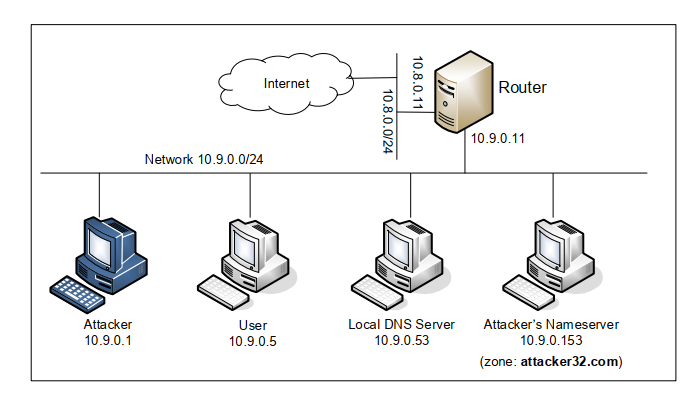
**Lab 5**

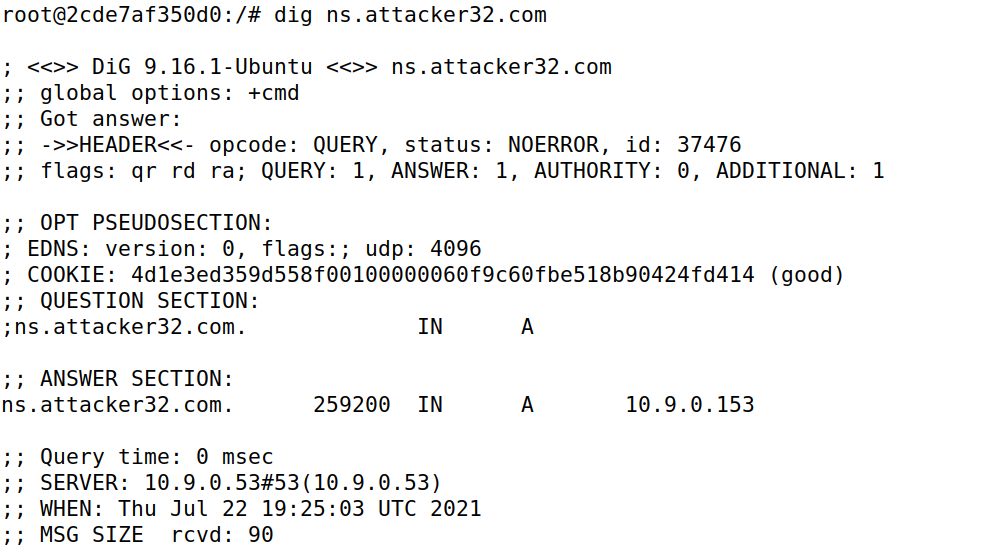
57118201邓彤

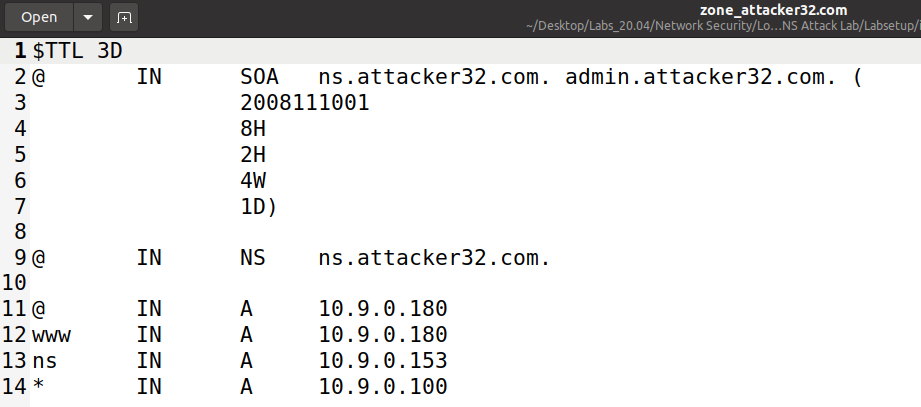


**Testing the DNS Setup**

**Get the IP address of ns.attacker32.com**

**测试 DNS 配置是否正确， 首先使用 dig 命令查询 ns.attacker32.com 的地址：**

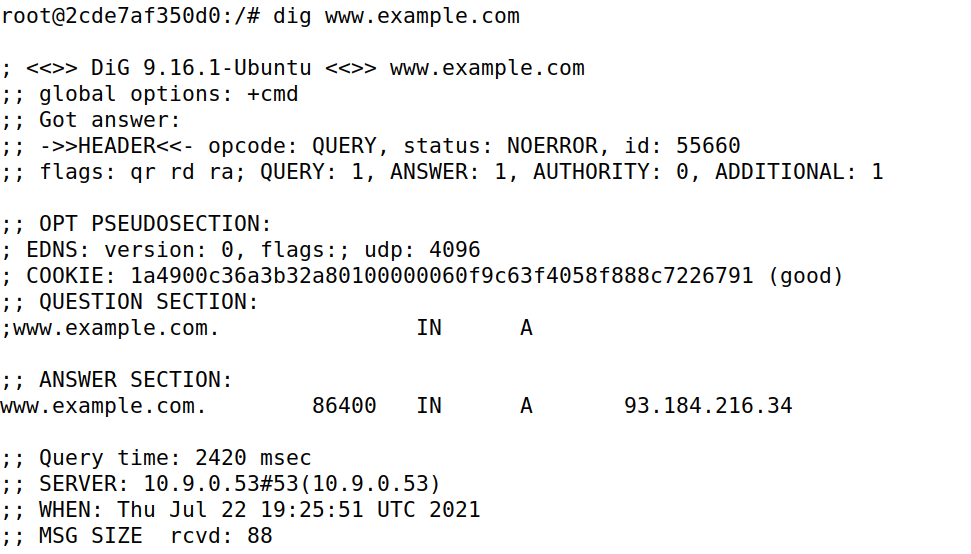




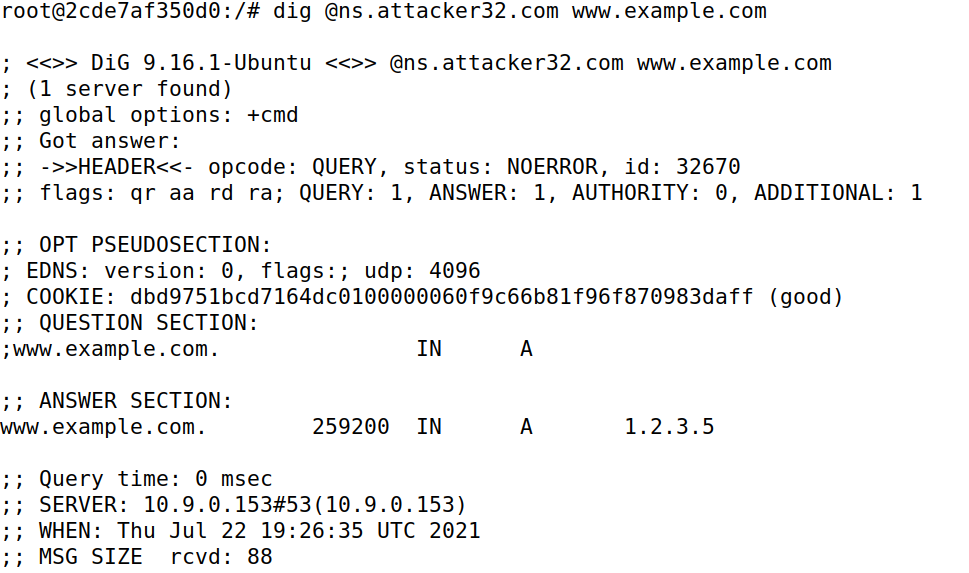
**记录中的 ip 地址与文件中一致， 说明设置没有问题**

**Get the IP address