Odne Rindheim

Task 1: SELinux

1. Check whether SELinux is currently enabled. Make sure SELinux is enabled and in targeted and enforcing mode.
    a. To check whether SELinux is enabled and its current mode, I ran getenforce
    And it returned ‘Enforcing’, which tells me its enabled and in enforcing mode.
2. Working with SELinux users.
    a. Check the ma page of semanage and list the mapping between Linux users and SELinux confined users on your computer.
        i. To list the mapping between linux users and SELinux, I run:
        semanage login -l (as sudo)
        and it returned: __default__ and root, both unconfined_u
    b. Check the man page of seinfo to list all available SELinux users on your computer.
        i. To list all the available SELinux users, I ran:
        seinfo –user (as sudo)
        and it returned 8 users: guest_u, root, staff_u, sysadm_u, system_u, unconfined_u, user_u, xguest_u.
    c. Create a new Linux user and use SELinux to prevent this user from using the su and sudo tools.
        i. When creating a new user, I run the command:
        sudo adduser newuser
        And to restrict the user from using su and sudo tools, I assign a more restricting role, like guest_u by running command:
        sudo semanage login -a -s guest_u newuser
3. Apache Web Server Access. (Installing Apache web server (httpd))
sudo yum install httpd -y

Odne Rindheim

a. Is systemd allowed to start the Apache web server?

   I ran the commands:

   sudo systemctl start httpd,

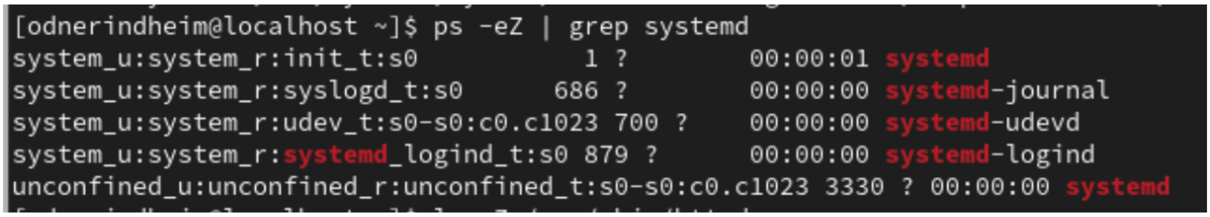
   sudo systemctl enable httpd,

   And it returned it created a symlink to httpd.service

   i. Determine the SELinux type of systemd.

      1. I ran the command:

         ps -eZ | grep systemd,

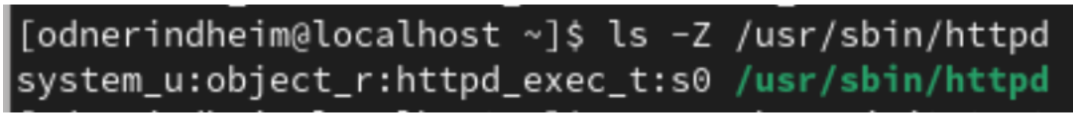


```
[odnerindheim@localhost ~]$ ps -eZ | grep systemd
system_u:system_r:init_t:s0           1 ?        00:00:01 systemd
system_u:system_r:syslogd_t:s0      686 ?        00:00:00 systemd-journal
system_u:system_r:udev_t:s0-s0:c0.c1023 700 ?    00:00:00 systemd-udevd
system_u:system_r:systemd_logind_t:s0 879 ?      00:00:00 systemd-logind
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 3330 ? 00:00:00 systemd
```

   ii. Determine the SELinux type of the Apache executable file.

      1. I ran the command:

         ls -Z /usr/sbin/httpd

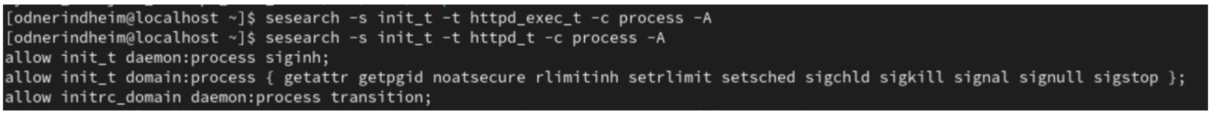


```
[odnerindheim@localhost ~]$ ls -Z /usr/sbin/httpd
system_u:object_r:httpd_exec_t:s0 /usr/sbin/httpd
```

   iii. Determine if system is allowed to run the Apache executable.

