# 第 14 章　iptables 防火墙

iptables 防火墙这部分的内容是为没有类似阿里云云服务器的安全组的其他厂商云服务器所准备的。是的，跟大家脑海里想的一样，安全组基本等价于 iptables 防火墙。只不过对大部分普通用户来说，阿里云所提供的安全组使用更方便一些。

## 14.1　准备

### 14.1.1　iptables 防火墙介绍

iptables 是一种用在 Linux 操作系统上面的防火墙，其作用与前面所说的安全组大致相同。可以利用它去设置一些特定规则，来使得数据包经过网卡时都会被检查。服务器的系统会根据所设置的防火墙的规则，来决定到底怎么处理这些要进来的、要出去的、要被转发到别处的数据包。在每条防火墙规则（规则的英文名称是 rules）里面，可以使用一些特性去描述一下这些经过网卡时被检查的数据包的特征。例如说数据包使用的网络协议的类型、数据包的来源和去向以及数据包使用的端口号等。

如果到网卡上的数据包符合设置的某一条规则，系统就会去执行这条规则里指定的动作（动作的英文名称是 target）。它可以接受（ACCEPT）这个数据包，也就是让这个数据包通过，让它去它想去的地方；它也可以丢弃（DROP）这个数据包，也就是直接把这个数据包丢掉。因为 DROP 这个动作不会向外回应，所以例如说有人想连接我们的服务器，那么如果我们把他发过来的数据包给扔了的话，他那边的连接会一直等着，直到连接超时；假如动作的回应是拒绝（REJECT）数据包，那么这个动作会给对方一个回应，告诉对方——你被我拒绝了。

### 14.1.2　Chain：防火墙规则的分组

Chain 这个概念有点像是防火墙规则的分组，它里面比较重要的是本身规则的顺序。可以根据自己的需求去创建 Chain，默认有三个 Chain：INPUT、OUTPUT、FORWARD。INPUT 表示输入，它处理的是进入到服务器里面的数据包；OUTPUT 表示输出，它处理的是服务器向外发送的数据包，例如说想要安装或升级服务器上的某一个服务时，服务器本身就会向外发送一些数据包；FORWARD 表示转发，这个分组的数据包会从一个地方被转移到另一个地方。

每一个分组都可以设置一个默认的动作，如果数据包不符合分组里面的所有防火墙规则的话，就会去执行这个默认的动作，这个动作可以是接受、丢弃或者是拒绝。

这些 Chain 也就是规则的分组里面，会包含很多条规则。在检查数据包的时候，分组里面的规则的顺序是很重要的，它的顺序意味着规则的优先级高低，在前面的规则优先级高，在后面的规则优先级低。数据包会被按照分组里的规则的顺序来进行检查，如果符合 Chain 里面的某一条规则的描述，就会执行相对应的动作，也就是上面所说的接受、丢弃、拒绝。

当执行过动作后，系统就不会继续去检查这个数据包是不是符合剩下的规则里的描述了，

因此 Chain 里排在最前面的规则比后面的规则的优先级要高一些。由于这个系统机制的存在，所以需要在设计规则时去把一些比较通用的规则放到前面，然后把更具体的规则放到后面。例如想要禁用从某一个 IP 地址发过来的数据，那么应该在 Chain 里面设置一些比较具体的规则。

# 14.2　端口

## 14.2.1　端口扫描

传输协议就是传输数据用的方法，例如说 TCP 协议和 UDP 协议。

端口号就是数据的一个通道。例如说用户的浏览器会在用户的计算机上打开一个端口，然后去连接到服务器上的 80 端口，在这两个端口之间计算机和服务器可以相互交流数据；服务器上的一些服务会默认监听一些端口，用户可以连接到这些端口上面。例如前面提到的 Web 服务会默认监听 80 端口，SSH 服务会默认使用 22 端口。

在服务器本地或者其他的地方，都可以去查看服务器打开的这些端口。如图 14-1 所示，登录服务器，去安装 EPEL 仓库，输入命令 yum install epel-release -y 后按 Enter 键，稍微等待即可。

```
[root@VM_56_159_centos ~]# yum install epel-release -y
Loaded plugins: fastestmirror, langpacks
Repository epel is listed more than once in the configuration
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
docker-main-repo
epel
extras
```

