2019831074 – Mehrajul Islam

LAB 6 REPORT

Objective

The objective of this lab is to extend the functionalities of a previously secured web server by incorporating authentication mechanisms to control access to the server. We aim to achieve the following:

- Redirect HTTP traffic to HTTPS.
- Implement basic authentication using .htpasswd.
- Implement authentication using MySQL database.

Checkpoint 1: Redirect HTTP to HTTPS

1. Used the command sudo a2enmod rewrite to enable the mod_rewrite module.
2. Edited the configuration file for port 80 located in /etc/apache2/sites-enabled/ directory.
3. Used sudo apache2ctl configtest to test the Apache configuration
4. Restarted the server and Accessed http://example.com in the browser to confirm it redirects to https://example.com.

Observations:

- Successfully redirected HTTP traffic to HTTPS, ensuring secure communication by default.

Checkpoint 2: Basic Authentication using .htpasswd

1. Added users to the Apache server using the htpaswd
2. Added another user in same way
3. Verified the users
4. Configured the example.com and added a code snippet at example.conf file to enable htpsswd authentication
5. Accessed https://example.com and provided the credentials when prompted.

Observations:

- Successfully implemented basic authentication using .htpasswd. Only authenticated users could access the website.

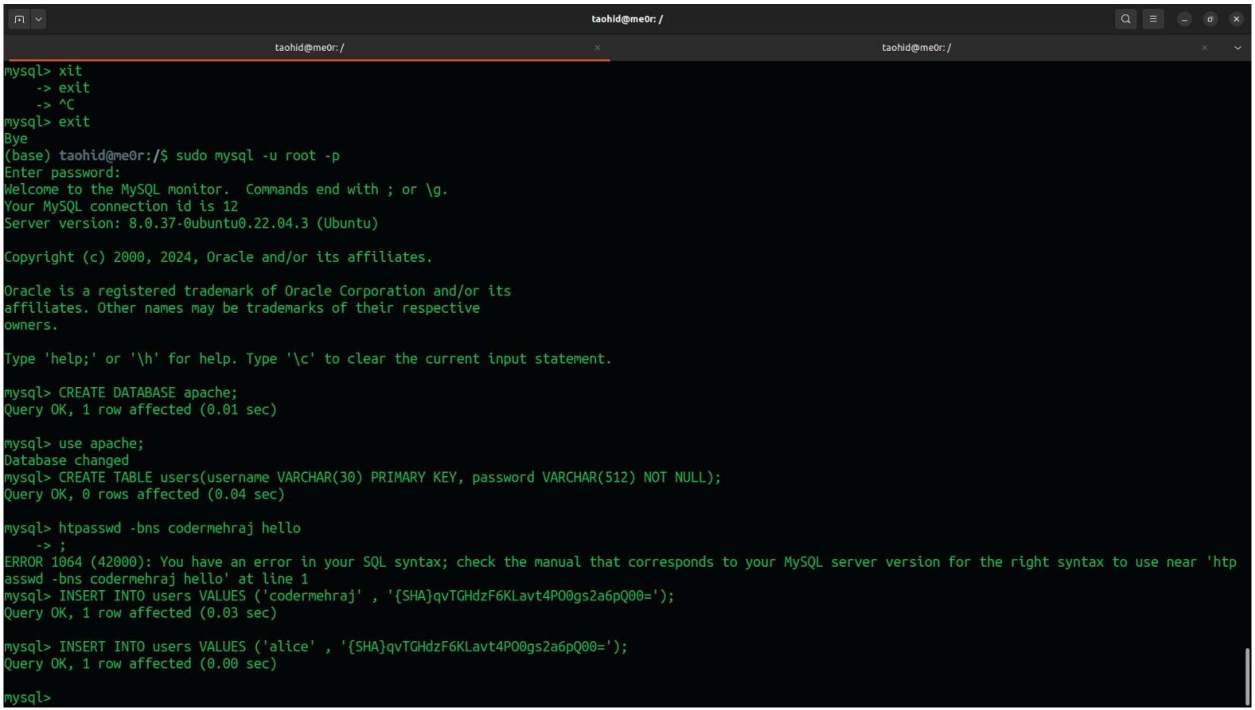
Checkpoint 3: Authentication using MySQL Database

1. Installed mysql
2. Configured mysql using mysql_secure_installation
3. Created Apache database and users table and added the users and hashed passwords
4. Then I enabled some Apache Modules
5. Then removed the configuration before and added new config for mysql
6. Accessed https://example.com and logged in with the MySQL credentials.

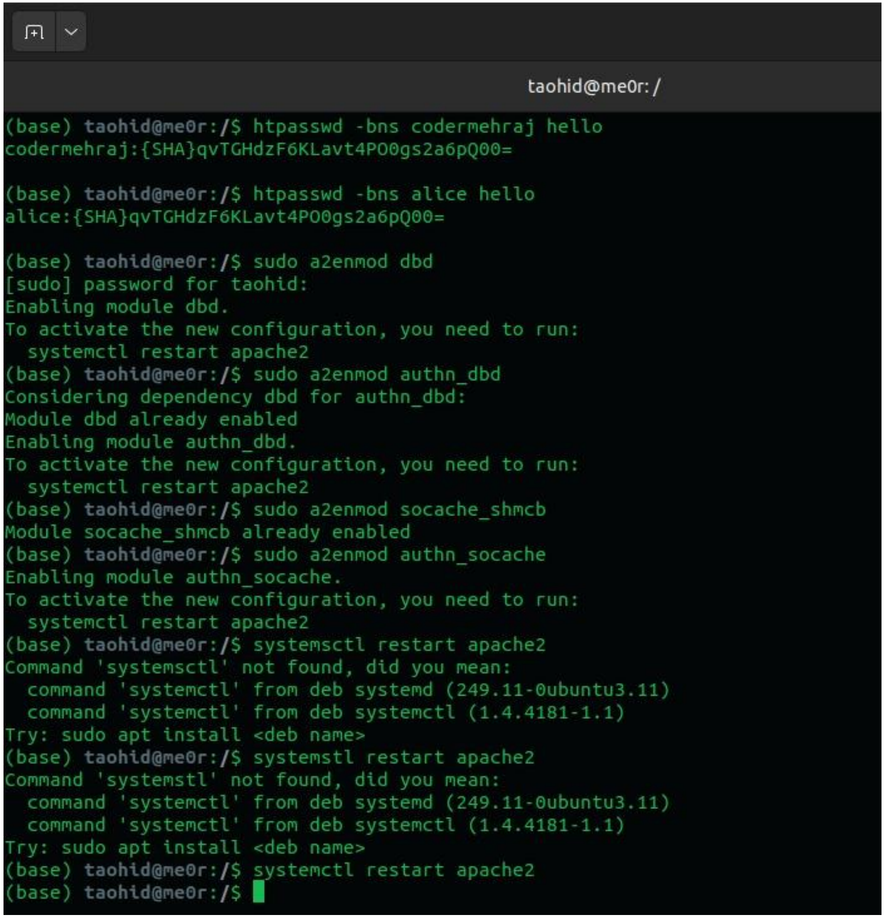
Observations:

- Successfully implemented database-backed authentication. The website could be accessed only by users authenticated through MySQL.

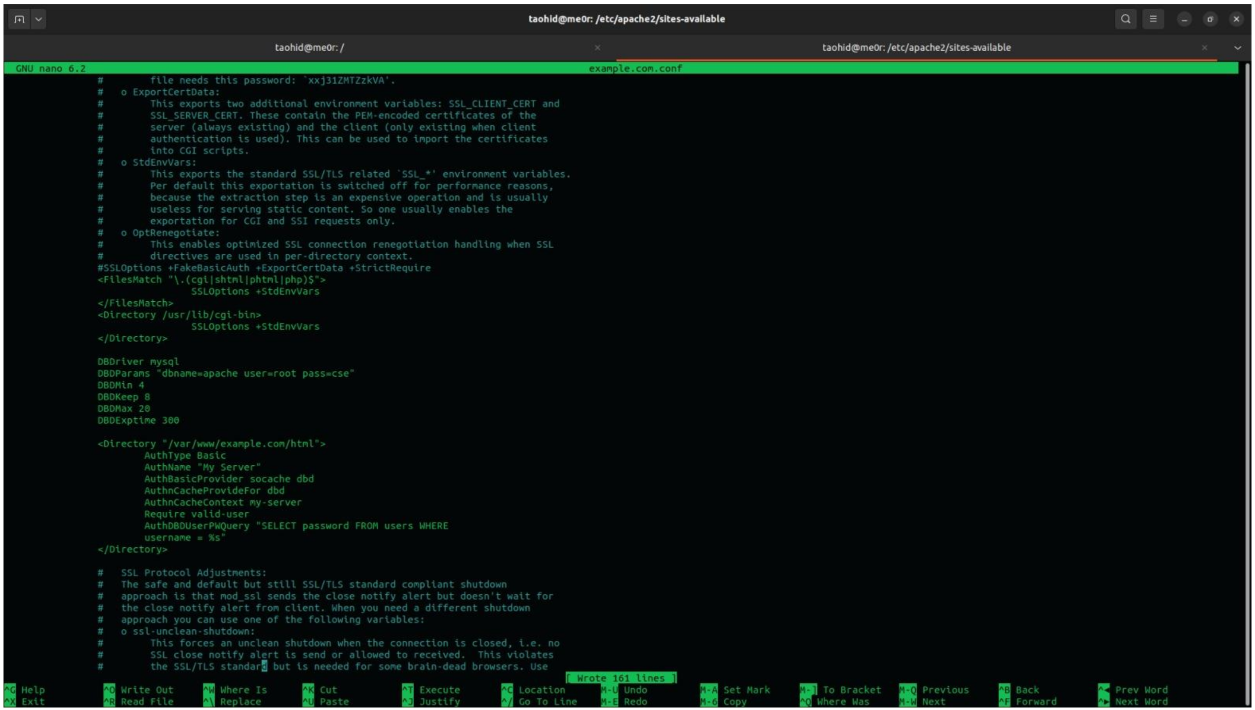
Here are some screenshots of my work:



