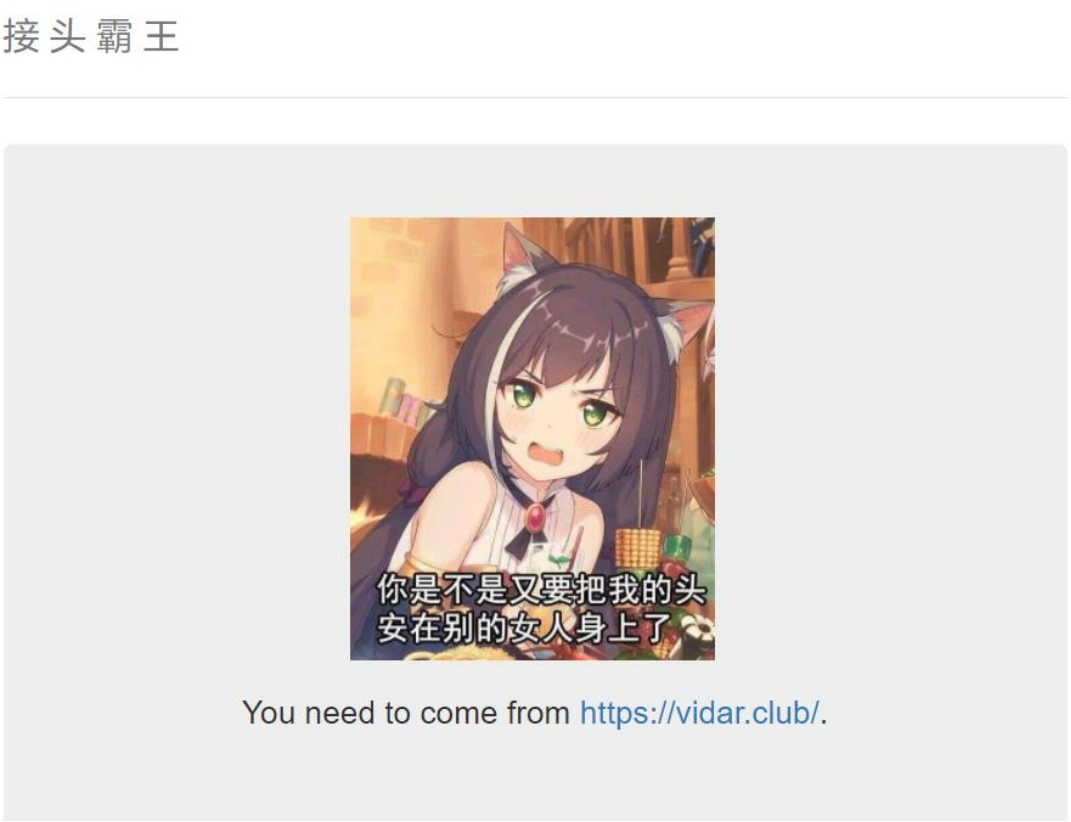
Web：

接 头 霸 王:



网页中有一串字符串：需要来自这个网址，又根据题目接头，应该是跟响应头有关，利用 burpsuite 连接网页。

```
GET / HTTP/1.1
Host: kyaru.hgame.n3ko.co
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:72.0) Gecko/20100101 Firefox/72.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 127.0.0.1
Referer: https://vidar.club/
```

在响应头最下方加一个 Referer

接头霸王

You need to use Cosmos Brower to visit.

图片下的内容变成了需要用 Cosmos Brower 来访问，于是再修改一个 User-Agent

```
GET / HTTP/1.1
Host: kyaru.hgame.n3ko.co
User-Agent: Cosmos Brower/5.0 (Windows NT 10.0; Win64; x64; rv:72.0) Gecko/20100101 Firefox/72.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 127.0.0.1
Referer: https://vidar.club/
```

接头霸王

Your should use POST method :)

又要我们改为 post 请求，真是怎样都满足不了呢。

接头霸王

The flag will be updated after 2077, please wait for it patiently.

