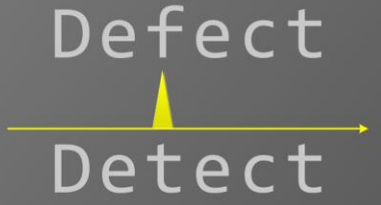

Windows Malware Analysis Accelerated

with Memory Dumps

Version 3.0

Dmitry Vostokov
Software Diagnostics Services

Published by OpenTask, Republic of Ireland

OpenTask books and magazines are available through booksellers and distributors worldwide. For further information or comments, send requests to press@opentask.com.

A CIP catalog record for this book is available from the British Library.

ISBN-l3: 978-1-912636-96-9 (Paperback)

Revision 3.00 (July 2022)

Contents

Exercise M1A

Goal: Look at module headers and version information before loading.

Patterns: Unknown Module.

1. Launch WinDbg Preview.

2. Open \AWMA-Dumps\Executables\M1.exe.

3. We get the EXE file loaded:

```
Microsoft (R) Windows Debugger Version 10.0.25136.1001 AMD64
Copyright (c) Microsoft Corporation. All rights reserved.


Loading Dump File [C:\AWMA-Dumps\Executables\M1.exe]

************* Path validation summary **************
Response                         Time (ms)     Location
Deferred                                       srv*
Symbol search path is: srv*
Executable search path is:
ModLoad: 00000001`40000000 00000001`40020000   C:\AWMA-Dumps\Executables\M1.exe
*** WARNING: Unable to verify checksum for M1.exe
M1+0x1748:
00000001`40001748 4883ec28        sub     rsp,28h
```

4. Open a log file:

```
0:000> .logopen C:\AWMA-Dumps\M1A.log
Opened log file 'C:\AWMA-Dumps\M1A.log'
```

5. **lmv** command lists module information:

```
0:000> lmv
start             end                 module name
00000001`40000000 00000001`40020000   M1       C (no symbols)
    Loaded symbol image file: M1.exe
    Mapped memory image file: C:\AWMA-Dumps\Executables\M1.exe
    Image path: C:\AWMA-Dumps\Executables\M1.exe
    Image name: M1.exe
    Browse all global symbols  functions  data
    Timestamp:        Mon Jul  4 18:01:41 2022 (62C31CF5)
    CheckSum:         00000000
    ImageSize:        00020000
    Translations:     0000.04b0 0000.04e4 0409.04b0 0409.04e4
    Information from resource tables:
```

Note module default load address.

6. **!lmi** command gives a bit more information:

```
0:000> !lmi 00000001`40000000
Loaded Module Info: [00000001`40000000]
         Module: M1
   Base Address: 0000000140000000
     Image Name: M1.exe
   Machine Type: 34404 (X64)
     Time Stamp: 62c31cf5 Mon Jul  4 18:01:41 2022
           Size: 20000
       CheckSum: 0
Characteristics: 22
Debug Data Dirs: Type  Size     VA  Pointer
             CODEVIEW    37, 16de8,   155e8 RSDS - GUID: {4DA766FB-D5ED-4091-9599-9C8098BCC72D}
               Age: 1, Pdb: C:\AWMA3\M1\x64\Release\M1.pdb
           VC_FEATURE    14, 16e20,   15620 [Data not mapped]
                 POGO   31c, 16e34,   15634 [Data not mapped]
                ILTCG     0,     0,       0  [Debug data not mapped]
     Image Type: FILE     - Image read successfully from debugger.
                 M1.exe
    Symbol Type: NONE     - PDB not found from symbol server.
    Load Report: no symbols loaded
```

Note a reference to a PDB file. If this reference is left by a developer, it might give some clues, as we see in other exercises.