of** [**www.example.com**](http://www.example.com)

**执行dig** [**www.example.com**](http://www.example.com)**命令，输出结果如下：**



**执行dig @ns.sttacker32.com** [**www.example.com**](http://www.example.com)**命令，输出结果如下：**

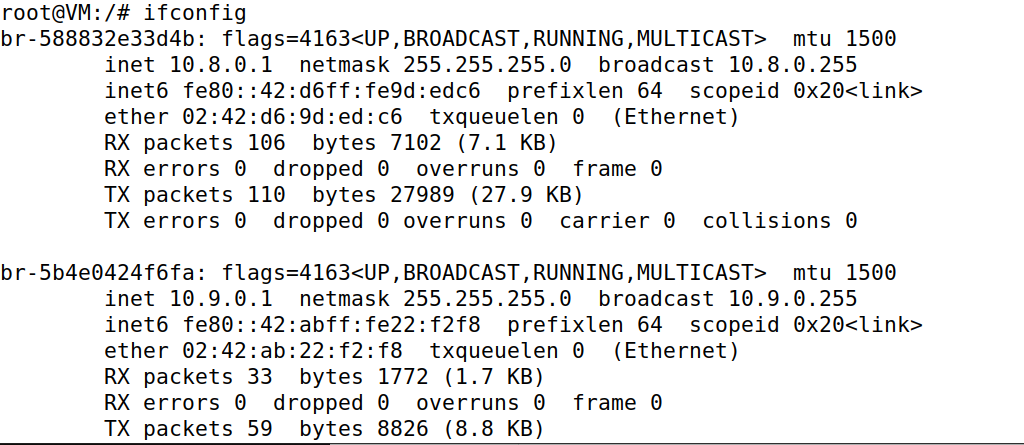


**可见两个命令得到的ip地址不同，第一个命令直接从官方域名服务器获取信息，而第二个是从攻击者得到了假的结果，本次实验的目的就是让用户执行第一个命令，即从官方域名服务器获取信息时依旧得到假的结果。**

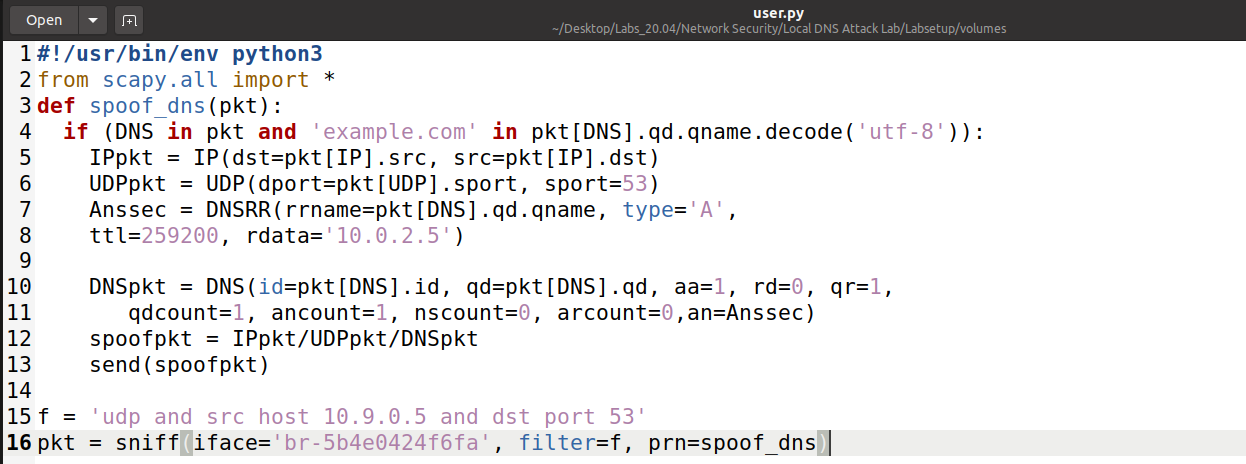
**Task1：Directly Spoofing Response to User**

