TD2 – Sécurisation réseau AWS (VPC, Subnets, Security Groups)

Objectifs

- Comprendre les bases du réseau AWS (VPC, subnets, routing)
- Identifier les mauvaises pratiques réseau
- Sécuriser une infrastructure existante
- Implémenter une segmentation réseau simple
- Industrialiser avec Terraform

Contexte

Vous intervenez dans la même startup.

Une première infrastructure a été déployée rapidement via Terraform et Ansible.

Objectif initial : mettre en ligne une application web.

Résultat :

- tout fonctionne
- mais aucune règle de sécurité réseau n’a été pensée

Un audit révèle :

- exposition publique excessive
- absence de segmentation réseau
- règles de sécurité trop permissives

Objectif technique

Reconcevoir le réseau pour :

- isoler les composants
- limiter les flux
- respecter les bonnes pratiques AWS

Contraintes

- Ne pas casser l’accès à l’application web
- Appliquer le principe du moindre accès réseau
- Utiliser Terraform uniquement
- Architecture simple et compréhensible

Infrastructure existante

Fichier : `network/main.tf`

```
provider "aws" {
  region = "eu-west-3"
}

resource "aws_vpc" "main" {
  cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "public" {
  vpc_id     = aws_vpc.main.id
  cidr_block = "10.0.1.0/24"
}

resource "aws_instance" "web" {
  ami           = "ami-123456"
  instance_type = "t2.micro"
  subnet_id     = aws_subnet.public.id
}

resource "aws_security_group" "web_sg" {
  name   = "web-sg"
  vpc_id = aws_vpc.main.id

  ingress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

Ressources

Fichier : `commandes/terraform.txt`

```
terraform init
terraform plan
terraform apply
terraform destroy
```

1. Compréhension de l’existant

Quels composants réseau sont présents ?

Que manque-t-il pour une architecture AWS correcte ?

Pourquoi cette infrastructure fonctionne malgré tout ?

Quels sont les risques immédiats ?

2. Analyse des problèmes

Identifier les problèmes de sécurité :

- au niveau du VPC
- au niveau des subnets
- au niveau des Security Groups

Quelle est la règle la plus dangereuse dans ce code ?

Pourquoi ?

3. Mise en pratique – segmentation réseau

Modifier l’architecture pour :

- créer 2 subnets :
  - public (web)
  - privé (backend ou base de données)
- ajouter un Internet Gateway
- rendre uniquement le subnet public accessible depuis Internet

4. Mise en pratique – sécurisation des flux

Corriger les Security Groups :

- Autoriser uniquement :
  - HTTP (80) depuis Internet vers le serveur web
  - SSH (22) uniquement depuis votre IP
- Interdire tout le reste

Point d’attention (erreur volontaire)

Le code actuel autorise :

- tous les ports
- tous les protocoles
- depuis n’importe où

Pourquoi cette configuration est-elle fréquente en phase de développement ?

Pourquoi devient-elle critique en production ?

5. Apparition d’un problème

Après sécurisation :

- votre application web ne répond plus

Quelles sont les causes possibles ?

Quels éléments réseau faut-il vérifier ?

6. Analyse

Vérifier :

- association du Security Group
- table de routage
- subnet public vs privé
- présence d’une IP publique

Lequel de ces éléments est souvent oublié ?

7. Correction

Corriger l’infrastructure pour :

- restaurer l’accès web
- conserver les règles de sécurité

8. Extension

Ajouter :

- un NAT Gateway pour le subnet privé

- permettre au backend d'accéder à Internet sans être exposé

Pourquoi ne faut-il jamais exposer directement une base de données sur Internet ?

9. Bonnes pratiques

Pourquoi faut-il privilégier :

- plusieurs subnets ?
- des Security Groups spécifiques par rôle ?
- une séparation public / privé ?

Challenge final

Objectif :

- Le serveur web doit être accessible depuis Internet (HTTP)
- Le backend doit être inaccessible depuis Internet
- Le backend doit être accessible uniquement depuis le serveur web

Implémenter cette architecture avec :

- 2 Security Groups
- règles de flux précises

Pourquoi utiliser un Security Group comme source plutôt qu'une IP ?

Bonus

Ajouter :

- un Load Balancer en frontal
- plusieurs instances web

Quel est l'impact sécurité d'un Load Balancer par rapport à une instance seule ?