

# White Paper:

## Securing XML Web Services: XML Firewalls and XML VPNs



## Table of Contents

The Challenge of XML and Web Services Security .....	1
XML Firewalls: A First Step .....	2
XML VPNs: Fast, Flexible Partner Enablement.....	3
Putting It All Together .....	4
About Layer 7 Technologies .....	5
Contact Layer 7 Technologies .....	5
Legal Information .....	5

## The Challenge of XML and Web Services Security

By providing a flexible, platform-neutral way for rendering diverse data types, XML has become a standard for exchanging information across heterogeneous applications. Web services, a set of XML based protocols for finding and communicating between loosely-coupled, internet callable application “services” have therefore become the preferred mechanism for integrating heterogeneous applications and enabling Service Oriented Architectures (SOAs).

Standardizing on XML and Web services for data exchange and integration provides significant IT benefits including flexibility, interoperability and reach. However, it also introduces new kinds of security challenges.

- Web services can be transmitted over any transport protocol including common Web protocols like HTTP. This makes it easy for Web services to bypass network firewalls.
- Web services expose business functionality through open API’s requiring new application aware security measures.
- Web services enable multi-hop composite applications requiring message level security and audit that can span multi-hop SOA transactions end-to-end.
- XML based messages can be deliberately or inadvertently malformed to cause parsers or application break creating new XML threat and vulnerability protection requirements.
- Web services transactions are principally machine-to-machine necessitating new thinking around machine-to-machine trust enablement and credentialing.
- Web services and their client applications must agree on security parameters (like crypto preferences and standards support) before they can successfully exchange data creating a need for new kinds of policy coordination.

Traditional security measures like network Firewalls and VPNs are not sufficient to address these new security challenges. Network Firewall’s are not service or application aware and therefore can’t regulate access based on service or service feature like operation type. Network firewalls also can’t protect against XML borne threats in a message or message attachment since they lack the ability to inspect XML messages, validate XML structures or detect anomalous XML content. Similarly, network based VPNs whether SSL or IPSec can’t preserve a message’s integrity and privacy as it gets passed across service hops in a SOA transaction. Moreover VPNs can’t provide message level audit trail or non-repudiation across a SOA transaction. As a result up until recently the only option for implementing application level XML and Web services security has been to program security directly into the application based service.

Coding security into a Web service however requires developers to understand how to implement emerging WS-\* standards like WS-Security, WS-SecureConversation, WS-Trust, WS-Federation, and WS-Policy, to name some, on both the Web services and its clients. It requires Web services coders and client developers to coordinate security preferences through

out-of-band mechanisms since a Web service can't communicate security expectations or capabilities to clients automatically. And if a Web service's security needs to be integrated with existing trust infrastructure like PKI, Directory, Single Sign-on (SSO) and Identity Federation, programmers will need to implement one-off integrations on both the service and client application. For most situations, programming XML and Web services security will therefore lack the consistency, flexibility, scalability and deployment speed end-users require.

As a result, two new classes of security infrastructure have emerged to try and satisfy customer demand for purpose-built XML and Web services security on both the service provider and client. To deal with the complexity of Web services security management and enforcement on the Web services provider side of an integration, end-users can now avail themselves a new class of XML security infrastructure often referred to as an XML Firewall or Web services Gateway. An XML Firewall or Web services Gateway like Layer 7's SecureSpan Gateway is a dedicated device or piece of software that can be implemented in a DMZ or data center to enforce XML and Web service security preferences around access control, credentialing, integrity, privacy, threat mediation and audit. In some cases like the XML Firewall from Layer 7, they can also perform hardware accelerated data transformation, routing, SLA and other policy operations. In all cases, XML Firewalls allow security administrators to define security policies for XML and Web services transactions and enforce them centrally without programming.

XML Firewalls are a necessary first step in securing Web services, however for some scenarios there is a further requirement to automate security on the client application through an XML VPN. When Web services are shared across security and identity domains or when the client application is a portal there is often a requirement to reconcile identity domains, provision PKI for certificate based trust, integrate with an existing Single Sign-on infrastructure, enable non-repudiation and manage policy change between a Web service and client application. Doing this manually while possible is complex. For this reasons some vendors like Layer 7 also offer an XML VPN product for automating client security and coordination.

This White Paper examines how and when to deploy XML Firewall and XML VPNs to deliver total XML and Web services security for SOA based infrastructures.

## **XML Firewalls: A First Step**

Taking their cue from the Web world, technology vendors like Layer 7 technologies have developed XML-specific firewalls (like Layer 7's SecureSpan Gateway) to address the unique security challenges of XML and Web services. XML Firewalls like the SecureSpan Gateway are designed to examine and evaluate the XML content of the incoming traffic and, based on that evaluation, perform an appropriate security action. That action may require routing the message to a designated end point, transforming the message based on its content, validating a signature, decrypting a field, or blocking access to certain operations. In the case of the SecureSpan Gateway these operations are accelerated through specialized ASIC accelerators.

XML Firewalls typically resolve an incoming message to a specific target Web service either by examining the SOAP message header or, with native XML, the HTTP header. Once the target Web service is resolved, the XML Firewall can apply a stored security policy based on the target address, originating caller identity, message content, and in some cases, the successful execution of prior policies. Most XML Firewalls can also examine elements of the message body like fields, parameters, and attachments. As part of Web service lifecycle management, several XML Firewalls also auto-generate virtualized WSDL views of back-end target Web services to simplify versioning, addressing, and SLA-based operations.

