The 60 Minute Network Security Guide

(First Steps Towards a Secure Network Environment)

Systems and Network Attack Center (SNAC)

October 16, 2001
Version 1.0

National Security Agency
9800 Savage Rd. Suite 6704
Ft. Meade, MD 20755-6704

SNAC.Guides@nsa.gov

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (FROM - TO)
16-10-2001		xx-xx-2001 to xx-xx-2001

4. TITLE AND SUBTITLE
The 60 Minute Network Security Guide (First Steps Towards a Secure Network Environment)
Unclassified

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME AND ADDRESS
Booz Allen & Hamilton
8283 Greensboro Drive
McLean, VA22102

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME AND ADDRESS
National Security Agency
9800 Savage Road, Suite 6704
Ft. Meade, MD20755-6704

10. SPONSOR/MONITOR'S ACRONYM(S)
11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
APUBLIC RELEASE
,

13. SUPPLEMENTARY NOTES

14. ABSTRACT
During the last four years the National Security Agency?s Systems and Network Attack Center (C4) has released Security Guides for operating systems, applications and systems that operate in the larger IT network. These security guides can be found at our web site www.nsa.gov / Security Recommendation Guides. Many organizations across the Department of Defense have used these documents to develop new networks and to secure existing IT infrastructures. This latest Security Guide addresses security a bit differently. Our goal is to make system owners and operators aware of fixes that become force multipliers in the effort to secure their IT network.

15. SUBJECT TERMS
IATAC Collection; information security; network security

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE	Public Release	36	Fenster, Lynn lfenster@dtic.mil
Unclassified	Unclassified	Unclassified			19b. TELEPHONE NUMBER International Area Code Area Code Telephone Number 703767-9007 DSN 427-9007

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	10/16/2001	Report 10/16/2001

4. TITLE AND SUBTITLE
The 60 Minute Network Security Guide (First Steps Towards a Secure Network Environment)

5. FUNDING NUMBERS

6. AUTHOR(S)
Systems and Network Attack Center (SNAC)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Booz Allen & Hamilton
8283 Greensboro Drive
McLean, VA 22102

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Security Agency
9800 Savage Road Suite 6704 Ft Meade, MD
20755-6704

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; Distribution is unlimited

12b. DISTRIBUTION CODE
A

13. ABSTRACT *(Maximum 200 Words)*

During the last four years the National Security Agency’s Systems and Network Attack Center (C4) has released Security Guides for operating systems, applications and systems that operate in the larger IT network. These security guides can be found at our web site www.nsa.gov / Security Recommendation Guides. Many organizations across the Department of Defense have used these documents to develop new networks and to secure existing IT infrastructures. This latest Security Guide addresses security a bit differently. Our goal is to make system owners and operators aware of fixes that become force multipliers in the effort to secure their IT network.

14. SUBJECT TERMS
IATAC Collection, information security, network security

15. NUMBER OF PAGES
35

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

Table of Contents

Introduction

During the last four years the National Security Agency's Systems and Network Attack Center (C4) has released Security Guides for operating systems, applications and systems that operate in the larger IT network. These security guides can be found at our web site www.nsa.gov / Security Recommendation Guides. Many organizations across the Department of Defense have used these documents to develop new networks and to secure existing IT infrastructures. This latest Security Guide addresses security a bit differently. Our goal is to make system owners and operators aware of fixes that become "force multipliers" in the effort to secure their IT network.

Security of the IT infrastructure is a complicated subject, usually addressed by experienced security professionals. However, as more and more commands become ``wired'', an increasing number of people need to understand the fundamentals of security in a networked world. This Security Guide was written with the less experienced System Administrator and information systems manager in mind, to help them understand and deal with the risks they face.

Opportunistic attackers routinely exploit the security vulnerabilities addressed in this document, because they are easily identified and rarely fixed. ISSMs, ISSOs and System Administrators provide a level of risk management against the multitude of vulnerabilities present across the IT infrastructure. The task is daunting when considering all of their responsibilities. Security scanners can help administrator identify thousands of vulnerabilities, but their output can quickly overwhelm the IT team's ability to effectively use the information to protect the network. This Security Guide was written to help with that problem by focusing the experience our research and operational understanding of the DoD and other US Government IT infrastructures.

This Security Guide should not be misconstrued as anything other than security "best practices" from the National Security Agency's Systems and Network Attack Center (C4). We hope that the reader will gain a wider perspective on security in general, and better understand how to reduce and manage network security risk.

We welcome your comments and feedback. SNAC.Guides@nsa.gov

General Guidance

The following section discusses general security advice that can be applied to any network.

Security Policy

(This section is an abstract of the security policy section of RFC 2196, Site Security Handbook. Refer to this RFC for further details.)

A security policy is a formal statement of the rules that people who are given access to an organization's technology and information assets must abide. The policy communicates the security goals to all of the users, the administrators, and the managers. The goals will be largely determined by the following key tradeoffs: services offered versus security provided, ease of use versus security, and cost of security versus risk of loss.

The main purpose of a security policy is to inform the users, the administrators and the managers of their obligatory requirements for protecting technology and information assets. The policy should specify the mechanisms through which these requirements can be met. Another purpose is to provide a baseline from which to acquire, configure and audit computer systems and networks for compliance with the policy. In order for a security policy to be appropriate and effective, it needs to have the acceptance and support of all levels of employees within the organization.

A good security policy must:

- Be able to be implemented through system administration procedures, publishing of acceptable use guidelines, or other appropriate methods
- Be able to be enforced with security tools, where appropriate, and with sanctions, where actual prevention is not technically feasible
- Clearly define the areas of responsibility for the users, the administrators, and the managers
- Be communicated to all once it is established
- Be flexible to the changing environment of a computer network since it is a living document

Operating Systems and Applications: Versions and Updates

As much as possible, use the latest available and stable versions of the operating systems and the applications on all of the following computers on the network: clients, servers, switches, routers, firewalls and intrusion detection systems. Keep the operating systems and the applications current by installing the latest updates (e.g., patches, service packs, hotfixes), especially updates that correct vulnerabilities that could allow an attacker to execute code. Note that some updates may not be applied to the computer until a reboot occurs. The following applications should be given particular attention because they have been frequently targeted (e.g., by CodeRed, Melissa virus, Nimda): IIS, Outlook, Internet Explorer, BIND and Sendmail.

Know Your Network

Developing and maintaining a list of all hardware devices and installed software is important to the security of the IT infrastructure. Understanding software applications that are installed by default is also important (e.g., IIS is installed by default by SMS and SQL Server on Windows platforms). A quick method for taking inventory of services running on the network is to port scan.

TCP/UDP Servers and Services on the Network

Scan the network for all active TCP/UDP servers and services on each computer in the network. Shut down unnecessary servers and services. For those servers that are necessary, restrict access to only those computers that need it. Turning off functional areas, which are seldom used but have vulnerabilities, prevents an attacker from being able to take advantage of them. Other applications install with sample CGI scripts, which sometimes contain problems. As a general rule do not install sample applications in production systems.

Passwords

Poor password selection is frequently a major problem for any system's security. Users should be forced to change their passwords regularly. Set up password aging via Account Policy for Windows systems or the `/etc/default/passwd` file in UNIX. Administrators should obtain and run password-guessing programs (i.e., *"John the Ripper," "L0phtCrack,"* and *"Crack"*) frequently to identify those users having easily guessed passwords. Because password cracking programs are very CPU intensive and can slow down the system on which it is running, it is a good idea to transfer the encrypted passwords (the dumped SAM database for Windows and the `/etc/passwd` and `/etc/shadow` files in UNIX) to a stand-alone (not networked) system. Also, by doing the work on a non-networked machine, any results found will not be accessible by anyone unless they have physical access to that system.

Passwords should:

- Be 12 or more characters in length on Windows systems, 8 characters in length on UNIX
- Include upper and lower case letters, numbers, and special characters
- Not consist of dictionary words
- Be changed regularly (every 30 to 90 days)
- For UNIX, be encrypted and stored in the /etc/shadow file (for some UNIX systems) with permissions set to 400 with ownership by root and group sys. The /etc/passwd file should have permissions 644 with owner root and group root.
- Be cracked every month to find users choosing easily guessed or cracked passwords

For UNIX, lock the following accounts by placing a *LK* in encrypted password field in /etc/shadow: adm, bin, daemon, listen, lp, nobody, noaccess, nuucp, smtp, sys, uucp. These accounts should not have login shells, rather they should be set to /dev/null.

