ANTI

ZARARLI YAZILIM ANALİZİ

BEN KİMİM?

Ahlaklı Korsan (E.H)	**Mesai saatlerinde...**
Zararlı Yazılım Analisti	**Boş zamanlarımda...**
Blog Yazarı	**http://www.mertsarica.com**
Güvenlik TV	**http://www.guvenliktv.org**
Python Programcısı	**http://www.mertsarica.com/programlar**
Sertifika Kolleksiyoncusu	**CISSP , SSCP , OSCP , OPST , CREA**

MESLEĞİM ?

NBG Grup şirketlerinden Finansbank'ın Bilgi Teknolojileri iştiraki olan IBTech firmasında Bilişim Güvenliği Uzmanı (Senior Penetration Tester / Ethical Hacker) olarak çalışmaktayım.

http://www.finansbank.com.tr

http://www.ibtech.com.tr

İÇERİK

Neden zararlı yazılım analizi ?

Analiz adımları

Anti analiz yöntemleri

Uygulamalı analiz

Sonuç

GÜNCEL TEHDİTLER

Avrupa

Atatürk Hav

İbrahim YILDIZ/İSTANB

Atatürk Havaliman terminalinde pasa kilitlendi. Sisteme iddia edildi. Saatle yolcular 08.00 itib başladı.

İSTANBUL İl Emniye meydana gelen arı Havalimanı'nda bir önünde uzun kuyru

Sabah 06.30 sıralar ve Sabiha Gökçen H Havalimanı ve Sabil

Arıza nedeniyle bir normale dönmesini

Bu arada yolcuların

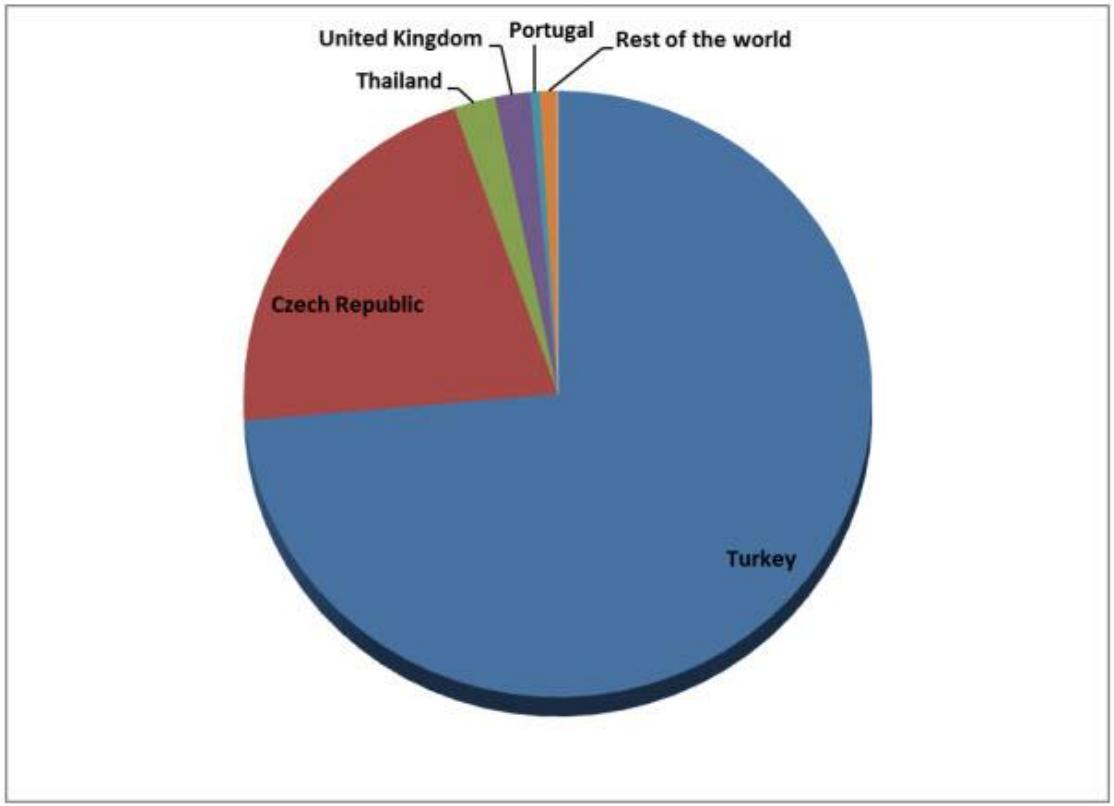


Figure 8 - Detection statistics of Win32/Hesperbot according to ESET LiveGrid

le Atatürk

rdi. Sistemin

havalanabildi.

YANILGILAR

Zararlı yazılım analizi yapmak için zararlı yazılım analisti olmanız gerekmektedir. (İleri seviye hariç)

Gostev says that because of its size and complexity, complete analysis of the code may take years.

