Saintek Vol 6, No 1 Tahun 2011

PENTINGNYA APLIKASI PENANGANAN SPYWARE UNTUK KEAMANAN PRIVASI USER PADA SEBUAH KOMPUTER

Zainudin Bonok

Jurusan Teknik Elektro Fakultas Teknik
Uiversitas Negeri Gorontalo

ABSTRAK

Spyware telah menimbulkan banyak kerugian, di antaranya penggunaan sumberdaya secara tidak sah dan pelanggaran terhadap privasi user. Oleh karena itu, penelitian ini dilakukan dengan tujuan untuk mengimplementasikan teknik pemeriksaan file yang baik untuk aplikasi penanganan *spyware*, dan membuat aplikasi penanganan *spyware* yang mampu menghapus file *spyware* yang ditemukan. Metode yang digunakan dalam penelitian ini adalah observasi eksperimental dengan melakukan eksperimen langsung membuat aplikasi penanganan *spyware* yang menggunakan fungsi-fungsi API dalam metode pencarian filenya. Serta studi kepustakaan dan observasi mengenai kerugian-kerugian yang ditimbulkan oleh *spyware*. Selanjutnya, analisis kebutuhan program menggunakan metode UML (*Unified Modelling Language*), kemudian diimplementasikan menggunakan bahasa pemrograman Microsoft Visual Basic 6.0 dan diuji menggunakan metode pengujian black box testing dan white box testing. Penelitian ini menunjukkan bahwa aplikasi yang dibuat dapat digunakan untuk melakukan pencarian terhadap file *spyware*, melakukan penghentian proses dan penghapusan file *spyware*.

Kata Kunci: *Spayware, Unified Modelling Language*

ABSTRACT

Spyware has made a lot of losses, such as using resource illegally and violation of user's privacy. Therefore this research is being done with some purpose to implement a good examination file technique for spyware handling application, and to make spyware handling application which can erase spyware file that is found by it. The method which is used in this research is experimental observation by directly doing an experiment in making spyware handling application that used API functions in the searching files method. Also being used are the literature review and observation of

the losses that is made by spyware. Furthermore, the analysis for program needs is using UML (Unified Modelling Language) method, then it is implemented using Microsoft Visual Basic 6.0 compiler and being tested using both black box and white box approaches. This research shows that the application which has been made can be used to search for spyware files, stopping process and deleting the spyware file.

Keywods: *Spayware, Unified Modelling Language*

PENDAHULUAN

Perkembangan internet sebagai media informasi yang tanpa batas telah menyentuh berbagai aspek kehidupan masyarakat. Kondisi tersebut menimbulkan dua masalah pokok yang berhubungan erat dan membutuhkan solusi dengan segera, yaitu perlindungan privasi dan membangun kepercayaan antara individu dengan komputer, masyarakat dengan komputer, serta sesama anggota masyarakat yang termediasi oleh komputer. Salah satu alasan kurangnya privasi adalah sifat internet yang tak berbatas. Tidak adanya aturan yang mendefinisikan informasi di internet yang tergolong personal, dan tanpa adanya batas-batas penggunaan informasi tersebut bagi pihak ketiga, banyak data tentang seseorang tersedia untuk diambil. Dan banyak juga orang yang bersiap mengambilnya. Internet telah menyuburkan perkembangbiakan "perusahaan" pemantau penggunaan *web*. Mereka memilih berdasarkan profil-profil pribadi yang dibuat saat seseorang mengunjungi suatu situs dan kemudian menjual data tersebut kepada orang lain.

