TP5 : translation d'adresse (NAT) et pare-feu

Le but de ce TP est de montrer en pratique comment on peut sécuriser un réseau. Dans la première partie du TP, on utilisera une configuration réseau proche de celle de votre domicile : des machines personnelles cachées derrière une box. Dans une second partie, on se rapprochera de la situation d'une entreprise : certaines machines accèdent à des services particuliers du réseau externe, et seuls certains services sont visibles depuis le réseau externe.

Conventions

Dans ce TP, les interfaces du routeur auront comme numéro de machine `1`. De plus, on s'arrangera pour que le réseau externe (qui réprésente Internet, le réseau public) soit toujours branché sur le port `0` du routeur. Les adresses privées du réseau `10.11.0.0` de ces exercices représentent des adresses publiques Internet dans les cas d'utilisation traités.

Partie 1 : cas d'un réseau personnel

Nous allons utiliser le réseau suivant pour la première partie du TP :

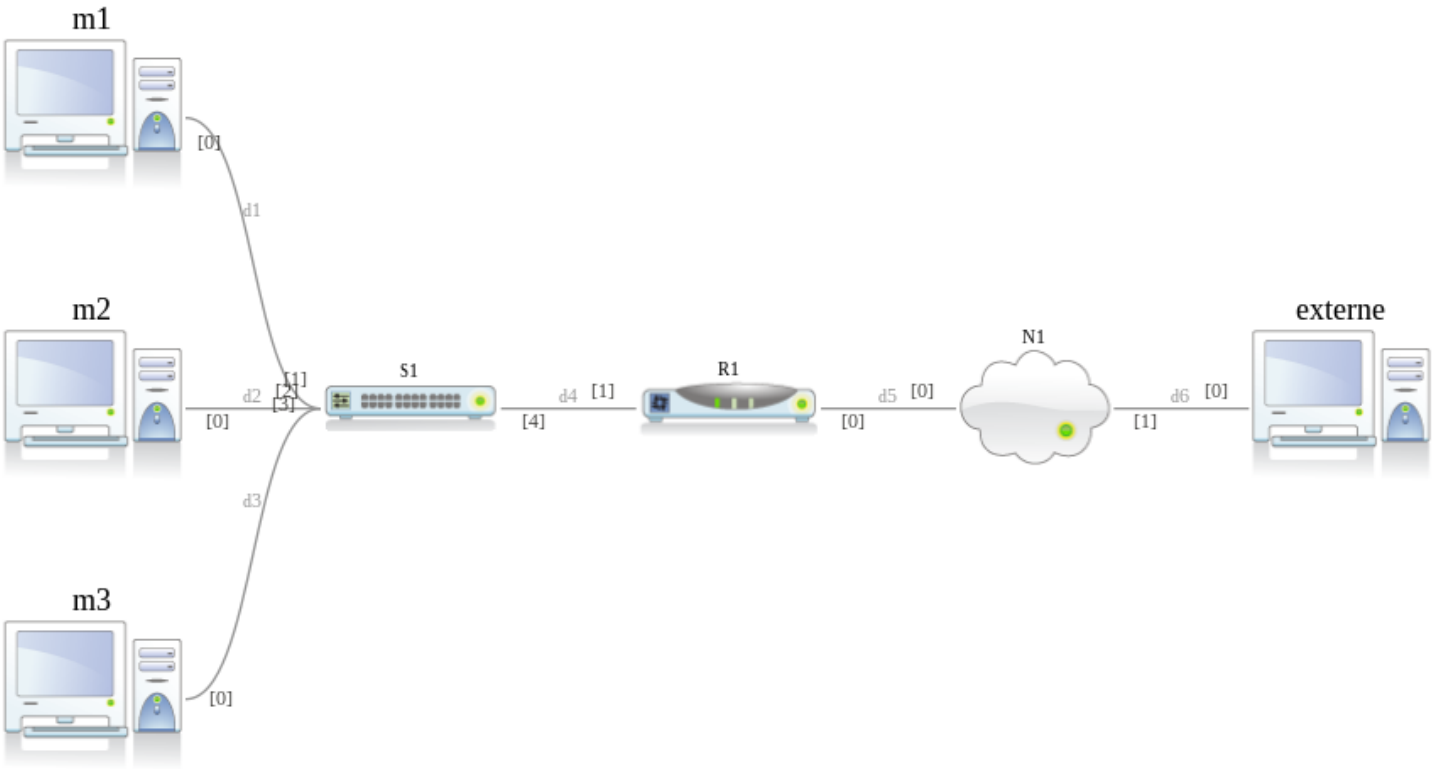

Figure 1: Un réseau personnel

Les machines `m1`,`m2` et `m3` représentent des machines personnelles, branchées sur le réseau local d'un logement. Le commutateur et le routeur correspondent à une box (qui comprend ces deux éléments). La machine `externe` représente une machine accessible sur Internet. Le sous-réseau Marionnet (le nuage) représente une interconnexion de machines qui se trouvent sur le même réseau que `externe`.

Les machines ont les adresses IP suivantes :

- m1 192.168.11.2/24
- m2 192.168.11.3/24
- m3 192.168.11.4/24
- externe 10.11.1.2/24

Le routeur sera configuré de la façon suivante :

- Choisir une distribution `the-one-and-only` et un noyau `2.6.18.ghost`
- port0 10.11.1.1/24
- port1 192.168.11.1/24

Cocher la case "Show unix terminal" dans la configuration du routeur, cela nous permettra de le configurer.

1. Démarrer les machines et configurer les passerelles par défaut pour que toutes les machines puissent communiquer entre elles.
2. Se connecter sur la machine `externe` et lancer l'utilitaire tcpdump à l'aide de la commande `tcpdump -n`.
3. Se connecter sur `m1` et lancer la commande `ping 10.11.1.2`. Vérifiez l'adresse source des paquets reçus par `externe`. Vous devez voir `192.168.11.2` comme adresse source.

Première étape : cacher les adresses privées

Afin d'éviter que les adresses privées n'apparaissent dans le réseau public, il faut demander au routeur de remplacer les adresses privées par son adresse "publique". On appelle cela la *translation d'adresses*.

On utilisera la commande suivante pour demander au routeur de cacher les adresses de destination des paquets qui sortent sur l'interface `eth0`.

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Lancer la commande `ping 10.11.1.2` sur m1. Vérifiez l'adresse source des paquets reçus par `externe`. Vous devez voir `10.11.1.1` comme adresse source.

Il devient ainsi possible de communiquer entre les machines du réseau privé et le réseau public sans passerelle.

Sur la machine `externe`, on pourrait enlever la route vers le réseau privé :

```
route del default gw 10.11.1.1
```

(on en a cependant besoin pour le reste de l'exercice).

Deuxième étape : empêcher les intrusions dans le réseau local

La deuxième mesure importante pour sécuriser le réseau local est d'empêcher les connexions entrantes dans le réseau.

