



LAB 4

LINUX KERNEL DEVELOPMENT

Fullname: Nguyen Minh Nhut

Student ID: B2205896

- Note: screenshots need to be clear and good-looking; submissions must be in PDF format.

1. Modify kernel parameters and install new modules

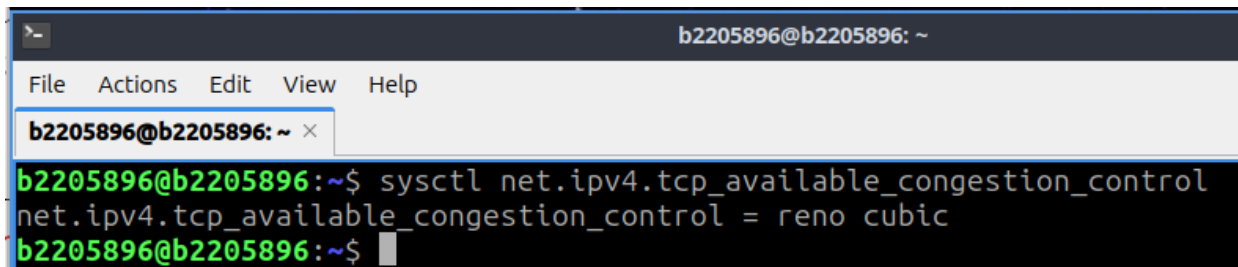
- List all linux kernel parameters on your OS:

```
sysctl -a
```

```
b2205896@b2205896: ~$ sysctl -a
abi.vsyscall32 = 1
debug.exception-trace = 1
debug.kprobes-optimization = 1
dev.cdrom.autoclose = 1
dev.cdrom.autoeject = 0
dev.cdrom.check_media = 0
dev.cdrom.debug = 0
dev.cdrom.info = CD-ROM information, Id: cdrom.c 3.20 2003/12/17
dev.cdrom.info =
dev.cdrom.info = drive name:          sr0
dev.cdrom.info = drive speed:          32
dev.cdrom.info = drive # of slots:       1
dev.cdrom.info = Can close tray:          1
dev.cdrom.info = Can open tray:          1
dev.cdrom.info = Can lock tray:          1
dev.cdrom.info = Can change speed:       1
dev.cdrom.info = Can select disk:         0
dev.cdrom.info = Can read multisession: 1
dev.cdrom.info = Can read MCN:           1
dev.cdrom.info = Reports media changed: 1
dev.cdrom.info = Can play audio:          1
dev.cdrom.info = Can write CD-R:          0
dev.cdrom.info = Can write CD-RW:         0
dev.cdrom.info = Can read DVD:           1
```

- List all available TCP congestion control algorithms:

```
sysctl net.ipv4.tcp_available_congestion_control
```



```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
b2205896@b2205896:~$ sysctl net.ipv4.tcp_available_congestion_control  
net.ipv4.tcp_available_congestion_control = reno cubic  
b2205896@b2205896:~$
```

- Show which TCP congestion control algorithm is using:

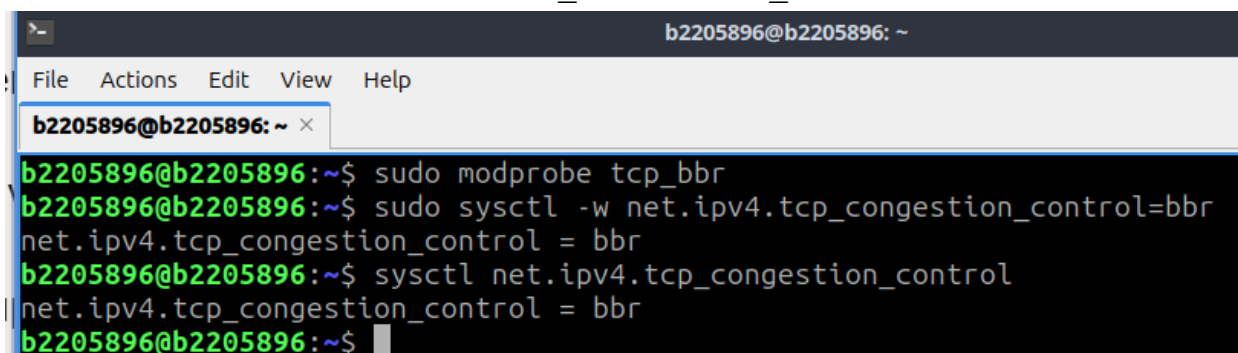
```
sysctl net.ipv4.tcp_congestion_control
```

- Install bbr TCP congestion control algorithm module:

```
sudo modprobe tcp_bbr
```

- Switch to the bbr TCP congestion control algorithm:

```
sudo sysctl -w net.ipv4.tcp_congestion_control=bbr  
sysctl net.ipv4.tcp_congestion_control
```



```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
b2205896@b2205896:~$ sudo modprobe tcp_bbr  
b2205896@b2205896:~$ sudo sysctl -w net.ipv4.tcp_congestion_control=bbr  
net.ipv4.tcp_congestion_control = bbr  
b2205896@b2205896:~$ sysctl net.ipv4.tcp_congestion_control  
net.ipv4.tcp_congestion_control = bbr  
b2205896@b2205896:~$
```

(take screenshots to show that you finish this exercise)

2. Install new kernel version

- Show your current kernel version:

```
uname -r
```

- Search for newer versions:

```
sudo apt search linux-image
```

```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
b2205896@b2205896:~$ uname -r  
6.8.0-31-generic  
b2205896@b2205896:~$ sudo apt search linux-image  
Sorting... Done  
Full Text Search... Done  
alsa-base/noble,noble,now 1.0.25+dfsg-0ubuntu7 all [installed,automatic]  
  ALSA driver configuration files  
  
linux-image-6.8.0-1003-gke/noble 6.8.0-1003.5 amd64  
  Signed kernel image gke  
  
linux-image-6.8.0-1004-gke/noble-updates,noble-security 6.8.0-1004.7 amd64  
  Signed kernel image gke  
  
linux-image-6.8.0-1005-gke/noble-updates,noble-security 6.8.0-1005.8 amd64  
  Signed kernel image gke  
  
linux-image-6.8.0-1005-ibm/noble 6.8.0-1005.5 amd64  
  Signed kernel image ibm  
  
linux-image-6.8.0-1005-intel/noble-updates,noble-security 6.8.0-1005.12 amd64  
  Signed kernel image intel  
  
linux-image-6.8.0-1005-oem/noble 6.8.0-1005.5 amd64
```

- Install the latest version you find:

```
sudo apt install linux-image-x.x.x-x-generic
```

```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
Signed kernel image generic  
  
linux-image-6.8.0-40-lowlatency/noble-updates,noble-security 6.8.0-40.40.1 amd64  
  Signed kernel image lowlatency  
  
linux-image-6.8.0-41-generic/noble-updates,noble-security 6.8.0-41.41 amd64  
  Signed kernel image generic  
  
linux-image-6.8.0-41-lowlatency/noble-updates,noble-security 6.8.0-41.41.1 amd64  
  Signed kernel image lowlatency  
  
linux-image-6.8.0-44-generic/noble-updates,noble-security 6.8.0-44.44 amd64  
  Signed kernel image generic  
  
linux-image-6.8.0-44-lowlatency/noble-updates,noble-security 6.8.0-44.44.1 amd64  
  Signed kernel image lowlatency  
  
linux-image-aws/noble-updates,noble-security 6.8.0-1015.16 amd64  
  Linux kernel image for Amazon Web Services (AWS) systems.
```

```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
b2205896@b2205896:~$ sudo apt install linux-image-6.8.0-44-generic  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  linux-libc-dev linux-modules-6.8.0-44-generic linux-tools-common  
Suggested packages:  
  fdutils linux-tools linux-headers-6.8.0-44-generic  
  linux-modules-extra-6.8.0-44-generic  
The following NEW packages will be installed:  
  linux-image-6.8.0-44-generic linux-modules-6.8.0-44-generic  
The following packages will be upgraded:  
  linux-libc-dev linux-tools-common  
2 upgraded, 2 newly installed, 0 to remove and 291 not upgraded.  
Need to get 0 B/56.0 MB of archives.  
After this operation, 54.0 MB of additional disk space will be used.  
Do you want to continue? [Y/n] y  
Selecting previously unselected package linux-modules-6.8.0-44-generic.  
(Reading database ... 280308 files and directories currently installed.)  
Preparing to unpack .../linux-modules-6.8.0-44-generic_6.8.0-44.44_amd64.deb ...  
Unpacking linux-modules-6.8.0-44-generic (6.8.0-44.44) ...  
Selecting previously unselected package linux-image-6.8.0-44-generic.  
Preparing to unpack .../linux-image-6.8.0-44-generic_6.8.0-44.44_amd64.deb ...  
Unpacking linux-image-6.8.0-44-generic (6.8.0-44.44) ...  
Preparing to unpack .../linux-libc-dev_6.8.0-44.44_amd64.deb ...  
Unpacking linux-libc-dev:amd64 (6.8.0-44.44) over (6.8.0-31.31) ...  
Preparing to unpack .../linux-tools-common_6.8.0-44.44_all.deb ...  
Unpacking linux-tools-common (6.8.0-44.44) over (6.8.0-31.31) ...  
Setting up linux-modules-6.8.0-44-generic (6.8.0-44.44) ...  
Progress: [ 53%] [#####.....]
```