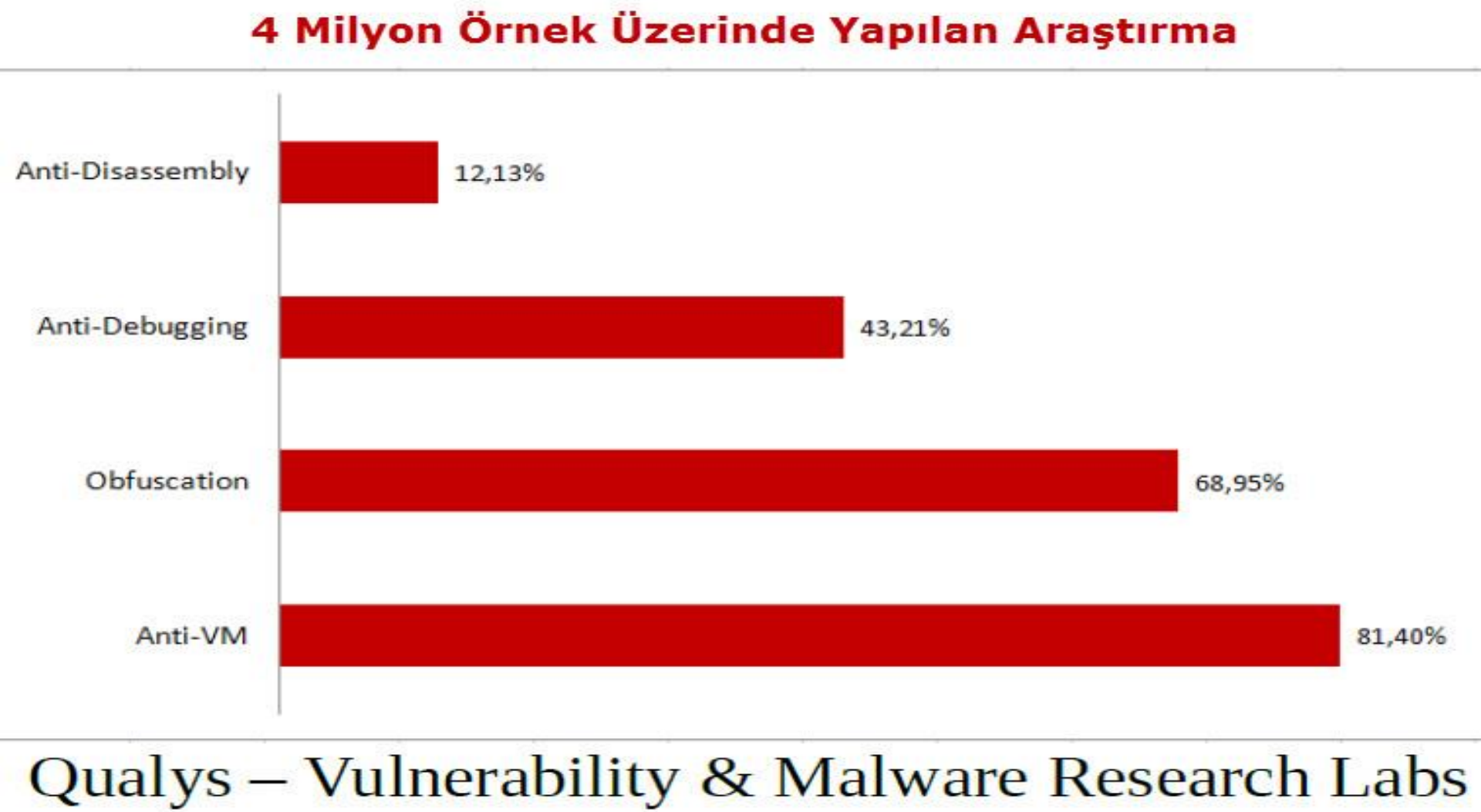
Zararlı yazılım analizi, zararlı yazılım ile ilgili olan tüm bilgileri elde etmektir. (Flame)

analyze Stuxnet," he said. "This is 20 times more complicated. It will take us 10 years to fully understand everything."

Assembly ile program yazamayan zararlı yazılım analizi yapamaz.

Güvenlik cihazları, yazılımları ve OS yamaları güncel olduğu sürece zararlı yazılım sistemlere bulaşamaz!

BİLMENİZ GEREKENLER



Qualys – Vulnerability & Malware Research Labs

3 ANALİZ ADIMI

Davranışsal Analiz: Zararlı yazılımın ağ, dosya sistemi ve kayıt defteri (registry) üzerindeki davranışları izlenir.

(Process Monitor, Process Explorer, Regshot, Wireshark, CaptureBat, Cuckoo Sandbox vb.)

Kod Analizi (Statik & Dinamik): Statik analizde zararlı yazılım sistem üzerinde çalıştırılmadan, dinamik analizde ise debugger ile sistem üzerinde çalıştırılıp çeşitli araçlar ile analiz edilir.

(S: Strings, IDA, Dependency Walker, PEiD, JAD, Reflector vb.)
(D: IDA, Ollydbg, Immunity Debugger, Windbg vb.)

Bellek Analizi: Zararlı yazılım çalıştırılıp hafızada çalışan kopyası diske kopyalanıp analiz edilir.

(Volatility, Memoryze, Redline)

ANALİST BEZDİRME ADIMLARI

Anti Debugging: Debugging işleminin zorlaştırılması/engellenmesi sağlanır.

Anti Disassembly: Disassembler yanıltarak, kodun yorumlanması, analiz edilmesi zorlaştırılır/engellenir.

Anti VM: Programın sanal makinede çalışıp çalışmadığı tespit edilir. (FatMal)

Gizleme (Obfuscation): Bayt kodların kaynak koduna çevrilmesi ve analiz edilmesi zorlaştırılır/engellenir.

Paketleme: Statik kod analizini zorlaştırma adına zararlı yazılımın sıkıştırılmasıdır.

ANTI DEBUGGING

API Yöntemi: Çeşitli APIler kullanılarak debugger tespit edilir. (IsDebuggerPresent(), CheckRemoteDebugger(), vs)

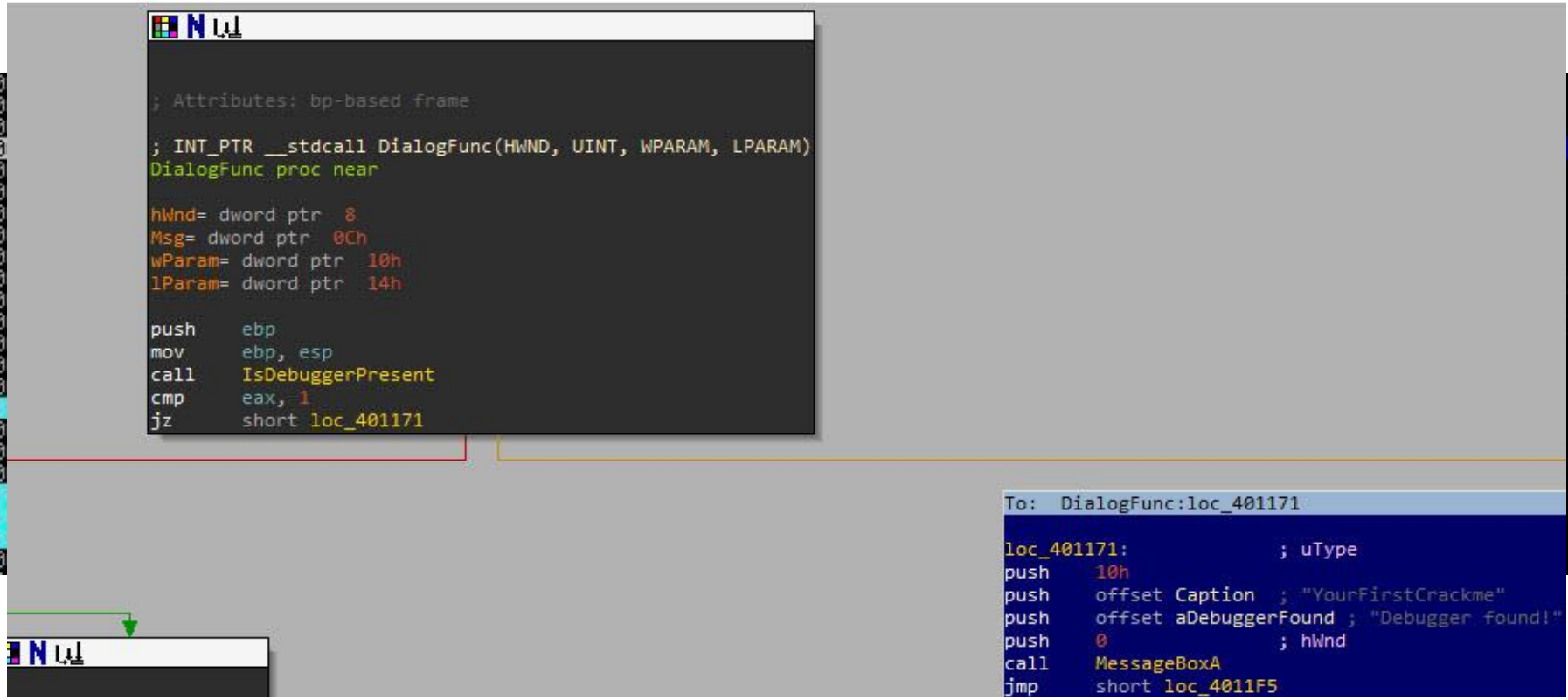
İstisnai Durum Yöntemi: Debugger'ın devam ettiremeyeceği hatalar üretilerek debugging işlemi durdurulur. (SEH, INT3, INT1, INT2D)

