



中南大學  
CENTRAL SOUTH UNIVERSITY

## 本地 DNS 攻击实验报告

学生姓名	王琪
学 号	0906140120
专业班级	信安 1401
指导教师	王伟平
学 院	信息科学与工程学院
完成时间	2016 年 12 月

# 目录

1 实验室概述.....	2
2 实验室环境.....	2
3 实验室任务.....	5
3.1 任务 1：攻击者已经攻击了受害者的机器.....	5
3.2 任务 2：对用户的直接欺骗响应 .....	7
3.3 任务 3：DNS 服务器缓存中毒 .....	9
4 实验总结.....	11

# 本地 DNS 攻击

## 1 实验室概述

DNS [1]（域名系统）是互联网的电话簿；它将主机名转换为 IP 地址（或 IP 地址到主机名）。这种翻译是通过 DNS 解析，发生在场景后面。DNSPharming [4] 攻击以各种方式操纵这个解析过程，意图误导用户到其他目的地，这通常是恶意的。本实验的目的是了解这种情况攻击工作。

学生将首先设置和配置 DNS 服务器[2]，然后他们将尝试各种 DNS 对同样在实验室环境中的目标的药物攻击。攻击本地受害者与远程 DNS 服务器的困难是完全不同的。因此，我们开发了两个实验室，一个侧重于本地 DNS 攻击，另一个侧重于远程 DNS 攻击。本实验关注本地攻击。

## 2 实验室环境

我们需要设置实验室环境，如图 1 所示。为了简化实验室环境，我们让用户的计算机，DNS 服务器和攻击者的计算机在一台物理机上，但使用不同的虚拟机。本实验中使用的网站可以是任何网站。

我们的配置是基于 Ubuntu，这是我们在预构建的虚拟机中使用的操作系统。我们设置了 DNS 服务器，用户机器和攻击者机器同一个局域网。我们假设用户计算机的 IP 地址是 192.168.0.100，DNS 服务器的 IP 是 192.168.0.10，攻击者的 IP 为 192.168.0.200。

1. 使用虚拟机软件。
2. 使用 Wireshark, Netwag 和 Netwox 工具。
3. 配置 DNS 服务器。

### 2.1 安装并配置 DNS 服务器

**步骤 1：**安装 DNS 服务器。在 192.168.0.10，我们使用安装 BIND9 [3] DNS 服务器。使用以下命令：

```
#sudo apt-get install bind9
```

BIND9 服务器已经安装在我们预先构建的 Ubuntu 虚拟机映像中。

**步骤 2：**创建 named.conf.options 文件。DNS 服务器需要读取 / etc / bind / named.conf 配置文件启动。此配置文件通常包括一个选项文件称为 / etc / bind / named.conf.options。请将以下内容添加到选项文件：

```
options {
dump-file "/var/cache/bind/dump.db" ;
};
```

应该注意，文件/var/cache/bind/dump.db 用于转储 DNS 服务器的缓存。

**步骤 3:** 创建区域。 假设我们拥有一个域：example.com，这意味着我们负责用于提供关于 example.com 的最终答案。 因此，我们需要在中创建一个区域 DNS 服务器通过添加以下内容到/etc/bind/named.conf。 应该注意的是 example.com 域名保留供文档使用，不属于任何人，因此是安全使用它。

```
zone      "example.com"      {          type      master;          file
"/var/cache/bind/example.com.db";  }; zone  "0.168.192.in-addr.arpa"
{ type master; SEED Labs - Local DNS Attack Lab 3 file
"/var/cache/bind/192.168.0"; };
```

注意，我们使用 192.168.0.x 作为示例。 如果使用不同的 IP 地址，则需要更改 /etc/bind/named.conf 和 DNS 查找文件（如下所述）。

