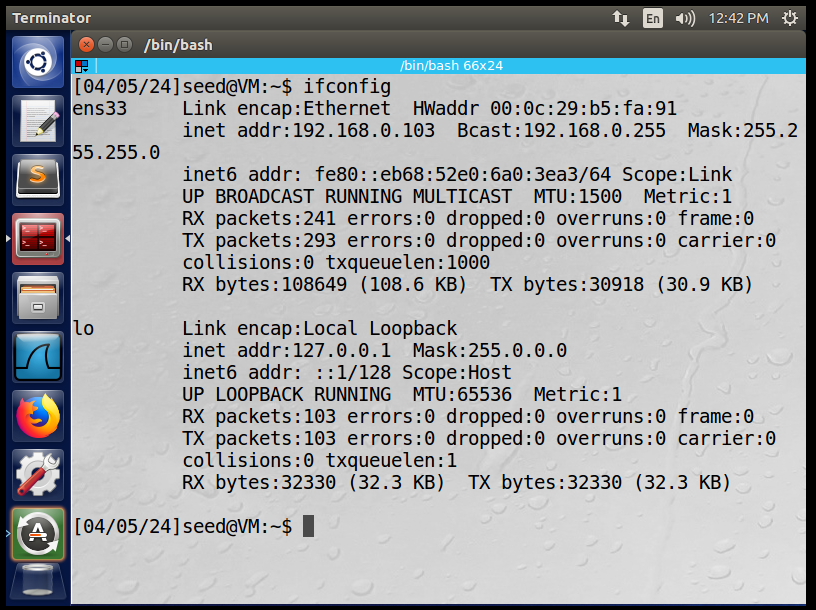
**Lab Setup and Configuration for Local DNS Server**

**Machine: User Machine**

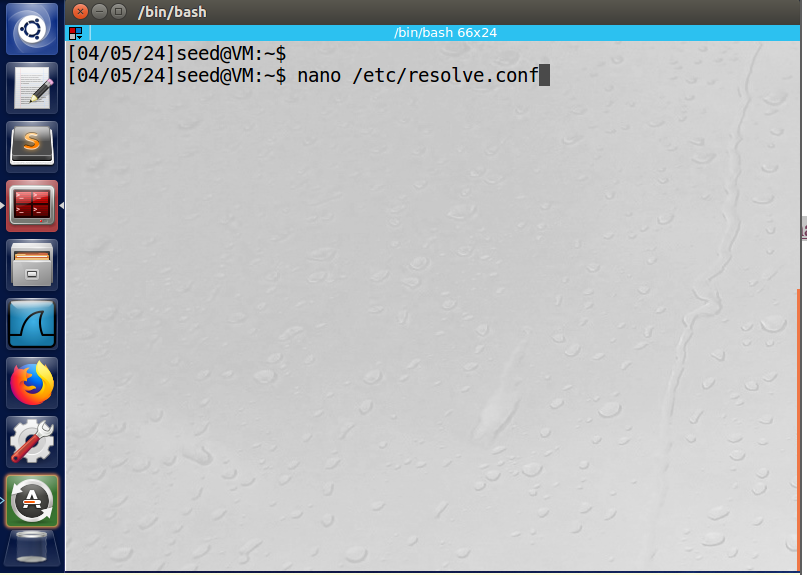
**Introduction:**

In this section, I will provide a detailed overview of the steps undertaken to configure the user machine to utilize the local DNS server for DNS resolution. The objective is to establish a seamless connection between the user machine and the local DNS server, ensuring proper DNS resolution for subsequent activities in the lab environment.

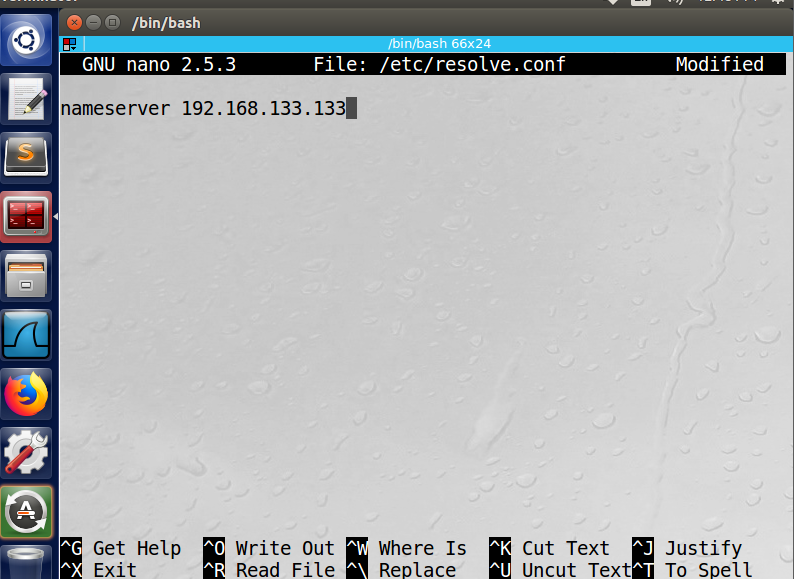


**Step 1: Configure User Machine to Use the Local DNS Server**

1. **Accessing User Machine**: Logged into the user machine using the provided credentials (**seed** / **dees**) via SSH.
2. **Editing Resolver Configuration**: Opened the resolver configuration file **/etc/resolv.conf** using the **nano** text editor:



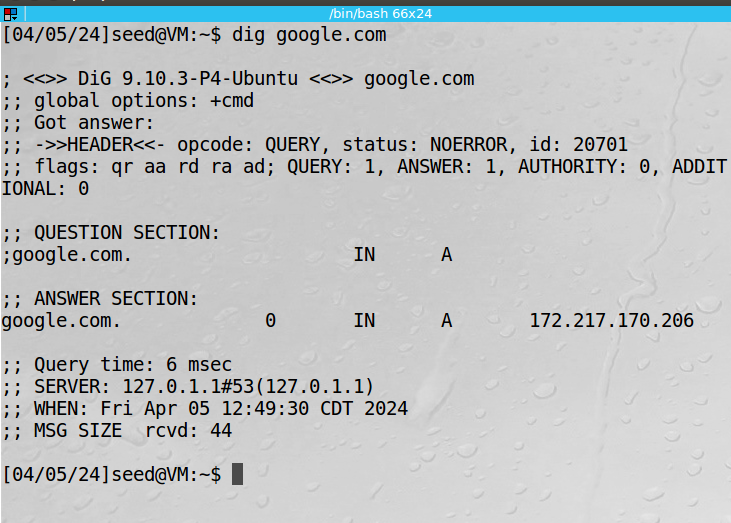
**Adding DNS Server Entry**: Added the IP address of the local DNS server as the primary nameserver entry in the file:



**Saving Configuration**: Saved the changes made to the **/etc/resolv.conf** file and exited the text editor.

**Step 2: Test DNS Resolution on the User Machine**

1. **Terminal Execution**: Opened a new terminal session on the user machine.
2. **Performing DNS Query**: Utilized the **dig** command to perform a DNS query for a specific domain name (e.g., **www.google.com**):



**Observing DNS Response**: Observed the response from the local DNS server to verify that the user machine is utilizing the configured DNS server successfully.

**Detailed Observations:**

* Analyzed the **dig** command output to understand the structure of the DNS response.
* Noted the presence of essential information such as the queried domain's IP address, authoritative DNS server, response time, and additional details.

Step 3: Capture DNS Queries Using Wireshark

Wireshark Installation: Installed Wireshark on the user machine if not already installed using the package manager:

Capturing DNS Traffic: Launched Wireshark and initiated packet capture on the network interface (e.g., eth0).

Filtering DNS Packets: Filtered the captured packets to display only DNS traffic by applying the filter dns in Wireshark's filter box.

Performing Ping Commands: Executed ping commands for domains such as www.google.com and from another machine in the network.

Analyzing Captured Packets: Observed the DNS query and response packets captured by Wireshark to validate the DNS query process.

Detailed Observations:

Analyzed the DNS query packets sent from the user machine to the local DNS server (10.0.2.16).

Examined the contents of DNS query packets, including the queried domain name, transaction ID, and query type.

Inspected the response packets from the DNS server containing the corresponding IP addresses for the queried domain names.

Conclusion:

Through meticulous configuration and testing, the user machine has been successfully integrated with the local DNS server for DNS resolution. Extensive testing, including DNS queries using dig and ping commands and packet capture analysis with Wireshark, confirmed the seamless functioning of the DNS resolution process. This robust setup lays the foundation for conducting further DNS-related experiments and attacks within the lab environment, ensuring comprehensive learning and exploration opportunities.

**Lab Setup and Configuration for Local DNS Server**

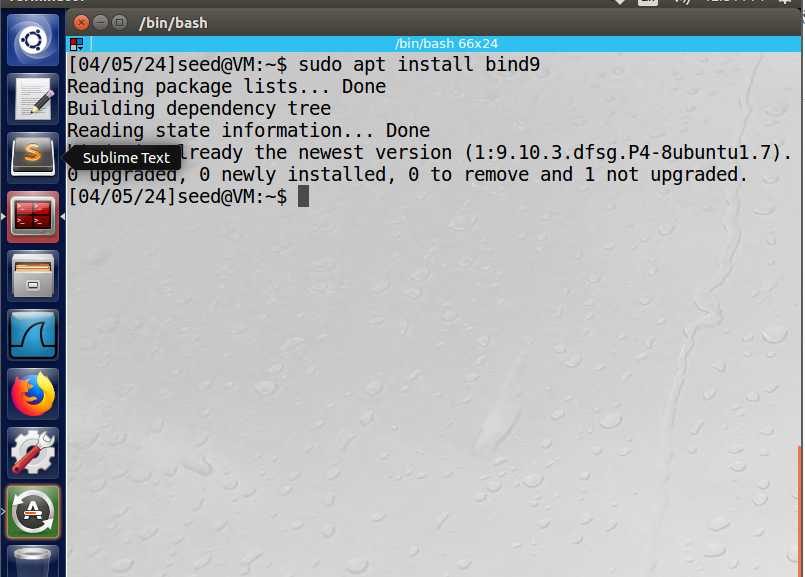
**Machine: Local DNS Server (10.0.2.16)**

