Lab Report: Securing Apache Web Server - 2

Task 6

Md Sadman Hafiz-2018831057

Introduction

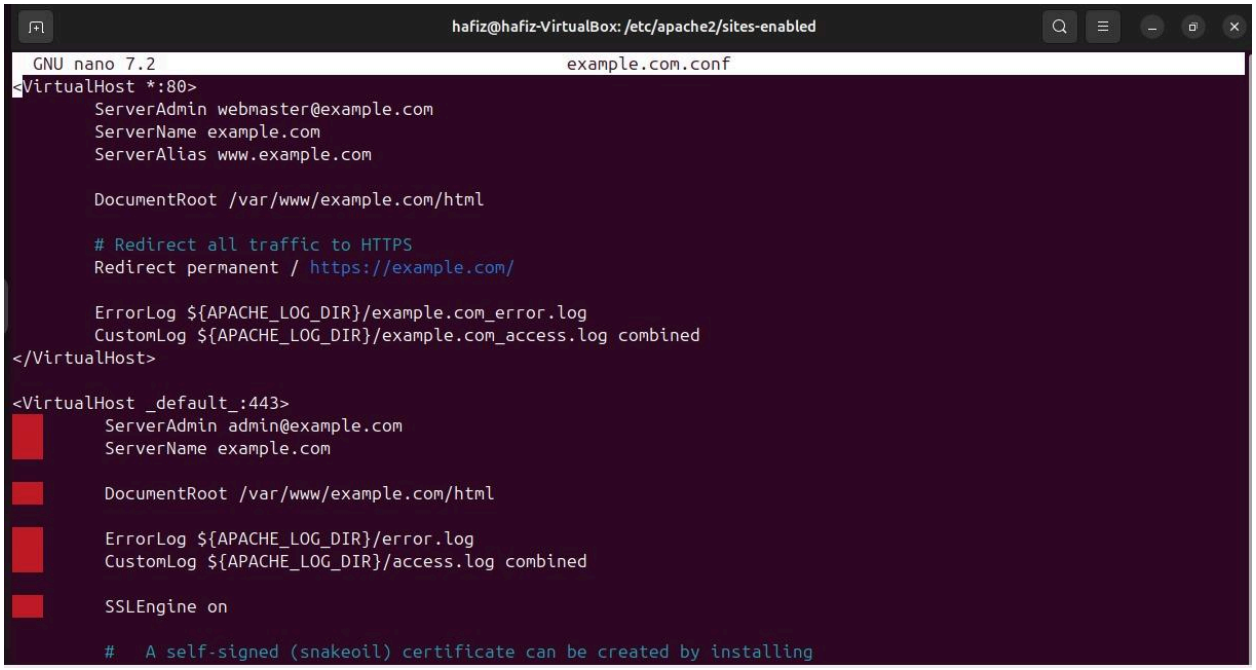
This lab focuses on securing an Apache web server using different authentication mechanisms. The main objectives are to ensure that the web server is only accessible via HTTPS, implement basic authentication, and use a database for user authentication. The tasks are divided into several checkpoints to ensure the proper setup and functioning of the secure web server.

Task-1 :