**步骤 4:** 设置区域文件。 上述区域中的 file 关键字后面的文件名称为区域文件。 实际的 DNS 解析被放在区域文件中。 在 / var / cache / bind / 目录中，撰写下面的 example.com.db 区域文件（注意下面提到的配置文件可以从本实验室的网页下载；输入这些文件可能会引入错误。 如果你感兴趣在这些配置文件的语法，请参考 RFC 1035 详细）

```
$TTL 3D @ IN SOA ns.example.com. admin.example.com. ( 2008111001 ;serial,
today' s date + today' s serial number 8H ;refresh, seconds 2H ;retry,
seconds 4W ;expire, seconds 1D) ;minimum, seconds @ IN NS
ns.example.com. ;Address of name server @ IN MX 10
mail.example.com. ;Primary Mail Exchanger www IN A 192.168.0.101 ;Address
of www.example.com mail IN A 192.168.0.102 ;Address of mail.example.com
ns IN A 192.168.0.10 ;Address of ns.example.com *.example.com. IN A
192.168.0.100 ;Address for other URL in ;example.com. Domain
```

符号 “@” 是一个特殊符号，表示来自 named.conf 的源。 因此，’@’ 在这里代表 example.com。 “IN” 是指互联网。 “SOA” 是开始权限的缩写。 此区域文件包含 7 个资源记录（RR）：SOA（开始权限）RR，NS（名称服务器）RR，MX（邮件 eXchanger）RR 和 4 A（主机地址）RR。 我们还需要设置 DNS 反向查找文件。 在目录 / var / cache / bind / 中，编写 a 反向 DNS 查找文件名为 192.168.0 的 example.com 域：

```
$TTL 3D @ IN SOA ns.example.com. admin.example.com. ( 2008111001 8H 2H
4W 1D) @ IN NS ns.example.com. 101 IN PTR www.example.com. 102 IN PTR
mail.example.com. 10 IN PTR ns.example.com.
```

**步骤 5:** 启动 DNS 服务器。现在我们准备好启动 DNS 服务器。运行以下命令：  
%sudo /etc/init.d/bind9 restart 要么 %sudo 服务 bind9 重启。

## 2.2 配置用户机器

在用户计算机 192.168.0.100 上，我们需要让机器 192.168.0.10 成为默认 DNS 服务器。我们通过更改用户计算机的 DNS 设置文件 /etc/resolv.conf 实现这一点：nameserver 192.168.0.10 # 刚刚设置的 DNS 服务器的 IP 注意：确保这是 /etc/resolv.conf 中唯一的名称服务器条目。还要注意，在 Ubuntu，/etc/resolv.conf 可能被 DHCP 客户端覆盖。为了避免这种情况，请禁用 DHCP 做以下（在 Ubuntu 12.04）：单击“系统设置” -> “网络”，单击“有线”选项卡中的“选项”选择“IPv4 设置” -> “方法” -> “自动（DHCP）地址”并仅更新具有 BIND DNS 服务器的 IP 地址的“DNS 服务器”条目。现在单击右上角的“网络图标”，然后选择“Auto eth0”。这将刷新有线网络连接和更新更改。

您应该重新启动您的 Ubuntu 计算机以使修改的设置生效。

## 2.3 配置攻击机

在攻击者机器上，没有太多配置。

## 2.4 预期产出

在你按照 Th 设置实验室环境后, 运行下列代码:

```
% dig www.example.com ns.example.com.
```

注意：ANSWER SECTION 包含 DNS 映射。您可以注意到的 IP 地址

www.example.com 现在是 192.169.0.101，这是我们在 DNS 服务器中设置的。对于一个简单明了的答案，我们可以使用 nslookup 代替。要做 DNS 反向查找，请发出 dig -xN.N.N.N.