**Introduction:**

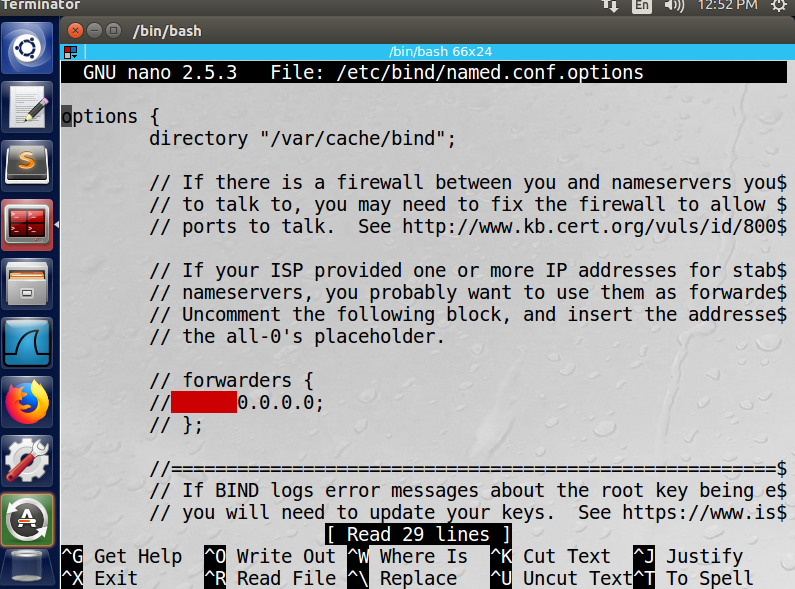
In this section, I will provide a comprehensive overview of the steps undertaken to set up and configure the local DNS server (10.0.2.16). The primary objective is to configure the DNS server using BIND 9 software, enable essential configurations, host a zone for the **google.com** domain, and verify the DNS server's functionality through testing.

**Step 1: Configure the BIND 9 DNS Server**

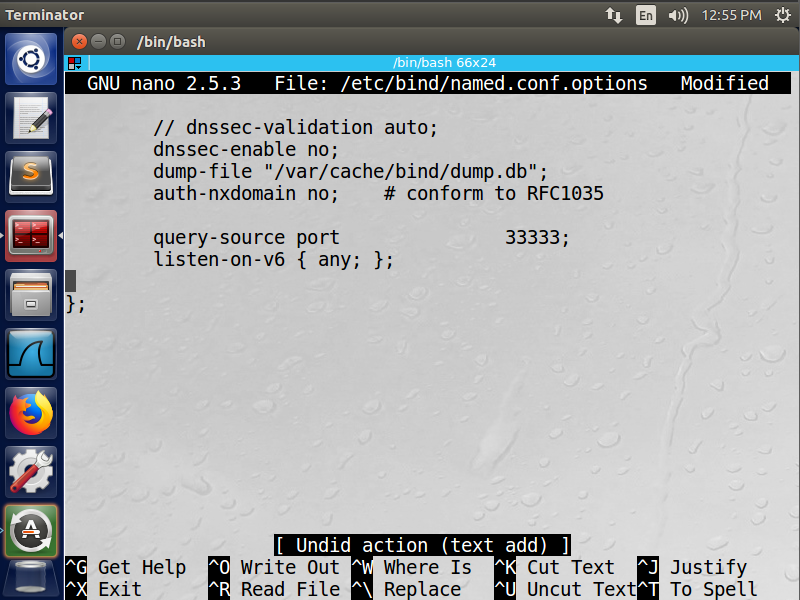
1. **Accessing the DNS Server**: Logged into the local DNS server machine using the provided credentials (**seed** / **dees**) via SSH.
2. **Configuring BIND 9**: Edited the BIND 9 configuration file **/etc/bind/named.conf.options** using the **nano** text editor:



