防火墙配置

一 简介

1 功能：

分割内网和外网

划分要被保护的服务器

通过源端口，源ip，源mac，包中特定标记 和目录端口，IP，mac 来确定数据包是否可以通过防火墙

2 分类

1）数据包过滤

分析ip和端口，mac是否符合规则，如果符合，接受

2）代理服务器

3 防火墙限制

1）防火墙不能有效防止病毒和木马

2）防火墙一般不设定对内部访问规则，所以对内部攻击无效

4 防火墙配置原则

拒绝所有，逐个允许

允许所有，逐个拒绝

5 linux常见防火墙软件

A. iptables

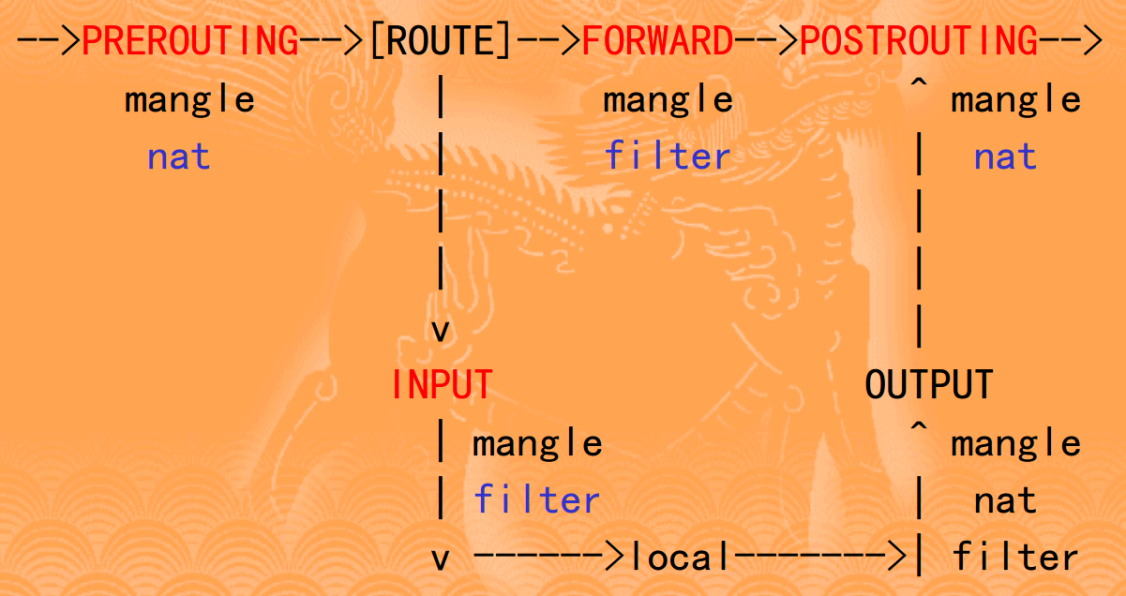
B. ipchains

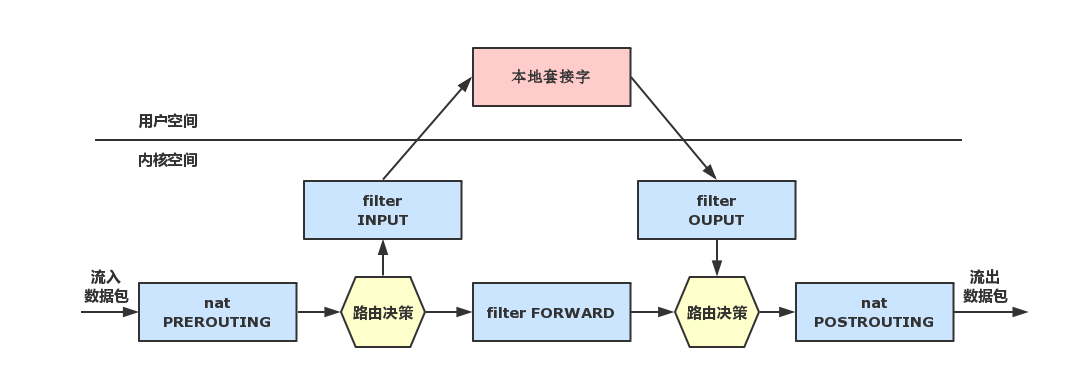
二 iptables防火墙

iptables 组成：四张表 + 五条链 + 规则

1 结构

表-------链--------规则





2 表

filter表 数据过滤表，一般的过滤功能

NAT表 网络地址转换，用于nat功能（ 端口映射，地址映射等 ）

Mangle 特殊数据包标记, 查找有没有某些字符串

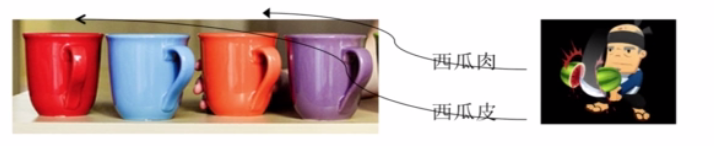
3 链

五条链：INPUT OUTPUT FORWARD PEROUTING POSTROUTING

filter表中： INPUT OUTPUT FORWARD

4 链与表的关系

表就是杯子，西瓜肉，西瓜片，西瓜籽等就是链。每个链往不同的表里面放。不同的表有不同的功能。



5 iptables规则组成

数据包访问控制：ACCEPT DROP REJECT

数据包改写： SNAT(改变他源地址) DNAT（改变目标地址）

信息记录： LOG

规则详细描述：

ACCEPT 放行

DROP 丢弃

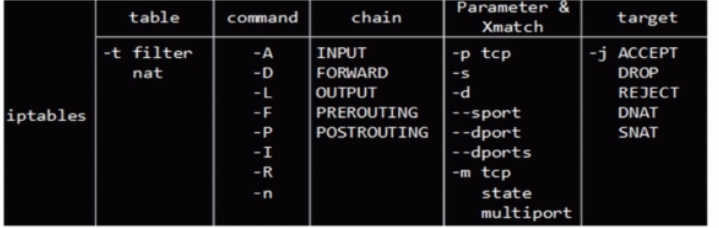
REJECT 拒绝

DNAT 目标地址转换（即修改目标地址）

SNAT 源地址转换

LOG 记录日志

三 iptables基础语法



1 规则的查看和清楚

iptables [-t 表名] [选项] -n

选项：

-L 查看

-F 清除所有规则

-X 清除自定义链

-Z 清除所有链统计

-n 以端口和ip显示

iptables -t nat -L

查看nat表中规则

iptables -L

查看filter表中规则

2 定义默认策略

iptables -t 表名 -P 链名 ACCEPT|DROP

-P（大） 定义默认策略

iptables -t filter -P INPUT DROP

注意：不要把自己踢出服务器，应该最后设定。

3 ip和网卡接口设置

