

TP 3 : Vagrant

Objectif : Le but de ce TP est que vous arriviez à créer votre premier environnement de développement avec Vagrant avec un interfaçage Ansible.

EXERCICE 1 : PROVISIONNING VIA LE SHELL

Partie 1 : provisionnement avec Shell

1. Déployer avec Vagrant une VM basée sur linux (ubuntu/focal64).
2. Modifiez Vagrantfile afin de lancer un script SHELL que vous aurez au préalable créé et qui au démarrage de la box va installer les packages `vim`, `nginx`, `git`, `console-data`. Ramenez la mémoire de la VM à 2048 Mo. Faites apparaître la console VirtualBox. La VM doit recevoir une adresse IP en DHCP.

Pour faire du provisionnement avec Shell vous allez utiliser la syntaxe suivante dans le Vagrantfile :

```
config.vm.provision "shell", inline: <<-SHELL
#vos commandes
SHELL
```

3. Vous allez rajouter un dossier synchronisé entre l'hôte et la VM, pour cela vous allez rajouter la ligne : `config.vm.synced_folder ".", "/var/www", create: true, group: "www-data", owner: "www-data"`
 - Le premier argument `"."` est le répertoire sur la machine hôte (dans ce cas, le répertoire courant).
 - Le second argument `"/var/www"` est le chemin où le dossier sera monté dans la machine virtuelle.
 - Les options supplémentaires comme `create`, `group`, et `owner` sont fournies dans un hash Ruby :
 - o `create: true` : Si le dossier cible n'existe pas, il sera créé.
 - o `group: "www-data"` et `owner: "www-data"` : Le groupe et le propriétaire du dossier seront définis sur `"www-data"`.
4. Faites un mappage entre les ports de la MH et la VM avec `guest: 80` et `host: 8081`
5. Lancer votre VM et vérifier bien le mappage des ports est effectué (`vagrant port`). Si la VM est déjà en cours d'exécution, vous pouvez lancer le provisionnement à nouveau avec la commande suivante : `vagrant provision`. Vous pouvez également redémarrer simplement votre VM : `vagrant reload`
6. Vérifier bien que vous pouvez accéder à la page par défaut de nginx depuis votre navigateur sur la machine hôte. Quelle adresse IP vous avez utilisé ?
7. Quelle commande vous permet de savoir l'adresse IP de la VM sans y accéder ?
8. Quel dossier s'est rajouté dans le dossier de partage ?
9. Personnaliser le fichier `index.html` directement depuis votre MH et vérifier que la modification est bien prise en compte sur la VM

Partie 2 : Gestion des Snapshots


À présent vous allez voir comment utiliser les snapshots avec Vagrant pour capturer l'état d'une VM et restaurer cet état ultérieurement. Les snapshots sont utiles pour tester des scénarios, effectuer des mises à jour en toute sécurité ou revenir à un état stable.

Pour gérer les snapshots avec Vagrant, vous pouvez utiliser les commandes suivantes :

- Pour prendre un instantané de la machine virtuelle :
`vagrant snapshot save [nom_de_l_instantané]`
 - Pour lister les instantanés disponibles :
`vagrant snapshot list`
 - Pour revenir à un instantané spécifique :
`vagrant snapshot restore [nom_de_l_instantané]`
 - Pour supprimer un instantané :
`vagrant snapshot delete [nom_de_l_instantané]`
 - Pour supprimer tous les instantanés :
`vagrant snapshot delete --all`
1. Prenez un snapshot de l'état actuel de votre VM et vérifiez sa présence dans la liste des snapshots.
 2. Accédez à la VM et installez le service DNS BIND puis vérifiez que le service fonctionne.
 3. Prenez un second snapshot
 4. Revenez à l'état initial (avant l'installation de BIND) et vérifiez que BIND n'est plus installé.
 5. Revenez à l'état avec BIND installé et vérifiez à nouveau que le service est actif.
 6. *Supprimer l'ensemble des snapshot réalisés.*

EXERCICE 2 : MULTI-VM AVEC PROVISIONNEMENT AUTOMATIQUE

Dans cet exercice vous devez créer un environnement de développement avec Vagrant, composé de deux VM. Une qui jouera le rôle de serveur web (avec Apache) et l'autre celui de serveur de base de données (avec MySQL). Le provisioning sera effectué via des scripts shell pour automatiser l'installation et la configuration des services sur chaque machine.

1. Créez un nouveau répertoire pour votre projet et initialisez un nouveau fichier Vagrantfile.
 2. Modifiez le fichier Vagrantfile pour configurer les deux VM :
 - a. Utilisez Ubuntu comme système de base pour les deux machines.
 - b. Assurez-vous que les deux machines peuvent communiquer via un réseau privé.
 - c. Le nom de la VM serveur web c'est 'web-server' et le nom de la VM serveur BD c'est 'db-server'
 - d. Les deux VM sont provisionnées automatiquement en shell par deux scripts bash : `install_apache.sh` et `install_mysql.sh`
-  Pour faire du provisioning avec Shell vous allez utiliser la syntaxe suivante dans le Vagrantfile : `config.vm.provision "shell", path: "script.sh"`
3. Créez un répertoire scripts dans votre projet où vous allez mettre les scripts de provisionnement.
 - a. Le script `install_apache.sh` pour installer Apache sur la machine "web".
 - b. Le script `install_mysql.sh` pour installer MySQL sur la machine "db".
 4. Démarrer les deux VM afin d'effectuer le provisioning automatique.

5. Accédez aux deux VM (en ssh) et vérifiez que les deux services sont installés et sont en cours d'exécution
6. Faites un test de connectivité entre vos deux VM.

EXERCICE 3 : PROVISIONNEMENT AVEC ANSIBLE

Dans cet exercice vous allez mettre en place un serveur DHCP sur une VM en utilisant Vagrant pour créer l'environnement et Ansible pour automatiser l'installation et la configuration du service DHCP.

1. Créez un nouveau répertoire pour votre projet et initialisez un fichier Vagrantfile
2. Modifiez le fichier Vagrantfile pour configurer les deux VM (serveur et client DHCP) :
 - a. Utilisez la box "ubuntu/focal64"
 - b. Assurez-vous que les deux machines peuvent communiquer via un réseau privé.
 - c. Le nom de la VM est "dhcp-server" pour le serveur et "dhcp-client"
 - d. La VM "dhcp-server" sera provisionnée automatiquement par Ansible avec le playbook vm_dhcp.yml
3. Créez le playbook vm_dhcp.yml dans le même répertoire. Ce fichier contient les instructions Ansible pour installer et configurer un serveur DHCP. Les tâches à faire sont :
 - a. Mise à jour des packages
 - b. Installer le serveur DHCP
 - c. Configurer le fichier dhcpd.conf à partir du fichier fourni sur moodle (ansi_dhcpd.conf)
 - d. Redémarrer le service DHCP
4. Démarrer la VM serveur afin d'effectuer le provisioning automatique.
5. Accédez à la VM (en ssh) et vérifiez que le service est bien installé et il est en cours d'exécution.
6. Démarrer la VM 'dhcp-client' et vérifiez qu'elle obtient bien une adresse IP depuis votre serveur DHCP