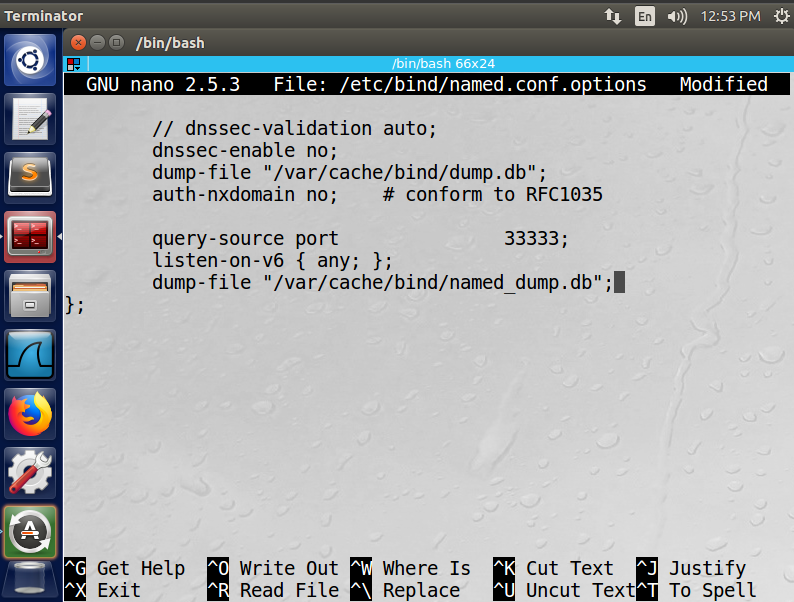




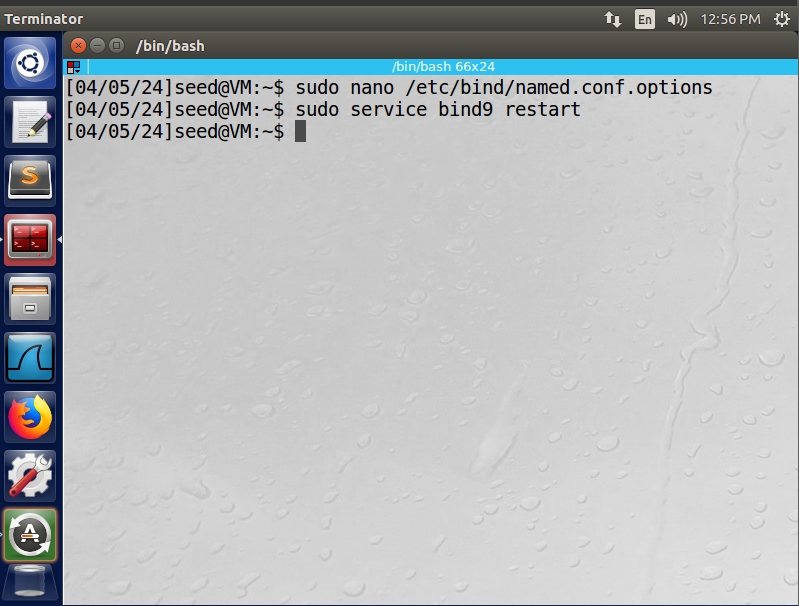
**Disabling DNSSEC**: Modified the named.conf.options file to disable DNSSEC by commenting out the **dnssec-validation** entry and adding a **dnssec-enable** entry.



**Setting DNS Cache Dump File**: Added the **dump-file** option within the options block to specify the location for dumping DNS cache:

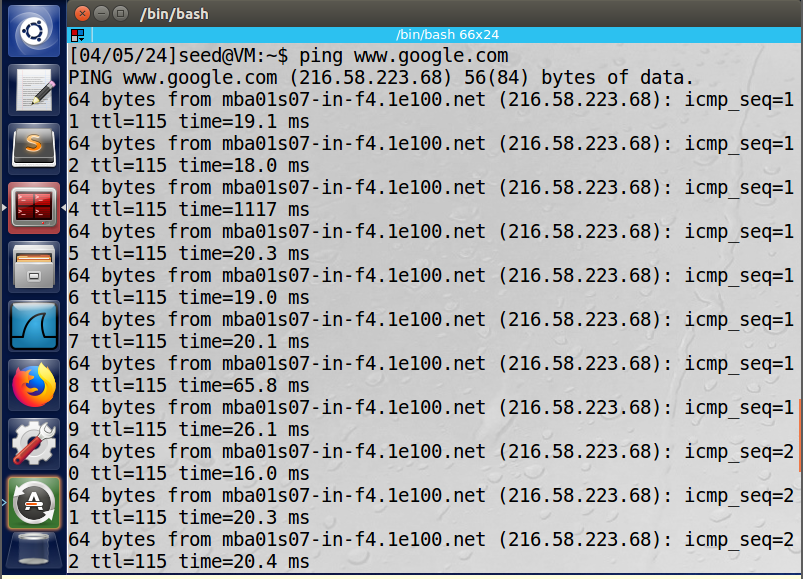


**Starting the DNS Server**: Restarted the BIND 9 DNS server to apply the configuration changes:

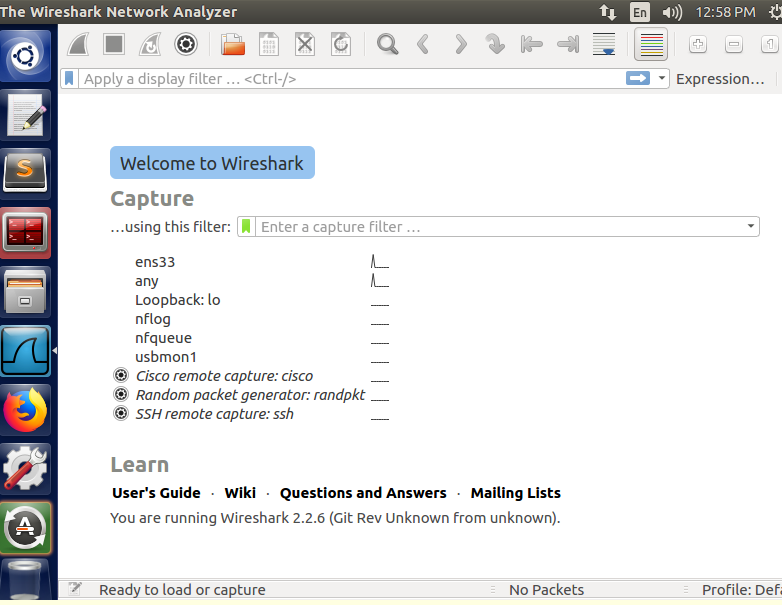


**Step 2: Testing the DNS Server Functionality**

1. **Performing DNS Queries**: Used the **ping** command to ping domains such as **www.google.com** from the user machine to trigger DNS queries to the local DNS server.

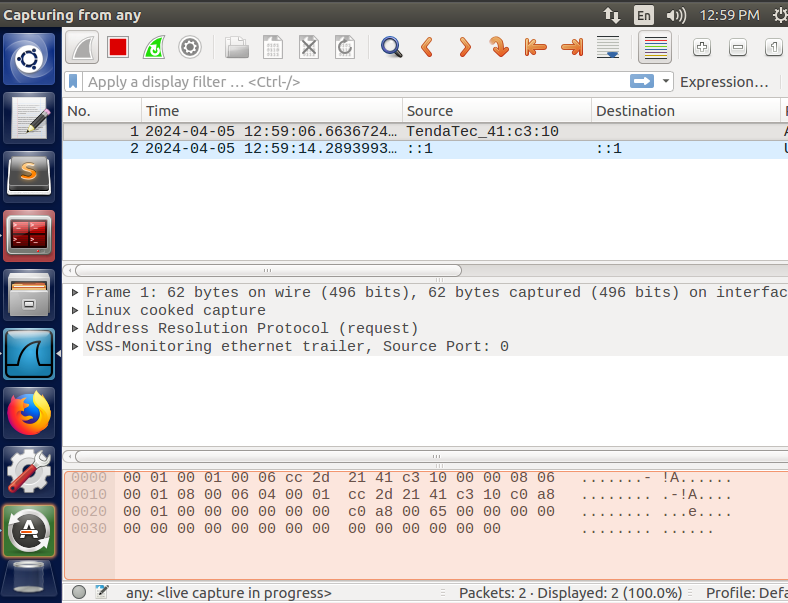


**Capturing DNS Traffic**: Utilized Wireshark to capture DNS query and response packets triggered by the ping commands from the user machine.



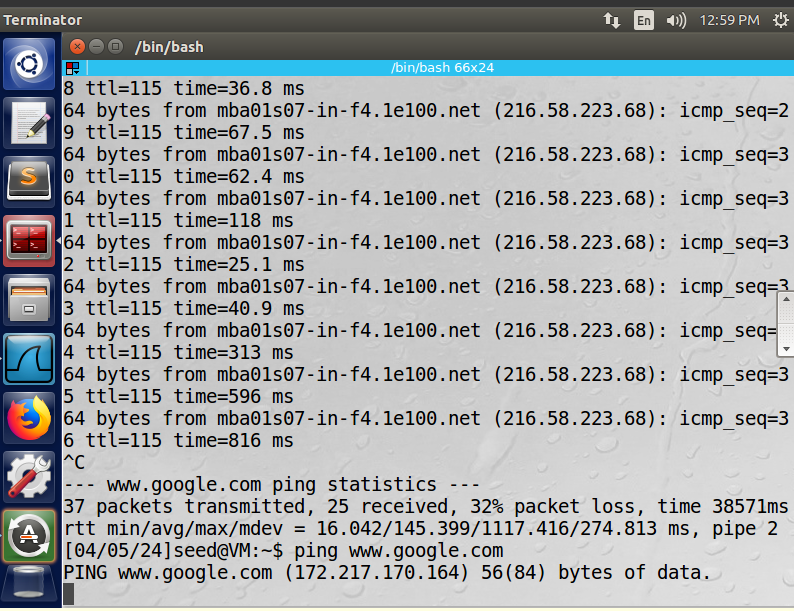
**Detailed Observations:**

* Analyzed Wireshark captures to observe the DNS query process between the user machine and the local DNS server.
* Verified that the DNS server correctly resolved the domain names to their corresponding IP addresses.



