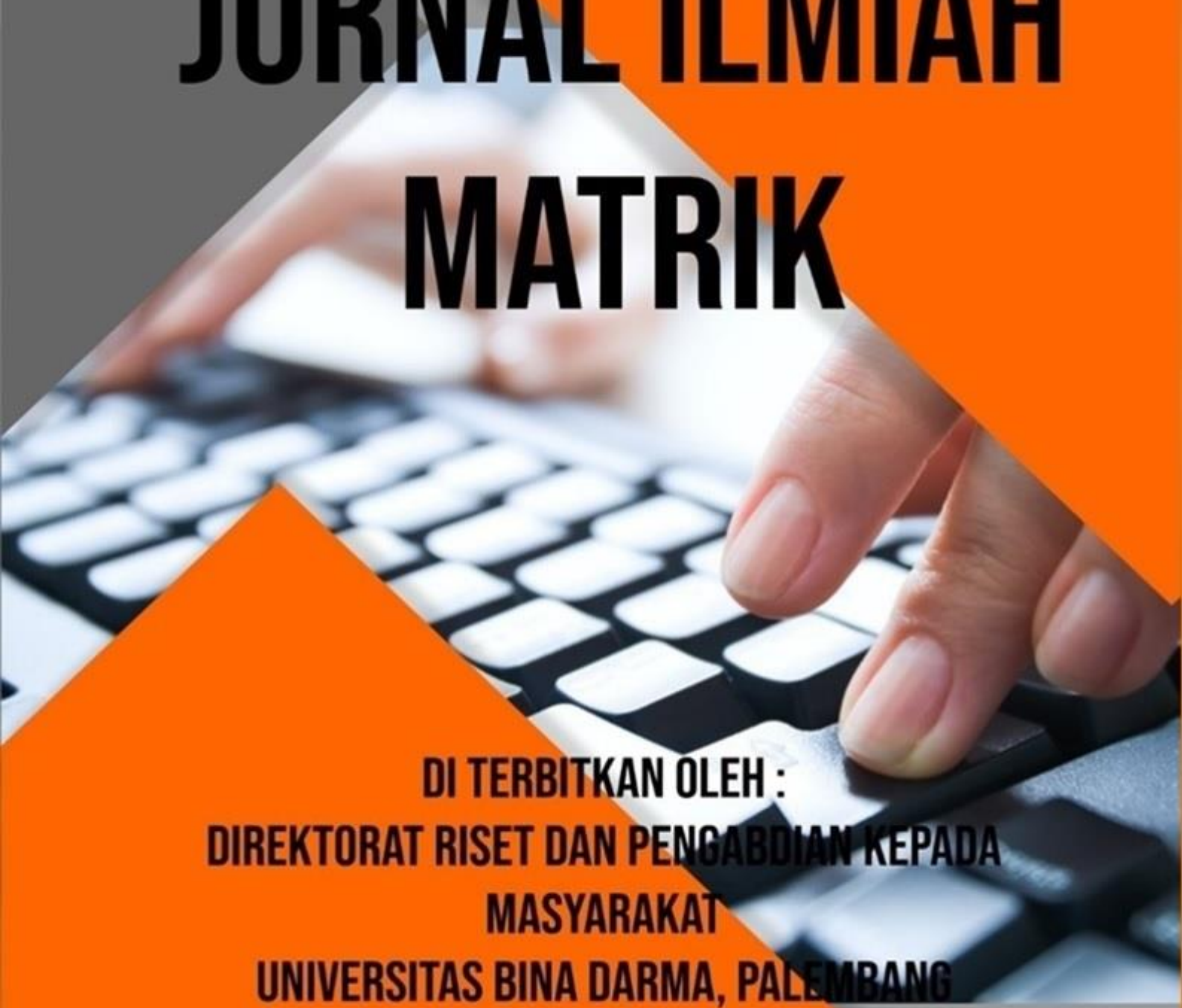
JURNAL ILMIAH MATRIK



DI TERBITKAN OLEH :
DIREKTORAT RISET DAN PENGABDIAN KEPADA MASYARAKAT
UNIVERSITAS BINA DARMA, PALEMBANG
JENDR. A.YANI NO.03
http://journal.binadarma.ac.id
email :jurnalmatrik@binadarma.ac.id | Hp. 081532791703

VOL. 23 NO. 1 APRIL 2021 P-ISSN : 1411-1624 E-ISSN : 2621 -8089 HAL : 1 - 132

Vol. 23 No. 1, April 2021 **ISSN : 1411-1624**
E-ISSN : 2621-8089

Jurnal Ilmiah MATRIK

Jurnal Ilmiah MATRIK diterbitkan oleh Direktorat Riset dan Pengabdian Kepada Masyarakat (DRPM) Universitas Bina Darma Bekerja sama dengan Fakultas Ilmu Komputer dan Jurnal Ilmiah Terpadu Universitas Bina Darma (JIT-UBD) Publikasi dilakukan secara berkala setiap tahun 3 (tiga) kali (April, Agustus dan Desember). Terbit pertama kali April 1999.

Alamat Redaksi:
Kampus Utama Lantai Universitas Bina Darma (UBD)
Jalan Ahmad Yani No.3 Palembang
Telp.0711-515679, Fax.0711-515582,
Website: http://journal.binadarma.ac.id/index.php/jurnalmatrik
Email: jurnalmatrik@binadarma.ac.id.

Dicetak di Pusat Penerbitan dan Percetakan Universitas Bina Darma Press (PPP-UBD Press).
Isi Diluar Tanggung Jawab Percetakan.

Vol. 23 No.1, April 2021

ISSN : 1411-1624
E-ISSN : 2621-8089

Jurnal Ilmiah MATRIK

DAFTAR ISI

Vol. 23 No.1, April 2021 **ISSN : 1411-1624**
E-ISSN : 2621-8089

Jurnal Ilmiah MATRIK

DAFTAR ISI

Vol. 23 No. 1, April 2021 **ISSN : 1411-1624**
E-ISSN : 2621-8089

Jurnal Ilmiah MATRIK

PENGANTAR REDAKSI

Puji Syukur kehadirat Allah SWT Tuhan Yang Mahas Esa, Jurnal Ilmiah Matrik untuk Edisi Bulan April 2021 Volume 23 Nomor 1 telah terbit sesuai dengan jadwal walaupun terdapat beberapa kendala.

