

3 August 2007

By: Marius Oiaga, Technology News Editor

Windows Vista
Microsoft

[Microsoft Is Defending x64 Windows Vista](#)

Kernel Mode Code Signing

The 64-bit editions of Windows Vista have come under assault recently (read about it [here](#) and [here](#)), especially when it comes down to the kernel mode code signing security mitigation introduced in the x64 versions of the operating system. Although Microsoft presented mandatory driver signing in 64-bit Vista as a breath of fresh air against rootkits, and a solution to prevent unsigned code from being loaded into the operating system's core, [Scott Field](#), Windows Security Architect, revealed that the feature is incapable of ensuring the kernel's protection. Kernel Mode Code Signing is not too much of a security barrier. And there are two main scenarios for its bypassing. Malformed code can be injected into the x64 Vista kernel via a driver with a legitimate or a malicious certificate. Additionally, the operating system's core can be breached through faulty drivers. "The Kernel Mode Code Signing is a not a security boundary, rather, it is only one aspect of a defense-in-depth approach to security. KMCS does not provide a means to determine the "intent" of the signed code (i.e., good or bad); indeed, signed code may contain bugs, be of poor quality, or may be malicious in nature," stated Field. "A primary benefit of KMCS is that it provides a means to identify the author of a piece of code, which helps enable follow-up with the author to address crashes that are observed through mechanisms such as Microsoft Online Crash Analysis. Identifying the source and ownership of code that is loaded by the kernel is a fundamental component of the operating system and overall ecosystem trust model. Furthermore, this also provides better transparency to the end user in terms of origin of code that is installed and running on a system." "Still, Microsoft is prepared to deal with any problems that will be associated with the unsigned code being loaded into the kernel via either faulty, legitimate or malicious drivers. This is why the Redmond company has built a flexible list of driver signing certificates, always opened to revocation. "Currently, the kernel mode revocation list is loaded into memory, from disk, once per system boot. The kernel revocation list is checked when the operating system kernel code loads a kernel driver/module. There are several reasons for keeping the logic for this simple in kernel space - for example, constraints in the kernel runtime environment, as well as limiting the attack surface associated with the kernel loader. There are also other practical factors to consider, such as kernel drivers that have already been loaded cannot always be unloaded or removed safely on a running system," Field added.