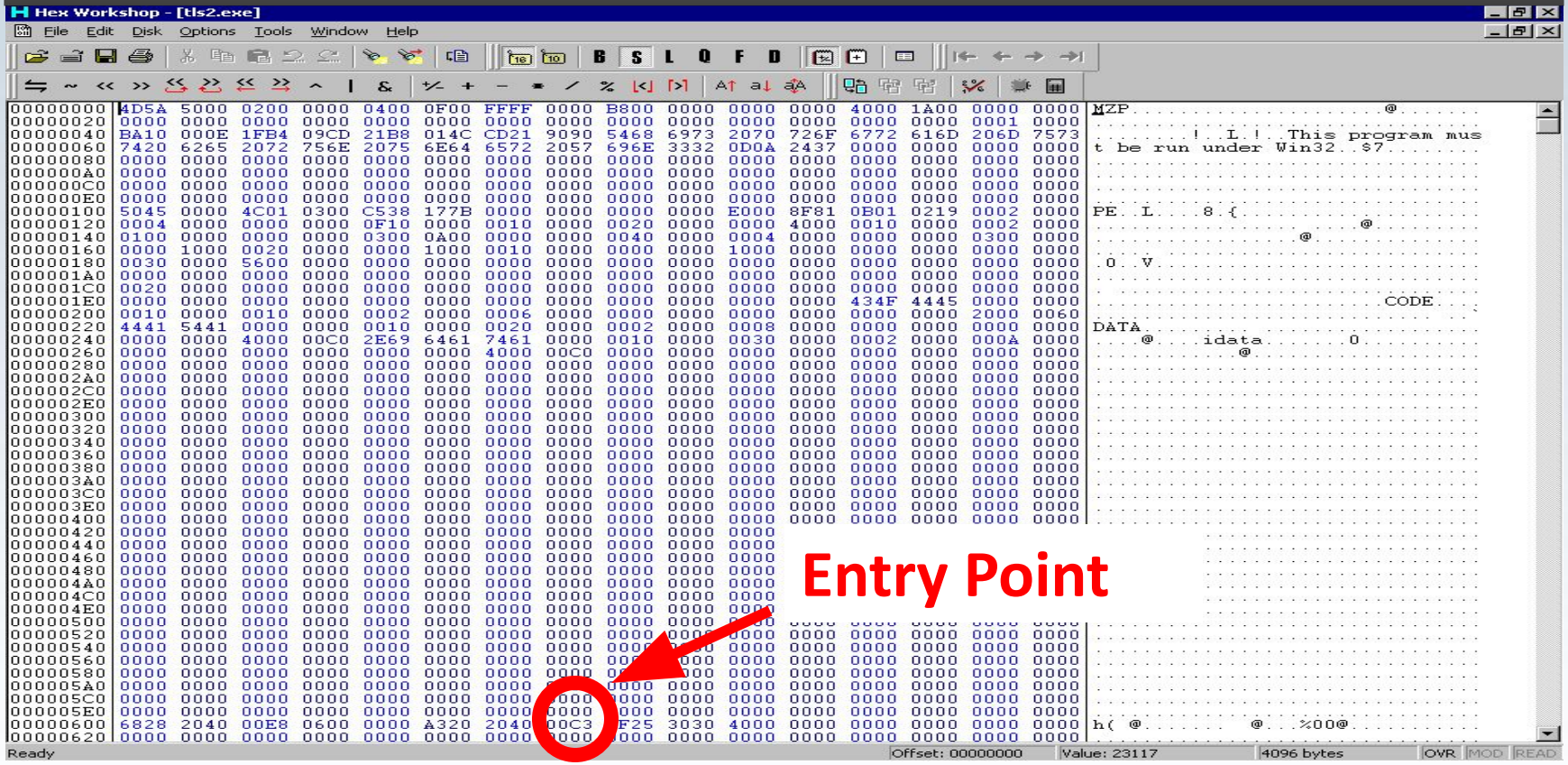
Fun With Thread Local Sto

Peter Ferrie
Senior Anti-virus Researcher
26 June, 2008

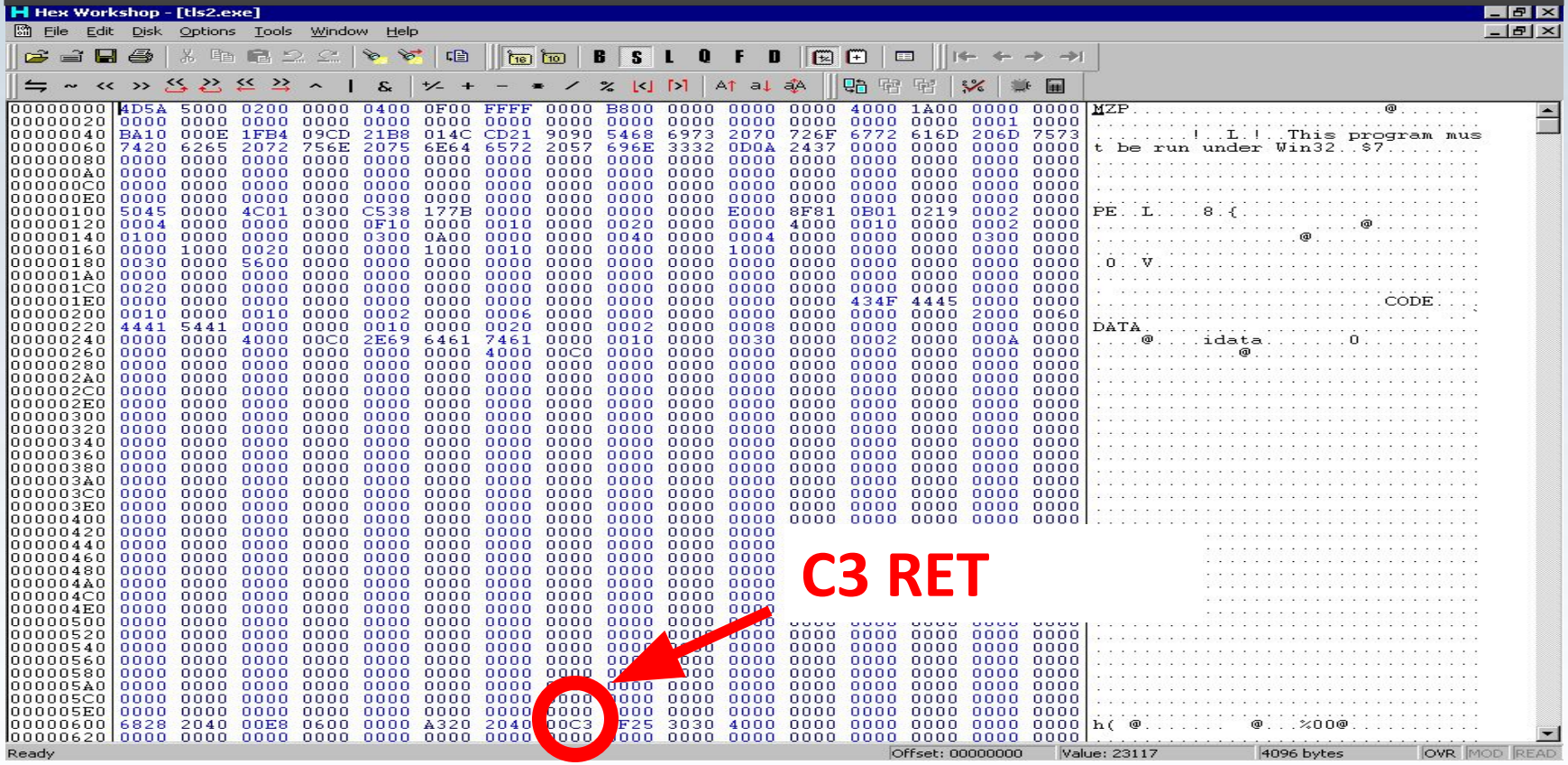
You Can Call Me Al

Thread Local Storage callbacks were discovered in 2000.
However, widespread use didn't occur until 2004.
Now, it should be the first place to look for code,
since it runs before the main entrypoint.
And that can make all the difference…

Empty!


Hex Workshop - [tls2.exe]
File Edit Disk Options Tools Window Help
Entry Point
Ready
Offset: 00000000
Value: 23117
4096 bytes
OVR MOD READ

