Perancangan Perangkat Lunak Antivirus Dengan Metode *Scanning Cyclic Redundancy Checksum*-32.

Muhammad Ilhamdi Rusydi[1], Meza Silvana[2], Aidil Akbar[3]
[1,2,3] *Jurusan Teknik Elektro Universitas Andalas*

Abstract— **Perkembangan teknologi komputer yang sangat pesat telah memicu perkembangan suatu sistem otomatisasi computer vision. Salah satu aplikasi pengembangan teknologi berkaitan dengan computer vision adalah sebuah aplikasi yang dapat mengenali objek pada citra. Pengenalan citra huruf sering terkendala dari kemampuan sistem yang hanya mampu mengenali citra pada ukuran font dan jarak tertentu. Oleh karena itu dibuatlah sebuah aplikasi pengenalan citra huruf yang mampu mengenali citra huruf dengan variasi ukuran *font* dan jarak. Tahapan proses pengenenalan citra huruf yaitu *capture*, konversi citra RGB ke citra intensitas, segmentasi, pelabelan-*filtering*, ekstraksi ciri dengan metode zoning, pelatihan dengan jaringan syaraf tiruan backpropagation dan pengujian. Hasil pengujian menunjukkan dari 1500 data citra huruf yang digunakan, 1217 dikenali dan 283 tidak dikenali, dengan tingkat keberhasilan yang berhasil dicapai sebesar 84 %.**

Kata kunci : *Computer vision*, ekstraksi ciri, jaringan syaraf tiruan backpropagation, ukuran *font*, jarak

I. PENDAHULUAN

Pada saat ini komputer sudah menjadi bagian penting dalam kehidupan masyarakat. Perkembangan teknologi komputer yang cukup pesat ikut berperan serta dalam mempercepat pertumbuhan ekonomi disuatu negara. Dimana teknologi komputer sudah banyak digunakan oleh setiap orang mulai dari pengguna korporat, akademis, sampai dengan pengguna rumahan.

Seiring dengan pesatnya kemajuan teknologi terutama teknologi komputer khususnya dibidang teknologi komputer dan jaringan, keamanan dalam penggunaan teknologi komputer menjadi isu yang sering dibahas belakangan ini. Mulai dari ancaman langsung dari para *cracker* atau *hacker* jahat seperti pencurian data ATM (Anjungan Tunai Mandiri) Bank yang terjadi belakangan ini atau yang dilakukan melalui suatu program yang di sebut *malcode* (*malicious code*). *Malicious Code* merupakan suatu program atau *script* apapun yang bersifat merusak atau merugikan, beberapa program yang dapat dikategorikan sebagai *malcode* yaitu virus komputer, *worm*, *trojan horse*, dan *spyware*.[19]

Maraknya penyebaran virus, *worm* atau *trojan horse* saat ini membuat pengguna komputer di Indonesia disibukan oleh penyebaran *worm* lokal yang diperkirakan berhasil menginfeksi ribuan komputer di Indonesia. Media penyebaran pun semakin canggih mulai dari disket, USB flash disk sampai dengan jaringan termasuk internet. Aktifitas *worm* di Indonesia pun semakin meningkat dengan beredarnya tutorial-tutorial untuk membuat *worm* mulai dari artikel-artikel di Internet sampai dengan buku-buku. Sehingga saat ini tidak sulit menemukan buku tutorial membuat *worm* atau virus. Namun hal ini tidak di barengi dengan dengan beredarnya buku-buku yang mengajarkan kita untuk memberantas *worm* kecuali hanya menggunakan antivirus atau penghapusan manual.[18]

Virus-virus komputer dapat dihapus dengan menggunakan aplikasi yang dikenal sebagai antivirus, hanya saja beberapa antivirus dipasarkan dengan harga yang relatif mahal. Bagi para *user* yang memiliki uang untuk membeli antivirus hal seperti ini dianggap bukan suatu masalah besar. Untuk membantu *user* yang tidak mampu membeli lisensi antivirus, maka beberapa perusahaan atau pembuat antivirus telah meluncurkan antivirus yang dapat dipakai oleh *user* tanpa harus membeli lisensi, tetapi *user* diharuskan untuk memiliki koneksi internet dalam melakukan *update* database virus ke komputer *server* pada produsen antivirus, misalnya AVG yang dirilis Grisoft dan PCMav yang dikeluarkan oleh PC Media dan SmadAV.[17]

