**第五次网安综合课程设计报告**

**（TCP/IP Attack Lab）**

57117228 陈佑

IP地址：

VM A 10.0.2.4 攻击者

VM B 10.0.2.5 受害者

VM M 10.0.2.6 DNS服务器

DNS攻击操作以各种方式执行DNS解析过程，目的是误导用户到其他目的地。

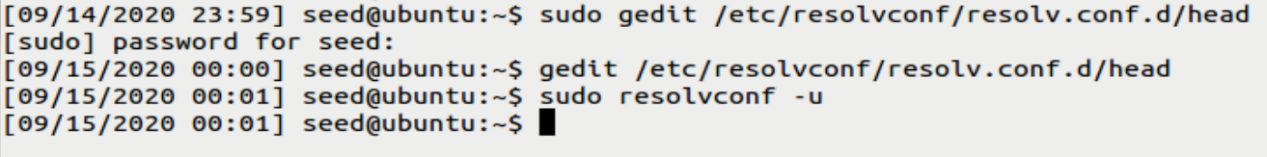
在本章实验中，我们需要搭建一个DNS服务器，针对这个DNS服务器进行攻击。实验需要三个独立的机器：一个用于攻击者（VM A）一个用于受害者（VM B），一个用于DNS服务器（VM M）。

# Configure the User Machine（配置用户主机）

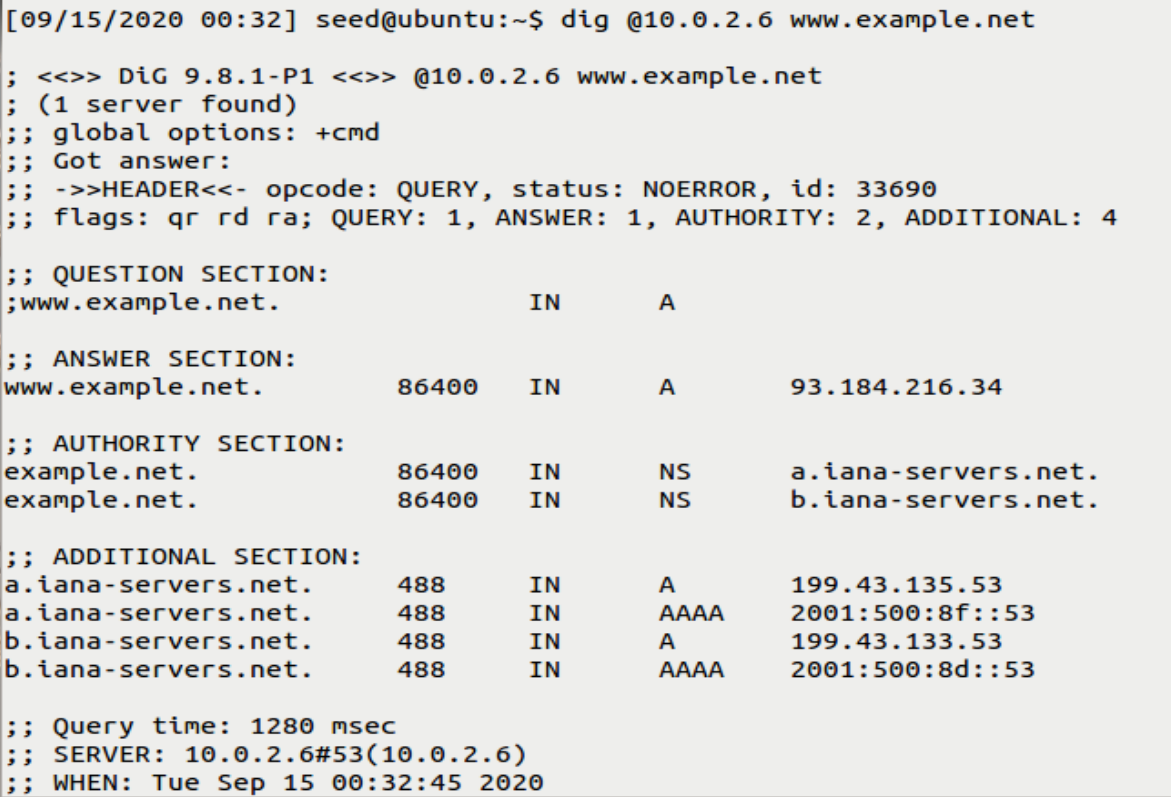
需要修改用户机的配置文件，将其本地DNS解析服务器设置为我们搭建的DNS服务器。

启动三台VM，然后在VM B中修改B使用的DNS服务器，sudo gedit /etc/resolvconf/resolv.conf.d/head，将nameserver 改为10.0.2.6。





配置结束后使用dig @10.0.2.6 [www.example.net查询](http://www.example.net查看)www.example.net对应的IP地址。

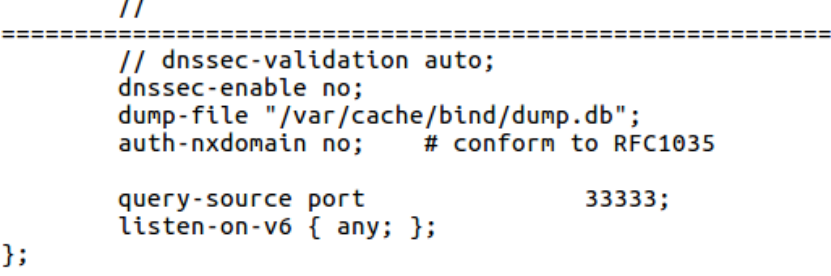


可以看到，在ANSWER SECTION中返回了www.example.net的IP地址——93.184.216.34。同时，返回信息的server为10.0.2.6，用户主机配置DNS服务器成功。

# Set up a Local DNS Server（搭建DNS服务器）

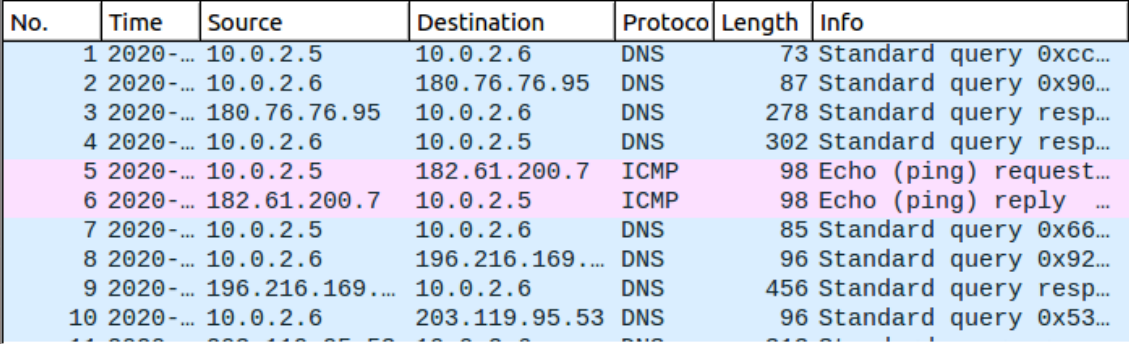
在本次实验中，我们使用BIND (Berkeley Internet Name Domain)搭建一个DNS服务器。