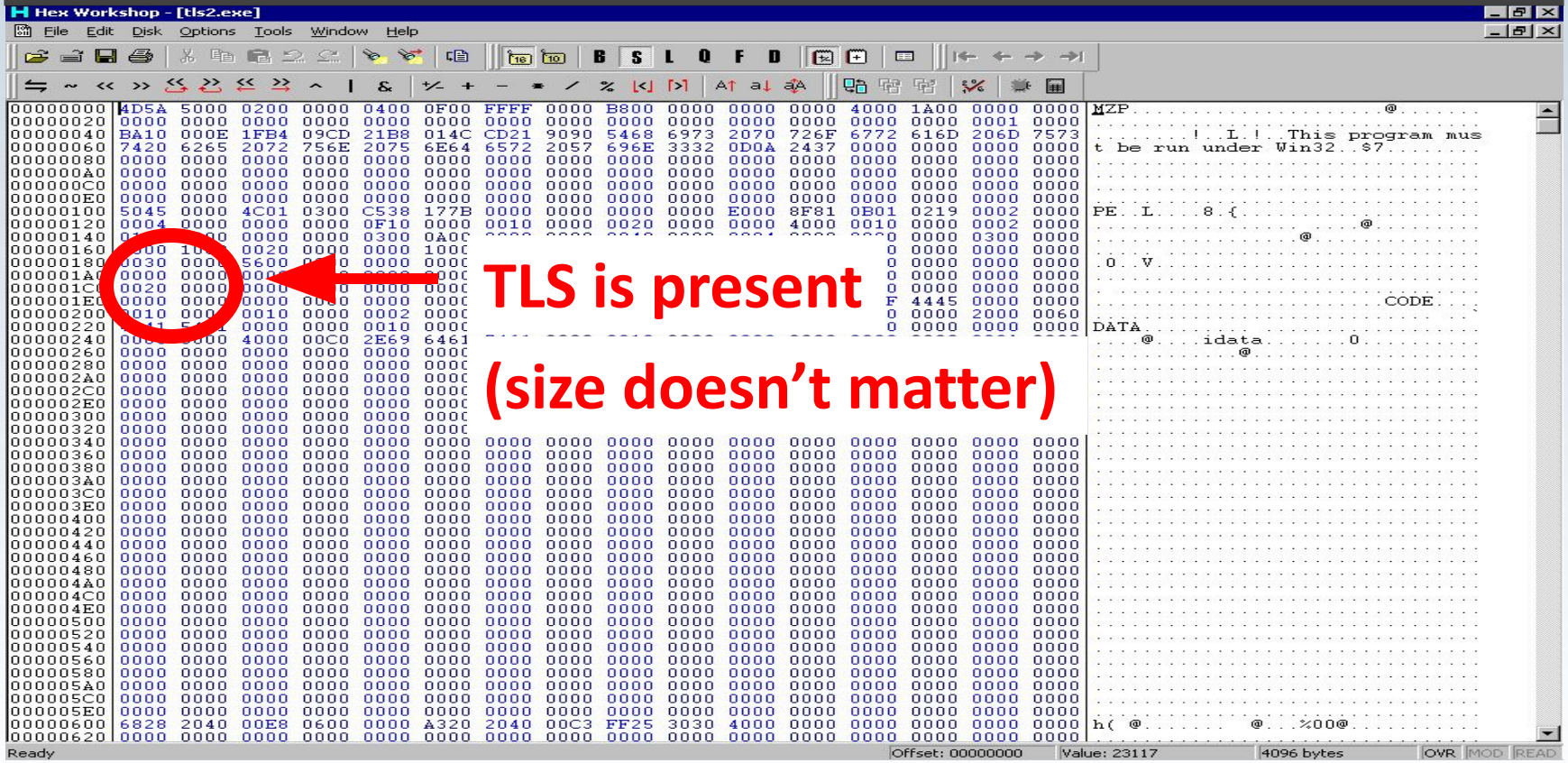
Empty!


Hex Workshop - [tls2.exe]
File Edit Disk Options Tools Window Help
C3 RET
Ready
Offset: 00000000
Value: 23117
4096 bytes
OVR MOD READ

