**Assignment 2**

**Scenario: Setting Up a Mail Server on Red Hat Enterprise Linux 7**

**Background:** I am system administrator for a medium-sized company that needs to set up an internal mail server to handle company emails efficiently. The company prefers Red Hat Enterprise Linux due to its robust security features, stability, and enterprise-level support. Your goal is to implement a secure and reliable mail server solution using RHEL.

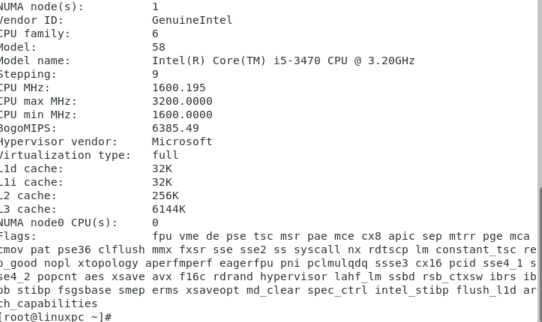
**Steps Involved:**

**Server Setup:**

Provision a new server with Red Hat Enterprise Linux installed. Ensure the server meets the hardware requirements for handling mail services, including CPU, RAM, and disk

1 .Provision a new server with Red Hat Enterprise Linux installed .

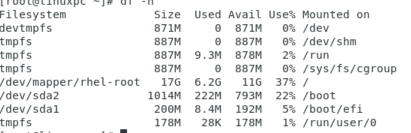
iscpu : display information about the CPU



free –h : check available and used RAM



df –h : see the disk usage on your system



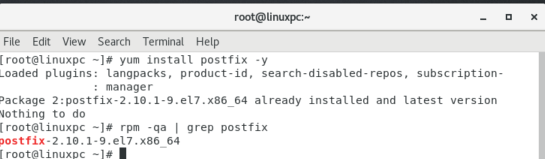
1. **Choose Mail Server Software:**

Select Postfix as the mail transfer agent (MTA), which is widely used, secure, and highly configurable.

Optionally, choose Dovecot as the mail delivery agent (MDA) and Internet Message Access Protocol (IMAP)/Post Office Protocol (POP3) server for users to retrieve their emails.

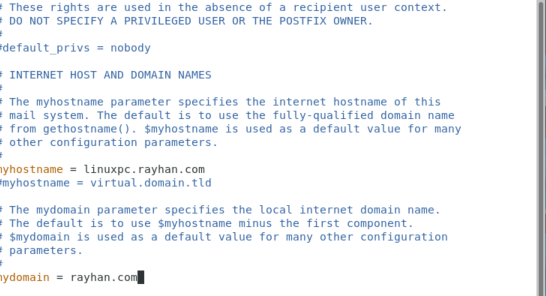
1. **Installation of Mail Server Components:**

Install Postfix and Dovecot packages on the RHEL server using package management tools (yum or dnf



**4.Configuring Postfix:** Edit the Postfix configuration files (main.cf, master.cf, etc.) located typically in /etc/postfix/ to define mail server settings, such as domain name, relay options, and security features.

**Vim /etc/postfix/main.cf**: That command opens the main.cf file, which is the primary configuration file for Postfix.

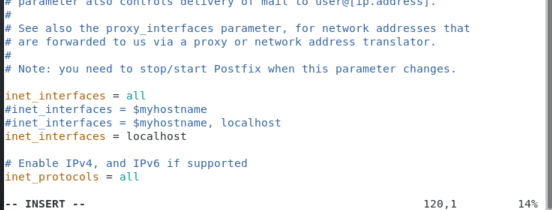


Rayhan.com



Inet\_interface =all

Inet\_protocols= all



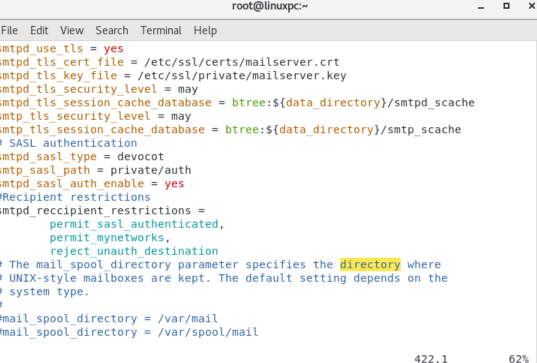
Accepts mail for local delivery to linuxpc.rayhan.com

