SLIIT

COMPUTING | BUSINESS | ENGINEERING

IE3020
Directory Structure and Network Management
3rd Year, 2nd Semester

Assignment

Implementation of Linux-based Domain Controller with Active Directory and Zabbix Monitoring

Submitted to

Sri Lanka Institute of Information Technology

In partial fulfilment of the requirements for the

Bachelor of Science Special Honors Degree in Information Technology

07/05/2025

Declaration

I certify that this report does not incorporate without acknowledgement, any material previously submitted for a degree or diploma in any university, and to the best of my knowledge and belief, it does not contain any material previously published or written by another person, except where due reference is made in text.

Registration Number: **IT22310132**

Name: **Muthukumarana T D**

Table of Contents

Table of Figures

Introduction

In this project, **Fedora 42** is employed as the operating system for establishing the Active Directory Domain Controller (AD-DC) and Zabbix Monitoring Server. In contrast, **Ubuntu** is utilized for the client machines. The selection of Fedora for server roles was driven by its current package updates, robust SELinux security integration, and comprehensive systemd support, which renders it suitable for running critical services such as Samba (for AD) and Zabbix. Ubuntu was chosen for client machines due to its user-friendly interface, extensive community support, and compatibility with Linux-based enterprise environments. This configuration facilitates centralized authentication and robust infrastructure monitoring across the network.

1. Setting up the Domain Controller

1.1. Preparing the System

- Update the Fedora 42 operating system

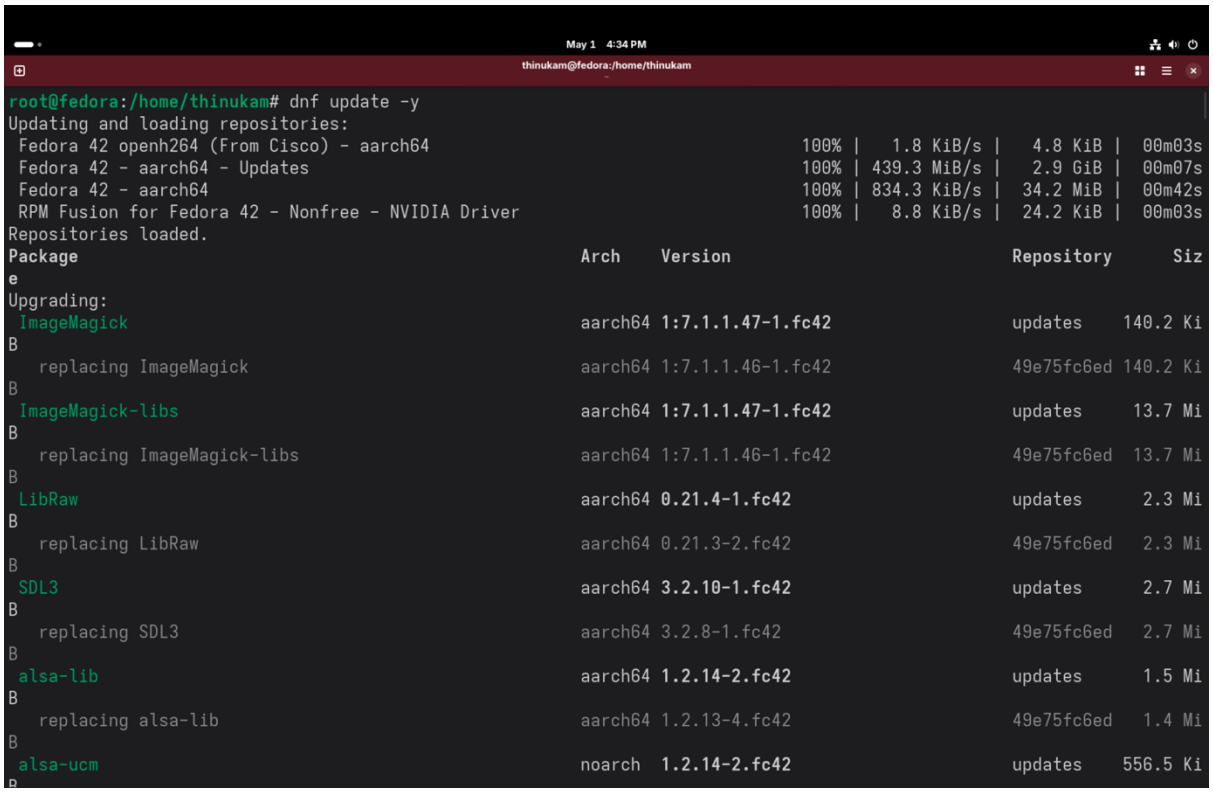

Figure 1.1. 1 - Update Fedora OS

- Install the required packages to run Samba Active Directory Domain Controller services

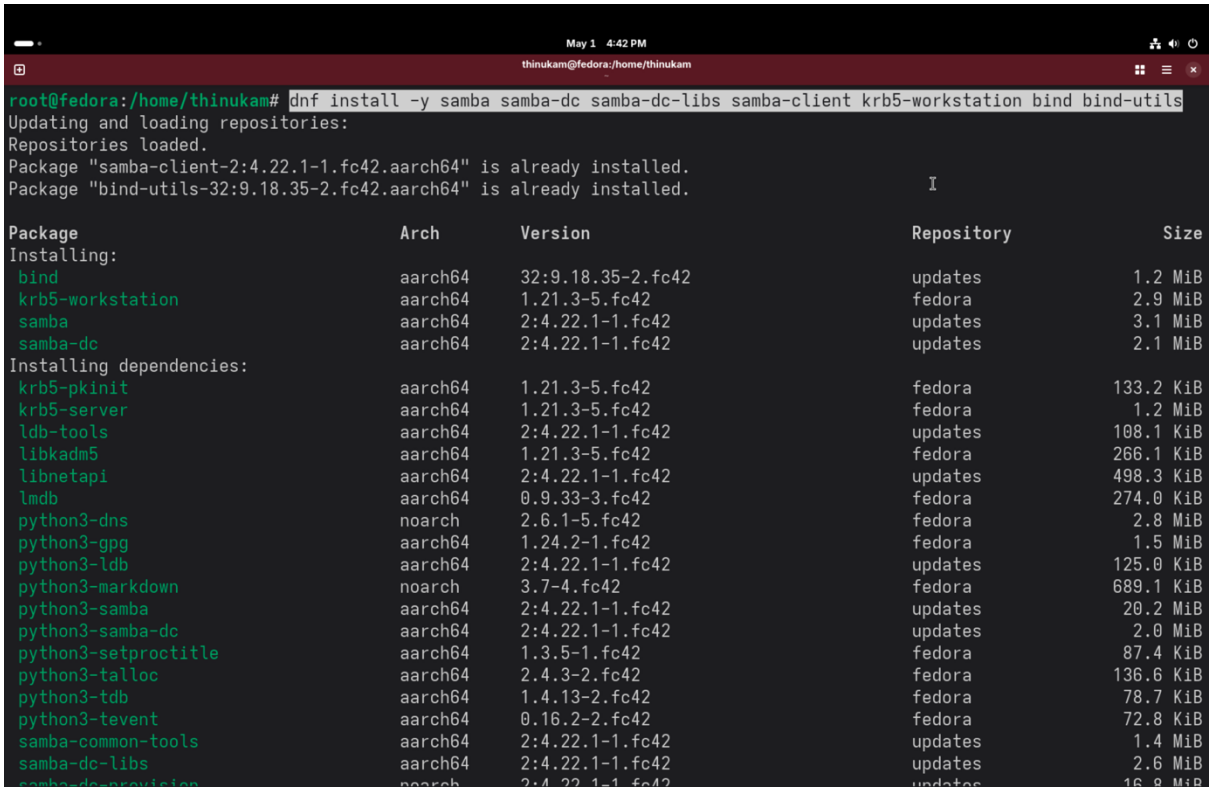

Figure 1.1. 2 - Install required packages for Samba AD-DC

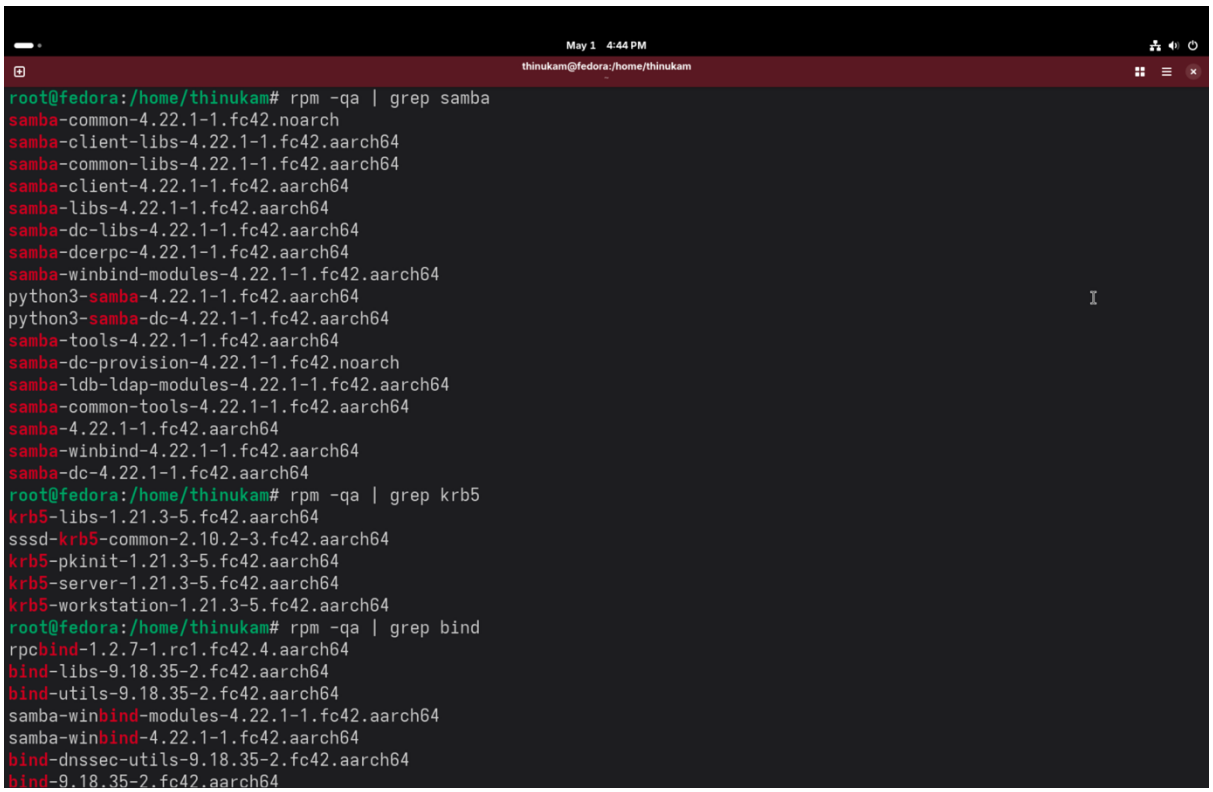

Figure 1.1. 3 - Verify packages

- Set a static IP address and a hostname for the Domain Controller system

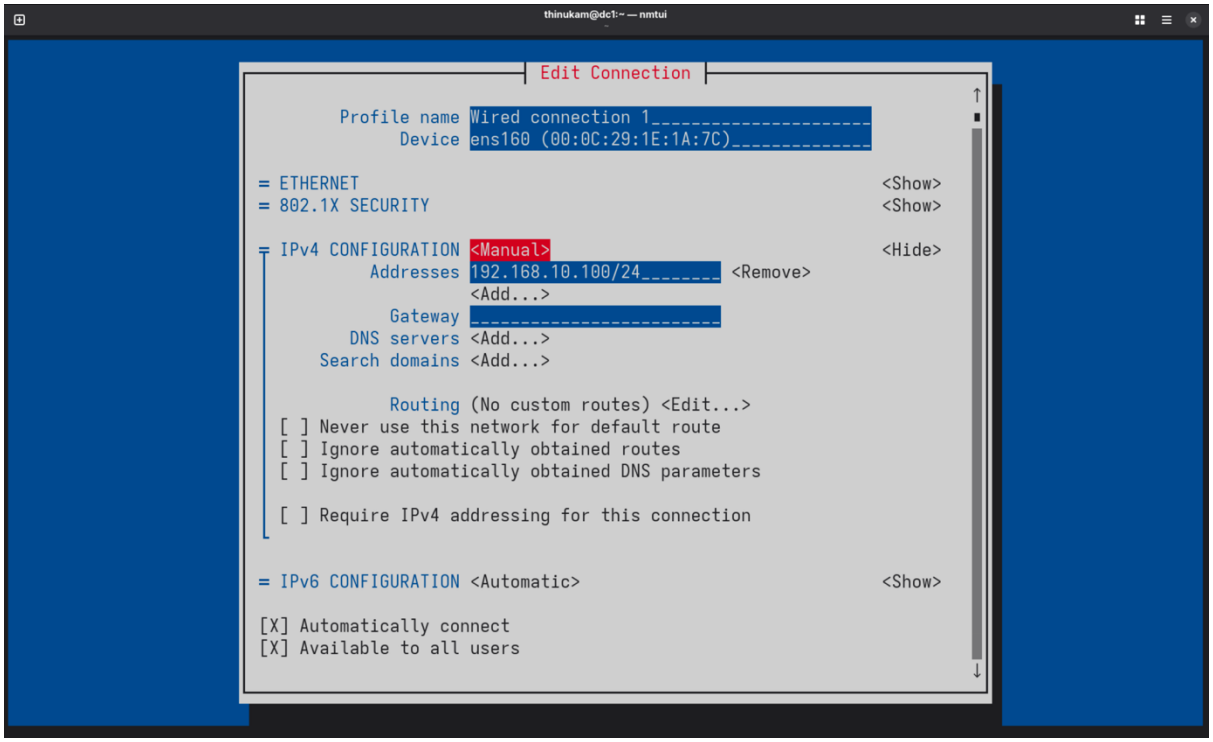

Figure 1.1. 4 - Set a Static IP via nmtui

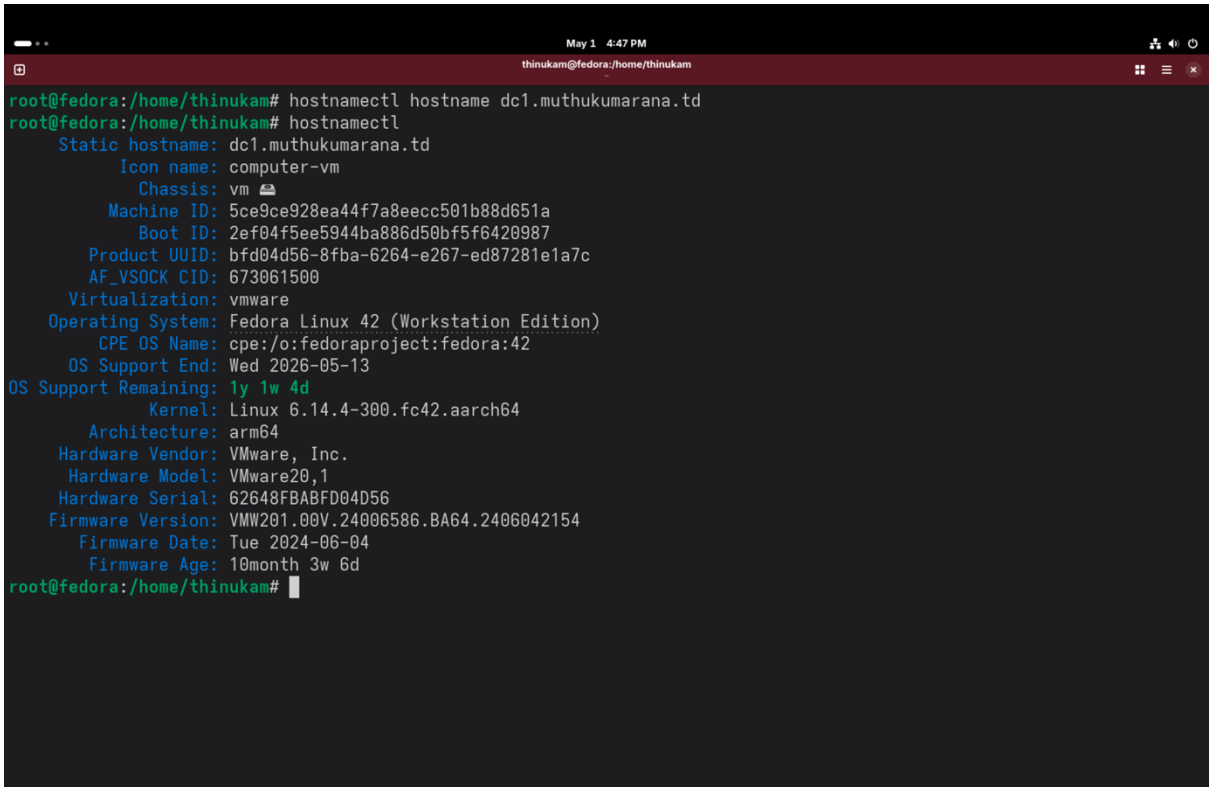

Figure 1.1. 5 - Set a static hostname

- Add the static IP address and the hostname to the */etc/hosts* file for name resolution

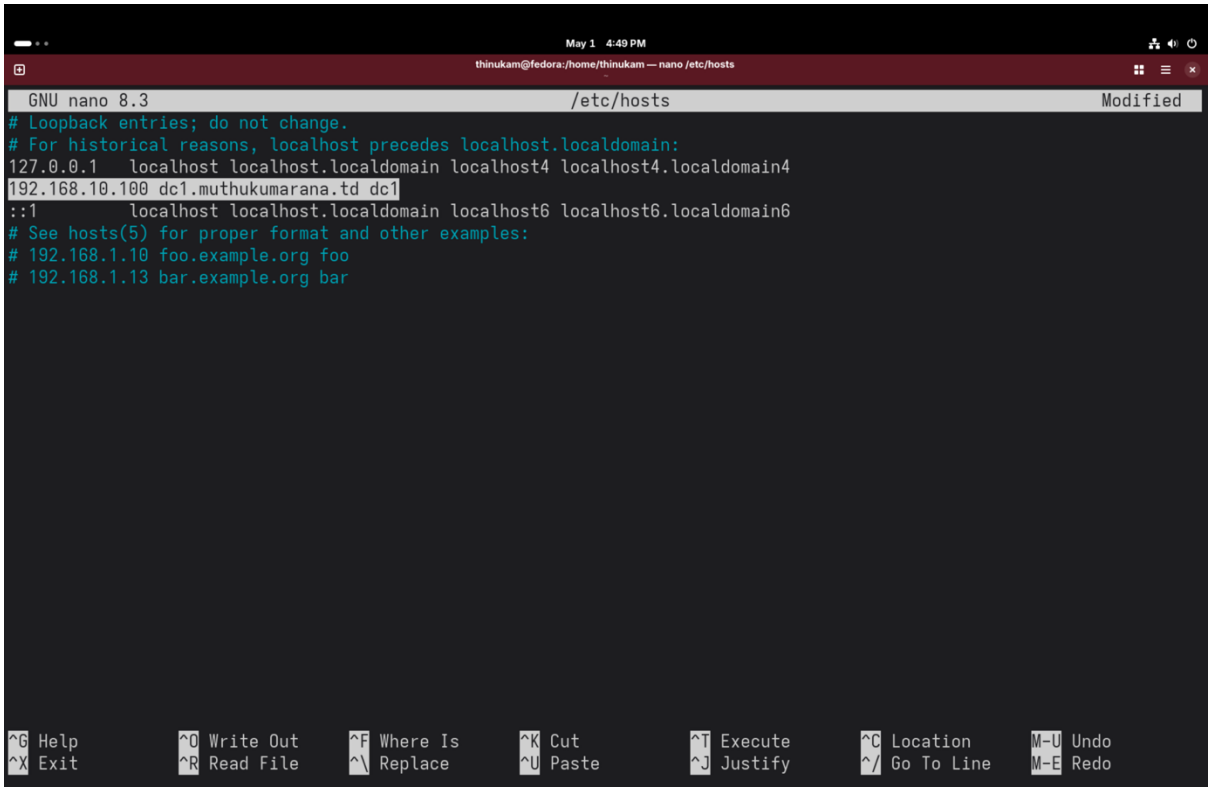

Figure 1.1. 6 - /etc/hosts file

1.2. Configuring Samba

- Set Samba Booleans for SELinux to run Samba DC with SELinux in enforcing mode.

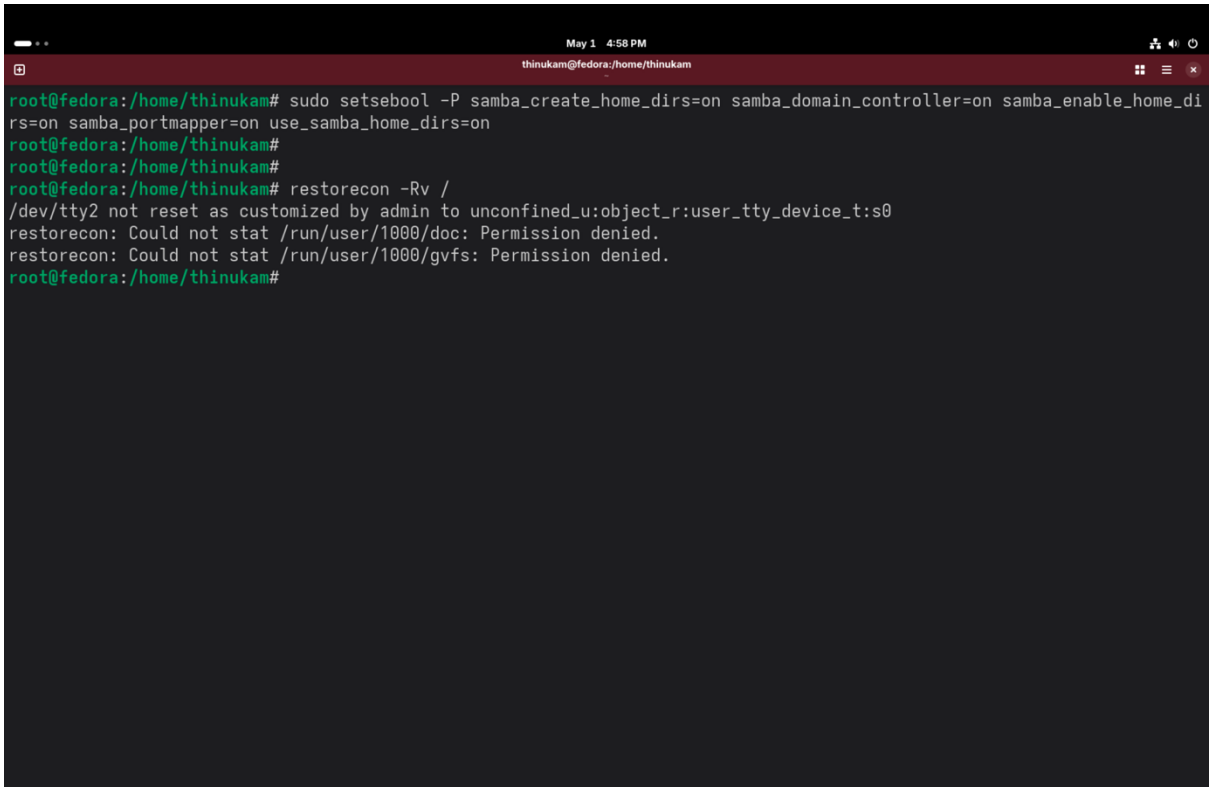

Figure 1.2. 1 - Set Booleans for SELinux

- Move the default Samba config file and rename it as *smb.conf.bak*

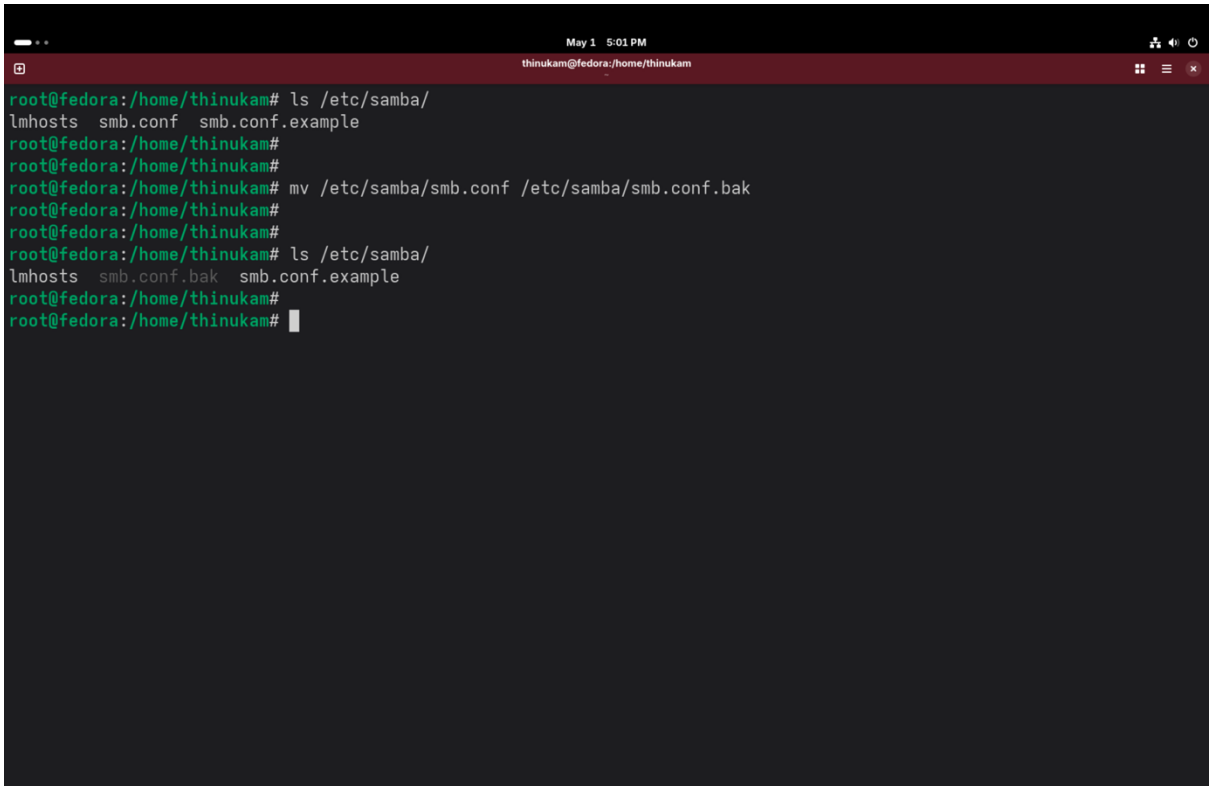

Figure 1.2. 2 - Move the default Samba config file

- Create the directory */etc/systemd/resolved.conf.d/* and create a custom DNS resolution config file in the directory, and restart the *systemd-resolved* service

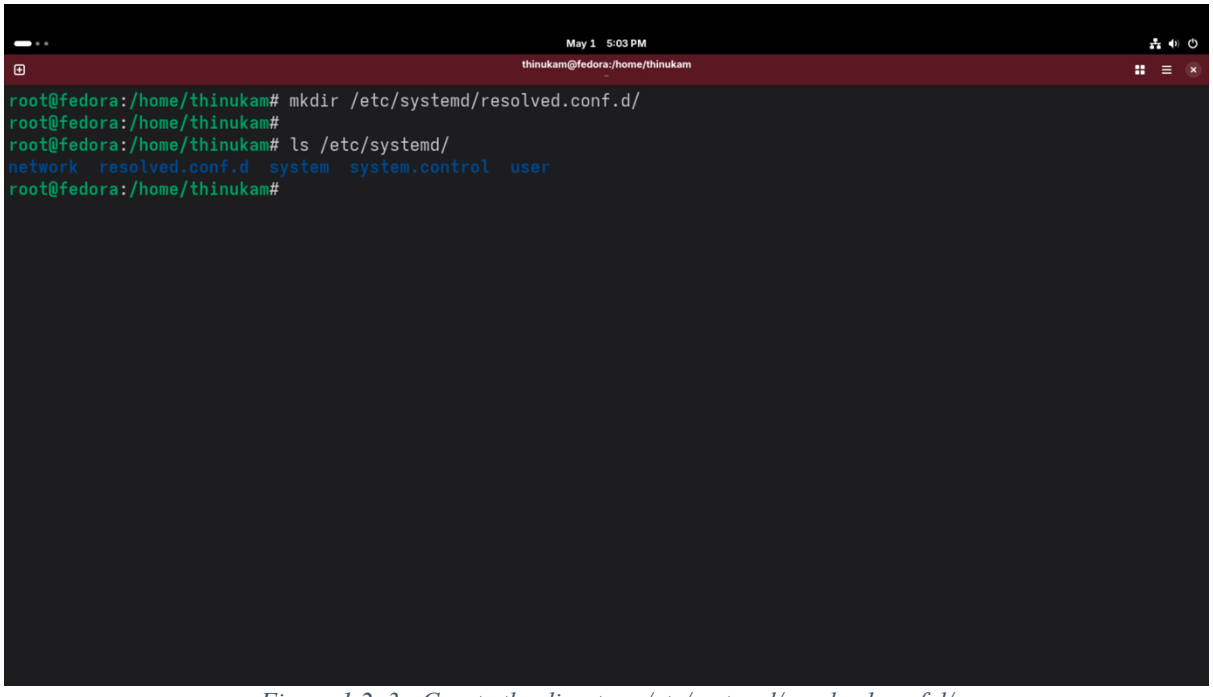

Figure 1.2. 3 - Create the directory /etc/systemd/resolved.conf.d/

- Use the *nano /etc/systemd/resolved.conf.d/custom.conf* command to create the custom file