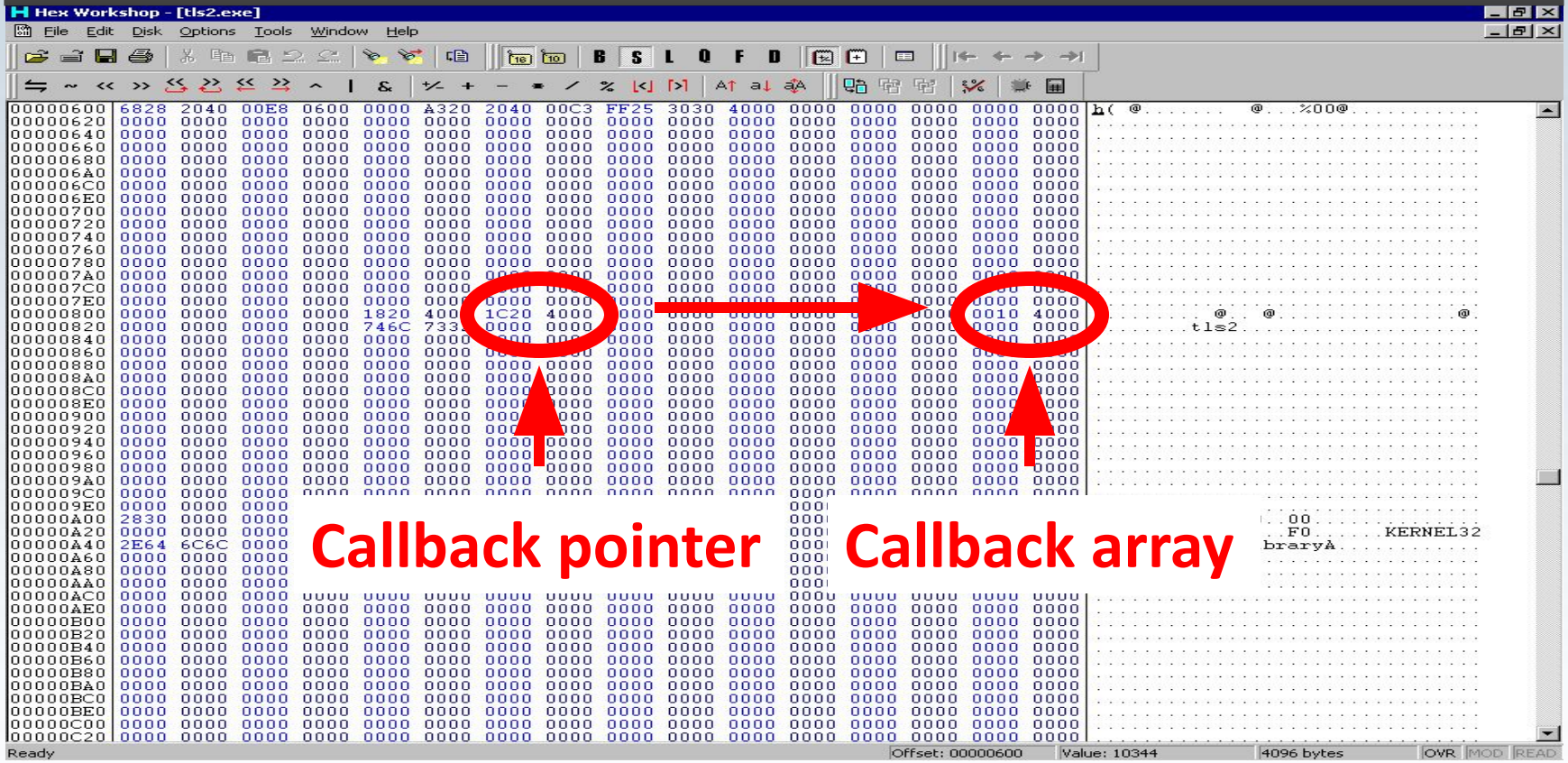
Empty!

So the main file does nothing.
If we assume that the structure is normal,
then we could check the thread local storage table.
Just in case.

Empty!

