

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



ZABBIX

Assignment Overview

This assignment explores the real-world application of network management principles through the implementation of a Linux-based Domain Controller with integrated monitoring, aligning with the ISO Network Management Framework.

In this project, a server was configured using Samba to emulate Active Directory (AD) services, enabling centralized user authentication and domain management within a controlled network environment. The setup includes:

- Deployment of a Samba-based AD-compatible domain controller.
- Integration of Kerberos for secure authentication and DNS for name resolution.
- Configuration of at least one client machine (Linux or Windows) to join the domain and authenticate using AD credentials.
- Installation and full configuration of **Zabbix**, a powerful open-source monitoring solution, to oversee the health and performance of both server and client systems.

The Zabbix server was used to create a comprehensive monitoring dashboard, tracking key metrics such as CPU usage, memory performance, disk activity, network status, and the availability of core authentication services. This hands-on project not only reinforces theoretical knowledge but also demonstrates practical skills in network service configuration, domain management, system monitoring, and troubleshooting in a virtualized environment.

The result is a functional mini-enterprise environment that showcases the core competencies expected of a network and systems engineer, bridging the gap between academic learning and professional implementation.

Contents

1. Introduction.....	6
2. Resources	7
3. Setting Up the Domain Controller	8
4. Configuring the Client Machine	20
4.1 Windows Client	20
4.2 Fedora Client.....	26
5. Install and configure Zabbix	29
6. Install the Zabbix agent on the clients and connect to the Zabbix server	38
6.1 Connect the Windows client to Zabbix.....	38
6.2 Connect the Fedora client to Zabbix.....	39
7. Errors Occurred During the Configuration Process.....	40
8. References.....	43

Figure Of Table

Samba configurations

Figure 1.0.1-Edit the Network Configuration on the Ubuntu machine	8
Figure 1.0.2-Add the server to the host file	9
Figure 1.0.3-Verify Hostname	9
Figure 1.0.4-Disable the DNS Resolver	10
Figure 1.0.5-Unlink the config.....	10
Figure 1.0.6>Create our own Resolv.conf.....	11
Figure 1.0.7-Edit the newly created resolver config file	11
Figure 1.0.8-Make the file immutable	12
Figure 1.0.9-Install the Samba	12
Figure 1.0.10-Disable Samba Services	13
Figure 1.0.11-Active samba-ad-dc.....	13
Figure 1.0.12-Back up the original smb.conf file	14
Figure 1.0.13-Provision the Active Directory.....	14
Figure 1.0.14-Backup and replace the Kerberos Config.....	15
Figure 1.0.15-Start the samba-ad-dc service and check the status	15
Figure 1.0.16-Set permissions.....	16
Figure 1.0.17-Update Chrony.conf	16

Figure 1.0.18-Restart Chronyd and get its status.....	17
Figure 1.0.19-Verifying Samba Active Directory	18
Figure 1.0.20-Verify the Kerberos and ldap services	18
Figure 1.0.21-Authenticate to Kerberos using an administrator.....	19
Figure 1.0.22-verify list cached Kerberos tickets	19
Figure 2.0.1 Edit network Configurations in the Windows machine	20
Figure 2.0.2-Verify the DNS resolver.....	21
Figure 2.0.3-Ping AD domain dc1.rangana.ma and rangana.ma	21
Figure 2.0.4-Add Windows 10 to Active Directory	22
Figure 2.0.5-Logging in with the administrator account	22
Figure 2.0.6-Download RSAT to administer the domain from Windows.....	23
Figure 2.0.7-AD user and computers.....	23
Figure 2.0.8-User's window	24
Figure 2.0.9>Create a new user called DSNM	24
Figure 2.0.10-Logged in DSNM user Account.....	25
Figure 2.0.11-Change hostname	26
Figure 2.0.12-Modify resolved information	26
Figure 2.0.13-Restart systemd-resolved	27
Figure 2.0.14-Ping the Samba domain.....	27
Figure 2.0.15-Ping dc1.rangana.ma	27
Figure 2.0.16-Discover the Domain.....	28
Figure 2.0.17-Join the Domain	28
Figure 2.0.18-Log in with the created demo user in Fedora	28
Figure 3.0.1-Downloads the latest Zabbix repository .deb package for Ubuntu 22.04	29
Figure 3.0.2-Installs the downloaded .deb file, which adds the Zabbix APT repository.....	29
Figure 3.0.3-Install Zabbix server, frontend, agent	30
Figure 3.0.4-Install Mysql server	30
Figure 3.0.5-mysql status	31
Figure 3.0.6-Loggin mysql server as root user	32
Figure 3.0.7-Creates a new database named zabbix	32
Figure 3.0.8-Creates a new user called zabbix.	32
Figure 3.0.9-Gives the Zabbix user full access to the Zabbix database only.....	33
Figure 3.0.10-Create stored functions.....	33
Figure 3.0.11>Show databases	33
Figure 3.0.12-Import the Zabbix database structure and data into the Mysql database	34
Figure 3.0.13-Disable the log_bin_trust_function_creators option after importing the database schema.....	34
Figure 3.0.14-Edit file zabbix_server.conf	35
Figure 3.0.15-Start Zabbix server and agent processes	35
Figure 3.0.16-Search https://dc1/zabbix	36

Figure 3.0.17-Configure Zabbix username, password and port.....	36
Figure 3.0.18-Successfully installed Zabbix server.....	37
Figure 3.0.19-Create a new host using the Zabbix server.....	37
Figure 4.0.1-Add Zabbix agent.....	38
Figure 4.0.2-Give the firewall access to Zabbix agent	38
Figure 4.0.3-Added Windows client to the Zabbix server.....	39
Figure 4.0.4-Install Zabbix agent.....	39
Figure 4.0.5-start and enable the agent	40
Figure 5.0.1-Issue with PowerShell syntax or session permission in Windows.....	40
Figure 5.0.2-Try to fix the Zabbix agent.....	41
Figure 5.3-Zabbix Agent on Fedora machine is not responding to server messages	42

1. Introduction

In modern network environments, centralized authentication and system monitoring are essential for maintaining security, efficiency and reliability. This project focuses on implementing a Linux-based Domain Controller using Samba to simulate Active Directory (AD) functionalities, enabling centralized user management and domain-based authentication within a local network. In addition to setting up the domain controller, the assignment involves configuring at least one client machine to join the domain and authenticate via AD user credentials.

To complement this setup, the project also integrates Zabbix, an open-source monitoring solution, to ensure visibility into network performance and system health. By combining Active Directory services with network monitoring on a single Linux server, this implementation provides a practical and efficient approach to managing and monitoring small to medium-scale network environments.

This assignment is divided into 3 major parts,

1. Set up a Linux-based domain controller using Samba to implement Active Directory services.
2. Configure a client machine to join the domain and authenticate using the Active Directory user accounts.
3. Install and configure Zabbix to monitor the server and client machine.

2. Resources

- **Operating System:** Ubuntu 22.04 – for Server
 - Windows 10 Pro – Client
 - Fedora -Client
- **Services:**
 - Samba – to provide Active Directory Domain Controller functionality.
 - Kerberos – for secure authentication within the domain.
- **Monitoring Tool:**
 - Zabbix Server – installed on the same server as the domain controller to monitor system and network performance.
- **Client Machine:**
 - A separate virtual machine running Fedora and Windows 10, joined to the domain for testing authentication and integration.

3. Setting Up the Domain Controller

Configured a Linux-based Domain Controller using Samba to provide Active Directory services for centralized authentication and domain management.

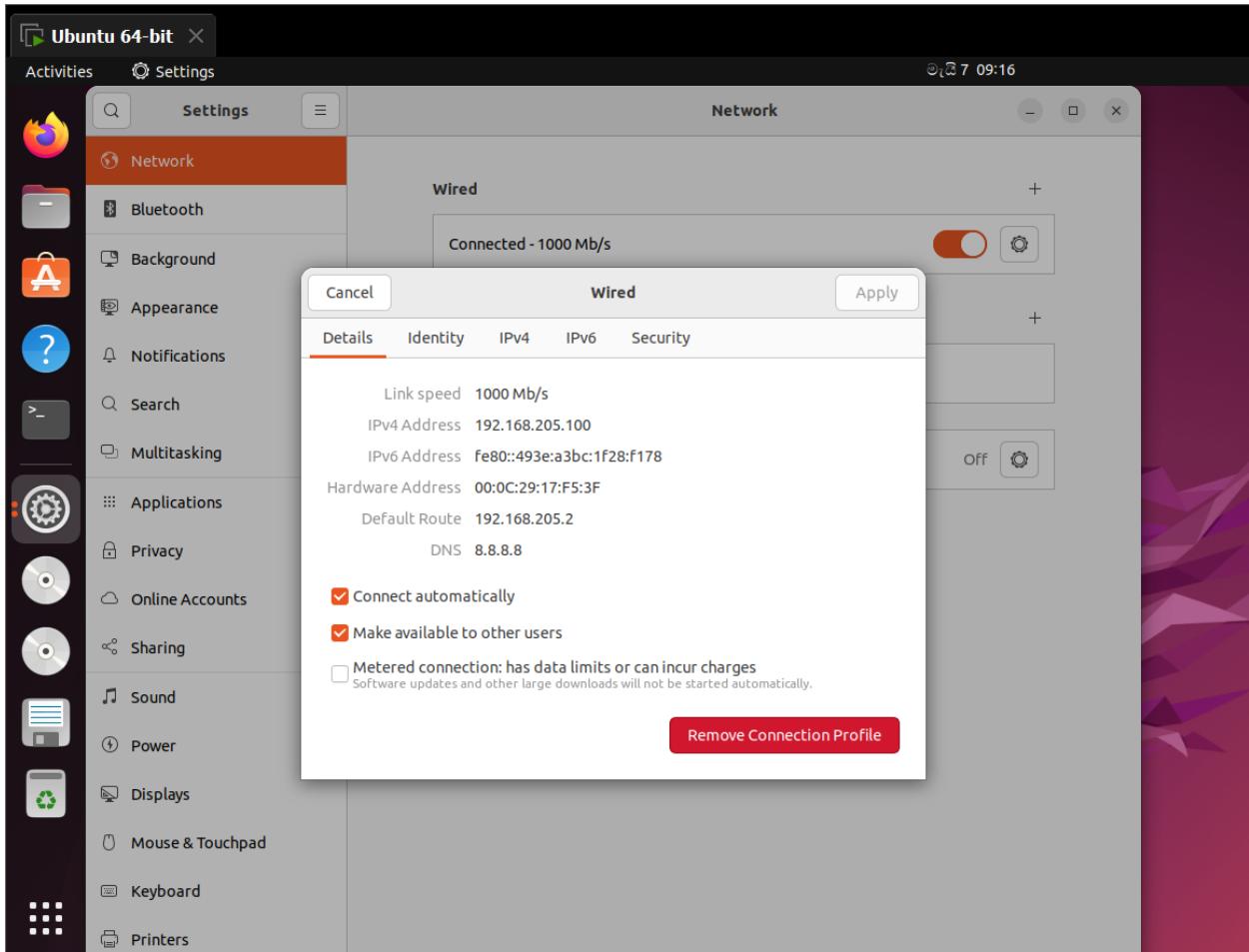
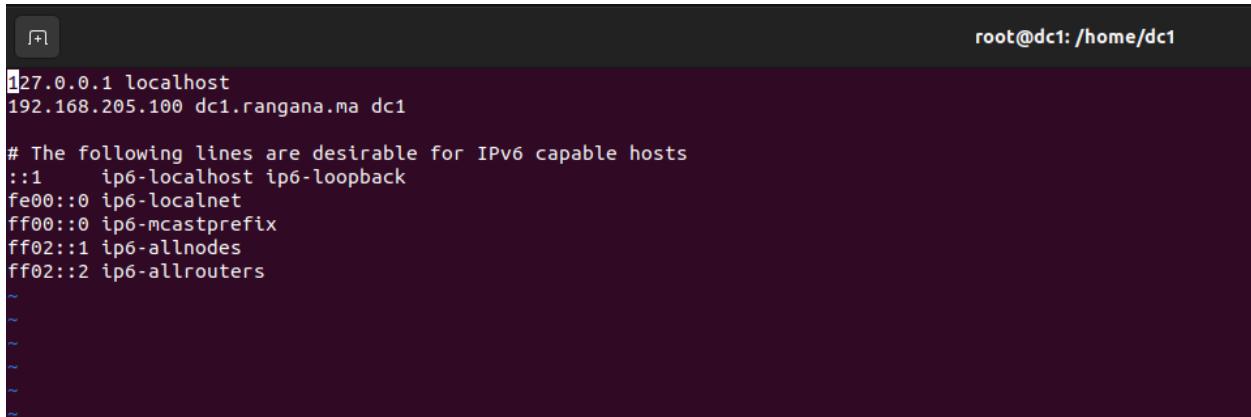


Figure 1.0.1-Edit the Network Configuration on the Ubuntu machine

Active Directory requires a stable and consistent network identity (IP address) for reliable domain services, client communication, and DNS resolution.

- A dynamic IP (from DHCP) can change after a reboot or lease renewal.
- If the IP changes, domain clients won't be able to find or trust the Domain Controller.
- A static IP ensures the DC is always reachable at the same address.

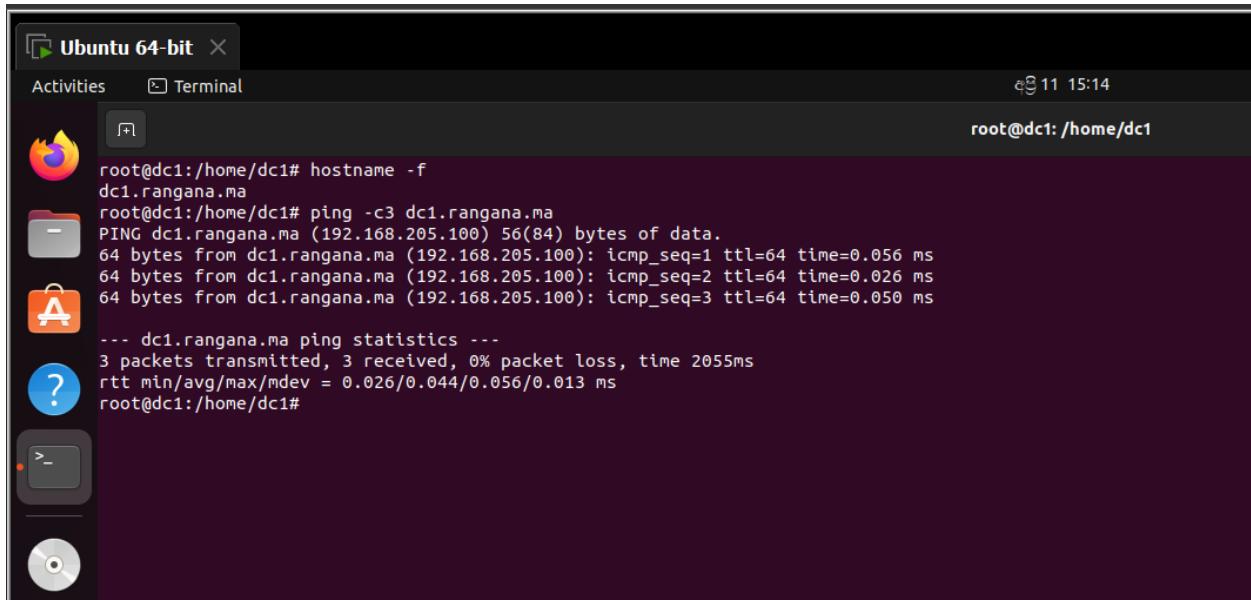


```
root@dc1: /home/dc1
127.0.0.1 localhost
192.168.205.100 dc1.rangana.ma dc1

# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Figure 1.0.2-Add the server to the host file

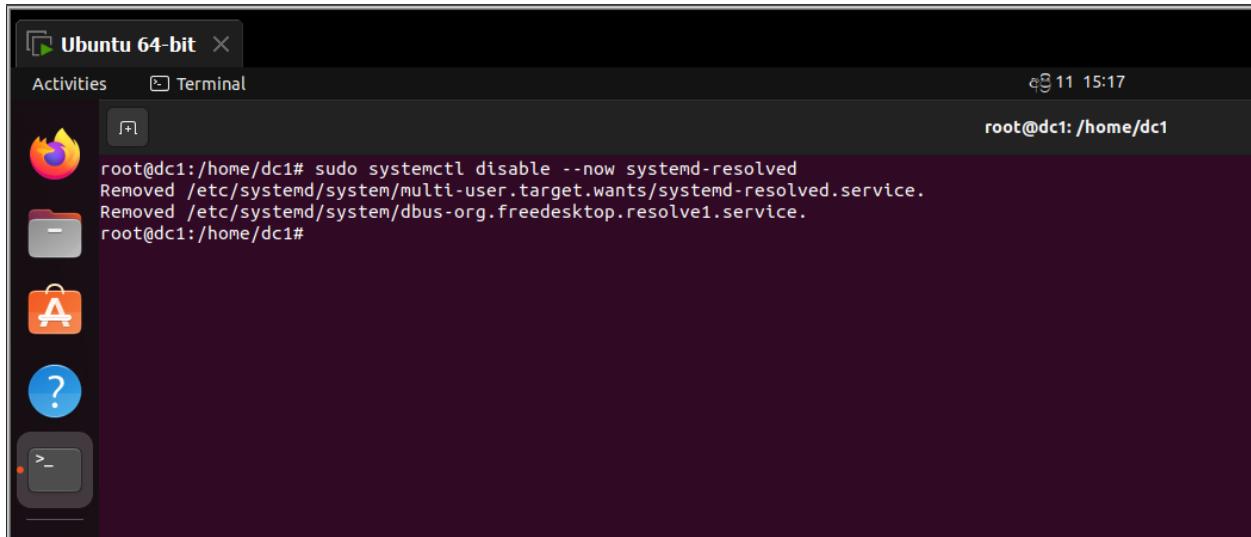
Add the Samba server IP and domain into the host file(/etc/hosts)



```
root@dc1:/home/dc1# hostname -f
dc1.rangana.ma
root@dc1:/home/dc1# ping -c3 dc1.rangana.ma
PING dc1.rangana.ma (192.168.205.100) 56(84) bytes of data.
64 bytes from dc1.rangana.ma (192.168.205.100): icmp_seq=1 ttl=64 time=0.056 ms
64 bytes from dc1.rangana.ma (192.168.205.100): icmp_seq=2 ttl=64 time=0.026 ms
64 bytes from dc1.rangana.ma (192.168.205.100): icmp_seq=3 ttl=64 time=0.050 ms
--- dc1.rangana.ma ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2055ms
rtt min/avg/max/mdev = 0.026/0.044/0.056/0.013 ms
root@dc1:/home/dc1#
```

Figure 1.0.3-Verify Hostname

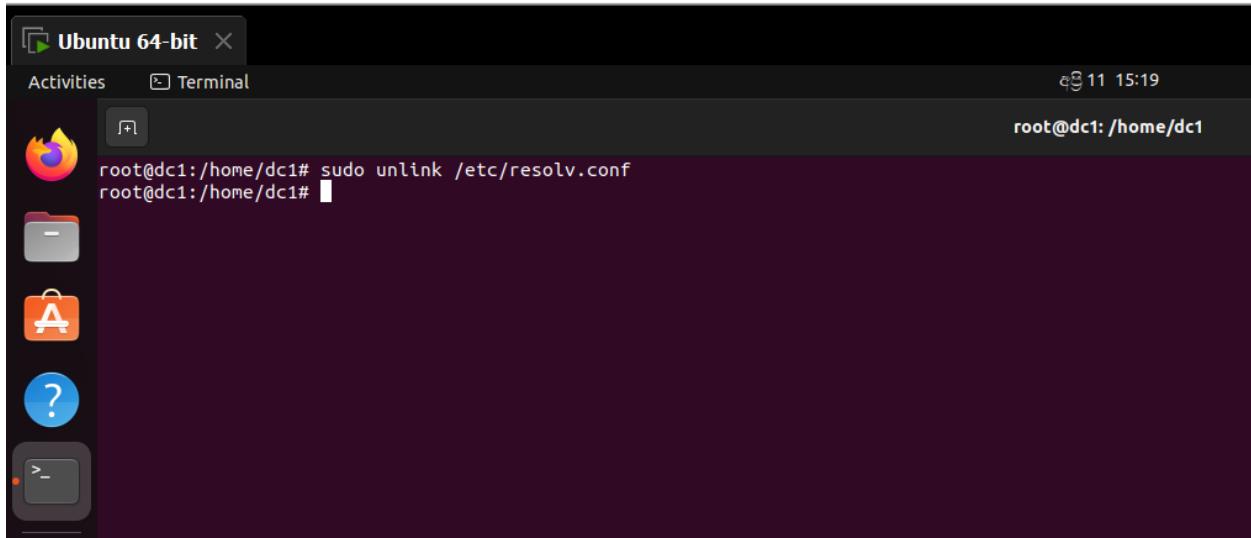
shows the **Fully Qualified Domain Name (FQDN)** of our system and verify that the FQDN resolves to the correct IP address. And also ensures that DNS or /etc/hosts mapping is working properly.



