eCommerce and Defense in Depth

Clayton Dillard

eCommerce and Defense in Depth

Clayton T. Dillard
October 24, 2001
Assignment Version 1.2e

Executive Summary

There has never been a time when so many businesses have offered their products over the Internet as now. No matter what your company is selling or who your customers are one truth remains concerning eCommerce - Security is critical. Every day hundreds of online commerce sites are broken into. You may not read about it in the paper or see it on the evening news but it happens. Some of these attacks and subsequent breaches go unnoticed even by those who are charged with the duty of maintaining some reasonable level of security at those organizations. Have you ever thought that the reason some of the attacks on eCommerce sites aren't reported is that the method of attack was executed in such a way as not to cause any visible damage? The hackers that deface web pages probably do so to make a statement, but what about your competitor who is trying to beat you to market with the latest product offering? To the person or group that is in the business of stealing secrets and corporate data, stealth is the only successful way.

Scope

This document gives an overview of some common methods that can be employed to build defense-in-depth into your eCommerce solution. Defense-in-depth can be defined as implementing multiple layers of defense in order to mitigate the risks associated with attacks.

Focus Topics

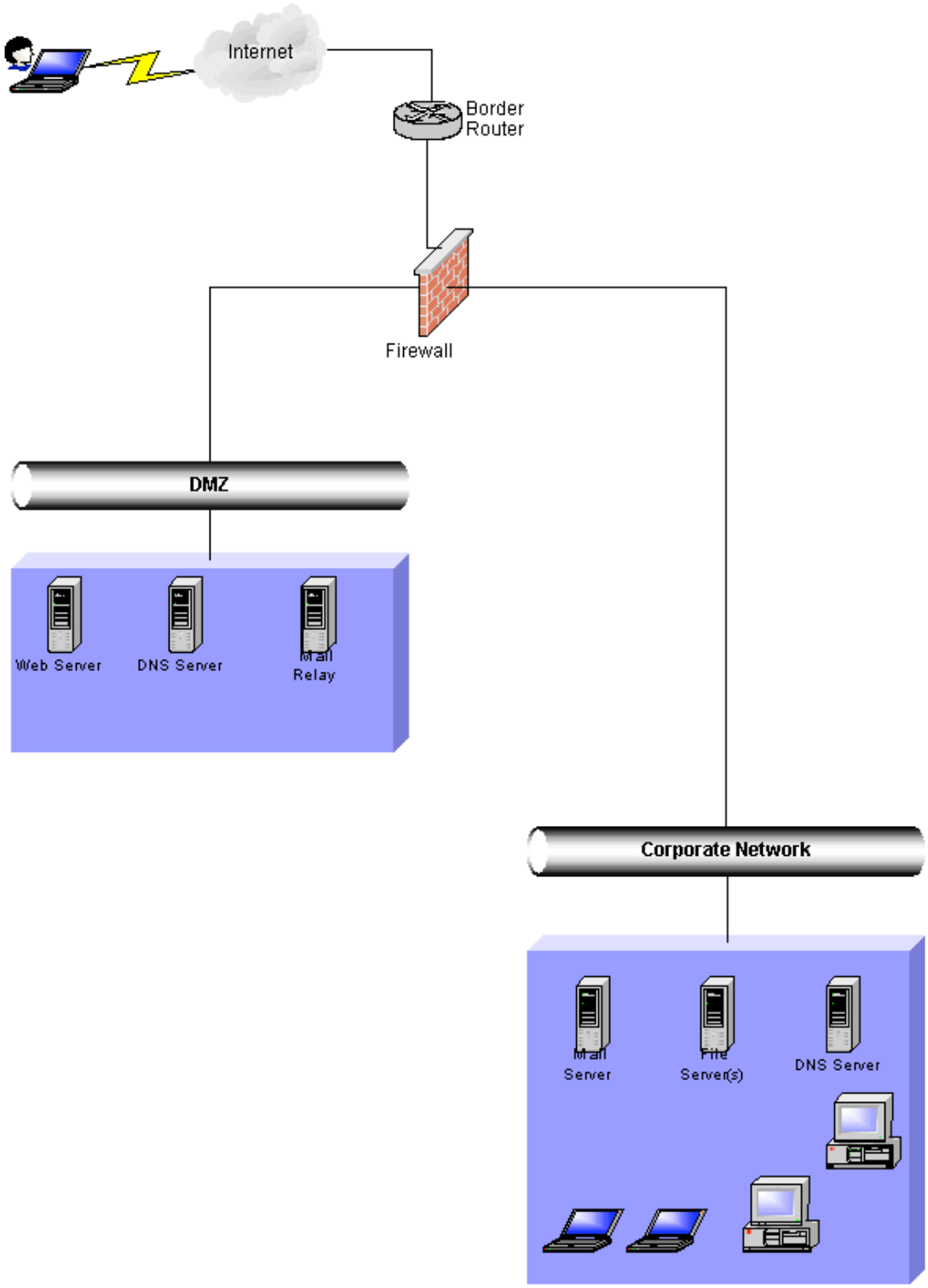
- Network Architecture
- Host Hardening
- Change Management and Detection
- Application & Database Security
- Disaster Recovery

