

# Guide COMPLET de iptables -t filter -A

Je vais t'expliquer en détail toutes les possibilités avec iptables -t filter -A. C'est la partie la plus utilisée d'iptables !

---

## STRUCTURE DE BASE

```
iptables -t filter -A CHAÎNE [CRITÈRES] -j ACTION
```

- -t filter = table FILTER (pare-feu) → optionnelle car c'est la table par défaut
  - -A = Append (ajouter à la fin)
  - CHAÎNE = INPUT, OUTPUT ou FORWARD
  - CRITÈRES = conditions pour appliquer la règle
  - -j = Jump (action à faire)
  - ACTION = ACCEPT, DROP, REJECT, etc.
- 

## Les 3 CHAÎNES principales de FILTER

### 1. CHAÎNE INPUT → Paquets POUR la machine

*# Exemple : Quelqu'un essaie de se connecter à TON serveur SSH*

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

*# "Accepte les connexions TCP sur le port 22 (SSH) qui viennent vers MOI"*

### 2. CHAÎNE OUTPUT → Paquets DE la machine

*# Exemple : Tu essaies d'aller sur un site web*

```
iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT
```

*# "Accepte les connexions TCP sortantes vers le port 80 (HTTP)"*

### 3. CHAÎNE FORWARD → Paquets qui TRAVERSENT la machine

```
# Exemple : Ta machine fait routeur entre 2 réseaux
iptables -A FORWARD -s 192.168.1.0/24 -d 192.168.2.0/24 -j ACCEPT

# "Accepte les paquets qui viennent de 192.168.1.0/24 vers 192.168.2.0/24"
```

## TOUTES les OPTIONS possibles (CRITÈRES)

### A. Par ADRESSE IP

Option	Signification	Exemple
-s ADRESSE	Source (qui envoie)	-s 192.168.1.100
-d ADRESSE	Destination (pour qui)	-d 10.0.0.1
!	SAUF (négation)	-s ! 192.168.1.100

Exemples :

```
# Bloquer une IP spécifique
iptables -A INPUT -s 192.168.1.100 -j DROP

# Autoriser seulement un réseau
iptables -A INPUT -s 192.168.1.0/24 -j ACCEPT
iptables -A INPUT -j DROP # Tout le reste bloqué

# Sauf mon serveur DNS
iptables -A INPUT -s ! 8.8.8.8 -p udp --dport 53 -j DROP
```

---

## B. Par INTERFACE

Option	Signification	Exemple
<code>-i INTERFACE</code>	Interface d'entrée	<code>-i eth0</code>
<code>-o INTERFACE</code>	Interface de sortie	<code>-o wlan0</code>

Exemples :

```
# Seulement sur l'interface WiFi
iptables -A INPUT -i wlan0 -p tcp --dport 22 -j ACCEPT
```

```
# Bloquer tout ce qui entre par eth1
iptables -A INPUT -i eth1 -j DROP
```

```
# Routeur : autoriser eth0-eth1
iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT
```

---

## C. Par PROTOCOLE

Option	Signification	Exemple
<code>-p PROTO</code>	Protocol	<code>-p tcp, -p udp, -p icmp</code>
<code>-p all</code>	Tous les protocoles	<code>-p all</code>

Pour TCP :

```
# Autoriser SSH (port 22)
```

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

```
# Autoriser HTTP (port 80) et HTTPS (443)
```

```
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

```
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

```
# Ports sources (quand tu envoies)
```

```
iptables -A OUTPUT -p tcp --sport 1024:65535 -j ACCEPT
```

## **Pour UDP :**

```
# Autoriser DNS (port 53)
```

```
iptables -A INPUT -p udp --dport 53 -j ACCEPT
```

```
# Autoriser DHCP (port 67-68)
```

```
iptables -A INPUT -p udp --dport 67:68 -j ACCEPT
```