Internet juga menjadi surga bagi para "pemburu elektronik". Begitu banyak situs yang menawarkan jutaan nomor telepon dan alamat, ada lagi yang dapat memasok nomor Jaminan Sosial dan data Surat Ijin Mengemudi, bahkan ada lagi yang bisa menelusuri data tentang dana kampanye politik. Baik pesan *e-mail* maupun transaksi finansial pada jaringan publik tidak seluruhnya aman dari mata-mata pengintip. Umumnya hal ini disebabkan oleh lemahnya kombinasi keamanan pada perusahaan-perusahaan pengelola situs dan kecerobohan orang yang melakukan transfer informasi. Adapun celah-celah yang bisa dimanfaatkan oleh para *hacker* untuk mengetahui kegiatan orang saat *browsing* di internet, antara lain biasanya melalui *e-mail*, *browser*, *cookie* dan *spam*. Biasanya hal-hal ini dilakukan dengan menggunakan bantuan *spyware*, di mana *spyware* ini akan bekerja dengan tenang tanpa keributan sehingga tidak dike-tahui oleh *user*. Adapun beberapa kerugian yang bisa disebabkan oleh *spyware* di samping pelanggaran terhadap privasi *user*, antara lain penurunan performa sistem komputer, penggunaan ruang di *harddisk*, penggunaan *bandwith* jaringan, dan sebagainya. Hal-hal ini dapat menyebabkan kerugian materi yang tidak kecil bagi user. Untuk mengatasi hal-hal yang berkenaan dengan *spyware*, telah banyak dibuat aplikasi-aplikasi untuk melacak dan menghapus file-file *spyware*. Namun, umumnya aplikasi-aplikasi semacam ini tidak bersifat gratis dengan harga yang lumayan mahal. Oleh karena itu, penulis, menggunakan hal ini sebagai bahan penelitian.

Sebuah definisi dari Dick Hazeleger (pendiri "Spyware List"): "Spyware adalah nama yang diberikan untuk software yang tanpa sepengetahuan user melakukan sejumlah hal, terutama melacak pemakaian Internet user dan mengirim informasi tersebut, juga tanpa sepengetahuan user ke suatu komputer (*Server*) yang dikelola oleh si pengembang *software spyware*. Dengan melakukan hal tersebut, profil user dapat dikumpulkan tanpa ijin dan sepengetahuan user yang nantinya bisa dipergunakan untuk tujuan komersial atau tujuan lainnya. Hal ini juga akan memakan sumberdaya di komputer user dan *bandwidth* internet, meski yang utama adalah pelanggaran *privacy* user (Utdirartatmo, 2005:83)

Spyware merupakan serangan yang tidak menyerang aplikasi tertentu, melainkan masuk ke dalam *personal computer* (PC) tanpa diketahui penggunanya. Menurut Wong Joon Hoong, Country Manager untuk Malaysia, Indonesia dan Brunei Trend Micro Inc kepada pers di Jakarta. Magdalena (2003). Menurut situs searchcrm.com, *spyware* adalah teknologi yang dapat mem-bantu dalam mengumpulkan informasi tentang seseorang atau organisasi tanpa sepengetahuan mereka. Di internet, di mana hal ini biasa disebut sebagai *spybot* atau *tracking software*, *spyware* adalah pemrograman yang ditempatkan pada komputer seseorang untuk secara rahasia mengum-pulkan informasi tentang user dan meneruskannya ke pembuat iklan atau pihak yang ketiga yang tertarik.

Di lingkungan komputer, istilah *spyware* merujuk kepada sebuah kategori yang luas dari *malicious software* yang didesain untuk memotong atau mengambil bagian dalam pengon-trolan operasi sebuah komputer tanpa persetujuan dari pemilik mesin atau *user* yang berhak. Selagi istilah tersebut dipakai untuk menggambarkan *software* yang secara diam-diam memoni-tor *user*, istilah tersebut berkembang lebih luas lagi menjadi *software* yang mengambil-alih operasi komputer untuk keuntungan pihak ketiga. Secara sederhana, *spyware* adalah tipe program yang mengawasi aktivitas *user* selama menggunakan komputer dan mengirimkannya kepada *hacker* melalui internet. *Spyware* dapat mengumpulkan berbagai macam informasi tentang *user*. Beberapa program yang lebih baik dapat melacak tipe *website* yang dikunjungi oleh *user* dan mengirimkannya ke agensi periklanan. Beberapa versi yang mengganggu dapat mencatat apa yang diketik oleh *user* untuk memperoleh informasi *password* atau nomor kartu kredit. Adapun versi yang lain hanya langsung memunculkan iklan-iklan. Situs en.wikipedia.org.

