**NAS: <https://109.190.105.24:5001/#/signin>**

**Projet devops github: https://github.com/jfrog/project-examples**

**Teams lien: <https://mail.google.com/mail/u/0/#search/partag%C3%A9+le+dossier/FMfcgzGxSRPkDcGTrMGSLTgfcqvprDWN>**

3 principeux OS Library biniare, cpu, pilot

**DevOps**

**Miss first video**

**SDLC(software development life cycle): is a process normally follow three steps: design,developpemnt and test.**

**The aim of SDLC the high quality delivery project**

****

**The main focus as DevOps engineer, building(developpement),testing(quality assurance QA) and deployment(production server for customer) a software.**

**DevOps basically fasten this process and improve this process to deliver the code quickly by ensuring all of these method (building, testing,deployment).**

**As a devops we should focus on automation and improving efficiency of building, testing and deployment.**

**Virtual Machine:virtual environnement that function as virtual computer systems and this systems has own cpu, ram and own hardware.**

**super hypervisors are vmware, xen**

**Aws:Amazon Web Services, Inc. (Amazon Web Services)**

**Azure: Microsoft Azure est la plate-forme applicative en nuage de Microsoft**

**Connect with virtual machine AWS from local via mobixtreme :**

**By mobiextrem we can connect free with 10 virtual machine**

**Copie ip address from Aws and connect with key-gen was created during creating virtual machine**

**Commande:ssh -i ‘C:/Users/moham/Downloads/test.pem’ [ubuntu@13.53.132.194](mailto:ubuntu@13.53.132.194) (-i for identity file)**

**Linux and shell scripting**

**Operating systeme: acts as bridge bentween software and hardware , so its drive of medium between soft and hard.**

**Why linux: more security, in production so fast, open source, no need to anti malware.**

**Heart of os kernel , on top you have system librairies, on top system software user process , compliles**

****

**What is kernal: is heart of os,the responsiblity communication between hard and soft.**

**Kernel has four primary reponsibility,device management, memory managment, proccess management, handinling system calls.**

**System libraries: responisibity for performing a task ex: libsi,gnu**

**Shell scripting: a way talk to your os, to communicate with os with command line is call shell command.**

**Basic command of shellscrpting: bash is the most popular**

**Command linux/shell: ls, pwd, etc**

**Free -g (memory of machine), nproc (number of cpu), df -h (disk size details)**

**Top(information on everything )**

**Anther video on shellscripting:**

**Automation: Automation is process where you will try to reduce manual activites.**

**Shellscripting: is a tool to execute script.**

**Commande: ls -ltr (who create where create permission, when create)**

**Man (is for manual then write commande): man ls**

**Shebang: #!/bin/bash(which take action for execute script)**

**Diefference between sh and bash: If shebang sh it will be converting on bash by linking.(sh is linking in linux to bash) Some os decide to use defalut dash**

**Executirng file with shebang: sh nom\_scrip.sh or ./nom\_script.sh**

**Chmod: which user, which gorup, what permission**

**Alluser chmod 777, 7 is magic number: 7 , 7myself,7mygroup,7everyone**

**7:4(read)+2(write)+1(execute)**

**Exemeple: 444**

**User(owner)------>4read**

**Group---------->4read**

**Everyone------>4read**

**Grep a process:**

**Ps -ef for all process in server, then grep desire process.**

**Ps -ef | grep ‘amazon’ (e=entire f=full process)**

**Pipe | normally send first command result to sencod command**

**But if we write date | echo «sohrab» the result is sohrab here pipe not send first command result to second result beacuse: date commande is stdin(system default) command , on the other hand ps -ef is stdout command.**

**Operating system: communicate between hardware and software**

**What does do kernel in operating system:1. kernel is heart of operating system , its has four primary responsibilities: device management,memory management,process management, handling system calls**

1. **System and libraries do Performing a task**
2. **System software,users process, compilers**

**Fundamental of shell scripting: shell: is way talk to operating system. Commande line is shell commande**

**Bash is a script-style command-line interpreter.**

**Commandes:**

**Commande to see property: free -g**

**Commande disk size: df -h**

**Commande for everything manage cpu, memory etc: top**

**Shell script:**

**For details : man touch/man mkdir**

**Open a file with editor: vi file, vim file (but vim need to install)**

**To write something ESC then i for insert,for save and exit ESC then:wq! , for close without save :q!**

**To write shell script at first write shebag #!/bin/bash ou #!/bin/sh**

**Bash, ksh,sh, dash etc work like as compiler java**

**Execute file: ./ means this file is executable(must give chmod) or write sh file with extensions(sh name.sh)**

**Chmod define for user, group, everyone**

**Linux chmod: 4 is read, 2 is for write, 1 is for execute.**

**Exemple: chmod 444 xyz**

**To see history commande: history**

**DevOps: architecture management, code, configure management all are doing by shell automatic.**

**In a word shell script is a tool for automatisation**

**Commande free for information ram: free**

**Top commande for what are processes are running in machine, see memory, cpu usages etc. (commande:top)**

**4 importants commands linux:df(all avaible space in virtaul machine),free(memory of the machine),nproc(display cpu of the current machine),top (print all infos process,slepping process, consuming memory) we can write: df | free | nproc |top**

**If any problem in virtual machine how to check: create a file script to check the problem.**

**It must be write meta-data,and debug mode(to)set -x (is echo state)**

**Another echo commande: set -e(exit the script wehn where there is an error),another echo commande: set -o(pipefail)(pipe fail is very important because if fail the script will execute otherwise not execute)**

**To see the process running in computer: ps -ef(p:process,-e:entire process)**

**To see particular process: ps -ef |grep ‘amazon’**

**Every virtuelle has 3 chanel: stdin, stdout,stderr so, if we write date | echo‘today is’ ,this case first commande not to send result to second commande beacause this is stdin.so result is today is.**

**For greping id instead of entire process or details we can use awk.**

**Awk: is pattern scanning and processing language.**

**Awk commande retrive information specific column.**

**Commande : ps -ef |grep amazon |awk -F’’ ‘‘ ‘{print$2}’**

**Grep something from file: grep grep\_target filename**

**Commande CURL: retrieve information from internet(api)**

**curl lien recover information then grep recover error :curl lien |grep ERROR**

**Another commande: curl -X Get api\_link**

**Wget commande: wget command as same as curl diffierncef download logfile and curl command display not download.**

**Linux commande su means (switch user)**

**Find commande: to find a file from your system: sudo find /pam.d**

**Trap commande: trap for use traffic signal,**

**Kill commande: kill commande uses for kill a process**

**Questions on shell scripting**

**I use shell scripting for listing file, to find something, etc.**

1. **List all the processes: ps -ef**
2. **Find me id a perticular process: ps -ef |awk -F’’ ’’ ‘{print$col\_num}’**
3. **Write all logs store in s3 backet or google storage: curl google.com | grep HREF**
4. **Write a shell script to print number numbers divided by 3 and 5 but no 15?**

**For n in {1..100}->do->if [$((n%3))==0] ->then ->echo «»->fi->done**

1. **Write a script to print number of s in mississippi**

**X=mississipi->grep -o ‘s’ <<<’$x’ | wc -l(here -o:only, ‘s’, <<<: ,wc:word count, l:length)**

1. **What is crontab in linux?can you provide an example?**

**Crontab : like alert , it will execute a certain time which is defined by devops**

**Report: node health**

1. **How to open read-only file?**

**Vim -r file\_name.txt**

1. **What is difference between soft link and hard link?**

**In Linux there is two link soft and hard link, hard link created a backup into memory so if original file deleted but hard link file will be rested.example:create and save file**

**Soft link: exam: python install in my machine so python file save into same point of memory if deleted the file il will be deleted no backup file exist.speacilly for create a alias**

1. **What is the difference between break and continue?**

**Break: breaking the execution file, Continue: continuing the execute->skip this and continue the next. Example: in for loop we can put break and continue.**

1. **What are disadvantages of the shell scripts.**

**-Errors are frequent and costly and a single error can alter the command.**

**-The execution speed is slow**

**-Bugs or inadequacies in the language’s syntax or implementation**

**-Every time a shell command is executed, a new process is lanuched.**

1. **What are different types of loops and when to use?**

**Such as while, do while, for etc. Purpose same as other languges**

1. **Is bash dynamically or statically typed and why?**

**Bash statically typed, example: x=5 (but dynamically declared var x string in other language)**

1. **Explain about a network troubleshooting utility?**

**Traceroute and tracepath**

1. **How will you sort list of names in a file?**

**Sort**

1. **How will you manage logs of a system that generate huge log files everyday?**

**Linux admin and administrateurs: if we save all logs it will consume memomry so we can use logrotate to manage linux machine logs.we can say every 24 hours save logs in different logs(gzip,tar,zip etc.) and can say after 30 days delete logs.**

**Why pepole move to aws**

1. **Mainaint own server**
2. **Have to create own datacenter**
3. **Security issue**
4. **Need to constant upgrade server**

**Of all this I need to dedicated team, engineer team, and cost is so high**

**Thats why move to aws pay as you go.**

1. **Reduce managging overhead**
2. **Reduceing maintaing overhead**
3. **Cost effective**

**Devops engineer to give reports HR on aws (how many ec2,s3,rds etc..)**

**Manually it is difficult if we atuomation this work in certain time exit report we can use crownjob.**

**Do realise this project I need to install aws cli**

**Connect with aws cli by aws configure:**

**For command line go to aws cli command referenes**

**Write scirpt: #!/bin/bash**

**########**

**Author: sohrab**

**Date: 13/08/2024**

**Version: v1**

**This srcipt will report the aws resource usage**

**###########**

**Set -x (result put into debug mode)**

**#list s3 bukcets**

**Echo «list of s3»**

**Aws s3 ls >resourceTracker**

**#lsit ec2 instance**

**Echo «list of ec2 instances»**

**Aws ec2 describe-instances | jq ‘Reservations[].Instances[].InstanceId’**

**#list lambda function**

**Echo «list of lamda»**

**Aws lambda list-functions >>resourceTracker:q!**

**#list IAM user**

**Echo «list of users»**

**Aws iam list-users**

**Here jq json file**

**Execute: ./reports.sh**

**8 video must be watch**

**Git**

**Svn is centralised version constrol system but git is distributed control system.**

**Fork: Entire copy of original source so if projet is down can collect from form thats why GIT is distributed**

**Git: Control version systeme**

**Github:**

**To see all commit: git log**

**To go previous commit: git reset --hard comit\_id**

**Git branching:**

**Branch : is separation to work wihtout happening incedent on main project.**

**Good brancing strategy: master branch, feautre branch, release branch hot fix branches etc....**

**Any branch change master must be up to date..**

**Git interview question:**

**How to iniiliez git in ur project;**

**Git init: initialise a git local repository for me**

**.git is responsible tracking and ensure there is no secret**

**What is the git flow in ur company**

**Ans : Git add, git cmmit and git push**

**Waht is git clone ? download the repositroy from github**

**Differnce between clone and fork.**

**Fork: githbu is distributed controlling version system, copy of repository**

**Commands: git cherry-pick,git rebase, git merge**

**NIS**

**Top most 15 aws services**

**Aws: cloud provier; provie infratsutre as service, help to service lelvel**

**Aws provide saas**

1. **Ec2**
2. **Vpc(serve: security group,CIDR, inbound, outbound)**
3. **EBS( volume staorage related services)**
4. **S3 (storage)encrypted by default**
5. **IAM**
6. **Cloud watch(take care of monitoring and observation related sevice and send notification)**
7. **Lamda(serverless compute, whiich used for shrot actions or shot functionalites)**
8. **Cloud build service(performing CD/CD,aws code pipeline,aws code build, aws code deploy)**
9. **Aws configuration (monitor configuartions service)**
10. **Billing and costing(biling pending**
11. **Aws kms:key managment service(shipting another regiion with secure data )**
12. **Cloud trail(enable operation and risk auditing, stores information API logs)**
13. **Aws eks (kubernetes service manage by aws)**
14. **Fargate, ECS**
15. **ELK(elastic serach )**

**What is the difference between eks and ecs?(both of them are container orchestration solutions but ECS is a aws proprietor solution aws by their won it is called as elastic container service, eks is mamnaged kubernetes service.**

**Fargate: serveless container service.**

**Cloud: ressource rendre accèssible service**

Cloud infrastructure:

Why need to move cloud structure like, aws or azure,

1. Manageability(for exemple startup need to much teams)
2. Cost (pay as you go)

**EC2**: Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud.

**AWS Lambda** is a server-less, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers.

**Amazon S3** gives every user, its service access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of websites

**AWS Identity and Access Management (IAM)** roles are entities you create and assign specific permissions to that allow trusted identities such as workforce identities and applications to perform actions in AWS.

**Cron jobs, sh**ort for "chronos" in Greek, are scheduled tasks that automate repetitive processes and streamline operations on AWS.

**AWS CLI**: is an open source tool from AWS. You can use it to interact with AWS services using commands in your command line shell.

**Clé d'accès:AKIA6ODU4YRF4HMY7D47, Clé d'accès secrète:**1saeAeqQ3HvnaGhdImGJy0JBWg1wAeHBRpV4VK2L

Shell scripting project used GITHub Api integration

For access repository in collaboratives need to write shell scripts github integration.we can two integration application one is api and other is CLI.

API: application interface, CLI: commande line interface

Github APIs( or Github ReST APIs) are the APIs that you can use to interact with GitHub. They allow you to create and manage repositories, branches, issues, pull requests, and many more.

Multistage and destroy-less image: Creating images in different stage, ex: build stage and run stage, destroy-less: docker runtime image.

Configuraion Managment

Why ansible:

What is configuration management: How devops engineer manage configuration of server/infrasturce.

3 principles tasks by devops in configuration: upgrades, secure pathes, isntallation.

