

Szymon Werema

1. Metoda stara

1.1. Pobieranie oraz rozpakowanie kernela

```
root@slack:~# cd /usr/src/
root@slack:/usr/src# wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.18.2.tar.xz
--2022-06-06 17:59:02-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.18.2.tar.xz
Translacja cdn.kernel.org (cdn.kernel.org)... 151.101.113.176, 2a04:4e42:1b::432
Łączenie się z cdn.kernel.org (cdn.kernel.org)[151.101.113.176]:443... połączono
Żądanie HTTP wysłano, oczekiwanie na odpowiedź... 200 OK
Długość: 129796020 (124M) [application/x-xz]
Zapis do: `linux-5.18.2.tar.xz'

linux-5.18.2.tar.xz          100%[=====>] 123,78M  21,5MB/s   w 6,6s

2022-06-06 17:59:09 (18,6 MB/s) - zapisano `linux-5.18.2.tar.xz' [129796020/129796020]

root@slack:/usr/src# ls
linux-5.18.2.tar.xz
root@slack:/usr/src#
```

```
root@slack:/usr/src# tar -xf linux-5.18.2.tar.xz
root@slack:/usr/src# ls
linux-5.18.2/  linux-5.18.2.tar.xz
root@slack:/usr/src#
```

1.2. Przejście, skopiowanie aktualnej konfiguracji jądra oraz utworzenie pliku z modułami, które obecnie używamy

```
root@slack:/usr/src/linux-5.18.2# zcat /proc/config.gz > .config
root@slack:/usr/src/linux-5.18.2# make localmodconfig
HOSTCC scripts/kconfig/conf.o
HOSTCC scripts/kconfig/confdata.o
HOSTCC scripts/kconfig/expr.o
LEX scripts/kconfig/lexer.lex.c
YACC scripts/kconfig/parser.tab.[ch]
HOSTCC scripts/kconfig/lexer.lex.o
HOSTCC scripts/kconfig/menu.o
HOSTCC scripts/kconfig/parser.tab.o
HOSTCC scripts/kconfig/preprocess.o
HOSTCC scripts/kconfig/symbol.o
HOSTCC scripts/kconfig/util.o
HOSTLD scripts/kconfig/conf
using config: '.config'
*
* Restart config...
*
*
* Timers subsystem
```

1.3. Sprawdzenie konfiguracji kernela

```
root@slack:/usr/src/linux-5.18.2# make menuconfig
UPD scripts/kconfig/mconf-cfg
HOSTCC scripts/kconfig/mconf.o
HOSTCC scripts/kconfig/lxdialog/checklist.o
HOSTCC scripts/kconfig/lxdialog/inputbox.o
HOSTCC scripts/kconfig/lxdialog/menubox.o
HOSTCC scripts/kconfig/lxdialog/textbox.o
HOSTCC scripts/kconfig/lxdialog/util.o
HOSTCC scripts/kconfig/lxdialog/yesno.o
HOSTLD scripts/kconfig/mconf

*** End of the configuration.
*** Execute 'make' to start the build or try 'make help'.

root@slack:/usr/src/linux-5.18.2#
```

1.4. Zbudowanie pliku konfiguracyjnego kernela z konfiguracją z poprzedniego kernela

```
root@slack:/usr/src/linux-5.18.2# make olddefconfig
#
# No change to .config
#
root@slack:/usr/src/linux-5.18.2#
```

1.5. Kompilacja obrazu jądra

```
LD [M] arch/x86/boot/compressed/vmlinux.bin.lzma
LZMA arch/x86/boot/compressed/vmlinux.bin.lzma
MKPIGGY arch/x86/boot/compressed/piggy.S
AS arch/x86/boot/compressed/piggy.o
LD arch/x86/boot/compressed/vmlinux
ZOFFSET arch/x86/boot/zoffset.h
OBJCOPY arch/x86/boot/vmlinux.bin
AS arch/x86/boot/header.o
LD arch/x86/boot/setup.elf
OBJCOPY arch/x86/boot/setup.bin
BUILD arch/x86/boot/bzImage
Kernel: arch/x86/boot/bzImage is ready (#1)
root@slack:/usr/src/linux-5.18.2#
```