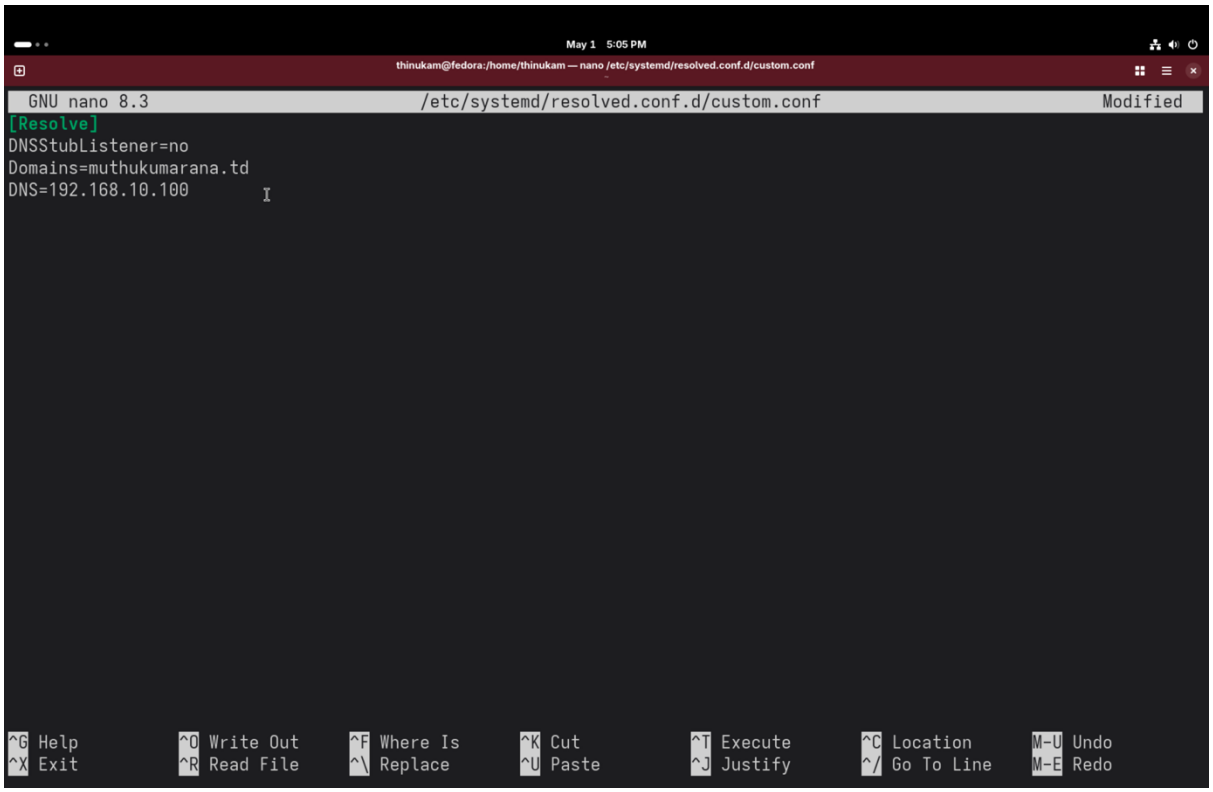

Figure 1.2. 4 - Custom DNS resolution file

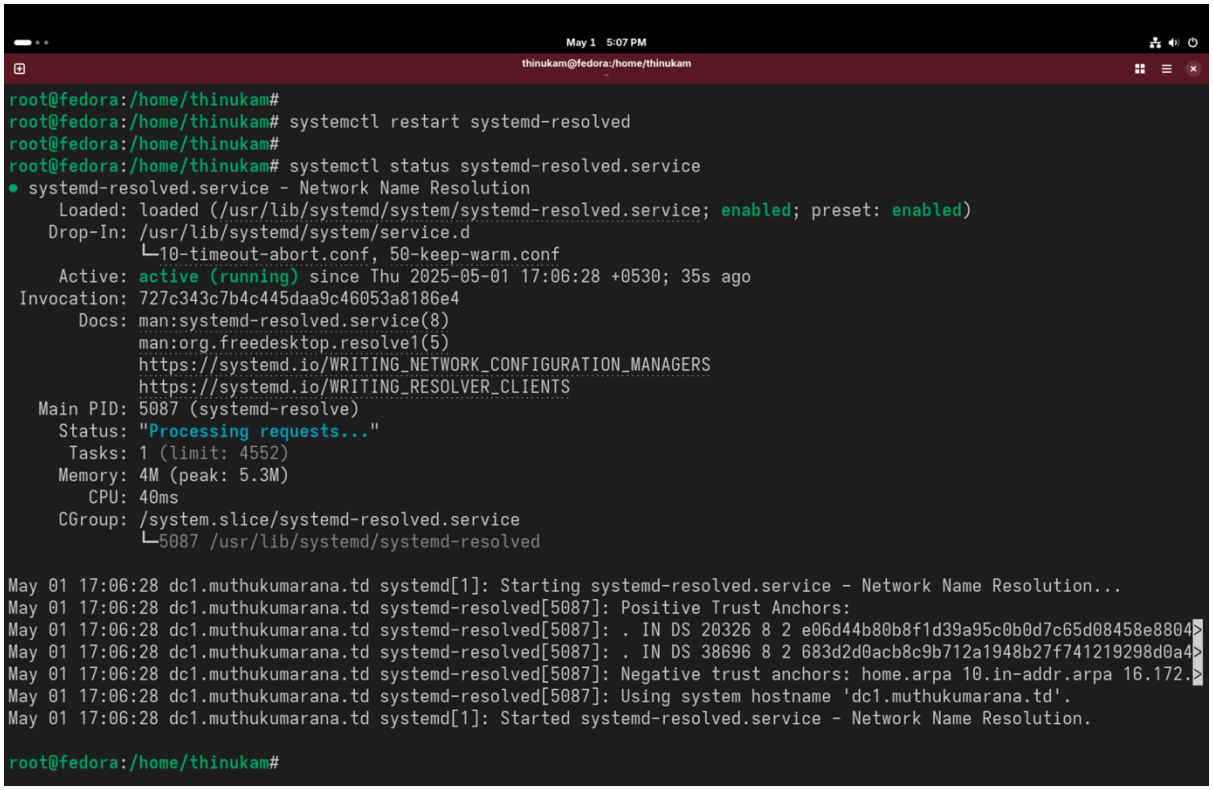

Figure 1.2. 5 - Restart the systemd-resolved service

- Using the *samba-tool,* provision the Samba configuration and verify the newly created *smb.conf* Samba configuration file

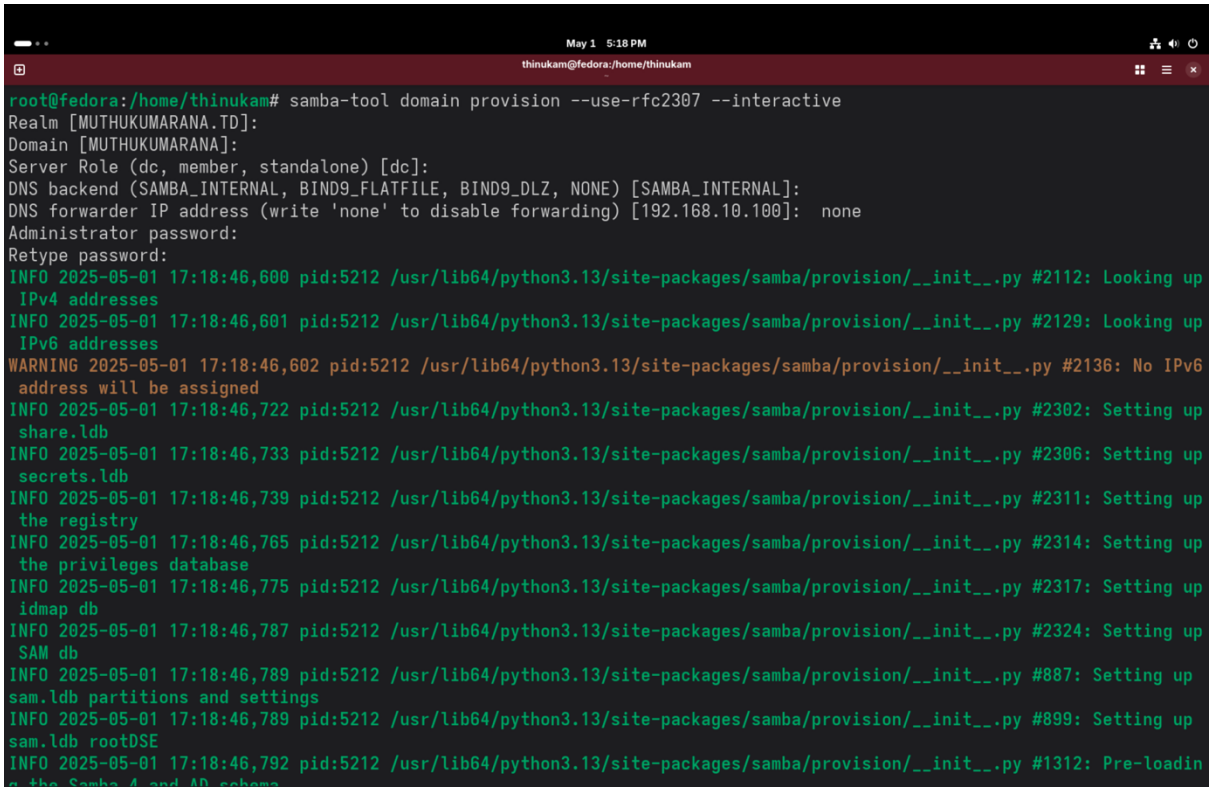

Figure 1.2. 6 - Provision the Samba configuration

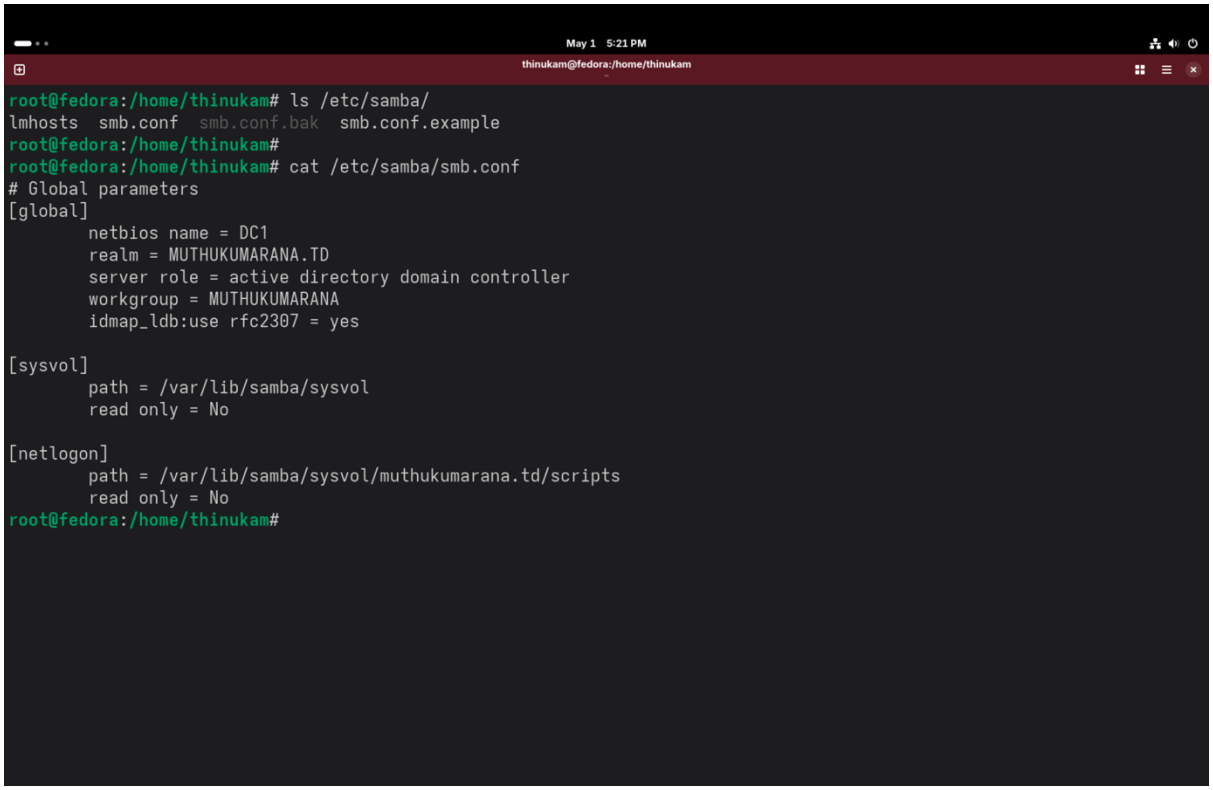

Figure 1.2. 7 - Verify the Samba configuration file

1.3. Configuring Kerberos

- The Kerberos configuration file *krb5.conf* is stored in the directory */usr/share/samba/setup/* by default.
- Copy the *krb5.conf* Kerberos configuration file from the default location to the directory */etc/krb5.conf.d/* as *samba-dc*

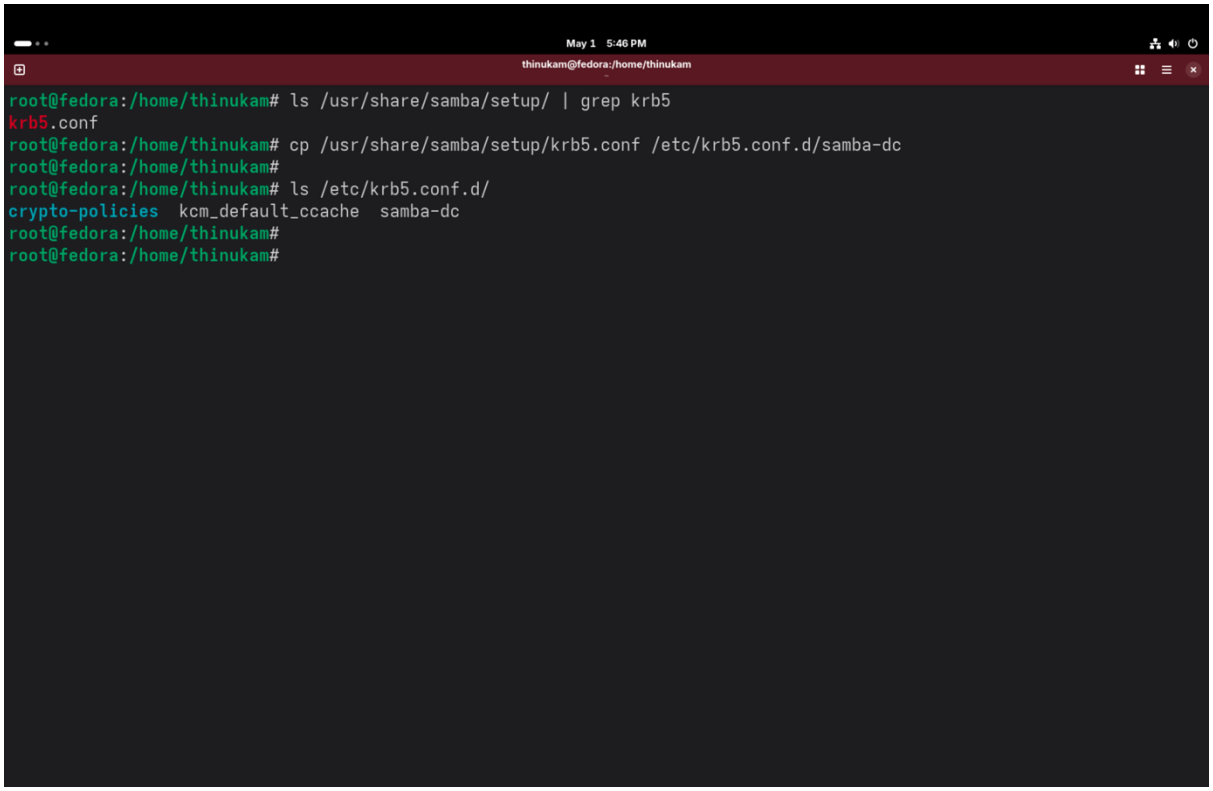

Figure 1.3. 1 - Copy the Kerberos Configuration file to /etc/krb5.conf.d/ directory

- Modify the *samba-dc* Kerberos configuration file to match the domain information

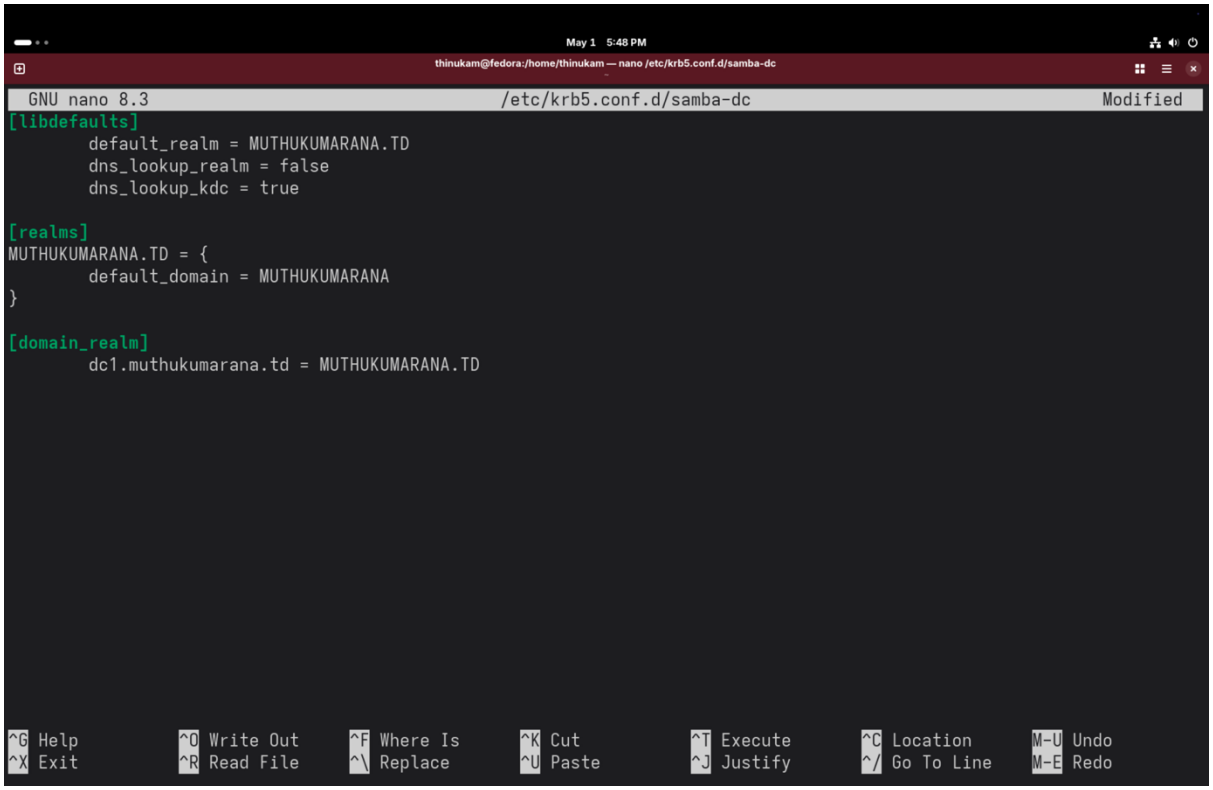

Figure 1.3. 2 - samba-dc Kerberos Configuration file

1.4. Starting the Samba Service

- Start and enable the Samba service to ensure the service starts on system initialization

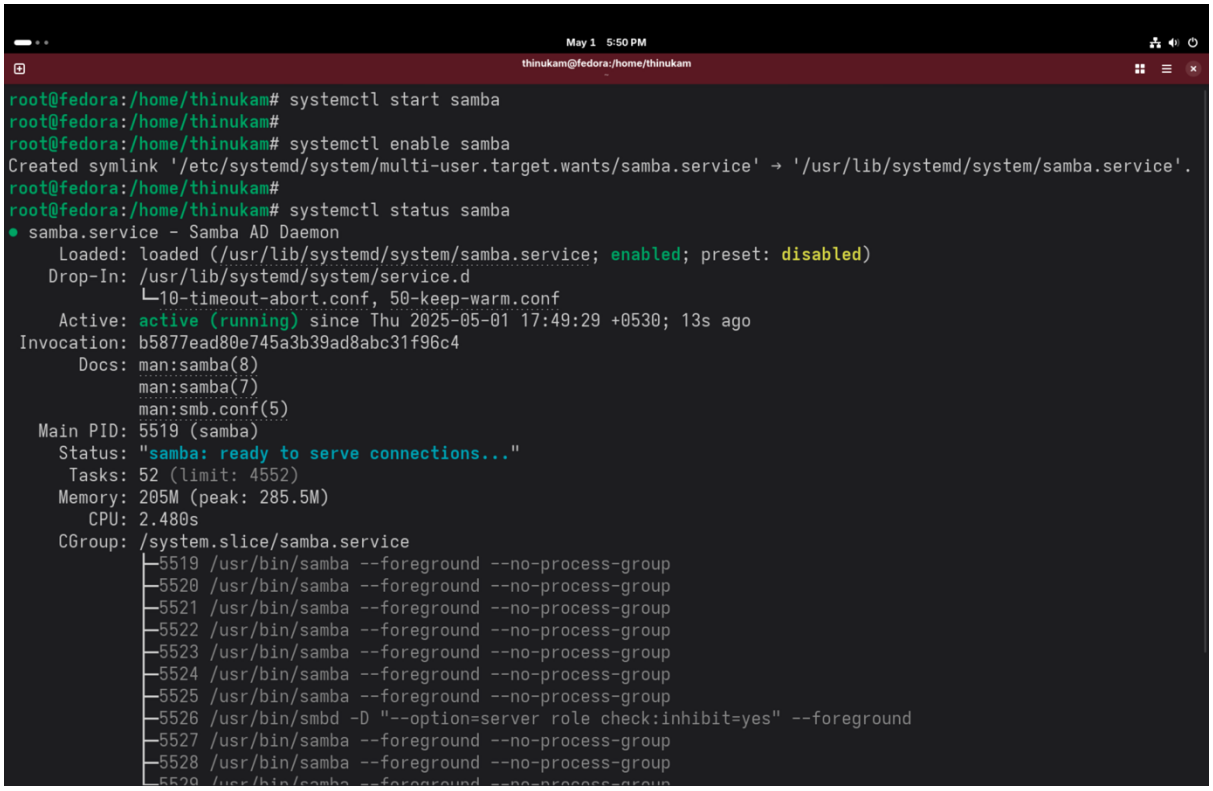

Figure 1.4. 1 - Start and enable the Samba service

- Allow TCP and UDP ports through the firewall so the clients will be able to connect to the Domain Controller

