网络与信息安全课内实验2

Host文件与DNS投毒

### 本次实验目的

1. 了解 host 文件和 DNS 系统的关系。

2. 使用 DNS 攻击工具进行 DNS 投毒，观察攻击情况，了解 DNS 攻击原理。

3. 了解资源记录并对MX资源记录类型进行观测；

4. 通过命令行直接向SMTP服务器投递邮件。

### 实验平台

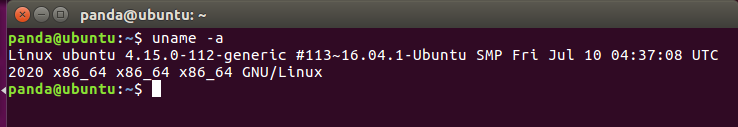
Server：ubuntu虚拟机，安装DNS服务器bind9，记为**Ubuntu服务端**（IP：192.168.8.135）

Attacker：ubuntu虚拟机，与server处于同一网段（局域网），记为**Ubuntu攻击端**（IP：192.168.8.133）

后续还会使用一台额外的Ubuntu用于单纯的发布测试，记为**Ubuntu测试机**（IP：192.168.8.136）

注意

1. 在没有特殊强调的情况下，例如dig命令使用，可以在任一虚拟机上进行
2. 本次教程实验中，Ubuntu虚拟机版本均为Ubuntu16如下图所示

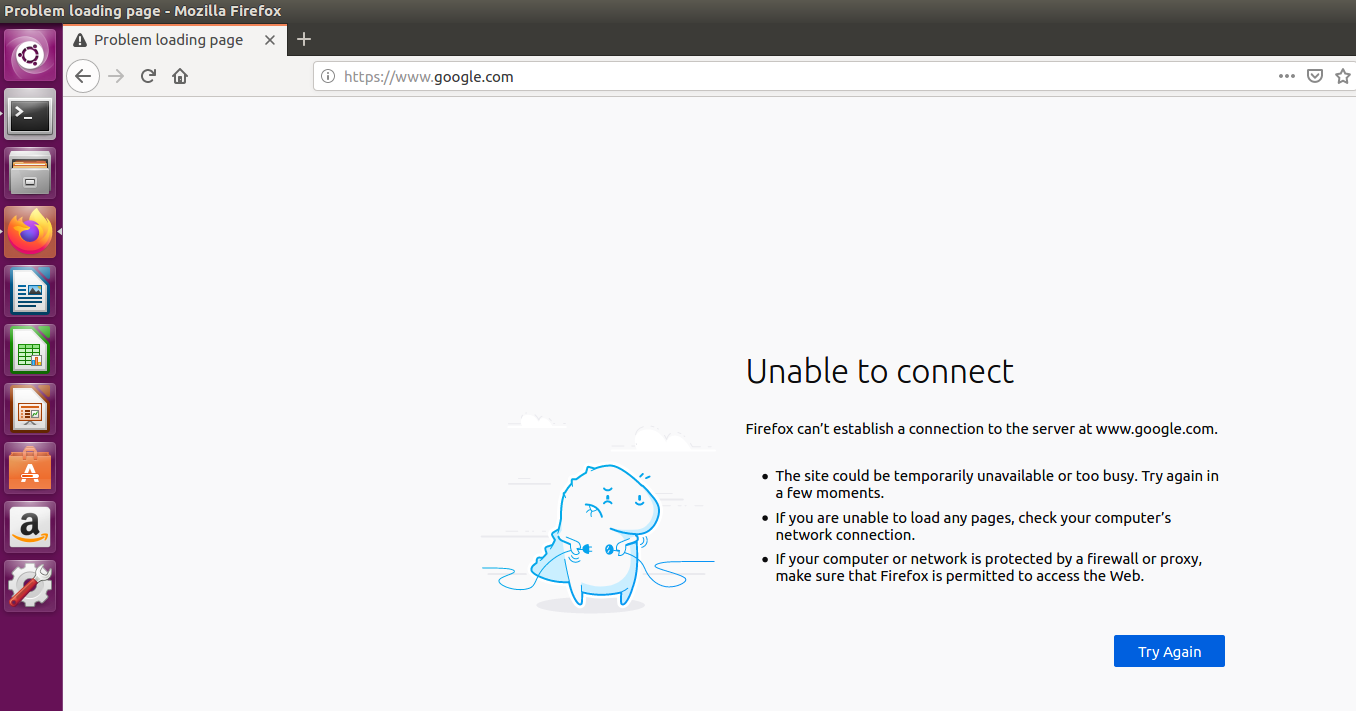


### 实验步骤

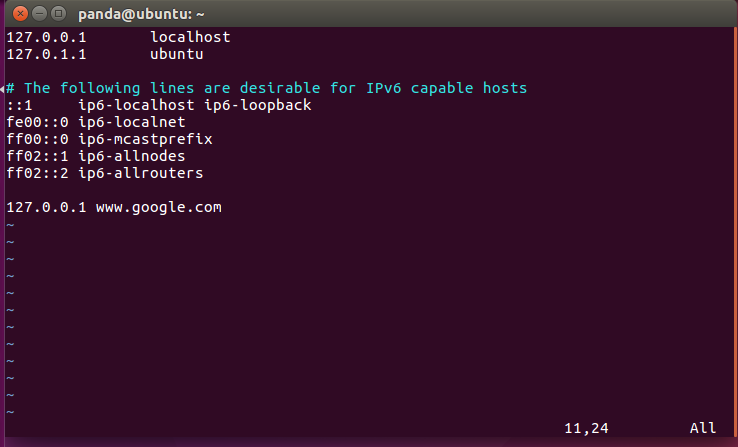
###### 了解Hosts文件

Hosts 是一个没有扩展名的系统文件，可以用记事本等工具打开，其作用就是将一些常 用的网址域名与其对应的 IP 地址建立一个关联“数据库”，当用户在浏览器中输入一个需要 登录的网址时，系统会首先自动从 Hosts 文件中寻找对应的 IP 地址，一旦找到，系统会立 即打开对应网页，如果没有找到，则系统会再将网址提交 DNS 域名解析服务器进行 IP 地址的解析。

在**ubutnu服务端**中，使用浏览器访问www.google.com，如下图所示，无法访问



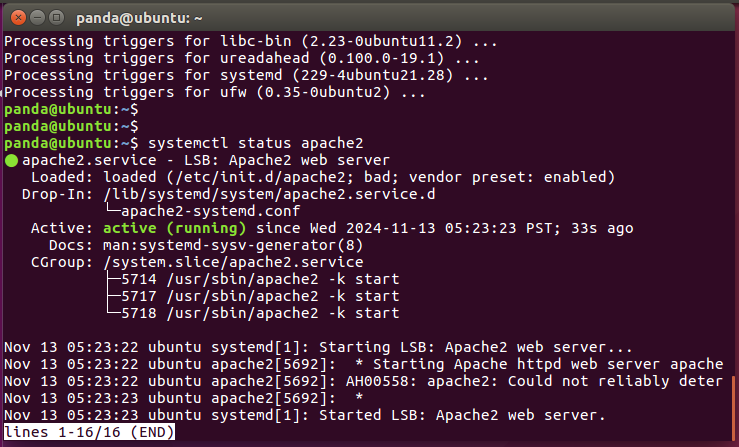
找到hosts，文件路径为在/etc/hosts，打开文件，通过vim打开（这里也可以使用其它软件进行打开，例如gedit），然后在末尾追加一行127.0.0.1 www.google.com：



