网络安全课外实验

实验报告

学　　院：信息科学与工程学院

专业班级：信息安全 1401 班

指导老师：王伟平

学　　号：0906140130

姓　　名：殷淑杰

目 录

DNS 本地攻击实验

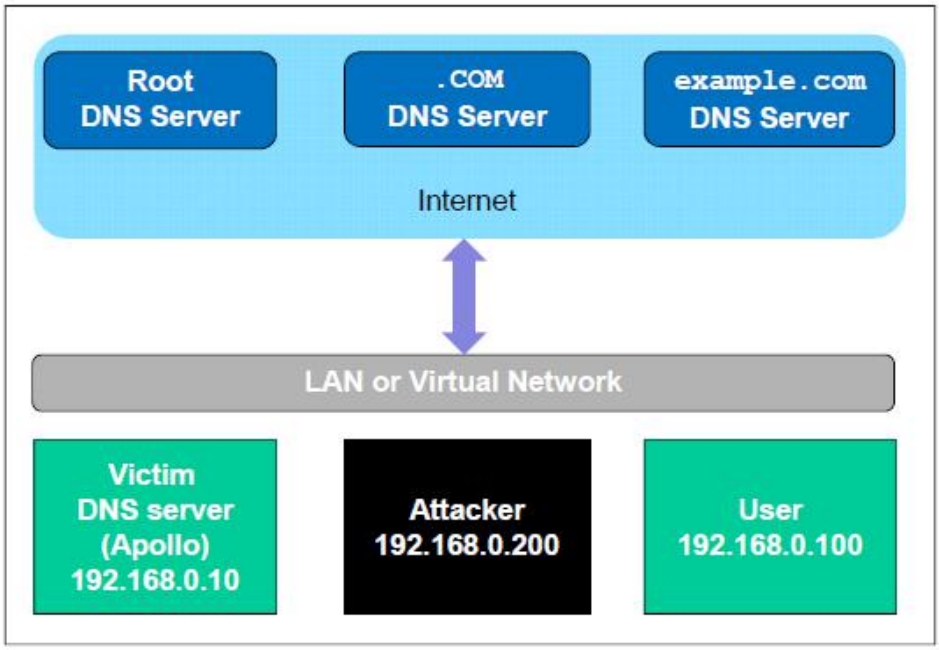
1. 概要介绍

DNS（域名系统）相当于互联网的通讯录，它是主机名与 IP 地址的相互对应。这个转换过程是通过 DNS 解析完成的。DNS 服务器嫁接攻击以不同的方式操纵着这一转换过程，企图误导用户映射到一个恶意目的地。实验目的是了解 DNS 攻击是如何进行的。要求首先创建并配置一个 DNS 服务器，然后尝试用不同的 DNS 嫁接攻击同一实验环境下的目标机。

2. 实验环境

我们需要建立如图所示的实验环境。为了简化实验，将用户机，DNS 服务器以及攻击机器设置在同一物理机上，使用不同的虚拟机完成。至于网站可以是实验室所提供的任意网站。从下图可以看出，我们将 DNS 服务器，用户机和攻击机器设置在同一局域网。假设用户机 IP 是 192.168.0.100，DNS 服务器是 192.168.0.10，攻击机器是 192.168.0.200。

（根据自己的虚拟机配置为，用户机 IP：192.168.11.128，DNS 服务器：192.168.11.130，攻击机器：192.168.1.131）



2.1 安装和配置 DNS 服务器

第一步：安装 DNS 服务器

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

第二步：创建 named.conf.options 文件，请在/etc/bind/named.conf.options. 下添加：

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

dump.db 是用来保存 DNS 服务器缓存数据的文件。

第三步：创建域。假设我们拥有一个域名：www.example.com，我们需要对其负责并提供相关的应答内容。所以，需要在 DNS 服务器上创建一个域。通过修改/etc/bind/named.conf：

```
zone "example.com" {
    type master;
    File "/var/cache/bind/example.com.db";
};
zone "0.168.192.in-addr.arpa" {
    type master;
    file "/var/cache/bind/192.168.0";
};
```

第四步：创建域文件。上述域中的文件关键字后的文件名称为域文件。实际的 DNS 解析放在域文件中。在/var/cache/bind/ directory 创建 example.com.db 文件，请注意配置语法：

```
$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
```

在/var/cache/bind/中创建反向域名解析：

```
$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.
```