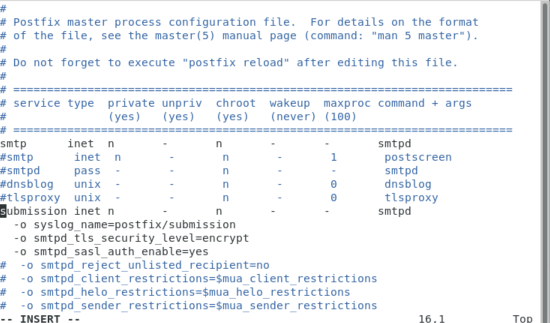


Smtp\_tls key\_file : /etc/ssl/certs/mailserver.crt

Smtp\_tls key\_file : /etc/ssl/certs/mailserver.key

Smtp\_tls\_security level=may



**Vim /etc/postfix/master.cf**: opens the master.cf file, which controls **how Postfix services are run** — basically, it’s the service manager for different components like SMTP, pickup, cleanup, etc.

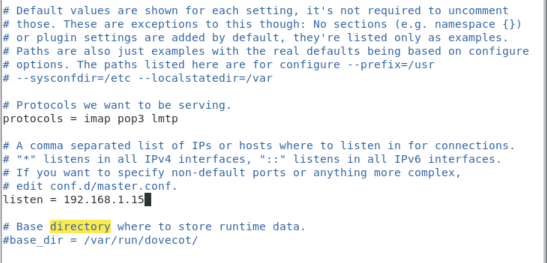
**Postfix restart** : used to **restart the Postfix mail server** on a Linux system that uses systemd



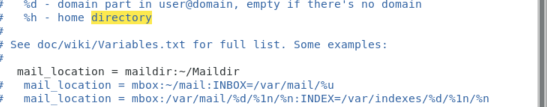
**5.Configuring Dovecot (if used):**

Edit the Dovecot configuration files (dovecot.conf, 10-mail.conf, 10-auth.conf, etc.) located in /etc/dovecot/ to set up IMAP/POP3 services, authentication mechanisms, and SSL/TLS encryption

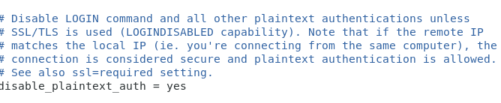
**Vim /etc/dovecot/dovecot.conf::** This is the **main configuration file for Dovecot**, which handles **IMAP** and **POP3** for your mail server. It works alongside Postfix, which handles sending mail — Dovecot handles receiving and storing mail.



**vim /etc/dovecot/conf.d/10-mail.conf:** - specifies mail location



**vim /etc/dovecot/conf.d/10-auth.conf**: This file is all about **authentication settings** — how Dovecot verifies users when they log in to fetch their mail





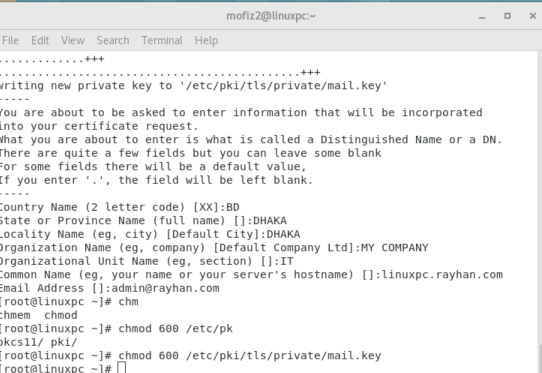
**vim /etc/dovecot/conf.d/10-ssl.conf**: **Dovecot handles SSL/TLS encryption** — super important for secure IMAP/POP3 connections.



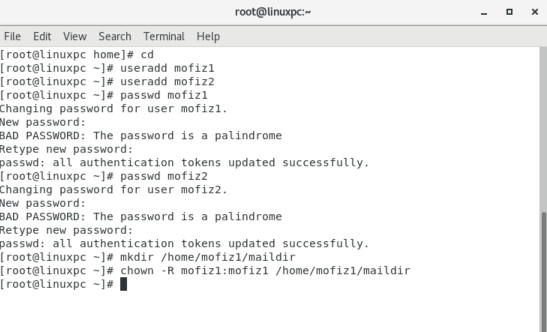
**vim /etc/dovecot/conf.d/10-master.conf**: This file configures the **master process and listener sockets** — especially important for integrating **Postfix and Dovecot**, enabling authentication services over a UNIX socket, and defining how services like IMAP and POP3 run.