Conceivably, almost any kind of message-level XML operation can be controlled and processed inside an XML Firewall. By assuming this burden for one or more shared Web services, application providers can centralize security provisioning and administration. This results in faster time-to-market for Web services deployments and improved flexibility under changing business conditions. But an XML Firewall only addresses half of the equation. While enforcing security for provider-side Web services, XML Firewalls fail to address the broader issue of managing security end-to-end across an integration. Blocking an unauthorized application or message from passing through an XML Firewall is obviously valuable, but without a corresponding mechanism to communicate security expectations to trusted client applications, there is no consistent way to ensure that the security applied on one side of an integration complies with security policies on the other side.

Although XML Firewalls can remove the need to program security into the Web services applications that they protect, they do nothing to help trusted client applications access those same protected Web services. Ensuring that security expectations are met by the client application requires out-of-band negotiation (through phone or email, for example), followed by independent client-side programming and comprehensive compliance testing. This high-touch process is slow, expensive, and prone to errors. Moreover, there is no timely way to communication and apply changed XML Firewall security policies to the client application.

XML Firewalls inherently do not address the challenge of security coordination. Managing security end-to-end across a Web services integration cannot be fully accomplished with an XML Firewall alone. Without some form of client-side coordination, essential operations like synchronizing cryptographic parameters between a client and Web service and provisioning client-side certificates and keys (to implement the WS-Security standard, for example) become tedious. Advanced applications like extending Single Sign-on to Web services, federating identities, and bridging Message Oriented Middleware islands become impossible. Any change in security policy on the XML Firewall will also break the integration with every authorized client application. Clearly true end-to-end XML Web services security requires more than just an XML Firewall.

## **XML VPNs: Fast, Flexible Partner Enablement**

XML Firewalls provide a critical element in a broader Web services security strategy, but are often not sufficient given the integrated nature of Web services transactions. Security

requirements between the services participating in a SOA transaction must be coordinated. Ideally, this coordination should be dynamic so that changes on one or more systems are automatically accommodated without developer involvement.

One possible security coordination model is to use an XML Firewall plus client-side technology for distributing security workload to and coordinating security preferences with client systems. Like VPN security, the client-side technology should be available as either software or hardware depending on deployment requirements. A purely developer-oriented option should also be available for customers uncomfortable with any client footprint. The client-side technology should also provide other value-added functions for a Web services transaction like Single Sign-on integration, PKI provisioning, federation coordination, non-repudiation and policy change management. The Web service provider-side and client-side components of this architecture could then coordinate their specific security preferences, terms, and conditions for a transaction by exchanging a virtual outline of a policy document. This would preserve the loosely-coupled nature of Web services by ensuring changes in policy in one system are automatically transferred to any others.

In conjunction with an XML Firewall, this type of client component can provide organizations with a security model spanning transactions both inside and outside traditional corporate security boundaries. Negotiating on-the-fly with an XML Firewall would not only save considerable developer effort and time, but would also remove the risk of errors and inconsistencies inherent in any programming-based security provision. While not a panacea, this type of two-way security model is potentially beneficial in many Web services integration scenarios.

The Layer 7 SecureSpan Bridge is an example of this kind of full featured XML VPN. Deployable on or in front of client applications, SecureSpan Bridge can seamlessly enable federated Web services, B2B and cross-domain Portal projects that leverage Web services as an integration framework.

## Putting It All Together

There is no one size fits all solution for Web services security. There will always be instances where access lists programmed into the Web services themselves are sufficient. Other times, SSL may be more than adequate for privacy and integrity. However, in scenarios where granular message processing and auditing is essential, dedicated XML and Web Services security technology like XML Firewalls and XML VPNs will prove necessary.

In those instances, an end-user should look for vendors that can deliver both XML Firewalls for defending access to Web services, as well as an optional client-side coordination component for enabling security across a Web services integration. This will ensure that XML Web services integrations are truly flexible and interoperable without compromising critical security or cost requirements.

## About Layer 7 Technologies

Layer 7 Technologies ([www.layer7tech.com](http://www.layer7tech.com)) helps enterprises realize secure, cost-effective business integration using XML and Web services. The Layer 7 SecureSpan set of products are designed to govern and accelerate Web service integrations spanning security and identity domains without expensive and inflexible programming. The benefit to business includes faster time-to-market, lowered integration costs and security consistency across federated departments and partners.

## Contact Layer 7 Technologies

Layer 7 Technologies welcomes your questions, comments, and general feedback.

**Email:**

[info@layer7tech.com](mailto:info@layer7tech.com)

**Web Site:**

[www.layer7tech.com](http://www.layer7tech.com)

**Phone:**

1-800-681-9377

## Legal Information

Copyright © 2004 by Layer 7 Technologies, Inc. ([www.layer7tech.com](http://www.layer7tech.com)). Contents confidential. All rights reserved. SecureSpan™ is a registered trademark of Layer 7 Technologies, Inc. All other mentioned trade names and/or trademarks are the property of their respective owners.