图 14-1　安装 EPEL 仓库

然后再输入命令 yum install nginx -y，去安装 Nginx 服务。如果按照第 11 章节中所讲述的内容安装过一整套的网站运行环境后，这里可以不用再进行重复安装。然后输入命令 systemctl start nginx，来启动 Nginx 服务。

如果有多余的服务器，可以在它上面安装 Nmap 来进行端口扫描。登录这台服务器，输入命令 yum install nmap -y，然后按 Enter 键安装。

如图 14-2 所示，接下来输入命令 nmap -sT 114.114.114.114，去扫描 114 公共 DNS 提供服务的 IP 地址。这里的命令中的“s”代表 scan ，中文意思为扫描；大写的“T”代表 TCP 协议。综上所述，上面的意思即为使用 Nmap 去扫描 114.114.114.114 这个公共 DNS 使用的 TCP 协议的公开端口。当然，这个 IP 地址也可以替换成所拥有的主服务器的 IP 地址。从图 14-2 所示的结果可以看出，114.114.114.114 开放了两个使用 TCP 协议的端口。

```
[root@VM_56_159_centos ~]# nmap -sT 114.114.114.114

Starting Nmap 6.40 ( http://nmap.org ) at 2017-09-12 23:43 CST
Nmap scan report for public1.114dns.com (114.114.114.114)
Host is up (0.018s latency).
Not shown: 998 filtered ports
PORT   STATE SERVICE
53/tcp open  domain
80/tcp open  http
```

图 14-2　使用 Nmap 扫描 114 公共 DNS

当然，如果想要知道主服务器开放了哪些端口，也可以在主服务器上安装 Nmap 来扫描自己。使用上面的 Nmap 安装命令在主服务器上安装 Nmap，然后执行命令 nmap -sT localhost 来扫描自己，如图 14-3 所示。这里的“localhost”，即为本地主机。从图 14-3 所示的结果可以看

出，主服务器开放了四个使用 TCP 协议的端口。

```
[root@VM_56_159_centos ~]# nmap -sT localhost

Starting Nmap 6.40 ( http://nmap.org ) at 2017-09-13 00:22 CST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00023s latency).
Other addresses for localhost (not scanned): 127.0.0.1
Not shown: 996 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
80/tcp   open  http
88/tcp   open  kerberos-sec
3306/tcp open  mysql
```

图 14-3　使用 Nmap 扫描本地主机

## 14.2.2　查看端口上的连接

如图 14-3 所示，其中列出的端口，可能会有少部分不知道它是做什么的，也不知道它属于哪一个服务。如果出现了不了解的服务和端口号，那么可以查询系统里面的一个 ANSI 编码文件，这个文件上面会列出大部分已经了解的服务和服务所对应的端口号。登录主服务器，可以输入命令 cat /etc/services，来输出这个文件里的详细内容，详细信息如图 14-4 所示。如图 14-4 所示，第一列是服务的名字，第二列是服务的端口号和使用的协议类型，第三列是服务的简略描述（#号后面）。在某些情况下，第三列显示的信息可能是服务的别名，第四列才是与服务相关的简略描述。如果发现服务器所开放的端口既不在这个文件里面，又没有出现手工设置的端口，这种情况下就需要多加注意了，有可能是服务器被入侵了。

```
sun-as-jpda     9191/udp                # Sun AppSvr JPDA
wap-wsp         9200/tcp                # WAP connectionless session service
wap-wsp         9200/udp                # WAP connectionless session service
wap-wsp-wtp     9201/tcp                # WAP session service
wap-wsp-wtp     9201/udp                # WAP session service
wap-wsp-s       9202/tcp                # WAP secure connectionless session service
wap-wsp-s       9202/udp                # WAP secure connectionless session service
wap-wsp-wtp-s   9203/tcp                # WAP secure session service
wap-wsp-wtp-s   9203/udp                # WAP secure session service
wap-vcard       9204/tcp                # WAP vCard
wap-vcard       9204/udp                # WAP vCard
wap-vcal        9205/tcp                # WAP vCal
wap-vcal        9205/udp                # WAP vCal
wap-vcard-s     9206/tcp                # WAP vCard Secure
wap-vcard-s     9206/udp                # WAP vCard Secure
wap-vcal-s      9207/tcp                # WAP vCal Secure
wap-vcal-s      9207/udp                # WAP vCal Secure
rjcdb-vcards    9208/tcp                # rjcdb vCard
rjcdb-vcards    9208/udp                # rjcdb vCard
almobile-system 9209/tcp                # ALMobile System Service
```