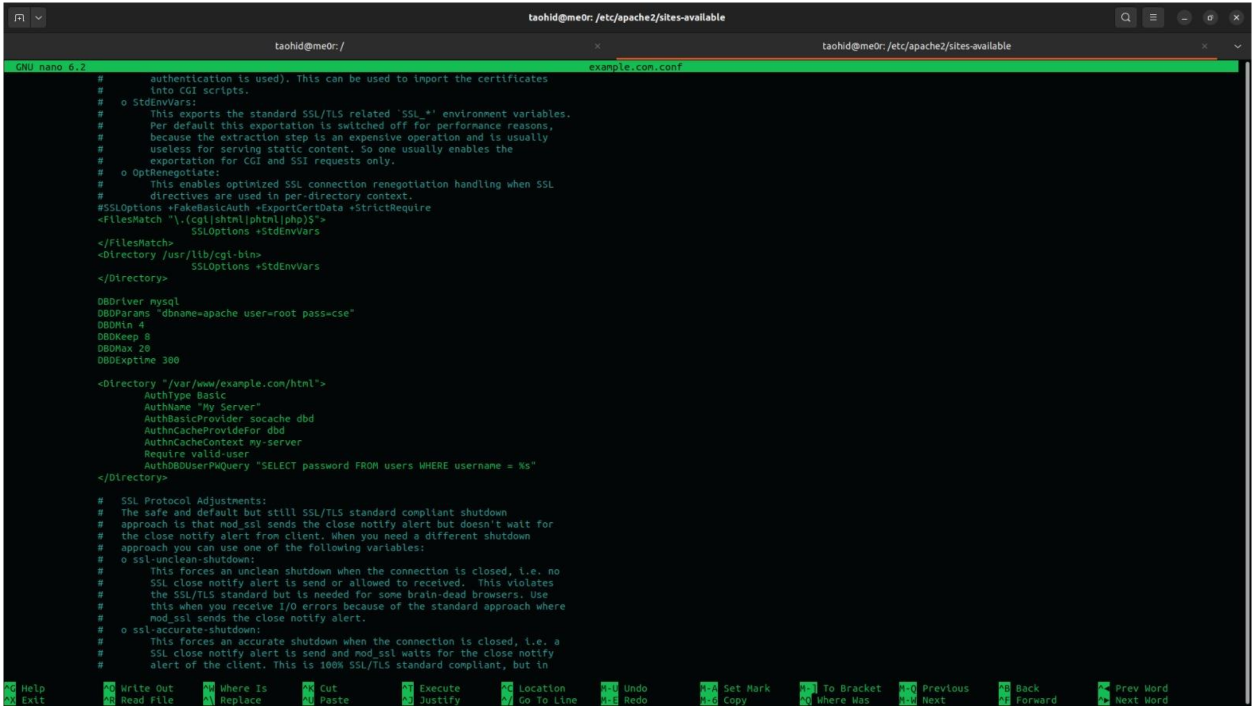
```
mysql> xit
    -> exit
    -> ^C
mysql> exit
Bye
(base) taohid@me0r:/$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.37-0ubuntu0.22.04.3 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> CREATE DATABASE apache;
Query OK, 1 row affected (0.01 sec)

mysql> use apache;
Database changed
mysql> CREATE TABLE users(username VARCHAR(30) PRIMARY KEY, password VARCHAR(512) NOT NULL);
Query OK, 0 rows affected (0.04 sec)

mysql> htpasswd -bns codermehraj hello
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'htp
asswd -bns codermehraj hello' at line 1
mysql> INSERT INTO users VALUES ('codermehraj' , '{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=');
Query OK, 1 row affected (0.03 sec)

mysql> INSERT INTO users VALUES ('alice' , '{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=');
Query OK, 1 row affected (0.00 sec)

mysql>
```



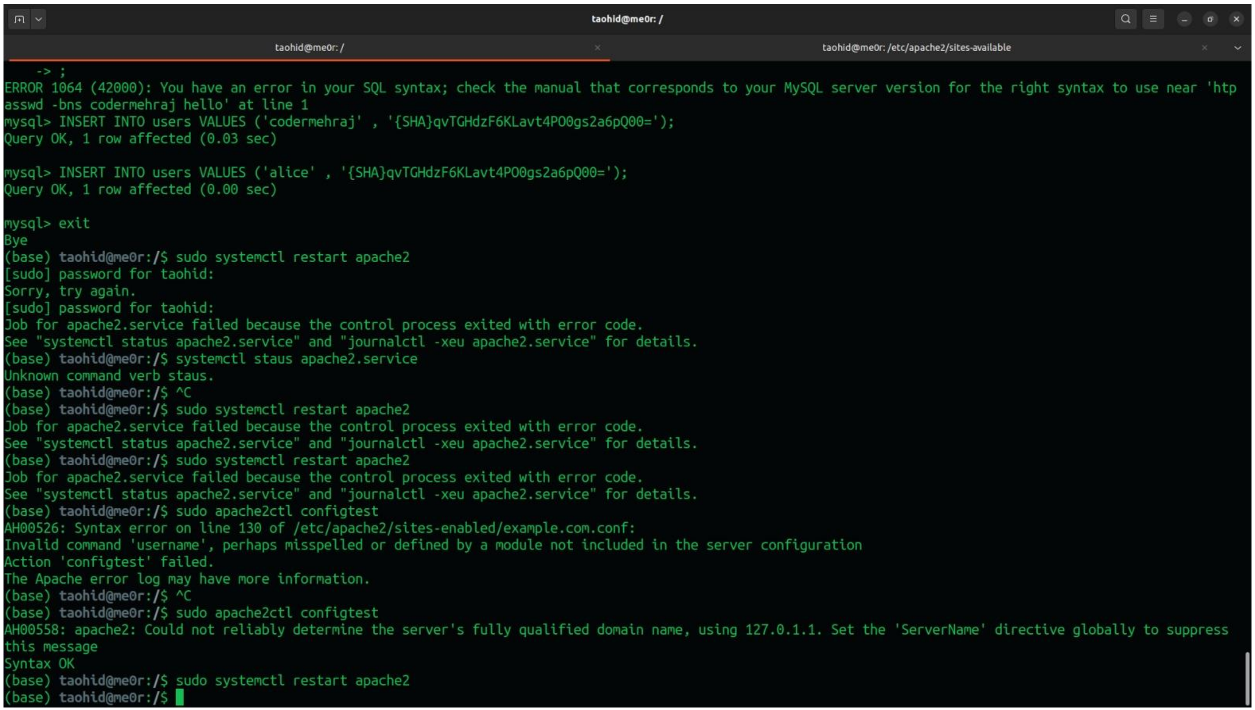
```
(base) taohid@me0r:/$ htpasswd -bns codermehraj hello
codermehraj:{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=

(base) taohid@me0r:/$ htpasswd -bns alice hello
alice:{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=

(base) taohid@me0r:/$ sudo a2enmod dbd
[sudo] password for taohid:
Enabling module dbd.
To activate the new configuration, you need to run:
  systemctl restart apache2
(base) taohid@me0r:/$ sudo a2enmod authn_dbd
Considering dependency dbd for authn_dbd:
Module dbd already enabled
Enabling module authn_dbd.
To activate the new configuration, you need to run:
  systemctl restart apache2
(base) taohid@me0r:/$ sudo a2enmod socache_shmcb
Module socache_shmcb already enabled
(base) taohid@me0r:/$ sudo a2enmod authn_socache
Enabling module authn_socache.
To activate the new configuration, you need to run:
  systemctl restart apache2
(base) taohid@me0r:/$ systemsctl restart apache2
Command 'systemsctl' not found, did you mean:
  command 'systemctl' from deb systemd (249.11-0ubuntu3.11)
  command 'systemctl' from deb systemctl (1.4.4181-1.1)
Try: sudo apt install <deb name>
(base) taohid@me0r:/$ systemstl restart apache2
Command 'systemstl' not found, did you mean:
  command 'systemctl' from deb systemd (249.11-0ubuntu3.11)
  command 'systemctl' from deb systemctl (1.4.4181-1.1)
Try: sudo apt install <deb name>
(base) taohid@me0r:/$ systemctl restart apache2
(base) taohid@me0r:/$ 
```


taohid@me0r: /etc/apache2/sites-available
taohid@me0r: /
taohid@me0r: /etc/apache2/sites-available

