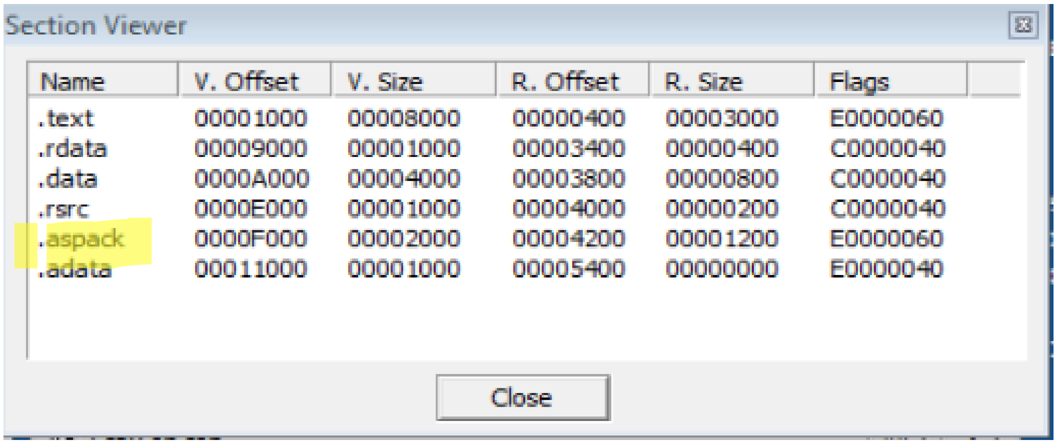
This is the next section in the lab for Module 6. Feedback is greatly appreciated.

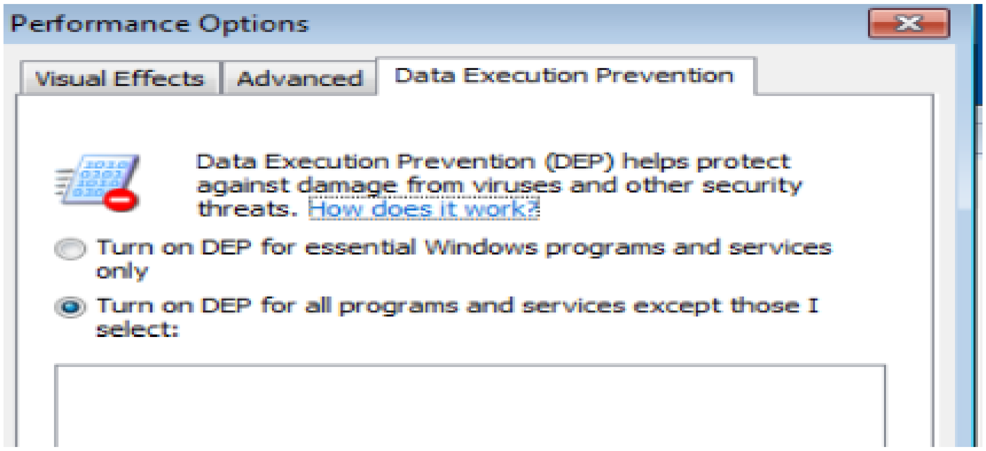
The Ixshe_aspack.bin file appears to be packed as indicated by the .aspack section name, by clicking on the EP Section



Section Viewer

Name	V. Offset	V. Size	R. Offset	R. Size	Flags
.text	00001000	00008000	00000400	00003000	E0000060
.rdata	00009000	00001000	00003400	00000400	C0000040
.data	0000A000	00004000	00003800	00000800	C0000040
.rsrc	0000E000	00001000	00004000	00000200	C0000040
.aspack	0000F000	00002000	00004200	00001200	E0000060
.adata	00011000	00001000	00005400	00000000	E0000040

For the next step, I turn on the Data Execuition Prevention, for all the programs.



