistent policies across all nodes. ❖ Difficult to prove that a given security policy has been implemented. ❖ Expensive to replicate security at each Web Service, particularly as Web Services move to the desktop
Security as a shared service	❖ Costs less to maintain interoperability ❖ Centralized visibility of Web Service network ❖ Centralized management and control ❖ Ability to leverage best of breed crypto accelerators	❖ Extra hop in the network ❖ Solution must support multiple standards and technologies ❖ Potential bottleneck if non-scalable solution

Similar to how Network Firewalls have moved as a shared service to protect multiple Web pages, many analysts believe application security will move to a shared service rather than being at each service.

Copyright © 2003 Westbridge Technology

Top 10 Requirements for Web Services

The drive to get business advantage from XML Web Services will cause turbulent times for IT managers. To successfully navigate these new issues, managers must change their mind set from “fragmented security systems focused on using network perimeter to shield closed business systems” to “consistent managed security systems focused on managing application level security for inherently distributed, complex business systems”.

While many pilot projects that are currently underway have only a handful of security requirements, ensuring that these networks are secure as they grow in usage is essential to creating an cost-effective and secure environment down the road. That being said, there are a number of requirements that are essential to build into the system early in the development process in order to ensure that security is an enabler of secure communications rather than a hindrance to the project.

Top 10 Requirements For Effective Web Services Security

- Authentication
- Authorization, Access Control
- Encryption via SSL, XML Encryption or dedicated lines
- Signature Support, XML Signature
- Malicious Attack Protection for Web Service interfaces
- Monitoring, alerts, threat containment, intrusion detection
- Auditing for proving security
- Interoperability of different security, data and transport schemes across departments and companies
- Sophisticated policy enforcement including content filtering
- Scalability of secure message throughput as well as scalability in management and administration of security

For more information, including access to further whitepapers, please visit the Westbridge Technology site at http://www.westbridgetech.com

About Westbridge Technology

Westbridge Technology is a leader for securing and monitoring XML and Web Services networks. Westbridge Technology’s flagship product, the XML Message Server, is infrastructure software that provides enterprises with unified security, reliability and manageability for mission-critical XML Web Services environments.

westbridge

Westbridge Technology, Inc.
215 Castro Street
Mountain View, CA 94041
Phone: 650-210-0700
Email: info@westbridgetech.com
Web: http://www.westbridgetech.com