Why ansible is preferable?: push model, agentless

Ansible develop by tema fedora

Disadvantage ansible:debuggint mode is not easy to standard devops, also some performing issue,

1. Ansisble uses language: python is uses by ansible.
2. Ansibel subpport both windows and linux: subpport linux ssh, support RM windows
3. Does ansible support all cloud provider?not matter to cloud provider( need ip address, and machine to access publically access ssh)ex azure,aws,gcp)
4. What is programming language uses ansible?YAML

Video 19- 15

**Why need container instead of virtual machine:**

**In virtual machine we can create lot of machine or instances and can run our application. Each machine or instances allocate memory and disk but alls are not use that's why need container. In a word too much physical server solved virtual machine and too much virtual machine extends container.If you deploy application in virtual machine you never seen message out of memory or storage so all time reste extra memory or disk , this problem is also solve by container.**

**Docker is containerisation platform.**

**A container is a bundle of Application, Application libraries required to run your application and the minimum system dependencies.**

**Containers vs Virtualmachine:**

**1. Resource Utilization: Containers share the host operating system kernel, making them lighter and faster than VMs. VMs have a full-fledged OS and hypervisor, making them more resource-intensive.**

**2. Portability: Containers are designed to be portable and can run on any system with a compatible host operating system. VMs are less portable as they need a compatible hypervisor to run.**

**3. Security: VMs provide a higher level of security as each VM has its own operating system and can be isolated from the host and other VMs. Containers provide less isolation, as they share the host operating system.**

### What is Docker ?:Docker is a containerization platform that provides easy way to containerize your applications, which means, using Docker you can build container images, run the images to create containers .(docker is an high level API which allow to containerisaton,

Container.d is runtime where is container is exexute)

Kernel: kernel is heart of operating system.



Docker daemon: is heart of docker A persistent background process that manages Docker images, containers, networks, and storage volumes, if daemon is down then docker will be down

### Docker LifeCycle

We can use the above Image as reference to understand the lifecycle of Docker.

There are three important things,

1. docker build -> builds docker images from Dockerfile
2. docker run -> runs container from docker images
3. docker push -> push the container image to public/private regestries to share the docker images.

#### Docker registrie

A Docker registry stores Docker images. Docker Hub is a public registry

#### Dockerfile

Dockerfile is a file where you provide the steps to build your Docker Image.

Difference between Github and Docker hub: both of versioning but github storing source code but dockerhub storing images.

Install docker in EC2: sudo apt update -y, install docker: sudo apt install docker.io -y, check docker: sudo systemctl status docker

If we run docker: docker run hello-world (it will not work because by default docker install everything in root user.so I need to add ubuntu user as docker group. Commande: sudo usermod -aG docker ubuntu and then must be restart if not will work.

Now it will be run: docker run hello-world

Now clone a project from github then enter into project,

Then build docker image :

docker build -t sohrab109/my-first-docker-image:latest **.** (here sohrab109 username, then create a repo like github, -t for tag latest, . is for create in same répo)

After build docker image run docker image:

docker run -it sohrab109/my-first-docker-image:latest

Now i want to share my docker image, I need to login and push:

Login:docker login success then:

Docker Volume and bind mount: if a container go down then log file also go down so admin cant monitor or may be lost information of all clients. To avoid this problem docker user Bind mount.

By default container does not have own file system thats why container very light weight by nature.It used of all ressources from host operation.Container can access only cpu, processeur,storage etc. from host.

Bind Mount: a directory inside into container which keep backup in host.

Volume: is offer better lifecycle,logical partition on host.

In a summary: in container raise three problems: 1. a container does not keep previous logs or auditing a company can get information of clients or others.

2.If two contaier front and back end try to share information each other if backend gone down there is problem persue means backup.

3.if app trying to read somes informations from host operating system it does not know how to read files from operating system.

Solve this problem docker propose solutions Bind Mounut and Volume,

Bind Mount: it allows to bind a directory inside into container with a folder on a host.(binding a specific directory on host with a contaienr)

Volumes: volume like as bind mount but also offer better life cycle.volume is logical partition on host.volume can manage by docker CLI.

Commande create volume: docker volume create volume\_name

To see list voulume: docker volume ls

Details where volume is: **docker volume inspect volume\_name**

For delete volume: docker volume rm volume\_name

Commande display first 5/6/7 things: docker images | head -5

Docker networking: is allows container to each other and core system.By default container can communicate with host or machine, container does not have complete operating machine. If we create a container by default container has different IP from host so normally form host to container not ping. But by default when create a container docker create a virtual network between host and container is called virtual bridge(called ethernet). generally when create a container it creates a network which difference from host but by virtual bridge container and host can communicate.

Docker offer three popular networking: virtual network and host network(container ip address machine itself) and third is overlay networking which is used by kubernetes.

Run a container: docker run -d --name container\_name image\_name:latest

Enter into docker container as root: docker exec -it contianer\_name /bin/bash

Now update : apt update then install ping

Commande : apt-get install iputils-ping -y

Check ping: ping -V

Now check ip address: go to terminal(out of root):docker inspect container\_name

This moment we can check network by ping command

Go to each container root then ping one to another :ping ip\_address

Check docker network: docker network ls

Create custom bridge/network: docker network create network\_name

Now create a container and assign network: docker run -d --name container\_name --network=creating\_network image\_name

Not create a container with host network: docker run -d --name container\_name --network=host image\_name (result: this network is null means it has network machine)

Voulume: create a volume anywere then do mount: (note mount must be creating new container time, not for container exist)

Command: docker run -d -p 8000:80 --mount source=nomvolume,target=/dest image

(dest is in container anywhrere)

Command for mount machine local: docker run -d -p 8000:80 -v ‘c:/../file: container\_path’ image

Docker commit:docker commit name-of-container (for create docker iamge)

List: docker images

Docker tag: docker tag image\_id new\_image\_name

Now we can delete container and recreate container from image

Summary of Dokecr image and container run commande:

Git pull

Docker docker build -t name\_dockerimage . (image)

Dokcer run -p 8000:8000 -it imageid (run container)

Creating a docker file:

From: base image container will build on top(node:14-alpine3.16)

Workdir: working directory for any commmand that follows in the docker file.(/app)

Any subsequent command in the docker file:Copy, Entrypoints, Run or CMD[]

Copy: copy local files from host to current working directory.(COPY . . means current to current directory)

Run: instruction to execute commands that will run during the image build process.(npm install)

CMD: the CMD instruction sets the command that will be executed when a container is run from the image.(["npm", "start"])

ENV NODE\_ENV production: instruction set environnement

Expose: which ports container will listen

Simple node application architecture

node-app/

├── Dockerfile

├── package.json

├── package-lock.json

└── app.js

Now create docker file:

# Use the official Node.js image as the base image

FROM node:14

# Set the working directory

WORKDIR /usr/src/app

# Copy package.json and package-lock.json

COPY package\*.json ./

# Install dependencies

RUN npm install

# Copy the rest of the application code

COPY . .

# Expose the port the app runs on

EXPOSE 3000

# Command to run the app

CMD ["node", "app.js"]

Python Flask appliction

flask-app/

├── Dockerfile

├── requirements.txt

└── app.py

Create docker file

# Use the official Python image as the base image

FROM python:3.9

# Set the working directory

WORKDIR /app

# Copy the requirements file

COPY requirements.txt .

# Install dependencies

RUN pip install --no-cache-dir -r requirements.txt

# Copy the rest of the application code

COPY . .

# Expose the port the app runs on

EXPOSE 5000

# Command to run the app

CMD ["python", "app.py"]

Requirements.txt

Flask==2.0.1

Java spring boot application

spring-boot-app/

├── Dockerfile

├── pom.xml

└── src/

└── main/

└── java/

└── com/

└── example/

└── demo/

├── DemoApplication.java

└── HelloController.java

Create docker file:

# Use the official Maven image to build the application

FROM maven:3.8.1-openjdk-11 AS build

# Set the working directory

WORKDIR /app

# Copy the pom.xml file and install dependencies

COPY pom.xml .

RUN mvn dependency:go-offline

# Copy the rest of the application code

COPY src ./src

# Package the application

RUN mvn package -DskipTests

# Use the official OpenJDK image as the base image for the runtime

FROM openjdk:11-jre-slim

# Set the working directory

WORKDIR /app

# Copy the packaged jar file from the build stage

COPY --from=build /app/target/demo-0.0.1-SNAPSHOT.jar ./demo.jar

# Expose the port the app runs on

EXPOSE 8080

# Command to run the app

CMD ["java", "-jar", "demo.jar"]

Pom.xml

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>demo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>demo</name>

<description>Demo project for Spring Boot</description>

<parent>

<groupId>org.springframework.boot</groupId>

<version>2.5.2</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

Nginx web server:

nginx-web-server/

├── Dockerfile

├── nginx.conf

└── html/

└── index.html

Create dokcerfile

# Use the official Nginx image as the base image

FROM nginx:latest

# Copy custom configuration file

COPY nginx.conf /etc/nginx/nginx.conf

# Copy website content

COPY ./html /usr/share/nginx/html

# Expose the port the server runs on

EXPOSE 80

Nginx.conf

events { }

http {

server {

listen 80;

server\_name localhost;

location / {

root /usr/share/nginx/html;

index index.html index.htm;

}

}

}

**Mulistage docker image**: For multistage application for exemple 3 tiers applicatoin react, java , mysql. The best practice first we start build a basic image like ubuntu (here why do not select java runtime beacuse we need lot dedependace image at the same time) all supports exists and after build image we delete the image and reduced total image size.

Without build multistage: need install all supports the regading java, then react and mysql. The build java application , front end and entry point , this maniere the docker image is too much .

We can solve it by distroless images: this very minimal stick image ligtweight docker image that will have only runtime environment. and multistage: build stage without entry point/cmd (may be one or more) then final stage with entry point or cmd

For go lang distroless image is : scratch, for other lang see <https://github.com/GoogleContainerTools/distroless>

Docker-compose

Install docekr-compose for debian 11

Install command:sudo curl -L "https://github.com/docker/compose/releases/download/v2.0.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

Permission: sudo chmod +x /usr/local/bin/docker-compose

Test: docker-compose --version

For systeme linux must install docker-compose but for system windows by default with docker.

The essenetial command of docker-compose:

Docker build: docker-compose up --build

docker-compose up -d vous permettra de démarrer l'ensemble des conteneurs en arrière-plan ;

docker-compose ps vous permettra de voir le statut de l'ensemble de votre stack ;

docker-compose logs -f --tail 5 vous permettra d'afficher les logs de votre stack ;

docker-compose stop vous permettra d'arrêter l'ensemble des services d'une stack ;

docker-compose down vous permettra de détruire l'ensemble des ressources d'une stack ;

docker-compose config vous permettra de valider la syntaxe de votre fichier docker-compose.yml  .

Delete cache and build new: docker-compose build --no-cache --force-rm

Stop container: docker-compose stop

Up container: docker-compose up -d

Update composer: docker exec container\_name bash -c ‘composer update’

Update composer: docker exec dbcontainer\_name bash -c ‘php artisan migrate’

Update composer: docker exec dbcontainer\_name bash -c ‘php artisan db:seed’

Jenkins

Machine Alma-projet=jenkins server, client\_alma\_projet=client

Pre-requis: java, git, jenkins

Jenkins is server integration

Point 1. Normally in jenkins its better to use docker agent beacuse we know docker is very light weigt. But if application has database do not use docekr agent.

In master node we do not install docker agent or another agent its better to install in slave node.

When modifysomething in jenkins its better to restart jenkins, https:ip\_add:8080/restart

Item : more than 4/5 years ago people uses freestyle project, in freestyle porject the modifictaion code means if add additional functionality all of team member will not see it.So this case pipeline method is better.

Pipeline: is declartive or scripted pipeline, by using declartive pipeline we can write our code and every body can see and access from github.

Two instances not pinging each other : for pinging each other

**Kubernetes**

Kubernetes container orchestration plateform.(un process isolé, namespace qui permet d’isole container et cgroup permet de limite ressource)

Create container in docker have 3 problems:1. one container impact another(if one conatiner take more ram and process may be other container will die).

2.auto healing: container start withself without intervention.

3.

Entreprise standard support: load balancer, firewall,auto scaling, atuo healing,api gateway.

But docker can not solve of all this probleme , this problem is solve by kubernets.

First probleme single host, auto scaling, auto healing and enterprise level support. Who is solving this problem in a word kubernetes.

Kubernetes: by default kubernets is cluster(group of nodes), kubernetes is installed in cluster. If a container affected whole application may be affected kubernetes immediately will put application in different nodes.(first problemes solved).By default kubernetes is cluster image so problem one is solved by kluster behavious of kubernetes.second solution tell kubernetes in yaml file increase container 1 to 10 tomorow have fastivel.(this is auto scaling)

Third problem: whenever healing kubernetes control and fix the damage.If a container down kubernetes immidialtely rollout new container end user does not understand one container down or something happen. This is auto healing.

In docker have not support enterprise nature like load balancer, firewall, support for api gateway, black listing,white listing etc. And docker has no support enterpirse solution that’s why docker never use in production.(for emxeple enterprise needs to list some client in black or white list but docker nerver possible).

K8s Architecture

Kubernetes have components data plane:kubelet, proxy, container runtime.

Kubetnetes create multiple node master and multiple node worker.

Kublet is responsible for running pod, pod is a collection of smaller container which have some advance capabilities.

By default kubernetes has feature called auto healing, if one pod down auto healing tell do something.

Kubernetes can support container D, creo, docker shim and any other contianer run times environment and support kubelet.

In docker has default bridge called docker 0 and in kubernetes has bridge called Q proxy and in every container and pod I have creating il will allocate ip address.

In worker node in kubernetes have three components : kube proxy, kublet,container runtime.

Kubelet: is responsible for creating pod and that the ensure pod is running state if it not it takes necessary action using kebernetes control plane and then we have Q proxy(kube proxy) which is responsible for the networking like generating ip address or load balnacing basically it uses ip tables your linux machine.

The components running appliction: cubelet is deploying, Q proxy is providing ip address, container runtime is providing the execution environnement for container. In spite of everyting why we need control plane tools? Because of needs some specific enterprise standards.

Now who is decided to create a pod and decided in node 1 or node 2 etc and may be need some more instructions so it should be has component which basically access core components kubernetes and takes all incomming request is called API server.In kubernetes Api server takes all request from externel world.

Api server is component that exposes kubernetes and receive all of request from external world.

For exemple a user want to create a pod and send a request to API server, Api server tell ok node 1 is free, schedule is a component on node in kubernetes is called scheduler.

