# ADMINISTRATION RÉSEAU

# IPTABLES ET NAT

A. Guermouche

# Plan

Logiciels de filtrage de paquets

Ipfwadm

Ipchains

Iptables

# Plan

Logiciels de filtrage de paquets

Ipfwadm

Ipchains

Iptables

# Logiciels de filtrage de paquets

- Fonctionnalités de “firewall” filtrant directement implémentée dans le noyau Linux.
- Filtrage de niveau 3 ou 4.
- 3 types de firewall filtrants :

  **ipfwadm.** Jusqu’à la version 2.1.102 du noyau linux

  **ipchains.** Entre les versions 2.2.0 et 2.4 du noyau linux

  **iptables.** À partir des noyaux 2.4

# Plan

Logiciels de filtrage de paquets

Ipfwadm

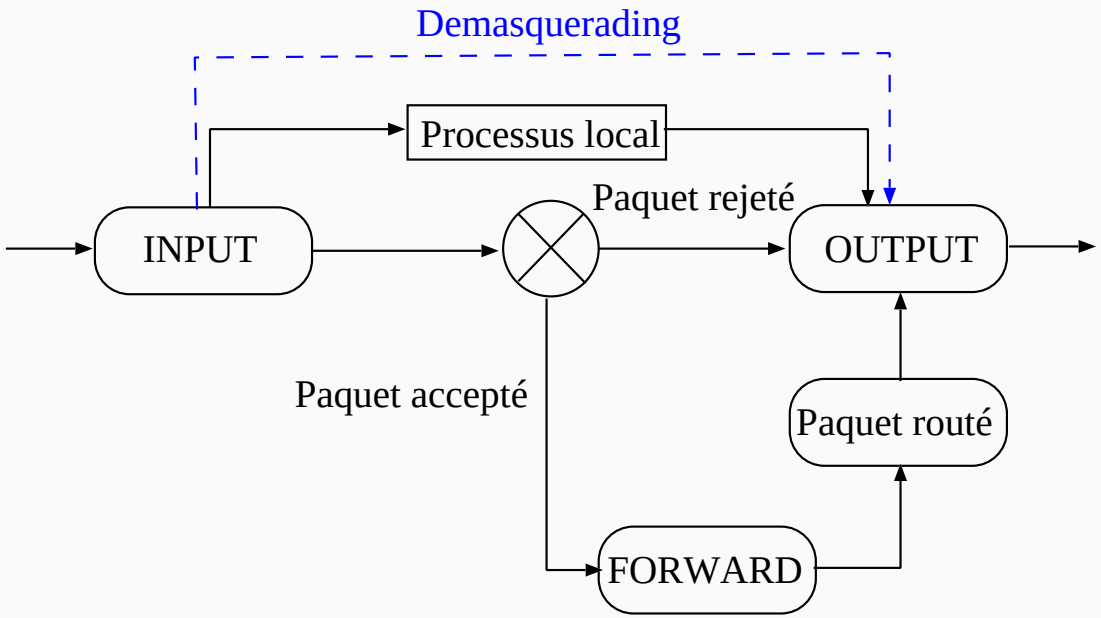
Ipchains

Iptables

# Ipfwadm

- Firewall permettant la gestion des paquets TCP, UDP et ICMP.
- 3 types de règles :
  - INPUT. sont appliquées lors de l'arrivée d'un paquet.
  - FORWARD. sont appliquées lorsque la destination du paquet n'est pas le routeur.
  - OUTPUT. sont appliquées dès qu'un paquet doit sortir du routeur.

Fonctionnement :



# Ipfwadm

- Firewall permettant la gestion des paquets TCP, UDP et ICMP.
- 3 types de règles :
  - INPUT. sont appliquées lors de l'arrivée d'un paquet.
  - FORWARD. sont appliquées lorsque la destination du paquet n'est pas le routeur.
  - OUTPUT. sont appliquées dès qu'un paquet doit sortir du routeur.

