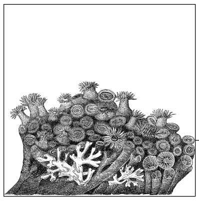

1 Introduction

Despite its large code base (over seven million lines of code), the Linux kernel is the most flexible operating system that has ever been created. It can be tuned for a wide range of different systems, running on everything from a radio-controlled model helicoptor, to a cell phone, to the majority of the largest supercomputers in the world. By customizing the kernel for your specific environment, it is possible to create something that is both smaller and faster than the kernel provided by most Linux distributions. This book will go into how to build and install a custom kernel, and provide some hints on how to enable specific options that you will probably wish to use for different situations.

No Linux distribution provides the exact kernel most of its users want. Modern distributions have gotten very accommodating, compiling in support for every known device, for sound, and even for power conservation. But you will likely have a need that's different from the majority of users (and every distribution has to try to meet the needs of the majority). You may just have different hardware. And when a new kernel comes out, you may want to start using it without waiting for a distribution to be built around it.

For a host of reasons, you will want during your Linux career to sometimes build a kernel, or to tweak the parameters of one you are running. This book gives you the information you need to understand the kernel from a user's point of view, and to make the most common changes.

There are also good reasons to remove features from the kernel, particularly if you are running it on an embedded system or one with a small form factor.

When tweaking, it's helpful to understand the internals of kernel behavior. These are beyond the scope of this book, except for brief summaries that appear with certain options. Appendix B includes references to other books and material that can give you more background.

Using This Book

Do not configure or build your kernel with superuser permissions enabled!

This warning is the most important thing to remember while working through the steps in this book. Everything in this book—downloading the kernel source code, uncompressing it, configuring the kernel, and building it—should be done as a normal user on the machine. Only the two or three commands it takes to install a new kernel should be done as the superuser (*root*).

There have been bugs in the kernel build process in the past, causing some special files in the */dev* directory to be deleted if the user had superuser permissions while building the Linux kernel.* There are also issues that can easily arise when uncompressing the Linux kernel with superuser rights, as some of the files in the kernel source package will not end up with the proper permissions and will cause build errors later.

The kernel source code should also never be placed in the */usr/src/linux/* directory, as that is the location of the kernel that the system libraries were built against, not your new custom kernel. Do not do any kernel development under the */usr/src/* directory tree at all, but only in a local user directory where nothing bad can happen to the system.

* This took quite a while to fix, as none of the primary kernel developers build kernels as root, so they did not suffer from the bug. A number of weeks went by before it was finally determined that the act of building the kernel was the problem. A number of kernel developers half-jokingly suggested that the bug remain in, to help prevent anyone from building the kernel as root, but calmer heads prevailed and the bug in the build system was fixed.