UNIVERSITY OF Baguio SCHOOL OF INFORMATION AND TECHNOLOGY		
NAME: FLEX D. PEDRO	DATE PERFORMED:11-13-24	
Section:IDC1	DATE SUBMITTED:11-13-24	/50

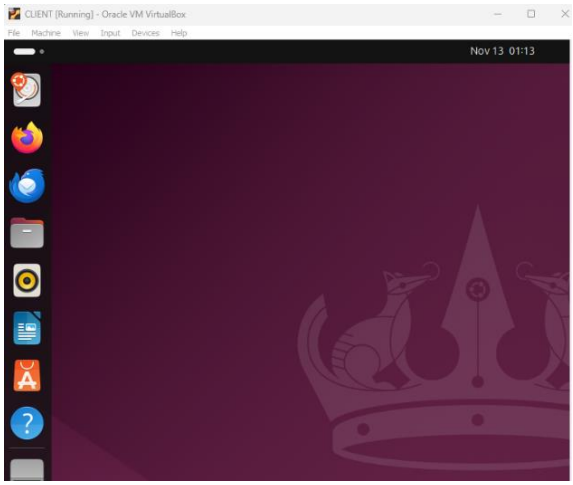
SYSADM1 – Kerberos Lab Activity: A step-by-step Guide

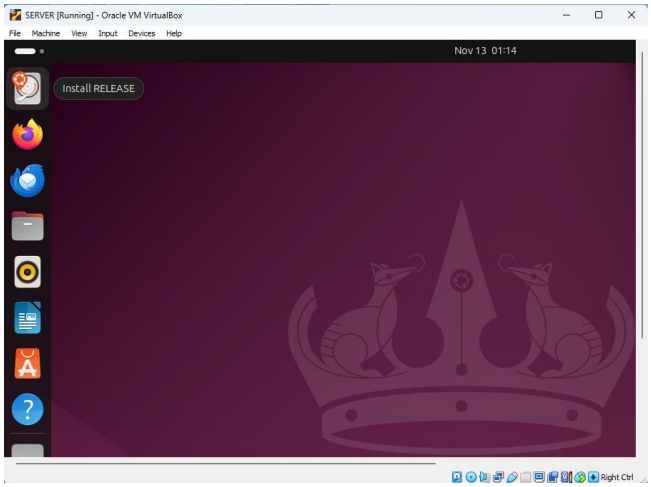
Objective:

Set up a basic Kerberos authentication system to understand how Kerberos manages secure logins through ticket-based access.

Setup Requirements:

- Two VMs in Oracle VM, both running a Linux distribution like Ubuntu or CentOS.





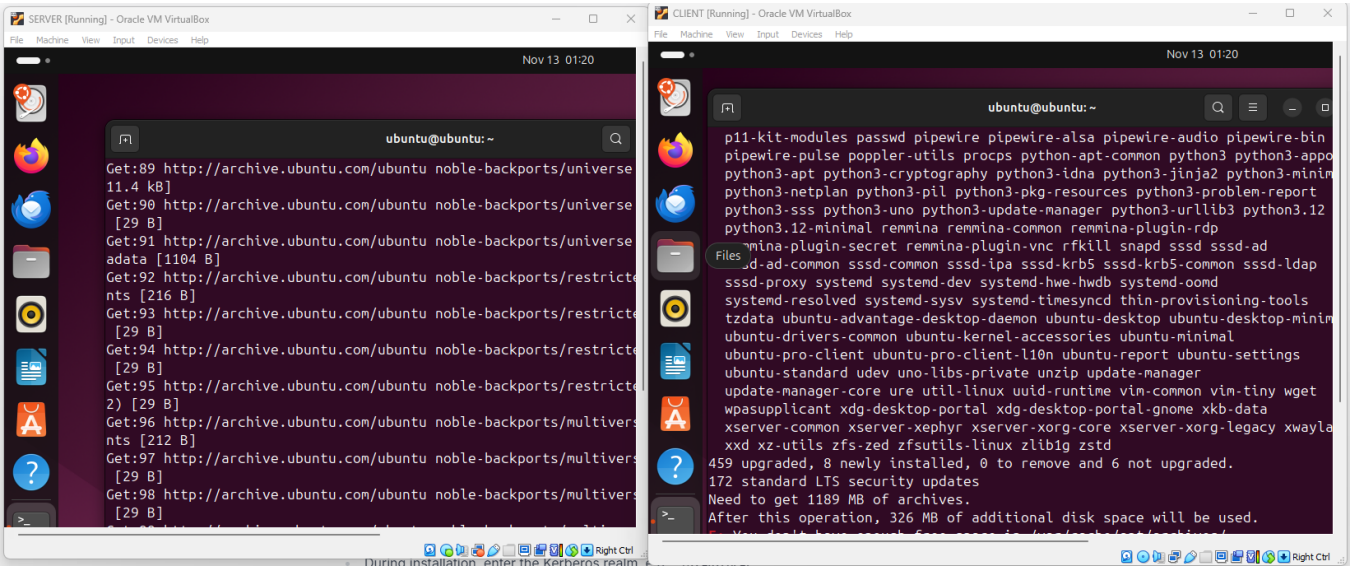
- VM1: Kerberos Server
- VM2: Kerberos Client