## 2.5 安装 Wireshark

Wireshark 是这个实验室非常重要的工具;你可以嗅闻每个正在经历的包 LAN。你可以从 <http://www.wireshark.org> 获取 Wireshark。虽然 Netwox 也来了与嗅探器, Wireshark 是一个更好的嗅探器。Wireshark 已经安装在我们的预建虚拟机。

## 3 实验室任务

对用户的 Pharming 攻击的主要目的是当用户重定向用户到另一台机器 B. 尝试使用 A 的主机名访问机器 A. 例如, 当用户尝试访问在线银行时, 如 `www.chase.com`, 如果对手可以将用户重定向到一个看起来很恶意的网站非常像 `www.chase.com` 的主要网站, 用户可能会被愚弄, 并给出密码他/她的网上银行帐户。当用户在他的浏览器中键入 `www.chase.com` 时, 用户的机器将发出 DNS 查询找出这个网站的 IP 地址。攻击者的目标是用伪造的 DNS 欺骗用户的机器回复, 它将 `www.chase.com` 解析为一个恶意 IP 地址。有几种方法来实现这样攻击在实验室说明的其余部分, 我们将使用 `www.example.com` 作为用户的网站想要访问, 而不是使用真实的网站名称 `www.chase.com`; `example.com` 域名称保留供文档使用, 不属于任何人。

### 3.1 任务 1: 攻击者已经攻击了受害者的机器

修改 HOSTS 文件。使用 HOSTS 文件 (`/ etc / hosts`) 中的主机名和 IP 地址对用于本地查找;它们优先于远程 DNS 查找。例如, 如果有以下内容在用户计算机的 HOSTS 文件中输入, `www.example.com` 将解析为 `1.2.3.4` 用户的计算机而不要求任何 DNS 服务器: `1.2.3.4 www.example.com` 攻击。如果攻击者破坏了用户的计算机, 他们可以修改 HOSTS 文件以重定向每当用户尝试访问 `www.example.com` 时, 用户都会转到恶意网站。假设你已经侵入了一台机器, 尝试这种技术将 `www.example.com` 重定向到任何 IP 地址。

注意: `/ etc / hosts` 由 `nslookup` 命令忽略, 但将在 `ping` 命令生效和网络浏览器等。

将 `example` 加入 `hosts` 文件:

```

127.0.0.1      www.SQLLabCollabtive.com
127.0.0.1      www.XSSLabCollabtive.com
127.0.0.1      www.SOPLab.com
127.0.0.1      www.SOPLabAttacker.com
127.0.0.1      www.SOPLabCollabtive.com

127.0.0.1      www.OriginalphpMyAdmin.com

127.0.0.1      www.CSRFLabElgg.com
127.0.0.1      www.XSSLabElgg.com
127.0.0.1      www.SeedLabElgg.com
10.0.2.5       www.heartbleedlabelgg.com
127.0.0.1      www.WTLabElgg.com

127.0.0.1      www.wtmobilestore.com
127.0.0.1      www.wtshoestore.com
127.0.0.1      www.wtelectronicstore.com
127.0.0.1      www.wtcamerastore.com

127.0.0.1      www.wtlabadserver.com
1.2.3.4        www.example.com

```

修改 exampleIP:



```

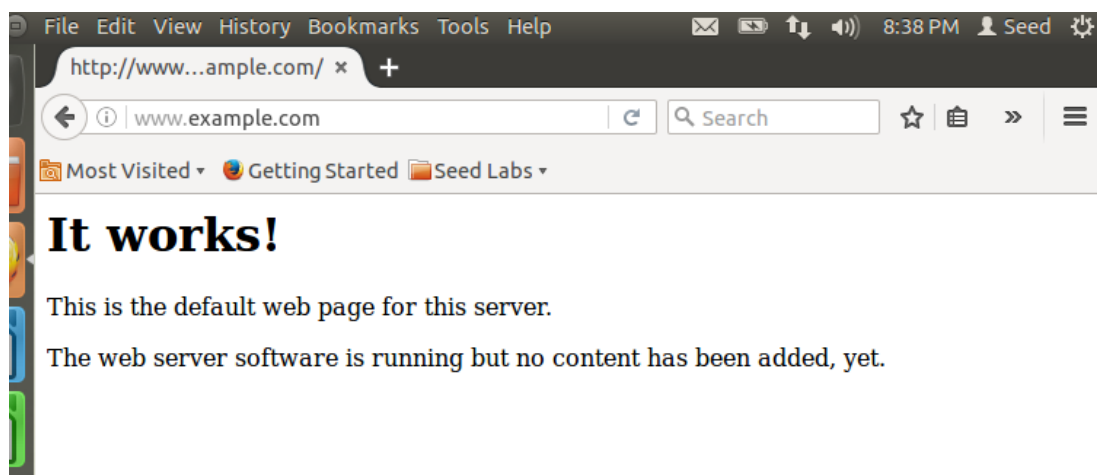
127.0.0.1      www.CSRFLabElgg.com
127.0.0.1      www.XSSLabElgg.com
127.0.0.1      www.SeedLabElgg.com
10.0.2.5       www.heartbleedlabelgg.com
127.0.0.1      www.WTLabElgg.com

127.0.0.1      www.wtmobilestore.com
127.0.0.1      www.wtshoestore.com
127.0.0.1      www.wtelectronicstore.com
127.0.0.1      www.wtcamerastore.com

127.0.0.1      www.wtlabadserver.com
127.0.0.1      www.example.com
# The following lines are desirable for IPv6 capable hosts

