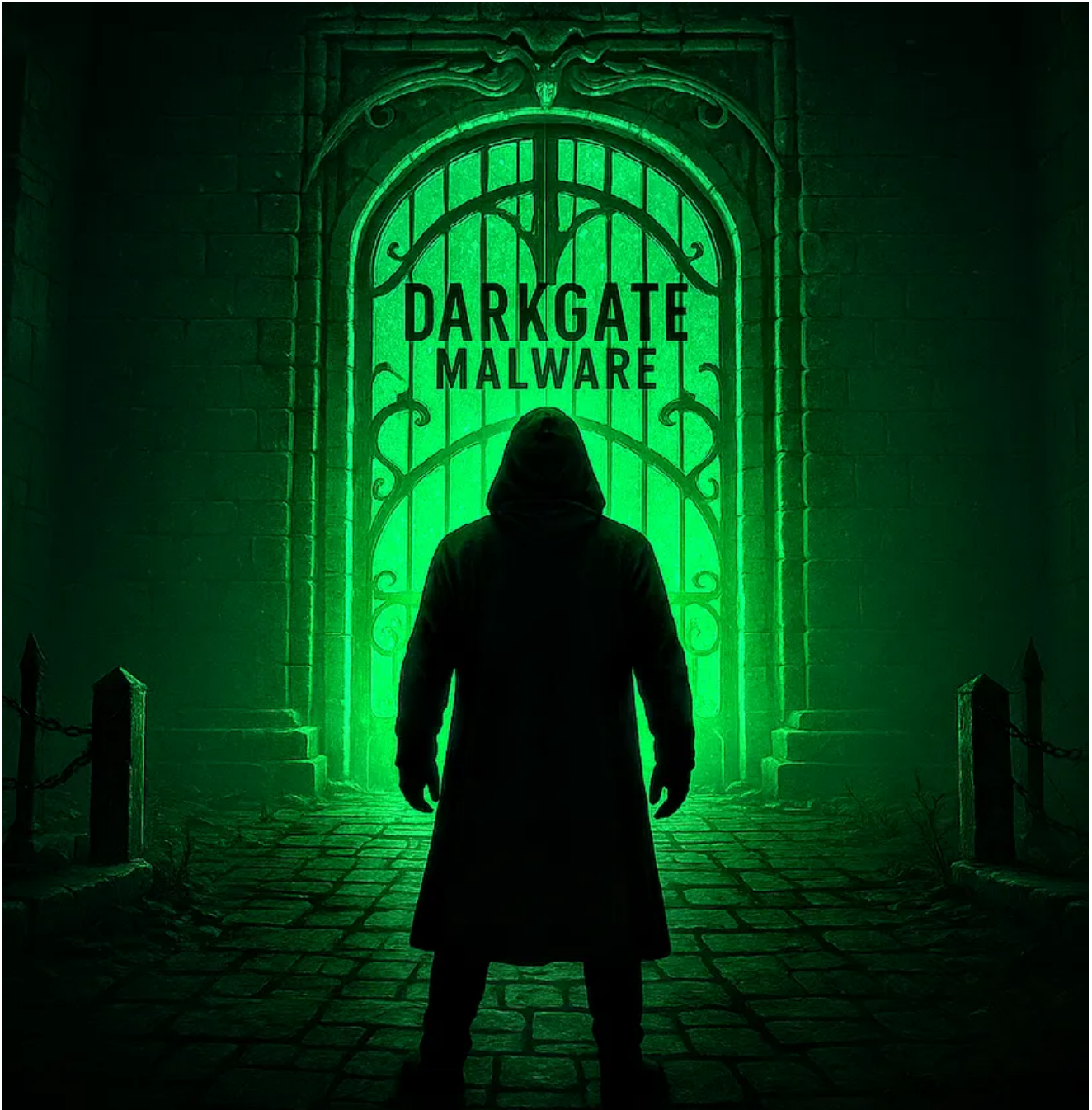


Inside DarkGate: In-Depth Technical Analysis of the Malware-as-a-Service Threat

 medium.com/@sapirtwig/inside-darkgate-in-depth-technical-analysis-of-the-malware-as-a-service-threat-76f32d51e2d2

June 29, 2025



Introduction

In this report, I present an extensive, step-by-step static analysis of a real-world sample of the DarkGate Remote Access Trojan (RAT), a sophisticated and highly modular malware that has become emblematic of the Malware-as-a-Service (MaaS) threat landscape. Originally discovered in 2018, DarkGate has evolved to incorporate a broad spectrum of malicious capabilities, including but not limited to remote desktop control, credential theft, keylogging, file exfiltration, cryptomining, and advanced anti-analysis features.

This document details my methodology, findings, and interpretations derived from a deep static analysis using open-source tools. The goal is to illuminate both the technical mechanisms underlying DarkGate's operations and the analytical workflow required to dissect such a complex threat.

Summary & Key Highlights

- Identified advanced , including keylogging, remote desktop control, credential theft, audio recording, and file exfiltration.
- Performs using `NtWriteVirtualMemory`, avoiding disk artifacts.
- Implements sophisticated , checking for common analysis tools.
- Establishes via dynamically loaded Winsock APIs, using legitimate-looking User-Agent strings.
- Leverages `cmdkey` and to exfiltrate browser and email credentials.
- Achieves via registry keys, startup folders, and scripting tools like AutoHotkey/AutoIt.
- Extracted numerous , including registry keys, suspicious strings, filenames, directories, and network indicators.

Sample Details

- 2143d7603258b2801f7ed154b5da3da6
- 3c64cbb7e7212d920322dae62665b05ceb63a0ad6074cac3ba518cedc5c6dd48
- 64 bytes (suggesting a loader or dropper, clarified through further analysis)

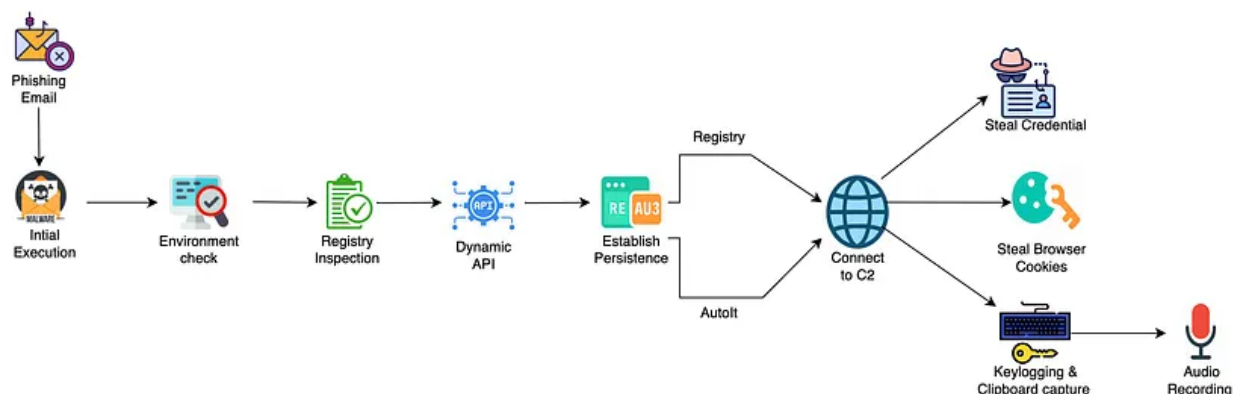


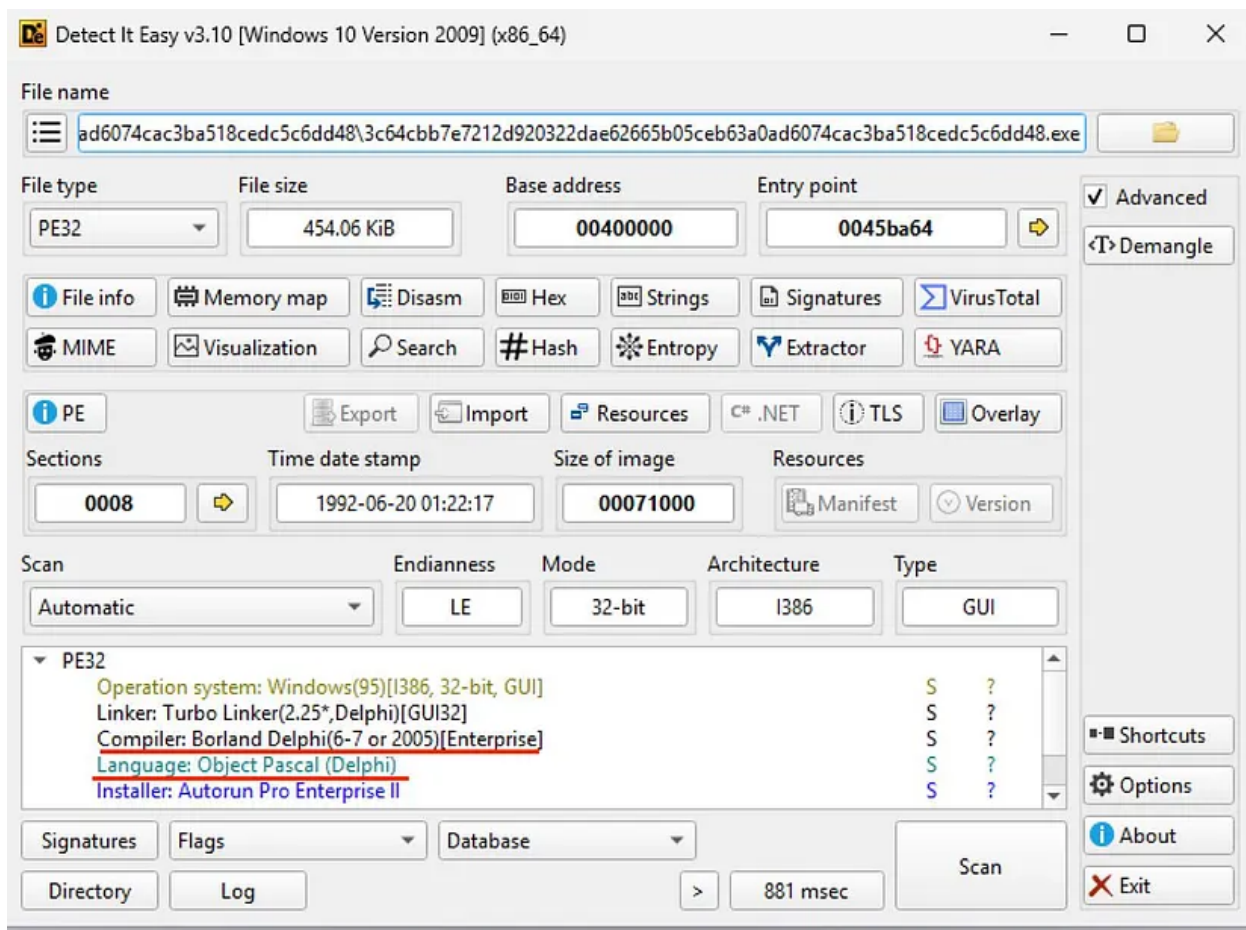
Figure: DarkGate Malware Infection Flow

My workflow began with a high-level triage to understand the file's structure, packing, and surface-level capabilities, followed by a systematic function-by-function reverse engineering process to uncover deeper behavioral logic and evasion techniques.