Fonctionnement :

1: lorsqu'un paquet entre, il traverse les règles de type INPUT
2: **Si** il est accepté **Alors**
3: **Si** il est destiné à une autre machine **Alors**
4: il est routé vers les règles FORWARD
5: **Sinon**
6: il est rejeté
7: le paquet est finalement émis

Dans tous les cas, le paquet traverse les règles OUTPUT

# Plan

Logiciels de filtrage de paquets

Ipfwadm

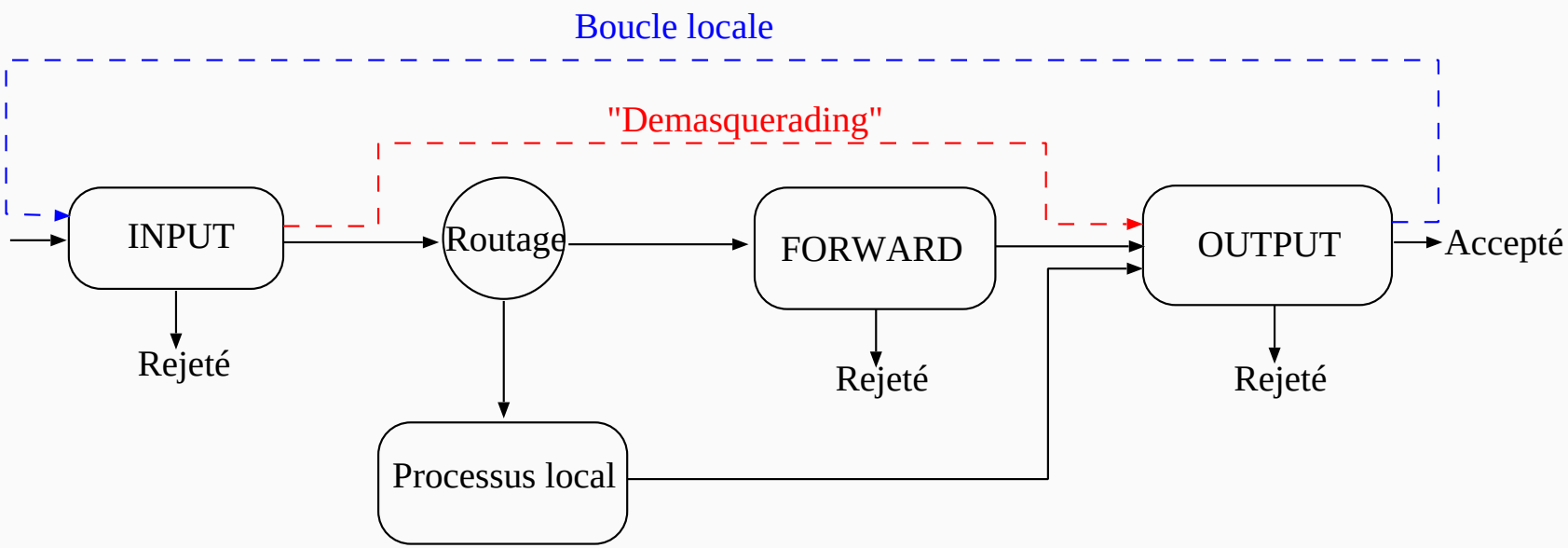
Ipchains

Iptables

# Ipchains

- Module du noyau Linux réalisant le filtrage de paquets.
- Inspiré du parre-feu BSD (tout comme ipfwadm)

Fonctionnement :



