

CSCE 231/2303 Fall 2018

Assignment 5: X86_64 Environment Setup and Boot Loader Enhancements

Assigned: Sunday, October 21th in Class

Due: Sunday, October 28th at 10:00 am

Delayed submission with penalty until Tuesday, October 30th at mid-night.

Goals

This assignment is an individual assignment and you will work on it on your own. The goal of this assignment is to achieve two things, which are to setup the development environment you will need in the coming assignments, and to introduce to you the skeleton code of the boot loader that you will complete its missing parts in the rest of the assignments until the end of the course. We will be using virtual box as a hypervisor to deploy your development environment. it is fundamentally important to attend Tuesday labs from now on to be able to get the needed help to do this and the coming assignments.

Details

This assignment is split into two parts. The first part is an ungraded tutorial to setup your development environment and the second part is a graded programming task to amend the boot sector packaged in the provided boot loader skeleton code.

1. Setting up the Development Environment:

Setting up Virtual Box:

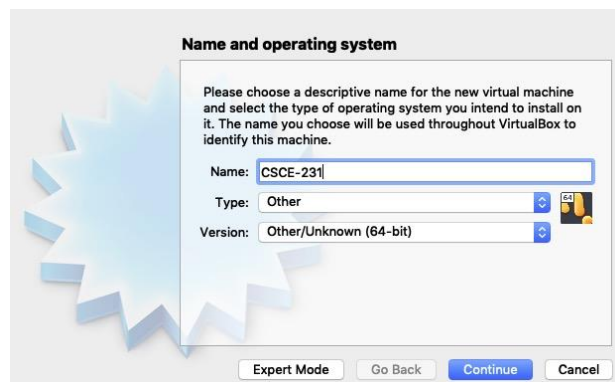
If you already have virtual box setup on your environment and can run X86_64 HW virtualization, then you can skip this part. Also if you have a native Ubuntu based x86_64 Linux installation you can skip this part and work on it natively, **but in any case we recommend very much to use Virtual Box as it will isolate your development environment and protect your main base environment from any data losses due to any mistakes.**

Follow the following steps to setup Virtual Box:

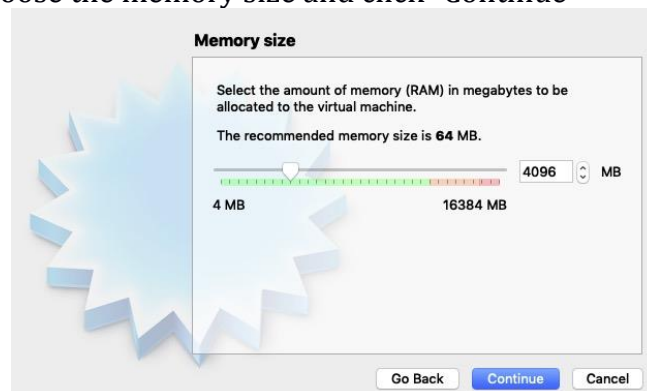
1. Start by checking your BIOS setup and make sure that the hardware virtualization support is enabled in your BIOS. Start your computer and enter the BIOS setup, then navigate until you find the “**Virtualization Support**” feature item and make sure that it is enabled. Entering the BIOS is very much dependent on your type of hardware and brand, check out google for that. (**For Mac users this step can be skipped**)
2. Download the latest Virtual Box (>= 5.2.x) for your OS from VirtualBox download site <https://www.virtualbox.org/wiki/Downloads>, and install

it. The installation is very simple, just keep on following the installation default screens.