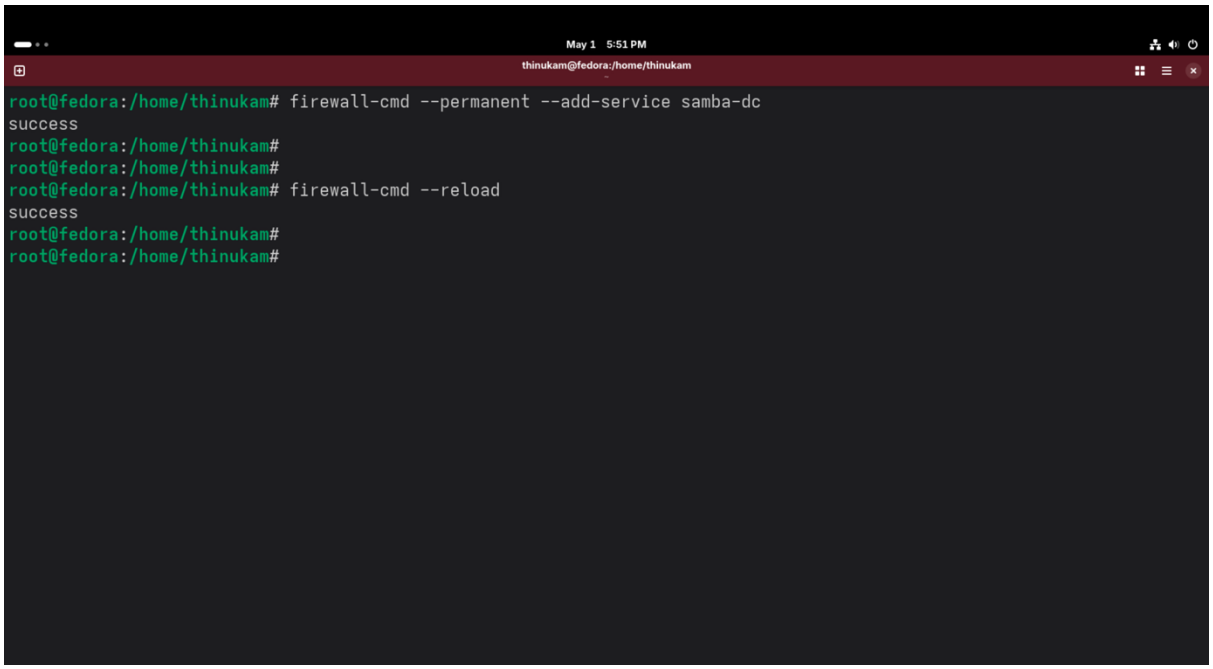

Figure 1.4. 2 - Allow TCP and UDP ports through the firewall

1.5. Testing the Samba Server

- Test connectivity to the local Samba Server

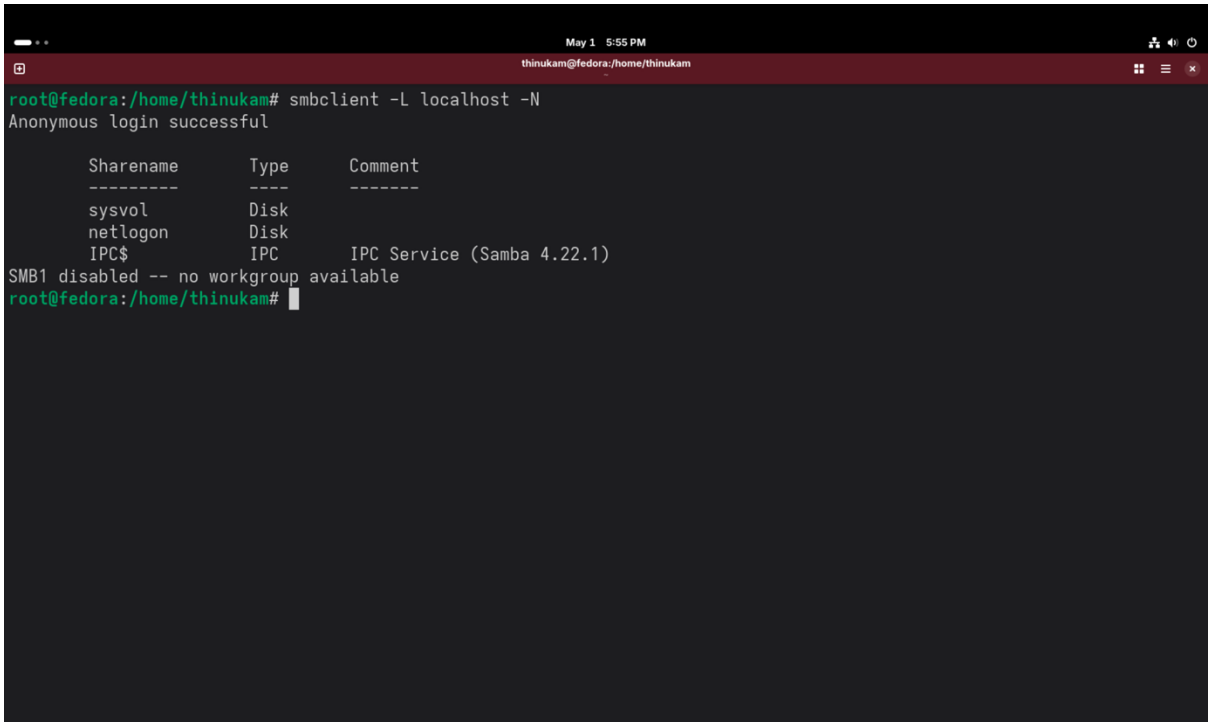

Figure 1.5. 1 - Local Samba Server connectivity test

- Test the *Administrator* login to the *netlogon* share

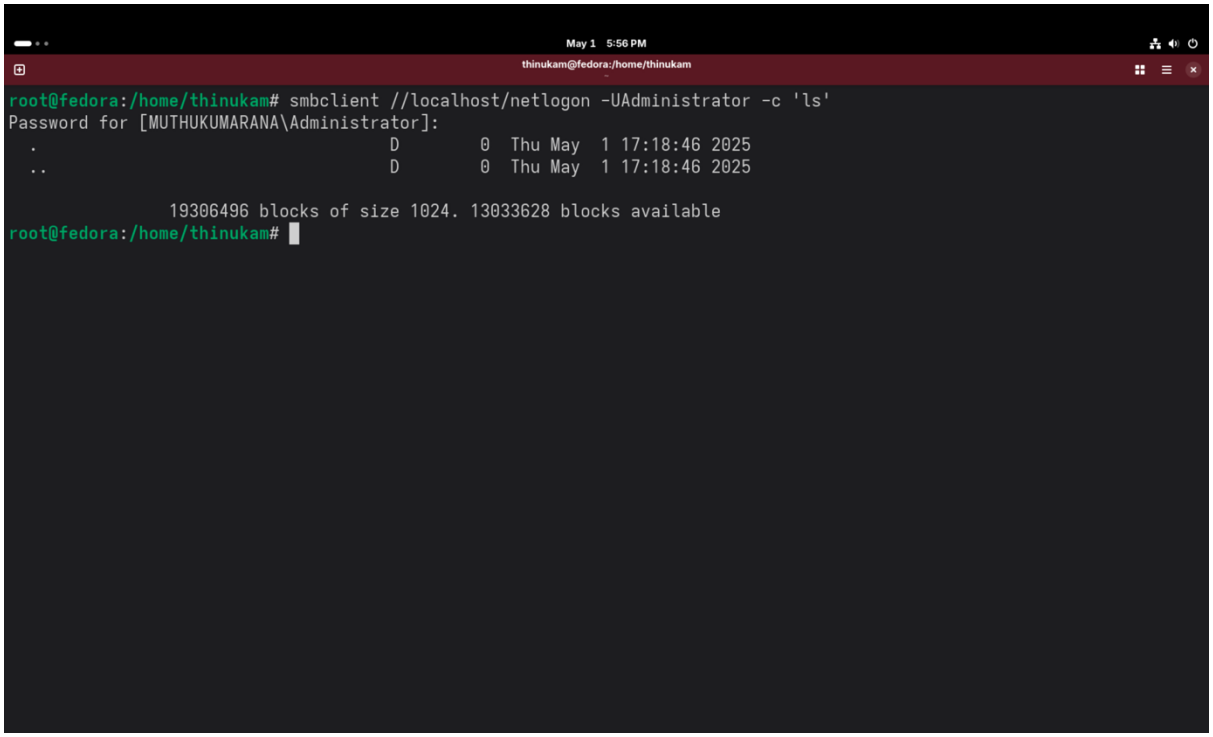

Figure 1.5. 2 - Administrator login test

- Test if the name resolution is working

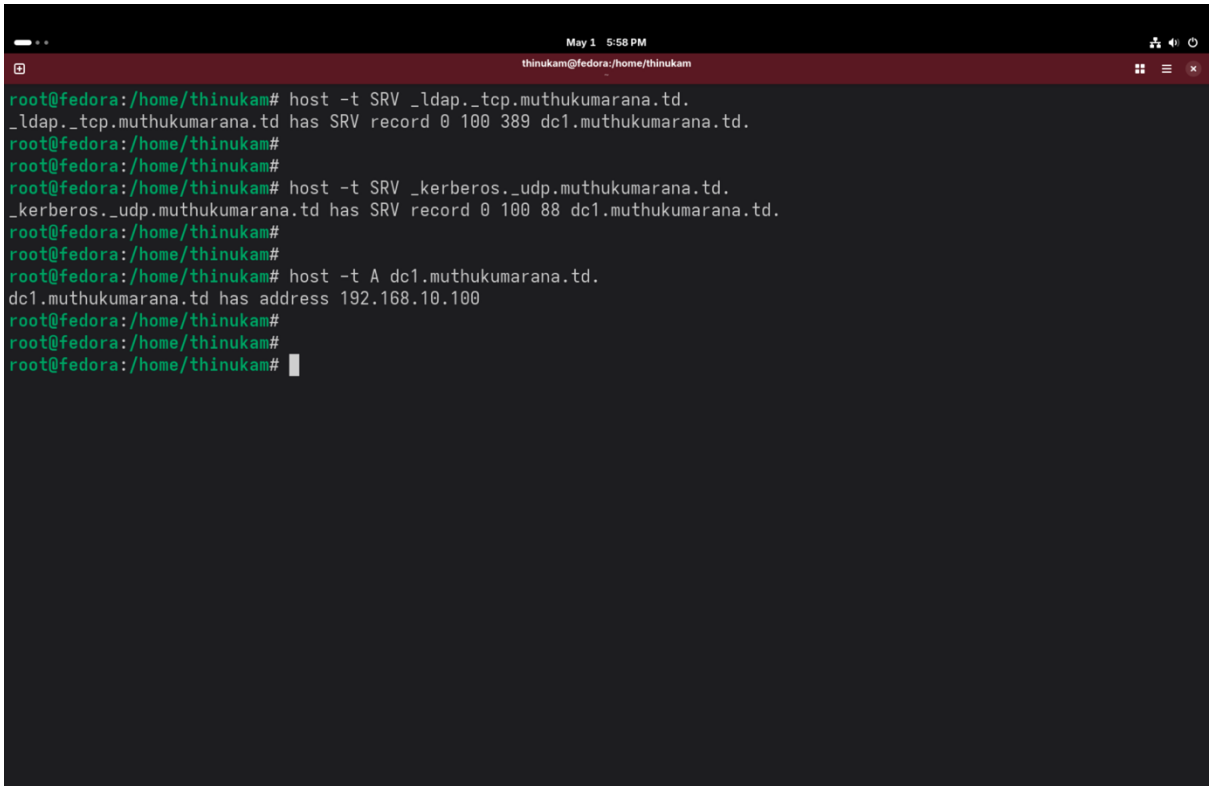

Figure 1.5. 3 – Name resolution (DNS) test

- Test if Kerberos Authentication is working

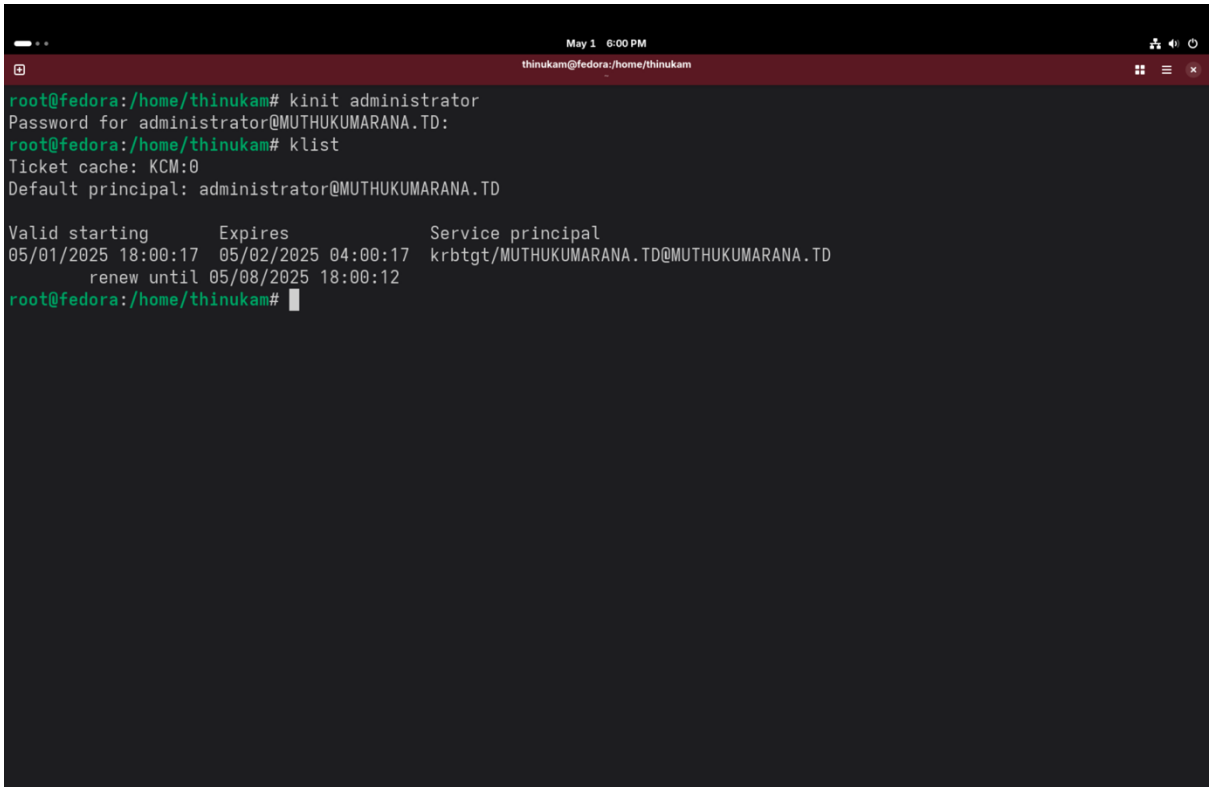

Figure 1.5. 4 - Kerberos test

1.6. Configuring the DHCP Server

- Install the required DHCP Server package

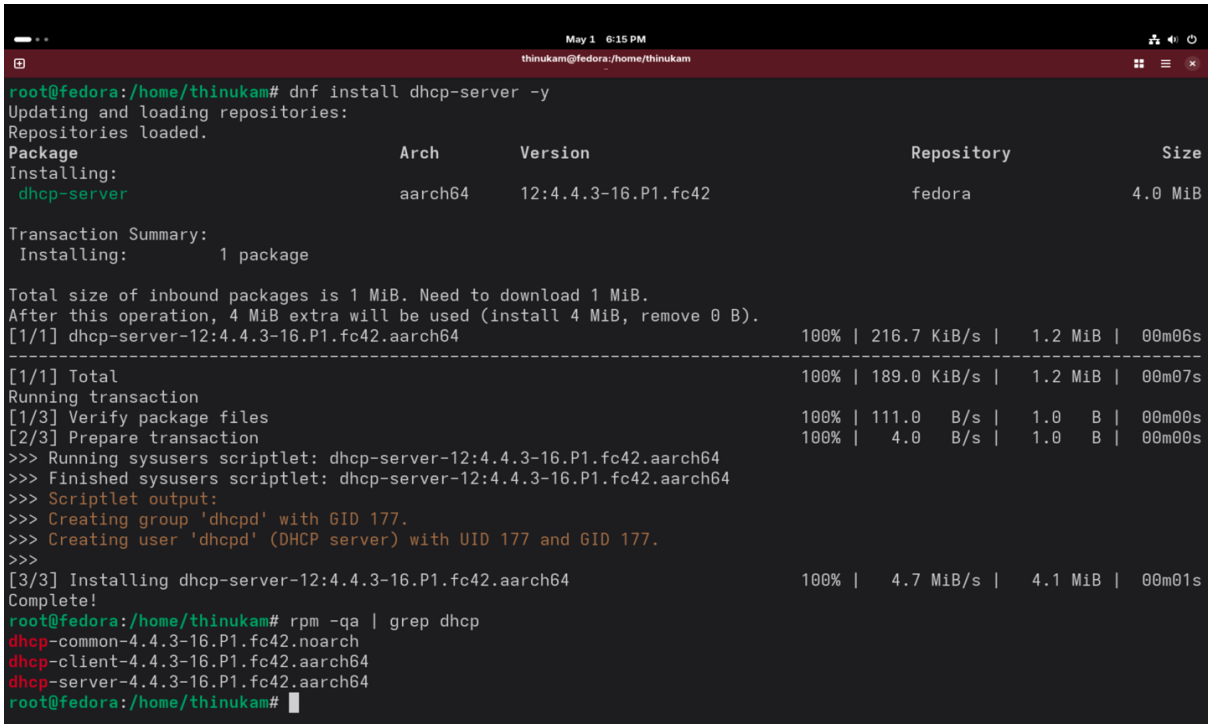

Figure 1.6. 1 - Install the DHCP package

- Specify the interface that distributes DHCP leases by modifying the */etc/sysconfig/dhcpd* file

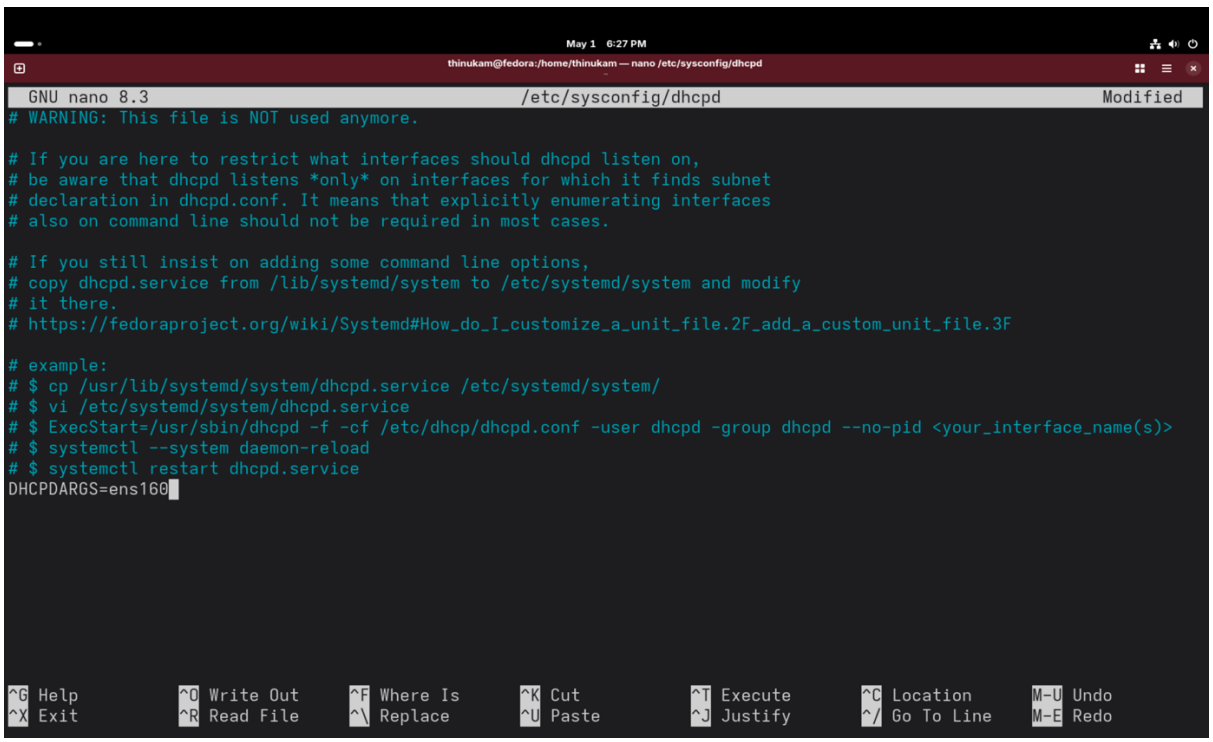

Figure 1.6. 2 – /etc/sysconfig/dhcpd file

- Modify the DHCP configuration file */etc/dhcp/dhcpd.conf* to distribute IP addresses by defining the pool

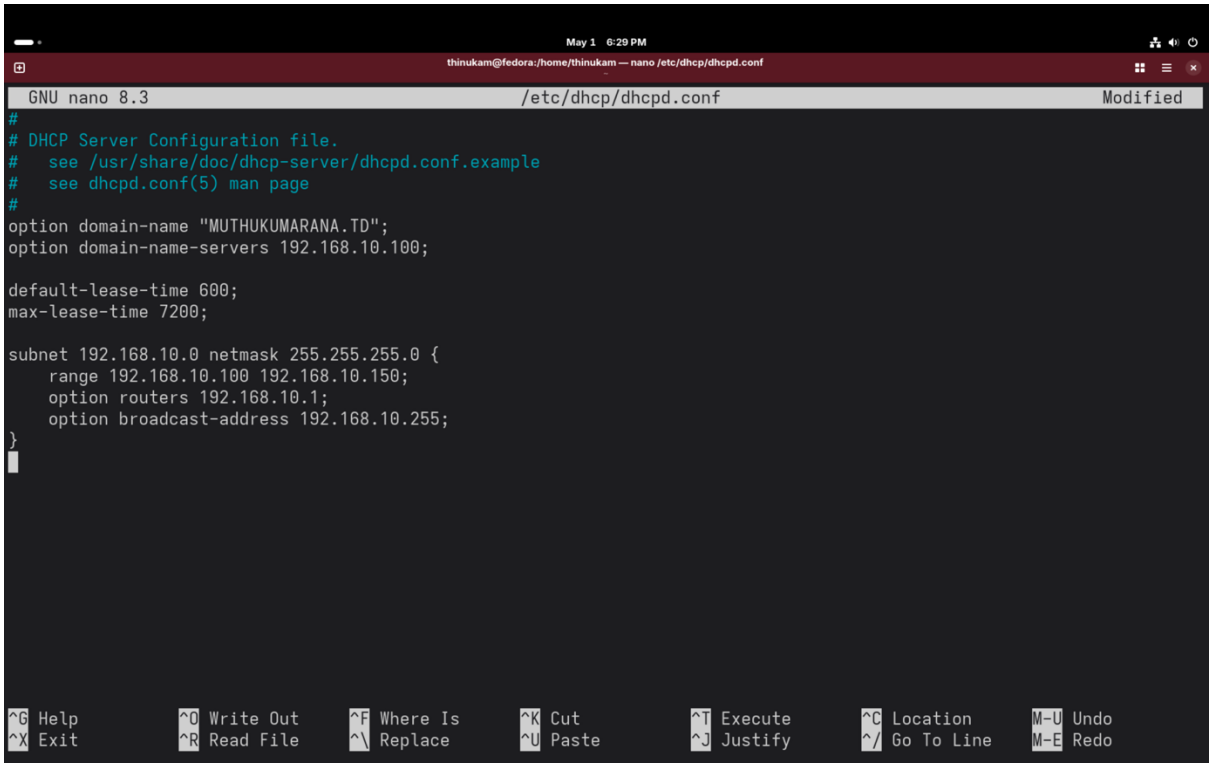

Figure 1.6. 3 - /etc/dhcp/dhcpd.conf file

- Start and enable the DHCP server, and verify the status of the server

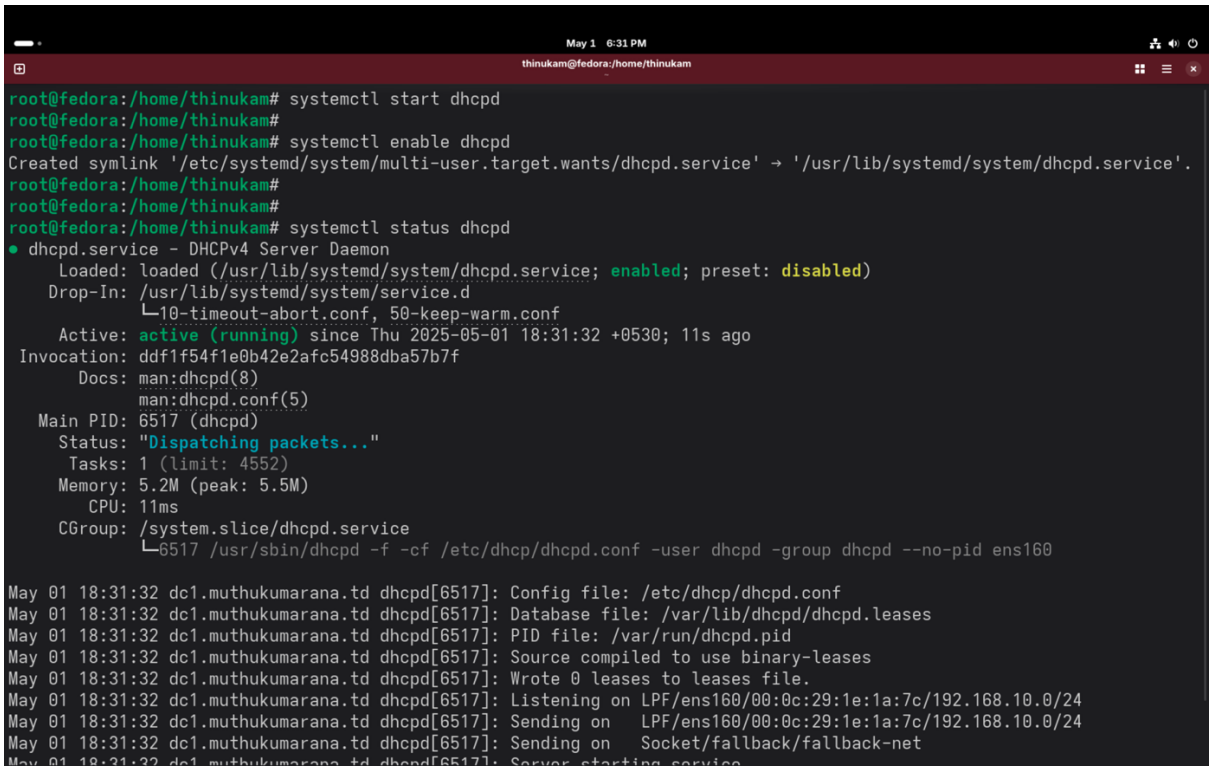

Figure 1.6. 4 - Start and enable the DHCP server

1.7. Adding Users to the Domain

- Use the *samba-tool* to add users to the domain and specify authentication passwords, and verify that the users are added to the list

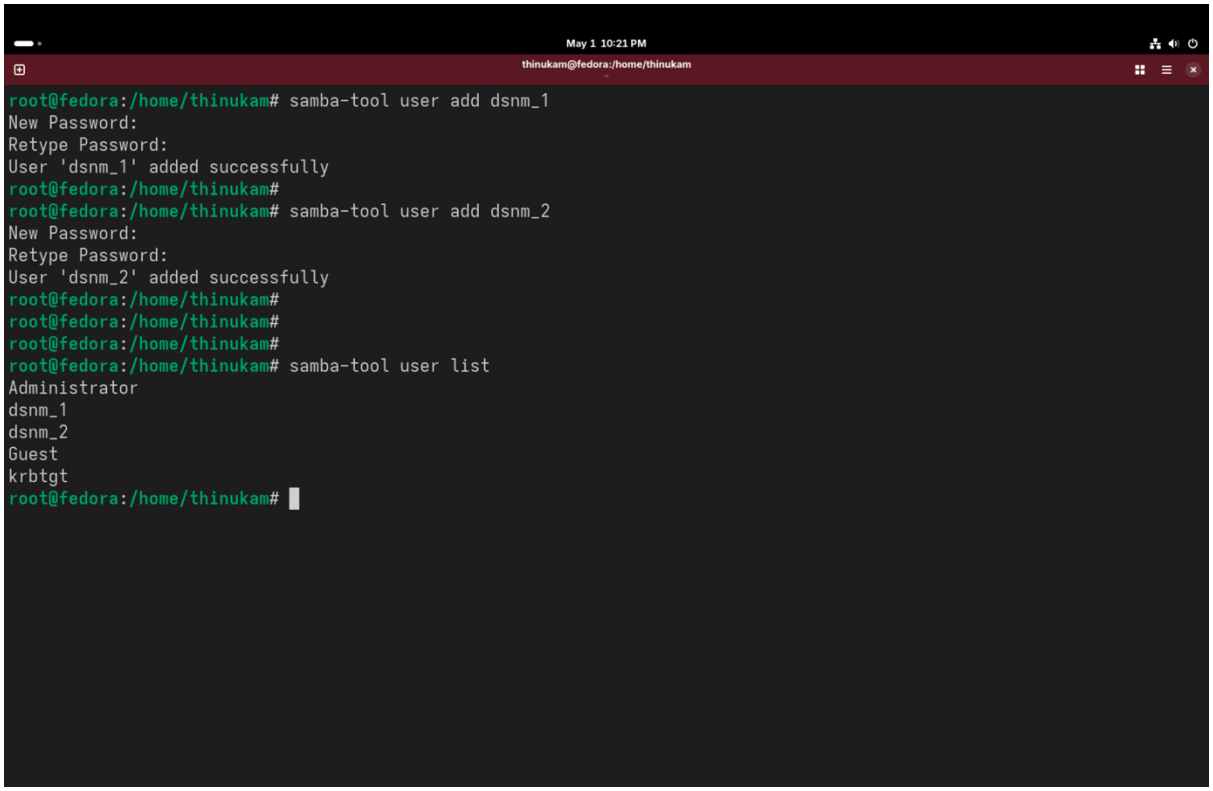

Figure 1.7. 1 - Adding users to the domain and verifying

2. Join Ubuntu Clients to the AD Domain

2.1. Preparing the client machine/s

- Update the Ubuntu operating system

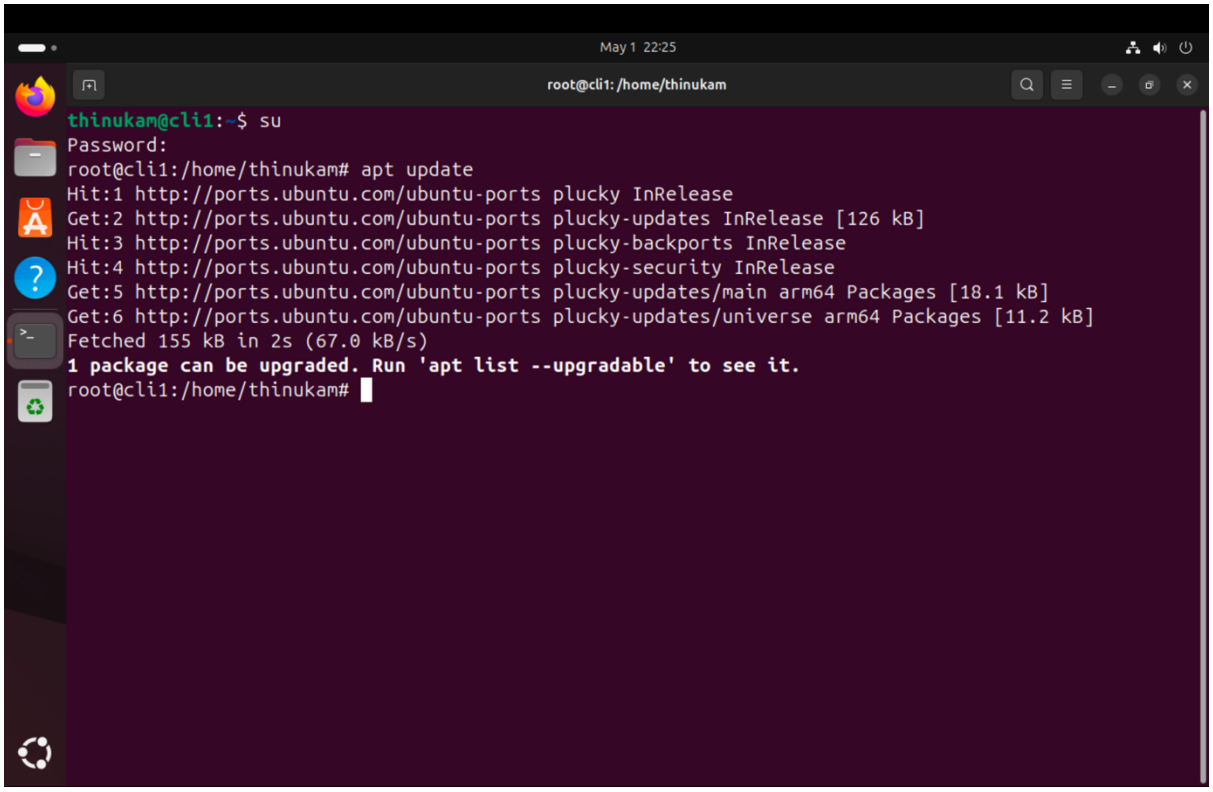

Figure 2.1. 1 - Update Ubuntu OS

- Install the required packages to join the domain

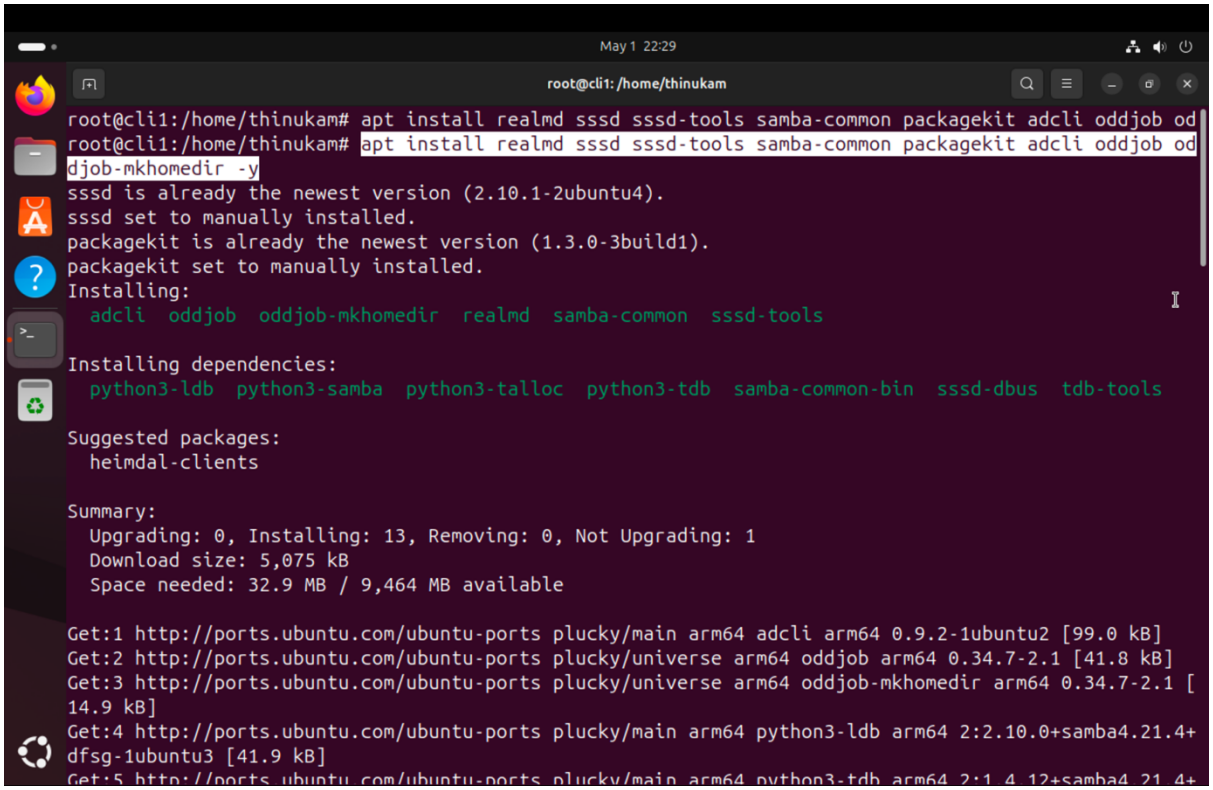

Figure 2.1. 2 - Install the required packages

- Modify the */etc/hosts* to ensure the DC is resolvable by the hostname

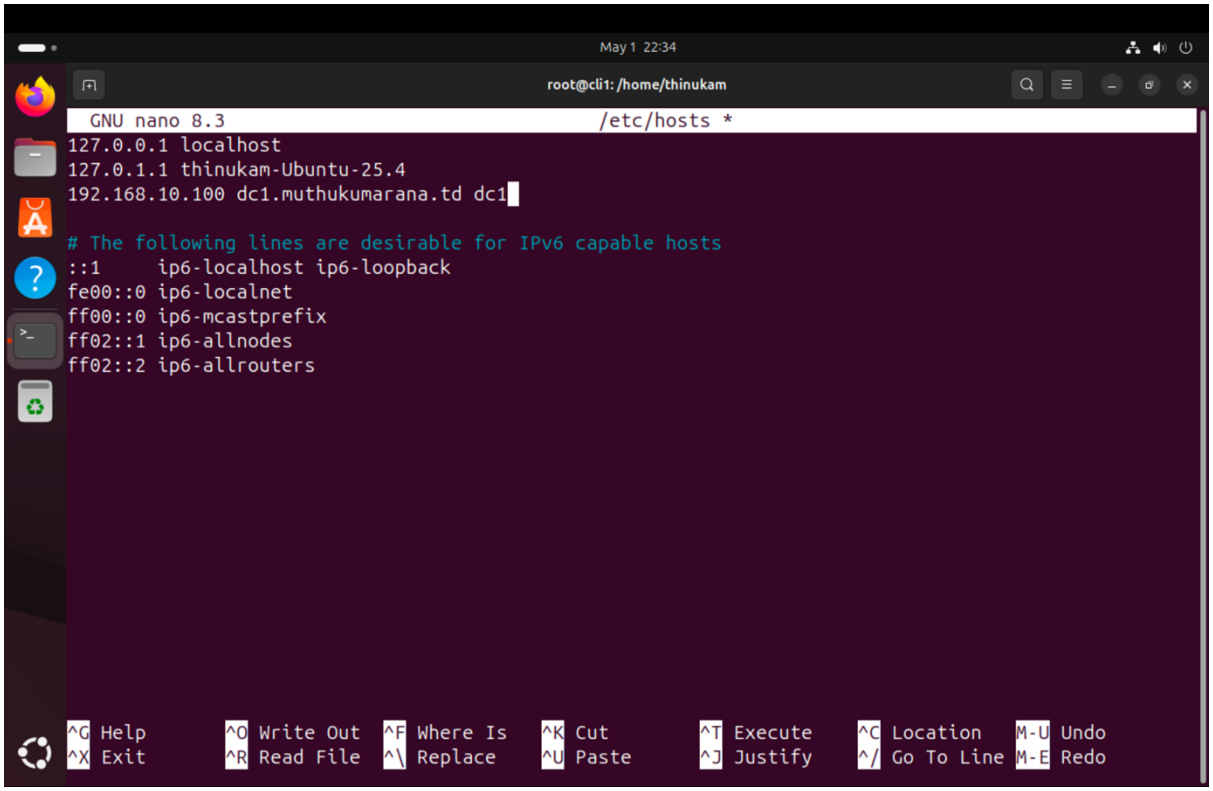

Figure 2.1. 3 - /etc/hosts file

2.2. Join the Domain

- Discover the domain with the *realm* and retrieve domain configuration information

