Assignment 3

Hareesh Nutalapati

October 17, 2016

1 Hosts File Attack

User machine IP address is 192.168.126.128
DNS server IP address is 192.168.126.129
Attacker machine IP address is 192.168.126.130
After the setup we run the command **dig www.example.com** on the user machine and the result is as shown in the screen shot.

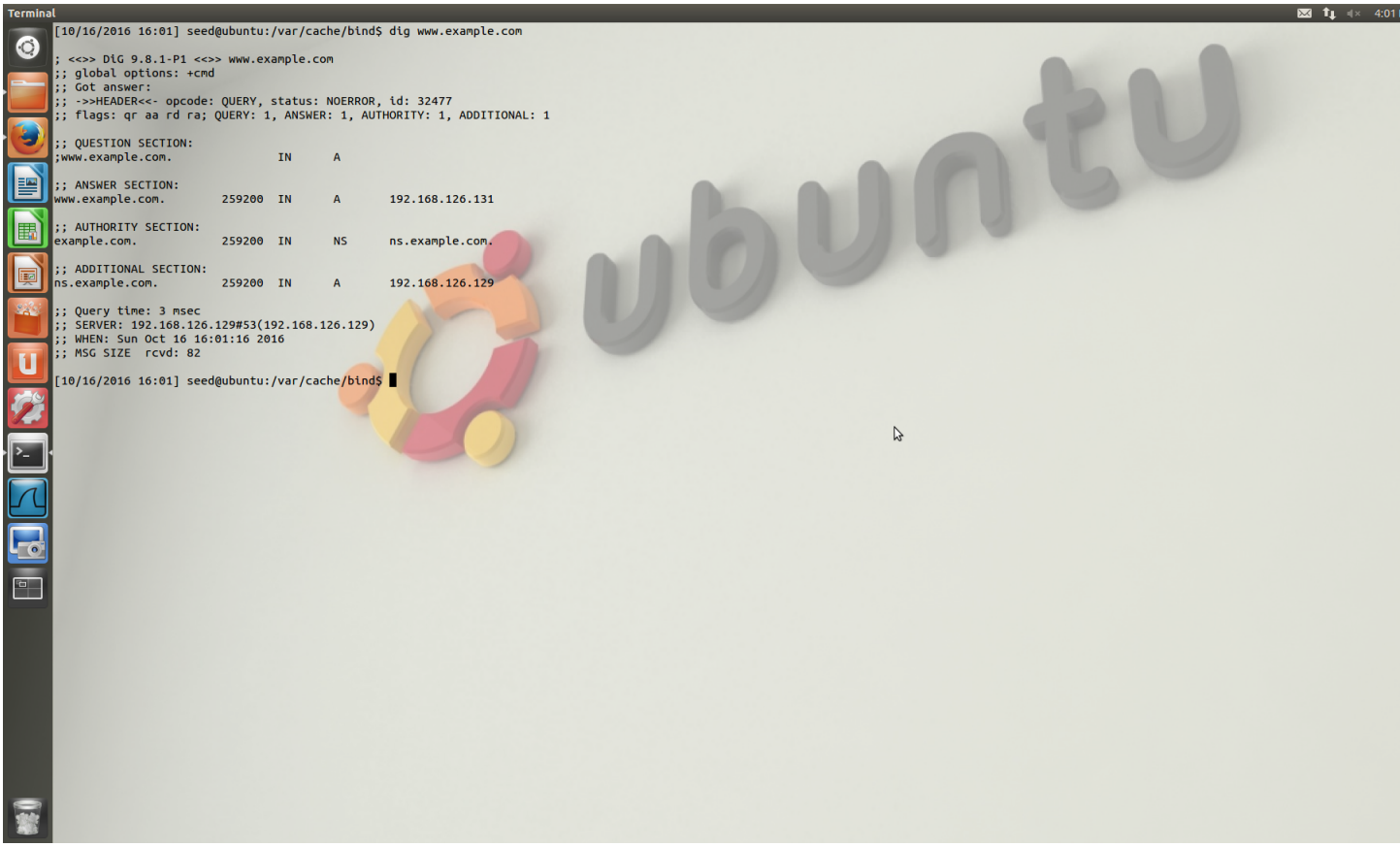

Figure 1: Dig command output

1. Description of attack
   Here we assume the attacker has already compromised the user machine. Local lookups uses the hosts

file(/etc/hosts) and it is also given more preference than the DNS lookup. Change of any entry here will open that IP address in the user machine without consulting any DNS server.

