Documentación: Inicio de Sesión y SQL Injection

Este documento explica el funcionamiento de un inicio de sesión en PHP orientado a objetos, mostrando dos casos: cuando el sistema es vulnerable a SQL Injection y cuando se protege utilizando consultas preparadas.

1. Inicio de sesión SIN consultas preparadas (Vulnerable)

En este caso, los datos introducidos por el usuario se insertan directamente en la consulta SQL. Esto permite que un atacante modifique la consulta y acceda sin credenciales válidas.

```
$sql = "SELECT nombre
FROM usuario WHERE nombre = '$nombre'
AND contrasena = '$contrasena'";

$resultado = $this->conexion->query($sql);
```

Si un atacante introduce algo como ' OR '1'='1 en los campos, la consulta se altera y puede devolver resultados sin verificar correctamente al usuario.

2. Inicio de sesión CON consultas preparadas (Seguro)

Aquí se utilizan consultas preparadas. La consulta se define primero y los datos se envían por separado, evitando que el usuario pueda modificar la estructura SQL.

```
$sql = "SELECT nombre
FROM usuario
WHERE nombre = ?
AND contrasena = ?";

$stmt = $this->conexion->prepare($sql);

 if($stmt) {
    $stmt->bind_param("ss", $nombre, $contrasena);
    $stmt->execute();
}
```

En este caso, aunque el usuario intente introducir código SQL malicioso, lo tratará únicamente como datos, impidiendo la inyección de SQL. Conclusión: El uso de consultas preparadas es fundamental para la seguridad de aplicaciones web, ya que protege contra uno de los ataques más comunes: la SQL Injection.