

Linux Exposed

Introduction to Denial Of Service

Date Sunday, December 21 @ 20:30:53

Topic Hacking

In this paper I have tried to answer the following questions:

- What is a denial of service attack?
- Why would someone crash a system?
- How can someone crash a system.
- How do I protect a system against denial of service attacks?

I also have a section called SUGGESTED READING were you can find information about good free information that can give you a deeper understanding about something.

=====
=INTRODUCTION TO DENIAL OF SERVICE=
=====

- .A. INTRODUCTION
 - .A.1. WHAT IS A DENIAL OF SERVICE ATTACK?
 - .A.2. WHY WOULD SOMEONE CRASH A SYSTEM?
 - .A.2.1. INTRODUCTION
 - .A.2.2. SUB-CULTURAL STATUS
 - .A.2.3. TO GAIN ACCESS
 - .A.2.4. REVENGE
 - .A.2.5. POLITICAL REASONS
 - .A.2.6. ECONOMICAL REASONS
 - .A.2.7. NASTINESS
 - .A.3. ARE SOME OPERATING SYSTEMS MORE SECURE?
- .B. SOME BASIC TARGETS FOR AN ATTACK
 - .B.1. SWAP SPACE
 - .B.2. BANDWIDTH
 - .B.3. KERNEL TABLES
 - .B.4. RAM
 - .B.5. DISKS
 - .B.6. CACHES
 - .B.7. INETD
- .C. ATTACKING FROM THE OUTSIDE
 - .C.1. TAKING ADVANTAGE OF FINGER

- .C.2. UDP AND SUNOS 4.1.3.
- .C.3. FREEZING UP X-WINDOWS
- .C.4. MALICIOUS USE OF UDP SERVICES
- .C.5. ATTACKING WITH LYNX CLIENTS
- .C.6. MALICIOUS USE OF telnet
- .C.7. MALICIOUS USE OF telnet UNDER SOLARIS 2.4
- .C.8. HOW TO DISABLE ACCOUNTS
- .C.9. LINUX AND TCP TIME, DAYTIME
- .C.10. HOW TO DISABLE SERVICES
- .C.11. PARAGON OS BETA R1.4
- .C.12. NOVELLS NETWARE FTP
- .C.13. ICMP REDIRECT ATTACKS
- .C.14. BROADCAST STORMS
- .C.15. EMAIL BOMBING AND SPAMMING
- .C.16. TIME AND KERBEROS
- .C.17. THE DOT DOT BUG
- .C.18. SUNOS KERNEL PANIC
- .C.19. HOSTILE APPLETS
- .C.20. VIRUS
- .C.21. ANONYMOUS FTP ABUSE
- .C.22. SYN FLOODING
- .C.23. PING FLOODING
- .C.24. CRASHING SYSTEMS WITH PING FROM WINDOWS 95 MACHINES
- .C.25. MALICIOUS USE OF SUBNET MASK REPLY MESSAGE
- .C.26. FLEXIm
- .C.27. BOOTING WITH TRIVIAL FTP

- .D. ATTACKING FROM THE INSIDE
 - .D.1. KERNEL PANIC UNDER SOLARIS 2.3
 - .D.2. CRASHING THE X-SERVER
 - .D.3. FILLING UP THE HARD DISK
 - .D.4. MALICIOUS USE OF eval
 - .D.5. MALICIOUS USE OF fork()
 - .D.6. CREATING FILES THAT IS HARD TO REMOVE
 - .D.7. DIRECTORY NAME LOOKUPCACHE
 - .D.8. CSH ATTACK
 - .D.9. CREATING FILES IN /tmp
 - .D.10. USING RESOLV_HOST_CONF
 - .D.11. SUN 4.X AND BACKGROUND JOBS
 - .D.12. CRASHING DG/UX WITH ULIMIT
 - .D.13. NETTUNE AND HP-UX
 - .D.14. SOLARIS 2.X AND NFS
 - .D.15. SYSTEM STABILITY COMPROMISE VIA MOUNT_UNION
 - .D.16. trap_mon CAUSES KERNEL PANIC UNDER SUNOS 4.1.X

- .E. DUMPING CORE
 - .E.1. SHORT COMMENT
 - .E.2. MALICIOUS USE OF NETSCAPE
 - .E.3. CORE DUMPED UNDER WUFTPD

.E.4. Id UNDER SOLARIS/X86

.F. HOW DO I PROTECT A SYSTEM AGAINST DENIAL OF SERVICE ATTACKS?