3. After installation, download the Virtual Box **Extension Pack** for your installed Virtual Box version from this link [https://download.virtualbox.org/virtualbox/5.2.20/Oracle VM VirtualBox Extension Pack-5.2.20.vbox-extpack](https://download.virtualbox.org/virtualbox/5.2.20/Oracle_VM_VirtualBox_Extension_Pack-5.2.20.vbox-extpack), and install it as well.
4. Make sure you have the latest Mint Linux x86_64 19 ISO file downloaded on your computer; I will provide the ISO file over a flash drive in the lab.
5. Start Virtual Box, and when the Virtual Box main console opens click on "New".
6. Fill in a name for your virtual machine and choose for the Type "Other", and for the Version "Other/Unknown (64-bit)", and click "Continue". If you can not find the 64-bit type then your hardware does not support virtualization or it is not enabled in the BIOS (go back to step1)



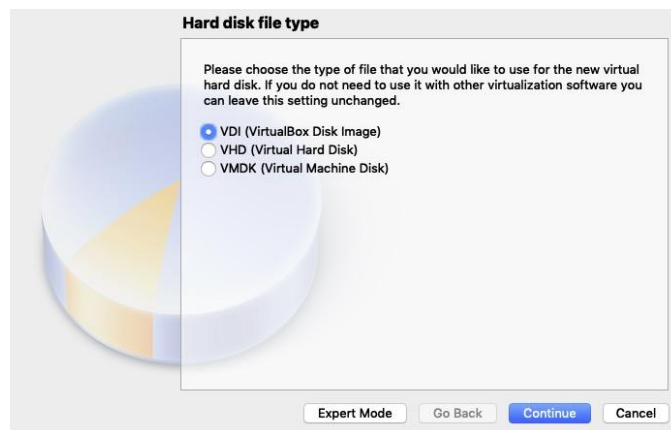
7. You need to assign your guest Virtual Machine a memory size. This depends on how much memory you have in your host computer, and you should always make sure to leave at least 1 GB of memory unused. The minimum you can assign to your guest machine is 3 GB to be able to conduct the assignments reliably. It is recommended to assign it 4GB or more. Choose the memory size and click "Continue"



8. You will need now to create a virtual disk which is in the form of a file, just click “Create”.



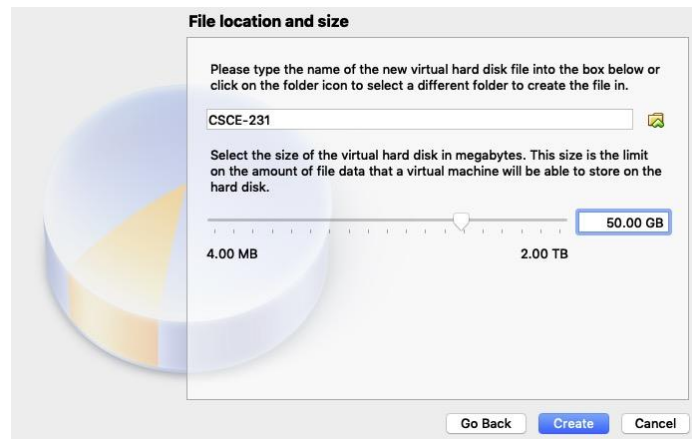
9. You will be asked which virtual disk type to use and you just need to use the default VDI format, and click “Continue”.



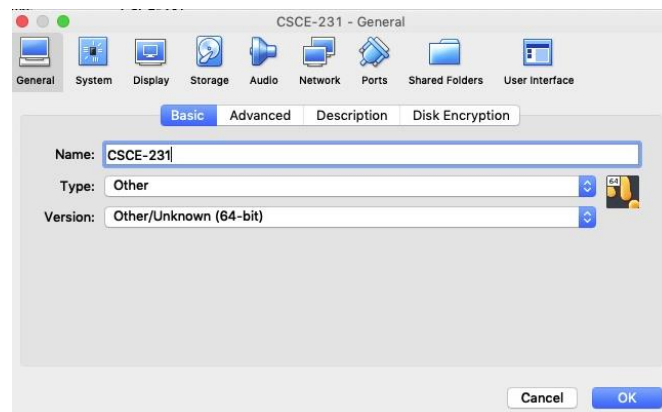
10. Choose the “Dynamic allocated” so you can only use disk space from your host as needed, and click “Continue”



