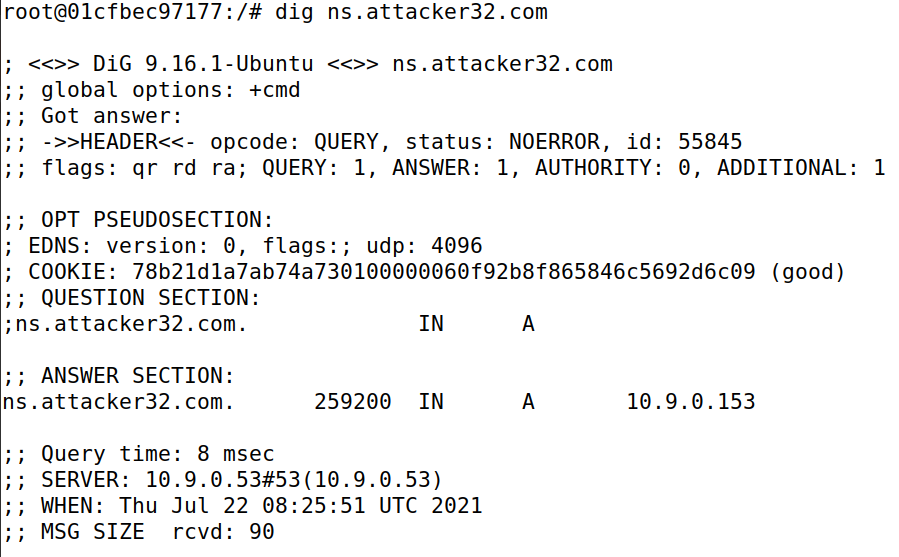
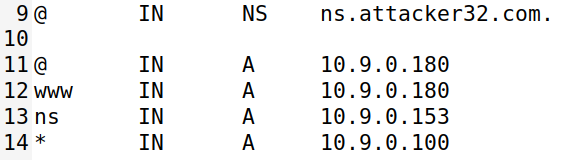
**57118110 杨紫瑄**

