ANALISIS DAMPAK *MALWARE* BERDASARKAN API *CALL* DENGAN METODE *CLUSTERING*

MALWARE IMPACT ANALYSIS BASED ON API CALL USING CLUSTERING METHOD

Ari Apridana[1], M. Teguh Kurniawan [2], Adityas Widjajarto [3]

[1,2,3]Prodi S1 Sistem Informasi, Fakultas Rekayasa Industri, Universitas Telkom
[1]**ariapridana@student.telkomuniversity.ac.id**, [2]**teguh.kurniawan@telkomuniversity.ac.id**,
[3]**adtwjrt@telkomuniversity.ac.id**

Abstrak

Setiap hari muncul berbagai jenis *malware* baru di Internet. Di mana teknologi yang digunakan *malware* sudah semakin berkembang. *Malware* sudah menggunakan berbagai teknik dan cara untuk mengelabui antivirus ketika menginfeksi korban. Perkembangan *malware* harus diimbangi dengan melakukan penelitian tentang deteksi malware. Maka dari itu, pada penelitian ini dilakukan kegiatan *malware analysis* menggunakan sebanyak 170 *malware* untuk melakukan kategoriasi berdasarkan *malicious activity* yang digunakan. *Malicious activity* dapat dilihat dari jenis API *call* yang digunakan. Pada penelitian ini menggunakan 3 (tiga) jenis API *call* yaitu API *file*, API *process* dan API *registry*. Teknik yang digunakan untuk mendapatkan informasi API *call* adalah *Static Analysis* dan *Dynamic Analysis*. Sehingga hasil yang diperoleh terdapat 5 kategorisasi dengan minimum 0 kombinasi dan maksimum 5 kombinasi *malicious activity.* Dimana pada anlisis ini menggunakan metode *clustering* sehingga dibagi menjadi 3 cluster yang memiliki lebih dari 1 (satu) kombinasi *malicious activity*. Analisis *impact* yang diberikan tergantung dari banyaknya kombinasi, semakin banyak kombinasi API *call* akan semakin besar dampaknya pada komputer yang terinfeksi. Pada penelitian ini *impact* yang diberikan adalah *high* karena terdapat malware yang memiliki 2 (dua) kombinasi sampai 5 (lima) kombinasi *malicious activity*.

Kata kunci : *malware, malware analysis, cyber crime, clustering,* deteksi *malware*, *malware signature, malicious activity.*

Abstract

Every day there are various types of new malware on the Internet. Where the technology used malware is growing. Malware already uses various techniques and ways to trick the antivirus when it infects the victim. The development of malware should be offset by doing research on malware detection. Therefore, in this study conducted malware analysis activities using as many as 170 malware to do the category based on its malicious activity. Malicious activity can be seen from the type of API call used. In this study using 3 (three) types of API call API file, API process and API registry. Techniques used to obtain API call information are Static Analysis and Dynamic Analysis. So the results obtained there are 5 categorizations with a minimum of 0 combinations and a maximum of 5 combinations of malicious activity. Where in this analysis using clustering method that is divided into 3 clusters that have more than 1 (one) combination of malicious activity. The impact analysis provided depends on the number of combinations, the more API API combinations the greater the impactny on the infected computer. In this study the impact given is high because there are malware that has 2 (two) combinations up to 5 (five) combination of malicious activity.

Keywords: malware, malware analysis, cyber crime, *clustering*, malware detection, malware signature, malicious activity.

1. Pendahuluan

Perkembangan Internet saat ini telah mempermudah manusia untuk saling terhubung menggunakan komputer. Didukung dengan perkembangan teknologi yang sangat cepat. Namun tidak semua perkembangan teknologi selalu memberikan dampak yang positif. Banyak juga dampak negatif dari perkembangan teknologi, salah satunya *cyber crime*. *Cyber crime* merupakan istilah kejahatan *cyber* yang menggunakan teknologi informasi secara langsung maupun tidak langsung untuk mendapatkan keuntungan (Saputra, 2016). Serangan *cyber crime* sekarang sangatlah terstruktur dan rapi. Serangan yang banyak dilakukan adalah *malware*. *Malware* adalah singkatan dari *malicious software* merupakan istilah yang digunakan untuk perangkat lunak berbahaya. *Malware* dirancang untuk merusak maupun melakukan *control* ke suatu sistem tertentu. Tujuan akhir serangan *malware* adalah dapat terinstall di komputer korban. Sehingga memungkinkan penyerang mendapatkan akses penuh dari komputer korban (Sikorski, 2012). Target dari serangan *malware* biasanya merupakan instansi pemerintah maupun organisasi. Data yang diambil berupa informasi *username, password, credit card*.

