Jumbo

Last Revised: August 21, 2018

Contents

1 Introduction

Jumbo is a virtual machine running Ubuntu Linux that you may find useful for several of my courses (CIS-3010, CIS-3012, CIS-3152, CIS-4230/5230, and CIS-4250/5250). It contains various development packages and related software preinstalled.

You may also use resources on remote Linux servers (such as lemuria or the cluster attached to lemuria) for some of the supported courses. However, accessing those resources may be difficult in some cases, due, for example, to network issues. Also Jumbo provides a full, graphical desktop experience whereas the remote Linux servers are typically accessed only via a shell.

This document describes how to set up the preconfigured Jumbo virtual machine. I describe how to install the VirtualBox VM software, how to import and run the Jumbo virtual machine, and how to exercise the system to verify that it is set up properly.

2 Installing VirtualBox

The lab machines should already have the VirtualBox VM software installed. If you are using a lab machine you can skip this section and continue with Section 3. If you intend to run Jumbo on your personal machine you will need to first install VirtualBox on your system.

Jumbo consumes significant resources. Before trying to install it on your personal machine you should be sure you have at least 6 GiB of memory (8 GiB is recommended) and at least 50 GiB of free disk space. You also need to have hardware virtualization support turned on in the BIOS of your host computer. Most new machines come with this feature turned on by default. Older machines may require you to enter the BIOS and activate this feature. Finally, Jumbo assumes that your host has at least two cores.

If the host does not meet the requirements above it may be possible to reconfigure the virtual machine so that it will work anyway. However you may have to sacrifice some features or endure suboptimal performance in that case.

If you still wish to proceed, begin by downloading VirtualBox from `http://www.virtualbox.org`. Be sure to download both the main installer for your system *and* the Extension Pack. The Extension Pack adds important functionality that Jumbo requires. *The virtual machine will likely not boot in its default configuration without the Extension Pack installed!*

Run the VirtualBox installer. After the installer completes double click on the Extension Pack to install that as well.

3 Booting the Guest

Jumbo is distributed as a virtual appliance. This is a standard file format that includes not only the virtual hard disk (compressed) but also the virtual machine's configuration. Once you import the virtual appliance into your virtualization software you can boot the virtual machine by just starting it as you might start a real computer.

The virtual appliance file format is accepted by several different virtualization products. However, Jumbo assumes is is running under VirtualBox, and, in particular, Jumbo has the VirtualBox "guest additions" pre-installed. The guest additions are a collection of software components that are loaded into the guest and that streamline the operation of the system.[1]

You may be able to run Jumbo using some other virtualization product. However, you will probably have to do some additional configuration before it will work well. VirtualBox is the only virtualization software that is officially supported for use in my classes.

To import the virtual appliances into VirtualBox follow the steps below.

1. Download the file `Jumbo-YYYY-MM-DD.ova`. See my web site for the locations where this file is currently available. The name of the file contains the date when that version was released. I normally release a new version of Jumbo shortly before the start of each semester.
2. Start VirtualBox and select "Import Appliance" from the File menu. Follow the prompts. This will unpack the OVA file and add the virtual machine to VirtualBox's start menu.

In principle no further configuration is necessary since the configuration of the virtual machine is contained in its original OVA file. However you might review the machine settings and tweak them if desired. Note especially the amount of memory allocated to the virtual machine. *Jumbo is configured to use 3 GiB of memory. If your host computer has less than 6 GiB of memory you may want to adjust the configured value downward.* If your host computer is well endowed with memory you might consider increasing the amount available to the virtual machines. A good rule of thumb is to allocate no more than 50% of your system's physical memory to all running virtual machines taken together.

After you have imported the appliances you can delete the OVA file to save disk space. However, if you have sufficient disk space you might consider keeping the file in case you need to reinitialize Jumbo from scratch. Having the OVA file on hand will save you another long download.

Jumbo has a user account named "student" with a password of "frenchfry." You should log in as this user. The student user can use `sudo bash` to create a root shell when necessary. All of the software preinstalled on Jumbo is open source. You may distribute the Jumbo OVA file freely to anyone.

4 Snapshots

One major benefit to doing your development inside a virtual machine is that you can use VirtualBox to "snapshot" your system just before attempting any kind of dangerous or complicated operation. When

[1] The term "guest" refers to the system running inside the virtual machine. The term "host" refers to the system running the virtualization software.

you create a snapshot VirtualBox remembers the entire state of the system. *Any* change made after the snapshot is provisional. If the system becomes corrupt, you can just restore to the snapshot and undo all changes made since the snapshot was taken.

