9:35 PM

Kernel

- The word **Linux** is often sloppily applied to the entire operating system and environment on computers which are equipped with a complete Linux distribution; but in fact, there are quite a few components which are necessary in order to have a fully functional platform
- Narrowly defined, Linux is only the **kernel** of the operating system (OS)
  - The kernel is the central component that connects the hardware to the software, and manages the system's resources, such as memory, CPU time sharing among competing applications and services
  - It handles all the devices that are connected to the computer by including so-called device drivers, and makes them available for the operating system to use
- A system running only a kernel has limited functionality, and the only place you will see that is in a dedicated device (often termed an **embedded device**) such as inside an **appliance**

9:36 PM

Operating System

In order to do something useful and to be able to do a variety of things as needs arise, you need some other components, which in and of themselves are not strictly part of Linux:

- Important **system libraries**
  - Usually these are **shared libraries** or **dynamic linked libraries**
  - They can be used simultaneously by more than one program
  - The most important one is **libc**, which is used by virtually every application (among other things it handles the communication between the applications and the kernel)
- Important **system services** (sometimes called **daemons**)
  - Started when the system runs to control and monitor activities on the system, e.g. networking, printing, disk maintenance, noticing when new equipment is plugged in, monitoring system load and performance, etc.

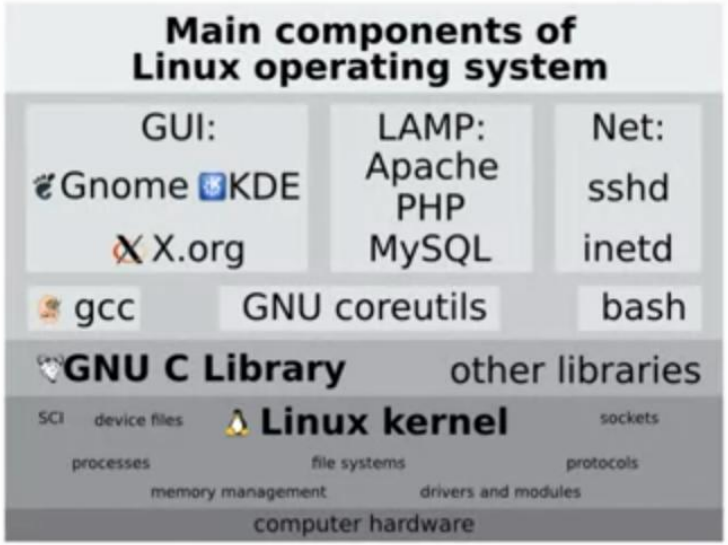
Operating System (Cont.)

And,

- Basic **system utilities**
  - Those which handle listing files, viewing them, renaming them, removing them, etc.; bringing network connections up and down; compressing and decompressing files, etc.
  - Many of these utilities (which are often simple ones) are needed by the services already mentioned
  - A particularly important program is the command **shell** program, which is what users interact with when they work at a command line, but which is also used by non-interactive scripts
  - The default shell for Linux is usually **bash**, which stands for **B**ourne **A**gain **SH**ell, since it is an extension of the older **sh**, or Bourne Shell program

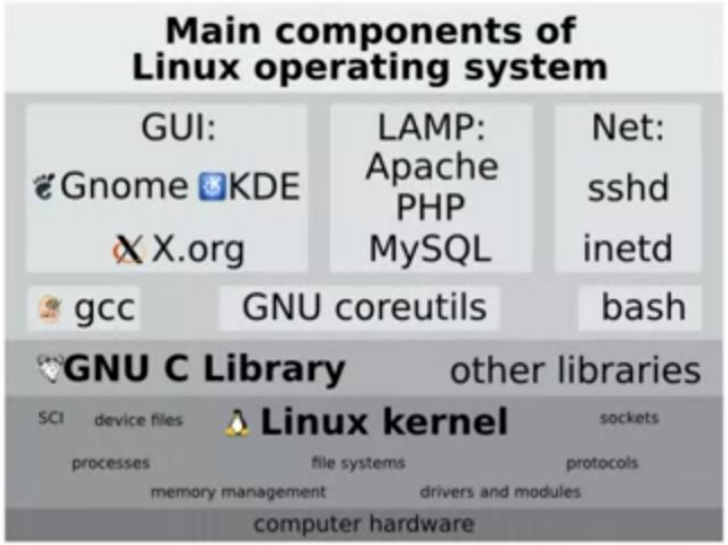
Main Components of Linux Operating System

Strictly speaking, the ingredients we discussed (kernel, including device drivers, services, and utilities) are enough to constitute a complete operating system, but other ingredients will be found on general service computers, as what we presented will only give you a **command-line based terminal**.



Main Components of Linux Operating System

Strictly speaking, the ingredients we discussed (kernel, including device drivers, services, and utilities) are enough to constitute a complete operating system, but other ingredients will be found on general service computers, as what we presented will only give you a **command-line based terminal**.



Graphical User Interface (GUI)

- Normal users will almost always be running a **Graphical User Interface** (GUI)
- Almost all desktop Linux systems will be built using the **X Window System** (or **X**) as the base of this interface; it has been around since at least 1984
- Besides **X**, there will be a so-called **window manager** which controls the appearances and behaviour of windows
- And a **desktop manager** which controls the entire desktop; the two most common choices in Linux are **GNOME** and **KDE**

Add-on Applications

- Many applications are part of the standard installation, but they are not part of the operating system - they are added on
  - The early versions of the Windows platform didn't contain many basic utility programs, and these had to be supplied by third-party providers until built-in implementations were deployed
  - The same is true of Internet browsers
- While many operating systems have now gotten so dependent on some of these programs being around that it is hard or impossible to operate without them, strictly speaking they are not absolutely required
- One advantage of Linux is that since there is so much choice for these components, they are used in a **modular way**, and the system can run with different or even no choices for them, unlike strictly controlled commercial platforms which either do not give such freedoms, or work hard to obscure and make it difficult to exercise options

9:40 PM

Add-on Applications

- Many applications are part of the standard installation, but they are not part of the operating system - they are added on
  - The early versions of the Windows platform didn't contain many basic utility programs, and these had to be supplied by third-party providers until built-in implementations were deployed
  - The same is true of Internet browsers
- While many operating systems have now gotten so dependent on some of these programs being around that it is hard or impossible to operate without them, strictly speaking they are not absolutely required
- One advantage of Linux is that since there is so much choice for these components, they are used in a **modular way**, and the system can run with different or even no choices for them, unlike strictly controlled commercial platforms which either do not give such freedoms, or work hard to obscure and make it difficult to exercise options

9:40 PM

Developmental Environment Tools

- Another important component is the **developmental environment tools**, such as compilers, debuggers, etc.
- While not all users need these, they are always available in Linux distributions

9:40 PM

Linux Distribution

- It is the role of the **Linux distribution** to bring all these ingredients together in a coherent way
- Since they all do this somewhat differently, the look and feel can be quite dissimilar even when everything is built on top of the same Linux operating system kernel

The Creation of Linux

- **Linux** was first announced to the world on August 25, 1991, by **Linus Torvalds**, then a 21-year-old student at The University of Helsinki, in a posting to a **Usenet** newsgroup
- Linus' original motivation was to write a terminal emulator for the purpose of accessing the University's **UNIX** servers
- He began using the **MINIX** operating system, but became frustrated by its licensing constraints (which limited it to educational use) and by the procedures for having changes made

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Message-ID: <1991Aug25.205708.9541@klaava.Helsinki.FI>
Date: 25 Aug 91 20:57:08 GMT
Organization: University of Helsinki

Hello everybody out there using minix –

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them 🙂

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes – it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(

9:47 PM

The Creation of Linux

- **Linux** was first announced to the world on August 25, 1991, by **Linus Torvalds**, then a 21-year-old student at The University of Helsinki, in a posting to a **Usenet** newsgroup
- Linus' original motivation was to write a terminal emulator for the purpose of accessing the University's **UNIX** servers
- He began using the **MINIX** operating system, but became frustrated by its licensing constraints (which limited it to educational use) and by the procedures for having changes made

From: torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
Newsgroups: comp.os.minix
Subject: What would you like to see most in minix?
Summary: small poll for my new operating system
Message-ID: <1991Aug25.205708.9541@klaava.Helsinki.FI>
Date: 25 Aug 91 20:57:08 GMT
Organization: University of Helsinki

Hello everybody out there using minix –

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them 🙄

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes – it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(.

Naming

- Originally, the name was intended to be **Freax** under which name the files were originally stored
- The name **Linux**, originally rejected by Linus Torvalds as being too egotistical, was adopted and implemented by a coworker, and Linus only consented to this later

1990s

- The original release 0.01 was released under its own license, which had restrictions on commercial activity; this changed in December 1992, when version 0.99 was released under the **GNU GPL**, under which it has remained ever since
- Linux (together with the GNU project components) filled a need in that the operating systems available for x86 computers were both proprietary and limited in ability, while the more robust **UNIX** variants were not available for personal computers
- The first really usable Linux distributions: **Slackware**, **Debian**, and the predecessor of **SUSE** were released in 1992; with **Red Hat** following in 1994
- In the meantime the number of contributors to Linux development grew from a handful to hundreds and then to thousands
- Version 1.0 of the Linux kernel was released in 1994, version 2.0 in 1996, Version 2.2 in 1999, Version 2.4 in 2001, and Version 2.6 in 2003

1990s

- The original release 0.01 was released under its own license, which had restrictions on commercial activity; this changed in December 1992, when version 0.99 was released under the **GNU GPL**, under which it has remained ever since
- Linux (together with the GNU project components) filled a need in that the operating systems available for x86 computers were both proprietary and limited in ability, while the more robust **UNIX** variants were not available for personal computers
- The first really usable Linux distributions: **Slackware**, **Debian**, and the predecessor of **SUSE** were released in 1992; with **Red Hat** following in 1994
- In the meantime the number of contributors to Linux development grew from a handful to hundreds and then to thousands
- Version 1.0 of the Linux kernel was released in 1994, version 2.0 in 1996, Version 2.2 in 1999, Version 2.4 in 2001, and Version 2.6 in 2003

2000s

- In 2000, the **Open Source Development Lab** (OSDL) was created as an independent non-profit organization with a mission to promote and optimize Linux, and became the employer of Linus Torvalds
- In 2007, it merged with the **Free Standards Group** to form **The Linux Foundation** whose base mission was to promote, protect, improve and standardize Linux
- Corporate involvement in Linux has steadily grown, and was accelerated by a one billion dollar investment by **IBM** in 2000
- Today, while there are many independent developers contributing to Linux, a large share comes from both major Linux distributors and major hardware and software industry players
- Linux has also grown to handle dozens of architectures, and systems from small embedded form factors to the vast majority of the world's supercomputers

The Differences

- Linux is only the kernel (everything else that makes up the full operating system is drawn from a number of sources); and it is *not* UNIX, although it's clearly UNIX-like
- UNIX and Linux have had very different evolutions:
  - UNIX was developed about 1969 by Thompson, Canaday, and Ritchie, and from the very outset it was designed to be a serious enterprise operating system
  - It grew up largely outside of the Intel family of CPUs, although it was later ported to it
  - By the time Linux first appeared in 1991, UNIX had already become quite fractured: there were many varieties, grouped in two major families; **System V** arising from the original code at Bell Labs, and **BSD**, arising from the University of California at Berkeley
  - Linux, on the other hand, began as a toy operating system only on the x86 architecture; it is doubtful anyone had any idea of how robust it would become or how many architectures it would wind up supporting

UNIX Variants

- Sorting out the fractious history and differences among the different UNIXs would be a lengthy task, but by 1991 there were many variants, often tied to a specific hardware platform and vendor:
  - SGI had IRIX
  - Sun had SunOS and Solaris
  - IBM had AIX
  - Hewlett Packard had HPUX
  - Cray had UNICOS
  - DEC had Ultrix
- Each one of these manufactures often had several varieties running even on its own universe of hardware
- SCO was one of the only variants not arising from a hardware company

UNIX Variants (Cont.)

- While there were various efforts to achieve some standardization, most vendors had strong self-interest in keep in things proprietary
- As a minimum there were always two major flavors to be considered, **System V** and **BSD**, and their behavior could be quite different even on quite basic matters such as signal handling
- Application developers interested in portability had to resort to the use of ugly **`#ifdef`** statements
- Even where the APIs were not that different, the actual implementation could be radically different from platform to platform
- Basic kernel architectures also differed
- Each platform had its own set of basic file utilities, shells, etc.

UNIX Variants (Cont.)

- While there were various efforts to achieve some standardization, most vendors had strong self-interest in keep in things proprietary
- As a minimum there were always two major flavors to be considered, System V and BSD, and their behavior could be quite different even on quite basic matters such as signal handling
- Application developers interested in portability had to resort to the use of ugly `#ifdef` statements
- Even where the APIs were not that different, the actual implementation could be radically different from platform to platform
- Basic kernel architectures also differed
- Each platform had its own set of basic file utilities, shells, etc.

GNU

- The **FSF**'s (Free Software Foundation) **GNU** (GNU's not UNIX) project developed freely-distributable versions of many basic utilities, such as **tar**, **ls**, **grep**, etc., and even more important, the **gcc** compiler and the basic C-library, **libc**
- Linux could not have been born or grown without the availability of tools from the GNU project
- We will stay out of the arguments about whether Linux should properly be loosely called GNU/Linux or something similar, but just note that what is often sloppily called Linux in fact contains many GNU components; properly speaking, Linux is only the kernel

GNU

- The **FSF**'s (Free Software Foundation) **GNU** (GNU's not UNIX) project developed freely-distributable versions of many basic utilities, such as **tar**, **ls**, **grep**, etc., and even more important, the **gcc** compiler and the basic C-library, **libc**
- Linux could not have been born or grown without the availability of tools from the GNU project
- We will stay out of the arguments about whether Linux should properly be loosely called GNU/Linux or something similar, but just note that what is often sloppily called Linux in fact contains many GNU components; properly speaking, Linux is only the kernel

Correlations

- While UNIX and Linux are not the same thing, Linux has always borrowed heavily from UNIX
- Most basic components of Linux, such as an **inode**-based filesystem, accessing hardware through device nodes, multi-process scheduling, process creation and destruction, are completely rooted in UNIX
- This is because the developers of Linux have always had a good footing in the UNIX world, and because of the availability of the UNIX tools from GNU and other non-GNU open-source projects
- Whenever possible, Linux has tried to accommodate both major variants of UNIX as far as the API is concerned; striving for POSIX type behavior is above that level

Correlations (Cont.)

- As such it has never been very difficult to port UNIX applications to Linux, unless the application has relied very heavily on certain idiosyncrasies of a particular UNIX implementation
- The open nature of the Linux development model has thus far avoided the serious fracturing that took place in UNIX
- It is perhaps ironic that the easily legal possibility of having Linux fork into competing versions at any time is perhaps what has prevented it
- Many hardware vendors now seriously support Linux on their hardware and the long range future of their own versions of UNIX is often in doubt
- It could well be that the Linux plan for world domination is inevitable; at least as far as killing off other UNIX-like operating systems

Variety of Distributions

- There are many Linux distributions, ranging from very widely used to obscure
- They vary by intended usage, hardware and audience, as well as support level

What Is the Role of Linux Distributions?

Some of the things Linux distributions do:

- Package up open software in a convenient fashion
- Form the bridge between the upstream developers and the end users
- Test software under more varied conditions than the upstream developers are able to; in particular, they find and resolve conflicts between components
- Push out updates, or patches, for security problems and bugs, in a timely fashion
- Provide a uniform packaging system that can resolve dependencies for software components, as well as aid system administrators in many ways
- Invest resources into the improvement of all system components, including the kernel and the desktop managers, by employing and funding many leading developers
- Represent the face of Linux to new users

Our Approach

- We have tried to keep this material as distribution-agnostic as possible; for all but the most specialized distributions this will not present any inconveniences
- Occasionally which distribution you are using will matter (e.g. when we must describe the location of particular files and directories or how to install certain required software packages; fortunately, modern distributions differ much less in these matters and we will rarely have to deal with such inconveniences
- This course has been developed for and tested thoroughly on the 64-bit versions of the three major families of Linux distributions:
  - **Red Hat** (includes RHEL and the CentOS and Scientific Linux systems based on RHEL, as well as recent Fedora and Oracle Linux)
  - **Debian** (includes Ubuntu and Linux Mint)
  - **SUSE** (includes openSUSE)

Other System Requirements

- If you are using a 32-bit system, almost everything will work acceptably, but we no longer make any effort to ensure it
- Generally, any recent (say last two) versions of these distributions are completely supported
- As far as software installation and control is concerned, distributions tend to use either **RPM**-based or **deb**-based package management
  - In our list, all Red Hat and SUSE derived systems are RPM-based, and all Debian derived systems are deb-based
  - When necessary, we will give required instructions for either of these two broad families

GENTOO and Arch Linux

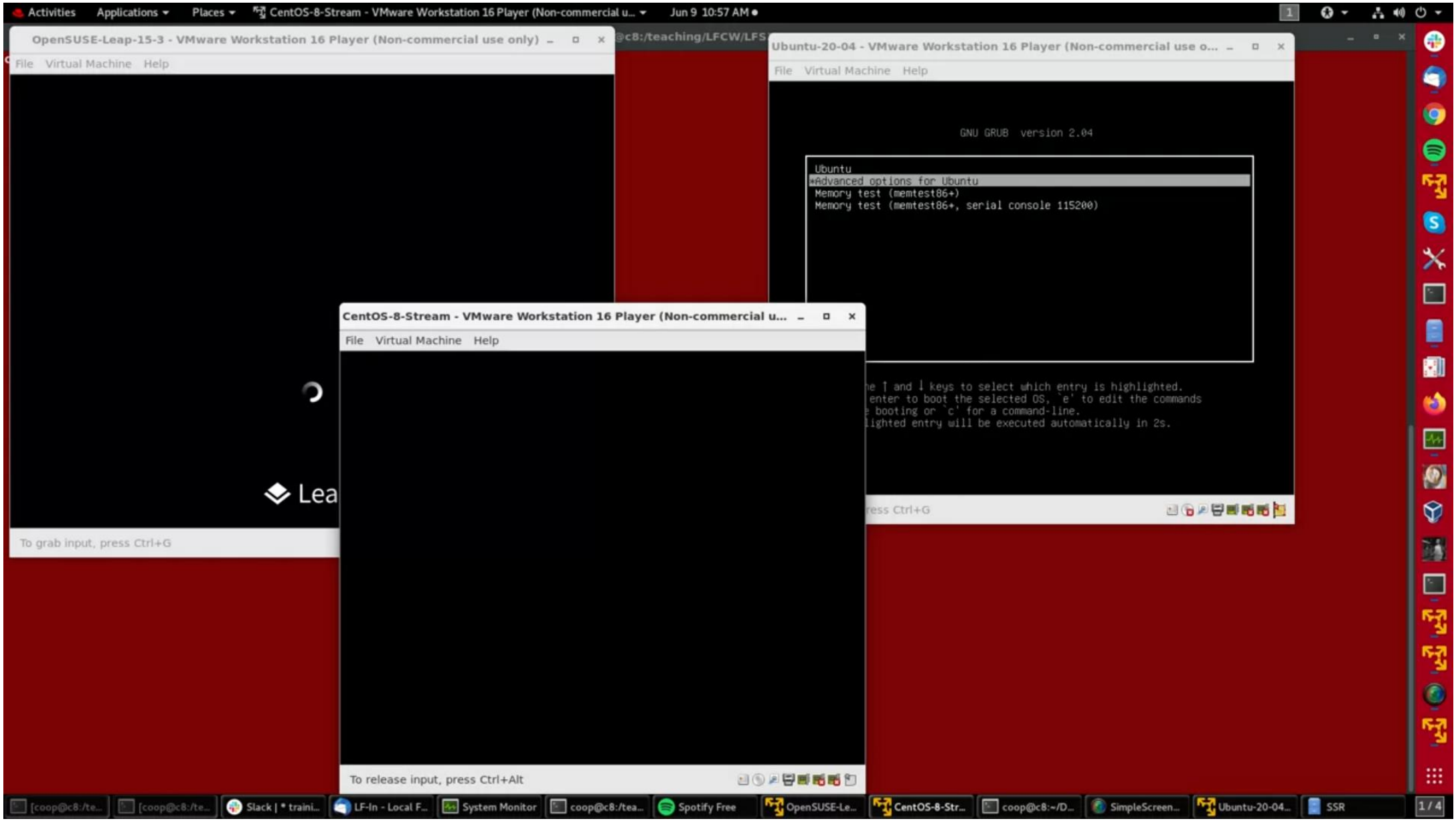
- GENTOO is based on neither of these packaging systems, and instead uses the portage/emerge system which involves compiling directly from source
  - If you are a GENTOO user, and you have successfully accomplished a fully functional installation (which is generally not a task for novices), you will not need detailed instructions in how to install software or find things
- Arch Linux is another cutting edge distribution which has its own packaging system, and uses a rolling release philosophy
  - Once again, if you are an Arch Linux user, you are already quite experienced and adjusting to the course to it should bring the most minimal quotient of pain
- If you are running any other distribution you should not have any trouble adapting what we are doing to your installation

Amazon Web Services (AWS)

- **Amazon Web Services** (AWS) offers a wide range of virtual machine products (**instances**) which remote users can access in the cloud
- In particular, one can use their **Free Tier** account level for up to a year and the simulated hardware and software choices available may be all one needs to perform the exercises for Linux Foundation training courses and gain experience with Linux
- Or they may furnish a very educational supplement to working on local hardware, and offer opportunities to easily study more than one Linux distribution
- For kernel-level courses there are some particular challenges

Amazon Web Services (AWS)

- **Amazon Web Services** (AWS) offers a wide range of virtual machine products (**instances**) which remote users can access in the cloud
- In particular, one can use their **Free Tier** account level for up to a year and the simulated hardware and software choices available may be all one needs to perform the exercises for Linux Foundation training courses and gain experience with Linux
- Or they may furnish a very educational supplement to working on local hardware, and offer opportunities to easily study more than one Linux distribution
- For kernel-level courses there are some particular challenges

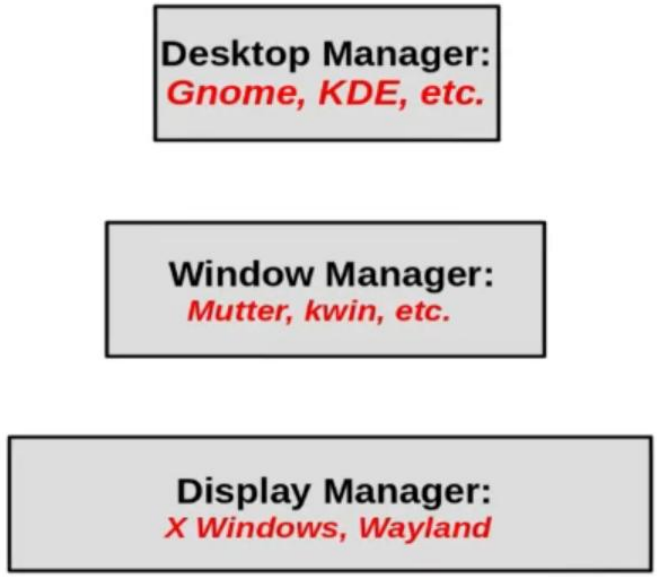

Activities
Applications
Places
CentOS-8-Stream - VMware Workstation 16 Player (Non-commercial u...
Jun 9 10:57 AM
OpenSUSE-Leap-15-3 - VMware Workstation 16 Player (Non-commercial use only)
File Virtual Machine Help
To grab input, press Ctrl+G
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use o...
File Virtual Machine Help
GNU GRUB version 2.04
Ubuntu
*Advanced options for Ubuntu
Memory test (memtest86+)
Memory test (memtest86+, serial console 115200)
CentOS-8-Stream - VMware Workstation 16 Player (Non-commercial u...
File Virtual Machine Help
To release input, press Ctrl+Alt
Slack | * traini...
LF-In - Local F...
System Monitor
Spotify Free
SimpleScreen...
SSR
1 / 4


Graphical Layers and Interfaces
THE LINUX FOUNDATION
0:07 / 6:20

Graphical Layers

The Linux graphical interface is actually composed of a number of layers, each of which having a choice of options. The three basic layers are:

- The X Window System
- The Window Manager
- The Desktop Manager



X Window System: Overview

- The **X Window System** (often called just X, Xorg or X11) has a long history in the UNIX world - its original versions can be traced back at least since 1984
- Since its inception, X was designed to handle displaying the results of activities on remote computers; at its roots it is fundamentally a communication protocol
  - This is unlike the graphical interfaces used in some other well-known operating systems, which were designed originally only to display programs running on the local machine, which may or may not have had network connections
- As far as the user experience, X's main function is to handle keyboard and pointer input, and handle showing the results on the screen in multiple **windows** (X is very strong at handling multiple screens, or terminals simultaneously)

Criticism

- A criticism of X sometimes heard is that it has a high inherent overhead because it works on a network paradigm, which also leads to a complicated design
- While this might have been true with early versions, modern implementations make very efficient use of **unix domain sockets**, **shared memory** and other optimizations, and the criticism does not hold weight anymore

Criticism

- A criticism of X sometimes heard is that it has a high inherent overhead because it works on a network paradigm, which also leads to a complicated design
- While this might have been true with early versions, modern implementations make very efficient use of **unix domain sockets**, **shared memory** and other optimizations, and the criticism does not hold weight anymore

Configuration

- There are many standard programs that ship with X, but the normal user will probably never have to interact with them directly, but will adjust properties and preferences through the graphical interface
- Almost all Linux distributions use the version of X that is supplied by X.org
  - The basic configuration file can be found at **`/etc/X11/xorg.conf`** and controls things like color-depth, display resolution, scanning rates, what pointers are available, which video card to use, etc.
  - If you are lucky, you will never have to touch **`xorg.conf`** directly; it is best to let it be generated by the installation program and then subsequently tweaked through graphical interfaces
  - In most modern Linux distributions this file has actually disappeared, and X is auto-configured on system start, although one can always substitute a custom file; sometimes, manual intervention is required (e.g. if you have recent or unusual hardware)

Wayland

- X is a rather old system; it dates back to the mid 1980s and as such has certain deficiencies on modern systems (with security for example) as it has been stretched rather far from its original purposes
- A newer system known as **Wayland** is expected to gradually supersede it and was adopted as the default display system in Fedora 25

Window Managers: Overview

- By itself, X has only limited functions; it does not control the exact placement and appearance of windows in the graphical interface
- This is the job of the **window manager**; some of its functions include:
  - Controlling the appearance of a window
  - Controlling pointer focus properties
  - Handling multiple desktops
  - Providing tabbed windows
  - Controlling visual effects
- This is not a complete list, and the line between window manager and desktop manager is not always well-defined
- Also, the window manager itself gives you the capability of altering many of these properties and different window managers can be tweaked to look very much alike

Window Managers for Linux

- There are a number of window managers available for Linux, and desktop managers have default choices:
  - For GNOME 3, the default is **mutter**
  - For KDE it is **kwin**
  - Other ones in use include **metacity**, **fvwm**, **fluxbox**, **enlightenment**, **sawfish**, and **xfwm**
- Some of the alternatives are very flashy, while some (such as **fvwm** and **fluxbox**) are very minimal, very fast, and work beautifully on limited hardware

Desktop Managers: Overview

- The **desktop manager** sits above X and the window manager; it is what the user is most likely to directly interact with
- The tasks of the desktop manager include:
  - Providing taskbars, menu bars, drop-down menus, and methods of configuring them
  - Offering applications (e.g. clocks, performance monitors, volume controls, etc.)
  - Enabling placing icons and program launchers on the desktop and/or the taskbar
  - Giving choices for themes, desktop backgrounds and wallpaper, screensavers, etc.
  - Providing methods for drag and drop functionality, etc.
  - Making it possible to save the desktop state from one session to another subsequent login

Desktop Managers: Overview

- The **desktop manager** sits above X and the window manager; it is what the user is most likely to directly interact with
- The tasks of the desktop manager include:
  - Providing taskbars, menu bars, drop-down menus, and methods of configuring them
  - Offering applications (e.g. clocks, performance monitors, volume controls, etc.)
  - Enabling placing icons and program launchers on the desktop and/or the taskbar
  - Giving choices for themes, desktop backgrounds and wallpaper, screensavers, etc.
  - Providing methods for drag and drop functionality, etc.
  - Making it possible to save the desktop state from one session to another subsequent login