Do Not Run Code From Non-Trusted Sources

For the most part, software applications run in the security context of the person executing them without any consideration to source. A PKI infrastructure may help, but when not available remember that spoofing the "From" line of an e-mail message and disguising URLs are trivial. DO NOT OPEN E-MAIL ATTACHMENTS OR RUN PROGRAMS UNLESS THE SOURCE AND INTENT ARE CONFIRMED AND TRUSTED. Always run Outlook so that it executes in the restricted zone and disable all scripting and active content for that zone. For more specific details, reference *"E-mail Client Security in the Wake of Recent Malicious Code Incidents"* available at http://www.nsa.gov.

Block Certain E-Mail Attachment Types

There are numerous kinds of executable file attachments that many organizations do not need to routinely distribute via e-mail. If possible, block these at the perimeter as a countermeasure against the malicious code threat. Organizations using Outlook can also block them using Outlook 2002 or, for earlier versions of Outlook, using the appropriate security patches.

The specific file types that can be blocked are:

.bas	.hta	.msp	.url
.bat	.inf	.mst	.vb
.chm	.ins	.pif	.vbe
.cmd	.isp	.pl	.vbs
.com	.js	.reg	.ws
.cpl	.jse	.scr	.wsc
.crt	.lnk	.sct	.wsf
.exe	.msi	.shs	.wsh

It may be prudent to add, or delete files from this list depending upon operational realities. For example, it may be practical to block applications within the Microsoft Office family, all of which can contain an executable component. Most notable are Microsoft Access files, which unlike other members of the Office family have no intrinsic protection against malicious macros.

Follow The Concept Of Least Privilege

Least privilege is a basic tenet of computer security that means users should be given only those rights required to do their job. Malicious code runs in the security context of the user launching the code. The more privileges the user has, the more damage the code can do. Recommendations pertaining to the least privilege principle include:

- Keep the number of administrative accounts to a minimum
- Administrators should use a regular account as much as possible instead of logging in as administrator or root to perform routine activities such as reading mail
- Set resource permissions properly. Tighten the permissions on tools that an attacker might use once he has gained a foothold on the system, e.g., `explorer.exe, regedit.exe, poledit.exe, taskman.exe, at.exe, cacls.exe, cmd.exe, finger.exe, ftp.exe, nbstat.exe, net.exe, net1.exe, netsh.exe, rcp.exe, regedt32.exe, regini.exe, regsvr32.exe, rexec.exe, rsh.exe, runas.exe, runonce.exe, svrmgr.exe, sysedit.exe, telnet.exe, tftp.exe, tracert.exe, usrmgr.exe,`

`wscript.exe`, and `xcopy.exe`. Unix tools or utilities that should be restricted are debuggers, compilers, and scripting languages such as `gcc, perl`, etc.

- The least privilege concept also applies to server applications. Where possible, run services and applications under a non-privileged account.

Application Auditing

Most server-level applications have extensive auditing capabilities. Auditing can be of value in tracking down suspected or actual intrusions. Enable auditing for server applications and audit access to key files (such as those listed above) that an attacker might use once he has gained a foothold on a compromised server.

Network Printer

Today's network printers contain built-in FTP, WEB, and Telnet services as part of their OS. Enabled network printers can be readily exploited and are often overlooked by system administrators as a security threat. These network printers can and are often exploited as FTP bound servers, Telnet jump-off platforms, or exploited by web management services. Change the default password to a complex password. Explicitly block the printer ports at the boundary router/firewall and disable these services if not needed.

Simple Network Management Protocol (SNMP)

SNMP is widely used by network administrators to monitor and administer all types of computers (e.g., routers, switches, printers). SNMP uses an unencrypted "community string" as its only authentication mechanism. Attackers can use this vulnerability in SNMP to possibly gather information from, reconfigure or shut down a computer remotely. If an attack can collect SNMP traffic on a network, then he can learn a great deal about the structure of the network as well as the systems and devices attached to it.

Disable all SNMP servers on any computer where it is not necessary. However, if SNMP is a requirement, then consider the following. Allow read-only access and not read-write access via SNMP. Do not use standard community strings (e.g., public, private). If possible, only allow a small set of computers access to the SNMP server on the computer.

Network Security Testing

Test regularly the security of all of the following computers on the network: clients, servers, switches, routers, firewalls and intrusion detection systems. Also, do this after any major configuration changes on the network.

Perimeter Routers and Firewalls

The following section addresses recommendations for securing network perimeter routers and firewalls.

Host Security

Recommendations for improved host security include:

- Shut down unneeded TCP/UDP servers (e.g., bootps, finger) on the router or the firewall. Servers that are not running cannot break. Also, more memory and processor slots are available with less servers running.
- For TCP/UDP servers on the router or the firewall that are necessary, make sure that access to them is limited only to the administrators.
- Shut down unneeded services (e.g., source routing, remote configuration) on the router or the firewall.
- Disable any unused interface on the router or the firewall. Protect each and every active interface on the router or the firewall from information gathering and attacks.
- Protect each and every management port on the router or the firewall from attacks. Disable any unused management port.
- Configure durable passwords on the router or the firewall. For each password use the following guidelines: be at least eight characters long, not be words, not begin with a number, and include at least one character from the sets of letters, numbers and all other characters (e.g., ,./<>;':"[]\{}|~!@#$%^&*()_+`-=). Consider using different passwords for each router and each firewall. Change passwords at least once every 90 days.

Example: Cisco IOS Routers

The following scenario steps through the recommendations listed above.

- The `show processes` command can help to show active information about the servers on the router. The following commands show how to disable the following servers: TCP/UDP small servers (echo, discard, daytime, chargen), bootps, finger, http, identd and snmp.

```
Router(config)# no service tcp-small-servers
Router(config)# no service udp-small-servers
Router(config)# no ip bootp server
Router(config)# no service finger
Router(config)# no ip http server
Router(config)# no ip identd
Router(config)# no snmp-server community <community string>
```

- If SNMP on the router is required, use the following commands to clear out any SNMP servers with default community strings.

```
Router(config)# no snmp-server community public
```

```
Router(config)# no snmp-server community private
```

- Then set up the SNMP server with a community string that is difficult to guess. Also, if possible, allow only read-only access to the server; do not allow read-write access to the server. Apply an access-list to the server. Refer to the following section on TCP/IP Filters for discussion of an access-list for SNMP in more detail. The following command is an example.

```
Router(config)# snmp-server community S3cr3t-str1n9 ro 10
```

- The following commands disable the following services: Cisco Discovery Protocol (CDP), remote configuration downloading, source routing and zero subnet.

```
Router(config)# no cdp run
Router(config)# no service config
Router(config)# no ip source-route
Router(config)# no ip subnet-zero
```

- The following command disables a router interface.

```
Router(config-if)# shutdown
```

Secure each and every active interface on the router from Smurf attacks, ad-hoc routing and access-list queries with the following commands.

```
Router(config-if)# no ip directed-broadcast
Router(config-if)# no ip proxy-arp
Router(config-if)# no ip unreachables
```

- Configure the console line () and the virtual terminal lines () on the router to time out a session, to require a password at login and to allow only telnet traffic. If the auxiliary line () is not needed, then it should be disabled. Use the following line configuration commands to configure the lines.

```
Router(config)# line con 0
Router(config-line)# exec-timeout 5 0
Router(config-line)# login
Router(config-line)# transport input telnet
Router(config)# line aux 0
Router(config-line)# no exec
Router(config-line)# exec-timeout 0 5
Router(config-line)# no login
Router(config-line)# transport input none
Router(config)# line vty 0 4
```

```
Router(config-line)# exec-timeout 5 0
Router(config-line)# login
Router(config-line)# transport input telnet
```

- Configure the Enable Secret password, which is protected with an MD5-based algorithm. The following global configuration command is an example.

  ```
  Router(config)# enable secret 0 2manyRt3s
  ```

  Configure passwords for the console line, the auxiliary line and the virtual terminal lines. Use a different password for the console line and the auxiliary line versus the virtual terminal lines. The following line configuration commands are examples.

  ```
  Router(config)# line con 0
  Router(config-line)# password Soda-4-jimmY
  Router(config)# line aux 0
  Router(config-line)# password Popcorn-4-sara
  Router(config)# line vty 0 4
  Router(config-line)# password Dots-4-georg3
  ```

  Provide a basic protection for the line passwords by using the following global configuration command.

  ```
  Router(config)# service password-encryption
  ```

TCP/IP Filters

Carefully consider which TCP/IP services will be allowed through and to the perimeter routers and firewalls (inbound and outbound). Use the following guidelines for creating filters: those services that are not explicitly permitted are prohibited. The following tables present common services to restrict because they can be used to gather information about the protected network or they have weaknesses that can be exploited against the protected network.