**实验目的是捕获用户发出的DNS请求，然后返回一个假的DNS请求，然后返回一个假的DNS响应，首先在本地域名服务器上清除缓存：**

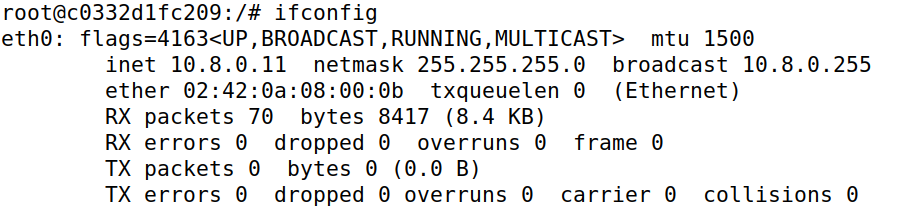
C:\Users\dengtong\AppData\Roaming\Tencent\Users\1977988055\QQ\WinTemp\RichOle\Z$6EJ}906VW(SEJ8~[W[7$X.png

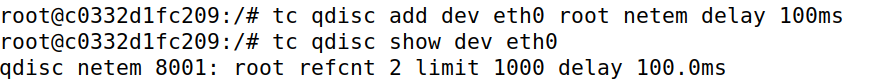


**代码如下：**

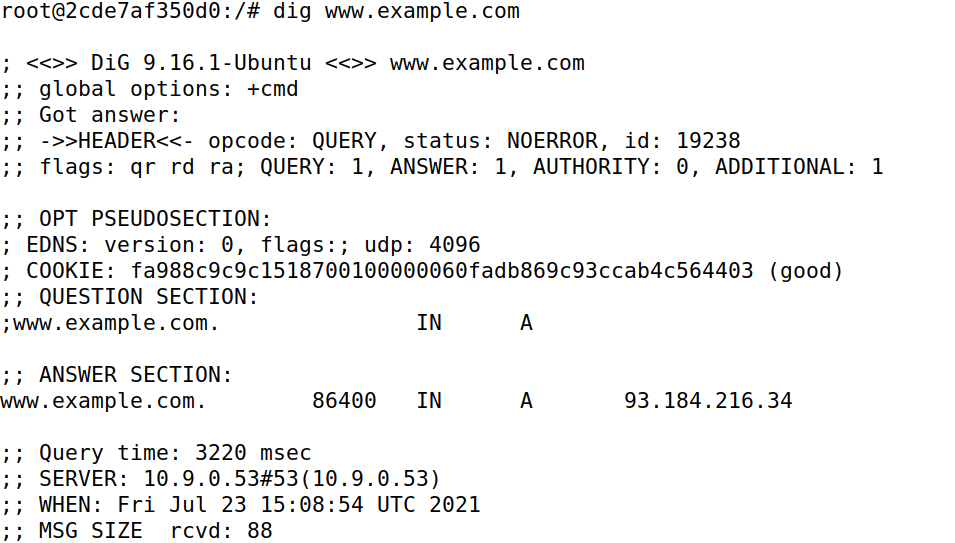


**在执行代码后发现无法捕获构造的DNS包，发现由于容器本来存在的问题，真正响应的DNS响应比伪造的先到达用户，所以在路由器上增加输出网络流量的延迟：**

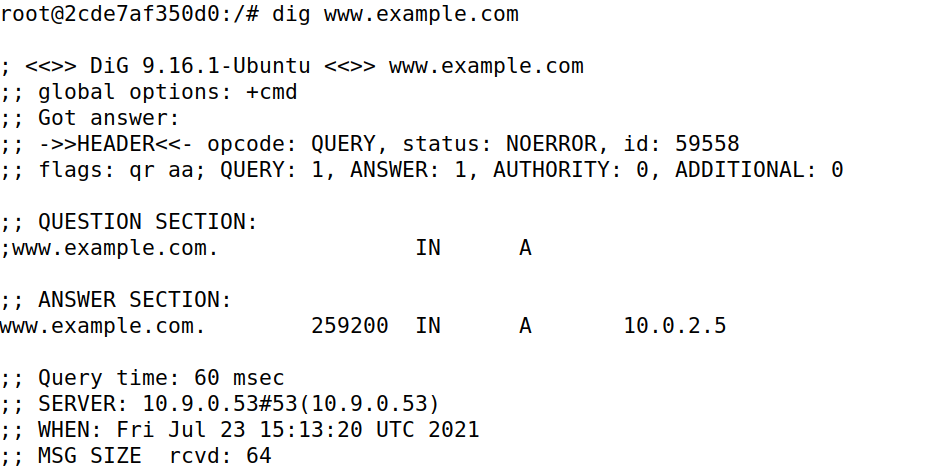




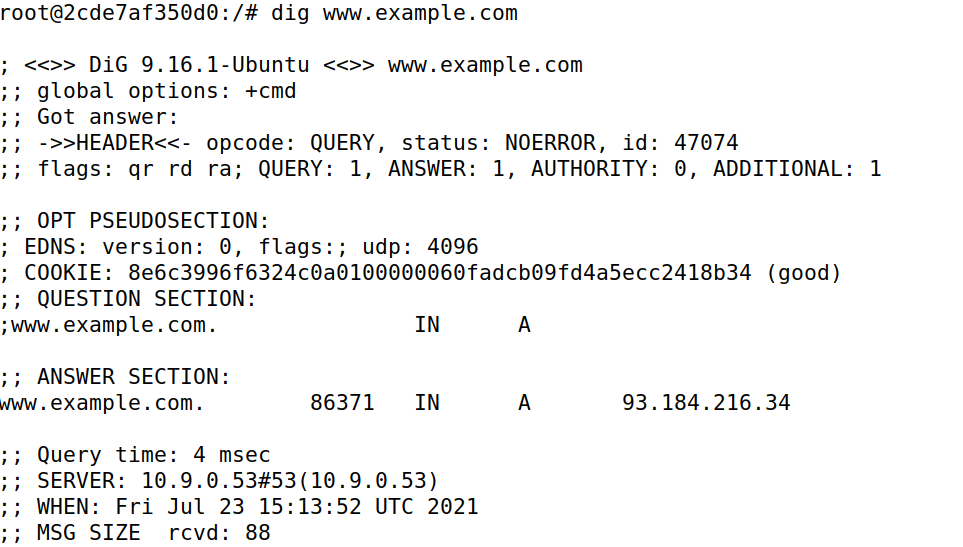
**攻击前先查看[www.example.com](http://www.example.com)的DNS 信息：**