```
Ubuntu 64-bit
Activities Terminal
root@dc1: /home/dc1#
root@dc1:/home/dc1# sudo systemctl disable --now systemd-resolved
Removed /etc/systemd/system/multi-user.target.wants/systemd-resolved.service.
Removed /etc/systemd/system/dbus-org.freedesktop.resolve1.service.
root@dc1:/home/dc1#
```

Figure 1.0.4-Disable the DNS Resolver

Because systemd-resolved often overwrites /etc/resolv.conf, which can interfere with custom DNS setups required by Samba AD (like pointing to your own DNS or Domain Controller).



```
Ubuntu 64-bit
Activities Terminal
root@dc1: /home/dc1#
root@dc1:/home/dc1# sudo unlink /etc/resolv.conf
root@dc1:/home/dc1#
```

Figure 1.0.5-Unlink the config

- This is necessary so we can manually create your own resolv.conf file with the correct DNS server IP, like the domain controller's IP address.

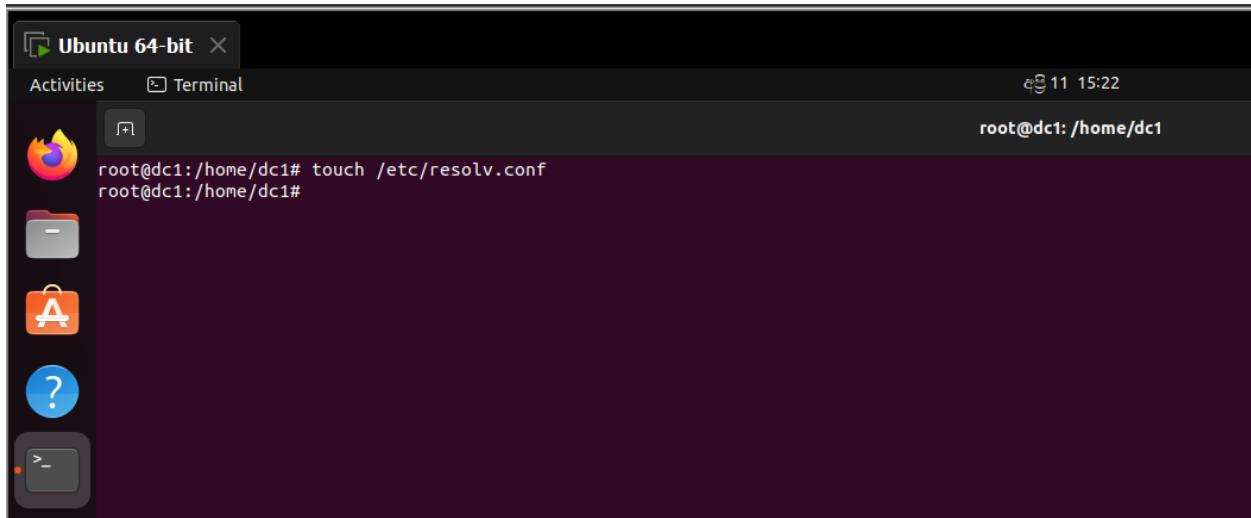


Figure 1.0.6-Create our own Resolv.conf

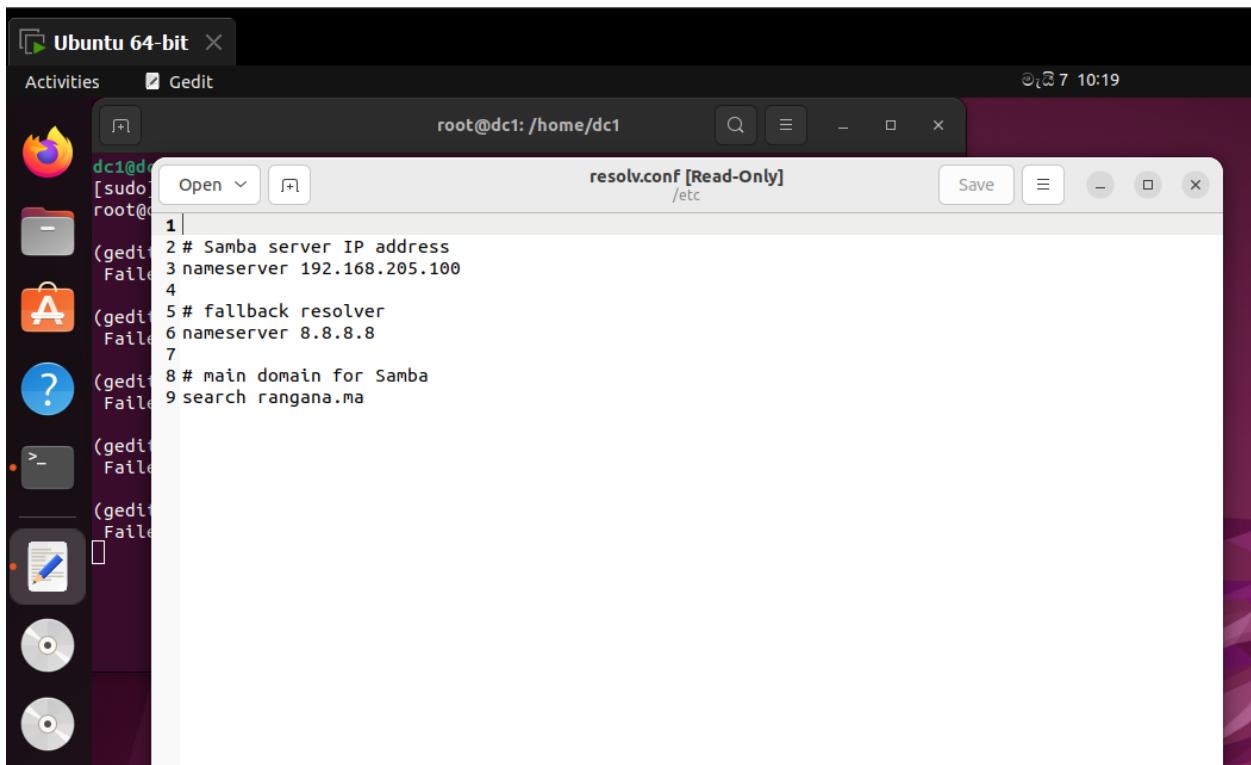


Figure 1.0.7-Edit the newly created resolver config file

This configuration ensures that local domain-related queries go to our Samba DC and internet-related queries fall back to a public DNS, all while using your domain's search path (rangana.ma).

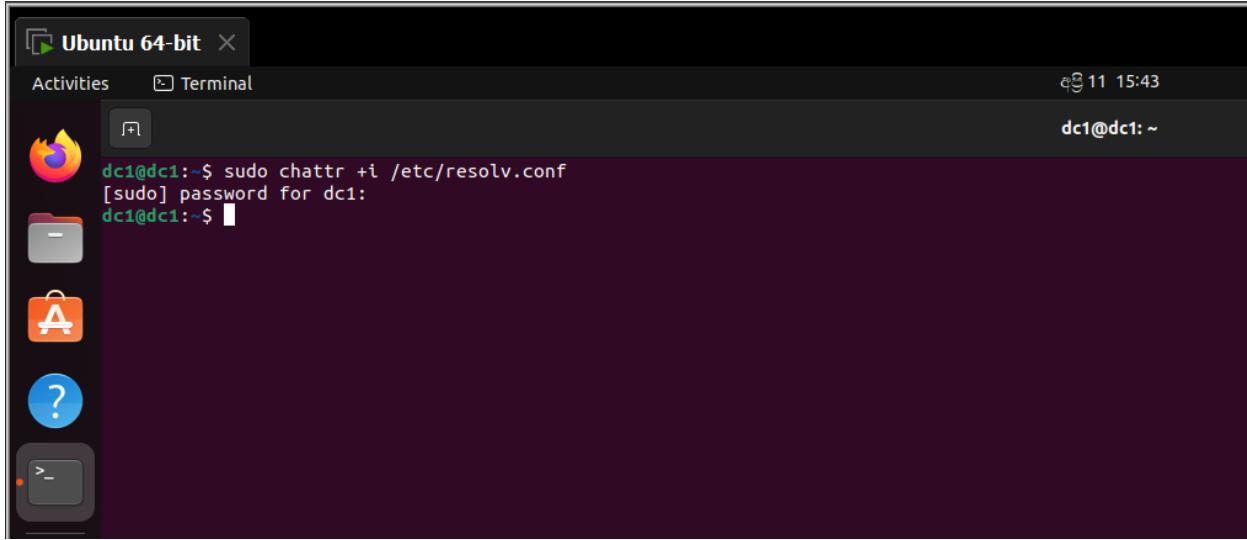


Figure 1.0.8-Make the file immutable

After you manually configure /etc/resolv.conf (with our Samba DC IP as the DNS server), we want to prevent system services like **NetworkManager** or **systemd-resolved** from overwriting it.

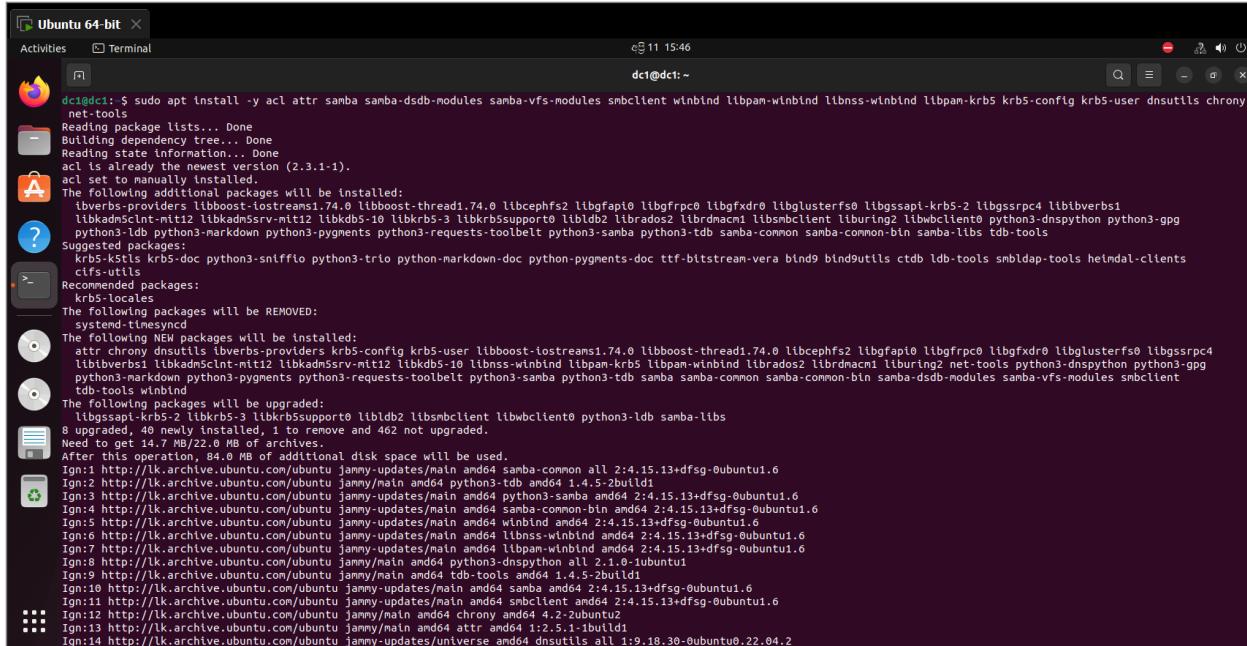
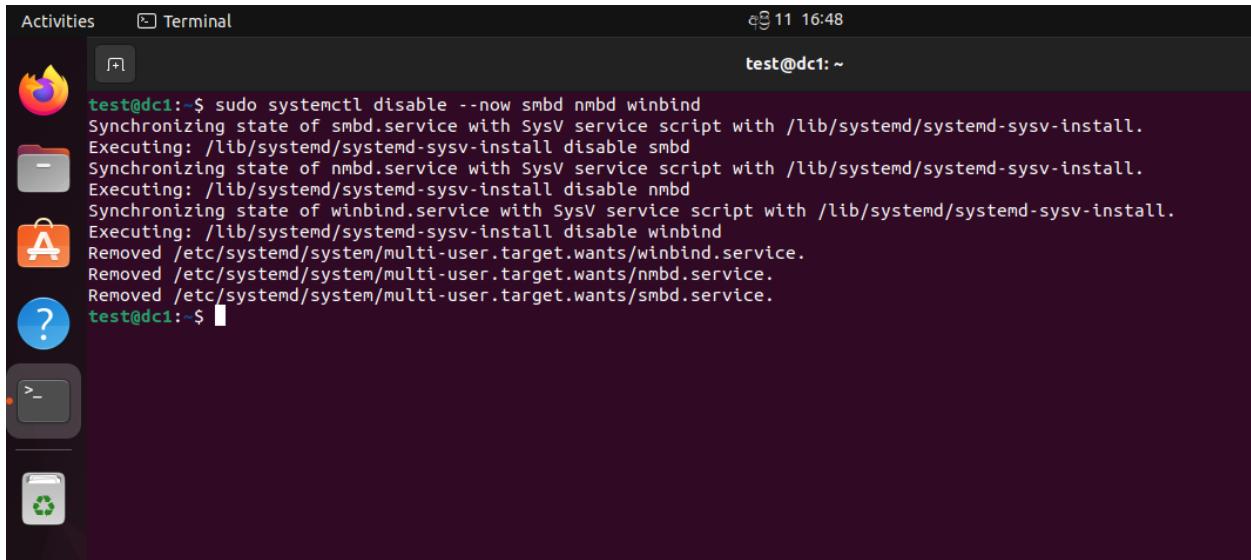


Figure 1.0.9-Install the Samba



A screenshot of an Ubuntu desktop environment. The terminal window shows the command `sudo systemctl disable --now smbd nmbd winbind` being run. The output indicates that the system is synchronizing the state of these services with SysV service scripts and disabling them. It also removes the service files from the multi-user.target.wants directory.

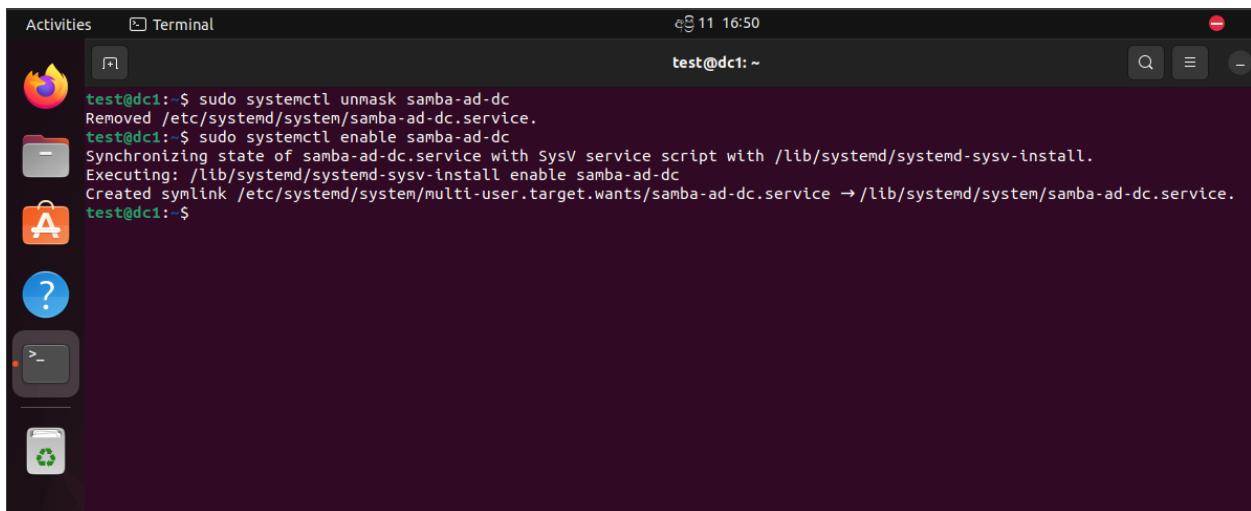
```
Activities Terminal test@dc1: ~
test@dc1:~$ sudo systemctl disable --now smbd nmbd winbind
Synchronizing state of smbd.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable smbd
Synchronizing state of nmbd.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable nmbd
Synchronizing state of winbind.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable winbind
Removed /etc/systemd/system/multi-user.target.wants/winbind.service.
Removed /etc/systemd/system/multi-user.target.wants/nmbd.service.
Removed /etc/systemd/system/multi-user.target.wants/smbd.service.
test@dc1:~$
```

Figure 1.0.10-Disable Samba Services

Stops (--now) and disables the following Samba-related services from starting on boot:

- smbd: Handles SMB/CIFS file sharing.
- nmbd: Manages NetBIOS name service (legacy).
- winbind: Connects Unix/Linux systems to Windows AD for user/group resolution.

You need to stop and disable these services because the Samba AD DC setup uses the Samba service instead, which internally handles all needed functionality. Running smbd, nmbd, or winbind separately can conflict with the Samba service



A screenshot of an Ubuntu desktop environment. The terminal window shows the command `sudo systemctl unmask samba-ad-dc` being run, followed by `sudo systemctl enable samba-ad-dc`. The output indicates that the system is synchronizing the state of the `samba-ad-dc.service` with a SysV service script and enabling it. A symlink is created in the `multi-user.target.wants` directory pointing to the service file.

