CSIM602155: Tugas #3 - *Data Parallelism*: Enkripsi dan Dekripsi File

Kuliah Sistem Operasi untuk Sistem Informasi
Semester Gasal 2023/2024

Heri Kurniawan

Due date: Sabtu, 4 November 2023, Pukul: 23.55 WIB

1 Tujuan

- Mahasiswa mampu mengimplementasikan *data parallelism* dengan memanfaatkan library `pthread` pada lingkungan sistem operasi Linux.
- Mahasiswa mampu mengimplementasikan sinkronisasi antar thread dengan menggunakan semaphore dalam bahasa C.

2 Deskripsi

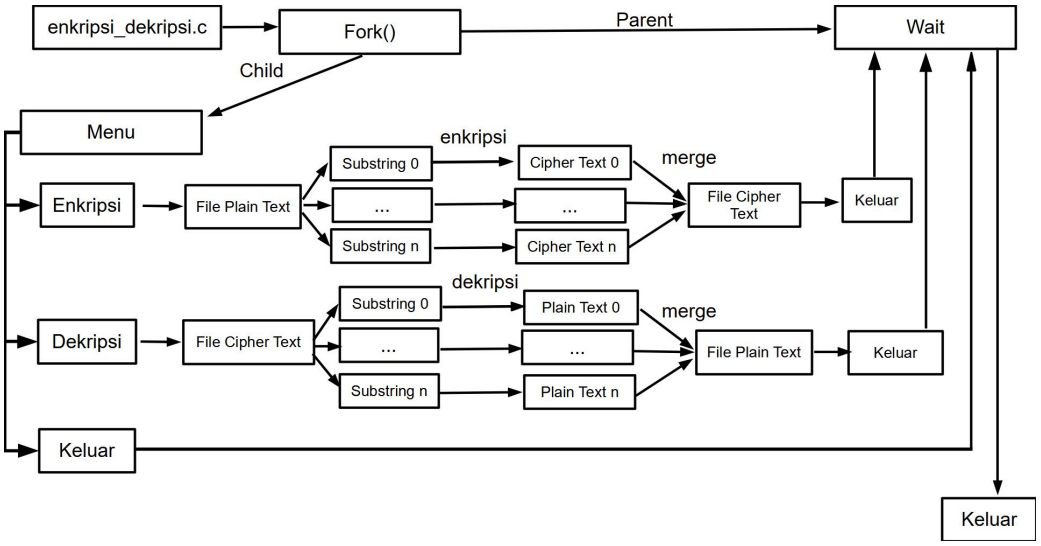

Figure 1: Diagram Program Enkripsi - Dekripsi

Pada tugas individu ini Anda diminta untuk mengimplementasikan *data parallelism* pada proses enkripsi dan dekripsi file dengan memanfaatkan fungsi-fungsi yang tersedia pada library `pthread`. Program yang dibuat mempunyai spesifikasi sebagai berikut :

1. **Menu utama**

   Setelah program dijalankan, program akan mengeksekusi `fork()` untuk membuat proses child. Proses child akan menampilkan menu program yang terdiri dari Enkripsi, Dekripsi atau Keluar sedangkan proses parent akan `wait()` hingga proses child *terminate*.

2. **Enkripsi**

   Menu ini akan melakukan enkripsi file. Program akan meminta input:

   - Jumlah thread yang akan melakukan proses enkripsi. Jumlah thread = 1-10 thread.
   - Nama input file yang akan dienkripsi
   - Secret key enkripsi dengan jumlah karakter, 8-20 karakter
   - Nama output file hasil dekripsi

   Setelah memasukkan nama output file hasil dekripsi, program akan melakukan pembacaan file dan memasukkan konten hasil pembacaan ke variabel string. Kemudian program akan membuat sejumlah thread yang akan melakukan operasi *data parallelism*, dimana potongan string akan di enkripsi oleh masing-masing thread. Sebagai contoh jika terdapat 3 thread dan variabel string berisikan 50 karakter, maka Thread0 mengenkripsi potongan string 0-15, Thread1: 16-31, dan Thread2: 31-49.

   Algoritma enkripsi yang diimplementasikan oleh setiap thread adalah *XOR encryption* dimana setiap karakter dari potongan string akan di ^(XOR) dengan karakter pada secret key dengan formula: `string[i]^secret_key[i%panjang_key]`, dimana `i` = index. Hasil konversi dari setiap thread digabungkan pada satu variabel string, kemudian ditulis pada file hasil dekripsi. Tips: hindari hasil XOR berupa karakter NULL ('\0').