- **Table 1** lists those TCP or UDP servers that should be completely blocked at the perimeter router or firewall. These services should not be allowed across the router or the firewall in either direction. Also, they should not be allowed to the router or the firewall.

- **Table 2** lists those TCP or UDP servers on the protected network, on the router or on the firewall that should not be accessible by external clients.

- **Table 3** lists the common TCP or UDP servers on the protected network, on the router or on the firewall that may need some access by internal or external clients and servers. Many of these services can be filtered to the few authorized computers (e.g., ftp server, mail server, domain name server, web server) on the protected network or on the DMZ subnet.

- **Table 4** lists the ICMP message types that can be allowed outbound from the protected network, while all other message types should be blocked.

- **Table 5** lists the ICMP message types that can be allowed inbound to the protected network, while all other message types should be blocked.

In general, the administrator should create filters focusing on what services and hosts are permitted and denying everything else. This method means that one may not need to block each service in the tables below with a specific filter statement. Finally, use an intrusion detection system on the protected network to monitor the TCP/IP traffic that is allowed past the perimeter routers and firewalls.

Table 1:
TCP or UDP Servers to Completely Block at the Perimeter Router/Firewall

Port(s) (Transport)	Server	Port(s) (Transport)	Server
1 (TCP & UDP)	tcpmux	1981 (TCP)	Shockrave
7 (TCP & UDP)	echo	1999 (TCP)	BackDoor
9 (TCP & UDP)	discard	2001 (TCP)	Trojan Cow
11 (TCP & UDP)	systat	2023 (TCP)	Ripper
13 (TCP & UDP)	daytime	2049 (TCP & UDP)	nfs
15 (TCP & UDP)	netstat	2115 (TCP)	Bugs
17 (TCP & UDP)	qotd	2140 (TCP)	Deep Throat
19 (TCP & UDP)	chargen	2222 (TCP)	Subseven21
37 (TCP & UDP)	time	2301 (TCP & UDP)	compaqdiag
43 (TCP & UDP)	whois	2565 (TCP)	Striker
67 (TCP & UDP)	bootps	2583 (TCP)	WinCrash
68 (TCP & UDP)	bootpc	2701 (TCP & UDP)	sms-rcinfo
69 (UDP)	tftp	2702 (TCP & UDP)	sms-remctrl
93 (TCP)	supdup	2703 (TCP & UDP)	sms-chat
111 (TCP & UDP)	sunrpc	2704 (TCP & UDP)	sms-xfer
135 (TCP & UDP)	loc-srv	2801 (TCP)	Phineas P.
137 (TCP & UDP)	netbios-ns	4045 (UDP)	lockd
138 (TCP & UDP)	netbios-dgm	5800 - 5899 (TCP)	winvnc web server
139 (TCP & UDP)	netbios-ssn	5900 – 5999 (TCP)	winvnc
177 (TCP & UDP)	xdmcp	6000 – 6063 (TCP)	X11 Window System
445 (TCP & UDP)	microsoft-ds	6665 - 6669 (TCP)	irc
512 (TCP)	rexec	6711 - 6712 (TCP)	Subseven
513 (TCP)	rlogin	6776 (TCP)	Subseven
513 (UDP)	who	7000 (TCP)	Subseven21
514 (TCP)	rsh, rcp, rdist, rdump, rrestore	12345 – 12346 (TCP)	NetBus
515 (TCP)	lpr	16660 (TCP)	Stacheldraht
517 (UDP)	talk	27444 (UDP)	Trinoo
518 (UDP)	ntalk	27665 (TCP)	Trinoo
540 (TCP)	uucp	31335 (UDP)	Trinoo
1024 (TCP)	NetSpy	31337 – 31338 (TCP & UDP)	Back Orifice
1045 (TCP)	Rasmin	32700 – 32900 (TCP & UDP)	RPC services
1090 (TCP)	Xtreme	33270 (TCP)	Trinity V3
1170 (TCP)	Psyber S.S.	39168 (TCP)	Trinity V3
1234 (TCP)	Ultors Trojan	65000 (TCP)	Stacheldraht
1243 (TCP)	Backdoor-G		
1245 (TCP)	VooDoo Doll		
1349 (UDP)	Back Orifice DLL		
1492 (TCP)	FTP99CMP		
1600 (TCP)	Shivka-Burka		
1761 – 1764 (TCP & UDP)	sms-helpdesk		
1807 (TCP)	SpySender		

Table 2:
TCP or UDP Servers to Block at the Perimeter Router/Firewall from External Clients

Port(s) (Transport)	Server
79 (TCP)	finger
161 (TCP & UDP)	snmp
162 (TCP & UDP)	snmp trap
514 (UDP)	syslog
550 (TCP & UDP)	new who

Table 3:
TCP or UDP Servers to Allow Limited Access at the Perimeter Router/Firewall

Port(s) (Transport)	Server
20 (TCP)	ftpdata
21 (TCP)	ftp
22 (TCP)	ssh
23 (TCP)	telnet
25 (TCP)	smtp
53 (TCP & UDP)	domain
80 (TCP)	http
110 (TCP)	pop3
119 (TCP)	nntp
123 (TCP)	ntp
143 (TCP)	imap
179 (TCP)	bgp
389 (TCP & UDP)	ldap
443 (TCP)	ssl
1080 (TCP)	socks
3128 (TCP)	squid
8000 (TCP)	http (alternate)
8080 (TCP)	http-alt
8888 (TCP)	http (alternate)

Table 4:
ICMP Message Types to Allow Outbound at the Perimeter Router/Firewall

Message Types	
Number	**Name**
4	source quench
8	echo request (ping)
12	parameter problem

Table 5:
ICMP Message Types to Allow Inbound at the Perimeter Router/Firewall

Message Types	
Number	**Name**
0	echo reply
3	destination unreachable
4	source quench
11	time exceeded
12	parameter problem

This section describes methods using filters to defend the router, the firewall and the protected network from information gathering and attacks. Note that one needs to be careful with combining the below recommendations together in any filter in order to prevent contradictions or other problems.

- When creating a TCP/IP filter always delete any previous filter.
- Set logging for each statement in the filter that blocks access. This feature will provide valuable information about what types of packets are being denied and can be used in intrusion detection against one's network. Refer to the following section on Logging and Debugging for discussion of logging configuration in more detail.
- Provide IP address spoof protection for the protected network. For inbound traffic do not allow any IP packet that contains an IP address in the source IP address field from the following: the protected network, any local host address (127.0.0.0 – 127.255.255.255), any reserved address (10.0.0.0 – 10.255.255.255, 172.16.0.0 – 172.31.255.255, 192.168.0.0 – 192.168.255.255), or any multicast address (224.0.0.0 – 239.255.255.255). For outbound traffic allow IP traffic from the protected network and do not allow IP traffic that contains an external IP address in the source IP address field.
- Protect the router or the firewall from the Land Attack. This attack involves sending a packet to the router with the same IP address in the source address and destination address fields and with the same port number in the source port and destination port fields. This attack can cause a denial of service.
- Protect the router or the firewall from the TCP SYN Attack. The TCP SYN Attack involves transmitting a volume of connections that cannot be completed at the destination. This attack causes the connection queues on the router or the firewall to fill up, thereby denying service to legitimate TCP traffic.
- Protect the router, the firewall or the protected network from unnecessary ICMP traffic. There are a variety of ICMP message types, and some are associated with programs. Some message types are used for network management and are automatically generated and interpreted by network devices. For example, the ping program works with message type Echo. With Echo packets an attacker can create a map of the protected networks behind the router or the firewall. Also, he can perform a denial of service attack by flooding the router, the firewall or the hosts on the protected network with Echo packets. With Redirect packets the attacker can cause changes to a host's routing tables.

  For outbound ICMP traffic, one should allow the message types Echo, Parameter Problem and Source Quench. Otherwise, block all other ICMP message types going outbound. With Echo packets users will be able to ping external hosts. Parameter Problem packets and Source Quench packets improve connections by informing about problems with packet headers and by slowing down traffic when it is necessary. For inbound ICMP traffic, one should allow the following message types: Echo Reply, Destination Unreachable, Source Quench, Time Exceeded and Parameter Problem. Otherwise, block all other ICMP message types coming inbound.

- Protect the router, the firewall or the protected network from inbound traceroute. Traceroute is a utility that prints the IP addresses of the routers that handle a packet as the packet hops along the network from source to destination. On Unix operating systems traceroute uses UDP packets and causes routers along the path to generate ICMP message types Time Exceeded and Unreachable. Similar to ICMP Echo

packets, an attacker can use traceroute to create a map of the protected network behind the router or the firewall.

