A little over a year ago, the SANS Institute and the National Infrastructure Protection Center (NIPC) released a document summarizing the Ten Most Critical Internet Security Vulnerabilities. Thousands of organizations used that list to prioritize their efforts so they could close the most dangerous holes first. This new list, released on October 1, 2001, updates and expands the Top Ten list. With this new release, we have increased the list to the Top Twenty vulnerabilities, and we have segmented it into three categories: General Vulnerabilities, Windows Vulnerabilities, and Unix Vulnerabilities.

The SANS/FBI Top Twenty list is valuable because the majority of successful attacks on computer systems via the Internet can be traced to exploitation of security flaws on this list. For instance, system compromises in the Solar Sunrise Pentagon hacking incident and the easy and rapid spread of the Code Red and NIMDA worms can be traced to exploitation of unpatched vulnerabilities on this list.

These few software vulnerabilities account for the majority of successful attacks, simply because attackers are opportunistic – taking the easiest and most convenient route. They exploit the best-known flaws with the most effective and widely available attack tools. They count on organizations not fixing the problems, and they often attack indiscriminately, scanning the Internet for any vulnerable systems.

In the past, system administrators reported that they had not corrected many of these flaws because they simply did not know which vulnerabilities were most dangerous, and they were too busy to correct them all. Some vulnerability scanners search for 300 or 500 or even 800 vulnerabilities, thus blunting the focus your system administrators need to ensure that all systems are protected against the most common attacks. The Top Twenty list is designed to help alleviate that problem by combining the knowledge of dozens of leading security experts from the most security-conscious federal agencies, the leading security software vendors and consulting firms, the top university-based security programs, and CERT/CC and the SANS Institute. A list of participants may be found at the end of this document.

We welcome your comments and feedback.

The SANS Institute

Five Notes For Readers:

Note 1. Updates
The SANS/FBI Top Twenty is a living document. It includes step-by-step instructions and pointers to additional information useful for correcting the flaws. We will update the list and the instructions as more critical threats and more current or convenient methods are identified, and we welcome your input along the way. This is a community consensus document – your experience in fighting attackers and in eliminating the vulnerabilities can help others who come after you. Send suggestions via e-mail to info@sans.org with the subject Top Twenty Comments.

Note 2. CVE Numbers
You'll find references to CVE (Common Vulnerabilities and Exposures) numbers accompanying each vulnerability. You may also see CAN numbers. CAN numbers are candidates for CVE entries that are not yet fully verified. For more data on the award-winning CVE project, see http://cve.mitre.org. In the General Vulnerabilities section, the CVE numbers listed are examples of some of the vulnerabilities that are covered by each listed item. Those CVE lists are not meant to be all-inclusive. However, for the Windows and Unix Vulnerabilities, the CVE numbers reflect the top priority vulnerabilities that should be checked for each item.

Note 3. Ports To Block At The Firewall
At the end of the document, you'll find an extra section offering a list of the ports used by commonly probed and attacked services. By blocking traffic to these ports at the firewall or other network perimeter protection device, you add an extra layer of defense that helps protect you from configuration mistakes. Note, however, that using a firewall to block network traffic directed to a port does not protect the port from disgruntled coworkers who are already inside your perimeter or from hackers who may have penetrated your perimeter using other means.

Note 4. Automated Scanning for the Top Twenty
Manual methods for checking a system to see whether it has each of the listed vulnerabilities are presented in this document. A more practical approach to finding the UNIX and Windows vulnerabilities – especially if you practice safe computing by checking every new system before you attach it to the Internet, and rechecking all your systems frequently – is to use an automated scanner. Bob Todd, the author of the free Internet scanner SARA, has created a special version of SARA designed specifically to find and report on the status of vulnerabilities on the SANS/FBI Top Twenty list. The Top 20 Scanner can be downloaded from the Center for Internet Security's website at www.cisecurity.org. Several commercial vulnerability scanners may also be used to scan for these vulnerabilities, and the SANS Institute will maintain a list of all scanners that provide a focused Top Twenty scanning function, at www.sans.org.

Note 5. Links to the ICAT Vulnerability Index
Each CVE vulnerability reference is linked to the associated vulnerability entry in the National Institute of Standards and Technology's ICAT vulnerability indexing service (http://icat.nist.gov). ICAT provides a short description of each vulnerability, a list of the characteristics of each vulnerability (e.g. associated attack range and damage potential), a list of the vulnerable software names and version numbers, and links to vulnerability advisory and patch information.

Top Vulnerabilities That Affect All Systems (G)

G1 - Default installs of operating systems and applications

G1.1 Description:

Most software, including operating systems and applications, comes with installation scripts or installation programs. The goal of these installation programs is to get the systems installed as quickly as possible, with the most useful functions enabled, with the least amount of work being performed by the administrator. To accomplish this goal, the scripts typically install more components than most users need. The vendor philosophy is that it is better to enable functions that are not needed, than to make the user install additional functions when they are needed. This approach, although convenient for the user, creates many of the most dangerous security vulnerabilities because users do not actively maintain and patch software components they don't use. Furthermore, many users fail to realize what is actually installed, leaving dangerous samples on a system simply because users do not know they are there.

Those unpatched services provide paths for attackers to take over computers.

For operating systems, default installations nearly always include extraneous services and corresponding open ports. Attackers break into systems via these ports. In most cases the fewer ports you have open, the fewer avenues an attacker can use to compromise your network. For applications, default installations usually include unneeded sample programs or

scripts. One of the most serious vulnerabilities with web servers is sample scripts; attackers use these scripts to compromise the system or gain information about it. In most cases, the system administrator whose system is compromised did not realize that the sample scripts were installed. Sample scripts are a problem because they usually do not go through the same quality control process as other software. In fact they are shockingly poorly written in many cases. Error checking is often forgotten and the sample scripts offer a fertile ground for buffer overflow attacks.

G1.2 Systems impacted:

Most operating systems and applications. Keep in mind that almost all third-party web server extensions come with sample files, many of which are extremely dangerous.

G1.3 CVE entries:

(Note: This list is not complete or all-inclusive. It is a sample of some of the vulnerabilities covered by this category.)

CVE-1999-0415, CVE-1999-0678, CVE-1999-0707, CVE-1999-0722, CVE-1999-0746, CVE-1999-0954, CVE-2000-0112, CVE-2000-0192, CVE-2000-0193, CVE-2000-0217, CVE-2000-0234, CVE-2000-0283, CVE-2000-0611, CVE-2000-0639, CVE-2000-0672, CVE-2000-0762, CVE-2000-0868, CVE-2000-0869, CVE-2000-1059

G1.4 How to determine if you are vulnerable:

If you have ever used an installation program to install system or service software (as nearly every company has), and you have not removed unnecessary services and installed all security patches, then your computer system is vulnerable to hacker attack.

Even if you did perform additional configuration steps, you could still be vulnerable. You should run a port scanner and a vulnerability scanner against any system that is to be connected to the Internet. When analyzing the results, keep in mind the principle that your systems should run the smallest number of services and software packages needed to perform the tasks required of your system. Every extra program or service provides a tool for attackers – especially because most system administrators do not patch services or programs that they are not actively using.