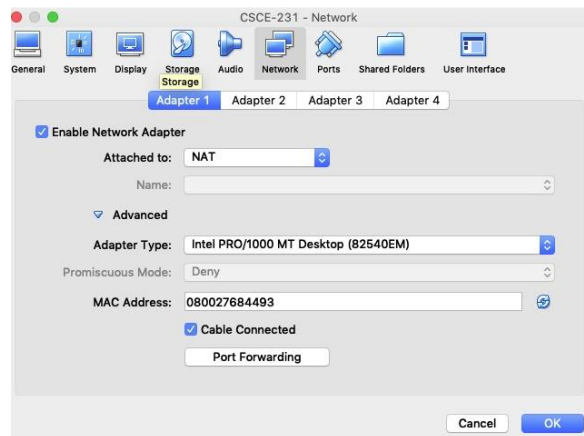
11. Now you can choose the size of your virtual disk. We recommend that you choose 50 GB as your virtual disk size to be in the safe side. Click “Create”.



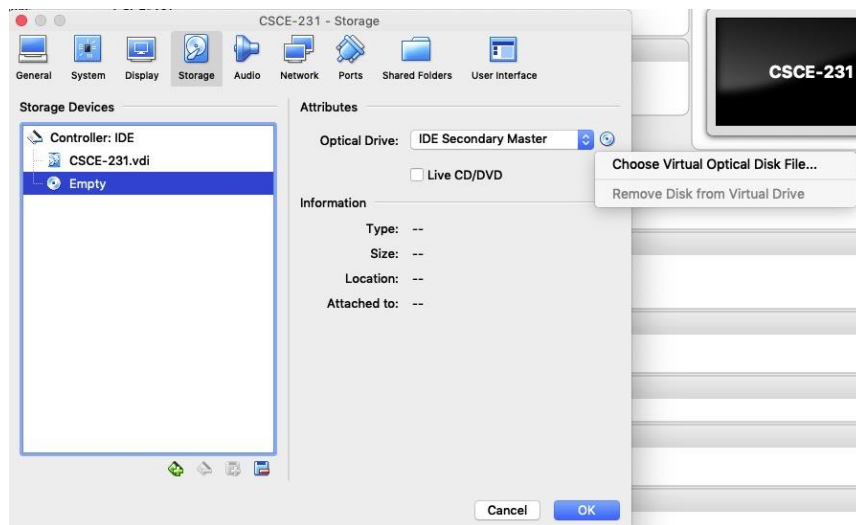
12. Your virtual machine is now created, and will appear on the left navigation panel of the console.
13. Select your virtual machine using a single mouse click on it.
14. Then click the button “Settings” on the top tool bar in the console. You will get the settings window displayed.



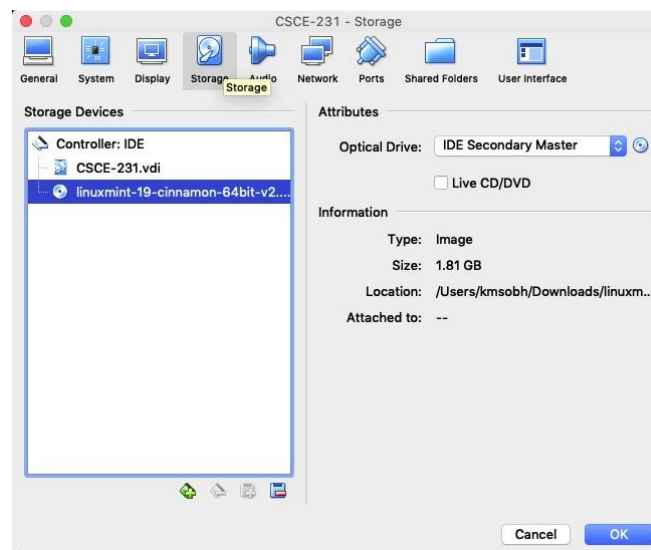
15. Click on “Network” in the toolbar, and expand the advanced section, then choose the “PRO/1000 MT Desktop (82540EM)” in the Adapter Type.



