*> Title:  Tutorial on hacking through a UNIX system

**

In the following file, all references made to the name Unix, may also be substituted to the Xenix operating system.

Brief history:  Back in the early sixties, during the development of third generation computers at MIT, a group of programmers studying the potential of computers, discovered their ability of performing two or more tasks simultaneously.  Bell Labs, taking notice of this discovery, provided funds for their developmental scientists to investigate into this new frontier.  After about 2 years of developmental research, they produced an operating system they called "Unix".

Sixties to Current:  During this time Bell Systems installed the Unix system to provide their computer operators with the ability to multitask so that they could become more productive, and efficient.  One of the systems they put on the Unix system was called "Elmos". Through Elmos many tasks (i.e. billing,and installation records) could be done by many people using the same mainframe.

Note: Cosmos is accessed through the Elmos system.

Current:  Today, with the development of micro computers, such multitasking can be achieved by a scaled down version of Unix (but just as powerful).  Microsoft,seeing this development, opted to develop their own Unix like system for the IBM line of PC/XT's.  Their result they called Xenix (pronounced zee-nicks).  Both

Unix and Xenix can be easily installed
on IBM PC's and offer the same function
(just 2 different vendors).

Note: Due to the many different
versions of Unix (Berkley Unix,
Bell System III, and System V
the most popular) many commands
following may/may not work. I have
written them in System V routines.
Unix/Xenix operating systems will
be considered identical systems below.

How to tell if/if not you are on a
Unix system:  Unix systems are quite
common systems across the country.
Their security appears as such:

Login;     (or login;)
password:

When hacking on a Unix system it is
best to use lowercase because the Unix
system commands are all done in lower-
case.
Login; is a 1-8 character field. It is
usually the name (i.e. joe or fred)
of the user, or initials (i.e. j.jones
or f.wilson).  Hints for login names
can be found trashing the location of
the dial-up (use your CN/A to find
where the computer is).
Password: is a 1-8 character password
assigned by the sysop or chosen by the
user.

Common default logins

| login; | Password: |
|---|---|
| root | root,system,etc.. |
| sys | sys,system |
| daemon | daemon |
| uucp | uucp |
| tty | tty |
| test | test |
| unix | unix |
| bin | bin |
| adm | adm |
| who | who |

```
   learn        learn
   uuhost       uuhost
   nuucp        nuucp

If you guess a login name and you are
not asked for a password, and have
accessed to the system, then you have
what is known as a non-gifted account.
If you guess a correct login and pass-
word, then you have a user account.
And, if you get the root p/w you have
a "super-user" account.
All Unix systems have the following
installed to their system:
root, sys, bin, daemon, uucp, adm
Once you are in the system, you will
get a prompt. Common prompts are:

$
%
#

But can be just about anything the
sysop or user wants it to be.

Things to do when you are in: Some
of the commands that you may want to
try follow below:

who is on  (shows who is currently
            logged on the system.)
write name (name is the person you
            wish to chat with)
      To exit chat mode try ctrl-D.
      EOT=End of Transfer.
ls -a      (list all files in current
            directory.)
du -a      (checks amount of memory
            your files use;disk usage)
cd\name    (name is the name of the
            sub-directory you choose)
cd\        (brings your home directory
            to current use)
cat name   (name is a filename either
            a program or documentation
            your username has written)
      Most Unix programs are written
      in the C language or Pascal
```

since Unix is a programmers' environment.

One of the first things done on the system is print up or capture (in a buffer) the file containing all user names and accounts. This can be done by doing the following command:

```
cat /etc/passwd
```

If you are successful you will see a list of all accounts on the system. It should look like this:

```
root:hvnsdcf:0:0:root dir:/:
joe:majdnfd:1:1:Joe Cool:/bin:/bin/joe
hal::1:2:Hal Smith:/bin:/bin/hal
```

Te "root" line tells the following info :

```
login name=root
hvnsdcf   = encrypted password
0         = user group number
0         = user number
root dir  = name of user
/         = root directory
```

In the Joe login, the last part "/bin/joe " tells us which directory is his home directory (joe) is.

In the "hal" example the login name is followed by 2 colons, that means that there is no password needed to get in using his name.

Conclusion: I hope that this file will help other novice Unix hackers obtain access to the Unix/Xenix systems that they may find.