```

修改 example 后访问的网站:



### 3.2 任务 2：对用户的直接欺骗响应

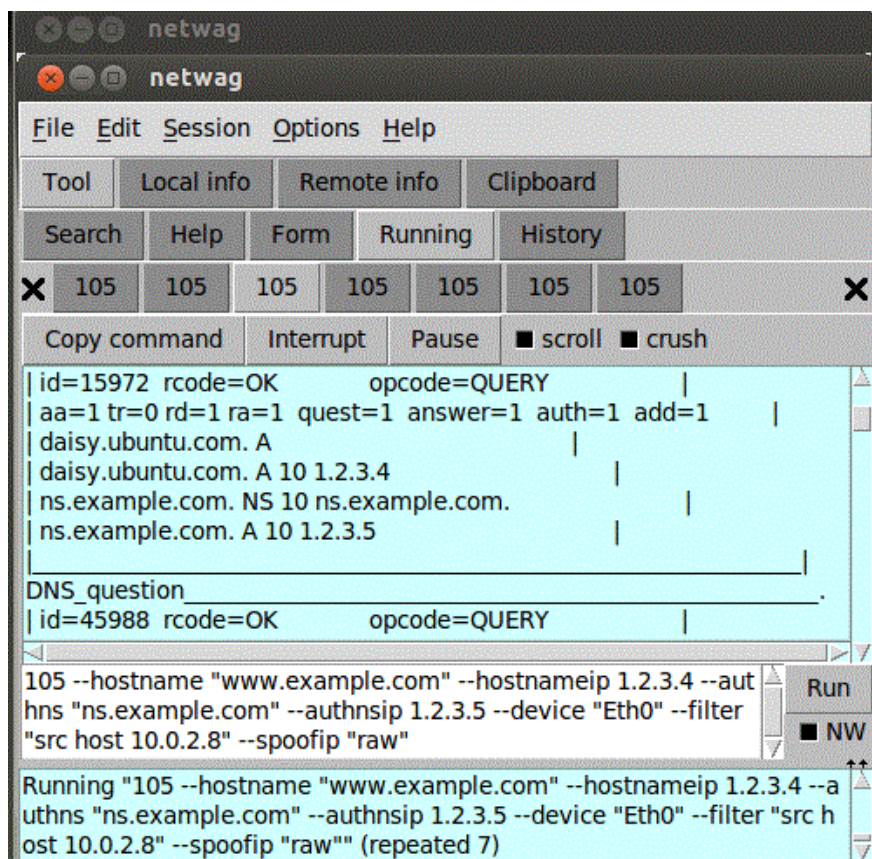
在这次攻击中，受害者的机器没有受到攻击，所以攻击者不能直接更改 DNS 查询进程在受害者的机器上。但是，如果攻击者处于同一个局域网上受害者，他们仍然可以实现巨大的伤害。如图 2 所示。当用户在 web 浏览器中键入网站的名称（主机名，例如 `www.example.com`）时，用户的计算机将向 DNS 服务器发出 DNS 请求以解析主机名的 IP 地址。在听到这个 DNS 请求后，攻击者可以欺骗假的 DNS 响应[6]。假 DNS 答复将被用户的计算机接受，如果它符合以下标准：

