Global Information Assurance Certification Paper

Securing E-commerce:
An Overview of Defense in Depth

Scot Hartman
March 2001

Introduction

One of the mantras of the network security industry is "Defense in Depth". The idea is that any one point of protection may, and probably will, be defeated. If the protection is multi-layered, the ability for an unauthorized person to gain entry and then do damage or exploit the weakness is somewhat mitigated. What you want to avoid is the hard-crunchy-outside / soft-chewy-middle syndrome.

While there are no guarantees of perfect security once you decide to connect your systems to a non-secure network (like the Internet), you can raise the level of work required on the part of would-be intruders. Raising the effort-required bar and lowering the amount available to steal or damage at each level, helps make you a much less palatable target. When you're walking in the woods and you come across an angry bear, you don't have to be able to outrun the bear, you only have to outrun the friend walking with you.

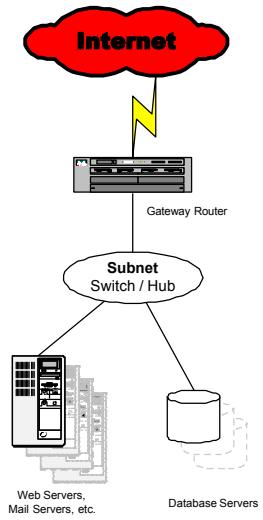
Network security has become a hot topic over the last few years; helped along by the rush of commerce to ply their wares on the Internet and the well-publicized exploits of crackers taking advantage of the new targets of opportunity. The resulting onslaught of security tools, both open-source and commercial, each claiming to have the answers to the various security holes, is enough to makes one's head swim. How does it all fit together? What should I do first?

The intent of this paper is to try to show a general overview of why defense in depth is important and where some of the pieces can be placed to protect e-commerce. Some of the pieces are tools or products designed to help provide security. Others are simply to follow best practices and avoid unnecessary exposures. This is by no means meant to be an exhaustive list of the tools, nor an example of the only architecture you can employ, to deliver e-commerce securely. It is only meant as an example to show how some of these tools / techniques can fit together into the larger whole that defines defense in depth.

A Common Problem

Many e-retailers start out with the same problem. They desire to have an Internet 'presence' but may lack funds, experience, time, or even a clear plan. If the core competency of the business is to build the mythical widget, but they want to sell it on the Internet, they have stepped onto a new field with a steep and treacherous learning curve. The problem is, most of them don't believe it. How hard can it be to buy a couple of servers, design a flashy web-page, plug it into the Internet, and sit back to roll in the dough? The sheer difficulty of 'specing' out a Sun server or a Cisco router, installing the operating system, or seeing the price of the equipment may be enough to start the panic. Even established companies with larger budgets may try to rush to market with a half-baked solution.

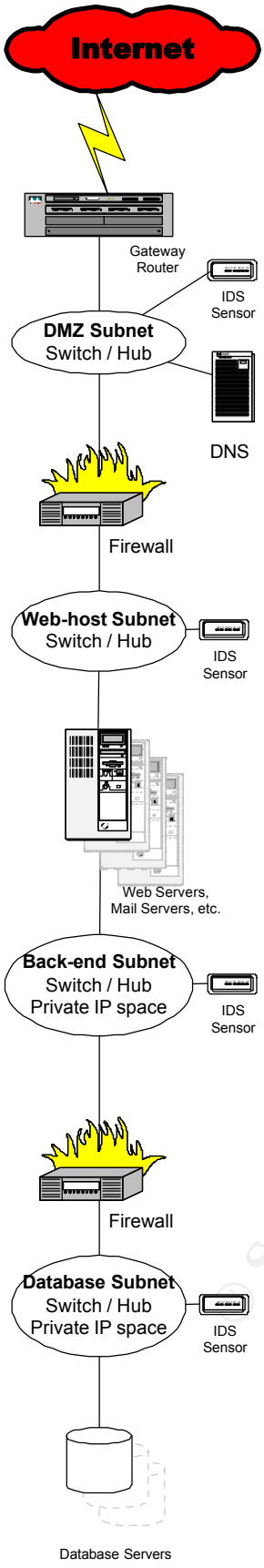
There is a large industry of consultants, co-location facilities, and value-added resellers who, for a fee, are more than happy to help. Even with these services (maybe in spite of or possibly, in some instances, because of), many e-sites may end up with little more than a subnet directly connected to the Internet. The router may or may not be under their control and it may or may not have any filtering enabled on it. One subnet houses their entire infrastructure, the servers may be little more than unsecured default installations, little or no security measures are taken at the host or network level, and they may be remotely managed with non-secure connections over the Internet. Some may upgrade this to include a firewall but take no other security precautions. This structure puts all the security measures up front and hopes that it is enough. "I'm protected, I have a firewall."



