



Overview

Introduction – Linux

- Command line basics – getting help
- The shell revisited: some features
- Navigating the file system
- File manipulation
- Text editing
- Various commands
- Archiving
- Groups, users, security
- Process control

1

Introduction

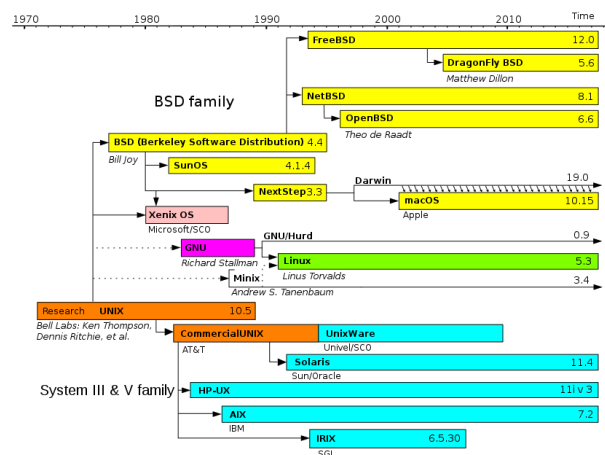
2

Operating system

- An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs.
(https://en.wikipedia.org/wiki/Operating_system)
 - Windows
 - Desktop systems
 - <https://gs.statcounter.com/os-market-share/desktop/worldwide/>
 - Linux
 - Server and supercomputing
 - MacOS
- Linux and Windows are based on foundations developed in the mid-1970s
- DOS, macOS and UNIX are proprietary, i.e., the source code of their kernel is protected

3

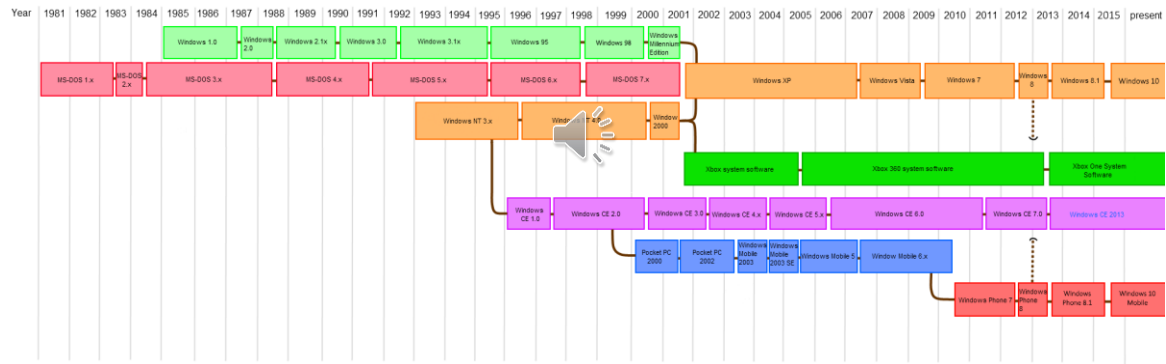
*nix time line



https://upload.wikimedia.org/wikipedia/commons/thumb/c/cd/Unix_timeline.en.svg/850px-Unix_timeline.en.svg.png

4

Windows time line



https://en.wikipedia.org/wiki/List_of_Microsoft_operating_systems#/media/File:Microsoft_timeline_of_operating_systems_2.png

5

Some history

- UNIX: roots in Bell Labs (AT&T)
- 1985 Free Software Foundation (FSF) founded by Richard Stallman. Along with other programmers creates the tools needed to make a UNIX compatible OS
- 1985 Professor Andy Tanenbaum creates a UNIX like operating system based on System V Unix for the IBM PC & PC/AT computers. It is called Minix.
- 1989 Richard Stallman releases GPL and GNU software but lacks a free kernel.
- **1991** Building on the concepts in Minix, **Linus Torvalds** (Finnish college student) develops Linux along with help from other users on the web.