```
Activities Terminal test@dc1: ~
test@dc1:~$ sudo systemctl unmask samba-ad-dc
Removed /etc/systemd/system/samba-ad-dc.service.
test@dc1:~$ sudo systemctl enable samba-ad-dc
Synchronizing state of samba-ad-dc.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable samba-ad-dc
Created symlink /etc/systemd/system/multi-user.target.wants/samba-ad-dc.service → /lib/systemd/system/samba-ad-dc.service.
test@dc1:~$
```

Figure 1.0.11-Active samba-ad-dc

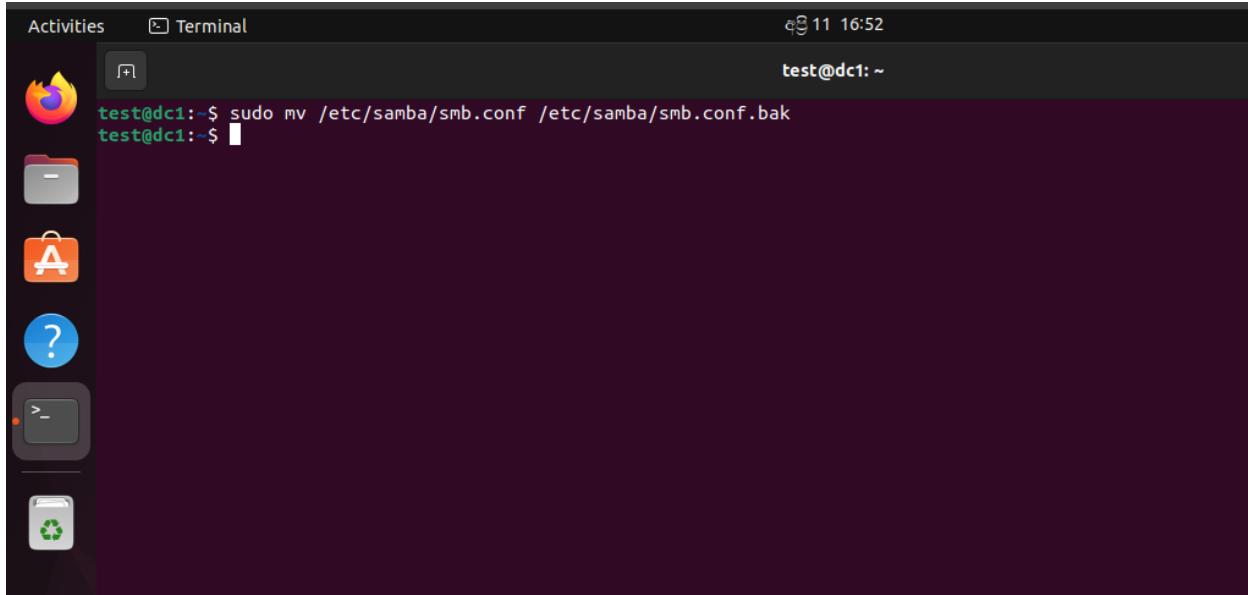


Figure 1.0.12-Back up the original smb.conf file

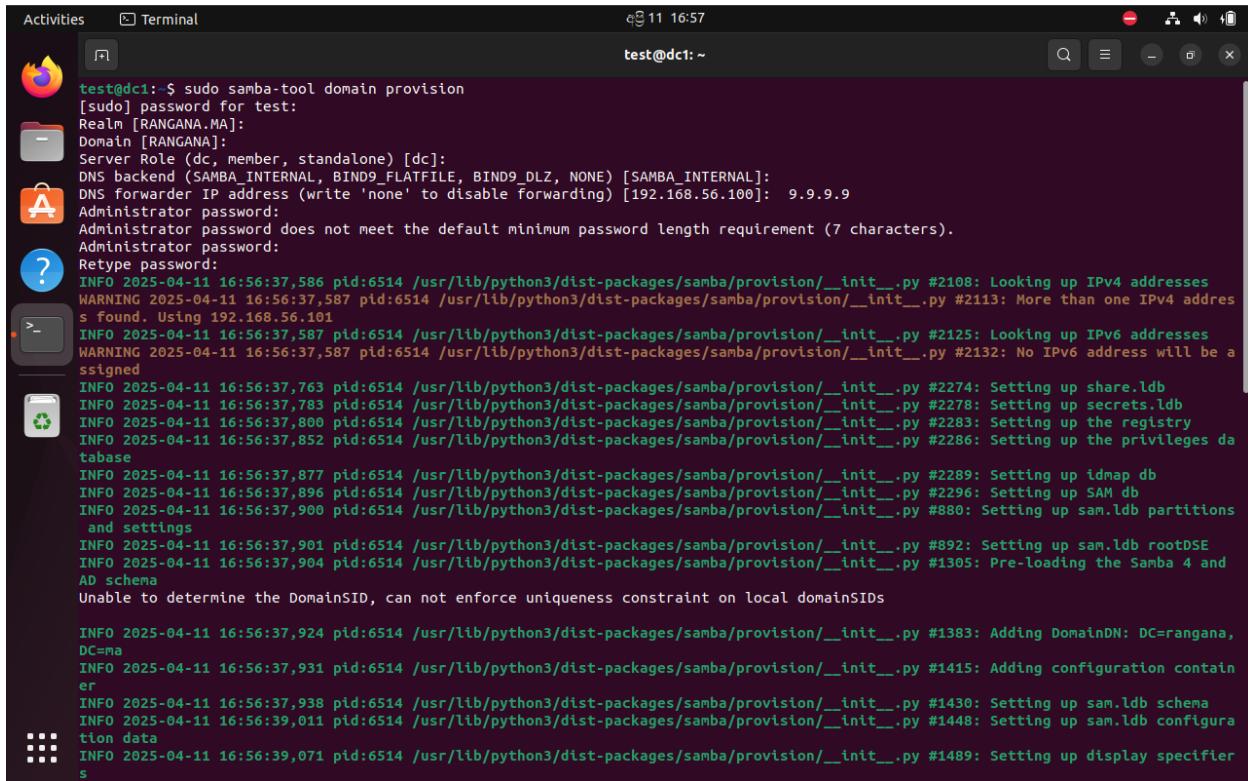
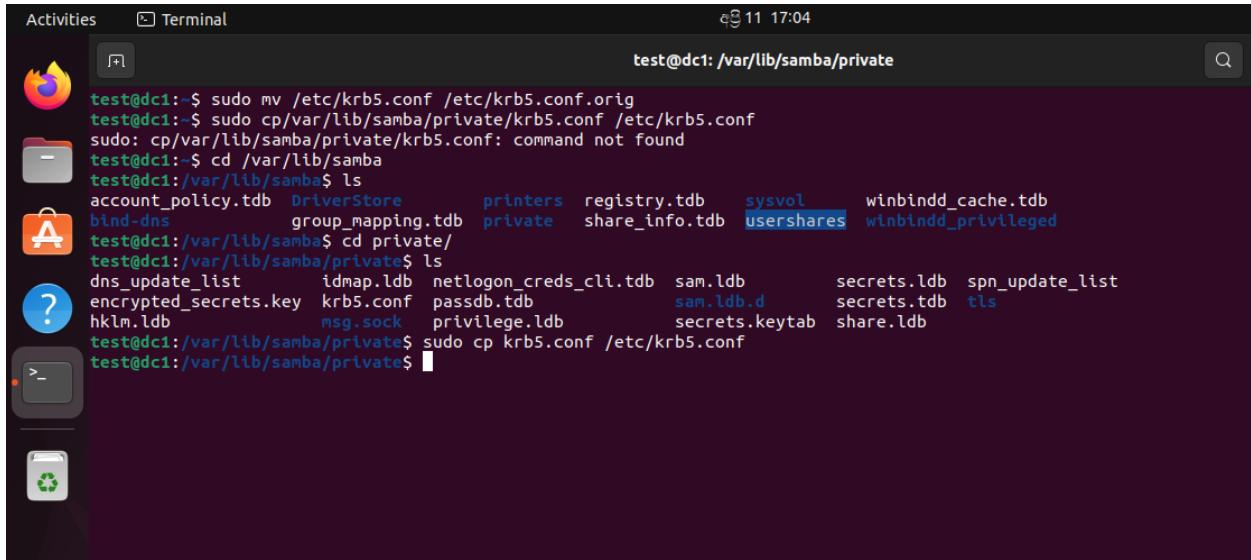
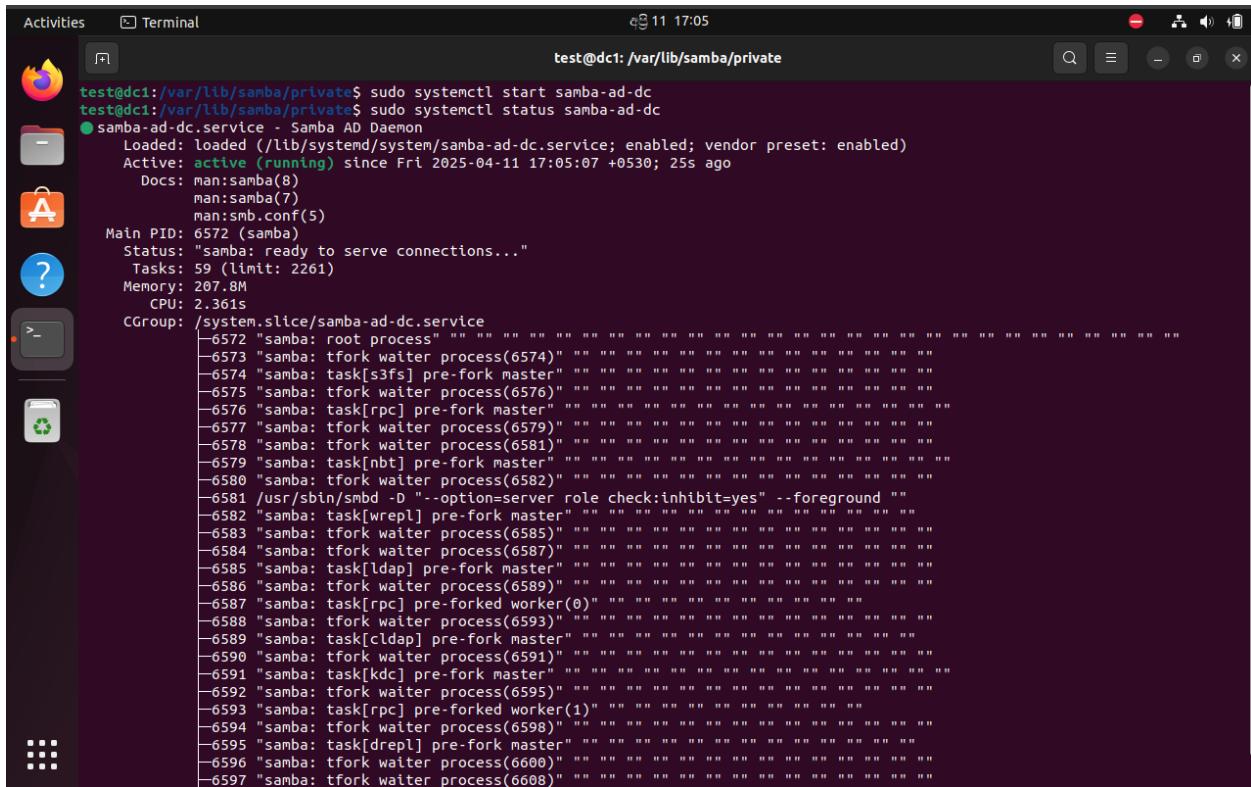


Figure 1.0.13-Provision the Active Directory



```
Activities Terminal test@dc1: /var/lib/samba/private
test@dc1:~$ sudo mv /etc/krb5.conf /etc/krb5.conf.orig
test@dc1:~$ sudo cp /var/lib/samba/private/krb5.conf /etc/krb5.conf
sudo: cp /var/lib/samba/private/krb5.conf: command not found
test@dc1:~$ cd /var/lib/samba
test@dc1:/var/lib/samba$ ls
account_policy.tdb  DriverStore    printers   registry.tdb    sysvol      winbindd_cache.tdb
bind-dns           group_mapping.tdb private    share_info.tdb  usershares  winbindd_privileged
test@dc1:/var/lib/samba$ cd private/
test@dc1:/var/lib/samba/private$ ls
dns_update_list    idmap ldb    netlogon_creds_cli.tdb  sam.ldb      secrets.ldb  spn_update_list
encrypted_secrets.key krb5.conf  passdb.tdb          sam.ldb.d    secrets.tdb  tls
hklm.ldb           msg.sock  privilege.ldb        secrets.keytab share.ldb
test@dc1:/var/lib/samba/private$ sudo cp krb5.conf /etc/krb5.conf
test@dc1:/var/lib/samba/private$
```

Figure 1.0.14-Backup and replace the Kerberos Config



```
Activities Terminal test@dc1: /var/lib/samba/private
test@dc1:/var/lib/samba/private$ sudo systemctl start samba-ad-dc
test@dc1:/var/lib/samba/private$ sudo systemctl status samba-ad-dc
● Samba-ad-dc.service - Samba AD Daemon
  Loaded: loaded (/lib/systemd/system/samba-ad-dc.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2025-04-11 17:05:07 +0530; 25s ago
    Docs: man:samba(8)
          man:samba(7)
          man:smb.conf(5)
  Main PID: 6572 (samba)
  Status: "samba: ready to serve connections..."
   Tasks: 59 (limit: 2261)
  Memory: 207.8M
     CPU: 2.361s
  CGroup: /system.slice/samba-ad-dc.service
          └─6572 "samba: root process" ...
  6573 "samba: tfork waiter process(6574)" ...
  6574 "samba: task[s3fs] pre-fork master" ...
  6575 "samba: tfork waiter process(6576)" ...
  6576 "samba: task[rpc] pre-fork master" ...
  6577 "samba: tfork waiter process(6579)" ...
  6578 "samba: tfork waiter process(6581)" ...
  6579 "samba: task[nbt] pre-fork master" ...
  6580 "samba: tfork waiter process(6582)" ...
  6581 "/usr/sbin/smbd -D --option=server role check:inhibit=yes" --foreground ...
  6582 "samba: task[wrepl] pre-fork master" ...
  6583 "samba: tfork waiter process(6585)" ...
  6584 "samba: tfork waiter process(6587)" ...
  6585 "samba: task[ldap] pre-fork master" ...
  6586 "samba: tfork waiter process(6589)" ...
  6587 "samba: task[rpc] pre-forked worker(0)" ...
  6588 "samba: tfork waiter process(6593)" ...
  6589 "samba: task[ldap] pre-fork master" ...
  6590 "samba: tfork waiter process(6591)" ...
  6591 "samba: task[kdc] pre-fork master" ...
  6592 "samba: tfork waiter process(6595)" ...
  6593 "samba: task[rpc] pre-forked worker(1)" ...
  6594 "samba: tfork waiter process(6598)" ...
  6595 "samba: task[drepl] pre-fork master" ...
  6596 "samba: tfork waiter process(6600)" ...
  6597 "samba: tfork waiter process(6608)" ...
```

Figure 1.0.15-Start the samba-ad-dc service and check the status

The screenshot shows a Linux desktop interface with a dark theme. At the top, there is a header bar with "Activities" and "Terminal" buttons. The system tray on the right shows the date and time as "11 17:12". Below the header bar is a vertical dock on the left containing icons for a browser (Mozilla Firefox), file manager (Nautilus), terminal (GNOME Terminal), application menu (Activities overview), and a recycle bin.

The main area is a terminal window titled "root@dc1: /home/test". It contains the following command history:

```
root@dc1:/home/test# chown root:_chrony /var/lib/samba/ntp_signd/
root@dc1:/home/test# sudo chmod 750 /var/lib/samba/ntp_signd/
root@dc1:/home/test#
```

Figure 1.0.16-Set permissions

Changes the ownership of the `ntp_signd` directory to the `_chrony` group, allowing the Chrony NTP service to access it.

Sets permissions so that only the owner (root) has full access, the group (`_chrony`) can read and execute, and others have no access.

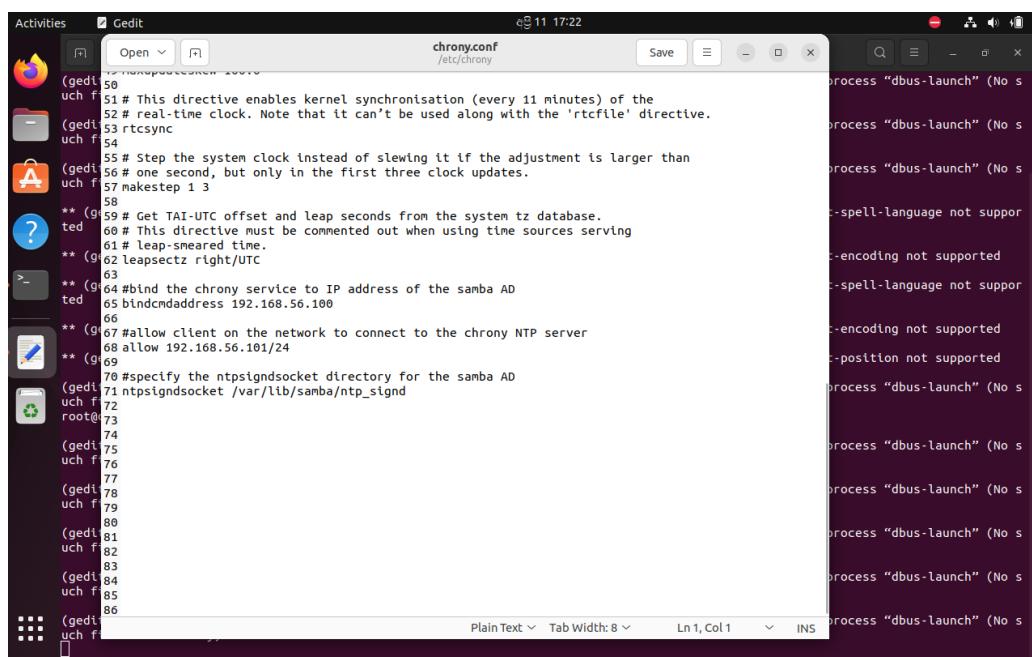
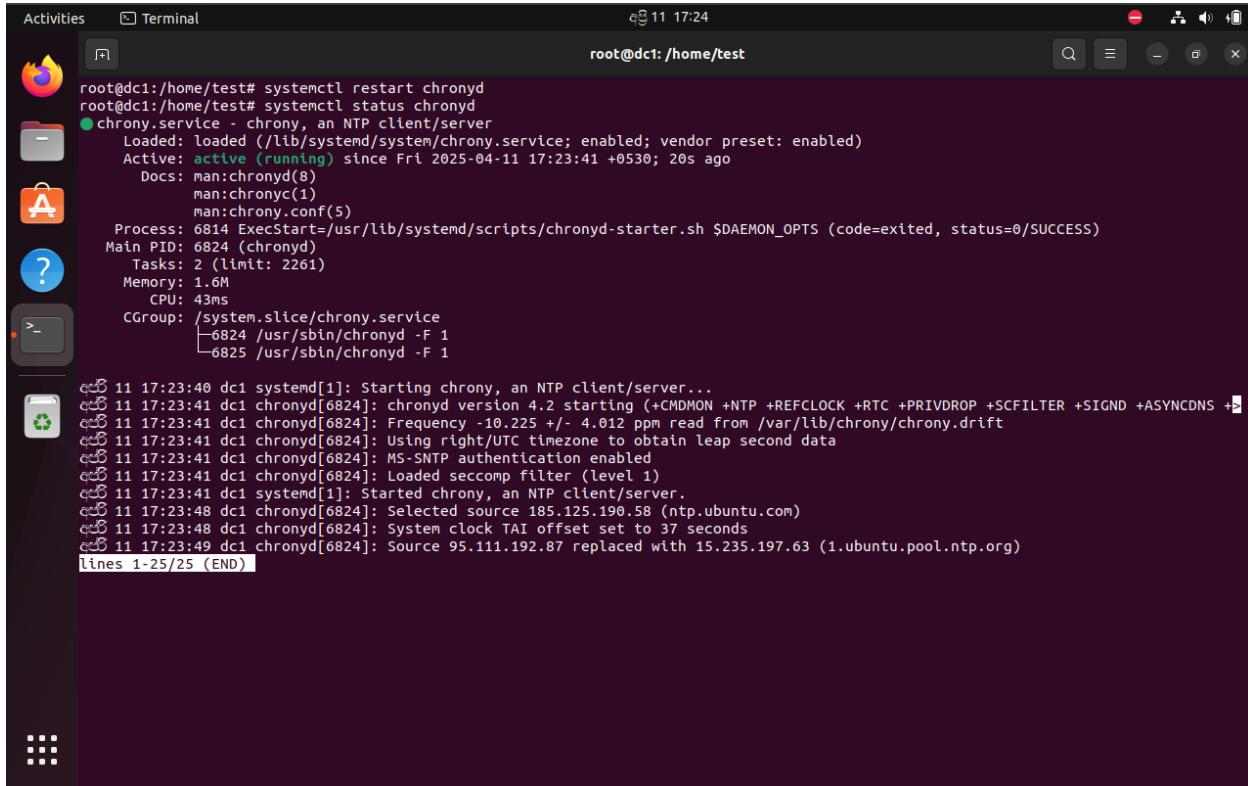


Figure 1.0.17-Update Chrony.conf

Chronyd is the background service of the Chrony time synchronization tool, used to keep a system's clock accurate by syncing with NTP servers. It's faster and more efficient than traditional NTP daemons like ntpd, making it ideal for systems with unstable clocks, such as laptops or virtual machines. chronyd can also act as an NTP server, supports offline correction, and is commonly used in Samba Active Directory environments to provide secure time synchronization for Windows clients.



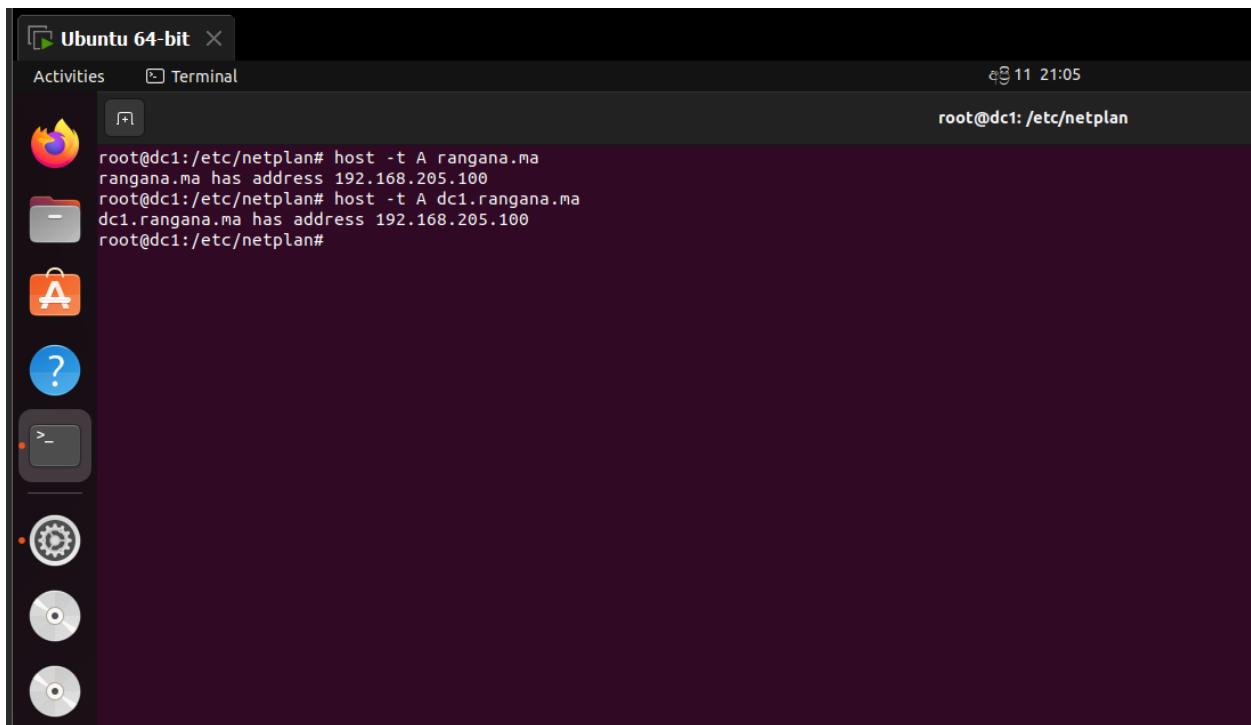
The screenshot shows a terminal window titled "root@dc1:/home/test". The terminal displays the following command output:

