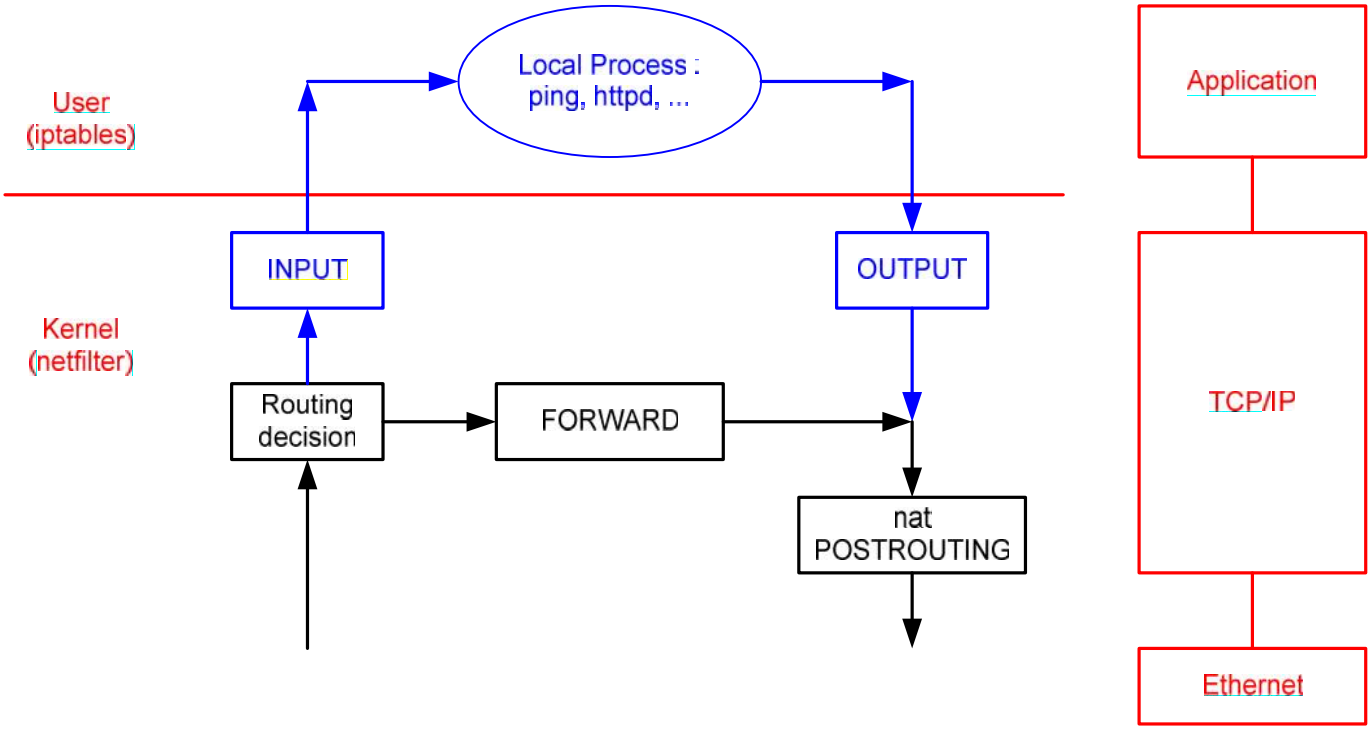
Laboratoire iptables (90 min)

0	Introduction	`sudo ./c 5 → CentOS6.4-CLI`

Préambule La figure ci-dessous illustre l'architecture des filtrages (INPUT, OUTPUT, FORWARD) présents dans le noyau Linux.
Elle comprend 2 parties distinctes : netfilter au niveau Kernel et iptables au niveau User.
Ce labo est complémentaire au précédent qui illustrait la partie **Routage** mise en noire dans la figure.



Objectifs Configurer et tester un firewall personnel sous Linux (partie mise en bleu)
Autoriser les flux ICMP, DNS et http

Cadre Ce labo s'effectue individuellement sur un PC Linux CentOS
Le §6 s'effectue par groupe de 2
Corrigé au format papier

Session Ouvrir une **session** : compte=**root** password=**rootroot**

1	Filtrage Linux en mode stateless	20 min

But 1.1 Mode par défaut du firewall Linux

Action `iptables -L`

Q1a Quelle est la configuration par défaut du firewall Linux ?

