# Global Information Assurance Certification Paper

# GIAC (GSEC) Gold Certification

## Securing the Enterprise Service Bus: Protecting business critical web-services

Author: Michael Taylor, warriormonkster@gmail.com
Advisor: Jim Purcell

Accepted: April 17th 2009

Abstract

*Corporations and large businesses are becoming more and more dependent on automated processes and procedures to provide services to customers. These can provide speedy, reliable, accurate and cost-effective services for customers.*

*This can be achieved by using Web-services, B2B Gateways and other "connectors" both "internally" (Inside a LAN) and externally (Interconnecting 2 separate LAN's/organizations). These "connectors" typically connect high-value information systems and services to other high-value systems and services using purpose-built application layer protocols such as HTTP, SOAP to transmit XML messages. Examples of this may be:*
*- Web services used to federate two separate identity management systems. (e.g. using SAML)*
*- An internal CRM system needing to communicate and synchronize information with a Billing System.*

*There are many approaches to designing architecture to achieve this. Traditionally Security is often not considered when implementing an enterprise service bus. This is of concern as these services: - Are usually responsible for generating revenue - Are usually "highly trusted" by "high-value" systems and services. - Can be used to transmit highly sensitive data. My paper will briefly discuss Enterprise Web Services and the uses of Enterprise Service Buses, but will concentrate on potential threats and vulnerabilities to these and suggest suitable means to mitigate risks.*

# 1. Introduction

Web services and B2B links are becoming the backbone for Electronic Commerce. Web services enable fast, automated, inexpensive and reliable links for core business activities such as Rating and Billing: e.g. passing of rating information to Billing Systems, Provisioning: e.g. providing information from an order system to a dispatch system, Identity Federation: e.g. synchronizing identity credentials between two organizations and/or Integration of Legacy systems and services: e.g. passing key information from a legacy network service to a trouble-ticketing solution.

As a result Web Services will attract the interest of malicious parties to cause disruption or for financial gain.

Web Services mean many things to many people, for the sake of this paper web services consist of:

- Two or More autonomous Information systems or services requiring interconnectivity to achieve an outcome such as Rating/Billing, Provisioning, Federation, Integration etc.
- XML Messages being passed between two or more systems

## 2. An overview of Enterprise Web Services

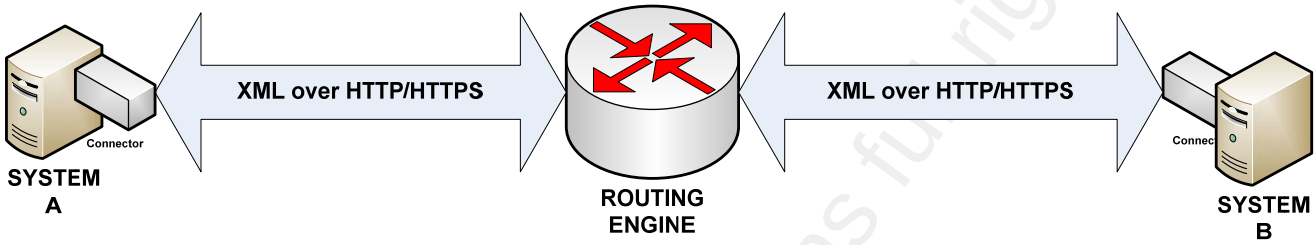


*Figure 1 - Web Service Components*

Web Services will have following components:

**A CONNECTOR:** This provides a gateway service from the System to the Routing Engine. It has the ability to provide the implement functions:

- Business rules and processes
- Security Controls
- Data Mapping and translation
- Protocol Management (SOAP)
- File transfer management
- Event Management (Logs, traps, events)
- Communications and Session management

**A ROUTING ENGINE:** This enables the connector to communicate with other systems and services. Sometimes this may be referred to as the *Enterprise Service Bus*. This component is very similar to a layer 2/3 switch in a network environment as its primary function to route Web Service Information from one connector to another.

# 3. Threats and Vulnerabilities to Enterprise Web Services

This section will aim to outline threats and potential mitigation relating to:

The **Confidentiality** of Web Services: Ensuring that web service information can be seen only by parties that have been authorized to see them.