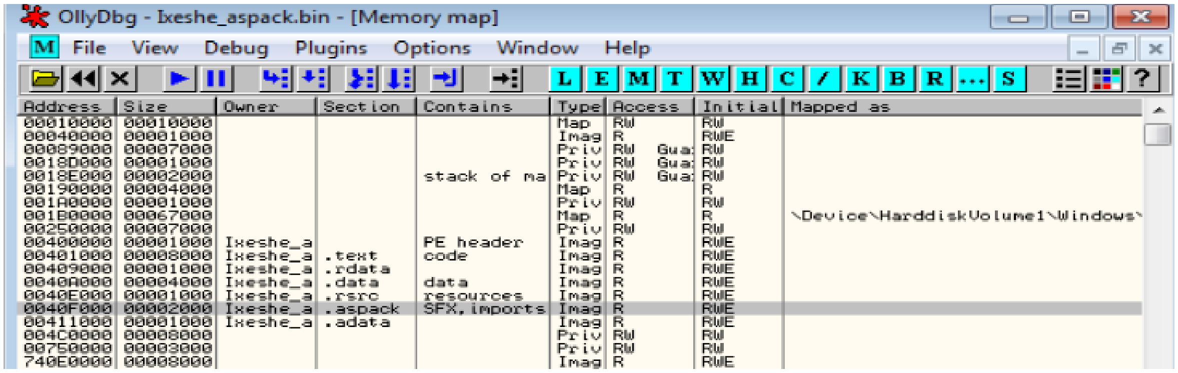
Performance Options

Data Execution Prevention (DEP) helps protect against damage from viruses and other security threats. How does it work?

○ Turn on DEP for essential Windows programs and services only

● Turn on DEP for all programs and services except those I select:

Next, I place the Ixshe file in OllyDBG to see if its, readonly, readwrite, and/or executable


OllyDbg - Ixeshe_aspack.bin - [Memory map]
M File View Debug Plugins Options Window Help
L E M T W H C / K B R ... S ?

```
Address  Size     Owner     Section  Contains     Type Access    Initial Mapped as
00010000 00010000                                 Map  RW        RW
00040000 00001000                                 Imag R         RWE
00089000 00007000                                 Priv RW   Gua: RW
0018D000 00001000                                 Priv RW   Gua: RW
0018E000 00002000                    stack of ma  Priv RW   Gua: RW
00190000 00004000                                 Map  R         R
001A0000 00001000                                 Priv RW        RW
001B0000 00067000                                 Map  R         R       \Device\HarddiskVolume1\Windows\
00250000 00007000                                 Priv RW        RW
00400000 00001000 Ixeshe_a           PE header    Imag R         RWE
00401000 00008000 Ixeshe_a  .text    code         Imag R         RWE
00409000 00001000 Ixeshe_a  .rdata                Imag R         RWE
0040A000 00004000 Ixeshe_a  .data    data         Imag R         RWE
0040E000 00001000 Ixeshe_a  .rsrc    resources    Imag R         RWE
0040F000 00002000 Ixeshe_a  .aspack  SFX,imports  Imag R         RWE
00411000 00001000 Ixeshe_a  .adata                Imag R         RWE
004C0000 00008000                                 Priv RW        RW
00750000 00003000                                 Priv RW        RW
740E0000 00008000                                 Imag R         RWE
```