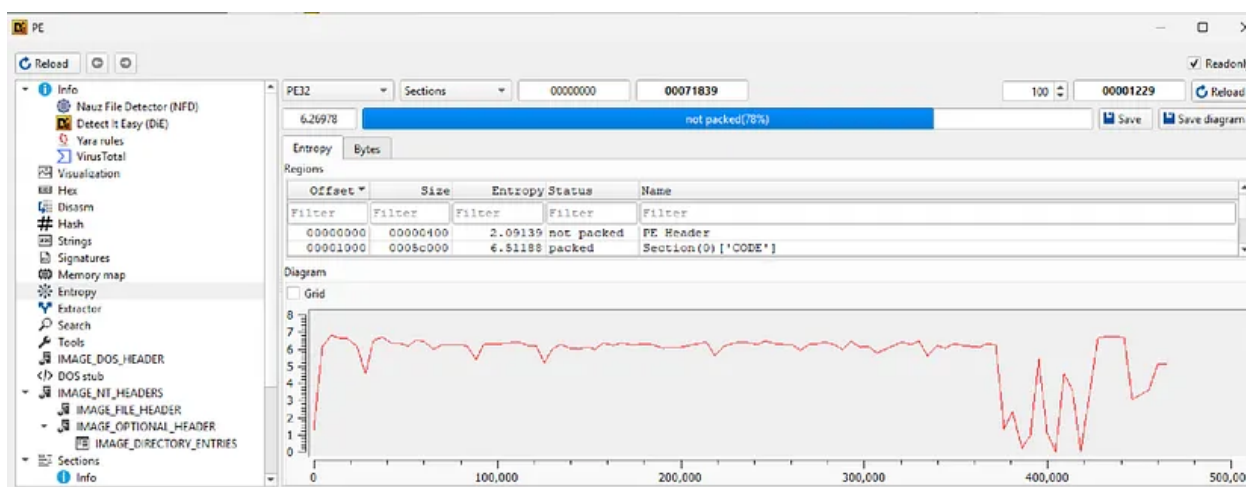
Static Analysis and Reverse Engineering Findings

1. Initial File Assessment

DIE identified Borland Delphi (Object Pascal) as the compiler, with no known commercial packers detected. However, entropy analysis revealed a value of 6.51 in the CODE section, a strong indicator of custom obfuscation or packing.



Detect It Easy — Malware information



Malware entropy analysis

PeStudio highlighted the presence of APIs associated with process injection (WriteProcessMemory, CreateRemoteThread, VirtualAllocEx), keylogging (GetAsyncKeyState, keybd_event, GetCursorPos), file and clipboard manipulation (ReadFile, WriteFile, CreateFileA/W, OpenClipboard, GetClipboardData), and audio capture (waveInOpen, waveInStart). These imports collectively suggest a RAT with extensive surveillance, data theft, and persistence capabilities.

pestudio 9.61 - Malware Initial Assessment - www.winitor.com | c:\malware lab\3c64cbb7e7212d920322dae62665b05ceb63a0ad6074cac3ba518cedc5c6dd48\3c64cbb7...

file settings about

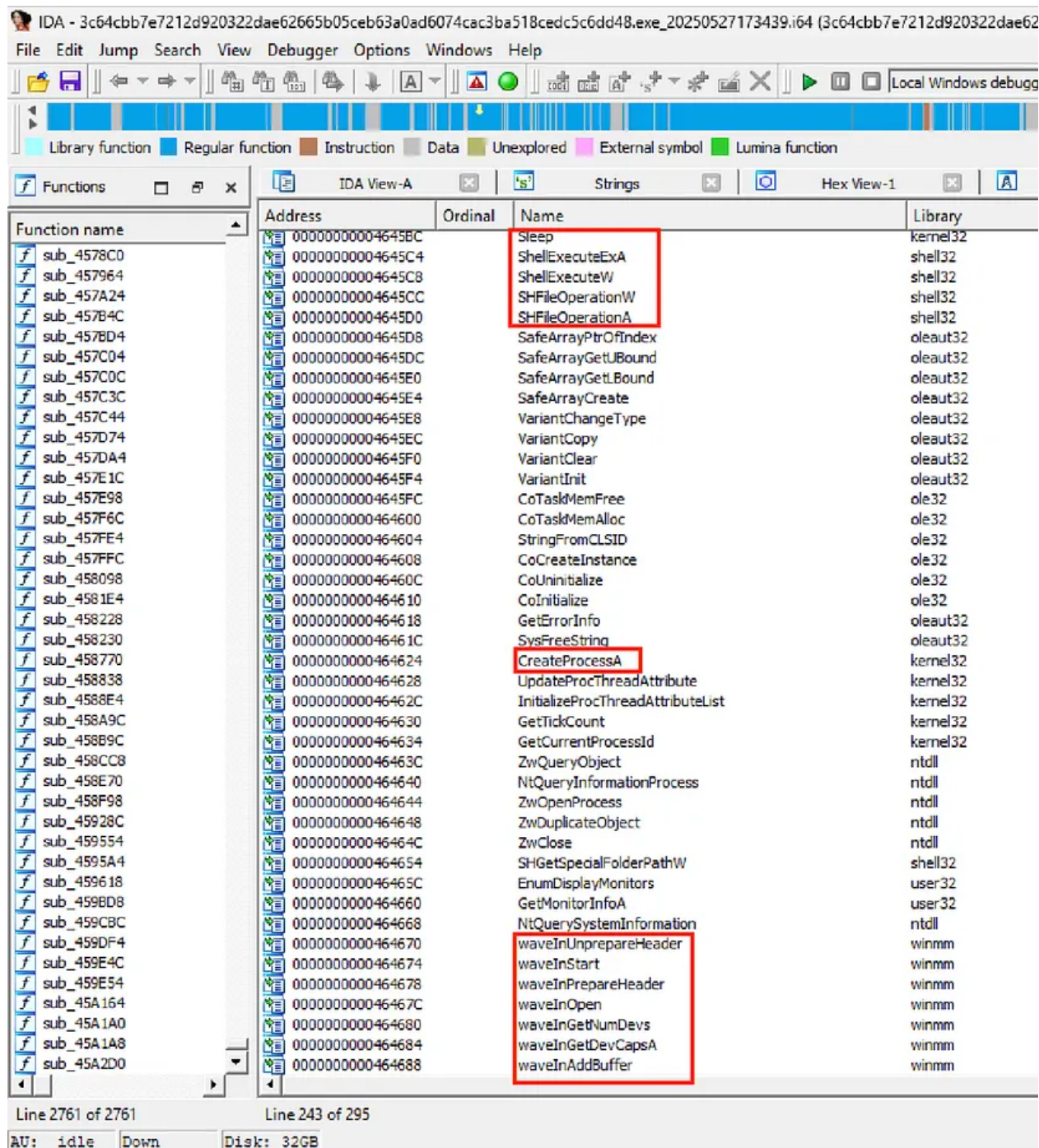
c:\malware lab\3c64cbb7e7212d920322dae62665b05

- indicators (wait...)
- footprints (wait...)
- virustotal (score > 60/71)**
- dos-header (size > 64 bytes)
- dos-stub (size > 192 bytes)
- rich-header (n/a)
- file-header (executable > 32-bit)
- optional-header (subsystem > GUI)
- directories (count > 4)
- sections (file > unknown)
- libraries (flag > 1)
- imports (flag > 68)**
- exports (n/a)
- thread-local-storage (n/a)
- .NET (n/a)
- resources (signature > unknown)
- strings (count > 9775)
- debug (n/a)
- manifest (n/a)
- version (n/a)
- certificate (n/a)
- overlay (signature > unknown)

imports (295)	flag (68)	type	ordinal	first-thunk (IAT)	first-thunk-original (INT)
WriteProcessMemory	x	implicit	-	0x00064AD4	0x00000000
WriteFile	x	implicit	-	0x00064AEA	0x00000000
WaitForSingleObject	-	implicit	-	0x00064AF6	0x00000000
VirtualQuery	x	implicit	-	0x00064B0C	0x00000000
VirtualFree	-	implicit	-	0x00064B1C	0x00000000
VirtualAllocE double-click > url	x	implicit	-	0x00064B2A	0x00000000
VirtualAlloc	x	implicit	-	0x00064B3C	0x00000000
TerminateThread	-	implicit	-	0x00064B4C	0x00000000
TerminateProcess	-	implicit	-	0x00064B5E	0x00000000
Sleep	-	implicit	-	0x00064B72	0x00000000
SetThreadPriority	-	implicit	-	0x00064B7A	0x00000000
SetFilePointer	-	implicit	-	0x00064B8E	0x00000000
SetFileAttributesA	x	implicit	-	0x00064BA0	0x00000000
SetEvent	-	implicit	-	0x00064BB6	0x00000000
SetErrorMode	-	implicit	-	0x00064BC2	0x00000000
SetEndOfFile	-	implicit	-	0x00064BD2	0x00000000
SetCurrentDirectoryA	-	implicit	-	0x00064BE2	0x00000000
ResumeThread	-	implicit	-	0x00064BFA	0x00000000
ResetEvent	-	implicit	-	0x00064C0A	0x00000000
ReadProcessMemory	x	implicit	-	0x00064C18	0x00000000
ReadFile	-	implicit	-	0x00064C2C	0x00000000
QueryPerformanceFrequency	x	implicit	-	0x00064C38	0x00000000
QueryPerformanceCounter	-	implicit	-	0x00064C54	0x00000000
PeekNamedPipe	-	implicit	-	0x00064C6E	0x00000000
OpenProcess	x	implicit	-	0x00064C7E	0x00000000

sha256 > 3C64CBB7E7212D920322DAE62665B05CEB63A0AD6074CAC3BA518CEDC5C6DD48 | cpu > 32-bit | file > type > executable | subsystem > GUI