Masalah yang dihadapi selama ini adalah semakin pesatnya perkembangan virus mengharuskan *user* untuk mendownload data base virus yang semakin besar ukurannya. Setelah *user* mendownload data base antivirus, belum tentu virus yang menginfeksi komputer *user* terdapat dalam data base antivirus tersebut. Metode yang dapat dipakai *user* untuk melakukan proses *scanning* salah satunya adalah metode Cyclic *Redundancy Checksum-32* atau CRC32.[8]

A Ada beberapa penelitian dan tulisan yang mengambil topik tentang Cyclic Redundancy Checksum-32 diantaranya adalah dari **Indra Sakti Wijayanto**[12] dalam makalahnya yang berjudul *"Penggunaan CRC-32 dalam Integritas Data"* yang membahas bahwa Cyclic Redundancy Check (CRC) adalah salah satu fungsi *hash* yang dikembangkan untuk mendeteksi kerusakan data dalam proses transmisi ataupun penyimpanan. CRC menghasilkan suatu *checksum* yaitu suatu nilai dihasilkan dari fungsi *hash*-nya, dimana nilai inilah yang nantinya digunakan untuk mendeteksi error pada transmisi ataupun penyimpanan data. Nilai CRC dihitung dan digabungkan sebelum dilakukan transmisi data atau penyimpanan, dan kemudian penerima akan melakukan verifikasi apakah data yang diterima tidak mengalami perubahan ataupun kerusakan.

Berdasarkan pemikiran serta penelitian-penelitian di atas maka dibuatlah implementasi metode *scanning Cyclic Redundancy Checksum-32* dalam pembuatan perancangan antivirus yang pada hasil akhir antivirus ini memiliki

kemampuan untuk dapat di *update* databasenya oleh setiap *user* nya..

Penelitian ini dilakukan bertujuan untuk merancang satu perangkat lunak antivirus yang mampu memberi kemudahan pada setiap pengguna antivirusnya untuk mengupdate atau menambah *database signature* antivirus dengan nilai crc32 dari virus dan *worm* nya sendiri setiap saat tanpa harus men*download database* pada suatu *server* pembuat aplikasi antivirus.

Ada beberapa manfaat yang didapatkan dari penelitian ini, diantaranya perancangan sistem atau program antivirus ini diharapkan dapat dijadikan solusi terhadap masalah penyebaran *worm* dan virus komputer yang terjadi belakangan ini dimana database virus dan *worm*nya bisa di *update* oleh siapapun. Penelitian ini diharapkan bisa menjadi bahan kajian dan dasar untuk dikembangkan atau dilanjutkan menjadi suatu program antivirus yang lebih kompleks.

Adapun batasan masalah dalam penelitian ini adalah :

1. Analisis pembuatana Antivirus ini sementara hanya dipergunakan sebagai bahan kajian, pembelajaran dan penelitian mengenai metode pencarian data *Cyclic Redundancy Code* (CRC32) untuk di gunakan dalam algoritma pencarian *file* virus atau *worm* pada antivirus.
2. Program ini di rancang hanya untuk mengatasi permasalahan virus dan *worm* komputer saja.
3. *Worm* dan virus yang diatasi masih sebatas virus dan worm yang berjalan pada sistem operasi Windows XP dengan program antivirus yang juga berjalan pada Windows XP juga, karena untuk permasalahan virus dan *worm* sampai saat ini masih sangat banyak terdapat pada sistem operasi windows XP.[16]
4. Antivirus yang akan dibangun adalah antivirus yang menggunakan metode *Cyclic Redundancy Code* (CRC32) dalam pendeteksian virus. Jadi virus yang dapat dikenali adalah virus yang telah terdefinisi nilai CRC32 nya dalam database virus.

II. METODOLOGI PENELITIAN

Ditinjau dari tujuan dasarnya maka penelitian ini termasuk ke dalam penelitian yang bersifat kajian namun memiliki manfaat tepat guna, maksudnya bahwa pemahaman akan hasil penelitian dapat dijadikan dasar sebagai solusi atau pemecahan suatu masalah yang terjadi untuk tujuan tertentu dan merupakan aplikasi dari penelitian yang sudah ada. Aplikasi yang diperoleh dari penelitian ini, diharapkan dapat langsung dipergunakan untuk solusi alternatif menghapus virus dan worm yang banyak beredar saat ini. Sedangkan ditinjau dari sifat-sifat asalahnya maka penelitian ini bersifat eksperimen.