16. Click on the “Storage” in the toolbar, and choose the virtual CD-ROM labeled as “Empty”, and then click on the CD-ROM icon beside the list box that has the item “IDE Secondary Master” selected, and then click on “Choose Virtual Optical Disk File” in the drop down.



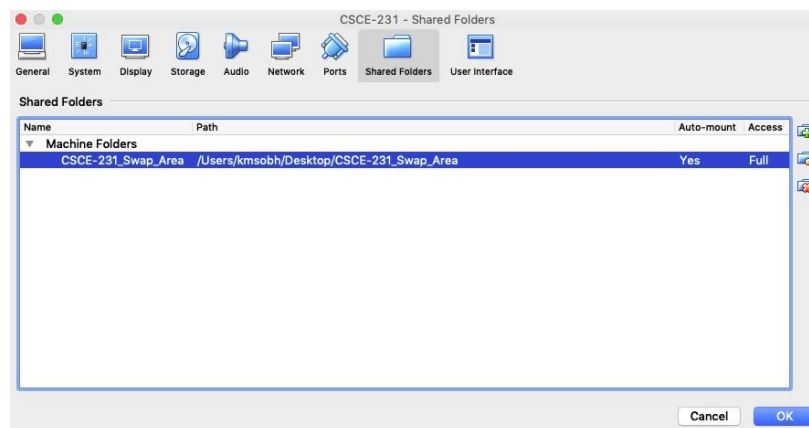
17. Locate the Linux Mint 9 ISO file and click “Open”. Now you have set the installation media in the bootable virtual CD-ROM of your guest virtual machine.



18. Click the “System” button on the toolbar, and under the “Processor” tab make sure to check “Enable PAE/NX”, and under the “Acceleration” tab choose “KVM” for the Paravirtualization Interface, and check “Enable Nested Paging”.
19. Next you choose the “Shared Folders” from the toolbar where you can assign a folder from your host environment over which you can exchange files with your guest environment. Just create an empty folder on your desktop and then create an entry for it inside your virtual machine by clicking the add folder sign on the right and filling the dialog as follows.



20. Click “Okay” on the pop-up and you will have your shared folder setup.



21. Click “Okay” on the settings window and your virtual machine is ready to start.
22. When your virtual machine starts it will look like a normal hardware and your Mint installation will start. Some times you get a faulty VGA display, and in that case you need to interrupt Mint boot count down and choose “Start in Comparability mode”.
23. When the Mint Live environment starts up, click the CD-ROM icon labeled “Install Linux Mint” on the desktop. Just follow the installation until it completes successfully.
 - Make sure to install third-party software when you are prompted to.
 - Continue with the default choices on all dialogs.
 - Create a user of your own.

Configuring and Tuning Mint:

After installing Mint Linux we need to fine tune it, install all needed packages for our development environment, and copy the boot loader skeleton and make sure it is working. You should follow the following steps:

1. Start your guest virtual machine and login using your username and password you created during installation.
2. Inside your guest machine click on the File Manage icon in the taskbar.
3. When the file manager opens choose from the menu bar **Edit→ Preferences**.

