Analisis Karakteristik Antivirus Berdasarkan Metrik Sumber Daya Komputasi dan Indikator Pendeteksian

1st Ramiz Qudamah
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
ramiz@student.telkomuniversity.ac.id

2nd Adityas Widjajarto
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
adtwjrt@telkomuniversity.ac.id

3rd Ahmad Almaarif
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
ahmadalmaarif@telkomuniversity.ac.id

***Abstrak*— *Malware*, kependekan dari *"Malicious Software"*, merupakan sebuah program yang dirancang khusus untuk melakukan sebuah aktivitas yang dapat membahayakan perangkat lunak pada perangkat korban. Contoh *malware* yang umum ditemukan seperti *trojan, ransomware dan downloader*. Penting bagi pengguna komputer untuk mengenali dan menghindari *malware* ketika sedang menggunakan perangkat komputer. oleh karena itu, pengguna komputer dapat mengatasi serangan *malware* menggunakan perangkat lunak proteksi yang dikhususkan untuk perangkat komputer menggunakan perangkat lunak Antivirus yang dirancang untuk mencegah, mencari, mendeteksi, dan menghapus jenis *malware*yang telah disebutkan sebelumnya. Pada penelitian ini, metode analisis statis hanya digunakan sebagai pendukung untuk mengetahui karakteristik dari antivirus. Hasil dari metrik analisis statis digunakan untuk mengetahui karakteristikantivirus pada fitur antivirus. Hasil dari penelitian ini menunjukan bahwa pada fitur *real-time scanning*, antivirus McAfee relatif lebih efisien dibandingkan dengan antivirus lain karena memiliki rata-rata penggunaan *CPU, memory* yang paling rendah diantara antivirus lain, tingkat deteksi ryang cukup tinggi yaitu sekitar 83,33%, dan rata-rata waktu *scan* yang cukup cepat sekitar 9 detik. Selain itu, Antivirus McAfee relatif paling ampuh dalam mendeteksi sampel *malware* dengantingkat deteksi 100%. Sedangkan pada antivirus Windows Defender relatif paling lemah dari segi tingkat deteksi karena memiliki tingkat deteksi yang paling rendah dan penggunaan sumber daya yang tinggi.**

***Kata kunci*— Analisis Statis *Malware*, Antivirus, Karakteristik Antivirus, *Malware*, *Real-time Protection*, Tingkat Deteksi Antivirus**

I. PENDAHULUAN

Teknologi *malware* adalah perangkat lunak yang dirancang untuk menyebabkan kerusakan, mencuri informasi, atau mengambil alih kontrol komputer tanpa persetujuan pengguna. Beberapa jenis *malware* yang umum termasuk trojan, ransomware, dan downloader. Antivirus adalah perangkat lunak yang digunakan untuk mendeteksi, mencegah, dan menghapus virus komputer. [1]. Potensi antivirus dalam menangani jenis malware seperti *trojan, ransomware,* dan *downloader* dapat dilihat dari cara menangani konsep *CIA (Confidentiality, Integrity,* dan *Availability).* Konsep keamanan *confidentiality* yang diterapkan pada antivirus dapat mendeteksi dan menghapus *trojan, ransomware* dan *downloader* yang digunakan untuk mengakses data pribadi yang dienkripsi, sehingga membantu melindungi kerahasiaan Informasi [2]. Dengan demikian, dapat dikatakan bahwa *malware (Malicious software)* merupakan program yang dirancang untuk disusupkan ke dalam sebuah sistem dengan tujuan untuk melakukan beraneka ragam aktivitas yang bersifat mengganggu kinerja atau dapat juga membahayakan sistem pada komputer.

