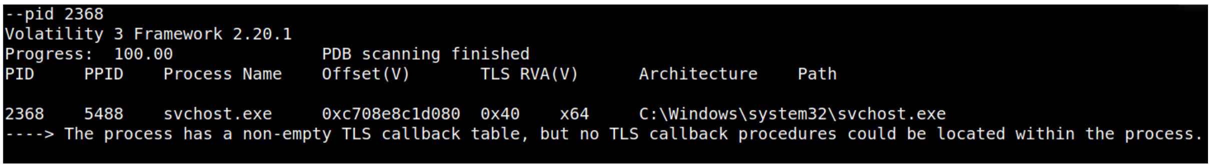
SCREENSHOTS

1. UrSnif Malware Sample

MD5: 13794d1d8e87c69119237256ef068043



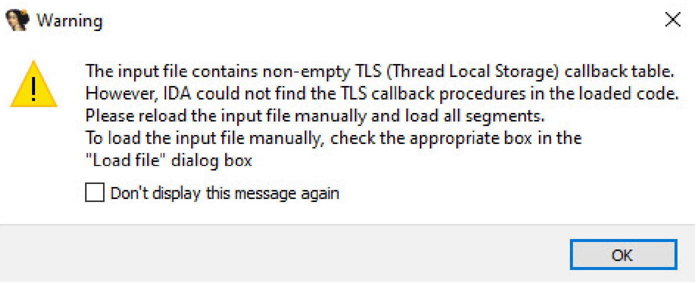
```
--pid 2368
Volatility 3 Framework 2.20.1
Progress:  100.00               PDB scanning finished
PID     PPID    Process Name    Offset(V)       TLS RVA(V)      Architecture    Path

2368    5488    svchost.exe     0xc708e8c1d080  0x40    x64     C:\Windows\system32\svchost.exe
----> The process has a non-empty TLS callback table, but no TLS callback procedures could be located within the process.
```

Fig. 1. TlsCheck output for the UrSnif variant.

This screenshot shows the output of TlsCheck for a 2017 UrSnif sample. The malware hollowed out `svchost.exe` and injected TLS callbacks. When analyzed with our plugin, the detected callbacks produced this output.

For comparison, the same process was examined using IDA Freeware, and its results are shown below.

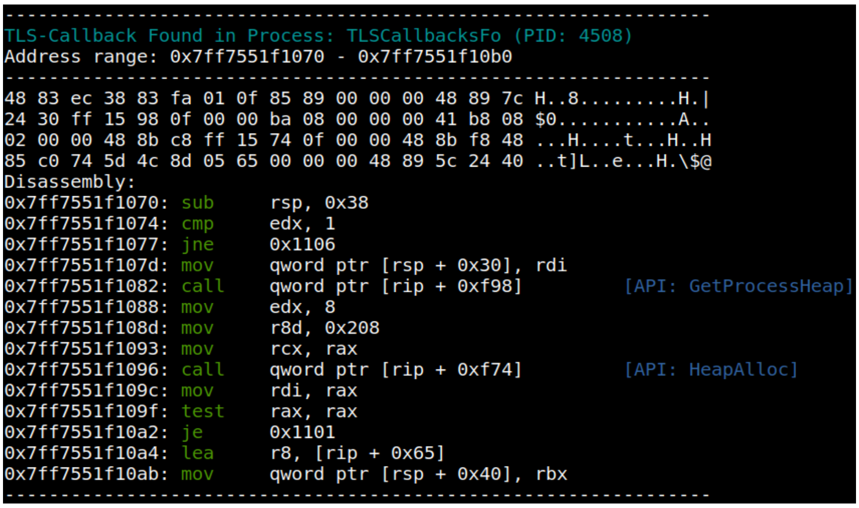


The input file contains non-empty TLS (Thread Local Storage) callback table.
However, IDA could not find the TLS callback procedures in the loaded code.
Please reload the input file manually and load all segments.
To load the input file manually, check the appropriate box in the
"Load file" dialog box

□ Don't display this message again

Fig. 2. IDA Freeware output for the UrSnif variant.

2. API Resolutions in TlsCheck (Under Development)



```
----------------------------------------------------------------
TLS-Callback Found in Process: TLSCallbacksFo (PID: 4508)
Address range: 0x7ff7551f1070 - 0x7ff7551f10b0
----------------------------------------------------------------
48 83 ec 38 83 fa 01 0f 85 89 00 00 00 48 89 7c H..8.........H.|
24 30 ff 15 98 0f 00 00 ba 08 00 00 00 41 b8 08 $0...........A..
02 00 00 48 8b c8 ff 15 74 0f 00 00 48 8b f8 48 ...H....t...H..H
85 c0 74 5d 4c 8d 05 65 00 00 00 48 89 5c 24 40 ..t]L..e...H.\$@
Disassembly:
0x7ff7551f1070: sub     rsp, 0x38
0x7ff7551f1074: cmp     edx, 1
0x7ff7551f1077: jne     0x1106
0x7ff7551f107d: mov     qword ptr [rsp + 0x30], rdi
0x7ff7551f1082: call    qword ptr [rip + 0xf98]         [API: GetProcessHeap]
0x7ff7551f1088: mov     edx, 8
0x7ff7551f108d: mov     r8d, 0x208
0x7ff7551f1093: mov     rcx, rax
0x7ff7551f1096: call    qword ptr [rip + 0xf74]         [API: HeapAlloc]
0x7ff7551f109c: mov     rdi, rax
0x7ff7551f109f: test    rax, rax
0x7ff7551f10a2: je      0x1101
0x7ff7551f10a4: lea     r8, [rip + 0x65]
0x7ff7551f10ab: mov     qword ptr [rsp + 0x40], rbx
----------------------------------------------------------------
```

Fig. 3. Output from TlsCheck representing the resolution of some API calls.