G1.5 How to protect against it:

Remove unnecessary software, turn off unneeded services, and close extraneous ports. This can be a tedious and time-consuming task. For this reason, many large organizations have developed standard installation guidelines for all operating systems and applications used by the organization. These guidelines include installation of only the minimal features needed for the system to function effectively.

The Center for Internet Security (CIS) has developed a consensus benchmark for minimum security configuration of Solaris and Windows 2000, based on the combined experience and knowledge of more than 170 organizations from a dozen countries (see www.cisecurity.org). Benchmarks and testing tools for other operating systems are in process. The CIS tools can be used to test the level of security and compare the security status of systems across divisions. The CIS guidelines can be used to improve the security of most operating systems.

G2 - Accounts with No Passwords or Weak Passwords

G2.1 Description:

Most systems are configured to use passwords as the first, and only, line of defense. User IDs are fairly easy to acquire, and most companies have dial-up access that bypasses the firewall. Therefore, if an attacker can determine an account name and password, he or she can log on to the network. Easy to guess passwords and default passwords are a big problem; but an even bigger one is accounts with no passwords at all. In practice all accounts with weak passwords, default passwords, and no passwords should be removed from your system.

In addition, many systems have built-in or default accounts. These accounts usually have the same password across installations of the software. Attackers commonly look for these accounts, because they are well known to the attacker community. Therefore, any default or built-in accounts also need to be identified and removed from the system.

G2.2 Systems impacted:

Any operating system or application where users authenticate via a user ID and password.

G2.3 CVE entries:

(Note: This list is not complete or all-inclusive. It is a sample of some of the vulnerabilities covered by this category.)

CVE-1999-0291, CAN-1999-0501, CAN-1999-0502, CAN-1999-0503, CAN-1999-0505, CAN-1999-0506, CAN-1999-0507, CAN-1999-0508, CAN-1999-0516, CAN-1999-0517, CAN-1999-0518, CAN-1999-0519

G2.4 How to determine if you are vulnerable:

In order to know if you are vulnerable, you need to know what accounts are on your system. The following are the steps that should be performed:

1. Audit the accounts on your systems and create a master list. Do not forget to check passwords on systems like routers and Internet-connected digital printers, copiers and printer controllers.
2. Develop procedures for adding authorized accounts to the list, and for removing accounts when they are no longer in use.
3. Validate the list on a regular basis to make sure no new accounts have been added and that unused accounts have been removed.
4. Run a password cracking tool against the accounts looking for weak or no passwords. (Make sure you have official written permission before employing a password cracking tool.)
    a. LC3 – Microsoft Windows NT and Microsoft Windows 2000, http://www.atstake.com
    b. Microsoft Personal Security Advisor, – Microsoft Windows NT and Microsoft Windows 2000, www.microsoft.com/security/mpsa
    c. John the Ripper – Unix, http://www.openwall.com/john
    d. Pandora – Novell, http://www.nmrc.org/pandora
5. Have rigid procedures for removing accounts when employees or contractors leave, or when the accounts are no longer required.

G2.5 How to protect against it:

To eliminate these password problems, two steps need to be performed. In the first step all accounts with no password are given a password or are removed, and weak passwords are strengthened. Sadly, when users are asked to change and strengthen their passwords, they often pick another one that is easy-to-guess. This brings us to the second step. User passwords should also be validated when they change their password. Computer programs are available to reject any password change that does not meet your security policy. The most popular are described at the urls below:

1a. For UNIX: Npasswd (SunOS 4/5, Digital Unix, HP/UX, and AIX)
http://www.utexas.edu/cc/unix/software/npasswd

1b. For Unix: Cracklib" and associated PAM modules (Linux)

2. For Windows NT: Passfilt, http://support.microsoft.com/support/kb/articles/Q161/9/90.asp

These programs ensure that when passwords are modified, they will be of the length and composition required to make guessing and cracking difficult. Note that many vendor Unix systems include internal support for password hardening, and that there are other packages available as well.

Many organizations supplement password control programs with controls that ensure that passwords are changed regularly, and that old passwords are not reused. If password aging is used, make sure that the users are given warning and chances to change their password before it expires. When faced with the message: "your password has expired and must be changed," users will tend to pick a bad password.

Microsoft Windows 2000 includes password constraint options in Group Policy. An administrator can configure the network such that user passwords must have a minimum length, a minimum and maximum age, and other constraints. It is important to require a minimum age on a password. Without it, users tend to change their password when required and then immediately change them back. Requiring minimum ages on passwords make users remember the passwords and makes them less likely to change them back.

Another important supplement is user awareness training that helps users understand why and how to pick strong passwords. The most common advice given for picking better passwords is to pick a phrase or line from a song that includes a number, and construct the password from the first or second letter of each non-numeric word in the phrase, and the numeral for any numbers. Adding punctuation makes the password even more difficult to crack.

Another way to protect against no passwords or weak passwords is to use an alternative form of authentication such as password-generating tokens or biometrics. If you are having trouble with weak passwords, use an alternative means of authenticating users.

G3 - Non-existent or Incomplete Backups

G3.1 Description:

When an incident occurs (and it will occur in nearly every organization), recovery from the incident requires up-to-date backups and proven methods of restoring the data. Some organizations make daily backups, but never verify that the backups are actually working. Others construct backup policies and procedures, but do not create restoration policies and procedures. Such errors are often discovered after a hacker has entered systems and destroyed or otherwise ruined data.

A second problem involving backups is insufficient physical protection of the backup medium. The backups contain the same sensitive information that is residing on the server, and should be protected in the same manner.

G3.2 Systems impacted:

Any mission critical system.

G3.3 CVE entries:

N/A

G3.4 How to determine if you are vulnerable:

An inventory of all critical systems must be identified. Then a risk analysis should be performed identifying what the risk and corresponding threat is for each critical system. The backup policies and procedures should clearly map to these key servers. Once these systems have been verified, the following should be validated:

1. Are there backup procedures for those systems?
2. Is the backup interval acceptable?
3. Are those systems being backed up according to the procedures?
4. Has the backup media been verified to make sure the data is being backed up accurately?
5. Is the backup media properly protected in-house and with off-site storage?
6. Are there copies of the operating system and any restoration utilities stored off-site (including necessary license keys)?
7. Have restoration procedures been validated and tested?

G3.5 How to protect against it:

Backups must be made at least daily. The minimum requirement in most organizations is to perform a full backup weekly and incremental backups every day. At least once a month the backup media should be verified by doing a restore to a test server to see that the data is actually being backed up accurately. This is the minimum requirement. Some companies perform full backups every day or backups multiple times a day. The ultimate backup solution is a fully redundant network with fail-over capability – a solution required for critical real-time financial and e-commerce systems, systems controlling the critical infrastructure, and some Department of Defense systems.

G4 - Large number of open ports

G4.1 Description:

Both legitimate users and attackers connect to systems via open ports. The more ports that are open the more possible ways that someone can connect to your system. Therefore, it is important to keep the least number of ports open on a system necessary for it to function properly. All other ports must be closed.

G4.2 Systems impacted:

Most operating systems.

G4.3 CVE entries:

(Note: This list is not complete or all-inclusive. It is a sample of some of the vulnerabilities covered by this category.)

