RegNotifyChangeKeyValue sucks less

devblogs.microsoft.com/oldnewthing/20150226-00

February 26, 2015

Raymond Chen

One of the gotchas of the `RegNotifyChangeKeyValue` function is that the notification registration has thread affinity. This is a problem if you want the notification registration to outlive the thread that generated it. In particular, if you register the notification from a nonpersistent thread pool thread, you get into an infinite loop:

1. Thread pool task calls `RegNotifyChangeKeyValue`, and waits for the associated event via `RegisterWaitForSingleObject`.
2. Thread pool thread goes idle.
3. Thread pool destroys the idle thread.
4. Due to thread affinity, this signals the handle.
5. The thread pool queues a task to process the handle that was signaled.
6. The task checks the registry key (observes that nothing changed) and calls `RegNotifyChangeKeyValue` again.
7. Repeat.

Windows 8 added a new flag to the `RegNotifyChangeKeyValue` function: `REG_NOTIFY_THREAD_AGNOSTIC`. If you pass this flag, then the notification registration does not have thread affinity. If the thread that called `RegNotifyChangeKeyValue` exits, the notification registration remains active.

Raymond Chen

Follow