**INTER-PROCESS COMMUNICATION TRONG LINUX**

1. **IPC là gì?**

Sự thực hiện giao tiếp giữa các thành phần khác và với kernel để hợp tác thực hiện một tiến trình nào đó. IPC là một phương thức không thể thiếu trong việc giúp các process trao đổi thông tin với nhau.

Hai models chính của IPC là shared memory( chia sẻ bộ nhớ) với nhiệm vụ hình thành khu vực lưu trữ bộ nhớ chung và message passing ( truyền tin) với nhiệm vụ truyền tải tin nhắn liên tục giữa các process.

1. **Semaphore**

Semaphore là gì?

* 1. **Define semaphore**

**Nó là một kỹ thuật cho phép xử ý nội dung hoặc thay đổi luồng, giám sát các truy vấn, điều khiển shared system resources. Semaphores giải quyết được vấn đề xử lý nhiều tiến trình trong hệ thống. Vấn đề xảy ra khi nhiều tiến trình cùng cố gắng truy cập vào tài nguyên. Lúc dó có thể xảy ra các vấn đề như là ngẽn...**

* 1. **Semaphore trong linux**

**Semaphore là một cấu trúc dữ liệu, được dùng để đồng bộ tại nguyên và đồng bộ hoạt động. Có nghĩa là ta quản lý được các thread được phép truy cập vào tài nguyên, hay đợi các thread khác.**

**Nó hoạt động như thế nào ? Ta giả sử có hai tiến trình cùng muốn truy cập vào database. Tôi sẽ lấy ví dự là binary semaphone. Bắt đầu giá trị của nó bằng 1 và có thể được truy cập đồng thời bởi nhiều tiến trình. Hai tiến trình đos được gọi từ một chương trình. Lúc đó hai tiến trình đó chia sẻ biến semaphore. Khi một trong các tiến trình thực hiến. nó sẽ có semaphore, lúc đó nó có quyền truy cập vào critical section và tiến trình thứ 2 bị ngăn cản.**

### **Hàm semget**

hàm semget tạo một semaphore mới và chưa semaphore key.

### **Hàm semop**

Hàm semop được dùng để thay đổi giá trị của semaphore. Cú pháp

int semop(int sem\_id, struct sembuf \*sem\_óp, size\_t num\_sem\_ops)

sem\_id là định danh semaphore.

sem\_ops là một con trỏ trỏ tới một mảng chứa các cấu trúc. chung có những thuộc tính

struct\_sembuf

{

short sem\_num; // là số semaphore number.

short sem\_op; // giá trị của semaphore được thay đổi.

short sem\_flg;// thường được đặt là SEM\_UNDO.

}

### **Hàm semctl**

Hàm cho phép điều khiển trực tiếp thông tin của semaphore.. Cú pháp int semctl (int sem\_id, int sem\_num, int commands, …);

1. **Shared Memory**

Shared memory là gì?

(chia sẻ bộ nhớ) với nhiệm vụ hình thành khu vực lưu trữ bộ nhơs chung. Vùng nhớ chung này được tạo thành từ nhiều vùng nhớ riêng của mỗi process. Các process khác muốn tiếp vận vùng nhớ đó sẽ phải lưu địa chỉ của vùng nhớ chung ấy vào vùng nhớ riêng của mình.

* 1. **Sử dụng shared memory**

Linux cung cấp các hàm hệ thống được sử dụng để tạo và sử dụng shared memory như sau:

* + 1. shmget()

Dùng để tạo shared memory

cú pháp int shmget(key\_t key, size\_t size ,int shmflg);

Đối với semaphore, trương trình cung cấp key,

* + 1. shmat()
    2. shmdt()
  1. **Đánh giá việc sử dụng shared memory**
* Ưu điểm

+

* Nhược điểm

+

1. **Message Queues**
   1. **Sử dụng message queue**
      1. msgget ():
      2. msgsnd ():
      3. msgrcv ():
      4. msgctl ():
   2. **Đánh giá việc sử dụng message queue**