A+

iptables详解（8）：iptables扩展模块之state扩展

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

当我们通过http的url访问某个网站的网页时，客户端向服务端的80端口发起请求，服务端再通过80端口响应我们的请求，于是，作为客户端，我们似乎应该理所应口，以便服务端回应我们的报文可以进入客户端主机，于是，我们在客户端放行了80端口，同理，当我们通过ssh工具远程连接到某台服务器时，客户端向服务端的请求，服务端再通过22号端口响应我们的请求，于是我们理所应当的放行了所有22号端口，以便远程主机的响应请求能够通过防火墙，但是，作为客户端，如果我80端口发起请求，也没有主动向22号端口发起请求，那么其他主机通过80端口或者22号端口向我们发送数据时，我们可以接收到吗？应该是可以的，因为我们为了的响应报文，已经放行了80端口与22号端口，所以，不管是"响应"我们的报文，还是"主动发送"给我们的报文，应该都是可以通过这两个端口的，那么仔细想想，安全呢？如果某些与你敌对的人，利用这些端口"主动"连接到你的主机，你肯定会不爽的吧，一般都是我们主动请求80端口，80端口回应我们，但是一般不会出现求我们的情况吧。

你心里可能会这样想：我知道哪些主机是安全的，我只要针对这些安全的主机放行对应的端口就行了，其他IP一律拒绝，比如，我知道IP为123的主机是安全的，所机开放了22号端口，以便123主机能够通过22号端口响应我们的ssh请求，那么，如果你需要管理的主机越来越多呢？你是不是每次都要为新的主机配置这些规则呢主机呢？如果有300台主机呢？80端口就更别提了，难道你每次访问一个新的网址，都要对这个网址添加信任吗？这显然不太合理。

你心里可能又会想：针对对应的端口，我用--tcp-flags去匹配tcp报文的标志位，把外来的"第一次握手"的请求拒绝，是不是也可以呢？那么如果对方使用的是UDP协议呢？似乎总是有一些不完美的地方。

那么我们仔细的思考一下，造成上述问题的"根源"在哪里，我们为了让"提供服务方"能够正常的"响应"我们的请求，于是在主机上开放了对应的端口，开放这些端口现了问题，别人利用这些开放的端口，"主动"的攻击我们，他们发送过来的报文并不是为了响应我们，而是为了主动攻击我们，好了，我们似乎找到了问题所在？

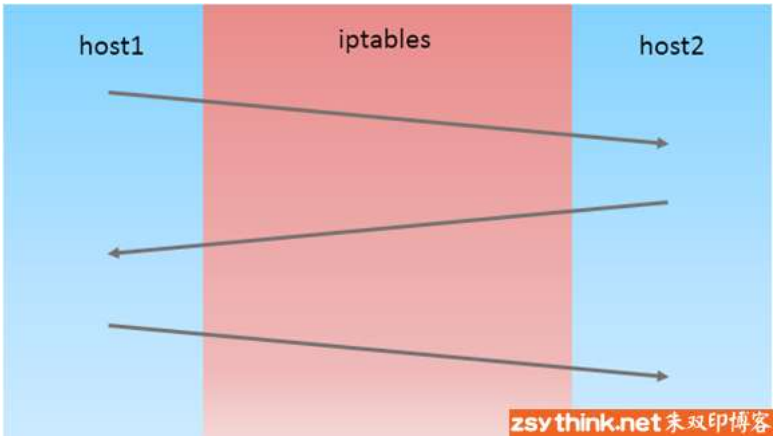
问题就是：怎样判断这些报文是为了回应我们之前发出的报文，还是主动向我们发送的报文呢？

我们可以通过iptables的state扩展模块解决上述问题，但是我们需要先了解一些state模块的相关概念，然后再回过头来解决上述问题。

从字面上理解，state可以译为状态，但是我们也可以用一个高大上的词去解释它，state模块可以让iptables实现"连接追踪"机制。

那么，既然是"连接追踪"，则必然要有"连接"。

咱们就来聊聊什么是连接吧，一说到连接，你可能会下意识的想到tcp连接，但是，对于state模块而言的"连接"并不能与tcp的"连接"画等号，在TCP/IP协议簇中，没有所谓的连接的，但是对于state模块来说，tcp报文、udp报文、icmp报文都是有连接状态的，我们可以这样认为，对于state模块而言，只要两台机器在"你来我就算建立起了连接，如下图所示



而报文在这个所谓的链接中是什么状态的呢？这是我们后面讨论的话题。

对于state模块的连接而言，"连接"其中的报文可以分为5种状态，报文状态可以为NEW、ESTABLISHED、RELATED、INVALID、UNTRACKED

