

# Cyber Bulls Session 1 – Linux Fundamentals

This session will teach the basic Linux command line skills that shall be required to understand and perform the exploits used in later sessions. You will be walked through the basic command syntax before having to think for yourself to solve some common problems by using command manipulation. There is an optional extension on scripting for those already comfortable with their command line skills.

## Stage one – Basic command line skills

Navigate to <http://www.webminal.org/> and create an account on the site (you will need to verify your email address to do this). We will be using this site as a terminal emulator so that we can play with the command line in a safe environment.

- Open up the terminal page and log in to the terminal using the username and password you chose when creating your account.
- Work through lessons 1 to 5 (found on the drop down box on right of terminal page) at your own pace, inputting the commands to the terminal as you go.

## Stage two – Manipulating input and output

The commands we have learnt provide solid building blocks, but we need to link them together to use them to their full potential. The pipe character "|" allows us to do this by passing the output of the command before the pipe into the input of the command after it. If we want to redirect that output to a file instead of another command we can use the ">" character. Pipes can be chained together indefinitely to provide complete control when running commands.

Example 1: **ps | grep http** will list the processes running and then provide that as input to a search for lines containing http.

Example 2: **ls > files.txt** will list all files in the working directory and then save that list in a file called files.txt

- Use the tools above to sort all files in your home directory in alphabetical order and save the result in a text file (as one command)
- Wouldn't it be nice to know how many total processes were running? Get a list of **all** the processes running and use the pipe tool to get a line count of the result. Redirect this into a suitably named file.
- The history command displays all the commands we have run in the current session. Find a way to save in one file the number of times the ls and cd commands were run.
  - Too easy? Save separate counts for ls and cd respectively while still performing the entire task with a single chained command.

## Stage three (extension work) - Scripting

The problem with entering commands on a command line is that it can be a laborious and repetitive task. Wouldn't it be nicer if we could write the commands once and then re-use them?

- Work through the scripting intro lesson on Webminal to gain familiarity with the syntax involved in shell scripts
- Write an archiving script that will move any files in the user's home directory into an archive folder (deleting the original in the home directory afterwards).

- Too easy? Use the jobs scheduling utilities in Linux to get this script to run hourly
- Write a script that keeps a running tally of all processes that are running. The script will need to load up the old tally each time it is run and increment it with the current processes.
  - Set this to run every hour so that the system admin can see the most common processes being used over time and spot anything outside the ordinary.