Network Architecture

Probably the biggest mistake companies make when they decide to do eCommerce is using the same old security methods and systems they used before. By this I mean, if in the beginning you had a corporate network protected by a firewall and you were hosting a website with static content and maybe an email server, the security measures you had in place most likely won't suffice when you deploy application and database servers to enable transactions on your site. The physical equipment and architecture of your network need to evolve in order to facilitate secure transactions and to protect the integrity of the data on your commerce servers. Figure 1.a below shows a common

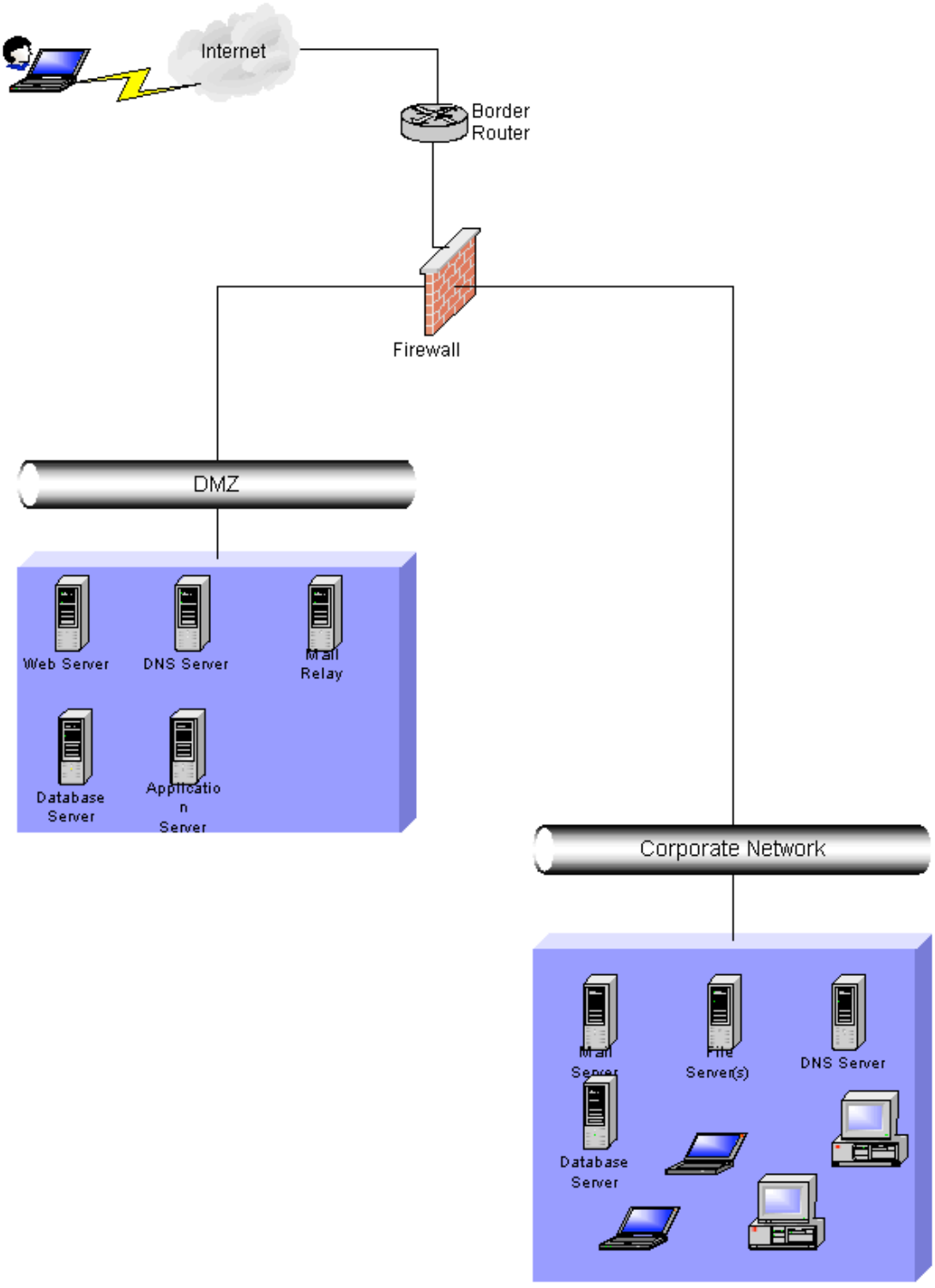
network layout used by many businesses that only need limited connectivity to the Internet.