2. Attack steps

A. Open the Hosts file in /etc folder of the user machine using **sudo vim /etc/hosts**

B. Add an entry to the host file for www.example.com with any IP address. Here I added the IP address of www.rvrjcce.ac.in and saved the file using **esc:+w!:q**

Figure 2: Malicious entry for www.example.com

C. If we open www.example.com on the user machine it will redirect to the website www.rvrjcce.ac.in.

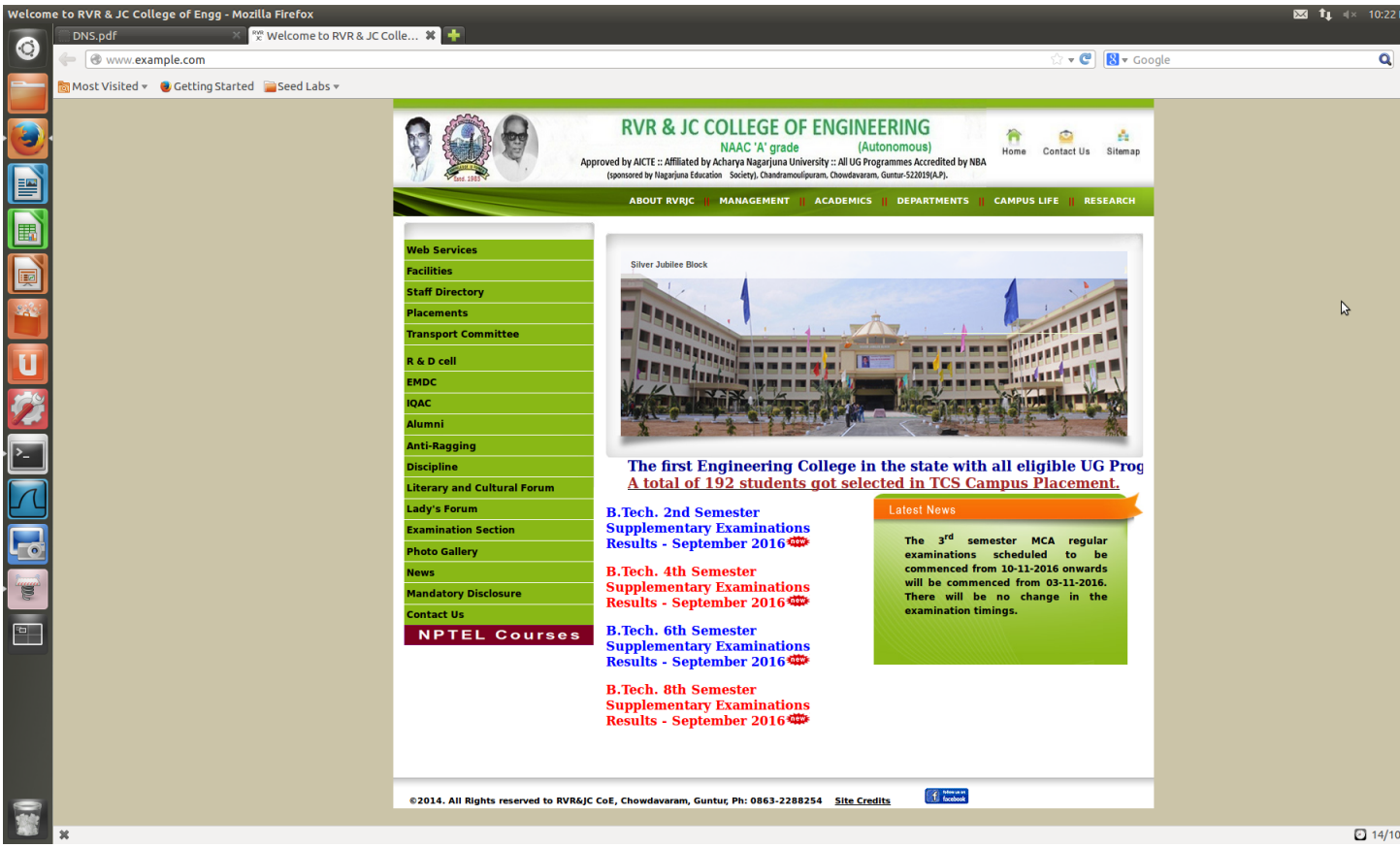

Figure 3: Malicious website openup

3. Leverage of attack in real world
   This attack can be used to steal the sensitive information from the user like banking passwords and username. This is achieved by redirecting him to the website which looks similar to the banking website.

4. Viability of attack
   This is a viable attack and can be achieved by injecting virus into the user machine which modifies the hosts file. But here we need to know the password of the user as modifying hosts file require root privilege.

2 Host-Level Response Spoofing

1. Description of attack
   Here, the user machine is not compromised by the attacker. The attacker sends a fake DNS response to the user as they are on same LAN. When user wants to goto www.example.com, a request is sent to DNS Server to resolve the IP address. Then the attacker sends a fake DNS response before the DNS Server reply back. When user opens www.example.com it leads to open a malicious site address sent by the attacker.

2. Attack steps
   A. As the attacker is on same LAN, he can see the query being sent from the user to DNS Server .

B. The attacker uses a netwag tool to send a fake response to the user. Parameters used in netwag tool are:
Hostname=www.example.com
Hostname IP=1.2.3.4
Auth Name Server=ns.example.com Authy Name Server IP=1.2.3.5
Filter=src host 192.168.126.128
After setting these parameters click on generate ana run the tool.