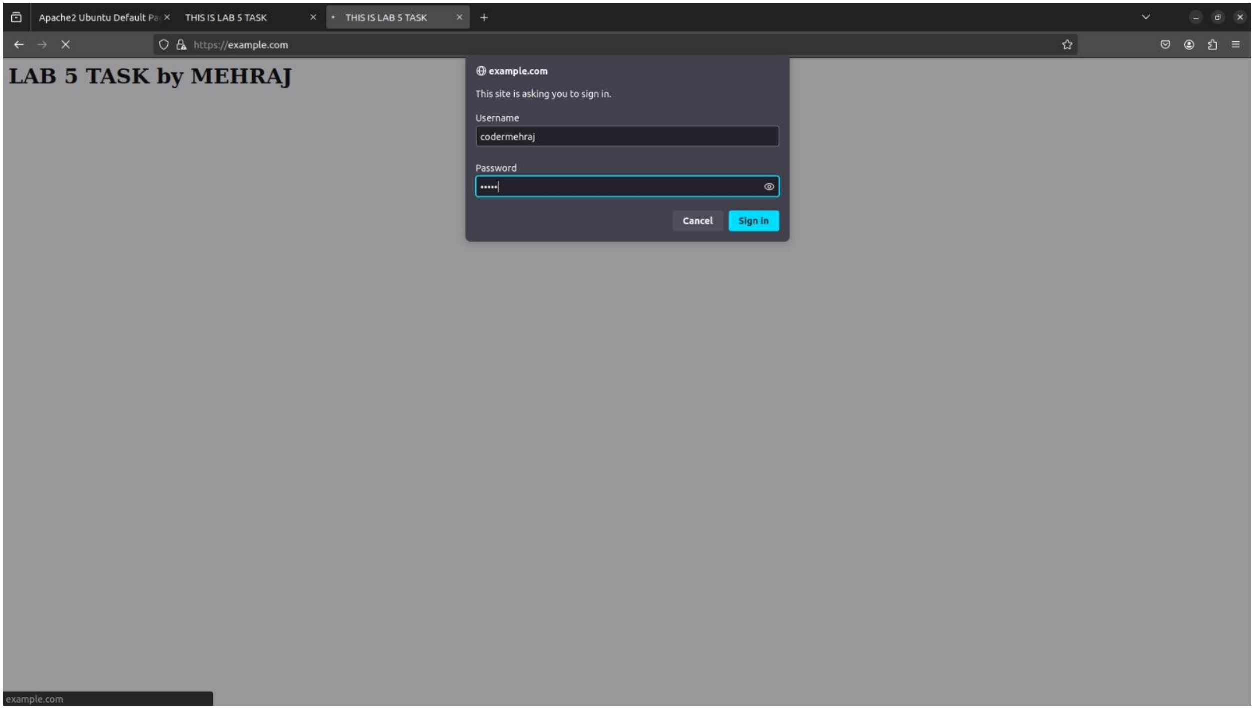
```
  GNU nano 6.2                                        example.com.conf
#        file needs this password: `xxj31ZMTZzkVA'.
#    o ExportCertData:
#        This exports two additional environment variables: SSL_CLIENT_CERT and
#        SSL_SERVER_CERT. These contain the PEM-encoded certificates of the
#        server (always existing) and the client (only existing when client
#        authentication is used). This can be used to import the certificates
#        into CGI scripts.
#    o StdEnvVars:
#        This exports the standard SSL/TLS related `SSL_*' environment variables.
#        Per default this exportation is switched off for performance reasons,
#        because the extraction step is an expensive operation and is usually
#        useless for serving static content. So one usually enables the
#        exportation for CGI and SSI requests only.
#    o OptRenegotiate:
#        This enables optimized SSL connection renegotiation handling when SSL
#        directives are used in per-directory context.
#SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire
<FilesMatch "\.(cgi|shtml|phtml|php)$">
                SSLOptions +StdEnvVars
</FilesMatch>
<Directory /usr/lib/cgi-bin>
                SSLOptions +StdEnvVars
</Directory>

DBDriver mysql
DBDParams "dbname=apache user=root pass=cse"
DBDMin 4
DBDKeep 8
DBDMax 20
DBDExptime 300

<Directory "/var/www/example.com/html">
        AuthType Basic
        AuthName "My Server"
        AuthBasicProvider socache dbd
        AuthnCacheProvideFor dbd
        AuthnCacheContext my-server
        Require valid-user
        AuthDBDUserPWQuery "SELECT password FROM users WHERE
        username = %s"
</Directory>

#   SSL Protocol Adjustments:
#   The safe and default but still SSL/TLS standard compliant shutdown
#   approach is that mod_ssl sends the close notify alert but doesn't wait for
#   the close notify alert from client. When you need a different shutdown
#   approach you can use one of the following variables:
#    o ssl-unclean-shutdown:
#        This forces an unclean shutdown when the connection is closed, i.e. no
#        SSL close notify alert is send or allowed to received.  This violates
#        the SSL/TLS standard but is needed for some brain-dead browsers. Use
                                                 [ Wrote 161 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute   ^C Location   M-U Undo    M-A Set Mark   M-] To Bracket  M-Q Previous  ^B Back     ^◀ Prev Word
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify   ^/ Go To Line M-E Redo    M-6 Copy       ^Q Where Was    M-W Next      ^F Forward  ^▶ Next Word
```


taohid@me0r: /etc/apache2/sites-available
taohid@me0r: /
taohid@me0r: /etc/apache2/sites-available

```
  GNU nano 6.2                                        example.com.conf
#        authentication is used). This can be used to import the certificates
#        into CGI scripts.
#    o StdEnvVars:
#        This exports the standard SSL/TLS related `SSL_*' environment variables.
#        Per default this exportation is switched off for performance reasons,
#        because the extraction step is an expensive operation and is usually
#        useless for serving static content. So one usually enables the
#        exportation for CGI and SSI requests only.
#    o OptRenegotiate:
#        This enables optimized SSL connection renegotiation handling when SSL
#        directives are used in per-directory context.
#SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire
<FilesMatch "\.(cgi|shtml|phtml|php)$">
                SSLOptions +StdEnvVars
</FilesMatch>
<Directory /usr/lib/cgi-bin>
                SSLOptions +StdEnvVars
</Directory>

DBDriver mysql
DBDParams "dbname=apache user=root pass=cse"
DBDMin 4
DBDKeep 8
DBDMax 20
DBDExptime 300

<Directory "/var/www/example.com/html">
        AuthType Basic
        AuthName "My Server"
        AuthBasicProvider socache dbd
        AuthnCacheProvideFor dbd
        AuthnCacheContext my-server
        Require valid-user
        AuthDBDUserPWQuery "SELECT password FROM users WHERE username = %s"
</Directory>

#   SSL Protocol Adjustments:
#   The safe and default but still SSL/TLS standard compliant shutdown
#   approach is that mod_ssl sends the close notify alert but doesn't wait for
#   the close notify alert from client. When you need a different shutdown
#   approach you can use one of the following variables:
#    o ssl-unclean-shutdown:
#        This forces an unclean shutdown when the connection is closed, i.e. no
#        SSL close notify alert is send or allowed to received.  This violates
#        the SSL/TLS standard but is needed for some brain-dead browsers. Use
#        this when you receive I/O errors because of the standard approach where
#        mod_ssl sends the close notify alert.
#    o ssl-accurate-shutdown:
#        This forces an accurate shutdown when the connection is closed, i.e. a
#        SSL close notify alert is send and mod_ssl waits for the close notify
#        alert of the client. This is 100% SSL/TLS standard compliant, but in
^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute   ^C Location   M-U Undo    M-A Set Mark   M-] To Bracket  M-Q Previous  ^B Back     ^◀ Prev Word
^X Exit      ^R Read File  ^\ Replace    ^U Paste     ^J Justify   ^/ Go To Line M-E Redo    M-6 Copy       ^Q Where Was    M-W Next      ^F Forward  ^▶ Next Word
```



```
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'htp
asswd -bns codermehraj hello' at line 1
mysql> INSERT INTO users VALUES ('codermehraj' , '{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=');
Query OK, 1 row affected (0.03 sec)

mysql> INSERT INTO users VALUES ('alice' , '{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=');
Query OK, 1 row affected (0.00 sec)

mysql> exit
Bye
(base) taohid@me0r:/$ sudo systemctl restart apache2
[sudo] password for taohid:
Sorry, try again.
[sudo] password for taohid:
Job for apache2.service failed because the control process exited with error code.
See "systemctl status apache2.service" and "journalctl -xeu apache2.service" for details.
(base) taohid@me0r:/$ systemctl staus apache2.service
Unknown command verb staus.
(base) taohid@me0r:/$ ^C
(base) taohid@me0r:/$ sudo systemctl restart apache2
Job for apache2.service failed because the control process exited with error code.
See "systemctl status apache2.service" and "journalctl -xeu apache2.service" for details.
(base) taohid@me0r:/$ sudo systemctl restart apache2
Job for apache2.service failed because the control process exited with error code.
See "systemctl status apache2.service" and "journalctl -xeu apache2.service" for details.
(base) taohid@me0r:/$ sudo apache2ctl configtest
AH00526: Syntax error on line 130 of /etc/apache2/sites-enabled/example.com.conf:
Invalid command 'username', perhaps misspelled or defined by a module not included in the server configuration
Action 'configtest' failed.
The Apache error log may have more information.
(base) taohid@me0r:/$ ^C
(base) taohid@me0r:/$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress
this message
Syntax OK
(base) taohid@me0r:/$ sudo systemctl restart apache2
(base) taohid@me0r:/$ 
```