图 14-4　查看系统中所有的服务和对应的端口等基本信息

在输出这个文件时，可以使用关键词，来搜索特定服务，例如可以输入命令 cat /etc/services | grep 80，来查找端口号中有“80”字段的服务。如图 14-5 所示，其关键词会用红色来着重显示。

```
[root@VM_56_159_centos ~]# cat /etc/services | grep 80
http            80/tcp          www www-http    # WorldWideWeb HTTP
http            80/udp          www www-http    # HyperText Transfer Protocol
http            80/sctp                         # HyperText Transfer Protocol
socks           1080/tcp                        # socks proxy server
socks           1080/udp                        # socks proxy server
corbaloc        2809/tcp                        # CORBA naming service locator
amanda          10080/tcp                       # amanda backup services
amanda          10080/udp                       # amanda backup services
omirr           808/tcp         omirrd          # online mirror
omirr           808/udp         omirrd          # online mirror
canna           5680/tcp        auriga-router
webcache        8080/tcp        http-alt        # WWW caching service
webcache        8080/udp        http-alt        # WWW caching service
tproxy          8081/tcp        sunproxyadmin   # Transparent Proxy
tproxy          8081/udp        sunproxyadmin   # Transparent Proxy
ris             180/tcp                 # Intergraph
ris             180/udp                 # Intergraph
http-mgmt       280/tcp                 # http-mgmt
http-mgmt       280/udp                 # http-mgmt
```

图 14-5　使用关键词搜索系统中特定的服务

如果想知道某一个端口到底是哪个服务打开的，可以使用命令 netstat -anp | grep 80，来查看使用该端口的服务。上面命令中的“a”代表 all，意为所有；“n”代表 numeric，意为直接使

用数字显示；“p”代表 programs，意为显示正在使用 Socket 的程序识别码和程序名称。如图 14-6 所示，从图中的结果了解到使用 80 端口的是 Nginx 服务，服务前面的数字则是进程的 ID。

```
[root@VM_56_159_centos ~]# netstat -anp | grep 80
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      3016/nginx: master
unix  2      [ ACC ]     STREAM     LISTENING     22358    698/dockerd         /run/docker/libnetwork/f24f775b6a481e088803b173a3f5d00f77487eb4da46ff90ca6bcae0fde76069.sock
unix  3      [ ]         STREAM     CONNECTED     21806    840/docker-containe  /var/run/docker/libcontainerd/docker-containerd.sock
unix  3      [ ]         STREAM     CONNECTED     21805    698/dockerd
unix  3      [ ]         STREAM     CONNECTED     15980    425/lsmd
```

图 14-6　查看正在使用 80 端口的服务

还可以只列出当前使用 TCP 协议的服务的进程，输入命令 netstat -ntp，可以得到如图 14-7 所示的反馈结果。如图 14-7 所示，图中的 Local Address，代表主服务器的本机地址；Foreign Address 代表正在连接主服务器的那台主机的地址。可以看到有两个地址，第一个地址是本地局域网地址，第二个地址则是当前使用的长城宽带的浮动公网 IP 地址；State 表示状态，上面显示的 ESTABLISHED 意为连接已创建；后面的则是进程的 ID 和服务名称。如果这时有其他活动连接到 Web 服务，本机地址等一些信息就会发生变化，再次输入上面的 netstat -ntp 命令，就可以看到其详细信息。

```
[root@VM_56_159_centos ~]# netstat -ntp
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 10.105.56.159:59600     10.252.230.17:9988      ESTABLISHED 899/secu-tcs-agent
tcp        0    356 10.105.56.159:10101     42.199.49.156:24664     ESTABLISHED 29772/sshd: root@pt
```

