Advanced Computer Networks

Exercise – Develop DNS Enumeration using Scapy

In this task, you will implement a Python script that performs DNS enumeration

The script will accept the domain name as a parameter (not using the **input** command). For example, running the script will look like this:

```
dnsenum.py jct.ac.il
```

Note: It is sufficient to do the entire exercise only on IPv4. The goal is to demonstrate your knowledge. Of course, you are welcome to add IPv6 as a challenge!

Step 1 – Find the DNS server of the desired domain

Start Of Authority (SOA) queries contain administrative information about the domain.

For example, the following is a response to a query performed on `jct.ac.il`:

```
C:\Users\BARAK>nslookup -type=SOA jct.ac.il
Server:  UnKnown
Address:  2a0d:6fc2:131c::1

Non-authoritative answer:
jct.ac.il
        primary name server = dns.jct.ac.il
        responsible mail addr = hostmaster.jct.ac.il
        serial  = 2024022900
        refresh = 43200 (12 hours)
        retry   = 7200 (2 hours)
        expire  = 2419200 (28 days)
        default TTL = 86400 (1 day)
```

Using **scapy**, create an SOA DNS packet. Extract the DNS server name of the requested domain from the reply.

Step 2 – Try DNSMAP tools

Linux or WSL must be used (Recommended due to simplicity of installation.)

Download the dnsmap tool used by penetration testers. Read how it is used.

Run the tool while sniffing with Wireshark.

```
sudo apt install dnsmap

dnsmap jct.ac.il
```

Research the captured packets and answer the question – how does the receiving party know whether the requested domain exists?

Step 3 – Prepare a file with options for trying it

Download the files wordlist_TLAs.txt and dnsmap.h.

https://github.com/makefu/dnsmap/blob/master/dnsmap.h

https://github.com/makefu/dnsmap/blob/master/wordlist_TLAs.txt

Copy the relevant content from these files into a file (name it whatever you want.) You may add words to the list for experimentation as you wish.

Step 4 – Create a dnsenum Python script

Create a Python script that uses the file from the previous step to map a selected domain and determine which servers are in the domain and their IP addresses.

Full credit is awarded if:

- The requested domain name is input as a parameter for the script, not through the **input** function.
- You should contact the DNS server of the requested domain you found in Step 1, not using a generic DNS server such as Google DNS (try and see that there are differences in the answers).
- You must print the server name and IP4 addresses for each server name in the requested domain.
- You need to find **all** the IP addresses for a particular server. For example, testing `jct.ac.il` should show that `mail.jct.ac.il` server has multiple IP addresses.

Good luck!