Solaris Network Security

Table of Contents

1. Purpose

The purpose of this document is to define the procedures and guidelines needed to insure the continued integrity, availability and confidentiality of the software systems, databases and data networks that are critical to <COMPANY NAME>.

2. Objective

Information is an important asset and <COMPANY NAME> will implement security measures to protect those assets. It is also in the company's interest to insure that each member of the <COMPANY NAME> community takes security measures individually to protect those assets.

3. Scope

The principal goal of this handbook is to provide Network Administrators for the <Company Name> Data Center a quick reference guide in proper security network and computer security. This policy addresses the <Company Name> Data Center and is not meant to be a guide for <Company Name> corporate. The Network Security Office (NSO) will provide specific guidelines for <COMPANY NAME> standards.

The policy pertains to the network administrators and all that work in or around the Data Center. This is to include contractors, customers, <Company Name> staff and vendors.

4. Physical security

Physical security involves the protection of computing resources from unauthorized access and from environmental hazards such as fire, water and power failure. Inadequate protection of computer hardware and software can limit its life, lead to system failures, and most important, result in loss of data.

4.1 Limiting Exposure To Environmental And Physical Threats

Perform a review of the physical and environmental threats that reside in the data center. Generally, a casual walk-through of your department will detect the more obvious threats. Procedures that can limit your environmental and physical exposures include:

- Place computer hardware in a clean environment, away from direct sunlight and windows.
- Connect all computer hardware to surge protectors and uninterruptable power supplies.
- Ensure there are sufficient electrical outlets.

- Ensure the file server is in an area not accessible by the public and lock the system console when not in use.

- Inventory hardware and software periodically.

5. Network Security

Protection of networks and their services from unauthorized modification, destruction, or disclosure, and provision of assurance that the network performs its critical functions correctly and there are no harmful side-effects.

6. Compliance

6.1 Introduction

<Company Name> is required to ensure that they have in place effective security systems designed to current industry accepted standards, including login control, data access permission, and remote/local sessions, in order to minimise the potential for unauthorised or fraudulent access to <COMPANY NAME> systems. The purpose of which deters fraudulent use of <COMPANY NAME> systems. Ref. ISP 090.12-2001, 090.12-2002

The purpose of proper Computer Security requirements and tasking on a system is to minimise the risks of that system. Risks occur when a computer system is vulnerable to a specific threat. By reducing either the system's vulnerabilities or the threat, the risk is reduced.

Vulnerabilities can be grouped into two categories; system vulnerabilities and data vulnerabilities. The computer portion of the system itself is relatively fragile by nature and is susceptible to system shutdown or slowdown without very much effort on the part of the threat. Even a small degradation in performance is sufficient to cause the system to become effectively worthless. The data that is stored transmitted, or processed in the system is a target for threats to the system. After all, the manipulation of the data is why the computer system is being developed. Whether classified or not, the data is needed by the user of the system or it should not be on the system. Any data that is needed by the user is also likely to be targeted by the malicious hacker.

Threats to systems come in a wide variety of areas. A system designed for use by an aircraft has the threat that the aircraft might crash. A system designed for use within a corporation or facility has the threat of power surges or fires. Either of these threats can effectively deny the user the use of the system without malicious intent. However, with our connectivity to the INTERNET growing at a rapid pace, there is an ever growing threat of computer hackers (those that intend to entry a computer system in an unauthorised manner). The modus operandi of

the malicious hacker is to exploit known vulnerabilities either for their own use of the system, or to deny use of the system to the users.

In order to limit your risks, you must configure your system with the proper security safeguards. Remember that the information that passes through your system is vital to the success of <Company Name> Business Systems.

7. Configuration Management

Configuring your security controls correctly is the single most important duty that you can do to thwart any unauthorised attempted entry into your system. Some system administrators avoid or ignore the security controls simply because they slow the system down, or take to much time to administer. System controls are a vital part of the information flow that we now enjoy today; without them, our systems become more vulnerable to malicious attacks. Please pay particular attention to the following paragraphs that highlight some configuration schemes that system administrators may employ.

7.1 Passwords

A strong password policy is the first line of defense against attack. Several methods exist to allow password files to be stolen (i.e. tftp, Mail Attack, etc.). Once a password file is downloaded, a hacker can then use a program such as "Crack" (described later in the tools section) to determine valid user ID's and plain-text passwords. Therefore, selection of strong passwords is essential. Additionally, requiring a password on each system minimize system breaches.

When choosing a password, do not use any word that can be found in a dictionary. Dictionary words also include foreign words, names, places, or mythological creatures. Instead, an alternative would be to use a "Pass Phrase". A Pass Phrase consists of upper and lower case letters with special characters intermixed (, & % etc.). For example, the phrase I would like to be home right now" would result in the password "iWl2BhrN". System administrators can use Crack to enforce the selection of good passwords. Passwords should also be changed at least every six months and be at least 8 alphanumeric characters long (Sun OS truncates passwords longer than 8 characters but other UNIX systems allow longer passwords). Finally, when initially assigning passwords, do not use a standard "default" password or the user's name in any form.

