RISTEKDIKTI

seminar nasional informatika 2017

PROSIDING

"e-Defense : Menjaga keamanan data menghadapi cyber warfare untuk memperkokoh kedaulatan Negara Kesatuan Republik Indonesia"

eDefense
seminar nasional informatika 2017

ISSN 1979-2328

Yogyakarta, 25 November 2017

SUSUNAN PANITIA

Penanggung Jawab : Dekan Fakultas Teknik Industri
Pengarah : 1. Wakil Dekan I FTI
2. Wakil Dekan II FTI
Ketua Umum : Ketua Program Studi Teknik Informatika
Wakil Ketua Umum : Sekretaris Program Studi Teknik Informatika

Ketua Pelaksana : Frans Richard Kodong, S.T., M.Kom.

Reviewer :
Assoc. Prof. Dr. Anton Satria Prabuwono, KSU
Dr. Tech. Ahmad Azhari UGM
Dr. Ir. Lukito Edi Nugroho, MT. UGM
Dr. Ashari SN, UGM
Ir. Balza Ahmad, M.Eng. UGM
Joko Siswantoro, Universitas Surabaya
Dr. Djoko Budianto, Atmajaya Yogyakarta
Dr. Slamet, Universitas Muhammadiyah Malang, Indonesia.
Dr. Abdul Kadir, STMIK Kartika Yani
Nuryono Setyo Widodo, S.T., M.T., Universitas Ahmad Dahlan
Dr. Herlina Jayadianti, S.T., M.T., UPN "Veteran" Yogyakarta
Hafsah, S.T., M.T., UPN "Veteran" Yogyakarta
Hidayatulah Himawan, S.T., M.M., M.Eng., UPN "Veteran" Yogyakarta
Bambang Yuwono, S.T., M.T., UPN "Veteran" Yogyakarta

Komite Pelaksana (Informatika UPN) :
Agus Sasmito Aribowo, S.Kom., M.Cs
Budi Santosa, S.Si.,M.T.
Dessyanto Boedi P, S.T.,M.T.
Frans Richard Kodong, S.T., M.Kom
Herry Sofyan, S.T., M.Kom.
Heriyanto, A.Md, S.Kom, M.Cs
Heru Cahya Rustamadji, S.Si.,M.T.
Juwairiah, S.Si., M.T.
Mangaras Yanu Florestiyanto, S.T., M.Eng
Nur Heri Cahyana, S.T.,M.Kom.
Oliver Samuel Simanjuntak, S.Kom, M.Eng
Paryati, S.T.,M.Kom.
Rifki Indra Perwira, S.Kom., M.Eng
Simon Pulung Nugroho, S.T.
Wilis Kaswidjanti, S.Si., M.Kom
Yuli Fauziah, S.T.,M.T.
Budi Cahyono
Pri Wahyu Eko Setiawan
Rahayu Ari Orbani.
Sugeng Rahmadi
Sukardi
Himpunan Mahasiswa Teknik Informatika (HIMATIF)

DAFTAR ISI

HALAMAN JUDUL **i**

KATA PENGANTAR **iii**

SUSUNAN PANITIA **iv**

DAFTAR ISI **v**

1	**SISTEM PAKAR BERBASIS WEB MENGGUNAKAN TEOREMA BAYES (STUDI KASUS PENYAKIT SAAT BANJIR DI CIREBON)**	*Bambang Yuwono, Hidayatulah Himawan, Adi Yusuf*	**1**
2	**SISTEM INFORMASI GEOGRAFIS KOMANDO RAYON MILITER (KORAMIL) DAN KECAMATAN BINAAN KORAMIL DI KOTA YOGYAKARTA**	*Budi Santosa, Sri Rahayu Astari, Wilis Kaswidjanti*	**13**
3	**ANALISIS SISTEM MANAJEMEN KEAMANAN INFORMASI ELECTRONIC SECURITY SYSTEM (ESS) MENGGUNAKAN STANDAR ISO 27001 STUDI KASUS KANTOR PERWAKILAN BANK INDONESIA PROVINSI BALI**	*I Gede Putu Krisna Juliharta, I Made Maha Primananda Budi, I Gusti Agung Lanang Agung Raditya*	**19**
4	**IMPLEMENTASI DAN ANALISA BISNIS RENTAL WEB SYSTEM (SEWALOKA.COM) DENGAN PENDEKATAN *SOFTWARE ARCHITECTURAL PATTERN MODEL-VIEW-CONTROLLER***	*I Putu Satwika, I Made Agus Apriliawan*	**26**
5	**REKAYASA SISTEM PENERIMA BEASISWA MISKIN DENGAN METODE C4.5 DAN ELECTRE**	*Made Henny Aryani, Rukmi Sari Hartati , Ni Wayan Sri Ariyani*	**37**
6	**APLIKASI *SINGLE ACCOUNT* BERBASIS *WEB SERVICE* MENGGUNAKAN *AUTHETICATION LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL* (*LDAP*)**	*Rifki Indra Perwira, Heru Cahya Rustamaji, Hendra Arya Syaputra*	**42**
7	**IMPLEMENTASI MAPPING OTOMATIS DARI DATABASE MYSQL 5.6 KE PROTEGE 4.3 DENGAN TURTLE ONTOLOGY, D2RQ, JENA, DAN NETBEANS 7.4**	*Widiatminingsih, Herlina jayadianti , Heru cahya Rustamaji*	**53**
8	**IMPLEMENTASI SISTEM PENGONTROLAN STOK BAHAN BAKU DAN BARANG JADI PADA GUDANG TEH**	*Wilis Kaswidjanti, Frans Ricard Kodong, Heru Tricahyono*	**64**
9	**KOMPARASI METODE DSS UNTUK MENENTUKAN PRIORITAS PROYEK PEMBANGUNAN DAERAH**	*Maya Marselia, Fathushahib*	**70**
10	**SURVEI PADA PENGGUNAAN TEKNIK DATA MINING PADA BIDANG KESEHATAN DI INDONESIA**	*Siti Khomsah*	**82**
11	**ANALISIS KEAMANAN SISTEM INFORMASI AKADEMIK UIN SUNAN KALIJAGA**	*Aries Firmansyah, Bambang Sugiantoro*	**91**

12	**PERANCANGAN MALWARE LOCAL DAN ANTI-MALWARE MEMANFAATKAN SCRIPT BATCH FILE PADA PLATFORM WINDOWS DENGAN METODE FORWARD CHAIN**	*Frans Richard, Jefri Hutama Arbi*	**100**
13	**REPRESENTASI BUDAYA YOGYAKARTA PADA DESAIN KAOS MENGGUNAKAN TEKNOLOGI *AUGMENTED REALITY* BERBASIS ANDROID**	*OliverSamuel Simanjunt, Hidayatulah Himawan), Reza Raditya Setyo Putra*	**110**