Pada penelitian ini, metode analisis statis digunakan untuk mengetahui aktivitas *malware* dengan cara mengambil informasi pada indikator pendeteksian seperti string *blacklist*, section, libraries, file ratio, dan function. Metode ini biasanya digunakan untuk mengidentifikasi fungsi *malware* melalui kode programnya [3]. Tindakan preventif juga berperan penting dalam mengurangi aktivitas serangan *malware* yang menginfeksi sistem komputer, salah satunya adalah dengan memanfaatkan program antivirus pada perangkat komputer. Antivirus juga dapat melakukan deteksi *signature based* dan *heuristic based* pada sistem menggunakan fitur *scanning*yang terdapat pada *software* itu sendiri.

Perangkat lunak antivirus adalah program yang dirancang untuk mencegah, mendeteksi, dan menghapus infeksi *malware* pada perangkat komputasi individu [4]. Antivirus disebut juga sebagai perangkat lunak perlindungan dari serangan *malware*, program ini dapat melindungi perangkat menggunakan beberapa fitur yang dimilikinya, salah satunya fitur *real-time protection*. *Real-time protection* adalah fitur yang terdapat pada beberapa antivirus yang memungkinkan pengguna untuk memindai perangkat penyimpanan sekunder, seperti *flash drive, hard drive* eksternal, atau kartu *memory*, untuk mencari tahu apakah terdapat *malware* yang menyebar melalui perangkat tersebut sekaligus membersihkan *malware* yang bersarang di dalamnya.

Pada penelitian ini, metode analisis statis digunakan untuk mengevaluasi perilaku *malware* dengan cara menjalankannya dan memonitor aktivitas yang terjadi. Metode ini biasanya digunakan untuk mengidentifikasi tindakan yang dilakukan oleh *malware* ketika dijalankan. Dengan menggunakan analisis statis, kode yang disembunyikan oleh *malware* untuk

melancarkan aktivitasnya akan terlihat selama proses monitoring. Jumlah aktivitas tersebut akan digunakan untuk mengetahui karakteristik antivirus pada fitur *real-time protection.* Ketika menjalankan fitur yang dapat mencegah malware secara terus menerus pada antivirus, performa antivirus sangat penting untuk diperhatikan. Hal ini dikarenakan jika antivirus memiliki performa yang buruk, maka proses pemindaian akan berlangsung lama dan menyebabkan kinerja sistem komputer menjadi terhambat. Selain itu, tingkat deteksi antivirus juga menjadi hal utama dalam melindungi perangkat dari serangan *malware*.

II. KAJIAN TEORI

A. Malware *(Malicious software)*

Malicious software atau yang biasa dikenal dengan *malware* merupakan sebuah perangkat lunak yang terpasang pada suatu system komputer tanpa sepengetahuan oleh user atau pemilik sistem tersebut. Sesuai dengan namanya, *malware* dapat melakukan *malicious action* atau tindakan jahat seperti mencuri informasi rahasia, perusakan pada suatu sistem, mendapatkan hak akses suatu komputer dan menjalankan program yang ada pada komputer tersebut. [5]. Setiap perangkat lunak yang melakukan sesuatu yang dapat menyebabkan kerugian pada user, komputer ataupun jaringan dapat dianggap sebagai *malware* [6].

B. Analisis Statis

Analisis statis merupakan salah satu metode pada analisis *malware* yang dilakukan pada saat sebelum *malware*dieksekusi atau dijalankan. sehingga informasi yang didapatkan sangatlah lengkap dan bisa memberikan gambaran yang sangat detail tentang mekanisme kerja *malware* tersebut secara keseluruhan. Analisis statis adalah metode yang digunakan untuk menganalisis *malware* tanpa menjalankannya. Ini dapat dilakukan dengan mengekstrak kode sumber dari *malware* dan menganalisisnya secara manual atau menggunakan alat otomatis [7].

C. *String*

Strings adalah sekumpulan karakter yang digunakan dalam kode sumber suatu aplikasi atau program, termasuk dalam *malware*. Penggunaan string dalam *malware* dapat digunakan untuk menyembunyikan aktivitas *malware*, menentukan sasaran, atau mengeksekusi perintah [8].