Generating a self-signed SSL certificate With this command:

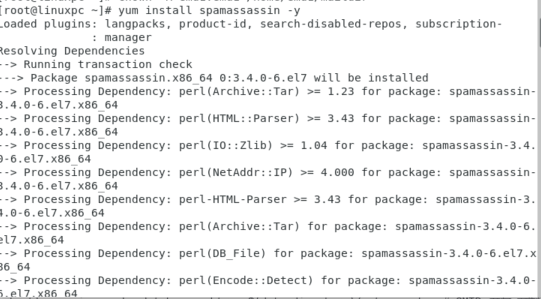


**6.User Authentication:** Integrate with a user authentication system such as LDAP (Lightweight Directory Access Protocol) or configure local system accounts for mail users. Set up authentication mechanisms (e.g., SASL) in Postfix to authenticate users sending outbound emails.



**7.SPAM Filtering and Security:**

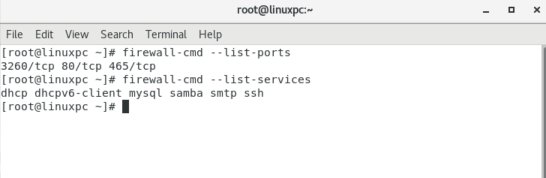
Implement SPAM filtering using tools like SpamAssassin or integrate with third-party SPAM filtering services.



Configure firewall rules (firewalld) to allow SMTP (port 25) and secure SMTPS (port 465) traffic to the mail server while blocking unnecessary ports.

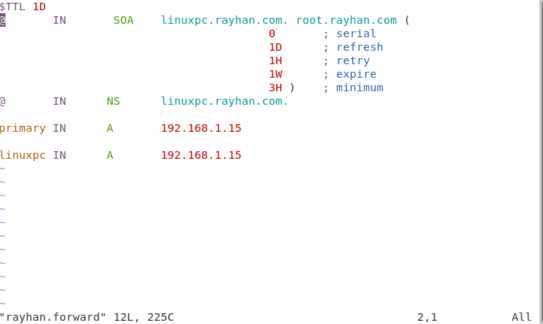
Add port: firewall-cmd --zone=public --add-port=25/tcp –permanent

Add port firewall-cmd --zone=public --add-port=465/tcp –permanent

Reload : firewall-cmd –reload

Show the service list and port list

**8.DNS Configuration:**

Ensure proper DNS configuration with MX (Mail Exchange) records pointing to your mail server's IP address for inbound mail delivery.

Configure reverse DNS (PTR) records to validate the server's identity and improve email deliverability.

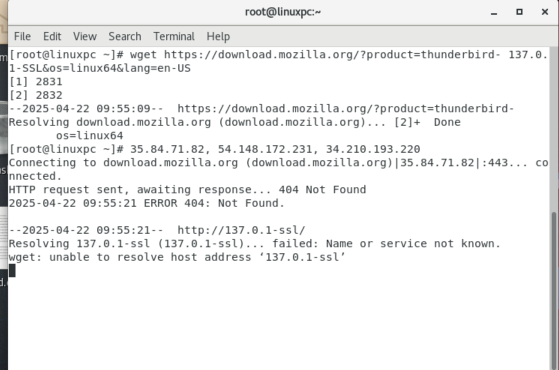


* .Configure reverse DNS (PTR) records to validate the server's identity and improve email deliverability.

