FOR610 – Reverse-Engineering Malware

Topics

Function

- AllocateVirtualMemory 5–26, 4–109
- BlockInput 5–76, 5–92
- CheckRemoteDebuggerPresent 5–12
- CloseClipboard 2–144
- CreateMutex 2–143
- CreateProcess 5–51, 5–54, 2–85
- CreateRemoteThread 4–108, 4–113, 4–123
- CreateThread 4–109
- CreateToolHelp32Snapshot 2–61, 5–85
- CreateUserThread 4–109
- CryptDecrypt 4–90-92
- EnumProcesses 4–108
- FindResource 2–140
- FindWindow 5–79
- FindWindows 5–92
- GetAsyncKeyState 2–144, 2–175
- GetClipboardData 2–144
- GetCursorPos 5–66
- GetForeGroundWindow 5–66
- GetKeyState 2–144
- GetLocalTime 5–13
- GetModuleHandle 5–77, 5–92
- GetProcAddress 5–53
- GetSystemTime 5–13
- GetTempFileName 2–61
- GetTempPath 2–121
- GetTickCount 5–13, 5–66
- GetWindowText 2–144
- HttpAddRequestHeaders 2–45
- HttpOpenRequest 2–45
- HttpSendRequest 2–45
- InternetConnect 2–45
- InternetOpen 2–45, 2–48
- InternetReadFile 2–45
- IsDebuggerPresent 5–6
- KdDebuggerEnabled 5–92
- LoadLibrary 5–53, 4–123, 5–146
- LoadResource 2–140
- LockResource 2–140
- OpenClipBoard 2–144
- OpenKey 4–109
- OpenProcess 4–108, 4–114
- OutputDebugString 5–12
- Process32First 5–85, 4–108
- Process32Next 5–85, 4–108
- ProtectVirtualMemory 5–26
- QueryInformationProcess 5–12, 5–38
- QuerySystemInformation 4–109
- QuerySystemTime 5–13
- ReadFile 4–87
- ReadProcessMemory 4–127, 4–129
- RegOpenKey 2–18, 4–87, 5–89, 4–109
- ResumeThread 5–54
- RtlDecompressBuffer 5–26, 5–29
- SetWindowsHookEx 5–67, 5–67-70, 5–70, 4–126
- ShellExecute 2–154
- SizeofResource 2–140
- UnmapViewOfSection 5–53, 54
- URLDownloadToFileA 3–72
- VirtualAlloc 5–52, 5–54, 4–92, 4–108
- VirtualProtect 4–127
- wcsicmp 2–106
- WinExec 3–72, 3–156
- WriteProcessMemory 5–52, 5–54, 4–108, 4–116, 4–127
- WriteVirtualMemory 5–26, 4–109

Structure

- EXCEPTION_REGISTRATION 2–179
- KBDLLHOOKSTRUCT 2–179
- PROCESS_INFORMATION 2–94
- STARTUPINFOA 2–94

Instruction

- ADD 2–37
- AND 2–37, 2–40
- CALL 2–20, 2–37, 2–39
- CMP 2–37, 2–40
- IMUL 2–37
- JA 2–41
- JB 2–41, 2–106
- JC 2–106
- JCC 2–41
- JE 2–41
- JG 2–41
- JL 2–41
- JMP 2–22, 2–37, 2–39
- JZ 2–41
- LEAVE 2–83
- LOOP 2–39
- LOOPcc 2–109
- MOV 2–22, 2–30, 2–37
- NOP 5–9
- OR 2–37
- PUSH 2–22
- RDTSC 5–13
- RET 2–39, 2–83
- SCASB 2–22
- SETZ 2–185
- SHL 2–37
- SHR 2–37
- SUB 2–37, 2–40
- TEST 2–37, 2–40
- XCHG 2–37

