COMPUTER NETWORKS SECURITY LABORATORY

ASSIGNMENT 4

BY: RAJDEEP SENGUPTA

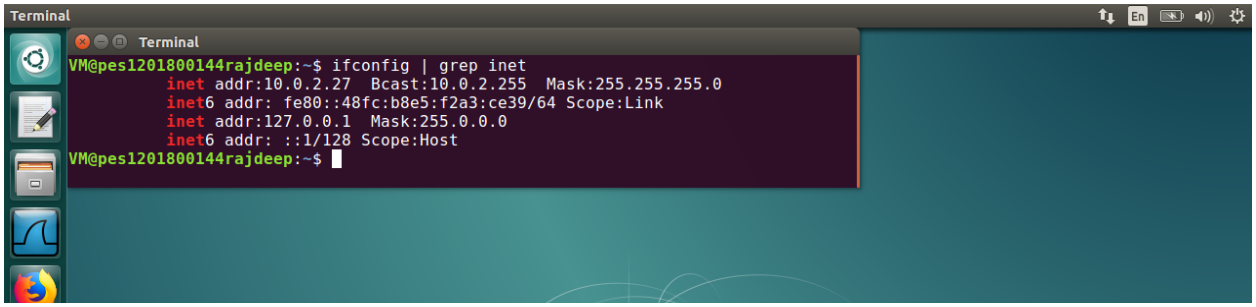
SRN: PES1201800144

SECTION: C

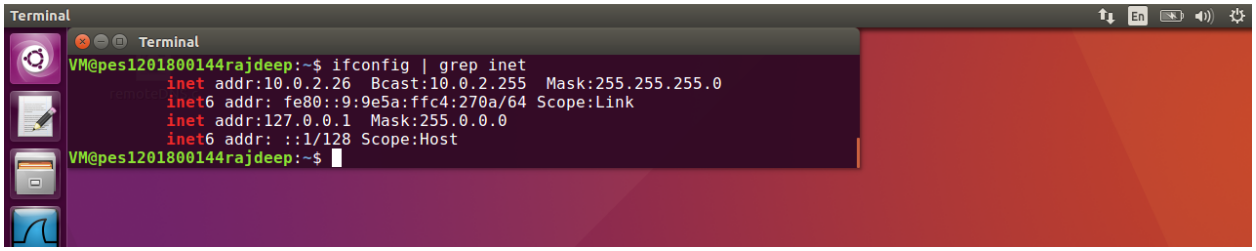
NOTE: Please find my SRN ‘PES1201800144rajdeep’ as the terminal username. Also find the description and result analysis and observation of each task in RED FONT following the screenshots for each task.

MY CONFIGURATION:

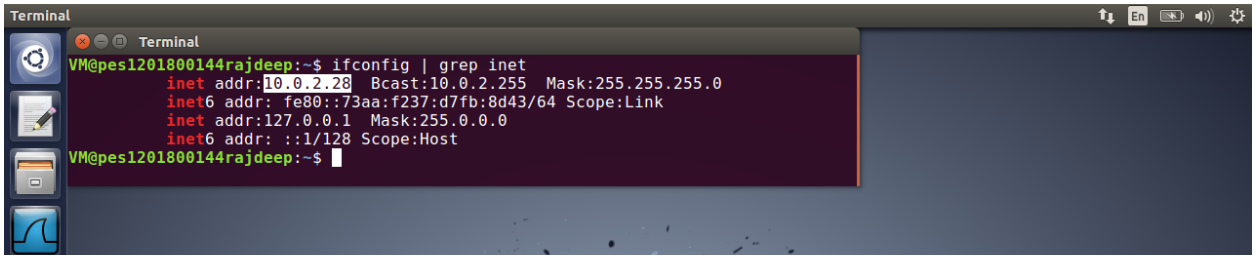
VM DNS Server: 10.0.2.28
VM Client: 10.0.2.27
VM Attacker: 10.0.2.26



