



Reverse Engineering (REV3)

UE 02 – Statische Analyse – Protokoll

Jakob Mayr

WS 2023/2024

Einleitung

...

Aufgabe 1 - Statische Analyse Windows

Erstellen der Dateien

In Aufgabe 1 ist eine Anwendung, welche "infected" auf `stdout` ausgibt in c zu schreiben und mit Visual Studio auf 4 verschiedene Varianten zu bauen.

Die vier Varianten mit deren Eigenschaften und Unterschieden (Quelle: ChatGPT):

1. Static Release (/MT):

- (a) Laufzeitbibliothek: Statisch
- (b) Debug-Informationen: Nein
- (c) Eigenschaften:
 - i. Die Laufzeitbibliothek wird in die ausführbare Datei eingebettet.
 - ii. Größere Dateigröße, da der Code der CRT (C Runtime Library) direkt in die Anwendung eingefügt wird.
 - iii. Keine Abhängigkeit von DLLs (Dynamically Linked Libraries) zur Laufzeit.
 - iv. Optimierte für Geschwindigkeit und nicht für das Debugging.

2. Static Debug (/MTd):

- (a) Laufzeitbibliothek: Statisch
- (b) Debug-Informationen: Ja
- (c) Eigenschaften:
 - i. Ähnlich wie /MT, aber mit zusätzlichen Debug-Informationen und weniger Optimierungen.
 - ii. Erleichtert das Debugging, da Variablen leichter überwacht werden können.
 - iii. Größerer Speicherbedarf und langsamere Ausführung im Vergleich zur Release-Version.

3. Dynamic Release (/MD):

- (a) Laufzeitbibliothek: Dynamisch
- (b) Debug-Informationen: Nein
- (c) Eigenschaften:
 - i. Verlinkt dynamisch mit den CRT-DLLs (Z.B. `msvcrXXX.dll`).
 - ii. Kleinere Dateigröße, da die Laufzeitbibliothek nicht eingebettet ist.
 - iii. Erfordert, dass die CRT-DLLs zur Laufzeit verfügbar sind.
 - iv. Optimierte für Geschwindigkeit.

4. Dynamic Debug (/MDd):

- (a) Laufzeitbibliothek: Dynamisch
- (b) Debug-Informationen: Ja
- (c) Eigenschaften:
 - i. Ähnlich wie /MD, aber mit zusätzlichen Debug-Informationen und weniger Optimierungen.
 - ii. Erleichtert das Debugging.
 - iii. Erfordert, dass die Debug-Version der CRT-DLLs zur Laufzeit verfügbar ist.

Unterschiede:

1. **Statische vs. Dynamische Verlinkung:** /MT und /MTd binden die Laufzeitbibliothek statisch ein, wodurch die ausführbare Datei unabhängig von externen DLLs ist. /MD und /MDd verlinken dynamisch und erfordern, dass die entsprechenden DLLs zur Laufzeit vorhanden sind.
2. **Debug vs. Release:** Die Debug-Optionen (/MTd und /MDd) enthalten zusätzliche Debug-Informationen und sind nicht so stark optimiert wie die Release-Optionen (/MT und /MD), was das Debugging erleichtert, aber die Leistung beeinträchtigen kann.

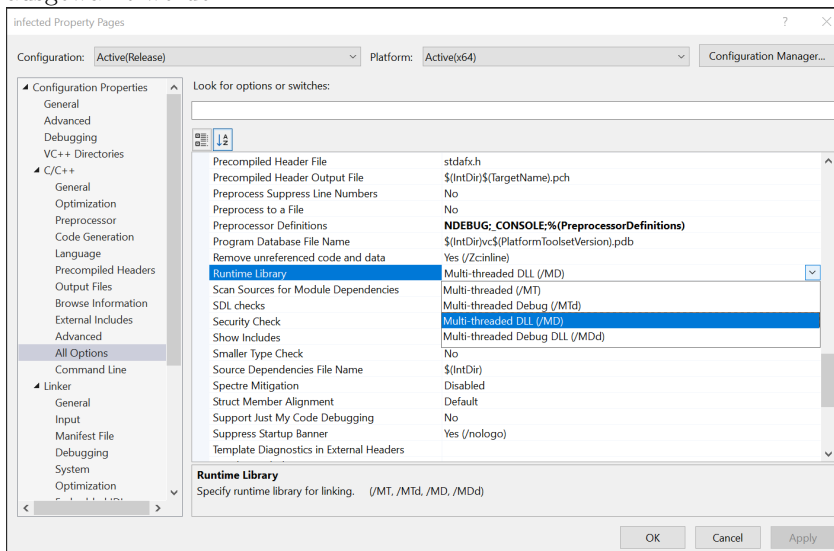
Der Programm-Code für die gewünschte Executable:

```

1  #include <stdio.h>
2
3  int main() {
4      printf("infected");
5      return 0;
6  }
7