### Step1.配置BIND 9服务器，关闭DNSSEC保护机制



### Step2.重启BIND 9服务器后ping www.baidu.com

在用户主机VM B上ping www.baidu.com，并利用wireshark抓包，抓包结果如下：

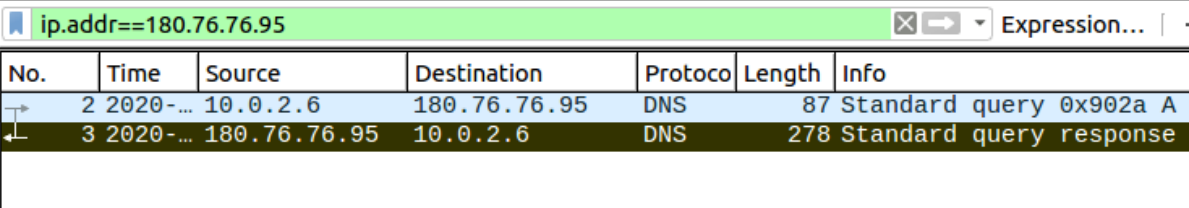


（报文1~4）可以看到，VM B（10.0.2.5）第一次ping [www.baidu.com时，需要先向本地DNS服务器（10.0.2.6）发送DNS解析请求，本地DNS服务器再向180.76.76.95发送DNS解析请求。180.76.76.95将www.baidu.com域名解析的IP地址返回给向本地DNS服务器（10.0.2.6），本地DNS服务器再将此IP地址返回给VM](http://www.baidu.com时，需要先向本地DNS服务器（10.0.2.6）发送DNS解析请求，本地DNS服务器再向180.76.76.95发送DNS解析请求。180.76.76.95将www.baidu.com域名解析的IP地址返回给向本地DNS服务器（10.0.2.6），本地DNS服务器再将此IP地址返回给VM) B主机。

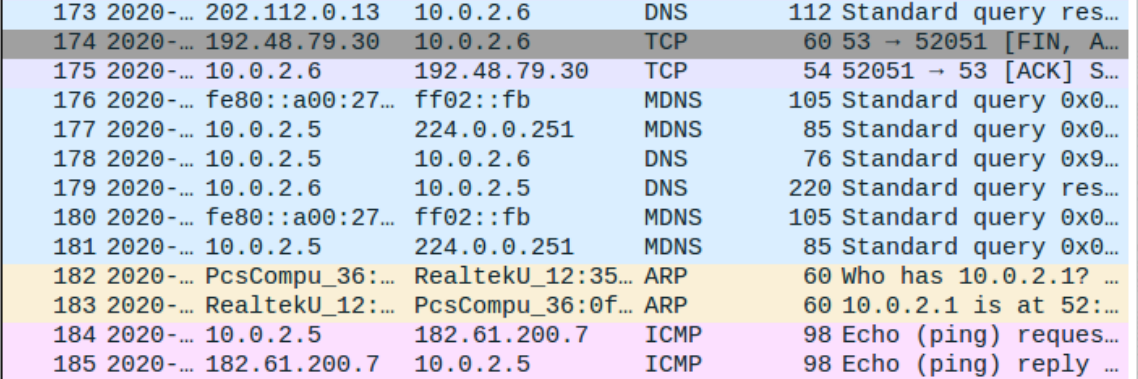
（报文5~6）VM B和www.baidu.com传输ICMP报文。

以上是一个完整的DNS域名解析与ping的过程。但是，由于DNS服务器缓存的存在，可以将已经解析过的域名及其IP地址存放在本地DNS服务器中，因而下一次ping的时候就不需要再重复域名解析了。

对发往DNS服务器（180.76.76.95）的报文过滤后可以看出，只有第一次发送ICMP报文之前，本地DNS服务器需要向180.76.76.95发送DNS解析请求。之后就没有报文发往DNS服务器（180.76.76.95）了。



当VM B第二次向www.baidu.com发送报文的时候，可以直接使用DNS服务器中缓存的IP地址。



**所以，在第二次向（已经解析过的）目标域名发送报文时，就可以直接使用DNS缓存了。**

# Host a Zone in the Local DNS Server

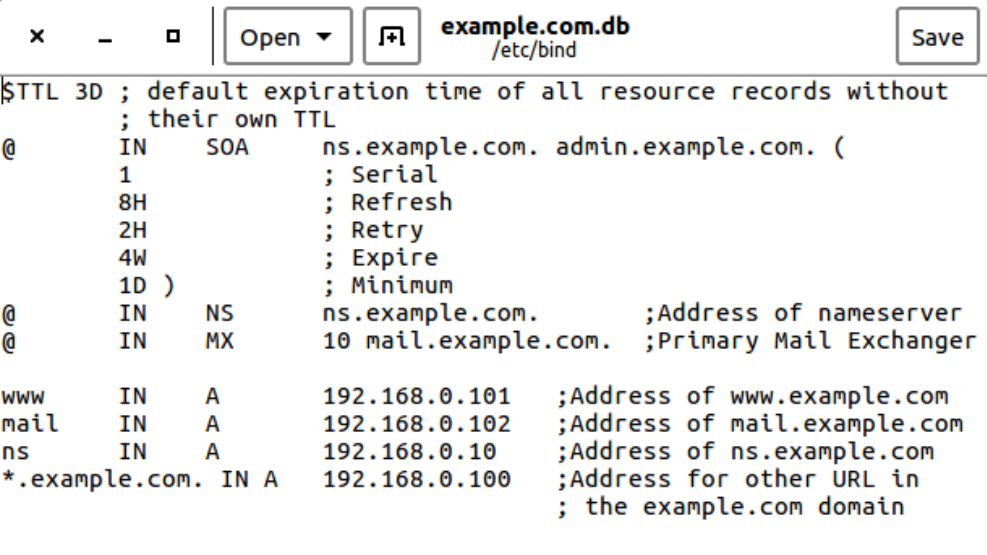
### Step1.创建zone

我们需要通过添加内容到/etc/bind/name.conf，来在DNS服务器中创建两个区域条目。第一个zone用于转发查找(从主机名到IP)，第二个zone用于反向查找(从IP到主机名)。



### Step2.Setup the forward lookup zone file

创建第一个zone（转发查找）对应的文件example.com.db，存放example.com下所有域名的记录。

****

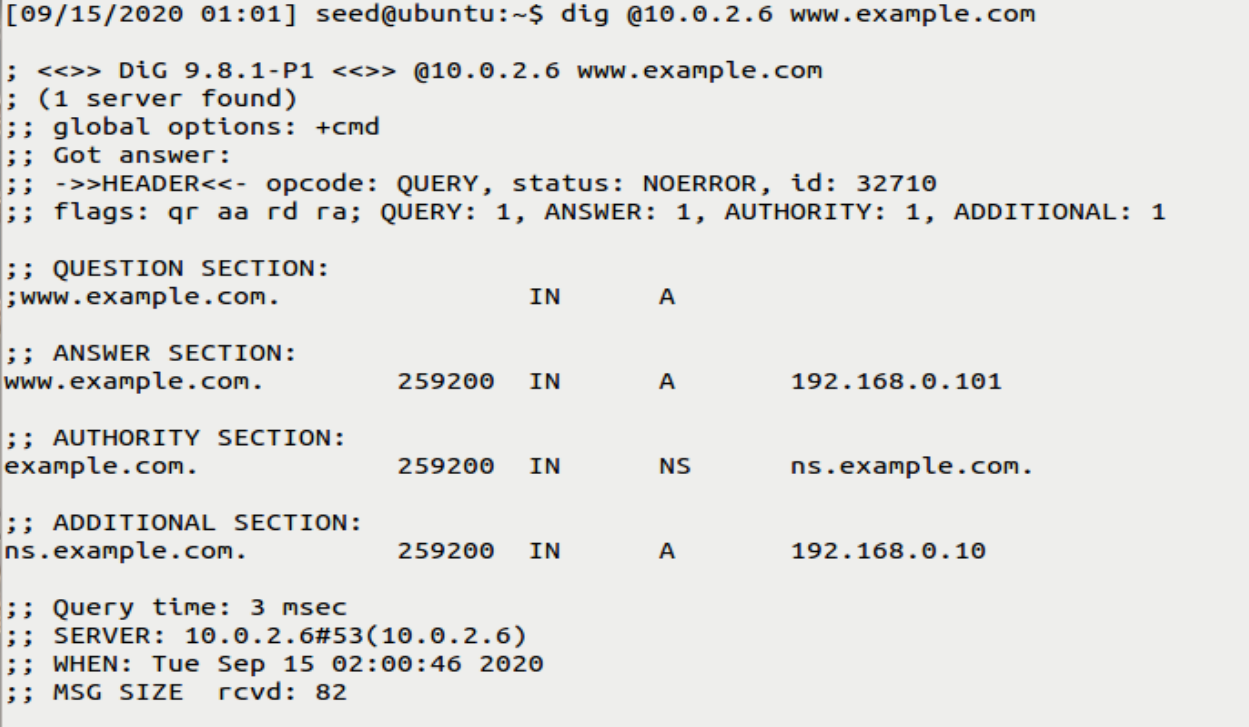
### Step3. Set up the reverse lookup zone file