PEB Yöntemi: API yardımı olmadan debugger tespit edilir. (CheckRemoteDebugger() yerine PEB -> BeingDebugged)

Breakpoint Yöntemi: 0xCC (INT3) baytı aranarak ve/veya istisna oluşturularak debug registerları ile debugger tespit edilir.

Zamanlama Yöntemi: Programın çalışması esnasında geçen süre hesaplanarak debugger tespit edilir. (RDTSC, QueryPerformanceCounter(), GetTickCount())

ANTI DEBUGGING



ANTI ANTI DEBUGGING

Adım adım dinamik kod analizi (debugging) yapabilirsiniz.

Anti-debug eklentileri kullanabilirsiniz.

- **IDA Pro eklentisi: IDAStealth** (http://newgre.net/idastealth)
- **Immunity Debugger eklentileri: Phant0m, HideDebug, HideOD** (http://woodmann.com/BobSoft/Pages/Plugins/ImmDbg)

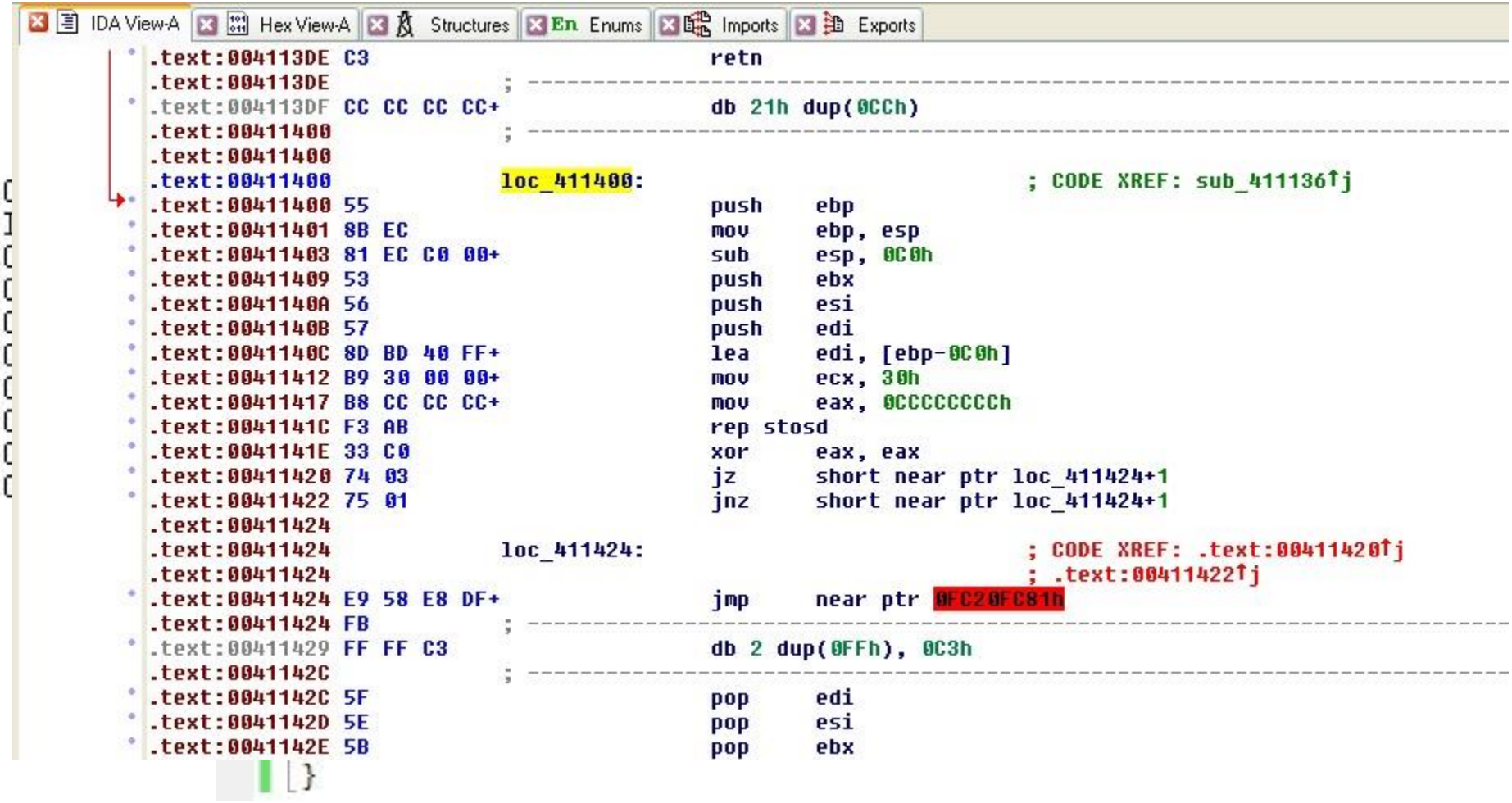
ANTI DISASSEMBLING

Disassembler algoritmalarındaki zafiyetler kötüye kullanılarak analizin zorlaştırılması, engellenmesidir.