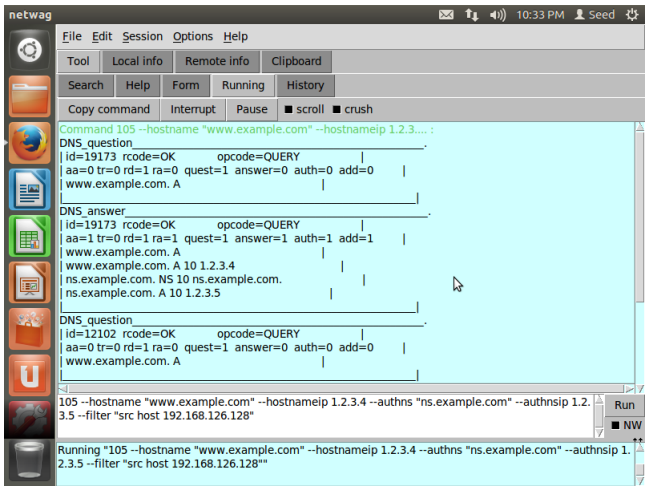

Figure 4: Output of netwag tool

C. On the user machine we can check the spoofed DNS response by using the command **dig www.example.com**.

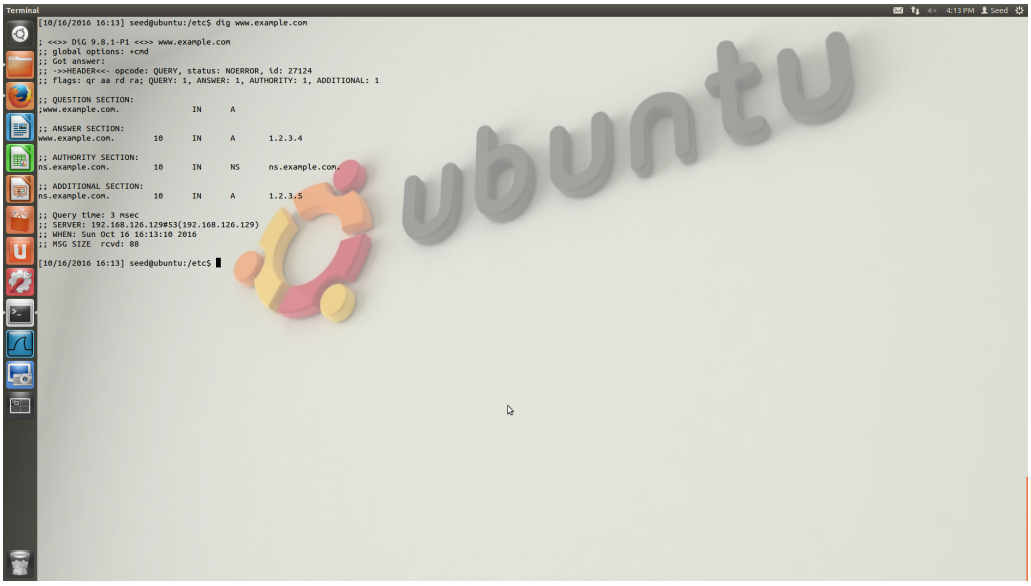

Figure 5: Output of dig command showing spoofed DNS response

3. Leverage of attack in real world
   This attack is used for phishing and to steal the sensitive information as the above Hosts file attack.

4. Viability of attack
   Here, the attacker needs to guess the query that the user is making which is not a easy task. Different type of requests need the tool to be run with different parameters. Considering these we can say that it is a less viable attack.

3 Server-Level Response Spoofing

1. Description of attack
   In this attack, the attacker would send the spoofed responses directly to the DNS Server. The attacker and the DNS Server needs to be on the same LAN. Whenever DNS Server receives a query and host name is not in the server's domain it checks the cache to see if it has an entry there. If there is an entry the server replies back with the information