Strings blacklist adalah metode deteksi *malware* yang mencari string atau kata kunci yang spesifik dalam kode atau file yang diperiksa, dan jika string tersebut ditemukan, maka file tersebut dianggap sebagai *malware* [9].

D. *Section*

Dynamic link libraries (*DLLs*) adalah *file* yang terdapat Section adalah bagian dari file executable yang digunakan untuk menyimpan informasi yang digunakan oleh sistem operasi dan loader. Setiap file executable memiliki beberapa seksi yang digunakan untuk menyimpan data seperti kode program, data statis, dan data statis [10].

Malware dapat menggunakan section yang ada dalam file executable untuk menyimpan kode yang digunakan untuk mengeksekusi aktivitas jahat [11].

E. *File Ratio*

File ratio adalah metode analisis yang digunakan untuk menentukan kemiripan antara dua atau lebih file yang berbeda. Ini bisa digunakan dalam analisis *malware* untuk menentukan apakah dua file yang berbeda merupakan variandari *malware* yang sama atau tidak. Metode ini biasanya digunakan untuk menentukan apakah file yang diduga sebagai *malware* adalah varian dari *malware* yang sudah dikenal atau tidak. [12].

Metode ini berdasarkan pada perbandingan byte-per-byte dari dua file yang dibandingkan. Jika dua file memiliki tingkat persamaan yang tinggi, maka file tersebut dianggap sebagai varian dari *malware* yang sama [11]

F. *Function*

Malware adalah perangkat lunak yang dirancang untuk menyebabkan kerusakan atau mengambil data tanpa persetujuan pengguna. Beberapa jenis *malware* menggunakan fungsi-fungsi tertentu untuk melakukan tugas mereka. Function ini dapat berupa fungsi dari sistem operasiatau fungsi yang ditulis oleh pengembang aplikasi [13].

G. *Libraries*

Libraries pada *malware* adalah kumpulan perintah atau fungsi yang digunakan oleh *malware* untuk menjalankan aktivitas tertentu, seperti mengambil data, menyebar atau menyembunyikan diri. *Library* berguna karena menampilkan *DLL (Dynamic Link Library)* apa yang sedang diimpor oleh *malware* [13].

H. Antivirus *Software*

Antivirus *software* adalah sebuah aplikasi yang digunakan untuk mendeteksi dan menghapus *malware* yang ada di dalam sistem computer [11]. Antivirus *software* dapat digunakan untuk melakukan *scan* terhadap *file-file* yang terdapat di dalam sistem komputer dengan tujuan untuk mencari tahu apakah ada *file* yang mengandung *malware* atautidak [14].

I. Fitur *Real-time Protection*

Fitur *Real-time Protection* pada antivirus merupakan fasilitas yang memungkinkan *software* antivirus untuk melakukan pemindai secara otomatis dan terus-menerus pada sistem komputer. Hal ini dilakukan untuk mengidentifikasi dan menghapus *malware* yang berpotensi mengancam keamanan sistem. [11].

III. METODE

Pada tahapan proses pengumpulan data yang mendukung penelitian ini adalah sebagai berikut:

A. Metode Kepustakaan

Metode keupstakaan adalah metode dalam mencari, mengumpulkan, serta menganalisis sumber data untukdiolah. Sumber data bisa melalui buku, jurnal, *e-book*, dan modul yang berhubungan dengan penelitian ini [15].

B. Sistematika Penelitian

Sistematika penelitian merupakan bagan yang menjelaskan tahapan yang harus dilakukan untuk menyelesaikan penelitian, tahapan yang dilakukan dalam penelitian ini sesuai dengan metode yang telah ditentukan

sebelumnya yaitu metode analisis berdasarkan eksperimen [16].