Perancangan Malware Lokal dan Anti-Malware Memanfaatkan Script Batch File Pada Platform Windows Dengan Metode Forward Chain

Frans Richard Kodong, Jefri Hutama Arbi
Jurusan Teknik Informatika
UPN "Veteran" Yogyakarta
frkodong@gmail.com

Abstrak

Malware ialah program biasa yang dapat merubah kebiasaan komputer secara otomatis saat terjangkit malware. Salah satu contoh algoritma yang digunakan oleh komputer ialah foward chain. Foward chain dapat dianalogikan sebagai pemecahan masalah yang terindikasi oleh komputer menggunakan decision tree. Pada saat kondisi awal permasalahan, komputer mencoba melakukan pemecahan masalah jika solusi ditemukan maka masuk kedalam kondisi selesai, jika tidak masuk kedalam pemecahan masalah lainnya hingga solusi ditemukan dan masuk kedalam kondusi selesai. Forward Chain dipilih sebagai metode yang dimplementasikan pada malware dan antimalware sebagai sistem pendukung keputusan untuk melakukan perusakan kedalam data pada komputer. Pada penelitian ini akan dibahas perancangan Malware lokal menggunakan Batch File pada Platform Windows.

***Kata Kunci** : Malware, Anti Malware, Forward Chain, Batch File.*

1. PENDAHULUAN

Belajar dari pengalaman Y2K yang terjadi pada awal tahun 2000, tidak sedikit *programmer* komputer yang memanfaatkan celah yang sejenis untuk membuat program dan mendatangkan keuntungan padanya namun merupakan kerugian pada orang lain. *Malware* ialah program biasa yang dapat merubah kebiasaan komputer secara otomatis saat terjangkit *malware*. Salah satu contoh algoritma yang digunakan oleh komputer ialah *foward chain*. *Foward chain* dapat dianalogikan sebagai pemecahan masalah yang terindikasi oleh komputer menggunakan *decision tree*. *Malware* dapat merusak algoritma tersebut dengan cara mengganti salah satu pemecahan masalah nya, sehingga komputer mengintepretasikannya seakan solusi telah ditemukan dan memasukkannya kedalam kondisi selesai. Kondisi selesai tersebut akan melakukan tugas yang bersifat salah atau bahkan destruktif pada komputer. Perlunya pemahaaman tentang *malware* akan membuat pengguna waspada sehingga komputer dapat terhindar dari *malware* dan melakukan tugasnya dengan baik.

Pembuatan *malware* dapat menggunakan berbagai macam bahasa pemrograman. Pada umumnya programmer *malware* menentukan spesifikasi targetnya terlebih dahulu, seperti sistem operasi, celah yang akan dimanfaatkan, serta nama ekstensi program yang dapat dengan mudah berjalan pada platform target. Sistem operasi windows satah satu contoh platform yang sering terkena *malware*. Umum nya pengguna sistem operasi Windows, terutama di negara indonesia lebih banyak dari pada sistem operasi lainnya. Hal tersebut menjadi salah satu alasan *programmer malware* lebih menargetkan ke sistem operasi windows, dikarenakan umumnya pengguna platform windows. Fitur *Command Prompt* (CMD) pada windows dapat dijadikan celah dimulainya pembuatan *malware.*

Malware dan *antimalware* merupakan entitas yang dapat dianalogikan menjadi dua peran yang saling bersinggungan, disatu sisi *malware* merupakan peran antagonis dan *antimalware* merupakan peran protagonis. Kedua entitas tersebut dapat menghasilkan keuntungan yang berlimpah, apabila saling bekerjasama. Sebuah perusahaan *malware* dapat melakukan kerja sama dengan programmer *malware* atau membuat *malware* sendiri yang hanya dapat diantisipasi dengan anti-*malware* khusus, namun hal tersebut dapat melanggar didalam implementasi kode etik perusahaan penyedia jasa *antimalware* sperti yang diungkapkan oleh Cristian Mairoll, CEO dari Emsisoft . Banyak rumor yang beredar didalam masyarakat bahwa *malware* yang tersebar di komputer merupakan rekayasa dari penyedia *antimalware* untuk meningkatkan keuntungan produsen *antimalware*, namun belum ada pemberitaan atau laporan yang valid mengenai isu tersebut.

Studi kasus pada penyerangan *ransomware* WannaCry yang beredar pada awal tahun 2017 contohnya. Pada saat *ransomware* tersebut menyerang komputer milik instansi pemerintah atau pun layanan masyarakat, belum ada *antimalware* yang dapat menaggulangi *ransomware* yang beredar. Pemilik komputer yang terserang *ransomware* dengan terpaksa membayar sejumlah uang kepada pembuat *ransomware* hanya untuk menyelamatkan data yang terdapat pada komputer mereka. Maraknya *ransomware* WannaCry yang menyerang komputer dan meenkripsi data didalam komputer membuat sejumlah organisasi dan perusahaan menjadi waspada akan *ransomware* tersebut. Kejadian tersebut menjadi salah satu latar belakang dalam penelitian mengenai *malware* lokal beserta *antimalware* menggunakan metode forward chain sebagai bahan pembelajaran

untuk mengatasi serta mengantisipasi terjadinya serangan *malware*. Memanfaatkan *file* script batch sebagai bahasa pemrograman yang cukup mudah dimengerti dan efektif untuk melakukan penyerangan *malware* dan antisipasi oleh anti *malware*. Pada *antimalware* dilengkapi dengan pemrograman Visual Basic Application sebagai tampilan pendukungnya agar memudahkan pengguna untuk menjalankannya.

2. TINJAUAN PUSTAKA

2.1 Batch file

Batch file mulai diperkenalkan pada komputer generasi kedua. Pada awalnya *batch file* merupakan *file* yang berisi dari antrian program data yang akan dijalankan karena pada saat komputer generasi kedua belum bisa melakukan *multitasking* untuk mengeksekusi program. Bermula dari *Disk Operating System* (DOS), pengguna diharuskan untuk menyusun tugas yang akan dikerjakan oleh komputer secara manual terlebih dahulu. Teknik penyusunan tugas secara manual menyebabkan tingginya waktu menganggur komputer. Menanggulangi hal tersebut, dikembangkan teknik pengurutan tugas secara otomatis serta mentransfer kontrol tugas-tugas yang telah tersusun sercara otomatis. Pada komputer modern, *batch file* dapat dimanfaatkan untuk membuat urutan aktivitas yang akan dieksekusi oleh komputer.