Jurnal Ilmiah Matrik untuk edisi ini telah menerima kiriman artikel dengan jumlah yang cukup banyak, tetapi dalam prosesnya telah dipilih beberapa artikel terbaik sesuai dengan hasil *review*. Untuk mempermudah dan mempercepat dalam proses *review* dan penyuntingan, kami mengharapkan kepada penulis untuk selalu mengikuti *template* dan/atau petunjuk Penulisan. Naskah atau Artikel yang dikirimkan tetapi tidak sesuai dengan template maka akan dikembalikan sebelum masuk dalam proses *review*. Edisi terbitan kali ini memuat 14 Artikel dari penulis yang berasal dari berbagai Perguruan Tinggi, antara lain; **Universitas Mulia Balikpapan, Universitas Pembangunan Nasional Veteran Jakarta, Universitas Katolik Widya Mandira Kupang, Universitas Raharja Tangerang, , STMIK Umel Mandiri Papua, Universitas Baturaja, STMIK Widya Cipta Dharma Samarinda dan Universitas Bina Darma Palembang**

Penghargaan setinggi-tingginya kami berikan kepada Penulis, Mitra Bestari, Tim editor dan semua Pihak yang terlibat dalam penyusunan serta penerbitan Jurnal Ilmiah Matrik untuk Edisi Volume 23 Nomor 1 Bulan April 2021. Dalam upaya perbaikan dan peningkatan kualitas baik dari isi maupun tampilan jurnal, kami mengharapkan saran dan kritik membangun untuk perbaikan Edisi Berikutnya.

Tim Redaksi

ISSN : 1411-1624
e-ISSN: 2621-8089

ANALISIS MALWARE DENGAN METODE SURFACE DAN RUNTIME ANALYSIS

Febriyanti Panjaitan [1], Helda Yudiastuti [2] , Maria ulfa [3]
Dosen Universitas Bina Darma [1,2,3]
Jalan Jenderal Ahmad Yani No.3 Palembang
Sur-el : febriyanti_panjaitan@binadarma.ac.id [1], helda.yudiastuti@binadarma.ac.id[2], Maria.ulfa@binadarma.ac.id[3]

Abstract : *Malware is a virus that has many ways of infecting data and giving dangerous damage, such as what happened to a hospital in Palembang, malware attacks all existing data so that the data cannot be accessed by related parties. Malware not only attacks data, but can damage operating systems that are vulnerable to being infiltrated, one of which is the Windows operating system. Malware is very difficult to identify with the original data file if it does not use the help of analysis tools, with this research will conduct analysis and testing of malware behavior, so that the pattern and type of malware can be identified with 2 analysis methods, namely the surface analysis method (Pestudio, Strings). , Exeinfo and Virus) and runtime analysis (Regshot, CaptureBAT, Noriben Malware Analysis). The analysis carried out will create a work environment as a testing ground, so that it does not interfere with the main system that has been running. With the trials conducted, it was found that the file with the name "games.exe" can be said to be malware because it has strings and can duplicate itself to the system32 folder as evidenced by MD5.*

Keywords: *Malware, Surface Analisis, Runtime Analisis, Windows*

Abstrak : *Malware salah satu virus yang memiliki banyak cara dalam menginfeksi sebuah data dan dapat memberikan kerusakan yang berbahaya, seperti yang terjadi pada salah satu rumah sakit ternama, dimana malware menyerang seluruh data yang ada sehingga data tidak dapat diakses kembali oleh pihak yang terkait. Malware bukan hanya menyerang data, tetapi dapat merusak sistem operasi yang sedang digunakan karena rentan untuk disusupi, salah satunya adalah sistem operasi windows. Malware sangat sulit di intifikasi dengan file data yang asli jika tidak menggunakan bantuan tools analisis, dengan hal tersebut penelitian akan melakukan analisis dan pengujian terhadap tingkah laku malware, sehingga dapat diketahui pola dan jenis malware dengan 2 metode analisis yaitu metode surface analisis (Pestudio, Strings, Exeinfo dan Virus) dan runtime analisis (Regshot, CaptureBAT, Noriben Malware Analysis). Analisa yang dilakukan menciptakan lingkungan kerja sebagai tempat uji coba, sehingga tidak menggangu sistem utama yang telah berjalan. Dengan uji coba yang dilakukan, jadi didapatkan bahwa File dengan nama "games.exe" dapat dikatakan malware karena memiliki strings dan dapat menggandakan diri ke folder system32 yang dibuktikan dengan MD5.*

Kata kunci*: Malware, Surface Analisis, Runtime Analisis, Windows*

1. PENDAHULUAN

Malicous Sofware atau dikenal sebagai *Malware* merupakan perangkat lunak yang didesain untuk melakukan aktifitas berbahaya atau perusak perangkat lunak [1]. *Malware* terbagi menjadi 9 kelompok : *Backdoor, Botnet, Downloader, Information-stealing Malware, Launcher, Rootkit, Scareware, Spam-sending Malware, Worm* atau *Virus* [2]. Kelompok *malware* ini terus berkembang semakin mutakhir, kompleks dan selalu melakukan perubahan guna mencari-cari celah keamanan dengan membuat banyak varian pada pola serangan. Tren *malware* yang menjadi

ISSN : 1411-1624
e-ISSN: 2621-8089

perbincangan saat ini salah satunya *Ransomware Wannacry* yang pernah terjadi pada rumah sakit yaitu Dharmais, *ransomwere* yang berjenis *malicious software* menyerang komputer dengan mengenkripsi seluruh data yang ada di rumah sakit, sehingga pengguna tidak bisa untuk mengakses data itu kembali [3]. Masuknya *malware* kedalam sistem bisa dengan berbagai cara, seperti diselundupkan didalam kumpulan *file* atau aplikasi tertentu sehingga pengguna tidak menyadari bahwa perangkat komputer yang digunakan telah disusupi *malware*.

Malware memiliki banyak cara dalam menginfeksi, seperti : *email attachment*, *script* dari halaman *web, link* ke dalam halaman *web* yang merupakan file yang siap di *install.* Melakukan *bug* pada sistem operasi, *USB Driver, file sharing* dan juga aplikasi bajakan [2]. *Malware* juga sering disamarkan dengan menggunakan *file* umum seperti *Driver* (.drv), data (.dat), *library* (.lib), *temporary* (.tmp) dan lain sebagainya,yang terkadang pengguna tidak menyadari kehadiran *file* tersebut didalam komputer yang digunakan [1].