第五步：启动 DNS 服务器：

% sudo /etc/init.d/bind9 restart

2.2 配置用户机

设置文件/etc/resolv.conf：

nameserver 192.168.11.130 # the ip of the DNS server you just setup

注意：确保这是此文件中唯一的 nameserver，它有可能被用户 DHCP 覆盖，为了避免这个问题，需要进行如下操作：

```
Click "System Settings" -> "Network",
Click "Options" in "Wired" Tab,
Select "IPv4 Settings" -> "Method" ->"Automatic(DHCP) Addresses Only"
and update only "DNS Servers" entry with IP address of BIND DNS Server.

Now Click the "Network Icon" on the top right corner and Select
"Auto eth0". This will refresh the wired network connection and
updates the changes.
```

2.3 配置攻击机

此实验不需对攻击机做过多配置。

2.4 预期输出

配置好环境，DNS 和用户机后，在用户机终端执行如下命令：

% dig www.example.com

```
<<>> DiG 9.5.0b2 <<>> www.example.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27136
;; 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. 259200 IN A 192.168.0.101

;; AUTHORITY SECTION:
example.com. 259200 IN NS ns.example.com.

;; ADDITIONAL SECTION:
ns.example.com. 259200 IN A 192.168.0.10

;; Query time: 80 msec
;; SERVER: 192.168.0.10#53(192.168.0.10)
;; WHEN: Tue Nov 11 15:26:32 2008
;; MSG SIZE  rcvd: 82
```