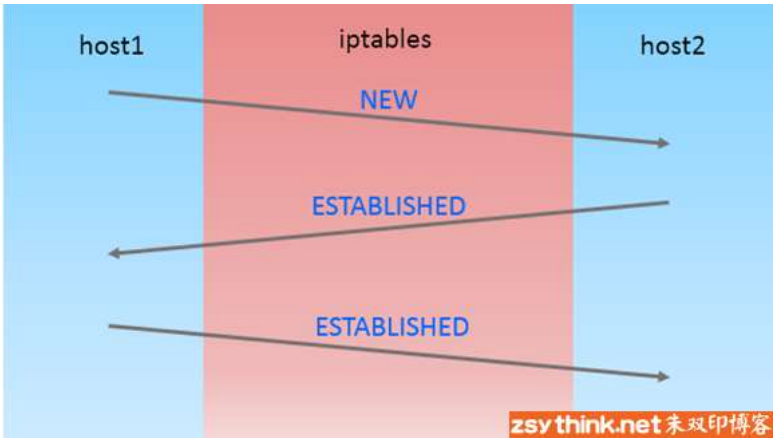
那么上述报文的状态都代表什么含义呢？我们先来大概的了解一下概念，然后再结合示例说明。

注意：如下报文状态都是对于state模块来说的。

NEW：连接中的第一个包，状态就是NEW，我们可以理解为新连接的第一个包的状态为NEW。

ESTABLISHED：我们可以把NEW状态包后面的包的状态理解为ESTABLISHED，表示连接已建立。

或许用图说话更容易被人理解



RELATED：从字面上理解RELATED译为关系，但是这样仍然不容易理解，我们举个例子。

比如FTP服务，FTP服务端会建立两个进程，一个命令进程，一个数据进程。

命令进程负责服务端与客户端之间的命令传输（我们可以把这个传输过程理解成state中所谓的一个"连接"，暂称为"命令连接"）。

数据进程负责服务端与客户端之间的数据传输（我们把这个过程暂称为"数据连接"）。

但是具体传输哪些数据，是由命令去控制的，所以，"数据连接"中的报文与"命令连接"是有"关系"的。

那么，"数据连接"中的报文可能就是RELATED状态，因为这些报文与"命令连接"中的报文有关系。

(注：如果想要对ftp进行连接追踪，需要单独加载对应的内核模块nf_conntrack_ftp，如果想要自动加载，可以配置/etc/sysconfig/iptables-config文件)

INVALID：如果一个包没有办法被识别，或者这个包没有任何状态，那么这个包的状态就是INVALID，我们可以主动屏蔽状态为INVALID的报文。

UNTRACKED：报文的状态为untracked时，表示报文未被追踪，当报文的状态为Untracked时通常表示无法找到相关的连接。

上述5种状态的详细解释可以参考如下文章的"User-land states"章节

http://www.iptables.info/en/connection-state.html

好了，我们已经大致了解了state模块中所定义的5种状态，那么现在，我们回过头想想刚才的问题。

刚才问题的根源就是：怎样判断报文是否是为了回应之前发出的报文。

刚才举例中的问题即可使用state扩展模块解决，我们只要放行状态为ESTABLISHED的报文即可，因为如果报文的状态为ESTABLISHED，那么报文肯定是之前发出的应，如果你还不放心，可以将状态为RELATED或ESTABLISHED的报文都放行，这样，就表示只有回应我们的报文能够通过防火墙，如果是别人主动发送过来的新的通过防火墙，示例如下。

```
[www.zsythink.net]# ifconfig | awk '/inet addr/ {print $1,$2}'
inet addr:192.168.43.104
inet addr:127.0.0.1
[www.zsythink.net]# iptables -F
[www.zsythink.net]# iptables -t filter -I INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
[www.zsythink.net]# iptables -t filter -A INPUT -j REJECT
[www.zsythink.net]#
[www.zsythink.net]# ssh 192.168.43.77
root@192.168.43.77's password:
```


当前主机IP为104，当放行ESTABLISHED与RELATED状态的包以后，并没有影响通过本机远程ssh到IP为77的主机上，那么此刻，我们在主机77上尝试访问104试试

```
[www.zsythink.net]# ifconfig | awk '/\<inet\>/{print $1,$2}'
inet 192.168.43.77
inet 127.0.0.1
inet 192.168.122.1
[www.zsythink.net]# ssh 192.168.43.104
ssh: connect to host 192.168.43.104 port 22: Connection refused
[www.zsythink.net]#
[www.zsythink.net]#
```


可以看到，由77主动发送到104的请求被拒绝了。

对于其他端口与IP来说，也是相同的，可以从104主动发送报文，并且能够收到响应报文，但是其他主机并不能主动向104发起请求。

好了，state模块就总结到这里，希望这篇文章能够对你有所帮助。

iptables 防火墙