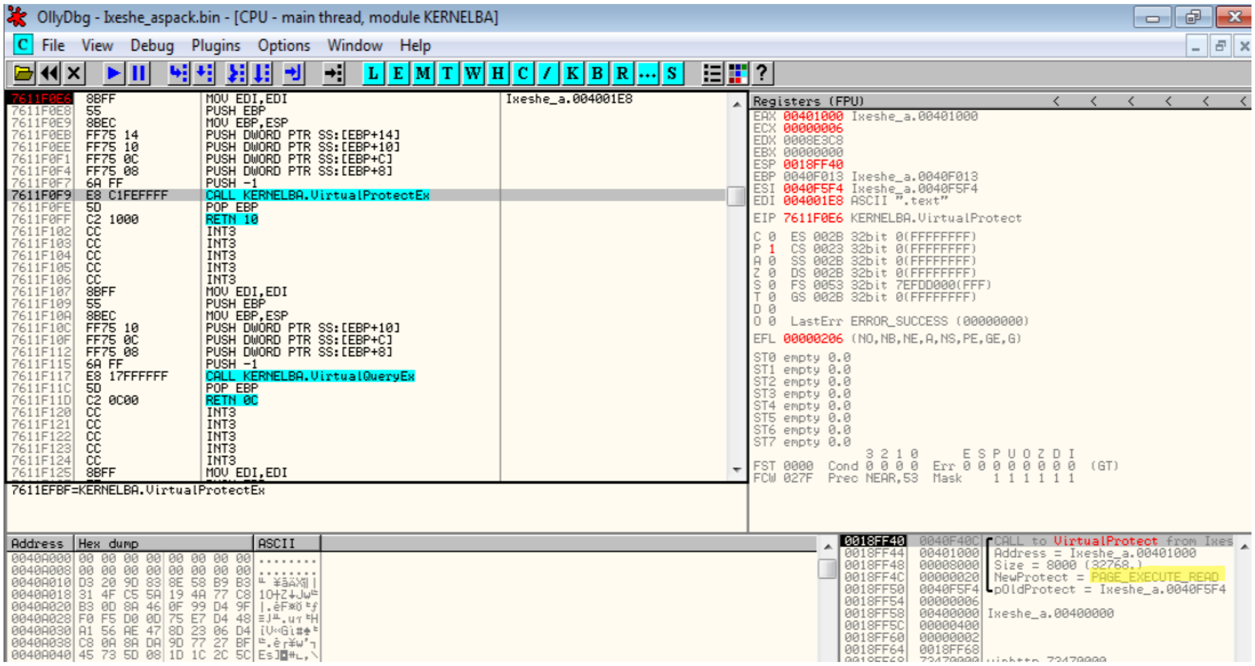
It looks like the .aspack file is Readonly

The entry point is the .aspack, the file appears to be packed, and is readonly

At this point, I began to struggle as far as using the step over/into commands, so I had to go to solution. As I can see I was beginning to go down a wrong path. I needed to set breakpoints, which help with these commands.

I began to search for the location of the VirtualProtect

Then I clicked the run command until I was able to get the Page_Execute_Read. (Success!)


OllyDbg - Ixeshe_aspack.bin - [CPU - main thread, module KERNELBA]
C File View Debug Plugins Options Window Help
L E M T W H C / K B R ... S ?

