Global Information Assurance Certification Paper

Evaluation of Noc & Copernicus
By: Timothy W. Weber

Executive Summary

During the past few years the Internet has grown into an invaluable and almost mandatory tool for doing business. From electronic commerce to a global market to maintaining contact with existing customers through electronic mail, the internet has proven itself.

Unfortunately many users of the Internet try and disrupt this business for either fun or their own personal gain. Information security has become very critical for a company to succeed in this electronic environment.

Two systems on this network were evaluated and detailed results of this evaluation are contained within this document. The systems were checked with common tools that are easily downloaded from the Internet and have been used successfully by attackers for some time now. At no time was either of these systems compromised during the evaluation.

The recommendations contained in this document are:

Add firewall to the network: This needs to be done to protect the critical infrastructure servers from attack.

Place these servers on a protected network: This will allow controlled access to the servers and the services that they provide.

Improve backups: The current system of just mirroring the disks is inadequate and should be augmented by a tape backup system.

Increase use of ssh: SSH (secure shell) provides an encrypted method to remotely connect to servers allowing administrators to patch software and general systems administration.

Remove vendor sendmail and replace it with a newer version: While the Sun supplied version of sendmail is adequate for sending and receiving mail, it is not fully adequate in regards to security. New versions of sendmail are being released and should be employed.

Separate critical services onto separate servers: By placing dns and sendmail onto separate servers this will increase the chances that, should your systems be attacked, your organization will still be able to function during the outage.

Reconfigure mail system: The current configuration of just one mail server is not adequate to insure redundancy in case your systems are attacked or just suffer a critical

component failure.

Implement One time passwords: Provides a replacement mechanism for password authentication that is more superior to the re-usable password schemes most organizations are currently using.

Improve monitoring of both network and systems: Several software options have been presented that will improve the ability of the Systems Administrators to monitor the systems and network, to determine if there have been any intrusions.

Create a central logging server: Putting all systems logs onto a separate computer, inside the protected network, helps to insure the validity of those logs.

Create a more specific plan to implement a recovery system: By improving the method and practicing the techniques of data recovery will help your administrators put your servers back into production and protect any possible evidence contained on corrupted systems.

Run BIND from a chroot()'d environment: This technique of running software is invaluable to limiting the damage that a unauthorized person can do to a name server.

Increase Physical Security: The current controls placed on access to the production servers is good, but needs to be expanded by putting these two servers into a locked cabinet or another secured room.

NNTP server: By adding a Network Time Protocol to the network, this will help to insure that all logging is synchronized. This is invaluable if logs have to be retained as evidence.

If these recommendations are accepted, it will take considerable time and effort on the part of the administrators, not to mention financial resources. Wherever possible free or low cost solutions have been provided for your consideration.

Overview

While the overall configuration of the two UNIX servers evaluated is fundamentally sound and reasonably secure, a lot of room exists for improvement, as will be explained throughout this document.

In Appendix A, a diagram of the proposals presented in this document are detailed.

Methodology

The two systems examined were looked at by using well known and commonly available tools to look at the system from the "outside", or the Internet. This is the point where most intruders will scan your network. After that was complete, the operating systems on these two computers was evaluated by examining their configuration files and the software installed.

No attempt was made to try and compromise these systems.

Assessment of Network

By using certain tools to look at the network from the Internet the following was discovered.

The server noc.xxx.xxx was scanned using nmap and the following was discovered.

Interesting ports on noc.xxx.xxx (192.xxx.x.x):
(The 1521 ports scanned but not shown below are in state: closed)

Port	State	Service
22/tcp	open	ssh
53/tcp	open	domain

TCP Sequence Prediction: Class=random positive increments
Difficulty=20751 (Worthy challenge)
Remote OS guesses: Solaris 2.6 - 2.7, Solaris 7

After reviewing the configuration files of the ssh daemon it was found that this compilation of ssh was compiled without tcp-wrapper support, but the /etc/sshd_config file was properly setup to limit access to only certain hosts.

