METODE KLASIFIKASI DAN ANALISIS KARAKTERISTIK MALWARE MENGGUNAKAN KONSEP ONTOLOGI

Abdul Haris Muhammad [(1)], Bambang Sugiantoro [(2)], Ahmad Luthfi [(3)]

Magister Teknik Informatika Universitas Islam Indonesia[(1,3)]

Jl. Kaliurang Km 14,5 Sleman, Yogyakarta

Teknik Informatika UIN Sunan Kalijaga Yogyakarta [(2)]

13917201@students.uii.ac.id

Abstrak

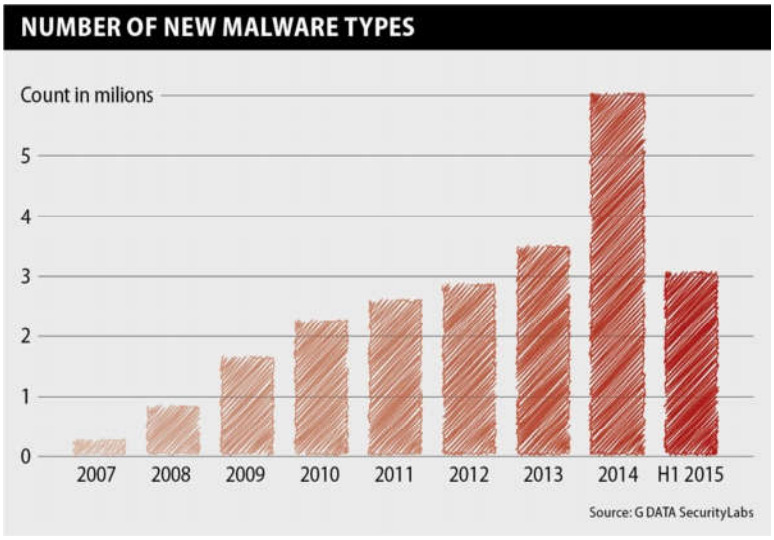
Analisis *malware* membutuhkan keterampilan khusus dalam proses pendeteksian dan pemahaman terhadap cara kerjanya. Program berbahaya atau *malware* menjadi sebuah ancaman atau masalah yang sulit dipecahkan bagi para peneliti karena tidak ada *platform* komputasi atau lingkungan yang kebal terhadap ancaman tersebut. Kompleksitas yang meningkat membuat para peneliti harus bekerja keras dan membutuhkan waktu untuk memahami cara kerja *malware* Terdapat dua teknik dasar yang sering digunakan untuk melakukan analisis *malware* yaitu statis dan dinamis analisis, dan penelitian *malware* yang dilakukan selama ini masih berfokus pada analisis perilaku yang keberhasilan metode tersebut tergantung pada model *malware*. Penggunaan teknik *signature based* sangat tergantung pada perilaku *malware* yang dianalisis, analisis menjadi sulit ketika *ditemukan malware* baru yang menggunakan suatu teknik baru untuk menyulitkan sistem analisis. Berdasarkan uraian fakta yang disampaikan, dianggap perlu dibangun sebuah ontologi dalam melakukan analisis terhadap *malware* sehingga dapat digunakan sebagai pengemabangan, pemetaan pengetahuan serta mengindetifikasi tren, dan pola dalam melakukan analisis *malware.* Pada penelitian ini metode yang diusulkan adalah pengembangan dari metode untuk memetakan karakteristik dan mengklasifikasi jenis *malware*. Pada penelitian ini berfokus kepada ontologi sebagai *knowledge base* dan pembahasannya lebih kepada memetakan karakteristik dan pengklasifikasian jenis *malware*.

Kata Kunci: malware, ontologi, forensik digital.

1. Pendahuluan

Analisis perangkat lunak berbahaya atau *malware* menjadi salah satu bagian penting dalam bidang forensik digital (Masood, 2004). Kemampuan untuk menganalisa perangkat lunak berbahaya bagi *investigator* menjadi tuntutan dalam setiap melakukan investigasi. Hal ini dikarenakan meningkatnya jumlah *malware* serta evolusi dan mampu beradaptasinya terhadap perangkat analisis yang selama ini digunakan. Analisis *malware* membutuhkan keterampilan khusus untuk melakukan pendeteksian dan memahami cara kerja dari *malware* tersebut,

secara garis besar *malware* dibagi atas beberapa kategori yaitu, *worm*, *virus, trojan horse*, *adware,* dan *exploit*. Lima jenis *malware* tersebut merupakan jenis *malware* yang paling sering ditemukan pada analisis *malware* pada umumnya, yang dimana setiap kategori ini mempunyai spesifikasi atau cara kerja yang berbeda (Valli and Brand 2008). Menurut data yang dirilis oleh *G Data Security Labs* pada tahun 2015, terdapat 3,045,722 varian *malware* baru. Program-program berbahaya ini memiliki ancaman keamanan yang dapat berdampak pada kerugian seperti pencurian informasi.



Gambar 1: New Varian Malware (G Data, 2015).