- Apply a filter to the router or the firewall to allow only a small set of computers (e.g., those used by the administrators) Telnet access to the router or the firewall. Log all successful and unsuccessful connections.
- If an SNMP server is necessary on the router or the firewall, then apply a filter to the router or the firewall to allow only a small set of computers (e.g., those used by the administrators) SNMP access to the router or the firewall. Log all successful and unsuccessful connections.

Example: Cisco IOS Routers

The following scenario steps through the recommendations listed above.

- The following commands show an example of how to clear out a previous version of an access-list before creating a new access-list.

```
Router(config)# no access-list 100
Router(config)# access-list 100 permit ip 10.2.9.0 0.0.0.255 any
Router(config)# access-list 100 permit ip 10.55.1.0 0.0.0.255 any
```

- The following commands show an example of how to set logging on an extended IP access-list statement.

```
Router(config)# access-list 102 permit tcp 10.4.6.0 0.0.0.255 any eq 80
Router(config)# access-list 102 deny ip any any log
```

Note that there is an implicit deny statement at the end of every access list on a Cisco router. This implicit statement blocks all other packets not permitted by the rest of the access-list. However, it does not log these packets. Thus, add the following statements at the end of each extended IP access-list. These statements will guarantee that the router will log the values for the source and destination ports for TCP and UDP traffic being denied.

```
Router(config)# access-list 106 deny udp any range 0 65535 any range 0
65535 log
Router(config)# access-list 106 deny tcp any range 0 65535 any range 0
65535 log
Router(config)# access-list 106 deny ip any any log
```

- Below are two example access-lists that provide IP address spoof protection. The first example is for inbound traffic to the protected network (e.g., 14.211.150.0).

```
Router(config)# access-list 100 deny   ip 14.211.150.0 0.0.0.255 any log
Router(config)# access-list 100 deny   ip 127.0.0.0 0.255.255.255 any log
Router(config)# access-list 100 deny   ip 10.0.0.0 0.255.255.255 any log
Router(config)# access-list 100 deny   ip 172.16.0.0 0.15.255.255 any log
Router(config)# access-list 100 deny   ip 192.168.0.0 0.0.255.255 any log
```

Router(config)# access-list 100 deny ip 224.0.0.0 15.255.255.255 any log

Router(config)# access-list 100 permit ip any 14.211.150.0 0.0.0.255

Router(config)# interface Ethernet1/2

Router(config-if)# description "external interface"

Router(config-if)# ip address 25.73.1.250 255.255.255.248

Router(config-if)# ip access-group 100 in

The second example is for outbound traffic from the protected network (e.g., 14.211.150.0).

Router(config)# access-list 102 permit ip 14.211.150.0 0.0.0.255 any

Router(config)# access-list 102 deny ip any any log

Router(config)# interface Ethernet0/1

Router(config-if)# description "internal interface"

Router(config-if)# ip address 14.211.150.17 255.255.255.240

Router(config-if)# ip access-group 102 in

Note that you can apply two access-lists to any interface on the router, one for network traffic leaving the interface and the other for network traffic entering the interface.

- The following commands show how to protect the router from the Land Attack.

Router(config)# access-list 101 deny ip host 198.26.171.178 host 198.26.171.178 log

Router(config)# access-list 101 permit ip any any

Router(config)# interface serial2/1

Router(config-if)# description "external interface"

Router(config-if)# ip address 198.26.171.178 255.255.255.248

Router(config-if)# ip access-group 101 in

- Protect the router against the TCP SYN Attack for the following two scenarios: blocking external access and limited external access. Below is an example for blocking external access on a Cisco router. The access list blocks packets from any external network that have only the SYN flag set. Thus, it allows traffic from TCP connections that were established from the protected network (e.g., 14.2.6.0), and it denies anyone coming from any external network from starting any TCP connection.

Router(config)# `access-list 100 permit tcp any 14.2.6.0 0.0.0.255 established`

Router(config)# `access-list 100 deny ip any any log`

Router(config)# `interface serial0/0`

Router(config-if)# `description "external interface"`

Router(config-if)# `ip access-group 100 in`

Below is an example for allowing limited external access on a Cisco router. Using the TCP intercept feature, the access list blocks packets from unreachable hosts; thus, it only allows reachable external hosts to initiate connections to a host on the protected network (e.g., 14.2.6.0). In intercept mode the router intercepts a TCP connection and determines if a host is reachable. If successful, the router establishes the connection; otherwise, it prevents the connection. This protection does not stop reachable hosts from performing this attack against the router or the protected networks.

```
Router(config)# ip tcp intercept list 100

Router(config)# access-list 100 permit tcp any 14.2.6.0 0.0.0.255

Router(config)# access-list 100 deny   ip  any any log

Router(config)# interface e0/0

Router(config-if)# description "external interface"

Router(config-if)# ip access-group 100 in
```

- The following commands show how to allow outbound from the protected network (e.g., 14.2.6.0) only the following ICMP message types: Echo, Parameter Problem and Source Quench.

```
Router(config)# access-list 102 permit icmp 14.2.6.0 0.0.0.255 any echo

Router(config)# access-list 102 permit icmp 14.2.6.0 0.0.0.255 any parameter-problem

Router(config)# access-list 102 permit icmp 14.2.6.0 0.0.0.255 any source-quench

Router(config)# access-list 102 deny   icmp any any log
```

The following commands show how to allow inbound to the protected network (e.g., 14.2.6.0) only the following ICMP message types: Echo Reply, Destination Unreachable, Source Quench, Time Exceeded and Parameter Problem.

```
Router(config)# access-list 100 permit icmp any 14.2.6.0 0.0.0.255 echo-reply

Router(config)# access-list 100 permit icmp any 14.2.6.0 0.0.0.255 unreachable

Router(config)# access-list 100 permit icmp any 14.2.6.0 0.0.0.255 source-quench

Router(config)# access-list 100 permit icmp any 14.2.6.0 0.0.0.255 time-exceeded

Router(config)# access-list 100 permit icmp any 14.2.6.0 0.0.0.255 parameter-problem

Router(config)# access-list 100 deny   icmp any any log
```

- The following command shows how to block inbound traceroute from a Unix computer.

```
Router(config)# access-list 111 deny udp any any range 33434 33534 log
```

- The following commands show how to allow Telnet access from certain computers on the protected network (e.g., 14.4.4.0) to the router via an extended IP access-list. The administrator can telnet to any interface IP address on the router. However, the router converts any interface IP address to 0.0.0.0. Thus, the unusual destination IP address 0.0.0.0 must be used in the access-list.

```
Router(config)# access-list 105 permit tcp host 14.4.4.10 host 0.0.0.0 eq 23 log
Router(config)# access-list 105 permit tcp host 14.4.4.11 host 0.0.0.0 eq 23 log
Router(config)# access-list 105 permit tcp host 14.4.4.12 host 0.0.0.0 eq 23 log
Router(config)# access-list 105 deny   ip  any any log
Router(config)# line vty 0 4
Router(config-line)# access-class 105 in
```

- The following commands show how to allow SNMP access from certain computers on the protected network (e.g., 14.4.4.0) to the router via a standard IP access-list.

```
Router(config)# access-list 10 permit 140.4.4.10
Router(config)# access-list 10 permit 140.4.4.11
Router(config)# access-list 10 permit 140.4.4.12
Router(config)# snmp-server community snmp72str1ng64 ro 10
```

Logging and Debugging

Logging on a router or a firewall offers several benefits. It informs the administrator if the router or the firewall is working properly or has been compromised. It can also show what types of attacks are being attempted against the router, the firewall or the protected network.

The following are recommendations for logging and debugging:

- Send the most serious level of logs to the console on the router or the firewall in order to alert the administrator.
- Send the logs to a log host, which should be a dedicated computer on the protected network whose only job is to receive logs. The log host should have all unnecessary servers and accounts disabled except for syslog.
- Configure the router or the firewall to include more specific time information in the logging and in the debugging. Direct the router or the firewall to at least two different, reliable network time protocol (NTP) servers to ensure accuracy and availability of time information. Set all NTP messages with the same IP source address of an interface on the internal network. This configuration will allow the administrator to create a TCP/IP filter that allows time information only from the internal IP address of the router or the firewall to the external NTP servers. This filter will help to prevent spoofing or flooding NTP messages to the router or the firewall. Include a more specific timestamp in each log message and each debug message. This will allow an administrator to trace network attacks more credibly.