Examples of poor passwords are:

- anybody's name (especially spouse, children or pets)
- anybody's birth date (especially spouse, children or pets)
- any phone, social security, driver' license, or license plate numbers
- any information that is easily obtainable about you
- a word in any dictionary (English or foreign)
- passwords of all the same letter like, eeeeee
- simple keyboard patterns like, "qwerty"
- any of the above examples spelled backwards
- any of the above followed or pre-appended by a single digit

Examples of good passwords are:

- they have both upper and lowercase letters
- they have digits and/or punctuation characters as well as letters
- they should be easy to remember so they are not written down
- they are at least 8 (eight) characters long
- use pass phrases (as mentioned above) that is special to you, for example:

"None of this fancy stuff works for me"

"I would like to be home right now"

would be:

NoTfSw4m

iWl2BhrN

As an additional measure of security, the password file should also be "shadowed". Shadowing moves the encrypted passwords from /etc/passwd and places them in a file with limited accessibility. A "pointer" to the encrypted password replaces the encrypted passwords. Using the C2 or the Basic Security Module (BSM) package on Sun OS enables password shadowing.

There are many other things to examine in the password file. The following is a list of some of more common items:

- Any user with a UID of "0" has special privileges. Therefore, insure only user "root" has UID of "0"
- Insure all users have a password.
- Disable and/or change the password on all default accounts such as "sysdiag".
- On accounts such as news or ingress which do not require a login, replace the "shell" field (/bin/csh) with /bin/false. Now, even if someone does break into these accounts, they are logged off the system before they can do anything.
- Check for the occurrence of a "+" in the "user" field. This tells the machine to look for the user name in the Network Information Service (NIS) maps. If you are not using NIS, the system can be tricked into allowing unauthorized users on the system. This problem also occurs in the /etc/group file.
- Never allow "shared" accounts. A shared account is an account that allows several users to access a single account. Allowing shared accounts removes user accountability. Typical shared accounts that have been found are oracle, guest, database, etc.

7.2 New Accounts

New users must be forced to change their default password upon initial log-in to the system. This is to prevent compromise of passwords on new user accounts.

7.3 Root Access

The key to being able to adjust the behavior of the unix operating system is having root access. It gives you the ability to build the system, as you need to. But it also gives you the ability to tear it apart just as easily. It is for this reason that access is limited to those who have absolute justifiable need.

7.3.1 Responsibilities

Approval for granting root access privileges must first be reviewed and approved by the SAM for the system which the user is requesting root access. A signed AAF must be then submitted to the Data Center DCS for review. Upon approval of the DCS, the SAM and user will be notified by email that root access has been approved. It is the user's responsibility to report to the Data Center in order to obtain the root access password.

7.3.2 Procedure

The candidate will submit a signed AAF specifying:

- The reason root access is required
- Length of time root access is required
- System(s) the user requires root access.
- Signature on the AAF of an approved SAM. The SAM must be listed in the AAL in order to accept the request.
- The user will then be interviewed by systems administrator to verify his/her level of knowledge related to the job functions and operating system commands.
- After review and approval of the justification by the DCS, and upon the recommendation of the Unix administrator, privileged access will be granted.

7.3.3 Review Process

A quarterly review of those having root access is performed by the DCS at the end of March, June, September, and December.

A review will be initiated when an employee is transferred or released from <Company Name> employment. The review will be initiated within 24 hours of being notified of an employee's transfer or release from the company's employ.

7.3.4 Review Procedure

- An email will be sent to the SAM listing currently approved users that have access to the root password account.
- The SAMs will have 48 hours in which to respond to the email message validating or invalidating the names on the list.
- If no response is received within the 48 hours, the root password for that system will be changed immediately and the SAM notified via mail that the root password has been changed.
- If the SAM responds and indicates that some of the users listed should not have root access, the DCS will inform the Unix Administrator to immediately change the root access password. An email message will then be sent to those users that have been approved access informing them that they need to stop by the Data Center to receive the new password for that system.

7.3.5 Accessing Root account

When accessing the root account, it is mandatory that those having access use their normal user logon account to gain access to the system. Once validated by the system, root access can be initiated from the command prompt. Under no circumstances will root be used to logon into a terminal.

- Enter in your normal user logon account to access server.
- At the command prompt type “su” to access the root account.
- When prompted, enter in password given in order to access root privileges.
- Once completed with task(s) that required root access, type “exit” to terminate the root access. This will put you back to your normal user logon account.

7.3.6 Passwords

It is the responsibility of each member that has authorized access to the root account to insure the protection of the root account password. Compromise of the password could cause system integrity to be placed in question. If at any time it is believed that the root account password has been compromised, the Data Center needs to be notified immediately so actions can be taken to change the password account. Under no condition allow the following to exist:

- Root passwords **will not** be written down on paper and stored for future use other then that authorized and prescribed by the Data Center.

- Root account passwords **will not** be entered into a word processing document or spreadsheet for use later on.

- Root account passwords **will not** be entered into a database program of any sort for retrieval later on.

- Root account passwords are only permitted to be written down and stored in the Data Center. Root account passwords will be written down on a piece of paper, stored into a sealed envelope and placed in the safe for protection.

7.4 Password Aging

When you change your password, the new one you give is compared against those, which you have used in the past. If the new password matches a prior password, or if it differs from a prior password by only one character change, then the new password is rejected.