Register

- CS 2–25
- DS 2–25
- EAX 2–24
- EBP 2–24, 2–67
- EBX 2–24
- ECX 2–24
- EDI 2–24
- EDX 2–24
- EIP 2–25
- ES 2–25
- ESI 2–24

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template

  - ESP 2–24
  - FS 2–25
  - GS 2–25, 5–100, 5–127
  - SS 2–25

A

- AcroForm 3–73
- ActiveXObject 3–141
- Addressing mode 2–22, 2–32
- Administrative rights 1–39
- Adobe Reader 3–55
- advapi32.dll 4–88
- Analysis detection 5–93
- And 2–124
- Anti-analysis 5–163
- Anti-Debugger 5–6
  - Countermeasure 5–9, 5–14
- Any.run 1–13
- ApateDNS 1–124
- API 2–18
  - Call 4–125
  - Concealed 5–61
  - Hooking 4–106
  - Inline hook 4–126
  - Native 4–109
  - Pattern 2–135
- API Monitor 1–94
- app.setTimeOut 3–66
- app.viewerVersion 3–66
- Archive (self-extracting) 3–129
- Arguments 2–80-81
  - 64-bits 2–156
- arguments.callee 3–33
- Armadillo 4–8
- ASLR 4–19, 5–56, 5–74
- AutoConfigURL 3–144
- AutoDetect 3–144
- Automated analysis 1–13, 1–18
- AutoRun 4–64
- AV detection 5–78
- Avast Antivirus 5–82
- AVCaesar 1–13
- avghookx.dll 5–78

B

- Balbuzard toolkit 5–18
- Base Offset 4–82
- Base64 3–59, 4–71
- base64dump.py 3–60, 3–70, 4–72
- BAT File 4–64, 4–144
- bbcrack.py 5–18
- Beaconing 1–17
- Behavioral Analysis 1–33, 1–51
- Behavioral analysis 2–4
- BeingDebugged (flag) 5–12
- BelkaSoft Live RAM Capturer 4–136
- BHOs 2–148
- Binary expression 2–123
- Binary Ninja 2–6, 1–80
- BinText 5–97
- 64-bits 2–150
- box-js 3–48, 3–143
  - Download 3–51
- Branch 2–38
- Breakpoint 1–85
  - Countermeasure 5–163
  - Hardware 5–107, 5–158
  - hardware 5–122
  - VirtualProtect 5–156
- brutexor.py 5–17
- brxor.py 5–17
- Burp 3–4
- Byte 2–27
- Bytehist 4–11

C

- C2 1–107
- Call Stack 5–30, 4–89
- Calling convention 2–71
- CAPE 1–13
- Capstone 2–6
- cdecl 2–71, 2–84
- Censys 1–13
- certutil 3–142
- CFF Explorer 4–13, 5–74, 2–145
  - ASLR 4–20-21
- Clonezilla 1–32
- Code analysis 2–4, 1–33, 1–80
  - Dynamic 1–81
  - Static 1–81
- Code Injection 5–61, 4–106
- Code lifecycle 2–5
- Collab.collectEmailInfo 3–67
- Command and control (C2) 1–17
- Compiler 2–5
- Compound File Binary Format (CFBF) 3–89
- Constant 2–35
- CREATE_SUSPENDED 5–51, 5–175
- CScript 3–37, 3–47, 3–160-163
- CurrentVersion/Run 1–41
- Cutter 2–6

D

- .data 2–12
- Data Structure 2–33, 2–92
- dd 1–32
- Debugger 3–34, 1–81
  - Detection 5–80
- Debugging 4–46
- Decompiler vs Dissassembly 2–159

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template

- Deep Freeze 1–32
- Delete 4–144
- Deobfuscation 3–17, 3–52
- DER format 3–145
- Dereferencing 2–28
- Detect It Easy (DIE) 4–12, 1–47
- Device Driver 2–148
- diec 4–12
- Disassembling 1–81
- DLL 1–14, 2–137
- DNS 2–44, 1–66
- DNS leakage 1–20
- Do (loop) 2–112
- document
  - .createElement 3–8
  - .getElementById 3–46
- DoomJuice 1–119
- Driver 4–124
- Dropper 2–140
- Dumping 4–23
  - Debugger 4–30
- DumpIt 4–136
- Dword 2–27
- DynamicBase 5–74, 5–98
- Dyre 5–66