**执行攻击程序，攻击时查询到的DNS信息：**



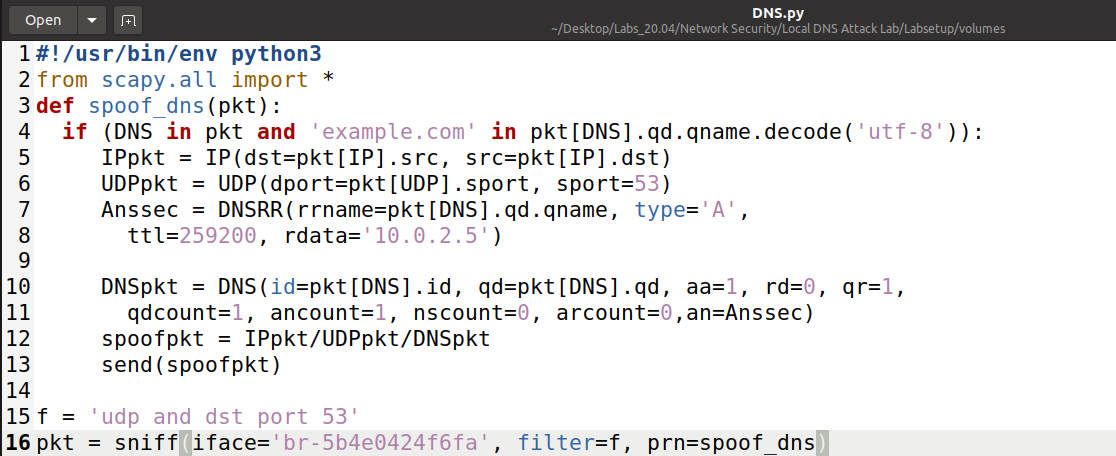
**关闭攻击程序后，DNS信息：**



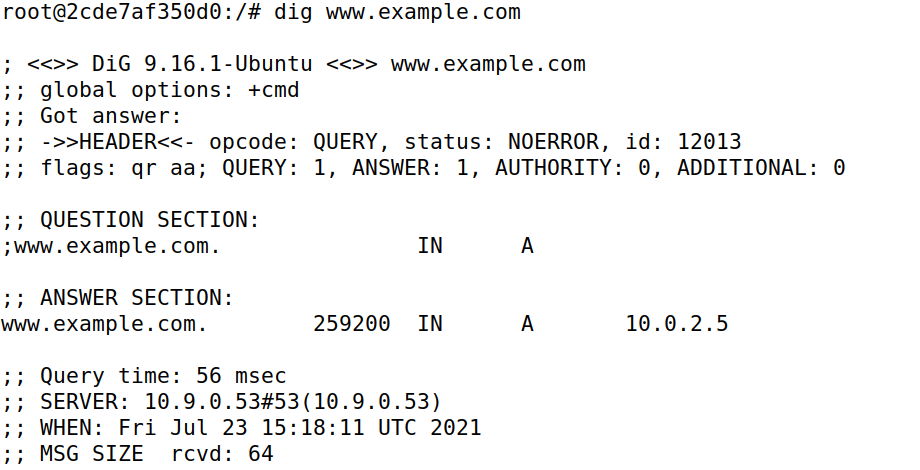
**可以发现攻击后 example.com 指向的 ip 地址发生了变化，变成了攻击者伪造的 ip 地址，攻击成功。但是停止攻击后再次执行dig 命令发现 ip 又恢复了**

**Task2：DNS Cache Poisoning Attack – Spoofing Answers**

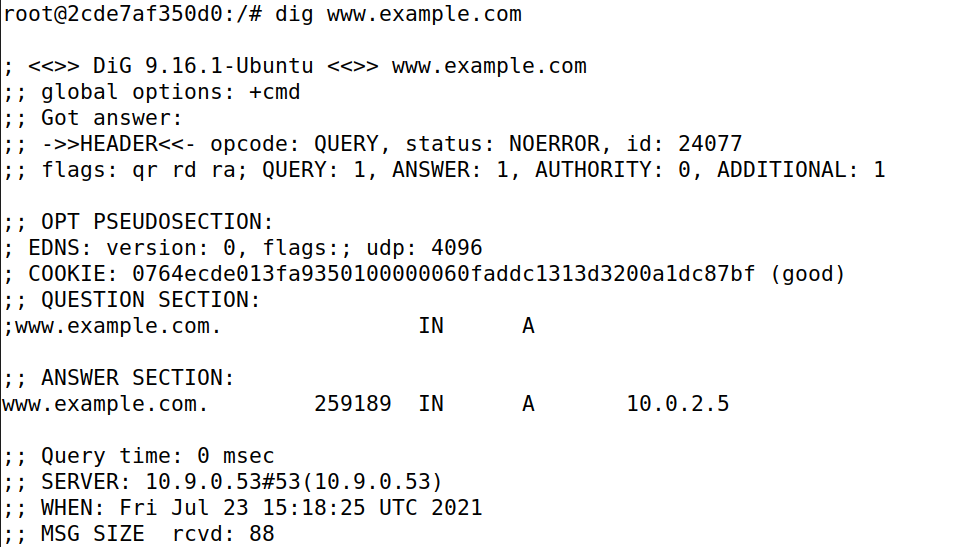
**在该实验中，为了达到每次用户的机器发出[www.example.com的DNS](http://www.example.com的DNS)查询时，攻击者必须发出欺骗的DNS响应。因此，需要伪造其他域名服务器发送给本地域名服务器的DNS响应，这样伪造的信息将会在本地服务器的缓存中保存一段时间，使得攻击者只需发送一次伪造响应，在缓存信息过期之前都有效。**