Screenshot Client IP address: 10.0.2.27

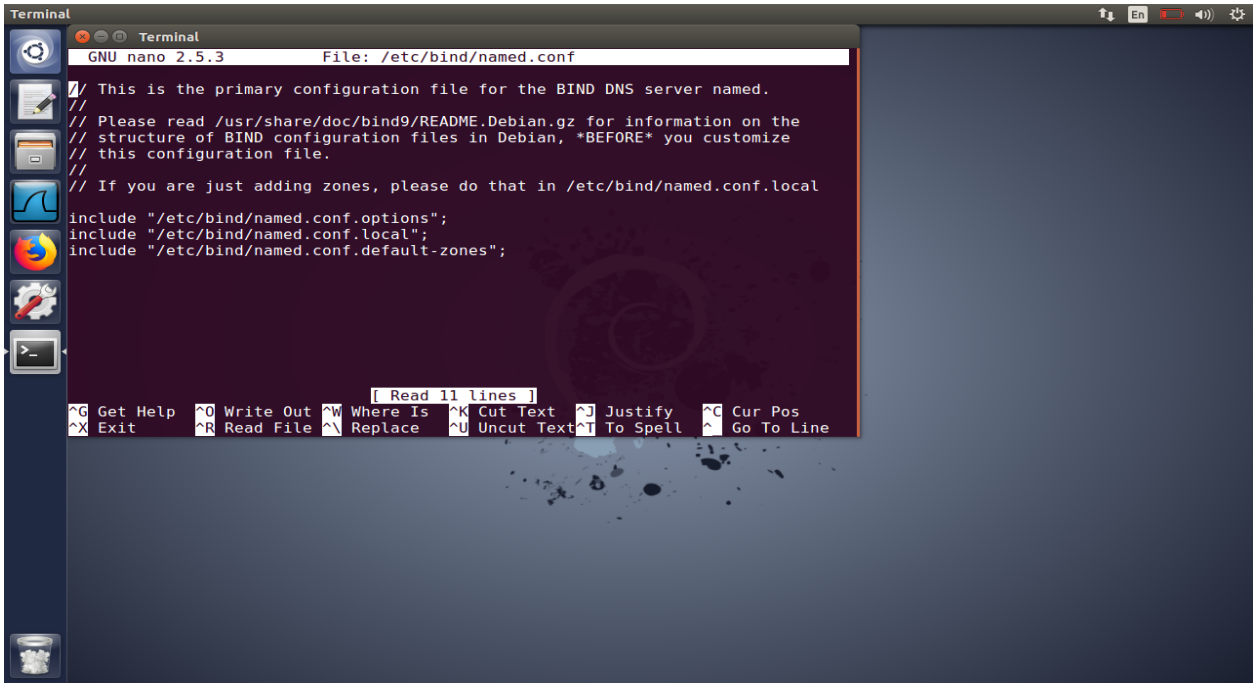


Screenshot Attacker IP address: 10.0.2.26

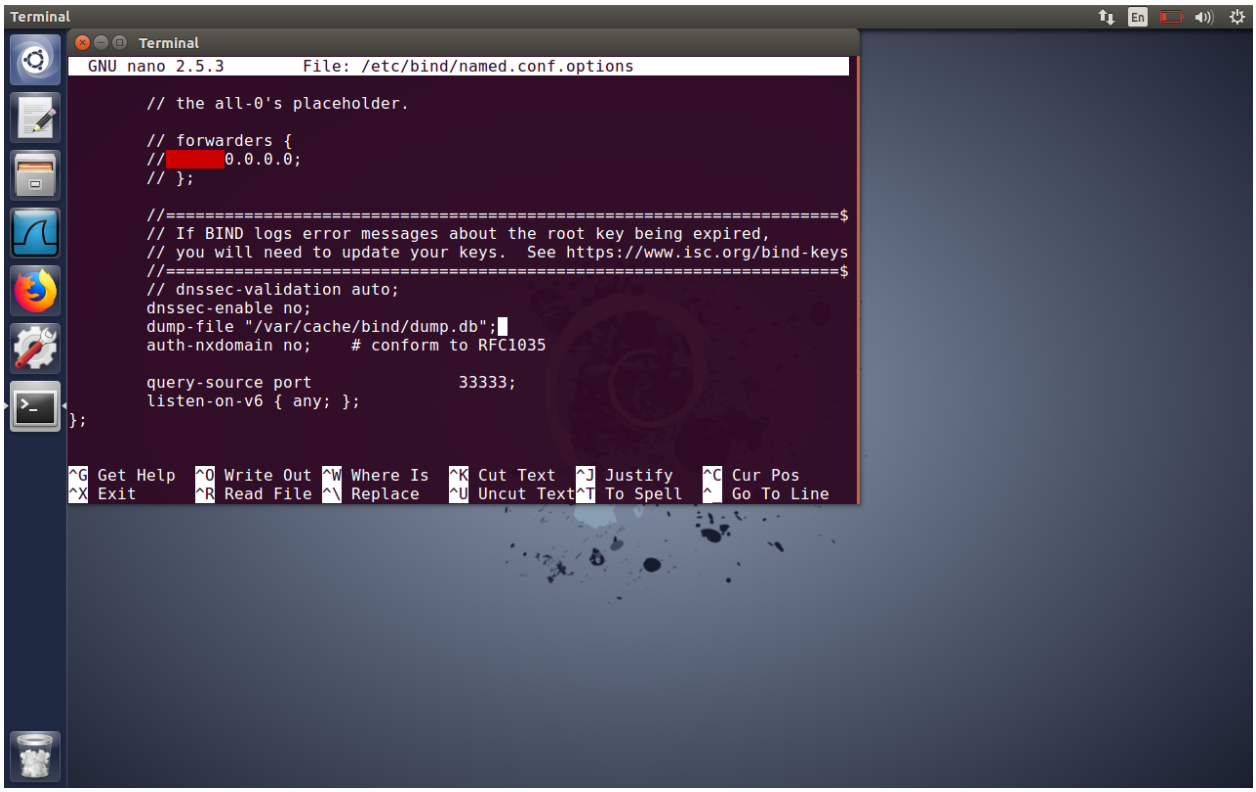


Screenshot DNS Server IP address: 10.0.2.28

TASK 1: SETTING UP DNS SERVER CONFIGURATIONS

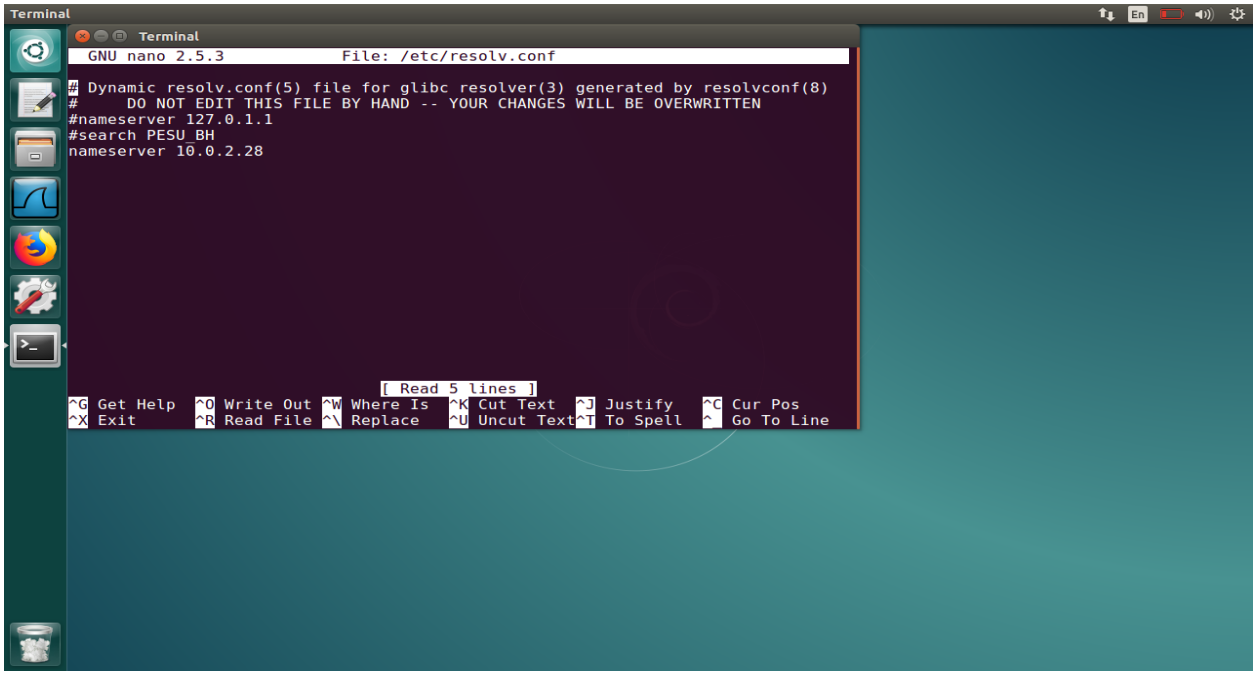


Screenshot 1.1: Including named.conf.options in /etc/bind/named.conf file

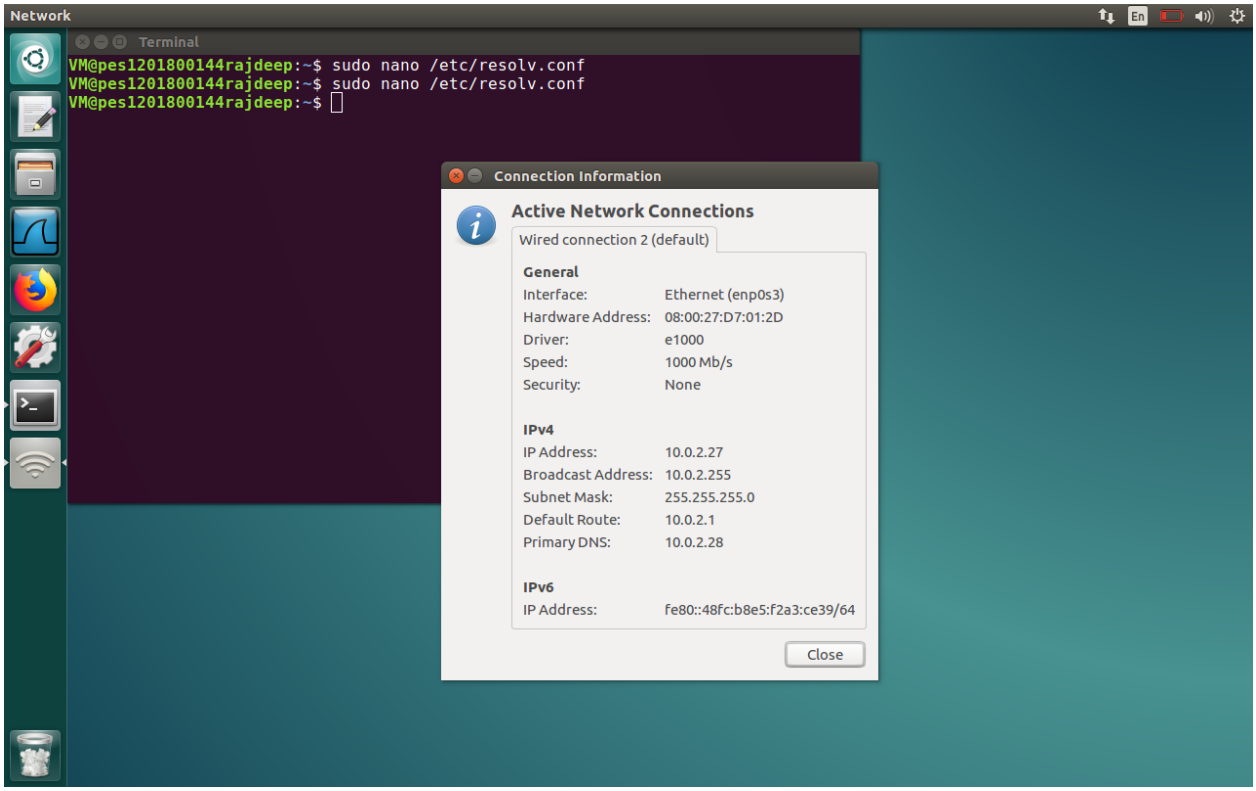