创建第二个zone（反向查找）对应的文件192.168.0.db。



### Step4. Restart the BIND server and test

重新启动BIND 9服务器后，在用户机上执行dig @10.0.2.6 www.example.com解析域名www.example.com。

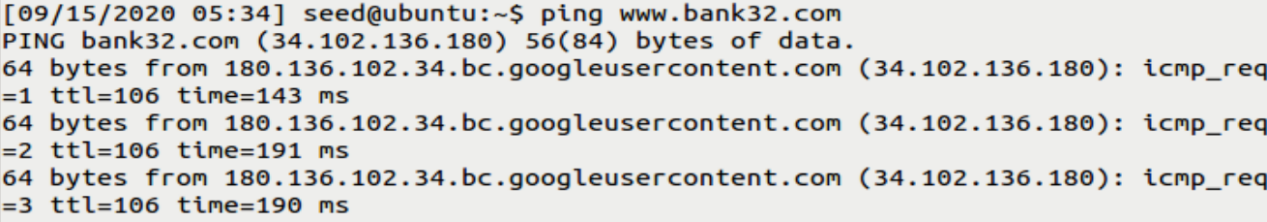


可以看到，此时dig @10.0.2.6 [www.example.com的ANSWER](http://www.example.com的ANSWER) SECTION中直接显示了www.example.com的解析结果IP地址。这是因为，在解析域名时直接查询example.com.db文件，查找其下所有域名的记录，所以直接返回www.example.com对应的IP地址。

# Modifying the Host File

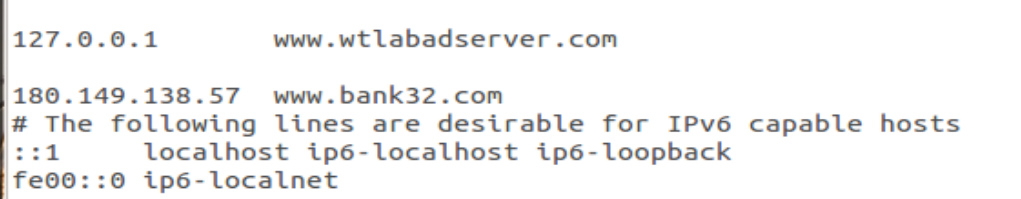
机器上的hosts文件记录了主机名到IP地址的映射，当此记录在hosts文件中存在时，就不会发出DNS查询请求。所以，攻击者可以修改hosts文件中主机名到IP地址的映射，使得当用户访问www.bank32.com时，访问会被重定向到恶意地址。

### Step1.修改/etc/hosts文件之前ping www.bank32.com



www.bank32.com经过DNS服务器解析之后的IP地址为180.136.102.34。

### Step2.修改/etc/hosts文件之后ping www.bank32.com



在/etc/hosts文件中添加一条错误的映射关系，使得180.149.138.57（实际为www.weibo.com的IP地址）映射到www.bank32.com



可以看到，修改了hosts文件后，www.bank32.com经过DNS服务器解析之后的IP地址为180.136.102.34。

# Directly Spoofing Response to User

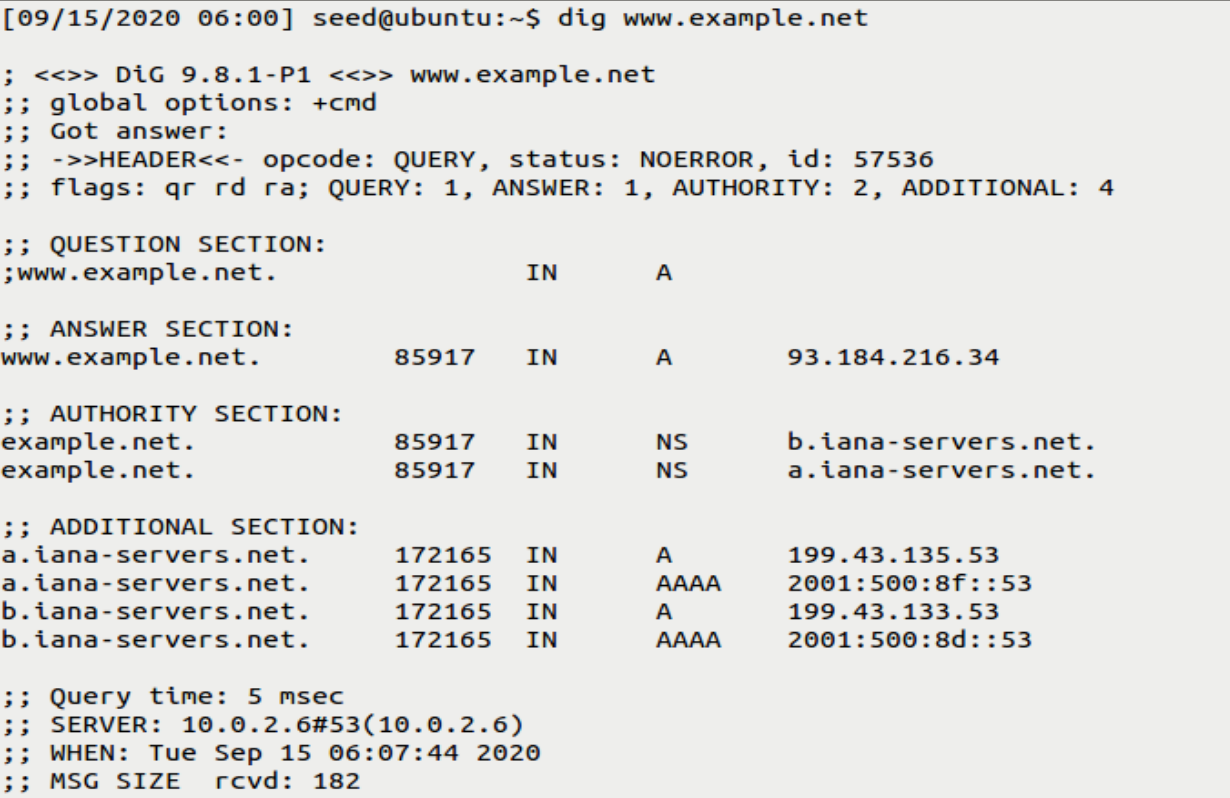
本task中，攻击者不能直接修改用户主机的hosts文件，但是会和用户处于一个局域网上，通过对局域网上的DNS请求报文（源地址为用户主机的DNS请求报文）进行嗅探，来伪造一个假的DNS响应报文。

Netwox工具105提供了进行这种嗅探和响应的工具。 我们可以在回复数据包中组成任意的DNS回应。此外，我们可以使用“filter”选项来指定要嗅探的数据包类型。例如，通过使用src host 10.0.2.5，我们可以将嗅探的范围限制为仅来自主机10.0.2.5的数据包。

