

## 5 Guía autenticación Apache con MySQL

### MATERIAL

- Los contenidos de la unidad y esta guía
- Máquina Virtual Ubuntu 22.04 Desktop.
- Virtualbox
- Ordenador con S.O. Windows 10.
- Servidor web Apache2 instalado.
- Navegador para comprobar la realización de la tarea.
- Procesador de textos para elaborar la documentación y los archivos de la tarea.
- Acceso a Internet.

### 5.1 Actividades

En la anterior actividad vimos como restringir el acceso a una parte de la web creando el archivo .htaccess, en esta actividad vamos a mostrar y operar con las credenciales de acceso por medio de MySQL, de manera que será la base de datos quien controle el acceso a los contenidos restringidos. Por lo que debe hacer:

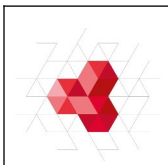
Se pide en un servidor web Apache (apache2):

1. El primer paso es instalar una utilidad para poder integrar MySQL con Apache:

**sudo apt-get install libaprutil1-dbd-mysql**

2. Para realizarlo necesitamos tener instalado el módulo dbd, authn\_dbd, socache\_shmcb, authn\_socache, Comprobamos si está disponible entre los módulos que tiene instalado Apache, por lo que lanzamos el siguiente comando:

**ls /etc/apache2/mods-available/**



```
Informatica@Informatica-VirtualBox:~$ ls /etc/apache2/mods-available/  
access_compat.load  dir.conf  proxy_connect.load  
actions.conf        dir.load  proxy_express.load  
actions.load        dump_io.load  proxy_fcgi.load  
alias.conf          echo.load  proxy_fdpass.load  
alias.load          env.load  proxy_ftp.conf  
allowmethods.load  expires.load  proxy_ftp.load  
asis.load           ext_filter.load  proxy_hcheck.load  
auth_basic.load     file_cache.load  proxy_html.conf  
auth_digest.load    filter.load  proxy_html.load  
auth_form.load       headers.load  proxy_http2.load  
authn_anon.load      heartbeat.load  proxy_http.load  
authn_core.load       heartmonitor.load  proxy.load  
authn_dbd.load        http2.load  proxy_scgi.load  
authn_dbm.load        ident.load  proxy_wstunnel.load  
authn_file.load       imagemap.load  ratelimit.load  
authn_socache.load    include.load  reflector.load  
authnz_fcgi.load      info.conf  remoteip.load  
authnz_ldap.load      info.load  reqtimeout.conf  
authz_core.load       lbmethod_bybusyness.load  reqtimeout.load  
authz_dbd.load        lbmethod_byrequests.load  request.load
```

Vemos que si está en la carpeta de módulos disponibles de Apache. Ahora comprobamos si están habilitados:

**ls /etc/apache2/mods-enabled/**

```
Informatica@Informatica-VirtualBox:~$ ls /etc/apache2/mods-enabled/  
access_compat.load  authz_user.load  mime.conf  reqtimeout.load  
alias.conf          autoindex.conf  mime.load  setenvif.conf  
alias.load          autoindex.load  mpm_prefork.conf  setenvif.load  
auth_basic.load     deflate.conf  mpm_prefork.load  socache_shmcb.load  
authn_core.load     deflate.load  negotiation.conf  ssl.conf  
authn_file.load     dir.conf  negotiation.load  ssl.load  
authz_core.load     dir.load  php7.2.conf  status.conf  
authz_groupfile.load  env.load  php7.2.load  status.load  
authz_host.load     filter.load  reqtimeout.conf
```

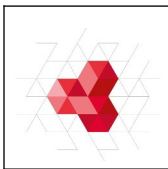
3. Como se puede comprobar en la imagen anterior, vemos que no están habilitados los módulos en la carpeta de Apache de módulos habilitados (/etc/apache2/mods-enabled/). Añadiremos el módulo `authn_socache` para permitir el almacenamiento de las variables de autenticación. Por lo que debemos habilitarlos para poder usarlos:

**a2enmod dbd**

**a2enmod authn\_dbd**

**a2enmod socache\_shmcb**

**a2enmod authn\_socache**



Reiniciamos el servicio apache2 como nos advierten los mensajes al cargar los módulos.