The **Integrity** of Web Services: Ensuring that web service information can not be corrupted or tampered with by a malicious party.

The **Availability** of Web Services: Ensuring that web services will be available when necessary and provide appropriate levels of service.

## 3.1. Threats to the Confidentiality of Web Services

Web services may be used to transmit sensitive information between Systems and Services. This information may also transverse untrusted networks (e.g. Public Internet) which are exposed to potential malicious activity. If the information is deemed valuable by a malicious party it is highly likely it will be targeted for misuse. Information targeted for Confidentiality attacks may be:

**Customer Personally Identifiable Information**: This may be attractive information for a malicious party to enable activity such as Identity Theft and Identity Fraud. This information is highly likely to be subject to your local privacy laws.

**Valuable Company Intellectual Property**: Information such as financial reports and even source code may be attractive to a malicious third party.

**Billing/Rating information:** In a world where core communications such as VOIP are being extensively used call logs and billing information may be an attractive target for malicious groups.

### 3.1.1. Threat to Confidentiality: Information is disclosed to malicious party in transit:

Communications between Web Service Connectors and other Information Systems and Services may transverse public and/or untrusted networks. These communications may be vulnerable to: Eavesdropping (e.g. By use of mirrored port on network equipment, by use of network taps, by monitoring unsecured Wireless Communications), Man in the Middle Attacks (e.g. where tokens and certificates are used), Human error (e.g. Firewall rules poorly configured, weak cryptographic controls used)

### 3.1.2. Threat to Confidentiality: Information is disclosed by Connector to malicious party:

Connectors are subject to vulnerabilities and weaknesses like any other application. Connector interfaces and logic may be vulnerable to: Injection Attacks, Cross-site-scripting attacks, Buffer-overflow attacks, Human error (e.g. poorly configured access controls)

## 3.2. Mitigating risks to the confidentiality of Web Services

These security controls can mitigate risks to the confidentiality of Web Services:

**Domain Isolation/Segregation:** Ensuring that components are logically segregated and isolated into different security domains based on levels of trust. This segregation can occur on both the Network and Host. From a Network perspective this may involve the use of firewalls providing stateful packet inspection and network based access controls. Many vendors are offering XML/SOAP based firewalls nowadays that perform deep packet inspection of SOAP communications. From a Host perspective this may involve the use of virtualization for connectors, and hosting connectors on infrastructure separate to that of the information system using it.

**Strong Cryptographic Controls for Network Communications:** The use of strong cryptographic controls to assure the Confidentiality of Web Service Information transmitted across networks. This involves the use of protocols such as TLS (HTTPS) to provide authenticated and encrypted point to point communications. Web services provide the ability for connectors to use HTTPS as a transport for Web Service information. It is important to ensure that cryptographic processes and algorithms are strong and are not vulnerable to attack.

**Use of appropriate authentication:** The process of establishing and determining the validity of a claimed system or service. Authentication that can be used with web services range from username/passphrases to client and server side certificates. Transactions with a high confidentiality requirement may require more than one factor of authentication.

**Appropriate Access Controls:** Ensuring that Security Controls are implemented to ensure that Web Service Information is made available on a "needs to know basis." This also includes restricting access via web services to functions on a target host.

## 3.3. Threats to the Integrity of Web Services

Web services may be used to transmit information between Systems and Services that a core business process or activity is dependent on. There may be considerable impact to a business if this information was to be corrupted or tampered with. Information targeted for integrity attacks may be:

**Billing/Rating information:** Web Services may be targeted to conceal fraudulent or other illegal activity.

**Key information used for authentication:** Information used to federate Identity Management Systems may be targeted to plant back doors into systems.

### 3.3.1. Threat to Integrity: Information is modified or corrupted by a malicious party in transit:

Communications between Web Service Connectors and other Information Systems and Services may transverse public and/or untrusted networks. These communications may be vulnerable to a malicious party intercepting and modifying web service communications (e.g. conducts a man in the middle attack to modify communications)

### 3.3.2. Threat to Integrity: Information is modified or corrupted by a malicious party attacking a connector:

Connectors are subject to vulnerabilities and weaknesses like any other application. Connector interfaces and logic may be vulnerable to Injection Attacks and/or Human error (e.g. poorly configured access controls)

## 3.4. Mitigating threats to the Integrity of Web Services

These security controls can mitigate risks to the integrity of Web Services:

**Input Validation:** Ensuring that information processed, stored or transmitted by a Connector is clean, correct and useful. Typically this involves validating incoming data against an expected type, length, format, and range. Input validation can reduce risks that information can be deliberately corrupted.