from the cache. If the cache does not have any entry for that domain, then the server will ask other DNS Servers. The other DNS Servers response is stored in the cache.
   Now, the attacker spoof the response of the other DNS Server and send that response before the other DNS Server sends the response. The server stores the malicious information sent by the attacker in its cache. This malicious information is stored in the server until the cache expires.

2. Attack steps
   A. We use the command **sudo rndc flush** to flush out the current cache in the DNS Server.
   B. The attacker uses the netwag tool to send fake response to the DNS Server. Parameters used in the netwag tool are:
   Hostname=www.google.com
   Hostname IP=1.2.3.4
   Auth Name Server=mail.google.com
   Auth Name Server IP=1.2.3.5
   TTL=200
   Filter=src host 192.168.126.129
   Spoof IP=raw

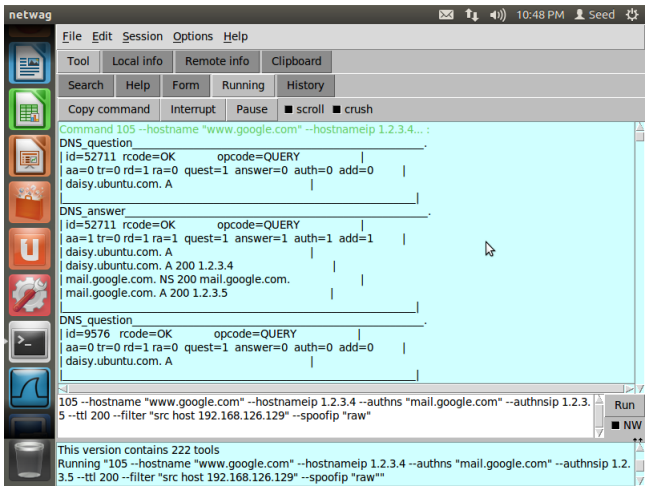

Figure 6: Output of netwag tool

C. Now the netwag tool is stopped and user runs the command **dig www.example.com** and output

shows that the DNS response it received is spoofed.