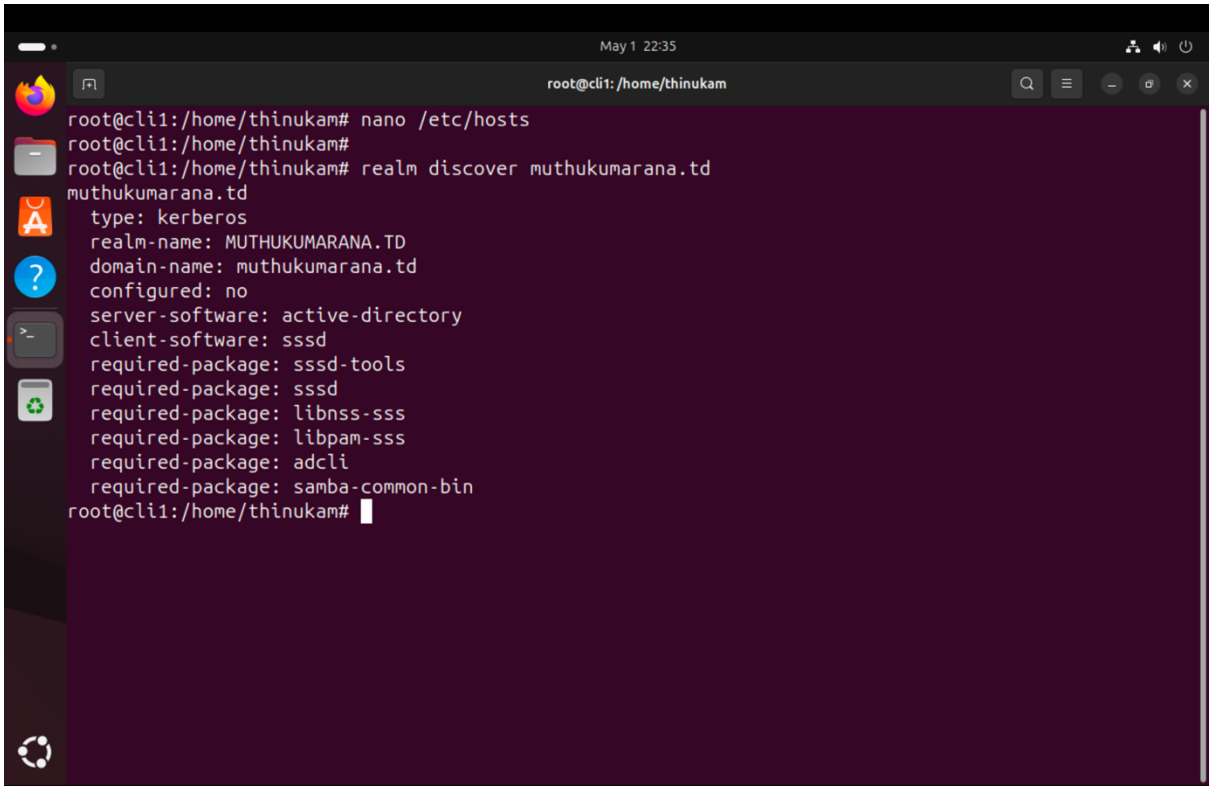

Figure 2.2. 1 - Discover the Domain

- Join the domain with the *realm* and verify the join

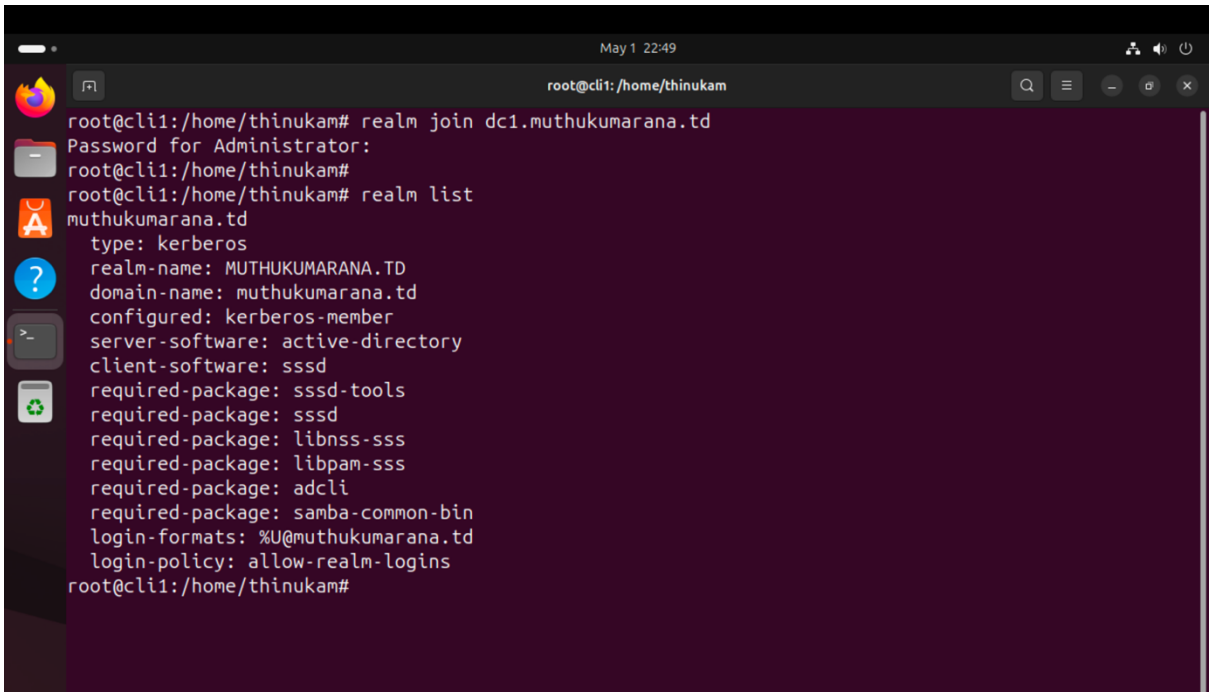

Figure 2.2. 2 - Join the domain

- Verify the SSSD configuration file */etc/sssd/sssd.conf*

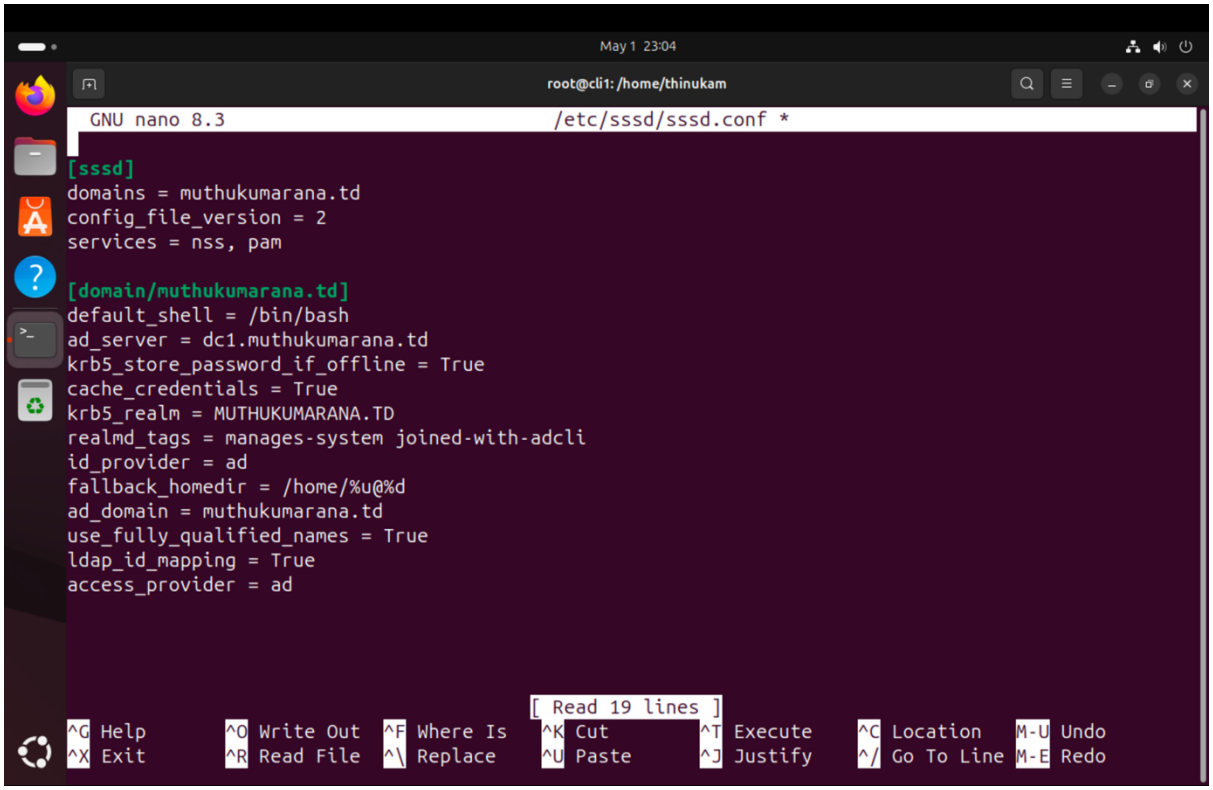

Figure 2.2. 3 - /etc/sssd/sssd.conf file

2.3. Domain Login

- Log in to the domain using the username and password created

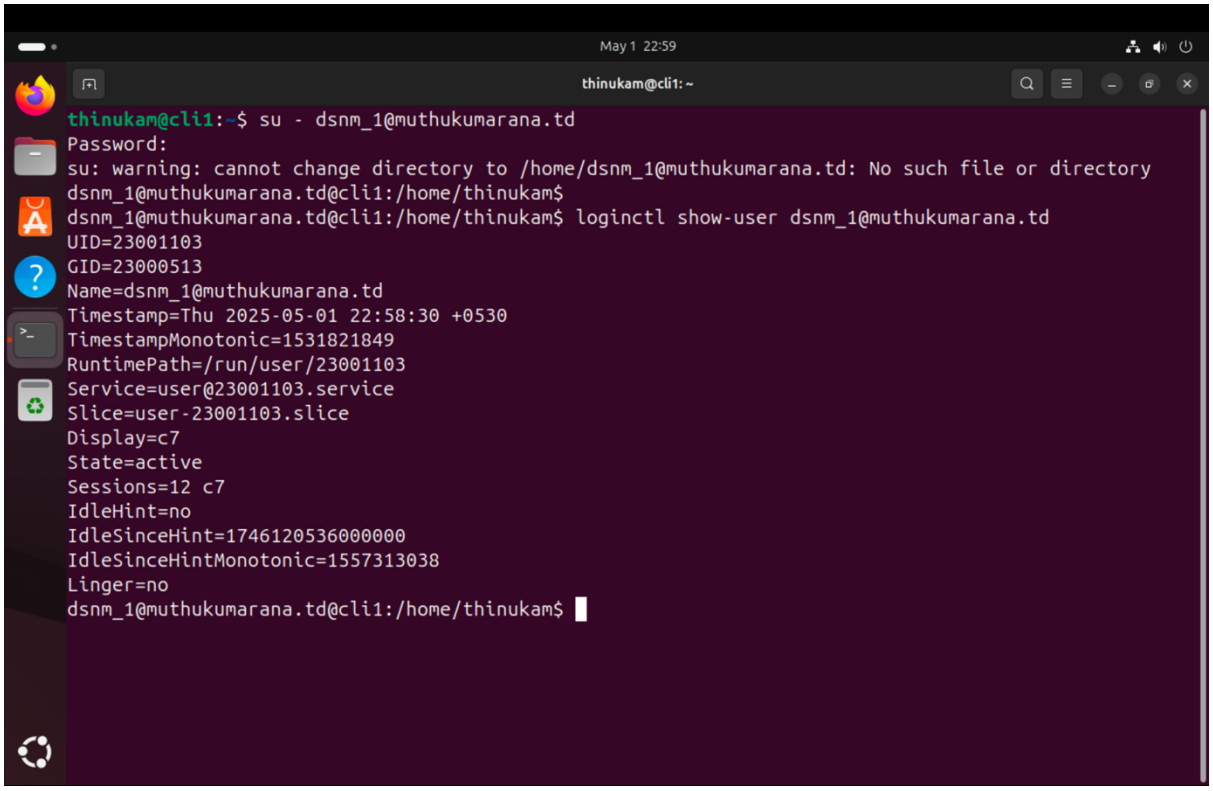

Figure 2.3. 1 - Domain login with username 1

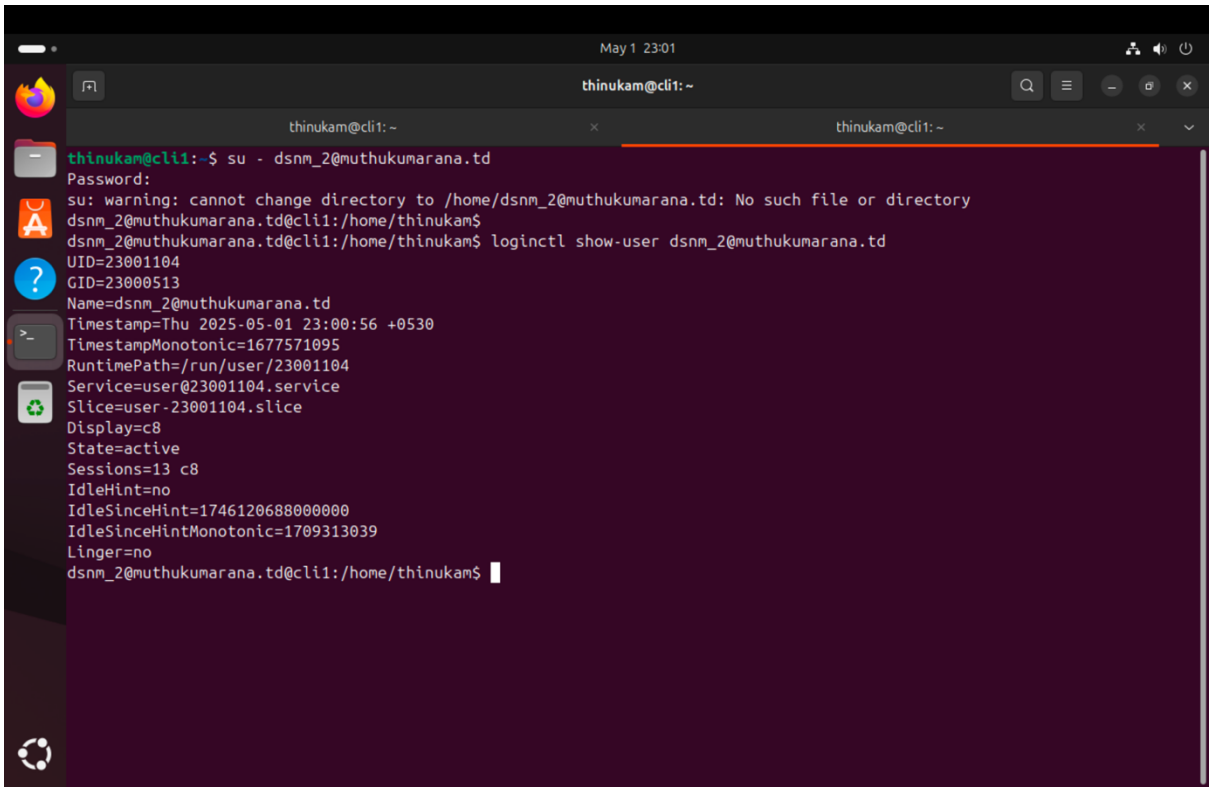

Figure 2.3. 2 - Domain login with username 2

- Optionally enable the creation of a home directory upon domain login for domain users with the command *pam-auth-update*

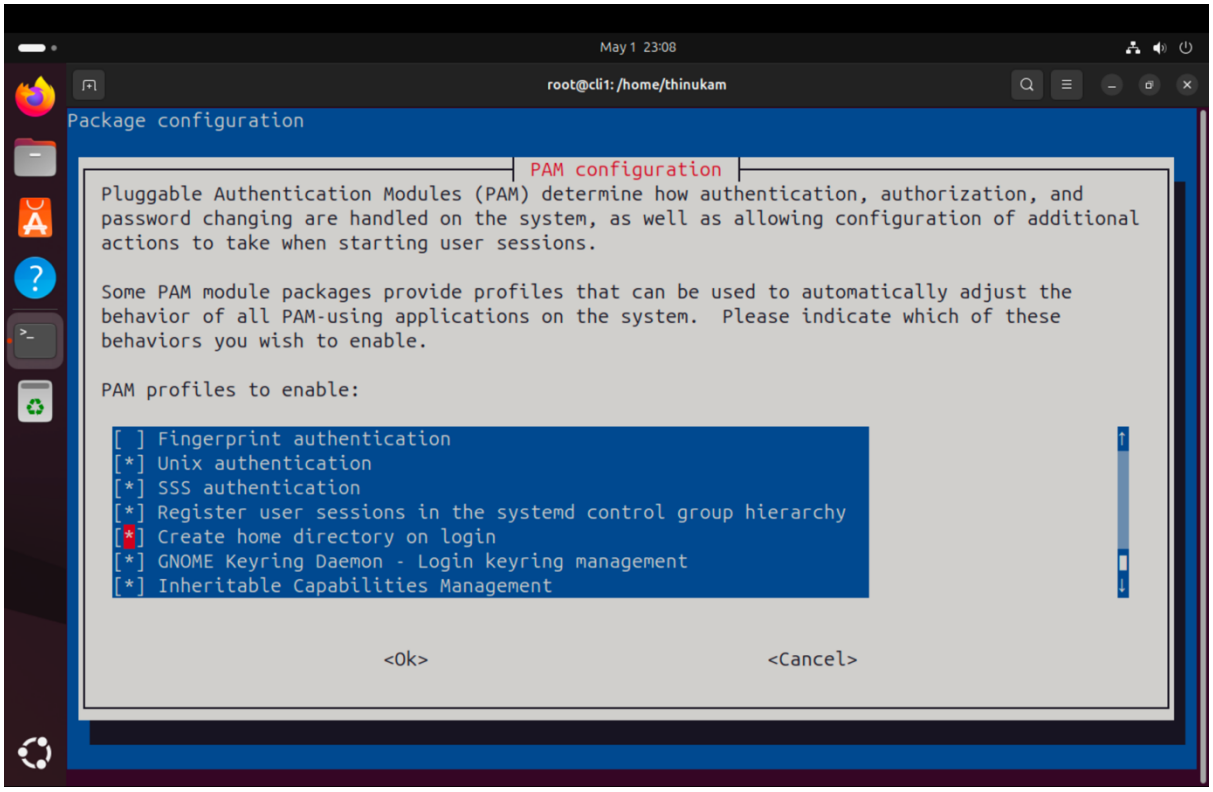

Figure 2.3. 3 - Enable home directory creation upon login

- Upon completion, analogous procedures are employed to establish the remaining client machines.
- To enable domain login upon initialization using the GUI,
  i. Reboot the machine and select the *'not listed'* option displayed in the login window
  ii. Enter the correct domain credentials
  iii. Verify domain information
  iv. Reboot the system, and the domain user will be visible