Remarques The default chain policy is ACCEPT.
Source = http://www.thegeekstuff.com/2011/06/iptables-rules-examples/

Voir figure ci-dessus pour situer les chaînes INPUT, FORWARD et OUTPUT

Action `ping 10.2.1.1 -c 1`

Q1b Le flux ICMP traverse-t-il ce firewall ?

But 1.2 Mettre la chaîne FORWARD en mode white list

Action
```
iptables -P FORWARD DROP
```

Test
```
iptables –L
ping 10.2.1.1 -c 1
```

Q1c Le flux ICMP traverse-t-il ce firewall ?

Remarques Ce filtre a été activé (**`net.ipv4.ip_forward = 1`**) dans le labo Routeur précédent.
Seuls les filtres INPUT et OUTPUT seront utilisés dans ce labo (voir partie mise en bleu dans la figure)

Conseil **Pensez à utiliser l'historique des commandes entrées avec les touches ↑ et ↓ pour éviter de retyper certaines commandes**

But 1.3 Mettre le firewall personnel en mode white list

Action
```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
```

Test
```
iptables -L
ping 10.2.1.1 -c 1
```

Q1d Le flux ICMP traverse-t-il ce firewall ?

Action **`CTRL c`** pour terminer la commande

But 1.4 Autoriser le flux ICMP

Action
```
iptables –A OUTPUT –p icmp –j ACCEPT
iptables –A INPUT –p icmp –j ACCEPT
```

Test
```
iptables –L
ping ...
```

Q1e Le flux ICMP traverse-t-il ce firewall ?

But 1.5 N'autoriser le flux ICMP qu'avec IP = 10.2.1.1

Action
```
iptables –D OUTPUT –p icmp –j ACCEPT
iptables –D INPUT –p icmp –j ACCEPT
```
Supprimer l'ancienne règle

```
iptables –A OUTPUT –d 10.2.1.1 –p icmp –j ACCEPT
iptables –A INPUT –s 10.2.1.1 –p icmp –j ACCEPT
```

Test
```
iptables –L
ping ...
```

Q1f Le flux ICMP traverse-t-il ce firewall ?

Q1g A quoi sert le filtre (la chaîne) INPUT ?

Q1h A quoi sert le filtre (la chaîne) OUTPUT ?

Q1i A quoi sert le filtre (la chaîne) FORWARD ?

2	tcpdump et iptables	10 min

But 2.1 Supprimer toutes les règles de la chaîne INPUT

Action
```
iptables –F INPUT
iptables –L
```
Vous devez obtenir le résultat suivant :

```
Chain INPUT (policy DROP)
target     prot opt source               destination

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     icmp --  anywhere             10.2.1.1
```

But 2.2 Observer avec tcpdump les paquets sortant et entrant :

Action Ouvrir un autre terminal : **<CTRL>+<ALT>+<F2>**
Dans terminal 2 : **`ifconfig`** pour connaître l'identifiant de la carte réseau
`tcpdump -i ethx icmp`
Dans terminal 1 : **`ping 10.2.1.1 -c 1`** puis **`CTRL c`** pour terminer

Q2a Répondre à l'aide de tcpdump, le paquet echo request est-il envoyé vers 10.2.1.1 ?

Q2b Répondre à l'aide de tcpdump, le paquet echo reply est-il reçu par votre PC ?

Q2c Répondre à l'aide de terminal 1, le test ping sur 10.2.1.1 est-il ok ?

Q2d Pourquoi tcpdump affiche les paquets echo request et reply alors que le test du ping n'est pas concluant ?

3	Loguer les paquets ICMP rentrant	10 min

But 3.1 Autoriser les paquets ICMP reçus de 10.2.1.1 à parvenir jusqu'au processus ping :

Action
```
iptables –A INPUT –s 10.2.1.1 –p icmp –j ACCEPT
iptables –L
ping 10.2.1.1 -c 1
```

But 3.2 Activer les logs en entrée pour les paquets ICMP :

Action
```
iptables –A INPUT –s 10.2.1.1 –p icmp –j LOG
iptables –L
ping 10.2.1.1 -c 1
dmesg | tail
```
afficher les dernières lignes de log du noyau

Q3a Avez-vous une trace du ping dans les logs du noyau ?

