Remote DNS cache Poisoning Attack Lab

Lab Setup

DNS Server	:	10.0.2.24
Attacker	:	10.0.2.22
Victim	:	10.0.2.23

Task 1: Configure the Local DNS Server

Step 1: Configure the BIND9 Server.
BIND9 gets its configuration from a file called /etc/bind/named.conf. This file is the primary configuration file, and it usually contains several "include" entries. One of the included files is called /etc/bind/named.conf.options. This is where we typically set up the configuration options. Let us first set up an option related to DNS cache by adding a dump-file entry to the options block.

Step 2: Turnoff DNSSEC
We turn the protection against spoofing the DNS server off. This is done by modifying the named.conf.options file. We comment out the dnssec-validation entry, and add a dnssec-enable entry.

Step 3: Fix the Source Ports
For the sake of simplicity, we assume that the source port number is a fixed number. We can set the source port for all DNS queries to 33333. This can be done by adding the following option to the file /etc/bind/named.conf.options.

Step 4: Remove the example.com zone
In the previous lab, you have probably configured the local DNS server Apollo to host the example.com domain.In this lab, this DNS server will not host that domain, so please remove its corresponding zone from /etc/bind/named.conf.

Step 5: Start DNS server
We start the DNS server using the command:

$ sudo service bind9 restart

Note: If bind9 server is not already installed, install using the command

$ sudo apt-get install bind9

Task 2: Configure the Victim and Attacker Machine

1. Open Edit Connection
2. Select IPv4 Settings

3. Choose Method as Automatic (DHCP) addresses only
4. Enter the IP Address of YOUR DNS Server in the DNS servers field

Tasks 3.1 The Kaminsky attack:

The main objective of this attack is to redirect the user to another machine B when the user tries to get to machine A using A's host name. For example, assuming www.example.com is an online banking site. When the user tries to access this site using the correct URL www.example.com, if the adversaries can redirect the user to a malicious website the looks very much like www.example.com, the user might be fooled and give away his/her credentials to the attacker.

The Kaminsky attack:

We configure the attacker machine, so it uses the targeted DNS server as its default DNS Server as its default DNS server. The attacker machine is on the same NAT network.

This attack is performed by spoofing DNS Request followed by DNS Replies.

Task 1.1: Spoofing DNS Request

In this task, we will spoof DNS Requests that trigger the target DNS server to send out DNS queries, so we can spoof DNS replies.

1. Start wireshark on the DNS Server.
2. Run the code assigned for this task on the Attacker Machine.
   $ sudo gcc -lpcap dns_request.c -o req
   $ sudo ./req "Victim_IP" "DNS_IP"

Insert wireshark captures of the DNS Request part and explain your observations.

Task 1.2: Spoofing DNS Replies

In this task, we will spoof DNS Responses to the local DNS Server for each query. We will create a DNS Header with DNS Payload with the Answer, Authority and Additional section. The answer section will give the IP address of the query domain, the authoritative section fills the authoritative nameserver for the query domain. So, after the attack is successful, any query with the domain name will be directed to the Attacker's nameserver "**ns.dnslabattacker.com**". Lastly, we will fill the additional section with the IP Address of the name server.

1. Start wireshark on the DNS Server.
2. Run the code assigned for this task on the Attacker Machine.
   $ sudo gcc -lpcap dns_reply.c -o rep
   $ sudo ./rep "Victim_IP" "DNS_IP"

Insert wireshark captures and explain your observations.

Task 3.2: The Kaminsky Attack

In this task, we will combine the above two tasks to perform Kaminsky Attack.
Write code to combine the above DNS Request and DNS Reply together and run it again. You may have to flush your DNS Cache before performing the attack.

Check the DNS Cache in the DNS server machine
sudo rndc dumpdb -cache
sudo gedit /var/cache/bind/dump.db

Insert screenshots and explain your observations.

Now, on the victim machine, we can check using "dig" command:

$ dig www.example.com

Provide your screen shot, observation

Task 3.3: Result Verification

If your attack is successful, Apollo's DNS cache the NS record for example.com becomes ns.dnslabattacker.net. To make sure that the attack is indeed successful,

We first configure the victim's DNS server Apollo. Find the file named.conf.default-zones in the /etc/bind/ folder, and add the following entry to it:

```
zone "ns.dnslabattacker.net" {
        type master;
        file "/etc/bind/db.attacker";
 };
```

Create the file /etc/bind/db.attacker, and place the following contents in it. We let the attacker's machine and ns.dnslabattacker.net share the machine (192.168.0.200). Be aware that the format of the following contents can be messed up in the PDF file if you copy and paste. We have linked the file db.attacker to the lab's website.

```
$TTL 604800
@ IN SOA localhost. root. localhost. (
```

2; Serial
604800 ; Refresh
86400 ; Retry
2419200 ; Expire
604800) ; Negative Cache TTL;
@ IN NS ns.dnslabattacker.net.
@ IN A 192.168.0.200
@ IN AAAA ::1

We need to configure the DNS server, so it answers the queries for the domain example.com. Add the following entry in /etc/bind/named.conf.local :

zone "example.com" {
type master;
file "/etc/bind/example.com.db";
};

Create a file called /etc/bind/example.com.db, and fill it with the following contents. Please do not directly copy and paste from the PDF file, as the format may be messed up.

```
$TTL 3D
@               IN          SOA ns.example.com. admin.example.com. (
                2008111001
                8H
                2H
                4W
                1D)
@               IN           NS          ns.dnslabattacker.net.
@               IN           MX          10 mail.example.com.
www             IN           A           1.1.1.1
mail            IN           A           1.1.1.2
*.example.com   IN           A           1.1.1.100
```

When the configurations are finished, do not forget to restart both Apollo's and the attacker's DNS servers; otherwise, the modification will not take effect. If everything is done properly, you can use the command like "dig www.example.com on the user machine. The reply would be 1.1.1.1, which is exactly what we put in the above file.

Provide screenshots and wireshark observations.