1.6. Zbudowanie modułów

```
LD [M] net/802/mrp.ko
LD [M] net/802/p8022.ko
LD [M] net/802/psnap.ko
LD [M] net/8021q/8021q.ko
LD [M] net/802/stp.ko
LD [M] net/llc/llc.ko
LD [M] net/ipv6/ipv6.ko
LD [M] net/rfkill/rfkill.ko
LD [M] net/wireless/cfg80211.ko
LD [M] sound/ac97_bus.ko
LD [M] sound/core/snd-pcm.ko
LD [M] sound/core/snd-timer.ko
LD [M] sound/core/snd.ko
LD [M] sound/pci/ac97/snd-ac97-codec.ko
LD [M] sound/pci/snd-intel8x0.ko
LD [M] sound/soundcore.ko
root@slack:/usr/src/linux-5.18.2# make -j8 modules
```

1.7. Instalacja modułów

```
INSTALL /lib/modules/5.18.2-smp/kernel/net/802/p8022.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/802/psnap.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/802/stp.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/8021q/8021q.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/ipv6/ipv6.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/llc/llc.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/rfkill/rfkill.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/wireless/cfg80211.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/ac97_bus.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/core/snd-pcm.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/core/snd-timer.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/core/snd.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/pci/ac97/snd-ac97-codec.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/pci/snd-intel8x0.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/soundcore.ko
DEPMOD /lib/modules/5.18.2-smp
root@slack:/usr/src/linux-5.18.2# sudo make modules_install
```

1.8. Kopiowanie plików kernela do systemu oraz utworzenie linku symbolicznego w systemie dla tablicy symboli kernela

```
root@slack:/usr/src/linux-5.18.2# cp arch/x86/boot/bzImage /boot/vmlinuz-custom-5.18.2-smp
root@slack:/usr/src/linux-5.18.2# cp System.map /boot/System.map-custom-5.18.2-smp
root@slack:/usr/src/linux-5.18.2# cp .config /boot/config-custom-5.18.2-smp
root@slack:/usr/src/linux-5.18.2# rm /boot/System.map
root@slack:/usr/src/linux-5.18.2# ln -s /boot/System.map-custom-5.18.2-smp /boot/System.map
root@slack:/usr/src/linux-5.18.2#
```

1.9. Utworzenie dysku ram

```
root@slack:/usr/src/linux-5.18.2# /usr/share/mkinitrd/mkinitrd_command_generator.sh -k 5.18.2-smp
#
# mkinitrd_command_generator.sh revision 1.45
#
# This script will now make a recommendation about the command to use
# in case you require an initrd image to boot a kernel that does not
# have support for your storage or root filesystem built in
# (such as the Slackware 'generic' kernels').
# A suitable 'mkinitrd' command will be:
mkinitrd -c -k 5.18.2-smp -f ext4 -r /dev/sda1 -m ext4 -u -o /boot/initrd.gz
root@slack:/usr/src/linux-5.18.2# mkinitrd -c -k 5.18.2-smp -f ext4 -r /dev/sda1 -m ext4 -u -o /boot
/initrd-custom-5.18.2-smp.gz
49039 bloków
/boot/initrd-custom-5.18.2-smp.gz created.
Be sure to run lilo again if you use it.
root@slack:/usr/src/linux-5.18.2#
```