Figure 1.a



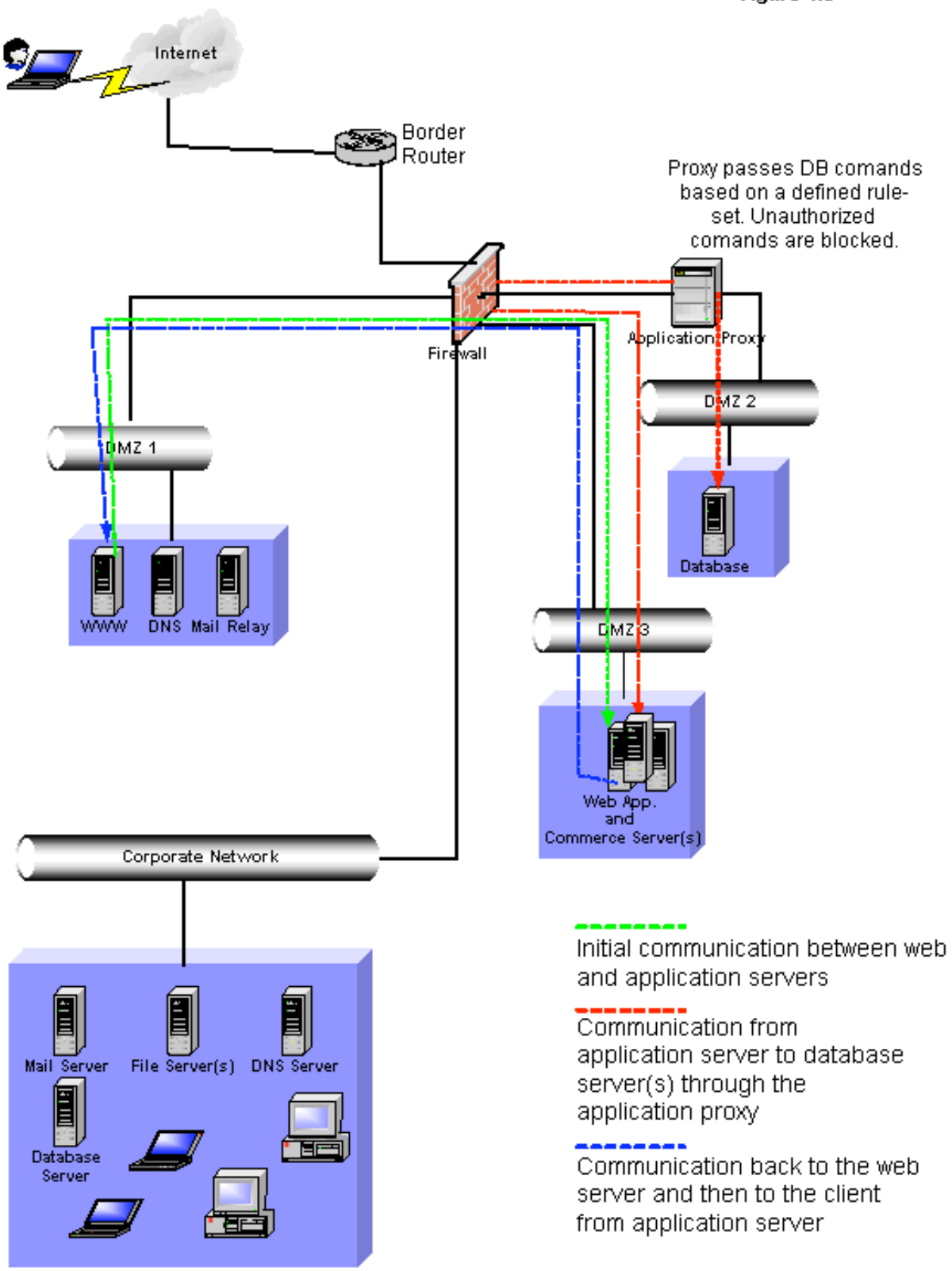
Notice that, in figure 1.a, there is no segregation of critical systems that are exposed to Internet traffic. If we expand on this architecture and deploy our commerce servers without changing the layout we end up with a network like the one depicted in figure 1.b.

