
# Linux & Bash

# Legal

The skills taught in these sessions allow identification and exploitation of security vulnerabilities in systems. We strive to give you a place to practice legally, and can point you to other places to practice. These skills should not be used on systems where you do not have explicit permission from the owner of the system. It is VERY easy to end up in breach of relevant laws, and we can accept no responsibility for anything you do with the skills learnt here.

If we have reason to believe that you are utilising these skills against systems where you are not authorised you will be banned from our events, and if necessary the relevant authorities will be alerted.

Remember, if you have any doubts as to if something is legal or authorised, just don't do it until you are able to confirm you are allowed to.

# Code of Conduct

Before proceeding past this point, you must read and agree to our Code of Conduct - this is a requirement from the University for us to operate as a society.

If you have any doubts or need anything clarified, please ask a member of the committee.

Breaching the Code of Conduct = immediate ejection and further consequences.

Code of Conduct can be found at
https://shefesh.com/downloads/SESH%20Code%20of%20Conduct.pdf

# Contents



# WHAT IS LINUX?

Linux is everywhere, because of its ability to be very small, very customisable and the ability to run on many devices.

## KERNEL

It is a product of UNIX

- Small and modular
- Same family of macOS
- ‘Everything is a file’

## UNIX

- Anyone can read the source code for Linux
- Anyone can contribute to the Linux project
- Windows and macOS are proprietary so cannot see their source code

## OPEN SOURCE

# DISTRIBUTIONS

As Linux is open source and anyone can edit it, there are multiple “distributions” by different maintainers you can download.

There are many distributions to choose from, you just need to have a look around to see what suits your needs

## Debian

Intermediate level. Highly stable, popular in servers. A popular distros based on Debian include:

- Ubuntu (easy)
- Kali Linux (easy)
- Linux Mint (easy)

## Arch Linux

Rolling-release, intermediate/expert user base. Very customisable.

## Fedora

Semi-rolling release. Easy to use enterprise grade distribution. Used in RHEL, backed by Red Hat.

- Workstation
- Server
- etc...

# BASH

## Versatile

It offers powerful control over files, software and can help you automate.

You can control anything and everything from BASH.

## Control Interface

BASH is an extension of SH and is POSIX compliant.

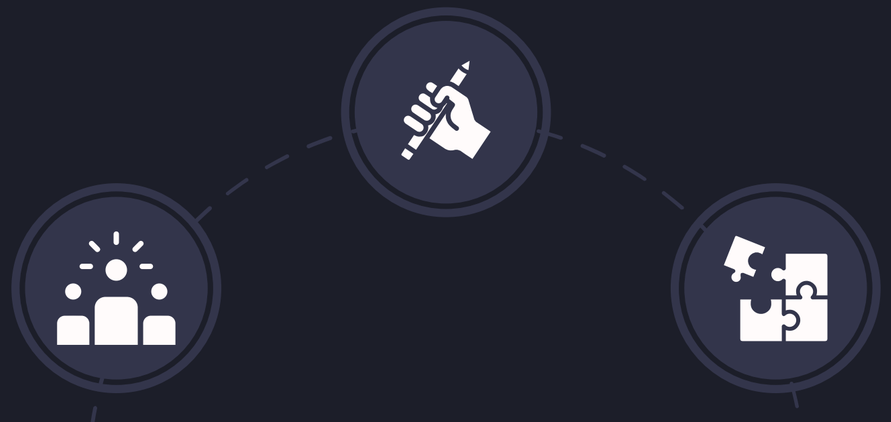
It is a text-driven way to control systems.

You use it to communicate with the computer rather than buttons in a GUI.

## Efficiency

It can be customized to very quick and efficient to perform tasks.

You can set aliases



# Navigation Commands

| | | |
|---|---|---|
| `cd` | change directory | `cd /home/user` |
| `ls` | list subdirectories | `ls /home/user` |
| `mv` | move files | `mv /home/user/old /home/user/new` |
| `cp` | copy files | `cp /home/user/source /home/user/dest` |
| `rm` | remove files | `rm /home/user/file` |
| `cat` | output file content | `cat /home/user/file` |
| `touch` | create a file | `touch /home/user/new_file` |
| `grep` | search for strings in files | `grep “hello world” /home/user/test` |
| `find` | find files in system | `find / -name user.txt -type f` |
| `locate` | find files in system (easier) | `locate hosts` |
| `nano` | Edit a text file | `nano /home/user/file` |
| `sudo/su` | Execute code as another user | `sudo ls /root OR su -user` |

# CONNECT

Connect from Command Prompt or Powershell

Command

```
ssh user[number]@18.170.42.238 -p 2222
```

Password

```
SESH_LINUX
```

## TASKS

1. What is in the file fruit.txt?
2. What file is in the folder “tree”?
3. What distribution and version of Linux are you using?
4. Can you move the file “rename.txt” to “task_list.txt”
5. Edit the file “task_list.txt” and add apple and banana to it.