```

In den Einstellungen des Visual Studio Projekts kann die gewünschte Variante für die "Runtime Library" ausgewählt werden:



Die Files nach erstellen, hier ist direkt ersichtlich, dass beide statischen Varianten größer sind (Length), da sich der Code der Libraries im File befindet:

```

PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> dir

Directory: C:\Users\admin_sin\Desktop\rev3-s2210239021\releases

Mode                LastWriteTime         Length Name
----                -
-a----          24.10.2023   16:15           10752 infected-md.exe
-a----          24.10.2023   16:16           14336 infected-MDd.exe
-a----          24.10.2023   16:10          139264 infected-mt.exe
-a----          24.10.2023   16:14          376832 infected-mtd.exe

PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases>

```

Analyse der Dateien

Um die erzeugten Executables zu analysieren sollen zumindest die Tools **dumpbin** und **strings** verwendet werden.

dumpbin

dumpbin ist ein Command-Line Tool, zur Verfügung gestellt von Microsoft Visual Studio um PE-Files (Portable Executable) zu analysieren. Es können beispielsweise die flags **/DEPENDENTS**, **/EXPORTS**, **/HEADERS** oder **/ALL** verwendet werden, um alle vom Tool lieferbaren Informationen zu erhalten.

Folgende Aufrufe zeigen die Ausgabe ohne Parameter:

```
PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> dumpbin.exe .\infected-mt.exe
Microsoft (R) COFF/PE Dumper Version 14.36.32535.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-mt.exe

File Type: EXECUTABLE IMAGE

Summary

2000 .data
2000 .pdata
8000 .rdata
1000 .reloc
1000 .rsrc
15000 .text
1000 .RDATA

PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases>
```

(a) dumpbin mt-file

```
PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> dumpbin.exe .\infected-mtd.exe
Microsoft (R) COFF/PE Dumper Version 14.36.32535.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-mtd.exe

File Type: EXECUTABLE IMAGE

Summary

3000 .data
4000 .pdata
15000 .rdata
1000 .reloc
1000 .rsrc
43000 .text
1000 .RDATA

PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases>
```

(b) dumpbin mtd-file

```
PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> dumpbin.exe .\infected-md.exe
Microsoft (R) COFF/PE Dumper Version 14.36.32535.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-md.exe

File Type: EXECUTABLE IMAGE

Summary

1000 .data
1000 .pdata
1000 .rdata
1000 .reloc
1000 .rsrc
1000 .text

PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases>
```

(c) dumpbin md-file

```
PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> dumpbin.exe .\infected-MDd.exe
Microsoft (R) COFF/PE Dumper Version 14.36.32535.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-MDd.exe

File Type: EXECUTABLE IMAGE

Summary

1000 .data
1000 .pdata
1000 .rdata
1000 .reloc
1000 .rsrc
2000 .text

PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases>
```

(d) dumpbin mdd-file

Figure 1: All infected.exe files analysed with dumpbin

Die Debug-Versionen (infected-MDd.exe und infected-mtd.exe) haben tendenziell größere .text Sektionen als ihre entsprechenden Release-Versionen, da sie zusätzliche Debug-Informationen enthalten.

Die statisch verlinkten Versionen (infected-mt.exe und infected-mtd.exe) haben größere .text und .rdata Sektionen im Vergleich zu den dynamisch verlinkten Versionen, da die C Runtime Library direkt in die ausführbare Datei eingebettet ist.

Die dynamisch verlinkten Versionen (infected-md.exe und infected-MDd.exe) sind im Allgemeinen kleiner, weil sie zur Laufzeit externe DLLs verwenden.

