Network Working Group A. Gwinn
INTERNET-DRAFT Networld+Interop NOC Team
Obsoletes: None April 1997
Category: Informational

Network Security For Large Trade Shows
<draft-rfced-info-gwinn-00.txt>

Status of this Memo

Abstract

This document is designed to assist vendors and other participants in large trade shows, such as Networld+Interop, in designing effective protection against network and system attacks by unauthorized individuals. Generally, it has been observed that many system administrators and trade show coordinators tend to overlook the importance of system security at trade shows. In fact, systems at trade shows are just as prone to attack as office-based platforms. Trade show systems should be treated as seriously as an office computer. A breach of security of a trade show system can render (and has rendered) a vendor's demonstrations inoperable--quite possibly for the entire show!

This dcoument is not intended to replace the multitudes of comprehensive books on the subject of Internet security. Rather, its purpose is to provide a checklist-style collection of frequently overlooked, simple ways to minimize the chance of a costly attack. Vendors are encouraged to pay special attention to this document and share it with all associated representatives.

Physical Security

Before addressing the technical, one of the most frequently underrated (and overlooked) security breaches is the simple low-tech attack. The common victim is the one who leaves a console logged in, perhaps as root, and walks away. Other times, an anonymous "helpful soul" might ask for a password in order to assist the user in "identifying a problem." This type of method allows an intruder, especially one logged in as "root", access to system files.

Tips:

* Educate sales and support staff as to the importance of keeping an eye on logged-in systems--especially "root" or other privileged accounts.
* Identify individuals who are not using exhibit systems for their intended purpose (i.e. playing "Quake" or "Doom" on one of your workstations).
* Request identification from anyone wishing to access systems for maintenance purposes unless they are known personally.

System Security

This section discusses technical security procedures for workstations on the vendor network. Although primarily aimed at Unix systems, many general procedures can be applied to other platforms.

Password Security

Lack of passwords or easy to guess passwords are a relatively low-tech door into systems, but are responsible for a significant number of breakins. Good passwords are a cornerstone of system security.

Tips:

* Check /etc/passwd for lack of passwords. Some vendors ship systems with null passwords, in some cases even in root accounts.
* Change passwords, especially system passwords.
* Mix case, numbers and punctuation especially on root passwords.
* Change system passwords on a regular basis.

Extra "root" Accounts

Some system vendors have been known to ship systems with accounts, other than root, that have root privileges (UID=0). For example, some

vendors may include a separate system administration account that places a user in a specific administrative program. If a system does not need additional root accounts, these can be disabled by placing "*" in the password field of /etc/passwd. Check all systems for extra "root" accounts and either disable them or change their password as

appropriate.

Use of Authentication Tokens

Authentication tokens such as SecureID, Cryptocard, DES Gold and others, provide a method of producing "one-use" passwords for specific access. The advantages are obvious. Remember that there are many packet sniffers and other administration tools constantly watching the network-especially at a large network-oriented trade show. Typed passwords, by default, are sent clear text across the network, allowing others to view them. Authentication tokens provide a password that is only valid for that one instance, and are useless after they are used. A logical extension of the use of authentication tokens would be to use them for "return trips home" (show network back to a home site) to minimize the chance of off-site security problems.

Tips:

* Contact vendors of authentication tokens/cards for further information as to how to integrate into specific environments, or on to specific platforms.
* The public-domain utility "cryptosu" (csu), when used with a Cryptocard, provides a replacement for Unix's "su" command, employing a challenge/response style of authentication for root access.

Anonymous FTP

Anonymous FTP accounts can easily turn into a security hole. The simplest rule-of-thumb to follow is to disable this service if it is not specifically needed. However, if anonymous FTP is to be used, the following tips may provide assistance into securing it.

* When a user logs in as "anonymous", they should be locked into a specific directory tree. Be sure that it properly chroots to the

appropriate directory. A "cd /" should put an anonymous user at the top of a tree.
* Some systems may allow symbolic links to take a user outside the allowed tree. Verify all links inside the anonymous hierarchy.
* Make sure that ftp's root directory is owned by someone other than the 'ftp' account. Typically, it should be owned by "root".
* Examine the need for a world-writeable incoming directory. Many sites use these as a way for outside users to transfer files into the site. This, however, can turn into an archive (and frequently does) for stolen software. Removing the "read" bit from the directory permissions (chmod 733) prohibits an anonymous user from being able to list the contents of a directory. Files can be

deposited as usual, but not retrieved unless the exact name of the file is known.

NFS Exports

Exporting an writeable NFS filesystem to the world grants anyone the ability to read and write any file in the exported mount point. If this is done, for example, with a system directory such as "/" or "/etc", it is a simple matter to edit password files to create one-self access to a system. Therefore, /etc/exports should be closely examined to be certain that nothing of a sensitive nature is exported to anyone but another trusted host. Anything exported to the general public should be exported "read-only".

Trusted Hosts

Trusted host entries are a method for allowing other hosts "equivalent" security access to another host computer. Some vendors ship systems with open trusted host files. This should be addressed.

Tips:

* Check for a '+' entry (all systems trusted) in /etc/hosts.equiv and all ".rhosts" files (there may be multiple .rhosts files) and remove it.
* Check for an "xhost +" entry in the "...X11/xdm/Xsession" file. Most often, an "xhost" entry will appear with a pathname such as "/usr/local/lib/xhost +". This should be disabled.

SetUID and SetGID binaries

The "suid" bit on a system executable program allows the program to execute as the owner. A program that is setUID to "root" will allow the program to execute with root privileges. There are multiple legitimate reasons for a program to have root privileges, and many do. However, it may be unusual to have suid programs in user directories, or other non-system places. A scan of the filesystems can turn up any program with its suid or sgid bit set. Before disabling any programs, however, it is strongly suggested that the legitimacy be confirmed.

