Debugging.TV

Frame 0x0C

Presenter: Dmitry Vostokov

Topics

- Multithreading
- Multiple exceptions
- Diagnostic Console reports
- GDB commands for thread navigation

Multiple Exceptions

```
void * thread_one (void *arg)
{
    int *p = NULL;
    *p = 1;

    return 0;
}

void * thread_two (void *arg)
{
    int *p = NULL;
    *p = 2;

    return 0;
}

int main(int argc, const char * argv[])
{
    pthread_t threadID_one, threadID_two;

    pthread_create (&threadID_one, NULL, thread_one, NULL);
    pthread_create (&threadID_two, NULL, thread_two, NULL);

    sleep(3);
    return 0;
}
```

Crash Report

Crashed Thread: 2

Exception Type: EXC_BAD_ACCESS (SIGSEGV)
Exception Codes: KERN_INVALID_ADDRESS at 0x0000000000000000

```
Thread 0:: Dispatch queue: com.apple.main-thread
0   libsystem_kernel.dylib                  0x00007fff854e0e42 __semwait_signal + 10
1   libsystem_c.dylib                       0x00007fff8ababdea nanosleep + 164
2   libsystem_c.dylib                       0x00007fff8ababc2c sleep + 61
3   MultipleThreads                         0x000000010f523ec3 main + 99 (main.c:36)
4   MultipleThreads                         0x000000010f523df4 start + 52

Thread 1:
0   MultipleThreads                         0x000000010f523e1e thread_one + 30 (main.c:16)
1   libsystem_c.dylib                       0x00007fff8abf58bf _pthread_start + 335
2   libsystem_c.dylib                       0x00007fff8abf8b75 thread_start + 13

Thread 2 Crashed:
0   MultipleThreads                         0x000000010f523e4e thread_two + 30 (main.c:24)
1   libsystem_c.dylib                       0x00007fff8abf58bf _pthread_start + 335
2   libsystem_c.dylib                       0x00007fff8abf8b75 thread_start + 13

Thread 2 crashed with X86 Thread State (64-bit):
  rax: 0x0000000000000000  rbx: 0x0000000000000000  rcx: 0x00007fff854e10c2  rdx: 0x0000000000000000
  rdi: 0x0000000000000000  rsi: 0x0000000000000000  rbp: 0x000000010f780f10  rsp: 0x000000010f780f10
   r8: 0x00007fff754c3fb8   r9: 0x0000000000000001  r10: 0x00007fff8abf8b94  r11: 0x0000000000000202
  r12: 0x0000000000001303  r13: 0x000000010f781000  r14: 0x0000000000000000  r15: 0x000000010f523e30
  rip: 0x000000010f523e4e  rfl: 0x0000000000010217  cr2: 0x0000000000000000
```

GDB Output

```
(gdb) info threads
  3 0x00000001062ffe4e in thread_two (arg=0x0)
    at /MultipleThreads/main.c:24
  2 0x00000001062ffe1e in thread_one (arg=0x0)
    at /MultipleThreads/main.c:16
* 1 0x00007fff854e0e42 in __semwait_signal ()

(gdb) thread 2
[Switching to thread 2 (core thread 1)]
0x00000001062ffe1e in thread_one (arg=0x0)
    at /MultipleThreads/main.c:16
16  *p = 1;

(gdb) disassemble 0x00000001062ffe1e
Dump of assembler code for function thread_one:
0x00000001062ffe00 <thread_one+0>:      push   %rbp
0x00000001062ffe01 <thread_one+1>:      mov    %rsp,%rbp
0x00000001062ffe04 <thread_one+4>:      mov    $0x0,%rax
0x00000001062ffe0e <thread_one+14>:     mov    %rdi,-0x8(%rbp)
0x00000001062ffe12 <thread_one+18>:     movq   $0x0,-0x10(%rbp)
0x00000001062ffe1a <thread_one+26>:     mov    -0x10(%rbp),%rdi
0x00000001062ffe1e <thread_one+30>:     movl   $0x1,(%rdi)
0x00000001062ffe24 <thread_one+36>:     pop    %rbp
0x00000001062ffe25 <thread_one+37>:     retq
End of assembler dump.

(gdb) info registers rdi
rdi            0x0    0
```

!Ad Hardcore Technical Support Training

April 11-16, 2012	Accelerated Windows Memory Dump Analysis
April 20-23, 2012	Advanced Windows Memory Dump Analysis
May 11-14, 2012	Accelerated .NET Memory Dump Analysis
June 22, 2012	Introduction to Pattern-Driven Software Diagnostics (Free Webinar)
July 20-23, 2012	Accelerated Windows Software Trace Analysis
July 27-30, 2012	Accelerated Mac OS X Core Dump Analysis

Training Schedule

Debugging.TV