Global Information Assurance Certification Paper

Application Firewalls: Don't Forget About Layer 7

GIAC Security Essentials
Certification (GSEC)
Practical Assignment
Version 1.4c

Option 1 - Research on Topics
In Information Security

Russell Eubanks
Las Vegas

Table of Contents

Figures

Abstract

Web and database communication have become the prevalent communication now integrated into nearly every production system in the corporate infrastructure. Most business processes rely heavily on the confidentiality, integrity and availability of these systems. Securing web-based communication is and will remain vital to existing business sustainability and future growth.

The enterprise web application environment is a rapidly evolving, mission-critical, domain consisting of web, application, and database servers. Backend data stores house information that must be protected from unauthorized access from both internal and external sources. Measures must be implemented that monitor web and database traffic on previously approved transmission ports and protocols.

Summary

So you have a website. Without a doubt you already follow security best practices by using various control measures such as firewalls, border routers, antivirus scans, a robust patching strategy and more. But what about traffic on previously approved routes? Just because you must permit web access does not mean that you have to allow malicious usage on your website. Never forget - you can not deny what you must permit. How do you resolve the conflict of providing access without making yourself susceptible to attack?

I suggest that a web application firewall should exist in your information security toolkit to provide yet another layer of defense. A traditional firewall can be defined as “a means to control what is allowed across some point in a network as a mechanism to enforce policy”. (SANS) What exactly is a web application firewall? This innovative technology is much more than a router with rules. It serves as a means to protect the application and its backend data store from malicious attack and inappropriate usage. While it is appropriate to allow a user to use your site, a user should not be allowed to abuse his or her privileges.

An application firewall is designed to permit only acceptable application traffic. This technology can be deployed inline or in passive mode. Each deployment type has definite advantages and disadvantages.

The Problem

Enterprise firewalls are pervasive in modern day network architectures and are truly considered a requirement. While they can be used to limit all but appropriate traffic on previously approved routes, nothing prevents these very same paths from being exploited. These paths must exist to promote and facilitate the usage of your web site and applications, yet most organizations currently have no means to stop, monitor and alert on malicious traffic utilizing these established routes.

More specifically, to enable your business, you allow traffic on web ports 80 and 443, as well as application specific ports. By allowing all traffic the opportunity to flow over these ports, users with malicious intent have the ability to take advantage of vulnerabilities at the application level. (Kennedy)

Current Means to Protect the Applications

What is in place to stop application exploit? How do you know if your applications and data stores are being used as they were intended if your toolkit stops short of the application layer? As a Defense in Depth approach, traditional firewalls should not be the only protective measure in place to defend your websites and their backend data stores.

Other measures often employed are Network Intrusion Detection and Prevention Systems (NIDS/NIPS). These solutions actively monitor traffic on the network for malicious activity. NIDS solutions are often set in passive or SPAN port mode. This means that NIDS can only send TCP resets to stop some of the bad TCP packets. A shortfall of a NIDS solution is that they can not actively block any UDP traffic.

NIPS perform the same functionality as a NIDS, except that it sits actively inline with the data flow it is monitoring. This option is able to actively block any packet deemed inappropriate for that network segment.

Host-Based Intrusion Detection Systems (HIDS) and Host-Based Intrusion Prevention Systems (HIPS) can also be used to protect servers. HIDS and HIPS are parasitic software that monitors respective hosts for anomalous behavior. This software can look for specific attacks directed at the server, whereas the network solutions monitor only the network traffic between them. (SANS)

Banners and warnings can be used to list Acceptable Use Policies to clearly

define the expectations and consequences of being a user of an application,

Authentication, Identification and Authorization - Forcing the usage of userids and password has initial advantages, but introduces sometimes a false sense of security. With this control, you must realize if you allow users to self-register, you may never really know your users are who they say they are. For the most part, you must be willing to accept the risk of not verifying the identity of some (if not all) of your web users.

Access Control Lists should already be in place to allow only least privilege access to your site.

The Solution

Even if you have a robust web security program, it is imperative that you do not give up too much information during the course of normal web usage. While it is a good practice to notify a user of an error that they may be able to correct, often too much information is made available. Examples of this common oversight include providing operating system version, application level and extended web server information. While not always detrimental, this meta data could be used by an attacker to focus their resources against your infrastructure. When practical and if the costs are not prohibitive, I suggest you force your attacker to execute a diligent discovery process.

Don't Stop Short

The seven layers of the Open System Interconnection (OSI) reference model are: Application, Presentation, Session, Transport, Network, Data Link and Physical. They exist as abstract levels to help describe how network traffic flows from one computer to another. (Cisco)

Application
Presentation
Session
Transport
Network
Data Link
Physical

(OSI Model)

Typically, network monitoring occurs below the Application layer. The introduction of application firewalls compliments the existing suite of network monitoring tools to help monitor and defend the OSI model in the corporate network.