- After a kernel upgrade, you must reboot the system. Then, if the device driver you need is in the latest kernel, your hardware will work as expected:

```
sudo shutdown -r now
```

- Show your new current kernel version:

```
uname -r
```

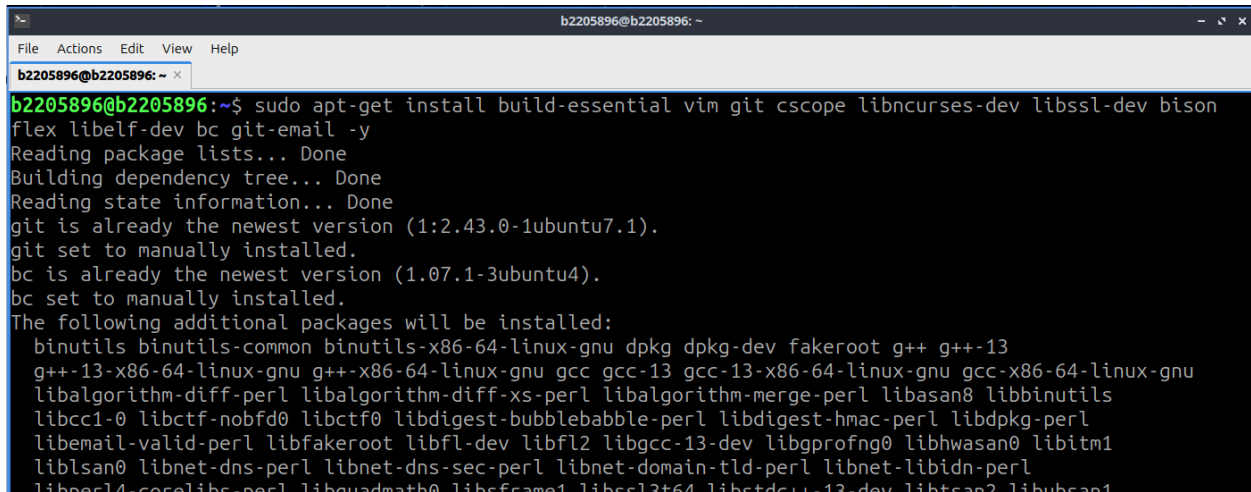
```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
b2205896@b2205896:~$ uname -r  
6.8.0-44-generic  
b2205896@b2205896:~$
```

(take screenshots to show that you finish this exercise)

3. Build and install a new kernel version

- Get your system ready

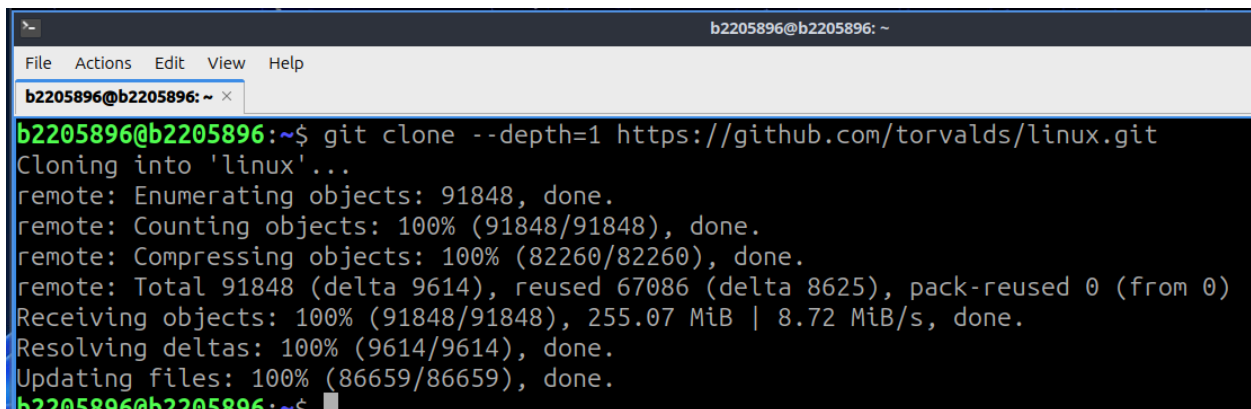
```
sudo apt update
sudo apt-get install build-essential vim git cscope
libncurses-dev libssl-dev bison flex libelf-dev bc
git-email -y
```



```
b2205896@b2205896: ~$ sudo apt-get install build-essential vim git cscope libncurses-dev libssl-dev bison
flex libelf-dev bc git-email -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.1).
git set to manually installed.
bc is already the newest version (1.07.1-3ubuntu4).
bc set to manually installed.
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu dpkg dpkg-dev fakeroot g++ g++-13
  g++-13-x86-64-linux-gnu g++-x86-64-linux-gnu gcc gcc-13 gcc-13-x86-64-linux-gnu gcc-x86-64-linux-gnu
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan8 libbinutils
  libbcc1-0 libctf-nobfd0 libctf0 libdigest-bubblebabble-perl libdigest-hmac-perl libdpkg-perl
  libemail-valid-perl libfakeroot libfl-dev libfl2 libgcc-13-dev libgprofng0 libhwasan0 libitm1
  liblsan0 libnet-dns-perl libnet-dns-sec-perl libnet-domain-tld-perl libnet-libidn-perl
  libperl4-corelibs-perl libquadmath0 libstdc++13 libstdc++13-dev libtsan2 libubsan1
```

- Clone a mainline kernel source code to your computer:

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



```
b2205896@b2205896: ~$ git clone --depth=1 https://github.com/torvalds/linux.git
Cloning into 'linux'...
remote: Enumerating objects: 91848, done.
remote: Counting objects: 100% (91848/91848), done.
remote: Compressing objects: 100% (82260/82260), done.
remote: Total 91848 (delta 9614), reused 67086 (delta 8625), pack-reused 0 (from 0)
Receiving objects: 100% (91848/91848), 255.07 MiB | 8.72 MiB/s, done.
Resolving deltas: 100% (9614/9614), done.
Updating files: 100% (86659/86659), done.
```

- To save time, just create a configuration file based on the list of modules currently loaded on your system (choose default values for other options).

```
lsmod > /tmp/my-lsmod
make LSMOD=/tmp/my-lsmod localmodconfig
```

```

b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~$ cd linux
b2205896@b2205896:~/linux$ lsmod > /tmp/my-lsmod
b2205896@b2205896:~/linux$ make LSMOD=/tmp/my-lsmod localmodconfig
using config: '.config'
module cpuid did not have configs CONFIG_X86_CPUID
module ip_tables did not have configs CONFIG_IP_NF_IPTABLES_LEGACY
module tcp_bbr did not have configs CONFIG_TCP_CONG_BBR

```

- Disable certificate stuff:

```

scripts/config --disable SYSTEM_TRUSTED_KEYS
scripts/config --disable SYSTEM_REVOCATION_KEYS

```

```

b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ scripts/config --disable SYSTEM_TRUSTED_KEYS
b2205896@b2205896:~/linux$ scripts/config --disable SYSTEM_REVOCATION_KEYS
b2205896@b2205896:~/linux$

