Global Information Assurance Certification Paper

Intrusion Prevention – A Look at StormWatch
Michael Dunklebarger
Version 2.0
February 24, 2002

Summary

Reaction (re·ac·tion) - (noun) 2 : a response to some treatment, situation, or stimulus[1].
My definition - The usual state of intrusion handling. Until recently, network intrusions of various sorts have been handled from the perspective of reaction. First we have the intrusion and then we react.

A new generation of software is now being developed, that will give us the ability to prevent damage caused by intrusions by stopping the intrusion before any major adverse effects can be caused on our systems. This paper will make a comparison between intrusion detection and prevention techniques and examine some of the solutions to intrusion prevention software producers have marketed.

Intrusion Detection

Network-based and host-based intrusion detection software is designed to detect when an intrusion is taking place or when malicious activity has taken place. Most plans for network security are based on the premise that the best we can do is to prepare for problems, monitor the system to detect problems, and then repair the damage. The CERT Guide To System and Network Security Practices recommends the steps listed below:[2]

1 Harden/Secure
2 Prepare
3 Detect
4 Respond
5 Improve

Hardening and securing the system would include the usual recommendations such as installing only the parts of the operating system that you will actually need, installing only the most recent and secure versions of software packages and including any available service packs and patches. Using the deny first-then allow method of granting only those access privileges absolutely necessary to individual users, is also critical. Finally, logging should be enabled so all critical information will be available in case of a problem.

The preparation phase includes the creation of baselines and policies that will allow for dealing with vulnerabilities that are known to exist and also for those that have not yet been discovered. A large issue in this phase will be that of what to do in case of an intrusion. A large part of any security policy will deal with the issues in this phase.

The third step, detect, includes the discovery of anomalous activity on systems as defined by the baseline that are set up for normal activity in the preparation phase. Included in this step might also be reports of problems from users on the system, users from other external systems, or from reports from administrators or vendors describing problems that have been detected elsewhere.

Once a problem has been detected, there usually has to be some sort of response. If there has been an intrusion into a system, steps must be taken to prevent further damage. Damage that has taken place will need to be corrected and contained. Additional action may need to be taken depending on the source of the problem. For example, if the source of the problem is a previously unknown one, the proper authorities or organizations may need to be notified. If the source of the problem can be verified, legal action may need to be pursued.

Finally, information must be gathered to determine what has happened, what caused it, and how the system may need to be modified to prevent a recurrence of the same problem. Policy may need to reviewed and modified, and changes may need to be made in procedures, software and administrative tools that are currently in place.

This intrusion detection paradigm, when applied with the level of commitment necessary, has served the networking community reasonably well for many years. One problem that has surfaced in recent years is that the level of attacks on our systems has been increasing at an ever increasing speed. For example, the number of new viruses that are released each year, has multiplied from about 200 per month in 1995 with about 8500 known viruses[3], to over 60,000 currently. Kaspersky Labs lists the known active viruses each day. A sampling of these listings shows 556 active viruses for the first 2 weeks of January and 528 for the first 2 weeks of February[4]. While there are duplications on these lists, they show that the problem continues to escalate. Many new threats now being seen are multi-dimensional threats that attack a system on a number of fronts at the same time.

Perhaps the greatest problem with this paradigm can be shown with the following example of how a new virus might be dealt with. A new virus is released into the wild. At some point, it is detected by someone, perhaps an alert administrator who discovers the effects of the virus. The new virus is reported to vendors who then analyze it and create an entry for their signature file or create a patch, if appropriate. At that point, the vendors advertise the existence of the new signature and make it available for downloading. Unfortunately, the latency in getting the new signature file installed can be significant, and the impact of the virus will continue. Additionally, new variants may or may not be discovered. The net effect is that costly damage can be done in a short time, causing significant down time and loss of data or business.

It appears to be evident that we need a better way.

