# Overview

## Goals

In this lab you will create two Virtual Machines: A Debian ‘server’ VM and a CentOS ‘Workstation’ VM, and create your Raspbian SD card image for your Raspberry Pi. The Debian VM and Raspbian Pi will be command line only systems (aka headless). The CentOS VM will have a graphical interface.

VMs: A Debian ‘server’ VM and a CentOS ‘Workstation’ VM. The Debian VM should be command line only. The CentOS VM will have a graphical interface.

Raspberry Pi: Install an initial OS for the Pi and do initial configuration. Do NOT enable automatic GUI! The GUI can be started when (and only if) needed. Note the use of a GUI on the Pi will significantly slow the system.

For the upcoming labs the Debian VM and the Raspberry Pi will be mainly used as various servers, while the CentOS and the lab workstations will be used as clients.

## Deliverables

### Printed lab report

* Documentation of items marked with \*\*, any questions, and noted items
* A single printed lab report that contains:
  + A proper cover sheet
  + A copy of the information table from this sheet as well as the answers to the questions posed at the end of the lab
  + Description of what you did and any problems you encountered
* Copy of report(s) uploaded to Moodle for backup
* The Moodle copy ***will not*** be graded

Reminder: all report information should be copied from text files except when a screenshot is required!

### Information Table \*\*

|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Formula** | **Used in Lab** | **Answer** |
| Assigned Subnet (See Web) | *n* | - |  |
| Debian VM | 172.16.*(n)*.10 | DEBIAN\_IP |  |
| CentOS VM | 172.16.*(n)*.50 | CENTOS\_IP |  |
| Raspberry Pi | 172.16.*(n)*.20 | PI\_IP |  |

**Note:** You have been assigned an entire subnet of IP addresses for your 3110 labs. Your specific subnet should replace the (n) in the formula above and you should use this subnet for the remainder of the course. The subnet assignments are posted on Moodle and the TA has a copy. **Your Debian IP should be your subnet and must be .10 as well for your later DNS lab to work properly.**

# Lab

## Installations

Note: under **Debian VM** the \*\* show where CentOS values differ from the Debian one on steps 1 through 2.e

### Debian VM

1. Start Oracle VM VirtualBox from Applications->System Tools->Oracle VM VirtualBox
2. Create a VirtualBox install image file in your scratch directory
   1. Click on "New”
   2. Type in a name (unique for yourself)
   3. Ensure Type and Version are set to “Linux” and “Debian (64 bit)” respectively, click “Next”
   4. Leave the defaults for:
      1. Verify Memory size (512 MB), click “Next” \*\*
      2. Select *Create a virtual hard drive now*, click “Create”
      3. Select *VDI,* click “Next”
      4. Select *Dynamically allocated*, click “Next”
      5. Verify the File location and size data
         1. Click little green arrow by name box
            1. In *File System* select *scratch*
            2. Verity the name is still good
            3. Click “Save”
         2. Verify size as 8GB
         3. Click “Create”
   5. Click on “Settings” icon
      1. Select “System”
      2. Under Boot Order check “Network”
         1. All boxes should be selected at this point
      3. Click on Network in the left hand column
      4. On “Attached to:” select “Bridged Adaptor”
      5. Click “OK”
      6. *CentOS only: Return to CentOS VM install step 2*
   6. Click on the green Start Arrow
   7. Click “Start” button
   8. Debian installation window should show with “Install Debian 7.0 (64-bit)” selected
   9. Hit Enter to start installation
      1. Use defaults until the exceptions below
      2. At “Configure the network”
         1. You may change the Hostname from debian to one of your choice
         2. At “Domain Name” prompt set to “hades.lab”
         3. Use tab to highlight the <continue> selection and hit enter
      3. On next panel named “Choose a mirror…”
         1. Scroll to top and select “Enter information manually”
         2. Change mirror to “lab302-repo.hades.lab”
         3. Leave “/debian/” as is
         4. Leave the HTTP proxy entry blank
      4. Set root password to “rootpw”
      5. Create your userid
      6. Select Eastern time zone
   10. When asked about partitioning the disk
       1. Choose: “Guided – Use entire disk”
       2. Remember you are setting up the virtual disk here for the virtual machine
       3. Choose the defaults (just hit enter) for the rest of the partition questions
       4. Eventually you will be asked to write the changes to disks. Change from the default answer of <no> to <yes> and continue
       5. Wait, wait, and wait some more….
   11. Under “Software Selection,” uncheck “Desktop environment” and “Print Server.” The only item that should be checked is “Standard System”
   12. Any other questions you see can be left with their default value
   13. When the installation is finished, the VM will reboot
3. Log into the new Debian VM you just created with the ID you created in step above
4. Shut down the VM when you’re done
   1. Run /sbin/poweroff as root (su first, if necessary)