7. We dump the first kilobyte:

```
0:000> dc 00000001`40000000 L100
00000001`40000000  00905a4d 00000003 00000004 0000ffff  MZ..............
00000001`40000010  000000b8 00000000 00000040 00000000  ........@.......
00000001`40000020  00000000 00000000 00000000 00000000  ................
00000001`40000030  00000000 00000000 00000000 00000110  ................
00000001`40000040  0eba1f0e cd09b400 4c01b821 685421cd  ........!..L.!Th
00000001`40000050  70207369 72676f72 63206d61 6f6e6e61  is program canno
00000001`40000060  65622074 6e757220 206e6920 20534f44  t be run in DOS
00000001`40000070  65646f6d 0a0d0d2e 00000024 00000000  mode....$.......
00000001`40000080  e5d9de50 b6b7bf14 b6b7bf14 b6b7bf14  P...............
00000001`40000090  b7b4c75f b6b7bf11 b7b2c75f b6b7bf9f  _......._.......
00000001`400000a0  b7b3c75f b6b7bf1e b7b2c574 b6b7bf3c  _.......t...<...
00000001`400000b0  b7b3c574 b6b7bf04 b7b4c574 b6b7bf1d  t.......t.......
00000001`400000c0  b7b6c75f b6b7bf11 b6b6bf14 b6b7bf79  _...........y...
00000001`400000d0  b7bec570 b6b7bf16 b648c570 b6b7bf15  p.......p.H.....
00000001`400000e0  b620bf14 b6b7bf15 b7b5c570 b6b7bf15  .. .....p.......
00000001`400000f0  68636952 b6b7bf14 00000000 00000000  Rich............
00000001`40000100  00000000 00000000 00000000 00000000  ................
00000001`40000110  00004550 00078664 62c31cf5 00000000  PE..d......b....
00000001`40000120  00000000 002200f0 200e020b 0000d400  ......".... ....
00000001`40000130  0000f200 00000000 00001748 00001000  ........H.......
00000001`40000140  40000000 00000001 00001000 00000200  ...@............
00000001`40000150  00000006 00000000 00000006 00000000  ................
00000001`40000160  00020000 00000400 00000000 81600002  .............`..
00000001`40000170  00100000 00000000 00001000 00000000  ................
00000001`40000180  00100000 00000000 00001000 00000000  ................
00000001`40000190  00000000 00000010 00000000 00000000  ................
00000001`400001a0  00017f0c 0000003c 0001d000 00001d78  ....<.......x...
00000001`400001b0  0001b000 00000f30 00000000 00000000  ....0...........
00000001`400001c0  0001f000 00000660 000169f0 00000070  ....`....i..p...
00000001`400001d0  00000000 00000000 00000000 00000000  ................
```

```
00000001`400001e0  00000000 00000000 000168b0 00000140  .........h..@...
00000001`400001f0  00000000 00000000 0000f000 000002e8  ................
00000001`40000200  00000000 00000000 00000000 00000000  ................
00000001`40000210  00000000 00000000 7865742e 00000074  .........text...
00000001`40000220  0000d230 00001000 0000d400 00000400  0...............
00000001`40000230  00000000 00000000 00000000 60000020  ............ ..`
00000001`40000240  6164722e 00006174 000098ac 0000f000  .rdata..........
00000001`40000250  00009a00 0000d800 00000000 00000000  ................
00000001`40000260  00000000 40000040 7461642e 00000061  ....@..@.data...
00000001`40000270  00001ec0 00019000 00000c00 00017200  ..............r..
00000001`40000280  00000000 00000000 00000000 c0000040  ............@...
00000001`40000290  6164702e 00006174 00000f30 0001b000  .pdata..0.......
00000001`400002a0  00001000 00017e00 00000000 00000000  .....~..........
00000001`400002b0  00000000 40000040 4144525f 00004154  ....@..@_RDATA..
00000001`400002c0  0000015c 0001c000 00000200 00018e00  \...............
00000001`400002d0  00000000 00000000 00000000 40000040  ............@..@
00000001`400002e0  7273722e 00000063 00001d78 0001d000  .rsrc...x.......
00000001`400002f0  00001e00 00019000 00000000 00000000  ................
00000001`40000300  00000000 40000040 6c65722e 0000636f  ....@..@.reloc..
00000001`40000310  00000660 0001f000 00000800 0001ae00  `...............
00000001`40000320  00000000 00000000 00000000 42000040  ............@..B
00000001`40000330  00000000 00000000 00000000 00000000  ................
00000001`40000340  00000000 00000000 00000000 00000000  ................
00000001`40000350  00000000 00000000 00000000 00000000  ................
00000001`40000360  00000000 00000000 00000000 00000000  ................
00000001`40000370  00000000 00000000 00000000 00000000  ................
00000001`40000380  00000000 00000000 00000000 00000000  ................
00000001`40000390  00000000 00000000 00000000 00000000  ................
00000001`400003a0  00000000 00000000 00000000 00000000  ................
00000001`400003b0  00000000 00000000 00000000 00000000  ................
00000001`400003c0  00000000 00000000 00000000 00000000  ................
00000001`400003d0  00000000 00000000 00000000 00000000  ................
00000001`400003e0  00000000 00000000 00000000 00000000  ................
00000001`400003f0  00000000 00000000 00000000 00000000  ................
```