CVE-1999-0189, CVE-1999-0288, CVE-1999-0351, CVE-1999-0416, CVE-1999-0675, CVE-1999-0772, CVE-1999-0903, CVE-2000-0070, CVE-2000-0179, CVE-2000-0339, CVE-2000-0453, CVE-2000-0532, CVE-2000-0558, CVE-2000-0783, CVE-2000-0983

G4.4 How to determine if you are vulnerable:

The netstat command can be run locally to determine which ports are open, but the best way to have confidence in the scans is to run an external port scanner against your systems. This will give you a list of all ports that are actually listening. If the results of netstat differ from the port scanning results, you should investigate why. Once the two lists agree, go through the list and validate why each port is open, and what is running on each port. Any port that cannot be validated or justified should be closed. The final list should be recorded and used to audit the ports on a regular basis to make sure no extraneous ports appear.

Among the many port scanners, the most popular is nmap. The Unix version of nmap can be found at: http://www.insecure.org/nmap/. This version will also compile on NT systems. The NT version of nmap can be found at: http://www.eeye.com/html/Research/Tools/nmapnt.html. Other port scanners also work well. Whatever scanner you use, you MUST scan both TCP and UDP ports over the entire range: 1-65,535.

You should always have written permission before performing comprehensive port scanning on systems within an organization. Some operating systems, and particularly devices with embedded TCP/IP stacks, can exhibit unpredictable behavior when scanned. Scanning may also trigger internal intrusion detection systems or firewalls, and may be interpreted as an attack if proper notice is not given.

G4.5 How to protect against it:

Once you have determined which ports are open, your task is to identify the minimal subset of ports that must remain open for your system to function effectively—then close all other ports. To close a port, find the corresponding service and turn it off/remove it.

On Unix systems, many services are controlled by inted and its corresponding configuration file, inetd.conf. Inetd.conf lists the services listening on a given port, and it can often be used to close ports. Removing a service from inetd.conf, then restarting inted, stops the port from being opened. Other services are started via scripts run at boot time (such as /etc/rc,

/etc/rc.local, or the scripts found in the /etc/rc* directories). Consult your system's documentation on how to disable these scripts, as the details vary between Unix versions.Also, a program called lsof can be used to audit open ports on Unix systems. Lsof can be downloaded
from – ftp://vic.cc.purdue.edu/pub/tools/UNIX/lsof/lsof.tar.gz

For Windows NT and Windows 2000, a program called fport from www.foundstone.com can be used to try to determine what service/program is listening on a certain port. In Windows XP, you can determine which program is listening on a port by running the netstat command with the –o switch. This information will allow you to turn off the service, which will close the port.

G5 – Not filtering packets for correct incoming and outgoing addresses

G5.1 Description:

Spoofing IP addresses is a common method used by attackers to hide their tracks when they attack a victim. For example, the very popular smurf attack uses a feature of routers to send a stream of packets to thousands of machines. Each packet contains a spoofed source address of a victim. The computers to which the spoofed packets are sent flood the victim's computer often shutting down the computer or the network. Performing filtering on traffic coming into your network (ingress filtering) and going out (egress filtering) can help provide a high level of protection. The filtering rules are as follows:

1. Any packet coming into your network must not have a source address of your internal network
2. Any packet coming into your network must have a destination address of your internal network
3. Any packet leaving your network must have a source address of your internal network
4. Any packet leaving your network must not have a destination address of your internal network.
5. Any packet coming into your network or leaving your network must not have a source or destination address of a private address or an address listed in RFC1918 reserved space. These include 10.x.x.x/8, 172.16.x.x/12 or 192.168.x.x/16 and the loopback network 127.0.0.0/8.
6. Block any source routed packets or any packets with the IP options field set.
7. Reserved, DHCP auto-configuration and Multicast addresses should also be blocked:
    - 0.0.0.0/8
    - 196.254.0.0/16
    - 192.0.2.0/24
    - 224.0.0.0/4
    - 240.0.0.0/4

G5.2 Systems impacted:

Most operating systems and network devices

G5.3 CVE entries:

(Note: This list is not complete or all-inclusive. It is a sample of some of the vulnerabilities covered by this category.)

CAN-1999-0528, CAN-1999-0529, CAN-1999-0240, CAN-1999-0588

G5.4 How to determine if you are vulnerable:

Try to send a spoofed packet and see if your external firewall or router blocks it. Not only should your device block the traffic, but it should also produce a record in the log showing that the spoofed packets have been dropped. Note, however, that this opens up the door to a new attack – flooding the logfile. Make sure your logging system can handle a heavy load, otherwise it could be vulnerable to a DOS attack. Programs like nmap can be used to send decoy packets or spoofed packets to test this type of filtering. Once filtering is set up, don't assume that it is working effectively. Test it often.

G5.5 How to protect against it:

To defend against this type of attack filtering rules should be setup on your external router or firewall. The following are sample rules for a Cisco router:

1. inbound or ingress filtering

```
interface Serial 0

            ip address 10.80.71.1 255.255.255.0

            ip access-group 11 in

access-list 11 deny 192.168.0.0 0.0.255.255

access-list 11 deny 172.16.0.0 0.15.255.255

access-list 11 deny 10.0.0.0 0.255.255.255

access-list 11 deny <your internal network>

access-list 11 permit any
```

2. outbound or egress filtering

```
interface Ethernet 0

ip address 10.80.71.1 255.255.255.0

ip access-group 11 in

access-list 11 permit <your internal network>
```

G6 - Non-existent or incomplete logging

G6.1 Description:

One of the maxims of security is, "Prevention is ideal, but detection is a must." As long as you allow traffic to flow between your network and the Internet, the opportunity for an attacker to sneak in and penetrate the network, is there. New vulnerabilities are discovered every week, and there are very few ways to defend yourself against an attacker using a new vulnerability. Once you are attacked, without logs, you have little chance of discovering what the attackers did. Without that knowledge, your organization must choose between completely reloading the operating system from original media, and then hoping the data back-ups were OK, or taking the risk that you are running a system that a hacker still controls.

You cannot detect an attack if you do not know what is occurring on your network. Logs provide the details of what is occurring, what systems are being attacked, and what systems have been compromised.

Logging must be done on a regular basis on all key systems, and logs should be archived and backed up because you never know when you might need them. Most experts recommend sending all of your logs to a central log server that writes the data to a write once media, so that the attacker cannot overwrite the logs and avoid detection.

G6.2 Systems impacted:

All operating systems and network devices.

G6.3 CVE entries:

CAN-1999-0575, CAN-1999-0576, CAN-1999-0578

G6.4 How to determine if you are vulnerable:

Review the system logs for each major system. If you do not have logs, or if they are not centrally stored and backed-up, you are vulnerable.

G6.5 How to protect against it:

Set up all systems to log information locally, and to send the log files to a remote system. This provides redundancy and an extra layer of security. Now the two logs can be compared against one another. Any differences could indicate suspicious activity on the system. In addition, this allows cross checking of log files. One line in a log file on a single server may not be suspicious, but the same entry on 50 servers across an organization within a minute of each other, may be a sign of a major problem.

Wherever possible, send logging information to a device that uses write-once media.