Detected APIs indicate injection, key-logging, and data access capabilities - pestudio



Malware imports table - IDA

2. Registry and Environment Inspection

One of the earliest behaviors observed is the malware's access to the Windows registry, specifically targeting the key SOFTWARE\Borland\Delphi\RTL and querying the value FPUMaskValue using RegOpenKeyExA and RegQueryValueExA. This serves multiple purposes:

- Potentially fetching runtime configuration or operational parameters.

- Checking for specific registry values may help the malware identify analysis environments or sandboxes.
- The focus on Borland Delphi keys further confirms the compiler and development environment used for the malware.

```

push    ebp
mov     ebp, esp
add     esp, 0FFFFFFF4h
movzx   eax, ds:word_45D024
mov     dword ptr [ebp+Data], eax
lea     eax, [ebp+phkResult]
push    eax                ; phkResult
push    1                  ; samDesired
push    0                  ; ulOptions
push    offset aSoftwareBorlan ; "SOFTWARE\\Borland\\Delphi\\RTL"
push    80000002h          ; hKey
call    RegOpenKeyExA
test    eax, eax
jnz     short loc_4035BC

xor     eax, eax
push    ebp
push    offset sub_4035B5
push    dword ptr fs:[eax]
mov     fs:[eax], esp
mov     [ebp+cbData], 4
lea     eax, [ebp+cbData]
push    eax                ; lpcbData
lea     eax, [ebp+Data]
push    eax                ; lpData
push    0                  ; lpType
push    0                  ; lpReserved
push    offset aFpumaskvalue ; "FPUMaskValue"
mov     eax, [ebp+phkResult]
push    eax                ; hKey
call    RegQueryValueExA
xor     eax, eax

```

Registry read from Delphi-specific key via RegQueryValueExA

3. Path Manipulation and Anti-Static Analysis

The function sub_405A20 is dedicated to resolving and manipulating filesystem paths. By dynamically loading GetLongPathNameA from kernel32.dll at runtime, DarkGate avoids static detection of its API usage. The function converts short DOS-style paths to their canonical long forms and verifies their existence using FindFirstFileA. It also handles UNC paths (\\server\share), suggesting readiness for network propagation or interaction with shared resources. The use of conditional logic and string operations (lstrcpynA) reveals a deliberate effort to evade static analysis and adapt to varying system configurations.



Uses GetLongPathNameA and FindFirstFileA to resolve file paths dynamically

4. Process and Memory Enumeration

The routine sub_40F7C8 demonstrates DarkGate's advanced system reconnaissance abilities. By dynamically resolving APIs such as CreateToolhelp32Snapshot, Process32First/Next, Thread32First/Next, Module32First/Next, and Toolhelp32ReadProcessMemory, the malware gains the ability to:

- Enumerate all running processes, threads, and loaded modules.
- Read memory from other processes, laying the groundwork for process injection, credential theft, and lateral movement.
- Evade static detection by resolving these APIs only at runtime, a hallmark of sophisticated malware.



ToolHelp32 APIs resolved at runtime to enumerate system components

5. Variant and COM Data Handling

The function sub_410028 loads numerous OLE automation APIs (e.g., VariantChangeTypeEx, Var*FromStr, VarBstrFrom* from oleaut32.dll). This empowers DarkGate to:

- Seamlessly convert and process various data types (numbers, dates, strings).
- Interact with COM objects and potentially parse complex C2 commands.

- Enhance its adaptability and flexibility in handling data received from or sent to its operators, making it more resilient to changes in C2 protocols or payload formats.

```

; Attributes: bp-based frame

sub_410028 proc near

var_4= dword ptr -4

push    ebp
mov     ebp, esp
push    ecx
push    offset aOleaut32Dll ; "oleaut32.dll"
call    GetModuleHandleA_1
mov     [ebp+var_4], eax
push    ebp
mov     edx, offset sub_40FB8C
mov     eax, offset aVariantchanget ; "VariantChangeTypeEx"
call    sub_40FFF0
pop     ecx
mov     ds:dword_4627FC, eax
push    ebp
mov     edx, offset sub_40FB8C
mov     eax, offset aVarNeg ; "VarNeg"
call    sub_40FFF0
pop     ecx
mov     ds:dword_462800, eax
push    ebp
mov     edx, offset sub_40FB8C
mov     eax, offset aVarNot ; "VarNot"
call    sub_40FFF0
pop     ecx
mov     ds:dword_462804, eax
push    ebp
mov     edx, offset sub_40FBC8
mov     eax, offset aVaradd ; "VarAdd"
call    sub_40FFF0
pop     ecx
mov     ds:dword_462808, eax
push    ebp
mov     edx, offset sub_40FBC8
mov     eax, offset aVarsub ; "VarSub"
call    sub_40FFF0
pop     ecx
mov     ds:dword_46280C, eax
push    ebp
mov     edx, offset sub_40FBC8
mov     eax, offset aVarMul ; "VarMul"
call    sub_40FFF0
pop     ecx
mov     ds:dword_462810, eax
push    ebp
mov     edx, offset sub_40FBC8
mov     eax, offset aVarDiv ; "VarDiv"
call    sub_40FFF0

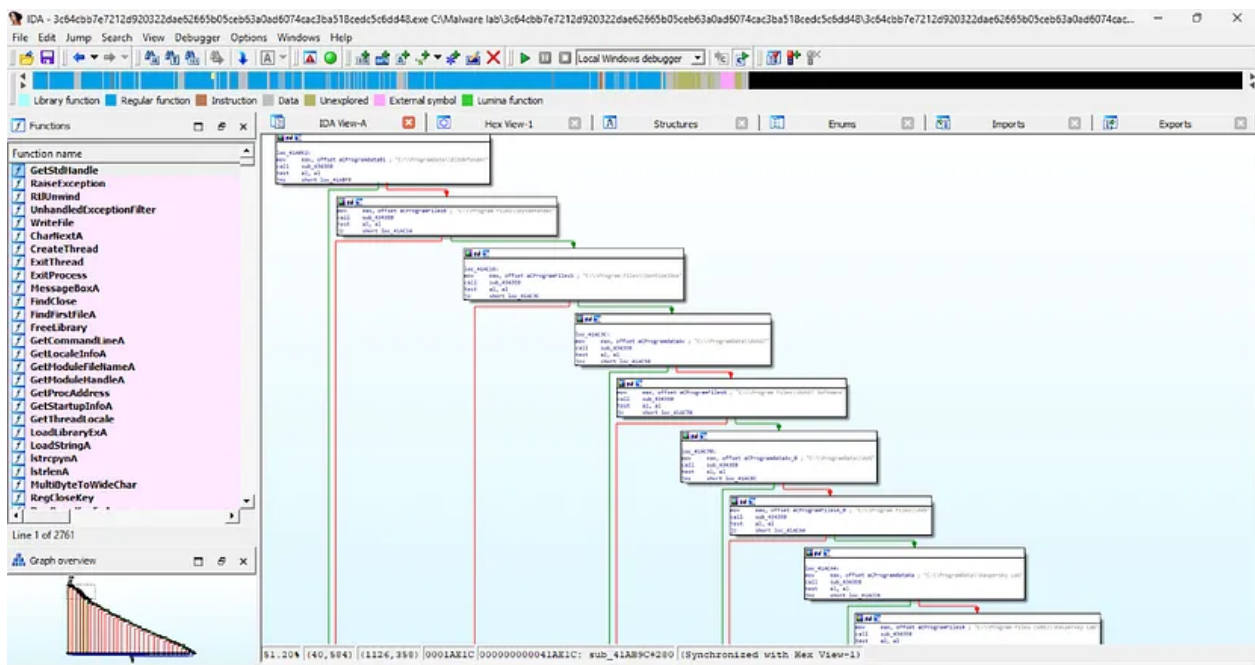
```