Beberapa sistem operasi rentan terhadap yang namnya sebuah *malware* salah satunya sistem operasi, beberapa yang popular yang dapat menginfeksi sistem operasi windows yaitu *shortcut, LNK.Autostart, Sality* dan *Ramit* yang pernah menjadi penyebaran tertinggi sebuah *malware* di Indonesia pada tahun 2010 sampai dengan 2013. [4]

Analisis *malware* diperlukan untuk mengetahui (1) ciri-ciri sebuah *malware,* (2) melihat pola serangan, (3) apa saja efek yang ditimbulkan yang nantinya memberikan informasi perbaikkan sistem, (4) mengantisipasi apabila terjadi serangan *malware.* Dalam menganalisis *malware* dapat menggunakan metode *Surface Analysis* dan *Runtime Analysis*.

Surface Analisis adalah metode yang menganalisis jenis *file* asli, ukuran *file* sebenarnya, sehingga memberikan informasi untuk mengetahui *malware* yang tersembunyi atau menyamar didalam *file* lain. Sedangkan *Runtime Analysis,* menjalankan atau mengaktifkan *file* yang diperiksa dalam mendapatkan informasi perilaku dari program yang menjalankan skenario jahat dan dapat dianalisis dampak terhadap sistem yang ada [1].

Dengan latar belakang yang ada, peneliti bermaksud melakukan pengujian analisis tingkah laku *malware* yang akan diterapkan pada sistem operasi *windows,* sehingga dapat diketahui pola serangan yang nantinya akan dikelompokkan berdasarkan pola-pola sebuah *malware.* Analisa dilakukan dengan menciptakan lingkungan kerja sebagai tempat uji coba sehingga tidak menggangu sistem utama yang telah berjalan.

2. METODOLOGI PENELITIAN

2.1 *Metode Analysis*

Metode yang digunakan dalam penelitian ini adalah *surface analysis* dan *runtime analysis*. Kedua metode ini merupakan pendekatan umum yang digunakan dalam mendeteksi *file* atau program telah teridentifikasi sebagai jenis *malware* atau bukan.[1].

ISSN : 1411-1624
e-ISSN: 2621-8089

2.1.1 Metode *Surface Analysis*

Kegiatan pertama dalam metode ini menyiapkan bahan-bahan yang diperlukan sampel maupun tool, tools yang digunakan adalah *Pestudio, Strings, Exeinfo* dan *Virus* total. Tahapan dari metode ini:

a. *File Attribute Analysis*

Pada tahapan ini tool yang digunakan adalah *Pestudio*, aplikasi ini digunakan untuk melihat atribut dari sample *malware*, mulai dari MD5, *hash, strings, header* sebagai kebutuhan dari analisa.

b. *Fuzzy Hashing*

Tahapan ini memastikan bahwa sample *malware* yang dianalisa sama dengan sample aslinya, apakah terjadi perubahan pada MD5 dan *hash* nya ketika file ”games.exe” dipindahkan dari *windows* ke *remnux*, maka perlu dilakukan verifikasi melalui kecocokan nilai MD5 yang ada pada *file.* Tools yang digunakan MD5SUM dan ssdeep dalam mencocokan nilai MD5 dan hash.

c. *Packer Check*

Kegiatan dalam tahapan ini untuk melihat apakah *file* tersebut di *packing* atau tidak, karena jika *file* di *packing* maka informasi yang akan ditampilkan dalam menganalisa stringnya akan di enkripsi. Jika *file* yang akan dianalisa di *pack* maka perlu melakukan *unpack* untuk dapat melihat *strings* dan membutuhkan tool yang sama digunakan untuk membongkarnya yaitu dengan Exeinfo

d. *Analisis String*

Strings adalah *squence of characters* sebuah program, sehingga mencari *string* dalam *malware* merupakan hal penting untuk mengetahui pola kerja dari *malware*. Biasanya *string* sebuah *file malware*, akan menampilkan *IP address* pencarian *file*, menduplikat *file*, dan mengirimkan *message*. Tool yang digunakan adalah *Pestudio* pada sistem operasi *windows*, dan Yara dan String pada sistem operasi *renmux*

e. *Malware Scan*

Setelah diterapkan ke 4 tahapan, selanjutnya *file* yang dijadikan sample diperiksa apakah telah terdeteksi *malware* dengan menggunakan MD5 yang di *upload* pada [5] Jika *file* terdeteksi *malware*, maka kita dapat melihat informasi apa saja yang didapat yang setelah dianalisa antivirus, sehingga lebih fokus dan telah terarah.

2.1.2 Metode *Runtime Analysis*

Setelah menyelesaikan tahapan pada metode *surface analysis*, maka akan dilanjutkan pada metode *runtime analysis* dengan menjalankan *file*. Metode ini menggunakan tools *Regshot, CaptureBAT, Noriben Malware Analysis.* Tahapan metode ini adalah:

a. Melihat Perubahan *Registry*

Tools yang digunakan pada tahapan ini adalah *regshot* untuk melihat perubahan pada *registry* ketika *file* dijalankan. Tool ini melakukan dua kali pemeriksaan*/shoot, shoot* yang pertama untuk mendapatkan *registry* sebelum *file* dijalankan. Selanjutnya melakukan *Shoot* yang kedua dan tunggu beberapa saat sampai selesai, kemudian dilihat perubahan apa saja yang telah terjadi dengan melakukan *compare* untuk mendapatkan hasil analisa. Hasil *shoot* akan ditampilkan dalam bentuk teks pada *notepad*, dan kemudian melakukan analisa pada *registry*

ISSN : 1411-1624
e-ISSN: 2621-8089

b. Melihat aktivitas *malware* didalam sebuah jaringan

Disini dapat dilakukan monitoring terhadap jaringan dengan menggunakan *CaptureBAT*. Aplikasi ini bekerja dengan cara memonitoring aktivitas jaringan sebuah sistem operasi *windows* dan mengubahnya dalam bentuk *capture* yang dapat dibuka dengan menggunakan aplikasi *wireshark*. Aplikasi memantau apa saja yang dilakukan ketika *file* ”game.exe” dijalankan

c. Analisa otomaris dengan *Noriben Malware Analysis Sanbox*

Analisa ini dilakukan untuk lebih menguatkan hasil yang didapat dari tahapan sebelumnya dengan menggunakan tools *Noriben* yang dapat mencatat aktivitas dalam sebuah sistem operasi secara otomatis yang bekerjasama dengan tool *procmon*, agar mendapatkan hasil yang optimal dan lebih akurat.

