

Rocky Arkan Adnan Ahmad

1806186566

SysProg-A

Hal yang sudah dipelajari dan dipahami:

1. Process, Program, dan Thread

Program adalah file yang berisi instruksi-intruksi untuk dieksekusi oleh system, atau bagaimana cara membuat proses untuk program tersebut ketika run time. Proses adalah entitas yang didefinisikan oleh kernel dimana resource dari system akan digunakan untuk mengeksekusi program. Sehingga, proses adalah instance dari program yang dieksekusi. Sebuah program dapat membutuhkan lebih dari satu proses yang dijalankan. Sebuah proses diassign dengan nomor unik bernama PID (Process Identification Number).

Lalu, thread adalah unit terkecil dari sebuah proses yang dapat dieksekusi. Misalnya, ketika menjalankan suatu program maka system akan membuat proses untuk program tersebut, dan proses tersebut diassign ke sebuah main thread. Sebuah proses bisa mempunyai lebih dari satu thread dengan tugasnya masing-masing.

Misalnya contoh dari program web browser. Dalam web browser terdapat proses yang menghandle networking untuk menyambung ke internet dan mendapatkan informasi dari alamat yang dicapai, dan juga renderer untuk HTML page yang didapatkan oleh web browser.

2. fork(), child process, parent process

fork() akan membuat sebuah proses baru dengan cara menduplikasi sebuah proses. Proses baru tersebut dinamakan child process, dan proses yang diduplikasikan dinamakan parent process. Di dalam linux, semua process merupakan sebuah child process, dengan pengecualian process init. Dalam linux, init adalah systemd dengan PID 1.

3. Command ps

Command ps di terminal merupakan command yang mengembalikan informasi dari list proses. Jika hanya mengetik ps saja tanpa option, maka hanya mengoutput informasi proses dari ps itu sendiri dan juga terminal yang memanggil.

4. Zombie process, orphaned process, and daemon process

Zombie Process adalah sebuah proses yang sudah menyelesaikan eksekusinya (via `exit()` system call) tapi entrinya masih ada di proses table karena parent processnya tidak memanggil `wait()` system call untuk mendapatkan exit code dari proses tersebut dan menghilangkannya dari proses table. Walaupun zombie process tidak menggunakan resource apapun (karena sudah selesai dan tidak mengeksekusi fungsi apa-apa lagi), zombie process memakan tempat dari proses table, dan karena proses table merupakan resource finite, jika proses table sudah terpenuhi maka OS tidak dapat membuat suatu proses lagi.

Orphaned Process adalah sebuah proses yang masih berjalan namun parent processnya sudah `exit` / menyelesaikan eksekusi. Ketika hal tersebut terjadi, maka proses tersebut akan langsung diadopsi oleh `init`.

Daemon Process adalah sebuah proses yang berjalan dibelakang layar atau background process yang tidak dalam kontrol langsung oleh user. Parent dari proses daemon adalah `init` process, sehingga secara esensi daemon process merupakan sebuah orphaned process.

Hal yang dipelajari namun belum dipahami:

Untuk minggu ini belum ada materi yang belum saya pahami.