改成 post 后又是让我们改时间到 2077，最后修利用 If-Unmodified-Since 改一下时间，获得 flag。

hgame{W0w!Your_heads_@re_s0_many!}

Code World：
打开网页后发现是 403 界面

```
        console.log("This new site is building....But our stupid developer Cosmos did 302
jump to this page..F**k!")
```

F12 一下发现有 302 跳转，利用 burpsuite 将 get 请求改为 post 请求。

人鸡验证

目前它只支持通过url提交参数来计算两个数的相加，参数为a

现在,需要让结果为10

题目中表明只支持 url 提交参数，且是两个数相加，猜测之所以这样应该是把‘+’屏蔽了。通过测试可知确实将‘+’屏蔽了，那就利用 url 编码将‘+’号转码，最终得到 flag

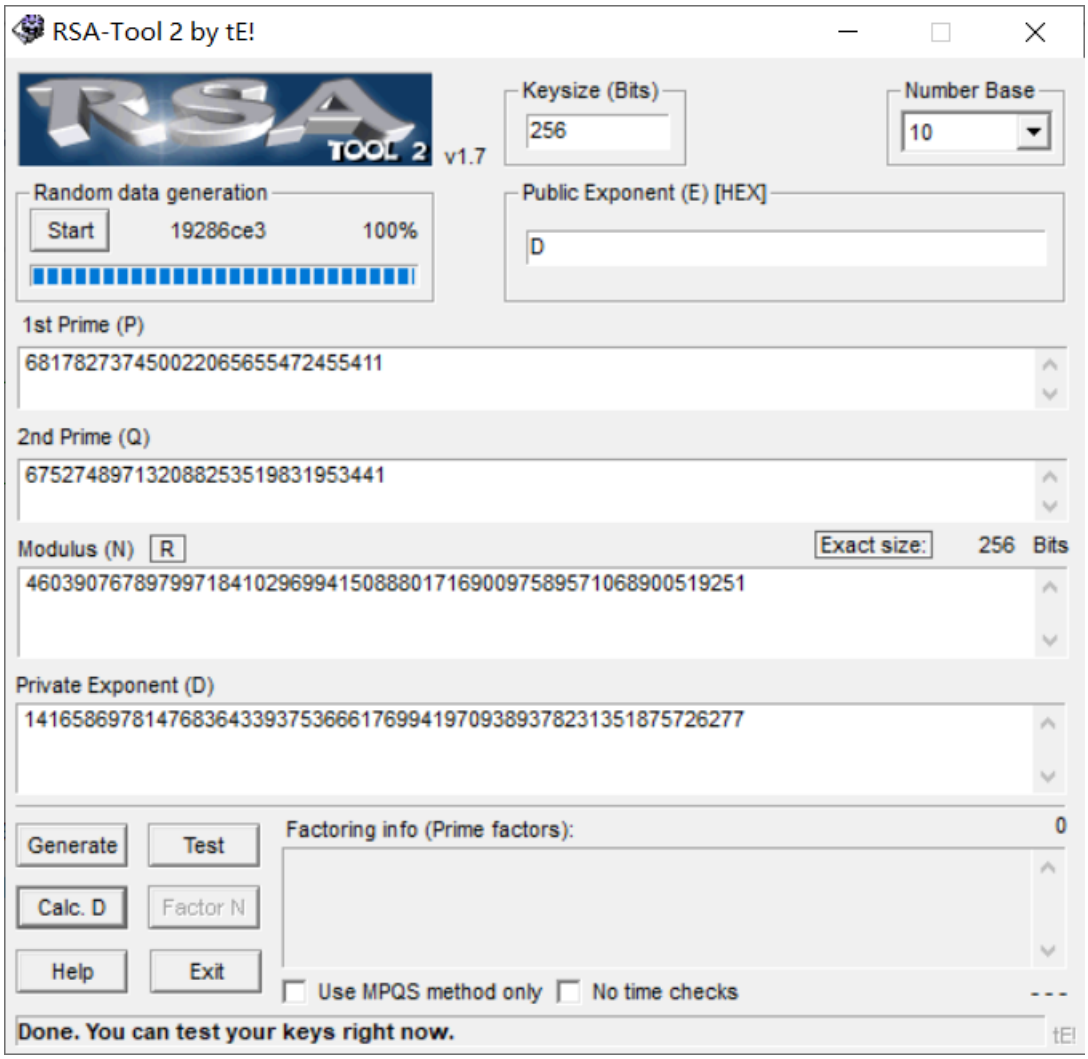
```
POST /?a=5%2b5 HTTP/1.1
Host: codeworld.hgame.day-day.work
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:72.0) Gecko/20100101 Firefox/72.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,im
age/webp,*/*;q=0.8
Accept-Language:
zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q
=0.2
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 0
Origin: http://codeworld.hgame.day-day.work
Connection: keep-alive
Referer: http://codeworld.hgame.day-day.work/?a=5%2b5
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 127.0.0.1
```

