

## SAE24 - Administration services réseaux

### Contexte :

Ce travail s'inscrit dans le cadre du projet SAE24 du BUT Réseaux & Télécommunications, dont le but est de mettre en place une infrastructure complète de services réseau au sein d'une entreprise simulée.

Le projet a été réalisé en équipe, chaque membre étant responsable d'un ou plusieurs services.

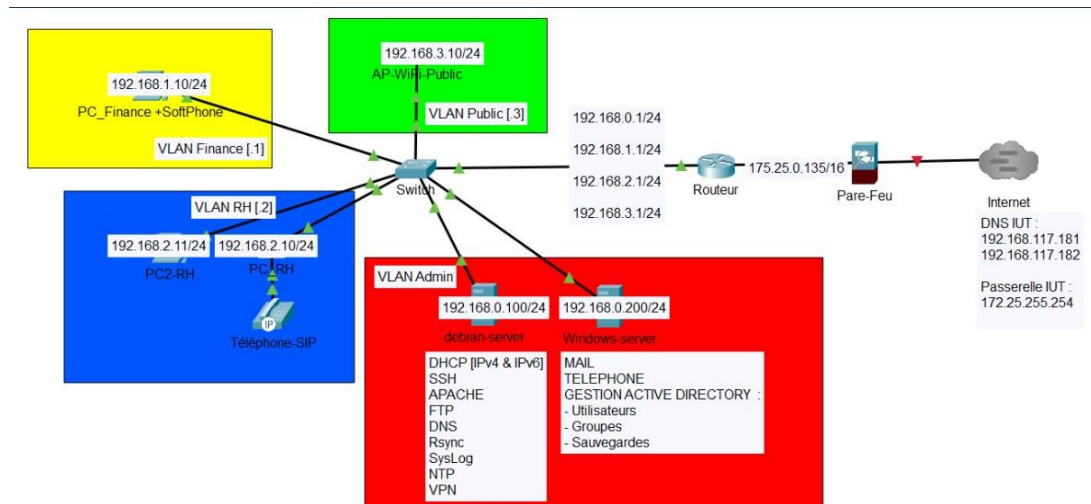
Pour ma part, j'ai été en charge de la configuration du serveur Linux (routage, DHCP, Apache) ainsi que de la mise en place du serveur de téléphonie Asterisk.

### Objectif :

L'objectif de cette SAE était de construire et d'administrer une architecture réseau complète et fonctionnelle, permettant la communication entre différents services et utilisateurs.

Le projet devait inclure la mise en place de services essentiels tels que le DHCP, le web, la téléphonie IP et le routage.

Plan global de l'entreprise:



Le réseau de l'entreprise est composé de plusieurs sous-réseaux :

- Un réseau administratif
- Un réseau utilisateurs
- Un réseau téléphonie

- Un réseau serveurs (hébergeant Linux, Apache, Asterisk, DHCP, etc.)
- Une passerelle connectée à Internet

## **Travail réalisé :**

### **Première partie : Configuration du serveur Linux**

J'ai commencé par configurer la carte réseau principale du serveur Linux pour lui attribuer une adresse IP statique.

Pour cela, j'ai modifié le fichier `/etc/network/interfaces` :

```
auto enp0s3
iface enp0s3 inet static
    address 192.168.1.1
    netmask 255.255.255.0
    gateway 192.168.1.254
```

Après avoir redémarré le service réseau, j'ai vérifié la connectivité par un ping vers les machines clientes.