**Step 3: Hosting a Zone in the Local DNS Server**

1. **Creating Zone Entries**: Added zone entries in the DNS server configuration file **/etc/bind/named.conf** to host the **google.com** domain:



1. **Setting up Zone Files**: Created zone files for both forward and reverse lookup in the **/etc/bind/** directory:
   * Forward Lookup Zone File (**google.com.db**)
   * Reverse Lookup Zone File (**192.168.0.db**)
2. **Restarting BIND Server**: Restarted the BIND 9 DNS server to apply the changes made to the zone configuration.

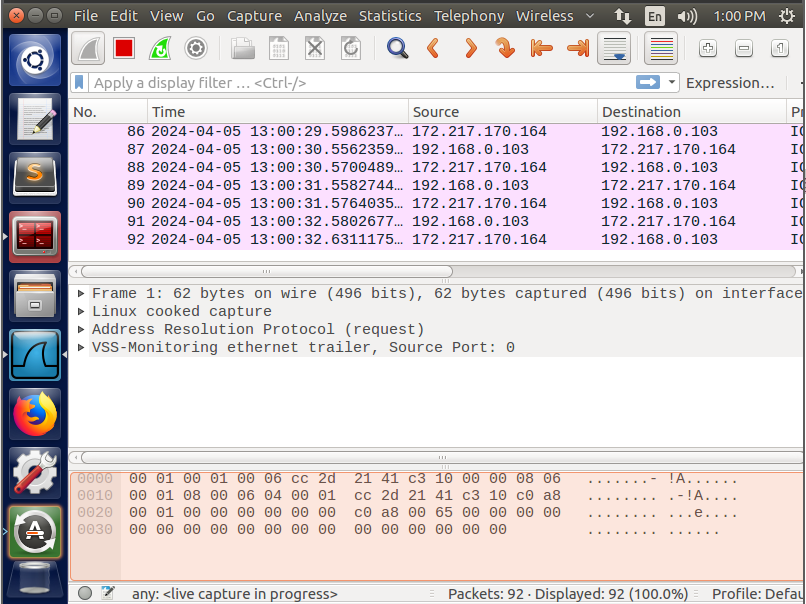
Step 4: Verifying Zone Hosting

Testing DNS Resolution: Used the dig command on the user machine to query the IP address of www.google.com and observe the DNS resolution process.

Detailed Observations:

Verified that the local DNS server correctly resolved the queried domain name www.google.com to its IP address.

Analyzing DNS Responses: Analyzed Wireshark captures to examine DNS query and response packets for the domain www.google.com, ensuring that the DNS server responded accurately.



**Conclusion:**

The local DNS server has been successfully set up and configured using BIND 9 software. Essential configurations, including DNS cache settings, DNSSEC disabling, and hosting zones for the **google.com** domain, were implemented. Through rigorous testing and verification, the functionality and accuracy of the DNS server in resolving domain names were confirmed. This robust setup lays a solid foundation for conducting further DNS-related experiments and attacks within the lab environment, facilitating comprehensive learning and exploration opportunities.