Pada umumnya penyebaran *malware* biasanya menggunakan banyak cara. Contohnya *Social Engineering , Email Phising, File Download Fraud* untuk membuat korban tertipu sehingga terinfeksi *malware*. Ketika *malware* sudah menginfeksi komputer korban maka *malware* mulai bekerja sesuai fungsinya. Dari fungsi yang dilakukan oleh *malware* tersebut akan dapat menentukan *malware* tersebut masuk dalam jenis *malicious activity* yang mana. Namun harus melakukan tahapan *malware analysis* terlebih dahulu (Sikorski, 2012).

Karena makin aktifnya penyebaran *malware* dan makin beragamnya jenis *malware* saat ini. Untuk itu akan dilakukan *malware anlysis* untuk mendapatkan informasi API *call* yang digunakan. API *call* merupakan fungsi *library* yang dapat digunakan pada sistem operasi Windows. Jika mengetahui informasi API *call* yang digunakan oleh *malware*, maka dapat mengetahui *malicious activity* yang dilakukan oleh *malware*. Karena itu dari informasi API *call* yang didapatkan, kemudian dilakukan kategorisasi menggunakan metode *clustering* berdasarkan *malicious activity* yang digunakan. *Clustering* digunakan untuk mengelompokan beberapa API *call* dari *malware* yang memiliki kesamaan ke dalam beberapa kelompok kecil, sehingga dapat memudahkan proses analis untuk mengetahui *impact* dari *malware* (Hector & Widom , 2002). Dengan menggunakan metode *clustering* proses pengelompokan data bisa menjadi lebih cepat (Atmajaya, 2016) . Hasil analisis *impact* dari kategorisasi akan berguna di masa yang akan datang.

2. Dasar Teori

2.1 *Malicious Software (Malware)*

Malware merupakan perangkat lunak yang diciptakan untuk mengganggu kinerja komputer, mengumpulkan informasi sensitif atau mendapatkan akses ke komputer pribadi(Sikorski, M. & Honig, A, 2012).

2.2 *Malware Analysis*

Malware Analysis adalah sekumpulan proses untuk menentukan tujuan dan fungsionalitas dari *sample malware* yang sudah ditentukan seperti *virus, worm* dan *trojan horse*. Proses-proses ini merupakan sebuah langkah yang diperlukan untuk mengembangkan teknik pendeteksian yang efektif terhadap *malicious code*. Proses *malware analysis* terbagi menjadi dua teknik umum yaitu *Static Analysis* dan *Dynamic Analysis* (Dipjoyoti Deka,Nityananda Sarma & Nithin J.Panicker, 2016).

2.3 *Static Analysis*

Static analysis dapat diartikan dengan analisis sebuah perangkat lunak sebelum dijalankan yaitu dengan kata lain *static analysis* dijalankan saat *pre-execution time*. Dengan adanya *static analysis* ini, penganalisis dapat memahami *source code* dari sebuah perangkat lunak. *Static analysis* memiliki teknik analisis yaitu *signature based* dan *heuristic* (Dipjoyoti Deka,Nityananda Sarma & Nithin J.Panicker, 2016).

2.4 Dynamic Analysis

Pada teknik *dynamic analysis* diharuskan untuk menjalankan *malware* yang yang ingin dianalisa. Untuk memulai menganalisa *malware* menggunakan teknik ini. Diharuskan menyiapkan lingkungan analisa yang aman atau terisolasi sepertti *sandbox* dan *virtual machine*. Teknik ini mengumpulkan informasi tentang *registry*, *network traffic* dan pemanggilan *file system* API di Windows (Dipjoyoti Deka,Nityananda Sarma & Nithin J.Panicker, 2016)..