- Reboot the client machine/s and ensure domain login with the GUI

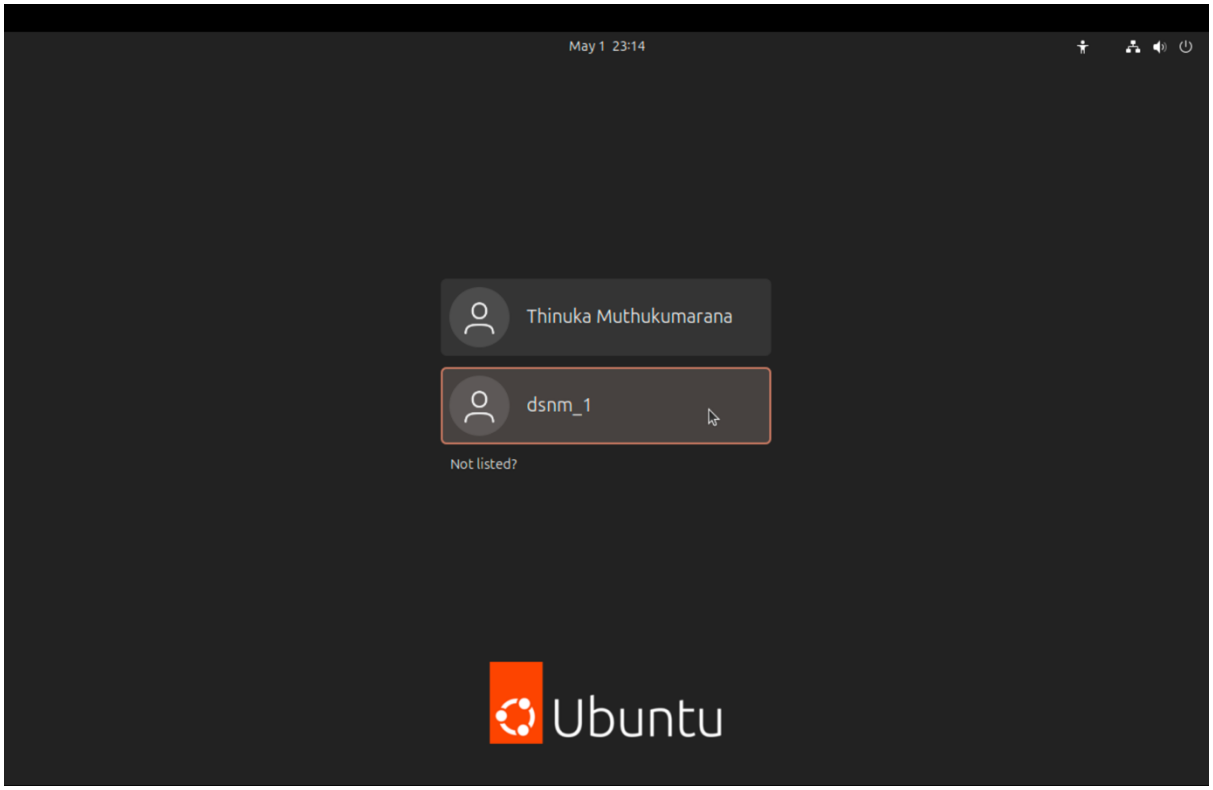

Figure 2.3. 4 - Domain login for user 1 with GUI

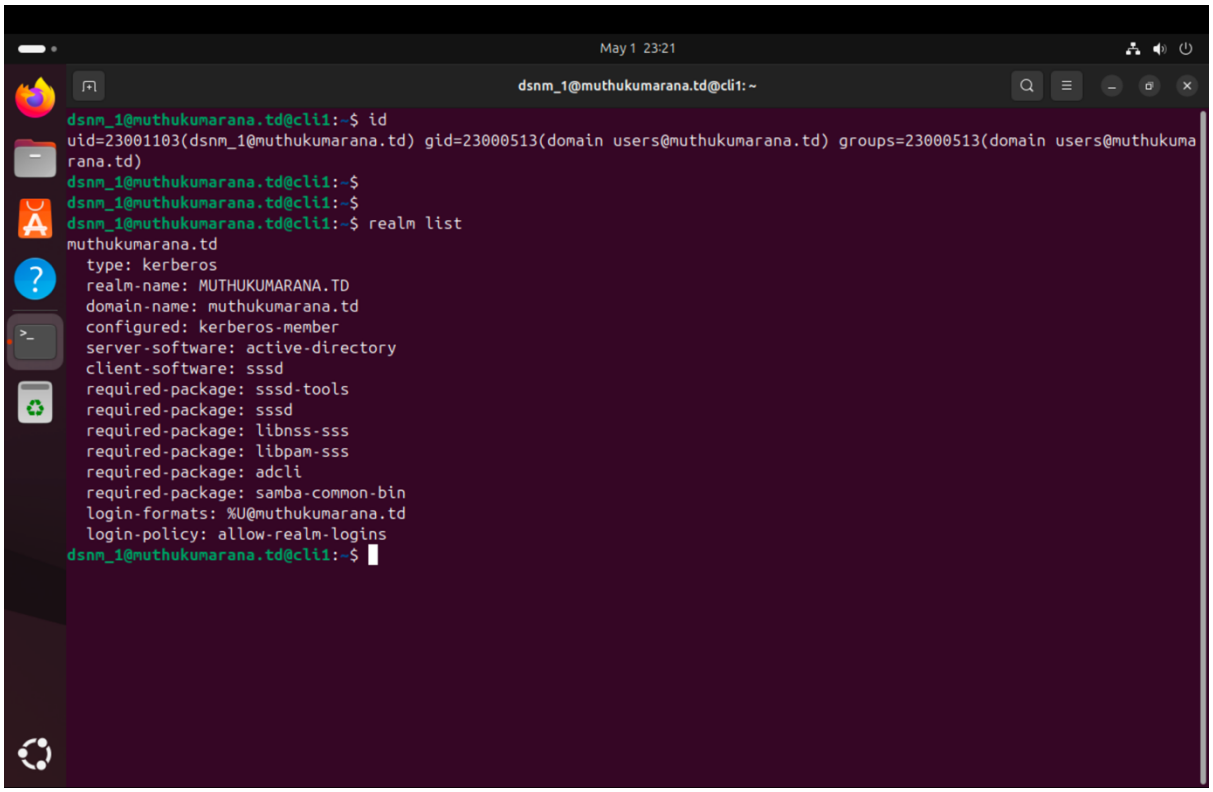

Figure 2.3. 5 – Domain information for user 1

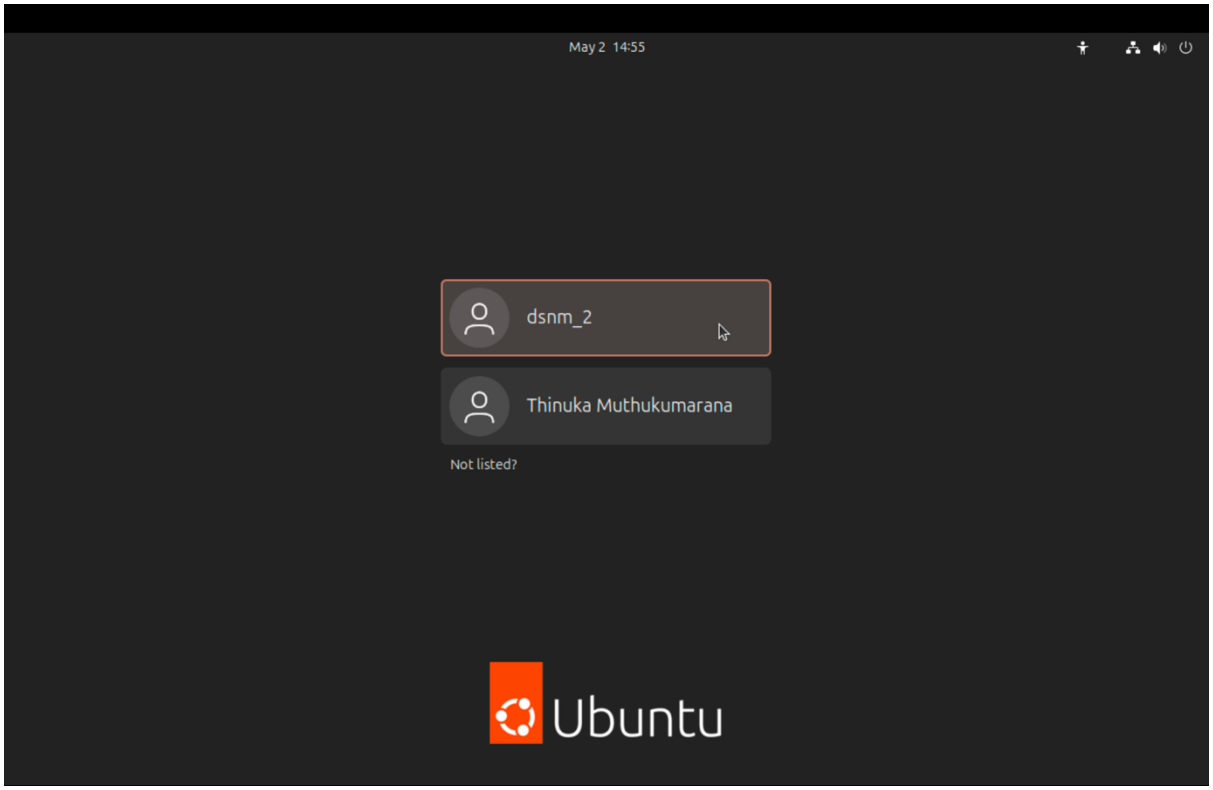

Figure 2.3. 6 - Domain login for user 2 with GUI

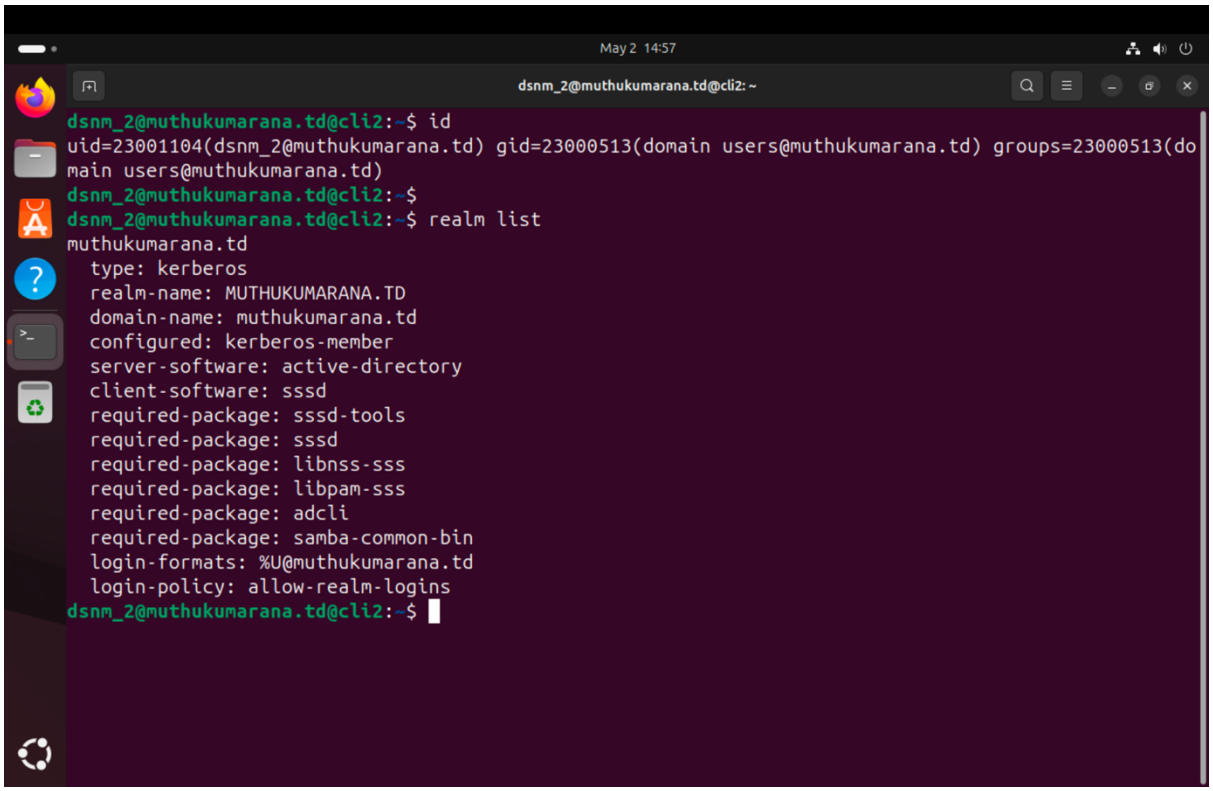

Figure 2.3. 7 - Domain information for user 2

3. Configuring the Zabbix server

3.1. Installing LAMP Server

- Install Apache and MariaDB server packages

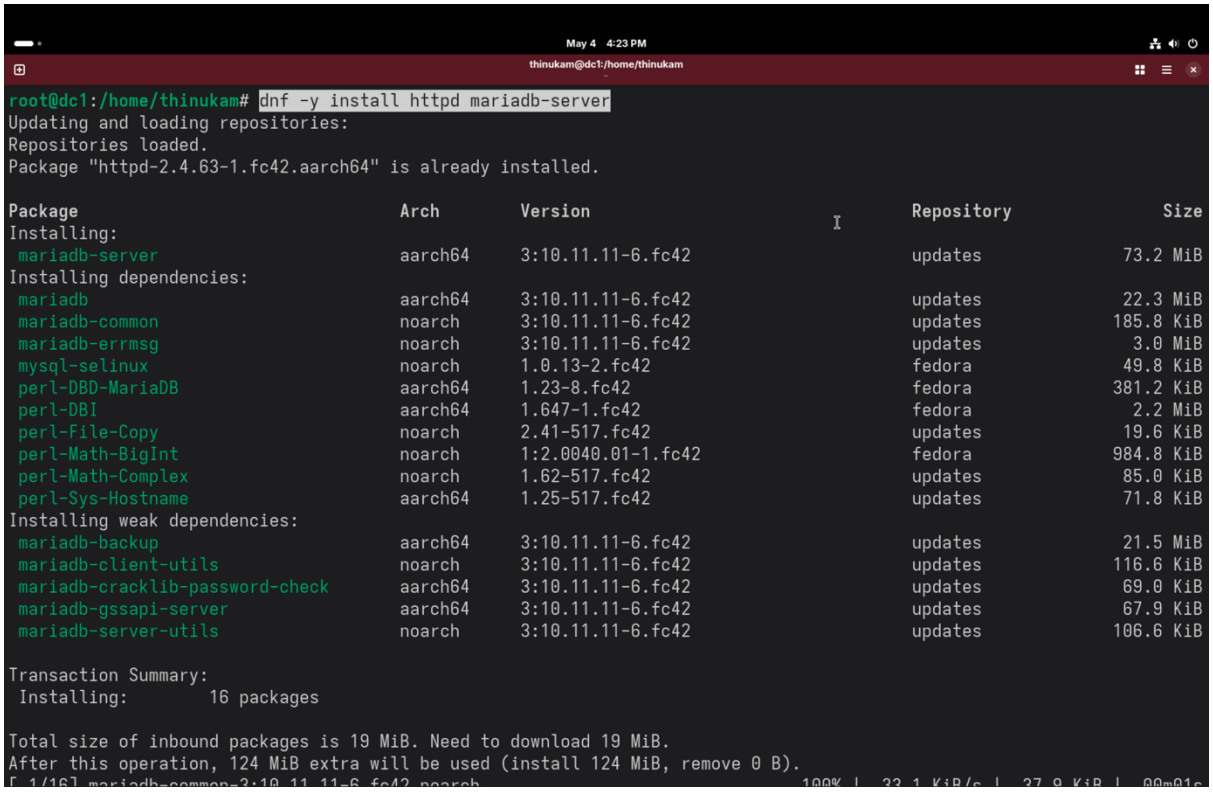

Figure 3.1. 1 - Install Apache and MariaDB server

- Install the PHP remi repository and enable the PHP 8.4 remi module

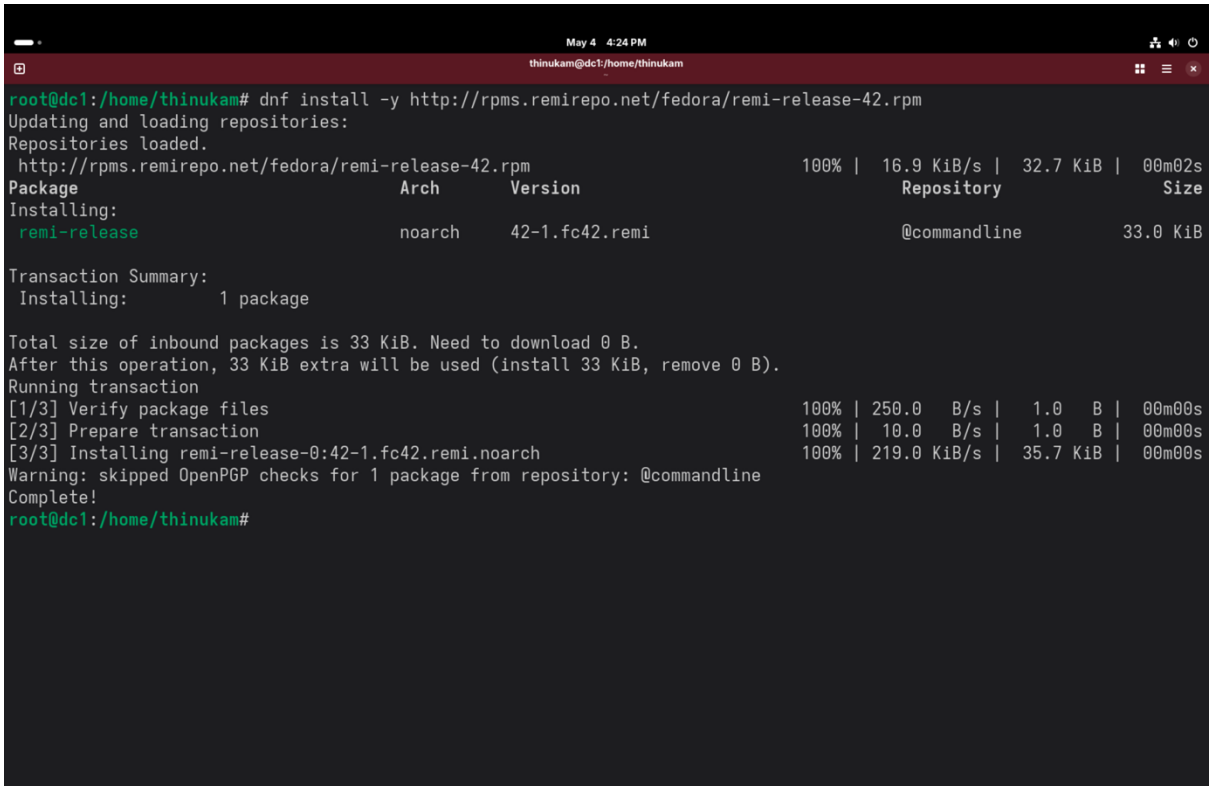

Figure 3.1. 2 - Install the PHP remi repository

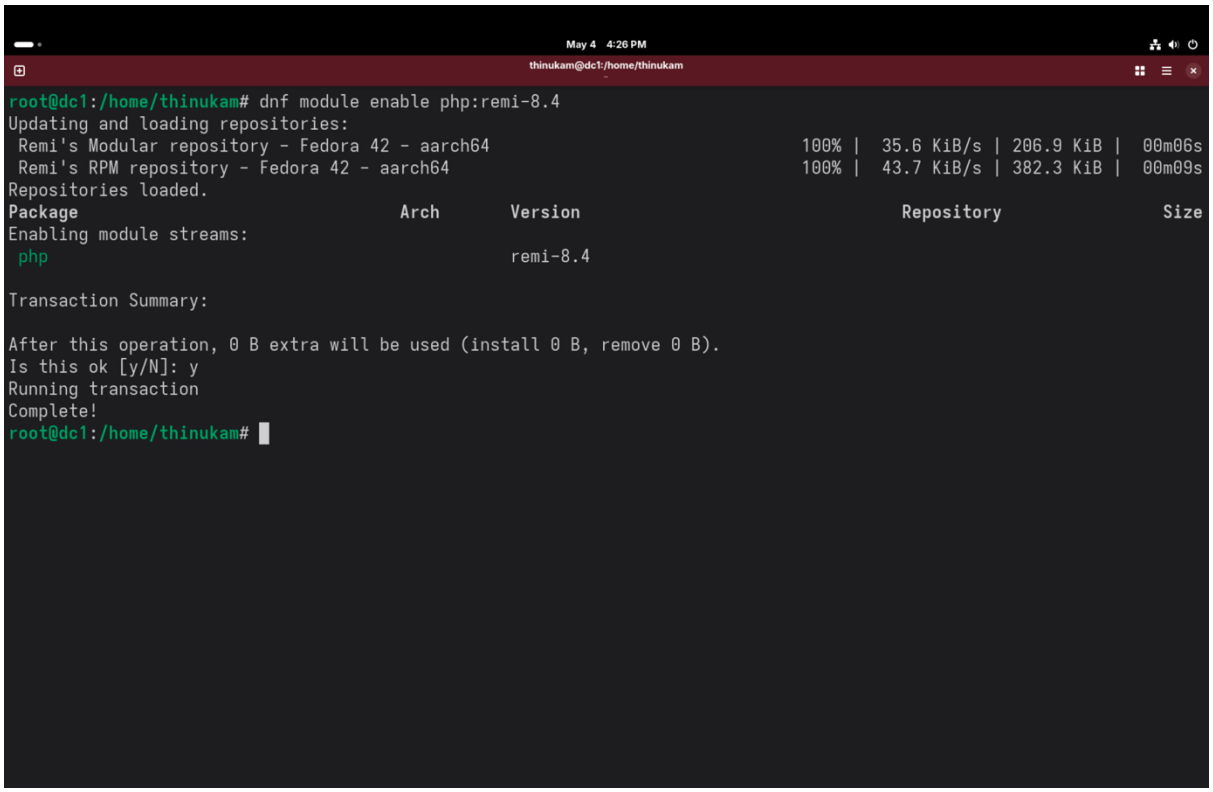

Figure 3.1. 3 - Enable the PHP 8.4 remi module

- Install other required PHP extensions

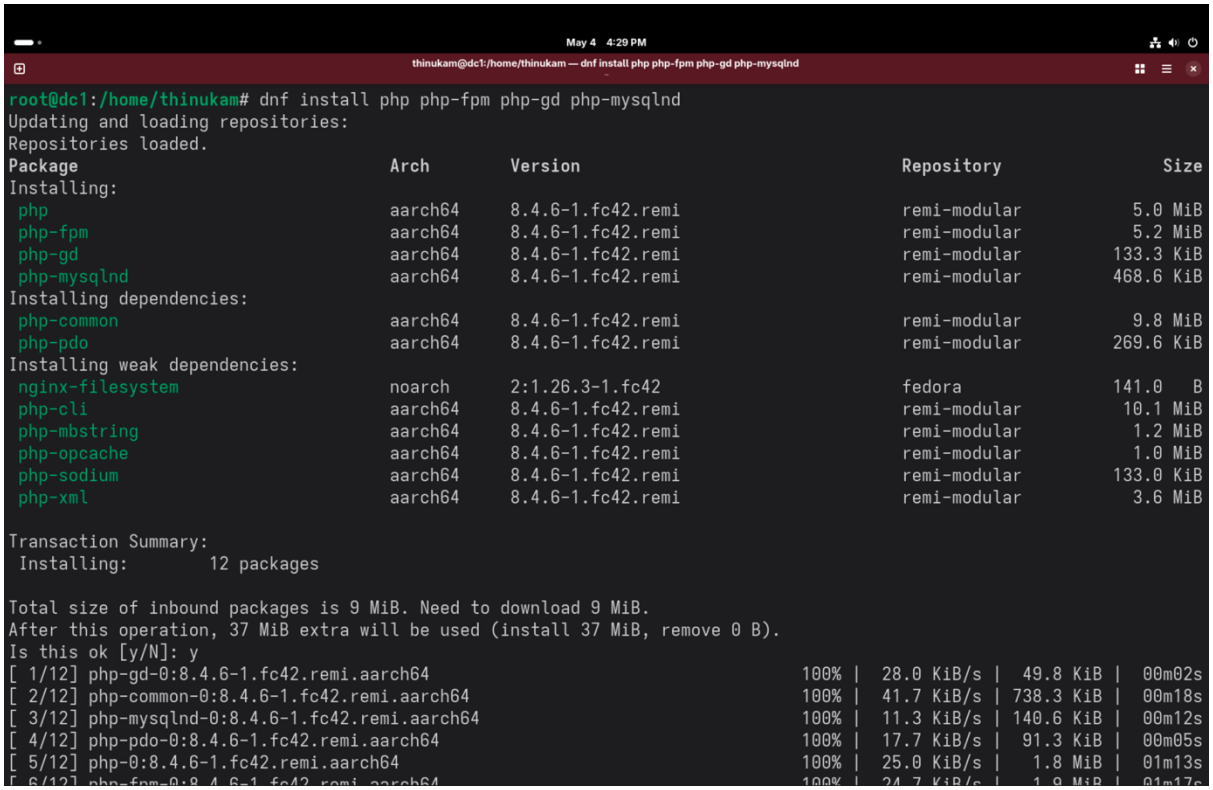

Figure 3.1. 4 - Install PHP extensions

- Modify the */etc/php-fpm.d/www.conf* file as given to allow listening for clients
  - *listen = /run/php-fpm/zabbix.sock*
  - *listen.allowed_clients = 0.0.0.0*

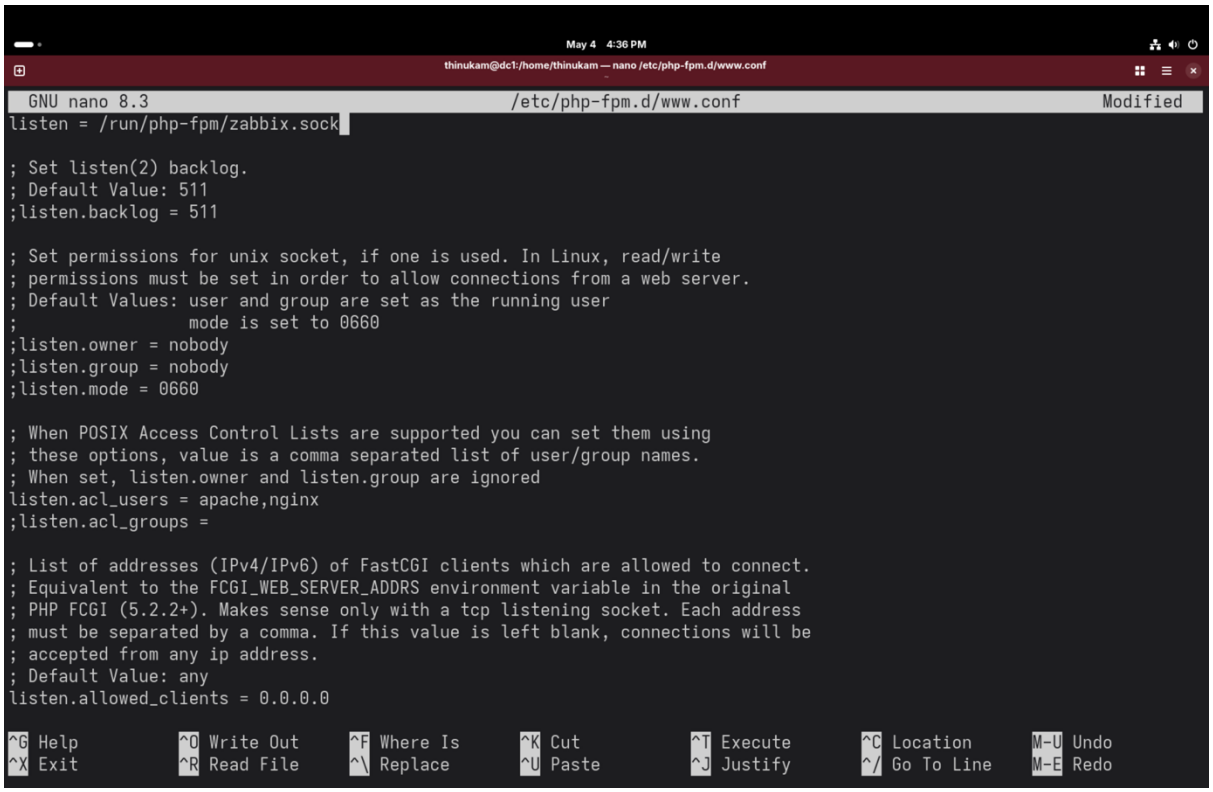

Figure 3.1. 5 - /etc/php-fpm.d/www.conf file

- Create a new Zabbix file */etc/php-fpm.d/zabbix.conf* to define date and time zone

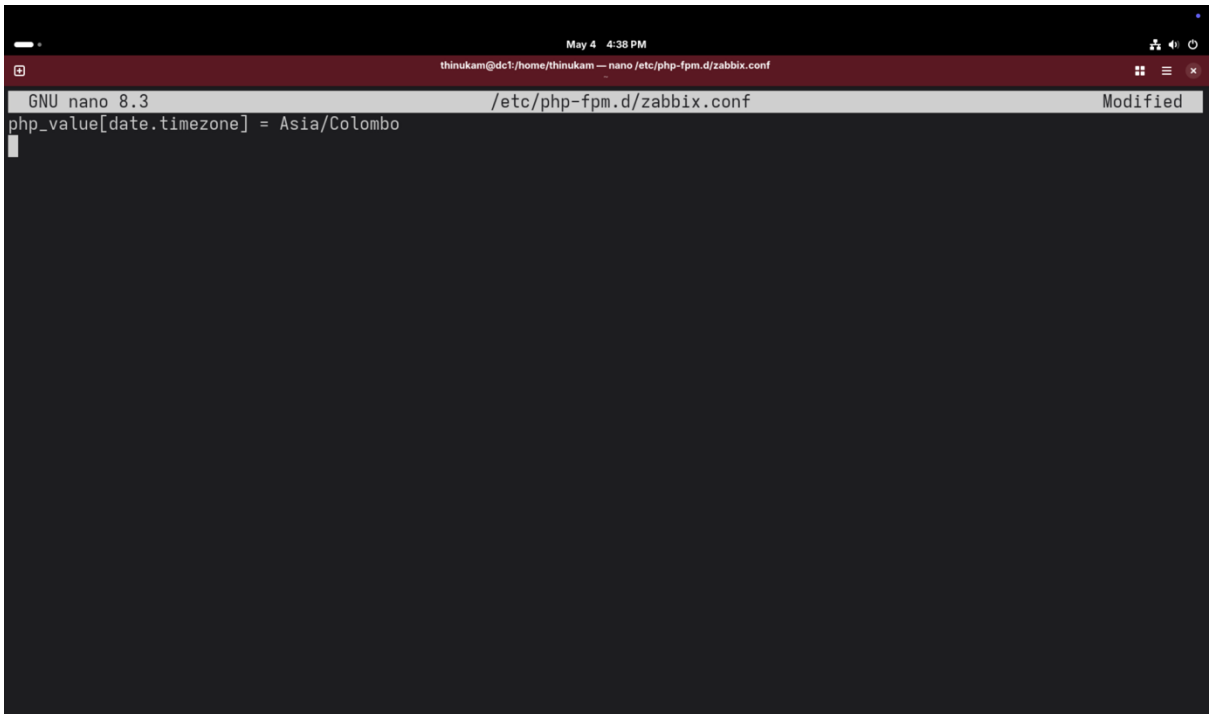

Figure 3.1. 6 - /etc/php-fpm.d/zabbixconf file

- Modify the /etc/php.ini PHP configuration file as given
  - post_max_size = 16M
  - max_execution_time = 300
  - max_input_time = 300

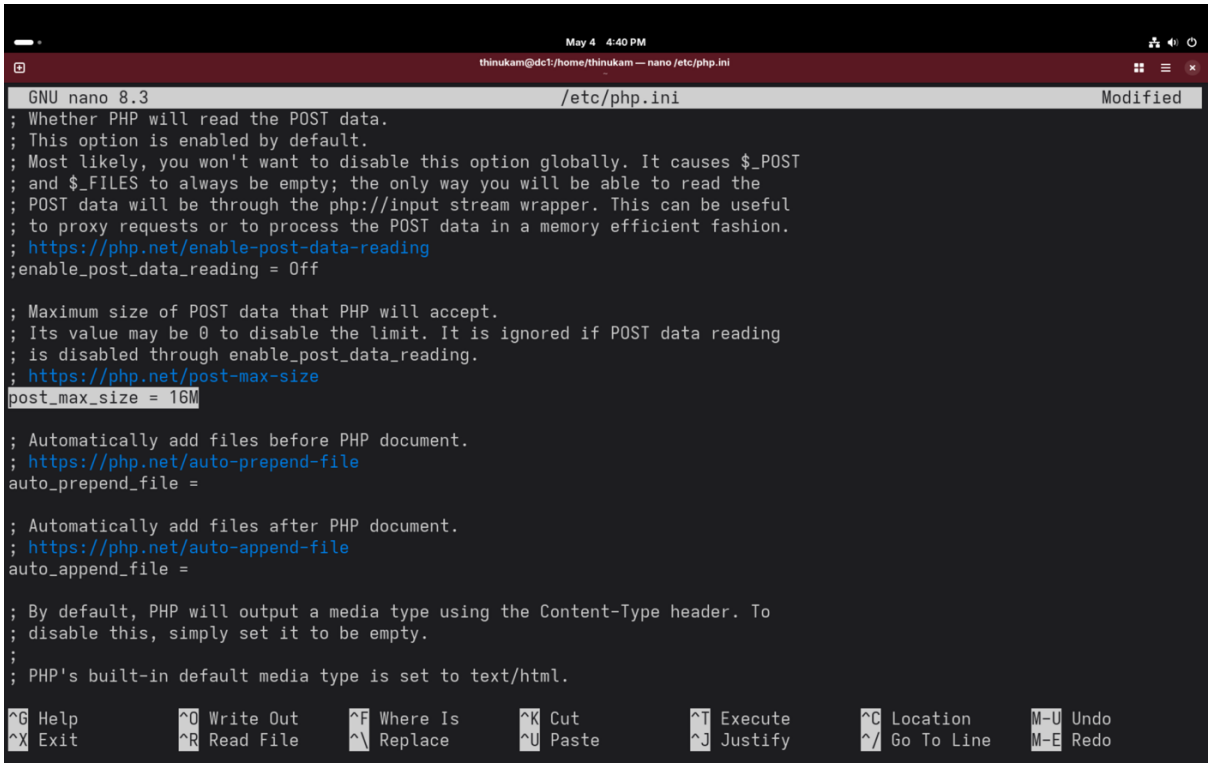

Figure 3.1. 7 - /etc/php.ini file (i)