```

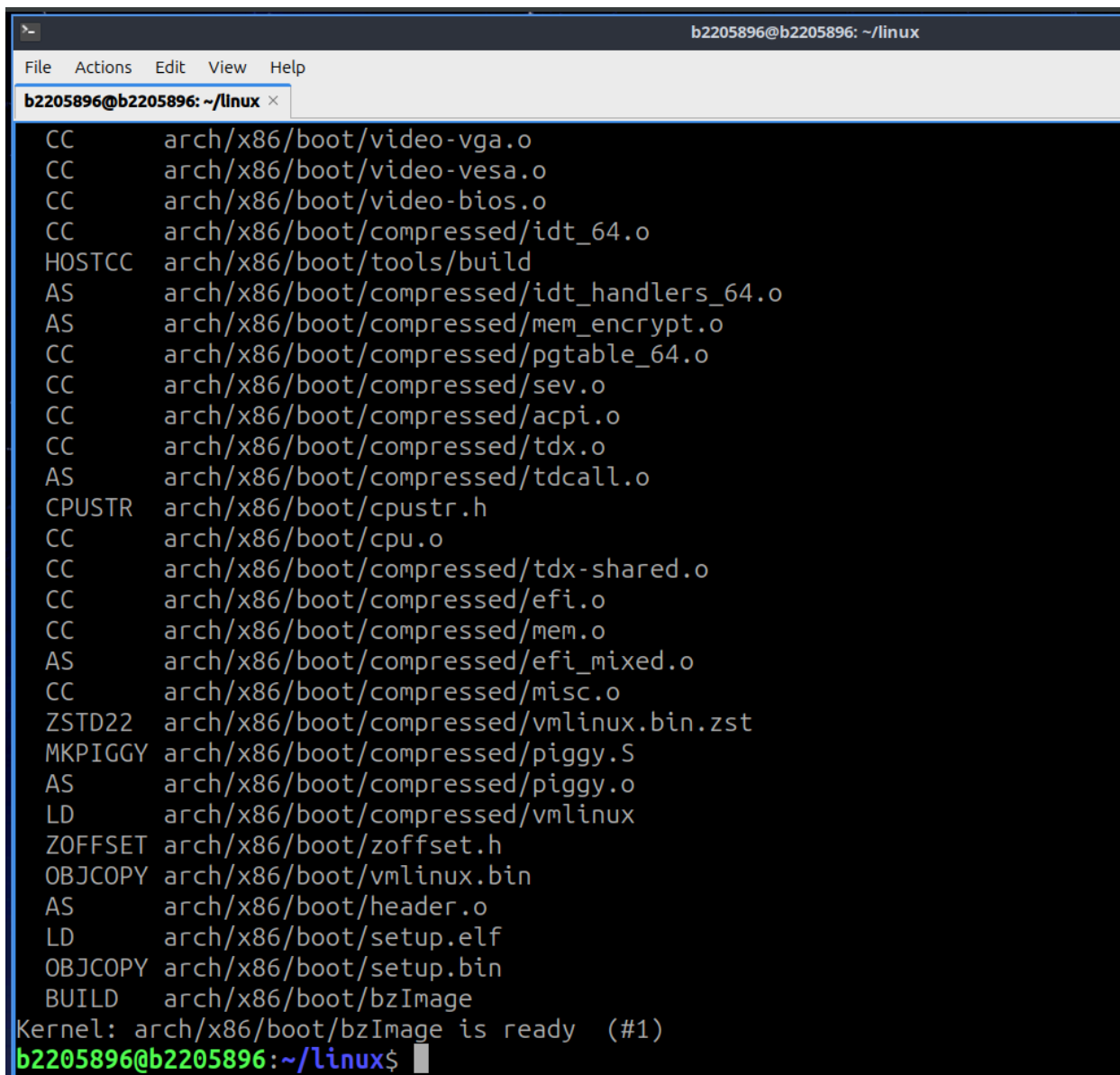
- Compile the kernel. The process takes about 1 hour, please be patient and enjoy a cup of coffee. It has been tested successfully on Ubuntu 20.04, if any errors occur, please try to fix them by yourself.

```
make -j3 all
```

```

b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ make -j4 all
SYNC include/config/auto.conf.cmd
*
* Restart config...
*
*
* Certificates for signature checking
*
File name or PKCS#11 URI of module signing key (MODULE_SIG_KEY) [certs/signing_key.pem] certs/sign
.pem
Type of module signing key to be generated
> 1. RSA (MODULE_SIG_KEY_TYPE_RSA)
2. ECDSA (MODULE_SIG_KEY_TYPE_ECDSA)
choice[1-2]: 1
Provide system-wide ring of trusted keys (SYSTEM_TRUSTED_KEYRING) [Y/?] y
Additional X.509 keys for default system keyring (SYSTEM_TRUSTED_KEYS) [] (NEW)
Reserve area for inserting a certificate without recompiling (SYSTEM_EXTRA_CERTIFICATE) [Y/n/?] y
Number of bytes to reserve for the extra certificate (SYSTEM_EXTRA_CERTIFICATE_SIZE) [4096] 409
Provide a keyring to which extra trustable keys may be added (SECONDARY_TRUSTED_KEYRING) [Y/n/?]
Only allow additional certs signed by keys on the builtin trusted keyring (SECONDARY_TRUSTED_K
SIGNED_BY_BUILTIN) [N/y/?] n
Provide system-wide ring of blacklisted keys (SYSTEM_BLACKLIST_KEYRING) [Y/n/?] y
Hashes to be preloaded into the system blacklist keyring (SYSTEM_BLACKLIST_HASH_LIST) []

```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
CC      arch/x86/boot/video-vga.o
CC      arch/x86/boot/video-vesa.o
CC      arch/x86/boot/video-bios.o
CC      arch/x86/boot/compressed/idt_64.o
HOSTCC  arch/x86/boot/tools/build
AS      arch/x86/boot/compressed/idt_handlers_64.o
AS      arch/x86/boot/compressed/mem_encrypt.o
CC      arch/x86/boot/compressed/pgtable_64.o
CC      arch/x86/boot/compressed/sev.o
CC      arch/x86/boot/compressed/acpi.o
CC      arch/x86/boot/compressed/tdx.o
AS      arch/x86/boot/compressed/tdcall.o
CPUTSTR arch/x86/boot/cpustr.h
CC      arch/x86/boot/cpu.o
CC      arch/x86/boot/compressed/tdx-shared.o
CC      arch/x86/boot/compressed/efi.o
CC      arch/x86/boot/compressed/mem.o
AS      arch/x86/boot/compressed/efi_mixed.o
CC      arch/x86/boot/compressed/misc.o
ZSTD22  arch/x86/boot/compressed/vmlinux.bin.zst
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)
b2205896@b2205896: ~/linux$
```

- Install the new kernel:

```
sudo make modules_install install
```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ sudo make modules_install install
[sudo] password for b2205896:
SYMLINK /lib/modules/6.11.0+/build
INSTALL /lib/modules/6.11.0+/modules.order
INSTALL /lib/modules/6.11.0+/modules.builtin
INSTALL /lib/modules/6.11.0+/modules.builtin.modinfo
INSTALL /lib/modules/6.11.0+/kernel/arch/x86/kernel/msr.ko
SIGN /lib/modules/6.11.0+/kernel/arch/x86/kernel/msr.ko
ZSTD /lib/modules/6.11.0+/kernel/arch/x86/kernel/msr.ko.zst
INSTALL /lib/modules/6.11.0+/kernel/arch/x86/crypto/aesni-intel.ko
SIGN /lib/modules/6.11.0+/kernel/arch/x86/crypto/aesni-intel.ko
ZSTD /lib/modules/6.11.0+/kernel/arch/x86/crypto/aesni-intel.ko.zst
INSTALL /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha1-ssse3.ko
SIGN /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha1-ssse3.ko
ZSTD /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha1-ssse3.ko.zst
INSTALL /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha256-ssse3.ko
SIGN /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha256-ssse3.ko
ZSTD /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha256-ssse3.ko.zst
INSTALL /lib/modules/6.11.0+/kernel/arch/x86/crypto/ghash-clmulni-intel.ko
SIGN /lib/modules/6.11.0+/kernel/arch/x86/crypto/ghash-clmulni-intel.ko

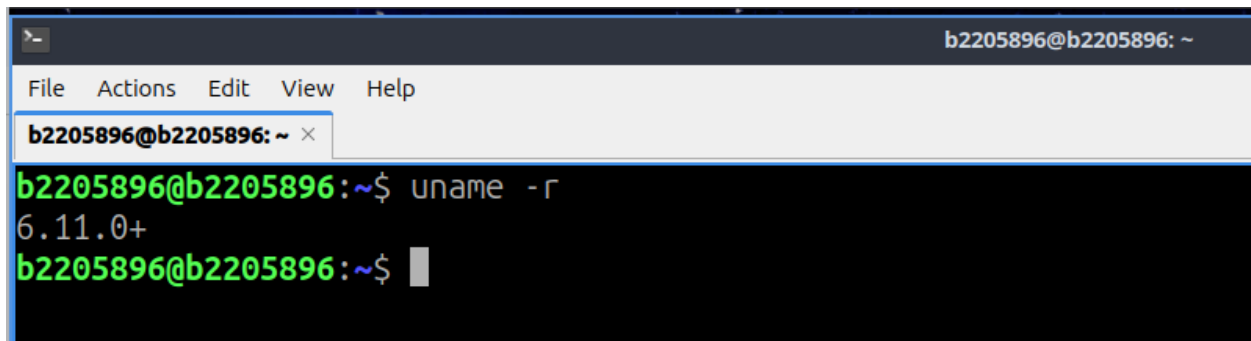
SIGN /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/ip_tables.ko
ZSTD /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/ip_tables.ko.zst
INSTALL /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/iptables_filter.ko
SIGN /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/iptables_filter.ko
ZSTD /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/iptables_filter.ko.zst
DEPMOD /lib/modules/6.11.0+
INSTALL /boot
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 6.11.0+ /boot/vmlinuz-6.11.0+
update-initramfs: Generating /boot/initrd.img-6.11.0+
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 6.11.0+ /boot/vmlinuz-6.11.0+
run-parts: executing /etc/kernel/postinst.d/xx-update-initrd-links 6.11.0+ /boot/vmlinuz-6.11.0+
I: /boot/initrd.img.old is now a symlink to initrd.img-6.8.0-45-generic
I: /boot/initrd.img is now a symlink to initrd.img-6.11.0+
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 6.11.0+ /boot/vmlinuz-6.11.0+
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/lubuntu-grub-theme.cfg'
Generating grub configuration file ...
Found theme: /usr/share/grub/themes/lubuntu-grub-theme/theme.txt
Found linux image: /boot/vmlinuz-6.11.0+
Found initrd image: /boot/initrd.img-6.11.0+
Found linux image: /boot/vmlinuz-6.8.0-45-generic
Found initrd image: /boot/initrd.img-6.8.0-45-generic
Found linux image: /boot/vmlinuz-6.8.0-44-generic
Found initrd image: /boot/initrd.img-6.8.0-44-generic
Found memtest86+x64 image: /boot/memtest86+x64.bin
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done
b2205896@b2205896:~/linux$
```


- Now it is time to reboot the system to boot the newly installed kernel:

```
sudo shutdown -r now
```

- Show your new current kernel version:

```
uname -r
```

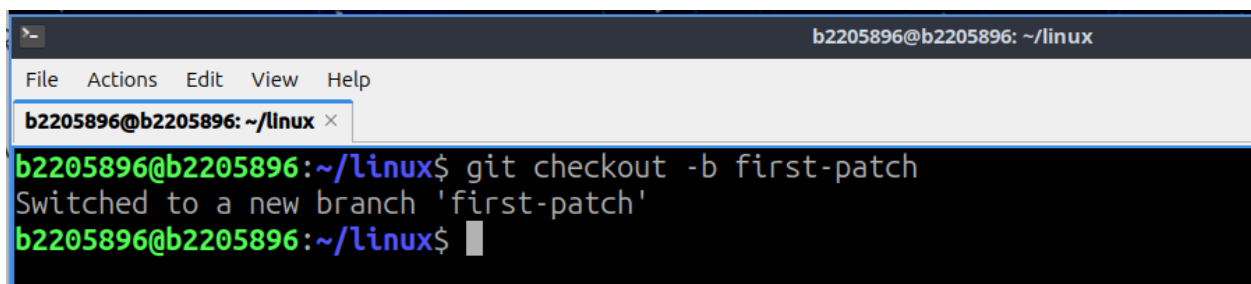
A terminal window with a dark background and light green text. The title bar shows 'b2205896@b2205896: ~'. The menu bar includes 'File', 'Actions', 'Edit', 'View', and 'Help'. The terminal shows the command 'uname -r' being executed, resulting in the output '6.11.0+'. The prompt is 'b2205896@b2205896:~\$'.

(take screenshots to show that you finish this exercise)

4. Writing Your First Kernel Patch

- Creating a new branch in the linux_mainline repository (has been cloned in exercise 3)

```
git checkout -b first-patch
```

A terminal window with a dark background and light green text. The title bar shows 'b2205896@b2205896: ~/linux'. The menu bar includes 'File', 'Actions', 'Edit', 'View', and 'Help'. The terminal shows the command 'git checkout -b first-patch' being executed, resulting in the output 'Switched to a new branch 'first-patch''. The prompt is 'b2205896@b2205896:~/linux\$'.

- Update the kernel

```
git fetch origin
```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ git fetch origin
remote: Enumerating objects: 10416409, done.
remote: Counting objects: 100% (10416395/10416395), done.
remote: Compressing objects: 100% (1803325/1803325), done.
remote: Total 10347633 (delta 8737769), reused 10098288 (delta 8489006), pack-reused 0 (from 0)
Receiving objects: 100% (10347633/10347633), 3.82 GiB | 10.80 MiB/s, done.
Resolving deltas: 100% (8737769/8737769), completed with 63285 local objects.
From https://github.com/torvalds/linux
 075dbe9f6e3c..e32cde8d2bd7 master -> origin/master
warning: it took 12.27 seconds to check forced updates; you can use
'--no-show-forced-updates' or run 'git config fetch.showForcedUpdates false'
to avoid this check

* [new tag] v2.6.12 -> v2.6.12
* [new tag] v2.6.12-rc2 -> v2.6.12-rc2
* [new tag] v2.6.12-rc3 -> v2.6.12-rc3
* [new tag] v2.6.12-rc4 -> v2.6.12-rc4
* [new tag] v2.6.12-rc5 -> v2.6.12-rc5
* [new tag] v2.6.12-rc6 -> v2.6.12-rc6
* [new tag] v2.6.13 -> v2.6.13
* [new tag] v2.6.13-rc1 -> v2.6.13-rc1
* [new tag] v2.6.13-rc2 -> v2.6.13-rc2
* [new tag] v2.6.13-rc3 -> v2.6.13-rc3
* [new tag] v2.6.13-rc4 -> v2.6.13-rc4
* [new tag] v2.6.13-rc5 -> v2.6.13-rc5
* [new tag] v2.6.13-rc6 -> v2.6.13-rc6
* [new tag] v2.6.13-rc7 -> v2.6.13-rc7
* [new tag] v2.6.14 -> v2.6.14
* [new tag] v2.6.14-rc1 -> v2.6.14-rc1
* [new tag] v2.6.14-rc2 -> v2.6.14-rc2
* [new tag] v2.6.14-rc3 -> v2.6.14-rc3

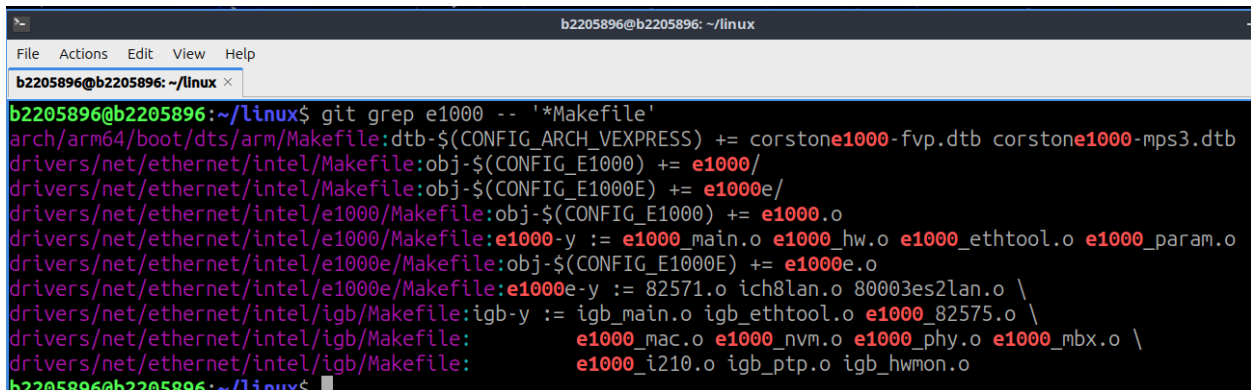
* [new tag] v6.7-rc5 -> v6.7-rc5
* [new tag] v6.7-rc6 -> v6.7-rc6
* [new tag] v6.7-rc7 -> v6.7-rc7
* [new tag] v6.7-rc8 -> v6.7-rc8
* [new tag] v6.8 -> v6.8
* [new tag] v6.8-rc1 -> v6.8-rc1
* [new tag] v6.8-rc2 -> v6.8-rc2
* [new tag] v6.8-rc3 -> v6.8-rc3
* [new tag] v6.8-rc4 -> v6.8-rc4
* [new tag] v6.8-rc5 -> v6.8-rc5
* [new tag] v6.8-rc6 -> v6.8-rc6
* [new tag] v6.8-rc7 -> v6.8-rc7
* [new tag] v6.9 -> v6.9
* [new tag] v6.9-rc1 -> v6.9-rc1
* [new tag] v6.9-rc2 -> v6.9-rc2
* [new tag] v6.9-rc3 -> v6.9-rc3
* [new tag] v6.9-rc4 -> v6.9-rc4
* [new tag] v6.9-rc5 -> v6.9-rc5
* [new tag] v6.9-rc6 -> v6.9-rc6
* [new tag] v6.9-rc7 -> v6.9-rc7
warning: it took 12.27 seconds to check forced updates; you can use
'--no-show-forced-updates' or run 'git config fetch.showForcedUpdates false'
to avoid this check