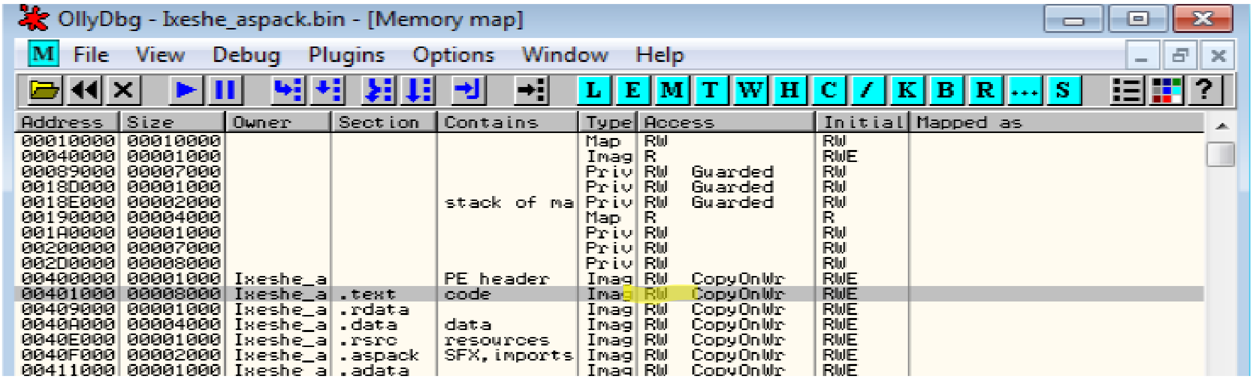
```
7611F0E6  8BFF           MOV EDI,EDI                          Ixeshe_a.004001E8
7611F0E8  55             PUSH EBP
7611F0E9  8BEC           MOV EBP,ESP
7611F0EB  FF75 14        PUSH DWORD PTR SS:[EBP+14]
7611F0EE  FF75 10        PUSH DWORD PTR SS:[EBP+10]
7611F0F1  FF75 0C        PUSH DWORD PTR SS:[EBP+C]
7611F0F4  FF75 08        PUSH DWORD PTR SS:[EBP+8]
7611F0F7  6A FF          PUSH -1
7611F0F9  E8 C1FEFFFF    CALL KERNELBA.VirtualProtectEx
7611F0FE  5D             POP EBP
7611F0FF  C2 1000        RETN 10
7611F102  CC             INT3
7611F103  CC             INT3
7611F104  CC             INT3
7611F105  CC             INT3
7611F106  CC             INT3
7611F107  8BFF           MOV EDI,EDI
7611F109  55             PUSH EBP
7611F10A  8BEC           MOV EBP,ESP
7611F10C  FF75 10        PUSH DWORD PTR SS:[EBP+10]
7611F10F  FF75 0C        PUSH DWORD PTR SS:[EBP+C]
7611F112  FF75 08        PUSH DWORD PTR SS:[EBP+8]
7611F115  6A FF          PUSH -1
7611F117  E8 17FFFFFF    CALL KERNELBA.VirtualQueryEx
7611F11C  5D             POP EBP
7611F11D  C2 0C00        RETN 0C
7611F120  CC             INT3
7611F121  CC             INT3
7611F122  CC             INT3
7611F123  CC             INT3
7611F124  CC             INT3
7611F125  8BFF           MOV EDI,EDI
7611EFBF=KERNELBA.VirtualProtectEx
```

```
Registers (FPU)
EAX 00401000 Ixeshe_a.00401000
ECX 00000006
EDX 0008E3C8
EBX 00000000
ESP 0018FF40
EBP 0040F013 Ixeshe_a.0040F013
ESI 0040F5F4 Ixeshe_a.0040F5F4
EDI 004001E8 ASCII ".text"

EIP 7611F0E6 KERNELBA.VirtualProtect

C 0  ES 002B 32bit 0(FFFFFFFF)
P 1  CS 0023 32bit 0(FFFFFFFF)
A 0  SS 002B 32bit 0(FFFFFFFF)
Z 0  DS 002B 32bit 0(FFFFFFFF)
S 0  FS 0053 32bit 7EFDD000(FFF)
T 0  GS 002B 32bit 0(FFFFFFFF)
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00000206 (NO,NB,NE,A,NS,PE,GE,G)

ST0 empty 0.0
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 0.0
ST7 empty 0.0
               3 2 1 0      E S P U O Z D I
FST 0000  Cond 0 0 0 0  Err 0 0 0 0 0 0 0 0  (GT)
FCW 027F  Prec NEAR,53  Mask    1 1 1 1 1 1
```

```
Address   Hex dump                  ASCII
0040A000  00 00 00 00 00 00 00 00   ........
0040A008  00 00 00 00 00 00 00 00   ........
0040A010  D3 20 9D 83 8E 58 B9 B3   ╙ ¥âÄX╣│
0040A018  31 4F C5 5A 19 4A 77 C8   1O┼Z↓Jw╚
0040A020  B3 0D 8A 46 0F 99 D4 9F   │.èF☼Ö╘ƒ
0040A028  F0 F5 D0 0D 75 E7 D4 48   ≡J╨.uτ╘H
0040A030  A1 56 AE 47 8D 23 06 D4   íV«Gì#♠╘
0040A038  C8 0A 8A DA 9D 77 27 BF   ╚.è┌¥w'┐
0040A040  45 73 5D 08 1D 1C 2C 5C   Es]◘↔∟,\
```

```
0018FF40  0040F40C  ┌CALL to VirtualProtect from Ixes
0018FF44  00401000  │Address = Ixeshe_a.00401000
0018FF48  00008000  │Size = 8000 (32768.)
0018FF4C  00000020  │NewProtect = PAGE_EXECUTE_READ
0018FF50  0040F5F4  └pOldProtect = Ixeshe_a.0040F5F4
0018FF54  00000006
0018FF58  00400000  Ixeshe_a.00400000
0018FF5C  00000400
0018FF60  00000002
0018FF64  0018FF68
0018FF68  73470000  winhttp.73470000
```

