

Nom :  
Prénom :  
Groupe :

Livret de TP CI/CD  
BUT 3  
Université d'Orléans -- 2023  
Y. Stroppa  
V0.9

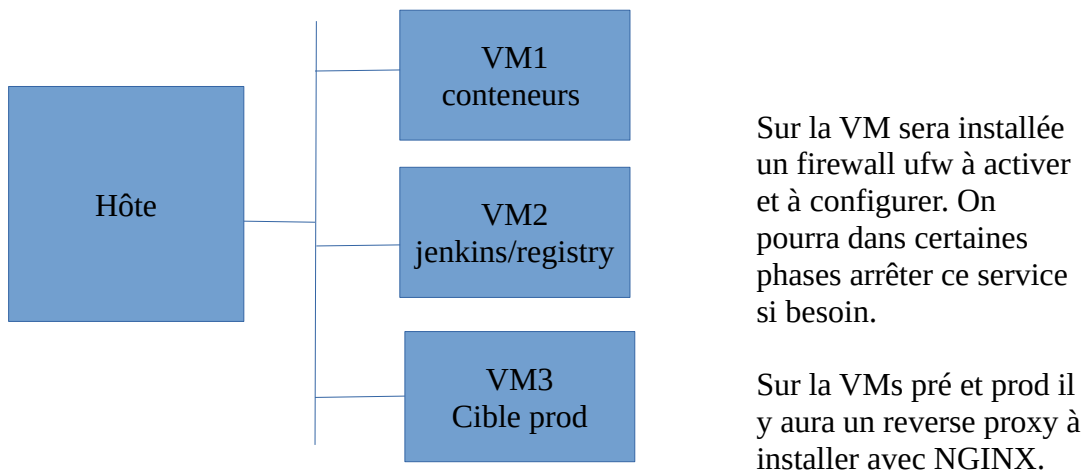
Ce livret est composé de différents exercices et questions auxquelles il va falloir répondre. Il est individuel et devra être mise à jour de façon régulière. les TPs sont des exercices qui se préparent pour être plus efficace lors de séances. Ce livret pourra à tout moment être évalué par les responsables de TPs et notés.

Objectif de ces TPs : maîtrise de la chaîne de développement et déploiement

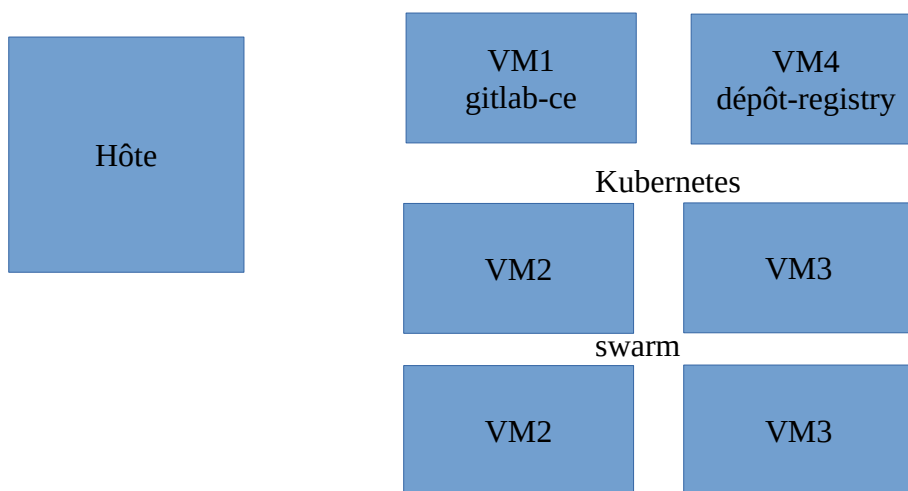
Tout au long de ces Tps nous allons mettre en place et utiliser différents services dans un contexte d'intégration continue et de déploiement continue ...

