

CPSC 458-01 Exercise 3

Wayne Muse

1. Set up

First, we need to download the tools used for this exercise and store them in the tools folder.

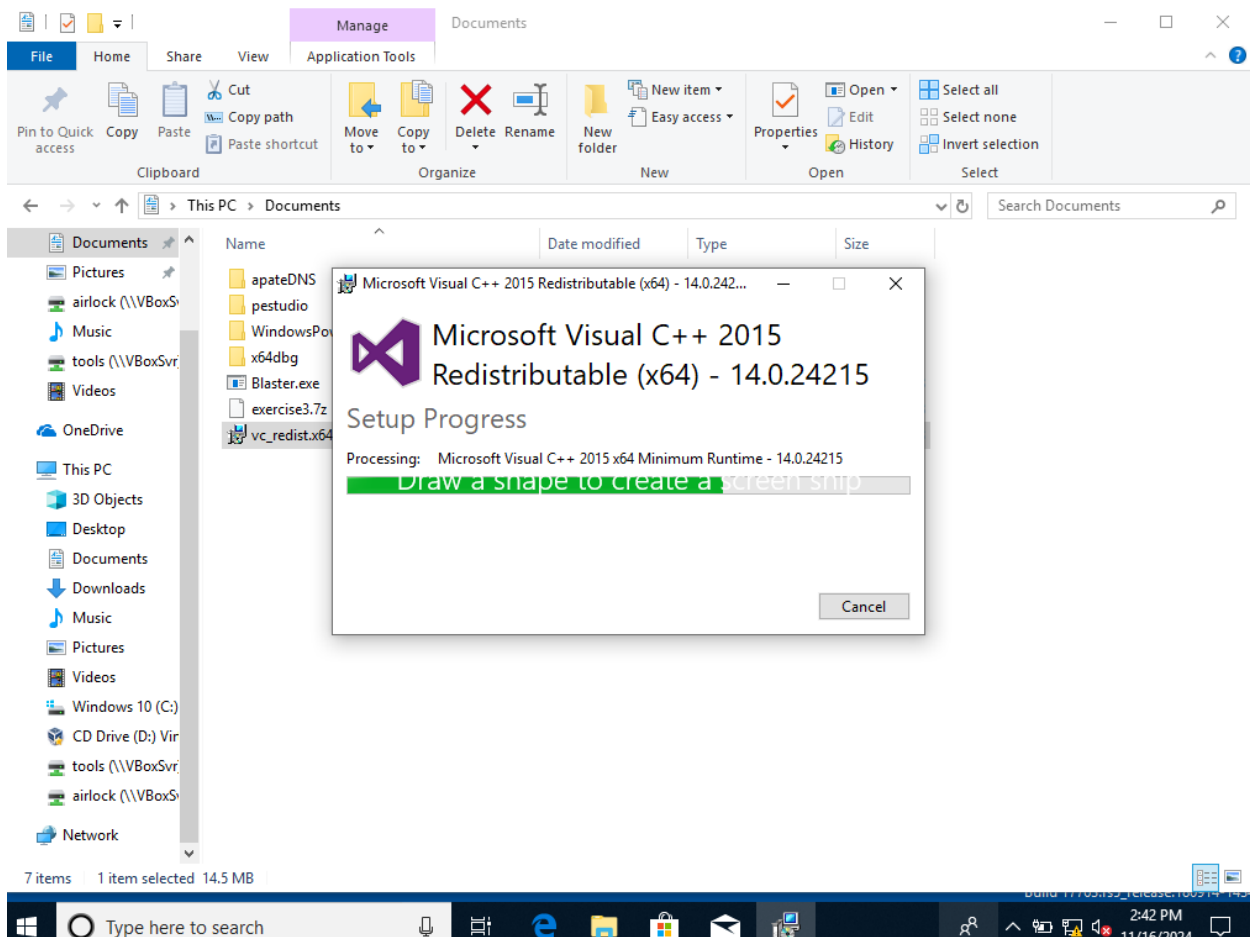
- [7Zip](#)
- [x32dbg](#)
- [Scylla](#)
- [Ghidra](#)
- [Microsoft Visual C++ Redistributable 2015 Update 3](#)
- (not needed but useful) [UPX](#)

