jamk.fi

Reverse Engineering TTC6510-3002

Joonatan Ovaska

A K M MAHMUDUL HAQUE
AB0208

Student number: 2110841

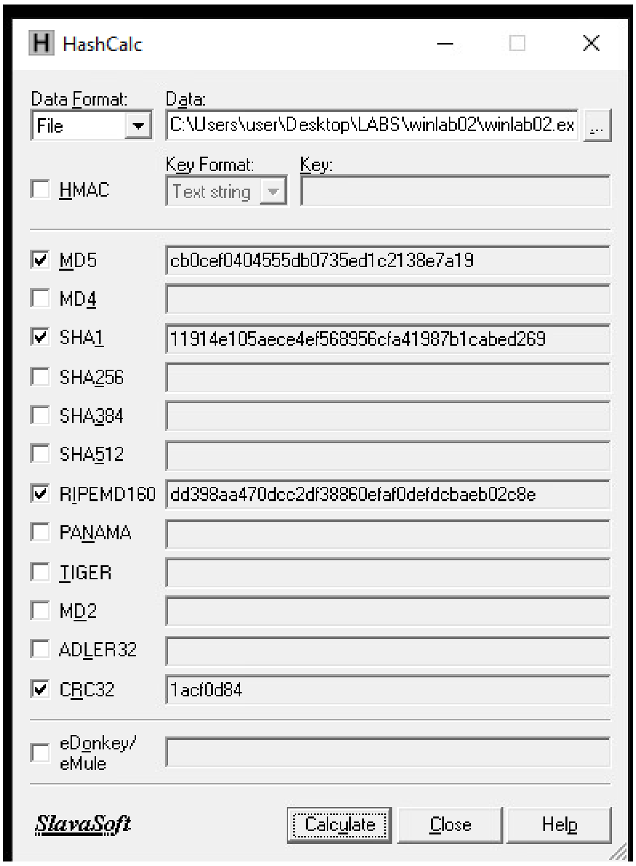
WinLab02

Date: 25.10.2023

Jyväskylän ammattikorkeakoulu
JAMK University of Applied Sciences

First Step

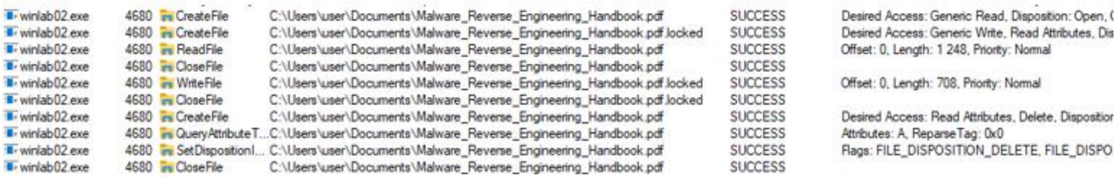
- Issue: When the winlab01.exe file is run, a file named IMPORTANT-INFORMATION.txt appears on the desktop.
- Content: The text file states that files have been locked and will be unlocked if a payment of 0.5 BTC is made. Instructions for payment are given via email.
- Encryption: The ransomware adds the .locked extension to affected files. Removing the extension won't recover the files; they remain encrypted.
- File Viewing: Encrypted files with the .locked extension can be viewed in a text editor, but the content remains scrambled.
- Scope: Only files within the user's folder are affected by this ransomware.
- Affected File Types: The ransomware targets specific file types, including .xlsx, .docx, .jpg, .png, .doc, .xls, .txt, and .pdf files. These files are encrypted and cannot be accessed without the decryption key.
- The .locked extension appears to be primarily intended to show which files are affected by the ransomware.



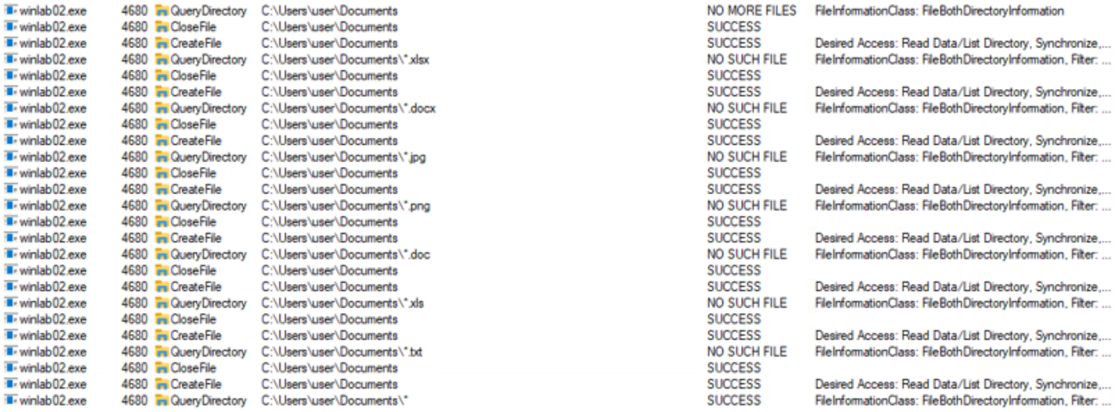
Data Format: File
Data: C:\Users\user\Desktop\LABS\winlab02\winlab02.ex
□ HMAC Key Format: Text string Key:
☑ MD5 cb0cef0404555db0735ed1c2138e7a19
□ MD4
☑ SHA1 11914e105aece4ef568956cfa41987b1cabed269
□ SHA256
□ SHA384
□ SHA512
☑ RIPEMD160 dd398aa470dcc2df38860efaf0defdcbaeb02c8e
□ PANAMA
□ TIGER
□ MD2
□ ADLER32
☑ CRC32 1acf0d84
□ eDonkey/ eMule