You must change your NetID password every 90 days. If you do not, your NetID will be deactivated, and with it your electronic mail account. Before the 90 days elapse, you will receive two electronic mail messages reminding you to change your password.

7.5 Failed Login Attempts

When a user forgets his or her password and tries to log into a system, this is known as a failed login. It could also be an attempt by an intruder trying to compromise your network defenses. It is for this reason that a policy needs to be in place that counters the intruder. It is the industry standard to lock out a user after a third attempt has failed to log into the system.

By forcing a user to re-establish a connection and reinitiate the login sequence, you deter dedicated hackers from running software designed to keep pumping usernames and passwords continuously in hopes of finding a right combination to gain access.

System administrators are required to enforce a maximum number of (5) consecutive unsuccessful login attempts before locking for a minimum of thirty (30) minutes. All unsuccessful log-in attempts will be logged to the administrators terminal and an entry added to the /var/adm/messages file.

7.6 Remote Logins

Do not allow remote logins (rlogin) unless absolutely necessary. Remote logins allow a user on another system to access your system without requiring a password. You can prevent this by not allowing any user to have a .rhost file in their home directory and properly configuring the /etc/hosts.equiv and /etc/hosts.lpd files. The /etc/hosts.equiv and /etc/hosts.lpd should list only systems absolutely essential to the operation of the system and should never contain a "+". The "+" says all other hosts are "trusted" and users from those hosts can perform remote logins. Also, NEVER have the first line of .rhosts, host.equiv, or hosts.lpd files begin with a "-". Remove any of the above mentioned files which only contain lines beginning with a "-".

7.7 Wheel Group And /Etc/Ttytab

Limit who can become root by using the "wheel" group in the /etc/group file. If the wheel group is empty, anyone can switch user (su) to root if they know the password. However, if there are any entries in the wheel group, only those users listed can su to root. This stops an unauthorised user who happens to learn the root password from becoming root. However, if someone does know the root password, they still will be able to log in directly as root if the /etc/ttytab file is not configured correctly. In the /etc/ttytab file, change the field "secure" to "unsecure" for all entries. This forces all users to login as an ordinary user then su to root and /etc/ttytab can now work in conjunction with the wheel group to control access to root. Another advantage of forcing users to su to root is it leaves a better audit trail of who has performed system administrator actions.

7.8 Path

Verify the default path for root and ordinary users. A "period (.)" in the path means check the current directory for the command. If someone has write permission to a directory, they could insert a Trojan horse and wait for someone else to run the command. For example, user A could modify the ls command to check the effective userid of the user/process performing the command. If the effective userid were root, the modified ls command would change the permissions to setuid root on another file owned by user A. Then the modified ls command would perform the real ls command. This would allow the user to become root at a later time. To eliminate this problem, a "period (.)" should appear last (or not at all) in the path specified in the .cshrc, .profile, or .login file. To protect against this, change the path in the /lib/Cshrc file and all user's .cshrc and/or .login files to place the "period(.)" at the end of the path. For root, we highly recommend removing the "period(.)" completely. This might cause you to type a little more when issuing commands, but it provides much greater security.

7.9 File/Directory Permissions

Secure user and root initialization files. Ensure only the owner can write to all initialization files such as .login and .cshrc. Since the read, write, and execute permission bits also pertain to directories, selection of the proper permissions is essential. For example, the /etc/security directory should not have world r/w permissions. Other directories which might require additional protection are /bin, /usr, /lib, and /dev. However, you must be aware that increasing the protection might stop a legitimate user/program from accessing a file that it requires. Therefore, care must be used when deciding to change the permission bits on a directory.

7.10 Suid/Sgid

Some programs require special privilege to perform their intended functions. For example, a user can change his password even though he does not have permission to write to the /etc/passwd file. This is accomplished by having a program, which is set-user-id (SUID), to root and runs with root's privileges. Programs, which are setuid or set-group-id (SGID), can be exploited if not properly protected. A favorite tactic of intruders is to get root permissions and then write a program with setuid permissions to root. Now after they logout, they can easily gain root access any time they want by executing the SUID program. All SUID/SGID programs should be known and protected to keep users from exploiting them. If unknown SUID/SGID programs are found, immediately investigate the programs function and how it was placed there. To detect SUID/SGID programs, the following command (or one similar) should be run at least monthly:

```
ls -algR | egrep '(rws|-ws|r-s)'
```

This command could produce a large listing of files and directories, which have a legitimate need to be SUID/SGID. Therefore, it would be a good idea to run this command soon after initial installation of the system to have a valid list to compare later results against.

7.11 Eeprom

On a multi-user operating system such as UNIX, running in single-user mode is the same as being user root. On Sun workstations, the EEPROM has three settings to control who can boot the system into single-user mode. The "NONE" setting allows anyone to boot into either multi-user or single-user mode. "COMMAND" allows the system to be booted into multi-user mode without the password and "FULL" does not allow the machine to booted at all without

the EEPROM password. We recommend setting the EEPROM to at least "COMMAND" to hinder unauthorised users from gaining access to the computer in single-user mode.