      1. I ran the command:

         seserach -s init_t -t httpd_exec_t -c process -A



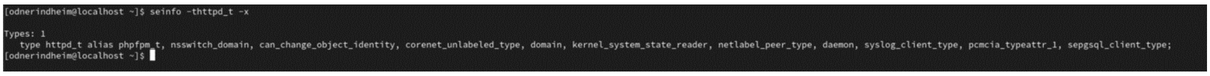
```
[odnerindheim@localhost ~]$ sesearch -s init_t -t httpd_exec_t -c process -A
[odnerindheim@localhost ~]$ sesearch -s init_t -t httpd_t -c process -A
allow init_t daemon:process siginh;
allow init_t domain:process { getattr getpgid noatsecure rlimitinh setrlimit setsched sigchld sigkill signal signull sigstop };
allow initrc_domain daemon:process transition;
```

b. Can the Apache web server run in domain httpd_t?

   i. Ran the command:

      seinfo -thttpd_t -x

      And returned:



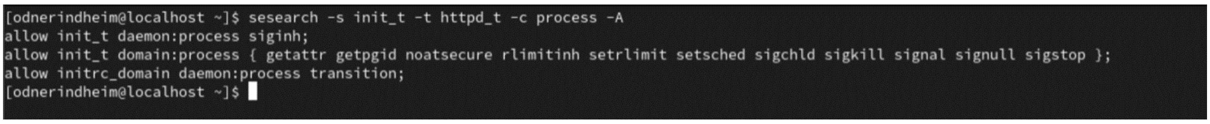
```
[odnerindheim@localhost ~]$ seinfo -thttpd_t -x

Types: 1
   type httpd_t alias phpfpm_t, nsswitch_domain, can_change_object_identity, corenet_unlabeled_type, domain, kernel_system_state_reader, netlabel_peer_type, daemon, syslog_client_type, pcmcia_typeattr_1, sepgsql_client_type;
[odnerindheim@localhost ~]$ 
```

c. Is system allowed a transition to httpd_t

i. I ran the command:

sesearch -s init_t -t httpd_t -c process -A

And returned:



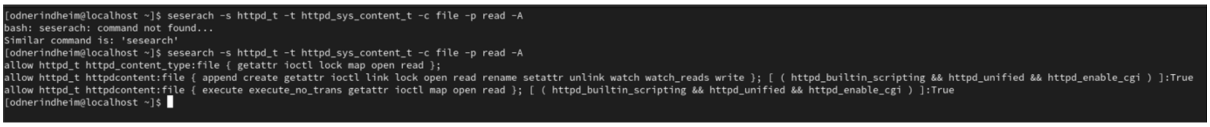
```
[odnerindheim@localhost ~]$ sesearch -s init_t -t httpd_t -c process -A
allow init_t daemon:process siginh;
allow init_t domain:process { getattr getpgid noatsecure rlimitinh setrlimit setsched sigchld sigkill signal signull sigstop };
allow initrc_domain daemon:process transition;
[odnerindheim@localhost ~]$
```

d. Has domain httpd_t access to open and read files in directory /var/www/html?

i. I ran the command:

sesearch -s httpd_t -t httpd_sys_content_t -c file -p read -A

And returned:



```
[odnerindheim@localhost ~]$ seserach -s httpd_t -t httpd_sys_content_t -c file -p read -A
bash: seserach: command not found...
Similar command is: 'sesearch'
[odnerindheim@localhost ~]$ sesearch -s httpd_t -t httpd_sys_content_t -c file -p read -A
allow httpd_t httpd_content_type:file { getattr ioctl lock map open read };
allow httpd_t httpdcontent:file { append create getattr ioctl link lock open read rename setattr unlink watch watch_reads write }; [ ( httpd_builtin_scripting && httpd_unified && httpd_enable_cgi ) ]:True
allow httpd_t httpdcontent:file { execute execute_no_trans getattr ioctl map open read }; [ ( httpd_builtin_scripting && httpd_unified && httpd_enable_cgi ) ]:True
[odnerindheim@localhost ~]$
```