图 14-7　查看正在使用 TCP 协议的服务的进程

# 14.3　基础

## 14.3.1　iptables 基本命令

由于执行 iptables 命令需要 root 权限，所以可以使用 root 用户登录或者在命令前加 sudo 获得超级管理员权限。

例如想要查看系统现有的防火墙规则，可以输入命令 iptables -L 或 iptables --list，这两个命令的意思相同，其作用是让系统列出所有的防火墙分组和里面的规则。如图 14-8 所示，一般情况下，会看到这三个 Chain，它们就是 iptables 默认的防火墙的分组。大多数情况下，默认的 iptables 里面是没有规则的，而且 policy 的动作都是 ACCEPT；当数据包不符合所有的规则后，系统才会去执行这个默认的 policy。

```
[root@VM_56_159_centos ~]# iptables --list
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ezmeeting-2
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ftp
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpts:dnp:ndmps
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:https
ACCEPT     icmp --  anywhere             anywhere             limit: avg 1/sec burst 10
ACCEPT     all  -f  anywhere             anywhere             limit: avg 100/sec burst 100
syn-flood  tcp  --  anywhere             anywhere             tcp flags:FIN,SYN,RST,ACK/SYN
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
DOCKER-ISOLATION  all  --  anywhere             anywhere
DOCKER     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere

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

图 14-8　查看系统现有的防火墙规则

如果想要查看现有的规则是执行什么样的 iptables 命令生成的，可以使用 iptables -S 或 iptables --list-rules 命令。如图 14-9 所示，如果没有往里面添加具体的防火墙规则，会看到这三条默认的命令，它们就是设置分组默认动作的那些命令，且 policy 的默认动作应为 ACCEPT。

```
[root@VM_56_159_centos ~]# iptables --list-rules
-P INPUT DROP
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
```

图 14-9　查看系统现有的防火墙规则的生成过程

如果想把 FORWARD 这个 Chain 的默认 policy 改成 DROP，可以输入命令 iptables -P FORWARD DROP 来更改。如图 14-10 所示，更改后输入命令 iptables --list-rules 查看，从图 14-10 中所示的反馈结果得知上面的更改命令生效了。如果想恢复原样，可以使用 iptables -P FORWARD ACCEPT 这个命令。命令中的“P”是 policy 的缩写，后面的 FORWARD 和 ACCEPT，可以根据实际需要灵活变通。

```
[root@VM_56_159_centos ~]# iptables -P FORWARD DROP
[root@VM_56_159_centos ~]# iptables --list-rules
-P INPUT DROP
-P FORWARD DROP
-P OUTPUT ACCEPT
```

图 14-10　将 FORWARD 默认策略改成 DROP

## 14.3.2　禁止指定来源的访问

这里的指定来源，一般情况下是指某个具体的 IP 地址或者 IP 地址段。

在防火墙的规则里面，要指定一下规则应用的 Chain，描述一下数据包的特征，然后再加上想要执行的具体动作。

如图 14-11 所示，在阿里云的云盾态势感知管理控制台上，得知有一些 IP 地址总是攻击服务器，去请求一些它没有权限的资源，不断地向服务器发出数据包，消耗了服务器的资源。

代码/命令执行 (301)　本地文件包含 (0)　远程文件包含 (0)　脚本木马 (186)　上传漏洞 (1)　路径遍历 (0)　拒绝服务 (0)　越权访问 (0)　CSRF (0)　CRLF (0)　其他 (285)

| 被攻击应用 ⑦ | 被攻击的URL地址 | 请求方式 | 攻击类型 | 攻击者IP |
|---|---|---|---|---|
| ipc.im | http://ipc.im/wp-login.php | POST | 其他 | 120.52.18.49(中国-河北省-廊坊市) |
| ipc.im | http://ipc.im/wp-login.php | POST | 其他 | 120.52.18.49(中国-河北省-廊坊市) |

图 14-11　阿里云的云盾态势感知管理控制台上显示的攻击服务器的 IP 地址

为了改善这种情况，可以添加一条防火墙规则去禁止这个 IP 地址的访问。登录服务器，输入命令 iptables -A INPUT -s 120.52.18.49 -j DROP 后按 Enter 键确认即添加成功。命令中的“A”即为 Append，意思是向规则链中添加条目，后面的“INPUT”则是要添加的规则链的名字；“s”则是 source，代表着数据包的来源，紧挨着的 IP 地址则是要禁止的指定来源；“j”则是 jump，作用是指定要跳转的目标，后面紧挨着的动作则是想要执行的 policy。这条规则的作用，就是扔掉 120.52.18.49 这个 IP 地址发来的所有的数据包。

## 14.3.3　禁止指定来源访问指定协议的端口号

继续在阿里云的云盾态势感知管理控制台上查看详细信息，发现攻击我们的 IP 地址与上面我们禁止掉的 IP 地址好像在同一个网段，即在 120.52.18.1 至 120.52.18.256 这一整个 C 类 IP 地

址的范围内。如果想要了解更多关于 IP 地址子网的知识，可以去百度查找“IP CIDR”这个关键词，这里就不再多做阐述。

如图 14-12 所示，登录主服务器，输入命令 iptables -A INPUT -s 120.52.18.0/24 -p tcp --dport 80 -j DROP，然后按 Enter 键确认，防火墙规则就添加成功了。命令中的“p”，即为 protocol，代表协议类型，后面紧挨着的“tcp”则是我们指定的协议；“dport”代表目标端口，后面的“80”则是我们指定的端口号。这条规则的作用，就是禁止 120.52.18.1 至 120.52.18.254 之间的 IP 地址使用 TCP 协议发送到 80 端口的数据包。虽然一整个 C 类 IP 地址有 256 个地址，但实际上可用的主机地址只有 254 个。