# Folder structure

| | |
|---|---|
| `/bin` | Used for essential binaries (applications) |
| `/etc` | ‘etcetera’, contains configuration files |
| `/home` | Contains the home directory of users |
| `/root` | Contains the home directory of the root user |
| `/opt` | Optional software, software that generally isn’t maintained by a package manager |
| `/usr` | User related binaries and program data |
| `/sbin` | Binaries to be run as a root user (run with sudo) |
| `/var` | Stores variable application data like logs and temporary files |



# PIPING AND REDIRECTION

We can run commands in the terminal and get its output. We can also chain multiple commands to get “processed” output.

## PIPING

```
❯ cat /etc/os-release | rg "NAME"
NAME="Arch Linux"
PRETTY_NAME="Arch Linux"
```

Piping commands allows us to use output from a command as the input to the next command

## REDIRECTION

```
[mizuuu@pathos ~]$ find / -name lost_file.txt
find: ‘/usr/lib/security/howdy’: Permission denied
find: ‘/usr/share/polkit-1/rules.d’: Permission denied
find: ‘/usr/NX/scripts/log’: Permission denied
```

```
[mizuuu@pathos ~]$ find / -name lost_file.txt 2>/dev/null
/home/mizuuu/lost_file.txt
```

Redirection helps to get rid of unwanted output

There are 3 different outputs in the terminal.

| | | |
|---|---|---|
| • stdin | • 0 | • Standard input |
| • stdout | • 1 | • Standard output |
| • stderr | • 2 | • Standard error |

# PERMISSIONS

## Group

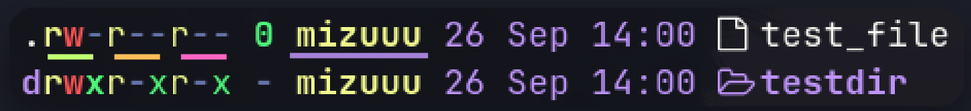
Permissions for the members of group.

## User

Permissions if they are logged in as that user.

## Other

If they are neither in the member nor logged in as user.





## User/Group

The specified user/group for the permissions.

# MODIFYING PERISSIONS

## CHOWN

### CHANGE OWNERSHIP

The chown command allows you to change the user and/or group ownership of a file or directory

```
chown USER FILE
chown USER:GROUP FILE
```

## CHMOD

### CHANGE MODE

The chmod command to change the access permissions of files and directories.

- r read 4
- w write 2
- x execute 1

```
chmod [OPTIONS] MODE FILE
chmod NUMBER FILE
chmod 600 FILE
```

# SCRIPTING

You can put bash commands in a file and have them run in order.

- They usually have the .sh extension
- They will start with #!/bin/bash
- You need to have execute permissions for a script to run it

## EXAMPLE

test_script.sh

```
#!/bin/bash
echo “Hello World”
```

```
~
❯ chmod +x test_script.sh

~
❯ ./test_script.sh
Hello world
```

## EDITING FILES

```
[user@host ~] nano test_script
```

After entering the file contents press `CTRL + x` → `y` → `ENTER`

We can edit files from the command line using tools like nano and vi/vim. We will be using nano for now.

But I recommend learning how to use vim. It is a powerful modal based text editor.

# CHALLENGES



1. What version of grep is installed?
2. What command returns the date on Linux?
3. Can you redirect the output of the date command to date.txt?
4. Make and run a script that prints out your name.
5. What command can I run to print the username of the logged-in user?
6. Can you create an alias command to run the script you made in challenge 4?
7. Create a script prints the time every 5 seconds. Can you quit out of the loop? (CTRL +C)
8. Can you make it so that the script is running in the background?
9. Can you kill the script thats running in the background?

Some of these might require you to look up how to do these.

# BONUS: PACKAGE MANAGERS

Installing applications or packages on Linux typically uses the distribution's package manager.

Instead of going to a website and downloading and executing binaries(.exe .dmg) you use the command line to search, install and upgrade them.

I will give some example for Ubuntu(Debian) as it is the most popular WSL distribution.

| | |
|---|---|
| `sudo apt update` | update package repositories |
| `sudo apt upgrade` | upgrade packages |
| `apt list --installed` | list installed packages |
| `sudo apt install coreutils` | install a package |

Message us for inquiries.

# Upcoming Sessions

2nd October: Introduction to Linux
9th October: OSINT/Reconnaissance
16th October: Enumeration
???: Regional Cybercrime Unit Guest Talk