2.2 Penjadwalan Proses

Penjadwalan ialah kumpulan mekanisme serta kebijakan dalam sistem operasi yang berhubungan dengan antrian kegiatan yang akan dieksekusi oleh komputer. Penjadwalan proses merupakan basis dari sistem operasi *multiprogramming.* Tujuan dari penjadwalan proses merupakanmengoptimalkan kinerja dari sistem operasi yang memenuhi kriteria Adil , Efisiensi, Waktu Tanggap, Waktu Kembali dan *Troughput* (Milenkovic, 1992; Tanenbaum, 1992).

2.3 Tipe Penjadwal (*scheduler*)

Terdapat tiga tipe penjadwal yang masing-masingnya memiliki rentang waktu atau kriteria yang berbeda-beda antara lain penjadwalan jangka pendek, penjadwalan jangka menengah, penjadwalan jangka panjang.

2.4 Command Line Interface (CLI)

Command line Interface atau *Command Shell* merukpakan fitur yang tersedia pada sistem operasi Windows yang berguna sebagai pengganti *Graphic User Interface* (GUI). CLI pada Windows 7 atau setelahnya memiliki dua fitur CLI yaitu Command Prompt dan Powershell. Kedua aplikasi tersebut merupakan aplikasi yang menyediakan komunikasi langsung antara pengguna dengan sistem operasi. CLI berjalan dengan tampilan *non-graphical* yang berarti hanya menggnuakan *character base aplication.* CLI dapat mengeksekusi program dan menampilkan hasilnya pada layar dalam bentuk karakter layaknya pada MS-DOS.

2.5 Command Prompt (CMD)

CMD merupakan salah aplikasi berbasis CLI yang mengeksekusi perintah dengan baris kode yang dimasukkan oleh pengguna. CMD akan mengintepretasikan masukkan informasi dan menampilkannya dengan bahasa yang dapat dimengerti oleh sistem operasi. Dalam menjalankan CMD, jumlah *string* maksimal yang dapat dieksekusi sepanjang 2047 karakter ("Command prompt (Cmd. exe) command-line string limitation," , 2015).

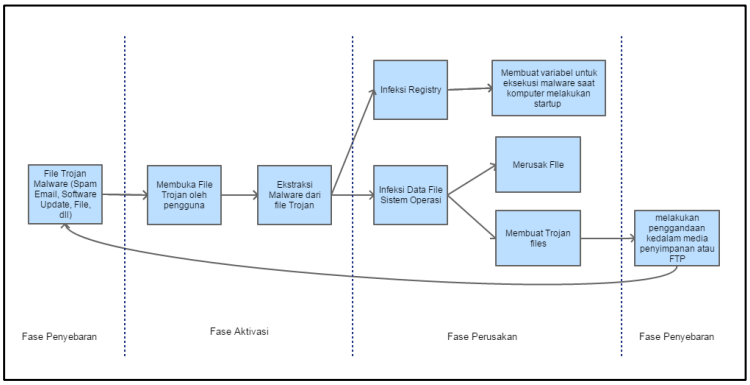
2.6 Bat To Exe Converter

Bat to Exe Converter merupakan aplikasi berbasis Windows yang berguna untuk merubah ekstensi batch *file* (.bat) menjadi executeable (.exe). Hasil dari aplikasi bat to exe converter berupa executable, sehingga baris program *batch file* berubah menjadi *source code* asing untuk meminimalisir plagiasi *source code*-nya.

2.7 Malware

malware merupakan akronim dari *Malicious Software* yang berarti perangkat lunak yang memiliki sifat perusak. Sebuah perangkat lunak yang dapat berjalan didalam sistem komputer dengan tujuan menggumpulkan informasi didalamnya untuk merusak sistem ataupun aplikasi yang ada didalamnya ("Batch *file* Help," , 2017).
Pada umumnya *malware* memiliki tiga tahap dalam melakukan penyerangan yaitu tahap penyebaran, tahap aktivasi, dan tahap perusakan. Tahap penyebaran *malware* memiliki banyak cara, salah satunya meggunakan *trojan file* sebagai media pembawa *malware* kedalam komputer. Tahap aktivasi *malware* dapat memanfaatkan pengguna untuk membuka *trojan file* pada komputernya. Pada tahap perusakan, *malware* akan menambahkan variabel kedalam register sistem operasi yang bertujuan untuk menambahkan aktifitas eksekusi *malware* saat komputer menyala. Secara bersamaan *malware* akan melakukan perusakan tertentu pada dokumen atau data yang ada dalam komputer. *malware* juga akan melakukan infeksi kedalam data atau dokumen untuk membuat *file trojan* baru dan menggandakan dirinya kedalam media penyimpanan atau *file Transfer Protocol* (FTP) yang

nantinya akan menyebarkan dirinya kembali. *malware* memiliki berbagai jenis, antara lain: *Bacteria, Logic Bom, Trapdoor, Trojan, Virus, Adware, Spyware, Worm dan Ransomware* (Comodo *website admin*, 2014).



Gambar 1. Arsitektur *malware* Pada Umumnya

2.8 Perbedaan *Malware* dengan *Virus*

Malware merupakan istilah untuk program yang bertujuan untuk merusak, mencuri atau hanya sekedar mengumpulkan informasi tanpa adanya izin dari pemilik data. *Virus* ialah salah satu kategori dari *malware* yang bersifat menginfeksi aplikasi atau *file* tertentu. *Virus* tidak akan menduplikasi ditinya secara otomatis, akan tetapi saat pengguna mencoba untuk menjalankan program yang terinfeksi *virus*, maka *virus* akan aktif dan melakukan eksekusi perusakan serta penyebaran, umumnya melalui media penyimpanan. *Malware* memiliki beragam cara untuk menginfeksi seperti *trojan* yang mengikatkan diri pada program lainnya, *worm* dengan cara menduplikasi diri secara otomatis dan merusak *file* sistem didalamnya, *spyware* yang secara sengaja diinstal oleh suatu pihak untuk kepentingannya, bahkan menggabungkan cara penyebaran *trojan* dengan fungsi *spyware.*