E

- EA → Effective Address
- Effective Address (EA) 2–31
- EFLAGS 2–25, 2–130
- Enable Content 3–148
- Entropy 4–9
  - PeStudio 4–10
- Entry Point 4–7, 5–121
- Entry Point (OEP) 4–23
- Entry Point field 5–112
- Epilogue 2–83, 2–90
- Ether 4–18
- Exception
  - Frame-based 5–99
  - Stack-based 5–99
- Executable File 2–5
- Exeinfo PE 4–12, 1–47, 5–97
- Exfiltration 1–17
- ExifTool 1–48
- exiftool 3–139
- Exports 2–15, 2–138

F

- fakedns 1–67
- FakeNet-NG 1–124
- fastcall 2–72
- feh 3–139
- Fiddler 3–6, 1–112
- file 4–13
- File extension 4–65
- File handle 3–125
- FileInsight 3–136, 3–158
- Flash 3–10
- FLOSS 5–24
- Fog 1–32
- For (loop) 2–112, 113
- FoxIT Reader 3–55
- Frame pointer 2–67
- FS:[0] 5–99
- FS:[30h] 5–12, 3–124
- FSG 4–8
- FTP 1–109
- Function 2–64
  - Epilogue 2–66
  - Prologue 2–66
- Function Call Graph 4–120
- Function Call Trees 4–117
- Function Graph 2–14

G

- getAnnots 3–66
- GetEIP 3–123
- Ghidra 2–6, 1–80
  - Bad instruction 5–120
  - Comment 2–43
  - Convert 5–21
  - Create Project 2–8
  - Data Convert 2–49
  - Data Type 2–47
  - Defined Strings 2–103
  - Equate 2–35
  - Function 2–57
  - Function Call Tree 2–55
  - Key binding 5–22
  - Local Variable Conflict 2–93
  - Project Analysis 2–10
  - Propagate External Parameters 2–47
  - Show References 2–79
  - String Reference 2–106
  - Symbol reference 2–16
  - Symbol References 4–112
  - TLS 5–118
- Global Descriptor Table (GDT) 5–127

H

- hachoir-urwid 3–139
- Handle 1–87-88
- Hardware breakpoint 4–53
- HashSets 1–13
- Headers 2–12
- Heap Spraying 3–64, 3–66
- Hex encoding 4–74
- Hexadecimal string 3–67
- Honeyclient 3–4

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template

- Honeypot 1–21
- Hooking 4–124, 4–126
  - Call Table 4–124
  - JMP 4–130
  - PUSH/RET 4–131, 4–152
- Hopper 2–6, 1–80
- Hot patching 2–66
- HTML 3–21
- HTTP 2–44
- httpd 1–71, 1–105
- Hybrid Analysis 1–13, 1–15

I

- I/O Request Packet (IRP) 4–124
- IAT → Import Address Table
- IDA 2–6, 1–80
- IDT → Interrupt Descriptor Table
- If-Else 2–99-102
- ImageBase 4–19
- Immediate value 2–22
- Import Address Table (IAT) 4–7, 2–16, 4–23, 5–162
  - Fix (Scylla) 4–25, 4–41-42
  - Rebuild 5–113
- Import/Export Tables 4–124
- Imports 2–15, 1–45, 5–146
  - Fixer 4–23
- Indicator of Compromise (IOC) 1–17, 1–43
- INetSim 3–78, 1–109
- Injection 4–122-123
- Inlining 2–76
- Instruction 2–22, 2–37
- Interactive Behavior Analysis 1–101
- Intermodular call 4–51
- Internet access 1–19
- Internet Explorer 3–25, 3–39
- Interpreter 3–36
- Interrupt Descriptor Table (IDT) 4–124
- Intezer Analyze 1–13
- IP redirection 1–115
- iptables 1–117
- IRC 2–44, 1–109
- IRP → I/O Request Packet