6

Some history

GNU project:

- Established in 1984 by **Richard Stallman** (goal: software should be free from restrictions against copying or modification in order to make better and efficient computer programs),
- GNU is a recursive acronym for “**GNU's Not Unix**”,

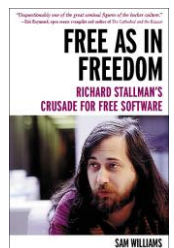


- Aim at developing a complete Unix-like operating system which is free for copying and modification,
- Companies make their money by maintaining and distributing the software, e.g. optimally packaging the software with different tools,
- Stallman built the first free GNU C Compiler in **1991**. But still, an OS was yet to be developed

7

Open Source Software

- **Open Source Software (OSS)** generally refers to software for which the source code is available and which the licensing scheme permits the user to modify it and redistribute it in modified or unmodified form.
- **GNU copyleft** (www.gnu.org)
 - Nobody should be restricted by the software they use. There are four freedoms that every user should have:
 - the freedom to use the software for any purpose,
 - the freedom to change the software to suit your needs,
 - the freedom to share the software with your friends and neighbors, and
 - the freedom to share the changes you make.
 - When a program offers users all of these freedoms, we call it free software.



8

Linux OS

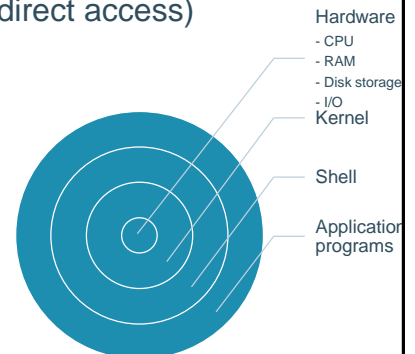
- Linux is not a single operating system, but rather a large family of free and open source operating systems based on the Linux kernel.
- Different variants within this family are referred to as **Linux Distributions** (www.distrowatch.com) CentOS, Ubuntu, etc.
- There are two major components of Linux:
 - The **kernel** is the core of the Linux operating system:
 - Schedules processes and interfaces directly with the hardware.
 - It manages system and user I/O, processes, devices, files, and memory.
 - The **shell** is a text-only interface to the kernel.
 - Users input commands through the shell, and the kernel receives the tasks from the shell and performs them.
 - The shell works interactively (REPL).
 - Users *do not* interact with the kernel directly, it is done through the shell or a desktop environment.

(<https://cvw.cac.cornell.edu/Linux/introduction>)

9

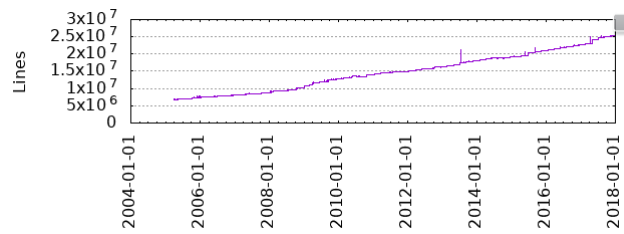
Linux kernel

- aka: executive, system monitor.
- Provides a layer between the computer hardware and user applications.
- Provides an interface for software to use hardware (no direct access)
- Planning and assigning:
 - memory, CPU, disk, etc.
 - security aspects
 - Fulfill user requests (system calls).
 - Filesystem, networking, ...



10

Linux kernel



- Lines of code

(https://www.phoronix.com/scan.php?page=news_item&px=Linux-Kernel-Commits-2017)