Figure 1.b



With the above configuration, if the web server becomes compromised it will be very easy to launch attacks against the database and application servers because all of these machines are on the same network. One could put each of the servers on their own VLAN but as noted in other research, VLANs were primarily developed as a way to increase network performance and only as a secondary benefit to offer more security. In fact, if VLANs are deployed incorrectly, serious security holes can be created (1). Figure 1.c below is a more appropriate architecture because it incorporates segregation -- multiple levels of defense, which brings us closer to our goal of building a defense-in-depth model. In order for an attacker to successfully compromise your database server he would have to first gain control over the web server, he would then have to negotiate the firewall and gain control over the application proxy. Only then would he have the opportunity to start working on compromising the database server.

Figure 1.c



Note the layers of defense we have built into figure 1.c. A firewall and/or proxy device separates each of the critical systems. More layers of defense make an attacker's job much harder if the goal is to compromise multiple systems. Part of the reasoning behind this architecture is that the example company might have sensitive information stored on the database servers such as customer data, order history, payment information and so on. Chances are that even though the web server is an important part of this company's ability to do eCommerce, it is not as critical from a loss standpoint when compared to losing the database. Various architectures could be implemented in order to increase both security and performance, but the point behind this is to illustrate a basic defense-in-depth model. Anti-virus, IDS sensors, a sound logging procedure, strict change management policies and a thoroughly tested and well planned disaster recovery procedure are just some of the things that need to be considered when building defense-in-depth into an eCommerce solution.

Host Hardening

We have incorporated system segregation and put obstacles in the path of the would-be hacker but we haven't given thought to one of the main premises of network security and that is -- assume that it's not a matter of if, but when your network and systems will be compromised. This is where good host hardening, change management and anti-virus come into play. It is beyond the scope of this document to lay out in detail all of the possible steps one could take to harden the operation systems of the servers on a network, but I will cover some of the major issues. As stated many times by Eric Cole in the last SANS conference I attended, one of the most important things for systems and security administrators is to "Know Thy System". If you don't know how the operating system works, what its vulnerabilities are and how hackers exploit them, you aren't going to get very far in the host hardening process. In this document I will relate most of the hardening principles to Windows servers but many of the steps also apply to the various flavors of UNIX that are available.