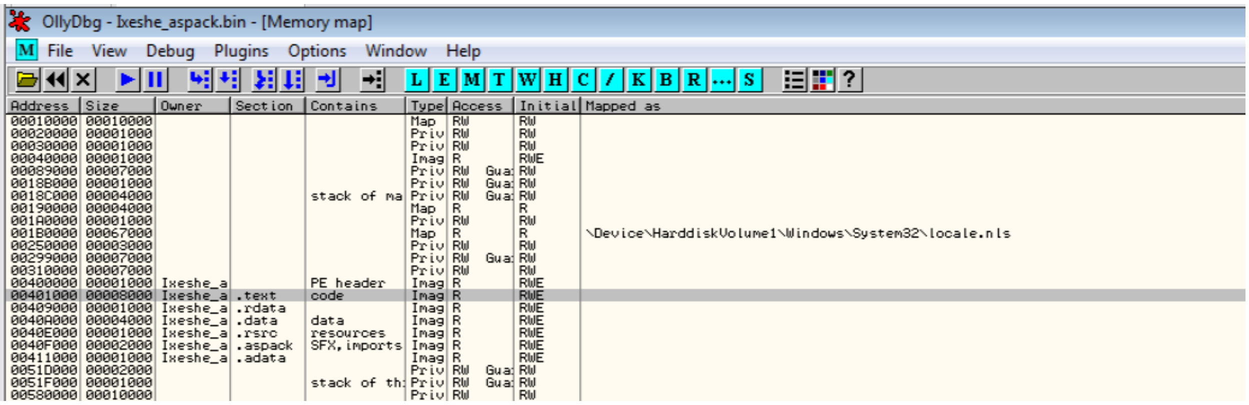
As you can see below, now the .text has been changed to RW.


OllyDbg - Ixeshe_aspack.bin - [Memory map]
M File View Debug Plugins Options Window Help
L E M T W H C / K B R ... S ?

```
Address   Size      Owner      Section   Contains      Type  Access           Initial  Mapped as
00010000  00010000                                     Map   RW               RW
00040000  00001000                                     Imag  R                RWE
00089000  00007000                                     Priv  RW  Guarded      RW
0018D000  00001000                                     Priv  RW  Guarded      RW
0018E000  00002000                       stack of ma   Priv  RW  Guarded      RW
00190000  00004000                                     Map   R                R
001A0000  00001000                                     Priv  RW               RW
00200000  00007000                                     Priv  RW               RW
002D0000  00008000                                     Priv  RW               RW
00400000  00001000  Ixeshe_a             PE header     Imag  RW  CopyOnWr     RWE
00401000  00008000  Ixeshe_a  .text      code          Imag  RW  CopyOnWr     RWE
00409000  00001000  Ixeshe_a  .rdata                   Imag  RW  CopyOnWr     RWE
0040A000  00004000  Ixeshe_a  .data      data          Imag  RW  CopyOnWr     RWE
0040E000  00001000  Ixeshe_a  .rsrc      resources     Imag  RW  CopyOnWr     RWE
0040F000  00002000  Ixeshe_a  .aspack    SFX,imports   Imag  RW  CopyOnWr     RWE
00411000  00001000  Ixeshe_a  .adata                   Imag  RW  CopyOnWr     RWE
```

Again, looking into the solution, removed the breakpoint and clicked run. Not sure where, the text message is, however. The next step is to convert the data to code by removing the analysis from the module option.