## **Pour ICMP (ping) :**

```
bash
```

```
# Autoriser le ping entrant
```

```
iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
```

```
# Autoriser les réponses ping
```

```
iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
```

```
# Bloquer tous les ICMP
```

```
iptables -A INPUT -p icmp -j DROP
```

---

## **D. Par ÉTAT de connexion (IMPORTANT !)**

Option	Signification
NEW	Nouvelle connexion
ESTABLISHED	Connexion déjà établie
RELATED	Connexion liée (ex: FTP data)
INVALID	Paquet invalide

EXEMPLE CRUCIAL pour ton TP :

```
# TOUJOURS mettre en PREMIER !
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT

# "Autorise les réponses aux connexions que J'AI initiées"
```

Pourquoi c'est important ?

Sans cette règle :

- Tu pingues 8.8.8.8 → paquet sort (NEW)
- 8.8.8.8 répond → paquet entre (ESTABLISHED) → BLOQUÉ si pas de règle !
- Résultat : le ping ne marche que dans un sens

## E. Par PORT (TCP/UDP)

Option	Signification	Exemple
--sport PORT	Source port	--sport 1024:65535
--dport PORT	Destination port	--dport 80

<code>--sport MIN:MAX</code>	Range de ports	<code>--sport 1000:2000</code>
<code>--multiport --sports</code>	Multiples ports source	<code>--multiport --sports 22,80,443</code>
<code>--multiport --dports</code>	Multiples ports destination	<code>--multiport --dports 21,22,23</code>

Exemples :

```
# Un seul port
iptables -A INPUT -p tcp --dport 22 -j ACCEPT # SSH

# Range de ports
iptables -A INPUT -p tcp --dport 1024:65535 -j ACCEPT # Ports éphémères

# Multiples ports
iptables -A INPUT -p tcp -m multiport --dports 80,443,8080 -j ACCEPT

# Ports source (quand tu es client)
iptables -A OUTPUT -p tcp --sport 32768:60999 -j ACCEPT
```

## F. Options avancées

Option	Signification	Exemple
<code>-m limit --limit</code>	Limiter le débit	<code>--limit 3/sec</code>
<code>-m mac --mac-source</code>	Par adresse MAC	<code>--mac-source 00:11:22:33:44:55</code>

<code>-m tcp --tcp-flags</code>	Flags TCP	<code>--tcp-flags SYN,ACK,FIN,RST SYN</code>
<code>-m time</code>	Par heure/jour	<code>--timestart 09:00 --timestop 17:00</code>

Exemples :

```
# Anti-DDoS sur ping
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j
ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-request -j DROP

# Par MAC address
iptables -A INPUT -m mac --mac-source 00:11:22:33:44:55 -j ACCEPT

# Seulement le jour
iptables -A INPUT -p tcp --dport 22 -m time --timestart 09:00 --timestop 17:00
-j ACCEPT
```

## ACTIONS possibles (-j)

Action	Signification	Exemple
<code>ACCEPT</code>	Accepter le paquet	<code>-j ACCEPT</code>
<code>DROP</code>	Supprimer silencieusement	<code>-j DROP</code>
<code>REJECT</code>	Rejeter avec erreur	<code>-j REJECT</code>

---

LOG

Logger (journaliser)

-j LOG

---

RETURN

Retourner à la chaîne parente

-j RETURN

---

Différence DROP vs REJECT :

- DROP = paquet jeté, pas de réponse → timeout
- REJECT = envoi "Connection refused" → erreur immédiate

*# Exemple LOG*

```
iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "SSH attempt: "
```

```
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

---

## EXEMPLES COMPLETS pour ton TP

### Exemple 1 : Pare-feu personnel

*# Politique par défaut DROP*

```
iptables -P INPUT DROP
```

```
iptables -P FORWARD DROP
```

```
iptables -P OUTPUT ACCEPT
```

*# Localhost toujours autorisé*

```
iptables -A INPUT -i lo -j ACCEPT
```

*# Réponses aux connexions établies*

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

*# SSH depuis réseau local seulement*