Desktop Managers for Linux

- The most common desktop managers in use for Linux are:
  - **GNOME** (heavily dependent on the **gtk** set of graphical libraries)
  - **KDE** (built upon the **QT** libraries)
  - Others exist including **XFCE**
- Furthermore, some of the lightweight window managers, such as **fluxbox** and **fvwm** do not even require a desktop manager, as they have already have enough functionality to survive on their own
- Many distributions give you a choice of selecting one or the other desktop during installation; some let you choose both at installation or let you install the other at a later point
- Then you can choose which desktop manager you would like when you login

GNOME and KDE

- It is entirely possible to run KDE programs when running the GNOME desktop and vice versa, as long as the underlying libraries are installed; you can even drag and drop between the two, as the developers have tried hard to maintain interoperability
- If your distribution has done a sensible job of controlling dependencies on installing programs and their requisites, this should all be transparent to the user
- Because each window manager is so flexible, they can be made to look very much the same; for example, on RHEL systems, the default themes for GNOME and KDE look so much alike it can be hard to tell which one you are actually using

GNOME and KDE

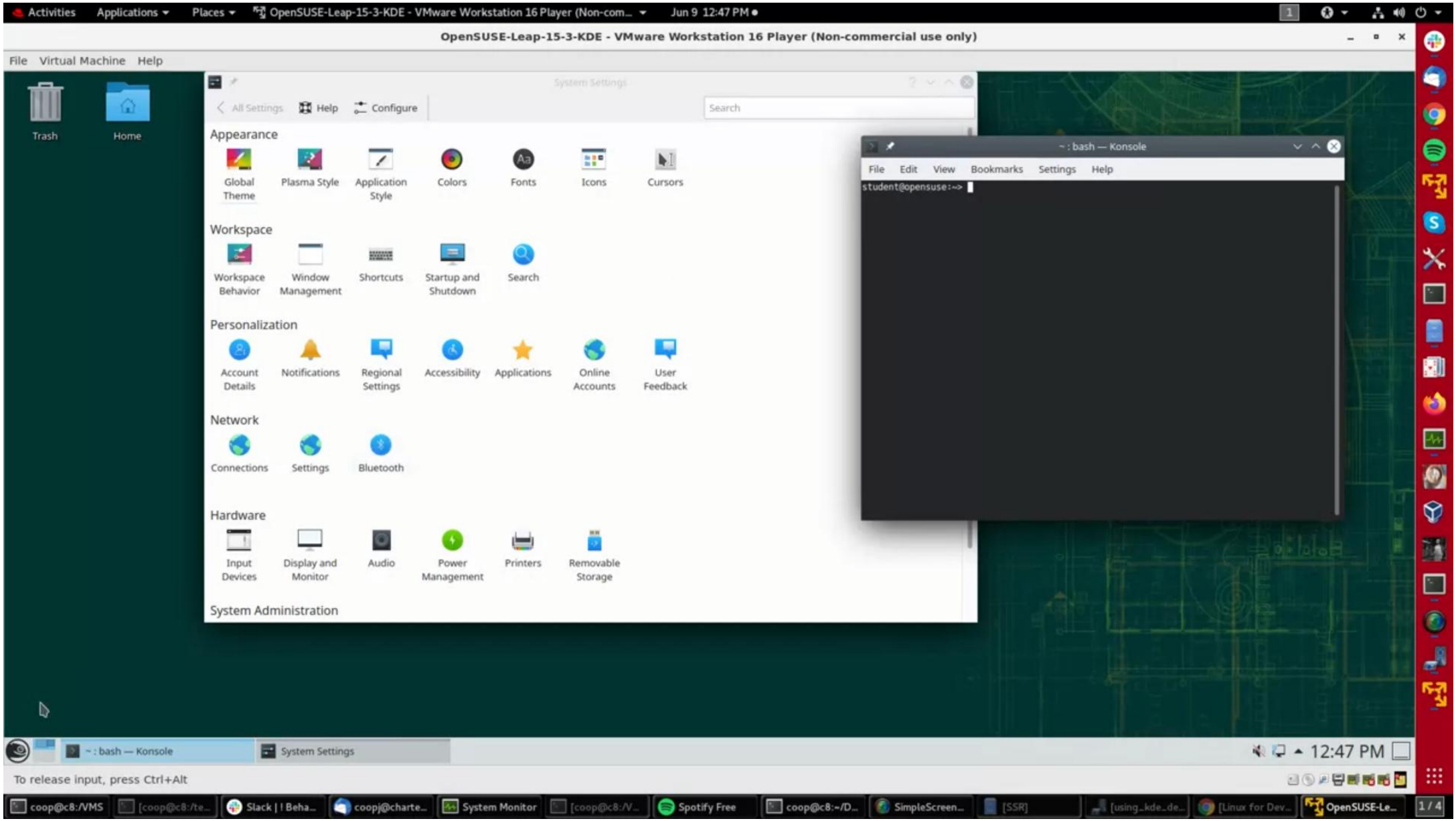
- It is entirely possible to run KDE programs when running the GNOME desktop and vice versa, as long as the underlying libraries are installed; you can even drag and drop between the two, as the developers have tried hard to maintain interoperability
- If your distribution has done a sensible job of controlling dependencies on installing programs and their requisites, this should all be transparent to the user
- Because each window manager is so flexible, they can be made to look very much the same; for example, on RHEL systems, the default themes for GNOME and KDE look so much alike it can be hard to tell which one you are actually using

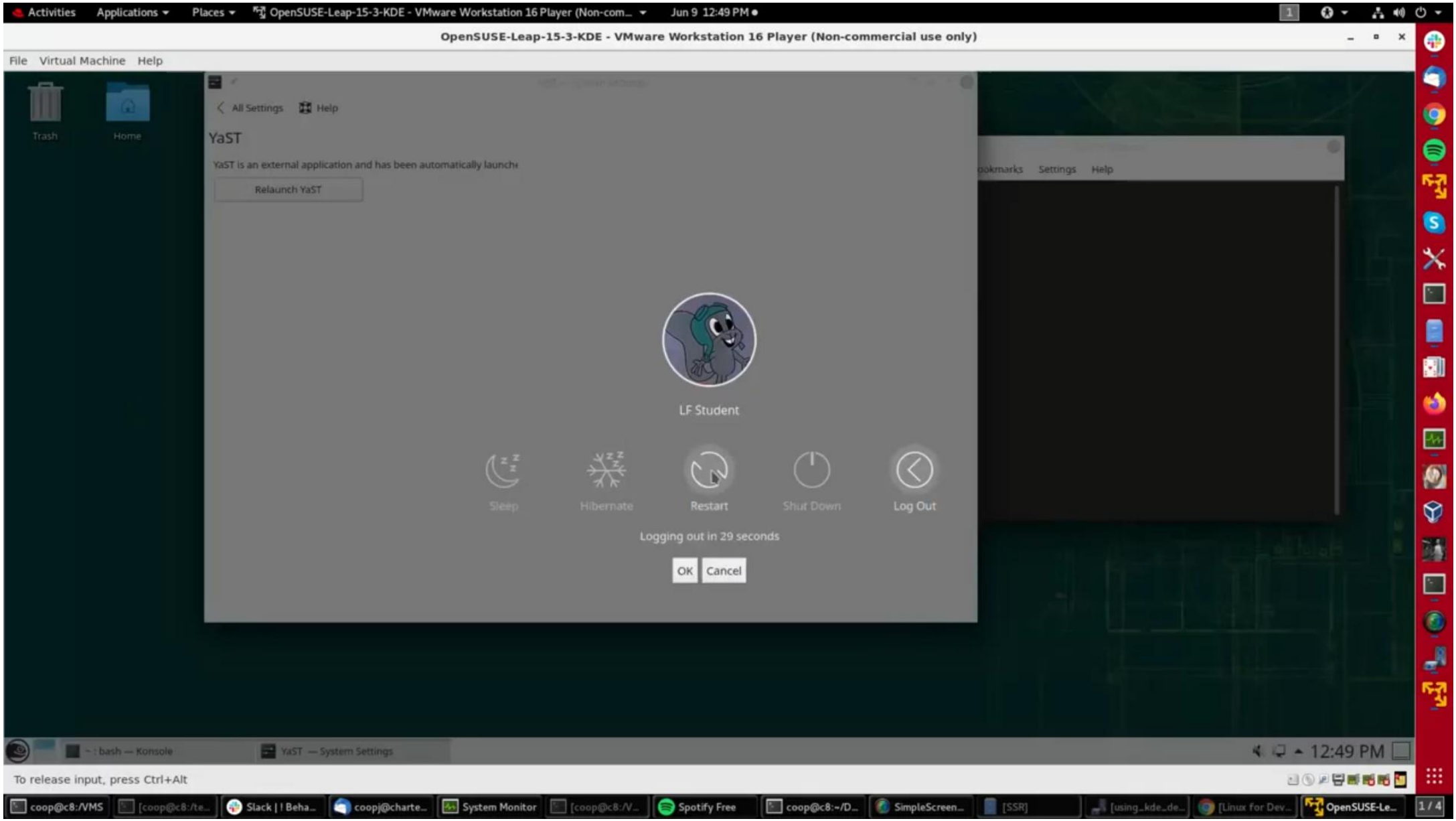
Terminal Window Options

- Linux users often take advantage of the command line to accomplish tasks
- While one can always bring up a one shot launch box (usually by hitting **`Alt-F2`** in most desktop managers), one needs a **terminal window** program to open up a window that can be used repeatedly
- The GNOME desktop contains the **gnome-terminal** program, which is very full-featured; it can be always be invoked from menus, which vary slightly between distributions, and also added to the *Favorites* panel
- If the **nautilus-open-terminal** package is installed on any but some of the most recent GNOME-based distributions, you can always open a terminal by right clicking anywhere on the desktop background and selecting *Open in Terminal* (this may work even if the package is not installed)

Terminal Window Options (Cont.)

- Menu items on the top bar make the presentation extremely flexible; right clicking anywhere in the terminal also provides configuration controls
- Extremely useful is the opportunity to use multiple tabs as in browsers, within the same terminal
- The KDE desktop contains the **konsole** program, which is similarly flexible; opening new tabs is termed as opening shells
- Desktop managers generally have a preferred applications menu choice in which you can pick the default terminal application
- The full-featured terminals, such as **gnome-terminal** and **konsole**, can be slower to display many lines of text, as compared to more ancient programs such as **xterm** or **rxvt**; however, in most modern systems this will not be noticeable, and the enhancements are well worth it


Activities
Applications
Places
OpenSUSE-Leap-15-3-KDE - VMware Workstation 16 Player (Non-com...
Jun 9 12:47 PM
OpenSUSE-Leap-15-3-KDE - VMware Workstation 16 Player (Non-commercial use only)
File
Virtual Machine
Help
Trash
Home
System Settings
All Settings
Help
Configure
Search
Appearance
Global Theme
Plasma Style
Application Style
Colors
Fonts
Icons
Cursors
Workspace
Workspace Behavior
Window Management
Shortcuts
Startup and Shutdown
Search
Personalization
Account Details
Notifications
Regional Settings
Accessibility
Applications
Online Accounts
User Feedback
Network
Connections
Settings
Bluetooth
Hardware
Input Devices
Display and Monitor
Audio
Power Management
Printers
Removable Storage
System Administration
~ : bash — Konsole
File
Edit
View
Bookmarks
Settings
Help
student@opensuse:~>
~ : bash — Konsole
System Settings
12:47 PM
To release input, press Ctrl+Alt
coop@c8:/VMS
Slack | ! Beha...
coopj@charte...
System Monitor
Spotify Free
coop@c8:~/D...
SimpleScreen...
[SSR]
[Linux for Dev...
OpenSUSE-Le...
1 / 4


Activities
Applications
Places
OpenSUSE-Leap-15-3-KDE - VMware Workstation 16 Player (Non-com...
Jun 9 12:49 PM
OpenSUSE-Leap-15-3-KDE - VMware Workstation 16 Player (Non-commercial use only)
File
Virtual Machine
Help
Trash
Home
All Settings
Help
YaST
YaST is an external application and has been automatically launch
Relaunch YaST
LF Student
Sleep
Hibernate
Restart
Shut Down
Log Out
Logging out in 29 seconds
OK
Cancel
Bookmarks
Settings
Help
~ : bash — Konsole
YaST — System Settings
12:49 PM
To release input, press Ctrl+Alt
coop@c8:/VMS
Slack | ! Beha...
coopj@charte...
System Monitor
Spotify Free
coop@c8:~/D...
SimpleScreen...
[SSR]
OpenSUSE-Le...
1 / 4

Sources of Documentation

- Linux is an open operating system which draws from a large variety of sources; as a result, the methods of getting help and obtaining documentation are myriad as well
- While the vast majority of the tasks one would pursue every day have extensive and well-made documentation organized in a relatively coherent way (already on your system), there are occasions where documentation is either missing or lacking in depth or clarity
  - Sometimes the developers associated with a project think everything is obvious
  - Sometimes it is simply that developers tend to push documentation and help facilities to the end of a project's development, and just never really get around to it before they move on to more exciting and new areas

Sources of Documentation (Cont.)

- A paucity of helpful documentation is the exception and not the rule, so we do not want you to think Linux is lacking in this regard; in fact, due to its open nature it is easy to argue that the help available is both more extensive and easily accessible than it is for other operating systems, particularly proprietary ones
- Linux distributors play an important role in making the local help information comprehensive and easy to use
- An essential task of the distributor is the **amalgamation** of material from many sources and presentation of it in a uniform way
- Commercial distributions offer technical support (for a fee) and depending on your circumstances you may find this a good value

Basic Command Line Tools

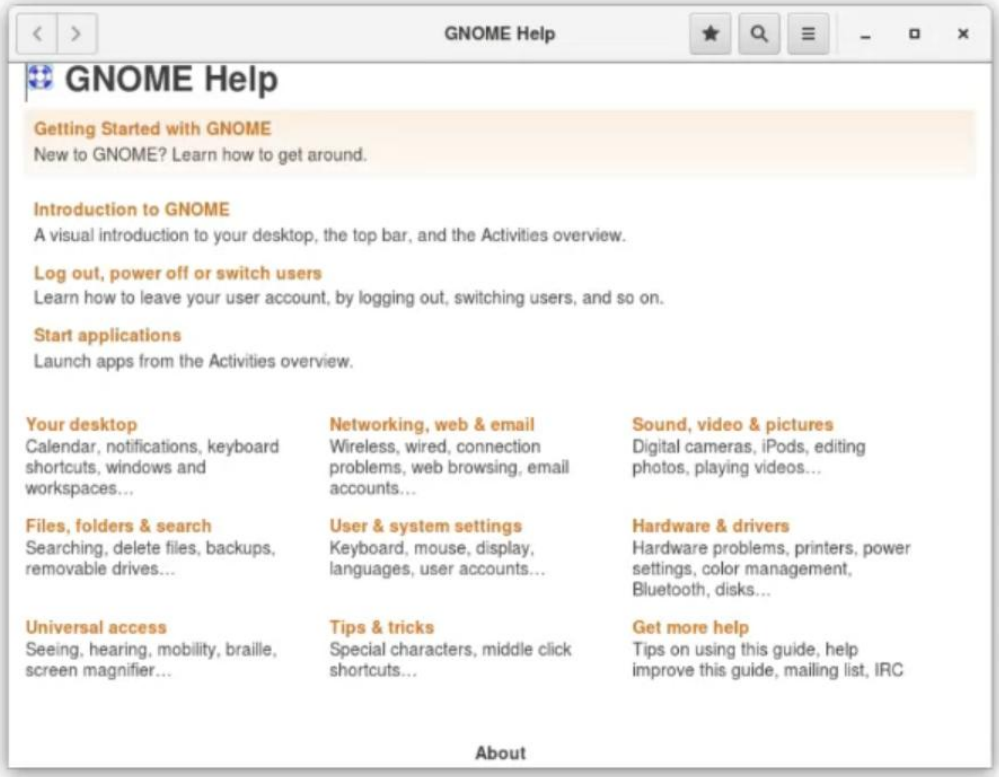
The basic command line tools that can be found on all Linux systems include:

- **man** Pages
- **info**
- **--help** and **help**

GNOME

If you are running a GNOME desktop you can invoke the graphical help system directly from the command line by typing:

```
$ gnome-help
```

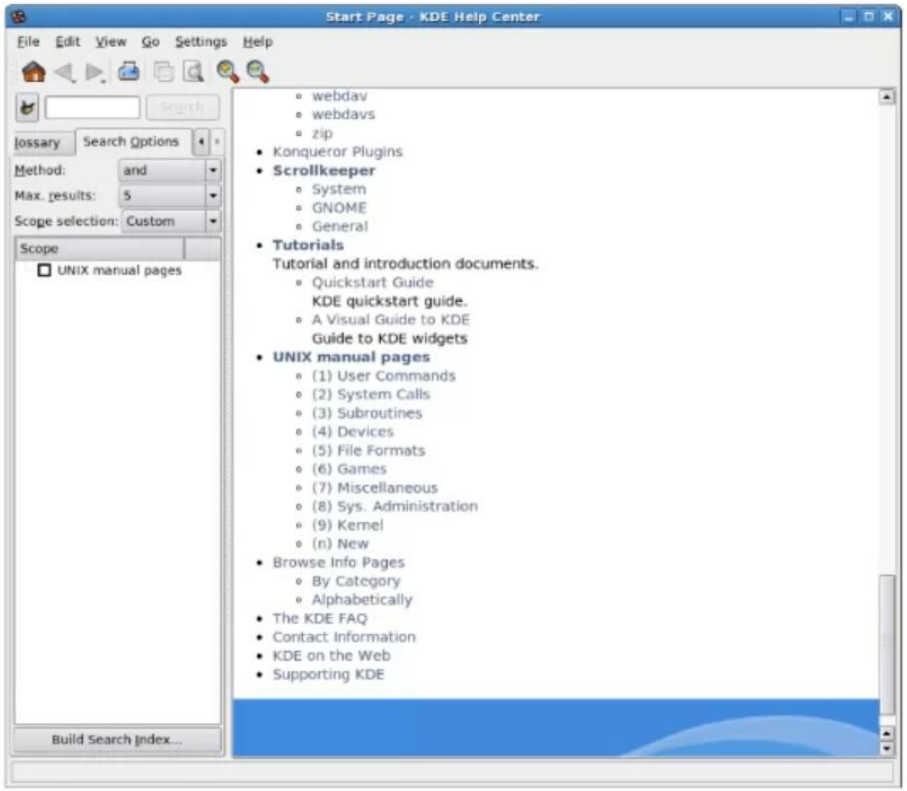


KDE

If you are running a KDE desktop you can invoke the graphical help system from the command line by typing:

```
$ khelpcenter
```

Note: The command may be **kdehelpcenter** instead.



Graphical Interfaces

- Depending on your distribution, the appearance may be different from the screenshots that we presented as distributors may customize the help system
- These graphical interfaces give access not only to the utilities we have already mentioned, such as **man** and **info**, but to other sources that are installed on your system; for instance, the **`/usr/share/doc`** subdirectories contain a lot of browseable material in html files
- Note that even if you are running a GNOME desktop you may have **khelpcenter** installed, and on a KDE desktop you may have **gnome-help** installed, depending on how your system was set up

Text Editors

- At some point you will have to edit text files, and while graphical system administration applications can help you avoid much of this, often it is far more laborious this way than it is to directly work on relevant files with a **text editor**
- By now you have realized Linux is full of choices, and when it comes to text editors, the inventory of alternatives is enormous; the editors vary from simple to very complex
- Many of these editors are already familiar to developers used to working on UNIX-like systems
- Venerable stalwarts such as **vi** and **emacs** are completely compatible with the versions used on other operating systems

Available Editors

Here is a partial list of widely available editors on any Linux distribution:

- **vi** (standard editor available on all UNIX-like systems
- **vim-enhanced** (vi with many enhancements)
- **vim-X11** (vi with full graphics support)
- **emacs** (standard editor available on all UNIX-like systems)
- **xemacs** (variant of emacs preferred by some)
- **nano** (small and easy to use)
- **gedit** (graphical editor part of the GNOME desktop)
- **KWrite** (graphical editor part of the KDE desktop)
- **kate** (graphical editor part of the KDE desktop)
- **nedit** (simple graphical editor)

vi end emacs

- **vi** and **emacs** are the two editors most commonly preferred by Linux users
- They both have very long histories and are available on virtually every Linux installation (although some do not install emacs by default due to some prejudice)
- The UNIX-like world tends to divide between advocates of each; rather silly, holy wars have often broken out between the two camps
- Both of these editors have a basic form which is purely text-based and can run in a non-graphical environment
- They also have one or more X-based graphical forms, which have extended capabilities, and may be easier to sort through for the inexperienced

nano

- There are many other editors available, some associated with desktop managers
- One particularly easy, terminal-based text editor is **nano**
- Just fire up nano on a file and all the help you need is at the bottom of the screen, and you should be able to proceed without a problem

gedit and KWrite

- **gedit** is the editor associated with the GNOME desktop system, and **KWrite** is associated with KDE
- Both are very easy to begin using, extremely capable and configurable

vi

- **vi** (pronounced as "vee-eye") can be found on any Linux system; usually the actual installed program is **vim** (**vi Improved**) which is aliased to the name vi
- Even if you do not want to use vi, it is good to gain some familiarity with it as it is such a standard tool, and there may be times when you have no other choice
- GNOME offers a very graphical interface known as **gvim** and KDE offers **kvim**; both may be found to be easier to use at first
- Using the basic vi (or vim) program, all commands are entered through the keyboard; you do not have to deal with the mouse, unless you are using a graphical version
- The most confusing part about vi is that it has two fundamental modes: **command** and **insert**, and you always have to remember which mode you are in; you switch from one to the other by hitting the escape key, and keyboard functions in each mode can be quite different

emacs

- **emacs** is available on all Linux systems; a less common variant, **xemacs**, is also generally available
- emacs has many other capabilities other than text editing: it can be used for email, debugging, etc.
- Rather than having different modes for command and insert, like vi, emacs uses the **Control** and **META** key for special commands
  - You can use `Alt` for the META key or you can use `Escape`; if you use `Escape`, you must release the key before you type the rest of the key combination or commands listed further in this lesson
  - It is purely a question of taste as virtually all keyboards today have both an `Alt` and `Escape` keys

Historical Order of Introduction

Most Linux users use the default **bash** shell, but those with long UNIX backgrounds with other shells may want to override the default; it is worth reviewing the main choices in the historical order of introduction:

- sh was written by Steve Bourne at AT&T in 1977, and is thus often known as the Bourne Shell; all other shells are descended from it in some fashion and it is available on all systems that have a UNIX bloodline
- csh was written by Bill Joy at UC Berkeley and released in 1978; the internal syntax is quite different that sh and is designed to resemble that of the C programming language, and hence the name
- tcsh was originally developed by Ken Greer at Carnegie Mellon University in the late 1970's; the t stands for TENEX, an operating system that was used on some DEC PDP-10s; it has many additional features as compared with csh and on virtually all modern systems csh is just a link to tcsh

More on Shells

- On most Linux systems, sh is just a link to bash, scripts which are invoked as sh will only work without the bash extensions; a similar relationship exists between csh and tcsh
- Maximum portability is obtained by writing scripts that use only the older features, so you will see many scripts using sh in place of bash; but these days bash is certainly present on all Linux systems, and indeed on almost all UNIX system
- Porting of scripts between sh, ksh and bash is relatively easy and straightforward
- Porting from csh variants is more complicated because of the quite different syntax
- Linux systems make all of these shells available so porting is generally optional
- bash and ksh offer enhanced readability of scripts as compared to sh due to extensions; they also tend to run somewhat faster because the original philosophy of sh was to be minimal, have few internal commands, and rely on external commands for simple operations such as **echo**

Newer Shells

- The newer shells replace some of these external commands with built-in versions
- While this is more efficient, sometimes subtle differences can wreck scripts
- Some of these snags can be avoided by using the full path in the scripts to the external command, such as **`/bin/echo`** instead of just **`echo`**, which would not search the path

Filesystems and the FHS

- In the UNIX tradition, all filesystems and partitions are located under the root filesystem or / directory
- Other partitions are generally mounted on various subdirectory points as are network file systems which may or may not be mounted at any given time
- The **Filesystem Hierarchy Standard** (FHS), administered originally by the Free Standards Group, and now The Linux Foundation, specifies the main directories that need to be present and describes their purposes
- While most Linux distributions respect the FHS, probably none of them follow it exactly, and the last official version is usually old enough that it does not take into account some of the newest developments

Partitioning Scheme

The exact partitioning scheme you use depends on your needs, and is quite flexible. For the most basic commonly used scheme you would have three partitions:

- **`/boot`** is relatively small, typically 100-200 MB, and holds kernels and other boot-related materials; these files are vital and rarely change, and it is safer to keep them on a separate partition
- **`/`** contains everything else and is as large as you need; depending on your distribution, system files and applications and basic programs and development tools will probably chew up 3-8 GB of space
- The **swap** partition should be at least as big as the amount of memory on your system; you can use swap files instead of partitions, but this is a weaker method due to both efficiency and stability considerations

Other Ways to Set Up Partitions

- There are many other ways to set up your partitions, especially on multi-user systems and on systems that use a lot of disk space and have large data files
- This may also depend on the kind of storage media you are using and where it is most efficient to use your fastest and slowest hardware, largest and smallest capacity devices, etc.
- Often the **`/home`** directory is put on a separate partition (or disk)
- **`/usr`** which is relatively static may be put on a separate partition, as might **`/var`** which is quite volatile, and **`/tmp`**, which is temporary
- In addition, one or more of these partitions might be available only as a network share, as in the use of NFS, and not even be mountable until the system is well-booted

Logical Volume Management (LVM)

- Use of **LVM** (Logical Volume Management) introduces even more flexibility, as many logical volumes can be placed on a group of physical volumes which can span more than one disk in a transparent way
- It is easy to dynamically resize and move such partitions

5:59 AM

GRUB

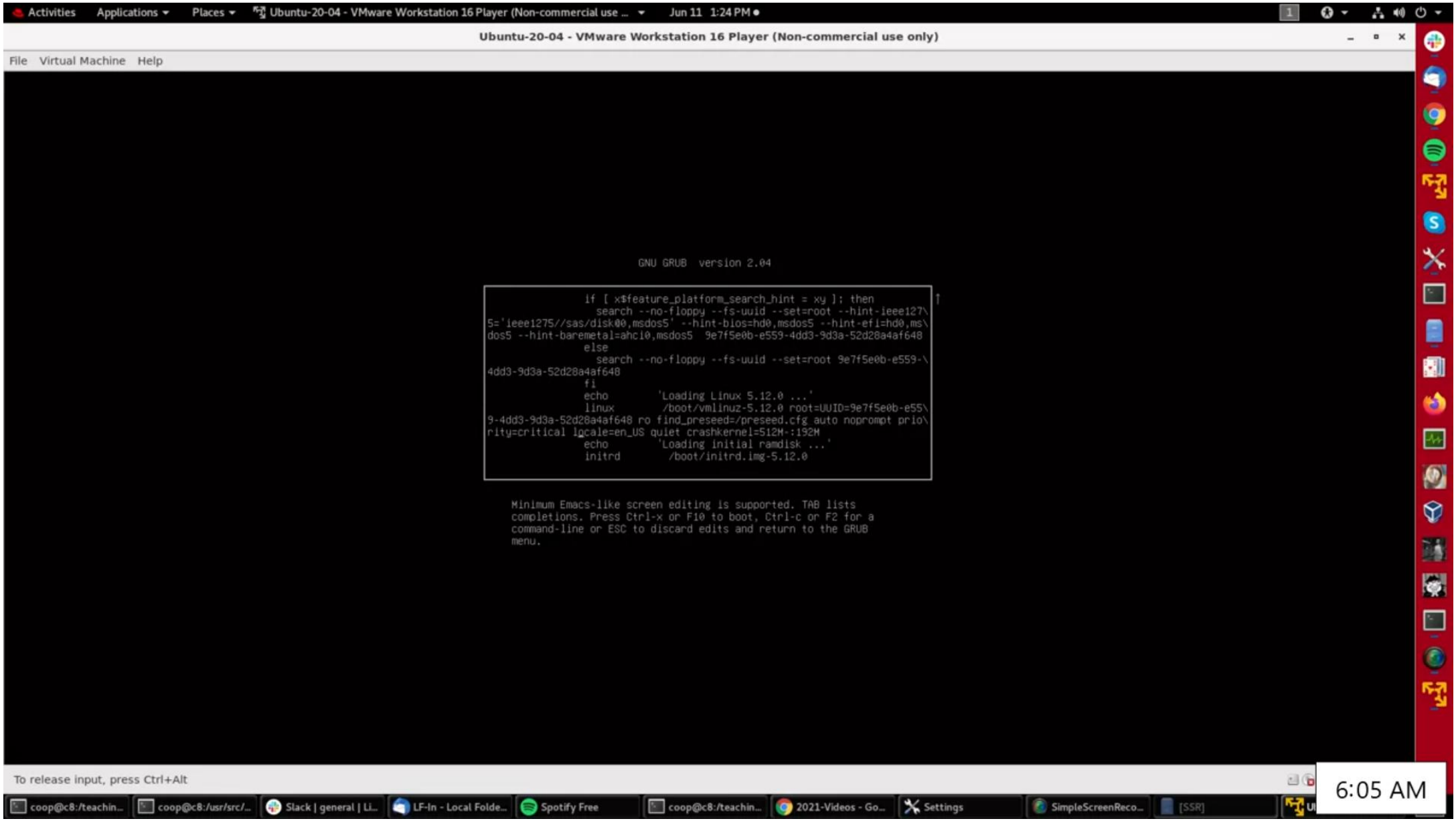
- Virtually all x86-based physical Linux systems (outside the embedded sphere) today use **GRUB** (**GR**and **U**nified **B**ootloader) to handle the early phases of system startup
- Other platforms may have other equivalents, such as ELILO used on IA64 (Itanium), and Das U-Boot used on many embedded configurations
- Some important features of GRUB are:
  - Alternative operating systems can be chosen at boot time
  - Alternative kernels and/or initial ramdisks can be chosen at boot time for a given operating system
  - Boot parameters can be easily changed at boot time without having to edit configuration files in advance