```
root@dc1:/home/test# systemctl restart chronyd
root@dc1:/home/test# systemctl status chronyd
● chrony.service - chrony, an NTP client/server
  Loaded: loaded (/lib/systemd/system/chrony.service; enabled; vendor preset: enabled)
  Active: active (running) since Fri 2025-04-11 17:23:41 +0530; 20s ago
    Docs: man:chrony(8)
          man:chronyc(1)
          man:chrony.conf(5)
   Process: 6814 ExecStart=/usr/lib/systemd/scripts/chronyd-starter.sh $DAEMON_OPTS (code=exited, status=0/SUCCESS)
 Main PID: 6824 (chronyd)
   Tasks: 2 (limit: 2261)
     Memory: 1.6M
        CPU: 43ms
      CGroup: /system.slice/chrony.service
              └─6824 /usr/sbin/chronyd -F 1
              ├─6825 /usr/sbin/chronyd -F 1

May 11 17:23:40 dc1 systemd[1]: Starting chrony, an NTP client/server...
May 11 17:23:41 dc1 chronyd[6824]: chronyd version 4.2 starting (+CMDMON +NTP +REFCLOCK +RTC +PRIVDROP +SCFILTER +SIGND +ASYNCDNS +S
May 11 17:23:41 dc1 chronyd[6824]: Frequency -10.225 +/- 4.012 ppm read from /var/lib/chrony/chrony.drift
May 11 17:23:41 dc1 chronyd[6824]: Using right/UTC timezone to obtain leap second data
May 11 17:23:41 dc1 chronyd[6824]: MS-SNTP authentication enabled
May 11 17:23:41 dc1 chronyd[6824]: Loaded seccomp filter (level 1)
May 11 17:23:41 dc1 systemd[1]: Started chrony, an NTP client/server.
May 11 17:23:48 dc1 chronyd[6824]: Selected source 185.125.190.58 (ntp.ubuntu.com)
May 11 17:23:48 dc1 chronyd[6824]: System clock TAI offset set to 37 seconds
May 11 17:23:49 dc1 chronyd[6824]: Source 95.111.192.87 replaced with 15.235.197.63 (1.ubuntu.pool.ntp.org)
```

lines 1-25/25 (END)

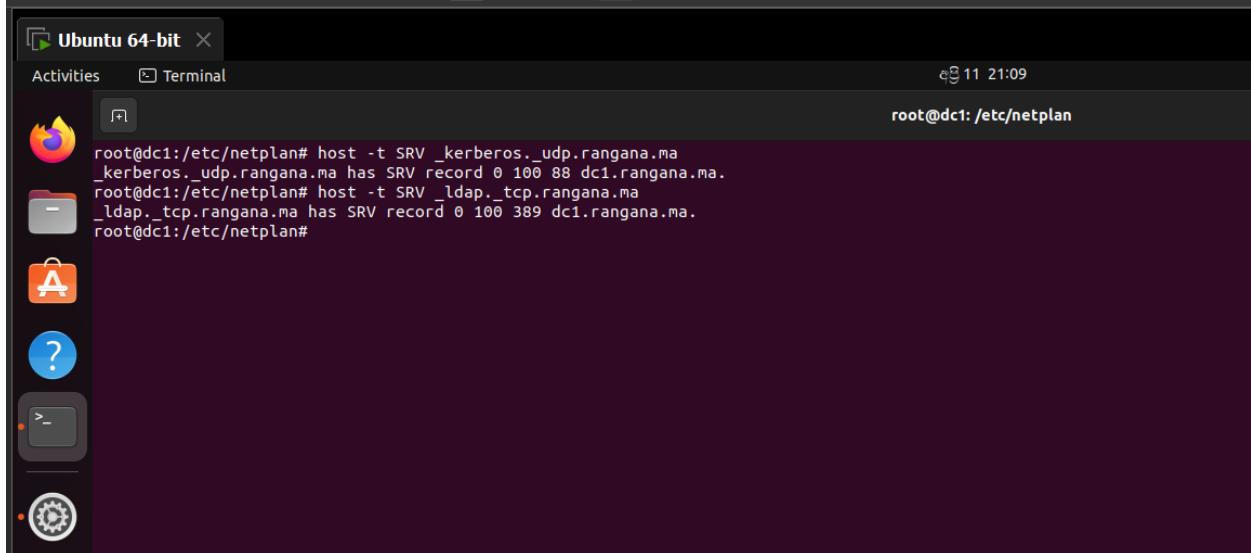
Figure 1.0.18-Restart Chronyd and get its status



A screenshot of an Ubuntu 64-bit desktop environment. The terminal window shows the command `host -t A rangana.ma` being run, which returns the IP address 192.168.205.100. It then runs `host -t A dc1.rangana.ma`, which also returns the IP address 192.168.205.100. The terminal prompt is `root@dc1:/etc/netplan#`. The desktop interface includes a dock with icons for Dash, Home, Applications, Help, and others.

```
root@dc1:/etc/netplan# host -t A rangana.ma
rangana.ma has address 192.168.205.100
root@dc1:/etc/netplan# host -t A dc1.rangana.ma
dc1.rangana.ma has address 192.168.205.100
root@dc1:/etc/netplan#
```

Figure 1.0.19-Verifying Samba Active Directory



A screenshot of an Ubuntu 64-bit desktop environment. The terminal window shows the command `host -t SRV _kerberos._udp.rangana.ma` being run, which returns an SRV record for port 88. It then runs `host -t SRV _ldap._tcp.rangana.ma`, which returns an SRV record for port 389. The terminal prompt is `root@dc1:/etc/netplan#`. The desktop interface includes a dock with icons for Dash, Home, Applications, Help, and others.

```
root@dc1:/etc/netplan# host -t SRV _kerberos._udp.rangana.ma
_kerberos._udp.rangana.ma has SRV record 0 100 88 dc1.rangana.ma.
root@dc1:/etc/netplan# host -t SRV _ldap._tcp.rangana.ma
_ldap._tcp.rangana.ma has SRV record 0 100 389 dc1.rangana.ma.
root@dc1:/etc/netplan#
```

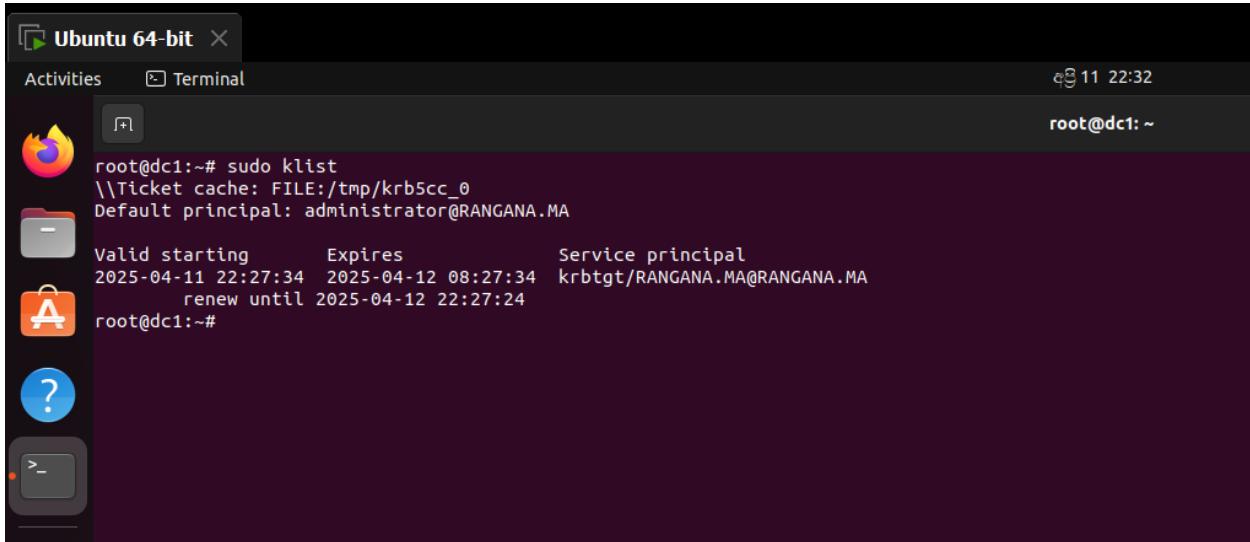
Figure 1.0.20-Verify the Kerberos and ldap services

These commands check if the DNS server correctly advertises essential services for a Samba Active Directory domain. The Kerberos SRV record ensures clients can locate the authentication service, while the LDAP SRV record confirms access to directory services. Proper resolution of these records is required for domain joins and secure client communication with the AD server.

```
root@dc1:~# sudo kinit administrator@RANGANA.MA
Password for administrator@RANGANA.MA:
Warning: Your password will expire in 41 days on 2025-05-23 22:08:09 +0530
root@dc1:~#
```

Figure 1.0.21-Authenticate to Kerberos using an administrator

These commands authenticate the administrator to the Kerberos system (kinit) and then display the cached tickets (klist) to verify successful authentication with the Samba AD domain.

A screenshot of an Ubuntu 64-bit desktop environment. The terminal window is open and shows the command 'sudo klist' being run. The output indicates a ticket cache was created at /tmp/krb5cc_0, the default principal is administrator@RANGANA.MA, and a service principal krbtgt/RANGANA.MA@RANGANA.MA has a valid starting time of 2025-04-11 22:27:34 and an expiration time of 2025-04-12 08:27:34, with a renewal until 2025-04-12 22:27:24.

```
root@dc1:~# sudo klist
\Ticket cache: FILE:/tmp/krb5cc_0
Default principal: administrator@RANGANA.MA

Valid starting     Expires            Service principal
2025-04-11 22:27:34  2025-04-12 08:27:34  krbtgt/RANGANA.MA@RANGANA.MA
                  renew until 2025-04-12 22:27:24
root@dc1:~#
```

Figure 1.0.22-verify list cached Kerberos tickets

4. Configuring the Client Machine

4.1 Windows Client

Network

Set a data limit to help control data usage on this network

IP settings

IP assignment:	Manual
IPv4 address:	192.168.205.30
IPv4 subnet prefix length:	24
IPv4 gateway:	192.168.205.2
IPv4 DNS servers:	192.168.205.100 8.8.8.8

Edit

Properties

Link speed (Receive/Transmit):	1000/1000 (Mbps)
Link-local IPv6 address:	fe80::d6da:49c9:c10c:7964%12
IPv4 address:	192.168.205.30
IPv4 DNS servers:	192.168.205.100 8.8.8.8
Manufacturer:	Intel Corporation
Description:	Intel(R) 82574L Gigabit Network Connection
Driver version:	12.17.10.8
Physical address (MAC):	00-0C-29-7A-E2-C2

Copy

Figure 2.0.1 Edit network Configurations in the Windows machine

We use the same network range IP for the Windows client. The gateway same as the Samba server machine gateway, and DNS is the Samba server IP, and the alternative DNS is 8.8.8.8.

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\malee> Get-DnsClientServerAddress

InterfaceAlias      Interface Address   ServerAddresses
Index      Family
-----
Ethernet0          12 IPv4    {192.168.205.100, 8.8.8.8}
Ethernet0          12 IPv6    {}
Loopback Pseudo-Interface 1  1 IPv4    {}
Loopback Pseudo-Interface 1  1 IPv6    {fec0:0:0:ffff::1, fec0:0:0:ffff::2, fec0:0:0:ffff::3}

```

Figure 2.0.2-Verify the DNS resolver

```

PS C:\Users\malee> ping dc1.rangana.ma

Pinging dc1.rangana.ma [192.168.205.100] with 32 bytes of data:
Reply from 192.168.205.100: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.205.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\malee> ping rangana.ma

```

```

PS C:\Users\malee> ping rangana.ma

Pinging rangana.ma [192.168.205.100] with 32 bytes of data:
Reply from 192.168.205.100: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.205.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\malee>