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

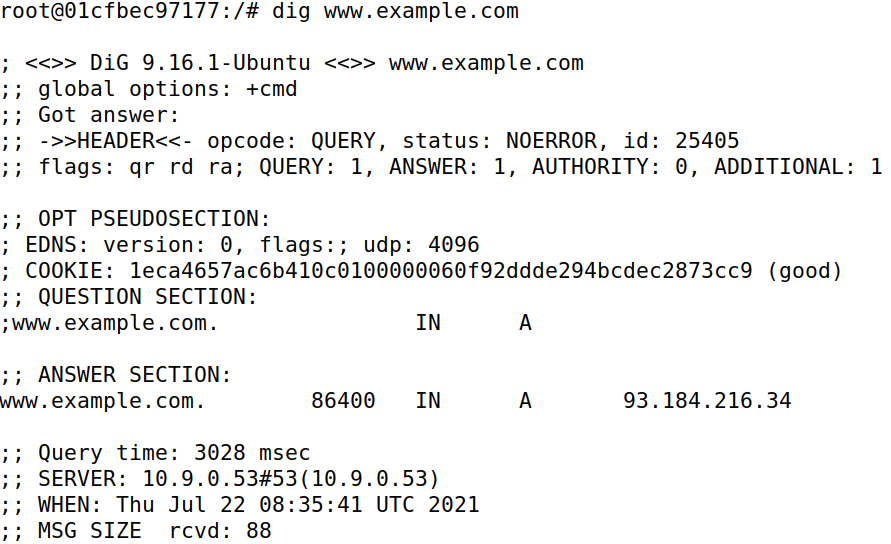


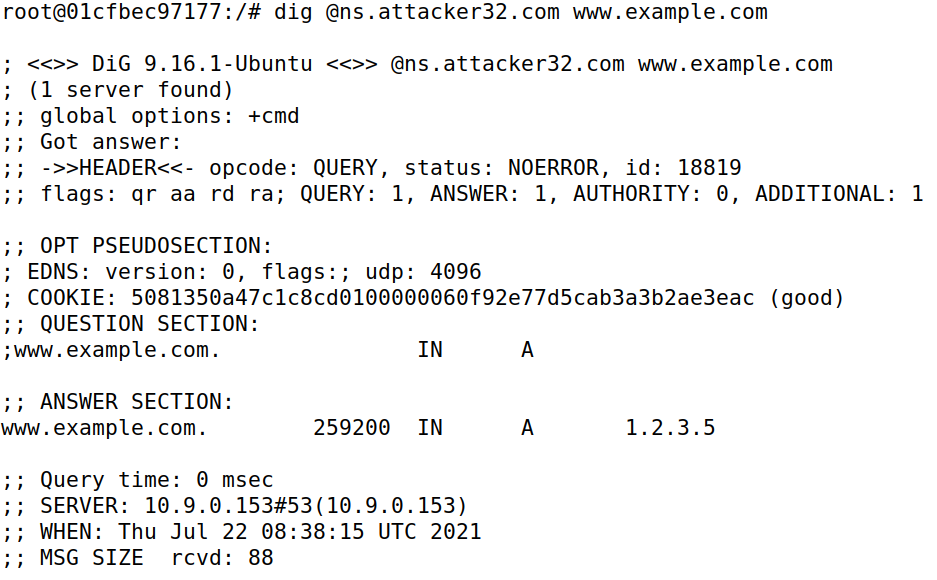
首先测试DNS配置是否正确，使用dig命令查询ns.attacker32.com的地址，发现显示域名指向的ip地址为10.9.0.153



查看攻击者的zone\_attacker32.com文件，发现与记录中的ip地址一致，说明设置没有问题

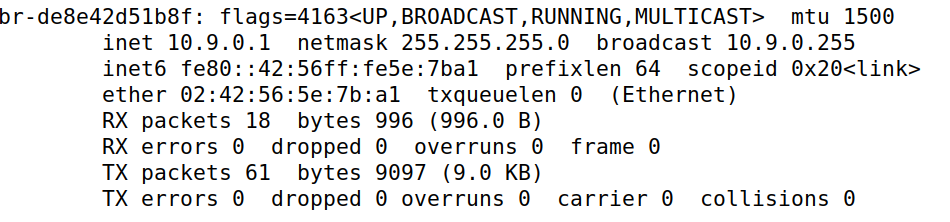
**Get the IP address of** [**www.example.com**](http://www.example.com)**.**



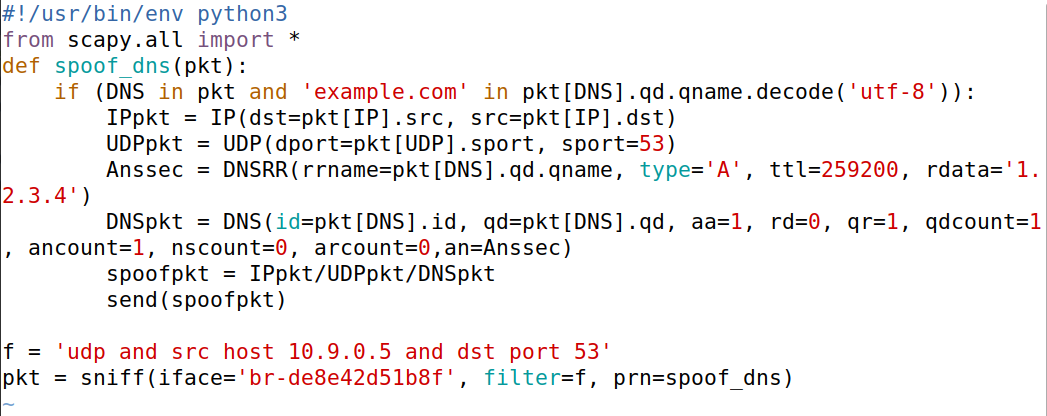


使用dig [www.example.com](http://www.example.com) 和dig @ns.attacker32.com [www.example.com](http://www.example.com)命令，发现二者得到的IP地址不同，其中，第二个命令所得的地址1.2.3.5是攻击者得到的假的地址

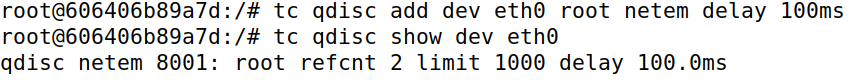
**Task 1 Directly Spoofing Response to User**