Defense in Depth

During an on-line thread, Lance Spitzner used, and subsequently defended, an analogy comparing network defense to a medieval castle. When asked if his defensive model would lead us to repeat the mistakes of the French during WWII, he adroitly countered that the Maginot line is better compared to organizations that rely solely on a single static defense, such as a firewall, to keep them safe. Firewalls are an important security measure, but should never be relied upon as a stand-alone solution. Once circumvented, they are useless. Castles are a nice analogy because the designers planned for the eventuality that each layer could be breached, and built successive layers to allow continued defense. Network security should also assume that no single layer will protect a system. You can also augment this analogy using the US military's concept of interlocking fields of fire. Multiple tools can be employed whose protections may overlap

in functionality, making it harder to defeat them all.



Layered Architecture

One approach to increase security is to layer the network architecture. The concept is similar to the old saying about not putting all your eggs into the same basket. Leveling separates the potential target systems so that a compromise of one subnet does not give free access to different types of systems. An example is to separate web servers from database servers. Since web servers need to receive connections from the Internet, they provide an obvious target for attack.

Depending upon the type of attacker, the web server may be the primary target (nuisance attacks such as vandalism or denial of service). Or the attacker may be looking to steal information from you. If the web server were compromised, it is nice to have another barrier protecting the sensitive information stored on the database server. Separating the architecture into security zones by function and by level of protection desired makes it harder to exploit.

The numbers of layers and the technologies used to separate them is only limited by the designer's imagination. A variation could have the mail servers and the web servers residing on different subnets to prevent an attacker from using an exposure on one to gain access to the other. It is important to note that, for an attacker to get to any layer, he/she must first get through or bypass all the previous layers. If proper care is taken to ensure that only necessary traffic is allowed between the networks, it will make it much more difficult for an attacker to peel the security onion.

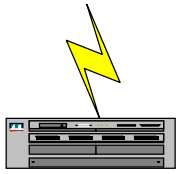
For example, if the web servers only need to listen for http and https requests, then the firewall or access control list on the router should only allow this traffic through. Traffic between the web servers and the database servers should be limited to the specific services needed and only between the specific addresses needed. Some additional security can be gained by using private IP space and by keeping the routing table 'dumb' on the router/firewall between the web servers and database. If the router/firewall only knows about the private, directly connected subnets, it cannot support a direct connection to the database from the Internet or be used to route information from the database directly to the Internet.

Once you have a layered architecture, you can augment it with other security measures to further enhance the protection.

Border Protection

The border router may be under the direct control of the e-retailer or possibly it's ISP / collocation provider. Regardless of who controls it, it is a prime place to start the defense in depth. Even if you plan to have a firewall, the border router can be used for initial protocol filtering, spoof filtering, and assistance with denial of service protection. Filtering in multiple places with a variety of tools allows you to take advantage of best-of-breed capabilities and best practices on where to filter.

Most routers can implement simple access control lists (ACLs) with little or no effort. ACLs are the lowest level of firewall technology but can be used to deflect a good portion of the garbage before it enters your DMZ. (Various sources define a demilitarized zone differently. Some purists may argue that any filtering on the router would make this a 'screened subnet'. So be it.) This subnet may be more open than your eventual web server subnet, possibly because of the presence of a public DNS server, a honeypot (a system intended to draw attacks as a diversionary tactic or to gather information for some reason or another), or some other system that you don't feel the need to protect with a full firewall.



If nothing else, the border router should be configured to help prevent DoS (and DDoS) attacks by following steps outlined by SANS to filter out private, reserved, and invalid source IP addresses. The router should also be configured to stop directed broadcasts.

Some vendors also allow filtering based upon thresholds that may help prevent some DoS attacks by allowing you to set limits on certain protocols. This is still fairly crude but some are working to create more sophistication and allow the thresholds to be 'learned' through normal usage patterns and refined to allow selectivity by source and destination. To work best (once they are improved), these filters should be employed as high in the architecture as possible. To prevent a flooding DoS attack, you need to be able to filter it out at a point where you have enough bandwidth to receive and drop the unwanted traffic with enough left over to allow desired traffic through. This is best done as early as possible.

Firewalling

Oh, the mighty firewall. If it seems that the concept of defense in depth lessens the importance of the firewall, this is not the case. Its role is critical. The firewall should serve as the logical choke point for network traffic into and out of the subnets it protects. The firewall's usual position is to serve a form of perimeter security but can be placed between any subnets to control traffic.