```

Figure 2.0.3-Ping AD domain dc1.rangana.ma and rangana.ma

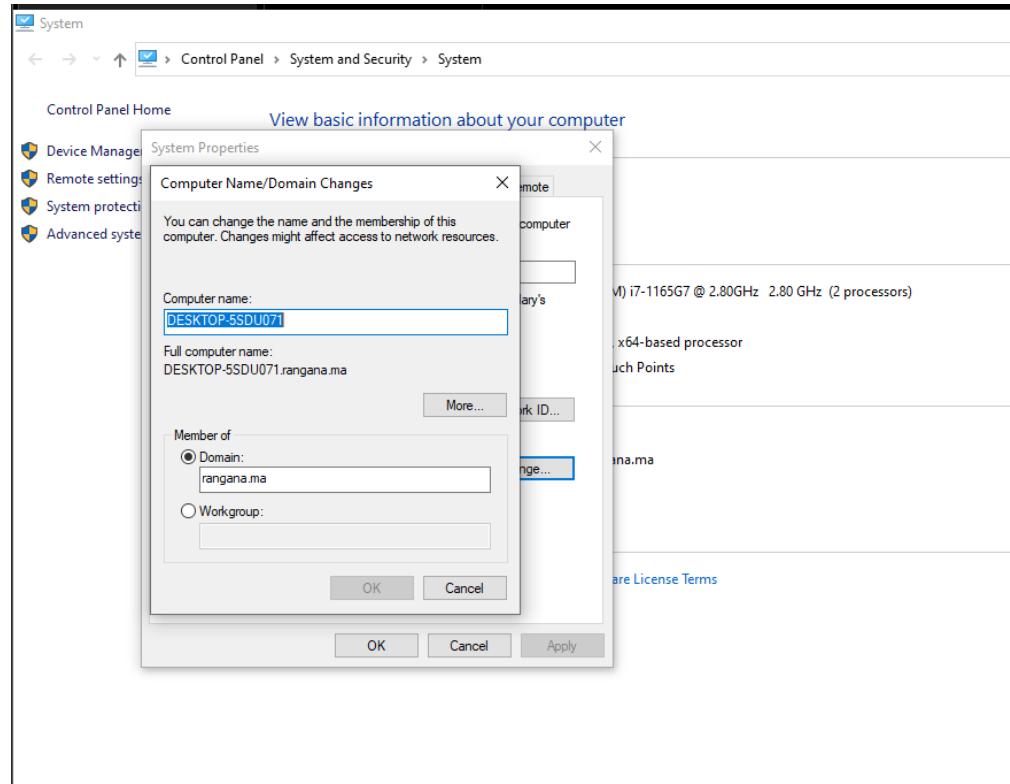


Figure 2.0.4-Add Windows 10 to Active Directory

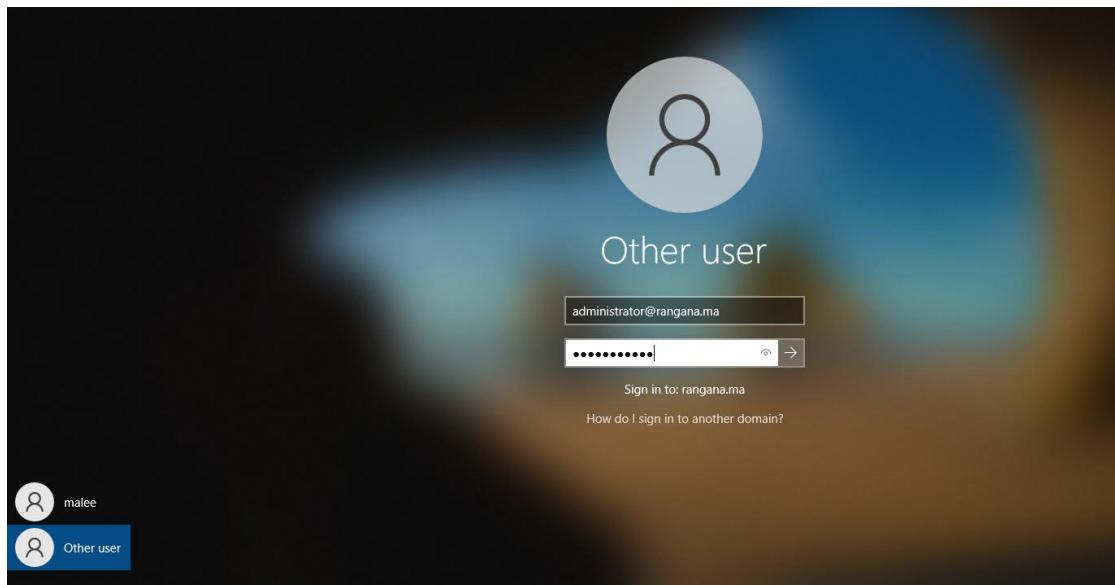


Figure 2.0.5-Logging in with the administrator account

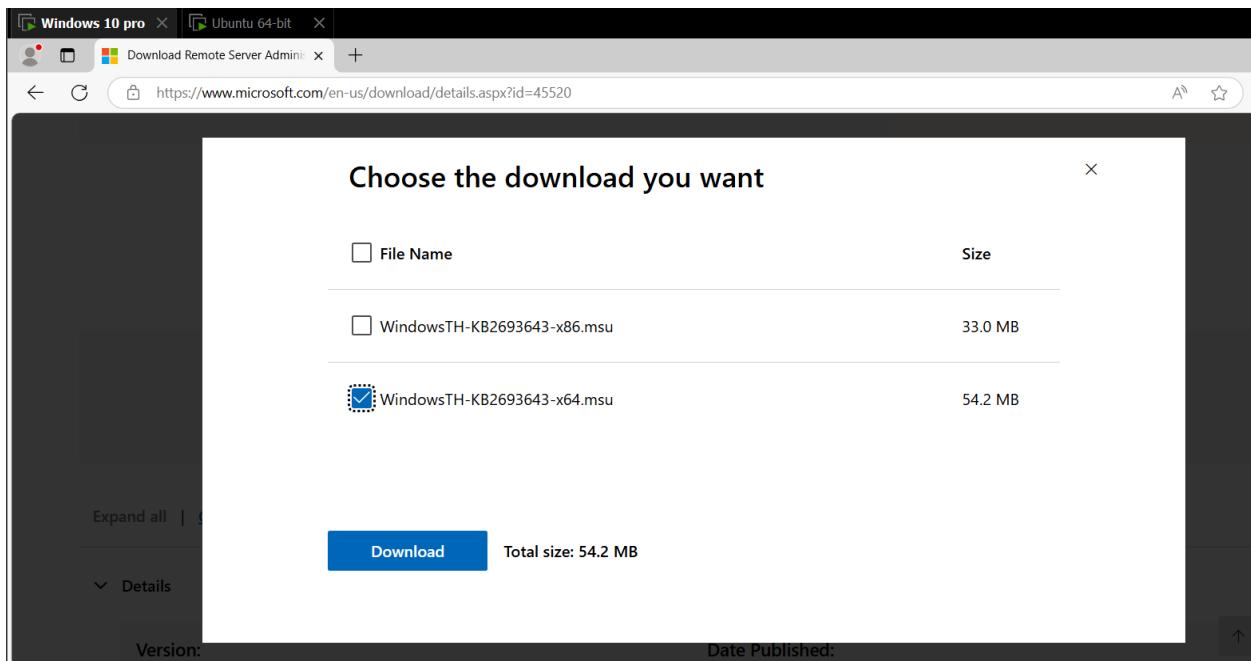


Figure 2.0.6-Download RSAT to administer the domain from Windows

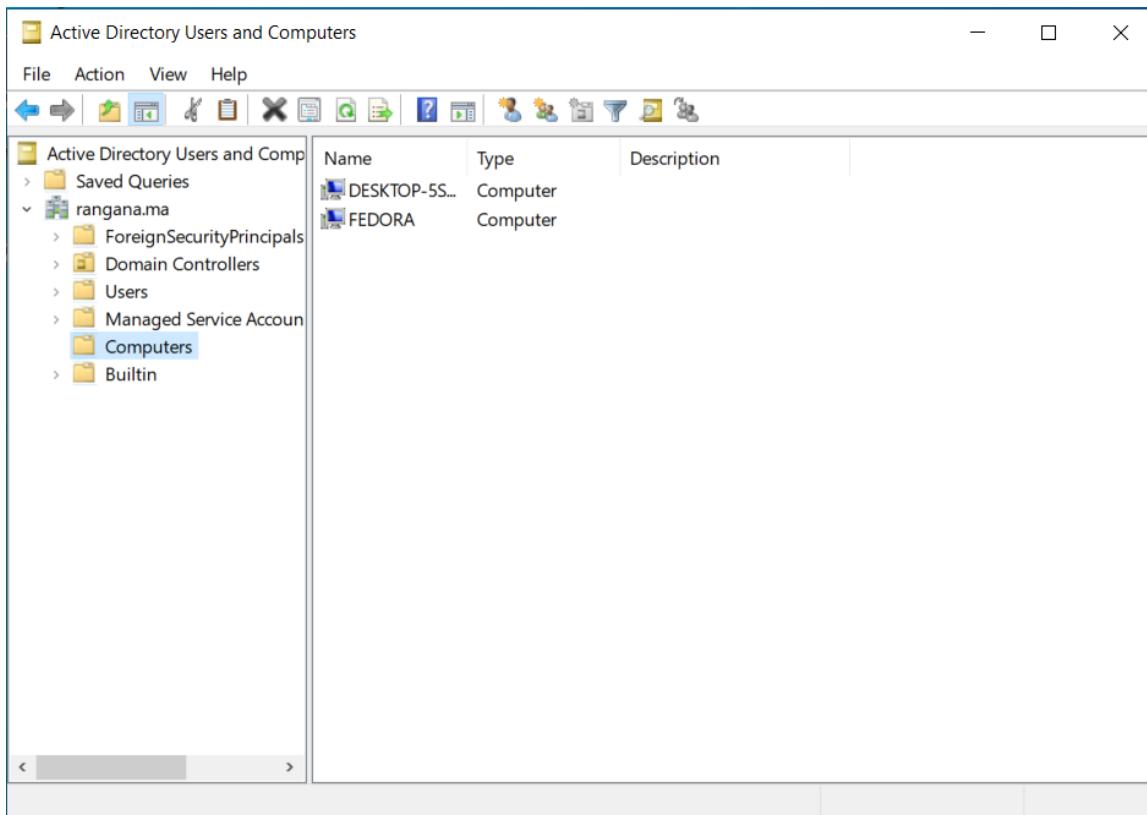


Figure 2.0.7-AD user and computers

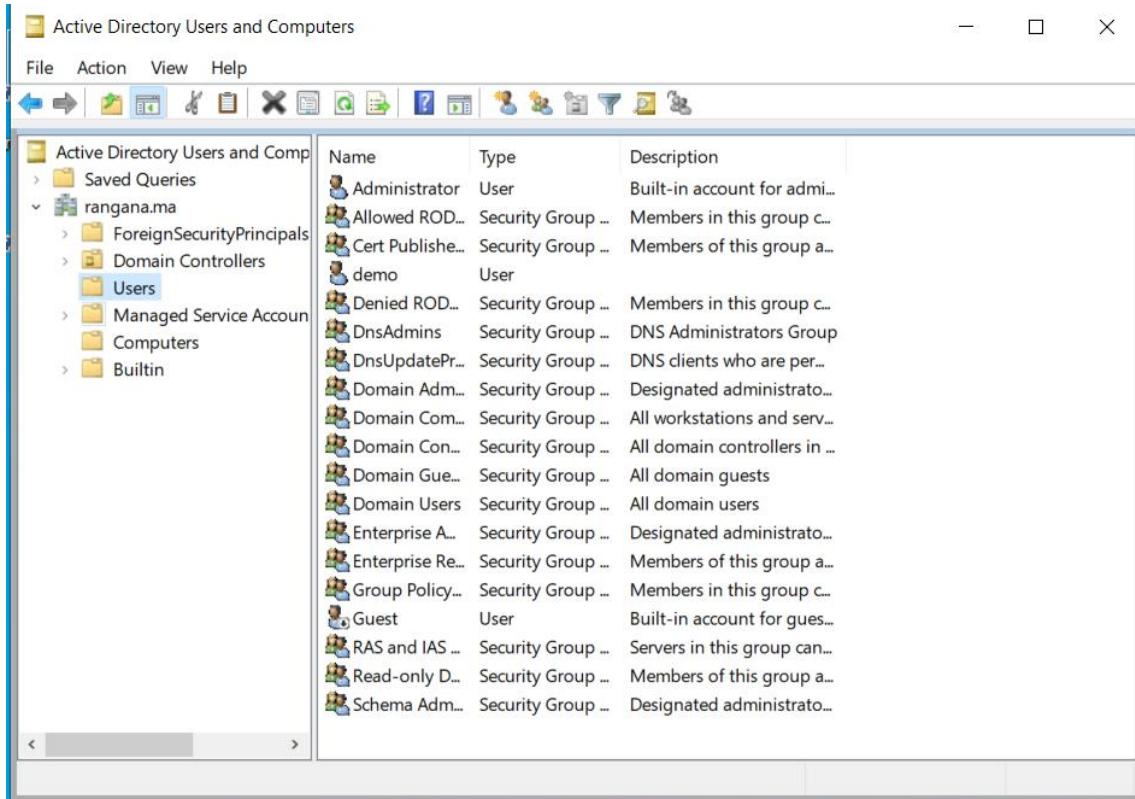


Figure 2.0.8-User's window

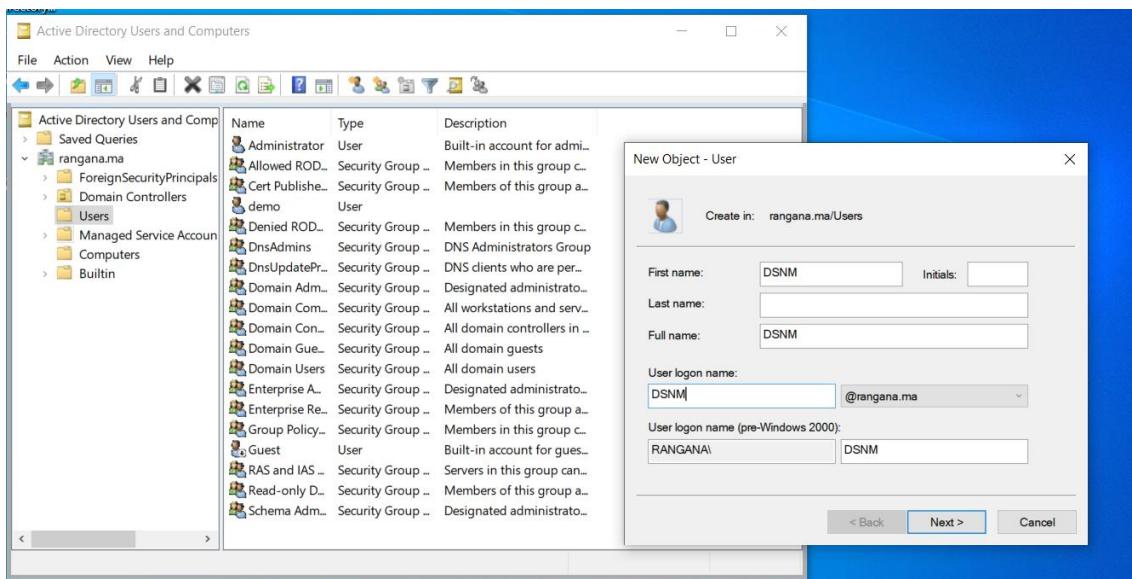


Figure 2.0.9-Create a new user called DSNM

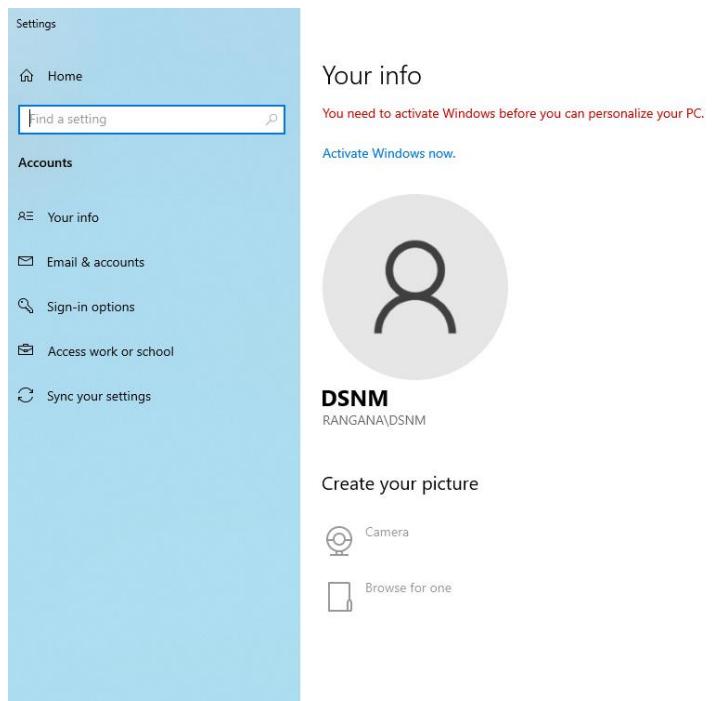
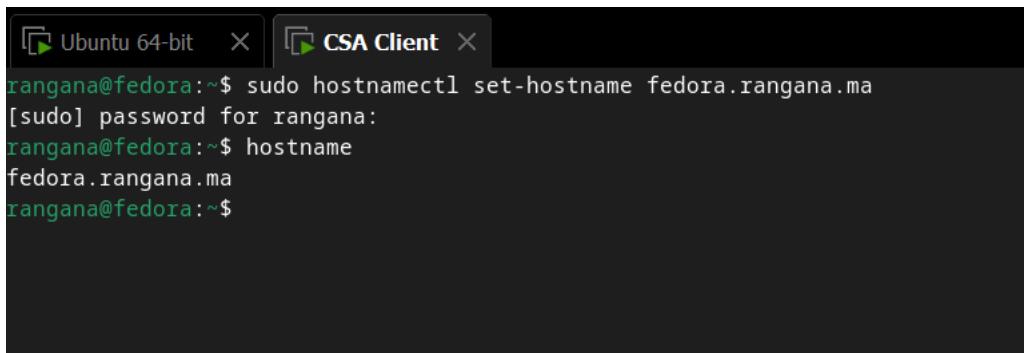


Figure 2.0.10-Logged in DSNM user Account

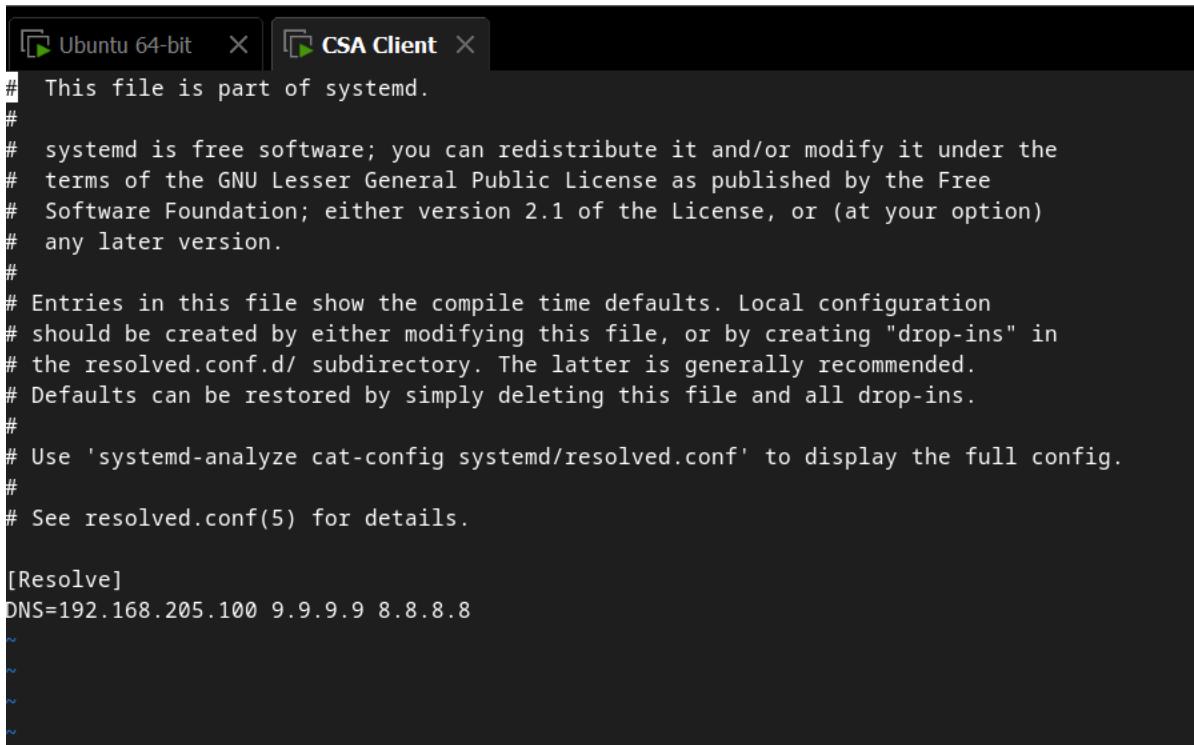
4.2 Fedora Client



A terminal window titled "Ubuntu 64-bit" and "CSA Client". The command "sudo hostnamectl set-hostname fedora.rangana.ma" is run, followed by a password entry prompt "[sudo] password for rangana:", then the new host name "fedora.rangana.ma" is displayed, and finally the prompt "rangana@fedora:~\$".

```
rangana@fedora:~$ sudo hostnamectl set-hostname fedora.rangana.ma
[sudo] password for rangana:
rangana@fedora:~$ hostname
fedora.rangana.ma
rangana@fedora:~$
```

Figure 2.0.11-Change hostname

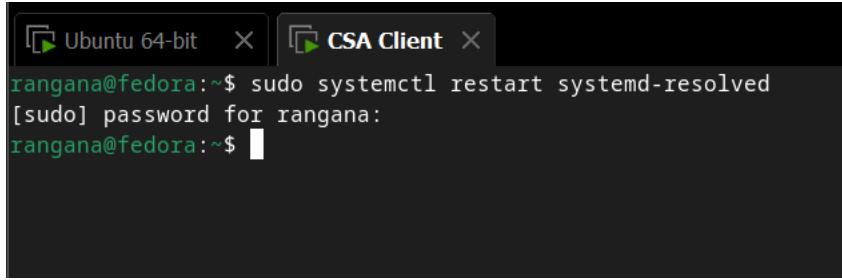


A terminal window titled "Ubuntu 64-bit" and "CSA Client". The file "/etc/systemd/resolved.conf" is viewed, showing its contents. The file contains comments about systemd, configuration entries, and the [Resolve] section which includes the DNS= setting. The bottom of the file shows several blank lines.

```
# This file is part of systemd.
#
# systemd is free software; you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation; either version 2.1 of the License, or (at your option)
# any later version.
#
# Entries in this file show the compile time defaults. Local configuration
# should be created by either modifying this file, or by creating "drop-ins" in
# the resolved.conf.d/ subdirectory. The latter is generally recommended.
# Defaults can be restored by simply deleting this file and all drop-ins.
#
# Use 'systemd-analyze cat-config systemd/resolved.conf' to display the full config.
#
# See resolved.conf(5) for details.

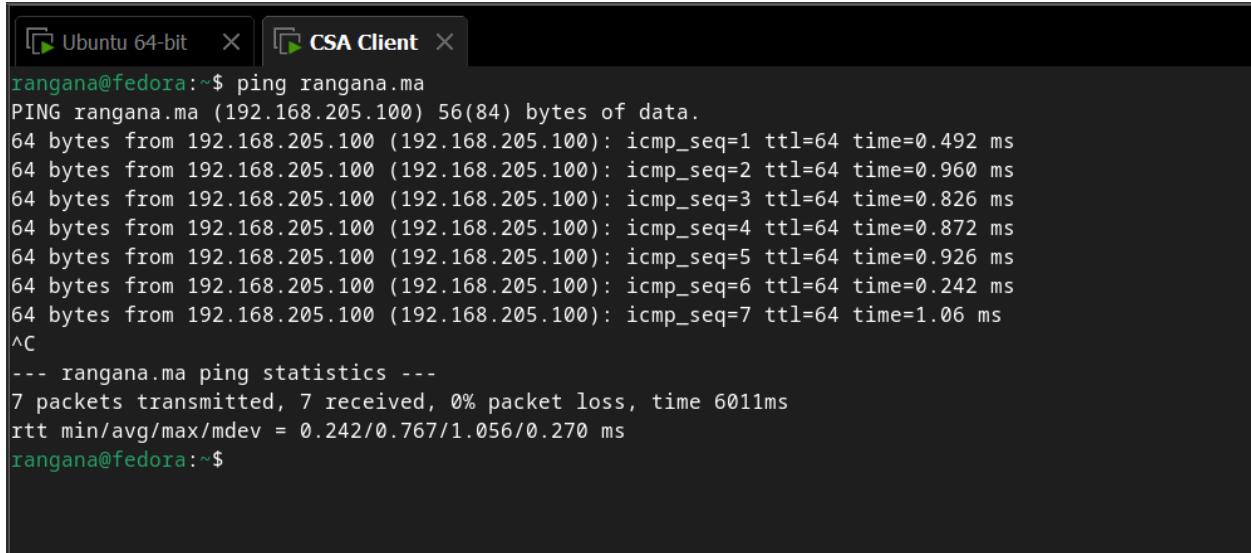
[Resolve]
DNS=192.168.205.100 9.9.9.9 8.8.8.8
```

Figure 2.0.12-Modify resolved information



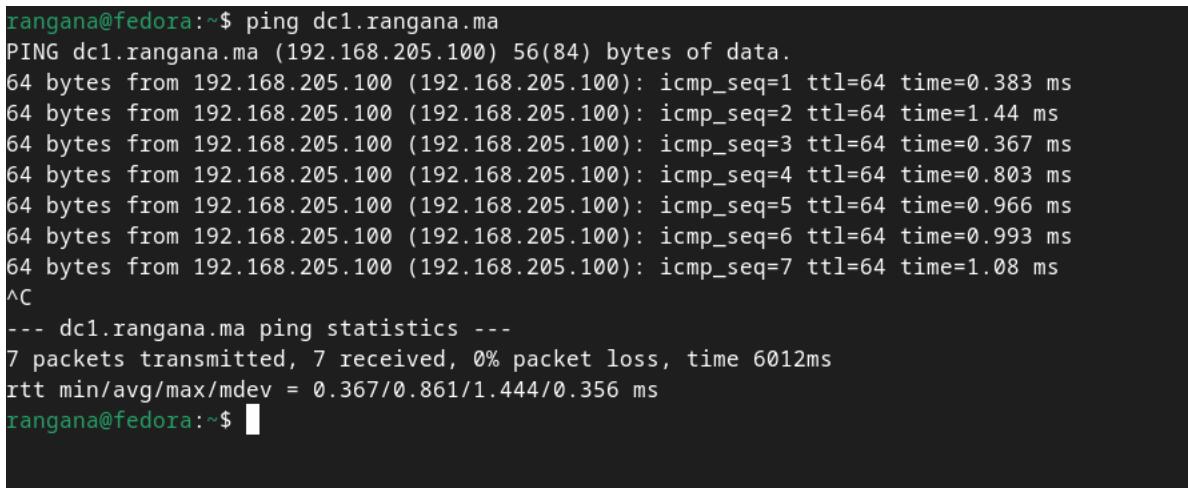
```
Ubuntu 64-bit X CSA Client X
rangana@fedora:~$ sudo systemctl restart systemd-resolved
[sudo] password for rangana:
rangana@fedora:~$
```

Figure 2.0.13-Restart systemd-resolved



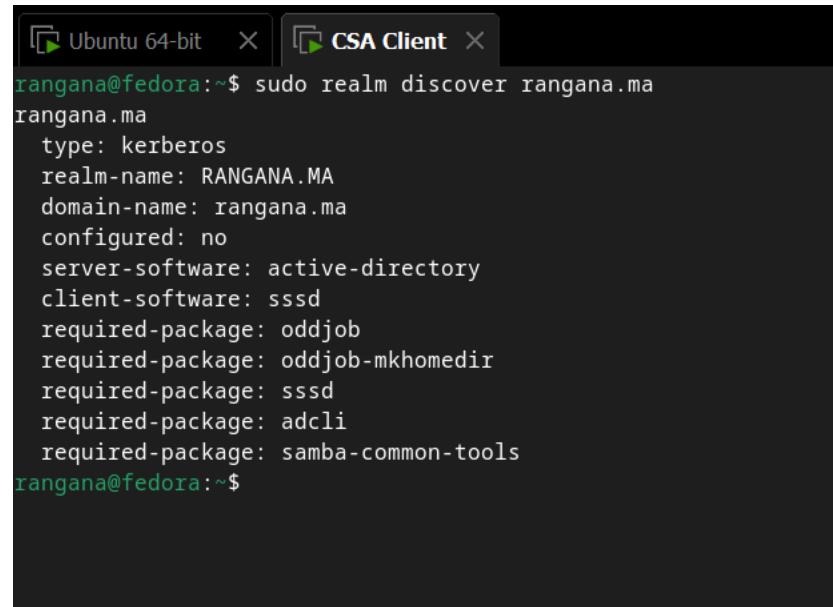
```
Ubuntu 64-bit X CSA Client X
rangana@fedora:~$ ping rangana.ma
PING rangana.ma (192.168.205.100) 56(84) bytes of data.
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=1 ttl=64 time=0.492 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=2 ttl=64 time=0.960 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=3 ttl=64 time=0.826 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=4 ttl=64 time=0.872 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=5 ttl=64 time=0.926 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=6 ttl=64 time=0.242 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=7 ttl=64 time=1.06 ms
^C
--- rangana.ma ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6011ms
rtt min/avg/max/mdev = 0.242/0.767/1.056/0.270 ms
rangana@fedora:~$
```