1.10. Konfiguracja lilo

```
GNU nano 6.0 /etc/lilo.conf Zmieniony
#vga=771
# VESA framebuffer console @ 640x480x64k
#vga=785
# VESA framebuffer console @ 640x480x32k
#vga=784
# VESA framebuffer console @ 640x480x256
#vga=769
# End LILO global section
# Linux bootable partition config begins
image = /boot/vmlinuz
  root = /dev/sda1
  label = "Slackware 15.0"
  read-only

image = /boot/vmlinuz-custom-5.18.2-smp
  root = /dev/sda1
  initrd = /boot/initrd-custom-5.18.2-smp.gz
  label = "kernel-custom"
  read-only
# Linux bootable partition config ends
^G Pomoc      ^O Zapisz      ^W Wyszukaj    ^K Wytnij     ^T Wykonaj    ^C LokalizacjaM-U Odwołaj
^X Wyjdź      ^R Wczyt.plik ^_ Zastąp     ^U Wklej      ^J Wyjustuj   ^/ Do linii   M-E Odtwórz
```

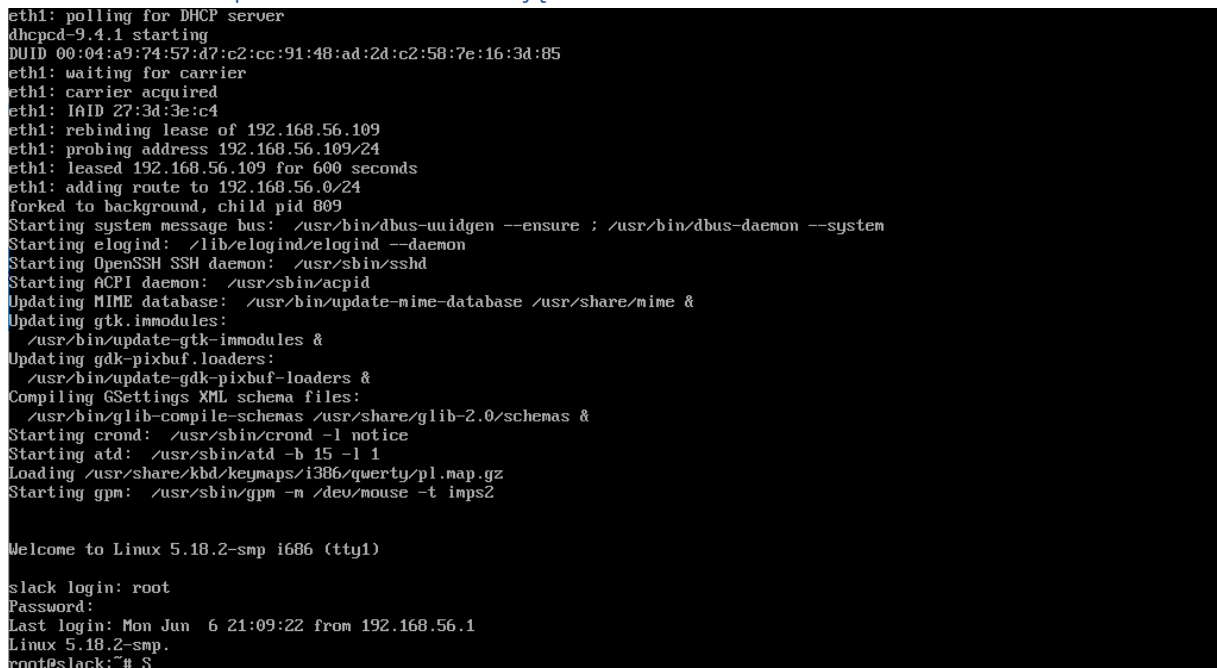
1.11. Sprawdzenie czy nowy kernel jest widziany przez lilo

```
root@slack:/usr/src/linux-5.18.2# lilo
Warning: LBA32 addressing assumed
Warning: Unable to determine video adapter in use in the present system.
Warning: Video adapter does not support VESA BIOS extensions needed for
display of 256 colors. Boot loader will fall back to TEXT only operation.
Added Slackware_15.0 *
Added kernel-custom +
3 warnings were issued.
root@slack:/usr/src/linux-5.18.2#
```

