Shadowsocks和服务器

服务器搭建SS.............................................................................................................2

手机端影俊SS操作....................................................................................................7

如何查看有多少人客户端连接了你的ss服务器...................................................8

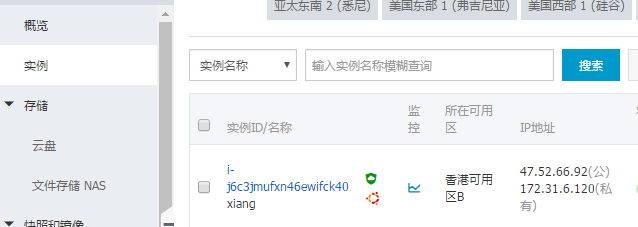
你买的VPS服务器网速测试.....................................................................................8

使用iptables建立防火墙..........................................................................................9

服务器查看eth0网口实时流量................................................................................17

Iptables给端口限速...................................................................................................18

1.购买阿里云服务器

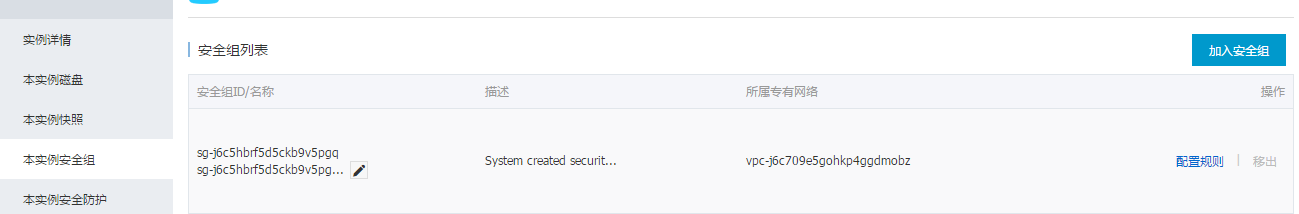


点击实例



点击管理查看端口

客户端SS软件登陆公网IP，不要搞错了搞成私网



点击配置规则

选择本地实例安全组

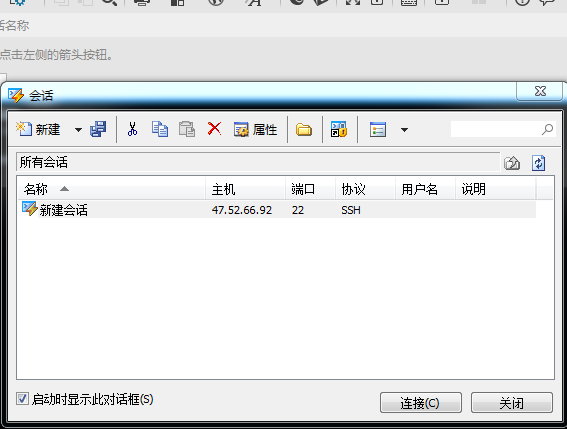


端口号88/88，意思不是服务器端口8888，而是客户端SS可以访问的端口是88号，如果你ss软件有事无事想换起换来的访问端口，你可以设置88/100，那么你客户端ss软件就可以访问88到100的12个端口其中一个，还有端口号要设置越大越好，小端口很容易被linux自己系统占用，到时候你端口就重了