What Are You Trying To Prevent?

Many websites use a portal as a means to define the user to a particular role. Legitimate website usage is preferred and many times assumed. Without diligent security foresight, many unintended consequences of hosting a website will likely become a reality. Some of these include:

Information leakage – What if the website user is able to trick the application into giving more information than you intended?

Application exploits – What if your website can be used against you?

Escalation of privileges – What if a regular user is able to trick the application logic into believing that the user is an administrative user?

Total system compromise – What if there is no separation of administrative accounts?

Legal issues – Regulatory compliance may very well impose penalties for failure to secure your infrastructure appropriately.

Lawsuits – Your organization could find itself liable should information be disclosed about a constituent without their explicit permission.

Employment - What if a system compromise causes a loss of your job?

Unfavorable media attention – In keeping up with the latest *publicized* data compromises, sometimes, no news really is good news.

Unwanted IP traffic and attention – Your IP range once compromised could end up circulating around as yet another example of “low hanging fruits”.

Company closure – Certainly worst case, but a company could cease to exist if the system compromise is deemed serious enough.

Perhaps if you execute your security program better than others, lesser skilled attackers would stay away from your digital doormat.

How Did I Get in This Mess?

There are generally two ways to acquire a web application; those that are purchased and those that are developed internally. Both have their advantages and disadvantages.

Purchased code has the advantage of being readily available. A third party that regularly performs these tasks often produces it. Unfortunately, when security vulnerabilities are found with purchased code, there is little your company can do to correct the problems, short of awaiting a future software release or patch. When you are in this dilemma, the decision must be made to either disable the application or accept the risks associated with its exploitation. In today's world of regulatory compliance, failure to comply is not an option.

In-house developed code, while also attractive, does not automatically lend itself to secure coding practices. Sure, the developers recently attended a secure coding workshop, webinar or conference, but the truth is that many Information System developers remain focused on production deadlines. While secure coding is important, it is not generally the focus of their efforts. More often than not, meeting production deadlines is the measure of a successful programming career.

Another pitfall of in-house developed code is the time commitment needed to produce the software product. Even though your programming staff has the skills needed to produce the application, often aggressive project scheduling will not allow for this method of software development.

Application, Protect Thyself

If you are not actively monitoring the application, how will you know it is being exploited? I suggest that unless there are verbose application logs (where available and where turned on) that are reviewed daily, you would likely never know of application exploit. Even if you check the application logs, how can you determine the subtle difference between application usage and application exploit?

Even if you are reviewing you logs, still more can and should be done. You are an experienced Information Security Analyst. Because you have attended the Security Essentials Track at a recent SANS conference you know the principles of Defense in Depth. You have installed the best Network and Host based Intrusion Detection Systems available. You also have a prudent and dynamic patching schedule. You regularly run Vulnerability Assessment scanners and try

diligently to harden your servers. You have network segregation and flow the principle of least privilege.

I suggest that another Defense in Depth concept is looking at the application level for web and database traffic. If you are not actively monitoring for application attacks, how will you know what is occurring there? How will you know if your own application is being used against your organization?

The application, while often defenseless, now has a robust means of protection - The Application Firewall. The application firewall can protect your organization against: cross site scripting, SQL Injection and discount cookie poisoning. Many solutions can also decrypt SSL sessions that have been the Achilles heel of traditional network intrusion devices. No longer can SSL application traffic pass by without first being evaluated for security risks.

Regulatory Compliance (or Tell Me Why I Have to Do This)

There are several regulatory compliance initiatives that call for an application firewall. Some do so explicitly, others infer to this solution. These regulations are Health Insurance Portability and Accountability Act of 1996 (HIPAA), Centers for Medicare & Medicaid Services (CMS) Acceptable Risk Safeguards (ARS), VISA Cardholder Information Security Program (CISP) and California Information Practice Act or Senate Bill (SB 1386).

Healthcare organizations have to comply with the HIPAA Security Rule by 4/21/2005. HIPAA regulation 164.312(a)(1) Access Controls states “Implement technical policies and procedures for electronic information systems that maintain electronic protected health information to allow access only to those persons or software programs that have been granted access rights as specified in Sec. 164.308(a)(4)”. The intent of the HIPAA security rule is to provide appropriate handling guidelines for protected health information (PHI). Failure to comply with these regulations is costly – up to $250,000 and 10 years imprisonment if the intent was to do harm is proven. (Federal Register)

To meet the regulatory requirements of HIPAA and CMS Medicare contractors must abide by the CMS ARS guidelines to become both HIPAA Security and Medicare compliant. These obligations will be satisfied by a number of hardware, software and business redesign implementations.