1.12. Restart oraz uruchomienie dodanego kernela



1.13. Poprawnie zbudowane jądro



2. Nowa metoda

2.1. Pobranie kernela

```
root@slack:/usr/src# wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.18.2.tar.xz
--2022-06-07 21:55:07-- https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.18.2.tar.xz
Translacja cdn.kernel.org (cdn.kernel.org)... 151.101.113.176, 2a04:4e42:3::432
Łączenie się z cdn.kernel.org (cdn.kernel.org)[151.101.113.176]:443... połączono
.
Żądanie HTTP wysłano, oczekiwanie na odpowiedź... 200 OK
Długość: 129796020 (124M) [application/x-xz]
Zapis do: `linux-5.18.2.tar.xz'

linux-5.18.2.tar.xz 100%[=====>] 123,78M  21,8MB/s   w 5,4s

2022-06-07 21:55:13 (22,7 MB/s) - zapisano `linux-5.18.2.tar.xz' [129796020/129796020]

root@slack:/usr/src# ls
linux-5.18.2.tar.xz
root@slack:/usr/src#
```

2.2. Rozpakowanie kernela

```
root@slack:/usr/src# tar -xf linux-5.18.2.tar.xz
root@slack:/usr/src# ls
linux-5.18.2/  linux-5.18.2.tar.xz
root@slack:/usr/src#
```

2.3. Stworzenie pliku .config za pomocą nowej metody

```
root@slack:/usr/src/linux-5.18.2# ./scripts/kconfig/streamline_config.pl > config_strip
using config: '/proc/config.gz'
root@slack:/usr/src/linux-5.18.2# mv config_strip .config
root@slack:/usr/src/linux-5.18.2#
```

2.4. Budowanie konfiguracji kernela

```
root@slack:/usr/src/linux-5.18.2# ./scripts/kconfig/streamline_config.pl > config_strip
using config: '/proc/config.gz'
root@slack:/usr/src/linux-5.18.2# mv config_strip .config
root@slack:/usr/src/linux-5.18.2# make oldconfig
  HOSTCC  scripts/basic/fixdep
  HOSTCC  scripts/kconfig/conf.o
  HOSTCC  scripts/kconfig/confdata.o
  HOSTCC  scripts/kconfig/expr.o
  LEX      scripts/kconfig/lexer.lex.c
  YACC     scripts/kconfig/parser.tab.[ch]
  HOSTCC  scripts/kconfig/lexer.lex.o
  HOSTCC  scripts/kconfig/menu.o
  HOSTCC  scripts/kconfig/parser.tab.o
  HOSTCC  scripts/kconfig/preprocess.o
  HOSTCC  scripts/kconfig/symbol.o
  HOSTCC  scripts/kconfig/util.o
  HOSTLD  scripts/kconfig/conf
*
* Restart config...
*
```

2.5. Kompilacja obrazu jądra

```
LZMA      arch/x86/boot/compressed/vmlinux.bin.lzma
CC        arch/x86/boot/compressed/misc.o
MKPIGGY   arch/x86/boot/compressed/piggy.S
AS        arch/x86/boot/compressed/piggy.o
LD        arch/x86/boot/compressed/vmlinux
ZOFFSET   arch/x86/boot/zoffset.h
OBJCOPY   arch/x86/boot/vmlinux.bin
AS        arch/x86/boot/header.o
LD        arch/x86/boot/setup.elf
OBJCOPY   arch/x86/boot/setup.bin
BUILD     arch/x86/boot/bzImage
Kernel: arch/x86/boot/bzImage is ready (#1)
```

```
real      5m53,216s
user      33m29,752s
sys       9m10,719s
root@slack:/usr/src/linux-kompilacja-jadra/metoda-nowa#
```

2.6. Kompilacja modułów

```
LD [M]    sound/pci/snd-intel8x0.ko
LD [M]    sound/soundcore.ko

real      0m34,458s
user      3m13,756s
sys       0m48,788s
root@slack:/usr/src/linux-5.18.2# time make -j8 modules
```