2.5 Metode Clustering

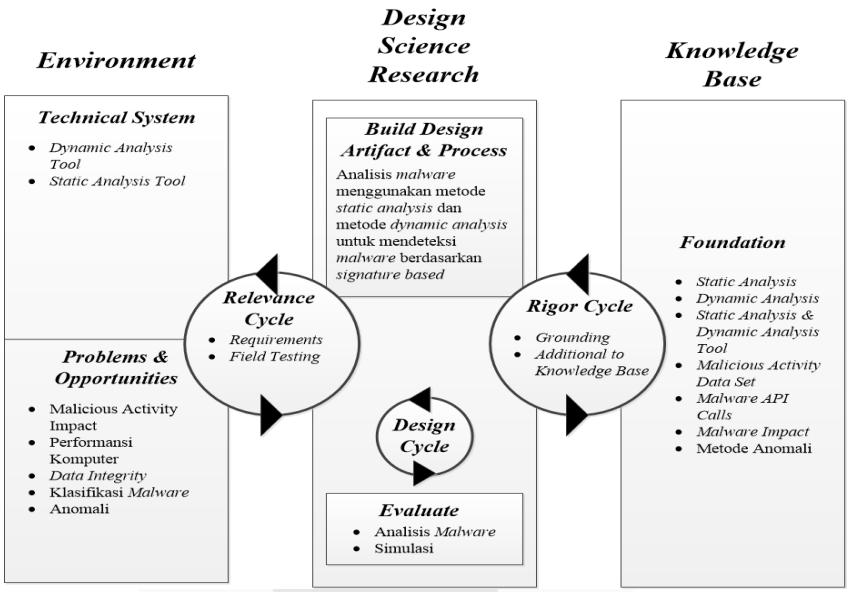
Clustering adalah metode pengelompokan data yang memiliki karakteristik tertentu ke dalam sejumlah kelompok kecil sehingga masing-masing kelompok mempunyai suatu persamaan atau kemiripan yang mendasar (Hector & Widom , 2002). Kelompok kecil tersebut disebut sebagai *cluster*.

Manfaat metode *clustering* berguna dalam memprediksi dan analisis masalah bisnis tertentu. Misalnya segmentasi pasar, pemetaan zona wilayah. Hal itu juga bermanfaat untuk memetakan *malware* berdasarkan *malicious activity* yang digunakan dengan mengelompokan beberapa API *call* yang mempunyai kemiripan antara *malware*. Dengan begitu bisa melakukan analisis lebih lanjut terhadap hasil pemetakan yang dihasilkan (Alfina, Santosa & Barkbah, 2012).

3. Metodologi Penelitian

3.1 Model Konseptual

Model konseptual merupakan suatu gambaran untuk memahami, melaksanakan dan mengevaluasi penelitian sistem informasi. Model konseptual juga biasa disebut kerangka dalam melakukan elaborasi penelitian atau bisa disebut juga dengan kumpulan dari strategi yang terbaik dari berbagai strategi yang akan digunakan (Saputra, R.W, 2016).. Tujuan dari model konseptual adalah sebagai acuan dan kerangka secara terstruktur agar dapat digunakan untuk memahami tujuan dari penelitian. Model konseptual yang digunakan dalam penelitian ini adalah sebagai berikut:



Gambar 1 Model Konseptual

4. Perancangan *software* dan *hardware*

4.1 *Data Set* API *Call Malicious Activity*

Untuk dataset yang akan digunakan sebagai data filter mengacu pada '*VirusShare Malware Sharing Platform'* (Pektas, A., Acarman, T, 2017). *Data set* ini berisi list API *call* berdasarkan *malicious activity*-nya. Hasil API *call* yang didapatkan dalam skenario *static* dan *dynamic analysis* akan dilakukan filterisasi. Guna mengetahui statistik dari *malicious activity* apa saja yang terdapat pada 170 jenis *malware*.

Tabel 3 Malicious Activity Data Set

Nomor	*Malicious Activity*	*Malicious* API *Call*
1	*Process Hollowing*	*CreateProcessInternalW*

Nomor	*Malicious Activity*	*Malicious* API *Call*
		GetModuleHandle *GetProcAddress* *VirtualAllocEx* *WriteProcessMemory* *SetThreadContext* *NtResumeThread*
2	*Create Remote Thread*	*NtOpenProcess* *GetModuleHandle* *GetProcAddress* *VirtualAllocEx* *WriteProcessMemory* *CreateRemoteThread*
3	*Enumerating All Processes*	*CreateToolhelp32Snapshot* *Process32FirstW* *Process32NextW*
4	*Drop Files from* PE *Resource Section*	*GetModuleHandle* *FindResource* *LoadResource* *NtCreateFile*
5	IAT *Hooking*	*GetModuleHandle* *Strcmp* *VirtualProtect*
6	*Delete Itself*	*GetModuleFileName* *ExitProcess* *DeleteFileW*
7	*Download & Execute* PE *Files*	*URLDownloadToFile* *ShellExecuteExW*
8	*Bind* TCP *Port*	*WSAStartup* *Socket*
9	*Capture Network*	*Socket* *Bind* *WSAIoctl* *recvfrom*

5. Pengujian Sistem dan Analisis

5.1 Kategorisasi Berdasarkan *Malicious Activity Data Set*

Berikut adalah hasil *filter* semua API *call* menggunakan *malicious activity data set*.