Q3b Pour quelle raison pensez-vous qu'il n'y a pas de trace du paquet icmp dans les logs ?

But 3.3 Corriger l'ordre des règles afin d'obtenir un log ICMP avec dmesg :

Q3c Quel est le bon ordre ?

Q3d Avez-vous une trace du ping dans les logs du noyau ?

4	Autoriser le protocole http	10 min

But 4.1 Ajouter ces 2 règles :

Action
```
iptables –A OUTPUT –p tcp --dport 80 –j ACCEPT
iptables –A INPUT –p tcp --sport 80 –j ACCEPT
```

But 4.2 Tester l'accès au serveur web sur 10.2.1.1 :

Action
```
wget 10.2.1.1
```

Q4a Est-ce que vous avez pu récupérer la page index.html ?

But 4.3 Tester l'accès au serveur web www.cern.ch :

Action
```
wget www.cern.ch
```

Q4b Avez-vous pu récupérer la page index.html ? Si non, pourquoi ?

Q4c Quelles sont les règles à ajouter ?

Q4d Avez-vous accès à www.cern.ch, à d'autres FQDN ?

5	Outil nmap	10 min

Introduction Cet outil teste l'accès à un serveur http, … en envoyant un paquet TCP:SYN sur chaque port de sa liste

Remarque Par défaut, nmap scanne 1000 ports jugés intéressants parmi les 65'536 possibles

But 5.1 Mettre le firewall personnel dans l'état par défaut (§1.1)

Action
```
iptables –F INPUT
iptables –P INPUT ACCEPT
iptables –F OUTPUT
iptables –P OUTPUT ACCEPT
```

`iptables –L` pour obtenir le résultat suivant :

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Q5a Quels sont les paquets envoyés avec **`nmap –p80 10.2.1.1`** ?
Utiliser tcpdump pour répondre

Tester l'accès au serveur web avec **`wget 10.2.1.1`**

Q5b Quels sont les paquets envoyés avec **`nmap –p79 10.2.1.1`** ?
Utiliser tcpdump pour répondre

`nmap 10.2.1.1` Lancer un scan avec les paramètres par défaut

Q5c Quels sont les ports ouverts ? Pourquoi ?

Q5d Combiens de ports sont fermés ?

Q5e Combiens de ports sont scannés par défaut ?

6	Comparer les modes stateless et stateful	20 min

Remarque Effectuer ces opérations par **groupe de 2.**
Choisir le PC de gauche comme outil de test utilisant nmap
Le PC de droite joue le rôle de cible à tester

But 6.1 Services réseau actifs sur le PC de droite ?
Action
```
service –-status-all | grep 'is running'
netstat –ltpn
```

Q6a Quels sont les services réseau actifs sur le PC de droite ?
Action
```
service –-status-all | grep 'is running'
```
```
[root@centos ~]# service --status-all | grep 'is running'
auditd (pid  929) is running...
crond (pid  1080) is running...
master (pid  1070) is running...
rsyslogd (pid  945) is running...
openssh-daemon (pid  994) is running...
```
```
netstat –ltpn
```
```
[root@centos ~]# netstat -lptn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             Stat
e       PID/Program name
tcp        0      0 0.0.0.0:22                  0.0.0.0:*                   LIST
EN      994/sshd
tcp        0      0 127.0.0.1:25                0.0.0.0:*                   LIST
EN      1070/master
tcp        0      0 :::22                       :::*                        LIST
EN      994/sshd
tcp        0      0 ::1:25                      :::*                        LIST
EN      1070/master
```

But 6.2 Sur le PC de droite, utiliser les règles http stateless du §4.1

Action
```
iptables -F INPUT
iptables –P INPUT DROP
iptables –A INPUT –p tcp --sport 80 –j ACCEPT
iptables -F OUTPUT
iptables –P OUTPUT DROP
iptables –A OUTPUT –p tcp --dport 80 –j ACCEPT
iptables –L
wget 10.2.1.1                          Tester l'accès au serveur web
```

But 6.3 Scanner les ports du PC de droite

Action **`nmap IP_PC_droite`** depuis le PC de gauche
Attendre environ 20 secondes pour la réponse

Q6b Quels ports sont accessible ?