Program berbahaya atau *malware* menjadi sebuah ancaman atau masalah yang sulit bagi para peneliti, tidak ada *platform* komputasi atau lingkungan yang kebal terhadap ancaman tersebut, secara tradisional *malware* dianggap sebagai *virus* atau *worm* yang memiliki fungsi tunggal atau *payload*. Seiring dengan evolusi dan perkembangannya, *malware* dapat menggabungkan beberapa vektor untuk melakukan infeksi, contohnya seperti pada *file hashing* yang dimana *file* tersebut dapat diduplikasi sehingga sangat sulit untuk dilakukan analisis, selanjutnya adalah penggunaan teknik anti-forensik yang digunakan untuk menghambat deteksi, menyamarkan kode sehingga kode tersebut tidak dianggap berbahaya oleh perangkat analisis. (Valli and Brand 2008)

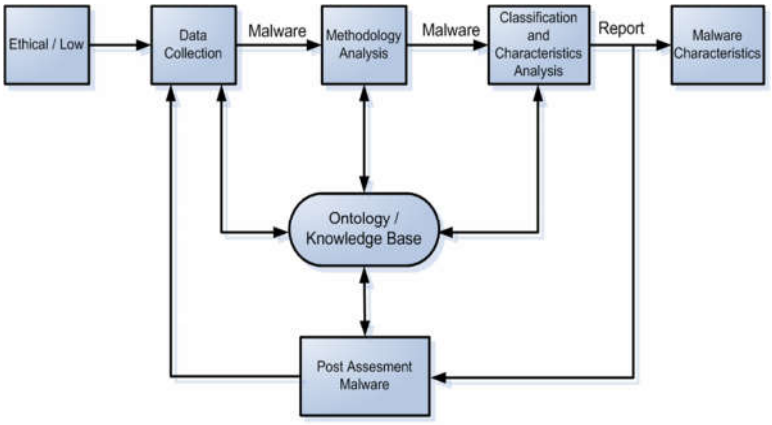
Penelitian *malware* masih berfokus pada analisis perilaku hal ini dibuktikan dengan seringnya analisis *malware* dilakukan menggunakan teknik *signature based*, teknik ini digunakan untuk melakukan analisis berbasis pada *database* perilaku *malware*, akan tetapi dalam beberapa penelitian seperti yang

dilakukan oleh (Chiang 2016) dan (Jasiul, Szpyrka, and Sliwa 2014) yang menjelaskan bahwa tingkat keberhasilan metode yang diterapkan untuk mendeteksi *malware* tergantung pada keadaan model *malware* tersebut. Penggunaan teknik *signature based* sangat tergantung pada perilaku *malware* yang dianalisis, analisis menjadi sulit ketika *malware* yang dianalisis merupakan *malware* baru yang menggunakan teknik kebingungan (*obfuscation*) dan kompresi (*packing*). Dengan fakta yang telah disampaikan, maka perlu adanya sebuah analisis malware dengan memanfaatkan ontologi sebagai *knowledge base* untuk melakukan analisis terhadap cara kerja serta karakteristik *malware* secara detail dan pengklasifikasian jenis *malware* agar memudahkan dalam melakukan analisis serta dapat digunakan sebagai pengemabangan, pemetaan pengetahuan dalam mengindetifikasi tren, dan pola dalam melakukan analisis *malware*, dan bagaimana penerapan ontologi sebagai *knowledge base* dasar dalam melakukan analisis karakteristik *malware,* serta penerapan analisis *malware* menggunakan konsep ontologi

2. Metode Penelitian

2.1 Tahapan Penelitian

Tahapan-tahapan dalam penelitian ini merupakan tahapan penerapan metode yang diusulkan dalam melakukan analisis *malware.*



Gambar 2: Tahapan penelitian

1. *Ethical and Low.* Hukum dan etika merupakan tahap awal dari metode yang diusulkan hal ini untuk memastikan bahwa proses analisis dilakukan dengan benar dan tidak terjadi penyebaran *malware*, yang terjadi pada kasus ekstraksi atau identifikasi perangkat tertentu.

2. *Data Collection.* Pada komponen *data collection malware* dikumpulkan dari berbagai sumber kemudian dipersiapkan untuk dilakukan analisis pada tahapan selanjutnya.
3. *Methodology Analysis.* Pada tahapan ini *malware* dianalisis menggunakan sistem analisis yang telah ada sebelumya dengan maksud untuk mengetahui tipe *malware*
4. *Classification and Characteristic Analysis.* Setelah tipe atau jenis *malware* diketahui kemudian dilakukan pengklasifikasian jenis *malware* dan dilakukan analisis terhadap karakteristik dengan munggunakan *tools* untuk mengetahui setiap karakteristik yang ada pada setiap *malware*.
5. *Post Assesment Malware.* Pada tahapan ini *malware* yang telah dilakukan analisis akan di kembalikan kepada tahap awal, yaitu pada tahapan data collection.
6. *Ontology.* Ontologi merupakan bagian yang paling penting karena merupakan basis pengetahuan yang dimana aktifitas yang dilakukan pada tahapan kedua sampai tahapan ke lima berhubungan langsung dengan ontologi, atau dengan kata lain menjadi satu kesatuan diproses dalam komponen ontologi. Didalam komponen ontologi terdapat aktivitas layer yang berperan untuk melakukan proses analisis diantaranya:
   - *Ontology layer.* Merupakan bagian penting yang berfungsi untuk merepresentasikan bentuk umum dari aturan atau kesepakatan mengenai pengertian dari data atau yang disebut dengan *vocabulary* aturan dari domain.
   - *Logic layer. Logic layer* pada komponen ontologi yang tedapat pada metode diatas merupakan penerapan *intelligent reasoning* dengan data yang bermakna yang mempunyai fungsi untuk pengembilan data yang diinginkan
7. *Malware Characteristics Ontology.* Merupakan hasil dari proses yang telah dilakukan pada tahap sebelumnya, selain dapat dihasilkan dalam bentuk ontologi, hasil dari malware characteristic ini juga dapat disajikan sebagai laporan analisis pada saat persidangan

2.2 Tahapan Pengujian

Pengujian dilakukan dengan dua tahapan yaitu, pengujian tahapan pertama menggunakan sampel *malware* yang didapatkan dari *data collection*

kemudian dilakukan analisis menggunakan beberapa alat bantu sehingga mendapatkan hasil yaitu berupa karakteristik *malware.* Selanjutnya pengujian tahapan kedua dilakukan terhadap ontologi klasifikasi dan karakterisktik *malware* dengan cara, hasil dari pengujian tahap pertama dijelaskan dalam bentuk ontograf secara detail sehingga hasil dari pengujian tahap pertama sama dengan pengujian tahap kedua.