2.9 Anti-Malware dan Anti-Virus

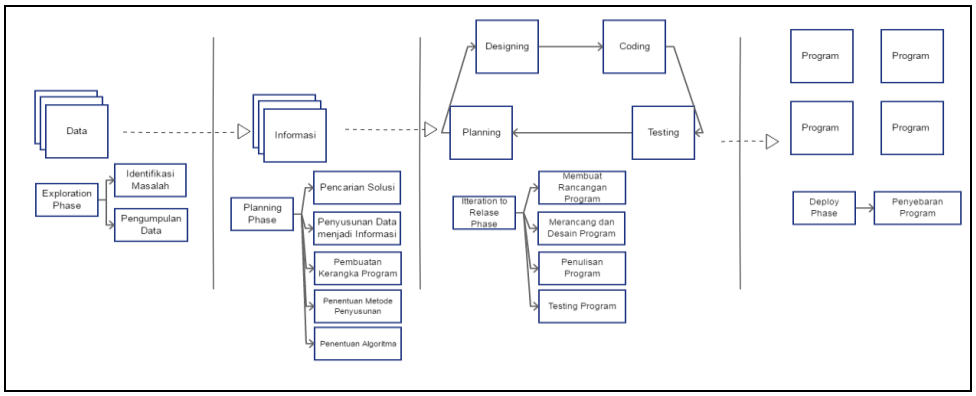
Anti-*malware* merupakan solusi dari infeksi yang disebabkan oleh *malware*. Umumnya produsen anti-*malware* memiliki data eksperimen untuk menanggulangi infeksi berbagai macam *malware* dan membuat program untuk membalikkan efek yang telah hasilkan oleh *malware* dengan *library* data eksperimen tersebut. Tidak jauh berbeda dengan anti-*malware*, anti-*virus* juga memiliki sistem kerja yang sama dengan anti-*malware*. Perbedaan terletak pada cara kerja kedua program tersebut untuk menanggulangi efek yang dihasilkan dari *malware* ataupun *virus*. Fase quarantine sering muncul dalam mengoperasikan anti-*virus* ataupun anti-*malware*. Quarantine atau karantina ialah fase dimana *malware* dan *virus* tidak dapat menimbulkan efek destruktif. Pada umumnya karntina ialah memindahkan *file* yang telah terinfeksi kedalam suatu direktori agar tidak dapat melakukan efek destruktif lebih besar, namun tidak menghapus *file* yang terinfeksi dikarenakan kemungkinan *file* yang terinfeksi merupakan *file* yang cukup penting bagi pengguna, sehingga tidak akan dihapus dari system ("Difference between Antivirus and Anti-Malware," 2017).

2.10 Metode forward Chain

Penelitian ini menggunakan metode foward chain yang diimplementasikan pada *malware* lokal beserta anti-*malware*nya dan bekerja pada sistem operasi Windows. Basis sistem *malware* memanfaatkan batch *file* yang berisikan baris program bersifat destruktif. Anti-*malware* juga menggunakan batch *file* yang nantinya akan dikonversi kedalam extensi executeable *file*s bersama dengan *virus*nya. Aplikasi anti-*malware* bersifat one-click program yang berarti program akan berjalan hanya dengan me-click aplikasinya. Hasil yang dikeluarkan oleh anti*malware* berupa notifikasi pop-up keterangan ada atau tidak adanya *malware* pada komputer, serta merecovery *file* yang telah dirusak oleh *malware*.

3. METODE PENELITIAN

Objek penelitian dalam tugas akhir ini ialah *malware* yang dikembangkan berdasarkan script batch yang telah ada dengan metodologi yang digunakan dalam penelitian secara kuantitatif dan *field study, meliputi* Observasi, Studi Pustaka dan, Metode Pengembangan Sistem *Extreme Programming* (XP).



Gambar 2. Bagan metode Pengembangan Sistem dengan *Extreme Programing*

Extreme Programming (XP), merupakan salah satu metodologi pengembangan sistem yang ada. Metode XP ialah model pengembangan perangkat lunak yang menyederhanakan proses pengembangan sistem menjadi lebih efisien, adaptif, dan fleksibel. Diperkenalkan oleh Kent Back pada sekitar tahun 1999 - 2000 . Metode ini dibagi menjadi 5 tahap antara lain Tahap Pencarian, Tahap Perencanaan, Tahap Pembangunan dan Pelepasan Sistem, *Deploy Phase.*

4. HASIL DAN PEMBAHASAN

4.1. Analisis

Pada bagian analisis dan perancangan terjadi dua tahap yang dilakukan yaitu tahap eksplorasi dan tahap perencanaan. Tahap pertama dalam menggunakan metode XP ialah tahap eksplorasi. Tahap ekplorasi merupakan tahap pengumpulan informasi yang nantinya akan digunakan dalam pengembangan *program*. Tahap eksplorasi terdiri dari analisis sistem, observasi, analisi kebutuhan dalam membangun *malware* dan anti-*malware*.

Tahap selanjutnya merupakan tahap perencanaan. Tahap perencanaan ialah tahap penyusunan informasi yang telah dikumpulkan menjadi suatu struktur rancangan program *malware* dan anti-*malware*. Tahapan perencanaan meliputi perancangan *parameter* pada metode *forward chain*, mekanisme penyerangan *malware*, mekanisme pencarian anti-*malware*, penyusunan proses penyerangan *malware*, penyusunan proses pertahanan anti-*malware*, penentuan kriteria penyerangan *malware*, perancangan *malware*, perancangan anti-*malware*, serta *Unified Modeling Language* (UML) sebagai *flowchart* dalam pengembangan rekayasa perangkat lunak.

4.2. Analisa Kebutuhan Sistem

Data yang diperlukan dalam penelitian "Implementasi *malware* Lokal dan Anti-*malware* Memanfaatkan *Scrip*t *Batch file* Pada Platform Windows Menggunakan Metode *Forward Chain*" ialah data yang diperoleh dari berbagai literatur yang tersedia pada *World Wide Web*, buku ilmu komputer, serta pedoman tentang platform Windows sebagai data sekunder. Data sekunder merupakan data yang diperoleh melalui perantara atau secara tidak langsung.

4.3. Observasi

Pengamatan terhadap script batch yang telah dikembangkan oleh programer lain yang berlisensi *open source.* Implementasi pengembangan *malware* dan anti-*malware* memerlukan data daftar perintah yang dapat berjalan pada CLI Windows atau CMD. Pengetahuan tentang path direktori yang tertanam dalam Windows merupakan kebutuhan pokok dalam pengerjaan penelitian agar mempermudah *file malware* dan anti-*malware* dalam melakukan penelusuran kedalam direktori-direktori yang ada pada platform Windows.

Registry Editor atau biasa disebut Regedit merupakan database hirarki yang digunakan untuk mengatur informasi yang dibutuhkan oleh sistem operasi windows. Pada gambar 3 terlihat beberapa tampilan menyerupai folder yang disebut key. Umumnya jumlah key pada komputer memiliki lima key yaitu hkey local machine (HKLM), hkey current user (HKCU), hkey users (HKU), hkey classes root (HKCR), dan hkey current config (HKCC).