阿里云的端口和IP地址我们搞懂了。现在用ssh登陆上阿里云服务器

我这次用的Xshell软件的ssh

创建连接

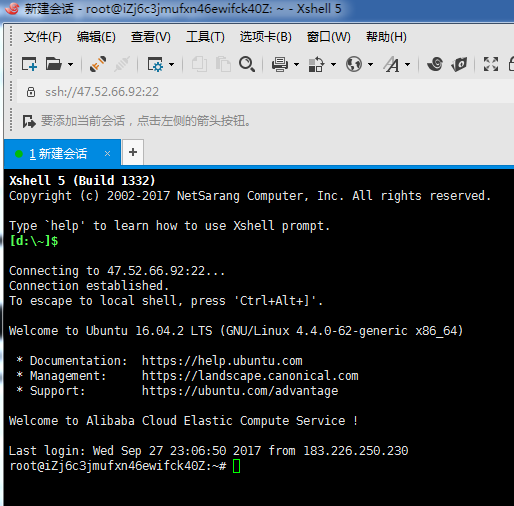


我的服务器公网IP

点击连接

用户名写root不要写我创建阿里云的用户名，因为linux默认是认为root就等于我阿里的用户名。

密码就是你创建用户时候的密码



我的电脑ssh正常连接香港服务器是有IP显示的，但是如果断开连接，这里的IP会取消

这就是成功登陆你在香港的阿里云服务器

apt-get **install** python-pip

pip **install** shadowsocks

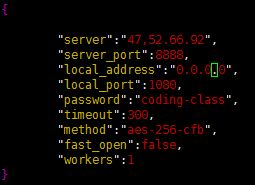
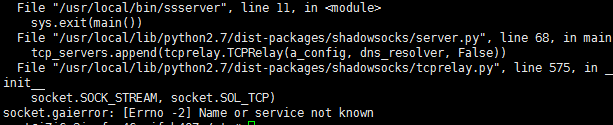
有时 Ubuntu 会遇到第一个命令安装 python-pip 时找不到包的情况。pip 官方给出了一个安装脚本，可以自动安装 pip。先下载脚本，然后执行即可：

wget https:*//bootstrap.pypa.io/get-pip.py* python **get**-pip.py

但是我没有遇到，因为阿里云给你安装了很多安装pip之前需要安装的东西，但是其他公司服务器就不一定咯

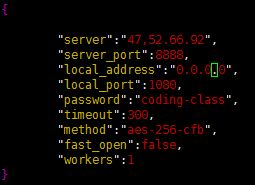
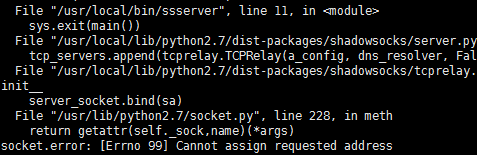
vi /etc/shadowsocks.json

新建一个json文件

逗号

这个socket.gaierror错误是因为你的josn文件sever ip地址里面有逗号

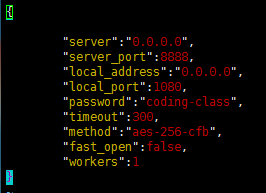
IP问题

这个socket.error问题在有些服务器上面是不会出问题的，在这里出问题是因为阿里云对IP做了手脚，所以你在josn文件的server段不能填服务器的公网ip，要填0.0.0.0

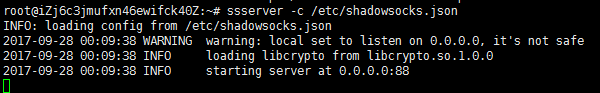
不知道是不是这个问题



修改成0.0.0.0就可以了

所以你填这个就没有问题了

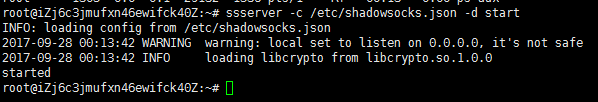
ssserver -c /etc/shadowsocks.json 启动阿里云ss服务器



这就是正常启动，我这里是前台打印

ssserver -c /etc/shadowsocks.json –d start 这是后台启动

ssserver -**c** /etc/shadowsocks.**json** -**d** **stop** 这是停止后台ss服务器

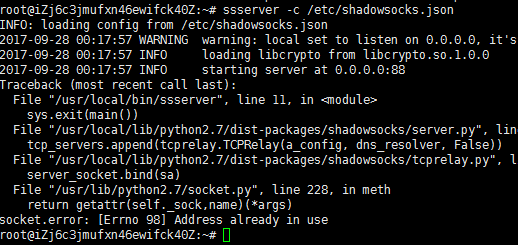


SS服务器启动后你可以用ps aux来查看



如果哪天服务器ss程序崩溃了，你就用ps aux查看有没有ss这个服务在运行

前面josn文件有个问题就是端口号是88而不是8888，我没来得及修改，因为阿里服务器是写的88/88端口



这个问题是我的端口已经被后台启动的ss服务占用了，如果你再启动ss服务，是会报错的[Errno 98]