2.2 *Sample Malware*

Dalam penelitian ini *sample malware* yang dianalisa adalah ”games.exe”, yang akan diidentifikasi apakah termasuk kedalam jenis *malware*, dan jenis *malware* yang seperti apa, serta bagaimana perilakunya ketika berhasil menginfeksi sebuah sistem operasi yang akan digunakan sebagai sample yaitu *windows.*

2.3. Alat dan Bahan

a. *Hardware* dan *Software*

Spesifikasi *Hardware* dan *Software*Dalam menganalisis *malware* dibutuhkan beberapa alat dan bahan yang terdiri dari perangkat keras (*Hardware*) dan perangkat lunak (*Software*), yang terdapat pada table 1 dan table 2

Tabel 1. Spesifikasi *Hardware*

Hardware	**Spesifikasi**
Laptop	Processor intel (R) Pentium(R) CPU B940 32 Bit
Flasdisk	Kapasistas 8GB

Tabel 2. Spesifikasi *Software*

Software	**Spesifikasi**
Sistem Operasi	
Windows	Pro 64-bit
Remnux	Versi 4.0
Tools* metode *Surface Analysis	
Pestudio	Versi 8.50
Exeinfo	Versi 0.0.4.4
Md5sum	*Malware Scan*
ssdeep	*Malware Scan*
Yara Rule	*String analysis*
www.virustotal.com	*Malware Scan*
Tools* metode *Runtime Analysis	
Process Hacker	*Task Manager*
CaptureBAT	Runing in Windows
Noriben Sanbox	Versi 1.7.2
Wireshark	Versi 2.4.11

b. *VMare*

VMware adalah teknologi buatan *Dell* yang menyediakan *platform* perangkat lunak (*software)* untuk melakukan virtualisasi [6], dengan *software* ini peneliti bisa menginstal beberapa system operasi yaitu *linux* dan *windows* pada waktu yang bersamaan tanpa kita merestart ulang PC atau bisa dikatakan PC didalam PC

c. *Remnux v* 4.0

Remnux adalah sistem operasi yang berbasis linux Ubuntu, yang membedakkan *Remnux* menyediakan lebih banyak *tools* dalam melakukan analisis pada *malware*. *Tools* yang akan digunakan untuk menganalisa malware yang berada pada *file* berbentuk “exe” [7].

d. *Wireshark*

Salah satu dari *tools Network Analyszer* adalah wireshark, yang banyak disukai karena menggunakan antarmuka yang baik atau *Graphical User Interface* (GUI), *tool* ini tersedia

ISSN : 1411-1624
e-ISSN: 2621-8089

opensource dan mampu menangkap semua paket data atau informasi yang berada dalam jaringan [8].

e. *Registry*

Registry adalah sebuah ruang control utama dari sebuah system operasi *windows*, dikarenakan *registry* adalah basis data dan sebagai pusat pengaturan/konfigurasi *windows.* [6]

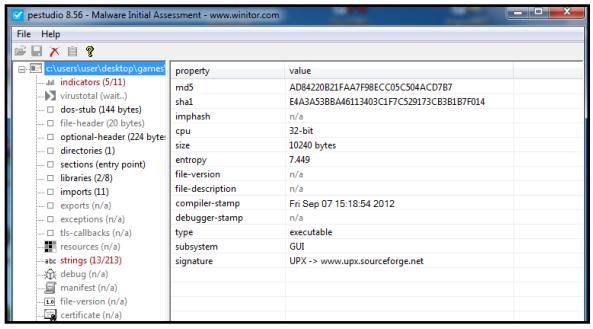
3. HASIL DAN PEMBAHASAN

3.1. *Metode Surface Analysis*

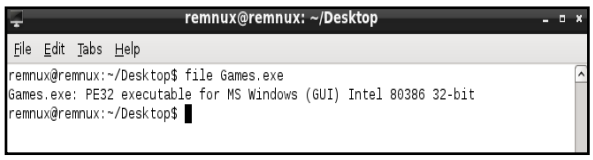
Metode ini akan mengidentifikasi unsur-unsur yang dterdapat didalam sebuah *file* yang dicurigai sebagai *malware.*

3.1.1 *File Attribute Analysis*

Tahapan ini berhasil mendapatkan atribut dari sample dengan tipe *executable* pada sistem 32-bit dan *MD5* melalui sistem operasi *Windows* dan *Remnux.*

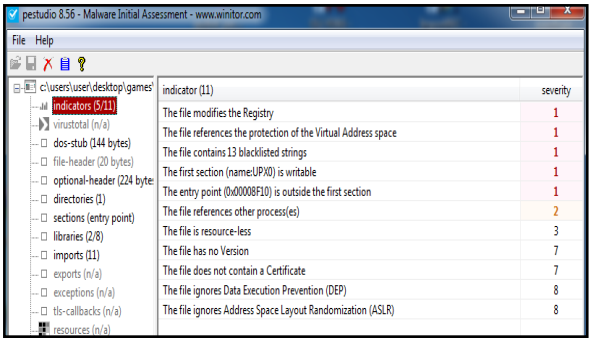


Gambar 1. *File Atrribute* di *Pestudio Windows.*



Gambar 2. *File Atrribute* di *Remnux*

Selanjutnya melihat indikator yang dihasilkan *pestudio* dengan menggunakan angka 1 sampai 9 sebagai alamat, angka 1 dan 2 menunjukkan bahwa *file* tersebut memiliki tingkat bahaya yang cukup tinggi, karena memiliki kemampuan yang biasanya ada pada *malware*, seperti memodifikasi *registry* dan terdapat *blacklist* pada stringnya yaitu melakukan perubahan pada *file*, menghapus serta merusak. Maka, semakin banyak angka 1 dan 2, maka semakin besar pula kemungkinan *file* tersebut merupakan sebuah *malware*. Terindentifikasi *file* "game.exe" terdapat 5 dan 11 indikator yang menunjukkan bahwa *file* tersebut kemungkinan merupakan *file* yang berbahaya.