在**Ubuntu服务端**安装apache2服务，安装前可以先查看一下，如下图所示，确实没有安装

sudo apt install apache2即可安装，如果遇到错误，一种简单的方法是重启虚拟机，然后再次执行安装

安装完成后，查看一下，按一下字母q键即可退出当前页面

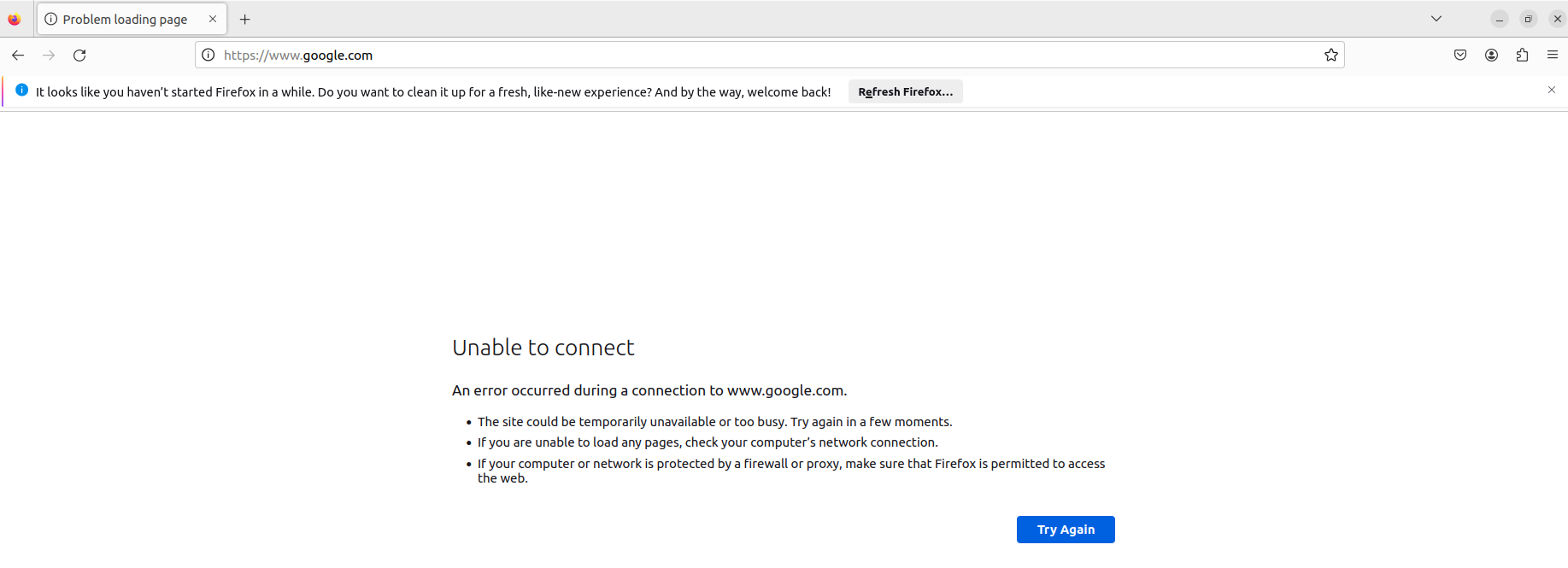


接着我们打开浏览器，输入www.google.com。由于本机之前安装了apache2服务，所以有如下效果：



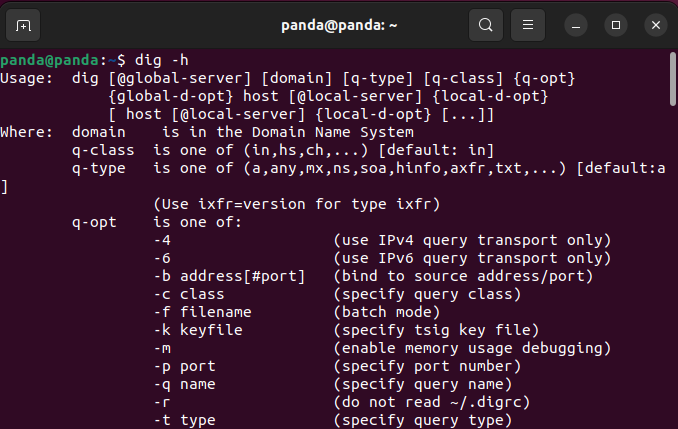
如果没有改变，可以尝试重启网络服务. /etc/init.d/networking restart

注意：如果修改完hosts文件，在浏览器中访问[www.google.com](http://www.google.com)出现如下界面，说明没有安装Apache，需要自行安装后重新访问方可得到上述结果

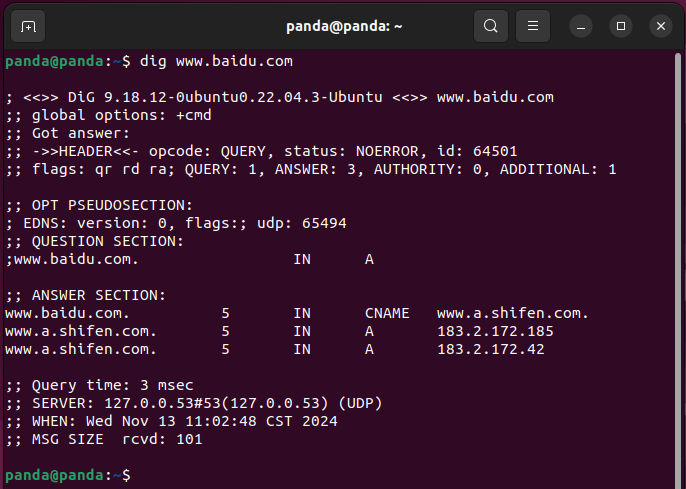


###### 2．使用 dig 工具查看网站域名解析过程

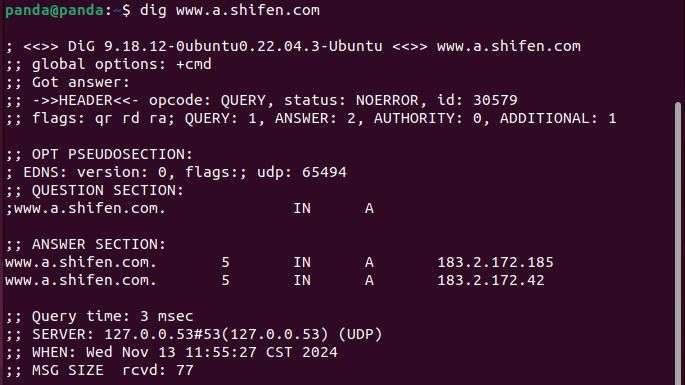
打开终端，查看 dig 命令用法：（这里只是测试一下dig的用法，用什么虚拟机都行）



使用dig名称解析www.baidu.com：

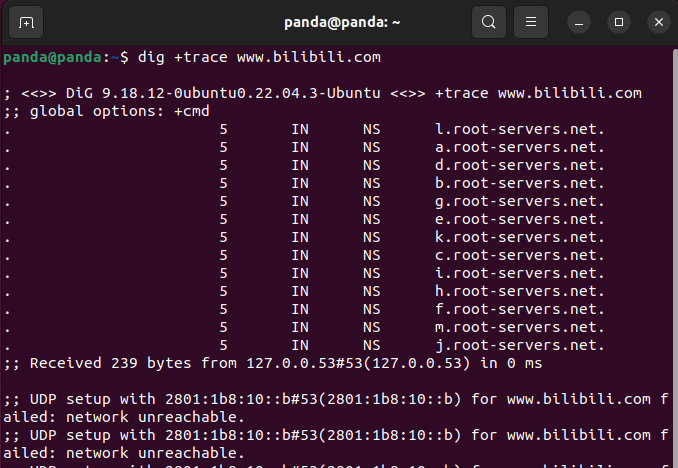


从图中可以看出，www.baidu.com 查询到了一个CNAME www.a.shifen.com。然后，继续查询www.a.shifen.com，获取最后的IP地址。