Tabel 7 Kategorisasi Malware Berdasarkan Malicious Activity

Malicious Activity	***Total***	***List Malware***
Process Hollowing	62 *Malware*	*Malware* 151, *Malware* 153, *Malware* 122, *Malware* 113, *Malware* 127, *Malware* 115, *Malware* 62, *Malware* 116, *Malware* 64, *Malware* 65, *Malware* 66, *Malware* 112, *Malware* 69, *Malware* 138, *Malware* 119, *Malware* 118, *Malware* 165, *Malware* 24, *Malware* 25, *Malware* 21, *Malware* 22, *Malware* 49, *Malware* 47, *Malware* 45, *Malware* 137, *Malware* 130, *Malware* 3, *Malware* 2, *Malware* 5, *Malware* 4, *Malware* 7, *Malware* 6, *Malware* 8, *Malware* 144, *Malware* 143, *Malware* 140, *Malware* 149, *Malware* 120, *Malware* 121, *Malware* 108, *Malware* 109, *Malware* 124, *Malware* 72, *Malware* 126, *Malware* 70, *Malware* 129, *Malware* 166, *Malware* 167, *Malware* 125, *Malware* 11, *Malware* 133, *Malware* 33, *Malware* 57, *Malware* 50, *Malware* 53, *Malware* 52, *Malware* 55, *Malware* 131, *Malware* 123, *Malware* 162, *Malware* 111, *Malware* 110
Create Remote Thread	27 *Malware*	*Malware* 115, *Malware* 117, *Malware* 116, *Malware* 111, *Malware* 65, *Malware* 113, *Malware* 112, *Malware* 69, *Malware* 119, *Malware* 118, *Malware* 121, *Malware* 122, *Malware* 123, *Malware* 124, *Malware* 125, *Malware* 126, *Malware* 127, *Malware* 129, *Malware* 167, *Malware* 162, *Malware* 38, *Malware* 19, *Malware* 57, *Malware* 55, *Malware* 64, *Malware* 33, *Malware* 110
Enumerating All Processes	24 *Malware*	(*Malware* 151, *Malware* 153, *Malware* 131, *Malware* 130, *Malware* 137, *Malware* 138, *Malware* 49, *Malware* 45, *Malware* 3, *Malware* 2, *Malware* 4, *Malware* 7, *Malware* 6, *Malware* 8, *Malware* 144, *Malware* 140, *Malware* 148, *Malware* 149, *Malware* 72, *Malware* 166, *Malware* 11, *Malware* 33, *Malware* 50, *Malware* 53