- Linux in 2020: 27.8 million lines of code in the kernel

https://www.theregister.co.uk/2020/01/06/linux_2020_kernel_systemd_code/

index : kernel/git/stable/linux.git

Linux kernel stable tree

mary	refs	log	tree	commit	diff	stats	log msg
<div>Commit message (Expand)</div> <div>Linux 5.5-rc5 HEAD v5.5-rc5 master</div> <div>Merge tag 'riscv-for-v5.5-rc5' of git://git.kernel.org/pub/scm/linux/kernel...</div> <div>Documentation: riscv: add patch acceptance guidelines</div> <div>riscv: prefix IRQ_ macro names with an RV_ namespace</div> <div>clocksource: riscv: add notrace to riscv_sched_clock</div> <div>Merge branch 'akpm' (patches from Andrew)</div> <div>Merge tag 'apparmor-pr-2020-01-04' of git://git.kernel.org/pub/scm/linux/kern...</div> <div>apparmor: fix aa_xattrs_match() may sleep while holding a RCU lock</div> <div>Merge tag 'mips_fixes_5.5_1' of git://git.kernel.org/pub/scm/linux/kernelgit...</div> <div>hexagon: define loremap_uc</div> <div>ocfs2: fix the crash due to call ocfs2_get_dlm_debug once less</div> <div>ocfs2: call journal flush to mark journal as empty after journal recovery whe...</div> <div>mm/hugetlb: defer freeing of huge pages if in non-task context</div>							<div>Author</div> <div>Linux Torvalds</div> <div>Linux Torvalds</div> <div>Paul Walmsley</div> <div>Paul Walmsley</div> <div>Zong Li</div> <div>Linux Torvalds</div> <div>Linux Torvalds</div> <div>John Johansen</div> <div>Linux Torvalds</div> <div>Nick Desaulniers</div> <div>Gang He</div> <div>Kai Li</div> <div>Waiman Long</div>

11

Linux kernel

- File Management
 - Controls the creation, removal of files and provide directory maintenance
 - For a multiuser system, every user should have its own right to access files and directories
- Process Management
 - Multitask system: multiple programs can be executed simultaneously
 - When a program starts to execute, it becomes a *process*. The same program executing at 2 different times will become 2 different processes
 - Processes are managed by the kernel: creating, suspending, terminating
 - A process is protected from other processes and can communicate with the others

13



Linux kernel

- Memory management
 - Memory in a computer is divided into main memory (RAM) and secondary storage (usually refer to hard disk)
 - Memory is small in capacity but fast in speed, and hard disk is vice versa
 - Data that are not currently used should be saved to hard disk first, while data that are urgently needed should be retrieved and stored in RAM
- Device drivers
 - Interfaces between the kernel and the BIOS (*basic input/output system*)
 - Different device has different driver