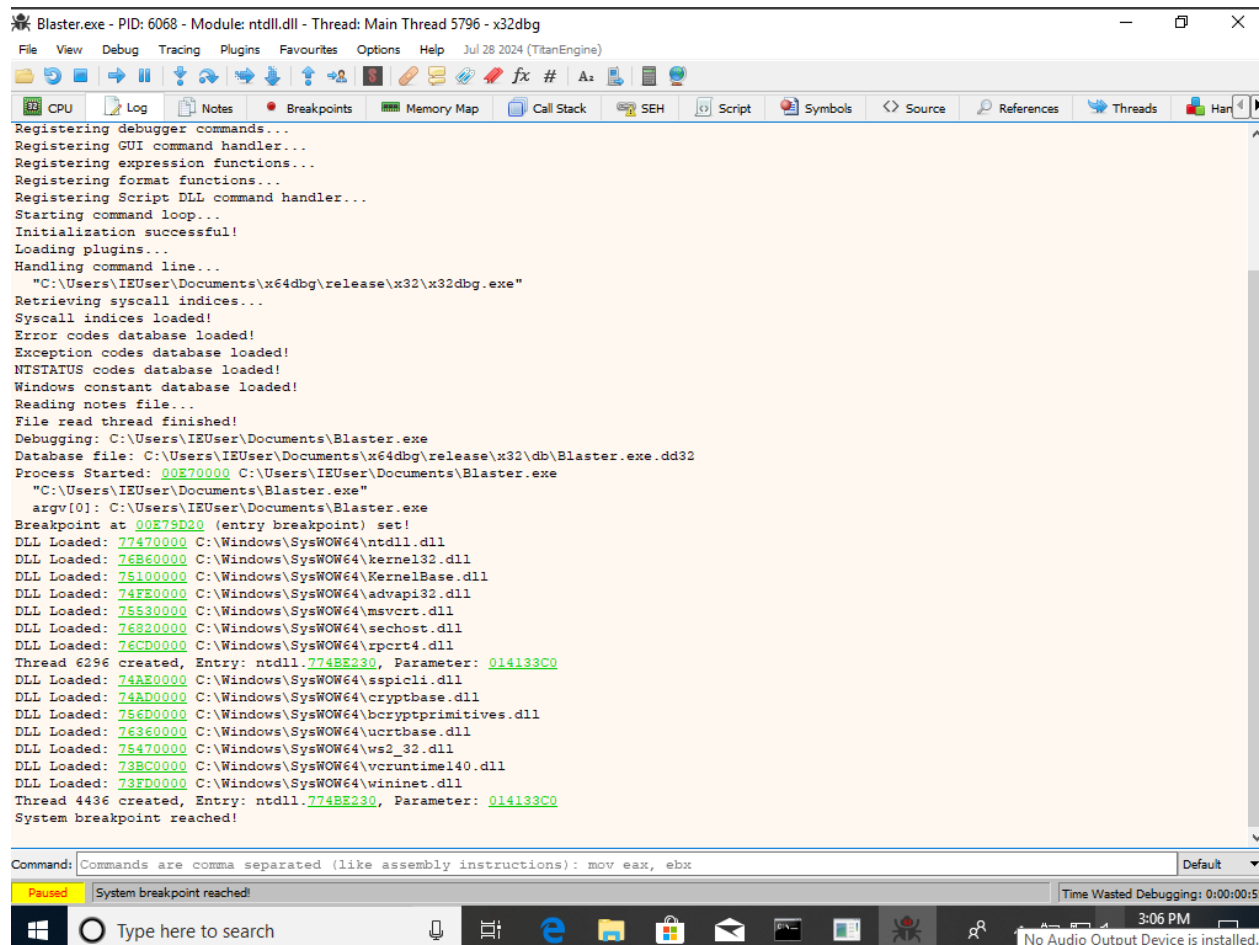
The main objective of this exercise, which is to unpack a packed executable, analyze it in x32dbg, dump the unpacked code, and inspect the result in Ghidra.

2. X64DBG

We booted up x64DBG and ran x32DBG on blaster.exe.

First, we needed the C++ Redistribution because we kept encountering a VCRUNTIME140.dll error. In the screenshot it shows x64, ignore it. I learned slowly and very painfully that it was the wrong version.