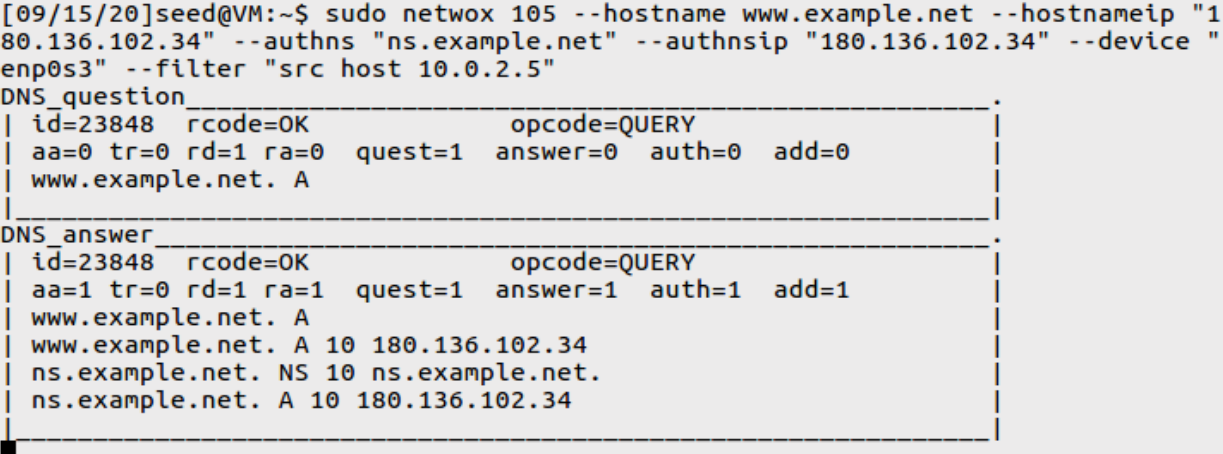
在VM A主机上运行netwox命令，在VMB 主机上ping www.example.net则可看到请求重定向到了另一个IP地址上。而当VM A上不运行该程序时，会直接返回未知主机ping: unknown host [www.example.net。](http://www.example.net。)

### Step1.攻击前，VM B发送DNS请求报文，请求解析域名www.example.net

用户主机VM B处收到了回应，DNS查询结果为93.184.216.34。



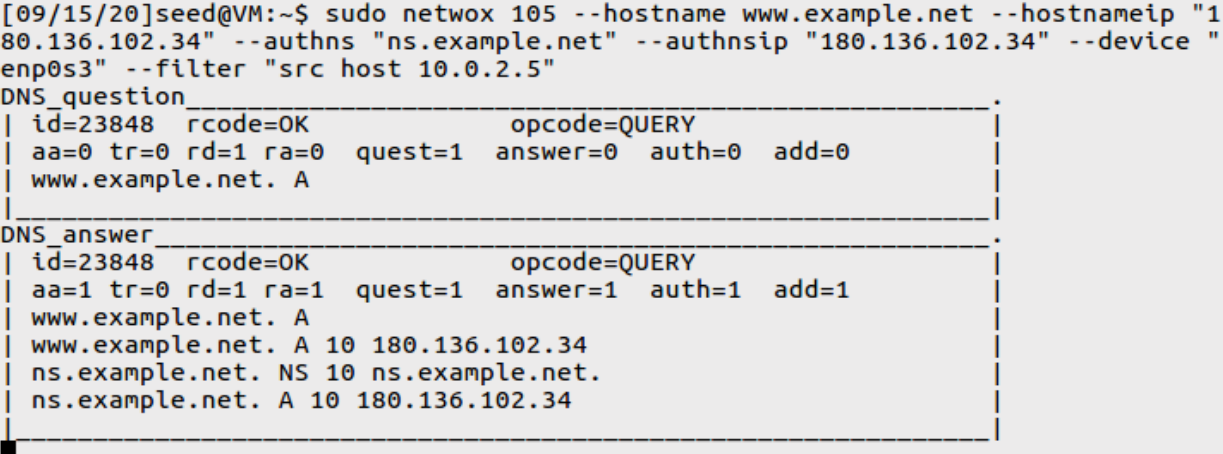
### Step2.VM A发送netwox指令



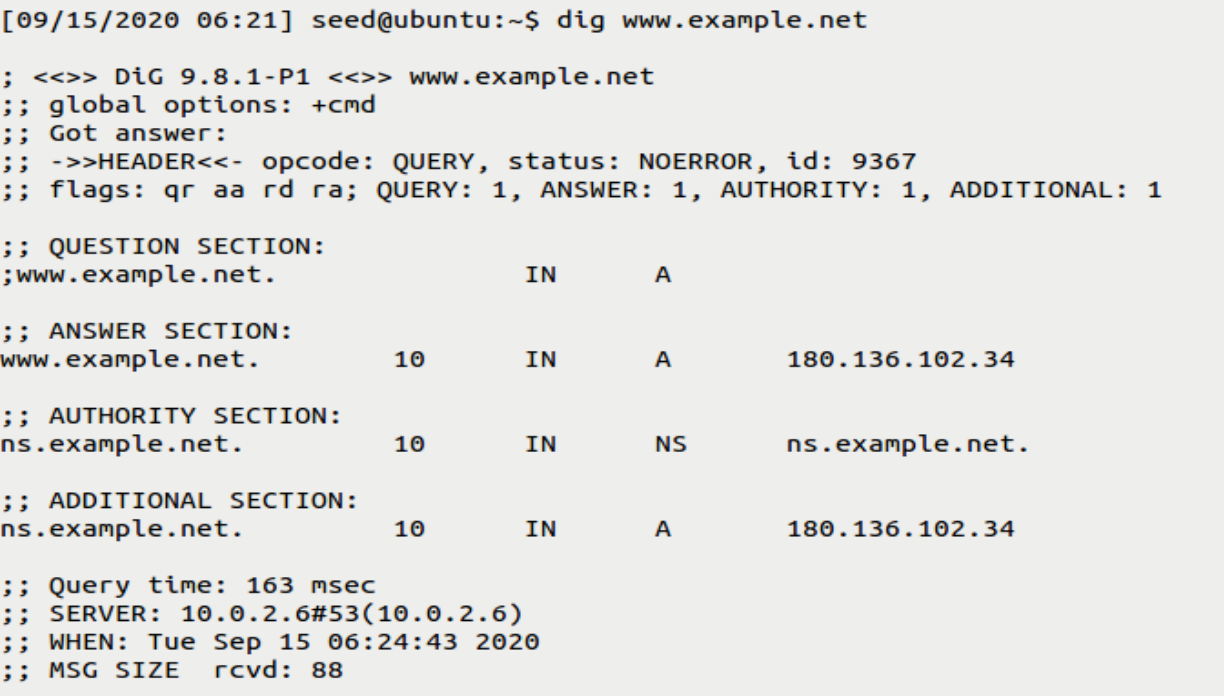
指令的含义是，当在局域网中嗅探到来自用户主机10.0.2.5的DNS查询报文，且报文要解析的域名为www.example.net的时候，返回一个解析内容为“180.136.102.34”的DNS响应报文。

