Statik Analiz
Statik analiz zararlıyı çalıştırmadan gerçekleştirilen analiz sürecidir. Hash kontrolü, disassembler yardımıyla opcode dizilerini incelenmesi, filetype incelenmesi, programın içinde bulunan stringlerin incelenmesi, PE headerları, importların ve dlllerin incelenmesi gibi süreçler statik analiz teknikleri içerisindedir. Zararlı sistemde çalıştırılmadığı için dinamik analizden çok daha güvenlidir.

Kod yapısını, fonksiyonunu anlamak için incelemek üzerinde yoğunlaşan bir analiz metodudur. Zararlı çalıştırılmadan gerçekleştirilir. (Heuristics & File Signatures)

- Online Araçlar (virustotal, anyrun, hybrid analysis...)
- Hash değerleri
- Dosyanın kendisinden bilgi edinmek

strings & **file** aracı

Packed - Obfuscated kavramları
Bu yöntemler “fonksiyonlara” ihtiyaç duyduğu için en azından LoadLibrary ve GetProcAddress API çağrılarını kullanırlar.

Paketlenmiş programlar (string ve diğer bilgileri açık) decompression için kullanılan daha küçük programlarla orijinal executable haline gelirler.

packed -> sıkıştırılmış
obfuscated -> gizlenmeye çalışılmış

.c - (compiler) > .o - (assembler) > .s - (linker) > çalıştırılabilir dosya

Linking Çeşitleri

Statik Linkleme: Çok az kullanılır. Uygulamanın boyutunu artırır. Unix ve Linuxta karşılaştığımız bir yöntemdir. Kütüphane çalıştırılabilir dosyanın içine kopyalanır.
Dinamik Linkleme: Uygulama çalıştırıldığında kütüphanelerin linklenmesidir.
Runtime’da Linkleme: Zararlılar tarafından çokça kullanılır. Kütüphaneler “gerektikçe”, kullanıldıkça linklenir.

API çağrılarını tanımak önemlidir. (MSDN)

PE Veri Yapısı
Windows çalıştırılabilir dosyaları, obje kodları, DLLler…
İşletim sisteminin loaderının çalıştırılabilir dosyayı çalıştırmak için ihtiyacı olan bilgilere sahiptir. (kod, uygulama tipi, gerekli kütüphaneler, alan bilgileri)

-DOS Header: ilk 64 byte, magic number, MZ-> 4D5A (çalıştırılabilir dosya), son dört byte'ı offset değeri

-DOS Stub: program işletim sistemi ile uyumlu değilse “This program cannot be run DOS mode” mesajını yazdırır

-PE File Header:
--Image File Header: ilk 20 byte

Offset	Name	Value	Meaning
F4	Machine	8664	AMD64 (K8)
F6	Sections Count	6	6
F8	Time Date Stamp	4a5bc9d4	Pazartesi, 13.07.2009 23:57:08
FC	Ptr to Symbol Table	0	0
100	Num. of Symbols	0	0
104	Size of OptionalHeader	f0	240
106	Characteristics	22	
		2	File is executable (i.e. no unre
		20	App can handle >2gb address

--Image Optional Header:

Magic	10B
Linker Ver. (Major)	6
Linker Ver. (Minor)	0
Size of Code	1000
Size of Initialized Data	26000
Size of Uninitialized Data	0
Entry Point	12FA
Base of Code	1000
Base of Data	2000
Image Base	10000000
Section Alignment	1000
File Alignment	1000
OS Ver. (Major)	4
OS Ver. (Minor)	0
Image Ver. (Major)	0

-Section Table:
IMAGE_SECTION_HEADER

Disasm | General | DOS Hdr | Rich Hdr | File Hdr | Optional Hdr | Section Hdrs

Name	Raw Addr.	Raw size	Virtual Addr.	Virtual Size	Characteristics	Ptr
.text	1000	3000	1000	295E	60000020	0
.rdata	4000	1000	4000	8CA	40000040	0
.data	5000	1000	5000	7FC	C0000040	0

.text: CPU komutları, çalıştırılabilir kod içeren tek section (olmalı)
.rdata: import export bilgileri, read-only veriler, (.i/edata)
.data: global veriler
.rsrc: kaynaklar

.text bölümünün Virtual Address değeri Raw Address değerinden büyük ise dosyanın paketlenmiş olduğu düşünülebilir.

-Data Dictionaries: IAT, import export fonksiyon yerleri

Önemli DLLler ve kullanım yerleri:

KERNEL32.DLL->Bellek yönetimi, I/O (Girdi/Çıktı) operasyonları, senkronizasyon, process oluşturma, process interruptlama.

GDI32.DLL->Grafik Araç Arayüzü, ilkel çizim fonksiyonları.