The responsibility of schedular is scheduling on pod or ressource on kubernetes.who decides the information Api server and who acts on the information that is scheduler.

In production we need keep a bakcup entire cluster information there has to be a component in kubernetes basically act as backup service, basically act as a backing sotre of entire cluster information, etcd is basically key value store and entire information of kubernetes cluster information stored as object key vlaue pairs inside the etcd.

We have another components called controller manager and cloud controller manager Kubernetes. We know kubernetes subpport auto scalling for thats it has some controllers for exmple replica set, that is totaly controlled by controller manager.

Cloud control manager: we all know that kubernetes is run on cloud platform that is EKS/AKS/GKE etc.

In a summary: kubernetes has two component control plan, data plan.data plan have two worker node.

Control plan control the action and data plan executing the action.



How to manage hundreds of cluster kubernetes: exemple of kubernetes developper environments: minikube,kind, k3s,micro k8s, k3d.This should not use in production level.

How devops engineers manage kubenetes cluster in production systems.Before I need to understand what are the distribution of or popular distrubution kubernetes.In generally people will asked what distribution kubernetes you used in produciton,did you manage the upgrates of that specific distribution.For exemple if I use amazan Linux and its distribution the advangtage is they will ensure that whenever there is security patches or any of these things they upgrate on time to time basis its not that open source platform dont do it beacuse i am paying for Redhat distribution(exemple) so they ensure that your oprating system or your kernet is safe from all of security vulnerable.

Lets build software on top of kubernetes or list build distributon on top of kubernetes. For exemple amazon they have come up with its own managed kubernetes service the EKS, Rethat distribution called openshipt, Vmwire has TANZU, and other is Rancher so all of this are distribution of kubernetes.

Suppose if I create an ec2 instance in amazon and installing kubernets as your own if you has some issue so you can get a support tikcet with amazon beacuse you pay it.If you want support form us on kubernetes get into our manage kubernets service that is eks and there are many distribution is popular beacause distribution provide you support.

Kubernetes itself used in production firstly list like as: kubernetes->openshift->Rancher->Tanzu->EKS->AKS->GKE->Docker engine (note docker swarm is kubernetes distribution)

Now what is the difference between installing kubernetes directly versus installing mini cube ? if you install minicube thats means you are installing kubernetes with all the capabilites for enterprise, for exemple mini cube can run on a single node architecture where as like two cpus and 4 GB RAM, so dealing with cluster need more memory or cpu.

What is difference between kubernets and EKS: if I install a coupte of ec2 instances and install kubernets on top of it and make a cluster that means I am managing this kubernetes cluster and amazan can not provide me anykind of support (misconfig,any kind of issue) but eks I can get support from amazon, so this is the difference.

We use KOPS(kubernetes operations): install, upgrade, modification,deletion all of these call lifyecye of kubernetes that is manged by KOPS.

Install : minikube and kubectl (from officiel document)

Start kubernetes: minikube start

Create pods: kubectl create -f pod.yml

See nodes and pods: kubectl get nodes/pods

See details of pods: kubectl get pods -o wide

Details pods.=/debug pod: kubectl describe pod pod\_name

Logs pod: kubectl logs pod\_name

Get all namespaces: kubectl get pods --all-namespaces

Get eveything pod/node: kubectl get all

To see all namespace : kubectl get all -A

To see live show what is happening in the pod: kubectl get pod -w

Create deployment: kubectl create/appyly -f nom\_deployment

Deploy from local image must be done: minikube image load <image name>

Modify service: kubectl edit svc nom\_svc (if edit no need apply but if change service.yml must be apply command)

Delete deploy: kubectl delete deploy nom\_deploy

Delete service: kubectl delete svc nom\_service(default service can not be deteted)

Delete pod: kubectl delete pod nom\_pod

Creer replicat par commande: kubectl scale deployment/nginx-deployment --replicas=12

Note : -L for any direction without -L not wrok if application has redirection.

To enter into kubernetes pod: ssh minikube then ->curl -L <http://ip_pod>,

To create deployment from images: kubectl create deploy nginx --image=nginx

To see Continious error: kubectl get pods -w

Inspection: kubectl describe deploy deploy\_name

Edit deploy: kubectl edit deploy deploy\_name

Create service: kubectl apply -f service\_name (but service create in nodeport)

Note: in real time keubernetes not advise to create a pod directly , its better to create a deployment(for auto scaling and auto healing) with replicatset and with other sources then create pod;

Note: if create deployment at same time pod also created automatically

Note: in realwork we will never crate pod, we will crate deployment

Note: in kubernetes 4 principle(service, deployment, pod, replicatset)

Kubernets services: devops creates pod as a deployment in kubernetes, what is

Create service, then->ssh minikube->send request curl ip\_service

Can aslo access from minikube ip(minikube ip , curl minikube\_ip)

Now access from browser from outsite company

idel pods size the result is it depend on number of concurrent.

Kubernets has auto healing capabilites and container is epimeral by nature if container go down it will not come up similarly pod alos go down it will not comeup automatically unless dont have auto healing behaviour that is implemented by the deployment in kubernetes or replica set controller in kubernetes.

Now if pod is down replicaset tell dont worry i will create a pod and application came up but problem is it will change ip address of application.

So user can not access you application thats why kubernetes uses service.

So we can say we create deployment using 3 replicaset on top of using something called service.

Now how service offering load balancer, user access application via service not ip addreese.

So the advantage of service is load balancing, service discovery,expose application external world



Service discovery is label of pod , user can communiceate by label so if change pod ip adresse no impact of applicaiton.

Service expose external: allow to acces application from outside of kubernetes cluster.

When we wirte .yaml file default type: clusterip(communication inside kubernetes), nodeport, loadbalnceing, etc. Now from outside user cannot access pod ip address only can access node ip address.

To access cluster kubernetes from outside organisation we need to a public ip addresse and active nodeportmode and loadbalancer mode.

Nodeportmode: application ip addresse exposeon the node ip address and

Load balancer only work on cloud provider.

Service directly looking for selector by using the pod not deployment,seletcor is must same as template section label in pod.

Namespaces: are a way to organize clusters into virtual sub-clusers.

Create kubernetes cluster: kind is distribution of kubernetes, kisnd is very faster.commande: kind create cluster --name=k8s-troubleshooting

To see cluster: kind get clusters

**Troubleshooting and comman problem of kubernetes**

Command scenario problme:

-ImagePullBackoff(invalid image/tag/permission):

See information of errors: kubectl describe deploy deploy\_name

-ErrImagePull (see spec: container)

-image pulled but pod is pending

ResourceQuota on namespace?

-Requests and Limits set?

Node or Nodes lacks Resources ?

Also, check the kube-scheduler component

-RegistryUnavailable

-InvalidmageName

-CrashLoopBackOff

-KillContainerError

EKS Aws

Prequisites: install kubectl, eksctl, aws cli

Install kubeclt:

curl -LO "https://dl.k8s.io/release/**$(**curl -L -s [https://dl.k8s.io/release/stable.txt](https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl\")**[)](https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl\")**[/bin/linux/amd64/kubectl"](https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl\")

curl -LO "https://dl.k8s.io/release/**$(**curl -L -s [https://dl.k8s.io/release/stable.txt](https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256\")**[)](https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256\")**[/bin/linux/amd64/kubectl.sha256"](https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl.sha256\")

echo "**$(**cat kubectl.sha256**)** kubectl" | sha256sum --check

sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl

kubectl version --client

Install eksctl:

# for ARM systems, set ARCH to: `arm64`, `armv6` or `armv7`

ARCH=amd64

PLATFORM=$(uname -s)\_$ARCH

curl -sLO "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl\_$PLATFORM.tar.gz"

# (Optional) Verify checksum

curl -sL "https://github.com/eksctl-io/eksctl/releases/latest/download/eksctl\_checksums.txt" | grep $PLATFORM | sha256sum --check

tar -xzf eksctl\_$PLATFORM.tar.gz -C /tmp && rm eksctl\_$PLATFORM.tar.gz

sudo mv /tmp/eksctl /usr/local/bin

Install aws cli: sudo yum remove awscli

Curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"

unzip awscliv2.zip

sudo ./aws/install

Create cluster:

eksctl create cluster --name demo-cluster --region eu-west-3 --fargate

Delete cluster: eksctl delete cluster --name demo-cluster --region eu-west-3

Update cluster for acces with kubectl: aws eks update-kubeconfig --name your-cluster-name

Find out cluster command: eksctl get cluster --name cluster\_name nom\_region

URL du fournisseur OpenID Connect: for connect via facebook or github exemple\*

Profiles Fargate: bydefault Fargate attach to the default, kube-system means right now I can deploy pod only this two(default,kube-system) namespace.

(fargate profile configure which pod will be execute on aws fargate)

Create a Fargate profile: for define namespace

eksctl create fargateprofile \

--cluster demo-cluster \

--region eu-west-3 \

--name alb-sample-app \

--namespace game-2048

Ingress: route the traffic inside the cluster, instead of elastic ip or load balnacer

In the yaml file: there is kind: namespace, deployment, service and ingress

It works on the way of first ingresss controller send to service and service send to deployment with namespace.(https://raw.githubusercontent.com/kubernetes-sigs/aws-load-balancer-controller/v2.5.4/docs/examples/2048/2048\_full.yaml)

Get pods: kubectl get pods -n game-2048(namespace)

Check pods situation: kubectl get pods -n game-2048 -w

Check service: kubectl get svc -n game-2048

Check ingress: kubectl get ingress -n game-2048 (here there is no address beacause there is no ingress controller)

Now create an ingress controller: ingress controller need ingress ressource,

To crate ingress controller I need OIDC provider(authentication provider)

Final stage: go to alb connector add on

Note: ALB(application load balancer)

1. Configure aws cli: go to nom\_user->secruity credential then create key and secret acces.

Kubernetes Ingress:



ClusterIp: Used for communication between applications inside k8s cluster(ex: frontend application accessing backend applicaiton)

Nodeport:

KUBESHARK: for monitoring traffic in kubernetes.

**Namespace**:

In Kubernetes, a **namespace** is a logical partition within a Kubernetes cluster. It allows you to divide cluster resources among multiple users, teams, or projects

**In AWS EKS**, the control plane (master node) is **managed by AWS** and is not accessible directly. You interact with it using the API server, which you can reach via kubectl.

CloudWatch

The main goal to move cloud: reduce maintainance and improve security and third is cost optimisation.

**CloudWatch:is gatekeeper for AWS which will help understanding implementing the a monitoring, alerting, reporting and logging, by using this feature we can keep track of activites that are happeing on aws account . watch activites on aws.**

**Advantages of aws: can implement monitoring, to allow get reallife metrics.**

**Metrics: is helping understand utilistaion and details of aws services.exemple: metrics api call that application receive, metrics cpu utilisation, memory consumption.**

**Logging: communicate between services, exemple: ec2 trying to communicate with other services.**

**EC2 is default metrics in aws**

**Features of metrics: monitoring, reallife meteric, alrams, loginsights, custom metrics, (cost optimization, scaling)(can integrate with other services so we can use lambda function can helping cloud cos!!t.**

Go to cloudwatch on aws:

First to see logg : atutomatically created

Metrics: all metrics:then go to ec2 metrics->seletct ec2utilisatio(in the mean time create a ec2 instanec)

Then go to ec2 instance->details->monitoring all inforamtion collection from metrics(by default aws have ec2 metrics)

How freuent send information to cloudwatch

Metrics: metric is something that will help to collect information, information can be aws cpu, instance, disk input bye etc.

Metrics and Alerms work with each other closely, in metric we can say if cpu utiise more than 80% send me a alerm.

Log insights: is basically a capability of providing you an insights of which service is accessing the other service

Real-life metrics

Custom metrics

Aws lamda

Aws lamda is one of cost optimization,

Cost optimisation: most of organizations do cost optimisation day to day basis.

Why people move to cloud? :

Cost efficiency: pay-as-you-go model, no harware cost, reduced operation cost

Scalabiilty and flexibility: on-demand ressources

Disaster recovery and backup: built-in redundacy, automatic backups

Increased collaboration: remote access, real-time collaboration

Security: advanced security features etc

Cost optimisation: devops engineer to look if there any steel ressources( the ressource created but forgot to delete) or created too much ressources, it is not possible to see one by one this case we can use lamda function

Loadbalancer

Loadbalancer: the reponsibility of laodbalancer to manage request and to manage the load on machine (ec2 instance ex)

Basic loadbalancer is round robin which is blance load.

If user access loadbalancer Not slowless and downtown. Actaully user access loadbalancer instead of application directly load balancer will manage the reqest which server will send the request.

Some exemple of loadbalncer: Amabassador, NY,traffic, nginx etc.

Requset response completed in 7 layers.

Application layer 7 decieded what kind of protocal want to use to acces the server.

My reqeuest is sent to layer6 for add security ssl;

In layer 5 you create a session

Then request go to transport layer in this layer data will be splited into small packets.

Then it will go to network layer 3 and finally request got tho server(data and physical layer)

This is the entrie traffic

1. Loadbalancer: ALB, NLB,GWLB
2. Layer 7

If I need to perform on applicaion in layer 7 I need to user ALB(application loadbalncer)

If I nedd to perform on layer4 I need to user NLB(network load balancer)

In most time we use http layer means layer 7,

ALB: is costly and latency involved.

NLB: low latency, low cost, high transmission of data, so without interreption acces use network load balancer, most use game server video streaming etc.

GLB(gateway load balancer) for vpn kind of application, firewall application.

**How to put application in production in aws**

I nedd four basic:

1. Auto scaling group
2. Load balancer
3. Target group
4. Bastian host/jamp server
5. Create a vpc (option vpc and more)
6. Create auto scaling group(go to Ec2 instance and select auto scalinggroup)
7. For create autoscaling must crate template (machine )
8. Create model/template (fill all of neccssary field except subnet/sousresaux:this field is empty)
9. Now creating auto scaling: fillup necssary field, then fill vpc and subnet/sousreseaux: must be select 2 (availability zone)private subnet then

Next select no load balancer, scaling/echell , capacite souhaite 2(auto scalling 2 machine), capacite max:4, next... and create auto scaling group.

