*** My Favourite UNIX Commands ***
*** A List Of Some OF The Most Useful UNIX **
*** Hacking Commands, and Some Hints On Their Usage ***
*** Written By ZeeBee Australia Inc. 1990 ***

Ok UNIX freaks, here is a little list of helpful UNIX commands and procedures which can be of much assistance in gaining the fullest potential from any UNIX system.

It is fun and often usefull to create a file that is owned by someone else. On most systems with slack security ie 99% of all UNIX systems, this is quite easily done. The chown command will change any of your files to make someone else the owner. Format is as follows:

chown ownername filelist

Where ownername is the new owner, and filelist is the list of files to change. You must own the file which your are goin to change, unless you are a superuser....then u can change ANYTHING!

chgrp is a similar command which will change the group ownership on a file. If you are going to do both a chown and a chgrp on a file, then make sure you do the chgrp first! Once the file is owned by someone else, you cant change nything about it!

Sometimes just seeing who is on the system is a challenge in itself. The best way is to write your own version of who in C, but if you can't do that then this may be of some help to you:

who followed by on or more of the following flags:

-b Displays time sys as last booted.
-H Precedes output with header.
-l Lists lines waiting for users to logon.
-q displays number of users logged on.
-t displays time sys clock was last changed.
-T displays the state field (a + indicates it is possible to send to terminal, a - means u cannot)
-u Give a complete listing of those logged on.

who -HTu is about the best choice for the average user

##by the way, the list of users logged on is kept in the file

/etc/utmp. If you want to write your own personalised version of who in C, you now know where to look!###

When a users state field (see -T flag option for who command) says that a user has their message function on, this actually means that it is possible to get stuff onto their screen.

Basically, every terminal on the system has a file corresponding to it. These files can be found in the /dev directory. You can to anything to these files, so long as you have access -eg you can read them, and write to them, but you will notice that they never change in size. They are called character specific files, and are really the link between the system and the terminals. Whatever you put in these files will go staright to the terminal it corresponds to.

Unfortunately, on most systems, when the user logs in, the "mesg n" command is issued which turns off write access to that terminal, BUT- if you can start cating to that terminal before system issues the mesg n command, then you will continue to be able to get stuff up on that terminal! This has many varied uses.

Check out the terminal, or terminal software being used. Often you will be able to remotely program another users terminal, simply by 'cating' a string to a users screen. You might be able to set up a buffer, capturing all that is typed, or you may be able to send the terminal into a frenzy- (sometimes a user will walk away without realizing that they are sill effectively logged on, leaving you with access to their account!). Some terminal types also have this great command called transmit screen. It transmits everything on the screen, just as if the user had typed it !

So just say I wanted to log off a user, then I would send a clear screen command (usually ctrl l), followed by "exit" followed by a carriage return, followed by the transmit screen code. Using ths technique you can wipe peoples directories or anything. My favourite is to set open access on all their files and directories so I can peruse them for deletion etc at my own leisure).

If you ever briefly get access to another persons account eg. they leave the room to go to toilet or whatever, then simply type the following:

```
chmod 777 $HOME
```

```
chmod 777 $MAIL
```

Then clear the screen so they dont see what you just typed.

Now you can go look at their directory, and their mail, and you can even put mail in their mail file. (just use the same format as any mail that is already there!). Next time they log in the system will automatically inform them they have new mail!

Another way to send fake mail to people is to use the mail server. This method produces mail that is slightly different to normal, so anyone who uses UNIX a bit may be suspiscious when they receive it, but it will fool the average dumb fuck user!

type telnet

the following prompt will appear:

telnet>

now type :

open localhost 25

some crap will come up about the mail server..now type:

mail from: xxxxxx Put any name you want.

some more bullshit will come up. Now type:

rcpt to: xxxxxx Put the name of the person to receive mail here.

now type:

data

now you can type the letter...end it with a "."
type quit to exit once you are done.

Heres one for any experimenters out there...
It is possible to create files which simply cannot be deleted from the standard shell. To do this you will have to physically CREATE THE FILE USING A C PROGRAM or SCRIPT FILE, and you will have to use a sequence of control characters which cannot be

typed from the shell. Try things like Ctrl-h (this is the code for the delete key). Just a file with the name Ctrl-h would not be deleteable from the shell, unless you used wildcards. So, make it a nice long series of characters, so that to delete the file, the user has no choice but to individually copy all his files elsewhere, then delete everything in his directory, and then copy all his files back.....this is one of my favourites..gets em every time!

     The following script file is an example which will create a file with the name Ctrl-h. You MUST tyoe this file in using the vi editor or similar.
*****If you are not very good with vi, type "man vi" and print the help file...it even contains stuff that I find useful now and then.*****

type the following in vi...

```
echo'' > 'a^h'
```

  ***NOTE...to get the ^h (this really means ctrl-h) from vi type:

```
Ctrl v
Ctrl h
```

  The Ctrl v instrcts vi to take the next character as a ascii character, and not to interpret it.
     change the access on the file you just created and now execute it. It will create a file which looks like it is called a, but try to delete it !..use wildcards if you really want to delete it.

Watch for more in the ZeeBee UNIX (ab)usage series.