# Plan

Logiciels de filtrage de paquets

Ipfwadm

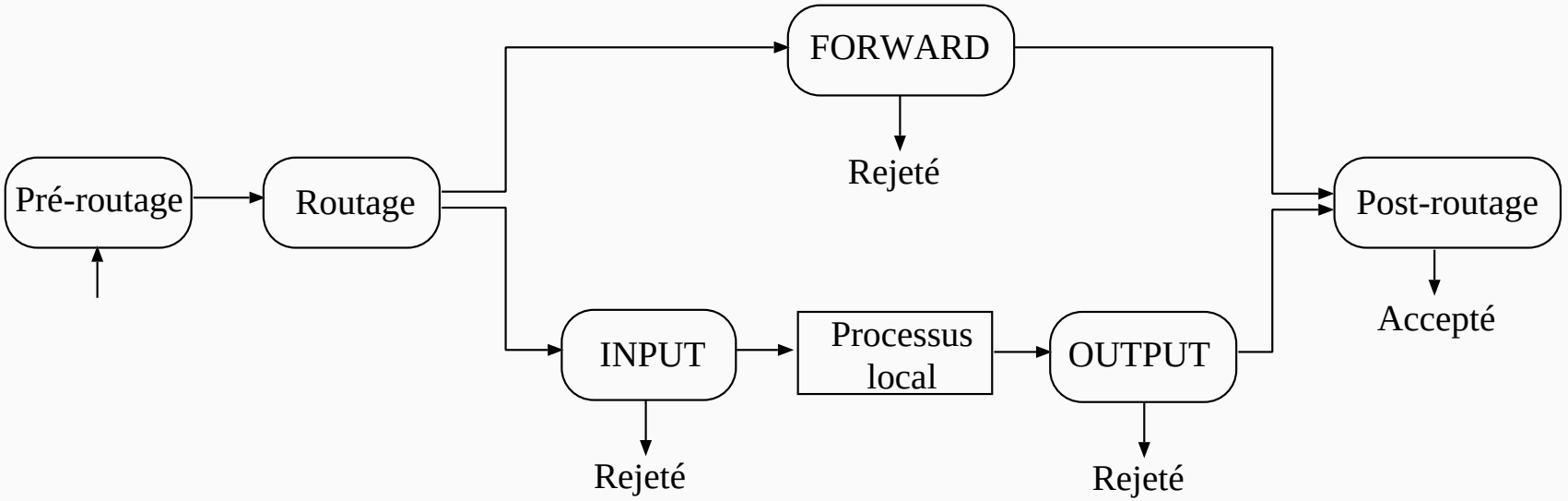
Ipchains

Iptables

# Iptables (1/2)

- Module du noyau Linux réalisant le filtrage de paquets (noyaux $\geq$ 2.4).
- Améliorations en matière de filtrage et de translation d'adresses par rapport à Ipchains.

Fonctionnement :



# Iptables (1/2)

- Module du noyau Linux réalisant le filtrage de paquets (noyaux $\geq$ 2.4).
- Améliorations en matière de filtrage et de translation d'adresses par rapport à Ipchains.

Fonctionnement :

**À l'arrivée d'un paquet (après décision de routage) :**

**Si** le paquet est destiné à l'hôte local **Alors**
    il traverse la chaîne INPUT.
    **Si** il n'est pas rejeté **Alors**
        il est transmis au processus impliqué.
**Sinon**
    **Si** le paquet est destiné à un hôte d'un autre réseau **Alors**
        il traverse la chaîne FORWARD
        **Si** il n'est pas rejeté **Alors**
            il poursuit alors sa route

# Iptables (1/2)

- Module du noyau Linux réalisant le filtrage de paquets (noyaux $\geq$ 2.4).
- Améliorations en matière de filtrage et de translation d'adresses par rapport à Ipchains.

Fonctionnement :

**À l'arrivée d'un paquet (après décision de routage) :**

**Si** le paquet est destiné à l'hôte local **Alors**
    il traverse la chaîne INPUT.
    **Si** il n'est pas rejeté **Alors**
        il est transmis au processus impliqué.
**Sinon**
    **Si** le paquet est destiné à un hôte d'un autre réseau **Alors**
        il traverse la chaîne FORWARD
        **Si** il n'est pas rejeté **Alors**
            il poursuit alors sa route

Tous les paquets émis par des processus locaux au routeur traversent la chaîne OUTPUT.

# Iptables (2/2)

Fonctionnalités :

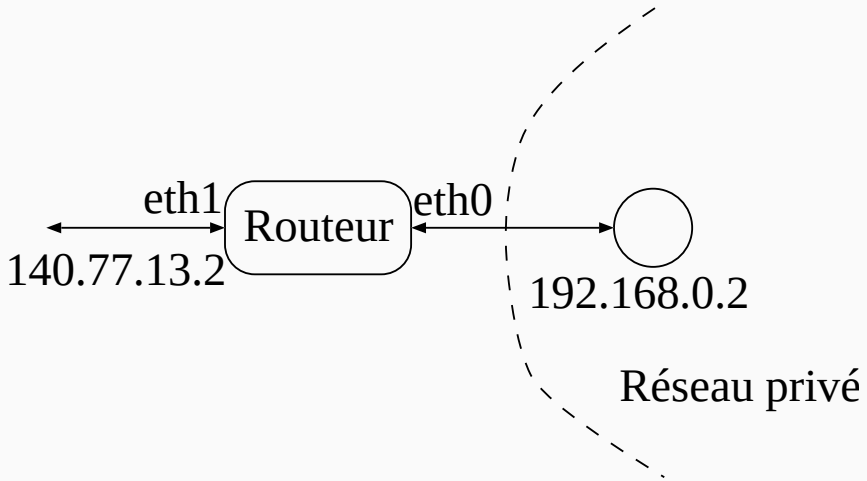
- Filtrage de paquets
- NAT
- Marquage de paquets

Architectures :Trois tables de chaînes (FILTER, NAT et MANGLE).