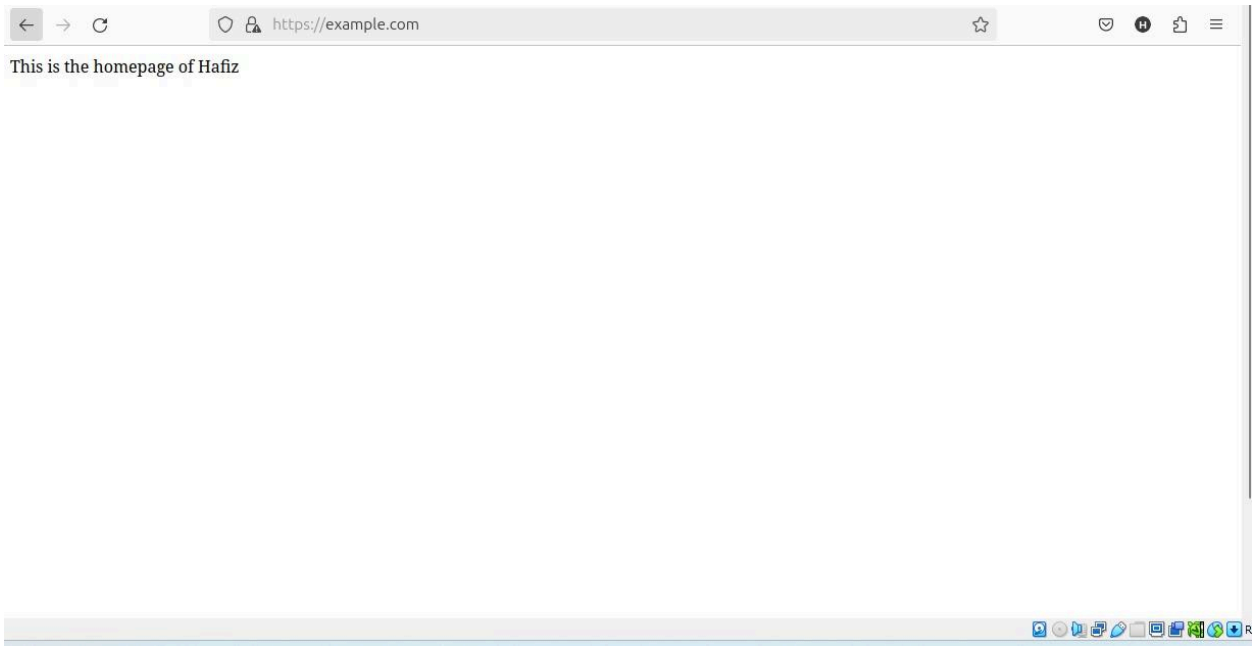
Redirect HTTP to HTTPS:

This step enables the `mod_rewrite` module in Apache, which allows URL rewriting.

This configuration ensures that any HTTP requests are redirected to HTTPS.



Access `http://example.com` in the browser and verify that it is redirected to `https://example.com`.



Task 2: Implement Basic Authentication:

Step 1: Add Users to Apache

These commands test the Apache configuration for syntax errors and restart the Apache server.

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ systemctl restart apache2
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo nano example.com.conf
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo htpasswd -c /etc/apache2/.htpasswd hafiz
New password:
Re-type new password:
Adding password for user hafiz
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$
```

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo htpasswd /etc/apache2/.htpasswd sadman
New password:
Re-type new password:
Adding password for user sadman
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ cat /etc/apache2/.htpasswd
hafiz:$apr1$U4/nOZ1j$1QBXi6zxsUPq.971o7F6A1
sadman:$apr1$cMmdn0vk$chkXKPz4kDfub/bpDkj5s.
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$
```

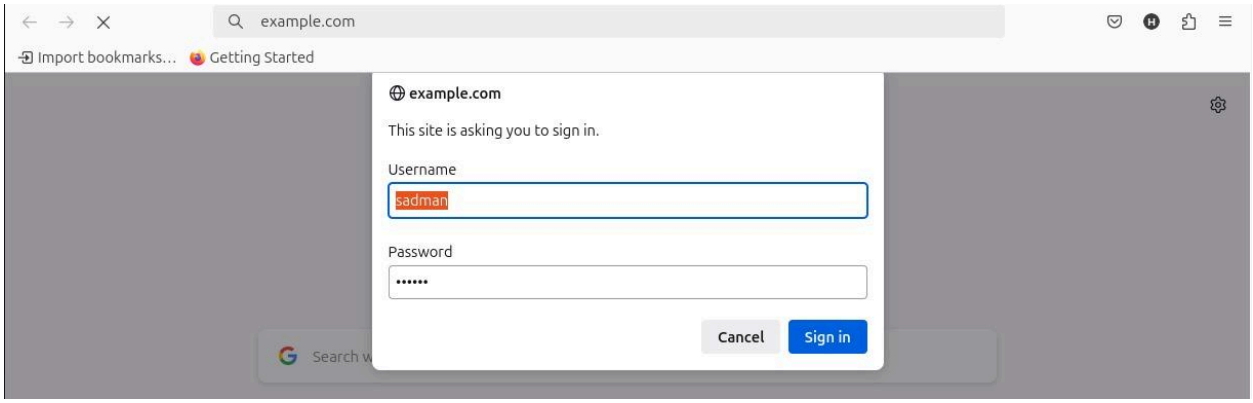
Configure Authentication in Apache:

This configuration restricts access to authenticated users.