iptables [-AI 链] [-io 网卡接口] [-p 协议] [-s 源IP] [-d 目标ip] -j 动作

-A 追加链规则 在链规则最后加入此规则

-I INPUT [n] 插入一条规则，插入对应链上的第n条，默认是第一条

-D 链 [n] 删除指定链的指定第n条规则

iptables -D INPUT 2 #删除input链第二条规则

-i eth0 指定进入接口 要在INPUT链上定义

-o eth0 指定传出接口 要在OUTPUT链上定义

-p 协议 tcp udp icmp all

-j 动作 ACCEPT DROP

iptables -A 　INPUT -i lo -j ACCEPT

允许本机回环网卡通信，在INPUT链

iptables -A INPUT -i eth0 -s 192.168.140.254 -j ACCEPT

允许254进入eth0

iptables -A INPUT -i eth0 -s 192.168.140.0/24 -j DROP

拒绝140网段访问

4 设定端口访问

iptables -A INPUT -i eth0 -p all -s 源ip --sport 源端口 -d 目标ip --dport 目标端口 -j 动作

iptables -A INPUT -i eth0 -p tcp -s 192.168.140.0/24 --dport 22 -j DROP

iptables -A INPUT -i eth0 -p tcp -s 192.168.140.0/24 --dport 137:139 -j ACCEPT

指定端口时，协议不能用all，要指定确切协议

5 模块调用

1）

-m 模块名 模块选项 加载iptables功能模块

-m state --state ESTABLISHED,RELATED

iptables -A INPUT -i eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT

state 状态模块 常见状态 ESTABLISHED 联机成功的状态 RELATED 返回包状态

limit 模块 作用：限速，控制流量，防止CC攻击

CC攻击的原理就是[攻击者](http://baike.baidu.com/view/2945352.htm" \t "http://baike.baidu.com/_blank)控制某些[主机](http://baike.baidu.com/view/23880.htm" \t "http://baike.baidu.com/_blank)不停地发大量[数据包](http://baike.baidu.com/view/25880.htm" \t "http://baike.baidu.com/_blank)给对方服务器造成服务器资源耗尽，一直到宕机崩溃