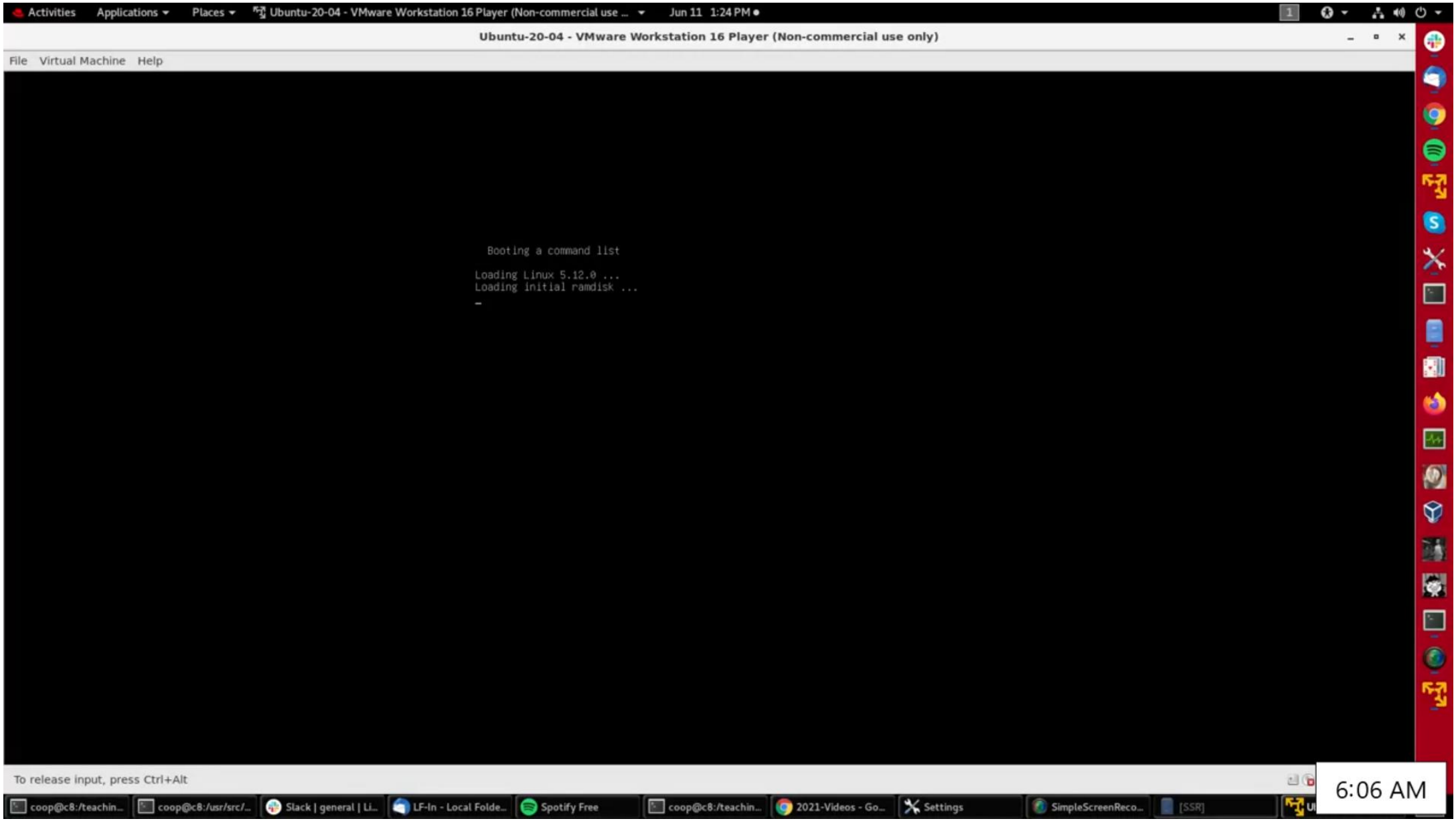
GRUB (Cont.)

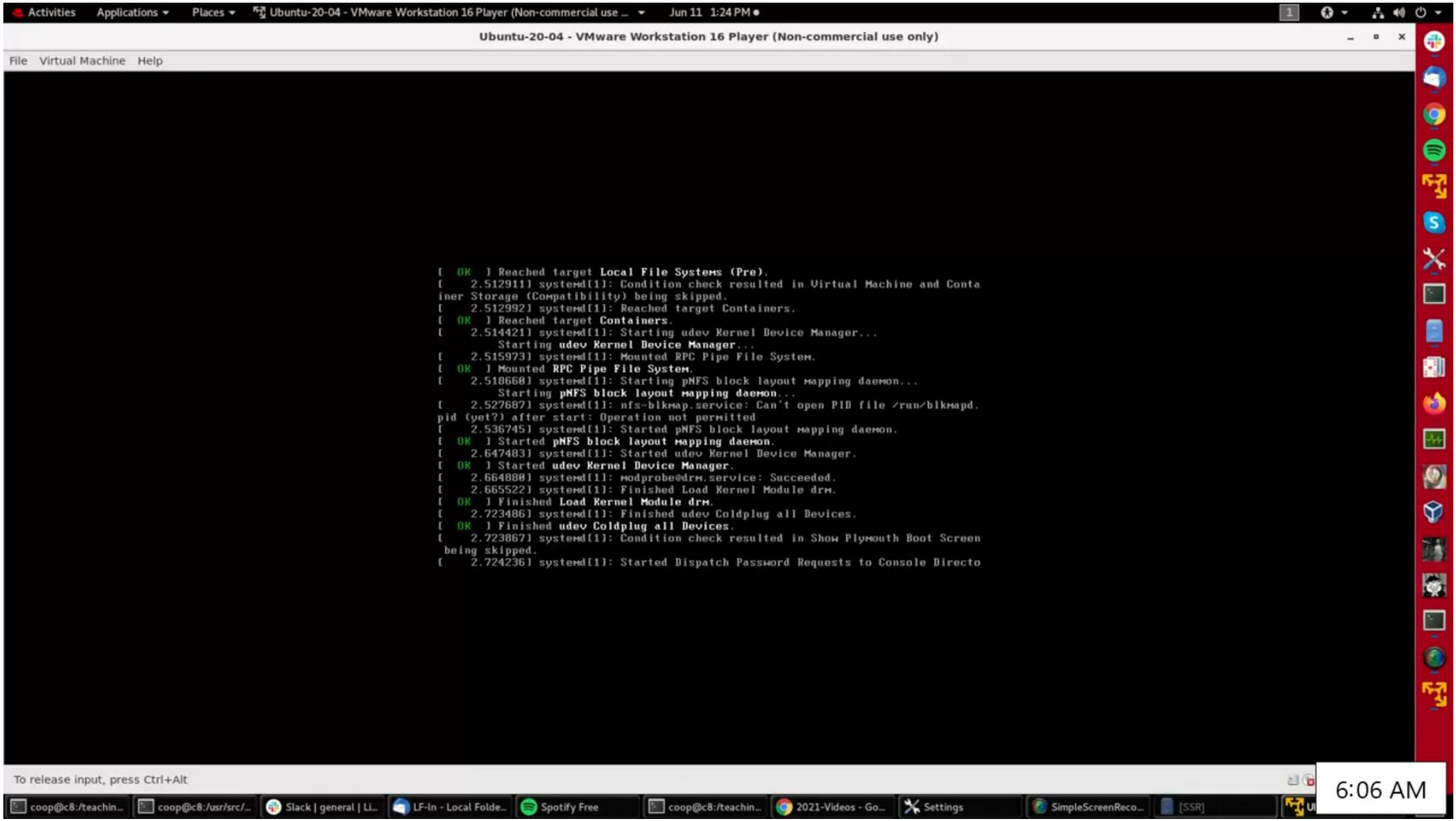
- Older Linux distributions (such as RHEL6) use older GRUB versions (1.0 or below), while newer ones are based on GRUB 2; while details are different between the versions, the basic philosophy is the same
- In GRUB 2 the basic configuration file is **`/boot/grub/grub.cfg`** or **`/boot/grub2/grub.cfg`**
- This file is auto-generated when you run **update-grub** or **grub2-mkconfig** (depending on distribution), based on files in the **`/etc/grub.d`** and **`/etc/default/grub`**; it should not be edited by hand
- On some purely EFI systems, this file might be found in a directory like **`/boot/efi/EFI/redhat/grub.cfg`**
- The essential configuration file contains some global information, and then a **stanza** for each operating system or kernel configured

Boot

- On x86 systems, boot begins with the **BIOS** identifying and initializing all system and attached peripheral devices
- If permitted by BIOS settings, the system can then boot off a peripheral device such as a CD, DVD, floppy, or USB drive, or through network PXE; more likely it will boot off the configured hard disk
- The **MBR** (Master Boot Record) contains both the **partition table** and the **boot loader**, a short program that is responsible for loading the operating system, and thus has to have at least sufficient smarts to locate and load the kernel from disk
- This kernel can have any name; usually on Linux systems it is called something like **`vmlinuz-4.18.0`** which includes the kernel version number
  - The z in **`vmlinuz`** indicates the kernel is compressed; it self-decompresses upon loading
  - In most circumstances it is placed in the **`/boot`** directory which is often on its own partition


Activities
Applications
Places
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use ...
Jun 11 1:24 PM
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use only)
File Virtual Machine Help
GNU GRUB version 2.04
if [x$feature_platform_search_hint = xy]; then
search --no-floppy --fs-uuid --set=root --hint-ieee127\
5='ieee1275//sas/disk00,msdos5' --hint-bios=hd0,msdos5 --hint-efi=hd0,ms\
dos5 --hint-baremetal=ahci0,msdos5 9e7f5e0b-e559-4dd3-9d3a-52d28a4af648
else
search --no-floppy --fs-uuid --set=root 9e7f5e0b-e559-\
4dd3-9d3a-52d28a4af648
fi
echo 'Loading Linux 5.12.0 ...'
linux /boot/vmlinuz-5.12.0 root=UUID=9e7f5e0b-e55\
9-4dd3-9d3a-52d28a4af648 ro find_preseed=/preseed.cfg auto noprompt prio\
rity=critical locale=en_US quiet crashkernel=512M-:192M
echo 'Loading initial ramdisk ...'
initrd /boot/initrd.img-5.12.0
Minimum Emacs-like screen editing is supported. TAB lists
completions. Press Ctrl-x or F10 to boot, Ctrl-c or F2 for a
command-line or ESC to discard edits and return to the GRUB
menu.
To release input, press Ctrl+Alt
coop@c8:/teachin...
coop@c8:/usr/src/...
Slack | general | Li...
LF-In - Local Folde...
Spotify Free
coop@c8:/teachin...
2021-Videos - Go...
Settings
SimpleScreenReco...
[SSR]
6:05 AM


Activities
Applications
Places
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use ...
Jun 11 1:24 PM
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use only)
File
Virtual Machine
Help
Booting a command list
Loading Linux 5.12.0 ...
Loading initial ramdisk ...
To release input, press Ctrl+Alt
coop@c8:/teachin...
coop@c8:/usr/src/...
Slack | general | Li...
LF-In - Local Folde...
Spotify Free
coop@c8:/teachin...
2021-Videos - Go...
Settings
SimpleScreenReco...
[SSR]
6:06 AM


Activities
Applications
Places
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use ...
Jun 11 1:24 PM
Ubuntu-20-04 - VMware Workstation 16 Player (Non-commercial use only)
File Virtual Machine Help
[OK] Reached target Local File Systems (Pre).
[2.512911] systemd[1]: Condition check resulted in Virtual Machine and Container Storage (Compatibility) being skipped.
[2.512992] systemd[1]: Reached target Containers.
[OK] Reached target Containers.
[2.514421] systemd[1]: Starting udev Kernel Device Manager...
Starting udev Kernel Device Manager...
[2.515973] systemd[1]: Mounted RPC Pipe File System.
[OK] Mounted RPC Pipe File System.
[2.518660] systemd[1]: Starting pNFS block layout mapping daemon...
Starting pNFS block layout mapping daemon...
[2.527687] systemd[1]: nfs-blkmap.service: Can't open PID file /run/blkmapd.pid (yet?) after start: Operation not permitted
[2.536745] systemd[1]: Started pNFS block layout mapping daemon.
[OK] Started pNFS block layout mapping daemon.
[2.647483] systemd[1]: Started udev Kernel Device Manager.
[OK] Started udev Kernel Device Manager.
[2.664880] systemd[1]: modprobe@drm.service: Succeeded.
[2.665522] systemd[1]: Finished Load Kernel Module drm.
[OK] Finished Load Kernel Module drm.
[2.723486] systemd[1]: Finished udev Coldplug all Devices.
[OK] Finished udev Coldplug all Devices.
[2.723867] systemd[1]: Condition check resulted in Show Plymouth Boot Screen being skipped.
[2.724236] systemd[1]: Started Dispatch Password Requests to Console Directo
To release input, press Ctrl+Alt
coop@c8:/teachin...
coop@c8:/usr/src/...
Slack | general | Li...
LF-In - Local Folde...
Spotify Free
coop@c8:/teachin...
2021-Videos - Go...
Settings
SimpleScreenReco...
[SSR]
6:06 AM

init

- **`/sbin/init`** (usually just called **init**) is the first user-level process (or task) that runs on the system and continues to run until the system is shutdown
- Traditionally, it has been considered the parent of all user processes, although technically that is not true as some processes are started directly by the kernel
- init coordinates the later stages of the boot process, configures all aspects of the environment and starts the processes needed for logging into the system; it also works closely with the kernel in cleaning up after processes when they terminate

SysVinit

- Traditionally, nearly all distributions based the init process on UNIX's venerable **SysVinit**
- This scheme was developed decades ago under rather different circumstances:
  - The target was multi-user mainframe systems (not personal computers, laptops, and other devices)
  - The target was a single processor system
  - Startup (and shutdown) time was not an important matter; it was far less important than getting things right
    - Startup was viewed as a serial process, divided into a series of sequential stages; each stage required completion before the next could proceed, so it didn't easily take advantage of the parallel processing that could be done on multiple processors or cores
    - Shutdown/reboot was seen as a relatively rare event and exactly how long it took was not considered important

SysVinit

- Traditionally, nearly all distributions based the init process on UNIX's venerable **SysVinit**
- This scheme was developed decades ago under rather different circumstances:
  - The target was multi-user mainframe systems (not personal computers, laptops, and other devices)
  - The target was a single processor system
  - Startup (and shutdown) time was not an important matter; it was far less important than getting things right
    - Startup was viewed as a serial process, divided into a series of sequential stages; each stage required completion before the next could proceed, so it didn't easily take advantage of the parallel processing that could be done on multiple processors or cores
    - Shutdown/reboot was seen as a relatively rare event and exactly how long it took was not considered important

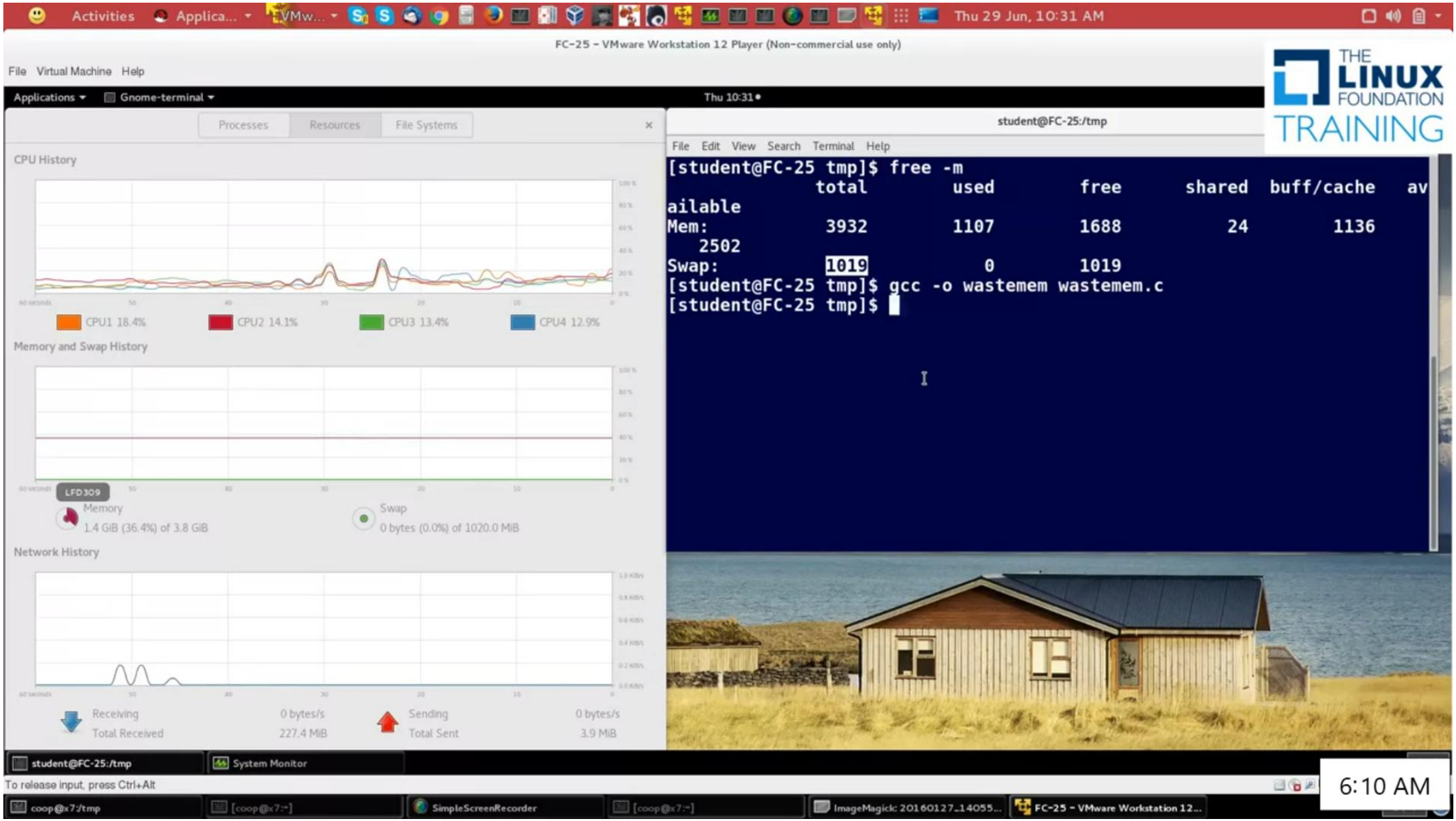
New Methods of Controlling System Startup

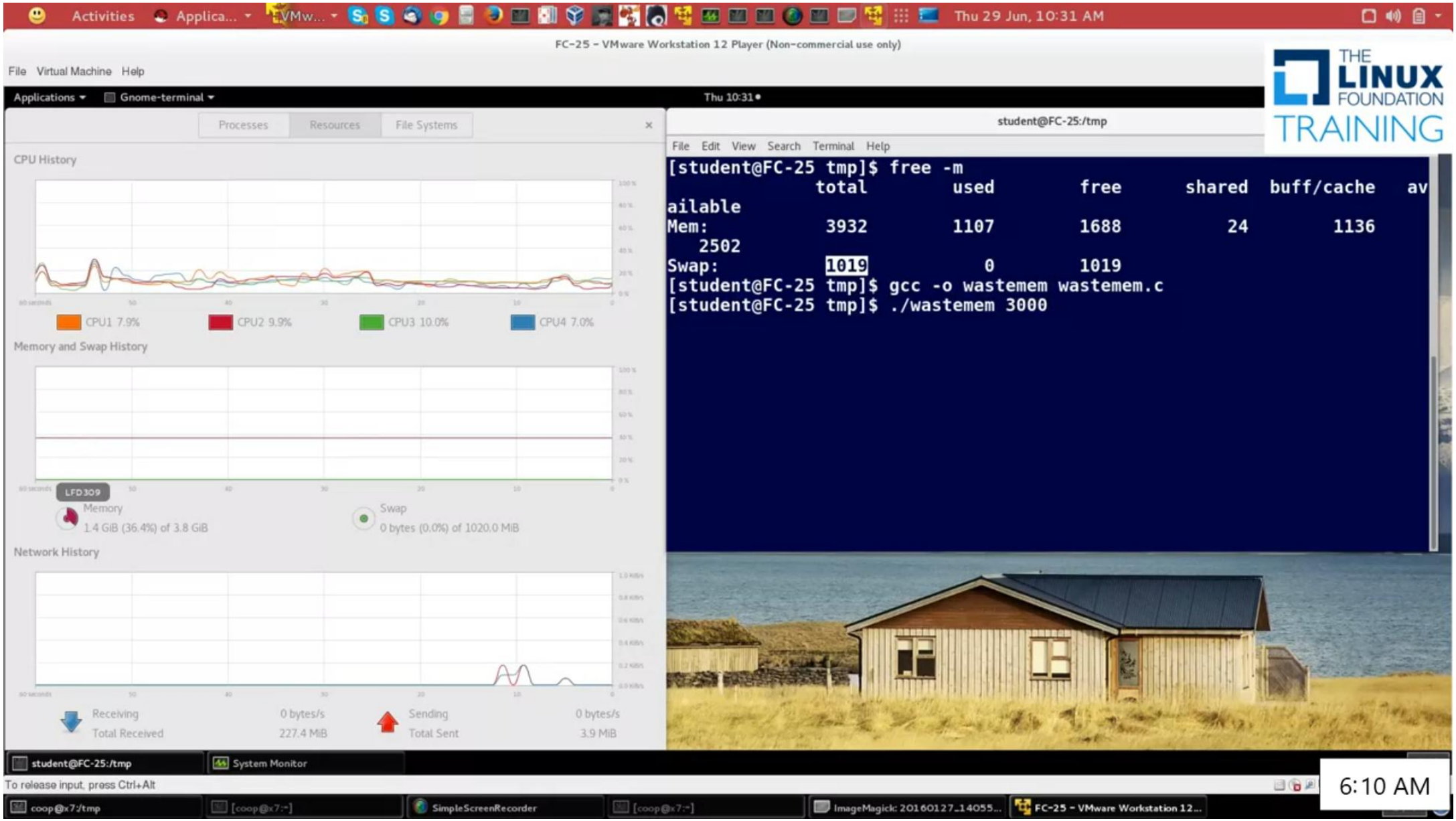
To deal with these intrinsic limitations in SysVinit, new methods of controlling system startup were developed and have replaced it in all new systems. Two main schemes were adopted by Linux distributors:

- Upstart
  - Developed by Ubuntu and first included in the 6.10 release in 2006, and made the default in the 9.10 release in 2009
  - Adopted in Fedora 9 (in 2008) and in RHEL 6 and its clones, such as CentOS, Scientific Linux and Oracle Linux, and in openSUSE it was offered as an option since version 11.3
  - It has also been used in various embedded and mobile devices
- systemd is more recent and Fedora was the first major distribution to adopt it in 2011

systemd

- RHEL 7 was based on **systemd** and since then every other major Linux distribution has adopted it and has made it the default; even Ubuntu phased out Upstart and is now systemd-based
- We will discuss systemd commands, in particular the use of **systemctl**, when we discuss controlling system services; starting and stopping, enabling and disabling at boot, showing status, etc.
- Compatibility wrappers ensure one can use SysVinit utilities and methods for quite some time, even if under the hood things are quite different
- However, eventually they will atrophy and disappear, so learning systemd is the best investment of your learning efforts


Activities
Applica...
VMw...
Thu 29 Jun, 10:31 AM
FC-25 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
THE LINUX FOUNDATION TRAINING
Applications
Gnome-terminal
Thu 10:31
Processes
Resources
File Systems
CPU History
CPU1 18.4%
CPU2 14.1%
CPU3 13.4%
CPU4 12.9%
Memory and Swap History
LFD309
Memory
1.4 GiB (36.4%) of 3.8 GiB
Swap
0 bytes (0.0%) of 1020.0 MiB
Network History
Receiving
0 bytes/s
Total Received
227.4 MiB
Sending
0 bytes/s
Total Sent
3.9 MiB
student@FC-25:/tmp
File Edit View Search Terminal Help
[student@FC-25 tmp]$ free -m
total used free shared buff/cache av
ailable
Mem: 3932 1107 1688 24 1136
2502
Swap: 1019 0 1019
[student@FC-25 tmp]$ gcc -o wastemem wastemem.c
[student@FC-25 tmp]$
student@FC-25:/tmp
System Monitor
To release input, press Ctrl+Alt
coop@x7:/tmp
[coop@x7:~]
SimpleScreenRecorder
[coop@x7:~]
ImageMagick 20160127_14055...
FC-25 - VMware Workstation 12...
6:10 AM


Activities
Applica...
VMw...
Thu 29 Jun, 10:31 AM
FC-25 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
Gnome-terminal
Thu 10:31
Processes
Resources
File Systems
CPU History
CPU1 7.9%
CPU2 9.9%
CPU3 10.0%
CPU4 7.0%
Memory and Swap History
LFD309
Memory
1.4 GiB (36.4%) of 3.8 GiB
Swap
0 bytes (0.0%) of 1020.0 MiB
Network History
Receiving
0 bytes/s
Total Received
227.4 MiB
Sending
0 bytes/s
Total Sent
3.9 MiB
student@FC-25:/tmp
File Edit View Search Terminal Help
[student@FC-25 tmp]$ free -m
total used free shared buff/cache av
ailable
Mem: 3932 1107 1688 24 1136
2502
Swap: 1019 0 1019
[student@FC-25 tmp]$ gcc -o wastemem wastemem.c
[student@FC-25 tmp]$ ./wastemem 3000
student@FC-25:/tmp
System Monitor
To release input, press Ctrl+Alt
coop@x7:/tmp
[coop@x7:~]
SimpleScreenRecorder
[coop@x7:~]
ImageMagick 20160127_14055...
FC-25 - VMware Workstation 12...
6:10 AM

Activities Applica... VMw... Thu 29 Jun, 10:32 AM

FC-25 – VMware Workstation 12 Player (Non-commercial use only)