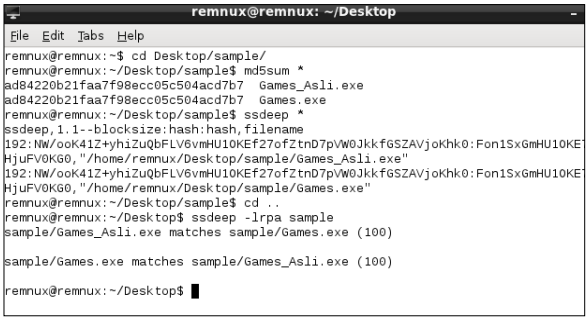


Gambar 3. Indikators *file Games.exe*

3.1.2 *Fuzzy Hashing*

Dalam mengindentifikasi keaslian *malware* yang akan dianalisa, kedua *file* diletakkan dalam satu folder yang sama untuk memudahkan pencocokan MD5 dan *hash* pada sistem operasi . Terlihat pada gambar 4 menunjukkan bahwa MD5 dan *hash* memiliki nilai kesamaan 100% yang berarti *file* yang dianalisa sama persis *file* asli.

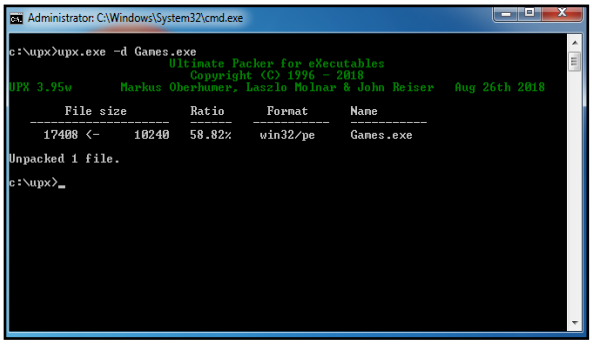
ISSN : 1411-1624
e-ISSN: 2621-8089



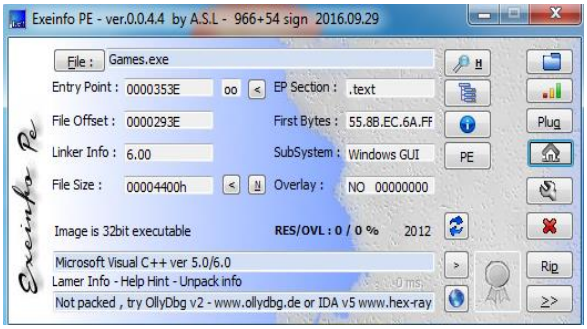
Gambar 4. *Fuzzy Hasing file* sampel

3.1.3 *Packer Chack.*

Sebelum mulai melakukan analisa *file* harus di *unpacking* untuk dapat melihat string yang ada pada *file* tersebut, agar tidak mempersulit dalam melakukan analisa nantinya. Tool *unpacking* yang didapat dari informasi *exeinfo* adalah UPX 0.89 dari [9], Setalah tool didapat maka tinggal dijalankan melalui *command prompt* (CMD) dengan menempatkan *file* satu *folder* bersama dengan tool tersebut.



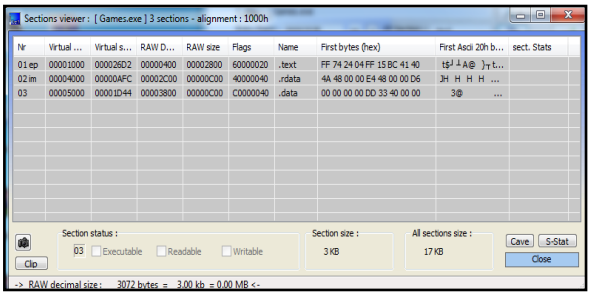
Gambar 5. Hasil *unpacking file Games.exe*



Gambar 6. Hasil setelah *unpacking* pada *Exeinfo*

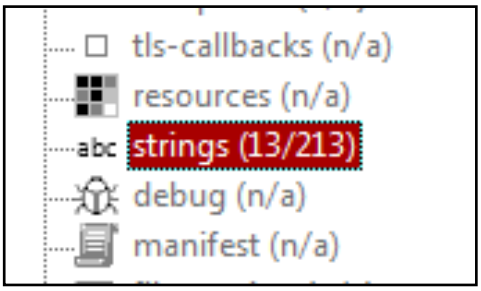
Kemudian pada *Portable Executable (PE) Section* terdapat *file* PE yang berisi *header* dan beberapa bagian yang penting.

- .text : berisikan kode yang dapat di eksekkusi
- .rdata : menampung dan membaca data yang dapat diakses secara global.
- .data : menyimpan data global yang dapat di akses hanya melalui program saja.

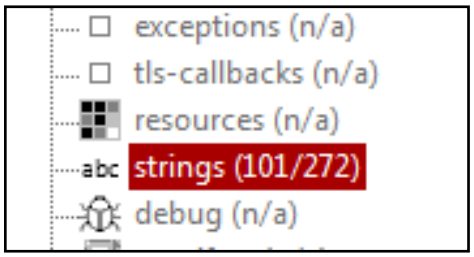


Gambar 6. Tampilan *Portable Executable (PE) Section*

Maka dapat melihat perubahan string pada *file* yang tampil lebih banyak dari sebelum *file* di *unpacking* seperti pada gambar 7 dan Gambar 8.