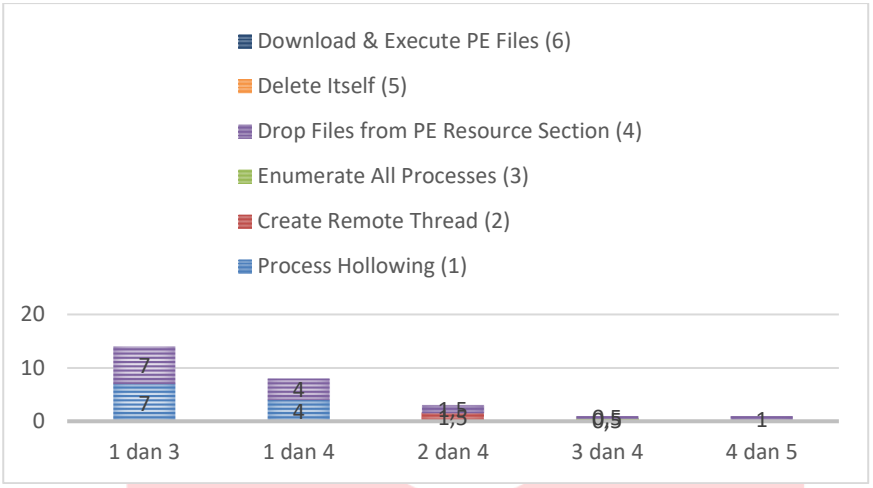
Malicious Activity	*Total*	*List Malware*
Drop Files from PE Resource Section	58 *Malware*	*Malware* 151, *Malware* 122, *Malware* 113, *Malware* 127, *Malware* 50, *Malware* 115, *Malware* 62, *Malware* 116, *Malware* 64, *Malware* 65, *Malware* 66, *Malware* 112, *Malware* 68, *Malware* 69, *Malware* 138, *Malware* 119, *Malware* 118, *Malware* 165, *Malware* 24, *Malware* 25, *Malware* 26, *Malware* 20, *Malware* 49, *Malware* 47, *Malware* 45, *Malware* 5, *Malware* 162, *Malware* 147, *Malware* 142, *Malware* 143, *Malware* 117, *Malware* 148, *Malware* 120, *Malware* 121, *Malware* 108, *Malware* 109, *Malware* 124, *Malware* 72, *Malware* 71, *Malware* 70, *Malware* 129, *Malware* 167, *Malware* 125, *Malware* 38, *Malware* 55, *Malware* 19, *Malware* 18, *Malware* 57, *Malware* 36, *Malware* 53, *Malware* 52, *Malware* 33, *Malware* 131, *Malware* 123, *Malware* 32, *Malware* 111, *Malware* 126, *Malware* 110
Delete Itself	13 *Malware*	*Malware* 151, *Malware* 26, *Malware* 20, *Malware* 49, *Malware* 33, *Malware* 57, *Malware* 45, *Malware* 50, *Malware* 55, *Malware* 64, *Malware* 65, *Malware* 162, *Malware* 72
Download & Execute PE Files	10 *Malware*	*Malware* 55, *Malware* 47, *Malware* 57, *Malware* 120, *Malware* 108, *Malware* 109, *Malware* 65, *Malware* 70, *Malware* 129, *Malware* 167

5.2 Keterkaitan Antar *Malicious Activity*

Malicious activity dari data API *call* yang diperoleh pada kategorisasi mempunyai keterkaitan sehingga dapat mempengaruhi *impact* dari suatu *malware*. *Impact* yang diberikan pada analisis ini terdapat 3 (tiga) jenis yaitu rendah (*low)*, menengah *(medium)* dan tinggi (*high)*. Berikut merupakan keterkaitan *malicious activity* dengan parameter jumlah keterkaitannya.

5.2.1 *Malicious Activity Malware* Dengan 2 Kombinasi

Pada analisis ini terdapat 5 (lima) *malicious activity* yang termasuk dalam kategorisasi 2 (dua) kombinasi dari sampel *malware* yang digunakan yaitu *process hollowing, create remote thread, enumerating all processes, drop files from* PE *resource section* dan *delete itself.* Berikut adalah *chart* mengenai jumlah *malware* yang termasuk dalam kategorisasi diatas.