Tahapan-tahapan pada sistematika penelitian yang dilakukan dapat dilihat pada gambar berikut:

GAMBAR 1
(Alur Tahapan Penelitian)

C. Pengumpulan Data

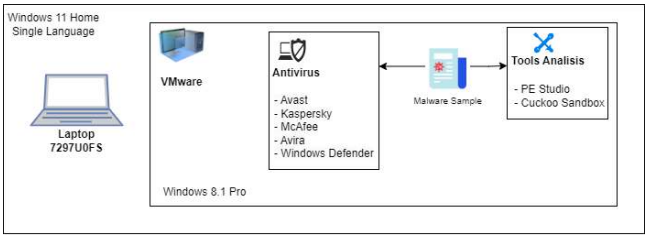
Dalam penelitian ini pengumpulan data dilakukan dengan menjalankan atau mengeksekusi *malware* pada sistem, setelahnya dapat dilakukan *monitoring* terhadap sistem yang telah di injeksi *malware* menggunakan metode analisis statis berdasarkan hasil decompiler *malware* yang terdeteksi menggunakan *tools* khusus analisis statis, dimana *tools* yang akan digunakan seperti PE Studio, Cuucko Sandbox berdasarkan *Strings*, String *blacklist*, libraries, folder function, file ratio dan section yang kemudian antivirus akan menjalankan metode pemindaian pada fitur *real-time scan* untuk mengetahui karakteristik antivirus.

D. Pengolahan Data

Berdasarkan temuan pada hasil deteksi *malware* menggunakan metode analisis statis dan hasil deteksi pemindaian oleh perangkat lunak antivirus dilakukan analisis dan diproses untuk dapat mengetahui karakteristik antivirus. Selanjutnya hasil deteksi dari kedua proses tersebut dapat diperoleh data kuantitatif berupa penggunaan sumber daya, waktu *scanning* dan tingkat deteksi terhadap *malware* berdasarkan matriks *malware* menggunakan metode analisis statis.

E. Desain Lingkungan Virtual

Gambar di bawah ini menunjukkan desain lingkungan *virtual machine* yang terdiri dari beberapa komponen utama, yaitu *core operating system*, VMware, sampel *malware*, *tools* analisis statis dan *software* antivirus yang dipisahkan menggunakan snapshot. VMware merupakan lingkungan yang terisolasi yang dapat menjalankan sistem operasi dan aplikasi seperti sebuah komputer fisik. Sistem operasi *core* merupakan sistem operasi utama dimana VMware di pasang,sedangkan sistem operasi *guest* merupakan sistem operasi yang di instal di dalam VMware, *tools* analisis statis dan antivirus merupakan aplikasi yang di install di dalam *guest OS,* dan *snapshot* merupakan *chekpoint* dimana keadaan *guest OS* disimpan.



GAMBAR 2
(Desain Lingkungan Virtual)

F. Sampel *Malware*

Gambar Sampel *malware* yang digunakan pada penelitian ini merupakan *file* yang memiliki format *exe*. Sampel *malware* tersebut akan dijalankan pada *virtual machine* Windows 8.1 dan digunakan untuk mengetahui kemampuan antivirus dalam menangani infeksi *malware*. Berikut daftar sampel *malware* yang digunakan pada penelitian ini:

TABEL 1
(Sampel *Malware* Yang Diujikan)

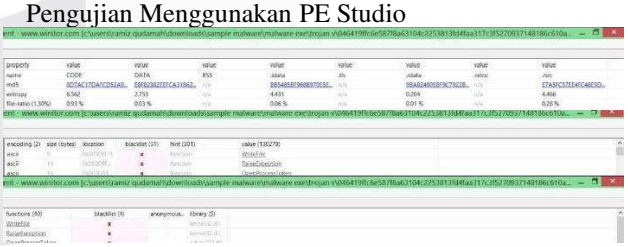
Nama *File/Hash*	**Ekstensi *File***	**Ukuran *File***	**Nama Sampel *Malware***
046419ffc6e587f8a631	*exe*	4,081,090 *bytes*	Sampel 1 *Trojan*
97437cdf7f697beb41d5	*exe*	2,261,024 *bytes*	Sampel 2 *Trojan*
348902db5e72113a54b	*exe*	90,112 *bytes*	Sampel 3 *Ransomware*
21957bfa277e386c9967	*exe*	71,168 *bytes*	Sampel 4 *Ransomware*
db825e1f70c6f9b265be	*exe*	4,096 *bytes*	Sampel 5 *Downloader*
a1b5a5cd5410656eb13	*exe*	52,224 *bytes*	Sampel 6 *Downloader*