首先在攻击者主机上查看10.9.0.0/24网段的端口名称

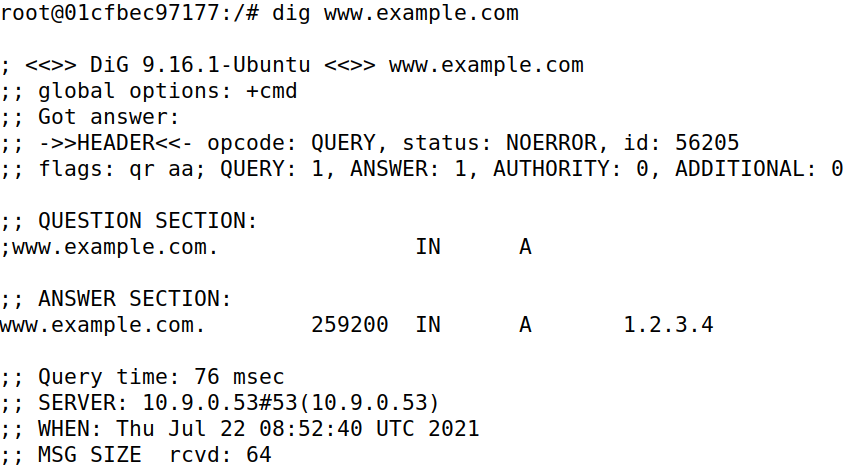


构造攻击代码如上



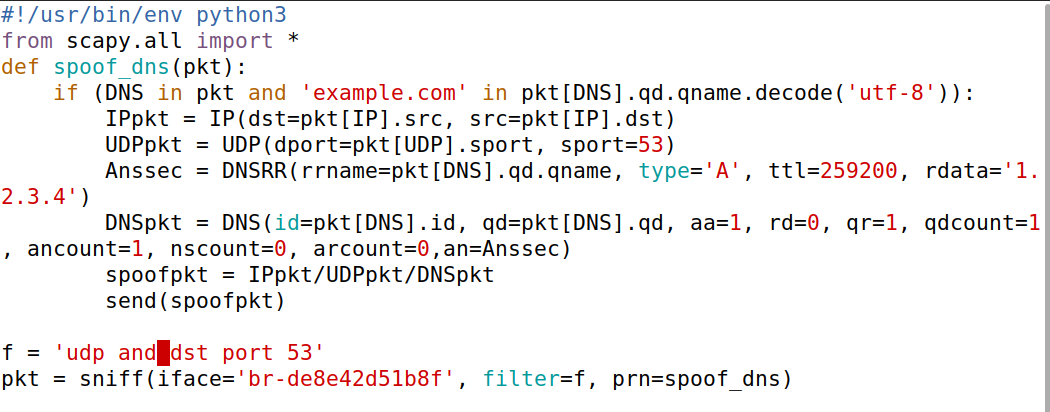
在本地DNS服务器上增加网络的延迟，并使用rndc flush命令清除缓存

执行攻击程序，再次在受害者主机上dig [www.example.com](http://www.example.com)

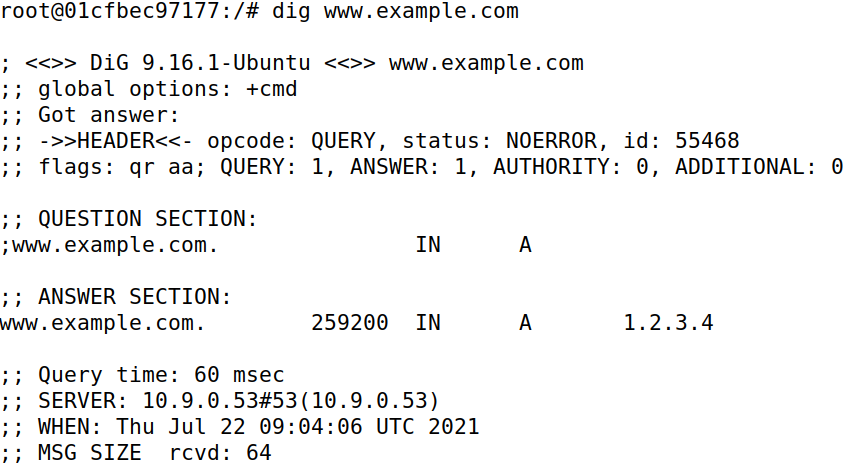


发现攻击后指向的ip地址变成了攻击者伪造的IP地址1.2.3.4，说明攻击成功

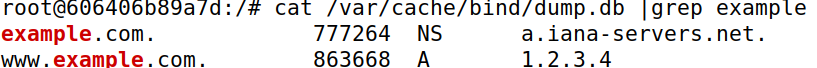
**Task 2 DNS Cache Poisoning Attack – Spoofing Answers**