2.7. Instalacja modułów

```
INSTALL /lib/modules/5.18.2-smp/kernel/net/802/stp.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/8021q/8021q.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/ipv6/ipv6.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/llc/llc.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/rfkill/rfkill.ko
INSTALL /lib/modules/5.18.2-smp/kernel/net/wireless/cfg80211.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/ac97_bus.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/core/snd-pcm.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/core/snd-timer.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/core/snd.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/pci/ac97/snd-ac97-codec.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/pci/snd-intel8x0.ko
INSTALL /lib/modules/5.18.2-smp/kernel/sound/soundcore.ko
DEPMOD /lib/modules/5.18.2-smp
root@slack:/usr/src/linux-5.18.2# make modules_install
```

2.8. Kopiowanie plików kernela do systemu oraz utworzenie linku symbolicznego w systemie dla tablicy symboli kernela

```
root@slack:/usr/src/linux-5.18.2# cp arch/x86/boot/bzImage /boot/vmlinuz-custom-5.18.2-smp
root@slack:/usr/src/linux-5.18.2# cp System.map /boot/System.map-custom-5.18.2-smp
root@slack:/usr/src/linux-5.18.2# rm /boot/System.map
root@slack:/usr/src/linux-5.18.2# ln -s /boot/System.map-custom-5.18.2-smp /boot/System.map
root@slack:/usr/src/linux-5.18.2#
```

2.9. Utworzenie dysku ram

```
root@slack:/usr/src/linux-5.18.2# /usr/share/mkinitrd/mkinitrd_command_generator.sh -k 5.18.2-smp
#
# mkinitrd_command_generator.sh revision 1.45
#
# This script will now make a recommendation about the command to use
# in case you require an initrd image to boot a kernel that does not
# have support for your storage or root filesystem built in
# (such as the Slackware 'generic' kernels').
# A suitable 'mkinitrd' command will be:
mkinitrd -c -k 5.18.2-smp -f ext4 -r /dev/sda1 -m ext4 -u -o /boot/initrd.gz
root@slack:/usr/src/linux-5.18.2# ^C
root@slack:/usr/src/linux-5.18.2# mkinitrd -c -k 5.18.2-smp -f ext4 -r /dev/sda1 -m ext4 -u -o /boot/initrd
-custom-5.18.2-smp.gz
49039 bloków
/boot/initrd-custom-5.18.2-smp.gz created.
Be sure to run lilo again if you use it.
root@slack:/usr/src/linux-5.18.2#
```