Linear Sweep Algoritması: Kod ile veri ayrımı yapamayan bu algoritmada, her bayt teker teker yorumlanır dolayısıyla verinin kod olarak yorumlanması hatalı yorumlamaya yol açar.

Recursive Traversal Algoritması: Kod, programın akışına göre, bloklar halinde yorumlanır. Veri ile kod ayrımı yaptığı için kandırılması biraz daha zordur.

ANTI DISASSEMBLING



ANTI ANTI DISASSEMBLING

Adım adım kod analizi yapabilirsiniz.

- **IDA Pro:** **c tuşu (data -> code) , d tuşu (code -> data)**
- **Immunity Debugger:** **Analysis -> Remove analysis**

Birden fazla disassembler ve debugger aracı kullanılabilirsiniz.

ANTI VM

Zararlı yazılımın sanal makinede çalışıp çalışmadığını kontrol edilir.

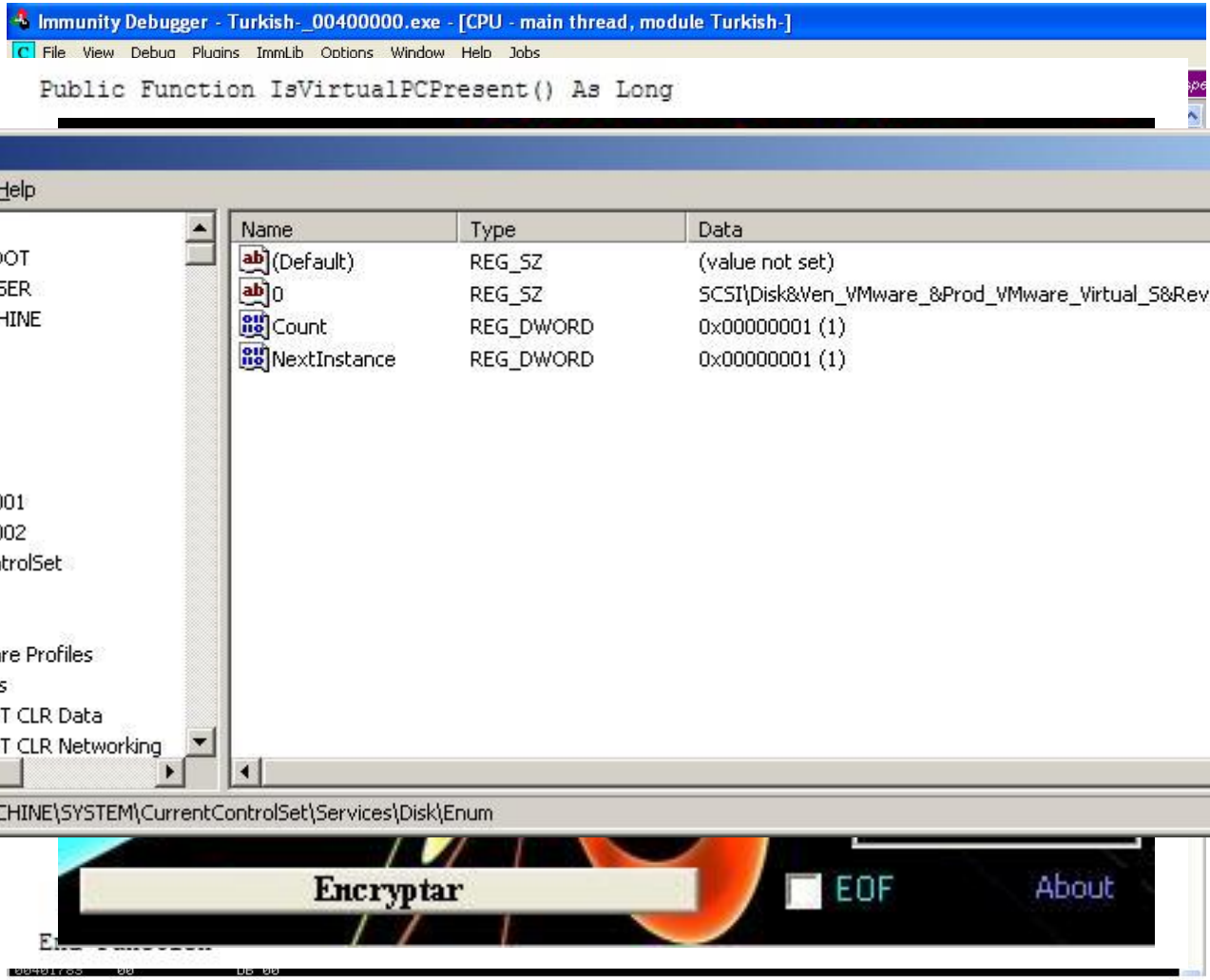
Dizin kontrolü: **Sanal makineye ait yardımcı uygulama dizinleri tespit edilir. (C:\Program Files\VMware\VMware Tools)**

Program kontrolü: **Sanal makine yazılımına ait programlar tespit edilir. (VMwareService.exe, VboxService.exe vs.)**

Kayıt defteri (registry) kontrolü: **Sanal makine yazılımına ait kayıtlar tespit edilir. (** [HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Class\{4D36E968-E325-11CE-BFC1-08002BE10318}\0000\Settings] "Device Description"="VMware SVGA II"**)**

Komut kontrolü: **Komut çıktılarından sanal makine tespit edilir. (sidt (red pill), I/O com port (0x564D5868 (VMXh) vs.)**

ANTI VM


Immunity Debugger - Turkish-_00400000.exe - [CPU - main thread, module Turkish-]
File View Debug Plugins ImmLib Options Window Help Jobs
Public Function IsVirtualPCPresent() As Long
Registry Editor
File Edit View Favorites Help
My Computer
HKEY_CLASSES_ROOT
HKEY_CURRENT_USER
HKEY_LOCAL_MACHINE
HARDWARE
SAM
SECURITY
SOFTWARE
SYSTEM
ControlSet001
ControlSet002
CurrentControlSet
Control
Enum
Hardware Profiles
Services
.NET CLR Data
.NET CLR Networking
Name
Type
Data
(Default)
REG_SZ
(value not set)
0
REG_SZ
SCSI\Disk&Ven_VMware_&Prod_VMware_Virtual_S&Rev_1.0\4&5fcaafc&0&000
Count
REG_DWORD
0x00000001 (1)
NextInstance
REG_DWORD
0x00000001 (1)
My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Disk\Enum
Encryptar
EOF
About

ANTI ANTI VM

Adım adım dinamik kod analizi yapılabilirsiniz.