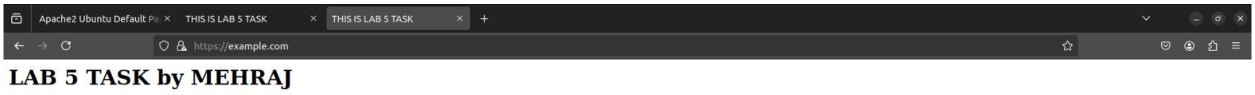
LAB 5 TASK by MEHRAJ

example.com

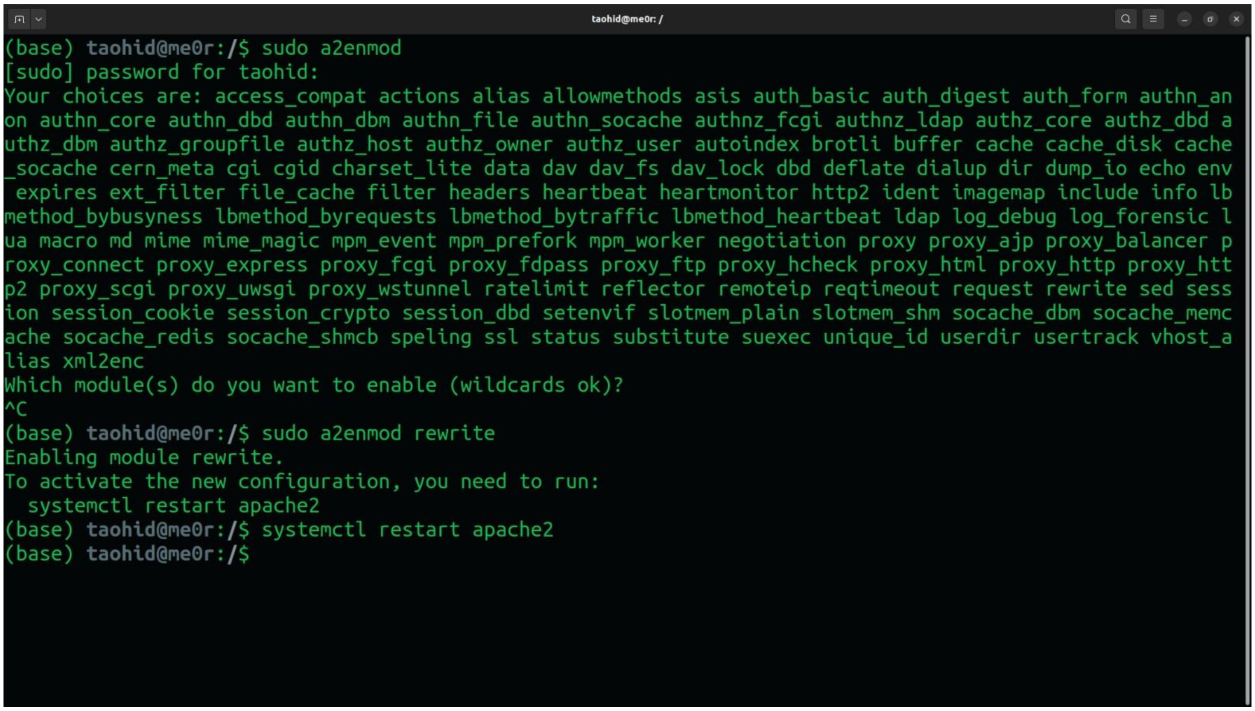
This site is asking you to sign in.