G7 - Vulnerable CGI Programs

G7.1 Description:

Most web servers, including Microsoft's IIS and Apache, support Common Gateway Interface (CGI) programs to provide interactivity in web pages enabling functions such as data collection and verification. In fact, most web servers are delivered (and installed) with sample CGI programs. Unfortunately, too many CGI programmers fail to consider that their programs provide a direct link from any user anywhere on the Internet directly to the operating system of the computer running the web server. Vulnerable CGI programs present a particularly attractive target to intruders because they are relatively easy to locate and operate with the privileges and power of the web server software itself. Intruders are known to have exploited vulnerable CGI programs to vandalize web pages, steal credit card information, and set up back doors to enable future intrusions. When the Department of Justice web site was vandalized, an in-depth assessment concluded that a CGI hole was the most probable avenue of compromise. Web server applications are similarly vulnerable to threats created by uneducated or careless programmers. As a general rule, sample programs should always be removed from production systems.

G7.2 Systems impacted:

All web servers.

G7.3 CVE entries:

(Note: This list is not complete or all-inclusive. It is a sample of some of the vulnerabilities covered by this category.)

CVE-1999-0067, CVE-1999-0346, CVE-2000-0207, CVE-1999-0467, CAN-1999-0509, CVE-1999-0021, CVE-1999-0039, CVE-1999-0058, CVE-2000-0012, CVE-2000-0039, CVE-2000-0208, CAN-1999-0455, CAN-1999-0477

G7.4 How to determine if you are vulnerable:

If you have any sample code on your web server, you are vulnerable. If you have legitimate CGI programs, ensure you are running the latest version, and then run a vulnerability scanning tool against your site. By simulating what an attacker would do, you will be prepared to protect your systems. To find vulnerable CGI scripts, you may use a CGI scanner called whisker that can be found at:

http://www.wiretrip.net/rfp/

G7.5 How to protect against it:

The following are the key things that need to be done to protect against vulnerable CGI programs:

1. Remove all sample CGI programs from your production web server.
2. Audit the remaining CGI scripts and remove unsafe CGI scripts from all web servers.
3. Ensure all CGI programmers adhere to a strict policy of input buffer length checking in CGI programs.
4. Apply patches for known vulnerabilities that cannot be removed.
5. Make sure that your CGI bin directory does not include any compilers or interpreters.

6. Remove the "view-source" script from the cgi-bin directory.
7. Do not run your web servers with administrator or root privileges. Most web servers can be configured to run with a less privileged account such as "nobody."
8. Do not configure CGI support on Web Servers that do not need it.

Top Vulnerabilities to Windows Systems (W)

W1 - Unicode Vulnerability (Web Server Folder Traversal)

W1.1 Description:

Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language. The Unicode Standard has been adopted by most vendors, including Microsoft. By sending an IIS server a carefully constructed URL containing an invalid Unicode UTF-8 sequence an attacker can force the server to literally 'walk up and out' of a directory and execute arbitrary scripts. This type of attack is also known as the directory traversal attack.

The Unicode equivalents of / and \ which are %2f and %5c, respectively. However, you can also represent these characters using so-called "overlong" sequences. Overlong sequences are technically invalid Unicode representations that are longer than what is actually required to represent the character. Both / and \ can be represented with a single byte. An overlong representation, such as %c0%af for / represents the character using two bytes. IIS was not written to perform a security check on overlong sequences. Thus, passing an overlong Unicode sequence in a URL, will bypass Microsoft's security checks. If the request is made from a directory marked as "executable" the attacker can cause the executable files to be executed on the server. Additional information on the Unicode threat can be found at:

http://www.wiretrip.net/rfp/p/doc.asp?id=57&face=2

W1.2 Systems impacted:

Microsoft Windows NT 4.0 with IIS 4.0 and Windows 2000 server with IIS 5.0, which do not have Service Pack 2 installed.

W1.3 CVE entries:

CVE-2000-0884

W1.4 How to determine if you are vulnerable:

If you are running an un-patched version of IIS, you are probably vulnerable. The best way to tell if you are vulnerable is to run hfnetchk. Hfnetchk is a tool designed for administrators to use to verify the patch level on one or several systems and works across a network. The Unicode directory traversal vulnerability was fixed in the following updates:

- Q269862 - MS00-057
- Q269862 - MS00-078
- Q277873 - MS00-086
- Q293826 - MS01-026
- Q301625 - MS01-044

- Windows 2000 Service Pack 2

If none of those are installed, the system is vulnerable to this issue.

For a more specific verification, test the exploit on your own system to see whether it is successful. Try typing the following command against your IIS web server:

http://victim/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+c:\

This URL may need to be modified to accurately test a particular system. If you have removed the scripts directory (which is recommended), this command will fail. You can test a system by temporarily creating a directory that has execute permissions, or by using another directory that has execute permissions, instead of the scripts directory in the exploit. For example, you may have removed the scripts directory, but have a directory called cgi-bin instead. Test your system by using the cgi-bin directory instead of the scripts directory.

If you are vulnerable, this URL will send back a directory listing of the contents of the c drive, for the vulnerable server. You are essentially running the exploit against your system, just like an attacker would. The only difference is you are issuing a non-intrusive command (like dir), where an attacker could do significant damage or create a back door into your system.

W1.5 How to protect against it:

In order to defend against this exploit, you must install the latest patches from Microsoft. For information on downloading those fixes, see the Microsoft Security Bulletin at:

http://www.microsoft.com/technet/security/bulletin/MS00-078.asp

Both the IIS Lockdown tool and URL Scan will also protect against this vulnerability. The IIS Lockdown tool is designed to help administrators lock down an IIS server, and is available at:

http://www.microsoft.com/technet/security/tools/locktool.asp

URLScan is a filter that will filter out many HTTP requests. For example, it can be used to filter requests containing UTF8 encoded characters. The URLScan tool is available at:

http://www.microsoft.com/technet/security/URLScan.asp

W2 - ISAPI Extension Buffer Overflows

W2.1 Description:

Microsoft's Internet Information Server (IIS) is the web server software found on most web sites deployed on Microsoft Windows NT and Windows 2000 servers. When IIS is installed, several ISAPI extensions are automatically installed. ISAPI, which stands for Internet Services Application Programming Interface, allows developers to extend the capabilities of an IIS server using DLLs. Several of the DLLs, like idq.dll, contain programming errors that cause them to do improper error bounds checking. In particular, they do not block unacceptably long input strings. Attackers can send data to these DLLs, in what is known as a buffer overflow attack, and take full control of an IIS web server.

W2.2 Systems impacted:

The idq.dll buffer overflow impacts Microsoft Index Server 2.0 and Indexing Service in Windows 2000.

The .printer buffer overflow impacts Windows 2000 Server, Advanced Server, and Server Data Center Edition with IIS 5.0 installed. The vulnerable DLL also ships with Windows 2000 Professional, but it is not mapped by default. As a precautionary matter, you should use Group Policy, if possible, to disable Web based printing (under Computer Configuration:Administrative Templates:Printers) on workstations.