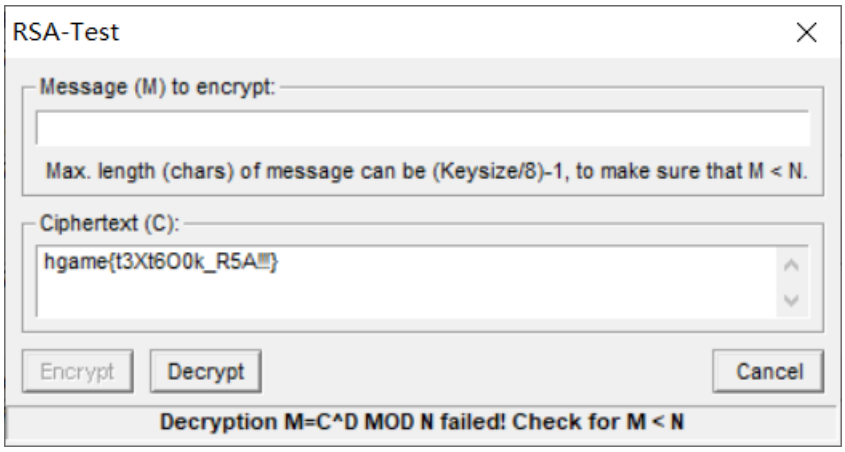
```
HTTP/1.1 200 OK
Server: nginx/1.14.0 (Ubuntu)
Date: Sat, 18 Jan 2020 07:35:39 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 224
Connection: keep-alive
Location: new.php
Vary: Accept-Encoding

<center><h1>□□□□</h1><br><br>□□□□□□□□url□□□□□□□□□□□□□□□
□□a<br><br>□□,□□□□□□10<br><h1>The result is:
10</h1><br>hgame{C0d3_1s_s0_S@_s0_C0ol!}</center>
```

Crypto

InfantRSA：

直接利用工具求出 d，然后利用 c 求出 flag





Affine:

根据代码可得 flag 的格式为 hgame{}, 关键是要求出 A 、B 两个未知数, MOD 为 len(TABLE)