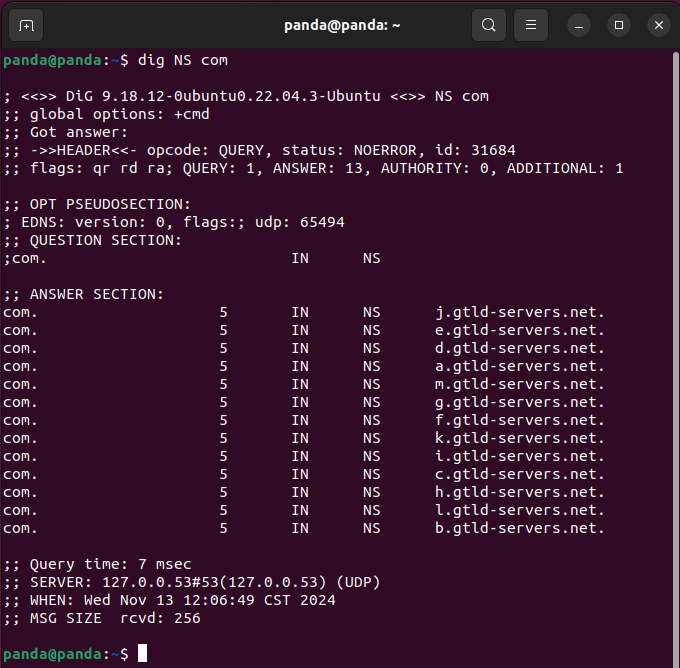


使用 dig +trace 命令，查看www.bilibili.com完整的解析过程：

第一步，查询 root 根服务器



第二步，查询.com 服务器



第三步，查询 bilibili.com 服务器



第四步，查询 www.bilibili.com 服务器地址

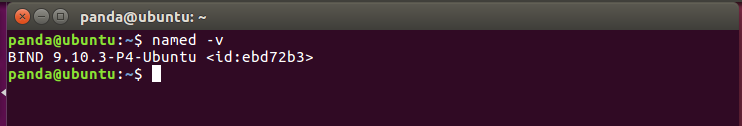


###### DNS 投毒实验

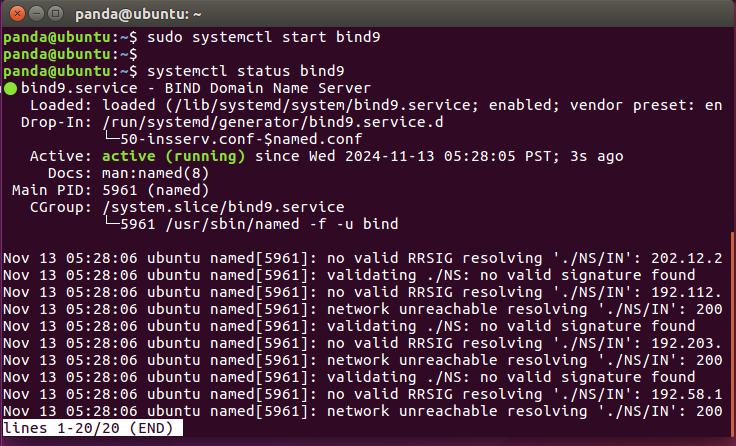
一、配置DNS服务端：

在**Ubuntu服务端**，我们使用bind9作为服务器软件。Bind9是一款开放源码的DNS服务器软件，由美国加州大学Berkeley分校开发和维护。

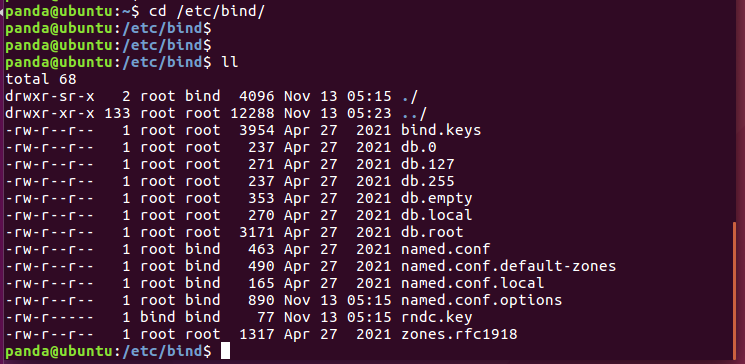
sudo apt-get install bind9



启动bind9 sudo systemctl start bind9



安装bind9后, /etc/bind/文件夹中会生成很多基础的配置文件，可以用ls /etc/bind/ -l进行查看。



关于上述文件有什么作用，这里不再一一陈述，大家可以根据需求自行查找。下面是DNS实验上一个版本中记录的一些文件及其作用，可以用作参考

named.conf  Bind主配置文件

named.conf.options全局选项

db.root   根服务器指向文件， 由Internet NIC创建和维护， 无需修改， 但是需要定期更新

db.local    localhost正向区文件，用于将名字localhost转换为本地回送IP地址 (127.0.0.1)

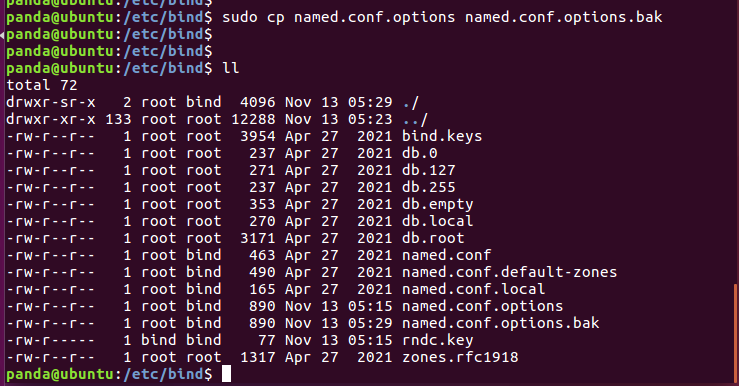
db.127    localhost反向区文件，用于将本地回送IP地址(127.0.0.1)转换为名字localhost

详见<https://blog.csdn.net/colourzjs/article/details/44491479>

1） 通过named.conf.options文件，实现修改DNS查询端口，以及关闭dnssec-validation服务。这些服务原本是用来防止DNS缓存投毒攻击的，如果不关闭的话，攻击会非常困难。注释掉named.conf.options文件中的对应条目，并且加入关闭dnssec服务的语句：

注意：如果担心操作有误，可以先将named.conf.options备份一下，后续如果有需要可以及时进行恢复

sudo cp named.conf.options named.conf.options.bak



接下来开始修改，执行sudo vim named.conf.options，内容如下图所示

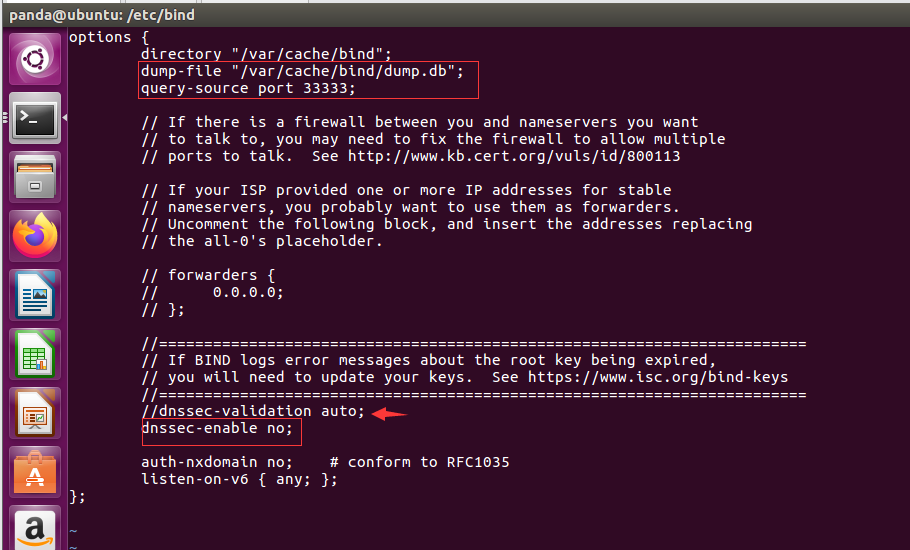
添加：

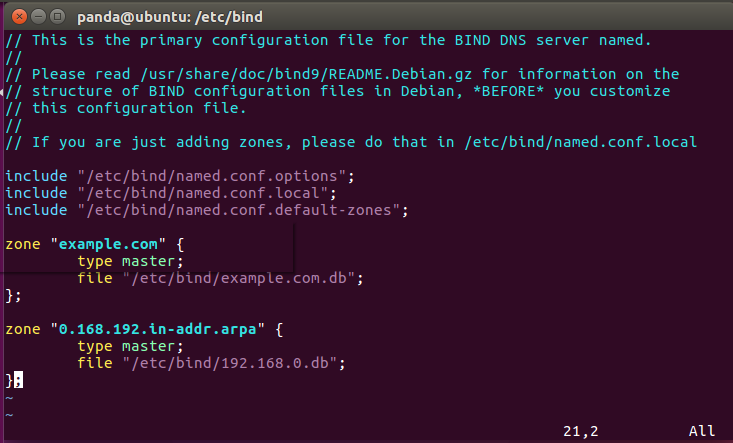
dump-file "/var/cache/bind/dump.db"; # 用于指定输出位置