Step 1: Initial Setup and Package Installation

1. **Update Packages on Both VMs:**
    - Open a terminal on each VM and run:

bash

sudo apt update && sudo apt upgrade –y



2. **Install Kerberos Server Packages on VM1 (Kerberos Server):**
    - In VM1, install the Kerberos Key Distribution Center (KDC) and admin server:

bash

sudo apt install krb5-kdc krb5-admin-server -y

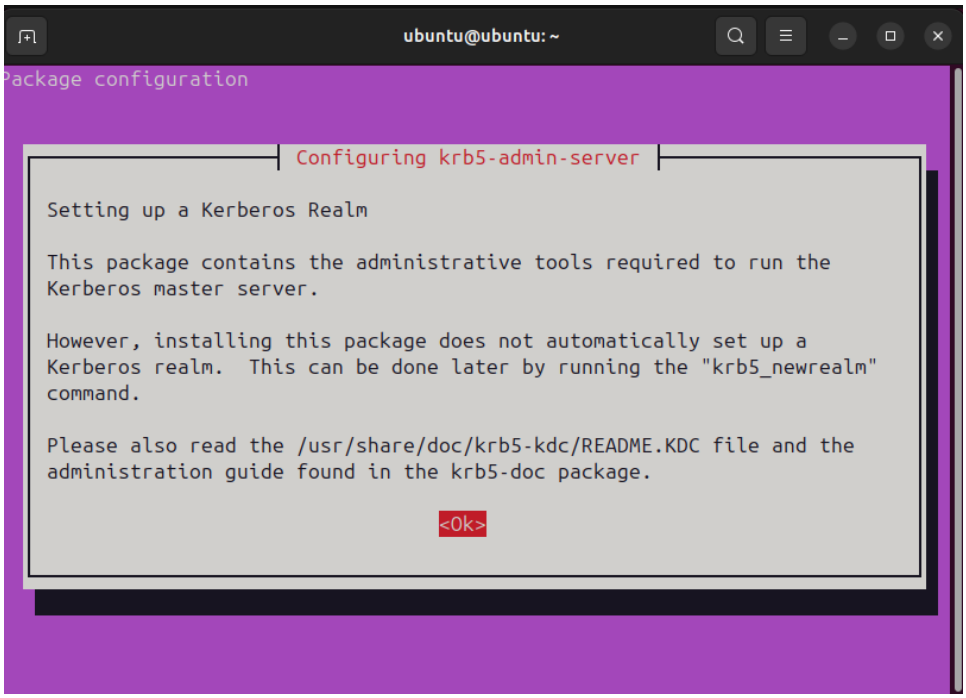
3. **Install Kerberos Client Package on VM2 (Kerberos Client):**
    - In VM2, install the Kerberos client software:

bash

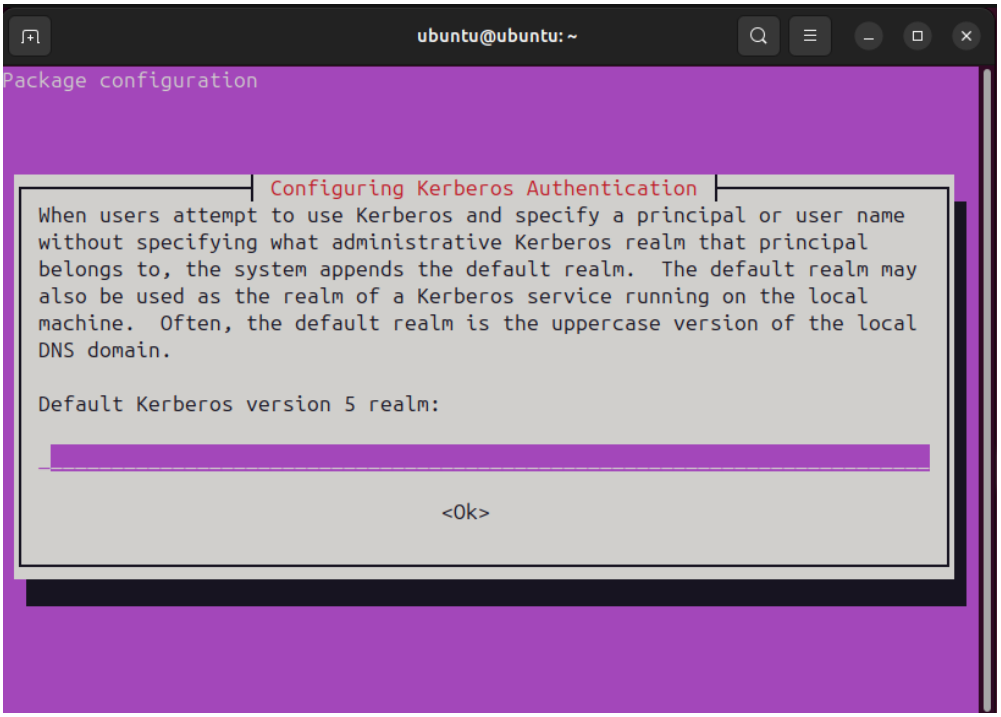
sudo apt install krb5-user -y

- During installation, when prompted, enter the Kerberos realm you plan to set up, e.g., MYLAB.LOCAL.

SERVER



CLIENT



Step 2: Configure the Kerberos Server (VM1)

1. **Edit the Kerberos Configuration File:**
    - Open /etc/krb5.conf for editing:

bash

sudo nano /etc/krb5.conf

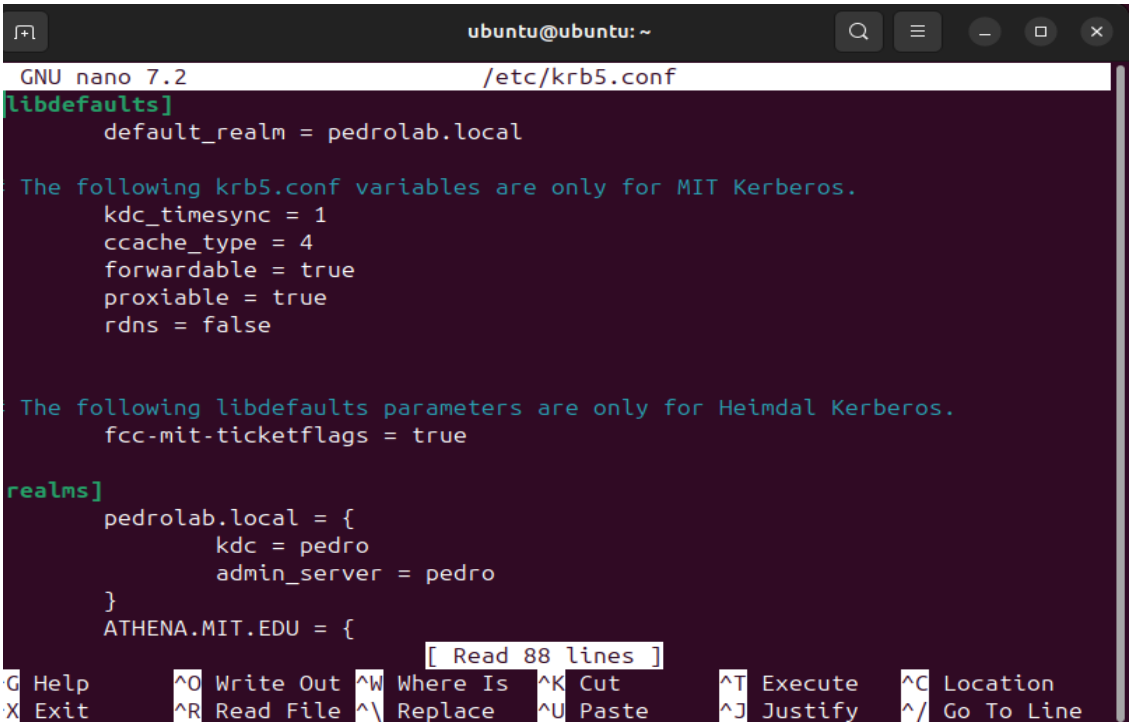
- Set the realm as MYLAB.LOCAL. You should also specify the KDC and admin server as VM1's hostname or IP address:

ini

```
[libdefaults]
   default_realm = MYLAB.LOCAL


[realms]
   MYLAB.LOCAL = {
      kdc = <VM1_IP_or_hostname>
      admin_server = <VM1_IP_or_hostname>
   }
```

- Save and close the file (Ctrl+X, then Y, and Enter to confirm).



2. **Initialize the Kerberos Database:**

- Create the database for the Kerberos realm:

bash

sudo krb5_newrealm

- You will be prompted to set a password for the Kerberos database.

```
ubuntu@ubuntu:~$ sudo krb5_newrealm
This script should be run on the master KDC/admin server to initialize
a Kerberos realm.  It will ask you to type in a master key password.
This password will be used to generate a key that is stored in
/etc/krb5kdc/stash.  You should try to remember this password, but it
is much more important that it be a strong password than that it be
remembered.  However, if you lose the password and /etc/krb5kdc/stash,
you cannot decrypt your Kerberos database.
Initializing database '/var/lib/krb5kdc/principal' for realm 'pedrolab.local',
master key name 'K/M@pedrolab.local'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key:
```

3. **Start and Enable the Kerberos Services:**
   - Start the KDC and admin server, and ensure they start automatically on boot:

bash

sudo systemctl start krb5-kdc

sudo systemctl start krb5-admin-server

sudo systemctl enable krb5-kdc

sudo systemctl enable krb5-admin-server

```
ubuntu@ubuntu:~$ sudo systemctl start krb5-kdc
ubuntu@ubuntu:~$ sudo systemctl start krb5-admin-server
ubuntu@ubuntu:~$ sudo systemctl enable krb5-kdc
Synchronizing state of krb5-kdc.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable krb5-kdc
ubuntu@ubuntu:~$ sudo systemctl enable krb5-admin-server
Synchronizing state of krb5-admin-server.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable krb5-admin-server
```

Step 3: Set Up a Kerberos User Principal

1. **Create a New User Principal:**
   - Run the following command to create a test user in the Kerberos realm:

bash

sudo kadmin.local -q "addprinc testuser@MYLAB.LOCAL"

- Set a password for testuser.; PEDRO

```
ubuntu@ubuntu:~$ sudo kadmin.local -q "addprinc testuser@PEDROLAB.LOCAL"
Authenticating as principal root/admin@pedrolab.local with password.
No policy specified for testuser@PEDROLAB.LOCAL; defaulting to no policy
Enter password for principal "testuser@PEDROLAB.LOCAL":
Re-enter password for principal "testuser@PEDROLAB.LOCAL":
Principal "testuser@PEDROLAB.LOCAL" created.
```

2. **Verify the User Principal:**
    - To confirm the principal is created, list all principals:

bash

sudo kadmin.local -q "listprincs"

```
ubuntu@ubuntu:~$ sudo kadmin.local -q "listprincs"
Authenticating as principal root/admin@pedrolab.local with password.
K/M@pedrolab.local
kadmin/admin@pedrolab.local
kadmin/changepw@pedrolab.local
krbtgt/pedrolab.local@pedrolab.local
testuser@PEDROLAB.LOCAL
```

