# Project 1

Ty Skelton

### Abstract

The first project of Operating Systems 2. This assignment is meant to introduce us to working with the linux kernel and writing concurrent programs in C.

CS 444

#### I. BOOTING THE KERNEL ON THE VM

## A. Log of Commands

1): Acquiring a local copy of the Kernel by running

```
$ git clone git://git.yoctoproject.org/linux-yocto-3.14
```

2) : Copying over all the necessary files into the root of my linux tree:

```
$ cp /scratch/spring2015/files/config -3.14.26-yocto-qemu ./.config
$ cp /scratch/spring2015/files/bzImage-qemux86.bin ./
$ cp /scratch/spring2015/files/core-image-lsb-sdk-qemux86.ext3 ./
```

3) : Building the kernel:

```
make -j4 all
```

4) : Writing a run script:

```
#!/bin/bash

source /scratch/opt/environment-setup-i586-poky-linux

qemu-system-i386 -gdb tcp::5618 -S -nographic -kernel bzImage-qemux86.bin \
-drive file=core-image-lsb-sdk-qemux86.ext3, if=virtio \
-enable-kvm -net none -usb -localtime --no-reboot \
-append "root=/dev/vda_rw_console=ttyS0_debug"
```

5) : Running the script for the first time:

```
$ chmod u+x run
$ ./run
```

6) : Creating the gdb initializer script:

```
target remote :5618
symbol-file linux-yocto-3.14/vmlinux
```

7) : Connecting gdb from another shell:

```
$ gdb
```

8) : After typing continue in the gdb instance, I was able to successfully login with the credentials of root.

## B. Qemu CLI Flags

-gdb tcp::5618 This flag will tell Qemu to open a gdb server on the following device. We specify to a reserved tcp port.

- -S This flag instructs Qemu to not start the CPU at start up and to wait for a continue from the device monitor. -nographic Normally Qemu displays output to VGA. With this flag it will bypass that entirely and spin up a headless command line application.
  - -kernel bzImage-qemux86.bin Specifies the particular kernel to use.
- -drive file=core-image-lsb-sdk-qemux86.ext3,if=virtio This flag specifies the drive to use, with some following options. The file option defines a disk image and the if option defines the type of interface the device is connected to.
  - -enable-kvm This flag enables full KVM (Kernel-based Virtual Machine) support.
  - -net none Instructs the VM that no network devices should be configured.
  - -usb Enables the USB drivers.
  - -localtime Sets the time to the localtime of the calling machine.
  - *–no-reboot* Exits rather than rebooting.
  - -append "root=/dev/vda rw console=ttyS0 debug" Sends command line arguments to the kernel.