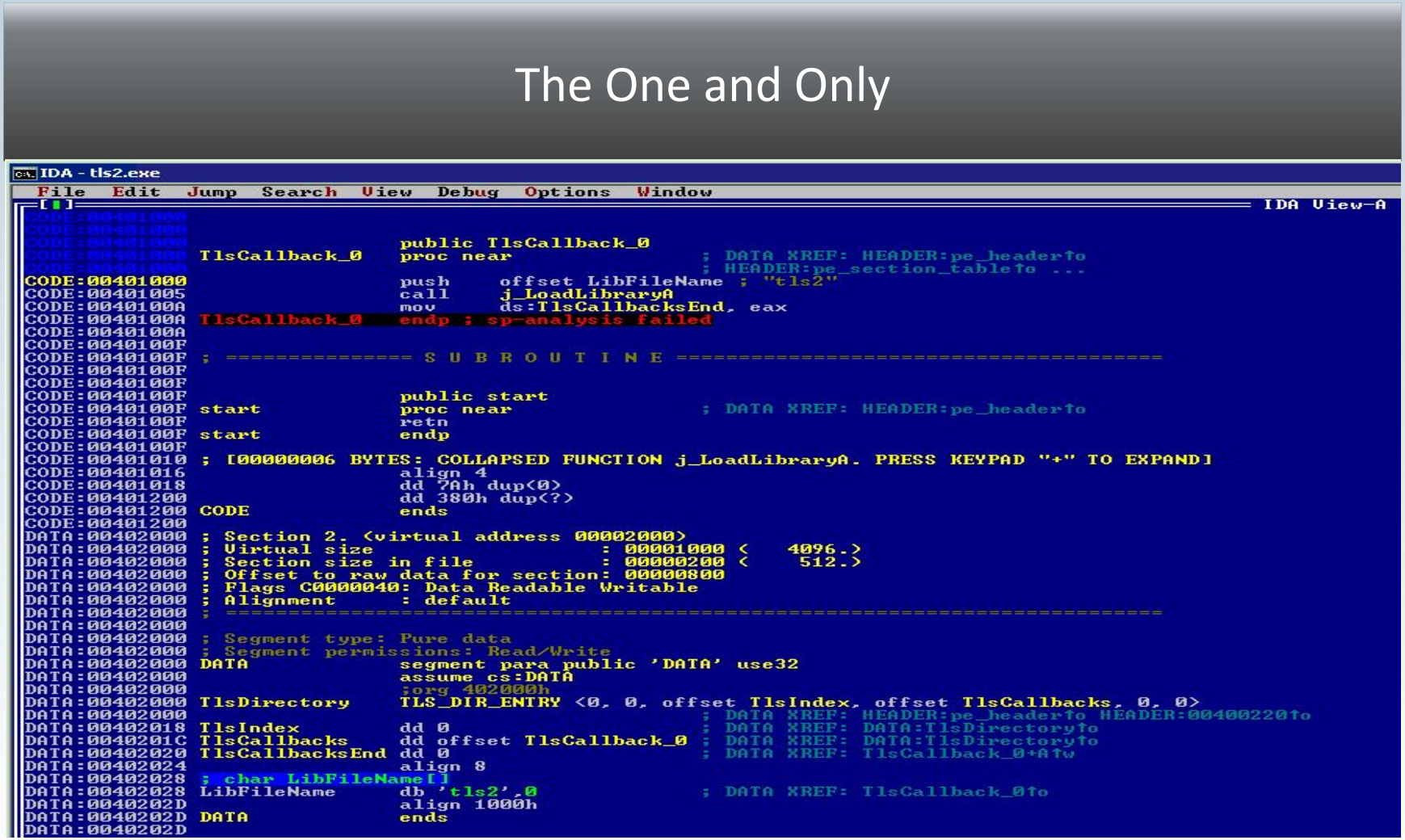

Hex Workshop - [tls2.exe]
File Edit Disk Options Tools Window Help
TLS is present
(size doesn't matter)
Ready
Offset: 00000000
Value: 23117
4096 bytes
OVR MOD READ

Empty!


Hex Workshop - [tls2.exe]
File Edit Disk Options Tools Window Help
Callback pointer
Callback array
tls2
KERNEL32
Ready
Offset: 00000600
Value: 10344
4096 bytes
OVR MOD READ

Empty!

So the search moves to the callbacks,
of which there is only one... or is there?