7.12 Trivial File Transfer Protocol (TFTP)

The Trivial File Transfer Protocol (tftp) is a mechanism that allows a user to download files without a formal login process. This allows anyone to take any world readable file off of your system. In the past, hackers have used this method to steal the password file so they can obtain a valid username and password for different systems. On many systems, this feature is turned on by default and is necessary for the booting of diskless clients. For systems without diskless clients this feature should be turned off by modifying the /etc/inetd.conf and /etc/services file. For systems with diskless clients using the TCP Wrappers and using the tftp -s (secure) option can restrict tftp.

7.13 Network File System (NFS)

The Network File System (NFS) is a convenient way of allowing machines on a network to access data stored on another machine on the network. There is some inherent security problems with the configuration of NFS. NFS "exports" data to other machines using the /etc/exports file. Where practical, the files should be exported with the least privilege possible and only to specified machines. For example, /export/home could be exported with both read and write privileges, but /usr should not be exported with write privileges. Also, it is better to list the machines the files are being exported to rather than export to "world".

Another problem is that some NFS servers export files to themselves. This creates an easily exploitable vulnerability, which could allow a hacker to gain root access to the machine. One way this occurs is by the use of "netgroups". For example, there is a network, which is comprised of two data servers, server A, and server B, with fifty client machines. Rather than add each machine individually into the /etc/exports file, a netgroup file listing all machines is made and the netgroup name is included in the /etc/exports file. If both server A and server B were listed in the netgroup, they would be exporting to themselves and therefore would be vulnerable. A better solution is to have two netgroups. Server A's netgroup would contain all machine names except server A and server B's netgroup would contain all machine names except server B. This way, neither server is exporting to itself. Refer to the system administration book and the systems man pages for a more detailed discussion of NFS exporting and configuration.

7.14 Network Information Service (NIS)

The Network Information Service (NIS formerly called the "Yellow Pages") is a method to manage a large network from a centralized location. A separate password table is made

available to other machines on the network to allow users on NIS clients to login using the centralized NIS password file. This is done through a mechanism on the client machine's password file. The line:

+::0:0:::

is a reference to check the NIS password file if the user name entered is not in the local password file. Unless properly configured, this entry introduces vulnerability in the system to allow users to log in with root privileges. This line on NIS clients should be changed to:

+:*:0:0::: (not applicable to Sun workstations Versions >= 4.X)

Also, the /etc/passwd file on local NIS client workstations should not contain any users except for default accounts (nobody, auditor, uucp, ...) and root. The permissions on the .pag files in the NIS directory also should not be given global write access. NIS is inherently insecure and has much unpatched vulnerability. Care should be taken when selecting which files should be "pushed" by NIS.

8. Vulnerability Countermeasures

Vulnerabilities come from many different aspects, from physical to mechanical. With the ever-growing population of connected computer users, the realization of vulnerabilities exists within the framework of the LAN. Most individuals use the Internet entirely for professional, business, or leisure activities; however, there are some users who use the Internet for personnel financial gain, malicious purposes, or just to see what they can get into. The following segments are countermeasures that you can employ.

8.1 Patches

Operating systems are very complex and interact with many applications (i.e. sendmail, ftp, telnet). Many "holes" in operating systems are known and others are constantly being discovered. These holes are exploitable by knowledgeable users to gain unauthorised privileges. When vendors learn of new vulnerabilities, they begin to work on a "patch" for the vulnerability; patches will modify permissions or binaries to fix known vulnerabilities. Security related patches should be loaded on a system as soon as they are released. The Configuration Management team announces patches. Most patches are available from the local vendor. Always contact your Configuration Management Office for information concerning patches.

8.2 Login Banner

The following login banner is mandated on all <Company Name> systems, which intend to monitor / audit intruders or authorized users performing unauthorised actions:

*"Official <**Company Name**> System for Authorized Use Only. Do Not Discuss, Enter, Transfer, Process, Or Transmit Information Of Greater Sensitivity Than That For Which This System Is Authorized. Use of This System Constitutes Consent of Security Testing And Monitoring, Unauthorized Use Could Result In Criminal Prosecution."*

The wording of the login banner is very important. The word "welcome" or any synonym of it should never appear in the banner. The welcome wording can be interpreted as an open invitation for anyone to enter your system.

The login banner shall be configured in such a way that the user must physically make a keystroke to remove it from the screen. Do not allow the banner to time out.

8.3 Basic Security Module (Bsm)

If using Sun workstations, consider using the "allocate" command to restrict use of printers, storage devices, and other peripheral devices. This feature protects media that may be accessed by others. A typical scenario could be that there may be only one tape drive located down the hall and a user needs to retrieve a file from a backup. Once that tape is placed in the tape drive anyone has access to it and could download, modify, or overwrite sensitive data they are not authorized for. This could occur either intentionally or accidentally.

8.4 Attack Risks

There are various kinds of attacks you can expect to occur. Some will be malicious while others could be accidental, however the damage to your network can be great in both cases. Here are a few samples of the kinds of attack risks you, as the system administrator should be looking for.

8.5 System Security Audits

Some years ago, before the Worm raised our consciousness about security risks, it was almost laughably easy for intruders to break into almost any system. Many sites didn't use passwords at all, or offered guest or admin passwords that users could share. Users who did have their own passwords routinely chose passwords that could be easily guessed (the names of their children or pets, their birth dates, their license plates). Because nobody bothered to encrypt files, an intruder who broke into the system could then invade almost anybody's files, take a

copy of the /etc/passwd file, and later run it through a password cracking program that quickly revealed the passwords of other users in the system. Once deciphered, these purloined passwords became bartering chips among underground groups that shared technical information about product vulnerabilities and site-specific security holes.

