# TP 1_Sécurité des Réseaux & Protocoles

**Support pédagogique : AWS Learner Lab – Ubuntu**

---

# PARTIE 0 — Introduction & objectifs pédagogiques

## Objectif

Comprendre :

- ce qu'est un **réseau**
- ce qu'est un **protocole**
- pourquoi la **sécurité réseau est indispensable**

## À retenir

- EC2 = un ordinateur
- VPC = un réseau
- Security Group = pare-feu
- Port = porte
- Protocole = type de communication

---

# 🔷 PARTIE 1 — Création de l'architecture (EC2 + VPC + Ubuntu)

## Objectif

Créer un **PC distant dans un réseau**, sans encore parler sécurité.

---

## Manipulations

1. Se connecter à **AWS Learner Lab**
2. Cliquer sur **Start Lab**
3. Ouvrir **AWS Console**
4. Aller dans **EC2 → Launch instance**

## Paramètres :

- Nom : `TP-Securite-Reseau`
- OS : **Ubuntu Server 22.04**
- Type : **t2.micro**
- Key pair : créer `tp-ubuntu-key.pem`
- Réseau :
  - VPC : Default
  - Public IP : Enable
- **Security Group : ne rien modifier**
- Lancer l'instance

---

### 📸 Capture demandée

- Instance **Running** avec IP publique

📄 Nom :

`01_instance_ec2_running.png`

---

# 🔷 PARTIE 2 — Observer le blocage par défaut (sécurité réseau)

## Objectif

Comprendre que **tout est bloqué par défaut**.

---

### 🧪 Tests à faire (depuis le PC étudiant)

#### Test 1 — Ping (ICMP)

```
ping IP_EC2
```

❌ Échec

#### Test 2 — SSH (TCP 22)

```
ssh ubuntu@IP_EC2
```

❌ Échec

---

### 📸 Captures demandées

```
02_security_group_default_block.png
03_ping_bloque.png
04_ssh_bloque.png
```

---

### Conclusion pédagogique

Bloquer par défaut = règle n°1 de la sécurité réseau

---

# 🔷 PARTIE 3 — Sécuriser le protocole ICMP (Ping)

## Objectif

Comprendre qu'un **protocole doit être explicitement autorisé**.

---

## 🧩 Manipulations

1. Aller dans **Security Group**
2. Edit inbound rules
3. Ajouter :
    - Type : All ICMP – IPv4
    - Source : My IP
4. Save

---

## 🧪 Test

```
ping IP_EC2
```

✅ OK

---

## 📸 Captures demandées

```
05_regle_icmp.png
06_ping_ok.png
```

---

# 🔷 PARTIE 4 — Sécuriser le protocole SSH (TCP 22)

### Objectif

Comprendre qu'un **accès distant est sensible**.

---

### 🧩 Manipulations

1. Ajouter règle :
    - Type : SSH
    - Port : 22
    - Source : My IP

---

### 🧪 Connexion

```
ssh -i tp-ubuntu-key.pem ubuntu@IP_EC2
```

✅ Connexion réussie

---

### 📸 Captures demandées

```
07_regle_ssh.png
08_ssh_ok.png
```

---

### Message clé

SSH ne doit JAMAIS être ouvert à tout Internet

---

# 🔷 PARTIE 5 — Sécuriser le protocole HTTP (TCP 80)

### Objectif

Comprendre la différence entre **service public et service privé**.

---

### Manipulations

1. Ajouter règle :
   - Type : HTTP
   - Port : 80
   - Source : 0.0.0.0/0
2. Installer Apache :

```
sudo apt update
sudo apt install apache2 -y
sudo systemctl start apache2
```

---

### 🧪 Test navigateur

```
http://IP_EC2
```

✅ Page Apache visible

---

### 📸 Captures demandées

```
09_regle_http.png
10_apache_running.png
11_http_page.png
```

---

# 🔷 PARTIE 6 — Analyse réseau depuis Ubuntu

### 🎯 Objectif

Voir **les ports ouverts et les connexions réseau**.

---

### 🧪 Commandes à exécuter

```
ip a
ss -tuln
ping 8.8.8.8
curl http://example.com
```

---

### 📸 Capture demandée

12_ports_ouverts.png

---

# 🔷 PARTIE 7 — Synthèse sécurité réseau & protocoles

## 🎯 Objectif

Faire le lien entre **réseau, protocole et sécurité**.

---

### Tableau à compléter par l'étudiant

| Protocole | Port | Autorisé à | Risque |
|---|---|---|---|
| ICMP | — | My IP | Scan |
| SSH | 22 | My IP | Intrusion |
| HTTP | 80 | Public | Attaques web |

---

## 📸 Capture finale

13_security_group_final.png