

So, you want to write a kernel module. You know C, you've written a few normal programs to run as

processes, and now you want to get to where the real action is, to where a single wild pointer can wipe out

your file system and a core dump means a reboot.

What exactly is a kernel module? Modules are pieces of code that can be loaded and unloaded into the kernel

upon demand. They extend the functionality of the kernel without the need to reboot the system.

For example,

one type of module is the device driver, which allows the kernel to access hardware connected to the system.

Without modules, we would have to build monolithic kernels and add new functionality directly into the

kernel image. Besides having larger kernels, this has the disadvantage of requiring us to rebuild and reboot the

kernel every time we want new functionality.