奇安信攻防社区 - Dump 内存得到 TeamViewer 账号密码

> 奇安信攻防社区 – Dump 内存得到 TeamViewer 账号密码

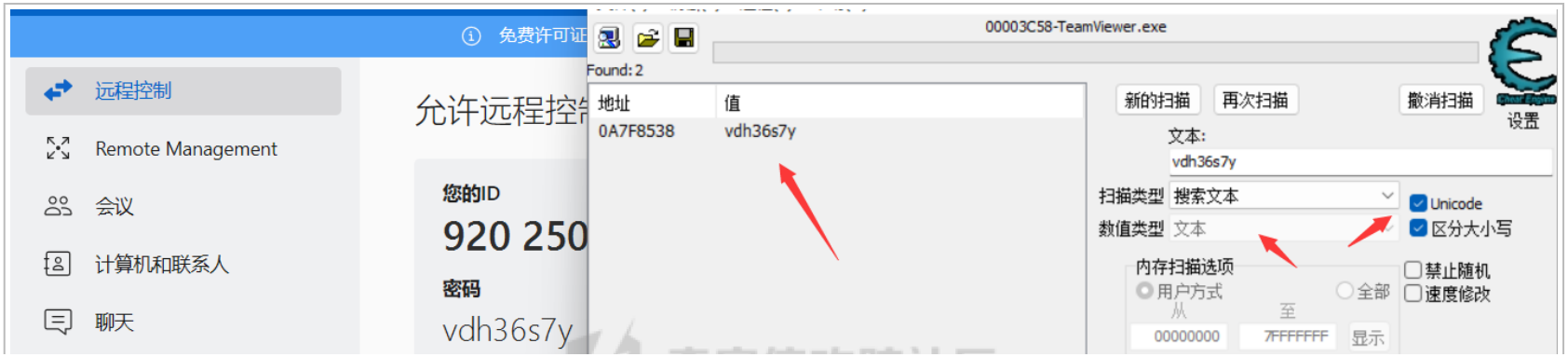
最近看到用窗体得到 TV 的账号密码在最新版不能用了 于是就想写个工具实现一下通过内存得到账号密码 ## 0x01 通过 CE 搜索账号密码存在的内存块 类型设置为文本，选择 unicode 编码，多搜...