**Robust authorization processes:** The process of allowing or denying access something before you can modify it. This control is different to Access Controls in that it involves the process of determining whether a party should have access to something rather than if it can have access. Web services can provide a lot of flexibility around the authorisation process. Some Web service implementations give the ability to orchestrate/automate the authorisation process for access based on attributes, predicates, or context. Attributes describe details about the party requesting access, predicates are conditions based on the web service environment that must be "true" before access is granted, and context is the actual state of the environment at the time of the access request. This control can be a good tool in assuring the integrity of web service information as it provides flexibility.

**Non-repudiation controls**: Ensuring that a party cannot repudiate, or refute their participation in a communication or transaction. Non-repudiation controls that can be used for Web Services may be the use of digital signatures for sensitive information or secured event logs.

## 3.5. Threats to the Availability of Web Services

The availability of Web services may be crucial to core business process or activities. *Availability is usually a business's greatest security concern with considerable financial impact as real-time web services may provide a direct revenue stream.* Availability attacks are also more likely to tarnish an organizations reputation due to the fact that they are public in their nature. (It is difficult for a business to conceal the fact a core service or system was made unavailable) Web Service activity targeted for integrity attacks may be:

- Billing, payment or Rating services: A malicious party may want hinder or prevent a business from being able to bill its customers.

- Customer Support services: A malicious party may want to hinder or prevent a business from being able to support its customer base.

### 3.5.1. Threat to Availability: Malicious Party targets Network Web Service Communications:

Communications between Web Service Connectors and other Information Systems and Services may transverse public and/or untrusted networks. These networks are likely to be shared by a number of parties.

These communications may be vulnerable to:

**Targeted Denial of Service (including Distributed Denial of Service) attacks**. (E.g. Malicious party targets essential Network Services: floods bandwidth in a network segment, or attacks/overwhelms a router providing networks services required for Web Services with a deliberate denial of service attack.)

**Denial of Service from other malicious activity.** (e.g. Upstream internet Service provider has limited bandwidth due to a worm outbreak, or another party is being targeted with a denial of service attack that shares the same network services that your web service depends on)

### 3.5.2. Threat to Availability: Malicious party targets a Web Service connector:

Connectors are subject to vulnerabilities and weaknesses like any other application. Connectors are attractive targets as they provide potential access to both the Information System they are connected to, and to upstream routing/service bus services. The connector may be vulnerable to:

**A direct denial of service attack on the connectors to overwhelm the connector itself.** (e.g. A Syn-flood style of attack where the connector is unable to manage the number of connections or sessions it has opened, or an attack on the connector application to exploit a vulnerability in the way it processes input)

**A denial of service attack on the Information System "behind" the connector**. This occurs when the connector processes and passes data through to the Information System at a rate higher than it can process.

**A denial of service attack on the Routing Engine/Service bus the connector is connected to.** (e.g. malicious party tries to overwhelm the routing engine in an attempt to attack it and other systems it is connected to)

## 3.6. Mitigating threats to Availability

These security controls can mitigate risks to the availability of Web Services:

Capacity Planning: Designing Web Service Systems, Networks and Services with enough capacity to process transactions even during times of excessive load. Having extra capacity will provide more time to react to a denial of service attack.

**Security Monitoring of key web service processes:** The areas that are vulnerable to a denial of service attack are typically the bottlenecks in your network. Elements such as Firewalls and routers are important to monitor, but of equal importance are web service connectors, routing engines/service buses, and back-end processes on Information Systems connected to connectors. It is a good idea to set thresholds based on the capacity planning (see above) you conducted when designing the web service. Although monitoring is reactive, it can be a very effective tool for limiting the damage of a denial of service attack.

**Network Rate limiting:** Putting processing limits on essential network services to ensure that web services are not vulnerable to a Denial of Service attack or event. Areas where you can effectively rate limit are Firewalls, routers, and even connectors. Many connectors have the ability to provide limits on the number of sessions they will manage.