W2.3 CVE entries:

CVE-1999-0412, CVE-2001-0241, CAN-2000-1147, CAN-2001-0500

W2.4 How to determine if you are vulnerable:

If your web server does not have at least Service Pack 2 installed, you are probably vulnerable. If you are not sure which patches have been installed, download and run hfnetchk from:

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/hfnetchk.asp

The following patches include the fix for the .printer buffer overflow:

- Q296576 - MS01-023
- Q300972 - MS01-033
- Q301625 - MS01-044
- Windows 2000 SP2
- Q299444 – The Windows NT 4.0 Security Roll-up Package

The following patches include the fix for the idq.dll buffer overflow:

- Q300972 - MS01-033
- Q301625 - MS01-044
- The Windows NT 4.0 Security Roll-up Package

W2.5 How to protect against it:

Install the latest patches from Microsoft. These can be found at:

- Windows NT 4.0:
  http://www.microsoft.com/ntserver/nts/downloads/critical/q299444/default.asp
- Windows 2000 Professional, Server and Advanced Server:
  http://www.microsoft.com/technet/security/bulletin/MS01-044.asp
- Windows 2000 Datacenter Server:
  Patches for Windows 2000 Datacenter Server are hardware-specific and available from the original equipment manufacturer.
- Windows XP:
  The vulnerability does not affect Windows XP.

Also, an administrator should go in and unmap any ISAPI extensions that are not needed.

Check on a regular basis that the extensions have not become re-mapped.

Remember the principle of least privilege, your systems should be running the least number of services needed for them to function properly.

Both the IIS Lockdown tool and URL Scan will also protect against this vulnerability. The IIS Lockdown tool is designed to help administrators lock down an IIS server, and is available at:

http://www.microsoft.com/technet/security/tools/locktool.asp

URLScan is a filter that will filter out many HTTP requests. For example, it can be used to filter requests containing UTF8 encoded characters. The URLScan tool is available at:

http://www.microsoft.com/technet/security/URLScan.asp

W3 - IIS RDS exploit (Microsoft Remote Data Services)

W3.1 Description:

Microsoft's Internet Information Server (IIS) is the web server software found on most web sites deployed on Microsoft Windows NT 4.0. Malicious users exploit programming flaws in IIS's Remote Data Services (RDS) to run remote commands with administrator privileges.

W3.2 Systems impacted:

Microsoft Windows NT 4.0 systems running Internet Information Server have the /msadc virtual directory mapped are most likely vulnerable.

W3.3 CVE entries:

CVE-1999-1011

W3.4 How to determine if you are vulnerable:

If you are running an un-patched system, you are vulnerable.

An excellent guide to the RDS weakness and how to correct it may be found at:
http://www.wiretrip.net/rfp/p/doc.asp?id=29&iface=2

W3.5 How to protect against it:

This is not fixable via a patch. To protect against this issue, follow the directions in the security bulletins:

- http://support.microsoft.com/support/kb/articles/q184/3/75.asp
- http://www.microsoft.com/technet/security/bulletin/ms98-004.asp
- http://www.microsoft.com/technet/security/bulletin/ms99-025.asp

Alternatively, you can prevent this problem by upgrading to a version of MDAC greater than 2.1. The most recent MDAC versions are available at:

http://www.microsoft.com/data/download.htm

W4 - NETBIOS - unprotected Windows networking shares

W4.1 Description:

The Server Message Block (SMB) protocol, also known as the Common Internet File System (CIFS), enables file sharing over networks. Improper configuration can expose critical system files or give full file system access to any hostile party connected to the Internet. Many computer owners unknowingly open their systems to hackers when they try to improve convenience for coworkers and outside researchers by making their drives readable and writeable by network users. Administrators of a government computer site used for software development for mission planning made their files world readable, so that people at a different government facility could get easy access. Within two days, attackers had discovered the open file shares and had stolen the mission planning software.

Enabling file sharing on Windows machines makes them vulnerable to both information theft and certain types of quick-moving viruses. Macintosh and Unix computers are also vulnerable to file sharing exploits if users enable file sharing.

The SMB mechanisms that permit Windows File Sharing may also be used by attackers to obtain sensitive system information from Windows systems. User and Group information (usernames, last logon dates, password policy, RAS information), system information, and certain Registry keys may all be accessed via a "null session" connection to the NetBIOS Session Service. This information is useful to hackers because it helps them mount a password guessing or brute force password attack against the Windows target.

W4.2 Systems impacted:

Microsoft Windows NT and Windows 2000 systems

W4.3 CVE entries:

CVE-1999-0366, CVE-2000-0222, CVE-2000-0979, CAN-1999-0518, CAN-1999-0519, CAN-1999-0520, CAN-1999-0621, CAN-2000-1079

W4.4 How to determine if you are vulnerable:

A quick, free, and secure test for the presence of SMB file sharing and its related vulnerabilities, effective for machines running any Windows operating system, is available at the Gibson Research Corporation web site at http://grc.com/. Click the "ShieldsUP" icon to receive a real-time appraisal of any system's SMB exposure. Detailed instructions are available to help Microsoft Windows users deal with SMB vulnerabilities. Note that if you are connected over a network where some intermediate device blocks SMB, the ShieldsUP tool will report that you are not vulnerable when, in fact, you are. This is the case, for example, for users on a cable modem where the provider is blocking SMB into the cable modem network. ShieldsUP will report that you are not vulnerable. However, the 4,000 or so other people on your cable modem link can still exploit this vulnerability.

The Microsoft Personal Security Advisor, will report whether you are vulnerable to SMB exploits, and can also fix the problem. Since it runs locally, its results will always be reliable. It is available at: http://www.microsoft.com/technet/security/tools/mpsa.asp

W4.5 How to protect against it:

Take the following steps to defend against unprotected shares:

1. When sharing data, ensure only required directories are shared.
2. For added security, allow sharing only to specific IP addresses because DNS names can be spoofed.
3. For Windows systems (both NT and 2000), use file system permission to ensure that the permissions on the shared directories allow access only to those people who require access.
4. For Windows systems, prevent anonymous enumeration of users, groups, system configuration and registry keys via the "null session" connection. See item W5 for more information
5. Block inbound connections to the NetBIOS Session Service (tcp 139) and Microsoft CIFS (TCP/UDP 445) at the router or the host.
6. Consider implementing the RestrictAnonymous registry key for Internet-connected hosts in standalone or non-trusted domain environments. For more information see the following web pages:

Windows NT 4.0: http://support.microsoft.com/support/kb/articles/Q143/4/74.asp
Windows 2000: http://support.microsoft.com/support/kb/articles/Q246/2/61.ASP

W5 - Information leakage via null session connections

W5.1 Description:

A Null Session connection, also known as Anonymous Logon, is a mechanism that allows an anonymous user to retrieve information (such as user names and shares) over the network, or to connect without authentication. It is used by applications such as explorer.exe to enumerate shares on remote servers. On Windows NT and Windows 2000 systems, many local services run under the SYSTEM account, known as LocalSystem on Windows 2000. The SYSTEM account is used for various critical system operations. When one machine needs to retrieve system data from another, the SYSTEM account will open a null session to the other machine.

The SYSTEM account has virtually unlimited privileges and it has no password, so you can't log on as SYSTEM. SYSTEM sometimes needs to access information on other machines such as available shares, user names, etc. -- Network Neighborhood type functionality. Because it cannot log into the other systems using a UserID and password, it uses a Null session to get access. Unfortunately attackers can also log in as the Null Session.

W5.2 Systems impacted:

Windows NT 4.0 and Windows 2000 systems

W5.3 CVE entries:

CAN-2000-1200

W5.4 How to determine if you are vulnerable:

Try to connect to your system via a Null session using the following command:

net use \\a.b.c.d\ipc$ "" /user:""

(where a.b.c.d is the IP address of the remote system.)

