**在Linux下使用iptables对ftp转发的操作说明**

1. Iptables的安装

一般Linux下已经自己安装iptables了，可以使用whereis iptables 命令查看确认，如果能看到如下类似信息,说明你已经安装了iptables

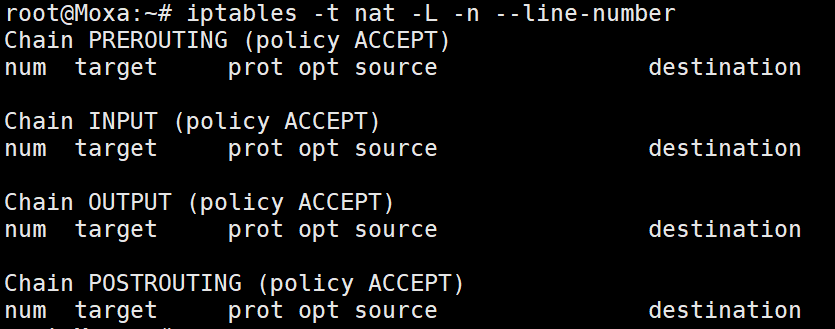
C:\Users\14307\AppData\Roaming\Tencent\Users\1430733107\TIM\WinTemp\RichOle\RMPZ]EY8M0J_K17HNYO`][H.png

如果不是这个提示,或者没有任何提示,那你的Debian上可能没有安装iptables 请使用如下命令安装: sudo apt-get install iptables

1. Iptables的使用

Iptables下有多张规则表，我们对于ftp的转发使用了nat表，下面就对其进行说明。

1．使用 iptables -t nat -L -n --line-number 命令对nat规则表进行查看，如图此时还未添加任何规则；



2. 使用 iptables -t nat -F 对nat表进行清空；

3. 使用 iptables -t nat -D 链名 行号 对nat表某一条规则进行删除

如 iptables -t nat -D POSTROUTING 1 将会删除POSTROUTING链的第一条规则；

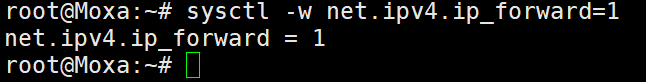
4. 使用 iptables -t nat -A 链名 规则 对nat表的各链添加规则，这里参数A表示添加到已有规则后，使用I则添加到已有规则前，默认位置1可指定。具体规则的设置将在下面说明；

5. 使用 iptables -t nat -R 链名 行号 规则 对nat表的某条规则修改（取代）。

1. sftp转发配置

sftp只使用一条链路进行连接传输，所以设置相对简单。

1. 首先开启nat转发功能 命令行输入 sysctl -w net.ipv4.ip\_forward=1 如图表示成功打开。



2. 分别对PREROUTING 和 POSTROUTING 链路进行规则配置。

Ps：这里简单说一下这两个链路的作用。

prerouting 和postrouting经常用于NAT配置网关机进行网络地址转换使用，这时候有以下规律：

PREROUTING是目的地址转换（DNAT），要把别人的公网IP换成你们内部的IP，才让访问到你们内部受防火墙保护的服务器。

POSTROUTING是源地址转换（SNAT），要把你内部网络上受防火墙保护的ip地址转换成你本地的公网地址才能让它们上网。

2.1书写PREROUTING链路规则：

iptables -t nat -A PREROUTING -s 源地址ip -p tcp –dport 源地址访问的本机端口 -j DNAT --to 目标地址ip：目标端口

如图

C:\Users\14307\AppData\Roaming\Tencent\Users\1430733107\TIM\WinTemp\RichOle\67PJQOIO_XS[VF@UJG%~[N1.png

