Lab 6- Multi-Machine

Vagrant est capable de définir et de contrôler plusieurs machines invitées par Vagrantfile. C'est ce qu'on appelle un environnement «**multi-machines**».

Ces machines sont généralement capables de fonctionner ensemble ou sont en quelque sorte associées les unes aux autres. Voici quelques cas d'utilisation que les gens utilisent aujourd'hui dans des environnements multi-machines:

- Modélisation précise d'une topologie de production multi-serveurs, telle que la séparation d'un serveur Web et d'un serveur de base de données.
- Modéliser un système distribué et comment ils interagissent les uns avec les autres.
- Tester une interface, telle qu'une API vers un composant de service.

Définition de plusieurs machines

Plusieurs machines sont définies dans le même projet <u>Vagrantfile</u> à l'aide d'appel de méthode **config.vm.define**.

```
Vagrant.configure("2") do |config|
    config.vm.provision "shell", inline: "echo Hello"

config.vm.define "web" do |web|
    web.vm.box = "apache"
    end

config.vm.define "db" do |db|
    db.vm.box = "mysql"
    end

end
```

L'objet de configuration est chargé et fusionné avant la configuration spécifique à la machine, tout comme les autres Vagrantfiles dans l'ordre de chargement de Vagrantfile. Vagrant applique l'ordre extérieur-intérieur, dans l'ordre indiqué dans le Vagrantfile.

Exemple pour l'approvisionnement :

```
Vagrant.configure("2") do |config|
  config.vm.provision "shell", inline: "echo Hello"
  config.vm.define "web" do |web|
    web.vm.box = "Centos/7"
         $script =<<-'SCRIPT'</pre>
            sudo yum -y update
            sudo yum -y install httpd
            SCRIPT
     web.vm.provision "shell", inline: $script
  end
  config.vm.define "db" do |db|
    db.vm.box = "ubuntu/focal64"
         $script =<<-'SCRIPT'</pre>
            sudo apt update -y
            sudo apt install -y mysql-server
            SCRIPT
    db.vm.provision :shell, inline: $script
  end
  config.vm.provision :shell, inline: "echo Machines prêtes"
end
```

Exercice 1:

Créer un fichier Vagrantfile pour préparer un lab avec la configuration suivante :

- 1. Machine serveur:
 - Box : Centos/7
 - Ip: 10.0.0.10
 - Mappage du port 8080 :80
 - Ram: 2048
 - CPU:2
- 2. Node 1:
 - Box: Centos/7
 - Ip: 10.0.0.20
 - Mappage du port 8091 :80
 - Ram: 1024

3. Node 2:

Box : Centos/7Ip : 10.0.0.21

• Mappage du port 8092 :80

• Ram: 1024

Multi-Machine - Les boucles

Dans le cas où les machines que vous allez créer partagent les mêmes caractéristiques vous pouvez utiliser les boucles pour créer une configuration Multi-Machines.

Voici un exemple

```
(1..5).each do |i|
    config.vm.define " ubuntu0#{i}" do |subconfig|
        subconfig.vm.box = "ubuntu/xenial64"
        subconfig.vm.hostname = "ubuntu0#{i}"
        end
end
```

Exercice 2:

Adaptez la configuration créée dans l'exercice 1 pour préparer un lab composé d'un serveur et 3 nœuds en utilisant les boucles.

Reprenez la même configuration de différentes Box de l'exercice 1.