Linux

18-Apr-2022

**Basic Linux Command**

|  |  |
| --- | --- |
| **Commands** | **Description** |
| ifconfig | To display IP Address and broadcast ID detail To display IP Address and broadcast ID detail |
| date | To display the date and time |
| Uptime | It will show the uptime |
| Cal | To display the calendar |
| whoami | To know the logged in user name |
| who | How many users are logged in |
| top | To know the what are the processes are running / CPU, Memory utilization / How many users are available / PID (Process ID) |
| Kill -9 PID | To stop the process Ex: kill -9 101 (this will kill the process ID 101) / We will use this when more slowness issue or more utilization time |
| Ps or ps –ef or ps –ef | grep any | It will show all the process with all details / by using grep we can search only particular process or any other details. |
| q or ctrl+c | To exit from the prompt |
| Clear shortcut is ctrl+l | It will clear the page and it will go to 1st line. / shortcut for that is ctrl+l key |
| free –m | It will show total RAM size, used and free |
| Pwd | It will show the present working directory |
| history | It will show all the commands we used early |
| Man | Man command is help to give the full details (Ex: Man pwd) It will explain the full details of pwd command |
| Uname | It will show which OS we are working on |
| Sudo passwd root | This will create a password for root |
| Ls | List command |
| Ls -l | Ls – l (List the command with full details) |
| Ls-a | it will show the hidden file |
| Useradd –m name | It will create a user (Ex: useradd –m faiz) –m menas home directory creation |
| -m | Home directory creation for users Ex: useradd -m |
| Su | Switchuser (Ex su faiz) |
| Reboot | Reboot the system |
| Halt | Brings the system down immediately |
| Shutdown | Shutdown the system |
| Cd ~ (tilt symbol) | Cd(Change Directory) It will useful to go to user directory folder Ex: cd ~ of user faiz, It will directly goes to faiz, to check pwd command  (Doubt if multiple users) |
| Cd .. / cd../.. | cd.. It will go to previous directory / cd../.. it will go to previous of previous folder |
| cd / | It will go to starting path |
| Mv | Rename file name or move the file Ex for rename: mv filename newfilename / mv faiz.txt renamed.txt (Now filename was changed from  Faiz.txt to renamed.txt) Ex for move: mv filename faizul/(foldername is faizul) / mv faiz.txt Faizul/ (It will move to Faizul folder) /  Same methos for moving or renaming for folders too |
| Wc | Word count of the file Ex: wc filename / wc faiz.txt / 1st column is no of lines / 2nd column is no of words / 3rd column is no of letter+line  (Ex: I am Faizul / (Letters is 11 + 1 line total 12) / 4th is filename |
| **File Managemnet Types** | |
| 1.Normal Files (Image, Text, Mp3), 2. Directories (Folders) 3 Special Files (Hidden, System related, link) / Hidden files b4 dot will be there | |
| Mkdir | It will Make directories Ex; mkdir ffshz |
| CD (Directories name) | Ex: CD ffshz (It will go to ffshz directory) |
| Touch (File name) | To create a empty file Ex: Touch expense (It will create a empty file and file name is expense) |
| Cat | Cat is used to see the content of the file (Ex: cat expense) |
| Rm / rm \* | Remove (it will delete the file) Ex: rm expense / Ex2: rm faiz.txt work.txt (It will delete multiple files) /  Ex3: rm expe\* (It will delete all the files name starting with expe) |
| Rm | Remove file from one directory location to another directory location Ex: rm faiz/(destination folder name)expense.txt(filename) /  rm faiz/expense.txt |
| Rmdir | It will delete directory Ex rmdir utest2(directoryname) |
| Rm –rf | Rm –rf ffshz (If directory have some files then rmdir ffshz won’t delete the directory bcos inside expense file is there) So if you want delete  then use rm -rf ffshz (now it will delete) f represents force / Forcefully deleting |
| Cp | Copy command / Ex: cp expense expensebackup (Cp sourcefilename destinationfilename) In case if you want to change anything in expense  b4 we can take the backup and make it anychange / This will copy to the same folder / Destination file name should be differ |
| Cp | This example is for to copy a file to different folder cp expense.txt(filename) faiz/(foldername) (It will copy that file here) or  cp expense.txt(filename) faiz/(foldername)expense2.txt / It will copy and paste that file expense2 name |
