onion soup

UNIX crash course

The following screed contains a relatively painless romp through the sometimes painful terrain of UNIX commands. My intent is to describe a simple (though comprehensive) presentation of the basic, UNIX commandline. The commands - below - are generic; that is to say, they *should* all be valid and execute on *most* modern, UNIX operating systems. (In the future, this page may be expanded to include networking.)

home

Navigating a filesystem, listing files and manipulating them

Display the absolute PATH of the current working directory.

```
 $ pwd
```

Display the absolute path of an executable, named: "rm".

```
 $ which rm
```

Change directory to /home/doctor/medical_abnormalities - using the absolute PATH.

```
 $ cd /home/doctor
 /medical_abnormalities
```

Change from the current working directory to its sub-directory, named: "music" - using the relative path.

```
 $ cd music
```

Change the working directory to $HOME.

```
 $ cd ~/
```

List all non-hidden files and directories in the current working directory.

```
 $ ls
```

Altering permissions and ownerships

"chmod" is employed to alter the read, write and execute permissions on files and directories. The breakdown of the absolute (numerical) mode for the permissions of files and directories is as follows: the first field signifies the owner, the second field the group and the third field all others, i.e. world.

(owner)(group)(other)

read = 4
write = 2
execute = 1

So - by placing 755 on a file means that the owner has read, write and execute permissions, while the group and other have only read and execute permissions on that file. Three examples follow:

The numerical bits, 644, signify that the owner of index.html will be able to write to and read it, all members of the owner's group will be able to read it, and all other system users will be able to read it.

home

```
 $ chmod 644 index.html
```

After the following command is executed, the owner will have read, write and execute permissions, the group will have read and execute permissions, and all others will have execute permissions over index.cgi.

```
 $ chmod 751 index.cgi
```

Change the permissions on the www directory to user read, write and execute, group execute and other execute.

Archiving data

Compress the frauds file with tar to create frauds.tar.

```
$ tar cvf frauds.tar frauds
```

Extract the contents of the frauds.tar file.

```
$ tar xvf frauds.tar
```

Extract the crimes.tar.gz file.

```
$ tar xzvf crimes.tar.gz
```

Extract the diseases.tgz file.

```
$ tar xzvf diseases.tgz
```

Compress music.tar with the highest level of **gzip** compression. After executing the command, the file created will be "music.tar.gz".

```
$ gzip -9 music.tar
```

Decompress hack.gz.

```
$ gzip -d hack.gz
```

Extract the entire pr0n archive, which has the .tar.bz2 extension.

```
$ tar xjvf pr0n.tar.bz2
```

Miscellaneous utilities

Search for and display a string of characters "Drone Assassination" in "cia.html" with **grep**. Below, "-i" = case insensitive.

```
$ grep -i "Drone Assassination"
cia.html
```

Search for and display "Charles Manson" in every file residing under the "prison" directory.

```
$ grep -r "Charles Manson"
prison \;
```

Determine which partitions are currently mounted and print their sizes in a human, understandable format.

home

```
$ df -h
```

Print the size of the contents of the "cracked" directory in a human, understandable format.

```
$ du -sh cracked
```

Change the password. The user will be prompted for his current password, a new password and the confirmation of the new password.

```
$ passwd
```

links

TORring the net

Nginx+Tor

Apache+Tor

Lighttpd+Tor

home