But 6.4 Scanner les ports du PC de droite avec le port source = 80

Action **`nmap IP_PC_droite --source-port 80`**

Q6c Quel est le port ouvert ?

Q6d Pourquoi observez-vous une différence entre les deux scans précédents ?

But 6.5 Sur le PC de droite, utiliser les règles http stateful

Action

```
iptables -F INPUT
iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT
iptables -F OUTPUT
iptables –A OUTPUT –p tcp --dport 80 –j ACCEPT
wget 10.2.1.1                          Tester l'accès au serveur web
```

But 6.6 Répéter les actions du §6.4

Action `nmap IP_PC_droite --source-port 80`

Q6e Quels ports sont accessible ?

But 6.7 Répéter les actions du §6.3

Action `nmap IP_PC_droite`

Q6f Quels ports sont accessible ?

Q6g Expliquer les différences entre modes stateful et stateless ?

Q6h Que signifie l'état ESTABLISHED ?

Remarque La suite de l'étude des firewalls stateful aura lieu dans le cours Sécurité des Systèmes d'Information

En réserve

Liens

Tutoriel en français
http://doc.ubuntu-fr.org/iptables

25 Most Frequently Used Linux IPTables Rules Examples
http://www.thegeekstuff.com/2011/06/iptables-rules-examples/

Reduce firewall configuration complexity using iptables with chains
http://ruleant.blogspot.ch/2011/04/less-complex-firewall-configuration.html

Linux: 20 Iptables Examples For New SysAdmins
http://www.cyberciti.biz/tips/linux-iptables-examples.html

Fedora : Structure de la commande iptables
http://doc.fedora-fr.org/wiki/Parefeu_-_firewall_-_netfilter_-_iptables#Structure_de_la_commande_iptables

Labo terminé

Toutes les unités centrales seront éteintes avec un script
Tous les écrans seront éteints depuis le tableau électrique

iptables v1.4.7

```
Usage: iptables -[AD] chain rule-specification [options]
       iptables -I chain [rulenum] rule-specification [options]
       iptables -R chain rulenum rule-specification [options]
       iptables -D chain rulenum [options]
       iptables -[LS] [chain [rulenum]] [options]
       iptables -[FZ] [chain] [options]
       iptables -[NX] chain
       iptables -E old-chain-name new-chain-name
       iptables -P chain target [options]
       iptables -h (print this help information)

Commands:
Either long or short options are allowed.
  --append  -A chain             Append to chain
  --delete  -D chain             Delete matching rule from chain
  --delete  -D chain rulenum
                                 Delete rule rulenum (1 = first) from chain
  --insert  -I chain [rulenum]
                                 Insert in chain as rulenum (default 1=first)
  --replace -R chain rulenum
                                 Replace rule rulenum (1 = first) in chain
  --list    -L [chain [rulenum]]
                                 List the rules in a chain or all chains
  --list-rules -S [chain [rulenum]]
                                 Print the rules in a chain or all chains
  --flush   -F [chain]           Delete all rules in  chain or all chains
  --zero    -Z [chain [rulenum]]
                                 Zero counters in chain or all chains
  --new     -N chain             Create a new user-defined chain
  --delete-chain
            -X [chain]           Delete a user-defined chain
  --policy  -P chain target
                                 Change policy on chain to target
  --rename-chain
            -E old-chain new-chain
                                 Change chain name, (moving any references)
Options:
[!] --proto     -p proto         protocol: by number or name, eg. `tcp'
[!] --source    -s address[/mask][...]
                                 source specification
[!] --destination -d address[/mask][...]
                                 destination specification
[!] --in-interface -i input name[+]
                                 network interface name ([+] for wildcard)
 --jump    -j target
                                 target for rule (may load target extension)
  --goto      -g chain
                                 jump to chain with no return
  --match  -m match
                                 extended match (may load extension)
  --numeric     -n               numeric output of addresses and ports
[!] --out-interface -o output name[+]
                                 network interface name ([+] for wildcard)
  --table  -t table              table to manipulate (default: `filter')
  --verbose     -v               verbose mode
  --line-numbers                 print line numbers when listing
  --exact  -x                    expand numbers (display exact values)
[!] --fragment   -f              match second or further fragments only
  --modprobe=<command>           try to insert modules using this command
  --set-counters PKTS BYTES      set the counter during insert/append
[!] --version   -V               print package version.
```