**TryHackMe Journal - [Name]**

| **Instructions**  1. Review the sample journal entry provided below 2. Scroll down to find the name of the room you have been assigned/are working on   (Pro Tip: Turn on “Outline View” so you can navigate more easily - go to View -> Show Outline)   1. Complete the required rooms on TryHackMe, compiling notes as you work through the room. This might include:    1. Commonly used Code/Commands    2. Definitions/Explanations of important terms and concepts    3. Screenshots of useful diagrams 2. Once you’ve completed the module, capture 2-4 important takeaways. 3. After you get the hang of things, delete these instructions and the sample you were provided! |
| --- |

[Entry 1- SAMPLE](#_23drygy33cs7)

[Room Name: Linux Fundamentals 1](#_9vvz4g4ibayw)

[Entry 1](#_1hsl3npa9rfw)

[Room Name: Linux Fundamentals 1](#_qu0yyb2ejc0w)

[Entry 2](#_rssbp32e2g8m)

[Room Name: Linux Fundamentals 2](#_638z74u39hd6)

[Entry 3](#_cltwrfvb2owe)

[Room Name: Linux Fundamentals 3](#_2kl4qua7c8a1)

[Entry 4](#_2egcbux2xso7)

[Room Name: Linux Strength Training](#_5f5qjok47fjb)

[Entry 5](#_str0vgf5d80)

[Room Name: Intro to Logs](#_9ozf7tyhj2z1)

[Entry 6](#_ak1pof8ebyk6)

[Room Name: Wireshark Basics](#_w7wl7e3uwrrn)

[Entry 7](#_nzyixrmzwseg)

[Room Name: Wireshark 101](#_zdodqul745hz)

[Entry 8](#_wus2efos6bcc)

[Room Name: Windows Fundamentals 1](#_ltkwqp4he38n)

[Entry 9](#_fregcgt9agb0)

[Room Name: Windows Fundamentals 2](#_wruho7cdogl4)

[Entry 10](#_elanzhcb97j)

[Room Name: Windows Fundamentals 3](#_95wrsci0dg9e)

[Entry 11](#_g86u54ixvozg)

[Room Name: Windows Forensics 1](#_xfv515m7w632)

[Entry 12](#_atdks5ge6rk5)

[Room Name: Windows Forensics 2](#_qhosa2k0l7a6)

[Entry 13](#_v6k9voxxs4zc)

[Room Name: Intro to Log Analysis](#_1ws0odcehb3h)

[Entry 14](#_ihmevfqj73au)

[Room Name: Splunk Basics](#_8thwcuu7w9t9)

[Entry 15](#_v4ppfi3lwy10)

[Room Name: Incident Handling with Splunk](#_bm0uzj3i092e)

[Entry 16](#_3g1yj8gcyfez)

[Room Name: Splunk 2](#_wnal444q34qr)

[Entry 17](#_hb9jz3666q35)

[Room Name: Splunk 3](#_god1fagavfvg)

## Entry 1- SAMPLE

### **Room Name**: Linux Fundamentals 1

**Date Completed**: 12/20/2023

**Notes During the Room**:

* Similar to how you have different versions of Windows (7, 8 and 10), there are many different versions/distributions of Linux.

| Command | Description |
| --- | --- |
| echo | Output any text that we provide |
| whoami | Find out what user we're currently logged in as! |

| Command | Full Name |
| --- | --- |
| ls | listing |
| cd | change directory |
| cat | concatenate |
| pwd | print working directory |

| Symbol / Operator | Description |
| --- | --- |
| & | This operator allows you to run commands in the background of your terminal. |
| && | This operator allows you to combine multiple commands together in one line of your terminal. |
| > | This operator is a redirector - meaning that we can take the output from a command (such as using cat to output a file) and direct it elsewhere. |
| >> | This operator does the same function of the > operator but appends the output rather than replacing (meaning nothing is overwritten). |

**Important Takeaways**

* Linux is an OS, like Windows. There are many different versions of Linux that serve different purposes.
* Linux systems rely more heavily on the command line to do tasks, like navigate the file system.
* Same basic commands while working with files are ls, cd, cat and pwd