L'architecture mise en place sera composée de plusieurs VMS

Premier lot



Deuxième lot



## Déroulement des séances de TP

TP 1 : prise en compte de l'environnement IUT :

- Mise en place et installation de VMs fedora38 à partir d'une image ISO

- Prise en compte du système

  - SELinux

  - Firewall

  - Gestionnaire de packages

- Communication entre les différentes VMs

- Installations sur VM:

  - docker, podman et cri-o

TP 2 : manipulation des conteneurs selon les différents RunTimes

- Docker, docker-compose

- Podman

- cri-o

- Objectifs :

  - recherche d'images (compréhension des dépôts)

  - manipulation et paramétrage

  - production d'images simples et en multi-phases

  - production de solution applicatives simples (serveur web) et composites (wordpress)

TP 3 : mise en place et pratique sous SWARM

- notion de cluster d'orchestration

- notion de service, notion de stack

- haute disponibilité voir exemple : service stateless et service statefull

TP 4 : déploiement et configuration Ansible

- compréhension de ces systèmes de configuration et d'automatisation

- mise en place et configuration pour une installation de base

- mise en place et définition d'une configuration d'un cluster Kubernetes

TP 5 : Jenkins

- Installation de jenkins et paramétrage pour couplage avec un github

- Intégration continue d'un projet sous Python

- Intégration continue d'un projet sous Java

TP 6 : Intégration de l'ensemble avec jenkins et kubernetes

TP 7 : Supervision et monitoring sur le cluster Kubernetes

- Mise en place des outils de surveillance et de monitoring de votre système

# TP N°1 : Prise en compte de l'environnement (Rappel)

Objets : VM, Conteneur, network, firewall, etc...

Pré-requis :

VirtualBox/VMware/qemu.../cokpilot/

Image ISO de Ubuntu 22.04 LTS / Fedora38

Travail sous environnement virtualisé,

Installation

Paramétrage

Utilisation

Objectif de ce premier TP est d'installer et de revoir les principes de base de la mise en place de machine virtuelle être capable de monter un environnement multi-VMs dans un même lan.

De maîtriser les installations, la configuration des VMs et la communication entre elles. D'avoir assez d'aisance pour manipuler dans les différents environnements en mode ssh et ou graphique.

## Questions :

Indiquez les différences entre NAT, NATNETWORK, BRIDGE, HOST ONLY dans un contexte virtuaBox.

Qui attribue les adresses IP aux différentes VMs : indiquez le service qui s'occupe de ça

## Actions :

Act : on vous demande d'installer dans cet environnement uns VM Ubuntu SRV 22.04 et une VM fedora38. On évitera d'installer un environnement graphique pour des raisons de coûts. Les deux Vms sont sur le même LAN et peuvent dialoguer ensemble et sortir vers l'extérieur.

Act : Faites en sorte que l'on puisse se connecter sur les deux VMs à partir de l'hôte dans un terminal en mode SSH pour effectuer différentes opérations d'installation.

Act : analyse des deux VMs :

Nommer les deux VMs :

ub2204\_stv1

fed38\_srv1

Faites en sorte qu'elles se connaissent

Faites l'inventaire sur les caractéristiques de ces VMs

Noyau linux

Système de fichiers

Modification interface réseau

Quel est le système de protection de ce système ? garde barrières ou autre

Act : faire en sorte de pouvoir se connecter d'une VM à l'autre sans mot de passe à fournir en mode ssh.

Indiquez la procédure à mettre en place pour ça.

Installations à effectuer : docker, docker-compose, podman et cri-o ....

Act : indiquez à quoi sert les commandes suivantes :

lsblk

lsns

pstree

lsmod

systemd-analyse blame ou plot

brctl show

ip a

Act : Vérification de vos installations

Veillez démarrer plusieurs conteneurs à partir de l'image httpd avec docker, podman et crictl ....