.F.1. BASIC SECURITY PROTECTION

.F.1.1. INTRODUCTION

.F.1.2. PORT SCANNING

.F.1.3. CHECK THE OUTSIDE ATTACKS DESCRIBED IN THIS PAPER

.F.1.4. CHECK THE INSIDE ATTACKS DESCRIBED IN THIS PAPER

.F.1.5. EXTRA SECURITY SYSTEMS

.F.1.6. MONITORING SECURITY

.F.1.7. KEEPING UP TO DATE

.F.1.8. READ SOMETHING BETTER

.F.2. MONITORING PERFORMANCE

.F.2.1. INTRODUCTION

.F.2.2. COMMANDS AND SERVICES

.F.2.3. PROGRAMS

.F.2.4. ACCOUNTING

.G. SUGGESTED READING

.G.1. INFORMATION FOR DEEPER KNOWLEDGE

.G.2. KEEPING UP TO DATE INFORMATION

.G.3. BASIC INFORMATION

.H. COPYRIGHT

.I. DISCLAIMER

.O. FOREWORD

In this paper I have tried to answer the following questions:

- What is a denial of service attack?
- Why would someone crash a system?
- How can someone crash a system.
- How do I protect a system against denial of service attacks?

I also have a section called SUGGESTED READING where you can find information about good free information that can give you a deeper understanding about something.

Note that I have a very limited experience with Macintosh, OS/2 and Windows and most of the material are therefore for Unix use.

You can always find the latest version at the following address:
<http://www.student.tdb.uu.se/~t95hhu/secure/denial/DENIAL.TXT>

Feel free to send comments, tips and so on to address:
t95hhu@student.tdb.uu.se

.A. INTRODUCTION

~~~~~

## .A.1. WHAT IS A DENIAL OF SERVICE ATTACK?

-----

Denial of service is about without permission knocking off services, for example through crashing the whole system. This kind of attacks are easy to launch and it is hard to protect a system against them. The basic problem is that Unix assumes that users on the system or on other systems will be well behaved.

## .A.2. WHY WOULD SOMEONE CRASH A SYSTEM?

-----

### .A.2.1. INTRODUCTION

-----

Why would someone crash a system? I can think of several reasons that I have presentated more precisely in a section for each reason, but for short:

- .1. Sub-cultural status.
- .2. To gain access.
- .3. Revenge.
- .4. Political reasons.
- .5. Economical reasons.
- .6. Nastiness.

I think that number one and six are the more common today, but that number four and five will be the more common ones in the future.

### .A.2.2. SUB-CULTURAL STATUS

-----

After all information about syn flooding a bunch of such attacks were launched around Sweden. The very most of these attacks were not a part of a IP-spoof attack, it was "only" a denial of service attack. Why?

I think that hackers attack systems as a sub-cultural pseudo career and I think that many denial of service attacks, and here in the example syn flooding, were performed for these reasons. I also think that many hackers begin their carrer with denial of service attacks.

### .A.2.3. TO GAIN ACCESS

-----

Sometimes could a denial of service attack be a part of an attack to gain access at a system. At the moment I can think of these reasons and specific holes:

.1. Some older X-lock versions could be crashed with a method from the denial of service family leaving the system open. Physical access was needed to use the work space after.

.2. Syn flooding could be a part of a IP-spoof attack method.

.3. Some program systems could have holes under the startup, that could be used to gain root, for example SSH (secure shell).

.4. Under an attack it could be usable to crash other machines in the network or to deny certain persons the ability to access the system.

.5. Also could a system being booted sometimes be subverted, especially rarp-boots. If we know which port the machine listen to (69 could be a good guess) under the boot we can send false packets to it and almost totally control the boot.

#### .A.2.4. REVENGE

-----

A denial of service attack could be a part of a revenge against a user or an administrator.

#### .A.2.5. POLITICAL REASONS

-----

Sooner or later will new or old organizations understand the potential of destroying computer systems and find tools to do it.

For example imagine the Bank A loaning company B money to build a factory threatening the environment. The organization C therefor crash A:s computer system, maybe with help from an employee. The attack could cost A a great deal of money if the timing is right.

#### .A.2.6. ECONOMICAL REASONS

-----

Imagine the small company A moving into a business totally dominated by company B. A and B customers make the orders by computers and depends heavily on that the order is done in a specific time (A and B could be stock trading companies). If A and B can't perform the order the customers lose money and change company.

As a part of a business strategy A pays a computer expert a sum of money to get him to crash B:s computer systems a number of times. A year later A is the dominating company.

#### .A.2.7. NASTINESS

-----

I know a person that found a workstation where the user had forgotten to logout. He sat down and wrote a program that made a kill -9 -1 at a random time at least 30 minutes after the login time and placed a call to the program from the profile file. That is nastiness.

### .A.3. ARE SOME OPERATING SYSTEMS MORE SECURE?

-----

This is a hard question to answer and I don't think that it will give anything to compare different Unix platforms. You can't say that one Unix is more secure against denial of service, it is all up to the administrator.

A comparison between Windows 95 and NT on one side and Unix on the other could however be interesting.

Unix systems are much more complex and have hundreds of built in programs, services... This always open up many ways to crash the system from the inside.

In the normal Windows NT and 95 network were is few ways to crash the system. Although were is methods that always will work.

That gives us that no big different between Microsoft and Unix can be seen regarding the inside attacks. But there is a couple of points left:

- Unix have much more tools and programs to discover an attack and monitoring the users. To watch what another user is up to under windows is very hard.
- The average Unix administrator probably also have much more experience than the average Microsoft administrator.

The two last points gives that Unix is more secure against inside denial of service attacks.

A comparison between Microsoft and Unix regarding outside attacks are much more difficult. However I would like to say that the average Microsoft system on the Internet are more secure against outside attacks, because they normally have much less services.

### .B. SOME BASIC TARGETS FOR AN ATTACK