On utilisera la commande suivante pour demander au routeur d'empêcher les connexions entrantes depuis l'interface `eth0`.

```
iptables -P FORWARD DROP
iptables -A FORWARD -o eth0 -j ACCEPT
iptables -A FORWARD -i eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

Essayer de contacter la machine m1 à partir de la machine `externe`. Qu'observe t'on ?

Partie 2 : cas d'un réseau d'entreprise

Nous allons utiliser le réseau suivant pour la deuxième partie du TP :

Les machines `m1` et `m2` sont des machines classiques des personnels de l'entreprise. Les machines `m3` et `m4` sont des machines particulières pour l'entreprise. `m3` est un serveur SSH et `m4` est un serveur web. Ces machines sont sur un réseau séparé des autres machines de l'entreprise : elles vont se trouver dans une *zone démilitarisée (DMZ)*.

Les machines ont les adresses IP suivantes :

- m1 192.168.11.2/24
- m2 192.168.11.3/24
- m3 192.168.12.2/24
- m4 192.168.12.3/24
- externe 10.11.1.2/24

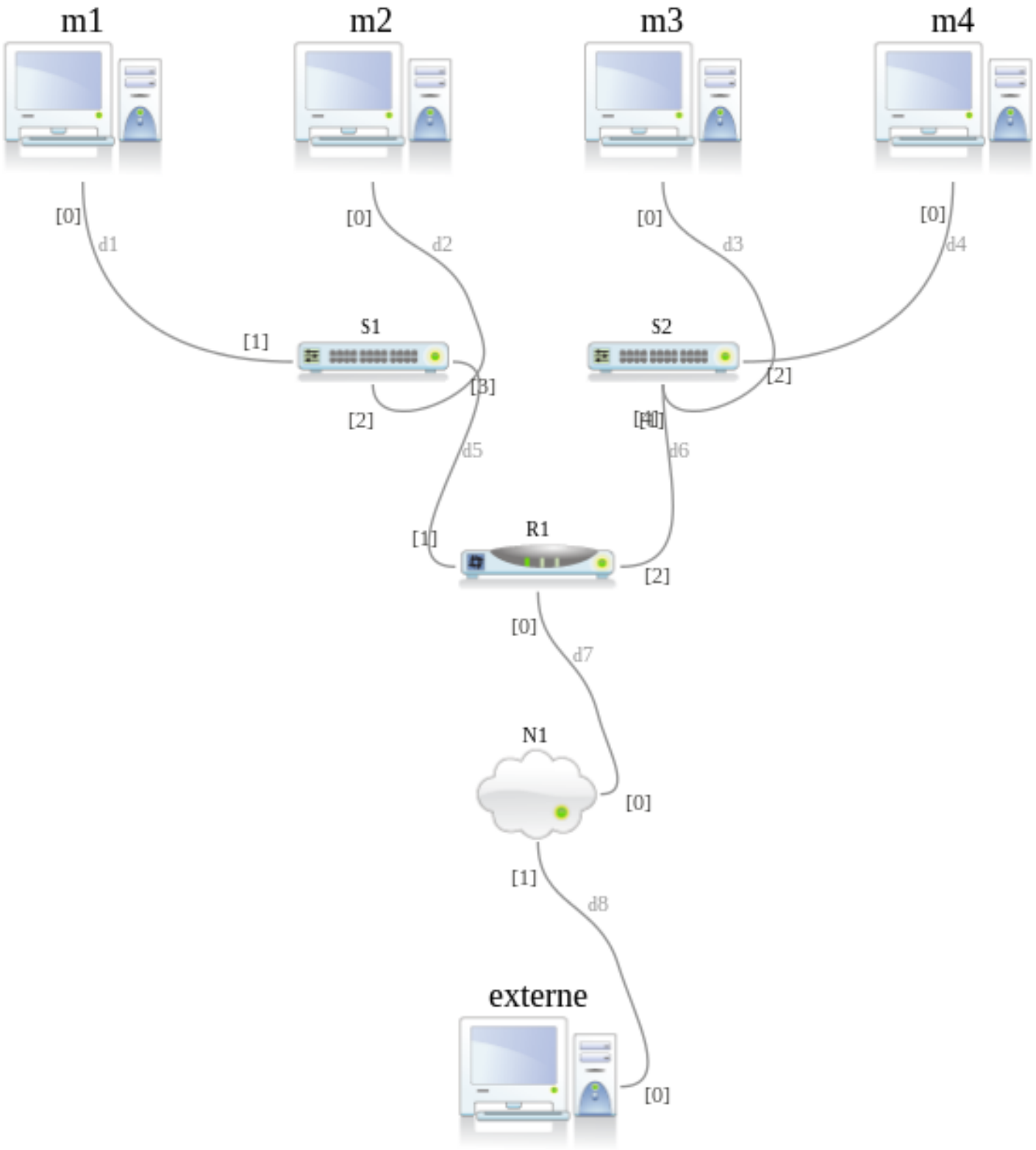

Figure 2: Un réseau d'entreprise

Le routeur sera configuré de la façon suivante :

- Choisir une distribution `the-one-and-only` et un noyau `2.6.18.ghost`
- port0 10.11.1.1/24
- port1 192.168.11.1/24
- port2 192.168.12.1/24

Cocher la case "Show unix terminal" dans la configuration du routeur, cela nous permettra de le configurer.

1. Vérifier que les machines peuvent communiquer entre elles (ne pas oublier d'activer les passerelles par défaut sur chaque machine).
2. Activer le service web sur la machine `m4`.

   On pourra changer le message affiché par la page d'accueil du serveur web en modifiant la page par défaut du serveur : `nano /var/www/index.html`

