NEXTSTEP In Focus, Spring 1993 (Volume 3, Issue 2).

Staying Secure

Marc Majka

Viruses. Worms. Password-cracking software. Intruders. Lost data. Enough to give you nightmares? Here's a guide to making your computer system a safer place for you and your data. In this article, we'll examine risks that you can avoid and positive steps that you can take to make your system more secure.

WHAT YOU'RE UP AGAINST

Computer security involves establishing safeguards and procedures to prevent system and data misuse or damage. Good security also provides the means to recover from system or data security failures. In general terms, your security practices should prevent bad things from happening, and help you recover if something gets past your watchful eye.

Discussions about computer security sometimes focus on preventing break-ins or viruses, but a good security system takes into account a number of risks to your system and its data.

Nasty people

Nasty people attempt to gain unauthorized privileges or data access, or to destroy or disrupt data or the system. They might be insiders who already have some legitimate access to your system, or they might be outsiders seeking to break in. Good passwords, network access controls, and file permissions are the most important tools in defending your system against nasty people.

Nasty people with screwdrivers

A computer is only as secure as its location. If a computer can be tampered with or stolen, then so can its data. If it can be dismantled, even the best password protection mechanisms can be disabled. Hardware security includes access to network cables, which could be ªtappedº for unauthorized access. Your best protection is to place your computers in secure areas where only trusted people can access them. If that isn't possible, you can still protect your system with case locks and low-level password mechanisms.

Nasty software

Nasty software can do anything that a nasty person can do, often more quickly and easily. There are many types of nasty programs, including viruses, worms, Trojan horses, back doors, and bombs (see the sidebar). You should never let anyone install or run software from unknown or untrustworthy sources. Also, the user **root** should run only system processes and administrative software.

Friendly people

Even friendly people make mistakes. Security measures that restrict data access and system privileges help prevent accidents. Help other computer users understand security risks so that they don't inadvertently create a security problem.

Natural disasters

Unforeseen disasters can interrupt computing and cause hardware and software to be damaged or destroyed. Contingency plans and backups provide the best protection and fastest recovery.

HOW TO HANDLE IT ALL

You provide good security by managing software security mechanisms and physical access to computer hardware. The most important software mechanisms for maintaining security are access control mechanisms, including system access passwords, file permissions, and applications and tools that control network access to your computers and data. It's equally important for you to stay informed about security issues and to make sure that everyone at your site contributes to good security.

Passwords

The importance of good passwords cannot be overstated. An account without a password is an open door for unauthorized access. Fast computers and password-guessing software make short work of poorly chosen passwords. Public domain password-guessing programs are even available at many Internet public archive sites. These programs test all the words in a dictionary as passwords. The entire on-line dictionary on a NEXTSTEP system provides more than 87,000 possible passwords; a simple password testing program could check the entire dictionary in a few hours.

Good passwords should be easy to remember, but hard to guess. A great password has the following attributes:

- It contains at least six characters. You can type in passwords of any length, but only the first eight characters are actually used as your password.
- It's not a dictionary word, the name of a friend or a pet cat, a car license number, a login name spelled backwards, or anything easily associated with the user or system.
- It contains a mix of upper- and lowercase characters and punctuation. Alternate and Control characters can't be used in a password.

And keep in mind that a great password written on a bit of paper stuck on the back of a computer monitor is still not secure.

File access controls

Important files and data should be protected with good access permissions. Create UNIX user groups to enable people to share data within a group, but to protect data from outside access.

Network access controls

If your site is connected to the Internet or has dial-in access, ensure that only legitimate users can access your systems and data. Keep watch on the use of resource-sharing programs such as NFS®, to limit access to only authorized computers and users. Some sites benefit from the use of a firewallÐa single point of contact between internal and external networks that can be tightly controlled.

NetInfo access controls

NetInfo contains sensitive and important administrative information. Choose the **root** password in each NetInfo domain carefully, and give it only to administrators who need to maintain those domains. The **root** directory of any domain can contain two security-related properties: **trusted_networks** and **security_options**. The **trusted_networks** property restricts access to NetInfo data to computers with Internet addresses that you specify. The property **security_options** controls several other security-related features. The *NEXTSTEP Network and System Administration* manual describes these features in detail.

Security information

One of the best ways to stay ahead of security risks is to keep informed. Books and regular publications on computer security are listed at the end of this article. If you have access to the USENET Internet news service, you can also subscribe to the **comp.security** newsgroup for information and discussions on security issues.

Furthermore, the Internet Computer Emergency Response Team (CERT) acts as an information clearinghouse and coordinating agency for security-related information and alerts. You can contact CERT by e-mail at **CERT@cert.org**, or by telephone at (412) 268-7090.

Several other agencies are concerned with specific aspects of computer-security. You can get information about them from CERT.

