

Linux Çekirdek Modülü Ekleme

Yusuf KAYA

February 14, 2025

Abstract

Bu belge, çekirdek günlüğüne mesajlar yazdıran basit bir çekirdek modülünün uygulanmasını açıklamaktadır. Bu tür modüller, çekirdek işlevselliğini dinamik olarak genişletmek için yaygın olarak kullanılır ve sistem yeniden başlatılmadan eklenip kaldırılabilir.

1 Giriş

Çekirdek modülleri, ihtiyaç duyulduğunda çekirdeğe yüklenebilen veya çekirdekten kaldırılabilen kod parçalarıdır. Bu modüller, sistem yeniden başlatılmadan çekirdeğin işlevselliğini genişletmenin bir yolunu sunar ve genellikle sürücüler veya belirli sistem işlevleri gibi özel görevler için kullanılır. Bu belgede, temel bir çekirdek modülünün nasıl oluşturulacağı ve kullanılacağı gösterilmektedir.

2 Kod Uygulaması

Kod uygulaması, iki ana bölümden oluşmaktadır: modülün sisteme yüklendiği zaman çağrılan başlatma fonksiyonu ve modülün sistemden kaldırıldığında çağrılan çıkış fonksiyonu.

2.1 Başlatma Fonksiyonu

Başlatma fonksiyonu, çekirdek modülünün yüklendiği anda çalıştırılır ve modülün kurulum işlevlerini yerine getirir. Aşağıdaki kod örneği, basit bir başlatma fonksiyonunu göstermektedir. Bu fonksiyon, yüklenirken "Hello, World" mesajını çekirdek günlüğüne yazdırır.

```
1 #include <linux/init.h>
2 #include <linux/kernel.h>
3 #include <linux/module.h>
4
5 static int __init modulismi_init(void) {
6     pr_info("Hello, World \n");
7     return 0;
8 }
9
10 module_init(modulismi_init);
11 MODULE_DESCRIPTION("kernel module - baslangic");
12 MODULE_AUTHOR("Me");
13 MODULE_LICENSE("GPL");
```

Listing 1: Başlatma fonksiyonu

2.2 Çıkış Fonksiyonu

Çıkış fonksiyonu, çekirdek modülünün sistemden kaldırılmasından hemen önce çağrılır ve modülün kapatma işlemlerini gerçekleştirir. Aşağıdaki kod örneği, çıkış fonksiyonunun nasıl tanımlandığını göstermektedir. Bu fonksiyon, kaldırılma sırasında "Goodbye, World" mesajını çekirdek günlüğüne yazar.

```
1 #include <linux/kernel.h>
2 #include <linux/module.h>
3
4 static void __exit modulismi_exit(void) {
5     pr_info("Goodbye, world \n");
6 }
7
8 module_exit(modulismi_exit);
9 MODULE_DESCRIPTION("kernel module - cikis");
10 MODULE_AUTHOR("Me");
11 MODULE_LICENSE("GPL");
```

Listing 2: Çıkış fonksiyonu

2.3 Makefile

Çekirdek modülünü derlemek için kullanılan Makefile, çekirdek modülünün derleme sürecini otomatik hale getirir. Bu dosya, çekirdek yapı sistemine hangi dosyaların derleneceğini ve hangi yolların izleneceğini bildirir.

```
1 # Makefile
2 obj-m += modulismi.o
3 yusufkaya-objs := modulismi_init.o modulismi_exit.o
4
5 PWD := $(CURDIR)
6
7 all:
8     make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
9
10 clean:
11     make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
```

Listing 3: Makefile

3 Adımlar

Bu çekirdek modülü, Linux çekirdeğinde modüllerin yüklenmesi ve kaldırılmasının temel yapısını ve işlevselliğini göstermektedir. Aşağıda, çekirdek modülünü oluşturmak ve çalıştırmak için gerekli adımlar yer almaktadır.

3.1 Kod Dosyalarını Oluşturma

Öncelikle, yukarıda verilen başlatma ve çıkış fonksiyonlarının kodlarını **aynı dizinde** oluşturuyoruz. Bu dosyalar, çekirdek modülümüzün kaynak kodunu içerir.

```
[yusuf@yusuf-modern14b10mw HW2]$ ls
Makefile yusufkaya_exit.c yusufkaya_init.c
[yusuf@yusuf-modern14b10mw HW2]$
```

3.2 Headers Dosyalarını Güncelleme

Modülleri yüklerken herhangi bir hata ile karşılaşmamak için sistemde yüklü olan çekirdek sürümüne uygun olan **headers** dosyalarının yüklenmesi gerekmektedir. Aşağıda, farklı dağıtımlar için ilgili komutlar verilmiştir.