Second Step

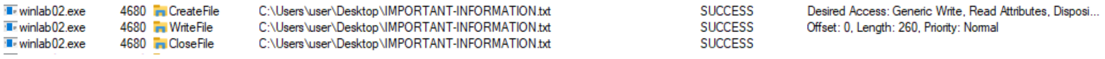
- Visibility: Using procmon, processes linked to the malware can be observed.
- Focus: Specifically, attention is given to processes involved in manipulating files.
- Relevance: These file-related processes are crucial for understanding the malware's behavior and impact on the system.



```
winlab02.exe   4680  CreateFile       C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf         SUCCESS   Desired Access: Generic Read, Disposition: Open, (
winlab02.exe   4680  CreateFile       C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf.locked  SUCCESS   Desired Access: Generic Write, Read Attributes, Dis
winlab02.exe   4680  ReadFile         C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf         SUCCESS   Offset: 0, Length: 1 248, Priority: Normal
winlab02.exe   4680  CloseFile        C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf         SUCCESS
winlab02.exe   4680  WriteFile        C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf.locked  SUCCESS   Offset: 0, Length: 708, Priority: Normal
winlab02.exe   4680  CloseFile        C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf.locked  SUCCESS
winlab02.exe   4680  CreateFile       C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf         SUCCESS   Desired Access: Read Attributes, Delete, Dispositior
winlab02.exe   4680  QueryAttributeT...C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf        SUCCESS   Attributes: A, ReparseTag: 0x0
winlab02.exe   4680  SetDispositionI... C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf       SUCCESS   Flags: FILE_DISPOSITION_DELETE, FILE_DISPO
winlab02.exe   4680  CloseFile        C:\Users\user\Documents\Malware_Reverse_Engineering_Handbook.pdf         SUCCESS
```



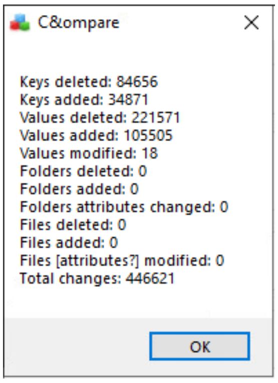
```
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents          NO MORE FILES  FileInformationClass: FileBothDirectoryInformation
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.xlsx   NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.docx   NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.jpg    NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.png    NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.doc    NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.xls    NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*.txt    NO SUCH FILE   FileInformationClass: FileBothDirectoryInformation, Filter: ...
winlab02.exe   4680  CloseFile       C:\Users\user\Documents          SUCCESS
winlab02.exe   4680  CreateFile      C:\Users\user\Documents          SUCCESS        Desired Access: Read Data/List Directory, Synchronize,...
winlab02.exe   4680  QueryDirectory  C:\Users\user\Documents\*        SUCCESS        FileInformationClass: FileBothDirectoryInformation, Filter: ...
```



```
winlab02.exe   4680  CreateFile  C:\Users\user\Desktop\IMPORTANT-INFORMATION.txt   SUCCESS  Desired Access: Generic Write, Read Attributes, Disposi...
winlab02.exe   4680  WriteFile   C:\Users\user\Desktop\IMPORTANT-INFORMATION.txt   SUCCESS  Offset: 0, Length: 260, Priority: Normal
winlab02.exe   4680  CloseFile   C:\Users\user\Desktop\IMPORTANT-INFORMATION.txt   SUCCESS
```

Third Step

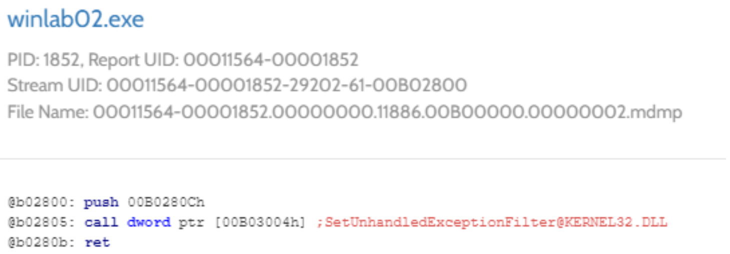
- Regshot Findings: Regshot analysis reveals that 18 registry entries were altered after executing the file.
- Observation: However, upon closer inspection of the regshot output, none of these modifications appear to be pertinent to the current issue at hand.



```
Keys deleted: 84656
Keys added: 34871
Values deleted: 221571
Values added: 105505
Values modified: 18
Folders deleted: 0
Folders added: 0
Folders attributes changed: 0
Files deleted: 0
Files added: 0
Files [attributes?] modified: 0
Total changes: 446621
```

- Exception Handling: The malware includes a noteworthy top-level exception handling feature.
- Debugging Countermeasure: This feature functions as a countermeasure against debugging attempts. It allows the malware to decide how to respond when an error occurs, a task usually handled by the operating system.
- Usual OS Behavior: Normally, operating systems take control during errors, displaying messages or terminating the program.

- Custom Handler: The malware's custom exception handler comes into play when the application operates without debugging. However, it remains inactive if the application is being debugged.