~~~~~

.B.1. SWAP SPACE

Most systems have several hundred Mbytes of swap space to service client requests. The swap space is typical used for forked child processes which have a short life time. The swap space will therefore almost never in a normal

cause be used heavily. A denial of service could be based on a method that tries to fill up the swap space.

.B.2. BANDWIDTH

If the bandwidth is too high the network will be useless. Most denial of service attacks influence the bandwidth in some way.

.B.3. KERNEL TABLES

It is trivial to overflow the kernel tables which will cause serious problems on the system. Systems with write through caches and small write buffers are especially sensitive.

Kernel memory allocation is also a target that is sensitive. The kernel has a kernelmap limit, if the system reaches this limit it can not allocate more kernel memory and must be rebooted. The kernel memory is not only used for RAM, CPU:s, screens and so on, it is also used for ordinary processes. Meaning that any system can be crashed and with a mean (or in some sense good) algorithm pretty fast.

For Solaris 2.X it is measured and reported with the sar command how much kernel memory the system is using, but for SunOS 4.X there is no such command. Meaning that under SunOS 4.X you don't even get a warning. If you do use Solaris you should write `sar -k 1` to get the information. `netstat -k` can also be used and shows how much memory the kernel has allocated in the subpaging.

.B.4. RAM

A denial of service attack that allocates a large amount of RAM can make a great deal of problems. NFS and mail servers are actually extremely sensitive because they do not need much RAM and therefore often don't have much RAM. An attack at a NFS server is trivial. The normal NFS client will do a great deal of caching, but a NFS client can be anything including the program you wrote yourself...

.B.5. DISKS

A classic attack is to fill up the hard disk, but an attack at the disks can be so much more. For example can an overloaded disk be misused in many ways.

.B.6. CACHES

SunOS 4.1.3. is known to boot if a packet with incorrect information in the header is sent to it. This is the cause if the ip_options indicate a wrong size of the packet.

The solution is to install the proper patch.

.C.3. FREEZING UP X-WINDOWS

If a host accepts a telnet session to the X-Windows port (generally somewhere between 6000 and 6025. In most cases 6000) could that be used to freeze up the X-Windows system. This can be made with multiple telnet connections to the port or with a program which sends multiple XOpenDisplay() to the port.

The same thing can happen to Motif or Open Windows.

The solution is to deny connections to the X-Windows port.

.C.4. MALICIOUS USE OF UDP SERVICES

It is simple to get UDP services (echo, time, daytime, chargen) to loop, due to trivial IP-spoofing. The effect can be high bandwidth that causes the network to become useless. In the example the header claim that the packet came from 127.0.0.1 (loopback) and the target is the echo port at system.we.attack. As far as system.we.attack knows is 127.0.0.1 system.we.attack and the loop has been establish.

Ex:

```
from-IP=127.0.0.1
to-IP=system.we.attack
Packet type:UDP
from UDP port 7
to UDP port 7
```

Note that the name system.we.attack looks like a DNS-name, but the target should always be represented by the IP-number.