**sudo systemctl restart apache2**

```
informatica@informatica-VirtualBox:~$ sudo systemctl restart apache2
Enter passphrase for SSL/TLS keys for www.empresa-tarea-daw02.local:443 (RSA): *
***
```

4. Ahora tenemos que realizar la configuración del módulo authn\_dbd, creando una base de datos con una tabla para almacenar las credenciales. Teniendo:

- Base de datos=DPL\_credenciales
- Tabla=mysql\_auth

Esto lo hacemos con las siguientes instrucciones:

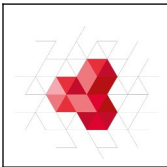
**Creación base de datos DPL\_credenciales**

**mysqladmin -u root -p create DPL\_credenciales**

```
informatica@informatica-VirtualBox:~$ mysqladmin -u root -p create DPL_credenciales
Enter password:
informatica@informatica-VirtualBox:~$
```

**Permisos acceso a la base de datos DPL\_credenciales:**

**sudo mysql -u root -p**



Accedemos a la consola de MySQL y lo primero es cambiar las políticas de seguridad si queremos poner una contraseña de como mínimo 4 caracteres de longitud y una política de fortaleza de contraseñas baja:

```
mysql>SHOW VARIABLES LIKE 'validate_password%'; //Muestra la tabla de política
mysql>SET GLOBAL validate_password_length=4;      //Bajamos longitud password
mysql>SET GLOBAL validate_password_policy=LOW;    //Bajamos a low la política
seguridad

//Ahora ya podemos crear la tabla con un usuario de contraseña de al menos 4
caracteres

mysql> GRANT SELECT, INSERT, UPDATE, DELETE ON DPL_credenciales.* TO
root@localhost IDENTIFIED BY '1234';GRANT SELECT, INSERT, UPDATE, DELETE ON
DPL_credenciales.* TO root@localhost IDENTIFIED BY '1234';

//Vamos a dar permisos para que pueda conectarse a la BBDD desde nuestro dominio

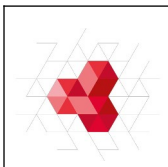
mysql>GRANT SELECT, INSERT, UPDATE, DELETE ON DPL_credenciales.* TO
root@www.dpl.local IDENTIFIED BY '1234';

mysql>FLUSH PRIVILEGES;
```

**Creamos la tabla mysql\_auth:**

```
mysql>USE DPL_credenciales;

mysql>CREATE TABLE mysql_auth(
    -> username varchar(255) not null,
    -> passwd varchar(255),
    -> groups varchar(255),
    -> primary key (username)
    -> );
```



5. Creada la tabla de las credenciales vamos a insertar las credenciales en nuestra tabla. Teniendo en cuenta que la contraseña la insertaremos encriptada para que no se pueda ver en claro si alguien la intercepta o accede a la base de datos, para ello procedemos de la siguiente manera:

Indicar que Apache cuenta con 5 formas de reconocer la autenticación.

#### Basic Authentication

There are five formats that Apache recognizes for basic-authentication passwords. Note that not all formats work on every platform:

##### bcrypt

"\$2y\$" + the result of the crypt\_blowfish algorithm. See the APR source file [crypt\\_blowfish.c](#) for the details of the algorithm.

##### MD5

"\$apr1\$" + the result of an Apache-specific algorithm using an iterated (1,000 times) MD5 digest of various combinations of a random 32-bit salt and the password. See the APR source file [apr\\_md5.c](#) for the details of the algorithm.

##### SHA1

"{SHA}" + Base64-encoded SHA-1 digest of the password. Insecure.

##### CRYPT

Unix only. Uses the traditional Unix crypt(3) function with a randomly-generated 32-bit salt (only 12 bits used) and the first 8 characters of the password. Insecure.

##### PLAIN TEXT (i.e. unencrypted)

Windows & Netware only. Insecure.

En la página de Apache se puede ampliar información de como encriptar la contraseña no sólo desde terminal sino con diferentes lenguajes en el siguiente enlace:

[https://httpd.apache.org/docs/2.4/misc/password\\_encryptions.html](https://httpd.apache.org/docs/2.4/misc/password_encryptions.html)

Antes de insertar la contraseña la debemos encriptar para que la reconozca el sistema de autenticaciones de Apache con los módulos correspondientes. Para ello desde el prompt de la terminal ejecutamos el siguiente comando de manera general.

