iptables 简单配置

安装与配置

CentOS 7.0 默认使用的是 firewall 作为防火墙
查看防火墙状态
```
firewall-cmd --state
```
停止 firewall
```
systemctl stop firewalld.service
```
禁止 firewall 开机启动
```
systemctl disable firewalld.service
```

安装
先检查是否安装了 iptables
```
service iptables status
```
#安装 iptables
```
yum install -y iptables
yum install iptables-services
```

注册 iptables 服务
#相当于以前的 chkconfig iptables on
```
systemctl enable iptables.service
```
#开启服务
```
systemctl start iptables.service
```
#查看状态
```
systemctl status iptables.service
```

将 iptables 加入服务
```
ln -s /usr/lib/systemd/system/iptables.service /etc/systemd/system/multi-user.target.wants
```

解决 vsftpd 在 iptables 开启后,无法使用被动模式的问题
1.首先在/etc/sysconfig/iptables-config 中修改或者添加以下内容
#添加以下内容,注意顺序不能调换
IPTABLES_MODULES="ip_conntrack_ftp"
IPTABLES_MODULES="ip_nat_ftp"
2.重新设置 iptables 设置
```
iptables -A INPUT -m state --state  RELATED,ESTABLISHED -j ACCEPT
```

iptables 规则

#保存规则
`service iptables save`（规则会保存在/etc/sysconfig/iptables 中，重启也自动生效）

查看 -t 表 -L 链（-t table 默认是 filter）
`iptables -t filter -L INPUT`
`iptables -t table 命令 chain rules -j target`
链的默认策略是 ACCEPT（允许）,可以修改策略
修改默认策略：
`iptables -t filter --policy FORWARD DROP`
或者：`iptables -t filter -P FORWARD DROP`
查看一下：`iptables -L -t filter`
Chain FORWARD (policy DROP)
可以看到 filter 表的 FORWARD 链默认策略变成 DROP
同样的策略上边的优先级高，所以冲突的策略上边的生效。
参数介绍
等价参数：
-P = --policy 定义默认策略
-A = --append 在规则列表的最后增加一条规则
-I = --insert 在指定的位置插入一条规则（如果不指定位置，默认在最上边插入，如果在第二个位置插入-I INPUT 2）
eg.:`iptables -t filter -I INPUT -p icmp -j DROP`（在 filter 表 INPUT 链插入一条规则，没指定位置）
-D = --delete 删除一条规则
eg.:`iptables -t filter -D INPUT 1`（删除第一条规则）
-R = --replace 替换规则表的某个规则
eg.:`iptables -t filter -R INPUT 2 -p icmp -j DROP`（替换第二条规则）
-F = --flush 清除表中所有规则，可以指定删除某个表或某个链上的所有规则
eg.:`iptables -t filter -F`（清除 filter 表中所有规则）
eg.:`iptables -t filter -F INPUT`（清除 filter 表 INPUT 链上的所有规则）

自定义链：
创建自定义链（-N = new）
`iptables -t filter -N self_control`
增加自定义链规则
`iptables -t filter -I self_control -s 192.151.102.2 -j REJECT`
引用自定义链
`iptables -t filter -I INPUT -j self_control`
删除自定义链
清空自定义链规则：`iptables -t filter -F self_ctl`
删除链引用规则：`iptables -t filter -D INPUT 1`
删除自定义链：`iptables -X self_ctl`
重命名自定义链
`iptables -E self_control self_ctl`
(-E = --rename-chain)

iptables 匹配选项

-i 或--in-interface 指定数据包从哪个网络接口进入，如 eth0、ppp0 等。
-o 或--out-interface 指定数据包从哪块网络接口输出，如 eth0、ppp0 等。
-p 或--protocol 协议类型，指定数据包匹配的协议，如 TCP、UDP 和 ICMP 等。
-s 或--source 指定数据包匹配的原地址。
-d 或--destination 指定数据包匹配的目的地址。
--sport 指定数据包匹配的源端口号，可以使用“起始端口号:结束端口号”的格式指定范围。
--dport 目标端口号，指定数据包匹配的目标端口号，可以使用“起始端口号:结束端口号”的格式指定范围。

