网络与信息安全课内实验2

Host文件与DNS投毒

一．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。

二．实验原理

DNS服务的工作过程

1、以http://www.qq.com为例，在浏览器中输入www . qq .com 域名，操作系统会先检查自己本地的hosts文件是否有这个网址映射关系。如果hosts里没有这个域名的映射，则查找本地DNS解析器缓存。

2、如果hosts与本地DNS解析器缓存都没有，首先会设置的首选DNS服务器，在此我们叫它本地DNS服务器，此服务器会查询本地配置区域资源，此解析具有权威性。如果还没有会查询服务器缓存，此解析不具有权威性。

3、如果本地DNS服务器本地区域文件与缓存解析都失效，则会进行查询，本地DNS服务器收到IP信息后，将会联系负责.com域的这台服务器。这台负责.com域的服务器收到请求后，如果自己无法解析，它就会找一个管理.com域的下一级DNS服务器地址(http://qq.com)给本地DNS服务器。当本地DNS服务器收到这个地址后，就会找http://qq.com域服务器，重复上面的动作，进行查询，直至找到www . qq .com主机。现在绝大多数DNS服务器都是该模式。

DNS缓存投毒原理

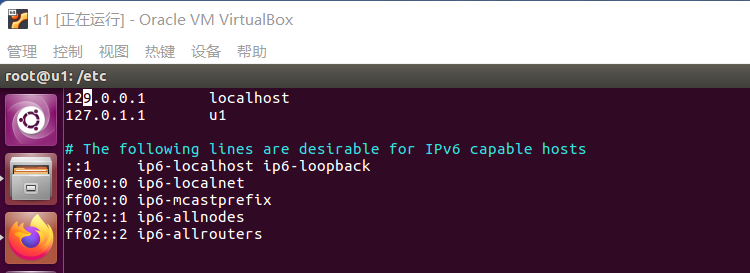
一般而言，各个网站所持有的IP地址不会轻易发生变化(当然也不排除按需变化的可能性)，因此DNS计算机会将此类映射关系的信息存储起来，以供日后使用。此后，当您要再次访问该银行的网站时，您的计算机就不必再去“麻烦”DNS服务器了。它直接查看自己的缓存，并找到上次接收到的IP地址即可。在计算机使用DNS缓存时，它不会去主动关注：自上次使用过以来，IP地址是否发生了变更。从某种程度上说，DNS缓存相当于计算机的内存，就算缓存中的值被修改了，计算机仍然会机械地去读取既有的映射信息。 目前广泛采用DNS缓存投毒攻击是最早由Kaminsky提出的，原理便是攻击者先向DNS缓存服务器发送一个不存在域名的请求报文，触发缓存服务器向DNS授权服务器发出查询请求，同时攻击者向缓存服务器发送大量的伪造回应报文，以期在DNS授权服务器回应到达之前命中缓存服务器的请求信息(命中请求包的ID)，将恶意授权服务器地址置入DNS缓存服务器缓存项中，篡改某些网站的域名和IP地址对应关系。