## Entry 1

### **Room Name:** Linux Fundamentals 1

**Date Completed**: 4/28/2024

**Notes During the Room**:

* Essential commands used to interact with the file system
* Introduced to how users and groups work on Linux

| Command | Description |
| --- | --- |
| echo | Output any text that we provide |
| whoami | Find out what user we're currently logged in as! |

| Command | Full Name |
| --- | --- |
| ls | listing |
| cd | change directory |
| cat | concatenate |
| pwd | print working directory |

The grep command allows us to search the contents of files for specific values that we are looking for.

| Symbol / Operator | Description |
| --- | --- |
| & | This operator allows you to run commands in the background of your terminal. |
| && | This operator allows you to combine multiple commands together in one line of your terminal. |
| > | This operator is a redirector - meaning that we can take the output from a command (such as using cat to output a file) and direct it elsewhere. |
| >> | This operator does the same function of the > operator but appends the output rather than replacing (meaning nothing is overwritten). |

**Important Takeaways**:

* Linux is an OS, like Windows. There are many different versions of Linux that serve different purposes.
* Linux systems rely more heavily on the command line to do tasks, like navigate the file system.
* Same basic commands while working with files are ls, cd, cat and pwd

## Entry 2

### **Room Name**: Linux Fundamentals 2

**Date Completed**: 4/28/2024

**Notes During the Room**:

* SSH allows us to remotely execute commands on another device remotely.
* Any data sent between the devices is encrypted when it is sent over a network such as the Internet

Arguments are identified by a hyphen and a certain keyword known as flags or switches.

ls lists the contents of the working directory. However, hidden files are not shown. We can use flags and switches to extend the behaviour of commands.

Files and folders with "**.**" are hidden files.

The –help option is a formatted output of what is called the man page (short for manual), which contains documentation for Linux commands and applications.

| Command | Full Name | Purpose |
| --- | --- | --- |
| touch | touch | Create file |
| mkdir | make directory | Create a folder |
| cp | copy | Copy a file or folder |
| mv | move | Move a file or folder |
| rm | remove | Remove a file or folder |
| file | file | Determine the type of a file |

*Protip: Similarly to using cat, we can provide full file paths, i.e. directory1/directory2/note for all of these commands*

You can simply remove files by using rm. However, you need to provide the -R switch alongside the name of the directory you wish to remove.

Starting with cp, this command takes two arguments:

1. the name of the existing file

2. the name we wish to assign to the new file when copying

cp copies the entire contents of the existing file into the new file.

Moving a file takes two arguments, just like the cp command. However, rather than copying and/or creating a new file, mv will merge or modify the second file that we provide as an argument. Not only can you use mv to move a file to a new folder, but you can also use mv to rename a file or folder.

This root directory is one of the most important root directories on your system. The etc folder (short for etcetera) is a commonplace location to store system files that are used by your operating system

The "/var" directory, with "var" being short for variable data, is one of the main root folders found on a Linux install. This folder stores data that is frequently accessed or written by services or applications running on the system. For example, log files from running services and applications are written here (**/var/log**), or other data that is not necessarily associated with a specific user (i.e., databases and the like).

Unlike the **/home** directory, the **/root** folder is actually the home for the "root" system user. There isn't anything more to this folder other than just understanding that this is the home directory for the "root" user. But, it is worth a mention as the logical presumption is that this user would have their data in a directory such as "**/home/root**" by default.

**/tmp**

This is a unique root directory found on a Linux install. Short for "temporary", the /tmp directory is volatile and is used to store data that is only needed to be accessed once or twice. Similar to the memory on your computer, once the computer is restarted, the contents of this folder are cleared out.

**Important Takeaways**:

**S**ecure **S**hell or **SSH** for short and is the common means of connecting to and interacting with the command line of a remote Linux machine.

Arguments are identified by a hyphen and a certain keyword known as flags or switches.

The manual pages are a great source of information for both system commands and applications available on both a Linux machine, which is accessible on the machine itself and [online](https://linux.die.net/man/).