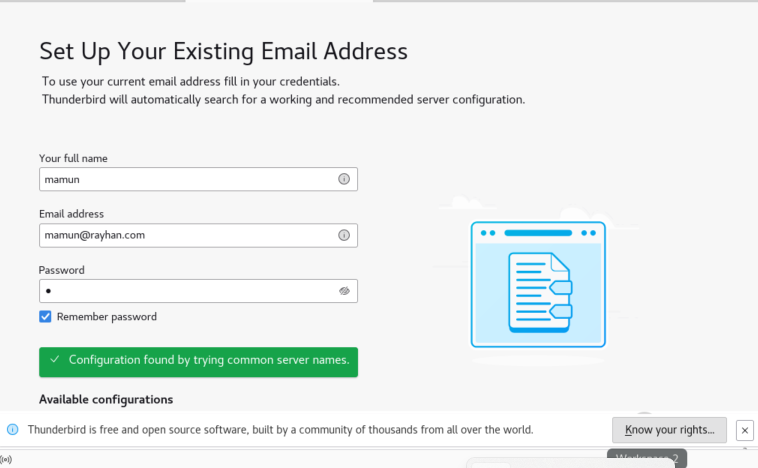
**9.Testing and validation**:

install thunderbird :

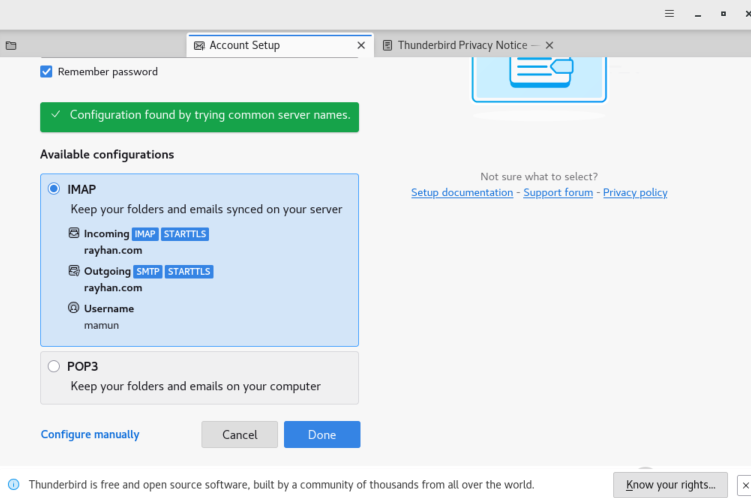
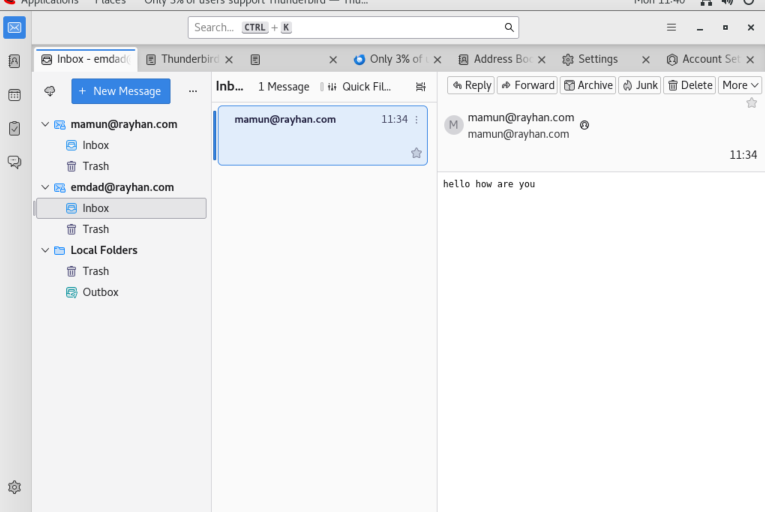
• Install thunderbird from web with wget command Wget https://download.mozilla.org/?product=thunderbird-137.0.1-SSL&os=linux64&lang=en-US



"

After finding configuration, configure Thunderbird client to see interface. "I can add another mail client

view both in the same interface, and send/receive emails

cheak the mail all are correct:

**10**. **monitoring and logging** :

**inastall rsyslog :** yum install ryslog –y



Edit the **rsyslog** config file: vi /etc/rsyslog.conf

### mail.\* -/var/log/maillog

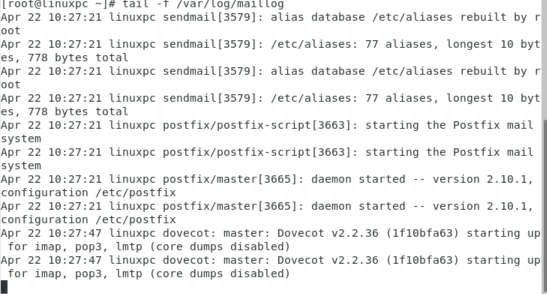
* mail.\*: This means **all log messages from the "mail" facility**, regardless of severity (info, warning, err, etc.).
* -/var/log/maillog: Logs are written to /var/log/maillog. The dash (-) before the file path is an optimization that tells rsyslog **not to sync the file to disk after every write**, improving performance.