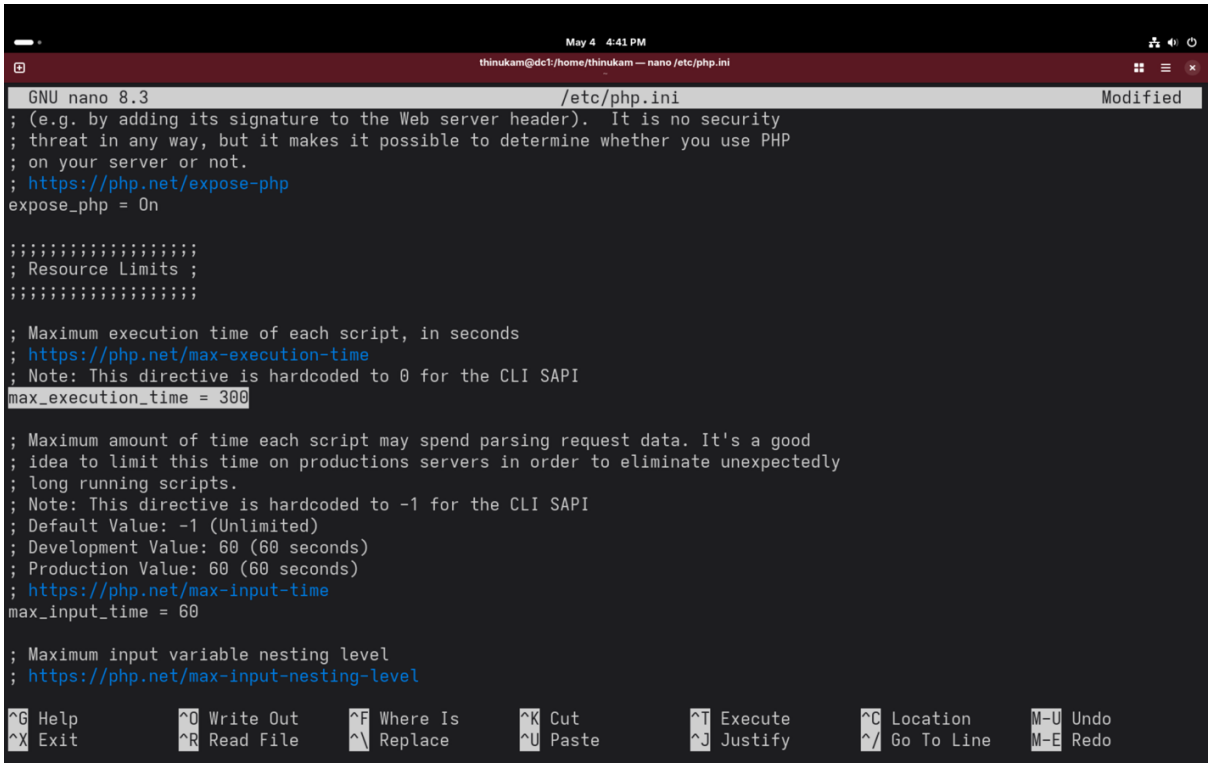

Figure 3.1. 8 - /etc/php.ini file (ii)

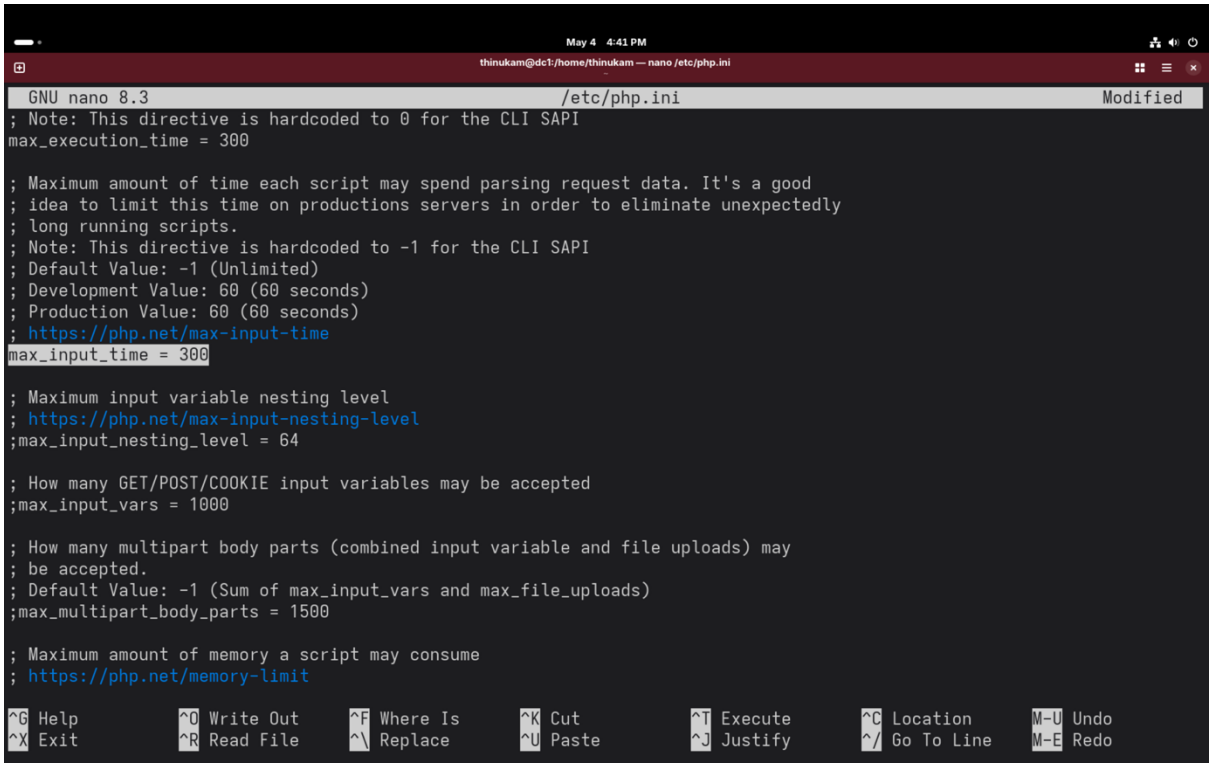

Figure 3.1. 9 - /etc/php.ini file (iii)

- Start and enable Apache, MariaDB, and PHP-FPM services

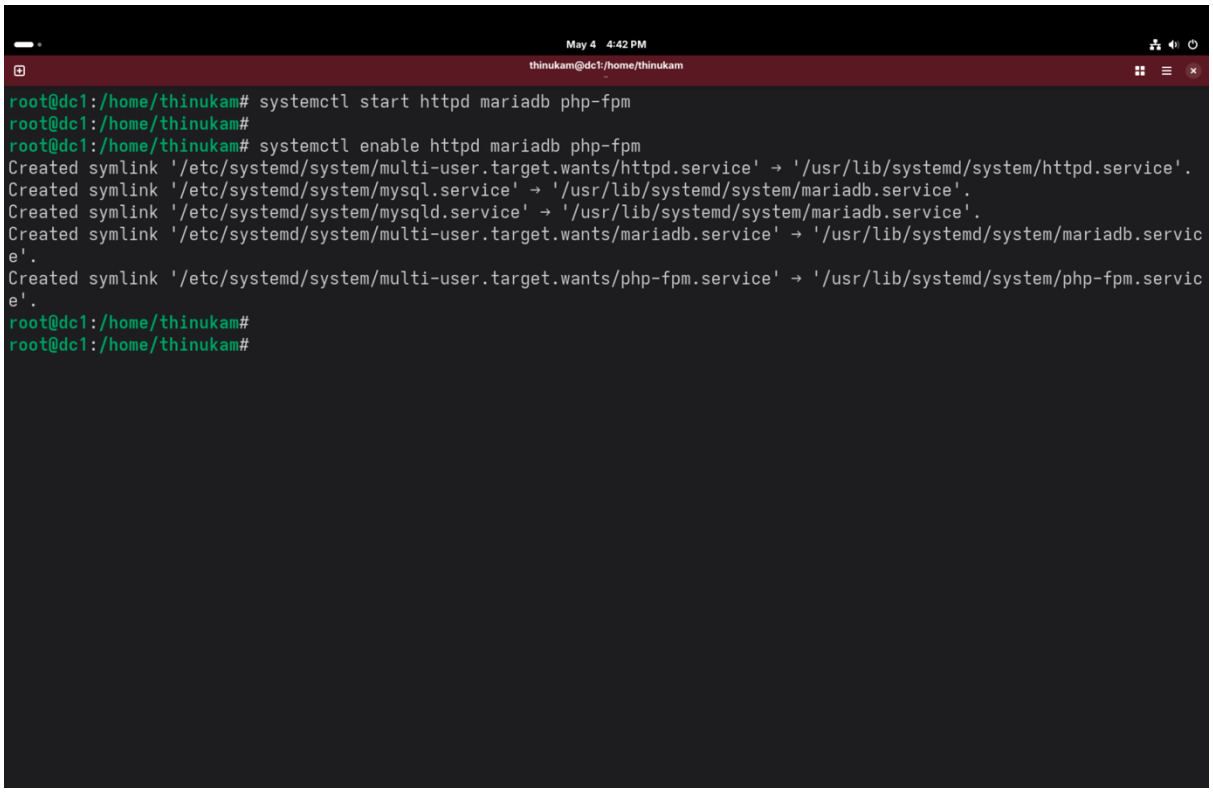

Figure 3.1. 10 - Start and enable services

3.2. Creating a Zabbix Database

- Create the Zabbix database as given
    - Connect to the MariaDB shell
    - Create a database and a user for Zabbix
    - Grant all the privileges to the Zabbix database
    - Flush the privileges and exit from the MariaDB shell

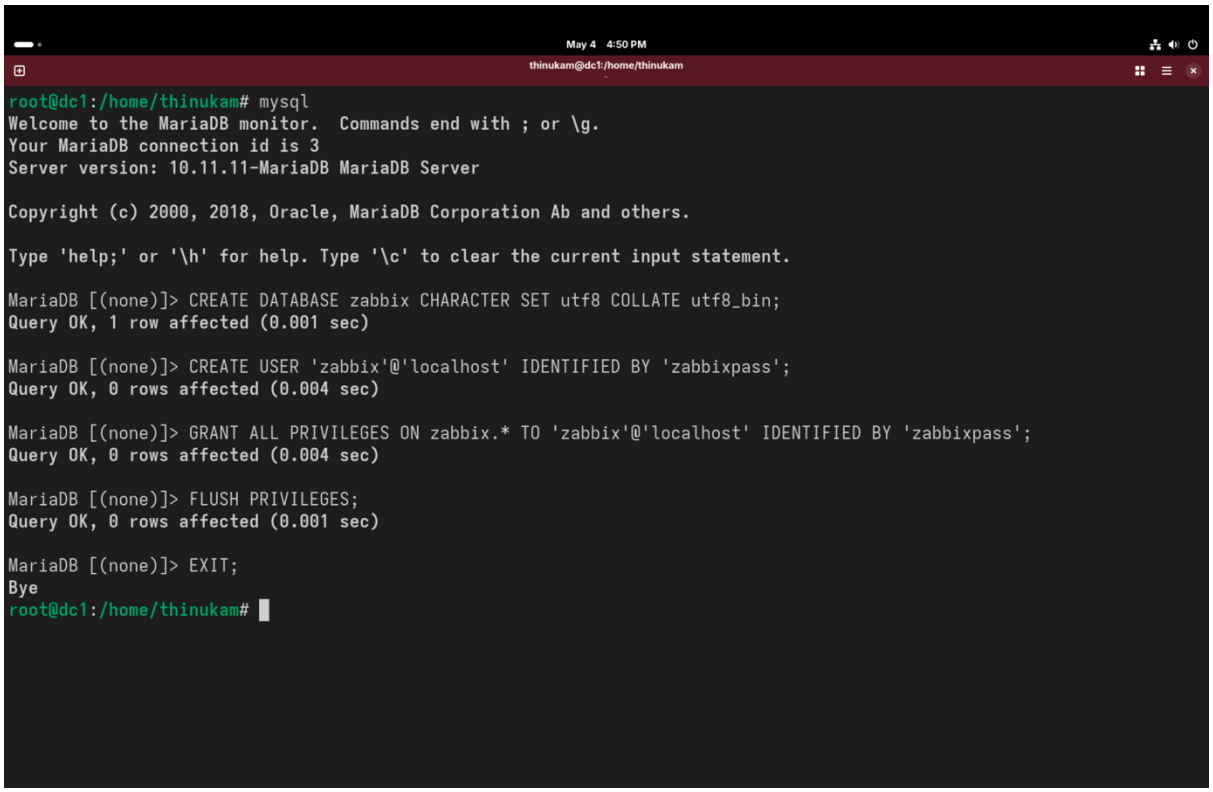

Figure 3.2. 1 - MariaDB shell for creating the database

3.3. Installing Zabbix Server

- Import a new Zabbix GPG key and install the Zabbix repository for the server

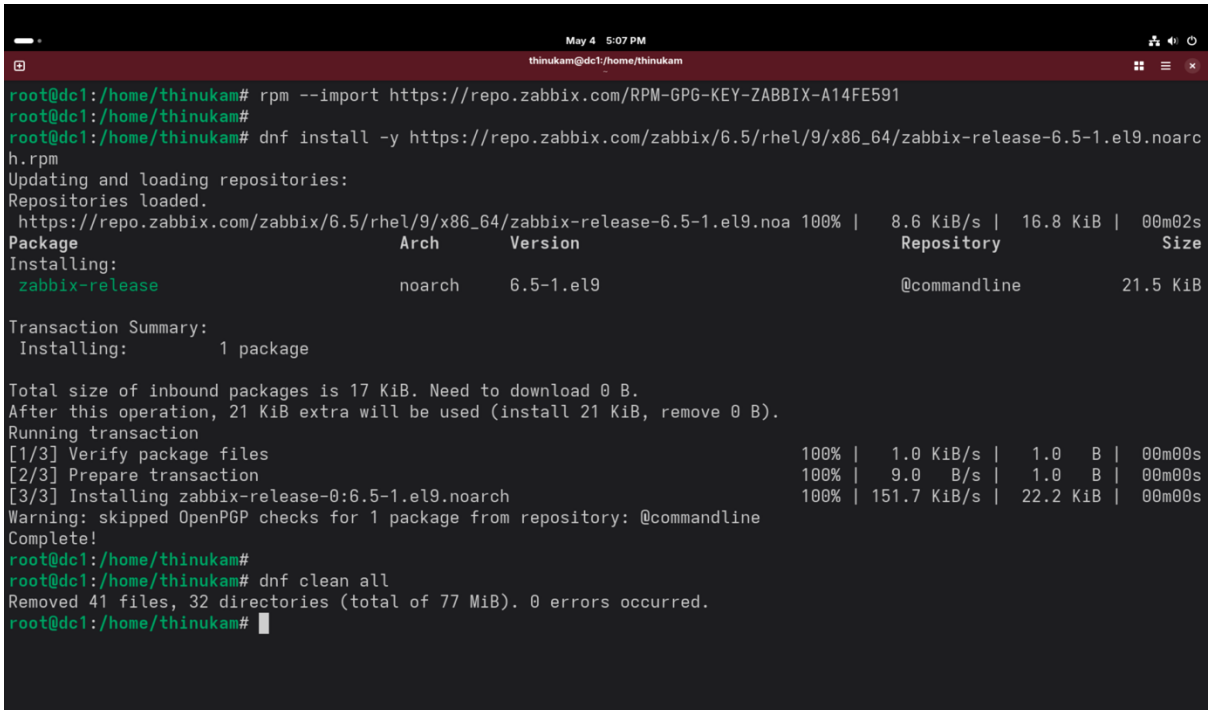

Figure 3.3. 1 - Install the Zabbix repo

- Install the packages for the Zabbix Server and Web Frontend with MySQL

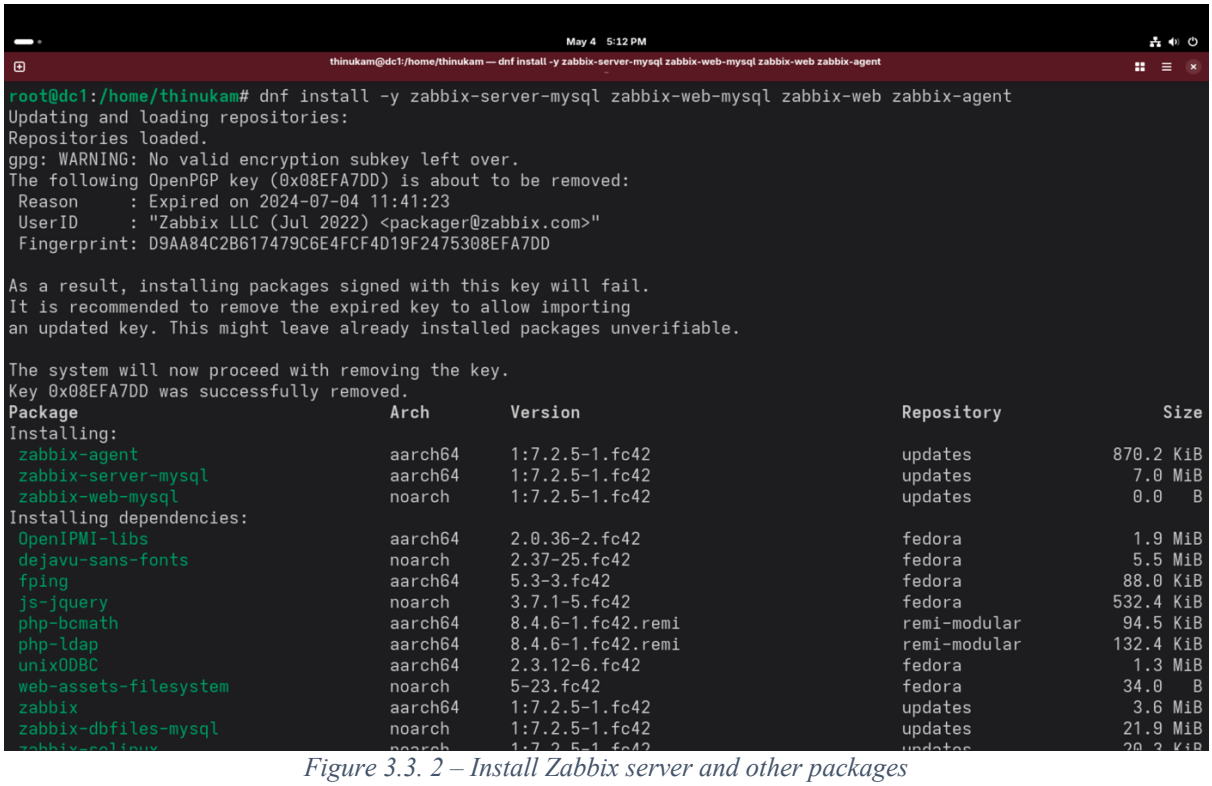

Figure 3.3. 2 – Install Zabbix server and other packages

- Download the Zabbix source from Zabbix Cloud Images and Appliances

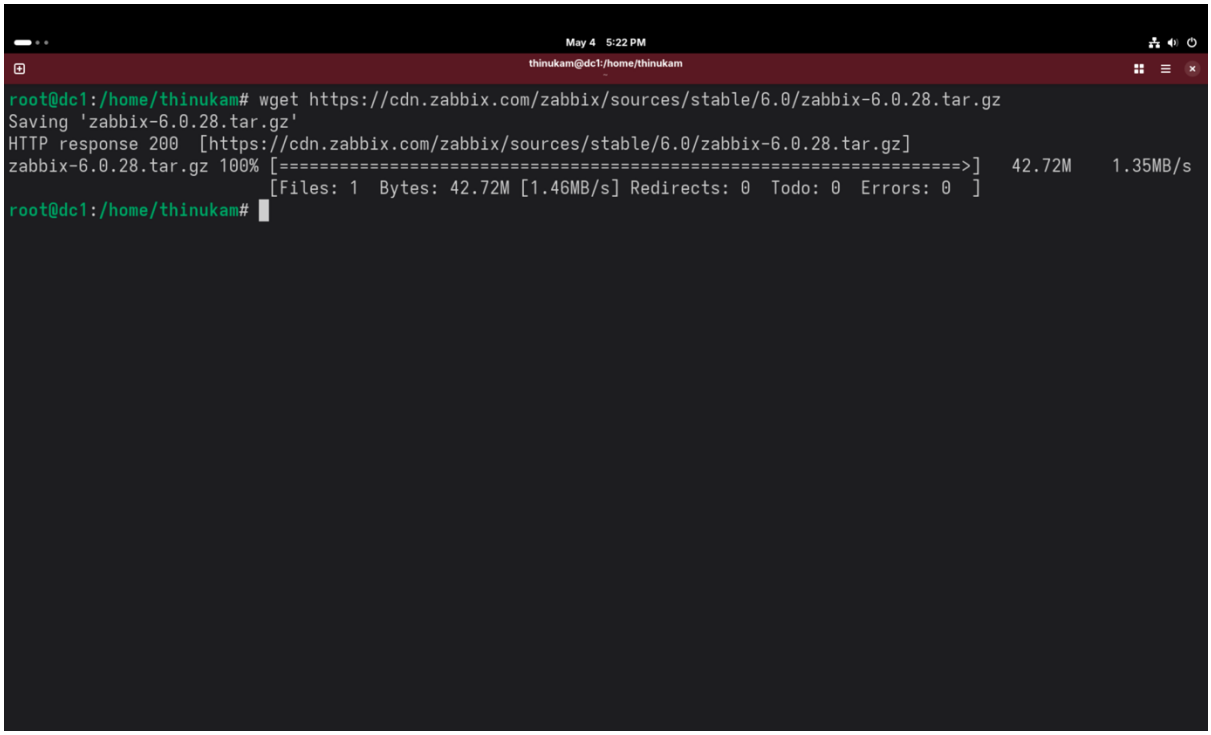

Figure 3.3. 3 - Download the Zabbix source

- Extract the downloaded Zabbix source files

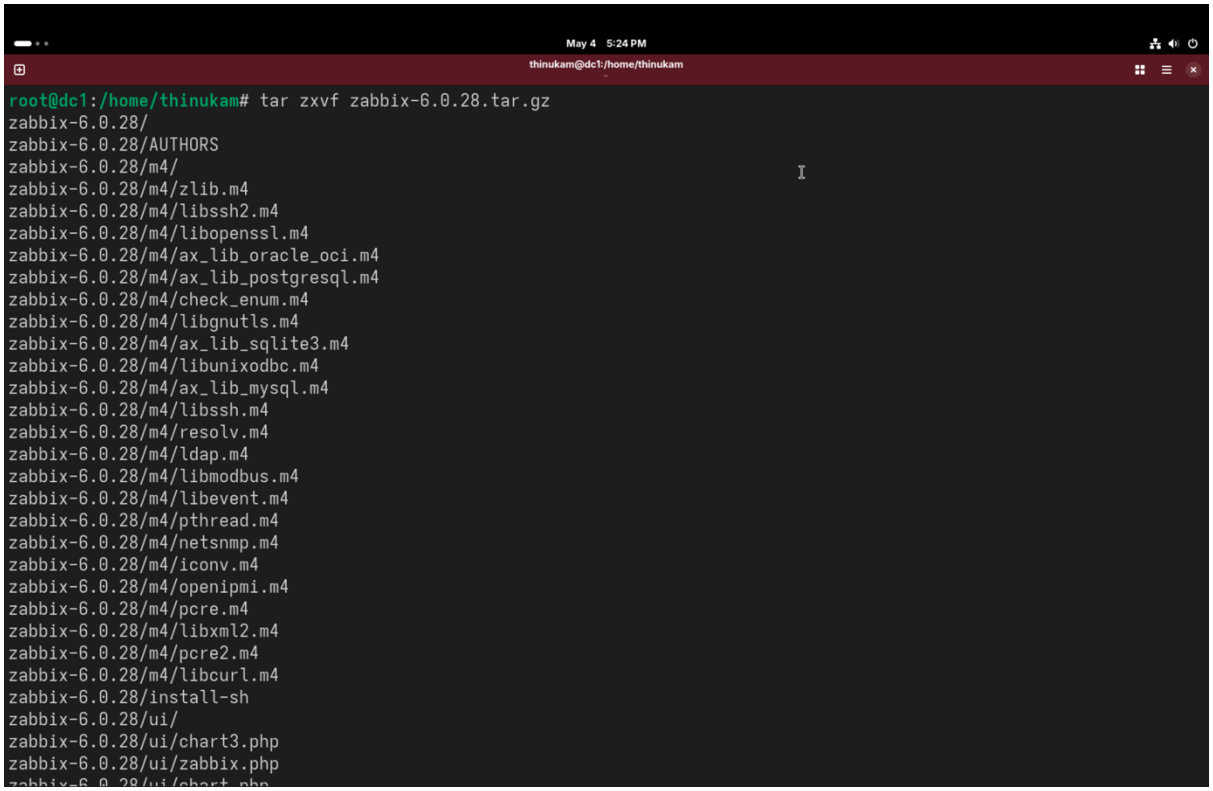

Figure 3.3. 4 - Extract the downloaded file

- Change the directory to the extracted directory and import the database schema, images, and data

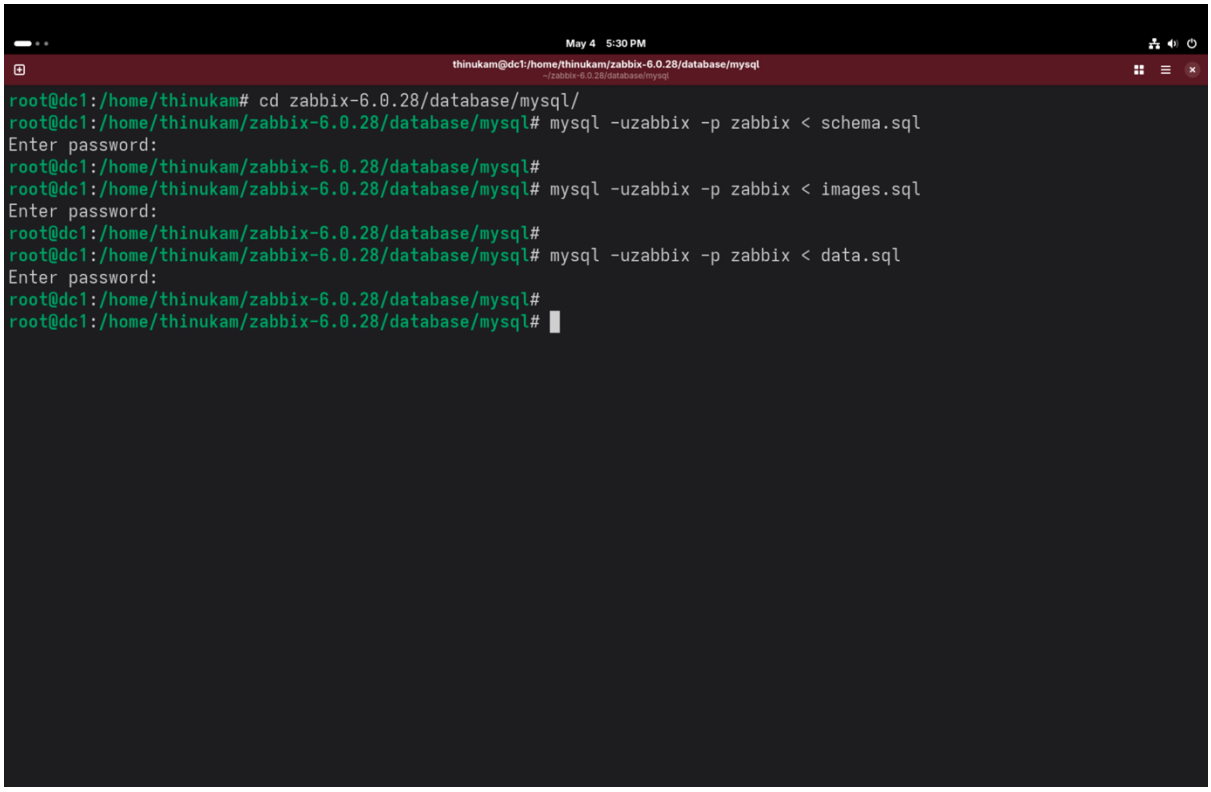

Figure 3.3. 5 - Import database schema, images, and data

- Modify the Zabbix configuration file */etc/zabbix_server.conf* to define the database as given
    - DBHost=localhost
    - DBName=zabbix
    - DBUser=zabbix
    - DBPassword=zabbixpass

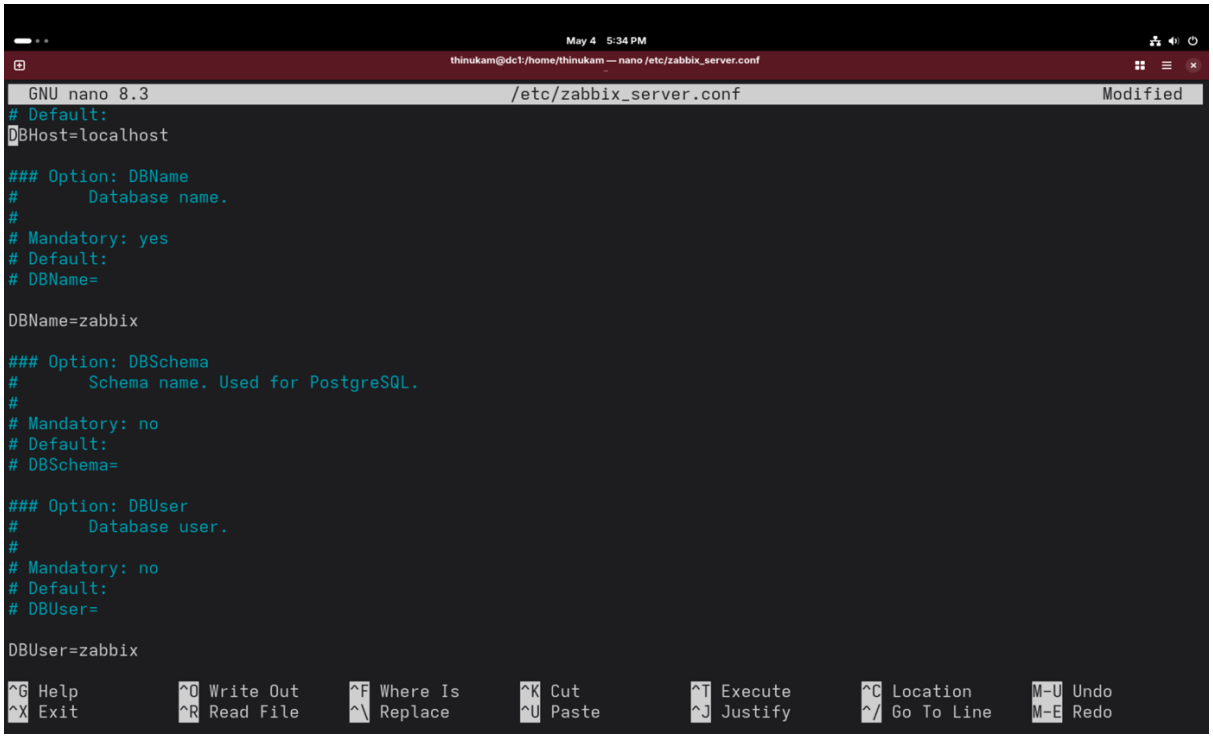

Figure 3.3. 6 - /etc/zabbix_server.conf file (i)

