Exercise 9.4: Enable relaying using SMTP Auth in postfix

- Ensure the mynetworks_style is set to host:

  ```
  # postconf -e "mynetworks_style = host"
  ```

  To avoid issues with incorrectly set up DNS, or enforced **ssl**, use this setting for your lab as well:

  ```
  # postconf -e "disable_dns_lookups = yes"
  # postconf -e "smtpd_tls_auth_only = no"
  ```

  NOTE: Don't enable these settings in production. Use them only for this lab.

  NOTE: We will re-enforce **SSL** authentication in the next exercise.

- Restart **Postfix** with above setting before starting the lab:

  ```
  # systemctl restart postfix
  ```

Solution 9.4

1. Enable the **SASL** authentication service in **Dovecot**.

   - Edit the file /etc/dovecot/conf.d/10-master.conf and after the section service auth add or un-comment the following lines:

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

2. Restart **Dovecot**:

   ```
   # systemctl restart dovecot
   ```

3. Enable sasl authentication in **Postfix**.

   Make the following setting changes:

   ```
   # postconf -e "smtpd_sasl_type = dovecot"
   # postconf -e "smtpd_sasl_auth_enable = yes"
   # postconf -e "smtpd_recipient_restrictions = \
               permit_mynetworks,               \
               permit_sasl_authenticated,       \
               reject_unauth_destination"
   ```

4. Configure the proper authentication path:

   ```
   # postconf -e "smtpd_sasl_path = private/auth"
   ```

5. Restart **Postfix**:

   ```
   # systemctl restart postfix
   ```

6. Test plain text authentication from a **remote host**.

   Notice: that any system listed in **permit_mynetworks** will be allowed to relay. The current settings of **permit_mynetworks** in conjunction with **mynetworks_style** will allow the local system to relay without authentication.

   If you wish to test authentication on a single machine eliminate the **permit_mynetworks** entry from
   smtpd_recipient_restrictions to force all systems attempting to relay to authenticate.

   ```
   $ telnet <SERVER> 25
   helo localhost
   mail from:student
   rcpt to:root@<OTHER MACHINE>
   quit
   ```

   This should fail with relay access denied. Test again with authentication:
   Create the base64 encoded user and password.

   ```
   $ echo -en "\0student\0student" | base64
   ```

   Using the encrypted user and password, send the email.

   ```
   $ telnet <SERVER> 25
   helo localhost
   auth plain AHN0dWRlbnQAc3R1ZGVudA==
   mail from:student
   rcpt to:root@<OTHER MACHINE>
   data
   Subject: I sent this using SASL SMTP auth

   Cool no?
   .
   quit
   ```