三．实验步骤

U1server

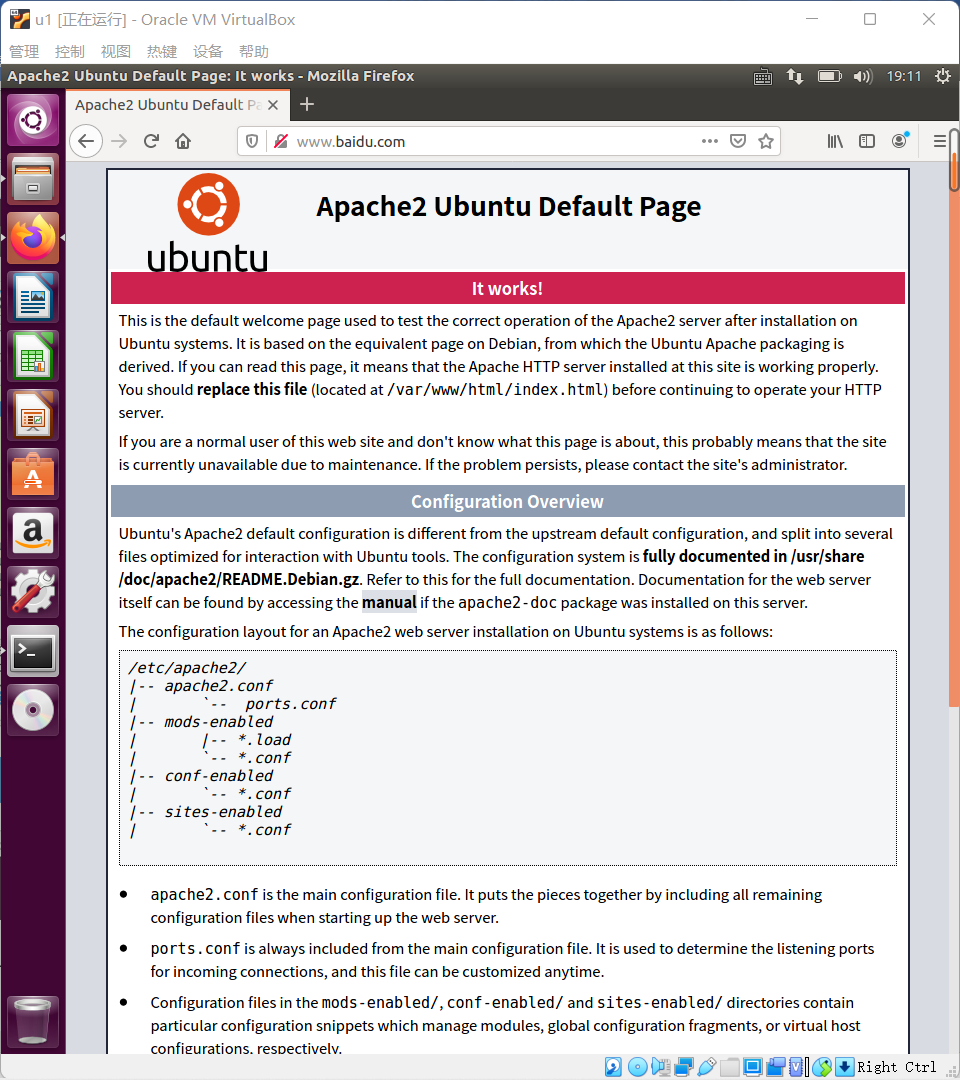
U2attacker

打开hosts查看并了解



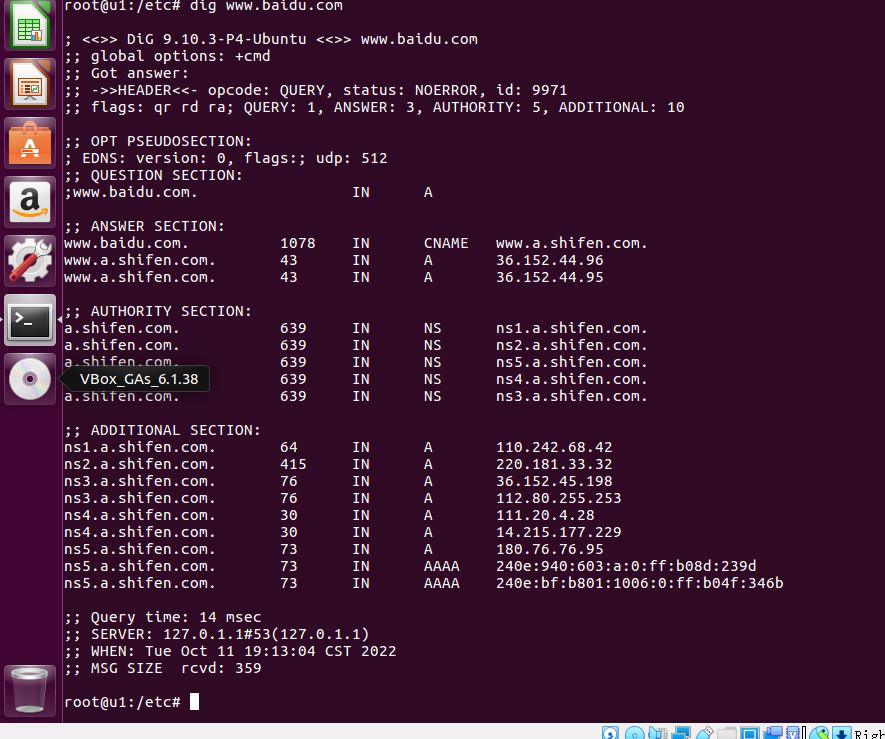
添加了127.0.0.1 [www.baidu.com](http://www.baidu.com)