Folgende zwei screenshots zeigen sowohl den Anfang als auch das Ende der 4 verschiedenen Files:

```

1  output-strings-mt... 2  output-strings-mt... 3  output-strings-mt... 4  output-strings-mt...
5  output-strings-mt.txt
6  I This program can be run in DOS mode.
7  C:\Users\hadmin_sin\Desktop\rev3-12201023\39021\infected
8  1_texttosmtp
9  2...data$voltmd
10 3...data$zzzdbg
11--C_specific_handler
12--current_exception
13--current_exception_context
14VCURNTIME100.dll
15--_actio_lob_func
1611_stdio_common_vfprintf
1712_stdio_common_vfprintf
1813_seh_filter_exe
1914_set_app_type
2015_set_app_type
2116_set_classname
2217_configure_narrow_argv
2318_initialize_narrow_environment
2419_set_initial_narrow_environment
2520_set_initial_narrow_environment
2621_set_mode
2722_set_mode
2823_argc
2924_argc
3025_register_thread_local_exe_atexit_callback
3126_configThreadLocate
3227_set_new_mode
3328_p_commande
3429_initialize_oneixit_table
3536_register_oneixit_function
3637_set_atexit
37api-ms-win-crt-stdio-l1-1-0.dll
38api-ms-win-crt-runtime-l1-1-0.dll
39api-ms-win-crt-math-l1-1-0.dll
40api-ms-win-crt-locale-l1-1-0.dll
41api-ms-win-crt-heap-l1-1-0.dll
42RTLCaptureContext
43RTLockupFunctionEntry
44RTVirtualUnwind
45api-ms-win-crt-exceptionfilter-l1-1-0.dll
46RTLockupContext
47RTLockupFunctionEntry
48RTVirtualUnwind
49UnhandleExceptionFilter
50SetUnhandleExceptionFilter
51GetCrtProcess
52TerminateProcess
53IsProcessorFeaturePresent
54QueryPerformanceCounter
55GetCrtProcessId
56GetCrtThreadId
57GetModuleHandleEx
58InitializeSystemAsFileTime
59KERNEL32.dll
60<xml version='1.0' encoding='UTF-8' standalone='yes'
61--assembly xmlns='urn:schemas-microsoft-com:asm.v1'
62--trustInfo xmlns='urn:schemas-microsoft-com:asm.v3'
63--security
64--requestedPrivileges
65--requestedExecutionLevel level='asInvoker' u
66--assembly
67--security
68--trustInfo
69--assembly
70main
71
72  output-strings-mt.txt
73 I This program can be run in DOS mode.
74 C:\Users\hadmin_sin\Desktop\rev3-12201023\39021\infected
75 1_texttosmtp
76 2...data$voltmd
77 3...data$zzzdbg
78--C_specific_handler
79--current_exception
80--current_exception_context
81VCURNTIME100.dll
82--_actio_lob_func
8311_stdio_common_vfprintf
8412_stdio_common_vfprintf
8513_seh_filter_exe
8614_set_app_type
8715_set_app_type
8816_set_classname
8917_configure_narrow_argv
9018_initialize_narrow_environment
9119_set_initial_narrow_environment
9220_set_initial_narrow_environment
9321_set_mode
9422_set_mode
9523_argc
9624_argc
9725_register_thread_local_exe_atexit_callback
9826_configThreadLocate
9927_set_new_mode
10028_p_commande
10129_initialize_oneixit_table
10236_register_oneixit_function
10337_set_atexit
104api-ms-win-crt-stdio-l1-1-0.dll
105api-ms-win-crt-runtime-l1-1-0.dll
106api-ms-win-crt-math-l1-1-0.dll
107api-ms-win-crt-locale-l1-1-0.dll
108api-ms-win-crt-heap-l1-1-0.dll
109RTLCaptureContext
110RTLockupFunctionEntry
111RTVirtualUnwind
112api-ms-win-crt-exceptionfilter-l1-1-0.dll
113RTLockupContext
114RTLockupFunctionEntry
115RTVirtualUnwind
116UnhandleExceptionFilter
117SetUnhandleExceptionFilter
118GetCrtProcess
119TerminateProcess
120IsProcessorFeaturePresent
121QueryPerformanceCounter
122GetCrtProcessId
123GetCrtThreadId
124GetModuleHandleEx
125InitializeSystemAsFileTime
126KERNEL32.dll
127<xml version='1.0' encoding='UTF-8' standalone='yes'
128--assembly xmlns='urn:schemas-microsoft-com:asm.v1'