4. Use a SELinux Boolean to allow Apache to read web content in a public_html directory in the home directory of users.

   a. I installed selinux-policy-doc with command:

      sudo yum install selinux-policy-doc -y

      and checked the man httpd_selinux:

      man httpd_selinx.

      Then I used command:

      sudo setsebool -P httpd_enable_homedirs 1,

      to allow apache to read web content in a public_html directory in the home directory of users.

5. Create a directory /www, and configure SELinux to allow Apache to read web content in this directory.

   a. I ran the commands:

      sudo mkdir /www

      sudo semaange fcontext -a -t httpd_sys_content_t "/www(/.*)?"

      sudo restorecon -Rv /www

To create the directory /www, and set the proper context to allow apache to read web content in this directory.

Task 2: Printing

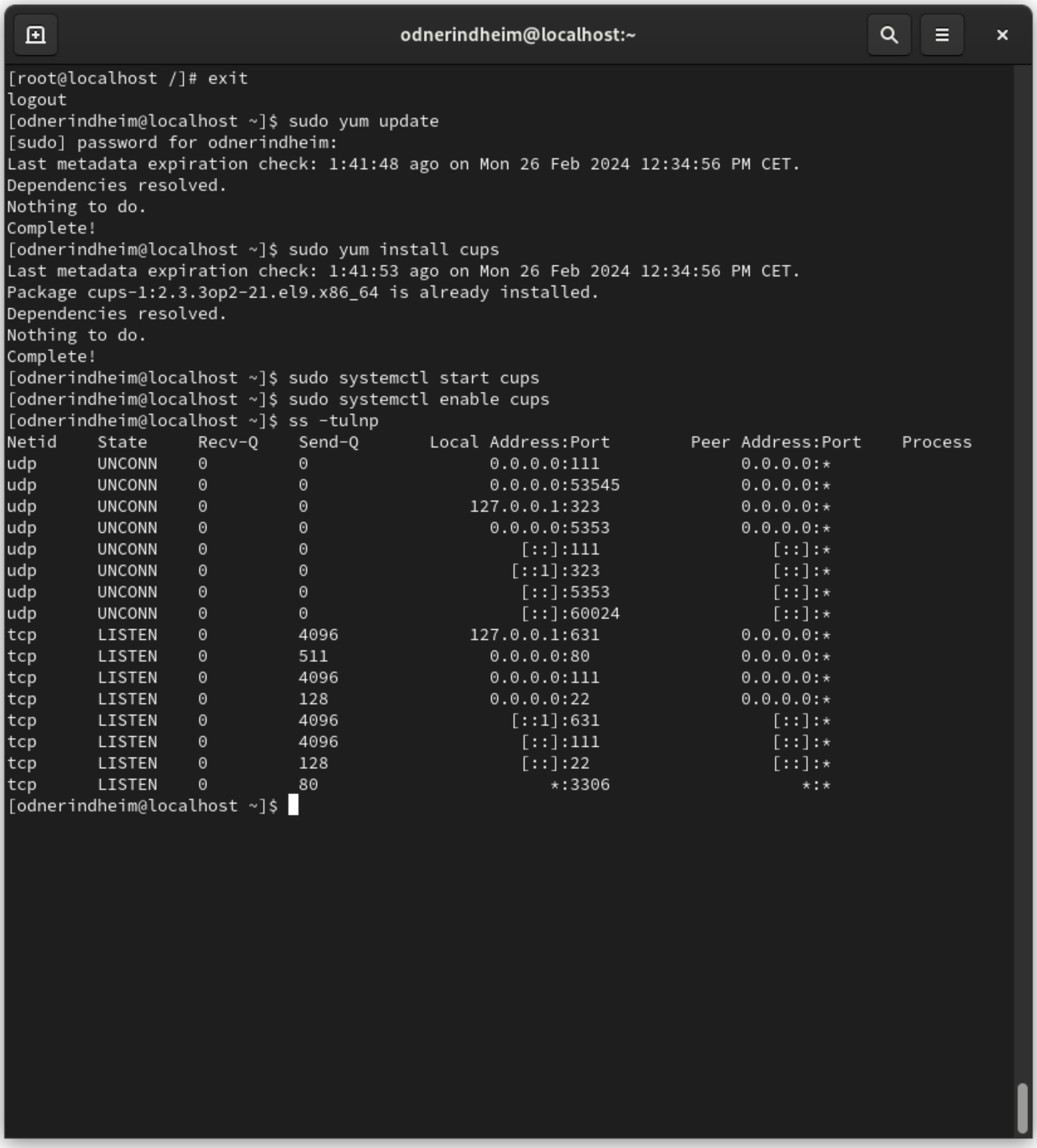
Installing CUPS and other Dependencies.