### CentOS VM

1. Follow steps 1 through 2.e above for CentOS but with the following changes:
   1. Use a different unique name for the name, eg. initialsCentOS or nameCentOS
   2. Ensure Red Hat is selected instead of Debian
   3. Under Setting System:
      1. Set Motherboard base memory to 4096MB
         1. To use at home do not make bigger than the memory on your home machine.
      2. Set Processor to 4 CPUs
         1. For home use do not make bigger that the cores your home machine has.
   4. Set the File Location and Size to 20 GB to allow more programs to be installed
2. Still in Settings:
   1. Under Storage:
      1. At Controller IDE click the “Circle +’ icon (add cd/dvd device)
      2. Click “choose disk”
      3. Select *file system* from the left column, then find /network-storage/iso/centos7 in the directory area
      4. Select the only CentOS item and “open”
      5. Under “Controller IDE” select the CentOS item
      6. Click on CD icon under Attributes and select the CentOS item there
      7. Click OK
      8. Click the green start arrow
   2. In the new window select Install CentOS 7, enter. Installation will begin.
   3. Select English, then continue
   4. Enable network in Network and Hostname, click done
   5. Set Installation Source
      1. lab302-repo.hades.lab/centos/
      2. Click Done
   6. Set Software Selection
      1. GNOME Desktop
   7. Installation Destination – open it and use the default
   8. Click “Begin Installation”
   9. While the software is installing set the root pw to “rootpw” and create a userid
   10. Wait, Wait, and Wait some more
3. The system will need to be rebooted
4. Accept the License
   1. Select “Done”
5. Scan the info on the KDUMP screen
   1. Click “Forward”
6. Logon to the system
7. Select English
   1. Next
8. Select English
   1. Next
9. Skip “Connect…”
   1. Next
10. Start Using CentOS Linux

#### CentOS Repository

After restarting or if trying to update/install packages you may need to update the package location information.

1. Open /etc/yum.repos.d/CentOS-Base.repo for editing
2. Comment out all the “baseurl=” statements, then add just after each one:

baseurl=http://lab302-repo.hades.lab/cenotos/Packages

Reminder: When switching to "superuser" don't forget the – after su. Otherwise you become root without the authority.

### Raspberry Pi

The Pi uses an SD card as its main storage where the OS and core programs reside. The Pi is designed to boot from the SD card. To do this the SD card must be properly configured to allow booting. Extra storage can be provided by a USB memory stick. The SD cards are easily removed and replaced which makes changing of OSes very easy.

#### Creating a bootable SD card

First step is to obtain a Linux Raspbian distribution image. One will be supplied to you for your first lab. That image, and others, may be found at: <http://www.raspberrypi.org/downloads> (if you want to do this at home). The image must be properly installed on your SD card. An SD card of at least 4GB is required for a small, modest system. Larger cards may be used if more main storage is required.

Get the image (a .img file) onto a usable workspace on your workstation, then create the bootable SD image. In the lab we will be using Linux to create the image. Look for a directory to download it. Note: Windows instructions are also included in this lab following the Linux instructions in case you want to do this at home on your Windows machine.

##### With a Linux Workstation in the Lab

Taken from the Raspberry Pi Users Guide pg 20. Note: must have root authority to for the fdisk and dd commands. Use a root terminal or su to get root authority for the following commands.

* Get a copy of the Raspbian distribution to your CentOS VM
  + Check with instructor or TA where the current copy is kept
* plug the SD card, with USB adapter, into your VM
* find the device name for the SD card by one of two ways:
  + fdisk –l
  + df –h
* You should find a listing entry similar to /dev/sdf1 with a size close to your SD card size
  + Note the first 3 characters of the last name
  + Note: the 3rd character will typically be in the range d to g.
* Change to the directory with the Linux .img file (by default will be your home Desktop)
  + Issue the following command:  
    dd if=*imagename*.img of=/dev/sd*X* bs=2M  
    Where:
    - imagename is the name of your Linux image
    - Replace the X in sdX with your letter as found above for your SD card device
      * IMPORTANT: only use the first 3 letters of the device name; the extra letters/numbers denote partitions within the device. If you use a parttion name the SD card will NOT boot
* Wait, wait, wait
* Safely unmount the SD card

Be sure to unmount the SD card before removing or you will lose data (umount /dev/sd*nn*). It may take some time for the copying and unmount to occur. Be patient.

