Linux basic commands

Introduction to linux systems

u train

Table of content

1

The Command Line Interface

The CLI

The Command Line Interface

- When using **Linux operating systems** on servers in the company, we mostly use the **Command Line Interface (CLI)**
- It is also called **Terminal, shell prompt or console.**
- We use this terminal to run **Linux commands** that will perform specific tasks on our servers.
- More explanations will be given as the training goes on...

2

What is a Command?

Linux command

What is a command in linux?

- A **command** is a small program that we run in the terminal to **accomplish a specific task**
- Before running a command, we must know its **syntax** ie **the way we are suppose to write it and the parameters we need to know**
- **Running a command** means to **type it in the terminal** and then **press** the **Enter** button on your keyboard.
- Let's run some basic command to get started!

Note: Linux Commands are CaSe SeNsItIvE !

- The first thing to know is that Linux commands are Case sensitive
- Meaning that if you type the commands in the wrong case (uppercase or lowercase or mixed-case), it may not work as expected
- You might sometimes get a “**command not found**” message on the terminal.

3

Basic Linux commands

Date, cal, uptime, ls, whoami, id, pwd, cd, ls, mkdir, touch, rm, rm -r, free -m, lsblk, lscpu, top, man, history

Basic commands

- As we previously discussed, we run commands in a CLI (terminal)
- Let’s open a terminal to run some linux commands
- Now, launch your **Visual studio code**
- In the menu bar, click on **Terminal** and select **New terminal**
- Navigate to that folder that you created for your server by running the command: **cd ubuntu**

Basic commands

- Now, boot up your server with the command: **vagrant resume**

```
PS C:\Users\suzie\ubuntu> vagrant resume
==> default: Waiting for machine to boot. This may take a few minutes...
```

- Connect remotely to your server using the command: **vagrant ssh**

```
PS C:\Users\suzie\ubuntu> vagrant ssh
```

- Make sure you get the $ sign (This indicates that you are now in a linux server). **Let's run some commands!!**

```
vagrant@vagrant-ubuntu-trusty-64:~$
```

Date and time

- The **date** command is used to display the system date and time.
- By default the **date** command displays the date in the time zone on which the unix/linux operating system is configured.
- This will display the day, the month, the year and the time
- To run this command, type date in the terminal: **$ date**

```
vagrant@vagrant-ubuntu-trusty-64:~$ date
Wed Jan  5 19:40:43 UTC 2022
```

System uptime

- The **uptime** command is a command that returns information about:
  - How long your system has been running
  - The current time,
  - The number of users with running sessions,
  - The system load average for the past 1, 5, and 15 minutes.
- To run this command, type **uptime** in the terminal

```
vagrant@vagrant-ubuntu-trusty-64:~$ uptime
 19:42:13 up  1:00,  1 user,  load average: 0.00, 0.01, 0.03
```

- The load average here gives the engineer, an idea on how the server is doing in terms of speed!

The Calendar

- To display the calendar, we use the command **cal**
- By default it will display the calendar of the current month.
- To use this command, just run: **$ cal**

```
vagrant@vagrant-ubuntu-trusty-64:~$ cal
    January 2022
Su Mo Tu We Th Fr Sa
                   1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30 31
```

The Calendar

- To display the calendar for a specific month of a specific year, run the command **cal** with:
  - the **number of the month**
  - followed by the **number of the year**
  - **Example: $ cal 12 2021**

```
vagrant@vagrant-ubuntu-trusty-64:~$ cal 12 2021
   December 2021
Su Mo Tu We Th Fr Sa
            1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30 31
```

- To display the calendar for a whole year, run the command **cal** with the number of the year as option: **$ cal 2021**

Whoami

- **The whoami** command will tell you which user account you are currently using in the system.
- To use this command, type in the terminal: **$ whoami**
- Since we are connected to the server with the vagrant user account, this command will just display **vagrant**

```
vagrant@vagrant-ubuntu-trusty-64:~$ whoami
vagrant
```

id and pwd

The id command

- **id** prints real user id, and various other details related to the account.
- To run the command we use: $ **id**

```
vagrant@vagrant-ubuntu-trusty-64:~$ id
uid=1000(vagrant) gid=1000(vagrant) groups=1000(vagrant)
```

The pwd command

- **pwd** stands for Print Working Directory,
- This will display the absolute path of the current directory.
- Run: **$ pwd**

```
vagrant@vagrant-ubuntu-trusty-64:~$ pwd
/home/vagrant
```

The cd command

- The next command we will learn is **cd**, short for **Change Directory.**
- This command will help you to change your current directory.
- **Example:** Check the present working directory, run $ **cd /tmp** then check again with **$ pwd**

```
vagrant@vagrant-ubuntu-trusty-64:~$ pwd
/home/vagrant
vagrant@vagrant-ubuntu-trusty-64:~$ cd /tmp
vagrant@vagrant-ubuntu-trusty-64:/tmp$ pwd
/tmp
```