| FILTER (filtrage des paquets) | | NAT (translation d'adresses) | |
|---|---|---|---|
| INPUT | paquet entrant sur le routeur | PREROUTING | NAT de destination |
| OUTPUT | paquet émis par le routeur | POSTROUTING | NAT de source |
| FORWARD | paquet traversant le routeur | OUTPUT | NAT sur les paquets émis localement |

La table MANGLE sert au marquage des paquets

# Fonctionnalités NAT d'Iptables (1/2)



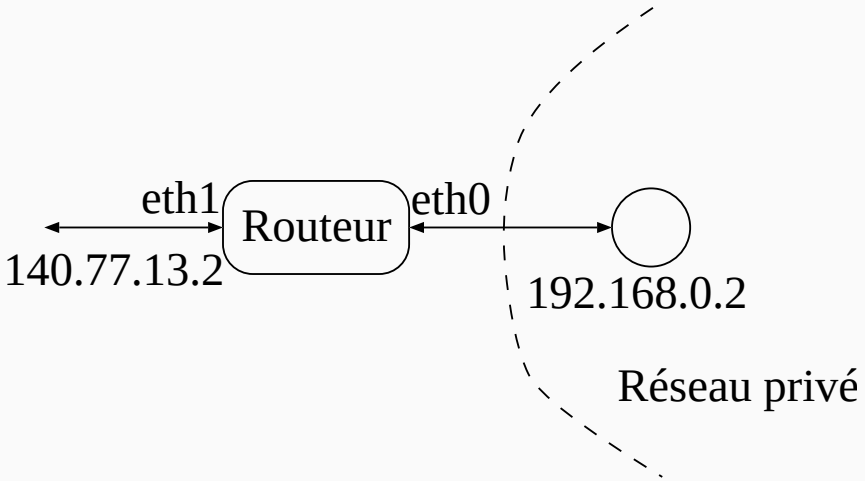

Modification de la destination du paquet avant le routage (paquet reçu de l'extérieur).

```
iptables -t nat -A PREROUTING -d 140.77.13.2 -i eth1 -j
DNAT --to-destination 192.168.0.2
```

Modification de la source du paquet après le routage (paquet émis à partir du réseau privé).

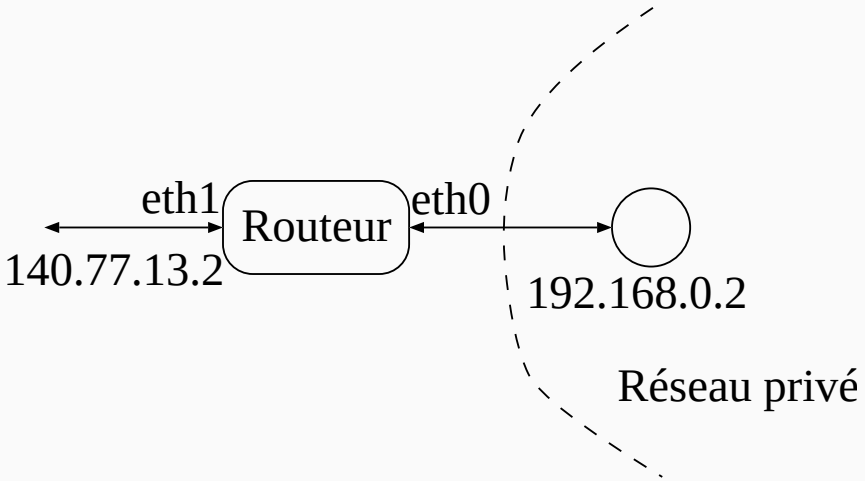
```
iptables -t nat -A POSTROUTING -s 192.168.0.2 -o eth1 -j
SNAT --to-source 140.77.13.2
```

# Fonctionnalités NAT d'Iptables (1/2)




Exercice : Comment faire pour que le routeur puisse envoyer un paquet à l'adresse 140.77.13.2?