除非你把ss服务器关了，或者你修改josn文件里面的端口

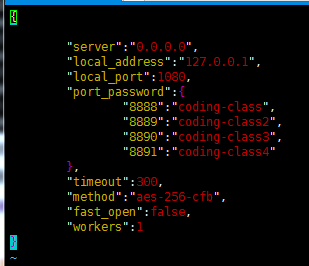
虽然SS单个端口可以实行多人连接使用但是我想用多个端口实行多人使用，然给每个人的端口配置密码

首先关闭ss服务器进程



用kill -9 16472 将其杀死，或者其它什么命令杀死进程

打开SS的json文件



将port\_password改成这种格式

其余不变

取消server\_port

4个端口8888端口对应的密码是coding-class。8889端口对应的密码是coding-class2，其余端口也是这样设置，还可以增加很多端口，但是你要保证你的阿里云端口范围是增加了的。



这是我阿里云端口修改的范围，从8888~9000端口都可以使用。

**客户端ss配置，我们以手机为例，电脑那个ss再另外一个文档里面**

打开传说中的影俊



安装josn来

安装josn来

随意，一般写1080

我的阿里云服务器端口88/88

我的阿里云公网IP

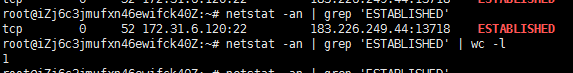
现在你可以正常使用ss与国际接轨了。

如何查询有多少人的客户端连接了你的SS服务器

netstat -an | grep 'ESTABLISHED' 用这个命令来查询客户端连接数量

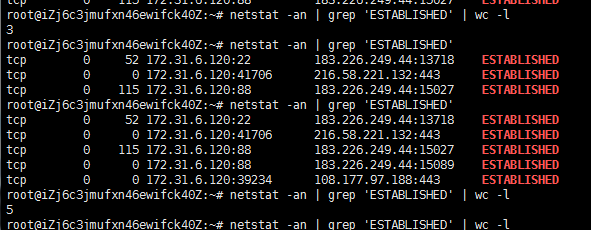


我启动了自己本地电脑的ss代理功能



但是连接数还是1，这是为什么呢？

然后我打开了一个网页



连接数从3到5再到3来回变化，所以这个只能证明有个人连接了我的SS服务器

但是这个netstat命令是用来检测端口的，我的端口是88号，并不是检测我的ss服务器本身，但是我可以通过端口来查看有没有人连接我的服务器。但是无法获取确定的人数，所以只能知道有人连接你的ss

你买的服务器的网速测试

用wget下载工具下载speedtest-cli命令软件

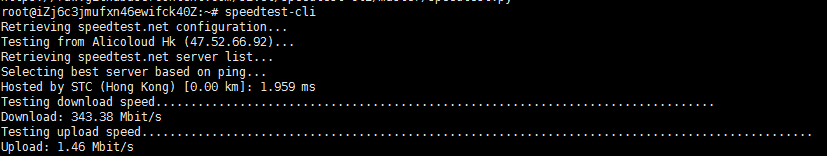
wget <https://raw.github.com/sivel/speedtest-cli/master/speedtest_cli.py>

chmod a+rx speedtest\_cli.py

mv speedtest\_cli.py /usr/local/bin/speedtest-cli

chown root:root /usr/local/bin/speedtest-cli

直接执行[root@localhost temp]# speedtest-cli



如果执行speedtest-cli失败,可能还缺少脚本什么的

wget [https:*//github.com/sivel/speedtest-cli/archive/master.zip*](https://github.com/sivel/speedtest-cli/archive/master.zip)

