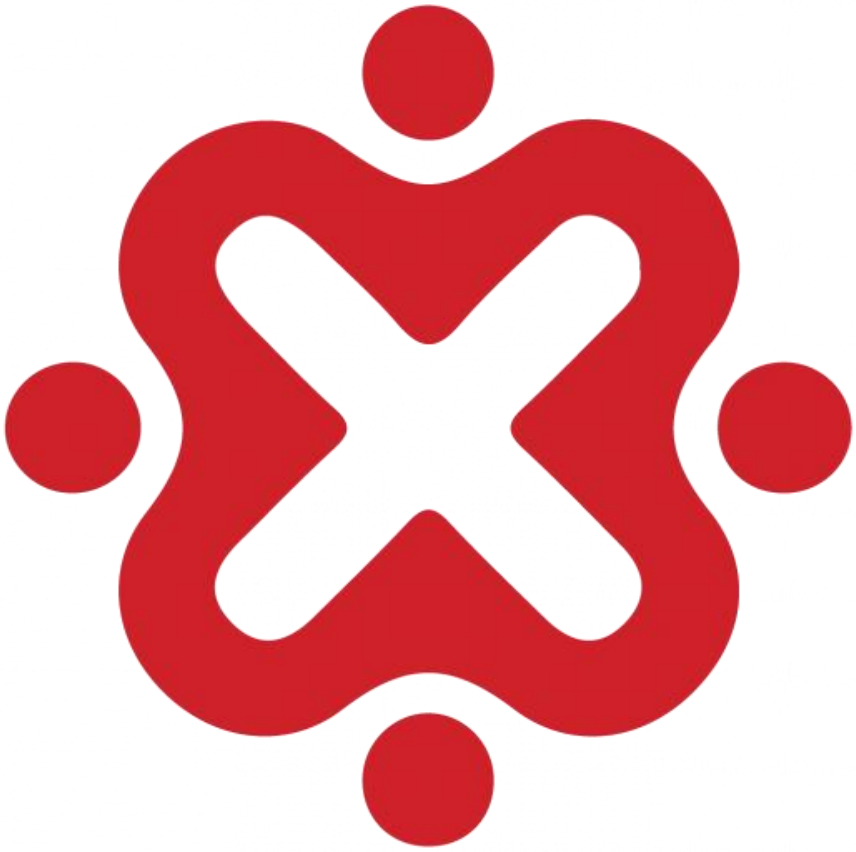
Static Malware Analysis Dan Implementasi Sigma & Yara Rules



ID-Networkers

Indonesian IT Expert Factory

Note:

This Report Is Based On My Experience And Point Of View As A Beginner, Thankyou

Nama Peserta : Raja Ubaid Fawwaz
Username : Raja x NG / NewbieGanas

A. Penjelasan

1. **Malware**
   Malware adalah singkatan dari malicious software, artinya, program yang diciptakan khusus untuk menyusup ke dalam sebuah sistem tanpa sepengetahuan pemiliknya dan bertahan di sana untuk jangka waktu tertentu. Biasanya, malware menyamar sebagai program yang tidak berbahaya untuk mengelabui pengguna.

   Dampak dari perangkat lunak berbahaya ini jauh lebih serius bagi perusahaan dibandingkan pengguna perorangan. Serangan malware pada sistem jaringan perusahaan bisa menyebabkan kerusakan dan gangguan yang luas, membutuhkan upaya pemulihan yang besar di seluruh organisasi.