The undoing of changes is complete. The process does not rely on the correct operation of the guest system. Even if the data on the (virtual) hard disk is totally shredded, restoring to a snapshot will reset every detail of the system back to the state it had when the snapshot was taken.

With this protection in place you are free to experiment without concern of causing irreparable damage. For example, if you want to try building and installing a new version of the C runtime library... go ahead! Take a snapshot first, and if the result is a major disaster you can just roll back to where you started and try again. In the worst case scenario you could delete your virtual machine and re-import it from the original OVA file. Of course this rolls back all changes you ever made to the system but the base configuration will be fully restored.

5 Basic Testing

It is nice after installing Jumbo to do some simple operations to verify that it is working for you in a useful way. Keep in mind that none of the steps described in this section are necessary for Jumbo to work. They are only intended to give you an opportunity to exercise the system in a relevant manner.

1. If any virtual machine is running, shut it down. If you are not running VirtualBox, start the VirtualBox application. Either way you should be looking at VirtualBox's window that lists your configured virtual machines, but you should have no virtual machine actually running.
2. *TODO: Finish me!*

When you are done with the steps above you can shutdown Jumbo using the usual method. After the system shuts down you can restore to the snapshot to undo any changes made during this session and put the system back into its initial pristine state. You may find it useful to do this after each experiment.

6 Making Backups

Most recovery operations such as restoring to a snapshot or re-importing the original OVA file will entail the lose of some or all of your work. Thus I strongly recommend that you back up your work often. This can be done by using a special backup script I created. To use it simply execute `backup` at a command prompt.

It is extremely important to understand that the `backup` script only archives the files in the course folder you specify (course folders are immediately inside student's home folder). This means the backup archives may be relatively small but it also means that any files you put elsewhere on the system will not be backed up. *I therefor strongly recommend that you store all course materials in the course folder when possible.* Be aware that configuration changes you make, for example to the desktop, are not backed up.

Once the `backup` script has created the archive it will ask you if you want to tranfer that archive to `lemuria`. Assuming you have an account on that system you can use `lemuria` as a repository for backups. If you do not transfer the backup file, the script will leave it in student's home directory where you can transfer it some other way (for example as an email attachment using webmail).

You should definitely transfer the backup archives off the virtual machine. The point of the backups are to save your work in case the VM is destroyed. Keeping the backup archives on the VM won't help you if you lose the state of the VM itself.

Figure 1 shows a sample backup session. Text entered by the user is show in an italic font.

Another, more elegant way to transfer a course directory to another host is to use the `rsync` program.

```
student@jumbo:~$ backup
1) None
2) cis-3010
3) cis-3012
4) cis-3152
5) cis-4230/5230
6) cis-4620/5620
=> 2
Creating backup file for cis-3010...
Done
Transfer to lemuria? [y/n] y
Username: pchapin
Password:
backup-cis-3010-2017-08-08.tar.gz                          100%   38KB  38.3KB/s   00:00
Transfer successful. Removing ~/backup-cis-3010-2017-08-08.tar.gz
student@jumbo:~$
```

Figure 1: Sample Backup Session

This program only copies files that have changed and is thus often faster than transferring an entire archive (even a compressed archive). The `rsync` program will add and remove files and directories on the target as necessary and, in archive mode, it even copies file permissions and date/time information. First move to student's home directory. Then do (replace the course folder as appropriate for your case):

```
$ rsync -vaz --delete -e "ssh -p 22" \
  cis-3010 username@lemuria.cis.vtc.edu:
```

The `--delete` option tells `rsync` to remove files on the target that are not in the source. The `-z` option specifies compressed mode; this is particularly useful if you are on a low speed network connection since it will reduce the amount of network traffic required. See the `rsync` manual page for more information.

Note that the `rsync` command above uses ssh as the underlying transport. Thus you will be prompted for your password, yet your password will not appear on the network unencrypted. Note also that the `rsync` command above will create (or update) a `cis-3010` directory beneath your home directory on the remote host.

7 Shutting Down

When you are finished using the virtual machines *do not just close the VirtualBox window!* Closing VirtualBox is equivalent to pulling the power on a real machine. Instead you should shut down the guest operating system properly. In the upper right corner of the desktop there is a menu with a "Shutdown" option.

Alternatively you can suspend the virtual machine. On the VirtualBox menu (not inside the guest) do "Machine → Close..." In the dialog box that appears select "Save the machine state." The next time you start the virtual machine will resume from the saved state. This is usually quicker than booting the system from scratch.