unzip master.zip

我解压失败，但是再去执行speedtest-cli居然成功了

使用iptables建立防火墙

包过滤防火墙：就是对端口进行过滤，比如其他端口数据包不能过我服务器，80端口允许过我的服务器。

应用层防火墙：针对同一个端口，检查接受到的数据包url地址里面的内容，如果是敏感的url地址，就把它过滤掉，这个和包过滤防火墙的不同在于，包过滤防火墙一杆子全部打死，应用层防火墙选择性的打死。

网络走向图：

用户空间

linux内核

这是我的本地主机服务器

1

2

3

用户电脑

eth0

eth1

1.用户电脑访问服务器

2.服务器收到用户数据包，先确认这个数据包是发往给本服务器网站的数据包，还是要给eth1网口进行转发给其它服务器

3.如果是发送给本服务器网站的数据包，那么eth0就会把数据包发给1号接口

用户电脑

eth1

eth0

用户空间

3

2

1

linux内核

这是我的本地主机

4.本地主机服务器收到用户电脑的数据包后，要用接口2给用户电脑发送一个回复

linux内核

这是我的本地服务器主机

1

2

3

用户空间

eth0

eth1

用户电脑

还有可能就是服务器把接受到的数据包又发给了另外一个用户电脑

linux内核

这是我的本地服务器主机

1

2

3

用户空间

eth0

eth1

用户电脑1

用户电脑1

5.用户电脑给我的服务器发送数据包，但是接口3发现数据包的IP地址不是指定的我的服务器ip地址，是另外一条服务器的IP

linux内核

这是我的本地服务器主机

1

2

3

用户空间

eth0

eth1

用户电脑

6.接口3将数据包转发给eth1网口转发出去，数据就不会进入我的服务器用户空间应用程序

所以我们用iptables设置防火墙规则，其实就是对接口1，2，3进行设置

Iptables规则设置也可以设置服务器的http程序或者ssh程序，或者其他程序能不能向2号接口外的用户电脑发数据包

Iptables规则设置就是设置这个接口

用户电脑发过来的数据包允不允许过1号接口给http程序接收，或者 是ssh程序接收，或者是服务器其他程序是否能接收

linux内核

这是我的本地服务器主机

1

2

3

用户空间

http程序 ssh程序

eth0

eth1

用户电脑

这是我们用户电脑客户端对指定服务器点对点，相互传输的规则设置方法。

如果我们电脑连接上自己的服务器，但是我电脑要求访问的地址是另外一台网络的服务器地址，那么这里就要用NET来设置路由转发功能，数据流向如下：

本地服务器/路由器

本地用户的电脑

地址192.168.1.1

要访问外网地址200.200.200.1

另外一个网站服务器的地址

200.200.200.1

对外网的地址是200.200.200.2

对内地址是192.168.1.2

我这台本地电脑的网线是接在192.168.1.2的服务器上面的，所以本地服务器接收到的数据包地址来源于192.168.1.1

本地服务器将192.168.1.1这个地址的电脑数据包发送给200.200.200.1是没有问题的，但是200.200.200.1服务器收到数据包后要返回一个回应数据给192.168.1.1的电脑，但是我发现这里不对，因为最先接受外网数据包的是本地服务器的公网地址是200.200.200.2，所以服务器接收不到外网服务器的回复数据，解决方法是在本地用户电脑发送数据给本地服务器之后，本地服务器要用NET工具将192.168.1.1转换成200.200.200.2，这样外网回复的时候就指定200.200.200.2这个地址，然后本地服务器在转发给内部本地用户电脑