IV. HASIL DAN PEMBAHASAN

A. Gambar

Bagian analisis sampel pada *malware* ini merupakan pengujian yang dilakukan oleh peneliti untuk mendapatkan hasil temuan atau aktivitas pada sampel *malware*. Berikut merupakan hasil analisis pada temuan aktivitas *malware:*

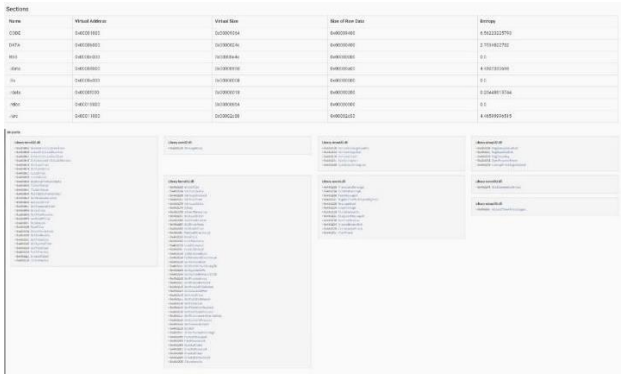
1. Pengujian Menggunakan PE Studio



GAMBAR 3
(Pengujian PE Studio Sampel 1)

Gambar 3 merupakan hasil analisis menggunakan PEStudio pada sampel 1 *trojan*. PE Studio menampilkan hasil bahwa terdapat adanya 31 string *blacklist* 40 function, dan 1.30% file ratio yang merupakan perhitungan dari decompilerfile *malware*.

2. Pengujian Menggunakan Cuucko Sandbox



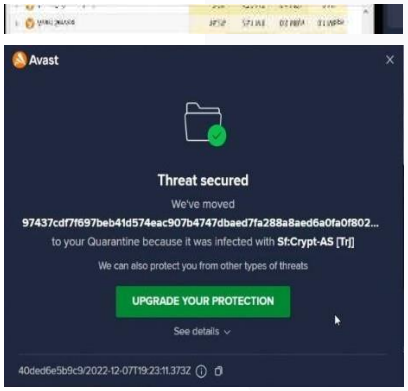
GAMBAR 4
(Pengujian Cuucko Sandbox Sampel 1)

Gambar 4 merupakan hasil analisis menggunakan Cuucko Sandbox pada sampel 1 trojan. Cuucko Sandbox menampilkan hasil bahwa terdapat adanya penambahan 8 section dan 8 libraries yang merupakan perhitungan dari decompiler file *malware*.

3. Pengujian Antivirus

Bagian analisis antivirus ini merupakan pengujian yang dilakukan oleh peneliti untuk mendapatkan hasil berupa *profiling* pada fitur *real-time protection* pada antivirus. Pada penelitian ini, antivirus yang akan diujikan untuk melakukan analisis fitur *real-time protection* adalah Avast, Kaspersky, Avira, McAfee, dan Windows Defender.