Istilah *spyware* pertama kali digunakan pada 17 oktober 1994 dalam sebuah pemberitahuan Usernet yang menonjolkan hiburan di model bisnis Mcrosoft. *Spyware* selanjutnya merujuk kepada peralatan mata-mata seperti kamera mini. Tetapi, pada awal 2000, pendiri Zone Labs, Gregor Freund, mengguna-kan istilah ini dalam sebuah pengumuman resmi untuk Zone Alarm Personal Firewall. Sejak itu pengguna komputer telah menjadikannya pengetahuan umum. Pada awal 2000, Steve Gibson dari Gibson Research menyadari bahwa sebuah *software* periklanan telah ter-*install* dalam sistemnya, dan dia mencurigai *software* tersebut telah mencuri informasi pribadinya. Setelah menganalisa *software* tersebut, dia memutuskan bahwa itu hanyalah kom-ponen *adware* dari perusahaan Aureate (selanjutnya Radiate) and Conducent. Dia akhirnya menarik kembali tuntutannya bahwa *software* tersebut mengumpulkan informasi tanpa sepenge-tahuan *user*, namun

tetap menuntut perusahaan tersebut karena meng-*install*-nya secara sembunyi-sembunyi dan membuatnya sukar untuk dihapus. Sebagai hasil dari analisisnya pada tahun 2000, Gibson merilis program *anti-spyware* yang pertama, yang diberi nama OptOut, dan masih banyak lagi *software* berbasis penangkal yang bermunculan setelahnya.

Berdasarkan survei pada tahun 2004 yang dilakukan oleh AOL dan National Cyber-Security Alliance, 80% pengguna komputer yang disurvei memiliki beberapa *spyware*, dengan rata-rata 93 komponen *spyware* per komputer. 89% dari user dengan *spyware* yang disurvei melaporkan bahwa mereka tidak mengetahui keberadaan *spyware* tersebut, dan 95% melaporkan bahwa mereka tidak memberikan ijin untuk peng-*install*-an *spyware* tersebut. Situs en. wikipedia.org.

Spyware tidak menyebar secara langsung seperti halnya *worm* atau virus komputer. Umumnya, sistem yang terinfeksi tidak dapat menyebarkannya ke komputer yang lain. Bahkan, *spyware* masuk ke sistem melalui penipuan terhadap *user* atau melalui eksploitasi kelemahan-kelemahan *software*. Cara yang paling sering menyebabkan sebuah komputer terinfeksi *spyware* melibatkan user yang meng-*install*-nya. Namun, user cenderung untuk tidak meng-*install software* jika mereka mengetahui bahwa hal tersebut akan mengganggu lingkungan kerja mereka dan membocorkan privasi mereka. Begitu banyak program *spyware* yang telah menipu user, baik dengan mengekori bagian dari sebuah *software* favorit, atau dengan memperdayai *user* untuk melakukan sesuatu hal yang akan meng-*install software* tersebut tanpa disadari. *Spyware* dapat juga dipaketkan dengan *shareware* atau *software-software* lain yang dapat di-*download*, seperti CD musik. User men-*download* sebuah program dan meng-*install*-nya, selanjutnya *installer* tersebut akan menambahkan peng-*install*-an *spyware*. Walaupun *software* yang diinginkan itu sendiri tidak menyebab-kan kerugian, tetapi *spyware* yang ikut dipaketkan dapat merugikan. Pada beberapa kasus, pembuat *spyware* membayar pembuat *shareware* untuk mempaketkan *spyware* ke dalam *software* mereka. Pada kasus yang lain, pembuat *spyware* mengemas ulang *software* gratis dengan *installer* yang menambah *spyware*. Situs en.wikipedia.org

METODE

Teknik pengumpulan data yang digunakan dalam penelitian ini adalah metode observasi eksperimental dengan melakukan eksperimen langsung membuat aplikasi penanganan *spyware* yang menggunakan fungsi-fungsi API dalam metode pencarian *file*-nya. Adapun metode pencari-an yang digunakan adalah metode pencarian linear. Instrumen yang digunakan dalam penelitian ini adalah lembar observasi. Hasil dari observasi akan dijadikan sebagai bahan kajian pada analisa kondisi awal untuk membandingkan kondisi komputer yang telah terinfeksi *spyware* dengan komputer yang belum terinfeksi.