1. 源 IP 地址必须与 DNS 服务器的 IP 地址匹配。
2. 目标 IP 地址必须与用户机器的 IP 地址匹配。
3. 源端口号（UDP 端口）必须与 DNS 请求发送到的端口号匹配（通常为端口 53）。
4. 目标端口号必须与发送 DNS 请求的端口号相匹配。
5. 必须正确计算 UDP 校验和。
6. 事务 ID 必须与 DNS 请求中的事务 ID 匹配。
7. 答复问题部分中的域名必须与问题中的域名匹配部分。
8. 答案部分中的域名必须与问题部分中的域名匹配 DNS 请求。
9. 用户的计算机在接收合法 DNS 之前必须接收攻击者的 DNS 回复响应。

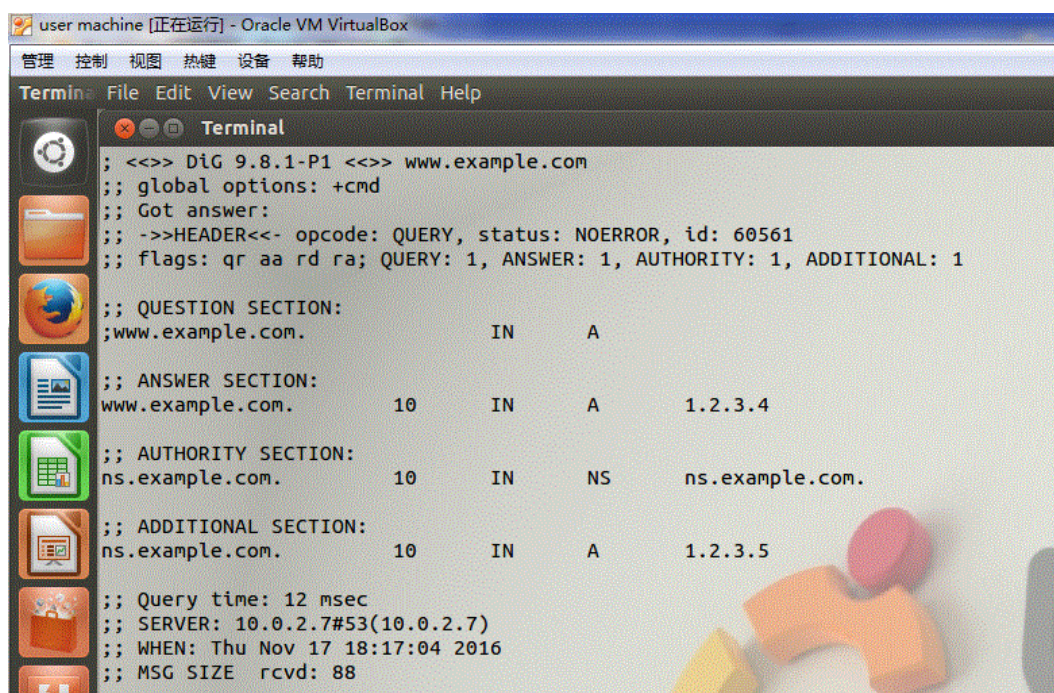
为了满足标准 1 到 8，攻击者可以窥探受害者发送的 DNS 请求消息；他们可以然后创建一个假的 DNS 响应，并发送回受害者，在真正的 DNS 服务器之前。Netwox 工具 105 提供进行这种嗅探和响应的实用程序。提示：在 Netwox / Netwag 工具