| **VI Editor is useful for editing in linux / In windows we will edit or create any excel files** | |
| Vi | 2 Modes (edit and escape) Text Editor (Ex: Vi expense) It will create new file and It will go to edit mode, There we can type what we want:  Ex: Welcome to DESIZORS |
| I | Insert mode |
| :wq | In VI editor use this command to save and exit |
| :q! | Without save and exit |
| dd or d8d or 8dd | To Delete the single line press 2times dd, If you want to delete more lines then press d8d or any required number. |
| Yy | To copy the line |
| P | Paste |
| U | Undo |
| :s | Single String replace Ex: :s/car/truck or :s/car/truck/g (This will replace only one string) / case sensitive is there, so if you want to replace  (Letter F) Need to type F, If u type f it won’t replace |
| /string or ?string | Ex /faiz (it will search faiz from top) / ?faiz (It will search from bottom) |
| :%s | String replace Ex : :%s/car/truck/ or :%s/car/truck/g (This will replace entire string) |
| Grep string filename | To search a file or a string in a file / Ex: grep welcome faiz.txt (It will search inside the faiz.txt file and if found it will display else not found) |
| Ls -l | grep foldername | To search the folder name or file name starting from fold (ex folder1, folder2, folder3 folders are availbale) So it will show all that three  Ex: ls -l | grep fold (Need to use pipeline symbol not exclamatory) |
| Sort Command is used for list in alphabetic or numeric or Ascending or descending order (n represent numeric, r is reverse, f is \*\*\*) | |
| Sort filename | It will display special characters 1st, numbers 2nd (1,11,2,25,3) 3rd Capital Letter in ascending order  and 4th small letters in ascending order |
| Sort –n filename | **It will show numbers in small to big**, It will display special characters 1st, 2nd Capital Letter in ascending order,  3rd small letters in ascending order and 4th numbers in small to big (1,2,3,11,25) |
| Sort –r filename | It will display in reverse (1st small letters in descending, 2nd capital letter in descending, 3rd numbers(3,25,2,11,1) and 4th special characters |
| Sort –f filename | Upper case and lower case together (Ex: A,a,am Faizul, B, C,c,cat…) / 1st special character, 2nd number  and 3rd upper case and lower case together |
| **Folder Permission** | |
| There are three permissions are there 1. Read 2. Write and 3. Execute (RWX) (R value is 4, W value is 2, X value is 1) | |
| When you use (ls -l filename) It will show some details like (–rw-rw-r—1 myfaizul sdteam 0 apr 21 10:20 faiz.txt) myfaizul is owner permission (rw-) /  sdteam group permission (rw-) / 0 others permission (r--)) (r - read value is 4 / w - write value is 2 / e – execute value is 1) Total 7  In the list first letter is – (Its is file) If 1st letter is D (Its is directory) | |
| Chmod | Chmod is used to change the folder permission details: Ex1: Owner only need access , groups and others don’t want access  Then chmod 700 faiz.txt / Ex2: need full permission for owner, groups read and write, others only read so chmod 764 |
| **Sudo privilege (To install anything user need the sudo privileges)** | |
| Su - | It’s indicate switch user to root / type the root password |
| Usermod – aG sudo  username | Ex: usermod –aG sudo myfaizul (aG is append group) |
| Now myfaizul user got the sudo privilege / To check the group of user command is groups username / If you see root / then successfull | |
| Groups username | Ex: groups myfaizul |
| **To install python in ubuntu** | |
| Sudo apt update -y | Need to update by using this command |
| Apt list | It will show all the packages of updated list |
| Sudo apt install software-  Properties-common | After this |
| Sudo add-apt-repository ppa:  Deadsnakes/ppa | Again enter to continue |
| Sudo apt install python3.9 | It will install |
| Python3.9 –version | It shows the full version of the python Ex:python3.9.12 |
| Ls –l /usr/bin/python\* | It shows the all the versions of python |
| Sudo apt update | It will update after entering the password (faizfaiz@58) |
| Sudo apt install gedit | It will install gedit |
| Sudo apt install xclip | Use to copy & paste with the terminal |
| Cat filename | xclip –sel clip | Use to copy & paste with the terminal |
|  |  |
|  |  |
|  |  |