意思为在地址127上的客户端访问本机（223）上的10002端口 让其访问服务器所在120的6666端口。

若不设定 –s 参数将接收所有ip的访问

2.2 书写POSTROUTING链路规则：

iptables -t nat -A POSTROUTING -p tcp -d目标地址ip --dport目标端口 -j MASQUERADE

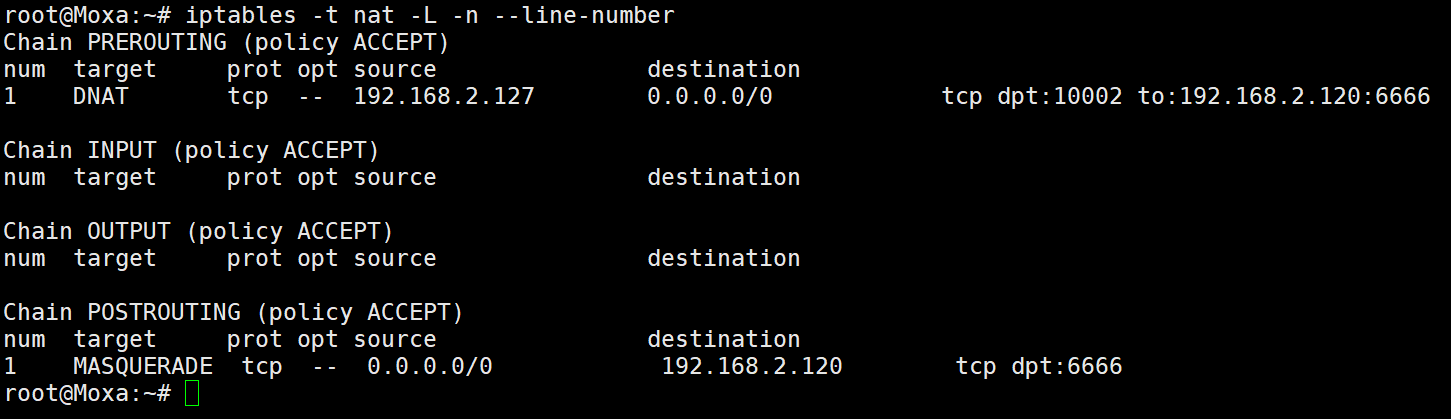
如图

C:\Users\14307\AppData\Roaming\Tencent\Users\1430733107\TIM\WinTemp\RichOle\8FU``37STOEP_K6BR`L[19A.png

意思为将访问120上6666端口的源地址换成本机地址

MASQUERADE 这个设定值就是IP伪装成为出口网卡的IP！不管现在eth0的出口获得了怎样的动态ip，MASQUERADE会自动读取eth0现在的ip地址然后做SNAT出去，这样就实现了很好的动态SNAT地址转换。

输入完成后可以通过上述查看命令 iptables -t nat -L -n --line-number 对已有规则进行查看 如图所示



1. ftp转发配置

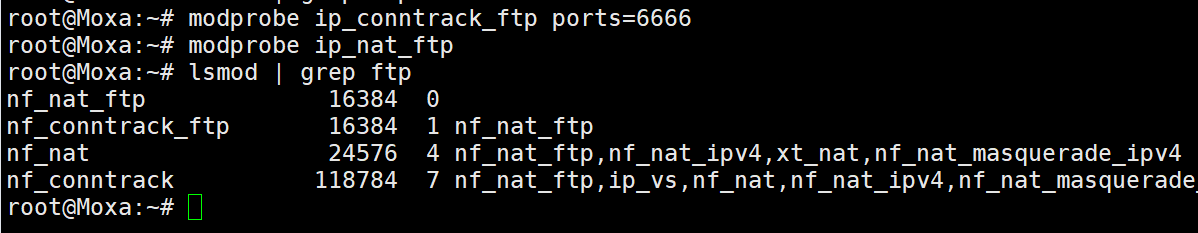
ftp传输方式与sftp不同，使用两条链路，一条为连接控制，另一条为数据传输。所以这时候就要用到两个iptables自带的ftp专用追踪和转发模块：ip\_conntrack\_ftp和ip\_nat\_ftp，否则可能出现可以连接却无法获得目录进行文件传输的情况。

1. 首先开启nat转发功能 命令行输入 sysctl -w net.ipv4.ip\_forward=1
2. 启动上述两个模块 命令行输入

modprobe ip\_conntrack\_ftp ports=服务器所在端口（不加ports 默认21）

modprobe ip\_nat\_ftp

可以使用 lsmod | grep ftp 对模块的使用进行查看以及modprobe ip\_nat\_ftp –r将模块关闭清空来重新设置服务器端口，如图



1. 分别对PREROUTING 和 POSTROUTING 链路进行规则配置，与上面的书写规则相同。

有一点需要注意：

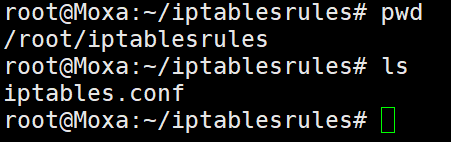
ftp连接有主动和被动模式选择，通过客户端决定。在综合各种安全性等因素下主动模式可用性并不高，一般使用被动模式，否则可能由于存在防火墙使连接不成功。