If you receive a "connection failed" response, then your system is not vulnerable. If no reply comes back that means that the command was successful and your system is vulnerable.

"Hunt for NT" can also be used. It is a component of the NT Forensic Toolkit from www.foundstone.com.

W5.5 How to protect against it:

Domain controllers require Null sessions to communicate. Therefore, if you are working in a domain environment, you can minimize the information that attackers can obtain, but you cannot stop all leakage. To limit the information available to attackers, on a Windows NT 4.0 machine, modify the following registry key:

HKLM/System/CurrentControlSet/Control/LSA/RestrictAnonymous=1

Setting RestrictAnonymous to 1 will still make certain information available to anonymous users. On Windows 2000 you can set the value to 2 instead. Doing so will bar anonymous users from all information where explicit access has not been granted to them or the Everyone group, which includes null session users.

Whenever you modify the registry, it could cause your system to stop working properly. Therefore any changes should be tested before hand. Also, the system should always be backed up to simplify recovery. If you do not need file and print sharing, unbind NetBIOS from TCP/IP.

Note here that configuring RestricAnonymous on domain controllers and certain other servers can disrupt many normal networking operations. For this reason, it is recommended that only those machines which are visible to the Internet have this value configured. All other machines should be protected by a firewall configured to block NetBIOS and CIFS.

Internet users should never be allowed to access any internal domain controller or other computer not specifically built for external access. To stop such access, block the following ports at the external router or firewall:

TCP and UDP 135 through139 and 445

W6 - Weak hashing in SAM (LM hash)

W6.1 Description:

Though most Windows users have no need for LAN Manager support, Microsoft stores LAN Manager password hashes, by default, on Windows NT and 2000 systems. Since LAN Manager uses a much weaker encryption scheme than do the more current Microsoft approaches, LAN Manager passwords can be broken in a very short period of time. Even strong password hashes can be cracked in under a month. The major weaknesses of LAN Manager hashes is the following:

- password truncated to 14 characters
- password padded with spaces to become 14 characters
- password converted to all upper case characters

- password split into two seven character pieces

This means that a password cracking program has to crack only two seven-character passwords without even testing lower case letters. In addition, LAN Manager is vulnerable to eavesdropping of the password hashes. Eavesdropping can provide attackers with user passwords.

W6.2 Systems impacted:

Microsoft Windows NT and 2000 computers

W6.3 CVE entries:

N/A

W6.4 How to determine if you are vulnerable:

If you are running a default installation of NT or 2000, you are vulnerable since LAN Manager hashes are created by default. You may (if you have specific written permission from your employer) test the ease of password cracking on your own systems using an automated password cracking tool like LC3 (l0phtcrack version 3) available from: http://www.atstake.com/research/lc3/download.html

W6.5 How to protect against it:

Protecting against password cracking of the LMHash can be done two ways. The first is to disable LAN Manger authentication across the network and use NTLMv2. NTLMv2 (NT LanManager version 2) challenge/response methods overcome most weaknesses in Lan Manager (LM) by using stronger encryption and improved authentication and session security mechanisms.

With Windows NT 4.0 SP4 and newer systems, including Windows 2000, Microsoft makes it posible to use only NTLMv2 in your network. The registry key that controls this capability in both Windows NT and 2000 is HKLM\System\CurrentControlSet\Control\LSA\ LMCompatibilityLevel. If you set its value to 3, the workstation or server will present only NTLMv2 credentials for authentication. If you set it to 5, any domain controller will refuse LM and NTLM authentication and will only accept NTLMv2.

You have to carefully plan the changes if you still have older systems, such as Windows 95, on your network. Older systems won't use NTLMv2 with the Microsoft Network Client. In Win 9x, the parameter is HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\LSA\LMCompatibility, and the allowed values are 0 or 3 (with Directory Services Client). The safest option is to get rid of those older systems, since they do not allow you to provide the minimum security level an organization requires.

The Microsoft Technet article "How to Disable LM Authentication on Windows NT [Q147706]" details the required changes in the registry for Windows 9x and Windows NT/2000. "LMCompatibilityLevel and Its Effects [Q175641]" explains the interoperability issues with this parameter. Another very useful article from Technet is "How to Enable NTLMv2 Authentication for Windows 95/98/2000/NT [Q239869]." It explains the use of the Windows 2000's Directory Services Client for Windows 95/98 to overcome the compatibility limitation for NTLMv2.

The problem with simply removing the LanMan hashes on the network is that the hashes are

still created and stored in the SAM or the Active Directory. Microsoft very recently made a new mechanism available for turning off the creation of the LanMan hashes altogether. On Windows 2000 systems, go to the following registry key:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa

On the Edit menu in RegEdt32 or RegEdit click Add Key… and add a key called NoLMHash. After doing this, quit the registry editor and reboot the computer. The next time a user changes his or her password, the computer will no longer create a LanMan hash at all. If this key is created on a Windows 2000 Domain Controller, the LanMan hashes will no longer be created and stored in Active Directory.

On Windows XP, the same functionality can be implemented by setting a registry value:

Hive: HKEY_LOCAL_MACHINE
Key: System\CurrentControlSet\Control\Lsa
Value: NoLMHash
Type: REG_DWORD
Data: 1

This will have the same exact effect as creating the NoLMHash key under Windows 2000.

For more information on these changes, refer to Microsoft KnowledgeBase Article Q299656 at:

http://support.microsoft.com/support/kb/articles/q299/6/56.asp.

Top Vulnerabilities To Unix Systems (U)

U1 - Buffer Overflows in RPC Services

U1.1 Description:

Remote procedure calls (RPCs) allow programs on one computer to execute programs on a second computer. They are widely used to access network services such as NFS file sharing and NIS. Multiple vulnerabilities caused by flaws in RPC are being actively exploited. There is compelling evidence that the majority of the distributed denial of service attacks launched during 1999 and early 2000 were executed by systems that had been victimized through the RPC vulnerabilities. The broadly successful attack on U.S. military systems during the Solar Sunrise incident also exploited an RPC flaw found on hundreds of Department of Defense systems.

U1.2 Systems impacted:

Most versions of Unix

U1.3 CVE entries:

CVE-1999-0003, CVE-1999-0693, CVE-1999-0696, CVE-1999-0018, CVE-1999-0019, CVE-1999-0704, CAN-2001-0236, CVE-2000-0666

U1.4 How to determine if you are vulnerable:

Check to see if you are running one of the three RPC services that are most commonly exploited:

- rpc.ttdbserverd
- rpc.cmsd
- rpc.statd

These services are commonly exploited through buffer overflow attacks which are successful because the RPC programs do not do proper error checking. A buffer overflow vulnerability allows an attacker to send data that the program is not expecting, and because the program does poor error checking, it passes the data on for processing.

