CSC362 Operating System

Project 0 – Configure Experimental Environment

Dr. Qixin Deng

Friday 24th January, 2025

Preface

Practice is crucial for an operating systems course, but this practice should not just be limited to the simulated implementation of certain algorithms. Even if one has programmed a complete 'file system', if it is not integrated with modules like disk drives, cache, and memory management, then, despite writing a lot of code, what is achieved is merely the implementation of certain concepts (more like a data structure course project) and still cannot control an actual running operating system. All assignments are designed to emphasize practice on an operating system that can actually run.

Therefore, the experiments are based on an actual runnable operating system, and an actual operating system is chosen for this purpose. There are many options for operating systems to choose from, such as Windows, Linux, xv6, etc. For undergraduates learning operating systems for the first time, they should select the operating system with which they have experience. This will fully stimulate their interest in learning. The Linux operating system, being completely open-source, has abundant materials and is increasingly used by more people, making Linux an excellent choice.

On Linux, there are two choices: either to use the latest version of Linux or to choose a simple early version. The new, mature version of Linux, though very mature, also has a very large amount of code, over 27 million lines, and is very complex. Modifications and compilations on it can be very difficult and slow, making it unsuitable for undergraduate operating system courses. Therefore, considering the simplicity needed for a first exposure to an operating system course, the ease of obtaining reference materials, and the potential for further exploration in the future, I chose to conduct operating system experiments on Linux 0.11. The Linux 0.11 kernel has less than 20,000 lines of code and is easy to understand and modify, but Linux 0.11 is also a complete system with not much difference from today's Linux systems. The time spent on it can provide students with ample exercise, laying a solid foundation for understanding operating systems and for writing their own operating systems.

Experimental Environment Setup

The main software environment consists of Bochs, gcc, your favorite editor/IDE, your favorite operating system, and the Linux 0.11 source code. The basic workflow of the experiment involves writing application programs and modifying the Linux 0.11 source code according to the experiment requirements. After compilation with gcc, the target code is run and debugged in the virtual environment of Bochs.

1. Virtual Machine

Please install a virtual machine on your computer. There are two virtual machines I recommend:

1. VMware
2. VirtualBox

VMware Fusion Player offers a Personal Use License, available for free with a valid CustomerConnect account. Home users, Open Source contributors, students, and anyone else can use Fusion Player Free for Non-Commercial activity [https://customerconnect.vmware.com/en/evalcenter?p=fusion-player-personal-13]. VirtualBox [https://www.virtualbox.org/] is the only professional solution that is freely available as Open Source Software under the terms of the GNU General Public License (GPL) version 3. If you are using a Mac with M-series chips, you cannot use VirtualBox.

Instead of a virtual machine, you can also choose to directly install a secondary operating system on your computer.

2. Install Ubuntu 16.04

Please download Ubuntu's old version 16.04 [https://old-releases.ubuntu.com/releases/], which is the one I tested. If you want you can also try other versions. Install Ubuntu on your virtual machine. There are a lot of tutorials on the Internet on how to install Ubuntu on VMware or VirtualBox. Please Google it and learn for yourself.

3. Environment Setup

The red words are not commands, but comments. Please download the zip file "oslab.zip" from Canvas/files. Unzip the file in your home directory, and go inside the folder, run the following Linux command:

```
$ ./setup.sh
```

You have correctly completed the environment setup if it prints that "installation finished". If you want to skip the software source update step, please add the parameter -s or –skip-update after ./setup.sh.
After setting up the environment, you need to go inside linux 0.11 folder to compile the system:

```
# in oslab directory
$ cd linux-0.11
$ make all
```

It's simple to run the latest compiled kernel in Bochs. Execute the following command in the oslab directory:

```
# in oslab directory
$ ./run
```

If a Bochs window appears, showing the boot process of Linux, and finally stops at '[/usr/root/]#', it indicates that the run was successful. If the Bochs window appears but the boot process of linux does not start, go back to the terminal, and run the following command:

```
# continue to boot
$ c
```

Considering that each operating system experiment requires resetting the linux code, I specially added a reset function. This function is provided by the ./run command.

```
# in oslab directory
$ ./run init
```

After it, you need to rerun the "make all" command to generate a new image.

4. File Exchange Between Ubuntu and Linux 0.11

In oslab folder, 'hdc0.11new.img' is the root file system image file after the 0.11 kernel boots, equivalent to the hard disk loaded in the Bochs virtual machine. The method to access its content on Ubuntu is:

```
$ ./mounthdc
```

After this, the 'hdc' directory will have the same file system as the 0.11 kernel, and you can read and write any file (some files may require sudo to access). After reading and writing, don't forget to unmount this file system:

```
$ sudo umount hdc
```

Note 1: Do not mount the image file while the 0.11 kernel is running, as it may damage the file system. Similarly, do not run the 0.11 kernel when it is already mounted.

Note 2: Before closing Bochs, you need to run 'sync' in the 0.11 command line to ensure that all cached data is written to disk before closing Bochs.

Something about VirtualBox

You need to activate "Drag and Drop". To do that, please follow the steps below:
1. Start your virtual machine, On the top bar, click "Devices"
2. Go to Drop and Drop option, and select "Bidirectional"
3. Still under "Devices", click "Insert Guest Additions CD Image..." to install it.

Experiment Content and Submission

You need to submit a demo video (Use screen recorder). In the video, I need to clearly see the steps I listed below:

1. start Ubuntu on a virtual machine
2. start Linux 0.11 in Bochs
3. check the gcc version in Linux 0.11
4. copy a file to Linux 0.11. The content of the file is just your name. Use the Linux command "cat" to print the file's content.

You can work in a group and seek help. Every student needs to submit a demo video. Every student needs to finish the environment setup. If you are using M-series Mac or Mackbook, you need to use a computer in the department which is running Linux or Windows.