使用被动模式，可以在服务器端指定数据传输端口范围，便于设定防火墙规则使端口开启。一般情况下我们的Linux并没有对端口进行封闭，所以可以正常使用。

1. 规则保存与自启

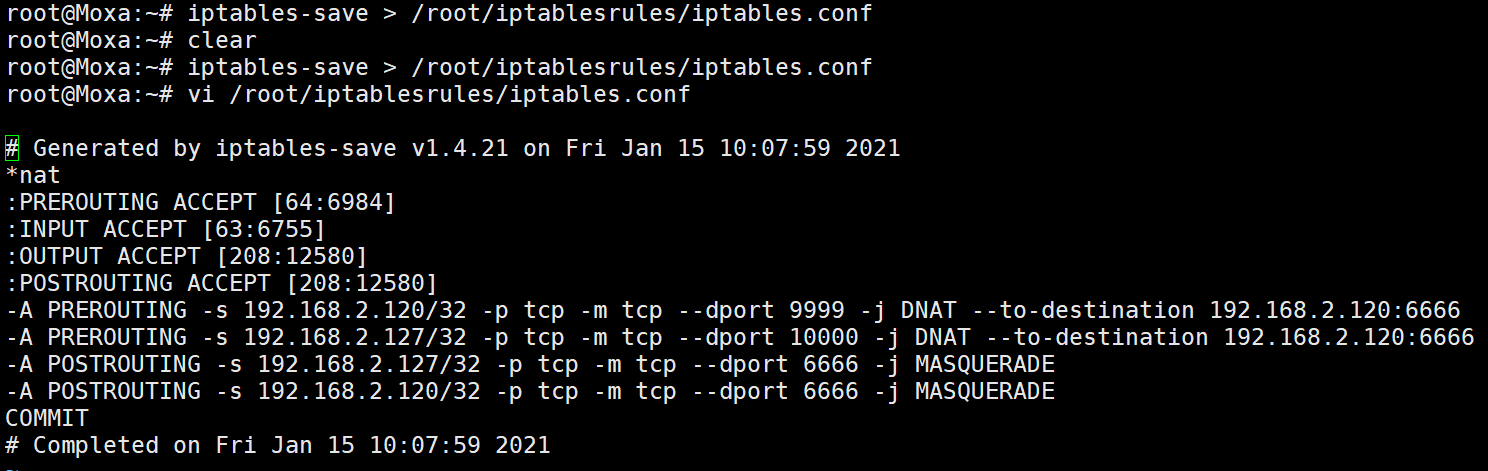
上述规则，模块的开启等都会在重启后清空，所以需要将规则保存并自启来减少维护。

iptables的工作方式就是去读取你所书写的规则文件，但是在Debian或Ubuntu等系统中是没有默认iptables的规则文件的，所以需要自己创建。



如图，我在root下创建了名为iptablesrules 的文件夹用来存放编写的规则。

创建完成后使用命令行 iptables-save > /你创建的文件夹/iptables.conf 进行保存，之后便可以使用 iptables-restore < /你创建的文件夹/iptables.conf 来对编辑的规则进行恢复。



如图，保存后可以对文件进行查看，发现已经在文件中保存。

最后添加开机启动项就可以完成iptables的开机直接使用。 使用开机启动脚本/etc/rc.local

使用命令 vim /etc/rc.local 在里面添加 iptables-restore < /你创建的文件夹/iptables.conf，并将上述模块开启和转发开启命令也一并放入。