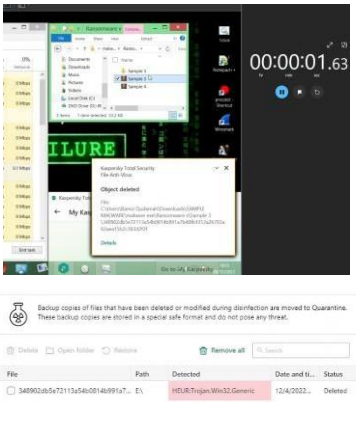
a. Pengujian Antivirus Avast



GAMBAR 6
(Pengujian pada Sampel 2)

Gambar 6 merupakan hasil Pengujian fitur *real-time protection* antivirus Avast pada sampel 2 *trojan* yang sudah disiapkan dan dianalisis sebelumnya. Beberapa indikator yang diujikan dan dipantau pada sampel tersebut merupakansumber daya yang digunakan oleh antivirus ketika melakukan proses *scanning* dan bagaimana antivirus menampilkan hasildeteksi *file* ketika proses *scanning* telah selesai.

b. Pengujian Antivirus Kaspersky



GAMBAR 7
(Pengujian pada Sampel 3)

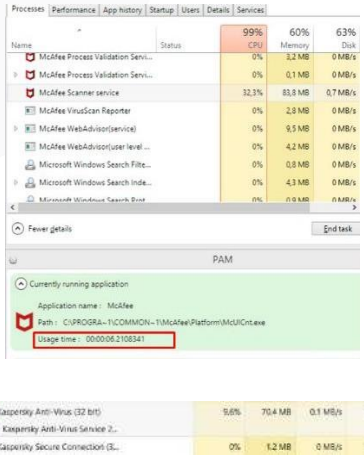
Gambar 7 merupakan hasil Pengujian fitur *real-time protection* antivirus Kaspersky pada sampel 3 *ransomware* yang sudah disiapkan dan dianalisis sebelumnya.

c. Pengujian Antivirus Avira

GAMBAR 8
(Pengujian pada Sampel 4)

Gambar 8 merupakan hasil Pengujian fitur *real-time protection* antivirus Avira pada sampel 4 *ransomware* yang sudah disiapkan dan dianalisis sebelumnya.

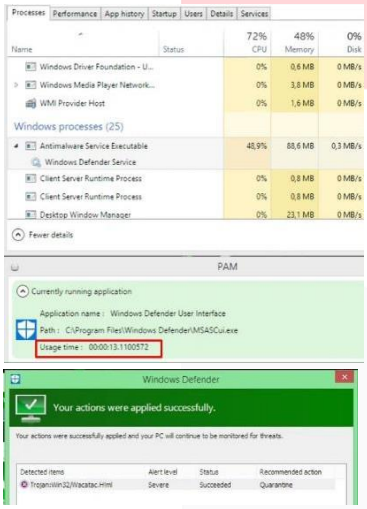
d. Pengujian Antivirus McAfee



GAMBAR 9
(Pengujian pada Sampel 3)

Gambar 9 merupakan hasil Pengujian fitur *real-time protection* antivirus McAfee pada sampel 3 *Ransomware*yang sudah disiapkan dan dianalisis sebelumnya.

e. Pengujian Antivirus Windows Defender

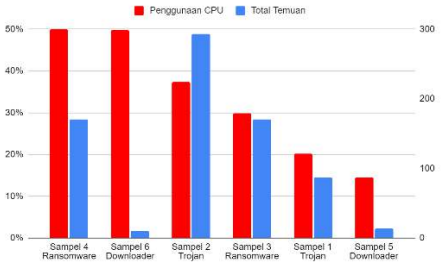


GAMBAR 10
(Pengujian pada Sampel 6)

Gambar 10 merupakan hasil Pengujian fitur *real-time protection* antivirus Windows Defender pada sampel 6 downloader yang sudah disiapkan dan dianalisis sebelumnya.

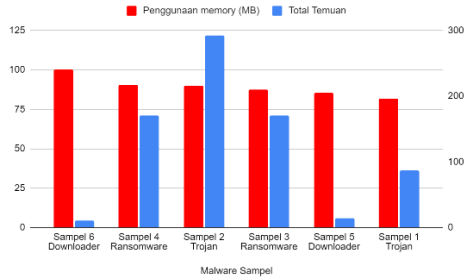
4. Perbandingan Total Aktivitas Malware dengan Metrik Antivirus

Berikut ini merupakan perbandingan hasil analisis dari total aktivitas *malware* dengan metrik antivirus pada hasil pengujian.