最近看到用窗体得到 TV 的账号密码在最新版不能用了

于是就想写个工具实现一下通过内存得到账号密码

0x01 通过 CE 搜索账号密码存在的内存块

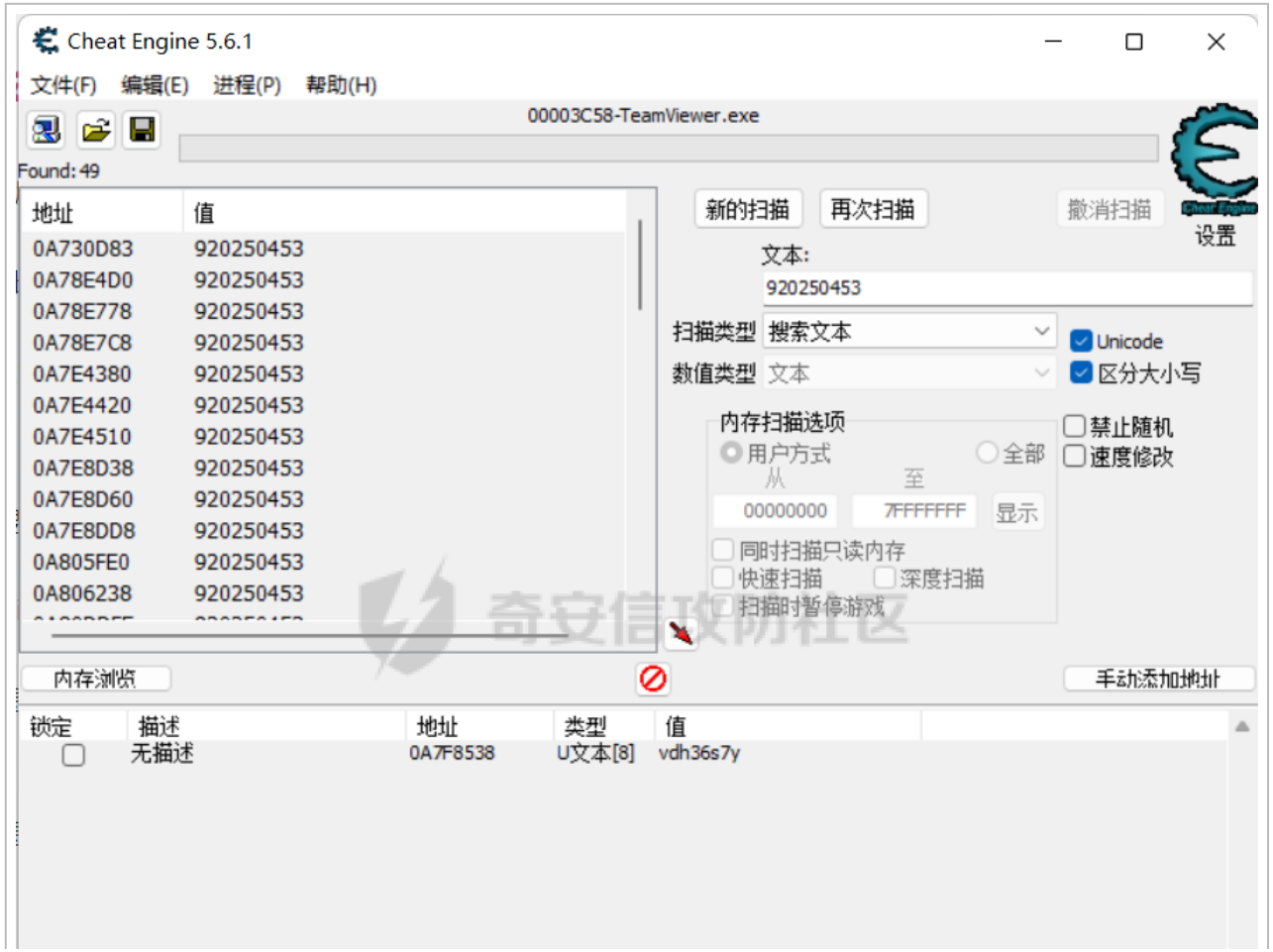
类型设置为文本，选择 unicode 编码，多搜索几次找到这个值



本来想的是应该有个指针直接指向密码，想把这个指针的基址找到就可以了，但是调了一下好像找不到这个基址

还有 ID 是不可以修改的，定位也不方便，想到遍历内存来得到 ID 和密码

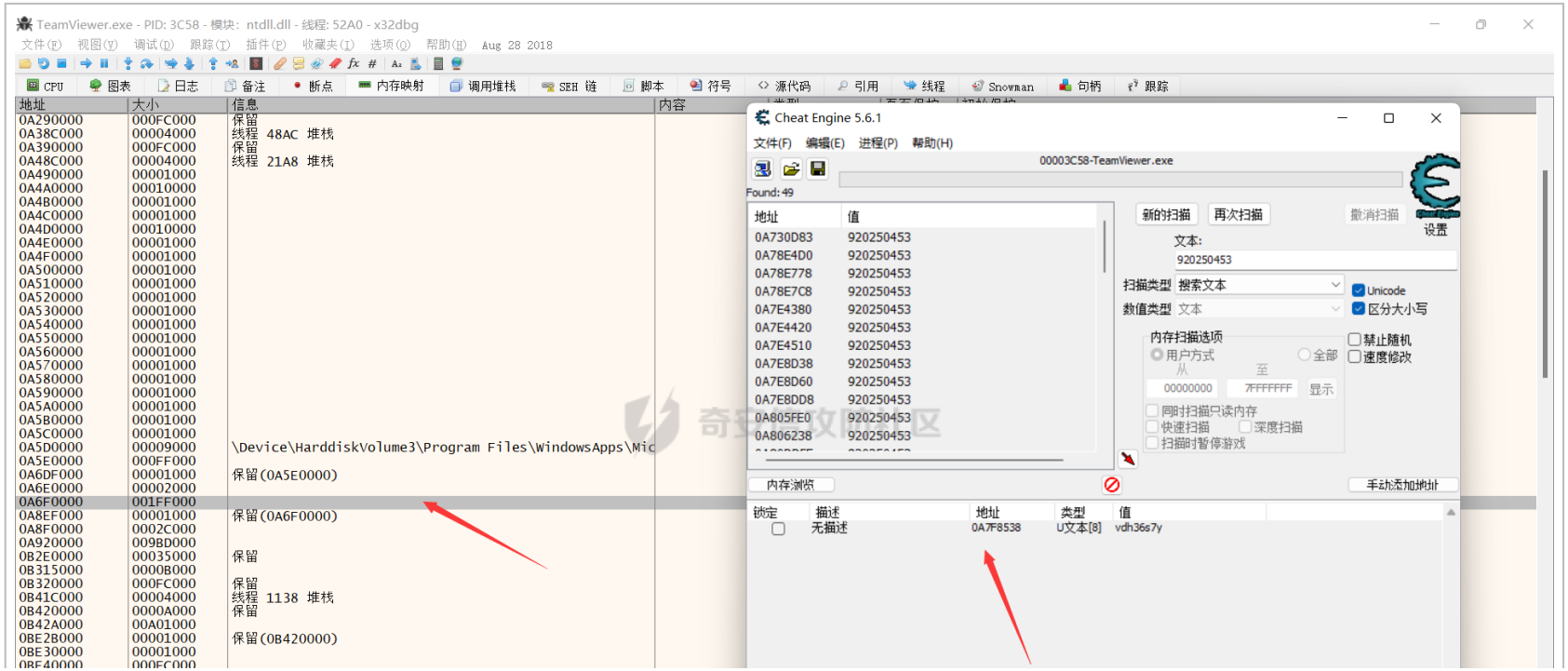
再用 CE 搜索一下 ID


Cheat Engine 5.6.1
文件(F) 编辑(E) 进程(P) 帮助(H)
00003C58-TeamViewer.exe
Found: 49
地址 值
0A730D83 920250453
0A78E4D0 920250453
0A78E778 920250453
0A78E7C8 920250453
0A7E4380 920250453
0A7E4420 920250453
0A7E4510 920250453
0A7E8D38 920250453
0A7E8D60 920250453
0A7E8DD8 920250453
0A805FE0 920250453
0A806238 920250453
新的扫描
再次扫描
撤消扫描
设置
文本:
920250453
扫描类型 搜索文本
数值类型 文本
Unicode
区分大小写
内存扫描选项
用户方式
全部
从
至
00000000
7FFFFFFF
显示
同时扫描只读内存
快速扫描
深度扫描
扫描时暂停游戏
禁止随机
速度修改
内存浏览
手动添加地址
锁定 描述 地址 类型 值
无描述 0A7F8538 U文本[8] vdh36s7y

可以看到在密码的附近都是有很多 ID

用的是遍历可以不用知道具体的位置，剩下的就是要思考怎么让遍历的内存更准确，遍历 0000000-7FFF0000 肯定是可以的，但是这样会出现很多误报，因为后面是准备使用正则匹配的，难免会匹配到别的字符串

先用 x32dbg 查看下内存的属性



从 CE 上看 ID 和密码就在这块内存里面，这里有个特征就是这块内存的大小是 1FF000，后面会用到

那么思路就是先得到进程的基址，然后遍历所有内存块基址，找到一个 1FF000 大小的内存，遍历内部内容，得到 ID 密码，这样遍历的内存也不会很大，也可以降低匹配误差

