JOVITA
CYBER SECURITY

MALWARE ANALYSIS REPORT

Presented for :
IDN Bootcamp Cyber

Author :
Jovita Kusuma

Date :
16-07-2025

Version:
1.0

Property	Value
File Name	Lab01-01.exe
SHA256	58898bd42c5bd3bf9b1389f0eee5639cd59180e0b330ea0b327bd6fe47
Size	16 KiB
Type	PE executable
Original Submission	2017-07-05 22:14:28 (UTC)
Last AV Scan	2025-06-14 00:16:49 (UTC)
Threat Score	100/100
AV Detection Rate	88%
Label	Trojan Generic

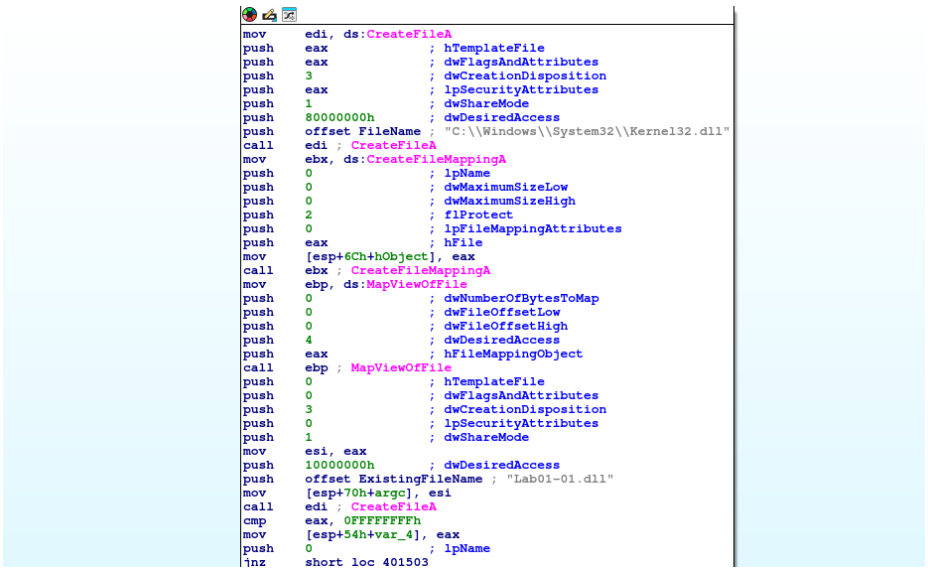
1. Pendahuluan

Laporan ini membahas analisis *reverse engineering* terhadap beberapa potongan kode assembly yang saling terhubung, yang diekstrak dari tiga berkas ZIP yang dilindungi kata sandi (kata sandi: "infected"): jadigini.zip (berisi Lab01-01.exe), walawe.zip (berisi Lab01-01.dll), dan wheheh.zip (berisi Lab01-02.bin). Tujuan dari tugas ini adalah untuk mempelajari fungsi dari malware, memahami perilakunya, serta merencanakan strategi mitigasi yang efektif.

Kode yang dianalisis menunjukkan karakteristik khas *malware* yang berupa *multi-stage payload* yang dirancang untuk persistensi, *command-and-control (C2)*, dan penghindaran deteksi. Analisis menunjukkan proses percobaan berupa manipulasi sistem file, pemetaan memori, obfuscation, dan penggunaan *command-and-control* (C2).

2. Kompromi Awal dan Mekanisme Persistensi

Pada tahap awal, kode yang dianalisis berfokus pada akses persistent pada sistem yang telah terkompromi:



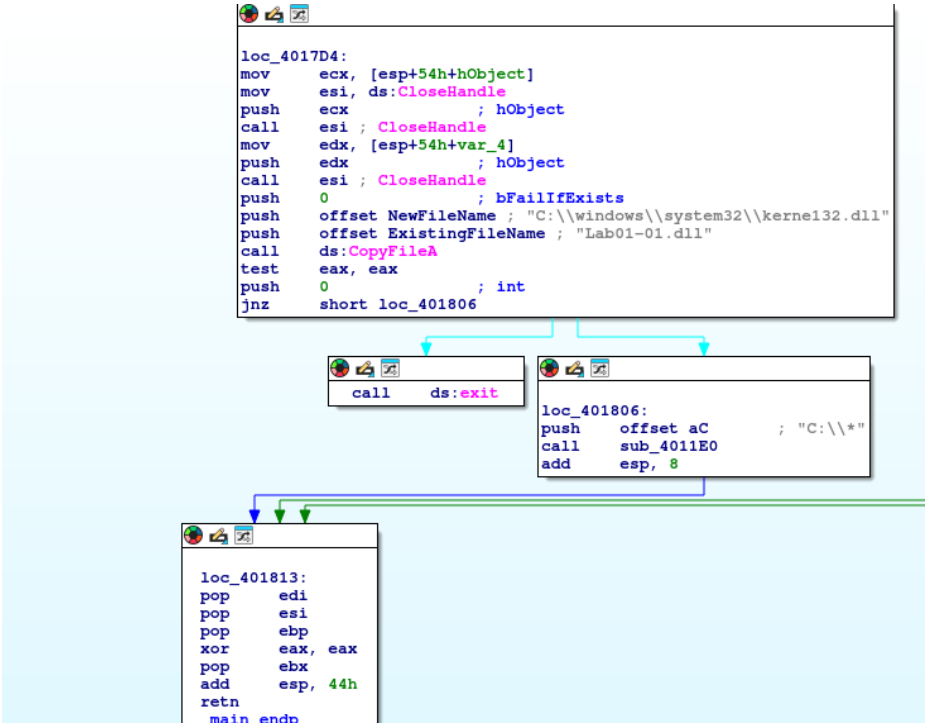
```
mov     edi, ds:CreateFileA
push    eax             ; hTemplateFile
push    eax             ; dwFlagsAndAttributes
push    3               ; dwCreationDisposition
push    eax             ; lpSecurityAttributes
push    1               ; dwShareMode
push    80000000h       ; dwDesiredAccess
push    offset FileName ; "C:\\Windows\\System32\\Kernel32.dll"
call    edi ; CreateFileA
mov     ebx, ds:CreateFileMappingA
push    0               ; lpName
push    0               ; dwMaximumSizeLow
push    0               ; dwMaximumSizeHigh
push    2               ; flProtect
push    0               ; lpFileMappingAttributes
push    eax             ; hFile
mov     [esp+6Ch+hObject], eax
call    ebx ; CreateFileMappingA
mov     ebp, ds:MapViewOfFile
push    0               ; dwNumberOfBytesToMap
push    0               ; dwFileOffsetLow
push    0               ; dwFileOffsetHigh
push    4               ; dwDesiredAccess
push    eax             ; hFileMappingObject
call    ebp ; MapViewOfFile
push    0               ; hTemplateFile
push    0               ; dwFlagsAndAttributes
push    3               ; dwCreationDisposition
push    0               ; lpSecurityAttributes
push    1               ; dwShareMode
mov     esi, eax
push    10000000h       ; dwDesiredAccess
push    offset ExistingFileName ; "Lab01-01.dll"
mov     [esp+70h+argc], esi
call    edi ; CreateFileA
cmp     eax, 0FFFFFFFFh
mov     [esp+54h+var_4], eax
push    0               ; lpName
jnz     short loc_401503
```

- **Pembukaan dan Pemetaan File**: Program memulai dengan membuka dua file penting:
  - C:\Windows\System32\Kernel32.dll : merupakan library dari core sistem Windows yang esensial untuk hampir semua proses. File ini dibuka dengan akses GENERIC_READ.
  - Lab01-01.dll : DLL khusus yang kemungkinan besar berisi *payload* berbahaya. File ini dibuka dengan akses GENERIC_WRITE, menunjukkan percobaan untuk memodifikasi atau membuat file.

```
call    ds:exit

loc_401503:             ; dwMaximumSizeLow
push    0
push    0               ; dwMaximumSizeHigh
push    4               ; flProtect
push    0               ; lpFileMappingAttributes
push    eax             ; hFile
call    ebx ; CreateFileMappingA
cmp     eax, 0FFFFFFFFh
push    0               ; dwNumberOfBytesToMap
jnz     short loc_40151B

call    ds:exit

loc_40151B:             ; dwFileOffsetLow
push    0
push    0               ; dwFileOffsetHigh
push    0F001Fh         ; dwDesiredAccess
push    eax             ; hFileMappingObject
call    ebp ; MapViewOfFile
mov     ebp, eax
test    ebp, ebp
mov     [esp+54h+argv], ebp
jnz     short loc_401538
```

- **Pemetaan Memori**: Kedua file tersebut kemudian dipetakan ke dalam proses virtual address menggunakan CreateFileMappingA dan MapViewOfFile.
  - Kernel32.dll dipetakan dengan hak akses PAGE_READONLY dan FILE_MAP_READ, mengindikasikan bahwa program berniat untuk membaca atau menganalisis isinya.