**htpasswd -nbs usuario contraseñas**

En el caso que nos ocupa lo haríamos de la siguiente manera:

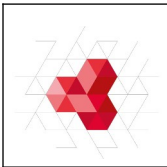
**htpasswd -nbs daw02 daw02**

Con lo que obtendríamos el código hash como se muestra en la siguiente imagen:

```
informatica@informatica-VirtualBox:~$ htpasswd -nbs daw02 daw02
daw02:{SHA}WsA3r3qmhJNKfCPgtsp8D/6ukdw=
```

El código que debemos introducir en el campo passwd es:

**{SHA}WsA3r3qmhJNKfCPgtsp8D/6ukdw=**



```
mysql>INSERT INTO mysql_auth VALUES('daw02','{SHA}WsA3r3qmhJNKfCPgtsp8D/  
6ukdw=', 'admin');  
mysql>QUIT
```

En el caso de que tuviésemos otro valor en el campo passwd procedemos a su actualización mediante la siguiente orden en la consola de MySQL:

```
mysql>UPDATE DPL_credenciales.mysql_auth SET passwd='{SHA}WsA3r3qmhJNK-  
fCPgtsp8D/6ukdw=' WHERE username='daw02';  
mysql>QUIT
```

6. Ahora vamos a configurar el virtualhost para que tome el acceso desde la base de datos MySQL. Para no perder el trabajo realizado en la anterior actividad crearemos un nuevo sitio virtual.

```
sudo mkdir /var/www/dpl
```

Creamos una carpeta con acceso restringido dentro de nuestro sitio:

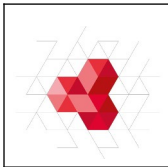
```
sudo mkdir /var/www/dpl/limitado
```

Vamos a dar permisos al Apache para que pueda acceder a la carpeta limitada de nuestro virtualhost:

```
sudo chown -R www-data:www-data /var/www/dpl/limitado/
```

Abrimos el editor nano para configurar el archivo de virtualhost con la conexión a la base de datos MySQL para la autenticación.

```
sudo nano /etc/apache2/sites-available/dpl.local.conf
```



Dentro del sitio virtual editamos lo siguiente:

```
<VirtualHost *:80>
    DocumentRoot /var/www/dpl/
    ServerName www.dpl.local
    ServerAlias dpl.local

    #mod_dbd configuration
    DBDriver mysql
    DBDParams "dbname=DPL_credenciales user=root pass=1234"

    DBDMin 4
    DBDKeep 8
    DBDMax 20
    DBDExptime 300
</VirtualHost>
```

Introducimos en **/etc/apache2/apache2.conf** las siguientes líneas donde se encuentran las líneas directivas de Directory:

```
<Directory "/var/www/dpl/limitado">
# mod_authn_core and mod_auth_basic configuration
# for mod_authn_dbd
AuthType Basic
AuthName "Servidor DPL"

# To cache credentials, put socache ahead of dbd here
AuthBasicProvider socache dbd

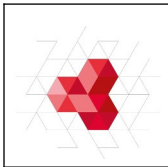
# Also required for caching: tell the cache to cache dbd lookups!
AuthnCacheProvideFor dbd
AuthnCacheContext servidor_DPL

# mod_authz_core configuration
Require valid-user

# mod_authn_dbd SQL query to authenticate a user
AuthDBDUserPWQuery "SELECT passwd FROM mysql_auth WHERE username =
%S"
</Directory>
```

Registramos nuestro dominio en /etc/hosts para que asocie nuestra máquina local al dominio [www.dpl.local](http://www.dpl.local)

**sudo nano /etc/hosts**



Añadimos:

127.0.0.1	localhost	www.empresa-tarea-daw02.local	www.dpl.local
-----------	-----------	-------------------------------	---------------

Activamos el sitio

**sudo a2ensite dpl.local.conf**

Damos permisos a las carpetas:

**sudo chown -R \$USER:\$USER /var/www/dpl/**

**sudo chmod -R 755 /var/www/**