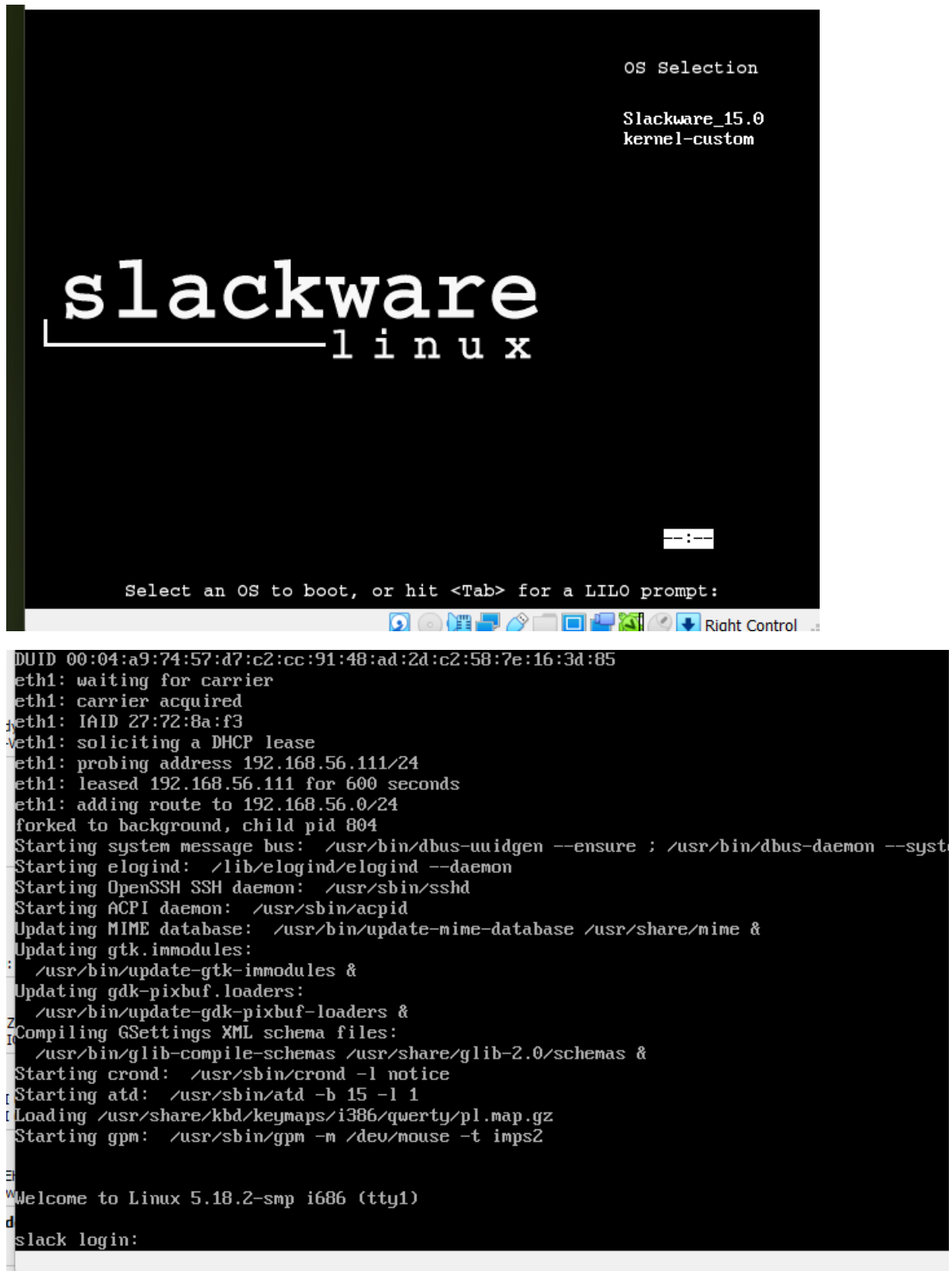
2.10. Konfiguracja lilo

```
GNU nano 6.0
#vga=771
# VESA framebuffer console @ 640x480x64k
#vga=785
# VESA framebuffer console @ 640x480x32k
#vga=784
# VESA framebuffer console @ 640x480x256
#vga=769
# End LILO global section
# Linux bootable partition config begins
image = /boot/vmlinuz
  root = /dev/sda1
  label = "Slackware 15.0"
  read-only
image = /boot/vmlinuz-custom-5.18.2-smp
  root = /dev/sda1
  initrd = /boot/initrd-custom-5.18.2-smp.gz
  label = "kernel-custom"
  read-only
# Linux bootable partition config ends
```

2.11. Sprawdzenie czy wpis dodany w lilo widzi obraz kernela

```
root@slack:/usr/src/linux-5.18.2# lilo
Warning: LBA32 addressing assumed
Warning: Unable to determine video adapter in use in the present system.
Warning: Video adapter does not support VESA BIOS extensions needed for
display of 256 colors. Boot loader will fall back to TEXT only operation.
Added Slackware 15.0 *
Added kernel-custom +
3 warnings were issued.
root@slack:/usr/src/linux-5.18.2#
```