3. 实验内容

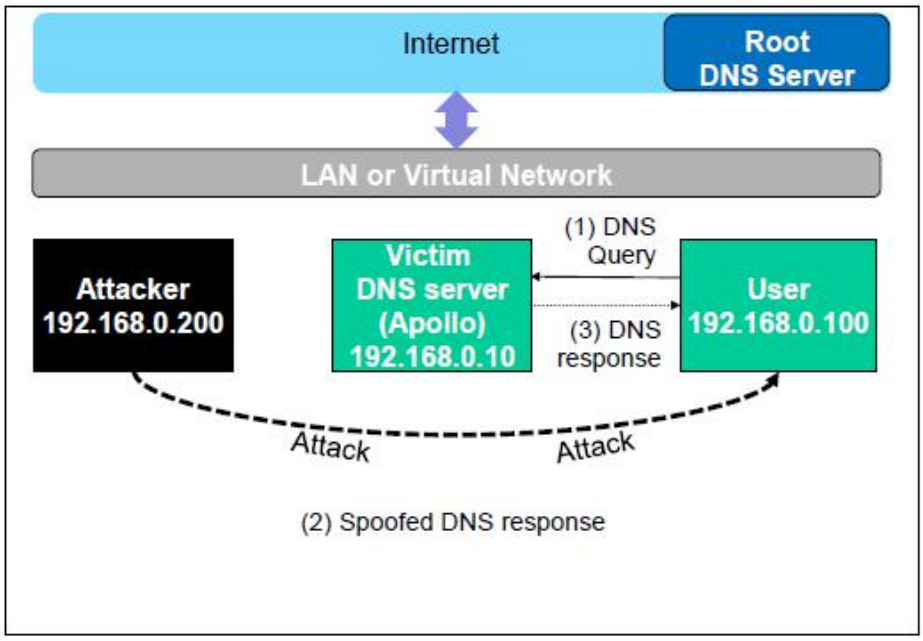
3.1 攻击者已侵入用户机

修改用户的 hosts 文件：

1.2.3.4　www.example.com

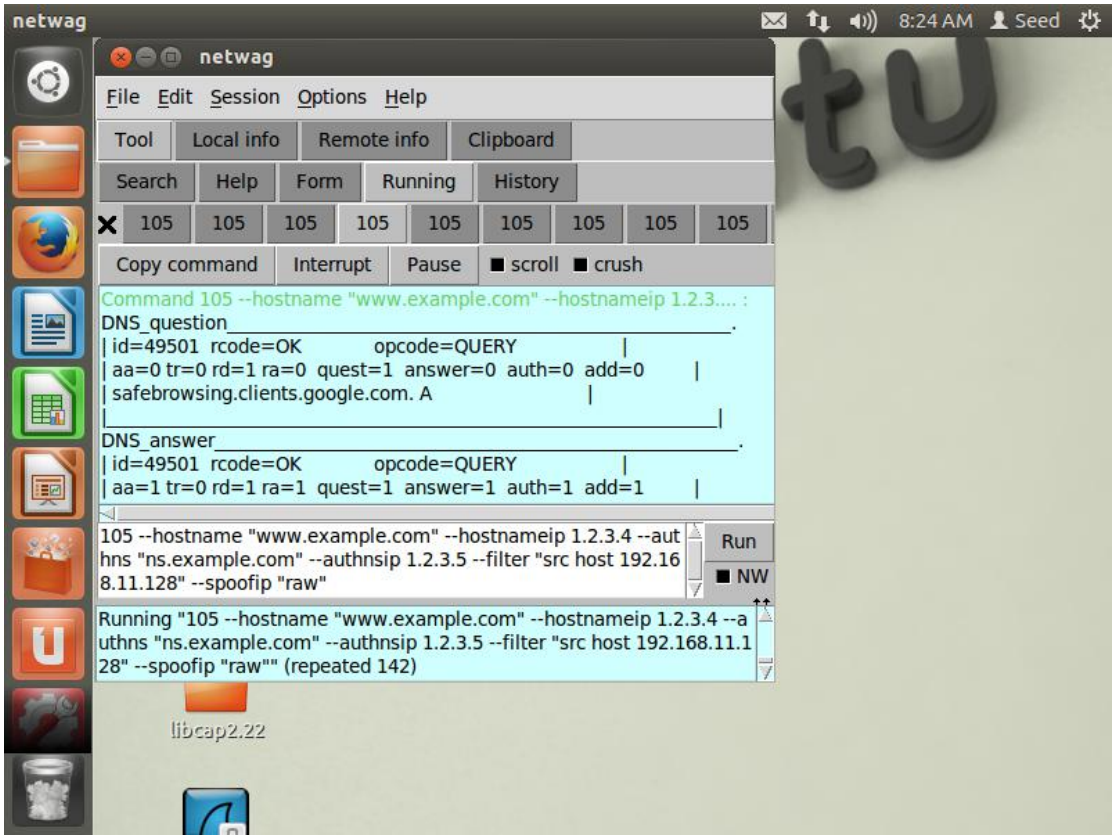
3.2 直接欺骗用户

在此攻击中，受害服务器并没有被破坏，所以攻击者无法直接更改 DNS 服务器上的查询处理机制。然而，如果攻击者和受害服务器在同一局域网，也可以造成一定的影响（如下图所示）。