##### With a Windows Workstation (at home)

Go to <http://sourceforge.net/projects/win32diskimager/> and download the binary Image Writer. Extract the zipped files. In the unzipped directory (wind32diskimager-binary) run the imager (Win32DiskImager.exe). Note the Image Writer can be used to write and copy SD card data. To write first put a blank SD card in your machine, note the drive letter assigned, then in Disk Imager:

1. In *Image File* find your Linux image
2. In *device* select the drive letter of your SD card
3. Click Write
4. Wait, wait, wait…

When done safely remove your SD card from the system.

#### Testing and configuring the Pi SD Card

Insert your SD card (face down) into the Pi and power the device. If you remembered to connect a display, in a few seconds you should see the system do its initial boot.

Do the following initializing steps in Setup Options:

* 1 Expand Filesystem
  + Highlight and hit <Enter>
  + Root partition will be resized
* 2 Change user password
  + Highlight and <Enter>
  + enter new password for the default pi id
  + Note: nothing will be echoed back
* 4 Internationalization Options
  + I1 Change Locale
    - Select and hit <Enter>, wait for data
    - scroll down and select en\_US.UTF-8 by hitting the space bar
    - Make sure no other Locale is selected (usually en\_GB.UTF-8 is default)
    - Tab until OK is highlighted and hit <Enter>
    - Select None, tab for OK and hit <Enter>
    - Wait for changes
  + I2 Change Timezone
    - Select
    - Highlight US, tab OK, <Enter>
    - Highlight Eastern, tab Ok, <Enter>
  + I3 Change Keyboard layout
    - Select your keyboard (Generic 101-key PC should be OK)
    - Tab OK, <Enter>
    - Select Other, <Enter>
    - Select English (US), tab Ok, <Enter>
    - Select English (US), tab Ok, <Enter>
    - Select Default, tab Ok, <Enter>
    - Select No Compose Key, tab Ok, <Enter>
    - Control+Alt+Backspace: tab <Yes>, <Enter>
    - Wait
* 8 Advanced Options.
  + A2 Hostname
    - Highligh, <Enter>
    - Read message, hit <Enter> again
    - Give the Pi a name meaningful to you
    - tab Ok, <Enter>
  + A4 SSH
    - Highlight, <Enter>
    - Select <Enable>, <Enter>
    - <Enter> for Ok
* Tab to <Finish>, <Enter>
* Select <Yes> for reboot
  + Wait for the Pi to reboot
* At login prompt the user id is pi, and use the password you entered earlier

## Configuring Static IP Addresses

**IMPORTANT:** These examples assume your network interface is *eth0.* Modify the examples accordingly if this is not the case. You can usually check the names of your interface by doing an *ifconfig*. Note that the name for the interface may be used as part of the configuration files also.

### Debian VM

1. Check if the network is up with ifconfig. If it is shut down the network. Use:
   1. ifdown eth0
   2. **This step is critical.**  If you skip this step, your IP address may revert randomly until the virtual machine is rebooted
2. Edit /etc/network/interfaces with either vi or nano
   1. vi /etc/network/interfaces
   2. nano –w /etc/network/interfaces

|  |
| --- |
| # Comment out the line below  #iface eth0 inet dhcp  # New definition of eth0. Lines are indented with tabs.  # Replace the IP address below with your Debian VM IP.  iface eth0 inet static  address **DEBIAN\_IP**  netmask 255.255.0.0 |

1. Bring networking back up
   1. ifup eth0
2. \*\* Check IP address
   1. /sbin/ifconfig

### CentOS VM

1. First, shut down the network. Use:
   1. ifdown eth0
   2. **This step is critical.** If you skip this step, your IP address may revert randomly until the virtual machine is rebooted.
2. Edit /etc/sysconfig/network-scripts/ifcfg-eth0

|  |
| --- |
| DEVICE=eth0  BOOTPROTO=none  # Do not change this line! It contains your VM’s MAC address  HWADDR=00:0c:29:??:??:??  ONBOOT=yes  NETMASK=255.255.0.0  # Replace the IP address below with your CentOS VM IP  IPADDR=**CENTOS\_IP**  TYPE=Ethernet  USERCTL=no  IPV6INIT=no  PEERDNS=no |

1. Bring the network back up
   1. ifup eth0
2. \*\* Check IP address
   1. /sbin/ifconfig

## Raspberry Pi

Raspbian is an ARM implementation of Debian. Use the Debian notes above to configure the Pi’s address.