可知 MOD = 62，且将 h 和 g 分别代入代码后答案为 A 和 8，可求出 A = 13 B =14。

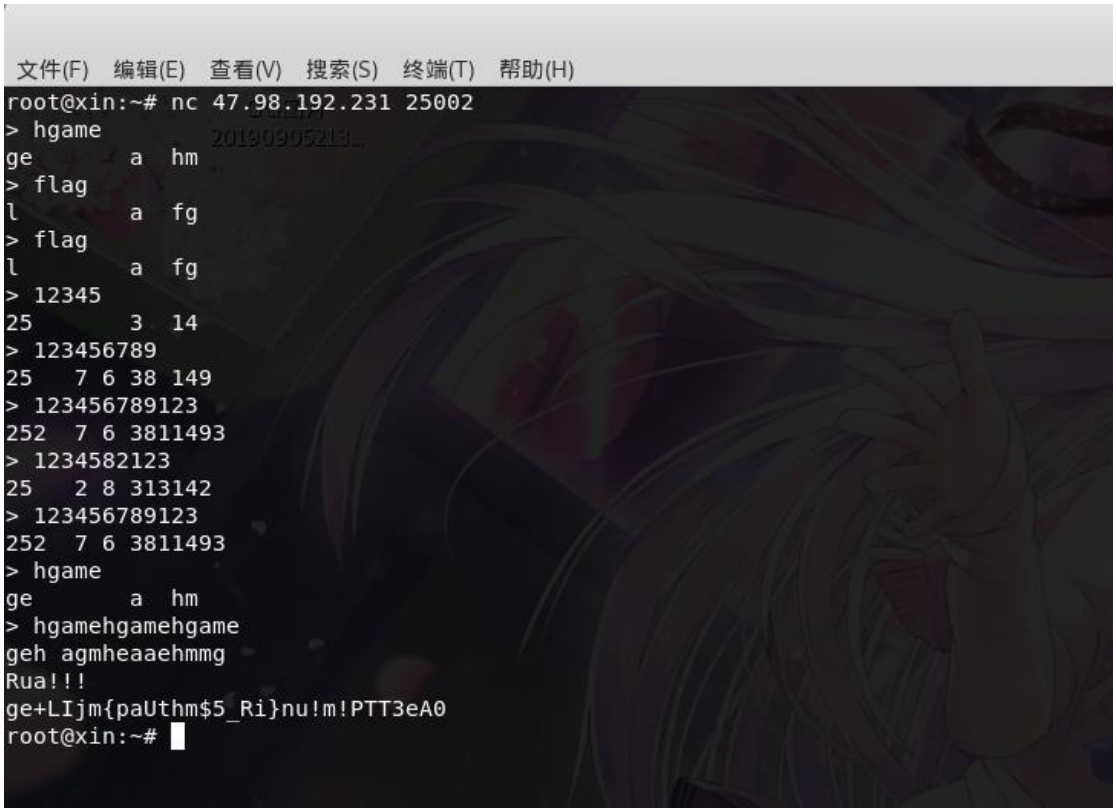
然后就可以写一个程序直接把答案跑出来了。

```python
a = input()
TABLE = 'zxcvbnmasdfghjklqwertyuiop1234567890QWERTYUIOPASDFGHJKLZXCVBNM'
flag = ''
for i in a:
    b = TABLE.find(i)
    if b == -1:
        flag += i
    else:
        c = (b - 14) / 13
        n = 1
        while(1):
            if int(c) == c:
                flag += TABLE[int(c)]
                break
            c = (b - 14 + 62*n) /13
            n += 1
print(flag)
```

```
A8I5z{xr1A_J7ha_vG_TpH410}
hgame{M4th_u5Ed_iN_cRYpt0}
```

Reorder:

一进去就是让我们瞎打一些字符，且打完之后会排序，中间还有许多空格。打了好几个发现突然出现了一个类似 flag 格式的字符串。通过观察发现 hgame{}均存在，盲猜栅栏密码。经过很多次测试之后发现并没有什么卵用。然后重新连进去又打了一遍发现排列顺序变了，最后的 flag 字符串也变了。



然后我就在想为什么前面我输入的字符串排序会出现空格，而后面的 flag 字符串却没有空格。后来发现可能是我字符串输入长度和 flag 字符串不一样，导致前面的排序将空格也算在内了。然后我就打了个和 flag 字符串相同长度的字符串。

发现排序内确实没有空格了，最后的 flag 应该就是按照这个加密规则加密的，且加密是随机的，每次 nc 链接后的加密顺序都不一样。最后写一个程序跑一下。

```python
a = 'abcdefghijklmnopqrstuvwxyz123456'
b = 'ljmkbdechfapogni2z31rtusxvq65w4y'
flag1 = '5tI+gmeaU{hLpjm$0Tni_eRPTm3}!u!A'
flag = ''
for i in a:
    c = b.find(i)
    flag += flag1[c]
print(flag)
```