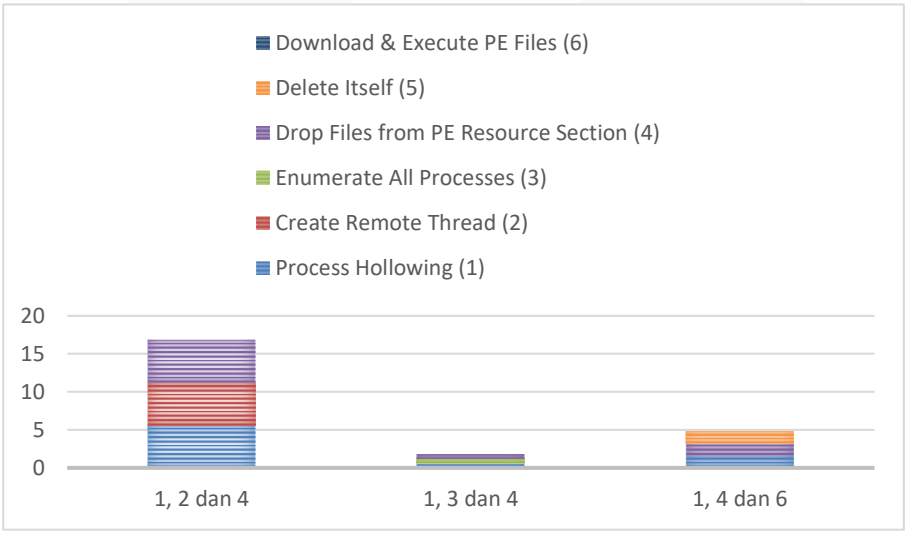


Gambar 6 Grafik 2 Kombinasi Malicious Activity

Terlihat pada gambar 6 terdapat total 28 (dua puluh delapan) *malware* dengan 2 (dua) kombinasi *malicious activity*. Terdapat 14 (empat belas) *malware* yang termasuk dalam kombinasi *process hollowing* dengan *enumerating all process*. Terdapat 8 (delapan) *malware* yang termasuk dalam kombinasi *process hollowing* dengan *drop files from PE resource section*. Terdapat juga 3 (tiga) *malware* yang termasuk dalam kombinasi *create remote thread* dengan *drop files from PE resource section*. Ada 1 (satu) *malware* dengan kombinasi *enumerating all process* dan *drop files from PE resource section*. Dan yang terakhir terdapat 2 (dua) *malware* yang termasuk dalam kombinasi *drop files from PE resource section* dengan *delete itself*.

5.2.2 *Malicious Activity Malware* Dengan 3 Kombinasi

Pada analisis ini terdapat 5 (lima) jenis *malicious activity* yang termasuk dalam kategorisasi 3 (tiga) kombinasi dari sampel *malware* yang digunakan yaitu *create remote thread, enumerating all processes, process hollowing, drop files from* PE *resource section* dan *Download & Execute PE Files.* Berikut merupakan *chart* mengenai detail dari jumlah *malware* yang termasuk dalam hasil kategorisasi.



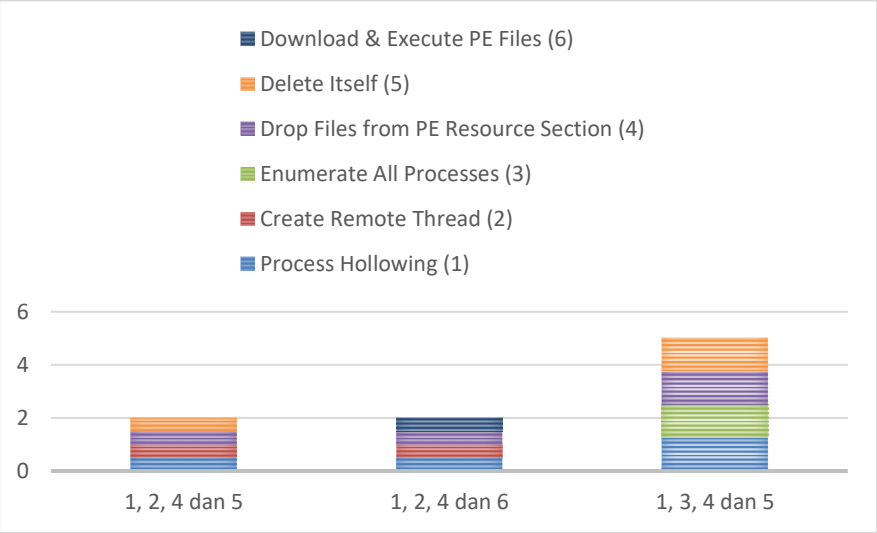
Gambar 7 Grafik 3 Kombinasi Malicious Activity

Pada gambar 7 dapat dilihat terdapat total 24 (dua puluh empat) *malware* dengan 3 (tiga) kombinasi. Terdapat 17 (tujuh belas) *malware* yang termasuk dalam kombinasi *process hollowing, create remote thread* dan *drop files from PE resource section*. Terdapat 2 (dua) *malware* yang termasuk dalam

kombinasi *process hollowing, enumerating all processes* dan *drop files from PE resource section*. Terdapat juga 5 (lima) *malware* yang termasuk dalam kombinasi *process hollowing, drop files from PE resource section* dan *download & execute PE files*.

5.2.3 *Malicious Activity Malware* Dengan 4 Kombinasi

Pada analisis ini terdapat 6 (enam) *malicious activity* yang termasuk dalam kategorisasi 4 (empat) kombinasi dari sampel *malware* yang digunakan yaitu *process hollowing, create remote thread, enumerating all processes, drop files from* PE *resource section, delete itself* dan *download & execute PE files.* Berikut adalah *chart* mengenai jumlah *malware* yang termask dalam kategorisasi diatas.