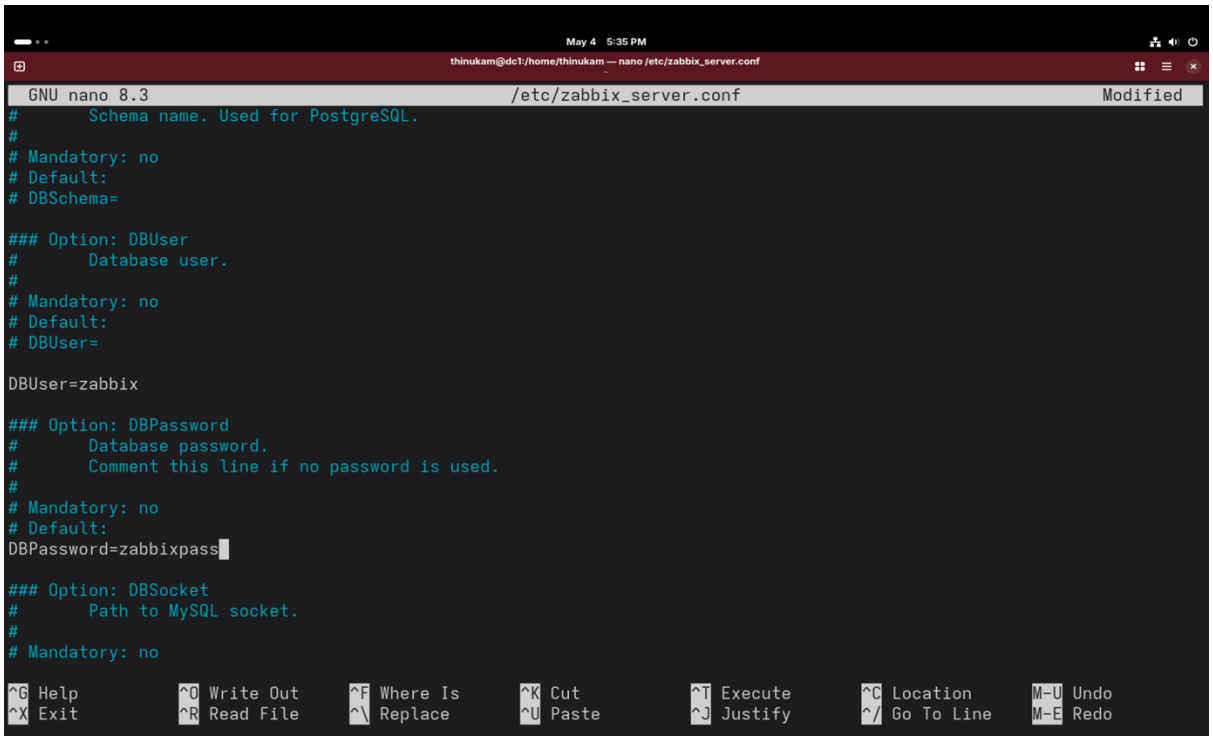

Figure 3.3. 7 - /etc/zabbix_server.conf file (ii)

- Restart all services required by Zabbix and enable them upon system initialization

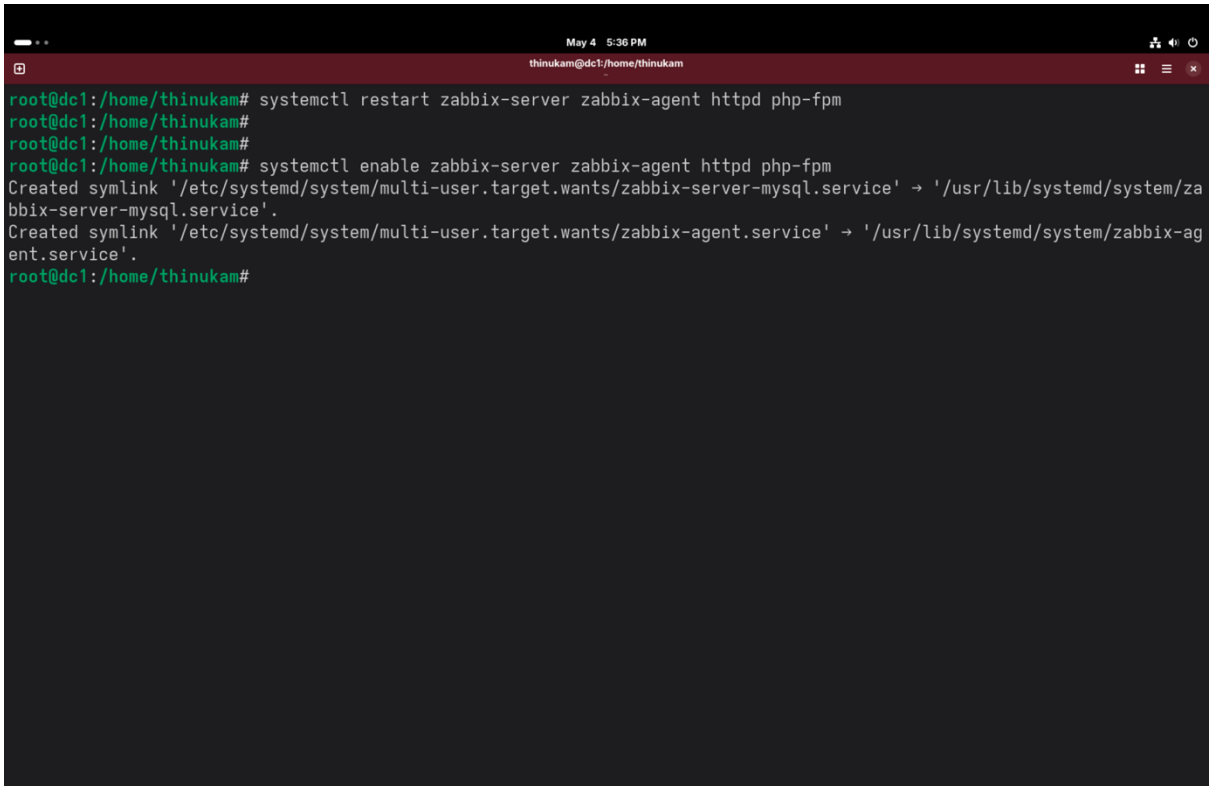

Figure 3.3. 8 - Restart and enable Zabbix services

3.4. Accessing Zabbix Web Installation

- Open the web browser and access the Zabbix web installation with the URL *http://{server-ip}/zabbix* and complete the installation

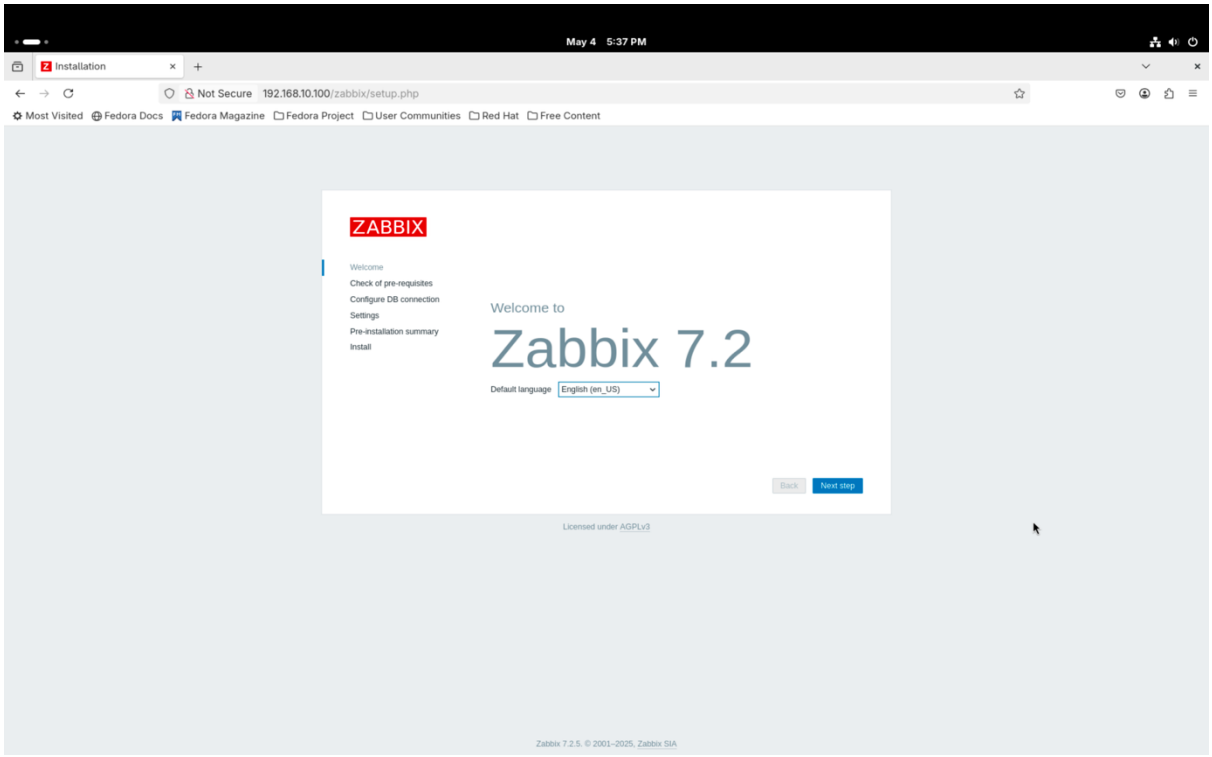

Figure 3.4. 1 - Zabbix Welcome screen

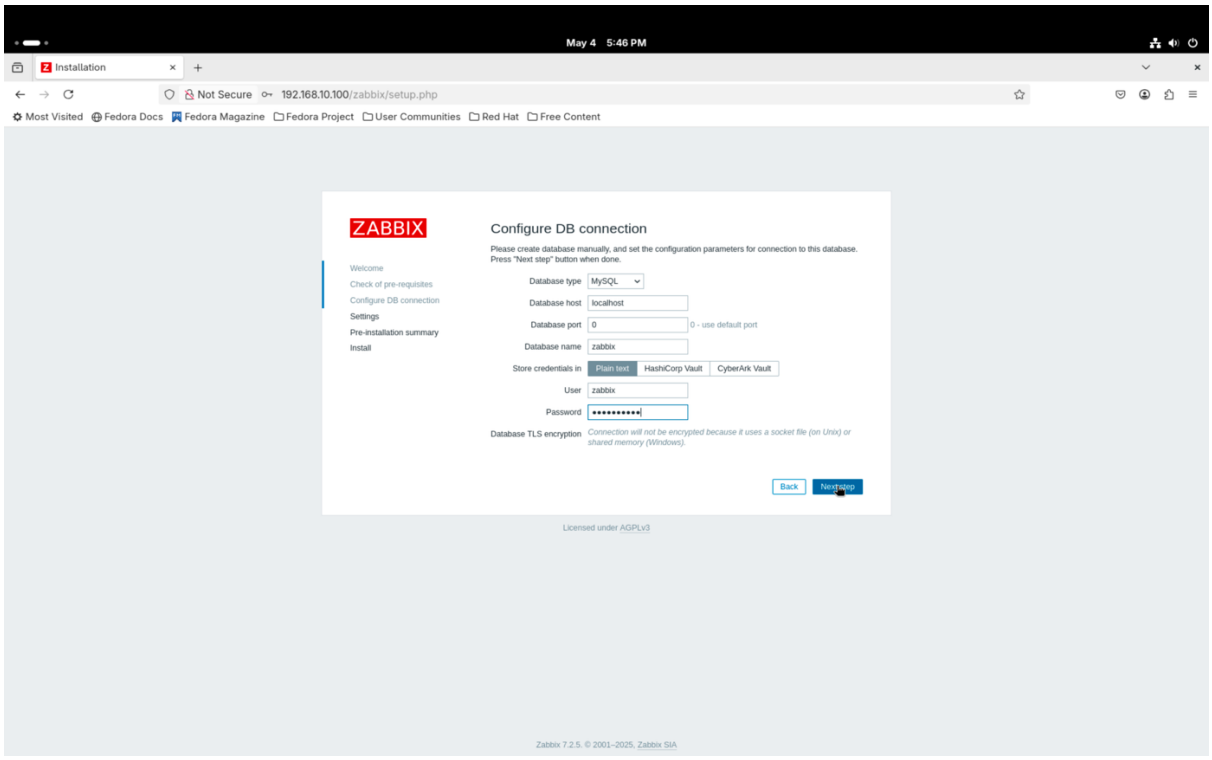

Figure 3.4. 2 - Zabbix database configuration screen

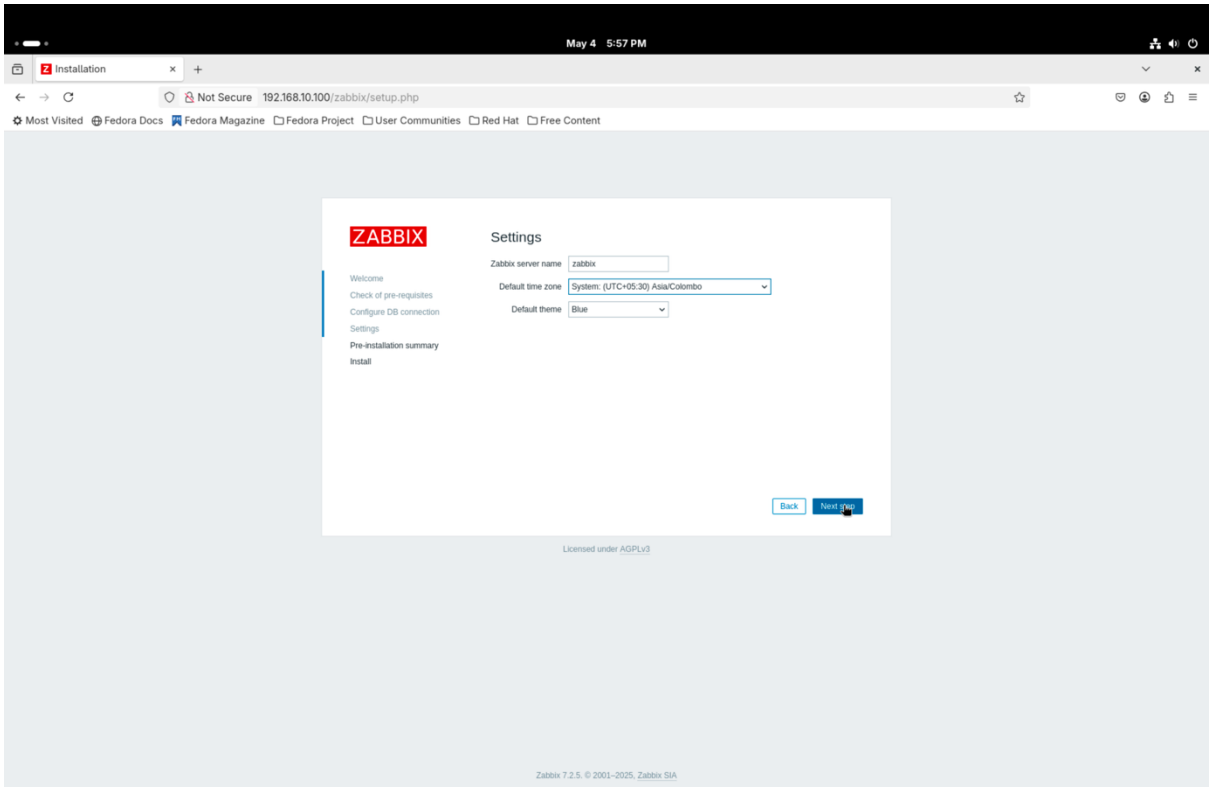

Figure 3.4. 3 - Zabbix server details screen

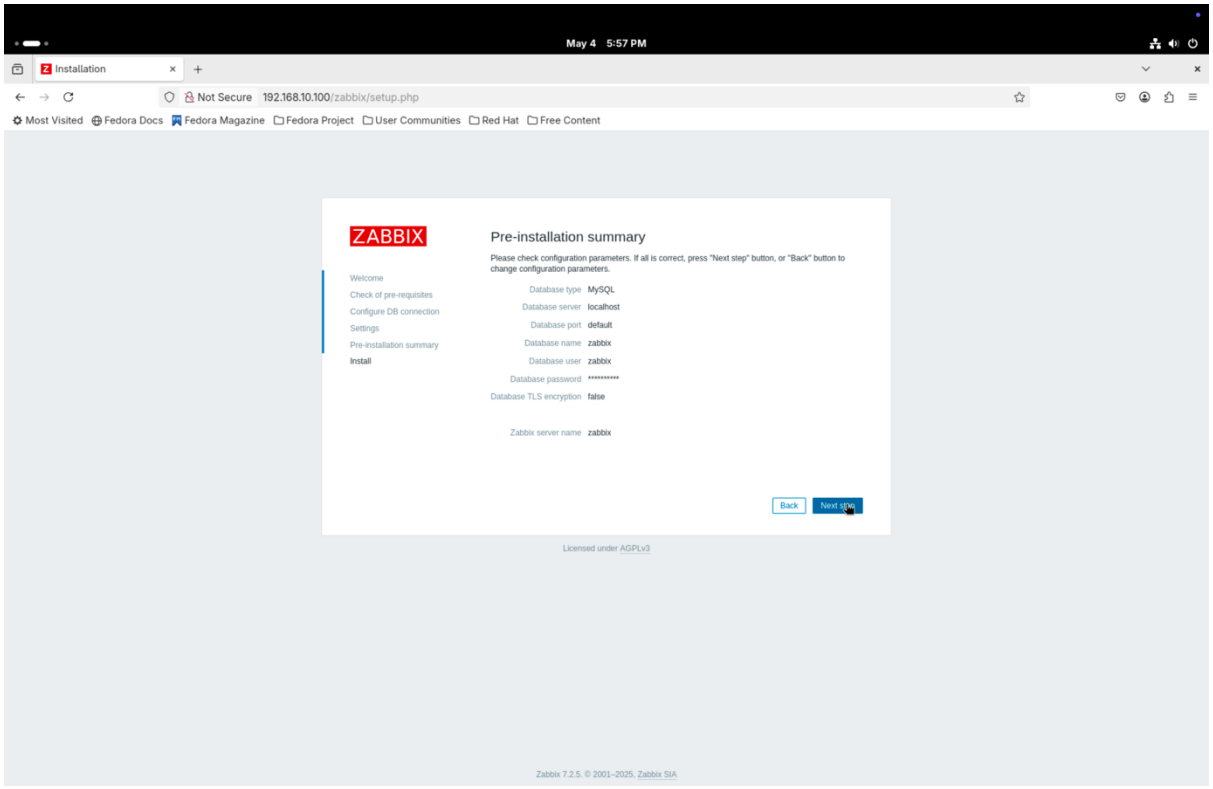

Figure 3.4. 4 - Zabbix web installation summary screen

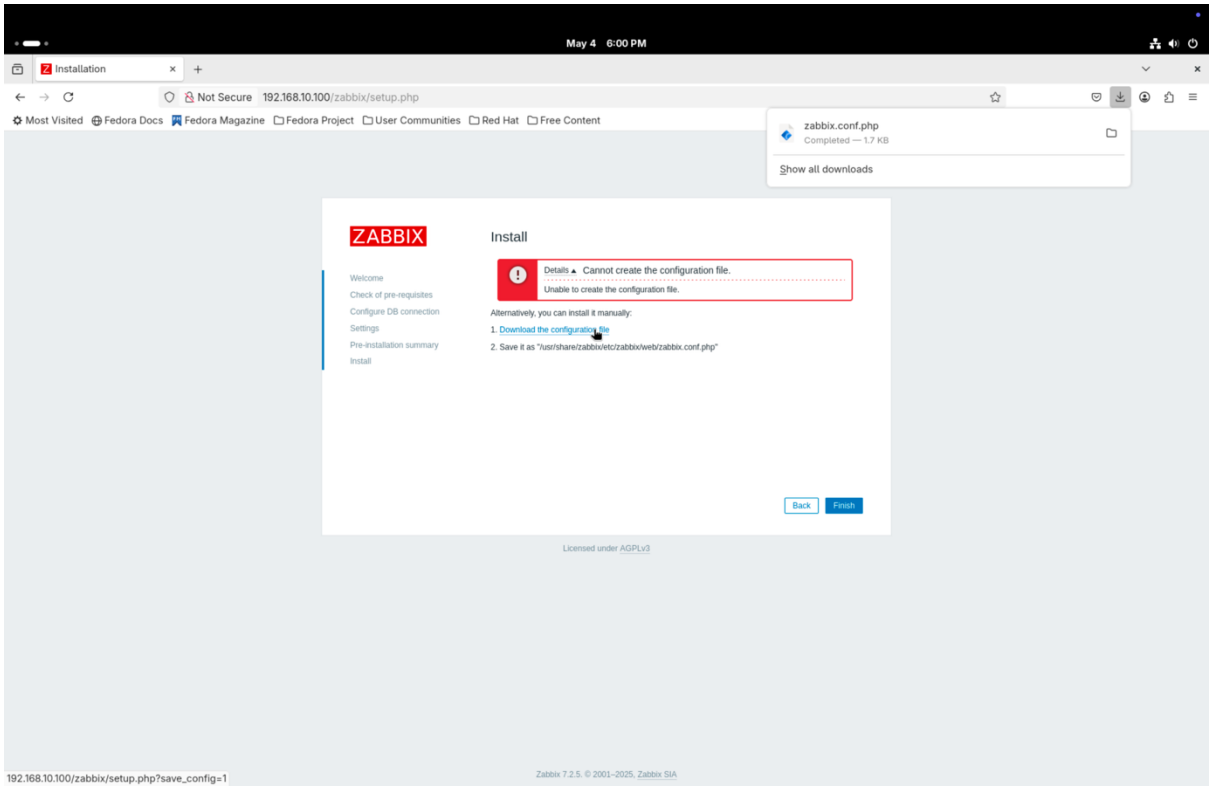

Figure 3.4. 5 - Configuration file download screen

- Download the Zabbix Web installation configuration file *(zabbix.conf.php)* manually from the given link
- Copy or move the downloaded configuration file to the directory */etc/zabbix/web*
- Click 'Finish' on the web

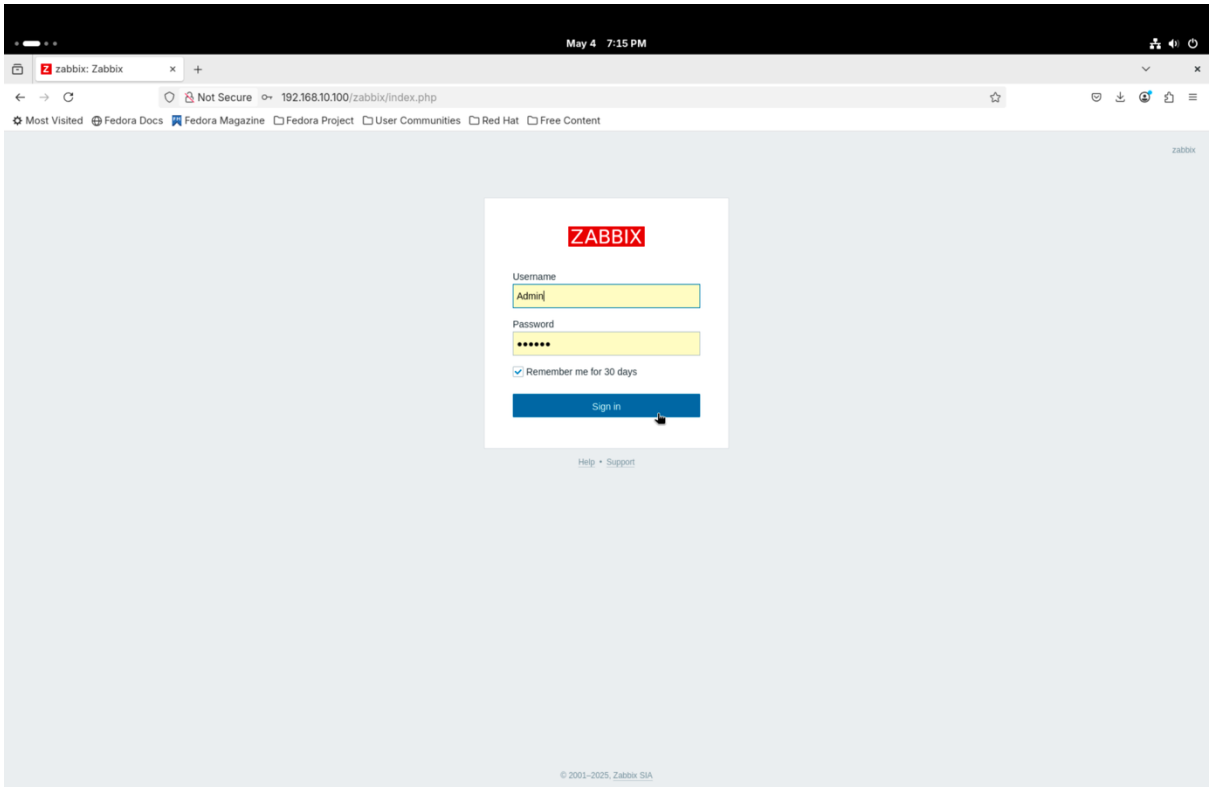

Figure 3.4. 6 - Zabbix login screen

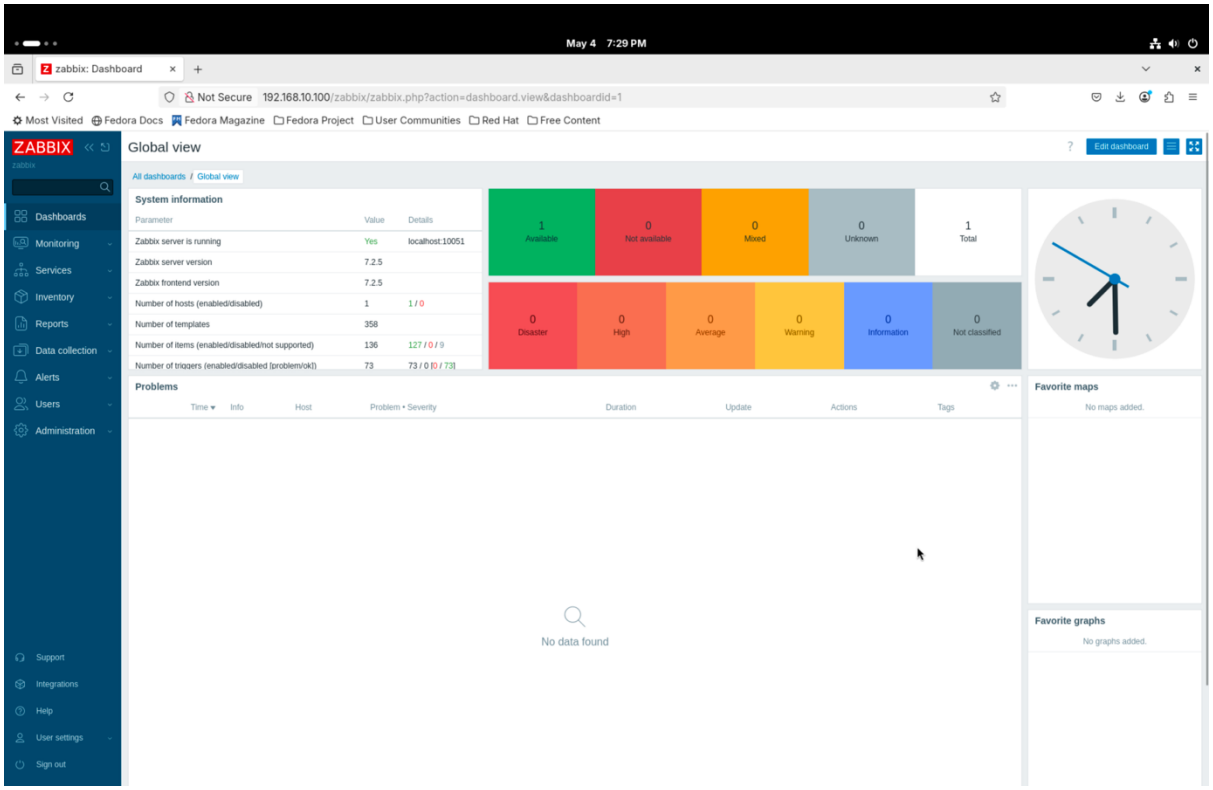

Figure 3.4. 7 - Initial Zabbix Dashboard

4. Monitoring Hosts with Zabbix

4.1. Setting up the host machine/s

- Install the required package/s for Zabbix Agent

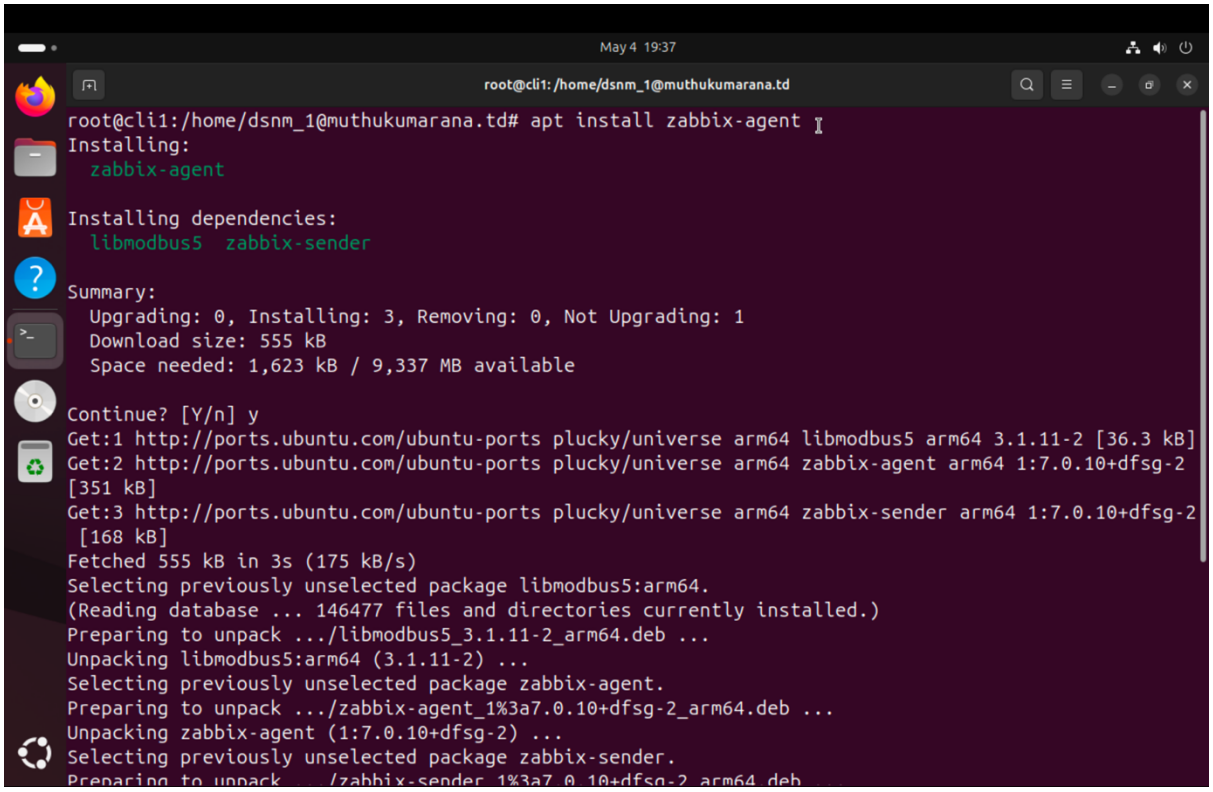

Figure 4.1. 1 - Install Zabbix Agent package

- Modify the Zabbix Agent configuration file */etc/zabbix/zabbix_agentd.conf* to define the Zabbix server