This HIPAA Security regulation might seem to imply latitude and reasonableness. Nowhere in this regulation does it state that an application firewall should be implemented. If required or inclined to have a comprehensive HIPAA checklist, one alternative is to map the HIPAA Security regulations to the

CMS ARS Standards. Standard 6.1, Firewall Hardware and Software, clearly states “Utilize stateful inspection/application firewall hardware and software”. (CMS ARS)

Additional regulations to support the business case for Application Firewalls include SB 1386. This bill, effective July 1, 2003 requires an entity that conducts business in California to disclose to its participants any breach of the security of that data. This applies to each resident of the state of California “whose unencrypted personal information was, or is reasonably believed to have been, acquired by an unauthorized person.” (SB 1386)

Personal information, as defined in this legislation includes: first and last name in combination with any one or more of the following data elements, when either the name or the data elements are not encrypted:

- Social security number
- Driver's license number
- Account number
- Credit or debit card number

Should an unauthorized data breach occur, all participants in that data store must be notified. User notification consists of either a written, electronic or substitute notice. The substitute notice is a provision for instances where the cost of written and electronic notice would exceed $250,000 or the number persons involved exceed 500,000. This provisional exception must include an e-mail notice, conspicuous posting of the notice on the agency's Web site and notification to major statewide media.

SB 1386 does just that. The law, which went into effect in July of this year, requires companies that own or have access to personal information of California residents to notify them if their data has (or may have) been accessed illegally.

Civil actions may be entered into by those whose data have been compromised. Fines that can be assessed due to breaches come in the form of an uncapped civil suit. (Lourie)

Visa USA has instituted the CISP program. Mandated since June 2001, the programs intent is to protect Visa cardholder data—wherever it resides—ensuring that members, merchants, and service providers maintain the highest information security standard. Some suggest that this program is an attempt for VISA to share the responsibility for VISA credit card fraud and abuse due to improper storage, transmission and processing of their products. (VISA 1)

Depending of the number of annual VISA transactions per year, merchants are

subject to a host of assessments and audits. These include quarterly vulnerability scans, annual questionnaires and onsite audits.

The Payment Card Industry Self-Assessment Questionnaire twelve requirements, each containing several questions used to gauge the effectiveness of the VISA merchants' security program. From the questionnaire two items are remedied by an Application Firewall: Item 6.6 states "when authenticating over the Internet, is the application designed to prevent malicious users from trying to determine existing user accounts?" Item 6.8 states "Are controls implemented on the server side to prevent SQL injection and other bypassing of client side-input controls?" (VISA 2)

Procedures also exist should a VISA merchant's site and backend data store become compromised. Substantial penalties are enforced if a compromise occurs to a non-VISA CISP merchant. The fines are up to $500,000 per incident. (VISA 3)

Protect Me From

In Beyond Fear, Bruce Schneier asserted, "Security is about preventing adverse consequences from the intentional and unwarranted actions of others".

Attack types that must be stopped include, but are by no means limited to, SQL Injection, cross site scripting and discount cookie poisoning. It is important to ensure these threats are understood.

SQL Injection "refers to the technique of inserting SQL meta-characters and commands into Web-based input fields in order to manipulate the execution of the back-end SQL queries." (Mookhey) Perhaps the best way to defend from SQL Injection and buffer overflows is from secure coding practices. (Gannon)

A definitive way to know a site is vulnerable to SQL Injection is when a database error is presented in the browser. Many of these errors contain the phrases "SQL Server, ODBC and Syntax"; however it is not limited to these. (SPI Dynamics)

The next application exploit to be discussed is Cross Site Scripting (CSS). Cross Site Scripting "attacks work by embedding script tags in URLs and enticing unsuspecting users to click on them, ensuring that the malicious JavaScript gets executed on the victim's machine. These attacks leverage the trust between the user and the server and the fact that there is no input/output validation on the server to reject JavaScript characters." (Mookhey)

Cross Site Scripting is often used as a means allow an attacker to have their malicious code to be executed on a legitimate website. Scripting errors can

even permit an attacker to use a counterfeit login system to gain credentials for a given account. (Hines)

Discount Cookie Poisoning is another way to use a website against its owner. With this attack, the malicious user is able to manipulate the cookie to grant them an unintended discount on a web-based purchase. This is accomplished by altering certain values in the cookie. I suggest that this attack could potentially be untraceable if the request is not very aberrant from other web sales. A strategy for this attack might be to find the best price and then better that price by ten more percent.

The risk from the above application exploits can be mitigated by an application firewall. By using this technology, your organization can focus its efforts elsewhere.

Listing and Overview of the Vendors

So you have been convinced that you need Layer 7 protection. Now what? How can you navigate the vendor offerings effectively and in a timely manner?