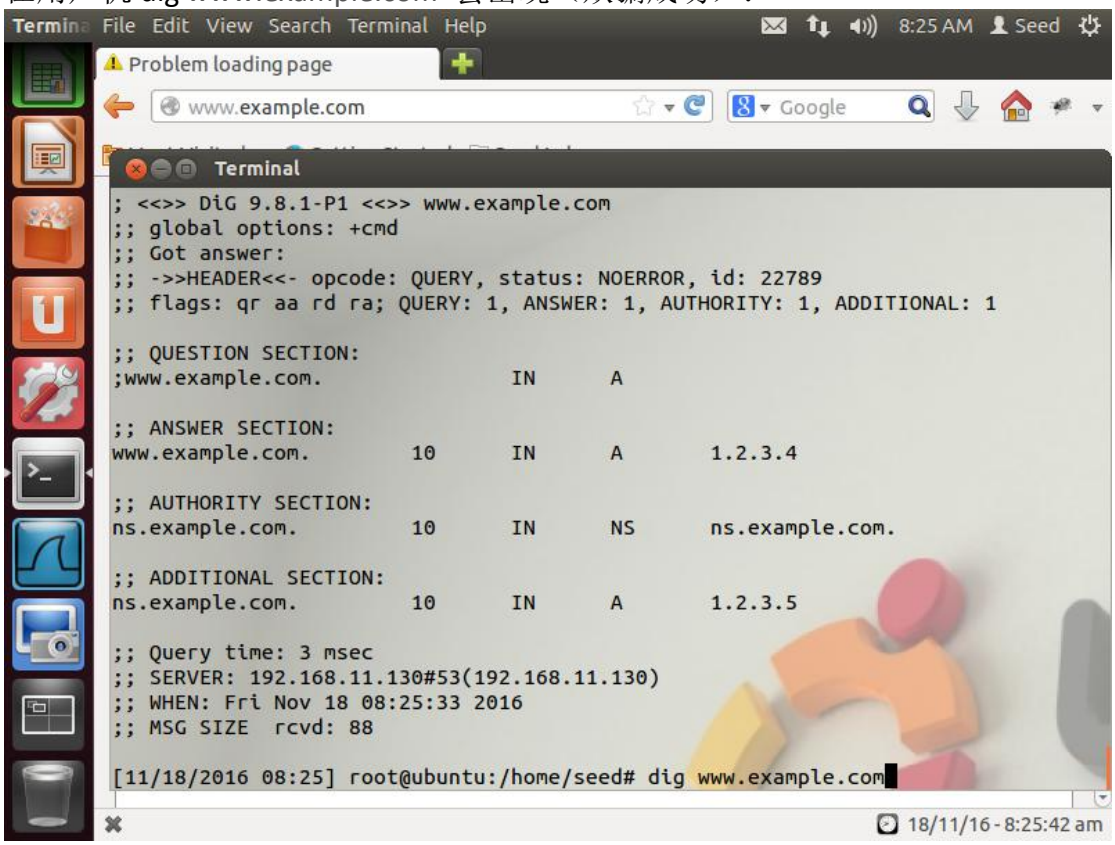


当用户在 Web 浏览器输入一个网站的名称（主机名，如 www.example.com），用户的计算机会发出 DNS 请求到 DNS 服务器来解析主机名的 IP 地址。攻击者可以伪造这个 DNS 响应，如果满足以下条件，这个伪造响应就会被用户机接收：

1. 源 IP 地址必须匹配的 DNS 服务器的 IP 地址。
2. 目标 IP 地址必须与用户的机器的 IP 地址相匹配。
3. 源端口号（UDP 端口）必须匹配的 DNS 请求被发送到的端口号(通常是 53 号端口）。
4. 目的端口号必须匹配的 DNS 请求发送的端口号。
5. UDP 校验和必须正确计算。
6. 事务 ID 必须匹配 DNS 请求的事务 ID。
7. 响应域名必须与请求中的域名相匹配。
8. 在响应部分中的域名必须与 DNS 请求中的域名相匹配。
9. 用户的计算机必须收到合法的 DNS 响应之前接受攻击者的 DNS 回复。