U1.5 How to protect against it:

Use the following steps to protect your systems against RPC attacks:

1. Wherever possible, turn off and/or remove these services on machines directly accessible from the Internet.

2. Where you must run them, install the latest patches:

For Solaris Software Patches:
http://sunsolve.sun.com

For IBM AIX Software
http://techsupport.services.ibm.com/support/rs6000.support/downloads
http://techsupport.services.ibm.com/rs6k/fixes.html

For SGI Software Patches:
http://support.sgi.com/

For Compaq (Digital Unix) Patches:
http://www.compaq.com/support

For Linux:
http://www.redhat.com/support/errata/RHSA-2000-039-02.html
http://www.debian.org/security/2000/20000719a
http://www.cert.org/advisories/CA-2000-17.html

3. Regularly search the vendor patch database for new patches and install them right away

4. Block the RPC port (port 111) at the border router or firewall.

5. Block the RPC "loopback" ports, 32770-32789 (TCP and UDP)

A summary document pointing to specific guidance about each of three principal RPC vulnerabilities may be found at: http://www.cert.org/incident_notes/IN-99-04.html

The following provides information on each of the vulnerable services:

statd: http://www.cert.org/advisories/CA-99-05-statd-automountd.html

ToolTalk: http://www.cert.org/advisories/CA-98.11.tooltalk.html
Calendar Manager: http://www.cert.org/advisories/CA-99-08-cmsd.html

U2 - Sendmail Vulnerabilities

U2.1 Description:

Sendmail is the program that sends, receives, and forwards most electronic mail processed on UNIX and Linux computers. Sendmail's widespread use on the Internet makes it a prime target of attackers. Several flaws have been found over the years. In fact, the very first advisory issued by CERT/CC, in 1988, made reference to an exploitable weakness in Sendmail. In one of the most common exploits, the attacker sends a crafted mail message to the machine running Sendmail, and Sendmail reads the message as instructions requiring the victim machine to send its password file to the attacker's machine (or to another victim) where the passwords can be cracked.

U2.2 Systems impacted:

Most versions of Unix and Linux

U2.3 CVE entries:

CVE-1999-0047, CVE-1999-0130, CVE-1999-0131, CVE-1999-0203, CVE-1999-0204, CVE-1999-0206

U2.4 How to determine if you are vulnerable:

Sendmail has a large number of vulnerabilities and must be regularly updated and patched. Check to see what the latest version and patch level is for sendmail; if you are not running it, you are probably vulnerable.

U2.5 How to protect against it:

The following steps should be taken to protect sendmail:

1. Upgrade to latest version of Sendmail and/or implement patches for sendmail. http://www.cert.org/advisories/CA-97.05.sendmail.html
2. Do not run Sendmail in daemon mode (turn off the -bd switch) on machines that are neither mail servers nor mail relays.

U3 - Bind Weaknesses

U3.1 Description:

The Berkeley Internet Name Domain (BIND) package is the most widely used implementation of Domain Name Service (DNS) -- the critical means by which we all locate systems on the Internet by name (e.g., www.sans.org) without having to know specific IP addresses -- and this makes it a favorite target for attack. Sadly, according to a mid-1999 survey, as many as 50% of all DNS servers connected to the Internet are running vulnerable versions of BIND. In a typical example of a BIND attack, intruders erased the system logs and installed tools to gain administrative access. They then compiled and installed IRC utilities and network

scanning tools, which they used to scan more than a dozen class-B networks in their search for additional systems running vulnerable versions of BIND. In a matter of minutes, they had used the compromised system to attack hundreds of remote systems, resulting in many additional successful compromises. This example illustrates the chaos that can result from a single vulnerability in the software for ubiquitous Internet services such as DNS. Outdated versions of Bind also include buffer overflow exploits that attackers can use to get unauthorized access.

U3.2 Systems impacted:

Multiple UNIX and Linux systems

U3.3 CVE entries:

CVE-1999-0024, CVE-1999-0184, CVE-1999-0833, CVE-1999-0009, CVE-1999-0835, CVE-1999-0848, CVE-1999-0849, CVE-1999-0851, CVE-2001-0010, CVE-2001-0011, CVE-2001-0013

U3.4 How to determine if you are vulnerable:

Run a vulnerability scanner, check the version of BIND, or manually check the files to see if they are vulnerable. If in doubt, err on the side of caution, and upgrade the system.

U3.5 How to protect against it:

The following steps should be taken to defend against the BIND vulnerabilities:

1. Disable the BIND name daemon (called "named") on all systems that are not authorized to be DNS servers. Some experts recommend you also remove the DNS software.
2. On machines that are authorized DNS servers, update to the latest version and patch level. Use the guidance contained in the following advisories:
3. For the NXT vulnerability: http://www.cert.org/advisories/CA-99-14-bind.html
   For the QINV (Inverse Query) and NAMED vulnerabilities:
   http://www.cert.org/advisories/CA-98.05.bind_problems.html
   http://www.cert.org/summaries/CS-98.04.html
4. Run BIND as a non-privileged user for protection in the event of future remote-compromise attacks. (However, only processes running as root can be configured to use ports below 1024 – a requirement for DNS. Therefore you must configure BIND to change the user-id after binding to the port.)
5. Run BIND in a chroot()ed directory structure for protection in the event of future remote-compromise attacks.
6. Disable zone transfers except from authorized hosts.
7. Disable recursion and glue fetching, to defend against DNS cache poisoning.
8. Hide your version string.

U4 - R Commands

U4.1 Description:

Trust relationships are widely used in the UNIX world, particularly for system administration. Companies frequently assign a single administrator to be responsible for dozens or even hundreds of systems. Administrators often use trust relationships and the related UNIX r

commands to switch from system to system conveniently. r commands enable someone to access a remote system without supplying a password. Instead of requiring a username/password combination, the remote machine authenticates anyone coming from a trusted IP addresses. If an attacker gains control of any machine in such a trusted network, he or she can gain access to all other machines that trust the hacked machine. The following r commands are often used:

1. rlogin – remote login
2. rsh – remote shell
3. rcp – remote copy

U4.2 Systems impacted:

Most variants of Unix, including Linux

U4.3 CVE entries:

CVE-1999-0046, CVE-1999-0113, CVE-1999-0185, CAN-1999-0651

U4.4 How to determine if you are vulnerable:

Trust relationships are established by configuring two files, either /etc/hosts.equiv or ~/.rhosts. Check for both of those files on your Unix systems to determine whether trust relationships have been configured.

U4.5 How to protect against it:

Do not allow IP-based trust relationships, and do not use the r commands. Authentication based on IP addresses is too easy to bypass. Authentication should be based on more secure means such as tokens or, at the least, passwords. If r commands are required, limit the access, and control the perimeter of the network extremely carefully. Never allow the ".rhosts" file in the "root" account. You can use the Unix "find" command regularly to look for any ".rhosts" files that may have been created in other user accounts.

U5 - LPD (remote print protocol daemon)

U5.1 Description:

In Unix, the in.lpd provides services for users to interact with the local printer. LPD listens for requests on TCP port 515. The programmers who developed the code that transfers print jobs from one machine to another made an error that creates a buffer overflow vulnerability. If the daemon is given too many jobs within a short time interval, the daemon will either crash or run arbitrary code with elevated privileges.

U5.2 Systems impacted:

The following systems are impacted:

- Solaris 2.6 for SPARC
- Solaris 2.6 x86
- Solaris 7 for SPARC
- Solaris 7 x86

- Solaris 8 for SPARC
- Solaris 8 x86
- Most variants of Linux

U5.3 CVE entries:

CVE-1999-0032, CVE-1999-0299, CVE-2000-0917, CAN-2001-0670, CAN-2001-0668, CAN-2001-0353, CAN-1999-0061

U5.4 How to determine if you are vulnerable:

Either a vulnerability scanner can be run against your system to look for this vulnerability, or a manual check can be run. The easiest way to run a manual check is to see if your system is running LPD and check the version number.