Step 4: Configure the Kerberos Client (VM2)

1. **Edit the Kerberos Configuration File on VM2:**
    - Open /etc/krb5.conf for editing on VM2:

bash

sudo nano /etc/krb5.conf

    - Set the default realm to MYLAB.LOCAL and point to the KDC and admin server on VM1. The configuration should match what you set on VM1.

```
 GNU nano 7.2                    /etc/krb5.conf
[libdefaults]
        default_realm = pedrolab.local

# The following krb5.conf variables are only for MIT Kerberos.
        kdc_timesync = 1
        ccache_type = 4
        forwardable = true
        proxiable = true
        rdns = false


# The following libdefaults parameters are only for Heimdal Kerberos.
        fcc-mit-ticketflags = true

[realms]
        pedrolab.local = {
                kdc = pedro
                admin_server = pedro
        }
        ATHENA.MIT.EDU = {
```

Step 5: Test Kerberos Authentication

1. **Request a Kerberos Ticket for the User on VM2:**
    - In the terminal on VM2, request a ticket for testuser:

bash

kinit testuser@MYLAB.LOCAL

```
ubuntu@ubuntu:~$ kinit testuser@PEDROLAB.LOCAl
kinit: Cannot find KDC for realm "PEDROLAB.LOCAl" while getting initial creden
tials
```

    - Enter the password you set for testuser.
2. **Verify the Ticket:**
    - Check if the ticket was issued by listing active Kerberos tickets:

bash

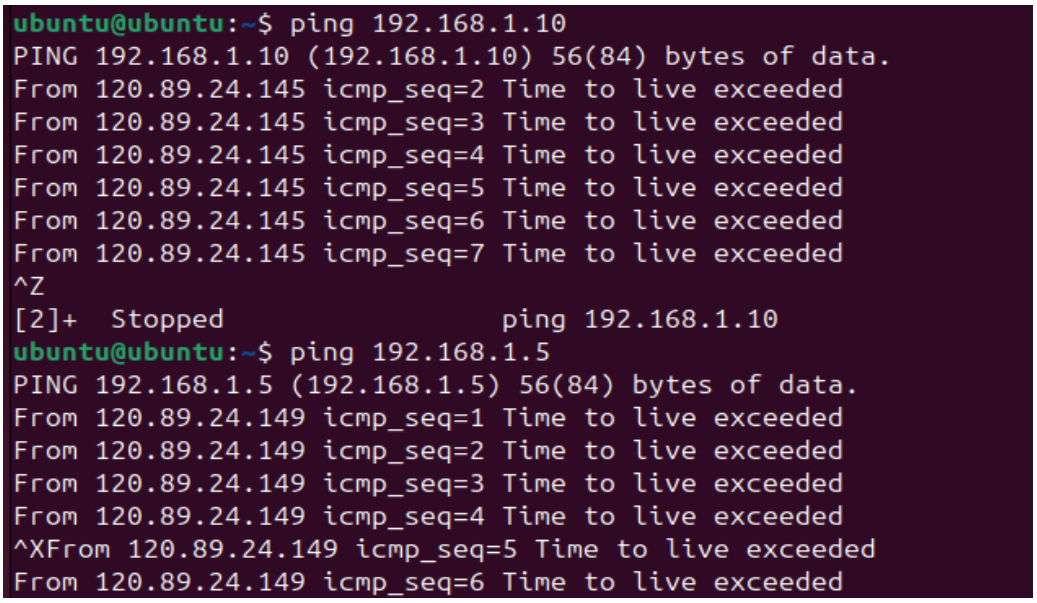
list

```
ubuntu@ubuntu:~$ list
Command 'list' not found, but there are 22 similar ones.
```