```
[root@VM_56_159_centos ~]# iptables -A INPUT -s 120.52.18.0/24 -p tcp --dport 80 -j DROP
[root@VM_56_159_centos ~]#
```

图 14-12　禁止指定来源访问指定协议的端口号

# 14.4　管理规则

## 14.4.1　列出防火墙规则

查看现有的防火墙的规则，可以使用命令 iptables -L，“L”代表 List。在 INPUT 这个规则链里面，可以看到之前添加的两条规则。如果使用 iptables -L --line-numbers 这个命令，就会在这两条规则前面加上数字行号。如果想要查看这两条规则是执行什么样的 iptables 命令生成的，可以使用 iptables -S 或 iptables --list-rules 命令。

如果想要列出指定的规则链例如 INPUT 这个 Chain 里面的规则，可以使用 iptables -L INPUT 这个命令去查看。如果想要得到更详细的信息，可以在这条命令后面加上-v，即使用 iptables -L INPUT -v 这个命令。如图 14-13 所示，会发现系统所展示的信息里多了一些内容，例如 pkts 和 bytes，它们分别是数据包的数量和数据包的大小。如果想要重置这里面的统计信息的话，可以使用 iptables -Z 这个命令，“Z”代表 Zero；如果想要清除指定的规则链里的信息的话，可以加上这个 Chain 的名字。清除完成以后，执行命令 iptables -L INPUT -v，会发现里面的 pkts 和 bytes 的数字都变成了零。

```
[root@VM_56_159_centos ~]# iptables -L INPUT
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ezmeeting-2
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ftp
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpts:dnp:ndmps
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:http
ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:https
ACCEPT     icmp --  anywhere             anywhere             limit: avg 1/sec burst 10
ACCEPT     all  -f  anywhere             anywhere             limit: avg 100/sec burst 100
syn-flood  tcp  --  anywhere             anywhere             tcp flags:FIN,SYN,RST,ACK/SYN
REJECT     all  --  anywhere             anywhere             reject-with icmp-host-prohibited
DROP       all  --  120.52.18.49         anywhere
DROP       tcp  --  120.52.18.0/24       anywhere             tcp dpt:http
[root@VM_56_159_centos ~]# iptables -L INPUT -v
Chain INPUT (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
 2012  101K ACCEPT     all  --  lo     any     anywhere             anywhere
3277K  282M ACCEPT     all  --  any    any     anywhere             anywhere             state RELATED,ESTABLISHED
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             state NEW tcp dpt:ssh
  234 11324 ACCEPT     tcp  --  any    any     anywhere             anywhere             state NEW tcp dpt:ezmeeting-2
  424 20624 ACCEPT     tcp  --  any    any     anywhere             anywhere             state NEW tcp dpt:ftp
    1    44 ACCEPT     tcp  --  any    any     anywhere             anywhere             state NEW tcp dpts:dnp:ndmps
    1    40 ACCEPT     tcp  --  any    any     anywhere             anywhere             state NEW tcp dpt:http
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             state NEW tcp dpt:https
 149K   14M ACCEPT     icmp --  any    any     anywhere             anywhere             limit: avg 1/sec burst 10
    0     0 ACCEPT     all  -f  any    any     anywhere             anywhere             limit: avg 100/sec burst 100
   52  2484 syn-flood  tcp  --  any    any     anywhere             anywhere             tcp flags:FIN,SYN,RST,ACK/SYN
 2500  117K REJECT     all  --  any    any     anywhere             anywhere             reject-with icmp-host-prohibited
    0     0 DROP       all  --  any    any     120.52.18.49         anywhere
    0     0 DROP       tcp  --  any    any     120.52.18.0/24       anywhere             tcp dpt:http
```

图 14-13　列出防火墙规则

### 14.4.2 追加与插入规则

防火墙规则的顺序很重要，因为一个数据包进入以后，会被从第一条规则开始检查，遇到符合的规则就会执行对应的动作，执行以后就不会再继续使用后面的规则来检查这个数据包了。

之前在添加规则的时候，都使用了一个“-A”的选项，它全称为 Append，中文意思为添加。使用这个选项后，新添加的规则会追加到现有的规则的底部，也就是新添加的规则会排列在原有规则的最后面。如果想避免这种情况发生，可以使用“-I”这个选项，把新添加的规则插入到指定的位置上。

可以输入命令 iptables -L --line-numbers，先查看一下目前所有的规则链和它里面的所有规则。如图 14-14 所示，现在想添加一条规则，作为 INPUT 这个 Chain 里面的第一条规则，输入命令 iptables -I INPUT 1 -i lo -j ACCEPT。可以在“-I”选项后加上想要插入的规则链的名字，以及想要插入的行号数字；“-i”的作用，是指定数据包进入本机的网络接口，后面的“lo”则是 loopback 的缩写。这条规则的作用是，可以允许本地流量自由进入，而且是最高优先级。再次执行 iptables -L --line-numbers 这个命令，可以看到新添加的命令已经被插入到了指定位置上。