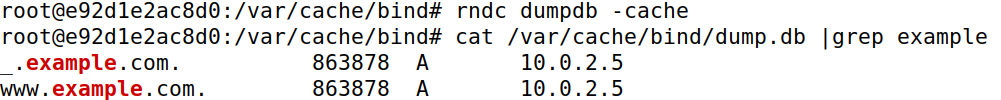
**执行攻击时输出如下：**



**停止攻击后输出如下：**



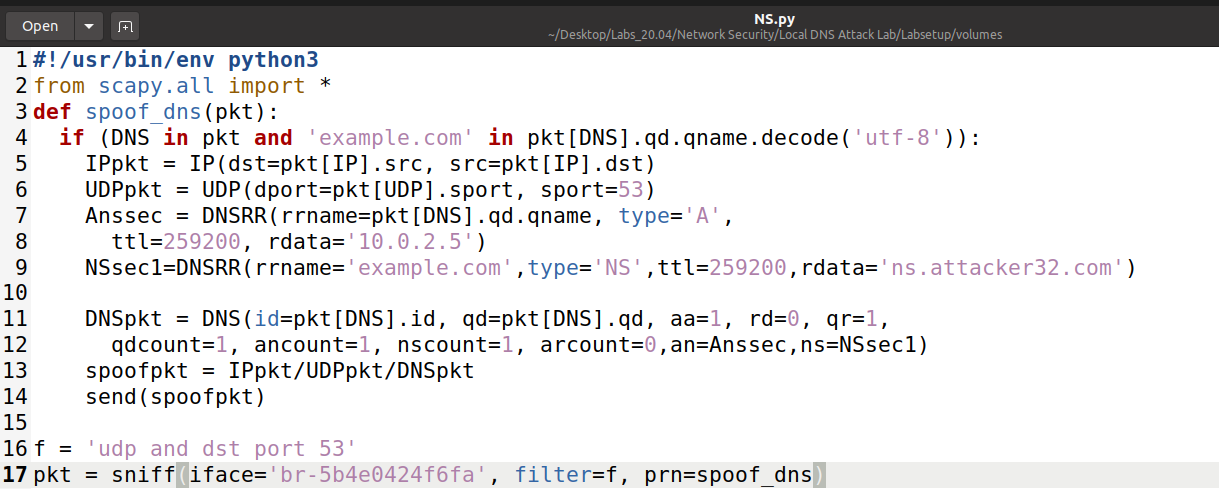
**可以发现在执行一次攻击后依然能维持攻击效果。查看本地域名服务器，可以看到伪造的DNS信息以及存储在缓存中：**