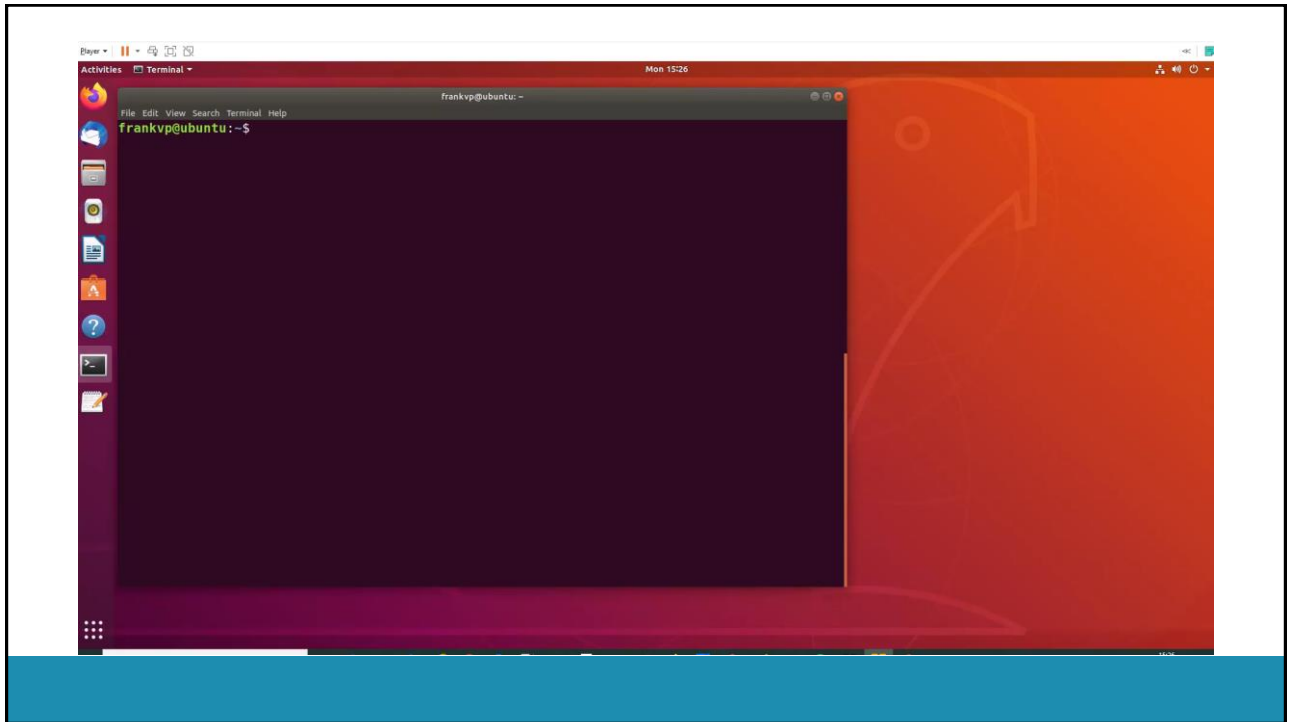
14



Linux kernel

- Get the kernel version
`uname -r`
- `uname -a` provides the information in the order
 - KERNEL-NAME NODENAME KERNEL-RELEASE KERNEL-VERSION
 - MACHINE PROCESSOR HARDWARE-PLATFORM OPERATING-SYSTEM

15



16

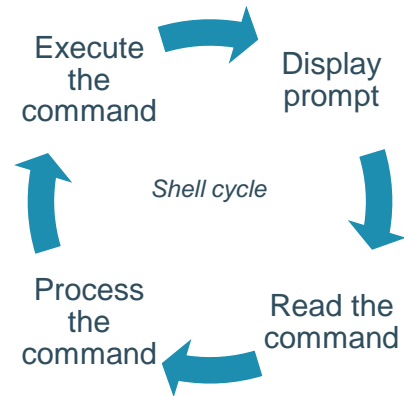
Linux User Interface

- Traditional *nux (Linux, Unix) systems use **command line interface (cli)** or text-based interface)
- User needs to type lines of command to instruct the computer to work, similar to Windows DOS command window
- Advantages:
 - Fast: few resources required
 - Reliable: some of these commands have been around for years. Commands change less frequently than the GUI counterparts. GUI counterparts often use these same commands under the hood.
 - Productivity: use the keyboard instead of navigating through a bunch of menus and screens with the mouse and tabbing back and forth between different applications
- Disadvantages:
 - Memorize the commands
 - Typing a command can be error prone

18

Linux shell

- A shell is an interface for accessing the underlying commands of an operating system.
- When you sign in at the command line or launch a terminal window on Linux, the system launches the shell program.
- Is the command line interpreter: a program that accepts input from a user (e.g. a command) and performs the requested task. (REPL)
- The shell's **prompt** identifies the type of shell being used



19

Most popular shells

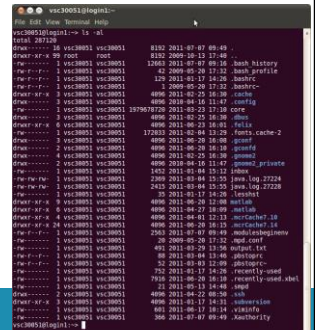
- There are several types of shells for Linux.
- Check it with
`$ echo $SHELL`
- Change the shell with
`chsh`

Shell	Prompt	Name	Note
sh	\$	Bourne Shell	Default on some Unix systems
bash	\$	Bourne Again Shell	Enhanced replacement for the Bourne shell Default on most Linux and Mac OS X systems
csh	%	C Shell	Default on many BSD systems
tcsh	>	TC Shell	Enhanced replacement for the C shell
ksh	\$	Korn Shell	Default on AIX systems

20

Command Line Interface: Beware!

- Common use on servers and HPC system
- Implicitly assumes that you know what you are doing. Don't be scared!
- Often there are no warnings with commands, no *undo*
- No recycle bin!
- If anything goes wrong, you can stop the command with **ctrl+C**



