bit_operation

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  int i; // [rsp+Ch] [rbp-84h]
  char s[104]; // [rsp+10h] [rbp-80h] BYREF
  unsigned __int64 v6; // [rsp+78h] [rbp-18h]

  v6 = __readfsqword(0x28u);
  __isoc99_scanf(&unk_2004, s, envp);
  encode(s);
  for ( i = 0; i < strlen(s); ++i )
  {
    if ( s[i] != b[i] )
    {
      printf("oh no!");
      return 0;
    }
  }
  puts("nice!");
  return 0;
}
```

丢到ida看一下main函数。

先用scanf读入字符串，然后将字符串s用encode()函数加密，最后把s的加密结果和b比较，如果完全相同则返回nice，不相同则返回oh no!

重点在于encode函数

```
__int64 __fastcall encode(const char *a1)
{
  __int16 v1; // kr00_2
  __int64 result; // rax
  unsigned int i; // [rsp+18h] [rbp-8h]
  int v4; // [rsp+1Ch] [rbp-4h]

  v4 = strlen(a1);
  for ( i = 0; ; ++i )
  {
    result = i;
    if ( (int)i >= v4 )
      break;
    v1 = 16 * (unsigned __int8)a1[i];
    a1[i] = HIBYTE(v1) + v1;
  }
  return result;
}
```

将字符串传入，for循环到字符串长度为止。将a1[i]的字符转换为无符号8位int，再*16，相当于左移4位。再取a1[i]的高4位加上之前左移的结果。

英文字符长度八位二进制，例如A： 0100 0001 详见ascii码表

将原符号的ascii值左移四位，相当于取了低四位，往后补四个0。

补完以后再取高1位相加，那实际上就是将高四位和第四位交换。

转换成16进制就能很清楚的看出来：

A： 0x41 左移4位后-> 0x10 加上高四位后-> 0x10+0x4 = 0x14

所以解密也很简单，只要再把每个字符的后四位和前4位交换一下即可。

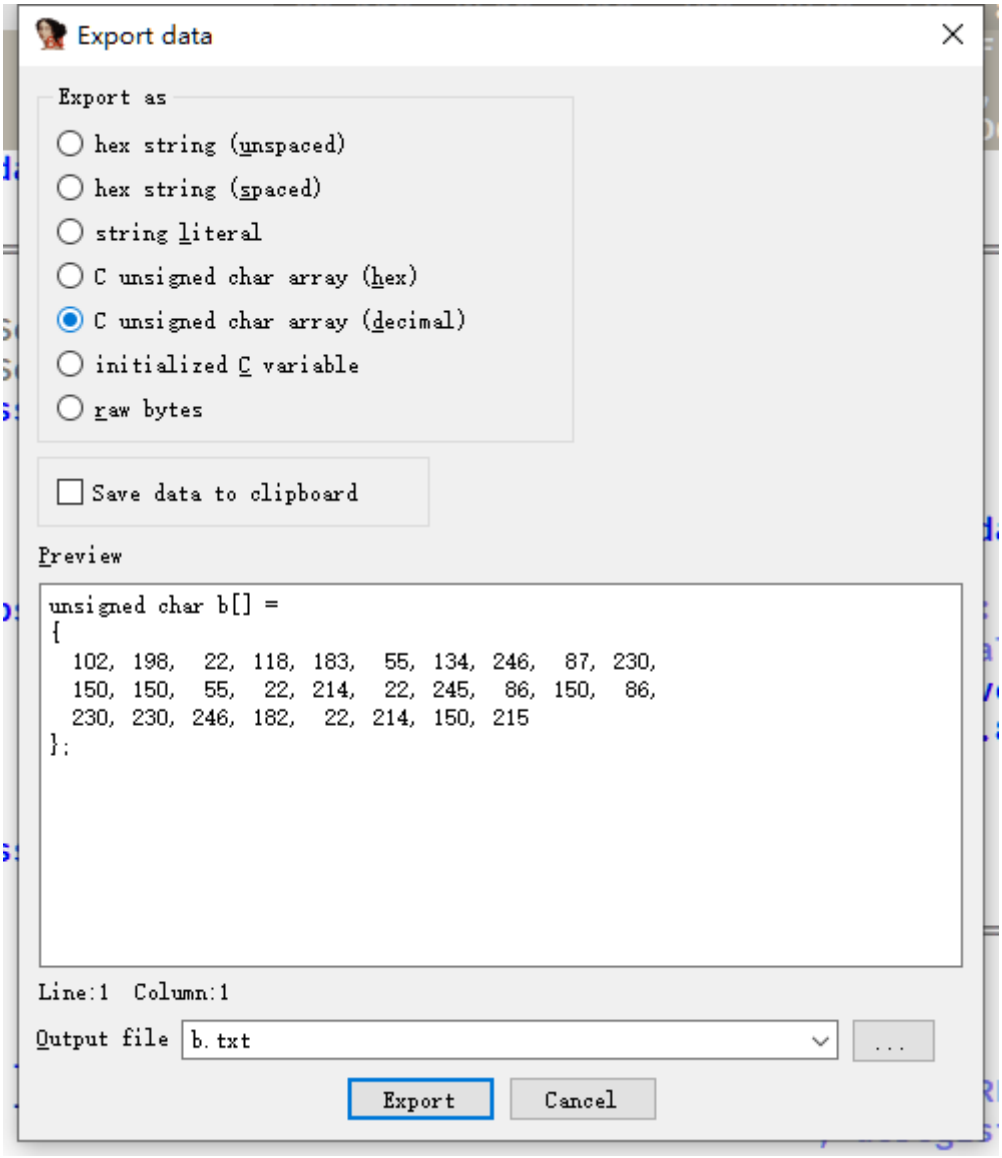
再来看密文

因为程序最后会和b[]数组比较，所以密文肯定存在b数组里：

```
.data:0000000000004010 b               db 66h, 0C6h, 16h, 76h, 0B7h, 37h, 86h, 0F6h, 57h, 0E6h
.data:0000000000004010                                         ; DATA XREF: main+69↑o
.data:0000000000004010                 db 2 dup(96h), 37h, 16h, 0D6h, 16h, 0F5h, 56h, 96h, 56h
.data:0000000000004010                 db 2 dup(0E6h), 0F6h, 0B6h, 16h, 0D6h, 96h, 0D7h
.data:0000000000004010 _data           ends
```

```
.data:0000000000004010 b               db 66h, 0C6h, 16h, 76h, 0B7h, 37h, 86h, 0F6h, 57h, 0E6h
.data:0000000000004010                                         ; DATA XREF: main+69↑o
.data:0000000000004010                 db 2 dup(96h), 37h, 16h, 0D6h, 16h, 0F5h, 56h, 96h, 56h
.data:0000000000004010                 db 2 dup(0E6h), 0F6h, 0B6h, 16h, 0D6h, 96h, 0D7h
.data:0000000000004010 _data           ends
```

选中 shift E



选择unsighed char array，复制下面的数据结果，即为密文数据。

exp

```
#include <stdio.h>
#include <string.h>
#include <stdint.h>

void encode(char *a)
{
    for (int i = 0; i < strlen(a); i++)
    {
        uint8_t c = *(a + i);
        *(a + i) = (c << 4) + (c >> 4);
    }
}

int main()
{
    //首先逆向分析出b数组为密文,ida提取。
    char b[] = {102, 198, 22, 118, 183, 55, 134, 246, 87, 230,
                150, 150, 55, 22, 214, 22, 245, 86, 150, 86,
                230, 230, 246, 182, 22, 214, 150, 215,0};
    encode(b);
    printf("%s\n", b);
}
```