

# Under what circumstances will `GetProcessTimes` report that a process exited before it was created?

 [devblogs.microsoft.com/oldnewthing/20161110-00](http://devblogs.microsoft.com/oldnewthing/20161110-00)

November 10, 2016



Raymond Chen

A customer reported that their automation started reporting strange values:

```
HANDLE process = ...;
FILETIME creation, exit, kernel, user;
if (GetProcessTimes(process, &creation,
                   &exit, &kernel, &user))
{
    // use the values of creation, exit, kernel, and user
}
```

Their test automation reported that the process had an exit time *earlier* than its creation time. How is this even possible? This apparent violation of causality was causing their automation to stop working.

If you take a closer look at [the documentation for `GetProcessTimes`](#), it says for the `lpExitTime` parameter:

┆ If the process has not exited, the content of this structure is undefined.

What probably is happening is that the process being monitored has not yet exited, so the exit time is undefined. The undefined value might be less than the creation time. It might be greater than the creation time. Heck, if you're really (un)lucky, it might even be *equal* to the creation time.

My guess is that the “undefined” result is coming from uninitialized stack garbage, and the nature of uninitialized stack garbage is that while it is unpredictable, it can also often be consistent over short stretches.

[Raymond Chen](#)

**Follow**