### Step3.VM B发送DNS请求报文，请求解析域名www.example.net

此时，攻击者VM A终端上显示了嗅探到的DNS请求报文以及伪造的DNS响应报文。



用户主机VM B处收到了回应，DNS查询结果为180.136.102.34（www.weibo.com的真实IP地址），攻击成功。



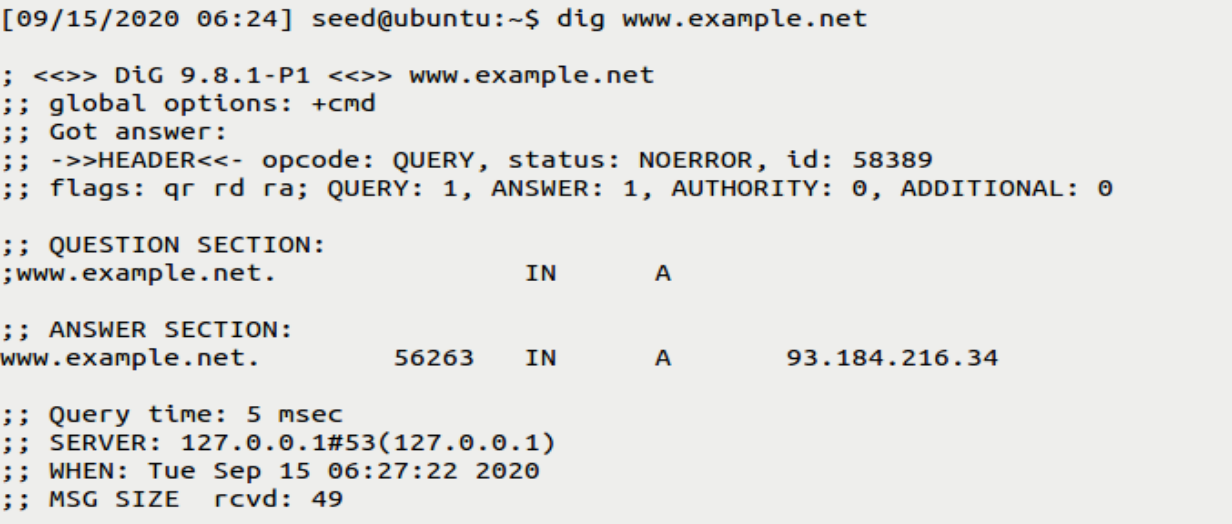
# DNS Cache Poisoning Attack

以上的攻击方法有一个很明显的缺点，每次用户查询都需要回复一个伪造的DNS回应包。所以接下来我们要去欺骗DNS服务器，然后让其缓存这个映射，之后每次都会从缓存中拿出结果回应给请求DNS查询的用户。这就是 DNS Cache Poisoning Attack。

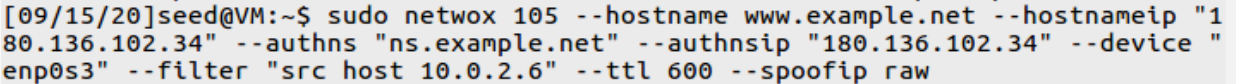
首先我们要修改Attacker的命令，将源主机IP改为DNS服务器的IP地址10.0.2.6，我们还使用ttl字段来表示我们希望假答案在DNS服务器的缓存中保留多长时间。DNS服务器中毒后，我们可以停止Netwox 105程序。

### Step1.攻击前，DNS服务器清除缓存（sudo rndc flush），VM B发送DNS请求报文，请求解析域名www.example.net

用户主机VM B处收到了回应，DNS查询结果为93.184.216.34。



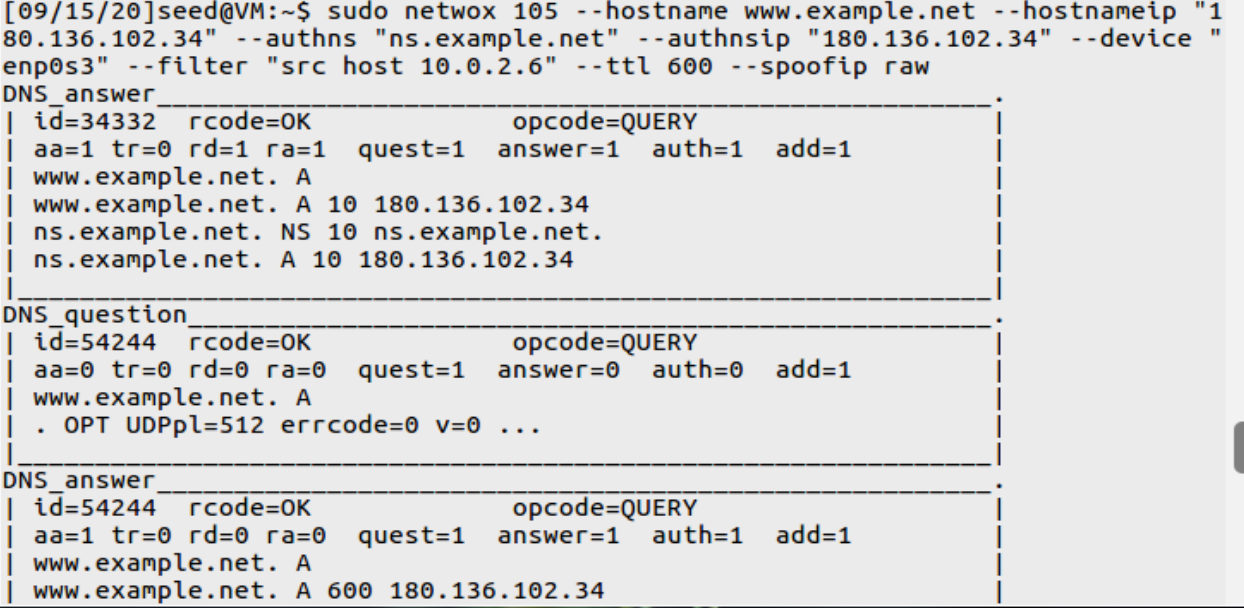
### Step2.VM A发送netwox指令



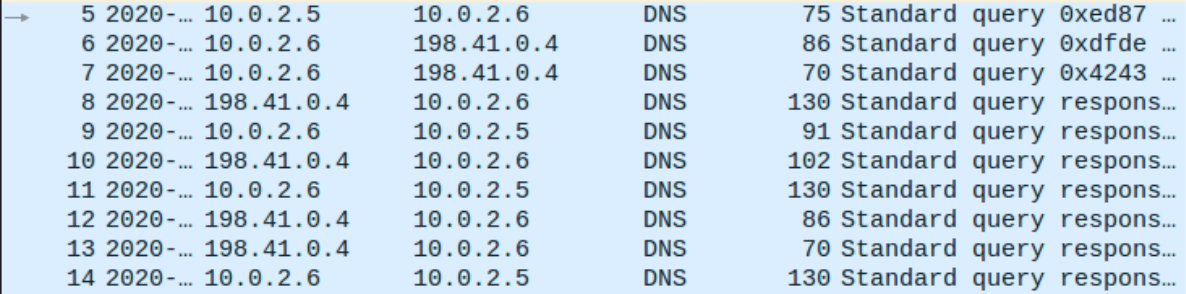
指令的含义是，当在局域网中嗅探到来自DNS服务器10.0.2.6的DNS查询报文，且报文要解析的域名为www.example.net的时候，返回一个解析内容为“180.136.102.34”的DNS响应报文。