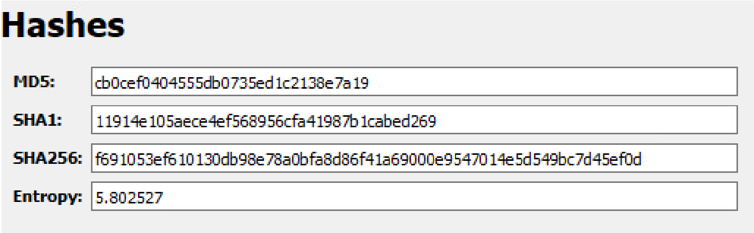
winlab02.exe

PID: 1852, Report UID: 00011564-00001852
Stream UID: 00011564-00001852-29202-61-00B02800
File Name: 00011564-00001852.00000000.11886.00B00000.00000002.mdmp

```
@b02800: push 00B0280Ch
@b02805: call dword ptr [00B03004h] ;SetUnhandledExceptionFilter@KERNEL32.DLL
@b0280b: ret
```

Fourth Step

- File Details: The winlab02.exe file is 14KB in size.
- Location: It is located in the C:\Users\user\Desktop\LABS folder on the Flare-VM machine.
- Analysis Tool: The Cutter analysis tool has been utilized to extract basic information from the file.
- MD5 Hash: Cutter indicates that the program has an MD5 hash value, but the specific hash value is not provided in the given information.



Hashes

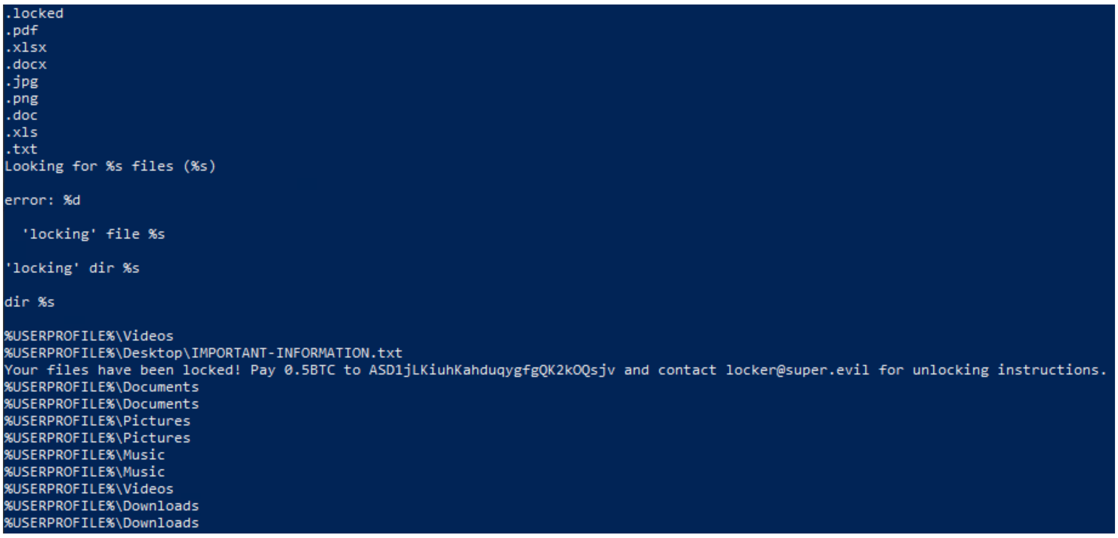
MD5: cb0cef0404555db0735ed1c2138e7a19
SHA1: 11914e105aece4ef568956cfa41987b1cabed269
SHA256: f691053ef610130db98e78a0bfa8d86f41a69000e9547014e5d549bc7d45ef0d
Entropy: 5.802527

Dependency Walker lists eleven dependencies.

Library	Description
shlwapi.dll	Shell Light-weight Utility Library. Provides various utility functions for working with shell features like shortcuts, file operations, and URLs.
kernel32.dll	Provides memory management, process and thread management, file handling, and input/output operations functionality.
vcruntime140.dll	Contains functions and resources required for running programs compiled with C++.
api-ms-win-crt-heap-l1-1-0.dll	API for managing memory allocation and deallocation in the C runtime library.
api-ms-win-crt-stdio-l1-1-0.dll	Provides functions for reading and writing data to and from the console or files.
api-ms-win-crt-string-l1-1-0.dll	Provides functions used for working with character strings.
api-ms-win-crt-filesystem-l1-1-0.dll	Provides functions for working with files and directories in the Windows file system.
api-ms-win-crt-convert-l1-1-0.dll	Provides functions to convert between different character encodings.
api-ms-win-crt-runtime-l1-1-0.dll	Provides runtime support for C and C++ programs. Includes functions for process and error handling.
api-ms-win-crt-math-l1-1-0.dll	Provides mathematical functions for use in C and C++ programs.
api-ms-win-crt-locale-l1-1-0.dll	Provides functions for working with different cultures, languages, and date/time formatting.