The second UNIX server on the network to be looked at is copernicus.xxx.xxx. It was also scanned using nmap and the following was found.

Interesting ports on copernicus.xxx.xxx (192.xxx.x.x):
(The 1519 ports scanned but not shown below are in state: closed)

Port	State	Service
21/tcp	open	ftp
22/tcp	open	ssh

25/tcp open smtp
53/tcp open domain

TCP Sequence Prediction: Class=random positive increments
Difficulty=33735 (Worthy challenge)
Remote OS guesses: Solaris 2.6 - 2.7, Solaris 7

This version of ssh was compiled with tcp-wrapper support and the /etc/hosts.allow file is properly configured to allow access from a limited number of hosts, and the host.deny is setup to e-mail the administrator when there is an attempted unauthorized access.

A telnet session was established to port 25 on copernicus.xxx.xxx and it clearly shows that this system is using the vendor supplied version of sendmail.

220 copernicus.xxx.xxx ESMTP Sendmail 8.9.3+Sun/8.9.1; Mon, 31 Jul 2000

This will be addressed later in this document.

Positive Aspect of the UNIX Servers.

When the time came to inspect the UNIX OS and installed software, the Administrators would not give the evaluators accounts, but instead provided us whatever information we requested. This is a good security policy.

By operating your system as a 100% switched network you limit the ability of an intruder to sniff passwords off of the network.

The bastion host configuration is fundamentally sound, but not protected by a firewall.

The use of ssh is a marked improvement over telnet or rsh. The configurations on these systems, while different, was effective at limiting access.

Both systems evaluated are using tcp wrappers to control remote access to these machines.

The logcheck utility has been deployed to check the log files on the systems on an hourly basis and the results sent to several administrators.

Logging has been increased to include an authentication log (auth log) and su log.

Relaying is not allowed on your sendmail server. By doing this you prevent "spammers" from using your sendmail server as a relay to send out junk e-mail to the rest of the Internet community.

The version of BIND you are using has been hidden from most outside users through the proper configuration of the /etc/named.conf file.

No compilers were found on any system evaluated. This is important from the stand point of, if someone is able to access your system, they will not be able to compile a root kit on the system, but will have to do it somewhere else and then install it on the system.

Recommendations

1. Add firewall to network

A properly configured firewall will protect the entire network from the prying eyes of unauthorized people.

Currently, copernicus and noc are sitting exposed on to the Internet with no firewall protection whatsoever. While the systems have been upgraded to Solaris 7 and have been configured in the bastion host configuration, it is just a matter of time before someone is able to illegally gain access to these systems and use them for their own purposes.

A firewall will also screen your internal computers from being scanned by outsiders, and help prevent their being used by other, unauthorized parties. A properly configured firewall can also alert the network administrators to any unusual behavior, which may or may not be a sign of an attack or intrusion.

Several options are available for a firewall on this network.

Purchase a commercial application that will shield your network from unauthorized access.

Firewall-1 is a packet filtering software that can be deployed on both UNIX and NT systems. (Ref. 1)

Axent's Raptor is a proxy firewall that is also available for both UNIX and NT platforms. (Ref. 2)

Most modern versions of the Linux OS comes pre-packaged with firewall support and software. This is a cost effective, quick, and easy solution to provide your network with some protection, especially when combined with the bastille Linux configuration. The OS and firewall can be deployed on an existing Intel based computer.

Options available for Linux based firewalls.