Intrusion Prevention

The new paradigm that is being developed currently is referred to as intrusion prevention. The basic theory is quite simple. The system is examined to determine the parameters describing what normal behavior is. Any behavior that does not match the behavior is not allowed. This method of intrusion prevention is being developed in a number of different forms, ranging from protection of individual applications or processes, to the protection of entire networks and hosts. An article in NetworkMagazine.com presents a comparison of a number of software packages that may be used for web server protection. The article makes the following explanation of how the systems work and how they are different in what they do[5].

Most operating systems of today have both a user and a kernel mode that provides a level of security to control access to resources. The user mode can only make use of resources such as hard drives, CPU resources, etc. by going through the kernel mode through the use of system calls. This way, user applications are kept from direct access to vital parts of the computer. Intruders have found ways, usually through the use of flaws or weaknesses in the software, to circumvent this separation and exploit the system. The intrusion protection software packages under consideration perform their protective activities at this level. They examine system calls and determine if the call is part of an expected, normal activity or if it is anomalous and should be blocked from access.

Entercept is an example of an intrusion prevention system that is installed adjacent to the operating system and uses checking of system calls. The operating system maintains a table of system calls and the corresponding function within the operating system kernel. Entercept changes the entries in this table to point to its own drivers instead of the system calls. If the Entercept software determines that the call is legitimate, its driver then calls the original system call. If the call is not legitimate, access is denied. Entercept uses a series of behavioral rules to determine the legitimacy of the call. The database that holds the behavioral rules also includes attack signatures of known attacks as well as generic attack behaviors[6]. Entercept's Console package starts at $4995, web server agents are $1595, and standard agents cost $1295.

Harris Corporation's STAT Neutralizer seems to work in a similar way, by defining correct behavior and not allowing unknown or malicious behavior. STAT ships with a standard set of security policies that can also be modified to fit individual requirements. It does not seem to use signatures to identify malicious behavior. As with other software described in this paper, STAT protects from internal and external threats. The source of the threat is unimportant to the process of protection. Also, since the software works at the operating system level, any programming code being checked will be in a decrypted form[7]. STAT Neutralizer pricing starts with a package of the administration server, 5 server agents and 10 workstation agents for $2995.

WatchGuard ServerLock is an intrusion prevention program that is designed to lock down your servers and protect them against internal and external threats. It works much the same way as the previous examples in that policies are created to protect files, applications and configurations. Again, this protection takes place at the dividing line between the user mode and kernel mode. A major difference in this package is the addition of a 2 mode system. The operational mode puts the server in a locked down condition and fully protects the designated services, even against changes being made by someone with root or privileged access rights. In this way, the system is protected from inadvertent changes made by an administrator, as well as change attempts by someone who has gained root access for malicious purposes. The second mode, administrative, is used to remove the protection from the machine in order to make necessary changes to files, applications, configurations or policies that effect the operation of the software. A kernel-based encryption system based on a 239 bit key is used to secure the access to the administrative mode, and access requires dual signatories as well so that more than one person is aware that changes are being made[8]. ServerLock Windows version costs $1295 while the Solaris version is $1695. A manager program is available for up to 100 servers. Pricing starts at $4995 for a 5-server version.

A number of software vendors have opted to make software packages that are primarily intended to protect the web server application itself, in effect, an application firewall. Standard firewalls can protect against many threats. In theory, applications are written that have levels of security built in to protect the application against exploitation. Protection should be included in an application that will limit the amount of input allowed to a level below that necessary to cause a buffer overflow. The amount of information about the system that an intruder can get through the use of an application should be limited. Depending upon the application, there might be a long list of such protections that should be written into the code. But we all know that that just doesn't usually happen. The next two examples fall into the application firewall category.

Sanctum, Inc. developed AppShield based on the premise that there are four elements in Internet security that help protect our servers. Antivirus protection, data encryption and authentication, and firewalls can do an adequate job of protection of the server, but none protect the web server application itself. AppShield, unlike the previous examples, is applied between the firewall and the web server. It has to examine the information coming in and react to it. The software examines HTML requests as they arrive from a browser. A dynamic security policy is generated that is based on the content in the web page that the HTML request is accessing. The software will only allow information to be passed to the server that matches the requirements of the web page. Some of the capabilities of the system are: only URLs embedded in the web page can be requested, information in drop down menus can't be modified, hidden field content can't be changed, field names can't be altered, the size of input can't be made large enough to cause a buffer overflow, and HTML code can't be included in text responses[9]. AppShield is priced at about $15,000 per server.