USER32.DLL->Sistemin USER (kullanıcı) bileşenini kullanarak Windows'un standart kullanıcı arayüzünü manipüle etme (masaüstü, pencereler, menü).

COMCTL32.DLL->Dosya Aç, Kaydet, Olarak Kaydet gibi Windows kontrollerini kullanıcıya sunma.

ADVAPI32.DLL->Windows Registrysini etkileyen güvenlik fonksiyonları ve çağrıları.

OLE32.DLL->Komponent Obje Modeli (Component Object Model), Obje Linkleme ve Gömme (Object Linking and Embedding)

NETAPI32.DLL->Query (sorgu) ve ağ arayüzlerini yönetme

COMDLG32.DLL->Diyalog kutucuğu oluşturma

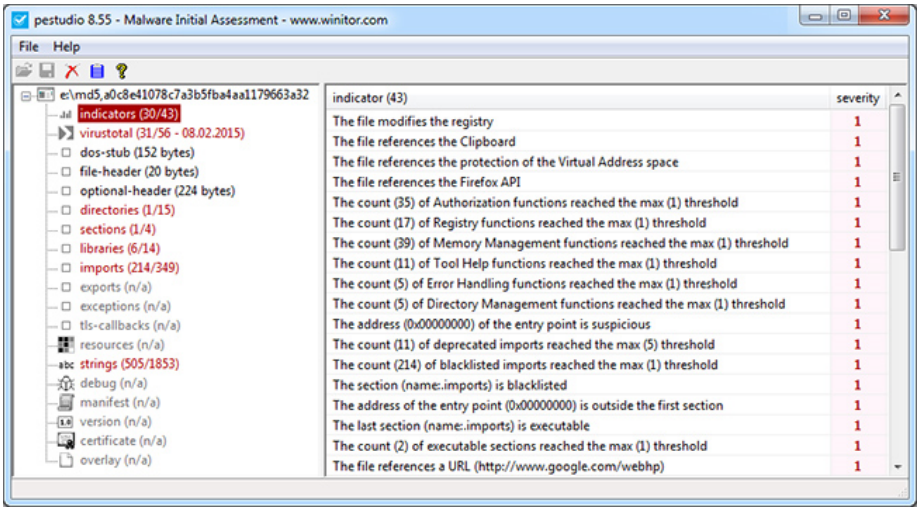
WS2_32.DLL->Winsock API (TCP/IP ağ fonksiyonları, diğer ağ APIlarıyla kısmi uyumluluk)

WININET.DLL->FTP ve HTTP protokoller

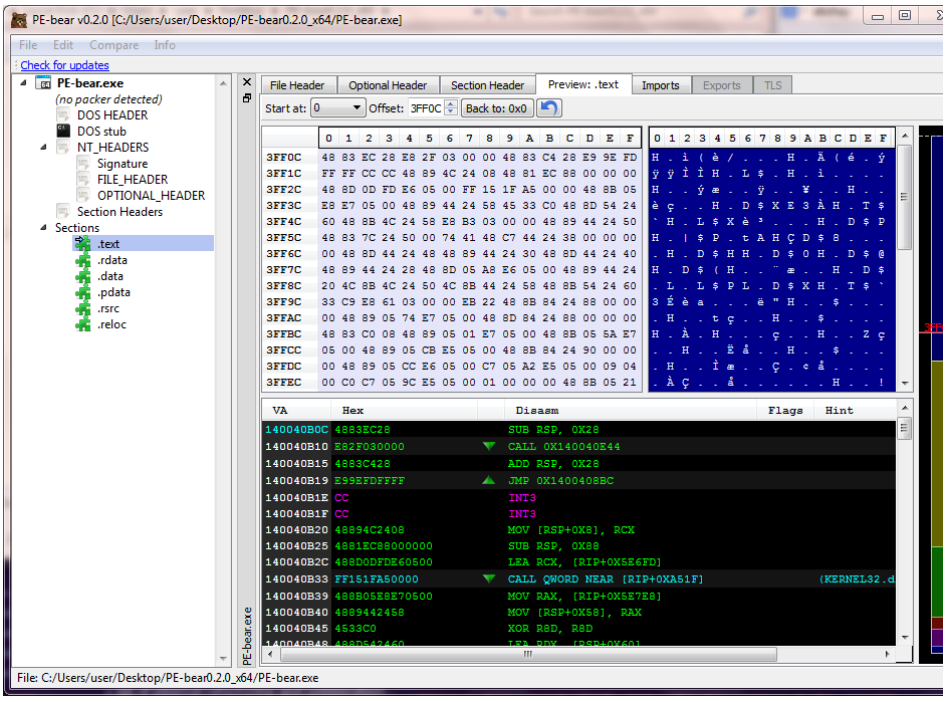
VS tarafından kullanılan bazı dlller:MSVCRT.DLL, MSVCPxx.dll, MSVBVM60.DLL, VCRUNTIMExx.DLL

