**rCS492 Assignment 1**

Adding a new system call and developing a kernel module

# 

**HOW TO MAKE YOUR COMPILING FASTER:**

<https://docs.google.com/presentation/d/1YgvzbRMgcaX0G2KeqQT4QXbQ3ACd9N_lhPOCM9gBPu8/edit#slide=id.p>

<https://kaltura.stevens.edu/media/Assignment+1+video+walkthrough/1_x0yhmnv3>

# Introduction

The objective of this assignment is to familiarize yourselves with the kernel source code. Specifically, you will:

* Add a new system call and call the system call from a user-space program;
* Add a kernel module

All of the assignment’s steps must be performed in the provided Debian virtual machine. The notions from the course involved in this assignment are the following:

* Linux source code exploration and compilation.
* Installing and running a modified kernel, and a module;
* The printk primitive;
* User / kernel space communication: system calls

# 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 use the copy that you will get as follows.**

b) Join the repo in the github **classroom** we prepared for this assignment. Students from all sections can join the repo by clicking <https://classroom.github.com/a/pOb9bFbN> (you will need to log into your github account and you will be asked to join the assignment after logging in)

c) The repo already includes a clean version of Linux kernel 4.9 (the same version as we prepared in the VM). You need to clone the repo to your VM by running: “git clone URL\_OF\_THE\_ASSIGNMENT\_REPO”. Git is not available in the VM (to install git, log in as “root”, and run “apt-get update” and then “apt-get install git”).

d) Change the Linux kernel configuration to add your name and 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 system call named “yourfirstname\_syscall” (replace “yourfirstname” with your first name) that takes as parameter a pointer to a character array containing a string, and returns a signed integer (int). If the string size is larger than 128, the system call should immediately return -1. Otherwise, the system call’s job is to replace all occurrences of the letter "x" by the letter "y" in that string. The system call returns the number of replacements performed. The syscall implementation should be in a separate file from the rest of the kernel, but still in the kernel source code (not implemented as a module).

f) Write a user-space C program invoking the system call. In particular, be sure to check for the case where the string size is larger than 128. The C program must print on console: 1) the arguments given to the syscall; 2) the return value of the syscall.

g) Install the new kernel and check that your system call works correctly by using the user-space C program. **You need to provide a screenshot of kernel log and screenshot of the output of the user-space C program.**

h) Write a Linux kernel module that has a printk() statement that outputs “Hello World from NAME (STUDENT\_ID)” in the kernel log after the module is loaded. Where NAME is your name, and STUDENT\_ID is your student id. When the kernel module unloads (using rmmod) it prints the PID and name of the current process. Compile, install, load and unload the kernel module. **You need to provide a screenshot of the kernel log after loading the module and a screenshot of the kernel log after unloading the module.**

i) Push a commit which includes the following to the github classroom repo:

* Your changes to the kernel code
* A new top-level folder (named “cs492\_2021s\_assignment1”) which contains:
  + The source code file of your C program to call “yourfirstname\_syscall”
  + The source code file of your kernel module and the Makefile to build the kernel module

# 

# Results to be handed - Deadline: 24th of Feb 2021

A PDF containing the following is expected to be submitted to Canvas **by 11:59:59pm on Feb 24, 2021**. This assignment is worth **10%** of your final grade.

a) Five screenshots related to the project steps d), i) and j). Note that the last two steps request 2 screenshots each.

b) The link to your commit to the GitHub classroom repo

Every screenshot should be taken with the entire VirtualBox window

Further Notes

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

# rcvboxadd setup

Useful links:

· [https://medium.com/anubhav-shrimal/adding-a-hello-world-system-call-to-linux-kernel-dad32875872](https://na01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fmedium.com%2Fanubhav-shrimal%2Fadding-a-hello-world-system-call-to-linux-kernel-dad32875872&data=02%7C01%7Cabarbala%40stevens.edu%7C841033b2740c498aac9b08d6906751f7%7C8d1a69ec03b54345ae21dad112f5fb4f%7C0%7C0%7C636855171521186867&sdata=uFAVfegfHAh0oukgaFndH4S0e91tp6k0gxNODI4nBTk%3D&reserved=0)

· [https://brennan.io/2016/11/14/kernel-dev-ep3/](https://na01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fbrennan.io%2F2016%2F11%2F14%2Fkernel-dev-ep3%2F&data=02%7C01%7Cabarbala%40stevens.edu%7C841033b2740c498aac9b08d6906751f7%7C8d1a69ec03b54345ae21dad112f5fb4f%7C0%7C0%7C636855171521186867&sdata=iZ60n%2F4Nksfi%2F4IfNgcJ%2BqAM45355TQc06W11kFlboc%3D&reserved=0)

· [https://stackoverflow.com/questions/17751216/writing-a-new-system-call](https://na01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fstackoverflow.com%2Fquestions%2F17751216%2Fwriting-a-new-system-call&data=02%7C01%7Cabarbala%40stevens.edu%7C841033b2740c498aac9b08d6906751f7%7C8d1a69ec03b54345ae21dad112f5fb4f%7C0%7C0%7C636855171521196876&sdata=W6bDPY8PoeiJk4O10%2Brk5BFjve%2BV6yCNGFHC8N0JcU0%3D&reserved=0)

· [http://tldp.org/LDP/lkmpg/2.6/html/index.html](https://na01.safelinks.protection.outlook.com/?url=http%3A%2F%2Ftldp.org%2FLDP%2Flkmpg%2F2.6%2Fhtml%2Findex.html&data=02%7C01%7Cabarbala%40stevens.edu%7C841033b2740c498aac9b08d6906751f7%7C8d1a69ec03b54345ae21dad112f5fb4f%7C0%7C0%7C636855171521196876&sdata=9eR%2FtHLtS25CKcnAzndijKkSFBKuAK7%2BiCA4qjdKwvI%3D&reserved=0)

In order to push a commit:

* Add a file you would like to commit (the file must locate inside the repo; it is OK that the file locates in a folder you newly create inside the repo):
  + $ git add /path/to/your/target/file (You can repeat this to add more files)
  + $ git commit -m “whatever message you would like”
  + $ git push
* You can get a URL to the commit by visiting the repo on GitHub

# Grading Policy

* This is an individual assignment. Individual assignments, as the words indicate, are to be done INDIVIDUALLY. Any sign of collaboration will result in a 0
* Any form of plagiarism will result in a 0 and being reported to the Honor Board.
* Any late submission without pre-approval will result in a 0
* The points of this assignment will be distributed as follows
  + Completing step d) (10%)
  + Completing step e) (30%)
  + Completing step f) (10%)
  + Completing step g) (10%)
  + Completing step h) (30%)
  + Well formatted code with comments (10%)
  + Incorrect implementations (based on the code submitted, instead of the screenshot) of a step will lead to 0 in that step
  + Missing code for a step in the commit will lead to 0 in that step, regardless of the presence of a snapshot or not