Global Information Assurance Certification Paper

How secure are your Web server applications?

By

Dennis Carter

GSAE Practical Assignment 1 Option 1

Version 1.1

Table Of Contents

Introduction

This paper covers Web server application security and the need to build security into our web applications. I am sure that most if not every security professional has heard of the "Defense In Depth" principle, which mainly relates to protecting the network and network devices as well as the operating systems. Companies' implement firewalls, intrusion detection systems (network and host), patch management for operating systems and applying best practices. Some companies may be implementing intrusion prevention systems (IPS) as well.

I have not heard much about having programmers or developers writing secure code when creating web applications. During my SANS class the issue was brought up how vulnerable web applications have become. This stayed in my mind the rest of the week. I was thinking to myself how vulnerable were the web applications running at our company.

When I returned from the SANS conference I talked to one of our senior programmer's in the web application development area and asked him if security was being built into the web applications being developed for our company. The programmer looked at me with a blank look. I knew right then he had no clue what I was talking about. Later in this paper I will tell how vulnerable our web applications were.

The problem with Web Server Applications

What makes web server applications vulnerable is not the application itself but the logic used in the application. Companies all over the world sell and provide services to their customers and suppliers by way of the Internet through web server applications. Time is money and when applications are needed to be developed to provide for managing customer services, deploy new products and services or enhance their supply chain operations, writing the application with security built into the logic code is not taken into account. How fast the application can be developed and put into service is the norm.

The web service protocols HTTP port 80 for general web traffic and SSL port 443 for encrypted traffic provides the gateways for hackers to access secure files such as customers private data and proprietary corporate information. Some of the vulnerabilities associated with web server applications include Cross Site-Scripting, Sql Injection, Buffer Overflows and Session Hijacking. These are just some of the vulnerabilities associated with web server portals. This does not take into account the known

vulnerabilities associated with the different web applications being used in development like Windows IIS, Apache, Tomcat/Jakarta, Java and PHP. No company or government agency is immune from hacker attacks on the web application level. We heard of such attacks against Yahoo, NASA and even the CIA.
Recently there have been a large number of patches released from the various web server application vendors to patch their software against the vulnerabilities mentioned above. Another scenario that will have to be looked at more closely is the development of Simple Object Access Protocol (SOAP) that is being developed jointly by Microsoft and DevelopMentor, which has a World Wide Web Consortium called (W3C). The development of (SOAP) is to provide application-to-application information exchange using web services over HTTP.

Open Web Application Security Project

The Open Web Application Security Project (OWASP) is an open source reference point for system architects, developers, vendors, consumers and security professionals involved in designing, developing, deploying and testing of web applications and services[1].
The OWASP is a good source with a wealth of information on how to write Web applications with security in mind. OWSAP also provides a list of the top 10 web application vulnerabilities and a document called “a guide to building secure web applications”. The OWASP is an open source document. If you are in the field of web application development and want to contribute you can send an email to owasp@owsap.org.
You don’t have to be technically savvy to understand the information in this guide. Security auditors can use this guide to create a check list for vulnerabilities that they want to test for in their own company web applications. The guide will take you through how web browsers interact with web applications that retrieve information from different information resources. You will also learn how different attacks are formed and what you can do to prevent such attacks.

Network Security Tools Are Not Enough

The network security tools that are used by most security professionals for protecting the enterprise network are not going to be able to stop web application attacks.

[1] A guide to building secure web applications.

Let take a look at firewalls. There are two types of firewalls, proxy application Firewalls and deep packet inspection. Both types of firewall will use a rule base to either “Allow” or “Deny” traffic through your network. Proxy application firewalls check the packet all the way to the application layer. So why isn’t the proxy application firewall not stopping web application attacks?
The proxy application firewall only validates the header information in the packet not the input or response for example an HTML application.
Deep packet inspection firewalls check the packet base on whether the packet is RFC complaint and against pre-programmed patterns (i.e., attack signatures).
Network Intrusion detection system (NIDS), Host intrusion detection system (HIDS) and even Intrusion prevention system (IPS) are not going to be enough.
Network Intrusion Detection Systems look at network traffic based on policies and attack signatures applied to the network sensor. The sensor reports to a central console that displays information of a possible vulnerability. If the network traffic matches an attack signature, the NIDS sensor will block the attack.
Host intrusion detection system work the same way except you can customize the policy to tell you who logs in as root or administrator, whose logon attempt failed, which files were changed and by whom. If registry changes were made and by whom and if you have to you can block all incoming and out going traffic to that particular host.
Intrusion prevention systems work the same way as NIDS and HIDS with the addition of being anomaly based. IPSes learn what is considered normal network traffic and what is normal server activity. IPSes will detect and block attacks when they are present on your network. Through policies you can allow or disallow who can update system files.
The reason why these security tools mentioned above will not protect your network from web application attacks is that they lack the built in intelligence required to understand application level conversations. The use of SSL encryption being used over networks causes another problem. None of the above devices can decrypt and inspect application traffic, allowing encrypted exploits and attacks right through your network security.