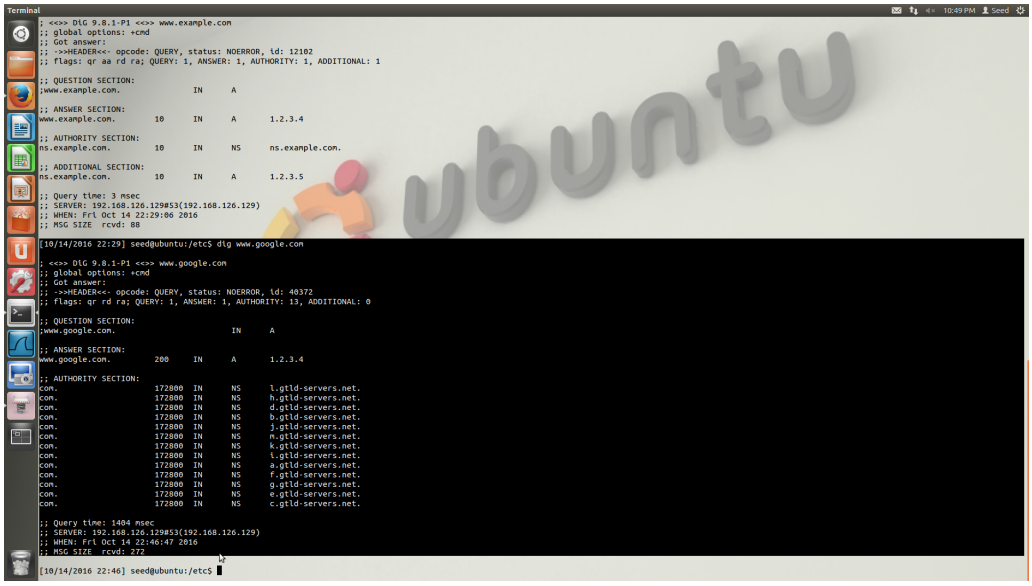

Figure 7: Output of dig command

D. We check the recorded entry in the cache by running the command **sudo rndc dumpdb -cache** and open the cache using **sudo vim /var/cache/bind/dump.db**

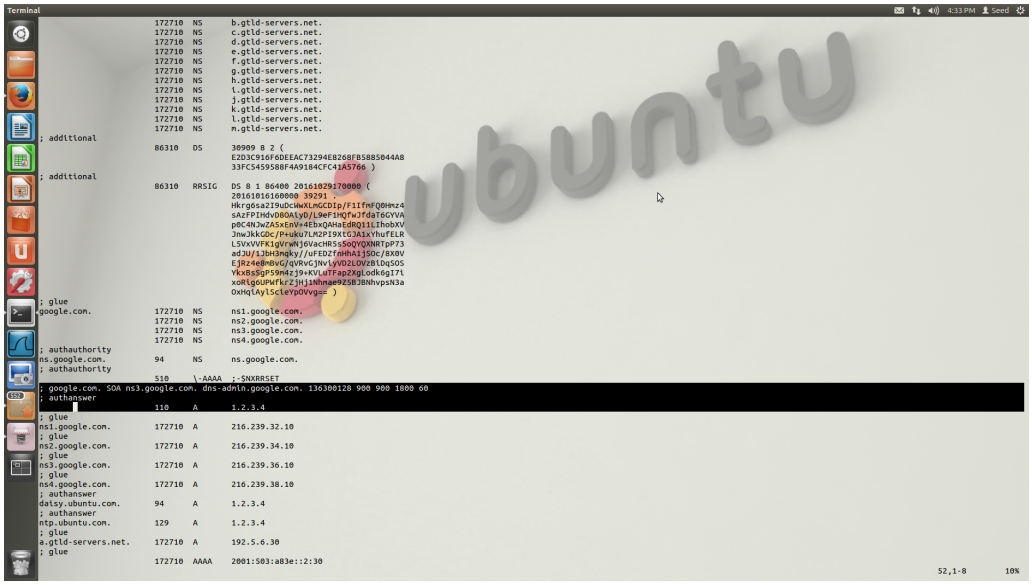

Figure 8: Cache file

3. Leverage of attack in real world
   This attack also leads to stealing of sensitive information. Here attacker poisons the cache of the DNS server which redirects all the users of that DNS server to the malicious site.

4. Viability of attack

Here the attacker and the DNS Server needs to be in the same network. The attcker needs to guess the queries that are made to the DNS Server and run the tool with different parameters, which makes the attack more difficult and less viable. But it is somewhat easier than the previous attack.