Gambar 7. *File* sebelum *Unpacking*

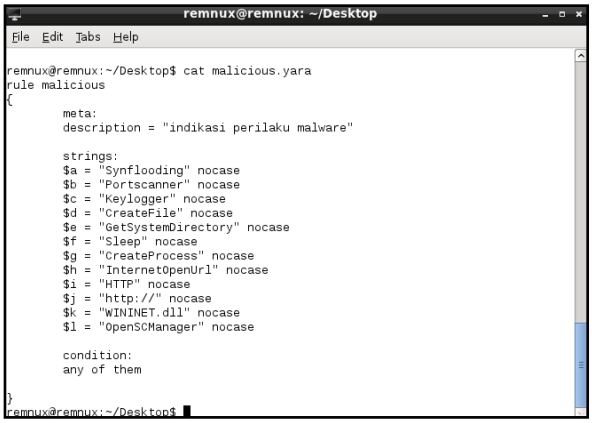


Gambar 8. *File* sesudah *Unpacking*

ISSN : 1411-1624
e-ISSN: 2621-8089

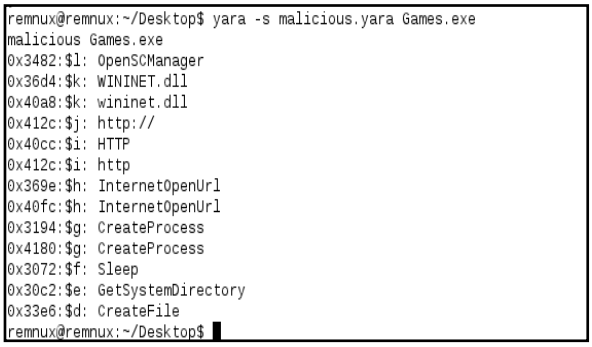
3.1.4 *String analysis*

Tools yang digunakan selanjutnya adalah Yara untuk menganalisa keberadaan string pada *file* dengan aturan yang sudah dibuat seperti pada gambar 12 dibawah ini.



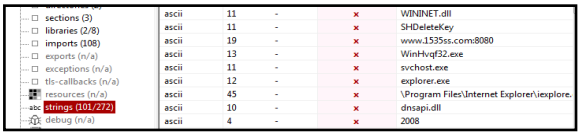
Gambar 9. Yara Rule.

Dengan menggunakan aturan ini akan dilihat ada berapa string yang didapatkan Yara Rule. Semakin banyak string yang didapatkan maka semakin besar kemungkinan *file* merupakan sebuah *malware* karena aturan yang ada pada Yara Rule mampu membaca string pada program.



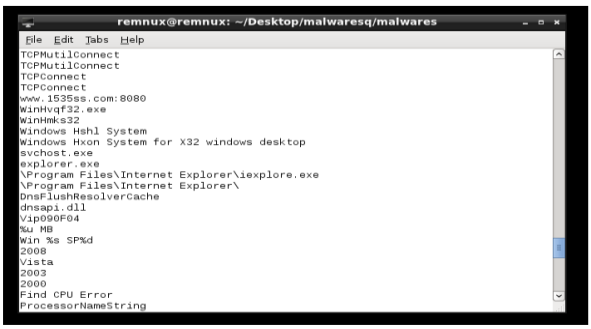
Gambar 10. Hasil Yara Rule

Selanjutnya menggunakan Pestudio untuk melihat string yang ada pada program, dan telihat bahwa *blacklist* dari program menampilkan banyak informasi bahwa *file* tersebut besar kemungkinan adalah sebuah *malware.*



Gambar 11. Analisa string dengan Pestudio di Windows.

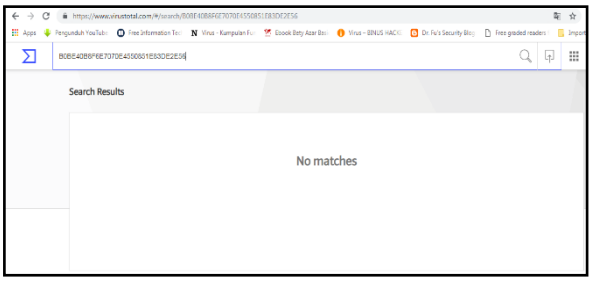
Dari informasi yang didapat, *file* tersebut memliki 101 *blacklist* dari 272 jumlah string, hal ini menunjukkan bahwa *file* tersebut banyak melakukan tindakan ketika di eksekusi. Hasil yang didapat diantaranya terlihat jelas bahwa *file* tersebut mengakses sebuah alamat web melalui port 8080, dengan menjalankan *file* ”*WinHmks32.exe*” dan ”*svchost.exe*”.



Gambar 12. Analisa string dengan Strings di Remnux.

3.1.5 *Malware Scan*

Pada tahap ini hasil scan dari *sample* menunjukkan bahwa *file* belum terdeteksi sebagai sebuah *malware* melalui MD5, dari hal tersebut kita mencari lebih banyak pada metode selanjutnya dengan cara mengeksekusi *file*.



Gambar 13. Scan *file* di Virus Total.

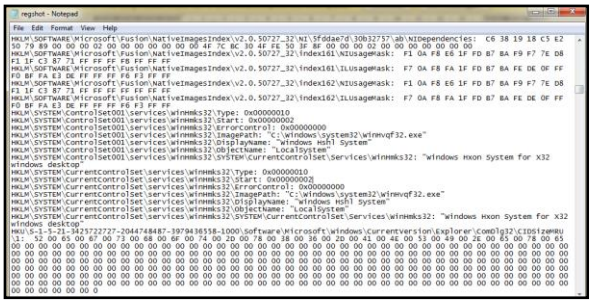
ISSN : 1411-1624
e-ISSN: 2621-8089

3.2 Metode *Runtime Analysis*

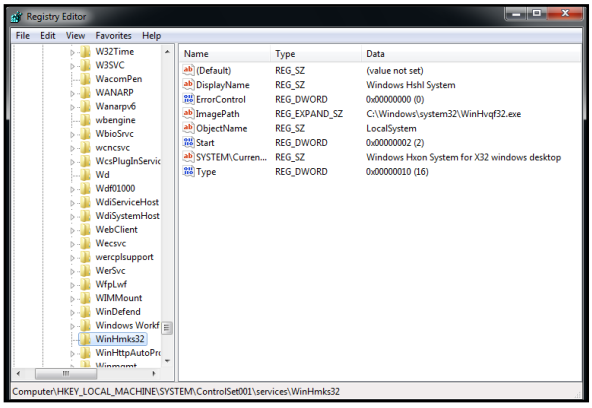
Tahap ini menjalankan *file* yang akan dianalisa untuk melihat aktivitas yang dilakukan, sehingga dapat dinilai apakah *file* tersebut merupakan *malware* atau bukan

3.2.1 *Melihat Perubahan Registry*

Dengan menggunakan *Regshot* dapat dilihat perubahan yang terjadi pada *registry* setelah *file* dijalankan. Terlihat *file* melakukan beberapa perubahan dengan menambahkan beberapa *file registry*.