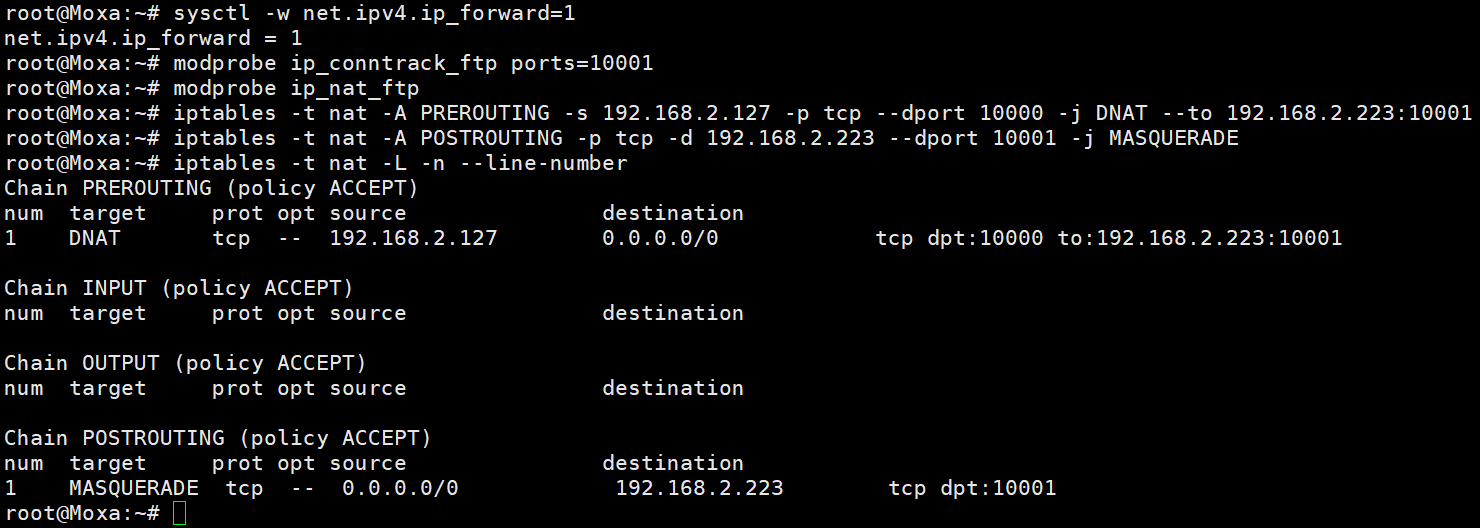
如图



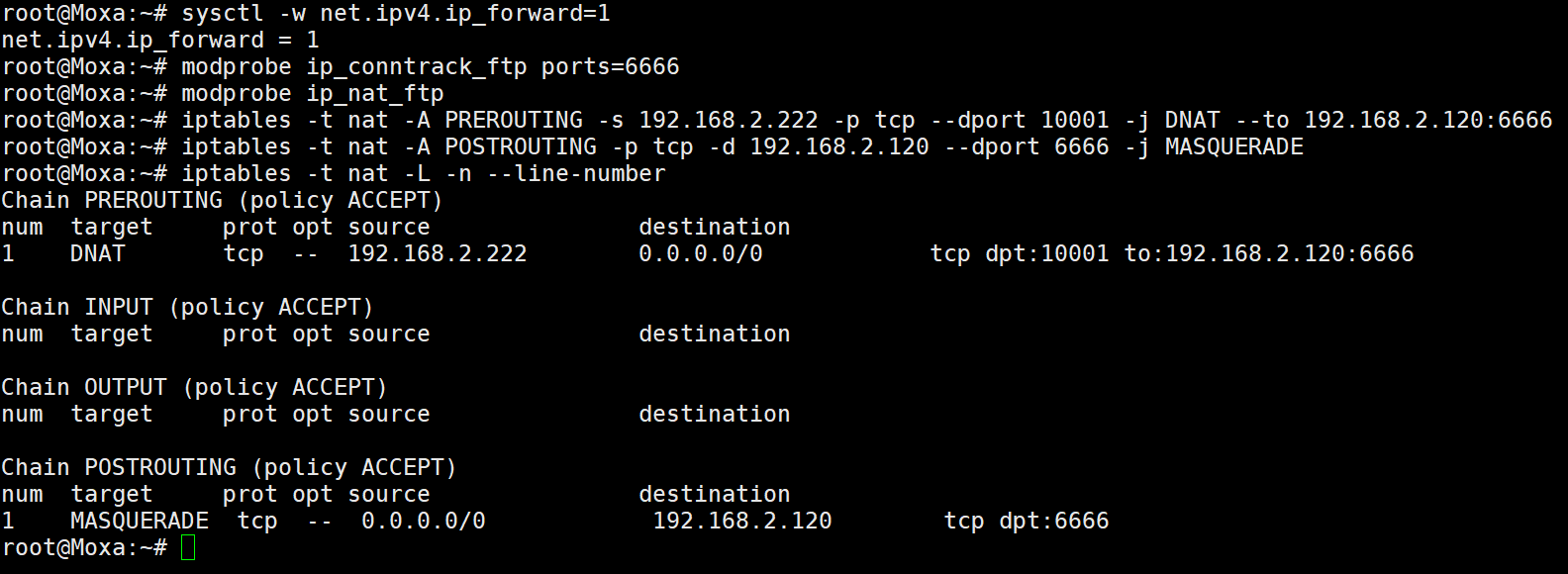
1. 实际例子演示

使用127上的客户端通过222和223连接到120上的服务器，整体思路为客户端使用被动模式连接222的10000端口，222收到后转发到223的10001端口再由223转到120的服务器6666连接端口，登陆成功后使用服务器设置的8888（可定义为一定范围内随机）端口进行数据传输。

首先在222上设置iptables规则如图