修改后访问www.baidu.com

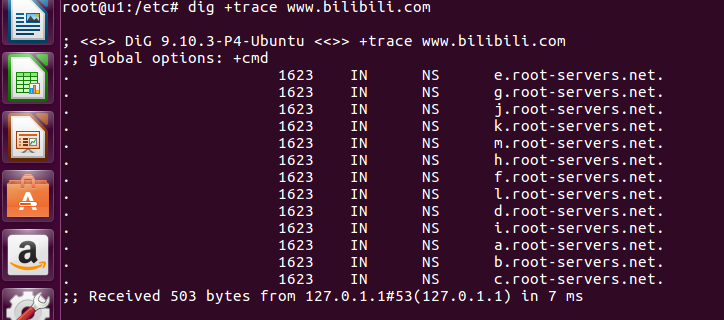


接下来出现了一个让我烦恼的事情，当我想打开百度网站的时候，就一直直接打开了127.0.0.1，导致我无法进行百度，接着我将hosts中之前加入的删除掉之后，发现依旧无法正常登录百度网站。经过一系列的操作之后，尝试重启了网络服务之后，才成功解决该问题

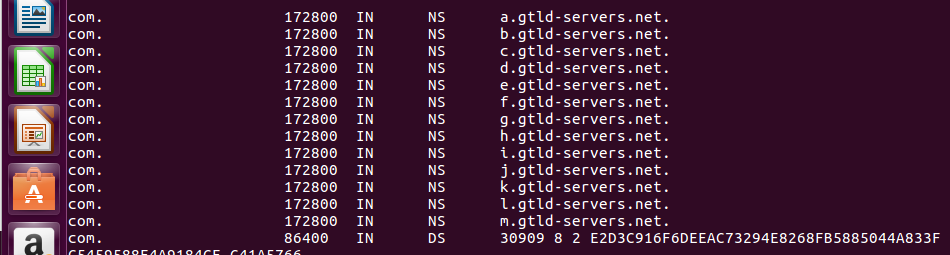
使用dig以及dig +trace查看[www.baidu.com](http://www.baidu.com)的解析过程