Username: codermehraj

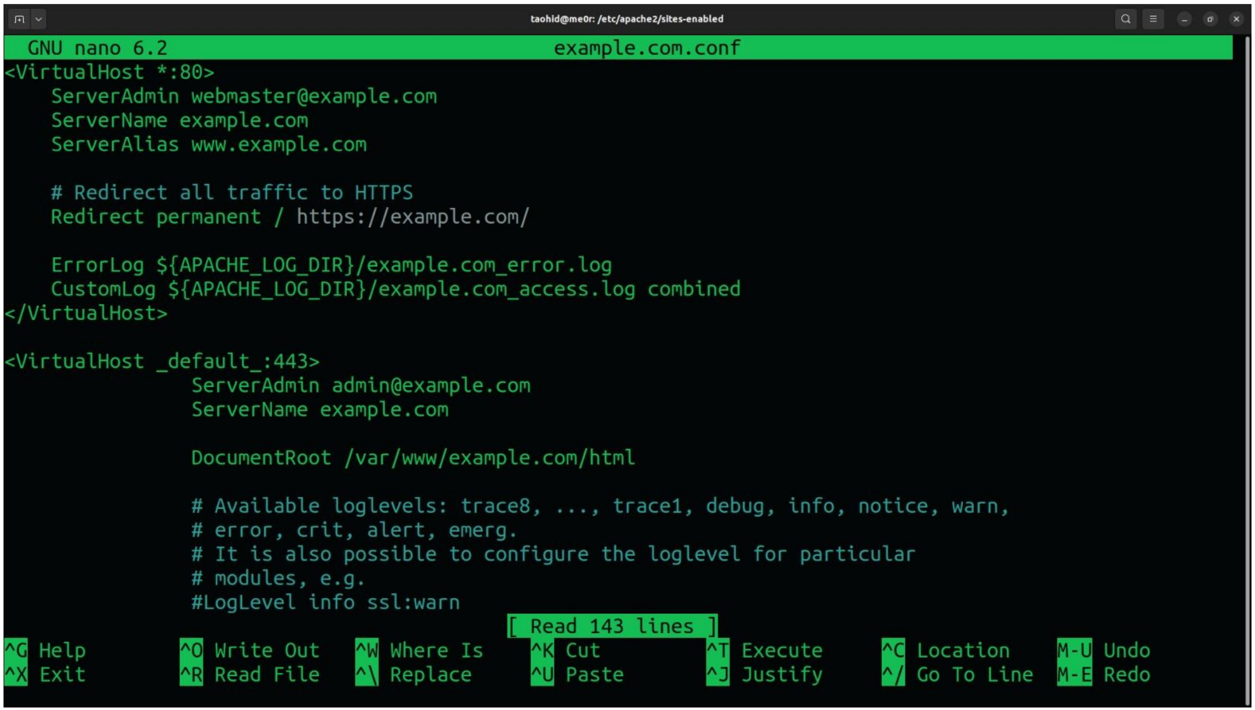
Password: •••••



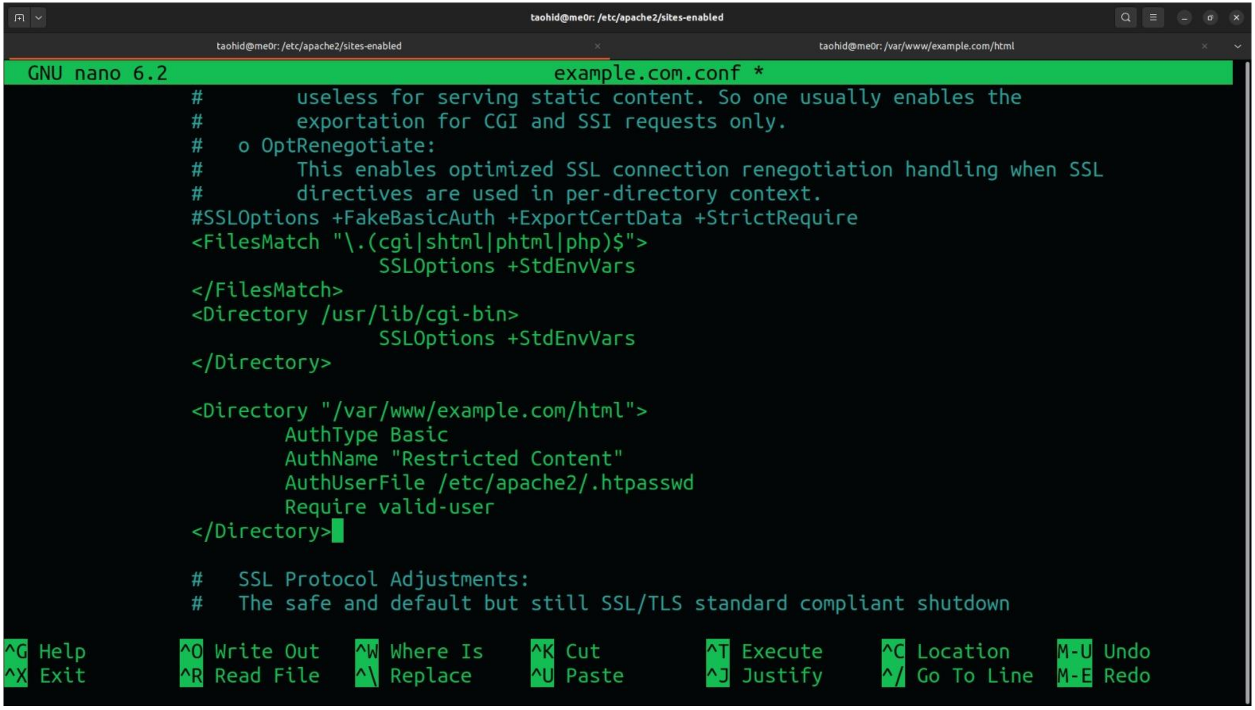
LAB 5 TASK by MEHRAJ



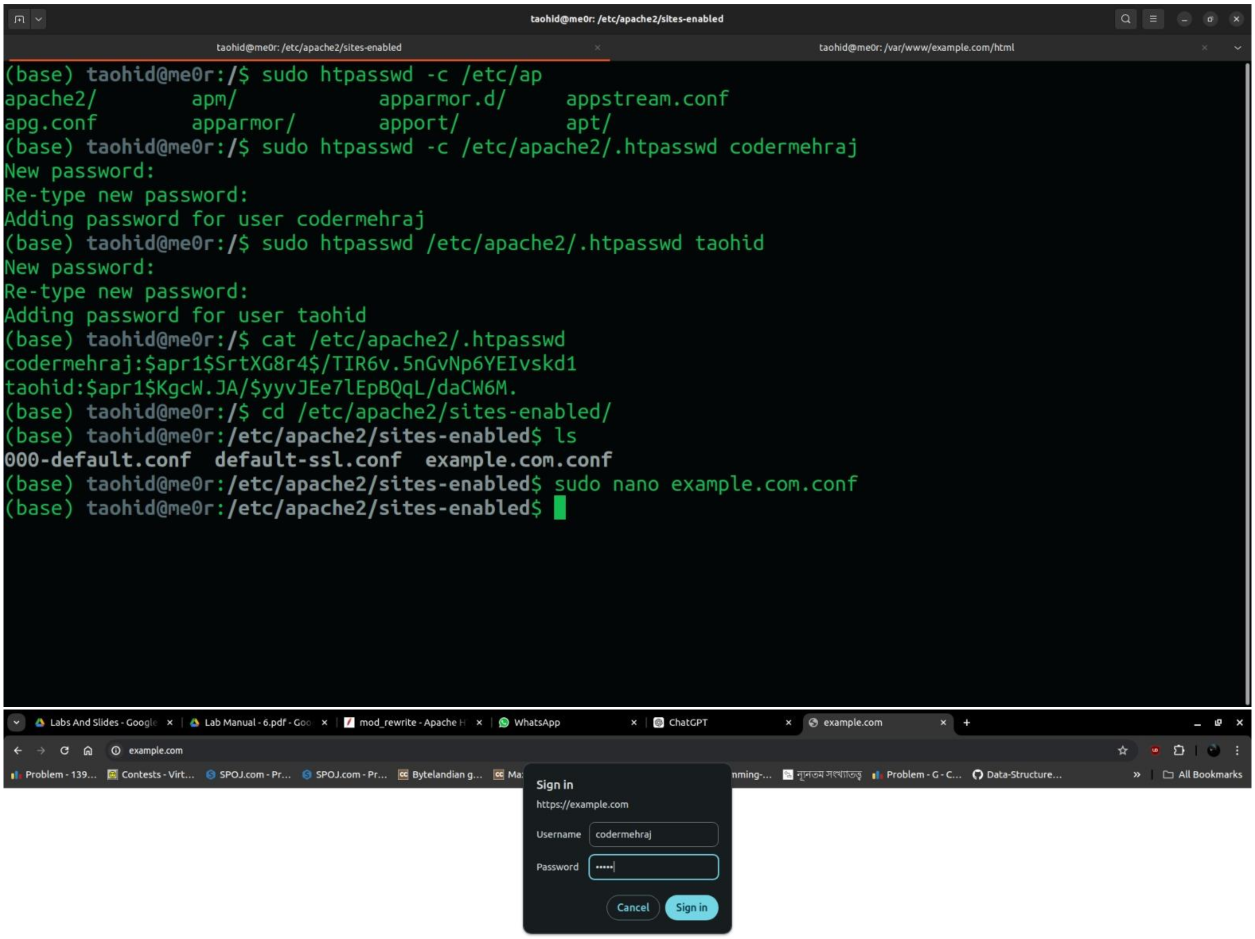
```
(base) taohid@me0r:/$ sudo a2enmod
[sudo] password for taohid:
Your choices are: access_compat actions alias allowmethods asis auth_basic auth_digest auth_form authn_an
on authn_core authn_dbd authn_dbm authn_file authn_socache authnz_fcgi authnz_ldap authz_core authz_dbd a
uthz_dbm authz_groupfile authz_host authz_owner authz_user autoindex brotli buffer cache cache_disk cache
_socache cern_meta cgi cgid charset_lite data dav dav_fs dav_lock dbd deflate dialup dir dump_io echo env
 expires ext_filter file_cache filter headers heartbeat heartmonitor http2 ident imagemap include info lb
method_bybusyness lbmethod_byrequests lbmethod_bytraffic lbmethod_heartbeat ldap log_debug log_forensic l
ua macro md mime mime_magic mpm_event mpm_prefork mpm_worker negotiation proxy proxy_ajp proxy_balancer p
roxy_connect proxy_express proxy_fcgi proxy_fdpass proxy_ftp proxy_hcheck proxy_html proxy_http proxy_htt
p2 proxy_scgi proxy_uwsgi proxy_wstunnel ratelimit reflector remoteip reqtimeout request rewrite sed sess
ion session_cookie session_crypto session_dbd setenvif slotmem_plain slotmem_shm socache_dbm socache_memc
ache socache_redis socache_shmcb speling ssl status substitute suexec unique_id userdir usertrack vhost_a
lias xml2enc
Which module(s) do you want to enable (wildcards ok)?
^C
(base) taohid@me0r:/$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
(base) taohid@me0r:/$ systemctl restart apache2
(base) taohid@me0r:/$
```



```
  GNU nano 6.2                                  example.com.conf
<VirtualHost *:80>
    ServerAdmin webmaster@example.com
    ServerName example.com
    ServerAlias www.example.com

    # Redirect all traffic to HTTPS
    Redirect permanent / https://example.com/

    ErrorLog ${APACHE_LOG_DIR}/example.com_error.log
    CustomLog ${APACHE_LOG_DIR}/example.com_access.log combined
</VirtualHost>

<VirtualHost _default_:443>
                ServerAdmin admin@example.com
                ServerName example.com

                DocumentRoot /var/www/example.com/html

                # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
                # error, crit, alert, emerg.
                # It is also possible to configure the loglevel for particular
                # modules, e.g.
                #LogLevel info ssl:warn
                                           [ Read 143 lines ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo
```


taohid@me0r: /etc/apache2/sites-enabled
taohid@me0r: /etc/apache2/sites-enabled
taohid@me0r: /var/www/example.com/html

```
  GNU nano 6.2                                  example.com.conf *
                #        useless for serving static content. So one usually enables the
                #        exportation for CGI and SSI requests only.
                #   o OptRenegotiate:
                #        This enables optimized SSL connection renegotiation handling when SSL
                #        directives are used in per-directory context.
                #SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire
                <FilesMatch "\.(cgi|shtml|phtml|php)$">
                                SSLOptions +StdEnvVars
                </FilesMatch>
                <Directory /usr/lib/cgi-bin>
                                SSLOptions +StdEnvVars
                </Directory>

                <Directory "/var/www/example.com/html">
                        AuthType Basic
                        AuthName "Restricted Content"
                        AuthUserFile /etc/apache2/.htpasswd
                        Require valid-user
                </Directory>

                #   SSL Protocol Adjustments:
                #   The safe and default but still SSL/TLS standard compliant shutdown

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo
```



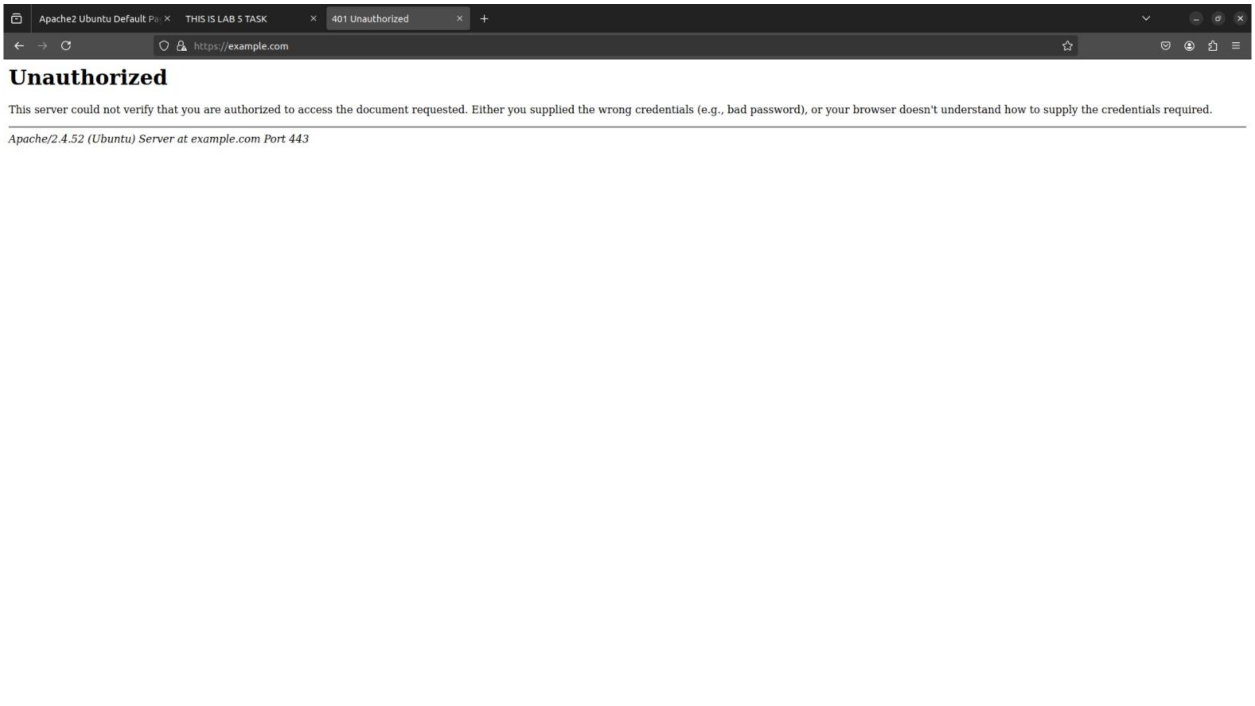
```
(base) taohid@me0r:/$ sudo htpasswd -c /etc/ap
apache2/        apm/            apparmor.d/     appstream.conf
apg.conf        apparmor/       apport/         apt/
(base) taohid@me0r:/$ sudo htpasswd -c /etc/apache2/.htpasswd codermehraj
New password:
Re-type new password:
Adding password for user codermehraj
(base) taohid@me0r:/$ sudo htpasswd /etc/apache2/.htpasswd taohid
New password:
Re-type new password:
Adding password for user taohid
(base) taohid@me0r:/$ cat /etc/apache2/.htpasswd
codermehraj:$apr1$SrtXG8r4$/TIR6v.5nGvNp6YEIvskd1
taohid:$apr1$KgcW.JA/$yyvJEe7lEpBQqL/daCW6M.
(base) taohid@me0r:/$ cd /etc/apache2/sites-enabled/
(base) taohid@me0r:/etc/apache2/sites-enabled$ ls
000-default.conf  default-ssl.conf  example.com.conf
(base) taohid@me0r:/etc/apache2/sites-enabled$ sudo nano example.com.conf
(base) taohid@me0r:/etc/apache2/sites-enabled$ 
```

