# TP – Hardening et Automatisation d'une Infrastructure Hybride Critique - 2h30 Heures

**Contexte :** Projet "SecureCloud"

L'entreprise **SecureCloud** entame la migration d'une application critique vers un modèle Replatform (Lift and Reshape). En tant qu'expert Cloud, vous recevez un premier jet de code Terraform (fourni ci-après) permettant de provisionner une instance IaaS et une base de données managée.

Votre mission est de transformer cette infrastructure "passoire" en une forteresse numérique avant tout déploiement en production.

## Table des matières



Julien Lallemand Février 2026

# Phase 1 : Provisioning "Insecure by Design" (40 min)

Le code qui vous a été transmis contient **7 failles de sécurité majeures**. Vous devez identifier et corriger les points suivants dans votre fichier .tf :

- **Gestion des Identités** : Ne laissez aucun secret ou clé API en dur dans le code.
- **Filtrage Réseau** : Configurez des règles de pare-feu **Stateful**. Supprimez les accès SSH (22) et RDP (3389) ouverts sur 0.0.0.0/0.
- **Cycle de Vie** : Remplacez l'AMI obsolète par une version maintenue et sécurisée.
- **Chiffrement** : Assurez-vous que toutes les données (disques EBS et RDS) sont chiffrées au repos.

## Alignement avec le Modèle de Tiering AD

Vous devez structurer votre réseau pour respecter le cloisonnement du **Tiering Model de l'Active Directory**:

- La base de données doit être isolée et ne posséder **aucune adresse IP publique**.
- La VM applicative (**Tier 1**) doit être la seule capable de communiquer avec la base de données via un flux sécurisé.
- Le **Tier 0** (serveurs les plus sensibles) doit être simulé par un sous-réseau totalement étanche.

## Responsabilité Partagée

Pour chaque correction effectuée, vous devrez être capable de justifier si l'action relève de votre responsabilité (client) ou de celle du fournisseur Cloud (AWS/Azure/GCP) selon le modèle **IaaS** ou **PaaS** utilisé.

## Livrables attendus (Phase 1)

1. **Code Terraform corrigé** : Un fichier .tf fonctionnel, propre et documenté.
2. **Preuve de validation** : Une capture d'écran de la commande terraform plan ne montrant aucun avertissement de sécurité évident.

**Attention :** L'usage de l'IA pour générer le code est détectable. Votre note sera basée sur la capacité de votre infrastructure à résister à un audit de conformité et sur la justesse de votre architecture réseau (Tiering).

# Phase 2 : Configuration & Hardening via Ansible (1h)

## Préparation de l'Environnement

- **Inventaire Dynamique** : Créez un fichier d'inventaire regroupant vos machines par groupes (ex: [web], [db]). Utilisez des variables pour définir l'utilisateur de connexion (ansible_user) et la clé SSH.

- **Connexion SSH** : Votre machine de contrôle doit "pousser" la configuration vers les cibles via SSH.

## Playbook de "Hardening" Système

Vous devez rédiger un Playbook YAML capable de réaliser les actions suivantes sur toutes les instances:

- **Mise à jour de sécurité** : Garantir que tous les paquets système sont à jour pour corriger les failles connues.
- **Sécurisation SSH** :
    - Désactiver l'authentification par mot de passe (force l'usage des clés).
    - Interdire la connexion directe du compte root.
- **Gestion des Secrets** : Utilisez **Ansible Vault** pour chiffrer les mots de passe de la base de données ou les clés sensibles. Aucun secret ne doit apparaître en clair dans vos fichiers.

## Pare-feu et Tiering Applicatif

Configurez un pare-feu **Stateful** (via le module ufw ou firewalld) pour appliquer le cloisonnement réseau:

- **Tier 1 (Web)** : Autoriser uniquement le trafic HTTP/HTTPS entrant et le flux SSH depuis votre IP de gestion.
- **Tier 0 (Simulé)** : Si une machine de gestion d'identité est présente, elle doit être isolée de tout flux non indispensable.
- **Flux Inter-niveaux** : La base de données ne doit accepter de connexions SQL *que* si elles proviennent de l'IP privée de la VM Web.

## Contraintes

### L'Idempotence

Le concept fondamental d'Ansible est l'idempotence : l'exécution multiple d'un script doit produire le même résultat sans erreurs ni modifications inutiles.

### Interdiction des Modules "Shell"

Pour valider votre expertise, l'usage des modules shell ou command est **interdit**. Vous devez utiliser les modules natifs (ex: apt, user, copy, lineinfile, ufw) qui garantissent la gestion de l'état.

## Livrables attendus (Phase 2)

1. **Le fichier Playbook (site.yml)** et les éventuels fichiers de variables chiffrés.
2. **Le fichier d'inventaire** structuré.
3. **Le rapport d'exécution** (log) prouvant l'idempotence.

# Phase 3 : Audit et Rapport de Sécurité (50min)

## Audit de Sécurité Post-Déploiement

Vous devez vérifier que les mesures de durcissement (hardening) sont effectives. Effectuez les tests suivants et consignez les résultats :

- **Test de Rejet SSH** : Tentez de vous connecter en SSH avec un mot de passe ou en tant que root. L'accès doit être strictement refusé.
- **Vérification de l'État (Stateful)** : Prouvez que votre pare-feu analyse le contexte des connexions (par exemple, en montrant que les paquets de retour d'une requête HTTP sortante autorisée sont acceptés sans règle entrante spécifique).
- **Audit des Secrets** : Confirmez qu'aucun secret n'est présent dans l'historique de vos commandes ou dans les fichiers de configuration non chiffrés.

## Matrice de Responsabilité Partagée

Pour votre infrastructure finale (mélange d'IaaS pour la VM et de PaaS pour la base de données), complétez une matrice de responsabilité précisant qui (le fournisseur ou SecureCloud) est responsable de:

- La sécurité physique du centre de données.
- Le déploiement des correctifs (patching) de l'OS de la VM.
- La protection des données stockées dans la base de données SQL.
- La configuration des politiques de sauvegarde (backup).

## Justification de la Stratégie de Migration

Rédigez un court paragraphe (10 lignes maximum) expliquant pourquoi l'approche choisie correspond bien à un scénario de **Replatform** plutôt qu'à un simple **Rehost**. Vous devez intégrer les notions suivantes :

- Réduction de la charge opérationnelle grâce aux services managés.
- Optimisation ciblée sans réécriture complète du code.

## Livrable Final : Le "Security Handover"

Vous devez remettre un document synthétique (PDF) incluant :

1. **Le schéma d'architecture final** annoté avec les niveaux de **Tiering AD** (Tier 0, Tier 1, Tier 2).
2. **La preuve de conformité** : Un tableau récapitulant les tests d'audit réussis.
3. **Une analyse FinOps rapide** : Identifiez un levier spécifique (ex: Savings Plans ou AHB) qui pourrait réduire le TCO de cette infrastructure à l'avenir.