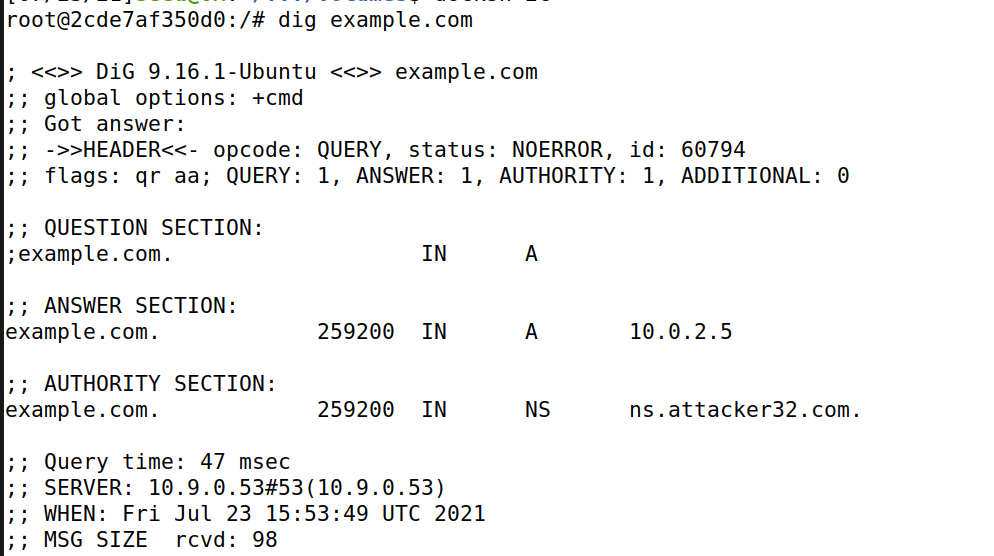
**Task3：Spoofing NS Records**

**在task2中，DNS缓存中毒只攻击一个主机名，即www.example.com。如果用户试图获得另一个主机名IP地址，如mail.example.com，就可增加一条NS记录。**

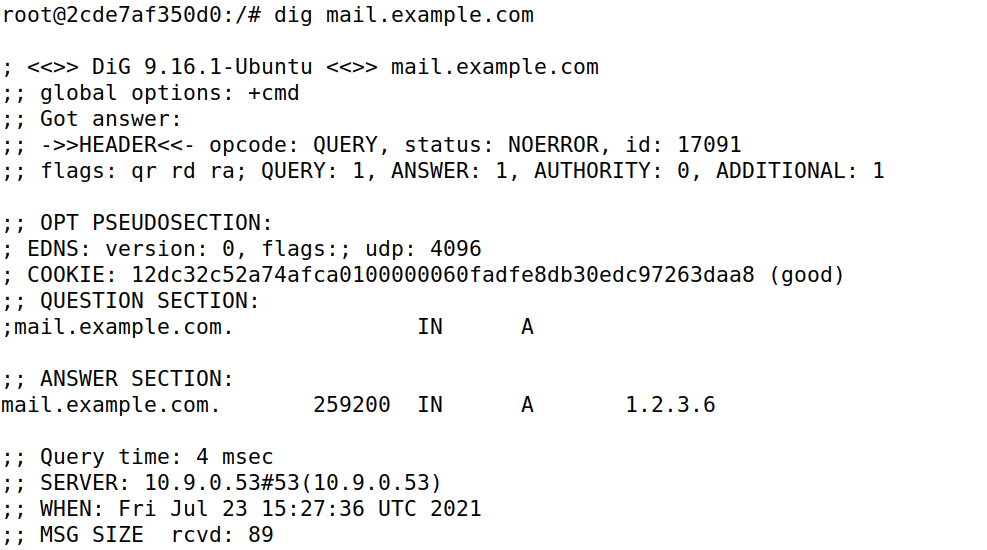
**代码如下：代码中增加一条NS记录，nscount=1**

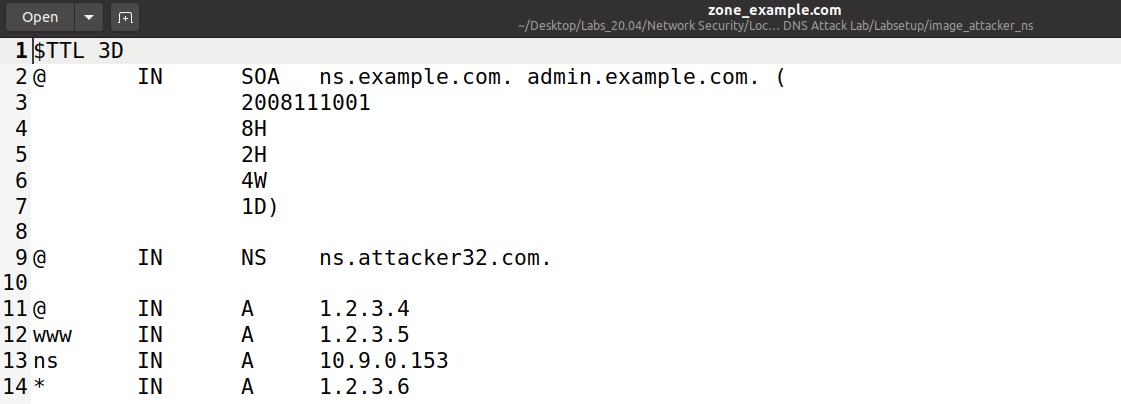


**执行程序，查询example.com的信息，结果如下：**

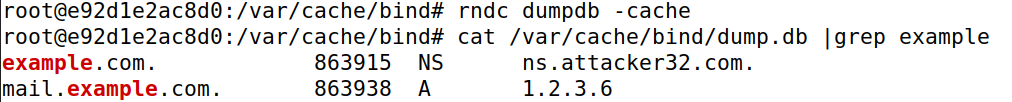


**发现该地址指向ns.attacker32.com域名，然后停止攻击程序，查询统一域名的不同主机名的信息，查询mail.example.com的DNS信息：**



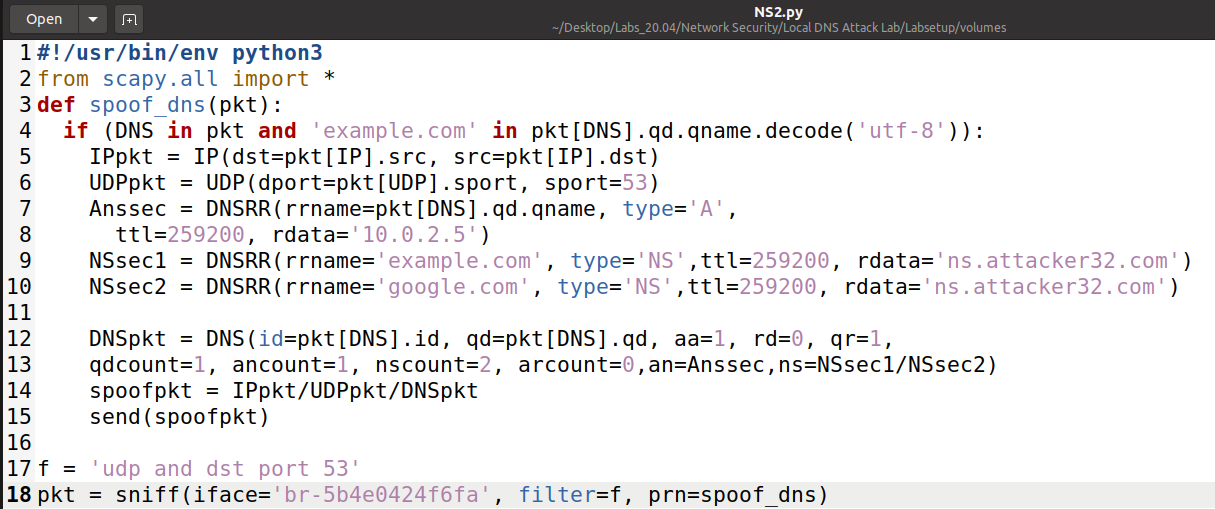


