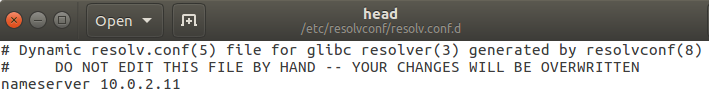
**实验1：Local DNS Attack Lab**

用户10.0.2.8，DNS 服务器10.0.2.11，攻击者10.0.2.7

## Task 1: Conﬁgure the User Machine

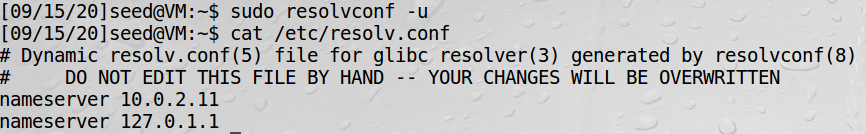
在配置文件中添加我们定义的DNS服务器的信息







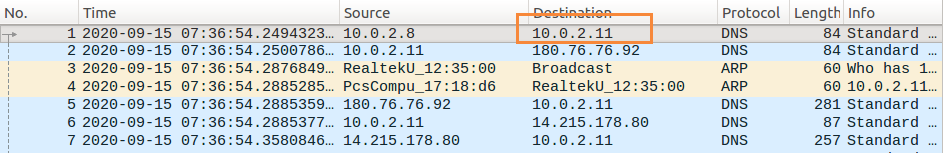
我们刚刚添加的信息出现在resolv.conf的第一行



使用dig命令验证设置是否成功



设置成功

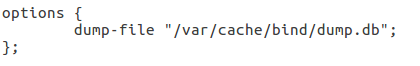


## Task 2: Setup a Local DNS Server

① 配置BIND 9服务器



添加以下内容

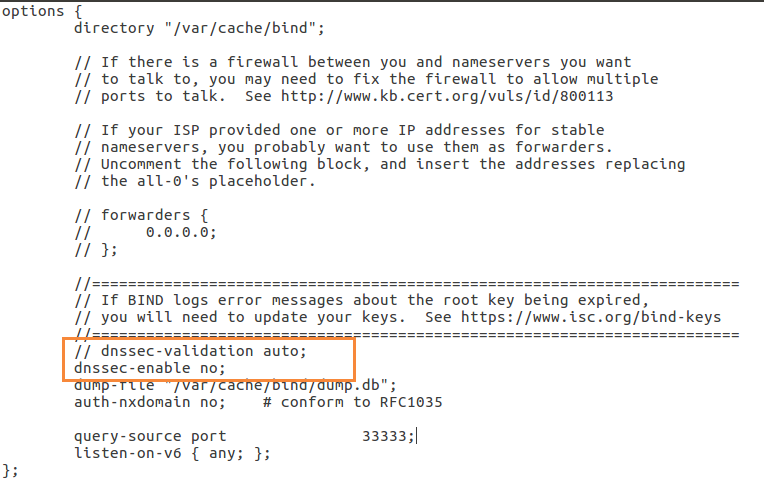


更新cache的内容



② 关闭DNSSEC

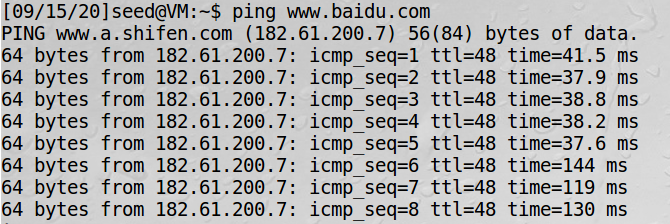
修改named.conf.options文件

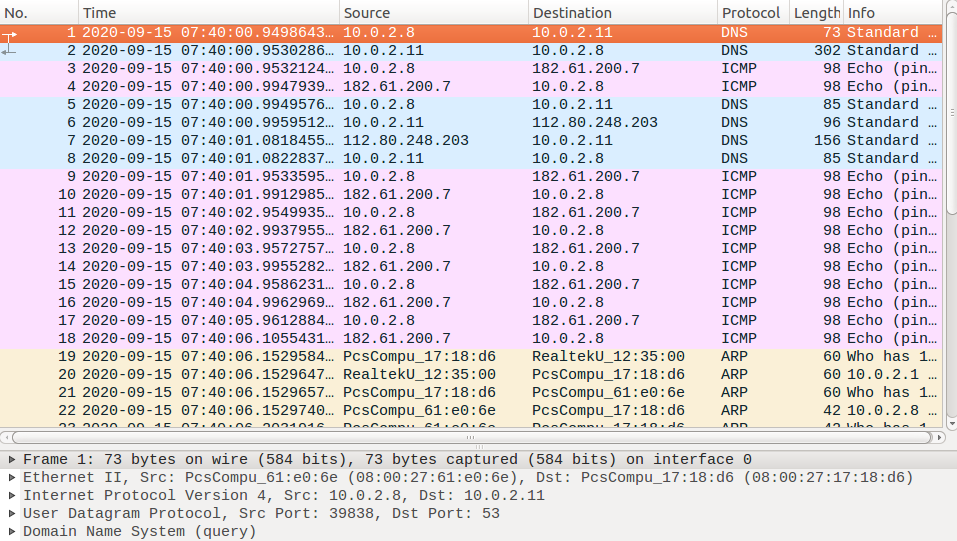


③ 打开DNS服务器



④ 使用DNS 服务器

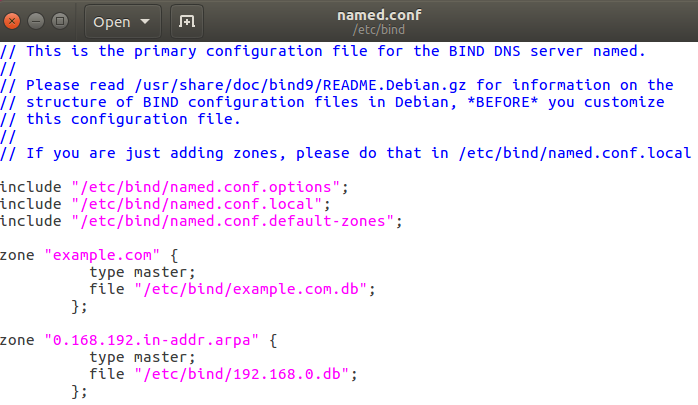




## Task 3: Host a Zone in the Local DNS Server

① Create zones





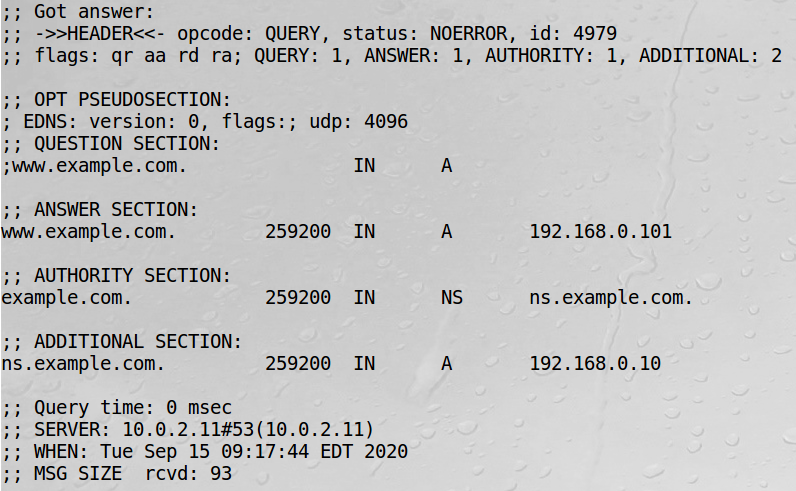
②③ Setup the forward && reverse lookup zone ﬁle