所以要服务器做转发数据包的话，这里eth0还多了一个接口4，接口4设置服务器路由之前的规则

本地服务器

1

2

3

用户空间

http程序 ssh程序

eth0

eth1

用户电脑

还有一个接口5是服务器做完转发以后的规则设置

接口4

接口5

所以根据以上网络路径的示意图，我们知道了每个接口都是可以设置规则的，只是规则的意义不一样。

1号接口：设置服务器入口规则

2号接口：设置服务器出口规则

链名表示你设置的规则是针对哪一条链

这里是INPUT链

这里是往filter表里面写

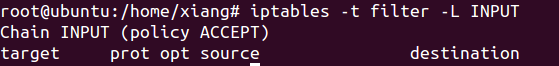
-t 是表名，表示你要往哪个表里面去写

3号接口：设置服务器转发规则

4号接口：设置路由前规则

5号接口：设置路由后规则

下面我们来操作一下iptables防火墙



这条命令就是查看filter表里面INPUT链有没有设置什么规则

-L查看规则

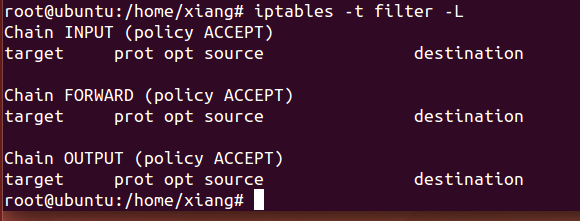
表有四种：raw raw表里面有PREROUTING链，OUTPUT。

mangle

nat

filter filter表里面有 INPUT链 , FORWARD链 ，OUTPUT链

我们上面就是向filter表里面写入INPUT链，记住每个表都有固定的几个链可以选择，但是raw表不能选择nat表里面的链



-j表示如果数据包是tcp协议，是接受还是不接受

我们查看了filter到底有哪些链可以设置规则。和上面一样只有三个链可以操作



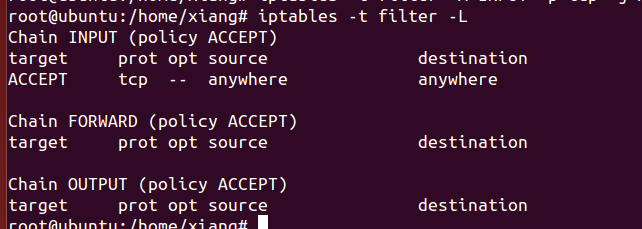