The SecureIIS Application Firewall is from eEye Digital Security. The software package is designed to be used with Microsoft IIS and actually installs as a code module within IIS. It provides security functionality that was not included in the original IIS software. The application firewall provides this security by being able to detect common hacking methods. The standard configuration protects against buffer overflow attack and parser evasion attacks which can allow intruders to add their own executable code to input. Also protection is included that prevents directory traversal attacks which can allow intruders to break out of the webroot directory and gain access to other parts of the system. Provision is included to prevent general exploitation attempts as well as to guard against as yet unknown types of attacks. Because of it's location within the IIS software, the application firewall can examine the data before and after it has been encrypted in a secure environment. SecureIIS costs about $495 per server.

eEye also has another software package, Retina Network Security Scanner. This software will perform a scan of a system and discover and disclose any vulnerability it finds. The second part of the software acts like a hacker in that it attempts to use common hacking techniques to compromise the system[10]. The administrator would then apply any patches to close the vulnerability. If none existed, a report would be sent to the software publisher who would notify the application developer of the flawed software. They may then create a patch to close the vulnerability or inform users that there is none and that they should do the best they can to protect their systems.

The context of this paper does not allow me to fully detail all of the available software packages that use intrusion preventions methods, so I have chosen to review this sample group and then concentrate on one package in depth. The package I have chosen to examine more thoroughly is StormWatch by Okena, Inc. Before I examine StormWatch in depth, I would like to look at Okena's idea of what intrusion prevention should be like.

Best Practices for Intrusion Prevention

Intrusion prevention is a new technology, but there already is a variety of approaches and software packages available to choose from. How do we make the choice of what might be best for our situation? Okena has developed a best practices document that may be helpful in making these choices[11]. Solutions to a company's security needs should be based on many factors and there are many versions of just what constitutes a best practice. According to Okena, anyone considering using intrusion prevention as part of their overall security plan should consider at least the following aspects.

Host-based protection. Networks must be protected against intruders at many points including access points to the Internet, access points from remote users, and from other local networks. All of the sources of intrusion are usually protected by firewalls and other methods of prevention that usually do an adequate job of keeping intruders out. However, with increasing speeds, the use of switches, and end-to-end encryption, network intrusion detection systems are being stressed to the breaking point. Also, with the use of a layered approach to security being obviously the best way to operate,

protection is becoming more important at the host level. Most intrusion activity is aimed at the host, so that is where it makes sense to place your best protection.

Real- time prevention decisions. The old saying "an ounce of prevention is worth a pound of cure" should be our guiding principle. Making use of a system that can make decisions in real time to stop destructive behavior is far better than using a system that only detects a problem after the damage has been done and can only act to prevent further damage after the fact. Prevention decisions must be made at the most secure area of our system, at the point where the user level and the kernel level of the operating system interact. It is at this point that actions can be evaluated rather than just evaluating content.

Defense in depth. Defense in depth is widely recognized as being necessary for network security. It is equally important in intrusion prevention for the same reasons; intruders attack a variety of parts of our systems. We need to be able to prevent a breach at every point of communication between applications and the operating system and hardware. Among the issues the system must be able to address is the protection of the registry from unwanted and unauthorized modifications. The file and folder system must be protected from unauthorized changes and access. Communication between processes and applications must be monitored and controlled. Communication between clients and servers must be controlled at the port and protocol levels. Many recent attacks have been aimed at doing more than one type of damage to our systems. The ability to monitor and control many aspects of the system in depth, gives intrusion prevention systems the ability to be able to deal with intrusions that have not been encountered before and have not been otherwise protected against.