Tips:

* "find / -user root -perm -4000 -print" will find any occurrence of a setuid file anywhere in the system, including those on NFS mounted partitions.
* "find / -group kmem -perm -2000 -print" will do the same for kmem group permissions.

System Directory Ownership and Write Permissions

Check ownership of all system directories and permissions needed to write or modify files. A directory with permissions such as "drwxrwxrwx" (such as /tmp) is world-writeable and anyone can create or modify files in such area. Pay special attention to "/" and "/etc". These should be owned by some system account-not by an individual user. If in doubt, contact the vendor of the system software for confirmation of these settings.

Network Services in /etc/inetd

Any servers not needed should be disabled. The notorious "R services" (rexec, rsh, and rlogin) are particularly prone to security problems and should be disabled unless specifically needed. If "R services" are required, pay particular attention to trusted hosts, and be aware of the risk of IP spoofing attacks from machines "pretending" to be trusted hosts.

Tips:

* Comment out "R services" (rexec, rsh, rlogin) in /etc/inetd.

* Check for unknown or unneeded services.

Trivial File Transfer Protocol (TFTP)

TFTP can be an easy way for an intruder to access system files. A good practice is to disable TFTP if it is not needed. If it is needed, check to see that sensitive files are not accessible. Attempting to tftp files such as /etc/passwd or /etc/motd will verify accessiblity of a system from the outside.

TCP Connection Montoring

Public domain software (TCP Wrappers or "tcpd") allows TCP connections to be restricted and monitored on a host by host basis. This software can be configured to notify an administrator (as well as syslog) attempts to access the host by unauthorized parties. This software is available from:

ftp://info.cert.org/pub/tools/tcp_wrappers/

BIND (Berkeley Internet Name Daemon)

Earlier versions of BIND have been prone to various attacks. If a host is going to be acting as DNS, the latest version of BIND should be used. It can be downloaded from:

ftp://ftp.isc.org/isc/bind

Sendmail and Mailer Security

A great number of previous versions of Sendmail have known security holes. All Sendmail versions should be checked for the most recent version. Alternatively, operating system vendors should be consulted for their most recent release.

Web Server cgi-bin Security

All server cgi-bin scripts and binaries should be checked (especially the "...httpd/cgi-bin" directory) for those that allow shell commands to be executed. Many attacks, of recent months, have centered around the use of utilities such as "phf" for accessing /etc/passwd on a

target system. Any cgi that is not needed in the course of operation of a web server should be removed.

Other Suggestions

* Check with the vendor of operating systems for known security issues. Make certain that all systems have the latest version of the software as well as any security patches to fix specific problems.

* Examine log files on the host frequently. The "last" command will furnish information on recent logins and where they came from. The "syslogs" will contain more specific information on system events. Web server logfiles (...httpd/log/access_log and ...httpd/log/error_log) will contain information on who has been accessing a WWW server, what has been accessed, and what has failed.

* Good backups are the best defense against system damage. A rule-of-thumb is to "back information up when it can't afford to be lost".

General Network Security

As would be expected at trade shows (large or otherwise), there are many entities running packet sniffers. Most are vendors who have a legitimate need to run them during the course of product demonstrations. A caveat to this is that there are many "listening ears" on network segments-any of whom can "hear" or "see" information as it crosses the net. Particularly prone to eavesdropping are telnet sessions. A good rule of thumb is to assume that "when you type your password, the only one that doesn't see it is you!"

It is a good practice to not log in (or "su") to an account with root privileges, across the network if at all possible. As mentioned previously, authentication tokens are a simple way to add security to system account access.

Packet Filtering

Many routers support basic packet filtering. Below is listed a good

"general" packet filter approach. The approach itself is ordered into categories:

* General global denials/acceptance.

* Specific global service denials.

* Specific service acceptance.

* Final denial of all other TCP/UDP services.

Based on this theory, a good approach to a filter ruleset, in order of execution priority, might be:

General Global Denials/Acceptance

1 Filter spoofed source addresses by interface. Match source addresses to routing information available for the interface. Discard packets with source addresses arriving on one interface (from the "outside" for example) claiming a source address on another interface (the "inside").
2 Filter all source routed packets unless source routing is specifically needed.
3 Allow outbound connections from "inside" hosts.
4 Allow established TCP connections (protocol field contains 6 and the TCP flags field either contains ACK or does NOT contain SYN bit). Only filter requests for 'new' connections.
5 Filter 'new' connections with source port of 25. Prevents people from pretending to be a remote mail server.
6 Filter loopback address (source address 127.0.0.1). Prevents packets resulting from misconfigured DNS resolver.

Specific Global Service Denials

1 If not required, specifically block all "R-command" ports (destination ports 512-515).
2 Block telnet (destination port 23) from any host not requiring telnet access from the outside.
3 Add specific filters to deny other specific protocols to the network, as needed.

Specific Host/Service Acceptance

1 Add general open access, if desired, to specific "public" hosts (unsecure FTP or WWW servers).
2 Allow SMTP (source and destination port 25) for electronic mail.
3 Allow inbound FTP connections (source port 20).
4 Allow DNS (source and destination port 53, UDP & TCP). If zone transfers are not needed, block the TCP ports.
5 Allow RIP packets in (source and destination port 520, UDP).
6 Add specific filters to allow other desired specific protocols and/or open certain ports to specific machines.

Final Service Denial

1 Deny all other UDP and TCP services not addressed in the previous filters.

Author's Address

R. Allen Gwinn, Jr.
Associate Director, Computing
Business Information Center
Southern Methodist University
Dallas, TX 75275

Phone: 214/768-3186
EMail: allen@mail.cox.smu.edu or allen@radio.net

Expire in six months