iptables

Очистка правил в таблице INPUT

```
iptables -F INPUT
```

Обнулить счетчики

```
iptables -Z INPUT
```

Посмотреть список правил в таблице (v -)

```
iptables -t nat -L -v
```

Создать цепочки пользователя

```
iptables -N tcp_filter
```

```
iptables -N udp_filter
```

```
iptables -N icmp_filter
```

Удалить цепочку

```
iptables -X tcp_filter
```

Установить политику по умолчанию

```
iptables -P INPUT DROP
```

```
iptables -P OUTPUT ACCEPT
```

Посмотреть пронумерованный список правил

```
iptables -t nat -L -v --line-numbers
```

Правило для NAT

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

Правило для MAC

```
iptables -A INPUT -m mac --mac-source 00:12:23:ad:bb:2d -j ACCEPT
```

Правило для

Правило для

Block the most common attacks

```
# iptables -F
```

First, we start with blocking null packets

```
# iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP
```

We told the firewall to take all incoming packets with tcp flags NONE and just DROP them. The next pattern to reject is a syn-flood attack

```
# iptables -A INPUT -p tcp ! --syn -m state --state NEW -j DROP
```

Syn-flood attack means that the attackers open a new connection, but do not state what they want (ie. SYN, ACK, whatever). They just want to take up our servers' resources. We won't accept such packages. Now we move on to one more common pattern: XMAS packets, also a recon packet.

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

Open up ports for selected services

Now we can allow web server traffic:

```
# iptables -A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
# iptables -A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
```

Now, let's allow users use our SMTP servers:

```
# iptables -A INPUT -p tcp -m tcp --dport 25 -j ACCEPT
# iptables -A INPUT -p tcp -m tcp --dport 465 -j ACCEPT
```

POP3 traffic:

```
# iptables -A INPUT -p tcp -m tcp --dport 110 -j ACCEPT
```

```
# iptables -A INPUT -p tcp -m tcp --dport 995 -j ACCEPT
```

Now we also need to allow IMAP mail protocol:

```
# iptables -A INPUT -p tcp -m tcp --dport 143 -j ACCEPT
# iptables -A INPUT -p tcp -m tcp --dport 993 -j ACCEPT
```

Limiting SSH access

We should also allow SSH traffic, so we can connect to the VPS remotely.
The simple way to do it would be with this command:

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

```
[root@cen752 svm]# w
 10:04:14 up 2 days, 20:42, 5 users, load average: 0,09, 0,07, 0,05
USER   TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
root   tty1                     Птн16    2days                -bash
svm      pts/0    gateway          Птн17     2days  sshd: svm [priv]
svm              pts/1          192.168.113.11   Птн16  sshd: svm [priv]
```

Now, you can create the firewall rule to only allow traffic to SSH port if it comes from one source: your IP address:

```
# iptables -A INPUT -p tcp -s YOUR_IP_ADDRESS -m tcp --dport 22 -j ACCEPT
```

Replace YOUR_IP_ADDRESS with the actuall IP, of course.

Right now, we need to add one more rule that will allow us to use outgoing connections
(ie. ping from VPS or run software updates);

```
# iptables -I INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

It will allow any established outgoing connections to receive

replies from the VPS on the other side of that connection. When we have it all set up, we will block everything else, and allow all outgoing connections.

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

*****

Предположим надо заблокировать ip-шник 123.123.123.123, тогда делаем это так:

```
iptables -A INPUT -s 123.123.123.123 -j DROP
```