Quoted from proberts@clark.net (Paul D. Robertson) comment on comp.security.firewalls on matter of "Introduction to denial of service"

" A great deal of systems don't put loopback on the wire, and simply emulate it. Therefore, this attack will only effect that machine in some cases. It's much better to use the address of a different machine on the same network. Again, the default services should be disabled in inetd.conf. Other than some hacks for mainframe IP stacks that don't support ICMP, the echo service isn't used by many legitimate programs, and TCP echo should be used instead of UDP where it is necessary. "

.C.5. ATTACKING WITH LYNX CLIENTS

A World Wide Web server will fork an httpd process as a respond to a request from a client, typical Netscape or Mosaic. The process lasts for less than one second and the load will therefore never show up if someone uses ps. In most causes it is therefore very safe to launch a denial of service attack that makes use of multiple W3 clients, typical lynx clients. But note that the netstat command could be used to detect the attack (thanks to Paul D. Robertson).

Some httpd:s (for example http-gw) will have problems besides the normal high bandwidth, low memory... And the attack can in those causes get the server to loop (compare with .C.6.)

.C.6. MALICIOUS USE OF telnet

Study this little script:

Ex:

```
while : ; do
telnet system.we.attack &
done
```

An attack using this script might eat some bandwidth, but it is nothing compared to the finger method or most other methods. Well the point is that some pretty common firewalls and httpd:s thinks that the attack is a loop and turn them self down, until the administrator sends kill -HUP.

This is a simple high risk vulnerability that should be checked and if present fixed.

.C.7. MALICIOUS USE OF telnet UNDER SOLARIS 2.4

If the attacker makes a telnet connections to the Solaris 2.4 host and quits using:

Ex:

```
Control-}
quit
```

then will inetd keep going "forever". Well a couple of hundred...

The solution is to install the proper patch.

.C.8. HOW TO DISABLE ACCOUNTS

Some systems disable an account after N number of bad logins, or waits N seconds. You can use this feature to lock out specific users from the system.

.C.9. LINUX AND TCP TIME, DAYTIME

Inetd under Linux is known to crash if to many SYN packets sends to daytime (port 13) and/or time (port 37).

The solution is to install the proper patch.

.C.10. HOW TO DISABLE SERVICES

Most Unix systems disable a service after N sessions have been open in a given time. Well most systems have a reasonable default (lets say 800 - 1000), but not some SunOS systems that have the default set to 48...

The solutions is to set the number to something reasonable.

.C.11. PARAGON OS BETA R1.4

If someone redirects an ICMP (Internet Control Message Protocol) packet to a paragon OS beta R1.4 will the machine freeze up and must be rebooted. An ICMP redirect tells the system to override routing tables. Routers use this to tell the host that it is sending to the wrong router.

The solution is to install the proper patch.

.C.12. NOVELLS NETWARE FTP

Novells Netware FTP server is known to get short of memory if multiple ftp sessions connects to it.

.C.13. ICMP REDIRECT ATTACKS

Gateways uses ICMP redirect to tell the system to override routing tables, that is telling the system to take a better way. To be able to misuse ICMP redirection we must know an existing connection (well we could make one for ourself, but there is not much use for that). If we have found a connection we can send a route that loses it connectivity or we could send false messages to the host if the connection we have found don't use cryptation.

Ex: (false messages to send)

DESTINATION UNREACHABLE
TIME TO LIVE EXCEEDED
PARAMETER PROBLEM
PACKET TOO BIG

The effect of such messages is a reset of the connection.

The solution could be to turn ICMP redirects off, not much proper use of the service.

.C.14. BROADCAST STORMS

This is a very popular method in networks there all of the hosts are acting as gateways.

There are many versions of the attack, but the basic method is to send a lot of packets to all hosts in the network with a destination that don't exist. Each host will try to forward each packet so the packets will bounce around for a long time. And if new packets keep coming the network will soon be in trouble.

Services that can be misused as tools in this kind of attack is for example ping, finger and sendmail. But most services can be misused in some way or another.

.C.15. EMAIL BOMBING AND SPAMMING

In a email bombing attack the attacker will repeatedly send identical email messages to an address. The effect on the target is high bandwidth, a hard disk with less space and so on... Email spamming is about sending mail to all (or rather many) of the users of a system. The point of using spamming instead of bombing is that some users will try to send a replay and if the address is false will the mail bounce back. In that cause have one mail transformed to three mails. The effect on the bandwidth is obvious.

There is no way to prevent email bombing or spamming. However have a look at CERT:s paper "Email bombing and spamming".

.C.16. TIME AND KERBEROS

If not the the source and target machine is closely aligned will the ticket be rejected, that means that if not the protocol that set the time is protected it will be possible to set a kerberos server of function.