Loads Variant APIs from oleaut32.dll to parse dynamic data types

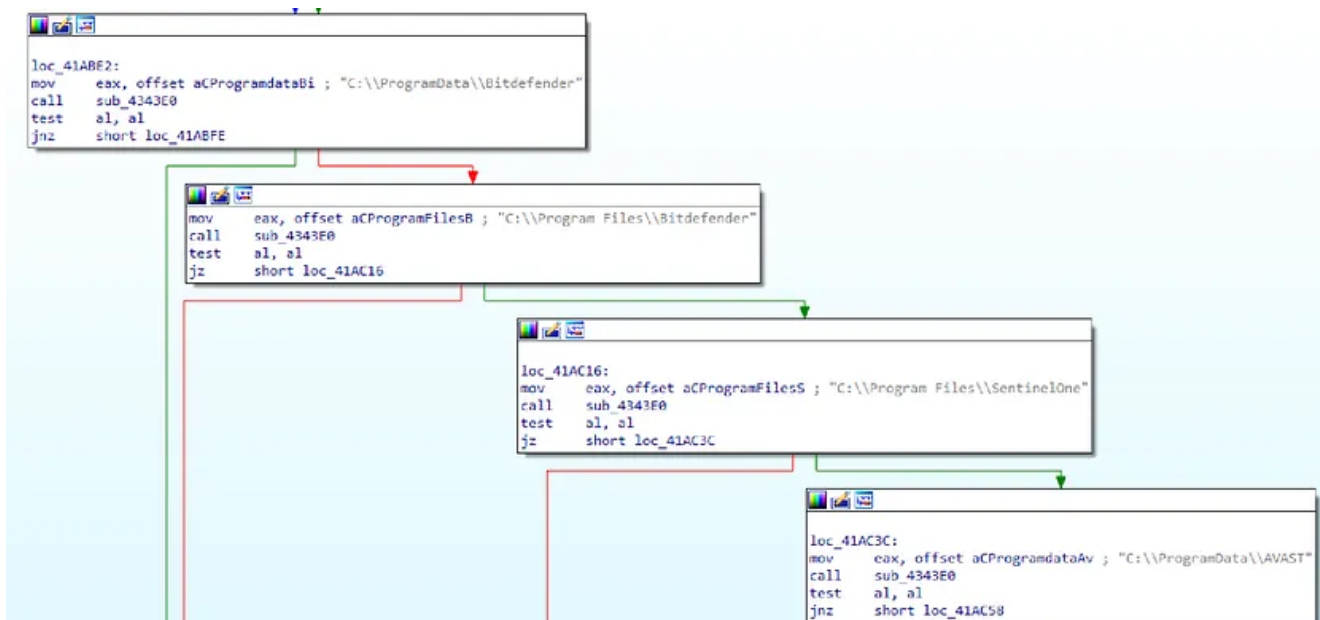
6. Security Software Evasion

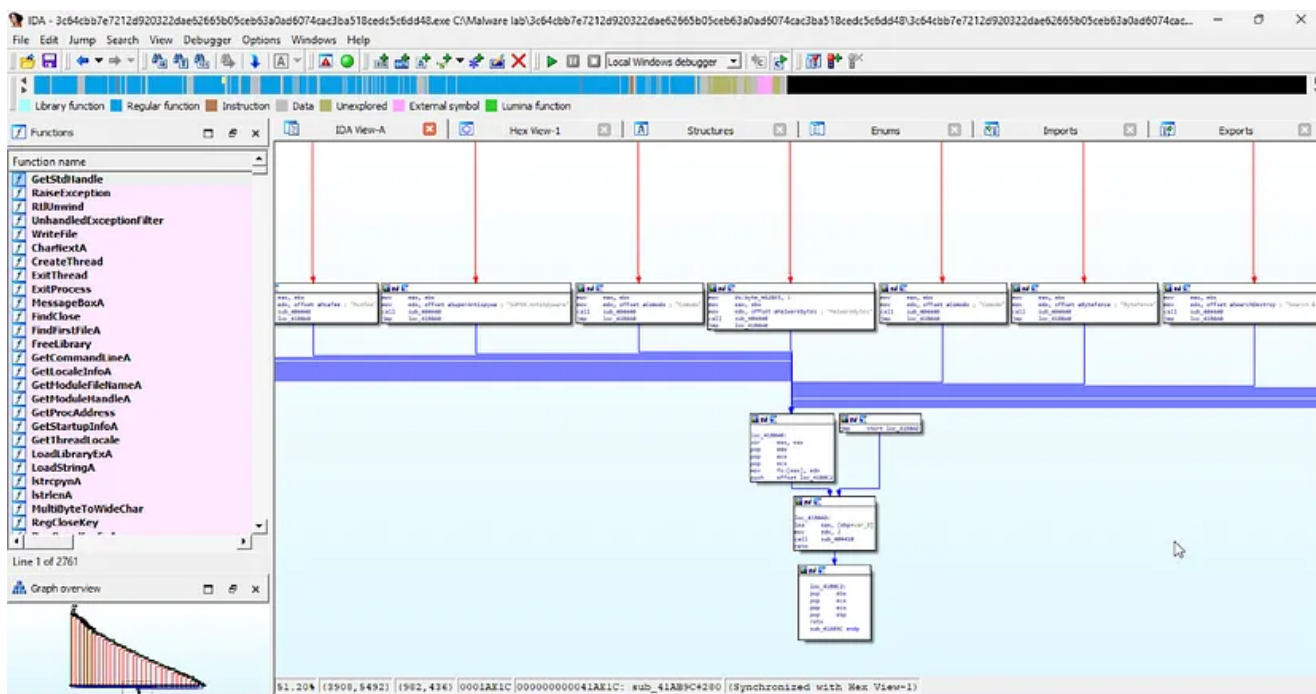
DarkGate systematically checks for directories and files associated with a wide array of antivirus products (Bitdefender, SentinelOne, Avast, AVG, Kaspersky, Norton, Symantec, Trend Micro, McAfee, SUPER AntiSpyware, Comodo, MalwareBytes, among others). This is a classic evasion technique:

- If security software is detected, DarkGate may alter its behavior, disable certain features, or even uninstall itself to avoid detection.
- By ensuring it does not operate in hostile environments, the malware increases its chances of long-term persistence.



Checks for antivirus and forensic tools in system directories

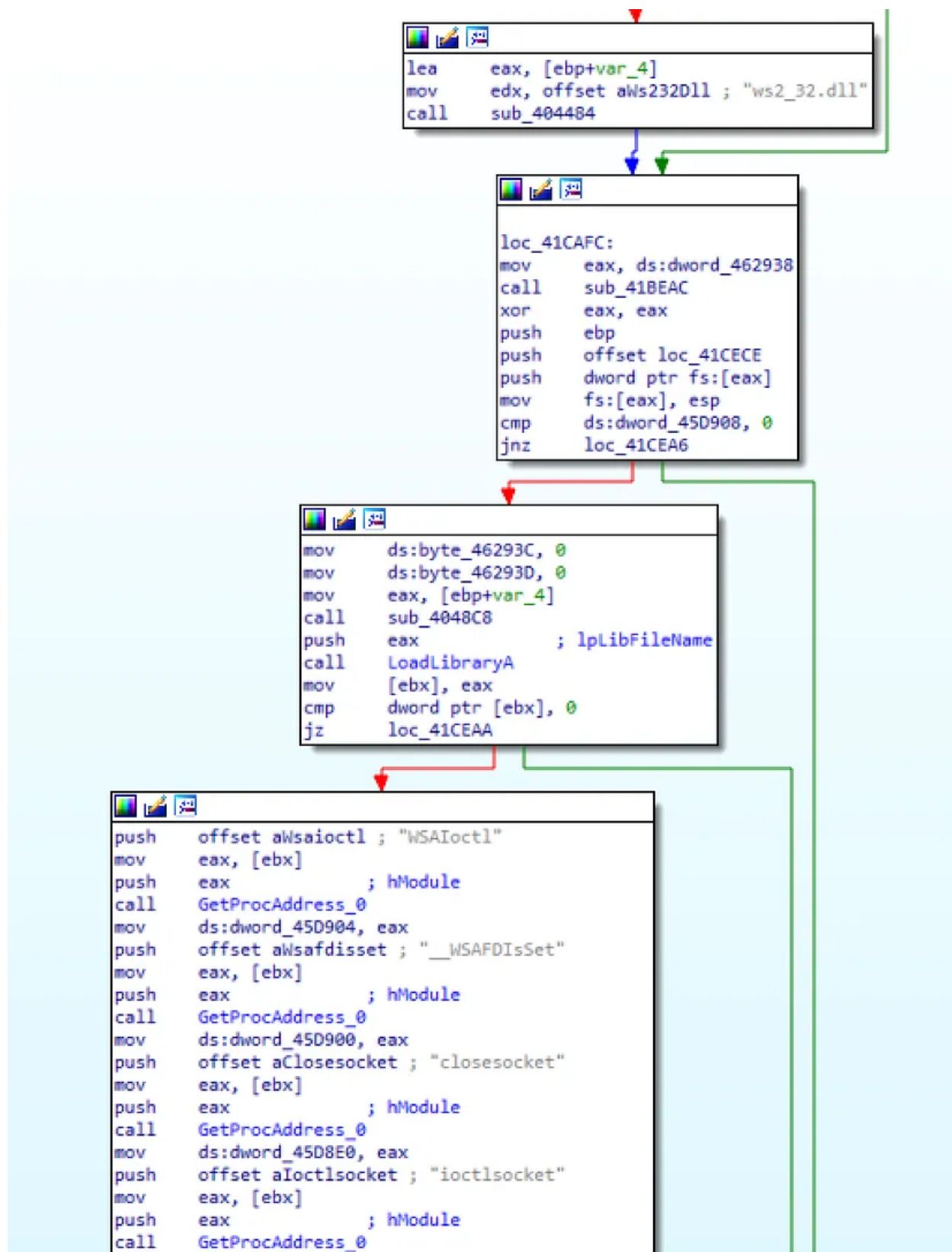




7. Stealthy Network Communication

The function sub_41CAC0 dynamically loads Winsock APIs (WSAStartup, send, recv) from ws2_32.dll at runtime. This approach:

- Avoids static detection by security solutions scanning for networking imports.
- Allows the malware to establish covert C2 channels, exfiltrate data, and receive commands while blending in with legitimate network traffic.



Winsock APIs resolved dynamically

Furthermore, DarkGate crafts its C2 traffic to mimic legitimate web traffic by:

- Using port 8080 (commonly associated with web services).

