实验吧解题笔记——编程（五）

原创

FunkyPants 于 2017-10-18 17:51:17 发布 575 收藏

分类专栏： CTF writeup 文章标签： python 编程

本文链接：https://blog.csdn.net/FunkyPants/article/details/78275717

0.说明

每五个题目写作一篇writeup，第一行对应解题笔记（一）……

1.字典

题目描述：

包含ctf的单词的总字符有多少？
字典文件格式如下

ascii_shiyanba.py × dictionary.txt ×

The file size (6.47M) exceeds configured limit

```
46  aani
47  aappmedic
48  aara
49  aarai
50  aaran
51  aardvark
52  aardvarks
53  aardwolf
54  aaren
55  aargau
56  aargh
57  aarhus
58  aari
59  aarika
60  aariya
61  aarnet
62  aaron
63  aaronic
64  aaronical
65  aaronite
66  aaronites
67  aaronitic
68  aarons
69  aaronsburg
70  aaronson
71  aarthur
72  aartjan
73  aaru
74  aaryn
75  aase
76  aasen
77  aastveit
```


分析

这个题目使用Python去按行读取文件，同时使用Python内置的index()方法查找字符串（通常来说Python内置的算法都经过了算法优化，运行效率高于自己编写的），如果再字符串中查找到”ctf”字符，记录其字符个数。这里有一个坑就是按行读取文件时会把换行符\n也读取进去，需要用replace方法替换再计数，代码如下：

```
re_str = 'ctf'
num = 0
with open('dictionary.txt', 'r') as f:
    while True:
        line = f.readline()
        #print(type(line))
        if not line:
            break
        try:
            if line.index(re_str) >= 0:
                num += len(line.replace('\n', ''))
                print(len(line.replace('\n', '')))
                print(line, end='')
        except:
            pass
print('CTF{' + str(num) + '}')
```

3.速度爆破

题目描述

给你一个sha1值，它是0-100000之间的整数的md5值再求取sha1值，请在2秒内提交该整数值

请在2秒内提交该整数： 提交

4c88313d10a276e754dbda03895715378c266c8a

Wrong parameter!

分析

可以看到，这个题目和“百米”属于同一类型的题目，需要编写程序自动完成抓取、计算、提交的步骤。抓取和提交的步骤在“百米”的题目中已经讲过了，这里略过，主要讲一下如何使用python计算sha1和md5值。

python内置的有一个叫hashlilb的库，里面实现了很多与密码学有关的算法，使用时十分便捷。
以md5的计算过程为例（计算sha1的方法相同），首先初始化一个hashlib.md5的对象，将以utf-8编码的字符串作为参数传入，可以得到计算结果，然后我们还需要将其转换成十六进制的形式打印出来才能便于我们阅读，代码如下：

```
data = ‘test’.encode(‘utf-8’)
hash_md5 = hashlib.md5(data)
print(hash_md5.hexdigest())
```

最终解题的代码如下：

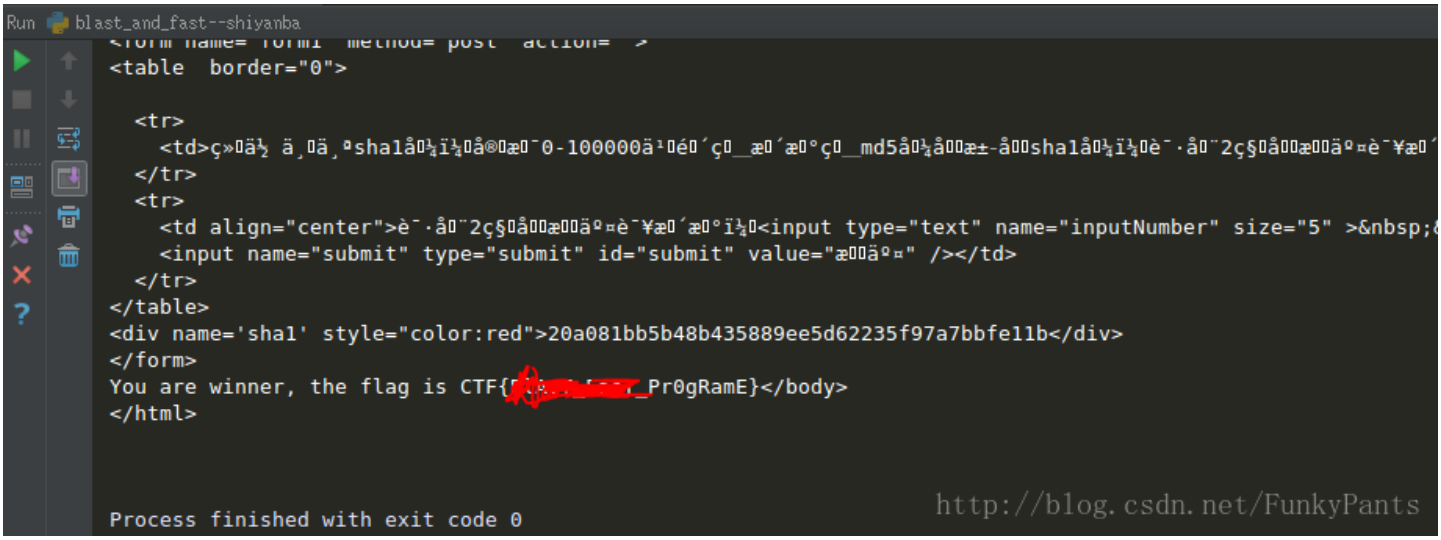
```
import hashlib
import requests
from bs4 import BeautifulSoup

get_url = 'http://ctf5.shiyanbar.com/ppc/sd.php'

session = requests.session()
html = session.get(get_url).content
soup = BeautifulSoup(html, 'lxml')
encode_str = soup.div.get_text()

for i in range(0, 100000 + 1):
    hash_md5 = hashlib.md5(str(i).encode('utf-8')).hexdigest()
    hash_sha1 = hashlib.sha1(hash_md5.encode('utf-8')).hexdigest()
    if encode_str == hash_sha1:
        #提交
        payload = {'inputNumber':i}
        post =session.post(get_url, payload)
        print(post.text)
```

FLAG：


Run blast_and_fast--shiyanba
<table border="0">
<tr>
<td>ç»ä½ ä¸ªsha1å¤ï¼å®æ¯0-100000ä¹éï¼ç_æ´æ°ç_md5å¼åæ±-åshaå¤ï¼è¯·å¨2ç§åæäº¤è¯¥æ´
</tr>
<tr>
<td align="center">è¯·å¨2ç§åæäº¤è¯¥æ´æ°ï¼<input type="text" name="inputNumber" size="5" >
<input name="submit" type="submit" id="submit" value="æäº¤" /></td>
</tr>
</table>
20a081bb5b48b435889ee5d62235f97a7bbfe11b
</form>
You are winner, the flag is CTF{[redacted]_Pr0gRamE}</body>
</html>
Process finished with exit code 0
http://blog.csdn.net/FunkyPants