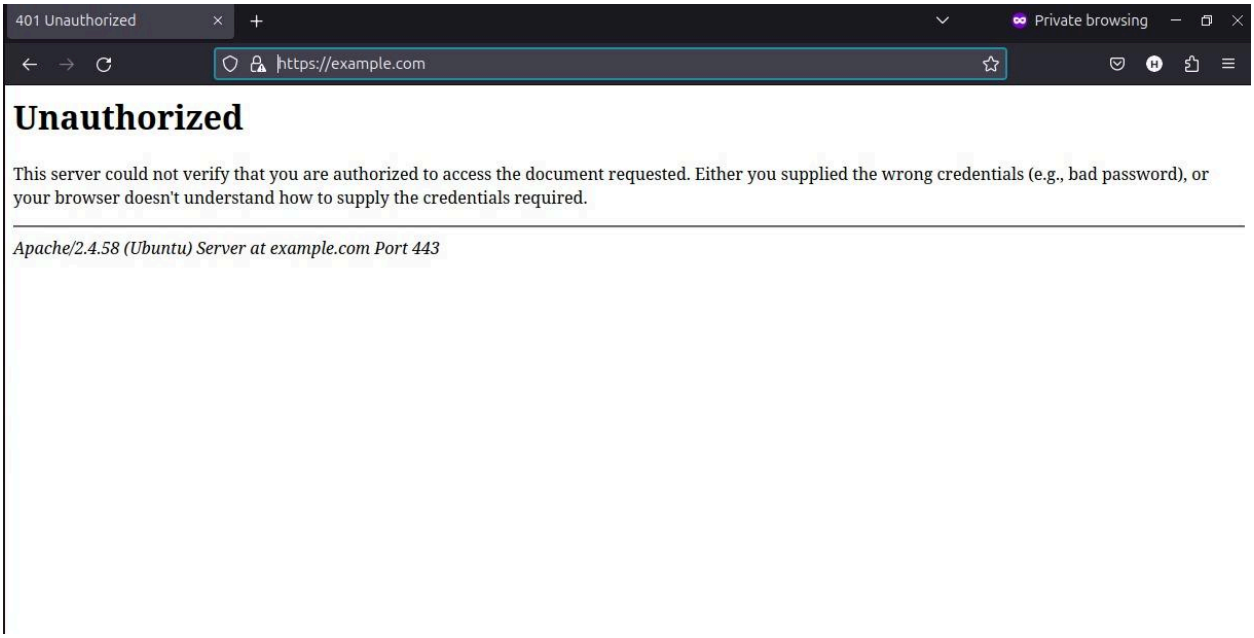
```
#     authentication is used). This can be used to import the certifica
<Directory "/var/www/example.com/html">
AuthType Basic
AuthName "Restricted Content"
AuthUserFile /etc/apache2/.htpasswd
Require valid-user
</Directory>
```

Verify Basic Authentication

Access `https://example.com` in the browser, enter the username and password, and verify access.



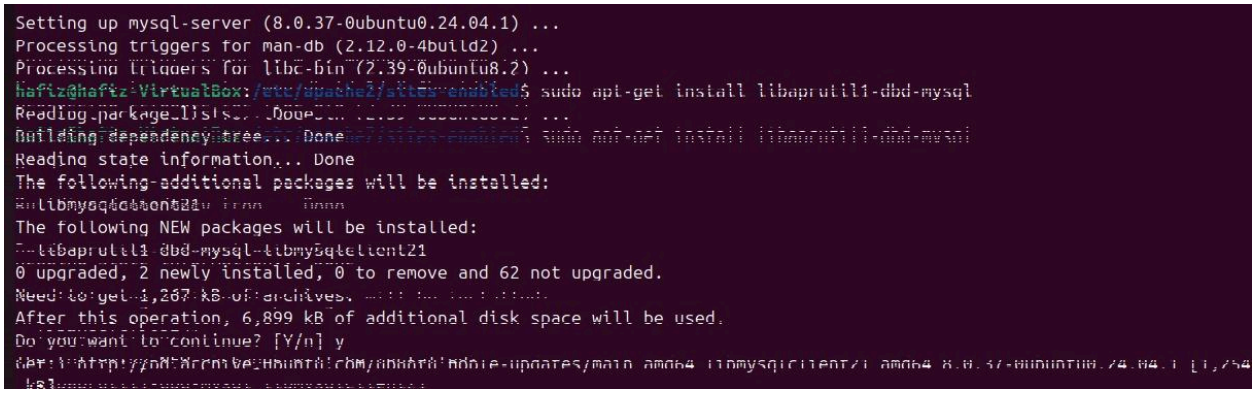
Whenever user gives wrong password:



Task-3: Database Authentication:

Step 1: Install MySQL and Configure

These commands install MySQL server and the necessary Apache modules for database authentication.



Step 2: Secure MySQL Installation

This command secures the MySQL installation by setting a root password and configuring security settings.

```
After this operation, 6,899 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://bd.archive.ubuntu.com/ubuntu noble-updates/main amd64 libmysqlclient21 amd64 8.0.37-0ubuntu0.24.04.1 [1,254
 kB]
Get:2 http://bd.archive.ubuntu.com/ubuntu noble/universe amd64 libaprutil1-dbd-mysql amd64 1.6.3-1.1ubuntu7 [13.4 kB]
Fetched 1,267 kB in 3s (405 kB/s)
Selecting previously unselected package libmysqlclient21:amd64.
(Reading database ... 148956 files and directories currently installed.)
Preparing to unpack .../libmysqlclient21_8.0.37-0ubuntu0.24.04.1_amd64.deb ...
Unpacking libmysqlclient21:amd64 (8.0.37-0ubuntu0.24.04.1) ...
Selecting previously unselected package libaprutil1-dbd-mysql:amd64.
Preparing to unpack .../libaprutil1-dbd-mysql_1.6.3-1.1ubuntu7_amd64.deb ...
Unpacking libaprutil1-dbd-mysql:amd64 (1.6.3-1.1ubuntu7) ...
Setting up libmysqlclient21:amd64 (8.0.37-0ubuntu0.24.04.1) ...
Setting up libaprutil1-dbd-mysql:amd64 (1.6.3-1.1ubuntu7) ...
Processing triggers for libc-bin (2.39-0ubuntu8.2) ...
```

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ 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: no

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.
```

```
Help
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
environment.
```

Step 3: Create Database and Users

These commands create a database and a users table in MySQL.

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.37-0ubuntu0.24.04.1 (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 (0.03 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 (0.12 sec)

mysql>
```

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ htpasswd -bns hafiz hafiz
hafiz:{SHA}DDWd1v0uDlBN4pzdx0gMalPQmbo=

hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$
```

Step 4: Add Users to Database

These commands generate a hashed password and insert a user into the MySQL database.

```
mysql> INSERT INTO users VALUES ('hafiz','{SHA}DDWd1v0uDlBN4pzdx0gMalPQmbo=');
Query OK, 1 row affected (0.03 sec)

mysql>
```

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ htpasswd -bns sadman sadman
sadman:{SHA}bKB7i5OOyzUr6XQxBecsxkrN3lc=
```

Step 5: Configure Apache for Database Authentication

Enable the necessary modules and configure the authentication in Apache:

```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo a2enmod dbd
[sudo] password for hafiz:
Enabling module dbd.
To activate the new configuration, you need to run:
  systemctl restart apache2
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$
```