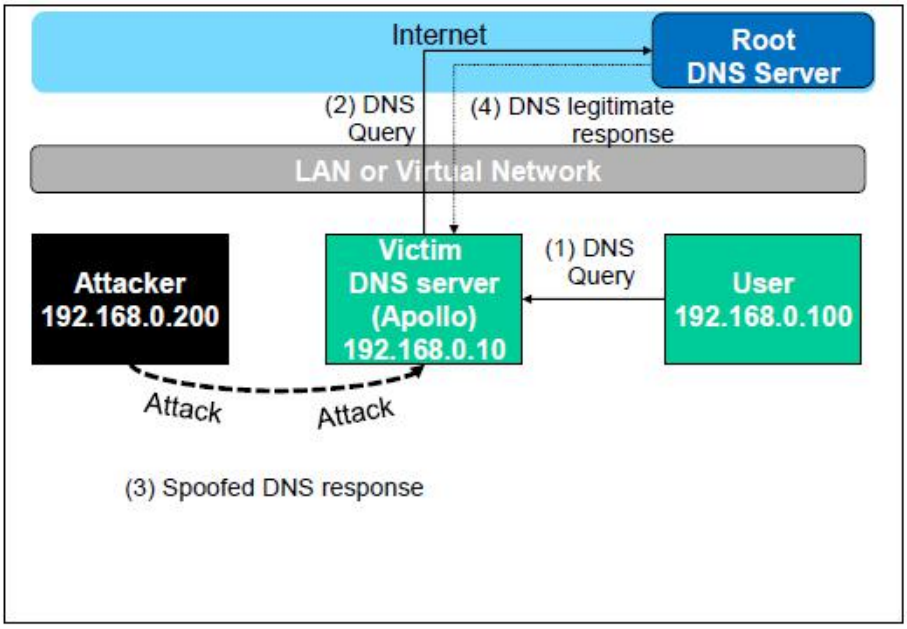
在用户机 dig www.example.com 会出现（欺骗成功）：

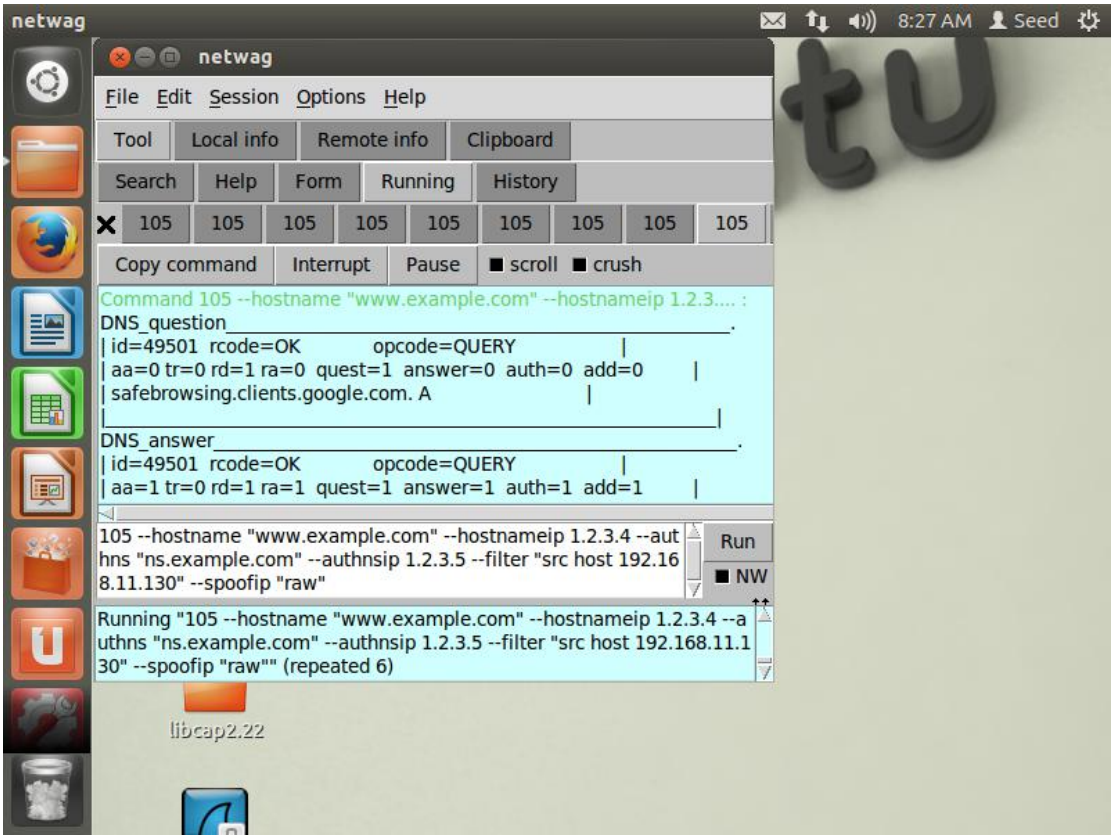


3.3 DNS 服务器缓存攻击

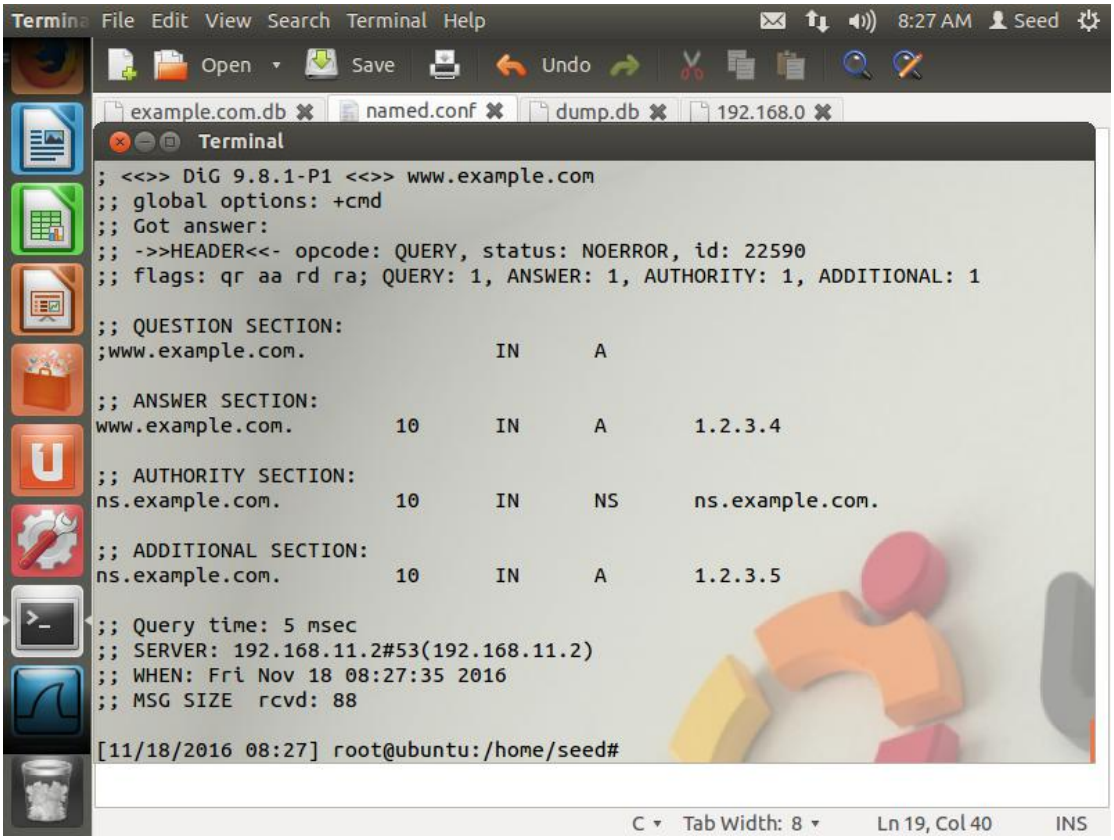
攻击用户机的方式，用户每发一次 DNS 请求，攻击者就要伪造一个响应，这样效果并不持久。为了提高效率，就是直接攻击 DNS 服务器而不是用户机。

