

## **Practical 1**

There are three main goals in our first computer practical.

- Installing a virtual machine platform;
- Installing the virtual machine to use Hadoop, already installed on the Linux operating system;
- Learning some basic Linux commands.

### **Part 0 - Downloading the VM**

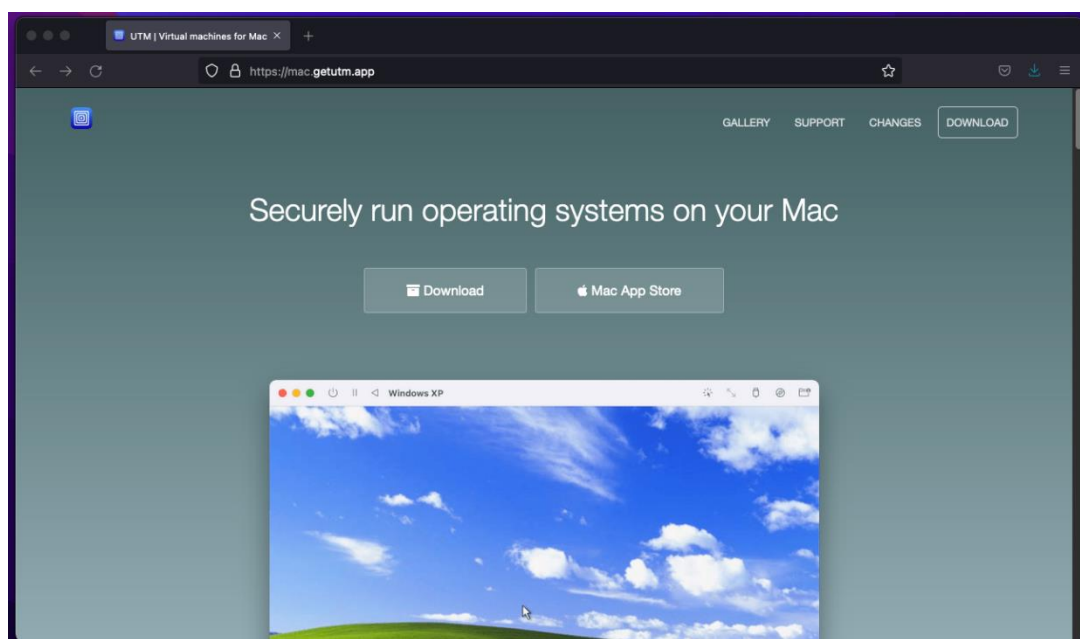
Please take the time to download the file at the link below prior to starting this practical. It is approximately 4 GB's in size and depending on your internet connection can take a while. The virtual machine links have been provided in the course website. You can simply leave the file in your downloads folder and unzip it there; you don't need to do anything with it yet. But a couple of things to note:

1. If you do not have a stable internet connection, such a large file is easily susceptible to corruption during download. If this is the case for you, you can usually install a download manager add-on for the browser of your choice to make sure the file downloads correctly.
2. The file can be unzipped where it downloads, but some unzipping tools may struggle with the file size. 7-Zip is a good choice of unzipper and may be worth a look if you encounter errors unzipping.