Before you begin the hardening process I recommend that you build a server profile sheet for each system and thoroughly document the steps you take. This can save you time later when you need to do maintenance or when troubleshooting connectivity and application problems. You should never connect the server to an un-trusted network while you are installing the OS and performing the hardening process. This can most certainly include the corporate network. Place your servers on a separate network to perform vulnerability scans and system audits so that you know you are working in a controlled environment. After getting the OS installed you should start out by disabling any unused services and network protocols. Next you should make the appropriate registry changes to insure that critical keys have tighter DACLs to prevent unauthorized tampering. Lock down the permissions on the file system, paying close attention to executables such as cmd.exe, regedit and regedt32.exe, telnet.exe, etc. Make sure only user accounts that *must* have READ and/or EXECUTE permissions are listed in the security properties of these files and folders. Turn on auditing of critical events and use appropriate log file settings. It is strongly recommended that you disable Client for Microsoft Networking and File and Printer Sharing if the server you are building will be connected to the Internet. I can't think of a legitimate reason to have such services

running on a production server. Make changes to the User Rights settings to only allow the most limited amount of access that is needed for the server to perform its duties. Developing a "least privilege" or “default-deny” stance will save you many times over. Download and install all relevant patches and subscribe to several of the security mailing lists so you can stay current on the latest vulnerabilities and exploits for the operating systems you have deployed. Windows 2000 has several new features including Kerberos and IPSec that can be implemented to further increase the security of both the operating system and the network communications on your networks. Be aware that using these features adds administrative overhead and complexity to the overall operational duties and if you are not familiar with these technologies, improper settings can decrease the security and performance of your production systems. Host hardening not only encompasses changes made to the operating system itself but it also applies to protecting your servers from malicious code. You should also implement an anti-virus solution in your production network and keep it updated. Many of the latest Internet worms are detectable using quality anti-virus software.

Change Management and Detection

As mentioned in the Executive Summary, there are generally two types of attacks on Internet facing systems these days – overt and covert. Usually an overt attack is meant to make a statement and/or to cause significant damage to systems. Examples of these types of attacks are website defacements and Denial of Service attacks -- easily noticed by both the victim and the public. A covert attack uses a more stealthy approach. The reasons for this type of attack can range from corporate espionage to blackmail. Chances are, if you are in charge of network security at your company and someone places distasteful content on your website you are going to hear about it real quick. But will you or anyone else be able to detect a more stealthy attack? If someone has installed a root kit on one or more of your servers and is stealing corporate data and at the same time taking steps to insure that your compromised server is running smoothly how would you know? The answer is simple; if you don't check the server logs and you don't have some sort of monitoring in place you won't know. You'd be surprised at how many systems and security administrators *do* turn on auditing and logging but never bother to check the logs. This is where diligence and a product called TripWire for Servers come into play. TripWire is a package that includes a management console and a server application. Tripwire for Servers monitors all file changes -- regardless of whether they originated inside or outside of your organization and also identifies changes to system attributes including file size, access flags, write time, and more. You can quickly assess the impact of changes using Tripwire for Servers' easy-to-read reports (2). The console enables the administrator to view logs from multiple systems, both UNIX and Windows. More information about TripWire can be found at www.tripwire.com. I can't stress how valuable an effective change management and system integrity strategy is. If you have no way to monitor system changes and you rely only on the limited protection that firewalls and application proxies give, you are setting yourself and your company up for potential disaster. I wanted to mention one more technology I think will play a major role in a good defense-in-depth strategy, the Application Firewall. One company, eEye Digital Security, has developed a solution for servers running Microsoft's Internet Information Server called SecureIIS (3). SecureIIS

is an application firewall that operates on the application layer of the OSI model and has the ability to detect hostile connections to the web server based on information contained within the data portion of the packets. Visit www.eeye.com for more information on SecureIIS. Microsoft has also released a tool called URLscan that works as a filter for IIS web servers to protect against attacks currently known to exist. URLscan is configurable and I've experienced good results, especially against the recent CodeRed and Nimda attacks. In summary, hardening a server involves much more than merely locking down a few file permissions and disabling unused services. Organizations must incorporate a full suite of tools and employ individuals that know how to use them in order to keep constant watch on the server and network environment and hence maintain a reasonable level of security.