第一步，查询 root 根服务器



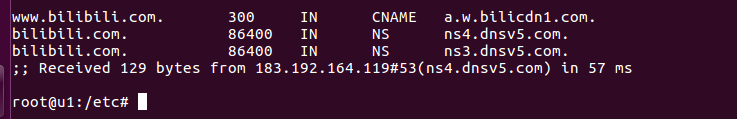
第二步，查询.com 服务器



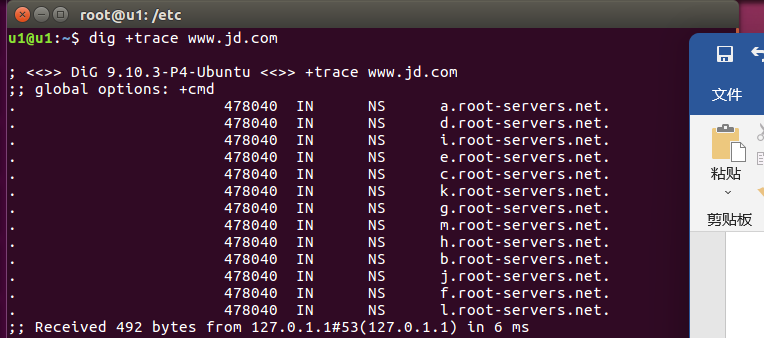
第三步，查询 bilibili.com 服务器

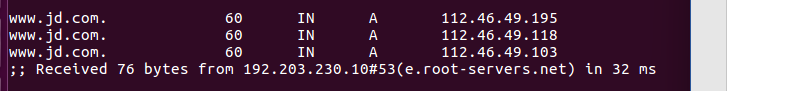