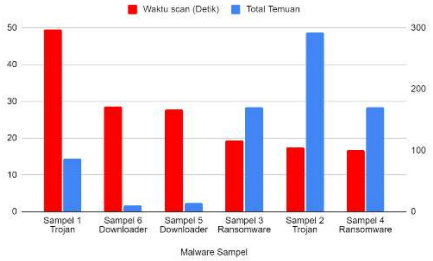
GAMBAR 10
(Total Aktivitas *Malware* dan Penggunaan *CPU* Antivirus)

Gambar 11 merupakan perbandingan total aktivitas *malware* dengan Penggunaan *CPU* Antivirus. Jika diperhatikan, gambar menunjukan terdapat beberapa sampel menunjukan bahwa semakin tinggi jumlah aktivitas *malware*, maka semakin tinggi pula tingkat penggunaan *CPU* yang dibutuhkan, namun terdapat juga anomali yang dikarenakan beberapa fungsi *malware* masih disembunyikan saat di analisis.



GAMBAR 10
(Total Aktivitas *Malware* dan Penggunaan *Memory* Antivirus)

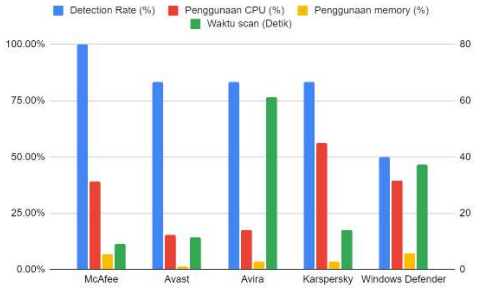
Gambar 12 merupakan perbandingan total aktivitas *malware* dengan Penggunaan *memory* Antivirus. Dari gambar tersebut menunjukan bahwa indikator metrik pada temuan *malware* tidak menunjukan adanya hubungan dikarenakan beberapa fungsi *malware* masih disembunyikan saat di analisis.



GAMBAR 13
(Total Aktivitas *Malware* dan Waktu Scan Antivirus)

Gambar 13 merupakan perbandingan total aktivitas *malware* dengan waktu *scan* antivirus. Dari gambar tersebut menunjukan bahwa indikator metrik pada temuan *malware* tidak menunjukan adanya hubungan dikarenakan beberapa fungsi *malware* masih disembunyikan saat di analisis.

5. Perbandingan *Software* Antivirus



GAMBAR 13
(Perbandingan Antivirus)

Pada perbandingan antivirus dalam mendeteksi *malware* pada fitur *real-time protection*, antivirus Avast merupakan antivirus yang relatif lebih efisien dibandingkan dengan antivirus lain yang diuji karena memiliki rata-rata penggunaan *CPU* dan *memory* yang rendah, tingkat deteksi yang cukup tinggi, dan waktu *scan* yang cepat. Antivirus McAfee merupakan antivirus yang relatif paling ampuh

dalam mendeteksi sampel *malware* karena memiliki tingkat deteksi 100%. Sedangkan pada antivirus Windows Defender merupakan antivirus yang relatif paling lemah dari segi tingkat deteksi karena memiliki tingkat deteksi yang paling rendah dibandingkan dengan antivirus lain.