Screenshot 1.2: turning off DNS security, setting query port to 33333 and adding dump file location in named.conf.options file

TASK 2: SETTING UP CLIENT MACHINE CONFIGURATIONS

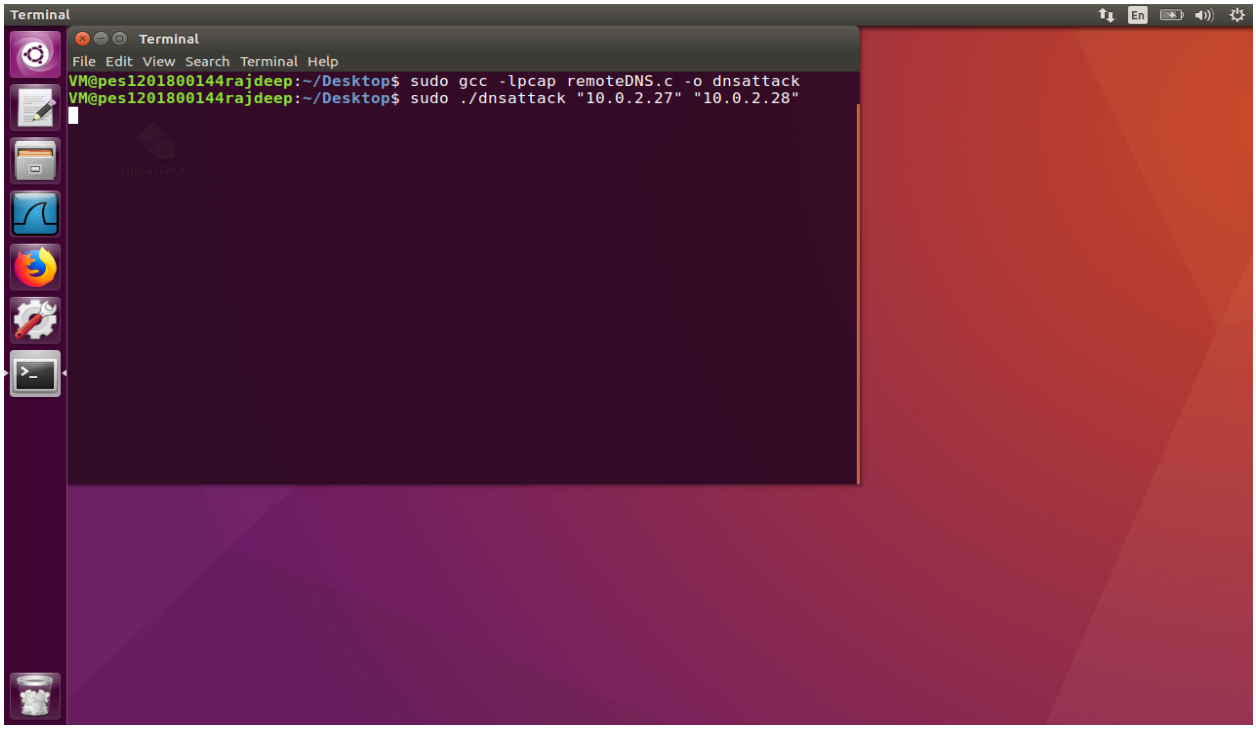


Screenshot 2.1: Setting DNS server for client machine as VM DNS server's IP address 10.0.2.28

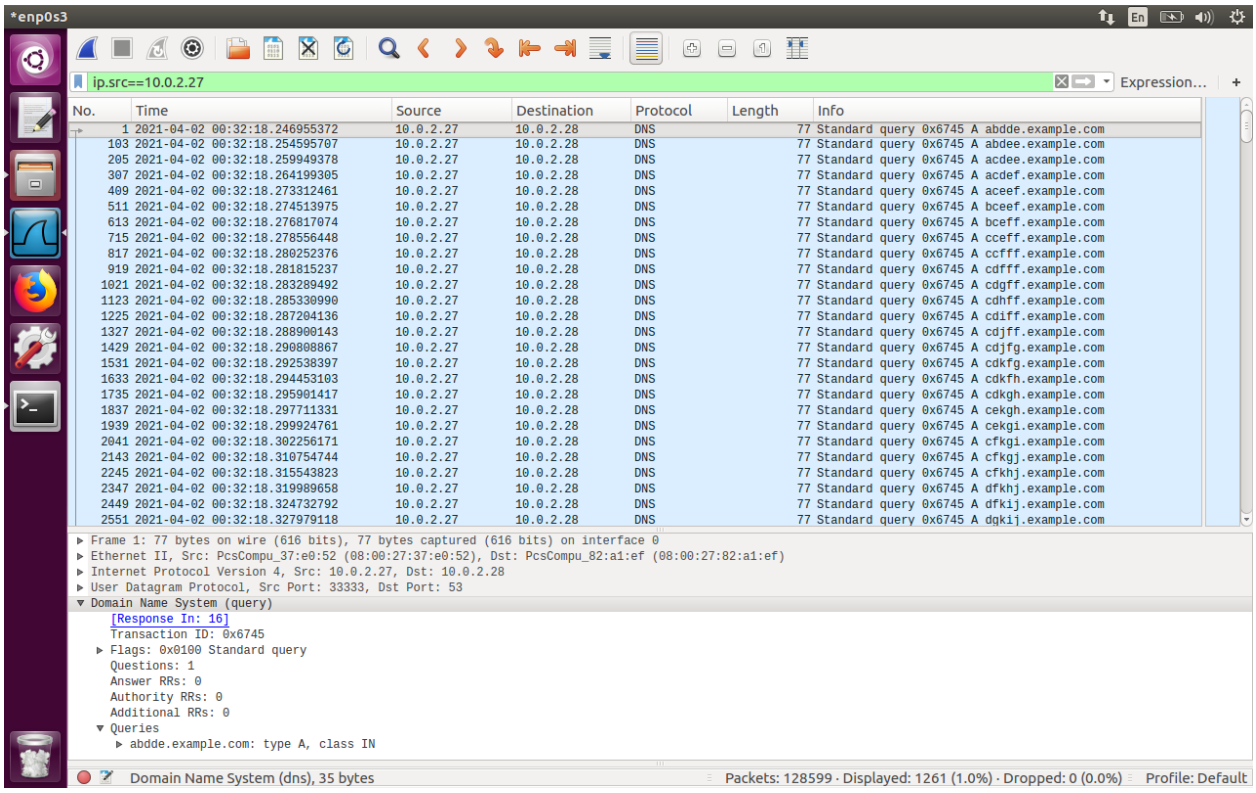


Screenshot 2.2: Setting DNS server as 10.0.2.28 in "edit connections" setting and displaying final configurations of client machine