Application & Database Security

While working for a large international company on a team that was charged with finding and deploying an eCommerce solution, I had the opportunity to talk with both sales and technical representatives from the largest vendors of application and commerce server software. We had some great breakout sessions and we all learned about each other and what our company's requirements were and everything was good -- except getting concise, solid information on each vendor's product security was a nightmare. They all talked about features, functionality, return on investment and so on, but when asked about what security measures were built into their products they either had to "get back to me on that" or they mumbled something about SSL and that was about it. That was quite disturbing, considering that if a company is doing business over the Internet one of the most important things *is* security. Online marketplaces need to concentrate on more than just having a good firewall and strong host security. The security features built into the application server software are just as important as the two aforementioned subjects. SSL is great but what about how the application server software is coded? And how is that sensitive data stored once it is passed to the server? It is beyond the scope of this document to go into any great detail about particular exploits for these vulnerabilities but I will give an overview of a few known problems that can occur with common application server software.

Session IDs

In most eCommerce applications, each session is kept track of by the server using a unique string of characters known as a session ID. This ID is how the server differentiates between users as they navigate the site. At first this sounds like a pretty secure way of managing sessions and it seems to solve the issue of one user jumping into another user's session. It sounds good until you begin to perform some tests on the system and discover that each time you log into the application your session ID is incremented by 5. So you do more testing with various test accounts and discover by performing a little prediction you are able to hijack other users' sessions by using those predictable session IDs! The point here is that when you are evaluating new or existing software for performing transactions keep in mind that predictable session IDs can be a serious security hole (4).

Source Code Exposure

The ability for an attacker to issue requests to an application or web server that return the source code is commonly referred to as a Source Code Exposure exploit. There are many products that are vulnerable to these types of attacks and the information gathered from such an attack can be a gold mine for perpetrating other attacks. If an attacker can download the code that makes your site do what it does they could possibly learn information about related systems, password storage locations and methods for passing them to other services, file locations and how the site was written in general. For this reason it's a good idea to stay current on the latest advisories for the application server software your company is using, and if you are developing your own application code in-house using JSPs or the like, you should encourage the developers to keep security in mind when building the code (5).

Cookies and SSL

When a user attempts to login to a commerce site, some set of credentials (usually a username and password) are presented to the server in order to perform authentication. At this point, cryptographically strong, random session ID cookies are generated and passed to the client's web browser by the server for future authentication. How this is handled greatly depends on the server software being used, but in general there are similarities between most vendor products due to standards. Your site is using SSL and there's no way for an attacker to read the data exchanged so everything's ok right? Most security administrators and application developers recognize that for authentication, usernames and passwords are sensitive information and should be handled as such, but some don't realize that session ID cookies should also be protected to prevent attackers from taking over a given user's session. If the initial login credentials are passed between the client and the server using SSL but the remainder of the user's session relies on a cookie for authentication, transmitted in clear text, there is a risk of an attacker being able to sniff the cookie data off the wire and subsequently take over the user's session. Even if the entire session is encrypted and the cookies are sent over the encrypted link, an attacker, if patient and knowledgeable enough, can spoof server responses and direct them at the client browser that would cause the client to transmit the cookie over an unencrypted link to www.fakeserver.com. Once the attacker has the session ID cookie he could then launch a denial of service attack against the client and then take over his session with the server (6).

Database Security

Database servers are the foundation of most eCommerce solutions because they generally hold the data that is used to perform transactions. This data usually consists of customer and payment information such as addresses, names, account numbers, credit card numbers and so on.