Berdasarkan batasan masalah yang telah dijabarkan sebelumnya, maka sampel penelitian yang digunakan dalam penelitian ini adalah *file* yang dianggap virus dan worm yang pada umumnya memiliki format file berupa exe,com, vbs, bat, scr, dll, db dan lain nya

Ada beberapan tahapan dalam desain sistem ini :

1. Tahapan pertama dimulai dengan mencari sampel virus ataupun worm, sampel bisa diperoleh dari file yang dicurigai sebagai sebuah virus ataupun worm berdasarkan prilakunya, dan sampel bisa juga diperoleh dari beberapa website tertentu.
2. Dari sampel virus yang diperoleh dihitung nilai crc32 nya dengan program penghitung crc32 yang terintegrasi langsung dalam antivirus itu sendiri.
3. Nilai crc32 dari sampel virus yang diperoleh, dimasukan langsung kedalam database antivirus dan diberi nama jenis virus atau worm nya.
4. Setelah database antivirus terisi dengan nilai-nilai crc32 dan nama virus atau worm, antivirus bisa memulai proses pencarian file virus atau worm yang terdapat dalam sistem operasi komputer sesuai dengan virus atau worm yang terdapat pada database antivirus
5. Antivirus melakukan *scanning* pada semua file yang terdapat pada sistem operasi komputer, dan menghitung nilai crc32 nya.
6. Kemudian nilai crc32 yang diperoleh dari setiap file dibandingkan dengan nilai yang terdapat dalam database antivirus. Jika terdapat kesamaan maka file tersebut merupakan virus atau worm yang akan ditampilkan pada sebuah jendela list virus yang ditemui pada antivirus.
7. Berdasarkan list virus atau worm yang diperoleh, lalu akan dilakukan pengambilan keputusan untuk menghapus file tersebut atau mengkarantinanya

Kinerja sistem yang akan dieksperimenkan dianalisa dengan menggunakan penilaian objektif. Hasil-hasil yang didapatkan melalui eksperimen (keluaran sistem) dibandingkan dengan teori-teori yang berasal dari literatur yang ada.

Agar penelitian dapat lebih terarah dan efektif, penulis telah menyusun dan akan mengikuti prosedur penelitian sebagai berikut:

a. Tinjauan dan Studi Kepustakaan

b. Penyusunan Algoritma Program : Program yang dirancang dalam penelitian ini terbagi atas 2 (dua) algoritma pokok, yaitu:

i. Algoritma penghitung nilai CRC32 dari sebuah file yang dianggap virus atau *worm*.

ii. Algoritma proses scanning antivirus terhadap setiap data yang terdapat didalam direktori komputer.

c. Perancangan dan Pembuatan Program (Visual Basic 6.0).

d. Analisa Keluaran Sistem

III. PENGUJIAN DAN ANALISA

Pada penelitian ini diambil 4 macam pengujian antivirus dimana dalam melakukan pengujian perangkat lunak anti virus, penulis menggunakan sampel virus dan beberapa perangkat lunak pendukung lainnya yang dapat menjadi faktor perangkat lunak yang dibangun dapat berjalan dengan baik, diantaranya :

1. Windows XP Service Pack 2
2. Deep Freeze, Sandboxie.
   Berfungsi untuk meminimalisir efek kesalahan yang dapat terjadi pada sistem operasi komputer saat proses pengujian antivirus terhadap sampel-sampel virus.
3. Sampel virus : Pada antivirus ini diambil 50 buah sampel antivirus yang diperoleh dari website yang menyediakan sampel virus yaitu "vx.netlux.org" dan Morphostlab.com. Sedangkan sampel virus yang digunakan dapat dilihat sebagai berikut berdasarkan klasifikasinya:
   a) P2P Worm: Zevity, Kersex, G_Spot, Blaxe, Anker.
   b) Internet Worm: Shadnake.a , Kibuv, Delf, Blue Code, Bizex. Dan local internet Worm yaitu: Momo Geisha, Bokep,PCMavirus, Ratu Felisha, Shiren Sungkar.
   c) E-mail Worm: Valcard, Enviar, Celebit, Bad Ass, Altice.
   d) IM Worm: Funner, Hargwig, Heva, Jitux, Lamar.G.
   e) Virus File: Devir, Creev, Cloz, BlackCat, Arcer, HantuRimba, Kemben, CintaLaura, Gendel32, Mita.
   f) Boot Sector Virus: AntiWin, Bagoes, LeftShift. Taekwondo, Vendetta.
   g) Virus File dan Boot Sector: Kangen, Bacalid.a.
   h) Trojan: RedEvil, Delf.a, ColdFusion, BlackBird, AntiMcAffe.

