Debugging.TV

Frame 0x11

Presenter: Dmitry Vostokov

Topics

- Stack region (Windows)
- Stack region (Mac OS X)
- Stack region (Windows, 2nd method)
- Patterns

Stack Region (W)

```
0:000> ~
.  0  Id: bdc.8c8 Suspend: 0 Teb: 000007ff`fffdc000 Unfrozen
   1  Id: bdc.aec Suspend: 0 Teb: 000007ff`fffda000 Unfrozen
   2  Id: bdc.674 Suspend: 0 Teb: 000007ff`fffd8000 Unfrozen
   3  Id: bdc.768 Suspend: 0 Teb: 000007ff`fffd6000 Unfrozen
   4  Id: bdc.b34 Suspend: 0 Teb: 000007ff`fffd4000 Unfrozen
   5  Id: bdc.868 Suspend: 0 Teb: 000007ff`fffae000 Unfrozen
   6  Id: bdc.9e4 Suspend: 0 Teb: 000007ff`fffac000 Unfrozen

0:000> !teb 000007ff`fffd6000
TEB at 000007fffffd6000
    ExceptionList:        0000000000000000
    StackBase:            0000000000920000
    StackLimit:           000000000091e000
    SubSystemTib:         0000000000000000
    FiberData:            0000000000001e00
    ArbitraryUserPointer: 0000000000000000
    Self:                 000007fffffd6000
    EnvironmentPointer:   0000000000000000
    ClientId:             0000000000000bdc . 0000000000000768
    RpcHandle:            0000000000000000
    Tls Storage:          000007fffffd6058
    PEB Address:          000007fffffde000
    LastErrorValue:       87
    LastStatusValue:      c000000d
    Count Owned Locks:    0
    HardErrorMode:        0

0:000> dps 000000000091e000 0000000000920000
00000000`0091e000  00000000`00000000
00000000`0091e008  00000000`00000000
[…]
```

Stack Region (M)

```
(gdb) info threads
  3 0x000000010540ce4e in thread_two (arg=0x0)
  2 0x000000010540ce1e in thread_one (arg=0x0)
* 1 0x00007fff885e9e42 in __semwait_signal ()
Current language:  auto; currently minimal
```

```
(gdb) thread 2
[Switching to thread 2 (core thread 1)]
0x000000010540ce1e in thread_one (arg=0x0)
16                  *p = 1;
```

```
(gdb) x $rsp
0x1054c0f10:    0x054c0f50
```

```
(gdb) thread 1
[Switching to thread 1 (core thread 0)]
0x00007fff885e9e42 in __semwait_signal ()
```

```
(gdb) x $rsp
0x7fff6500ba38:  0x8324bdea
```

```
(gdb) maintenance info sections
Core file:
    `/cores/core.925', file type mach-o-le.
    0x0000000105441000->0x00000001054c3000 at 0x00037000: LC_SEGMENT. ALLOC LOAD CODE HAS_CONTENTS
    […]
    0x00007fff6480c000->0x00007fff6500c000 at 0x03a3c000: LC_SEGMENT. ALLOC LOAD CODE HAS_CONTENTS
    […]
```

Stack Region (W2)

```
0:000> ~
.  0  Id: bdc.8c8 Suspend: 0 Teb: 000007ff`fffdc000 Unfrozen
   1  Id: bdc.aec Suspend: 0 Teb: 000007ff`fffda000 Unfrozen
   2  Id: bdc.674 Suspend: 0 Teb: 000007ff`fffd8000 Unfrozen
   3  Id: bdc.768 Suspend: 0 Teb: 000007ff`fffd6000 Unfrozen
   4  Id: bdc.b34 Suspend: 0 Teb: 000007ff`fffd4000 Unfrozen
   5  Id: bdc.868 Suspend: 0 Teb: 000007ff`fffae000 Unfrozen
   6  Id: bdc.9e4 Suspend: 0 Teb: 000007ff`fffac000 Unfrozen

0:000> ~3s
ntdll!NtDelayExecution+0xa:
00000000`7790f9fa c3              ret

0:003> !address rsp
Usage:                  Stack
Allocation Base:        00000000`00820000
Base Address:           00000000`0091e000
End Address:            00000000`00920000
Region Size:            00000000`00002000
Type:                   00020000          MEM_PRIVATE
State:                  00001000          MEM_COMMIT
Protect:                00000004          PAGE_READWRITE
More info:              ~3k

0:000> dps 0000000000091e000 0000000000920000
00000000`0091e000  00000000`00000000
00000000`0091e008  00000000`00000000
[…]
```

Patterns

Structural memory patterns:

- *Memory Region*
- *Region Boundary*

Unified diagnostics/debugging pattern language (software post-construction):

- **Analysis Patterns**

  Execution Residue

- **Architectural Patterns**

  Command Pipe

- **Design Patterns**

  Memory Region

- **Implementation Patterns**

  Memory address attributes

- **Usage Patterns**

  Memory value inspection

!Ad Hardcore Technical Support Training

July 18-23, 2012	Accelerated Windows Memory Dump Analysis
July 27-30, 2012	Accelerated Mac OS X Core Dump Analysis
Sept 3, 2012	Systemic Software Diagnostics (FREE Webinar)
Sept 7-10, 2012	Accelerated .NET Memory Dump Analysis (with x64)
Sept 14-17, 2012	Advanced Windows Memory Dump Analysis
October 12-15, 2012	Accelerated Windows Software Trace Analysis
October 17-22, 2012	Accelerated Windows Malware Analysis

Debugging.TV