Project One Writeup

Spring 2017

Shuai Peng (pengs), Anya Lehman (lehmana), Andrew Bowers (bowerand),

April 19, 2017

Abstract

This is our write up for the project one *Getting Acquainted*.

1 Command Log

1) To start we logged on to the os-class *ssh username@os-class.oregonstate.edu*
2) Then we used cd to get to the correct folder in scratch/spring2017 *cd /scratch/spring2017*
3) Then we made a group folder for us all to work in *mkdir 11-04*
4) Next we struggled to make said folder accessible to all of our gorup members by changing the permitions on it so not just the group member that cr eated the directory could work in it *chmod 777 11-04*
5) Then we called git clone to download the project from the GitHub account and we checked to make sure we got all the correct files *git clone git://git.yoctoproject.org/linux-yocto-3.14*
6) Then we switched to the tag we needed by using cd again and going into the directory that was cloned into our folder *cd linux-yocto-3.14*
7) Following this we checked out the v3.14.26 *git checkout v3.14.26*
8) Next came configuring the environment which we did by calling *source /scratch/opt/environment-setup-i586-poky-linux*
9) Then we made a kernel instance for our group
10) Then we coppied in the files that lets us configure *cp /scratch/spring2017/files/config-3.14.26-yocto-qemu .config*
11) Then we ran *make menuconfig*
12) A window popped up
13) In this window we pressed / and typed *LOCALVERSION*
14) Next we pressed *1* and edited the value to be *-11-04-hw1* to make that the name of the kernel
15) Then we built our kernel with four threads by running *-j4*
16) Then we ran *cd ..* followed by *gdb*
17) Our next step was to move onto a different laptop and called *source /scratch/opt/environment-setup-i586-poky-linux* again
18) Then we made a copy for the starting kernel and the drive file located in the scratch directory by calling */scratch/spring2017/files/cor image-lsb-sdk-qemux86.ext3*
19) Then we tried running the starting kernel *qemu-system-i386 -gdb tcp::5604 -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"*
20) Since we previously ran the qemu in debug mode, we used gdb to control it so, back on the origonal computer, we connected the qemu by running *target remote :5604*
21) Then we rebooted the VM *reboot*
22) Then we tried running the kernel instance we had created *linux-yocto-3.14/arch/x86/boot/*
23) Then we ran *qemu-system-i386 -gdb tcp::5601 -S -nographic -kernel linux-yocto-3.14/arch/x86/boot/bzImage -drive file=core-image-lsb-sdk-q emux86.ext3,if=virtio -enable-kvm -net none -usb -localtime –no-reboot –append "root=/dev/vda rw console=ttyS0 debug"*
24) Finally we rebooted the vm and used *q* to quit