Installed CUPS and set to start on startup with systemctl enable.

1. Send a print job from LibreOffice
    a. Followed the installation guide from MyPrint to install, sent a print job from TextEditor instead of LibreOffice as I didn't have it installed at the time.
2. Send a print job from the command line using the lpr command to the HVL printer system.
    a. Sent a print job from the command line using lpr through the GUI, which prompted me with a authentication, where I entered my authentication. I also did it through the command line, using lpr -P MyPrint ~/Downloads/myPrintInstaller_LINUX/How_to_install.txt, got a GUI pop up for print started, clicked it and got into the printer settings tab, where I entered my authentication and printing was completed.

Task 3: Open port and processes

1. IG
    a. Used the command:
       ss -tulnp
       And output was:



```
[root@localhost /]# exit
logout
[odnerindheim@localhost ~]$ sudo yum update
[sudo] password for odnerindheim:
Last metadata expiration check: 1:41:48 ago on Mon 26 Feb 2024 12:34:56 PM CET.
Dependencies resolved.
Nothing to do.
Complete!
[odnerindheim@localhost ~]$ sudo yum install cups
Last metadata expiration check: 1:41:53 ago on Mon 26 Feb 2024 12:34:56 PM CET.
Package cups-1:2.3.3op2-21.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[odnerindheim@localhost ~]$ sudo systemctl start cups
[odnerindheim@localhost ~]$ sudo systemctl enable cups
[odnerindheim@localhost ~]$ ss -tulnp
Netid    State     Recv-Q    Send-Q       Local Address:Port        Peer Address:Port    Process
udp      UNCONN    0         0                  0.0.0.0:111              0.0.0.0:*
udp      UNCONN    0         0                  0.0.0.0:53545            0.0.0.0:*
udp      UNCONN    0         0                127.0.0.1:323              0.0.0.0:*
udp      UNCONN    0         0                  0.0.0.0:5353             0.0.0.0:*
udp      UNCONN    0         0                     [::]:111                 [::]:*
udp      UNCONN    0         0                    [::1]:323                 [::]:*
udp      UNCONN    0         0                     [::]:5353                [::]:*
udp      UNCONN    0         0                     [::]:60024               [::]:*
tcp      LISTEN    0         4096             127.0.0.1:631              0.0.0.0:*
tcp      LISTEN    0         511                0.0.0.0:80               0.0.0.0:*
tcp      LISTEN    0         4096               0.0.0.0:111              0.0.0.0:*
tcp      LISTEN    0         128                0.0.0.0:22               0.0.0.0:*
tcp      LISTEN    0         4096                 [::1]:631                 [::]:*
tcp      LISTEN    0         4096                  [::]:111                 [::]:*
tcp      LISTEN    0         128                   [::]:22                  [::]:*
tcp      LISTEN    0         80                       *:3306                   *:*
[odnerindheim@localhost ~]$
```

Odne Rindheim

Task 4: 2FA.