J

- JavaScript 3–21
  - Anti-analysis 3–33, 3–42
  - Beautify 3–20
  - BreakPoint 3–28
  - Console 3–30
  - Debug 3–25
  - Search and Replace 3–45
  - Ternary Operator 3–44
  - Tuple 3–43
  - Watch 3–30
- javascript 4–67
- jmp2it 4–83, 3–128
  - Pause 4–83
  - x32dbg 4–83-85
- JonDonym 1–20
- js 3–38, 4–70
- JSTool 3–20
- Jump table 2–131, 2–184

K

- Kahu Security 3–48
- Kahu tools 5–18
- KdDebuggerFlag 5–80
- Key logger 2–172

L

- Layering 4–124
- libemu 3–72
- Linker 2–5
- Loader 2–5
- location.href 3–39
- Loop 2–108, 2–110

M

- Macro 3–83, 3–114
  - AutoOpen 3–88
  - Debug 3–101, 3–111
  - Dump 3–92
  - Enable Content 3–85, 3–107
  - Extraction 3–87
  - Locals 3–103
- Malware 1–5
- Malware analysis 1–5
  - input/output 1–10
  - stage 1–9
- Malware Hash Registry 1–13
- Malware lab 1–24-25
  - Configuration 1–28
  - Isolation 1–28
- Malware report 1–11
- malware-jail 3–48
- MASTIFF 1–48
- Memory
  - Page 4–143
  - Virtual Offset 4–154
  - Virtual vs Physical 4–143
- Memory Forensics 4–135
- Memory map 4–49
- message_extract.py 3–150
- MetaDefender 1–13
- Minidriver 4–124
- Misdirection 5–96, 5–142
- Most significant bit (MSB) 2–182
- mshta.exe 4–67

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template

- Mutant → Mutex
- Mutex 1–15, 2–143
- MyDoom 1–119
- MZ 2–142, 4–147

N

- NanoLocker 5–141
- nc 1–120
- netcat 1–120
- Nightmare 3–48
- NoMoreXOR.py 5–18
- nslookup 1–67
- NTVDM 2–149

O

- Obfuscation 3–18
- Object Code 2–5
- OLE1 3–117
- OLE2 3–89
- olebrowse.py 3–92, 3–136
- olecfinfo 3–92, 3–136
- oledir.py 3–92, 3–136
- oledump.py 3–92, 3–94, 3–134
- oleid.py 3–136
- olemap.py 3–136
- olevba.py 3–87, 3–108
- OllyDbg 5–79
- OllyDumpEx 4–37, 5–109, 5–112
- OOXML 3–89
- Open Threat Exchange 1–13, 1–16
- openssl 3–145
- Operand 2–22
- Operation 2–22
- Or 2–125
- Original Entry Point (OEP) 4–23, 4–30, 4–38, 5–106
- OSINT 1–17