**Application Rate Limiting:** Putting processing limits on application processing to ensure that web services are not vulnerable to a Denial of Service attack or event. Many connectors provide the ability to limit and manage the amount of application processing that will be done for a particular session or connection. *Although the author of this document does not encourage implementing security controls within the routing/service bus – effective baseline rate limits can be very effectively applied to processing within the routing engine/service bus.*

## 4. Other security controls

Alongside the listed Security Controls to mitigate risks to confidentiality, integrity and availability of web services, the following controls are also recommended:

Auditing/Logging: Determine which security events are of interest and ensure they are captured. These security events of interest may require the correlation of logs from multiple systems and services. (e.g. Collector + Routing Engine + Firewalls)

Pre-production Penetration Testing: Pre production testing by a competent, independent third party tester is recommended for high value web services. The tester should show competence not only in testing networks and operating systems but also have experience with SOAP, XML and understand the concepts behind web services. *Alongside standard vulnerability testing, it is valuable to instruct the penetration tester to behave "as an attacker attempting to . . ."This is a great way of determining whether the security controls are appropriate and if you have defense in depth.*

Policies, standards and procedures: All of these encourage good security processes and behaviors'. Ensure that appropriate policy is in place to govern the way third parties are connected with. Ensure that security incident response procedures are developed before go-live for web services.

## 5. Summary and recommendations

Web services provide new and interesting problems for security professionals. As more people come to use and rely on online services the use of web services is more than likely going to spread.

New threats, attacks and vulnerabilities will eventuate – but lessons and practices used to manage risk from other technologies can be applied.

Alongside the threats and security controls suggested, the author would like to share the following recommendations from his experience in designing, reviewing and discussing web services:

### RECOMMENDATION ONE: Cover the Security basics before you start.

Alongside understanding the threats to Web Services, these steps will help ensure appropriate security controls are selected to manage risks.

Determine business impact. Ask the right people in your business what the impact is to the bottom line of a Potential Security Event targeting Confidentiality, Integrity or Availability of Web Services to the business, its staff, and its customers. Understand what is important to the business.

Determine the impact of an outage. Availability of services is always important to your organization and the impact changes over time – remember to ask questions like "What would be the impact of an outage after 1 hr? 1 day? 2 days? A week? Etc."

Classify your information. Make sure you classify the information you are transmitting or providing access to via web services. This should be based on the potential impact of the information being inappropriately disclosed or leaked. If you don't have information classification scheme adopt one. *Make it simple though. Having more than 4 classifications of information usually leads to confusion.*

**RECOMMENDATION TWO: Avoid implementing security controls in the Routing Engine/Service bus (other than basic rate limiting**.)

If you don't you run the risk of rules becoming messy and very difficult to audit and maintain. If you are able to – implement as many controls on the Connector and have a rule "Before connecting to the Routing Engine/Service Bus these security controls must be in place."

**RECOMMENDATION THREE: Use combinations the following Security Controls as part of your means to mitigate risk to web services:**

INPUT VALIDATION, Authentication, Access Controls, Auditing/Logging, Non-repudiation controls, Rate Limiting, Authorization processes, Domain Isolation/Segregation, Security Monitoring.

**RECOMMENDATION FOUR: Ensure you Validate Web Service input!**

**RECOMMENDATION FIVE: If this web service is sensitive and/or of high value to the business engage perform a penetration test on your web service environment.**

Ensure that your testing provides assurance that your security controls are adequate and that defense in depth is followed.



Michael Taylor, warriormonkster@gmail.com

# 6. References

Kwabi, C (2003). "XML Services Security and Web based Application Security." Retrieved 11th September 2008 from http://www.sans.org/reading_room/whitepapers/securecode/xml_web_services_security_and_web_based_application_security_1201

Baresi L. and Di Nitto E.(2007). "Test and Analysis of Web Services." READING: Springer, Nov 2007.

Ceramin E (2002). "Web Services Essentials." READING: O'Reilly Publications, 2002.