Figure 2.0.14-Ping the Samba domain



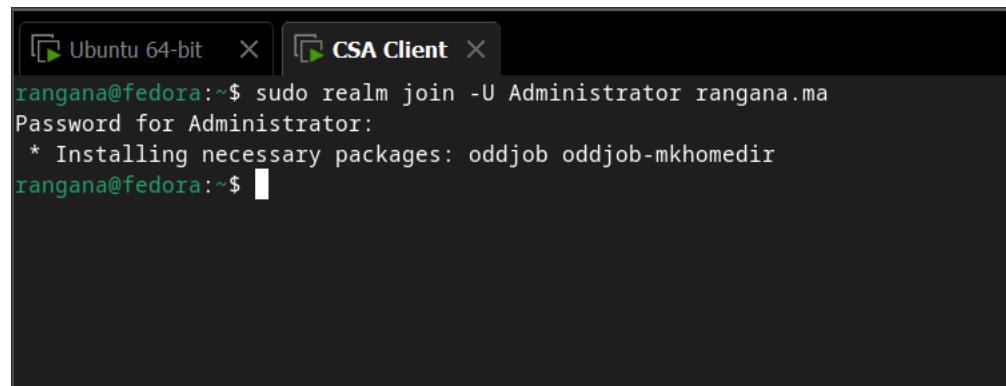
```
rangana@fedora:~$ ping dc1.rangana.ma
PING dc1.rangana.ma (192.168.205.100) 56(84) bytes of data.
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=1 ttl=64 time=0.383 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=2 ttl=64 time=1.44 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=3 ttl=64 time=0.367 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=4 ttl=64 time=0.803 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=5 ttl=64 time=0.966 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=6 ttl=64 time=0.993 ms
64 bytes from 192.168.205.100 (192.168.205.100): icmp_seq=7 ttl=64 time=1.08 ms
^C
--- dc1.rangana.ma ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6012ms
rtt min/avg/max/mdev = 0.367/0.861/1.444/0.356 ms
rangana@fedora:~$
```

Figure 2.0.15-Ping dc1.rangana.ma



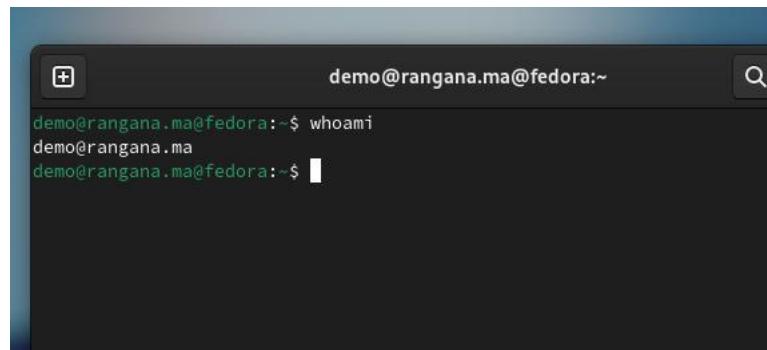
```
Ubuntu 64-bit X | CSA Client X
rangana@fedora:~$ sudo realm discover rangana.ma
rangana.ma
  type: kerberos
  realm-name: RANGANA.MA
  domain-name: rangana.ma
  configured: no
  server-software: active-directory
  client-software: sssd
  required-package: oddjob
  required-package: oddjob-mkhomedir
  required-package: sssd
  required-package: adcli
  required-package: samba-common-tools
rangana@fedora:~$
```

Figure 2.0.16-Discover the Domain



```
Ubuntu 64-bit X | CSA Client X
rangana@fedora:~$ sudo realm join -U Administrator rangana.ma
Password for Administrator:
 * Installing necessary packages: oddjob oddjob-mkhomedir
rangana@fedora:~$
```

Figure 2.0.17-Join the Domain

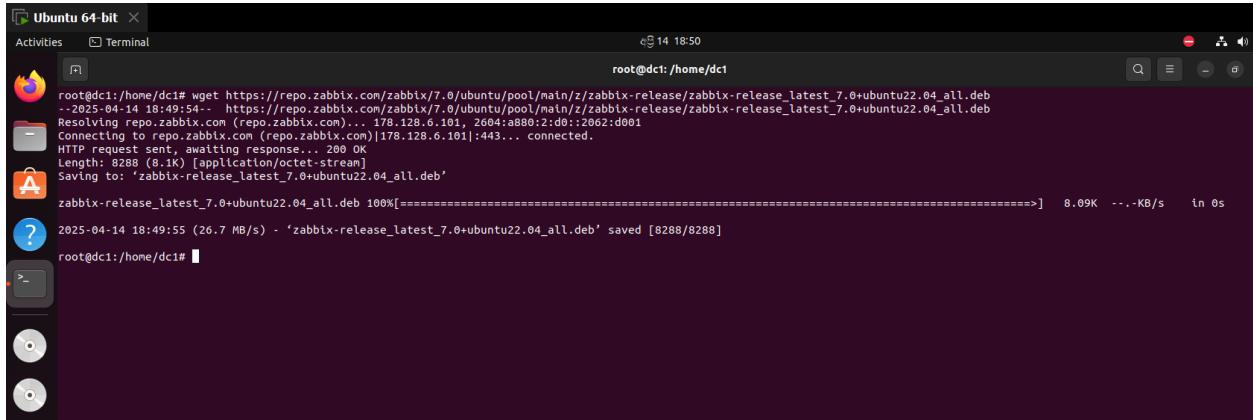


```
demo@rangana.ma@fedora:~$ whoami
demo@rangana.ma
demo@rangana.ma@fedora:~$
```

Figure 2.0.18-Log in with the created demo user in Fedora

5. Install and configure Zabbix

This setup installs and configures Zabbix 7.0 LTS on Ubuntu 22.04 (Jammy) with support for Server, Frontend, and Agent components. It uses Mysql as the database and Apache as the web server, providing a stable and long-term supported monitoring solution for networks and systems.

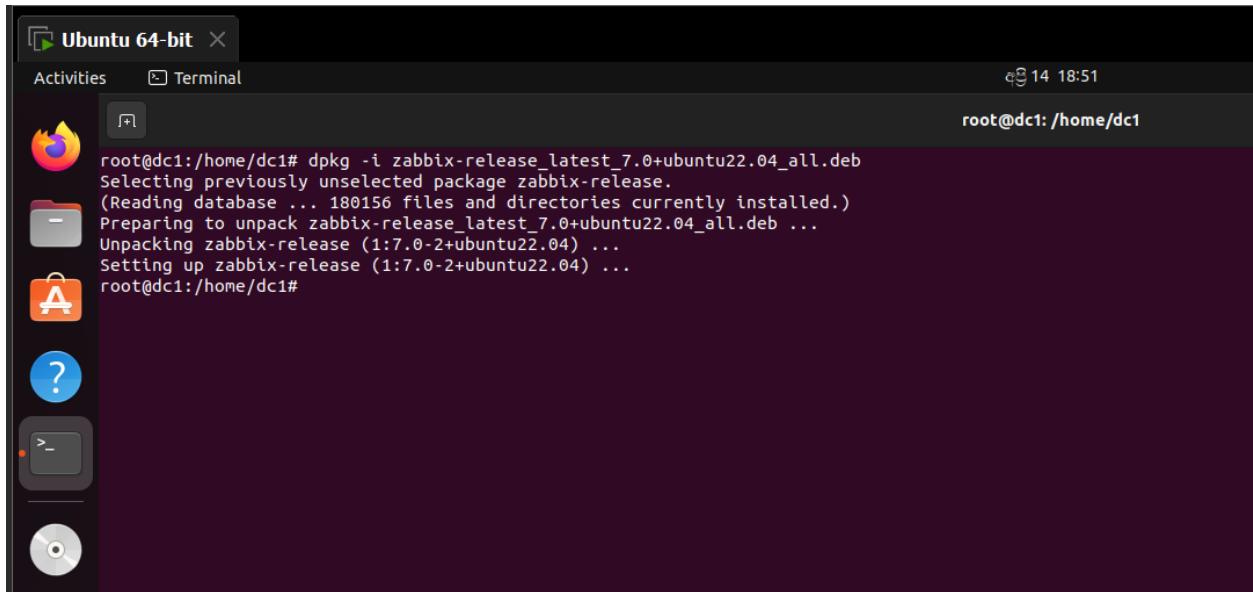


```
Ubuntu 64-bit
Activities Terminal root@dc1:/home/dc1
root@dc1# wget https://repo.zabbix.com/zabbix/7.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest_7.0+ubuntu22.04_all.deb
--2025-04-14 18:49:54-- https://repo.zabbix.com/zabbix/7.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_latest_7.0+ubuntu22.04_all.deb
Resolving repo.zabbix.com (repo.zabbix.com)... 178.128.6.101, 2604:a880:2:d0::2062:d001
Connecting to repo.zabbix.com (repo.zabbix.com)|178.128.6.101|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 8288 (8.1K) [application/octet-stream]
Saving to: 'zabbix-release_latest_7.0+ubuntu22.04_all.deb'

2025-04-14 18:49:55 (26.7 MB/s) - 'zabbix-release_latest_7.0+ubuntu22.04_all.deb' saved [8288/8288]

root@dc1:/home/dc1#
```

Figure 3.0.1-Downloads the latest Zabbix repository .deb package for Ubuntu 22.04



```
Ubuntu 64-bit
Activities Terminal root@dc1:/home/dc1
root@dc1# dpkg -i zabbix-release_latest_7.0+ubuntu22.04_all.deb
Selecting previously unselected package zabbix-release.
(Reading database ... 180156 files and directories currently installed.)
Preparing to unpack zabbix-release_latest_7.0+ubuntu22.04_all.deb ...
Unpacking zabbix-release (1:7.0-2+ubuntu22.04) ...
Setting up zabbix-release (1:7.0-2+ubuntu22.04) ...
root@dc1:/home/dc1#
```

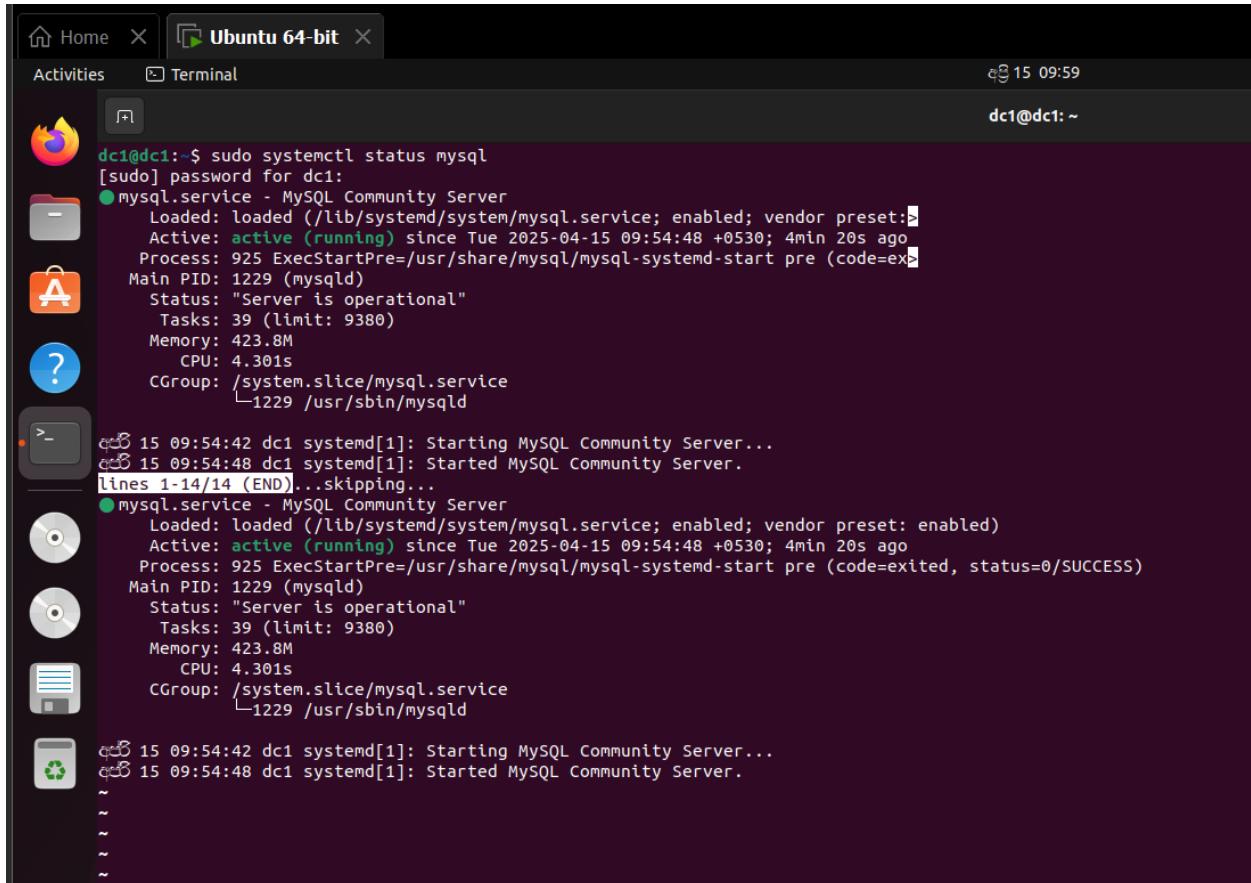
Figure 3.0.2-Installs the downloaded .deb file, which adds the Zabbix APT repository

```
Ubuntu 64-bit
Activities Terminal
c@14 18:53
root@dct:/home/dct#
root@dct:/home/dct# apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
apache2 apache2-bin apache2-utils fonts-dejavu fonts-dejavu-extra fping libapache2-mod-php libapache2-mod-php8.1 libaprpri libaprutil libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-dlap libevent-core-2.1.7 libevent-extra-2.1.7 libevhttp-pthreads-2.1.7 liblbu5 3.0 libmbodbus libmysqclient21 libodbc2 libonig5 libopenlpn10 libsmrmp40 mysql-client mysql-client-8.0 mysql-client-core-8.0 mysql-common php-bcmath php-common php-curl php-gd php-ldap php-mbstring php-mysql php-xml php8.1-bcmath php8.1-cll php8.1-common php8.1-curl php8.1-gd php8.1-ldap php8.1-mbstring php8.1-mysql php8.1-opcache php8.1-readline php8.1-xnl snmp
Suggested packages:
apache2-dbg apache2-suexec-pristine | apache2-suexec-custom www-browser php-pear odbc-postgresql tdsodbc smptrapd zabbix-nginx-conf virtual-mysql-server
The following NEW packages will be installed:
apache2 apache2-bin apache2-utils fonts-dejavu fonts-dejavu-extra fping libapache2-mod-php libapache2-mod-php8.1 libaprpri libaprutil libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-dlap libevent-core-2.1.7 libevent-extra-2.1.7 libevhttp-pthreads-2.1.7 liblbu5 3.0 libmbodbus libmysqclient21 libodbc2 libonig5 libopenlpn10 mysql-client mysql-client-8.0 mysql-client-core-8.0 mysql-common php-bcmath php-common php-curl php-gd php-ldap php-mbstring php-mysql php-xml php8.1-bcmath php8.1-cll php8.1-common php8.1-curl php8.1-gd php8.1-ldap php8.1-mbstring php8.1-mysql php8.1-opcache php8.1-readline php8.1-xnl snmpd zabbix-agent zabbix-apache-conf zabbix-frontend-php zabbix-server-mysql zabbix-sql-scripts
The following packages will be upgraded:
libsmrmp40
1 upgraded, 51 newly installed, 0 to remove and 222 not upgraded.
Need to get 32.8 MB/33.9 MB of archives.
After this operation, 173 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprpri amd64 1.7.0~ubuntu0.22.04.2 [108 kB]
Get:2 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil libaprutil1 amd64 1.6.1~ubuntu0.22.04.2 [92.8 kB]
Get:3 https://repo.zabbix.com/zabbix/7.0/ubuntu jammy/main amd64 zabbix-server-mysql amd64 1.7.0.11+ubuntu2.04 [1,747 kB]
Get:4 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.1~ubuntu0.22.04.2 [11.3 kB]
Get:5 https://repo.zabbix.com/zabbix/7.0/ubuntu jammy/main amd64 libaprutil1-dlap libaprutil1-dlap amd64 1.6.1~ubuntu0.22.04.2 [9,170 B]
Get:6 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main libaprpri libaprpri1 amd64 5.3.0~ubuntu0.22.04.2 [140 kB]
Get:7 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main libaprutil libaprutil1 amd64 2.4.52~ubuntu0.22.04.2 [1,349 kB]
Get:8 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main libaprutil1-dbd-sqlite3 libaprutil1-dbd-sqlite3 amd64 2.4.52~ubuntu0.22.04.2 [165 kB]
Get:9 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main libaprutil1-dlap libaprutil1-dlap amd64 2.4.52~ubuntu0.22.04.2 [89.0 kB]
Get:10 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main libaprpri libaprpri1 amd64 5.3.0~ubuntu0.22.04.2 [97.9 kB]
Get:11 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main libaprpri libaprpri1 amd64 5.9.1~dfsg~ubuntu0.22.04.2 [60.3 kB]
Get:12 http://lk.archive.ubuntu.com/ubuntu jammy/main and01 libbevent-core-2.1.7 libbevent-core-2.1.7-stable~ibuild3 amd64 2.1.12-stable~ibuild3 [93.9 kB]
Get:13 http://lk.archive.ubuntu.com/ubuntu jammy/main and01 libbevent-extra-2.1.7 libbevent-extra-2.1.7-stable~ibuild3 amd64 2.1.12-stable~ibuild3 [65.4 kB]
Get:14 http://lk.archive.ubuntu.com/ubuntu jammy/main and01 libbevent-pthreads-2.1.7 libbevent-pthreads-2.1.7-stable~ibuild3 amd64 2.1.12-stable~ibuild3 [7,642 B]
Get:15 http://lk.archive.ubuntu.com/ubuntu jammy/main and01 libmysql-common libmysql-common all 5.8.0+1.0.8 [7,212 B]
Get:16 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libmysqclient21 libmysqclient21 amd64 8.0.41~ubuntu0.22.04.1 [1,308 kB]
Get:17 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libodbc2 libodbc2 amd64 2.3.9~ubuntu0.1 [159 kB]
Get:18 http://lk.archive.ubuntu.com/ubuntu jammy/main and01 libopenlpn10 libopenlpn10 amd64 2.0.29.0~ubuntu0.1 [517 kB]
Get:19 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-core-8.0 mysql-client-core-8.0 amd64 8.0.41~ubuntu0.22.04.1 [2,716 kB]
```

Figure 3.0.3-Install Zabbix server, frontend, agent

```
Ubuntu 64-bit
Activities Terminal
root@dc1:/home/dc1# sudo apt install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  liblber1 liblber1-fast-perl liblber1-pm-perl liblber1-bin liblber1-perl liblber1oldbl liblber1-template-perl libmecab2 libprotobuf-lite23 mecab-ipadic mecab-ipadic-utf8 mecab-utils
  mysql-server-8.0 mysql-server-core-8.0
Suggested packages:
  liblber-sharedcache-perl mailx tinyca
  liblber1 liblber1-fast-perl liblber1-pm-perl liblber1-bin liblber1-perl liblber1oldbl liblber1-template-perl libmecab2 libprotobuf-lite23 mecab-ipadic mecab-ipadic-utf8 mecab-utils
  mysql-server mysql-server-8.0 mysql-server-core-8.0
  0 upgraded, 15 newly installed, 0 to remove and 222 not upgraded.
Need to get 26.5 MB of archives.
After this operation, 188 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1 amd64 0.3.112-13build1 [7,176 B]
Get:2 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 libmecab2 amd64 0.996-14build9 [129 kB]
Get:3 http://lx.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libprotobuf-lite23 amd64 3.12.4-1ubuntu7.22.04.1 [209 kB]
Get:4 http://lx.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libmecab2 amd64 0.996-14build9 [17.6 kB]
Get:5 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 mysql-server-core-8.0 amd64 8.0.41-0ubuntu0.22.04.1 [1,443 kB]
Get:6 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1oldbl amd64 2.4.2-1 [188 kB]
Get:7 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1oldbl amd64 2.4.2-2 [28.0 kB]
Get:8 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1-perl amd64 0.82-2ds1build1 [22.8 kB]
Get:9 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1-fast-perl all 1.12.15-1 [10.5 kB]
Get:10 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1oldbl amd64 2.4.2-2build2 [11.2 kB]
Get:11 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 liblber1-template-perl all 2.97-1.1 [59.1 kB]
Get:12 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 mecab-utils amd64 0.996-14build9 [4,850 B]
Get:13 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 mecab-ipadic all 2.7.0-20070801+main-3 [6,718 kB]
Get:14 http://lx.archive.ubuntu.com/ubuntu jammy/main amd64 mecab-ipadic-utf8 all 2.7.0-20070801+main-3 [4,384 B]
Get:15 http://lx.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-server all 8.0.41-0ubuntu0.22.04.1 [9,460 B]
Fetched 26.5 MB in 8s (3,497 kB/s)
Preconfiguring packages ...
Selecting previously unselected package liblber1:amd64.
(Reading database ... 183633 files and directories currently installed.)
Preparing to unpack .../00-liblber1_0.3.112-13build1_amd64.deb ...
Unpacking liblber1:amd64 (0.3.112-13build1) ...
Selecting previously unselected package libmecab2:amd64.
Preparing to unpack .../01-libmecab2_0.996-14build9_amd64.deb ...
Unpacking libmecab2:amd64 (0.996-14build9) ...
Selecting previously unselected package libprotobuf-lite23:amd64.
Preparing to unpack .../02-libprotobuf-lite23_3.12.4-1ubuntu7.22.04.1_amd64.deb ...
```