Cross-Site Scripting

Cross Site Scripting has been defined as an attack aimed at pushing a script tag into a server that would be sent from the server to an innocent user browsing the web server thus causing the script to be activated in the innocent user's browser.[2] Most of the Cross Site Scripting occurs when a user receives e-mail with a URL that the user can click on. Once the user clicks on the site, the script embedded in the URL header is executed. What if the site the user goes to is one that the user authenticates to. The following example can show you what can happen. Suppose we go to a site that allows legitimate input to the application via a form. The HTTP request would look like this.

http://www.somewhere.com/test.cgi?userid=dennis

The simple attack URL may look something like this.

http://www.somewhere.com/test.cgi?userid=dennis<script>alert(hello there)<script>

If this were a legitimate site the user would receive a browser pop-up message "hello there". A more malicious script could have been placed into the link that could cause real damage or even steal information from an innocent user.

Defense-In-Depth-Extended

The "Defense In Depth" principle as I stated earlier in this paper mainly pertains to securing the enterprise network. I believe the "Defense In Depth" needs to include web application security at the application layer, so I call this "Defense In Dept Extended". The web applications developed today need to be built with security in mind. During my research of this paper I found articles going back as far as 4 years ago saying how vulnerable web applications are. Developing security into web applications is not easy. It requires tools that developers and programmers can use to check their application code for vulnerabilities while the application is in the development stage. Tools are needed by security auditors to check existing web applications for vulnerabilities and provide a solution to correcting the problem.

[2] www.imperva.com - How safe is it out there?

Security tools

There are many companies out there that have tools that can help Web server developers build security into their Web server applications and help security auditors do vulnerability and penetration testing. You have to decide whether your company should use a software solution or hardware solution (Application Security Gateway).
Sanctum and SPI Dynamics provide software tools for developers and security auditors. Both companies claim their software applications support the life cycle of the Web application from development to production. Sanctum AppScan and SPI Dynamics WebInspect can be used for vulnerability testing in different Web platforms such as IIS, Apache and WebLogic, as well as individual Web applications. You can find more information on their set of tools at the following websites.
Sanctum – www.sanctum.com and SPI Dynamics at www.spidynamics.com.
Aspect security provides the same type of tools as Sanctum and SPI Dynamics in addition to providing a consulting service and Web application development training for your development team.
There are appliances that will intercept input traffic from and un-trusted site and examine the packet for invalid requests as well as application layer attacks. Some are policy based with anomaly detection and some actually captures the packet and write a checksum hash to the response then compares the hash for the request. If the request has been modified the request is blocked.
Some companies that have Secure Gateway Appliances are as followed.
Imperva's SecureSphere - www.imperva.com.
Sentryware's Sentryware Hive – www.sentryware.com
Teros Fireline – www.teros.com
Vormetric – Core Guard – www.vormetric.com.

Summary

There's know doubt that trying to secure private information from hackers is a daunting task. Most Web server applications and databases have been in service for years. If companies don't protect the sensitive information they hold in their databases about their customers, the results could lead to civil law suits and penalties against the company. Some small businesses that depend on the Internet for their survival in the market place would go out of business.

Identity theft is the fastest growing crime in America today. Security auditors have their work cut out for then in trying to minimize risk in the corporate enterprise. Vulnerabilities in Web server applications are getting a lot of press lately and every day I receive information on patches for PHP, Apache, Java and XML. It appears that there is not a Web server application being used today that is not vulnerable to hacker attacks.
I previously I stated that I was able to have a company come in and do vulnerability testing on one of our Web servers. The testing was done on our Web server development system. The results were as followed. There were 10245 potential vulnerabilities found. 43 vulnerabilities were confirmed. 5 of the confirmed vulnerabilities were admin issues and 38 coding issues. Thanks to the vulnerability testing these issues are being taken care of. The web server development manager was at the post meeting with the company doing the vulnerability testing. The manager was surprised to see and how easy our web applications were exploited.