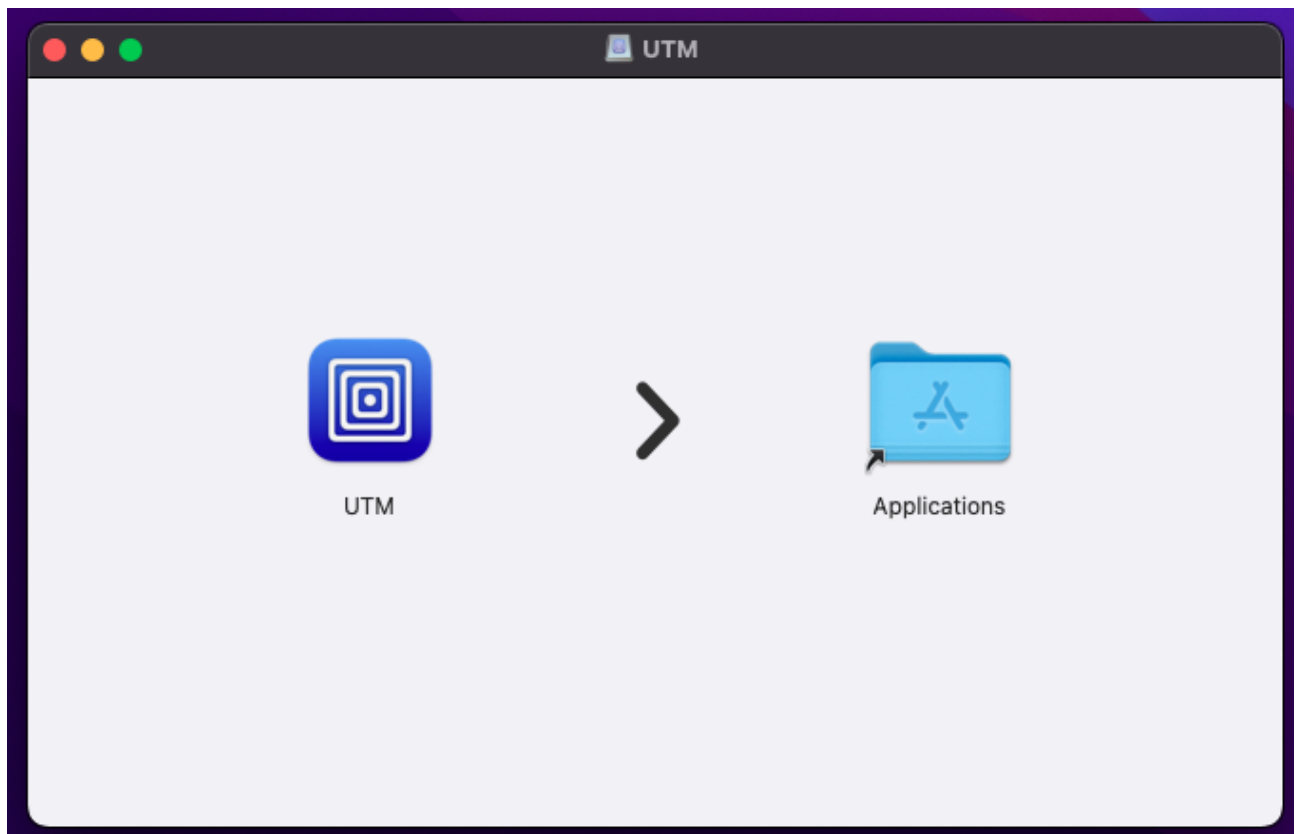
### **Part 1 - Installing UTM (Mac M1)**