0x02 需要用到的函数和结构

下面介绍一下需要用到的函数和结构

ZwQueryVirtualMemory

```
typedef NTSTATUS(WINAPI* fnZwQueryVirtualMemory) (
```

这个函数就是获取内存块的属性然后存放到 MEMORY_BASIC_INFORMATION 结构

MemoryInformationClass

```
HANDLE ProcessHandle,                //进程句柄
```

这是一个枚举类型，选择需要什么内存信息，这里需要遍历内存选择 MemoryBasicInformation 就可以

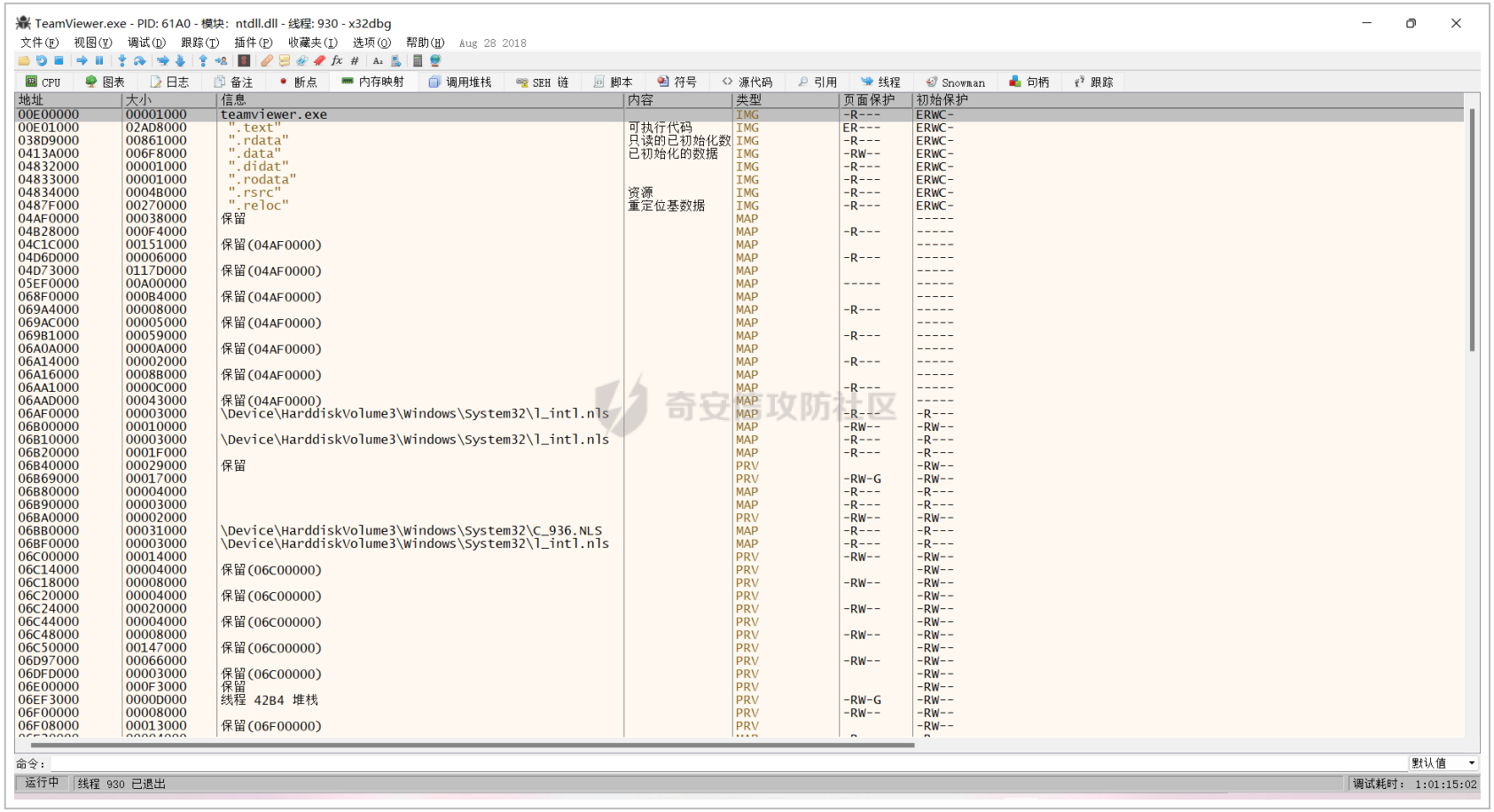
MEMORY_BASIC_INFORMATION

```
PVOID BaseAddress,                    //内存地址
```

EnumProcessModules

```
MEMORY_INFORMATION_CLASS MemoryInformationClass,        //选择需要的内存信息，下面介绍
```

这个函数主要是用来找到进程的基地址



可以看到进程的基地址是偏向上面的，只要往下遍历就好

0x03 实现过程

1. EnumProcessModules 得到进程的基地址

2. 用 do…while 循环配合 ZwQueryVirtualMemory 得到内存块属性，如果不是 1FF000 就加上内存块的大小跳到下一个内存块，如果是的话直接得到模块的基地址然后遍历这个模块的内存