-p tcp 表示想INPUT链写入tcp规则

-A 表示向某个链写入规则

ACCEPT表示接受

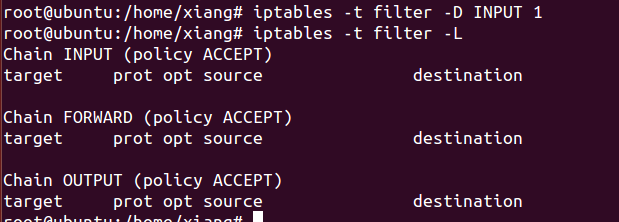
这里是向INPUT链

这条规则的意思就是向filter表里面的INPUT链写入规则，规则是对tcp发来的数据进行接收



你再查看filter表INPUT链就多了一条规则

前面查看的时候是没有任何规则。

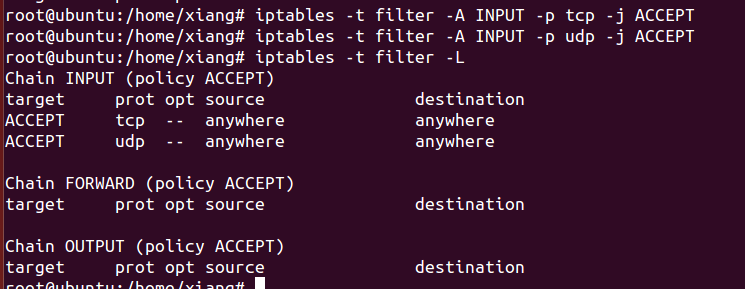


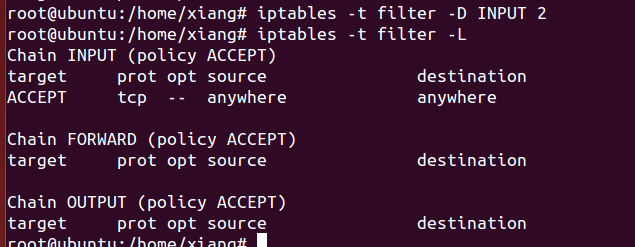
删除INPUT链那一条呢？

这里是第1条

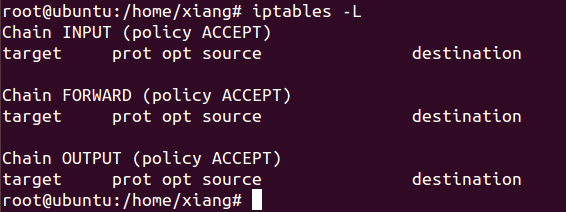
-D指定删除filter表INPUT链的规则

我们下面多加两条规则试试



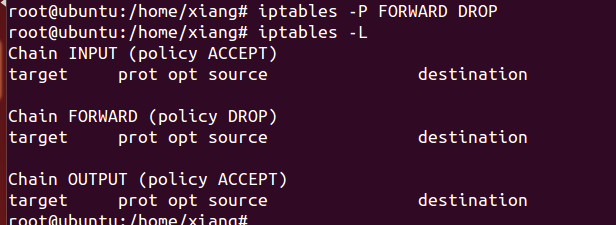


你看我INPUT 后面跟的2 就是删除INPUT链第2条规则，-D就是删除规则的意思



Ubuntu系统开机后防火墙默认规则都是ACCEPT

比如我想修改filter表里面的FORWARD链数据包转发的默认规则。



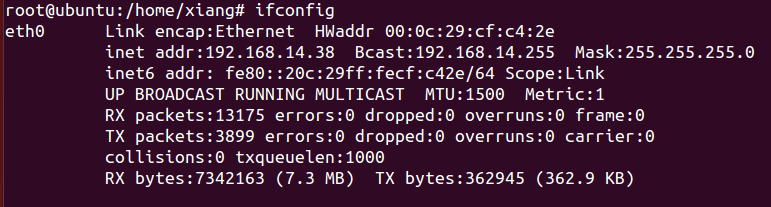
你发现默认规则修改了

DROP 扔掉

-P 修改默认规则

这句话就是将FORWARD链里面允许转发数据包的功能，改成丢掉转发的数据包。这样FORWARD就不具备转发数据包的功能了。

我们下面来设置一系列防火墙功能



主机IP

我主机的IP地址是192.168.14.38