### Step3.VM B发送DNS请求报文，请求解析域名www.example.net

此时，攻击者VM A终端上显示了嗅探到的来自DNS服务器的DNS请求报文以及伪造的DNS响应报文。

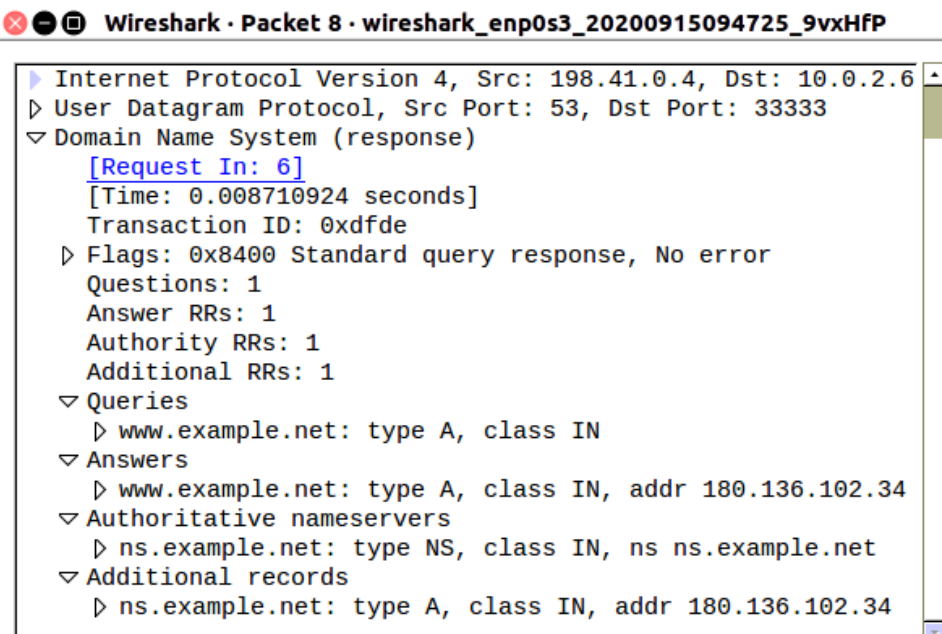


在用户主机VM B上用wireshark抓包，可以观察到整个DNS解析的报文交换流程。



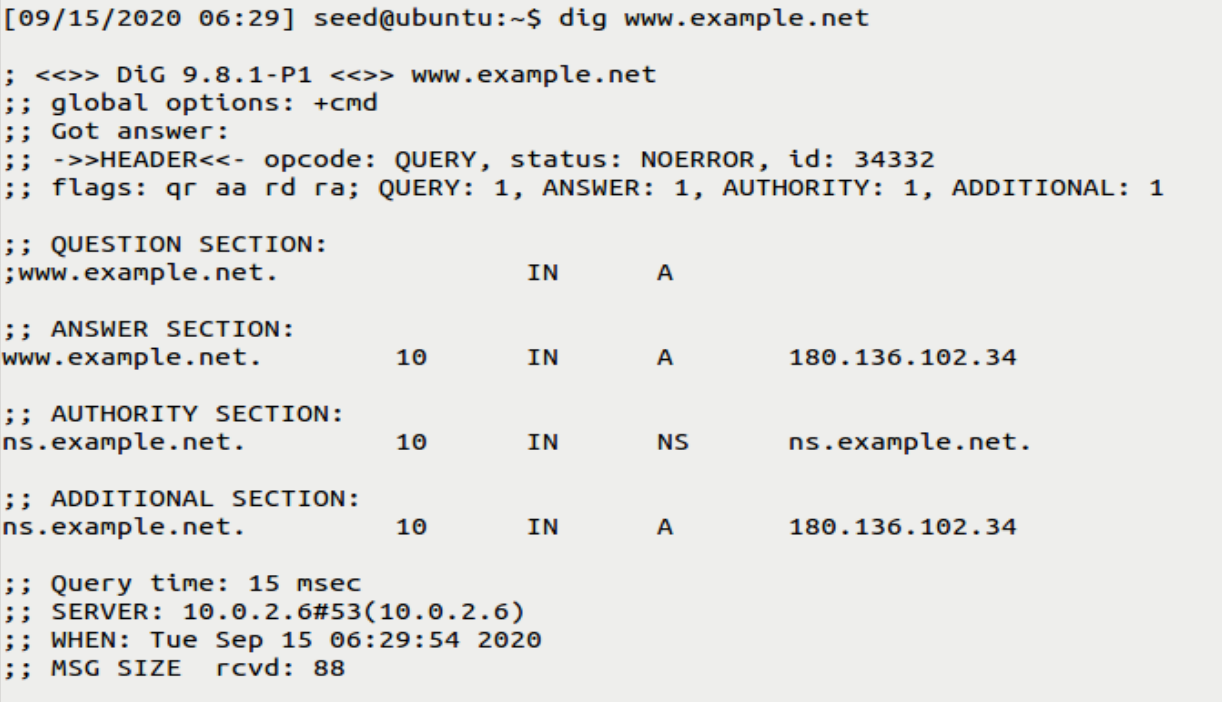
可以看到，用户VM B（10.0.2.5）首先向本地DNS服务器（10.0.2.6）发送一个DNS查询报文（报文5），但是本地DNS服务器无法解析，所以DNS服务器向IP地址为198.41.0.4的DNS服务器发送一个DNS查询报文（报文6）。

在检测到报文6来自于本地DNS服务器时，攻击者立马伪装一个来自198.41.0.4的DNS相应报文（报文8），将www.example.net恶意地映射到IP地址180.136.102.34上。报文8内容如下图所示：



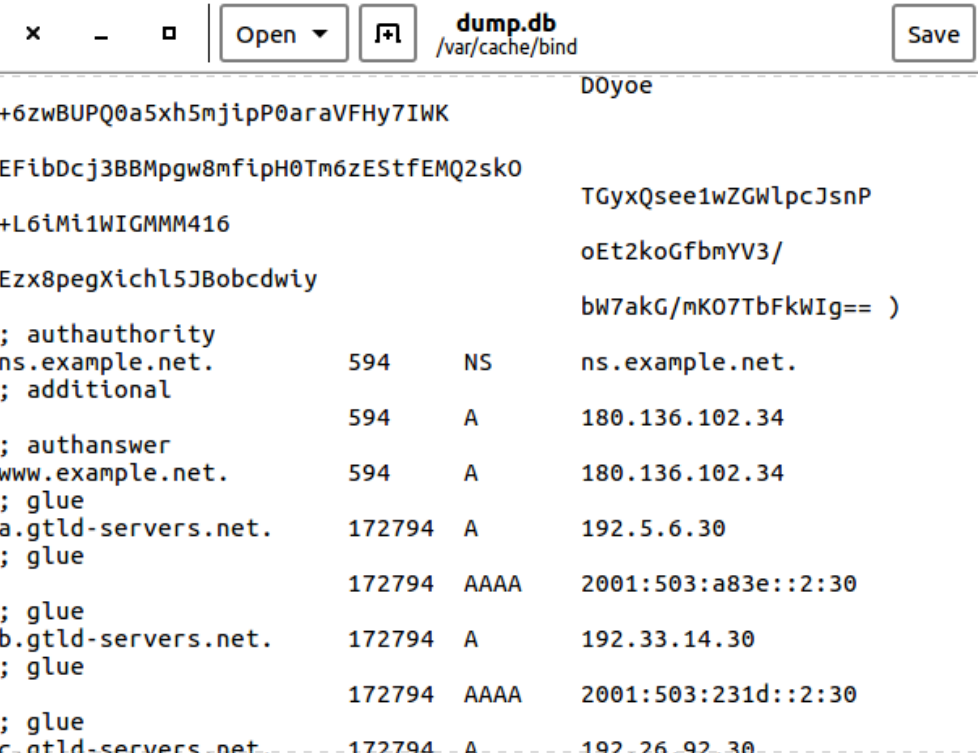
在收到报文8后，本地DNS服务器误以为这就是正确的DNS响应报文，将报文中的解析结果写在报文9中返回给用户主机VM B。报文9内容如下图所示：



至此，整个攻击流程结束，用户主机VM B错误地认为www.example.net的IP地址是180.136.102.34，dig [www.example.net的](http://www.example.netde)结果如下图所示：