Sign in

https://example.com

Username: codermehraj

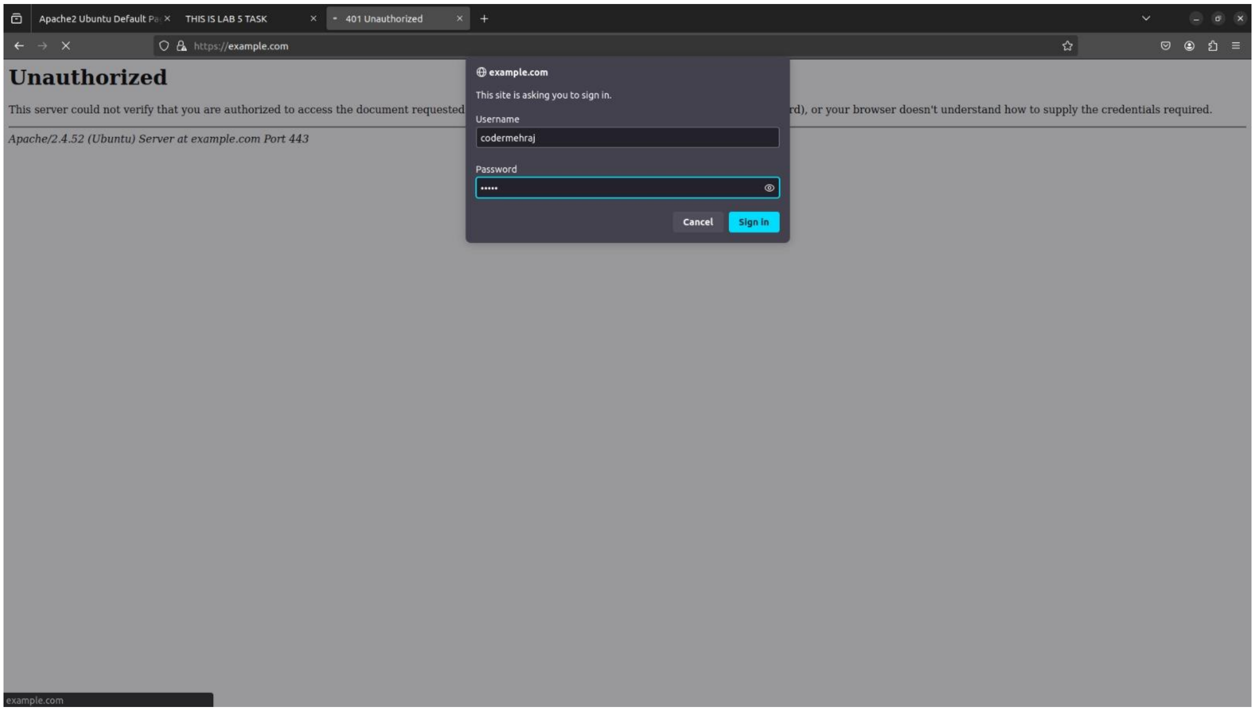
Password: •••••



Unauthorized

This server could not verify that you are authorized to access the document requested. Either you supplied the wrong credentials (e.g., bad password), or your browser doesn't understand how to supply the credentials required.

Apache/2.4.52 (Ubuntu) Server at example.com Port 443



Unauthorized

This server could not verify that you are authorized to access the document requested ... rd), or your browser doesn't understand how to supply the credentials required.

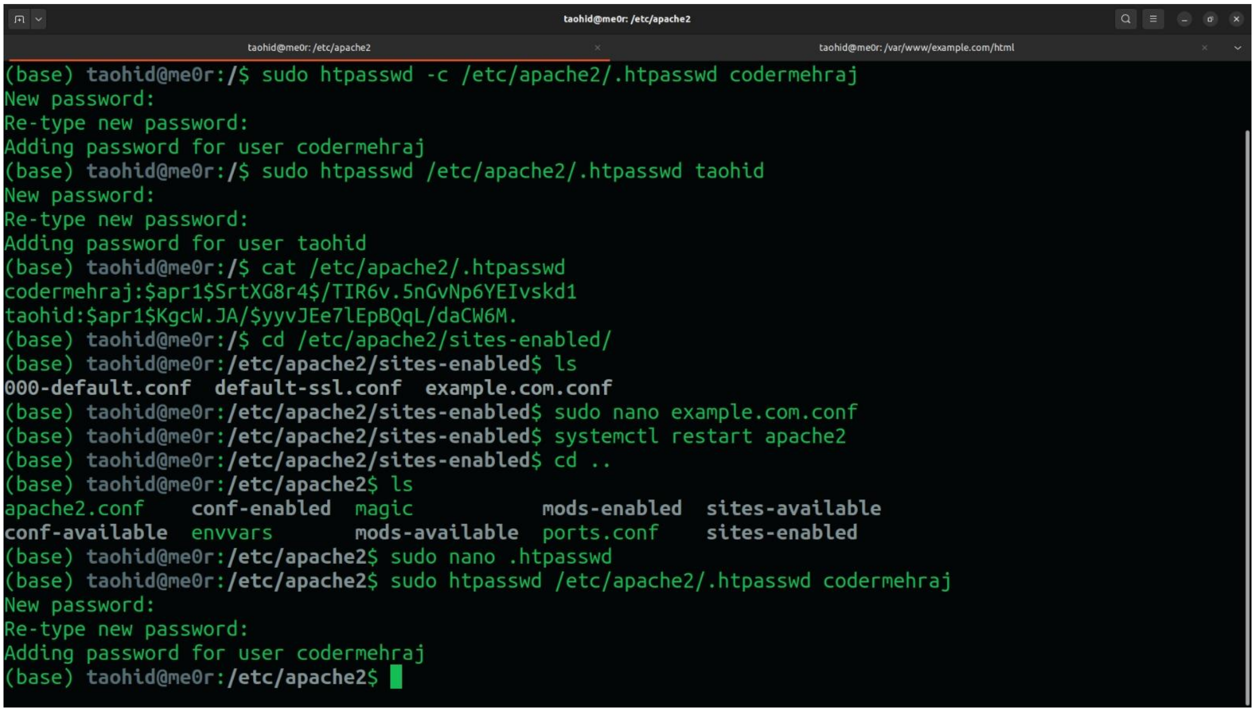
Apache/2.4.52 (Ubuntu) Server at example.com Port 443

example.com

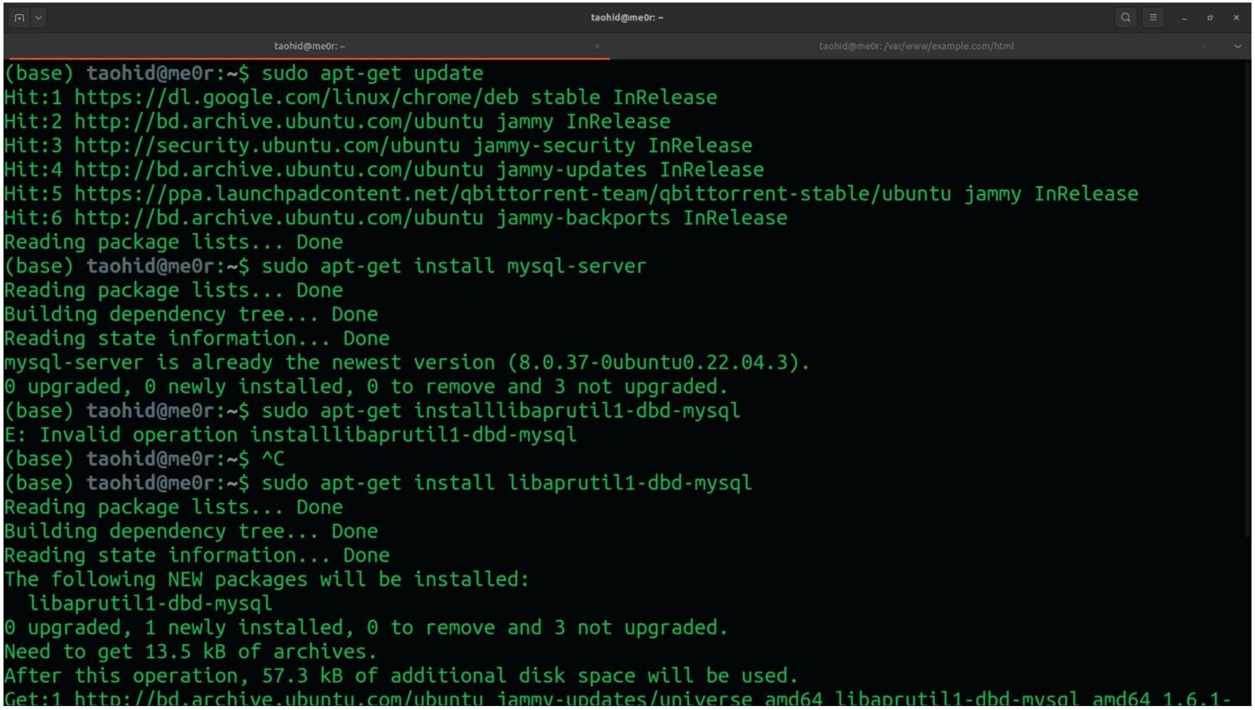
This site is asking you to sign in.