129--trustInfo xmlns='urn:schemas-microsoft-com:asm.v3'
130--security
131--requestedPrivileges
132--requestedExecutionLevel level='asInvoker' u
133--assembly
134--security
135--trustInfo
136--assembly
137main
138
139  output-strings-mt.txt
140 LoadLibraryExW
141 CloseHandle
142 RaiseException
143 GetProcAddress
144 LoadLibraryExW
145 EncodePointer
146 RaiseException
147 RTCPtoFileHeader
148 GetModuleHandleExW
149 GetModuleFileNameW
150 ExitProcess
151 GetModuleHandleExW
152 GetModuleFileNameW
153 ExitProcess
154 GetModuleHandleExW
155 GetModuleFileNameW
156 FLSEValue
157 FLSEValue
158 FLSEValue
159 LCMapiString
160 CompareStringW
161 GetFileType
162 OutputDebugStringW
163 FindFirstFileExW
164 FindNextFileW
165 GetLastError
166 InvalidDataPage
167 MultiByteToWideChar
168 WideCharToMultiByte
169 GetEnvironmentStringsW
170 FreeEnvironmentStringsW
171 FreeEnvironmentStringsW
172 SetEnvironmentVariableW
173 SetStdHandle
174 GetStdHandle
175 GetProcessHeap
176 HeapReAlloc
177 HeapQueryInformation
178 GetConsoleOutputCP
179 FlushFileBuffers
180 FlushFileBuffers
181 FlushFileBuffers
182 GetConsoleOutputCP
183 GetFileSize
184 SetFilePointerEx
185 HeapReAlloc
186 CloseHandle
187 CreateFileW
188 WriteConsoleW
189 KERNEL32.dll
190
191 abcdcfghijklnopqrstuvwxy
192 ABCDEFHIJKLMNOPQRSTUVWXYZ
193
194 abcdcfghijklnopqrstuvwxy
195 ABCDEFHIJKLMNOPQRSTUVWXYZ
196
197 abcdcfghijklnopqrstuvwxy
198 ABCDEFHIJKLMNOPQRSTUVWXYZ
199
200 --AVAXceptionInfo@0
201 --AVAXceptionInfo@0
202 --AVType_info@0
203 --AVType_info@0
204
205 <xml version='1.0' encoding='UTF-8' standalone='yes'
206 --assembly xmlns='urn:schemas-microsoft-com:asm.v1'
207 --trustInfo xmlns='urn:schemas-microsoft-com:asm.v3'
208 --security
209 --requestedPrivileges
210 --requestedExecutionLevel level='asInvoker' u
211 --assembly
212 --security
213 --trustInfo
214 --assembly
215 main
216
217  output-strings-mt.txt
218 LoadLibraryExW
219 CloseHandle
220 RaiseException
221 RTCPtoFileHeader
222 GetModuleHandleExW
223 GetModuleFileNameW
224 ExitProcess
225 GetModuleHandleExW
226 GetModuleFileNameW
227 ExitProcess
228 GetModuleHandleExW
229 GetModuleFileNameW
230 FLSEValue
231 FLSEValue
232 FLSEValue
233 LCMapiString
234 CompareStringW
235 GetFileType
236 OutputDebugStringW
237 FindFirstFileExW
238 FindNextFileW
239 GetLastError
240 InvalidDataPage
241 MultiByteToWideChar
242 WideCharToMultiByte
243 GetEnvironmentStringsW
244 FreeEnvironmentStringsW
245 FreeEnvironmentStringsW
246 SetEnvironmentVariableW
247 SetStdHandle
248 GetStdHandle
249 GetProcessHeap
250 HeapReAlloc
251 HeapQueryInformation
252 GetConsoleOutputCP
253 FlushFileBuffers
254 FlushFileBuffers
255 FlushFileBuffers
256 GetConsoleOutputCP
257 GetFileSize
258 SetFilePointerEx
259 HeapReAlloc
260 CloseHandle
261 CreateFileW
262 WriteConsoleW
263 KERNEL32.dll
264
265 abcdcfghijklnopqrstuvwxy
266 ABCDEFHIJKLMNOPQRSTUVWXYZ
267
268 abcdcfghijklnopqrstuvwxy
269 ABCDEFHIJKLMNOPQRSTUVWXYZ
270
271 --AVAXceptionInfo@0
272 --AVAXceptionInfo@0
273 --AVType_info@0
274 --AVType_info@0
275
276 <xml version='1.0' encoding='UTF-8' standalone='yes'
277 --assembly xmlns='urn:schemas-microsoft-com:asm.v1'
278 --trustInfo xmlns='urn:schemas-microsoft-com:asm.v3'
279 --security
280 --requestedPrivileges
281 --requestedExecutionLevel level='asInvoker' u
282 --assembly
283 --security
284 --trustInfo
285 --assembly
286 main
287

