简单的rsa题目

```
v10 = __readfsqword(0x28u);
puts("[sign in]");
printf("[input your flag]: ");
__isoc99_scanf("%99s", v8);
sub_96A(v8, (__int64)v9);                       // 输入转换成16进制
__gmpz_init_set_str((__int64)v7, (__int64)"ad939ff59f6e70bcbfad406f2494993757eee98b91bc244184a377520d06fc35", 16LL);
__gmpz_init_set_str((__int64)v6, (__int64)v9, 16LL);
__gmpz_init_set_str(
  (__int64)v4,
  (__int64)"10346103590081691412139010129904904441395040517371217043416168653987816098454 9",
  10LL);
__gmpz_init_set_str((__int64)v5, (__int64)"65537", 10LL);
__gmpz_powm(v6, v6, v5, v4);
if ( (unsigned int)__gmpz_cmp(v6, v7) )
  puts("GG!");
else
  puts("TTTTTTTTTql!");
return 0LL;
```

搜一下rsa的基本概念和gmpy库的基本写法就能得到答案

```
from gmpy2 import *
from Crypto.Util.number import long_to_bytes
n = 10346103590081691412139010129904904441395040517371217043416168653987816098454 9
p = 28216458745951212484424511395059334827 1
q = 3666691020029668568766056698370142294 19
c = 0xad939ff59f6e70bcbfad406f2494993757eee98b91bc244184a377520d06fc35
e = 65537
d = invert(e,(p-1)*(q-1))
p = powmod(c,d,n)
print(p)
print(long_to_bytes(p))
```

```
185534734614696481020381637136165435809958101675798337848243069
b'suctf{Pwn_@_hundred_years}'
>>>
```