2.12. Restart oraz uruchomienie nowego jądra



```
OS Selection

Slackware_15.0
kernel-custom

slackware
linux

Select an OS to boot, or hit <Tab> for a LILO prompt:

DUID 00:04:a9:74:57:d7:c2:cc:91:48:ad:2d:c2:58:7e:16:3d:85
eth1: waiting for carrier
eth1: carrier acquired
eth1: IAID 27:72:8a:f3
eth1: soliciting a DHCP lease
eth1: probing address 192.168.56.111/24
eth1: leased 192.168.56.111 for 600 seconds
eth1: adding route to 192.168.56.0/24
forked to background, child pid 804
Starting system message bus: /usr/bin/dbus-uuidgen --ensure ; /usr/bin/dbus-daemon --system
Starting elogind: /lib/elogind/elogind --daemon
Starting OpenSSH SSH daemon: /usr/sbin/sshd
Starting ACPI daemon: /usr/sbin/acpid
Updating MIME database: /usr/bin/update-mime-database /usr/share/mime &
Updating gtk.immodules:
/usr/bin/update-gtk-immodules &
Updating gdk-pixbuf.loaders:
/usr/bin/update-gdk-pixbuf-loaders &
Compiling GSettings XML schema files:
/usr/bin/glib-compile-schemas /usr/share/glib-2.0/schemas &
Starting crond: /usr/sbin/crond -l notice
Starting atd: /usr/sbin/atd -b 15 -l 1
Loading /usr/share/kbd/keymaps/i386/qwerty/pl.map.gz
Starting gpm: /usr/sbin/gpm -m /dev/mouse -t imps2

Welcome to Linux 5.18.2-smp i686 (tty1)

slack login:
```


3. Wnioski

3.1. Odczucia

Kompilacja w moim odczuciu przebiegała z równym poziomem trudności. Podczas niej nie napotkałem żadnych większych problemów. Jest to pewnie wynikiem bardzo dobrego opisu tego procesu na zajęciach.

3.2. Porównanie czasu kompilacji jądra oraz modułów

3.2.1. Stara metoda

```
Kernel: arch/x86/boot/bzImage is ready (#1)

real    1m10,355s
user    3m33,530s
sys     1m12,856s
root@slack:/usr/src/linux-5.18.2# time make -j8 bzImage

LD [M]  sound/pci/snd-intel8x0.ko

real    0m9,507s
user    0m32,399s
sys     0m15,042s
root@slack:/usr/src/linux-5.18.2# time make -j8 modules
```

3.2.2. Nowa metoda

```
BUILD   arch/x86/boot/bzImage
Kernel: arch/x86/boot/bzImage is ready (#1)

real    5m22,025s
user    31m55,483s
sys     7m31,935s
root@slack:/usr/src/linux-5.18.2#

LD [M]  sound/pci/snd-intel8x0.ko
LD [M]  sound/soundcore.ko

real    0m34,458s
user    3m13,756s
sys     0m48,788s
root@slack:/usr/src/linux-5.18.2# time make -j8 modules
```

3.2.3. Wnioski

Kompilacja zarówno jądra jak i modułów w nowej metodzie trwała dłużej. Zapewne różnica byłaby jeszcze większa gdyby użyć mniej wątków.

Kompilacja jądra w metodzie starej trwała około 5 razy krócej niż w metodzie nowej

Kompilacja modułów w metodzie starej trwała w 3 razy krócej niż w metodzie nowej

3.3. Czas uruchomienia

Porównywane wyniki za pomocą polecenia demsg

3.3.1. Stara metoda

System uruchamiał się średnio z w ciągu 21 sekund. Czas ten był dosyć niestabilny. System potrafił uruchomić się w 16-17 sekund, lecz czas ten potrafił się wydłużyć do 28 sekund.

3.3.2. Nowa metoda

System uruchamiał się średnio w ciągu 13 sekund. Czas ten był bardzo powtarzalny.

3.4. Bootowanie

Komunikaty wypisywane podczas bootowania obu metod przebiegały w identycznej kolejności.

3.5. Załadowane pakiety

Pakiety załadowane w obu metodach są identyczne.