

screw everything...the main level (exec level) prompt on the unix system is the \$, and if you are on the root, you have a # (superuser prompt).
Ok, a few basics for the system... To see where you are, and what paths are active in regards to your user account, then type

=> pwd

This shows your acct. separated by a slash with another pathname (acct.), possibly many times. To connect through to another path, or many paths, you would type:

You=> path1/path2/path3

And then you are connected all the way from path1 to path3. You can run the programs on all the paths you are connected to. If it does not allow you to connect to a path, then you have insufficient privs, or the path is closed and archived onto tape. You can run programs this way also:

you=> path1/path2/path3/program-name

Unix treats everything as a program, and thus there a few commands to learn...
To see what you have access to in the end path, type:

ls

for list. This show the programs you can run. You can connect to the root directory and run it's programs with=>

/root

By the way, most unix systems have their log file on the root, so you can set up a watch on the file, waiting for people to log in and snatch their password as it passes thru the file. To connect to a directory, use the command:

=> cd pathname

This allows you to do what you want with that directory. You may be asked for a password, but this is a good way of finding other user names to hack onto. The wildcard character in unix, if you want to search down a path for a game or such, is the *.

=> ls /*

Should show you what you can access. The file types are the same as they are on a dec, so refer to that section when examining file. To see what is in a file, use the

=> pr

filename command, for print file. We advise playing with pathnames to get the hang of the concept. There is on-line help available on most systems with a 'help' or a '?'. We advise you look thru the help files and pay attention to anything they give you on pathnames, or the commands for the system. You can, as a user, create or destroy directories on the tree beneath you. This means that root can kill everything but root, and you can kill any that are below you. These are the

=> mkdir pathname

=> rmdir pathname

commands. Once again, you are not alone on the system... type=>

who

to see what other users are logged in to the system at the time. If you want to talk to them=>

write username

Will allow you to chat at the same time, without having to worry about the parser. To send mail to a user, say

=> mail

And enter the mail sub-system. To send a message to all the users on the system, say

=> wall