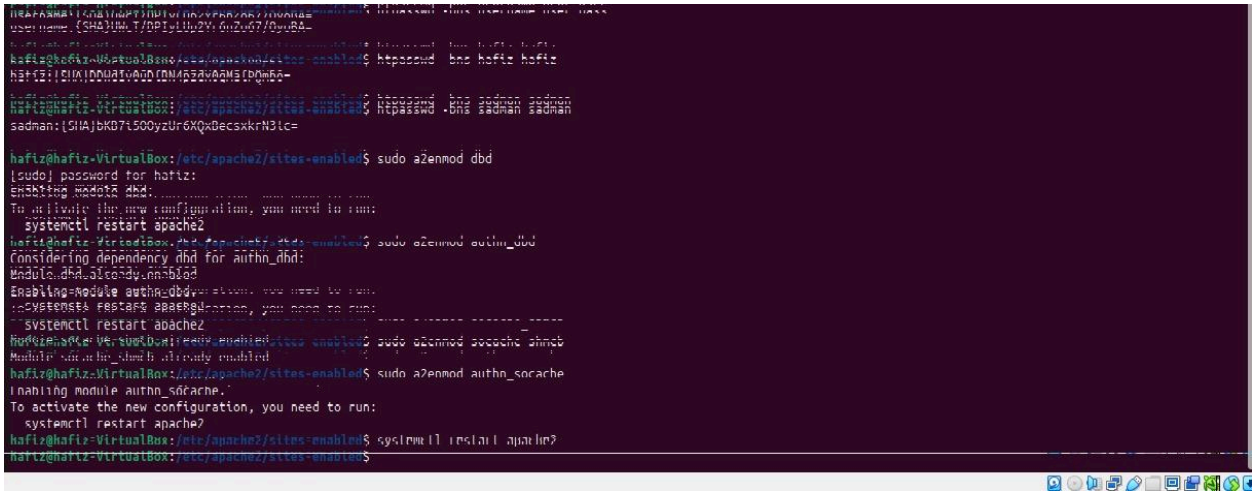
```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo a2enmod authn_dbd
Considering dependency dbd for authn_dbd:
Module dbd already enabled
Enabling module authn_dbd.
To activate the new configuration, you need to run:
  systemctl restart apache2
```

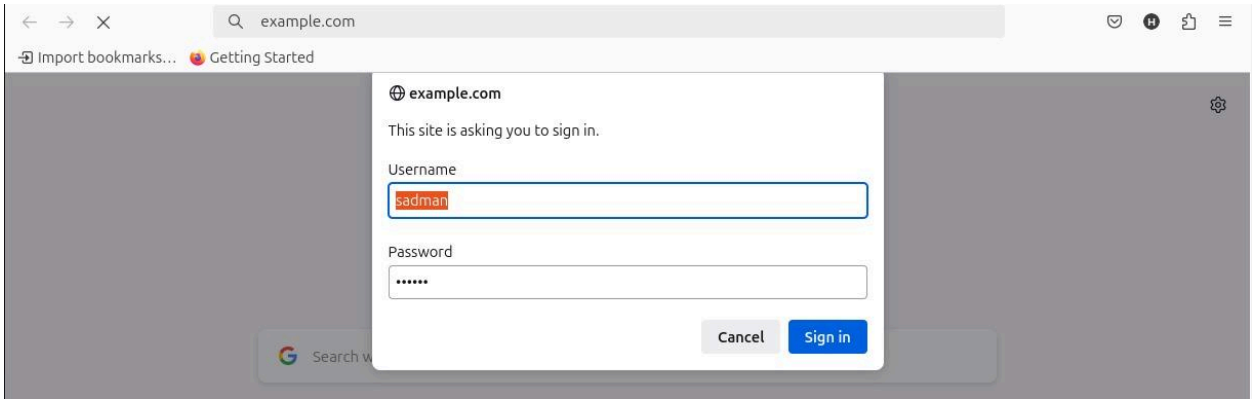
```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo a2enmod socache_shmcb
Module socache_shmcb already enabled
```

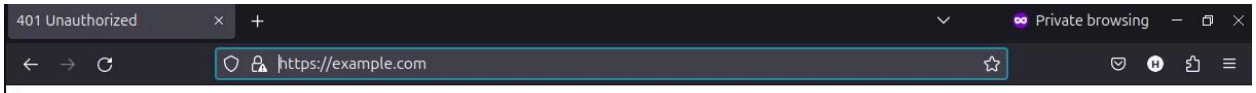
```
hafiz@hafiz-VirtualBox:/etc/apache2/sites-enabled$ sudo a2enmod authn_socache
Enabling module authn_socache.
To activate the new configuration, you need to run:
  systemctl restart apache2
```

Step 6: Restart Apache and Verify Database Authentication
These configurations enable database authentication in Apache.sd

Access `https://example.com` in the browser, enter the username and password from the database, and verify access.







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/2.4.58 (Ubuntu) Server at example.com Port 443