构造攻击代码如上，与task1中相似，将过滤器改为’udp and dst port 53’



清除本地DNS服务器缓存后，执行攻击程序，dig www.example.com,发现example.com指向的IP地址变成了攻击者伪造的ip地址

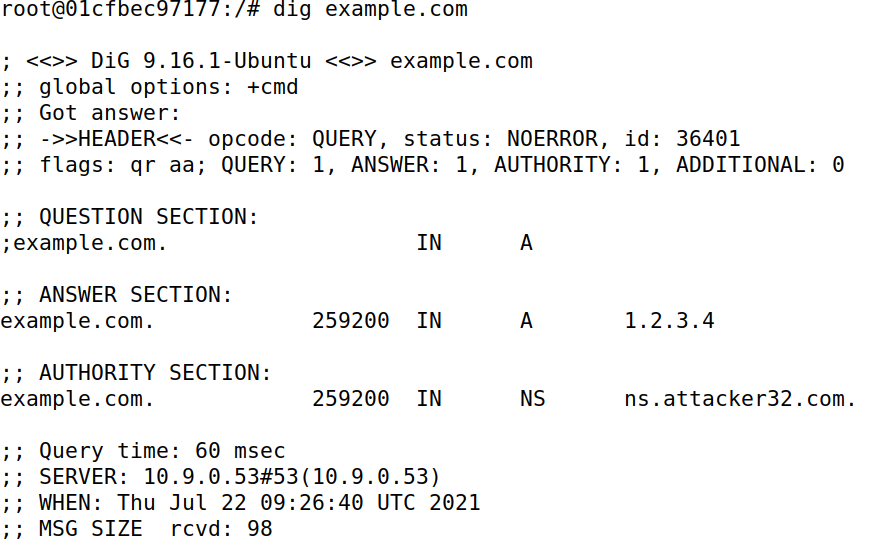


查看本地DNS服务器缓存发现伪造的DNS信息已存储在缓存中，说明DNS缓存中毒攻击成功

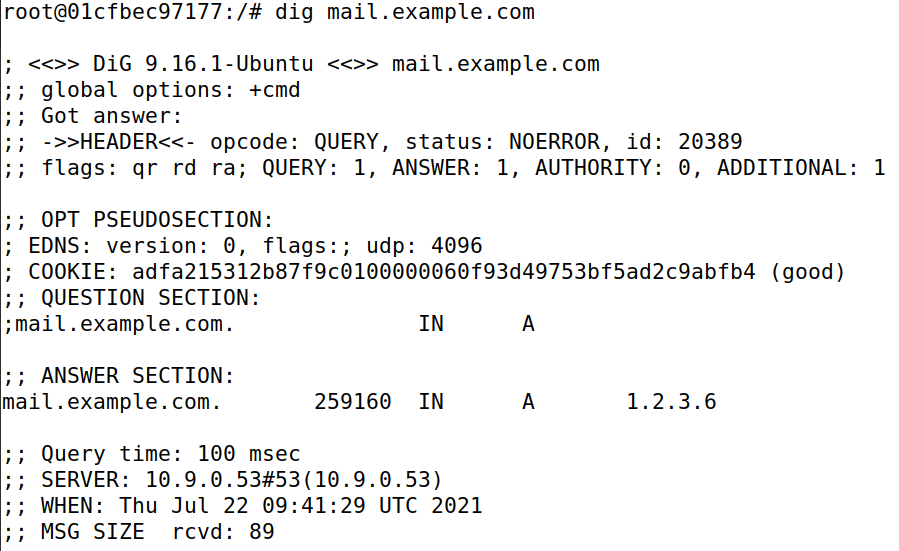
**Task 3 Spoofing NS Records**



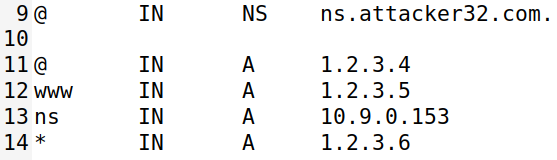
构造攻击代码如上，与Task2相似，增添了一条NS记录内容，将nscount改为1，使example.net域名下所有地址都指向ns.attacker32.com域名