8. **!dh** command dumps PE header:

```
0:000> !dh 00000001`40000000


File Type: EXECUTABLE IMAGE
FILE HEADER VALUES
    8664 machine (X64)
       7 number of sections
62C31CF5 time date stamp Mon Jul  4 18:01:41 2022

       0 file pointer to symbol table
       0 number of symbols
      F0 size of optional header
      22 characteristics
            Executable
            App can handle >2gb addresses

OPTIONAL HEADER VALUES
     20B magic #
   14.32 linker version
    D400 size of code
    F200 size of initialized data
       0 size of uninitialized data
    1748 address of entry point
```

```
    1000 base of code
         ----- new -----
0000000140000000 image base
    1000 section alignment
     200 file alignment
       2 subsystem (Windows GUI)
    6.00 operating system version
    0.00 image version
    6.00 subsystem version
   20000 size of image
     400 size of headers
       0 checksum
0000000000100000 size of stack reserve
0000000000001000 size of stack commit
0000000000100000 size of heap reserve
0000000000001000 size of heap commit
    8160  DLL characteristics
            High entropy VA supported
            Dynamic base
            NX compatible
            Terminal server aware
       0 [       0] address [size] of Export Directory
   17F0C [      3C] address [size] of Import Directory
   1D000 [    1D78] address [size] of Resource Directory
   1B000 [     F30] address [size] of Exception Directory
       0 [       0] address [size] of Security Directory
   1F000 [     660] address [size] of Base Relocation Directory
   169F0 [      70] address [size] of Debug Directory
       0 [       0] address [size] of Description Directory
       0 [       0] address [size] of Special Directory
       0 [       0] address [size] of Thread Storage Directory
   168B0 [     140] address [size] of Load Configuration Directory
       0 [       0] address [size] of Bound Import Directory
    F000 [     2E8] address [size] of Import Address Table Directory
       0 [       0] address [size] of Delay Import Directory
       0 [       0] address [size] of COR20 Header Directory
       0 [       0] address [size] of Reserved Directory


SECTION HEADER #1
   .text name
    D230 virtual size
    1000 virtual address
    D400 size of raw data
     400 file pointer to raw data
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
60000020 flags
         Code
         (no align specified)
         Execute Read

SECTION HEADER #2
  .rdata name
    98AC virtual size
    F000 virtual address
    9A00 size of raw data
    D800 file pointer to raw data
```

```
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
40000040 flags
         Initialized Data
         (no align specified)
         Read Only


Debug Directories(4)
       Type       Size     Address  Pointer
       cv           37       16de8    155e8    Format: RSDS, guid, 1,
C:\AWMA3\M1\x64\Release\M1.pdb
       (   12)      14       16e20    15620
       (   13)     31c       16e34    15634
       (   14)       0           0        0

SECTION HEADER #3
   .data name
    1EC0 virtual size
   19000 virtual address
     C00 size of raw data
   17200 file pointer to raw data
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
C0000040 flags
         Initialized Data
         (no align specified)
         Read Write

SECTION HEADER #4
  .pdata name
     F30 virtual size
   1B000 virtual address
    1000 size of raw data
   17E00 file pointer to raw data
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
40000040 flags
         Initialized Data
         (no align specified)
         Read Only

SECTION HEADER #5
  _RDATA name
     15C virtual size
   1C000 virtual address
     200 size of raw data
   18E00 file pointer to raw data
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
40000040 flags
         Initialized Data
```

```
         (no align specified)
         Read Only

SECTION HEADER #6
   .rsrc name
    1D78 virtual size
   1D000 virtual address
    1E00 size of raw data
   19000 file pointer to raw data
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
40000040 flags
         Initialized Data
         (no align specified)
         Read Only

SECTION HEADER #7
  .reloc name
     660 virtual size
   1F000 virtual address
     800 size of raw data
   1AE00 file pointer to raw data
       0 file pointer to relocation table
       0 file pointer to line numbers
       0 number of relocations
       0 number of line numbers
42000040 flags
         Initialized Data
         Discardable
         (no align specified)
         Read Only
```

Note **Import Directory, Import Address Table Directory,** and code **.text** section.