Installed Google Authenticator PAM Module

```
sudo yum install google-authenticator.
```

Configured Google Authenticator by running:

```
google-authenticator.
```

Modified SSh connections

```
Sudo nano /etc/ssh/sshd_config
```

Where I added the following:

```
ChallengeResponseAuthenticaiton yes
AuthenticationMethods publickey,keyboard-interactive.
```

Updated the PAM SSHD Configuration file:

```
Sudo nano /etc/pam.d/sshd
```

Where i added the following line towards the top of the file:

```
Auth required pam_google_authenticator.so.
```

To avoid conflict with sshd_config and 50-redhat.conf, we copied the file and created another to avoid complications.

```
Sudo nano /etc/ssh/sshd_config.d/60-mfa.conf
```

Where I made the changes to include:

```
AuthenticatoinMethods publickey,keyboard-interactive.
```

Restating SSH service

```
Sudo systemctl restard sshd
```

Moved the google authenticator secret:

```
Mv ~/.google_authenticator ~/.ssh/
Restorecon -Rv ~/.ssh/
```

Made a ssh keys from the computer 10.0.0.70 and added them onto my computer inside the folder authorized_keys

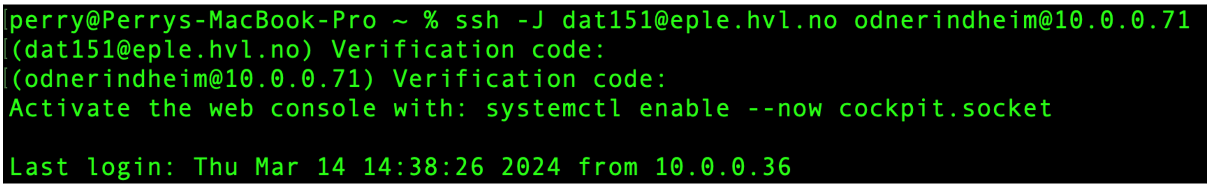
Tried ssh into my computer from the other computer with:

```
Ssh odnerindheim@10.0.0.71
```

Entered the authentication code from my Authy app and connected the successfully.

Here I show me logging in with my personal computer using SSH through eple. As I didn't have the access to the other lab computer.

Odne Rindheim



```
[perry@Perrys-MacBook-Pro ~ % ssh -J dat151@eple.hvl.no odnerindheim@10.0.0.71
[(dat151@eple.hvl.no) Verification code:
[(odnerindheim@10.0.0.71) Verification code:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Mar 14 14:38:26 2024 from 10.0.0.36
```

Odne Rindheim

Task 5: Secure your computer

Installing Fail2Ban:
Sudo yum install fail2ban

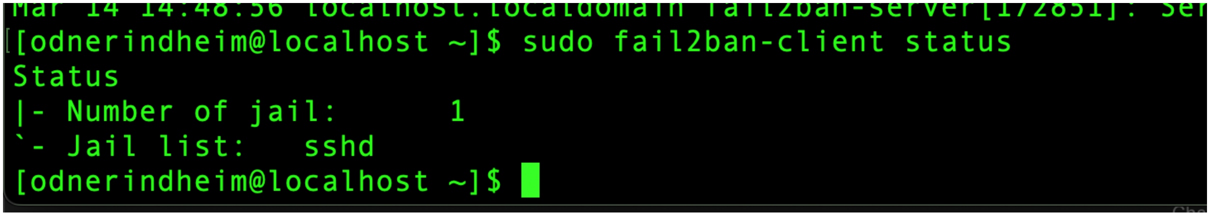
Copying the default configuration file to a new file to avoid overwriting:
Sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
Sudo nano /etc/fail2ban/jail.local

Control + W to find sshd
Enabled sshd jail by writing
[sshd]
enabled=true

Started fail2ban, and enabled to start on boot.
Sudo systemctl start fail2ban
Sudo systemctl enable fail2ban

Checked Fail2ban Status:
sudo fail2ban-client status

Output:

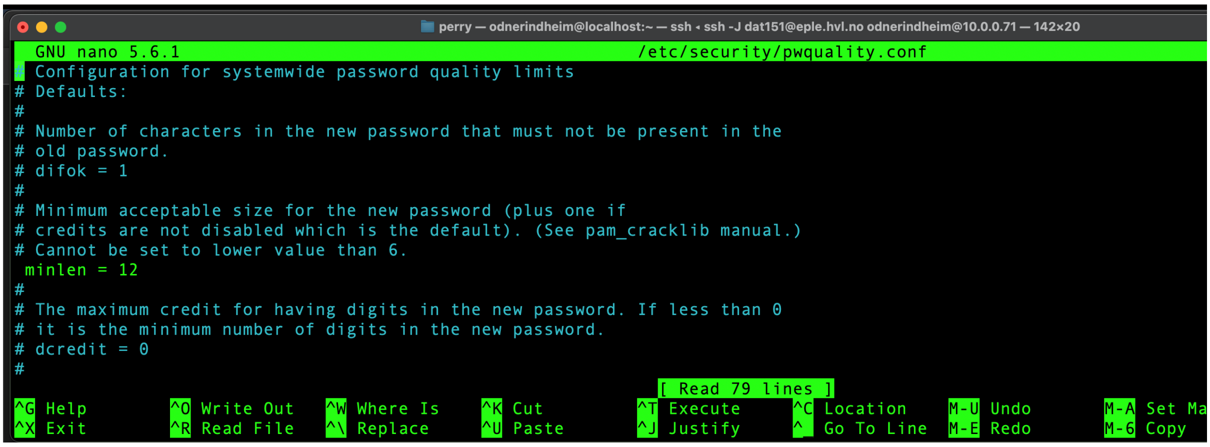


```
Mar 14 14:48:56 localhost.localdomain fail2ban-server[172851]: Ser
[odnerindheim@localhost ~]$ sudo fail2ban-client status
Status
|- Number of jail:      1
`- Jail list:   sshd
[odnerindheim@localhost ~]$ 
```

Setting PAM requirements.
Selecting current profile
Sudo authselect current

Sudo authselect enable-feature with-faillock
Sudo authselect enable-feature with-pwhistory
Sudo nano /etc/security/pwquality.conf

Odne Rindheim



perry — odnerindheim@localhost:~ — ssh ◂ ssh -J dat151@eple.hvl.no odnerindheim@10.0.0.71 — 142×20

```
  GNU nano 5.6.1                                          /etc/security/pwquality.conf
# Configuration for systemwide password quality limits
# Defaults:
#
# Number of characters in the new password that must not be present in the
# old password.
# difok = 1
#
# Minimum acceptable size for the new password (plus one if
# credits are not disabled which is the default). (See pam_cracklib manual.)
# Cannot be set to lower value than 6.
 minlen = 12
#
# The maximum credit for having digits in the new password. If less than 0
# it is the minimum number of digits in the new password.
# dcredit = 0
#
                                                [ Read 79 lines ]
^G Help          ^O Write Out     ^W Where Is      ^K Cut           ^T Execute       ^C Location      M-U Undo         M-A Set Ma
^X Exit          ^R Read File     ^\ Replace       ^U Paste         ^J Justify       ^_ Go To Line    M-E Redo         M-6 Copy
```

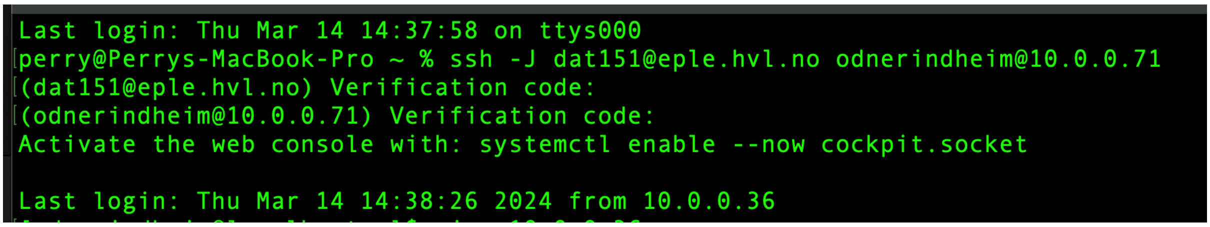
Made min length to 12 instead of 8, keeping it simple.
The file pwquality.conf controls behavior of the pam_pwquality.so module.
Applied configuration to PAM by checking that /etc/pam.d/system-auth is using pam_pwquality.so, which it was.