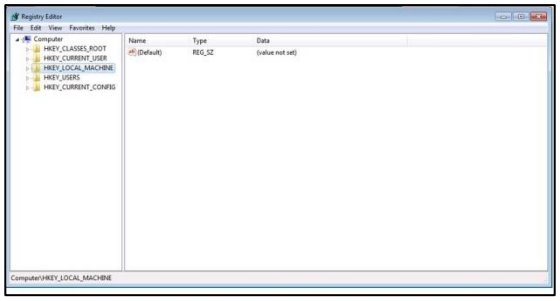
4.4. Komponen *malware*

Pada pengembangan *malware* ini, terdapat komponen penyusun *malware* berupa distribusi *malware*, pencarian direktori *malware*, efek yang ditimbulkan oleh *malware*, serta backup *file malware*. Keempat komponen tersebut dibentuk dalam *file* yang terpisah untuk meminimalisir terjadinya error yang disebabkan oleh variabelnya.

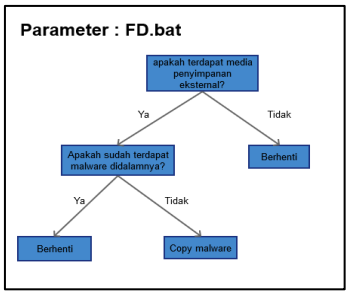
4.5 Perancangan parameter *malware*

Parameter yang akan terpasang pada *malware* merupakan parameter-parameter yang digunakan pada proses sebagai acuan kondisi yang akan ditetapkan oleh *malware*. Setiap komponen *malware* akan memiliki parameter

masing masing untuk menentukan langkah yang akan dilakukan oleh *malware* untuk menghadapi kondisi tertentu. Komponen distribusi *malware* merupakan komponen yang berperan sebagai program menggandakan *malware*. Jika terdapat *malware* yang sama pada media penyimpanan tersebut, maka *malware* tidak akan menggandakan dirinya lagi. Jika suatu media penyimpanan yang mengandung *malware* terhubung dengan komputer, perlunya interaksi dari pengguna untuk mengaktifkan fungsi *malware*.



Gambar 3. Aplikasi Registry Editor



Gambar 4. Parameter FD.bat

Pencarian direktori *malware* ialah fungsi untuk menempatkan *malware* pada komputer. Direktori yang digunakan ialah direktori system32 yang terdapat dibawah diterktori Windows. Fungsi PrimarySearch.bat untuk memastikan setiap *malware* yang telah terpasang tetap ada pada direktorinya. Apabila ditemukan salah satu bagian dari *malware* tidak pada direktorinya, PrimarySearch akan menggandakan ulang bagian yang hilang kedalam direktori tersebut. Efek yang dihasilkan oleh *malware* ialah, mengubah konten pada *file* jenis .doc ataupun .docx sebanyak empat puluh baris.

Apabila terdapat Shortcut Firefox pada tampilan desktop pengguna. *malware* akan menggantikannya dengan *virus* yang memiliki tampilan menyerupai shortcut Firefox, namun efek yang akan ditimbukannya ialah membuka halaman Fire fox dan mengeluarkan pesan dalam halaman cmd.exe secara dengan tujuan untuk mengganggu pengguna dalam mengoperasikan perangkat lunak Firefox. Fungsi ini dinamakan dengan *file* FirePhon.bat.

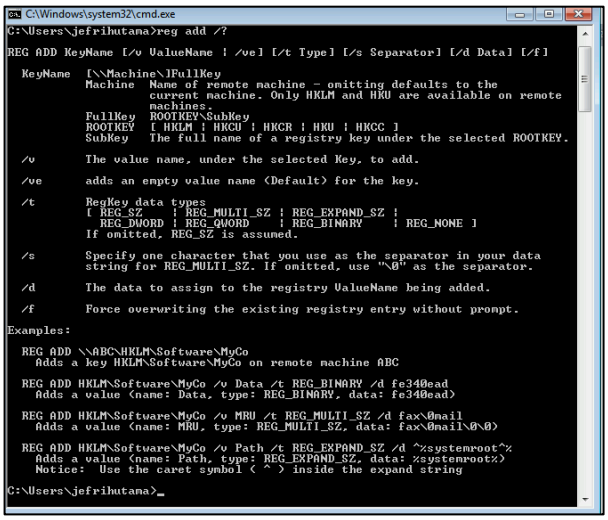
4.6. Mekanisme Pemasangan *malware*

Penyerangan *malware* diawali dengan instalasi *malware* pad komputer yang dilakukan oleh pengguna. Saat *malware* telah terpasang, *malware* akan menjalankan fungsi pencarian direktori untuk menempatkan *malware* tersebut. Fungsi *User Access Control* (UAC) pada komputer akan dinonaktifkan agar *malware* tidak menanyakan otoritas admin dari komputer, sehingga *malware* tidak menimbulkan kecurigaaan pada pengguna. Setalah UAC dinonkatifkan, *malware* akan melakukan reaksi berantai kepada fungsi *malware* lainnya yang merupakan peran otomatisasi dari *malware*. Reaksi berantai yang akan dilewati pada mekanisme pemasangan *malware* ialah, penggandaan *malware* kedalam direktori yang dituju, penggandaan *file backup malware*, dan menjalankan proses serangan *malware*.

4.7. Proses dan Kriteria Penyerangan *malware*

Kriteria penyerangan *malware* dapat dibagi menjadi dua kategori yaitu, kriteria sistem operasi komputer dan kriteria target.

Sistem operasi yang menjadi basis agar *malware* dapat berjalan ialah sistem operasi Windows. *malware* memanfaatkan bebebrapa fungsi yang ada pada sistem operasi Windows untuk mengeksekusi program yang ada. *malware* juga memiliki kriteria dalam pencarian target yang akan diserang diantaranya *file* berekstensi doc atau docx yang terdapat dalam partisi penyimpanan C, *shortcut browser* Firefox pada halaman desktop, koneksi jaringan.



