
Subject: Re: 2019.2 (pre)released
Posted by [Tom1](#) on Tue, 22 Oct 2019 09:03:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

Mirek,

Here's one backtrace for 2019.2. They are different for each crash.

Thread ID: 0xfa0

```
Upp::Dbf_Unlink<Upp::BlkHeader_<256> >(x=1eed9c1f840->{ prev=ff182412ff0f1b0c,
next=ff15210dff182412, prev_size=5387, size=65291, free=1, last=1, heap=ff0f1b0cff091408 })
Upp::BlkHeader_<256>::UnlinkFree()
Upp::BlkHeap<Upp::Heap::LargeHeapDetail,256>::Free(h=1eed9c20340->{ prev=6cc,
next=1ee90d31050, prev_size=11, size=8, free=0, last=0, heap=1eed4fed460 })
Upp::Heap::LFree(ptr=1eed9c20350)
Upp::Heap::Free(ptr=1eed9c20350)
Upp::MemoryFree_(ptr=1eed9c20350)
Upp::MemoryFree(ptr=1eed9c20370)
Upp::Vector<Upp::Ctrl::MoveCtrl>::RawFree(ptr=1eed9c20370->{ ctrl={ prec=1eed6a40090 },
from={ left=2297, top=1995, right=2325, bottom=2035 }, to={ left=3812, top=1995, right=3840,
bottom=2035 } })
Upp::Vector<Upp::Ctrl::MoveCtrl>::Free()
Upp::Vector<Upp::Ctrl::MoveCtrl>::Clear()
Upp::AMap<Upp::Ctrl *,Upp::Ctrl::MoveCtrl,Upp::Vector<Upp::Ctrl::MoveCtrl> >::Clear()
Upp::Ctrl::SyncMoves()
Upp::Ctrl::WndProc(hWnd=3b04fe->{ unused=?? }, message=3, wParam=0, lParam=2949120)
CallWindowProcW()
SendMessageW()
GetWindowTextW()
MapWindowPoints()
IsCompositionActive()
IsCompositionActive()
GetWindowTextW()
Upp::Ctrl::WindowProc(message=71 'G', wParam=0, lParam=592725532528)
Upp::TopWindow::WindowProc(message=71 'G', wParam=0, lParam=592725532528)
Upp::Ctrl::WndProc(hWnd=3b04fe->{ unused=?? }, message=71 'G', wParam=0,
lParam=592725532528)
CallWindowProcW()
DispatchMessageW()
MBToWCSEx()
KiUserCallbackDispatcher()
NtUserMessageCall()
GetWindowTextW()
MapWindowPoints()
IsCompositionActive()
```

```

Ordinal96()
IsCompositionActive()
IsCompositionActive()
GetWindowTextW()
Upp::Ctrl::WindowProc(message=274, wParam=61488, lParam=14944857)
Upp::TopWindow::WindowProc(message=274, wParam=61488, lParam=14944857)
Upp::Ctrl::WndProc(hWnd=3b04fe->{ unused=?? }, message=274, wParam=61488,
lParam=14944857)
CallWindowProcW()
DispatchMessageW()
IsWindowVisible()
KiUserCallbackDispatcher()
NtUserMessageCall()
GetWindowTextW()
MapWindowPoints()
IsCompositionActive()
Ordinal96()
IsCompositionActive()
IsCompositionActive()
GetWindowTextW()
Upp::Ctrl::WindowProc(message=161, wParam=9, lParam=14944857)
Upp::TopWindow::WindowProc(message=161, wParam=9, lParam=14944857)
Upp::Ctrl::WndProc(hWnd=3b04fe->{ unused=?? }, message=161, wParam=9,
lParam=14944857)
CallWindowProcW()
DispatchMessageW()
Upp::Ctrl::sProcessMSG(msg=8a0131fc00->{ hwnd=3b04fe, message=161, wParam=9,
lParam=14944857, time=613160312, pt={ x=2649, y=228 } })
Upp::Ctrl::ProcessEvent(quit=8a0131fc90->0)
Upp::Ctrl::ProcessEvents(quit=8a0131fc90->0)
Upp::Ctrl::EventLoop(ctrl=0)
GuiMainFn_()
Upp::AppExecute__(app=7ff6b05416e7)
WinMain(hInstance=7ff6b0510000->{ unused=9460301 }, __formal=0, lpCmdLine=1eed4ff2211
"", nCmdShow=10)
invoke_main()
__srt_common_main_seh()
__srt_common_main()
WinMainCRTStartup()
BaseThreadInitThunk()
RtlUserThreadStart()