### mail.err -/var/log/mail.err

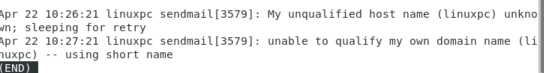
* mail.err: This logs only **error messages from the "mail" facility**.
* -/var/log/mail.err: Logs are written to /var/log/mail.err, again with the dash for performance.



tail –f /var/log/maillig: Use tail to view logs in real-time



**less /var/log/mail.err**:



**10.**

**Back up disaster recovery** :

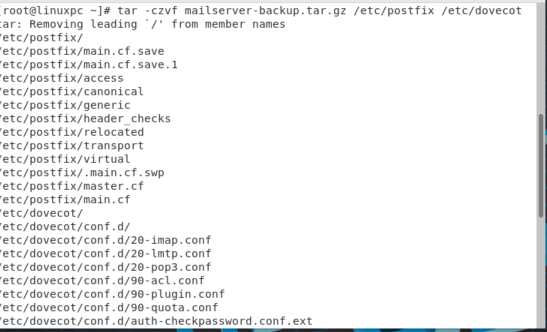
**Tar –czvf mailserver –backup.tar.gz /etc/postfix /etc/dovecot :**  tar: This is the command-line tool used to **create archives** (like zip files).

 -c: Create a new archive.

 -z: Compress the archive using **gzip** (makes the file smaller).

 -v: **Verbose** – shows progress in the terminal while running.

 -f mailserver-backup.tar.gz: This tells tar to **name the archive** as mailserver-backup.tar.gz



**Tar –tzvf mailserver-backup.tar.gz:**

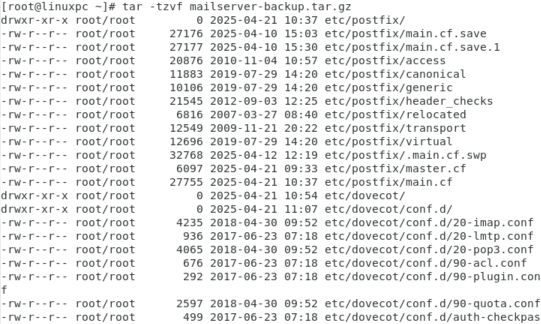
 tar: The command-line tool used to manage archive files (like .tar or .tar.gz files).

 -t: **List the contents** of the archive. It doesn't extract, just shows what's inside.

 -z: Tells tar to use **gzip decompression** because the file is .gz compressed.

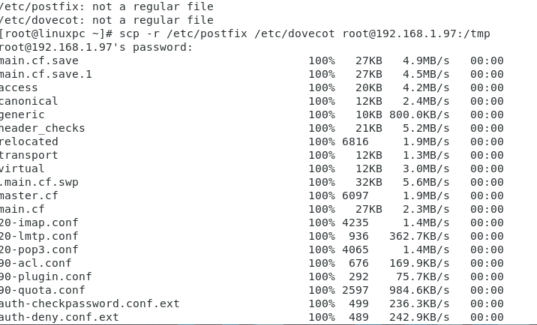
 -v: **Verbose mode** – shows detailed information about each file in the archive.

 -f mailserver-backup.tar.gz: Specifies the **name of the archive file** you want to look into.



this file disaster or fail backup other server :

**scp –r /etc/postfix /etc/dovecot root@192.168.1.97:/tmp :**



Ssh @192.168.1.97 : SSH (Secure Shell) is a protocol used to securely connect to a remote computer or server over a network



**conclusion:**

deploying a mail server using Postfix and Dovecot on RHEL involves careful planning, secure configuration, thorough testing, and ongoing monitoring to ensure reliable, secure, and efficient email communication within the organization. Regular maintenance and backups are essential.