Gambar 5. Perintah REG ADD pada Program Command Prompt

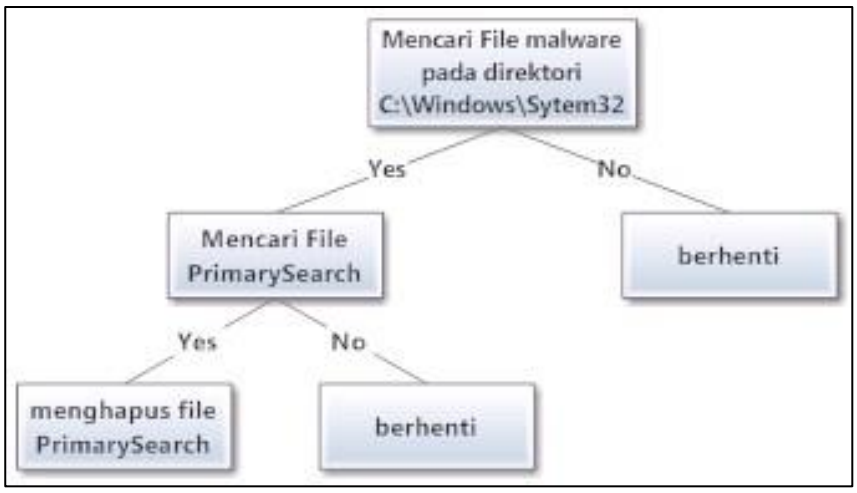
Pada kriteria target, *malware* akan melakukan pencarian terhadap *file* berekstensi doc dan docx. Kedua ekstensi tersebut menandakan bahwa *file* tersebut merupakan *file* pengolah kata. Mayoritas pengguna komputer menggunakan jenis *file* tersebut, oleh karenanya ekstensi doc dan docx dipilih sebagai target *malware*. Pencarian target meliputi seluruh partisi penyimpanan drive C dan jika tersedia partisi penyimpanan drive D, maka akan menjadikannya sebagai target sekunder. Kriteria target juga mencari kemungkinan penggua menggunakan *browser* Mozilla Firefox sebagai salah satu media pengaktifan beberapa fungsi dari *malware*. Untuk pencarian *browser* Firefox, lingkup pencarian meliputi direktori program *file*s (x86) atau Firefox utnuk 32-bit.

4.8. Komponen Anti-*malware*

Anti-*malware* ialah program khusus yang dibangun untuk menaggulangi efek yang disebabkan oleh *malware*. Pada dasarnya, komponen anti-*malware* merupakan pembalikan dari komponen *malware* yang dibangun. Anti-*malware* memiliki komponen yang meliputi *recovery file* doc dan docx yang semula telah dirubah kontennya oleh *malware*.

4.9. Perancangan parameter Anti-*malware*

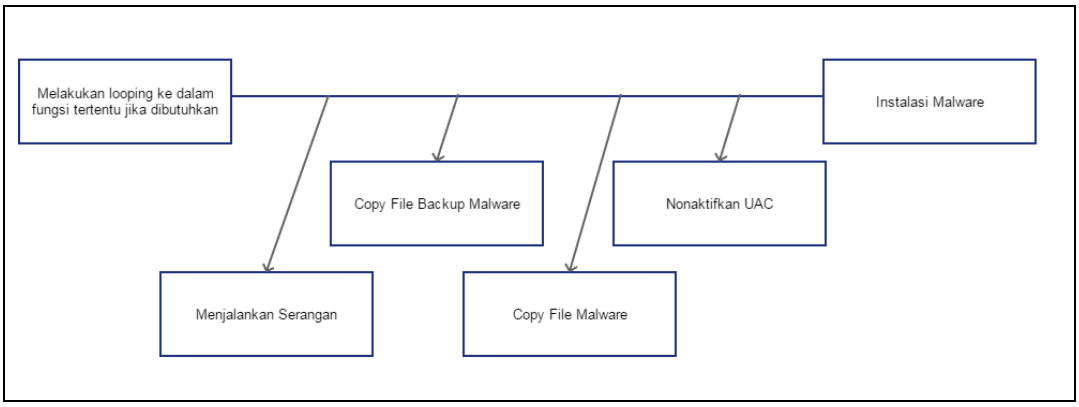
Parameter yang terdapat pada fungsi anti-*malware* ialah rangkaian kebalikan dari parameter *malware* itu sendiri. Parameter anti-*malware* meliputi parameter pencarian *file*, pengembalian *enviroment* pada Registry Editor, dan penghapusan *file malware* yang terdapat pada komputer. Berbekal dengan parameter *malware* yang telah dibangun, anti-*malware* melakukan proses *recovery* dengan parameter *malware* dan melakukan aksi yang merupakan kebalikan dari parameter *malware* itu sendiri.



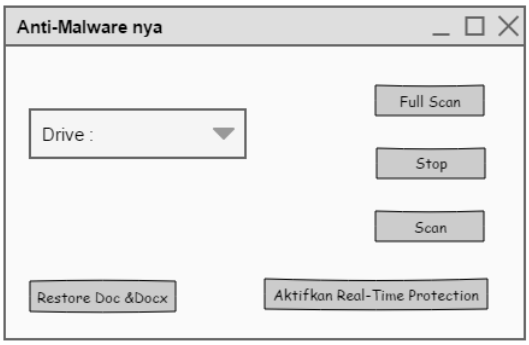
Gambar 6. Parameter Pencarian *file malware*

4.10. Mekanisme Anti-*malware*

Mekanisme anti-*malware* yang dimaksud ialah mekanisme pembersihan komputer dari *malware* yang telah terpasang. Berbeda dengan *malware* yang merupakan program otomatis setelah terpasang dan selalu aktif setelah komputer *booting*.