   Setiap thread yang telah selesai melakukan penggabungan string, harus melakukan operasi pertambahan pada variabel `counter` yang dishare bersama dengan thread lain. Counter bersifat akumulatif, sebagai contoh:

   ```
   Thread Id = 0 - status:  (16/50) karakter telah dienkripsi
   Thread Id = 2 - status:  (32/50) karakter telah dienkripsi
   Thread Id = 1 - status:  (50/50) karakter telah dienkripsi
   ```

   Thread0, menambahkan nilai 16 pada counter, dilanjutkan dengan Thread2 sehingga counter bertambah menjadi 32. Angka 50, menandakan total karakter. Saat satu thread melakukan operasi pertambahan, thread lain harus menunggu hingga thread tersebut selesai. Terapkan semaphore pada operasi tersebut.

3. **Dekripsi**

   Proses dekripsi akan melakukan konversi konten file dekripsi ke konten file awal. Program akan meminta input jumlah thread, nama file yang akan didekripsi, secret key dan nama output file. Gunakan secret key sama dengan secret key pada proses enkripsi. Selebihnya proses dekripsi akan sama dengan proses enkripsi yaitu mengimplementasikan *data parallelism* dan menggunakan *XOR encryption* untuk konversi setiap potongan string.

4. **Verifikasi input (BONUS 10 Poin)**

   Lakukan verifikasi input agar program berjalan dengan baik. Verifikasi dilakukan pada:

   (a) Menu pilihan enkripsi, dekripsi dan keluar.
   Pastikan user tidak memasukkan input karakter lain selain 1, 2, atau 3

   (b) Jumlah thread
   Pastikan user tidak memasukkan bilangan atau karakter lain selain angka dalam range 1-10.

   (c) Ketersediaan file input
   Cek apakah file input yang akan dibaca ada pada sistem.

   (d) Panjang secret key
   Verifikasi panjang karakter untuk secret key, apakah panjangnya berada dalam range 8-20 karakter atau tidak.

   (e) Pengecekan file output
   Verifikasi apakah nama file output yang sama ada pada sistem. Jika sama, maka lakukan konfirmasi apakah file yang sudah ada akan di *overwrite* atau tidak

5. **Lain-lain**

   - Program tidak membuat thread jika user memilih menu keluar.
   - Thread main pada proses *child* akan melakukan `pthread_join` untuk setiap thread yang sudah dibuat.
   - Gunakan file `plain.txt` dan `kernel.xml` untuk pengetesan enkripsi dan pengecekan hasil dekripsi. Pastikan konten file dekripsi program Anda, sama dengan kedua file tersebut. Kedua file berada pada folder yang sama dengan program `enkripsi_dekripsi.c`.

Implementasikan program diatas pada mesin virtual sistem operasi Linux. Berikan komentar pada script untuk memperjelas cara kerja script Anda. Untuk mendapatkan nilai maksimal 110 poin (poin total + bonus) dan mempermudah penilaian, output program Anda HARUS sama seperti pada tampilan program berikut: `https://youtu.be/-NEF3vNQQuM`.

Saran untuk mempercepat pengerjaan tugas:

- Pahami alur logik program.
- Lakukan teknik *divide and conquer* untuk mempermudah pengembangan program.
- Implementasi sinkronisasi variable `counter` dengan menggunakan binary semaphore.
- Kerjakan tugas sejak dini karena Anda tidak tahu masalah apa yang akan Anda hadapi kedepannya.
- Gunakan google jika menemui permasalahan. Google adalah teman Anda :)

3 Pengumpulan

1. Buat video demo program beserta penjelasannya pada youtube dengan mode unlisted (tidak publik). Pada video, presenter dapat menampakkan wajah ataupun hanya audio suara tanpa wajah. File yang harus Anda kumpulkan adalah:

   - `enkripsi_dekripsi.c`

     Berisikan program utama beserta komentar program. Jangan lupa informasikan Nama dan NPM Anda dalam blok komentar.

   - `readme.txt`

     Berisikan informasi Nama, NPM, link video youtube dan cara menjalankan program

   - `Makefile`

2. Zip semua file diatas dan upload program via Scele dengan penamaan: `t3-npm-nama.zip`

Hindari tindakan PLAGIARISME. Terbukti melakukan plagiarisme akan mendapatkan SANKSI maksimal nilai E.

Selamat Mengerjakan