Selanjutnya data dari setiap percobaan tersebut tersebut adalah sebagai berikut :

1. Percobaan Pengujian update database antivirus dari sampel virus yang digunakan.

Gambar 1. Penambahan database antivirus.

Pada percobaan ini bertujuan untuk melihat kemampuan sistem dalam menambah database antivirus yang akan digunakan sebagai signature untuk mencari virus dalam sebuah sistem operasi computer seperti yang ditunjukan pada gambar 1. Hasil dari percobaan ini dapat dilihat pada tabel 1.

Tabel 1
List nilai crc32 dan Nama Virus

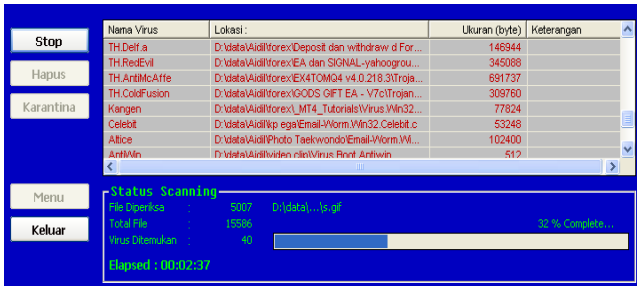
Nilai crc32	Nama Virus	Nilai crc32	Nama Virus
C9989D3F	Zevity	8202AA25	LameKill
35F77190	Kersex	DB635DFB	Collector
7ADDB59A	G_Spot	D533356	Redlof
5C82F1CB	Blaxe	474B841B	Kangen
C04496FF	Anker	E6B3446D	Devir
884BECE8	Shadnake.a	F329D274	Creev
6DB23CDB	Kibuv	FBDE8359	Cloz
BE411E00	Delf	42655DCD	BlackCat
F5E9F874	BlueCode	97BD0D35	Bacalid.a
634C343	Bizex	5C2FF6A2	W32.Arcer
3CF103F4	Valcard	D25765A8	RedEvil
4F46152	Enviar	E29D70F8	Delf.a
BF5046D5	Celebit	208D87B	ColdFusion
FD494CEA	BadAss	F7ECD2C2	BlackBird
F5A11E7F	Altice	EAF7C48D	AntiMcAffe
ED127615	HantuRimba	8DA3D04F	AntiWin
8722CF50	Kemben	54A40B6F	Bagoes
703E323A	CintaLaura	7D289CF	LeftShift
D41690E7	gendel32	FD8352DF	Taekwondo
5703BBD9	mita	5DF2F45	Vendetta
6A1DF270	momogeisha	EFE25014	Funner
693C0035	bokep	FF1C95ED	Harwig
DB757AF3	PcMavirus	91C135DA	Heva
F016EE35	RatuFelisha	50AB1461	Jitux
D88875A4	ShirenSungkar	52A5A570	Lamar.g

Pada table 1 diatas dapat dilihat nilai crc32 yang diperoleh dari setiap sample virus. Dan nilai crc32 inilah yang akan ditambahkan kedalam database antivirus dengan disertai memasukan nama virusnya. Setelah database terisi dengan nilai crc32 maka Antivirus siap digunakan untuk mencari virus pada sistem operasi komputer sesuai dengan *signature* dari nilai crc32 setiap sample virus yang terdapat dalam database.