   Pour activer le serveur web, lancer la commande suivante sur `m4` : `/etc/init.d/apache2 start`
3. Vérifier que le serveur web de `m4` est bien accessible de `m1` en utilisant la commande suivante : `lynx 192.168.12.3`

   Lynx est un navigateur web en mode texte, très utile pour naviguer en mode console.
4. Activer de manière similaire un serveur web sur la machine `externe`.

   On pourra changer le message affiché par la page d'accueil du serveur web en modifiant la page par défaut du serveur : `nano /var/www/index.html`
5. Activer le service SSH sur la machine `m3`. Pour cela, lancer la commande suivante sur `m3` : `/etc/init.d/ssh start`
6. Vérifier que le serveur SSH est bien accessible de `m1` en vous y connectant : `ssh 192.168.12.2`

Cacher l'adresse des machines du réseau privé par translation d'adresses (NAT).

En utilisant la même commande que pour le réseau personnel, configurer le routeur pour qu'il cache les adresses privées du réseau lors des communications avec la machine `externe`.

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Vérifier à l'aide de `tcpdump` que les paquets reçus par la machine `externe` ne mentionnent plus d'adresses en `192.168.x.x`.

Empêcher les connexions externes vers les sous-réseaux privés

Dans notre situation, nous souhaitons éviter toute intrusion depuis l'extérieur sur les machines des réseaux privés. Il faut donc demander au routeur d'empêcher toute connexion entrante de l'interface `eth0` vers `eth1` ou `eth2`, et ne permettre que les sorties sans contraintes des connexions venant de `eth1`.

