

OPERATING SYSTEMS CCGC-5000

Modules 1, 2.1

Agenda

Authentic information is available from the given resources in course outline and URL's mentioned from this slides, and this presentation is only supportive document to be read with the given resources and corrected accordingly if required..

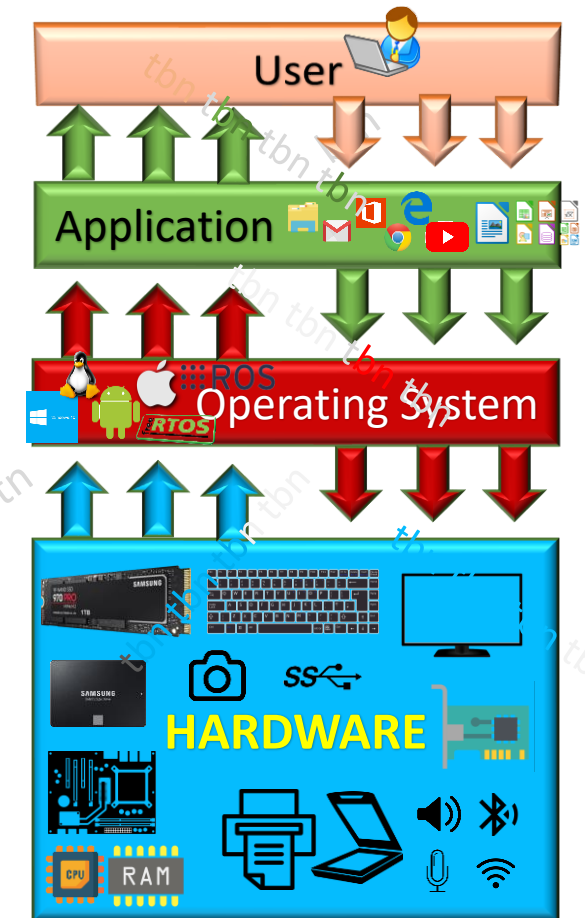
- Operating System
- History of Unix, Linux, Microsoft
- Unix vs Linux & Linux vs Windows
- Linux Installation,
- Linux GUI, CLI interface
- Virtual consoles, system restart/shutdown
- Navigation in Linux, command sequence
- Linux CLI, basic commands, System information
- Kernel, Shell



Must read

- Chapters 1,2 of RHEL8, 2nd Edition book
- RedHat documentation
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/

Operating Systems (OS)

- OS
 - is a program or system software that efficiently manages hardware acting as intermediary between users and application programs
 - controls and coordinates the use of hardware among the various application programs for various users
 - is software that manages computer hardware resources and provides common services for computer programs
- Some OS Examples
 - Windows 10, Windows 2008, 2012, 2016,
 - CentOS, Ubuntu, Debian, RedHat, SuseLinux, Oracle Linux,
 - Android, IOS, Symbian, Chrome
 - Mac(Unix-like called Darwin),
 - QNX - RTOS (*realtime OS*), ROS (*robotic OS*)





 represents Bluetooth and wireless controllers, devices respectively.

Operating System









- An operating system consists of
 - **Kernel**
 - Kernel is the heart of an operating system built in with core components for the OS functionality.
 - It resides in the memory when the system starts up
 - It controls, manages the processes and process interactions that take place within an operating system
 - **Shell**
 - Interfaces the user with kernel using a control language acting as interpreter

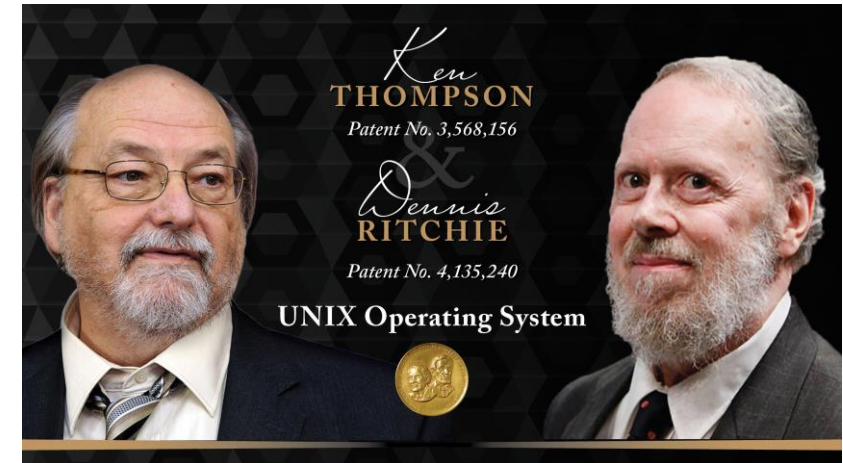
• Core functions of Operating System

- Process Management
- Memory Management
- Secondary Storage Management
- I/O System Management
 - Buffer caching system
 - Device drivers in general
 - Specific HW device drivers
- File Management
- Protection System (Security)
- Network Management
- User Interface
 - Command Line Interface (CLI)
 - Graphical User Interface (GUI)