### Explore the PI

Check out the directory structure, hostname, df –h, du –h /home, etc.

Note: the id pi is sudo enabled by default

sudo shutdown –F now

# Appendix

## Rooting

### Rooting a Linux Machine

Knowing how to root a Linux machine is a useful skill if something ever goes wrong or you forgot the root password. Only use this method on machines you own or are responsible for maintaining. Do not use this method on the physical lab machines.

#### Method One

1. This method boots the machine in single-user mode. Some Linux versions require the root password to boot single user mode (e.g. Debian).
2. Interrupt the boot loader
   1. Most modern Linux systems use GRUB as the boot loader
   2. Hitting the <ESC> key immediately after the BIOS is usually sufficient.
3. Select a entry to edit
   1. The default entry is normally the one you want to edit
   2. The ‘e’ key will edit an entry in GRUB
4. Edit the kernel boot parameters
   1. Often the line with the boot parameters are on a line that starts with ‘kernel’
   2. In GRUB, the kernel parameters are normally the second line
   3. The ‘e’ key will edit a line in GRUB
5. Append ‘single’ to the end of the kernel parameters
6. Boot the edited entry
   1. The ‘b’ key will boot an entry in GRUB
7. This will either:
   1. Drop you to a root terminal
   2. Ask for the root password before dropping you to a root terminal
8. Typing ‘exit’ will either completely boot the machine or will reboot the machine

#### Method Two

A slightly more complex method but will work anywhere where the boot loader is not password protected. Bypassing a password-protected boot loader is outside the scope of this course.

1. Follow Steps 1-3 of Method One
2. Append ‘init=/bin/sh’ to the end of the kernel parameters
   1. This overrides the normal call to ‘init’ after the kernel loads and calls a shell instead
3. Boot the edited entry
   1. The ‘b’ key will boot an entry in GRUB
4. Remount the root file system read-write
   1. mount –o remount,rw /
   2. This is necessary to make changes
   3. You may have to mount other file systems
      1. mount –a
5. Perform any tasks you wish
6. When finished:
   1. Unmount any mounted file systems
      1. umount –a
   2. Mount the root file system as read-only
      1. mount –o remount,ro /
   3. A hard-reset of the computer is required

## Secure Shell basics

Secure shell is useful for remote access to a computer (VM or otherwise). For use in the lab, it is extremely useful to SSH to a VM so you can see a larger window as well as copy and paste between the VM and the host computer. Secure Shell also allows you to copy files between machines: another extremely important feature when working with VMs.

* Installing SSH
  + aptitude –R install ssh (Debian)
  + yum install openssh (CentOS)
* Connecting to another machine
  + ssh <userid>@<ip address>
  + ssh jwatso8@10.3.254.7
* Copying files
  + scp <userid>@<ip address>:<source> <destination>
  + scp <source> <userid>@<ip address>:<destination>
  + scp jwatso8@10.3.254.7:/etc/syslog.conf syslog.conf
  + scp syslog.conf jwatso8@10.3.254.7:/etc/syslog.conf

## NooBs

If you have an extra 8GB SD card you might want to try NooBs (New Out Of Box Software. It has 6 OS in one package (last count as of August 2013). When first started it will give you the option to load a Linux variant to the SD card’s main partition. Every boot after that will then start that OS. At boot time there is an option to replace the OS. The old is wiped out and replaced with the new. When I tried it in June it had some bugs. Then it only had 4 variants, and two of them did not boot cleanly.

## Debian VM file sharing with Workstation

The headless workstation can share files with a directory on the workstation:

1. Open Virtual Box
2. Select the Debian VM – do not start it yet
3. Open *Settings*
   1. Select *Shared Folders*
   2. Create a shared folder by clicking on the little green folder with the + sign (on the right)
   3. Click on the dropdown arrow in the *Folder Path*
      1. Click *Other…*
      2. Find and click *File System*
      3. Click on *scratch* then *Open*
      4. Give the folder a name so you know it is yours
      5. Click Auto-mount
      6. Click OK
4. Start the VM
   1. Switch to the /media directory and check the contents
   2. There should be a *sf\_yourname* entry
   3. You are good to go!

The directory in /media on your Debian is now available on the /scratch directory on the workstation. You may change directory names as appropriate for your authority.