Rapport pour le projet final de data engineering

Sacha Papiernik

July 20, 2024

Abstract

Dans ce rapport, je montrerai la structure de base du code terraform que j'ai utilisé pour télécharger des données électorales publiques et les stocker dans une base de données azure. Tout en déployant superset sur une machine virtuelle azure et en gérant le TLS.

1 Introduction

Pour ce rapport, j'ai décidé d'utiliser les données du dépôt public appelé data gov, j'ai voulu utiliser les élections de 2022 et 2024 et j'ai comparé les résultats des deux premiers tours.

Pour faciliter le déploiement sur Azure, j'ai décidé d'utiliser Terraform

2 Main.tf

Je définis ici le fournisseur et je crée un groupe de ressources.

```
provider "azurerm" {
  features {}
}

resource "azurerm_resource_group" "rg" {
  name     = var.resource_group_name
  location = var.location
}
```

3 Network.tf

Cette configuration Terraform crée un réseau virtuel Azure avec un sous-réseau, une adresse IP publique, une interface réseau et un groupe de sécurité réseau avec des règles autorisant le trafic SSH, HTTP, HTTPS et Superset (sur le port 8088), et associe le groupe de sécurité réseau à l'interface réseau.

```
resource "azurerm_virtual_network" "main" {...
}

resource "azurerm_subnet" "main" {...
}

resource "azurerm_public_ip" "superset_ip" {...
}

resource "azurerm_network_interface" "superset_nic" {...
}

resource "azurerm_network_security_group" "superset_nsg" {...
}

resource "azurerm_network_interface_security_group_association" "
    superset_nic_sg" {...
}
```

4 Storage.tf

Cette configuration Terraform crée deux comptes de stockage Azure : un pour le stockage général et un autre pour un lac de données avec l'espace de noms hiérarchique (HNS) activé. Elle crée également un système de fichiers Data Lake Gen2 dans le compte de stockage Data Lake.

```
resource "azurerm_storage_account" "staging" {...
}

resource "azurerm_storage_account" "datalake" {...
}

resource "azurerm_storage_data_lake_gen2_filesystem" "example" {...
}
```

5 Superset.tf

Cette configuration Terraform crée une machine virtuelle (VM) dans Azure avec un système d'exploitation Debian 10, attaché à une interface réseau et à un compte de stockage. Elle utilise ensuite une ressource nulle pour provisionner Apache Superset sur la VM en copiant et en exécutant un script d'installation via SSH.

```
resource "azurerm_virtual_machine" "superset_vm" {..
}

resource "null_resource" "provision_superset" {...
}
```

6 Synapse.tf

Cette configuration Terraform met en place un espace de travail Azure Synapse lié à un système de fichiers Data Lake Gen2, crée un pool SQL dans l'espace de travail Synapse et ajoute une règle de pare-feu pour autoriser l'accès à partir d'une plage d'adresses IP spécifiée.

```
resource "azurerm_synapse_workspace" "synapse_workspace" {...
}

resource "azurerm_synapse_sql_pool" "synapse_sql_pool" {...
}

resource "azurerm_synapse_firewall_rule" "allow_my_ip" {...
}
```

7 install_docker_superset.sh

Ce script, installe Docker, Nginx et les dépendances nécessaires sur un système basé sur Debian. Il configure Docker pour qu'il exécute Apache Superset avec une configuration sécurisée et initialise la base de données Superset. De plus, il génère un certificat SSL auto-signé, configure Nginx pour envoyer des requêtes à Superset avec SSL, et redémarre Nginx pour appliquer la configuration.

8 write_to_db.py

Ce script se connecte à une base de données, crée des tables basées sur les schémas des DataFrames et remplit ces tables avec les données ces DataFrames.

Il définit des fonctions permettant de vérifier si une table existe, de créer des tables à partir des schémas des DataFrames et d'insérer des données par batch.