UNIX Development



- 1970 - UNIX conceived and implemented by **Ken Thompson** and **Dennis Ritchie** of **Bell Labs**, to be used in Bell Labs (Now Nokia Bell Labs) https://en.wikipedia.org/wiki/Bell_Labs
- The Unix operating system was first presented formally to the outside world at the 1973 Symposium on Operating Systems Principles, where Ritchie and Thompson delivered a paper
- In 1973, AT&T released Version 5 Unix and licensed it to educational institutions, and licensed 1975's Version 6 to companies for the first time.
- AT&T licensed Unix to outside parties from the late 1970s, leading to a variety of both academic and commercial variants of Unix from vendors such as
 - the University of California, Berkeley (BSD), 
 - Microsoft (Xenix), 
 - IBM (AIX) 
- In 1987, AT&T and Sun Microsystems collaborated to release UNIX System V Release 4 (SVR4)   
- In 1991, Sun Microsystems introduced Solaris 2, internally known as SunOS 5  
- Later in 2010 Oracle acquired Sun Microsystems, and it become Oracle Solaris.
- Apple Mac also used/uses variants of Unix A/ROSE, A/UX, MAE, Darwin and Linux variant MkLinux
- Two major versions of UNIX are **System V** which is popular and **BSD** (Berkley Software Distribution) version released from University of California in Berkley.
- Portable Operating System Interface – POSIX set standard for UNIX operating systems.



<https://www.invent.org/blog/inventors/new-inductee-class-Thompson-Ritchie>



<https://medium.com/@joseph.crick/lessons-from-the-development-of-unix-8a324caaf1b4>

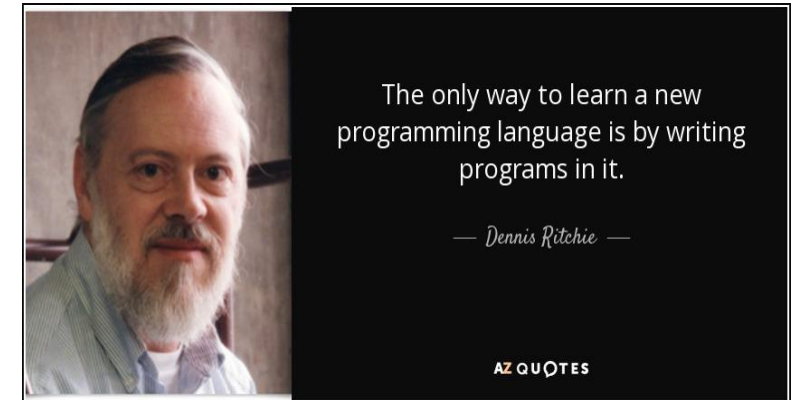
https://en.wikipedia.org/wiki/History_of_Unix

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

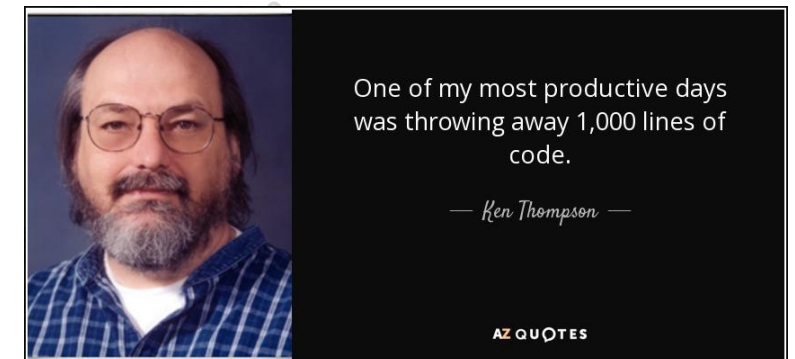
UNIX Development



In 1970s, Ken Thompson (sitting) and Dennis Ritchie working together at a PDP-11, https://en.wikipedia.org/wiki/History_of_Unix



<https://www.azquotes.com/quote/721047>



https://www.azquotes.com/author/14620-Ken_Thompson

View the video of Ken Thompson and Dennis Ritchie
<https://www.youtube.com/watch?v=JoVQTPbD6UY>

Evolution of Linux

- 1983, Richard Stallman started GNU (*GNU's not UNIX*) project to create a free UNIX-like operating system.
- However, GNU kernel Hurd failed to attract enough development effort, leaving GNU incomplete.
- In 1987, MINIX, a Unix-like system intended for academic use, was released by Andrew S. Tanenbaum to exemplify the principles conveyed in his textbook, *Operating Systems: Design and Implementation*.
- While source code for the system was available, modification and redistribution were restricted
- 1991, **Linus Torvalds** began a project while studying at University of Helsinki, which later became Linux kernel
- Development was done on MINIX using the GNU C Compiler
- In 1992, he suggested releasing the kernel under the GNU General Public License
- In 1996, Linus Torvalds announced mascot for Linux, the penguin which was called as TUX, for Torvald's UniX
- Linux version **SystemV** moved over to **systemd** version, Ubuntu moved to systemd in 15.04 version and RedHat Linux from version 7.
- Recommended video to watch : <https://www.youtube.com/watch?v=o8NPllzkFhE>