-s 是指定哪个IP地址的主机连接我的主机

这是开发板上面看到的主机内容

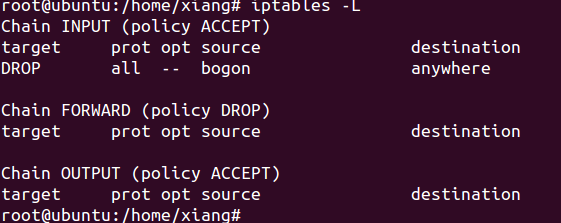
开发板用ssh连接我主机

我开发板用ssh连接我主机成功



我给主机加了一条防火墙规则

意思是IP地址 为192.168.14.44的客户机，不准连接我主机，DROP就是丢掉192.168.14.44机器的数据包

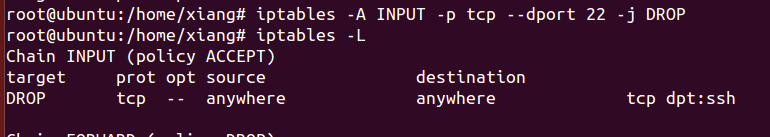


已经增加了一条规则



我开发板想再用ssh连接我主机就连接不上了。

 iptables –F 是清楚掉所有规则



-p 是指协议

Tcp协议

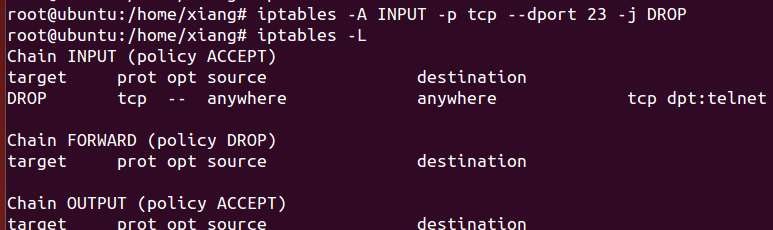
任意地址

指定端口

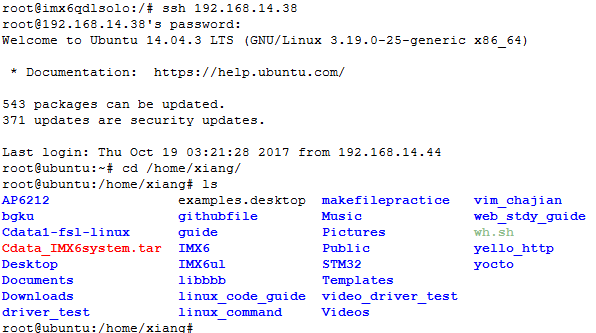
这句话的意思就是任意地址的tcp协议数据包，发送到我主机22号端口，都将会被丢弃。

你看我开发板还是连接不上主机，因为ssh端口就是22

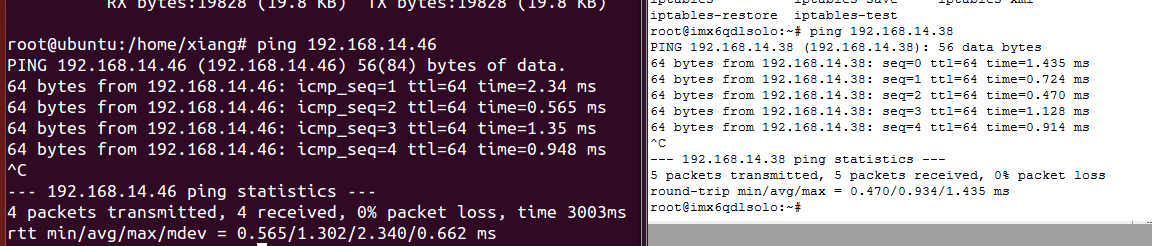
我下面修改一下



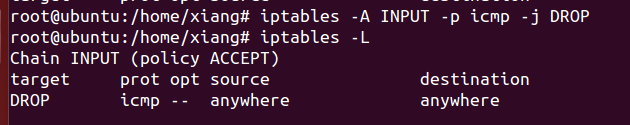
我把端口改成23号

 你看我开发板就连接上了。

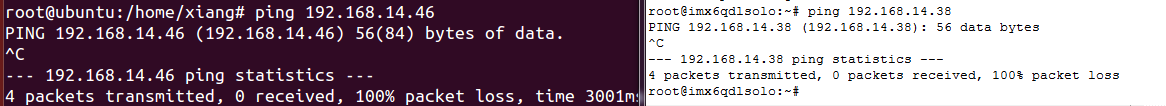
Iptables 多条匹配规则使用



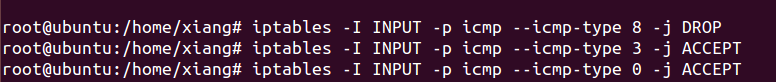
现在我两边的主机都能相互ping通，我现在不用端口规则和IP地址规则，我们用协议规则来做防火墙