J'ai ensuite activé le routage IP en modifiant le fichier `/etc/sysctl.conf` et en décommentant la ligne :

```
net.ipv4.ip_forward=1
```

Puis, j'ai ajouté une règle NAT pour permettre la sortie vers Internet :

```
iptables -t nat -A POSTROUTING -o enp0s8 -j MASQUERADE
```

### **Deuxième partie : Mise en place du service DHCP**

J'ai installé le paquet `isc-dhcp-server` avec la commande :

```
sudo apt install isc-dhcp-server
```

Dans le fichier `/etc/default/isc-dhcp-server`, j'ai précisé l'interface utilisée par le service :

```
INTERFACESv4="enp0s3"
```

Ensuite, j'ai configuré le fichier `/etc/dhcp/dhcpd.conf` pour définir la plage d'adresses IP à distribuer, le routeur par défaut et le DNS :

```
subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.10 192.168.1.100;
    option routers 192.168.1.1;
    option domain-name-servers 8.8.8.8;
    option domain-name "entreprise.local";
}
```

Après avoir redémarré le service (*systemctl restart isc-dhcp-server*), j'ai vérifié son état avec :

```
systemctl status isc-dhcp-server
```

Sur le poste client, j'ai configuré l'interface en DHCP. Après redémarrage, la machine a bien reçu une adresse IP du serveur.

J'ai également utilisé Wireshark pour observer les échanges DHCP (Discover, Offer, Request, ACK).

Les baux actifs ont été vérifiés dans le fichier */var/lib/dhcp/dhcpd.leases*.

### Troisième partie : Mise en place du serveur web Apache

J'ai installé le serveur Apache2 avec :

```
sudo apt install apache2
```

Une fois le service démarré, j'ai créé une page de test dans */var/www/html/index.html* :

```
<html>
  <head><title>Serveur Web - SAE24</title></head>
  <body><h1>Serveur Web opérationnel</h1></body>
</html>
```

J'ai ensuite vérifié le bon fonctionnement depuis un navigateur client en accédant à l'adresse IP du serveur.

Le site s'affichait correctement, prouvant le bon fonctionnement du service Apache.

### Quatrième partie : Mise en place du serveur de téléphonie Asterisk

J'ai choisi d'installer la solution **FreePBX** sur une machine virtuelle Debian 12 (64 bits), en lui allouant 40 Go de stockage et 2 Go de RAM. Une fois l'installation terminée, j'ai configuré l'adresse IP du serveur sur **172.25.254.201** pour l'intégrer au VLAN téléphonie.

Pour assurer le bon démarrage des modules de virtualisation nécessaires au serveur, j'ai dû créer un script de service dans le fichier */etc/systemd/system/vmware-startup.service* :

```
[Unit]
Description=Initialisation de VMware modules
After=network.target
```

```
[Service]
Type=oneshot
```

```
ExecStart=/sbin/modprobe vmmon  
ExecStartPost=/bin/systemctl restart vmware
```

[Install]

```
WantedBy=multi-user.target
```

J'ai ensuite sécurisé et ouvert les flux nécessaires. J'ai autorisé les adresses IP des téléphones via la commande `fwconsole firewall trust`. Puis, j'ai configuré iptables pour autoriser le protocole SIP et les flux RTP (voix) :

```
sudo iptables -A INPUT -p udp --dport 5060 -j ACCEPT
```

```
sudo iptables -A INPUT -p udp --dport 10000:20000 -j ACCEPT
```

Via l'interface web de FreePBX, j'ai créé les extensions utilisateurs (ex: 1351 pour la Finance, 1352 pour les RH) et activé la messagerie vocale accessible par le code \*97. J'ai également configuré un **Trunk SIP** vers l'opérateur (10.172.5.10) pour gérer les appels entrants et sortants.

J'ai enfin configuré les postes physiques (SIP55) via leur interface web. Les tests d'appels internes et vers l'extérieur ont confirmé le bon fonctionnement du service.

## Résultat :

Le serveur Linux remplit correctement son rôle central de distribution d'adresses IP, d'hébergement web et de gestion d'appels internes.  
Les tests réalisés se sont révélés concluants.

Ce projet m'a permis de développer mes compétences en administration réseau sous Linux. J'ai appris à manipuler et configurer différents services, à diagnostiquer leurs interactions et à documenter leurs comportements.

La partie téléphonie m'a particulièrement permis de comprendre les principes de la VoIP et du protocole SIP.

Cette SAE m'a apporté une vision plus complète de la gestion d'un réseau d'entreprise et de la collaboration entre plusieurs services.