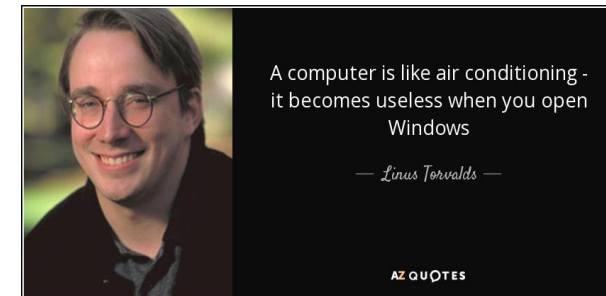


Tux



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

Linus Torvalds

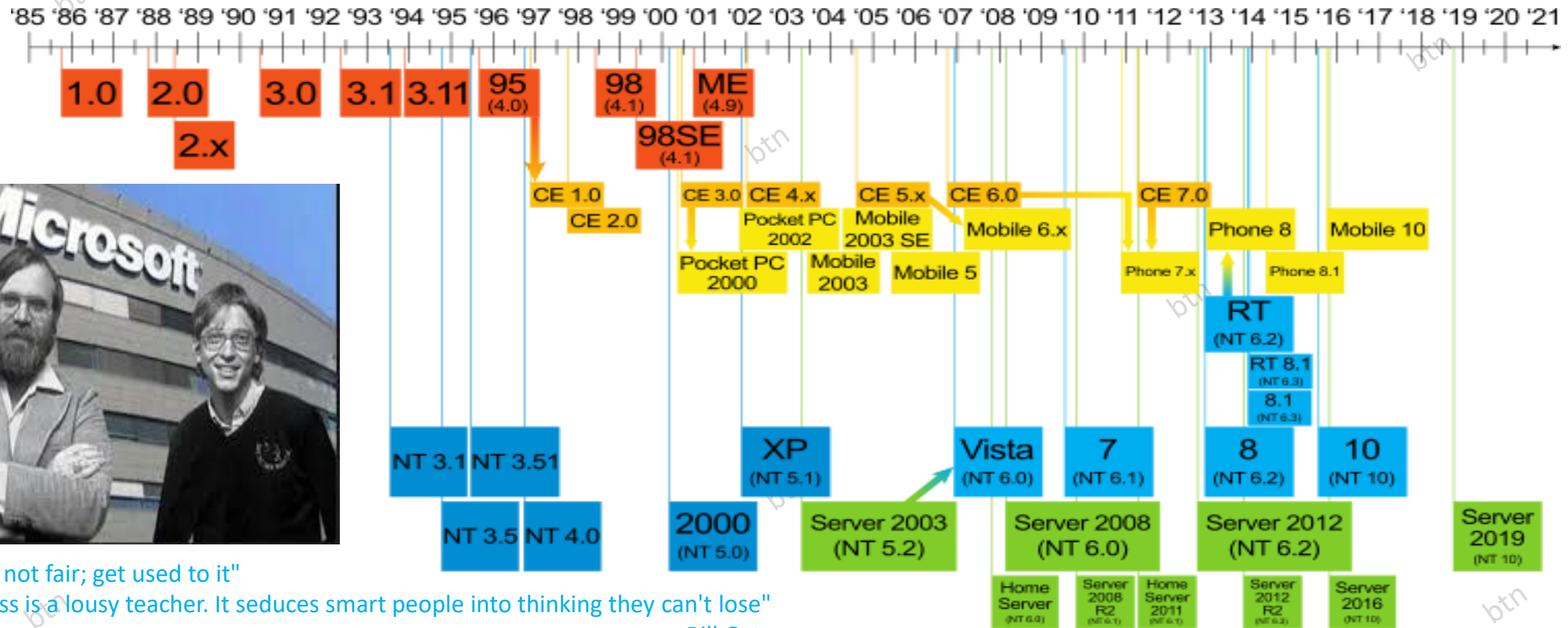


<https://www.azquotes.com/quote/755276>

@ 17:08 "I'm perfectly happy with all the people who are walking around and just staring at the clouds and looking at the stars and saying, "I want to go there," but I'm looking at the ground and I want to fix the pothole that's right in front of me before I fall in. This is the kind of person I am" - Linus Torvalds

History of Microsoft OS

- Two friends, Bill Gates and Paul G. Allen from Seattle, in 1975 founded Microsoft and released the first Windows version on Nov 20, 1985. https://en.wikipedia.org/wiki/Timeline_of_Microsoft_Windows



"Life is not fair; get used to it"

"Success is a lousy teacher. It seduces smart people into thinking they can't lose"

-- Bill Gates

"Any crusade requires optimism and the ambition to aim high"

"Technology is notorious for engrossing people so much that they don't always focus on balance and enjoy life at the same time."

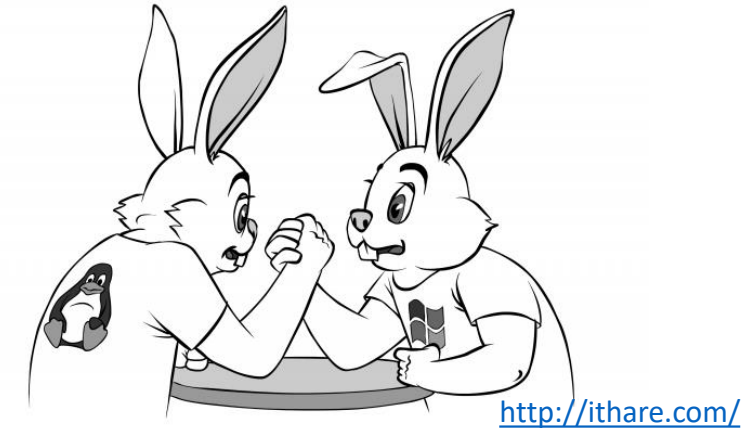
--Paul Allen

Linux vs UNIX

- Unix was proprietary but Linux was free under GNU GPL (General Public License)
- There is also open BSD, free BSD, open Solaris, open SCO Unix available
- Linux is same as UNIX but does not use UNIX codes
- Structure of Linux and UNIX are similar
- Naming of disks are different in Linux and UNIX
- Many commands are different in Linux and UNIX for the similar purpose.
- Both are multi-user, multi-tasking systems
- Linux is widely used as compared to UNIX
- Linux low deployment cost
- Less Hardware resources as compared to UNIX
- Linux is supported by Big players like Oracle, RedHat(IBM), etc.,