```
[root@VM_56_159_centos ~]# iptables -I INPUT 1 -i lo -j ACCEPT
[root@VM_56_159_centos ~]# iptables -L --line-numbers
Chain INPUT (policy DROP)
num  target     prot opt source               destination
1    ACCEPT     all  --  anywhere             anywhere
2    ACCEPT     all  --  anywhere             anywhere
3    ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
```

图 14-14　追加与插入规则

### 14.4.3 删除规则与清空所有规则

想要删除现有的规则，最简单的方法就是去使用规则的行号的数字来作为依据来删除。输入命令 iptables -L --line-numbers，执行后发现我们的 INPUT 这个规则链里面，第一条规则和第二条规则重复了，于是需要删除一条多余的规则。

如图 14-15 所示，为了与前面添加的规则作出区别，删除掉 INPUT 这个 Chain 里面的第二条规则，输入并执行命令 iptables -D INPUT 2；“D”是英文 Delete 的缩写，意为删除。完成后使用命令 iptables -L --line-numbers 再查看一下，会发现多余的规则已经被成功删除了。

```
[root@VM_56_159_centos ~]# iptables -D INPUT 2
[root@VM_56_159_centos ~]# iptables -L --line-numbers
Chain INPUT (policy DROP)
num  target     prot opt source               destination
1    ACCEPT     all  --  anywhere             anywhere
2    ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
3    ACCEPT     tcp  --  anywhere             anywhere             state NEW tcp dpt:ssh
```

图 14-15　删除现有的规则

如果想要清空某个规则链例如 INPUT 这个规则链里面的所有规则，可以执行命令 iptables -F INPUT 来完成；这里的“F”是英文 Flush 的缩写，中文意思为冲洗。如果不在选项“-F”后面指定某个规则链，iptables 里面的所有规则都会被清空，使用的时候一定要观察清楚了解命令后再执行，以免意外情况发生。上面的命令执行后，再使用命令 iptables -L --line-numbers 去查看一下，会发现 INPUT 这个规则链里的所有规则已经被清空干净了。

### 14.4.4　CentOS：保存防火墙规则

前面添加的防火墙规则，如果不对它们进行保存，那么它们会在系统重新启动后消失，故必须找到一种方法把这些配置完成的规则保存下来。

CentOS 7 这个系列的系统自带一个叫做 FirewallD 的防火墙，不过系统目前使用的还是 iptables。可以先把系统自带的 FirewallD 防火墙关掉，然后再去安装一个叫做 iptables-services 的东西。

执行命令 yum install iptables-services -y，完成以后使用命令 systemctl start iptables 去启动这个服务。如果之前使用了 OneinStack 这个一键开源工具，那么就有可能已经安装了这个服务；如果不确定自己是否已经安装，可以使用 rpm -qa | grep iptables 这个命令查看详细信息。启动服务以后，再使用 systemctl enable iptables 这个命令，让服务加入到开机自启动中。

防火墙规则会被保存到一个配置文件里面，可以使用命令 cat /etc/sysconfig/iptables 这个命令去查看文件里面的内容，内容里面列出的是一些比较常用的防火墙的规则。当然，自己新添加的规则，也会被保存到这个文件里面。重新启动 iptables 这个服务，可以立即应用这个配置文件里面的防火墙规则。执行命令 systemctl restart iptables，可以达到重启服务的目的。输入 iptables -L 这个命令，系统会反馈给目前系统里面的防火墙规则的信息，这些规则保存在根目录下的 etc/sysconfig 的 iptables 配置文件里。

如图 14-16 所示，可以使用命令 iptables -A INPUT -s 120.52.18.50 -j DROP，去新添加一条防火墙规则，禁止 120.52.18.50 这个 IP 地址的访问，然后输入 service iptables save 这个命令，去保存新添加的防火墙规则到 iptables 配置文件里面，完成后输入命令 systemctl restart iptables，使得新添加的规则立即被重启生效。如图 14-17 所示，使用命令 iptables -L --line-numbers 查看，发现新添加的防火墙规则已经成功运行了。

