Advanced Persistent Thread (APT)

Sinem Ozden – Ozan Emre Sayilir

What is APT

- Sponsored Cyber attacks targeting a specific organization to achieve a clear objective (espionage most of the time) without being detected for a long period of time.

What is APT

Advanced:

- combine multiple targeting methods, tools, and techniques to reach and compromise target
- Generally take advantage of a zero day attack that has malware payloads and uses kernel rootkits and evasion detection technologies

What is APT

Persistent:

- continuous monitoring and interaction to achieve objectives.
- not constant attacks and malware updates
- "low-and-slow"
- if the access lost, the attacker reattempt access, and most often, successfully
- goals is to maintain long-term access to the target, in contrast to threats who only access to execute a specific task

What is APT

Threat:

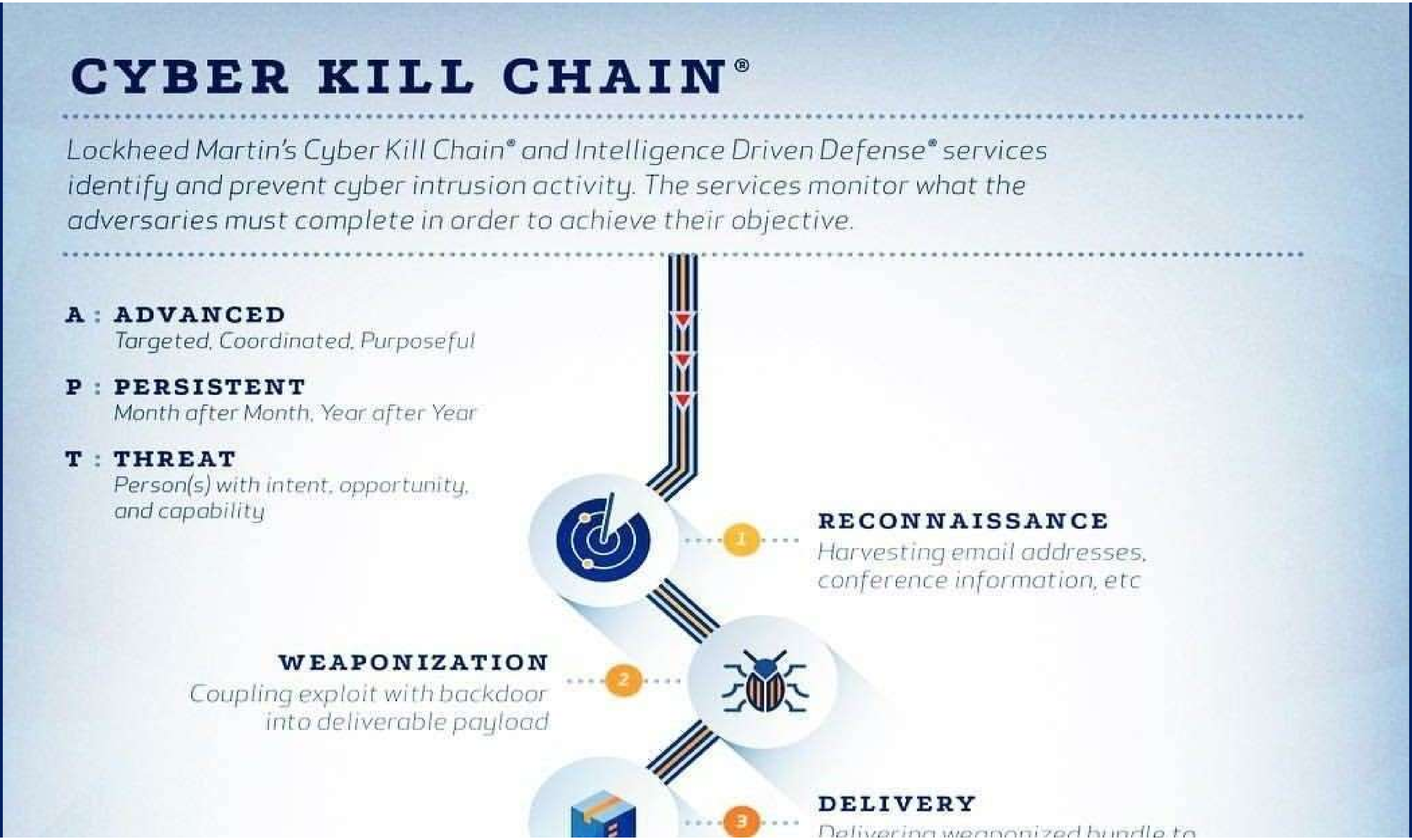
- APTs are a difficult to detect threats , until detected they can stay in their host system up to years.
- attacks are executed by coordinated human actions (sponsored by nation states or organizations that can produce similar resources)
- attackers have a specific objective and are skilled, motivated, organized and well funded.

General Goals

- Espionage.
- Intellectual property theft
- Organization Embarrassment
- As a bargaining advantage

Examples

- **GhostNet** — based in China, attacks were conducted by spear phishing emails containing malware. The group compromised computers in over 100 countries, focusing on gaining access to networks of government ministries and embassies. Attackers compromised machines inside these organizations, turned on their cameras and microphones and turned them into surveillance devices.
- **Stuxnet** — a worm used to attack Iran's nuclear program, which was delivered via an infected USB device, and inflicted damage to centrifuges used to enrich Uranium. Stuxnet is malware that targets SCADA (industrial Supervisory Control and Data Acquisition) systems – it was able to disrupt the activity of machinery in the Iranian nuclear program without the knowledge of their operators.
- **Deep Panda** — an APT attack against the US Government's Office of Personnel Management, probably originating from China. A prominent attack in 2015 was code named Deep Panda, and compromised over 4 million US personnel records, which may have included details about secret service staff.



CYBER KILL CHAIN®

Lockheed Martin's Cyber Kill Chain® and Intelligence Driven Defense® services identify and prevent cyber intrusion activity. The services monitor what the adversaries must complete in order to achieve their objective.

A : ADVANCED
Targeted, Coordinated, Purposeful

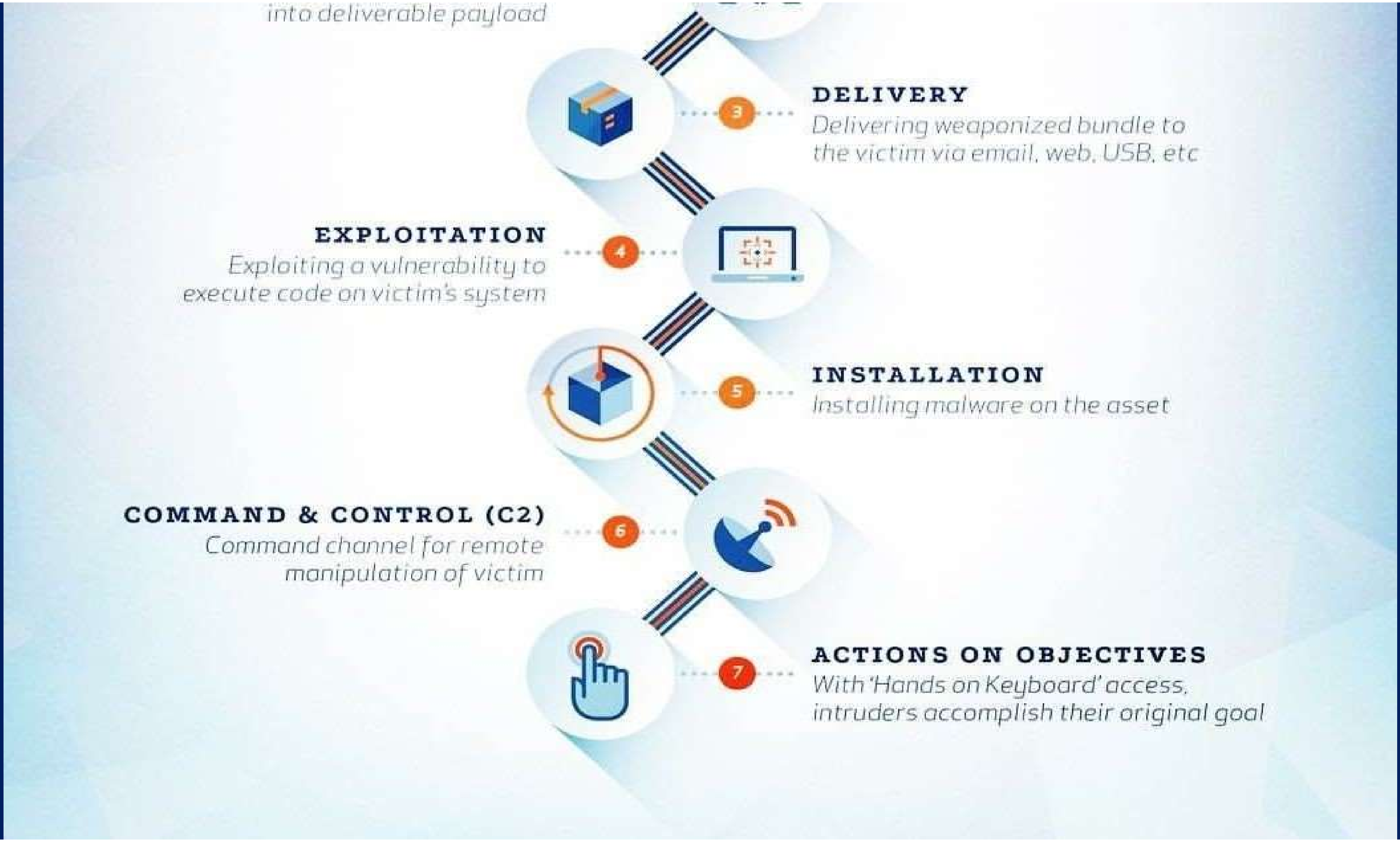
P : PERSISTENT
Month after Month, Year after Year

T : THREAT
Person(s) with intent, opportunity, and capability

1 **RECONNAISSANCE**
Harvesting email addresses, conference information, etc

2 **WEAPONIZATION**
Coupling exploit with backdoor into deliverable payload

3 **DELIVERY**
Delivering weaponized bundle to



into deliverable payload

DELIVERY

3

Delivering weaponized bundle to the victim via email, web, USB, etc

EXPLOITATION

4

Exploiting a vulnerability to execute code on victim's system

INSTALLATION

5

Installing malware on the asset

COMMAND & CONTROL (C2)

6

Command channel for remote manipulation of victim

ACTIONS ON OBJECTIVES

7

With 'Hands on Keyboard' access, intruders accomplish their original goal

Life cycle

- **Initial compromise** – social engineering,
  - Generally over email, using zero-day viruses.
  - Another popular infection method planting malware on a website that the victim's employees will be likely to visit.
- **Establish Foothold** – plant remote administration software in victim's network
  - create net backdoors and tunnels allowing stealth access to its infrastructure.
- **Escalate privileges** – use exploits and password cracking to acquire administrator privileges
  - possibly expand it to Windows domain administrator accounts.
- **Internal reconnaissance** – collect information on surrounding infrastructure
  - trust relationships, Windows domain structure.
- **Move laterally** – expand control to other workstations, servers and infrastructure elements and perform data harvesting on them.
- **Maintain presence** – ensure continued control over access channels and credentials acquired in previous steps.
- **Complete mission** – exfiltrate stolen data from victim's network.

Cost

- Among 29 different APT groups
- For Phishing
  - **$300+** cost of tool to create malicious file
  - **$ 2,500** monthly subscription fee for a service to create documents with malicious content
- A single exploit for a zeroday vulnerability
  - ***More than $1,000,000***
- Example
  - **$1.6 million** cost of the FinSpy spyware framework. Also known as FinFisher, the FinSpy framework is surveillance software able to spy on users through an infected computer's webcam and microphone, capture chat messages and emails, and steal passwords and other sensitive data.

Avoidance: Countermeasures

- Network monitoring
- Email protection - Spam filtering
- Protection against the spread of malware - Antivirus
- Intrusion detection system or intrusion protection system - Antivirus
- System and network configuration
- Security awareness
- Automatic patching

Emotet

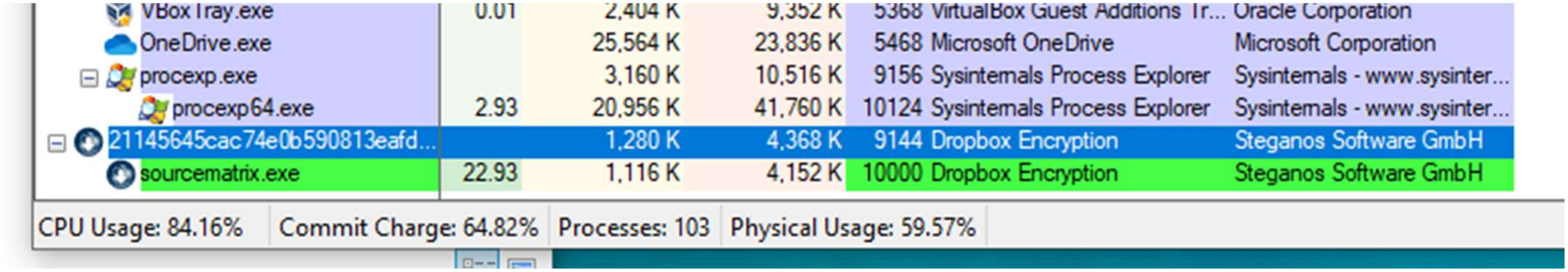
Emotet is a trojan that is primarily spread through spam emails.

During its lifecycle, it has gone through a few iterations. Early versions were delivered as a malicious JavaScript file.

Later versions evolved to use macro-enabled Office documents to retrieve a malicious payload from a C2 server.

Emotet

- The virus we selected classified as malicious after the suspicions such as:
    - **Application launched itself from a word file that was infected:** 21145645ac…
    - **Starts itself from another location:** sourcematrix.exe
    - **Executable content was dropped or overwritten:** the exe 21145645ac… deleted itself after creation of sourcematrix.exe

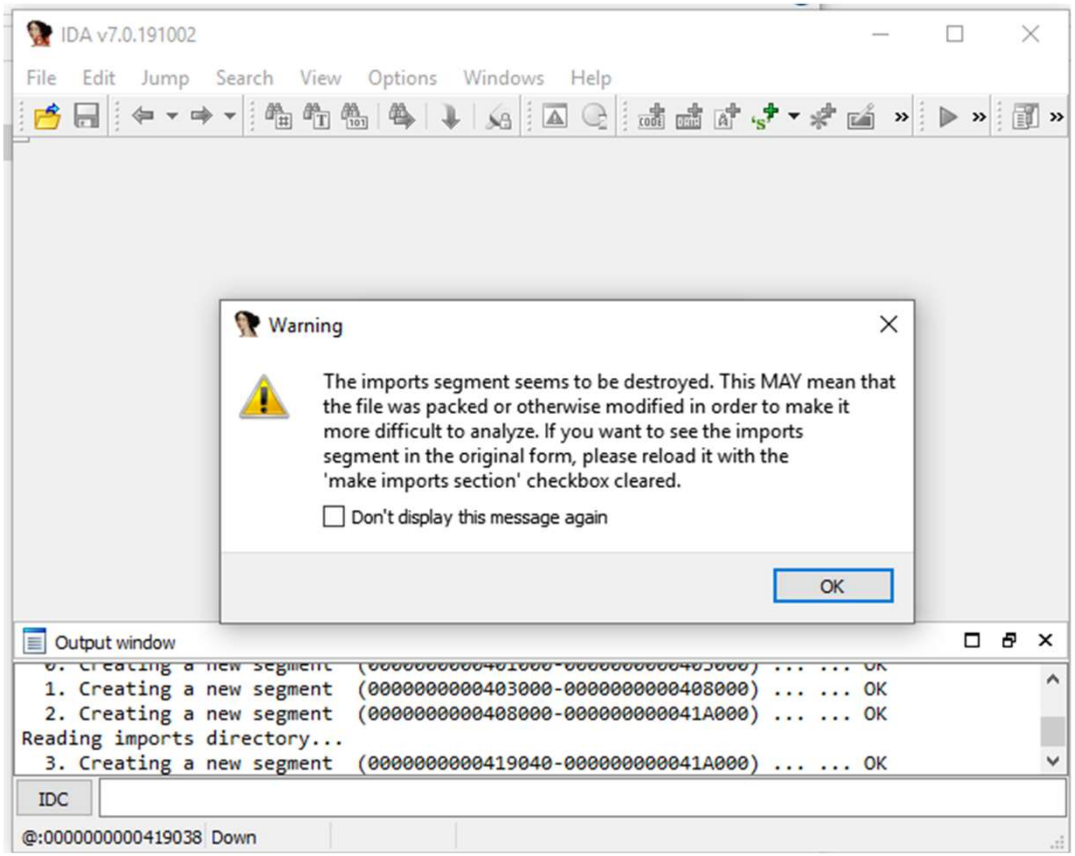


VBoxTray.exe	0.01	2,404 K	9,352 K	5368	VirtualBox Guest Additions Tr...	Oracle Corporation
OneDrive.exe		25,564 K	23,836 K	5468	Microsoft OneDrive	Microsoft Corporation
procexp.exe		3,160 K	10,516 K	9156	Sysinternals Process Explorer	Sysinternals - www.sysinter...
procexp64.exe	2.93	20,956 K	41,760 K	10124	Sysinternals Process Explorer	Sysinternals - www.sysinter...
21145645cac74e0b590813eafd...		1,280 K	4,368 K	9144	Dropbox Encryption	Steganos Software GmbH
sourcematrix.exe	22.93	1,116 K	4,152 K	10000	Dropbox Encryption	Steganos Software GmbH

CPU Usage: 84.16% Commit Charge: 64.82% Processes: 103 Physical Usage: 59.57%

Emotet

- First, we tried to obverse the virus without unpacking it, the IDA give a warning.
- Then, we decided to run the code after creating a safe zone.
- First, we could track the second exe file that was created. However, when we cut the internet the exe file disappeared from the process monitor.
- We tried again with the active internet connection .



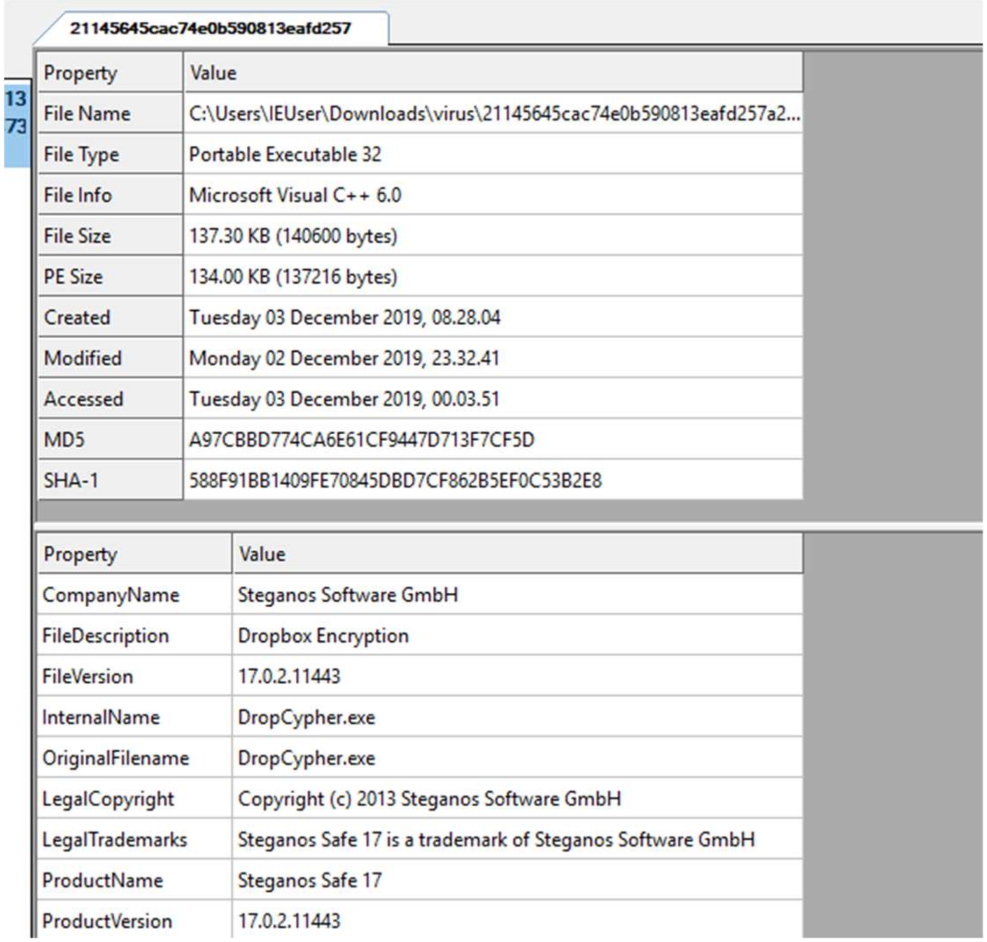
The imports segment seems to be destroyed. This MAY mean that the file was packed or otherwise modified in order to make it more difficult to analyze. If you want to see the imports segment in the original form, please reload it with the 'make imports section' checkbox cleared.

□ Don't display this message again

```
  0. Creating a new segment  (0000000000401000-0000000000403000) ... ... OK
  1. Creating a new segment  (0000000000403000-0000000000408000) ... ... OK
  2. Creating a new segment  (0000000000408000-000000000041A000) ... ... OK
Reading imports directory...
  3. Creating a new segment  (0000000000419040-000000000041A000) ... ... OK
```

sourcematrix.exe

- We then started to examine the sourcematrix.exe.
- We found information of its hash code file type and other information of its appearance.

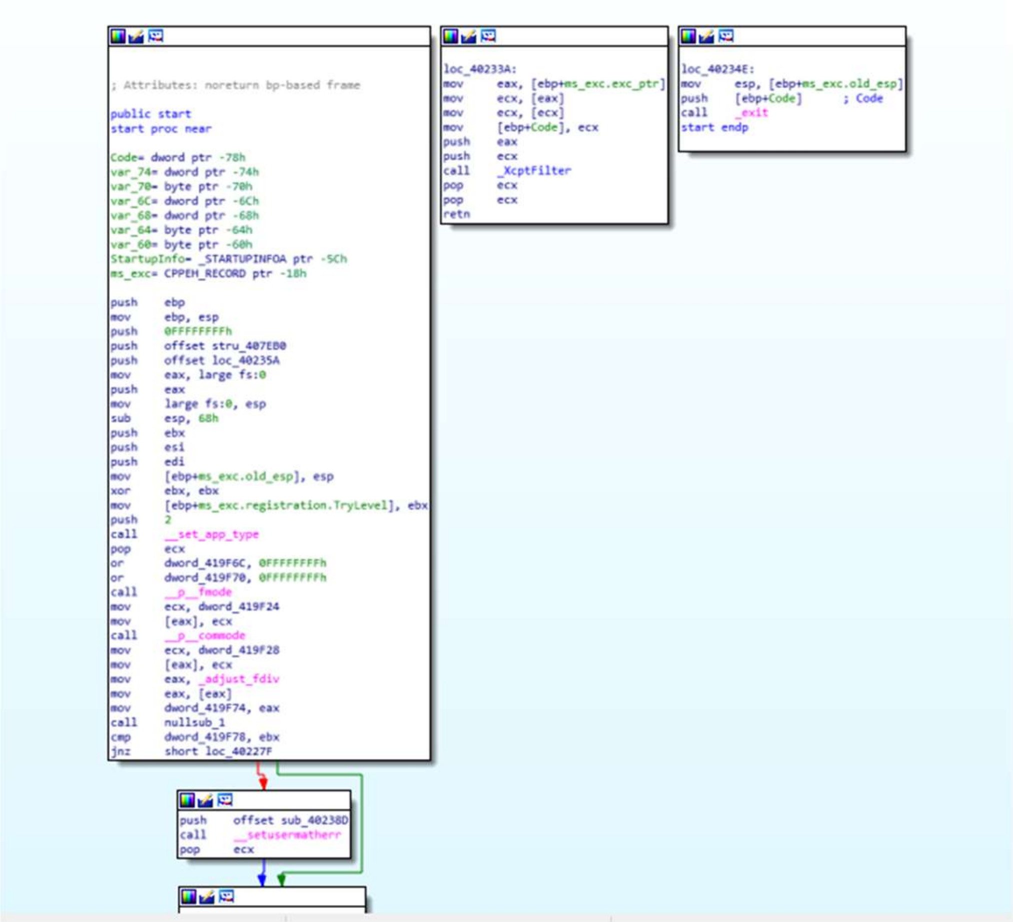


Property	Value
File Name	C:\Users\IEUser\Downloads\virus\21145645cac74e0b590813eafd257a2...
File Type	Portable Executable 32
File Info	Microsoft Visual C++ 6.0
File Size	137.30 KB (140600 bytes)
PE Size	134.00 KB (137216 bytes)
Created	Tuesday 03 December 2019, 08.28.04
Modified	Monday 02 December 2019, 23.32.41
Accessed	Tuesday 03 December 2019, 00.03.51
MD5	A97CBBD774CA6E61CF9447D713F7CF5D
SHA-1	588F91BB1409FE70845DBD7CF862B5EF0C53B2E8

Property	Value
CompanyName	Steganos Software GmbH
FileDescription	Dropbox Encryption
FileVersion	17.0.2.11443
InternalName	DropCypher.exe
OriginalFilename	DropCypher.exe
LegalCopyright	Copyright (c) 2013 Steganos Software GmbH
LegalTrademarks	Steganos Safe 17 is a trademark of Steganos Software GmbH
ProductName	Steganos Safe 17
ProductVersion	17.0.2.11443

The Code

- This is the start point of the code. We think that the separate two subroutines are the packers.
- At the near end of the process subroutine 4012d0 is called. This subroutine then calls 4012e0 subroutine. In this subroutine the suspicious and possible malicious intended code starts.

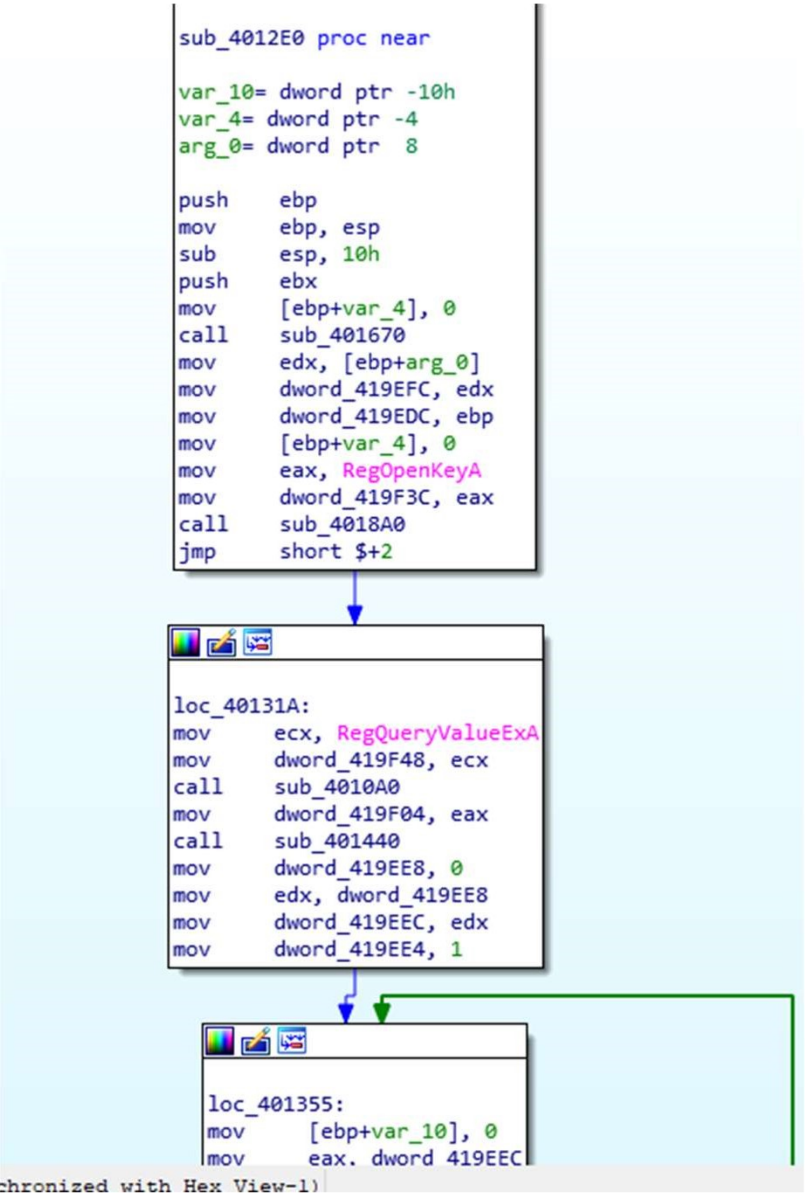


```
; Attributes: noreturn bp-based frame

public start
start proc near

Code= dword ptr -78h
var_74= dword ptr -74h
var_70= byte ptr -70h
var_6C= dword ptr -6Ch
var_68= dword ptr -68h
var_64= byte ptr -64h
var_60= byte ptr -60h
StartupInfo= _STARTUPINFOA ptr -5Ch
ms_exc= CPPEH_RECORD ptr -18h

push    ebp
mov     ebp, esp
push    0FFFFFFFFh
push    offset stru_407EB0
push    offset loc_40235A
mov     eax, large fs:0
push    eax
mov     large fs:0, esp
sub     esp, 68h
push    ebx
push    esi
push    edi
mov     [ebp+ms_exc.old_esp], esp
xor     ebx, ebx
mov     [ebp+ms_exc.registration.TryLevel], ebx
push    2
call    __set_app_type
pop     ecx
or      dword_419F6C, 0FFFFFFFFh
or      dword_419F70, 0FFFFFFFFh
call    __p__fmode
mov     ecx, dword_419F24
mov     [eax], ecx
call    __p__commode
mov     ecx, dword_419F28
mov     [eax], ecx
mov     eax, _adjust_fdiv
mov     eax, [eax]
mov     dword_419F74, eax
call    nullsub_1
cmp     dword_419F78, ebx
jnz     short loc_40227F
```

```
loc_40233A:
mov     eax, [ebp+ms_exc.exc_ptr]
mov     ecx, [eax]
mov     ecx, [ecx]
mov     [ebp+Code], ecx
push    eax
push    ecx
call    _XcptFilter
pop     ecx
pop     ecx
retn
```

```
loc_40234E:
mov     esp, [ebp+ms_exc.old_esp]
push    [ebp+Code]      ; Code
call    _exit
start endp
```

```
push    offset sub_40238D
call    __setusermatherr
pop     ecx
```



```
sub_4012E0 proc near

var_10= dword ptr -10h
var_4= dword ptr -4
arg_0= dword ptr  8

push    ebp
mov     ebp, esp
sub     esp, 10h
push    ebx
mov     [ebp+var_4], 0
call    sub_401670
mov     edx, [ebp+arg_0]
mov     dword_419EFC, edx
mov     dword_419EDC, ebp
mov     [ebp+var_4], 0
mov     eax, RegOpenKeyA
mov     dword_419F3C, eax
call    sub_4018A0
jmp     short $+2
```

```
loc_40131A:
mov     ecx, RegQueryValueExA
mov     dword_419F48, ecx
call    sub_4010A0
mov     dword_419F04, eax
call    sub_401440
mov     dword_419EE8, 0
mov     edx, dword_419EE8
mov     dword_419EEC, edx
mov     dword_419EE4, 1
```

```
loc_401355:
mov     [ebp+var_10], 0
mov     eax, dword_419EEC
```

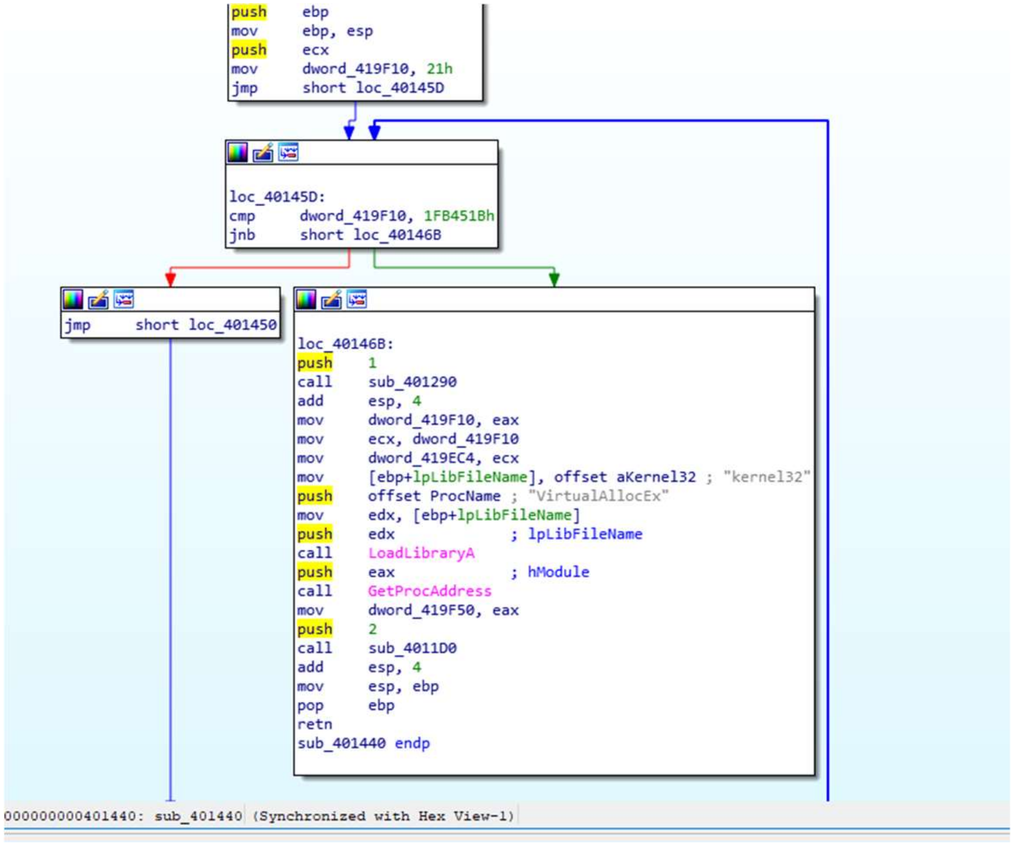
```
chronized with Hex View-1)
```

4012e0 Subroutine

- The RegOpenKeyA and RegQueryValueExA are win32 registry functions.
- These functions extracts the content of the registry keys.
- In the registry keys the information such as user preferences and settings kept.
- These functions probably used for get user information.
- This subroutine the calls 401440 subroutine to allocate a memory.

401440 Subroutine

- The VirtualAllocEx function allocates a space in memory for the program.
- In this memory that allocated information which was copied from a specific address is kept.
- Then the program jumps to instructions that extracts the address spaces of process.



```
push    ebp
mov     ebp, esp
push    ecx
mov     dword_419F10, 21h
jmp     short loc_40145D
```

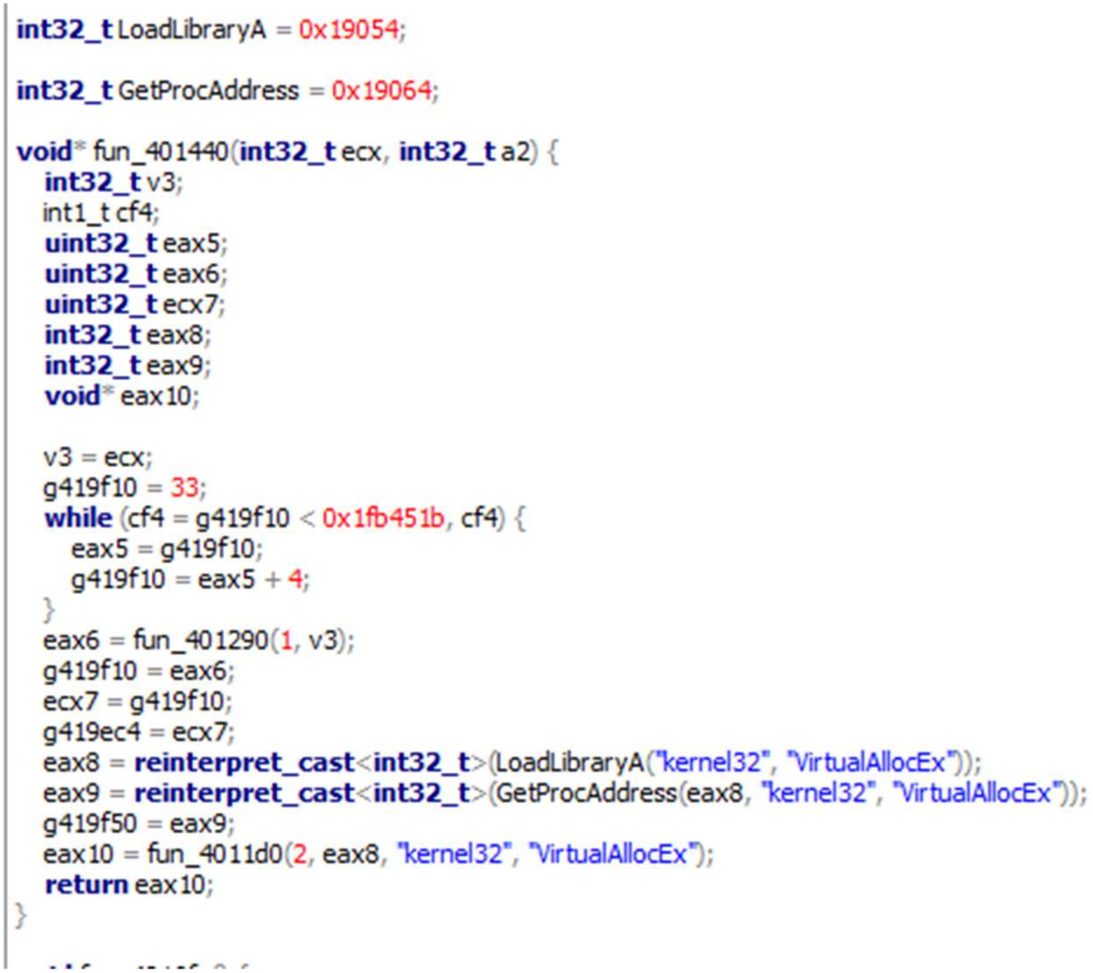
```
loc_40145D:
cmp     dword_419F10, 1FB451Bh
jnb     short loc_40146B
```

```
jmp     short loc_401450
```

```
loc_40146B:
push    1
call    sub_401290
add     esp, 4
mov     dword_419F10, eax
mov     ecx, dword_419F10
mov     dword_419EC4, ecx
mov     [ebp+lpLibFileName], offset aKernel32 ; "kernel32"
push    offset ProcName ; "VirtualAllocEx"
mov     edx, [ebp+lpLibFileName]
push    edx             ; lpLibFileName
call    LoadLibraryA
push    eax             ; hModule
call    GetProcAddress
mov     dword_419F50, eax
push    2
call    sub_4011D0
add     esp, 4
mov     esp, ebp
pop     ebp
retn
sub_401440 endp
```

```
00000000401440: sub_401440 (Synchronized with Hex View-1)
```

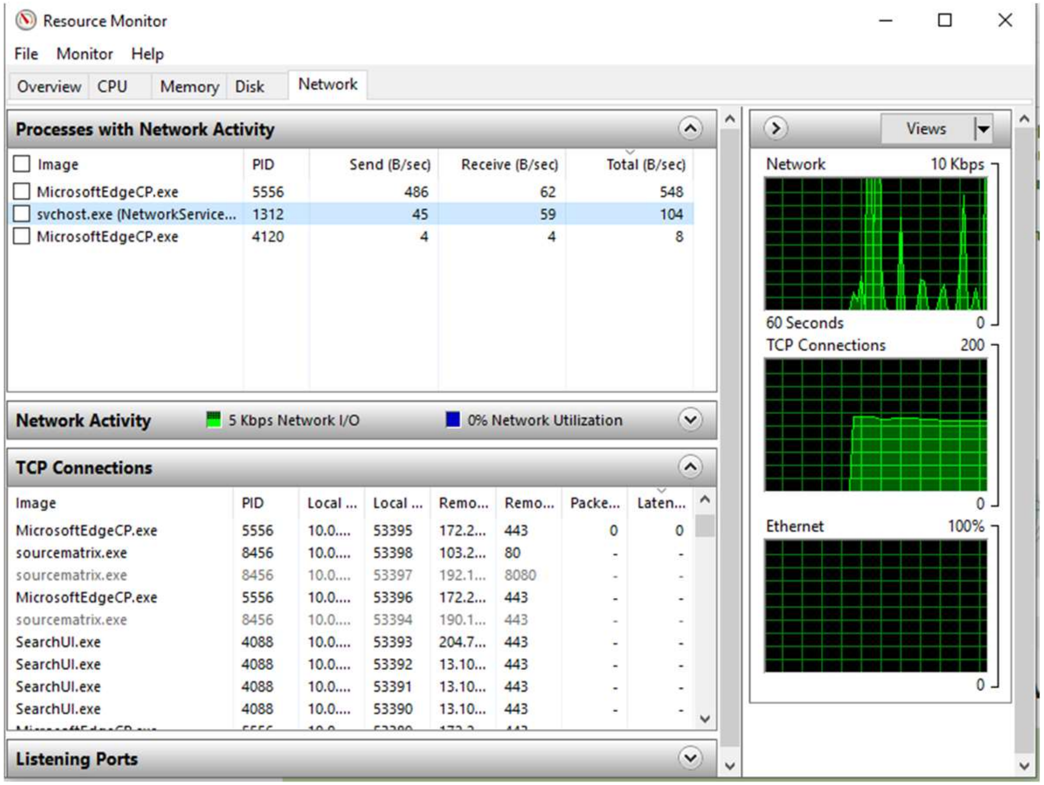
401440 Subroutine as C Code



```
int32_t LoadLibraryA = 0x19054;

int32_t GetProcAddress = 0x19064;

void* fun_401440(int32_t ecx, int32_t a2) {
    int32_t v3;
    int1_t cf4;
    uint32_t eax5;
    uint32_t eax6;
    uint32_t ecx7;
    int32_t eax8;
    int32_t eax9;
    void* eax10;

    v3 = ecx;
    g419f10 = 33;
    while (cf4 = g419f10 < 0x1fb451b, cf4) {
        eax5 = g419f10;
        g419f10 = eax5 + 4;
    }
    eax6 = fun_401290(1, v3);
    g419f10 = eax6;
    ecx7 = g419f10;
    g419ec4 = ecx7;
    eax8 = reinterpret_cast<int32_t>(LoadLibraryA("kernel32", "VirtualAllocEx"));
    eax9 = reinterpret_cast<int32_t>(GetProcAddress(eax8, "kernel32", "VirtualAllocEx"));
    g419f50 = eax9;
    eax10 = fun_4011d0(2, eax8, "kernel32", "VirtualAllocEx");
    return eax10;
}
```

The Internet Connection

Then we checked if the program is connected and used internet actively.



Processes with Network Activity

□ Image	PID	Send (B/sec)	Receive (B/sec)	Total (B/sec)
□ MicrosoftEdgeCP.exe	5556	486	62	548
□ svchost.exe (NetworkService...	1312	45	59	104
□ MicrosoftEdgeCP.exe	4120	4	4	8

Network Activity 🟩 5 Kbps Network I/O 🟦 0% Network Utilization

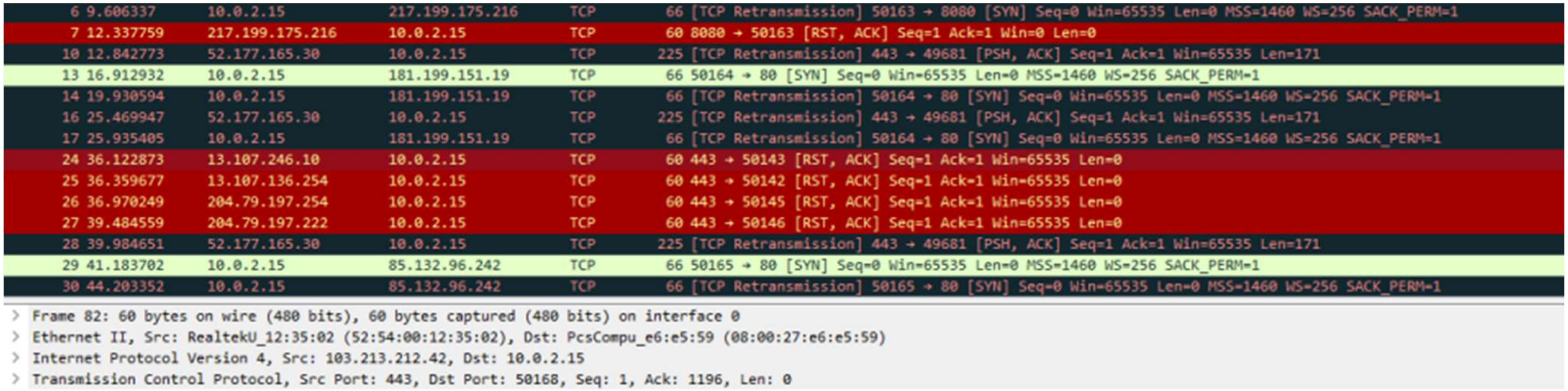
TCP Connections

Image	PID	Local ...	Local ...	Remo...	Remo...	Packe...	Laten...
MicrosoftEdgeCP.exe	5556	10.0....	53395	172.2...	443	0	0
sourcematrix.exe	8456	10.0....	53398	103.2...	80	-	-
sourcematrix.exe	8456	10.0....	53397	192.1...	8080	-	-
MicrosoftEdgeCP.exe	5556	10.0....	53396	172.2...	443	-	-
sourcematrix.exe	8456	10.0....	53394	190.1...	443	-	-
SearchUI.exe	4088	10.0....	53393	204.7...	443	-	-
SearchUI.exe	4088	10.0....	53392	13.10...	443	-	-
SearchUI.exe	4088	10.0....	53391	13.10...	443	-	-
SearchUI.exe	4088	10.0....	53390	13.10...	443	-	-

Listening Ports

IP - Check

- While the program is working we opened the Wireshark.
- This program show the IP addresses that the all the programs communicate with.



```
  6 9.606337   10.0.2.15        217.199.175.216  TCP   66 [TCP Retransmission] 50163 → 8080 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
  7 12.337759  217.199.175.216  10.0.2.15        TCP   60 8080 → 50163 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
 10 12.842773  52.177.165.30    10.0.2.15        TCP  225 [TCP Retransmission] 443 → 49681 [PSH, ACK] Seq=1 Ack=1 Win=65535 Len=171
 13 16.912932  10.0.2.15        181.199.151.19   TCP   66 50164 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 14 19.930594  10.0.2.15        181.199.151.19   TCP   66 [TCP Retransmission] 50164 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 16 25.469947  52.177.165.30    10.0.2.15        TCP  225 [TCP Retransmission] 443 → 49681 [PSH, ACK] Seq=1 Ack=1 Win=65535 Len=171
 17 25.935405  10.0.2.15        181.199.151.19   TCP   66 [TCP Retransmission] 50164 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 24 36.122873  13.107.246.10    10.0.2.15        TCP   60 443 → 50143 [RST, ACK] Seq=1 Ack=1 Win=65535 Len=0
 25 36.359677  13.107.136.254   10.0.2.15        TCP   60 443 → 50142 [RST, ACK] Seq=1 Ack=1 Win=65535 Len=0
 26 36.970249  204.79.197.254   10.0.2.15        TCP   60 443 → 50145 [RST, ACK] Seq=1 Ack=1 Win=65535 Len=0
 27 39.484559  204.79.197.222   10.0.2.15        TCP   60 443 → 50146 [RST, ACK] Seq=1 Ack=1 Win=65535 Len=0
 28 39.984651  52.177.165.30    10.0.2.15        TCP  225 [TCP Retransmission] 443 → 49681 [PSH, ACK] Seq=1 Ack=1 Win=65535 Len=171
 29 41.183702  10.0.2.15        85.132.96.242    TCP   66 50165 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
 30 44.203352  10.0.2.15        85.132.96.242    TCP   66 [TCP Retransmission] 50165 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1

> Frame 82: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_e6:e5:59 (08:00:27:e6:e5:59)
> Internet Protocol Version 4, Src: 103.213.212.42, Dst: 10.0.2.15
> Transmission Control Protocol, Src Port: 443, Dst Port: 50168, Seq: 1, Ack: 1196, Len: 0
```

Conculusion

- Generally APTs are aim is to watch, get targeted information and send it back to the attacker.
- The code we analyzed (Emotet) downloads itself to host machine. The malware tries to learn the current running processes, host names and sends it back to the attacker.
- It is hard to detect, it takes small amount of CPU and Memory .
- The attacker could send payloads through the malware.