3. 用 ReadProcessMemory 将内存读出来

4. 用正则表达式加特征匹配内存中的字符

关于最后一点的特征，光知道大小只能定位模块，还需要知道一些 ID 密码附近的内存特征

发现 ID 的前面会有个 0x80，后面会用 0x00，0x00 结尾

密码前面有 0x88，用 0x00，0x00 结尾

还有一个坑点就是 unicode 的正则表达式匹配，没找到特别好的方法

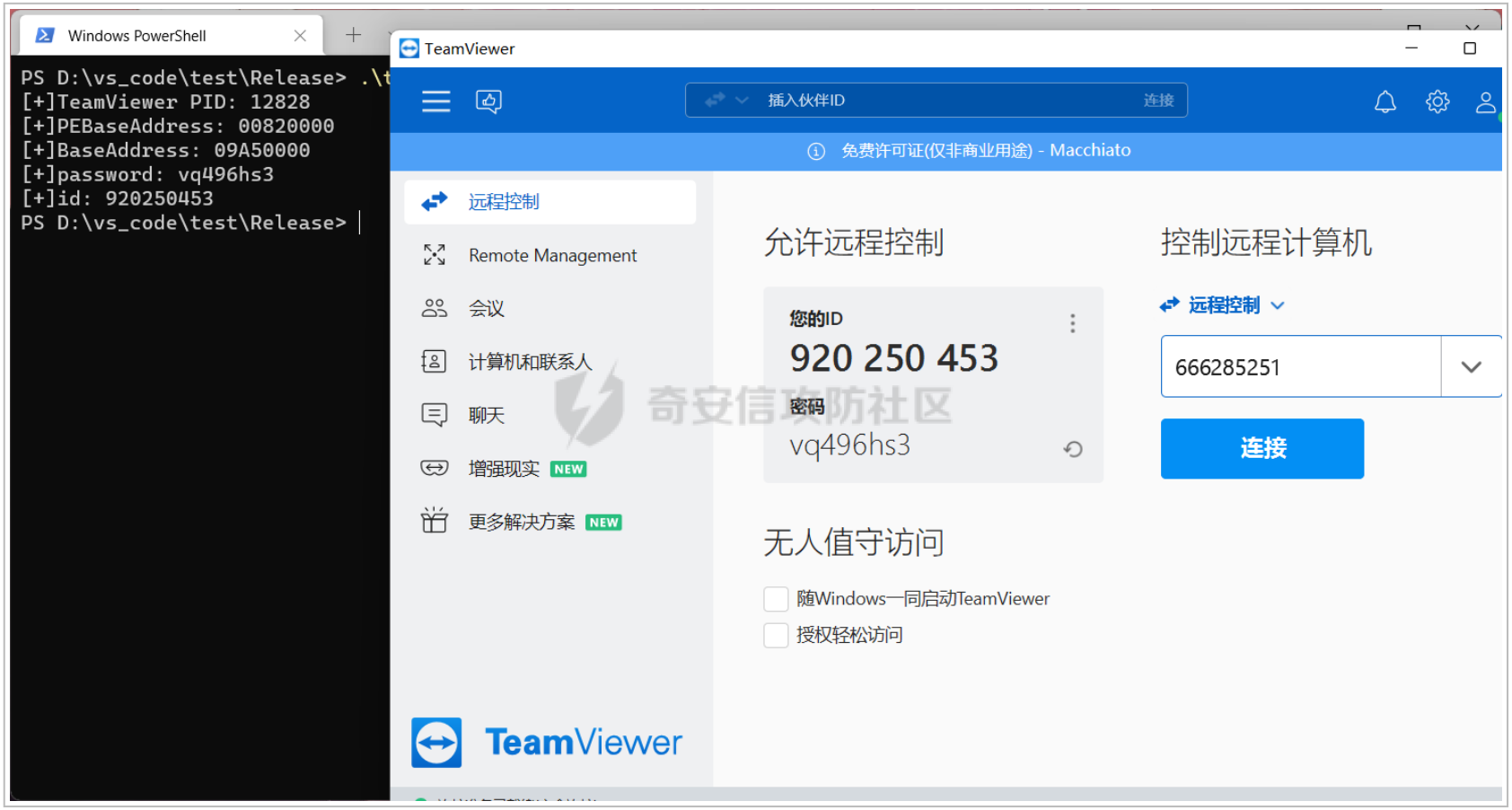
还好这里都是英文和数字，只要取出 13579 位置的值然后放入一个 char 类型的数组中，就可以用正则匹配了