File Virtual Machine Help

THE LINUX FOUNDATION TRAINING

Applications Gnome-terminal Thu 10:32

Processes Resources File Systems

CPU History

CPU1 11.3% CPU2 8.9% CPU3 49.5% CPU4 12.7%

Memory and Swap History

LFD309

Memory
985.8 MiB (25.1%) of 3.8 GiB

Swap
395.4 MiB (100.0%) of 395.4 MiB

Network History

Receiving 0 bytes/s
Total Received 227.4 MiB

Sending 0 bytes/s
Total Sent 3.9 MiB

student@FC-25:/tmp

File Edit View Search Terminal Help

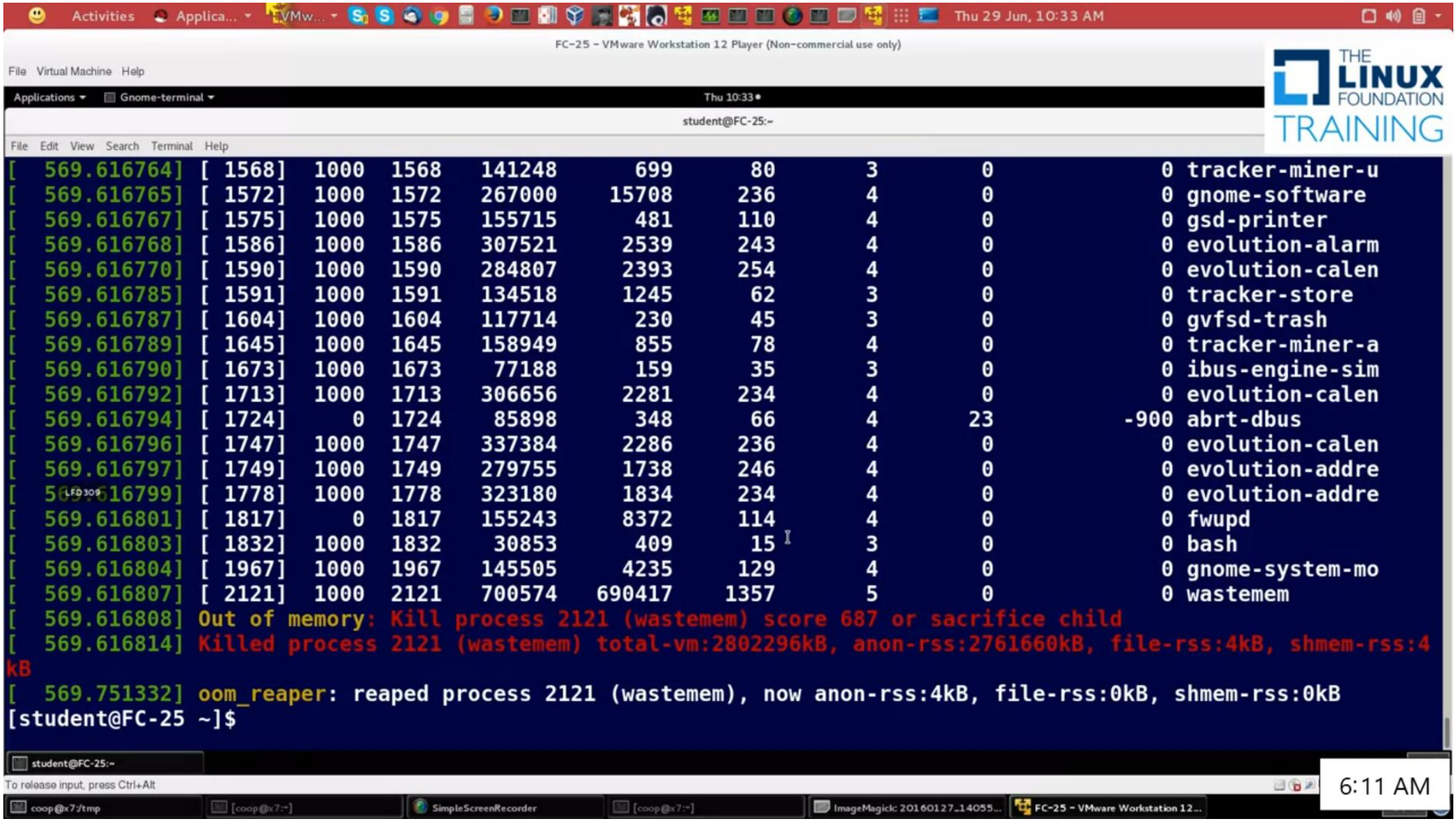
```
4 2765 2766 2767 2768 2769 2770 2771 2772 2773 2774 2775 2776 2777 2778
2779 2780 2781 2782 2783 2784 2785 2786 2787 2788 2789 2790 2791 2792 27
93 2794 2795 2796 2797 2798 2799 2800 2801 2802 2803 2804 2805 2806 2807
 2808 2809 2810 2811 2812 2813 2814 2815 2816 2817 2818 2819 2820 2821 2
822 2823 2824 2825 2826 2827 2828 2829 2830 2831 2832 2833 2834 2835 283
6 2837 2838 2839 2840 2841 2842 2843 2844 2845 2846 2847 2848 2849 2850
2851 2852 2853 2854 2855 2856 2857 2858 2859 2860 2861 2862 2863 2864 28
65 2866 2867 2868 2869 2870 2871 2872 2873 2874 2875 2876 2877 2878 2879
 2880 2881 2882 2883 2884 2885 2886 2887 2888 2889 2890 2891 2892 2893 2
894 2895 2896 2897 2898 2899 2900 2901 2902 2903 2904 2905 2906 2907 290
8 2909 2910 2911 2912 2913 2914 2915 2916 2917 2918 2919 2920 2921 2922
2923 2924 2925 2926 2927 2928 2929 2930 2931 2932 2933 2934 2935 2936 29
37 2938 2939 2940 2941 2942 2943 2944 2945 2946 2947 2948 2949 2950 2951
 2952 2953 2954 2955 2956 2957 2958 2959 2960 2961 2962 2963 2964 2965 2
966 2967 2968 2969 2970 2971 2972 2973 2974 2975 2976 2977 2978 2979 298
0 2981 2982 2983 2984 2985 2986 2987 2988 2989 2990 2991 2992 2993 2994
2995 2996 2997 2998 2999All memory allocated, pausing 5 seconds
Quitting and releasing memory
[student@FC-25 tmp]$ sudo swapoff -a
```

student@FC-25:/tmp System Monitor

To release input, press Ctrl+Alt

coop@x7:/tmp [coop@x7:~] SimpleScreenRecorder [coop@x7:~] ImageMagick 20160127_14055... FC-25 - VMware Workstation 12...

6:11 AM


Activities
Applica...
VMw...
Thu 29 Jun, 10:33 AM
FC-25 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
Gnome-terminal
Thu 10:33
student@FC-25:~
File Edit View Search Terminal Help
[569.616764] [1568] 1000 1568 141248 699 80 3 0 0 tracker-miner-u
[569.616765] [1572] 1000 1572 267000 15708 236 4 0 0 gnome-software
[569.616767] [1575] 1000 1575 155715 481 110 4 0 0 gsd-printer
[569.616768] [1586] 1000 1586 307521 2539 243 4 0 0 evolution-alarm
[569.616770] [1590] 1000 1590 284807 2393 254 4 0 0 evolution-calen
[569.616785] [1591] 1000 1591 134518 1245 62 3 0 0 tracker-store
[569.616787] [1604] 1000 1604 117714 230 45 3 0 0 gvfsd-trash
[569.616789] [1645] 1000 1645 158949 855 78 4 0 0 tracker-miner-a
[569.616790] [1673] 1000 1673 77188 159 35 3 0 0 ibus-engine-sim
[569.616792] [1713] 1000 1713 306656 2281 234 4 0 0 evolution-calen
[569.616794] [1724] 0 1724 85898 348 66 4 23 -900 abrt-dbus
[569.616796] [1747] 1000 1747 337384 2286 236 4 0 0 evolution-calen
[569.616797] [1749] 1000 1749 279755 1738 246 4 0 0 evolution-addre
[569.616799] [1778] 1000 1778 323180 1834 234 4 0 0 evolution-addre
[569.616801] [1817] 0 1817 155243 8372 114 4 0 0 fwupd
[569.616803] [1832] 1000 1832 30853 409 15 3 0 0 bash
[569.616804] [1967] 1000 1967 145505 4235 129 4 0 0 gnome-system-mo
[569.616807] [2121] 1000 2121 700574 690417 1357 5 0 0 wastemem
[569.616808] Out of memory: Kill process 2121 (wastemem) score 687 or sacrifice child
[569.616814] Killed process 2121 (wastemem) total-vm:2802296kB, anon-rss:2761660kB, file-rss:4kB, shmem-rss:4kB
[569.751332] oom_reaper: reaped process 2121 (wastemem), now anon-rss:4kB, file-rss:0kB, shmem-rss:0kB
[student@FC-25 ~]$
student@FC-25:~
To release input, press Ctrl+Alt
coop@x7:/tmp
SimpleScreenRecorder
ImageMagick 20160127_14055...
FC-25 - VMware Workstation 12...
6:11 AM
THE LINUX FOUNDATION TRAINING

Processes and Threads

- A **process** is a running instance of a program and contains information about environment variables, file descriptors and current directory, etc.
- It can contain one or more **threads**, each of which has the same process ID and shares the same environment, memory regions (except for stack), etc.
- While Linux permits thread (or process) creation through the low level **`clone()`** system call, the most usual method is to use **`pthread_create()`** which is part of the standard **POSIX Threads** (**pthreads**) library

Implementation of pthreads

- In a perfect world any pthreads compliant program should be *write once*, *use anywhere*
- The real world is not always so simple; for example, one operating system's implementation of pthreads may be more forgiving of errors than another and when an application is ported it may stop working
- Instances have been seen, where **Solaris** was automatically initializing improperly uninitialized mutexes, while Linux did not and thus led to program failure
- Also system differences such as default stack sizes can lead to unforeseen problems
- Other operating systems may also introduce their own specific multi-threading environments, such as Solaris threads
- Porting applications from such environments may be simple or difficult depending on APIs; it is best to stick with pthreads if possible, for portability alone

Naming Networks

- Simply naming network devices as **eth0**, **eth1**, etc. can be problematic when multiple interfaces exist, or when the order in which the system probes for and then finds them is not deterministic and can depend on kernel version and options
- Many system administrators have solved this problem in a simple manner, by hard-coding associations between hardware (MAC) addresses and device names in system configuration files and startup scripts
- A more modern approach is offered by the **Predictable Network Interface Device Names** scheme, which is strongly correlated with the use of udev and integration with systemd

Five Types of Names

- There are now 5 types of names that devices can be given:
  - Incorporating Firmware or BIOS provided index numbers for on-board devices, e.g. **eno1**
  - Incorporating Firmware or BIOS provided PCI Express hotplug slot index numbers, e.g. **ens1**
  - Incorporating physical and/or geographical location of the hardware connection, e.g. **enp2s0**
  - Incorporating the MAC address, e.g. **enx7837d1ea46da**
  - Using the old classic method, e.g. **eth0**
- On a machine with two onboard PCI network interfaces that would have been **eth0** and **eth1** in the older naming system:

```
$ ip -s -o -a | grep eno
2: eno1    inet 192.168.1.100/24 brd 192.168.1.255 scope global noprefixroute eno1 ....
3: eno2    inet 192.168.1.210/24 brd 192.168.1.255 scope global noprefixroute eno2 ....
```

- It is easy to turn off the new scheme and go back to the classic names, if so desired

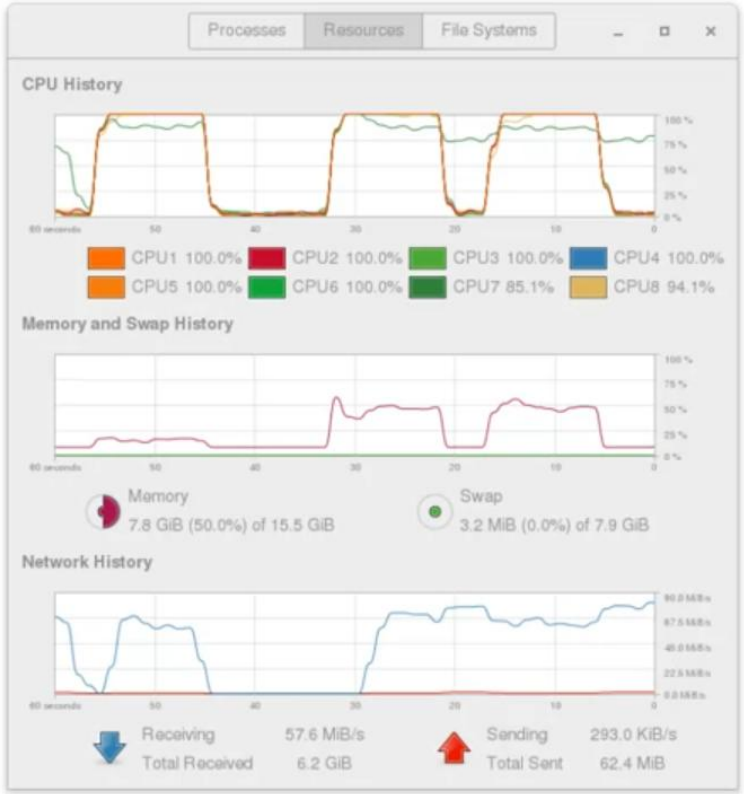
File Transfer Tools

- On Linux systems file transfer tools are plentiful and familiar
- **FTP** clients abound, from the familiar command line ftp and enhanced (or dumbed down depending on your point of view) versions such as lftp, ncftp and qftp
  - An excellent graphical client is provided by **FileZilla**, and of course any competent browser can handle FTP downloads, although not uploads
  - The dominant ftp server in use on Linux systems is **vsftpd**, which is both simple and highly configurable
- Somewhat more general transfer opportunities are provided by both **curl** and **wget** and their associates, which can handle protocols such as ftp, http, etc.

gnome-system-monitor

- **gnome-system-monitor** is a simple graphical monitoring tool; it is installed on any Linux distribution that provides the GNOME desktop manager (even if you are running another desktop manager, such as KDE, it will be available)
- All statistics are generated in real time, and it is easy to examine CPU load, memory and swap usage, and network bytes transmitted and received
- However, it does not have facilities to save data and the choice of display items is limited

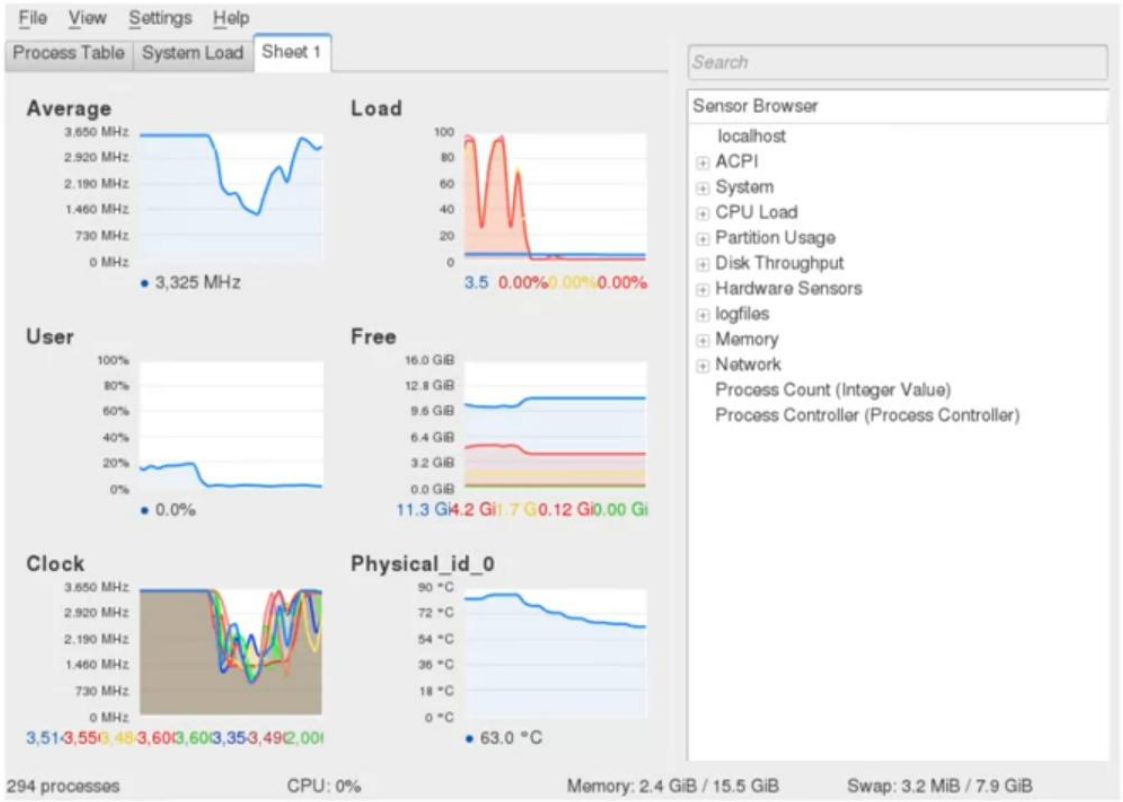
gnome-system-monitor (Cont.)



gnome-system-monitor in action on a RHEL 7 system

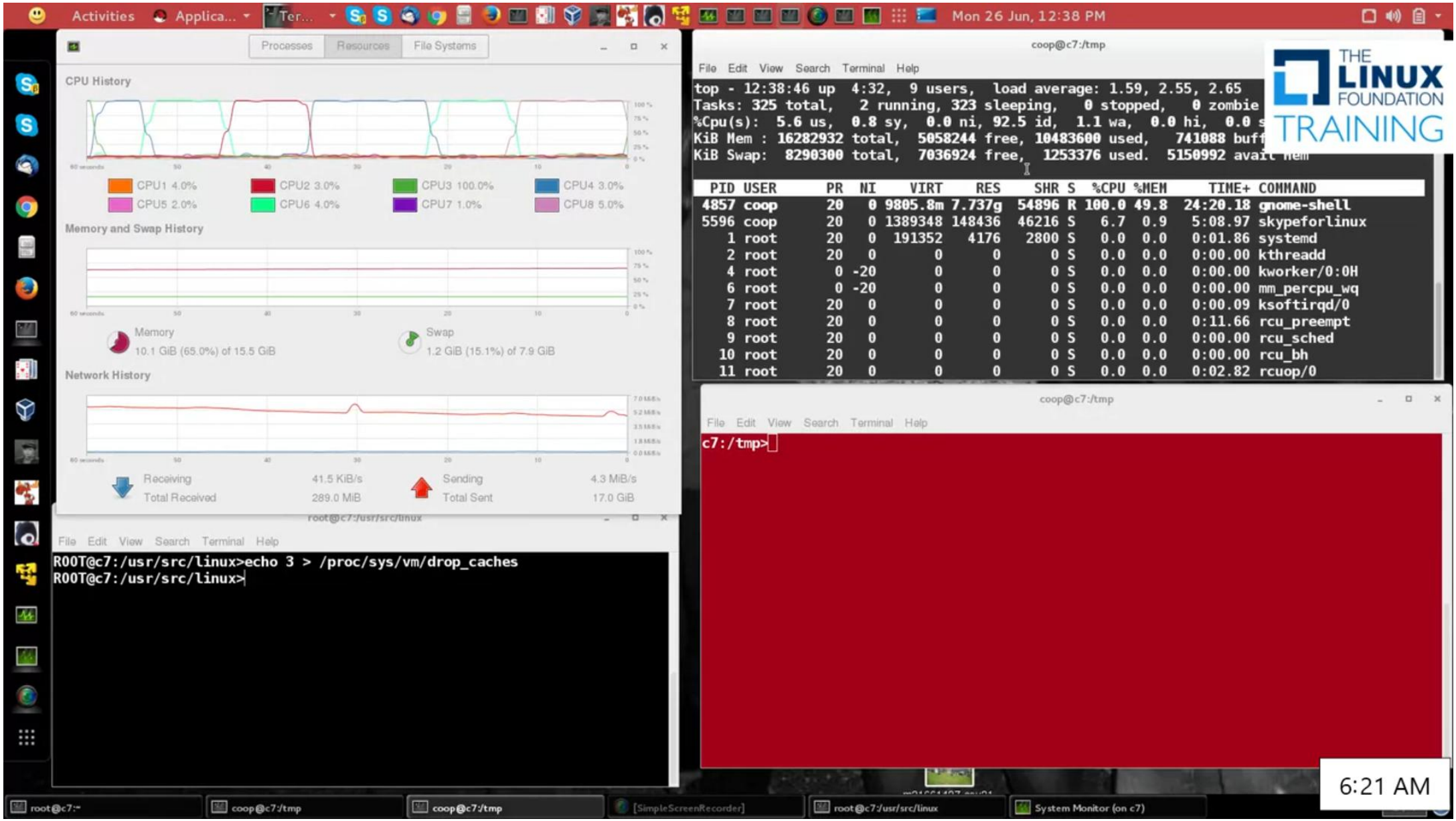
6:21 AM

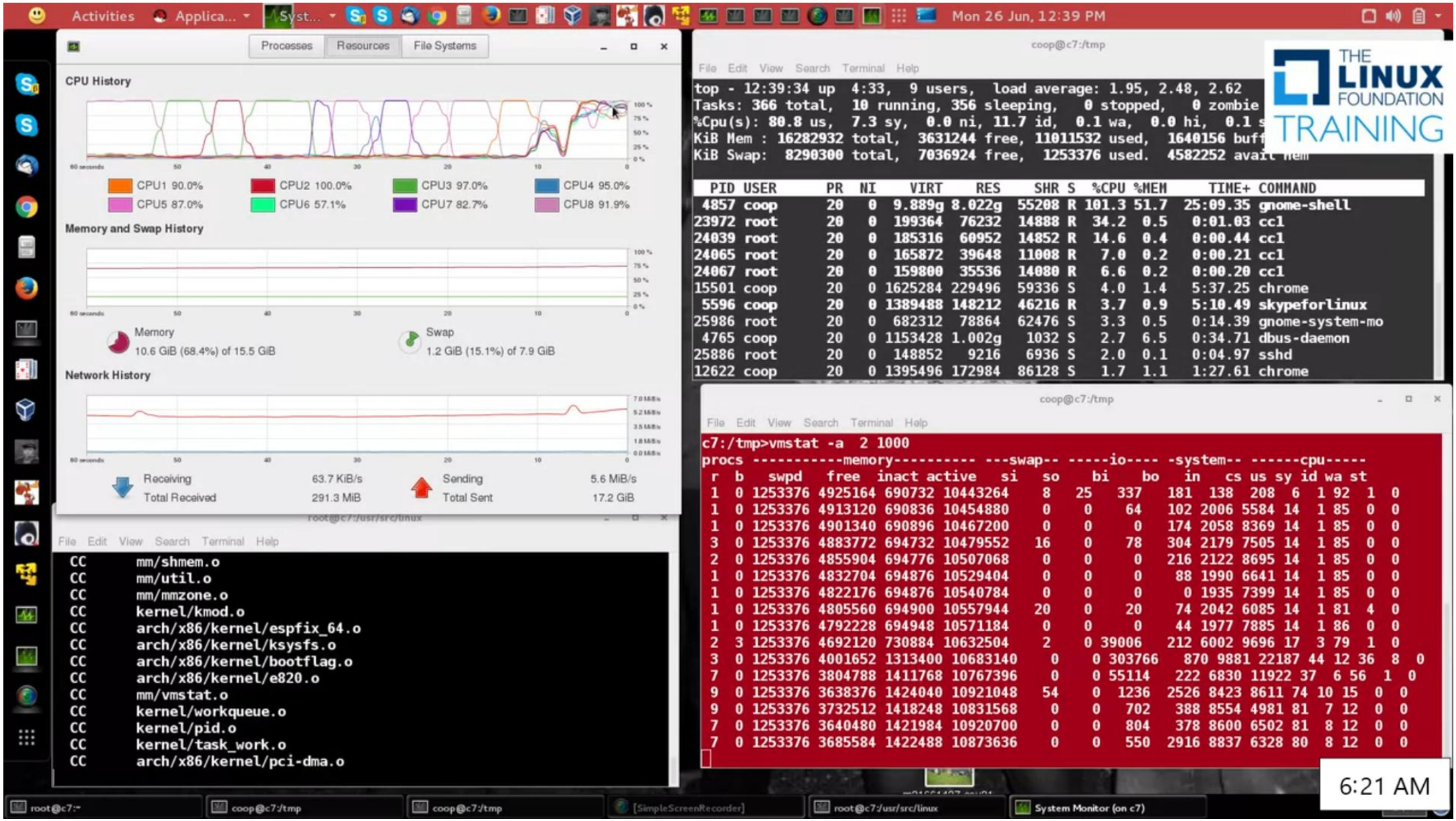
ksysguard (Cont.)