Linux Vs MS Windows

- License
 - MS Windows is proprietary software whereas Linux is GPL (General Public License)
- Price
 - Linux is free as compared to costed MS Windows OS licenses.
- Reliability
 - Reliable than Windows, Linux runs for months without the need to be rebooted
- User friendly
 - Windows is user friendly compared to Linux, but linux GUI has improved consistently and edges out Windows.
- Software
 - Linux has variety of software as GPL as compared to priced Windows applications, example office products like word, excel, power point etc.,
- Hardware
 - Hardware manufacturers have made advancements to support linux but sometimes device drivers for linux may not be easily available as Windows
- Security
 - Linux is secure and gains edge over Windows security. Windows is more vulnerable to virus and malware attacks in comparison to Linux OS.
- Support
 - Linux support based on online community and Windows has its own help desk customer support added with online community that is less compared to Linux
- Source Code
 - Source code for Linux is available and hence customizable, but Windows source code is not available.



Linux Distributions

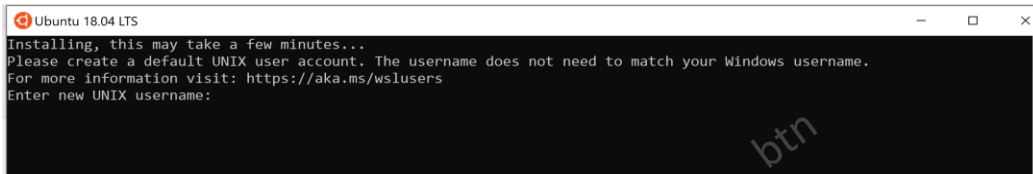
- Linux Distribution (also called Linux Distros) is Linux Operating system developed from software collection based on Linux Kernel and from a package management system.
- A typical Linux distribution comprises a Linux kernel, GNU tools and libraries, additional software, documentation, a window system (the most common being the X Window System), a window manager, and a desktop environment.
- **Our course will be using RPM based RedHat Enterprise Linux (RHEL) 8.3**

Distribution	Description	Examples
Debian	GNU/Linux is a distribution that emphasizes free software. It supports many hardware platforms. Debian and distributions based on it use the .deb package format and the dpkg package manager and its frontends	Debian, Ubuntu, Kubuntu, Lubuntu, Xubuntu, Kali Linux, Pure OS, Raspbian, etc.,
RPM base	Red Hat Linux and SUSE Linux were the original major distributions that used the RPM file format, which is today used in several package management systems	RedHat Enterprise Linux, Oracle Linux, CentOS (community supported),
Fedora Based	Fedora is a community supported distribution based on Red Hat	Fedora, EduLinux, Fuduntu
Gentoo	Gentoo is a distribution designed to have highly optimized and frequently updated software	Gentoo, Gentoox – an adaptation for Xbox, Chrome OS,
Pacman	Pacman is a package manager that is capable of resolving dependencies and automatically downloading and installing all necessary packages. In theory, a user need only run a single command to completely update the system	Arch Linux, ArchBang, Managro, etc.,x`
Slackware	Slackware is known as a highly customizable distribution that stresses ease of maintenance and reliability over cutting-edge software and automated tools. Generally considered a distribution for advanced users, it is often suggested to those who want to learn the inner workings of a Linux operating system	Slackware, Sentry Firewall, Freenix, etc.,

For more detailed distribution information https://en.wikipedia.org/wiki/Linux_distribution & https://en.wikipedia.org/wiki/List_of_Linux_distributions

Linux Installation and WSL

- Ubuntu Terminal can be installed as **Windows subsystem** in windows using **Ubuntu 18.04 LTS** or **Ubuntu 20.04 LTS app** in Microsoft Store
- Ubuntu is Debian linux distribution.
- After installation, launch it., if already installed run Ubuntu from Windows Programs to get Ubuntu Terminal. *(you may need to enable WSL in windows from **Turn Windows features on or off** - refer <https://wiki.ubuntu.com/WSL>)*
- Need to give a username and password and can start using Ubuntu linux



```
Ubuntu 18.04 LTS
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username:
```

information on Windows Subsystem for Linux(WSL)

<https://ubuntu.com/wsl>

• Physical Installation

- Use the boot CD/DVD or USB from the iso file downloaded, to start the physical system and follow the steps.
- Refer Exercise 1-2, Linux course book.

• Virtual Installation

- Install VMWare Player or Oracle Virtual Box or VMWare Workstation Pro (if license is available)
- Create new VM, using the iso file and follow the steps from Exercise 1-2, Linux course book.

• Cloud

- You can launch VM by choosing an instance type

• Installation could be a GUI or CLI interface

• In RHEL, minimal install ends in CLI interface.

• Register with <https://developers.redhat.com/> from

• RHEL 8.3 can be downloaded from https://developers.redhat.com/download-manager/file/rhel-8.3-x86_64-dvd.iso

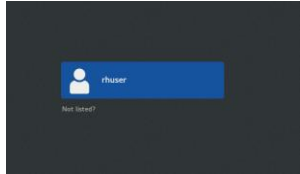
RHEL Installation

- Installation requires information on language, timezone, keyboard, storage, network, username, hostname, domain name, softwares to be installed, etc.,
- Also need to test the instance for compatibility before going for live or real time.
- RHEL installer program is called **Anaconda**
- Installation log files are stored /root, /var/log/, /tmp.
- **Refer Table1-1 Installation Logs of course book**
- The Virtual consoles can be used to check on ongoing progress of installation
- Virtual console 1 is main installer console screen
- Virtual console 2 can be used as command prompt
- Virtual console 6 is the graphical screen
- Virtual screen 3,4,5 shows messages related to installation, storage and program respectively, otherwise bash command prompt is displayed.