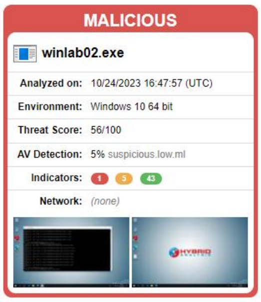
Fifth Step

- Analysis Tool: The malware can be examined for valuable information using the strings2 tool.
- In the analysis output, figure shows the filetypes that have been locked and the affected directories.
- Ransom Message: Additionally, the extracted strings reveal a message that requests a payment of 0.5 BTC in exchange for unlocking the encrypted files.



```
.locked
.pdf
.xlsx
.docx
.jpg
.png
.doc
.xls
.txt
Looking for %s files (%s)

error: %d

  'locking' file %s

'locking' dir %s

dir %s

%USERPROFILE%\Videos
%USERPROFILE%\Desktop\IMPORTANT-INFORMATION.txt
Your files have been locked! Pay 0.5BTC to ASD1jLKiuhKahduqygfgQK2kOQsjv and contact locker@super.evil for unlocking instructions.
%USERPROFILE%\Documents
%USERPROFILE%\Documents
%USERPROFILE%\Pictures
%USERPROFILE%\Pictures
%USERPROFILE%\Music
%USERPROFILE%\Music
%USERPROFILE%\Videos
%USERPROFILE%\Downloads
%USERPROFILE%\Downloads
```

- Some relevant assembly functions displayed:
  - PathCombineW
  - PathAppendW
  - StrCmpW
  - SHLWAPI .dll and so on
- MD5 Hash Lookup: Upon searching for the MD5 hash, it is confirmed that the winlab02.exe file is identified as known malware.
- Hybrid Analysis Overview: Figure 13 provides an overview of the file on the malware analysis site Hybrid Analysis. The analysis assigns the winlab02.exe file a threat score of 56/100.



MALICIOUS

winlab02.exe

Analyzed on: 10/24/2023 16:47:57 (UTC)

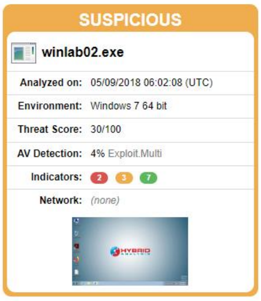
Environment: Windows 10 64 bit

Threat Score: 56/100

AV Detection: 5% suspicious.low.ml

Indicators: 1 5 43

Network: (none)



SUSPICIOUS

winlab02.exe

Analyzed on: 05/09/2018 06:02:08 (UTC)

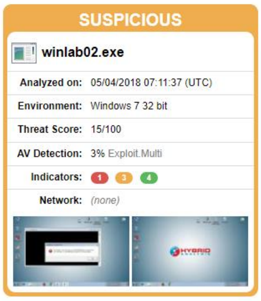
Environment: Windows 7 64 bit

Threat Score: 30/100

AV Detection: 4% Exploit.Multi

Indicators: 2 3 7

Network: (none)



SUSPICIOUS

winlab02.exe

Analyzed on: 05/04/2018 07:11:37 (UTC)

Environment: Windows 7 32 bit

Threat Score: 15/100

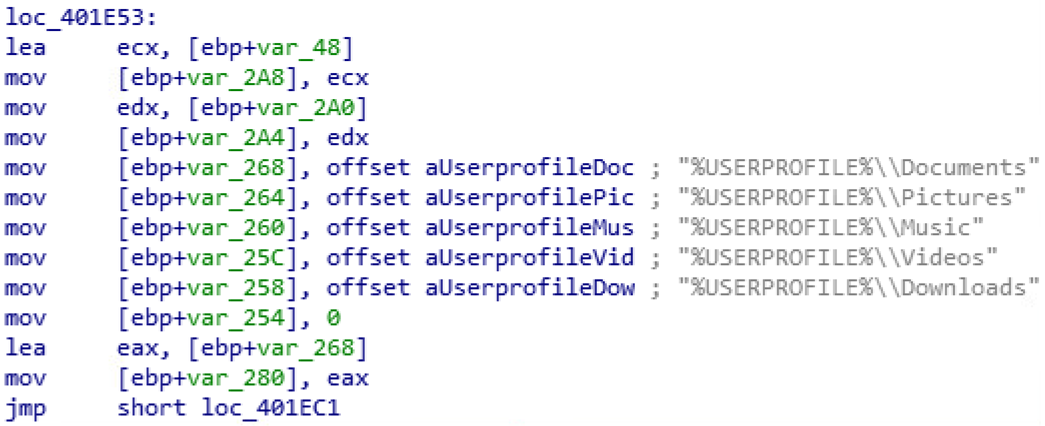
AV Detection: 3% Exploit.Multi

Indicators: 1 3 4

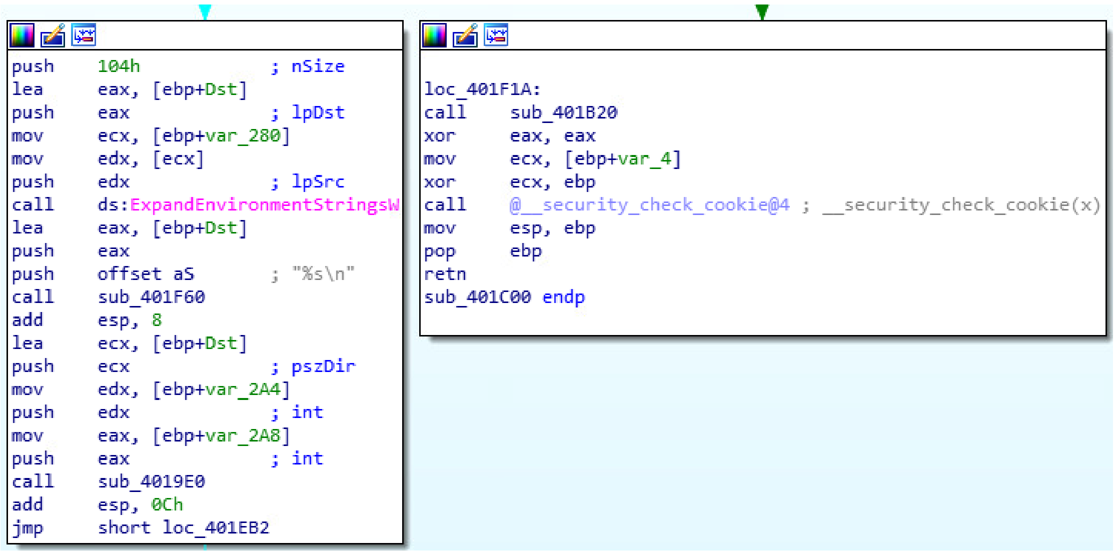
Network: (none)