1. Download the UTM application

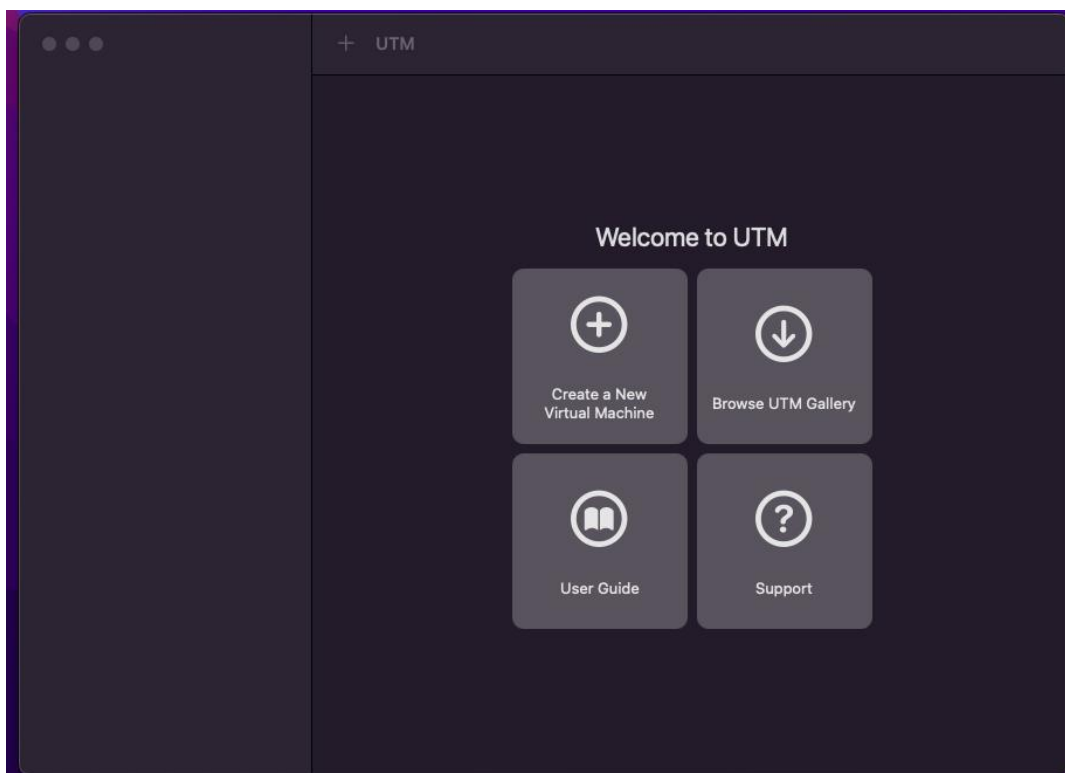
<https://mac.getutm.app/>



2. Double click to open and drag the file to applications