There are many forms of firewalling from simple packet filtering to application layer

gateways and stateful inspection. There is usually a religious battle revolving around which type is best, where to position them, and what type of platform to put them on. The purpose of this paper is not to delve to deeply into this one subject. Suffice to say, there is an abundance of information available on deploying and configuring firewalls. Starting points on the background and the types of firewall technology are available (good one at http://www.avolio.com/apgw+spf.html) and the CERT Coordination Center has a FAQ on deploying firewalls (found at the URL: http://www.cert.org/security-improvement/modules/m08.html). The decision on the firewall technology used usually comes down to many decision factors. Some of the most influential are usually security (can be dependent upon the type of technology used), performance, manageability, and cost.

Important to defense in depth is that, whatever firewall technology used, it should be configured properly. Only needed traffic should be allowed through, the filters should be as specific as possible, both ingress and egress should be controlled, and the firewall itself needs to be securely managed and kept up to date on patches/upgrades to prevent vulnerabilities.

Allowing traffic onto your network that you do not need is taking a pointless chance. This is usually an act of laziness to not spend the time needed to find what protocols or ports you actually need. The firewall is only as good as it's filter set. Take the time to research what is really required.

Filters should also be as specific as possible. Permissions for inherently dangerous traffic, such as rules that allow remote management should always be as specific as possible (more on this in Secure Management section). But less obvious threats are often overlooked. If you have different systems for your web site and your mail, don't just allow web and mail traffic to the entire subnet. Specify mail traffic allowed to the mail server and web traffic to the web server. Allowing SMTP traffic to your web server may not seem to be all that large of a risk, but it is an unnecessary one. If a new web server is built, care will probably be taken to ensure is resilience to ports 80 and 443 by loading the latest web server code and patches. But since it isn't intended to be a mail server, a vulnerable default version of Sendmail may accidentally be left running from the base install. With a tightened firewall policy (and good system auditing discussed later), this mistake may not leave a gaping hole in your security. In this way, defense in depth can sometimes provide you with an 'Oops Shield' that can allow you to make a mistake at one level that is covered by another. Never count on it; diligence is it's own reward.

The firewall should filter both ingress and egress traffic. To most people, the major concern is protection from inbound traffic. Even if they tighten the inbound filters correctly, many are guilty of not paying the same attention to outbound filtering. The usual reason is laziness or the belief that allowing everything outbound cannot cause them any security problems. Outbound filtering has many uses, among them preventing a successful attacker from using your system as a launch-point to attack others on any protocol/port of his/her choosing. This is partially good netizenship and partially self-

protection. Coming is the day when laws may hold people/companies responsible for their unwilling (ignorant) participation in network attacks. It is best to take proactive measures to never allow yourself to be placed in that position. Outbound filtering may actually help prevent a breach to begin with if the attack tool relies upon the protected system to start a service or a connection on a protocol / port that is prevented outbound by the firewall. Egress filtering can also lower your attractiveness as a target by lowering the value of compromising you network. If the outbound freedom is tightly restricted, a successful attacker will not value it as much.

As with any system, the firewall needs to have the latest patches and upgrades. For software-based firewalls, this applies the operating system and the firewall software. The firewall also needs to be securely managed to limit the possibility of someone breaking into itself. As the protector of the network, the firewall is a juicy target for intruders. Managing the firewall via the Internet interface is probably the least desirable situation, especially if the management connection is of questionable encryption strength or can be spoofed. Taking care to not allow management ports to be available to the Internet also helps prevent fingerprinting the type of firewall used. (This information can be used by attackers to take advantage of any known vulnerabilities of that specific type of firewall.) Managing from the protected subnet is better, but can still leave an opening. A clever attacker can compound a successful breach of a protected system by then attacking the firewall from within. A better approach is to use out-of-band (OOB) management or direct console access. (Console access may seem to be the most secure method but makes it harder to regularly check the firewall and many command line interfaces are prone to configuration mistakes. Management is discussed further later.)

Network Based IDS

A relatively new tool in the network security kit is Intrusion Detection Systems (IDS). This tool can be very powerful if properly wielded. Unfortunately, IDS has become just a buzzword to many and people ask for it without really understanding what it is, what is entails, what it can provide, or what it can't.

In a nutshell, IDS attempts to ferret out anomalous traffic that identifies some specific event, such as an attack. After detecting the event, the system can be configured to simply log the activity, send out an alert, or possibly even take some action. The most common form employs a signature matching strategy. The IDS watches the bit stream, trying to match traffic against known attack patterns. The problem is, if the signature-matching strategy is too specific, any small change by the attacker will allow it to slip through without detection. On the other side, if the signatures are too general, then there are many false alarms that desensitize the user. Striking the right mix is generally difficult and time consuming.