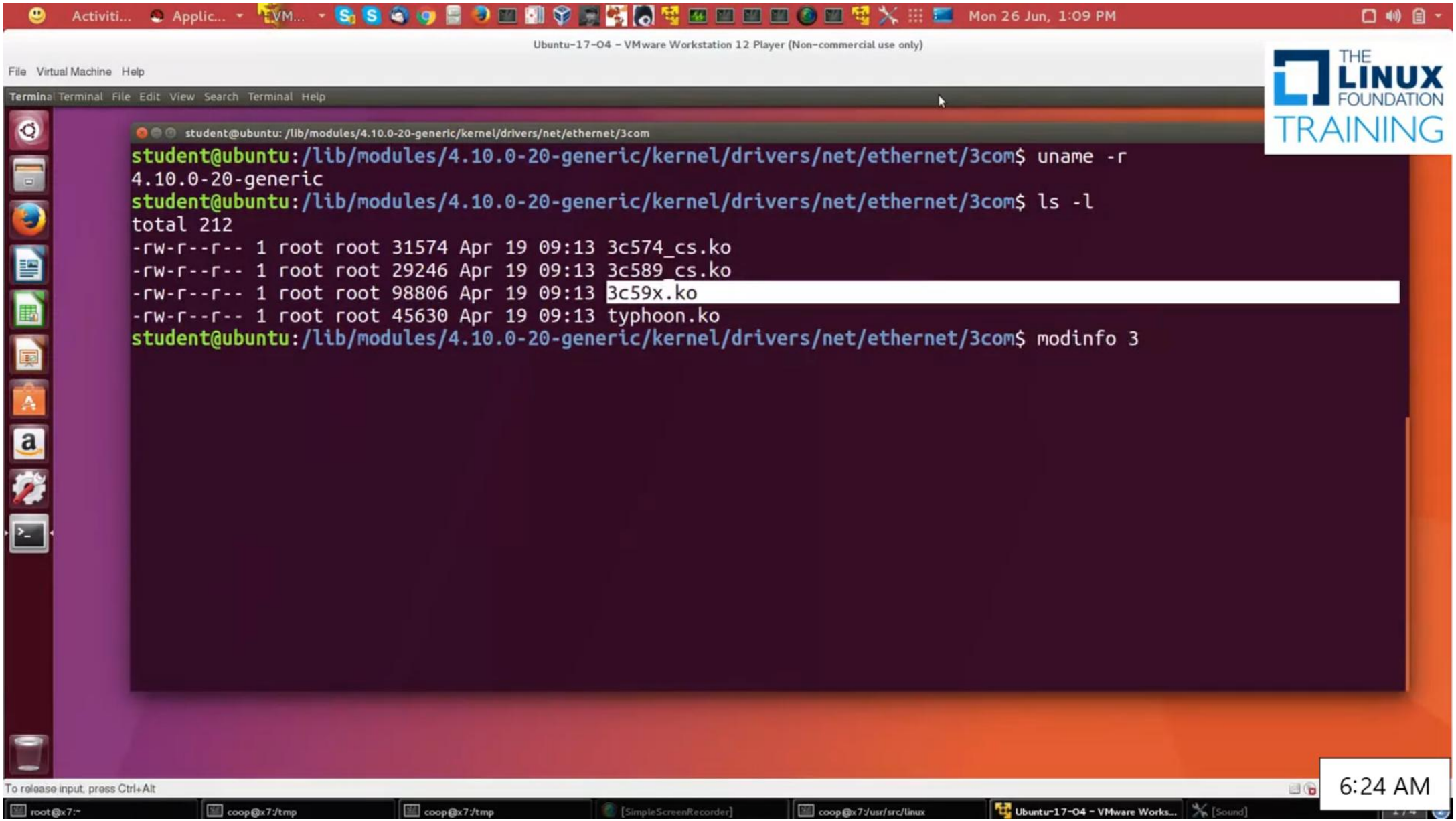


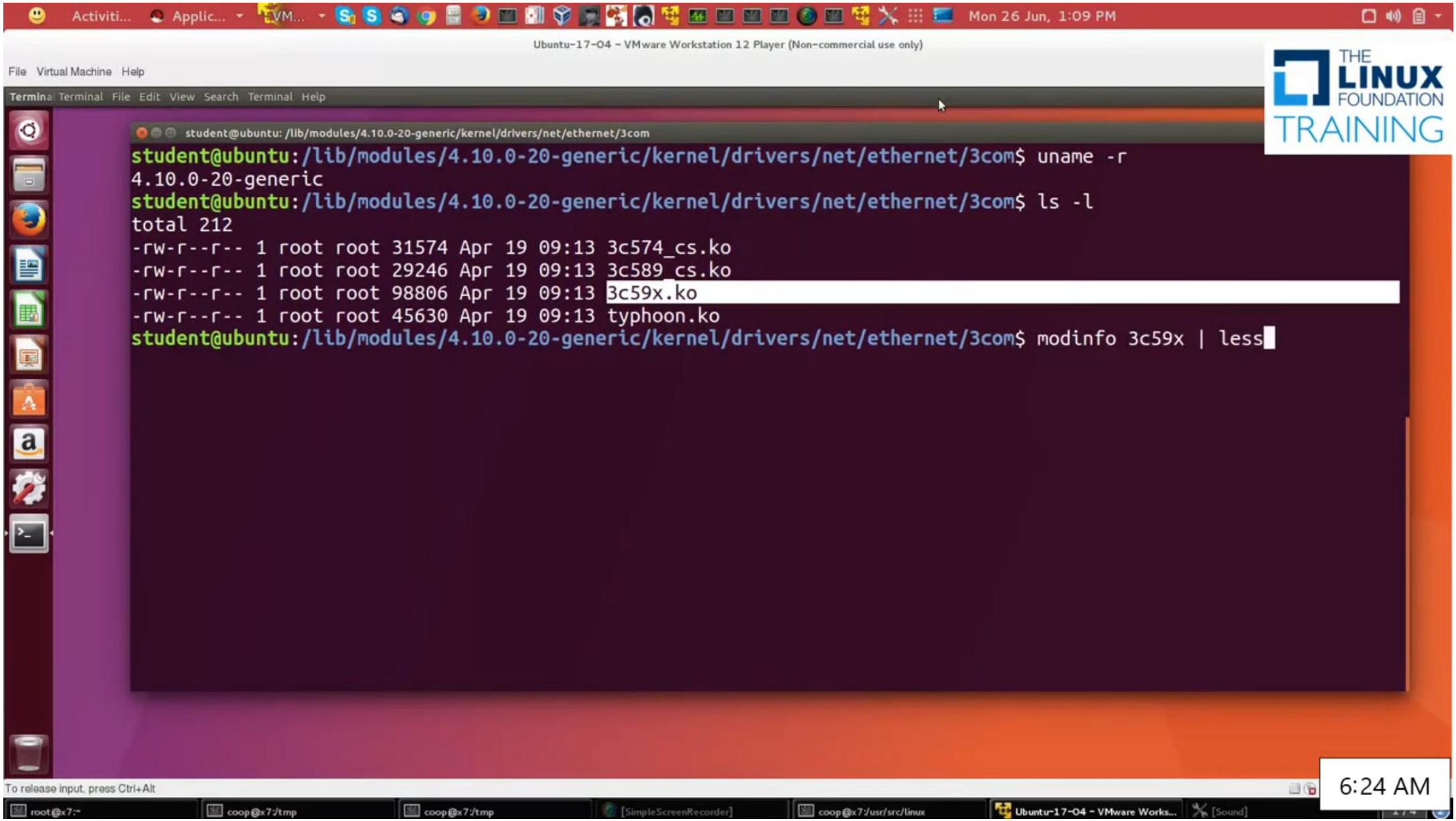
ksysguard in action
on a RHEL 7 system

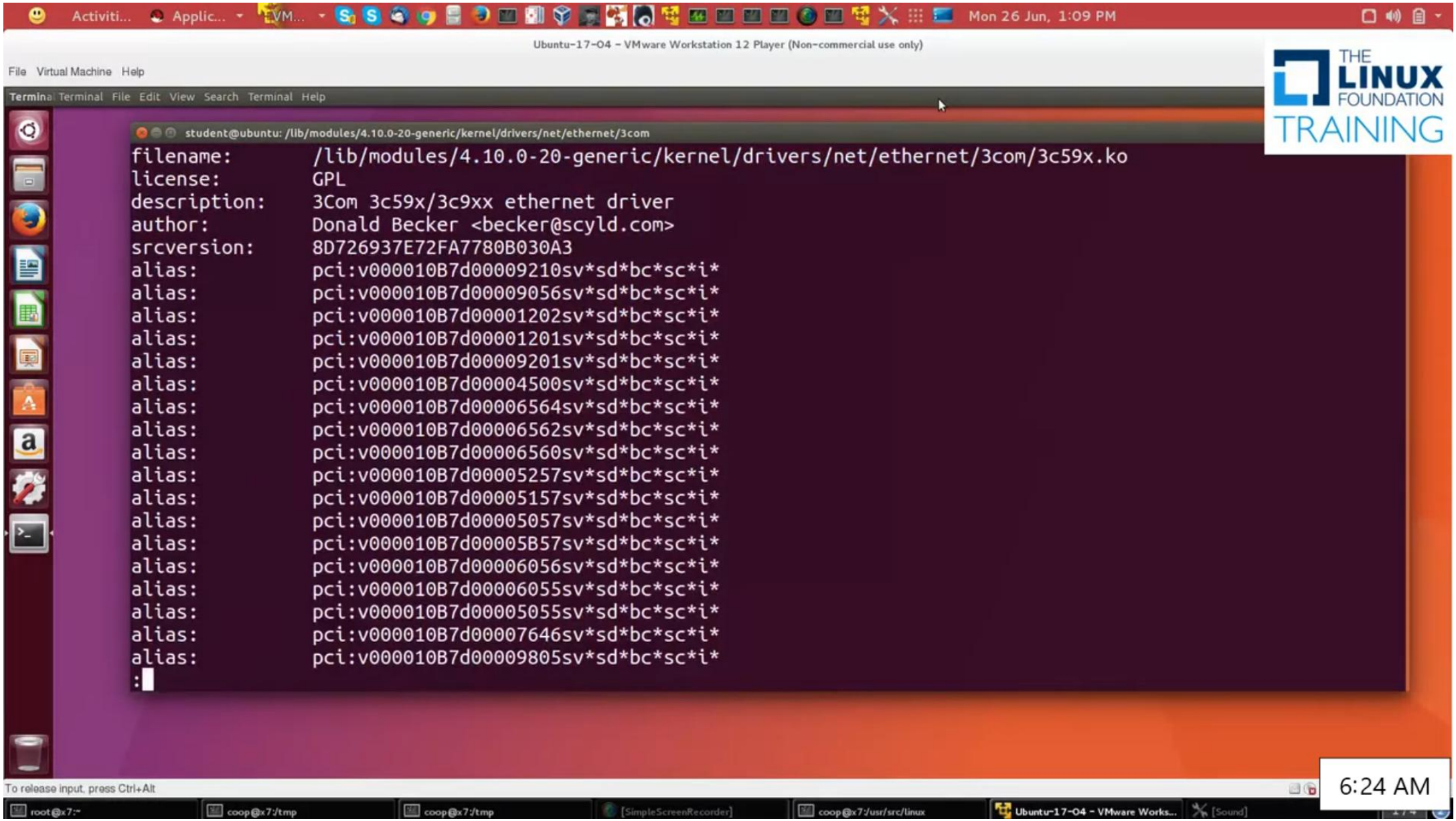
6:21 AM

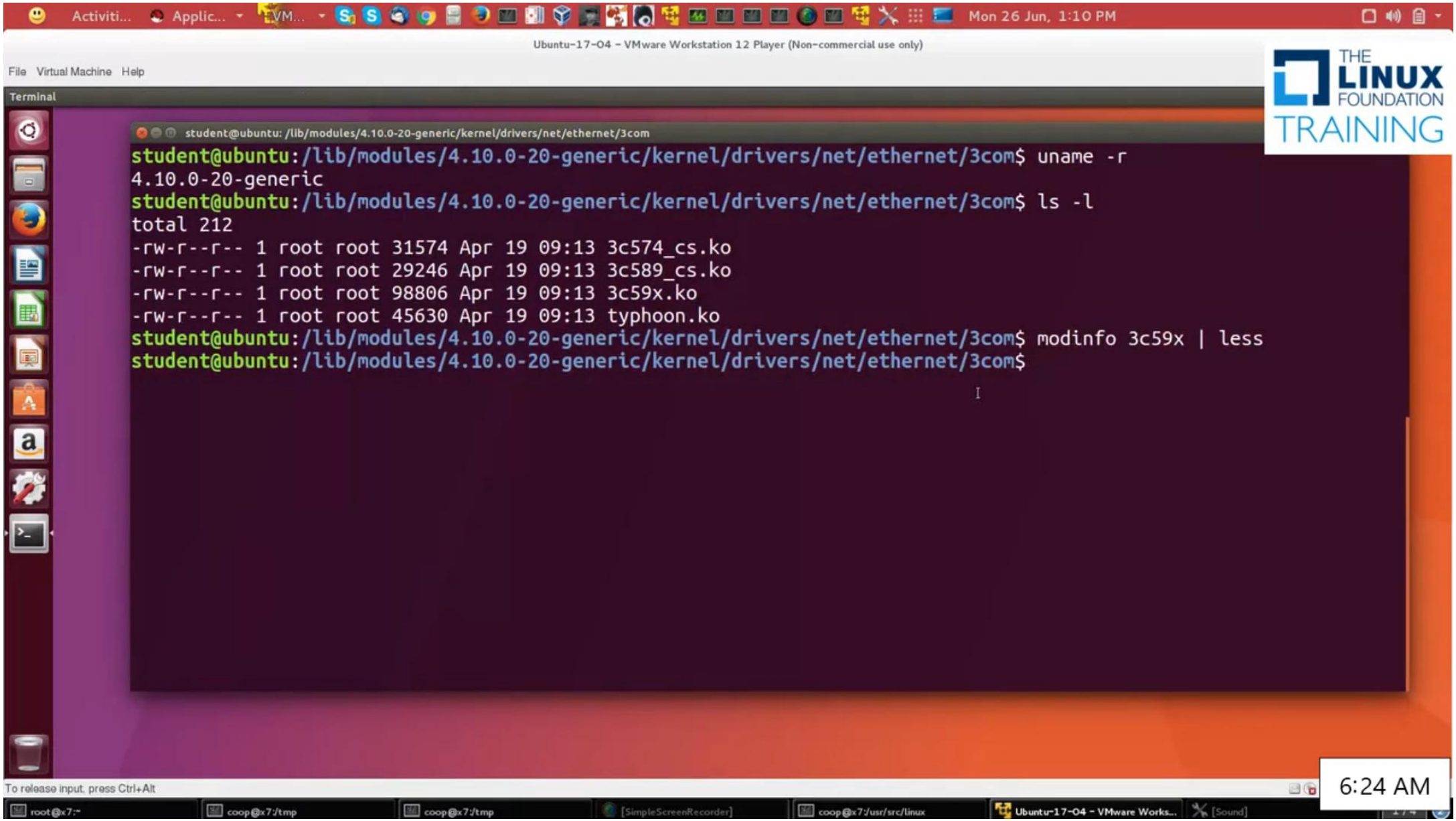

Activities
Applica...
Ter...
Mon 26 Jun, 12:38 PM
Processes
Resources
File Systems
CPU History
CPU1 4.0%
CPU2 3.0%
CPU3 100.0%
CPU4 3.0%
CPU5 2.0%
CPU6 4.0%
CPU7 1.0%
CPU8 5.0%
Memory and Swap History
Memory
10.1 GiB (65.0%) of 15.5 GiB
Swap
1.2 GiB (15.1%) of 7.9 GiB
Network History
Receiving
41.5 KiB/s
Total Received
289.0 MiB
Sending
4.3 MiB/s
Total Sent
17.0 GiB
coop@c7:/tmp
File Edit View Search Terminal Help
top - 12:38:46 up 4:32, 9 users, load average: 1.59, 2.55, 2.65
Tasks: 325 total, 2 running, 323 sleeping, 0 stopped, 0 zombie
%Cpu(s): 5.6 us, 0.8 sy, 0.0 ni, 92.5 id, 1.1 wa, 0.0 hi, 0.0 s
KiB Mem : 16282932 total, 5058244 free, 10483600 used, 741088 buff
KiB Swap: 8290300 total, 7036924 free, 1253376 used. 5150992 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
4857 coop 20 0 9805.8m 7.737g 54896 R 100.0 49.8 24:20.18 gnome-shell
5596 coop 20 0 1389348 148436 46216 S 6.7 0.9 5:08.97 skypeforlinux
1 root 20 0 191352 4176 2800 S 0.0 0.0 0:01.86 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
4 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 kworker/0:0H
6 root 0 -20 0 0 0 S 0.0 0.0 0:00.00 mm_percpu_wq
7 root 20 0 0 0 0 S 0.0 0.0 0:00.09 ksoftirqd/0
8 root 20 0 0 0 0 S 0.0 0.0 0:11.66 rcu_preempt
9 root 20 0 0 0 0 S 0.0 0.0 0:00.00 rcu_sched
10 root 20 0 0 0 0 S 0.0 0.0 0:00.00 rcu_bh
11 root 20 0 0 0 0 S 0.0 0.0 0:02.82 rcuop/0
THE LINUX FOUNDATION TRAINING
coop@c7:/tmp
File Edit View Search Terminal Help
c7:/tmp>
root@c7:/usr/src/linux
File Edit View Search Terminal Help
ROOT@c7:/usr/src/linux>echo 3 > /proc/sys/vm/drop_caches
ROOT@c7:/usr/src/linux>
root@c7:~
coop@c7:/tmp
coop@c7:/tmp
[SimpleScreenRecorder]
root@c7:/usr/src/linux
System Monitor (on c7)
6:21 AM

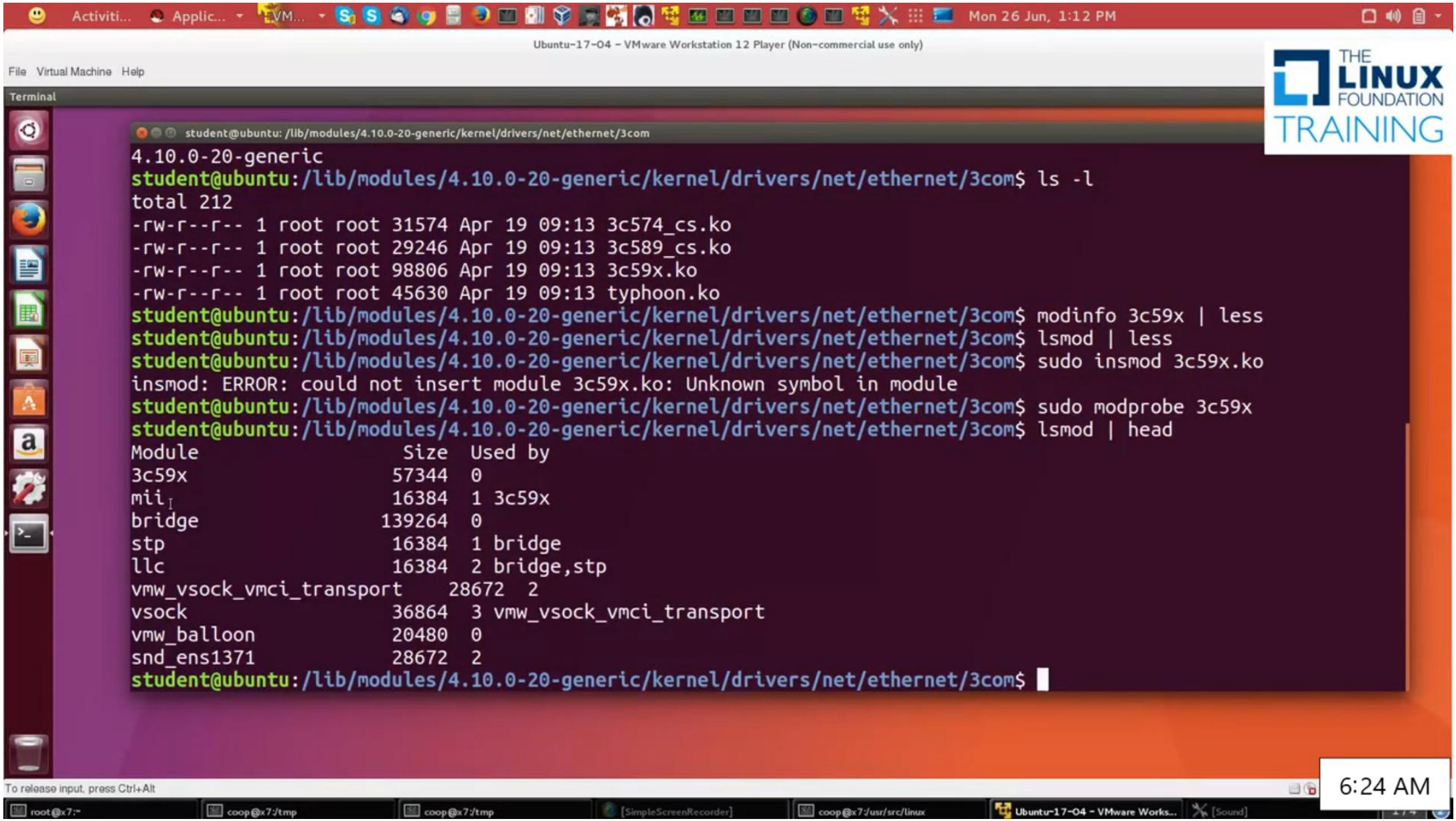

Activities
Applica...
Syst...
Mon 26 Jun, 12:39 PM
Processes
Resources
File Systems
CPU History
CPU1 90.0%
CPU2 100.0%
CPU3 97.0%
CPU4 95.0%
CPU5 87.0%
CPU6 57.1%
CPU7 82.7%
CPU8 91.9%
Memory and Swap History
Memory
10.6 GiB (68.4%) of 15.5 GiB
Swap
1.2 GiB (15.1%) of 7.9 GiB
Network History
Receiving
63.7 KiB/s
Total Received
291.3 MiB
Sending
5.6 MiB/s
Total Sent
17.2 GiB
coop@c7:/tmp
File Edit View Search Terminal Help
top - 12:39:34 up 4:33, 9 users, load average: 1.95, 2.48, 2.62
Tasks: 366 total, 10 running, 356 sleeping, 0 stopped, 0 zombie
%Cpu(s): 80.8 us, 7.3 sy, 0.0 ni, 11.7 id, 0.1 wa, 0.0 hi, 0.1 s
KiB Mem : 16282932 total, 3631244 free, 11011532 used, 1640156 buff
KiB Swap: 8290300 total, 7036924 free, 1253376 used. 4582252 avail Mem
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
4857 coop 20 0 9.889g 8.022g 55208 R 101.3 51.7 25:09.35 gnome-shell
23972 root 20 0 199364 76232 14888 R 34.2 0.5 0:01.03 cc1
24039 root 20 0 185316 60952 14852 R 14.6 0.4 0:00.44 cc1
24065 root 20 0 165872 39648 11008 R 7.0 0.2 0:00.21 cc1
24067 root 20 0 159800 35536 14080 R 6.6 0.2 0:00.20 cc1
15501 coop 20 0 1625284 229496 59336 S 4.0 1.4 5:37.25 chrome
5596 coop 20 0 1389488 148212 46216 R 3.7 0.9 5:10.49 skypeforlinux
25986 root 20 0 682312 78864 62476 S 3.3 0.5 0:14.39 gnome-system-mo
4765 coop 20 0 1153428 1.002g 1032 S 2.7 6.5 0:34.71 dbus-daemon
25886 root 20 0 148852 9216 6936 S 2.0 0.1 0:04.97 sshd
12622 coop 20 0 1395496 172984 86128 S 1.7 1.1 1:27.61 chrome
THE LINUX FOUNDATION TRAINING
coop@c7:/tmp
File Edit View Search Terminal Help
c7:/tmp>vmstat -a 2 1000
procs -----------memory---------- ---swap-- -----io---- -system-- ------cpu-----
r b swpd free inact active si so bi bo in cs us sy id wa st
1 0 1253376 4925164 690732 10443264 8 25 337 181 138 208 6 1 92 1 0
1 0 1253376 4913120 690836 10454880 0 0 64 102 2006 5584 14 1 85 0 0
1 0 1253376 4901340 690896 10467200 0 0 0 174 2058 8369 14 1 85 0 0
3 0 1253376 4883772 694732 10479552 16 0 78 304 2179 7505 14 1 85 0 0
2 0 1253376 4855904 694776 10507068 0 0 0 216 2122 8695 14 1 85 0 0
1 0 1253376 4832704 694876 10529404 0 0 0 88 1990 6641 14 1 85 0 0
1 0 1253376 4822176 694876 10540784 0 0 0 0 1935 7399 14 1 85 0 0
1 0 1253376 4805560 694900 10557944 20 0 20 74 2042 6085 14 1 81 4 0
1 0 1253376 4792228 694948 10571184 0 0 0 44 1977 7885 14 1 86 0 0
2 3 1253376 4692120 730884 10632504 2 0 39006 212 6002 9696 17 3 79 1 0
3 0 1253376 4001652 1313400 10683140 0 0 303766 870 9881 22187 44 12 36 8 0
7 0 1253376 3804788 1411768 10767396 0 0 55114 222 6830 11922 37 6 56 1 0
9 0 1253376 3638376 1424040 10921048 54 0 1236 2526 8423 8611 74 10 15 0 0
9 0 1253376 3732512 1418248 10831568 0 0 702 388 8554 4981 81 7 12 0 0
7 0 1253376 3640480 1421984 10920700 0 0 804 378 8600 6502 81 8 12 0 0
7 0 1253376 3685584 1422488 10873636 0 0 550 2916 8837 6328 80 8 12 0 0
root@c7:/usr/src/linux
File Edit View Search Terminal Help
CC mm/shmem.o
CC mm/util.o
CC mm/mmzone.o
CC kernel/kmod.o
CC arch/x86/kernel/espfix_64.o
CC arch/x86/kernel/ksysfs.o
CC arch/x86/kernel/bootflag.o
CC arch/x86/kernel/e820.o
CC mm/vmstat.o
CC kernel/workqueue.o
CC kernel/pid.o
CC kernel/task_work.o
CC arch/x86/kernel/pci-dma.o
root@c7:~
coop@c7:/tmp
coop@c7:/tmp
[SimpleScreenRecorder]
root@c7:/usr/src/linux
System Monitor (on c7)
6:21 AM


Activiti... Applic... VM...
Mon 26 Jun, 1:09 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal File Edit View Search Terminal Help
student@ubuntu: /lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ uname -r
4.10.0-20-generic
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ ls -l
total 212
-rw-r--r-- 1 root root 31574 Apr 19 09:13 3c574_cs.ko
-rw-r--r-- 1 root root 29246 Apr 19 09:13 3c589_cs.ko
-rw-r--r-- 1 root root 98806 Apr 19 09:13 3c59x.ko
-rw-r--r-- 1 root root 45630 Apr 19 09:13 typhoon.ko
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ modinfo 3
THE LINUX FOUNDATION TRAINING
To release input, press Ctrl+Alt
6:24 AM


Activiti... Applic... VM...
Mon 26 Jun, 1:09 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal File Edit View Search Terminal Help
student@ubuntu: /lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ uname -r
4.10.0-20-generic
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ ls -l
total 212
-rw-r--r-- 1 root root 31574 Apr 19 09:13 3c574_cs.ko
-rw-r--r-- 1 root root 29246 Apr 19 09:13 3c589_cs.ko
-rw-r--r-- 1 root root 98806 Apr 19 09:13 3c59x.ko
-rw-r--r-- 1 root root 45630 Apr 19 09:13 typhoon.ko
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ modinfo 3c59x | less
THE LINUX FOUNDATION TRAINING
To release input, press Ctrl+Alt
root@x7:~
coop@x7:/tmp
coop@x7:/tmp
[SimpleScreenRecorder]
coop@x7:/usr/src/linux
Ubuntu-17-04 - VMware Works...
[Sound]
6:24 AM


Activiti...
Applic...
VM...
Mon 26 Jun, 1:09 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal File Edit View Search Terminal Help
student@ubuntu: /lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com
filename: /lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com/3c59x.ko
license: GPL
description: 3Com 3c59x/3c9xx ethernet driver
author: Donald Becker <becker@scyld.com>
srcversion: 8D726937E72FA7780B030A3
alias: pci:v000010B7d00009210sv*sd*bc*sc*i*
alias: pci:v000010B7d00009056sv*sd*bc*sc*i*
alias: pci:v000010B7d00001202sv*sd*bc*sc*i*
alias: pci:v000010B7d00001201sv*sd*bc*sc*i*
alias: pci:v000010B7d00009201sv*sd*bc*sc*i*
alias: pci:v000010B7d00004500sv*sd*bc*sc*i*
alias: pci:v000010B7d00006564sv*sd*bc*sc*i*
alias: pci:v000010B7d00006562sv*sd*bc*sc*i*
alias: pci:v000010B7d00006560sv*sd*bc*sc*i*
alias: pci:v000010B7d00005257sv*sd*bc*sc*i*
alias: pci:v000010B7d00005157sv*sd*bc*sc*i*
alias: pci:v000010B7d00005057sv*sd*bc*sc*i*
alias: pci:v000010B7d00005B57sv*sd*bc*sc*i*
alias: pci:v000010B7d00006056sv*sd*bc*sc*i*
alias: pci:v000010B7d00006055sv*sd*bc*sc*i*
alias: pci:v000010B7d00005055sv*sd*bc*sc*i*
alias: pci:v000010B7d00007646sv*sd*bc*sc*i*
alias: pci:v000010B7d00009805sv*sd*bc*sc*i*
:
THE LINUX FOUNDATION TRAINING
To release input, press Ctrl+Alt
root@x7:~
coop@x7:/tmp
coop@x7:/tmp
[SimpleScreenRecorder]
coop@x7:/usr/src/linux
Ubuntu-17-04 - VMware Works...
[Sound]
6:24 AM


Activiti... Applic... VM...
Mon 26 Jun, 1:10 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal
student@ubuntu: /lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ uname -r
4.10.0-20-generic
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ ls -l
total 212
-rw-r--r-- 1 root root 31574 Apr 19 09:13 3c574_cs.ko
-rw-r--r-- 1 root root 29246 Apr 19 09:13 3c589_cs.ko
-rw-r--r-- 1 root root 98806 Apr 19 09:13 3c59x.ko
-rw-r--r-- 1 root root 45630 Apr 19 09:13 typhoon.ko
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ modinfo 3c59x | less
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$
THE LINUX FOUNDATION TRAINING
To release input, press Ctrl+Alt
root@x7:~
coop@x7:/tmp
coop@x7:/tmp
[SimpleScreenRecorder]
coop@x7:/usr/src/linux
Ubuntu-17-04 - VMware Works...
[Sound]
6:24 AM


Activiti... Applic... VM... Mon 26 Jun, 1:12 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal
student@ubuntu: /lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com
4.10.0-20-generic
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ ls -l
total 212
-rw-r--r-- 1 root root 31574 Apr 19 09:13 3c574_cs.ko
-rw-r--r-- 1 root root 29246 Apr 19 09:13 3c589_cs.ko
-rw-r--r-- 1 root root 98806 Apr 19 09:13 3c59x.ko
-rw-r--r-- 1 root root 45630 Apr 19 09:13 typhoon.ko
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ modinfo 3c59x | less
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ lsmod | less
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ sudo insmod 3c59x.ko
insmod: ERROR: could not insert module 3c59x.ko: Unknown symbol in module
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ sudo modprobe 3c59x
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$ lsmod | head
Module Size Used by
3c59x 57344 0
mii 16384 1 3c59x
bridge 139264 0
stp 16384 1 bridge
llc 16384 2 bridge,stp
vmw_vsock_vmci_transport 28672 2
vsock 36864 3 vmw_vsock_vmci_transport
vmw_balloon 20480 0
snd_ens1371 28672 2
student@ubuntu:/lib/modules/4.10.0-20-generic/kernel/drivers/net/ethernet/3com$
THE LINUX FOUNDATION TRAINING
To release input, press Ctrl+Alt
root@x7:~ coop@x7:/tmp coop@x7:/tmp [SimpleScreenRecorder] coop@x7:/usr/src/linux Ubuntu-17-04 - VMware Works... [Sound]
6:24 AM

Types of Devices

There are three main types of devices:

- **Character devices** are sequential streams; they mainly implement **open**, **close**, **read**, and **write** functions, e.g. serial and parallel ports (**`/dev/ttyS0`**, **`/dev/lp1`**), sound cards (**`/dev/dsp0`**), etc.
- **Block devices** are randomly accessed only in block-size multiples, and I/O operations are usually cached and deal with mounted filesystems, e.g. hard drive partitions (**`/dev/sda1`**, **`/dev/sdb8`**), CD-ROMs, etc.
- **Network devices** transfer packets of data, not blocks or streams, and usually employ a socket interface; packet reception/transmission functions replace read/write operations, and there are no corresponding filesystem nodes; instead, the interfaces are identified by a name, such as **`eth0`** or **`wlan0`**.