```

Fragen

1. Welche Imports werden verwendet?

Durch den Aufruf von `dumpbin /IMPORTS <PE-filename>` können die Imports ermittelt werden:

```
PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases> dumpbin /IMPORTS .\infected-mt.exe
Microsoft (R) COFF/PE Dumper Version 14.37.32825.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-mt.exe

File Type: EXECUTABLE IMAGE

Section contains the following imports:

  KERNEL32.dll
    140016000 Import Address Table
    14001FE98 Import Name Table
    0 time date stamp
    0 Index of first forwarder reference

    4F5 RtlCaptureContext
    4FD RtlLookupFunctionEntry
    504 RtlVirtualUnwind
    5E6 UnhandledExceptionFilter
    5A4 SetUnhandledExceptionFilter
    232 GetCurrentProcess
    5C4 TerminateProcess
    3A8 IsProcessorFeaturePresent
    470 QueryPerformanceCounter
    233 GetCurrentProcessId
    237 GetCurrentThreadId
    30A GetSystemTimeAsFileTime
    38A InitializeSlistHead
```

- (a) infected-mt.exe
 - i. KERNEL32.dll
- (b) infected-mtd.exe
 - i. KERNEL32.dll
- (c) infected-md.exe
 - i. VCRUNTIME140.dll
 - ii. api-ms-win-crt-stdio-l1-1-0.dll
 - iii. api-ms-win-crt-runtime-l1-1-0.dll
 - iv. api-ms-win-crt-math-l1-1-0.dll
 - v. api-ms-win-crt-locale-l1-1-0.dll
 - vi. api-ms-win-crt-heap-l1-1-0.dll
 - vii. KERNEL32.dll
- (d) infected-MDd.exe
 - i. VCRUNTIME140D.dll
 - ii. ucrtbased.dll
 - iii. KERNEL32.dll

2. Welche Sektionen werden verwendet?

Durch den Aufruf von `dumpbin <PE-filename>` können die Sektionen ermittelt werden:

```
PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> dumpbin.exe .\infected-md.exe
Microsoft (R) COFF/PE Dumper Version 14.36.32535.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-md.exe

File Type: EXECUTABLE IMAGE

Summary

    1000 .data
    1000 .pdata
    1000 .rdata
    1000 .reloc
    1000 .rsrc
    1000 .text
PS C:\Users\admin_sin\Desktop\rev3-s2210239021\releases> |
```

(a) infected-mt.exe

- i. .text
- ii. .rdata
- iii. .data
- iv. .pdata
- v. _RDATA
- vi. .rsrc
- vii. .reloc

(b) infected-mtd.exe

- i. .text
- ii. .rdata
- iii. .data
- iv. .pdata
- v. _RDATA
- vi. .rsrc
- vii. .reloc

(c) infected-md.exe

- i. .text
- ii. .rdata
- iii. .data
- iv. .pdata
- v. .rsrc
- vi. .reloc

(d) infected-MDd.exe

- i. .text
- ii. .rdata
- iii. .data
- iv. .pdata
- v. .rsrc
- vi. .reloc

3. **Was kannst du über die*den Author*in sagen?** Ein direkter Author ist nicht erkennbar. Allerdings könnte man auf "admin.sin" schließen, da das File für die "Program Database" (infected.pdb) im Pfad den User nennt:

```

1      Debug Directories
2
3      Time Type          Size          RVA      Pointer
4      -----
5      6537D1B0 cv          66 00003464      2264      Format: RSDS, {6F72B84A-D687
-4743-A104-E88EF40E7E96}, 4, C:\Users\admin.sin\Desktop\rev3-s2210239021\
infected\x64\Release\infected.pdb
6      6537D1B0 feat        14 000034CC      22CC      Counts: Pre-VC++ 11.00=0, C/C
++=30, /GS=30, /sdl=1, guardN=29
7      6537D1B0 coffgrp      284 000034E0      22E0      4C544347 (LTCG)
8      6537D1B0 iltcg         0 00000000         0
9
10

```

Dies ist in allen Files gleich herauszulesen.

4. **Was kannst du über die Umgebung, in der das Sample erzeugt wurde, sagen?**
 Auskünfte über die Umgebung können durch die Compiler-Version, eingebettete Ressourcen oder Abhängigkeiten und der Weiteren gegeben werden.