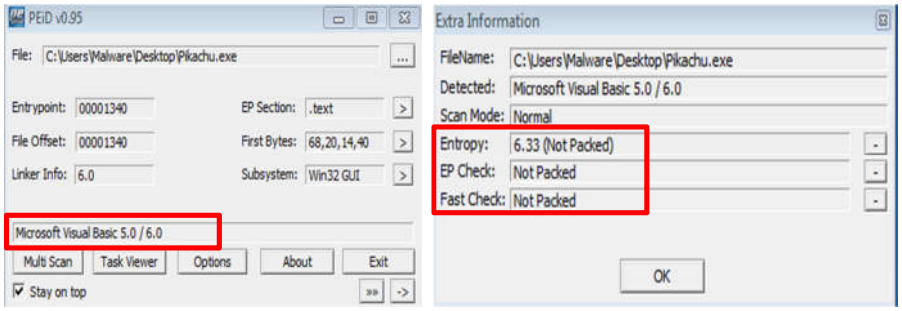
3. Hasil Penelitian

Pada bab ini diuraikan tentang hasil dan pembahasan dalam penyelesaian masalah yang telah diangkat sebagai tema penelitian. Adapun langkah-langkah seluruh tahapan dalam melakukan analisis karakteristik *malware* dan ontologi sebagai *knowledge base* adalah sebagai berikut:

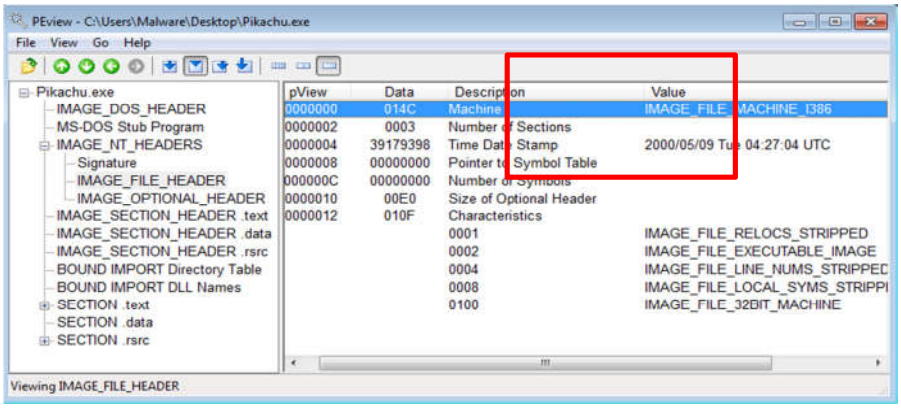
1. **Skema analisis**. Tahapan perancangan skema dibuat untuk menjelaskan proses representasi dari metode yang telah diusulkan sebagai tahapan untuk melakukan analisis karakteristik *malware* dan proses pada ontologi sebagai basis pengetahuan (*knowledge base*).
2. **Pra-analisis**. Sebelum analisis *malware* dilakukan, persiapan membangun sebuah lingkungan dibutuhkan agar *malware* yang dianalisis tetap terisolasi dan tidak menggangu sitem utama.
3. **Analisis**. Tahapan ini merupakan proses untuk menganalisis semua kegiatan yang dilakukan oleh *malware*. Analisis dibagi menjadi empat tahap yaitu analisis packed, analisis header file, analisis dependency modul dan analisis kode. Analisis terdiri dari tiga tahapan sebagai berikut:
    a. **Analisis paket**. Analisis paket dilakukan dengan melihat teknik yang digunakan pada *malware* dan mengetahui bahasa compiler yang digunakan. Pada gambar 3 dapat dilihat bahwa tidak terdapat teknik penyamaran yang digunakan, dapat dijelaskan pula pada email-worm.win32.pikachu menggunakan compiler Microsoft visual basic 6.0
    b. **Analisis header file**. Analisis *Header file* dimaksudkan untuk melihat susunan dari waktu eksekusi *malware* atau worm yang dianalisis. Pada gambar 4 diatas di tampilkan deskripsi dari waktu eksekusi dan target yang akan menjadi sasaran dari *malware* ini
    c. **Analisis modul.** Analisis ini merupakan analisis hirarki dari sebuah susunan modul pada sebuah aplikasi atau pada kasus ini adalah worm. Terlihat jelas pada gambar 5, bahwa worm ini hanya

memliki satu modul utama yaitu MSVBVM60.DLL yang telah dimodifikasi untuk menampilkan pesan error pada layar komputer, dikarenakan MSVBVM60.DLL merupakan sebuah *file library* yang berfungsi untuk menjaga kestabilan sistem utama sebuah komputer.