```

Thread ID: 0x1db4

```

ZwWaitForWorkViaWorkerFactory()
RtlInitializeResource()
BaseThreadInitThunk()

```

RtlUserThreadStart()

Thread ID: 0xad8

ZwWaitForWorkViaWorkerFactory()
RtlInitializeResource()
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0xe44

ZwDelayExecution()
SleepEx()
LicenseEnforcement::t1cb()
<lambda_2eac692c55e4d4ef341436c9b465fd37>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_2eac692c55e4d4ef341436c9b465fd37>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6a53660)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6a53660)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed5088ff0)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0xec0

ZwDelayExecution()
SleepEx()
LicenseEnforcement::t2cb()
<lambda_2eac692c55e4d4ef341436c9b465fd37>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_2eac692c55e4d4ef341436c9b465fd37>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6a53720)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6a53720)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed5089190)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0x146c

ZwWaitForWorkViaWorkerFactory()
RtlInitializeResource()
BaseThreadInitThunk()

RtlUserThreadStart()

Thread ID: 0x27d0

ZwWaitForWorkViaWorkerFactory()
RtlInitializeResource()
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0x220

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820, LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffffff, SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=0)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac3460)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3460)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed8920a90)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0x300

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820, LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffffff, SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=2)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac32e0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac32e0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed8924ab0)
BaseThreadInitThunk()

RtlUserThreadStart()

Thread ID: 0x28c4

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=1)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac33a0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac33a0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed8924f40)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0x2a44

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=3)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac34a0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac34a0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed891cfb0)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0xc54

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()

```
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=5)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac3760)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3760)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed891d110)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x1abc

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=4)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac36a0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac36a0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed891bb30)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x1fbc

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=6)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
```

```
Upp::sThreadRoutine(arg=1eed6ac3820)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3820)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed8980bc0)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x21c

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=7)
<lambda_ ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void (__cdecl)(void)>::Wrapper<<lambda_ ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void (__cdecl)(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac38e0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac38e0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed898fe60)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x182c

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=8)
<lambda_ ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void (__cdecl)(void)>::Wrapper<<lambda_ ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void (__cdecl)(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac39a0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac39a0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed899eb70)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x1a68

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=9)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac3a60)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3a60)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed899f460)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x1974

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=10)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac3b20)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3b20)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed89ada40)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x1f68

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=fffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=11)
```

```
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac3be0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3be0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed89bbe10)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x23cc

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=12)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac3ca0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3ca0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed9c6eb60)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x2b4c

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=13)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator>()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator>()
Upp::sThreadRoutine(arg=1eed6ac3d60)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3d60)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed89bc760)
BaseThreadInitThunk()
```

RtlUserThreadStart()

Thread ID: 0x7f8

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820, LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffffff, SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=14)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac3e20)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3e20)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed89bd680)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0x1e88

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820, LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffffff, SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=15)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac3ee0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3ee0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed9cbab60)
BaseThreadInitThunk()
RtlUserThreadStart()

Thread ID: 0x1e78

ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()

```
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=16)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6ac3fa0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6ac3fa0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed89bff30)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x54c

```
ZwWaitForAlertByThreadId()
RtlSleepConditionVariableCS()
SleepConditionVariableCS()
Upp::ConditionVariable::Wait(m=7ff6b21420c8->{ section={ DebugInfo=1eed50d7820,
LockCount=-1, RecursionCount=0, OwningThread=0, LockSemaphore=ffffffffffff,
SpinCount=33556432 }, mti=0 })
Upp::CoWork::Pool::ThreadRun(tno=17)
<lambda_ae0911e3bb62843b7fb2857684178ebc>::operator()()
Upp::Function<void __cdecl(void)>::Wrapper<<lambda_ae0911e3bb62843b7fb2857684178ebc>
>::Execute()
Upp::Function<void __cdecl(void)>::operator()()
Upp::sThreadRoutine(arg=1eed6afb0a0)
invoke_thread_procedure(procedure=7ff6b0a69670, context=1eed6afb0a0)
thread_start<unsigned int (__cdecl*)(void * __ptr64)>(parameter=1eed9cd8030)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Thread ID: 0x87c

```
NtUserGetMessage()
GetMessageA()
Upp::Ctrl::Win32OverwatchThread(__formal=0)
BaseThreadInitThunk()
RtlUserThreadStart()
```

Next I will go for the earlier version...

Best regards,

Tom