P

- p-code 3–104
  - Dump 3–106
- P2P 2–44
- packerid 4–13
- Packing 4–3, 4–5-6, 1–47
  - Debug 4–55
- Page Memory Rights 4–39, 4–119, 5–160, 5–171
- PassiveTotal 1–13, 1–16
- Patching 5–9, 4–86, 5–130
- pcodedmp.py 3–106, 3–109
- PDF 3–55
  - Analysis 3–80
  - Dictionary 3–58
  - File structure 3–56
  - Object 3–57
  - Stream 3–80
- pdf-parser.py 3–61-63
- pdfid.py 3–61
- pdftk 3–80
- PE (Portable Executable) 1–47
- PE Capture 1–124
- PE Tools 4–23
- pe_unmapper 5–161, 5–170
- PECompact 5–97
- peeppdf.py 3–65
- peframe 1–46, 4–149
- pepack 4–13
- Persistence 1–41
- pescan 4–13, 1–48
- pestr 1–42, 4–149, 5–170
- PeStudio 1–44, 2–138
  - ASLR 4–20
  - Packed 4–10
  - TLS 5–117
- Pev Toolkit 1–42
- PhantomJS 3–48
- Pinpoint 3–4
- Pointer 2–28
- PolarProxy 1–124
- POP3 1–109
- PortEx 1–48
- Position-independent code (PIC) 2–150
- PowerShell 4–71
  - Debug 4–77
  - Dump 4–78
- ProcDOT 5–46, 1–52, 1–63
- Process Environment Block (PEB) 5–12, 3–124
- Process Hacker 5–44, 1–52
  - Dump 5–171
  - String 4–22
- Process hollowing 5–54, 5–61
  - Dump 5–55-56
- Process Monitor 3–12, 5–44, 1–52, 1–54, 1–61
  - Process Tree 5–45
- Process replacement 5–54
- Program Tree 2–12
- Prologue 2–82, 2–89
- ProtectionID 4–13
- PXE booting 1–32

Q

- qpdf 3–80
- Queuing 4–124
- Quttera 1–13
- Qword 2–27

R

- Radare 2–6, 1–80
- .rdata 2–12
- RDG Packer Detector 4–13
- Redline 4–137

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template

- reg_export 4–68
- Register 2–22, 2–24, 2–26
  - 64-bits 2–151
  - Edit 5–9
- Register-based CPU 2–23
- Registry 1–41, 4–64, 4–68
- Regshot 1–52, 1–55, 1–59
- Rekall 4–137
- .reloc 2–12
- Remapping 5–161
- Ressource Dumpimp 2–142
- Return Pointer 2–65
- RIP-relative addressing 2–150
- ROL 1–96
- RollBack Rx 1–32
- RootKit 4–107, 2–148
- ROR 1–96
- RTF 3–117
  - Auto extraction 3–117
  - Controls/Group/Object 3–118
  - Nesting 3–120
- rtfdump.py 3–119-212, 3–151-153
  - Carving 3–155
- rtfobj.py 3–157
- RunPE 5–54

S

- Sandbox detection 5–63, 5–66, 5–93
- Sandboxie 5–82
- scdbg 3–72, 4–75, 4–79, 3–125, 3–156
  - Debug 4–81
  - Stack manipulation 4–82
- Scout 3–4
- sctest 3–72
- Scylla 4–24, 5–113
  - IAT 4–25, 5–162
- ScyllaHide 5–14, 5–27
- Sections 4–7, 2–12, 1–45
- Segment 2–25
- selector 5–127
- Self-defense 1–29
- set-static-ip 1–115
- SetBPX 1–85
- setdllcharacteristics 4–20-21
- Shellcode 3–69
- shellcode2exe.py 3–77, 3–128
- Signed 2–182
- signsrch 1–48
- Single-step 4–34, 1–83
- SMTP 1–109
- Snapshot 1–31
- SpiderMonkey 3–37, 3–66, 4–70
- SRE → Software Reversing Engineering
- SRP Stream 3–93
- SSDT → System Service Descriptor Table
- SSView 3–92, 3–136
- Stack 2–67-69
  - Breakpoint 5–115
  - Cleanup 5–106
  - Pointer 2–67
- Stack String → String,stack
- Static properties 1–41
  - Analysis 1–33
- stdcall 2–71
- strace 2–18
- strdeob.pl 5–23
- String 2–33, 2–103
  - Process Hacker 4–22
  - Stack 5–19
- Strings 1–42, 4–70, 3–91
  - pestr 5–170
- strlen 2–111
- Structured Exception Handling (SEH) 5–99
- Structured Storage (SS) 3–89
- swf_mastah.py 3–80
- Switch 2–130
- SYN 1–108
- syncAnnotScan 3–66
- SysAnalyzer 5–82
- System call 2–18
- System Service Descriptor Table (SSDT) 4–124

