INSTALLATION ET CONFIGURATION DE ANSIBLE

RAPPORT DE PROJET D'AUTOMATISATION DES TACHES

Auteurs :

Adama Coly
Seydou Hanne
Mamadou Baba Ly
Ousseynou Ngom
Mariama Tambedou

GROUPE 1 PROMO 2 AWS DE LA SONATEL ACADEMY SENEGAL

INTRODUCTION

Ansible est un outil Open Source d'automatisation informatique qui automatise le provisionnement, la gestion des configurations, le déploiement des applications, l'orchestration et bien d'autres processus informatiques manuels. À la différence des outils de gestion plus simples, avec Ansible les utilisateurs (administrateurs système, développeurs, architectes) peuvent recourir aux fonctions d'automatisation pour installer des logiciels, automatiser des tâches quotidiennes, provisionner une infrastructure, améliorer le niveau de sécurité et de conformité, appliquer des correctifs système et partager leurs processus automatisés avec toute l'entreprise.

Dans ce TP, on faire ensemble :

1. L'installation et la configuration de ansible
2. Editer le fichier d'inventaire(host)
3. Connecter le serveur central d'ansible et les serveurs nodes (Le serveur central est le serveur ou est installé ansible alors que les serveurs nodes sont ceux qui sont connectes au serveur central)
4. Vérifier la connectivité entre le serveur central et les serveurs nodes(SN)
5. Exécuter des commandes ad-hoc à partir du serveur central (SC)
6. Et enfin écrire un playbook qui permet d'installer git et apache serveur.

Durant tout le tutoriel SC fait référence au serveur central avec ansible et SN aux serveurs node ; c'est-à-dire les serveurs contrôlés par le SC.

Attacher vos ceintures, c'est parti 😊 …

Step 1 — Pre-requis de Ansible:

Avant tout, installer deux machine virtuelles linux : le serveur central (ou node manager avec ansible) et le serveur node (l'hôte contrôlé par le SC).

Au niveau du node manager :

- Créer un nouveau compte utilisateur avec des privilèges sudo pour de questions de securite ---> # adduser nom_user
- L'ajouter au sudo group --> # usermod -aG sudo nom_user
- Change user avec la commande → # su nom_user)
- Installer ssh avec → apt install ssh

- Générer une paire de clé associée a cette utilisateur --> $ ssh-keygen : une paire de 2048 bit par défaut est générée et choisissez une Passphrase.
- Copier la clé publique du SC au SN avec--> ssh-copy-id username_du_SN@adress_Ip_SN.

Au niveau du SN :

- Installer ssh avec : apt install ssh

NB : Le SC et les SN doivent se trouver dans le même réseau.

Step 2 — Installation de Ansible:

- sudo apt-add-repository ppa:ansible/username_de_ansible
- sudo apt update : pour actualiser le système de package
- sudo apt install ansible : installer ansible
- ansible --version : pour vérifier que ansible est bien installe

Step 3 — Configurer le fichier Inventaire

Maintenant que nous avons bien installe ansible, nous allons déclarer nos serveurs nodes a contrôlé. Le fichier inventaire contient toutes les informations des serveurs nodes que nous allons gérer.
Pour éditer le fichier inventaire, nous allons nous rendre dans : /etc/ansible/hosts (sudo nano /etc/ansible/hosts). Et ajouter ces lignes :

```
[servers]

Serveur1 ansible_host=@IP_du_SN1

[all:vars]
ansible_python_interpreter=/usr/bin/python3
```

Pour vérifier notre fichier inventaire qui détient les infos des serveurs gérés par ansible, taper ansible-inventory --list -y

Step 4 — Tester la connexion

Apres la configuration du fichier hosts, il est maintenant temps de vérifier si le serveur central arrive à communiquer avec le serveur node via ssh --> ansible serveur1 -m ping -u nom_user_SN.

Si vous avez un ping pong, la connexion est établie entre le SC et les SNs

Explications des paramètres :

- -m --> comme module, ici on appelle le module ping
- -u --> user
- serveur1 --> c'est le SN ; mais on pouvait mettre all pour prendre en compte tous les hosts dans le fichier inventaire (hosts).

On peut aussi spécifier un groupe de serveur pour lancer un commande : #ansible serveur_1:serveur_2:serveur_n -m ping -u ngom pour ne cibler que les serveurs 1, 2, jusqu'à n.

Step 5 — Execution des Ad-Hoc Commands (Optional)

Les commandes ad-hoc d'Ansible permettent d'exécuter une tache précise et particulière comme vérifier la configuration réseau ou installer une application.

- ansible serveur1 -a "ifconfig" -u username_SN
- ansible serveur1 -a "df -h" -u username_SN
- ansible serveur1 -m apt -a "name=vim state=latest" -u username_SN

Step 6 – Installation de apache et git et avec ansible

Un playbook Ansible est un modèle de tâches d'automatisation, qui sont des opérations informatiques complexes exécutées sans intervention humaine. Les playbooks Ansible sont écrits au format YAML lisible par l'homme et exécutés sur un ensemble, un groupe ou une classification d'hôtes, qui forment ensemble un inventaire.
Y'aura probablement certains qui vont dire ***« mais on peut exécuter des commandes a distances avec ssh »***. Hum ☹ d'accord mais imaginez que vous voulez configurer 50 serveurs en même temps, allez-vous utiliser ssh ??? Ohhhhh Nonnn toi aussiiiii. Fait appel a la magie mon ami, utilise un playbook ansible pour faire le travail et bois ton café. Procédons comme ci-dessous pour notre cas.

1. Créer un fichier nomme **myplaybook.yml** et placer y ces lignes de codes suivant :

```
---
- name: Mon premier playbook
  hosts: serveur1
  become: yes
  vars:
    message1: cache update avec success
    message2: Apache installe avec success
    message3: Git avec success
  tasks:
    - name: Update le cache
      apt: update_cahce=yes
    - debug: msg={{ message1 }}
    - name: Installation de apache server
      apt: name=apache state=latest
    - debug: msg={{ message2 }}
    - name: Installation de git
      apt:
        name: git
        state: latest
    - debug: msg={{ message3 }}
  handlers:
    - name: Redemarrage de apache
      service:
        name: apache2
        state: restarted
```

2. Enregistrer le fichier
3. Exécuter le playbook avec la commande : ansible-playbook myplaybook.yml

Bravooooo 😋 vous venez d’installer apache et git dans tous vos serveurs grâce à un playbook. Maintenant avez-vous toujours l’idée d’utiliser ssh ?? 😋

N’hésitez pas à visiter la documentation de ansible pour avoir un niveau pousse sur l’écriture des playbook ansible afin de pouvoir automatiser tous vos taches répétitives.

Merci à la prochaine 😋 😉 😉 😉