The One and Only

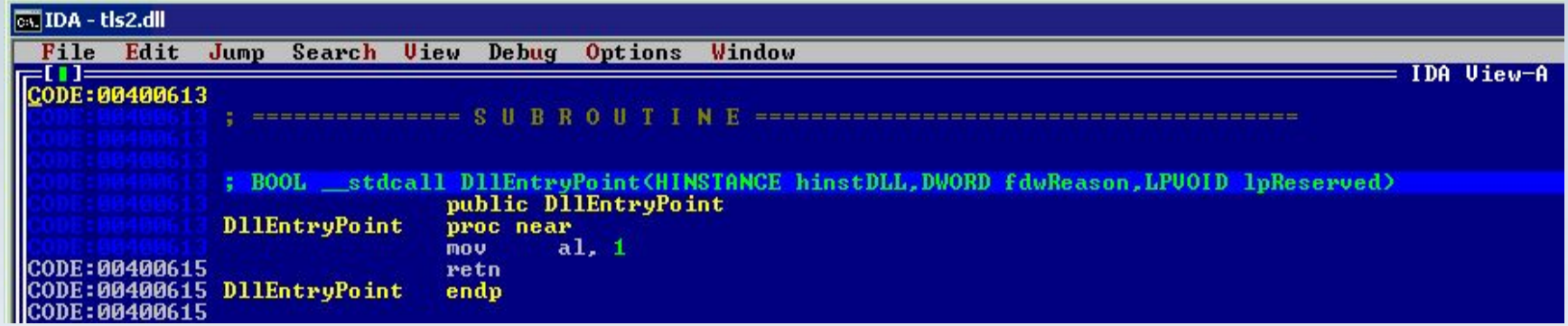
```
IDA - tls2.exe
File  Edit  Jump  Search  View  Debug  Options  Window
[ ]                                                                    IDA View-A
CODE:00401000                   public TlsCallback_0
CODE:00401000 TlsCallback_0     proc near               ; DATA XREF: HEADER:pe_header↑o
CODE:00401000                                           ; HEADER:pe_section_table↑o ...
CODE:00401000                   push    offset LibFileName ; "tls2"
CODE:00401005                   call    j_LoadLibraryA
CODE:0040100A                   mov     ds:TlsCallbacksEnd, eax
CODE:0040100A TlsCallback_0     endp ; sp-analysis failed
CODE:0040100A
CODE:0040100F
CODE:0040100F ; =============== S U B R O U T I N E =======================================
CODE:0040100F
CODE:0040100F
CODE:0040100F                   public start
CODE:0040100F start             proc near               ; DATA XREF: HEADER:pe_header↑o
CODE:0040100F                   retn
CODE:0040100F start             endp
CODE:0040100F
CODE:00401010 ; [00000006 BYTES: COLLAPSED FUNCTION j_LoadLibraryA. PRESS KEYPAD "+" TO EXPAND]
CODE:00401016                   align 4
CODE:00401018                   dd 7Ah dup(0)
CODE:00401200                   dd 380h dup(?)
CODE:00401200 CODE              ends
CODE:00401200
DATA:00402000 ; Section 2. (virtual address 00002000)
DATA:00402000 ; Virtual size                  : 00001000 (    4096.)
DATA:00402000 ; Section size in file          : 00000200 (     512.)
DATA:00402000 ; Offset to raw data for section: 00000800
DATA:00402000 ; Flags C0000040: Data Readable Writable
DATA:00402000 ; Alignment     : default
DATA:00402000 ; ===========================================================================
DATA:00402000
DATA:00402000 ; Segment type: Pure data
DATA:00402000 ; Segment permissions: Read/Write
DATA:00402000 DATA              segment para public 'DATA' use32
DATA:00402000                   assume cs:DATA
DATA:00402000                   ;org 402000h
DATA:00402000 TlsDirectory      TLS_DIR_ENTRY <0, 0, offset TlsIndex, offset TlsCallbacks, 0, 0>
DATA:00402000                                           ; DATA XREF: HEADER:pe_header↑o HEADER:004002201o
DATA:00402018 TlsIndex          dd 0                    ; DATA XREF: DATA:TlsDirectory↑o
DATA:0040201C TlsCallbacks      dd offset TlsCallback_0 ; DATA XREF: DATA:TlsDirectory↑o
DATA:00402020 TlsCallbacksEnd   dd 0                    ; DATA XREF: TlsCallback_0+A↑w
DATA:00402024                   align 8
DATA:00402028 ; char LibFileName[]
DATA:00402028 LibFileName       db 'tls2',0             ; DATA XREF: TlsCallback_0↑o
DATA:0040202D                   align 1000h
DATA:0040202D DATA              ends
DATA:0040202D
```

Am I Missing Somethi

```
CODE:00401010          push    offset LibFileName ; "tls2"
CODE:00401005          call     j_LoadLibraryA
CODE:0040100A          mov     ds:TlsCallbacksEnd, eax
```

Hmm, LoadLibrary("tls2")
Maybe DllMain contains something interesting?

I Am Missing Somethi



No, that's not it.

Take 2

Let's revisit the code:

```
CODE:00401010        push    offset LibFileName ; "tls2"
CODE:00401005        call     j_LoadLibraryA
CODE:0040100A        mov     ds:TlsCallbacksEnd, eax
```

It's All About Image

It's the TlsCallBacks extended array trick again.
Q. What value does the new callback contain?
A. The DLL's imagebase.
Q. DEP won't let that run, right?
A. …

Surprise!

Not OK

Of course it will.

You just have to ask nicely.