Username: codermehraj

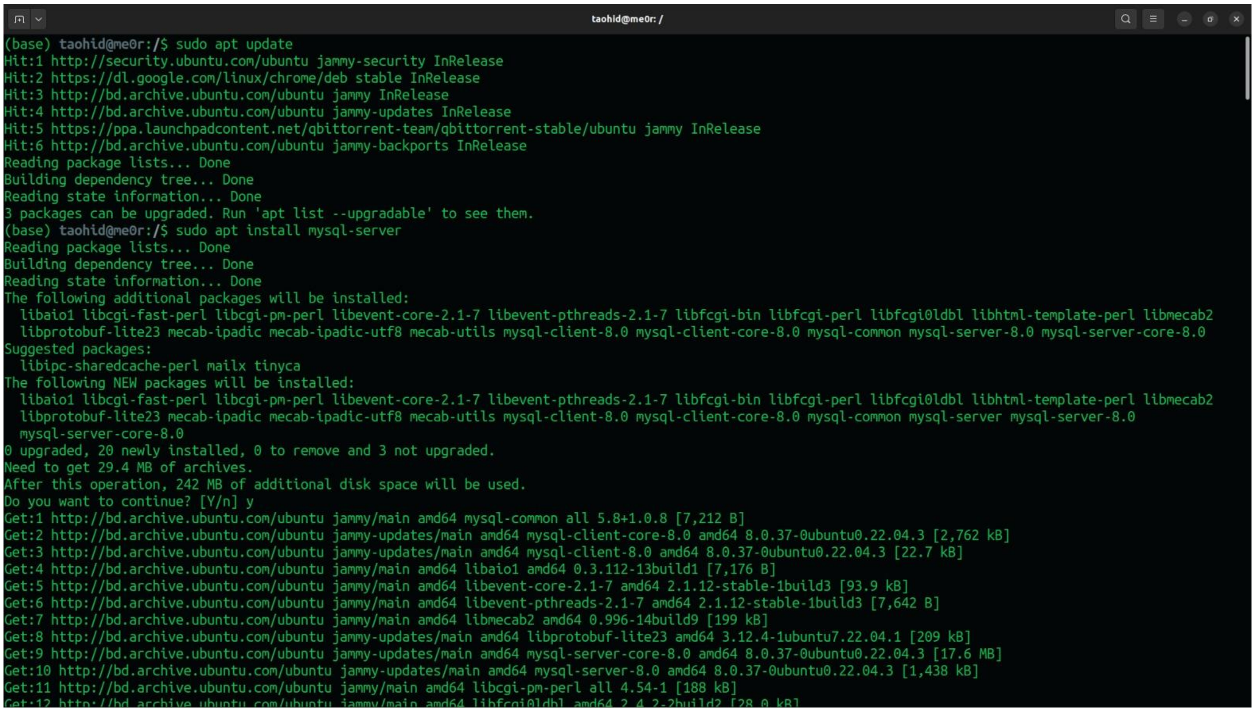
Password: •••••


taohid@me0r: /etc/apache2
taohid@me0r: /etc/apache2
taohid@me0r: /var/www/example.com/html

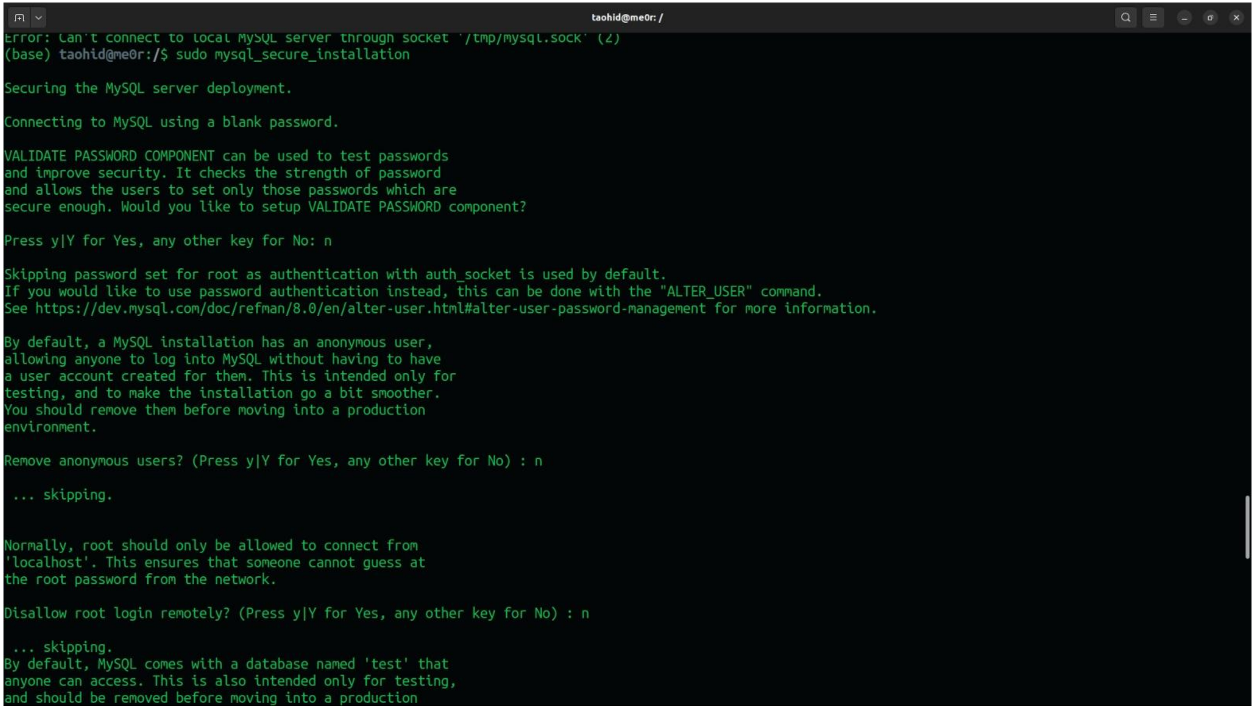
```
(base) taohid@me0r:/$ sudo htpasswd -c /etc/apache2/.htpasswd codermehraj
New password:
Re-type new password:
Adding password for user codermehraj
(base) taohid@me0r:/$ sudo htpasswd /etc/apache2/.htpasswd taohid
New password:
Re-type new password:
Adding password for user taohid
(base) taohid@me0r:/$ cat /etc/apache2/.htpasswd
codermehraj:$apr1$SrtXG8r4$/TIR6v.5nGvNp6YEIvskd1
taohid:$apr1$KgcW.JA/$yyvJEe7lEpBQqL/daCW6M.
(base) taohid@me0r:/$ cd /etc/apache2/sites-enabled/
(base) taohid@me0r:/etc/apache2/sites-enabled$ ls
000-default.conf  default-ssl.conf  example.com.conf
(base) taohid@me0r:/etc/apache2/sites-enabled$ sudo nano example.com.conf
(base) taohid@me0r:/etc/apache2/sites-enabled$ systemctl restart apache2
(base) taohid@me0r:/etc/apache2/sites-enabled$ cd ..
(base) taohid@me0r:/etc/apache2$ ls
apache2.conf    conf-enabled  magic           mods-enabled  sites-available
conf-available  envvars       mods-available  ports.conf    sites-enabled
(base) taohid@me0r:/etc/apache2$ sudo nano .htpasswd
(base) taohid@me0r:/etc/apache2$ sudo htpasswd /etc/apache2/.htpasswd codermehraj
New password:
Re-type new password:
Adding password for user codermehraj
(base) taohid@me0r:/etc/apache2$ 
```



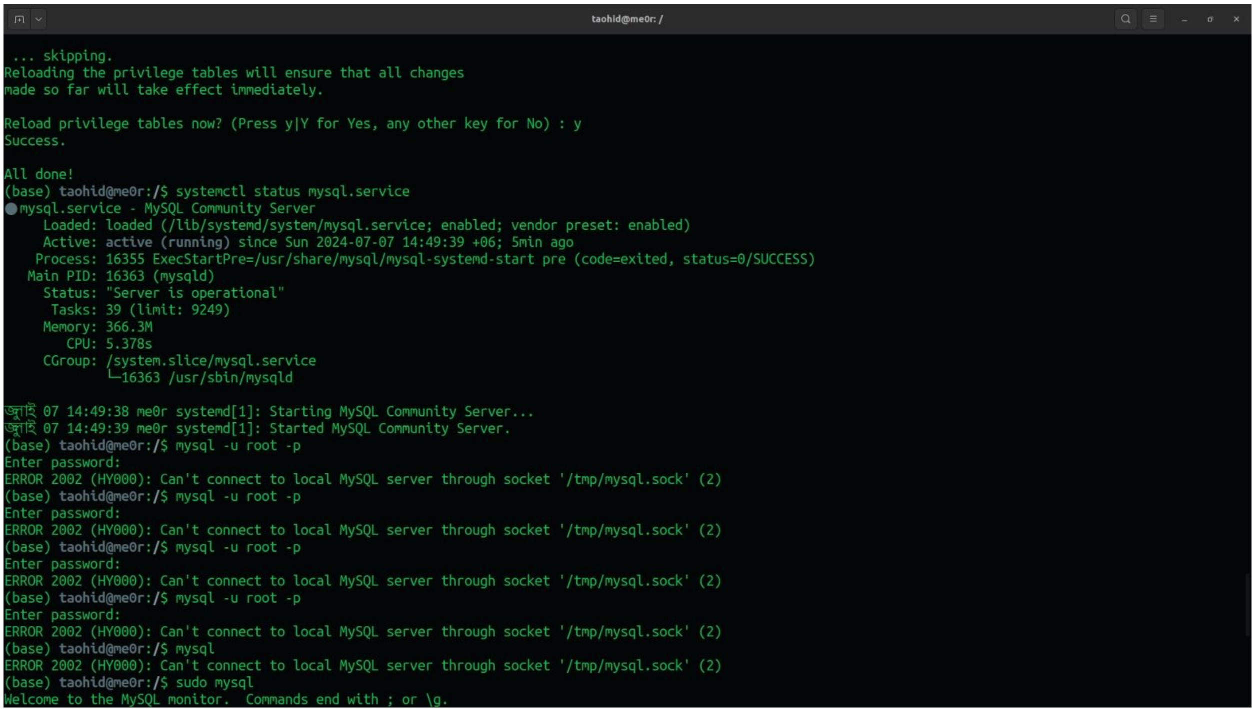
```
(base) taohid@me0r:~$ sudo apt-get update
Hit:1 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:2 http://bd.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:4 http://bd.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:5 https://ppa.launchpadcontent.net/qbittorrent-team/qbittorrent-stable/ubuntu jammy InRelease
Hit:6 http://bd.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
(base) taohid@me0r:~$ sudo apt-get install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mysql-server is already the newest version (8.0.37-0ubuntu0.22.04.3).
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
(base) taohid@me0r:~$ sudo apt-get installlibaprutil1-dbd-mysql
E: Invalid operation installlibaprutil1-dbd-mysql
(base) taohid@me0r:~$ ^C
(base) taohid@me0r:~$ sudo apt-get install libaprutil1-dbd-mysql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  libaprutil1-dbd-mysql
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 13.5 kB of archives.
After this operation, 57.3 kB of additional disk space will be used.
Get:1 http://bd.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 libaprutil1-dbd-mysql amd64 1.6.1-
```