如下

```
PVOID MemoryInformation,            //指向MEMORY_BASIC_INFORMATION结构的指针
```

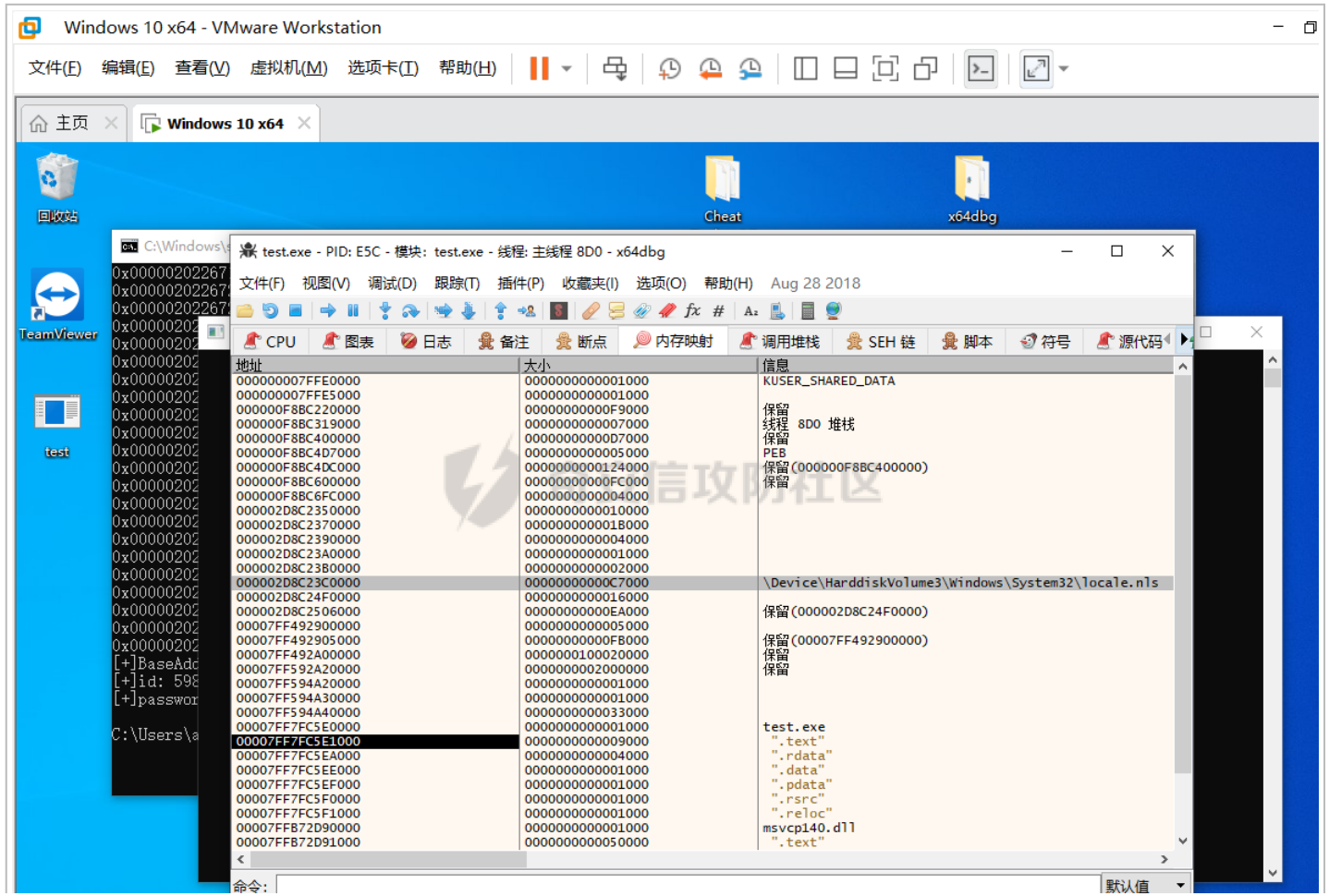
0x04 代码实现 x32

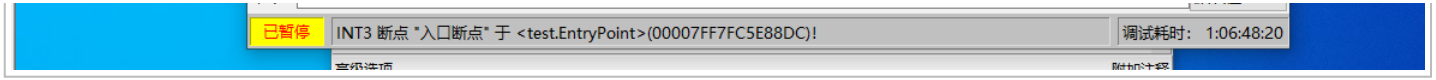
```
SIZE_T MemoryInformationLength,      //MEMORY_BASIC_INFORMATION结构的大小
```