第四步，查询 www.bilibili.com 服务器地址

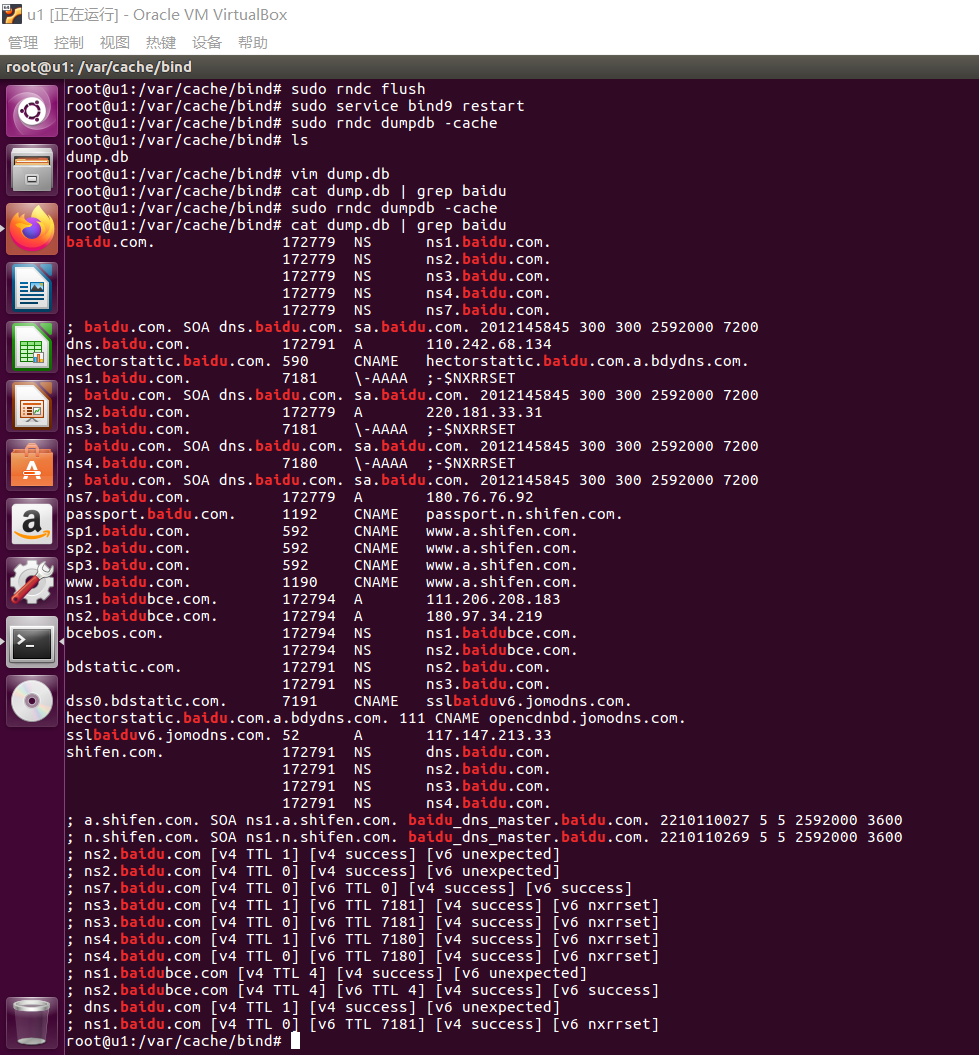


此处完成dig +trace解析另一个域名：



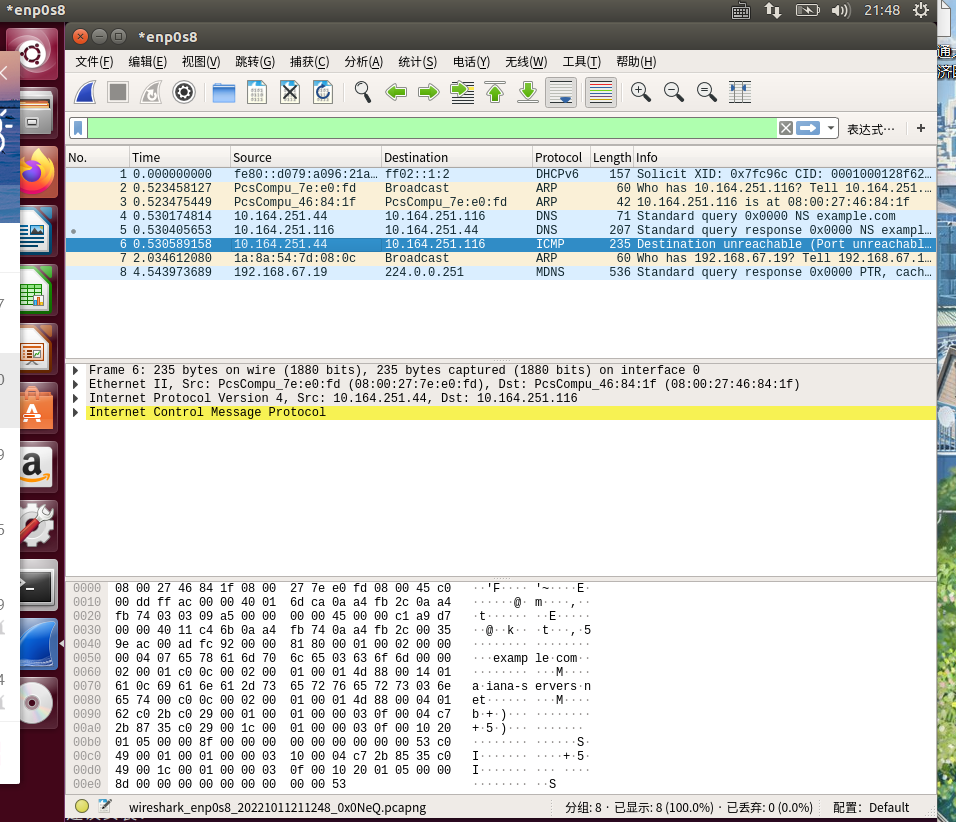