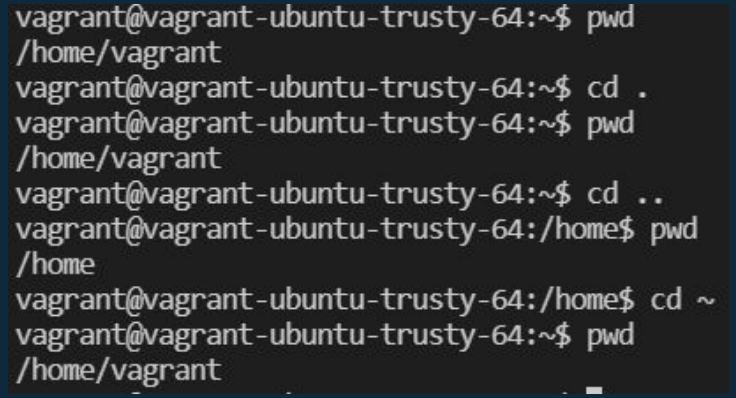
The cd command

- After the command, the current directory has changed to **/tmp**
- To quickly move to the current user's home directory, use **cd** with the ~ symbol: **$ cd ~**

```
vagrant@vagrant-ubuntu-trusty-64:/tmp$ cd ~
vagrant@vagrant-ubuntu-trusty-64:~$ pwd
/home/vagrant
```

The cd command: . and .. options

- . and .. has special meaning in the Linux:
  - . means the **current directory**
  - .. means the **parent directory**
- We can use these in various situations for daily activities.



The ls command

- We use **ls** command to **list the files and directories inside any given directory.**
- If you use **ls** command **without any argument**, then it will **list the content on the current directory.**
- As an example, let's display the content of the **/** directory: **ls /**

```
vagrant@vagrant-ubuntu-trusty-64:~$ ls /
bin   etc         initrd.img.old  lost+found  opt   run   sys   vagrant  vmlinuz.old
boot  home        lib             media       proc  sbin  tmp   var
dev   initrd.img  lib64           mnt         root  srv   usr   vmlinuz
```

- **/** is a special directory, which represents root directory in Linux filesystem. You will learn more about that in future lessons.

The ls command

- Let's list the content of the /home directory: **ls /home**

```
vagrant@vagrant-ubuntu-trusty-64:~$ ls /home
ubuntu  vagrant
```

- We can also choose to navigate to a specific directory before running the ls command:
  - **$ cd /home**
  - **$ pwd**
  - **$ ls**

```
vagrant@vagrant-ubuntu-trusty-64:~$ cd /home
vagrant@vagrant-ubuntu-trusty-64:/home$ pwd
/home
vagrant@vagrant-ubuntu-trusty-64:/home$ ls
ubuntu  vagrant
```

Create a directory

- To create directories in Linux, we use the **mkdir** command
- Let's create a new directory in the home folder of the current user:
  - **$ cd ~**
  - **$ mkdir basic**
  - **$ ls**

```
vagrant@vagrant-ubuntu-trusty-64:~$ cd ~
vagrant@vagrant-ubuntu-trusty-64:~$ mkdir basic
vagrant@vagrant-ubuntu-trusty-64:~$ ls
basic
```

Create a file

- To create a file we use the **touch** command.
- Let's create two files in the basic directory we created previously,
  - **$ cd basic**
  - **$ touch file1**
  - **$ touch file2**

```
vagrant@vagrant-ubuntu-trusty-64:~$ cd basic
vagrant@vagrant-ubuntu-trusty-64:~/basic$ touch file1
vagrant@vagrant-ubuntu-trusty-64:~/basic$ touch file2
vagrant@vagrant-ubuntu-trusty-64:~/basic$ ls
file1  file2
```

Delete a file or a directory

- To delete a file we use the **rm** command.
- Let's delete the files we previously created: (use **ls** to check)
  - **$ rm file1**
  - **$ rm file2**
- To delete a directory, we use the **rm** command with the option **-r**
- To delete the directory we previously created, we need to navigate to its parent directory then use **rm -r** to delete it
  - **$ cd ~** (Since we created it in the home directory)
  - **$ rm -r basic**

Memory characteristics

- To display the memory characteristics of a server, we use the **free -m** command
- This command displays the **total size** of the memory, the **used size**, the **free size** etc in **megabytes**

```
               total        used        free      shared  buff/cache   available
Mem:             486          82         263           4         141         387
Swap:           2047           0        2047
```

The Hard drive, the CPU

- To check the hard drive characteristics, we use the command
  - $ **lsblk**

```
NAME   MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
sda      8:0    0  40G  0 disk
└─sda1   8:1    0  40G  0 part /
```

- To check the CPU characteristics, we use the command
  - $ **lscpu**

The top command

- The **top** command presents a real time view of a running system.
- This command can be used to display the memory characteristics just as the **free -m** command
- It displays more information about the system
- With this command, you can check the **memory usage** as well as the **CPU usage** of your server
  - $ **top**
- **Press q to quit the top command interface**

The man command

- **The man command** in Linux is **used to display the user manual of any command that we can run on the terminal.**
- For example to get more information on the command **uptime**, we can run (Press **q** to quit the man page)
  - $ **man uptime**
- To better understand how to use the **lscpu** command, you run:
  - $ **man lscpu**

The history command

- The **# history** command displays all the commands you have typed in the session.
- Each command is displayed on a single line for better visibility

```
vagrant@vagrant-ubuntu-trusty-64:~$ history
    1  date
    2  uptime
    3  cal
    4  whoami
    5  id
    6  pwd
    7  ls
    8  ls /
    9  mkdir basic
   10  ls
```

That is all for this lesson

Don't bother if you don't understand it now, we will come back to most of these commands during the training

Now, exit the server and suspend it:

```
$ exit
> vagrant suspend
```

u train

“ Don’t try to memorize all of this!

Just try to understand how it works, practise it and have fun playing around with commands ”

NB: Don’t forget to exit and suspend the server when you are done

See you guys in the next lesson!

Thanks!

Any questions?

You can find us at:

website: http://utrains.org/

Phone: +1 (302) 689 3440

Email: contact@utrains.org

Click on the link below to contact the support team for any issue!

utrains.org/support/

Create a ticket for your problem and we will get back to you soon!