这句表示所有icmp协议的数据包来到我主机，我主机都把他们丢弃



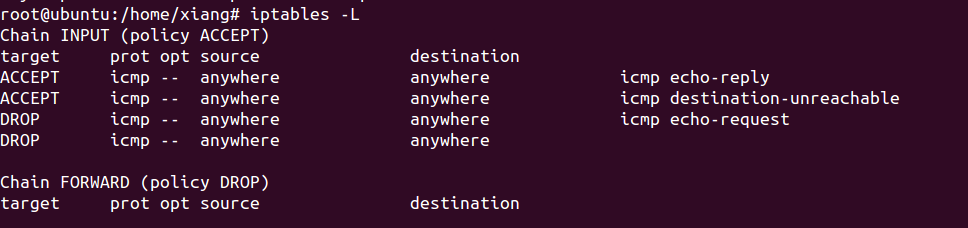
开发板ping主机平不通是因为主机设置了icmp协议丢弃，但是主机怎么会ping不通开发板呢？开发板上面什么防火墙都没有，其实主机是ping通了开发板的，只是开发板返回的包被主机icm协议防火墙拦截了



我在原来的iptables规则上又加了新的规则



你看主机就可以ping通开发板了，开发板ping不通主机是正常的，因为我主机有防火墙。



这句的意思是其它主机发给我的请求全部丢掉

这是最先加的一条规则，意思是拒绝所有icmp数据

这就是允许我去ping别人，但是不允许别人ping我的规则

我ping其他主机，不可接受其他主机的应答，我这里要求放行，就是可以接受其他主机的数据包回复

所以这个规则就是先一杆子打死

然后放行一些东西。

**服务器查看网卡实时流量**

apt-get install ifstat

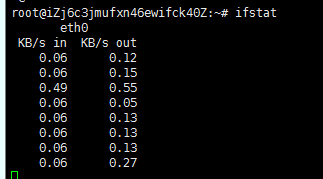
报错：Unable to locate package ifstat

你要更新下apt-get源

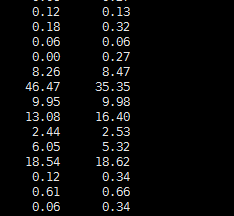
执行sudo apt-get update

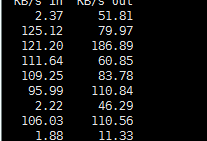
再安装apt-get install ifstat

问题解决



这是SS没连接查看视频的流量

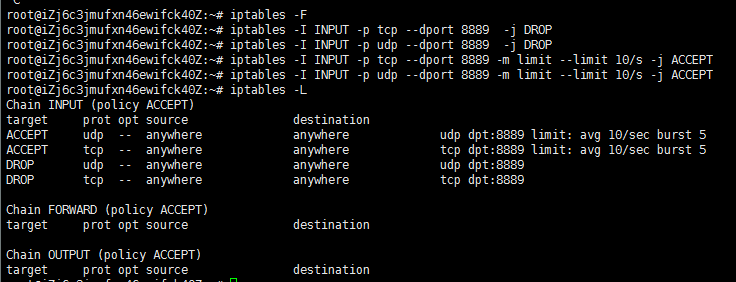
这是SS连接上我的服务器了产生的连接流量



这是看twiter产生的流量，看直播视频流量就起来了。

限速前先将端口全部设置成丢包

现在我用iptables给端口限速试试



然后再选择性限速

iptables -I INPUT 是对下载做设置  
iptables -I OUTPUT 是对上传做设置  
在ubuntu论坛查到的资料说限速的时候规则是成对使用的，第一条是-j ACCEPT，接受符合规则的数据包，第二条是-j DROP抛弃不符合的数据包，有错误欢迎指正。  
限速的时候--limit 60/s 根据查到的资料说是这个“60”不是直接的速度而是数据包的数量，60/s是指每秒转发60个数据包，每个数据包是1.5Kbyte（貌似有人说是1480byte？），所以限制60/s的时候，被限制的IP平均速度是90K/s的样子