What is GIT & Uses?

What is repository & Code repository?

What is global install?

What is kernel?

Where that python will save in Ubuntu?

Development +

SDLC: Software dev life cycle

ICICI: Repositori

Modules / Componenets: Inter Dependency

**GIT:**

**Monalithical vs Micro service appl (Zero time downtime)**

**Virtualization / Containerization**

**Focus:**

**Devops Technologies:**

**GIT, Jenkins, Shell Scripting, Linux**

**GIT: Source Code Manager /**

**Version Control Mgmt:**

**Repository**

**Metadata: all history**

**Unit Testing**

**Install: Linux / Work on this GIT**

**Ubuntu / shell script**

**Zero Down Time: Micro Service Application**

Devops:

**Development team: Developing of application, testing of application**

**Operations team: deployment of applications, maintained on a server**

**Devops is link between two**

What tools are required to carry out this task?

**Concept of software development:**

**Development**: Software developer will use different kind of languages with any technologies to build software / need code repository have to work on the code / most famous code repository is GIT

**Operations:** Applications needs to be deploy into the servers, Then only user can access / so we need some kind of IT infrastructure for servers either on-premise server or cloud server / These servers should be created and configure to run our application.

**Devops**:

**In Development side:** Devops engineer will not programming the application but need to understand the concepts of / **How developers work** / **Which GIT Workflow they are using** / **How application is configured** / **Concepts of automatic testing**

**In Operations side:** Responsible for prepare the infrastructure to run the application

Devops Pipeline is: test🡪build🡪deploy🡪configure🡪monitor

Most servers are running in Linux: So we need knowledge in below

* Linux Basics
* comfortable using CLI(Command Line Interface) (Bcos most of the stuff working on the server in command line)
* Shell commands (Knowing basic linux commands)
* Linux file system
* Server management (Basics of how to administrate the server)
* SSH key management
* Basics of networking and security (To configure firewall to secure the application, Proxy, Load Balancers, HTTP/HTTPS, IP, DNS Name Resolution)

**Devops vs IT operations:**

Devops engineer don’t want be specialized in advanced super operating system, networking and security skills.

For administrate the server from start to finish there are own professions like network, system engineers and security engineers.

So Devops engineers is to understand the concepts and know all this to the extent that you’re able to prepare the server to run your application, But no need to take completely take over managing the servers and whole infrastructure.

**CONTAINERS:**

Containers is the new standard, Need to running your application as containers on a server

So we need to generally understand

* Concept of virtualization.
* Concepts of containers. (Containerized applications on a server (Most popular container technologies is Docker container/ So it’s must to learn)

In 2017 20% of companies have containers deployed and 2020 50% of companies have containers deployed.



DEV: After software development again developers is creating new features and bug fixes on one side.

OPS: Another side we have Servers which are managed and configured to run this application.