Sixth Step

- Scope of Section: This section does not extensively analyze the Assembly code.
- Focus: Instead, it highlights relevant functions in connection with the earlier discussion.
- Comments: Noteworthy comments are included where essential to understanding the context and functionality of the code.

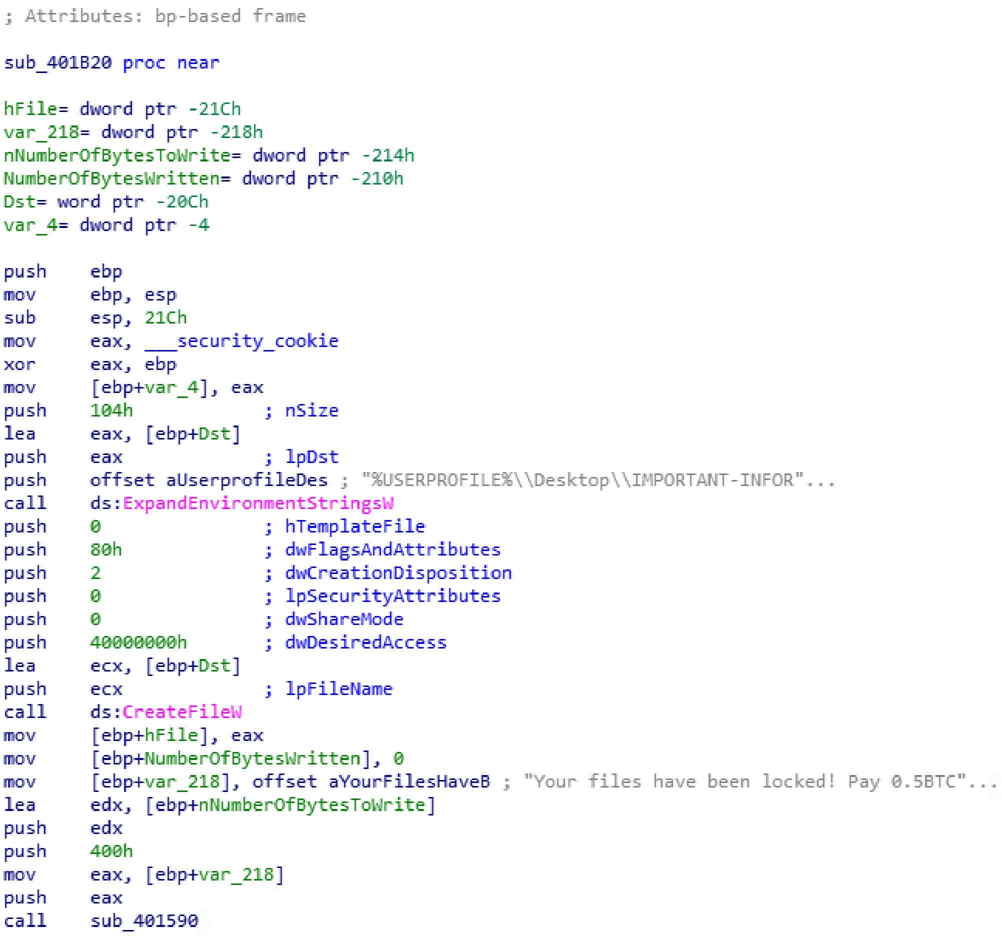


```
loc_401E53:
lea     ecx, [ebp+var_48]
mov     [ebp+var_2A8], ecx
mov     edx, [ebp+var_2A0]
mov     [ebp+var_2A4], edx
mov     [ebp+var_268], offset aUserprofileDoc ; "%USERPROFILE%\\Documents"
mov     [ebp+var_264], offset aUserprofilePic ; "%USERPROFILE%\\Pictures"
mov     [ebp+var_260], offset aUserprofileMus ; "%USERPROFILE%\\Music"
mov     [ebp+var_25C], offset aUserprofileVid ; "%USERPROFILE%\\Videos"
mov     [ebp+var_258], offset aUserprofileDow ; "%USERPROFILE%\\Downloads"
mov     [ebp+var_254], 0
lea     eax, [ebp+var_268]
mov     [ebp+var_280], eax
jmp     short loc_401EC1
```



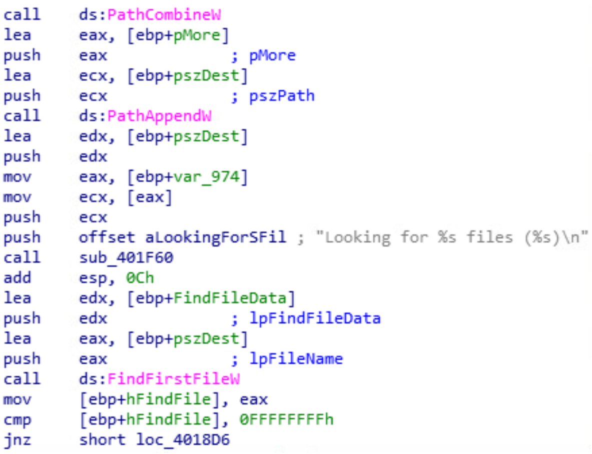
```
push    104h            ; nSize
lea     eax, [ebp+Dst]
push    eax             ; lpDst
mov     ecx, [ebp+var_280]
mov     edx, [ecx]
push    edx             ; lpSrc
call    ds:ExpandEnvironmentStringsW
lea     eax, [ebp+Dst]
push    eax
push    offset aS       ; "%s\n"
call    sub_401F60
add     esp, 8
lea     ecx, [ebp+Dst]
push    ecx             ; pszDir
mov     edx, [ebp+var_2A4]
push    edx             ; int
mov     eax, [ebp+var_2A8]
push    eax             ; int
call    sub_4019E0
add     esp, 0Ch
jmp     short loc_401EB2
```

```
loc_401F1A:
call    sub_401B20
xor     eax, eax
mov     ecx, [ebp+var_4]
xor     ecx, ebp
call    @__security_check_cookie@4 ; __security_check_cookie(x)
mov     esp, ebp
pop     ebp
retn
sub_401C00 endp
```

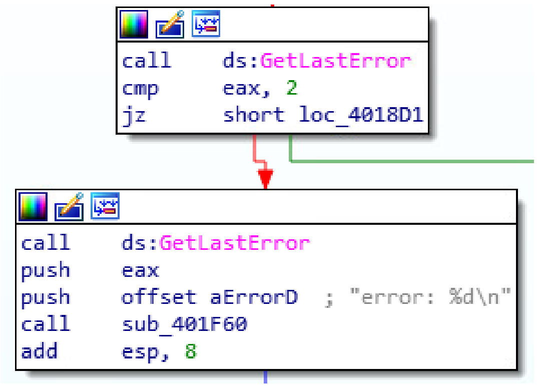


```
; Attributes: bp-based frame

sub_401B20 proc near

hFile= dword ptr -21Ch
var_218= dword ptr -218h
nNumberOfBytesToWrite= dword ptr -214h
NumberOfBytesWritten= dword ptr -210h
Dst= word ptr -20Ch
var_4= dword ptr -4

push    ebp
mov     ebp, esp
sub     esp, 21Ch
mov     eax, ___security_cookie
xor     eax, ebp
mov     [ebp+var_4], eax
push    104h            ; nSize
lea     eax, [ebp+Dst]
push    eax             ; lpDst
push    offset aUserprofileDes ; "%USERPROFILE%\\Desktop\\IMPORTANT-INFOR"...
call    ds:ExpandEnvironmentStringsW
push    0               ; hTemplateFile
push    80h             ; dwFlagsAndAttributes
push    2               ; dwCreationDisposition
push    0               ; lpSecurityAttributes
push    0               ; dwShareMode
push    40000000h       ; dwDesiredAccess
lea     ecx, [ebp+Dst]
push    ecx             ; lpFileName
call    ds:CreateFileW
mov     [ebp+hFile], eax
mov     [ebp+NumberOfBytesWritten], 0
mov     [ebp+var_218], offset aYourFilesHaveB ; "Your files have been locked! Pay 0.5BTC"...
lea     edx, [ebp+nNumberOfBytesToWrite]
push    edx
push    400h
mov     eax, [ebp+var_218]
push    eax
call    sub_401590
```

- The malware will searches for files of the specified types (see Figure 13) and append them with *.locked.*