# Fonctionnalités NAT d'Iptables (1/2)



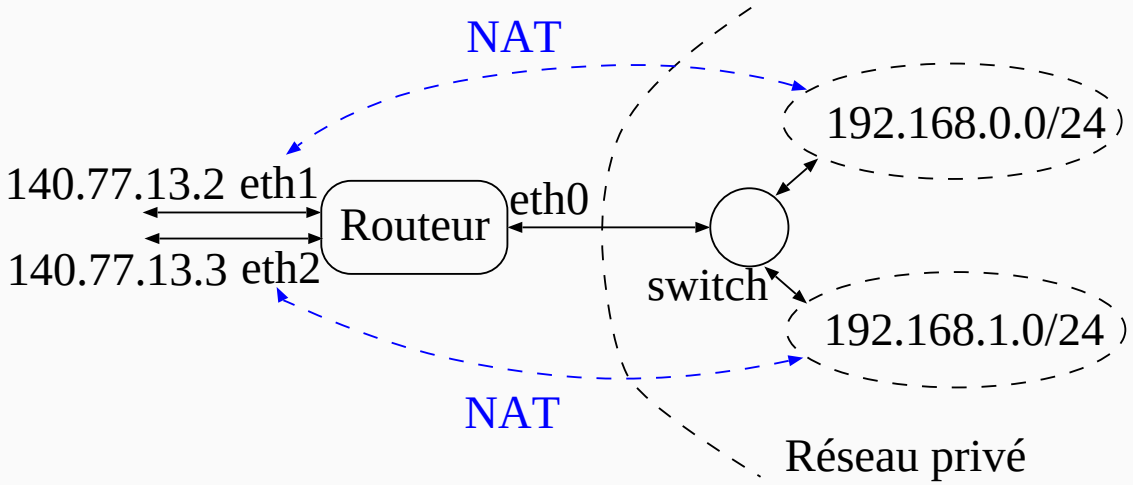

Exercice : Comment faire pour que le routeur puisse envoyer un paquet à l'adresse 140.77.13.2?

Réponse :

Il faut modifier la destination du paquet émis localement avant le routage.

```
iptables -t nat -A OUTPUT -d 140.77.13.2  -j
DNAT --to-destination 192.168.0.2
```

# Fonctionnalités NAT d'Iptables (2/2)



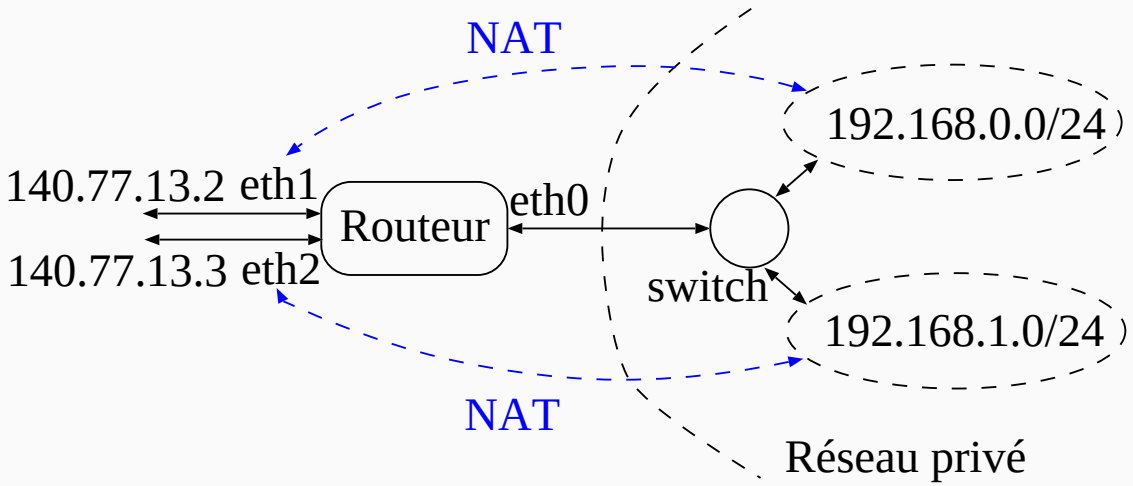

Association entre toutes les adresses privées du sous-réseau 192.168.0.0/24 avec l'interface eth1.

```
iptables -t nat -A POSTROUTING -o eth1 -s 192.168.0.0/24 -j MASQUERADE
```

Association entre toutes les adresses privées du sous-réseau 192.168.1.0/24 avec l'interface eth2.

```
iptables -t nat -A POSTROUTING -o eth2 -s 192.168.1.0/24 -j MASQUERADE
```

# Transfert de ports




Transférer les connexions sur le port 80 de l'adresse 140.77.13.2 sur la machine ayant l'adresse privée 192.168.0.200 sur le port 8080 :

```
iptables -t nat -A PREROUTING -p tcp -d 140.77.13.2
--dport 80 --sport 1024:65535 -j DNAT --to
192.168.0.200:8080
```