Wannacry勒索病毒分析

目录

[Wannacry勒索病毒分析 1](#_Toc22764292)

[1. 概述 3](#_Toc22764293)

[2. 实验环境 3](#_Toc22764294)

[3. 样本信息 3](#_Toc22764295)

[4. 基础静态分析 3](#_Toc22764296)

[4.1 PEiD分析 3](#_Toc22764297)

[4.2 ResHacker分析 5](#_Toc22764298)

[5. 基础动态分析 6](#_Toc22764299)

[5.1进程监控 6](#_Toc22764300)

[5.2 注册表监控 6](#_Toc22764301)

[5.3 网络监控 7](#_Toc22764302)

[5.4 文件监控 7](#_Toc22764303)

[6. 详细分析 8](#_Toc22764304)

[6.1 wcry.exe分析 8](#_Toc22764305)

[GetAscii() 获取随机数 9](#_Toc22764306)

[SetRegedit() 设置注册表项 10](#_Toc22764307)

[FreeRsrcFile() 释放资源文件 11](#_Toc22764308)

[Write\_cwnry() 写入比特币账户信息 11](#_Toc22764309)

[RunCmdCommand() 执行命令行参数 12](#_Toc22764310)

[GetSomeAPI() 获取必要API函数 12](#_Toc22764311)

[InitCriticalSec() 初始化临界区 13](#_Toc22764312)

[KeyAndAllocHeap() 导入密钥和申请内存空间 13](#_Toc22764313)

[DecrypttFile() 解密t.wnry文件 14](#_Toc22764314)

[PeToMem() 拷贝PE文件到内存 15](#_Toc22764315)

[GetFuncAddr() 获取TaskStart函数地址 17](#_Toc22764316)

[6.2 t.wnry.dll分析 18](#_Toc22764317)

[SingleRun() 单一运行 19](#_Toc22764318)

[GetUserIdAndCmp() 获取当用户SID并比较 20](#_Toc22764319)

[SetAccessControl() 设置访问控制 20](#_Toc22764320)

[CheckFileExist() 检测dky文件 21](#_Toc22764321)

[CreatePkyAndEky() 创建pky和eky文件 21](#_Toc22764322)

[线程函数1：CreateResFile() 创建Res文件 23](#_Toc22764323)

[线程函数2：CheckDky() 检测Dky文件 23](#_Toc22764324)

[线程函数3：EncryptAllFiles() 加密所有文件（核心函数） 24](#_Toc22764325)

[线程函数4：RunTaskdl() 隐蔽启动taskdl.exe 33](#_Toc22764326)

[线程函数5：RunExeAndSetReg() 启动文件和注册表写入 33](#_Toc22764327)

[RepeatOperation() 重复操作 34](#_Toc22764328)

[6.3 taskdl.exe分析 36](#_Toc22764329)

[DeleteFile() 删除回收站和临时目录下的.WNCRY文件 36](#_Toc22764330)

[6.4 taskse.dll分析 37](#_Toc22764331)

[ChangePrivileges() 修改权限 37](#_Toc22764332)

[7. 总结：wcry释放文件 38](#_Toc22764333)

[8. 总结：wcry行为分析 39](#_Toc22764334)

[9. 解决方案 39](#_Toc22764335)

[10. 参考资料 40](#_Toc22764336)

# 概述

Wannacry又称“永恒之蓝”，是利用Windows系统的SMB漏洞可以获取系统最高权限。

该病毒分为蠕虫传播和勒索病毒部分，这里只分析勒索病毒部分。分析参考了他人的分析资料和MSDN文档，分析结果中可能存在一些遗漏或错误。

# 实验环境

|  |  |
| --- | --- |
| 实验平台 | Windows 7 (32/64位)虚拟机 |
| 分析工具 | IDA 6.8，OllyDBG |
| 其他工具 | PEiD，ResHacker，ProcMon，hrsword等 |

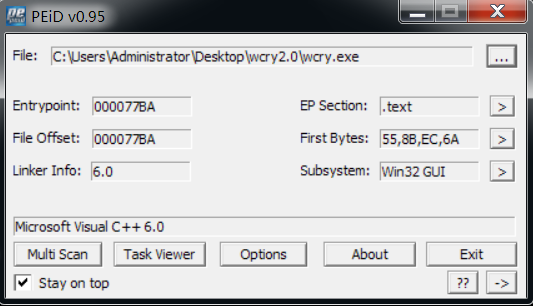
# 样本信息

|  |  |
| --- | --- |
| 文件名 | ~\wcry2.0\wcry.exe |
| 文件大小 | 3,514,368 字节 |
| 文件版本 | 6.1.7601.17154 |
| 修改日期 | 2017/5/13 2:21 |
| MD5 | 84C82835A5D21BBCF75A61706D8AB549 |
| SHA1 | 5FF465AFAABCBF0150D1A3AB2C2E74F3A4426467 |
| SHA256 | ED01EBFBC9EB5BBEA545AF4D01BF5F1  071661840480439C6E5BABE8E080E41AA |

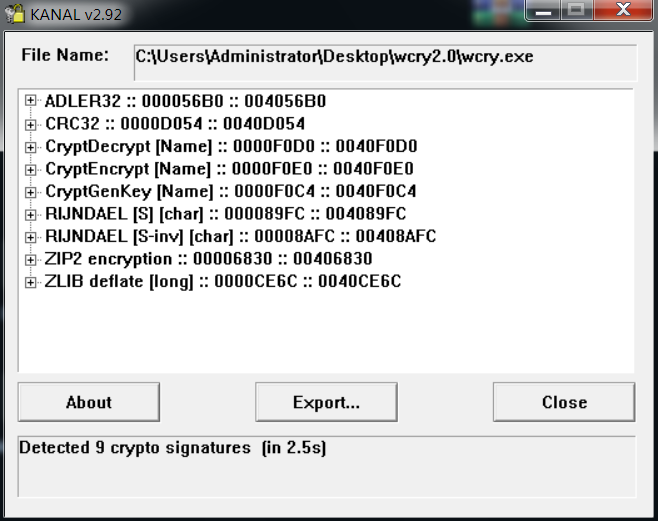
# 基础静态分析

## 4.1 PEiD分析

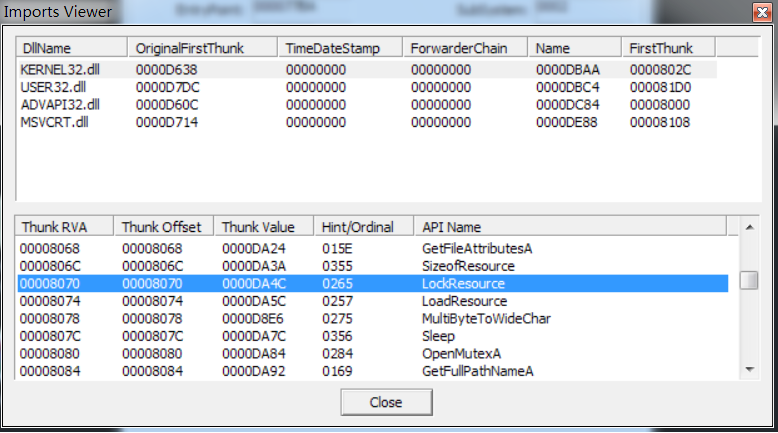
1. 首先使用PEid分析该勒索病毒是否加壳。结果可见，该勒索病毒并未加壳，且使用VC++6.0编译器编写。



1. 使用PEiD的Kyrpto ANALyzer插件识别病毒的加密算法，共扫描得到了9个加密算法签名，包括调用了CryptEncrypt和CryptDecrypt加密类库，ZIP2和ZLIB压缩算法，CRC32和AES加密算法。



1. 使用PEid分析病毒的导入表，共导入了如下4个模块。其中，kernel32.dll模块中，调用了LoadResource、FindResourceA、LockResource等函数，可以猜测资源段中存在资源释放等操作。

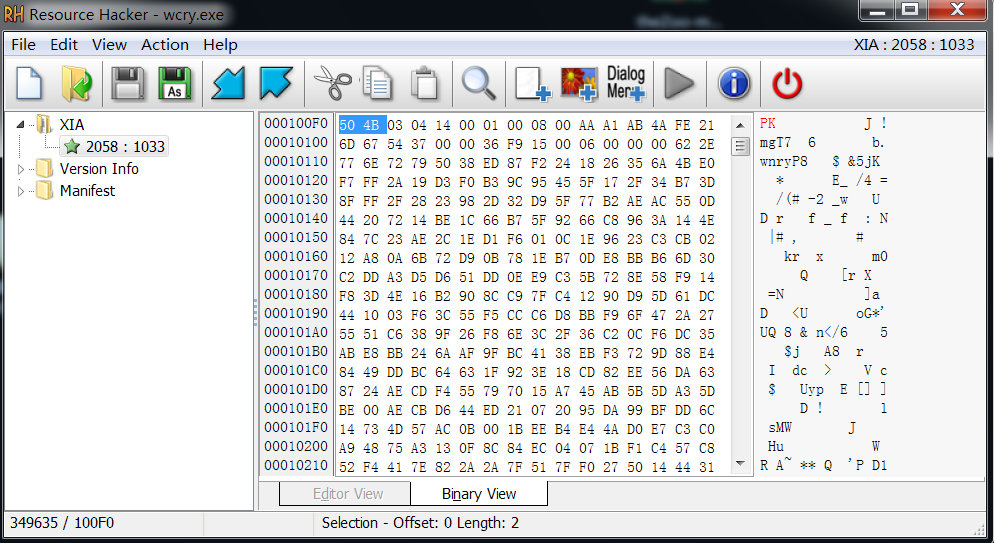


在advapi32.dll中，调用了RegCreateKeyW、RegSetValueExA等函数，表明该勒索病毒存在注册表写入功能。

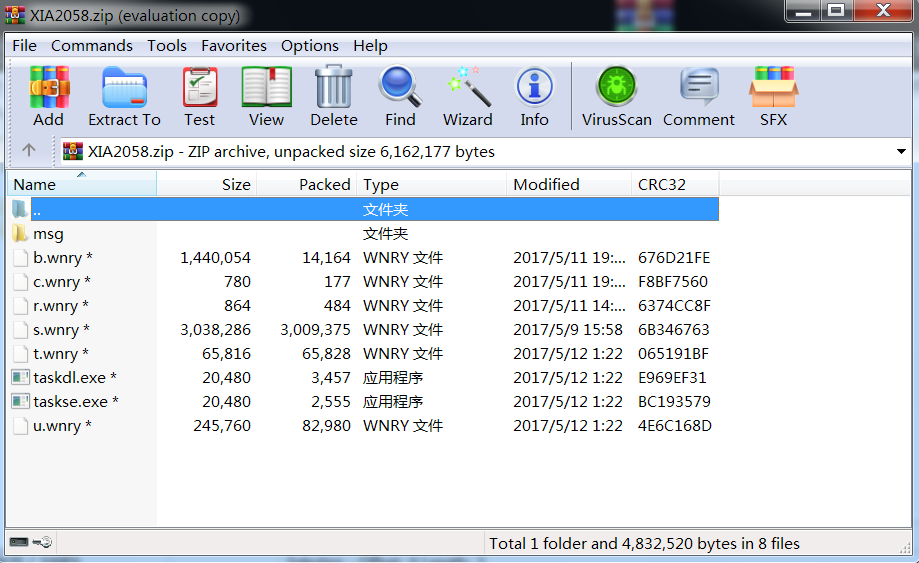


## 4.2 ResHacker分析

使用Resource Hacker工具查看病毒的资源段，如下图所示：



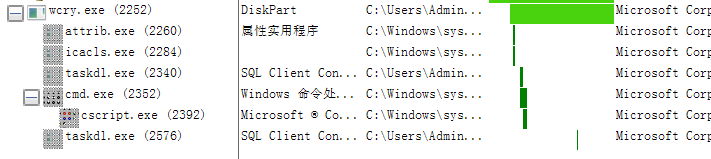
发现一个资源ID为2058，资源名为XIA的自定义资源。其中，“50 4B 03 04”文件头标识表示这是个zip压缩文件。将该资源提取出来，尝试使用WinRAR查看，可以查看病毒所要释放的文件（但没有解压密码，无法解压）。如下图所示：



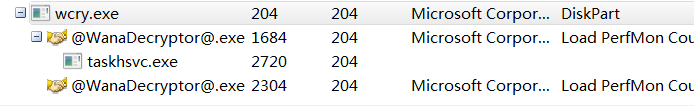
# 基础动态分析

## 5.1进程监控

在虚拟机上使用Process Monitor查看运行后的病毒进程，wcry.exe共创建了5个子进程，其中还使用cmd.exe执行了一个批处理脚本。显示如下：

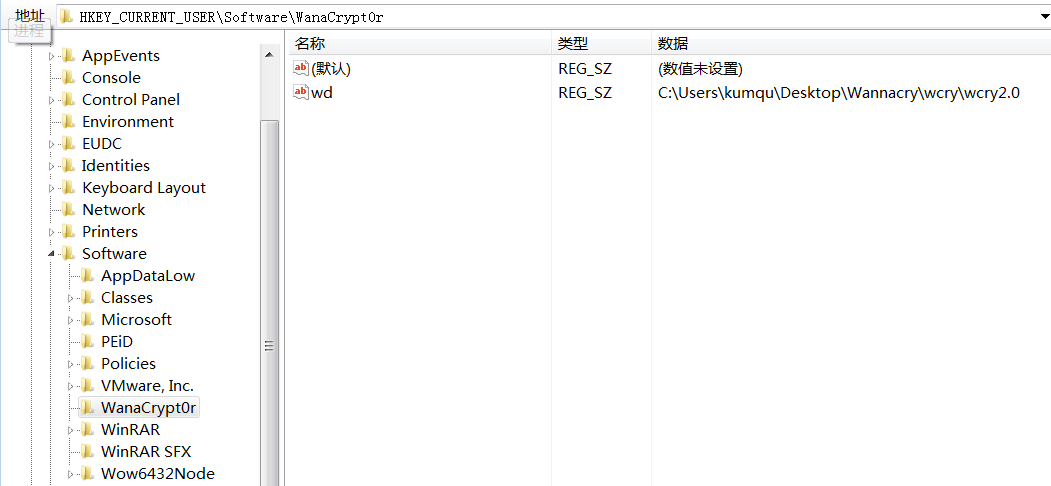


勒索病毒执行完全部的功能后，使用hrsword再次查看病毒进程（此时ProcMon已不可用），病毒进程存在两个子进程运行，一个是解密器，另一个猜测是与病毒的传播有关。



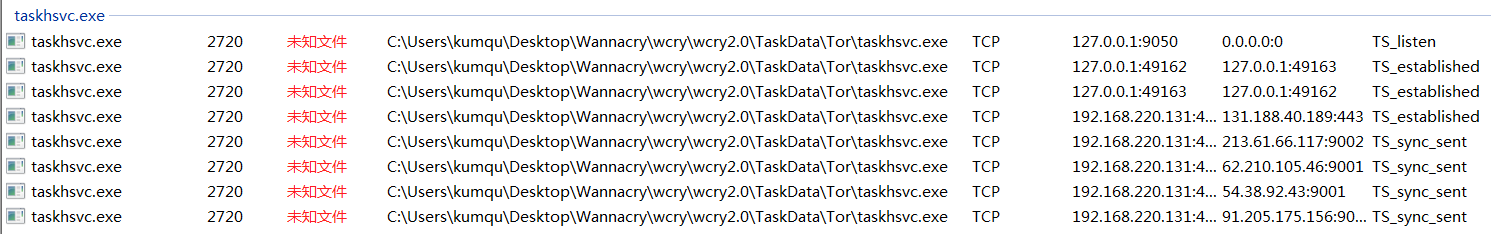
## 5.2 注册表监控

使用hrsword监控病毒的注册表，显示在HKCU\SOFTWARE中添加了一个键，键值为病毒的当前路径，如下图所示：



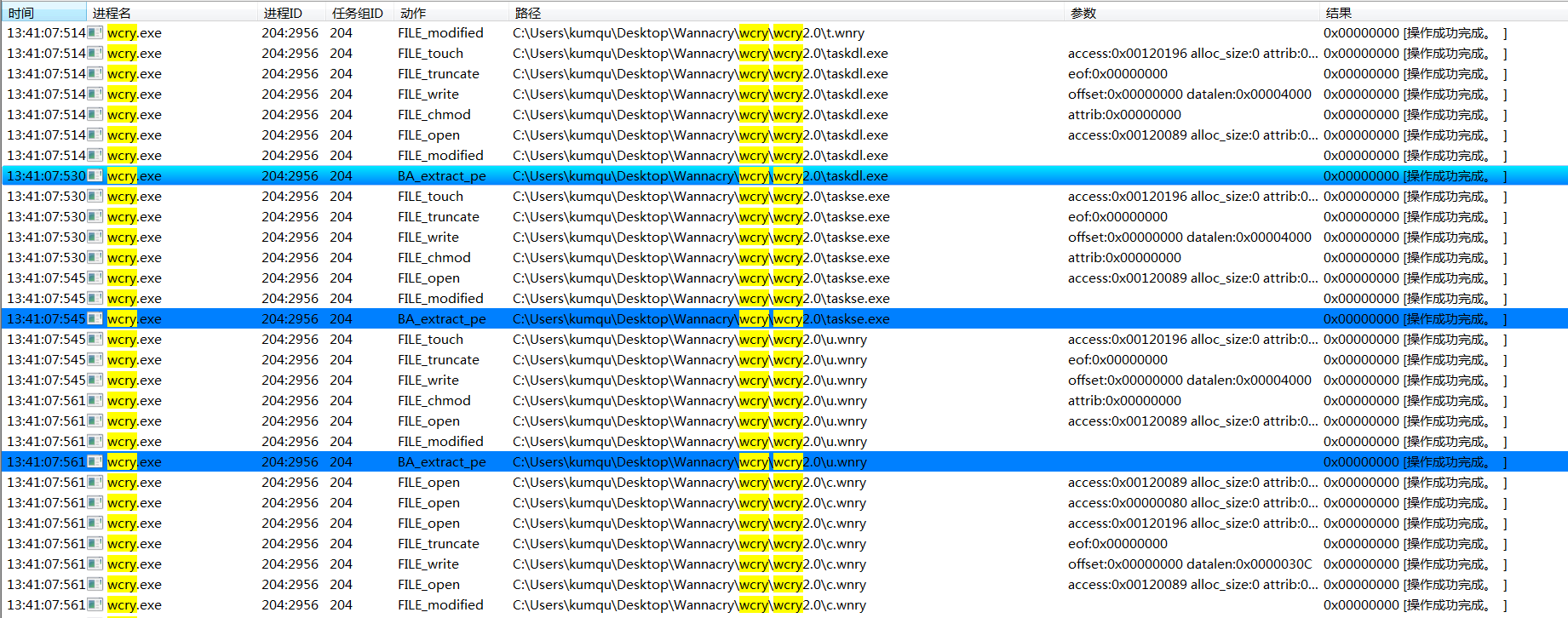
## 5.3 网络监控

使用hrsword监控网络，发现病毒监听了9050、49162和49163三个端口，并且有尝试连接内网IP和一些随机的IP地址，猜测这可能是病毒的扩散行为。

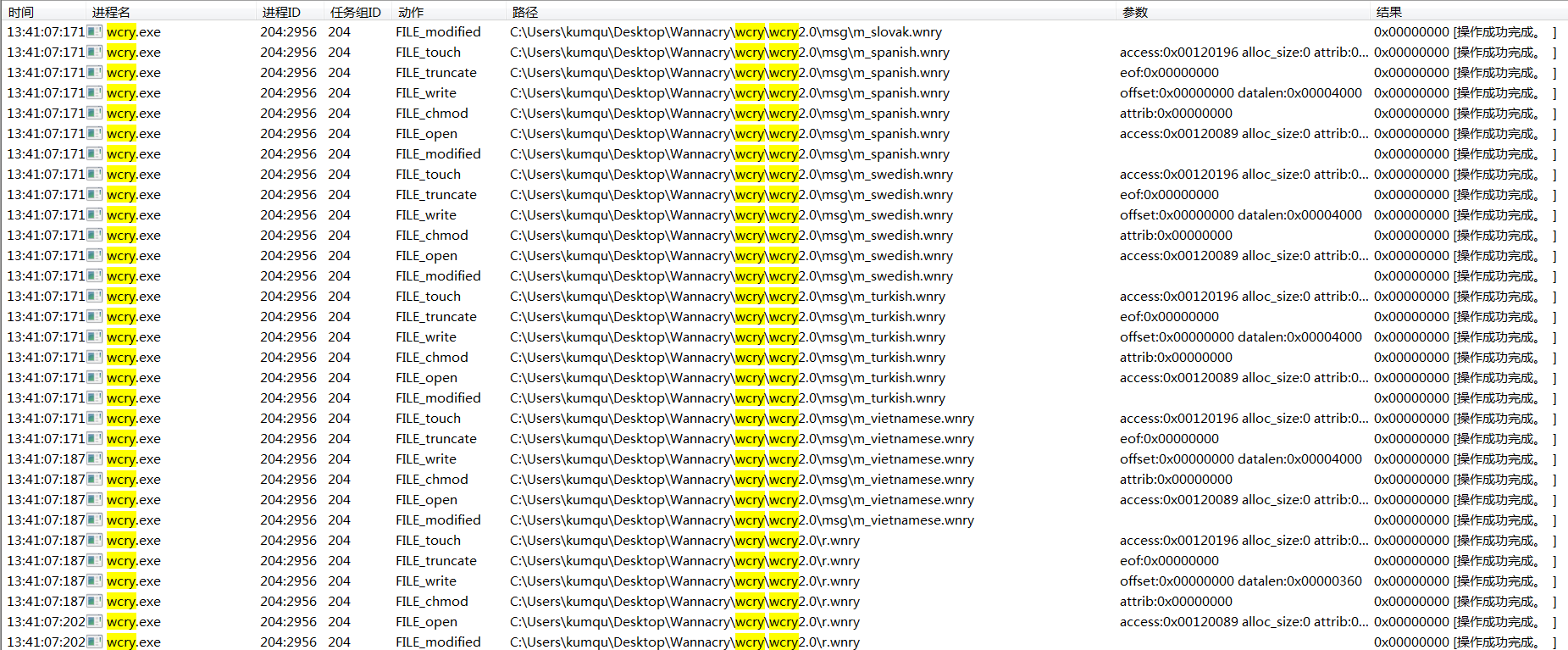


## 5.4 文件监控

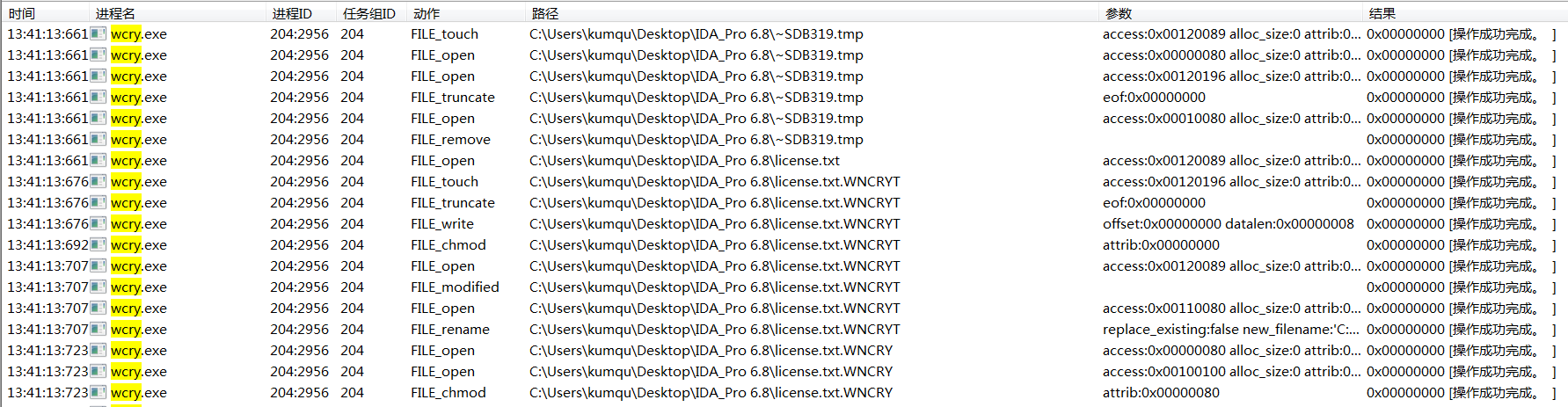
首先，使用hrsword可以观察到病毒释放了一些文件，包括taskdl.exe、taskse.exe、u.wnry等（可以验证前面的分析），如下图所示：



其次，显示了对这些释放的文件频繁的读取、修改、保存等操作，如下图所示：



最后，可以看到病毒对宿主机上的文件进行加密的操作，如下图所示：

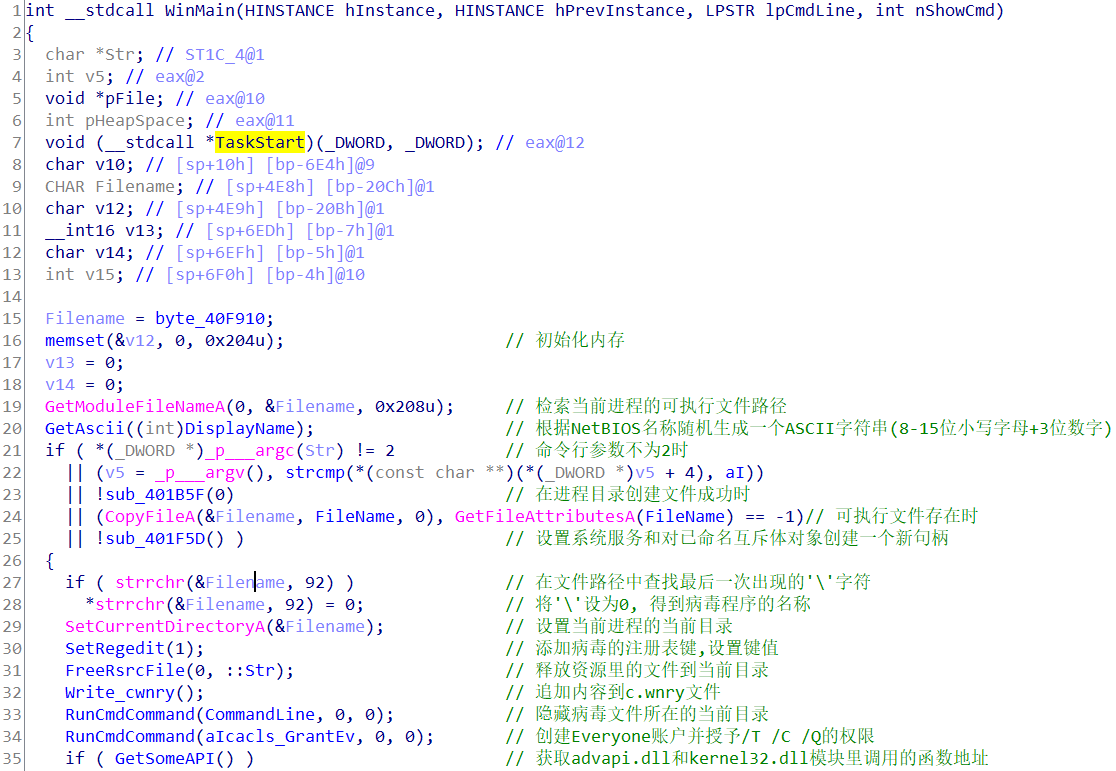


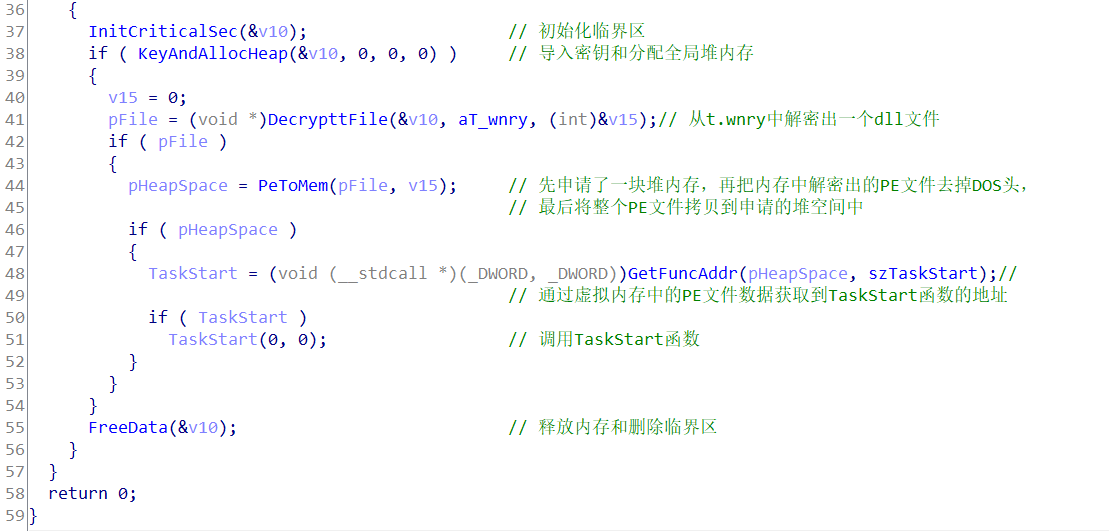
# 详细分析

这部分使用IDA和OD对勒索病毒进行了详细的分析。

## 6.1 wcry.exe分析

这个可执行文件是病毒的主程序。在IDA中找到WinMain函数，并对其中函数逐一分析。接下来按照主体逻辑图顺序分析关键函数，主体逻辑图如下所示，：

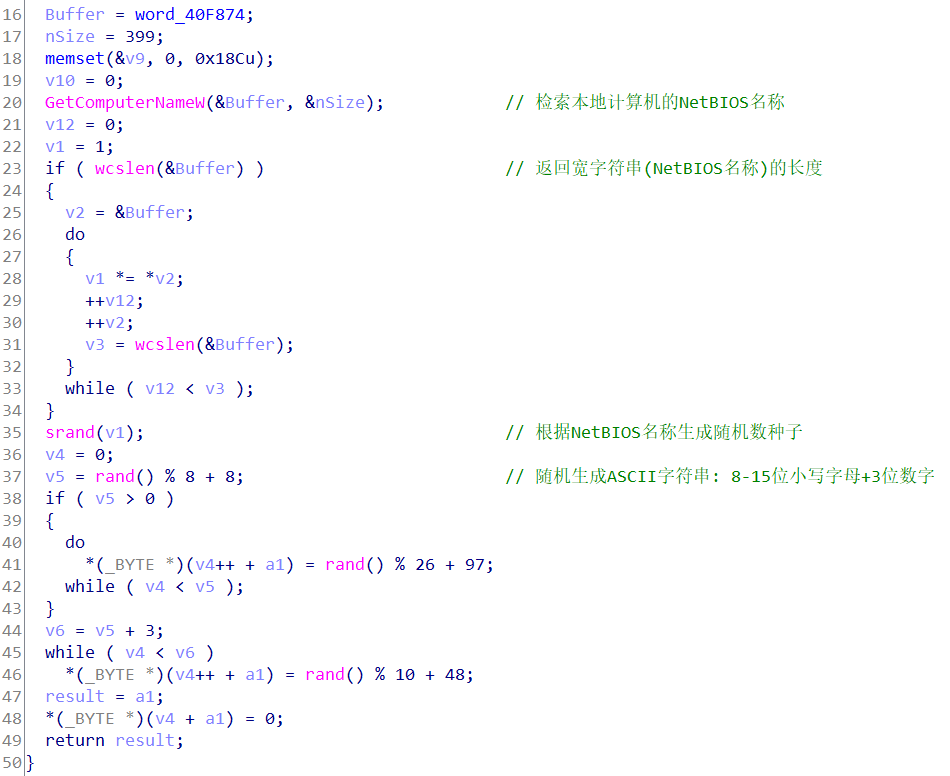




这部分的代码并不是病毒的核心代码，其主要功能是进行一些病毒初始化操作和解密病毒核心代码，接下来对这部分代码的关键函数逐一分析。

### GetAscii() 获取随机数

第20行的函数是根据NetBIOS名称随机生成一个ASCII字符串，将其重命名为GetAscii，对其分析如下：



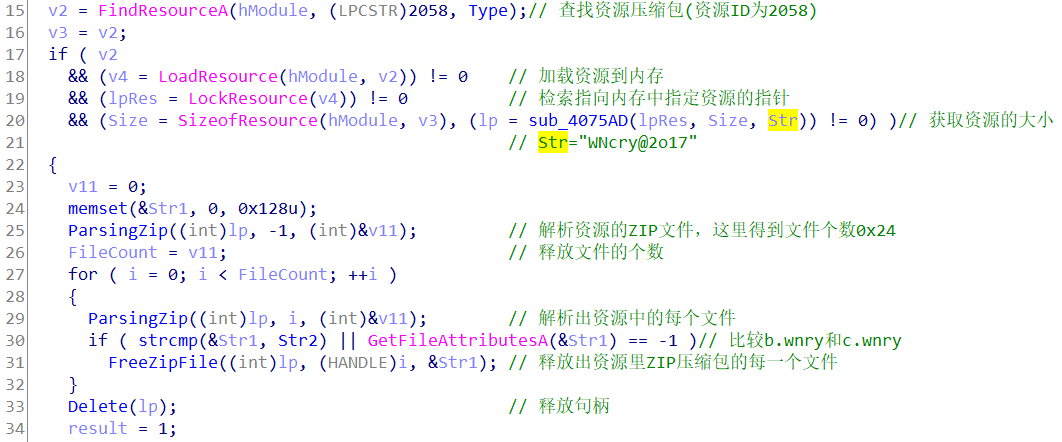
### SetRegedit() 设置注册表项

第20行的SetRegedit函数修改了注册表，添加了一个病毒当前路径的注册表项。如果有管理员权限，注册表项会设置在HKLM\SOFTWARE下；否则，注册表项设置在HKCU\SOFTWARE下。如下图所示：

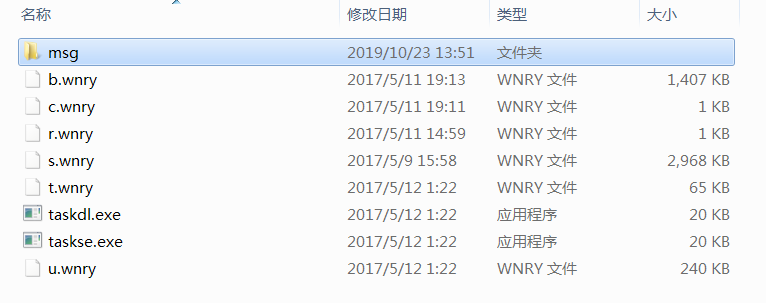


### FreeRsrcFile() 释放资源文件

第31行FreeRsrcFile函数将资源中隐藏的压缩包进行解压（解压密码“WNcry@2ol7”作为参数Str输入）,并且释放压缩包中的所有文件到当前进程的路径下，分析如下图所示：

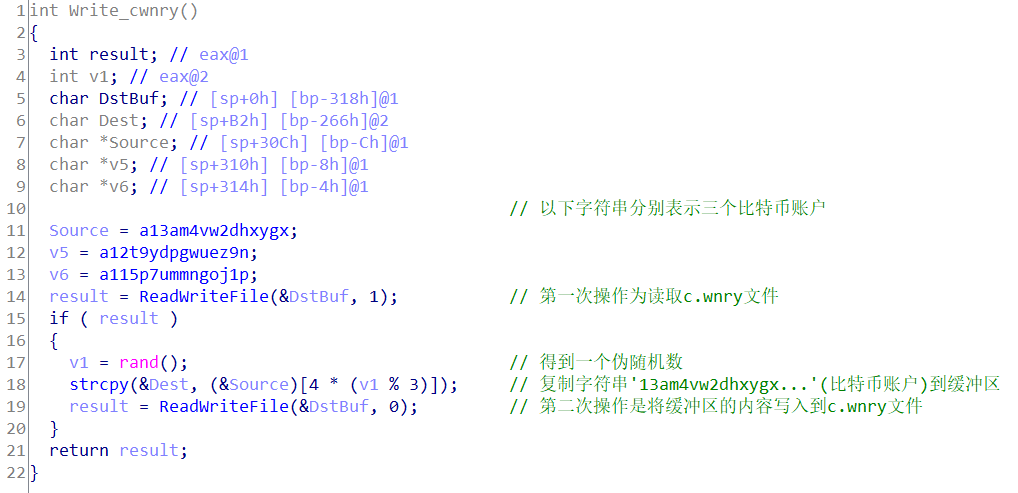


因此，使用上述解压密码解压提取出的资源文件，解压得到的文件如下图所示。其中，msg文件夹下包含的是病毒使用的语言包，其他文件暂不得知作用。



### Write\_cwnry() 写入比特币账户信息

第32行Write\_cwnry函数是随机从三个比特币账户中选取一个，并将该比特币账户信息写入c.wnry文件中。可以猜测c.wnry文件被显示勒索信息的程序使用。



### RunCmdCommand() 执行命令行参数

第33行、34行的RunCmdCommand函数执行了两次。

第一次，RunCmdCommand函数创建了一个进程，参数为“attrib +h”，作用是将当前目录下的所有文件设置为隐藏属性。

第二次，RunCmdCommand函数的参数为“icacls . /grant Everyone:F /T /C /Q”，作用是创建一个名为Everyone的用户，赋予这个用户所有权限。



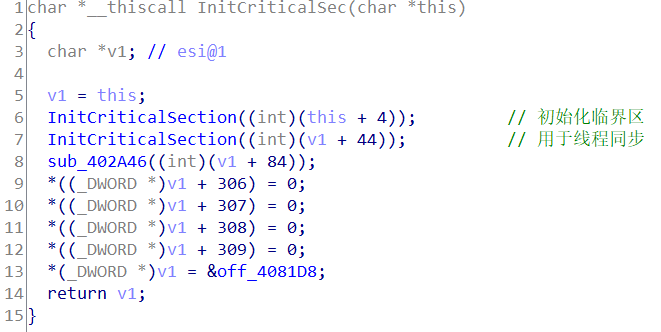
### GetSomeAPI() 获取必要API函数

第35行的GetSomeAPI函数获取了advapi.dll和kernel32.dll模块里必要的函数地址。



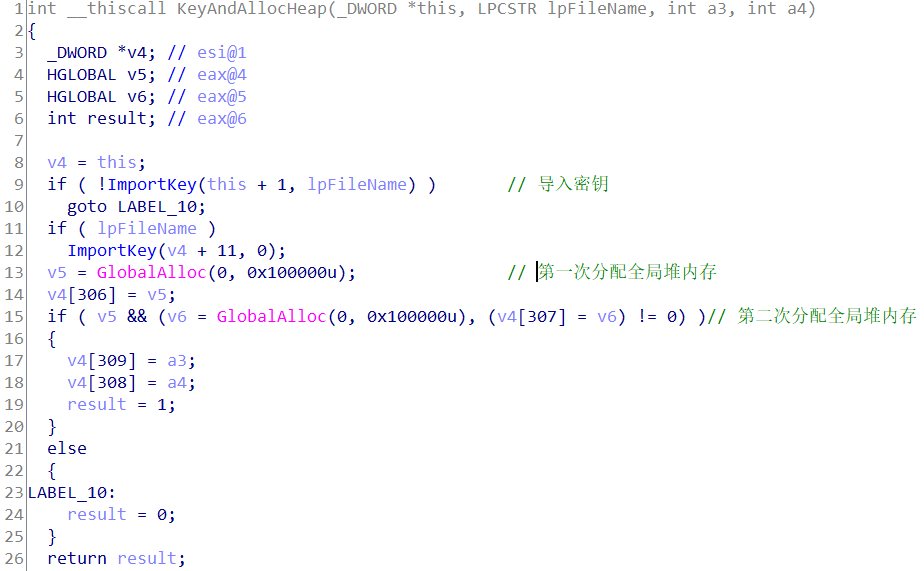
### InitCriticalSec() 初始化临界区

第37行的InitCriticalSec函数初始化临界区，用于线程同步。该病毒接下来会有5个线程同时运行（前面的基础动态分析-进程监控可以验证，接下来的分析也也可以验证），对于多线程程序，使用临界区来实现资源的访问，保证线程同步。



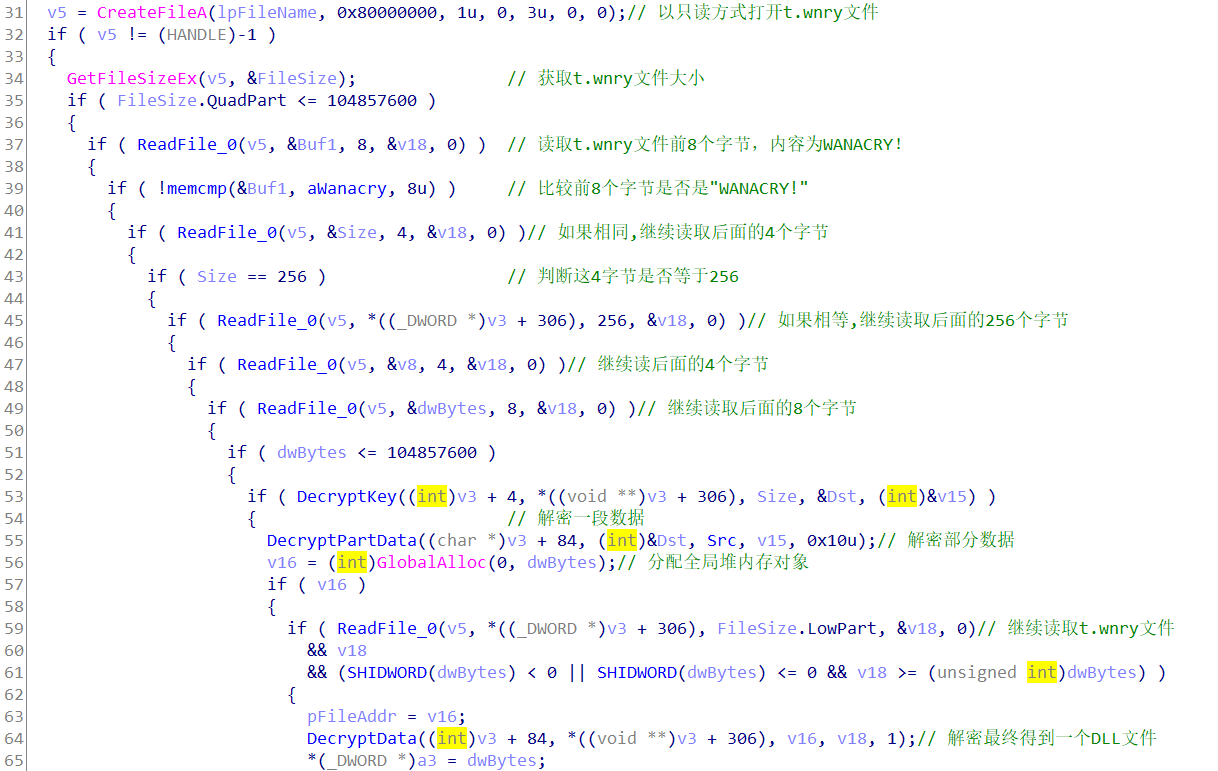
### KeyAndAllocHeap() 导入密钥和申请内存空间

第38行的KeyAndAllocHeap函数，功能为：导入私钥，用于后面解密文件；申请两块大小为0x100000的堆内存空间。

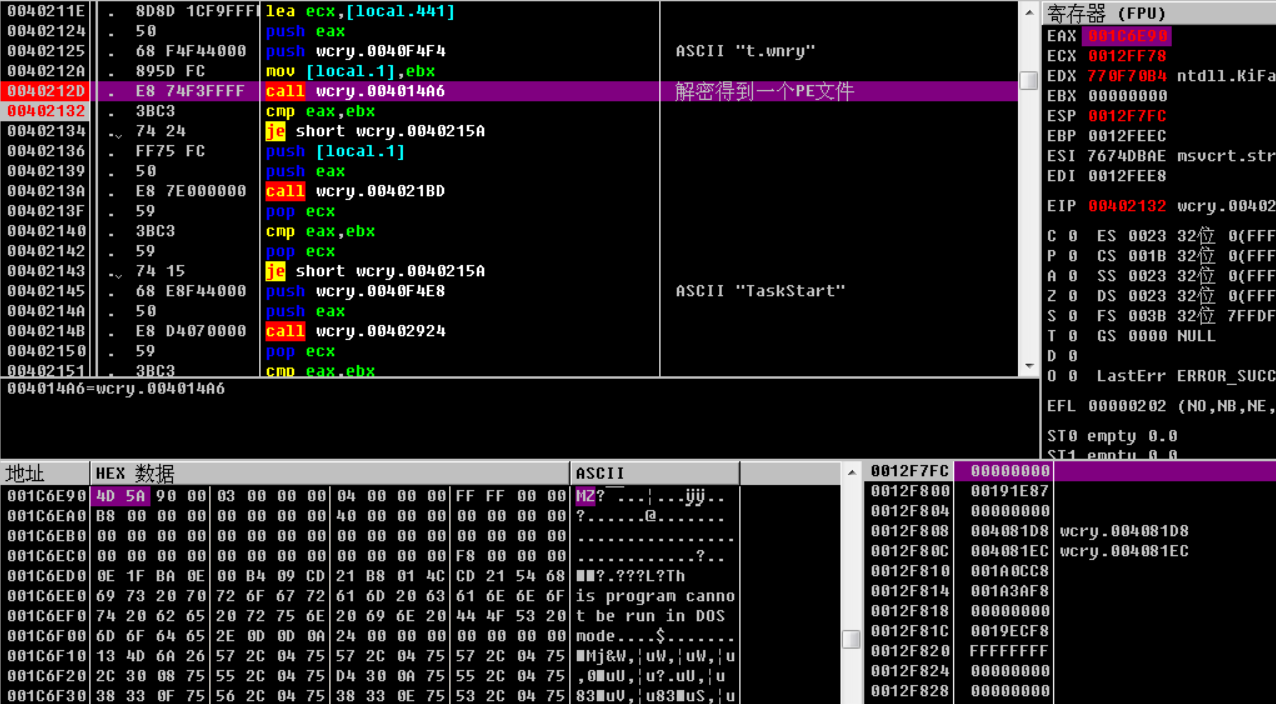


### DecrypttFile() 解密t.wnry文件

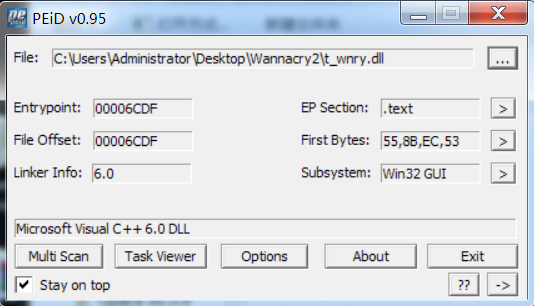
第41行的DecrypttFile函数是从t.wnry文件中解密出一个DLL文件，且该文件的大小为0x10000字节。该函数的主要部分分析如下所示：



这个函数对读取t.wnry文件数据到内存中，然后传入上个函数拿到的密钥句柄，在内存中解密数据，并返回解密之后的文件内容。“4D 5A”表示这是个PE文件。在OD中分析显示如下：

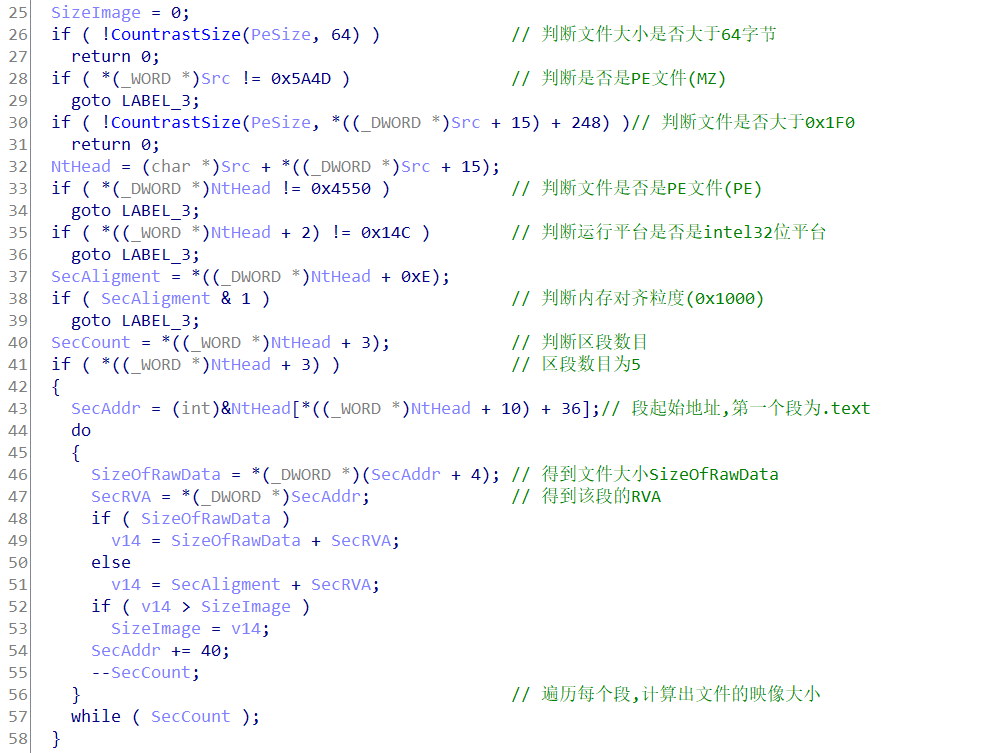


提取出解密后的0x10000字节的内存数据，保存为二进制文件。使用PEiD查看，显示这是个DLL文件，如下图所示：

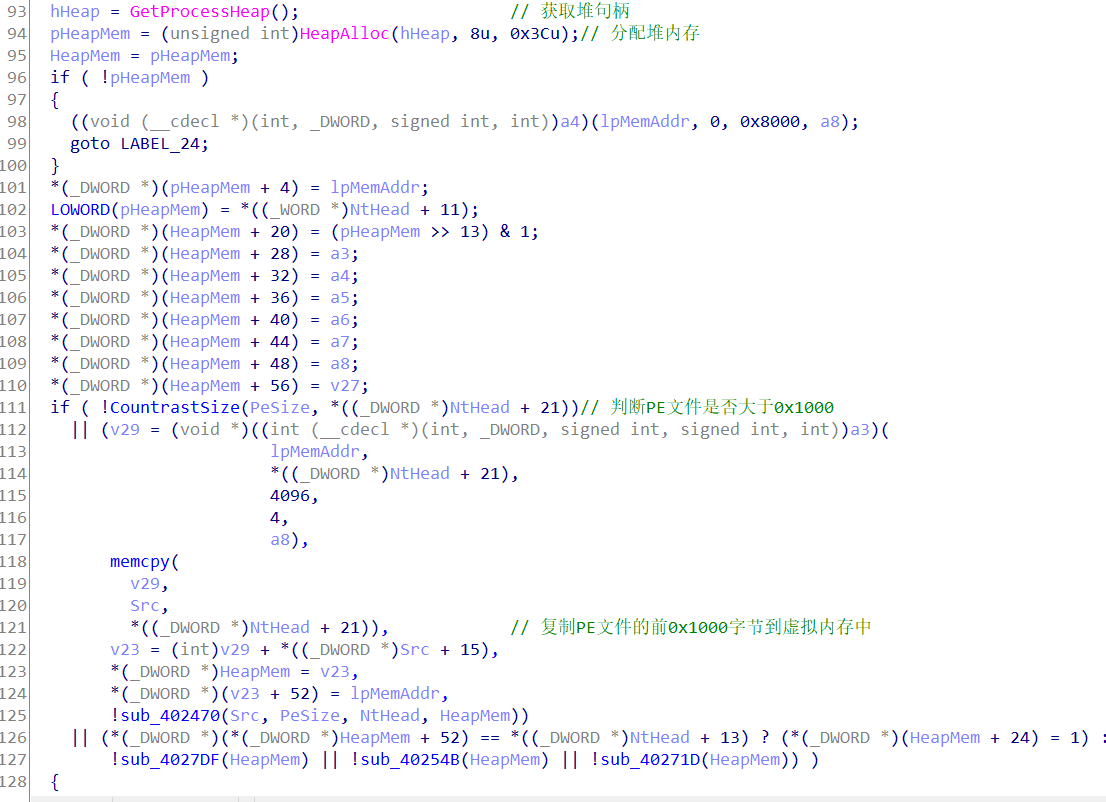


### PeToMem() 拷贝PE文件到内存

第44行的PeToMem函数先申请了一块堆内存，再把内存中解密出的PE文件去掉DOS头，最后将整个PE文件拷贝到申请的堆空间中。分析如下：







### GetFuncAddr() 获取TaskStart函数地址

第48行的GetFuncAddr函数，遍历虚拟内存中的PE文件数据(t.wnry.dll），最终得到TaskStart函数的地址。这个函数首先取出数据目录表，然后根据数据目录表得到导出表，最后根据导出函数名“TaskStart”得到其在内存中的函数地址。具体的分析如下：

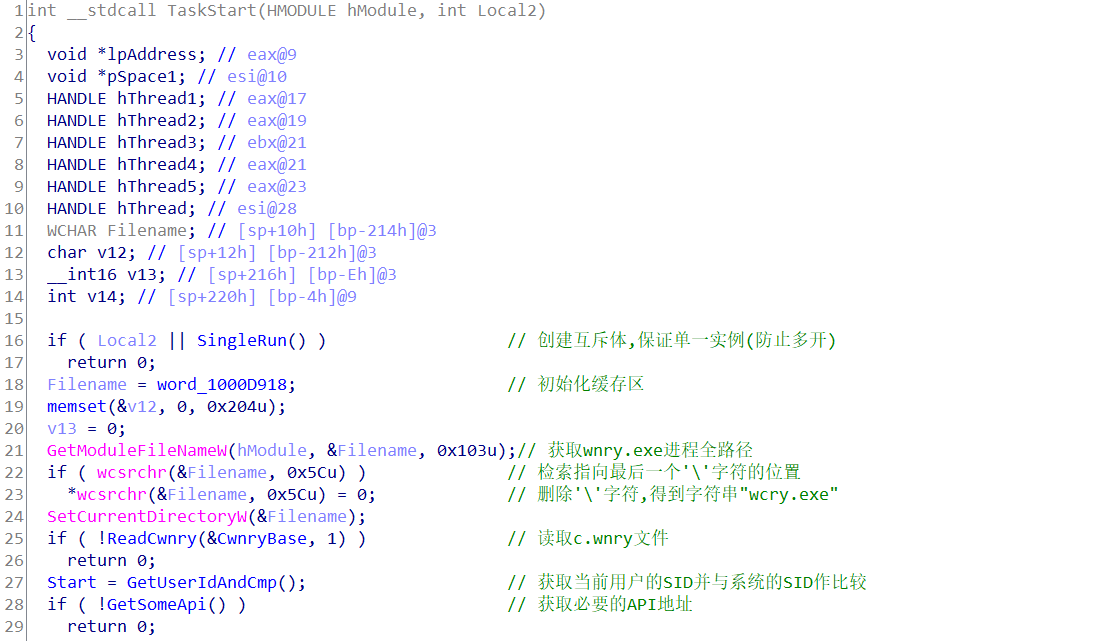




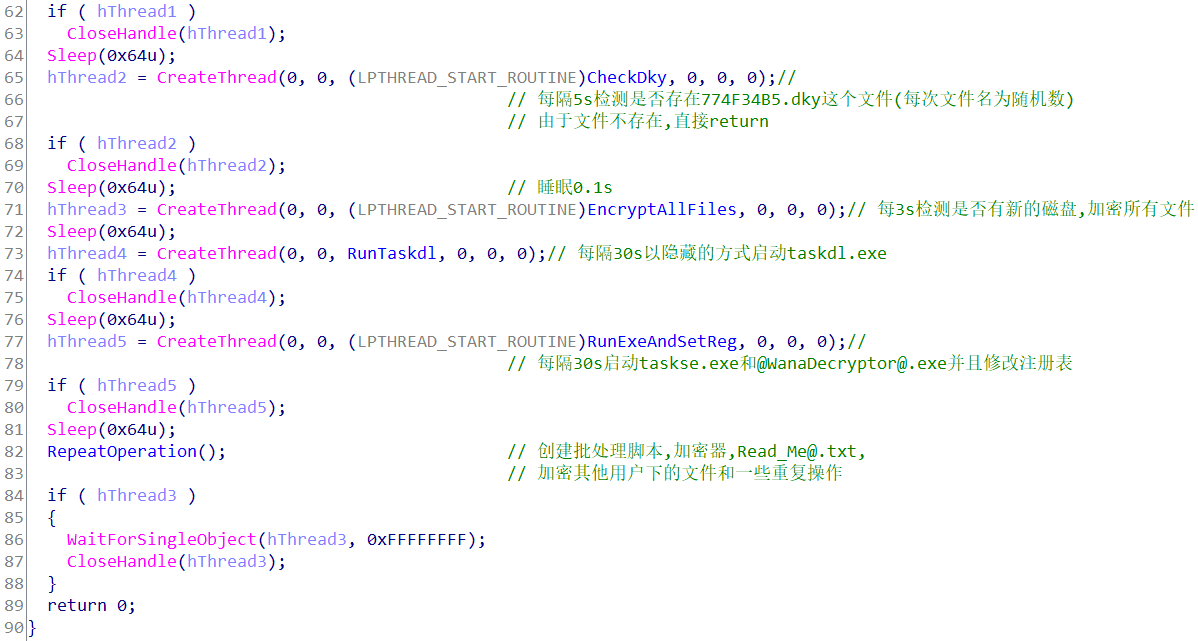
得到TaskStart函数的地址后，第50行直接执行这个函数。通过分析，这个函数才是病毒的核心函数，其实现了病毒的具体功能。而TaskStart函数位于t.wnry.dll中，因此，接下来对这个模块进行分析。

## 6.2 t.wnry.dll分析

在t.wnry.dll中定位到TaskStart函数进行分析。TaskStart函数包含病毒的所有操作，诸如加密文件、显示勒索信息等。接下来逐一分析其中的关键函数，其主体逻辑如下所示：

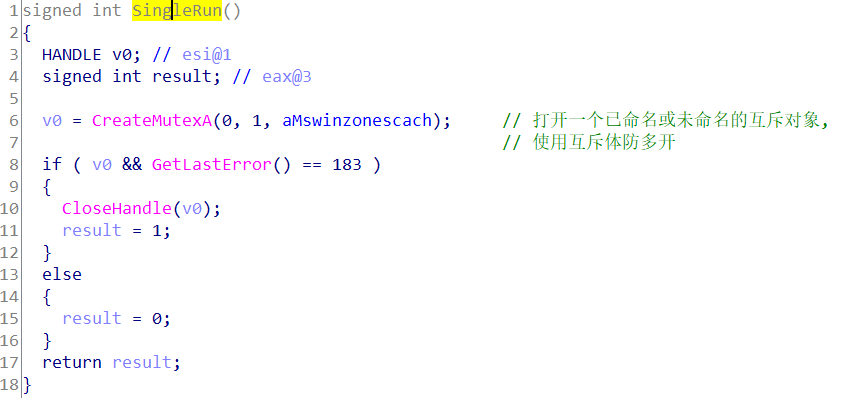






### SingleRun() 单一运行

第16行SingleRun函数使用了互斥体防止多开，确保单一实例运行。



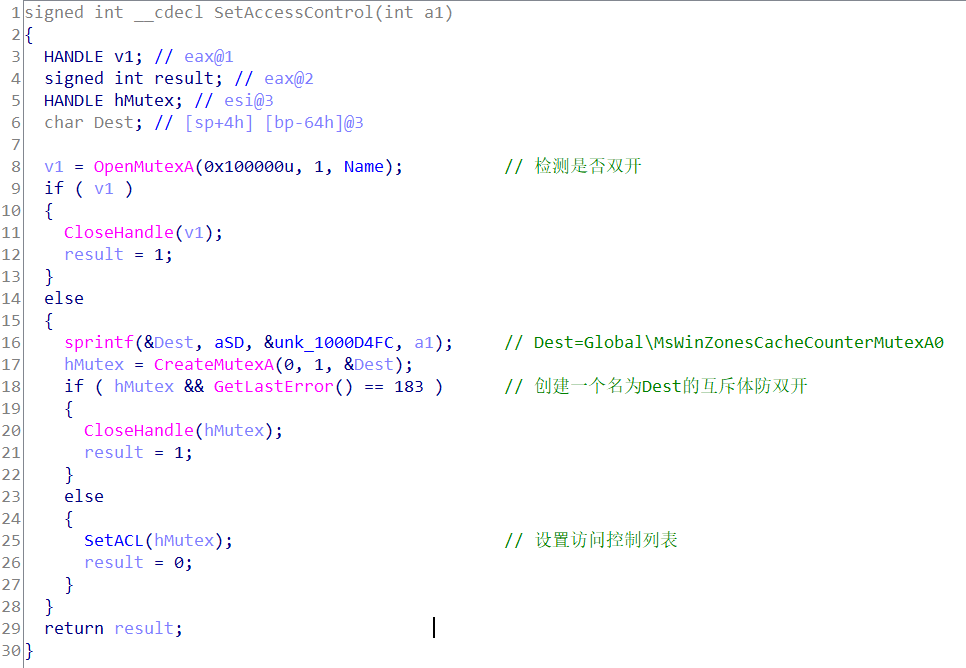
### GetUserIdAndCmp() 获取当用户SID并比较

第27行GetUserIdAndCmp函数从注册表中获取当前用户的SID，并与系统的SID做比较，返回比较的结果。



### SetAccessControl() 设置访问控制

第33行SetAccessControl函数设置了访问控制属性，互斥体防止是否双开，且设置访问控制列表。



### CheckFileExist() 检测dky文件

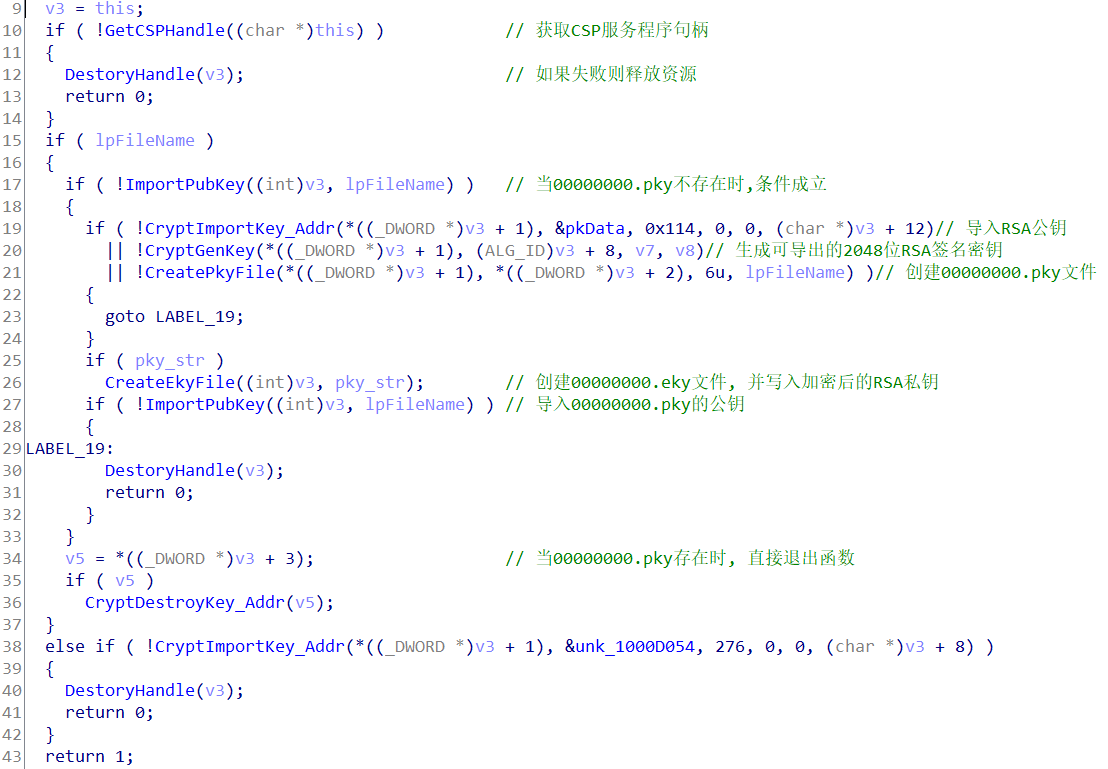
第33行CheckFileExist函数判断是否存在00000000.pky文件, 由于不存在直接返回。



### CreatePkyAndEky() 创建pky和eky文件

第48行CreatePkyAndEky函数创建了00000000.pky和00000000.eky两个文件。

前一个是公钥，后一个是加密后的私钥。分析如下：



其中，CreatePkyFile是创建00000000.pky文件并写入公钥的函数，如下图：

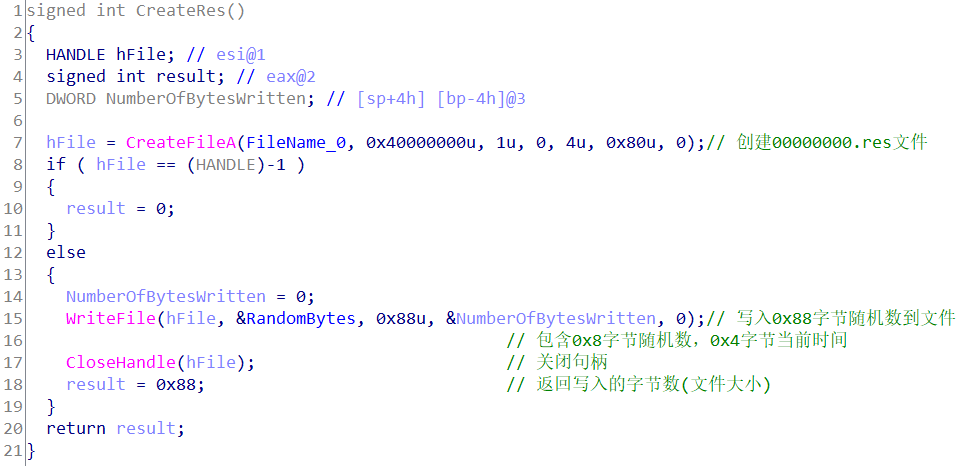


CreateEkyFile函数作用是创建00000000.eky文件, 并写入加密后的RSA私钥，如下图：



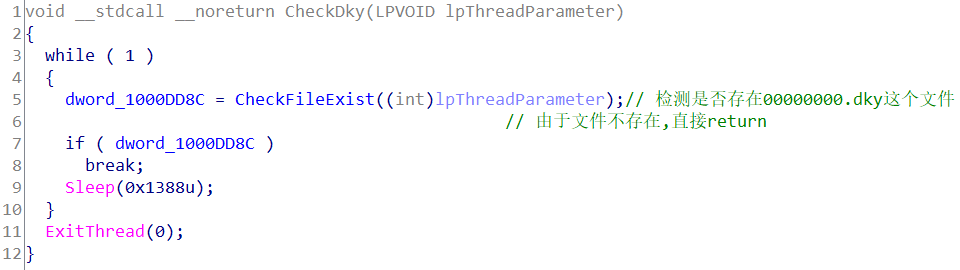
### 线程函数1：CreateResFile() 创建Res文件

第一个线程回调函数是用于创建一个00000000.res文件，写入0x88个字节数据，包含0x8字节的随机数和0x4个字节的当前时间，睡眠1s。



### 线程函数2：CheckDky() 检测Dky文件

第二个线程回调函数CheckDky，每5s检测是否存在00000000.dky文件。由于第一次运行时文件不存在，直接返回。

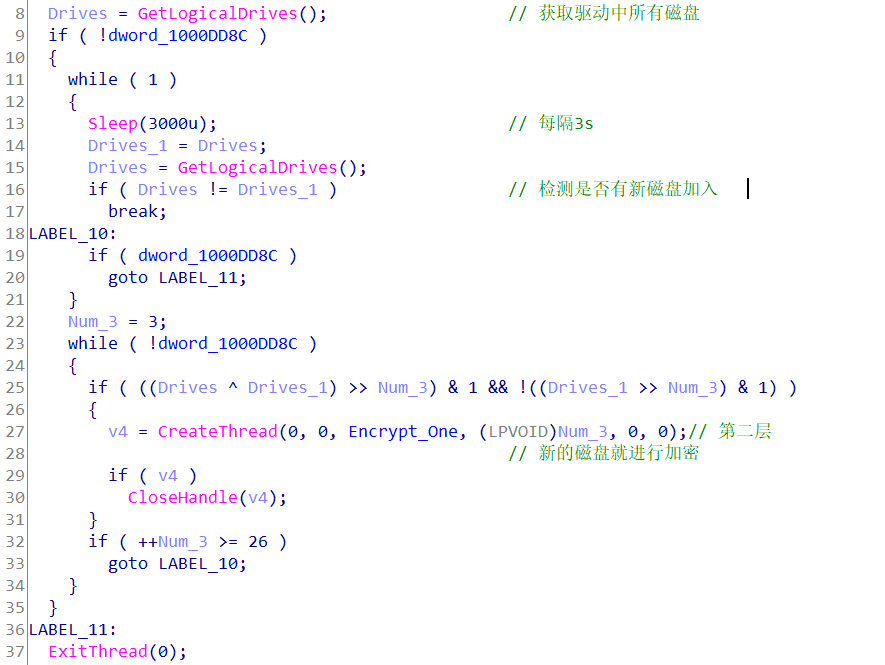


### 线程函数3：EncryptAllFiles() 加密所有文件（核心函数）

EncryptAllFiles函数是整个病毒文件最核心的函数，共嵌套了十几层函数。接下来，按层次划分，对这个函数逐一分析。

#### EncryptAllFiles() 循环检测磁盘

这一层函数循环检测是否有新的磁盘加入，如果有，则加密；否则，继续循环。



#### 第一层：Encrypt\_One() 加密函数

这一层又包含四个关键的函数。分别是MovFileToTemp、EnCrypt\_Two()、FillDisk()和ReleaseResource()。

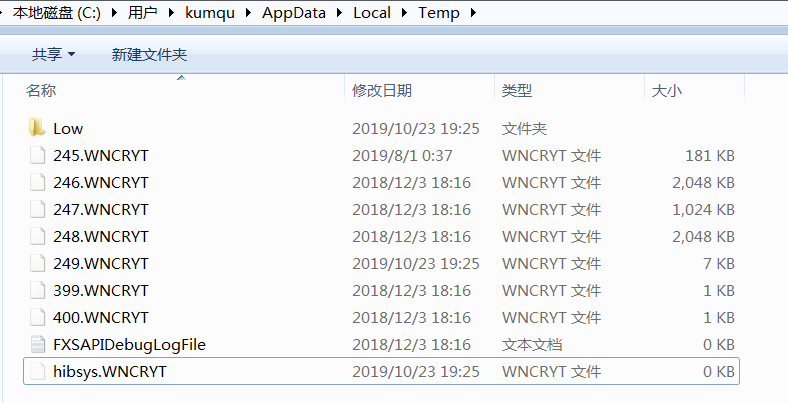
EnCrypt\_Two()函数是下一层的加密函数；ReleaseResource()函数释放临界区资源；下面分析另外两个关键函数。

##### MovFileToTemp() 移动文件到临时目录

这个函数创建一个线程，将一部分文本文件移动到临时目录下，进行重命名。



由于这些文件只是修改了后缀名而没有进行加密，因此可以通过修改后缀名进行恢复。



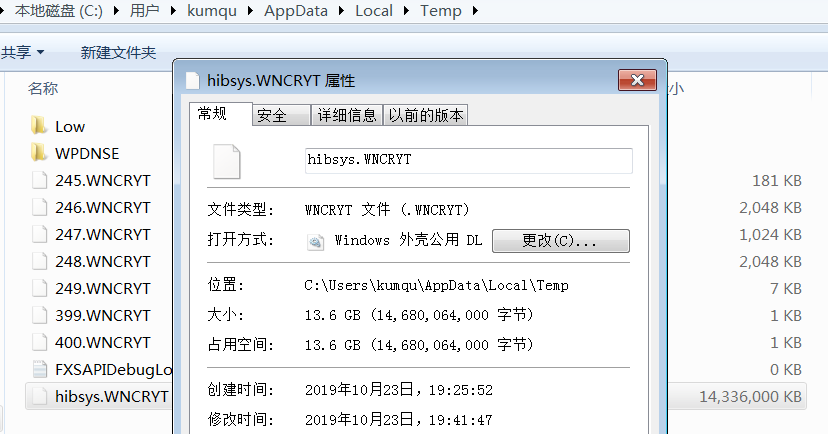
##### FillDisk() 占用回收站

FillDisk()这个函数会在回收站创建一个文件，循环写入数据直到磁盘空间不足。具体的是，会在$RECYCLE下创建一个名为hibsys.WNCRYT的文件，设置属性为隐藏，循环往这个文件里写入数据，直到磁盘空间不足跳出循环。





可能是由于磁盘空间不足，临时目录下存有一个十几G的hibsys.WNCRYT文件，而没有在回收站发现这个占用空间的文件。



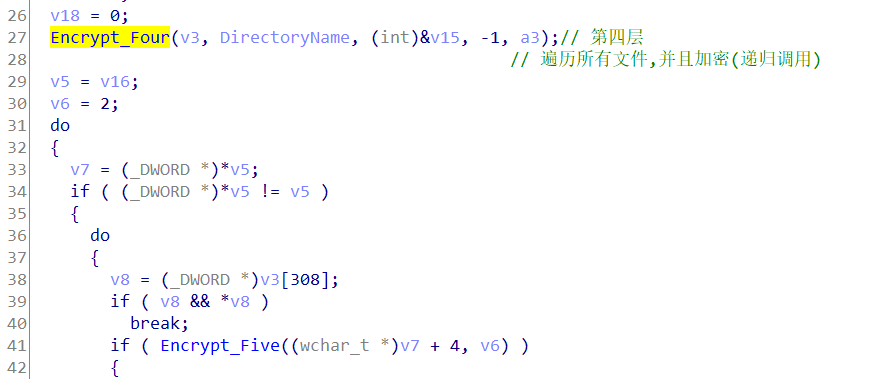
#### 第二层：Encrypt\_Two() 磁盘和路径判断

这个函数对磁盘和路径进行判断，然后继续调用下一层加密函数。



#### 第三层：Encrypt\_Three() 函数包装

这层函数分析没有发现特别的内容，只是对下一层函数的包装。



#### 第四层：Encrypt\_Four() 遍历目录和文件过滤

这个函数首先遍历了所有的文件，对不同的文件夹和文件执行不同的操作，并且根据文件后缀名进行过滤。

例如，@Please\_Read\_Me@.txt，@WanaDecryptor@.exe.lnk，@WanaDecryptor@.bmp文件就被过滤不被加密。

不同后缀的返回值对应如下：

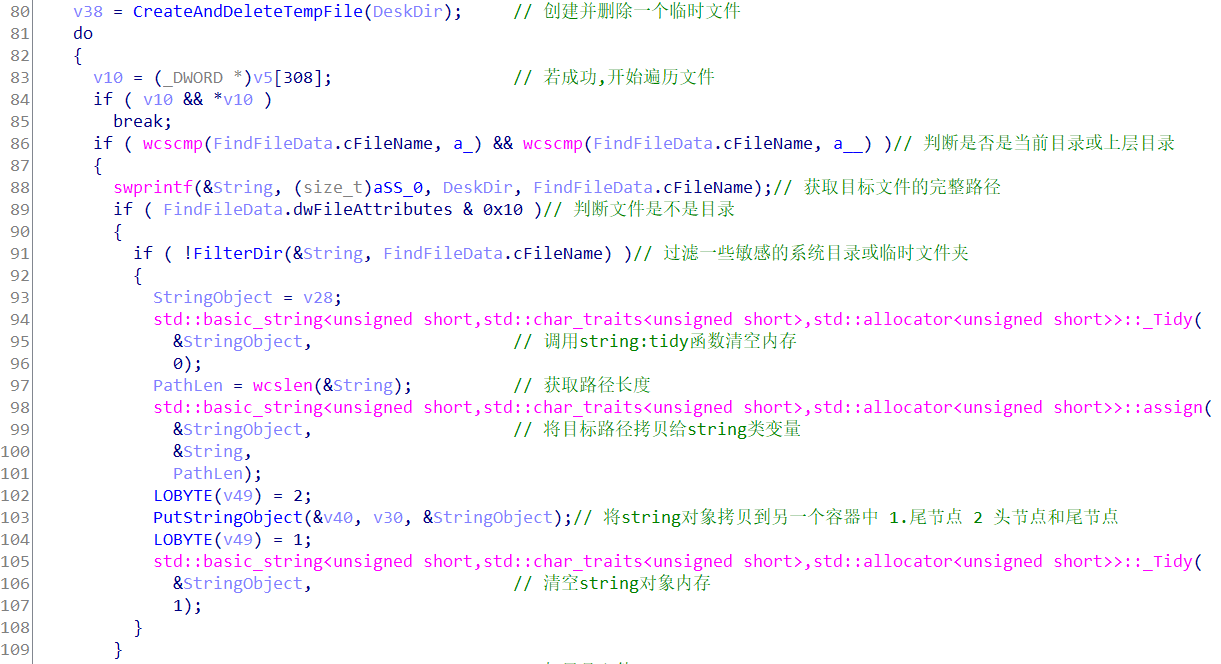
0：没有后缀或其他类型后缀

1：.exe或.dll

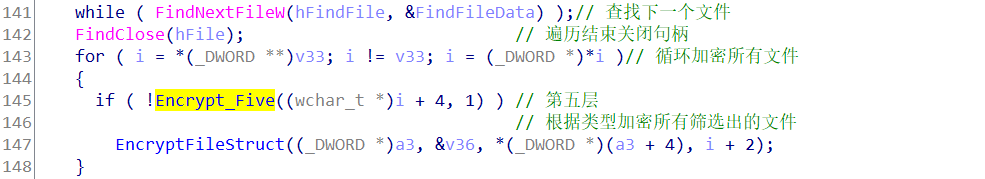
4：.WNCRYT

5：.WNCYR

6：.WNCRY







根据分析，所属文件格式为下列之一时，即加密该文件：

|  |
| --- |
| .der, .pfx, .key, .crt, .csr, .p12, .pem, .odt, .ott, .sxw, .stw, .uot, .3ds, .max, .3dm, .ods, .ots, .sxc, .stc, .dif, .slk, .wb2, .odp, .otp, .sxd, .std, .uop, .odg, .otg, .sxm, .mml, .lay, .lay6, .asc, .sqlite3, .sqlitedb, .sql, .accdb, .mdb, .dbf, .odb, .frm, .myd, .myi, .ibd, .mdf, .ldf, .sln, .suo, .cpp, .pas, .asm, .cmd, .bat, .ps1, .vbs, .dip, .dch, .sch, .brd, .jsp, .php, .asp, .java, .jar, .class, .mp3, .wav, .swf, .fla, .wmv, .mpg, .vob, .mpeg, .asf, .avi, .mov, .mp4, .3gp, .mkv, .3g2, .flv, .wma, .mid, .m3u, .m4u, .djvu, .svg, .psd, .nef, .tiff, .tif, .cgm, .raw, .gif, .png, .bmp, .jpg, .jpeg, .vcd, .iso, .backup, .zip, .rar, .tgz, .tar, .bak, .tbk, .bz2, .PAQ, .ARC, .aes, .gpg, .vmx, .vmdk, .vdi, .sldm, .sldx, .sti, .sxi, .602, .hwp, .snt, .onetoc2, .dwg, .pdf, .wk1, .wks, .123, .rtf, .csv, .txt, .vsdx, .vsd, .edb, .eml, .msg, .ost, .pst, .potm, .potx, .ppam, .ppsx, .ppsm, .pps, .pot, .pptm, .pptx, .ppt, .xltm, .xltx, .xlc, .xlm, .xlt, .xlw, .xlsb, .xlsm, .xlsx, .xls, .dotx, .dotm, .dot, .docm, .docb, .docx, .doc |

#### 第五层：Encrypt\_Five() 根据类型加密文件

这个函数先调用了GetFileType函数判断文件类型，根据返回值执行不同的加密策略。

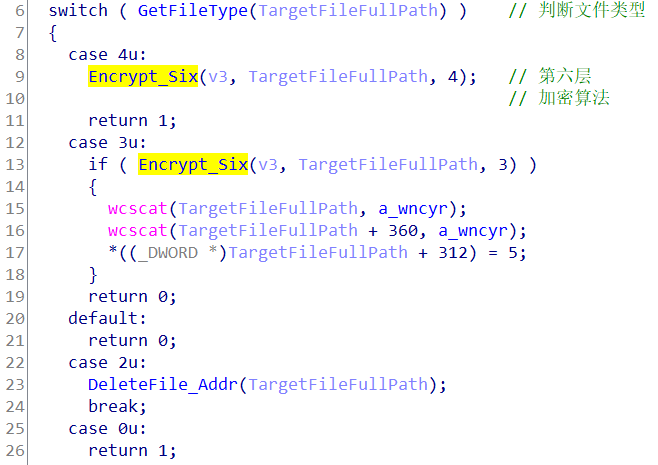
返回值1：普通文件直接加密为.WNCRY，大文件加密为.WNCYR，其他未处理的文件加入链表待处理。

返回值2：加密FILE\_TYPE\_DOCEX普通文件为.WNCRY，以及FILE\_TYPE\_DOCEX大文件为.WNCYR。所有加密过的文件移出链表

返回值3：加密链表中的所有文件，所有加密过的文件移出链表

返回值4：加密可能剩余链表中的文件

因此，虽然操作不同，但都采用了同一个加密函数Encrypt\_Six()。



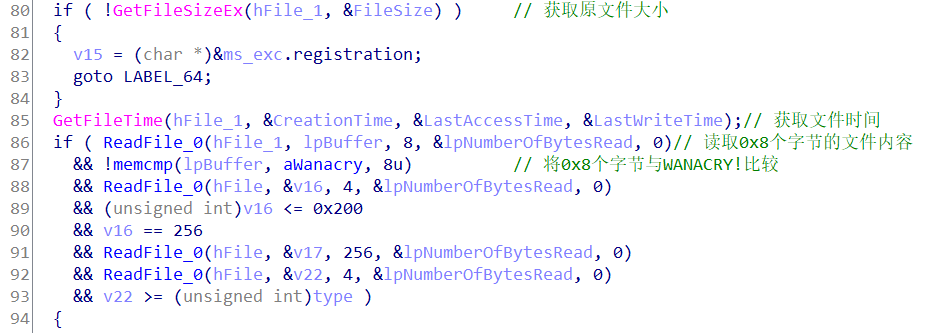
#### 第六层：Encrypt\_Six() 文件后缀判断

这个函数在加密前，获取文件后缀名，判断是否是.WNCRY文件：若是，不加密；否则，拼接字符串，调用下一层加密函数。

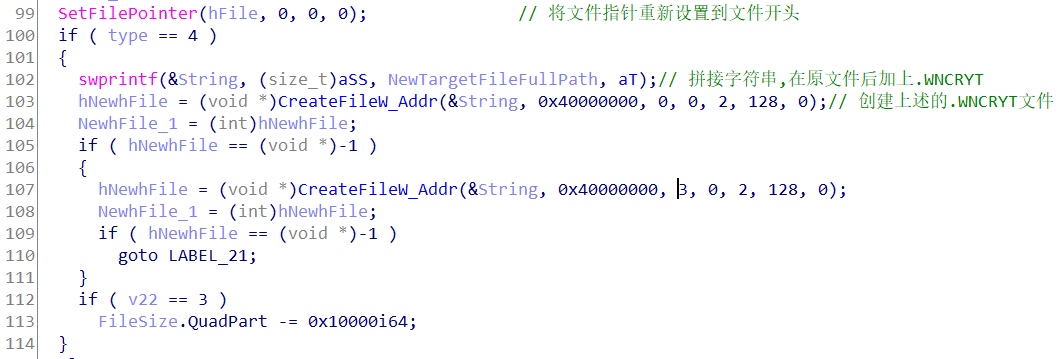


#### 第七层：Encrypt\_Seven() 文件读取

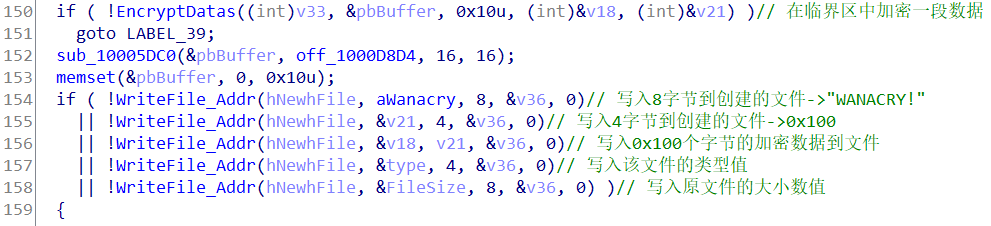
1. 读取文件前8个字节的内容，跟WANACRY!比较：



1. 使用原文件名+.WNCRYT创建一个新文件（内容暂时为空）：



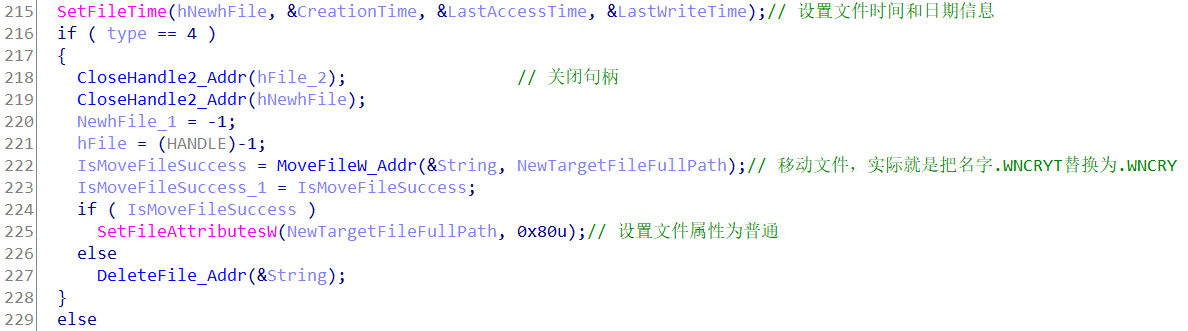
1. 写入数据到创建的文件中：



1. 读取文件，将加密后的文件内容写入到创建的文件（调用了加密函数Encrypt\_Eight）：

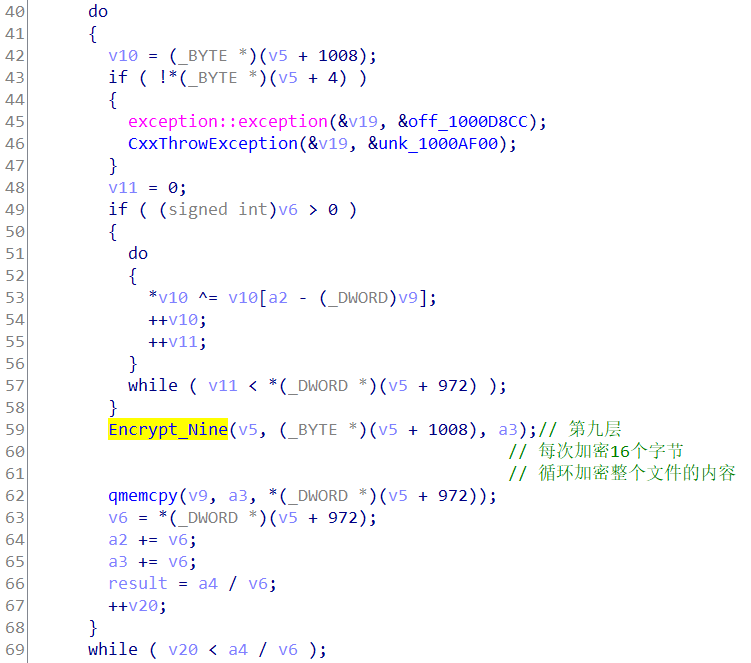


1. 修改文件后缀名，设置日期和时间，设置文件属性等：



#### 第八层：Encrypt\_Eight() 文件加密

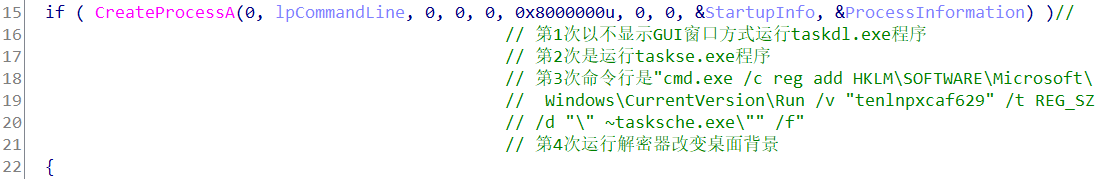
这个函数是通过解析原文件，每次输入16个字节，通过第九层具体的加密算法对文件进行加密，输出16个字节。依次循环直到加密完成，输出加密后的数据到缓存区。