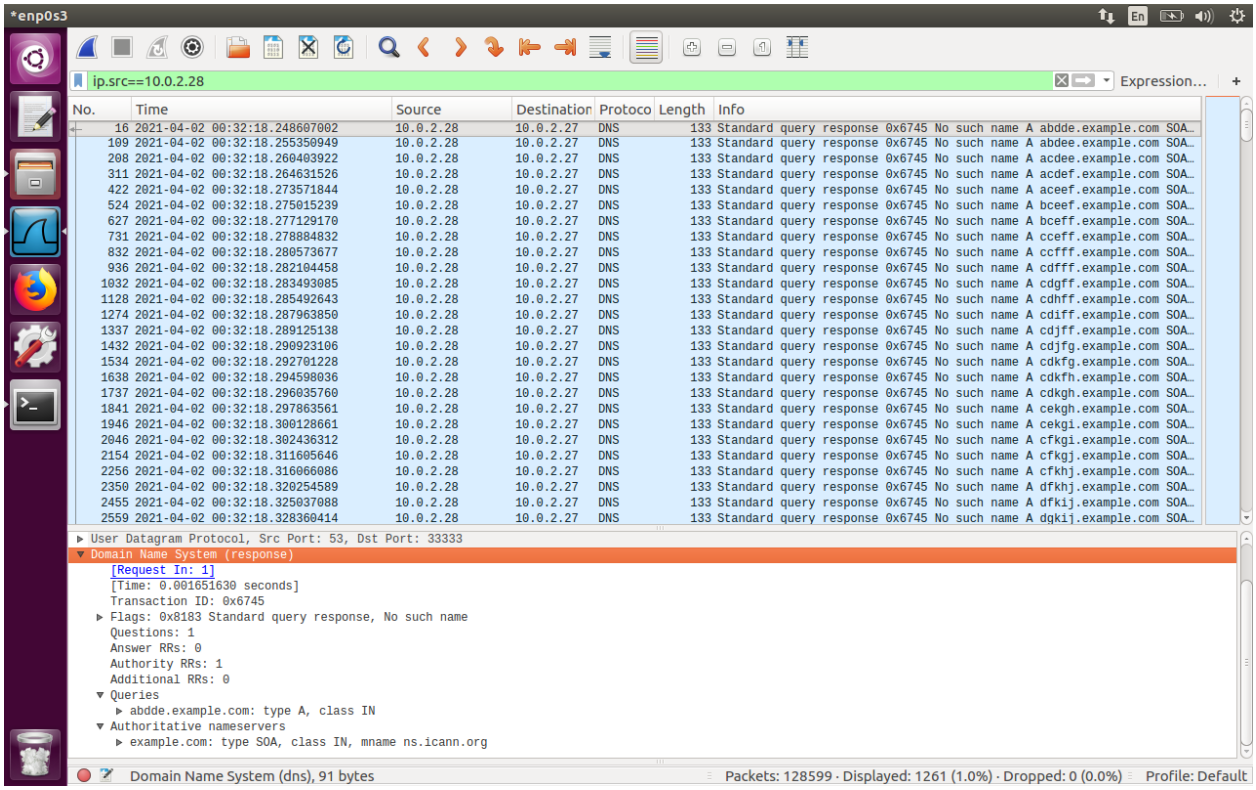
TASK 3: KAMINSKY ATTACK



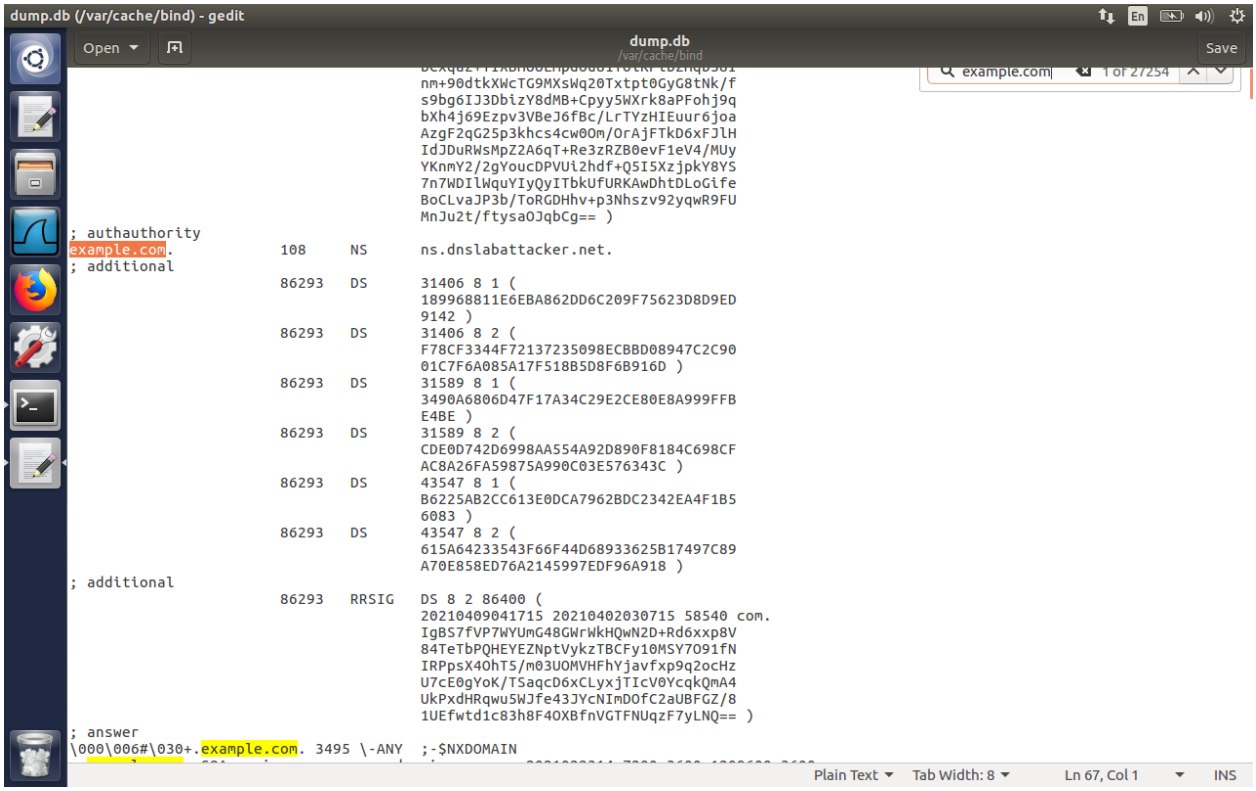
Screenshot 3.1.1: Compiling and executing the dns spoofing file