- Embedding a full “Mozilla/5.0...” User-Agent string. This enables its malicious communications to blend seamlessly into normal web traffic, significantly increasing its chances of bypassing network security measures.

```

CODE:004252B3      call     sub_4222C4
CODE:004252B8      mov     dword ptr [edi+10h], 493E0h
CODE:004252BF      lea     eax, [edi+8]
CODE:004252C2      mov     edx, offset dword_425364
CODE:004252C7      call    sub_404440
CODE:004252CC      lea     eax, [edi+48h]
CODE:004252CF      call    sub_4043EC
CODE:004252D4      lea     eax, [edi+4Ch]
CODE:004252D7      mov     edx, offset a8080 ; "8080"
CODE:004252DC      call    sub_404440
CODE:004252E1      lea     eax, [edi+50h]
CODE:004252E4      call    sub_4043EC
CODE:004252E9      lea     eax, [edi+54h]
CODE:004252EC      call    sub_4043EC
CODE:004252F1      lea     eax, [edi+24h]
CODE:004252F4      call    sub_4043EC
CODE:004252F9      lea     eax, [edi+28h]
CODE:004252FC      call    sub_4043EC
CODE:00425301      lea     eax, [edi+38h]
CODE:00425304      mov     edx, offset dword_425380
CODE:00425309      call    sub_404440
CODE:0042530E      mov     byte ptr [edi+3Ch], 1
CODE:00425312      mov     byte ptr [edi+44h], 0
CODE:00425316      lea     eax, [edi+60h]
CODE:00425319      mov     edx, offset aMozilla50Windo ; "Mozilla/5.0 (Windows NT 10.0; Win64; x6"...
CODE:0042531E      call    sub_404440
CODE:00425323      xor     eax, eax
CODE:00425325      mov     [edi+68h], eax
CODE:00425328      xor     eax, eax
CODE:0042532A      mov     [edi+6Ch], eax
CODE:0042532D      mov     byte ptr [edi+78h], 1
CODE:00425331      mov     dword ptr [edi+40h], 12Ch
CODE:00425338      mov     eax, edi

```

HTTP headers mimic browser traffic

8. Code Injection and Memory Residency

The function sub_427EE4 leverages low-level Windows APIs (NtWriteVirtualMemory, NtProtectVirtualMemory) to inject malicious code into other processes. This technique:

- Allows the malware to run without ever touching disk, making detection and forensic analysis much more difficult.
- Maintains control over the infected system even if the original process is terminated.

```

mov     [ebp+var_18], 5
lea     eax, [ebp+var_3C]
push    eax
push    offset aNtwritevirtual ; "NtWriteVirtualMemory"
call    sub_427840
mov     [ebp+var_14], eax
push    4
mov     [ebp+var_3C], ebx
mov     [ebp+var_38], 0
lea     eax, [ebp+var_C]
mov     [ebp+var_34], eax
mov     [ebp+var_30], 5
lea     eax, [ebp+var_8]
mov     [ebp+var_2C], eax
mov     [ebp+var_28], 5
mov     eax, [ebp+var_10]
mov     [ebp+var_24], eax
mov     [ebp+var_20], 0
lea     eax, [ebp+var_10]
mov     [ebp+var_1C], eax
mov     [ebp+var_18], 5
lea     eax, [ebp+var_3C]
push    eax
push    offset aNtprotectvirtu ; "NtProtectVirtualMemory"
call    sub_427840
mov     eax, [ebp+var_14]
call    sub_427E0C
test    al, al
jz      short loc_427FB4

```

```

mov     [ebp+var_4], 0FFFFFFFh
push    2
mov     [ebp+var_54], ebx
mov     [ebp+var_50], 0
mov     [ebp+var_4C], esi
mov     [ebp+var_48], 5
mov     [ebp+var_44], edi
mov     [ebp+var_40], 0
lea     eax, [ebp+var_54]
push    eax
push    offset aNtflushinstruc ; "NtFlushInstructionCache"
call    sub_427840

```

```

loc_427FB4:
mov     eax, [ebp+var_4]
pop     edi
pop     esi
pop     ebx
mov     esp, ebp

```

Injects shellcode using NtWriteVirtualMemory & runs in-memory.

9. Persistence Mechanisms

DarkGate ensures its continued execution through multiple persistence strategies:

- Uses SHGetSpecialFolderPathW with CSIDL_STARTUP and CSIDL_DESKTOP to locate standard Windows directories, then moves or copies itself using SHFileOperationW to these locations for automatic execution on startup or user login.
- Attempts to create entries under SOFTWARE\Microsoft\Windows\CurrentVersion\Run to guarantee launch at every system boot.
- Tries to run AutoHotkey.exe or Autolt3.exe with malicious scripts, leveraging legitimate automation tools to evade detection and facilitate persistence.

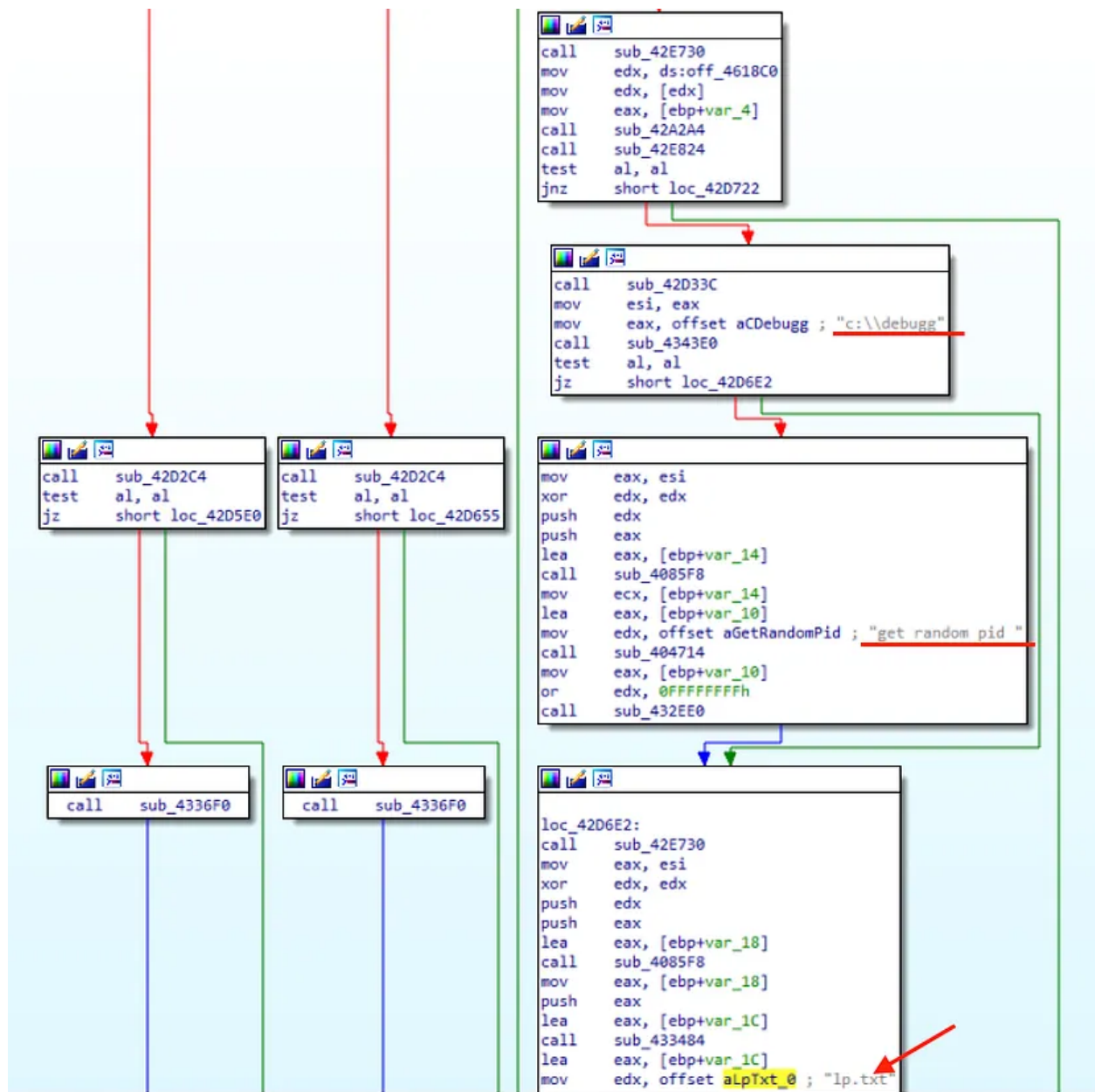
```
sub_42C98C proc near
var_10= dword ptr -10h
var_C= dword ptr -0Ch
pszPath= dword ptr -8
var_4= dword ptr -4

push    ebp
mov     ebp, esp
add     esp, 0FFFFFF0h
xor     edx, edx
mov     [ebp+var_10], edx
mov     [ebp+var_C], edx
mov     [ebp+var_4], eax
xor     eax, eax
push    ebp
push    offset loc_42CA57
push    dword ptr fs:[eax]
mov     fs:[eax], esp
lea     eax, [ebp+var_C]
call    sub_432E04
mov     edx, [ebp+var_C]
mov     eax, [ebp+var_4]
mov     ecx, offset aMicrosoftWindo ; "Microsoft\\Windows\\Start Menu\\Program"
call    sub_404714
mov     aMicrosoftWindo db 'Microsoft\\Windows\\Start Menu\\Programs\\Startup\\',0
mov     ; DATA XREF: sub_42C98C+2Df0
call    sub_4343C0
test    al, al
jnz     short loc_42CA3C
```