Yardımcı sanal makine uygulamalarını kaldırabilirsiniz. (VMWare Tools, VirtualBox Guest Additions vs.)

VMWare için vmx ayar dosyasına aşağıdaki satırları ekleyebilirsiniz.

monitor_control.restrict_backdoor = "TRUE"	monitor_control.disable_chksimd = "TRUE"
monitor_control.virtual_rdtsc = "false"	monitor_control.disable_ntreloc = "TRUE"
monitor_control.restrict_backdoor = "true"	monitor_control.disable_selfmod = "TRUE"
isolation.tools.getPtrLocation.disable = "true"	monitor_control.disable_reloc = "TRUE"
isolation.tools.setPtrLocation.disable = "true"	monitor_control.disable_btinout = "TRUE"
isolation.tools.setVersion.disable = "true"	monitor_control.disable_btmemspace = "TRUE"
isolation.tools.getVersion.disable = "true"	monitor_control.disable_btpriv = "TRUE"
monitor_control.disable_directexec = "true"	monitor_control.disable_btseg = "TRUE"

GIZLEME (OBFUSCATION)

Java, .Net (C#, VB) gibi interpreted diller kaynak koduna çevrilebilmektedir.

Gizleme yöntemi ile kaynak koduna çevrilen yazılımın analiz edilmesi engellenmeye çalışılır.

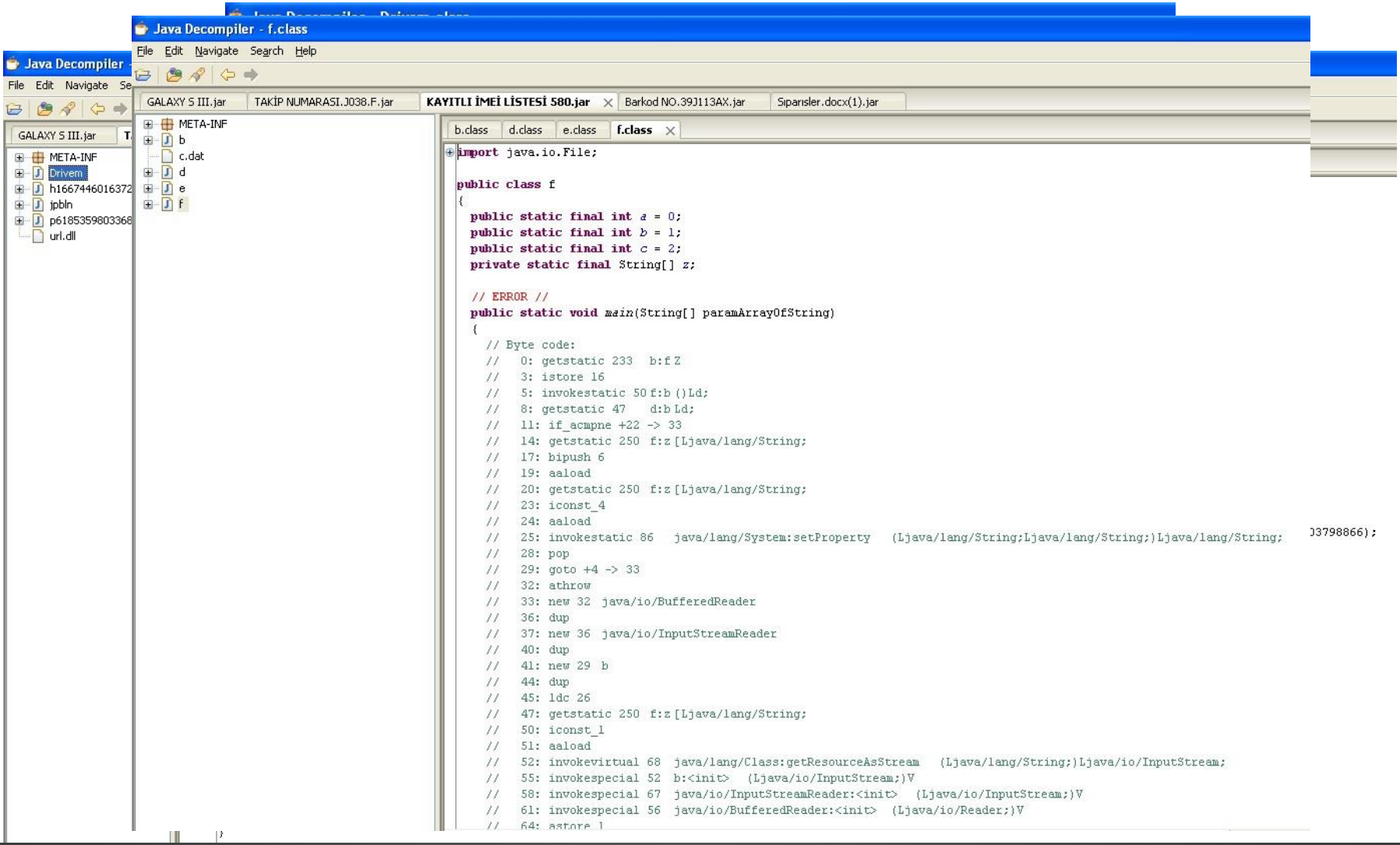
Yüzeysel gizleme: Değişken ve fonksiyon isimlerinin değiştirilmesi, karakter dizilerinin (string) şifrelenmesi

Derin gizleme: Yazılımcının bildiği fakat program çalışana dek (debug) disassemblerın bilemediği ifadeler (opaque predicates)

```
mov     eax, ebx
push    eax
pop     esp
call    GetLastError
add     eax, 401034h
jmp     eax
```

GetLastError == 0x57

GIZLEME (OBFUSCATION)