将下面两个文件放到/etc/bind目录下





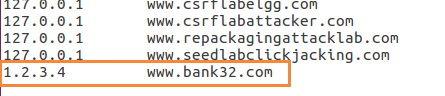
④ Restart the BIND server and test



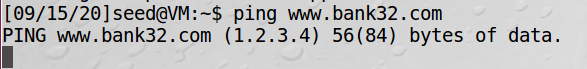
说明有效

## Task 4: Modifying the Host File

在被攻击主机的/etc/hosts文件中添加一项

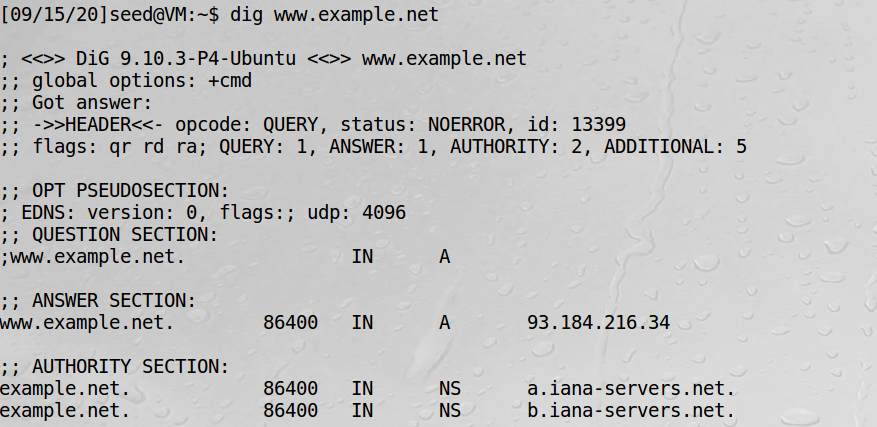


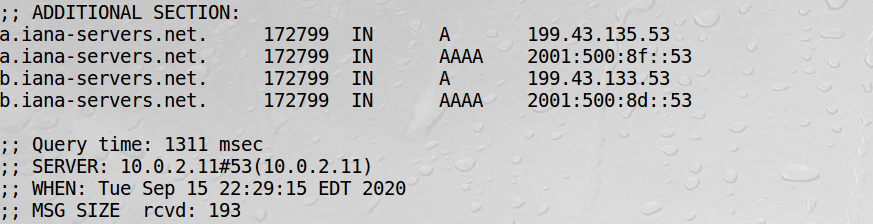
发现可以把www.bank23.com解析到1.2.3.4



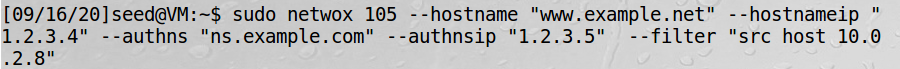
## Task 5: Directly Spooﬁng Response to User