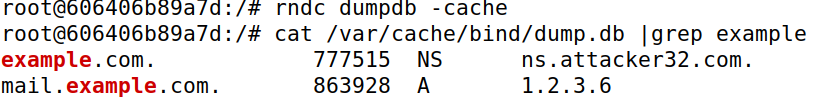
执行攻击程序，并在受害者主机中dig [example.com](http://www.example.com) 发现改地址指向ns.attacker32.com域名，说明成功对域名[example.com](http://www.example.com)污染



停止攻击程序后，查询同一域名下不同主机的信息，如mail.example.com，发现改地址指向1.2.3.6



这与zone\_example.com文件中一致，说明该地址是攻击者伪造的地址

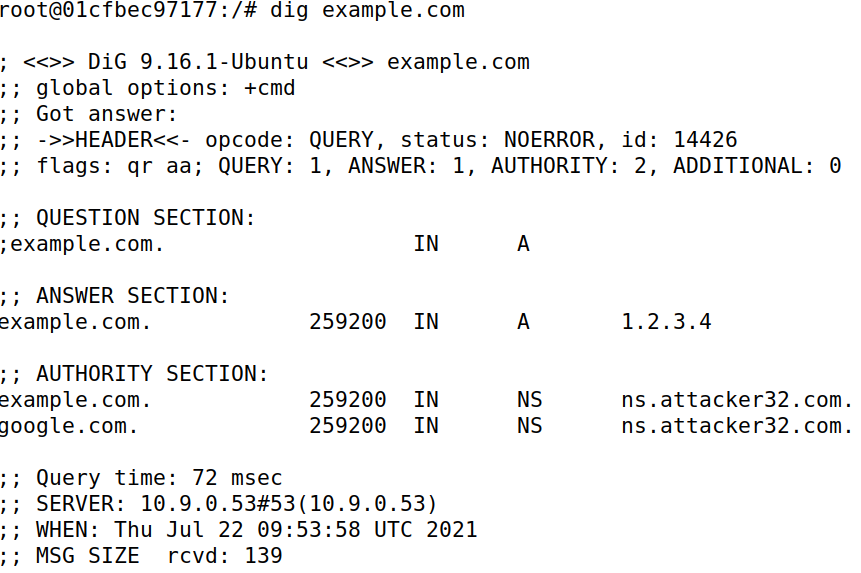


在本地DNS服务器中查看缓存，发现NS记录也在缓存中，说明攻击成功

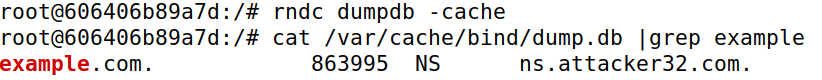
**Task 4 Spoofing NS Records for Another Domain**



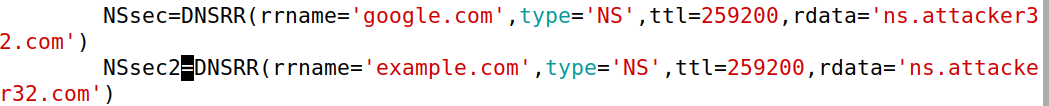
构造攻击代码如上，与Task3相似，增加一条NS记录，使google.com域名下的地址都指向ns.attacker32.com