```
[root@VM_56_159_centos ~]# iptables -A INPUT -s 120.52.18.50 -j DROP
[root@VM_56_159_centos ~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
[root@VM_56_159_centos ~]# systemctl restart iptables
```

图 14-16　添加新的防火墙规则并保存防火墙规则

```
[root@VM_56_159_centos ~]# iptables -L --line-numbers
Chain INPUT (policy DROP)
num  target     prot opt source               destination
1    ACCEPT     all  --  anywhere             anywhere
2    ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
3    ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:ssh
4    ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:ezmeeting-2
5    ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:ftp
6    ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpts:dnp:ndmps
7    ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:http
8    ACCEPT     tcp  --  anywhere             anywhere            state NEW tcp dpt:https
9    ACCEPT     icmp --  anywhere             anywhere            limit: avg 1/sec burst 10
10   ACCEPT     all  -f  anywhere             anywhere            limit: avg 100/sec burst 100
11   syn-flood  tcp  --  anywhere             anywhere            tcp flags:FIN,SYN,RST,ACK/SYN
12   REJECT     all  --  anywhere             anywhere            reject-with icmp-host-prohibited
13   DROP       all  --  120.52.18.49         anywhere
14   DROP       tcp  --  120.52.18.0/24       anywhere            tcp dpt:http
15   DROP       all  --  120.52.18.50         anywhere
```

图 14-17　查看新添加的防火墙规则是否成功生效

## 14.5　实施规则

### 14.5.1　默认的动作：Default Policy

每一个规则链都有一个默认的策略，也就是说一个数据包如果不符合所有的规则的描述的

话，就会去执行这个默认的策略的指定动作。一般来说，这个默认的策略的指定动作是 ACCEPT，意思就是允许通过，这个默认的策略有点像黑名单，需要添加不需要的所有的数据包的特征，让它们被扔掉或者被拒绝。实际上，这么做会让防火墙规则越来越庞大，以至于某一天拖累系统的运行。

为了解决这个潜在的麻烦，可以去更改默认的策略，设置为默认拒绝所有的数据包，然后在防火墙规则里面描述一些所需要的数据包的特征，以方便这些符合需要的数据包通过。可以把规则链默认的策略，设置为 DROP 或者 REJECT。这样做会有许多好处，因为知道我们的服务器需要提供什么样的服务，只需要让这些服务发送或者接受的数据包正常通过就行了；同时，也可以去拒绝个别场景下的数据包，以让我们的服务器正常运行，例如禁止某一个想要干坏事的 IP 地址来访问我们的服务器。