A SECURITY CHECKLIST

Here's a list of specific actions you can take to help provide good security on your system. No such list can ever be complete, for it's impossible to predict all possible security threats. So, security also requires vigilance: continue to check your system for security risks and evidence of breaches to your security system.

Secure hardware

Secure hardware from being opened by unauthorized people.

Install case locks to prevent a computer from being opened or stolen. Such locks are available from a number of sources.

Secure network cables from unauthorized access.

For sites that require very high security, place network cables in conduits to prevent wiretaps.

Use hardware passwords to prevent startup from an insecure device.

Unauthorized users can gain access to a system by booting from an external disk drive, floppy disk drive, or network. Once they boot the system from an insecure source, they can modify the internal disk. Case locks are available to prevent access to external device ports and floppy drives. Some hardware vendors support hardware passwords in the boot ROM that prevent booting from a floppy disk drive. Not all hardware password systems provide the same type of security, so check that your hardware provides the security you really need.

Manage NFS file sharing

Configure NFS servers to export only to trusted systems.

Never set up an NFS server to export directories to client computers that are untrustworthy. To make server configuration easier, create a netgroup and export only to the group. Use HostManager to create and manage netgroups.

Avoid NFS setuid access unless necessary.

If possible, choose the NFSManager option that ignores setuid bits when you configure NFS clients. The setuid mechanism by itself isn't a security risk. However, if a nasty program is installed on a file server with the setuid bit set, your client systems will be safer if they ignore the setuid bit.

Watch the root account

Prevent unauthorized booting in single-user mode.

On NeXT computers, you can turn the ROM start-up program on to require a hardware password before the computer boots in UNIX single-user mode. On non-NeXT computers this password protection mechanism is not available, but you can still require a password for single-user mode. One way to do this is to change the shell startup file, **/.profile**, so that it runs a program that requires a password before the shell will start. A sample **/.profile** file and source code for a password-checking program are on the floppy disk that accompanies this issue (in the directory **/SecureSingleUser**).

Remove ª.º from root's search path.

root should run only system programs. Having a ª**.**º in **root**'s shell search path (**$path** or **$PATH**) is a security risk, since **root** could accidentally run a user's private version of a program rather than the system version.

Run only trusted software as root.

Since the **root** user account has unlimited privileges, a bomb or Trojan horse program run by **root** can do unlimited damage!

Control root passwords.

Minimize the number of people with **root** access, and change **root** passwords frequently. A nasty person with a **root** password can do extensive damage to an entire computer network.

Watch the me account

Remove the me account, or restrict its access.

The **me** account is intended for standalone computers with no requirement for security. In a networked, multi-user environment, you are best off removing the **me** account entirely. If you keep the **me** account, you should at least remove it from the **wheel** user group, which has some privileged file system access rights.

Secure the file system

Check for unauthorized setuid programs.

The security chapter of the *NEXTSTEP Network and System Administration* manual specifies how to use the **find** command to search for setuid programs.

Check for loose access permissions on system files and directories.

Restrict write access to most system files and directories to **root**. The security chapter of the *NEXTSTEP Network and System Administration* manual has more information on this topic.

Check for hidden files and directories.

Sometimes nasty software is hidden from view by having an obscure file name such as ª**...**º (dot dot dot), or ª**.** º (dot space).

Use NetInfo

Use NetInfo security options.

These options are discussed in the *NEXTSTEP Network and System Administration* manual.

Remove _writers properties, especially in /printers and /fax_modems.

The _**writers** property allows users other than **root** to modify certain NetInfo directories. If the property _**writers** makes printers and fax modems writable by all users, unauthorized users can gain **root** access to a system. The *NEXTSTEP Network and System Administration* manual describes how to locate and remove these _**writers** properties.

Watch traditional UNIX administrative files

Place passwords in /etc/passwd.

The UNIX **/etc/passwd** file is consulted when NetInfo is not running (for example, in UNIX single-user mode). This file is also consulted if your computer is a client of the Network Information Services (NIS) system. Ensure that passwords are assigned to all user accounts, and that there are no back door user accounts in this file.

Check /etc/group.

The UNIX **/etc/group** file is also consulted if NetInfo isn't running, or if your computer is an NIS client. Check this file to ensure that no user has extra file system access privileges by virtue of being in a group in this file.

Use accounting tools

Enable login and process auditing.

Login and process accounting add to security by allowing you to detect and trace unauthorized activities on your computer. The presence of an accounting system can act as a deterrent as well.

To enable login accounting, create a file named **/usr/adm/wtmp**. Then, use the **last** and **ac** commands to display login reports and summaries.

To enable process accounting, create an accounting file with the command:

```
/bin/touch /usr/adm/acct
```

Then, start the accounting program **accton** during startup with the command:

```
/usr/etc/accton /usr/adm/acct
```

You can then use the **lastcomm** and **sa** commands to print process reports and summaries.