Blaster.exe - PID: 6068 - Module: ntdll.dll - Thread: Main Thread 5796 - x32dbg

File View Debug Tracing Plugins Favourites Options Help Jul 28 2024 (TitanEngine)

CPU Log Notes Breakpoints Memory Map Call Stack SEH Script Symbols Source References Threads Har

Type	Address	Module/Label/Exception	State	Disassembly	Hits	Summary
Software	00E79D20	<blaster.exe.OptionalHeader.AddressOfEntr	One-time	pushad	0	entry

Command: Commands are comma separated (like assembly instructions): mov eax, ebx

Default

Paused System breakpoint reached! Time Wasted Debugging: 0:00:00:36

Type here to search

Type	Address	Module/Label/Exception	State	Disassembly	Hits	Summary
Software	00E79D20	<blaster.exe.OptionalHeader.AddressOfEntr	One-time	pushad	0	entry
	00E79D20	<blaster.OptionalHeader.AddressOfEntryPoint>		pushad mov esi,blaster.E78000 lea edi,dword ptr ds:[esi-7000] push edi or ebp,FFFFFFFF jmp blaster.E79D42 nop nop nop nop nop nop mov al,byte ptr ds:[esi] inc esi mov byte ptr ds:[edi],al inc edi add ebx,ebx jne blaster.E79D49 mov ebx,dword ptr ds:[esi] sub esi,FFFFFFFC		

Next we worked through the [How to unpack UPX malware with a single breakpoint](#) by Saket Upadhyay.

The screenshot shows the Immunity Debugger interface with the following components:

- Top Bar:** Blaster.exe - Module: blaster.exe - Thread: Main Thread 5796 - x32dbg
- Menu Bar:** File, View, Debug, Tracing, Plugins, Favourites, Options, Help
- Toolbars:** CPU, Log, Notes, Breakpoints, Memory Map, Call Stack, SEH, Script, Symbols, Source, References, Threads, Hardware
- CPU Window:** Displays assembly instructions for the current thread. The instruction at address 00E79D20 is highlighted, showing a jump to blaster.E79D42.
- Register Window:** Shows the values of registers. EAX is 012FFB78, ECX is 00E79D20, EDX is 00E79D20, EBP is 012FFB2C, ESP is 012FFB20, ESI is 00E79D20, and EDI is 00E79D20.
- Stack Window:** Shows the stack frame. The return address is 012FFB20, and the return value is 00000000.
- Dump Window:** Shows memory dumps. The first dump is at address 00E79D20, and the second is at address 00E79D42.
- Command Window:** Shows the command: Commands are comma separated (like assembly instructions): mov eax, ebx
- Status Bar:** Shows the current state: Paused, INT3 breakpoint "entry breakpoint" at <blaster.OptionalHeader.AddressOfEntryPoint> (00E79D20).

3. Scylla

Blaster.exe - PID: 6068 - Module: blaster.exe - Thread: Main Thread 5796 - x32dbg

FileViewDebugTracingPluginsFavouritesOptionsHelpJul 28 2024 (TitanEngine)

CPULogNotesBreakpointsMemory MapCall StackSEHScriptSymbolsSourceReferencesThreadsHandlesTrace