HEADERS

Durch den Aufruf von `dumpbin /HEADERS <PE-filename>` können mehrere Informationen über die Umgebung herausgefunden werden:

```

PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases> dumpbin /headers .\infected-md.exe
Microsoft (R) COFF/PE Dumper Version 14.37.32825.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-md.exe

PE signature found

File Type: EXECUTABLE IMAGE

FILE HEADER VALUES
 8664 machine (x64)
 6 number of sections
 6537D186 time date stamp Tue Oct 24 16:15:34 2023
 0 file pointer to symbol table
 0 number of symbols
 F0 size of optional header
 22 characteristics
    Executable
    Application can handle large (>2GB) addresses

```

(Der Screenshot zeigt nur den Beginn des outputs...)

Folgende Informationen können über die Files gefunden werden:

- (a) infected-mt.exe
 - i. **Architektur:** 8664 machine (x64)
 - ii. **Linker-Version:** 14.36 (typisch für Visual Studio-Version)
 - iii. **Zeitstempel:** Tue Oct 24 16:10:20 2023
 - iv. **Subsystem:** Windows CUI (Konsolenanwendung)
 - v. **DLL-Charakteristika:**
 Kompatibel mit "High Entropy Virtual Addresses", "Dynamic Base", "NX (No eXecute)", "Terminal Server Aware".
 Diese Eigenschaften sind Sicherheitsmerkmale, die oft in modernen Anwendungen verwendet werden.

- vi. Speicherreservierung: Die Größen für Stack- und Heap-Reservierung und -Commit sind angegeben, was Hinweise auf den für die Ausführung der Anwendung benötigten Speicher gibt.
- vii. **Verzeichnisse und Sektionen:** Verschiedene Verzeichnisse wie das Importverzeichnis, das Resourceverzeichnis, das Exceptionverzeichnis usw. sind aufgelistet. Diese geben Informationen über die Struktur der ausführbaren Datei und welche externen Funktionen oder Ressourcen sie verwendet.
Beispielsweise nur für mt-Version angeführt:

```

1          10 number of directories
2          0 [          0] RVA [size] of Export Directory
3          28DC [          A0] RVA [size] of Import Directory
4          5000 [          1E0] RVA [size] of Resource Directory
5          4000 [          174] RVA [size] of Exception Directory
6          0 [          0] RVA [size] of Certificates Directory
7          6000 [          30] RVA [size] of Base Relocation Directory
8          23A0 [          70] RVA [size] of Debug Directory
9          0 [          0] RVA [size] of Architecture Directory
10         0 [          0] RVA [size] of Global Pointer Directory
11         0 [          0] RVA [size] of Thread Storage Directory
12        2260 [          140] RVA [size] of Load Configuration Directory
13         0 [          0] RVA [size] of Bound Import Directory
14        2000 [          1A0] RVA [size] of Import Address Table Directory
15         0 [          0] RVA [size] of Delay Import Directory
16         0 [          0] RVA [size] of COM Descriptor Directory
17         0 [          0] RVA [size] of Reserved Directory
18

```

- viii. **Betriebssystem- und Subsystem-Version:** Das Betriebssystem und das Subsystem sind auf Version 6.00 eingestellt. Dies könnte auf eine Kompatibilität mit bestimmten Windows-Versionen hinweisen (z.B. Windows Vista, 7, 8, 10), die alle NT 6.x-Versionen sind.
 - ix. **Art der Ausführbaren:** Die Ausführbare ist als "Executable" markiert, was darauf hindeutet, dass es sich um eine Standard-Ausführbare (und nicht um eine DLL) handelt.
- (b) infected-mtd.exe
- i. **Architektur:** 8664 machine (x64)
 - ii. **Linker-Version:** 14.36 (typisch für Visual Studio-Version)
 - iii. **Zeitstempel:** Tue Oct 24 16:14:17 2023
 - iv. **Subsystem:** Windows CUI (Konsolenanwendung)
 - v. **DLL-Charakteristika:**
Kompatibel mit "High Entropy Virtual Addresses", "Dynamic Base", "NX (No eXecute)", "Terminal Server Aware".
Diese Eigenschaften sind Sicherheitsmerkmale, die oft in modernen Anwendungen verwendet werden.
 - vi. Speicherreservierung: Die Größen für Stack- und Heap-Reservierung und -Commit sind angegeben, was Hinweise auf den für die Ausführung der Anwendung benötigten Speicher gibt.
 - vii. **Verzeichnisse und Sektionen:** Verschiedene Verzeichnisse wie das Importverzeichnis, das Resourceverzeichnis, das Exceptionverzeichnis usw. sind aufgelistet. Diese geben Informationen über die Struktur der ausführbaren Datei und welche externen Funktionen oder Ressourcen sie verwendet.
 - viii. **Betriebssystem- und Subsystem-Version:** Das Betriebssystem und das Subsystem sind auf Version 6.00 eingestellt. Dies könnte auf eine Kompatibilität mit bestimmten Windows-Versionen hinweisen (z.B. Windows Vista, 7, 8, 10), die alle NT 6.x-Versionen sind.
 - ix. **Art der Ausführbaren:** Die Ausführbare ist als "Executable" markiert, was darauf hindeutet, dass es sich um eine Standard-Ausführbare (und nicht um eine DLL) handelt.