b2205896@b2205896:~/linux$
```

- Run `lsmod` to see the modules loaded on your system, and pick a driver to change. One driver that's included in all VM images is the `e1000` driver, the Intel ethernet driver, or you can choose another driver depending on your working environment.

- Run `git grep` to look for `e1000` files

```
git grep e1000 -- '*Makefile'
```



```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux
b2205896@b2205896:~/linux$ git grep e1000 -- '*Makefile'
arch/arm64/boot/dts/arm/Makefile:dtb-$(CONFIG_ARCH_VEXPRESS) += corstonee1000-fvp.dtb corstonee1000-mps3.dtb
drivers/net/ethernet/intel/Makefile:obj-$(CONFIG_E1000) += e1000/
drivers/net/ethernet/intel/Makefile:obj-$(CONFIG_E1000E) += e1000e/
drivers/net/ethernet/intel/e1000/Makefile:obj-$(CONFIG_E1000) += e1000.o
drivers/net/ethernet/intel/e1000/Makefile:e1000-y := e1000_main.o e1000_hw.o e1000_ethtool.o e1000_param.o
drivers/net/ethernet/intel/e1000e/Makefile:obj-$(CONFIG_E1000E) += e1000e.o
drivers/net/ethernet/intel/e1000e/Makefile:e1000e-y := 82571.o ich8lan.o 80003es2lan.o \
drivers/net/ethernet/intel/igb/Makefile:igb-y := igb_main.o igb_ethtool.o e1000_82575.o \
drivers/net/ethernet/intel/igb/Makefile:      e1000_mac.o e1000_nvmm.o e1000_phy.o e1000_mbx.o \
drivers/net/ethernet/intel/igb/Makefile:      e1000_i210.o igb_ptp.o igb_hwmmon.o
b2205896@b2205896:~/linux$
```

- Make a small change to the probe function of the `e1000` driver

```
nano drivers/net/ethernet/intel/e1000/e1000_main.c
```

```
# Add a line of code as below
```

```
static int e1000_probe(struct pci_dev *pdev, const
struct pci_device_id *ent) {
    ...
    struct e1000_hw *hw;
    printk(KERN_DEBUG "I can modify the Linux kernel!\n");
    static int cards_found = 0;
    ...
}
```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
GNU nano 7.2 drivers/net/ethernet/intel/e1000/e1000_main.c
* @ent: entry in e1000_pci_tbl
*
* Returns 0 on success, negative on failure
*
* e1000_probe initializes an adapter identified by a pci_dev structure.
* The OS initialization, configuring of the adapter private structure,
* and a hardware reset occur.
**/
static int e1000_probe(struct pci_dev *pdev, const struct pci_device_id *ent)
{
    struct net_device *netdev;
    struct e1000_adapter *adapter = NULL;
    struct e1000_hw *hw;
    printk(KERN_DEBUG "I can modify the Linux kernel!\n");
    static int cards_found;
    static int global_quad_port_a; /* global ksp3 port a indication */
    int i, err, pci_using_dac;
    u16 eeprom_data = 0;
    u16 tmp = 0;
```

- Compile and install your changes:

```
make -j3
```

```
sudo make modules_install install
```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ make -j4
mkdir -p /home/b2205896/linux/tools/objtool && make O=/home/b2205896/linux subdir=tools/objtool
ectory -C objtool
  CALL scripts/checksyscalls.sh
  INSTALL libsubcmd_headers
  CC [M] drivers/net/ethernet/intel/e1000/e1000_main.o
  LD [M] drivers/net/ethernet/intel/e1000/e1000.o
  MODPOST Module.symvers
Kernel: arch/x86/boot/bzImage is ready (#1)
  CC [M] drivers/net/ethernet/intel/e1000/e1000.mod.o
  LD [M] drivers/net/ethernet/intel/e1000/e1000.ko
b2205896@b2205896:~/linux$ sudo make modules_install install
[sudo] password for b2205896:
  INSTALL /lib/modules/6.11.0+/modules.order
  INSTALL /lib/modules/6.11.0+/modules.builtin
  INSTALL /lib/modules/6.11.0+/modules.builtin.modinfo
  SYMLINK /lib/modules/6.11.0+/build
  INSTALL /lib/modules/6.11.0+/kernel/arch/x86/kernel/msr.ko
  SIGN /lib/modules/6.11.0+/kernel/arch/x86/kernel/msr.ko
  ZSTD /lib/modules/6.11.0+/kernel/arch/x86/kernel/msr.ko.zst
  INSTALL /lib/modules/6.11.0+/kernel/arch/x86/crypto/aesni-intel.ko
  SIGN /lib/modules/6.11.0+/kernel/arch/x86/crypto/aesni-intel.ko
  ZSTD /lib/modules/6.11.0+/kernel/arch/x86/crypto/aesni-intel.ko.zst
  INSTALL /lib/modules/6.11.0+/kernel/arch/x86/crypto/sha1-ssse3.ko
```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
INSTALL /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/ip_tables.ko
SIGN /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/ip_tables.ko
ZSTD /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/ip_tables.ko.zst
INSTALL /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/iptables_filter.ko
SIGN /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/iptables_filter.ko
ZSTD /lib/modules/6.11.0+/kernel/net/ipv4/netfilter/iptables_filter.ko.zst
DEPMOD /lib/modules/6.11.0+
INSTALL /boot
run-parts: executing /etc/kernel/postinst.d/initramfs-tools 6.11.0+ /boot/vmlinuz-6.11.0+
update-initramfs: Generating /boot/initrd.img-6.11.0+
run-parts: executing /etc/kernel/postinst.d/unattended-upgrades 6.11.0+ /boot/vmlinuz-6.11.0+
run-parts: executing /etc/kernel/postinst.d/xx-update-initrd-links 6.11.0+ /boot/vmlinuz-6.11.0+
run-parts: executing /etc/kernel/postinst.d/zz-update-grub 6.11.0+ /boot/vmlinuz-6.11.0+
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/lubuntu-grub-theme.cfg'
Generating grub configuration file ...
Found theme: /usr/share/grub/themes/lubuntu-grub-theme/theme.txt
Found linux image: /boot/vmlinuz-6.11.0+
Found initrd image: /boot/initrd.img-6.11.0+
Found linux image: /boot/vmlinuz-6.11.0+.old
Found initrd image: /boot/initrd.img-6.11.0+
Found linux image: /boot/vmlinuz-6.8.0-45-generic
Found initrd image: /boot/initrd.img-6.8.0-45-generic
Found linux image: /boot/vmlinuz-6.8.0-44-generic
Found initrd image: /boot/initrd.img-6.8.0-44-generic
Found memtest86+x64 image: /boot/memtest86+x64.bin
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done
b2205896@b2205896:~/linux$
```