Berikut adalah langkah-langkah yang akan dilakukan dalam merancang sistem:

1. Membuat *Use Case Diagram* dan *use-case spesification*, di antaranya:
   a. *Use Case* menginput nama file *spyware* di database.

b. *Use Case* menghapus nama file *spyware* di database.
c. *Use Case* melakukan validasi.
d. *Use Case* melakukan pencarian / *scanning*.
e. *Use Case* menampilkan hasil pencarian / *scanning*.
f. *Use Case* menghapus file *spyware* yang ditemukan.

2. Membuat *Deployment Diagram*.
3. Membuat *Class Diagram* dan identifikasinya, di antaranya:
   a. *Class Diagram* untuk form_utama.
   b. *Class Diagram* untuk form_dbeditor.
   c. *Class Diagram* untuk form_carifile.
4. Membuat *Statechart Diagram*.
5. Membuat *Activity Diagram*, di antaranya:
   a. *Activity Diagram* untuk form utama.
   b. *Activity Diagram* untuk form database editor.
   c. *Activity Diagram* untuk form pencarian file.
6. Membuat *Sequence Diagram*, di antaranya:
   a. *Sequence Diagram* untuk *Use Case* menginput nama file *spyware* di database.
   b. *Sequence Diagram* untuk *Use Case* menghapus nama file *spyware* di database.
   c. *Sequence Diagram* untuk *Use Case* melakukan validasi.
   d. *Sequence Diagram* untuk *Use Case* melakukan pencarian / *scanning*.
   e. *Sequence Diagram* untuk *Use Case* menampilkan hasil pencarian / *scanning*.
   f. *Sequence Diagram* untuk *Use Case* menghapus file *spyware* yang ditemukan.
7. Membuat *Component Diagram*.

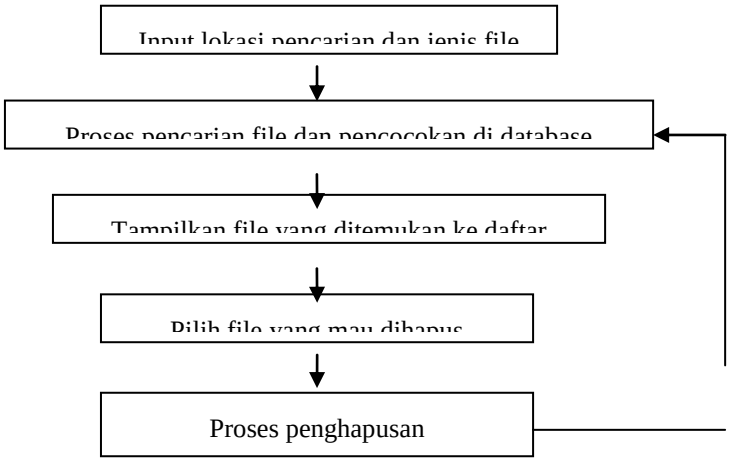
HASIL DAN PEMBAHASAN

Analisa Kondisi Awal

Ancaman *spyware* tergolong sebagai ancaman yang kasat mata. Umumnya aplikasi *spyware* menciptakan lubang keamanan pada sistem teknologi informasi yang memungkinkan penyusup mengacak-acak, mencuri informasi, bahkan mengambil alih sistem. Biasanya sebagian besar pengguna tidak menyadari komputernya telah terjangkiti *spyware*, mereka beranggapan gangguan kinerja sistem, stabilitas dan koneksi komputer disebabkan masalah peranti keras, kesalahan instalasi atau infeksi virus. Mayoritas *spyware* masuk ketika pengguna menginstalasi peranti lunak gratis (*freeware*). Dalam satu pasal pada kesepakatan lisensi *freeware* - EULA (*End User License Agreement*), secara implisit disebut-kan bahwa pengguna menyetujui "Tindakan apa pun yang akan dilakukan pengembang untuk meningkat-kan layanan," termasuk *spyware*. Dalam praktiknya, pengguna jarang sekali meneliti EULA dan langsung meng-klik "*I Agree*." Padahal, alangkah baiknya jika pengguna mengetahui betul apa saja yang dilakukan peranti lunak terhadap sistem