9. Let's look at **Import Address Table Directory** before dynamic linking takes place:

```
0:000> dps 00000001`40000000+F000
00000001`4000f000  ????????`????????
00000001`4000f008  ????????`????????
00000001`4000f010  ????????`????????
00000001`4000f018  ????????`????????
00000001`4000f020  ????????`????????
00000001`4000f028  ????????`????????
00000001`4000f030  ????????`????????
00000001`4000f038  ????????`????????
00000001`4000f040  ????????`????????
00000001`4000f048  ????????`????????
00000001`4000f050  ????????`????????
00000001`4000f058  ????????`????????
00000001`4000f060  ????????`????????
00000001`4000f068  ????????`????????
00000001`4000f070  ????????`????????
00000001`4000f078  ????????`????????
```

We see it is inaccessible or not present. However, **Import Directory** is available, and we can dump its contents using the module image address, relative offset, and size (in bytes). It is an array of structures, each of 5 double words (4 bytes per double word). This is why we use the **dd** command and divide the size by 4:

```
0:000> dd 00000001`40000000+17F0C L3C/4
00000001`40017f0c  00017f48 00000000 00000000 00018240
00000001`40017f1c  0000f000 00018190 00000000 00000000
00000001`40017f2c  00018388 0000f248 00000000 00000000
00000001`40017f3c  00000000 00000000 00000000
```

The first double word in each structure is a relative offset to a relative offset to an array of names such as function names, and the fourth double word is a relative offset to an import DLL name:

```
0:000> da 00000001`40000000+00018240
00000001`40018240  "KERNEL32.dll"

0:000> da 00000001`40000000+00018388
00000001`40018388  "USER32.dll"
```