1. Now create a bastien host to access application where in auto scaling machine
2. Create a machine/instance normally, in security group add security ssh to acces bastien, network setting: select vpc (must be in same vpc of auto scaling group otherwise bastien not be able to access auto scaling )

**Auto public address: Enable**

1. **T**o access application via bastien host I need to copy pem key to bastien host

Then create a bastian host(create instance with same vpc) beacuse bastian host will access application. Bastian host in public subnet.

Now copy and send pem file/key to the bastian host from my desktop

Sudo scp -i /mnt/c/Users/moham/Downloads/master-key.pem /mnt/c/Users/moham/Downloads/master-key.pem [ubuntu@15.237.94.125:/home/ubuntu](mailto:ubuntu@15.237.94.125:/home/ubuntu)

1. Now connect from ubuntu/bastian-host with auto scaling machine where application will be loaded, ssh -i master-key.pem ubuntu@address\_id

Test access: this is python server: python3 -m http.server 8000

1. Now creat loadbalancer , go to ec2 select loadbalnacer

Type of load banalcer is application loadbalancer, vpc: as same vpc, both availabity zone, subnet normally public, for creating load balancer you need create target group like template,

1. Create target group: nom de group, type: instance, ...create

and access application

Create load balancer , in the load balancer must need create target group

Then register target: now select instaces(auto scaling instance not bastian)->include as pending->create

1. Now create load balancer: nom->acces internet->same vpc->availabity zone must be public->select target group->

Note: load balancer need target, auto scaling group need template

**S3(simple storate service) bucket**

**Charatistics of S3**

**-scalable**

**-high available**

**-secure and**

**-cost effective**

Data can be store and retrive from anywhere

Anything sotre into can be access by using http protocol.

S3 bucket has unique name (convention name : name-prod-companysite ) and it can by access from anywhere globally.

No more than 5TB data sotre in a single s3

30 video rest

**Ansible**

**Why configuration managements in devops: configuration mangement in serveurs or infrasturucture.**

Configuration management: Managing the configuration of multiple server: the tools for configuation management puppet, chef,ansible,salt etc. Ansible is push mechanism model on the other hands puppet is pull mechanism.ansible is agentless.

18 most interview question:

What programming language ansible uses?there is no requirements but i am comport with Python,

What protocol support ansible connect with windosw and linux: windosw ; winRM and Linux:SSH

Difference between puppet and assible?: push mechanisqme, agentless.

What language used for ansible: yaml

All cloud provier support ansible:?not matter cloud provider , only required to publicaly acessible or connect with ssh is allowed.

Sudo for root previlages.

Puppet uses puppet language but ansible uses yaml and ansible support windows server.

Install kubernetes: https://medium.com/@areesmoon/installing-minikube-on-ubuntu-20-04-lts-focal-fossa-b10fad9d0511

CI/CD PipeLine

Standard of organization for delivery a application: unit testing(unit test passed), static code analysis(), code quality/vulnerbility(code quality testing in different stage), automation(), reports(how many test passed, quality,experience), deployment(versioning in github and select which version to go client) etc.

What is CI/CD: when code commited into github repository that will pass by zenkins pipeline, that means zenkins is orchestrator.

This processing of CI/CD , application code into git->jenkins different mode dev, stage, produciton.

Installtion link:https://github.com/iam-veeramalla/Jenkins-Zero-To-Hero/tree/main

Jenkins and hudson are same tools pipeline CI/CD..Jenkins is a java based application so for jenkins needs to install java runtime.Jenkins is a Java-based open-source automation server.

Jenkins is an open source continuous integration (CI) server. It manages and controls several stages of the software delivery process, including build, documentation, automated testing, packaging, and static code analysis.

Check jenkins access port: **ps -ef | grep jenkins** il will be show jenkins access port 8080.

Now configure network securty:inbound security (select my pc/any can access ressourc type 0.0.0.0/0

To move root on ubuntu: sudo su -

EC2 instance(serveur\_ubuntu) root pass: ubuntu and admin user:hossain109, admin pass: \*Taspiasohrab109\*

Why used jenkins: for exemple a company have lot of teams if each teams work on same node (master node) may be conflits or will be loaded, thats why jenkins have multple node, one master node and others worker node, for jenkins pipeline we integrate docker beacuse advantage of docker no need to create virtual machine, creating just only container.

Now install docker:

Docker daemon: A persistent background process that manages Docker images, containers, networks, and storage volumes.

Root user can only acces doker daemon so need to give grant access jenkins to acces docker daemon then restart jenkins from browser

Argo CD is a Kubernetes-native continuous deployment (CD) tool. Unlike external CD tools that only enable push-based deployments, Argo CD can pull updated code from Git repositories and deploy it directly to Kubernetes resources.

1. What is real time ci/cd process?

There are two git repository in whole process of jenkins CI/CD, first one is source code and second is manifest.

First part up to docker is continuous integraton and second part is delivery integration

CI:is basically continuous integration, CI ensure that build is smooth and all tests are excuted, code quality is maintained, image is created into CI Process.

Whereas CD is continous delivery which ensure that your delivery process done so in our case we use kubernetes.

1. What type of agent you are unsing in jenkins: we are using docker agent which is very light in wegiht, very useful, no need to installation lot tools.(in general for build we use lot of tools).
2. What is artifact: Pipeline artifacts are packages that contain all the necessary components required to define and manage your CI/CD pipeline.

How ot explian entire process in interview: we have git repository where we have application source code, exemple java application->as developpeur raises a pull request to this git repo we have configured web hooks(http request connection between two app, here jenkins with git repo),using the web hook we trigger the jenkins pipelines. We have done declartive jenkins pipeline beacause declarative pipelines are easy pipeline to wirte and collaborate.Then process is complete in multi stage buiod, unit test , sast/dast(check app vulnarabilty), when every stage passed forword and create a docker image by using shell command and send to the container registry dockerhub,ECR of aws or play.IO. Untill all are CI Process.

Now In CD process: docker image is pushed to the elastic container registry ECR,Query.io we have a kubernetes cluster inside kubernetes cluster we have deployed two continuous delivery tools one is the image uppdater called Argo image updater and another is Argo CD. Both of this are kubnetes controllers that we have deployed on our kubernetes cluster. The first one image updater monitor image registry if any image is created it will update new iamge in another repository, This repository is purely for the image manifest that is Helm Charts or our customize or pod.yml

As soon as git repo upadted with new image then the other kubernetes contoller we have which is Argo CD , it takes the new image and deployed on kubernetes cluster this is our delivery process.

This is our CI and CD process

**Docker(https://labs.play-with-docker.com/)**

**What is Docker(un ouitls qui simplement creer, depoloyer, executer application en utilisant container.)**

* **Docker is a containerization technology.**
* **Docker is a platform for packaging, deploying and running applications. Docker applications run in containers that can be used on any system: a developers laptop, systems on permises or in the cloud**
* **Containerization is technology that's been around for a long time, but its seen new life with Docker. It packages applications as images that contain everything needed to run them: code, runtime environment, libraires and configuration.**

**Why do you need Docker?**

* **Portablity**
* **Performance**
* **DevOps**
* **Cross Platform**

**What can it do?**

* **Containerize Applications**
* **Run each service with its dependencies in separate containers**

**Docker Commands**

1. **Docker --version**
2. **Docker images**
3. **Docker ps //list docker running container**
4. **Docker ps -a**
5. **Docker run -d -p 80:80 nginx**
6. **Docker stop container\_id**
7. **Docker rm container\_id**
8. **Docker rmi image\_id**
9. **Docker start container\_id**
10. **Docker logs -f container\_id**
11. **Docker logs --since=2h -f container\_id**
12. **Docker exec -it container\_id bash**

**Docker compose**

1. **Docker-compose up**
2. **Docker-compose up -d**
3. **Docker-compose rm**
4. **Docker-compose down**

**Run Applications as Docker Container**

**Multi-stage Docker: for reduce size of application so starting application run time will reduced.**

**Installing docker**

1. **Sudo apt-get update**
2. **Sudo apt-get install docker.io**
3. **Sudo systemctl enable docker**
4. **Sudo groupadd docker**
5. **Sudo usermod -aG docker $USER**
6. **Newgrp docker**
7. **Docker --version**

**Download your first Image**

**Docker pull nginx**

**Docker pull sohrab/appname**

**Docker pull sohrab/appname:latest**

**Note: Make docker file then create image and finally create container which is running portable.**

**To see docker container means which container is runnging: docker container ps**

**-stop container : docker container id**

**-to see images: docker images(after pull image)**

**-build images then it will be convert into container: Docker run -d -p 80:80 --name create\_container\_name shawon10/pythonapp**

**-stop service : docker stop container\_name**

**Git clone the project then enter the project.**

**Build docker as docker image: docker build -t imagename . (here t is target)**

**Check docker image: docker images**

**Run docker image as docker container: docker run -d -p 80:80 --name create\_containername imagename**

**Check docker container: docker container ps**

**To enter container: docker exec -it containername/id sh**

**Then enter: cd usr/share/nginx**

**(it interactive terminal)**

**Make docker image/Build docker image: docker build -t pythonapp . (here .means docker file in same file otherwise to show where is docker file like as ./foldername) (t means command run in termainal)**

**Run project or build container: docker run -d -p 4200:80 --name create\_containername imagename (here d means detach mode(run in background): I dont want to show log file, what is runing).P is port forwarding on which port want to run)**

**To enter docker image file: docker exec -it imagename**

**Show running container: docker ps**

**Show stop container: docker ps -a**

**Remove stop container : docker rm container\_name**

**Start container: docker start container\_name**

**Best practice note: multistage,naming conversion(image name=project name),**

**Enter into image: docker exec -it 798bdc524756 /bin/bash**

**Without enter image and see details: docker exec -it 798bdc524756 df -h**

**Without enter image: docker exec -it 798bdc524756 find / -name nginx**

**Difference between copy and add:Docker use copy and add at same purpose they let you copy from specific location into a docker image.**

**Copy <src><dest> || Add<src><dest>**

**Little difference is: in ‘add’ compress file converted into uncompress then work like as ‘copy’**

**Installing Docker another train**

**Install docker engine: sudo apt-get update**

**Sudo apt-get install docker-ce docker-ce-cli container.io**

**Here ce=community edition, cli=docker engine,container.io=docker damen**

**(note first create virtual machine then give ip address remove docker command if exists)**

**Remove docker engine if exits: sudo apt-get remove docker docker-engine docker.io contianerd runc**

**Check dockers elements: sudo systemcli status docker**

**follow:https://docs.docker.com/engine/install/ubuntu/#install-using-the-repository**

**Inspect docker image: docker image inspect docker\_image**

**To find image: find / -name image\_id**

**Running a basic web server: docker run -it --rm -d -p 8080:80 --name nginxcontain3 nginx (this command for running and create container)**

**Create a dockerfile and push into dockerhub**

**Create a Dockerfile and index.html then copy /usr/share/nginx/html/index.html**

**Then build docker file: docker build -t imagename .**

**Then run docker file: docker run -idt --name containername -p 8080:80 imagename**

**Push: docker login**

**docker tag firstimage:latest sohrab109/firstimage:latest (tag for image)**

**Docker push sohrab109/firstimage:latest (push)**

**Do link dockerhub with container: docker run -it -rm -d -p 8080:80 --name containername -v~/webcontent:/usr/share/nginx/html nginx**

**Link: https://www.stacksimplify.com/aws-eks/kubernetes-for-absolute-beginners/kubernetes-for-absolute-beginners/**

**Create docker network**

Why need to create network: allow containerized apps to function similarly to network services running directly on your host.

1. Docker network create chatbot
2. Docker network ls
3. Docker network inspect chatbot
4. Docker inpect --format = ’{{range.NetworkSettings}} {{ipaddress}} {{end}}’ chat\_app

****

**Docker 23**

**Linux**

**Linuxtricks.fr and quickref.me**

**sohrab@lenovo: ~$**

**Ici sohrab nom de utiilsatuer, lenovo nom de host, ~ dans le repartoire utilisateur, si c’etais root # au lieu de $, $ indicate utilisateur non privileger.**

**Super utilisateur: su -**

**Changer privillege: sudo**

**Open root fichier: nano /etc/sudoers (to see all rights and change)**

**To see all file even hide: ls -a /file nom**

**To install must be copy .bashrc user to root**

**Copy cp /home/user/.bashrc /root/.bashrc**

**Commande: /etc/.bashrc**

**To enter nano commande for configuartion nanorc: nano /etc/nanorc (nano is editeur)**

**Change color: to enter nano /root/.bashrc and line 60 to change color(**

**Test : echo $PS1 (majuscule ps)**

**Commande for give debian offciel source install of all application : nano /etc/apt/sources.list , then type: deb <http://deb.debian.org/debian> bookwrom main contrib non-free non-free-firmware ,and search documentation officel**

**Then commande : apt update && apt upgrade -y**

**For to see all privilizes: sudo /etc/sudoers**

**For restart machine: Init 0**

**Cloning a virtual machine**

**Create a snapshot then edit machine->option->advanced->enable template**

**Then click on machine: manage->next->select snapshot->clone link**

**Change usernmae: su - then hostname deb1**

**Then enter into machine : nano /etc/hostname**

****

**Deb1**

**Vm->setting->network adapter->lansegment**

**Then**

**Service network manager: nano /etc/network/interfaces (configuration)**

**Commande : systemctl stop NetworkManager**

**Commande: systemctl disabled NetworkManager**

**Commande: ip a flush networkcard**

**Commande: systemctl restart networking.service**

**Check adress: ip a**

**Config deb2: echo deb2> /etc/hostname**

**Service network manager: nano /etc/network/interfaces**

**Vm->setting->network adapter->add 2 and 3 network adapter and select lansegment 1 for network adapter, lansegment 2 network adapter 2, network adapter nat**