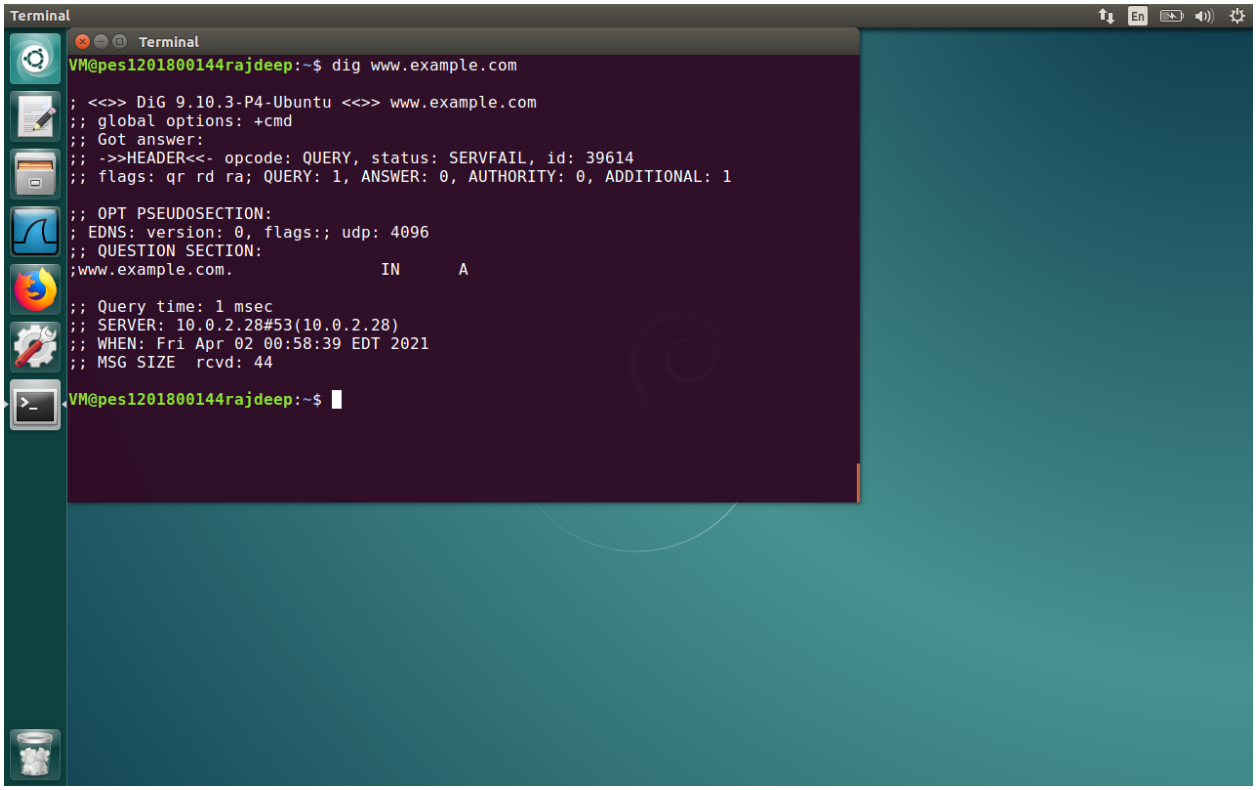
Screenshot 3.1.2: DNS requests on wireshark



Screenshot 3.1.3: DNS replies on wireshark



Screenshot 3.1.4: Dump file in DNS server VM



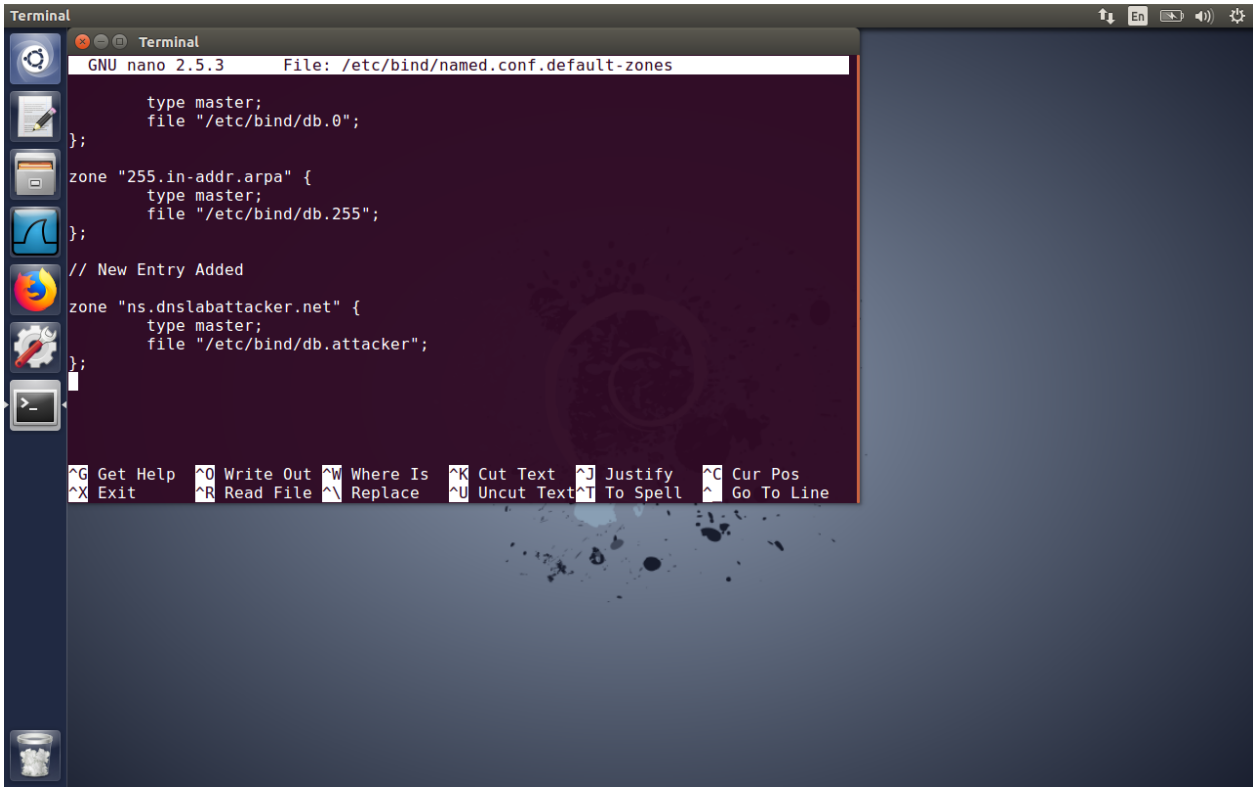
Screenshot 3.1.5: dig command on client machine

In Kaminsky attack, the attacker machine runs a spoofing program which sends different queries to the DNS server VM each time as it can be seen in Screenshot 3.1.2. Since the names are different, the DNS server VM queries example.com domain for the names each time. Before the reply arrives from the example.com nameserver, the attacker floods DNS server VM with spoofed replies. In the spoofed reply, the attacker sends ns.dnslabattacker.net as the authoritative nameserver. If transaction ID matches for the spoofed reply, the Kaminsky attack is successful and the DNS server VM's cache is poisoned.