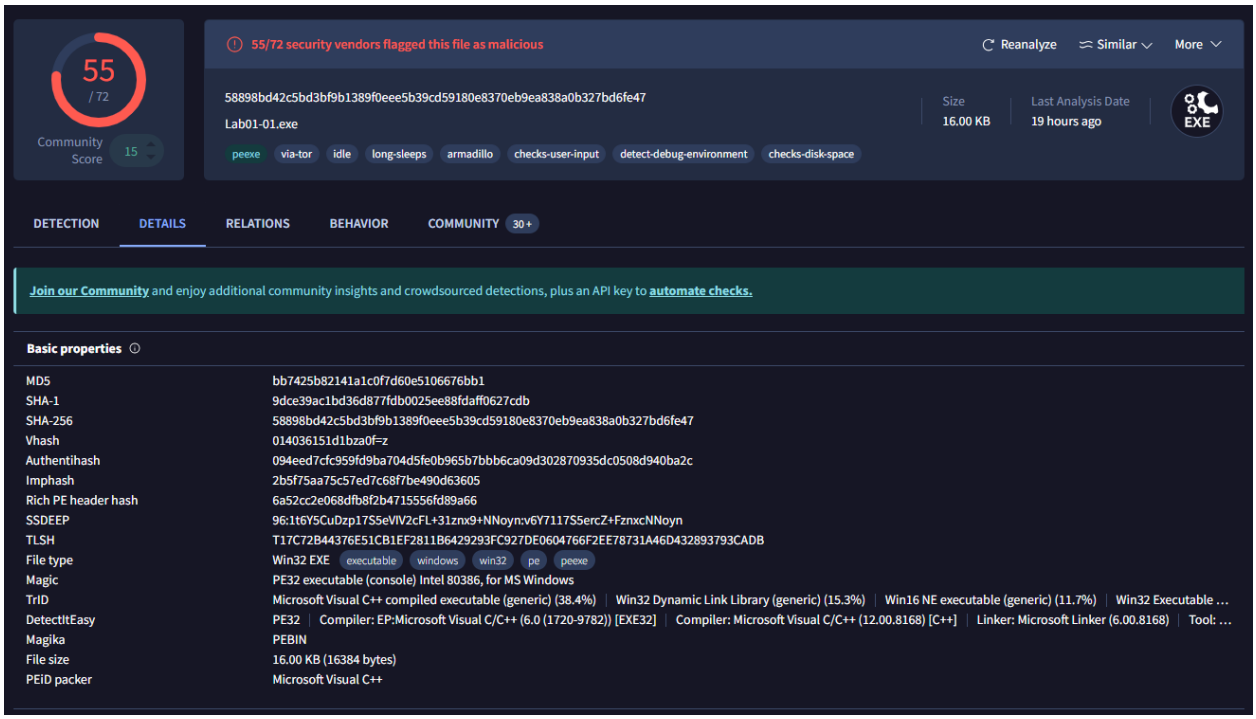
2. **Sigma Rules**
   Sigma adalah Generic Signature Format for SIEM Systems yang berarti sebuah format khusus yang bisa digunakan atau dikonversikan oleh banyak SIEM yang ada secara generic. Tujuannya adalah agar sebuah aturan deteksi cukup dibuat sekali dalam format Sigma, lalu bisa secara otomatis diterjemahkan ke dalam berbagai bahasa query yang spesifik untuk setiap platform SIEM (Security Information and Event Management).

3. **Yara Rules**
   YARA merupakan singkatan dari Yet Another Recursive Acronym atau Yet Another Ridiculous Acronym berupa tools yang bersifat Open Source yang dibuat oleh Victor Alvarez dari VirusTotal. YARA digunakan untuk mengidentifikasi dan mengklasifikasikan malware melalui penggunaan aturan berbasis signature dan karakteristik target lainnya yang dapat dijalankan pengguna terhadap file dan YARA rules yang lebih maju, seperti mencari data di alamat virtual memory tertentu dalam proses yang sedang berjalan.