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/performing_a_standard_rhel_installation/preparing-for-your-installation_installing-rhel

Linux GUI and CLI interface

- GUI interface

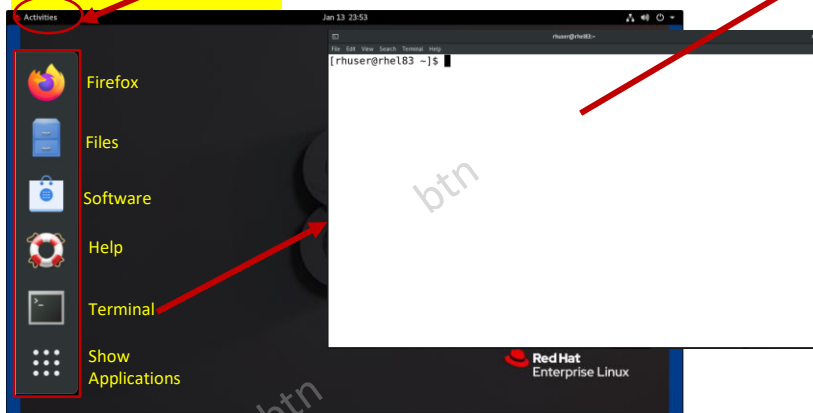


MINIMAL interface

```
Red Hat Enterprise Linux 8.3 (Ootpa)
Kernel 4.18.0-240.el8.x86_64 on an x86_64
minimal login: _
```

(minimal seen here is the hostname)

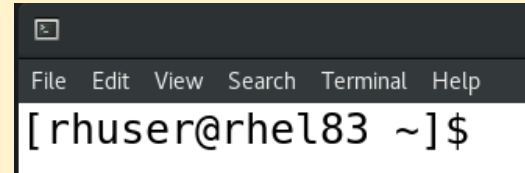
- Click on the user to login
- After login, click on **Activities** to display the vertical bar, in RHEL9 it should be at the bottom



- Select Terminal for CLI interface
- command **tty** display terminal filename

- CLI interface is interactive in text mode responding to commands entered at command prompt.

- In CLI interface, for GUI and MINIMAL interface



```
Red Hat Enterprise Linux 8.3 (Ootpa)
Kernel 4.18.0-240.el8.x86_64 on an x86_64
minimal login: rhuser
Password:
[rhuser@minimal ~]$
```

- you have **cursor** waiting for commands to be given at bash command prompt (bash prompt) (bash shell is default for redhat)
- Bash prompt looks like `[rhuser@rhel83 ~]$` `[rhuser@minimal ~]$`
- bash prompt contains
 - **username** and the **hostname** separated by **@**
 - hostname followed by **:** then **~** and **\$**.
- **~** will be seen when you are in your home directory,
- If you are in a different directory **~** will be replaced with the directory path you are in.
- For example you change to **/etc** directory, **~** is replaced with **/etc**, the directory you are in.
- **\$** displays when logged in user is not root user (userid(UID) is not zero), if **logged in as root user (UID = 0)** it will be displayed as **#**
- you land by default to **your home directory** that is **/home/logged_in_user_name**
- it shows the default command prompt as defined in (Prompt String1) PS1.

```
[rhuser@rhel83 ~]$ echo $PS1
[\u@\h \W]\$
```

Virtual consoles

- In addition to Desktop GUI, there are 4 virtual consoles
- It can be accessed
 - CTL+ALT+F3 for tty3
 - CTL+ALT+F4 for tty4
 - CTL+ALT+F5 for tty5
 - CTL+ALT+F6 for tty6
- Command **tty** displays terminal file name
- CTL+ALT+F1 displays login
- CTL+ALT+F2 displays your GUI

To poweroff, restart

- In GUI interface
 - Click on the power button on the right top to poweroff, restart or cancel
 - In the terminal type **poweroff** command to poweroff the system
 - To restart the system, from terminal **reboot** can be typed.
 - To close terminal gracefully, type **exit** to close the terminal
- In Minimal, CLI interface
 - Typing **poweroff** will poweroff the system
 - Typing **exit** or **logout** will logout
 - Type **reboot** will restart the system

Navigation in UNIX/Linux

- To find the directory currently logged in or where you are currently in the UNIX file system directory structure, command **pwd** can be used.
- To move around the UNIX file system directory structure or change the current working directory, we can use **cd** command
- Command **cd** usages :
 - cd** – changes to your own home directory
 - cd ..** – changes to one higher directory level
 - cd path** – changes to the path specified which could be absolute or relative path
 - cd ../dirpath** – changes to one level higher and then to directory in that level specified
- Absolute path** or **Full path** is the path of the file from root which is **/**
- Relative path** is the path of file relative the current position

```
[rhuser@rhel83 ~]$ pwd
/home/rhuser
[rhuser@rhel83 ~]$ cd /var/log
[rhuser@rhel83 log]$ cd /etc
[rhuser@rhel83 etc]$ pwd
/etc
[rhuser@rhel83 etc]$ cd
[rhuser@rhel83 ~]$ pwd
/home/rhuser
```

- pwd** displays where you are
- cd /var/log** changing your current location to /var/log
- cd /etc**, now it changes to /etc directory
- just **cd** and enter will return to your home directory.

```
[rhuser@rhel83 ~]$ pwd
/home/rhuser
[rhuser@rhel83 ~]$ cd ..
[rhuser@rhel83 home]$ pwd
/home
[rhuser@rhel83 home]$ ls -l
total 4
drwx----- 15 rhuser rhuser 4096 Jan 14 14:50 rhuser
[rhuser@rhel83 home]$ cd
[rhuser@rhel83 ~]$ pwd
/home/rhuser
```