实例：

规则：本机（192.168.1.168）只允许10个包过来，超过了，就每分钟是1个包

iptables -A INPUT -p icmp -m limit --limit 1/m --limit-burst 10 -j ACCEPT

# 只允许第一条规则生效，否则就会将数据包丢弃

iptables -A INPUT -p icmp -j DROP

测试（在其他机器ping 192.168.1.168）

ping 192.168.1.168

2）

-m mac --mac-source 按照mac地址限制访问

iptables -A INPUT -m mac --mac-source aa:bb:cc:dd:ee:ff -j DROP

#拒绝某mac访问

3）

-m string --string "想要匹配的数据包中字串"

iptables -A FORWARD -p udp --dport 53 -m string --string "tencent" --algo kmp -j DROP

#通过dns拒绝qq登录

--algo 选择模式匹配策略，支持KMP和BM两种字符串搜索算法，任意指定一个即可

6 简易防火墙

iptables -F

iptables -A INPUT -i lo -j ACCEPT

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

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

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

iptables -A INPUT -p tcp --dport 873 -j ACCEPT

iptables -A INPUT -p tcp --dport 139 -j ACCEPT

iptables -A INPUT -p tcp --dport 21 -j ACCEPT

iptables -P INPUT DROP

7 防火墙服务开机自启动

chkconfig iptables on

8 防火墙规则开启自启动的两种方法

1）

#写完防火墙规则后

service iptables save

#会把规则保存到 /etc/sysconfig/iptables文件中，重启会自动读取

chkconfig iptables on #设置开机自启动

2） 手工写防火墙脚本

如 vi /root/iptables.rule

iptables -A INPUT -i lo -j ACCEPT ##允许本机ping本机

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

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

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

iptables -A INPUT -p tcp --dport 873 -j ACCEPT

iptables -A INPUT -p tcp --dport 139 -j ACCEPT

iptables -A INPUT -p tcp --dport 21 -j ACCEPT

iptables -P INPUT DROP

赋予执行权限 chmod 755 /root/iptables.rule

开机运行 vi /etc/rc.local

写入 /root/iptables.rule

其他常用命令：

#查看iptables版本

iptables -v

# 列出已经存在的iptables

iptabls -L

# 清除已存在的iptables

iptabls -F

-X 删除一个自定义的空链

查看相关参数：

-L 显示指定表中的规则

-n 以数字格式显示主机地址和端口号

实例：

场景一

本机：192.168.1.168

其他主机： 192.168.1.250

规则1：对所有的地址开放本机(192.168.1.168)的tcp(80、22、10-21)端口的访问

规则2：允许对所有的地址开发本机的基于ICMP(ping)协议的数据包访问（也就是允许ping）

规则3：其他未被允许的端口则禁止访问

实例步骤：

#允许其他机器访问本机的80端口

iptables -I INPUT -p tcp --dport 80 -j ACCEPT

#22端口一定要开启，否则无法远程登录

iptables -I INPUT -p tcp --dport 22 -j ACCEPT

#允许10到21端口访问

iptables -I INPUT -p tcp --dport 10:21 -j ACCEPT

#允许ping

iptables -I INPUT -p icmp -j ACCEPT

#将其他的都给拒绝 -A代表追加

iptables -A INPUT -j REJECT

删除操作：

#删除已有的iptables

iptables -D INPUT -p tcp --dport 80 -j ACCEPT

#将80端口禁用

iptables -I INPUT -p tcp --dport 80 -j REJECT

其他：

这个时候，其他机器就不能登录MySQL了，因为3306端口没有开放。

查看本机开发的端口，可以通过端口扫描查看， nmap -sS -p 0-1000 192.168.1.168

nmap工具需要安装 yum -y install nmap

以上的例子存在问题：

1. 本机无法访问其他主机。可以通过curl [http://192.168.1.250](http://192.168.203.250) 查看。
2. 本机无法访问本机

解决方法：

#解决本机不能访问本机的问题

iptables -I INPUT -i lo -j ACCEPT

#解决本机无法访问其他主机的问题

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

实例2：

在场景一的基础上，修改只允许192.168.1.250访问本机的httpd(ssh)服务。

22 端口运行

#先清除原来场景一添加的iptables 针对httpd的规则

iptables -D INPUT -p tcp --dport 80 -j ACCEPT

#只允许192.168.1.250访问本机

iptables -I INPUT -p tcp -s 192.168.1.250 --dport 80 -j ACCEPT