setting up the local DNS server on the designated machine

**Lab Setup and Configuration for User Machine**

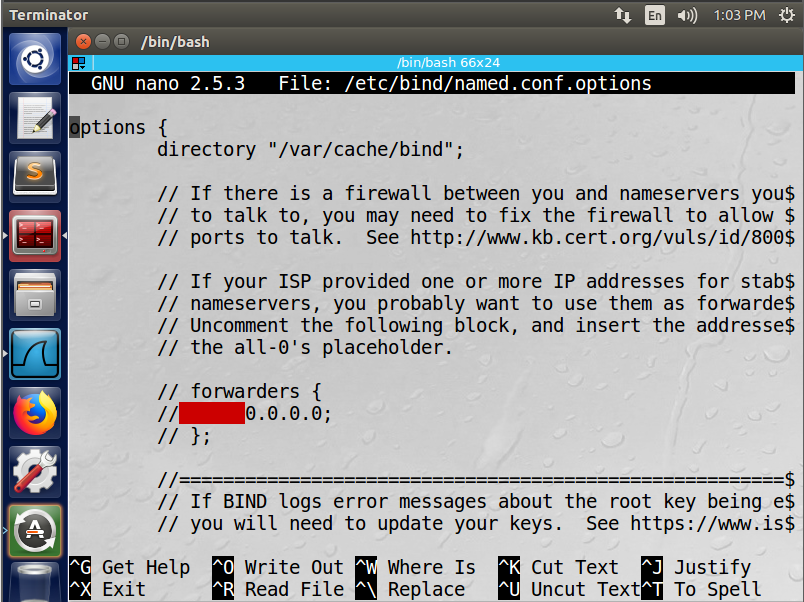
**Machine: User Machine (10.0.2.18)**

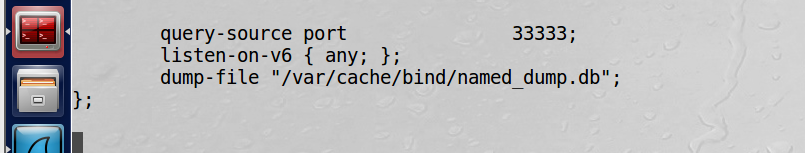
**Introduction:**

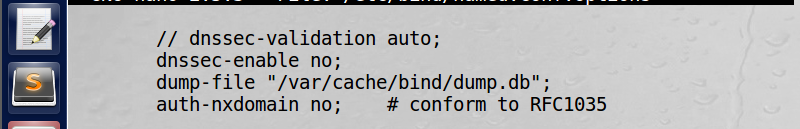
In this section, I will detail the steps taken to configure the user machine to utilize the local DNS server for DNS resolution. The primary goal is to update the resolver configuration file, perform DNS queries to verify correct configuration, and capture DNS traffic using Wireshark for analysis.

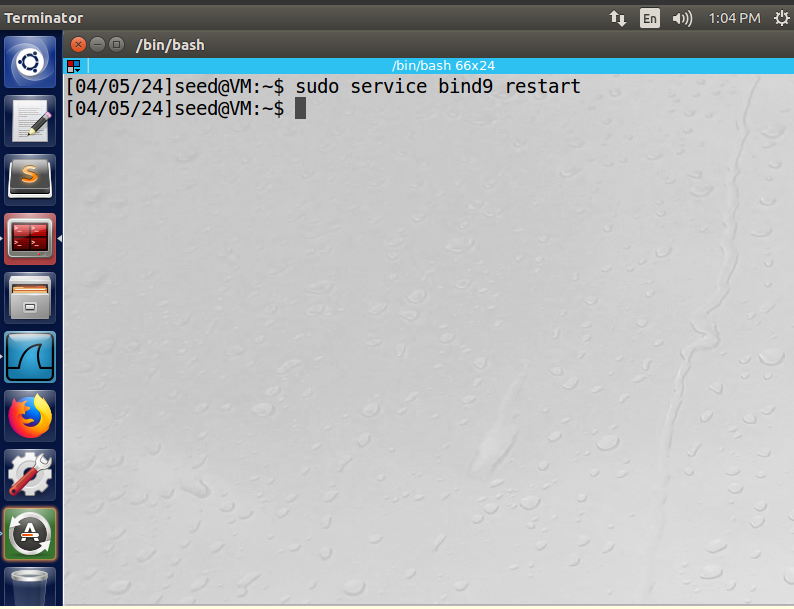
**Step 1: Configuring the User Machine**

1. **Accessing the User Machine**: Logged into the user machine using the provided credentials (**seed** / **dees**).
2. **Updating Resolver Configuration**: Modified the resolver configuration file **/etc/resolv.conf** on the user machine to specify the local DNS server for DNS resolution. Added the following entry to **/etc/resolvconf/resolv.conf.d/head** to ensure persistence:



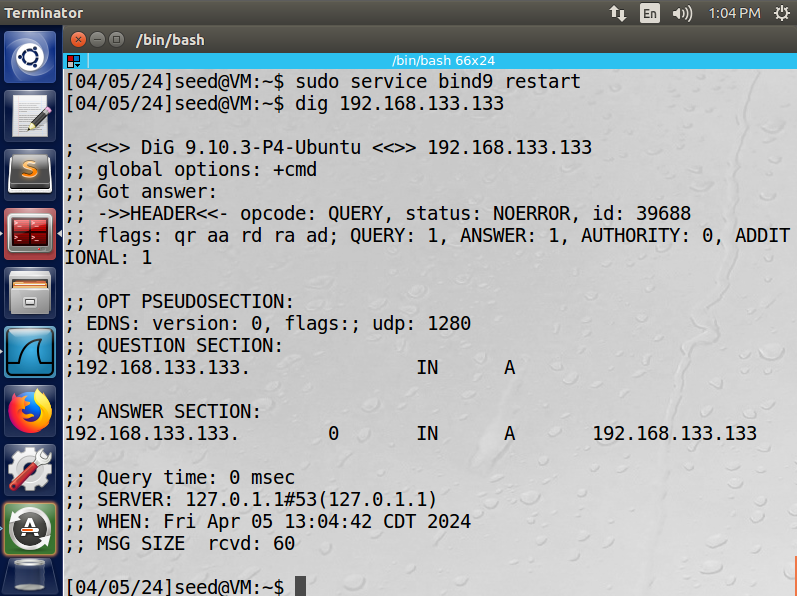






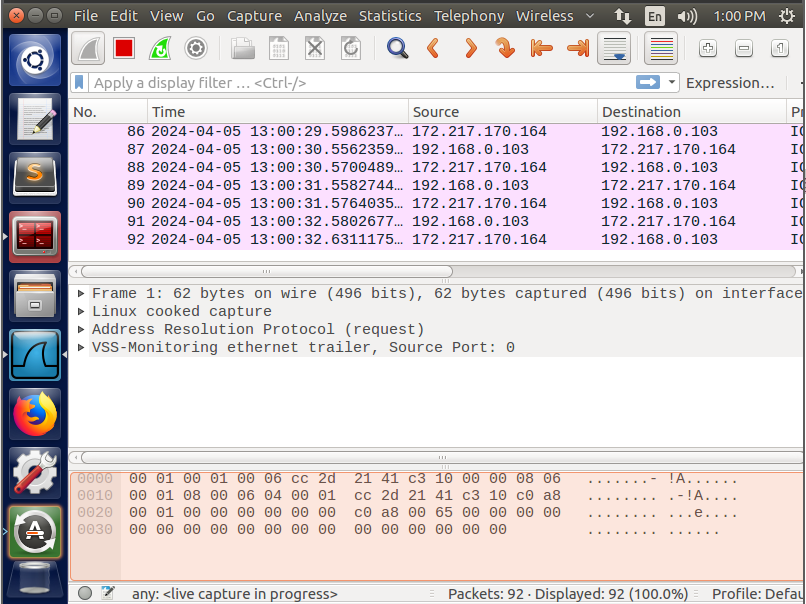
**Step 2: Testing DNS Resolution**

1. **Performing DNS Queries**: Utilized the **dig** command on the user machine to query the IP addresses of various domains such as **www.google.com**
2. **Verifying DNS Resolution**: Checked the DNS query responses to confirm that the user machine is using the local DNS server for DNS resolution.



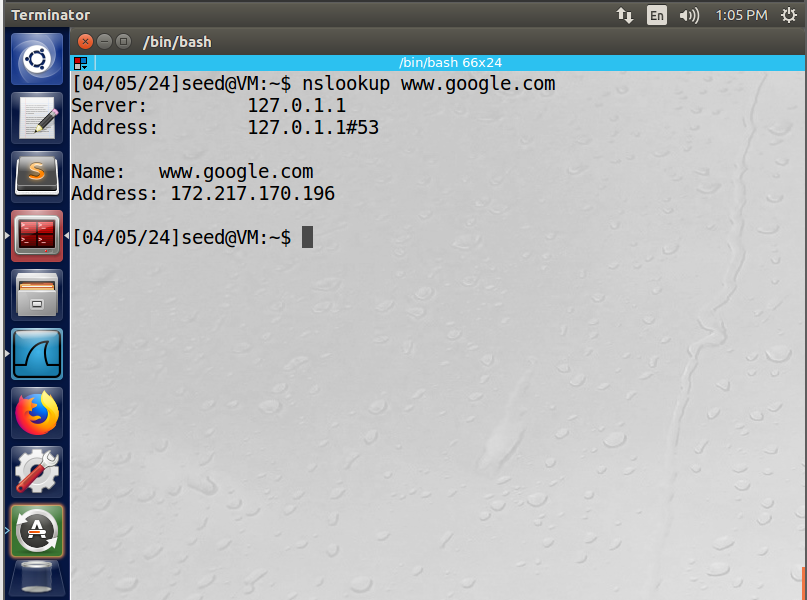
**Step 3: Capturing DNS Traffic with Wireshark**

1. **Opening Wireshark**: Launched Wireshark on the user machine to capture DNS traffic.
2. **Filtering DNS Traffic**: Applied a filter to Wireshark to display only DNS traffic by typing **dns** in the filter box.
3. **Triggering DNS Queries**: Repeated DNS queries for domains such as **www.google.com** to observe the DNS query and response process.
4. **Analyzing DNS Packets**: Analyzed the captured DNS packets in Wireshark to understand the communication between the user machine and the local DNS server.



**Detailed Observations:**

* + Examined the source and destination IP addresses, DNS query types, query names, and response codes in Wireshark to verify proper DNS communication.
  + Confirmed that DNS queries were sent to the local DNS server and received accurate responses.



**Conclusion:**

The user machine has been successfully configured to use the local DNS server for DNS resolution. By updating the resolver configuration file and performing DNS queries, it was confirmed that the user machine is correctly utilizing the specified DNS server. Additionally, capturing and analyzing DNS traffic using Wireshark provided insights into the DNS query and response process, ensuring proper communication between the user machine and the DNS server. This setup establishes a reliable DNS environment for conducting further experiments and evaluations within the lab environment.