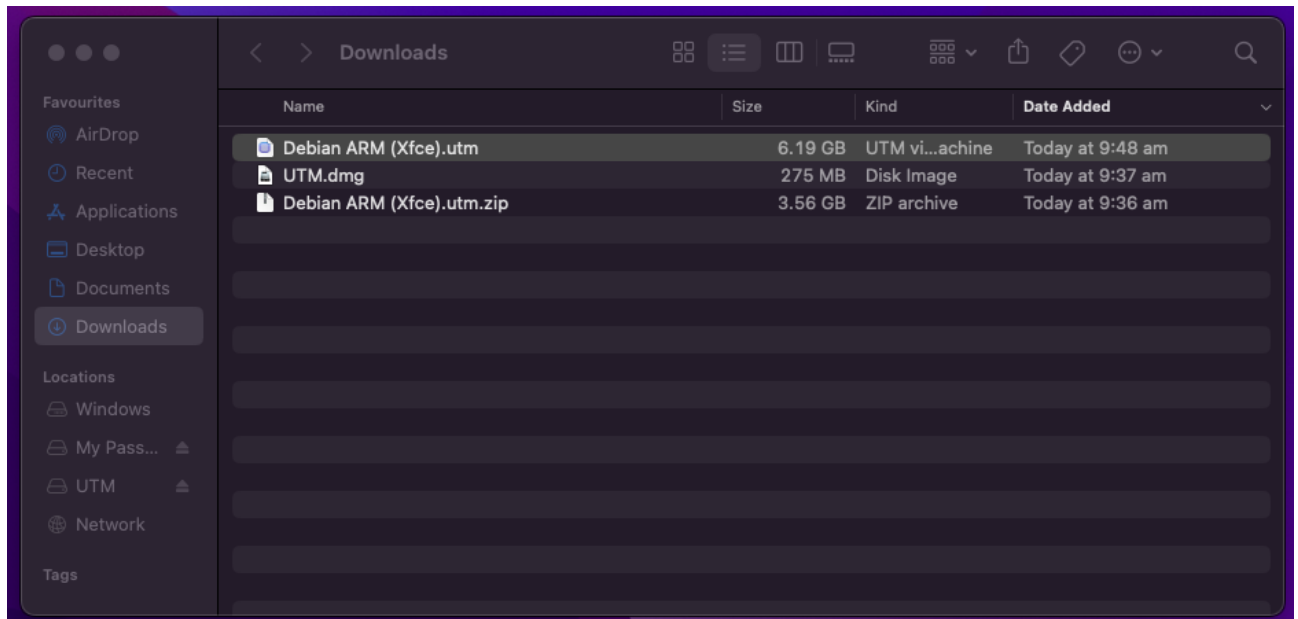


3. Open the UTM application



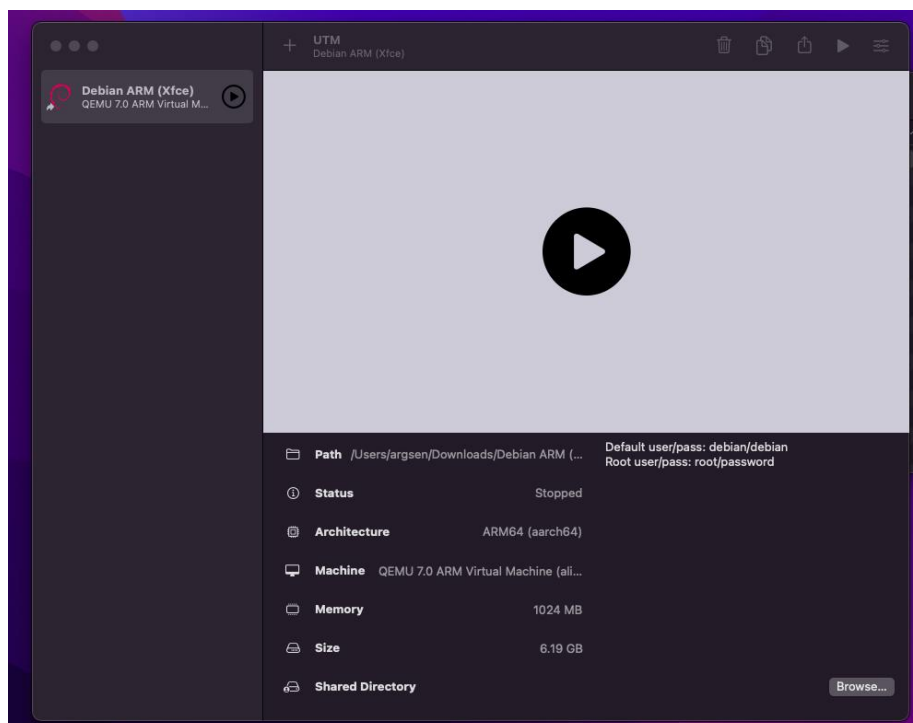
## Part 2 – Import the Mac M1 UTM Hadoop virtual machine