query-source port 33333; # 指定发送请求包时使用的端口

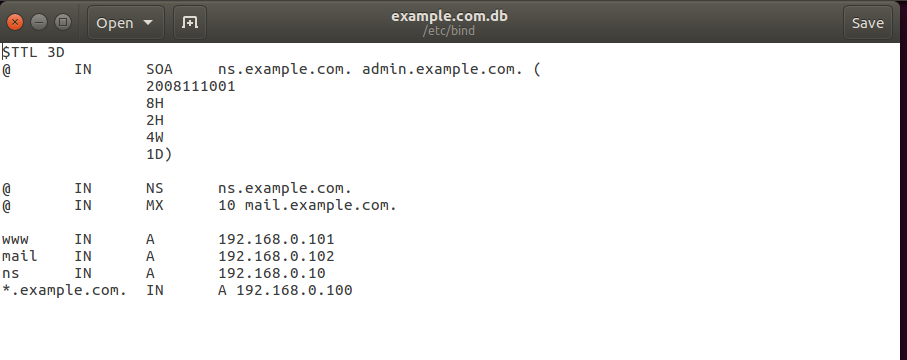
dnssec-enable no; # 禁用安全机制

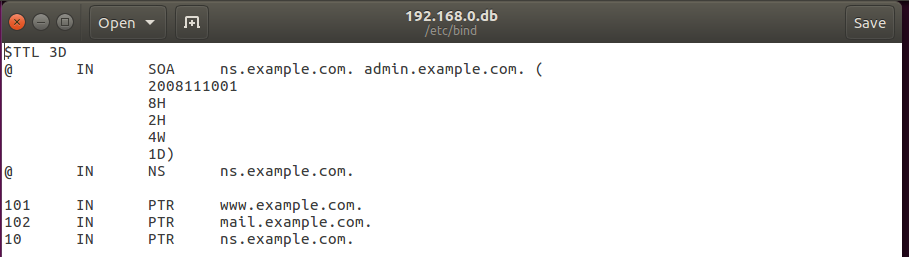
进行添加和注释语句，一共添加三句（红框部分），注释一句（箭头指向部分）





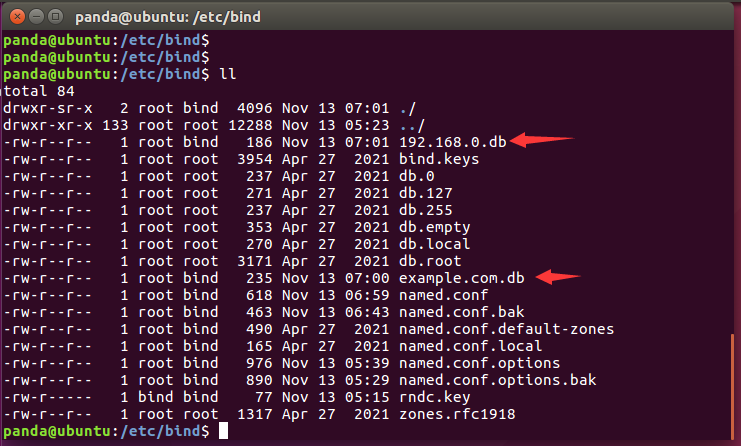
创建两个文件，建议使用gedit创建并写入，会比vim方便很多





可以在<https://blog.csdn.net/LostUnravel/article/details/127819578>

中找到并复制粘贴

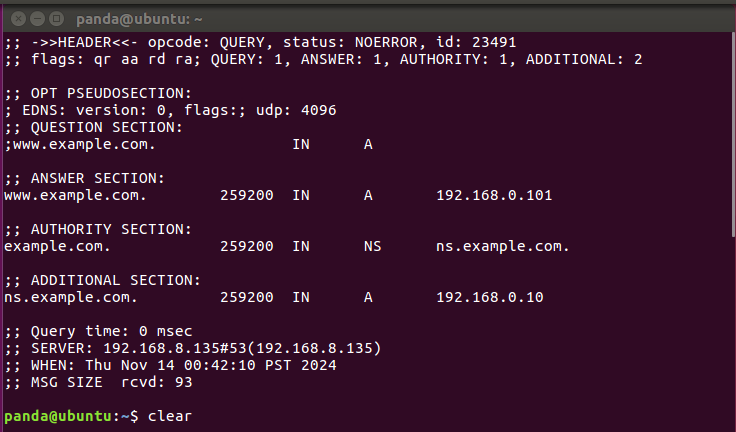


随后，重启bind9服务



在**Ubuntu攻击端**

此时，可以执行一下dig [www.example.com](http://www.example.com) ，可以看到返回的内容就是我们上述配置的内容

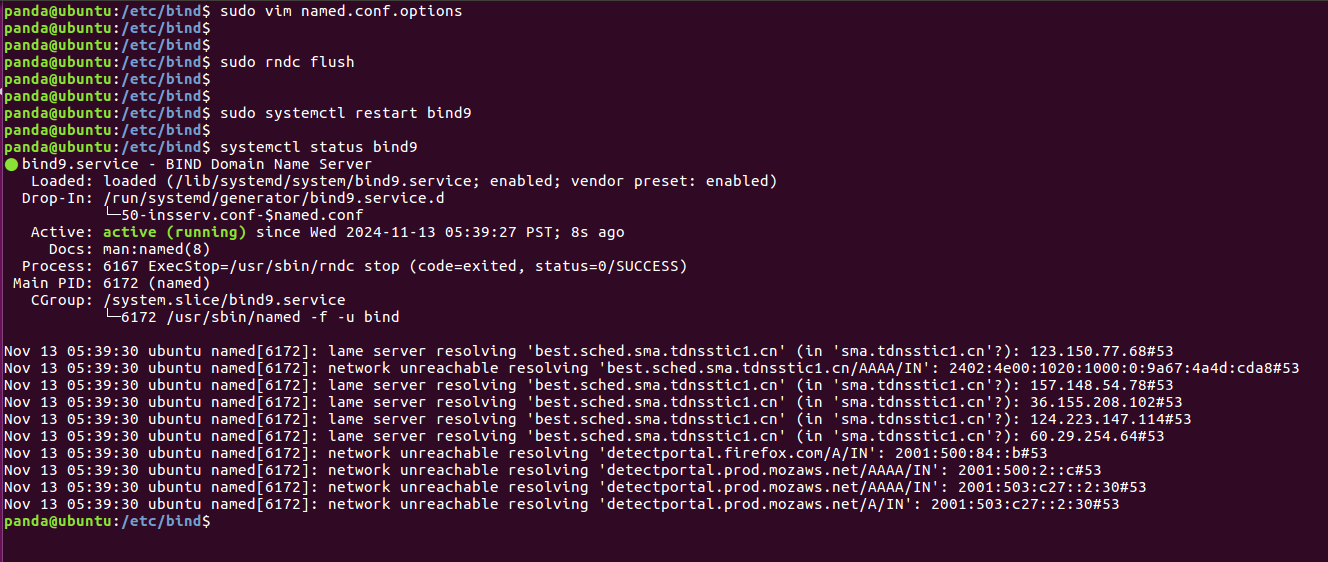


刷新DNS缓存，然后重启DNS服务器，将DNS数据导出并查看初始状态

sudo rndc flush

sudo systemctl restart bind9

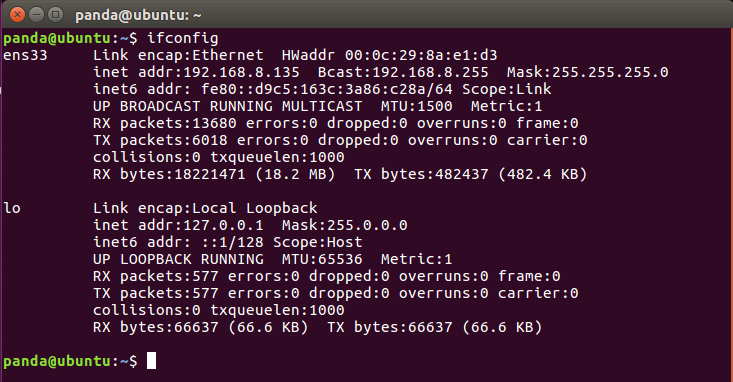
systemctl status bind9



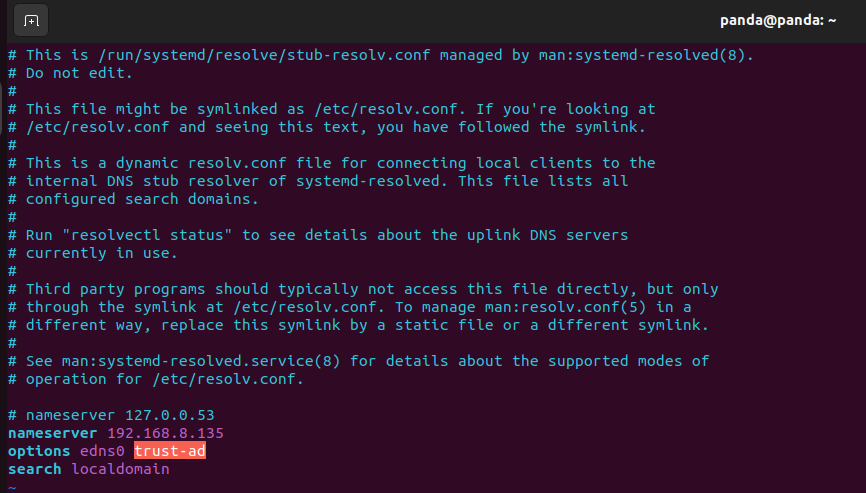