If you are running one of the vulnerable versions of the software, and have not applied a patch, then you are vulnerable.

U5.5 How to protect against it:

Sun released Sun Security Bulletin #00206 regarding this issue on August 30, 2001 detailing the patch information. The bulletin is available from: http://sunsolve.sun.com/security The CERT Advisory for this topic can be found at: http://www.cert.org/advisories/CA-2001-15.html

A patch for Linux can be found at http://redhat.com/support/errata/RHSA-2001-077.html

Other options for defending against attacks using this vulnerability include:

1. Disable the print service in */etc/inetd.conf* if remote print job handling is unnecessary.
2. Enable the **noexec_user_stack,** tunable by adding the following lines to the */etc/system* file, and reboot:

- set noexec_user_stack = 1
- set noexec_user_stack_log = 1

3. Block access to network port 515/tcp
4. Deploy tcpwrappers, which are part of the **tcpd-7.6** package and can be downloaded from: http://www.sun.com/solaris/freeware.html#cd

U6 – sadmind and mountd

U6.1 Description:

Sadmind allows remote administration access to Solaris systems, providing a graphical user interface for system administration functions. Mountd controls and arbitrates access to NFS mounts on UNIX hosts. Buffer overflows in these applications, enabled by programming errors made by the software developers, can be exploited to allow attackers to gain control with root access.

Note: This item is a special case of U.1 Buffer Overflows in RPC Services. The contributors

saw this occur so often that they felt it was important to break it out into a second item.

U6.2 Systems impacted:

Multiple versions of Unix

U6.3 CVE entries:

CVE-1999-0977, CVE-1999-0002, CVE-1999-0493, CVE-1999-0210

U6.4 How to determine if you are vulnerable:

Use a vulnerability scanner to see whether these services are running and whether they are vulnerable to attack.

U6.5 How to protect against it:

The following actions will protect against NFS vulnerabilities, including sadmind and mountd:

1. Wherever possible, turn off and/or remove sadmind and mountd on machines directly accessible from the Internet.

2. Install the latest patches:

For Solaris Software Patches:
http://sunsolve.sun.com

For IBM AIX Software
http://techsupport.services.ibm.com/support/rs6000.support/downloads
http://techsupport.services.ibm.com/rs6k/fixes.html

For SGI Software Patches:
http://support.sgi.com/

For Compaq (Digital Unix) Patches:
http://www.compaq.com/support

3. Use host/ip based export lists

4. Setup export file systems for read-only or no suid whereever possible

5. Use nfsbug to scan for vulnerabilities

Additional information can be found at:

http://www.cert.org/advisories/CA-99-16-sadmind.html
http://www.cert.org/advisories/CA-98.12.mountd.html

U7 - Default SNMP Strings

U7.1 Description:

The Simple Network Management Protocol (SNMP) is widely used by network administrators to monitor and administer all types of network-connected devices ranging from routers to printers to computers. SNMP uses an unencrypted "community string" as its only authentication mechanism. Lack of encryption is bad enough, but the default community string used by the vast majority of SNMP devices is "public", with a few "clever" network equipment vendors changing the string to "private" for more sensitive information. Attackers can use this vulnerability in SNMP to reconfigure or shut down devices remotely. Sniffed SNMP traffic can reveal a great deal about the structure of your network, as well as the systems and devices attached to it. Intruders use such information to pick targets and plan attacks.

Note: SNMP is not unique to Unix. However, the reason it is listed under Unix is because the contributors have seen a majority of attacks on Unix systems caused by poor SNMP configurations. The contributors have not seen this as a major problem on Windows Systems.

U7.2 Systems impacted:

All UNIX systems and network devices

U7.3 CVE entries:

CAN-1999-0517, CAN-1999-0516, CAN-1999-0254, CAN-1999-0186

U7.4 How to determine if you are vulnerable:

Check to see if you have SNMP running on your devices. If you do, check the configuration files for the common vulnerabilities:

- Default or blank SNMP community names
- Guessable SNMP community names

Hidden SNMP community strings

U7.5 How to protect against it:

The following steps will help defend against SNMP exploits:

1. If you do not absolutely require SNMP, disable it.
2. If you must use SNMP, use the same policy for community names as used for passwords. Make sure they are difficult to guess or crack, and that they are changed periodically.
3. Validate and check community names using snmpwalk. Additional information can be found at: http://www.zend.com/manual/function.snmpwalk.php
4. Filter SNMP (Port 161/UDP) at the border-router or firewall unless it is absolutely necessary to poll or manage devices from outside of the local network.
5. Where possible make MIBs read only. Additional information can be found at: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/snmp.htm#xtocid210315

Appendix A – Common Vulnerable Ports

In this section, we list ports that are commonly probed and attacked. Blocking these ports is a minimum requirement for perimeter security, not a comprehensive firewall specification list. A far better rule is to block all unused ports. And even if you believe these ports are blocked, you should still actively monitor them to detect intrusion attempts. A warning is also in order: Blocking some of the ports in the following list may disable needed services. Please consider the potential effects of these recommendations before implementing them.

Keep in mind that blocking these ports is not a substitute for a comprehensive security solution. Even if the ports are blocked, an attacker who has gained access to your network via other means (a dial-up modem, a trojan e-mail attachment, or a person who is an organization insider, for example) can exploit these ports if not properly secured on every host system in your organization.

1. Login services-- telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), rlogin et al (512/tcp through 514/tcp)
2. RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)
3. NetBIOS in Windows NT -- 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)
4. X Windows -- 6000/tcp through 6255/tcp
5. Naming services-- DNS (53/udp) to all machines which are not DNS servers, DNS zone transfers (53/tcp) except from external secondaries, LDAP (389/tcp and 389/udp)
6. Mail-- SMTP (25/tcp) to all machines, which are not external mail relays, POP (109/tcp and 110/tcp), IMAP (143/tcp)
7. Web-- HTTP (80/tcp) and SSL (443/tcp) except to external Web servers, may also want to block common high-order HTTP port choices (8000/tcp, 8080/tcp, 8888/tcp, etc.)
8. "Small Services"-- ports below 20/tcp and 20/udp, time (37/tcp and 37/udp)
9. Miscellaneous-- TFTP (69/udp), finger (79/tcp), NNTP (119/tcp), NTP (123/udp), LPD (515/tcp), syslog (514/udp), SNMP (161/tcp and 161/udp, 162/tcp and 162/udp), BGP (179/tcp), SOCKS (1080/tcp)
10. ICMP-- block incoming echo request (ping and Windows traceroute), block outgoing echo replies, time exceeded, and destination unreachable messages **except** "packet too big" messages (type 3, code 4). (This item assumes that you are willing to forego the legitimate uses of ICMP echo request in order to block some known malicious uses.)

In addition to these ports, block "spoofed" addresses-- packets coming from outside your company sourced from internal addresses, private (RFC1918 and network 127) and IANA reserved addresses. Also block source routed packets or any packets with IP options set.