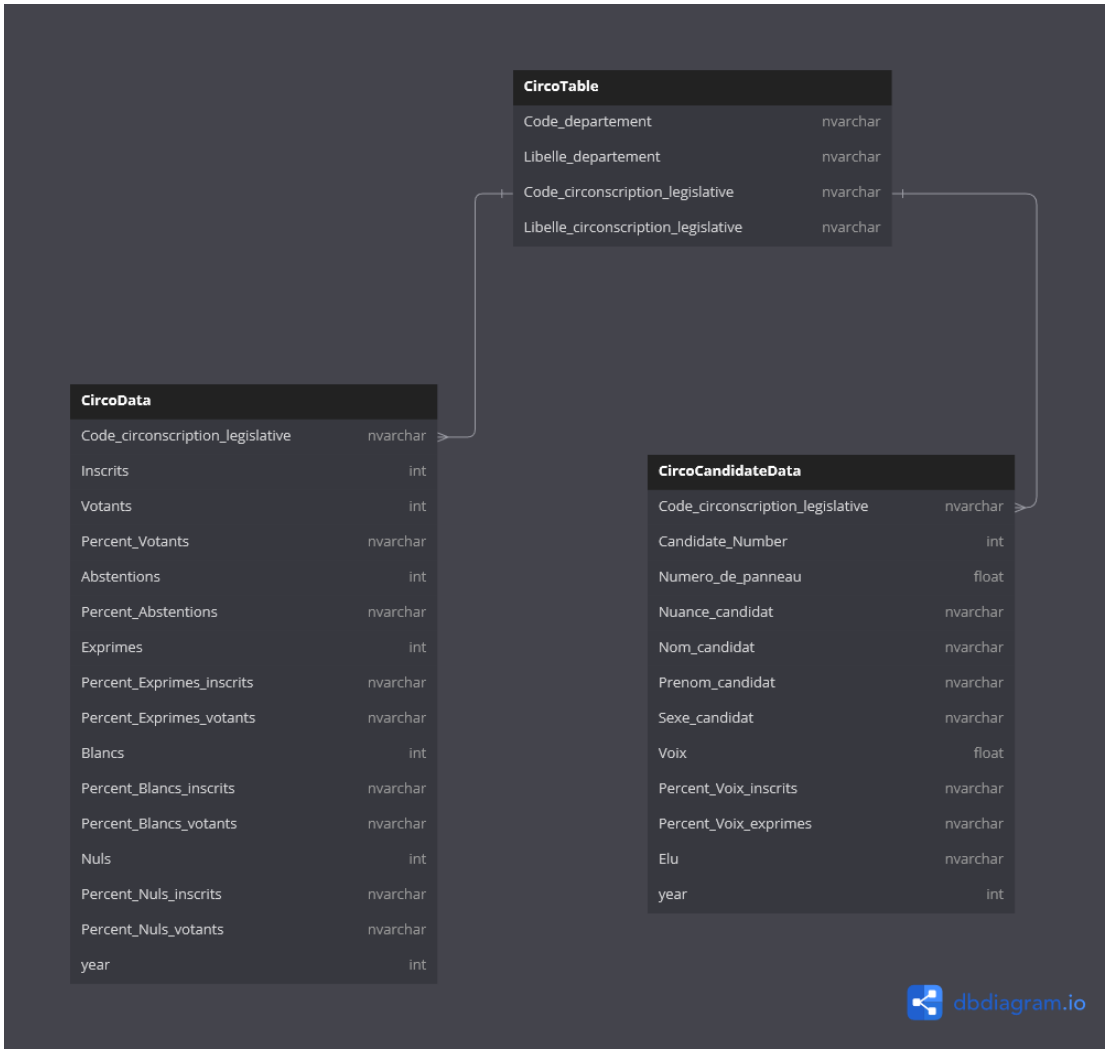
La fonction principale orchestre ces opérations, en utilisant logger pour suivre la progression et gérer les erreurs.

9 function.py

Ce script est appelé par write_to_db.py, pour importer et convertir les données sous une forme utilisable à partir de datagouv.

10 MCD

La base de données est composée de 3 tables, une de l'étiquette Circonscription en elle-même, une avec les données Circonscription des 2 périodes (d'où la relation 1 à plusieurs), et une table des données candidats, (d'où également les relations, 1 à plusieurs).



11 Superset

Bien que j'ai pu installer et exécuter superset, je n'ai pas pu le connecter à la base de données, il semble que ce soit un problème de pilote, mais même après avoir mis à jour le script d'installation pour s'assurer que le conteneur superset a le bon pilote, et même en essayant d'entrer dans le conteneur manuellement via "docker exec -it" et en installant manuellement le pilote manquant, je n'ai pas pu le connecter à la base de données ou à n'importe quelle base de données. Je m'excuse pour le manque de dashboard.