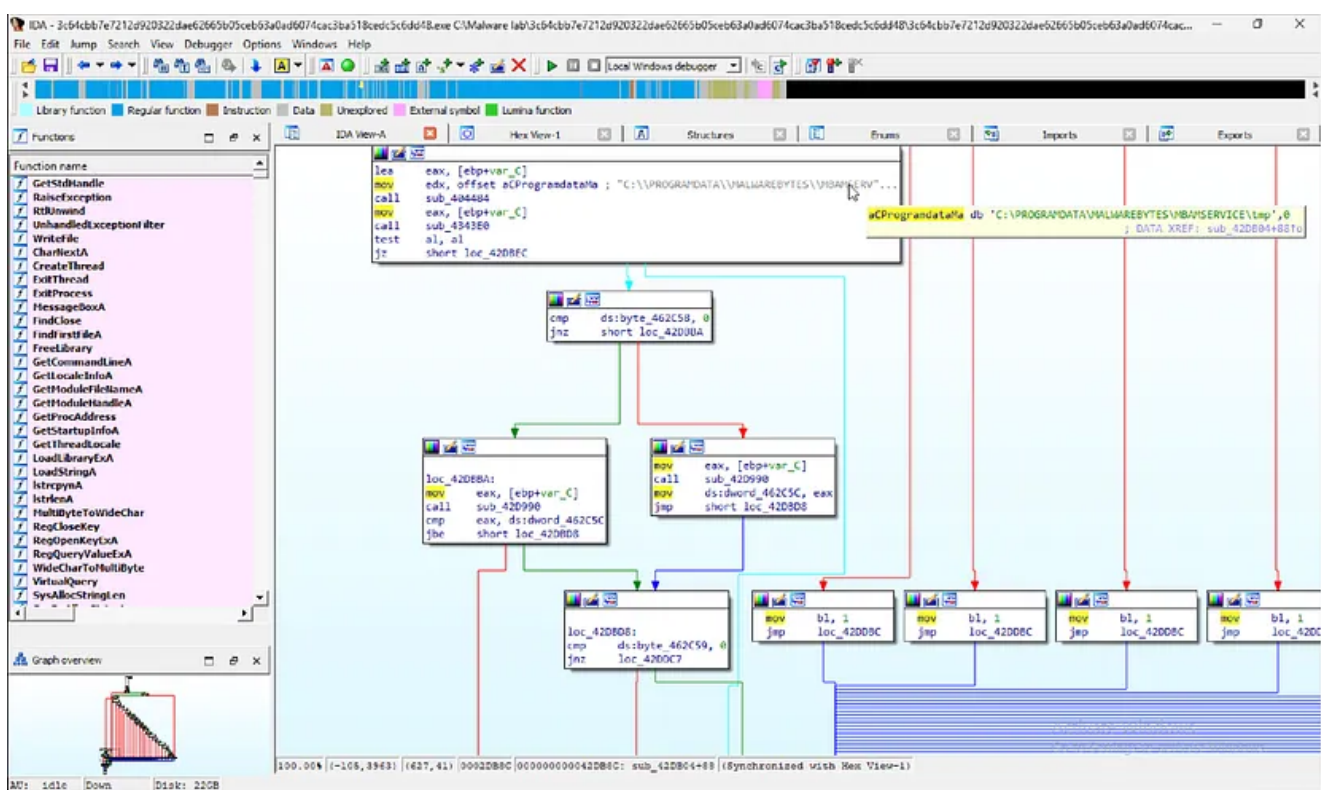
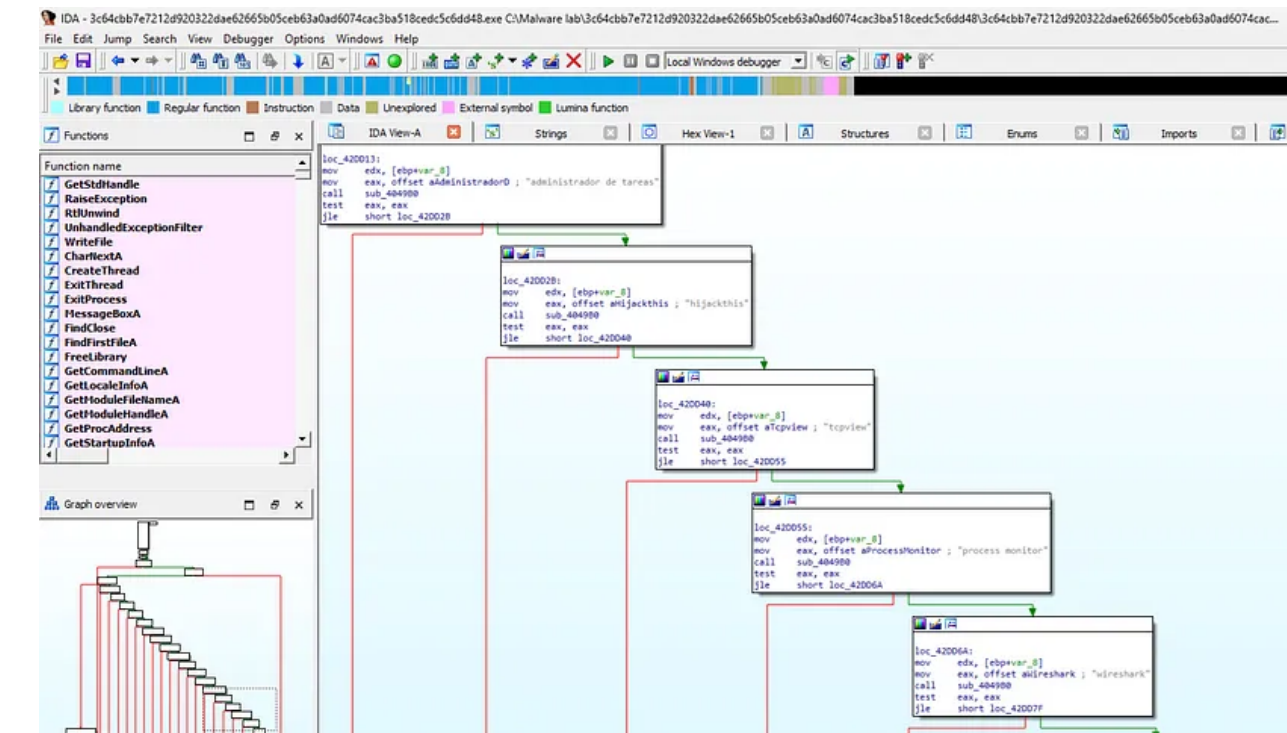
```
push    208h ; cb
call    CoTaskMemAlloc
mov     [ebp+pszPath], eax
xor     eax, eax
push    ebp
push    offset loc_42CA1D
push    dword ptr fs:[eax]
mov     fs:[eax], esp
push    0 ; fCreate
push    7 ; csidl
```

```
jmp     short loc_42CA13
```

SHGetSpecialFolderPathW and SHFileOperation used for startup persistence.



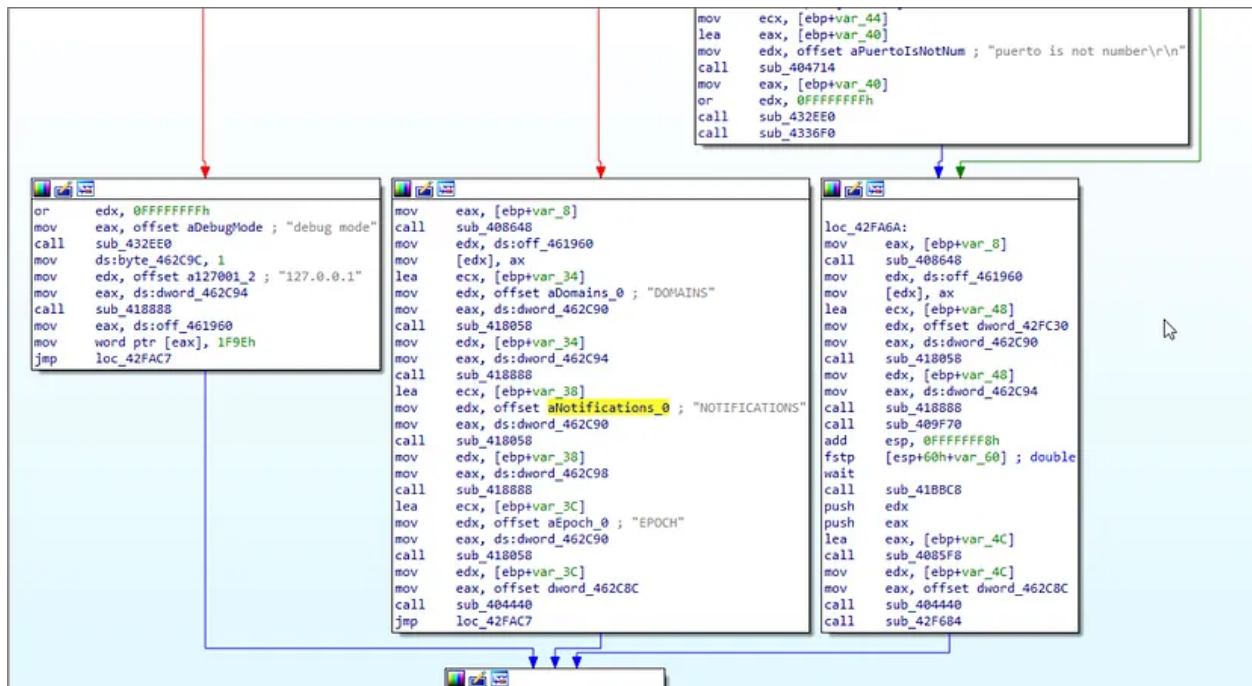
sub_42D594 checks for the presence of debugging tools



11. Configuration and Debug Modes

The initialization routine `sub_42F7A0` sets up operational directories within `C:\ProgramData\` (e.g., mainfolder, logsfolder, settings). It checks for a “debug mode” flag and attempts to connect to `127.0.0.1:8094` — likely a local C2 test or fallback channel. If debug mode is

disabled, it loads configuration parameters (such as C2 domains, notification settings, and epoch values) from files or the registry, validating port values and preparing for subsequent network communication.