Most systems and users have tightened up their security in the wake of the Internet Worm. Guest and admin passwords have become rarer, but password security as a whole is still laughable in most places. Group accounts abound, and invariably at least 10 percent of the passwords users select are poor (the only way to make them better is to install a password program that forces good passwords). Readily available password dictionaries, cracking programs, and password sniffing combine to make passwords very vulnerable.

How can you avoid password attacks? Educate the users on your system so they pick better passwords. Consider using system-generated passwords or, better still, stronger types of authentication, such as one-time (non-reusable) passwords.

8.6 Password Sniffing Attacks

The recent wave of password sniffing attacks on the Internet makes the strength of your passwords almost irrelevant.

How does password sniffing work? In many network setups, it is possible for any machine on a given network to hear the traffic for every machine on that network. This is true for most Ethernet-based networks, and Ethernet is by far the most common local area networking technology in use today. This characteristic of Ethernet is especially dangerous because most of the protocols in use today are unencrypted. As a result, the data sent and received is there for anybody to snoop on. This data includes files accessed via network file systems, passwords sent to remote systems during Telnet, FTP, and rlogin sessions, electronic mail sent and received, and so on.

A password sniffer is a program that takes advantage of this characteristic to monitor all of the IP (Internet Protocol) traffic on its part of the network. By capturing the first 128 bytes of every FTP or Telnet session, for example, password sniffers can easily pick up your user name and password as you type them. Password sniffers may use programs provided for network debugging as building blocks, or may be written to use the services directly. Special-purpose password sniffing toolkits are widely available to attackers.

The danger of password sniffing attacks is in their rapid spread. Favorite targets for sniffers are network providers and public access systems where the volume of Telnet and FTP connections is huge. One sniffer on large public access systems can collect thousands of sniffed account names and passwords, and then compromise every system accessed. Even if your systems are

as secure as possible and your user passwords are not exploitable, you can be infected by a packet sniffer running at any site that your users can log in from, or at any site their packets will cross to get to you.

Password sniffing can happen anywhere. Many people make the mistake of assuming that because they're using a well-known, commercial service, there is no danger in remotely accessing their own machines across the network. In fact, the commercial services are prime targets, and most of them are periodically compromised. In any case, a connection may cross a large number of intermediate networks, which each represent unknown risks. How can you avoid being sniffed? In general, you can't and still provide remote network access. If your password ever passes across a network, which might be insecure--electronically or physically--, it is likely to be captured. What you can do is ensure that an intruder who gets your password can't use it. One-time (non-reusable) passwords are probably the most effective way.

8.7 NFS and Other Data Service Attacks

A number of services exist to allow computers to share information with each other and to allow users to move easily from computer to computer. These services are an important part of the power of UNIX networks. Unfortunately, attackers, who convince these services to share more information than intended or to share it with unintended recipients, often exploit them. Often this occurs because designers were concerned with local area network access and did not realize that services might also be available across wide area networks to other organizations.

The Network File System (NFS) and Network Information Service (NIS) are notoriously easy ways to attack a system. NFS allows systems to share files over a network by letting a client mount a disk on a remote server machine. NIS maintains a distributed database of password tables, group files, host tables, and other information that systems on a network can share. Many sites choose not to support NIS at all, and some avoid even NFS. However, these services are not a problem if they are run in a protected environment (for example, behind a firewall).

If you haven't properly protected your site, an attacker may be able to simply NFS-mount your file systems. The way NFS works, client machines are allowed to read and change files stored on the server without having to log into the server or enter a password.

Because NFS doesn't log transactions, you might not even know that someone has full access to your files.

NIS is most often used to distribute password information, and most implementations of NIS provide absolutely no control over which machines can request information. As long as an attacker can guess the name of your NIS domain and can send an NIS request to your NIS

server, that attacker can get a full copy of your password information (including encrypted passwords), even if you are running shadow passwords and the passwords are not in the /etc/passwd file. The attacker is then free to crack your passwords at leisure.

NFS, NIS, and other services have additional security vulnerabilities, both obvious and not so obvious. For example, NFS has very weak client authentication, and an attacker may be able to convince the NFS server that a request is coming from a client that is permitted in the exports file (the file that lets you specify which file systems can be mounted via NFS, and which other machines can mount them). There are also situations in which an attacker can hijack an existing NFS mount. (See the discussion of hijacking attacks later in this article.)

8.8 Denial of Service Attacks

There are two classic types of denial of service attacks, both particularly devastating when used on a network. One such service attack is the "electronic mail bomb" that shuts down service by flooding an email mailbox. That's one type of denial of service--the same type performed by the Internet Worm. What happens here is that an intruder so floods a system or network--with messages, processes, or network requests--that no work can be done. The system or network spends all its time responding to messages and requests, and cannot actually satisfy any of them.

In the other category of attack, equipment or services are completely shut down or disabled. With ICMP attacks, which are becoming more common, an attacker sends an ICMP message to a host or router, telling it to stop sending packets to all or part of the network.