B. Implementasi

1. Malware Analysis Result

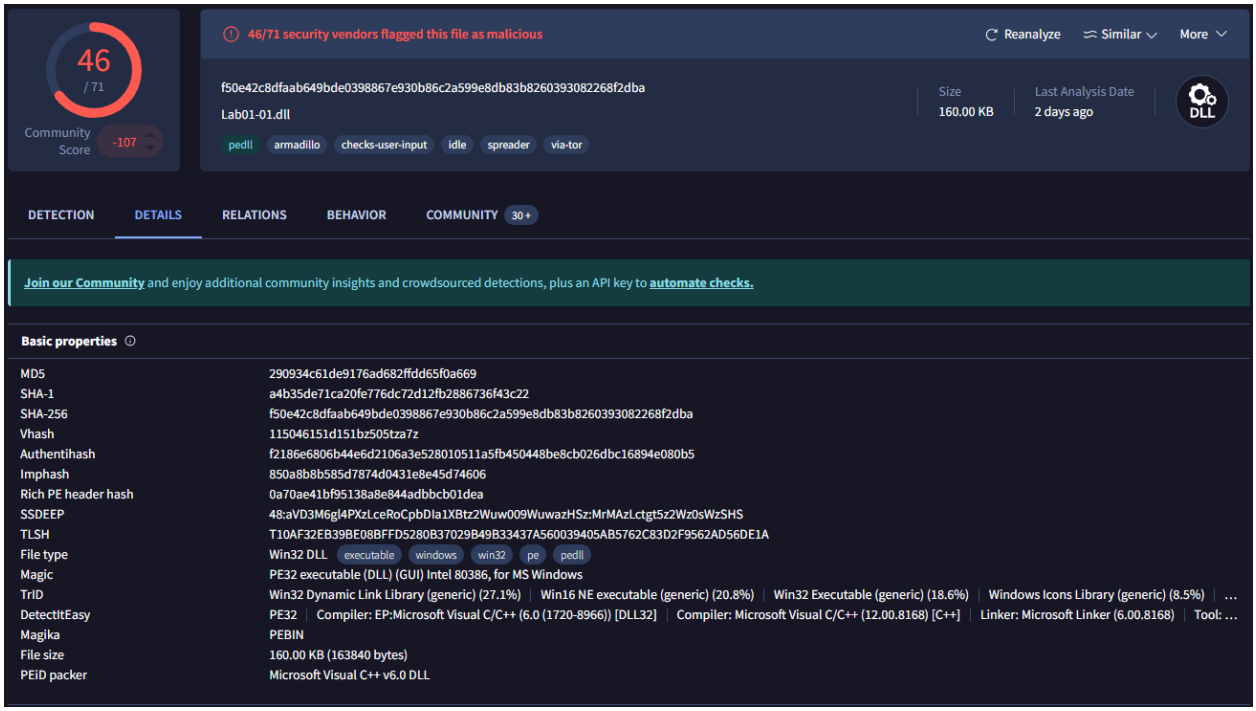
- Lab01-01.exe



Merupakan Malware yang menjadi eksekutor file .dll dengan nama yang sama (Lab01-01.dll). dilihat dari fungsi import pada KERNEL32.dll seperti **FindNextFile, CopyFile, FindFirstFile**. Malware ini bisa mencari file yang menjadi pasangannya dan mengeksekusinya