The screenshot shows 127.0.0.1:8094 being used as a potential debug/test C2

12. Browser Data and Cookie Theft

DarkGate aggressively targets browser data:

- Searches for “chrome”, “edge”, and “brave” directories, specifically seeking “User Data” and “Default\Network\Cookies” paths.
- Iterates through multiple browser profiles to locate and exfiltrate cookies.
- By stealing cookies, DarkGate can bypass password-based authentication, enabling attackers to hijack user sessions on various platforms without needing actual credentials.

```

CODE:00455FFD      mov     eax, [ebp+var_8]
CODE:00456000      mov     edx, offset aChrome ; "chrome"
CODE:00456005      call    sub_404814
CODE:0045600A      jnz     short loc_456026
CODE:0045600C      lea     eax, [ebp+var_10]
CODE:0045600F      call    sub_432D3C
CODE:00456014      mov     edx, [ebp+var_10]
CODE:00456017      lea     eax, [ebp+var_C]
CODE:0045601A      mov     ecx, offset aGoogleChromeUs ; "Google\\Chrome\\User Data\\"
CODE:0045601F      call    sub_404714
CODE:00456024      jmp     short loc_456076
CODE:00456026      ; -----
CODE:00456026      loc_456026:      ; CODE XREF: sub_455FD0+3A↑j
CODE:00456026      mov     eax, [ebp+var_8]
CODE:00456029      mov     edx, offset aEdge ; "edge"
CODE:0045602E      call    sub_404814
CODE:00456033      jnz     short loc_45604F
CODE:00456035      lea     eax, [ebp+var_14]
CODE:00456038      call    sub_432D3C
CODE:0045603D      mov     edx, [ebp+var_14]
CODE:00456040      lea     eax, [ebp+var_C]
CODE:00456043      mov     ecx, offset aMicrosoftEdgeU ; "Microsoft\\Edge\\User Data\\"
CODE:00456048      call    sub_404714
CODE:0045604D      jmp     short loc_456076
CODE:0045604F      ; -----
CODE:0045604F      loc_45604F:      ; CODE XREF: sub_455FD0+63↑j
CODE:0045604F      mov     eax, [ebp+var_8]
CODE:00456052      mov     edx, offset aBrave ; "brave"
CODE:00456057      call    sub_404814
CODE:0045605C      jnz     short loc_456076
CODE:0045605E      lea     eax, [ebp+var_18]
CODE:00456061      call    sub_432D3C
CODE:00456066      mov     edx, [ebp+var_18]
CODE:00456069      lea     eax, [ebp+var_C]
CODE:0045606C      mov     ecx, offset aBravesoftwareB ; "BraveSoftware\\Brave-Browser\\User Data"...
CODE:00456071      call    sub_404714
CODE:00456076      loc_456076:      ; CODE XREF: sub_455FD0+54↑j
CODE:00456076      ; sub_455FD0+7D↑j ...
CODE:00456076      lea     eax, [ebp+var_1C]
CODE:00456079      mov     ecx, offset aDefault_0 ; "Default\\"
CODE:0045607E      mov     edx, [ebp+var_C]

```

This function locates browser profiles and cookie storage paths ([Network\\Cookies](#)) for exfiltration

13. Browser Manipulation and Cleanup

The function sub_456268 manages directories associated with Firefox, Chrome, Brave, and Opera. It uses cmd.exe to move or rename browser directories and delete files, employing Sleep calls to wait for completion. This serves multiple purposes:

- Steals browser data before cleanup.
- Deletes evidence to hinder recovery and post-infection analysis.
- The use of generic directory operations allows the malware to operate across different browser installations and user environments.

```

CODE:00456291      mov     edx, [ebp+var_C]
CODE:00456294      lea     eax, [ebp+var_4]
CODE:00456297      mov     ecx, offset aMozilla ; "Mozilla\\"
CODE:0045629C      call    sub_404714
CODE:004562A1      mov     eax, [ebp+var_4]
CODE:004562A4      call    sub_4343E0
CODE:004562A9      test    al, al
CODE:004562AB      jz      loc_456341
CODE:004562B1      mov     dl, 1
CODE:004562B3      mov     eax, offset aFirefoxExe ; "firefox.exe"
CODE:004562B8      call    sub_432884
CODE:004562BD      push    9C4h ; dwMilliseconds
CODE:004562C2      call    Sleep
CODE:004562C7      push    offset aCCdD ; "/c cd /d \""
CODE:004562CC      push    [ebp+var_4]
CODE:004562CF      push    offset aMoveFirefoxFir ; "\" && move firefox firefox"
CODE:004562D4      lea     edx, [ebp+var_14]
CODE:004562D7      mov     eax, 6
CODE:004562DC      call    sub_4328EC
CODE:004562E1      push    [ebp+var_14]
CODE:004562E4      lea     eax, [ebp+var_10]
CODE:004562E7      mov     edx, 4
CODE:004562EC      call    sub_404788
CODE:004562F1      mov     edx, [ebp+var_10]
CODE:004562F4      mov     eax, offset aCmdExe_2 ; "cmd.exe"
CODE:004562F9      call    sub_4312C0
CODE:004562FE      lea     eax, [ebp+var_18]
CODE:00456301      mov     ecx, offset aFirefox ; "firefox"
CODE:00456306      mov     edx, [ebp+var_4]
CODE:00456309      call    sub_404714
CODE:0045630E      mov     eax, [ebp+var_18]
CODE:00456311      call    sub_4343E0
CODE:00456316      test    al, al
CODE:00456318      jz      short loc_456341
CODE:0045631A      push    offset aCDelQFS ; "/c del /q /f /s "
CODE:0045631F      push    [ebp+var_4]
CODE:00456322      push    offset aFirefox_0 ; "firefox\\"
CODE:00456327      lea     eax, [ebp+var_1C]
CODE:0045632A      mov     edx, 3
CODE:0045632F      call    sub_404788
CODE:00456334      mov     edx, [ebp+var_1C]
CODE:00456337      mov     eax, offset aCmdExe_2 ; "cmd.exe"
CODE:0045633C      call    sub_4312C0

```

The malware uses `cmd.exe` to rename or delete browser directories

14. Credential Theft via cmdkey and NirSoft Tools

The subroutine `sub_456720` interacts directly with Windows credential management using `cmdkey`. It lists credentials to a temporary file and then deletes them, logging actions and waiting for operations to complete. This is a clear data exfiltration step, targeting stored Windows credentials for lateral movement or privilege escalation.

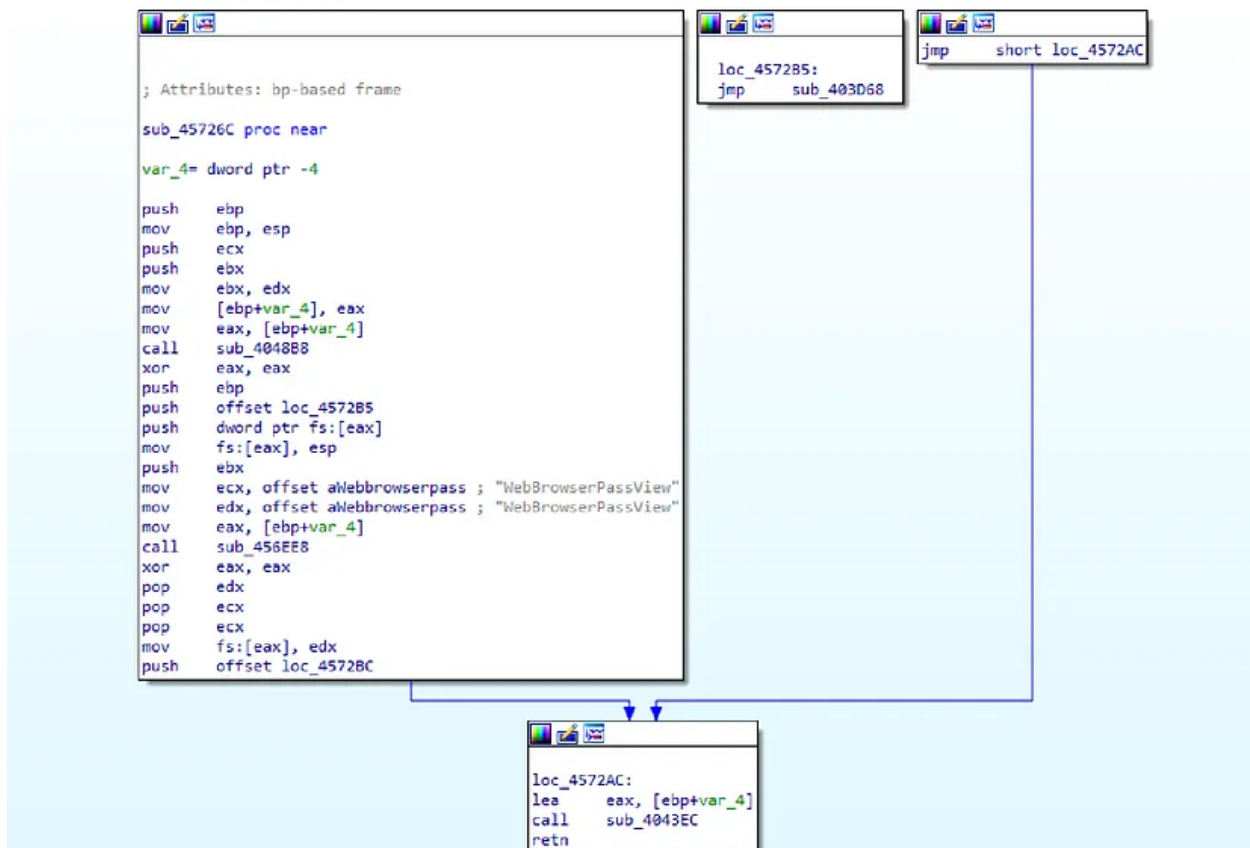
```

CODE:00456741    lea     eax, [ebp+var_14]
CODE:00456744    call   sub_430488
CODE:00456749    mov     eax, [ebp+var_14]
CODE:0045674C    mov     edx, offset dword_456984
CODE:00456751    call   sub_404814
CODE:00456756    jnz     short loc_456767
CODE:00456758    mov     eax, offset aDeleteCredenti ; "Delete Credentials not worked because I do not have Admin Rights",0
CODE:0045675D    call   sub_426E38
CODE:00456762    aDeleteCredenti db 'Delete Credentials not worked because I do not have Admin Rights',0
CODE:00456767    ; -- ; DATA XREF: sub_456720+381to
CODE:00456767    loc_456767: ; CODE XREF: sub_456720+361j
CODE:00456767    lea     eax, [ebp+var_C]
CODE:0045676A    mov     edx, offset aCTempCredIxt ; "c:\\temp\\cred.txt"
CODE:0045676F    call   sub_404484
CODE:00456774    push    0
CODE:00456776    push    1
CODE:00456778    lea     eax, [ebp+var_18]
CODE:0045677B    mov     ecx, [ebp+var_C]
CODE:0045677E    mov     edx, offset aCCmdkeyList ; "/c cmdkey /list > "
CODE:00456783    call   sub_404714
CODE:00456788    mov     edx, [ebp+var_18]
CODE:0045678B    xor     ecx, ecx
CODE:0045678D    mov     eax, offset aCmdExe_3 ; "cmd.exe"
CODE:00456792    call   sub_431344
CODE:00456797    mov     eax, [ebp+var_C]
CODE:0045679A    call   sub_434004
CODE:0045679F    test    al, al
CODE:004567A1    jnz     short loc_4567C0
CODE:004567A3    lea     eax, [ebp+var_1C]
CODE:004567A6    mov     ecx, offset aNotExists ; " not exists"
CODE:004567AB    mov     edx, [ebp+var_C]
CODE:004567AE    call   sub_404714
CODE:004567B3    mov     eax, [ebp+var_1C]
CODE:004567B6    call   sub_426E38
CODE:004567B9    jmp     loc_45696A
CODE:004567C0    ; -----
CODE:004567C0    loc_4567C0: ; CODE XREF: sub_456720+811j
CODE:004567C0    mov     dl, 1
CODE:004567C2    mov     eax, off_4168D0
CODE:004567C7    call   sub_403680
CODE:004567CC    mov     [ebp+var_4], eax
CODE:004567CF    mov     edx, [ebp+var_C]