- Debian tabanlı sistemler (örneğin, Ubuntu) için:

```
1 sudo apt-get install linux-headers-$(uname -r)
```

Listing 4: Debian için

- Arch tabanlı sistemler için:

```
1 sudo pacman -S linux-headers
```

Listing 5: Arch için

```
resolving dependencies...
looking for conflicting packages...

Packages (2) pahole-1:1.27-2 linux66-headers-6.6.54-2

Total Download Size:    32,58 MiB
Total Installed Size:  122,11 MiB

:: Proceed with installation? [Y/n] y
:: Retrieving packages...
linux66-headers-6.6.54-2-x86_64           32,3 MiB   4,65 MiB/s  00:07 [#####] 100%
pahole-1:1.27-2-x86_64                   298,3 KiB   941 KiB/s  00:00 [#####] 100%
Total (2/2)                               32,6 MiB   4,36 MiB/s  00:07 [#####] 100%
(2/2) checking keys in keyring [#####] 100%
(2/2) checking package integrity [#####] 100%
(2/2) loading package files [#####] 100%
(2/2) checking for file conflicts [#####] 100%
(2/2) checking available disk space [#####] 100%
:: Processing package changes...
(1/2) installing pahole [#####] 100%
Optional dependencies for pahole
  ostra-cg: Generate call graphs from encoded traces
(2/2) installing linux66-headers [#####] 100%
:: Running post-transaction hooks...
(1/3) Arming ConditionNeedsUpdate...
(2/3) Updating module dependencies...
(3/3) Refreshing PackageKit...
```

Headers dosyalarını yükleme örneği

3.3 Make Dosyasını Çalıştırma

Headers dosyaları doğru bir şekilde yüklendikten sonra, dosyaların bulunduğu dizinde Makefile dosyamızı çalıştırarak çekirdek modülünü derliyoruz. Başarıyla tamamlandığında, **modulismi.ko** adlı bir çekirdek modülü oluşturulacaktır.

```
[yusuf@yusuf-modern14b10mw HW2]$ make
make -C /lib/modules/6.6.54-2-MANJARO/build M=/home/yusuf/Masaüstü/HW2 modules
make[1]: Entering directory '/usr/lib/modules/6.6.54-2-MANJARO/build'
CC [M] /home/yusuf/Masaüstü/HW2/yusufkaya_init.o
CC [M] /home/yusuf/Masaüstü/HW2/yusufkaya_exit.o
LD [M] /home/yusuf/Masaüstü/HW2/yusufkaya.o
MODPOST /home/yusuf/Masaüstü/HW2/Module.symvers
CC [M] /home/yusuf/Masaüstü/HW2/yusufkaya.mod.o
LD [M] /home/yusuf/Masaüstü/HW2/yusufkaya.ko
BTF [M] /home/yusuf/Masaüstü/HW2/yusufkaya.ko
make[1]: Leaving directory '/usr/lib/modules/6.6.54-2-MANJARO/build'
```

Makefile çalıştırma örneği

3.4 Modül Yükleme ve Kaldırma

Çekirdek modülü oluşturulduktan sonra, **insmod** komutuyla modül yüklenir. Eğer daha önce yüklenmiş olan bir modülü kaldırmak istersek, **rmmod** komutunu kullanırız.

```
1 sudo insmod moduleismi.ko
```

Listing 6: Modül yüklemek için

```
1 sudo rmmod moduleismi.ko
```

Listing 7: Yüklü modülü kaldırmak için

```
[yusuf@yusuf-modern14b10mw HW2]$ sudo insmod yusufkaya.ko
[sudo] password for yusuf:
[yusuf@yusuf-modern14b10mw HW2]$ sudo rmmod yusufkaya.ko
[yusuf@yusuf-modern14b10mw HW2]$ sudo dmesg | tail -10 > out.txt
[yusuf@yusuf-modern14b10mw HW2]$
```

Modül yükleme örneği

Daha sonra, yüklenen modül tarafından oluşturulan çekirdek log mesajlarını görmek için **dmesg** komutunu kullanabiliriz. Örneğin, son 10 satırı bir dosyaya kaydetmek için şu komut kullanılır:

```
1 sudo dmesg | tail -10 > out.txt
```

Listing 8: dmesg çıktısının son 10 satırı out.txt'ye kaydedilir.

Örnek bir **out.txt** dosyası:

```
1 [35753.316505] wlo1: authenticated
2 [35753.319904] wlo1: associate with 54:46:17:d6:fd:b5 (try 1/3)
3 [35753.323354] wlo1: RX AssocResp from 54:46:17:d6:fd:b5 (capab=0x1c31
  status=0 aid=1)
4 [35753.338149] wlo1: associated
5 [35753.343007] wlo1: Limiting TX power to 27 (27 - 0) dBm as advertised by
  54:46:17:d6:fd:b5
6 [35949.438458] ACPI Error: No handler for Region [VRTC] (0000000091bc31ab) [
  SystemCMOS] (20230628/evregion-131)
7 [35949.438464] ACPI Error: Region SystemCMOS (ID=5) has no handler
  (20230628/exfldio-261)
8 [35949.438469] ACPI Error: Aborting method \_SB.PCI0.LPCB.EC._Q9A due to
  previous error (AE_NOT_EXIST) (20230628/psparse-529)
9 [37263.949530] Hello, world
10 [37283.307497] Goodbye, world
```

Son iki satırda da görüldüğü üzere modül yükleme ve çıkartma işlemleri başarıyla gerçekleşmiş.