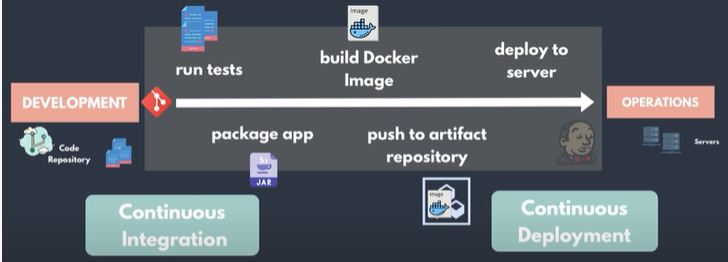
Now question is how to get these new features and bug fixes from development team to the servers to make it available to the end users.

How to release the updated application (new version)?

DEVOPS: The main task and responsibilities of Devops comes in. With Devops the question is not just how we do this in any possible way but how we do this continuously and in an efficient fast and automated way.

**Devops engineer work is below:**

1. **Continuously (CI/CD) (Continuous Integration / Continuous Deployment or Continuous Delivery)**
2. **Automated**



When the features are bugs fixed is done. We need to **run the tests** and **package the application** as an artifact like jar file or zip file, etc.. so we can deploy it that’s where build tools and package manager tools come in some of the examples are maven and gradle for java applications, Npm for javascript applications and so on.

So we need to understand how this process of packaging testing applications works. Need to **building docker images** fromour application, Next this image must be saved in an image repository so docker **artifact repository** on nexus or docker hubs, etc. So we need to understand how to create and manage artifact repositories as well.

We don’t want to do any of this manually instead we want one pipeline that does all of these in sequential steps to you need build automation and one of the most popular build **automation tools is Jenkins.** We need to connect this pipeline with git repository to get the code so this is part of **continuous intergration process** where code changes from the code repository get continuously tested and we need to deploy that new new features or bug fix to the server after its tested built and packaged which is part of **continuous deployment process.**

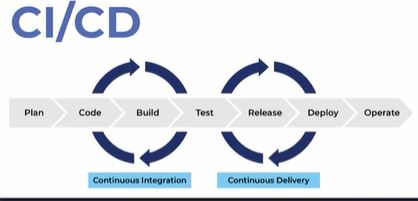
Where code changes get deployedcontinuouslyon a deployment server. Additional steps in this pipeline sending notification to team about the pipeline state or handling failed deployment etc. This flow represents the core of the CICD pipeline.

1. Build Tools & Package Manager Tools
2. Artifact Repository
3. Build Automation

The CICD pipeline happens to be at the heart of the Devops tasks and responsibilities, So Devops engineers need to configure the complete CICD pipeline for our application.

Pipeline should be continuous that’s why the official logo of Devops is an infinite cycle because the application improvement is infinite.

New features and bug fixes get added all the time that need to be deployed.



**CLOUD PROVIDERS**

Now a days many companies are using virtual infrastructure on the cloud instead of creating and managing their own physical infrastructure.

These are the infrastructure as a service platforms like **AWS, Google Cloud, Azure, Linux, Etc..**

The main reason for that is to **save costs** of setting up your own infrastructure.

These platforms also manage a lot of stuff for making it much easier to manage the infrastructure there. Example: using a UI (User interface) we can create a network, configure firewall, route tables and all parts of your infrastructure through services and features that these platforms provide.

**Save Cost**

**Offer a range of services**

If we use AWS we don’t want to learn all the services, we just want to learn those concepts and services that we need to deploy and run the specific application on the AWS infrastructure.

**CONTAINER ORCHESTRATION**

Application will run as a container because we’re building docker images.

Containers need to be managed for **smaller applications docker compose or docker swarm** is enough to manage them.



But if you have a lot more container like **big microservices** then we need a more **powerful container orchestration tool** to do the job.



So most popular of which is kubernetes cluster, so we need to know how kubernetes works and be able to administer and manage the cluster as well as deploy applications.

**MONITORING**

When you have thousands of containers running in kubernetes on hundreds of servers then how can we **track performance** of our individual applications or everything runs successfully or whether infrastructure has any problems.

How do we know in real time users are experiencing any problems.