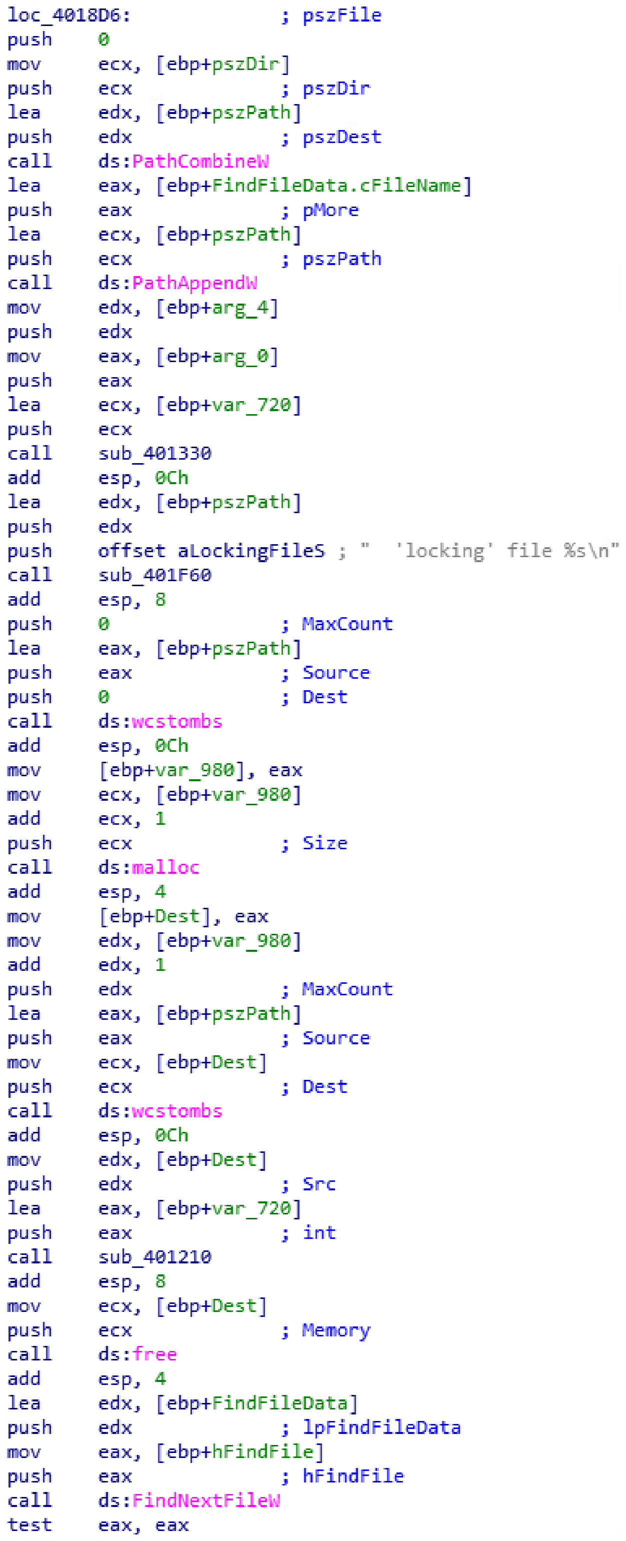


```
call    ds:PathCombineW
lea     eax, [ebp+pMore]
push    eax             ; pMore
lea     ecx, [ebp+pszDest]
push    ecx             ; pszPath
call    ds:PathAppendW
lea     edx, [ebp+pszDest]
push    edx
mov     eax, [ebp+var_974]
mov     ecx, [eax]
push    ecx
push    offset aLookingForSFil ; "Looking for %s files (%s)\n"
call    sub_401F60
add     esp, 0Ch
lea     edx, [ebp+FindFileData]
push    edx             ; lpFindFileData
lea     eax, [ebp+pszDest]
push    eax             ; lpFileName
call    ds:FindFirstFileW
mov     [ebp+hFindFile], eax
cmp     [ebp+hFindFile], 0FFFFFFFFh
jnz     short loc_4018D6
```

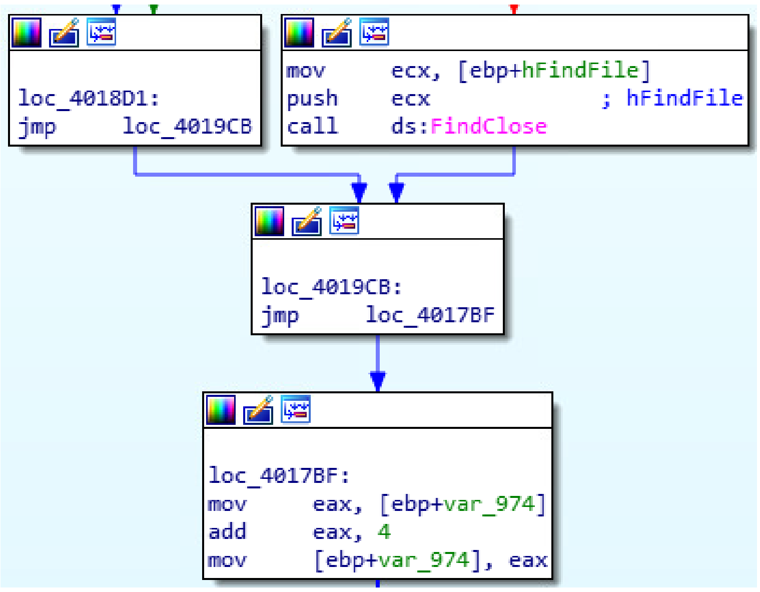


```
call    ds:GetLastError
cmp     eax, 2
jz      short loc_4018D1
```

```
call    ds:GetLastError
push    eax
push    offset aErrorD  ; "error: %d\n"
call    sub_401F60
add     esp, 8
```



```
loc_4018D6:             ; pszFile
push    0
mov     ecx, [ebp+pszDir]
push    ecx             ; pszDir
lea     edx, [ebp+pszPath]
push    edx             ; pszDest
call    ds:PathCombineW
lea     eax, [ebp+FindFileData.cFileName]
push    eax             ; pMore
lea     ecx, [ebp+pszPath]
push    ecx             ; pszPath
call    ds:PathAppendW
mov     edx, [ebp+arg_4]
push    edx
mov     eax, [ebp+arg_0]
push    eax
lea     ecx, [ebp+var_720]
push    ecx
call    sub_401330
add     esp, 0Ch
lea     edx, [ebp+pszPath]
push    edx
push    offset aLockingFileS ; "  'locking' file %s\n"
call    sub_401F60
add     esp, 8
push    0               ; MaxCount
lea     eax, [ebp+pszPath]
push    eax             ; Source
push    0               ; Dest
call    ds:wcstombs
add     esp, 0Ch
mov     [ebp+var_980], eax
mov     ecx, [ebp+var_980]
add     ecx, 1
push    ecx             ; Size
call    ds:malloc
add     esp, 4
mov     [ebp+Dest], eax
mov     edx, [ebp+var_980]
add     edx, 1
push    edx             ; MaxCount
lea     eax, [ebp+pszPath]
push    eax             ; Source
mov     ecx, [ebp+Dest]
push    ecx             ; Dest
call    ds:wcstombs
add     esp, 0Ch
mov     edx, [ebp+Dest]
push    edx             ; Src
lea     eax, [ebp+var_720]
push    eax             ; int
call    sub_401210
add     esp, 8
mov     ecx, [ebp+Dest]
push    ecx             ; Memory
call    ds:free
add     esp, 4
lea     edx, [ebp+FindFileData]
push    edx             ; lpFindFileData
mov     eax, [ebp+hFindFile]
push    eax             ; hFindFile
call    ds:FindNextFileW
test    eax, eax
```