- Reboot the system:

```
sudo shutdown -r now
```

- Show kernel buffer log:

```
sudo dmesg | less
```

Search for your printk in the log file by typing "/I
can modify"

```

b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
[ 0.902049] ata3: SATA max UDMA/133 abar m8192@0xf0806000 port 0xf0806100 irq 21 lpm-pol 0
[ 0.907558] e1000: Intel(R) PRO/1000 Network Driver
[ 0.907561] e1000: Copyright (c) 1999-2006 Intel Corporation.
[ 0.907578] I can modify the Linux kernel!
[ 0.938509] usb 1-1: new full-speed USB device number 2 using ohci-pci
[ 1.158433] psmouse serio1: pixart_ps2: init: Unable to query PixArt touchpad hardware.
[ 1.198930] usb 1-1: New USB device found, idVendor=80ee, idProduct=0021, bcdDevice= 1.00
[ 1.198934] usb 1-1: New USB device strings: Mfr=1, Product=3, SerialNumber=0
[ 1.198936] usb 1-1: Product: USB Tablet
[ 1.198937] usb 1-1: Manufacturer: VirtualBox
[ 1.227275] hid: raw HID events driver (C) Jiri Kosina
[ 1.247916] ata3: SATA link up 3.0 Gbps (SStatus 123 SControl 300)
[ 1.248712] ata3.00: ATA-6: VBOX HARDDISK, 1.0, max UDMA/133
[ 1.248715] ata3.00: 104857600 sectors, multi 128: LBA48 NCQ (depth 32)
[ 1.248715] ata3.00: configured for UDMA/133

```

- Committing changes, and view your commit

```

git add .
git commit -s -v -m "My first kernel patch"
git show HEAD

```

```

b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ git add .
b2205896@b2205896:~/linux$ git commit -s -v -m "My first kernel patch"
[first-patch 1e8f2df8b] My first kernel patch
1 file changed, 1 insertion(+), 1 deletion(-)
b2205896@b2205896:~/linux$ git show HEAD
commit 1e8f2df8b75b7591851fffd59db49e17f5c60271 (HEAD -> first-patch)
Author: Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>
Date:   Fri Sep 27 09:30:14 2024 +0700

    My first kernel patch

    Signed-off-by: Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>

diff --git a/drivers/net/ethernet/intel/e1000/e1000_main.c b/drivers/net/ethernet/intel/e1000/e1000_main.c
index ab7ae418d..ef796edac 100644
--- a/drivers/net/ethernet/intel/e1000/e1000_main.c
+++ b/drivers/net/ethernet/intel/e1000/e1000_main.c
@@ -920,7 +920,7 @@ static int e1000_probe(struct pci_dev *pdev, const struct pci_device_id *ent)
     struct net_device *netdev;
     struct e1000_adapter *adapter = NULL;
     struct e1000_hw *hw;

+    printk(KERN_DEBUG "I can modify the Linux kernel!\n");
     static int cards_found;
     static int global_quad_port_a; /* global ksp3 port a indication */
     int i, err, pci_using_dac;
b2205896@b2205896:~/linux$

```

- Find whom to send the patch to

```
git show HEAD | scripts/get_maintainer.pl
```



```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ git show HEAD | scripts/get_maintainer.pl
Tony Nguyen <anthony.l.nguyen@intel.com> (supporter:INTEL ETHERNET DRIVERS)
Przemek Kitszel <przemyslaw.kitszel@intel.com> (supporter:INTEL ETHERNET DRIVERS)
"David S. Miller" <davem@davenloft.net> (maintainer:NETWORKING DRIVERS)
Eric Dumazet <edumazet@google.com> (maintainer:NETWORKING DRIVERS)
Jakub Kicinski <kuba@kernel.org> (maintainer:NETWORKING DRIVERS)
Paolo Abeni <pabeni@redhat.com> (maintainer:NETWORKING DRIVERS)
intel-wired-lan@lists.osuosl.org (moderated list:INTEL ETHERNET DRIVERS)
netdev@vger.kernel.org (open list:NETWORKING DRIVERS)
linux-kernel@vger.kernel.org (open list)
b2205896@b2205896:~/linux$
```

- Create a patch

```
git format-patch -1 <commit ID> --to=<your email>
```

Note: Please do not send your patch to a maintainer,
send it to yourself instead.

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ git format-patch -1 1e8f2df8b75b7591851fffd59db49e17f5c60271 --to=nmnhut.work@gmail.com
0001-My-first-kernel-patch.patch
b2205896@b2205896:~/linux$
```

- Modify ./git/config file to configure send-email

```
#.git/config
```

```
[sendemail]
```

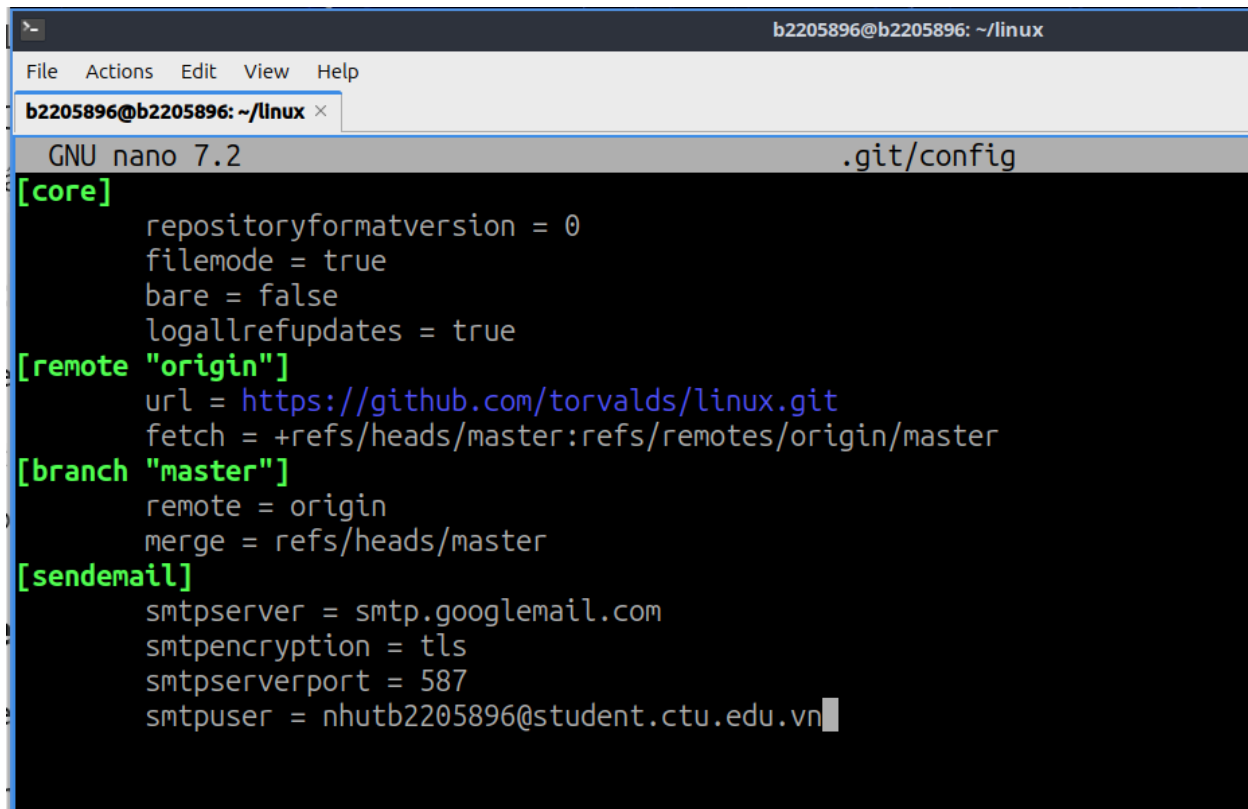
```
smtpserver = smtp.googlemail.com
```

```
smtpencryption = tls
```

```
smtpserverport = 587
```

```
smtpuser = your gmail address (CTU student email is
```

OK

A screenshot of a terminal window with a dark background. At the top, a status bar shows 'b2205896@b2205896: ~/linux'. Below it is a menu bar with 'File', 'Actions', 'Edit', 'View', and 'Help'. A tab labeled 'b2205896@b2205896: ~/linux' is open. The main area shows the 'GNU nano 7.2' editor editing the file '.git/config'. The file content is as follows:

```
[core]
  repositoryformatversion = 0
  filemode = true
  bare = false
  logallrefupdates = true
[remote "origin"]
  url = https://github.com/torvalds/linux.git
  fetch = +refs/heads/master:refs/remotes/origin/master
[branch "master"]
  remote = origin
  merge = refs/heads/master
[sendemail]
  smtpserver = smtp.googlemail.com
  smtpencryption = tls
  smtpserverport = 587
  smtpuser = nhutb2205896@student.ctu.edu.vn
```

- Send the patch