攻击之前，用户dig www.example.net，得到的是正常的响应

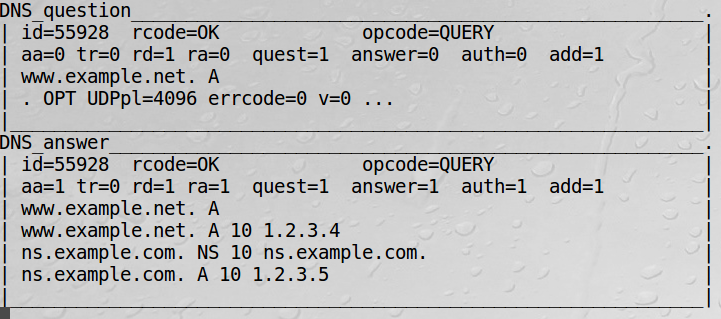


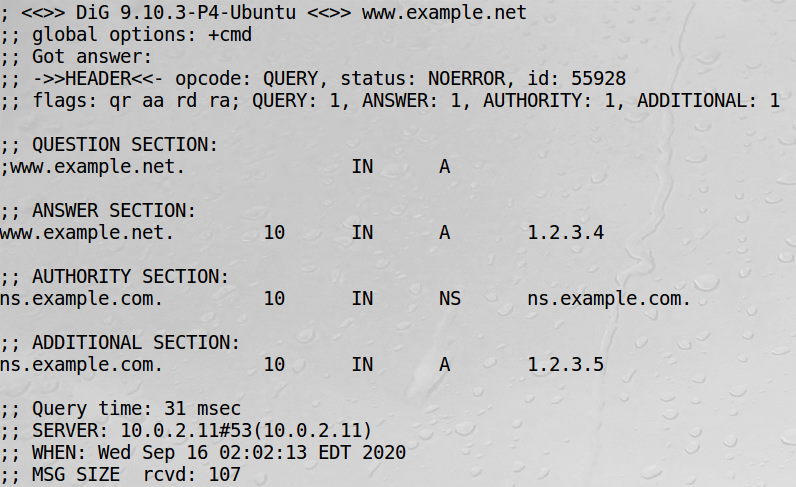


构造以下命令进行攻击



运行攻击命令后，得到了我们伪造的响应



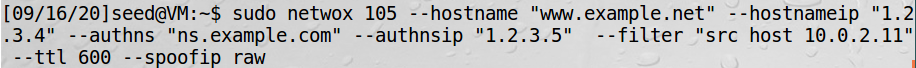


## Task 6: DNS Cache Poisoning Attack

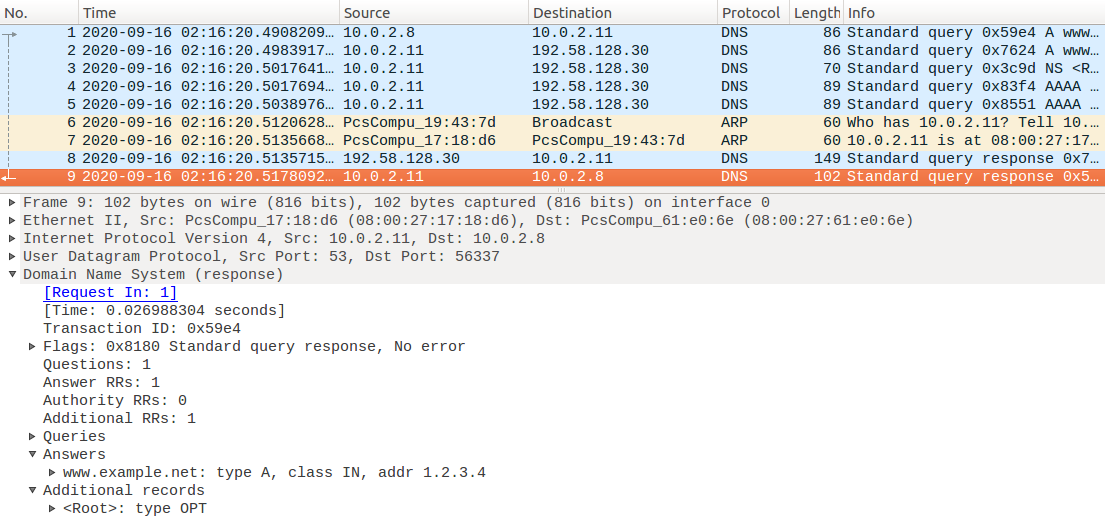
清空域名服务器的dns缓存



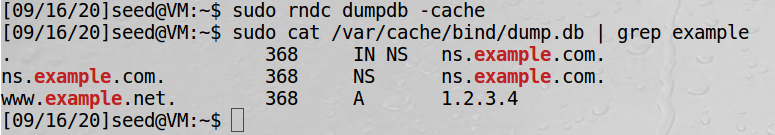
在攻击者处运行以下命令



在用户处dig www.example.net，并用wireshark抓包，可以观察到我们伪造的响应报文

