TD2 suite - Plateforme web avec routage

Objectif

Déployer une plateforme composée de plusieurs services accessibles via un point d'entrée unique.

Vous devez concevoir l'infrastructure avec Terraform et configurer les services avec Ansible.

1. Résultat attendu

À la fin du TP, les URLs suivantes doivent fonctionner :

- http://localhost:8080/app1 → affiche "app1"
- http://localhost:8080/app2 → affiche "app2"

Une seule entrée (port 8080) Plusieurs services derrière

2. Architecture imposée

Vous devez déployer :

- 1 conteneur reverse proxy (nginx)
- N conteneurs applicatifs

Contraintes :

- les conteneurs applicatifs NE DOIVENT PAS exposer de port vers l'extérieur
- seul le proxy expose un port

3. Comportement attendu

Le reverse proxy doit :

- rediriger /app1 vers le service app1
- rediriger /app2 vers le service app2

Chaque service doit :

- retourner une page HTML simple avec son nom

4. Terraform

Vous devez :

- créer un module pour les services applicatifs
- pouvoir définir plusieurs services via une map

Exemple attendu (indicatif) :

```
services = {
  app1 = { port = 5001 }
  app2 = { port = 5002 }
}
```

Contraintes :

- utilisation de `for_each`
- aucun copier/coller de ressource

5. Ansible

Vous devez :

- configurer le reverse proxy
- générer dynamiquement sa configuration

Le proxy doit connaître :

- le nom des services
- leur port interne

Indice :

- template Jinja2 recommandé

6. Vérification

Vous devez pouvoir :

- ajouter un service (ex: app3)
- relancer Terraform + Ansible
- accéder à /app3 sans modifier la logique existante

7. Contraintes

- séparation Terraform / Ansible
- code factorisé
- aucun hardcoding des services dans le proxy

8. Questions



- Comment Terraform transmet-il les informations à Ansible ?
- Où est définie la liste des services ?
- Votre solution fonctionne-t-elle avec 5 services ?

From:
http://slamwiki2.kobject.net/ - **SlamWiki 2.1**

Permanent link:
http://slamwiki2.kobject.net/eadl/bloc4/fm2/td2-suite

Last update: **2026/05/12 21:47**