Zbot Analysis Report

(The malware sample is given by SecurityXploded Student Mentorship Programme)

Author

PHAM Ngoc Truc

Student -TELECOM PARISTECH

Email: pham@telecom-paristech.fr

Mentor

Amit Malik

Security Researcher

Email : m.amit30@gmail.com

Student Mentorship Programme

Introduction:

a893.exe is a malware of being password stealer and remote access Trojan. It provides a lot of functions for the hacker to do dirty things on the victim. Each target is being affected only one time because the malware creates a unique Mutex for each one and store it in the registry

. Besides, it has some tricks to shelter malware analyst from unpacking and does against some anti-malware tools such as: Trustee Report. Some new code segments are also decrypted in order to prevent static analysis.

Basic Static Analysis

Virustotal.com	Spyware/Win32.Zbot - Trojan-Spy.Win32.Zbot.epqg
RDG packer Detector v0.6.9 PEID, EXEINFO	Compiled by Borland Delphi V6-7 Couldn't detect its file compiler
PEView:	Address of Entry point is outside of CODE section. CODE section: raw file is 400h size compared with Virtual size which is 2B72C, and BSS section ⇨ The file is packed

Basic Dynamic Analysis

a893.exe

Create new file if icfii.exe is not existed and run it (the malware name and that folder that stored it are randomly created as I run the second time that obtain new different names)

(Malware copied a file to C:\Documents and Settings\Administrator\Application Data\Ulyliz)

Create a registry key in “HKCU\Software\Microsoft\Ihohte”

Foyvyfby = (REG_BINARY) value (that will store the malware mutex value)

Create a file C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\tmp3a1d24ea.bat which is used to delete the file itself and the malware that has been running.

MD5 of a893.exe: a893bbf7c1d45bc0532e7b336a442e22

Icfii.exe ‘s MD5: 2cb4e5b48dbd8baab02adcb61a832fbe

Their MD5 are different but they are almost the same, a little difference is revealed when using strings.exe.

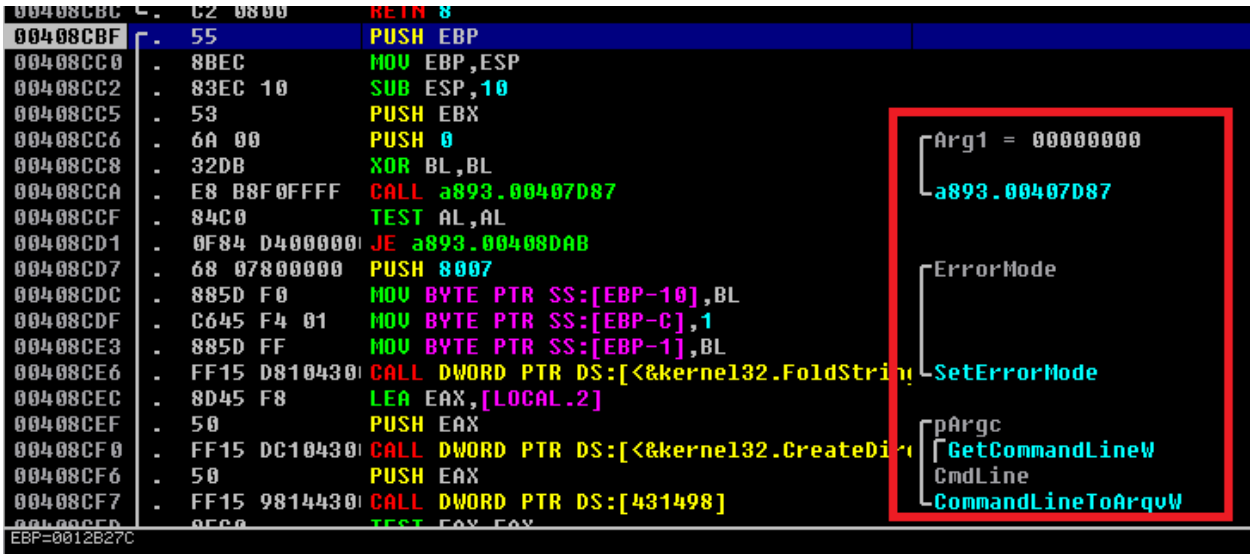
Unpacking File

When I loaded this malware into Immunity Debugger and tried running its code, at first, there were a lot of strange functions and it seemed that it was trying to load several critical dll libraries, and decrypt section code for its unpacking work.

Continuing to step through the code until I found a piece of code marking the real OEP as:

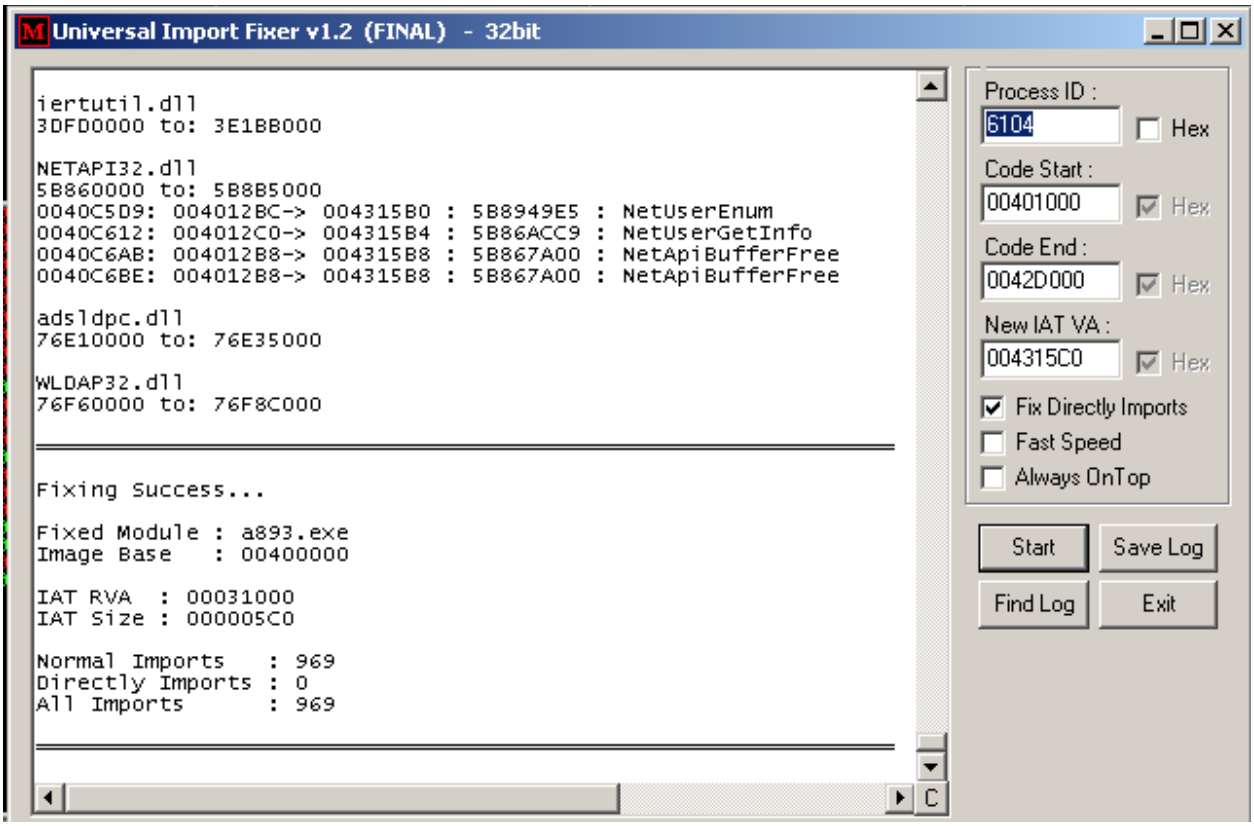
So we have the actual OEP is 8CBF (RVA) and its base address is 400000h

And the tail jump is at CC037B: JMP EDX

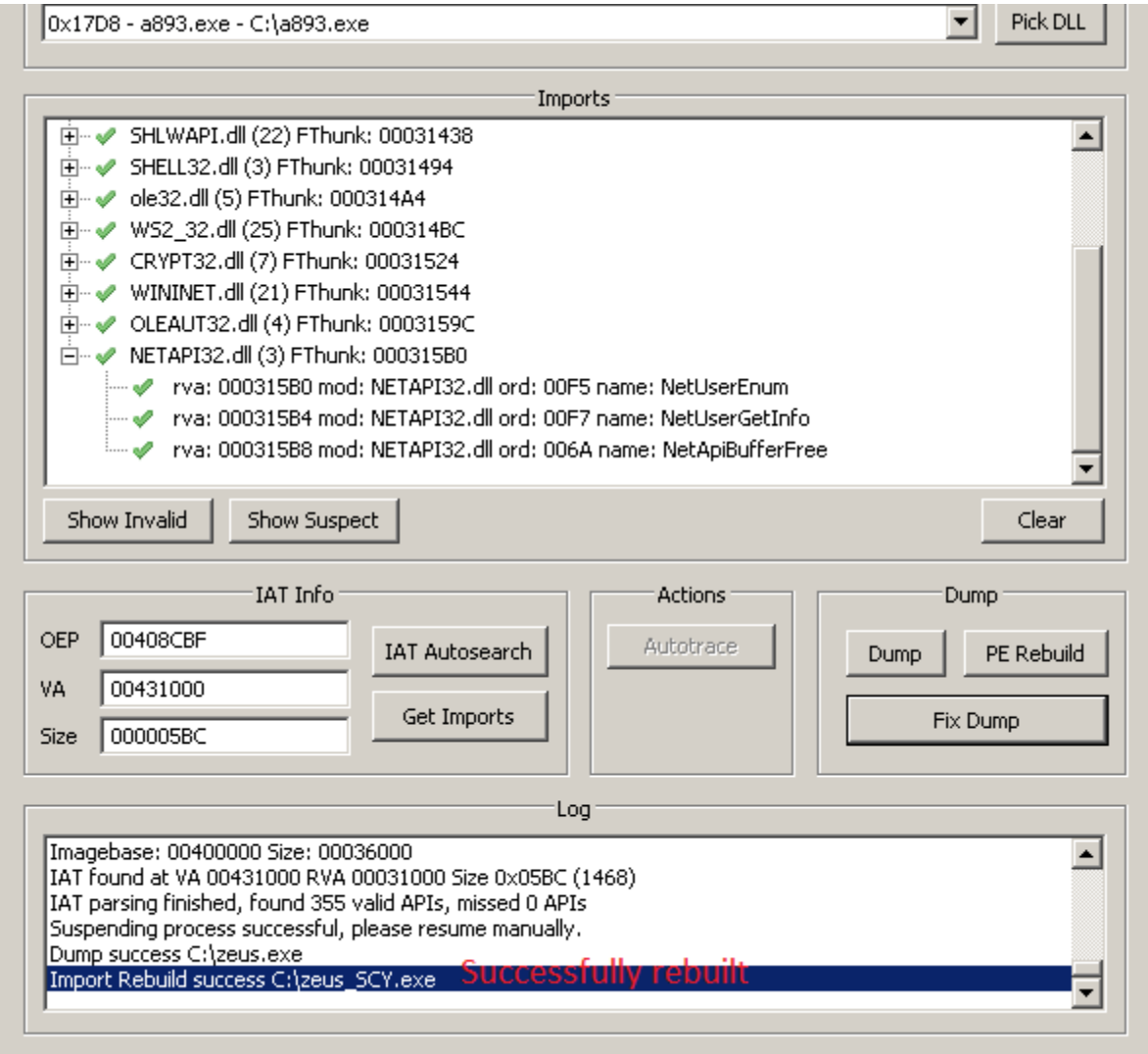


Rebuild import address table:

Using Universal Import Fixer v1.2, we filled like the figure. This tool would resolve the IAT table for us:



After that, using Scylla x86 to dump and fix import to file with new OEP: 408CBF



So now, we could start analyzing malware by using static analysis in IDA and combining it with dynamic analysis by running a893.exe file to have a better understand what the malware does.

Analysis

Dependency Walker:

Kernel32.DLL :

CreateMutexW
CreateProcessW / CreateThread
CreateRemoteThread (hook ?) / WriteProcessMemory
FindFirstFileW/FindNextFileW : find a specific file
Process32FirstW/Process32NextW: find a particular process
Thread32First/Thread32Next
WriteFile

WININET.DLL

Httpxxx functions

Internetxxx functions: This malware may connect to internet and download a web page or any other malicious internet activities.

WS2_32.DLL

It might open socket to get command from hacker.

Some encryption manipulating libraries are SECUR32.DLL, CRYPT32.DLL

ADVAPI32.DLL

Functions handle registries

Analysis on IDA pro + Ollydbg

…

Call 407D87

Which in turn calls all functions whose task is to initialize the environment, doing checking internet and file manipulation.

Especially the function sub_406CA9 : there are a lot of Httpxxx functions inside, such as: HttpSendRequesstExA, InternetReadFileExA…

Offset 407DAE: call Find_kernel32_dll

Get module of ntdll.dll:

GetprocAddress: ZwCreateThread [423930], NtCreateUserProcess (failed), ZwQueryInformationProcess[423938], LdrLoadDLL [423940], RtlUserThreadStart (failed), LdrLoadDllHandle [423944]

Offset 407F7A: call 41DA47 – check if exist a process named “Rapport util” – a program that is used to protect bank user from keylogger.

Run WSAStartup

Offset 407FB0: call 4079B6

Check the process running on Windows 64bit

Check windows version

Offset 40801D: call 407A35

Find the folder :

```
00423964                                        43 00 3A 00             C:
00423974  5C 00 44 00 6F 00 63 00 75 00 6D 00 65 00 6E 00  \Documen
00423984  74 00 73 00 20 00 61 00 6E 00 64 00 20 00 53 00  ts and S
00423994  65 00 74 00 74 00 69 00 6E 00 67 00 73 00 5C 00  ettings\
004239A4  41 00 64 00 6D 00 69 00 6E 00 69 00 73 00 74 00  Administ
004239B4  72 00 61 00 74 00 6F 00 72 00 5C 00 41 00 70 00  rator\Ap
004239C4  70 00 6C 00 69 00 63 00 61 00 74 00 69 00 6F 00  plicatio
004239D4  6E 00 20 00 44 00 61 00 74 00 61 00              n Data
```

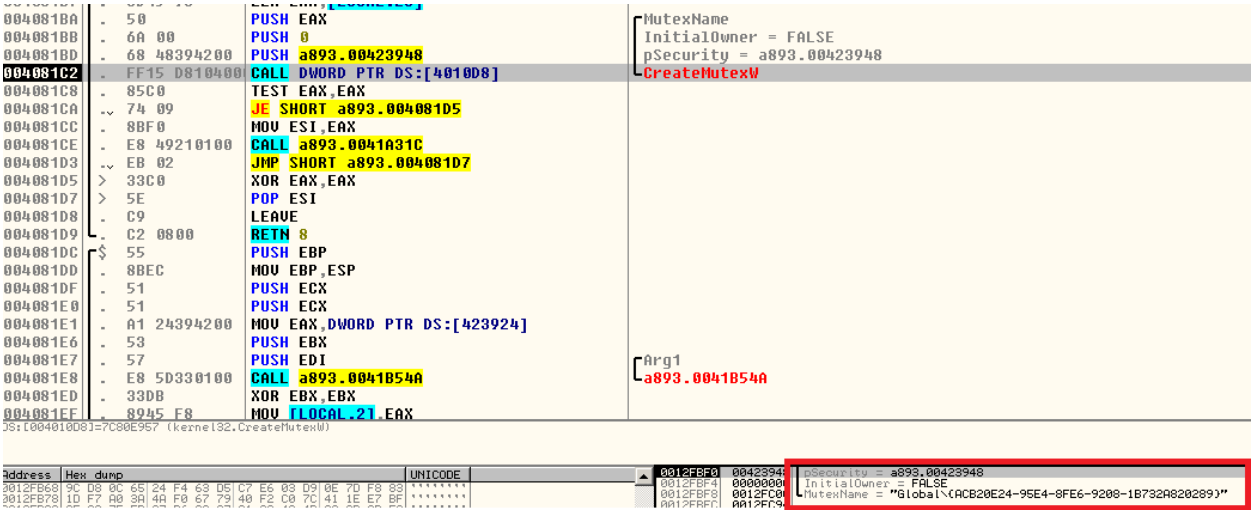
And

```
0012FBC4  43 00 3A 00 5C 00 44 00 6F 00 63 00 75 00 6D 00  C:\Docum
0012FBD4  65 00 6E 00 74 00 73 00 20 00 61 00 6E 00 64 00  ents and
```

0012FBE4 20 00 53 00 65 00 74 00 74 00 69 00 6E 00 67 00 Setting
0012FBF4 73 00 5C 00 41 00 64 00 6D 00 69 00 6E 00 69 00 s\Admini
0012FC04 73 00 74 00 72 00 61 00 74 00 6F 00 72 00 5C 00 strator\
0012FC14 44 00 65 00 73 00 6B 00 74 00 6F 00 70 00 5C 00 Desktop\
0012FC24 61 00 38 00 39 00 33 00 2E 00 65 00 78 00 65 00 a893.exe

Copy the file path to heap memory that has been just created

Offset 408AD2: Create a new its own mutex as:



Offset 408BDE : write a new code at 40BE15 at call the newly created function

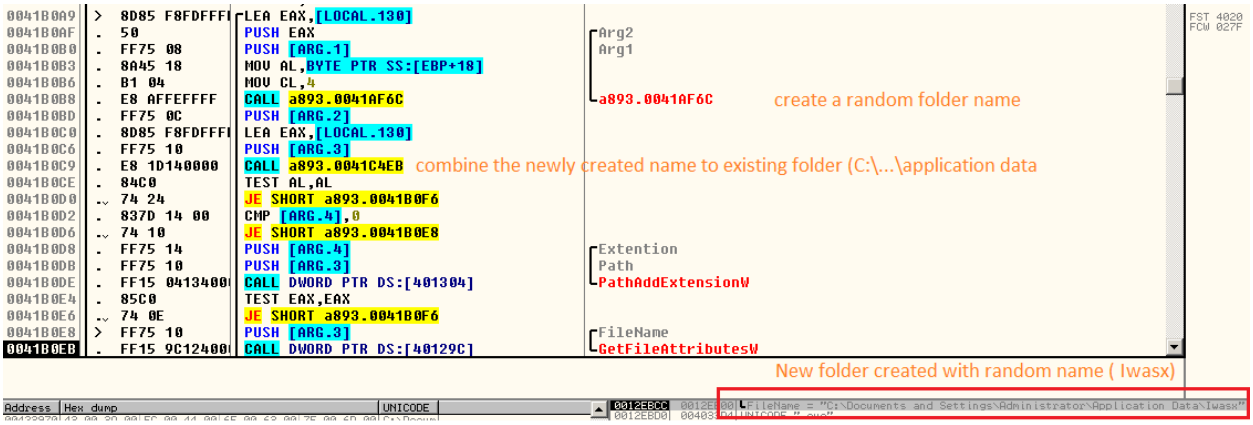
```
0040BE15  $  55            PUSH EBP
0040BE16  .  8BEC          MOV EBP,ESP
0040BE18  .  83E4 F8       AND ESP,FFFFFFF8
0040BE1B  .  81EC 3C0C000  SUB ESP,0C3C
0040BE21  .  53            PUSH EBX
0040BE22  .  56            PUSH ESI
0040BE23  .  8BD9          MOV EBX,ECX
0040BE25  .  57            PUSH EDI
0040BE26  .  53            PUSH EBX                          FileName
0040BE27  .  895424 20     MOV DWORD PTR SS:[ESP+20],EDX
0040BE2B  .  895C24 18     MOV DWORD PTR SS:[ESP+18],EBX
0040BE2F  .  C64424 17 00  MOV BYTE PTR SS:[ESP+17],0
0040BE34  .  FF15 9C12400  CALL DWORD PTR DS:[40129C]        GetFileAttributesW
0040BE3A  .  83F8 FF       CMP EAX,-1
0040BE3D  .v 75 06         JNZ SHORT a893.0040BE45
0040BE3F  .  53            PUSH EBX
0040BE40  .  E8 28040100   CALL a893.0041C26D
0040BE45  >  33F6          XOR ESI,ESI
0040BE47  .  56            PUSH ESI                          Arg3 => 00000000
0040BE48  .  8D8424 54020  LEA EAX,DWORD PTR SS:[ESP+254]
0040BE4F  .  50            PUSH EAX                          Arg2
0040BE50  .  53            PUSH EBX                          Arg1
0040BE51  .  B8 D4334000   MOV EAX,a893.004033D4             UNICODE ".exe"
0040BE56  .  E8 7FFCFFFF   CALL a893.0040BADA                a893.0040BADA
0040BE5B  .  84C0          TEST AL,AL
0040BE5D  .v 0F84 6102000  JE a893.0040C0C4
0040BE63  .  6A 01         PUSH 1                            Arg3 = 00000001
0040BE65  .  8D8424 64060  LEA EAX,DWORD PTR SS:[ESP+664]
0040BE6C  .  50            PUSH EAX                          Arg2
0040BE6D  .  53            PUSH EBX                          Arg1
0040BE6E  .  33C0          XOR EAX,EAX
0040BE70  .  E8 65FCFFFF   CALL a893.0040BADA                a893.0040BADA
0040BE75     84C0          TEST AL,AL
```

Offset 40bE56: call 40BADA

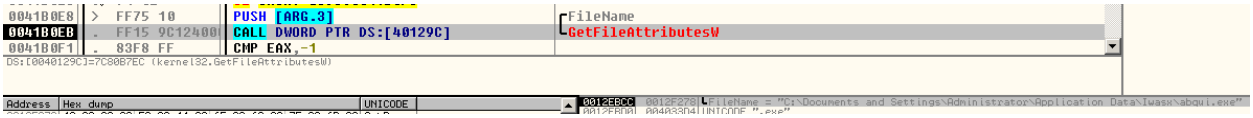
And we get:

```
0040BB05  >  6A 06         PUSH 6                            Arg5 = 00000006
0040BB07  .  53            PUSH EBX                          Arg4
0040BB08  .  8D85 ECFDFFF  LEA EAX,[LOCAL.133]
0040BB0E  .  50            PUSH EAX                          Arg3
0040BB0F  .  FF75 08       PUSH [ARG.1]                      Arg2
0040BB12  .  6A 02         PUSH 2                            Arg1 = 00000002
0040BB14  .  E8 84F50000   CALL a893.0041B09D                a893.0041B09D
0041B09D=a893.0041B09D

Address   Hex dump                                          UNICODE
00423970  43 00 3A 00 5C 00 44 00 6F 00 63 00 75 00 6D 00   C:\Docum
00423980  65 00 6E 00 74 00 73 00 20 00 61 00 6E 00 64 00   ents and
00423990  20 00 53 00 65 00 74 00 74 00 69 00 6E 00 67 00    Setting
004239A0  73 00 5C 00 41 00 64 00 6D 00 69 00 6E 00 69 00   s\Admini
004239B0  73 00 74 00 72 00 61 00 74 00 6F 00 72 00 5C 00   strator\
004239C0  41 00 70 00 70 00 6C 00 69 00 63 00 61 00 74 00   Applicat
004239D0  69 00 6F 00 6E 00 20 00 44 00 61 00 74 00 61 00   ion Data
004239E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ........

0012EDE4  00000002  Arg1 = 00000002
0012EDE8  00423970  Arg2 = 00423970
0012EDEC  0012EE00  Arg3 = 0012EE00
0012EDF0  00000000  Arg4 = 00000000
0012EDF4  00000006  Arg5 = 00000006
0012EDF8  00000000
0012EDFC  00423970  UNICODE "C:\Docume
0012EE00  7C91A3CB  RETURN to ntdll.7C
0012EE04  00140000
```

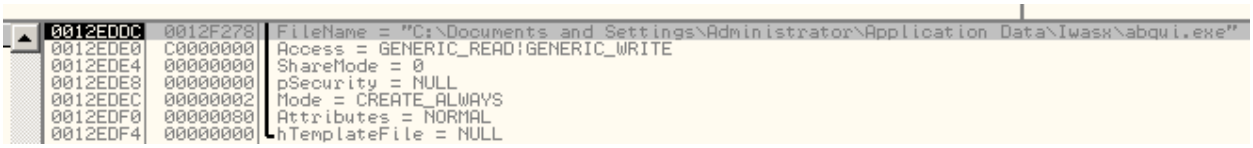
And in that function, the folder that we saw in the dynamic analysis is shown in:



Similarly using that function to crate the executable having the random name as:



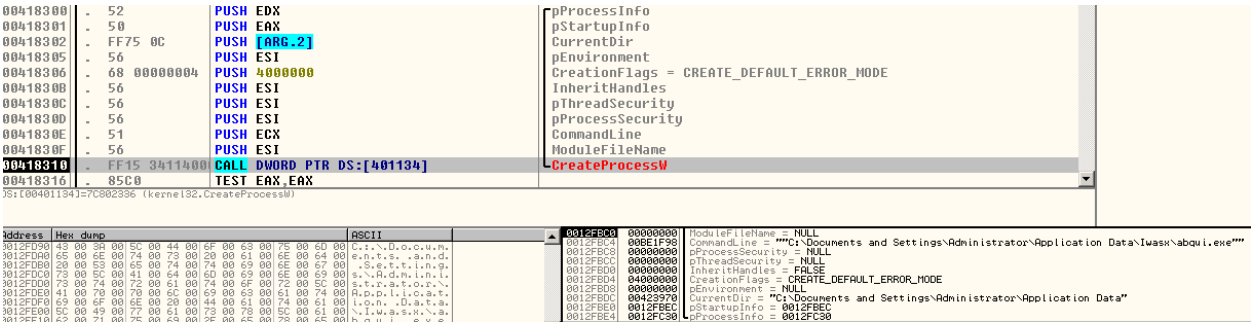
Offset 40BB76: create empty malware file



And create a new folder as before but it's storing PUT file

Open the registry : HKEY_CURRENT_USER\SOFTWARE\Microsoft and create a new subkey "Kiqixi" (random name)

Offset 40C054: call 40BBE0 : set the content of malware executable file



Offset 4185AA: call 41BE16

00BE1F98 40 65 63 68 6F 20 6F 66 66 0D 0A 3A 64 0D 0A 64 @echo off..:d..d

00BE1FA8 65 6C 20 22 43 3A 5C 44 6F 63 75 6D 65 6E 74 73 el "C:\Documents

00BE1FB8 20 61 6E 64 20 53 65 74 74 69 6E 67 73 5C 41 64 and Settings\Ad

00BE1FC8 6D 69 6E 69 73 74 72 61 74 6F 72 5C 44 65 73 6B ministrator\Desk

00BE1FD8 74 6F 70 5C 61 38 39 33 2E 65 78 65 22 0D 0A 69 top\a893.exe"..i

00BE1FE8 66 20 65 78 69 73 74 20 22 43 3A 5C 44 6F 63 75 f exist "C:\Docu

00BE1FF8 6D 65 6E 74 73 20 61 6E 64 20 53 65 74 74 69 6E ments and Settin
00BE2008 67 73 5C 41 64 6D 69 6E 69 73 74 72 61 74 6F 72 gs\Administrator
00BE2018 5C 44 65 73 6B 74 6F 70 5C 61 38 39 33 2E 65 78 \Desktop\a893.ex
00BE2028 65 22 20 67 6F 74 6F 20 64 0D 0A 64 65 6C 20 2F e" goto d..del /
00BE2038 46 20 22 43 3A 5C 44 4F 43 55 4D 45 7E 31 5C 41 F "C:\DOCUME~1\A
00BE2048 44 4D 49 4E 49 7E 31 5C 4C 4F 43 41 4C 53 7E 31 DMINI~1\LOCALS~1
00BE2058 5C 54 65 6D 70 5C 74 6D 70 34 31 37 39 39 36 35 \Temp\tmp4179965
00BE2068 32 2E 62 61 74 22 0D 0A 00 00 00 00 00 00 00 00 2.bat"........

These are what we got in the basic analysis.