00E79E8E	86C4	xchg ah,al	
00E79E90	C1C0 10	rol eax,10	
00E79E93	86C4	xchg ah,al	
00E79E95	01F0	add eax,esi	esi:EntryPoint
00E79E97	8903	mov dword ptr ds:[ebx],eax	
00E79E99	E8 E2	jmp blaster.E79E7D	
00E79E9B	24 0F	and al,F	
00E79E9D	C1E0 10	shl eax,10	
00E79EA0	66:8B07	mov ax,word ptr ds:[edi]	edi:EntryPoint
00E79EA3	83C7 02	add edi,2	edi:EntryPoint
00E79EA6	E8 E2	jmp blaster.E79E84	
00E79EA8	8BAE 48930000	mov ebp,dword ptr ds:[esi+9348]	
00E79EAB	80BE 00F0FFFF	lea edi,dword ptr ds:[esi-1000]	edi:EntryPoint
00E79EB4	8B 00100000	mov ebx,1000	
00E79EB9	50	push eax	
00E79EBA	54	push esp	
00E79EBB	6A 04	push 4	
00E79EBD	53	push ebx	
00E79EBE	57	push edi	edi:EntryPoint
00E79EBF	FFD5	call ebp	
00E79EC1	80B7 1F020000	lea eax,dword ptr ds:[edi+21F]	
00E79EC7	8020 7F	and byte ptr ds:[eax],7F	
00E79ECA	8060 28 7F	and byte ptr ds:[eax+28],7F	
00E79ECE	58	pop eax	
00E79ECF	50	push eax	
00E79ED0	54	push esp	
00E79ED1	50	push eax	
00E79ED2	53	push ebx	
00E79ED3	57	push edi	edi:EntryPoint
00E79ED4	FFD5	call ebp	
00E79ED6	58	pop eax	
00E79ED7	61	popad	
00E79ED8	8D4424 80	lea eax,dword ptr ss:[esp-80]	
00E79EDC	6A 00	push 0	
00E79EDE	39C4	cmp esp,eax	
00E79EE0	75 FA	jnb blaster.E79EDC	
00E79EE2	83EC 80	sub esp,FFFFFFF0	
00E79EE5	E9 CC83FFFF	jmp blaster.E72286	
00E79EEA	0000	add byte ptr ds:[eax],al	
00E79EEB	C000 00	rol byte ptr ds:[eax],0	
00E79EEF	0000	add byte ptr ds:[eax],al	
00E79EF1	0000	add byte ptr ds:[eax],al	
00E79EF3	0000	add byte ptr ds:[eax],al	
00E79EF5	0000	add byte ptr ds:[eax],al	
00E79EF7	0000	add byte ptr ds:[eax],al	
00E79EF9	0000	add byte ptr ds:[eax],al	
00E79EFB	0000	add byte ptr ds:[eax],al	
00E79EFD	0000	add byte ptr ds:[eax],al	
00E79EFF	0000	add byte ptr ds:[eax],al	
00E79F01	0000	add byte ptr ds:[eax],al	
00E79F03	0000	add byte ptr ds:[eax],al	
00E79F05	0000	add byte ptr ds:[eax],al	

Blaster.exe - PID: 6068 - Module: blaster.exe - Thread: Main Thread 5796 - x32dbg

FileViewDebugTracingPluginsFavouritesOptionsHelpJul 28 2024 (TitanEngine)

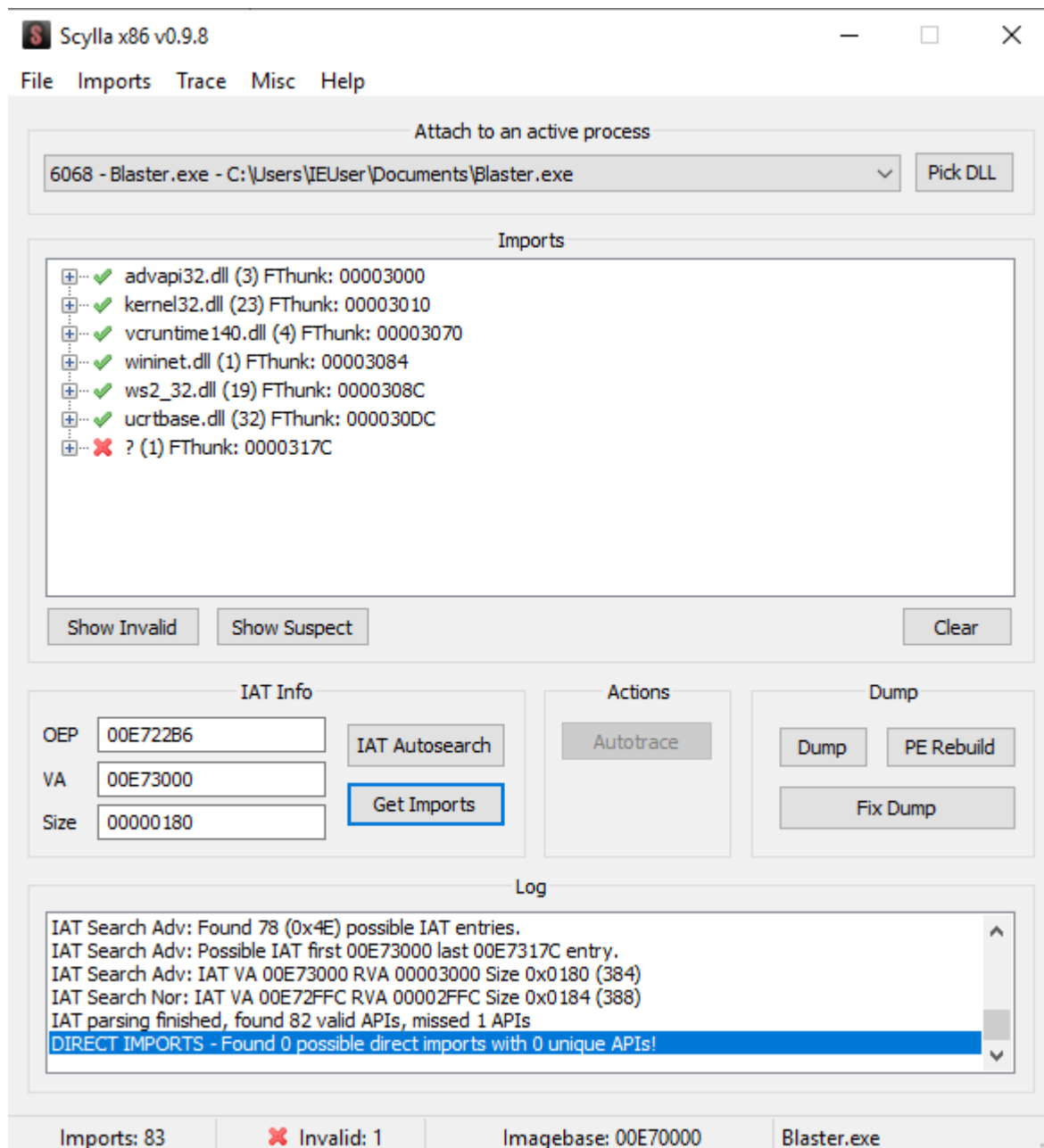
CPULogNotesBreakpointsMemory MapCall StackSEHScriptSymbolsSourceReferencesThreadsHandlesTrace