Other Types of Devices

- There are also other types of devices which are classified somewhat differently, according to the type of controller bus they are attached to, such as **SCSI** (**S**mall **C**omputers **S**ystems **I**nterconnect) and **USB** (**U**niversal **S**erial **B**us)
- These devices share an underlying protocol regardless of function
- Besides the driver for the device itself, hard work goes into writing the driver for the controller hardware which may run many devices
- One also has **user-space drivers**, which work completely in user-space, but are given hardware access privileges; these are not as efficient as in-kernel drivers, but are less likely to bring a system down

Other Types of Devices

- There are also other types of devices which are classified somewhat differently, according to the type of controller bus they are attached to, such as **SCSI** (**S**mall **C**omputers **S**ystems **I**nterconnect) and **USB** (**U**niversal **S**erial **B**us)
- These devices share an underlying protocol regardless of function
- Besides the driver for the device itself, hard work goes into writing the driver for the controller hardware which may run many devices
- One also has **user-space drivers**, which work completely in user-space, but are given hardware access privileges; these are not as efficient as in-kernel drivers, but are less likely to bring a system down

Device Nodes

- Character and block devices have filesystem entries associated with them
- These **nodes** can be used by user-level programs to communicate with the device, using normal I/O system calls such as **`open()`**, **`close()`**, **`read()`**, and **`write()`**
- A device driver can manage more than one device node

/dev Directory

- Device nodes are normally placed in the **/dev** directory and can be created with:

  ```
  $ sudo mknod [-m mode] /dev/name <type> <major> <minor>
  ```

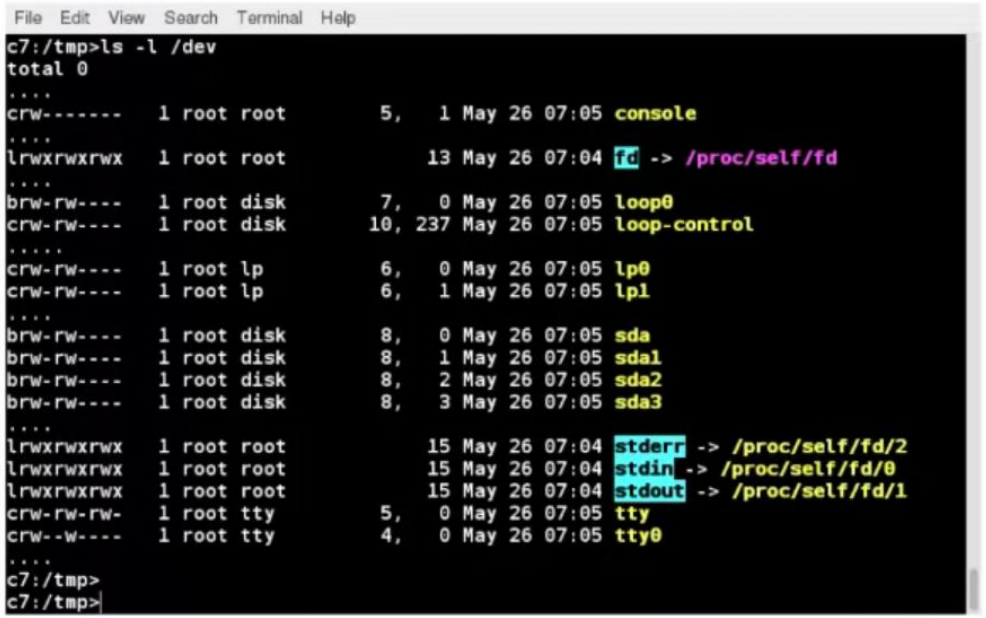
  e.g.

  ```
  $ mknod -m 666 /dev/mycdrv c 254 1
  ```

  or from the **mknod()** system call
- The **major** number identifies the driver associated with the device; all device nodes of the same type (block or character) with the same major number use the same driver
- The **minor** number is used only by the device driver to differentiate between the different devices it may control
  - Either different instances of the same kind of device, (such as the first and second sound card, or hard disk partition) or
  - Different modes of operation of a given device (e.g. different density floppy drive media)

/dev Directory (Cont.)

- Device numbers have meaning in user-space as well
- Two POSIX system calls, **`mknod()`** and **`stat()`** return information about major and minor numbers



/dev Directory

Managing Device Nodes

- The methods of managing device nodes became clumsy and difficult as Linux evolved
- The number of device nodes lying in **`/dev`** and its subdirectories reached numbers in the 15,000 - 20,000 range in most installations during the 2.4 kernel series
- Nodes for all kinds of devices which would never be used on most installations were still created by default, as distributors could never be sure exactly which hardware would be needed
- Many developers and system administrators trimmed the list to what was actually needed, especially in embedded configurations, but this was essentially a manual and potentially error-prone task

Managing Device Nodes (Cont.)

- Note that while device nodes are not normal files and do not take up significant space on the filesystem, having huge directories slowed down access to device nodes, especially upon first usage
- Furthermore, exhaustion of available major and minor numbers required a more modern and dynamic approach to the creation and maintenance of device nodes
- Ideally, one would like to register devices by name; however, major and minor numbers cannot be gotten rid of altogether, as POSIX requires them

udev

- The **udev** method creates device nodes on the fly as they are needed; there is no need to maintain a ton of device nodes that will never be used
- The **u** in udev stands for user, and indicates that most of the work of creating, removing, and modifying devices nodes is done in user-space
- udev handles the dynamical generation of device nodes and it evolved to replace earlier mechanisms such as devfs and hotplug
- It supports persistent device naming; names need not depend on the order of device connection or plugging in - such behavior is controlled by specification of udev rules
- udev runs as a **daemon** and monitors a **netlink** socket
  - When new devices are initialized or removed the uevent kernel facility sends a message through the socket which udev receives and takes appropriate action to create or remove device nodes of the right names and properties according to the rules

udev Components

The three components of udev are:

- The **libudev** library which allows access to information about the devices
- The **udevd** daemon that manages the **`/dev`** directory
- The **udevadm** utility for control and diagnostics

Creating and Removing Device Nodes

- As devices are added or removed from the system, working with the hotplug subsystem, udev acts upon notification of events to create and remove device nodes
- The information necessary to create them with the right names, major and minor numbers, permissions, etc., are gathered by examination of information already registered in the **sysfs** pseudo-filesystem (mounted at **`/sys`**) and a set of configuration files
- The main configuration file is **`/etc/udev/udev.conf`**; it contains information such as where to place device nodes, default permissions and ownership, etc.
- By default, rules for device naming are located in the **`/etc/udev/rules.d`** directory
- By reading the **man** page for udev, one can get a lot of specific information about how to set up rules for common situations

Installation

- In the early days of Linux, installation was a back-breaking process, involving downloading dozens of floppy disks (through slow telephone lines) and loading them one by one
- Eventually, distributors developed CD-based installations which were far more user-friendly
- As the size of the installation gradually expanded, DVD-based methods also became widespread, and USB-based methods are now probably the most frequent

Installation Choices

- The early incarnations of installations based on optical media often presented one with many configurable choices, and many still do, especially as regards software selection
- Unlike vendors of other operating systems, Linux distributors offer not just the basic operating system and utilities, but also a wide range of applications and utilities
- On other operating systems, these ingredients are separately installed (after the basic system setup) by downloading and/or purchasing from a potentially large number of vendors
- Installation choices start with which desktop manager to use (usually GNOME or KDE), etc., and then cascade into many other alternatives

Installation Choices (Cont.)

r has neither the knowledge or experience to evaluate
d consequences of these choices, much less their long

s have attempted to limit the number of choices made
a small number of basic questions, and to have the default
mmonly useful ones

on can be made post-install through the use of the various
gement systems that are in common use

- Sometimes Linux is insta
systems
- In this case, an essential
enough free space for th
- The most user-friendly ir
process, by running gpa
program

6:29 AM

Multi-boot Environment

- Sometimes Linux is installed in a multi-boot environment alongside other operating systems
- In this case, an essential step is to rework the disk partitioning scheme to open up enough free space for the installation
- The most user-friendly installation media offer this as part of the initial installation process, by running **gparted**, a widely used graphical partition manipulation program

Live CDs, DVDs and USBs

- Virtually all major Linux distributors offer live CD, DVD and/or USB versions
- These make it possible for you to try out Linux without actually touching your hard disk
- It is always a good idea to try this first as these disks do a great job of detecting and configuring hardware and can help you avoid problems that may arise later when doing the real installation
- The live media usually include a copy of disk partitioning utilities, including gparted
- Finally, the live media always have an *Install* icon you can click on, once you have made sure your hardware can be handled by the Linux distribution

Live CDs, DVDs and USBs (Cont.)

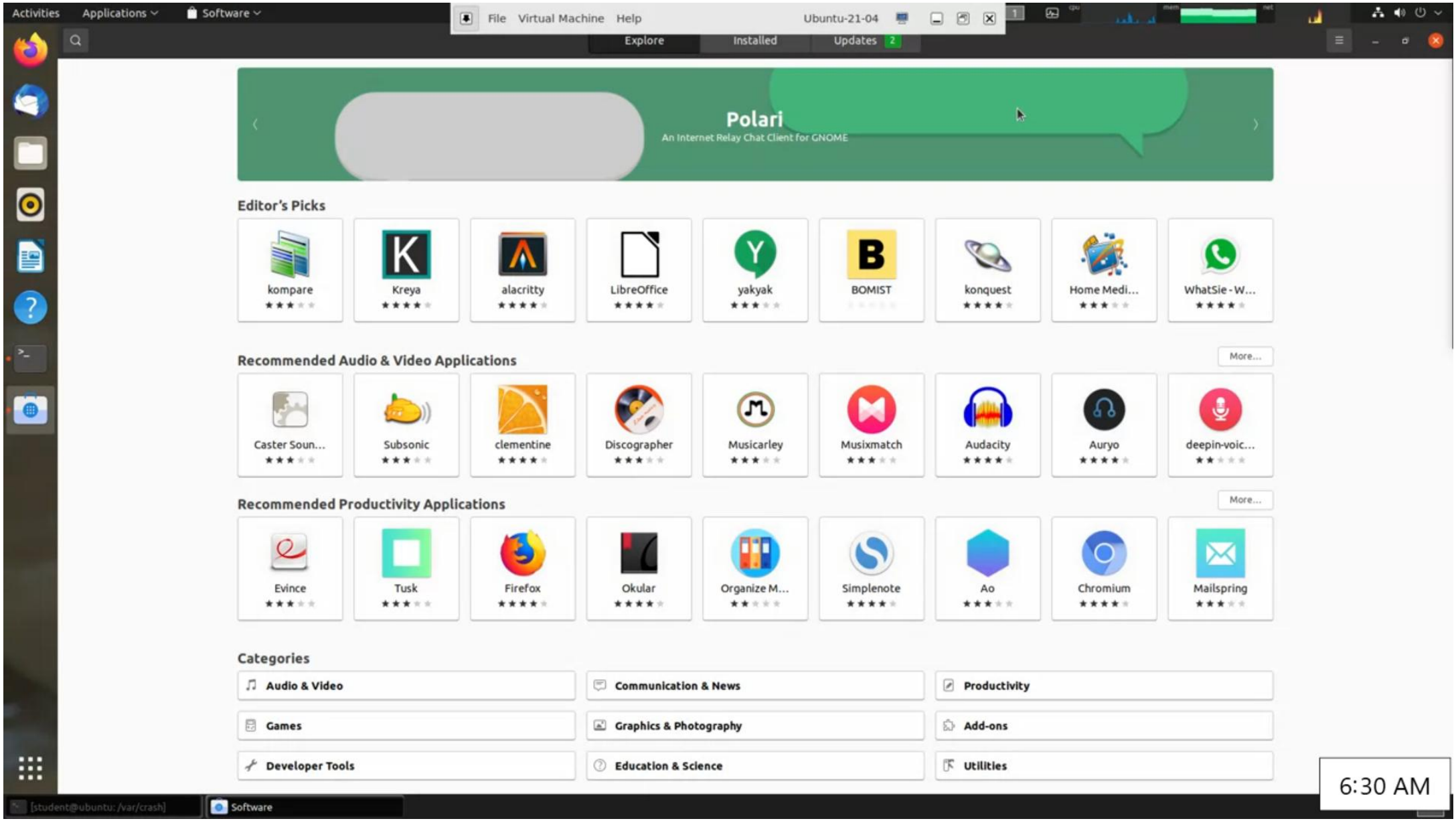
- This gives you a chance to play with Linux before doing the real installation, as they include all major applications you are likely to need
- If you use live media, performance can be slow, and more memory might be needed than is available
- It is even possible to never install Linux on the hard disk, but be able to save your state and work on a file or partition on the disk, or on removable media such as USB disks

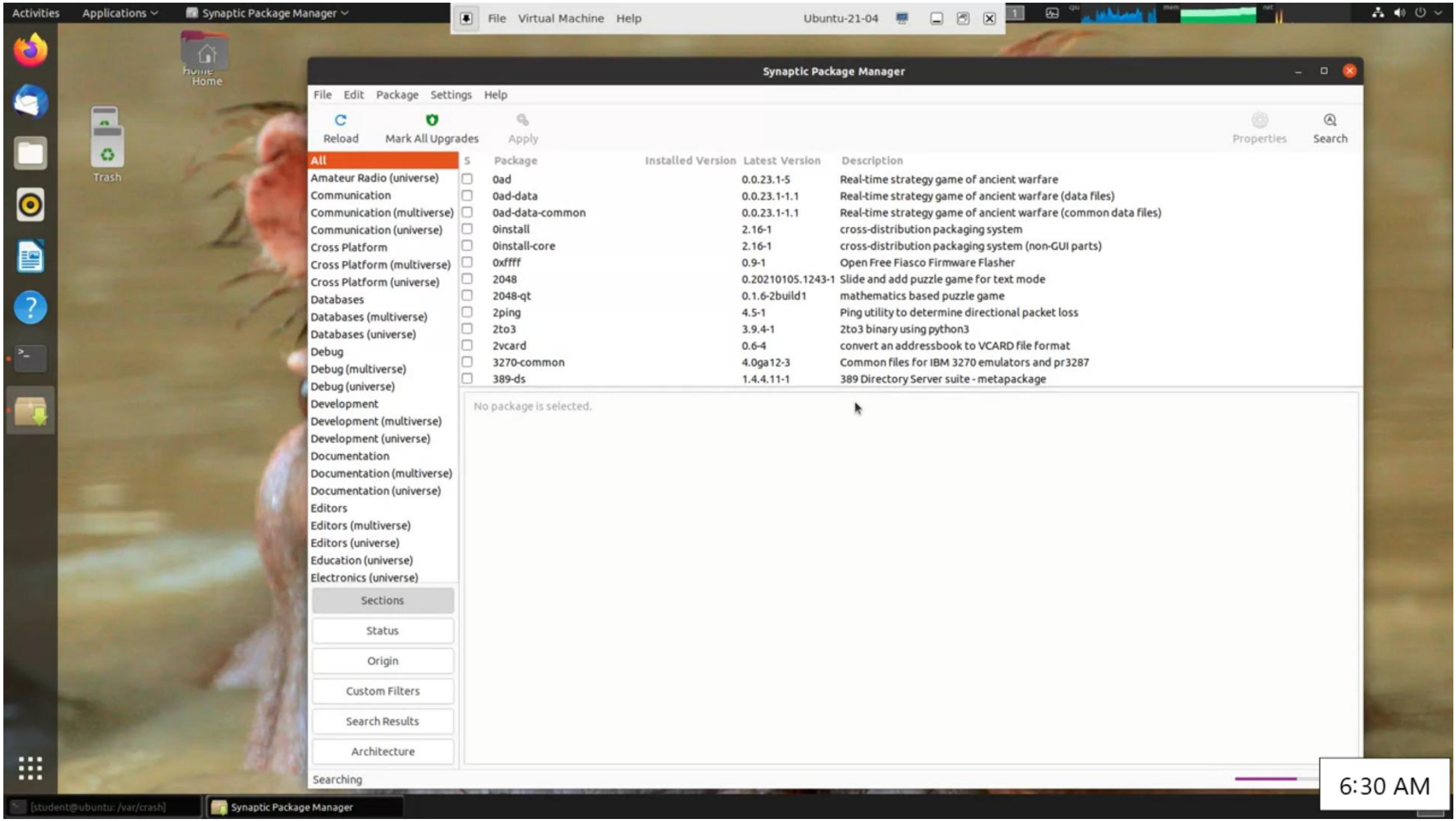
Network-based Installation

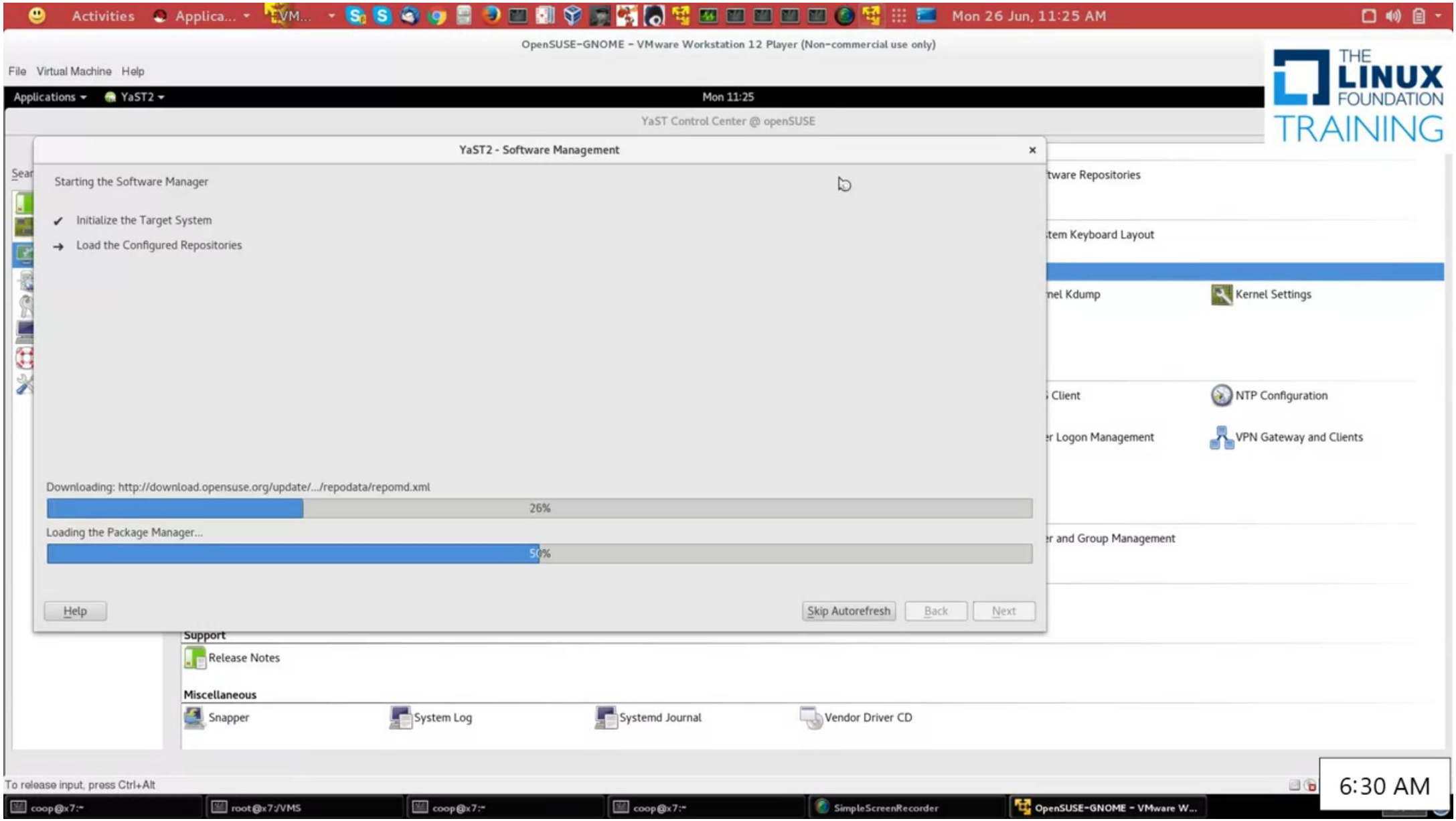
- Many distributions offer the option of doing a network-based installation, where you boot either off a small CD or USB stick with just a few files, or from within another operating system, and then download the remaining files required through the Internet
- In fact, you can often choose to do a network-based installation after booting off a full installation CD
- This is often useful when performing either a simultaneous installation on a number of machines, such as in a corporate environment, or a classroom, or when requiring a standard image, or set of software and system administration features
  - Red Hat-based systems often use the **kickstart** utility where one constructs a script itemizing the disk partitioning, software selection etc., and then stores that on the central server which holds the installation media
- This can also be combined with **PXE** (**P**reboot e**X**ecution **E**nvironment) to boot using the network interface alone without requiring the insertion of any boot media

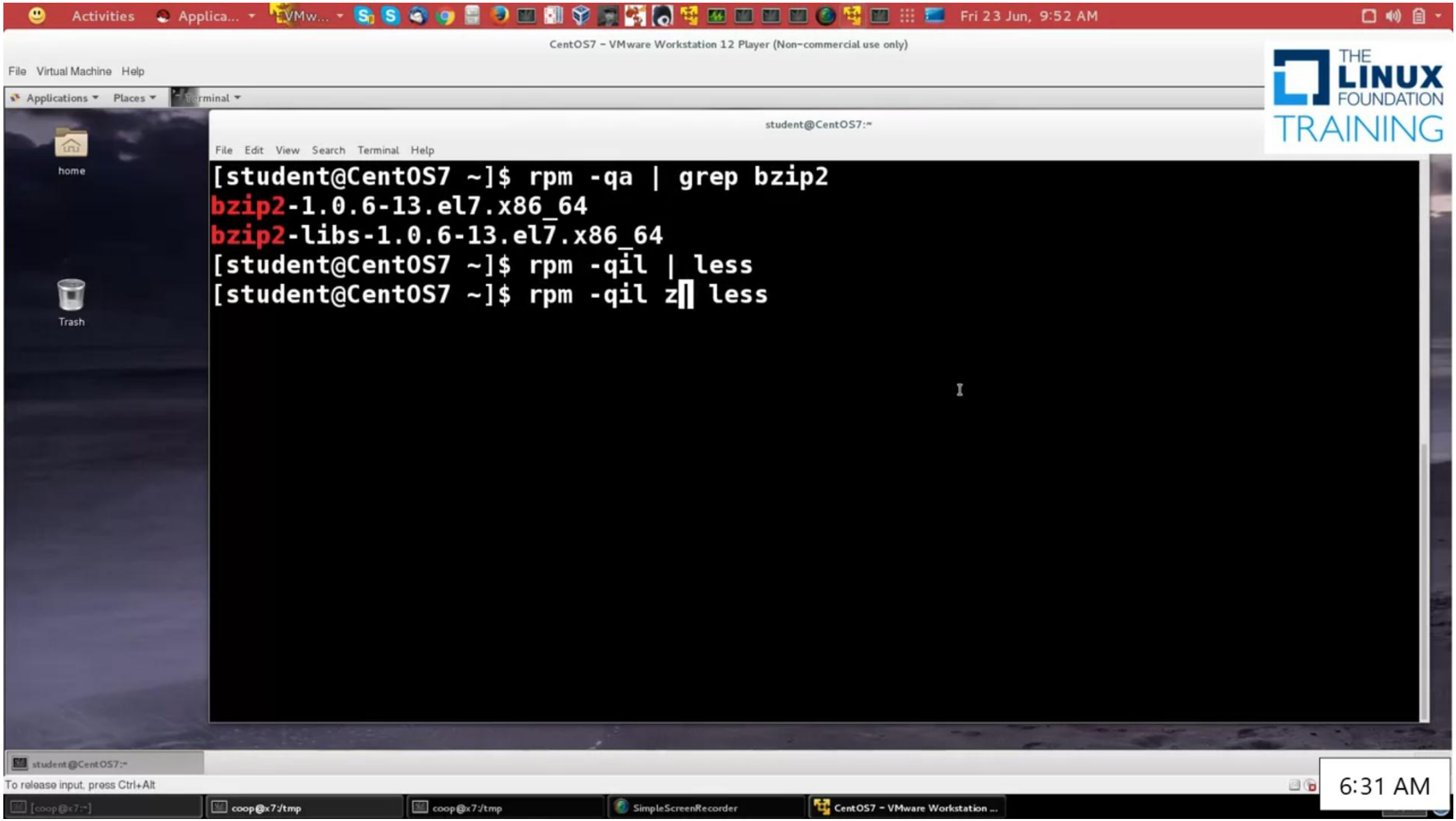
Linux Installation Guide

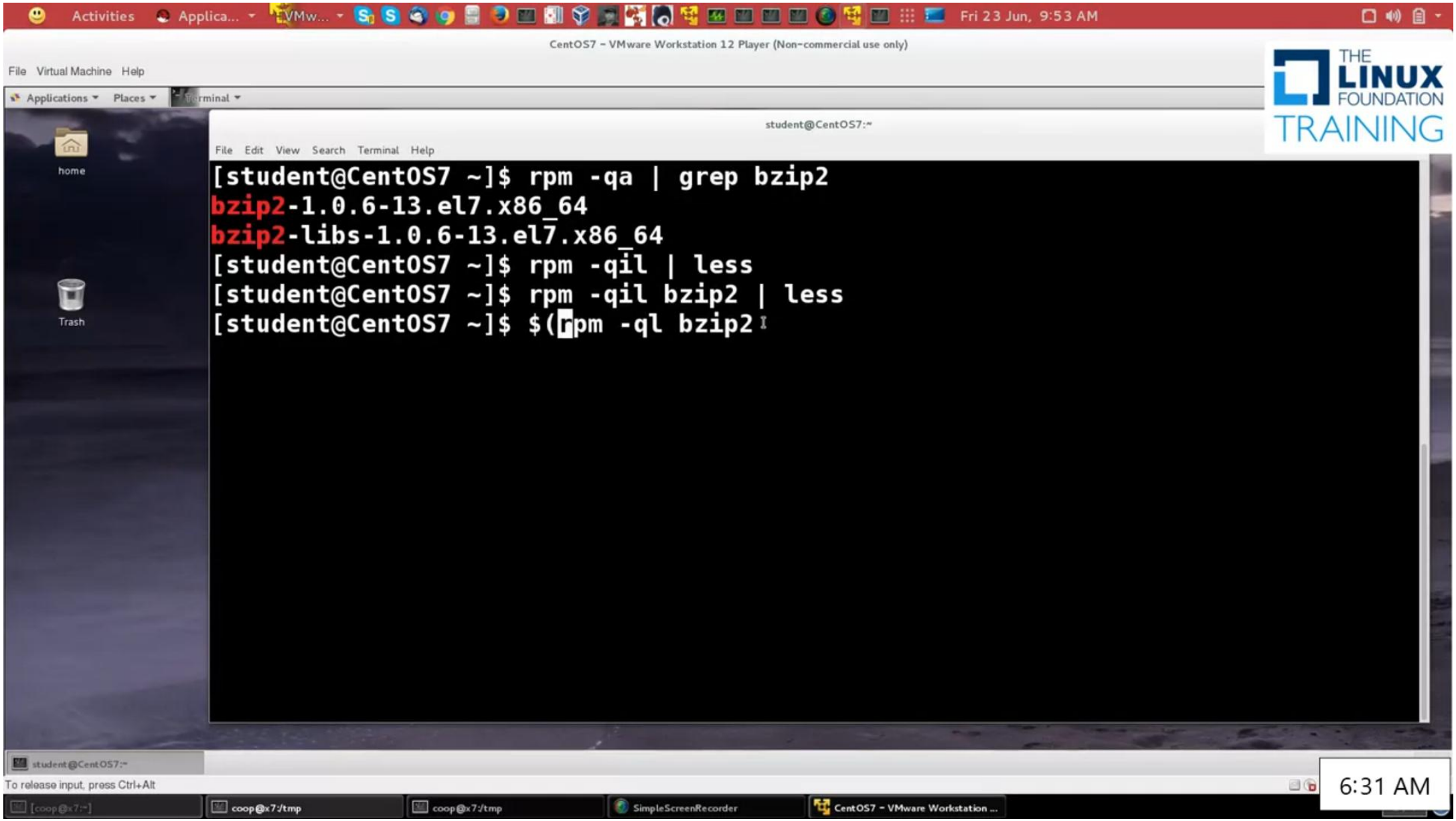
- For a detailed guide to approaching and accomplishing Linux installation in a variety of scenarios, see the guide prepared by The Linux Foundation
- As explained in this document, one can also do an install of a Virtual Machine (VM) using a hypervisor such as Oracle VirtualBox or those distributed by VMWare or Linux's native hypervisor, KVM
- This is less dangerous and avoids a dual boot system, at a cost of reduced performance; however, for this course, such a virtual machine is more than adequate