**用户主机VM B 对www.example.net的DNS查询结果错误的显示为180.136.102.34（www.weibo.com的真实IP地址）。**

**查看DNS服务器的缓存，看到缓存中www.example.net解析的结果是180.136.102.34。**



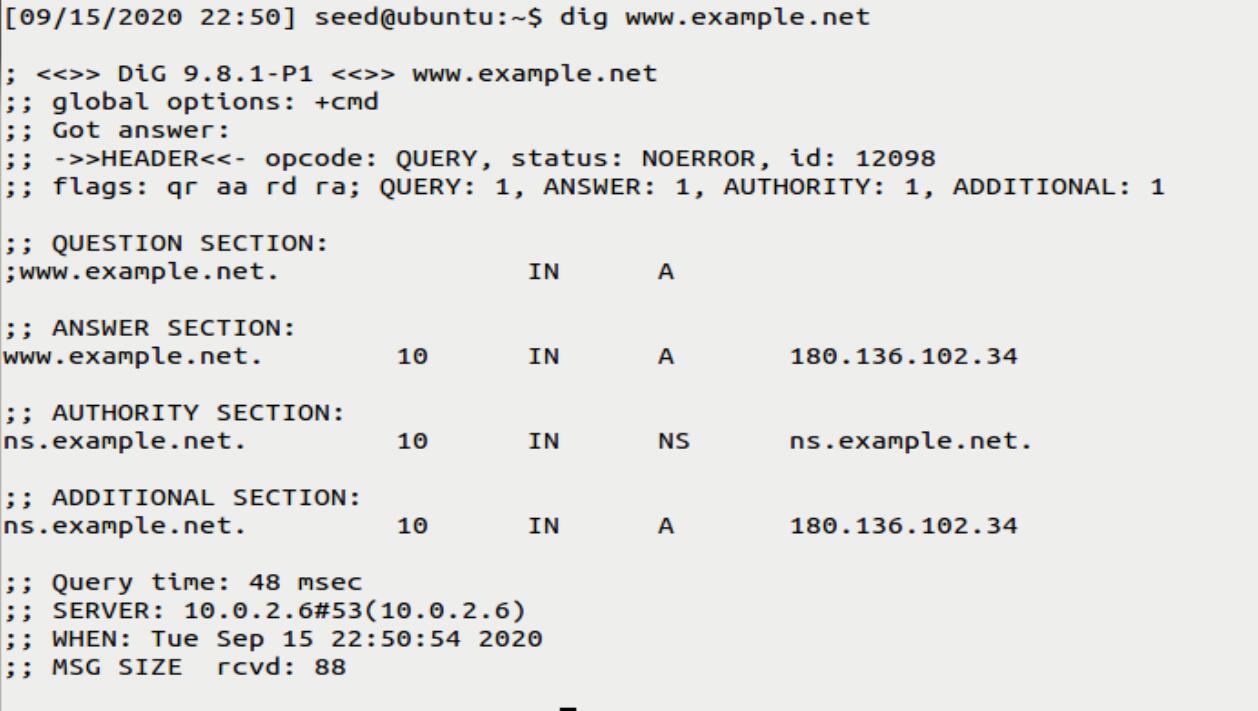
**综上，DNS缓存污染攻击成功。**

# DNS Cache Poisoning: Targeting the Authority Section

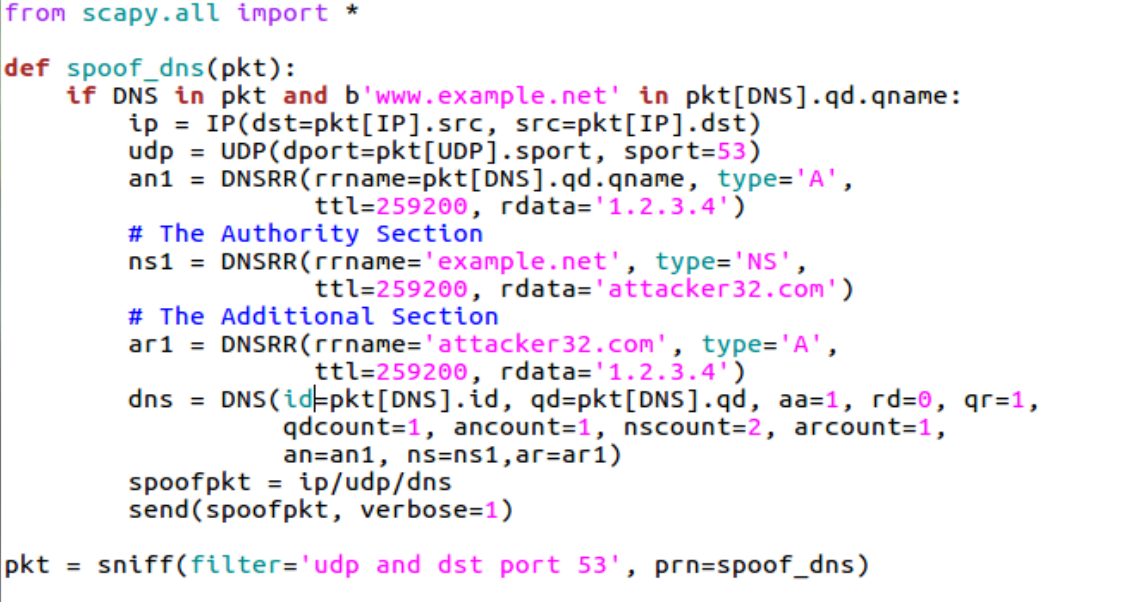
在上一个任务中，我们的DNS缓存中毒攻击仅影响一个主机名，即www.example.net。 如果用户尝试获取另一个主机名的IP地址，例如mail.example.net，我们需要再次发起攻击。

**所以，如果我们发起一种可能影响整个example.net域的攻击，它将更加高效。当查询example.net域内的所有域名时，都需要查询attacker32.com这个域名服务器。由于attacker32.com由攻击者操控，所以它可以返回想要的攻击结果。**