- Lab01-01.dll dipetakan dengan hak akses PAGE_WRITECOPY, FILE_MAP_READ | FILE_MAP_WRITE | FILE_MAP_EXECUTE. Ini mengindikasikan kemungkinan adanya injeksi kode atau manipulasi kode di memori.



- **DLL Hijacking/Penggantian File**: Tindakan paling krusial untuk persistensi adalah pemanggilan CopyFileA, yang mencoba menyalin Lab01-01.dll ke lokasi C:\Windows\System32\Kernel32.dll. Parameter bFailIfExists diatur ke 0, artinya Kernel32.dll yang asli akan ditimpa jika sudah ada. Ini adalah bentuk ekstrem dari *DLL hijacking*, yang membuat DLL berbahaya akan dijalankan oleh hampir setiap program di sistem yang menggunakan Kernel32.dll.

3. Mekanisme Obfuscation dan Unpacking

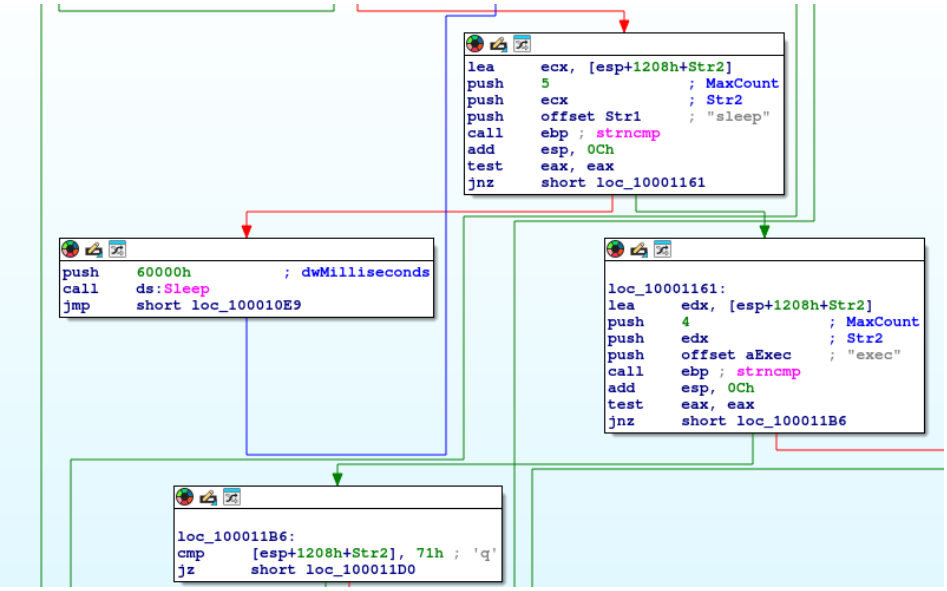
Sebagian besar dari kode yang dianalisis ditujukan untuk *unpacking* atau *deobfuscating* payload tersembunyi:

- **Loop Kompleks dan Transformasi Data**: Kode mengandung nested loop dengan operasi aritmatika dan bitwise yang rumit (misalnya sub, adc, add, rol, xor, xchg), yang diterapkan pada blok memori.
- **Dekripsi/Deobfuscation**: Pola operasi menunjukkan dengan jelas adanya algoritma dekripsi

atau deobfuscation. Kemungkinan besar program sedang membuka *payload* berbahaya yang dienkripsi atau disamarkan.