```
git send-email <patch_file>
```

```
b2205896@b2205896: ~/linux
File Actions Edit View Help
b2205896@b2205896: ~/linux x
b2205896@b2205896:~/linux$ git send-email 0001-My-first-kernel-patch.patch
0001-My-first-kernel-patch.patch
To whom should the emails be sent (if anyone)?
Message-ID to be used as In-Reply-To for the first email (if any)?
(mbox) Adding cc: Nhut Nguyen <nhutb2205896@student.ctu.edu.vn> from line 'From: Nhut Nguyen <nhutb2205896@student.ctu.edu.vn>'
Adding to: nmnhut.work@gmail.com from line 'To: nmnhut.work@gmail.com'
(body) Adding cc: Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn> from line 'Signed-off-by: Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>'

From: Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>
To: nmnhut.work@gmail.com
Cc: Nhut Nguyen <nhutb2205896@student.ctu.edu.vn>,
    Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>
Subject: [PATCH] My first kernel patch
Date: Fri, 27 Sep 2024 10:27:33 +0700
Message-ID: <20240927032736.3621-1-nhutb2205896@studnet.ctu.edu.vn>
X-Mailer: git-send-email 2.43.0
MIME-Version: 1.0
Content-Transfer-Encoding: 8bit

The Cc list above has been expanded by additional
addresses found in the patch commit message. By default
send-email prompts before sending whenever this occurs.
This behavior is controlled by the sendemail.confirm
configuration setting.

For additional information, run 'git send-email --help'.
To retain the current behavior, but squelch this message,
run 'git config --global sendemail.confirm auto'.

Send this email? ([y]es|[n]o|[e]dit|[q]uit|[a]ll): y

Send this email? ([y]es|[n]o|[e]dit|[q]uit|[a]ll): y
Password for 'smtp://nhutb2205896@student.ctu.edu.vn@smtp.googlemail.com:587':
OK. Log says:
Server: smtp.googlemail.com
MAIL FROM:<nhutb2205896@studnet.ctu.edu.vn>
RCPT TO:<nmnhut.work@gmail.com>
RCPT TO:<nhutb2205896@student.ctu.edu.vn>
RCPT TO:<nhutb2205896@studnet.ctu.edu.vn>
From: Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>
To: nmnhut.work@gmail.com
Cc: Nhut Nguyen <nhutb2205896@student.ctu.edu.vn>,
    Nhut Nguyen <nhutb2205896@studnet.ctu.edu.vn>
Subject: [PATCH] My first kernel patch
Date: Fri, 27 Sep 2024 10:27:33 +0700
Message-ID: <20240927032736.3621-1-nhutb2205896@studnet.ctu.edu.vn>
X-Mailer: git-send-email 2.43.0
MIME-Version: 1.0
Content-Transfer-Encoding: 8bit

Result: 250
b2205896@b2205896:~/linux$
```

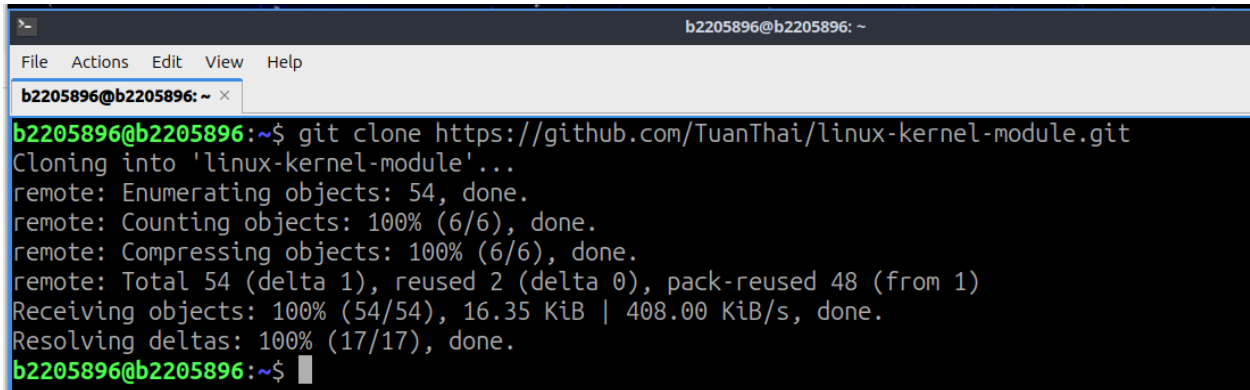
(take screenshots to show that you finish this exercise)

5. Writing a simple Linux kernel module: Greeter sample

This module simply takes a name as a parameter, and writes a greeting to the kernel log (/var/log/kern.log):

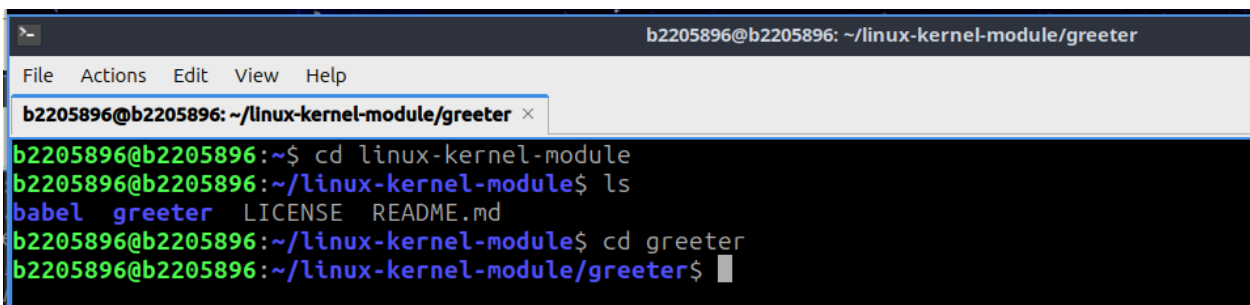
- Clone this repository to your computer:

<https://github.com/TuanThai/linux-kernel-module.git>



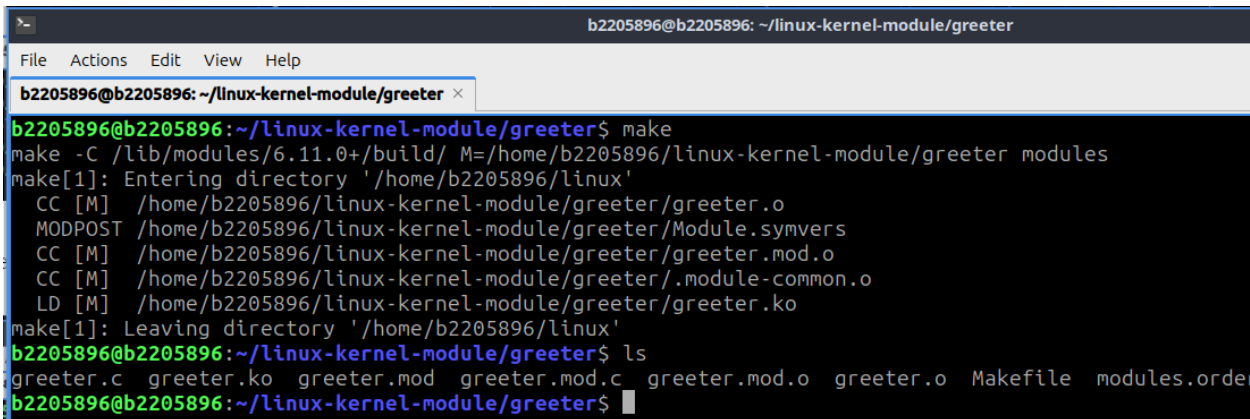
```
b2205896@b2205896: ~  
File Actions Edit View Help  
b2205896@b2205896: ~ x  
b2205896@b2205896:~$ git clone https://github.com/TuanThai/linux-kernel-module.git  
Cloning into 'linux-kernel-module'...  
remote: Enumerating objects: 54, done.  
remote: Counting objects: 100% (6/6), done.  
remote: Compressing objects: 100% (6/6), done.  
remote: Total 54 (delta 1), reused 2 (delta 0), pack-reused 48 (from 1)  
Receiving objects: 100% (54/54), 16.35 KiB | 408.00 KiB/s, done.  
Resolving deltas: 100% (17/17), done.  
b2205896@b2205896:~$
```

- Move into greeter/ directory.