## Entry 3

### **Room Name**: Linux Fundamentals 3

**Date Completed**: 04/28/2024

**Notes During the Room**:

To create or edit a file using nano, we simply use nano filename -- replacing "filename" with the name of the file you wish to edit.You can navigate each line using the "up" and "down" arrow keys or start a new line using the "Enter" key on your keyboard. You can use these features of nano by pressing the "**Ctrl**" key (which is represented as an ^ on Linux) and a corresponding letter.

VIM is a much more advanced text editor

Some of VIM's benefits, albeit taking a much longer time to become familiar with, includes:

* Customisable - you can modify the keyboard shortcuts to be of your choosing
* Syntax Highlighting - this is useful if you are writing or maintaining code, making it a popular choice for software developers
* VIM works on all terminals where nano may not be installed
* There are a lot of resources such as [cheatsheets](https://vim.rtorr.com/), tutorials, and the sorts available to you use.

wget allows us to download files from the web via HTTP -- as if you were accessing the file in your browser.

Secure copy, or SCP,allows you to:

* Copy files & directories from your current system to a remote system
* Copy files & directories from a remote system to your current system

ps command to provide a list of the running processes as our user's session and some additional information such as its status code, the session that is running it, how much usage time of the CPU it is using, and the name of the actual program or command that is being executed:

To see the processes run by other users and those that don't run from a session (i.e. system processes), we need to provide **aux** to the ps command like so: ps aux

You can send signals that terminate processes; there are a variety of types of signals that correlate to exactly how "cleanly" the process is dealt with by the kernel. To kill a command, we can use the appropriately named kill command and the associated PID that we wish to kill.

Some of the signals that we can send to a process when it is killed:

* SIGTERM - Kill the process, but allow it to do some cleanup tasks beforehand
* SIGKILL - Kill the process - doesn't do any cleanup after the fact
* SIGSTOP - Stop/suspend a process

**Systemd** is used to provide a way of managing a user's processes and sits in between the operating system and the user.

Enter the use of systemctl -- this command allows us to interact with the **systemd** process/daemon

We can do four options with systemctl:

* Start
* Stop
* Enable
* Disable

Crontab is one of the processes that is started during boot, which is responsible for facilitating and managing cron jobs.

A crontab is simply a special file with formatting that is recognised by the cron process to execute each line step-by-step. Crontabs require 6 specific values:

| Value | Description |
| --- | --- |
| MIN | What minute to execute at |
| HOUR | What hour to execute at |
| DOM | What day of the month to execute at |
| MON | What month of the year to execute at |
| DOW | What day of the week to execute at |
| CMD | The actual command that will be executed. |

If we do not wish to provide a value for that specific field, i.e. we don't care what month, day, or year it is executed -- only that it is executed every 12 hours, we simply just place an asterisk.

The apt command is a part of the package management software also named apt. Apt contains a whole suite of tools that allows us to manage the packages and sources of our software, and to install or remove software at the same time.

**Important Takeaways**: Commonly used terminal text editors are Nano and VIM.

The logs for services such as a web server contain information about every single request - allowing developers or administrators to diagnose performance issues or investigate an intruder's activity

## Entry 4

### **Room Name**: Linux Strength Training

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 5

### **Room Name**: Intro to Logs

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 6

### **Room Name**: Wireshark Basics

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 7

### **Room Name**: Wireshark 101

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 8

### **Room Name**: Windows Fundamentals 1

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 9

### **Room Name**: Windows Fundamentals 2

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 10

### **Room Name**: Windows Fundamentals 3

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 11

### **Room Name**: Windows Forensics 1

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 12

### **Room Name**: Windows Forensics 2

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 13

### **Room Name**: Intro to Log Analysis

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 14

### **Room Name**: Splunk Basics

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 15

### **Room Name**: Incident Handling with Splunk

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 16

### **Room Name**: Splunk 2

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**:

## Entry 17

### **Room Name**: Splunk 3

**Date Completed**:

**Notes During the Room**:

**Important Takeaways**: