Practical Malware Analysis & Triage

Malware Analysis Report

ProcessInjector Malware

Jan 2024 | _fort3 | v1.0

Table of Contents

Executive Summary

SHA256 hash	fca62097b364b2f0338c5e4c5bac86134cedffa4f8ddf27ee9901734128952e3

The ProcessInjector malware is a malicious sample first noticed in the wild 2021-05-14. It is a C portable executable that is built to run on at least a 32-bit Windows operating system. It consists of a binary file hiding under a legitimate process in a 2 stage execution pattern from the initial detonation of the malware binary. Symptoms of infection include the presence of an unknown executable in arbitrary directories.

Typically such malware are difficult to spot but you may also pick up patterns with network based IOCs that stand out as odd.

YARA signature rules are attached in Appendix A.

The Malware sample and hashes have been submitted to VirusTotal for further examination.

High-Level Technical Summary

ProcessInjector consists of 3 parts: a binary named Malware.stage0.exe, a packed legitimate process as a mask and a command execution program that triggers the reverse shell.

When host is connected to internet:

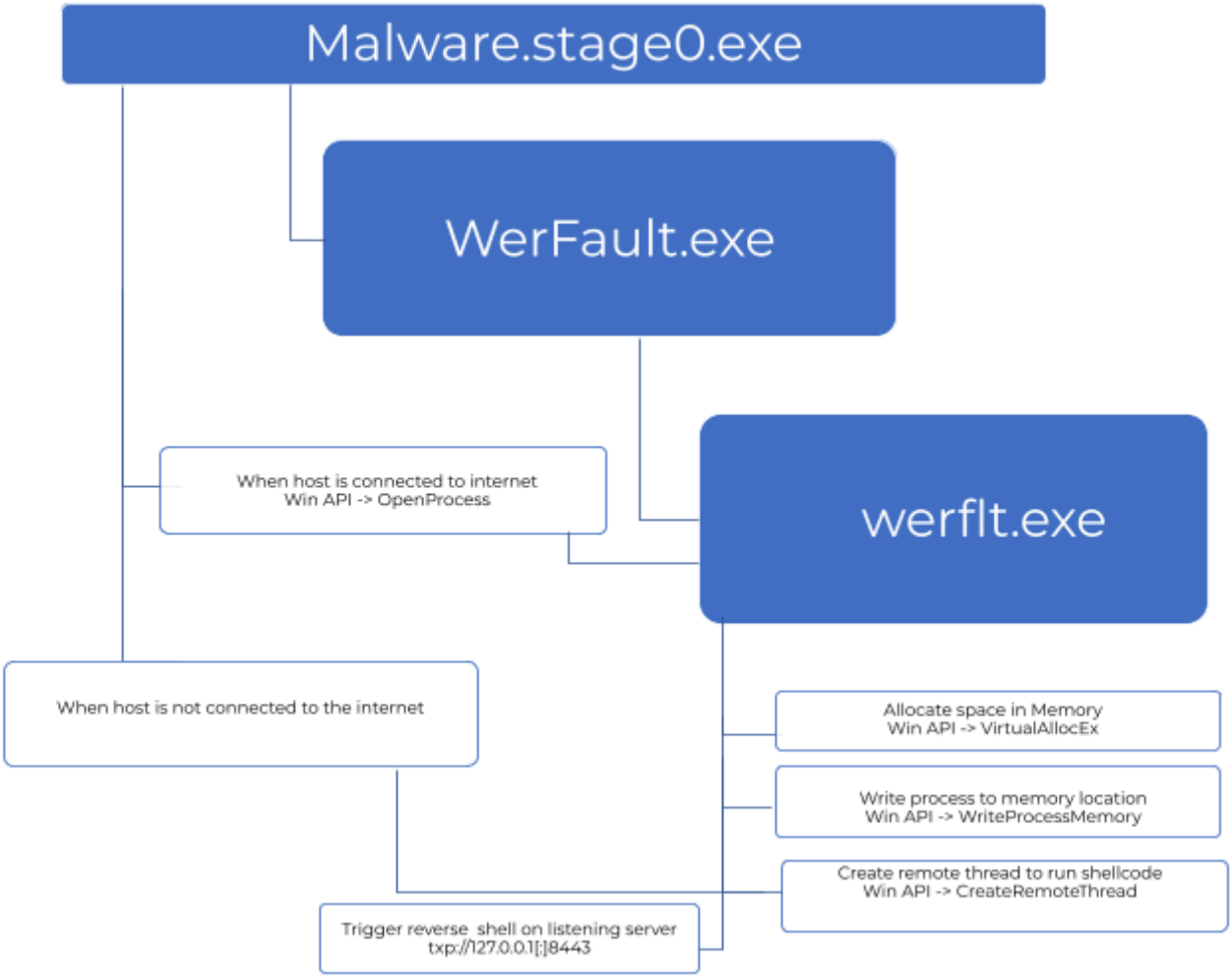
- The malware creates a malicious file named werflt.exe after opening a process and injecting it into parent process WerFault.exe.
- Malicious file allocates a space in memory through the VirtualAllocEx API function call.
- Writes process to memory location allocated using the WriteProcessMemory API.
- Creates a remote thread that allows this shellcode to run and execute using the CreateRemoteThread API.
- Trigger a reverse shell when listening on the port 8443 on the malicious actor's machine IP 127.0.0.1

When host is not connected to internet:

- The malware creates a malicious file named werflt.exe after opening a process and injecting it into parent process WerFault.exe.
- Malicious file allocates a space in memory through the VirtualAllocEx API function call.
- Writes process to memory location allocated using the WriteProcessMemory API.
- Creates a remote thread that allows this shellcode to run and execute using the CreateRemoteThread API.

Notice the difference between both scenarios is in the last step when the host is connected to the internet.

See the flow diagram below for a more descriptive explanation.



ProcessInjector Malware
Jan 2024
v1.0

Malware Composition

ProcessInjector consists of the following components:

File Name	SHA256 Hash
Malware.stage0.exe	fca62097b364b2f0338c5e4c5bac86134cedffa4f8ddf27ee9901734128952e3
werflt.exe	0516009622b951c6c08fd8d81a856eaab70c02e6bc58d066bbdfafe8c6edabea

Malware.stage0.exe

The initial executable that runs after detonation.

werflt.exe

The binary injected into the legitimate parent process WerFault.exe

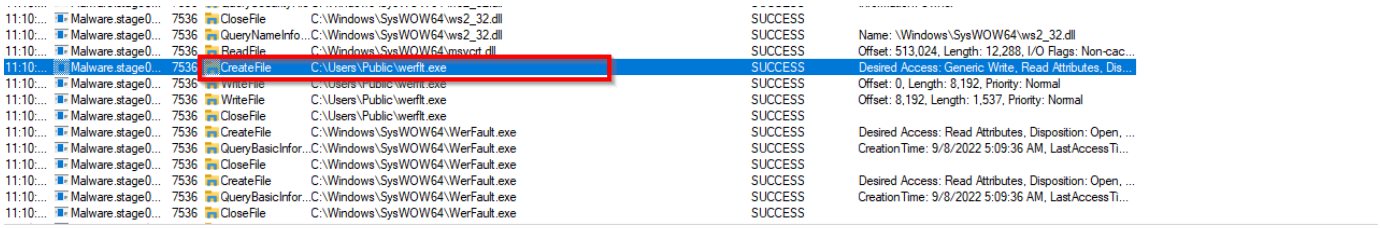


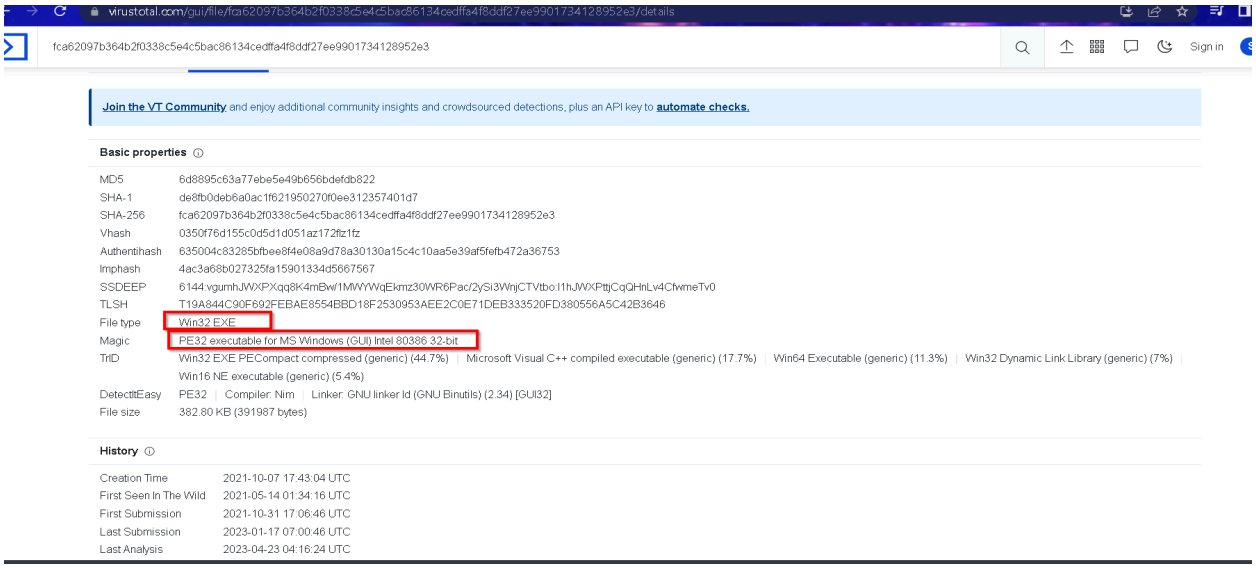
Fig 1: Image of the file created in Procmon..

Basic Static Analysis

Using various online and tools on our sandbox environment, we put together a series of interesting findings.

See the outline below.

VirusTotal Signature:



Contacted IP addresses (6)

IP	Detections	Autonomous System	Country
192.168.0.21	0 / 87	-	-
192.229.211.108	0 / 87	15133	US
20.99.133.109	0 / 87	8075	US
20.99.184.37	2 / 87	8075	US
23.215.176.163	0 / 86	20940	US
23.216.147.76	1 / 87	20940	US

Dropped Files (1)

Scanned	Detections	File type	Name
2022-09-03	49 / 71	Win32 EXE	werfit.exe

SHA-256 0516009622b951c6c08fd8d81a856eaab70c02e6bc58d066bbdfafe8c6edabea
File Size 9.50 KB

Graph Summary

ProcessInjector Malware
Jan 2024
v1.0

Floss:
interesting strings:

@\.\pipe\stdin

@\.\pipe\stdout

@C:\Users\Public\werflt.exe

@C:\Windows\SysWOW64\WerFault.exe

@C:\Users\Public\werflt.exe

C:\Users\Administrator\source\repos\CRTInjectorConsole\Release\CRTInjectorConsole.pdb

GCTL

WriteProcessMemory

OpenProcess

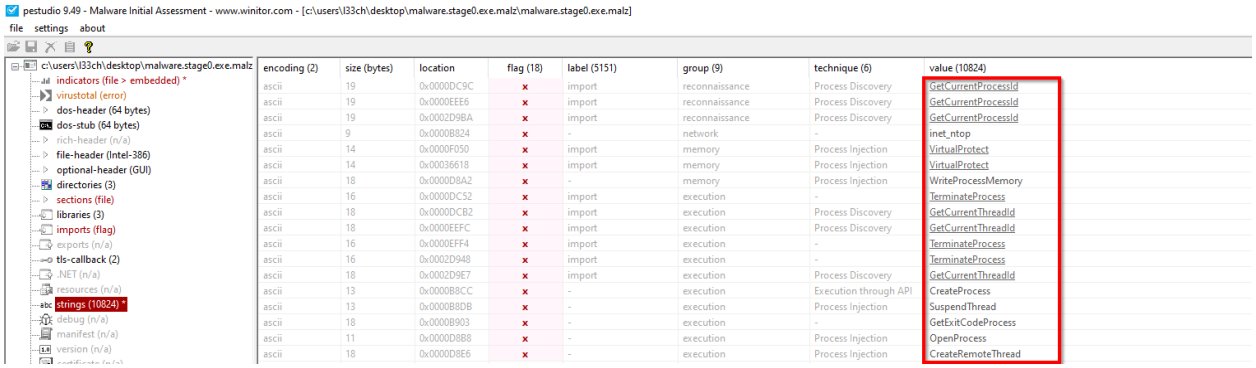
CloseHandle

VirtualAllocEx

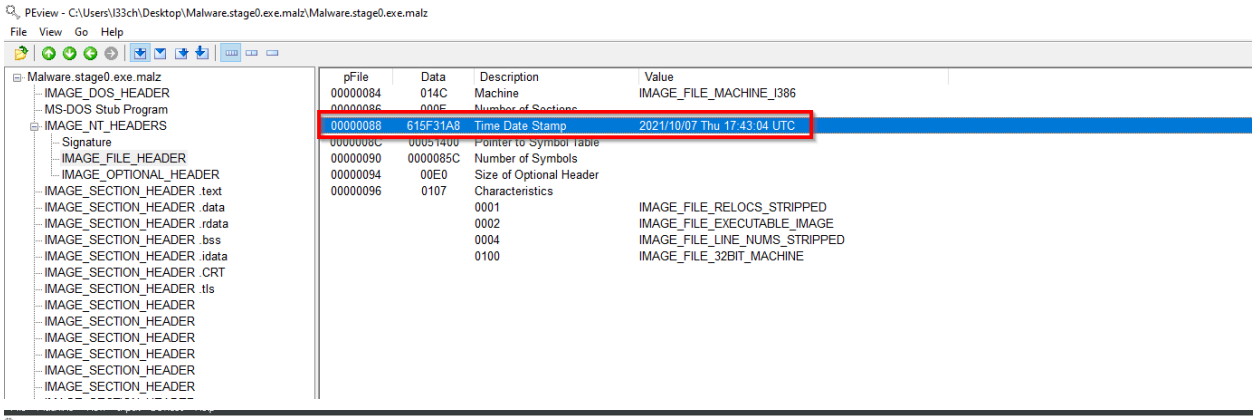
CreateRemoteThread

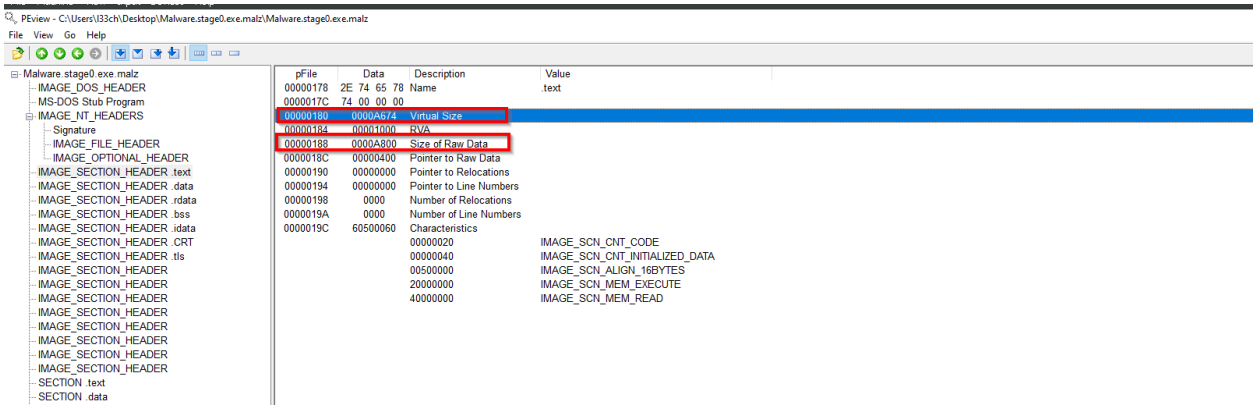
KERNEL32.dll

PEstudio:



PEview:





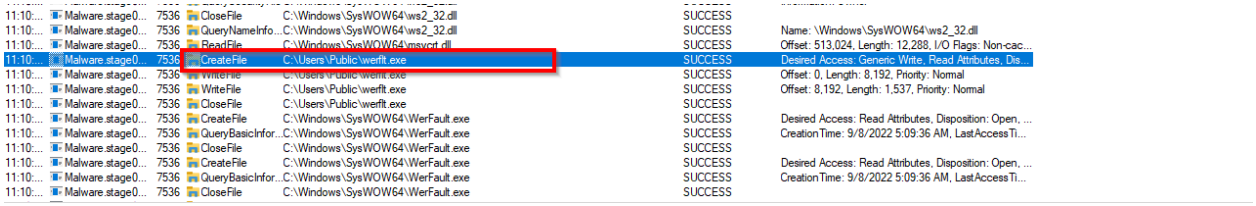
Based on preliminary static analysis using the PEview tool, we see;
Virtual Size in Decimals: 42,612
Size of Raw Data in Decimals: 43,008
Based on the difference in both values, we determined that the malware is packed but to confirm we needed more analysis.

Basic Dynamic Analysis

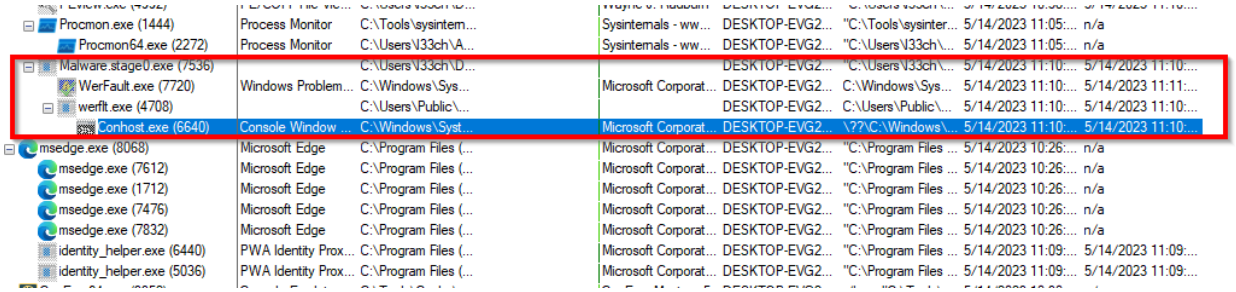
Using various tools in our sandbox environment, we put together a series of interesting findings for the dynamic analysis of this sample.

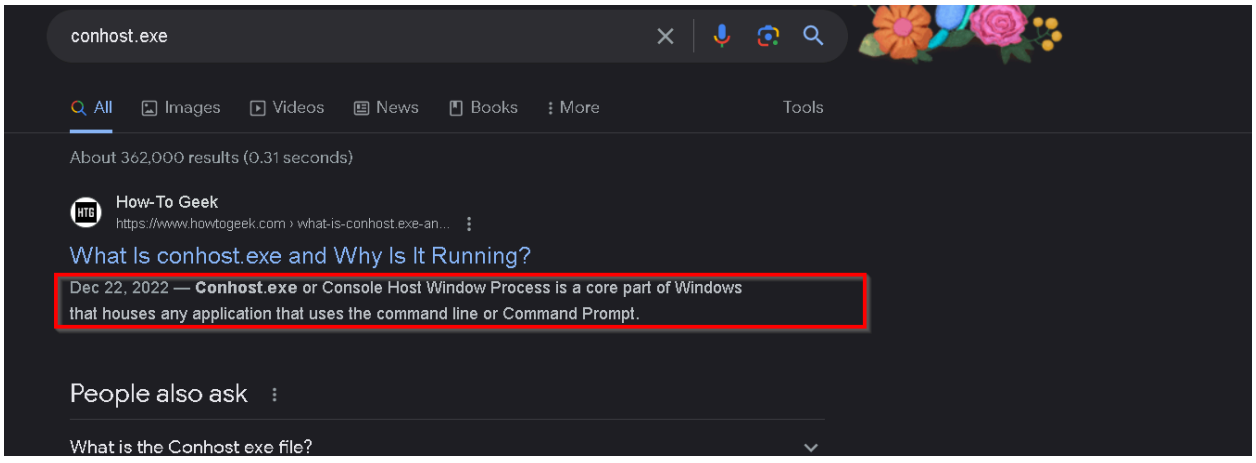
Procmon:

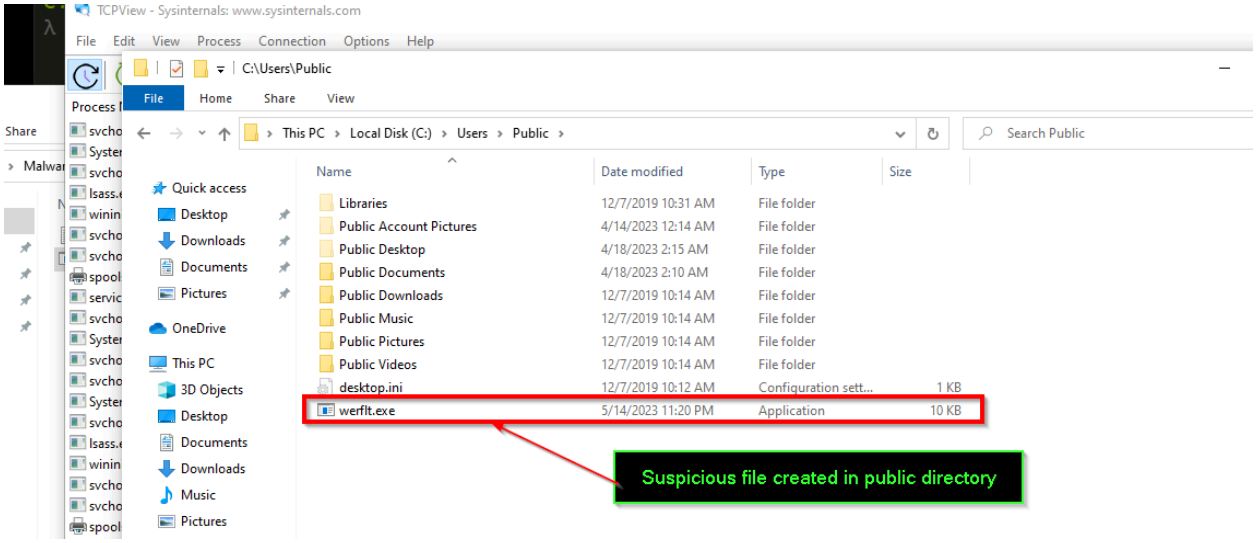
Interesting string confirmed to be a file created ==> C:\Users\Public\werflt.exe



Checking the process tree for the malware exe file, we see another process was created when the legitimate werfault.exe was run.



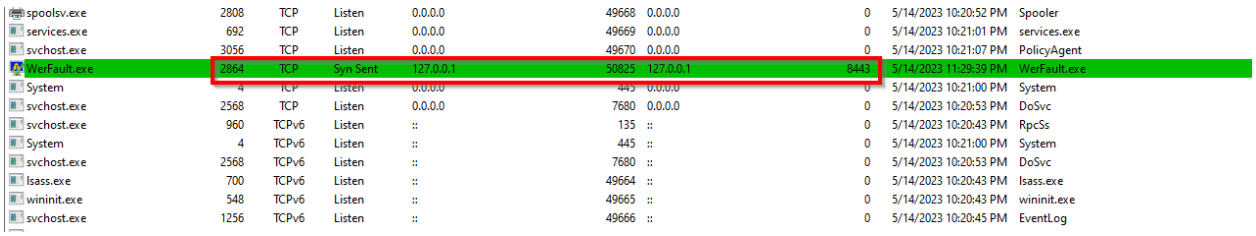




TCPview:

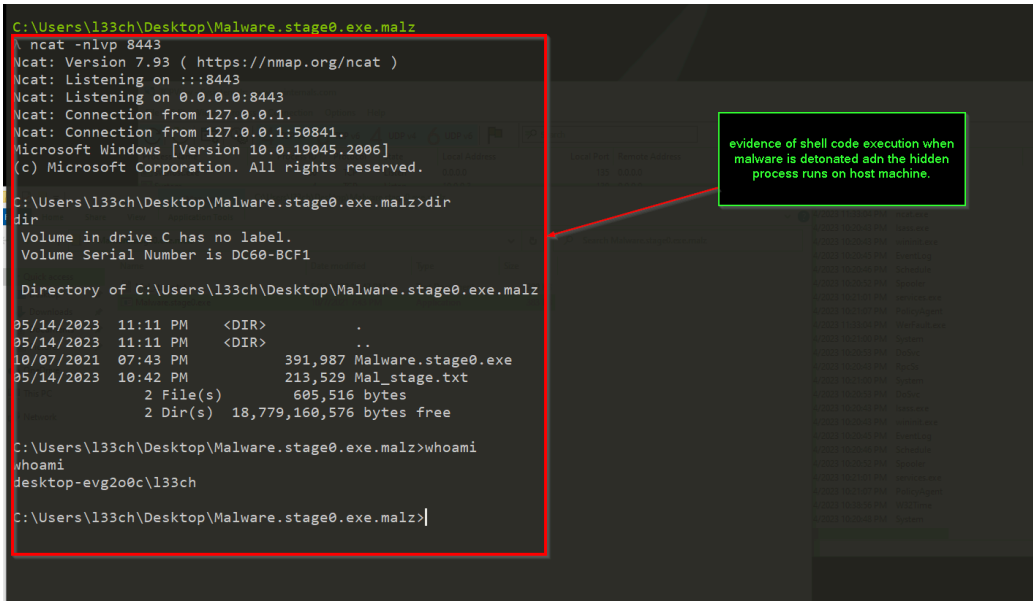
Investigating the suspicious behavior of the process of interest:

Port opened and listening on 8443



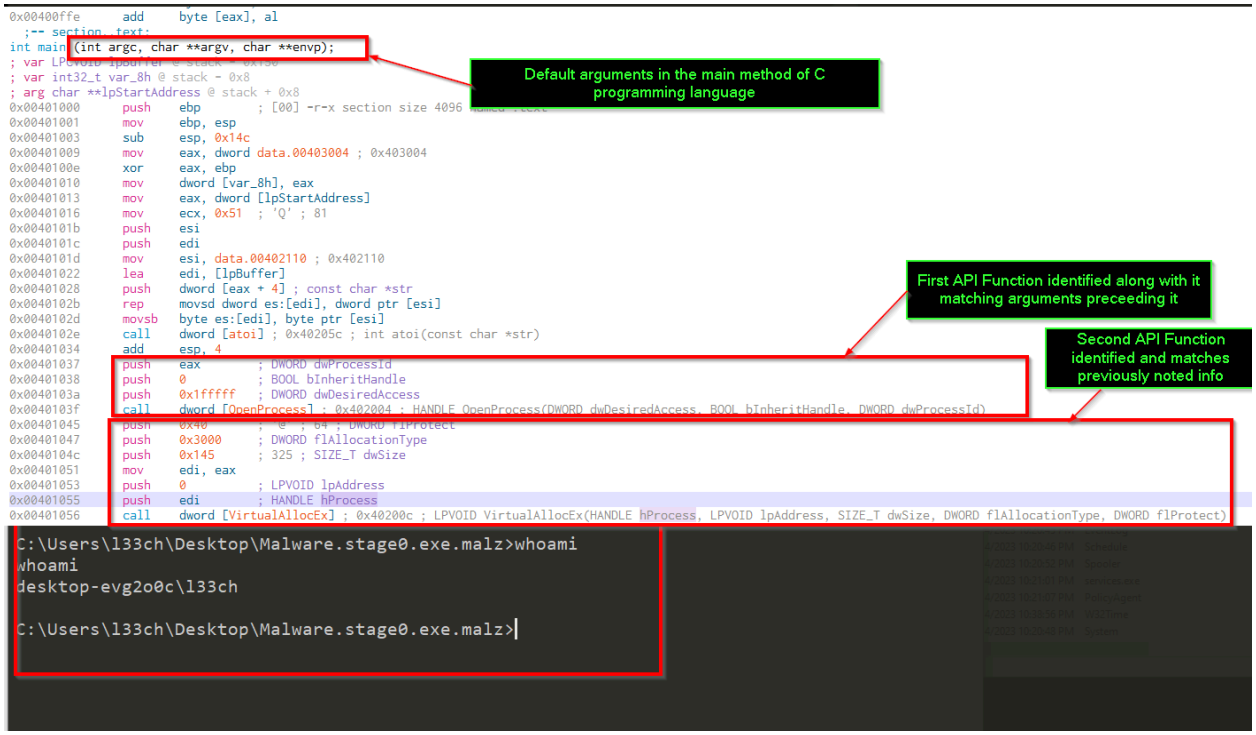
spoolsv.exe	2808	TCP	Listen	0.0.0.0	49668	0.0.0.0	0	5/14/2023 10:20:52 PM	Spooler
services.exe	692	TCP	Listen	0.0.0.0	49669	0.0.0.0	0	5/14/2023 10:21:01 PM	services.exe
svchost.exe	3056	TCP	Listen	0.0.0.0	49670	0.0.0.0	0	5/14/2023 10:21:07 PM	PolicyAgent
WerFault.exe	2864	TCP	Syn Sent	127.0.0.1	50825	127.0.0.1	8443	5/14/2023 11:29:39 PM	WerFault.exe
System	4	TCP	Listen	0.0.0.0	445	0.0.0.0	0	5/14/2023 10:21:00 PM	System
svchost.exe	2568	TCP	Listen	0.0.0.0	7680	0.0.0.0	0	5/14/2023 10:20:53 PM	DoSvc
svchost.exe	960	TCPv6	Listen	::	135	::	0	5/14/2023 10:20:43 PM	RpcSs
System	4	TCPv6	Listen	::	445	::	0	5/14/2023 10:21:00 PM	System
svchost.exe	2568	TCPv6	Listen	::	7680	::	0	5/14/2023 10:20:53 PM	DoSvc
lsass.exe	700	TCPv6	Listen	::	49664	::	0	5/14/2023 10:20:43 PM	lsass.exe
wininit.exe	548	TCPv6	Listen	::	49665	::	0	5/14/2023 10:20:43 PM	wininit.exe
svchost.exe	1256	TCPv6	Listen	::	49666	::	0	5/14/2023 10:20:45 PM	EventLog

Evidence of compromise and code execution in the screenshot below.

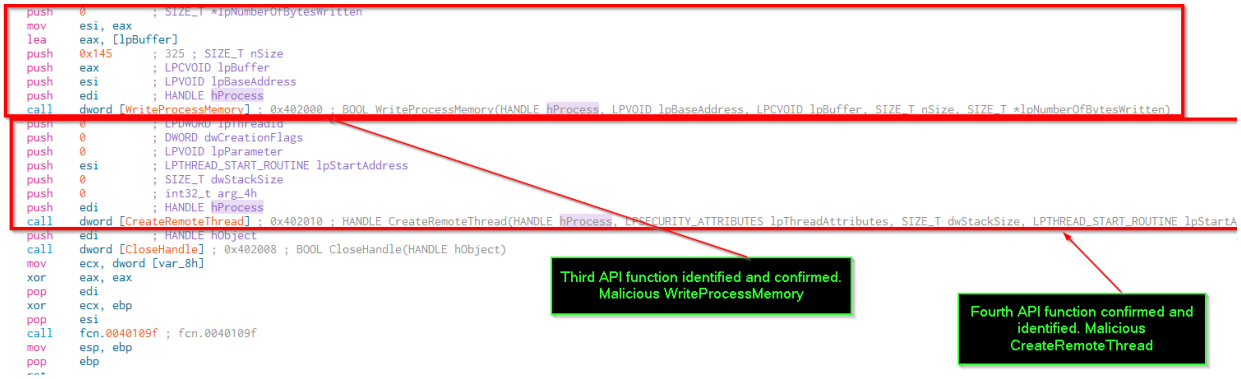


Advanced Static Analysis

Using Cutter:

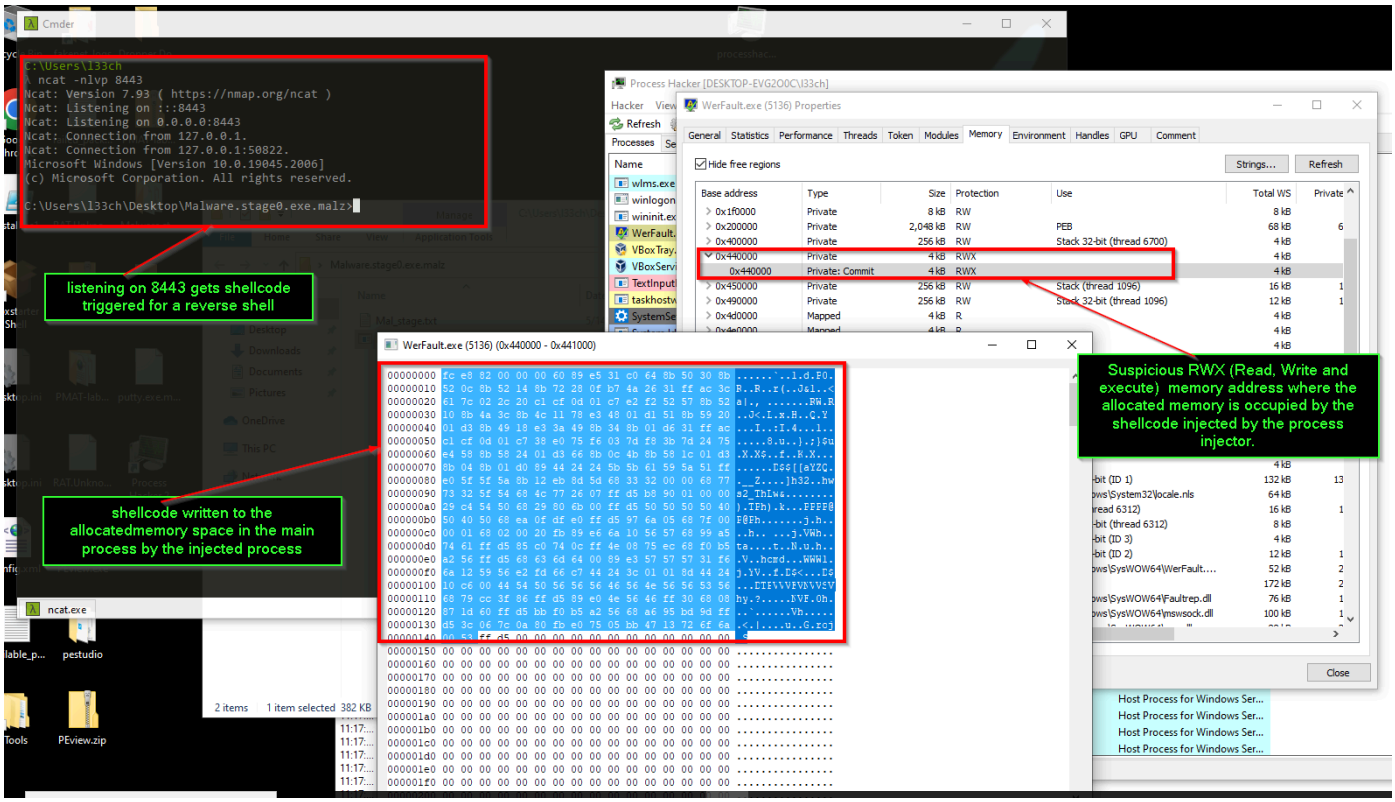


ProcessInjector Malware
Jan 2024
v1.0



Advanced Dynamic Analysis

Output on Process Hacker showing the memory address location of the injected shellcode that triggers a reverse shell when listening on port 8443.



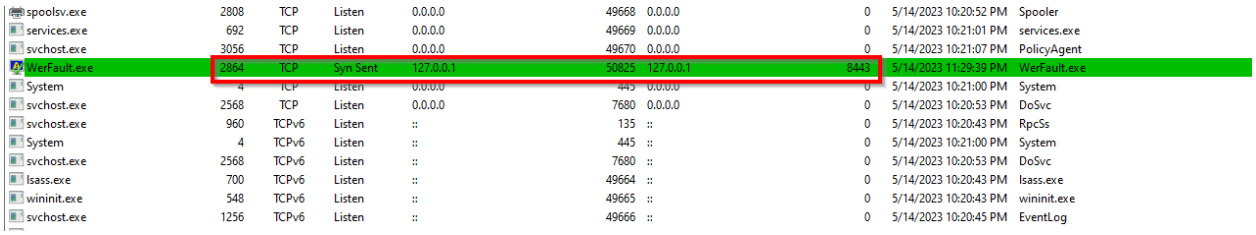
ProcessInjector Malware
Jan 2024
v1.0

Indicators of Compromise

Network Indicators

Shell triggered while listening on port 8443

- tcp://127.0.0.1[:]8443

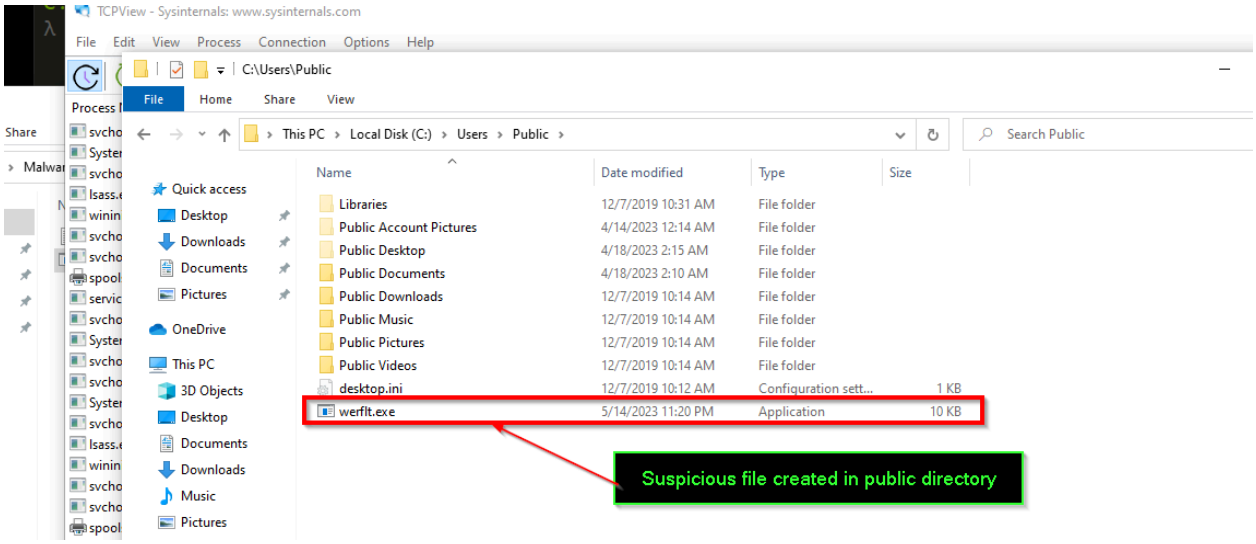


spoolsv.exe	2808	TCP	Listen	0.0.0.0	49668	0.0.0.0	0	5/14/2023 10:20:52 PM	Spooler
services.exe	692	TCP	Listen	0.0.0.0	49669	0.0.0.0	0	5/14/2023 10:21:01 PM	services.exe
svchost.exe	3056	TCP	Listen	0.0.0.0	49670	0.0.0.0	0	5/14/2023 10:21:07 PM	PolicyAgent
WerFault.exe	2864	TCP	Syn Sent	127.0.0.1	50825	127.0.0.1	8443	5/14/2023 11:29:39 PM	WerFault.exe
System	4	TCP	Listen	0.0.0.0	445	0.0.0.0	0	5/14/2023 10:21:00 PM	System
svchost.exe	2568	TCP	Listen	0.0.0.0	7680	0.0.0.0	0	5/14/2023 10:20:53 PM	DoSvc
svchost.exe	960	TCPv6	Listen	::	135	::	0	5/14/2023 10:20:43 PM	RpcSs
System	4	TCPv6	Listen	::	445	::	0	5/14/2023 10:21:00 PM	System
svchost.exe	2568	TCPv6	Listen	::	7680	::	0	5/14/2023 10:20:53 PM	DoSvc
lsass.exe	700	TCPv6	Listen	::	49664	::	0	5/14/2023 10:20:43 PM	lsass.exe
wininit.exe	548	TCPv6	Listen	::	49665	::	0	5/14/2023 10:20:43 PM	wininit.exe
svchost.exe	1256	TCPv6	Listen	::	49666	::	0	5/14/2023 10:20:45 PM	EventLog

Host-based Indicators

File/process creation and Injection

- C:\Users\Public\werflt.exe
- C:\Windows\SysWOW64\WerFault.exe
- WriteProcessMemory
- OpenProcess
- CloseHandle
- VirtualAllocEx
- CreateRemoteThread
- KERNEL32.dll



ProcessInjector Malware
Jan 2024
v1.0

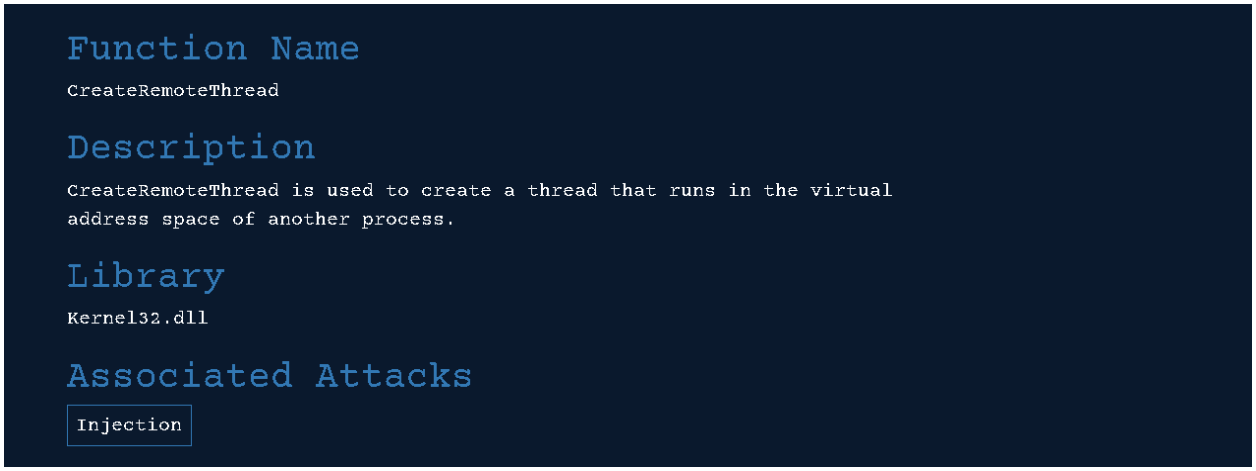
Rules & Signatures

A full set of YARA rules is included in Appendix A.

Interesting string confirmed to be a file created C:\Users\Public\werflt.exe in addition to the legitimate binary used as a mask C:\Windows\SysWOW64\WerFault.exe.

This string is added to the Yra rules created to detect this malware.

There's also the CreateRemoteThread Windows API which would certainly be an odd function that we can identify.



Also since we're dealing with a PE (Portable Executable), then we also need to include a rule to look for the first magic byte for this type of file.

Appendices

A. Yara Rules

```
rule ProcessInjected {

    meta:
        last_updated = "2024-01-31"
        author = "Fortune Sam Okon"
        description = "A sample Yara rule for PMAT course final"

    strings:
        // Fill out identifying strings and other criteria
        $string1 = "@C:\Users\Public\werflt.exe"
        $string2 = "CreateRemoteThread"
        $string3 = "@C:\Windows\SysWOW64\WerFault.exe"
        $PE_magic_byte = "MZ"
        $sus_hex_string = {8E ?? ??}

    condition:
        // Fill out the conditions that must be met to identify the binary
        $PE_magic_byte at 0 and
        ($string1 and $string2 and $string3) or
        $sus_hex_string
}
```

B. Callback URL

127.0.0.1[	Port
txp://127.0.0.1	8443

C. Decompiled Code Snippets

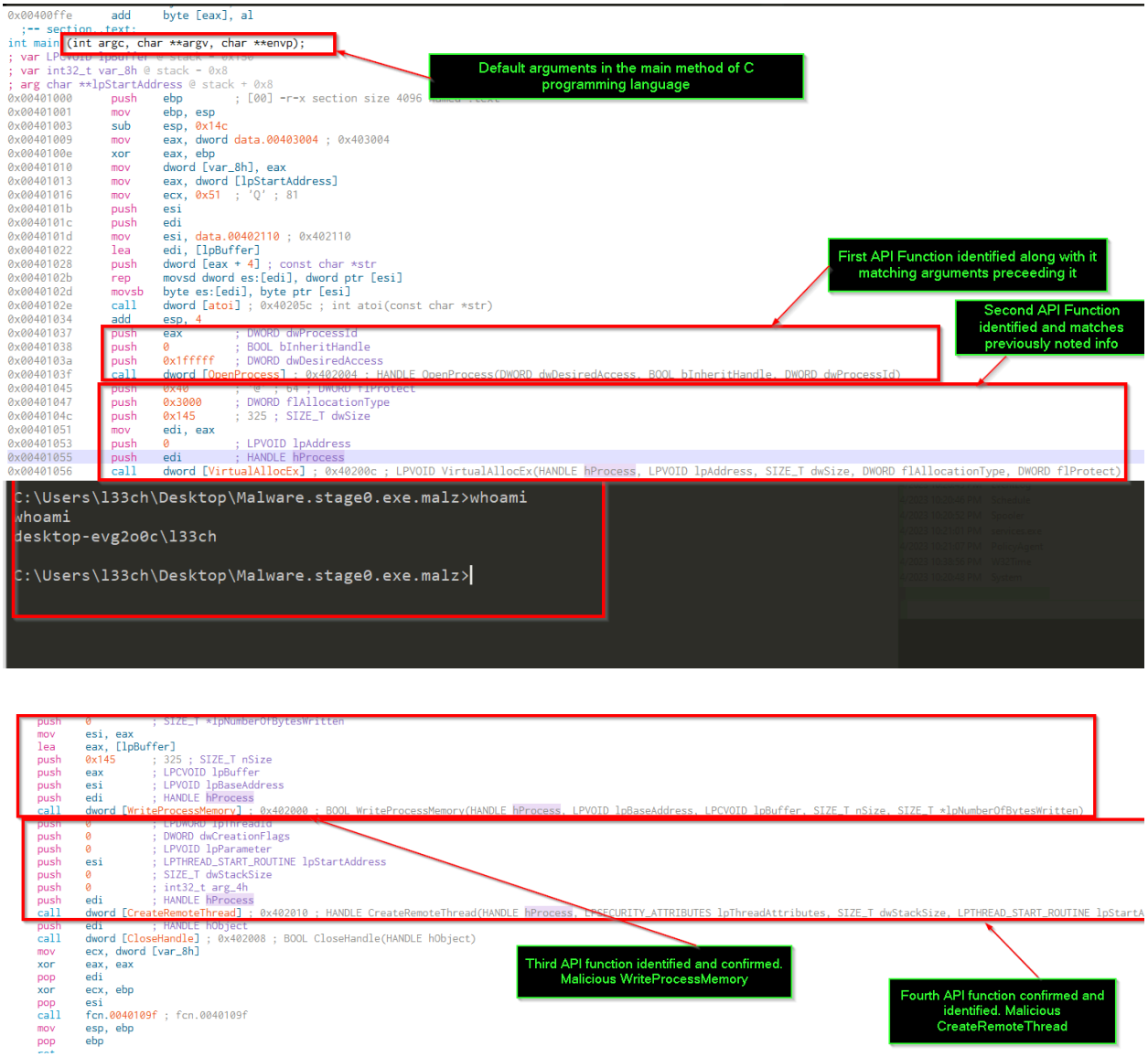


Fig 4&5: Process Injection Routine in Cutter