

What does it mean when `GetQueuedCompletionStatus` return `ERROR_SEM_TIMEOUT`?

devblogs.microsoft.com/oldnewthing/20140717-00

July 17, 2014



Raymond Chen

A customer asked for assistance interpreting a failure of the `GetQueuedCompletionStatus` function.

We are observing that `GetQueuedCompletionStatus` is intermittently behaving as follows:

- The handle is a `SOCKET` .
- The function returns `FALSE` .
- `lpOverlapped != NULL` .
- `GetLastError` reports `ERROR_SEM_TIMEOUT` : “The semaphore timeout period has expired.”

That’s all the information we have in our log files. We don’t know the value of `numberOfBytes` or `completionKey` , sorry.

We realize that this is a rather vague question, but when this problem hits our machines, it causes our internal logic to go into a reset state since it doesn’t know what the error means or how to recover. Resetting is expensive, and we would prefer to handle this error in a less drastic manner, if only we knew what it meant.

The error code `ERROR_SEM_TIMEOUT` is a rather bad translation of the underlying status code `STATUS_IO_TIMEOUT` , which is much more meaningful. It means that the I/O operation timed out. Colleagues of mine from the networking team chimed in with additional information: A common source of this error with TCP sockets is that the maximum retransmission count and timeout have been reached on a bad (or broken) link.

If you know that the handle is a socket, then you can use `WSAGetOverlappedResult` on the `lpOverlapped` that got returned. Winsock will convert the status code to something more Winsocky. In this case, it would have given you `WSAETIMEDOUT` , which makes it clearer what happened.

[Raymond Chen](#)

Follow