- pwd** to check current location
- cd ..** moves one directory higher from /home/user1 to /home
- pwd** to check current location
- ls -l** list files in long list format in /home directory
- cd** returns to user1 home directory
- pwd** displays current location.

```
[rhuser@rhel83 ~]$ cd /var/spool/mail
[rhuser@rhel83 mail]$ pwd
/var/spool/mail
[rhuser@rhel83 mail]$ cd ../postfix
[rhuser@rhel83 postfix]$ pwd
/var/spool/postfix
[rhuser@rhel83 postfix]$ cd
[rhuser@rhel83 ~]$ pwd
/home/rhuser
```

- cd /var/spool/mail** changes to /var/spool/mail
- pwd** check where you are
- cd ../postfix** moves one directory higher and then changes to directory postfix in /var/spool.
- pwd** check where you are
- cd** to return to your home directory
- pwd** check where you are

UNIX/Linux – command sequence

- At CLI command prompt, interactively commands are used to get responsive output.
- **Linux is case sensitive** - tech is different from Tech, and all **commands are in lower case**,
- UNIX/Linux commands are typed in the following sequence as command syntax

command **options** **arguments**

- As an example, consider the command **ls -l tech***
- In the above unix/linux command

(ls -l tech command displays in long listing format all files starting with tech in the current location)*

ls is the **command**,
-l is the **option**
tech* is the **argument**

- **Options** are mostly in **lower case** but in some instances it could be in upper case, example **ls -Z**, **ls -ltr**
- Also note in case of **options** mostly
- **one hyphen (-)** is used before single letter option, example **ls -l**
- **2 hyphens (--)** are used before word option, example: **ls --help**

- **Space** is must between **command** and **option**, and again **space** between **option** and **arguments**

ls -l tech*

- *(options and/or arguments are not always required with command, examples **ls -l**, **history**, **cd** /etc)*
- Likewise command syntax are used in UNIX/Linux

Unix Help command & Run as Administrator

- Linux OS provides help in the following ways
 - **man *command_name*** helps to display manual pages of the command which has description of the command, its options and most cases usage syntax.
 - example: **man cat**, **man ls**, etc.,
 - **info *command_name*** also helps in providing information on the command, developed in GNU project.
 - ***command_name* --help** in most cases provides options available for that command, in some commands **-h** option can also be used to display the options. example: **ls --help**, **man -h**
- All commands cannot be run by regular user, you may need to run some commands as administrator or root, example when you need to install packages, create users, modify user permissions, etc.,
- To run command as administrator, the command must be preceded with **sudo**, example **sudo mkdir *directory_name***, **sudo dnf install *packagename***
- To use **sudo** command the user running the command should be member of group **wheel** for **redhat** and for **ubuntu** the group is **sudo**.

man man

- Try command **man man** for man command information.
- man is the system's manual pager.
- Each page argument given to man is normally the name of a program, utility or function.
- The manual page associated with each of these arguments is then found and displayed.
- A section, if provided, will direct man to look only in that section of the manual.

The table below shows the section numbers of the manual followed by the types of pages they contain.

1	Executable programs or shell commands
2	System calls (functions provided by the kernel)
3	Library calls (functions within program libraries)
4	Special files (usually found in /dev)
5	File formats and conventions eg /etc/passwd
6	Games
7	Miscellaneous (including macro packages and conventions), e.g. man(7) , groff(7)
8	System administration commands (usually only for root)
9	Kernel routines [Non standard]

A manual page consists of several sections.

Conventional section names include **NAME**, **SYNOPSIS**, **CONFIGURATION**, **DESCRIPTION**, **OPTIONS**, **EXIT STATUS**, **RETURN VALUE**, **ERRORS**, **ENVIRONMENT**, **FILES**, **VERSIONS**, **CONFORMING TO**, **NOTES**, **BUGS**, **EXAMPLE**, **AUTHORS**, and **SEE ALSO**.

LS(1) **User Commands** **LS(1)**

NAME
ls - list directory contents

SYNOPSIS
ls [OPTION]... [FILE]...

DESCRIPTION
List information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

- output of **man ls**
- **ls** is non admin command

PASSWD(5) **File Formats and Conversions** **PASSWD(5)**

NAME
passwd - the password file

DESCRIPTION
/etc/passwd contains one line for each user account, with seven fields delimited by colons (":"). These fields are:

- login name
- optional encrypted password

- output of **man passwd**
- **passwd** is non admin command

USERADD(8) **System Management Commands** **USERADD(8)**

NAME
useradd - create a new user or update default new user information

SYNOPSIS
useradd [options] LOGIN

useradd -D

useradd -D [options]

DESCRIPTION
useradd is a low level utility for adding users. On Debian, administrators should usually use **adduser(8)** instead.

