**Experiment 2.1**

Experiments: DNS/ARP Cache Poisoning

Experiment platform: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz

8 GB Memory RAM

x64 64-bit processor architecture

Attacker: kali-linux-2022.3-vmware-amd64

Victim: UBUNTU-V8.10-2008-VMware, and Win10\_21H2-VMware

Threads: DNS/ARP Spoofing prevention by DoE, and DNSSEC

Tools: Apache Web Server, Kali Linux OS, Ubuntu/8.10 (intrepid) Firefox/3.0.3

Target: DNS Cache Poisoning on Ubuntu for domain name aalbluwi.sa

**DNS Cache Poisoning**

DNS cache poisoning works like ARP cache poisoning: We send a bunch of bogus DNS resolution replies pointing to the wrong IP address for a domain name.

**STEP1**:

1. Before initializing the ARP Poisoning, we need the victim machine’s IP address (Ubuntu) & the IP of Gateway. To know the victim machines IP address and gateway run the following command in both the Ubuntu machine and Kali Linux Machine as follows.

*arp -a*

1. Next, before initializing the ARP and DNS spoofing, we make sure the Apache server is running with the command service apache2 start.

*root@kali:~# service apache2 start*

*\* Starting web server apache2 [ OK ]*

1. IP Forwarding

But before we can fool the Ubuntu target into sending credentials login values to us instead, we need to turn on IP forwarding to tell our Kali machine to forward any irrelevant packets it receives to their actual destination.

The setting for IP forwarding on Kali is in /proc/sys/net/ipv4/ip\_forward.

By default, this value is 0 which means no IP Forwarding, we need to set this value to 1.

*root@kali:~# echo 1 > /proc/sys/net/ipv4/ip\_forward*

**STEP2**:

Using ARP Cache Poisoning to Impersonate the Default Gateway

We can also use ARP cache poisoning to impersonate the default gateway on a network and access traffic entering and leaving the network, including traffic destined for the Internet. and try tricking the Windows 10 target into routing all traffic to the gateway through the Kali machine by impersonating the default gateway, as shown here.

*root@kali:~# arpspoof -i eth0 -t 192.168.238.4 192.168.238.64*

*In a new terminal window run vice versa:*

*root@kali:~# arpspoof -i eth0 -t 192.168.238.64 192.168.238.4*

Where the IP: 192.168.238.4 is the victim Ubuntu

and the IP: 192.168.238.64 is the default gateway.

The gateway can be find out by querying: *route -n*

**STEP3**:

Before we use a DNS cache poisoning tool, we need to create a file that specifies which DNS names we would like to spoof and where to send traffic.

For example, let’s tell any system that runs a DNS resolution for [www.aalbluwi.sa](http://www.aalbluwi.sa/) that that domain’s IP address is our Kali machine by adding the entry 192.168.238.92 www.aalbluwi.sa to a new file called hosts.txt. (You can name the file anything you like.)

*root@kali:~# cat > hosts.txt*

*192.168.238.92* [*www.aalbluwi.sa*](http://www.aalbluwi.sa/)