2. Pengujian Pencarian *File* Virus

Pengujian dilakukan dengan memasukan *file* virus pada beberapa lokasi berbeda didalam *Drive* hardisk. Selanjutnya proses pengujian perangkat lunak antivirus

dalam mencari *file* virus yang terdapat dalam sistem operasi komputer dapat dimulai. Dengan memilih menu scan yang secara otomatis akan langsung membuka lokasi *Drive* atau *directory* pencarian *file* virus, setelah directory scan dipilih maka proses scan langsung dimulai. Selama proses scan berjalan, *file* antivirus yang ditemukan akan langsung ditampilkan dalam jendela informasi seperti yang ditunjukan pada gambar 2 dibawah ini:



Gambar 2 Proses Scan

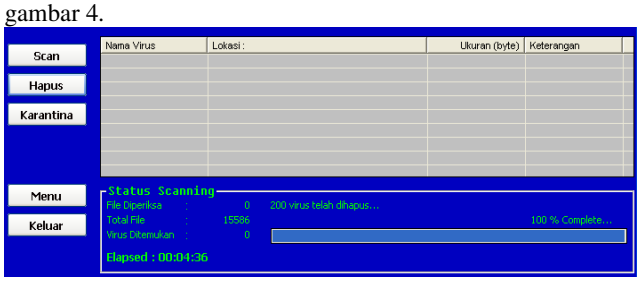
Setelah semua *file* yang terdapat dalam *drive* atau *directory* habis diperiksa maka proses scan akan dihentikan.

3. Pengujian Eksekusi *file*

Pada perangkat lunak antivirus dalam eksekusi *file* yang dianggap virus memiliki pilihan untuk menghapus atau mengkarantina *file* tersebut. Pada gambar 3 terdapat total 200 *file* virus yang telah ditemukan dari 15.586 *file* yang telah diperiksa dalam waktu tiga menit tiga detik.

Gambar 3 Virus yang ditemukan

Selanjutnya proses eksekusi *file* akan dilakukan, dimana dalam pengujian ini dilakakukan penghapusan *file*. Dan hasilnya semua *file* virus dihapus seperti ditunjukan pada gambar 4.



Gambar 4 Hasil eksekusi penghapusan *file* virus

4. Perbandingan dengan *Freeware* Antivirus Populer

Untuk membandingkan dengan antivirus gratis atau freeware yang banyak digunakan saat ini maka dilakukan pengujian terhadap sampel yang sama dengan tujuan untuk mengetahui kemampuan antivirus yang telah dirancang apakah cukup layak dibandingkan dengan antivirus gratis populer yang banyak beredar saat ini.

Dan Antivirus populer yang dipilih yaitu SmadAV dan PCMAV untuk antivirus lokal dan Avira untuk antivirus luar negeri. Antivirus ini dipilih karena banyaknya pengguna dari antivirus ini, dimana Avira merupakan "Top 5 Free Antivirus 2011" versi majalah PCWorld dan SmadAV sendiri untuk Rev 8.4 sudah didownload oleh 24.246.082 pengguna, kemudian Panda Cloud Antivirus yang merupakan keluaran dari Panda Antivirus Security dan juga masuk dalam "Top 5 Free Antivirus 2011". Antivirus lain yang digunakan adalah antivirus berbayar Kaspersky antivirus sebagai bahan pembanding terhadap antivirus freeware lain.

Proses pengujian dilakukan terhadap sampel yang sama, hanya saja untuk antivirus populer ini tidak dilakukan eksekusi terhadapa file virus. Karena tujuannya hanya untuk membandingkan proses dan kemampuan pencarian terhadap sampel virus yang sama.

Dari hasil pengujian ini diperoleh hasil dimana antivirus populer luar negeri seperti Avira, Panda Cloud Security dan Kaspersky sangat baik dalam mendeteksi virus-virus internasional tapi kesulitan dalam mendeteksi virus-virus buatan lokal. Sedangkan antivirus lokal mampu mendeteksi virus lokal yang ada tapi hanya mampu menemukan sedikit virus internasional. Hal ini disebabkan karena antivirus-antivirus tersebut tidak memiliki signature-signature virus yang tidak dapat dideteksinya sesuai sampel dalam database antivirus tersebut.

ANALISA

Dari hasil pengujian yang dilakukan dengan menggunakan tiga puluh buah sampel virus, yang kemudian dicari nilai crc32 dari sampel virus tersebut dan ditambahkan atau di*update* sebagai *signature* virus dalam database antivirus.

Setelah database antivirus berisi dengan *signature* virus berupa nilai crc32nya dan nama virus, ternyata antivirus mampu mendapatkan semua *file* virus yang diletakan secara random dalam beberapa *Drive* atau direktori. Antivirus mampu menemukan setiap file virus walaupun file tersebut memodifikasi diri seperti merubah nama file maupun ekstensi filenya.

