Threading Models

THE LINUX FOUNDATION

Processes and Threads

- A **process** is a running instance of a program and contains information about environment variables, file descriptors and current directory, etc.
- It can contain one or more **threads**, each of which has the same process ID and shares the same environment, memory regions (except for stack), etc.
- While Linux permits thread (or process) creation through the low level **`clone()`** system call, the most usual method is to use **`pthread_create()`** which is part of the standard **POSIX Threads** (**pthreads**) library

Implementation of pthreads

- In a perfect world any pthreads compliant program should be *write once*, *use anywhere*
- The real world is not always so simple; for example, one operating system's implementation of pthreads may be more forgiving of errors than another and when an application is ported it may stop working
- Instances have been seen, where **Solaris** was automatically initializing improperly uninitialized mutexes, while Linux did not and thus led to program failure
- Also system differences such as default stack sizes can lead to unforeseen problems
- Other operating systems may also introduce their own specific multi-threading environments, such as Solaris threads
- Porting applications from such environments may be simple or difficult depending on APIs; it is best to stick with pthreads if possible, for portability alone