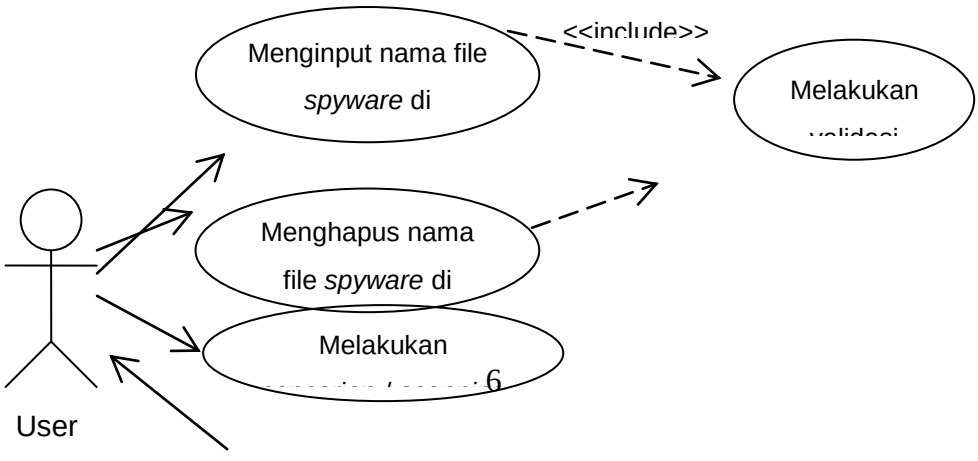
komputernya. Perlu diketahui juga bahwa pengembang *freeware* mendapatkan pendapatan dari informasi yang dikumpulkan *spyware*.

Umumnya pada sistem komputer yang telah terjangkiti *spyware* terjadi penurunan per-forma pada kinerja komputer, misalnya akses internet yang semakin berat, muncul banyak *pop-up* dalam jendela *browser*, sering terjadi *crash* pada sistem. *Spyware* juga menyebabkan terja-dinya penggunaan sumber-daya-sumberdaya yang tidak sah tanpa sepengetahuan user, misalnya penggunaan *memory* dan peng-gunaan ruang di *harddisk* serta penggunaan *bandwidth* yang melambatkan lalu-lintas jaringan. Oleh karena itu, aplikasi yang dibuat oleh penulis akan me-nitikberatkan pada penghapusan file *spyware* yang terdapat di *memory* dan *harddisk*. Setelah menyadari banyaknya kerugian-kerugian yang dapat disebab-kan oleh *spyware* dan semakin banyaknya celah-celah yang dapat digunakan untuk menyusupkan *spyware*. Salah satu cara yang dapat digunakan untuk menangani *spyware* adalah dengan melakukan pencarian atau pengecekan terhadap file-file yang ada dalam sebuah sistem komputer. Namun, terdapat kendala dalam mengenali file yang ter-golong *spyware* bagi pengguna komputer yang awam. Oleh karena itu, untuk memudahkan hal tersebut maka melalui penelitian ini akan dibuat sebuah aplikasi penanganan *spyware* yang akan melakukan pengecekan terhadap sistem komputer. Adapun data-data mengenai jenis file-file yang termasuk *spyware* harus di-input ke database secara manual. Skema program secara umum dapat digambarkan pada gambar 1.