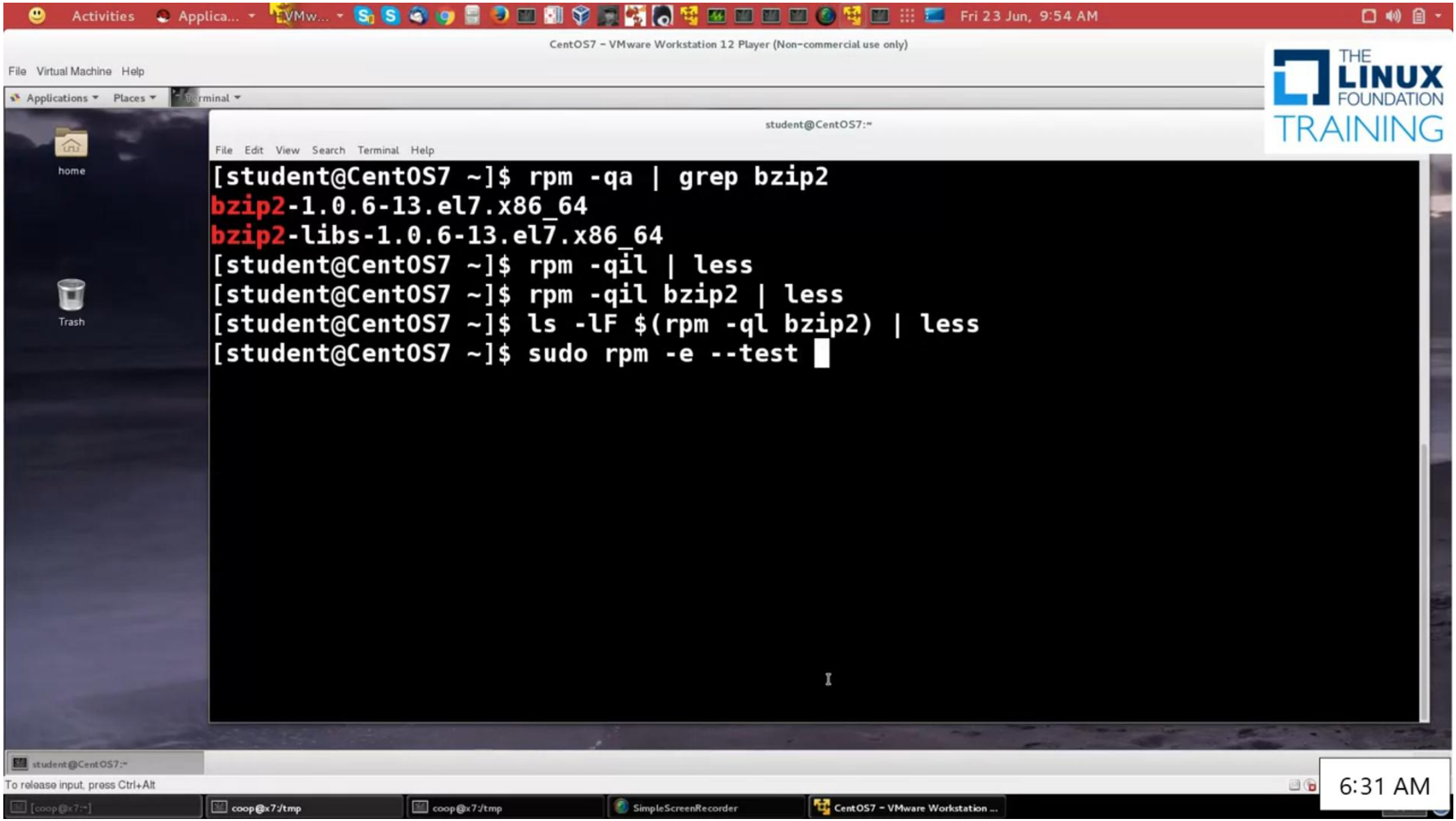

Activities
Applications
Software
File
Virtual Machine
Help
Ubuntu-21-04
Explore
Installed
Updates 2
Polari
An Internet Relay Chat Client for GNOME
Editor's Picks
kompare
Kreya
alacritty
LibreOffice
yakyak
BOMIST
konquest
Home Medi...
WhatSie - W...
Recommended Audio & Video Applications
More...
Caster Soun...
Subsonic
clementine
Discographer
Musicarley
Musixmatch
Audacity
Auryo
deepin-voic...
Recommended Productivity Applications
More...
Evince
Tusk
Firefox
Okular
Organize M...
Simplenote
Ao
Chromium
Mailspring
Categories
Audio & Video
Communication & News
Productivity
Games
Graphics & Photography
Add-ons
Developer Tools
Education & Science
Utilities
[student@ubuntu: /var/crash]
Software
6:30 AM

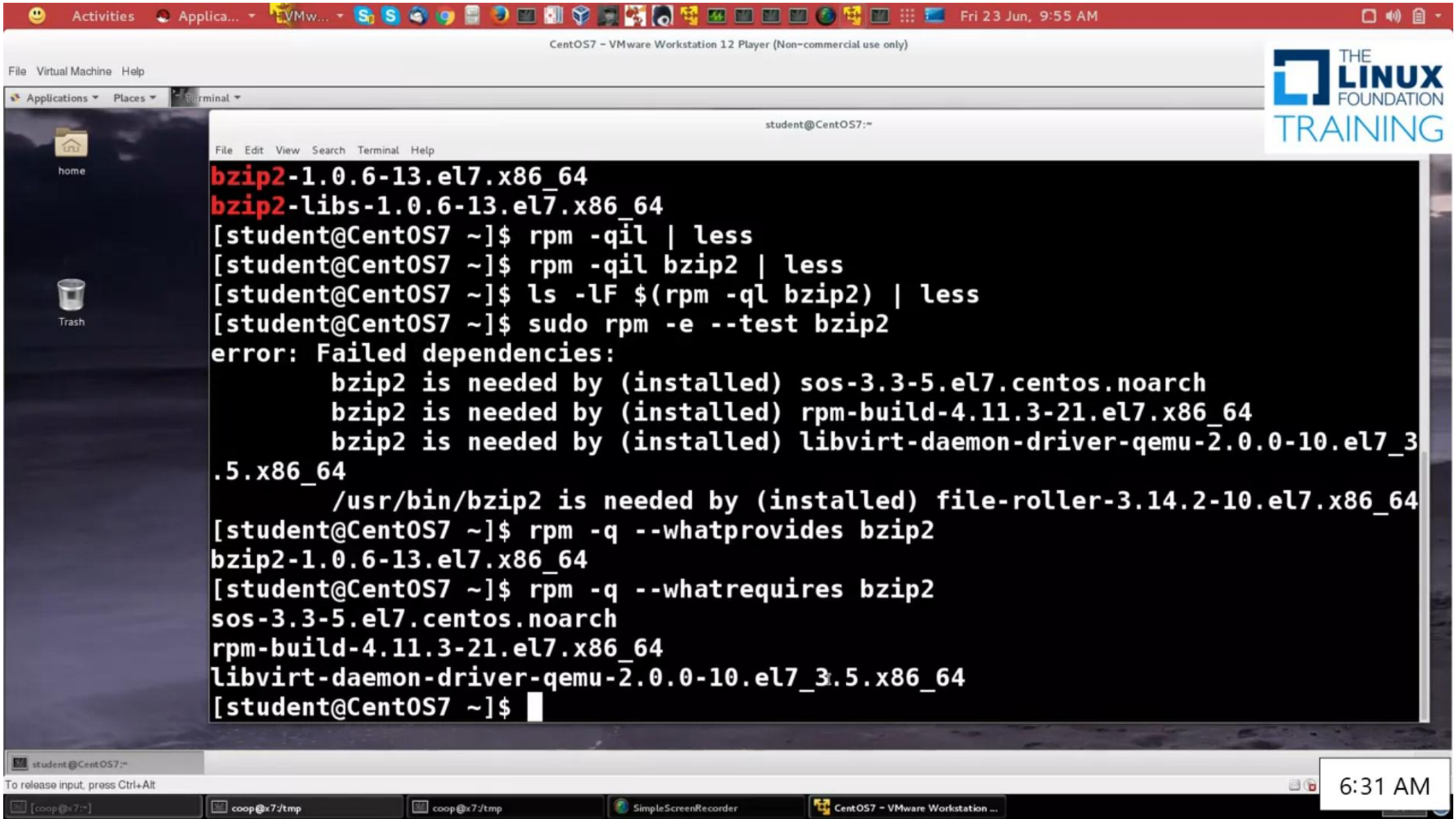

Activities
Applications
Synaptic Package Manager
File Virtual Machine Help
Ubuntu-21-04
Home
Trash
Synaptic Package Manager
File Edit Package Settings Help
Reload
Mark All Upgrades
Apply
Properties
Search
All
Amateur Radio (universe)
Communication
Communication (multiverse)
Communication (universe)
Cross Platform
Cross Platform (multiverse)
Cross Platform (universe)
Databases
Databases (multiverse)
Databases (universe)
Debug
Debug (multiverse)
Debug (universe)
Development
Development (multiverse)
Development (universe)
Documentation
Documentation (multiverse)
Documentation (universe)
Editors
Editors (multiverse)
Editors (universe)
Education (universe)
Electronics (universe)
Sections
Status
Origin
Custom Filters
Search Results
Architecture
S Package Installed Version Latest Version Description
0ad 0.0.23.1-5 Real-time strategy game of ancient warfare
0ad-data 0.0.23.1-1.1 Real-time strategy game of ancient warfare (data files)
0ad-data-common 0.0.23.1-1.1 Real-time strategy game of ancient warfare (common data files)
0install 2.16-1 cross-distribution packaging system
0install-core 2.16-1 cross-distribution packaging system (non-GUI parts)
0xffff 0.9-1 Open Free Fiasco Firmware Flasher
2048 0.20210105.1243-1 Slide and add puzzle game for text mode
2048-qt 0.1.6-2build1 mathematics based puzzle game
2ping 4.5-1 Ping utility to determine directional packet loss
2to3 3.9.4-1 2to3 binary using python3
2vcard 0.6-4 convert an addressbook to VCARD file format
3270-common 4.0ga12-3 Common files for IBM 3270 emulators and pr3287
389-ds 1.4.4.11-1 389 Directory Server suite - metapackage
No package is selected.
Searching
[student@ubuntu: /var/crash]
Synaptic Package Manager
6:30 AM

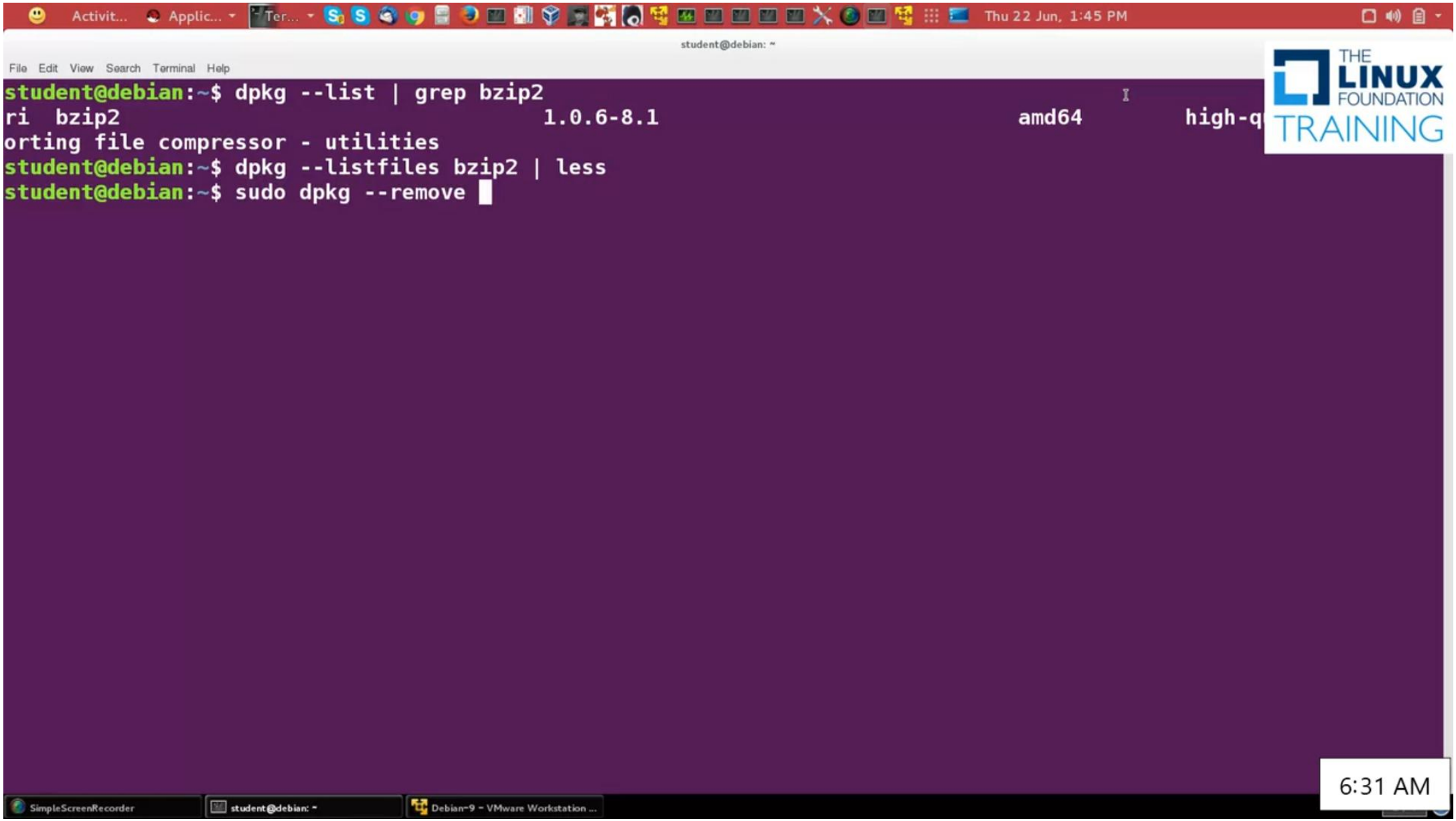

Activities
Applica...
VM...
Mon 26 Jun, 11:25 AM
OpenSUSE-GNOME - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
YaST2
Mon 11:25
YaST Control Center @ openSUSE
YaST2 - Software Management
Starting the Software Manager
Initialize the Target System
Load the Configured Repositories
Downloading: http://download.opensuse.org/update/.../repodata/repomd.xml
26%
Loading the Package Manager...
50%
Help
Skip Autorefresh
Back
Next
Kernel Settings
NTP Configuration
VPN Gateway and Clients
Support
Release Notes
Miscellaneous
Snapper
System Log
Systemd Journal
Vendor Driver CD
To release input, press Ctrl+Alt
coop@x7:~
root@x7:/VMS
SimpleScreenRecorder
OpenSUSE-GNOME - VMware W...
6:30 AM

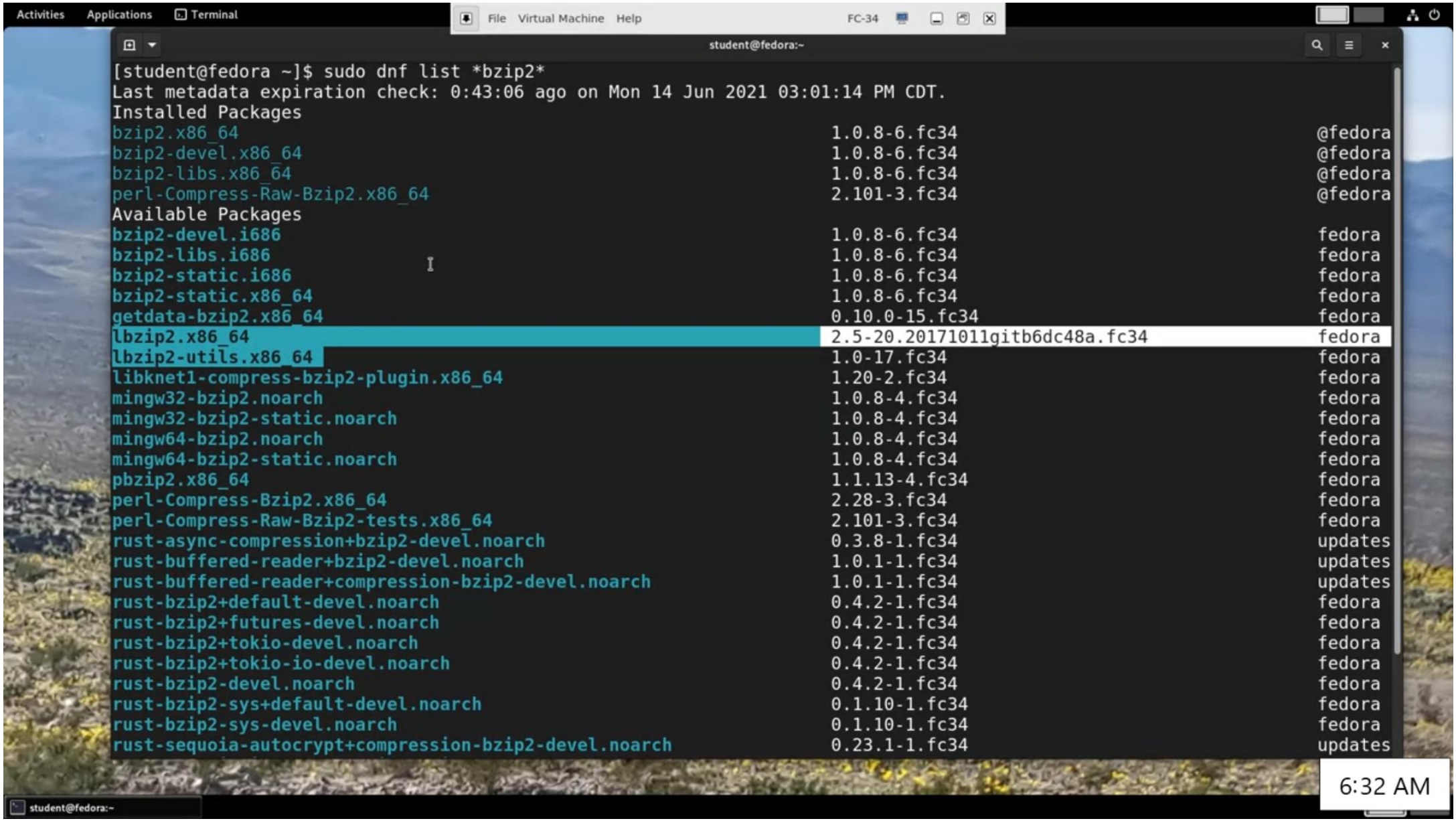

Activities
Applica...
VMw...
Fri 23 Jun, 9:52 AM
CentOS7 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
Places
Terminal
THE LINUX FOUNDATION TRAINING
home
Trash
student@CentOS7:~
File Edit View Search Terminal Help
[student@CentOS7 ~]$ rpm -qa | grep bzip2
bzip2-1.0.6-13.el7.x86_64
bzip2-libs-1.0.6-13.el7.x86_64
[student@CentOS7 ~]$ rpm -qil | less
[student@CentOS7 ~]$ rpm -qil z less
student@CentOS7:~
To release input, press Ctrl+Alt
[coop@x7:~]
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
CentOS7 - VMware Workstation ...
6:31 AM

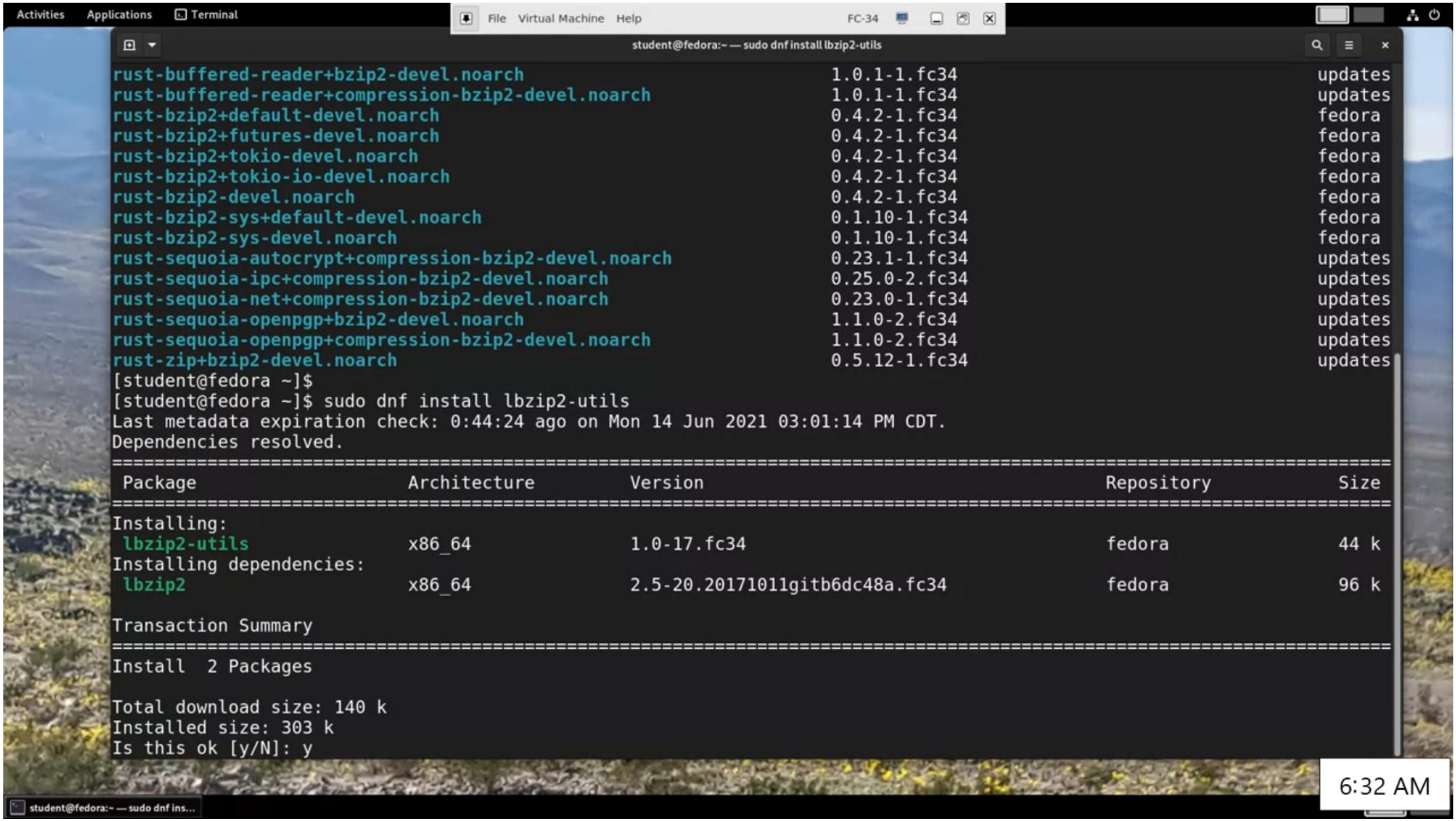

Activities
Applica...
VMw...
Fri 23 Jun, 9:53 AM
CentOS7 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
Places
Terminal
THE LINUX FOUNDATION TRAINING
home
Trash
student@CentOS7:~
File Edit View Search Terminal Help
[student@CentOS7 ~]$ rpm -qa | grep bzip2
bzip2-1.0.6-13.el7.x86_64
bzip2-libs-1.0.6-13.el7.x86_64
[student@CentOS7 ~]$ rpm -qil | less
[student@CentOS7 ~]$ rpm -qil bzip2 | less
[student@CentOS7 ~]$ $(rpm -ql bzip2
student@CentOS7:~
To release input, press Ctrl+Alt
[coop@x7:~]
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
CentOS7 - VMware Workstation ...
6:31 AM

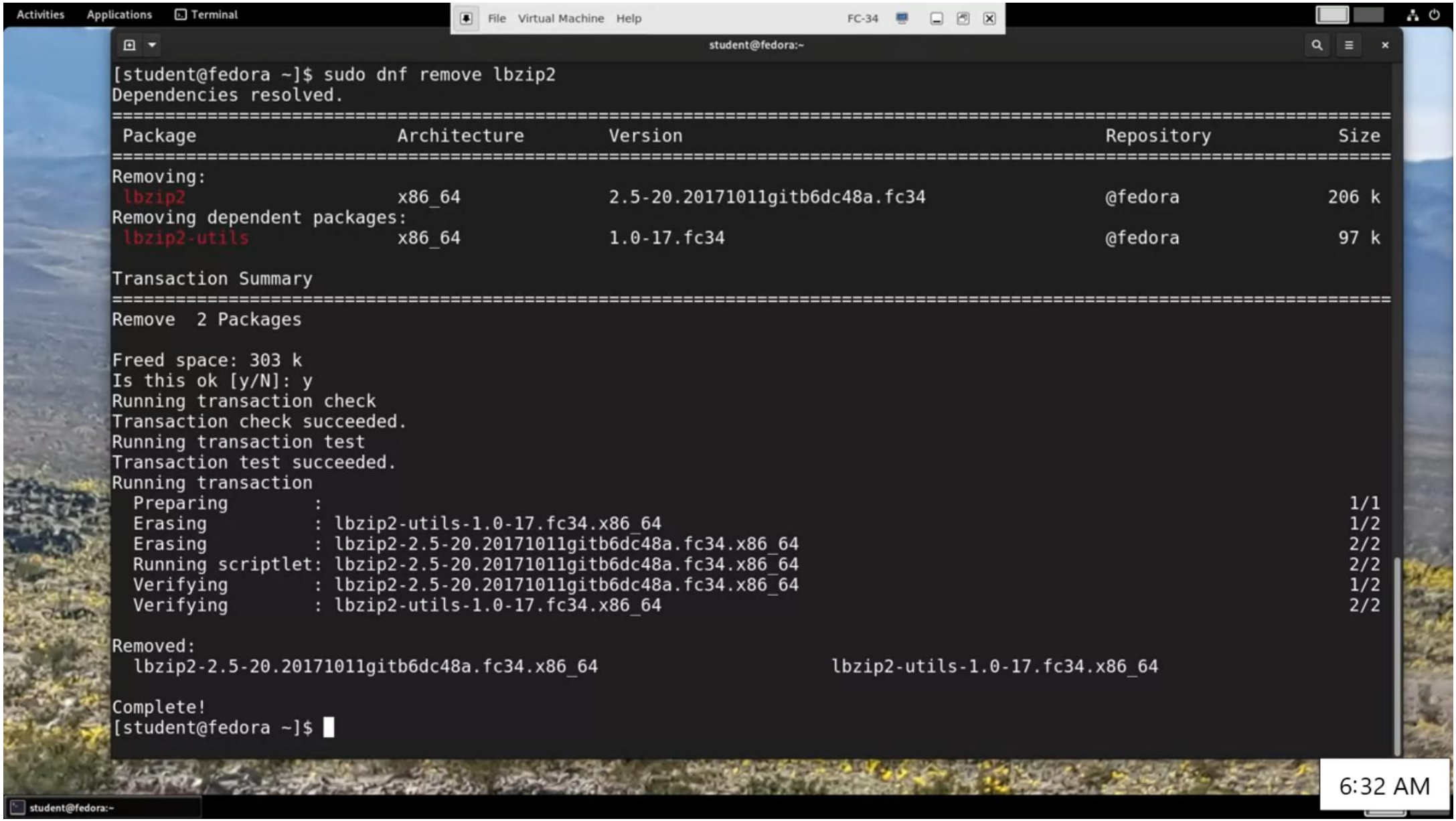

Activities
Applica...
VMw...
Fri 23 Jun, 9:54 AM
CentOS7 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
Places
Terminal
THE LINUX FOUNDATION TRAINING
home
Trash
student@CentOS7:~
File Edit View Search Terminal Help
[student@CentOS7 ~]$ rpm -qa | grep bzip2
bzip2-1.0.6-13.el7.x86_64
bzip2-libs-1.0.6-13.el7.x86_64
[student@CentOS7 ~]$ rpm -qil | less
[student@CentOS7 ~]$ rpm -qil bzip2 | less
[student@CentOS7 ~]$ ls -lF $(rpm -ql bzip2) | less
[student@CentOS7 ~]$ sudo rpm -e --test
student@CentOS7:~
To release input, press Ctrl+Alt
[coop@x7:~]
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
CentOS7 - VMware Workstation ...
6:31 AM