- By default, a log message contains the IP address of the interface it uses to leave the router or the firewall. Instead, set all log messages with the same IP source address of an interface on the internal network, regardless of which interface the messages use. This configuration will allow the administrator to create a TCP/IP filter that allows logs only from the internal IP address of the router or the firewall to the logging host. This filter will help to prevent spoofing or flooding log messages to the logging host.
- Finally, consider also sending the logs to a dedicated printer to deal with worst-case scenarios, e.g., failure of the log host.

Example: Cisco IOS Routers

The following scenario steps through the recommendations listed above.

- Enable the router's logging capability with the following command.

```
Router(config)# logging on
```

- Set the syslog level to be sent to the router console. The following command is an example.

```
Router(config)# logging console informational
```

Note that the effect of the `log` keyword with the IP extended access-list statements depends on the setting of the `logging console` command. The `log` keyword takes effect only if the `logging console` syslog level is set to 6 (`informational`) or 7 (`debugging`). If the level is changed to a value less than 6 and if the `log` keyword is used within an IP extended access-list command, then no information is logged to the log host or displayed to the console. Refer to the previous section on TCP/IP Filters for discussion of access-lists in more detail. Finally, disable logging to all terminal lines except for the router console with the following command.

```
Router(config)# no logging monitor
```

- Set the IP address of the log host. Set the syslog level to be sent to the log host. Set the syslog facility type in which log messages are sent. The following commands are examples.

```
Router(config)# logging 10.1.1.200
Router(config)# logging trap debugging
Router(config)# logging facility local7
```

- The following commands show an example of how to set time information for the logging and for the debugging.

```
Router(config)# ntp server 192.168.41.40
Router(config)# ntp server 192.168.41.41
```

```
Router(config)# ntp source Ethernet0/1
Router(config)# service timestamps log datetime localtime show-timezone
Router(config)# service timestamps debug datetime localtime show-timezone
Router(config)# clock timezone EST –5
Router(config)# clock summer-time EDT recurring
```

- The following command shows an example of how to set all log messages with the same IP source address of a router interface.

```
Router(config)# logging source-interface e0/1
```

General Recommendations

It is highly recommended that the configuration files for the router or the firewall be created, stored and maintained on a computer offline in ASCII format. These files will contain any comments that can help give perspective to the configuration settings and the filters. Also, changes to the filters can be done with much more ease and accuracy. Then the file can be transferred from the computer to the router or the firewall. This is invaluable for diagnosing suspected attacks and recovering from them. Finally, protect the contents of the configuration files from unauthorized individuals.

Windows NT 4.0 and Windows 2000

Service Packs And Hotfixes

A service pack is a periodic update to the operating system that contains fixes to vulnerabilities and bugs. To date, Microsoft has released six service packs for Windows NT 4.0 and two service packs for Windows 2000. Updates addressing specific vulnerabilities and bugs introduced between Service Packs are called hotfixes. Service packs are cumulative, meaning they include all hotfixes from previous service packs, as well as new fixes.

In addition to installing the latest service packs, it is important to install new hotfixes, as these patches will often address current attacks that are proliferating throughout networks. Although Microsoft recommends applying a hotfix only if a system experiences the specific problem, it is recommended that all security-related hotfixes be installed immediately after installation of the latest service pack. If a service pack is reapplied at any time, the hotfixes must also be re-installed.

Checking System Patch Status

A major challenge for network administrators is keeping up to date on the latest patches. Microsoft now provides a Network Security Hotfix Checker (Hfnetchk.exe) tool that lets administrators scan their servers -- including remote ones -- to ensure that that they are up to date on all security patches for Windows NT 4.0, Windows 2000, IIS 4.0, IIS 5.0, IE and SQL Server. Detailed information on Hfnetchk, including download location, is available in Knowledge Base article Q303215 at http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/hfnetchk.asp.

Windows NT 4.0 Patches

To achieve the highest level of Windows NT security, install Service Pack 6a and the post Service Pack 6a hotfixes. For a complete list of available service packs and hotfixes go to http://www.microsoft.com/ntserver/nts/downloads/recommended/SP6/.

Microsoft has provided the Security Rollup Package (SRP) as a mechanism for managing the rollout of security related fixes. The SRP includes the functionality from many security patches released for Windows NT 4.0 since the release of Service Pack 6a. The SRP includes post-Service Pack 6a fixes that were delivered via Microsoft security bulletins as well as a small number of fixes that were not addressed through this forum. For a complete listing of all fixes in the SRP, refer to Microsoft Knowledge Base Article (Q299444), "Post-Windows NT 4.0 Service Pack 6a Security Rollup Package (SRP)," at http://support.microsoft.com/support/kb/articles/q299/4/44.asp.

Fixes not included in the SRP:

Fixes for newer vulnerabilities may not be included in the SRP. These must be applied separately and may be downloaded from http://www.microsoft.com/ntserver/nts/downloads/recommended/SP6/. In addition, the following vulnerability affecting Windows NT 4.0 systems is not included in the SRP.

Enhanced Security Level Hotfix - When changing the domain password with the C2 security registry entry enabled a “Stop 0x1E” error message may occur. The problem occurs if the administrator has Service Pack 6a (SP6a) installed and the following registry entry is set:

Hive: HKEY_LOCAL_MACHINE
Key: SYSTEM\CurrentControlSet\Control\Session Manager
Value: EnhancedSecurityLevel
Type: REG_DWORD
Data: 1

This key ensures that Object Manager can change the attributes of a kernel object in the Object table for the current process if the previous mode of the caller is kernel mode. When attempting to change the password after setting this registry value, the following error message will be received: Stop 0x0000001e (0xc0000005, 0x8019bb12, 0x00000000, 0x0000022c)

A supported fix that corrects this problem is now available from Microsoft, but it is not available for public download. To resolve this problem immediately, contact Microsoft Product Support Services to obtain the fix. This hotfix is also available from NSA. For a complete list of Microsoft Product Support Services phone numbers and information on support costs, please go to the following address on the World Wide Web:
http://support.microsoft.com/directory/overview.asp

Windows 2000 Patches

To achieve the highest level of Windows 2000 security, install Service Pack 2and the post Service Pack 2 hotfixes. For a complete list of available service packs and hotfixes, refer to http://www.microsoft.com/windows2000/downloads/default.asp

List Of NT/Windows 2000 Security Measures

This list of NT/Windows 2000 security measures is by no means exhaustive. There are approximately 400 known vulnerabilities with Windows NT/2000 and associated applications. This list addresses less than 10 percent of those vulnerabilities. It should also be understood that alleviating one's network of these vulnerabilities does not render the network "secure".

- Ensure that the file system is NTFS versus FAT. NTFS allows file access control to be set; FAT does not.
- Limit the information available from a null connection. Null connections (anonymous users) are included in the built-in **Everyone** security group; thus, anonymous users have access to any resources that the **Everyone** group has access to. Windows NT Service Pack 6a limits much of what an anonymous user can do. Prevent anonymous users from being able to enumerate account names and shares by setting the following registry key:

  Hive: HKEY_LOCAL_MACHINE
  Key: System\CurrentControlSet\Control\Lsa
  Name: RestrictAnonymous
  Type: REG_DWORD
  Value: 1

- Remove the Everyone group from the “Access this Computer from the Network” user right. Replace it with the Authenticated Users group. In Windows NT 4.0, this can be

accomplished under User Manager -> Policies -> User Rights. In Windows 2000, this can be done via the Security Configuration Toolset and Group Policy.

- Do not allow remote registry access. There are many registry keys that allow the Everyone group, and therefore anonymous users, read and/or set value permissions. If an unauthorized user was able to remotely edit the registry, he could modify registry keys in an attempt to gain elevated privileges. Restricting remote registry access is accomplished by setting security permissions on the HKLM\SYSTEM\CurrentControlSet\Control\SecurePipeServers\winreg key. It is highly recommended that only Administrators and System have remote access to the registry.
- Ensure that the Guest Account is disabled. Ensure that all accounts (service and user) have passwords regardless if the account is enabled or disabled.
- Disable LanMan authentication. LanMan passwords are used for backwards compatibility with older Windows operating systems (e.g., Windows 9x) and are simply the NT/2000 password converted to all uppercase and encrypted in a different way. LanMan passwords are easier to crack than NTLM hash because they are treated as two 7-character passwords. It is recommended that LanMan passwords be disabled. If Windows 9x boxes reside on the network, Directory Client Services (available on the Windows 2000 CD) must be installed on these systems in order to allow NTLM version 2 authentications. To disable LanMan authentication, set the following registry key:

  Hive: HKEY_LOCAL_MACHINE
  Key: System\CurrentControlSet\Control\Lsa
  Name: LMCompatibilityLevel
  Type: REG_DWORD
  Value: 5

- Close ports 135, 137, 138, and 139 either at the premise router or firewall. For networks containing Windows 2000 systems, also block port 445. These ports are needed in an internal network, but not externally. Blocking these ports will stop many attacks against Windows NT and Windows 2000. Also, remove unneeded protocols (e.g. NetBeui, IPX).
- Out-of-the-box permissions on Windows NT system files and registry keys are overly permissive. Replace the Everyone group with the Authenticated Users group on critical system folders and files (e.g. WINNT, system32) and registry keys (e.g., HKLM\Software\Microsoft\Windows\Run and HKLM\Software\Microsoft\Windows NT\CurrentVersion\AEDebug).
- Restrict permissions on network shares. When a share is created, the default access control is Everyone having Full Control. Restrict the share permissions to only those groups that need access.
- Remove all services that are not required (e.g., Telnet, FTP, Web). Ensure proper placement of services on the network (e.g. RAS or Web service should not be on a Domain Controller).
- Enable auditing. At a minimum, audit logons and logoffs, failed attempts at exercising user privileges, and system events such as shutdowns.
- Review Trust Relationships between domains. Remove unnecessary trusts.