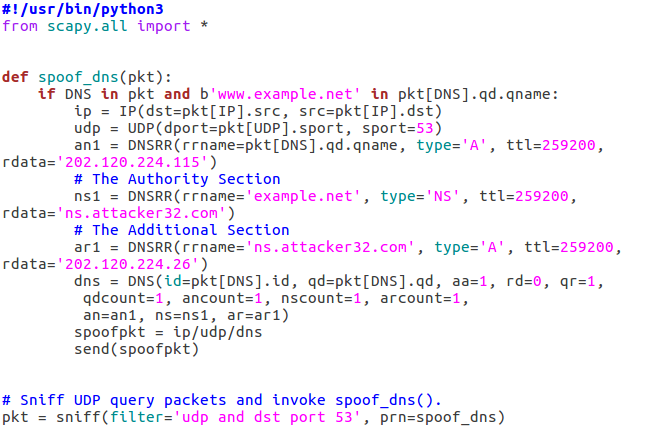


这时我们查看域名服务器的缓存信息，可以看到我们已经成功污染了缓存

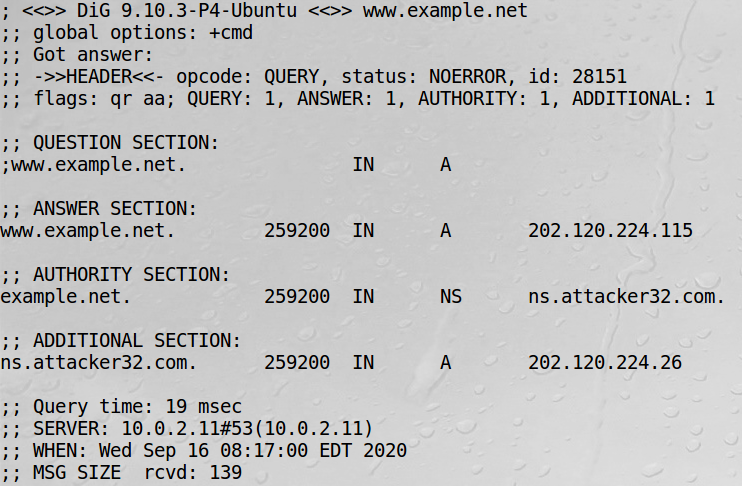


## Task 7: DNS Cache Poisoning: Targeting the Authority Section

攻击代码如下

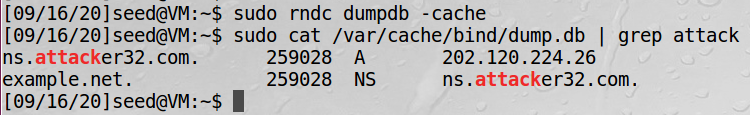


在攻击者主机上运行攻击代码，然后在用户主机上dig www.example.net，得到的输出如下



可以发现authority section已经变成了ns.attacker32.com

我们伪造的信息也已经存在了域名服务器的缓存中



然后我们在用户主机dig mail.example.net，并用wireshark抓包，可以看到发往ns.attacker32.com的报文