Activities
Applica...
VMw...
Fri 23 Jun, 9:55 AM
CentOS7 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Applications
Places
Terminal
THE LINUX FOUNDATION TRAINING
home
Trash
student@CentOS7:~
File Edit View Search Terminal Help
bzip2-1.0.6-13.el7.x86_64
bzip2-libs-1.0.6-13.el7.x86_64
[student@CentOS7 ~]$ rpm -qil | less
[student@CentOS7 ~]$ rpm -qil bzip2 | less
[student@CentOS7 ~]$ ls -lF $(rpm -ql bzip2) | less
[student@CentOS7 ~]$ sudo rpm -e --test bzip2
error: Failed dependencies:
bzip2 is needed by (installed) sos-3.3-5.el7.centos.noarch
bzip2 is needed by (installed) rpm-build-4.11.3-21.el7.x86_64
bzip2 is needed by (installed) libvirt-daemon-driver-qemu-2.0.0-10.el7_3
.5.x86_64
/usr/bin/bzip2 is needed by (installed) file-roller-3.14.2-10.el7.x86_64
[student@CentOS7 ~]$ rpm -q --whatprovides bzip2
bzip2-1.0.6-13.el7.x86_64
[student@CentOS7 ~]$ rpm -q --whatrequires bzip2
sos-3.3-5.el7.centos.noarch
rpm-build-4.11.3-21.el7.x86_64
libvirt-daemon-driver-qemu-2.0.0-10.el7_3.5.x86_64
[student@CentOS7 ~]$
student@CentOS7:~
To release input, press Ctrl+Alt
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
CentOS7 - VMware Workstation ...
6:31 AM


Activit...
Applic...
Ter...
Thu 22 Jun, 1:45 PM
student@debian: ~
File Edit View Search Terminal Help
student@debian:~$ dpkg --list | grep bzip2
ri bzip2   1.0.6-8.1   amd64   high-q
orting file compressor - utilities
student@debian:~$ dpkg --listfiles bzip2 | less
student@debian:~$ sudo dpkg --remove
THE LINUX FOUNDATION TRAINING
SimpleScreenRecorder
student@debian: ~
Debian-9 - VMware Workstation ...
6:31 AM


Activities
Applications
Terminal
File Virtual Machine Help
FC-34
student@fedora:~
[student@fedora ~]$ sudo dnf list *bzip2*
Last metadata expiration check: 0:43:06 ago on Mon 14 Jun 2021 03:01:14 PM CDT.
Installed Packages
bzip2.x86_64 1.0.8-6.fc34 @fedora
bzip2-devel.x86_64 1.0.8-6.fc34 @fedora
bzip2-libs.x86_64 1.0.8-6.fc34 @fedora
perl-Compress-Raw-Bzip2.x86_64 2.101-3.fc34 @fedora
Available Packages
bzip2-devel.i686 1.0.8-6.fc34 fedora
bzip2-libs.i686 1.0.8-6.fc34 fedora
bzip2-static.i686 1.0.8-6.fc34 fedora
bzip2-static.x86_64 1.0.8-6.fc34 fedora
getdata-bzip2.x86_64 0.10.0-15.fc34 fedora
lbzip2.x86_64 2.5-20.20171011gitb6dc48a.fc34 fedora
lbzip2-utils.x86_64 1.0-17.fc34 fedora
libknet1-compress-bzip2-plugin.x86_64 1.20-2.fc34 fedora
mingw32-bzip2.noarch 1.0.8-4.fc34 fedora
mingw32-bzip2-static.noarch 1.0.8-4.fc34 fedora
mingw64-bzip2.noarch 1.0.8-4.fc34 fedora
mingw64-bzip2-static.noarch 1.0.8-4.fc34 fedora
pbzip2.x86_64 1.1.13-4.fc34 fedora
perl-Compress-Bzip2.x86_64 2.28-3.fc34 fedora
perl-Compress-Raw-Bzip2-tests.x86_64 2.101-3.fc34 fedora
rust-async-compression+bzip2-devel.noarch 0.3.8-1.fc34 updates
rust-buffered-reader+bzip2-devel.noarch 1.0.1-1.fc34 updates
rust-buffered-reader+compression-bzip2-devel.noarch 1.0.1-1.fc34 updates
rust-bzip2+default-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2+futures-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2+tokio-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2+tokio-io-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2-sys+default-devel.noarch 0.1.10-1.fc34 fedora
rust-bzip2-sys-devel.noarch 0.1.10-1.fc34 fedora
rust-sequoia-autocrypt+compression-bzip2-devel.noarch 0.23.1-1.fc34 updates
6:32 AM
student@fedora:~


Activities
Applications
Terminal
File Virtual Machine Help
FC-34
student@fedora:~ — sudo dnf install lbzip2-utils
rust-buffered-reader+bzip2-devel.noarch 1.0.1-1.fc34 updates
rust-buffered-reader+compression-bzip2-devel.noarch 1.0.1-1.fc34 updates
rust-bzip2+default-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2+futures-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2+tokio-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2+tokio-io-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2-devel.noarch 0.4.2-1.fc34 fedora
rust-bzip2-sys+default-devel.noarch 0.1.10-1.fc34 fedora
rust-bzip2-sys-devel.noarch 0.1.10-1.fc34 fedora
rust-sequoia-autocrypt+compression-bzip2-devel.noarch 0.23.1-1.fc34 updates
rust-sequoia-ipc+compression-bzip2-devel.noarch 0.25.0-2.fc34 updates
rust-sequoia-net+compression-bzip2-devel.noarch 0.23.0-1.fc34 updates
rust-sequoia-openpgp+bzip2-devel.noarch 1.1.0-2.fc34 updates
rust-sequoia-openpgp+compression-bzip2-devel.noarch 1.1.0-2.fc34 updates
rust-zip+bzip2-devel.noarch 0.5.12-1.fc34 updates
[student@fedora ~]$
[student@fedora ~]$ sudo dnf install lbzip2-utils
Last metadata expiration check: 0:44:24 ago on Mon 14 Jun 2021 03:01:14 PM CDT.
Dependencies resolved.
Package Architecture Version Repository Size
Installing:
lbzip2-utils x86_64 1.0-17.fc34 fedora 44 k
Installing dependencies:
lbzip2 x86_64 2.5-20.20171011gitb6dc48a.fc34 fedora 96 k
Transaction Summary
Install 2 Packages
Total download size: 140 k
Installed size: 303 k
Is this ok [y/N]: y
6:32 AM
student@fedora:~ — sudo dnf ins...


Activities
Applications
Terminal
File Virtual Machine Help
FC-34
student@fedora:~
[student@fedora ~]$ sudo dnf remove lbzip2
Dependencies resolved.
Package Architecture Version Repository Size
Removing:
lbzip2 x86_64 2.5-20.20171011gitb6dc48a.fc34 @fedora 206 k
Removing dependent packages:
lbzip2-utils x86_64 1.0-17.fc34 @fedora 97 k
Transaction Summary
Remove 2 Packages
Freed space: 303 k
Is this ok [y/N]: y
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : 1/1
Erasing : lbzip2-utils-1.0-17.fc34.x86_64 1/2
Erasing : lbzip2-2.5-20.20171011gitb6dc48a.fc34.x86_64 2/2
Running scriptlet: lbzip2-2.5-20.20171011gitb6dc48a.fc34.x86_64 2/2
Verifying : lbzip2-2.5-20.20171011gitb6dc48a.fc34.x86_64 1/2
Verifying : lbzip2-utils-1.0-17.fc34.x86_64 2/2
Removed:
lbzip2-2.5-20.20171011gitb6dc48a.fc34.x86_64 lbzip2-utils-1.0-17.fc34.x86_64
Complete!
[student@fedora ~]$
6:32 AM
student@fedora:~

Linux Users

- All Linux users are assigned a unique user ID, which is just an integer, as well as one or more group IDs (one of which is the default one and is the same as the user ID)
- The normal prescription is that normal users start with a user ID of 1000 and then go up from there
- These numbers are associated with more convenient strings, or names, through the files **`/etc/passwd`** and **`/etc/group`**
  - For example, the first file may contain:
    `george:x:1000:1000:George Metesky:/home/george:/bin/bash`
  - And the second one:
    `george:x:1000:`

/etc/passwd

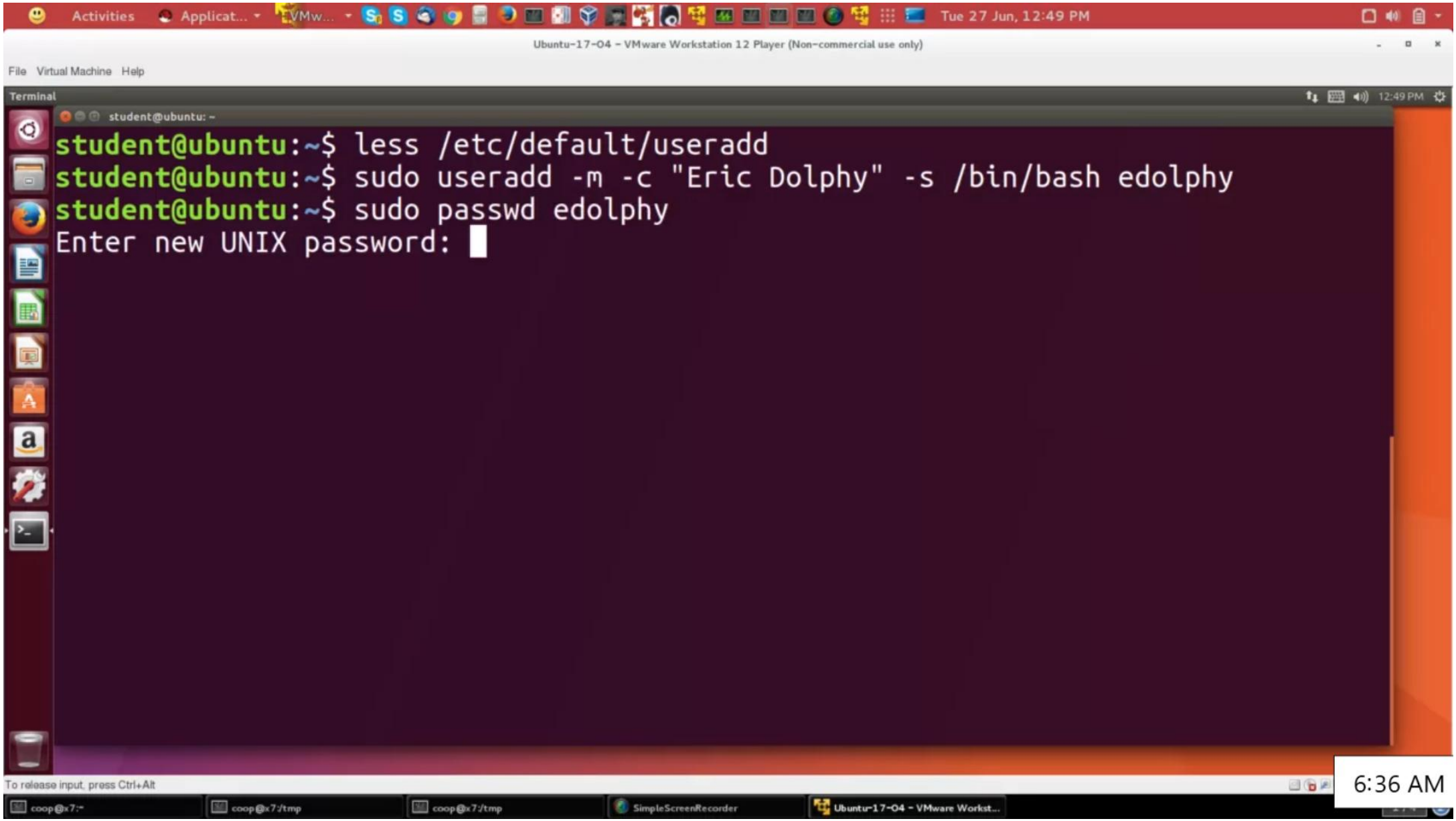
- **`/etc/passwd`** (**`george:x:1000:1000:George Metesky:/home/george:/bin/bash`**) contains some important pieces of information, separated by colons:
  - Account name: (**`george`**)
    The name of the user on the system, which should not contain capital letters
  - Password: (**`x`**)
    This can be an encrypted password, an asterisk, or an x depending on how security is set up on the system
  - User ID: (**`1000`**)
    The numerical value for the User ID
  - Group ID: (**`1000`**)
    The numerical value for the primary Group ID

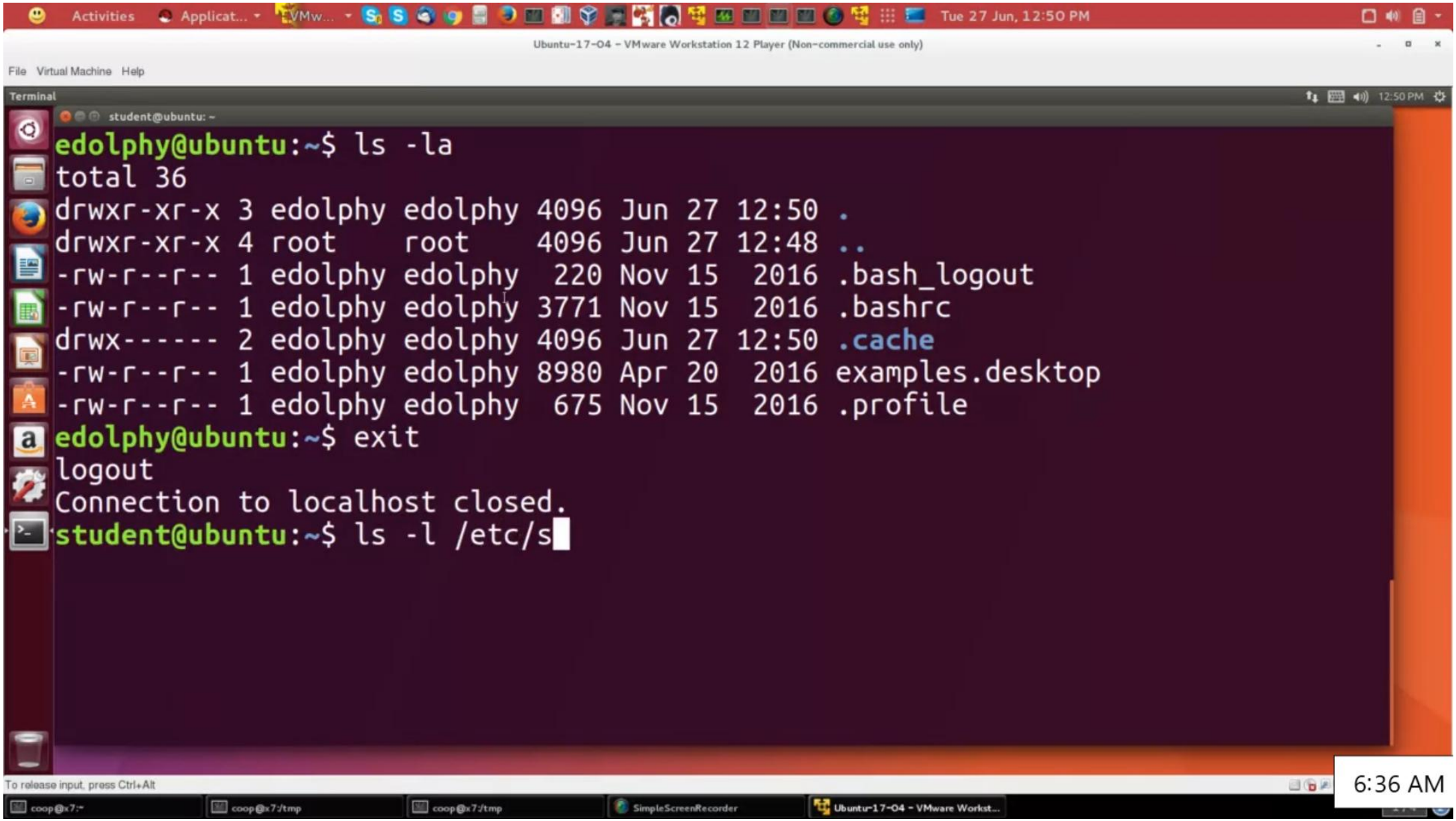
/etc/passwd (Cont.)

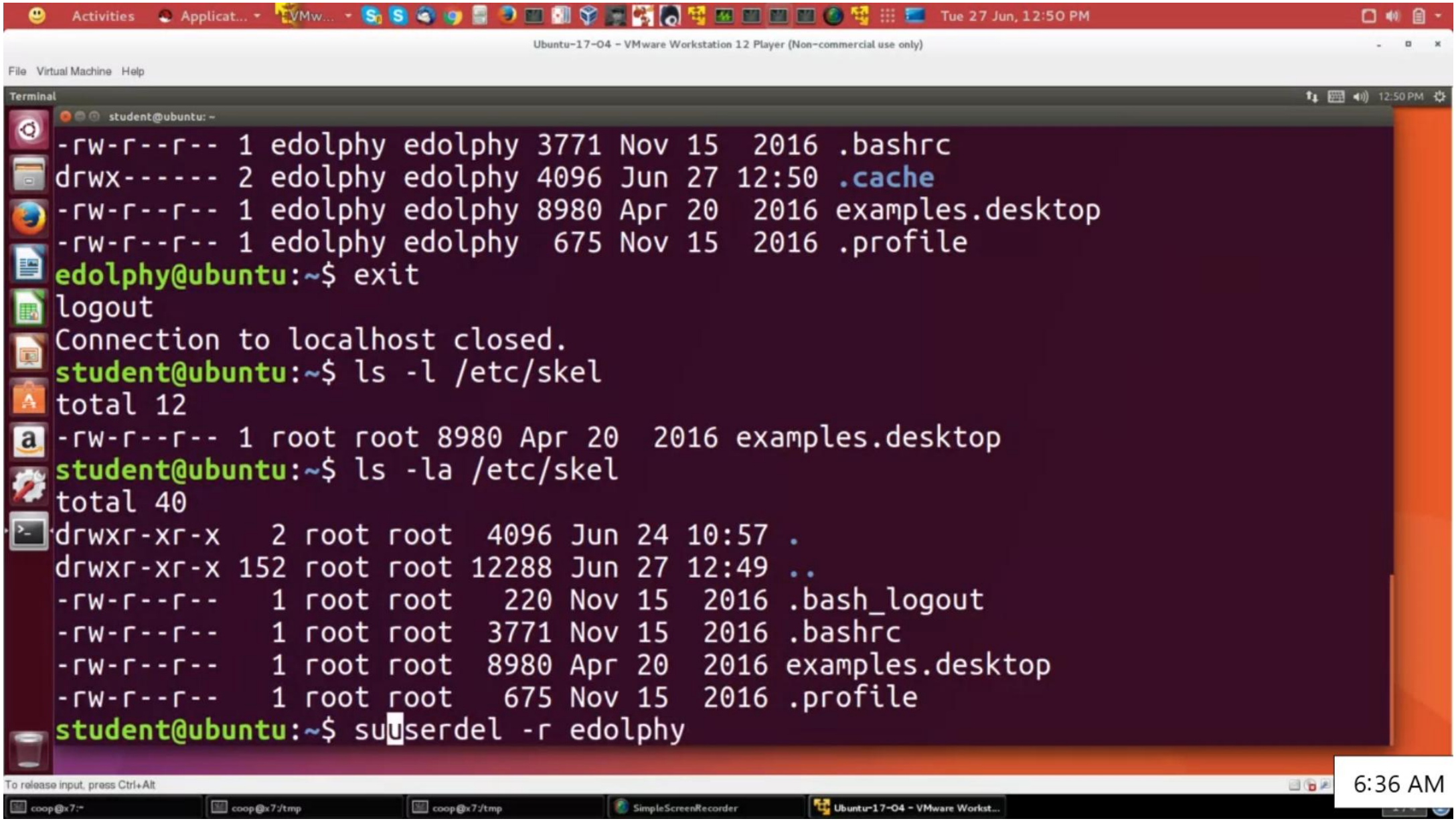
- **`/etc/passwd`** (**`george:x:1000:1000:George Metesky:/home/george:/bin/bash`**) contains some important pieces of information, separated by colons:
  - Full user name: (**`George Metesky`**)
    This field can be used for other purposes, but is almost always the full user name
  - Directory: (**`/home/george`**)
    The user's home directory
  - Shell: (**`/bin/bash`**)
    The user's default shell, which is the program run when logging in; if you see a **`/sbin/nologin`** or any non-executable program, this means the user cannot directly login (this is used for a lot of system daemons)
- If you look at **`/etc/passwd`**, you will see that almost all entries do not correspond to real users in the normal sense, but are special entities used for certain system utilities and functions

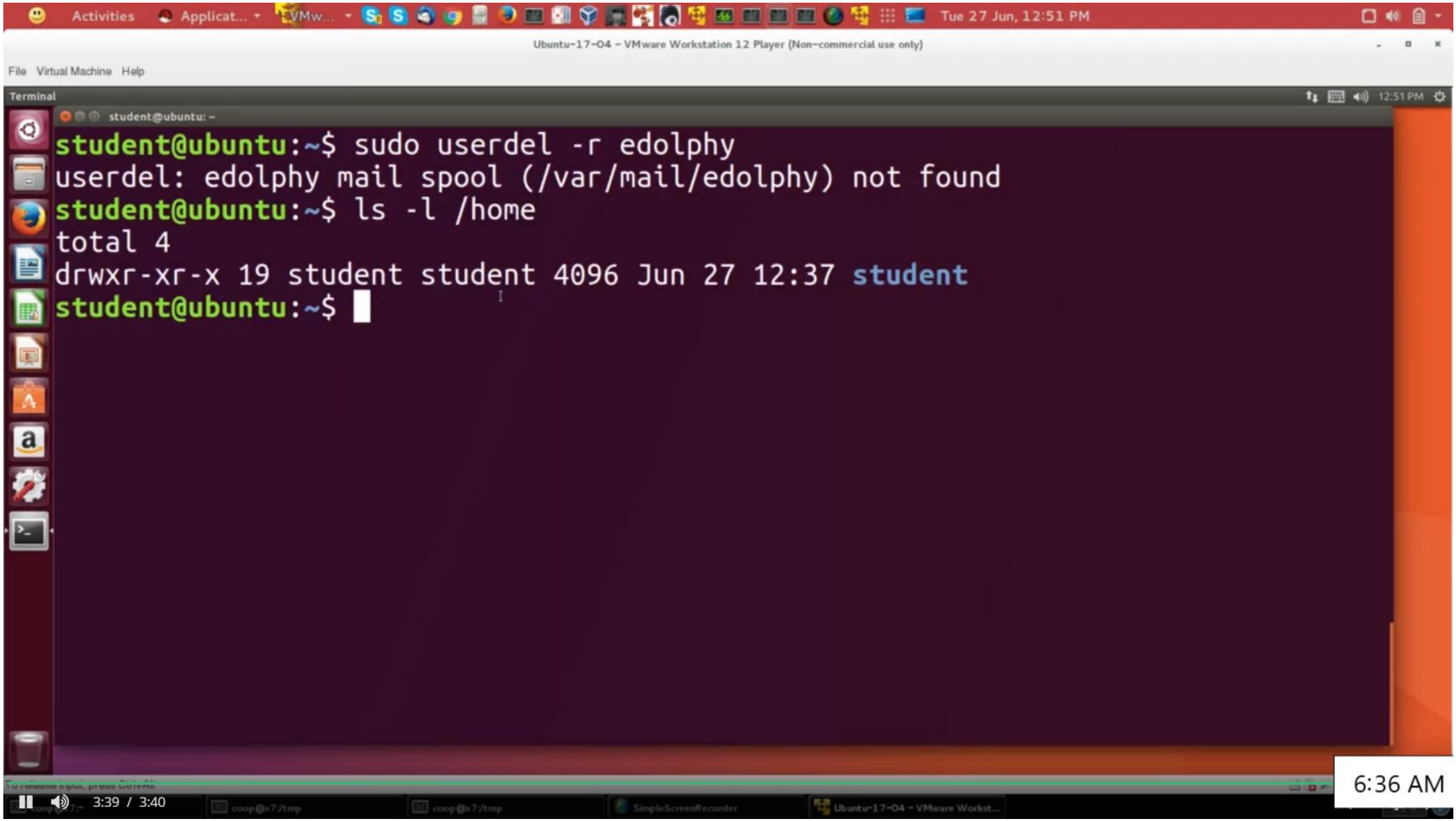
/etc/group

- **`/etc/group`** (**`fuse:x:106:root,george`**) is also straightforward and says that the **`fuse`** group, with numerical Group ID **`106`**, has as members **`root`** and **`george`**
- Groups are used to establish a set of users who have common interests for the purposes of access rights, privileges, and security considerations
- Access rights to files (and devices) are granted on the basis of the user and the group it belongs to


Activities
Applicat...
VMw...
Tue 27 Jun, 12:49 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal
12:49 PM
student@ubuntu: ~
student@ubuntu:~$ less /etc/default/useradd
student@ubuntu:~$ sudo useradd -m -c "Eric Dolphy" -s /bin/bash edolphy
student@ubuntu:~$ sudo passwd edolphy
Enter new UNIX password:
To release input, press Ctrl+Alt
coop@x7:~
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
Ubuntu-17-04 - VMware Workst...
6:36 AM


Activities
Applicat...
VMw...
Tue 27 Jun, 12:50 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal
12:50 PM
student@ubuntu: ~
edolphy@ubuntu:~$ ls -la
total 36
drwxr-xr-x 3 edolphy edolphy 4096 Jun 27 12:50 .
drwxr-xr-x 4 root root 4096 Jun 27 12:48 ..
-rw-r--r-- 1 edolphy edolphy 220 Nov 15 2016 .bash_logout
-rw-r--r-- 1 edolphy edolphy 3771 Nov 15 2016 .bashrc
drwx------ 2 edolphy edolphy 4096 Jun 27 12:50 .cache
-rw-r--r-- 1 edolphy edolphy 8980 Apr 20 2016 examples.desktop
-rw-r--r-- 1 edolphy edolphy 675 Nov 15 2016 .profile
edolphy@ubuntu:~$ exit
logout
Connection to localhost closed.
student@ubuntu:~$ ls -l /etc/s
To release input, press Ctrl+Alt
coop@x7:~
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
Ubuntu-17-04 - VMware Workst...
6:36 AM


Activities
Applicat...
VMw...
Tue 27 Jun, 12:50 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal
12:50 PM
student@ubuntu: ~
-rw-r--r-- 1 edolphy edolphy 3771 Nov 15 2016 .bashrc
drwx------ 2 edolphy edolphy 4096 Jun 27 12:50 .cache
-rw-r--r-- 1 edolphy edolphy 8980 Apr 20 2016 examples.desktop
-rw-r--r-- 1 edolphy edolphy 675 Nov 15 2016 .profile
edolphy@ubuntu:~$ exit
logout
Connection to localhost closed.
student@ubuntu:~$ ls -l /etc/skel
total 12
-rw-r--r-- 1 root root 8980 Apr 20 2016 examples.desktop
student@ubuntu:~$ ls -la /etc/skel
total 40
drwxr-xr-x 2 root root 4096 Jun 24 10:57 .
drwxr-xr-x 152 root root 12288 Jun 27 12:49 ..
-rw-r--r-- 1 root root 220 Nov 15 2016 .bash_logout
-rw-r--r-- 1 root root 3771 Nov 15 2016 .bashrc
-rw-r--r-- 1 root root 8980 Apr 20 2016 examples.desktop
-rw-r--r-- 1 root root 675 Nov 15 2016 .profile
student@ubuntu:~$ suuserdel -r edolphy
To release input, press Ctrl+Alt
coop@x7:~
coop@x7:/tmp
coop@x7:/tmp
SimpleScreenRecorder
Ubuntu-17-04 - VMware Workst...
6:36 AM


Activities
Applicat...
VMw...
Tue 27 Jun, 12:51 PM
Ubuntu-17-04 - VMware Workstation 12 Player (Non-commercial use only)
File Virtual Machine Help
Terminal
12:51 PM
student@ubuntu: ~
student@ubuntu:~$ sudo userdel -r edolphy
userdel: edolphy mail spool (/var/mail/edolphy) not found
student@ubuntu:~$ ls -l /home
total 4
drwxr-xr-x 19 student student 4096 Jun 27 12:37 student
student@ubuntu:~$
3:39 / 3:40
6:36 AM