Internet References:

http://www.spidynamics.com/support/whitepapers/Webapp_Dev_Process.pdf

http://www.imperva.com/application_defense_center/papers/how_safe_is_it.html
Written by Morgan Surf.

http://www.spidynamics.com/support/whitepapers/webappwhitepaper.pdf
Written by Caleb Sima.

http://www.owasp.org/index.jsp

http://www.didata.com/documents/Application_Networks.WP.pdf
Author: Ettienne Reinecke.

http://www.spidynamics.com/whitepapers/SPIcross-sitescripting.pdf
Written by Kevin Spett

http://www.cert.org/advisories/CA-2000-02.html

Magazine Articles:

Information Security – "Wide open on port 80".
By Kelly White and Yong-Gon Chon.

Web server Audit

By

Dennis Carter

GSAE Practical Assignment 2 Version 1.1

Table of Content

Background

The Dookie Dook corporation provide services to universities and veterinarians around the world on the diseases and anatomy of the exotic animal known as the Ferret. The corporation also provides information to the general public as well as Ferret supplies such as ferret food,vitamins,cages and accessories. The company has 20 web servers running on windows 2000 platform.

A large portion of the company revenue is derived from the services they provide and the selling of goods over the Internet.

The Dookie Dook corporation web servers are exposed to the threats and vulnerabilities associated with doing business over the internet. The risks are and not limited to viruses, worms, Trojan horses, denial of service attack, unauthorized access and arbitrary code execution as well as insider threat of corporate information.

Scope:

The audit scope will be primarily on the configuration and access to the web servers as they relate to best practices and the corporation's security policy.

Objective:

The Dookie Dook corporation is seeking a comprehensive audit of there web servers to identify and obtain effective procedures to mitigate the threats to and vulnerabilities of critical assets in the corporation.

Audit process:

All items in the audit check list will be addressed in the audit results phase. The audit will consist of a physical examination and questions to the system administrators related to the web servers configuration, based on the audit checklist. The auditor will need the assistance of a system administrator. The following components will be audited on each web server.

- Access Control and Authorization
- Files and Directories
- Auditing and Logging
- Latest patches and Updates

- Registry
- IIS Metabase

The time frame to complete the audit will be based on the time available for appropriate staff to assist the auditor. Additional recommendations will be listed under post analysis.

Audit checklist:

Access Control and Authorization:
Accounts on a web server grant authenticated access to the computer system and web applications. Access accounts should be set with the least privileges. A common vulnerability is elevation of privileges caused by either over-privileged process accounts or over-privileged service accounts.

What is the purpose of the user account?
How much access does the user account have?
Who has access to the administrator account?
Is strong password policy enforced?
Is the logon session terminated after a number of failed login attempts?
Is the guest account disabled?
Is remote access to the web severs encrypted?

Files and Directories:
The volumes should be NTFS and not FAT. NTFS allows the system administrator to create permissions to critical files and directories. Web site content should be located on a non-system NTFS volume. The anonymous Internet account should not be allowed to write to web site root directory or to any of the content directories.

Are the files and directories contained on a NTFS volume?
Is web site content on a non-system NTFS volume?
Are sample applications that were installed when installing web server applications removed?
Does the anonymous Internet account have a deny write access to the web site root directory and content directories?
Is there appropriate restrictions placed on the Everyone group?

Auditing and logging:
Event and system logs should be turned on to help trouble

shoot any security problems within the operating system. Additional IIS auditing should be configured on the web services. Audit logs help in identifying intruders, attacks in progress, and evidence of attacks that have occurred. These logs should be archived and reviewed frequently.

Is system, application and security logging turned on?
Is IIS audit logging turned on?
Is IIS configured for W3C extended log file format?
How often are the logs reviewed?
Are the Logs archived for later review?
How often are the log files backed up?
Who review's the logs?

Latest patches and updates:
The latest system patches and hotfixes should be applied to the operating system. Latest updates should be applied to the web server applications as these patches and updates address current vulnerabilities that may be exploited.

What is the patch release level of the operating system?
Do the web server applications have the latest updates installed?
How are patches and updates administered?

Registry:
Authorized administrators should be the only one's to have access to the registry. The registry proves a repository for many of the vital server configuration settings. An attacker could cause damage to the system rendering it non-functional.

Is remote access to the registry permitted?
Who has remote access to the registry?
Is the remote connection encrypted?

IIS Metabase:
The IIS configuration files are maintained in the IIS Metabase. Access to the IIS Metabase should be restricted with harden NTFS permissions. You should not install any services that are not needed.

Is the IIS Metabase restricted by NTFS permissions?
Is banner information restricted?