### 1. Unzip the file after downloading



### 2. Double click the Debian ARM (Xfce).utm

UTM will import the virtual machine and start the virtual machine



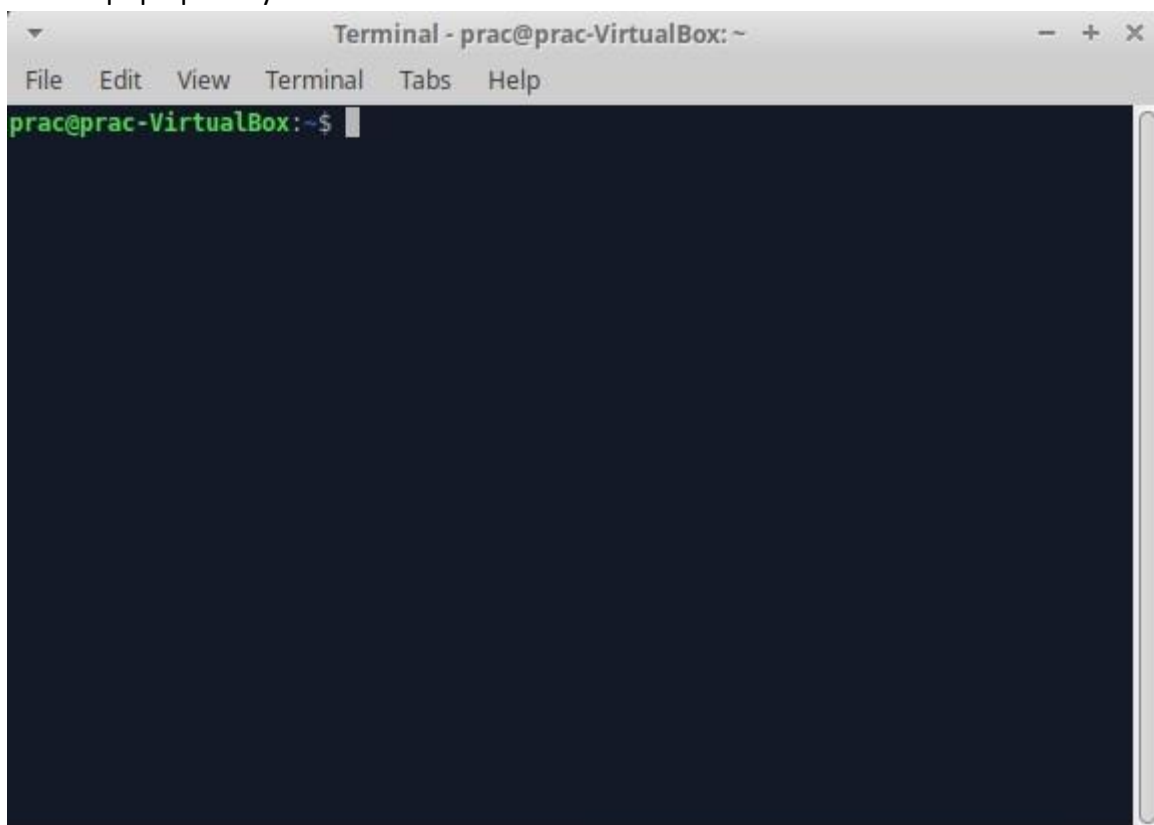
### Part 3 - Basic Linux Commands

Now have a VM up and running and looking nice. The default username/password for the VM is '**prac/unisa**'. The root password is also '**unisa**'.

Apache Hadoop is a collection of open-source software utilities that facilitates using a network of many computers to solve problems involving massive amounts of data and computation, which is installed on our VM in single node mode.

Most of the open-source tools and software are running on an open-source operating system called Linux. If you are not familiar with Linux before, that's totally fine. There are lots of resources Online for basic tutorials on Linux, and with a quick Google search you can probably find any command you're looking for. But we'll go over some basic commands below as well.

1. Open the terminal window by clicking the '*Xfce Terminal*' icon on the Desktop. A terminal window will pop up and you can use Linux commands here.



2. Enter the following command to show your working directory. Can you guess what '*pwd*' stands for?

```
$ pwd
```

3. Enter the following command to create a subdirectory called '*sub*'. You might be able to guess what '*mkdir*' stands for.

```
$ mkdir sub
```

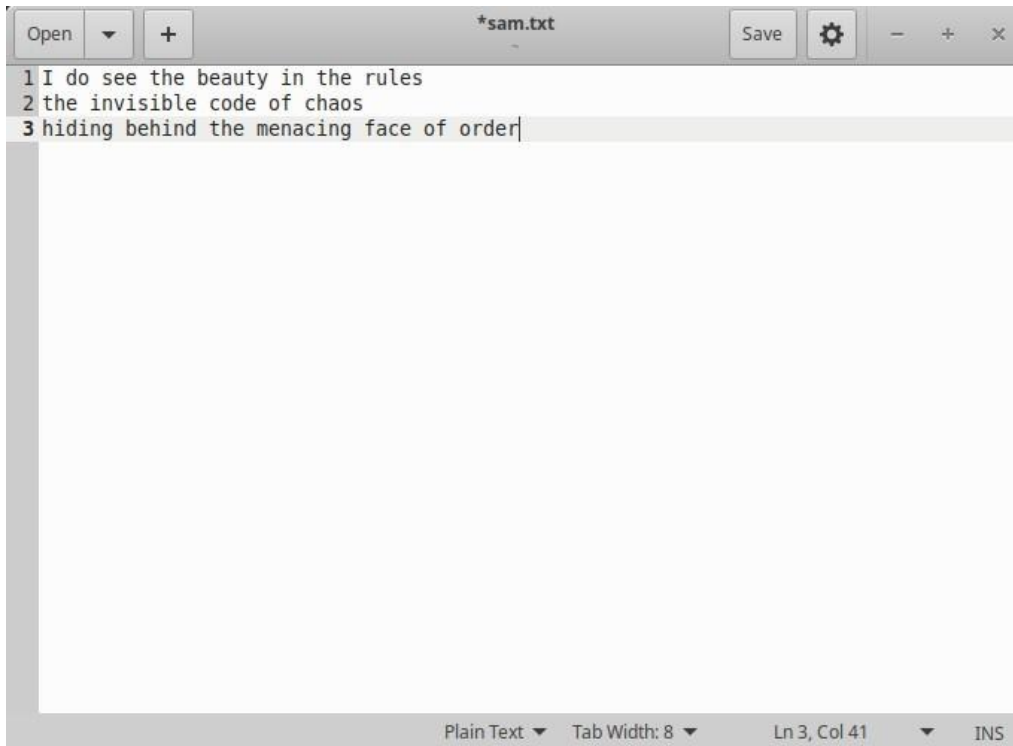
4. Move ourselves to the new *'sub'* directory. You might also be able to guess what *'cd'* stands for.

```
$ cd sub
```

5. Create a text file and call it whatever you like. I'll call mine *sam.txt*.

```
$ gedit sam.txt
```

This opens the gedit application. Fill your text file with some text of your choice. It looks like this



Click save and exit the gedit application.