.C.17. THE DOT DOT BUG

Windows NT file sharing system is vulnerable to the under Windows 95 famous dot dot bug (dot dot like ..). Meaning that anyone can crash the system. If someone sends a "DIR .." to the workstation will a STOP messages appear on the screen on the Windows NT computer. Note that it applies to version 3.50 and 3.51 for both workstation and server version.

The solution is to install the proper patch.

.C.18. SUNOS KERNEL PANIC

Some SunOS systems (running TIS?) will get a kernel panic if a getsockopt() is done after that a connection has been reset.

The solution could be to install Sun patch 100804.

.C.19. HOSTILE APPLETS

A hostile applet is any applet that attempts to use your system in an inappropriate manner. The problems in the java language could be sorted in two main groups:

- 1) Problems due to bugs.
- 2) Problems due to features in the language.

In group one we have for example the java bytecode verifier bug, which makes is possible for an applet to execute any command that the user can execute. Meaning that all the attack methods described in .D.X. could be executed through an applet. The java bytecode verifier bug was discovered in late March 1996 and no patch have yet been available (correct me if I'am wrong!!!).

Note that two other bugs could be found in group one, but they are both fixed in Netscape 2.01 and JDK 1.0.1.

Group two are more interesting and one large problem found is the fact that java can connect to the ports. Meaning that all the methods described in .C.X. can be performed by an applet. More information and examples could be found at address:

<http://www.math.gatech.edu/~mladue/HostileArticle.html>

If you need a high level of security you should use some sort of firewall for protection against java. As a user you could have java disable.

.C.20. VIRUS

Computer virus is written for the purpose of spreading and destroying systems. Virus is still the most common and famous denial of service attack method.

It is a misunderstanding that virus writing is hard. If you know assembly language and have source code for a couple of virus it is easy. Several automatic toolkits for virus construction could also be found, for example:

- * Genvir.
- * VCS (Virus Construction Set).
- * VCL (Virus Construction Laboratory).
- * PS-MPC (Phalcon/Skism - Mass Produced Code Generator).
- * IVP (Instant Virus Production Kit).
- * G2 (G Squared).

PS-MPC and VCL is known to be the best and can help the novice programmer to learn how to write virus.

An automatic tool called MtE could also be found. MtE will transform virus to a polymorphic virus. The polymorphic engine of MtE is well known and should easily be catch by any scanner.

.C.21. ANONYMOUS FTP ABUSE

If an anonymous FTP archive have a writable area it could be misused for a denial of service attack similar with with .D.3. That is we can fill up the hard disk.

Also can a host get temporarily unusable by massive numbers of FTP requests.

For more information on how to protect an anonymous FTP site could CERT:s "Anonymous FTP Abuses" be a good start.

.C.22. SYN FLOODING

Both 2600 and Phrack have posted information about the syn flooding attack. 2600 have also posted exploit code for the attack.

As we know the syn packet is used in the 3-way handshake. The syn flooding attack is based on an incomplete handshake. That is the attacker host will send a flood of syn packet but will not respond with an ACK packet. The TCP/IP stack will wait a certain amount of time before dropping the connection, a syn flooding attack will therefore keep the syn_received connection queue of the target machine filled.

The syn flooding attack is very hot and it is easy to find more information about it, for example:

[.1.] <http://www.eecs.nwu.edu/~jmyers/bugtraq/1354.html>

Article by Christopher Klaus, including a "solution".

[.2.] <http://jya.com/flood.txt>

2600, Summer, 1996, pp. 6-11. FLOOD WARNING by Jason Fairlane

[.3.] <http://www.fc.net/phrack/files/p48/p48-14.html>

IP-spoofing Demystified by daemon9 / route / infinity
for Phrack Magazine

.C.23. PING FLOODING

I haven't tested how big the impact of a ping flooding attack is, but it might be quite big.

Under Unix we could try something like: ping -s host
to send 64 bytes packets.

If you have Windows 95, click the start button, select RUN, then type
in: PING -T -L 256 xxx.xxx.xxx.xx. Start about 15 sessions.

.C.24. CRASHING SYSTEMS WITH PING FROM WINDOWS 95 MACHINES

If someone can ping your machine from a Windows 95 machine he or she might
reboot or freeze your machine. The attacker simply writes:

```
ping -l 65510 address.to.the.machine
```

And the machine will freeze or reboot.

Works for kernel 2.0.7 up to version 2.0.20. and 2.1.1. for Linux (crash).

AIX4, OSF, HPUX 10.1, DUnix 4.0 (crash).

OSF/1, 3.2C, Solaris 2.4 x86 (reboot).