4. Choose the Toolbar Tab in the preferences window.
5. Check all unchecked check boxes and click "Close".
6. Close the file manager.
7. First we need to install VirtualBox Guest Tools. Choose from the VirtualBox menu **Devices→ Insert Guest Additions CD image**.
8. You will get a popup inside your guest desktop, just click cancel.
9. You will find a new CD icon on your desktop labeled "Vbox_GAs_5....".
10. Double click on it, and when the file manager opens right click in an empty area and choose "Open in Terminal".
11. A terminal will be opened for you.
12. Enter the command: **sudo ./VBoxLinuxAdditions.run**
13. You will be prompted for a password, just enter your account password.
14. Wait until the guest additions get installed, and the terminal prompt returns back.
15. Now you need to reboot, so enter the command: **sudo reboot**
16. When the guest virtual machine starts again, login using your account.
17. Now you can stretch the guest virtual machine desktop to the dimensions that suites you.
18. Right click on CD icon labeled on "Vbox_GAs_5...." and choose "Eject".
19. Notice that you have another disk icon on your desktop which is basically your shared drive with your host that you can exchange files over.
20. Before using it we need to do some stuff, so open a terminal from the taskbar, and issue the following command one after the other and wait for each command to finish execution before issuing the next one. Also enter your account password whenever prompted (Each of the commands below must be written on a single separate line):
 - **sudo apt-get update**
 - **sudo apt-get upgrade -y**
 - **curl https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > microsoft.gpg**
 - **sudo install -o root -g root -m 644 microsoft.gpg /etc/apt/trusted.gpg.d/**
 - **sudo sh -c 'echo "deb [arch=amd64] https://packages.microsoft.com/repos/vscode stable main" > /etc/apt/sources.list.d/vscode.list'**
 - **rm -rf microsoft.gpg**
 - **sudo apt-get install nasm make autoconf automake g++ qemu-system-* bochs* code -y**
 - **sudo adduser \$USER vboxsf**
21. Reboot the virtual machine by issuing the command: **sudo reboot**
22. Login to your account when the virtual machine is restarted.
23. Get the boot loader skeleton zip file from blackboard and copy it to the shared folder on your host environment.
24. Now open the shared folder on your guest and you should find the skeleton zip file available.

25. Copy the file to your home directory under a new directory. Create a new directory and call it "csce-231" under your home folder from the file manager.
26. Execute the following commands:
 - **cd \$HOME/csce-231**
 - **tar -xvzf bootloader_csce_231.tgz**
27. Open your home start menu in your guest, and search using the keyword "code" and you will find "Visual Studio Code" application. Right click on it and choose add to panel.
28. Now start from the taskbar Visual Studio, and open the folder containing the boot loader code.
29. To test the boot loader, from the same terminal as above, execute the following commands:
 - **cd bootloader**
 - **make run_hello**And you should have Qemu started with your test boot loader.
30. Now you are all set up, and you can navigate using VSCode to inspect sources/boot_hello.asm

We will go through the code organization and the Makefile in the lab.

2. Modifying the boot loader

Now, the boot loader written and documented thoroughly will only work from within Qemu. It will definitely not work on a native bare metal hardware. There is something missing in the provided boot loader that you are requested to investigate and add. After applying the amendments you need to create a bootable flash drive with your new boot loader and boot a real computer from it.

What to submit

1. Your full in-line documented boot_hello.asm of the part you have added to the boot loader;
2. A PDF report that includes:
 - a. A detailed description of your investigation.
 - b. The steps needed to create a workable flash drive.
3. A read me file indicating how to compile and test your code.

How to submit:

Compress all your work: source code, report, readme file, and any extra information into a zip archive. You should name your archive in the specific format <Student_ID>_<Name>_Assignment5.zip. Finally, upload your code to blackboard.

Grade

This assignment is worth 3% of the overall course grade. The assignment will be graded on a 100% grade scale, and then will be scaled down to the 3% its worth. The grading of the assignment will be broken down as follows:

1. 10 % for just submitting a meaningful assignment before or on the due date. This 10% does not account for the correctness of your assignment but submitting an empty assignment without code will definitely result in losing this 10% and consequently the whole grade of this assignment.
2. 65 % for the correctness and the quality of your code.
3. 25 % for the quality of your inline documentation, the report, and the readme file.

Delays

You have up to 2 working days of delay, after which the assignment will not be accepted and your grade in that case will be ZERO. For every day (of the 2 allowed days), a penalty of 10% will be deducted from the grade. And of course you will lose the 10% mentioned in point 1 above under the “Grade” section.