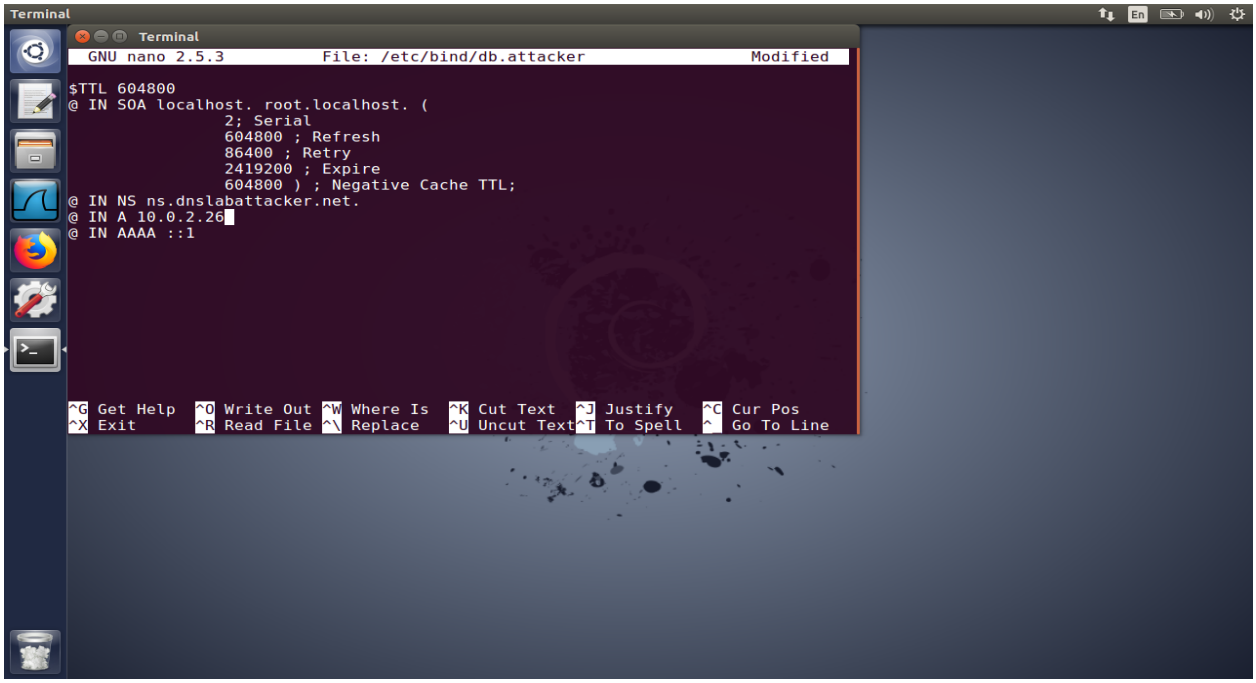
Sending different queries each time prevents caching effect. The result of the cache poisoning can be verified in the cache dump file as shown in Screenshot 3.1.4.

But in the dig command on the client machine, the IP address cannot be found in the Answer Section.

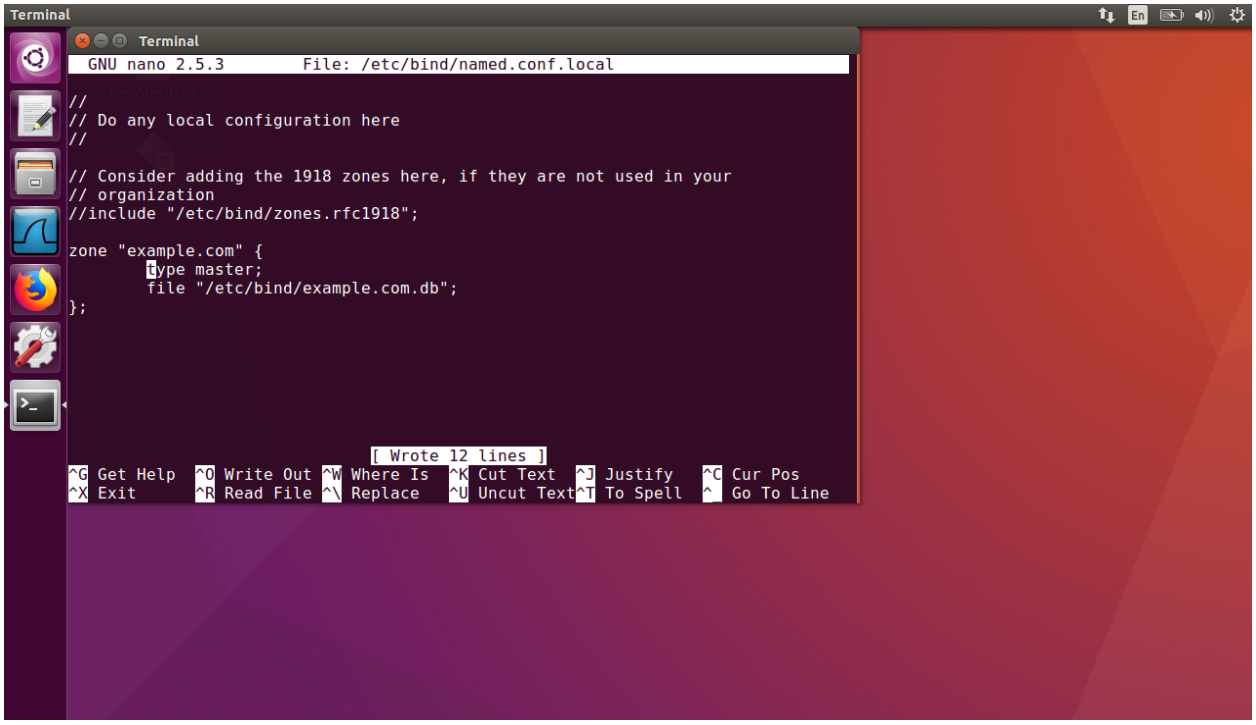
TASK 3.2: RESULT VERIFICATION



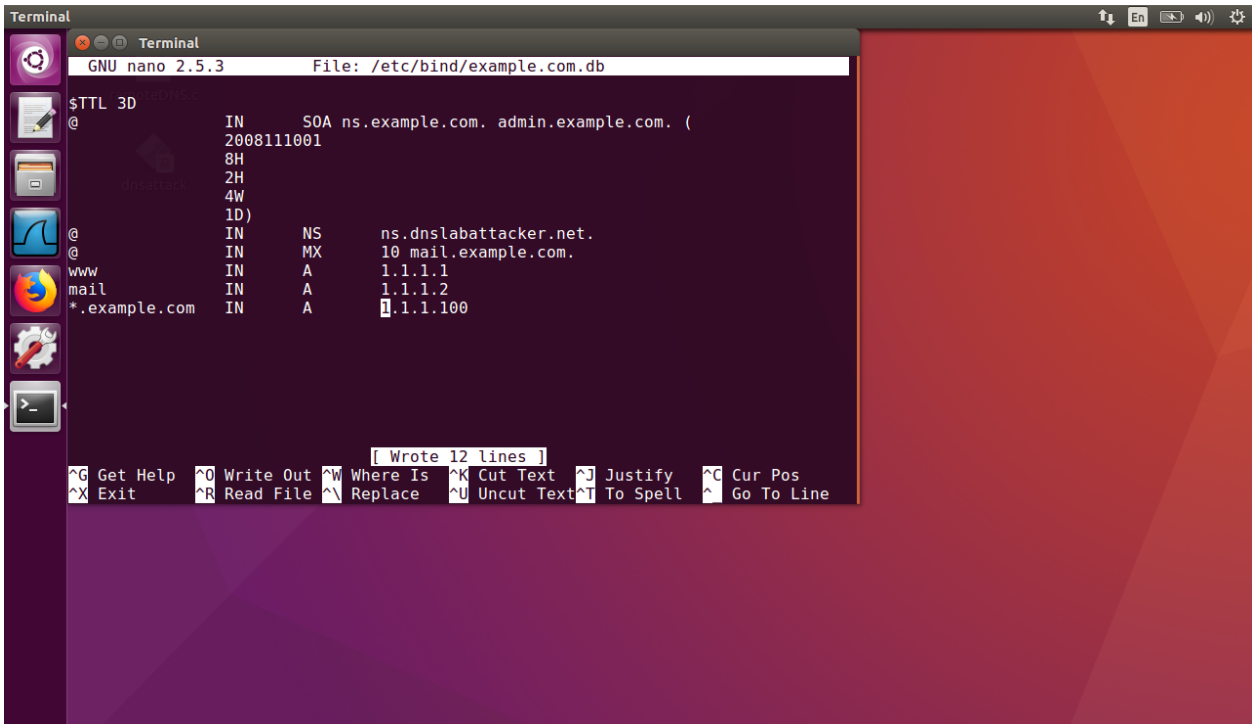
Screenshot 3.2.1: Adding zone file entry in DNS server VM in named.conf.default-zones file



