OPERATING SYSTEM DESIGN

S Thenmozhi

Department of Computer Applications

OPERATING SYSTEM DESIGN

OS Structures & Kernel Programming

S Thenmozhi
Department of Computer Applications

Kernel Modules

- Can the OS functionality be extended without compiling the whole OS again?
- Yes. Kernel Modules.
- Kernel Modules are pieces of code that can be loaded and unloaded into the kernel upon demand
- Loads only the needed modules, unloads when not required and frees up memory and other resources
- Reduces the kernel size
- Possible to extend the kernel capabilities without modifying the rest of the code
- Possible to insert the module while the kernel is running

Kernel Modules

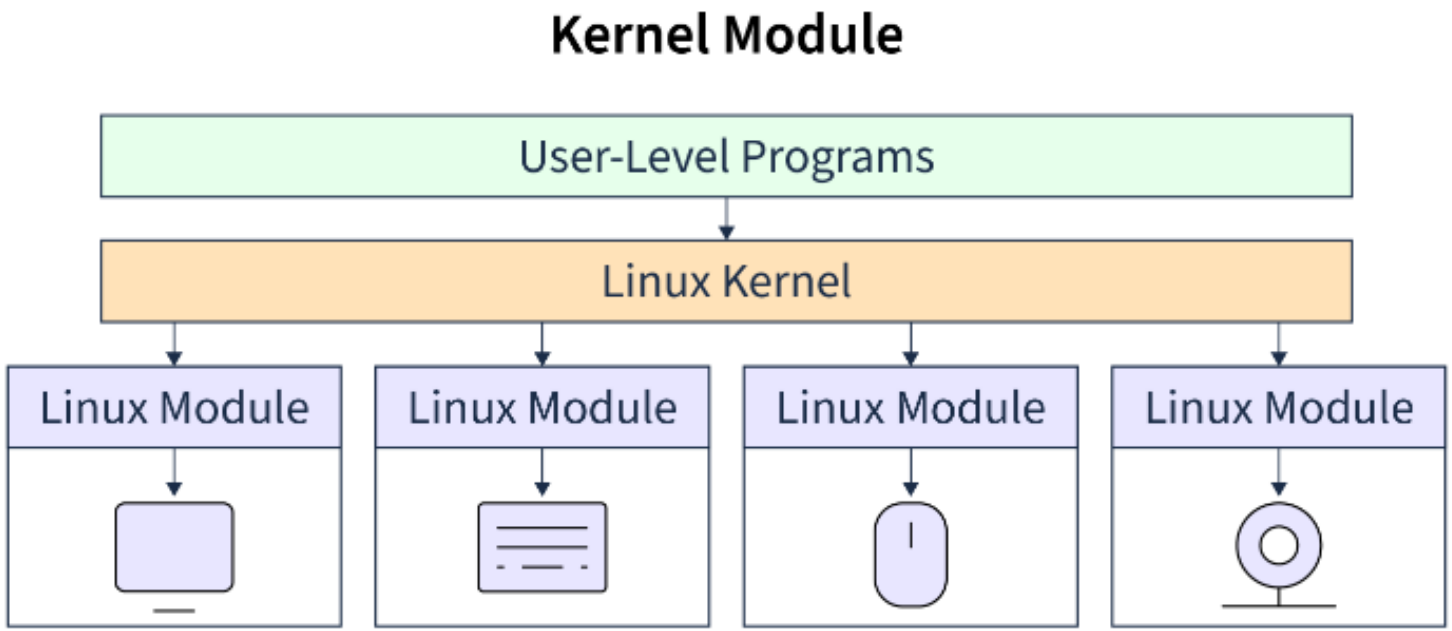
- Enables independent development of drivers for different devices
- For example, you have got a new webcam
- In order to use the web cam, your OS must need to talk to the web cam.
- For this, the web cam manufacturers must have developed a kernel module which will be added to your kernel to talk to that device
- Use Cases: Supporting new network cards, adding file system support, supplying device drivers for peripherals, implementing security features, and enabling additional kernel functionality

Kernel Modules

- Older Monolithic kernels lack the property of modularity and extensibility
- But modern monolithic kernels, solves the problem with the help of kernel modules
- Kernel module is an object file that contains a code that can extend the kernel functionality at runtime
- Most of the device drivers are used in the form of kernel modules