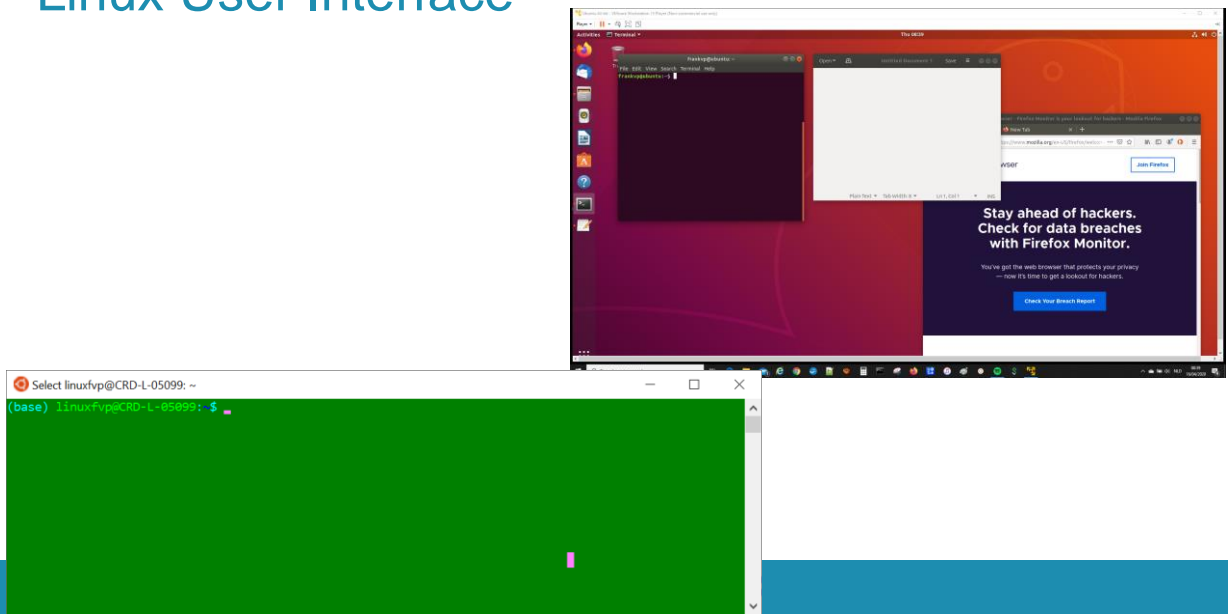
21

Linux User Interface

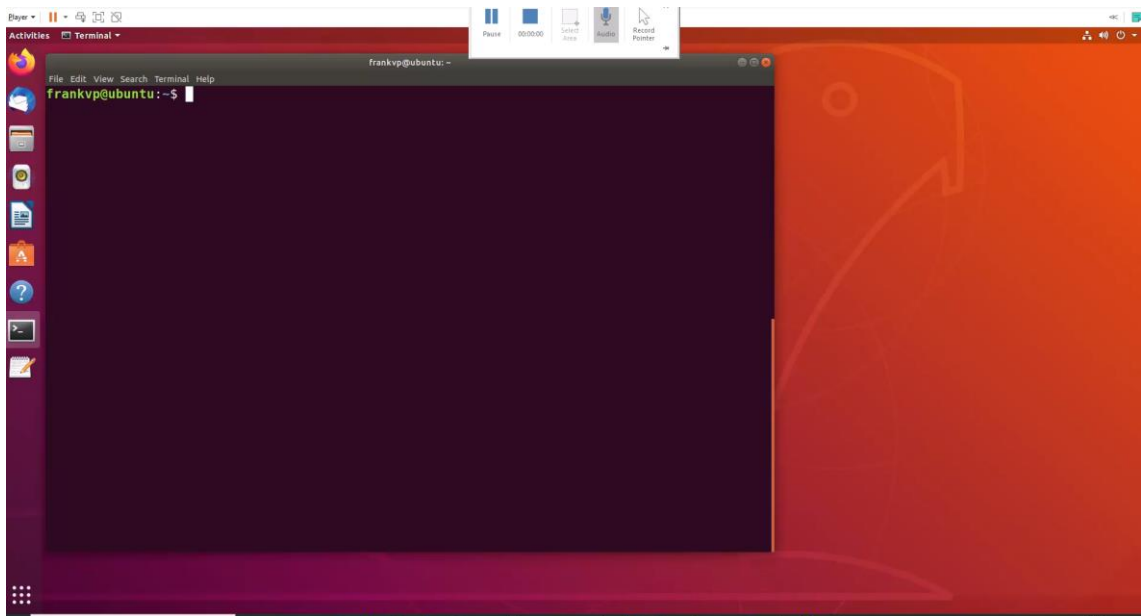
- By adopting the X-Window technology, graphical user interface (GUI) is available for Linux.
- Uses pointing devices (e.g. mouse) to control the system, similar to Microsoft's Windows
- Provide menu-driven and/or icon-driven interfaces
 - menu-driven: user is provided with a menu of choices. Each choice refers to a particular task
 - icon-driven: tasks are represented by pictures (icon) and shown to user. Click on an icon invokes one task
- Advantages:
 - No need to memorize commands. Always select task from menus or icons
- Disadvantages:
 - Slow and require certain resource for its implementation
 - Suitable for general users and systems, such as PC