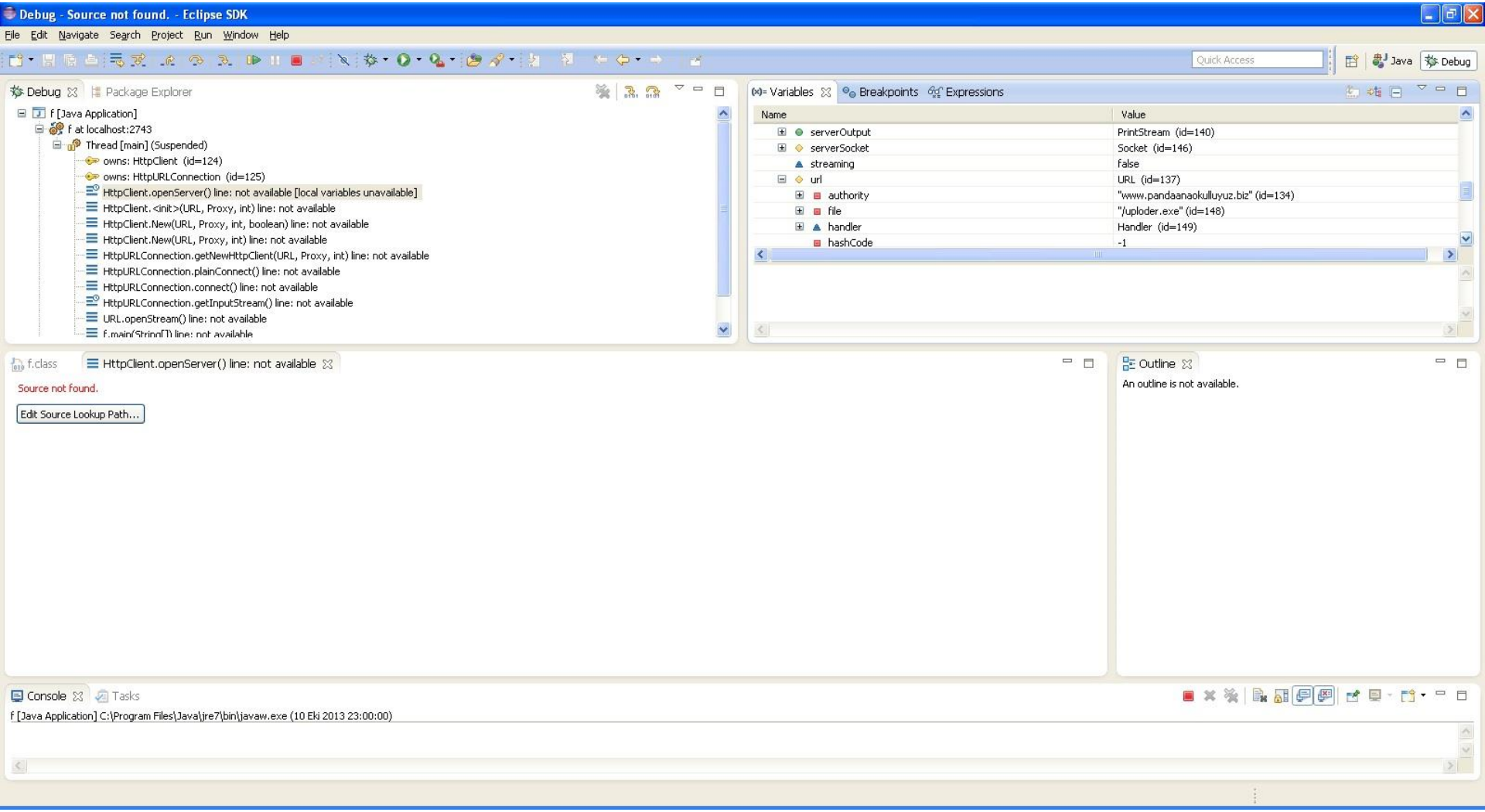
ANTI GIZLEME (OBFUSCATION)

Adım adım dinamik kod analizi (bytecode debugging) yapılabilirsiniz.

- **Java için Eclipse eklentisi olan Bytecode Visualizer aracını kullanabilirsiniz. (**http://www.drgarbage.com/bytecode-visualizer.html**)**
- **.Net için Reflector eklentisi olan Deblector aracını kullanabilirsiniz. (**http://deblector.codeplex.com/**)**

De4dot vb. deobfuscator araçlarından faydalanabilirsiniz. (https://bitbucket.org/0xd4d/de4dot/**)**

ANTI GIZLEME (OBFUSCATION) - JAVA



PAKETLEME

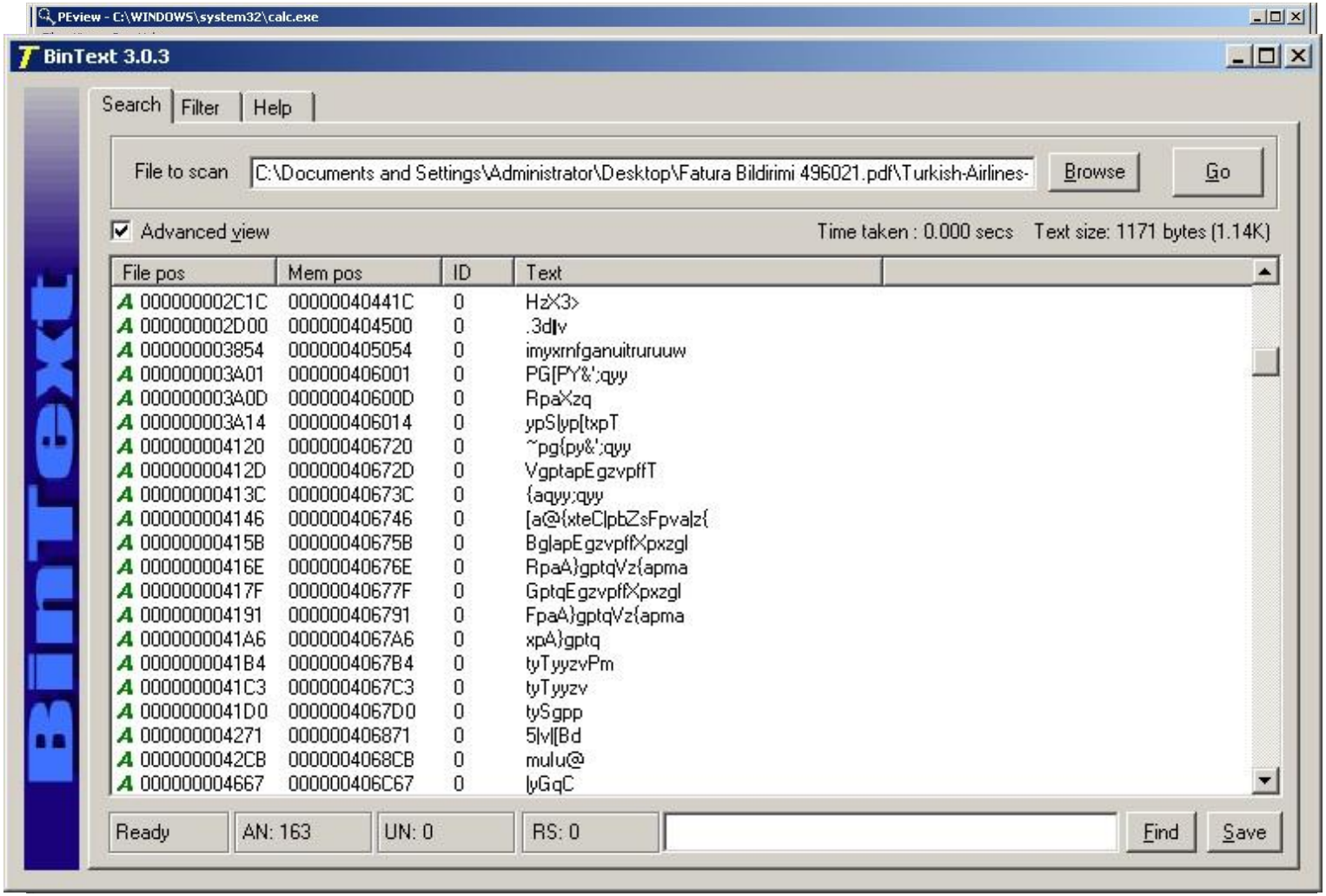
Yazılımın çalıştırılmadan (statik) assembly seviyesinde analiz edilmesi zorlaştırılır.