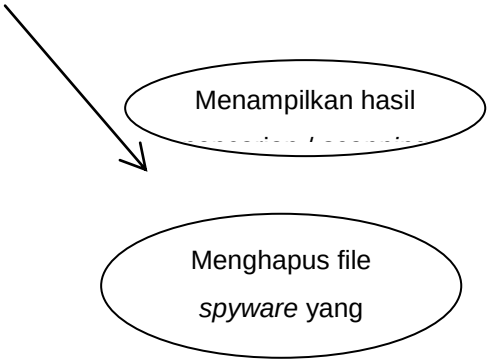


Gambar .1 Skema Alur Program Secara Umum

Rancangan Sistem Use Case Diagram

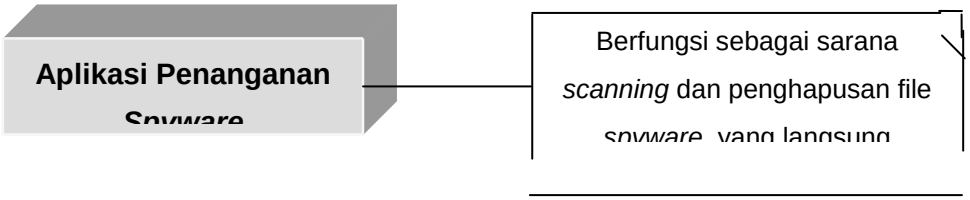


<<include>>



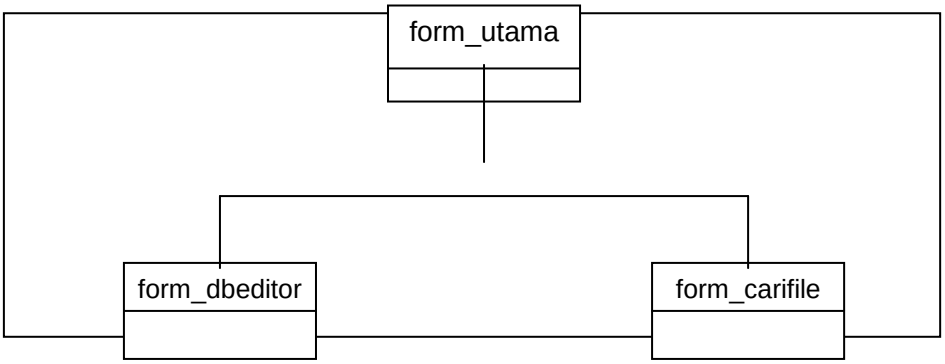
Gambar .2 Use Case Diagram.