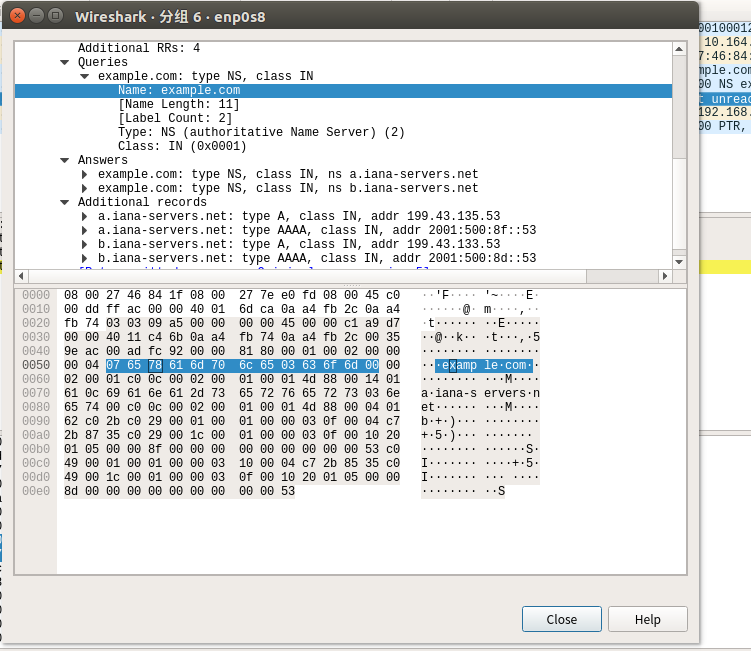
DNS 投毒实验：



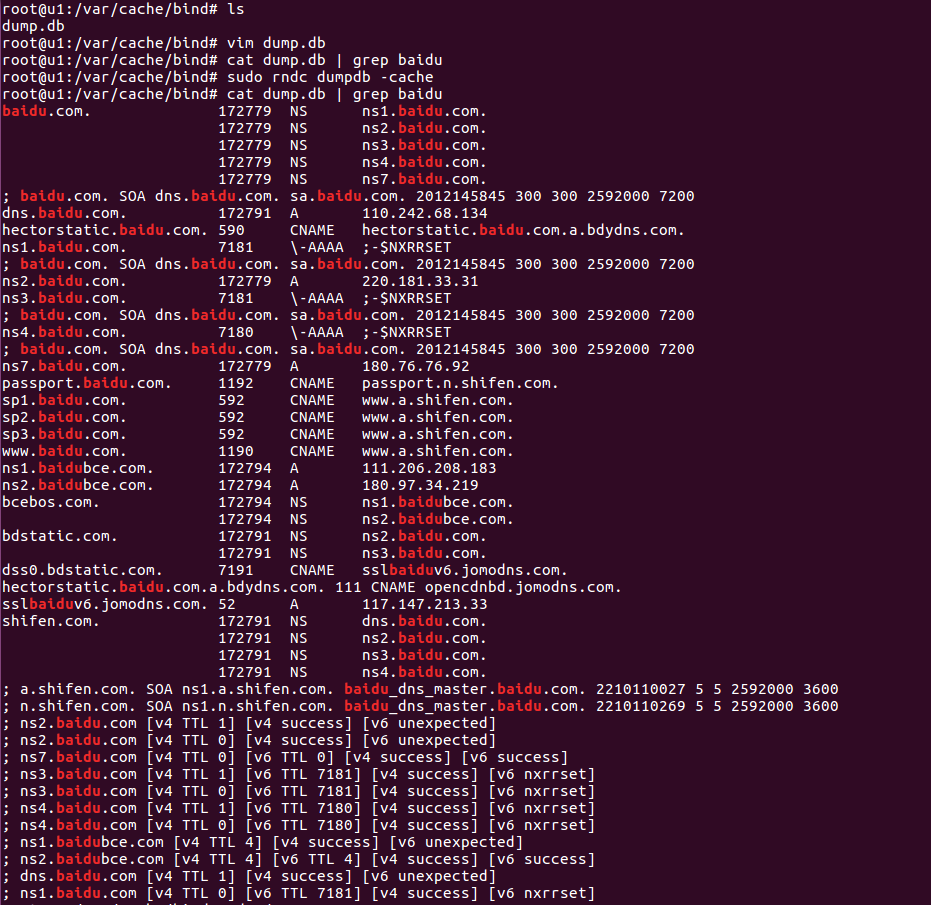
刷新DNS缓存，然后重启DNS服务器，将DNS数据导出并查看初始状态

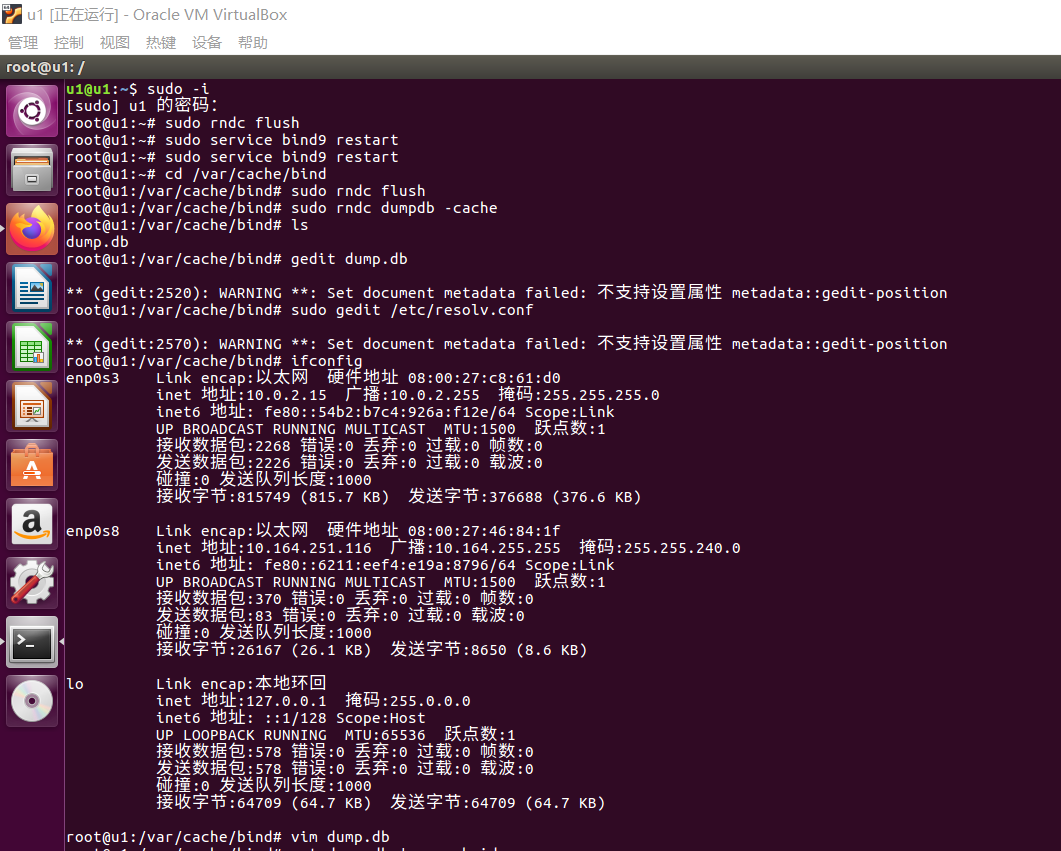