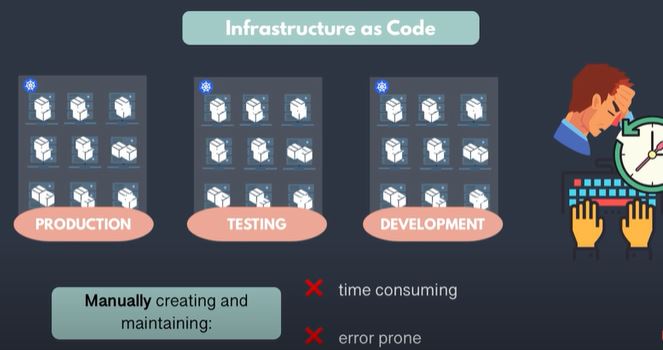
Main responsibilities for **Devops engineer to set up monitoring** for the running application the underlying kubernetes cluster and the servers on which the cluster is running.

So we need to know a monitoring tool like **Prometheus or Nagios** etc.

**INFRASTRUCTURE AS CODE**

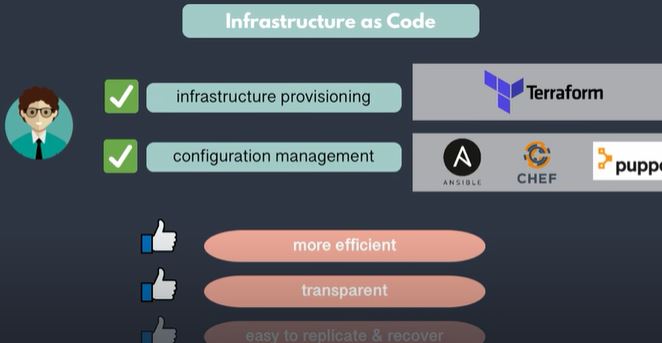
Here production environment is there, to properly test our application before deploying it to the production we need the same deployment environment for multiple times (Testing and development environment).

To creating and maintaining that infrastructure for one environment already takes a lot of time and is very error prone, so we don’t want to do it manually three times.

****

We try to automate as much as possible. To automate this process creating the infrastructure as well as configuring it to run our application and then deploying our application on that configured infrastructure can be done using a combination of two types of **infrastructure as code** **tools**

1. Infrastructure provisioning tool (Terraform)
2. Configuring Management tool (ANSIBLE, CHEF, PUPPET)

****

**1.Its’ more efficient 2. Transparent 3. Easy to replicate & recover**

**SCRIPTING LANGUAGE**

Need to work with developers and system administrators to **automate some of the tasks**, we need to know write script skill.

Ex: Small applications to automate tasks like doing **backups, system monitoring tasks, cron jobs, network management** , etc..

We need to know a scripting language this could be an operating system specific scripting language like bash or powershell or python

Python, RUBY, GOLINE is more powerful and flexible language, which are also operating system independent.

Python is popular and demanded one in Devops space, so we need to learn python.

PYTHON is easy to learn, easy to read and very flexible, Python has libraries for most of the databases operating system tasks as well as for different cloud platforms.

**VERSION CONTROL**

To manger our code just like the application code you manage this also using version control like **git.** So we need to learn git

**SDLC (Software Development Life Cycle)**

1. **Planning**
2. **Requirement Analysis (SRS: Software Requirement Specification)**
3. **Design**
4. **Development**
5. **Testing**
6. **Implementation & Maintenance**

**PLANNING MODULE TO LEARN DEVOPS:**

1. Overview & Intro to DevOps Over
2. OS Linux & Basics Over
3. Version Control With GIT
4. Build & Package Manager Tools (Gradle, NPM)
5. Cloud & IaaS
6. Artifact Repo Manager
7. Containers
8. Build Automation (CI/CD)
9. AWS Services
10. Container Orchestration
11. K8s on AWS
12. IaC With Terraform
13. Programming Basics
14. Automation with python
15. Configuration Management
16. Monitoring with Prometheus