Audit Results:

The following is the results of the audit for each category

in the audit check list. The results are based on interviews with the system administrators and physical examination of each web server in the dookie dook company. The results to the audit are compared to “best practices for Windows 2000 web server” and the company security policy. If there are any recommendations to be made they will be presented at the end of the results for each category.

Access control and authorization:
The user account is used for non-administrative staff that need to have access to web server applications. The user account is a read only account with no write or executable permissions.
Strong password length and complexity are enforced. The password length has to be eight or more characters and must include both alphabetical and numeric characters.
Logon attempts that fail 3 times are locked out of the system. The locked out person has to contact the system administrator to have password reset.
The guest account was disabled on all web servers.
Remote access is allowed and done through a VPN account using IPSEC with two-factor authentication.

The access control and authorization comply with the company security policy and best practices.

Files and Directories:
All files and directories are created on a NTFS volume.
All web servers except for two had their web applications on a separate non-system volume.
All sample applications were removed after the installation of web server applications. However I noticed that some system, utilities and debugging tools were still available on the system. The system and Utilities tools showed to have restrictions applied.
The anonymous Internet account is disabled.
The appropriate restrictions are placed on the Everyone group.

Recommendation:
The two web servers that have web applications on the same volume as the operating should be on a separate volume. If vulnerabilities exist in either the system or web server applications an attacker could exploit both the system and web server applications.
Debugging tools should not reside on the operating system

instead the debug tools should be on a CD and only be used by system administrators.

Auditing and logging:
All web servers have the appropriate logging turned on.
IIS logging is turned on but without IIS W3C extended log file format auditing.
Log files are normally reviewed whenever a problem arises within the system or web server applications.
Log files are backed up whenever full system backups are scheduled which is nightly.
Full volume backups are rotated on a 30-day rotation.
Only system administrators have access to the log files.

Recommendation:
IIS W3C extended log file format should be used as W3C allows the system administrator to turn on more in depth logging versus the IIS default logging.
Log files should be scanned daily looking for any potential problems.
Log files should be archived for at least 90 days instead of the 30-day rotation. This gives the system administrators enough logs if they have to go back to determine when a problem started.
Since the company has a security section it would be beneficial for a security analyst to review the logs.

Latest Patches and Updates:
All web server operating systems are running windows 2000 service pack 3. Some system hotfixes have not been applied.
All web server applications have the latest updates applied.
Patches and updates are applied manually to all systems.

Recommendation:
All web server systems should be on windows 2000 service pack 4. It is advisable to have a test system with the web server configuration available to test any service pack or hotfixes be applying to a production system.
A patch management tool should be purchased so that patches and updates can be scheduled and pushed to each web server instead of having the system administrators go to each system.

Registry:
Remote access to the registry is allowed only if a problem

develops after hours. Only the system administrators group has after hour's access. After hours access is only permitted through a VPN connection using two-factor authentication.

IIS Metabase:
Proper restrictions placed on the IIS Metabase by NTFS permissions.
The auditor went to one of the company websites and examined the banner information from the page. Here is the result of the banner page.

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Content-Location: http://10.10.100.1/Default.htm
Date: Thu, 8 Jul 2004 14:03:52 GMT
Content-Type: text/html
Accept-Ranges: bytes
Last-Modified: Wed, 16 Jun 2004 18:56:06 GMT
ETag: "067d136a639be1:15b6"
Content-Length: 4325
```

As you can see the content-location information is being passed in the html page, which gives the receiver the IP address and not the fully qualified domain name (FQDN).

Recommendation:
Set the w3svc/UseHostName to True. By default this is set to False.
Please read the following Microsoft knowledge base for explanation of modifying the above value. The website is

http://support.microsoft.com/default.aspx?scid=kb;en-us;218180&sd=tech

Post Analysis:

Over all the audit shows that the system administrators are adhering to the company security policy and best practices for the windows 2000 server. The auditor also noted that the company has an IT security department. It would be beneficial for the company to have the IT security personnel to work with the system administrators and do further analysis on the company web servers. Ex: scanning

for unnecessary ports or services that could lead to an outside attack.
The IT security personnel should also have access to all log files for further analysis.
Web Severs that post and retrieve sensitive information for goods and services purchased over the Internet should have a Host Intrusion detection software or appliance to monitor and protect against attack signatures or unscrupulous activity. A Host Intrusion detection software or appliance can alert security personnel and system administrators to an attempt or unauthorized access to the web servers.
The IT security department should work with the Web applications section to schedule a vulnerability or penetration testing on their web applications.