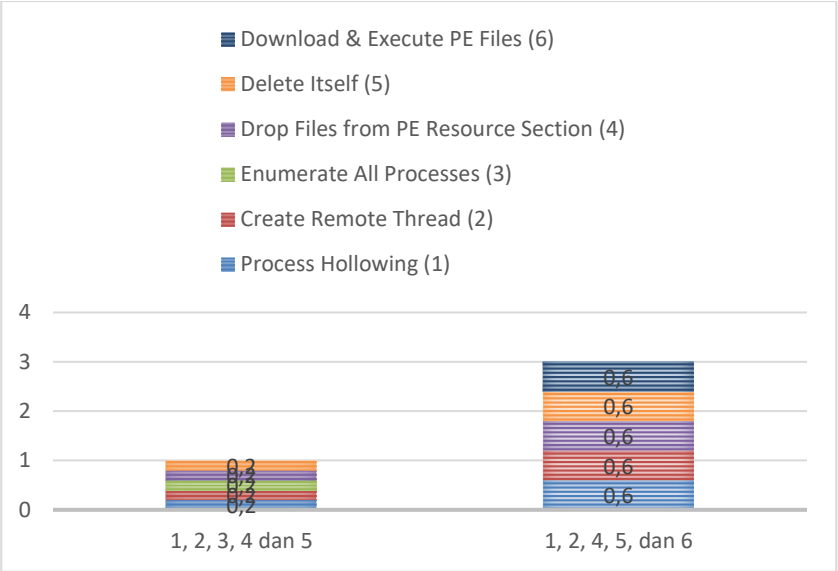


Gambar 8 Grafik 4 Kombinasi Malicious Activity

Pada gambar 8 terlihat terdapat total 9 (sembilan) *malware* dengan 4 (empat) kombinasi. Terdapat 2 (dua) *malware* yang termasuk dalam kombinasi *process hollowing, create remote thread, drop files from PE resource section* dan *delete itself*. Terdapat 2 (dua) *malware* yang termasuk dalam kombinasi *process hollowing, create remote thread, drop files from PE resource section* dan *download & execute PE files*. Dan yang terakhir terdapat 5 (lima) *malware* yang termasuk dalam kombinasi *process hollowing, enumerating all processes, drop files from PE resource section* dan *delete itself*.

5.2.4 *Malicious Activity Malware* Dengan 5 Kombinasi

Pada analisis ini terdapat 6 (enam) jenis *malicious activity* yang termasuk dalam kategorisasi 5 (lima) kombinasi dari sampel *malware* yang digunakan yaitu *process hollowing, create remote thread, enumerating all processes, drop files from* PE *resource section, delete itself* dan *download & execute PE files.* Berikut merupakan *chart* mengenai jumlah *malware* yang termask dalam kategorisasi diatas.



Gambar 9 Grafik 5 Kombinasi Malicious Activity

Terlihat pada gambar 9 terdapat total 4 (empat) *malware* dengan 5 (lima) kombinasi. Terdapat 1 (satu) *malware* yang termasuk dalam kombinasi *process hollowing, create remote thread, enumerating all process, drop files from PE resource section* dan *delete itself*. Dan terdapat 3 (tiga) *malware* yang termasuk dalam kombinasi *process hollowing, create remote thread, drop files from PE resource section, delete itself* dan *download & execute PE files*.

5.3 Analisa Hasil Menggunakan Metode Anomali

Dari hasil kategorisasi beserta analisis yang diperoleh. Terdapat kategorisasi *malware* dengan jumlah 0 (nol) hingga 5 (lima) *malicious activity*. Metode *clustering* adalah proses mengelompokan data ke dalam himpunan bagian yang disebut *cluster*. Objek yang ada didalam cluster memiliki kemiripan karakteristik antara satu dengan yang lain. Pada penelitian ini, pola yang memiliki kemiripan antara satu dengan yang lain adalah kategorisasi dengan jumlah 2(dua) , 3 (tiga), 4 (empat) dan 5 (lima) kombinasi *malicious activity*. Hasil kombinasi tersebut merupakan *cluster data* yang berhasil didapatkan.