00E79E8E	86C4	xchg ah,al	
00E79E90	C1C0 10	rol eax,10	
00E79E93	86C4	xchg ah,al	
00E79E95	01F0	add eax,esi	esi:EntryPoint
00E79E97	8903	mov dword ptr ds:[ebx],eax	
00E79E99	E8 E2	jmp blaster.E79E7D	
00E79E9B	24 0F	and al,F	
00E79E9D	C1E0 10	shl eax,10	
00E79EA0	66:8B07	mov ax,word ptr ds:[edi]	edi:EntryPoint
00E79EA3	83C7 02	add edi,2	edi:EntryPoint
00E79EA6	E8 E2	jmp blaster.E79E84	
00E79EA8	8BAE 48930000	mov ebp,dword ptr ds:[esi+9348]	
00E79EAB	80BE 00F0FFFF	lea edi,dword ptr ds:[esi-1000]	edi:EntryPoint
00E79EB4	8B 00100000	mov ebx,1000	
00E79EB9	50	push eax	
00E79EBA	54	push esp	
00E79EBB	6A 04	push 4	
00E79EBD	53	push ebx	
00E79EBE	57	push edi	edi:EntryPoint
00E79EBF	FFD5	call ebp	
00E79EC1	80B7 1F020000	lea eax,dword ptr ds:[edi+21F]	
00E79EC7	8020 7F	and byte ptr ds:[eax],7F	
00E79ECA	8060 28 7F	and byte ptr ds:[eax+28],7F	
00E79ECE	58	pop eax	
00E79ECF	50	push eax	
00E79ED0	54	push esp	
00E79ED1	50	push eax	
00E79ED2	53	push ebx	
00E79ED3	57	push edi	edi:EntryPoint
00E79ED4	FFD5	call ebp	
00E79ED6	58	pop eax	
00E79ED7	61	popad	
00E79ED8	8D4424 80	lea eax,dword ptr ss:[esp-80]	
00E79EDC	6A 00	push 0	
00E79EDE	39C4	cmp esp,eax	
00E79EE0	75 FA	jnb blaster.E79EDC	
00E79EE2	83EC 80	sub esp,FFFFFFF0	
00E79EE5	E9 CC83FFFF	jmp blaster.E72286	
00E79EEA	0000	add byte ptr ds:[eax],al	
00E79EEB	C000 00	rol byte ptr ds:[eax],0	
00E79EEF	0000	add byte ptr ds:[eax],al	
00E79EF1	0000	add byte ptr ds:[eax],al	
00E79EF3	0000	add byte ptr ds:[eax],al	
00E79EF5	0000	add byte ptr ds:[eax],al	
00E79EF7	0000	add byte ptr ds:[eax],al	
00E79EF9	0000	add byte ptr ds:[eax],al	
00E79EFB	0000	add byte ptr ds:[eax],al	
00E79EFD	0000	add byte ptr ds:[eax],al	
00E79EFF	0000	add byte ptr ds:[eax],al	
00E79F01	0000	add byte ptr ds:[eax],al	
00E79F03	0000	add byte ptr ds:[eax],al	
00E79F05	0000	add byte ptr ds:[eax],al	