.C.25. MALICIOUS USE OF SUBNET MASK REPLY MESSAGE

The subnet mask reply message is used under the reboot, but some
hosts are known to accept the message any time without any check.
If so all communication to or from the host is turned off, it's dead.

The host should not accept the message any time but under the reboot.

.C.26. FLEXIm

Any host running FLEXIm can get the FLEXIm license manager daemon
on any network to shutdown using the FLEXIm lmdown command.

```
# lmdown -c /etc/licence.dat
```

lmdown - Copyright (C) 1989, 1991 Highland Software, Inc.

Shutting down FLEXlm on nodes: xxx

Are you sure? [y/n]: y

Shut down node xxx

#

.C.27. BOOTING WITH TRIVIAL FTP

To boot diskless workstations one often use trivial ftp with rarp or bootp. If not protected an attacker can use tftp to boot the host.

.D. ATTACKING FROM THE INSIDE

~~~~~

### .D.1. KERNEL PANIC UNDER SOLARIS 2.3

-----

Solaris 2.3 will get a kernel panic if this is executed:

EX:

```
$ndd /dev/udp udp_status
```

The solution is to install the proper patch.

### .D.2. CRASHING THE X-SERVER

-----

If stickybit is not set in /tmp then can the file /tmp/.x11-unix/x0 be removed and the x-server will crash.

Ex:

```
$ rm /tmp/.x11-unix/x0
```

### .D.3. FILLING UP THE HARD DISK

-----

If your hard disk space is not limited by a quota or if you can use /tmp then it's possible for you to fill up the file system.

Ex:

```
while : ;  
mkdir .xxx  
cd .xxx  
done
```

### .D.4. MALICIOUS USE OF eval

-----  
 Some older systems will crash if eval '!!' is executed in the C-shell.

Ex:

```
% eval '!!'
```

#### .D.5. MALICIOUS USE OF fork() -----

If someone executes this C++ program the result will result in a crash on most systems.

Ex:

```
#include
#include
#include

main()
{
int x;
while(x=0;x -xxx
^C
$ ls
-xxx
$ rm -xxx
rm: illegal option -- x
rm: illegal option -- x
rm: illegal option -- x
usage: rm [-fiRr] file ...
$
```

Ex.II.

```
$ touch xxx!
$ rm xxx!
rm: remove xxx! (yes/no)? y
$ touch xxxxxxxxx!
$ rm xxxxxxxxx!
bash: !": event not found
$
```

(You see the size do count!)

Other well know methods is files with odd characters or spaces in the name.

These methods could be used in combination with ".D.3 FILLING UP THE HARDDISK". If you do want to remove these files you must use some sort

of script or a graphical interface like OpenWindow:s File Manager. You can also try to use: `rm ./`. It should work for the first example if you have a shell.

#### .D.7. DIRECTORY NAME LOOKUPCACHE

-----

Directory name lookupcache (DNLC) is used whenever a file is opened. DNLC associates the name of the file to a vnode. But DNLC can only operate on files with names that has less than N characters (for SunOS 4.x up to 14 character, for Solaris 2.x up 30 characters). This means that it's dead easy to launch a pretty discreet denial of service attack.

Create lets say 20 directories (for a start) and put 10 empty files in every directory. Let every name have over 30 characters and execute a script that makes a lot of `ls -al` on the directories.

If the impact is not big enough you should create more files or launch more processes.

#### .D.8. CSH ATTACK

-----

Just start this under `/bin/csh` (after proper modification) and the load level will get very high (that is 100% of the cpu time) in a very short time.

Ex:

```
|l /bin/csh
nodename : *****b
```

#### .D.9. CREATING FILES IN /tmp

-----

Many programs creates files in `/tmp`, but are unable to deal with the problem if the file already exist. In some cases this could be used for a denial of service attack.

#### .D.10. USING RESOLV\_HOST\_CONF

-----

Some systems have a little security hole in the way they use the `RESOLV_HOST_CONF` variable. That is we can put things in it and through ping access confidential data like `/etc/shadow` or crash the system. Most systems will crash if `/proc/kcore` is read in the variable and access through ping.

Ex:

```
$ export RESOLV_HOST_CONF="/proc/kcore" ; ping asdf
```

## .D.11. SUN 4.X AND BACKGROUND JOBS

-----

Thanks to Mr David Honig for the following:

" Put the string "a&" in a file called "a" and perform "chmod +x a".  
Running "a" will quickly disable a Sun 4.x machine, even disallowing  
(counter to specs) root login as the kernel process table fills."

" The cute thing is the size of the  
script, and how few keystrokes it takes to bring down a Sun  
as a regular user."

## .D.12. CRASHING DG/UX WITH ULIMIT

-----

ulimit is used to set a limit on the system resources available to the  
shell. If ulimit 0 is called before /etc/passwd, under DG/UX, will the  
passwd file be set to zero.

## .D.13. NETTUNE AND HP-UX

-----

/usr/contrib/bin/nettune is SETUID root on HP-UX meaning  
that any user can reset all ICMP, IP and TCP kernel  
parameters, for example the following parameters:

- arp\_killcomplete
- arp\_killincomplete
- arp\_unicast
- arp\_rebroadcast
- icmp\_mask\_agent
- ip\_defaultttl
- ip\_forwarding
- ip\_intrqmax
- pmtu\_defaulttime
- tcp\_localsubnets
- tcp\_receive
- tcp\_send
- tcp\_defaultttl
- tcp\_keepstart
- tcp\_keepfreq
- tcp\_keepstop
- tcp\_maxretrans
- tcp\_urgent\_data\_ptr
- udp\_cksum
- udp\_defaultttl
- udp\_newbcastenable
- udp\_pmtu
- tcp\_pmtu

- tcp\_random\_seq

The solution could be to set the proper permission on /sbin/mount\_union:

```
#chmod u-s /sbin/mount_union
```

#### .D.14. SOLARIS 2.X AND NFS

-----

If a process is writing over NFS and the user goes over the disk quota will the process go into an infinite loop.

#### .D.15. SYSTEM STABILITY COMPROMISE VIA MOUNT\_UNION

-----

By executing a sequence of mount\_union commands any user can cause a system reload on all FreeBSD version 2.X before 1996-05-18.

```
$ mkdir a
$ mkdir b
$ mount_union ~/a ~/b
$ mount_union -b ~/a ~/b
```

The solution could be to set the proper permission on /sbin/mount\_union:

```
#chmod u-s /sbin/mount_union
```

#### .D.16. trap\_mon CAUSES KERNEL PANIC UNDER SUNOS 4.1.X

-----

Executing the trap\_mon instruction from user mode can cause a kernel panic or a window underflow watchdog reset under SunOS 4.1.x, sun4c architecture.

#### .E. DUMPING CORE

~~~~~

.E.1. SHORT COMMENT

The core dumps things don't really belongs in this paper but I have put them here anyway.

.E.2. MALICIOUS USE OF NETSCAPE

Under Netscape 1.1N this link will result in a segmentation fault and a core dump.

Ex:

.F.1.7. MONITORING SECURITY

Also monitor security regular, for example through examining system log files, history files... Even in a system without any extra security systems could several tools be found for monitoring, for example:

- uptime
- showmount
- ps
- netstat
- finger

(see the man text for more information).

.F.1.8. KEEPING UP TO DATE

It is very important to keep up to date with security problems. Also understand that then, for example CERT, warns for something it has often been dark-side public for sometime, so don't wait. The following resources that helps you keeping up to date can for example be found on the Internet:

- CERT mailing list. Send an e-mail to cert@cert.org to be placed on the list.
- Bugtraq mailing list. Send an e-mail to bugtraq-request@fc.net.
- WWW-security mailing list. Send an e-mail to www-security@ns2.rutgers.edu.

.F.1.9. READ SOMETHING BIGGER AND BETTER

Let's start with papers on the Internet. I am sorry to say that it is not very many good free papers that can be found, but here is a small collection and I am sorry if have have over looked a paper.

(1) The Rainbow books is a long series of free books on computer security. US citizens can get the books from:

INFOSEC AWARENESS OFFICE
National Computer Security Center
9800 Savage Road
Fort George G. Meader, MD 20755-600

We other just have to read the papers on the World Wide Web. Every paper can not however be found on the Internet.

(2) "Improving the security of your Unix system" by Curry is also very

nice if you need the very basic things. If you don't know anything about computer security you can't find a better start.

(3) "The WWW security FAQ" by Stein is although it deal with W3-security the very best better on the Internet about computer security.

(4) CERT have also published several good papers, for example:

- Anonymous FTP Abuses.
- Email Bombing and Spamming.
- Spoofed/Forged Email.
- Protecting yourself from password file attacks.

I think however that the last paper have overlooked several things.

(5) For a long list on papers I can recommend:
"FAQ: Computer Security Frequently Asked Questions".

(6) Also see section ".G. SUGGESTED READING"

You should also get some big good commercial book, but I don't want to recommend any.

