

## ## Week 1 Logbook

### ### System Call

Pada minggu ini saya mempelajari apa itu system call. System call adalah suatu fungsi yang menjadi entry point antara aplikasi ke kernel mode untuk mengakses kebutuhan yang diperlukan. Misalnya untuk mengakses memori, hardware, task management, i/o dan lainnya.

### ### Flow Pemanggilan System Call dari Aplikasi

1. Ketika aplikasi membutuhkan fungsi yang ada pada kernel, aplikasi tersebut akan menggunakan wrapper function yang berfungsi sebagai inisialisasi fungsi yang akan dipanggil serta menyiapkan memori yang akan dipanggil sampai register yang akan digunakan oleh kernel.
2. Lalu, wrapper function akan meminta trap handler, yaitu suatu program/fungsi yang bisa mengubah dari user mode ke kernel mode dan sebaliknya untuk memanggil fungsi-fungsi system call yang telah diminta oleh wrapper function.
3. Setelah dipanggil oleh trap handler, fungsi system call tersebut akan dieksekusi pada register dan stack.  
Jika eksekusi fungsi tersebut berhasil maupun gagal, fungsi tersebut akan mengembalikan sebuah value kepada trap handler dan diteruskan kepada wrapper function yang dipanggil oleh aplikasi.

### ### Library Functions

Library Functions adalah sebuah fungsi built in pada program C yang bisa dijalankan juga pada terminal/shell. Library Functions ini bisa didefinisikan menjadi 2, yaitu Library Functions yang memanggil system call dan Library Functions yang tidak menggunakan system call. Contoh library functions adalah printf, malloc dan lain-lain.

### ### Error Handling pada System Call

Cara untuk mengetahui adanya error ketika sedang mengeksekusi fungsi system call adalah dengan menyimpan sebuah integer ke dalam suatu variabel yang dideklarasikan secara eksternal bernama errno. Setelah disimpan ke dalam variabel, kita dapat memilih untuk mencetak suatu pesan tertentu berdasarkan isi/value dari variabel errno ini dengan menggunakan fungsi perror(String). note: dalam case ini adalah pada program c.

Command yang dapat menampilkan list informasi error code apa saja serta menampilkan angka yang merepresentasikan error code tersebut adalah `errno -ls`.

### ### Contoh-contoh System Call dan Fungsinya

1. ``gettimeofday``: get today time
2. ``getrusage``: get resource usage
3. ``write``: send a message to another user
4. ``mmap``: map or unmap files or devices into memory
5. ``fork``: create a child process
6. ``kill``: send a signal to a process for termination
7. ``brk``: change data segment size
8. ``ioctl``: control device
9. ``socket``: create an endpoint for communication
10. ``getcpu``: determine CPU and NUMA node on which the calling thread is running

### ### Alasan Menggunakan Library Functions ketimbang System Call

Dengan menggunakan library functions, error/bug akan lebih mudah debug ketimbang menggunakan system call. Hal itu terjadi karena system call langsung dieksekusi fungsinya oleh kernel sehingga kita tidak dapat menentukan dimana letak kesalahannya. Selain itu, library functions dapat dengan mudah dipanggil oleh programmer ketika membuat aplikasi. Sedangkan untuk system call, fungsi-fungsi tersebut tidak dapat dipanggil oleh programmer karena fungsi tersebut hanya berjalan di kernel mode saja.