blaster.00E72286

UPX1:00E79EE5 blaster.exe:\$9EE5 #22E5

00E72285 CC | int3 | || 00E72286 | E8 C5030000 | call blaster.E72640 | |
00E7228B	E9 74FEFFFF	jmp blaster.E72134	
00E722C0	55	push ebp	
00E722C1	8BEC	mov ebp,esp	
00E722C3	6A 00	push 0	



Attach to an active process

6068 - Blaster.exe - C:\Users\IEUser\Documents\Blaster.exe

Pick DLL

Imports

+

 ✓ advapi32.dll (3) FThunk: 00003000

+

 ✓ kernel32.dll (23) FThunk: 00003010

+

 ✓ vcruntime140.dll (4) FThunk: 00003070

+

 ✓ wininet.dll (1) FThunk: 00003084

+

 ✓ ws2_32.dll (19) FThunk: 0000308C

+

 ✓ ucrtbase.dll (32) FThunk: 000030DC

Show Invalid

Show Suspect

Clear

IAT Info

OEP 00E722B6

VA 00E73000

Size 00000180

IAT Autosearch

Get Imports

Autotrace

Dump

Dump

PE Rebuild

Fix Dump

Log

IAT parsing finished, found 82 valid APIs, missed 1 APIs

DIRECT IMPORTS - Found 0 possible direct imports with 0 unique APIs!

Dump success C:\Users\IEUser\Documents\Blaster_dump.exe

Import Rebuild success C:\Users\IEUser\Documents\Blaster_dump_SCY.exe

Dump success C:\Users\IEUser\Documents\Blaster_dump.exe

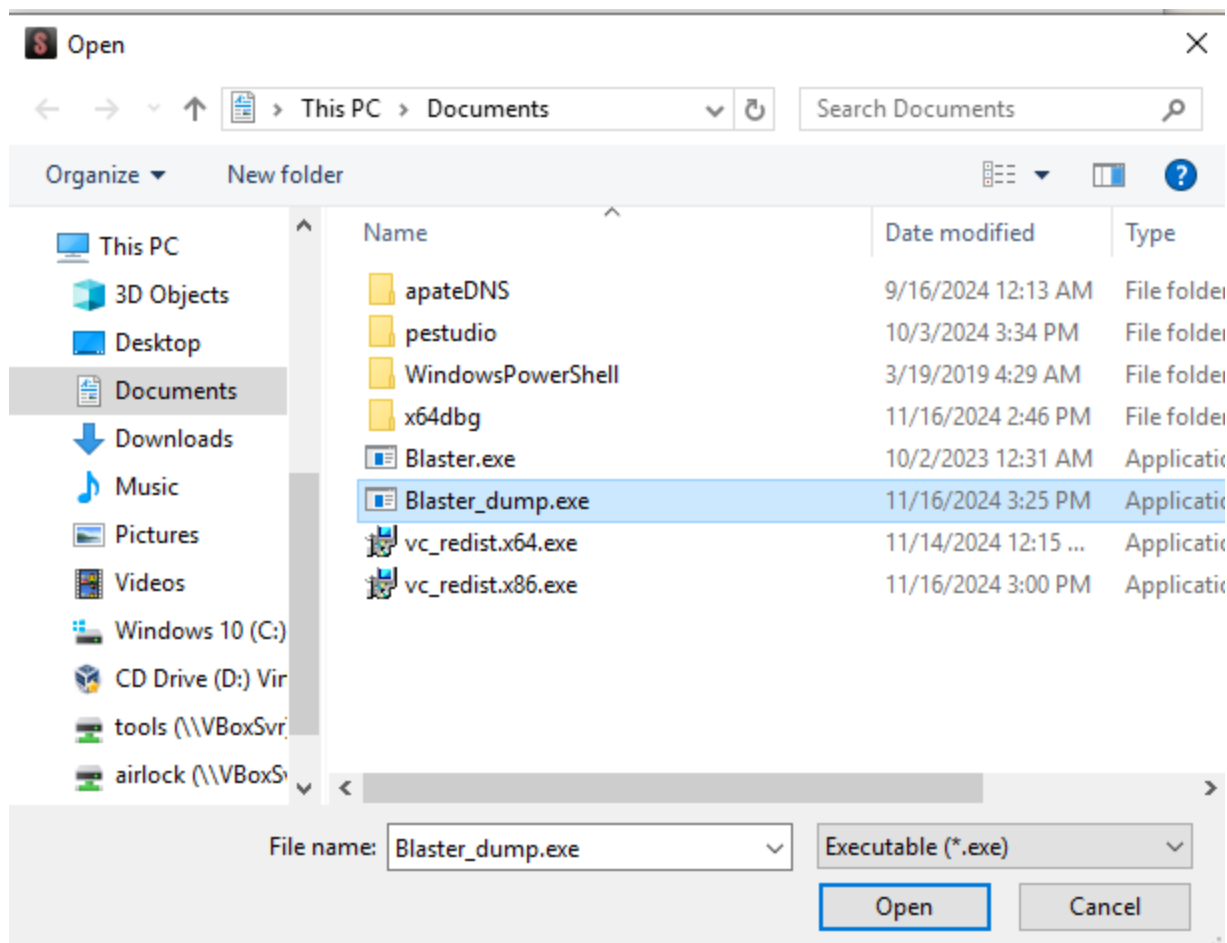
Import Rebuild success C:\Users\IEUser\Documents\Blaster_dump_SCY.exe