- output of **man 5 passwd**
- **passwd** is file /etc/passwd

Linux Operating System : Kernel

- **Kernel**, heart of an operating system was first developed by Linux Torvalds in 1991
- Linux kernel is **Linux**, this can be found by command **uname -s**

```
[rhuser@rhel83 ~]$ uname -s
```

```
Linux
```
- Linux kernel is loaded by boot loader into memory and kernel manages system resources, loads modules, device drivers, IPC, Security, etc.,
- For Kernel version **uname -r** or **cat /proc/version**

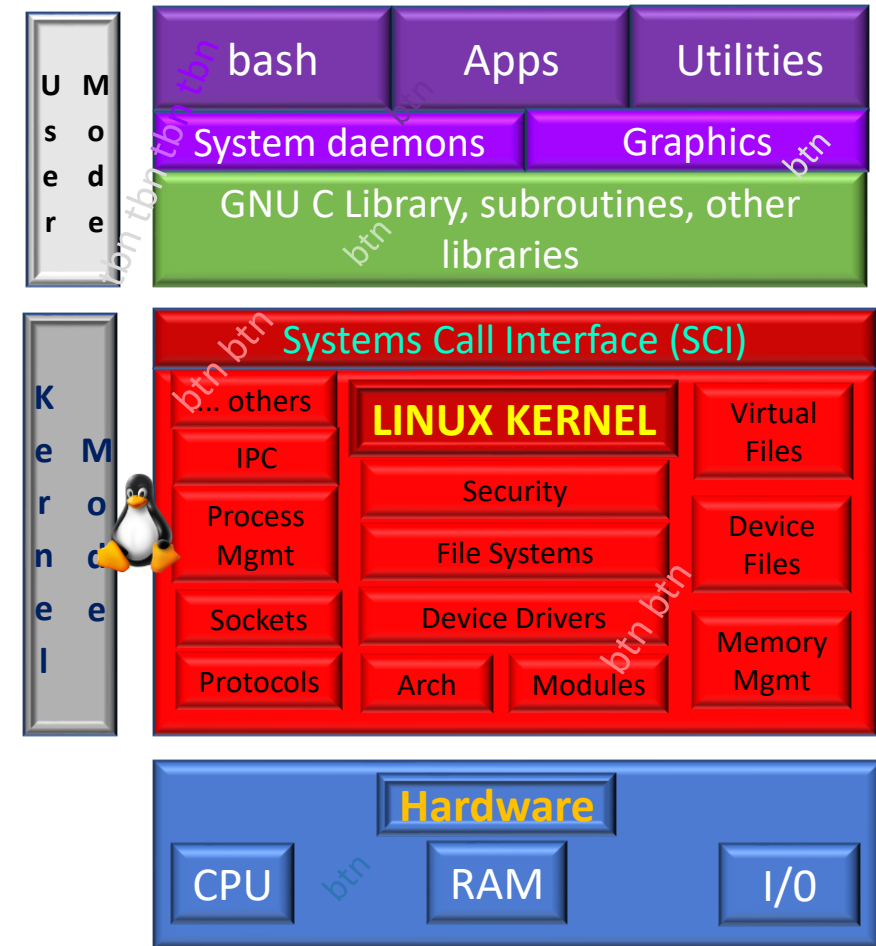
```
[rhuser@rhel83 ~]$ uname -r
```

```
4.18.0-240.el8.x86_64
```

```
[rhuser@rhel83 ~]$ cat /proc/version
```

```
Linux version 4.18.0-240.el8.x86_64 (mockbuild@x86-vm-09.  
build.eng.bos.redhat.com) (gcc version 8.3.1 20191121 (Re  
d Hat 8.3.1-5) (GCC)) #1 SMP Wed Sep 23 05:13:10 EDT 2020
```
- The kernel version is 4.18.0.15, where **4 is Kernel version**, **18 is Major revision**, **0 is Minor revision** and **15 is Patch Number**
- For more info on Ubuntu Kernel refer <https://ubuntu.com/kernel>
- Ken Thompson and Dennis Ritchie on Unix Kernel, [click here](https://www.youtube.com/watch?v=JoVQTPbD6UY) @1.28th <https://www.youtube.com/watch?v=JoVQTPbD6UY>

Linux Operating System



Linux – CLI, System Information

- To find **current day and time** use command **date**
- Systems information** can be displayed with **uname** command, uname command with various options is given below and each option description with **uname --help** command

```
[rhuser@rhel83 ~]$ uname -a
Linux rhel83.domain.net 4.18.0-240.el8.x86_64 #1 SMP Wed Sep 23 05:13:10 EDT 2020 x86_64 x86_64 GNU/Linux
[rhuser@rhel83 ~]$ uname -s
Linux
[rhuser@rhel83 ~]$ uname -n
rhel83.domain.net
[rhuser@rhel83 ~]$ uname -r
4.18.0-240.el8.x86_64
[rhuser@rhel83 ~]$ uname -v
#1 SMP Wed Sep 23 05:13:10 EDT 2020
[rhuser@rhel83 ~]$ uname -m
x86_64
[rhuser@rhel83 ~]$ uname -p
x86_64
[rhuser@rhel83 ~]$ uname -i
x86_64
[rhuser@rhel83 ~]$ uname -o
GNU/Linux
```

```
[rhuser@rhel83 ~]$ uname --help
Usage: uname [OPTION]...
Print certain system information. With no OPTION, same as -s.

-a, --all                print all information, in the following order,
                        except omit -p and -i if unknown:
-s, --kernel-name        print the kernel name
-n, --nodename            print the network node hostname
-r, --kernel-release     print the kernel release
-v, --kernel-version     print the kernel version
-m, --machine            print the machine hardware name
-p, --processor          print the processor type (non-portable)
-i, --hardware-platform  print the hardware platform (non-portable)
-o, --operating-system   print the operating system
--help                  display this help and exit
--version               output version information and exit
```

