



Линукс курс 2016/2017



Упознавање са Линукс кернелом и превођење кернела

Циљ

Научити како да се преузме изворни код кернела и да се примене закрпе. Упознавање са изворним кодом. Упознавање са конфигурисањем и превођењем кернела. Покретање кернела ће бити омогућено на Raspberry Pi платформи. На њему ће бити покренут минималистички кернел.

Исход

Након ове вежбе ћете моћи да:

- Преузмете изворни код са званичне локације
- Примените закрпе за кернел
- Истражите изворни код, прегледате датотеке, претражите заглавља или друге информације
- Конфигуришете, преведете и покренете кернел на Raspberry Pi платформи

ПРЕУЗМИТЕ ИЗВОРНИ КОД

Позиционирајте се у директоријум `~/linux-kernel-labs/src`.

Са `github` репозиторијума преузмите код помоћу одговарајуће `git` команде:

```
git clone --depth=1 https://github.com/raspberrypi/linux
```

Позиционирајте се у директоријум `~/linux-kernel-labs/src/linux` и затим направите нову грану `dan04` и позиционирајте се у исту:

```
git checkout -b dan04
```



Линукс курс 2016/2017



За каснији рад (комитовање измена) на GIT-у, конфигуришите своје податке:

```
git config --global user.email "ime.prezime@mail"
```

```
git config --global user.name "Ime Prezime"
```

ПРИМЕНИТЕ ЗАКРПЕ

Инсталирајте `patch` команду или преко графичке спреге или преко командне линије користећи команду:

```
sudo apt-get install patch
```

Примените `~/linux-kernel-labs/modules/data/Linuks.patch` на преузети код.

Прегледајте закрпу са `vi` или `gvim` да разумете информације које се преносе у таквој датотеци. Како су описане додате или уклоњене датотеке?

Преименујте преузети кернел директоријум да означите новију радну верзију.

УПОЗНАЈТЕ СЕ СА КОДОМ

Као корисник кернела Линукса, често ћете морати да нађете која датотека имплементира дату функцију. Стога, корисно је упознати се са изворним кодом кернела прегледањем датотека.

1. Пронађите слику Линукс логоа у изворном коду.
2. Пронађите ко одржава 3C505 мрежни драјвер.
3. Пронађите декларацију `platform_device_register()` функције.

Поред ручне претраге кода постоје и алати који нам у томе помажу. Испробајте претрагу помоћу LXR (Linux Cross Reference) на адреси <http://lxr.free-electrons.com> и изаберите верзију Линукса најближу оној коју користите.



Линукс курс 2016/2017



Такође, користан алат у ситуацијама када приступ интернету није омогућен је `cscope`. Коришћењем `cscope` алата и `LXR` пронађите претходно ручно пронађене датотеке и информације и уочите предност коришћења алата за индексирање. Да би се извршило индексирање целог Линукс кода са `cscope` алатом потребно је покренути команду `cscope -Rk`. Додатне информације о коришћењу овог алата пронаћи покретањем команде `cscope --help`. Уколико је потребно претходно инсталирајте `cscope` са командом `sudo apt-get install cscope`.

ИНСТАЛАЦИЈА АЛАТА ЗА ПРЕВОЂЕЊЕ

Пре него што пређемо на конфигуравање и превођење Линукс језгра потребно је инсталирати скуп алата за превођење. За превођење Линукса за RPI плочу користимо `gcc-arm-linux-gnueabi` преводилац. Инсталирајте поменути преводилац са командом:

```
sudo apt-get install gcc-arm-linux-gnueabi
```

КОНФИГУРАЦИЈА КЕРНЕЛА

Пошто кернел преводимо за RPI плочу која подржава ARM скуп инструкција, а само превођење се обавља на x86 архитектури, најпре је потребно подесити одговарајуће окружење за превођење. Минимално што је потребно да се уради је да се подесе циљна архитектура и префикс алата за превођење који ће се користити. Ово се може урадити на више начина од којих сваки има своје предности и мане.

Први начин је директна промена вредности `ARCH` и `CROSS_COMPILE` варијабли у `Makefile` датотеци. Овај приступ је погодан уколико кернел увек преводимо за исту архитектуру и са истим преводиоцем. Ипак, како то није чест случај, овај начин подешавања окружења за превођење се не препоручује.