Pada pengujian ini antivirus menemukan 200 *file* virus dari total 15.586 *file* yang terdapat dalam *drive* atau direktori berbeda. Antivirus ini mampu menemukan semua 50 sampel virus yang disebar menjadi 200 file virus, hal ini berarti antivirus berhasil mencari virus sesuai dengan yang terdapat dalam dabatase antivirus dengan tingkat keberhasilan 100%. Dimana pencariannya dapat diseleseaikan dalam waktu 4 menit 36 detik. Jadi antivirus yang menggunakan metode *scanning* Cyclic Redundancy *Checksum*-32 ini mampu memeriksa lebih kurang 56 *file* dalam satu detiknya. Berarti antivirus mampu memeriksa *file* dalam waktu yang cukup

cepat. Berdasarkan *situs* antivirusware.com yang melakukan ujicoba kecepatan scanning berbagai macam antivirus, dimana sebagian besar antivirus mampu memeriksa file dengan kecepatan mulai dari 15 file/detik sampai dengan 70 file/detik tergantung dengan ukuran file yang diperiksa.

Dari pengujian yang dilakukan antivirus ini mampu menemukan semua sampel antivirus yang walaupun memiliki klasifikasi yang berbeda, dimana antivirus ini bisa menemukan tipe internet worm, email worm, P2P Worm, virus File, Virus Boot Sector, bahkan Trojan Horse. Karena pada dasarnya antivirus ini melakukan pencarian file virus berdasarkan *signature* atau *fingerprint* (sidik jari) berdasarkan nilai crc32 dari file virus tersebut. Dan Antivirus ini sendiri tidak akan bisa menemukan *file* virus pada sebuah sistem operasi komputer yang belum terdapat *signature* nilai crc32 nya dalam *database* Antivirus. Sehingga untuk dapat mencari sebuah *file* virus, terlebih dahulu mesti diperoleh sampel virus yang akan dicari. Kemudian dari sampel virus tersebut dicari nilai crc32 nya dan langsung ditambahkan dalam *database* Antivirus.

Berdasarkan pengujian pembandingan antivirus hasil rancangan dengan antivirus populer lainnya yaitu Avira, SmadAV, Panda Cloud Antivirus, PCMAV dan Kaspersky dapat dilihat bahwa penggunaan metode Scanning Cyclic Redundancy Checksum-32 pada rancangan antivirus ini memiliki kemampuan yang hampir setara dengan antivirus populer gratis yang banyak beredar saat ini. Sehingga pemahaman akan metode Scanning Cyclic Redudancy Checksum-32 sangat penting untuk diketahui sebagai landasan dalam mengembangkan sebuah antivirus.

Dari hasil perbandingan dengan antivirus lain juga dapat dilihat jika pengupdate database haruslah dilakukan agar antivirus mampu mendeteksi lebih banyak virus. Dimana antivirus luar seperti Avira, Panda Cloud dan Kaspersky tidak mampu mendeteksi beberapa virus lokal yang tidak terdapat dalam database antivirusnya. Sedangkan antivirus lokal SmadAV dan PCMAV mampu membaca semua virus lokal dari sampel virus yang digunakan tapi hanya mampu membaca sedikit virus luar karena tidak terdapat dalam database SmadAV dan juga PCMAV. Namun dengan menggunakan sebuah antivirus yang dapat ditambah atau update databasenya setiap saat pengguna akan mampu mencari berbagai macam virus yang terdapat dalam sistem operasi setelah databasenya ditambahkan sesuai sampel virus yang ditemukan dalam sistem operasi.

IV. KESIMPULAN

Berdasarkan analisa terhadap hasil percobaan yang didapat, maka dapat diambil beberapa simpulan sebagai berikut:

1. Kemampuan penambahan database secara manual oleh pengguna pada sistem antivirus, mampu memudahkan pengguna untuk mengupdate database antivirus setiap saat berbasis objek virus yang ada pada sebuah sistem operasi. Dan pengguna bisa menghemat pemakaian media penyimpanan dengan mengisi database sesuai file virus yang akan menjadi target pencarian pada sebuah sistem operasi komputer. Nilai crc32 yang dihitung dari dari sebuah file virus digunakan sebagai *signature* atau *fingerprint* (sidik jari) dari virus tersebut dalam mendeteksi keberadaan virus tersebut dalam *directory* sistem operasi komputer.
2. Pencarian file virus oleh sistem antivirus dilakukan dengan menghitung nilai crc32 dari setiap file yang ada pada suatu directory, kemudian nilai crc32 yang diperoleh akan dibandingkan dengan nilai crc32 yang terdapat pada database, jika tidak terdapat dalam database maka file tidak dapat dianggap sebagai virus.