```
iptables -P FORWARD DROP
iptables -A FORWARD -o eth0 -i eth1 -j ACCEPT
iptables -A FORWARD -i eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

1. Vérifier qu'il n'est plus possible d'atteindre une machine des sous-réseaux privés depuis la machine `externe`.
2. Vérifier qu'il est possible d'atteindre la machine `externe` depuis la machine `m1`.
3. Vérifier qu'il n'est plus possible d'atteindre la machine `externe` depuis la machine `m3`.

Permettre les connexions entre les deux réseaux privés

Avec la configuration actuelle, est-il possible d'accéder aux machines `m3` et `m4` depuis la machine `m1` ?

Pour corriger ce problème, on va permettre les transmissions de `eth1` vers `eth2` sans restrictions et depuis `eth2` seulement si la connexion a été établie ailleurs.

```
iptables -A FORWARD -o eth2 -i eth1 -j ACCEPT
iptables -A FORWARD -i eth2 -o eth1 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

Rendre visibles les services de la DMZ depuis l'extérieur

Les services SSH et web correspondent à des requêtes sur des ports différents (respectivement 22 et 80). Il est donc possible de demander au routeur d'agir comme un serveur web et un serveur SSH en redirigeant les requêtes sur ces ports particuliers vers les machines qui offrent réellement ce service. C'est ce que l'on appelle une redirection de ports.

La commande suivante permet de rediriger les requêtes sur le port 80 du routeur vers le port 80 de la machine `m4`.

```
iptables -t nat -A PREROUTING -p tcp --dport 80 -i eth0 -j DNAT --to-destination 192.168.12.3:80
iptables -A FORWARD -i eth0 -o eth2 -p tcp --dport 80 -j ACCEPT
```

La commande suivante permet de rediriger les requêtes sur le port 22 du routeur vers le port 22 de la machine `m3`.

```
iptables -t nat -A PREROUTING -p tcp --dport 22 -i eth0 -j DNAT --to-destination 192.168.12.2:22
iptables -A FORWARD -i eth0 -o eth2 -p tcp --dport 22 -j ACCEPT
```

Il faut permettre aux services des machines de la DMZ de répondre aux requêtes.

```
iptables -A FORWARD -i eth2 -o eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

En vous connectant sur la machine `externe`, vérifier que le serveur web et le serveur SSH sont accessibles depuis l'adresse publique du routeur.

```
lynx 10.11.1.1
ssh 10.11.1.1
```