.F.2. MONITORING PERFORMANCE

.F.2.1. INTRODUCTION

There is several commands and services that can be used for monitoring performance. And at least two good free programs can be found on Internet.

.F.2.2. COMMANDS AND SERVICES

For more information read the man text.

- netstat Show network status.
- nfsstat Show NFS statistics.
- sar System activity reporter.
- vmstat Report virtual memory statistics.
- timex Time a command, report process data and system activity.
- time Time a simple command.
- truss Trace system calls and signals.
- uptime Show how long the system has been up.

Note that if a public netstat server can be found you might be able to use netstat from the outside. netstat can also give information like tcp sequence numbers and much more.

.F.2.3. PROGRAMS

Proctool: Proctool is a freely available tool for Solaris that monitors and controls processes.

<ftp://opcom.sun.ca/pub/binaries/>

Top: Top might be a more simple program than Proctool, but is good enough.

.F.2.4. ACCOUNTING

To monitor performance you have to collect information over a long period of time. All Unix systems have some sort of accounting logs to identify how much CPU time, memory each program uses. You should check your manual to see how to set this up.

You could also invent your own account system by using crontab and a script with the commands you want to run. Let crontab run the script every day and compare the information once a week. You could for example let the script run the following commands:

- netstat
- iostat -D
- vmstat

.G. SUGGESTED READING

~~~~~

#### .F.1. INFORMATION FOR DEEPER KNOWLEDGE

-----

- (1) Hedrick, C. Routing Information Protocol. RFC 1058, 1988.
- (2) Mills, D.L. Exterior Gateway Protocol Formal Specification. RFC 904, 1984.
- (3) Postel, J. Internet Control Message Protocol. RFC 792, 1981.
- (4) Harrenstien, K. NAME/FINGER Protocol, RFC 742, 1977.
- (5) Sollins, K.R. The TFTP Protocol, RFC 783, 1981.
- (6) Croft, W.J. Bootstrap Protocol, RFC 951, 1985.

Many of the papers in this category was RFC-papers. A RFC-paper is a paper that describes a protocol. The letters RCS stands for Request For Comment. Hosts on the Internet are expected to understand at least the common ones. If you want to learn more about a protocol it is always good to read the proper RFC. You can find a nice sRFC index search form at URL:

<http://pubweb.nexor.co.uk/public/rfc/index/rfc.html>

#### .F.2. KEEPING UP TO DATE INFORMATION

-----

- (1) CERT mailing list. Send an e-mail to [cert@cert.org](mailto:cert@cert.org) to be placed on the list.
- (2) Bugtraq mailinglist. Send an e-mail to [bugtraq-request@fc.net](mailto:bugtraq-request@fc.net).
- (3) WWW-security mailinglist. Send an e-mail to [www-security@ns2.rutgers.edu](mailto:www-security@ns2.rutgers.edu).
- (4) Sun Microsystems Security Bulletins.
- (5) Various articles from:
  - comp.security.announce
  - comp.security.unix
  - comp.security.firewalls
- (6) Varius 40Hex Issues.

### .F.3. BASIC INFORMATION

- 
- (1) Husman, H. INTRODUKTION TILL DATAS?KERHET UNDER X-WINDOWS, 1995.
  - (2) Husman, H. INTRODUKTION TILL IP-SPOOFING, 1995.
  - (3) The following rainbow books:
    - Teal Green Book (Glossary of Computer Security Terms).
    - Bright Orange Book( A Guide to Understanding Security Testing and Test Documentation in Trusted Systems).
    - C1 Technical Report-001 (Computer Viruses: Prevention, Detection, and Treatment).
  - (4) Ranum, Marcus. Firewalls, 1993.
  - (5) Sun Microsystems, OpenWindows V3.0.1. User Commands, 1992.
  - (6) Husman, H. ATT SP?RA ODOKUMENTERADE S?KERHETSLUCKOR, 1996.
  - (7) Dark OverLord, Unix Cracking Tips, 1989.
  - (8) Shooting Shark, Unix Nasties, 1988.
  - (9) LaDue, Mark.D. Hostile Applets on the Horizone, 1996.
  - (10) Curry, D.A. Improving the security of your unix system, 1990.
  - (11) Stein, L.D. The World Wide Web security FAQ, 1995.
  - (12) Bellovin, S.M. Security Problems in the TCP/IP Protocol, 1989.

This article comes from Linux Exposed

<http://www.linuxexposed.com>

The URL for this story is:

<http://www.linuxexposed.com/modules.php?op=modload&name=News&file=article&sid=545>