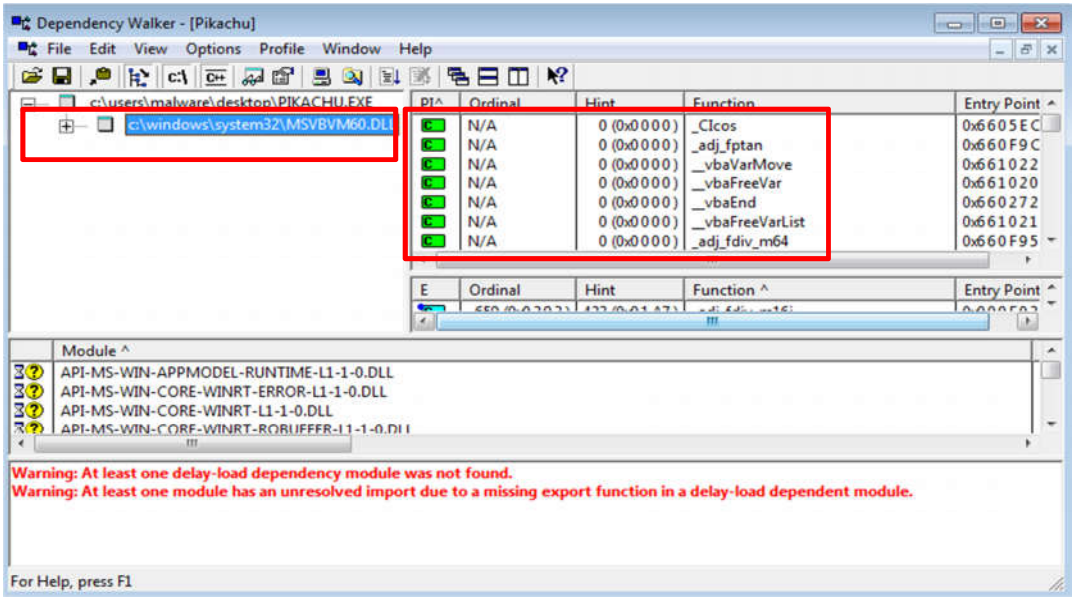
d. **Analisis kode** merupakan analisis yang digunakan untuk melihat isi dari suatu *malware* yang di analisis, fungsi dari analisis kode adalah mencari letak kode yang dianggap berbahaya pada sebuah program, seperti yang diperlihatkan oleh gambar 6 komputer atau sebuah sistem yang sedang aktif.



Gambar 3: Analisis Paket Worm.Win32.Pikachu



Gambar 4: Analisis Header File Worm.Win32.Pikachu



Gambar 5: Analisis Dependensi Modul Worm.Win32.Pikachu

```
text:00402020                   unicode 0, <\PikachuPokemon.exe>,0
text:00402048 aAttachments:                             ; DATA XREF: sub_404040+896↓o
text:00402048                   unicode 0, <Attachments>,0
text:00402060 aSend:                                    ; DATA XREF: sub_404040+8FD↓o
text:00402060                   unicode 0, <Send>,0
text:0040206A                   align 4
text:0040206C a4:
text:0040206C                   unicode 0, <4>,0
text:00402070 aScripting_file:                          ; DATA XREF: sub_404B10+5D↓o
text:00402070                   unicode 0, <Scripting.FileSystemObject>,0
text:004020A6                   align 4
text:004020A8 aGetspecialfold:                          ; DATA XREF: sub_404B10+C6↓o
text:004020A8                                           ; sub_404B10+133↓o
text:004020A8                   unicode 0, <GetSpecialFolder>,0
text:004020CA                   align 4
text:004020CC                   dd 1Eh
text:004020D0 aCAutoexec_bat:                           ; DATA XREF: sub_404B10+158↓o
text:004020D0                   unicode 0, <C:\AUTOEXEC.BAT>,0
text:004020F0                   dd 12h
text:004020F4 a@echoOff:                                ; DATA XREF: sub_404B10+173↓o
text:004020F4                   unicode 0, <@ECHO OFF>,0
text:00402108                   dd 4
text:0040210C dword_40210C      dd 880001h, 8           ; DATA XREF: sub_404B10+191↓o
text:004020D0                   unicode 0, <C:\AUTOEXEC.BAT>,0
text:004020F0                   dd 12h
text:004020F4 a@echoOff:                                ; DATA XREF: sub_404B10+173↓o
text:004020F4                   unicode 0, <@ECHO OFF>,0
text:00402108                   dd 4
text:0040210C dword_40210C      dd 880001h, 8           ; DATA XREF: sub_404B10+191↓o
text:00402114 aDel:                                     ; DATA XREF: sub_404B10+1AF↓o
text:00402114                                           ; sub_404B10+259↓o
text:00402114                   unicode 0, <del >,0
text:0040211E                   align 10h
text:00402120                   dw 0Ah, 0
text:00402124 a_:                                       ; DATA XREF: sub_404B10+1C3↓o
text:00402124                                           ; sub_404B10+26D↓o
text:00402124                   unicode 0, <\*.* >,0
text:00402130                   dd 4
```

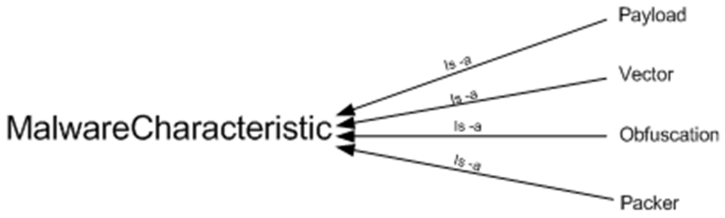
Gambar 6: Analisis Kode Worm.Win32.Pikachu

3.1 Karakteristik Malware

Dalam penelitian ini disajikan ontologi karekteristik *malware* yang dimaksudkan untuk membuat sebuah ontologi yang saling berhubungan antara jenis *malware* dan karakteristik. Ontologi pada penelitian ini dibagi menjadi dua bagian yaitu ontologi klasifikasi *malware* dan ontologi karakteristik *malware.*

3.1.1 Hirarki Karakteristik Malware

Hirarki karakteristik *malware* digunakan untuk membangun sebuah kelas ontologi yang berhubungan langsung dengan kelas karakteristik *malware.* Melalui proses analisis yang sudah dilakukan sebelumnya. Tiga kakateristik *malware* secara umum telah dilakukan analisis pada penelitian sebelumnya dan beberapa penelitian lainnya yaitu *payload, obfuscation* dan *vector*, akan tetapi pada penelitian ini, peneliti menambahkan satu karakteristik baru, dikarenakan pada saat proses analisis, karakteristik tersebut terdapat pada hampir setiap *malware* yang dianalisis yaitu *packer*.