```

cmdkey

The functions sub_4571CC and sub_45726C automate the use of NirSoft's Mail PassView and WebBrowserPassView, extracting stored passwords from email clients and web browsers. This demonstrates DarkGate's ability to leverage legitimate tools for malicious purposes, maximizing credential theft with minimal custom code.



WaveIn API calls (e.g., **waveInOpen**) initialize audio capture from the system microphone

15. Audio Recording

The function sub_4577E0 enables DarkGate to record audio from the victim's microphone. By calling Windows multimedia APIs (waveInOpen, waveInPrepareHeader, waveInAddBuffer), the malware initializes audio input, sets up buffers, and starts capturing sound. This capability extends DarkGate's surveillance reach, allowing attackers to eavesdrop on conversations and ambient sounds in the victim's environment.

```
mov     edx, ds:dword_463060
mov     [eax+4], edx
push    20h ; ' ' ; cbwh
mov     eax, ds:pwh
push    eax ; pwh
mov     eax, ds:hwi
push    eax ; hwi
call    waveInPrepareHeader
test    eax, eax
jnz     short loc_4578B1
```

```
push    20h ; ' ' ; cbwh
mov     eax, ds:pwh
push    eax ; pwh
mov     eax, ds:hwi
push    eax ; hwi
call    waveInAddBuffer
test    eax, eax
jnz     short loc_4578B1
```

```
mov     eax, ds:hwi
push    eax ; hwi
call    waveInStart
test    eax, eax
jnz     short loc_4578B1
```

```
mov     bl, 1
```

```
loc_4578B1:
mov     eax, ebx
add     esp, 0Ch
pop     esi
```

WaveIn API calls (e.g., `waveInOpen`) initialize audio capture from the system microphone

DarkGate Malware — Key Functions and Capabilities

Function	Purpose	API Usage
Risk		
resolution & evasion	`GetLongPathNameA`, `FindFirstFileA`, `lstrcpynA`	sub_405A20 Path
Medium	sub_40F7C8 Process/thread/module enumeration	
	`CreateToolhelp32Snapshot`, `Process32First`, `Toolhelp32ReadProcessMemory`	High
sub_410028	COM and data type handling	`VariantChangeTypeEx`,
	`VarBstrFromStr`, `oleaut32.dll` APIs	Medium
sub_41CAC0		C2
communication setup	`WSAStartup`, `send`, `recv`, HTTP headers,	
port 8080	High	sub_427EE4 Code injection & memory execution
	`NtWriteVirtualMemory`, `NtProtectVirtualMemory`	High
sub_42D594	Anti-debugging detection	`Sleep`, debugger tool
checks		High
sub_42DB04		Anti-
analysis & security tool scanning	Strings: `Wireshark`, `Procmon`, `Regedit`, etc.	
Medium	sub_42F7A0 Debug mode & configuration loading	
	`CreateDirectoryW`, `GetPrivateProfileStringW`, registry	Low
sub_456720	Windows credential theft	`cmdkey /list`, file
output, `cmdkey /delete`		High
sub_4571CC		Browser
credential theft	`Mail PassView`, `WebBrowserPassView` automation	
High	sub_45726C Email password theft	`Mail PassView`
usage		High
sub_4577E0		Audio
recording from microphone	`waveInOpen`, `waveInStart`, `waveInAddBuffer`	
High	sub_456268 Browser data deletion & cleanup	`cmd.exe`,
	`del`, `move`, `Sleep`	Medium
sub_42C98C		
Persistence via filesystem	`SHGetSpecialFolderPathW`,	
`SHFileOperationW`		High
sub_42E03C		Persistence via
AutoHotkey	`AutoHotkey.exe`, `.ahk`/.a3x` scripts, registry `Run` keys	
High		

MITRE ATT&CK Mapping

MITRE ID	Technique	Description
T1566.001	Phishing: Spearphishing Attachment	Initial infection via malicious email with a disguised payload.
T1059.005	Command & Scripting: AutoIt	Uses AutoIt scripts for persistence and execution.
T1055	Process Injection	Injects shellcode using NT API calls.
T1027	Obfuscated Files or Information	High entropy and dynamic API resolution to evade static analysis.
T1562.001	Disable or Modify Tools	Detects tools like Wireshark and Process Monitor.
T1056.001	Input Capture: Keylogging	Logs keystrokes and cursor activity.
T1555.003	Credentials from Web Browsers	Extracts saved passwords using NirSoft tools.
T1005	Data from Local System	Harvests cookies and credential files from disk.
T1071.001	Application Layer Protocol: Web Protocols	C2 communication over HTTP using spoofed User-Agent headers.
T1547.001	Registry Run Keys / Startup Folder	Establishes persistence via registry and startup locations.
T1123	Audio Capture	Records audio through Windows multimedia APIs.

DarkGate IOCs List

Registry Keys

- SOFTWARE\Borland\Delphi\RTL
- SOFTWARE\Microsoft\Windows\CurrentVersion\Run
- SOFTWARE\Microsoft\Windows NT\CurrentVersion
- Control Panel\Desktop\WindowMetrics

Persistence & Execution Artifacts

- AutoHotkey.exe
- Autoit3.exe
- Microsoft\Windows\Start Menu\Programs\Startup\

Credential and Data Theft

- cmdkey /list >
- cmdkey /delete:
- Mail PassView, MailPassView
- Network Password Recovery
- NetPass
- Default\Network\Cookies
- Google\Chrome\User Data\
- BraveSoftware\Brave-Browser\User Data\
- Microsoft\Edge\User Data\

- Mozilla\
- Opera Software

AV/EDR Detection & Evasion

- Bitdefender
- Avast
- AVG
- Kaspersky
- Norton
- Panda Security
- MalwareBytes
- SentinelOne
- ESET
- Avira
- F-Secure
- McAfee
- Comodo
- IObit Malware Fighter
- Emsisoft
- Quick Heal
- G DATA
- Sophos
- ByteFence

File System & Temporary Artifacts

- C:\Program Files\Bitdefender
- C:\Program Files\AVAST Software
- C:\Program Files\AVG
- C:\Program Files\Kaspersky Lab
- C:\Program Files\Malwarebytes
- C:\Program Files\SentinelOne
- C:\Program Files (x86)\Avira
- C:\Program Files (x86)\F-Secure
- C:\Program Files\Quick Heal
- C:\Program Files\ESET
- C:\Program Files\Emsisoft
- C:\Program Files\G DATA
- C:\Program Files\Sophos
- C:\ProgramData\Bitdefender
- C:\ProgramData\AVAST

- C:\ProgramData\AVG
- C:\ProgramData\Kaspersky Lab
- C:\ProgramData\ESET
- C:\ProgramData\Emsisoft
- C:\ProgramData\G DATA
- C:\ProgramData\Sophos
- C:\temp\

Command-Line & Process Injection

- /c cmdkey /list >
- /c cmdkey /delete:
- /c del /q /f /s
- /c ping 127.0.0.1 & del /q /f /s c:\temp & del /q /f /s
- /c cd /d \
- /c shutdown -f -r -t 0
- /c shutdown -f -s -t 0

C2 Communication & Network

- Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.0.0 Safari/537.36 (User-Agent)
- HTTP/1.0, HTTP/
- Authorization: Basic
- Proxy-Authorization: Basic
- 127.0.0.1
- 0.0.0.0
- 255.255.255.255

Other Notable Strings

- :::Clipboard::: (clipboard data marker)
- .0xCrypt (potential cryptographic or obfuscation marker)
- Build
- EPOCH
- NOTIFICATIONS

File Names and Dropped Artifacts

- ccleaner, system config, malwarebytes, farbar recovery, avast, startup, rootkit, autoruns, editor de registro, editor del registro, registry editor, gerenciador de tarefas, zhpcleaner, task manager, junkware removal, administrador de tareas, hijackthis, tcpview, process monitor, wireshark, taskmanager

- Phishing and lure files: Navigating Future Changes October 2023.pdf.msi, clarify_27-May_{6 random digits}.html, Job description_salary_policy_marketing products_new_list_2023.zip
- Temporary/working directories: C:\test\, C:\ProgramData\cccddcb\

Conclusion

DarkGate is a stealthy and modular malware that combines persistence, credential theft, and evasion in a compact MaaS package. Even with static analysis alone, it was possible to uncover key capabilities like Autolt-based persistence, C2 communication, and data exfiltration. These findings highlight the malware's sophistication and the value of manual reverse engineering.

References

-
-
- Sandbox execution () shows immediate downloader behavior, rapid persistence establishment, and swift command-and-control (C2) initiation within seconds of launch.