How can you prevent denial of service attacks? The best defense against an ICMP attack is to install a firewall that ignores or filters ICMP messages.

In general, though, denials of service attackers are tough to prevent--electronically, as well as in real life. If you accept things from the external world--electronic mail, telephone calls, or packages--it's possible to get flooded. In the electronic world, denial of service is as likely to happen by accident as on purpose. The most important thing is to set up services so that if one of them is flooded, the rest of your site keeps functioning while you fix the problem.

Fortunately, denial of service attacks is not terribly popular. They're easy enough to be unsporting; they tend to be simple to trace back and therefore risky to the attacker; and they don't provide the attacker with the information or the ability to use your computers that is the payoff for most other attacks. Intentional denial of service attacks are the work of people who are angry at your site in particular--and at most sites, there are very few such people.

8.9 IP Attacks

Attackers sometimes take advantage of a little-used option--the source routing option--in the IP header of packets being sent across the Internet. Even systems protected by firewalls have fallen victim to these types of attacks.

Certain kinds of firewalls work by keeping packets from being routed from an outside system into your internal network. In normal packet routing, packets are routed in the most efficient way from source to destination. However, if the source routing option is specified for a packet, it shows the particular route that the packet is to follow. Unfortunately, turning off the regular routing of packets from the Internet to an inside network doesn't turn off the routing of source-routed packets. Attackers have exploited this peculiarity and used it to penetrate systems that are expecting their firewalls to keep all such outside packets out.

Another attack involves attackers creating packets with false IP addresses. By exploiting applications that use authentication based on IP addresses (such as the so-called Berkeley RrS commands, which include rlogin, rsh, and rcp), intruders have been able to gain access. Most of the attacks take advantage of the ability of intruders to guess sequence numbers associated with network connections and the acknowledgements passed between machines. These attacks are technically tricky, because the intruder doesn't receive the responses to the packets it sends; when they succeed, however, the payoff for these attacks can be high.

How can you prevent these attacks? Firewalls are the only sufficient defense. You want to look for packets on your external interface (that is, packets coming from outside your internal network) that claim to have internal source IP addresses and for packets that have source routes specified. You can do this by installing an appropriately configured packet filtering router. It's also best to avoid address-based authentication completely, if you can.

8.10 Hijacking Attacks

Another emerging Internet threat involves the hijacking of any open terminal or login session from users on the system. Once intruders have root access on a system, they use a tool that lets them dynamically modify the UNIX kernel. This allows them to take over terminal connections after any authentication procedures have been completed. Even the strongest authentication (e.g., one-time passwords) are irrelevant because the attack occurs after the user successfully logs in. (This is another way that your systems can be compromised from any system that your users can log in from.)

This sort of attack has always been possible, but is easier to do and harder to detect with the new tools. Various forms of hijacking--from the completely unsubtle method of waiting for someone to get up for a cup of coffee without locking their screen, to the devious exploitation of window systems--have long been the most popular attacks at universities and other places

where people may legitimately have access and yet simultaneously be hackers. In the past, these attacks have mostly been aimed at users at the site where the attacks were taking place. The new attacks are aimed at getting from a compromised system to an otherwise uncompromisable system across the Internet.

How can you prevent this attack? Once intruders have root access, you can't. So keep them out to begin with.

8.11 Enforce Good Host Security

With host security, you enforce the security of every machine at your site separately, and you make every effort to learn about, and plug, any security holes that your particular operating system presents. Although host security isn't a complete solution to Internet risks--there are simply too many machines, vendors, and operating systems to be sure that you've successfully been able to secure them all--you need to make sure that every system on your local network is as secure as you can make it. Systems exposed directly to Internet traffic need especially strong host security.

8.12 Security Tools

Acquire and run tools to test the proper configuration of the system. These tools should be run immediately after initial installation to provide a baseline for the system, and then run periodically to determine changes in the system.

There are several tools are available via anonymous ftp from AFCERT, ASSIST, and many sites on the Internet including cert.sei.cmu.edu in the /pub/tools directory. These tools include:

Security Profile Inspector for UNIX (SPI)

Description: SPI for UNIX inspects various aspects of the computer system and reports on inconsistencies or insecure features. It consists of the following inspection modules: Access Control List, File Data Change Detector, Password Security Inspection, File Inode Change Detector, Quick System Profile, and Binary Inspector Tool.

Computer Oracle Password and Security (COPS)

Description: COPS is a suite of UNIX shell scripts, which forms an extensive security testing system. It has a password cracker, routines to check changes in setuid programs, routines to check permissions of essential systems and user files, and other routines to see whether any system software behaves in a way which could cause problems.

Crack

Description: Crack is a password cracking utility. It has an built-in networking capability, allowing the load of cracking to be spread over as many machines as are available on a given network. It is supplied with an optimized version of the UNIX crypt algorithm.

Distributed Intrusion Detection System (DIDS)

Description: Client/server intrusion detection system designed to identify and report security anomalies/ runs in a UNIX environment, which supports TCP/IP.

MD5

Description: A message-digest algorithm by RSA Security. Calculates a message-digest fingerprint (checksum) for a file. It is intended for digital signature applications where a large file must be compressed in a secure manner before being encrypted.

TCP Wrappers

Description: Monitors and logs tcp requests. Provides filtering on tcp requests.