当一个 DNS 服务器 Apollo 接收到查询请求，如果主机名不在 DNS 服务器域中，那么它会向其他 DNS 获取主机名解析。然而在 Apollo 向其他 DNS 问询前，会现在自己的缓存中查找，如果存在，就直接回应，如果不存在，其将会向其他 DNS 服务器询解。Apollo 获取之后便会存在自己的缓存中，以便下一次使用。所以，如果攻击者如果从其他 DNS 服务器发送响应给 Apollo，那么它将一直持有伪造的响应，从而造成了 DNS 攻击。（如下图所示）：





在一段时间内 dig www.example.com 都会出现以下情况：



4. 参考文献

[1] RFC 1035 Domain Names - Implementation and Specification : http://www.rfc-base.org/rfc-1035.html
[2] DNS HOWTO : http://www.tldp.org/HOWTO/DNS-HOWTO.html
[3] BIND 9 Administrator ReferenceManual : http://www.bind9.net/manual/bind/9.3.2/Bv9ARM.ch01.html
[4] Pharming Guide : http://www.technicalinfo.net/papers/Pharming.html
[5] DNS Cache Poisoning: http://www.secureworks.com/resources/articles/other articles/dns-cachepoisoning/
[6] DNS Client Spoof: http://evan.stasis.org/odds/dns-client spoofing.txt

心得体会

很开心自己真的去做个 seed project 上的实验，这真的是一个很不错的网站，自带的虚拟机配置好环境，同时附带了配置方法，可以供高阶人员使用。这次实验，我最初的目标是做一个简单级别的和一个中等级别的。但最后却成功完成了两个简单级别的实验。

第二个实验我选择了 DNS 本地攻击，起初不想选择它的，因为自己的机器带三个虚拟机很吃力。最开始是打算和同学合作的，首先要让我们的物理机器同处一个局域网下，互相 ping 到彼此，通过资料查找，需要使用桥接模式来连接我们的物理机，但是多次尝试后，修改 IP，虚拟机无法连接到网络，最终放弃。尝试着去开三个虚拟机，一步步完成实验，让我对 DNS 解析有了进一步的了解，特别是在配置文件时，有一个域文件，就会有一个反向解析文件，一一对应。在攻击机上构造虚假响应时，总是失败，多测试几次，终于有成功伪造响应，成功完成了实验。DNS 更进一步的实验室远程攻击，这个如果有机会，我愿意去尝试一下。通过这个实验，我了解到 DNS 攻击的一种方式，同时也意识到网络攻击无处不在，我们更要加强安全意识。

其实在选择 DNS 本地攻击前，尝试着做过包嗅探和包欺骗，但是根据实验包嗅探和包欺骗只做了一部分，后面的部分有一些困惑也一直没有得到解决，就暂时搁置了，不能算完全完成，就不在此赘述。

总之，通过这次实验，发现自己的动手实践能力真的有待提高，学习不能只是理论，要多去动手实践操作才能加深对理论的印象和理解。最后，感谢王老师以及学长们的帮助和点评。