Orjinal yazılımın Import/Export tabloları, karakter dizileri (strings) gizlenir.

Yazılımın boyutu küçültülür ve antivirüs yazılımlarından gizlenmesi sağlanır.

Popüler paketleme araçları: UPX, ASPack, ASProtect, Armadillo vs.

PAKETLEME


PEview - C:\WINDOWS\system32\calc.exe
BinText 3.0.3
Search
Filter
Help
File to scan
C:\Documents and Settings\Administrator\Desktop\Fatura Bildirimi 496021.pdf\Turkish-Airlines-
Browse
Go
Advanced view
Time taken : 0.000 secs
Text size: 1171 bytes (1.14K)
File pos
Mem pos
ID
Text
A 000000002C1C 00000040441C 0 HzX3>
A 000000002D00 000000404500 0 .3dlv
A 000000003854 000000405054 0 imyxrnfganuitruruuw
A 000000003A01 000000406001 0 PG[PY&';qyy
A 000000003A0D 00000040600D 0 RpaXzq
A 000000003A14 000000406014 0 ypSlyp[txpT
A 000000004120 000000406720 0 ~pg{py&';qyy
A 00000000412D 00000040672D 0 VgptapEgzvpffT
A 00000000413C 00000040673C 0 {aqyy;qyy
A 000000004146 000000406746 0 [a@{xteClpbZsFpvalz{
A 00000000415B 00000040675B 0 BglapEgzvpffXpxzgl
A 00000000416E 00000040676E 0 RpaA}gptqVz{apma
A 00000000417F 00000040677F 0 GptqEgzvpffXpxzgl
A 000000004191 000000406791 0 FpaA}gptqVz{apma
A 0000000041A6 0000004067A6 0 xpA}gptq
A 0000000041B4 0000004067B4 0 tyTyyzvPm
A 0000000041C3 0000004067C3 0 tyTyyzv
A 0000000041D0 0000004067D0 0 tySgpp
A 000000004271 000000406871 0 5lvl[Bd
A 0000000042CB 0000004068CB 0 mulu@
A 000000004667 000000406C67 0 lyGqC
Ready
AN: 163
UN: 0
RS: 0
Find
Save
BinText

ANTI PAKETLEME

Adım adım dinamik kod analizi (debugging) yaparak hafıza açılmış yazılımı diske kaydedip analiz edebilirsiniz.

- **Debug ederek OEP bulunur.**
- **OEP'te hafızada çalışan yazılım diske kaydedilir. (dump)**
- **IAT, ImpREC aracı ile düzeltilir.**

PeiD ve RDG Packet Detector araçları ile paketleme aracını tespit ettikten sonra FUU vb. unpacker araçları ile pakedi açabilirsiniz. (https://code.google.com/p/fuu/**)**

VAKA

VAKA

> From: bilgi@kvk.com
> To: [redacted]@hotmail.com
> Subject: MERHABA SIPARISLERINIZ KARGOYA VERILMISTIR...
> Date: Mon, 4 Feb 2013 06:10:43 +0200
>
>
> SIPARISINIZ KARGOYA VERILMISTIR.
> KARGO TESLIMATI VE FATURA ILE ILGILI BILGILERIN BIRER
> NUSHASI KARGO ILE GONDERILMISTIR
>
> KVK TEKNOLOJI URUNLERI VE TIC A.S
> ADRES: BAYAR CAD GULBAHAR SOK KVK PLAZA NO:14 KOZYATAGI/KADIKOY/ISTANBUL
>
> MUHASEBE SORUMLUSU
> FATMA CUKUR
>

GALAXY S III.jar
5K Download

From: info@yurticikargo.com
To [redacted]@hotmail.com
Subject: yurticikargo.com 1 ADET KARGONUZ VAR
Date: Tue, 16 Jul 2013 03:07:32 +0300

KARGONUZ EN GEC 1 İŞ GUNU İCİNDE TESLİM OLACAKTIR KARGO GÖNDERİ TAKİP NUMARASI İLE KARGONUN İÇERİĞİNİ ÖĞRENEBİLİRSİNİZ Çağrı Merkezi (444 99 99 Ana Sayfa > Bize Ulaşın > Çağrı Merkezi (444 99 99) Çağrı merkezimizden Pazar günleri hariç saat 08.00 ile 24.00 saatleri arasında aşağıdaki hizmetlerimizi alabilirsiniz.

TAKİP NUMARASI.J038.F.jar
10K Download

DEMO

SONUÇ

Zararlı yazılım analiz becerisine artık hemen hemen her kurum sahip olmalıdır, analist yetiştirmek, eğitmek için geç kalmayın.

Kurumunuz için ağ üzerinden zararlı yazılım tespit edebilen cihazlardan temin edin.