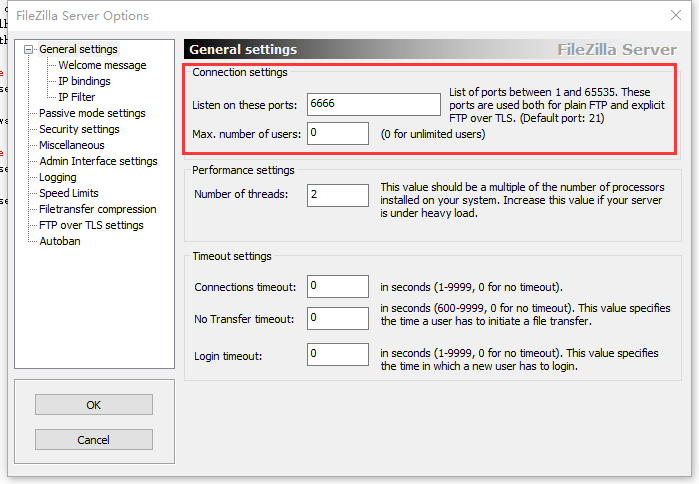


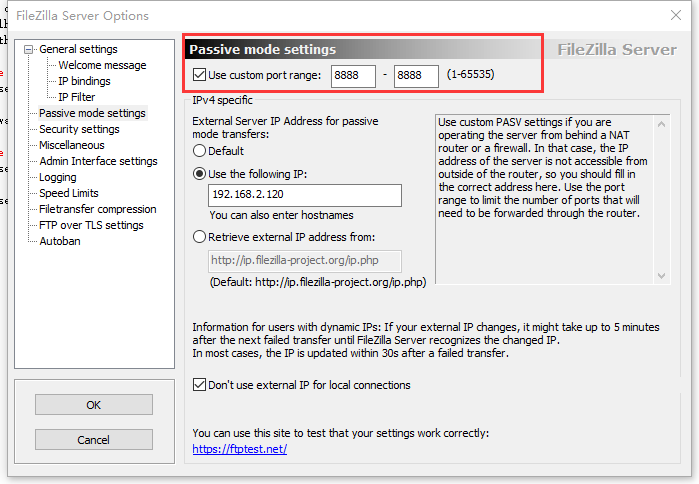
然后在223上设置如图



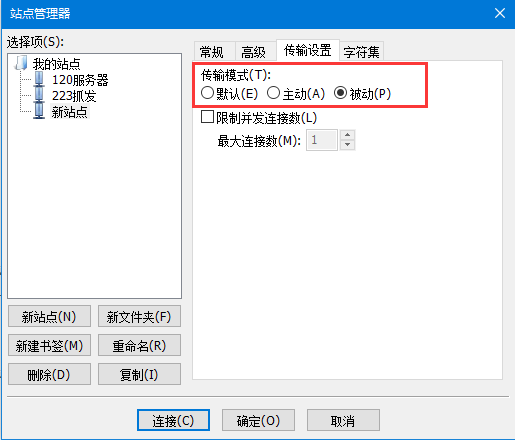
服务器设置

设定连接端口为6666，被动数据传输端口8888（可定义范围内随机）



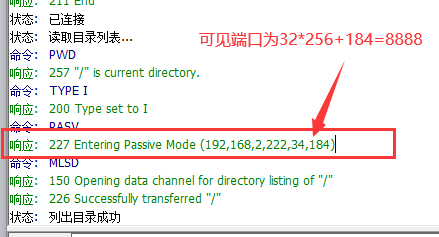


使用127上的客户端进行连接

最后连接成功

客户端显示



服务器显示