**Configure network card: ..pic**

**Commande : systemctl stop NetworkManager**

**Commande: systemctl disabled NetworkManager**

**Commande: ip a flush networkcard**

**Commande: systemctl restart networking.service**

**Check adress: ip a**

**Routage for diefferent network (ex: deb1 and deb3):**

**Nano /etc/sysctl.conf**

**Active line 28 , by remove # //now different network communicate established.**

**Then for access internet**

**Sysctl --system //for display**

**Update commande: apt update && upgrade**

**Commande : apt install iptables iptables-persistent -y (next config No, NO, NO)**

**Do Nat: commande: iptables -t nat -A POSTROUTING -o ens37 -j MASQUERADE**

**To see all tables: iptables -t nat -L**

**This commande take iptables Iptables-save> /etc/iptables/rules.v4**

**Deb3:**

**Vm->setting->network adapter->lansegment**

**Then**

**Service network manager: nano /etc/network/interfaces (configuration)**

**Commande : systemctl stop NetworkManager**

**Commande: systemctl disabled NetworkManager**

**Commande: ip a flush networkcard**

**Commande: systemctl restart networking.service**

**Check adress: ip a**

**Some commands linux:**

**Useradd -D -s /bin/bash (D:default, s:shell)**

**Change mot de passe direct: echo ‘solo:pricesse’ | chpasswd //auto change et enter user solo)**

**Commmande2: export PS1=’\u-\h:\’**

**Commande: bash pour reviens initials**

**Pour un nouveau bash il faut aller sur linux home->.bashrc->ecrit**

**export PS1=’\u-\h:\$’;enregistre et ouvrer nouveu bash**

**Commande echo $SHELL(majuscule) il affiche nom shell lorsque appel d’authentification**

**On peux verifier dans quelle shell on travaille on peux lancer cette command echo $SHELL**

**Commande: ls (lister repartoie)**

**Commande: ls -l(pour plus d’information)**

**Coomande: ls -l -t(trier par plus dsc)**

**Commnade: ls -l -t -r(trier par asc)**

**Commande: ls -lrt (tous contenu recurrent)**

**Commande: ls -lrth (comprendre humanin)**

**Commande: type echo/type bash (affiche qui situ ou)**

**Commande: file /usr/bin/bash (commande pour voir file)**

**Commande: id (affiche pas ou situe la id)**

**Commande: echo $PATH(majuscule) (pour voir ou situé id la arborescence)**

**Commande: help (pour aide) pour particulier: help echo (aide a echo)**

**Commande interne et externe (interne commande en meme fichier, externe commande un autre fichier exemple: type id )**

**Commmande interne: help echo , commande externe: id --help**

**Plus details aide: man id (plus details)**

**3 aide resume: commande interne: help cmd, externe:cmd --help,plus details: man cmd**

**Avantages du Bash:**

**Commnade: history(toutes les commande deja passe)then pour recupere le commande on peux saisie: !numero\_commande**

**Commande : CTRL+R then cherche le mot (pour recherche une commande)**

**On peux creer un alisa aller go : home->bashrc>ecirt alias pour exemple: alias ll: ‘ls -lrtha’**

**Commande: alias (lister tous les alias)**

**FHS Linux: /boot : noyau linux**

**/bin: commandes**

**/etc : fichier de configuration**

**Commnade: /root compte super utilisateur**

**Ne pas stoker dans le fichier root**

**Le répartoire /home (il permet stoker les fichiers compte perso d’utilisateur)**

**Le commande: /proc et /sys (ces sont vitrine virtuelle elles ne sont pas dans le disque dur, linux maintien en mémoire.**

**Le répartoire /home equivalent /var/www pour le serveur web**

**A partir du racine du systeme /**

**Commande: /root (pour stocker les donées personelle d l’administrateur)**

**Root est le compte privilégie de linux, il doit etre reservé exclusivement aux taches administratives sur le systeme.**

**Le repartoire /usr ,qui contiens deux fichier /usr/bin(ce répertoire contient majoritairement les commandes à destination de tous les utilisateurs du système, privilégiés ou non), /usr/sbin(ce sont des commandes à nouveau à destination unique de l'administrateur, mais non critiques pour le bon fonctionnement du système)**

**Repartoire /var, /usr en lecteur seule.**

**Afficher les contenu de fichier par cat(concaténer):**

**Commande : type cat (affiche**

**Commande: cat /etc/passwd**

**Commande cat: cat -n /etc/os-release (affiche le contenu de fichier avec le numero)**

**Deux commande meme temps: cat -n /etc/os-release /etc/passwd**

**Less: permet de visualiser les fichier sous linux.**

**Utiliser les canaux de Linux: stdin, numero 0(standard input),sdtout numero 1,(standard output),stderr, numero 2(stardard error)**

**Grep : chercher le ligne de mot adjectivement. Exempmle: grep debian /etc/os-release**

**Grep -rnio debian /etc/\* (o isoler le pattern dans la ligne, n le numero ligne, r la partie recursive, i le pattern censive la case)**

**L’editeur sous linux: vi**

**Vi nom de fichier: exemple: vi /etc/fichier1**

**Lettre I pour inserer /ecrire**

**Then Exc ou Echap pour passer en mode commande etpuis ajoter type A pour append.**

**Ajouter une ligne au dessous rester en mode commande + Shift+O**

**Supprimer une ligne rester en mode command etpuis DD**

**Pour copie rester en mode commande + YY etpuis bouge cursor et P**

**Replace une lettre rester en mode commande etpuis slectioner la lettre type R**

**Pour mettre le numéro type: :set number, save and exit :wq! /:x!**

**Principle command linux: des I nodes: ls -li**

**Copie et coller un fichier: cp ficheir1 fichier2 , voir i nodes: ls -li (different i node fichier1 et fichier2)**

**Move un fichier : mv fichier1 fichier2 (voir ls -li meme indice fichier2 et fichier1)**

**Faire lien entre deux fichier ln: ln fichier1 fichier(voir nodes meme node que fichier1 et fichier**

**Faire lien symbolique ln -s: ln -s fichier1 noveaunomfichier**

**Difference entre lien dur et symbolic (symbolic un lien comme raccourci)**

**Droits des fichiers: chmod u+rwx fichier**

**chmod u=rwx fichier**

**puis chmod g=rx fichier**

**puis chmod o=r fichier**

**Chmod calculate rwx=r:2²=4,w:2=2,x:2°=1(total droit 4+2+1=7)**

Commande linux

Bin:biniare un lien raccourcir

Boot: pour grab2

Dev: device/material

Etc:dossier configuration

Home: repartoire toutes les utilisateur

Lib: toutes les libraires pour system

Media: toutes media

Var: toutes les logs stokcer dans le var

Create new user: adduser nom\_user

Display user: cat /etc/group

Connection with Alma Lunux: ssh NomdeMachine@ipaddress

Ssh: chiffrement asymetric(demande de deux cle public et prive)

Les tache a faire:

. Bridge

. Créer les users

. Envoyer ses clés au serveur

. Modifier le server pour qu’il n’accepts quels personne qui ont envoyé la clé

Projet Answer:

Create user: adduser name

Give mot de passe: passwd name\_user

Then Active Bridge: on virtual box network adpater Bridge

Create key via SSH

Then create key :

Summary of connect with server from windows if already have created (reste in root then..): dir ->cd .\.ssh\->dir->rm .\known\_hosts

Create hidden file in linux: mkdir .ssh

To see hidden file: ls -a

Create key: ssh-keygen -t rsa -b 2048(b for sizz, t for type cryptage)

Phasephrasse: a key for authentification.

It will create two key public and private. Now send public key to server

Send public key to client: commande: scp C:\Users\

scp C:\Users\moham\.ssh\id\_alma.pub mohammad@192.168.1.31:/home/mohammad/.ssh

(scp:exchange between two, first part is source, last part destination)

Renmae key: enter .ssh/ ->mv id\_alma.pub authorized\_keys

The target of Authorized key is connect with server distance without password, but this case must be connected with private key:

Commande: ssh -i C:\Users\moham\.ssh\id\_alma [mohammad@192.168.1.31](mailto:mohammad@192.168.1.31)

When block from server we can not connect with server, ssh [mohammad@192.168.1.31](mailto:mohammad@192.168.1.31)

Permission denied. If we connect ssh -i C:\Users\moham\.ssh\id\_alma [mohammad@192.168.1.31](mailto:mohammad@192.168.1.31) by this (means with key) then connected.

Configure server: /etc/ssh/sshd\_config

After change port of server: restart server: systemctl restart sshd

Commande: ssh -i C:\Users\moham\.ssh\id\_alma [mohammad@192.168.1.31](mailto:mohammad@192.168.1.31) -p 60001

<https://fr.wikiversity.org/wiki/Certification_Linux_LPI/Administrateur_syst%C3%A8me_d%C3%A9butant/Examen_101>

Procedure Password recovery:

1. Redemarer de l’equipement
2. Interommpe process de boot(bios->bout louder(GRUB: linux, MBR/GPT:window)->noyaux->unit.d
3. En tanque bip choisir le deuxieme option apres swap ligne de commande: rd.break enforcing=0 , ensuite CTRL+X, apres ls voir sysroot,
4. Apres commande:mount -o remount, rw /sysroot
5. Apres commande: chroot /sysroot
6. Then automatically enter into noyux sh-5.1(5.1 vesion)
7. Then commande : passwd
8. Give strong password
9. Then commande: touch /.autorelabel
10. Now start linux , commande: exit, exit (2 times)

Informatoin /etc/passwd, /etc/shadow

MFA(double authetifiation), Clé PKI

Respecter RGPD

## **Configurez les cartes réseaux**

Configurer le nom réseau du serveur:

Regarde hostname: cat /proc/sys/kernel/hostname

Autre facon par noyux: sysctl kernel.hostname

Details de hostname: hostnamectl

Set hostname: hostnamectl set-hostname sohrab

Entre dans le shell: bash

### ****Détectez les interfaces réseaux de votre système****

Commande : dmesg (display message du noyaux pour essayer de détecter les cartes réseaux reconnues par le noyau lors du démarrage)

Liste des networks: dmesg | grep Network

Liete des différents des reseaux: ls -l /sys/class/net/

### ****Configurez les cartes réseaux de manière dynamique****

Ifconfig commande ne pas utilsé depuis 2009, e profitant pas non plus des nouvelles fonctionnalités des noyaux Linux.

Commande ip adresse toutes les configuration reseaux: ip a

Commande lister carte reseaux: ip link show

Ajouter une adresse ip: 192.128.1.24/24(exemple) eth0 (ethernet)

Delete adresse ip: 192.128.1.24/24 eth0

### ****Configurez les routes et les passerelles****

Voir toutes les routes: ip route list

Pour enleve le route par defaut: ip route del default via default\_route\_ip

Ajoute une route: ip route add default via new\_route\_ip nom\_carte\_reseaux(beacause of route add in same network card)

Ajouter sous reseaux en meme carte reseaux: ip route add 192.138.1.0/24 via ip\_carte\_resaux

Verifiy route list: ip route list

Renseigner les fichiers permettant d'utiliser des services de résolutions de noms de domaine:

## **Connectez-vous à distance avec SSH**

Commande install: sudo apt-get install openssh-server

Enable ssh service: sudo systemctl enable sshd

Voir le server port et ip: ss -lptun

Verifer la service running ssh: systemctl status sshd

Install ssh-client:

Commande installation ssh-client: sudo apt-get install ssh-client

Verify ssh client: ssh

Connexion avec serveur: ssh username@ip\_adresse

Récupéré ip address: ifconfig

Déconnexion: exit

Peut connecter également par nom de machine:ssh [nom\_de\_machine(sohrab@..)](mailto:nom_de_machine(sohrab@..))

Déconnexion: exit

Visauliser le cle public en tant que connecter avec serveur: cat .ssh/known\_hosts

Génération de cle ssh:

Commande genere la cle: ssh-keygen -t rsa (-t rsa:algorithm,

## **Transférez des fichiers par le réseau**

deux logiciels les plus utilisés pour télécharger des fichiers en HTTP sur le réseau, à savoir wget et curl.

Install wget: sudo apt-get install wget

Telecharger fichier: wget lien

Install curl: sudo apt-get install curl

Telecharger fichier: curl lien

Summary of linux:

3 layer of OS Linux: Noyux, Driver, Libraray

Redhat: alma Linux, centos, Rochez linux,Fedora

Debian:une version ubuntu

When start OS , the OS start by BIOS, BIOS is checked of all hardare ex: cpu,memory etc.

He will boot, first boot CD/ROM, Disque dur, Support externe, Reseaux,

Disque dur: Sequence d’amorsage (MBR/GPT)

For linux GRUB( noyux)

2 memoire exitw: meomire morte: Flash, HDD, SSD,NVME, Meomoire vivre: RAM, CACHE(3 niveau cahce l1,l2,l3)

Disk:material en totale

Volume: file system

Partition: il y debut en fin

Start of VMWire 1990

Exemple of hypervisor type 1: proxmox, ESXI(vmwire),Citrix, hyper -v-microsoft,KVM

Installation Alma Linux:

New machine->typical->selection iso->if not detect iso , slelect redhat entreprise linux9->single files->next et disk personalisé for fichier /12g, /var2g,/home2g,/tmp 1g,/boot 1g, swap2g

Configure Windows server-22

Mettre l’infrastructure system composer en server: router, server DHCP, client

Give a ip static ip address , via carte reseaux->TCP IP: 192.168.10.1

Then rename PC parametre->renommer pc->redemarrer

Then Clone server vm->manage->clone->next->clone link->terminer

Configure network card in differents server. Right click on server on tehn settings: Srv2 : create a network adapter and then network adapter 1 for lan segment 1, network adapter 2 for lan segment 2. Then configure srv1(main window server22) lan segment 1 and srv2 lan segment 2

Then rename and configure Adresse ip each server:

For ping must be desactive fire-wall: start->outil de addministration->fire -feu windows defendeur->propriete windows defandeur->desactive 3 profile : profile domaine,profile prive, profie public.

Now can ping in same reseau , for ping another network need to cinfigure with router:

Sur srv2 install routage: Start menu->Gestionare de serveur->ajouiter des roles et des fonctionlalites->suivant->suivant->Access a distance->routage->suivant .....->installation

Start menu->outils administration->routage et acces a distance->Right click on srv2->configaure et routage et access a distance->Configuation personalise->suivant -> routage reseau

Serv1: install role Serveur DHCP: start menu->gestionare de serveur->ajouter role et fonctions->suviant ->.....->click on server DHCP->ajoter des fontionalite->suivant ...->installtion

Si notification for configuration pour server DHCP , installer et vailider

Start menu->outils adminstration->click on DHCP

For dynmique DHCP must create nouvelle entendu

Creation d’entendu

click droite ipv4->Nouvelle entendu->suivant..->nom de lan ->........