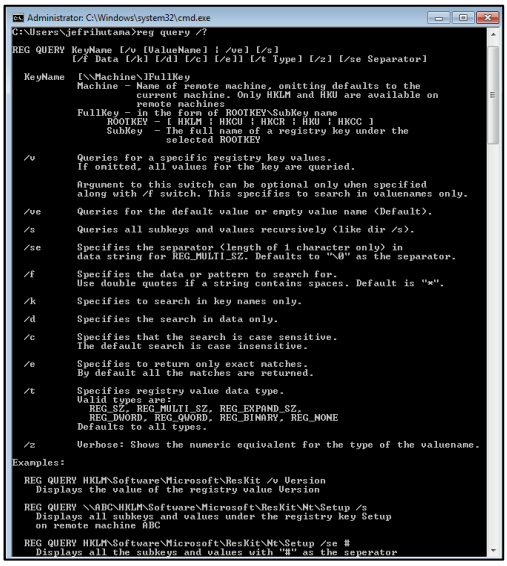
Gambar 7. Mekanisme Anti-*malware*



Gambar 8. *Graphic User Interface Anti-malware*

4.11. Kriteria Proses Pendeteksi *Malware*

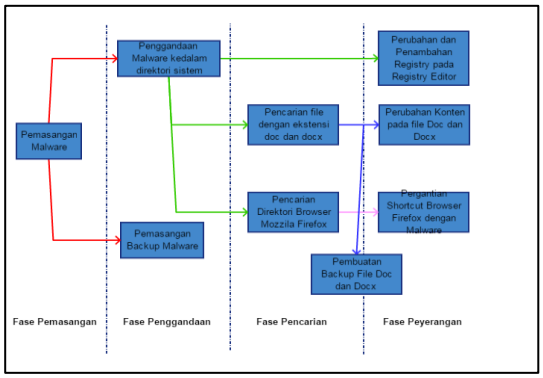
Kriteria pencarian pada anti *malware* meliputi pencarian *file malware* berdasarkan nama *file*nya, pencarian variabel pada Registry Editor, pencarian *file* dokumen yang telah disembunyikan. Nama *file* yang akan dicari antara lain PrimarySearch.bat, fd,bat, AlterReg.bat, DocRemoval.bat, DisDisk.bat, MasterPing.bat, Hide.bat, FirePhon.bat. Pada pencarian variabel dalam Regitry Editor, anti-*malware* akan menggunakan perintah REG QUERY pada cmd.exe untuk mencarinya.



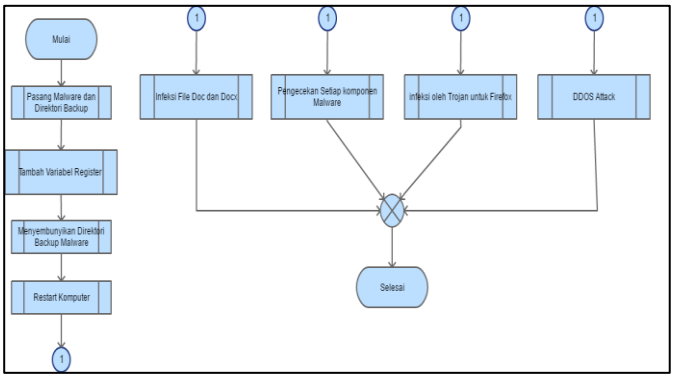
Gambar 9. Perintah REG QUERY Pada cmd.exe

ISSN: 1979-2328

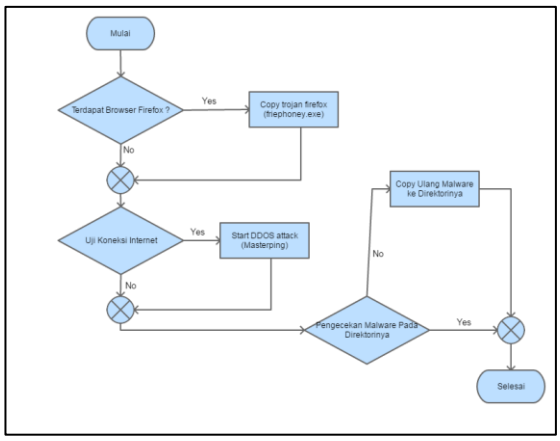
4.12. Diagram *malware* dan Anti-*malware*



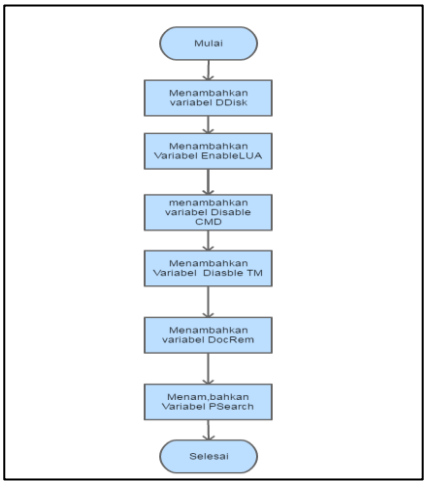
Gambar 3.10 Arsitektur *malware*



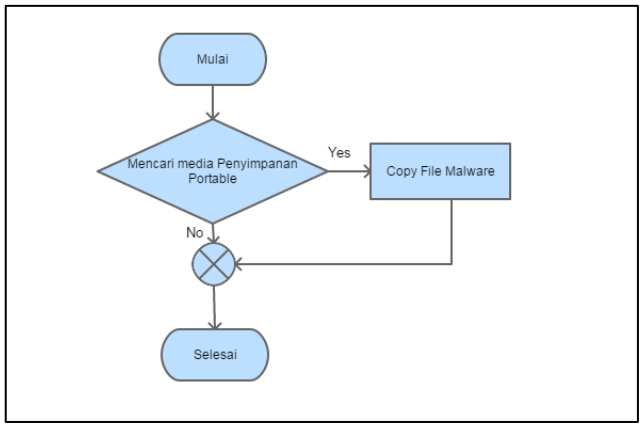
Gambar 3.11 *Flowchart malware*



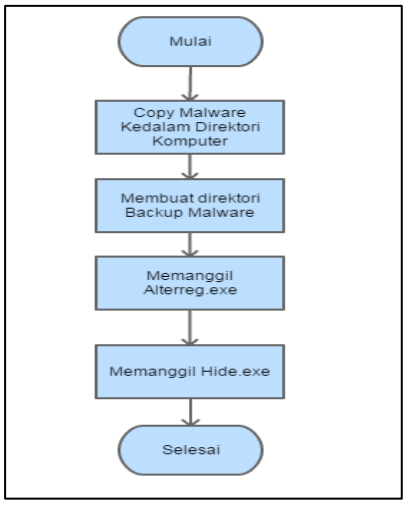
Gambar 3.12 *flowchart* Primary Search



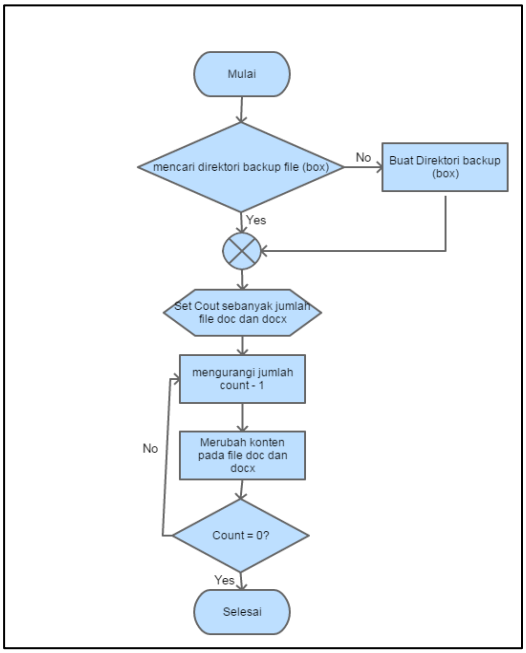
Gambar 3.13 *flowchart* Program AlterReg