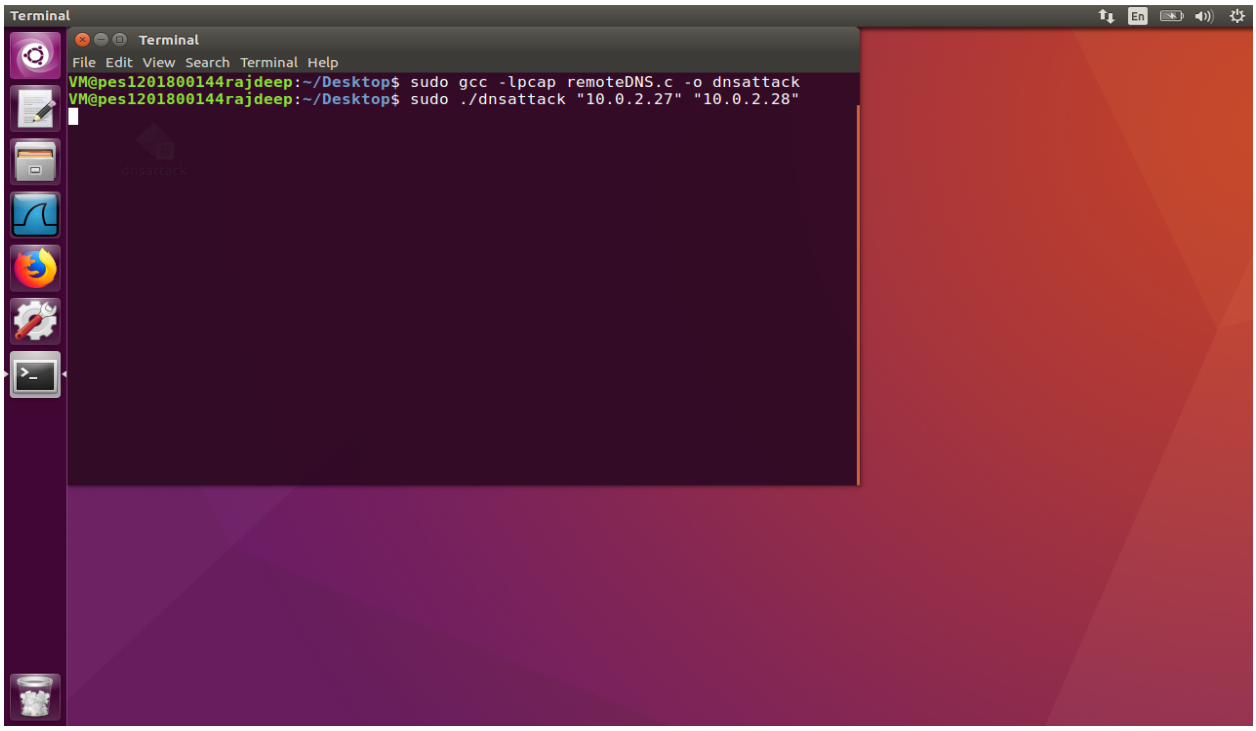
Screenshot 3.2.2: Adding the db.attacker zone file in /etc/bind directory



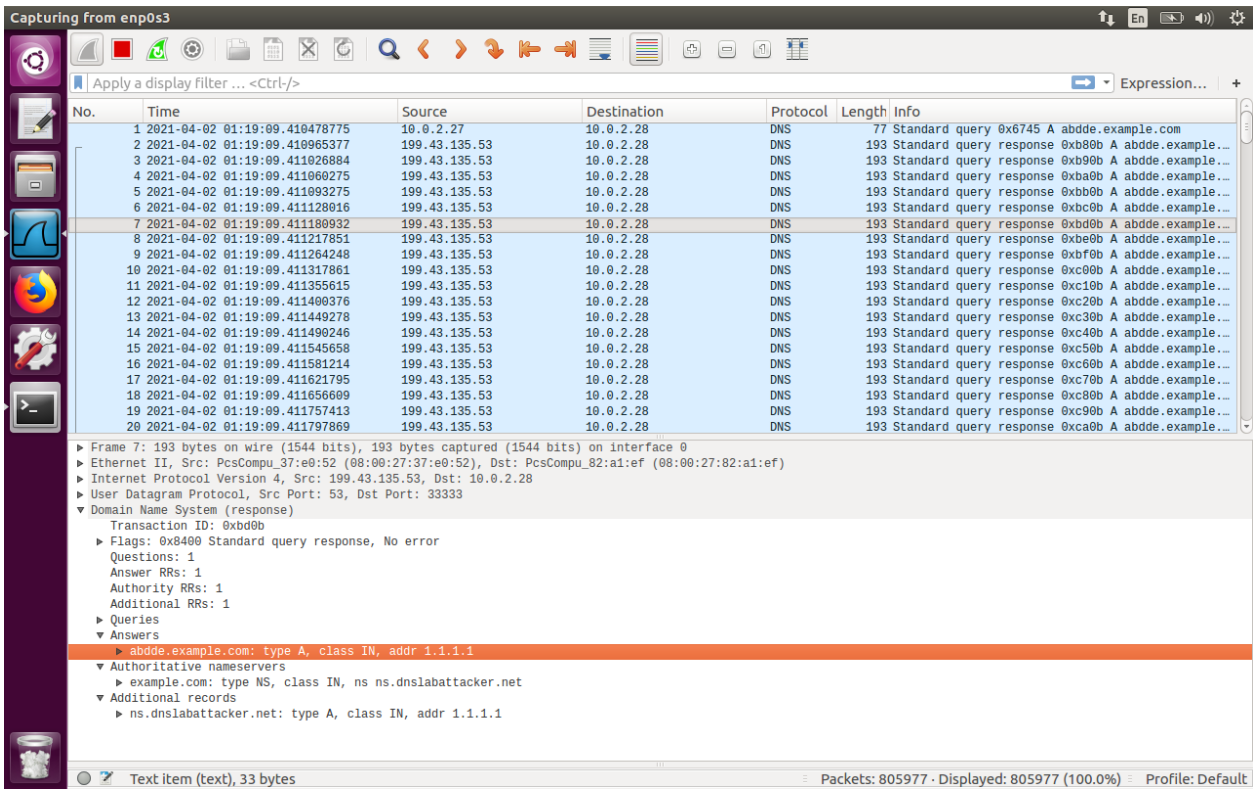
Screenshot 3.2.3: Adding zone file entry in /etc/bind/named.conf.local on attacker machine



Screenshot 3.2.4: Adding the example.com.db zone file in /etc/bind directory of attacker machine and restarting bind9 server

