Rapport SAÉ 3.01B

Compétence 3 (Réseau)

Chef du Groupe :

- Maxime CHAGNON

Membres de l'équipe :

- Mikkel ALMONTE--RINGAUD
- Michel WANG
- Sasha COULOMBEL
- Maxime CHAGNON
- Antoine BERTELOOT

Table des Matières :

1. Contexte de la SAÉ **2**

2. Analyse de besoins **3**

3. Découpage réseau **4**

4. Choix par rapport au filtrage **6**

5. Evaluation de l'efficacité de notre solution **7**

6. Améliorations possible **8**

1. Contexte de la SAÉ

L'objectif de cette SAE est de garantir une sécurité dans l'infrastructure réseau d'un centre hospitalier universitaire (CHU). Il est donc primordial que ce réseau donne l'accès aux services demandés, et en limitant les accès non autorisés tout en préservant les données sensibles.

On retrouve notamment plusieurs catégories d'utilisateurs : les **patients**, les **visiteurs**, les **enseignants**, les **chercheurs**, les **enseignants-chercheurs**, les **étudiants**, le **personnel soignant**, ainsi que la **DSI**. Chaque catégorie a ses propres besoins, et il est donc très important de garantir leur sécurité.

2. Analyse de besoins

Les principales machines à sécuriser sont les suivantes

La machine PC_S.
Elle héberge le site public (disponible en HTTP/HTTPS), l'Intranet (disponible aussi en HTTP/HTTPS) ainsi que le serveur pour l'application de gestion des patients et des rendez-vous.

La machine PC_BDD.
Elle héberge une base de données MySQL ainsi qu'un SFTP. Elle est notamment utilisée pour le stockage des informations médicales sensibles et pour fournir des données sécurisées aux chercheurs.

RÉSEAUX	ACCÈS
Réseau **Patients** et **Visiteurs**	- Accès au site web HTTP/HTTPS
Réseau **Étudiants** et **Enseignants**	- Accès au site web HTTP / HTTPS - Accès à l'Intranet HTTP/HTTPS - Accès aux mails
Réseau **Chercheurs** et **Enseignant-Chercheurs**	- Accès au site web HTTP/HTTPS - Accès à l'intranet HTTP/HTTPS - Accès aux mails - Accès SFTP à la machine BDD
Réseau **Comptabilité** et **Administration** :	- Accès au site web HTTP/HTTPS - Accès à l'intranet HTTP/HTTPS - Gestion des patients (TCP/1024) - Accès aux mails
Réseau **Personnel Soignant**	- Accès au site web HTTP/HTTPS - Gestion des patients (TCP/1024) - Accès aux mails
Réseau **DSI** (RSSI, MAIL, BDD, AUX) :	- PING à toutes les machines - Accès à tous les services