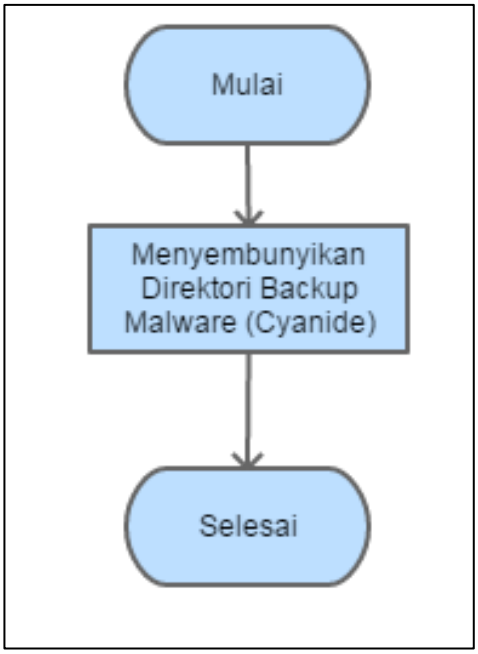
Gambar 3.14 *flowchart* Program Ddisk



Gambar 3.15 *flowchart* Program FD

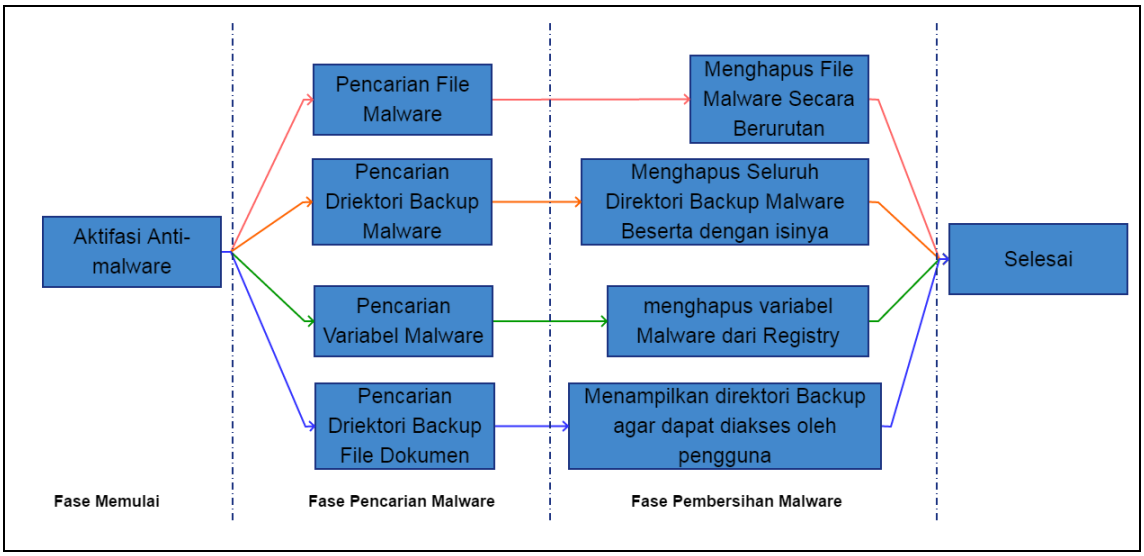


Gambar 3.16 *Flowchart* Program DocRemoval



Gambar 3.17 *flowchart* Program Hide

Beberapa flowchart yang tidak di tampilkan antaralain ; *flowchart* Program SlavePing, *flowchart* Program MasterPing, *flowchart* Program FirePhoney dan *flowchart* Program Slavescare



Gambar 3.18 Arsitektur Anti-*malware*

5. KESIMPULAN

Perancangan Malware standalone bekerja pada level sistem operasi windows dan melakukan penyebaran melalui media penyimpanan, dimana Malware yang dihasilkan akan dapat merubah konten dari dokumen yang tersimpan dalam partisi penyimpanan C: dengan ekstensi .doc dan .docx secara rekursif dan tidak akan mengakibatkan kerusakan pada komponen aplikasi lainnya secara permanen. Anti-malware yang dihasilkan dapat mengembalikan efek yang ditimbulkan oleh malware.

DAFTAR PUSTAKA

Computer Hope, Information on Batch Files, https://www.computerhope.com/batch.htm (2017)

Comodo Antivirus, Malware Vs. Viruses, https://antivirus.comodo.com/blog/ omputer-safety /malware-vs-viruses-whats-difference/

Difference Between, Difference Between Malware and Virus, http://www.differencebetween.com/ difference-between-malware-and-virus/

Hariyanto, B., 2007. Sistem Operasi Edisi 2. *Bandung: Informatika*.

Malware Fox, Anti-Viruses Vs. Anti-Malware, https://www.malwarefox.com/difference-antivirus- antimalware/

Microsoft, Command Prompt (CMD.exe), https://support.microsoft.com/en-us/help/830473/command-prompt-cmd--exe-command-line- string-limitation (2015)

Malware and Antivirus Architecture, eForensics Magazine, https://eforensicsmag.com/malware-and- anti-virus-architecture/

Putra, Y.S., Muslim, M.A. and Naba, A., 2013. Game Chicken Roll dengan Menggunakan Metode Forward Chaining. *Jurnal EECCIS*, *7*(1), pp.41-46.

Rizal, H., Adhy, S. and Wisnu Wirawan, P., 2013. Perancangan dan Pembuatan Mobile Learning Interaktif Berbasis Android dengan Metode Personal Extreme Programming. *Jurnal Masyarakat Informatika*, *4*(8).

Santoso, S. and Widada, S., TEKNIK MEMBONGKAR PERTAHANAN VIRUS LOKAL MENGGUNAKAN VISUAL BASIC SCRIPT DAN TEXT EDITOR UNTUK PENCEGAHAN.

Stallings, William. 1995, "Operating Systems", 2nd Edition, Englewood Cliffs, New Jersey: Prentice-Hall Inc

Tanenbaum, Andrew S. 1992. "Modern Operating Systems", Englewood Cliffs, New Jersey: Prentice-Hall Inc

Widiyanto, A., 2014. SISTEM PORTABLE UNTUK APLIKASI WEB DENGAN MEMANFAATKAN BATCH FILE PROGRAMMING. *SEMNASTEKNOMEDIA ONLINE*, *2*(1), pp.3-05.

Wahyudi, Y., 2015. INSTRUKSI BAHASA PEMROGRAMAN ADT (ABSTRACT DATA TYPE) PADA VIRUS DAN LOOP BATCH. *Media Infotama*, *9*(2).