Imports: 82

✓ Invalid: 0

Imagebase: 00E70000

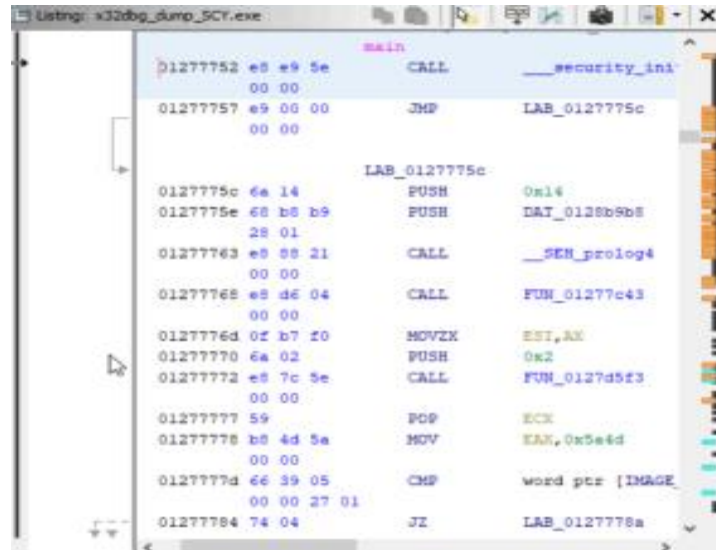
Blaster.exe



Now we have the fixed dump and PE rebuilt of the Blaster.exe file now renamed to Blaster_dump_SCY.exe

```
Dump success C:\Users\IEUser\Documents\Blaster_dump.exe
Import Rebuild success C:\Users\IEUser\Documents\Blaster_dump_SCY.exe
```

4. Ghidra



```
Listing: x32dbg_dump_SCY.exe

main
01277752 e8 e9 5e CALL __security_init
00 00
01277757 e9 00 00 JMP LAB_0127775c
00 00

LAB_0127775c
0127775c 6a 14 PUSH 0x14
0127775e 68 b8 b9 PUSH DAT_0128b9b8
28 01
01277763 e8 88 21 CALL __SEH_prolog4
00 00
01277768 e8 d6 04 CALL FUN_01277c43
00 00
0127776d 0f b7 f0 MOVZX ESI,AX
01277770 6a 02 PUSH 0x2
01277772 e8 7c 5e CALL FUN_0127d5f3
00 00
01277777 59 POP ECK
01277778 b8 4d 5a MOV EAX,0x5a4d
00 00
0127777d e6 39 05 CMP word ptr [IMAGE_00000270],EAX
00 00 27 01
01277784 74 04 JZ LAB_0127778a
```

Then we loaded the new SCY version of the blaster.exe and finding the entry point similar to the 2nd exercise. Then we verified that Ghidra's Listing window shows the same addresses and as x32dbg