105 中，您可以使用“过滤器”字段指示您的 IP 地址目标。  
 例如，在下面显示的场景中，您可以使用“src host 192.168.0.100”。  
 attackers 构造包：



user 收到 attack 构造的包





收到 attackers 的应答包

```
; <<>> DiG 9.8.1-P1 <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60561
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www.example.com.                IN      A

;; ANSWER SECTION:
www.example.com.                10      IN      A      1.2.3.4

;; AUTHORITY SECTION:
ns.example.com.                 10      IN      NS      ns.example.com.

;; ADDITIONAL SECTION:
ns.example.com.                 10      IN      A      1.2.3.5

;; Query time: 12 msec
;; SERVER: 10.0.2.7#53(10.0.2.7)
;; WHEN: Thu Nov 17 18:17:04 2016
;; MSG SIZE rcvd: 88
```

### 3.3 任务 3: DNS 服务器缓存中毒

上述攻击针对的是用户的机器。为了达到持久的效果，每次用户的机器发出一个 DNS 查询 `www.example.com`，攻击者的机器必须发出一个欺 DNS 响应。这可能不是那么高效；有一个更好的方式来进行攻击的目标 DNS 服务器，而不是用的机器当一个 DNS 服务器 Apollo 收到一个查询时，如果主机名不在 Apollo 的域内，它将会请求其他 DNS 服务器获取主机名解析。

请注意，在我们的实验室设置中，我们的 DNS 域服务器是 `example.com`；因此，对于其他域（例如 `www.google.com`）的 DNS 查询，DNS 服务器 Apollo 将询问他 DNS 服务器。然而，在阿波罗询问其他 DNS 服务器之前，它首先从自己的缓存中寻找答案；如果答案是肯定的，DNS 服务器阿波罗会简单地回复与来自其缓存的信息。如果答案不在缓存中，DNS 服务器将尝试获取答案从其他 DNS 服务器。Apollo 得到答案时，它会将答案存储在缓存中，所以接下来时间，没有必要其他 DNS 服务器。因此，如果攻击者可以欺骗来自其他 DNS 服务器的响应，Apollo 将保留欺骗响应在其缓存[5]中一段时间。下一次，当用户的机器想要解析时相同的主机名，Apollo 将使用在缓存中的欺骗响应来回复。这样，攻击者只需要一次欺骗，并且影响将持续直到缓存的信息过期。这种攻击称为 DNS 缓存中毒。下图（图 3）说明了这种攻击。我们可以使用相同的工具（Netwox 105）进行这次攻击。在攻击之前，请确保 DNS 服务器的缓存为空。

您可以使用以下命令刷新缓存：

```
#sudo rndc flush
```

这种攻击和以前的攻击之间的区别是，我们欺骗了对 DNS 的响应服务器，因此我们将过滤器字段设置为“src host 192.168.0.10”，这是 DNS 的 IP 地址服务器。我们还使用 ttl 字段（生存时间）来指示我们希望假的答案在 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 如果我们设置 ttl600（秒），然后 DNS 服务器将继续给出假的答案在接下来的 10 分钟。