Microsoft Applications

Vulnerabilities in applications such as Outlook, Microsoft Exchange, SQL Server, and IIS may open a network to attack. Therefore, it is important that applications be kept current with the latest patches and service packs. Microsoft provides several tools for improving application security. Some of these tools are listed below, along with a web reference to follow for more information.

URL Scan Security Tool – Allows web server administrators to restrict servers to ensure that they only respond to legitimate requests.
http://www.microsoft.com/technet/security/URLScan.asp

IIS Lockdown Tool - A Microsoft tool for securing IIS 4.0 or 5.0 web server.
http://www.microsoft.com/technet/security/tools/locktool.asp

Improved Outlook E-mail Security Update - A new version of the Outlook E-mail Security Update is available that provides protection against additional types of e-mail-based attacks.
http://office.microsoft.com/downloads/2000/Out2ksec.aspx

HFNetChk Security Tool – In addition to operating system patches, checks security patches for IIS 4.0, IIS 5.0, IE, and SQL Server.
http://www.microsoft.com/technet/security/tools/hfnetchk.asp

Microsoft Personal Security Advisor - A Microsoft tool for checking that workstations are current with all security patches and configured for secure operation.
http://www.microsoft.com/technet/security/tools/mpsa.asp

UNIX Networks

The following recommendations can be taken to secure UNIX networks.

Startup Scripts

Check the permissions and ownership of files. If they allow world access, browse scripts to see if any unusual process or script is started, especially if in user directories. Files and directories should be owned by root/root or root/sys with limited or no world write or execute permissions so that they cannot be modified or exploited by unauthorized users. User startup files should be owned by the individual user and have permissions of 640. In each user's directory, check for hidden files (e.g.,. `login`, `.profile`, etc...) that have extensions, such as `.old/.backup` or begin with *"..", "...".*

Services/Ports

Run a port scanner, such as *nmap* (available at http://www.insecure.org/nmap) to list open ports and services. Many UNIX services have well known security vulnerabilities associated with them, which allow root access. All unnecessary services (e.g., r*exd, rquotad, talk, sadmind, kcmsd, rstatd, fs, exec, daytime, walld, fingerd, systat, rusersd, sprayd, uucpd,*

chargen, time, echo, display, tftp, comsat and discard) should be disabled by placing a # at the beginning of the lines in the */etc/rc** files or in the */etc/inetd.conf* file that caused the program to be executed. In addition, these ports should be blocked at the perimeter router or firewall.

System Trust

There are various ways for UNIX systems to allow access to a machine or an account without providing a password. Through the use of *.rhosts ,.forward, .netrc, hosts.lpd,* and *hosts.equiv* files, it is possible for a user on one system to access another system without providing a password. This practice should be reviewed for necessity. An intruder breaking into an authorized user's account can use those same trusts to reach multiple machines with little effort. Do not use plus signs (+) in these files as they allow wider access (to users and/or machines) than might be intended. Prohibit root from logging directly into a remote system through either the */etc/ttys, /etc/ttytab,* or */dev/default/login* files.

R Commands

Telnet and the *"r commands"* (*rlogin, rcp, rsh* and *rexec*) may transmit the username and password in the clear making it easy for an attacker with a sniffer to capture this information and act as a trusted user. If trust relationships are set up, "r" commands enable someone to access a remote system without supplying a password. If an attacker gains control of any machine in a trusted network, then he or she can gain access to all other machines that trust the hacked machine. If these services are not required, they should be disabled; otherwise, install *openssh* (available at http://www.openssh.com/). In addition, *ssh*, which includes *sftp*, is an alternative solution to FTP. The service encrypts all traffic including the password to reduce the threat of eavesdropping. Do not allow trust relationships.

Network Configurations

Check to see if network configuration files (such as *hosts, defaultdomain, defaultrouter, netmasks,* etc.) are owned by root/root and have permissions of 644. This is suggested to alleviate unauthorized modifications.

Patches

Ensure applicable system and security patches are current and have been installed. Note that patches may not be applied until a reboot occurs. Therefore, if a patch is listed in the output from *"Patchdiag", "showrev",* or whatever specific patch checker tool or UNIX command is used, but the machine has not been rebooted in awhile, there is a possibility that the machine may still be vulnerable.

User Accounts

Review all user accounts. Do they all have unique UIDs? This is important to enforce so that a person will not obtain the privileges associated with someone else's account or be able to read, delete, or modify another person's files. Check to make sure each shell field is set to a valid shell to alleviate malicious code from being executed and granting root access. The **nobody4** account is for SunOS backward capability and should be deleted, if not needed. Make sure every line in the */etc/passwd* file is in the proper format to alleviate accidental logins by an unauthorized person. Permissions for most home directories should be 740. Ftp and uucp users may be exceptions. Check automount directories for unauthorized

automount maps. All maps should be protected with permissions 755 and owned by root/root.

System administrators should not directly log in as root, but rather as themselves and then switch user (*su*) to root. This is important for accountability. An administrative group (e.g. wheel) should be created in the */etc/group* file and each administrative user should belong to that group. Once the administrative group has been created, the *"su"* program should have its ownership, group, and permissions changed (root/wheel, 750).

Permissions

Look for *'setuid'* or *'setgid'* files and programs. Drop the *'suid'* and/or *'sgid'* bits, if not needed. Look for world writable directories and files and drop the world permissions, if not needed. This will help prevent unauthorized access or the insertion of malicious code. Also check for files owned by root and are mode world read/write. These files may indicate a potential symbolic link attack if one of the parent directories are writable by the attacker. Check umask values. Suggest that user umasks be set to 022.

Cron/At Jobs

Check permissions on *cron* and *at* job *.allow* and *.deny* files. They should be 644, root/sys. *.allow* files permit users to use *crontab* and *atjobs. .deny* restricts these users from access. If *.allow* files do not exist, then the system checks the *.deny* files. If neither file exists, depending on system configurations, it either allows just root or everyone to write cron/at jobs. Check to make sure that all *cron* and *at* jobs have valid users associated with them. Crontab and atjob files should be owned by the specific user associated with them and have permissions of 600. Make sure that all cron or at jobs use absolute paths (full path names).

Core Dumps

Check for *core* files. Most reside in the "/" directory, but others may be located elsewhere. Core files may contain sensitive system data or user passwords. Remove core files from the system. Configure the system so that when core files are created, they are automatically redirected to */dev/null* or have a *ulimit=0*.

Network Services

NIS

Ensure NIS maps do not contain system accounts. Establish a securenets file in the NIS environment as an effective way to secure access. Look for strange entries within the NIS *ypserv.log* file. This is suggested to prohibit unauthorized access.

NIS +

Check to see if NIS + is running in yp compatibility mode. If the "-YP" argument is there, the server is in NIS emulation mode and all exploits for NIS apply. Delete **nobody** permissions so that unauthorized persons don't have access to the NIS+ tables. Make sure world is given read-only permissions, except for the password table, which shouldn't allow any world access. When checking table permissions and access rights, they should match. Individual users should only have read access to the password table to prevent users from changing their UID value to 0, which would give them root access.

NFS

Ensure the NFS environment is not exporting sensitive file systems to the world (i.e., /etc,...) regardless of permission settings. Ensure no critical file systems are shared to the world with read-write access. Ensure exported file systems are directed to specific hosts via the */etc/dfs/dfstab* file or via *netgroups*. Ensure files are not exported to *"localhost"*. Ensure files are shared with the *"nosuid"* designator, unless suid is required. Ensure the anonymous user has been established correctly. If the system has anon=0, then *"root"* users of remote machines will have the UID specified after the "=" equal sign. If the "root=" user has been established, then root users of the machines specified after the "=" equal sign will have a UID of zero on the remotely mounted file systems. Check all clients and servers to see which file systems are being mounted locally or remotely.