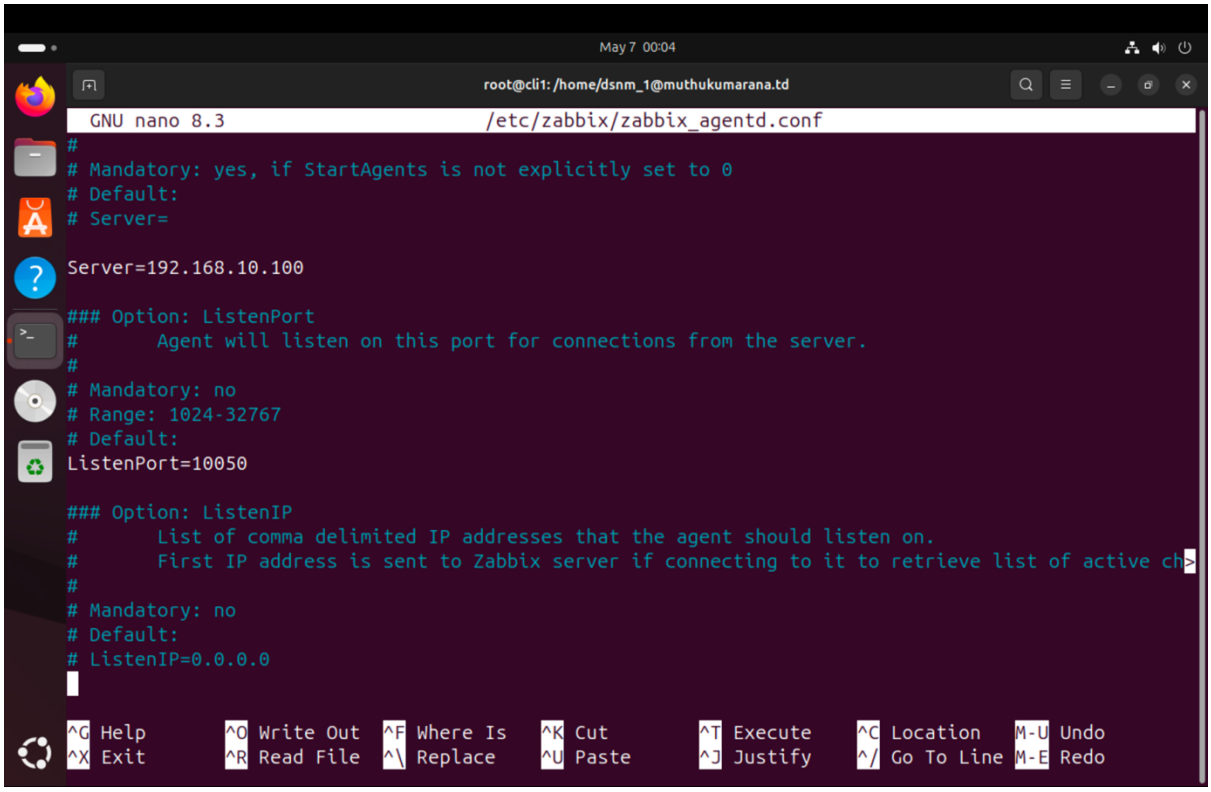

Figure 4.1. 2 - /etc/zabbix_agentd.conf file (i)

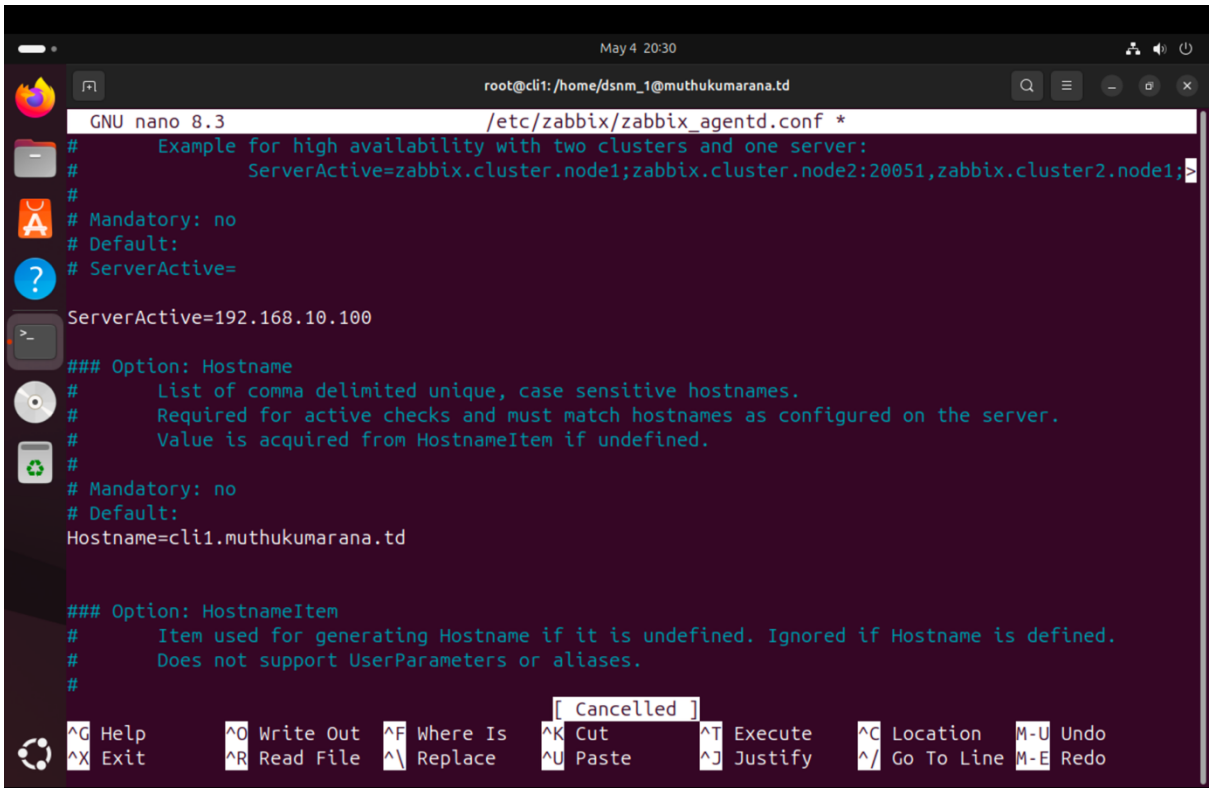

Figure 4.1. 3 - /etc/zabbix_agentd.conf file (ii)

- Start the Zabbix Agent service and enable it upon system initialization

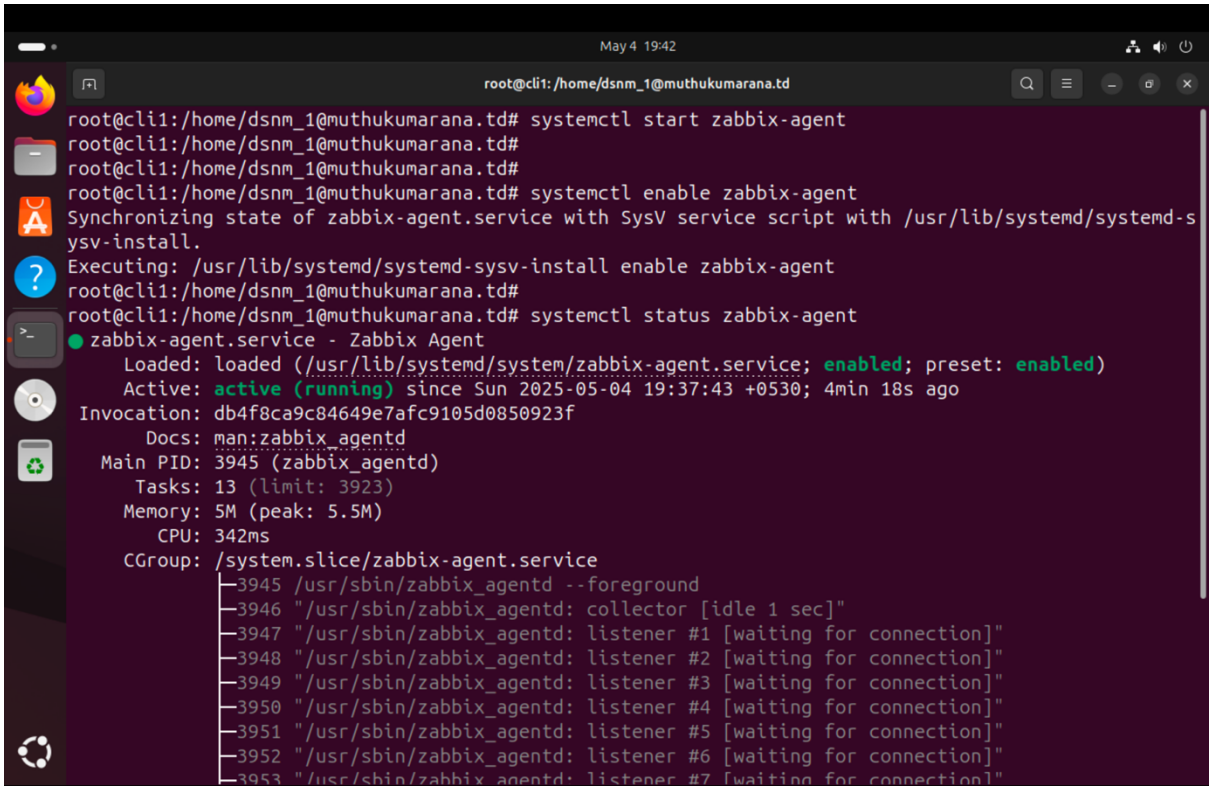

Figure 4.1. 4 - Start and enable the Zabbix Agent service

- Similar steps are followed to install and establish the Zabbix Agent on other hosts.

4.2. Adding host/s to the Zabbix Monitor

- Navigate to **Monitoring > Hosts > Create host**
- Define remote host details

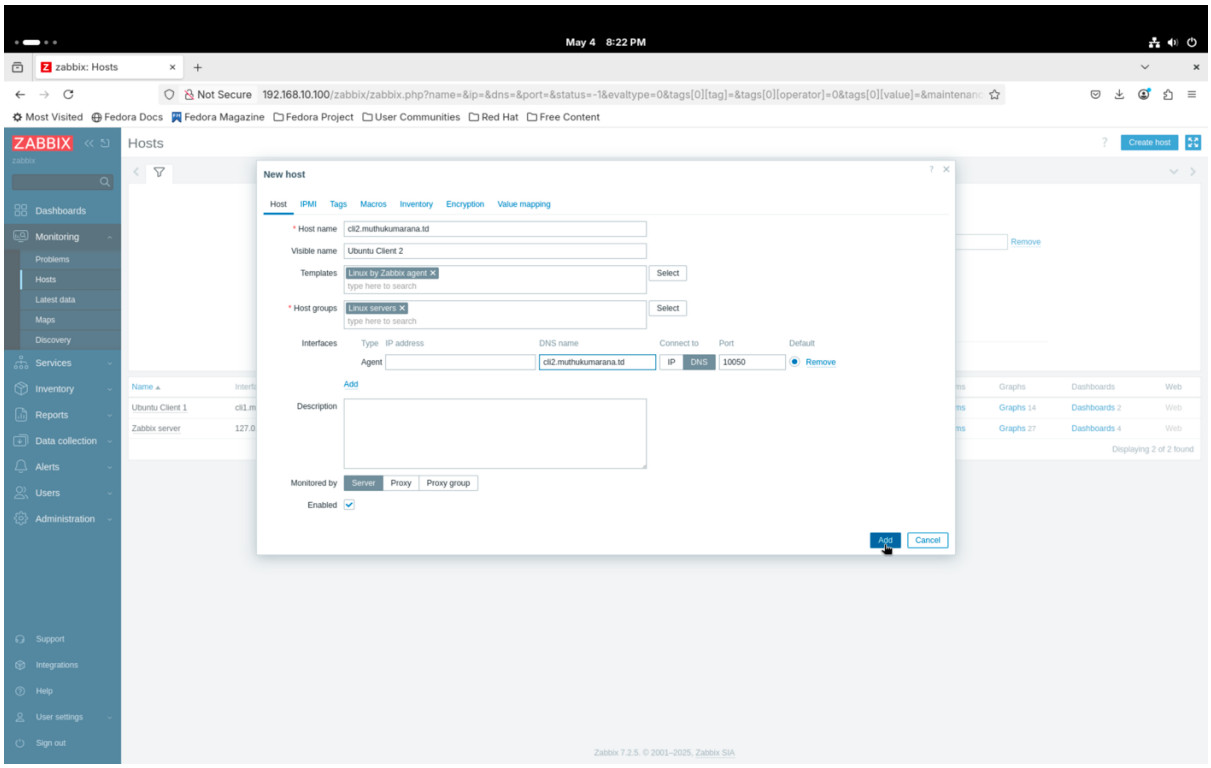

Figure 4.2. 1 - Adding a host to the Zabbix monitor

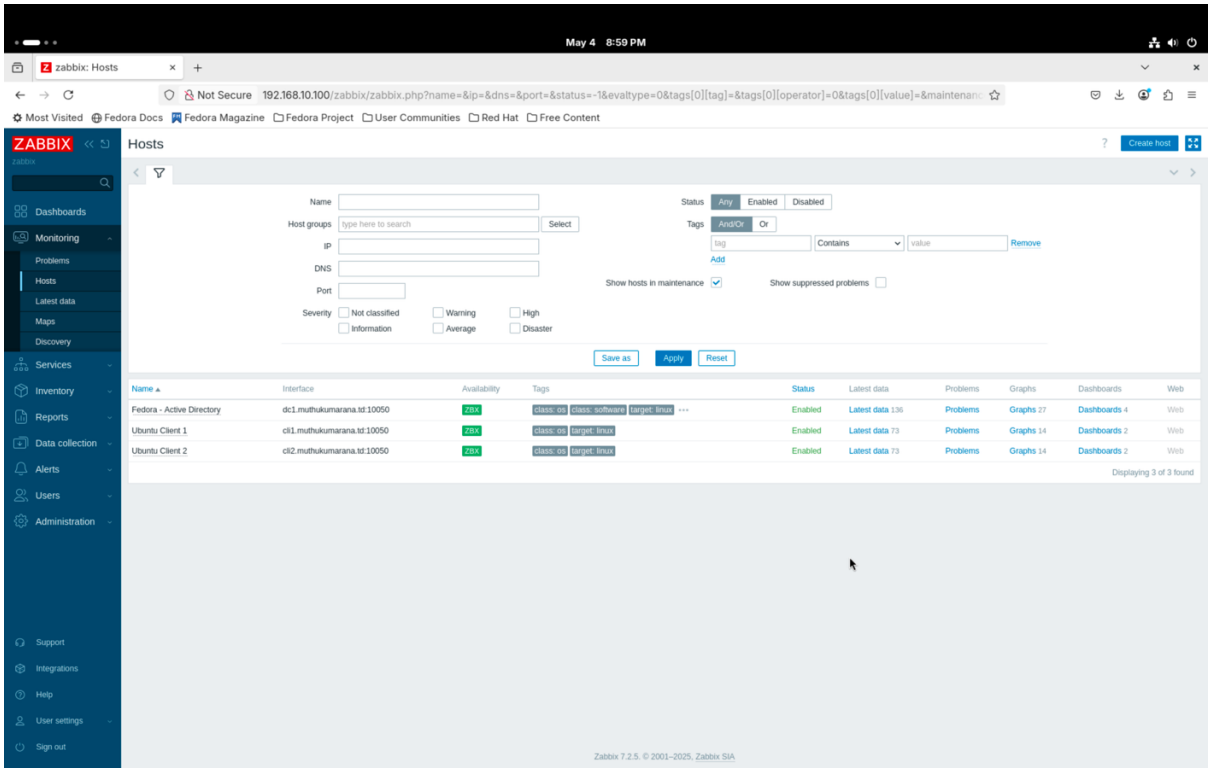

Figure 4.2. 2 - Added hosts to the Zabbix monitor

4.3. Customizing the dashboard to monitor critical services

- Navigate to **Dashboards > Edit dashboard**
- Customize the dashboard accordingly to monitor critical services such as CPU usage and utilization, Memory usage and utilization, Disk usage, and problems

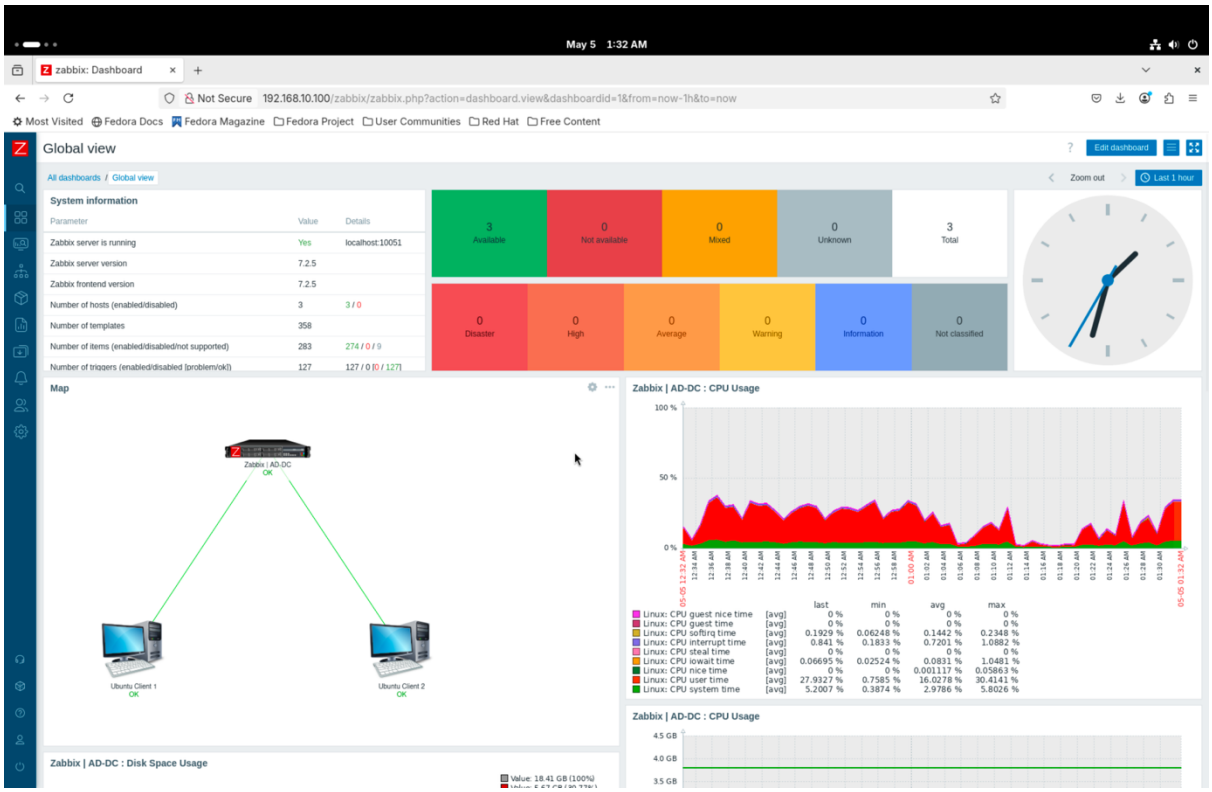

Figure 4.3. 1 - Customized Zabbix Monitoring dashboard (i)

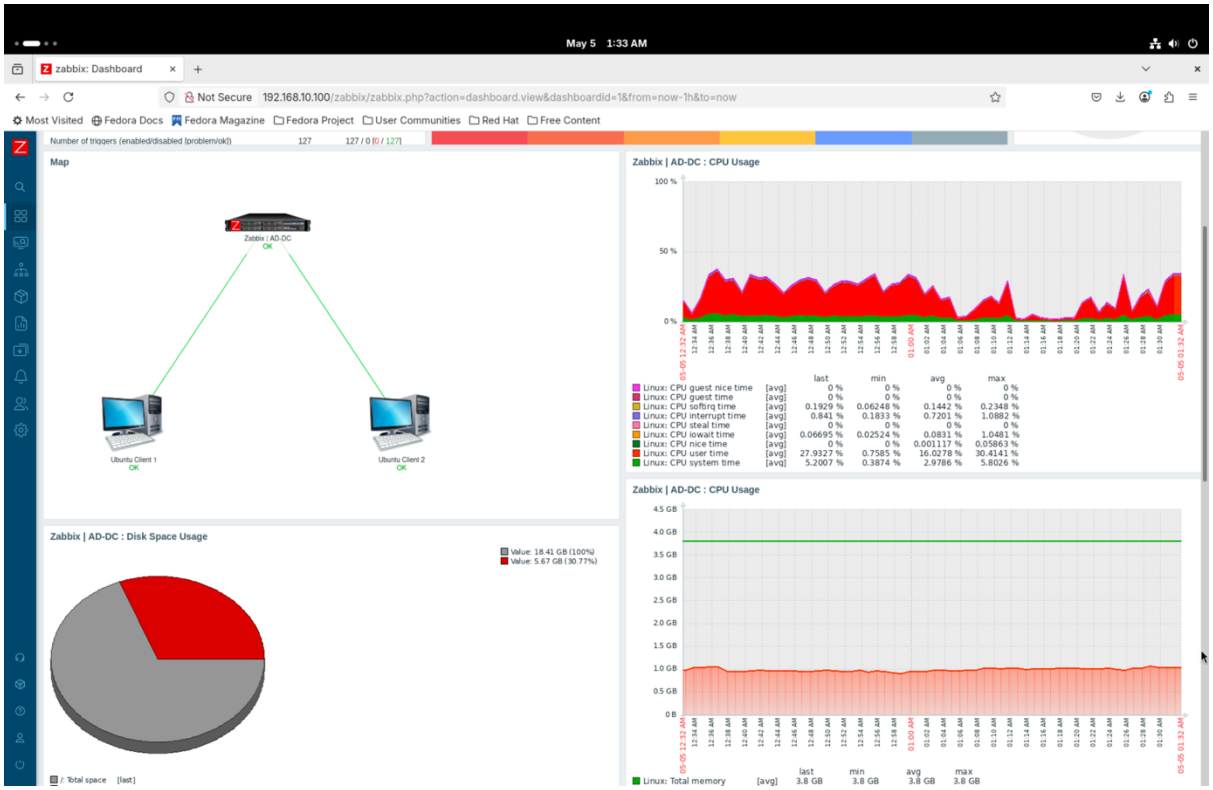

Figure 4.3. 2 - Customized Zabbix Monitoring dashboard (ii)

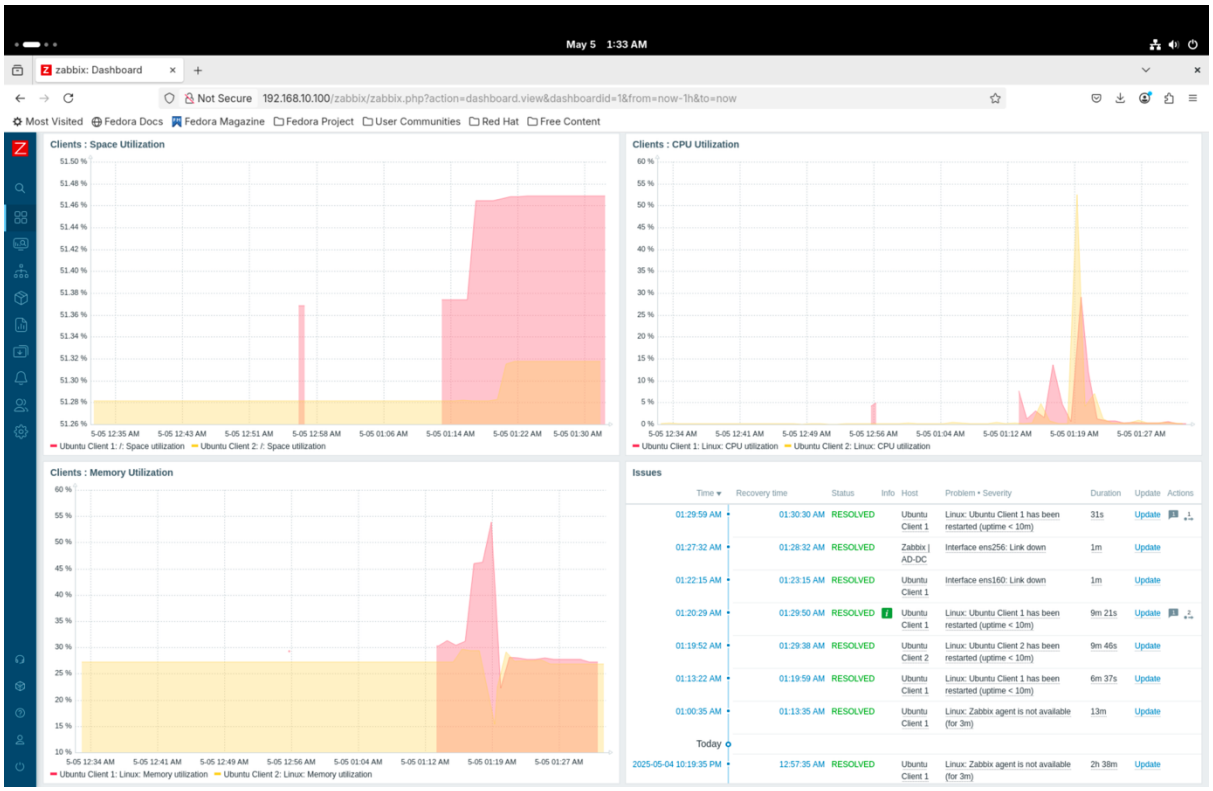

Figure 4.3. 3 - Customized Zabbix Monitoring dashboard (iii)

References

[1] https://fedoramagazine.org/samba-as-ad-and-domain-controller/

[2] https://documentation.ubuntu.com/server/how-to/sssd/with-active-directory/index.html

[3] https://www.atlantic.net/dedicated-server-hosting/how-to-install-zabbix-monitoring-tool-on-fedora/

[4] https://serverspace.io/support/help/installing-zabbix-agent-on-ubuntu/

[5] https://youtu.be/48v0jjrr36o