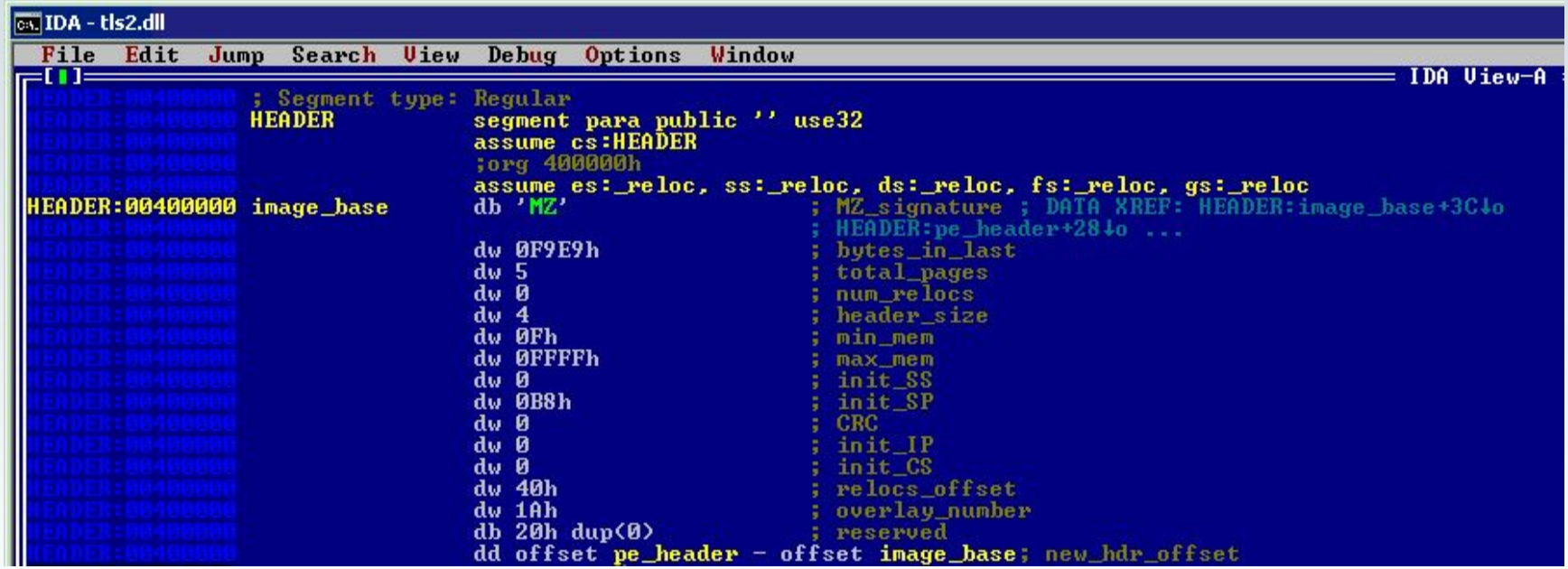
Or take a cue from a driver.

Chaotic-Evil

When the SectionAlignment value is less than 4kb,
the file header is marked Writable and Executable
(unless the last section characteristics override it).
That turns the ImageBase into code.