DNS

The Domain Name Service is the mechanism that Internet hosts use to determine the IP address that corresponds to a given hostname. Attackers often attempt zone transfers in order to gather information about a local network. One way to prevent zone transfers is to block tcp port 53. This can be done via firewall or or router access filters. Disable the BIND name daemon (*named*) on systems not authorized to be DNS servers. On the servers, upgrade to the latest version of BIND and run it as a non-privileged user. Run DNS in a *chroot*ed environment. Hide the version string via the version option in named.conf.

Sendmail

Upgrade to the latest version of *Sendmail*. Do not run *Sendmail* in daemon mode (turn off the -bd option) on machines that are not mail servers or relays. Do not display the version number through sendmail banners. Ensure that the decode alias is not available. Decode should be removed or commented out of the `/etc/aliases` file so that it does not pipe to the 'uudecode' command and allow an attacker to overwrite system files. Check for `.forward` files as they can open up the system to attacks. If not needed, remove them or link to `/dev/null`. If needed, permissions should be 740 and owned by the user. If the system is not a server or does not have to listen for incoming mail, rename the *sendmail* startup script, binaries, and configuration files and change their permissions to 000.

Logs

System logging is crucial for troubleshooting and tracking unauthorized user accesses. Ideally, logs should be kept locally AND sent to a central loghost that does nothing but accept and store log messages. Your network security policy should help dictate which events need to be audited. *Logcheck* and *swatch* are tools that system administrators can use to examine log files for unusual activity, based on key phrases or specially set string patterns. They can also send emails to the system administrators, alerting them to possible unauthorized activity. Both are open source tools.

X-Window Environments

Remove the X Windowing environment on the server. By removing the Common Desktop Environment (CDE) and/or SUN's OPENWINDOW environment, the network server will not be susceptible to a variety of vulnerabilities.

Distributed Server Functions

It is commonly considered a good security practice to distribute the server functions of a network among separate systems. For instance, the DNS server should be separate from the mail server, which should be separate from the firewall, etc. A number of products, such as

Borderware's firewall product, include the software to run a web server, mail server, DNS server and other server functions all from the firewall. However, this presents a single point of failure for the network and therefore an avoidable vulnerability. Ideally, network servers should be set apart from the user segment in a secure DMZ or secure server network. Most firewalls allow this and if it does not, it can easily be accomplished by using routers behind the firewall.

Chroot Environments

chroot() is a UNIX command used to run a command or interactive shell with a special root directory. This command can also be used to create a "virtual" operating system and directory tree. It would be inside of the new "virtual" directory tree that DNS, Sendmail, Web, and other various servers could run. This would provide a potentially safe location for the applications. Building a *chroot()*ed environment can be very useful in protecting the rest of the system and keeping hackers out, however, it is easy to make a mistake while creating this *chroot()*ed environment. If it is improperly installed, it could create more ways for the hacker to infiltrate the machine.

Interesting Files

Check for files that have no permissions or have invalid owners or groups. Sometimes admins will have specific files which have no permissions assigned to them. These files can be kicked off by a script, cronjob, or app that temporarily changes the permissions during the execution of the program, then resets the program back to the original state.

Peripheral Devices

Consider removing or restricting access to local or network peripheral devices. Malicious code is easily introduced into secure networks via their peripheral devices. If an external device is not required for a specific client or server, have it removed. If the device cannot be removed, disable access to it via the hardware or software. Check to see if local and network printers are secure. Floppies should not be introduced to a client or server without the prior consent of the local Network Security Officer representative.

Buffer Overflows

Ensure that SOLARIS systems have a non-executable stack environment enabled. This will help prevent buffer overflows that originate from within the memory stack. For buffer overflows in RPC services, block the RPC port 111 at the router or firewall.

System Utilities and Commands

Restrict access or remove system utilities such as compilers, debuggers, etc. These utilities aid an adversary in informational reconnaissance. System commands like "*strings*" and "*ln*" should either have their permission bits restricted or have them removed form the system.

Current OS Packages

Ensure that the system packages are current. Solaris 7 and 8 can check the integrity and accuracy of system packages. Sometimes malicious code can be introduced to a system as a system package.

Rootkits

There are several scripts that can be implemented on a UNIX system that will search for rootkits on clients and servers. Checking the integrity of system files against a master backup known not to be altered by malicious code is also a good practice.

Security Tools

To ensure and maintain the integrity of the network servers, it is important to constantly monitor them for signs of malicious activity. There are a number of tools that can aid an administrator in this task. Two of these tools that are commonly implemented are Tripwire and TCPD.

Tripwire

Tripwire monitors the permissions and checksums of important system files to easily detect files that have been replaced, corrupted, or tampered. For example, if an intruder gains access to the server and replaces the */bin/ls* command with one that performs unwanted functions, tripwire will send an alert. Tripwire will send the system administrator a report each night. Tripwire calculates the checksums of executable files from a clean install. It then recalculates these checksums and compares them on a regular basis. Since some hackers are skilled enough to spoof the checksums on modified files, tripwire uses two different checksum methods. It is important to save the original checksums on a non-rewriteable CD on the system. This ensures data integrity.

TCPD

TCPD, also referred to as "TCP wrappers," allows one to log connections to TCP services such as *telnet, rlogin* and *finger*. In addition, it allows one to restrict which systems can connect to these services via two files, *hosts.allow* and *hosts.deny*. Both of these features can be very useful when tracking or controlling unwanted guests on a network. TCPD is easy to install and does not require modification to existing network programs. Just modify the */etc/inetd.conf* file to execute TCPD instead of the actual program. TCPD will then do any necessary logging and security checks before running the real daemon.

For example, if the */etc/inetd.conf* originally contained this line:

```
telnet stream tcp nowait root /etc/in.telnetd in.telnetd
```

Change it to this:

```
telnet stream tcp nowait root /usr/etc/tcpd in.telnetd
```

UNIX Web Servers

This section describes security configuration for UNIX web servers, using Apache as the example. It is assumed that Apache has been installed from the distribution and that none of the security parameters has been modified that come default in the original setup.

General Guidance

- Ensure that the computer that runs the web server is dedicated. It should not have other uses, e.g., being a client workstation or print server. Always upgrade to the latest version of the web server available that is not the beta version.
- Do not perform development work on the operational web server. All data should be in final form and simply copied into place. Create a secondary mirror of the server for all development services and experimentation. Transfer data to the web server by tape, disk, or CD. Do not use FTP or telnet for data transfer.
- Remove all unnecessary services on the web server, including FTP, telnet, and X Windows. If that is not an option, make sure to run `tcpwrappers` on the open services. Use a port scanner to check for open ports on both the TCP and UDP protocols. If possible, use command line interfaces instead of X Windows. Using an X windowed interface opens up ports that cannot be effectively closed and still have the system remain functional. Since the server should be in production mode only, only a command line is required to update the site. Testing of the site should be done from a separate client.
- Isolate the web server physically and virtually. If possible allow local access to the web server to the fewest number of people with a minimal number of users. Keep the web server close to the administrator, the web engineer, or the webmaster. Keep the web server on a LAN segment separate from the rest of the IT infrastructure. Do not mount or share services to and from the server.

Example: Apache

As of 26 September 2001, Apache 1.3.20 is the latest version and is available at `http://httpd.apache.org`

- Ensure the user running the Apache web server is set to `nobody`. In the `httpd.conf` file in the `/usr/local/apache/conf` directory, make sure that the effective user is `nobody` and that the group option is also set to `nobody`. Below are the lines to add to the file.

  User nobody

  Group nobody

- Ensure that user nobody does not own or have write access to the htdocs or cgi-bin subdirectories or any other subdirectory under these. Below are the commands to set ownership of these directories to root and to restrict write access to only root.

  chown -R root /usr/local/apache/htdocs

chown -R root /usr/local/apache/cgi-bin

chmod 755 /usr/local/apache/htdocs

chmod 755 /usr/local/apache/cgi-bin

- Do not store cgi-bin related data in a directory accessible to the web server. For example, create another directory called cgi-data in /usr/local/apache alongside cgi-bin and htdocs. Have the cgi scripts use that directory for data storage and manipulation.
- Turn off AutoIndexing and Follow Symbolic Links. By default, Apache usually comes with automatic indexing of directories enabled. Look in the httpd.conf file (usually in the /usr/local/apache/conf directory) for the following line.