```
iptables -A INPUT -s 192.168.1.0/24 -p tcp --dport 22 -j ACCEPT
```

*# HTTP/HTTPS*

```
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

```
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```



```
# Ping limité
```

```
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j  
ACCEPT
```

## Exemple 2 : Routeur (comme dans ton TP)

```
bash
```

```
# Activer le forwarding
```

```
sysctl -w net.ipv4.ip_forward=1
```

```
# --- RÈGLES FORWARD ---
```

```
# 1. TOUJOURS en premier : réponses
```

```
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# 2. S1 → S2 autorisé
```

```
iptables -A FORWARD -s 172.16.1.0/24 -d 172.16.2.0/24 -j ACCEPT
```

```
# 3. S2 → S1 autorisé
```

```
iptables -A FORWARD -s 172.16.2.0/24 -d 172.16.1.0/24 -j ACCEPT
```

```
# 4. S1 → S3 bloqué
```

```
iptables -A FORWARD -s 172.16.1.0/24 -d 172.16.3.0/24 -j DROP
```

```
# 5. S3 → S1 bloqué
```

```
iptables -A FORWARD -s 172.16.3.0/24 -d 172.16.1.0/24 -j DROP
```

```
# 6. Politique par défaut
```

```
iptables -P FORWARD DROP
```

## Exemple 3 : Serveur web sécurisé

```
# Vider toutes les règles
```

```
iptables -F
```

```
iptables -X
```

```
# Politiques par défaut
```

```
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Localhost
iptables -A INPUT -i lo -j ACCEPT

# Réponses
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# SSH avec limite
iptables -A INPUT -p tcp --dport 22 -m limit --limit 3/min -j ACCEPT

# HTTP/HTTPS
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# DNS
iptables -A INPUT -p udp --dport 53 -j ACCEPT

# Ping limité
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j
ACCEPT

# Protection contre les scans
iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP

iptables -A INPUT -p tcp --tcp-flags ALL ALL -j DROP
```

---

## ORDRE DES RÈGLES (TRÈS IMPORTANT !)

Les règles sont vérifiées dans l'ordre ! La première qui correspond s'applique.

### MAUVAIS ordre :

```
iptables -A INPUT -j DROP # 1. Tout bloqué !

iptables -A INPUT -p tcp --dport 22 -j ACCEPT # 2. Jamais atteint
```

## BON ordre :

```
iptables -A INPUT -i lo -j ACCEPT          # 1. Localhost
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT # 2.
Réponses
iptables -A INPUT -p tcp --dport 22 -j ACCEPT # 3. SSH
iptables -A INPUT -j DROP                  # 4. Tout le reste bloqué
```

### Règle d'or :

1. localhost (lo)
  2. ESTABLISHED,RELATED
  3. Règles spécifiques (ACCEPT)
  4. Règles de blocage spécifiques
  5. Politique par défaut (DROP)
- 

## COMMANDES UTILES pour tester

```
# Voir toutes les règles
iptables -L -v -n

# Voir une chaîne spécifique
iptables -L INPUT -v -n

# Voir avec numéros de ligne
iptables -L --line-numbers

# Supprimer une règle (ex: ligne 3 de INPUT)
iptables -D INPUT 3

# Vider toutes les règles
iptables -F

# Sauvegarder
iptables-save > /etc/iptables.rules

# Restaurer
```

```
iptables-restore < /etc/iptables.rules
```

---

Une règle iptables =

```
iptables -t filter -A CHAINE
```

↓

CRITÈRES :

```
-s 192.168.1.0/24    (source)
-d 10.0.0.1          (destination)
-p tcp               (protocole)
--dport 80           (port)
-i eth0              (interface)
-m state --state NEW (état)
```

↓

```
-j ACTION
```

ACCEPT / DROP / REJECT / LOG