Group 15 Assignment 1 Report

Ryan Howerton, Jinfeng Li, Vincenzo Piscitello
CS444 Spring 2018

Abstract

In this first assignment for Operating Systems II, we made contact with our groups and prepared our group directory on the os2 server by preparing the kernel and virtual machine source code and setting up a Git repository. We copied several files to our repository, ran the Qemu virtual machine, and built the Yocto Linux Embedded Kernel. We kept a log of our commits to the aforementioned Git repository, beginning with the creation of the repository. Finally, we logged the work done to reach a final solution.

I. COMMAND LOG

cd /scratch/spring2018/
mkdir group15_444
cd group15_444
git clone –branch v3.19.2 git://git.yoctoproject.org/linux-yocto-3.19
cp /scratch/files/bzImage-qemux86.bin .
cp /scratch/files/core-image-lsb-sdk-qemux86.ext4 .
source environment-setup-i586-poky-linux
cd linux-yocto-3.19/
git checkout -b master
git commit -a -m "Initial Commit"
cp /scratch/files/config-3.19.2-yocto-standard .config
make menuconfig
cd ..
../bin/acl_open group15_444 howertor piscitev
../bin/acl_open group15_444 lijinf piscitev
cd /scratch/spring2018/
make -j4 all
qemu-system-i386 -gdb tcp::5515 -S -nographic -kernel bzImage-qemux86.bin -drive file=core-image-lsb-sdk-qemux86.ext4,if=virtio -enable-kvm -net none -usb -localtime –no-reboot –append "root=/dev/vda rw console=ttyS0 debug"
gdb -tui
target remote:5515
continue
git commit -a -m "Initial Commit"

II. QEMU COMMAND FLAGS

Command: qemu-system-i386 -gdb tcp::???? -S -nographic -kernel bzImage-qemux86.bin -drive file=core-image-lsb-sdk-qemux86.ext4,if=virtio -enable-kvm -net none -usb -localtime –no-reboot –append "root=/dev/vda rw console=ttyS0 debug"

- qemu-system-i386: Boot from image emulating i386 architecture
- -gdb tcp::???? -S: Boot in gdb debug mode with tcp port ???? and wait for gdb connection
- -nographic: Do not display video output
- -kernel bzImage-qemux86.bin: Use bzImage-qemux86.bin as kernel image
- -drive file=core-image-lsb-sdk-qemux86.ext4,if=virtio: Use core-image-lsb-sdk-qemux86.ext4 as drive and specify the controller's PCI address
- -enable-kvm: Enable KVM full virtualization support
- -net none: No network connection
- -usb: Enable the USB driver
- -localtime: Use local time as time
- –no-reboot: Exit instead of rebooting
- –append "root=/dev/vda rw console=ttyS0 debug": Use "root=/dev/vda rw console=ttyS0 debug" as kernel command line

III. GUIDED QUESTION ANSWERS

1) What do you think the main point of this assignment is?
   - The main point of the assignment is to prepare our group directory for future assignments and prepare us for working with the kernel and VM in the future.
2) How did you personally approach the problem? Design decisions, algorithm, etc.
   - We made sure we understood the final goal of the assignment, and then laid out a plan for how to achieve it. We decided that it would be better to gather all of the components of the VM and the kernel before attempting to test the individual parts.
3) How did you ensure your solution was correct? Testing details, for instance.
   - We built the kernel and ran the VM using the provided commands. It appeared that both of the parts ran correctly, so we are pretty sure that the solution is correct.
4) What did you learn?

- First, we learned a lot about working with Git locally. For example, we discovered that we were unable to follow that standard Git workflow with multiple people working in the same directory, because every time a user attempted to branch off of the main, every user in the directory would be switched to that branch.

  We also learned a great deal about how to manage a directory accessed by multiple people, specifically that not all users had access to files owned by someone who is not themselves. We spent a lot of time trying to get everybody on the same page, due to the fact that a lot of people working in the same directory caused necessary files to become disassociated (many files in the .git folder became owned by different people).

IV. VERSION CONTROL LOG

Commits
commit f0c5ea77098c2ded47d2b438509a628907489ca6   **Author:** Vincenzo Piscitello <piscitev@os2.engr.oregonstate.edu >   **Date:** Sun Apr 15 14:19:20 2018 -0700   Initialization Complete
commit 1222d7e6c7804d9a7232654df3906500f653c2b8   **Author:** Vincenzo Piscitello <piscitev@os2.engr.oregonstate.edu >   **Date:** Sat Apr 14 15:36:44 2018 -0700   Initial Commit
commit 660613d1a4e94144490850b6c3d350331860fac4   **Author:** Greg Kroah-Hartman <gregkh@linuxfoundation.org >   **Date:** Wed Mar 18 14:11:52 2015 +0100   Linux 3.19.2

V. WORK LOG

Wednesday, April 11th	Created group directory, copied all requisite files from /scratch/files. Set permissions on all folders and files in the directory to rwxrwxrwx..
Thursday, April 12th	Cloned the linux yoctoproject repository to the group directory and created a git repository in the new folder. Discovered we needed to retroactively set the permissions on new folders and files. Also discovered that if other people do work in the directory under this architecture, then ownership on important files change.
Saturday, April 14th	Reverted most previous work to fix issues pertaining to user permissions. Re-cloned the repository and copied all necessary files from one user's account. Then appriopiate permissions were given to other team members. Then the menuconfig and kernal were built to initialize the system.
Sunday, April 15th	Completed the homework document and tested the system to ensure it would build.