Gambar 7: Karakteristik Malware

3.1.2 Karakteristik *Payload*/Muatan

Payload merupakan efek yang ditimbulkan oleh serangan sebuah *malware* atau virus, ataupun komponen dari *malware* yang menjalankan aktivitas berbahaya. Terlepas dari kecepatan *malware* tersebut menyebar, dan tingkat ancaman *malware* dihitung dengan tingkat kerusakan yang ditimbulkan.

3.1.3 Karakteristik *Vector*

Karakteristik *vector* dari suatu *malware* menjadi sangat penting karena karaktersitik ini mendefenisikan bagaimana suatu *malware* dapat di sebarkan atau di distribusikan.

3.1.4 Karakteristik *Obfuscation*

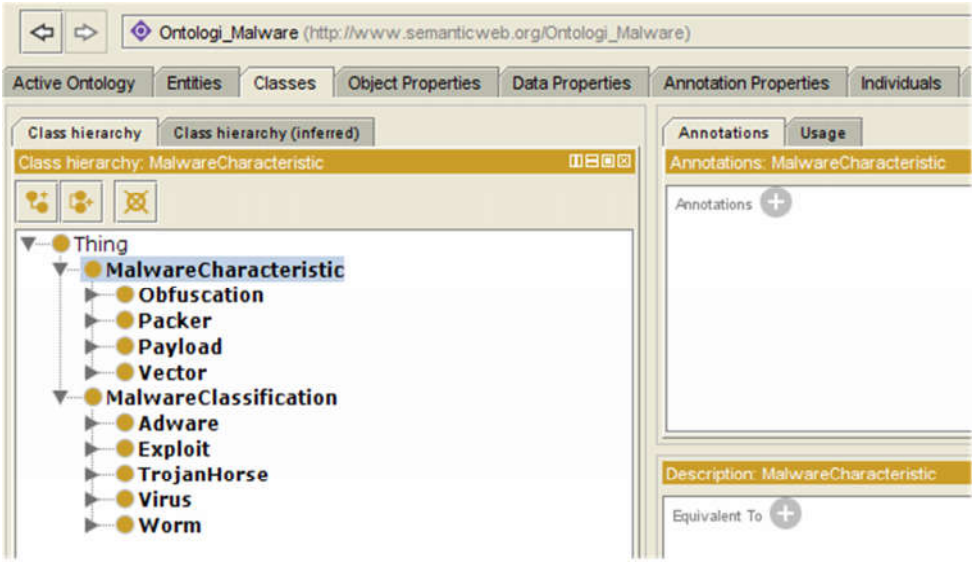
Obfuscation adalah sebuah karakteristik pada *malware* yang fungsinya adalah untuk menghindari proses analisis oleh sistem, ketika *malware* tersebut melakukan infeksi dan menyebar pada sistem

3.1.5 Karakteristik *Packer*

Suatu *malware* dibuat dengan bebagai komponen yang ada didalam *malware* tersebut, tergantung dari tujuan dan fungsinya, akan tetapi terdapat satu komponen yang sering ditemukan yaitu *packer. Packer* merupakan sebuah kompresi pada *malware* yang hasilnya tersimpan pada memori. Hal ini menjadi sulit untuk dianalisis, penggunaan *packer* pada *malware* menjadikan ukuran *malware* menjadi lebih kecil dari ukuran sebenarnya, tujuan dari *packer* adalah untuk menghindari deteksi antivirus.

4. Implementasi Ontologi

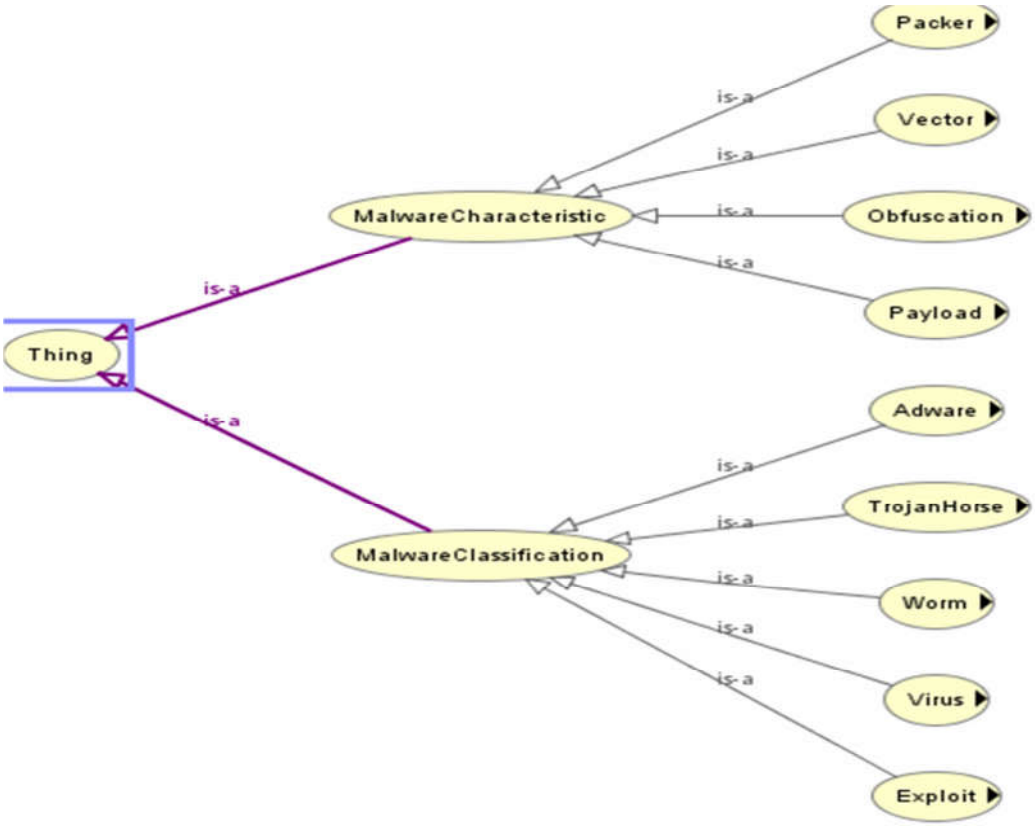
Secara umum dapat dikatakan bahwa ontologi merupakan suatu teori tentang makna dari suatu objek, property dari suatu objek, serta relasi objek tersebut yang mungkin terjadi pada suatu domain pengetahuan (Gruber, 1993). Pada penelitian ini penggunaan aplikasi ontologi menggunakan protégé untuk membuat *class* dan *sub class* klasifikasi dan arakteristik *malware*.