SOCKS: This is a proxy firewall package for the Linux OS. A proxy server acts as an intermediary system between your users at their workstations and the Internet. In

effect, the only IP that should be seen on the outside is the IP of the firewall. The firewall will make the connection to the requested outside service and then relay the information to the user making the request. (Ref. 3)

IPChains: IP Chains comes standard with most versions of Linux. IP Chains allows for outbound connections to the Internet, while it blocks any unsolicited inbound connections to any of your internal workstations. IP Chains will also provide IP masquerading services for your network. If your internal workstations do not have routable IP addressees or you just do not want the world to know which computers are connecting to the Internet, IP chains will mask your IP addresses and only the IP of the Linux firewall will be seen on the outside. (Ref. 4)

2. Place these servers on a protected network.

Due to the functions that these two servers perform it is necessary that they have to accept connections from the Internet. This opens them up for attack by people both outside and inside your network. The network should be configured so that all connections to these systems are routed through the firewall and only on the ports allocated for that service. Some firewall software allows you to give these systems on the protected network with non-routeable IP address, with the firewall accepting the connections and then re-routing the request to these machines. The firewall can also be used to protect your internal servers from being attacked if one of the servers on your protected network is compromised.

This will decrease the possibility of your own servers being used against you by an attacker or against another network in a Denial of Service attack. It can also decrease the possibility of someone gaining access to the system and installing packet sniffing software in the attempt to gather user names and reusable passwords.

3. Improve Backups.

The backups that the organization are conducting is currently insufficient for most organizations. While disk mirroring is a good first step in recovering from a failure of the primary disk, it is insufficient for recovering data that has been lost due to a system compromise. The ability to recover the corrupted data, determine when the system was accessed and regain control of the system are vital to not only protecting your company data, but can also be used to prosecute the parties involved in the compromise of the system.

It is recommended that a external tape drive be purchased and deployed on a protected system. This system, with tape drive can be used to back up the UNIX systems deployed on your network through the use of ssh.

The backup tapes that are made should be stored off-site in a protected location, and the backups occasionally tested.

4. Increase use of ssh.

SSH (Secure Shell) is currently being used on these systems as a replacement for telnet and rsh. It is recommended that ssh also be used to copy software updates and paths to these systems through the scp binary that is also provided with the ssh software, replacing ftp services.

The ssh software can also be used to provide encrypted tunnels to a separate logging server.

5. Remove vendor sendmail and replace it with a newer version.

As was demonstrated during the network assessment, currently copernicus is running the Sun supplied version of sendmail. The configuration of your sendmail systems tells a potential attacker exactly which version of sendmail you are using. Replace this version of sendmail with one of the more recent releases, i.e. 8.10.1 or higher. (Ref 5)

The documentation that comes with the sendmail package, as provided by sendmail.org is quite adequate to provide the administrators with the information that is needed to insure that the daemon is quite secure. Much thought has been put into security vulnerabilities and code is provided in the package.

This software , or any software, should not be compiled on these servers. Create a separate workstation, on a protected network segment, and install the compilers on this machine. Compile the necessary software and then manually install it or create a script that will install the software. By eliminating compilers on servers, it is just a little more difficult for an intruder to compile and install software on your systems.

While looking at the sendmail.cf file it was noted that the shell sendmail uses is the bourne shell. This should be replaced with smrsh. This sendmail restricted shell will still allow the process to perform its function, but provides very limited services to an attacker if they are able to compromise the sendmail daemon and get a shell.

6.. Separate dns and sendmail onto two different servers.

Copernicus is currently serving as both a dns server and is acting as your sendmail server. It is suggested that these two services be separated onto different machines. This will not only limit the servers exposure to unauthorized access, by running only critical services, but it should improve performance of both your e-mail system and your dns records.

Both sendmail and BIND will run on Linux based servers running on Intel based computers. A version of Solaris is also available for Intel based, inexpensive workstations.

7. Add redundancy to mail servers.

Only one mail server is servicing your entire organization. By adding a second mail gateway into your network, mail will not stop if one system is compromised. In addition to this these "mailgates" should only be allowed to connect to a central mailhub that is located inside of your network. The mail would then be routed to another system that the users can access to get their mail.

The mailgates can be prevented from connections to any other systems through the proper configuration of the firewall and compiling sendmail on the mailhub to use tcp wrapper support and an entry placed in the /etc/hosts.allow file for smtp connections.

8. Implement One time passwords.

With the limited number of administrators that your organization has these systems are good candidates for one time password systems. The use of these passwords will help prevent a unauthorized person from being able to use a password cracking program if they were to access your password file.