Part of the problem of using a signature-based strategy is it's reactive nature. Intruders

continually evolve their attacks and, only if someone recognizes a new attack for what it is, signature databases can only be updated in hindsight. This is similar to the way viruses are combated. It may not be the best approach in the long run, but it is one more method of protection and can augment defense in depth.

Deployment of IDS sensors is a personal preference issue. You may choose to only place one on your outermost network to see what traffic is knocking on your door. Be aware that this location will generate a lot of matches. A good strategy is to locate a sensor both in front of and behind a firewall. This can allow for a comparison of traffic and possibly help diagnose any attacks you may see. These sensors should be set up in a way to prevent direct access to it from the network it is monitoring. This can be accomplished by mirroring switch ports or using network taps. The sensor can also be precluded from transmitting data on the monitored net by not assigning an IP or even by physically cutting or shorting the transmit wires.

IDS is good tool to have, but it needs to be properly researched to work effectively (Carnegie Mellon has a good technical report on IDS called "State of the Practice of Intrusion Detection Technologies" see References). When shopping for a solution, be aware of what sets some IDS apart. Be careful to not fixate too much on one statistic without delving deeper into what it really means. (i.e. some vendors claim to have high numbers of signatures but many may be of attacks against operating systems and applications that are not in use anymore. It's good to keep some of the old signatures because old attacks resurface, but of real importance is how quickly they update their lists as new attacks are found.) Care must also be taken to not allow yourself to be lulled into a false sense of security just because you have IDS. They will not detect all attacks and usually demand a great deal of tweaking.

IDS can be a strong addition to security in depth if you don't rely on it too heavily. It is not the silver bullet and should never be deployed as a stand-alone solution. IDS should be considered as a complement to defense in depth provided by the firewalls, border, host-protections, etc. Not as a replacement of them.

Host-based Protection

Since it is always possible that someone may get past the border protection, avoid the perimeter protection, and hide their activities from the IDS, there should be steps taken to protect the server itself. To best accomplish this, each server should be built with the mindset that there is no other protection available (or that they will fail). There are many host-based protective measures that can be taken. Among the most important are operating system and application maintenance (keeping up with patches and upgrades), proper authentication procedures for access, connection monitoring tools, file integrity checking, host-based IDS, and auditing of the system.

Operating system and application maintenance should be the easiest protection to implement, but is often the hardest. Loading the latest OS and applications with all the patches is usually done at the beginning but maintaining diligence to ensure that they stay current isn't. It is important to know what OS and software you use, keep up to date on Bugtraq or CERT advisories, and to apply the necessary patches to ensure you aren't exposed to any newly discovered vulnerabilities. A firewall cannot usually protect a host that has a vulnerability that can be hidden within a permitted protocol/port. (Proxies can help with this by running an emulation daemon that is designed to be more secure and robust, but they themselves may be vulnerable to a new DoS attack. Proxies are also limited in ability by the number of proxy services available from any given vendor and the amount of effort dedicated to each one. There is also usually a performance price.)

Proper authentication techniques are vital. This can come down to having users log onto the system with their own user IDs (never directly as root or administrator) and care being taken to ensure proper authority levels are granted. Actions of users should be logged and reviewed. Different authentication schemes can be utilized such as Kerberos, Radius or LDAP but it comes down to enforcing logins with proper privileges and enforcing strong password usage. In fact, weak passwords are a member of the SANS Institute top-ten vulnerabilities list. Care should also be taken in how a login is achieved. Remote connections should always be avoided across the Internet that use a non-secure medium that could be easily captured and read (such as telnet).

Connection monitoring tools greatly augment a system's security level. Using something like TCP-wrappers allows control over what services are allowed to connect to the server, from which addresses, and on which interface. This is like having a light version of a firewall running locally. The added benefit of this type of tool is that you can better control which services can be seen on any given interface. When coupled with the layered architecture, you can ensure that only the ports you want the world to see are available on the Internet-side NIC and allow the back-end network to carry the other traffic that may be more vulnerable or has protected information.

File integrity checking tools, like Tripwire or YASSP (YASSP also serves other functions), can help determine if the layers have been broken. This can be viewed as somewhat of a reactive, rather than proactive, measure. But file integrity checking can help to limit whatever exposure you may suffer from a break-in by allowing you to react sooner and to help diagnose the weakness to correct it for the future. This type of tool is helpful to give assurance that your system has not been tampered with, or can be critical to help determine the scope of any confidentiality or destructive breaches you may suffer.

Host-based IDS have many of the same strengths and requirements of network-based IDS but are focused on traffic destined for the given server. This is one more layer that an attacker would have to breach.