Last Updated: October 24th, 2017

# Upcoming Training


| | | | |
|---|---|---|---|
| San Diego Fall 2017 - SEC401: Security Essentials Bootcamp Style | San Diego, CA | Oct 30, 2017 - Nov 04, 2017 | vLive |
| SANS Seattle 2017 | Seattle, WA | Oct 30, 2017 - Nov 04, 2017 | Live Event |
| SANS San Diego 2017 | San Diego, CA | Oct 30, 2017 - Nov 04, 2017 | Live Event |
| SANS Gulf Region 2017 | Dubai, United Arab Emirates | Nov 04, 2017 - Nov 16, 2017 | Live Event |
| Community SANS Colorado Springs SEC401~ | Colorado Springs, CO | Nov 06, 2017 - Nov 11, 2017 | Community SANS |
| SANS Miami 2017 | Miami, FL | Nov 06, 2017 - Nov 11, 2017 | Live Event |
| Community SANS Vancouver SEC401^ | Vancouver, BC | Nov 06, 2017 - Nov 11, 2017 | Community SANS |
| SANS Sydney 2017 | Sydney, Australia | Nov 13, 2017 - Nov 25, 2017 | Live Event |
| SANS Paris November 2017 | Paris, France | Nov 13, 2017 - Nov 18, 2017 | Live Event |
| Community SANS St. Louis SEC401 | St Louis, MO | Nov 27, 2017 - Dec 02, 2017 | Community SANS |
| Community SANS Portland SEC401 | Portland, OR | Nov 27, 2017 - Dec 02, 2017 | Community SANS |
| SANS San Francisco Winter 2017 | San Francisco, CA | Nov 27, 2017 - Dec 02, 2017 | Live Event |
| SANS London November 2017 | London, United Kingdom | Nov 27, 2017 - Dec 02, 2017 | Live Event |
| SANS Khobar 2017 | Khobar, Saudi Arabia | Dec 02, 2017 - Dec 07, 2017 | Live Event |
| Community SANS Ottawa SEC401 | Ottawa, ON | Dec 04, 2017 - Dec 09, 2017 | Community SANS |
| SANS Austin Winter 2017 | Austin, TX | Dec 04, 2017 - Dec 09, 2017 | Live Event |
| SANS Munich December 2017 | Munich, Germany | Dec 04, 2017 - Dec 09, 2017 | Live Event |
| SANS vLive - SEC401: Security Essentials Bootcamp Style | SEC401 - 201712, | Dec 11, 2017 - Jan 24, 2018 | vLive |
| SANS Bangalore 2017 | Bangalore, India | Dec 11, 2017 - Dec 16, 2017 | Live Event |
| SANS Cyber Defense Initiative 2017 | Washington, DC | Dec 12, 2017 - Dec 19, 2017 | Live Event |
| SANS Cyber Defense Initiative 2017 - SEC401: Security Essentials Bootcamp Style | Washington, DC | Dec 14, 2017 - Dec 19, 2017 | vLive |
| Community SANS Nashville SEC401^ | Nashville, TN | Jan 08, 2018 - Jan 13, 2018 | Community SANS |
| SANS Security East 2018 | New Orleans, LA | Jan 08, 2018 - Jan 13, 2018 | Live Event |
| Community SANS Hawaii SEC401 | Honolulu, HI | Jan 08, 2018 - Jan 13, 2018 | Community SANS |
| Mentor Session - SEC401 | Memphis, TN | Jan 09, 2018 - Mar 13, 2018 | Mentor |
| Northern VA Winter - Reston 2018 | Reston, VA | Jan 15, 2018 - Jan 20, 2018 | Live Event |
| SANS Amsterdam January 2018 | Amsterdam, Netherlands | Jan 15, 2018 - Jan 20, 2018 | Live Event |
| Mentor Session - SEC401 | Minneapolis, MN | Jan 16, 2018 - Feb 27, 2018 | Mentor |
| Las Vegas 2018 - SEC401: Security Essentials Bootcamp Style | Las Vegas, NV | Jan 28, 2018 - Feb 02, 2018 | vLive |
| SANS Las Vegas 2018 | Las Vegas, NV | Jan 28, 2018 - Feb 02, 2018 | Live Event |
| SANS Miami 2018 | Miami, FL | Jan 29, 2018 - Feb 03, 2018 | Live Event |