最终可得答案：

```
hgame{jU$t+5ImpL3_PeRmuTATi0n!!}
```

Misc：

欢迎参加 HGame：
将题目中的字符串通过 base64 解密可获得一串摩斯电码，然后解密可得 flag

壁纸：
下载压缩包可得压缩包内有一个图片，通过 foremost 分离可以分离出一个加密的压缩包，压缩包内有个 flag.txt 很明显就是 flag 所在。注释中写着密码为图片 ID。用记事本打开图片看看有没有 id

```
:?xpacket begin="锘? id="W5M0MpCehiHzreSzNTczkc9d"?> <x:xmpr
```

在记事本中还真找到了个 id。输入密码试试看，可是怎么试都是错误的。然后我就卡在这好久。之后重新思考了一下，图片 id 可能是文件名，我看了下文件名是 p 站某作品。

之后想着线上 p 站看看这幅画

还真找到了这幅画的 ID，输入 id 发现确实是正确的密码。（这题题目建议改成“如何上 P 站”）

> \u68\u67\u61\u6d\u65\u7b\u44\u6f\u5f\u79\u30\u75\u5f\u4b\u6e\u4f\u57\u5f\u75\u4e\u69\u43\u30\u64\u33\u3f\u7d

打开记事本可以发现一串编码，很明显为 asc 编码，利用在线工具解密。
最终可得 flag

> hgame{Do_y0u_KnOW_uNiC0d3?}

克苏鲁神话：
下载压缩包，得到压缩包里面有一个 Bacon.txt 和另一个压缩包，另一个压缩包内还有一个 Bacon.txt，可以想到为明文攻击。利用 ARCHPR 进行明文攻击，可以得到一个破解密码后的压缩文件。压缩文件内有一个加密的 word 文档和一个 Bacon.txt。用记事本打开 word 在最后可以发现一个字符串。
Bacon is tasty
Bacon.txt 有一串字符串，中间既有大写又有小写。下面还有一个提示密码为大写字母。一开始我一直 Password in capital letters.以为的意思是密码在大写字母里，然后将上面的字符串中的大写字母分离出来寻找关系，可一直找不到有什么关系。

> of SuCh GrEAt powers OR beiNGS tHere may BE conCEivAbly A SuRvIval oF HuGely REmOTE periOd.
>
> *Password in capital letters.

然后通过各种地方都出现 bacon 字符，可想这个单词肯定是一个提示，然后发现可能是培根密码。但培根密码只会出现 AB 两个字母。经过各种尝试最终得出了，将小写字母写成 A，大写字母写成 B，可以解出 word 密码。

```
flaghiddenandoc
```

将密码输入后可以看到 word 文档内有一段类似是故事中的一段剧情，全文观察后没有和 flag 有关的东西。利用记事本打开搜索 hgame 也是未找到。由于 word 文档加密前和加密后记事本内的内容不一样，所以先将 word 保存为未加密状态，可还是未找到。
最后用 winhex 打开慢慢找

```
+R Q⊧i,{斉蘱<w[w
 wOR儆 0  h g a
m e { Y 0 u _ h
@ V e _ F 0 U n
d _ m Y _ S 3 c
R e T }
```