4 Kaminsky Attack

1. Description of attack
   Here the attacker is not on the same LAN as the DNS Server. We setup another DNS Server which handles forward requests for www.dnsphishing.com. If the user queries for some non-existing subdomain of www.dnsphising.com, then it goes to the first DNS Server and it could not resolve it and forward it to the next DNS Server. The attacker now needs to respond back with a fake DNS response before the other DNS Server respond back. Here the mentions that the address for www.dnsphishing.com has changed and that forces the first DNS Server to update the cache with the malicious IP address. The attacker needs to use the same transaction id which is used by the DNS Server to query the other DNS Server.

2. Attack steps
   A. The IP address of the user is 192.168.161.100
   The IP address of the DNS Server1 is 192.168.161.10
   The IP address of the DNS Server2 is 192.168.161.20
   The second DNS Server is also setup similar to the first one. The setting in the first DNS Server is modified to include the forward zone for www.dndphishing.com. The following lines are added to the /etc/bind/named.conf file.
   Zone "dnsphishing.com" type forward;forwarders(192.168.161.20);
   The above setting looks after that packets for this domain do not leak to the internet.

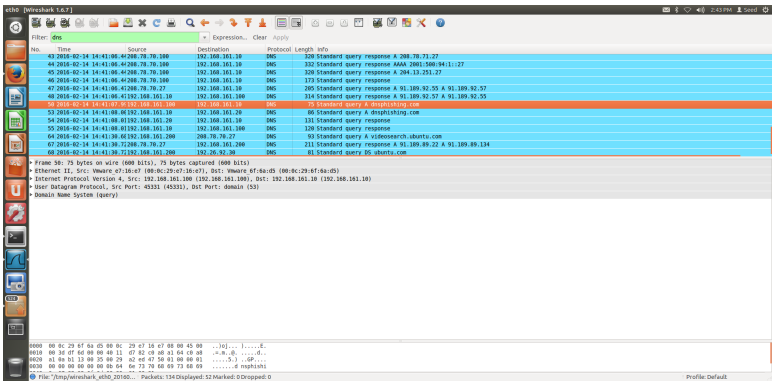

Figure 9: Packet sent from user machine to DNS Server1

The above figure indicate the packet forwarded from the user to the DNS Server1.

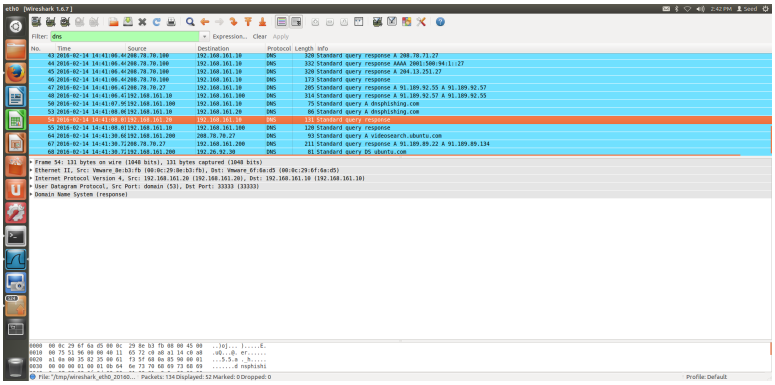

Figure 10: Query forwarded from DNS Server1 to DNS Server2

B. Some delay is injected in the DNS Server2 so that response from attacker reaches first than the response from the DNS Server2 to DNS Server1.
C. The pacgen tool is downloaded into the attacker machine which helps to send the packets from attacker to the DNS Server1. A function is added into pacgen file which generates random transaction id.
D. The attacker asks random sub domain queries of www.dnsphishing.com. At the same time pacgen is run to send spoofed DNS packets.
E. The attack was not successful for me as I was able to send only 1000 packets at most.

3. Leverage of attack in real world
   This attack is used to steal sensitive information and also redirect the users to fake websites. This is dangerous as the DNS Server cache is modified and users who query that DNS Server are redirected to malicious website.

4. Viability of attack
   The attack is easy if the attacker knows the transaction id of the DNS Server but there are 2^{16} ways to guess the transaction id. The attacker needs to try all possible ways before the DNS Server respond back. So, time is an important factor. Even though the attack fails in some attempts the attacker can keep trying by giving other non-existing names in the DNS Server. The attack can become viable in real life scenario.