Deployment Diagram



Gambar 3 Deployment Diagram

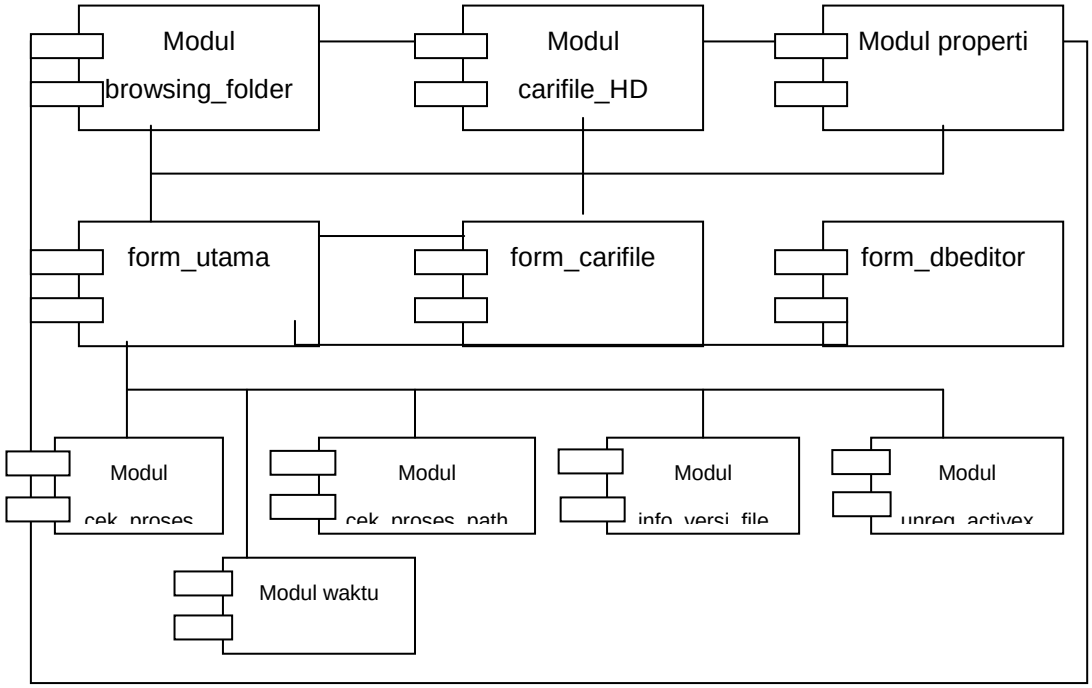
Class Diagram



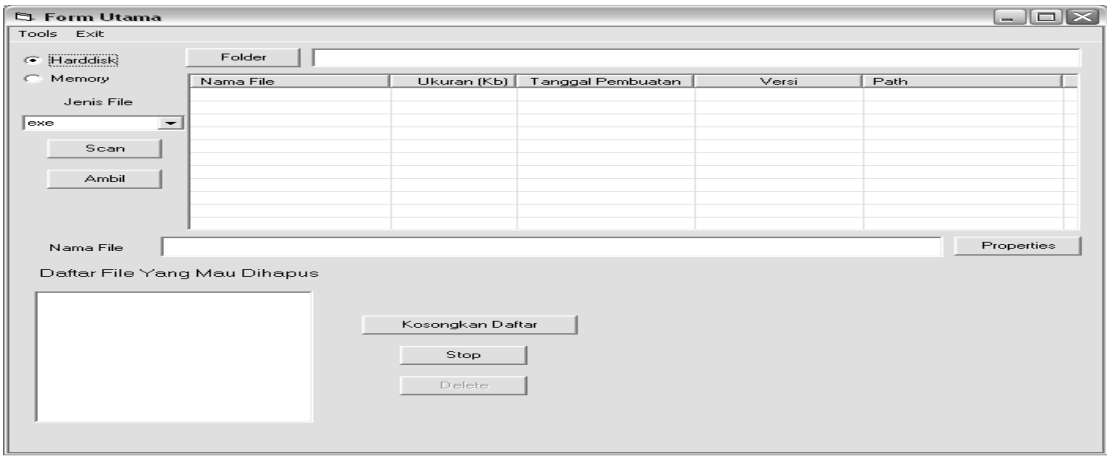
Gambar 4 Class Diagram

Gambar 6 merupakan form utama dari aplikasi yang dibuat pada penelitian ini. Dalam form inilah proses *scanning* dan penghapusan file *spyware* dilakukan. Proses *scanning* dapat dilakukan dengan terlebih dahulu memilih lokasi pencarian dan tipe file yang dicari. Di samping itu, dalam form ini terdapat dua menu, yaitu menu tools dan menu exit. Dalam menu tools terdapat submenu database editor dan pencarian file. Gambar 7 merupakan form pencarian file, di mana dalam form ini user dapat melakukan pen-carian file di *harddisk* dengan terlebih dahulu menginput lokasi pencarian dan kata kunci dari file yang ingin dicari.