TIS Firewalls

Description: Collection of tools used to design and implement a proxy-based firewall to enforce user defined security policy.

Tripwire

Description: Monitors a designated set of files for changes. Can notify a system administrator of corrupted or tampered files.

Wuarchive ftpd

Description: Secure ftp daemon for setting up an anonymous ftp site. Verifies checksum information to ensure retrieved copy is intact.

npasswd

Description: passwd/yppasswd replacement

skey

Description: Provides one-time password capability

There are also tools, which analyze the security posture of a system and notify system administrators of common security holes found. Two of these tools are the Internet Security Scanner (ISS) and Security Analysis Tool for Auditing Networks (SATAN). However, experienced system administrators should only use these two tools. Also, since these tools do

attempt to break into computer systems, system administrators should run these tools in performance of their jobs and with the permission and knowledge of the system being attacked.

8.13 Firewalls

A firewall restricts access from your internal network to the Internet--and vice versa. A firewall may also be used to separate two or more parts of your local network (for example, protecting finance from R&D).

The dictionary definition of "firewall" is: "A fireproof wall used as a barrier to prevent the spread of a fire." A fire may damage, or even destroy, one section of a building, but a firewall may keep that fire from spreading to other sections of the building; at the very least, it may slow down the spread until the fire can be brought under control.

On computer networks, firewalls serve an analogous purpose. A security problem somewhere on a network--for example, eavesdropping, a major break-in, or a worm program--may do a great deal of damage to one portion of the network. But if a firewall is in place, it can isolate what's behind it from the security problem. Without firewalls network security problems can rage out of control, dragging more and more systems down. Once one system on a network has been compromised, it's often trivial to compromise the others. Shared system resources, homogeneous services, and trust policies may all contribute to the spread of a security problem from one system to another.

Think of a firewall as a checkpoint; all traffic is stopped and checked at this point--usually, at the perimeter of your internal network, where you connect to the Internet. Your own site's security policy determines what happens at the checkpoint. Some requests (e.g., requests for email service) might pass right through. Others (e.g., requests for potentially dangerous service like NFS or NIS) might be turned away. Still others (e.g., requests for FTP file transfers) might be routed to proxy services, which satisfy the requests without directly exposing internal systems.

8.14 Back-Ups

Back-ups are typically not considered when implementing a security policy. However, if a system crashes or is otherwise damaged, data stored at that site will be lost forever unless it is backed-up. A recommended back-up policy is to do a “0 Level Dump” on a weekly basis and to do incremental dumps on a nightly basis. Monthly back-ups should be done and then stored off site in a fireproof container. This will quicken recovery efforts if data is lost at a sight.

8.15 Security Checklist

In summary, here is a quick checklist of topics covered so you may improve the security posture of your UNIX system:

- Install security related patches
- Make sure a "+" does not appear in the .rhost or hosts.equiv file
- Develop a good password policy with a pass-phrase, 8 characters, password aging and enforce it with "Crack"
- Use shadowed passwords
- Verify that only the only active account that has UID 0 is root
- For NIS clients change the +::0:0::: field to +:*:0:0:::
- Add a password to all new user accounts immediately; use your password policy for the original password selection (do not use username, SS#, etc)
- Do not allow or use shared accounts
- Set up restricted shells for users who need limited features
- Turn off tftp or use the secure (-s) option
- Set the EEPROM Password to a minimum of COMMAND (SunOS only)
- Place system administrators who require root privilege in the wheel group (SunOS and BSD UNIX only)
- Change the field secure to unsecure in the /etc/ttytab file (SunOS and BSD Unix only)
- Use the recommended Department of Justice login banner
- If using BSM use the allocate feature (SunOS only)
- Check for suspicious SUID or SGID programs
- Check the path of users for a period(.), if it appears make sure it appears last in the path

- Check r/w permissions on sensitive files and directories
- When using NFS do not export files to the world; use netgroups
- When using netgroups do not include the host machine in the netgroup
- When using NIS ensure that the .pag files are not world writable
- Back-up your system often and keep the tapes off site in a fireproof container
- Use security tools such as TCP Wrappers, ISS, SATAN and SPI

There are many more aspects to security on the UNIX platform, but these are the most common problems found in the field. By implementing these recommendations, your system will be more secure. It seems new vulnerabilities are found daily, so security will always be an ongoing battle. If you practice security oriented configuration management and use security tools periodically, hackers may decide your system is not worth the effort and move on to easier prey. If you have any questions regarding this handbook or have any other computer security concerns please contact your Network Security Office.

9. Appendix A: Password Selection

- Passwords must be a minimum of seven (7) characters where one is non-alpha. If the system or software does not support alpha-numeric passwords or the required length, log-in must require a "one-time" password (i.e. Cryptocard, SecurID) or multiple passwords.
- Passwords must not be words or abbreviations that can be found in any dictionary or can be easily guessed in any language.
- Passwords must not be something common and/or unique to the user such as a spouse's, child's or pet's name, birthday, address, phone number, company name, etc...
- Passwords must not be stored in plain-text format in scripts, files, or executable programs.
- Passwords will not left unsecured.
- If the same user-ID exists in production, test and development environments, the associated password must differ by at least five (5) digits.
- A user may not re-select a password once it has expired.

