PEMBANGUNAN SOFTWARE PROTEKSI MENGGUNAKAN USB DRIVE

Suwastika Eka Putra[1], Fazmah Arief Yulianto[2], Eddy Muntina Dharma[3]

[1]Teknik Informatika, Fakultas Teknik Informatika, Universitas Telkom

Abstrak

Software atau perangkat lunak merupakan salah satu bidang di dunia teknologi informasi yang semakin berkembang dalam mencukupi kebutuhan otomatisasi atau komputerisasi bidang-bidang pekerjaan manusia. Banyak sekali vendor software atau pengusaha di bidang software yang mengembangkan produkproduknya secara rutin tentunya untuk mendapatkan kepercayaan dari penggunanya. Namun sangat disayangkan kerja keras dalam pembangunannya harus dibayar mahal karena sistem keamanan produk yang kurang memadai sehingga banyak sekali bertebaran produk-produk bersangkutan yang tanpa diketahui dengan jelas informasi penggunaan software. Untuk itu salah satu alternatif dalam memberikan keamanan produk software penelitian ini dilakukan.

Sistem kerja dalam penelitian ini adalah menyisipkan sebuah kode baru yang disebut dengan STUB ke file aplikasi (*.exe) yang dipilih oleh pengguna. Kode tersebut diambil dari no.seri USB drive yang merupakan perangkat kecil namun bisa digunakan untuk menyimpan data dan saat ini dengan mudah serta murah didapatkan di berbagai tempat penjualan perangkat keras komputer. Perangkat USB tersebut digunakan sebagai fasilitator validasi file aplikasi dengan kata lain jika file aplikasi yang bersangkutan tidak terdapat di USB Drive yang bersangkutan maka file aplikasi tersebut tidak dapat dijalankan.

Penelitian ini menghasilkan system proteksi yang mampu mengamankan software tanpa merusak informasi software tersebut, mengatasi software cracking, proses loading lebih cepat dari software sebelum di proteksi, ukuran file lebih kecil dari software sebelum di proteksi.

Kata Kunci : File Aplikasi, STUB, USB Drive, keamanan software.

Abstract

Software is one most of usefully in information technology to improvement otomation and computerity of human work necessity. At now there are a lot of developper in software whose called software vendor that develop any kind of software product. It's very routine to always improve and maintenance to take away user trusted. Unfortunately that hard work must be pay so expensive because one of problem that's product security. Many product have been distributed an illegal version and absolutely that's make a big suffer a financial loss. In this fact this research developed for give an altervative solution about software security.

The system work is doing inject new code that called STUB in executable file which user choosed. The code is extracted from serial number of USB Drive that a litte hardware but usefully for store data and now we can get it easy and cheaply in many hardware store place. The USB is used as facilitator executable file, in other word if executable not in this USB so that's file can not executed.

This thesis will produce a protecion system with the ability to protect software without damaging the information in the system, solving software cracking, improve the loading system speed before the protection, reducing software file size before the protection.

Keywords : Executable file, STUB, USB Drive, Software Security.

PENDAHULUAN

1.1 Latar belakang

Seiring dengan perkembangan teknologi, maka diikuti pula sarana penting yang mendukungnya, yaitu perangkat lunak serta perangkat keras komputer. Bersamaan dengan itu semakin banyak bermunculan para cracker yang dengan mudah merusak serta mengubah system keamanan dari suatu software. Hal ini tentu sangat tidak menyenangkan serta melanggar Hak Intelektual pembuat software tersebut. Sehingga meyebabkan kerugian baik secara materiil maupun secara intelektual.

Perkembangan seperti ini sangatlah merugikan banyak pihak, sehingga dibutuhkan suatu sistem keamanan yang dapat mengatasinya. Sistem keamanan yang dibutuhkan adalah sistem keamanan yang dapat memproteksi suatu software dari penggandaan serta pengubahan tanpa sepengetahuan maupun seizin pemiliknya atau pemegang lisensi.

Software proteksi ini nantinya dengan menyisipkan sebuah kode disebut STUB yang didalamnya terdapat nomor seri USB drive sebagai kunci proteksinya. Untuk menghindari membengkaknya ukuran file yang di proteksi maka nantinya juga akan di kompres sehingga memiliki ukuran yang tidak lebih besar dari ukuran file sebenarnya serta akan di lakukan proses enkripsi untuk bisa lebih mengamankan file aplikasi proteksi. Software ini tidak dapat diakses atau dijalankan jika USB yang merupakan kuncinya tidak terhubung pada PC dimana software tersebut akan dijalankan.

1.2 Perumusan masalah

Permasalahan yang dijadikan objek penelitian dan pengembangan pada tugas akhir ini adalah bagaimana kita mengamankan software berupa file executable (*.exe) yang akan dibungkus oleh aplikasi proteksi sehingga cracker ataupun hacker tidak dapat merubah atau menjalankannya. Selain itu aplikasi proteksi software ini juga diharapkan tahan terhadap serangan berupa: debugging, copying, breakpoint serta perubahan data binary dari pihak-pihak yang tidak berkepentingan.

Batasan masalah dalam tugas akhir ini adalah :

1. Komputer yang akan digunakan untuk memproteksi harus memiliki USB port.
2. Asumsi driver USB yang digunakan sebagai proteksi sudah terinstal.
3. Aplikasi hanya dapat memproteksi dari beberapa hacking tool, yaitu :
   - Regmon.
   - Filemon.
   - W32Dasm.
   - OllyDB.

4. File aplikasi yang bisa di proteksi merupakan file PE standar. Belum di *compress* atau di *encrypt* oleh software proteksi lainnya.