Real-time correlation at the agent and enterprise level. Approaches to intrusion detection that recognize an attack on one part of a system as a discrete incident cannot be as effective as an intrusion prevention approach that looks at all parts of the system and can "realize" that perhaps there is a multi-faceted attack taking place and that a number of actions will be necessary to stop it. Looking at a sequence of events can also help to keep false positive alerts at a minimum.

Correlation is also important at the enterprise level. It is important to have the ability to "spread the word" about an attack on one host to others who may also be a target. This notification makes it more likely that additional damage can be contained.

Behavioral approach. Intrusion detection is signature based. New intrusions are prevented by studying old intrusions and looking for recurrences. A behavioral approach does not rely on signatures which are reactive. Instead, behavior in real time is monitored and actions taken immediately.

Flexibility to meet unique corporate needs. Businesses have a wide variety of unique needs that must be accounted for in intrusion prevention. The system must be easily changeable to fit the current and future needs of the business. The customization should be automated to keep support needs at a minimum.

Ease of deployment. Solutions must come in a format that can be easily implemented. Minimal customization should be required and it should be easy to accomplish and interface well with existing distribution strategies.

Centralized event management. A system must be able to maintain centralized logs that can be used to prepare reports and track problems system-wide. Notification of events in progress that require intervention by administration must be provided in whatever format is appropriate.

Platform coverage with support for desktops and servers. The prevention system should provide protection for all platforms used by the corporation and should be useable on servers and hosts in the same manner.

Administration. The system must be manageable throughout the enterprise in a way that is secure and easy. Web management is the preferred method. Policies must be easily distributed.

A consideration not covered in this discussion to this point is the cost of the installation. Any package will need to be cost effective. While the cost of the various packages being considered here varies, all of them are priced in a way that their cost is a small fraction of the cost necessary to repair damage to systems that are not protected.

With all of these considerations in mind, let's take a closer look at StormWatch.

StormWatch

StormWatch, now in version 2.1, was released in April 2001 by Shaun McConnon, the founder of Okena, Inc. and its chief executive officer. Okena is reported to be the leading developer of intrusion prevention software. StormWatch is one of two products meant to be used in conjunction with each other. StormFront was developed to analyze systems and create policies to be used with the StormWatch package that is designed to protect the host or server from intrusion.

StormWatch is an intrusion prevention scheme that is based on the behaviors of individual applications. Rather than try to identify malicious activity or applications as they arrive on a network, a task that has become very difficult on the high speed networks of today, the software is designed to recognize aberrant behavior on the part of applications, and can stop the behavior before it causes damage. While it might seem that the task of mapping the normal behaviors for an application is a difficult one, most of the process is done for the user. The software ships with a set of preconfigured policies that cover many standard applications. These may be modified as needed and new ones may be created using StormFront, a companion product that will be discussed later.

The out-of-the-box policies have been written to be used with Microsoft IIS, Office, SQL Server, Personal Firewall, Instant Messenger, DNS servers and DHCP servers. Policies are included that cover workstation, server and network issues.

The heart of the software is the INCORE (**IN**trusion **CO**rrelate **R**ules **E**ngine) technology. Policies are established that reflect the rules for determining normal behavior. File and network operations are then verified against these policies. If the operations being performed conform to the rules, they are allowed to proceed. If they are outside the bounds of the policy, they are stopped. These actions are performed by the Intelligent Agent that is installed on any workstation or server that needs to be protected. The agent and its operation are invisible to the user and require no intervention.

The INCORE software works at the division point between the operating system's user mode and kernel mode. System calls made by an application in user mode to services available through kernel mode, are intercepted and compared with policies that govern that application. The state of the application is also verified and a decision is made as to how to proceed.

StormWatch has two parts. The Management Console, which uses a web-based user interface, is used to configure policies and transfer them to the agents placed on workstations and servers. It also provides the service of notifying agents whenever a malicious attack is taking place on the network, at which time the agents can keep offending applications from being run. The Intelligent Agents, which come in both a workstation and server version, are installed locally on a machine through a single click network installation process. The agents poll the console on a regular basis to retrieve updates to the policies and to send information used to log events.

Okena presents the following items as protections provided by StormWatch "out-of-the-box."