**发现该地址指向1.2.3.6，说明该地址是攻击者伪造的内容。查看缓存：发现NS记录也在缓存中，攻击成功。**

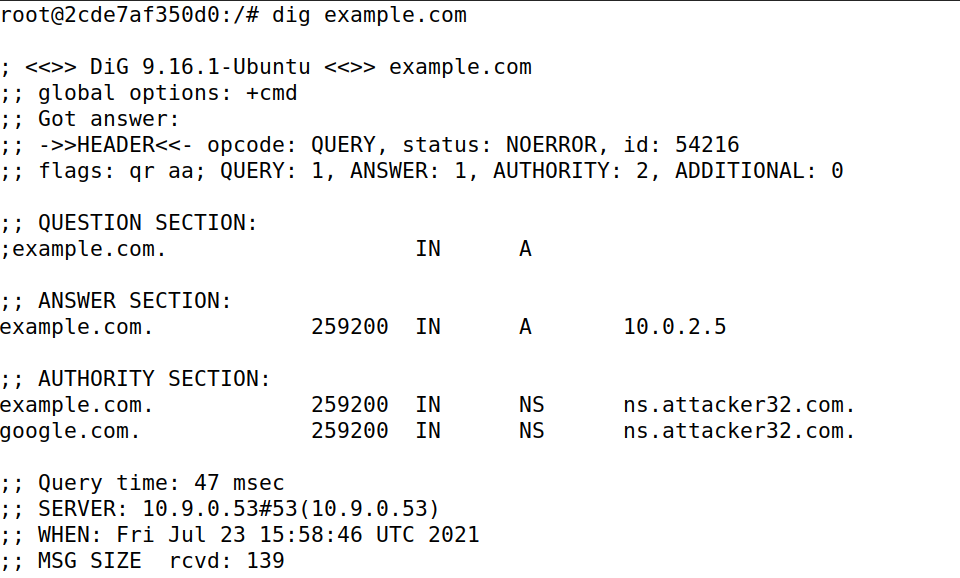


**Task4：Spoofing NS Records for Another Domain**

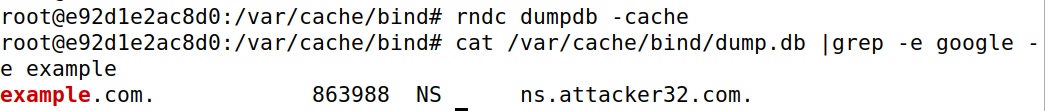
**在task3的基础上，增加一条NS记录，代码如下：**



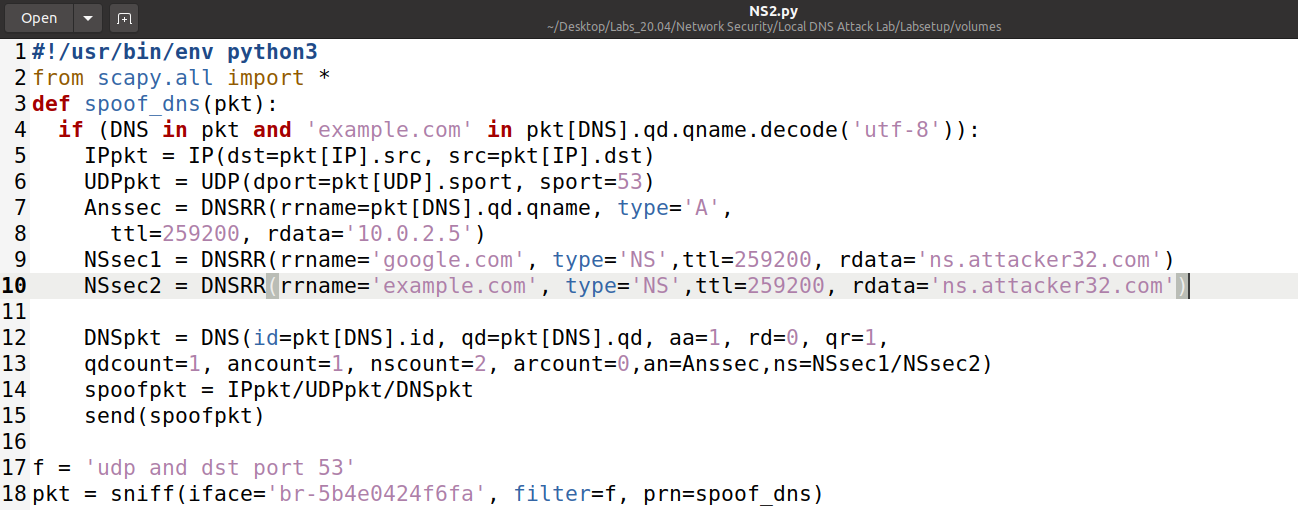
**执行程序，查询example.com：**



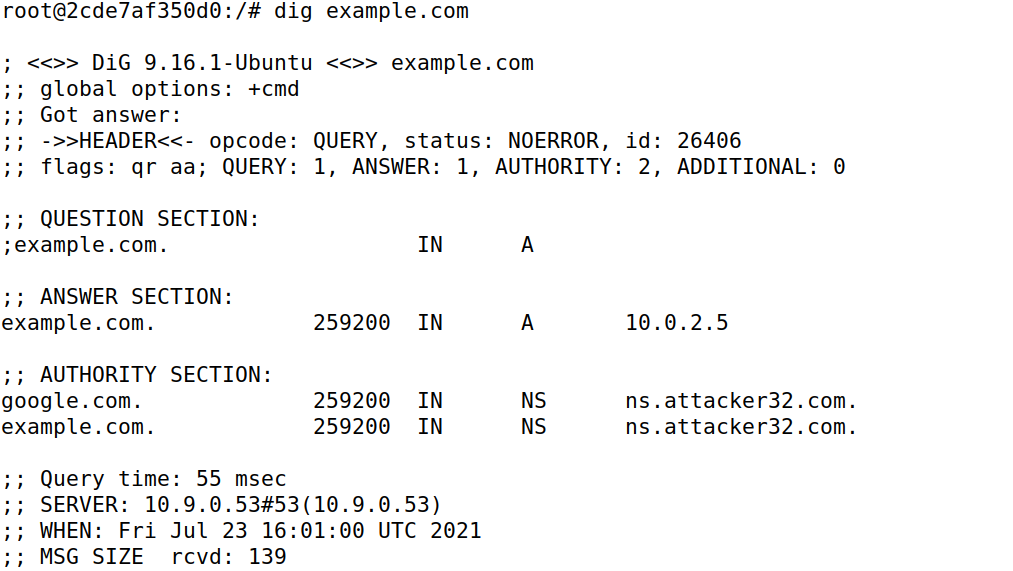
**发现增加了一条google的NS记录，查看缓存：**



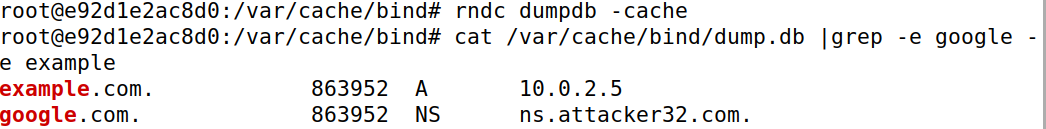
**发现缓存中只有example.com的NS记录，但原始代码中，设置了两条NS记录，于是将代码中这两条换个顺序：**



