Name:Aditya Tomer
SBU ID: 111491409
CSE508: Network Security, CSE 508

Homework 4: DNS Packet Injection

Short Description:

ENVIRONMENT

OS:

MAC - Yosemite - 10.10.4

UBUNTU - 16.04

Modules To Install to MAC =>

Install scapy

```
git clone https://github.com/phaethon/scapy
cd scapy
sudo python3 setup.py install
```

Install libdnet =>

```
brew install libdnet
```

Install netifaces

```
pip3.6 install netifaces
```

DNS Injector =>

The dns injector 'dnsinject' capture the traffic from a network interface in promiscuous mode and inject forged responses to selected DNS A requests with the goal to poison the resolver's cache.

Format:

dnsinject [-i interface] [-h hostnames] expression

Where,

-i Listen on network device <interface> (e.g., eth0). If not specified, dnsinject should select a default interface to listen on. The same interface should be used for packet injection.

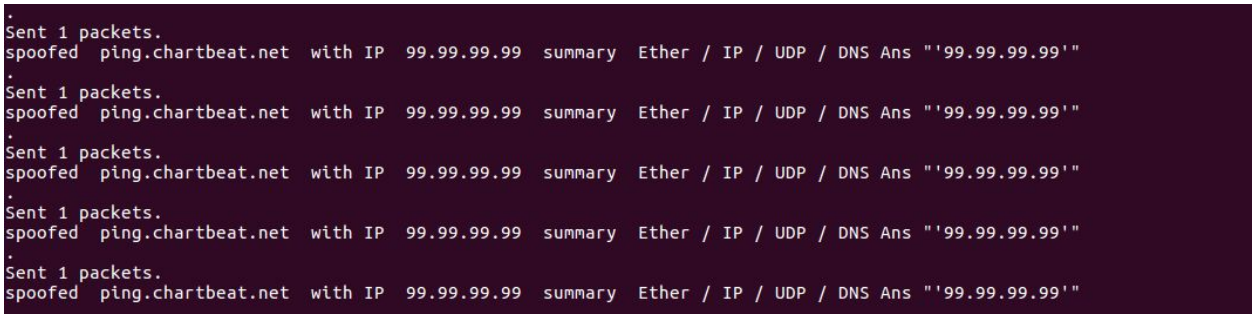
-h Read a list of IP address and hostname pairs specifying the hostnames to be hijacked. If '-h' is not specified, dnsinject should forge replies for

all observed requests with the local machine's IP address as an answer.

<expression> is a BPF filter that specifies a subset of the traffic to be monitored. This option is useful for targeting a single or a set of particular victims.

The <hostnames> file should contain one IP and hostname pair per line, separated by whitespace, in the following format:
10.6.6.6 foo.example.com
10.6.6.6 bar.example.com
192.168.66.6 www.cs.stonybrook.edu

WORKING EXAMPLE - INJECT



###############
Implementation

Commands =>
sudo python3 dnsinject.py -i en1 udp
Here -i en1 is the interface and udp is the bpf filter expression.

Given the filter expression, the program starts sniffing for network packets on given network interface for DNS packets for query type A. Given the hostname file mapped with spoofed IP address by the attacker. The attacker waits for DNS packets for query type A and whenever the hostname of victim matches with the hostname provided in the hostname file, the attacker forges the DNS response with same ID, reversing the addresses and packets are flooded back to the victim such that it poison the DNS cache of the victim.

##############

Part -2 DNS Poisoning Attack Detector

It captures the traffic in promiscuous mode and detects any DNS poisoning attack attempts.

Format:
dnsdetect [-i interface] [-r tracefile] expression

Where,
-i = interface (e.g. eth0). If not specified all the interfaces are scanned.

-r = Read packets from <tracefile>. Tracefile must be in pcap format, otherwise an exception occurs.

<expression> is a BPF filter that specifies a subset of the traffic to be monitored. here ('udp port 53'). For detector it must be a single string (in quotes if multiple)

Once an attack is detected, dnsdetect prints Detected DNS transaction ID, attacked domain name, and the original and malicious IP addresses

20160406-15:08:49.205618 DNS poisoning attempt
TXID 0x5cce Request www.example.com
Answer1 [List of IP addresses]
Answer2 [List of IP addresses]

False positive is taken care by taking the buffer size small such that only a small amount of packets within a few milliseconds is stored in the buffer for comparison is made. Incase every fields of DNS matches except the final answer/response against the query, it is marked as an attempt to poison the DNS cache.

WORKING EXAMPLE - DETECT

```
DNS poisoning attempt detected
TXID  6454  Request URL  ping.chartbeat.net
Answer1 [107.22.229.59]
Answer2 [99.99.99.99]
DNS poisoning attempt detected
aditya@aditya:~/Desktop/4$
```

PCAP File output

```
aditya@aditya:~/Desktop/4$ sudo python3 dnsdetect.py -r ~/Downloads/aditya.pcap
[sudo] password for aditya:
WARNING: No route found for IPv6 destination :: (no default route?). This affects only IPv6
interface => None , traceFilePath=> /home/aditya/Downloads/aditya.pcap , bfpExpr=> None
bfpExpr => udp port 53 , interface=> enp0s3
DNS poisoning attempt detected
TXID  20571  Request URL  bbc.com
Answer1 [212.58.246.79]
Answer2 [90.24.224.78]
DNS poisoning attempt detected
TXID  20571  Request URL  bbc.com
Answer1 [212.58.246.79]
Answer2 [90.24.224.78]
DNS poisoning attempt detected
TXID  20571  Request URL  bbc.com
Answer1 [212.58.246.79]
Answer2 [90.24.224.78]
DNS poisoning attempt detected
TXID  20571  Request URL  bbc.com
Answer1 [212.58.246.79]
Answer2 [90.24.224.78]
```

How to compile:

DNS Injector =>

sudo python3 dnsinject.py [-i interface] [-h hostnames] expression

DNS Detector =>

sudo python3 dnsdetect.py [-i interface] [-r tracefile] expression