  <Directory "/usr/local/apache/htdocs">

  Within those set of options you will see an Options line that may look like the following.

  Options Indexes FollowSymLinks Multiviews

  This configuration means any requests for a directory that do not find an index file will build an index of what is in the directory. Also, any symbolic link in the document directory will also be followed even if it is outside of the web server's purview. For example, a symbolic link may be made to the root directory, giving at least read access to a great deal of the system as the owner of the web server process.

  For the most secure/functional Directory options, this segment of the httpd.conf file should look like the following.

  <Directory "/usr/local/apache/htdocs">

  Options Multiviews

  AllowOverride None

  Order allow,deny

  Allow from all

  </Directory>

Refer to the following URLs for further guidance:

- http://httpd.apache.org/docs/misc/security_tips.html
- http://www.linuxplanet.com/linuxplanet/tutorials/1527/1/
- http://www.modperl.com/perl_conference/apache_security/
- http://www.bignosebird.com/apache/a11.shtml

Intrusion Detection Systems (IDS)

This section of the 60 Minute Network Security Guide departs from the explicit detail of previous sections and provides a brief overview of Intrusion Detection Systems, describing in general terms the steps to be taken when deploying IDS in your environment.

Generally, there are two types of IDS: host based and network based. Host based IDS monitor security within a network component, such as a server or a workstation. Network based ID systems monitor the traffic between network components and networks. Some IDS are strictly network based, whereas others are a combination of network and host based.

Most IDS are comprised of two components, sensors and managers. Depending on the IDS type, sensors can be either network based or host based.

The following are steps to be taken when deploying an IDS.

Step 1 - Identify what needs to be protected

To maximize the utilization of IDS, the organization must first determine in order of priority what needs to be protected. For many organizations, the various servers, i.e., application, database, file and domain controllers, contain mission critical resources. Furthermore, depending on the organization, some departments may be more critical than others or must enforce different trust relationships. All of this must be defined in a priority list prior to deploying any IDS.

Step 2 - Determine what types of sensors are required

The types of sensors that are required are dependant on the priority list defined in Step 1. A host sensor would be used to monitor a critical server, whereas a network sensor would be used to monitor network entry points and critical network segments.

Another important issue to consider is how many sensors the organization can afford to buy. This number will influence how the sensors are deployed throughout the network, as the number of critical resources must be balanced against how many sensors can be acquired and maintained.

Step 3 - Configure host system securely

Prior to loading any IDS, the host that the IDS will reside on must be configured securely. Often, the vendor of the IDS will supply its own host to run the IDS sensor, in which case, the vendor should supply guidelines on how to secure that host. Otherwise, the IDS typically reside on Unix and Microsoft Windows NT/2000 hosts. The guidelines for securing Unix and Microsoft Windows NT/2000 systems are well documented elsewhere in this document.

Step 4 - Keep signature database current

The majority of IDS that are currently available for use are signature based. Because new vulnerabilities and attacks are being discovered daily, the signature database must be kept current. The respective vendors should supply the latest signatures for their IDS.

Step 5 - Deploy IDS sensors

The final phase is to actually deploy the IDS. The following scenarios are based on how many sensors are available for deployment versus what is deemed critical.

Scenario 1

If the organization can only afford to purchase and monitor one sensor of any type, then it should be a network sensor. As described earlier, a network sensor is much better suited to monitoring large segments of a network, whereas a host sensor is limited to monitoring the system that it resides on. In this scenario, the ideal location to place the sole network sensor is in the DMZ, between the external router and the firewall, as shown in Figure 1. In spite of having only one sensor, this design allows the IDS to be used for maximum effectiveness. By placing the IDS sensor between the external router and the firewall, the sensor can monitor all network traffic going to and coming from the Internet.

Furthermore, because the router can filter all incoming traffic from the Internet, the IDS sensor can be tuned to ignore certain types of attacks, thereby allowing the sensor to operate with maximum efficiency.

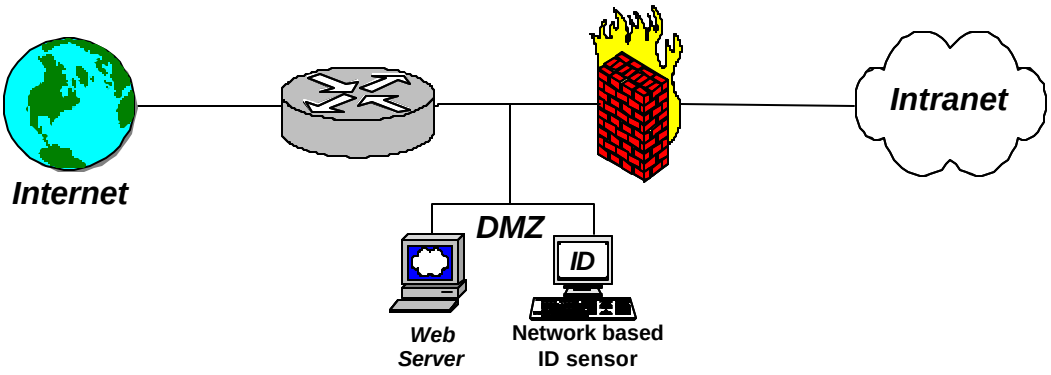

Figure 1 - Deploying 1 ID system

Scenario 2

In the case where only two sensors of any type can be acquired and maintained, then they should be network sensors. Like the previous scenario, one of the sensors should be placed in the DMZ, between the external router and the firewall. The second sensor should then be placed between firewall and the intranet, as shown in Figure 2. The second sensor can indicate what attack breached the firewall. By strategic placement of these two sensors, all access points from the Internet will be monitored.

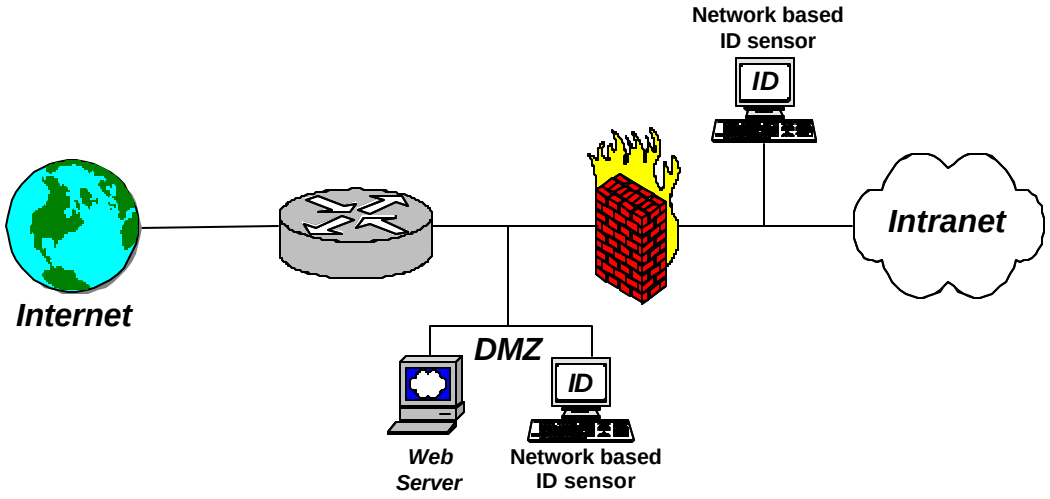

Figure 2 - Deploying 2 ID systems

Scenario 3

If more than two sensors of any type can be acquired and maintained, then at least two should be network sensors. Those sensors should be deployed as described in Scenario 2. If a critical LAN within the intranet needs to be protected, then a network sensor should be placed at the entry point to that LAN. The remaining sensors should be host sensors that are loaded onto critical servers, such as domain controllers, file servers, web servers, and mail servers. The order of what is deemed critical is determined by the organization, as directed in Step 1.

Step 6 - Management and Configuration

The other component of IDS, the manager, should be centrally located where dedicated security staff can monitor the health of the systems and network. Many organizations have a Network Operations Centers (NOC) that fulfills the role of a central location to place the manager. IDS sensors could then report all alerts to the NOC, thereby allowing the security staff to respond quickly to attacks and to notify the appropriate authorities, such as CERT technicians.

The other issue to consider is how to configure the sensors. Careful configuration of the sensors can increase the effectiveness of IDS and all unnecessary signatures should be disabled. For example, if the network is entirely composed of Microsoft Windows NT systems, then the sensors can be configured to ignore any attacks that are directed against Unix systems. Therefore, if the organization has a priority list as defined in Step 1, as well as knowing the network intimately, it can benefit greatly from having a properly configured IDS.