We now examine function names to be imported from *KERNEL32.dll*:

```
0:000> dp 00000001`40000000+00017f48
00000001`40017f48  00000000`00018230 00000000`0001889c
00000001`40017f58  00000000`0001888e 00000000`00018880
00000001`40017f68  00000000`0001886c 00000000`0001885a
00000001`40017f78  00000000`00018844 00000000`00018830
00000001`40017f88  00000000`00018822 00000000`00018816
00000001`40017f98  00000000`00018804 00000000`000187f4
00000001`40017fa8  00000000`000187ea 00000000`000187dc
00000001`40017fb8  00000000`000187ce 00000000`00018394

0:000> dc 00000001`40000000+00000000`00018230 L100
00000001`40018230  6f4c03e7 694c6461 72617262 00005779  ..LoadLibraryW..
00000001`40018240  4e52454b 32334c45 6c6c642e 02680000  KERNEL32.dll..h.
00000001`40018250  64616f4c 69727453 0057676e 6f4c0253  LoadStringW.S.Lo
00000001`40018260  63416461 656c6563 6f746172 00577372  adAcceleratorsW.
00000001`40018270  6547018b 73654d74 65676173 03b40057  ..GetMessageW...
00000001`40018280  6e617254 74616c73 63634165 72656c65  TranslateAcceler
00000001`40018290  726f7461 03b60057 6e617254 74616c73  atorW...Translat
00000001`400182a0  73654d65 65676173 00bd0000 70736944  eMessage....Disp
00000001`400182b0  68637461 7373654d 57656761 025b0000  atchMessageW..[.
00000001`400182c0  64616f4c 6e6f6349 02590057 64616f4c  LoadIconW.Y.Load
00000001`400182d0  73727543 0057726f 655202df 74736967  CursorW...Regist
00000001`400182e0  6c437265 45737361 00005778 72430076  erClassExW..v.Cr
00000001`400182f0  65746165 646e6957 7845776f 03960057  eateWindowExW...
00000001`40018300  776f6853 646e6957 0000776f 705503d0  ShowWindow....Up
00000001`40018310  65746164 646e6957 0000776f 694400ba  dateWindow....Di
00000001`40018320  676f6c61 50786f42 6d617261 00b50057  alogBoxParamW...
00000001`40018330  74736544 57796f72 6f646e69 00a70077  DestroyWindow...
00000001`40018340  57666544 6f646e69 6f725077 00005763  DefWindowProcW..
00000001`40018350  65420011 506e6967 746e6961 00f40000  ..BeginPaint....
00000001`40018360  50646e45 746e6961 02af0000 74736f50  EndPaint....Post
00000001`40018370  74697551 7373654d 00656761 6e4500f2  QuitMessage...En
00000001`40018380  61694464 00676f6c 52455355 642e3233  dDialog.USER32.d
00000001`40018390  00006c6c 745204f5 7061436c 65727574  ll....RtlCapture
00000001`400183a0  746e6f43 00747865 745204fd 6f6f4c6c  Context...RtlLoo
00000001`400183b0  4670756b 74636e75 456e6f69 7972746e  kupFunctionEntry
00000001`400183c0  05040000 566c7452 75747269 6e556c61  ....RtlVirtualUn
00000001`400183d0  646e6977 05e60000 61686e55 656c646e  wind....Unhandle
00000001`400183e0  63784564 69747065 69466e6f 7265746c  dExceptionFilter
00000001`400183f0  05a40000 55746553 6e61686e 64656c64  ....SetUnhandled
00000001`40018400  65637845 6f697470 6c69466e 00726574  ExceptionFilter.
00000001`40018410  65470232 72754374 746e6572 636f7250  2.GetCurrentProc
00000001`40018420  00737365 655405c4 6e696d72 50657461  ess...TerminateP
```

```
00000001`40018430  65636f72 00007373 734903a8 636f7250  rocess....IsProc
00000001`40018440  6f737365 61654672 65727574 73657250  essorFeaturePres
00000001`40018450  00746e65 75510470 50797265 6f667265  ent.p.QueryPerfo
00000001`40018460  6e616d72 6f436563 65746e75 02330072  rmanceCounter.3.
00000001`40018470  43746547 65727275 7250746e 7365636f  GetCurrentProces
00000001`40018480  00644973 65470237 72754374 746e6572  sId.7.GetCurrent
00000001`40018490  65726854 64496461 030a0000 53746547  ThreadId....GetS
00000001`400184a0  65747379 6d69546d 46734165 54656c69  ystemTimeAsFileT
00000001`400184b0  00656d69 6e49038a 61697469 657a696c  ime...Initialize
00000001`400184c0  73694c53 61654874 03a00064 65447349  SListHead...IsDe
00000001`400184d0  67677562 72507265 6e657365 02f10074  buggerPresent...
00000001`400184e0  53746547 74726174 6e497075 00576f66  GetStartupInfoW.
00000001`400184f0  65470295 646f4d74 48656c75 6c646e61  ..GetModuleHandl
00000001`40018500  00005765 74520503 776e556c 45646e69  eW....RtlUnwindE
00000001`40018510  027d0078 4c746547 45747361 726f7272  x.}.GetLastError
00000001`40018520  05640000 4c746553 45747361 726f7272  ..d.SetLastError
00000001`40018530  01490000 65746e45 69724372 61636974  ..I.EnterCritica
00000001`40018540  6365536c 6e6f6974 03e00000 7661654c  lSection....Leav
00000001`40018550  69724365 61636974 6365536c 6e6f6974  eCriticalSection
00000001`40018560  01230000 656c6544 72436574 63697469  ..#.DeleteCritic
00000001`40018570  65536c61 6f697463 0386006e 74696e49  alSection...Init
00000001`40018580  696c6169 7243657a 63697469 65536c61  ializeCriticalSe
00000001`40018590  6f697463 646e416e 6e697053 6e756f43  ctionAndSpinCoun
00000001`400185a0  05d60074 41736c54 636f6c6c 05d80000  t...TlsAlloc....
00000001`400185b0  47736c54 61567465 0065756c 6c5405d9  TlsGetValue...Tl
00000001`400185c0  74655373 756c6156 05d70065 46736c54  sSetValue...TlsF
00000001`400185d0  00656572 724601c5 694c6565 72617262  ree...FreeLibrar
00000001`400185e0  02cd0079 50746547 41636f72 65726464  y...GetProcAddre
00000001`400185f0  00007373 6f4c03e6 694c6461 72617262  ss....LoadLibrar
00000001`40018600  57784579 01450000 6f636e45 6f506564  yExW..E.EncodePo
00000001`40018610  65746e69 04870072 73696152 63784565  inter...RaiseExc
00000001`40018620  69747065 00006e6f 745204ff 5463506c  eption....RtlPcT
```

We can also get offsets by using **-i** or **-a** options for **!dh** command:

```
0:000> !dh -i 00000001`40000000
  _IMAGE_IMPORT_DESCRIPTOR 0000000140017f0c
    KERNEL32.dll
      000000014000F000 Import Address Table
      0000000140017F48 Import Name Table
                     0 time date stamp
                     0 Index of first forwarder reference


  _IMAGE_IMPORT_DESCRIPTOR 0000000140017f20
    USER32.dll
      000000014000F248 Import Address Table
      0000000140018190 Import Name Table
                     0 time date stamp
                     0 Index of first forwarder reference
```

10. Close the log file:

```
0:000> .logclose
Closing open log file C:\AWMA-Dumps\M1A.log
```

To avoid possible confusion and glitches, we recommend exiting WinDbg Preview or WinDbg after each exercise.