### 14.5.2 把默认的 Policy 改成 DROP

可以先执行命令 iptables -L，查看一下防火墙规则的列表。会发现 INPUT、OUTPUT、FORWARD 这三个规则链默认的策略都是 ACCEPT。

如果想把这些 Chain 的默认策略改成 DROP，可以先执行 iptables -F 命令，把所有的防火墙规则都清洗掉，以便重新添加规则。然后执行命令 service iptables save，保存一下配置文件。

接下来准备把 INPUT 这个 Chain 的默认的策略改成 DROP，不过需要注意的是，不能把自己挡在外面，所以需要添加一个规则，允许当前已经建立的相关连接的数据包能够进入服务器。如图 14-18 所示，可以执行命令 iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT，来达到上面描述的意图。这里的“m”是英文 module 的缩写，意思是模块；后面的 conntrack 则是模块的名称，中文意为连接跟踪，它提供了一些功能，例如说 ctstate。其中的“ESTABLISHED”是允许已经创建的连接保持当前状态；紧接着在英文逗号后面，又加了一个“RELATED”，表示跟这个连接相关的数据包。


图 14-18 允许当前已经建立的相关连接的数据包能够进入服务器

然后再为 SSH 服务打开需要的端口，一般情况下默认端口为 22。输入命令 iptables -A INPUT -p tcp --dport 22 -j ACCEPT 后按 Enter 键确认，即可打开 SSH 服务默认的 22 端口，让 SSH 服务能够使用 22 端口传输数据包。

做完上面的操作以后，就可以把 INPUT 这个规则链的默认策略更改成 DROP 了。输入命令 iptables -P INPUT DROP 后按 Enter 键确认，然后输入命令 service iptables save 保存配置。

### 14.5.3 允许本地流量

有一些数据包会从本地主机发出，目的地也是本地主机。这些流量会用到一个虚拟网卡，名为 loopback，它还有一个缩写“lo”。可以查看一下网络相关的配置，输入命令 ifconfig，可以看到 lo 相关的设备信息，如图 14-19 所示。

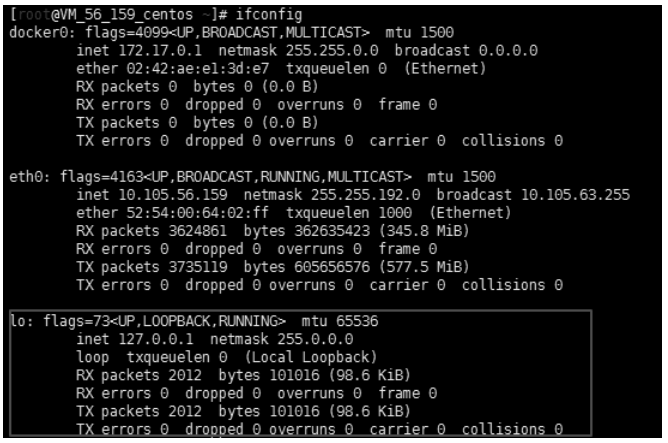


图 14-19　查看系统的网络配置

一般本地主机的服务与服务之间进行通信的时候，会用到这个虚拟网卡，例如说在服务器上安装数据库的服务、本地主机上的应用连接到本地主机上的数据库的时候，就会用到这个名叫 lo 的虚拟网卡。一般来说，数据库连接的端口就是给数据库服务的那个端口号，需要允许这样的流量通过虚拟网卡。

输入命令 iptables -I　INPUT 1 -i lo -j ACCEPT，把一条新规则插入到 INPUT 这个 Chain 的第一行；因为这种流量比较常见，所以需要把这条规则插入到第一行。这里的“i”则是 In.interface 的缩写，作用是指定数据包进入的网络接口，后面的“lo”则是网络设备的缩写。设置好以后，使用 service iptables save 这个命令，来保存一下新添加的防火墙规则。

### 14.5.4　允许 Web 服务

Web 服务的默认端口号是 80，如果我们的网站使用了 SSL 安全证书加密，那么还需要一个额外的端口——443。

为了让潜在的用户可以访问到我们的网站，需要允许在这两个端口上传输的数据包通过。如图 14-20 所示，执行命令 iptables -A INPUT -p tcp --dport 80 -j ACCEPT，然后再执行命令 iptables -A INPUT -p tcp --dport 443 -j ACCEPT；最后输入命令 service iptables save，保存一下配置文件。

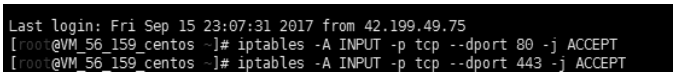


图 14-20　允许 Web 服务

在系统重启以后，就可以应用刚刚设置的规则；也可以选择立即重启服务，让这些规则马上生效。具体命令请大家翻阅本章节前面的文字后再进行操作，这里就不再多做阐述。

**拓展知识：**

iptables 命令线上手册 http://man.linuxde.net/iptables