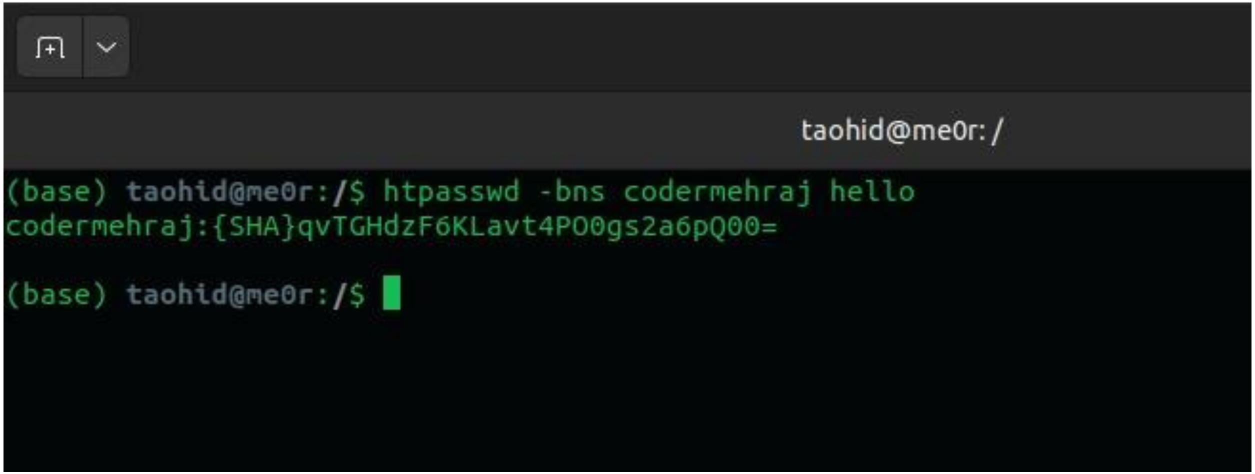
```
(base) taohid@me0r:/$ sudo apt update
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 https://dl.google.com/linux/chrome/deb stable InRelease
Hit:3 http://bd.archive.ubuntu.com/ubuntu jammy InRelease
Hit:4 http://bd.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:5 https://ppa.launchpadcontent.net/qbittorrent-team/qbittorrent-stable/ubuntu jammy InRelease
Hit:6 http://bd.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
3 packages can be upgraded. Run 'apt list --upgradable' to see them.
(base) taohid@me0r:/$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7 libevent-pthreads-2.1-7 libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl libmecab2
  libprotobuf-lite23 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common mysql-server-8.0 mysql-server-core-8.0
Suggested packages:
  libipc-sharedcache-perl mailx tinyca
The following NEW packages will be installed:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7 libevent-pthreads-2.1-7 libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-template-perl libmecab2
  libprotobuf-lite23 mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common mysql-server mysql-server-8.0
  mysql-server-core-8.0
0 upgraded, 20 newly installed, 0 to remove and 3 not upgraded.
Need to get 29.4 MB of archives.
After this operation, 242 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 mysql-common all 5.8+1.0.8 [7,212 B]
Get:2 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-core-8.0 amd64 8.0.37-0ubuntu0.22.04.3 [2,762 kB]
Get:3 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-8.0 amd64 8.0.37-0ubuntu0.22.04.3 [22.7 kB]
Get:4 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 libaio1 amd64 0.3.112-13build1 [7,176 B]
Get:5 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 libevent-core-2.1-7 amd64 2.1.12-stable-1build3 [93.9 kB]
Get:6 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 libevent-pthreads-2.1-7 amd64 2.1.12-stable-1build3 [7,642 B]
Get:7 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 libmecab2 amd64 0.996-14build9 [199 kB]
Get:8 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libprotobuf-lite23 amd64 3.12.4-1ubuntu7.22.04.1 [209 kB]
Get:9 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-server-core-8.0 amd64 8.0.37-0ubuntu0.22.04.3 [17.6 MB]
Get:10 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-server-8.0 amd64 8.0.37-0ubuntu0.22.04.3 [1,438 kB]
Get:11 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 libcgi-pm-perl all 4.54-1 [188 kB]
Get:12 http://bd.archive.ubuntu.com/ubuntu jammy/main amd64 libfcgi0ldbl amd64 2.4.2-2build2 [28.0 kB]
```



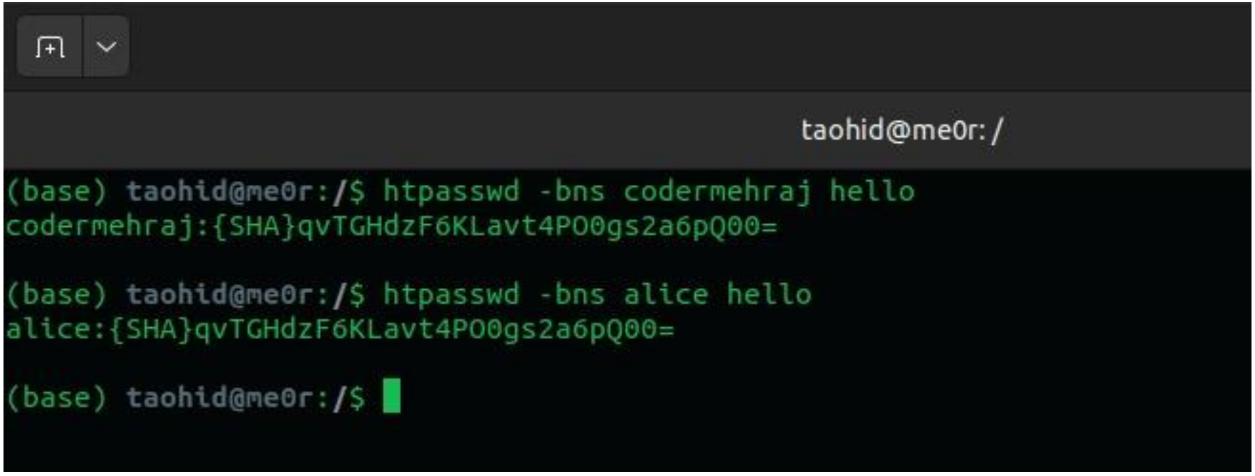
```
Error: Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)
(base) taohid@me0r:/$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: n

Skipping password set for root as authentication with auth_socket is used by default.
If you would like to use password authentication instead, this can be done with the "ALTER_USER" command.
See https://dev.mysql.com/doc/refman/8.0/en/alter-user.html#alter-user-password-management for more information.

By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : n

 ... skipping.


Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : n

 ... skipping.
By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
```



```
 ... skipping.
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
(base) taohid@me0r:/$ systemctl status mysql.service
● mysql.service - MySQL Community Server
     Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2024-07-07 14:49:39 +06; 5min ago
    Process: 16355 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=exited, status=0/SUCCESS)
   Main PID: 16363 (mysqld)
     Status: "Server is operational"
      Tasks: 39 (limit: 9249)
     Memory: 366.3M
        CPU: 5.378s
     CGroup: /system.slice/mysql.service
             └─16363 /usr/sbin/mysqld

জুলাই 07 14:49:38 me0r systemd[1]: Starting MySQL Community Server...
জুলাই 07 14:49:39 me0r systemd[1]: Started MySQL Community Server.
(base) taohid@me0r:/$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)
(base) taohid@me0r:/$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)
(base) taohid@me0r:/$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)
(base) taohid@me0r:/$ mysql -u root -p
Enter password:
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)
(base) taohid@me0r:/$ mysql
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/tmp/mysql.sock' (2)
(base) taohid@me0r:/$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
```



```
(base) taohid@me0r:/$ htpasswd -bns codermehraj hello
codermehraj:{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=

(base) taohid@me0r:/$ 
```



```
(base) taohid@me0r:/$ htpasswd -bns codermehraj hello
codermehraj:{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=

(base) taohid@me0r:/$ htpasswd -bns alice hello
alice:{SHA}qvTGHdzF6KLavt4PO0gs2a6pQ00=

(base) taohid@me0r:/$ 
```