"SERVER INTRUSION PREVENTION

Generic Server Protection

- Detects and prevents buffer overflows
- Prevents hijacked email or web browser apps from compromising the operating system through attempted writes to the command shell or registry
- Prevents unauthorized writes to system executables - preserves integrity of the OS
- Provides real-time file monitoring (file base lining or integrity monitoring)
- Monitors and enforces which applications can run on the server
- Detects and prevents Trojans
- Detects port scans
- Protects against SYN floods

IIS Web Server Protection

- Detects and prevents buffer overflows
- Prevents attacks from invoking arbitrary commands on a system via a command shell
- Prevents a Trojan from posing as the IIS server (stealing secrets from application)
- Prevents attempted changes to the IIS Web server configuration
- Prevents Web Graffiti (changes to HTML)
- Restricts access to IIS data files to the Web server itself
- Restarts IIS server when not responsive

SQL Server Protection

- Blocks SQL server from invoking local SQL server administrator management tools
- Prevents a Trojan from posing as the SQL server
- Stops attempted changes to the SQL server configuration
- Restricts access to SQL server data files to the SQL server itself (a user, regardless of privilege or level of authentication cannot bypass these policy rules)

DNS Server Protection

- Prevents unauthorized applications from modifying specified DNS data files
- Restricts network access to only permissible applications[12] "

"DESKTOP INTRUSION PREVENTION

- Detects and Prevents Buffer Overflows
- Prevents unauthorized writes to system executables - preserves integrity of the OS
- Prevents attacks from invoking arbitrary commands on a system via a shell
- Prevents possibly compromised applications from damaging existing applications or downloading new ones
- Monitors and enforces which applications can run on the desktop
- Dynamically quarantines attachments identified as worm carriers
- Detects and Prevents Trojans
- Detects Distributed Port Scans
- Protects against SYN Floods
- Detects and Prevents Network Worms
- Allows Instant Messenger to run safely on corporate desktops[13]"

DISTRIBUTED FIREWALL FUNCTIONALITY

- Inbound port blocking
- Outbound port blocking
- Flexible policy definition which extends to any application
- Desktop and application protection against network and file-based attacks
- Ability to allow users to run applications but prevent undesired functionality within those applications
- "Locks Down" the Operating System, preventing unauthorized modification
- Zero Update architecture - Intrusion Prevention system does not use signatures, therefore reduces administration burden
- A browser-based management console that easily enables remote administration
- Real-time correlation and automated policy updates in response to enterprise wide events (distributed port scans, network and email worms)
- Prevention of 'never-before-seen' virus infections with global quarantining
- Prevention of Trojan horses and Buffer Overruns which traditionally by-pass personal firewalls and subvert remote clients via end user applications like web browsers
- Protects against outbound connections from Leaktest.exe, Firehole.exe, and tooleaky.exe
- Prevents keyboard sniffing
- Hides operating system identity from TCP stack fingerprinting programs like nmap or queso
- Blocks file download via Instant Messenger
- Prevents attacks from malicious mobile code such as ActiveX, signed Java, Javascript, and VBScript
- Provides sand boxing for Office, browsers, and Instant Messenger applications, protecting them from virus or web-based attack[14]"

Built-in capability is used to provide protection for a wide variety of classes of attack. Unauthorized access to system functions from code that executes in data or stack memory space is not allowed in order to prevent buffer overflow attacks. Port scans are monitored by agents and reported to the console so patterns indicative of a wide-spread attack are noted and this information is shared with other agents. Trojan horse applications exhibit certain activities that can be detected and blocked. These activities include trapping of system keystrokes; memory access and use of memory being used by another application; attempts to access system passwords; the immediate execution of downloaded executables; and, applications running as web servers. The software can detect and drop malformed packets that may be part of a "Ping of Death" attack.

All of these capabilities come as part of the standard package. Most businesses will need to employ these and also modify and add to them. Okena has a companion product, StormFront, which is designed to aid in this customization.