## TP N°2 : Prise en compte de l'environnement (Rappel)

Travail sur les conteneurs : docker, podman et crio

A partir d'une VM, veuillez générer au moins deux comptes : user1 et user2 avec les autorisations nécessaires pour utiliser ces services. Vous switcherez d'un compte à l'autre en fonction des actions à effectuer.

Sous user1:veuillez démarrer plusieurs fois avec docker un conteneur à partir de l'image httpd. On se connectera sous user2 pour vérifier l'isolation entre les différents utilisateurs d'un même serveur.

Une fois installer docker, indiquez quel est le service qui doit être démarré et quels sont les pré-requis aux utilisateurs (user1, user2) pour manipuler des conteneurs.

Questions :

Le fait de démarrer plusieurs conteneurs httpd nécessite t-il plusieurs images.

Veuillez modifier le fichier par défaut du site web de ces différents conteneurs, explique comment docker gère cette situation ? Site A / Site B / Site C

Après ces différentes modifications, veuillez arrêter et supprimer les conteneurs.  
Indiquez la commande pour effectuer de type d'opérations ?

On souhaite pour conserver une persistance et accéder au site directement à partir de la machine Hôte du conteneur, indiquez comment on peut définir et attacher un volume et ainsi conserver les données.

## Notion de multi-conteneurs

Objectif : on souhaite travailler sur plusieurs conteneurs et garantir la communication entre ces différents éléments sur un exemple de type wordpress (CMS). En effet, Wordpress est composé de deux parties : frontal (service web) et persistance (Mysql ou Mariadb)

Veillez indiquer et effectuer le montage de ce sous système sur une les 2 VMs selon deux mode :

Mode manuelle avec les instructions de base de docker

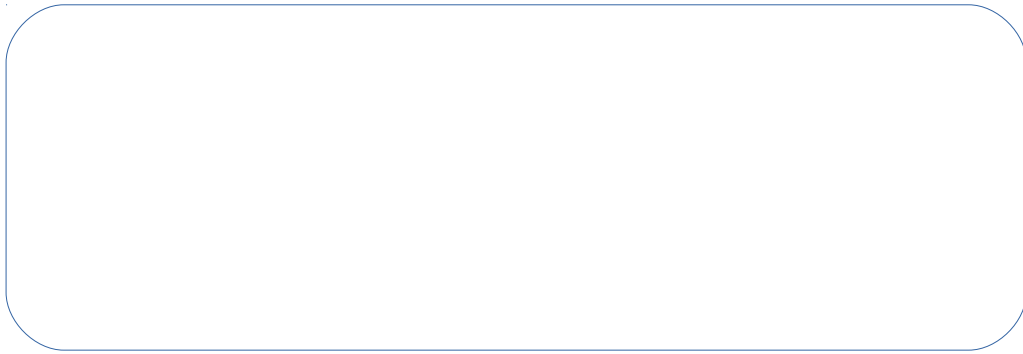
Mode avec docker-compose

Indiquez les différentes opérations à effectuer et les instructions à saisir pour ces deux modes d'installation.

Docker

docker-compose.yml

Indiquez ce que réalise par défaut docker-compose, et comment les conteneurs se connaissent ?



Fabrication d'image en multi-stages, vous avez un code java à déployer pour se faire veuillez indiquer les différents étapes pour constituer le jar et produire une image docker.

Faire la même chose avec Podman :

Veuillez à partir du dépôt quay.io installer une version de httpd.

Définir à partir d'un pod, une application de type wordpress, comment se passe la communication entre la partie apache et la partie base de données ?

Expliquez les différences si il y en a entre les deux approches ...

Voir l'utilisation de crictrl

Création d'une image à partir d'un code Java et Python .... (voir les notions multi-stages)



**TP N°3 : mise en place de l'orchestrateur SWARM :**

**TP N°4 : installation de Jenkins**

**TP N°5 : Ansible**

**TP N°6 : préparation de l'infrastructure Kubernetes 1/2**

**TP N°7 : Mise en place d'un monitoring sous Kubernetes**