Below is a matrix template that will hopefully remove some of the fear, uncertainty and doubt associated with evaluating this emerging technology I have listed some of the major Web Application Firewall vendors and what I believe are key criteria respective to each of them. Data for this matrix was obtained from information available on the respective vendors' websites. Good luck!

Vendor	iMPERVA	f5	Teros	Kavado	NetContinuum
Product	SecureSphere	TrafficShield	100/200	InterDo	NC-1000
Website	www.imperva.com	www.f5.com	www.teros.com	www.Kavado.com	www.NetContinuum.com
Inline	X	X	X	X	X
Passive	X				
Web sites	X	X	X	X	X
Learning Mode	X	X	X		X
SSL Certificates	X	X	X	X	X
SSL Acceleration		X	X		
Mask Sensitive Fields		X	X	X	X
Pricing	$35,000	$35,000	$25,000	$15,000	$29,000

(Network World Fusion)

Conclusion

Application Firewalls are a relatively inexpensive means to help fortify your websites, particularly compared to the cost of an application compromise. With so many regulatory compliance initiatives to be in compliance with, there is plenty of justification for such a solution.

Inline brings with it additional risk. Because it is sitting in the traffic flow, it can drop any offensive packets. Passive mode is likely to be the initial method of deployment. This is due to the fact that it is placed in passive mode via SPAN ports that only observe traffic and send only TCP resets to attacks previously identified.

References

1 - SANS Institute. Track 1 - SANS Security Essentials. Volume 1.3. SANS Press, Sep 2004.

2 - Kennedy, Susan. "Common Web Application Vulnerabilities." COMPUTERWORLD. 2 Feb. 2005 URL: <http://www.computerworld.com/printthis/2005/0,4814,99981,00.html>

3 – Open Standards Interconnect Model - an Operational Example <http://www.inetdaemon.com/tutorials/theory/osi/operation.html>

4 – SANS Institute. Track 1 - SANS Security Essentials. Volume 1.3. SANS Press, Sep 2004.

5 – Cisco Systems Web Page. "Open System Interconnection Reference Model." URL: <http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/introint.htm>

6 – Federal Register: 20 Feb. 2003 (Volume 68, Number 34). URL: <http://www.cms.hhs.gov/hipaa/hipaa2/regulations/security/nprm/secnprm.txt>

7 – Centers for Medicare and Medicaid Services. Acceptable Risk Safeguards. Version 1.2. 2004. URL: <www.cms.hhs.gov/it/security/docs/ars.pdf>

8 – California Information Practice Act or Senate Bill (SB 1386). September 26, 2002. URL: <http://info.sen.ca.gov/pub/0102/bill/sen/sb_13511400/sb_1386_bill_20020926_chaptered.html>

9 – Lourie, Sarah. "The FAQs about SB-1386." SearchCIO.com. 12 Dec. 2003 URL: <http://searchcio.techtarget.com/originalContent/0,289142,sid19_gci941077,00.html>

10 – Cardholder Information Security Program. URL: <http://usa.visa.com/business/accepting_visa/ops_risk_management/cisp.html?it=l2|/business/accepting_visa/ops_risk_management/cisp_training_tools%2Ehtml|Cardholder%20Information%20Security%20Program>

11 – PCI Self-Assessment Questionnaire. Word Document and Adobe Acrobat formats available. URL: <http://usa.visa.com/business/accepting_visa/ops_risk_management/cisp_training_tools.html?it=l2|/business/accepting_visa/ops_risk_management/cisp%2Ehtml|Training%20and%20Tools>

12 – Securing Visa Cardholder Data. URL:
<http://usa.visa.com/business/accepting_visa/ops_risk_management/cisp.html>

13 – Schneier, Bruce. Beyond Fear Thinking Sensibly about Security in an Uncertain World. New York: Copernicus Books. 2003.

14 – Mookhey, K. “Detection of SQL Injection and Cross-site Scripting Attacks.” SecurityFocus. 17 Mar. 2004 URL:
<http://www.securityfocus.com/infocus/1768>

15 – Gannon, Darren. “Secure Your Web Application through Security Testing.” 13 Jul. 2004. URL:
<http://www.giac.org/practical/GSEC/Darren_Gannon_GSEC.pdf>

16 – “SQL Injection - Are Your Web Applications Vulnerable?” SPI Dynamics, Inc. 2002. URL:
<http://www.spidynamics.com/papers/SQLInjectionWhitePaper.pdf>

17 – Hines, Matt. "A phishing wolf in sheep's clothing." News.com. 14 Mar. 2005 URL: <http://news.com.com/A+phishing+wolf+in+sheeps+clothing/2100-7349_3-5616419.html?tag=nefd.top>

18 – “Web application firewall buyer's guide.” Network World Fusion. URL:
<http://www.nwfusion.com/bg/2004/appsecurity/index.jsp>