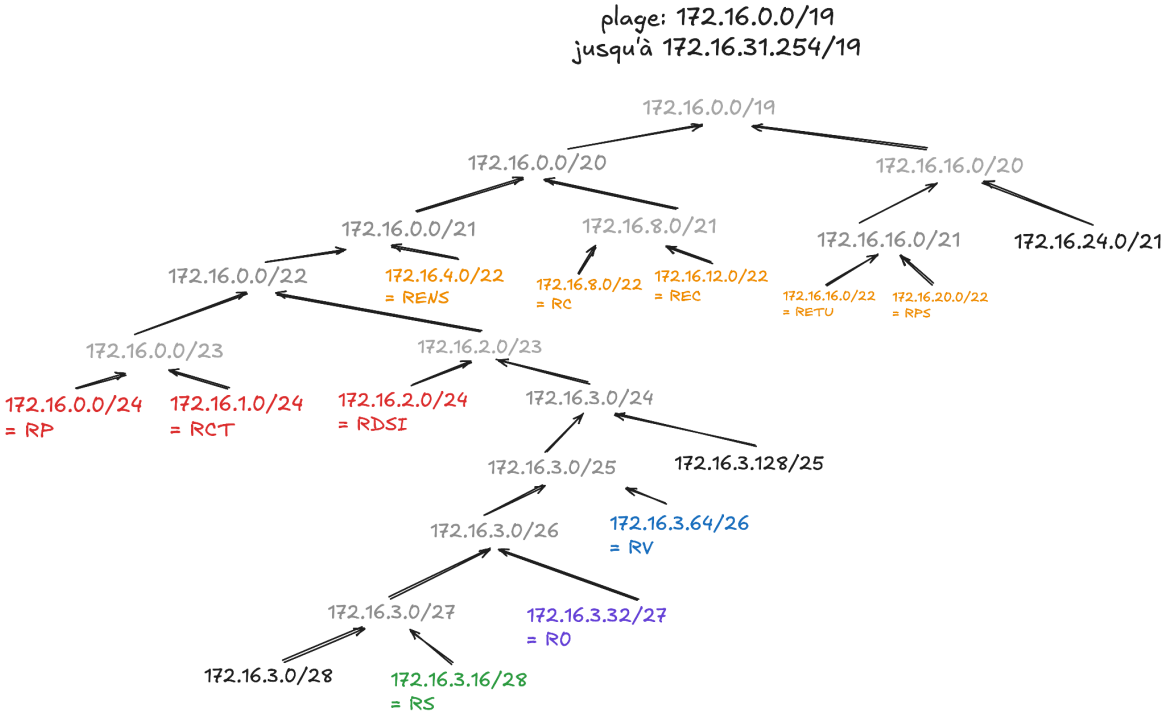
La machine PC_MAIL.
Assure la gestion des communications internes et permet la double authentification pour l'accès aux différents services importants.

La machine PC_AUX.
Elle stocke des sauvegardes des données importantes pour assurer la continuité en cas de cyberattaque ou de panne.

3. Découpage réseau

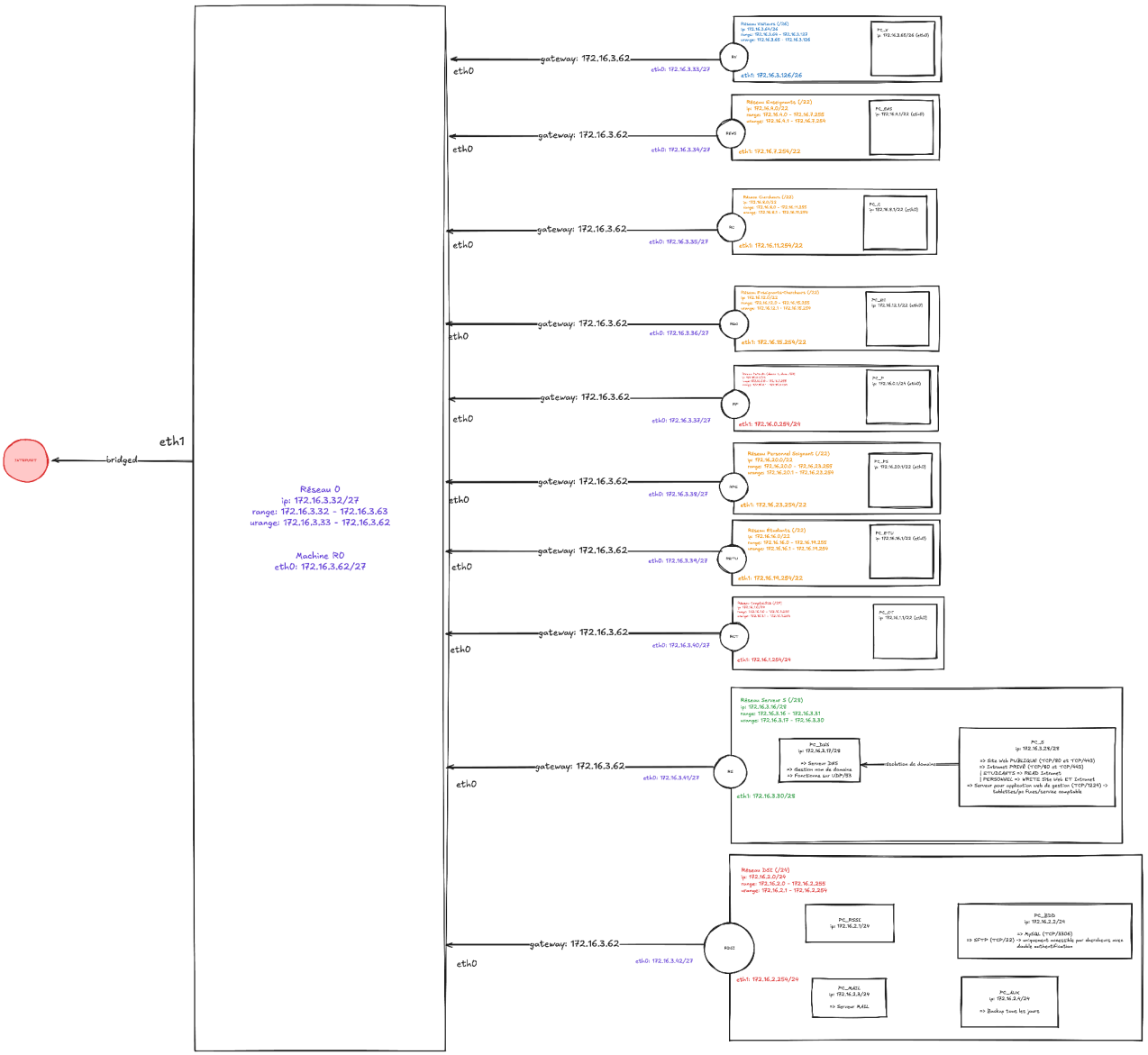
Nous savons que la machine S doit être sur l'IP 172.16.3.28/28
Ainsi, on peut déterminer la plage racine : 172.16.0.0/19