下一层就是非常具体的数据加密算法了，不作分析。

### 线程函数4：RunTaskdl() 隐蔽启动taskdl.exe

这个线程回调函数每隔30s，以隐藏的方式启动taskdl.exe。其中，RunTaskdl()函数调用了CmdRunPro函数，CmdRunPro函数才是具体实现隐蔽启动的功能。下面是CmdRunPro函数的伪代码：



### 线程函数5：RunExeAndSetReg() 启动文件和注册表写入

这个函数以命令行的方式启动taskse.exe，参数为~\Desktop\ @WanaDecryptor@.exe。

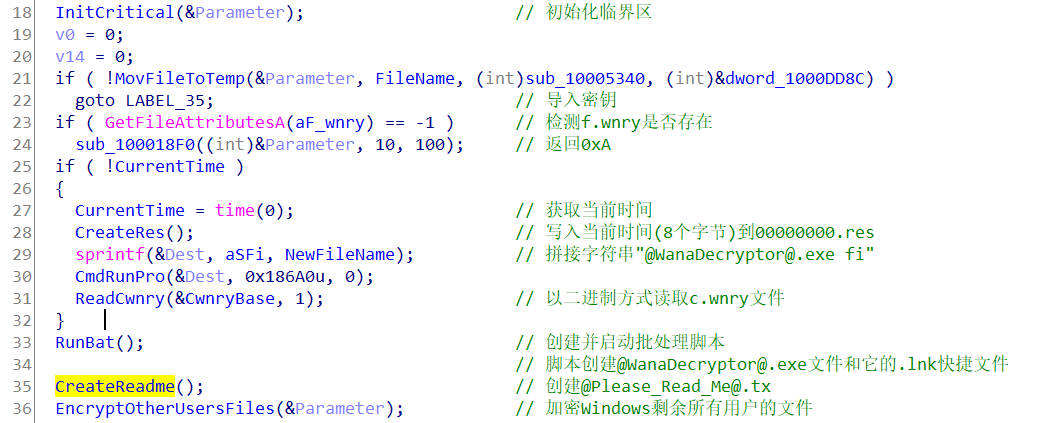
同时，以命令行的方式显式启动@WanaDecryptor@.exe 这个是病毒的解密器。最后，设置注册表项。



### RepeatOperation() 重复操作

这个函数主要功能是：创建并启动批处理脚本；创建@Please\_Read\_Me@.txt文件；

加密其他所有用户的文件。



上述功能主要由第33行~36行的三个函数实现，下面逐一分析。

#### RunBat() 启动批处理脚本

这个函数会检测@WanaDecryptor@.exe.lnk是否存在，如果不存在，就创建一个批处理脚本，并且将命令写入到.bat脚本。所以，这个脚本的作用是给@WanaDecryptor@.exe创建一个快捷方式。



#### CreateReadme() 创建@Please\_Read\_Me@.txt

这个函数检测工作路径下的@Please\_Read\_Me@.txt是否存在，如果不存在，就从r.wrny文件中读取内容，并写入到@Please\_Read\_Me@.txt 文件中。这个是病毒的勒索文档。



#### EncryptOtherUsersFiles() 加密其他用户文件

这个函数的作用是获取系统剩余所有用户的文件，如果不是当前用户，加密其他用户的桌面和文档路径下的文件。



## 6.3 taskdl.exe分析

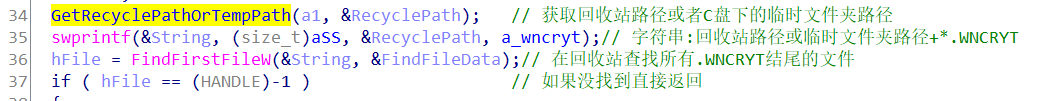
t.wnry.dll的线程函数4隐蔽启动了taskdl.exe可执行文件，因此这里对其进行分析，观察其作用。这个模块的作用是主要的作用就是删除回收站和临时目录下的.WNCRY文件。该模块的主体逻辑如下：



分析得知，这里只有一个关键函数DeleteFile，下面对其进行分析。

### DeleteFile() 删除回收站和临时目录下的.WNCRY文件

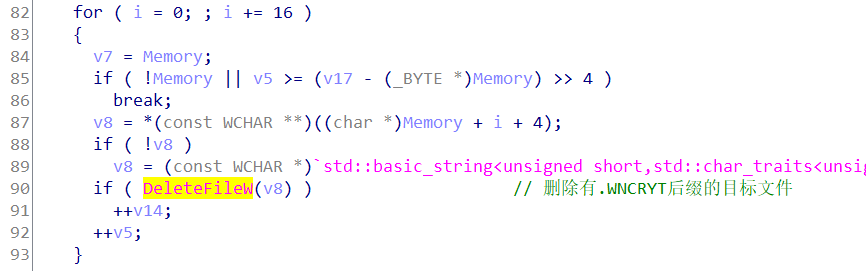
1. 首先，使用GetRecyclePathOrTempPath函数返回回收站路径和临时文件路径，如下所示：



1. 下一步，使用FindFirstFileW函数查找目标文件夹中所有.WNCRYT 结尾的文件，如果发现存在目标文件，则遍历所有的.WNCRY文件并保存到容器中：

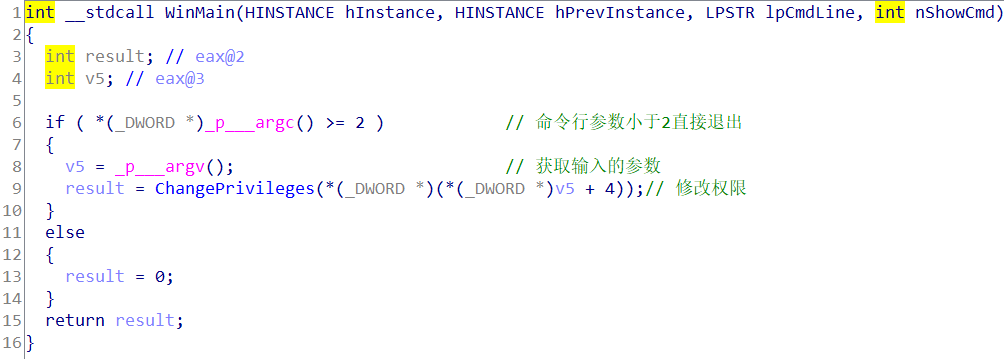


1. 最后，文件遍历结束，调用DeleteFileW函数删除所有的.WNCRY后缀文件。



## 6.4 taskse.dll分析

t.wnry.dll的线程函数5启动了taskse.exe可执行文件，这里对其进行分析，观察其作用。该函数的主体逻辑如下：

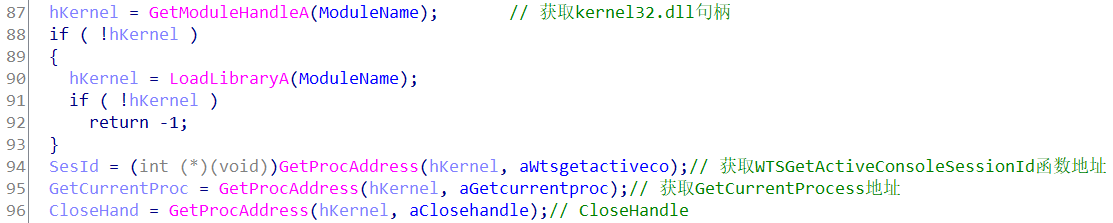


这个模块的主要是调用了一个提权函数，下面对其进行分析。

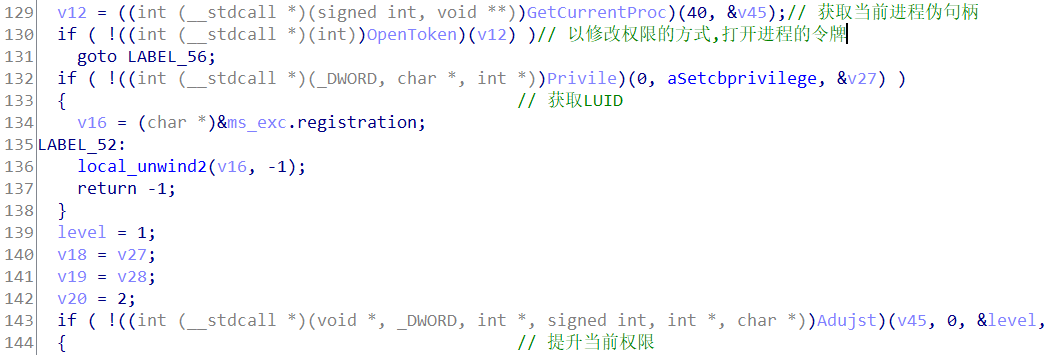
### ChangePrivileges() 修改权限

1. 首先，获取advapi32.dll、kernel32.dll、userenv.dll和wtsapi32.dll中的与权限相关的函数，如下所示：

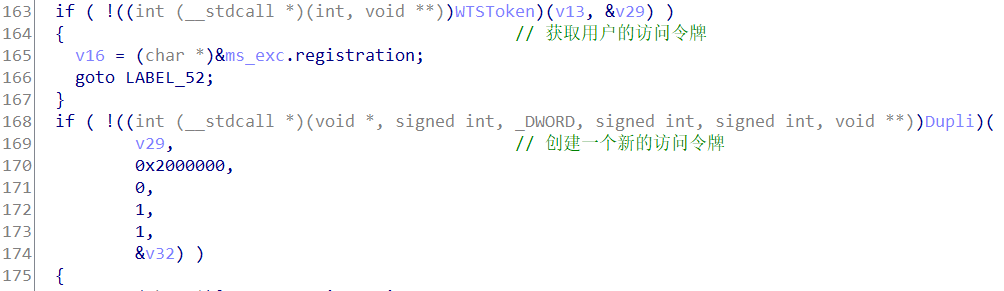




1. 获取当前进程伪句柄，以修改权限的方式,打开进程的令牌，提升权限：



1. 最后，继续获取用户的访问令牌，并创建一个新的访问令牌，再次提升权限。

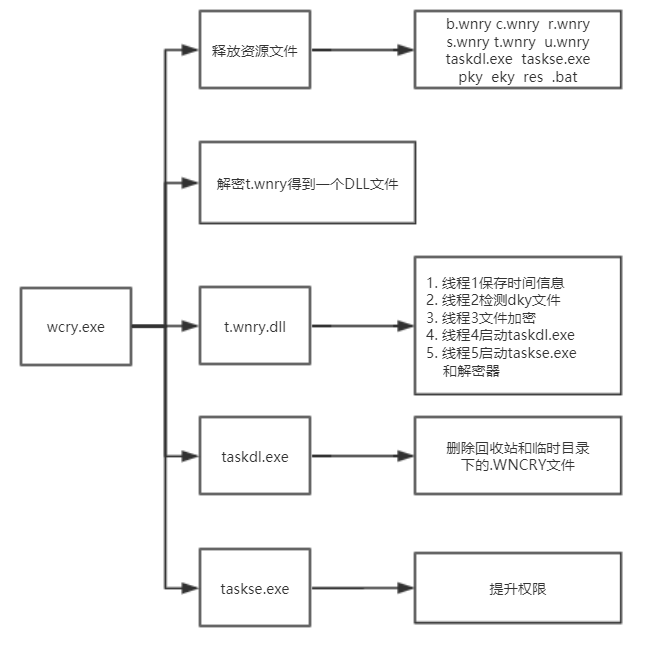


# 总结：wcry释放文件

wcry释放的文件及作用如下：

* msg：病毒的语言包。
* c.wnry：存储了三个比特币账户，包含一个下载链接，与勒索相关。
* t.wnry：隐藏了一个DLL文件，DLL文件的taskstart导出函数是病毒的核心代码
* u.wnry：解密器。
* r.wrny：勒索文档。
* @WanaDecryptor@.exe：解密器。
* taskse.exe：提升权限。
* taskdl.exe：删除临时文件和回收站的.WNCRY文件。
* 00000000.pky：加密公钥。
* 00000000.eky：被循环加密的私钥。
* 00000000.res：8字节的随机数和4字节当前时间。
* Bat：创建解密器快捷方式。

# 总结：wcry行为分析



# 解决方案

1. 打补丁：该病毒使用“永恒之蓝”漏洞进行传播的，打补丁可以阻止感染该病毒。
2. 关闭端口：该病毒利用漏洞需要445端口进行传播，关闭该端口漏洞就无法利用。
3. 创建互斥体：创建名为“MsWinZonesCacheCounterMutexA”互斥体阻止病毒运行。
4. 特征查杀：将该病毒的特征码加入病毒库，若发现即查杀。

# 参考资料

[1] <https://bbs.pediy.com/thread-252023.htm>

Wannacry勒索病毒分析

[2] <https://bbs.pediy.com/thread-249520.htm>

对WannaCry的深度分析

[3] <https://bbs.pediy.com/thread-217723.htm>

Wannacry之t.wnry核心库解密算法完全分析

[4] <https://www.antiy.com/response/wannacry.html>

安天针对勒索蠕虫“魔窟”（WannaCry）的深度分析报告