StormFront

StormFront is the policy creation and data analysis tool created to work with StormWatch. It uses a three part procedure to analyze an application and create a new policy to be implemented in StormWatch. Step 1 is to use the Management Console to configure the application to be monitored, when it is to be monitored, and which intelligent agent will do the monitoring and reporting. The application is then used in the manner it would normally be used and in an environment that has not been compromised. Step 2 is used to gather the data about the normal operation of the application over a period of time and then send the data to the analysis workstation. Finally, in step 3, the analysis workstation examines the logs and creates a policy to be used to compare normal the actions of the application with its actions in the future.

Processing Overhead

The additional processing caused by StormWatch has been estimated by Okena as being well below 5%. Testing by others has the added load as being negligible. It is generally believed that with today's fast processors, the CPU has a significant amount of down time while waiting for data access and that this down time will make up most of the time necessary for processing by the intrusion prevention software.

Comparison of Okena with IDS, Sandboxing and DFW Systems

The following chart, which is a modification of one published by Okena[15], shows some attributes of an attack and charts those aspects of the attack that each system is prepared to prevent or detect. IDS refers to an intrusion detection system, sandbox to a system that isolates an application for protection, and DWF refers to a distributed firewall.

Lifecycle of an Attack	Okena	IDS	Sandbox	DFW
Probe				
Ping addresses	Yes	Yes	X	Yes
Scan Ports	Yes	Yes	X	Yes
Guess Passwords	X	X	X	X
Guess Mail Users	X	X	X	X
Penetrate				
Mail Attachments	Yes	X	Yes	Yes
Buffer Overflows	Yes	Yes	X	Yes
ActiceX Controls	Yes	X	Yes	X
Network Installs	Yes	Yes	X	X
Compressed Messages	Yes	X	Yes	X
Backdoors	Yes	X	X	Yes
Persist				
Create New Files	Yes	Yes	X	X
Modify Existing Files	Yes	Yes	X	X
Weaken Registry Security Settings	Yes	X	X	X
Install New Services	Yes	Yes	X	Yes
Register Trap Doors	Yes	X	X	X
Propagate				
Mail Copy of Attack	Yes	X	X	X
Web Connection	Yes	X	X	X
IRC	Yes	X	X	Yes
FTP	Yes	X	X	Yes
Infect File Shares	Yes	X	X	X
Paralyze				
Delete Files	Yes	Yes	X	X
Modify Files	Yes	Yes	X	X
Drill Security Hole	Yes	X	X	X
Crash Computer	Yes	X	X	X
Denial of Service	Yes	X	X	X
Steal Secrets	Yes	X	X	X

Preventative Security Provided During an Attack

Reviews, Case Studies and other Hype

Okena has published a listing of the SANS Institute's Top Twenty Most Critical Internet Security Vulnerabilities and included information about how StormWatch is equipped to protect assets against them. The report can be found on Okena's site at http://www.okena.com/pdf/Stormwatch%20Sans%20Top20.pdf.

Okena also has published a case study showing how the software is able to protect against the Goner worm. It can be found at the URL http://www.okena.com/areas/solutions/solutions_attack.html.

System Requirements

To be able to effectively run StormWatch and StormFront, you should have a machine that is running Windows 2000 Server or Advanced Server with SP2 installed. The machine should have a minimum 400Mhz processor (dual processor support is available), 256MB RAM, 2 GB hard drive space and a single NIC (no multi-homed support is available). Intelligent agents can be installed on Windows 2000 or Windows NT systems. A new version of StormWatch for the Solaris platform is expected to be released in the first quarter of 2002.

Cost

Okena has a number of options for purchase of StormWatch. The base price for the Management Console is $3500. Intelligent agents for servers are priced at $1285 each and for desktops the price is $55 each. Pricing may be for perpetual service or for a fixed term of years.

Conclusions

Most experts agree that a layered approach to security is best. Such an approach implies protection at all levels of your network from perimeter protection to host-based protection. Many papers have been written that describe methods to protect your perimeter with firewalls, examine packets traveling over the network, detecting intrusions, using signature-based virus detection, etc to protect the basic assets that are contained on servers or workstations. All of these strategies are good and have their place in a well rounded security plan. However, all of these systems can be, and have been breached.