- Regardless of the circumstances, individual passwords (those that give access to a users' home directory or identify a specific user, must never be or revealed to anyone else besides the authorized user.
- Passwords may only be shared if there are a limited number of accounts available on a system or for administrative access (i.e. Unix root).

10. Appendix B: Daily Requirements

The following checks should be performed on a daily basis. All findings should be reported to system owners and handled appropriately. Reports should be kept for a minimum of 120 days.

- login failures
- logins from unknown hosts
- failed access to system files
- inappropriate access permissions to system files
- heavy system activity by a user after hours
- unexpected mounted file systems
- unexpected .rhosts and .netrc files
- unexpected SUID/SGID, orphan or disguised files (especially those with a privileged owner)
- successful and unsuccessful attempts to su to root
- unexpected changes in file permissions or ownership
- unexpected activity in the UUCP log file
- successful and unsuccessful attempts to use su to change to another user id
- free disk space and free inodes on local disks
- changes to the system and/or clock
- system reboots and shutdowns

- validity of /etc/passwd file
  - owned by root
  - read permission for other
  - valid password field for every account
  - only root as UID of 0
  - proper syntax
- run integrity checking tools against SUID programs and other critical system files
- ensure removal of all core files after daily backups

11. Appendix C: Weekly Requirements

The following checks should be performed on a weekly basis. All findings should be reported to system owners and handled appropriately. Reports should be kept for a minimum of six months.

- investigate all open logins (those that have not logged out overnight or for several days)
- review UUCP log files
- check for damaged log files
- check for unauthorized setuid/setgid programs
- check to ensure correct user file and directory permissions
- check for hidden files
- analyze daemon and cron failures

12. Appendix D: Monthly Requirements

The following checks should be performed on a monthly basis. All findings should be reported to system owners and handled appropriately. Reports should be kept for a minimum of one (1) year.

- check for trojan horses (check for duplicate binaries or unusual placement or programs)
- re-evaluate all trusted hosts
- change all administrator and privileged passwords
- disable all unused accounts
- verify all group memberships
- perform an account inventory to identify unauthorized accounts
- evaluate all software for version control and performance
- ensure all auditing tools are functioning as expected
- evaluate total system usage by user

13. Appendix E: Random, Frequent Checks

The following checks should be performed on a random and frequent basis. All findings should be reported to system owners immediately.

- unexpected users logged into the system
- users from unexpected hosts logged on
- users logged on at unexpected times
- normal system processes that are not running
- unexpected system processes that are running

14. Appendix F: Checklist For Removing User Accounts

Whenever a user leaves the company, transfers positions, or for any other reason is not longer authorized to use the system, the following steps should be taken.

- change all passwords that the user might have known

- change the users encrypted password entry in the password file to prevent future logins (disable access)
- change the initial program entry in the password file to be either a null one or one that records all attempted login attempts
- backup all files owned by the user and transfer them to a restricted area;
- evaluate all the user's cron jobs or other programs and remove them if necessary
- remove the user from the group file
- remove the user from any mail aliases
- remove the user from the NIS netgroup file
- do not delete the user's UID for one year
- remove and process any pending mail for the user from the mail queue
- remove any processes running with the user's UID
- retrieve any smart tokens or other access devices

15. Appendix G: Sun Solaris Patches

Security patches are available at: http://sunsolve1.sun.com for individuals with Sun contracts or ftp://ftp.uu.net/systems/sun/sun-dist for individuals without Sun contracts. They may also be retrieved at http://sunsolve.Sun.com/pub-cgi/patchpage.pl .

16. Appendix H: Valuable Mailing Lists

- Bugtraq: A comprehensive security mailing list. Send mail to listserv@netspace.org with the message body containing the words "subscribe bugtraq". An archive is available at http://www.geek-girl.com/bugtraq/index.html
- CIAC-Bulletin: CIAC Information Bulletins and Advisory Notices containing important, time-critical computer security information. Send e-mail to: majordomo@rumpole.llnl.gov . In the BODY (not subject) of the message put "subscribe ciac-bulletin".

- Sun maintains a "patch club" mailing list that includes details of Sun patch releases. Summaries are mailed weekly. The list also receives "EarlyNotifier Alerts" that contain important patch information. To change, add, or delete your email address to the list send mail to: SunSolve-EarlyNotifier@Sun.COM .

- CERT (Computer Emergency Response Team): To subscribe to the cert-advisory mailing list, send email to cert-advisory-request@cert.org . In the subject line, type SUBSCRIBE your-email-address. For example, the subject line might look like this: "Subject: SUBSCRIBE me@myplace.com"

17. Appendix I: Valuable Web Sites

- CERT (Computer Emergency Response Team): http://www.cert.org

- Rootshell: http://www.rootshell.com

- COAST-- Computer Operations, Audit, and Security Technology: http://www.cs.purdue.edu/coast/coast.html

- Infowar: http://www.infowar.com

- CIAC (Computer Incident Advisory Capability): http://ciac.llnl.gov/

- Sun Microsystems: http://www.sun.com

18. Bibliography

- The Solaris Security FAQ, by Peter Galvin; http://www.sunworld.com/sunworldonline/common/security-faq.html

- Practical UNIX and Internet Security – Simson Garfinkel and Gene Spafford