

# BÁO CÁO TÌM HIỂU VỀ ĐỒ ÁN 1

Nguyễn Trần Hậu - MSSV 1612180

21/09/2018

## Tóm tắt nội dung

Mục tiêu hiểu về Linux kernel module và hệ thống quản lý file và device trong Linux, giao tiếp giữa tiến trình ở user space và kernel space.

## 1 Linux kernel module

Kernel module là một đoạn code có thể được tải vào kernel hoặc ngừng tải tùy theo yêu cầu. Kernel module dùng để mở rộng các tính năng của kernel mà không cần phải khởi động lại máy. Ví dụ như driver của thiết bị là một kernel module, khi máy tính nhận thiết bị mới như bàn phím hay chuột, driver của thiết bị sẽ được tải vào kernel.

Nếu không có kernel module, chúng ta phải tải tất cả driver của thiết bị vào trong kernel dẫn đến dung lượng của kernel trở nên rất lớn. Bên cạnh đó mỗi lần sửa lỗi driver của thiết bị hoặc thêm tính năng mới đều yêu cầu build lại kernel và khởi động lại máy, dẫn đến tốn thời gian và chi phí.

Trong Linux, kernel module viết bằng C, sau khi compile sẽ có dạng \*.ko. Cài đặt kernel module bằng câu lệnh `insmod` hoặc `modprobe` và gỡ cài đặt bằng câu lệnh `rmmmod`. Xem thông tin của kernel module về tác giả, license, bằng câu lệnh `modinfo`.

Chương trình C thông thường sẽ thực thi những câu lệnh trong hàm `main()`. Còn kernel module khi tải vào kernel sẽ gọi hàm được xác định bằng `module_init`. Và khi kernel module kết thúc, sẽ gọi hàm được xác định bằng `module_exit`

## 2 Hệ thống quản lý file và device trong Linux

Linux duy trì một cấu trúc duy nhất để quản lý file. Đỉnh của cấu trúc là root directory, tên là /. Tất cả mọi thứ trong Linux đều là file[6]. Có nghĩa là tiến trình, file, thư mục, socket, ... đều được biểu diễn bằng file. Trong `/dev` chứa các file đặc biệt dùng gọi là device file.

Có 2 loại là character device và block device. Những file này là interface cho driver của các thiết bị. Ví dụ như `/dev/cdrom` chỉ tới ổ CD-ROM của máy tính.

Character device là các device có driver giao tiếp bằng cách gửi nhận từng character. Character device thường là bàn phím, chuột, .... Block device là các device có driver giao tiếp bằng cách gửi nhận từng block dữ liệu. Block device thường là ổ cứng, USB, ....

## 3 Giao tiếp giữa tiến trình ở user space và kernel space

Bộ nhớ trong Linux được chia thành 2 vùng phân biệt là user space và kernel space. Kernel space là nơi kernel thực thi các tác vụ. User space là nơi các tiến trình người dùng được thực thi.

Trong user space, tiến trình bị giới hạn khả năng truy cập vào bộ nhớ còn kernel space thì có toàn quyền truy cập. Tiến trình trong user space có thể truy cập vào một phần nhỏ của kernel space bằng **system calls**.

## Tài liệu

- [1] Writing a Linux Kernel Module - Part 1  
<http://derekmolloy.ie/writing-a-linux-kernel-module-part-1-introduction/>
- [2] Writing a Linux Kernel Module - Part 2  
<http://derekmolloy.ie/writing-a-linux-kernel-module-part-2-a-character-device/>
- [3] The Linux Kernel Module Programming Guide  
<https://www.tldp.org/LDP/lkmpg/2.6/html/index.html>

- [4] The Linux Kernel, Kernel Modules And Hardware Drivers  
<http://haifux.org/lectures/86-sil/kernel-modules-drivers/>
- [5] Kernel module on Arch Linux wiki  
[https://wiki.archlinux.org/index.php/Kernel\\_module](https://wiki.archlinux.org/index.php/Kernel_module)
- [6] Everything is a file  
[https://en.wikipedia.org/wiki/Everything\\_is\\_a\\_file](https://en.wikipedia.org/wiki/Everything_is_a_file)
- [7] Linux Filesystem Hierarchy  
<https://www.tldp.org/LDP/Linux-Filesystem-Hierarchy/html/dev.html>
- [8] Kernel Space Definition  
[http://www.linfo.org/kernel\\_space.html](http://www.linfo.org/kernel_space.html)
- [9] What is difference between User space and Kernel space?  
<https://unix.stackexchange.com/questions/87625/what-is-difference-between-user-space-and-kernel-space>