

Démonstration complète : Authentification avec un SGBD (MySQL) dans Apache HTTPD

Cette démonstration vous guide étape par étape pour configurer Apache HTTPD afin d'utiliser un SGBD (MySQL) pour l'authentification des utilisateurs et la gestion des groupes.

Étape 1 : Installer les prérequis

a. Installer Apache et MySQL

Sur Debian/Ubuntu :

```
sudo apt update
sudo apt install apache2 mysql-server libaprutil1-dbd-mysql
```

Sur CentOS/RHEL :

```
sudo yum install httpd mariadb-server apr-util-mysql
```

b. Activer les modules nécessaires dans Apache

Activez les modules `authn_dbd` et `dbd` :

```
sudo a2enmod authn_dbd
sudo a2enmod dbd
sudo systemctl restart apache2
```

Étape 2 : Configurer la base de données MySQL

1. Lancer MySQL

```
sudo systemctl start mysql
```

2. Créer la base de données et les tables Connectez-vous à MySQL en tant qu'administrateur :

```
mysql -u root -p
```

Créez une base de données pour l'authentification :

```
CREATE DATABASE authdb;
USE authdb;

-- Table pour les utilisateurs
CREATE TABLE users (
    username VARCHAR(50) PRIMARY KEY,
    password VARCHAR(255) NOT NULL
);

-- Table pour les groupes
CREATE TABLE user_groups (
    username VARCHAR(50),
    group_name VARCHAR(50),
    PRIMARY KEY (username, group_name)
);
```

3. **Ajouter des utilisateurs et des groupes** Ajoutez des utilisateurs avec des mots de passe hachés (par exemple, avec MD5) :

```
INSERT INTO users (username, password) VALUES ('john',
MD5('password123'));
INSERT INTO users (username, password) VALUES ('alice',
MD5('alicepass'));

INSERT INTO user_groups (username, group_name) VALUES ('john',
'admin');
INSERT INTO user_groups (username, group_name) VALUES ('alice',
'developer');
```

4. **Créer un utilisateur MySQL pour Apache** Créez un utilisateur avec des permissions limitées pour accéder à cette base :

```
CREATE USER 'auth_user'@'localhost' IDENTIFIED BY 'auth_password';
GRANT SELECT ON authdb.* TO 'auth_user'@'localhost';
FLUSH PRIVILEGES;
```

Étape 3 : Configurer Apache HTTPD

1. **Configurer les paramètres de connexion à MySQL** Créez ou modifiez le fichier `/etc/apache2/conf-available/authn_dbd.conf` :

```
DBDriver mysql
DBDParams "host=localhost dbname=authdb user=auth_user
pass=auth_password"
DBDMin 1
```

```
DBDKeep 2
DBDMax 10
DBDExptime 300
```

Activez la configuration et redémarrez Apache :

```
sudo a2enconf authn_dbd
sudo systemctl restart apache2
```

2. **Configurer un Virtual Host pour l'authentification** Créez un fichier `/etc/apache2/sites-available/secure-db-auth.conf` :

```
<VirtualHost *:443>
    ServerName secure.example.com
    DocumentRoot "/var/www/secure"

    SSLEngine On
    SSLCertificateFile "/etc/ssl/certs/example.com.crt"
    SSLCertificateKeyFile "/etc/ssl/private/example.com.key"

    <Directory "/var/www/secure">
        Options Indexes FollowSymLinks
        AllowOverride None

        # Authentification via SGBD
        AuthType Basic
        AuthName "Database Authentication"
        AuthBasicProvider dbd

        # Requête SQL pour valider les utilisateurs
        AuthDBUserPWQuery "SELECT password FROM users WHERE username
= %s"

        # Requête SQL pour vérifier les groupes
        AuthDBGroupQuery "SELECT group_name FROM user_groups WHERE
username = %s"

        # Autoriser uniquement les utilisateurs du groupe admin
        Require dbd-group admin
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/secure-db-auth-error.log
    CustomLog ${APACHE_LOG_DIR}/secure-db-auth-access.log combined
</VirtualHost>
```

3. **Activer le site et redémarrer Apache**

```
sudo a2ensite secure-db-auth.conf
sudo systemctl reload apache2
```

Étape 4 : Tester la configuration

1. Accéder au site

- Naviguez vers <https://secure.example.com>.
- Une boîte de dialogue s'affichera pour demander un nom d'utilisateur et un mot de passe.

2. Vérifier les permissions

- Connectez-vous avec :
 - Nom d'utilisateur : [john](#)
 - Mot de passe : [password123](#).
- Résultat attendu : Seul [john](#), appartenant au groupe [admin](#), peut accéder au site.

3. Logs Vérifiez les journaux en cas de problème :

```
tail -f /var/log/apache2/secure-db-auth-error.log
```

Étape 5 : Extensions possibles

a. Autoriser plusieurs groupes

Pour autoriser plusieurs groupes, modifiez la directive [Require](#) :

```
<RequireAny>
  Require dbd-group admin
  Require dbd-group developer
</RequireAny>
```

b. Restreindre l'accès à certains utilisateurs

Pour autoriser uniquement des utilisateurs spécifiques :

```
Require dbd-user john alice
```

c. Utiliser une autre méthode de hachage

Si vous utilisez [SHA256](#) au lieu de [MD5](#) :

```
INSERT INTO users (username, password) VALUES ('mike', SHA2('securepass', 256));
```

Et configurez Apache pour utiliser SHA256.

Résumé des fichiers

Fichier	Description
/etc/apache2/conf-available/authn_dbd.conf	Paramètres de connexion à la base de données.
/etc/apache2/sites-available/secure-db-auth.conf	Configuration du Virtual Host sécurisé.
authdb (base MySQL)	Stocke les utilisateurs et groupes pour l'authentification.