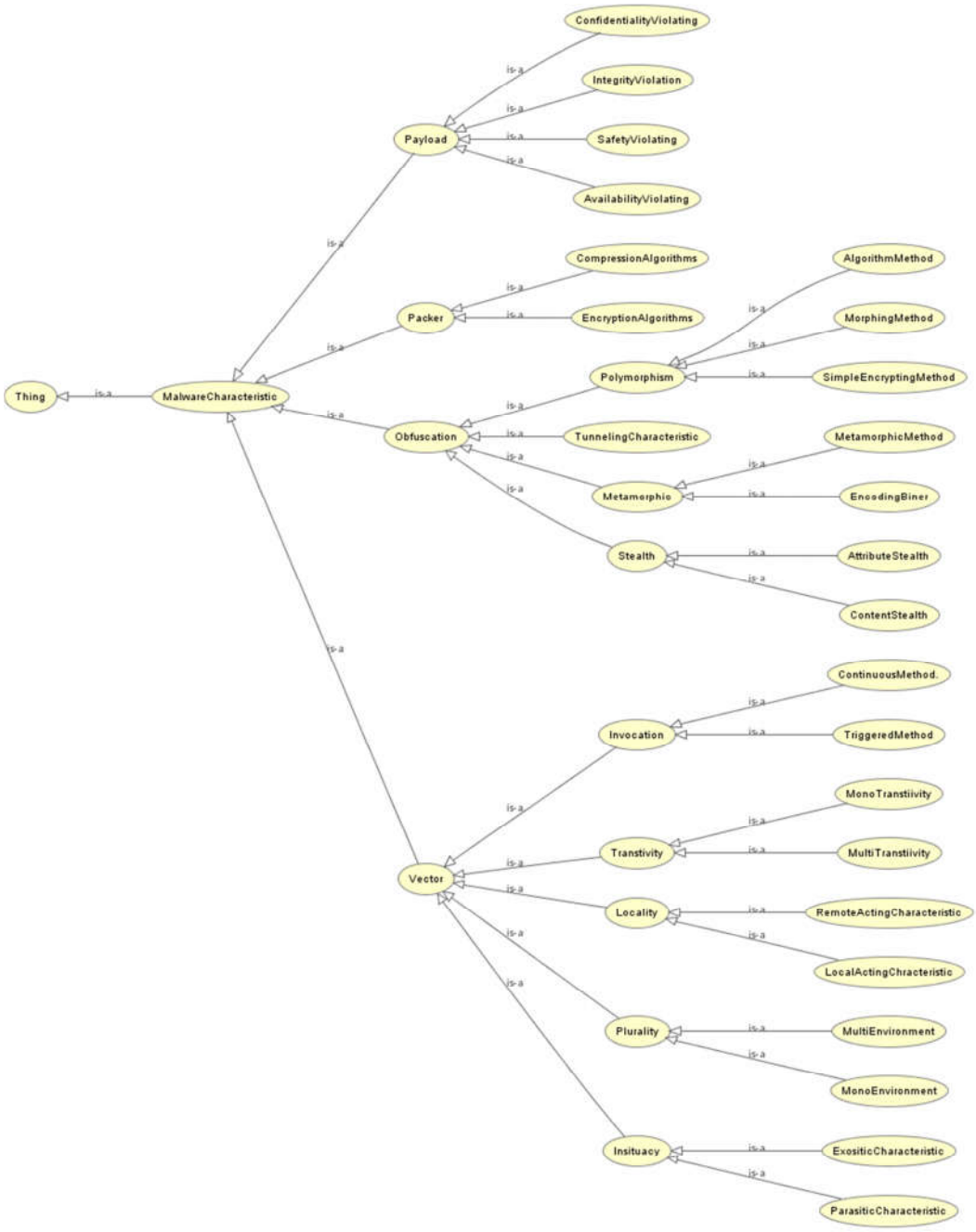
Gambar 8 : Class dan Sub class Analisis Malware

Terdapat dua class ontologi *malware* yaitu Malware Characteristic dan Malware Classification dan dari masing-masing *class* tersebut memiliki *sub class.* Setelah proses pembuatan *class* dan *object property* selesai, maka akan ditampilkan dalam *owl viz* diagram, dimana kedua bagian tersebut representasi dari *class* dan *object property* yang dibuat sebelumnya dengan secara detail sehingga dapat dilihat bagian dari karakteristik *malware* dan klasifikasi *malware* secara lengkap. *Owl viz* dapat dilihat pada gambar 9, 10 dan 11.