An important last point is to perform a system audit when a server is deployed and to periodically re-audit the system. An Internet connected system should have only the

daemons running that are absolutely needed and should be up to date on patches. During the audit, if it is running, you should either know why or research to find out. This is the point where you can catch the unwanted Sendmail daemon and shut it down. Running periodic re-audits are important to ensure that nothing has breached all of your other defenses. A good rule of thumb is to take a snapshot of the services running when you conduct your first audit and compare them during subsequent checks.

Cryptography

An important technology to security is cryptography. The ability to encrypt traffic to protect against prying eyes allows remote access to be done with a measure of security. However, security through encryption is never an absolute. All encryption can be broken, eventually. The key is to stay ahead of processing power and to limit the amount of information that be gained by cracking a single key. An essential concept to understand for applied cryptography is that it is usually broken into two components: encryption algorithms and encryption schemes.

The algorithms are the math-based piece. They are usually given strength values measured by the key size in bits. Care must be taken to not always assume that key size is an absolute value of strength. This can be like comparing apples to oranges because some algorithms can actually be weaker even if they have a longer key. Algorithms can also be open or closed. With a closed algorithm, the strength measurement should always be suspect. Since it is hidden, no one has had a real chance to test its mettle. If there is a fundamental flaw (or if it is laughably simple) that allows it to be cracked easily, all protection is lost. If it is open and allowed to be beaten upon, and eventually cracked, its strength can be gauged by how long and how much processing power it takes. Since any encryption can be cracked, your best protection is simply the economics of time and resources: ensure your data is encrypted with an algorithm that takes long enough to break that makes it not worth the effort to crack it. If you want to ensure you use a secure algorithm without having to personally evaluate them yourself (maybe your math, like min, isn't quite up to it), then you can look to the US Government standard for guidance. The old standard, DES, is being phased out, 3-DES is the current standard. The newly selected AES (Advances Encryption Standard) called Rijndael is in its 90-day comment period and is expected to gain acceptance by this summer. The standards continue to evolve to get stronger as relative computing power continues to increase and its price continues to decrease. The idea is to stay ahead of the economics battle.

The second part is an encryption scheme. This is basically the method of key transfer; the timing and the types of keys used. This can get complex, but this piece greatly increases the strength of the provided security. An intruder can continue to collect and store encrypted traffic as he/she tries to crack it. Once the key is cracked, if the same key was used, all the data is now unveiled for the future as well as back into the past. Schemes are developed to regularly change the keys to allow compartmentalization of data and to limit

the bang-for-buck for the cracker. The more often keys are changed; the more expensive it is to crack. This does not, however, prevent someone from storing all the traffic against the day when processing power is cheap enough to make it possible to crack the amassed data. As a result, the strength of encryption algorithms and schemes should be chosen to prevent cracking of a usable portion of the traffic within the useful life of the information it contains.

Secure Management

Managing your servers and network devices is an important piece of the puzzle, and the piece that is very capable of rendering all the rest of your efforts null and void. Spending the time and effort to build a layered architecture with strong perimeter protection, network intrusion, and strong authentication is for naught if you manage your routers, firewalls and servers with telnet over the Internet.