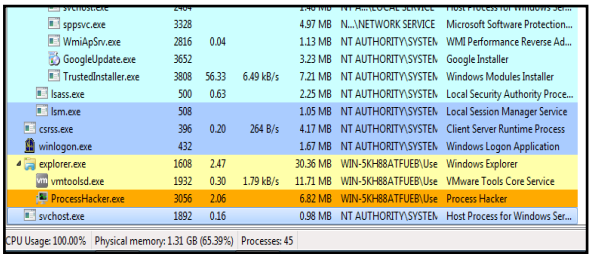
Gambar 14. Hasil monitoring *Regshot*



Gambar 15. Penambahan *file registry*

3.2.2 Melihat aktivitas malware didalam sebuah jaringan

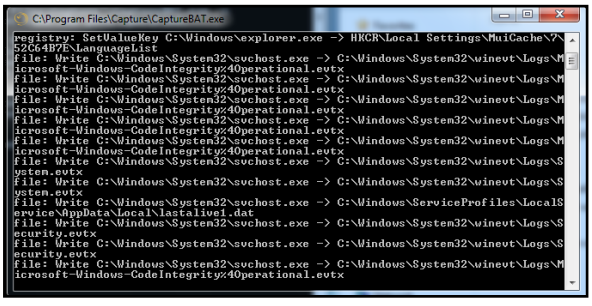
Aplikasi ini memeriksa semua *file* yang sedang berjalan didalam sistem, mirip seperti *task manager* yang ada pada windows, namun lebih jelas dan rinci. Pada proses menjadi Hacker terlihat adanya proses *svchost.exe*. ketika *file* ”games.exe” di eksekusi. Ketika *file* ”svchost.exe” yang diterminate, maka dapat dilihat pada *wireshark* layanan yang meminta akses ke *www.1535ss.com* berhenti.



Gambar 16. Proses yang berjalan pada system dengan *Process Hacker*

3.2.3 Analisa otomatis dengan *Noriben Malware Analysis Sanbox*

Dengan menggunakan *captureBAT* akan melihat perilaku dari *file* ”games.exe” setelah *file* tersebut di eksekusi. Eksekusi dilakukan dengan meng*install file captureBAT* pada windows, selanjutnya mempersiapkan wireshark pada remnux untuk melihat paket data yang berjalan selama proses berlangsung pada *captureBAT*. *CaptureBAT* akan menyimpan hasil *capture* secara otomatis bentuk ”*capture_11102018_1056.zip”*.



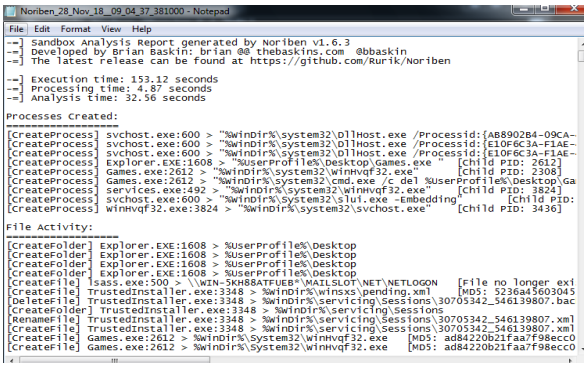
Gambar 17. Gambar *CaptureBAT*.

3.2.4 Analisa menggunakan *Noriben Malware Analysis Sanbox*

Noriben bekerjasama dengan procmon dalam memonitoring aktivitas yang terjadi dalam

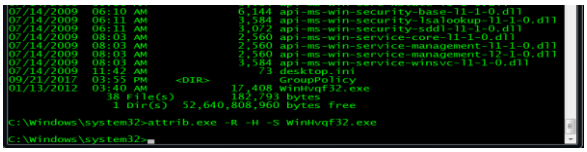
ISSN : 1411-1624
e-ISSN: 2621-8089

sistem, sehingga perlu meletakkan *file* procmon dalam satu folder. Tempatkan *sample* di desktop agar mudah ditemukan, dan selanjutnya mengecek IP yang ada pada windows, jika semua sudah selesai berahli ke sistem operasi remnux untuk menjalankan wireshark yang nantinya akan memonitoring semua lalulintas jaringan dengan ditambah sebuah perangkat lunak **Inetsim** untuk memanipulasi jaringan lokal, sehingga seperti terlihat berjalan pada jaringan internet.



Gambar 18. Hasil Analisa Noriben

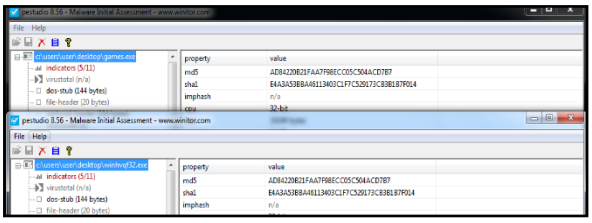
Hasil dari analisa Noriben menunjukkan pada *proseses crated* adanya proses ”*WinHvqf32.exe”* ketika *file ”games.exe”* dijalankan dan *file* tersebut meminta layanan ”*svchost.exe”* merupakan bagian integral dari sistem operasi windows untuk melalui layanan service yang dapat melakukan *automatic update*, hal ini menjadi mungkin bagi *file ”games.exe”* untuk mengakses jaringan. Pada *file activity* juga menampilkan adanya aktivitas pembuatan *file* ”*WinHvqf32.exe”* oleh *”games.exe”*yang memiliki MD5 sama persis di lokasi *system32*. Ada yang menarik dengan *file* ini, ketika dicari secara biasa tidak bisa ditemukkan, karena *file* di *super hidden* atau disembunyikan dengan sangat baik, sehingga meskipun pengaturan *show hidden folder, file* dan *driver* sudah diaktifkan, *file* tetap tidak muncul dan terlihat, dari hal tersebut dapat menggunakan trik tambahan untuk menampilkan *file* yang sudah di *super hidden* dengan cara masuk ke *commmand prompt (cmd)*, ketika ”*dir/ah”* seluruh *file* akan tampil semua termasuk yang telah di *hidden*. Selanjutnya akan menampilan *file* tersebut di *windows explorer* karena meskipun sudah tampil di *cmd*, *file* tersebut belum tampil pada *windows explorer*.



