Linux - Setup Rocky 9 SMTP Server

> **Security warning:** this guide is written for a lab or internal relay. It disables SELinux and the firewall, allows plaintext authentication (`disable_plaintext_auth = no`, `auth_mechanisms = plain login`) and sets the Dovecot auth socket to `mode = 0666` (any local user can use it). For anything internet-facing: keep SELinux enforcing, keep firewalld running and open only the mail ports you need, require TLS before authentication (for example Postfix `smtpd_tls_auth_only = yes`), and restrict the auth socket to Postfix (`mode = 0660`, `user = postfix`, `group = postfix`, as in the Postfix SASL README).

System Configuration

Upgrade Current System

```
dnf install epel-release -y
dnf upgrade -y
```

Configure SELinux

```
setenforce 0
sed -i 's/^SELINUX=.*/SELINUX=disabled/g' /etc/selinux/config
```

Disable Firewall

```
systemctl disable firewalld.service
```

Install Core Tools

```
dnf install bind-utils bzip2 cups cifs-utils enscript ftp gdb ghostscript java-1.8.0-openjdk-headless java-11-openjdk-headless krb5-workstation ksh lftp lrzsz lsof libnsl lzop mariadb-server mlocate mutt ncompress net-tools net-snmp net-snmp-utils net-tools nfs-utils nmap nvme-cli openldap-clients openssh-clients psmisc realmd rsync samba-client strace sysstat tcpdump telnet telnet-server tmux unix2dos vim vim-enhanced vsftpd wget xfsdump vsftpd htop mc rsyslog rsyslog-doc postfix dbus-daemon s-nail dovecot cyrus-sasl cyrus-sasl-lib cyrus-sasl-plain -y
```

Configure Virtual Tool

```
dnf install open-vm-tools -y
sysctl vm.swappiness=10
```

Time Sync

```
systemctl enable --now chronyd
```

Configure Postfix

Postfix Settings

We now have to configure Postfix. One thing to keep in mind is that we're configuring Postfix to only send email, not receive it (as that is a far more complicated topic that requires considerable setup time and understanding to prevent the server from becoming an open relay, which could lead to a serious spam issue). Because of this, we can skip setting up Postfix to listen and instead go right to the hostname.

The Postfix hostname must be set to match the system hostname. We'll use the mail.example.com address (so make sure to change this to match your hostname). Set that hostname with the command:

```
sudo postconf -e "myhostname = mail.yourdomain.com"
```

Make sure to check that the apex domain (aka root domain) is correct with the command:

```
postconf mydomain
```

The apex domain for our example should be listed as http://example.com . If not, set it with:

```
sudo postconf -e "mydomain = example.com"
```

Set the myorigin parameter with:

```
sudo postconf -e "myorigin = $mydomain"
```

Set to allow all IP to access the server with:

```
sudo postconf -e "inet_interfaces = all"
```

Set to only allow IPv4 to use this server with:

```
sudo postconf -e "inet_protocols = ipv4"
```

Set the mydestination parameter with:

```
sudo postconf -e "mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain"
```

Set the allowed IP address to relay on this server with:

```
sudo postconf -e "mynetworks = 127.0.0.0/8, 10.0.0.0/24, 192.168.0.0/16"
```

Set the mail folder with:

```
sudo postconf -e "home_mailbox = Maildir/"
```

Set the banner with:

```
sudo postconf -e "smtpd_banner = $myhostname ESMTP"
```

Set to disable verify with:

```
sudo postconf -e "disable_vrfy_command = yes"
```

Set to require the HELO for senders with:

```
sudo postconf -e "smtpd_helo_required = yes"
```

Set the message limit for example 10MB with:

```
sudo postconf -e "message_size_limit = 10240000"
```

Set SMTP Authentication with:

```
sudo postconf -e "smtpd_sasl_type = dovecot"
sudo postconf -e "smtpd_sasl_path = private/auth"
sudo postconf -e "smtpd_sasl_auth_enable = yes"
sudo postconf -e "smtpd_sasl_security_options = noanonymous"
sudo postconf -e "smtpd_sasl_local_domain = $myhostname"
sudo postconf -e "smtpd_recipient_restrictions = permit_mynetworks, permit_auth_destination,
permit_sasl_authenticated, reject"
```

With these taken care of, restart Postfix with:

```
sudo systemctl restart postfix
```

Extra Authentications

Configure additional settings for Postfix if you need.
It's possible to reject many spam emails with the settings below.

However, you should consider to apply the settings, because sometimes normal emails are also rejected with them. Especially, there are SMTP servers that forward lookup and reverse lookup of their hostnames on DNS do not match even if they are not spammers.

```
sudo postconf -e "smtpd_client_restrictions = permit_mynetworks, reject_unknown_client_hostname, permit"
sudo postconf -e "smtpd_sender_restrictions = permit_mynetworks,
reject_unknown_sender_domain,reject_non_fqdn_sender"
sudo postconf -e "smtpd_helo_restrictions = permit_mynetworks,
reject_unknown_hostname,reject_non_fqdn_hostname, reject_invalid_hostname, permit"
```

Enable Postfix

```
sudo systemctl enable --now postfix
```

Configure Dovecot

Dovecot Settings

This example shows to configure to provide SASL function to Postfix.

vi /etc/dovecot/dovecot.conf and uncomment and if not use IPv6, remove [::]

```
listen = *, ::
```

vi /etc/dovecot/conf.d/10-auth.conf and uncomment and change for the case you allow plain text auth

```
disable_plaintext_auth = no
```

and then add login to

```
auth_mechanisms = plain login
```

vi /etc/dovecot/conf.d/10-mail.conf and uncomment and add

```
mail_location = maildir:~/Maildir
```

vi /etc/dovecot/conf.d/10-master.conf and uncomment and add like follows Postfix smtp-auth

```
  unix_listener /var/spool/postfix/private/auth {
    mode = 0666
    user = postfix
    group = postfix
  }
```

vi /etc/dovecot/conf.d/10-ssl.conf and change to use SSL if available but not require SSL

```
ssl = yes
```

Enable Dovecot

```
sudo systemctl enable --now dovecot
```

Test the setup

Now that everything is set up, test Postfix by sending an email from the command line like so:

```
echo "Rocky Linux Rocks" | sendmail EMAIL
```

Where EMAIL is a valid email address.

If you receive the email, congratulate yourself on a job well done. If the email fails to arrive, you might need to verify if your DNS records are correct and the changes have taken effect (they can take up to 24 hours). You can also check the maillog with a command like:

```
tail -f /var/log/maillog
```

With the tail running, open another terminal window and attempt to send another email to see what kind of logs are written. From that information, you can start troubleshooting any issues that are causing problems.

Used ref from

https://www.server-world.info/en/note?os=Rocky_Linux_8&p=mail&f=1

https://www.server-world.info/en/note?os=Rocky_Linux_8&p=mail&f=2

Revision #1
Created 2 September 2024 22:54:26 by Admin
Updated 24 October 2026 14:10:46 by Admin