

Linux Kernel Modül Geliştirme

Modül Nedir ?

Kernel modülü, kernele sonradan eklenip çıkartılabilen kodlardır. C ile geliştirilen Kernel Modülleri aslında sıradan bir programdan çok da farklı değildir. Kernelinizde kurulu olan modülleri **lsmod** komutuyla görebilirsiniz.

Neden Modül Kullanmalıyız ?

Linux'da donanımlar kernel sürücüleri tarafından kullanılırlar. Bu modüller genellikle lib/modules dizisi altında bulunurlar ve açılış anında yüklenirler. Sistem içerisinde hazır modüller olduğu gibi kendimiz de modül ekleyebiliriz. Kernelin fonksiyonelliğini sistemi yeniden başlatmadan modüller sayesinde arttırılabilir. Bir donanım sürücüsünü düşünecek olursak, sürücü ile kernel, sistem ve donanım arasındaki bağlantıyı sağlayacaktır. Fakat bu işlem için kernele eklenti yapmamız gerekecektir. Kernele yapılacak her hangi bir müdahalede, kernelin tekrar derlenmesi ve sistemin yeniden başlatılması gerekir. Suncu sistemleri gibi sürekli açık kalması gereken bilgisayarlar için yeniden başlatma işlemi istenmeyen bir durumdur. Bu tip durumları önlemek için modüller kullanılmaktadır.

Kernel Modülü Nasıl Yüklenir ?

Kernel modülleri ile yapılacak işlemler için "**insmod**", "**modprobe**", "**rmmod**" ve "**lsmod**" gibi modül araçları kullanılır.

Kernele yeni bir modül eklemek için insmod ve ya modprobe kullanılır. Insmod kernele tek bir modül kurar, modprobe ise ilgili modüle bağımlı diğer modülleri otomatik olarak yükler.

Bazı modüller diğer modüllerin yüklenmiş olmasına bağımlıdır ve bağımlı olunan modül yüklü değilse insmod komutu başarısız olur. Bu durumda önce bağımlı olunan modülleri tek tek yüklemeniz gerekir ya da modprobe kullanılabilir.

```
# insmod /home/merve/my_module.ko
```

komutu, belirttiğiniz koutmdaki modülü yükleyecektir.

Bağımlılıkların otomatik olarak bulunup yüklenmesi için ise:

```
# modprobe my_module
```

İki durumda da yükleme sırasında ek seçenekler belirtilebilir. Ekran kartı sürücüsüne masaüstü çözünürlüğünü belirtmek gibi.

Insmod komutunda modülün yolu dosya uzantısıyla birlikte belirtilmelidir. Ancak modprobe komutunda böyle bir durum yoktur, sadece modül adını belirtmek yeterlidir.

Kernel modüllerini kaldırmak için ise “**rmmod**” komutu kullanılır.

```
# rmmod my_module
```

Burada dikkat edilmesi gereken husus, silmek istediğiniz modülün başka modüller tarafından kullanılabilir olmasıdır. Eğer modül kullanılmıyorsa, silinecektir. Eğer kullanılıyorsa bağımlı olduğu modüller listelenir ya da bu modül kullanılıyor şeklinde hata verir. Eğer modülü bağımlı modüller ile birlikte kaldırmak istiyorsanız “modprobe -r” komutunu kullanmalısınız.

Kernel Modülü Nasıl Yazılır ?

Modüller programlardan farklı olarak main() işlevine sahip değildirler. **main()** işlevi yerine "init_module" ve "cleanup_module" işlev çifti vardır:

init_module() işlevi, modül çekirdeğe ilk yüklendiği zaman çağırılır. Modül kaldırılmak istendiğinde ise cleanup_module() işlevi çağırılarak init_module() ün yaptıklarını geri alınır. Eğer modül dinamik yapıda yazılmış ise bu çalışma sırasında yeni değişkenlerin kayıt işlemleri gerçekleşir. Ama modülünüz statik bir yapıda ise değişkenlerin kayıt işlemi "boot" sırasında yapılmaktadır.

Örnek Bir Modül

my_module.c dosyası içerisine yerleştirilecek kod :

```
#include <linux/module.h>    // included for all kernel modules
#include <linux/kernel.h>    // included for KERN_INFO
#include <linux/init.h>      // included for __init and __exit macros

MODULE_AUTHOR("Merve");
MODULE_DESCRIPTION("A Simple Hello World module");

static int __init hello_init(void)
{
    printk(KERN_INFO "Hello world!\n");
    return 0;    // Non-zero return means that the module couldn't be
loaded.
}

static void __exit hello_cleanup(void)
{
    printk(KERN_INFO "Cleaning up module.\n");
}

module_init(hello_init);
module_exit(hello_cleanup);
```

linux/module.h header dosyası bütün kernel modüllerde kullanılır. linux/kernel.h header

dosyası ise **printk()** kullanabilmek için kullanılır. `printk()` işlevi, kernel seviyesinde ekrana yazdırma için kullanılır. C kütüphanesindeki `printf()` fonksiyonu gibi davranır fakat kernel seviyesinde ekrana direk olarak çıktı vermez. Yazdırılan değerler `"/var/log/syslog"` dosyasında ya da `dmesg` komutunun çıktısında görülebilir.

Makefile dosyası içerisine yerleştirilecek kod :

```
obj-m += my_module.o
all:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
clean:
    make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

`make` komutu ile modülümüzü derleyelim.

Derleme İşlemi Nasıl Yapılır?

Öncelikle oluşturmuş olduğumuz iki dosyayı bir dizin içerisine yerleştiriyoruz. Daha sonra **“make”** komutu ile modülümüzü derliyoruz. `Modinfo` komutu ile derlenen modül hakkında bilgilere erişebiliriz.

Bu işlemlerden sonra, modülümüzü kernelin içerisine yerleştiriyoruz. Bunun için **root** kullanıcı olmanız gerekmektedir. **“sudo insmod my_module.ko”** komutu ile kernel'a modülümüzü ekliyoruz.

```
$ make
$ sudo insmod my_modul.ko // içeri aktarma.
$ lsmod | grep my_modul //kontrol.
$ sudo rmmod my_modul // modulun silinmesi.
```

Kaynaklar

- www.thegeekstuff.com
- www.tldp.org
- www.bekirkarul.com
- www.obenseven.com.tr
- www.cagdastopcu.com
- en.wikipedia.org

Yazan ve Derleyen: Merve AKIN