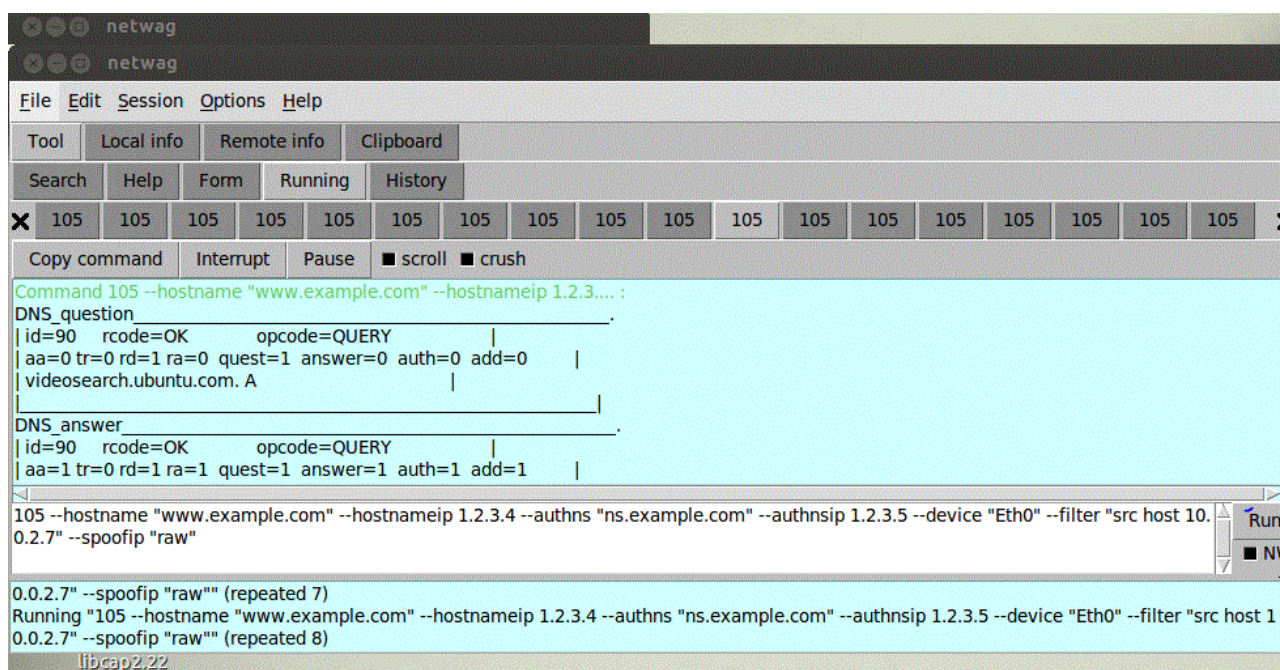
注意：请在 spoofip 字段中选择 raw；否则，Netwox 105 会尝试也欺骗 IP 地址的 MAC 地址。为了获得 MAC 地址，工具发出一个 ARP 请求，询问用于骗 IP 的 MAC 地址。这个欺骗的 IP 地址通常是根 DNS 服务器（这通常是 DNS 服务询问其是否无法解析名称的第一个位置），显然根 DNS 服务器是不在同一个 LAN 上因此，没有人会回复 ARP 请求。该工具将等待 ARP 答复一段时间之前，没有 MAC 地址。等待将延迟工具发出欺骗性响应。如果实际 DNS 响应比欺骗性响应早，将失败。这就是为什么你需要问工具不去欺骗 MAC 地址。

您可以通过使用捕获的网络流量来判断 DNS 服务器是否中毒 Wireshark 或通过转储 DNS 服务器的缓存。要转储和查看 DNS 服务器的缓存，请发出以下命令：

```
#sudo rndc dumpdb -cache
```

