# Rapport de Configuration : Bootstrap de l'Infrastructure Terraform

Ce document détaille la phase d'initialisation (bootstrap) de Terraform, correspondant à la section 2.2 du guide d'implémentation. L'objectif de cette étape était de résoudre le paradoxe de "la poule et de l'œuf" en créant une fondation pour la gestion de l'état de l'infrastructure en tant que code.

## 1. Objectif

L'objectif principal était de créer un **backend distant** pour Terraform. Un backend distant est un emplacement de stockage en ligne (ici, un bucket Google Cloud Storage) où Terraform sauvegarde le fichier qui décrit l'état de l'infrastructure qu'il gère. Cette approche est cruciale pour le travail en équipe et la sécurité, car elle évite de stocker ce fichier sensible sur un ordinateur local.

La procédure de bootstrap a consisté à utiliser Terraform en mode local une seule fois pour créer ce backend distant.

## 2. Prérequis et Configuration Initiale

* **Installation de Terraform :** L'outil en ligne de commande Terraform (version 1.13.0) a été installé sur Windows 11. Le processus a nécessité de placer l'exécutable terraform.exe dans un dossier dédié (C:\Terraform) et d'ajouter ce dossier aux variables d'environnement système (Path) pour le rendre accessible depuis le terminal.
* **Création de la Structure de Dossiers :** La structure de dossiers terraform/environments/\_bootstrap/ a été créée dans le dépôt local skillforge-ai-monorepo pour isoler le code de cette phase d'initialisation.

## 3. Processus de Création du Backend

1. **Définition de la Ressource :** Un fichier bootstrap.tf a été créé dans le dossier \_bootstrap. Ce fichier contenait la configuration pour :
   * Le **fournisseur (provider) Google**, authentifié sur le projet skillforge-ai-mvp-25.
   * Une **ressource google\_storage\_bucket**, définissant un bucket GCS nommé skillforge-ai-mvp-25-tfstate avec le versioning activé.
2. **Exécution des Commandes Terraform :** Les commandes suivantes ont été exécutées depuis le dossier \_bootstrap :
   * terraform init : Pour initialiser le répertoire et télécharger le provider Google.
   * gcloud auth application-default login : Pour autoriser Terraform à utiliser les identifiants de l'utilisateur gcloud.
   * terraform apply : Pour créer le bucket sur Google Cloud.

## 4. Problèmes Rencontrés et Solutions Appliquées

Cette phase a présenté plusieurs défis techniques typiques de la mise en place d'une infrastructure.

### Problème 1 : Erreur de Connectivité Réseau

* **Symptôme :** La commande terraform init a échoué avec une erreur could not connect to registry.terraform.io: context deadline exceeded.
* **Cause Racine :** Un problème de connectivité réseau temporaire empêchait le téléchargement du provider Google depuis le registre officiel de Terraform.
* **Solution Appliquée :** Le problème a été résolu en **relançant simplement la commande terraform init** une seconde fois, après avoir vérifié que la connexion réseau était stable.

### Problème 2 : Erreur d'Authentification (ADC)

* **Symptôme :** La commande terraform apply a échoué en indiquant No credentials loaded et could not find default credentials.
* **Cause Racine :** Terraform n'avait pas l'autorisation d'agir au nom de l'utilisateur. L'authentification gcloud auth login est pour l'utilisateur, mais les applications tierces comme Terraform nécessitent une autorisation supplémentaire (Application Default Credentials).
* **Solution Appliquée :** La commande **gcloud auth application-default login** a été exécutée. Cela a généré un fichier d'autorisation que Terraform a pu utiliser pour s'authentifier avec succès auprès de Google Cloud.

### Problème 3 : Violation de Contrainte d'Organisation

* **Symptôme :** La commande terraform apply a échoué avec l'erreur Request violates constraint 'constraints/storage.uniformBucketLevelAccess'.
* **Cause Racine :** L'organisation Google Cloud (emacsah.com) possède une règle de sécurité par défaut qui impose à tous les nouveaux buckets d'activer l'accès uniforme au niveau du bucket. Notre code Terraform initial n'incluait pas cette directive.
* **Solution Appliquée :** Le fichier bootstrap.tf a été modifié pour inclure explicitement le paramètre **uniform\_bucket\_level\_access = true** dans la définition de la ressource google\_storage\_bucket, satisfaisant ainsi la contrainte de l'organisation.

## 5. Résultat Final

La procédure de bootstrap s'est terminée avec succès par la création du bucket Google Cloud Storage **skillforge-ai-mvp-25-tfstate**. Ce bucket est maintenant prêt à être utilisé comme backend distant pour stocker de manière centralisée et sécurisée l'état de l'infrastructure de l'environnement staging.