T

- TcpLogView 1–124
- TDL RootKit 2–148
- .text 2–12
- TFTP 1–109
- Themida 4–8
- thiscall 2–72
- Thread Information Block (TIB) 5–99, 3–124
- Thread Local Storage (TLS) 5–116
- ThreatAnalyzer 5–82
- Thug 3–4
- Tinba 5–66
- TitanMist 4–18
- TOR 1–20
- translate.py 1–97
- TrickBot 5–161
- TrId 1–48
- trid 4–13
- try/except 5–99
- Two complements 2–182

U

- Unicode conversion 4–70
- unicode2raw 3–70
- Universal Import Fixer 4–23
- Unpacking 5–145
- UnpacMe 4–18
- Unsigned 2–182
- unXOR 5–18
- Upatre 5–66

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template

- UPX 4–8
  - Unpacking (auto) 4–17
- urlscana.io 1–13

V

- V8 3–37
- Variable
  - Global 2–53
  - Local 2–53, 5–151
  - Static 2–53
- VBA → Visual Basic for Applications
- VBScript 3–18, 3–47
  - Deobfuscation 3–160
- VCF 3–49
- vftable → Virtual Function Table
- Viper 1–48
- VirSCAN 1–13
- Virtual Address Descriptor (VAD) 4–146
- Virtual Function Table (vftable) 4–124
- Virtual size 4–10
- Virtualization 1–26
- Virtualization detection 5–65
- VirusTotal 1–13, 14
- Visual Basic for Applications (VBA) 3–84
  - Downloader 3–84
  - Stomping 3–104
  - Watches 3–112
- Volatility 4–137, 5–170
  - apihooks 4–150
  - cmdline 4–142
  - dlllist 4–145
  - kdbgscan 4–138
  - ldrmodules 4–145
  - malfind 4–146
  - memdump 4–144
  - pslist 4–139
  - pstree 4–140
  - Virtual Offset 4–150
- VPN 1–20
- vURL 1–13

W

- Website 3–4
- WH_MOUSE_LL 5–70
- While (loop) 2–112
- WinDbg 1–80, 5–82
- Windows Virtual PC 5–82
- WinPMEM 4–136
- Wireshark 1–52, 1–56, 1–66
  - Filter 3–13
  - TCP Stream 1–73
- WM_(SYS)KEY(XX) 2–179
- WM_MOUSEMOVE 5–71
- WMIC 3–12
- Word 2–27
- wow64 2–149
- WPE Pro 5–82
- WSF 3–49

X

- x32dbg
  - cleardb 4–48
  - Dump 5–29, 5–34, 5–57, 4–95, 4–101, 5–160
  - Follow in Memory Map 5–58
  - Ignore Exception 5–105
  - jmp2it 4–83-85
  - Save Patch 5–11
  - SEH 5–101, 5–104
  - SetBPX 5–28
  - Stack edition 5–77
  - Strings 4–35
  - Thread 5–177
  - TLS 5–121
- xAnalyzer 5–151
- XFA → XML Forms Architecture
- XML Forms Architecture (XFA) 3–73-75
- XOR 1–96
- XOR (VBA) 3–97
- XOR Encoding 5–15, 3–98
- XOR Obfuscation 5–119
- xor-kpa.py 3–100-101
- xorBruteForcer.py 5–18
- XORSearch 5–16, 3–122
- xortool 5–18
- xxd 1–75, 1–97, 1–118

Y

- Yara 3–137
- Yotalhash 1–13

Z

- zipdump.py 3–137-138

This work is licensed under a Creative Commons Attribution 4.0 International License, by A. Cailliau
Template by A. D'hondt, https://github.com/dhondta/tex-course-index-template