### Step1.运行攻击代码前的结果

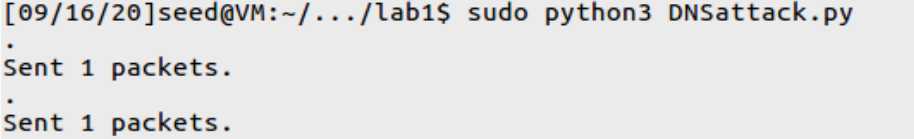
****

### Step2.攻击者VM A利用scapy编写攻击代码DNSattack.py

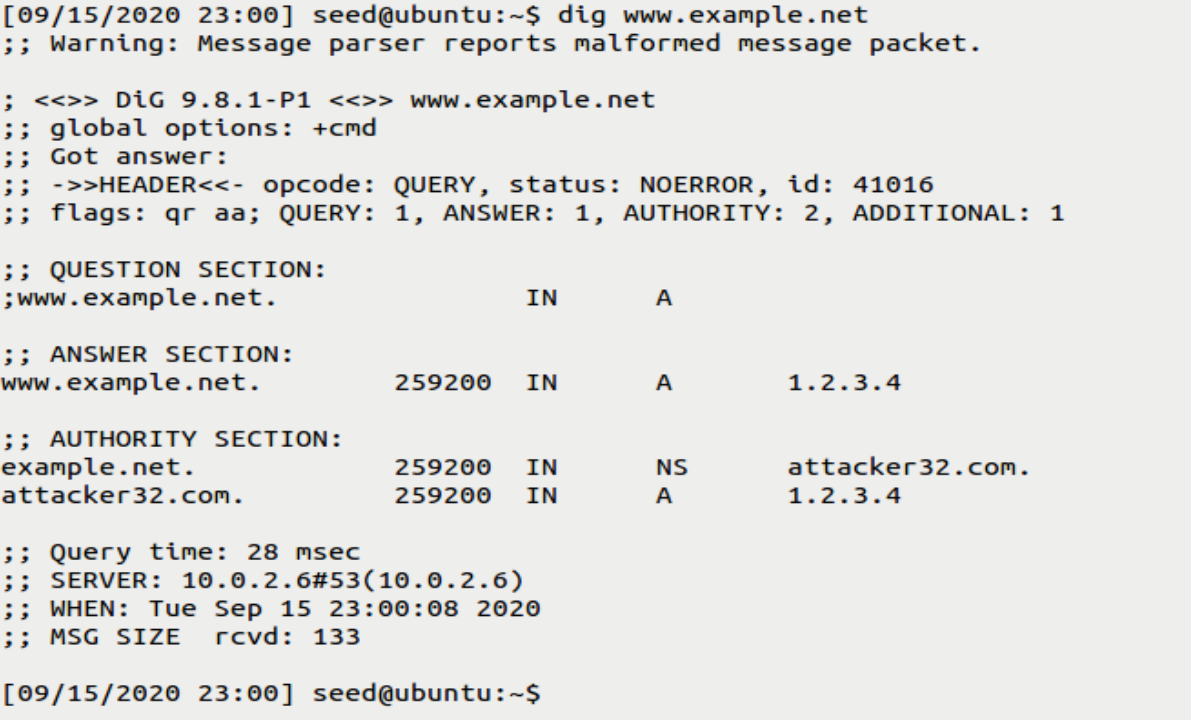


### Step3.运行攻击代码后的结果

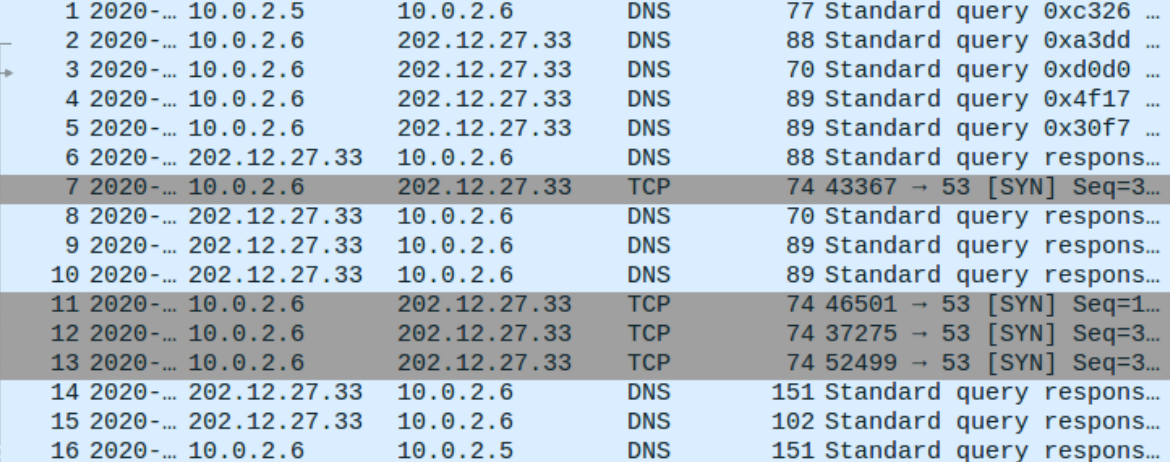
攻击者运行攻击代码



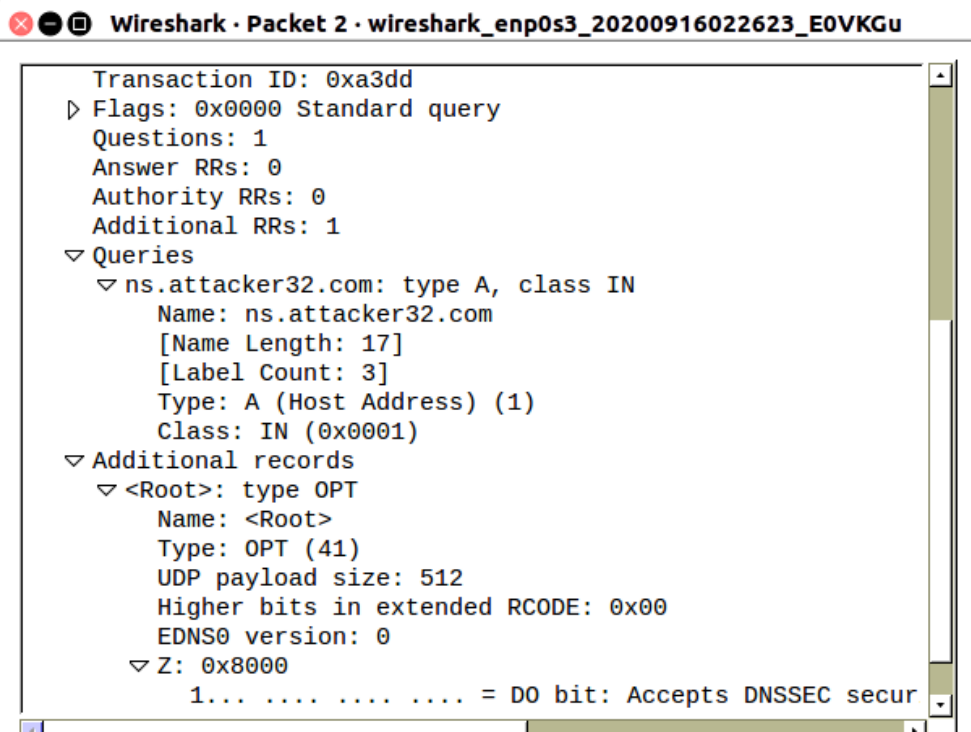
同时，在VM B处查询www.example.net的IP地址，可以看到此时在AUTHORITY SECTION中，对于所有example.net下的域名，都需要查询attacker32.com这个域名服务器。www.example.net最终返回的查询结果为攻击者自定义的IP地址1.2.3.4。



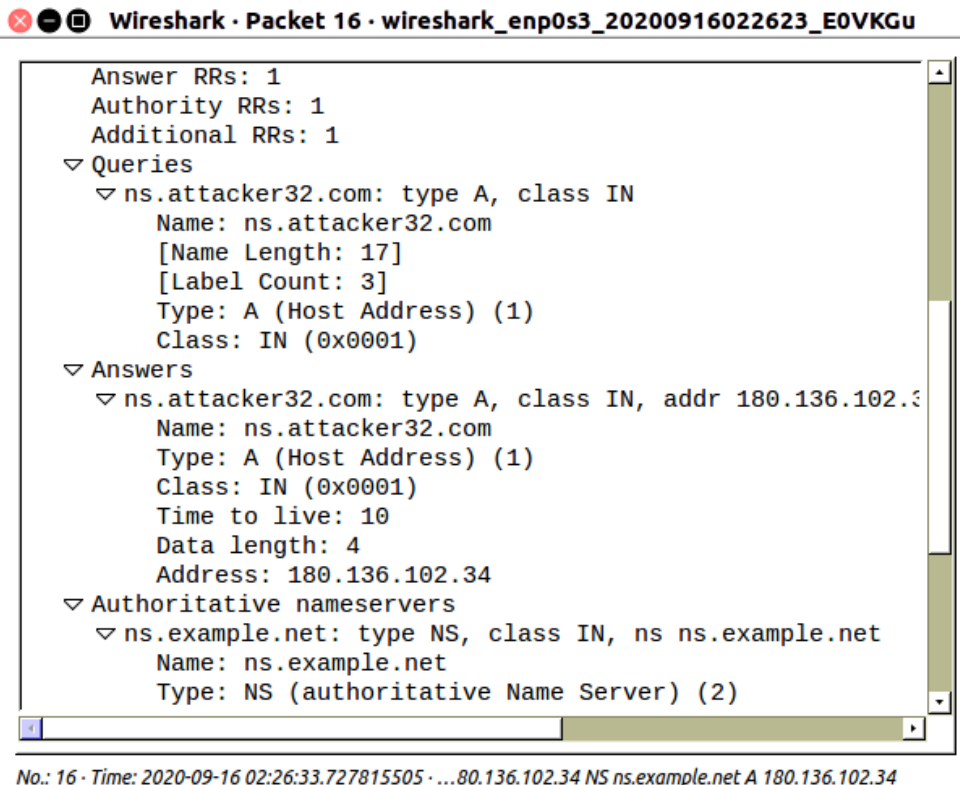
在用户主机VM B上用wireshark抓包，结果如下图所示：



可以看到，用户VM B（10.0.2.5）首先向本地DNS服务器（10.0.2.6）发送一个DNS查询报文（报文1），但是本地DNS服务器无法解析，所以DNS服务器向IP地址为202.12.27.33的DNS服务器发送一个DNS查询报文（报文2），在报文中询问ns.attacker32.com的IP地址。



在检测到报文2来自于本地DNS服务器时，攻击者立马伪装一个DNS响应报文（报文6），本地DNS服务器将响应结果返回给用户主机10.0.2.5（报文16）。报文16的内容如下图所示：



从以上抓包结果可知，对于所有example.net下的域名，都需要查询attacker32.com这个域名服务器。攻击成功。