Saving changes made for good measures.
Sudo authselect apply-changes

Odne Rindheim

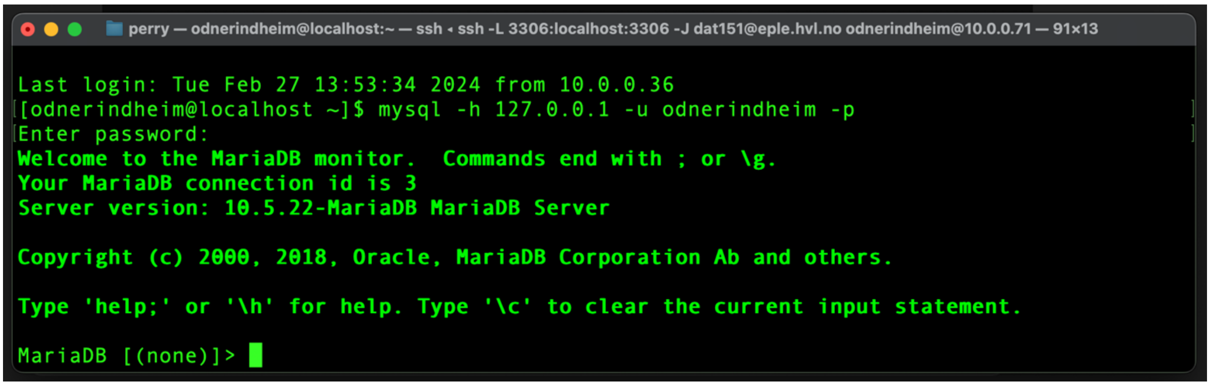
Task 6: SSH

1. Try to log in to your lab computer from your own computer. Explain the result and why this happened.
    a. Tried to log in to lab computer from own computer:
    ssh odnerindheim@10.0.0.71
    Which fails as the lab network is probably a closed/separate network than eduroam and therefore wont have access in the same way.
2. From your own computer, log into lab computer with a jump through eple.hvl.no.
    a. Jumping through eple.hvl.no
    ssh -J dat151@eple.hvl.no odnerindheim@10.0.0.71
    This worked, because eple is available from eduroam network and is also connected to the lab network, which makes so I can connect to the lab computer by going through eple!



```
Last login: Thu Mar 14 14:37:58 on ttys000
[perry@Perrys-MacBook-Pro ~ % ssh -J dat151@eple.hvl.no odnerindheim@10.0.0.71
[(dat151@eple.hvl.no) Verification code:
[(odnerindheim@10.0.0.71) Verification code:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Thu Mar 14 14:38:26 2024 from 10.0.0.36
```

3. Install MariaDB client tool on your own computer. Set up an SSH tunnel for MariaDB from your own computer through eple.hvl.no to the MariaDB server on your lab computer. Then use MariaDB client tool on your computer to access the MariaDB server on your lab computer.
    a. Installing MariaDB on my own computer:
    brew install mariadb


perry — odnerindheim@localhost:~ — ssh ◂ ssh -L 3306:localhost:3306 -J dat151@eple.hvl.no odnerindheim@10.0.0.71 — 91×13

```
Last login: Tue Feb 27 13:53:34 2024 from 10.0.0.36
[odnerindheim@localhost ~]$ mysql -h 127.0.0.1 -u odnerindheim -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 3
Server version: 10.5.22-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]>
```

Setting up SSH Tunnel:
ssh -L 3306:localhost:3306 -J dat151@eple.hvl.no odnerindheim@10.0.0.71

Accessing MariaDB Server using Client Tool:
mysql -h 127.0.0.1 -u odnerindheim -p