Routeur jamais fait les request de Broadcast, Routeur fait toujour unicast

Configure en ligne de commande:

Serveur

Windows server: 15/04/2024

Active directory: une service d’annuaire(repartorier),Identification,Authentification.L'Active Directory peut être votre base d'identité numérique pour l'entreprise.

une base de données et un ensemble de services qui permettent de mettre en lien les utilisateurs avec les ressources réseau dont ils ont besoin pour mener à bien leurs missions.Active directory fonctionne en 2FA.

Az-900 certification

Server apache(Apache est le serveur web. Son rôle est d'écouter les requêtes émises par les navigateurs (qui demandent des pages web), de chercher la page demandée et de la renvoyer), server IIS:Ce rôle est composé notamment de IIS (Internet Information Service), qui n'est autre qu'un concurrent du serveur Apache couramment utilisé sur les systèmes Unix/Linux.

Install winwos server(best practice) on active directory:Service annunaire Qui permet de identification et latuhentifction

Install on virtual machine: typical->image CD I will install later->use as single file and size min:35->then choice image->then before install right click on server for configure boot->power->on firmwire->configure boot->change boot order->boot roder->cd drive as first boot dirve.->enter->select language->select system as user experince->accept licence->second option personalisé->next->termine

Install windows 22: essentials: coute 900bals, standard : est gratuit 180 jours, si on fiat lingne de commande in powershell slmgr --rearm, il vas renouvler 6 mois trois fois.,Datacenter: Azure Edition

Configure windows server2022 on Active directory:

Change name of the machine(identified the server and easy access) systeme->rename pc, IP address Fixe(ip address same network as connect pc: window+R ->ncpa.cpl->Tcp/IPV4->property->ip address same network as connect pc, mask auto, DNS prefere same as ip address), Now configure another pc ip address same way same network ip address and DNS is server DNS.

Then do jonction between server and client(go to client pc parametre->parametre advance du systeme->ordinateur->modifier(rename or change domaine)->domain=give server name (authentification server name and password)

Server windows2022:Add role and functionalities Gerer->add role and functionlalites->next...->selection des roles serveurs->DNS,AD DS add fontionalities ->next....->installer->promovoir (from notification)->AJouter nouvelle foret(Bangladesh.local)->next->mot de passe->next..->

Establised Active Diretory: (gestionaire serveur)Outils->users et ordinateurs active diretroy->dans le foret create a folder(corporation name) right click->nouveau->unite d’organisation(exemple BDCORP)->then in BDCORP create different service(folder)ex RH,TECHNO right click->nouveau->unite d’association->then create user and groupe user inside servce(RH,TECHNO) right clicl->nouveau->user, for user group on top small icon, right click on user->add in group

GPO: create a lecteure mapping ouitls->gestion des groupes et strategies->select folder corporation/service(inside foret) right click->creer GPO avec strategies->then double click on lecture mapping and add group or user for this GPO->right click on GPO->modifier->preference->mappage de lecture->then right click on empty field->nouveau->lecture mappe->emplacement(copy path sharing file name start with [\\)](\\\\)) and select a drive mostly distance.

Sharing file: right click on file->property->partage->recherche des personnes->add user for sharing this file

Check client: go to client pc first connection user name: prenom.nom, password=given serverpc and give new password.

Share a file on active directory by script: create a sharing file and inside create cmd file and give path as content of cmd file, exemple: test.cmd into file=net use L:\\..\. In the mean time sharing file right click->propery->partage->pargage->assign name for sharing this file

Create GPO: outil->gestion des groupes and strategies-> right click on file/service creer GPO->name gpo->double click->add group for this GPO->right click on lecture mappe->modifier->strategie->parametre windows->script->ouvrir une session->ajouter->nom de script path sharing folder

For install windows server: change nom de machine(identified the server and easy access), IP Fixe(must fix a ip address for connect with another machine), DNS(give a domaine name systeme(name:a-z,A-Z,0-9,-,\_,)),mis en place AD/ADDS(active directory). then promovoir controller de domaine. (installer dns,, adds then promovoir ->creer un foret et give DNS)

Rename: system parametre: rename and restart, open all network cards: cmd->ncpa.cpl

FOR troubleshooting we can follow two methods: 1. model OSI, QQOQCP,l'arret d'ishikawa



Windows server: utilisateurs et ordinateurs Active directory->un dossier(creer service)->create user->

GPO(groupe Policy Oject) by this tool we can install and control goup pc.

Testing in powershell: nslookup

Adresse ip APIPA: si deux server DHCP confilt l’adresse ip commnecer par 169....

Open connection a distance on run exe: mstsc.exe

Connect with diiferent netwrok in virtual box, must configure address with port: Edit->virtual box network->change settings->Nat setting->add-> map port

Desactiver parefau

Ordinateur->parametre windos->parametres de scurite profile domaine->inactive

Cloud computing: SAAS, PAAS, IAAS(on-promise)

On-promise vs cloud:

Sommaire mis en place server microsoft et

AUTOMATISER LA CREATION D'UN DOSSIER UTILISATEUR AVEC UNE GPO

Configure windows 2022 without graphic: open windos powershell: CNTRL+SHITP+ECAP ->ficher->new exeute tache->name of applicaiton(ex:powershell.exe)

Commande: sconfig permet de configurer windows server

Commande : hostname for name of pc

commande: netipinterface for to see details interface network

Commande: new-netipaddress -interfaceindex 3(numero ethernet) -ipaddress 192.168.1.65 -prefixlength 24 -defaultgateway 192.168.1.65

Commande: ipconfig /all

Dns configure: set-dnsclientserveraddress -interfaceindex 3 -serveraddresses 127.0.0.1,8.8.8.8

Verifier all: ipconfig /all

Test account connection: whoami

Commande to see all privilage: whoami /priv

Commande install AD DS: install-windowsfeature ad-domain-service -includemanagementtools

Commande install forest: install-addsforest -domainname ‘BANGLADESH.LOCAL’ -domainnetbios ‘BNAGLADESH’ -installdns:$true -norebootcompletion:$true

Commande retart: restart-computer

Commande to see domain : get-adddomain

Cloud computing devOps 3 principles: Iaas: ressrouces(cpu,ram, reseuax, stockage), PaaS:environement dev(faire environnenment runtime), SaaS:service logiciel / rendre service plus souvant sur cloud.

Cloud: acceder ressources via internet. 3 type de deploiment en cloud: public(azure,aws,GCP)pay n go, privee(lorsque entreprise decider creer son propre cloud) avantage: gouvernane,souveraineté(ex:banque),hybride, (commmmunataire)

30/04/2024: Securiser un infrasture: blocquer tous les autre ports

Comment verifier port:

Run script on powershell test in powershell: Get-ExecutionPolicy (if remoteSigned means script local function but other script from different pc not work)

For set policy: Set-ExecutionPolicy name\_policy (Restricted=stop, All Signed=all script,unrestricted)

Powershell un autre maniere: run->powershell\_ise.exe

Open notepad from powershell: start-process notepad //open notepad already open

Open new notepad: start notepad

Equivalent of echo in powershell: write-host

Automatisation Voir VSPHERE( ESXI ) , type hypervisor 1, 2....

Installation vagrant and virtualbix

Note start with vagrant file( create a file and into browse by powershell then->vagrant init->then vagrant file modifed for deploy machine->vagrant up)

Infrastructure: ensemble des machines.

vagrant cloud(search image corresponding needs)

Vagrant: un outil permet de deployer des outils provider(fournisseur) appel automatisation.

Provider: CPU, RAM, I/O(stockage), Réseaux

Install : vagrant et virtual box (https://developer.hashicorp.com/vagrant/install windows amd64)

Then: crate a file : mkdir ubuntu, after enter into ubuntu: cd ubuntu

-> commande: vagrant init ubuntu/trusty64 (verify fichier ls)

->commande: vagrant up

->it will create ubuntu on virtaul box

->for destroy machine: vagrant destroy

Comment mettre la machine en bridge directement.

Vagrant up->vagrant ssh->sudo su->ifconfig

Now add linge: config.vm.network "public\_network" ->vagrant reload->now check in network adapter it will add a new bridge.

Comment choisir la taille/sizing

vagrant plugin install vagrant-disksize

Add file in vagrantfile: config.disksize.size = '50GB' then vagrant reload

Install vagrant et systeme sous Windows

Install vagrant: download vagrant from: <https://developer.hashicorp.com/vagrant/docs/providers/vmware/installation> ->vagrant utility download->download and install

Install vagrant plugin desktop by powershell: vagrant plugin install vagrant-vmware-desktop

Then mkdir AlmaLinux9 ->entree Almalinux9->chercher image sur vagrant cloud ->vagrant init nom\_image

Then -> vagrant up

->connecnt with machine(AlmaLinux9)->vagrant ssh (note in vmwire no physical interface and connection must be stay in file AlmaLinux9

Declarer variable in vb in this line: Vagrant.configure("2") do |vb|

Plus detials: https://developer.hashicorp.com/vagrant/docs/providers/configuration

Then configure vagrantmakefile a, then->

Vagrant global-status

We can run file for virtual box

Vagrant up --provider=virtualbox (by default vmwire)

Display of all network: window+R then ncpa.cpl

Automations infrastructure de machine linux par Vagrant

Creer vagrant file: 3 Almaxinux, box:almalinux /8, bridge,cpu,meomry

Connect with machine: cd /.vagrant ->vagrant ssh nom de machine

Install nmap sous windos:

Sous powershell: nmap network plage(192.168.1.0) -F

Comment trouver 3 addresse mac de machine sur virtaul box on avait creée.??

Une fois vagrant file est fait on peux pas modifier, si on veux modifier beaucoup de machine on doit faire automatisiation Ainsible

Ainsible connecter with shell linux ssh, with window winrm

Creer un key dans le server par root : entree dans root(su -) ->commande: ssh-keygen -t rsa -b 4096

Ensuite envoie au client: ssh-copy-id vagrant@ipa1(ip client)

Maintenant on peut connecter a partir de client: ssh vagrant@ip\_adress

Le cle on peut trouver meme endroit que server vagrant->ssh->nom de cle autorized\_key

Tp 1.. chaque machine creer 2 users/password :osadmin,genesys

2.Creer les cles ssh suivantes

srv-ans: /home/osadmin/.ssh/id\_rsa,

srv-ans : /home/genesys/ainsible\_key

Cli-01: /home/vagrant/.ssh/key

3.envoyer les cles suivate:

Srv-ans:genesys ->genesys@cli-01

Srv-ans:osadmin ->osadmin@cli-02

Cli-01:vagrant->vagrant@cli-02

Answer:

1.Create user and password: enter into root in each machine : adduser name\_user, passwd name\_user

2.create key: enter into machine and user needed(by command: su name\_user) ssh-keygen -t rsa -b 4096 then choice paths according listed)

3.Send key: enter into machine and user needed then commande ssh-copy-id -i path(when created key).pub user\_name @ip\_address

Pour tester connection: vagrant ssh nom de machine then swith to user as you want

06/05/2024

Mot de passe systeme: Inaya109 et mot de passe foret : \*Taspiasohrab109\*

**CCP 1 Automatiser le déploiement d'une infrastructure dans le cloud distance**

Creer un virtaul windows server then

Sysprep: un outils pour cloner windows,mais attention avec cet ouitil vas suprimer tout les fichier

Cet outil doit etre faire sous virtual machine..

1. Install windows then windows+R sysprep:(mode audit+redemarrer) ok : now mode edit for customize
2. Go to gestionnalire de sereur: seveur local: disactive : configuration internet exlorer et desactive: tous les parefeu
3. Now select the option obe+ generilseé et arret la systeme
4. Then clone windows machine: open virtaul machine choice master->.ovf
5. Open machine clone: see all soft and configuration are same as master.
6. Ejecter dirve (c drive click a droite ejected) then also settring cd/dvd decocher le iso+decorcher power on
7. Restart la machine clone
8. Win+R sysprep: option mode oobe+generalise+
9. Create network for each machine (network setting edit->network setting)+ decroche dhcp ip reseaux
10. Give network to each machine
11. Configure ip address and renmae pc (serveur local from gestionnaire servuer) et give network name on reseux

Faire routage: gerer->access a distance ->chocher routage ->install then

Tools click on acces distane et routage ->click a droite ->click active routage

Commande:

Update commande: get windowsfeature

Install ADDS: install-windowsfeature -name AD-Domaine-Service -includeManagementTools (si sucess et tools ok ADDA est installe)

Create forest: install-ADDForest -DomainNmae «greta-Mohamad.local» -installDNS

Role active directory check: Install-WindowsFeature -Name AD-Domain-Services -includeManagementTools

We have done atuomationsation infrastructure, virtualiseation et configure reseaux.. ccp1

Active directory: centralise les utilisateur pour gerer, LDAP etc.