Voici le découpage que nous proposons, dans le format d'un arbre des adresses réseaux.



« Arbre des adresse réseau »

Voici un schéma de vue d'ensemble de notre infrastructure (que vous pouvez retrouver sur le GitHub de la SAE : https://raw.githubusercontent.com/username2000w/SAE-3.01B/refs/heads/main/infrastructure.png)



« Schéma réseau du CHU »

4. Choix par rapport au filtrage

Nous utilisons iptables pour effectuer le filtrage.

Lorsque nous utilisons le filtrage, nous appliquons seulement un ensemble limité d'IP.
Par exemple, pour le protocole ICMP (pour la commande ping, uniquement accessible depuis le réseau DSI), nous autorisons l'IP réseau directement car celà nous évite de mentionner les IP de chaque machine dans le réseau DSI qui a le droit de ping la machine.

```
13  # PING (ICMP) pour DSI
14  iptables -A INPUT -s 172.16.2.0/24 -p icmp -j ACCEPT
15  iptables -A OUTPUT -d 172.16.2.0/24 -p icmp -j ACCEPT
```

Vu que l'adresse du DNS est fixe, nous appliquons une règle très stricte qui autorise uniquement l'IP de celle-ci sur le port exact.

```
17  # DNS (53)
18  iptables -A INPUT -s 172.16.3.17 -p udp --sport 53 -j ACCEPT
19  iptables -A OUTPUT -d 172.16.3.17 -p udp --dport 53 -j ACCEPT
```

Chaque machine qui a un filtrage refuse toutes les requêtes par défaut, sinon le filtrage n'a aucun effet.

```
 9  iptables -P FORWARD DROP
10  iptables -P OUTPUT DROP
11  iptables -P INPUT DROP
```