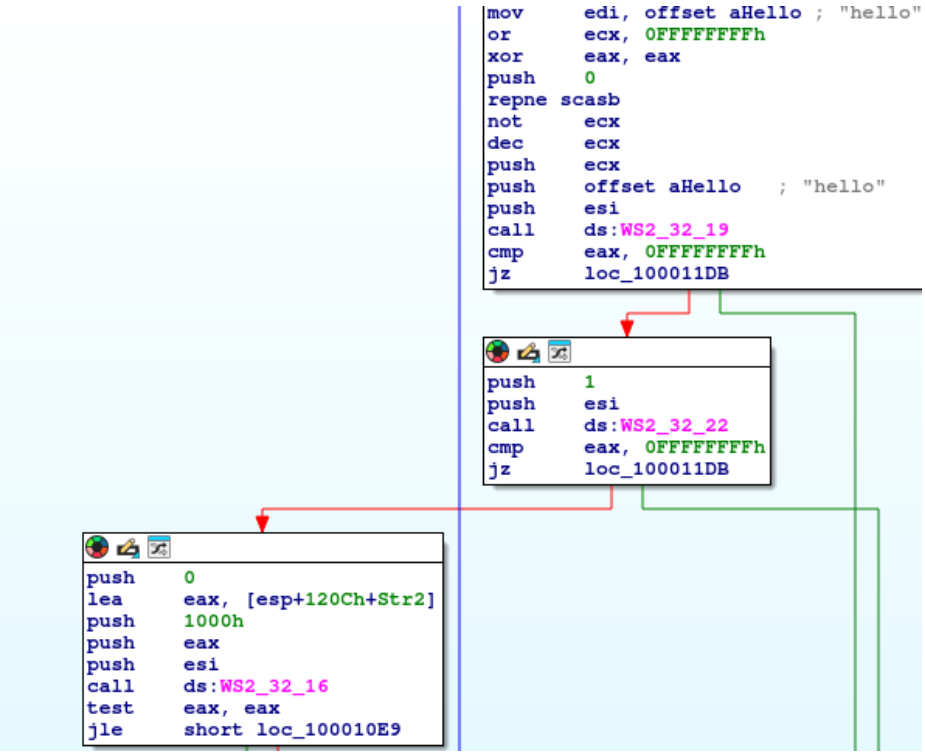
- **Eksekusi Dinamis**: Setelah proses dekripsi, kode melakukan pemanggilan fungsi tidak langsung (call dword ptr [esi+5064h]). Alamat eksekusi ditentukan pada saat runtime, mengarah ke kode berbahaya yang baru didekripsi. Ini adalah teknik umum untuk menghindari analisis statis.

4. Fungsi Client Command and Control (C2)



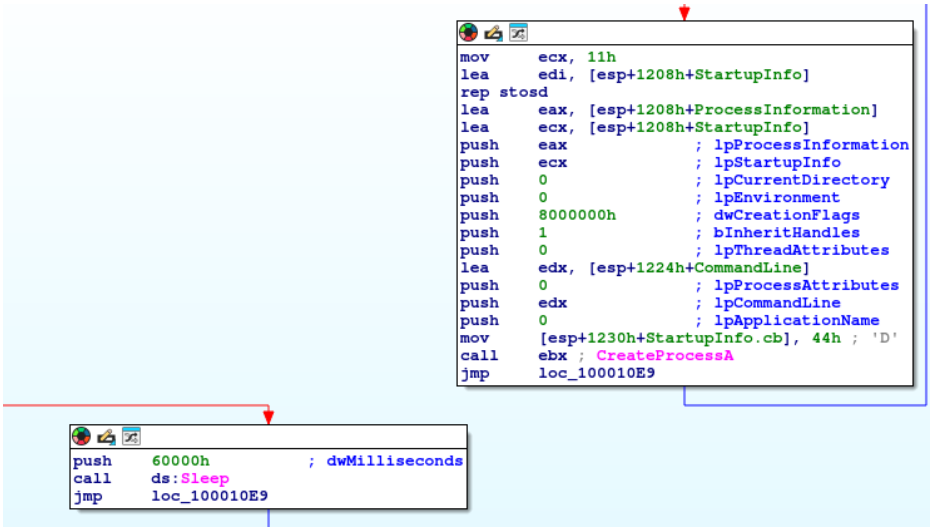
Setelah payload dijalankan, program mengalami transisi menjadi klien C2:

- **Single Instance Enforcement**: Program menggunakan *named mutex* ("SADFHUHF") untuk memastikan hanya satu instansi yang berjalan. Jika mutex sudah ada, program langsung keluar.
- **Komunikasi Jaringan (WS2_32)**: Kode menggunakan banyak fungsi dari library WS2_32, mengindikasikan adanya komunikasi jaringan aktif.
  - **IP Server C2**: Program mencoba terhubung ke IP yang telah dikodekan: 127.26.152.13. Alamat dari IP ini adalah IP privat, kemungkinan besar server lokal untuk pengujian atau bagian dari kompromi jaringan internal.