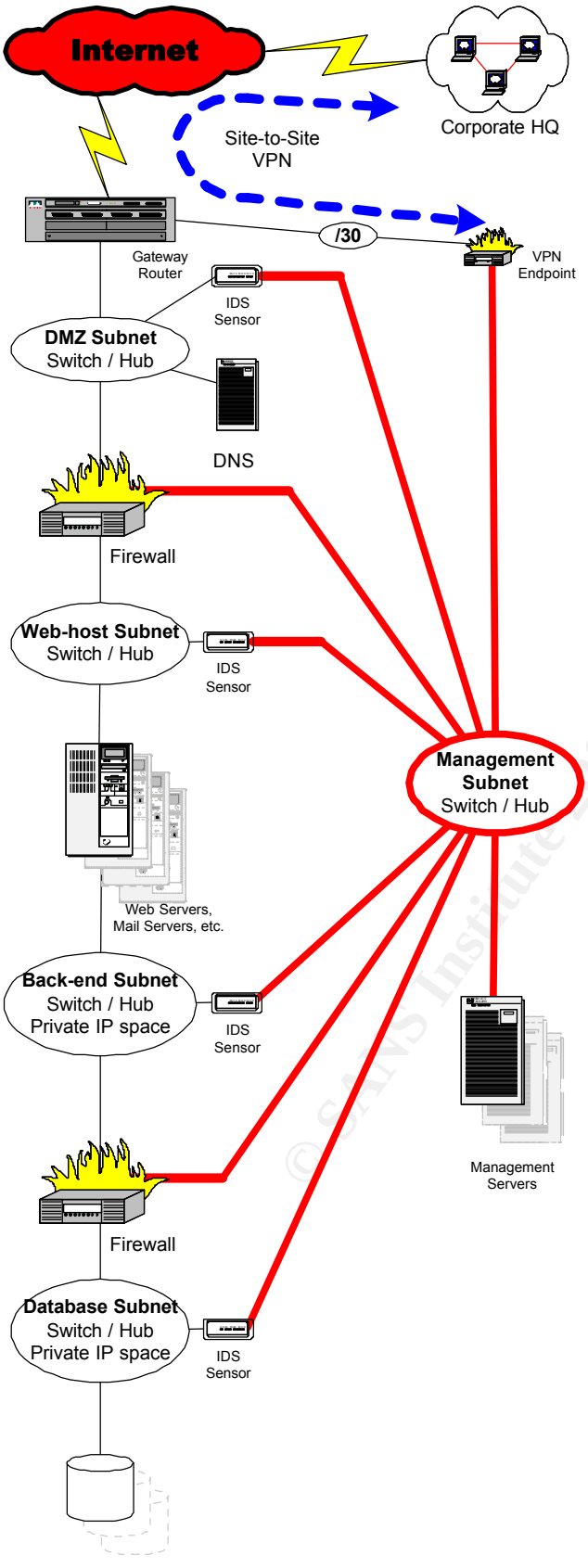
The answer is not to condemn remote management; managing remotely can actually augment security if done correctly. If an administrator is limited to checking up on or maintaining systems only by being physically in front of the equipment, it quickly becomes cumbersome to the point that the work is just not done. Additionally, there needs to be an avenue for logging and alerts to be sent as well as the ability to react in a timely manner; this requires remote access. The key is for the remote access to be done securely.

At a minimum, direct connections to servers should be done using services that encrypt the traffic. Secure shell should be used in place of telnet wherever possible and secure copy instead of ftp. Other software packages, such as Citrix, are also available in an encrypted version or make use of secure socket layer for connections. Many of these tools, even though encrypted, can still be vulnerable to brute force attacks since most only prompt the client for a user ID and password during a connection attempt. Care should be taken to only allow these connections from and to specific destinations. Some management or logging services are inherently dangerous or are sent without encryption and should never be allowed to traverse the open Internet: telnet, ftp, snmp, and syslog are a few.

A more secure approach would be to utilize a Virtual Private Network (VPN) to set up a secure link to the subnets for management. This usage of encryption is usually used to set up an encapsulating tunnel from one protected network to another. (Or from a remote client to a protected network.) This usually takes advantage of an encryption algorithm with a scheme to regularly exchange keys and can mask all traffic traveling through the 'tunnel'. The only thing visible to anyone on the Internet is encrypted data between the two VPN end points: all the services used, source / destination addresses utilized, and the data is hidden. An attacker is left without some of the clues that can help determine which traffic is valuable and which is not. All traffic would have to be decrypted, hopefully at great expense, in the hope that some of it is valuable.

To make this even more secure, the management traffic can traverse the VPN as encrypted payload to a separate management network instead of to the managed servers directly. From here, access to the servers and network equipment can be from back-end networks or out-of-band. This will ensure that those services will not even traverse the Internet-facing subnets nor will your systems be listening on those ports on their Internet-facing NICs. This limits any amount of information available or exposure if the next

farther out layer of protection is compromised.



To show the management piece with a layered infrastructure…

This company has a corporate HQ, with its own security, which is linked to the management subnet through a site-to-site VPN. As a matter of personal preference, the VPN could be terminated in the external firewall between the DMZ and the Web-host subnet. If this is chosen as a solution, care should be taken that the traffic over the VPN doesn't overly task the firewall to the point where it doesn't have enough CPU power left over to handle the Internet traffic. Encryption is very processor intensive and you can quickly provide yourself a self-inflicted DoS attack.

Management servers can be utilized on this out-of-band network to securely access the firewalls and IDS sensors. Access to the web, mail, dns, and database servers can now be done through the back-end connections provided through the firewalls.

This design can be redrawn many ways, depending upon the resources available and the level of security desired. The VPN device, for example, is shown with its own subnet link to the gateway router. This isn't needed. It was done so that it could have a completely different IP address range from the advertised public access subnets. This may make it more difficult to correlate the two. (A little added security through obscurity is fine as long as it isn't your primary defense.) The management subnet could even have a layered design of its own to split out management of the servers from the network devices. The important point is to truly think about where your management traffic flows and how to protect it from being

intercepted. Management traffic can be the Achilles heel in many ways and needs to be closely guarded.

