

# Why does killing Winlogon take down the entire system?

 [devblogs.microsoft.com/oldnewthing/20081013-00](http://devblogs.microsoft.com/oldnewthing/20081013-00)

October 13, 2008



Raymond Chen

Commenter Demeli asks, “Why does Winlogon take down the entire system when you attach a debugger to it? (drwtsn32 -p <pid of Winlogon>)”

This question already has a mistaken in it. Running drwtsn32 on a process isn’t attaching a debugger to it. Attaching a debugger would be something like `ntsd -p <pid of Winlogon>` , and this does work, assuming you have the necessary privileges, of course. (Indeed, this is how the Windows team debugs problems with Winlogon.) In other words, the literal answer to the question is “No, Winlogon does not take down the entire system when you attach a debugger to it.”

What `drwtsn32` does is take a crash dump of the process and then kills the target process. And it is killing Winlogon that crashes the system.

Winlogon is what is known as a “critical system process”, the death of which forces a system restart. And you can probably guess why the system considers Winlogon critical to its functioning. For example, Winlogon is responsible for handling the secure attention sequence, also known as Ctrl+Alt+Del. If Winlogon were to die, then the secure attention sequence would stop working. That’s kind of bad.

[Raymond Chen](#)

**Follow**