二、配置**Ubuntu攻击端**

配置**Ubuntu攻击端**的目的是使自己的DNS服务IP地址变为设置好的bind9 server的地址。

先在**Ubuntu服务端**进行IP地址查询

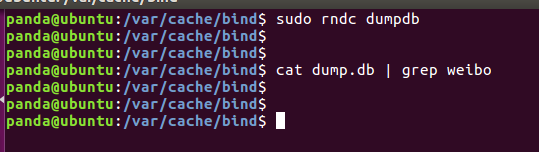


进入**Ubuntu客户端**，打开sudo vim /etc/resolv.conf，将nameserver修改为**Ubuntu服务端**的的ip地址。



以weibo网站为例，先在**Ubuntu服务端**查看存储文件，可以看到是没有内容的

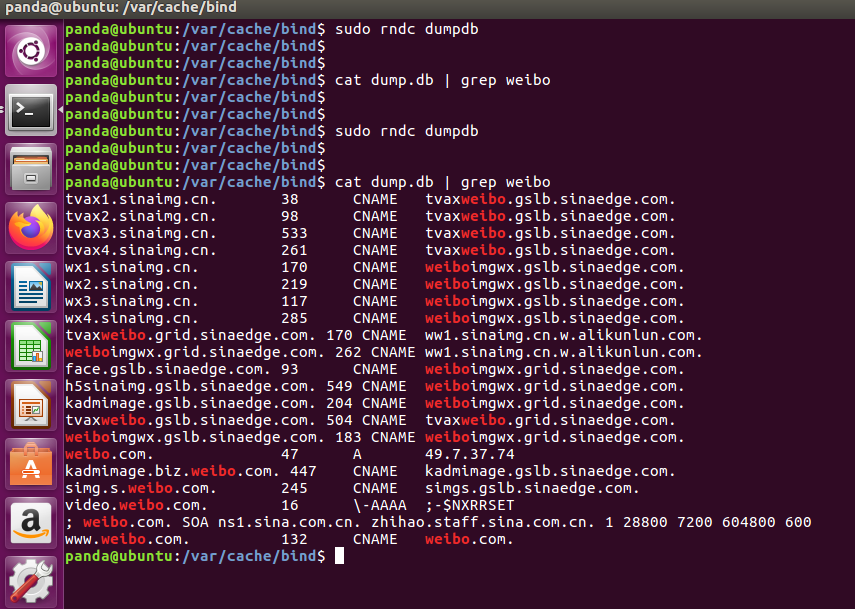
存储文件需要使用sudo rndc dumpdb 生成一下



回到**Ubuntu客户端**，使用浏览器访问[www.weibo.com](http://www.weibo.com)，不过有点慢



再回到Ubuntu服务端，重新使用rndc生成日志记录，可以看到已经产生记录

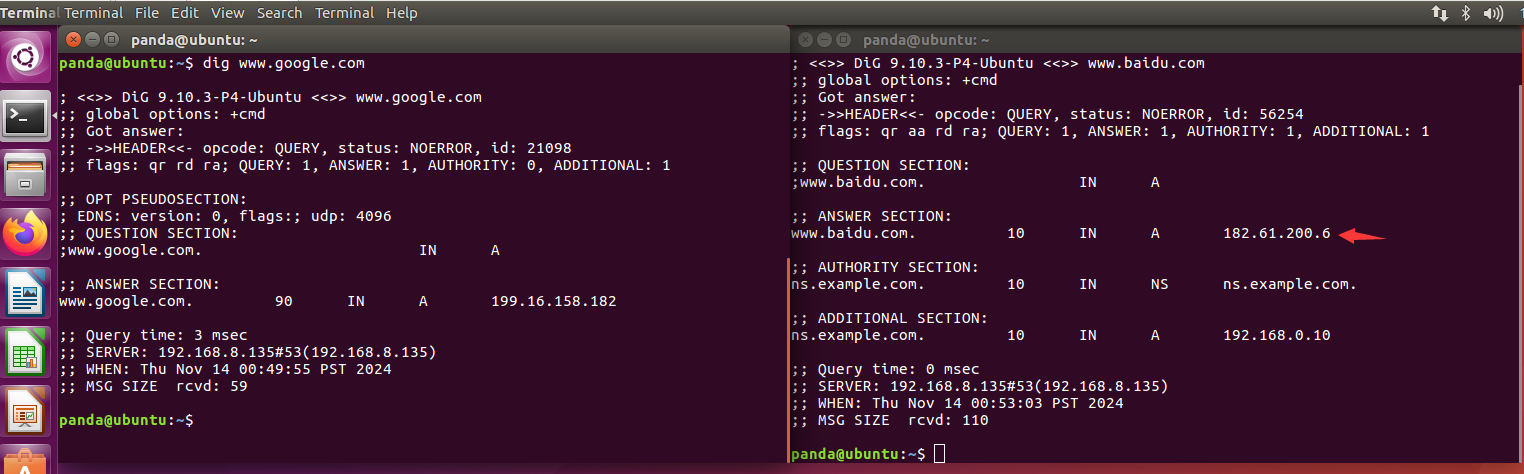


在**Ubuntu攻击端**执行DNS欺骗攻击，攻击前需要在**Ubuntu攻击端**安装scapy等python包

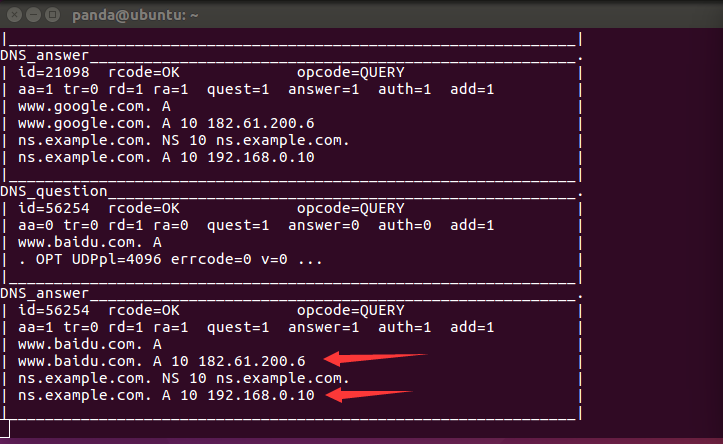
sudo netwox 105 -h "www.google.com" -H "182.61.200.6" -a "ns.example.com" -A "192.168.0.10" -f "src host 192.168.8.136" -d ens33