6. We can look at the contents of our text document at any time by using the *cat* command.

```
$ cat sam.txt
```

7. Now we can copy that text file, to a new text file. You'll need to choose another name. I'm calling mine *'Sepiol'*, so my command looks like this

```
$ cp sam.txt sepiol.txt
```

8. In this step, we will choose a word from our second text file and change its appearance. I'll change the word *'chaos'* to the word *'burgers'*. When using the *sed* command, *-i* means that the files are processed in place, *s* represents substitution and *g* represents global ((It perform substitution globally on a line, without that flag, only the first hyphen on every line would get substituted.)).

```
$ sed -i 's/chaos/burgers/g' sepiol.txt
```

9. To check the differences between the two text files we use the diff command.

```
$ diff sam.txt sepiol.txt
```

For me, the output looks like this:

```
prac@prac-VirtualBox:~/sub$ diff sam.txt sepiol.txt
2c2
< the invisible code of chaos
---
> the invisible code of burgers
```

There are a few things to note here.

First, we can see that sub has been included in the prompt, since we moved to the 'sub' subdirectory.

Regarding the 'diff' command, the first line of output shows that the difference was identified in the 2<sup>nd</sup> line of the left-hand file in our command, compared with, the 2<sup>nd</sup> line of the right-hand file in our command. Those two lines are then shown with the < symbol indicating the left-hand file, and the > symbol indicating the right-hand file.

10. We can use the following two commands to show the first line from my sam.txt file, and the last line from sepiol.txt.

```
$ head -1 sam.txt $ tail -1 sepiol.txt
```

And we can combine these two commands into one command.

```
$ head -1 sam.txt ; tail -1 sepiol.txt
```

11. We can store the output from our last command as a separate text file. I'll call mine bill.txt.

```
$ (head -1 sam.txt ; tail -1 sepiol.txt) > bill.txt
```

12. Check the contents of the new text file.

```
$ wc bill.txt
```

Mine looks like this:

```
prac@prac-VirtualBox:~/sub$ wc bill.txt
 2 15 74 bill.txt
```

Here 'wc' is showing us that the file has 2 lines, 15 words and 74 characters.

13. Another commonly used command is 'ls'. This command shows us the content of the current folder. We should still be in the newly created 'sub' folder, so the only contents should be our 3 new text files.

```
$ ls
```

14. We can also move and delete files. Use the below command to move the third text file up one level in the directory hierarchy.

```
$ mv bill.txt ..
```

Here the term '..' is a special name used in Linux to refer to the parent directory.

We can then remove the first two text files from the current directory.

```
$ rm *.txt
```

The \* character is used to select all files with the suffix '.txt' within our current directory. As we just moved our third file, this will select the original two files, in my case sam.txt and sepiol.txt.

15. Can you figure out how to change directory to the previous folder (one level up)?

16. Now that there's nothing left in our 'sub' directory we can remove it. Can you have a guess at the command for removing a directory? It's okay if you can't, Google will have the answer.

17. Challenges (optional)

One of the most important parts of learning new technologies and using them in practice is being able to effectively both troubleshoot and research based on your needs. Below are some tasks using commands that we haven't covered yet, use Google (or any search engine) to help you complete them.

First, recreate the sub directory and the sam.txt file inside of it, then:

- Use the **awk** command to print the first two words of each line;
- Use the **awk** command to print the second to last word of each line;
- Run the same command you did in the previous question, but use three lines in the terminal;
- Use the **grep** command to find all lines containing the word 'chaos';
- From the command line, determine the size of the sam.txt file;
- Use two different commands to print your username in the terminal.

18. Feel free to play around with the commands you've learned so far or try out some new ones. When you're done, use the exit command to close the terminal window.

```
$ exit
```

## Useful Resources

- [34 Linux Basic Commands Every User Should Know](#)