Several options are available to your organization,

Obtain a commercial product such as SecurID. While SecurID is not the only available solution it is the one we are most familiar with. This solution requires a token that is given to each user and a separate stand alone server to authenticate the token code.(Ref 6)

Obtain and install a open source one time password scheme. The most commonly used open source One time password software is OPIE (One Time Passwords In Everything). (Ref 7)

9. Improve monitoring of both network and systems.

Noc and copernicus are currently only running logcheck and increased logging for system activities. Other tools are available to assist in monitoring the systems to insure their integrity. Many free tools are available to assist your administrators in determining if there are any vulnerabilities in these systems and to monitor these systems to insure their integrity. Some of the tools that can be placed on these systems are:

Tiger: This software package will check different portions of the software, looking for known vulnerabilities and advise your administrators of these vulnerabilities. It also has the ability to check a system, after it has been put into production for symptoms of a break in. (Ref. 8)

lsof: This is a good package for monitoring a system that has been placed in production. It will provide a list of processes and the files that those processes are using. It can provide information on what users are doing and who is connecting to your servers

and what Internet sockets are open. (Ref. 9)

nmap: This is a freely available tool that will scan the network and report what it finds. This is one of the tools that a person looking for a way into your system may use to gather needed intelligence on your network. Careful use of this tool will not only allow your Administrators the ability to detect and close possible security problems, but with frequent use will allow the administrators the ability to detect new services that have shown up on a system or network, that may indicate a possible intrusion. (Ref. 10)

***NOTE:** Prior to running this software on a network or against a particular system it is advisable that permission be obtained from someone in authority. It is recommended that a policy be put in place to conduct this type of testing. This will help prevent someone from taking this test as an active attack.

10. Create a central logging server.

Currently these two systems act as their own loghost. It is always a good practice to configure systems to log to another, independent host. If the logs for these two systems are on another computer it makes it difficult for an attacker to alter these logs, as many of them do, to help cover their tracks.

An inexpensive Linux system can be used as a loghost. This should be a secure system that is protected by a firewall. This can be run on an existing x86 platform. The web site **www.bastille-linux.org** has more information on securing the Linux operating system.

11. Create a more specific plan to implement a recovery system.

The current plan to recover your systems in case of an unauthorized intrusion or even a catastrophic hardware failure is to re-install the OS and software that these systems run and get them back into production quickly. While this is acceptable for getting your systems back on-line as quickly as possible, it does not lend itself to providing the administrators with the information that they need to determine why the systems filed, was it a hardware failure or "were we hacked."

One plan that can be implemented is to maintain several "master" copies of the systems. A good plan is to, as much as possible, insure that the configurations on the production systems are as identical as they can be made, this is where the Bastion host configuration on the Solaris systems is handy. If possible keep this master on a second hard drive that can be quickly swapped, and the corrupted hard drive removed.

The data, dns or sendmail configurations, can be recovered from a master backup tape that should be made before the deployment of any system. By doing this your administrators can not only put the server back into production as quickly as possible, but they also have maintained the data on the corrupted system.

12. Run BIND from a chroot()'d environment.

The dns servers that are on this network are currently running BIND from root (/). BIND provides a mechanism where it can be run from a chroot'd environment, run command or interactive shell with special root directory. If an intruder is able to get a shell by compromising the named daemon, the amount of damage that can be done should be confined to only that special root directory.

The BIND documentation describes the process for configuring this chroot'd environment.

13. Increase Physical Security.

The systems evaluated are currently in an access controlled computer room, but they are sitting in the open on a desk top. It was noted that there is quite a bit of air conditioner related work going on in this facility. It is recommended that these systems be moved to a locked computer cabinet or another locked room and limit access to only authorized administrators.

