CS492 Spring 2020 Lab 1
Linux kernel source and git

This is an individual assignment. Individual assignments, as the word indicate, are to be done **INDIVIDUALLY**. Any sign of collaboration will result in a 0 and being reported to the Honor Board.

Introduction

The objectives of this project are to familiarize with the kernel source code, using the git version control software. Specifically, you will:

- Add a printk statement in the kernel log during the boot process; with your name and id;

All the project steps will be performed in the supplied Debian virtual machine. The notions from the course involved in this project are the following:

- Linux source code exploration and compilation. Installing and running a modified kernel;
- ~~Version control with git: init, cloning, branching, creating a patch;~~
- The printk primitive.

Project Steps

a) Download the Debian virtual machine from Canvas, use the provided instructions to import it in VirtualBox. Eventually, customize the number of CPUs and amount of memory according to your computer. It is recommended to configure the virtual machine with at least 2 CPUs and 4GB of RAM. The user "student" has been created with associated username: "student" and password: "cs492"; the super-user's username is "root" and its password is "cs492". There is a graphical user interface. There are at least the following editors: *vi*, *vim*, and **nano**. The source code of the Linux kernel is in the ***linux-4.9/*** subdirectory of the "student" 's home folder. The kernel code has been compiled at least once. Please don't use this directory for your work, but create a copy of it for this assignment.

b) ~~Create a git repository in **github classroom** using a different classroom based on the section you are enrolled. Links to github classroom will be provided in Canvas.~~

c) ~~Use the git repository to track all your modifications to the linux kernel source-code – i.e., the repository must contain the linux kernel source-code.~~

d) Change the Linux kernel configuration in order to add your name or student ID in the LOCALVERSION macro (CONFIG_LOCALVERSION). Compile and install the Linux kernel from the sources, then reboot the virtual machine. When Linux is up again, check that the currently running kernel is the one you just compiled. You need to provide a screenshot.

e) Add a printk() statement that outputs "Hello World from NAME (STUDENT_ID)" in the kernel log during the kernel boot process. Where NAME is your name, and STUDENT_ID is your student id. Print this line as early as possible during the boot process (in the first 16 lines of the kernel log buffer). In order to print this string as early as possible during the kernel boot, you should identify what is printed in the early kernel boot first – by having a look at dmesg, and searching

one of these strings in the kernel source-code. Compile, install, reboot and check that the line is present in the kernel log. You need to provide a screenshot.
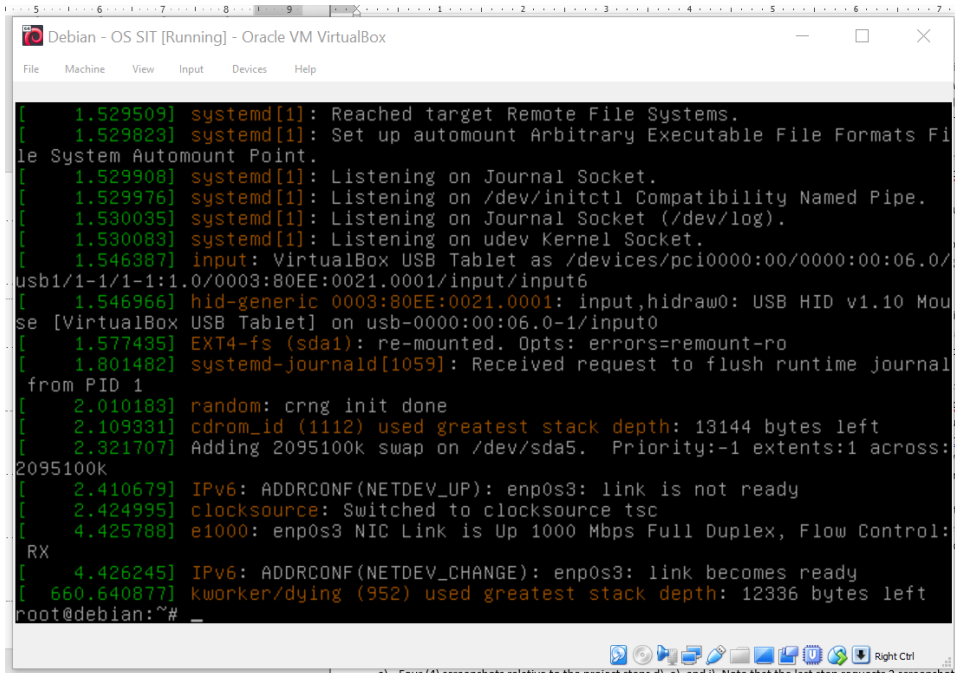
To be submitted