That is why it is absolutely critical to protect the database server at all costs. There are many methods of protecting database servers from encrypting the data that resides on the actual disks to using complex authentication methods. One simple way to offer protection is to place the server on a separate network segment so that connections are made through a firewall or proxy server as illustrated in figure 1.c. Depending on which vendor's product you choose you will have a range of options for adding security to your database server, and you should seriously consider the security features of any product when deciding on which database to use.

Disaster Recovery

Disaster recovery can be defined as the ability to recover data and to restore functionality after severe damage or total loss of systems has been sustained. Such damage can occur as a result of natural occurrences such as floods, fires or tornados, system failures due to faulty or ill-configured software or hardware, malicious attacks perpetrated by employees or persons unknown or unintentional damage caused by employees or persons unknown and acts of war. However a disaster manifests itself, a plan must be in place and known to work in order for all of the data and systems to be recovered in a timely manner. Part of any sound disaster recovery plan is a good backup and tape storage method. Backup media should be stored off-site in a fireproof container, preferably at a location that is resistant to weather, fire and acts of war. There are many companies that specialize in media storage and it's a good idea to talk to several of them before making a decision on which one to trust your backup data to. The time and frequency in which you perform backups as well as the backup software and hardware you employ are also very important. If you are running critical systems and only back them up once a week you are at high risk if a massive failure occurs. Imagine your company losing days worth of transactional data and how much money that could cost. When choosing backup software and hardware insure that this is an area you don't cut costs on if at all possible. Choosing quality software and equipment can mean the difference between disaster recovery and just a plain old disaster. Some corporations even choose a concept called co-location as part of their disaster recovery plan. In this scenario the company has its production systems in one location and a set of identical systems on stand-by in another location. In the event of a disaster the stand-by systems get brought online until the production environment can be replaced. If you don't test your disaster recovery plan how can you count on it saving you when you need it most? I can't tell you how many systems administrators have felt the dread of loosing data and then trying to recover it from backup media that is either faulty or contains corrupted or even no data. Test your backup media periodically to insure that you actually have the data you think you do.

Final Thoughts

Developing a comprehensive defense-in-depth strategy is a long and difficult

process and it only gets more complex as your company adds services and systems to their eCommerce solutions. There are so many pitfalls and caveats that you must watch out for, and the skills and resources available to protect your network against attacks are sometimes hard to find and even harder to implement. If you look closely at some of the latest successful attacks on systems you will see that often the attackers are youngsters that got their hands on a few tools, yet they know very little about network topologies, programming or even the operating systems they have compromised (7). It is the open availability of hacking tools and the overwhelming number of vulnerabilities in most operating systems and applications that are a major threat to companies. Keeping a network secure is to embark on a constant vigil, a journey that will most likely result in many sleepless nights for the security administrator. It's important for businesses to recognize the threats that exist and to do their part to help the security community and yes, the Internet as a whole, by taking the challenge of securing their systems, both internal and external. Without the cooperation of companies around the world, the problems of rampant worms and viruses, intrusions and compromised systems will never end.

Bibliography

(1) Tyler, David “Are there Vulnerabilites in VLAN Implementations? VLAN Security Test Report” July 12, 2000.
URL: http://www.sans.org/infosecFAQ/switchednet/switch_security.htm
(2) TripWire, TripWire for Servers product information
URL: http://www.tripwire.com/products/servers/index.cfml?
(3) eEye Digital Security, SecureIIS product information
URL: http://www.eeye.com/html/Products/SecureIIS/index.html
(4) HX “WebSphere Cookie and Session-id Predictability” October 4, 2001
URL: http://neworder.box.sk/showme.php3?id=5708
(5) HX “IBM WCS JSP Source Code Exposure” May 30, 2001
URL: http://neworder.box.sk/showme.php3?id=4336
(6) HX “Remote Retrieval of IIS Session Cookies from web browsers” October 29, 2000
URL: http://neworder.box.sk/showme.php3?id=3153
(7) The Honeynet Project “Know Your Enemy - The Tools and Methodologies of the Script Kiddie”
URL: http://project.honeynet.org/papers/enemy/