- (c) infected-md.exe
- i. **Architektur:** 8664 machine (x64)
 - ii. **Linker-Version:** 14.36 (typisch für Visual Studio-Version)
 - iii. **Zeitstempel:** Tue Oct 24 16:15:34 2023
 - iv. **Subsystem:** Windows CUI (Konsolenanwendung)
 - v. **DLL-Charakteristika:**
Kompatibel mit "High Entropy Virtual Addresses", "Dynamic Base", "NX (No eXecute)", "Terminal Server Aware".
Diese Eigenschaften sind Sicherheitsmerkmale, die oft in modernen Anwendungen verwendet werden.
 - vi. **Speicherreservierung:** Die Größen für Stack- und Heap-Reservierung und -Commit sind angegeben, was Hinweise auf den für die Ausführung der Anwendung benötigten Speicher gibt.
 - vii. **Verzeichnisse und Sektionen:** Verschiedene Verzeichnisse wie das Importverzeichnis, das Resourceverzeichnis, das Exceptionverzeichnis usw. sind aufgelistet. Diese geben Informationen über die Struktur der ausführbaren Datei und welche externen Funktionen oder Ressourcen sie verwendet.
 - viii. **Betriebssystem- und Subsystem-Version:** Das Betriebssystem und das Subsystem sind auf Version 6.00 eingestellt. Dies könnte auf eine Kompatibilität mit bestimmten Windows-Versionen hinweisen (z.B. Windows Vista, 7, 8, 10), die alle NT 6.x-Versionen sind.
 - ix. **Art der Ausführbaren:** Die Ausführbare ist als "Executable" markiert, was darauf hindeutet, dass es sich um eine Standard-Ausführbare (und nicht um eine DLL) handelt.
- (d) infected-MDd.exe
- i. **Architektur:** 8664 machine (x64)
 - ii. **Linker-Version:** 14.36 (typisch für Visual Studio-Version)
 - iii. **Zeitstempel:** Tue Oct 24 16:16:16 2023
 - iv. **Subsystem:** Windows CUI (Konsolenanwendung)
 - v. **DLL-Charakteristika:**
Kompatibel mit "High Entropy Virtual Addresses", "Dynamic Base", "NX (No eXecute)", "Terminal Server Aware".
Diese Eigenschaften sind Sicherheitsmerkmale, die oft in modernen Anwendungen verwendet werden.
 - vi. **Speicherreservierung:** Die Größen für Stack- und Heap-Reservierung und -Commit sind angegeben, was Hinweise auf den für die Ausführung der Anwendung benötigten Speicher gibt.
 - vii. **Verzeichnisse und Sektionen:** Verschiedene Verzeichnisse wie das Importverzeichnis, das Resourceverzeichnis, das Exceptionverzeichnis usw. sind aufgelistet. Diese geben Informationen über die Struktur der ausführbaren Datei und welche externen Funktionen oder Ressourcen sie verwendet.
 - viii. **Betriebssystem- und Subsystem-Version:** Das Betriebssystem und das Subsystem sind auf Version 6.00 eingestellt. Dies könnte auf eine Kompatibilität mit bestimmten Windows-Versionen hinweisen (z.B. Windows Vista, 7, 8, 10), die alle NT 6.x-Versionen sind.
 - ix. **Art der Ausführbaren:** Die Ausführbare ist als "Executable" markiert, was darauf hindeutet, dass es sich um eine Standard-Ausführbare (und nicht um eine DLL) handelt.

Unterschiede in den Versionen finden sich in der Anzahl der Sektionen, dem Zeitstempel, der Größe des Codes, der Relativ Virtuellen Adresse (RVA) sowie der Größe verschiedener Verzeichnisse (Import Directory, Resource Directory, Exception Directory, Base Relocation Directory, Debug Directory, Load Configuration Directory, Import Address Table Directory) wieder. Ebenfalls unterscheidet sich die Anzahl der Verzeichnisse.