13/05/2024 (Migration controller de domaine)

1. Install core windows
2. Then change name pc by typing
3. Change network by typing 8 (net work adapter network select NSL)
4. Configurer server DNS: configurer carte reseaux 8->network address:1->DNS server Preferer :2 put AD-01 (must be switch on AD-01)
5. Restart pc
6. Test on core1 by commande line: nslookup dnsname(greta-mohammad.local)
7. Do junction between AD-01 et Core1 , by ligne de commande: add computer (or sconfig number1 :nom computer)
8. Add-computer -DomainNmae greta-mohammad.local -Credential greta-mohmmad\Administrateur -restart -force
9. Test go to user and active direcotry ->domain name->computer->name computer( AD-02)
10. Ajouter un controlleur domaine la domane deja existante
11. Must install ADDS en tant que admin: (install-WindowsFeature -Name AD-Domain-Services -IncludeManagementTools)
12. Installe les roles: install-ADDSDomainController -DomainName greta-mohammad.local -Credential (Get-Credential) AD-01.greta.mohammad.local -InstallDns -ReplicationSourceDC
13. Creer utiilsateur (start core->ECHAP->ECHAP
14. Test Core is admin see in AD-01 contoller AD-02 (gerer->user active directory->contoller donmain)

Changment routage by vyos (for configure carte each time command: configure then command set interface then commit )

1. Change router name: commnande: configure
2. Set system host-name router\_name(RTR-04)
3. Commit
4. Add two network on settring NSL, EXT
5. Test (show interfaces , ip a)
6. Set interfaces ethernet eth0 address 192.168.10.253/24(ip address must be before existing because 254 exist)(must be ensure that network mac address is equal which one to configure:)
7. Now command: commit then save then exit then reboot
8. This stage router1/vyos1 ok
9. Now configure static route for others networks to access internet
10. Now configure :set protocols static route 192.168.2.0/24 next-hop 192.168.255.252
11. Now configure :set protocols static route 0.0.0.0/0 next-hop 192.168.255.254

Now can acces internet from vyos1(ping: 8.8.8.8)

14/05/2024

ESXI for creation des virtual machine , VCenter whose controll all cluster ESXI, for create more machine need to cluster,

10.05.2024

## IMG-20240507-WA0004

## Note: connect with different network router in same interfaec we must need gateway, but without sameinterface connect with differnt network we must need routage configure.Router has never gateway address..ip address last one : broadcase, before last one: gateway, firstone :network address. Ip address start from 0

[https://github.com/vyos/vyos-rolling-nightly-builds/releases/download/1.5-rolling-202405100019/vyos-1.5-rolling-202405100019-amd64.iso](https://github.com/vyos/vyos-rolling-nightly-builds/releases/download/1.5-rolling-202405100019/vyos-1.5-rolling-202405100019-amd64.iso" \t "https://meet.google.com/_blank)

2 core, 3 vyos

Page de garde

Sommaire

Presentation perso(parcours - choix devops - represente devops)

Projet(pk avoir choisi ce projet)

Problemethique

Outils

6/7

Table de matiere

Niveau de police roboto/

Taille de police 11

14/05/2024

administration un evironement(creer reseaux, machine,etc) vsspare "gestion de cluster, gestion vcenter, gestion mis en plance datasotre"

1. Install deux esxi, pfsense, TureNas, vcenter, etc

Link installation ESXI: https://computingforgeeks.com/install-vcenter-server-appliance-on-esxi-host/

Terraform +open tofu open source

TP CCP1: create 3 machine with vagrant(automatisation) connection via protocol ssh

TP infrastructure microsoft

Ccp3 : TP creation de systeme reservation

03/05/2024

Installation ESX6: Creer une virtuelle machine , typical, ressourece enregistrer sous racine C(non document..) processor: enleve virtaulise intel vt,

Choise iso ESX6:

Configuration ip et gateway et dns: regarde network setting le reseaux ip: edit->virtaul networking

Ensuite dans ESX 6configure network manager, dns, Ipv4 address et desctiva ipv6

Virtual machine : settings->processing->enleve cohce vlt

Active windows composer/winodws composer : plateform hyperviosor dechroché

Securite windows->desactive isolation noyux(integratlite du memoire)

Apures powershell en tant que admin: bcdedit /set hypervisorlaunchtype off et remearrer

21/05/2024

**Ecrire un role NFS(server des fichier)**

**Role nfs:**

**Cli-01 server nfs**

**Cli-02 client nfs**

Load after change: sysctl -p

From: /etc/sysconfig/network-scripts/

-> cd nom\_de\_ethernet(cd ens160)

En suite change : onboot:yes et save

->shutdown -rf now

22/05/2024

Sudo shutdonw

CronTab: planification des tache

Jinja2

Insallation Ansible 8 minimul iso pour Ansible

docs.ansible.com/ansible

Sommaire

SUDO: Cela signifie que l'utilisateur user peut exécuter n'importe quelle commande en tant que n'importe quel utilisateur ou groupe, sur n'importe quelle machine (utile pour les systèmes multi-utilisateurs ou en réseau).

SU: The su command in Unix and Linux systems stands for "substitute user" or "switch user." It allows a user to start a shell session as another user, typically the superuser (root).

Yaml est un langauge de description comme xml, json etc

Handlesr: Les handlers sont un mécanisme puissant dans Ansible pour gérer les actions de suivi après des changements d'état du système

/etc/hosts (information de domain name corresponding ip address)

1 . At first broadcast up et disable ipv6

1. Pour Debind l'ipv6 3 solution : NMTUI et modifier la connexion , disable ipv6 et active connect automatic et puis activer une connexion
2. modifier le fichier /etc/sysctl.conf et ajouter les lignes suivantes : net.ipv6.conf.all.disable\_ipv6 = 1 net.ipv6.conf.default.disable\_ipv6 = 1 Puis sauvegarder et relancer avec la commande sysctl -p
3. Sinon modifier le fichier /etc/default/grub Y ajouter la ligne GRUB\_CMDLINE\_LINUX="$GRUB\_CMDLINE\_LINUX ipv6.disable=1" Sauvegardez et quittez puis relancez un nouveau fichier grub avec la commande grub2-mkconfig -o /boot/grub2/grub.cfg

Check dhcp ip addresse : ip a

1. Changer nom pardefaut de pc by ligne de commande hostnamectl set-hostname XXX
2. Add user and his password then user mod group: add user

adduser XXX

passwd XXXX

usermod -aG wheel XXX

Verifer user dans le group wheel: cat /etc/group

Verifier sudo :sudo vi /etc/passwd

1. Creer clé ssh au serveur et enovyer aux clients (attention user)

Ssh-keygen -t rsa -b 4096

Ssh-copy-id user@ipadresse

1. Installation ansilbe (attention user , en tant que user osadmin pour exemple)
2. Installation du repo il faut faire epel-release: est un dépôt qui fournit des paquets additionnels pour les distributions basées sur RedHat

Cd /etc/yum.repos.d

Sudo yum install epel-release -y

Note: pour ansible classic besoin de faire epel-release mais pour ansible core redhat deja dans repo

1. Installation de ansible: Sudo yum install ansible -y
2. Editer le host ansible : il faut mettre addresse ip de tous les client dans un group

Exemple: sudo vi /etc/ansible/hosts

[nom\_group]

Addresip1

Addressip2 etc.

1. Collecter les information sur les host

Ansible -m setup nom-group

La commande ansible -m setup host est utilisée pour collecter des informations détaillées (appelées "faits") sur l'hôte spécifié (host)

1. Tester toutes configuration ok par ping

ansible -m ping nom\_group

1. Creer playbook et lancer playbook

Un playbook Ansible est un fichier écrit en YAML (Yet Another Markup Language) qui définit une série d'instructions à exécuter sur un ou plusieurs hôtes (machines cibles). Les playbooks sont le moyen principal par lequel les utilisateurs définissent des tâches d'automatisation et de gestion de configuration dans Ansible.

Creer playbook: sudo vim createDirectory.yml

Execute playbook: ansible-playbook nom\_playbook -vK (v for message error, K for become:Utiliser become: true permet d'élever temporairement les privilèges de l'utilisateur courant pour ces tâches spécifiques.)

Exemple ecrire un fichier yaml pour creer un directory:



1. Repondre

Ansible-galaxy init nom-de-galaxy

5.2 mkdir -p roles/nfs\_client/{tasks,defaults,templates,handlers}

**roles/nfs\_client/tasks/main.yml** roles/nfs\_client/tasks/main.yml :

---

- name: Install NFS client package on Red Hat-based systems

ansible.builtin.package:

name: nfs-utils

state: present

when: ansible\_os\_family == "RedHat"

- name: Install NFS client package on Debian-based systems

ansible.builtin.package:

name: nfs-common

state: present

when: ansible\_os\_family == "Debian"

- name: Create mount point

ansible.builtin.file:

path: "{{ mount\_point }}"

state: directory

- name: Add NFS mount to /etc/fstab

ansible.builtin.template:

src: fstab.j2

dest: /etc/fstab

owner: root

group: root

mode: '0644'

notify: Mount NFS share

27/05/2024 (Dossier partage) **Automatiser le déploiement d'une infrastructure dans le cloud Prés**

Summary of file sharing

1. For atuotmation need a ansible server and need at least two clients
2. Configure dossier hosts(ansible: /etc/hosts)

[nfs\_server] to define a client as server

192.168.178.130

[nfs\_client] to define a client as client

192.168.178.129

Define password for nfs\_server to avoid each time put password:become true

[nfs\_server:vars]

ansible\_become\_pass=Inaya109

Define password for nfs\_client to avoid each time put password:become true

[nfs\_client:vars]

ansible\_become\_pass=Inaya109

1. Installation des pacquets
2. Create du repartoire a partager
3. Modifier le fichier /etc/export /dossier partage ippartage rw
4. Lancer et relancer les service

Note: for sharing file server\_nfs(clint) firewall must be stop:

sudo systemctl stop firewalld

Df -f to see la partage sur le

Install yaml lint: sudo yum install yamllint -y

Execute playbook: ansible-playbook nom\_playbook -v( in /etc/ansible, Not k because password already define in hosts)

**Automatiser le déploiement d'une infrastructure dans le cloud Prés**

**28/05/2024**

Playbook,

Important command for test: shutdown ip\_address,

Showmount ip\_address

Ping ip\_address

Stop firewall: sudo systmctl firewalld

Sudo yum install net-tools -y //for verifier ports with ipadress

Sudo netstat -lpant

Start nfs server: sudo systemctl start nfs-server

Maintenant duex playbook dans un seul playbook et viabiliser

Sudo rpm -qa | grep nfs

Ansible -m setup alma for affihc\_er tout les variable en format json

Verifiy httpd active ou desactive: sudo systemctl status httpd

Start httpd: sudo systemctl start httpd

Disabled firewall: sudo systemctl stop firewalld

29/05/2024:

Auotmatisation deplploimnet server web et server mysql.

Sudo yum search nom-de-software

Ngois

Nagois (serveur)+1 client

Un outils de supersvison

. 1 metrologie(rr

1. Monitoring (qui
2. Un tutorial ecrit.
3. Un playbook avec des roles pour l’automatisation (commentaire + justification)
4. 1 presentation

Install serveur nagois,

1. Install necessary dependencies.
2. Download and install Nagios Core.
3. Download and install Nagios Plugins.
4. Configure the Nagios web interface.
5. Start and enable the Nagios service.
6. Set up an admin user for the web interface

<https://computingforgeeks.com/install-and-configure-nagios-on-debian/>

Playbook in Ansible

1. Create and send ssh key
2. Create a file into username (exemple /home/osadmin ) ansible
3. Then create hosts and and playbook
4. Run playbook : ansible-playbook -i hosts nom-the-playbook -v

Deploy HAPROXY

1. First install http in alma and apache in ubuntu/debian into two different server (enable httpd and disaple firewall)
2. Then configure each hosts ( 3 machine info) : sudo vim /etc/hosts put each ipaddress nom\_domain nom\_server (3 machien information on each machine; 2 server web, 1 haproxy)
3. Install haproxy into 3rd server (haproxy server)
4. Now systemctl start haproxy
5. Finaly test by curl local host (send request http)
6. Note check firewal stop or not and check httpd/apache2 active or not

Rediect ipadress into domain: sudo httpd -M | grep rewrite //enable rewirte mode

Now open /etc/httpd/conf/httpd.conf see if not exits add : LoadModule rewrite\_module modules/mod\_rewrite.so

Configure HAPROXY;

Entrer /etc/haproxy/haproxy.cfg

fornend app\_server \*!80

default\_backend apps

Backend apss

Server domain

Server domainhttps://www.youtube.com/watch?v=TkiGgUkn\_PI&t=16s

Now retart haproxy: systemctl restart haproxy

Check port 80 active or not : netstat -natlp | grep 80

Check log errors: tail -f /var/log/messages

Veuillez trouver ci-jointe mon cv au format pdf ou word pour un stage dans le domaine de l’informatique et si possible avec des compétence DevOps.

Le stage est conventionne et non rémunéré, il débutera du 2 septembre 2024

Bien a vous /Cordiales salutations

ERRATUM:object

ANUULE ET REMPLACE

Offre de emploi/proposition d’offre

Veuillez trouver ci-jointe mon cv concernant l’intégration en stage informatique(DevOps)

Le stage est conventionne et non rémunéré, il débutera du 2 septembre 2024.

Au plaisir de vous lire.

Je suis attiré par votre entreprise, je peux vous apporter mon expertise sur les thématique suivantes:

- Le DevOps (Docker, kubernetes..

- Microsoft Azure

Disponible actuellement pour un échange.

Bonjour Madame, Monsieur,

Veuillez trouver ci-jointe mon cv concernant l’intégration en stage informatique(DevOps)

Le stage est conventionne et non rémunéré, il débutera du 2 septembre 2024.

Je suis attiré par votre entreprise, je peux vous apporter mon expertise sur les thématique suivantes:

- Le DevOps (Docker, kubernetes, Ansible, Terrafomr..

- Aws, Microsoft Azure

Disponible actuellement pour un échange.

Administration suvervision avec nagios

Aide au stage, welcome to the jungle, indeed, linkedin

12/06/2024

Installation d’exchange serveur et serveur ADDS.

Lien de tutoriel suivi: <https://www.it-connect.fr/installation-pas-a-pas-de-microsoft-exchange-2019-sur-windows-server-2022/>

A partir de exchange server creer un utilisatuer on peux voir dierectement le user sur serveur ADDS sur user list apres l’user peux connecter et peux envoyer mail

1. Service mail partager, 6 users, 2 group partage et donne de droit
2. adressipchnageserver/0wa
3. Adressipexchangeserver/ecp

Creae docker file for laravel project.

FROM php:8.1.0-apahce

WORKDIR /var/www/html

«Mod Rewrite

RUN a2enmod rewrite

https://medium.com/@sushantkumarsinha22/dockerize-your-php-laravel-mysql-application-94333d0a1f46

--mount source=name, target=/name nom\_server

17/06/2024

Exam ansible et instattion de jenkins

Ansible a connecter avec client sans agent via ssh , du c’est accessible très facile. Il est déconseille install playbook sur machine ansible.