Physical Security

Setting up the best, most-expensive, highest-tech network security is a waste of time and money if you don't control access to your equipment. If it's important, lock it up. If you don't think it's important in and of itself, reevaluate objectively about whether something else that is important could be gained or inferred from it.

If the electronic phalanx you have amassed is truly daunting, but the information you have is valuable enough to steal, then physical security needs to be up to the task. A really solid network defense is not complete if someone can physically gain access your equipment or private networks. Keeping with the theme of defense in depth, your equipment should have several layers of physical security. Controlled access to your building, a secure network room, locked cabinets for the equipment, and maybe a screen to prevent wireless communications from leaving the room. If you use a co-location facility, just as important as customer service, they need to have a way of securing your equipment with a layered approach. Controlled access to the raised floor, an authentication scheme to allow your company to control who is authorized, locked cabinets that possibly require more than one key, preventive measures and monitoring to keep other customers from tampering with wiring, etc.

There are sources available on physical security and combating social engineering. (Do not forget that Mother Nature may also attack you and that disaster recovery should be planned for.) The essential point is to not be so wrapped up in the possibility of attacks from cyberspace, that you leave open a blind spot in real space.

Vulnerability Assessments

Once you have built your castle, how do you know it will hold? You conduct vulnerability assessments. These can be conducted via network tools or directly on a system being assessed. Vulnerabilities should be evaluated at the operating system and the application level. Assessments should also be run from some point external to your network as well as from inside.

Assessment tools are available from either the commercial arena or the open source community and are periodically reviewed in comparison studies (see references on a recent Network Computing report). Assessments can be run in-house, or can be conducted by a third party. A word of caution: do not blindly trust any third party to run vulnerability assessments for obvious reasons.

Assessments can vary in scope and depth from a simple port sweep to trying to break in using any means possible (which may even test your physical security). Before conducting any assessment on a production network, ensure that you have signed permission (from someone authorized to do so). If you don't, ensure that you at least

have an updated copy of your resume (or maybe a lawyer). If something goes wrong, or if someone challenges your legitimacy of conducting the assessment, you may need strong evidence in your favor. Conducting vulnerability assessments are vital, but need to be approached with caution.

Security Policy

A security policy needs to lay out, in writing, the security steps of your organization and outlines who, what, when, how, and why of ongoing actions and procedures. It should be a detailed document that identifies risks, defines steps taken to reduce them to acceptable levels, and outlines the tasks and job-descriptions of those mandated to conduct care and feeding of security. It is both a checklist and a shield. The checklist portion forces the organization to ensure it has performed due diligence to create a secure environment. It is also a shield because it outlines people's roles and responsibilities so that they can point to the document to show legitimacy and direction to their actions.

Conclusion

Defense in depth is more than just a catchy phrase; it needs to be a mentality. Never rely on a single, static barrier to protect you from everything. Design a layered defense that forces an attacker to spend an inordinate amount of time and energy for little gain. You will never have a perfectly secure network (a fact of connecting to the Internet), but you can have a relatively secure network that makes you either not worth the effort (or possibly even dangerous) to attack. This is your goal; let your friend be bear food. Once you have built a well-designed and diligently maintained defense in depth, I offer one last piece of advice; don't brag about it and dare the cracker community to try and get in. The bear might still be hungry, don't dangle yourself in front of him like a sausage.

References (web accessible):

Spitzner, Lance. "Castles and Security" Thread archive from Firewall-wizards@nfr.net, 28 December 2000
URL: http://www.securityfocus.com/archive/15/153918 (last visited 16 March 2001)
URL: http://www.securityfocus.com/archive/15/155837 (last visited 16 March 2001)

Potter, Al. Manager, ICSA Labs "Difference between a firewall and a perimeter router" Thread archive, 1 February 2001
URL: http://www.securityfocus.com/archive/19/159854 (last visited 17 March 2001)

SANS Institute. "Consensus Roadmap for Defeating Distributed Denial of Service Attacks" Version 1.10 - February 23, 2000
URL: http://www.sans.org/ddos_roadmap.htm (last visited 15 March 2001)

SANS Institute. "Help Defeat Denial of Service Attacks: Step-by-Step" Revision: 1.41 – 23 March 2000
URL: http://www.sans.org/dosstep/index.htm (last visited 15 March 2001)

Senie, D. "Best Current Practice" RFC 2644. August 1999
URL: http://www.rfc-editor.org/rfc/rfc2644.txt

Captus Networks company announcement on new products at New Orleans, SANS Security 2001 – 30 January 2001
URL: http://www.captusnetworks.com/denial_service.html (last visited 10 February 2001)
Used as reference for what vendors are targeting.