```
b2205896@b2205896: ~/linux-kernel-module/greeter  
File Actions Edit View Help  
b2205896@b2205896: ~/linux-kernel-module/greeter x  
b2205896@b2205896:~$ cd linux-kernel-module  
b2205896@b2205896:~/linux-kernel-module$ ls  
babel greeter LICENSE README.md  
b2205896@b2205896:~/linux-kernel-module$ cd greeter  
b2205896@b2205896:~/linux-kernel-module/greeter$
```

- Build the module using make command. The module is compiled to greeter.ko



```
b2205896@b2205896: ~/linux-kernel-module/greeter  
File Actions Edit View Help  
b2205896@b2205896: ~/linux-kernel-module/greeter x  
b2205896@b2205896:~/linux-kernel-module/greeter$ make  
make -C /lib/modules/6.11.0+/build/ M=/home/b2205896/linux-kernel-module/greeter modules  
make[1]: Entering directory '/home/b2205896/linux'  
CC [M] /home/b2205896/linux-kernel-module/greeter/greeter.o  
MODPOST /home/b2205896/linux-kernel-module/greeter/Module.symvers  
CC [M] /home/b2205896/linux-kernel-module/greeter/greeter.mod.o  
CC [M] /home/b2205896/linux-kernel-module/greeter/.module-common.o  
LD [M] /home/b2205896/linux-kernel-module/greeter/greeter.ko  
make[1]: Leaving directory '/home/b2205896/linux'  
b2205896@b2205896:~/linux-kernel-module/greeter$ ls  
greeter.c greeter.ko greeter.mod greeter.mod.c greeter.mod.o greeter.o Makefile modules.order  
b2205896@b2205896:~/linux-kernel-module/greeter$
```

- Install the module using insmod greeter.ko command, then show that the module has been installed using lsmod | grep greeter command

```
b2205896@b2205896: ~/linux-kernel-module/greeter
File Actions Edit View Help
b2205896@b2205896: ~/linux-kernel-module/greeter x
b2205896@b2205896:~/linux-kernel-module/greeter$ sudo insmod greeter.ko
[sudo] password for b2205896:
b2205896@b2205896:~/linux-kernel-module/greeter$ lsmod | grep greeter
greeter                12288  0
b2205896@b2205896:~/linux-kernel-module/greeter$
```

- Show the information of the module using `modinfo greeter.ko`

```
b2205896@b2205896: ~/linux-kernel-module/greeter
File Actions Edit View Help
b2205896@b2205896: ~/linux-kernel-module/greeter x
b2205896@b2205896:~/linux-kernel-module/greeter$ modinfo greeter.ko
filename:           /home/b2205896/linux-kernel-module/greeter/greeter.ko
version:            0.1
description:        A simple kernel module to greet a user
license:            GPL v2
author:             Dave Kerr
srcversion:         92DAF73EE64FF6362E081BD
depends:
name:               greeter
retpoline:          Y
vermagic:           6.11.0+ SMP preempt mod_unload modversions
parm:               name:The name to display in /var/log/kern.log (charp)
b2205896@b2205896:~/linux-kernel-module/greeter$
```

- Show kernel log with `dmesg`

```
b2205896@b2205896: ~/linux-kernel-module/greeter
File Actions Edit View Help
b2205896@b2205896: ~/linux-kernel-module/greeter x
b2205896@b2205896:~/linux-kernel-module/greeter$ sudo dmesg
[ 0.000000] Linux version 6.11.0+ (b2205896@b2205896) (gcc (Ubuntu 13.2.0-23ubuntu4) 13.2.0, GNU ld (GNU Bin
ntu) 2.42) #1 SMP PREEMPT_DYNAMIC Fri Sep 27 08:39:56 WIB 2024
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-6.11.0+ root=UUID=e5858f73-1012-495c-a53a-3f91f05afaf6 ro
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Hygon HygonGenuine
[ 0.000000] Centaur CentaurHauls
[ 0.000000] zhaoxin Shanghai
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000000009fc00-0x00000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000000f0000-0x0000000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000000100000-0x000000000000dfffff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000000dfff0000-0x000000000000dfffffff] ACPI data
```

```
[ 543.230072] workqueue: e1000_watchdog [e1000] hogged CPU for >10000us 7 times, consider switching to WQ_UNBOUND
[ 681.979457] workqueue: blk_mq_run_work_fn hogged CPU for >10000us 11 times, consider switching to WQ_UNBOUND
[ 776.937850] workqueue: e1000_watchdog [e1000] hogged CPU for >10000us 11 times, consider switching to WQ_UNBOUND
[ 1733.232671] workqueue: vmstat_shepherd hogged CPU for >10000us 4 times, consider switching to WQ_UNBOUND
[ 1832.381868] workqueue: vmstat_shepherd hogged CPU for >10000us 5 times, consider switching to WQ_UNBOUND
[ 2029.042063] greeter: loading out-of-tree module taints kernel.
[ 2029.042075] greeter: module verification failed: signature and/or required key missing - tainting kernel
[ 2029.044973] greeter: module loaded at 0x00000000d969e2b8
[ 2029.044979] greeter: greetings Bilbo
b2205896@b2205896:~/linux-kernel-module/greeter$
```

- Remove the module using `rmmod greeter.ko` command, then show that the module has been removed using `lsmod | grep greeter` command.

```
b2205896@b2205896: ~/linux-kernel-module/greeter
File Actions Edit View Help
b2205896@b2205896: ~/linux-kernel-module/greeter x
b2205896@b2205896:~/linux-kernel-module/greeter$ sudo rmmod greeter
b2205896@b2205896:~/linux-kernel-module/greeter$ lsmod | grep greeter
b2205896@b2205896:~/linux-kernel-module/greeter$
```

- Show kernel log with `dmesg`

```
b2205896@b2205896: ~/linux-kernel-module/greeter
File Actions Edit View Help
b2205896@b2205896: ~/linux-kernel-module/greeter x
b2205896@b2205896:~/linux-kernel-module/greeter$ sudo dmesg
[ 0.000000] Linux version 6.11.0+ (b2205896@b2205896) (gcc (Ubuntu 13.2.0-23ubuntu4) 13.2.0) #1 SMP PREEMPT_DYNAMIC Fri Sep 27 08:39:56 WIB 2024
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-6.11.0+ root=UUID=e5858f73-1012-4955-8000-000000000000
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Hygon HygonGenuine
[ 0.000000] Centaur CentaurHauls
[ 0.000000] zhaoxin Shanghai
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x00000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000000009fc00-0x00000000000009ffff] reserved
[ 681.979457] workqueue: blk_mq_run_work_fn hogged CPU for >10000us 11 times, consider switching to WQ_UNBOUND
[ 776.937850] workqueue: e1000_watchdog [e1000] hogged CPU for >10000us 11 times, consider switching to WQ_UNBOUND
[ 1733.232671] workqueue: vmstat_shepherd hogged CPU for >10000us 4 times, consider switching to WQ_UNBOUND
[ 1832.381868] workqueue: vmstat_shepherd hogged CPU for >10000us 5 times, consider switching to WQ_UNBOUND
[ 2029.042063] greeter: loading out-of-tree module taints kernel.
[ 2029.042075] greeter: module verification failed: signature and/or required key missing - tainting kernel
[ 2029.044973] greeter: module loaded at 0x00000000d969e2b8
[ 2029.044979] greeter: greetings Bilbo
[ 2186.924088] workqueue: e1000_watchdog [e1000] hogged CPU for >10000us 19 times, consider switching to WQ_UNBOUND
[ 2234.939024] greeter: goodbye Bilbo
[ 2234.939034] greeter: module unloaded from 0x00000000a4dc832e
b2205896@b2205896:~/linux-kernel-module/greeter$
```


- Move to `greeter.c` file, then briefly explain below functions:

```
greeter_init
greeter_exit
module_init(greeter_init)
module_exit(greeter_exit)
```

The brief explanation of the functions in `greeter.c`:

- **greeter_init**: It's invoked when the module is loaded into the kernel. Within this function:

- `pr_info` logs a message indicating the module's loading address.
- It displays a greeting to the user, and the **name** parameter.
- Returns 0 to signify successful initialization.

- **greeter_exit**: This function is called when the module is unloaded from the kernel. Its responsibilities include:

- Logging a message to the user.
- Indicating the module's unloading address.

- **module_init(greeter_init)**: This macro registers the `greeter_init` function as the entry point for the module. It tells the kernel to call `greeter_init` when the module is loaded.

.

- **module_exit(greeter_exit)**: This macro registers the `greeter_exit` function as the exit point for the module. It tells the kernel to call `greeter_exit` when the module is unloaded.

(take screenshots to show that you finish this exercise)

---END---