V. KESIMPULAN

Setelah dilakukan pengujian dan analisis karakteristik antivirus dengan menggunakan indikator pendeteksian analisis statis, dapat disimpulkan bahwa kebanyakan tidak adanya hubungan dari indikator pendeteksian statis pada karakteristik antivirus seperti *CPU, memory, disk,* dan waktu *scan*. Selain itu, pada fitur *real-time protection,* antivirus Avast relatif lebih efisien dibandingkan dengan antivirus lain karena memiliki rata-rata penggunaan *CPU* dan *memory* yang rendah, tingkat deteksi yang cukup tinggi, dan waktu *scan* yang cepat. Antivirus McAfee paling ampuh dalam mendeteksi sampel *malware* dengan tingkat deteksi paling tinggi di antara antivirus lain. Sedangkan pada antivirus Windows Defender relatif paling lemah dari segi tingkat deteksi karena memiliki tingkat deteksi yang paling rendah

REFERENSI

[1] C. H. Malin, E. Casey, J. M. Aquilina, and C. W. Rose, Malware forensic field guide for Windows systems digital forensics field guides. Amsterdam Elsevier Rockland, Mass. Syngress Waltham, Mass. Syngress, 2012.

[2] S. Samonas and D. Coss, "THE CIA STRIKES BACK: REDEFINING CONFIDENTIALITY, INTEGRITY AND AVAILABILITY IN SECURITY," 2014. [Online]. Available: https://www.proso.com/dl/Samonas.pdf

[3] C. C. Elisan, Advanced malware analysis. New York Mcgraw-Hill Education, 2015.

[4] L. Rosencrance, "What is antivirus software (antivirus program)?" *SearchSecurity*, Aug. 2017. https://www.techtarget.com/searchsecurity/definition/antivir us-software (accessed Nov. 09, 2021).

[5] Michael Hale Ligh, Malware analyst's cookbook and DVD : tools and techniques for fighting malicious code. Indianapolis, In: Wiley, 2011.

[6] Sans Institute, "Understanding Malware: Types and Characteristics," *Sans.org*, 2017. https://www.sans.org/security-resources/malware-types-characteristics (accessed Dec. 20, 2022).

[7] T. A. Cahyanto, V. Wahanggara, and D. Ramadana, "Analisis dan Deteksi Malware Menggunakan Metode Malware Analisis Dinamis dan Malware Analisis Statis," JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia), vol. 2, no. 1, 2017, doi: 10.32528/justindo.v2i1.1037.

[8]]N. Fox, "PeStudio Overview: Setup, Tutorial and Tips," www.varonis.com, Oct. 06, 2021. https://www.varonis.com/blog/pestudio

[9] priyampatel9911, "What is PeStudio?," GeeksforGeeks, Sep. 14, 2022. https://www.geeksforgeeks.org/what-is-pestudio/ (accessed Jan. 20, 2023).

[10] Karl-Bridge-Microsoft, "PE Format - Win32 apps," learn.microsoft.com, Jun. 24, 2022. https://learn.microsoft.com/en-us/windows/win32/debug/pe-format

[11] E. Skoudis and L. Zeltser, *Malware fighting malicious code*. Upper Saddle River, Nj Prentice Hall Ptr, 2008.

[12] Microsoft, "Walkthrough: Calling Windows APIs - Visual Basic," *learn.microsoft.com*, Nov. 03, 2022. https://learn.microsoft.com/en-

[13] N. Fox, "PeStudio Overview: Setup, Tutorial and Tips," *www.varonis.com*, Oct. 16, 2021. https://www.varonis.com/blog/pestudio#:~:text=PeStudio%20is%20a%20tool%20used (accessed Nov. 19, 2022).

[14] Techslang, "What is Antivirus Software? — Definition by Techslang," *Techslang — Tech Explained in Simple Terms*, Apr. 04, 2019. https://www.techslang.com/definition/what-is-antivirus-software/ (accessed Dec. 21, 2022).

[15] J. Danandjaja, "Metode Penelitian Kepustakaan," Antropol. Indones., 2014, doi: 10.7454/ai.v0i52.3318.

[16] A. A. Pradipta, Y. A. Prasetyo, and N. Ambarsari, "Pengembangan Web E-Commerce Bojana SariMenggunakan Metode Prototype," *eProceedings of Engineering*, vol. 2, no. 1, Apr. 2015, Accessed: Dec. 21, 2022. [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/2726