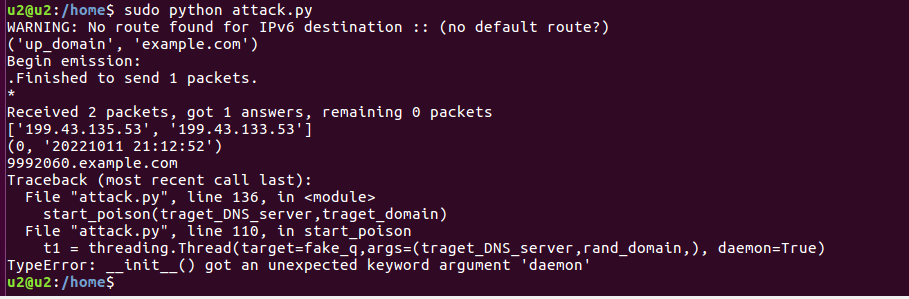


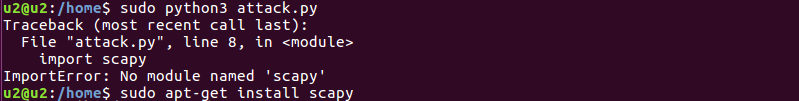


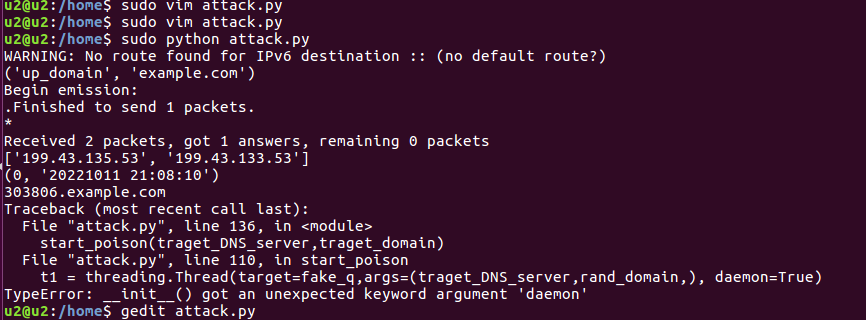
在客户端访问[www.baidu.com](http://www.baidu.com)之后，我们可以看到server的解析记录。