最终在中间发现了 flag，原来是在 hgame 中加了空格，这个真的是太坑了。

签到题 ProPlus：

下载文件后可获得一个压缩包，压缩包内有个 password.txt 和一个加密的 OK.ZIP，很显然是要解密。

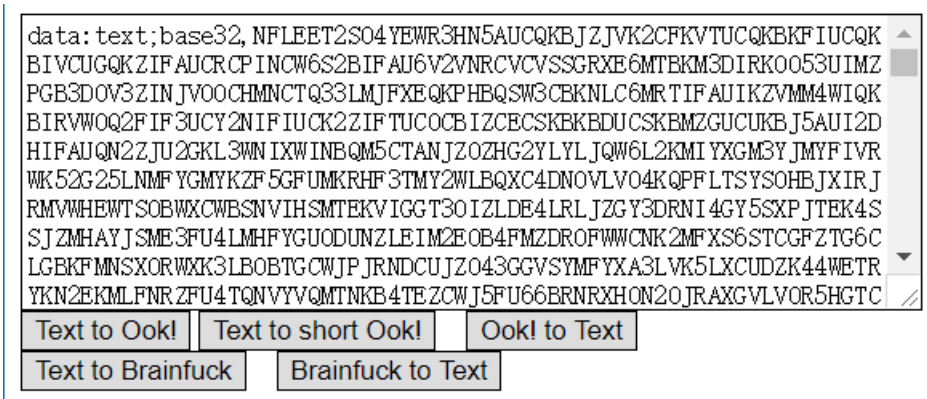
Rdjxfwxjfimkn z,ts wntzi xtjrwm xsfjt jm ywt rtntwhf f y h jnsxf qjFjf jnb rg fiyykwtbsnkm tm xa jsdwqjfmkjy wlviHtqzqsGsffywjjyyn

JRFVJYFZVRUAGMAI

* Three fenses first, Five Caesar next. English sentense first, zip password next.

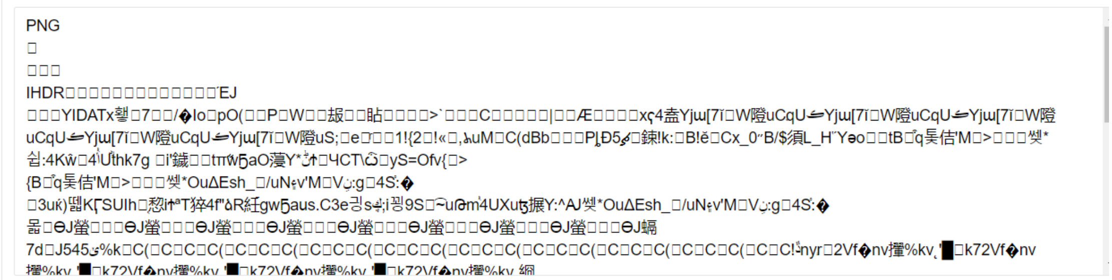
将上面一串英文利用 3 栅栏，5 凯撒的方式解密可获得一句话，但没有什么大用处。将下面的 16 位字符串通过同样方法解密，可获得一个解密后的字符串

输入后果真解开了 ok.zip 的密码，ok.zip 中有一个 ok.txt 文件，里面全是 ok，可想而知大概率是 ok 密码。利用在线工具破解，可得

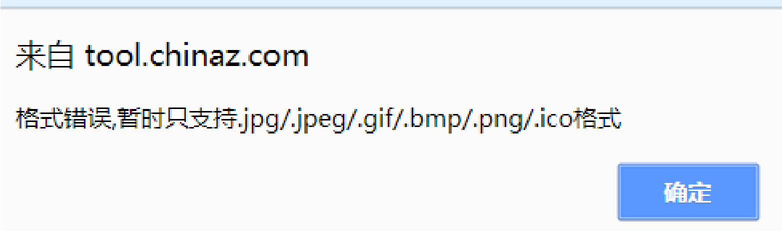


是一个 base32 位密码，利用 base32 解密后可获得另一个字符串

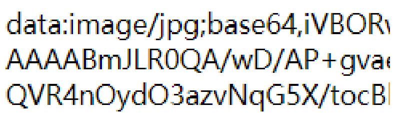
因为有+和/存在，可想是 base64 编码，再利用 base64 解密一次。



发现是一个 png 图片，那就直接利用 base64 转图片在线转化一下。



可是一直显示格式错误，但根据 base64 解码来看肯定是 png 格式，就想可能写密文时有格式要求。于是随便将一张图片转化成 base64 看看格式是什么



发现需要加上一些格式要求。

加上后进行解密

发现转化成了一个二维码，扫描后即可获得 flag