

Actividad de programación en servidor

Vamos a trabajar con una funcionalidad muy común en cualquier sitio web como es la autenticación de usuarios (login). La autenticación es necesaria para comprobar que la persona que está accediendo es quien dice ser. Para ello, el usuario que accede debe presentar unas credenciales , las cuales deberán ser comprobadas por el sistema.

La contraseña es el elemento fundamental en las credenciales (es algo que en principio sólo el usuario sabe). **Aunque este es el nivel de seguridad más bajo**, comenzaremos a trabajar con él.

Objetivos de la práctica:

- Trabajar con formularios HTML para enviar datos al servidor.
- **Usar funciones de cifrado de contraseñas en PHP (password_hash y password_verify).**
- Comparar contraseñas cifradas para la autenticación de usuarios. Para ello el programa debe implementar una función que cifre las contraseñas (no se guardarán en claro nunca)

Instrucciones generales sobre la actividad:

1. **Se necesita un formulario de entrada:** El usuario debe ingresar su nombre de usuario y contraseña.
2. **Cifrado de contraseña:** Al enviar los datos, el servidor compara la contraseña introducida cifrándola y verificándola con la contraseña almacenada en el array asociativo. La contraseña inicialmente deberá almacenarse en una array en el servidor

Paso 1: Estructura básica del formulario HTML

Crea un archivo llamado index.php con el siguiente contenido:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Autenticación de Usuario</title>
</head>
```

```

<body>
    <h2>Iniciar sesión</h2>
    <form action="autenticar.php" method="POST">
        <label for="username">Usuario:</label>
        <input type="text" name="username" id="username"
required><br><br>

        <label for="password">Contraseña:</label>
        <input type="password" name="password" id="password"
required><br><br>

        <button type="submit">Iniciar sesión</button>
    </form>
</body>
</html>

```

NOTA: Fíjate en el método del formulario y cual es el recurso que lo va a procesar

Paso 2: Archivo autenticar.php para procesar el formulario

En este archivo php se maneja la autenticación. Se utilizará un array asociativo con usuarios y contraseñas cifradas.

NOTA: Añade al menos tres o cuatro usuarios para que simule la base de datos de usuarios

```
<?php
```

```

// Array asociativo con usuarios y contraseñas cifradas
$users = [
    'user1' => password_hash('pd123', PASSWORD_DEFAULT),
    'user2' => password_hash('my5678', PASSWORD_DEFAULT),
    'admin' => password_hash('adminpass', PASSWORD_DEFAULT),
];

```

```
// Obtener los datos del formulario
```

Aquí debes obtener los datos del formulario. Recuerda que se pasan por el método POST. Verifica en el formulario html los nombre de las variables que tienen cada uno de los datos pasados y úsalos para recuperar su valor

```
// Verificar si el usuario existe en el array
```

Usa la function `array_key_exists()` para comprobar si el usuario está dado de alta en la base de datos

Usa la función `password_verify()` de php para verificar la password pasada por el formulario

Imprime en la salida el mensaje "Usuario autenticado correctamente" o "Credenciales erróneas" en otro caso

Cuestiones:

1. Imprime el array `$usuarios` para ver como las contraseñas se han guardado cifradas
2. Si esa función se usa cuando el usuario se registra por primera vez y los datos se cifran por el canal de comunicación ¿ Crees que alguien puede ver la contraseña en claro alguna vez? ¿ Y el administrador?
3. ¿Qué mejoras se pueden proponer o que ya implementan los sistemas actuales para dotar la autenticación de un mayor nivel de seguridad?