The following should be submitted.

a) Two (2) screenshots relative to the project steps d) and e).
b) A patch containing the modifications made to the kernel sources. The patch should be applicable to your original *linux-4.9/* directory; therefore, do not overwrite it!

All of these should be contained in a tarball, with the following format: STUDENT_ID.lab1.tar.gz. Where STUDENT_ID is your student identification number. The tarball must be submitted on Canvas.

Every screenshot should be taken with the entire VirtualBox window, an example of a valid screenshot and a non-valid one foll

Debian - OS SIT [Running] - Oracle VM VirtualBox

File   Machine   View   Input   Devices   Help

```
[    1.529509] systemd[1]: Reached target Remote File Systems.
[    1.529823] systemd[1]: Set up automount Arbitrary Executable File Formats Fi
le System Automount Point.
[    1.529908] systemd[1]: Listening on Journal Socket.
[    1.529976] systemd[1]: Listening on /dev/initctl Compatibility Named Pipe.
[    1.530035] systemd[1]: Listening on Journal Socket (/dev/log).
[    1.530083] systemd[1]: Listening on udev Kernel Socket.
[    1.546387] input: VirtualBox USB Tablet as /devices/pci0000:00/0000:00:06.0/
usb1/1-1/1-1:1.0/0003:80EE:0021.0001/input/input6
[    1.546966] hid-generic 0003:80EE:0021.0001: input,hidraw0: USB HID v1.10 Mou
se [VirtualBox USB Tablet] on usb-0000:00:06.0-1/input0
[    1.577435] EXT4-fs (sda1): re-mounted. Opts: errors=remount-ro
[    1.801482] systemd-journald[1059]: Received request to flush runtime journal
 from PID 1
[    2.010183] random: crng init done
[    2.109331] cdrom_id (1112) used greatest stack depth: 13144 bytes left
[    2.321707] Adding 2095100k swap on /dev/sda5.  Priority:-1 extents:1 across:
2095100k
[    2.410679] IPv6: ADDRCONF(NETDEV_UP): enp0s3: link is not ready
[    2.424995] clocksource: Switched to clocksource tsc
[    4.425788] e1000: enp0s3 NIC Link is Up 1000 Mbps Full Duplex, Flow Control:
 RX
[    4.426245] IPv6: ADDRCONF(NETDEV_CHANGE): enp0s3: link becomes ready
[  660.640877] kworker/dying (952) used greatest stack depth: 12336 bytes left
root@debian:~#
```

Right Ctrl

a)   Four (4) screenshots relative to the project steps d), e), and i). Note that the last step requests 2 screenshots.

Further Notes

After installing a new kernel it may be necessary to re-install the VirtualBox extensions. As root run:

```
# rcvboxadd setup
```

In order to create a patch you can use one of the following solutions.

- Use git. You can find *original_commit_hash*, the hash of your initial git commit, using *git log*

  ```
  $ git diff original_commit_hash
  ```

- Use diff. *linux-4.9/* is the vanilla Linux kernel source code and *lab1-linux-4.9/* is a copy of it, which you modify for this homework

  ```
  $ diff -ur linux-4.9/ lab1-linux-4.9/
  ```