

在本博客中，从理论到实践，系统的介绍了iptables，如果你想要从头开始了解iptables，可以查看iptables文章列表，直达链接如下

iptables零基础快速入门系列

注意：在参照本文进行iptables实验时，请务必在个人的测试机上进行，因为如果iptables规则设置不当，有可能使你无法连接到远程主机中。

前文中一直在强调一个概念：报文在经过iptables的链时，会匹配链中的规则，遇到匹配的规则时，就执行对应的动作，如果链中的规则都无法匹配到当前报文，则使用链的默认策略（默认动作），链的默认策略通常设置为ACCEPT或者DROP。

那么，当链的默认策略设置为ACCEPT时，如果对应的链中没有配置任何规则，就表示接受所有的报文，如果对应的链中存在规则，但是这些规则没有匹配到报文，报文还是会被接受。

同理，当链的默认策略设置为DROP时，如果对应的链中没有配置任何规则，就表示拒绝所有报文，如果对应的链中存在规则，但是这些规则没有匹配到报文，报文还是会被拒绝。

所以，当链的默认策略设置为ACCEPT时，按照道理来说，我们在链中配置规则时，对应的动作应该设置为DROP或者REJECT，为什么呢？

因为默认策略已经为ACCEPT了，如果我们在设置规则时，对应动作仍然为ACCEPT，那么所有报文都会被放行了，因为不管报文是否被规则匹配到都会被ACCEPT，所以就失去了访问控制的意义。

所以，当链的默认策略为ACCEPT时，链中的规则对应的动作应该为DROP或者REJECT，表示只有匹配到规则的报文才会被拒绝，没有被规则匹配到的报文都会被默认接受，这就是”黑名单”机制。

同理，当链的默认策略为DROP时，链中的规则对应的动作应该为ACCEPT，表示只有匹配到规则的报文才会被放行，没有被规则匹配到的报文都会被默认拒绝，这就是”白名单”机制。

如果使用白名单机制，我们就要把所有人都当做坏人，只放行好人。

如果使用黑名单机制，我们就要把所有人都当成好人，只拒绝坏人。

白名单机制似乎更加安全一些，黑名单机制似乎更加灵活一些。

那么，我们就来做一个简单的白名单吧，也就是说，只放行被规则匹配到的报文，其他报文一律拒绝，那么，我们先来配置规则。

假设，我想要放行ssh远程连接相关的报文，也想要放行web服务相关的报文，那么，我们在INPUT链中添加如下规则。

```
[www.zsythink.net]# iptables -I INPUT -p tcp --dport 22 -j ACCEPT
[www.zsythink.net]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
```

如上图所示，我们已经放行了特定的报文，只有上述两条规则匹配到的报文才会被放行，现在，我们只要将INPUT链的默认策略改为DROP，即可实现白名单机制。

示例如下。

```
[www.zsythink.net]# iptables -nvL INPUT
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source               destination          tcp dpt:80
    0      0 ACCEPT    tcp  --  *      *        0.0.0.0/0            0.0.0.0/0            tcp dpt:22
   56 4528 ACCEPT    tcp  --  *      *        0.0.0.0/0            0.0.0.0/0            tcp dpt:22
[www.zsythink.net]# iptables -P INPUT DROP
[www.zsythink.net]# iptables -nvL INPUT
Chain INPUT (policy DROP 1 packets, 143 bytes)
 pkts bytes target    prot opt in     out     source               destination          tcp dpt:80
   105 8692 ACCEPT    tcp  --  *      *        0.0.0.0/0            0.0.0.0/0            tcp dpt:22
[www.zsythink.net]#
```

zsythink.net 朱双印博客

上图中，我们已经将INPUT链的默认策略改为DROP，并且已经实现了所谓的白名单机制，即默认拒绝所有报文，只放行特定的报文。

如果此时，我不小心执行了”iptables -F”操作，根据我们之前学到的知识去判断，我们还能够通过ssh工具远程到服务器上吗？

我想你已经判断出了正确答案，没错，按照上图中的情况，如果此时执行”iptables -F”操作，filter表中的所有链中的所有规则都会被清空，而INPUT链的默认策略为DROP，所以所有报文都会被拒绝，不止ssh远程请求会被拒绝，其他报文也会被拒绝，我们来实验一下。

```
[www.zsythink.net]# iptables -F
[www.zsythink.net]#
Connection closed by foreign host.

Disconnected from remote host
```

zsythink.net 朱双印博客

如上图所示，在当前ssh远程工具中执行”iptables -F”命令后，由于INPUT链中已经不存在任何规则，所以，所有报文都被拒绝了，包括当前的ssh远程连接。

这就是默认策略设置为DROP的缺点，在对应的链中没有设置任何规则时，这样使用默认策略为DROP是非常不明智的，因为管理员也会把自己拒之门外，即使对应的链中存在放行规则，当我们不小心使用”iptables -F”清空规则时，放行规则被删除，则所有数据包都无法进入，这个时候就相当于给管理员挖了个坑，所以，我们如果想要使用”白名单”的机制，最好将链的默认策略保持为”ACCEPT”，然后将”拒绝所有请求”这条规则放在链的尾部，将”放行规则”放在前面，这样做，既能实现”白名单”机制，又能保证在规则被清空时，管理员还有机会连接到主机，示例如下。

因为刚才的ssh连接已经被拒绝，所以，此时直接在控制台中设置iptables规则

```
[www.zsythink.net]# iptables -P INPUT ACCEPT
[www.zsythink.net]#
```

如上图所示，先将INPUT链的默认策略设置为ACCEPT

然后继续配置需要放行的报文的规则，如下图所示，当所有放行规则设置完成后，在INPUT链的尾部，设置一条拒绝所有请求的规则。

```
[www.zsythink.net]# iptables -I INPUT -p tcp --dport 22 -j ACCEPT
[www.zsythink.net]# iptables -I INPUT -p tcp --dport 80 -j ACCEPT
[www.zsythink.net]# iptables -A INPUT -j REJECT
[www.zsythink.net]#
[www.zsythink.net]# iptables -nL INPUT
Chain INPUT (policy ACCEPT)
target    prot opt source               destination          tcp dpt:80
ACCEPT    tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
ACCEPT    tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:22
REJECT    all  --  0.0.0.0/0            0.0.0.0/0            reject-with
[www.zsythink.net]#
```

zsythink.net 朱双印博客

上图中的设置，既将INPUT链的默认策略设置为了ACCEPT，同时又使用了白名单机制，因为如果报文符合放行条件，则会被前面的放行规则匹配到，如果报文不符合放行条件，则会被最后一条拒绝规则匹配到，此刻，即使我们误操作，执行了”iptables -F”操作，也能保证管理员能够远程到主机上进行维护，因为默认策略仍然是ACCEPT。

其实，在之前知识的基础上，理解所谓的黑白名单机制是很容易的，此处只是将最佳实践总结了一下，希望这篇文章能够对你有所帮助。