然后在**Ubuntu测试机**上分别先后执行dig [www.google.com](http://www.google.com) 和dig [www.baidu.com](http://www.baidu.com)

如下图所示



此时Ubuntu攻击端显示如下

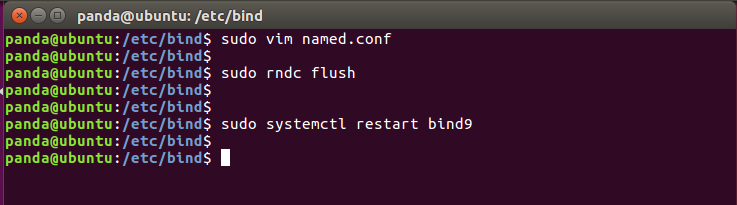


接下来，我们进行DNS投毒攻击。

先把named.conf中这些注释掉



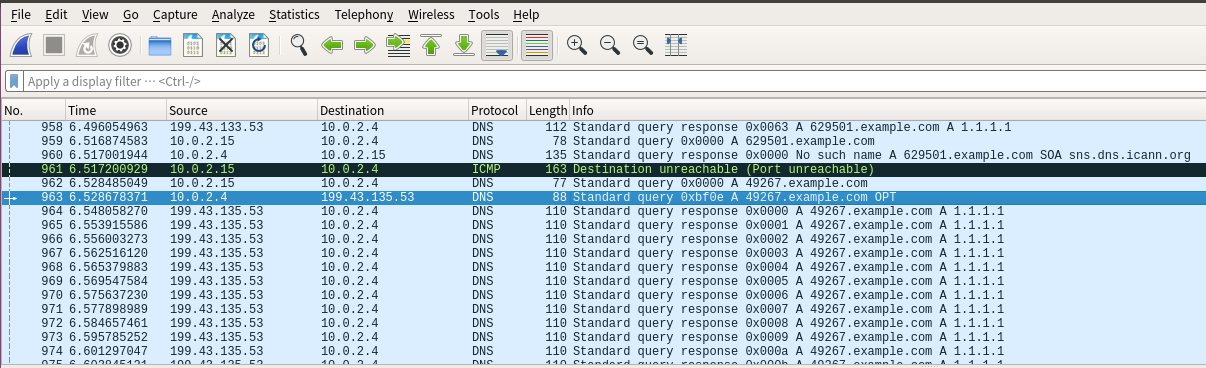
刷新并重启bind9



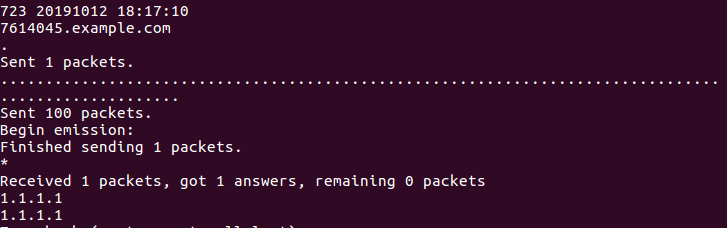
由于DNS污染概率较小，清理缓存比较麻烦，所以这次实验的目标是污染不存在的域名，即随机的一个域名，成功即可。

程序代码已经写好在附件中。文件中的最后几行用来配置目标IP。

做好上述准备后，开启wireshark进行抓包，并执行程序



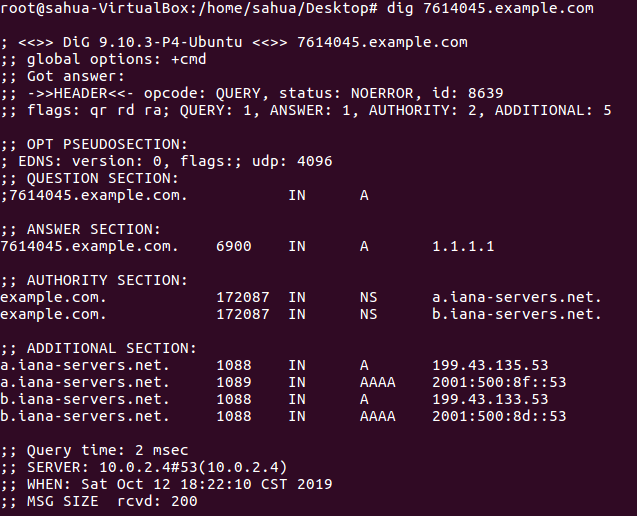
攻击成功后，程序结束：



这时，我们输入rndc dumpdb –cache，然后查看server端的cache，发现已经被污染

C:\Users\90415\Desktop\捕获.PNG

我们再在服务端进行dig，确实域名解析到了1.1.1.1。



注意点：

1. 如果程序输出结果中出现no ip！字样，那么请重新运行程序，连接超时了。
2. 投毒成功率较低，一般要运行十几分钟，时间和虚拟机发包速率有关。投毒程序ttl设置为了7200，记得在超时前完成剩余操作。验收时候验收截图。

<https://blog.csdn.net/freedomDA/article/details/121447367>

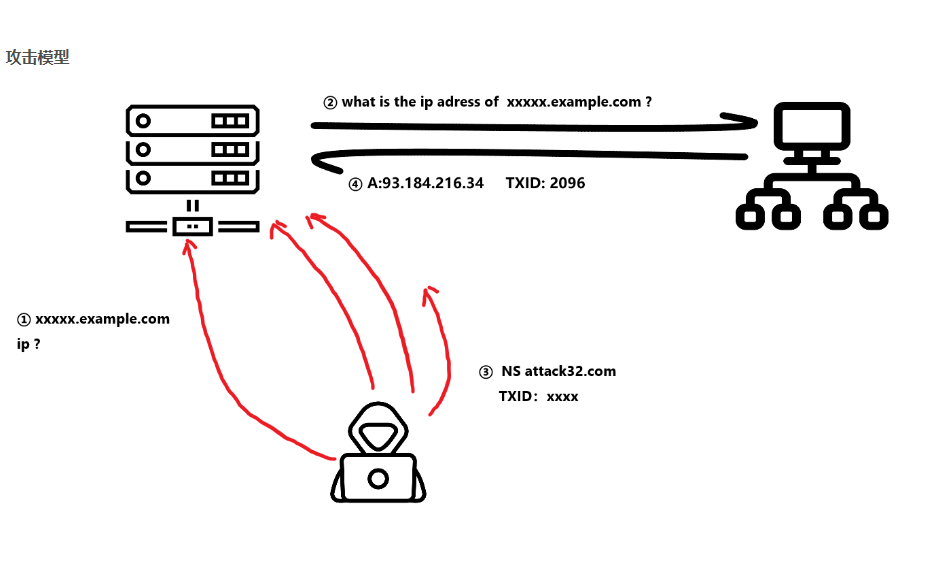
<https://blog.csdn.net/LostUnravel/article/details/127819578>