Figure 3.0.4-Install Mysql server



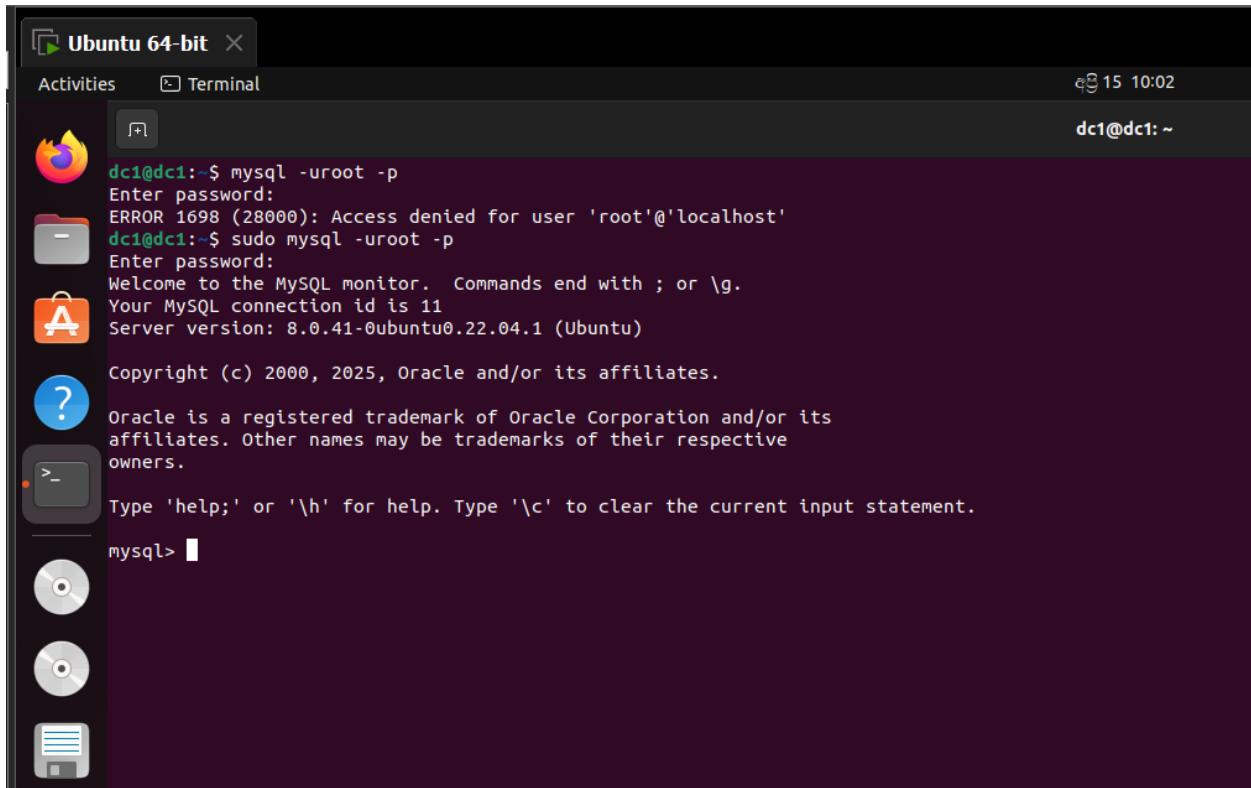
The screenshot shows a terminal window titled "Ubuntu 64-bit" running on an Ubuntu system. The command `sudo systemctl status mysql` was run, and the output is displayed. The MySQL service is shown as active (running) since the previous day, with a main PID of 1229 and various resource usage statistics. The terminal also shows log messages from the MySQL daemon starting and then exiting.

```
dc1@dc1:~$ sudo systemctl status mysql
[sudo] password for dc1:
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2025-04-15 09:54:48 +0530; 4min 20s ago
     Process: 925 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 1229 (mysqld)
      Status: "Server is operational"
        Tasks: 39 (limit: 9380)
       Memory: 423.8M
          CPU: 4.301s
        CGroup: /system.slice/mysql.service
                  └─1229 /usr/sbin/mysqld

May 15 09:54:42 dc1 systemd[1]: Starting MySQL Community Server...
May 15 09:54:48 dc1 systemd[1]: Started MySQL Community Server.
lines 1-14/14 (END)...skipping...
● mysql.service - MySQL Community Server
   Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2025-04-15 09:54:48 +0530; 4min 20s ago
     Process: 925 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 1229 (mysqld)
      Status: "Server is operational"
        Tasks: 39 (limit: 9380)
       Memory: 423.8M
          CPU: 4.301s
        CGroup: /system.slice/mysql.service
                  └─1229 /usr/sbin/mysqld

May 15 09:54:42 dc1 systemd[1]: Starting MySQL Community Server...
May 15 09:54:48 dc1 systemd[1]: Started MySQL Community Server.
~
~
~
~
```

Figure 3.0.5-mysql status



The screenshot shows a terminal window titled "Ubuntu 64-bit" running on an Ubuntu desktop environment. The terminal session starts with a failed attempt to log in as root without sudo:

```
dc1@dc1:~$ mysql -uroot -p
Enter password:
ERROR 1698 (28000): Access denied for user 'root'@'localhost'
```

After using sudo to gain root privileges, the MySQL monitor is successfully connected:

```
dc1@dc1:~$ sudo mysql -uroot -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.41-Ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

The MySQL prompt "mysql>" is visible at the bottom of the terminal window.

Figure 3.0.6-Loggin mysql server as root user

```
mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;
Query OK, 1 row affected (0.00 sec)
```

Figure 3.0.7-Creates a new database named zabbix

```
mysql> create user zabbix@localhost identified by '123';
Query OK, 0 rows affected (0.01 sec)

mysql>
```

Figure 3.0.8-Creates a new user called zabbix.

```
mysql> grant all privileges on zabbix.* to zabbix@localhost;
Query OK, 0 rows affected (0.01 sec)
```

```
mysql>
```

Figure 3.0.9-Gives the Zabbix user full access to the Zabbix database only

```
mysql> set global log_bin_trust_function_creators = 1;
Query OK, 0 rows affected, 1 warning (0.00 sec)

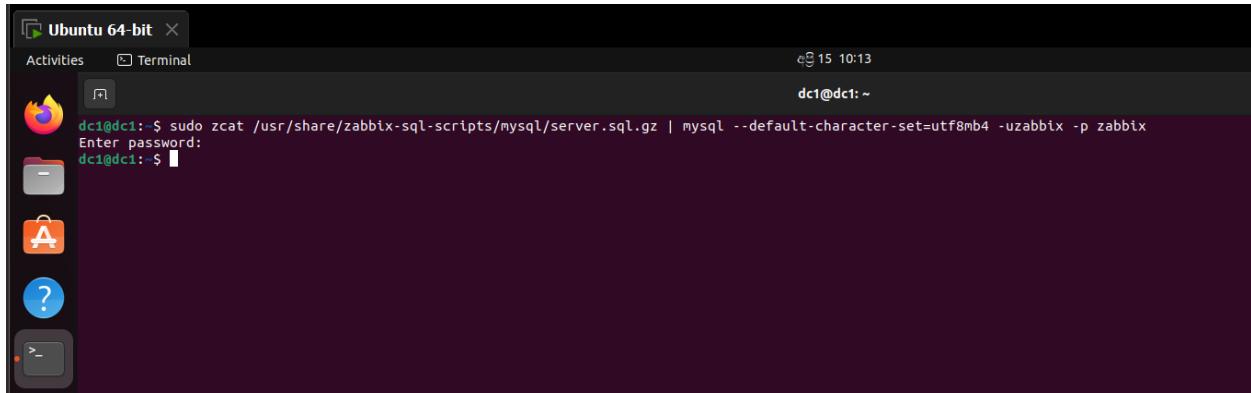
mysql> quit
Bye
dc1@dc1:~$
```

Figure 3.0.10-Create stored functions.

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
| zabbix         |
+-----+
5 rows in set (0.01 sec)

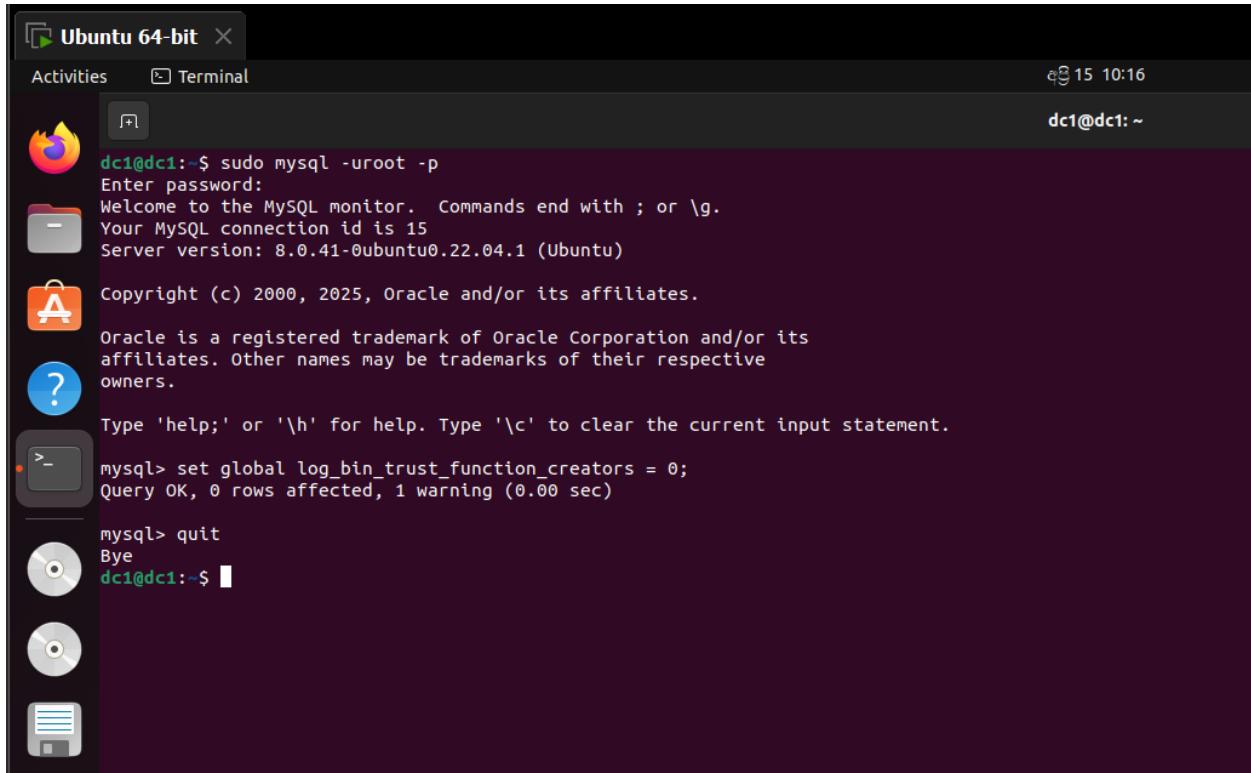
mysql> |
```

Figure 3.0.11-Show databases



A screenshot of an Ubuntu 64-bit terminal window. The title bar says "Ubuntu 64-bit". The terminal shows the command: `sudo zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -uzabbix -p zabbix`. It prompts for a password and ends with `dc1@dc1:~$`.

Figure 3.0.12-Import the Zabbix database structure and data into the Mysql database



A screenshot of an Ubuntu 64-bit terminal window. The title bar says "Ubuntu 64-bit". The terminal shows the command: `sudo mysql -uroot -p`. It prompts for a password and then connects to the MySQL monitor. The output includes: "Welcome to the MySQL monitor. Commands end with ; or \g.", "Your MySQL connection id is 15", "Server version: 8.0.41-0ubuntu0.22.04.1 (Ubuntu)", "Copyright (c) 2000, 2025, Oracle and/or its affiliates.", "Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.", "Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.", "mysql> set global log_bin_trust_function_creators = 0;", "Query OK, 0 rows affected, 1 warning (0.00 sec)", "mysql> quit", "Bye", and ends with `dc1@dc1:~$`.

Figure 3.0.13-Disable the `log_bin_trust_function_creators` option after importing the database schema

```

Ubuntu 64-bit x
Activities   Gedit
dc1@dc1: ~
e 15 10:18
dc1@dc1: ~

zabbix_server.conf
/etc/zabbix

106
107 DBName=zabbix
108
109 ### Option: DBSchema
110 # Schema name. Used for PostgreSQL.
111 #
112 # Mandatory: no
113 # Default:
114 # DBSchema=
115
116 ### Option: DBUser
117 # Database user.
118 #
119 # Mandatory: no
120 # Default:
121 # DBUser=
122
123 DBUser=zabbix
124
(gedit)
125 ### Option: DBPassword
126 # Database password.
127 # Comment this line if no password is used.
128 #
129 # Mandatory: no
130 # Default:
131 DBPassword=123
132
133 ### Option: DBSocket
134 # Path to MySQL socket.
135 #
136 # Mandatory: no
137 # Default:
138 # DBSocket=
139
140 ### Option: DBPort
141 # Database port when not using local socket.
142 # If the Net Service Name connection method is used to connect to Oracle database, the

```

Plain Text Tab Width: 8 Ln 131, Col 15 INS

Figure 3.0.14-Edit file zabbix_server.conf

```

Ubuntu 64-bit x
Activities Terminal
dc1@dc1: ~
e 15 10:21
dc1@dc1: ~

dc1@dc1: $ sudo systemctl restart zabbix-server zabbix-agent apache2
dc1@dc1: $ sudo systemctl status zabbix-server zabbix-agent apache2
Synchronizing state of zabbix-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable zabbix-server
Synchronizing state of zabbix-agent.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable zabbix-agent
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
Created symlink /etc/systemd/system/multi-user.target.wants/zabbix-server.service → /lib/systemd/system/zabbix-server.service.
dc1@dc1: $ sudo systemctl status zabbix-server zabbix-agent apache2
● zabbix-server.service - Zabbix Server
   Loaded: loaded (/lib/systemd/system/zabbix-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2025-04-15 20:57:0530; 48s ago
     Main PID: 4986 (zabbix_server)
        Tasks: 7 (limit: 9380)
       Memory: 73.2M
          CPU: 51ms
         CGroup: /system.slice/zabbix-server.service
             └─4986 /usr/sbin/zabbix_server -c /etc/zabbix/zabbix_server.conf

-4993 "/usr/sbin/zabbix_server: ha manager" [processed 0 events, updated 0 event tags, deleted 0 problems, synced 0 service updates, idle 5.006308 sec during 5.006308 sec]
-4997 "/usr/sbin/zabbix_server: service manager #1 [processed 0 events, updated 0 event tags, deleted 0 problems, synced 0 service updates, idle 5.006308 sec during 5.006308 sec]"
-4999 "/usr/sbin/zabbix_server: configuration syncer [synced configuration in 0.017658 sec, idle 10 sec]"
-5001 "/usr/sbin/zabbix_server: alert manager #1 [sent 0, failed 0 alerts, idle 5.008671 sec during 5.008732 sec]"
-5002 "/usr/sbin/zabbix_server: alerter #1 started"
-5003 "/usr/sbin/zabbix_server: alerter #2 started"
-5004 "/usr/sbin/zabbix_server: alerter #3 started"
-5005 "/usr/sbin/zabbix_server: preprocessing manager #1 [queued 0, processed 0 values, idle 5.008121 sec during 5.008271 sec]"
-5007 "/usr/sbin/zabbix_server: lld manager #1 [processed 0 LLD rules, idle 5.005303sec during 5.005338 sec]"
-5009 "/usr/sbin/zabbix_server: lld worker #1 [processed 1 LLD rules, idle 15.856053 sec during 15.862999 sec]"
-5010 "/usr/sbin/zabbix_server: lld worker #2 [processed 1 LLD rules, idle 11.960473 sec during 11.995221 sec]"
-5026 "/usr/sbin/zabbix_server: http poller #1 [got 0 values in 0.000043 sec, idle 5 sec]"
-5032 "/usr/sbin/zabbix_server: browser poller #1 [got 0 values in 0.000010 sec, idle 5 sec]"
-5034 "/usr/sbin/zabbix_server: discovery manager #1 [processing 0 rules, 0 unsaved checks]"
-5035 "/usr/sbin/zabbix_server: history syncer #1 [processed 1 values, 1 triggers in 0.002542 sec, idle 1 sec]"
-5038 "/usr/sbin/zabbix_server: history syncer #2 [processed 0 values, 0 triggers in 0.000019 sec, idle 1 sec]"
-5039 "/usr/sbin/zabbix_server: history syncer #3 [processed 0 values, 0 triggers in 0.000008 sec, idle 1 sec]"
-5042 "/usr/sbin/zabbix_server: escalator #1 [processed 0 escalations in 0.000550 sec, idle 3 sec]"
-5044 "/usr/sbin/zabbix_server: proxy poller #1 [exchanged data with 0 proxies in 0.000006 sec, idle 5 sec]"
-5045 "/usr/sbin/zabbix_server: self-monitoring [processed data in 0.000021 sec, idle 1 sec]"