0x05 代码实现 x64

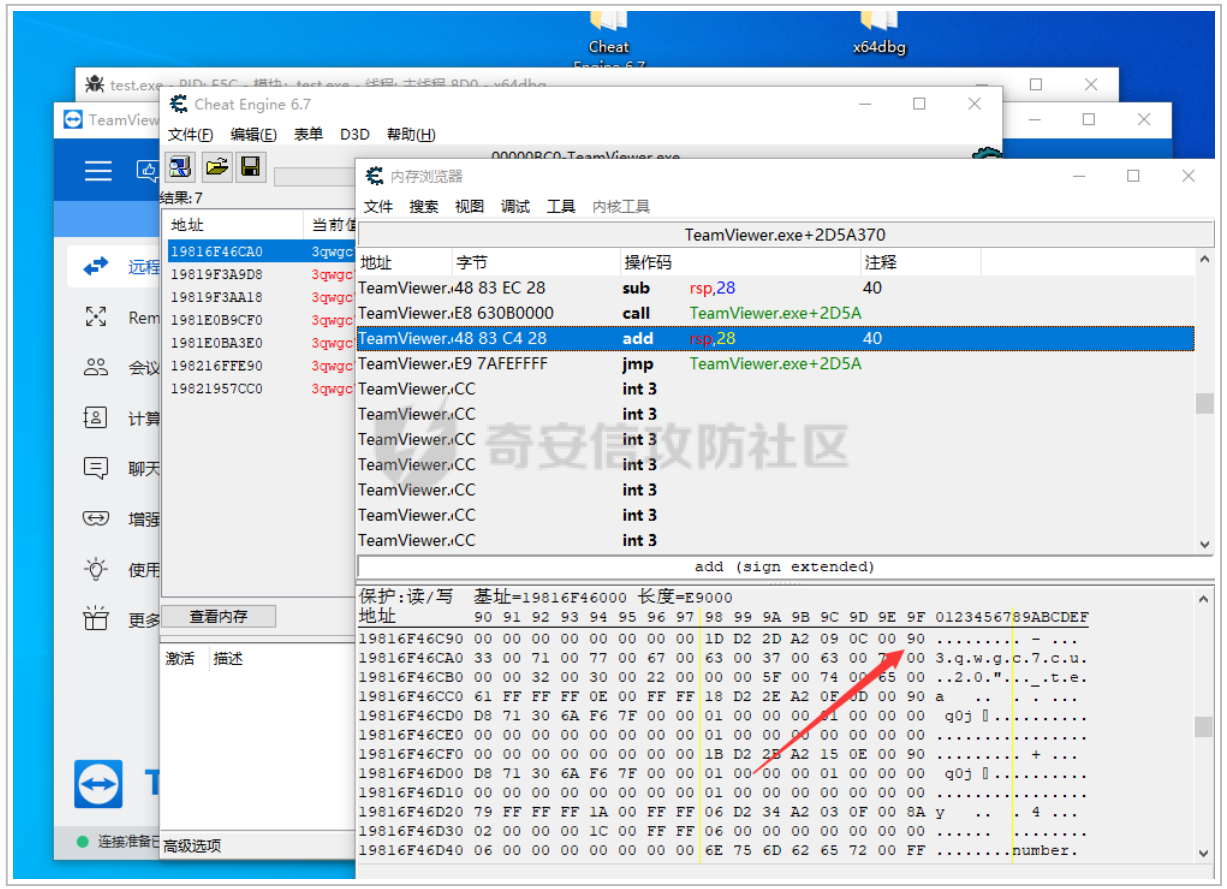
64 位中线程的内存地址都比进程基址小了，就是存有 ID 密码的内存都到进程上面了





都是 7FFE0000 开始，这样就不用先得到进程基址，可以直接遍历

还有在 64 位中密码开头的数字变成了 0x90，这也是需要改下的，别的基本都是相同的



贴一下修改后的代码

```
PSIZE_T ReturnLength                    //返回结构的大小
```