Sızma testi ve zafiyet analizi ile ağınıza ve sistemlerinize sızılabilecek noktaları tespit edin.

Bilgi güvenliği farkındalık eğitimlerine daha çok önem verin.

Zararlı yazılım geliştiricileri hangi araçları, sistemleri ve yöntemleri kullandığınızı çok iyi biliyorlar, bol bol okuyun ve pratik yapın.

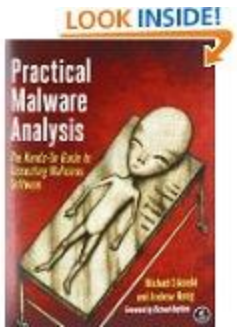
OKUNASI KİTAPLAR

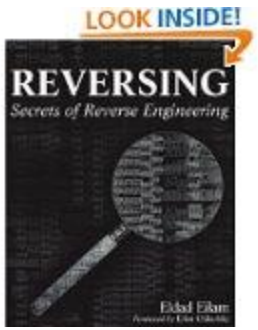
Malware Analyst's Cookbook

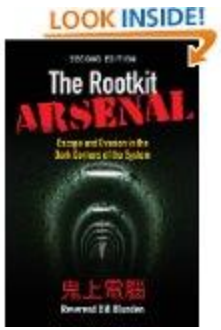
Practical Malware Analysis

Secrets of Reverse Engineering

The Rootkit Arsenal

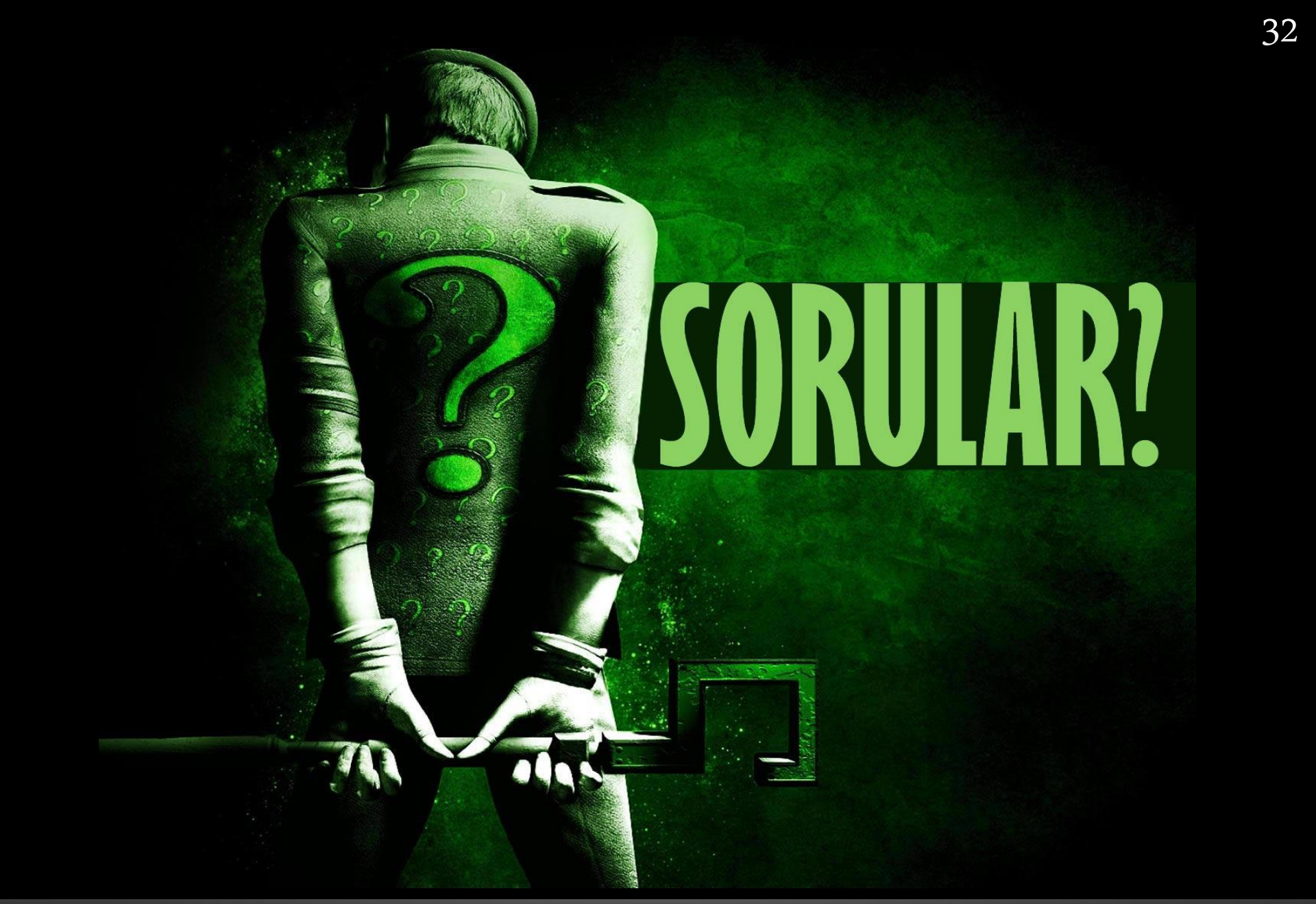






EĞİTİMLER


Eğitim İçeriği
SANS
Find Training
Live Training
Online Training
Programs
FOR610: Reverse-Engineering Malware: Malware Analysis Tools and Techniques
Contents | Additional Info
Delivery Methods:
Live | Online
GREM Certification
30 CPE/CMU
Laptop Required
This popular malware analysis course has helped forensic investigators, incident responders and IT administrators acquire practical skills for examining malicious programs that target Microsoft Windows. This training also teaches how to reverse-engineer Web browser malware implemented in JavaScript and Flash, as well as malicious documents, such as PDF and Microsoft Office files. The course builds a strong foundation for reverse-engineering malicious software using a variety of system and network monitoring utilities, a disassembler, a debugger and other tools for turning malware inside-out.
Learn Reverse-Engineering Malware: Promo Trailer
0:00 / 0:58
The malware analysis process taught in this class helps incident responders assess the severity and repercussions of a situation that involves malicious software and plan recovery steps. Forensics investigators also learn how to understand key characteristics of malware discovered during the examination, including how to establish indicators of compromise (IOCs) for scoping and containing the incident.


SORULAR?

TEŞEKKÜRLER