1. **FOR EACH MEMBER IN YOUR TEAM, PROVIDE 1 PARAGRAPH DETAILING WHAT PARTS OF THE LAB THAT MEMBER IMPLEMENTED / RESEARCHED. (YOU MAY SKIP THIS QUESTION IF YOU ARE DOING THE LAB BY YOURSELF).**

Shweta -

1. Tried unsuccessfully on macbook using virtualBox. In Virtual Box, I was not able to enable vmx features. So the output was wrong.
2. Then I successfully cloned linux source code and built the kernel from that source code using VMware fusion on the same machine. This time I was able to enable the vmx features and get the correct output after inserting the 283_1.c module. Then I edited the file to detect other MSRs and check the 55th bit of IA32_VMX_BASIC MSR.

Radhika -

Set up the environment in Windows using VMWARE workstation player and installed Linux Ubuntu 17. Built the kernel from source after 4 attempts of unsuccessful building the kernel due to various issues. such as space, locked VM,etc. Researched about MSRs to be read in the SDM and helped to modify and execute the code.

Discussed and wrote the steps in this PDF file.

2. Describe in detail the steps you used to complete the assignment. Consider your reader to be someone skilled in software development but otherwise unfamiliar with the assignment. Good answers to this question will be recipes that someone can follow to reproduce your development steps.

Note: I may decide to follow these instructions for random assignments, so you should make sure they are accurate.

Step 1:Environment Set up

1. We installed Vmware Fusion in Mac. Downloaded the ISO disk image file for Ubuntu Desktop from the link https://www.ubuntu.com/download/desktop
2. Created new virtual machine by allocating 100GB storage and 4GB RAM for the virtual machine.
3. Once above steps were done we built the linux kernel using the following:

Install Git:

```
sudo apt-get update
sudo apt-get install git fakeroot build-essential ncurses-dev xz-utils libssl-dev bc
```

Check version of current linux kernel

```
Uname –r
```

Clone the Git repository for the latest linux kernel source code :

```
Git clone https://github.com/linux.git
```

Note the COMMIT id
Git log [Note the first commit id, which will be of the form as below:
Record the head commit ID of your tree:

▪

For example, if the output of "git log" shows the following:

```
commit 89970a04d70c6c9e5e4492fd4096c0b5630a478c
Merge: 806276b7f07a 3ea3217cf918
Author: Linus Torvalds <torvalds@linux-foundation.org>
Date:   Wed Mar 29 19:59:49 2017 -0700
```

▪

.. you would record "commit 89970a04d70c6c9e5e4492fd4096c0b5630a478c"

Change to the Linux folder and configure the modules to be included/excluded

```
make menuconfig
```

Compile the kernel and its modules by checking number of processing units available

nproc [to know the number of processing units, in our case it was 1]

```
sudo make -j 1 && sudo make modules_install -j 1 && sudo make install -j 1
```

Once the kernel is built, use the following command to automatically look for the /boot folder and adds them to the grub's config file

```
update-grub
```

Check version now so that it shows the latest kernel version

```
Uname –r
```

Step 2: Inserting the module into the kernel

Create folder named "xx" on desktop and save the files cmpe283-1.c and Makefile from files section of Canvas provided by the professor. Modify the file cmpe283-1.c to implement the assignment functionality as below:

- Determine if your CPU supports VMX true control. Enable those features from settings.

- Based on the above, read various MSRs to ascertain support capabilities/features

- Entry / Exit / Procbased / Secondary Procbased / Pinbased controls
  • For each group of controls above, interpret and output the values read from the MSR to the system
- via printk(..), including if the value can be set or cleared.

We need to create the makefile to compile the module. Read online about how to create the same. However, in this assignment, the makefile is present in the Canvas.

Use the below command to compile

```
Cd  xx
make
```

Insert the module into the kernel

```
Insmod cmpe283.ko
```

Then to generate the git diff for submission, do -

```
Git add .
Git commit -m “some message”
git diff HEAD~1 > cmpe283-1.diff
```

This generated the diff file.