REFERENSI

[1] Aat Shadewa. (2006). *Rahasia Membuat Antivirus Menggunakan Visual Basic*. Yogyakarta : DSI Publishing.

[2] Anhar. (2009). *Checksum CRC32*. http://ilmukomputer.org/wp–content/uploads/2009/ 06 / anharku-*checksum*crc32.pdf/. Diakses tanggal 10 Januari, 2011.

[3] Aycock, John. (2006). *Computer Viruses and Malware.* Canada : Springer.

[4] Darmal, Achmad. (2006). Computer *Worm* 1 Secret Underground Coding. Jakarta: Jasakom.

[5] Darmal, Achmad. (2006). Computer *Worm* 2 Secret Underground Coding. Jakarta: Jasakom.

[6] Erbschloe, Michael, et. al., (2005), *Trojan*, *Worms*, *and Spyware: A Professional Guides to Mallicious Code,* Elsevier inc. Burlington MA.

[7] Gordon, A., Lawrence et. al., (2006), *CSI/FBI Computer Crime and Security Survey 2006,* CSI Publication, Washington DC, http://www.GoCSI.com/, 1 November 2006.

[8] Hirin,A.M. (2010). *Cara Praktis Membuat Antivirus Komputer*. Jakarta: MediaKita

[9] Narapratama, Ditto. (2006). *Perbandingan Performansi Algoritma Adler-32 dan CRC-32 pada Library Zlib.* Bandung : Institut Teknologi Bandung

[10] Nazario, Jose, et. al., (2004), *Defense and Detection Srategies Againts Internet Worms,* Artech House inc., Norwood MA.

[11] Szor, Peter (2005), *The Art of Computer Virus Research and Defense,* Addison Wesley Proffesional, New Jersey.

[12] Wijayanto, I. S. (2006). *Penggunaan CRC32 Dalam Integritas Data.* Bandung : Institut Teknologi Bandung. 12

[13] Yohanes Nugroho (2005), *Analisis Lengkap Virus Brontok*, http://www.compactbyte.com /brontok/Analisis Lengkap Virus Brontok.html, 11 September 2006.

[14] Anonima. (2006), *PC Viruses in-the-wild 2005-2006*, Wildlist Organization Internationa, http://www.wildlist.org/, 8 Januari 2011.

[15] Anonima. (2009). Download Virus & Software, Virus, http://morphians.wordpress.com/download-virussoftware/, Virus, diakses 11 Januari 2011

[16] Anonima. (2010). Windows7 more secure than previous versions of 10 reasons.

http://www.softcov.com/operating-system/windows7-more-secure-than-previous-versions-of-10.html/, diakses tanggal 11 Januari 2011

[17] Prihardanto, Muhammada Dinto (2010). Studi Perbandingan Beberapa Fungsi Hash dalam Melakukan Checksum Berkas. Bandung: Institute Teknologi Bandung

[18] Anonima. (2011). Perangkat Lunak Antivirus. http://id.wikipedia.org/wiki/Perangkat_lunak_antivirus/, diakses tanggal 11 Januari 2011

[19] Rysna, Mugni Agnina. (2010) Beberapa Kasus Penyebaran Virus. http://teknologi.kompasiana.com/internet/2010/12/06/beberapa-kasus-penyebaran-virus/. diakses tanggal 8 Januari 2011

[20] Yanuar Nugraheddy, Dwi. (2010). Jenis-Jenis Malware Pada Komputer. http://teknologi.kompasiana.com/internet/2010/07/07/jenis-jenis-malware-pada-komputer/ . diakses tanggal 8 Januari 2011

[21] Mediati, Nick. (2011). Top 5 Free Antivirus for 2011. http://www.pcworld.com/reviews/collection/5928/2011_free_av.html/. Diakses tanggal 13 April 2011.

[22] Tim SmadAV. (2011). Smadav 2011 Rev. 8.4 dirilis. http://www.smadav.net/. Diakses tanggal 13 April 2011