Installation Ansible

Introduction

Ansible va permettre l'automatisation de taches via des playbook écrit en YAML

Ansible est une plateforme open-source d'automatisation informatique qui simplifie la gestion et la configuration des infrastructures. Contrairement à d'autres outils, Ansible n'exige pas l'installation d'un agent sur les machines cibles, offrant ainsi une simplicité et une flexibilité accrues.

Avec Ansible, vous pouvez décrire l'état désiré de votre infrastructure dans des fichiers appelés playbooks. Ces playbooks décrivent les tâches à effectuer, les machines cibles et les conditions pour leur exécution. Grâce à sa syntaxe lisible et à sa modularité, Ansible permet d'automatiser diverses tâches telles que le déploiement d'applications, la configuration des serveurs et la gestion des services.

Installation

Il faut d'abord installer le paquet « ansible »

Il faut au d'abord les créer et les remplir sois-même

```
root@testVPN:/etc/ansible# touch ansible.cfg hosts
root@testVPN:/etc/ansible#
```

Le fichier de configuration d'Ansible est celui-ci « /etc/ansible/ansible.cfg »

```
  GNU nano 5.4                                  ansible.cfg
[defaults]
inventory = /etc/ansible/hosts
remote_user = root
#private_key_file =
become = true
become_user = root
remote_port = 22
```

Dans Ansible, l'attribut become: true est utilisé pour indiquer que l'utilisateur exécutant les tâches doit passer en tant qu'utilisateur avec des privilèges élevés (tel que root) sur l'hôte cible. Cela permet d'exécuter des tâches qui nécessitent des droits d'administration.

Lorsque vous définissez become: true dans un playbook Ansible, les tâches qui suivent seront exécutées avec les privilèges de l'utilisateur spécifié (par défaut, root). Cela peut être utile pour des actions telles que l'installation de logiciels, la modification de fichiers système ou la gestion des services.

Pour la gestion des hôtes là où l'on renseignera les IP + mdp ssh ou clef ssh « /etc/ansible/hosts »

```
  GNU nano 5.4                                      hosts *
[web]
192.168.1.118 ansible_ssh_pass=
```

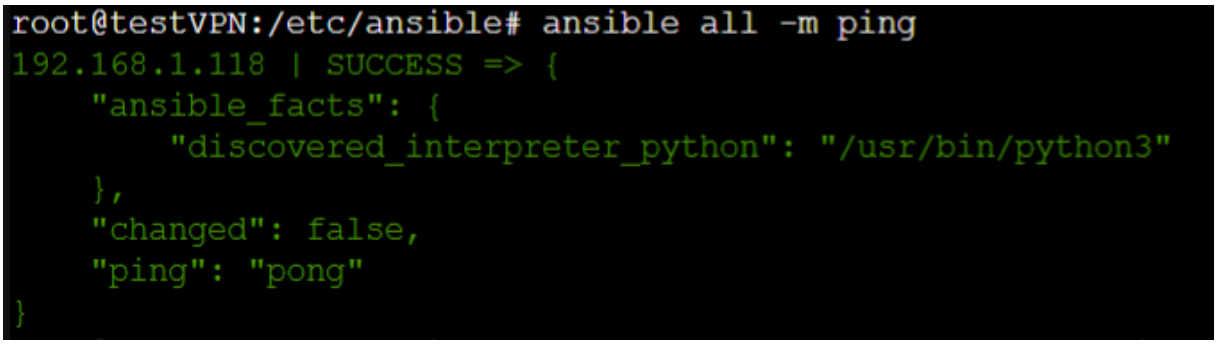
Pour que notre fichier de configuration sois pris en compte et utiliser par défaut il faut modifier la variable d'environnement destiner à cet effet

```
root@testVPN:/etc/ansible# export ANSIBLE_CONFIG=/etc/ansible/ansible.cfg
```

Test connection

Pour utiliser une authentification via mot de passe il faut installer le paquet « sshpass ».