Malware Ini Dicompile Pada 2010-12-19 16:16:19 UTC

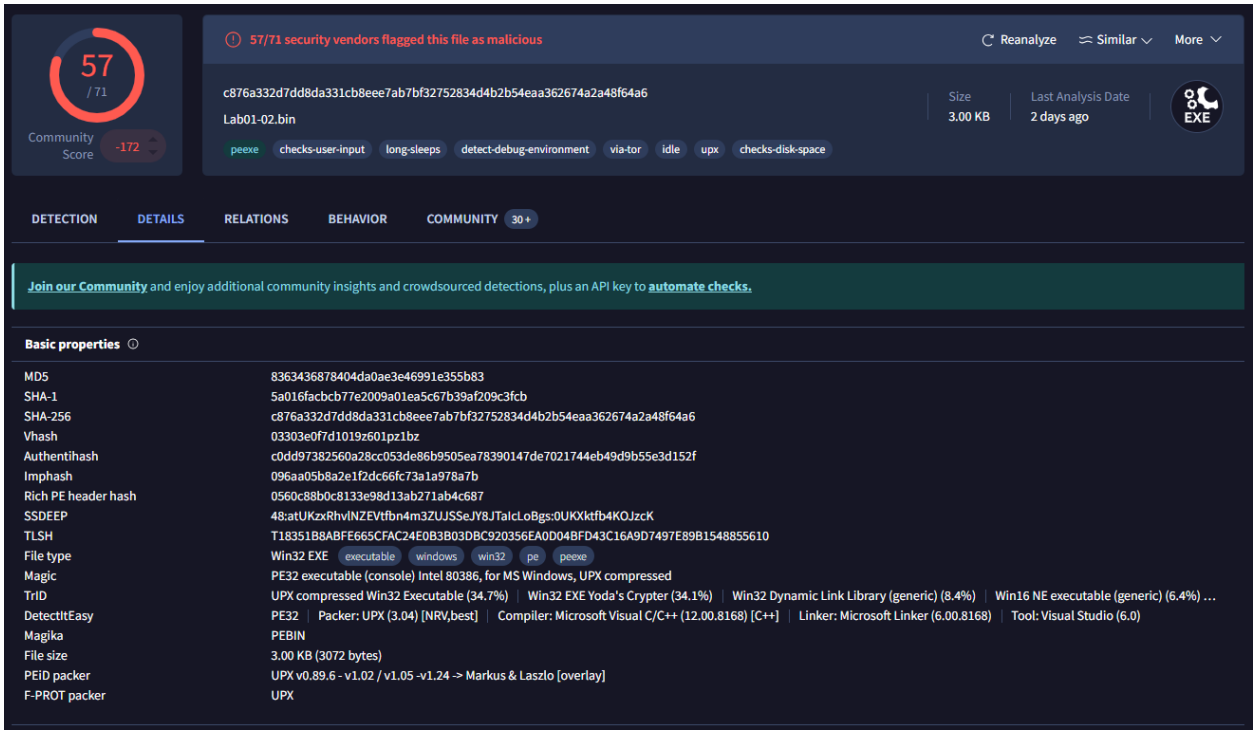
- Lab01-01.dll



Merupakan pasangan dari malware Lab01-01.exe tadi. Karena file .dll tidak bisa berjalan sendiri. Dilihat dari fungsi import pada KERNEL32.dll yaitu **CreateMutex, CreateProcessA, Sleep**. Dan WS2_32.dll yaitu **socket, closesocket, inet_addr dan connect** dapat diasumsikan bahwa pasangan malware ini bisa berkomunikasi dengan server eksternal untuk melakukan Command and Control

Malware Ini Dicompile Pada : 2010-12-19 16:16:38 UTC
Pasangan Malware Ini Menargetkan Sistem Windows, namun dilihat dari waktu compilenya sepertinya windows versi lama. Namun tidak menutup kemungkinan bisa menargetkan versi terbaru

- Lab01-02.bin



Malware ini dipacked dengan upx, setelah di-unpacked dengan command upx di cmd windows fungsi import melalui beberapa file .dll dan strings yang terbaca di file ini terlihat seperti mencari sebuah service dangan nama MalService dan malware ini kuat dugaannya memiliki hubungan dengan web http://www.malwareanalysisbook(.)com. Malware ini juga bisa berkomunikasi melalui internet dibuktikan dengan fungsi import pada WININET.dll yaitu **InternetOpenUrl dan InternetOpen**

Malware Ini Dicompile Pada 2011-01-19 16:10:41 UTC
Dan masih menargetkan sistem windows

2. Sigma Rules Creation

```
Sigconverter
title: Malware Detection
id: 5fe6a618-c510-40bb-b94b-47db34f4e026
status: experimental
description: Detects if a three specific file has been opened or running in the windows system
references:
    - https://medium.com/@Architekt.exe/writing-your-first-sigma-rule-5ed783c87570
    - Noctra Lupra Community
author: Raja x NewbieGanas
date: 2025-07-24
logsource:
    category: file_event
    product: windows
Look Up
detection:
    selection:
        Hashes|contains:
        - 'sha256=58898bd42c5bd3bf9b1389f0eee5b39cd59180e8370eb9ea838a0b327bd6fe47' # Lab01-01.exe
        - 'sha256=f50e42c8dfaab649bde0398867e930b86c2a599e8db83b8260393082268f2dba' # Lab01-01.dll
        - 'sha256=c876a332d7dd8da331cb8eee7ab7bf32752834d4b2b54eaa362674a2a48f64a6' # Lab01-02.bin
    condition: selection
falsepositives:
    - hash value of the file is different either manipulated or the has been rebuild with some manipulation
level: high
```

Sigma rules yang akan saya buat akan mendeteksi kalau file dengan value hash sesuai yang ada di rules dibuka/berjalan. Saya membuatnya seperti ini untuk meminimalisir terjadinya false positive dan langsung spesifik mengarah ke file tertentu saja

3. Yara Rules Creation

Berikut rules yang saya buat

```
// Combo Lab01-01

rule Malware_TwoStage_Backdoor_Lab01_01
{
    meta:
        description = "Detect two specific file which identified as a malware"
        author      = "Raja x NewbieGanas"
        date        = "2025-07-24"
        reference   = "Noctra Lupra Community, Gemini AI"

    strings:
        // String .exe
        $loader_mutex = "WARNING_THIS_WILL_DESTROY_YOUR_MACHINE"
        $loader_dll_typo = "kerne132.dll" ascii

        // String .dll
        $payload_mutex = "SADFHUHF" ascii
        $payload_ip = "127.26.152.13" ascii

    condition:
        uint16(0) == 0x5a4d and
        (
            ($loader_mutex and $loader_dll_typo)
            or
            ($payload_mutex and $payload_ip)
        )
}
```

Konsepnya sederhana, rulesnya akan mendeteksi file yang didalamnya terdapat strings khusus yang diduga mengakibatkan eksploitasi. Namun tidak semua saya input karena akan memicu banyak false positive. Sedangkan bagian condition akan terpenuhi jika file yang dideteksi merupakan file windows dan strings yang dideteksi sama dengan file yang discan, scopenya dipersempit lagi dengan mendeteksi IP spesifik yang menjadi dugaan Server C2 pada malware ini

