Environment Setup

Advanced Embedded Software Development

with **Dan Walkes**

University of Colorado **Boulder**

Learning objectives:

Introduction to Linux Workshop
Installing a Linux Build VM
VM Snapshots
Distributions and Packages
Ubuntu and root access

Lecture Backup Material

- Introduction to Linux Workshop
  https://nsdl.oercommons.org/courses/an-introduction-to-linux-2/view
  - Shells, Text Editors (vim section), Remote Connections, Filesystem, File Permissions, Optional Topics
- Bash Scripting Tutorials:
  https://linuxconfig.org/bash-scripting-tutorial-for-beginners
  https://ryanstutorials.net/bash-scripting-tutorial/bash-variables.php

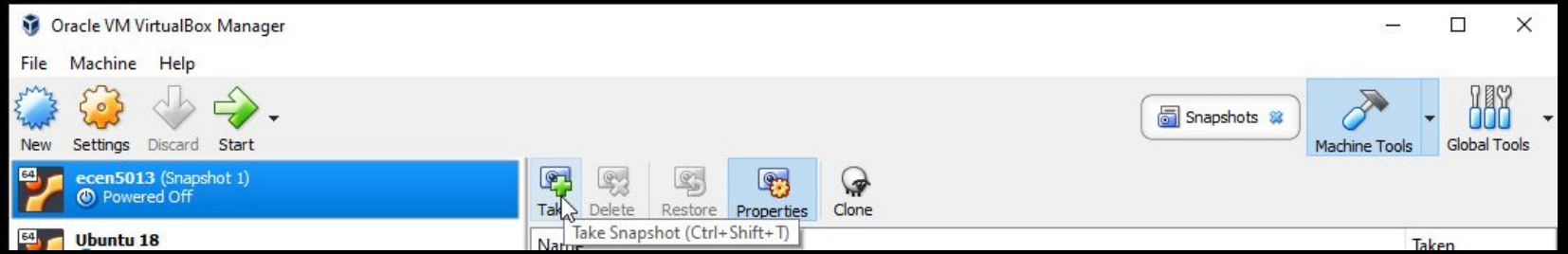
Installing a Linux Build VM

- If you are running Windows or MacOS, we will use VirtualBox for our Linux based build environment.
  - See https://www.virtualbox.org/
- We will use VirtualBox + Ubuntu.
  - See the Environment Setup document for details.

https://www.virtualbox.org/
http://releases.ubuntu.com/

Virtual Machine Snapshots

- Use these in case your VM crashes, to recover your last known working configuration



https://www.virtualbox.org/manual/UserManual.html#snapshots

Using a Physical Linux Machine

- If your laptop is not powerful enough to run VirtualBox you can also run on dedicated Linux hardware.
  - Does not need to be expensive - lots on ebay or in pawn shops for < $200
- You may also consider dual booting

Distributions and Packages

- Linux is available in many distributions.
  - Distributions are collections of preconfigured software maintained by the community or companies.
- Software version and configuration is managed in Packages.
- Ubuntu is a popular Linux distribution.
- Each distribution has its own mechanism of adding software packages through the command line.

https://en.wikipedia.org/wiki/Linux_distribution

Packages vs Binaries

- Why doesn’t Linux just use binary installers like Windows does? Why use package management?
  - Packages were essentially the precursor to “app stores”
  - Package managers provide a centralized location for trusted software sources.
  - Also solves fragmentation problems
    - Software packages may need slight customizations for different distributions.

What is Ubuntu?

- Pronounced “uu-boonto”
  - African word meaning “humanity to others”
- First release in 2004
- Based on another distribution known as “Debian”
  - Compared to Debian it’s (arguably) less stable but more up to date.
  - Ubuntu is more focused on usability than licensing.

https://ubuntu.com/about
https://linuxconfig.org/debian-vs-ubuntu

Ubuntu Package Manager

- Ubuntu's package management was traditionally performed by the "apt" utility
- Recent Ubuntu releases also support the "snap" utility
- Ubuntu will often suggest packages

```
ecen5013@ecen5013-VirtualBox:~/lecture2$ tree a_copy_target/

Command 'tree' not found, but can be installed with:

sudo snap install tree  # version 1.8.0+pkg-3fd6, or
sudo apt  install tree

See 'snap info tree' for additional versions.
```

https://help.ubuntu.com/community/AptGet/Howto
https://snapcraft.io/

Linux Root user

- A default user account has access limitations.
  - Prevents you from accidentally deleting/modifying things you didn’t intend.
    - example:
      - rm -rf /${undefined_variable}
- The “root” user can access/modify anything on the system
- Different permission levels allow the system administrator to configure appropriate permissions per user

Ubuntu Root Access Philosophy

- By default your account is able to run commands as root using the “sudo” (short for super user do) command.
- sudo <command name>
  - Asks you for your password
  - Runs the command you specified as root

https://www.howtoforge.com/tutorial/sudo-beginners-guide/
https://searchsecurity.techtarget.com/definition/sudo-superuser-do