To find **system related information** **hostnamectl**

```
[rhuser@rhel83 ~]$ hostnamectl
Static hostname: rhel83.domain.net
Icon name: computer-vm
Chassis: vm
Machine ID: 634757c24f4a4ab0996fb633dd5f8abf
Boot ID: c0232ababaa046029a6e9c49ba2a918f
Virtualization: vmware
Operating System: Red Hat Enterprise Linux 8.3 (Ootpa)
CPE OS Name: cpe:/o:redhat:enterprise_linux:8.3:GA
Kernel: Linux 4.18.0-240.el8.x86_64
Architecture: x86-64
```

echo command is used to display the text.

```
[rhuser@rhel83 ~]$ echo "Hello World"
Hello World
```

- To find **where you are** : **pwd**
- To find **logged in user** : **whoami**

```
[rhuser@rhel83 ~]$ pwd
/home/rhuser
```

```
[rhuser@rhel83 ~]$ whoami
rhuser
```

- To **list files in current directory** **ls** command is used

```
[rhuser@rhel83 ~]$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
```

- To list files in **long list format** from current directory **-l** option is used with **ls** command

```
[rhuser@rhel83 ~]$ ls -l
total 0
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Desktop
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Documents
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Downloads
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Music
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Pictures
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Public
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Templates
drwxr-xr-x. 2 rhuser rhuser 6 Jan 10 14:34 Videos
```

UNIX/Linux commands

- Command **whatis** display one line description of manual page of the command

- Example : **whatis cal**

```
[rhuser@rhel83 ~]$ whatis cal
cal (1)          - display a calendar
cal (1p)         - print a calendar
```

- Command **man -f** is also equivalent to **whatis**

```
[rhuser@rhel83 ~]$ man -f cal
cal (1)          - display a calendar
cal (1p)         - print a calendar
```

- Command **whereis** locates source/binary and manuals sections for specified files

- Example: **whereis cal**

```
[rhuser@rhel83 ~]$ whereis cal
cal: /usr/bin/cal /usr/share/man/man1/cal.1.gz /usr/share/man/man1p/cal.1p.gz
```

- Command **which** returns the pathnames of the files (or links) which would be executed in the current environment - **which command searches for commands**

- Example: **which ls**, **which cal**

- Try **man which** for more info on which command.

```
[rhuser@rhel83 ~]$ which cal
/usr/bin/cal
[rhuser@rhel83 ~]$ which mkdir
/usr/bin/mkdir
[rhuser@rhel83 ~]$ which dnf
/usr/bin/dnf
```

- Command **type** display information about command type

- Example **type ls**, **type type**, **type cp**

- Try **help type** for more information

```
[rhuser@rhel83 ~]$ type ls
ls is aliased to `ls --color=auto'
[rhuser@rhel83 ~]$ type mkdir
mkdir is /usr/bin/mkdir
[rhuser@rhel83 ~]$ type cp
cp is /usr/bin/cp
[rhuser@rhel83 ~]$ type find
find is /usr/bin/find
```

Utilities – System Information

date

- print or set the system date and time
- change date and time using **--set** option
syntax : **date --set "YYYY-MM-DD HH:MM"**
- To display date using echo command **echo `date`**
- Refer **man date** for more options and information about **date** command
- command **timedatectl** also displays date and time with time zones and provides information on synchronization, NTP (Network Time Protocol) and RTP (Real time clock)

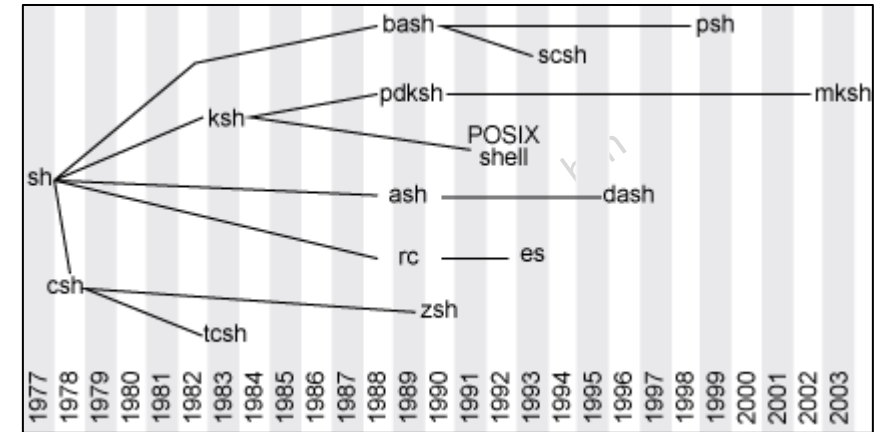
history

- Command **history** helps to list all the commands that has been run previously.
- The listed commands can be used to repeat by just using ! with the history number of the command.
- Re-run the history command by
!number in history
- To execute last command **!!**

Linux Operating System - Shell

• Shell

- The shell sits between user and the kernel, acting as a **command interpreter**.
- It reads your terminal input and translates the commands into actions taken by the system.
- The shell is analogous to **command.com** in MSDOS
- Ken Thompson (of Bell Labs) developed the first shell for UNIX called the *V6 shell* in 1971 which lacked ability to script
- Stephen Bourne at AT&T Bell Labs for V7 UNIX, introduced Bourne shell in 1977, followed by C shell, T(enex) C Shell, K Shell, Bash(Bourne again shell), etc., as per the timeline given aside
- Bash shell is default for recent versions of RedHat and Ubuntu and also for many OS, example CentOS, etc.,
- Installed shells could be known from **/etc/shells** file



<https://developer.ibm.com/tutorials/l-linux-shells/>

To find which shell you have logged in type **echo \$SHELL**
The screenshot shows **/bin/bash** which is **bash shell**.

```
[rhuser@rhel83 ~]$ echo $SHELL
/bin/bash
```

```
user1@ubuntu:~$ echo $SHELL
/bin/bash
```