DEPENDENCIES

Durch den Aufruf von `dumpbin /DEPENDENCIES <PE-filename>` können mehrere Informationen über die Umgebung herausgefunden werden:

```
PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases> dumpbin /dependents .\infected-mt.exe
Microsoft (R) COFF/PE Dumper Version 14.37.32825.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-mt.exe

File Type: EXECUTABLE IMAGE

Image has the following dependencies:

    KERNEL32.dll

Summary

    2000 .data
    2000 .pdata
    5000 .rdata
    1000 .reloc
    1000 .rsrc
    1000 text
    1000 .DATA

PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases>
```

(a) dependencies mt-file

```
PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases> dumpbin /dependents .\infected-mtd.exe
Microsoft (R) COFF/PE Dumper Version 14.37.32825.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-mtd.exe

File Type: EXECUTABLE IMAGE

Image has the following dependencies:

    KERNEL32.dll

Summary

    3000 .data
    4000 .pdata
    15000 .rdata
    1000 .reloc
    1000 .rsrc
    43000 text
    1000 .DATA

PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases>
```

(b) dependencies mtd-file

```
PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases> dumpbin /dependents .\infected-md.exe
Microsoft (R) COFF/PE Dumper Version 14.37.32825.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-md.exe

File Type: EXECUTABLE IMAGE

Image has the following dependencies:

    VCRUNTIME140.dll
    api-ms-win-crt-stdio-l1-1-0.dll
    api-ms-win-crt-runtime-l1-1-0.dll
    api-ms-win-crt-math-l1-1-0.dll
    api-ms-win-crt-locale-l1-1-0.dll
    api-ms-win-crt-heap-l1-1-0.dll
    KERNEL32.dll

Summary

    1000 .data
    1000 .pdata
    1000 .rdata
    1000 .reloc
    1000 .rsrc
    1000 text

PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases>
```

(c) dependencies md-file

```
PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases> dumpbin /dependents .\infected-mdd.exe
Microsoft (R) COFF/PE Dumper Version 14.37.32825.0
Copyright (C) Microsoft Corporation. All rights reserved.

Dump of file .\infected-mdd.exe

File Type: EXECUTABLE IMAGE

Image has the following dependencies:

    VCRUNTIME140.dll
    ucrtbased.dll
    KERNEL32.dll

Summary

    1000 .data
    1000 .pdata
    1000 .rdata
    1000 .reloc
    1000 .rsrc
    2000 text

PS C:\Users\Quickemu\repos\REV3\UE02\rev3-s2210239021\windows\releases>
```

(d) dependencies mdd-file

Figure 2: All dependencies

Die Unterschiedlichen Abhängigkeiten geben ebenfalls Auskunft über die Umgebung.

Weitere Informationen zur Umgebung

Es können natürlich noch weitere Informationen über die Umgebung gefunden werden, hilfreich können auch die Parameter `/DEBUG`, `/IMPORTS` oder `/RESSOURCES` sein, es kommt jedoch immer darauf an, wonach gesucht wird.

2. Aufgabe - Statische Analyse Linux

create file

Gleicher code wie zuvor:

```
1  #include <stdio.h>
2
3  int main() {
4      printf("infected");
5      return = 0;
6  }
7
```

Kompilieren und auflisten der Executables:

```
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ gcc -std=c99 -Wall -pedantic infected.c -o infected.out time:1ms
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ gcc -std=c99 -Wall -pedantic infected.c -ggdb -o infected-ggdb.out
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ gcc -std=c99 -Wall -pedantic infected.c -static -o infected-static.out
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ gcc -std=c99 -Wall -pedantic infected.c -static -ggdb -o infected-static-ggdb.out
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ time:80ms
```

```
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ la time:1ms
total 1.6M
-rw-r--r-- 1 mendacium mendacium 69 Oct 24 16:54 infected.c
-rwxr-xr-x 1 mendacium mendacium 18K Oct 24 16:58 infected-ggdb.out*
-rwxr-xr-x 1 mendacium mendacium 17K Oct 24 16:55 infected.out*
-rwxr-xr-x 1 mendacium mendacium 777K Oct 24 17:01 infected-static-ggdb.out*
-rwxr-xr-x 1 mendacium mendacium 776K Oct 24 17:00 infected-static.out*
mendacium fedora ./REV3 > UE02 > rev3-s2210239021 > linux
→ time:1ms
```

Fragen

1. Welche Imports werden verwendet?

References

- [1] *The Official Radare2 Book*, [Online; abgerufen im Oktober 2023], <https://book.rada.re/>.