## **iptables基本管理**

**关闭firewalld，启动iptables服务**

**[**root@proxy **~]**# systemctl stop firewalld**.**service

**[**root@proxy **~]**# systemctl disable firewalld**.**service

在linux5/6的时候一直都是iptables来进行防火墙设置的，redhat7出来的firewall，也是基于iptables出来的，在redhat上执行iptables也是ok的。

安装iptables软件，启动。

Yum -y install iptables-services

systemctl start iptables**.**service

**iptables框架**

1）iptables默认有4个表，nat表（地址转换表）、filter表（数据过滤表）、raw表（状态跟踪表）、mangle表（包标记表）。

2）iptables的5个链（区分大小写）：

INPUT链（入站规则）

OUTPUT链（出站规则）

FORWARD链（转发规则）

PREROUTING链（路有前规则）

POSTROUTING链（路由后规则）

iptabels语法格式

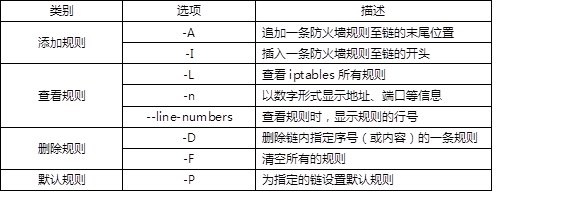
iptables **[-**t 表名**]** 选项 **[**链名**]** **[**条件**]** **[-**j 目标操作**]**

如：

iptables **-**t filter **-**I INPUT **-**p icmp **-**j ACCEPT （允许所有人对这台服务器进行icmp访问）

1. //注意事项与规律：
2. //可以不指定表，默认为filter表
3. //可以不指定链，默认为对应表的所有链
4. //除非设置默认策略，否则必须指定匹配条件
5. //选项/链名/目标操作用大写字母，其余都小写
6. //目标操作：
7. // ACCEPT：允许通过/放行
8. // DROP：直接丢弃，不给出任何回应
9. // REJECT：拒绝通过，必要时会给出提示
10. // LOG：记录日志，然后传给下一条规则

iptables命令的常用选项如表-1所示。



Iptables -t filter -A INPUT -p tcp -j ACCEPT (在末尾追加一条防火墙规则)INPUT链

Iptables -t filter -I INPUT -p tcp -j ACCEPT（在行首插入一条防火墙规则）INPUT链

Iptables -t filter -I INPUT 2 -p tcp -j ACCEPT（在第二行插入一条防火墙规则）INPUT链

Iptables -nL 查看防火墙中的所有规则（默认是查看filter表中的）

iptables -nL INPUT 查看filter表中INPUT链中的规则

iptables -t nat -nL INPUT 查看 nat 表中 INPUT 链的规则

iptables -t nat 查看 nat 表中的规则

iptables -nL --line-numbers 查看规则，显示行号

Iptables -D INPUT 3 删除INPUT 中第三行规则

Iptables -F 删除所有规则 （默认是filter）

Iptables -t nat -F 删除nat表中的所有规则

Iptables -t filter -P INPUT DROP 设置默认规则

## **filter过滤和转发控制**

**iptables防火墙规则的条件**



iptables **-**I INPUT **-**p tcp **--**dport **80** **-**j REJECT (所有访问80端口的都拒绝)

iptables **-**I INPUT **-**s **192.168.2.100** **-**j REJECT （从192.168.2.100过来的ip被拒绝）

iptables **-**I INPUT **-**d **192.168.2.5** **-**p tcp **--**dport **80** **-**j REJECT（访问192.168.2.5的80端口的被拒绝）

iptables **-**I INPUT **-**i eth0 **-**p tcp **--**dport **80** **-**j REJECT （访问eth0上80端口的被拒绝）

iptables **-**A INPUT **-**s **114.212.33.12** **-**p tcp **--**dport **22-**j REJECT（114.212.33.12访问22端口被拒绝）

**开启Linux的路由转发功能**

**[**root@proxy **~]**# echo **0** **>** /proc/sys**/**net**/**ipv4**/**ip\_forward            //关闭路由转发

**[**root@proxy **~]**# echo **1** **>** /proc/sys**/**net**/**ipv4**/**ip\_forward            //开启路由转发

//注意以上操作仅当前有效，计算机重启后无效

**[**root@proxy **~]**# echo 'net.ipv4.ip\_forward=1' **>>** /etc/sysctl**.**conf

//修改/etc/sysctl.conf配置文件，可以实现永久有效规则

设置proxy主机的防火墙规则，保护防火墙后面的Web服务器

iptables **-**I FORWARD **-**s **192.168.4.100** **-**p tcp **--**dport **80** **-**j DROP（当主机192.168.4.100访问后面web服务器80端口的时候被丢弃）

**禁ping的相关策略**

服务器不可以被外部的主机ping，但是自己可以ping外界的主机

Iptables -I INPUT -p icmp --icmp-type echo-request -j DROP(禁止入站的ping请求，不拒绝入站的回应包)

关于 icmp 的类型， 可以参考iptables -p icmp --help

## **防火墙扩展规则**

根据IP过滤的确定是对方修改IP后，防火墙是失效,所以要**根据MAC地址过滤**

iptables -I INPUT -p tcp --dport **22** **-**m mac --mac-source **52:54:00:00:00:**0b **-**j DROP

(源mac地址是这个的，访问22 端口会被拒绝)

**基于多端口设置过滤规则**

iptables -A INPUT -p tcp -m multiport --dports **20:22,25,80,110,143,16501:16800** **-**j ACCEPT

//一次性开启20,21,22,25,80,110,143,16501到16800所有的端口

允许从 192.168.4.10-192.168.4.20 登录

iptables -A INPUT -p tcp --dport **22** -m iprange --src-range **192.168.4.10-192.168.4.20** **-**j ACCEPT

**设置防火墙规则，实现IP地址的伪装（SNAT源地址转换）**

首先要确保服务器的路由转发功能是开的

iptables -t nat -A POSTROUTING -s **192.168.4.0/24** –p tcp --dport **80** -j SNAT --to-source **192.168.2.5**

（所有从192.168.4.0/24过来访问web服务器80端口的，被转换为192.168.2.5。）实现ip地址伪装（把私有ip转换为公有ip，在外边以公网的身份进行数据的传输，如果抓包的话，也只能抓到共有地址的ip）