<https://blog.csdn.net/qq_36348899/article/details/115591486>

<https://makuang.net/post/9315>

备注：实验可能最终跑不出结果，原因可能如下

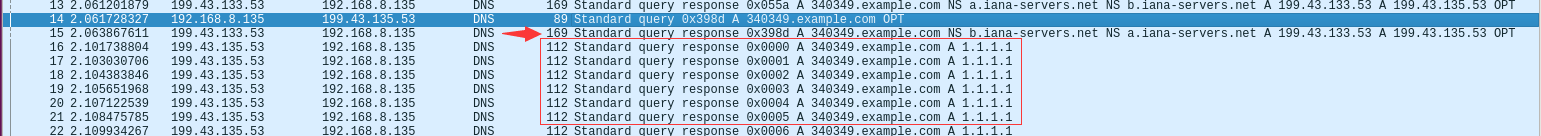
如下图所示，攻击成功的话，顺序必然是①②③④，然而实际测试时发现，可能出现①②④③，即上层DNS服务器对本地DNS服务器（Ubuntu客户端）的响应，比攻击者（Ubuntu客户端）的响应速度要快，导致攻击失败



图片来源：<https://blog.csdn.net/qq_36348899/article/details/115591486>

在Ubuntu服务端抓包时发现，④比③先出现

蓝色选中部分是②，红色箭头是④，红色方框是③



可以使用tc命令，将上层服务器IP地址的数据包延后1秒

sudo iptables -t mangle -A PREROUTING -i ens33 -s 199.43.133.53 -m length --length 131:0xffff -j MARK --set-mark 1

sudo iptables -t mangle -A PREROUTING -i ens33 -s 199.43.135.53 -m length --length 131:0xffff -j MARK --set-mark 1

sudo tc qdisc add dev ens33 root handle 1: prio

sudo tc filter add dev ens33 parent 1: protocol ip prio 1 handle 1 fw flowid 1:1

sudo tc qdisc add dev ens33 parent 1:1 handle 10: netem delay 1s

###### 了解资源记录

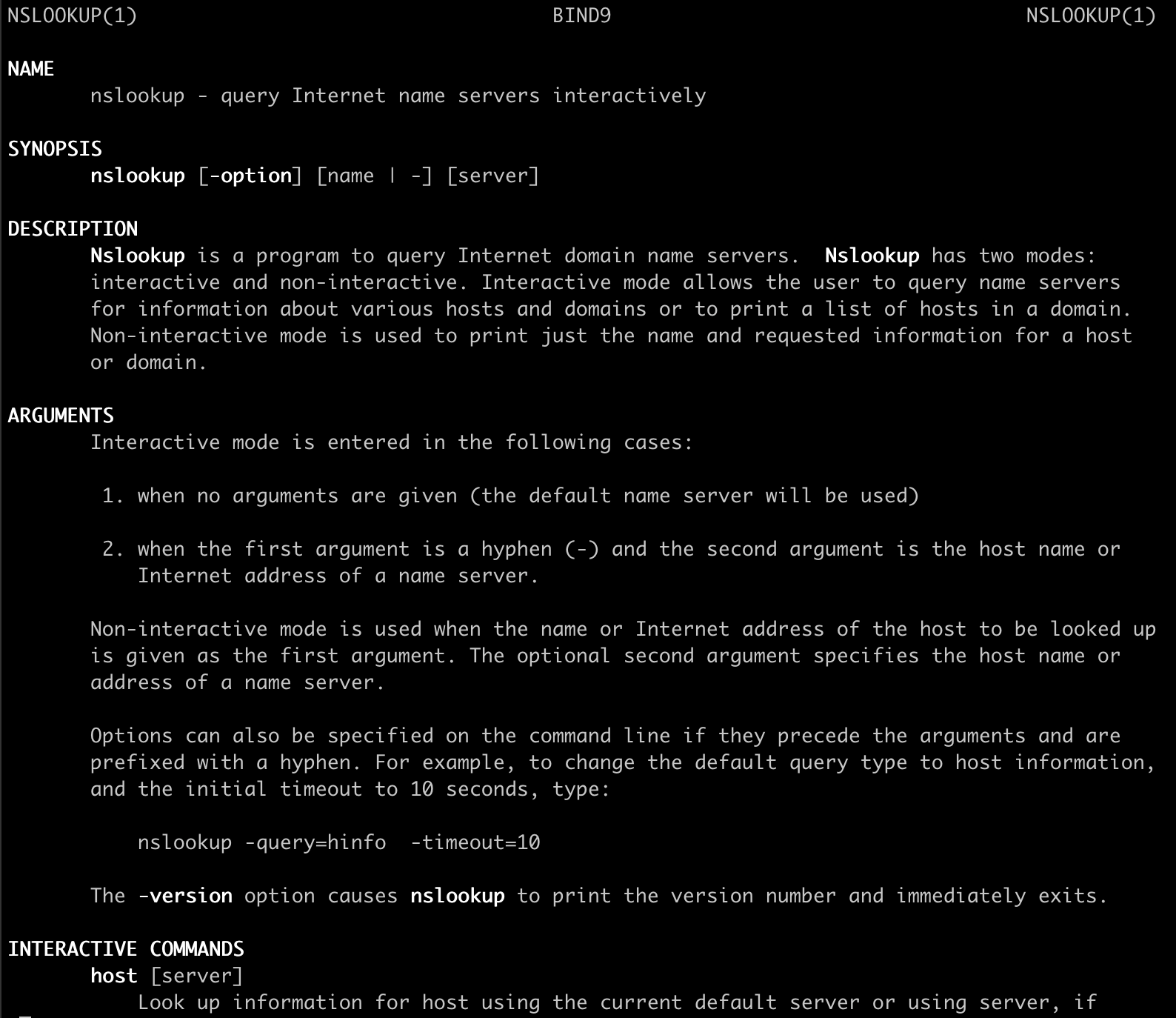
资源记录(Resource Records, RR)是域名系统(DNS)的基本信息单元，它们由的“记录类型”（例如A，MX，NS等）和“DNS类”（例如Internet，CHAOS等）来加以区分。每条记录都拥有一个有效期([TTL](https://baike.baidu.com/item/TTL?fromModule=lemma_inlink)，time-to-live)，每当这个时间耗尽后，它们所包含的信息必须从一个权威的名称服务器上进行更新。

RR的四元组组成形式：(Name, Value, Type, TTL)

1. Type=A：主机记录（A记录）。A记录用于名称解析的重要记录，提供标准的主机名到IP的地址映射，此时Name为主机名，Value为IP地址。
2. Type=CNAME：别名记录（CNAME记录）。向查询的主机提供主机名对应的规范主机名，此时Name为规范名字的别名，Value为规范名字。
3. Type=NS：域名服务器记录（NS记录）。用来指定该域名由哪个DNS服务器来进行解析，此时Name为域名，Value为该域名的权威服务器的域名。
4. Type=MX：Value为Name对应的邮件服务器名字。

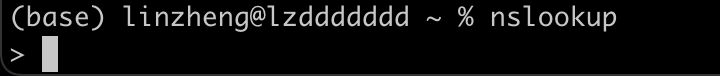
DNS中的资源记录是按照名称字段组织的，即DNS树中一个节点的完整网域名称(FQDN)。而对于MX记录而言，这就是收件人的[电子邮件地址](https://baike.baidu.com/item/%E7%94%B5%E5%AD%90%E9%82%AE%E4%BB%B6%E5%9C%B0%E5%9D%80?fromModule=lemma_inlink" \t "/Users/linzheng/Documents\\x/_blank)的域名部分，即@后面的部分。也就是说，对于someone@example.com这个电子邮件地址，example.com会用做MX记录的查询。

###### 使用nslookup工具对MX记录进行观测

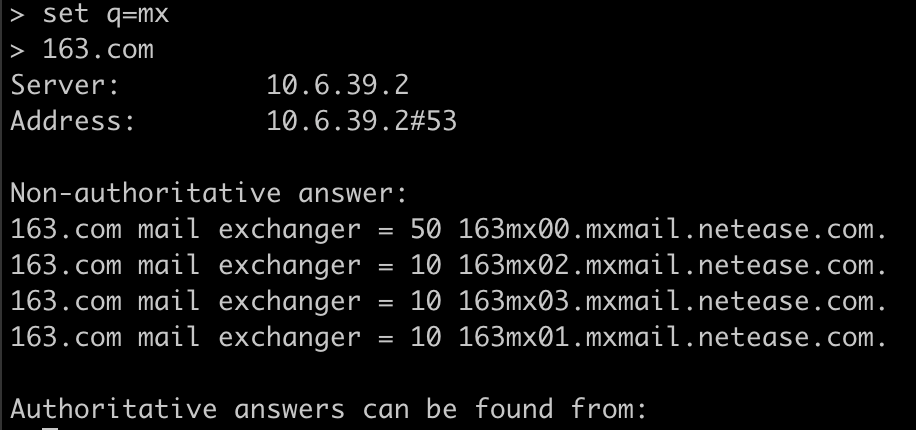
打开终端，查看nslookup命令使用方法：

交互模式进行查询:

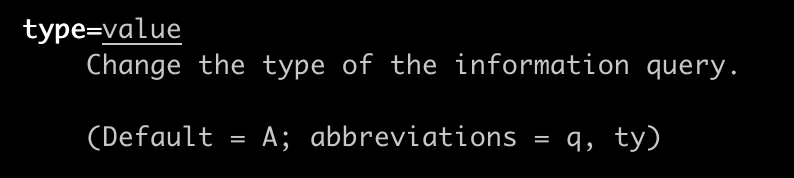
（1）进入交互模式：nslookup



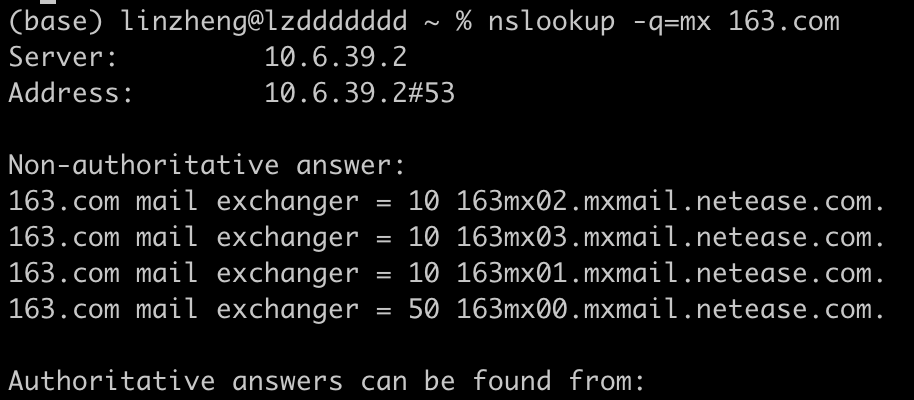
1. 设置资源记录类型为MX记录：



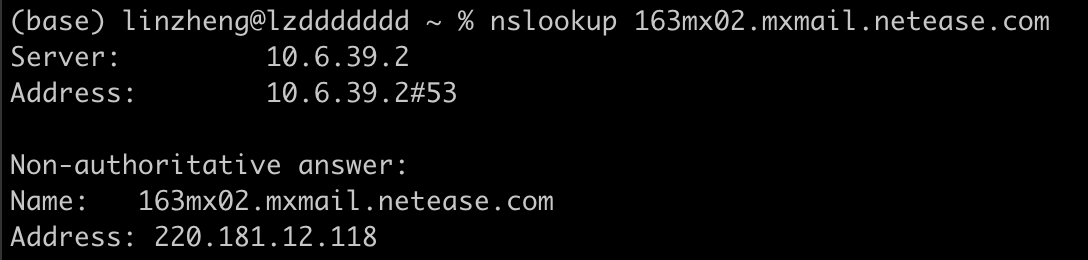
查看命令帮助手册，可知-q或-ty（即-querytype和-type的缩写）可设置查询资源记录的类型。默认值为A记录。



非交互模式进行查询：



此时通过向DNS服务器进行查询，可以知道163邮箱存储邮件的准确服务器地址。



###### 向SMTP服务器投递邮件实验

1. 选择邮箱并启用SMTP协议：

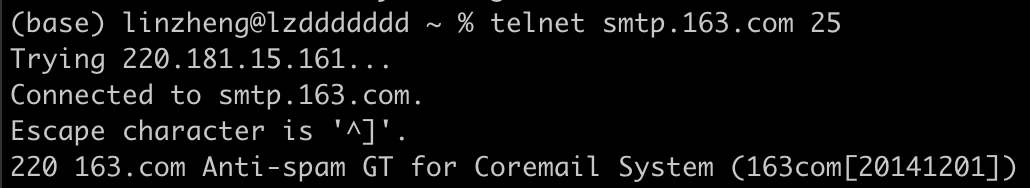
我选择了网易邮箱为发送邮箱，QQ邮箱为接收邮箱，大家可以选择适合自己的邮箱进行实验，并到对应邮箱启用SMTP协议。下图为网易邮箱的SMTP服务器地址截图。



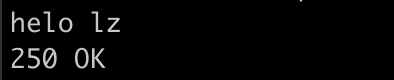
在启用SMTP协议后，往往会生成一个指令作为第三方登录邮箱的密码，要妥善保存。

1. 连接SMTP服务器并投递邮件：

SMTP默认使用端口号25，连接网易邮箱的SMTP服务器：telnet + <SMTP服务器> + <端口号>

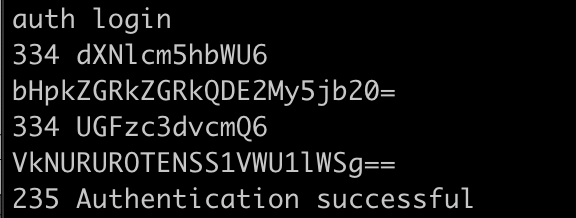


向SMTP服务器表明身份：helo + <任意字符串>



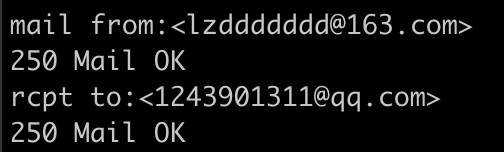
输入用户名和密码：auth login，然后依次输入用户名+回车和密码+回车。

我在之后输入的用户名和密钥都选择使用base64编码（默认编码方式），该密码为开启SMTP时自动生成的密码。



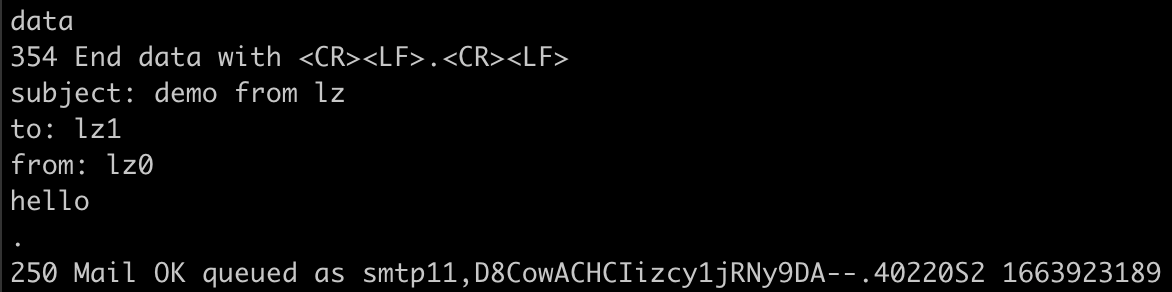
申明邮件发送方：mail from:<发送邮箱>

申明邮件接收方：rcpt to:<接收邮箱>



编辑发送信息：data+回车

data并回车之后，及发送的内容，可以申明subject，from，to等信息。输入“.”后回车，即发送邮件。



邮箱收到邮件：



尝试不要使用任何邮件服务器账号，通过自己电脑直接向你所申请的163邮件服务器账号投递邮件，记录实验过程，要求收到邮件后显示发件人邮箱为：你的学号@oaurewouerw.com。

### 实验报告要求

1. 完成所有实验步骤，并截图。
2. 使用dig+trace解析另一个域名，并写出每一步过程解释。
3. 包括但不仅限于实验环境，实验原理，实验过程，实验结果，实验中遇到的困难，解决方式等。