

TD1 : Introduction à l'Infrastructure as Code (IaC)

Objectifs

- Comprendre les limites des approches manuelles
- Découvrir les principes de l'Infrastructure as Code
- Manipuler Terraform et Ansible sur un exemple simple

1. Préparation de l'environnement

1.1 Prérequis

- Linux / macOS / WSL recommandé
- Docker installé
- Accès terminal

1.2 Installation de Terraform

Vérification des packages :

```
sudo apt-get update && sudo apt-get install -y gnupg software-properties-common
```

Installer la clé HashiCorp's GPG

```
wget -O- https://apt.releases.hashicorp.com/gpg | \  
gpg --dearmor | \  
sudo tee /usr/share/keyrings/hashicorp-archive-keyring.gpg > /dev/null
```

Ajouter le dépôt HashiCorp :

```
echo "deb [arch=$(dpkg --print-architecture) signed-  
by=/usr/share/keyrings/hashicorp-archive-keyring.gpg]  
https://apt.releases.hashicorp.com $(grep -oP '(?<=UBUNTU_CODENAME=).*' /etc/os-  
release || lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
```

Installer :

```
sudo apt update  
sudo apt-get install terraform
```

Vérification :

```
terraform version
```

1.3 Installation d'Ansible

Sur Ubuntu / Debian :

```
sudo apt update
sudo apt install ansible -y
```

Vérification :

```
ansible --version
```

1.4 Vérification Docker

```
docker --version
docker run hello-world
```

2. Arborescence du projet

Créer un dossier de travail :

```
mkdir iac-demo
cd iac-demo
```

Structure :

```
iac-demo/
├── terraform/
│   └── main.tf
├── ansible/
│   ├── inventory.ini
│   └── playbook.yml
```

3. Exemple Terraform

3.1 Objectif

Créer un conteneur NGINX avec Terraform

3.2 Fichier main.tf

Chemin :

terraform/main.tf

Contenu :

```
terraform {
  required_providers {
    docker = {
      source = "kreuzwerker/docker"
    }
  }
}

provider "docker" {}

resource "docker_container" "nginx" {
  image = "nginx:latest"
  name  = "mon_nginx"

  ports {
    internal = 80
    external = 8090
  }
}
```

3.3 Exécution

Depuis le dossier terraform :

```
cd terraform
terraform init
terraform apply
```



Validation :

- Ouvrir <http://localhost:8090>

Vérification du state créé

```
terraform state list
```

4. Exemple Ansible

4.1 Objectif

Configurer un système local avec Ansible

4.2 Fichier inventory

Chemin :

ansible/inventory.ini

Contenu :

```
localhost ansible_connection=local
```

4.3 Fichier playbook.yml

Chemin :

ansible/playbook.yml

Contenu :

```
- hosts: localhost
  connection: local

  tasks:
    - name: créer une page web temporaire
      copy:
        dest: /tmp/index.html
        content: |
          <html>
          <head>
            <meta charset="UTF-8">
            <title>Déploiement avec Ansible</title>
          </head>
          <h1>Déploiement avec Ansible</h1>
          <p>IaC en action</p>
          </html>

    - name: copier la page dans le conteneur nginx
      command: docker cp /tmp/index.html mon_nginx:/usr/share/nginx/html/index.html
```

4.4 Exécution

Depuis le dossier ansible :

```
cd ../ansible
ansible-playbook -i inventory playbook.yml
```



Validation :

- Ouvrir <http://localhost:8090>

5. Lecture globale

Terraform :

- crée une ressource (conteneur)
- via API (Docker ici)

Ansible :

- configure un système
- exécute des tâches

Contrairement à Terraform, Ansible ne maintient pas un fichier d'état centralisé.

Cependant, Ansible est conçu pour être idempotent : il vérifie l'état du système avant d'appliquer des changements, et n'exécute une action que si nécessaire.



Terraform

- sait ce qu'il a créé → grâce au state
- compare : état actuel vs état désiré

Ansible

- ne stocke rien
- regarde directement le système cible
- agit seulement si besoin (si modules bien utilisés)

6. Nettoyage

Terraform :

```
cd ../terraform
terraform destroy
```

7. Points importants

- Terraform est déclaratif

- Ansible est procédural
- Les deux outils sont complémentaires

8. Question

Pourquoi séparer la création d'infrastructure et la configuration logicielle ?

From:

<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:

<http://slamwiki2.kobject.net/eadl/bloc4/fm2/td1?rev=1777110079>

Last update: **2026/04/25 11:41**