```

Figure 3.0.15-Start Zabbix server and agent processes

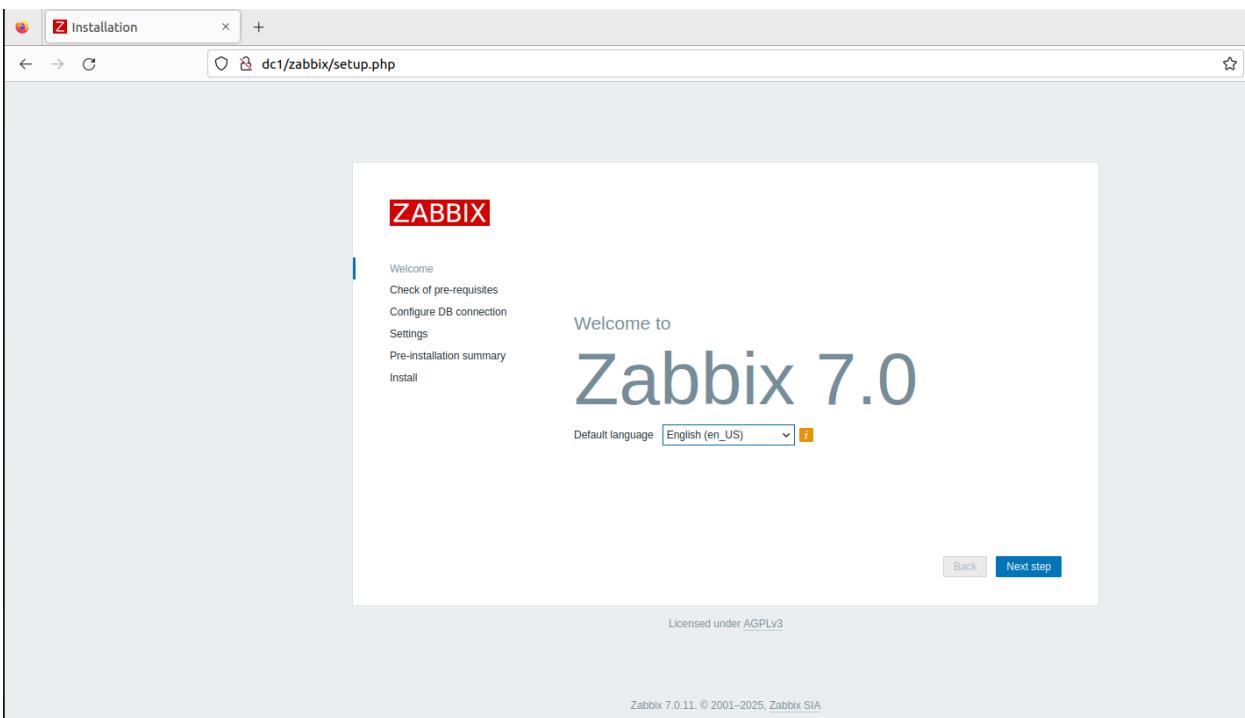


Figure 3.0.16-Search <https://dc1/zabbix>

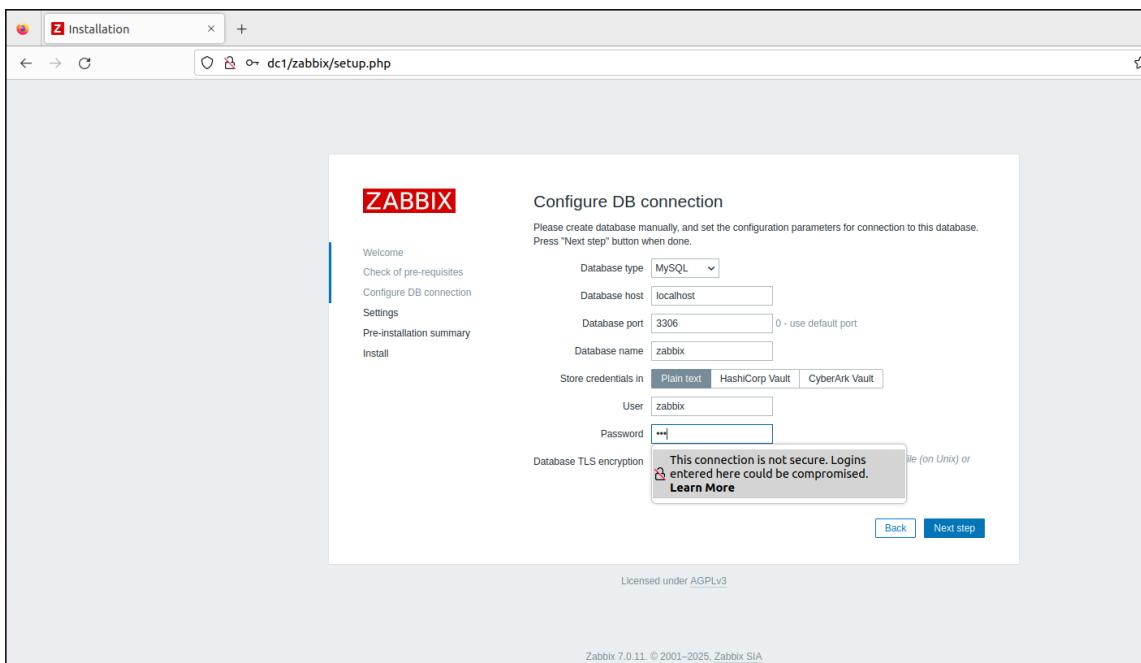


Figure 3.0.17-Configure Zabbix username, password and port

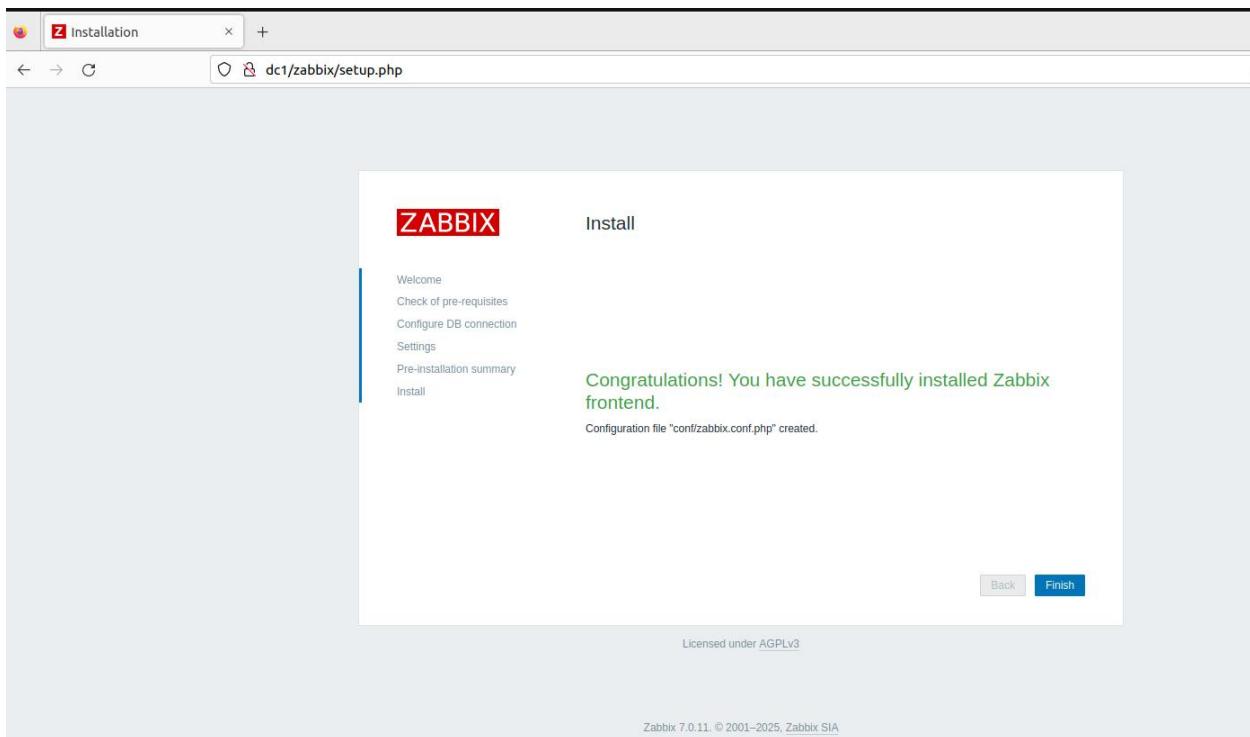


Figure 3.0.18-Successfully installed Zabbix server

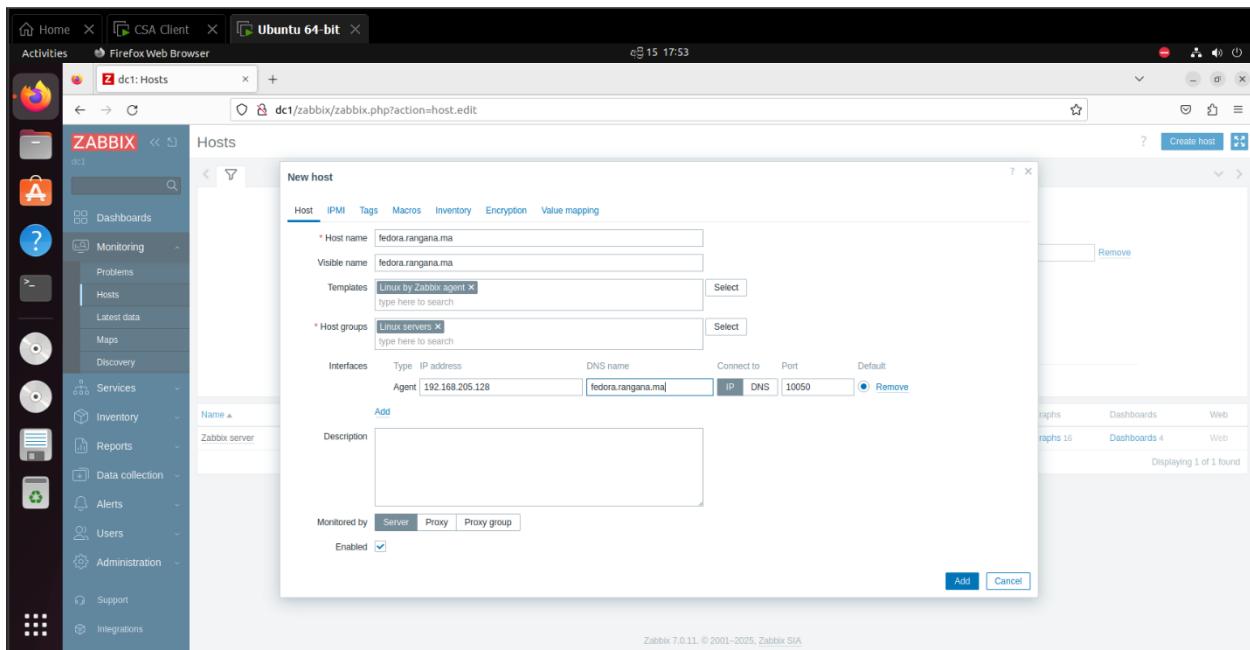


Figure 3.0.19-Create a new host using the Zabbix server

6. Install the Zabbix agent on the clients and connect to the Zabbix server

6.1 Connect the Windows client to Zabbix

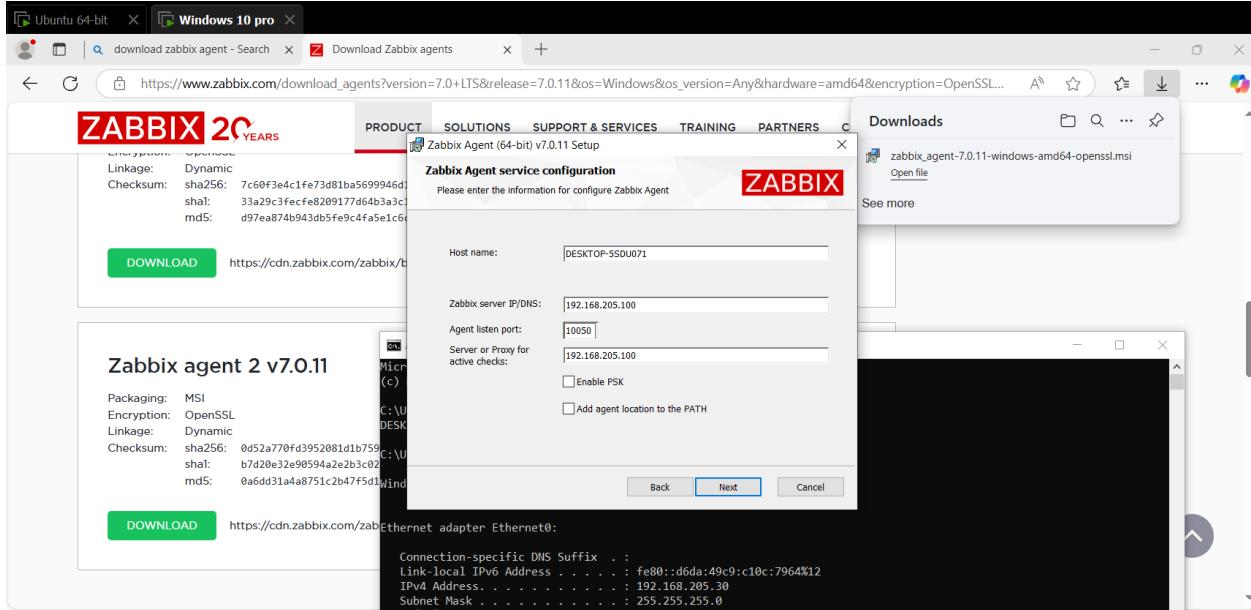


Figure 4.0.1-Add Zabbix agent

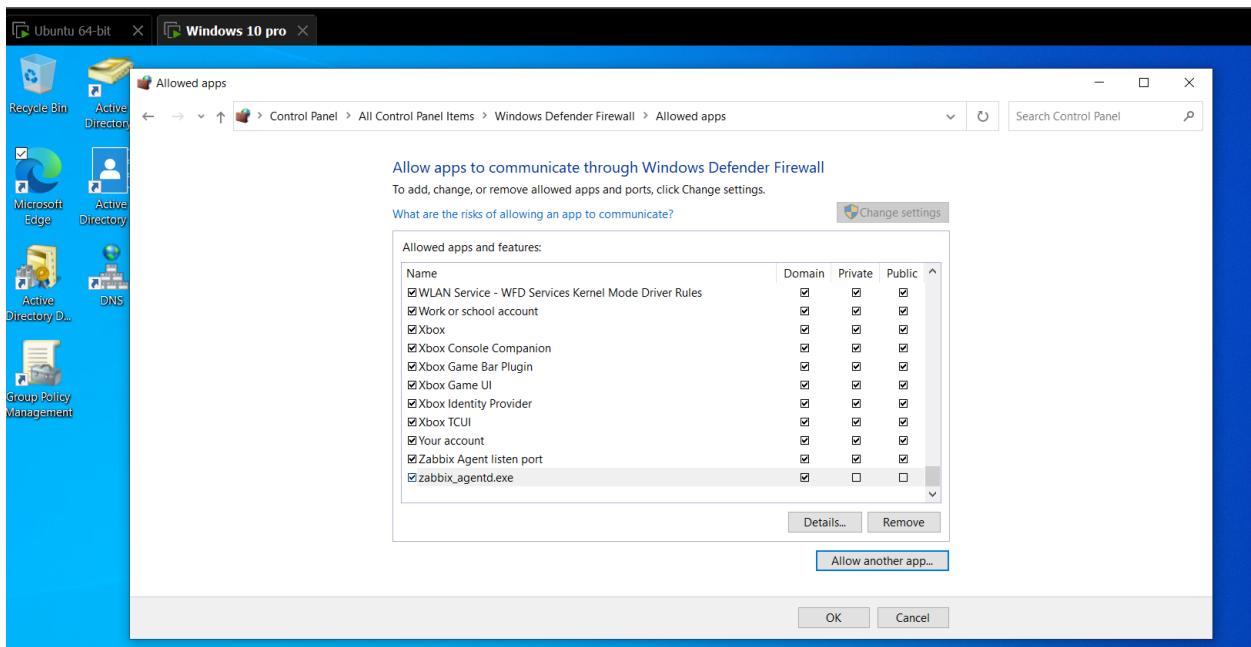


Figure 4.0.2-Give the firewall access to Zabbix agent

Name	Interface	Availability	Tags	Status	Latest data	Problems	Graphs	Dashboards	Web
DESKTOP-5SDU071	192.168.205.30:10050	zbx	<code>class: os target: windows</code>	Enabled	Latest data 109	1	Graphs 12	Dashboards 3	Web
fedora.rangana.ma	192.168.205.128:10050	zbx	<code>class: os target: linux</code>	Enabled	Latest data 43	1	Graphs 8	Dashboards 3	Web
Zabbix server	127.0.0.1:10050	zbx	<code>class: os class: software target: linux</code> + ...	Enabled	Latest data 148	Problems	Graphs 16	Dashboards 4	Web

Displaying 3 of 3 found

Figure 4.0.3-Added Windows client to the Zabbix server

6.2 Connect the Fedora client to Zabbix

```
Ubuntu 64-bit × [ CSA Client ×
rangana@fedora: $ sudo dnf install zabbix-agent
[sudo] password for rangana:
Last metadata expiration check: 1:04:23 ago on Tue 15 Apr 2025 03:00:28 AM EDT.
Dependencies resolved.
=====
Package           Architecture      Version       Repository   Size
=====
Installing:
zabbix-agent     x86_64          1:6.0.33-2.fc39   updates      297 k
Installing dependencies:
zabbix           x86_64          1:6.0.33-2.fc39   updates      729 k
zabbix-selinux   noarch          1:6.0.33-2.fc39   updates      27 k
Transaction Summary
=====
Install 3 Packages

Total download size: 1.0 M
Installed size: 3.5 M
Is this ok [y/N]: y
Downloading Packages:
(1/3): zabbix-selinux-6.0.33-2.fc39.noarch.rpm           20 kB/s | 27 kB   00:01
(2/3): zabbix-agent-6.0.33-2.fc39.x86_64.rpm             137 kB/s | 297 kB  00:02
(3/3): zabbix-6.0.33-2.fc39.x86_64.rpm                  280 kB/s | 729 kB  00:02
Total                                         277 kB/s | 1.0 MB  00:03

Preparing:
  Running scriptlet: zabbix-selinux-1:6.0.33-2.fc39.noarch 1/1
  Installing       : zabbix-selinux-1:6.0.33-2.fc39.noarch 1/3
  Running scriptlet: zabbix-6.0.33-2.fc39.noarch            1/3

```

Figure 4.0.4-Install Zabbix agent

```

Home X | Ubuntu 64-bit X | CSA Client X
rangana@fedora:~$ sudo gedit /etc/zabbix/zabbix_agentd.conf
[sudo] password for rangana:
rangana@fedora:~$ hostname
fedora.rangana.ma
rangana@fedora:~$ sudo gedit /etc/zabbix/zabbix_agentd.conf
rangana@fedora:~$ sudo systemctl enable --now zabbix-agent
Created symlink /etc/systemd/system/multi-user.target.wants/zabbix-agent.service → /usr/lib/systemd/system/zabbix-agent.service.
rangana@fedora:~$ sudo systemctl status --now zabbix-agent
● zabbix-agent.service - Zabbix Monitor Agent
   Loaded: loaded (/usr/lib/systemd/system/zabbix-agent.service; enabled; preset: disabled)
   Drop-In: /usr/lib/systemd/system/service.d
             └─10-timeout-abort.conf
     Active: active (running) since Tue 2025-04-15 06:42:07 EDT; 15s ago
       Main PID: 4097 (zabbix_agentd)
         Tasks: 6 (limit: 7597)
        Memory: 5.1M
          CPU: 12ms
        CGroup: /system.slice/zabbix-agent.service
                  ├─4097 /usr/sbin/zabbix_agentd -f
                  ├─4101 "/usr/sbin/zabbix_agentd: collector [idle 1 sec]"
                  ├─4102 "/usr/sbin/zabbix_agentd: listener #1 [waiting for connection]"
                  ├─4103 "/usr/sbin/zabbix_agentd: listener #2 [waiting for connection]"
                  ├─4104 "/usr/sbin/zabbix_agentd: listener #3 [waiting for connection]"
                  └─4105 "/usr/sbin/zabbix_agentd: active checks #1 [idle 1 sec]"

Apr 15 06:42:07 fedora.rangana.ma systemd[1]: Started zabbix-agent.service - Zabbix Monitor Agent.
Apr 15 06:42:07 fedora.rangana.ma zabbix_agentd[4097]: Starting Zabbix Agent [Zabbix server]. Zabbix 6.0.33 (revision e05e6ba9dca).
Apr 15 06:42:07 fedora.rangana.ma zabbix_agentd[4097]: Press Ctrl+C to exit.
rangana@fedora:~$ █

```

Figure 4.0.5-start and enable the agent

7. Errors Occurred During the Configuration Process

```

PS C:\Users\malee> Add-computer -DomainName "rangana.ma" -Restart
cmdlet Add-Computer at command pipeline position 1
Supply values for the following parameters:
Credential
Add-computer : Computer 'DESKTOP-5SDU071' failed to join domain 'rangana.ma' from its current workgroup 'WORKGROUP' with following error message: Access is denied.
At line:1 char:1
+ Add-computer -DomainName "rangana.ma" -Restart
+ CategoryInfo          : OperationStopped: (DESKTOP-5SDU071:String) [Add-Computer], InvalidOperationException
+ FullyQualifiedErrorId : FailToJoinDomainFromWorkgroup,Microsoft.PowerShell.Commands.AddComputerCommand

```

Figure 5.0.1-Issue with PowerShell syntax or session permission in Windows

This issue was caused by insufficient permissions in the PowerShell session or incorrect credentials. To resolve it, I switched to using the **Windows GUI method**:

1. Opened **System Properties > Computer Name > Change**.
2. Selected **Domain**, entered rangana.ma, and provided valid **domain administrator credentials**.
3. Successfully joined the domain and **restarted the system**.

This ensured the client was added to the Samba AD domain without further issues.

```
rangana@fedora: $ sudo tail -f /var/log/zabbix/zabbix_agentd.log
5052:20250415:083111.931 Active check configuration update started to fail
5049:20250415:083136.943 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5050:20250415:083236.943 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5051:20250415:083337.944 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5051:20250415:083437.943 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5051:20250415:083537.945 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5051:20250415:083637.945 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5049:20250415:083737.944 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5049:20250415:083837.944 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5049:20250415:083937.944 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
5049:20250415:084037.943 failed to accept an incoming connection: connection from "192.168.205.100" rejected, allowed hosts: "127.0.0.1"
^C
rangana@fedora: $ telnet 192.168.205.128 10050
Trying 192.168.205.128...
Connected to 192.168.205.128.
Escape character is '^J'.
Connection closed by foreign host.
rangana@fedora: $
```

Figure 5.0.2-Try to fix the Zabbix agent

In the Zabbix agent configuration file (/etc/zabbix/zabbix_agentd.conf), the Server= and/or ServerActive= parameter was set to 127.0.0.1, which means the agent only accepted connections from the local machine.

Resolution Steps:

1. **Edit the Zabbix Agent Configuration:**

```
sudo nano /etc/zabbix/zabbix_agentd.conf
```

2. **Update the Server and ServerActive parameters:**

Server=127.0.0.1

ServerActive=127.0.0.1

To:

Server=127.0.0.1,192.168.205.100

ServerActive=192.168.205.100

This allows passive checks from both localhost and your Zabbix server, and active checks to the Zabbix server.

3. **Restart the Zabbix Agent:**

```
sudo systemctl restart zabbix-agent
```

Figure 5.3-Zabbix Agent on Fedora machine is not responding to server messages

In the agent configuration file (/etc/zabbix/zabbix_agentd.conf), the Server= parameter was limited to 127.0.0.1, so it rejected connections from the actual Zabbix server's IP (192.168.205.100).

- **Open the agent config file:**

```
sudo nano /etc/zabbix/zabbix_agentd.conf
```

- **Update the access control parameters:**

```
Server=127.0.0.1
```

```
ServerActive=127.0.0.1
```

After:

```
Server=127.0.0.1,192.168.205.100
```

```
ServerActive=192.168.205.100
```

This allows the Zabbix server (at 192.168.205.100) to perform both passive and active checks.

- **Restart the agent to apply changes:**

```
sudo systemctl restart zabbix-agent
```

- **Verify Firewall Rules:**

Make port 10050 is open to the Zabbix server:

```
sudo firewall-cmd --add-port=10050/tcp --permanent
```

```
sudo firewall-cmd --reload
```

- **Check Agent Logs for Confirmation:**

```
sudo tail -f /var/log/zabbix/zabbix_agentd.log
```

8. References

1. Samba Team, *Samba Documentation*. [Online]. Available:
<https://www.samba.org/samba/docs/>
2. Zabbix LLC, *Zabbix Documentation 6.0 LTS*. [Online]. Available:
<https://www.zabbix.com/documentation/6.0/manual>
3. Red Hat, “How to configure Samba on Linux,” *Red Hat Customer Portal*, 2023. [Online]. Available: <https://access.redhat.com/solutions/40492>
4. Zabbix LLC, “Installing Zabbix Agent on Linux,” *Zabbix Documentation 6.0*, 2023. [Online]. Available:
https://www.zabbix.com/documentation/6.0/en/manual/installation/install_from_package
5. Fedora Project, “Firewalld Documentation,” *Fedora Docs*, 2023. [Online]. Available:
<https://docs.fedoraproject.org/en-US/quick-docs/firewalld/>