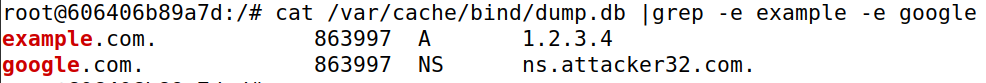


执行程序后，清除本地DNS缓存，在受害者主机中dig example.com，发现增加了一条权威记录



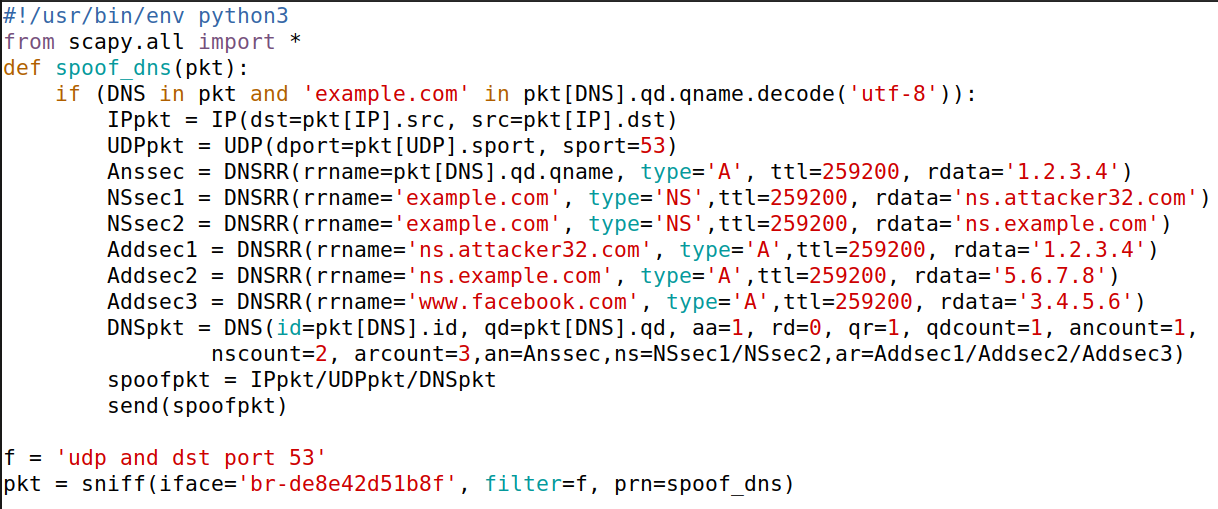
但是此时发现本地DNS缓存中只有example.com的记录，调换攻击代码中两条NS代码顺序后再次尝试



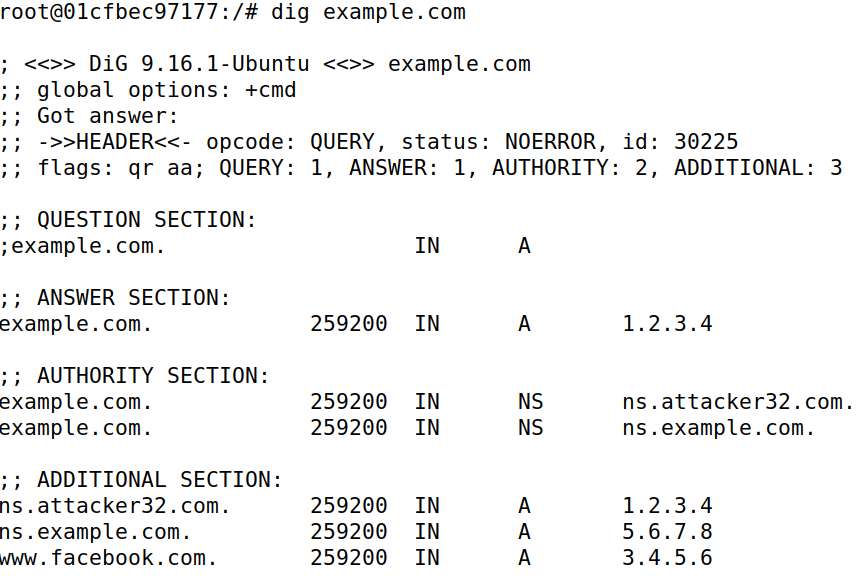


发现出现的换成google.com的NS记录了，但是没有example.com的记录，出现这种结果推测本地DNS缓存只会保存一条排在前面的权威字段的NS记录

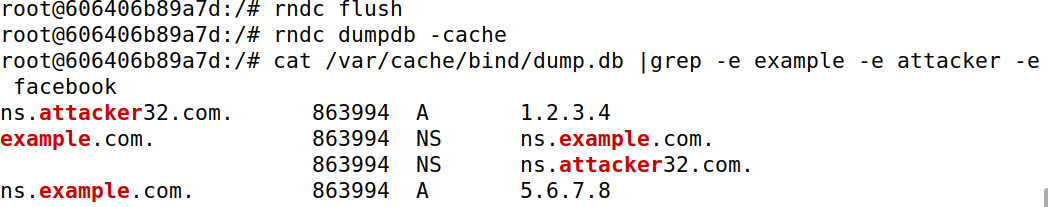
**Task 5 Spoofing Records in the Additional Section**



构造攻击代码如上，与Task3中相似，增加三条附加字段的内容以及一条NS记录内容



清理本地DNS服务器缓存后，执行攻击程序后在受害者主机中dig example.com显示出权威条目以及附加条目如上所示



查看本地DNS缓存如上所示，发现缓存中只有attack32.com和ns.example.net，而[www.facebook.com](http://www.facebook.com)的记录不会被缓存，推测这是因为附加字段条目中的记录与权威条目相关，只有记录存在于权威条目中时才会被存到DNS缓存中