Avolio and Blask. Trusted Information System, Inc. "Application Gateways and Stateful Inspection: A Brief Note Comparing and Contrasting" Revised: 1/22/98
URL: http://www.avolio.com/apgw+spf.html (last visited 18 March 2001)

Carnegie Mellon Software Engineering Institute, CERT® Coordination Center. "Deploying Firewalls"
URL: http://www.cert.org/security-improvement/modules/m08.html (last visited 18 March 2001)

Carnegie Mellon Software Engineering Institute, CERT® Coordination Center. "Packet Filtering for Firewall Systems"
URL: http://www.cert.org/tech_tips/packet_filtering.html (last visited 18 March 2001)

Allen et al. Carnegie Mellon Software Engineering Institute. "State of the Practice of Intrusion Detection Technologies" Technical report - 24 January 2001
URL: http://www.sei.cmu.edu/publications/documents/99.reports/99tr028/99tr028abstract.html (last visited 18 March 2001)

Wiens, Richard. "Realistic Expectations for Intrusion Detection Systems" last updated March 19, 2001
URL: http://www.securityfocus.com/frames/?focus=ids&content=/focus/ids/articles/expect.html%3F%26_ref%3D1456207040 (last visited 18 March 2001)

Krebs, Brian. "Intrusion Detection Systems: An Opening For Hackers?" Newsbytes - 15 Mar 2001
URL: http://www.newsbytes.com/news/01/163221.html (last visited 17 Mar 2001)

Wilmink, Chuck. Director of the Canadian Centre for Information Technology Security (CCITS) "Rush to get latest computer gear running can lead to lax security" Vancuver Sun – 17 Mar 2001
URL: http://www.vancouversun.com/newsite/networks/5007398.html (last visited 17 Mar 2001)

SANS Institute. "How To Eliminate The Ten Most Critical Internet Security Threats" Version 1.32 January 18, 2001
http://www.sans.org/topten.htm (last visited 19 March 2001)

Donovan, Craig. "Strong Passwords" SANS Practical, June 2, 2000
URL: http://www.sans.org/infosecFAQ/policy/password.htm (last visited 19 March 2001)

Sites that have recommendations for strong passwords
URL: http://dab.nfc.usda.gov/news/nh-2000/nh-0900/password-0900.html
URL: http://www.fin.ucar.edu/it/dsn/userdocs/pswdguide.htm (last visited 17 March 2001)
URL: http://www.microsoft.com/ntserver/security/deployment/planguide/password.asp

Dunne, Paul. "Securing your network: An introduction to TCP wrappers"
URL: http://www.sunworld.com/unixinsideronline/swol-06-2000/swol-06-tcp.html (last visited 19 March 2001)

Litterio, Francis. "Cryptography: The Study of Encryption" Reference site with links on cryptography.
URL: http://world.std.com/~franl/crypto.html (last visited 20 March 2001)

NIST site for Advanced Encryption Standard (AES)
URL: http://csrc.nist.gov/encryption/aes/ (last visited 20 March 2001)

King, Steven et al. Collaborateive Task-Force funded by the NCES of the U.S. Department of Education
"Safeguarding Your Technology" Chapter 5 – "Protecting Your System: Physical Security"
URL: http://nces.ed.gov/pubs98/safetech/chapter5.html (last visited 21 March 2001)

Higgins, Scott. "Physical Penetrations: The Art of Advanced Social Engineering" SANS Practical. February 22, 2001
URL: http://www.sans.org/infosecFAQ/audit/penetrations.htm (last visited 21 March 2001)

Forristal, Jeff and Shipley, Greg. "Vulnerability Assessment Scanners" A review/comparison. Network Computing. 8 January 2001.
URL: http://www.nwc.com/1201/1201f1b1.html (last visited 22 March 2001)

Zuckerbrod, Nancy. Associated Press "Agency Hacked Into IRS E-File System" - 16 Mar 2001
URL: http://www.washtech.com/news/govtit/8351-1.html (last visited 16 Mar 2001)

Farnsworth, William. "What Do I Put in a Security Policy?" SANS Practical. - 10 August 2000
URL: http://www.sans.org/infosecFAQ/policy/policy.htm (last visited 21 March 2001)

References (Books):

Northcut, Stephen. "Security Essentials" SANS Institute Track 1 courseware, January 2001

Northcut, Cooper, Fearnow, and Frederick. "Intrusion Detection Signatures" New Riders Publishing, January 2001.

Stevens, W. Richard. "TCP/IP Illustrated, Volume 1" Addison Wesley Longman, Inc, 1994.

McClure, Scambray, and Kurtz. "Safeguarding the E-Business Network" Excerpts from "Hacking Exposed" Cisco Press, 2000.

© SANS Institute 2000 - 2005 Author retains full rights.