1.3 Tujuan

Adapun tujuan dalam tugas akhir ini adalah membangun sebuah sistem yang mampu :

1. Membuat suatu kode STUB yang mengambil nomor serial dari USB Storage untuk digunakan sebagai otentikasi atau validasi sistem proteksi.
2. Membuat sistem pengaman software dengan menyisipkan kode STUB ke dalam file aplikasi yang akan di proteksi serta melakukan proses kompresi dan enkripsi terhadap file aplikasi tersebut.
3. Mengamankan file aplikasi atau software tanpa merusak informasi yang ada pada software tersebut.
4. Melakukan pengujian kemampuan sistem proteksi menghadapi beberapa software yang digunakan sebagai cracking software seperti OllyDBG, RegMon, FileMon dan WinDASM32.

1.4 Metodologi penyelesaian masalah

Metode pemecahan masalah yang digunakan dalam pembuatan tugas akhir ini adalah :

- Studi literatur
  Mempelajari dasar teori mengenai proteksi, hacking, cracking suatu software.
- Analisis dan perancangan perangkat lunak
  Membuat analisis dan perancangan perangkat lunak dengan menggunakan analisa dan perancangan berbasis objek oriented.
- Implementasi perancangan perangkat lunak
  Mengimplementasikan perancangan perangkat lunak dalam Aplikasi Software proteksi menggunakan Serial number USB Drive sebagai kuncinya.
- Uji coba dan evaluasi sistem
  Menguji perangkat lunak yang telah dibuat untuk mengukur tingkat performansi dan keamananyanya, kemudian akan dilakukan analisis dan evaluasi dari hasil uji coba ini.
- Penyusunan Laporan Tugas Akhir.

2. PENGUJIAN DAN ANALISIS SISTEM

5.1 Kesimpulan

Berdasarkan hasil implementasi, analisis dan percobaan yang dilakukan dalam pembangunan tugas akhir ini, maka dapat disimpulkan bahwa sistem proteksi software dengan menggunakan USB Drive ini telah mampu :

1. Membangun STUB yang didalamnya terdapat nomor serial USB untuk otentikasi/ validasi sistem proteksi.
2. Melakukan penyisipan STUB ke file aplikasi yang akan di proteksi serta melakukan kompresi dan enkripsi sehingga ukuran file hasil proteksi memiliki ukuran yang lebih kecil daripada ukuran file aslinya.
3. Mampu mengamankan software tanpa merusak informasi software tersebut, sehingga software dapat berjalan sebagaimana mestinya dan memberikan informasi apa adanya dari software aslinya.
4. Mengatasi software-software yang di gunakan untuk cracking software. Proses cracking dengan bantuan tools software seperti RegMon,FileMon,WinDASM32 dan OllyDBG sudah terbukti tidak mampu merusak sistem ini.
5. Sistem proteksi ini berpengaruh terhadap waktu proses *loading* standar dari file aplikasi yang di proteksi yaitu terjadi penurunan dibandingkan waktu proses loading sebenarnya atau lebih cepat dibandingkan dengan waktu proses yang sebenarnya. Hal ini dikarenakan file aplikasi hasil proteksi memiliki besar ukuran file yang lebih kecil dibandingkan yang sebenarnya.
6. File aplikasi hasil proteksi ketika di jalankan membutuhkan alokasi memory proses yang lebih besar dibandingkan alokasi memory dalam kondisi belum di proteksi. Hal ini dikarenakan terjadi proses *Uncompress* ketika program yang sudah di proteksi dijalankan.

5.2 Saran

Dalam tugas akhir ini penulis berharap sistem yang dibangun dapat bermanfaat bagi pengguna terutama dalam hal memberikan keamanan produk-produk yang berupa software. Sistem yang penulis bangun ini tentunya belum sempurna sesuai dengan perkembangan teknologi informasi yang begitu cepatnya. Untuk itu penulis sarankan sistem ini nantinya dikembangkan berdasarkan perkembangan dunia informasi sehingga masih tetap handal dalam menangani masalah-masalah yang berhubungan dengan pembajakan terhadap kekayaan intelektual kita. Selain itu penulis juga menyarankan untuk membangun sistem proteksi yang bisa menangani masalah software yang berhubungan dengan jaringan komputer.

Daftar Pustaka

[1] Iclee_vx, *Scan Win32 API Functions*, Group F-13 Labs, 2005

[2] Jeff Duntemann, *Assembly Language : Step by Step*, John Wiley and Sons, Inc. 1992

[3] Konstantin Rozinov, *Reverse Code Engineering : An In Depth Analysis of The Bagle Virus*, Bell Labs – Government Communication Laboratory – Internet Research, August 2004

[4] Konstantin Rozinov, *PE File Infection Techniques Part 1*, Polytechnic University Scholarship for Service, February 2005.

[5] Kris Kaspersky, *Shellcoder's Programmming Uncovered*, A-List Publishing, 2005.

[6] Peter Szor, *Attack on Win32 – Part II*, Symantec (SARC Division) USA, 2000.

[7] Peter Szor, *The Art of Computer Virus Research and Defense*, Addision Wesley Publisher, 2005.

[8] Randall Hyde, *The Art of Assembly Language*, www.oreilly.com, 2003.

[9] Rohitab Batra, *BlackBat*, http://www.rohitab.com, 1999.

[10] Skape, *Undestanding Windows Shellcode*, www.nologin.org, 2003.

[11] S'to, *Pemrograman Dengan Bahasa Assembly Edisi Online Versi 1.0*, www.jasakom.com, 2001.