Nous faisons un filtrage INPUT/OUTPUT sur les machines qui ont un service (exemple avec PC_S ici)

```
29  # INTERNAL SERVER RDV (1224) pour personnel soignants (PS)
30  iptables -A INPUT -s 172.16.20.0/22 -p tcp --dport 1224 -j ACCEPT
31  iptables -A OUTPUT -d 172.16.20.0/22 -p tcp --sport 1224 -j ACCEPT
32
33  # INTERNAL SERVER RDV (1224) pour comptabilité (CT)
34  iptables -A INPUT -s 172.16.1.0/24 -p tcp --dport 1224 -j ACCEPT
35  iptables -A OUTPUT -d 172.16.1.0/24 -p tcp --sport 1224 -j ACCEPT
36
37  # INTERNAL SERVER RDV (1224) pour DSI
38  iptables -A INPUT -s 172.16.2.0/24 -p tcp --dport 1224 -j ACCEPT
39  iptables -A OUTPUT -d 172.16.2.0/24 -p tcp --sport 1224 -j ACCEPT
40
41  # MySQL (3306)
42  iptables -A INPUT -s 172.16.2.2 -p tcp --sport 3306 -j ACCEPT
43  iptables -A OUTPUT -d 172.16.2.2 -p tcp --dport 3306 -j ACCEPT
44
```

Les routeurs vont toujours refuser INPUT/OUTPUT et uniquement gérer FORWARD pour rediriger les requêtes vers les machines respectives qui vont ensuite gérer la requête avec INPUT/OUTPUT. Voici un exemple avec le routeur pour la DSI qui est assez permissif.

```
 9  iptables -P INPUT DROP
10  iptables -P FORWARD ACCEPT
11  iptables -P OUTPUT DROP
12
```

Mais il peut être plus sophistiqué pour uniquement FORWARD des requêtes précises pour, encore une fois, une précision exacte sur les machines/réseaux qui ont le droit de passer.

```
 9  iptables -P FORWARD DROP
10  iptables -P OUTPUT DROP
11  iptables -P INPUT DROP
12
13  # PING (ICMP) pour la DSI
14  iptables -A FORWARD -s 172.16.2.0/24 -p icmp -j ACCEPT
15  iptables -A FORWARD -d 172.16.2.0/24 -p icmp -j ACCEPT
16
17  # DNS (53)
18  iptables -A FORWARD -s 172.16.3.17 -p udp --sport 53 -j ACCEPT
19  iptables -A FORWARD -d 172.16.3.17 -p udp --dport 53 -j ACCEPT
20
21  # HTTP (80)
22  iptables -A FORWARD -p tcp --dport 80 -j ACCEPT
23  iptables -A FORWARD -p tcp --sport 80 -j ACCEPT
```

5. Evaluation de l'efficacité de notre solution

Notre solution répond quasiment aux objectifs fixés.

- Le serveur DNS pourrait garantir la résolution des noms s'il était implémenté
- Les accès sont conformes en fonction des différents besoins de chaque groupe d'utilisateurs
- Un filtrage des accès entre tous les sous-réseaux, qui bloquent tout trafic non autorisé

Nous avons testé notre solution avec un script qui vérifie les accès sur chaque machine : *erreur si on a accès à une machine que l'on ne devrait pas, erreur si l'on a pas accès à une machine que l'on devrait avoir*.

6. Améliorations possible

Il serait utile de pouvoir implémenter les services pour pouvoir appliquer correctement la consigne demandée : la non-existence du serveur DNS ne permet pas de vérifier que les noms de domaines fonctionnent correctement.

Pour nos tests nous avons utilisé netcat et il est vraiment possible d'améliorer la qualité de notre filtrage en ayant les vrais services car ils ne fonctionnent pas tous pareil.

Concernant l'intranet, vu qu'il n'y a pas de serveur web (apache2, nginx ou caddy) ni de serveur DNS, il est compliqué de pouvoir simuler une protection entre le site web public et le site web interne. Les deux seront sur les ports 80/443, il faudrait des hôtes virtuels pour différencier les deux sites en fonction de leur nom de domaine.