To accomplish this all automatically, you can add something like the following lines to the system startup file, **/etc/rc.local**:

```
if [ -f /usr/etc/accton -a -f /usr/adm/acct ]; then
   /usr/etc/accton /usr/adm/acct
   (echo " accounting") > /dev/console
fi
```

Watch the workspace

Avoid setting Public Window Server and Public Sound Server.

Public Window Servers and Public Sound Servers can be exploited to defeat security. Use these settings carefully, or not at all. You can set the NetInfo **discourage_public_servers** security option to make it difficult for users to create Public Window Servers and Public Sound Servers using Preferences.

Always set Display EPS Securely.

Set this option in the UNIX Expert view in Preferences. It prevents PostScript® programs from accessing parts of the system that could be used to defeat security.

Manage network access

Require a password for automatic host addition.

Network security can be compromised if someone can add a new computer to your network. If you have enabled Automatic Host Addition using SimpleNetworkStarter or HostManager, require a password to add new hosts to the network.

Check for overly permissive .rhosts and /etc/hosts.equiv files.

The file **/etc/hosts.equiv** and the file **.rhosts** (which can appear in any user's home directory) make it more convenient to use the **rlogin** and **rsh** commands to access other systems across a network connection. These files are configured to make users ªequivalentº among systems. The effect is that **rlogin** allows a remote connection without password authentication, and programs can be started on a remote system with **rsh**.

Although convenient, this open network environment is ideal for virus and worm propagation. Check the contents of users' individual **.rhosts** files and the system-wide **/etc/hosts.equiv** file to ensure that this mechanism is used only as necessary, and that only trustworthy computers are named in these files. Remember to check **root**'s **/.rhosts** file too.

Ensure correct configuration if anonymous FTP is enabled.

If you have configured one of your computers as an anonymous FTP server, double check the configuration of the **ftp** user. The UNIX manual page for **ftpd** describes security measures that you should implement for anonymous FTP.

Install a firewall for dial-in and Internet access.

If your site is connected to the Internet, consider installing a firewall. Firewall configurations are described in *Practical UNIX Security* and other books on security. See the references later in this article.

Be careful with DOS

Enforce high DOS security standards.

If you have a computer with both NEXTSTEP and DOS partitions, or if you occasionally boot DOS from your computer's floppy disk drive, be extremely diligent about the security of your DOS computers. Any DOS program can access the NEXTSTEP partition on your disk. Since there are no ownership or privilege restrictions on DOS programs, any NEXTSTEP program or data file can be read, modified, or destroyed from within DOS. Although there are no known DOS viruses that attack NEXTSTEP yet, some DOS viruses contain bombs that will destroy all disk data, including NEXTSTEP data.

DOS programs running in the SoftPC environment don't pose a high security risk to NEXTSTEP. SoftPC restricts program privileges and UNIX file system access rights so that a DOS program running under SoftPC has no more privileges than any other user program.

SUMMARY

Computer security is a broad subject, and this article presents only the basics. Security involves planning, analysis, regular inspection and re-evaluation, contingency plans and facilities, and backups. If you follow the guidelines in this article and keep informed about security issues, you can keep your system safe from common security threats.

If your site requires extremely high security, then it's time to take a trip to the bookstore to learn more. If you're managing a site that's part of a larger governmental or industrial organization, there may be security agencies that exist to help you. You can also contact CERT for more information.

FOR MORE INFORMATION

To learn more about making your computer or network secure, check out these books:

Denning, Peter J., ed. *Computers Under Attack: Intruders, Worms, and Viruses*. New York: Addison-Wesley Publishing/ACM Press, 1990.

Garfinkel, Simson, and Gene Spafford. *Practical UNIX Security*. Sebastopol, CA: O'Reilly & Associates, Inc., 1991.

Gasser, Morrie. *Building a Secure Computer System*. New York: Van Nostrand Reinhold, 1988.

Nemeth, Evi, Garth Snyder, and Scott SeeBass. *UNIX System Administration Handbook*. Englewood Cliffs, NJ: Prentice Hall, 1989.

The following organizations also provide excellent information on security issues:

Computer Emergency Response Team (CERT)
E-mail: **CERT@cert.org**
Phone: (412) 268-7090

Computer Security Institute
600 Harrison Street
San Francisco, CA 94107
Phone: (415) 267-7666

Forum of Incident Response and Security Teams (FIRST)
E-mail: **first-sec@first.org**
Phone: (301) 975-5200

*Marc Majka is a Trainer in NeXT Education. You can reach him at **Marc_Majka@next.com**.*

A FIELD GUIDE TO NASTY SOFTWARE

Virus

A virus is a program that installs itself into the software that normally runs on a computer system, and spreads itself to other systems using network or disk data transfer. Viruses rely on being difficult to detect: They generally add extra instructions to an existing program. The program continues to function, but the extra instructions install new copies of the virus in other programs. A ªpureº virus copies itself from place to place, but most viruses include a bomb, back door, or Trojan horse.