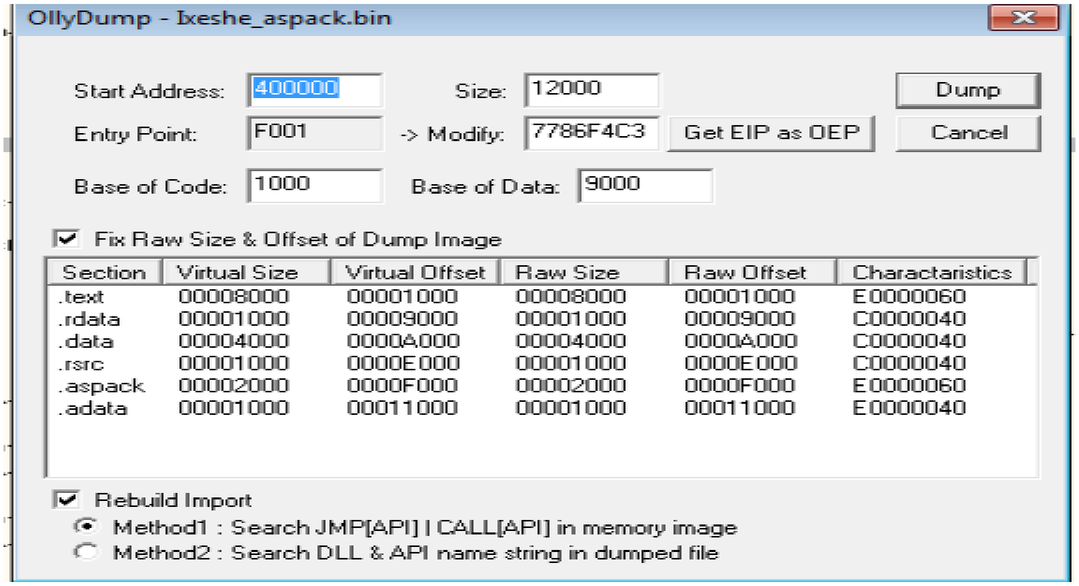
After that, I went back to the memory map, and noticed the executable had changed.



OllyDbg - Ixeshe_aspack.bin - [Memory map]

```
Address  Size     Owner     Section  Contains     Type Access   Initial Mapped as
00010000 00010000                                 Map  RW       RW
00020000 00001000                                 Priv RW       RW
00030000 00001000                                 Priv RW       RW
00040000 00001000                                 Imag R        RWE
00089000 00007000                                 Priv RW  Gua: RW
0018B000 00001000                                 Priv RW  Gua: RW
0018C000 00004000                    stack of ma  Priv RW  Gua: RW
00190000 00004000                                 Map  R        R
001A0000 00001000                                 Priv RW       RW
001B0000 00067000                                 Map  R        R       \Device\HarddiskVolume1\Windows\System32\locale.nls
00250000 00003000                                 Priv RW       RW
00299000 00007000                                 Priv RW  Gua: RW
00310000 00007000                                 Priv RW       RW
00400000 00001000 Ixeshe_a           PE header    Imag R        RWE
00401000 00008000 Ixeshe_a .text     code         Imag R        RWE
00409000 00001000 Ixeshe_a .rdata                 Imag R        RWE
0040A000 00004000 Ixeshe_a .data     data         Imag R        RWE
0040E000 00001000 Ixeshe_a .rsrc     resources    Imag R        RWE
0040F000 00002000 Ixeshe_a .aspack   SFX,imports  Imag R        RWE
00411000 00001000 Ixeshe_a .adata                 Imag R        RWE
0051D000 00002000                                 Priv RW  Gua: RW
0051F000 00001000                    stack of th: Priv RW  Gua: RW
00580000 00010000                                 Priv RW       RW
```

After that, I saved the dump using OllyDump. However, the next steps called for a tool Import REconstructor. I was not able to fine the tool in the VM.



OllyDump - Ixeshe_aspack.bin

Start Address: 400000

Size: 12000

Entry Point: F001

-> Modify: 7786F4C3

Base of Code: 1000

Base of Data: 9000

☑ Fix Raw Size & Offset of Dump Image

Section	Virtual Size	Virtual Offset	Raw Size	Raw Offset	Charactaristics
.text	00008000	00001000	00008000	00001000	E0000060
.rdata	00001000	00009000	00001000	00009000	C0000040
.data	00004000	0000A000	00004000	0000A000	C0000040
.rsrc	00001000	0000E000	00001000	0000E000	C0000040
.aspack	00002000	0000F000	00002000	0000F000	E0000060
.adata	00001000	00011000	00001000	00011000	E0000040

☑ Rebuild Import

● Method1 : Search JMP[API] | CALL[API] in memory image

○ Method2 : Search DLL & API name string in dumped file