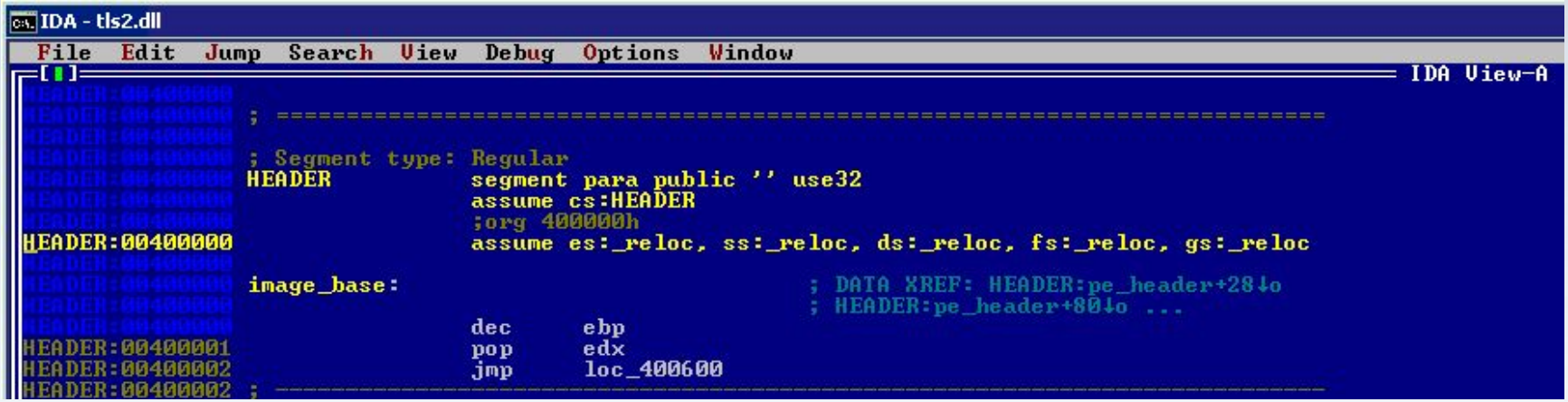
Before

So we go from this…

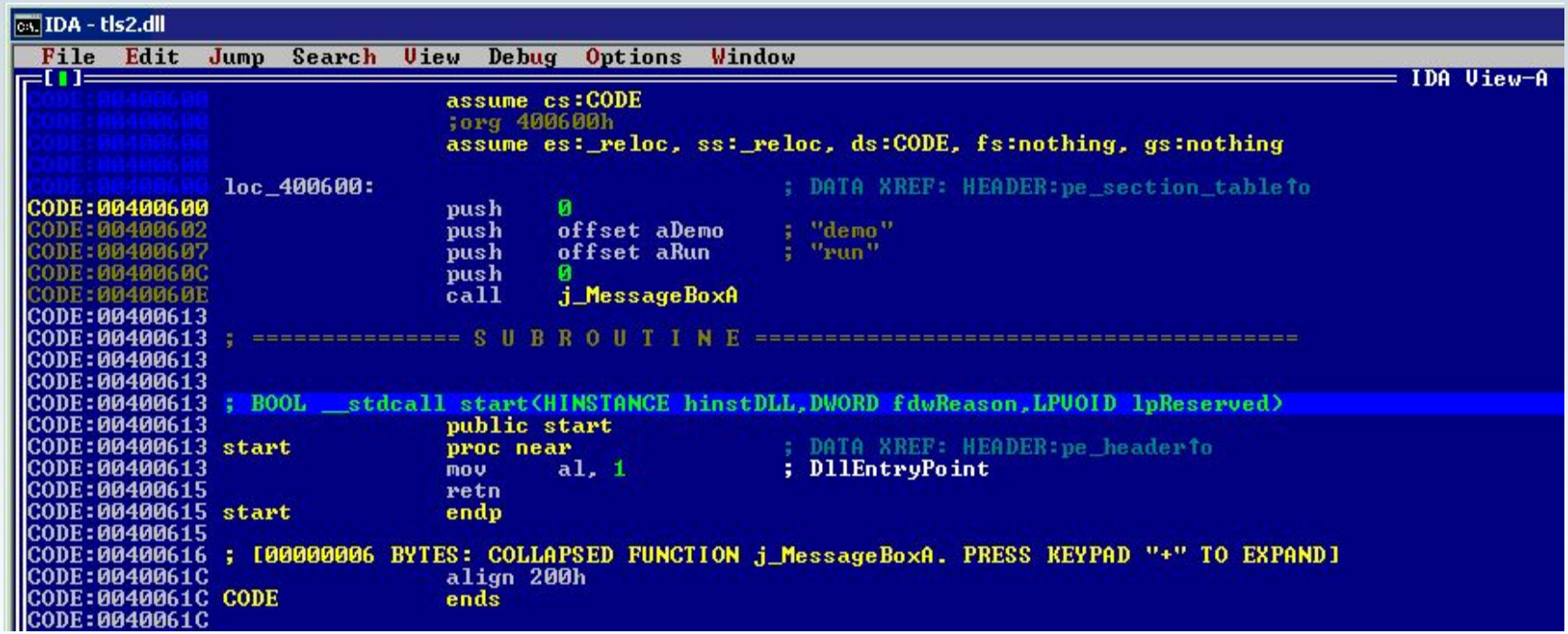


After

To this…



Presto!


IDA - tls2.dll
File Edit Jump Search View Debug Options Window
IDA View-A
assume cs:CODE
;org 400600h
assume es:_reloc, ss:_reloc, ds:CODE, fs:nothing, gs:nothing
loc_400600: ; DATA XREF: HEADER:pe_section_table↑o
CODE:00400600 push 0
CODE:00400602 push offset aDemo ; "demo"
CODE:00400607 push offset aRun ; "run"
CODE:0040060C push 0
CODE:0040060E call j_MessageBoxA
CODE:00400613
CODE:00400613 ; =============== S U B R O U T I N E =======================================
CODE:00400613
CODE:00400613
CODE:00400613 ; BOOL __stdcall start(HINSTANCE hinstDLL,DWORD fdwReason,LPVOID lpReserved)
CODE:00400613 public start
CODE:00400613 start proc near ; DATA XREF: HEADER:pe_header↑o
CODE:00400613 mov al, 1 ; DllEntryPoint
CODE:00400615 retn
CODE:00400615 start endp
CODE:00400615
CODE:00400616 ; [00000006 BYTES: COLLAPSED FUNCTION j_MessageBoxA. PRESS KEYPAD "+" TO EXPAND]
CODE:0040061C align 200h
CODE:0040061C CODE ends
CODE:0040061C

Really Not OK

Just a little something to add to the workload.