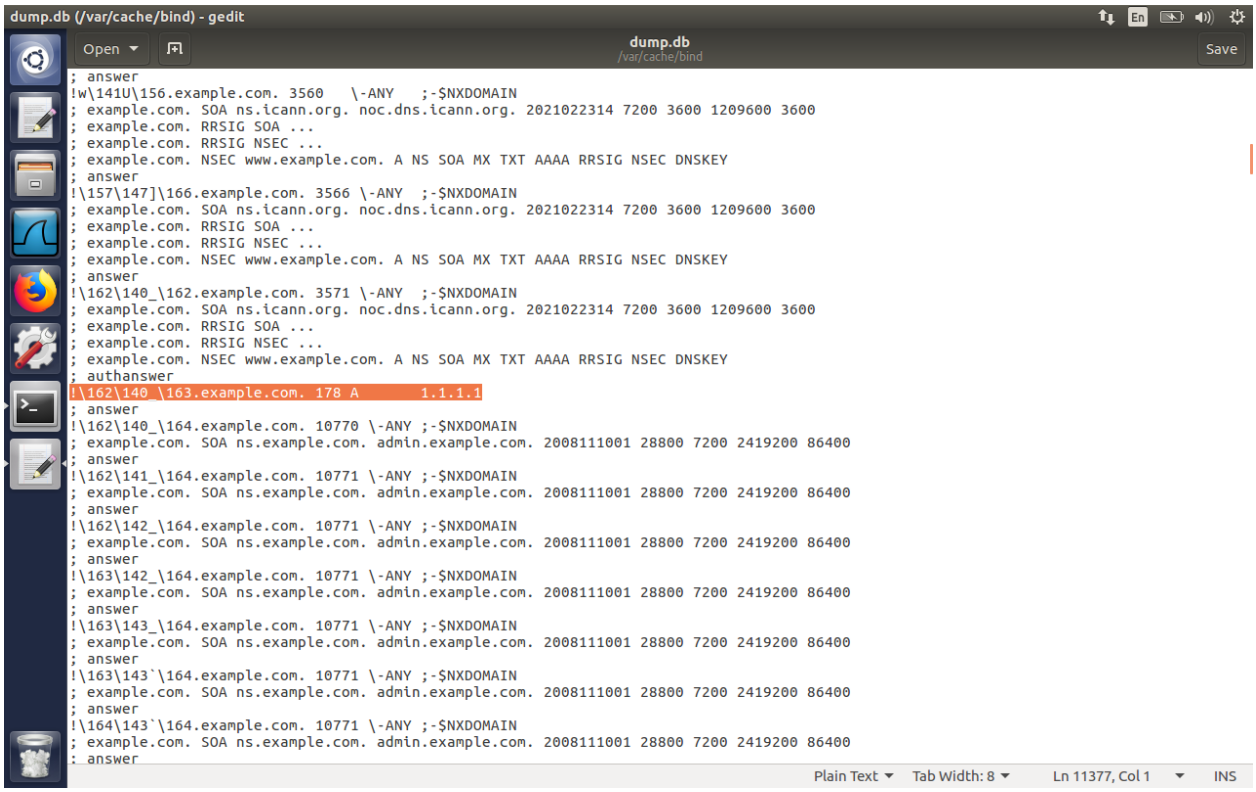


Screenshot 3.2.5: Rerunning the spoofing code

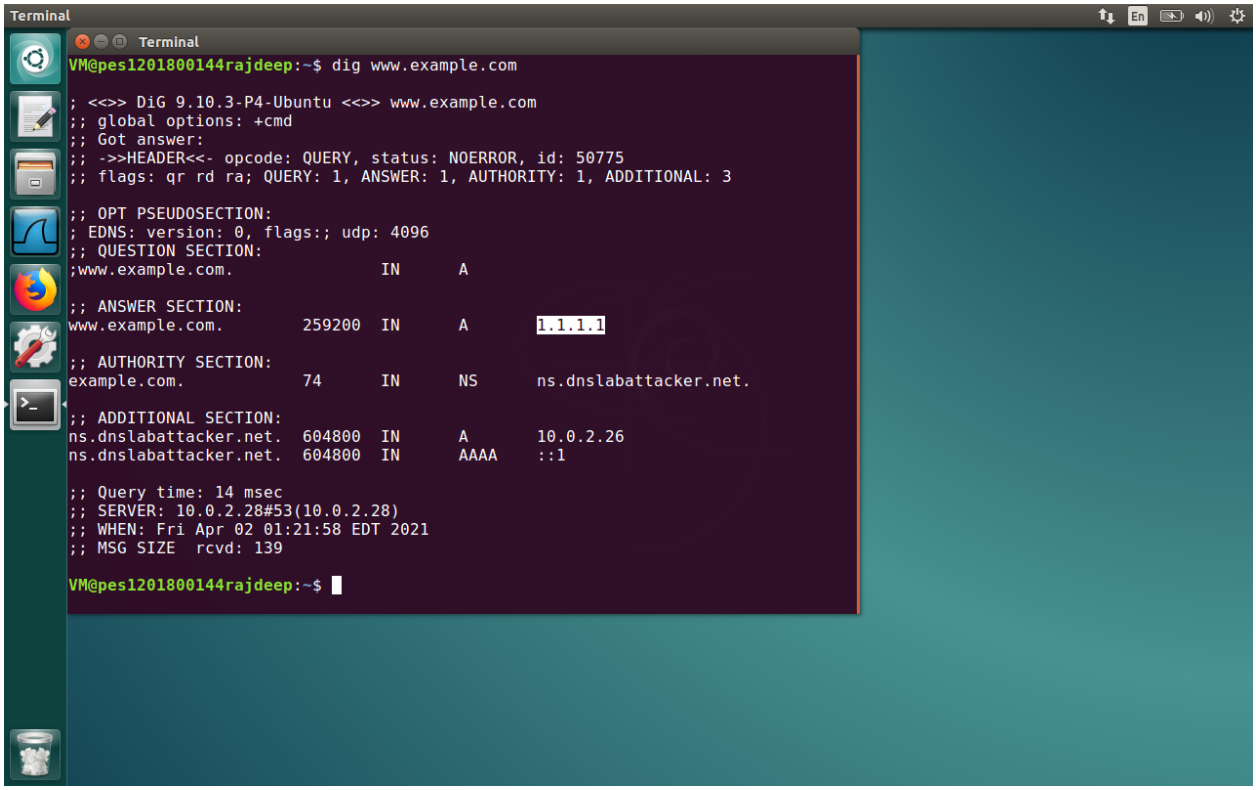


Screenshot 3.2.6: We can see Kaminsky attack is successful as in the answer section, we can find ns.dnslabattacker.net

Screenshot 3.2.7: Dump file on DNS server VM



Screenshot 3.2.8: In the dump file, example.com can be seen with IP address 1.1.1.1



Screenshot 3.2.9: Dig command replies 1.1.1.1 for www.example.com as desired

In the Kaminsky attack, when the dig command for www.example.com is sent to the DNS server VM, it searches it's cache and finds ns.dnslabattacker.net. Further, it sends a query to ns.dnslabattacker.net but gets no valid IP address. Therefore it's not a fullproof since it has a flaw which will easily be recognized and flagged by the DNS server since it doesn't have a valid IP address.

To prevent this, DNS server VM machine is configured to recognize dnslabattacker.net as real domain. This is done by creating a zone file in the server machine. By doing this, DNS server VM machine won't query the dnslabattacker.net domain and refer to the attacker machine. The domain dnslabattacker.net is looked on the attacker machine because in db.attacker file, the IP address of the attacker VM machine is given(Screenshot 3.2.2).

In the attacker VM machine, a zone file is created for example.com with the IP address 1.1.1.1 (Screenshot 3.2.4).

Hence, when the client VM machine queries dig command for www.example.com, the DNS server VM queries attacker VM which in turn returns 1.1.1.1 which is displayed in the Answer Section of dig command of client VM.