# 1.tcp\_wrappers

在第四层(传输层)的安全工具通过 /etc/hosts.allow 和/etc/hosts.deny文件管控:

只有由对supper daemon(xinetd)管理的服务或者含有libwrap.so模块的服务起作用

[root@www ~]# ldd `which sshd` | grep libwrap.so

libwrap.so.0 => /lib64/libwrap.so.0 (0x00007f4dc946f000)

root@www xinetd.d]# ldd `which xinetd` | grep libwrap.so

libwrap.so.0 => /lib64/libwrap.so.0 (0x00007fa4362dd000)

规则:

先以/etc/hosts.allow为优先对比,符合条件就放行

再以/etc/hosts.deny对比,符合条件就抵挡

如果不在这两个文件中,则放行

注意写法:

程序名:ip

程序名:网段/子网掩码

ALL:xxxxxxx

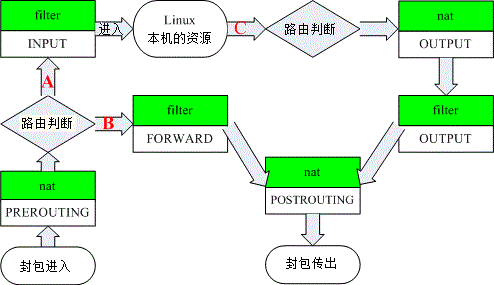
程序名:ALL

例如:/etc/hosts.deny

sshd:192.168.200.146

# 2.iptables

## 2.1简介



filter(过滤器):主要跟进入Linux本机的数据包有关

INPUT:主要与想进入到本机的数据包有关

OUPUT:主要与主机所要送出的数据包有关

FORWARD:和主机本身没关系,可以传递包到后端机器中

nat(地址转换):进行来源和目的ip或port的转换和本机无关,和后端机器相关

## 2.2命令

查看格式化后的规则:

[root@www ~]# **iptables [-t tables] [-L] [-nv]**

选项与参数：

-t ：后面接 table ，例如 nat 或 filter ，若省略此项目，则使用默认的 filter

-L ：列出目前的 table 的规则

-n ：不进行 IP 与 HOSTNAME 的反查，显示讯息的速度会快很多！

-v ：列出更多的信息，包括通过该规则的封包总位数、相关的网络接口等

iptables -L -n

iptables -t nat -L -n

查看完整的非格式化的规则:

iptables-save

清楚规则:

[root@www ~]# **iptables [-t tables] [-FXZ]**

选项与参数：

-F ：清除所有的已订定的规则；

-X ：杀掉所有使用者 "自定义" 的 chain (应该说的是 tables ）啰；

-Z ：将所有的 chain 的计数与流量统计都归零

范例：清除本机防火墙 (filter) 的所有规则

配置规则:

root@www ~]# **iptables [-AI 链名] [-io 网络接口] [-p 协议] \**

> **[-s 来源IP/网域] [-d 目标IP/网域] -j [ACCEPT|DROP|REJECT|LOG]**

选项与参数：

-AI 链名：针对某的链进行规则的 "插入" 或 "累加"

-A ：新增加一条规则，该规则增加在原本规则的最后面。例如原本已经有四条规则，

使用 -A 就可以加上第五条规则！

-I ：插入一条规则。如果没有指定此规则的顺序，默认是插入变成第一条规则。

例如原本有四条规则，使用 -I 则该规则变成第一条，而原本四条变成 2~5 号

链 ：有 INPUT, OUTPUT, FORWARD 等，此链名称又与 -io 有关，请看底下。

-io 网络接口：设定封包进出的接口规范

-i ：封包所进入的那个网络接口，例如 eth0, lo 等接口。需与 INPUT 链配合；

-o ：封包所传出的那个网络接口，需与 OUTPUT 链配合；

-p 协定：设定此规则适用于哪种封包格式

主要的封包格式有： tcp, udp, icmp 及 all 。

-s 来源 IP/网域：设定此规则之封包的来源项目，可指定单纯的 IP 或包括网域，例如：

IP ：192.168.0.100

网域：192.168.0.0/24, 192.168.0.0/255.255.255.0 均可。

若规范为『不许』时，则加上 ! 即可，例如：

-s ! 192.168.100.0/24 表示不许 192.168.100.0/24 之封包来源；

-d 目标 IP/网域：同 -s ，只不过这里指的是目标的 IP 或网域。

-j ：后面接动作，主要的动作有接受(ACCEPT)、丢弃(DROP)、拒绝(REJECT)及记录(LOG)

## 2.3配置规则