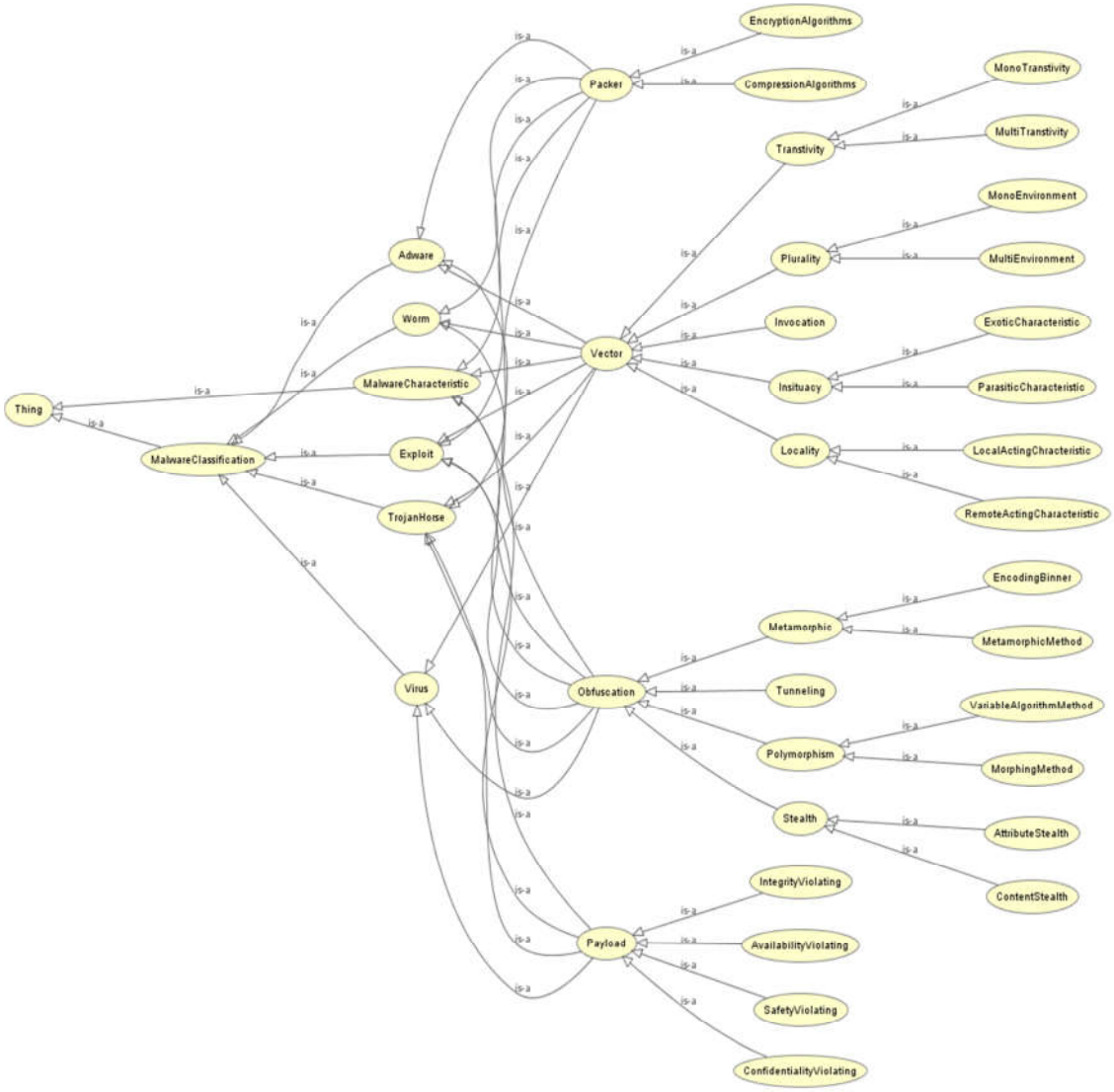
Dari diagram *owl viz* ontologi karakteristik analisis *malware* diatas dapat dijelaskan bahwa terdapat empat kekateristik dasar pada suatu jenis *malware* yaitu, *payload, vector, obfuscation* dan *packer,* empat karakteristik ini selalu terdapat pada hampir setiap jenis *malware* yang diteliti, dengan sub katergori yang berbeda-beda yang memiliki fungsi dan cara kerja masing-masing. Sedangkan pada diagram *owl viz malware* terdapat lima jenis *malware* yaitu *trojan horse*, *virus, exploit*, *worm* dan *adware*, dengan memiliki empat karakter dasar *malware* yang saling berhubungan satu dengan yang lainnya dengan jenis *malware*, sehingga memungkinkan pada satu jenis *malware* memiliki lebih dari satu karakteristik *malware,* sehingga memiliki dampak kerusakan yang sangat besar pada setiap melakukan serangan