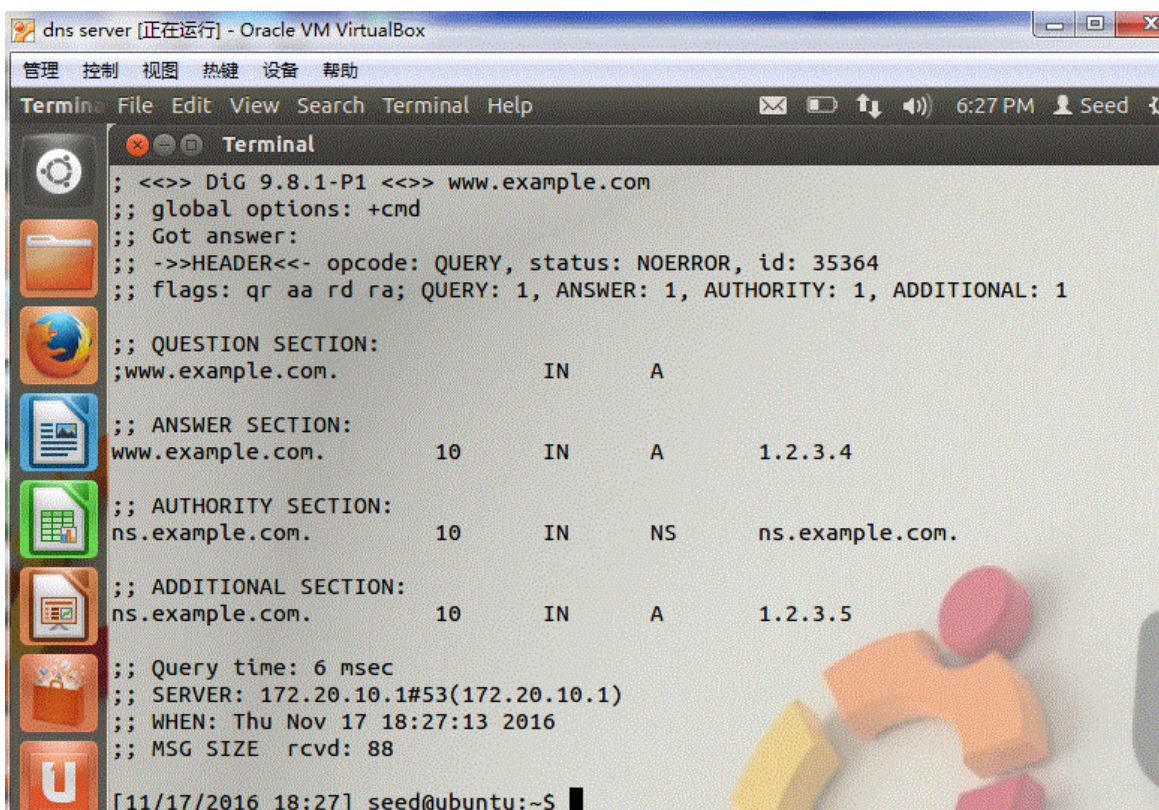
```
#sudo cat /var/cache/bind/dump.db
```

attackers 给 DNS 构造包





dns 收到 attacker 构造的包



```
dns server [正在运行] - Oracle VM VirtualBox
管理 控制 视图 热键 设备 帮助
Terminal File Edit View Search Terminal Help
; <<>> DiG 9.8.1-P1 <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 35364
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1
;; QUESTION SECTION:
;www.example.com.                IN      A
;; ANSWER SECTION:
www.example.com.                10      IN      A      1.2.3.4
;; AUTHORITY SECTION:
ns.example.com.                 10      IN      NS      ns.example.com.
;; ADDITIONAL SECTION:
ns.example.com.                 10      IN      A      1.2.3.5
;; Query time: 6 msec
;; SERVER: 172.20.10.1#53(172.20.10.1)
;; WHEN: Thu Nov 17 18:27:13 2016
;; MSG SIZE rcvd: 88
[11/17/2016 18:27] seed@ubuntu:~$
```

## 4 实验总结

DNS 是一个用于管理主机名字和地址信息映射的分布式数据库系统，它将便于记忆和理解的名称同枯燥的 IP 地址联系起来，大大方便了人们的使用。DNS 是大部分网络应用的基础，但是由于协议本身的设计缺陷[1]，没有提供适当的信息保护和认证机制，使得 DNS 很容易受到攻击。

DNS 欺骗攻击可能存在于客户端和 DNS 服务器间，也可能存在于各 DNS 服务器之间，但其工作原理是一致的，步骤如下：

- (1) DNS 客户端向首选 DNS 服务器发送对于 www.hit.edu.cn 的递归解析请求。
- (2) 攻击者监听到请求，并根据请求 ID 向请求者发送虚假应答包，通知与 www.hit.edu.cn 对应的 IP 地址为 1.2.3.4。
- (3) 本地 DNS 服务器返回正确应答，但由于在时间上晚于监听者的应答，结果被丢弃。
- (4) 攻击完成，客户端对 www.hit.edu.cn 的访问被重定向到 1.2.3.4

此次实验也使我更加明白了网络安全的重要性，作为一名信息安全专业的学生，更应该了解更多的安全问题及相关防御知识，平时应该多积累。日后的学习还有很长的路要走，也感谢在这期间老师和同学给予我的帮助。