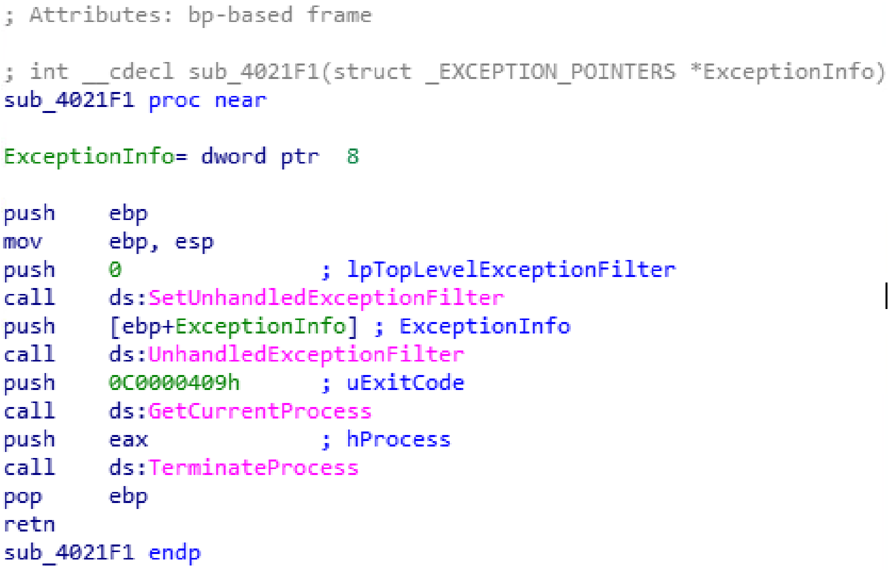
```
loc_4018D1:
jmp     loc_4019CB
```

```
mov     ecx, [ebp+hFindFile]
push    ecx             ; hFindFile
call    ds:FindClose
```

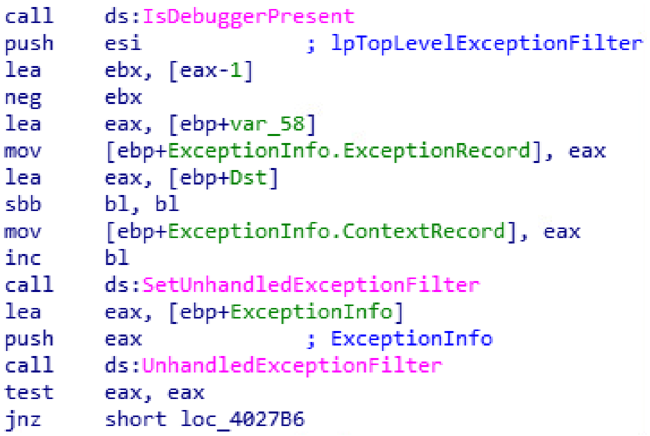
```
loc_4019CB:
jmp     loc_4017BF
```

```
loc_4017BF:
mov     eax, [ebp+var_974]
add     eax, 4
mov     [ebp+var_974], eax
```

- Anti debuggibg:



```
; Attributes: bp-based frame

; int __cdecl sub_4021F1(struct _EXCEPTION_POINTERS *ExceptionInfo)
sub_4021F1 proc near

ExceptionInfo= dword ptr  8

push    ebp
mov     ebp, esp
push    0               ; lpTopLevelExceptionFilter
call    ds:SetUnhandledExceptionFilter
push    [ebp+ExceptionInfo] ; ExceptionInfo
call    ds:UnhandledExceptionFilter
push    0C0000409h      ; uExitCode
call    ds:GetCurrentProcess
push    eax             ; hProcess
call    ds:TerminateProcess
pop     ebp
retn
sub_4021F1 endp
```



```
call    ds:IsDebuggerPresent
push    esi             ; lpTopLevelExceptionFilter
lea     ebx, [eax-1]
neg     ebx
lea     eax, [ebp+var_58]
mov     [ebp+ExceptionInfo.ExceptionRecord], eax
lea     eax, [ebp+Dst]
sbb     bl, bl
mov     [ebp+ExceptionInfo.ContextRecord], eax
inc     bl
call    ds:SetUnhandledExceptionFilter
lea     eax, [ebp+ExceptionInfo]
push    eax             ; ExceptionInfo
call    ds:UnhandledExceptionFilter
test    eax, eax
jnz     short loc_4027B6
```