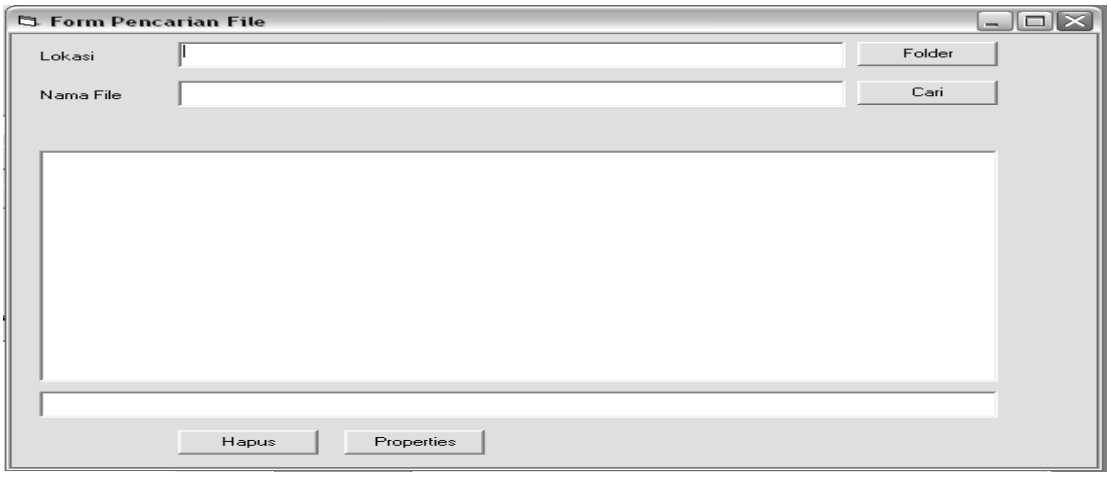
Component Diagram



Gambar 5 Component Diagram



Gambar 6 Tampilan Form Utama



Gambar 7 Tampilan Form Pencarian File

Implementasi Sistem

Tahap implementasi sistem merupakan tahap meletakkan sistem supaya siap untuk dioperasikan. Adapun daftar kebutuhan perangkat keras dan perangkat lunak untuk sistem ini adalah sebagai berikut:

a. Perangkat Keras (*hardware*)
   - 1 (satu) unit komputer dengan processor Pentium 4 2,4 GHz.
   - *Harddisk* Seagate 40 GB.
   - RAM 256 MB DDR
   - VGA : GeForce 2 Ti 64 MB.
   - Modem (optional).

b. Perangkat Lunak (*software*)

- Sistem operasi : Windows XP.
- Microsoft Visual Basic 6.0.

SIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan beberapa hal sebagai berikut:

b. Pentingnya Pemanfaatan Aplikasi ini karena mampu memberikan output berupa file-file yang tergolong *spyware* sesuai dengan yang terdapat di dalam *database*. Selain itu, aplikasi ini juga mampu melakukan penghentian proses dan penghapusan file *spyware*.

c. Pengujian aplikasi, baik dengan menggunakan metode *Black Box Testing*, maupun *White Box Testing*, dapat berjalan dengan baik. Tidak ditemukan kesalahan baik pada saat *compile* maupun saat *runtime*.

DAFTAR PUSTAKA

Fowler, M. 2004. *UML Distilled Edisi 3, Panduan Singkat Bahasa Pemodelan Standar*. Yogyakarta: ANDI.

Hariyanto, B. 2004. *Rekayasa Sistem Berorientasi Objek*. Bandung: INFORMATIKA.

Kristanto, A. 2004. *Rekayasa Perangkat Lunak*. Yogyakarta: Gava Media.

Kusumo, A. S. 2000. *Buku Latihan Microsoft Visual Basic 6.0.* Jakarta: PT Elex Media Komputindo.

Nugroho, A. 2005. *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*. Bandung: INFORMATIKA.

Pressman, R. S. 2002. *Rekayasa Perangkat Lunak Pendekatan Praktisi (buku 1)*. Yogyakarta: ANDI.

Saputra, J. 2005. *Eksplorasi Kekuatan WIN32-API dengan Visual Basic*. Jakarta: PT Elex Media Komputindo.

Utdirartatmo, F. 2005. *Ancaman Internet Hacking dan Trik Menanganinya*. Yogyakarta: ANDI.

Dharwiyanti, S. & Wahono, R. S. 2003. *Pengantar Unified Modeling Language (UML)*. http://ilmukomputer.com/umum/yanti-uml.php. 25 Agustus 2010.

Magdalena, M. 2003. *"Spyware", Ancaman Lain Setelah Virus*. http://www.sinarharapan.co.id/ berita/0508/11/ ipt01.html. 30 Juni 2010.

Miller, R. 2005. *Practical UML™: A Hands-On Introduction for Developers*. http://bdn.borland.com/ article/0,1410,31863,00. html, 25 Agustus 2010.