- **Beaconing**: Program mengirim pesan "hello" ke server C2 untuk menandakan kehadirannya dan siap menerima perintah.
- **Command Loop**: Program masuk ke dalam loop untuk menerima dan memproses perintah.

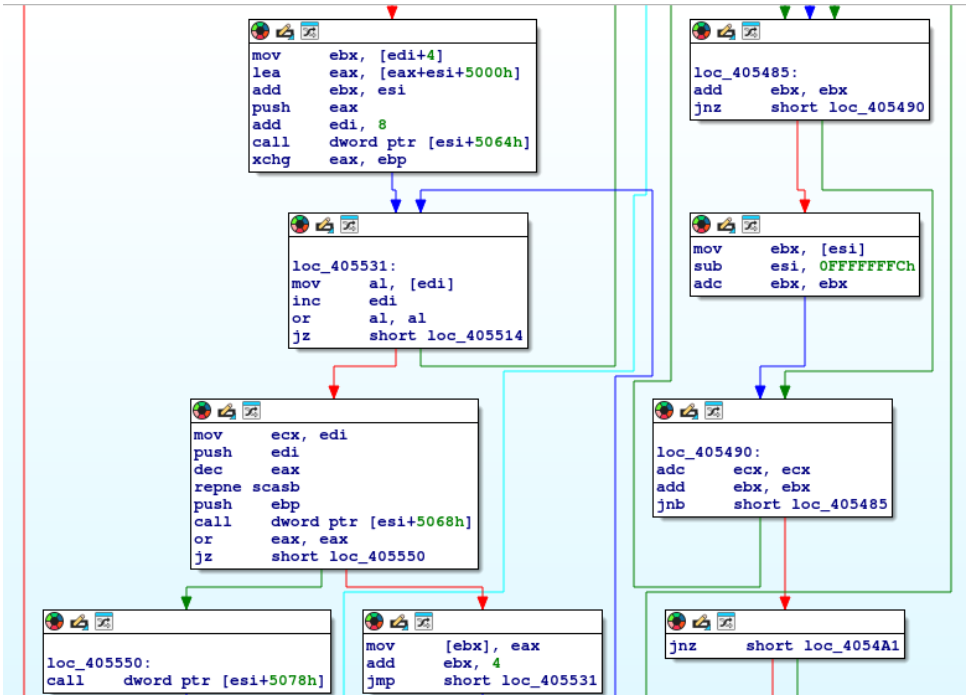
- **Pemrosesan Perintah**



- **"sleep"**: Program akan berhenti selama ~65 detik (60000h miliseconds) menggunakan ds:Sleep.

- **"exec"**: Program menggunakan CreateProcessA untuk menjalankan proses baru berdasarkan perintah dari server. Baris perintah untuk proses baru ini diterima dari server C2, memungkinkan penyerang untuk mengeksekusi perintah apa pun secara bebas pada sistem yang telah dikompromikan. Flag CREATE_NO_WINDOW (8000000h) digunakan agar proses yang dijalankan berjalan secara diam-diam di latar belakang tanpa menampilkan jendela apa pun.
- **'q' (Quit)**: Program menutup diri dengan membersihkan sumber daya jaringan (menggunakan WSACleanup, closesocket, dll.).