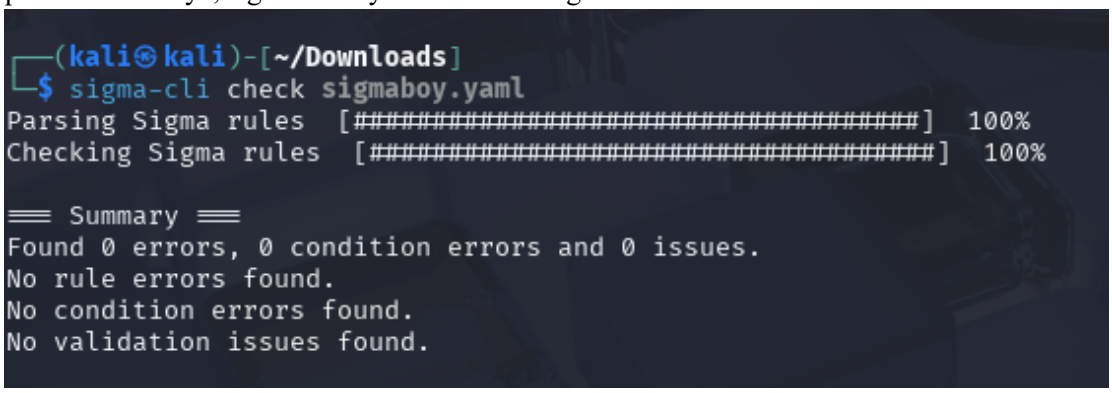
```
// Lab01-02

rule Malware_Packed_Service_Lab01_02
{
    meta:
        description = "Detect a suspicious file which is identified as a malware"
        author      = "Raja x NewbieGanas"
        date        = "2025-07-24"
        reference   = "Noctra Lupra Community, Gemini AI"

    strings:
        // Indikator Packer UPX
        $upx1 = "UPX0" ascii
        $upx2 = "UPX!" ascii

        // Indikator Behavior
        $behavior_service = "MalService" ascii
        $behavior_mutex = "sHGL345" ascii
        $behavior_url = "wareanalysisbook.com" ascii
        $behavior_useragent = "Int6net Explo!r 8FEI" ascii

    condition:
        uint16(0) == 0x5a4d and
        all of ($upx*) and
        2 of ($behavior_*)
}
```

Konsepnya sama seperti sebelumnya perbedaannya ada penambahan deteksi strings UPX karena malware ini dipacked menggunakan upx

Penulisan 2 rules diatas bisa langsung digabung menjadi 1 dokumen

4. Validating The Rules

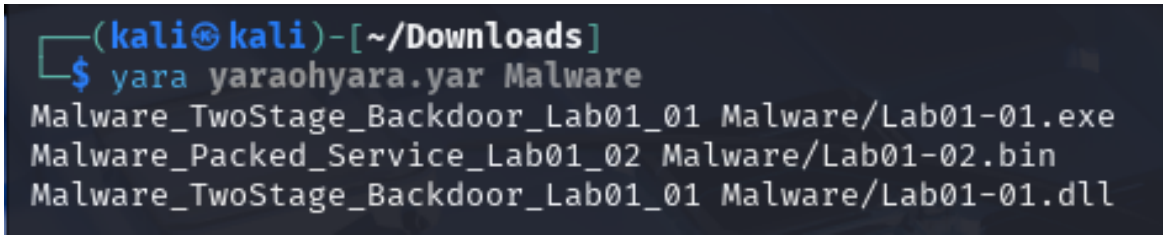
Untuk sigma rules, saya menggunakan sigma-cli untuk mendeteksi apakah ada kesalahan dalam penulisan rulenya, sigma-cli saya sertakan di bagian referensi



Dan untuk yara rules, kali linux saya sudah terdapat command yara yang bisa digunakan, saya menaruh 3 file malware diatas di dalam satu folder dengan susunan seperti ini

- /Downloads
- Yaraohyara.yaml (yara rules yang dibuat)
  - /Malware
    - Lab01-01.exe
    - Lab01-01.dll
    - Lab01-02.bin

Dan saya menginput command yang hasilnya menunjukan kalau 3 filenya sudah terdeteksi



5. **Reference**

https://www.jaiminton.com/Tutorials/PracticalMalwareAnalysis/Chapter1/
https://medium.com/@Architekt.exe/writing-your-first-sigma-rule-5ed783c87570
https://github.com/SigmaHQ/sigma-cli