Gambar 18. Menampilkan *file* yang *hidden*

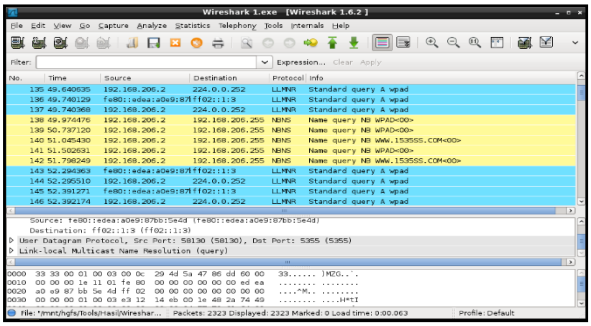
Gambar 19. *file* di system32 yang tampil

Jika semua telah dilakukan, *file* akan terlihat pada *windows explore* dan dapat menganalisa apakah benar *file* dengan *file ”games.exe”* dengan menggunakan *pestudio.*



Gambar 20. Gambar kecocokan MD5 di *Pestudio*

Monitoring paket data jaringan menggunakan *wireshark.*



Gambar 21. Hasil Capture *Wireshark*

ISSN : 1411-1624
e-ISSN: 2621-8089

Hasil monitoring yang dilakukan wireshark ketika *file "games.exe"* di eksekusi pada sistem operasi windows telah menangkap adanya komunikasi yang dilakukan *IP 192.168.206.2* kepada *192.168.208.225* menggunakan protocol NBNS. Rangkuman dari analisis dengan metode *surface analysis* dan *runtime analysis* dirangkum pada tabel 2 dan tabel 3.

Tabel 2. Hasil Analisa metode *Surface Analysis*

No	*Tahapan*	*Surface Analysis*	
		Tool	***Hasil Temuan***
1	*File Attribute analysis*	Pestudio	*File Name* *Games.exe* *MD5* *AD84220 B21FAA7 F98ECC0 5C504AC D7B7* *CPU Type* *32-bit executable* *Indicators* 4/14
2	*Phackers Check*	Exeinfo	*UPX 0.89 - 3.xx -> Markus & Laszlo ver. [3.95] unpack "upx.exe -d" from http://upx.sf.net or any UPX/Generic unpacker* *.text* *.rdata* *.data*
3	*Fuzzy Hashing*	Md5sum ssdeep	Sama 100%
4	*Analisa Strings*	Pestudio Yara dan Strings	101/272 *www.1535ss.com* *OpenSCManager* *WININET.dll* *http://* *HTTP* *ExitProcess* *CreateProcess* *InternetOpenUrl* *Sleep* *GetSystemDirectory*
5	*Malware Scan*	*www.virustotal.com*	Belum teridentifikasi sebagai *malware*

Tabel 3. Hasil Analisa metode *Runtime Analysis*

No	*Temuan*	*Runtime Analysis*			
		Process Hacker	Capture BAT	Noriben Sanbox	Wireshark
1	Penambahan registry	-	✓	✓	-
2	Penambahan *file* baru pada C:\Windows\system32	-	✓	✓	-
3	*File* hidden	-	✓	✓	-
4	Alamat IP/web Program : www.1535ss.com	-	-	-	✓
5	Nomor Port yang digunakan : 8080	-	-	-	✓
6	Protocol yang digunakan ; NBNS	-	-	-	✓
7	*File* yang berjalan dilatar belakang	✓	-	-	-

4. KESIMPULAN

Pada analisis yang terlah dilakukan dengan menggunakan kedua metode analisi *malware yaitu Surfcae dan Runtime* dapat dinyatakan bahwa : *File "games.exe"* jelas dapat dikatakan sebuah *malware* dikarenakan miliki strings yang pada umumnya terdapat sebuah *malware*. *File* dapat menggandakan diri ke folder *system32* "*WinHvaq32*.exe" yang dibuktikan dengan kesamaan MD5.

ISSN : 1411-1624
e-ISSN: 2621-8089

UCAPAN TERIMA KASIH

Saya ingin menyampaikan ucapan terimakasih yang sangat besar kepada Universitas Bina Darma dan Direktorat Riset dan Pengabdian Masyarakat Bina Darma (DRPM) atas dukungannya selama pengembangan karya penelitian ini.

DAFTAR PUSTAKA

[1] R. E. Indrajit, "Skenario Kombinasi Tools yang Efektif dalam Analisis Malware," *Am. J. Appl. Sci.*, vol. 9, no. 3, pp. 1–22, 2011.

[2] T. A. Cahyanto, V. Wahanggara, and D. Ramadana, "Analisis dan Deteksi Malware Menggunakan Metode Malware Analisis Dinamis dan Malware Analisis Statis," *Justindo, J. Sist. Teknol. Inf. Indones.*, vol. 2, no. 1, pp. 19–30, 2017.

[3] C. Indonesia, "Dua Rumah Sakit di Jakarta Kena Serangan Ransomware WannaCry," *https://www.cnnindonesia.com*. [Online]. Available: https://www.cnnindonesia.com/teknologi/20170513191519-192-214642/dua-rumah-sakit-di-jakarta-kena-serangan-ransomware-wannacry. [Accessed: 05-Mar-2018].

[4] "Quick Count Malware Top Indonesia," *Detikinet.com*, 2014. [Online]. Available: https://inet.detik.com/security/d-2640265/quick-count-malware-top-indonesia. [Accessed: 20-Mar-2018].

[5] Virustotal.com, "Virustotal," 2018. [Online]. Available: https://www.virustotal.com/gui/home/upload. [Accessed: 05-May-2018].

[6] E. Haryanto, "Analisis Forensik WSO Webshell.....Platform Linux." .

[7] Y. A. Utomo *et al.*, "Membangun Sistem Analisis Malware Pada Aplikasi Android Dengan Metode Reverse Engineering Menggunakan Remnux," vol. 4, no. 3, pp. 2000–2012, 2018.

[8] R. Yuvandra and M. Zulfin, "Analisis Kinerja Trafik Video Chatting Pada Sistem Client-Client Dengan Aplikasi Wireshark."

[9] Http://upx.sf.net, "upx," 2018. [Online]. Available: http://upx.sf.net. [Accessed: 10-May-2018].