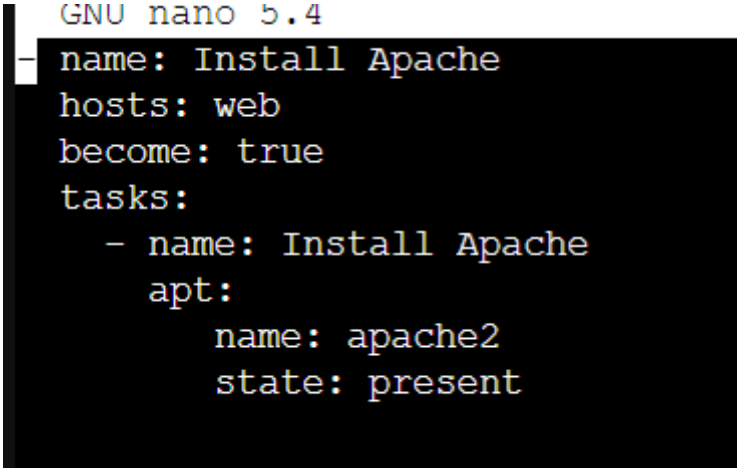
Apparemment il faut enregistrer la fingerprint de l'hote (clef publique) pour que Ansible puisse ce connecter il n'accepte pas automatiquement.



Le test de connection fonctionne correctement

Installation d’apache2

Je crée ce qu’on appelle un playbook



Le playbook est une structure de haut niveau utilisée par Ansible pour décrire une série d'actions à exécuter sur un ensemble d'hôtes distants. Il permet d'automatiser la configuration, le déploiement et la gestion des infrastructures.

Le playbook que vous avez fourni est un exemple simple qui installe le serveur Apache sur les hôtes du groupe "web". Voici une explication détaillée de chaque élément du playbook :

name: Install Apache: Cette ligne définit le nom du playbook, qui est utilisé pour l'identifier lors de l'exécution.

hosts: web: Cette ligne spécifie les hôtes cibles sur lesquels les tâches du playbook seront exécutées. Dans cet exemple, les tâches seront exécutées sur les hôtes faisant partie du groupe "web".

become: true: Cette ligne indique à Ansible d'exécuter les tâches avec les privilèges de superutilisateur (root) sur les hôtes distants. Cela permet d'effectuer des opérations qui nécessitent des droits d'administration.

tasks: Cette ligne marque le début de la section des tâches du playbook.

- name: Install Apache: Cette ligne définit le nom de la tâche. Il s'agit d'une description textuelle de la tâche à effectuer.

apt:: Cette ligne spécifie le module à utiliser pour cette tâche. Dans cet exemple, le module "apt" est utilisé pour gérer les paquets sur les systèmes Debian/Ubuntu.

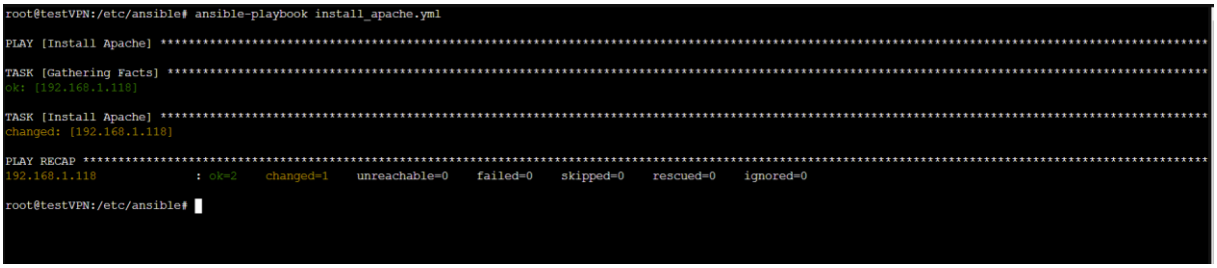
name: apache2: Cette ligne définit le nom du paquet à installer, qui est "apache2" dans ce cas.

state: present: Cette ligne indique que le paquet doit être présent sur le système. Si le paquet est déjà installé, Ansible ne fera rien. Sinon, il sera installé.

En résumé, ce playbook installe le serveur Apache (paquet "apache2") sur les hôtes du groupe "web" en utilisant les privilèges de superutilisateur. Vous pouvez exécuter ce playbook en utilisant la commande ansible-playbook avec le fichier YAML contenant ce playbook comme argument.

J’exécute le playbook comme ceci

```
root@testVPN:/etc/ansible# ansible-playbook install_apache.yml
```



PLAY RECAP: Cette partie récapitule les résultats de l'exécution du playbook pour chaque hôte. Dans votre cas, l'hôte 192.168.1.118 a donné le résultat suivant : 2 tâches exécutées (ok=2), 1 tâche a apporté des modifications (changed=1), et il n'y a pas eu d'erreurs (unreachable=0, failed=0, skipped=0, rescued=0, ignored=0).

Cela indique que le playbook a réussi à installer Apache sur l'hôte distant (192.168.1.118). Vous pouvez vérifier sur cet hôte si Apache est bien installé et fonctionne correctement.