Tabel 7. Hasil objek cluster

Cluster	Objek
1 (satu)	2 Kombinasi *malicious activity*
2 (dua)	3 Kombinasi *malicious activity*
3 (tiga)	4 Kombinasi *malicious activity*
4 (empat)	5 Kombinasi *malicious activity*

Pada tabel 7 diatas terdapat 4 objek *cluster* berdasarkan kategorisasi yang berhasil didapatkan. Dengan rincian *cluster* pertama berisi objek dengan 2 (dua) kombinasi *malicious activity*. *Cluster* kedua berisi objek dengan 3 (tiga) kombinasi *malicious activity*. *Cluster* ketiga berisi objek dengan 4 (empat) kombinasi *malicious activity*. *Cluster* ke 4 berisi objek dengan 5 (lima) kombinasi *malicious activity*.

Kategorisasi berdasarkan metode clustering pada penelitian ini memiliki *impact* tinggi (*high*) dengan semakin banyaknya jumlah kombinasi yang dihasilkan. Karena apabila suatu *malware* tergolong dalam lebih dari 1 kategori, serangannya menjadi maksimal sehingga berdampak besar pada komputer yang sudah terinfeksi.

6. Kesimpulan

Dari hasil penelitian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut :

1. Untuk melakukan analisis *malware* pada sistem operasi Windows, dapat dilakukan dengan menggunakan beberapa *environment* yang dijadikan sebagai *sandbox.* Penelitian ini menggunakan sistem operasi Windows pada Vmware sebagai *sandbox* supaya tidak terjadi kesalahan infeksi *malware* pada sistem operasi utama yang digunakan. Utamanya saat melakukan *dynamic analysis* dimana *malware* dianalisis saat sedang dieksekusi.
2. Penentuan kategori *malware* pada penelitian ini dilakukan dengan mencari *data set* untuk mengkategorikan *malware.* Dalam proses analisis *malware,* penelitian ini mencari informasi mengenai API *process,* API *file* dan API *registry.* Dari data yang diperoleh, kategorisasi dapat dilakukan dengan menggunakan *malicious activity data set* dimana parameter tersebut menggunakan parameter API *calls.*
3. Strategi deteksi *malware* pada penelitian ini, dapat dilakukan dengan melihat keterkaitan antar *malware* pada setiap *malicious activity.* Dari data yang diperoleh, akan menghasilkan gambaran kategori dari 170 sampel *malware* yang digunakan. Dari keterkaitan *malware* pada setiap *malicious activity* juga dapat diperoleh *impact* yang akan terjadi apabila *malware* tersebut menginfeksi suatu komputer.
4. Apabila *malware* telah dideteksi *impactnya,* rekomendasi yang diberikan yaitu mengenali kategori suatu *malware* dari dampak terhadap performansi komputer. Sebagai contoh proses, memori, *file* dan system I/O.

Daftar Pustaka:

Dolly Upal ,Vishakha Mehra & Vinod Verma. (2014). *Basic Survey on Malware Analysis, Tools and Techniques*,India.

Dipjoyoti Deka,Nityananda Sarma,Nithin J.Panicker. *Malware Detection Vectors and Analysis Tehniques*: *A Brief Survey*,India.

Ehab M.Alkhateeb, (2017) .*Dynamic Malware Detection using API Similarity*,United Arab Emirates.
Garcia-Molina, Hector; Ullman, JD., & Widom, Jennifer. (2002). *Database systems the complete book,International edition*. New Jersey, Prentice Hall.

Ismahani Ismail, (2010).*Detecting Worms Using Data Mining Technique Learning in the Presence of Class Noise*,Malaysia.

Saputra, R.W.(2016). *A Survey of Cyber Crime in Indonesia*. Indonesia.

Shijo, P.V. & Salim, A. (2015). *Integrated Static and Dynamic Analysis for Malware Detection.*

Sikorski, M. & Honig, A. (2012). *Practical Malware Analysis*. San Francisco, USA.

Ushukabhayar Baldangombo.*A Static Malware Detection System Using Data Mining Methods*,Taiwan.
Yunan Zhang & Qingjia Huang. (2017). Based on Multi-Features and Clustering Ensemble Method for Authomatic Malware Categorization.

Zalavadita, N. & Dr. Sharma, Priyanka. (2007). *A Methodology of Malware Analysis, Tools and Technique for windows platform – RAT Analysis.* Ahmedabad, India.

Zeltser, L. (2014). *What is Malware*. The SANS Institute

Das Malwerk 2016 . http://dasmalwerk.eu/. Diakses pada 05 Desember 2017.

Pektas, A., Acarman, T. (2017). *Malware Classification Based on API Calls and Behaviour Analysis.* Istanbul, Turki.