**然后再次执行dig example.com的命令，**

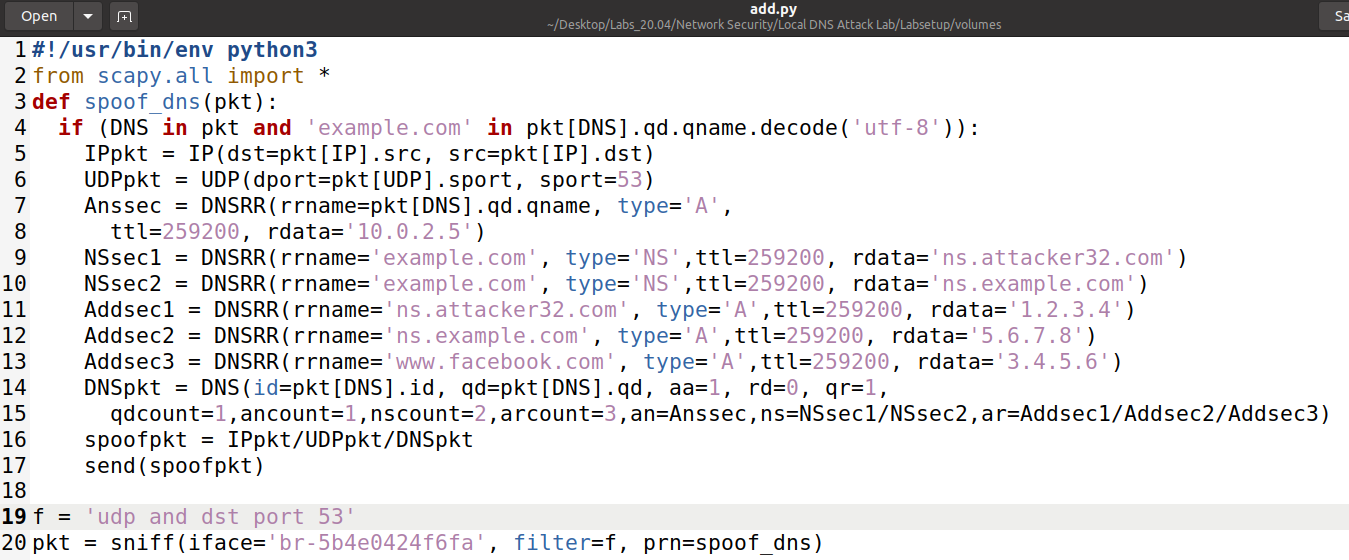


**缓存发现只有一条NS记录，这次为另一条NS记录，可能只会保存一条NS记录，且是排在前面的NS记录：**

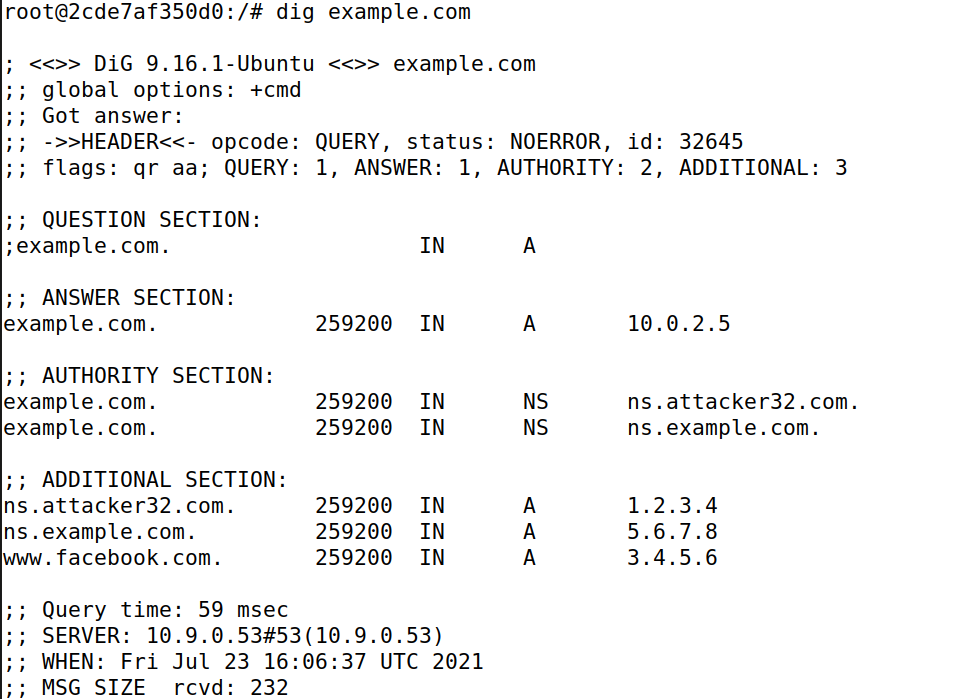


**Task5：Spoofing Records in the Additional Section**

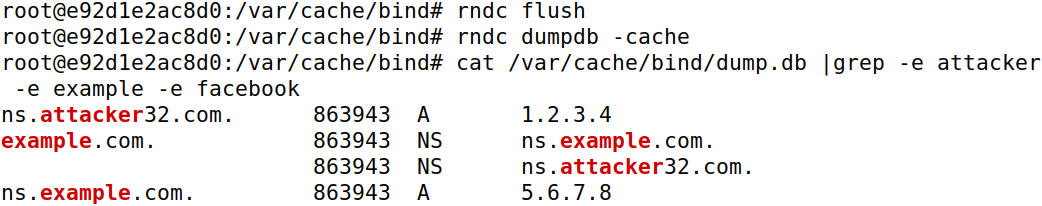
**代码中添加三条附加字段内容，代码如下：**



**执行程序，输出如下：**



**查看缓存:**



**发现在缓存中，只有attack32.com 和ns.example.net 的缓存，附加字段additional 中的记录只有与权威字段authority 中条目相关，才会将其存入到dns 的缓存中。**