Statik Analiz Araçları:

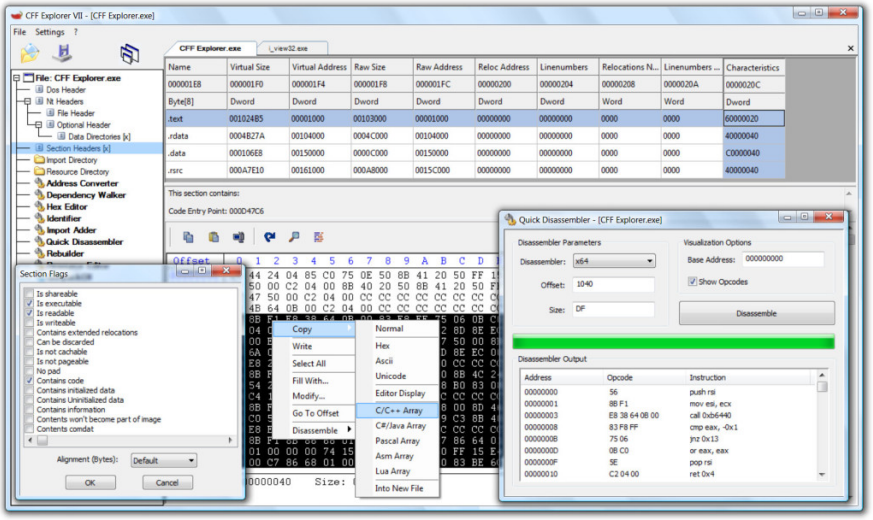
PEStudio



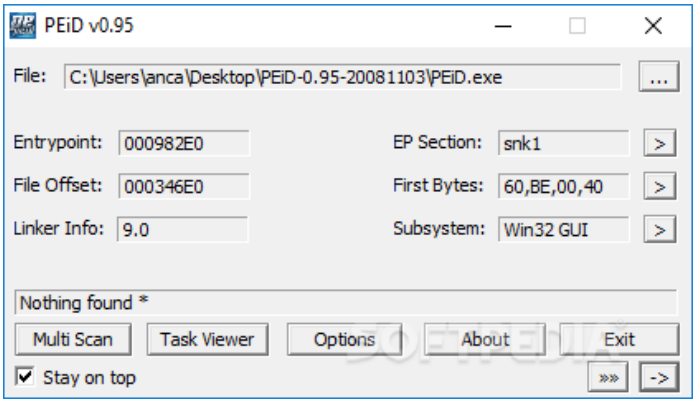
PEBear



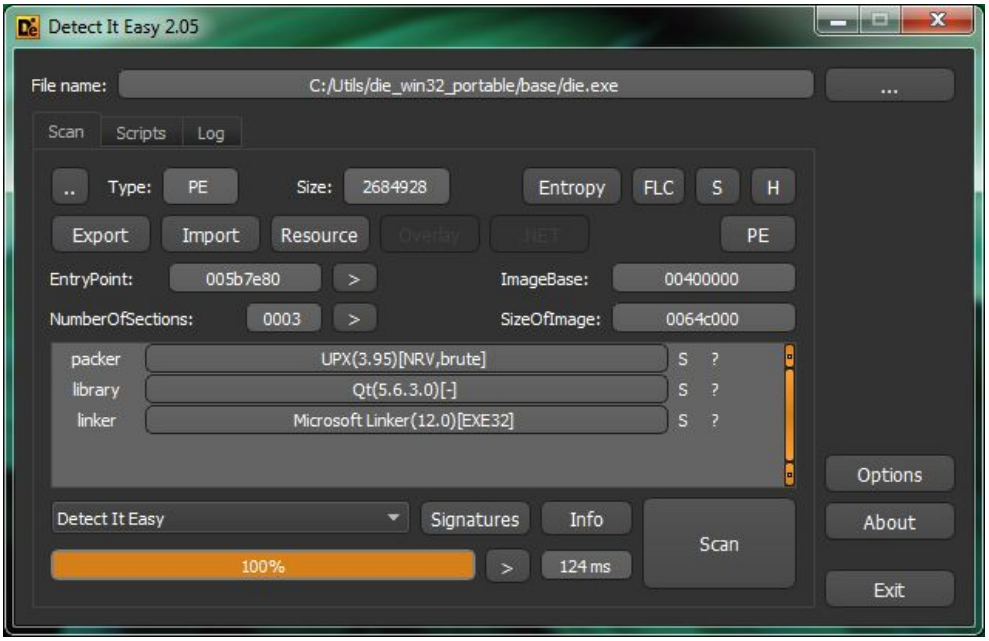
CFFExplorer



PEiD



DiE



PEView