OPERATING SYSTEMS DESIGN

Kernel Modules



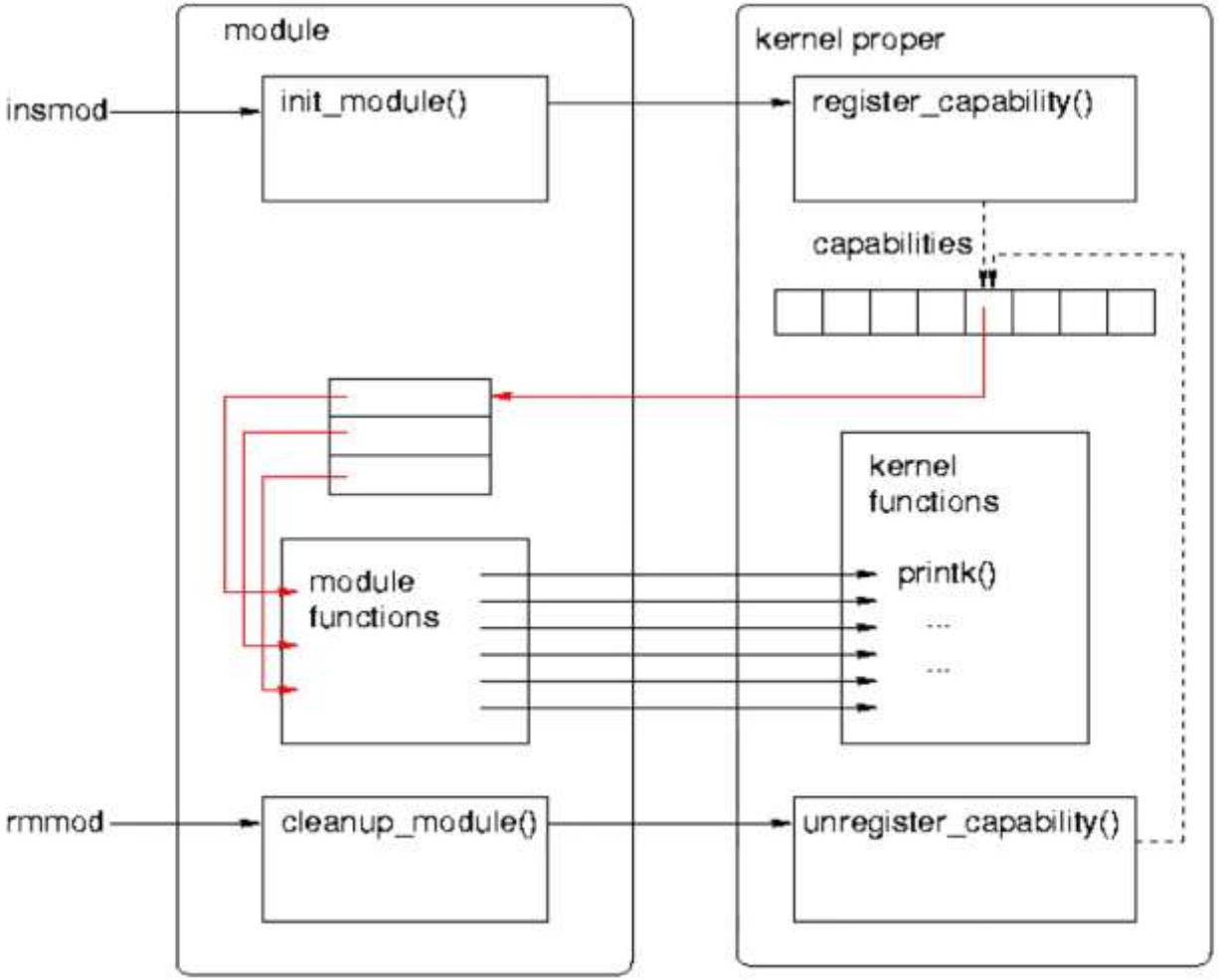
Kernel Modules

What does these modules do?

- Register to the running kernel
- Once registered, whenever the user uses the functionality of that device, the kernel module loaded in the kernel call the kernel function and executes
- When the module is no longer required, it is unregistered from the kernel

OPERATING SYSTEMS DESIGN

Kernel Modules



Kernel Modules

- kernel modules are stored in distinct files with the extension ".ko“
- Linux kernel modules are organized into subdirectories based on their purpose or category
- These subdirectories provide a structured way to manage and organize kernel modules
- The main subdirectories for kernel modules are typically found under the **/lib/modules//** directory

Kernel Modules

Common sub directories

- **kernel:** This directory contains core kernel modules, including fundamental functions such as process management, memory management, and system calls.
- **drivers:** The drivers' directory is further divided into various subdirectories, each representing a different category of hardware drivers. For example, you may find subdirectories like a net for network drivers, sound for audio drivers, USB for USB drivers, and so on
- **fs:** This directory contains file system modules. Each file system, such as **ext4**, **NTFS**, or **FAT**, may have its subdirectory

Kernel Modules

- **net:** The net directory contains network-related modules, including protocols, device drivers, and network stack components
- **crypto:** This directory holds cryptographic modules that provide encryption, decryption, hashing, and other cryptographic operations
- **sound:** Sound-related modules, such as sound card drivers, mixer controls, and audio frameworks, are located in this directory.
- **video:** The video directory contains modules related to video and graphics, including graphics card drivers, frame buffer drivers, and video-related utilities.

Kernel Modules

Difference between kernel Module and System Call

Aspect	Kernel Module	System Call
Definition	Extensible part of the kernel	Interface for user-kernel interaction
Modularity	Can be loaded/unloaded dynamically	Predefined, cannot be added/removed dynamically
Scope	Extends kernel functionality	Provides services to user-space applications
Execution	Runs in kernel space	Invoked from user space, executed in kernel mode
Examples	Device drivers, file systems	open(), read(), write()

THANK YOU

S Thenmozhi

Department of Computer Applications

thenmozhis@pes.edu

+91 80 6666 3333 Extn 393