13. NTP (Network Time Protocol) server.

Having a time server on your network insures that all of the systems on that network have synchronized their internal time clocks so that the logging is consistent across all the systems. By doing this it improves the chances of having usable data in your logs to assist systems administrators and law enforcement agencies in determining the sequence of events as to how a system was compromised and the events that led up to that compromise. It will also help law enforcement agencies in gathering evidence that can be used in court to prosecute someone who attacks your systems. (Ref. 11)

Conclusions

While the configuration on the systems evaluated is good, a lot of room for improvement exists. While there is no one "magic bullet" that can be placed on your systems to insure there security and the integrity of the information on your systems, there are steps that can be taken to help the administrators make it difficult for unauthorized person to gain access and to detect the presence of those unauthorized persons.

Administrators need to be security conscious at all times, even while doing such mundane things as patching a system, turning on a new service or installing new software. Most software that can be downloaded from the Internet and the software described in this document comes with a MD5 checksum. This tool will provide the administrators the ability to check that they are getting a valid copy of the software.

The changes that have been recommended will consume valuable resources, time for administrators to plan and configure and the purchase of new software and hardware,

some room exists to reduce the burden on your organization. Some examples are:

Use Linux where ever possible on inexpensive Intel based workstations. The only real cost in this type of configuration is the purchase of the needed hardware. Additionally, a lot of freeware is available to Linux users from many different sites on the Internet.

A version of Solaris (7 & 8) is available for Intel based computers. The software is freely distributed from Sun with the only expense being the cost of the media that it is distributed on. Currently the cost of this media is about $79.00. (Ref. 12)

References:

1. Firewall 1:
Web site: http://www.checkpoint.com

2. Raptor Firewall:
Web site: http://www.axent.com

3. SOCKS can be obtained from:
http://www.socks.nec.com/

4. Linux Firewalls:
RedHat Linux Bible.
Christopher Negus
Pages 467-473

5. Sendmail can be downloaded from the sendmail website. The URL is:
http://www.sendmail.org

6. Information about SecurID can be obtained from:
http://www.rsasecurity.com

7. The OPIE software can be downloaded from:
http://www.inner.net/opie

8. Tiger: This software package is freely available at:
ftp://net.tamu.edu/ftp/security/TAMU/tiger-2.2.4p1.tar.gz

9. lsof: This tool can be downloaded from:
ftp://vci.cc.purdue.edu/pub/tools/unix/lsof

10. nmap: This software can be freely downloaded from:
http://www.insecure.org/nmap/dist/nmap-2.53.tgz

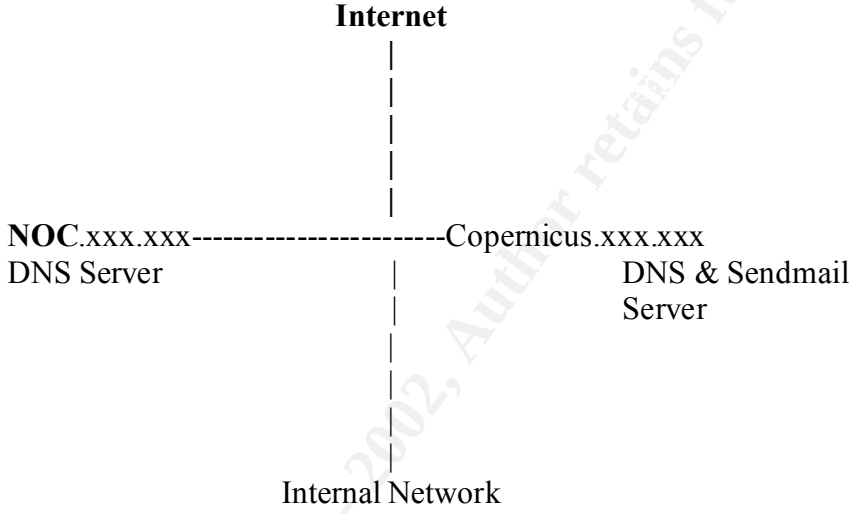
11. A list of time servers on the Internet can be downloaded from:
http://www.ntp.org

12. http://www.sun.com

Appendix A.

Diagram of Existing Network
and
Recommended Changes

Current Network



Network design if all proposed changes implemented.