Gambar 9: *Owl Viz Diagram Ontologi* Analisis *Malware*



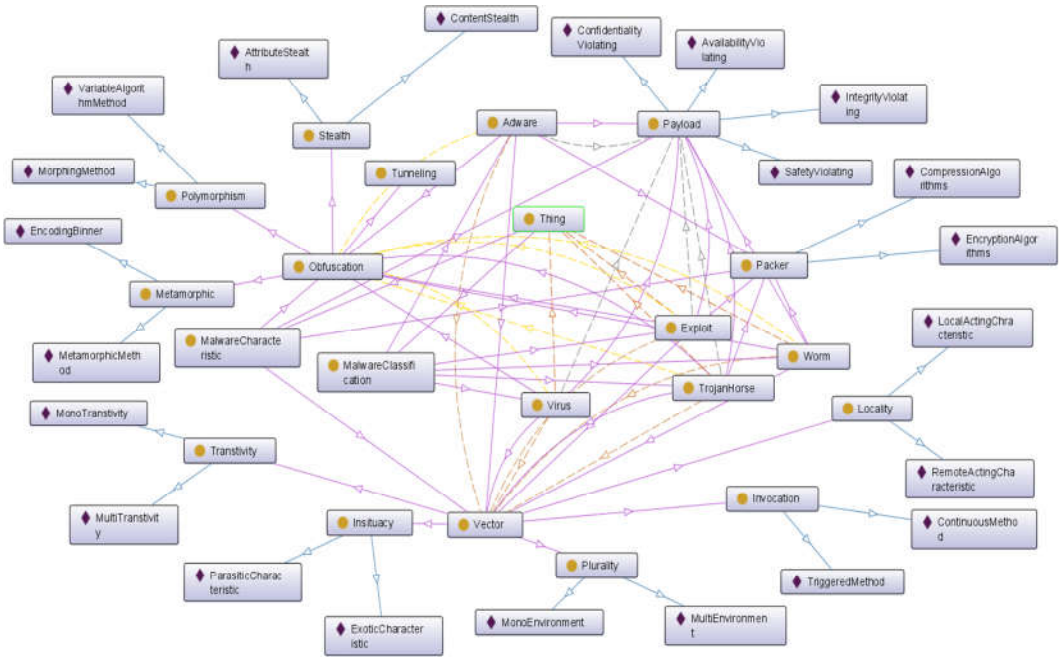
Gambar 10: Owl Viz Diagram Karakteristik Analisis Malware



Gambar 11: Owl Viz Diagram Ontologi Analisis Malware

4.1 Ontograf

Output dari ontologi *malware* dijelaskan lebih lanjut menggunakan ontograf yang dimana fungsinya adalah untuk memvisualisasikan dengan rinci bagian penting dalam ontologi *malware* ini dan juga untuk menggambarkan konsep dasar dari pengetahuan yang terintegrasi.



Gambar 12 : Ontograf Diagram Analisis Malware

Pada diagram ontograf analisis *malware* diatas dapat dijelaskan bahwa setiap jenis *malware* mempunyai lebih dari satu karakteristik yang saling berhubungan dengan karakteristik yang lainnya, hal ini memungkinkan satu jenis *malware* dapat melakakukan serangan atau injeksi secara bersama-sama. dalam ontograf diatas setiap sub kelas memiliki individu, yang dimana individu tersebut merupakan isi yang pada umumnya terdapat ciri-ciri yang membedakan antara satu jenis karakteristik *malware* dengan lainnya, akan tetapi pada kondisi tertentu satu sub kelas pada karakteristik *malware* yang berbeda dapat digabungkan, hal ini dimungkinkan apabila dilihat dari bagaiman cara infeksi dari *malware* tersebut,

5. Penutup

Peta karakteristik yang dibagun dalam melakukan proses klasifikasi *malware* pada penelitian ini dibagi menjadi dua bagian yaitu pemetaan berdasarkan jenis *malware* dalam hal ini terdapat lima jenis *malware* yaitu, *torjan hores*, *virus*, *exploi*t, *worm* dan *adware,* yang pada jenis malware ini terdapat kelas dan sub kelas yang saling berhubungan satu dengan yang lainnya, sedangkan berdasarkan karakteristik terdapat empat karakteristik dasar *malware* yang digunakan dalam melakukan proses pengklasifikasian *malware*, dan pemetaan berdasarkan karakteristik malware yaitu *payload*, atau efek yang ditimbulkan, *vecto*r atau cara *malware* menyebarkan dirinya, *obfuscation* atau

cara *malware* memanipulasi kode untuk tidak dikenali sebagai *malware*, dan *packer* atau cara *malware* untuk menghindar dari deteksi sistem analisis anti malware, kedua bagian ini saling berhubungan satu dengan yang lainnya, sehingga memungkinkan pada satu jenis *malware* memiliki lebih dari satu karakteristik *malware,* sehingga memiliki dampak kerusakan yang sangat besar pada setiap melakukan serangan

Penerapan ontologi sebagai *knowledge base* dasar dalam melakukan analisis karakteristik *malware* sebagai *knowledge base* sangat dibutuhkan dalam melakukan analisis karakteristik *malware*, penggunaan ontologi yang menggunkan pendekatan domain dalam penggalian data, sangat berguna dalam menentukan alur dan cara kerja *malware* yang saling berhubungan antara kelas dan sub kelas serta individu dari *malware,* sehingga dapat memudahkan proses analisis dalam menentukan jenis dan karakteristik *malware*.

Daftar Pustaka

Chiang, Hsiu-sen. 2016. Mobile Malware Behavioral Analysis and Preventive Strategy Using Ontology.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge acquisition, 5, 199–199

Jasiul, Bartosz, Marcin Szpyrka, and Joanna Sliwa. 2014. Detection and Modeling of Cyber Attacks with Petri Nets. : 6602–23.

Masood, S. G. (2004). Malware Analysis for Administrators. Retrieved 17 March, 2007 from http://www.securityfocus.com/infocus/1780

Valli, Craig, and Murray Brand. 2008. The Malware Analysis Body of Knowledge. 2008.