18/06/2024

Jenkins(3 machine jenkins server=alma projet, client connected ssh=client\_alma\_projet, client sans ssh= web2)

1. Installaiton jenkins from officeil document jenkins
2. Note: éviter build et test sur serveur jenkins toujours en client
3. Agent permanent: un agent permanent toujours resté a connecte (marché et écouté)
4. Nombre de execution: Nombre de build simultane sur ce machine(is beet
5. Agent: toujours envoie client vers server
6. Voir port et adresse ip commande: sudo ls -lpant

Now connected with second client with same agent without ssh.

Sans ssh connecter avec clinet il finxer un port sur Administrer jenkins->secriite->agent port->fixe (exemple 5000)

1. Install git and then install maven sur serveur jenkins
2. Check version: mvn --version
3. Application serveur web avec maven il pusher sur git
4. Create pipeline lorsque push jenkins prendre et compiler avce maven build la application sur client passe par agent

Apahce dirctory for distribution : https://downloads.apache.org/

Mettre en production, Automatisation, déployer en manière continu but de devops

Summary of Installation and troubleshooting Jenkins

For installation follow officiel documentation according oprating system: <https://www.jenkins.io/doc/book/installing/linux/>

In my case Almalinux8. For almalinux 8 must install open-jdk -11 otherwise jenkins not work because almalinux8 compatible with jdk11.

Start jenkins:

Enable: sudo systemctl enable jenkins

Start: sudo systemctl start jenkins

Stop: sudo systemctl start jenkins

Then by default jenkins run on 8080 port. Access jenkins on browser <http://ipaddress:8080>

Then unblocak jenkins by put password: cat path

Troubleshoot:

To see jdk version installed: sudo alternatives --config java

To see used port: ls -lpant

Note: If connected with ssh no need to install maven or the dependancies it will work directly But if connected via Agent must install maven or the dependicies on Agent.

Stop firewall permanent specific port : sudo firewall-cmd --add-port=8080/tcp --permanent

sudo firewall-cmd --add-port=8080/udp --permanent

sudo firewall-cmd --reload

Commande : Top permet de voir tous les processus de linux

Container: (namespace et control group)un processus isole des autres.

Socket: un interface fichier reseaux qui permet de commeqeunt entre eux

Command: commande for suprimer tous les container?

Dockerization Laravel application

Php artisan config:cache

Php artisan key:generate

Php artisan

Create .env file from .env-exemple

01/07/2024 (mot de passe et id pardefaut: glpi/glpi)

Outils monitoring et supervision: GLPI, et add plugin FusionInventroy

Link: <https://faq.teclib.com/03_knowledgebase/procedures/install_glpi/#secure-mariadb-installation>

For time zone: instead of (mysql\_tzinfo\_to\_sql /usr/share/zoneinfo | mysql mysql)

Commande execute: mysql\_tzinfo\_to\_sql /usr/share/zoneinfo then

Mysql | mysql and execute myql line

Id : glpi, password: root, database: glpi, password: root

Now connnect glpi with server ADDS -> confiugure( login glpi)->authentification->Annuaire ldap->ajouter->

Check version glpi: enter into ->var/www/html/glpi: cat inc/define.php | grep "'GLPI\_VERSION'"|cut -d "'" -f 4

Enable service maintainable: sudo php bin/console glpi:maintenance:enable

Work with GLPI:

1. Ticketing
2. Gestion de parc informatique hadware
3. Gestion de licencing
4. Supervision du materiels-> des incident informatique
5. Utilisation local ou distante
6. Commnet un user peut signler un incident informatique?
7. Telephone
8. Difference incident/probleme/une demande : marche mal/marche pas/demande quelque chose.

Migration inventroy: /var/www/html/glpi/plugin/fusioninvetroy/setup.php

Setup terraform and configuration

Setup on workspace on github:

1. Create a repository on github
2. Click sur workspace open with vs code(click on code->workspace normally open with vscode)
3. Install tools : click on search button->then put > wright dev container->modified dev container
4. Now search tools exemple like: terraform (see on case is tick) ->ok->

TP Docker 02: 02/07/2024

1. Hello From alpine

1.1 Docker run --rm alpine echo hello

1.2 Docker run, rm ,image

1.3 sudo systemctl start docker

Docker run -it alpine

1. Shell interactif

2.1 Docker run -it alpine

2.2 Ls

2.3 Bin/sh

Commande top affiche les processus en executant

2.4 Ls->cd.....

2.5 docker run -it alpine->apk update->apk add curl

1. Foreground/Background

3.1 docker run --rm alpine ping 8.8.8.8

3.3 docker run -it alpine ping 8.8.8.8

Docker attach ping 8.8.8.8

Architecture microservice(mis a jour, peut utiliser differents langages, si module tombe en penne l’autre n’impacte plus)

Appuyez sur Ctrl + P suivi immédiatement de Ctrl + Q

-d lancer un container si en mode arret (backgournd):docker start

Normalement docker run -p

Docker ps -a -q(for afficher only ids)

Recover ip add: dokcer inspect -f {{.NetworkSettings.IPAddress}} nom\_container

Doker run -d --name ping alpine ping 8.8.8.8

Docker container logs ping

Docker exec -it ping /bin/sh ->top

Delete all container live & dead: docker rm $(docker contaienr ps -a -q)

Apk add curl dans le container ->curl www.google.com

Deployer l’application sur cloud public un serveur serveur eks

Jenkins, git , container , kubernets, aws

Delete all container: docker rm $(docker ps -aq)

Delete all images: docker rmi -f $(docker images -aq)

Install php,mysql,myadmin on :https://www.malekal.com/installer-php-ubuntu/

03/07/2024 Docker

Ubuntu

Server ansible

Avec clé ssh

Fichier inventory

2 projets:

Vagrant+ansible (deploiment kube)

23/07/2024

Automatisaion depolioment kuster kuberntets avec l’outils vagrant+ansible

Etape1: vagrant va deployer les vms

Etable2: ansible vas faire la post-conf

Etape1: vagrant vas deployer 3 machines

1. Master kub
2. Worker
3. Serveur ansible

3 Almalinux 8

!!! il faut que l’environemenet ansible soit pret a l’emploi

- installation ansible

-createion d’un user specifique

-generation de cles

- envoie de le clé

-droit sudo sans mot de passe(ssh)

Attention aux pres -requis kubernetes sur les machine(VCPU et RAM)

Etape2: Ansible qui déploit le cluster

Role qui déploit les pres requis(sur touts les machies)

Role Master

Role worker

Autre possbilite

Role sui deploit les pres requis (sur toutes les amchines)

Role qui deploit Master+worker

scrappe les articel ouitls python , une pi traduction, blog tribu, basculer

Aws CICD pipeline

First Build a project: on aws->Code Build->Build Project->name & Description

Source ->Fournisseur de source->github

Repository->repository public->url github

Connect with github->github with help of OAuth

Environnmement->image gerer-

Systeme exploitation->ubuntu, Environnement d’execution->standard, image->latest,

Role de service->Nom de role

Note: this role actually for permission for docker

Give permission: got to IAM role->add autorisation->add strategy->search by ssm(select ssmfullaccess)

Fichier Buildspec->specification de la génération->inserer des commnades de generation->busculer ver fichier

Create role-> go to IAM->Role->create new role->service aws-

Select cas de utilisation->code build->suivant->also suivant->create(attention later I need to add politique)

Add plolique: awscodebuildreadonly as IAM

Configurer buildspec fichier:

Now create parameters from aws system manager for environnemental variable for buildspec(to push docker image): create new parameter (go to gestions d’applications->stockage de parameter)

Name of parameter: name like : /app/docker-credentail/username ->chaine securite

Valuer: docker\_username(sohrab109)

Then go to build projets->details project->edit buildspec

version: 0.2

env:

variables:

DOCKER\_REGISTRY\_USERNAME: /app/docker-credential/username

DOCKER\_REGISTRY\_PASSWORD: /app/docker-credential/password

DOCKER\_REGISTRY\_URL: /app/docker-credential/url

phases:

install:

runtime-versions:

python: 3.11

pre\_build:

commands:

- pip install -r requirements.txt

build:

commands:

- echo "build simple python application"

- echo "$DOCKER\_REGISTRY\_PASSWORD" | docker login -u "$DOCKER\_REGISTRY\_USERNAME" --password-stdin "$DOCKER\_REGISTRY\_URL"

- docker build -t "$DOCKER\_REGISTRY\_URL/$DOCKER\_REGISTRY\_USERNAME/simple-python-app:latest" .

- docker push "$DOCKER\_REGISTRY\_URL/$DOCKER\_REGISTRY\_USERNAME/simple-python-app:latest"

post\_build:

commands:

- echo "build successful"

Note: by default aws codebuild not permitted to create docker image, I need to give privilige go to app->edit->suplimentaire->click on enable docker image

At the same time update github buildspec file

Now invoke pipeline for do automatically instead of manually:

So go to CodePipeline: creer un pipeline

Name of pipeline ->new role service by default il will be display the role ->next->

Source->gihub version2

Connect with github-> name of application(random name)->autorisation->then select list conncection or nouvelle connection navigeur le repository souhaite ajouter

Select repository name->branch name

Select code pipeline bydefault->next

Etape generation: select aws codebuild->name of project->build type:build unique

Etape deployment: this is CD part skip this->next

Summary: when code change in commitcode or github codepipeline will invoke code build atutomatically

Now part of CD;

Go to aws CodeDeploy->

Le 26/08/2024

Outils de IaC Populaires: ansible, aws cloudforamtion,Terraform, chef, puppet etc.

Idempotence: Les scripts ou fichiers de configuration IaC sont conçus pour être idempotents, ce qui signifie que les exécuter plusieurs fois produira toujours le même état final, sans créer de duplications ou d’erreurs.

Terraform: un outil pour la gestion des infrastructures

Language:description commen ansible

Pre-requis:

1. Almalinux 9
2. Docker
3. Terraform
4. Aws-cli
5. Localstack

Emulation: replicating the functions and behavior of one system

Simulation: Recreer un system avec des materials differents

Virtualisation: Presenation virtual

At first create credential:

Demarrer localstack:

Start localstack: loacalstack start -d

At first create credential:aws configure

Aws configure set aws\_access\_key\_id hossain

Aws configure set aws\_access\_secret\_key Inaya109

Aws configure set aws region us-east-1

To see services: localstack status services

Number of s3 affiche: aws s3 ls --endpoint-url http://localhost:4566

Create s3 bucket: aws s3api --endpoint-url http://localhost:4566 create-bucket --bucket mydata

Display s3: aws s3 ls --endpoint-url http://localhost:4566

Create services aws:

1. dynamo db:

aws dynamodb --endpoint-url http://localhost:4566 create-table \

--table-name Artist \

--attribute-definitions \

AttributeName=Artist,AttributeType=S \

AttributeName=SongTitle,AttributeType=S \

--key-schema \

AttributeName=Artist,KeyType=HASH \

AttributeName=SongTitle,KeyType=RANGE \

--provisioned-throughput \

ReadCapacityUnits=5,WriteCapacityUnits=5 \

--table-class STANDARD

1. Instance ec2: aws ec2 --endpoint-url http://localhost:4566 run-instances \

--image-id ami-0abcdef1234567890 \

--instance-type t2.micro

1. Utilisatuer IAM: aws iam create-user --user-name MyUser --endpoint-url http://localhost:4566
2. File sqs: aws sqs create-queue --queue-name MyStandardQueue --endpoint-url http://localhost:4566
3. Function lamda(server less:evenment,calcul)

Terraform(provisonning): HASHI corp programation language(declaratif)

The file principle main.tf

Create file: main.tf then initialise terraform init

->terraform plan (for change execute)

->terrafom apply (execute apply)

Deploy appaliction go

1. Run the application in go environnement
2. Create multistage dockerfile for create image and push into dockerhub
3. Then create deployment.yaml, service.yaml and ingress.yaml(there are differents types of ingres, need to choice hostname wildcards)
4. Now install eksctl, kubectl and aws cli
5. Configure aws cli for create cluster kubernetes by command line
6. Install Install a EKS cluster with EKSCTL: eksctl create cluster --name demo-cluster --region us-east-1
7. Now apply appliaiton: kubectl apply -f k8s/manifests/deployment.yaml, kubectl apply -f k8s/manisfests/service.yaml, kubectl apply -f k8s/manisfests/ingress.yaml

Kubctl get pods, kubectl get svc, kubectl get ingress(here normally we can not see ip of address, the target is assign a ip adress and forward to host)

1. Now edit svc type: kubectl edit svc(type:NodePort), check NodePort service running or not: kubectl get svc(result: type=NodePort, port=80:32099)
2. Take an address ip of any node (check : kubectl get nodes -o wide), now can access the application by address ip of node and node port, 54.12.52.125:32099
3. Now install ingress controller : kubectl apply -f <https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.11.1/deploy/static/provider/aws/deploy.yaml>
4. Check pods with ingress controller(kubectl get pods -n ingress-nginx
5. Now check ingress controller give a dns name or not: kubectl get ingress(it will give a loadbalnacer domain name)if it access a domanine name then go /etc/hosts and put ip address correspondance hostname
6. Install helm and create a folder help enter into helm and run the command: helm create go-web-app-chart
7. Insert into go-web-app-chart : delete charts folder, empty template and copy 3 k8s file and paste into template. In the template deployment file change image tag as variable({{.Values.image.tag}} )
8. Now change values.yaml in helm: (see code git)
9. Now we can delete all service like deployment, service, ingress and can create everything by helm; the command is(enter into helm file): helm install go-web-app ./go-web-app-chart
10. Do ci with github actions: create a file github->workflows->ci.yaml
11. In yaml file configure CI: configure with documentation «github marketplace» (create dockerhub token from dockerhub:account setting->create token and put into github:enter repo->settings->Secrets and variables->new repo.., create a github personel token:settings->developper settings->classic token, and put into secrets and variables )

Summary of how ci/cd work: the stage of ci(build&unit test,static code analysis,docker images, update help). when developper push into github docker image will be created and update helm(values.yaml) automatically.

CD: argoCD will watch helm(values.yaml) and pull the image and install on k8s

Note: in service.yaml selector must be same name of deployment.yaml pod lavel(temmplate->labels->app)

Note : in deployment.yaml ressource lavel(metadata->labels->app) and template->metadata->labels->app must be same