Viruses have long been an enormous problem for primitive operating systems. These systems don't have mechanisms to restrict privileges or data access rights to running programs. Once a virus gets into one program, there's nothing to prevent it from copying itself anywhere, including startup and system programs. For example, programs running on a DOS system effectively ªownº the entire computer while they run, as well as all data stored on disks, making them extremely vulnerable to virus attacks.

Fortunately, programs running on NEXTSTEP are restricted by UNIX system protection mechanisms. A program has access only to files belonging to the user running the program. A virus in this environment can't spread itself to system programs and programs owned by other users. If a virus tries to disrupt data or the system, it can only affect one user. Only the **root** user has system-wide permissions, so be extremely careful about installing or running software when you are logging in as **root**.

Although viruses are possible in NEXTSTEP, we aren't aware of any NEXTSTEP-specific virus, and are aware of only a few UNIX viruses created by virus researchers and tested in controlled environments. A related kind of nasty program called a worm was introduced on the Internet several years ago, but the worm didn't spread a virus to any UNIX systems. At this time, there are no NEXTSTEP or UNIX ªanti-viralº programs, since there are no viruses!

Worm

A worm is a program or set of cooperating programs that propagates itself from computer to computer, often using network security weaknesses. Unlike a virus, a worm doesn't infect existing software; it is a separate program. Once it starts running, it tries to spread itself to still more systems. Unlike viruses, worms are not persistentÐa worm is killed by removing the worm program and rebooting.

A UNIX worm propagated through large parts of the Internet in 1988. This has been the only widespread worm observed in a UNIX environment. The worm didn't contain any destructive code, but it did disrupt computing by placing a large processor load on affected systems, slowing them down. Further disruptions were caused when several important Internet sites disconnected their Internet access to prevent re-infection.

The Internet worm propagated by a number of means. It exploited security holes in some common UNIX network services, which have since been removed by all leading UNIX system vendors. It also exploited security weaknesses (like poorly chosen passwords) introduced by users and administrators on some systems.

Trojan horse

The citizens of Troy were happy to install a lovely statue of a horse in their city, but were rudely surprised when it turned out to be more than it appeared! Similarly, a software Trojan horse is a program that appears to be legitimate, but contains malicious instructions.

A Trojan horse can contain a bomb or a back door. It may also be used as an illicit data-gathering tool. For example, a Trojan horse can appear to be a login program such as **loginwindow**. Such a program might act just like the normal **loginwindow**, but instead collect the password of each user who logs in, e-mailing it to the author of the Trojan horse program.

Since Trojan horses are not what they appear to be, they are often difficult to detectÐuntil the Greek soldiers climb out! You can defend yourself against them by installing only trustworthy programs. Be very suspicious of programs that ask for passwords and any software that must be installed with system privileges. You can also compare your current programs with the original versions on your installation media to detect tampering. You can use the **showmods** program to detect changes made to system files since NEXTSTEP was installed. (Since **showmods** checks your files against a ªbill of materialsº or BOM file, make a preliminary check to ensure that the BOM file is valid. Check against the BOM file on your CD-ROM, or save an original copy from your system on a protected floppy disk.)

Bomb

A software bomb is a malicious program triggered by some event, often as simple as the system clock reaching a certain time. Bombs can take many different forms. Some are relatively benign, just displaying a message, while others cause damage.

Bombs have no way to propagate themselves, although viruses often carry bombs along. A ªpureº bomb must be run by a user before it ªexplodes.º A nasty user with unauthorized access to another user's account or to system files can plant a bomb that starts automatically.

Since bombs are simple programs, they are easy to create. Fortunately, UNIX system permissions prevent them from having wide ranging effects on a NEXTSTEP system. A bomb can affect only files owned by the user who runs the bomb program. Good passwords and file system permissions make it difficult to plant bombs. Never run untrustworthy and untested software that might contain a bomb.

Back door

A back door allows a nasty person to gain unauthorized system access, bypassing normal access control mechanisms. There are many kinds of back doors, although they often take the form of modified system programs (Trojan horses) or user accounts with extra privileges. Some examples:

- A modified **loginwindow** or **su** program that allows users to log in with either their own passwords or a special extra password

- A modified system utility that contains a secret command allowing extra privileges
- A user account with the UID number set to ª0º, giving the user **root** access to the system
- A private copy of **/bin/sh** that has been marked setuid with owner **root**, so it runs with **root** privileges
- A special case in a system startup routine that allows a system to boot without normal protections in place, or with a special **root** password

Since some back doors take the form of Trojan horses, be on the lookout for system modifications. Also be alert for private copies of setuid programs and privileged user accounts.ÐMM