5. API Windows yang Diidentifikasi

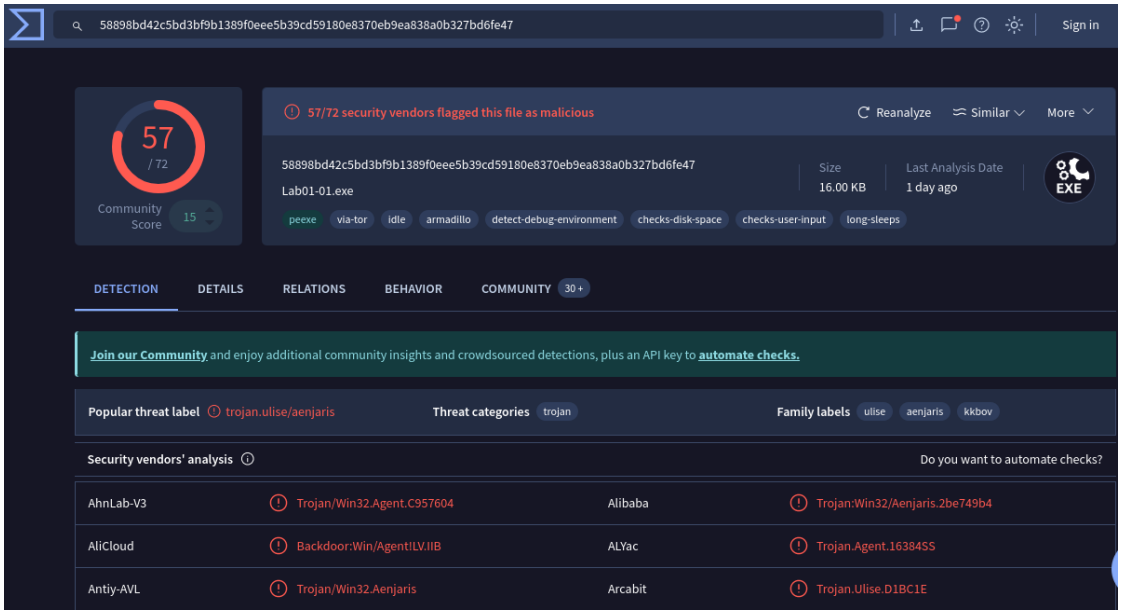


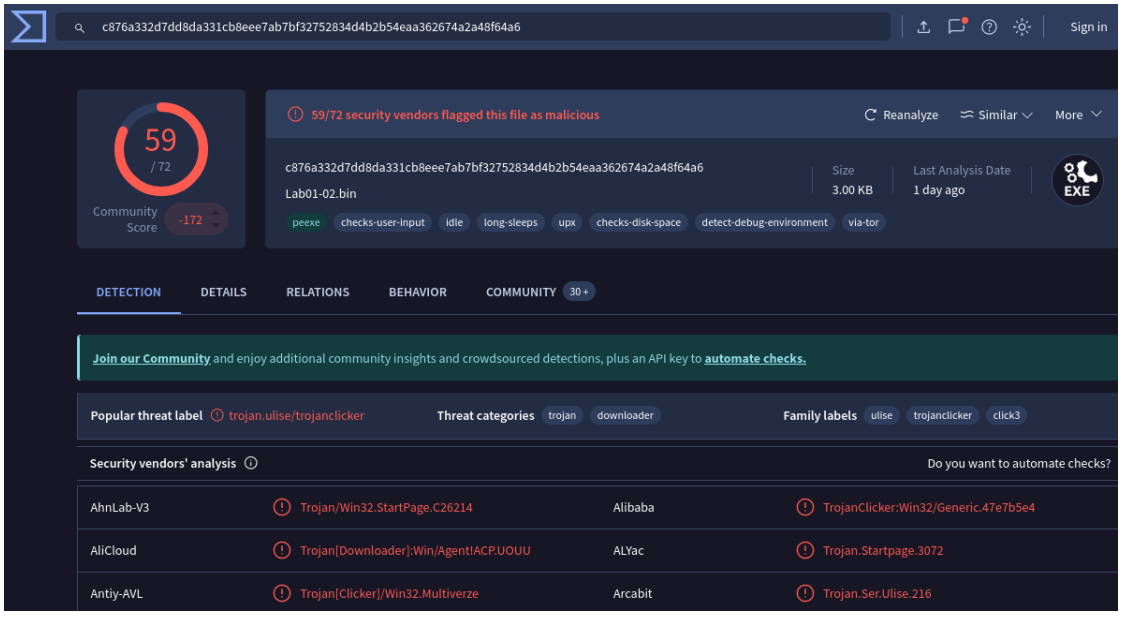
Analisis menunjukkan penggunaan beberapa fungsi critical API Windows berikut:

- CreateFileA: Membuka atau membuat file.
- CreateFileMappingA: Membuat objek pemetaan file.
- MapViewOfFile: Memetakan tampilan file ke memori.
- CloseHandle: Menutup objek sistem.
- CopyFileA: Menyalin file (digunakan untuk *DLL hijacking*).
- OpenMutexA, CreateMutexA: Sinkronisasi antar-proses dan kontrol satu instansi.
- Fungsi WS2_32 (seperti send, recv, socket, connect, WSACleanup, closesocket) yang digunakan untuk komunikasi jaringan.

- Sleep: Memberhentikan eksekusi sementara.
- CreateProcessA: Menjalankan perintah atau program secara arbitrer.

6. Penilaian Ancaman





Temuan dari analisis kode assembly ini dikonfirmasi oleh laporan VirusTotal untuk:

- Lab01-01.exe (SHA256: 58898bd42c5bb3d39fb1389f0eee5639cd59180e0b330ea0b327bd6fe47)
- Lab01-02.bin (SHA256: c876a332d7dd8d3a331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6)

Tingkat Deteksi Tinggi: Masing-masing terdeteksi sebagai berbahaya oleh lebih dari 50 vendor keamanan.

Label Ancaman: Termasuk Trojan.Ulisje/Aenjaris, TrojanClicker, serta label keluarga seperti ulisje, aenjaris, kkbov, dan click3.

Program ini merupakan ancaman serius karena kapabilitasnya yang multifungsi dan berbahaya:

- **Persistensi Tinggi**: Dengan menimpa Kernel32.dll, malware mencapai persistensi tingkat sistem yang dalam.
- **Penghindaran Deteksi**: Menggunakan obfuscation dan eksekusi dinamis untuk menghindari deteksi oleh antivirus atau analis.
- **Kontrol Jarak Jauh**: Fitur C2 memungkinkan penyerang untuk menjalankan perintah jarak jauh, mengunduh payload tambahan, atau mencuri data.
- **Stealth**: Beroperasi tanpa jendela tampilan (CREATE_NO_WINDOW) dan menggunakan jeda panjang untuk menyembunyikan aktivitas.

7. Kesimpulan

Kode assembly yang dianalisis menggambarkan malware canggih yang memanfaatkan API Windows tingkat rendah untuk mencapai persistensi melalui *DLL hijacking*, obfuscation untuk menghindari deteksi, dan membangun koneksi C2 yang kuat untuk menjalankan perintah dari jarak jauh. Kombinasi teknik ini menjadikan malware ini sebagai ancaman serius pada sistem yang terinfeksi.

8. Langkah-Langkah Mitigasi

Untuk melindungi sistem dari malware seperti ini, langkah-langkah mitigasi berikut direkomendasikan:

1. **Endpoint Detection and Response (EDR)**: Gunakan EDR yang mampu mendeteksi dan memblokir modifikasi file sistem, perilaku proses mencurigakan, dan koneksi jaringan tidak sah.
2. **Whitelisting Aplikasi**: Terapkan kebijakan whitelist yang ketat, seperti AppLocker, untuk mencegah eksekusi aplikasi/DLL yang tidak terpercaya.
3. **Patch dan Update Rutin**: Pastikan sistem operasi, aplikasi, dan perangkat keamanan selalu diperbarui.

4. **Segmentasi Jaringan & Filtering Egress**: Segmentasi jaringan untuk membatasi pergerakan lateral malware, dan blokir koneksi keluar ke IP mencurigakan.
5. **Pelatihan Keamanan Pengguna**: Latih pengguna agar waspada terhadap email phishing, lampiran mencurigakan, dan taktik rekayasa sosial.
6. **Prinsip Least Privilege**: Berikan hak akses minimum yang diperlukan untuk pengguna dan aplikasi.
7. **Backup & Rencana Pemulihan**: Lakukan backup berkala dan simpan secara terisolasi, serta uji rencana pemulihan data.
8. **Pemantauan Perilaku & Deteksi Anomali**: Implementasikan pemantauan proses dan koneksi jaringan yang tidak biasa untuk mendeteksi aktivitas berbahaya.