NAT
nat 常见动作：
SNAT 源地址目标转换、DNAT 目标网络地址转换、MASQUERADE 改写数据包来源 IP 为防火墙 NIC IP、REDIRECT 将包重新导向到另一个端口

SNAT 改写封包来源 IP 为某特定 IP 或 IP 范围，可以指定 port 对应的范围，进行完此处理动作后，将直接跳往下一个规则（mangleostrouting）。
eg.: `iptables -t nat -A POSTROUTING -p tcp -o eth0 -j SNAT --to-source 194.236.50.155-194.236.50.160:1024-32000`
如下命令表示把所有 10.8.0.0 网段的数据包 SNAT 成 192.168.5.3 的 ip 然后发出去
`iptables -t nat -A POSTROUTING -s 10.8.0.0/255.255.255.0 -o eth0 -j SNAT --to-source 192.168.5.3`

DNAT 改写封包目的地 IP 为某特定 IP 或 IP 范围，可以指定 port 对应的范围，进行完此处理动作后，将会直接跳往下一个规炼（filter:input 或 filter:forward）。
eg.: `iptables -t nat -A PREROUTING -p tcp -d 15.45.23.67 --dport 80 -j DNAT --to-destination 192.168.1.1-192.168.1.10:80-100`

REDIRECT 将包重新导向到另一个端口（PNAT），进行完此处理动作后，将会继续比对其它规则。 这个功能可以用来实作通透式 porxy 或用来保护 web 服务器。
eg.: `iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-ports 8080`

MASQUERADE 是用发送数据的网卡上的 IP 来替换源 IP，因此，对于那些 IP 不固定的场合，比如拨号网络或者通过 dhcp 分配 IP 的情况下，就得用 MASQUERADE。
eg.: `iptables -t nat -A POSTROUTING -p TCP -j MASQUERADE --to-ports 1024-31000`

PREROUTING 是目的地址转换（DNAT），要把别人的公网 IP 换成你们内部的 IP，才让访问到你们内部受防火墙保护的服务器。
`eg:iptables -t nat -A PREROUTING -i eth0 -j MASQUERADE`

POSTROUTING 是源地址转换（SNAT），要把你内部网络上受防火墙保护的 ip 地址转换成你本地的公网地址才能让它们上网。

```
eg:iptables -t nat -A POSTROUTING -o wlan0 -j MASQUERADE
```

WEB 服务器常用配置

```
iptables -L -n
#先允许所有,不然有可能会杯具
iptables -P INPUT ACCEPT
#清空所有默认规则
iptables -F
#清空所有自定义规则
iptables -X
#所有计数器归 0
iptables -Z
#允许来自于 lo 接口的数据包(本地访问)
iptables -A INPUT -i lo -j ACCEPT
#开放 22 端口
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
#开放 21 端口(FTP)
iptables -A INPUT -p tcp --dport 21 -j ACCEPT
#开放 80 端口(HTTP)
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
#开放 443 端口(HTTPS)
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
#允许 ICMP
iptables -A INPUT -p icmp --icmp-type 8 -j ACCEPT
#允许接受本机请求之后的返回数据 RELATED,是为 FTP 设置的
iptables -A INPUT -m state --state  RELATED,ESTABLISHED -j ACCEPT
#其他入站一律丢弃
iptables -P INPUT DROP
#所有出站一律绿灯
iptables -P OUTPUT ACCEPT
#所有转发一律丢弃
iptables -P FORWARD DROP
```

端口映射：

```
iptables -t nat -A PREROUTING -i eth3 -d 192.168.20.10/32 -p tcp --dp
ort 3389 -j DNAT --to 192.168.10.110:3389
```

```
#如果要添加内网 ip 信任（接受其所有 TCP 请求）
iptables -A INPUT -p tcp -s 45.96.174.68 -j ACCEPT
#要封停一个 IP，使用下面这条命令：
```

```
iptables -I INPUT -s ***.***.***.*** -j DROP
```
#要解封一个 IP，使用下面这条命令:
```
iptables -D INPUT -s ***.***.***.*** -j DROP
```