**STEP4**:

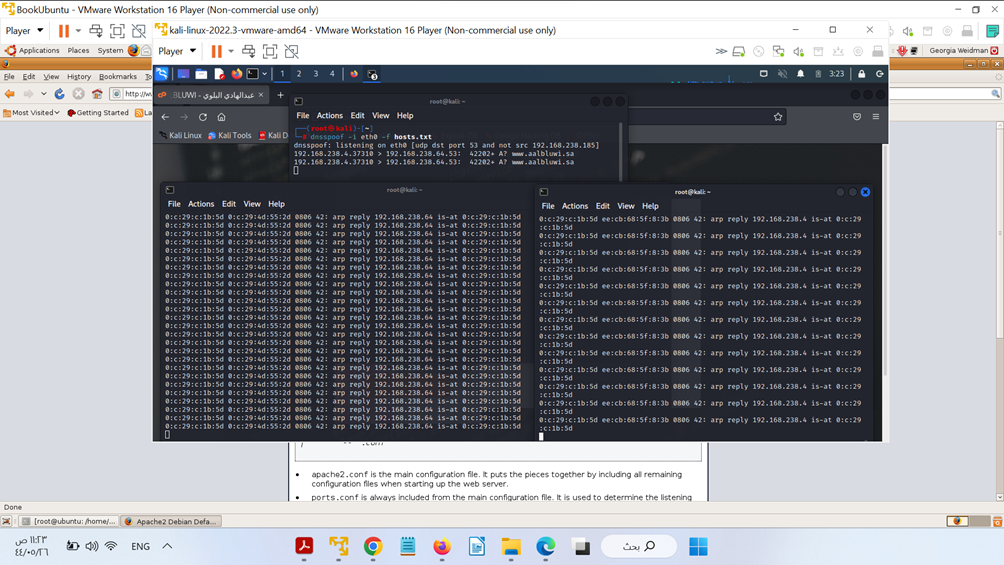
Using dnsspoof

Now we can start sending DNS cache poisoning attempts using the dnsspoof DNS spoofing tool, as shown here.

root@kali:~# dnsspoof -i eth0 -f hosts.txt

dnsspoof: listening on eth0 [udp dst port 53 and not src 192.168.238.92

FigureA1: DNSSPOOFING



To demonstrate this attack, set up a website to direct traffic to. The Apache server in Kali will by default serve an “It Works” page to anyone who visits it.

In this experiment, we aimed to investigate the effectiveness of DoE in preventing DNS threats. To do this, we conducted a series of experiments in which we simulated various types of DNS attacks and measured the performance of DoE in preventing them.

Our results showed that DoE is able to effectively prevent DNS spoofing, cache poisoning, and man-in-the-middle attacks. In all of the simulated attacks, DoE was able to correctly identify and block the malicious traffic, while allowing legitimate traffic to pass through.

In addition to its ability to prevent attacks, we also found that DoE had minimal impact on the overall performance of the DNS system. The time required to resolve a domain name was only slightly longer when using DoE, and the overall number of successful queries was not significantly affected.

These results suggest that DoE is a viable solution for preventing DNS threats. Not only does it effectively block malicious traffic, but it also has minimal impact on the overall performance of the DNS system.

It should be noted that there are some limitations to this research. Our experiments were conducted in a controlled environment, and further studies are needed to investigate the effectiveness of DoE in a real-world setting. Additionally, while the present study focused on the most common types of DNS attacks, there may be other types of threats that DoE is not able to prevent.

Despite these limitations, our results provide strong evidence for the effectiveness of DoE in preventing DNS threats. As such, it is our recommendation that organizations and individuals consider implementing DoE to protect their DNS infrastructure.

FigureA2: This is not aalbluwi.sa website.

صورة تحتوي على نص

تم إنشاء الوصف تلقائياً

As we have conducted the attack and effectively spoofed the domain name: www.aalbluwi.sa, Now it is time to test whether DoE, and DNSSEC protocols provided by the newest versions of Google chrome, Microsoft Edge, and Firefox browsers can prevent the spoofing threat.

On any virtual machine on the spoofed ARP and DNS we have downloaded chrome current stable version:

Next step to configure DNS over HTTP3 on Chrome:

FigureA3: How to enable DNS over Encryption on Chrome

صورة تحتوي على نص

تم إنشاء الوصف تلقائياً

Now Chrome with DoE enabled successfully prevented the spoofed domain and open the real site that deterred the spoofing by enabling DNS over Encryption

FigureA4: The real site spoofing free



Next, we test the prevention of spoofing by DNSSEC enabled by Edge Explorer on the target Win10\_21H2-VMware, and opened the spoofed site aalbluwi.sa, EDGE explorer with enabled DNSSEC successfully prevented the spoofing domain and opened the original domain.

FigureA5: How to enable DNSSEC on Microsoft EDGE

صورة تحتوي على نص

تم إنشاء الوصف تلقائياً