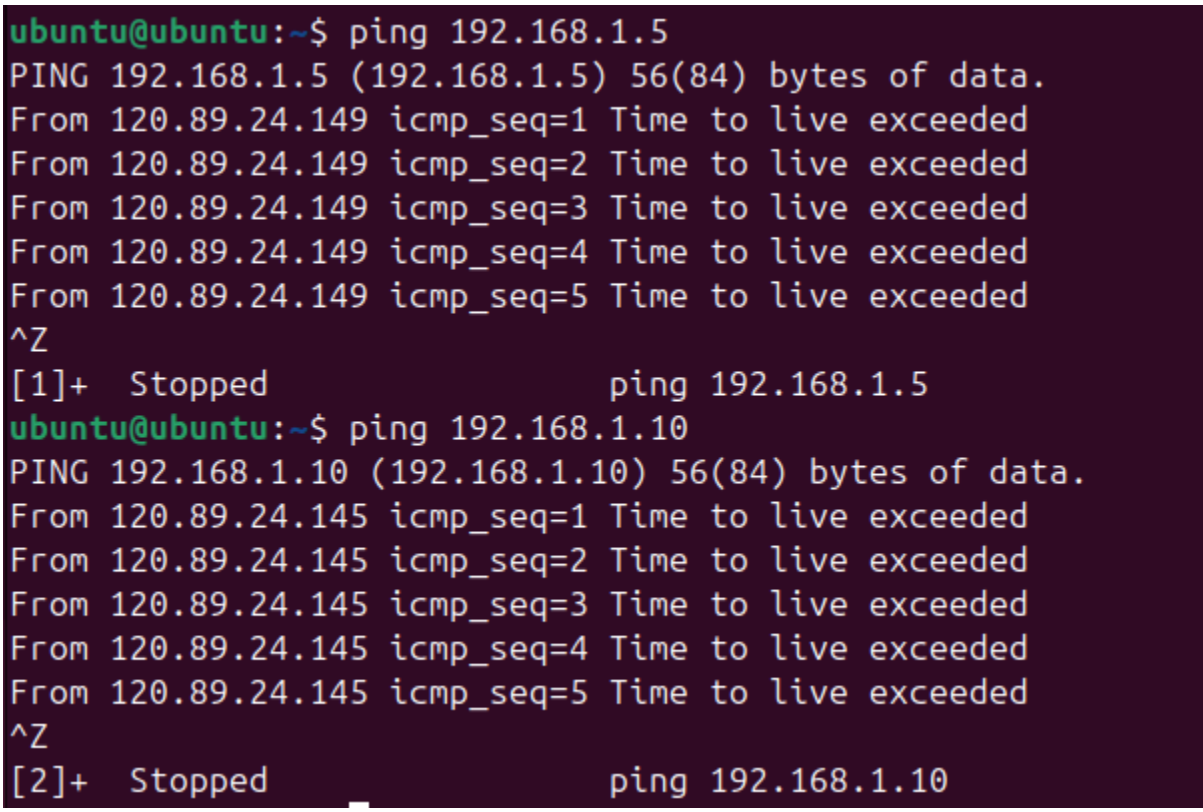
    - You should see details about the ticket, such as the principal and expiration time, confirming successful Kerberos authentication.

TICKET NOT VERIFED BUT VM1 AND VM2 BOTH CAN PING

CLIENT TERMINAL:



SERVER TERMINAL:


ubuntu@ubuntu:~$ ping 192.168.1.5
PING 192.168.1.5 (192.168.1.5) 56(84) bytes of data.
From 120.89.24.149 icmp_seq=1 Time to live exceeded
From 120.89.24.149 icmp_seq=2 Time to live exceeded
From 120.89.24.149 icmp_seq=3 Time to live exceeded
From 120.89.24.149 icmp_seq=4 Time to live exceeded
From 120.89.24.149 icmp_seq=5 Time to live exceeded
^Z
[1]+ Stopped ping 192.168.1.5
ubuntu@ubuntu:~$ ping 192.168.1.10
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
From 120.89.24.145 icmp_seq=1 Time to live exceeded
From 120.89.24.145 icmp_seq=2 Time to live exceeded
From 120.89.24.145 icmp_seq=3 Time to live exceeded
From 120.89.24.145 icmp_seq=4 Time to live exceeded
From 120.89.24.145 icmp_seq=5 Time to live exceeded
^Z
[2]+ Stopped ping 192.168.1.10