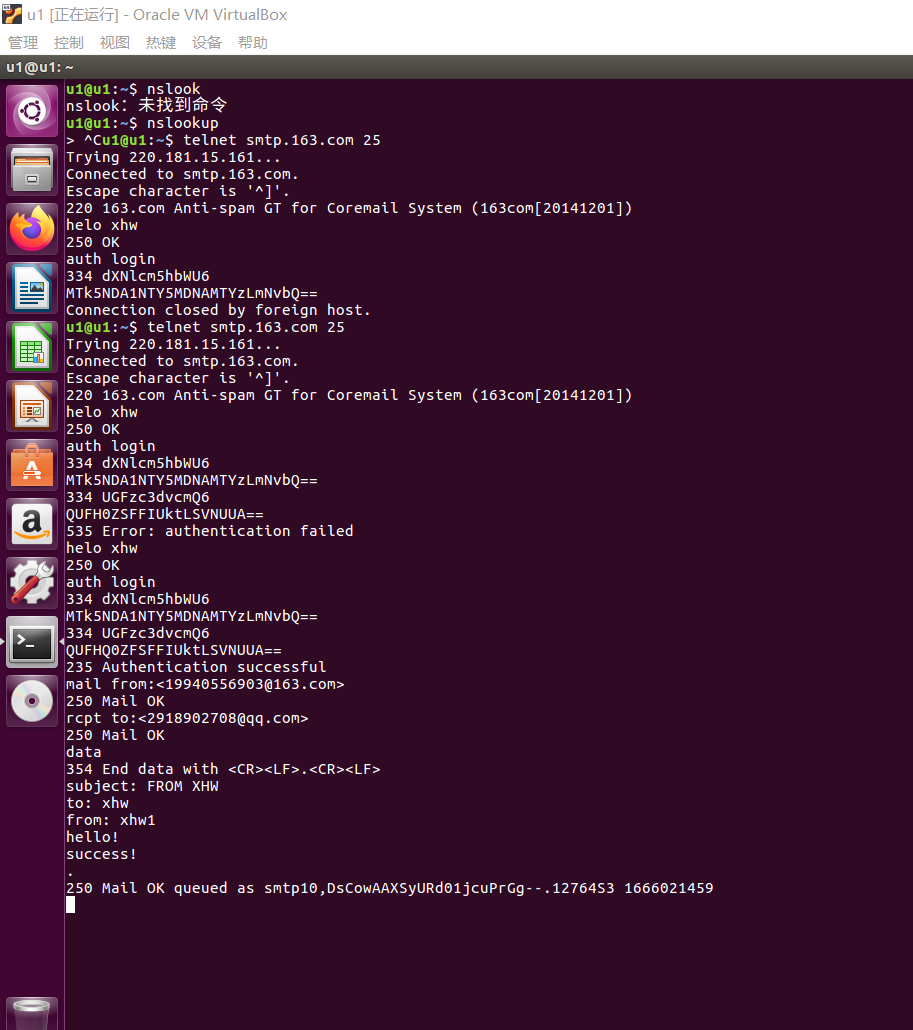


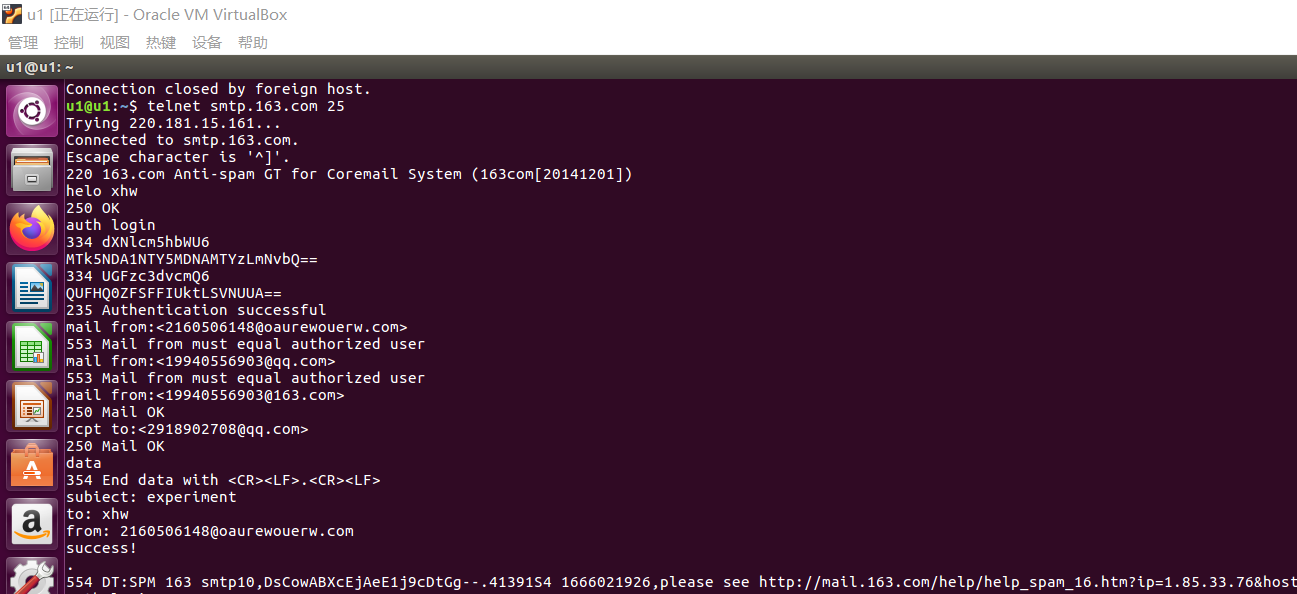




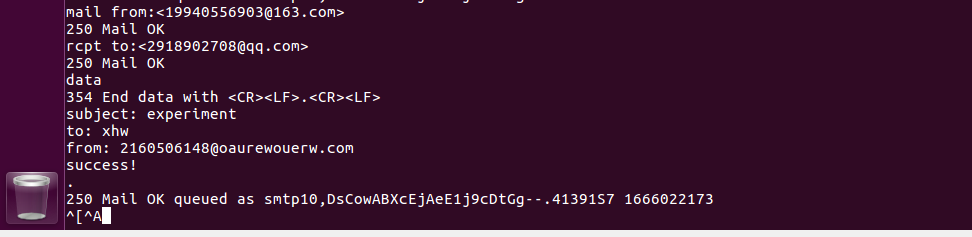
向SMTP服务器投递邮件实验：

其中遇到了不少的问题，包括base64编码没转换以及如何转换的问题，花费了大量的时间去解决





邮件的发送



实验结果：成功收到邮件