Други начин, који је и најчешћи је подешавање варијабли окружења `ARCH` и `CROSS_COMPILE` помоћу команде `export`. Уколико желимо да подесимо окружење за превођење за RPI плочу на овај начин потребно је да извршимо следеће команде:



Линукс курс 2016/2017



```
export ARCH=arm
```

```
export CROSS_COMPILE=arm-linux-gnueabihf-
```

Мана овог приступа је у томе што су подешене варијабле видљиве само у терминалу у ком су извршене претходно наведене команде.

Трећи начин подешавања окружења за превођење је да се вредности варијабли ARCH и CROSS_COMPILE проследи при сваком позиву make команде, на пример:

```
make ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- -j2
```

При коришћењу овог приступа постоји велика шанса да се у неком од позива make команде изоставе потребне варијабле што резултује грешком у превођењу.

Након што сте подесили окружење за превођење за RPI плочу можете да приступите конфигурисању кернела.

Позиционирајте се у преузети и patch-овани кернел директоријум. Примените подразумевану конфигурацију која је додата у изворни код кернела применом закрпе. Искористите у те сврхе наредну команду која ће бити објашњена на наредним предавањима и вежбама:

```
make bcm2709_rtrk_linux_kurs_defconfig
```

Покрените команду xconfig да покренете спрегу за конфигурацију кернела. Спрега за конфигурацију кернела је дата као изворни код у самом кернелу и команда make xconfig је преводи аутоматски, али су неопходна заглавља и библиотеке. Можда ће вам бити потребно да инсталирате додатне пакете, попут libqt4-dev који садржи Qt развојне датотеке, као и g++ пакет који представља C++ преводилац.

У покренутој спрези за конфигурацију, укључите Options --- Show Name опцију. Ово је понекад корисно, када је име параметра експлицитније од описа или када пратимо нека упутства која су нам задата са именима параметара.

Такође, пробајте и опције Options --- Show All Options и Options -- Show Debug Info. Ове опције ће вам дозволити да видите све параметре



Линукс курс 2016/2017



који се иначе не би приказали, јер зависе од вредности других параметара. Тако што кликнете на опис неког од тих параметара, видећете његове предуслове за укључивање и разумећете зашто није могуће да се одаберу.

Специфицирајте суфикс верзију; тако ћете моћи да разликујете свој кернел командом `uname -r` или `cat /proc/version`, кад покренете систем.

Конфигуришите мали кернел за RPI плочу:

- Broadcom BCM2709 Development platform (ARCH_BCM2709)
- Подршка за MMC
- Ext4 и vfat системе датотека
- Подршка за elf binaries (BINFMT_ELF)
- Подршка за NEON у кернел моду
- MMC DMA подршка за BCM2835 Arasan контролер
- Подршка за SDHCI платформу за MMC
- SDHOST подршка за BCM2835
- Укључите ARM errata 643719
- Укључите подршку за ARM Architecture timer
- Искључите подршку за L2X0 cache
- Поставите LOCALVERSION на вашу верзију

Одвојите време да прегледате нове одлике које укључујете!

Такође, можете пробати `make menuconfig`. Иако није графичка спрега, неки људи преферирају овај начин конфигурације. Пошто је `menuconfig` спрега базирана на `Ncurses` `biblioteci`, мораћете да инсталирате `libncurses-dev` пакет да бисте је користили.

ПРЕВЕДИТЕ КЕРНЕЛ



Линукс курс 2016/2017



Потребно је само да покренете:

```
make
```

Добијени кернел је потребно покренути на RPI плочи. Обзиром да у кернел није укључена подршка за NFS, U-Boot треба подесити тако да се rfs преузима са картице. Покрените из U-Boot-а нови кернел тако што ћете привремено (без позива `saveenv`) променити променљиву `bootargs`, а потом покренути кернел:

```
setenv bootargs root=/dev/mmcblk0p2 rootwait console=tty1  
console=ttyAMA0,115200
```

```
boot
```

САЧУВАЈТЕ СВЕ ИЗМЕНЕ

Да бисте потврдили и сачували све измене, најбоље је да их додате, а потом и локално комитујете на GIT, док сте позиционирани у неки од директоријума репозиторијума који је мењан, нпр. `~/linux-kernel-labs/src/linux`:

```
git add -A
```

```
git commit -as -m "dan04 završen"
```

Да би измене постале видљиве и у репозиторијуму на серверу, потребно би још било урадити нпр. `git push`, али то у овом случају није неопходно нити имамо неопходна права за то.