With the development of the intrusion prevention systems described in this paper, we now have another valuable tool in our toolbox that we can use to further protect our valuable assets. Since this tool works at the operating system level, it is able to stop malicious activity that has made it through all of our other defenses just before damage to the system is incurred. With the dramatic increase in the number of new attacks and their destructive power, this real-time protection will be seen by many as vital.

There are some problems. Since the technology is just over a year old, there is little broad-based experience with it. Reported problems with the software packages have been few but as more and more sites use them, more problems may arise. Additionally, if intrusion prevention software is effective in stopping attacks, intruders will work harder at finding ways to compromise them.

Intrusion prevention packages are not able to stop 100% of the attacks aimed at a protected system. Packages such as anti-virus software are still needed to eradicate malicious software once it has slipped past all defenses. It seems apparent that a natural approach to protection from attacks would include a combination of anti-virus and behavior-based detection systems into one package. Indeed, while many AV vendors have tried to minimize the potential of intrusion prevention software, others are working on just such a package.

Finally, no two of the packages I described above have exactly the same functions included. Some are aimed at protecting a wide base of machines while others are aimed strictly at protecting web servers. Care must be taken by a prospective user of these packages that they understand just what protection they are buying.

References

1 Merriam-Webster Collegiate Dictionary. URL http://www.m-w.com/dictionary.htm 2002.

2 Allen, Julia. The CERT Guide to System and Network Practices. Boston: Addison-Wesley 2001.

3 "Current Computer Virus Threats, Countermeasures and Strategic Solutions." 1996. URL http://download.nai.com/products/media/vil/pdf/wpb_6046.pdf (Feb. 17, 2002)

4 Conroy-Murray, Andrew. "Web Server Lockdown." Feb. 6, 2002. URL http://networkmagazine.com/article/NMG20020206S0013 (Feb. 17, 2002).

5 Kaspersky Labs Virus Calendar. URL http://www.viruslist.com/eng/viruscalendar.html (Feb. 24, 2002).

6 Entercept Security Technologies. "System Call Interception." URL http://www.entercept.com/products/entercept/whitepapers/downloads/systemcall.pdf (Feb. 17, 2002).

7 Harris Corporation. "STAT Neutralizer." URL http://www.statonline.harris.com/products/neutralizer/neutralizer.pdf (Feb. 17, 2002).

8 WatchGuard Technologies, Inc. "Online Training." URL 1st Page http://www.watchguard.com/training/server/splash.htm (Feb. 18, 2002).

9 Sanctum, Inc. "AppShield White Paper." Mar. 2001. URL http://www.sanctuminc.com/pdf/AppShield_31_WhitePaper_v2.pdf (Feb. 17, 2002)

10 eEye Digital Security. "SecureIIS Application Firewall." URL http://www.eeye.com/html/Products/SecureIIS/index.html (Feb. 18, 2002).

11 Okena Inc. "Technology Best Practices for Intrusion Prevention." URL http://www.okena.com/pdf/IP%20Best%20Practices.pdf (Feb. 22, 2002).

12 Okena, Inc. "Server Intrusion Prevention." URL http://www.okena.com/areas/solutions/solutions_server.html (Feb. 24, 2002).

13 Okena, Inc. “Server Intrusion Prevention.” URL http://www.okena.com/areas/solutions/solutions_generic.html (Feb. 24, 2002).

14 Okena, Inc. “Server Intrusion Prevention.” URL http://www.okena.com/areas/solutions/solutions_distributed.html (Feb. 24, 2002).

15 Okena, Inc. “How OKENA is Positioned Favorably vs. IDS; … Sandboxing Systems: … Distributed Firewalls.” URLs http://www.okena.com/pdf/Okena%20V%20IDS.pdf http://www.okena.com/pdf/Okena%20V%20Sandboxing.pdf http://www.okena.com/pdf/Okena%20V%20Dist%20Firewalls.pdf (Feb. 24, 2002)

Last Updated: May 29th, 2017

Upcoming Training

SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event