22

Linux User Interface



23



24



Someone is doing something somewhere

- Linux is a multiuser OS
 - Allows multiple users to use the resource of a computer at the same time
 - Before you can do anything, you have to log in.
 - This can be either a graphical or a nongraphical prompt.
 - Linux desktop: you are likely to see a graphical environment.
 - A server: you'll just see a shell login prompt.
- Check who you are on the computer: `whoami`
- List the users logged on with: `who`
- What is the machine you are working on: `hostname`
- What is the directory you are working in: `pwd`
- What time is it: `date`

25



Useful commands

- Clear the contents of the current screen
`$ clear`
- `$ logout`
 - The `logout` command logs your account out of the system (in a text mode).
 - This will end your terminal session and return to the login screen.
 - Some systems may have a file called `.logout` or `.bash_logout` in each user's home directory.
- `$ exit`
 - Exit the current shell. The `exit` command is similar to the `logout` command with the exception that it does not run the `logout` script located in the user's home directory.

26

HPC



```

Mate Terminal
File Edit View Search Terminal Help
bash-4.2$ hostname
tier2-p-login-3
bash-4.2$ whoami
vsc30051
bash-4.2$ who
vsc33412 pts/1      Apr 16 08:00 (10.118.224.28)
vsc30760 pts/24      Apr 14 21:16 (:1003)
vsc33476 pts/31      Apr 16 08:03 (2a02:1811:d16:4c00:71b9:bb3c:311)
vsc33562 pts/4       Apr 10 21:05 (:1015)
vsc32242 pts/20      Apr 13 20:27 (:1032.0)
vsc33355 pts/0       Apr 15 01:36 (:1016.0)
vsc33313 pts/37      Apr 15 19:59 (10.118.224.28)
vsc30539 pts/39      Mar 31 12:39 (10.33.208.21)
vsc32366 pts/42      Apr 14 13:57 (10.118.224.28)
vsc32456 pts/44      Apr 15 15:57 (:1025)
vsc31953 pts/45      Apr 14 14:19 (10.118.230.3)
vsc32448 pts/19      Apr 15 10:17 (:1010)
vsc32366 pts/46      Apr 14 14:07 (10.118.224.28)
vsc33313 pts/48      Apr 15 20:00 (10.118.224.28)
vsc32448 pts/10      Apr 14 17:58 (:1010)
x0090231 pts/33      Apr 14 12:09 (:1047)
x0090231 pts/47      Apr 14 16:54 (:1047)
vsc33355 pts/86      Apr 15 17:37 (:1035.0)
vsc33397 pts/26      Apr 12 16:35 (:1013)
vsc32101 pts/30      Apr 14 15:00 (:1043)
vsc32448 pts/6       Apr 15 10:11 (:1010)
vsc31402 pts/5       Apr 8 08:45 (:1045)
vsc32319 pts/49      Apr 14 10:04 (:1027)
x0007234 pts/36      Apr 16 08:18 (:1006)

Mate Terminal
File Edit View Search Terminal Help
bash-4.2$ pwd
/vsc-hard-mounts/leuven-user/300/vsc30051
bash-4.2$ date
Thu Apr 16 08:33:04 CEST 2020
bash-4.2$ uname -a
Linux tier2-p-login-3 3.10.0-957.27.2.el7.x86_64 #1 SMP Mon Jul 29 17:46:05 UTC
2019 x86_64 x86_64 x86_64 GNU/Linux
bash-4.2$ uname -r
3.10.0-957.27.2.el7.x86_64
bash-4.2$ echo $SHELL
/bin/bash
bash-4.2$
```

27

Hands-on 1

28