

# Introduction to UNIX, Linux, and BASH Shell

- What is an Operating System?
- What is UNIX?
- What is Linux?
- What is BASH Shell?



# What is an Operating System?

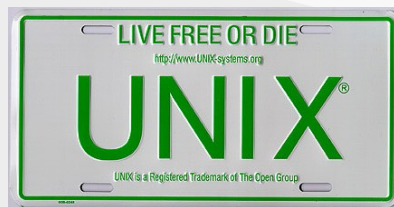
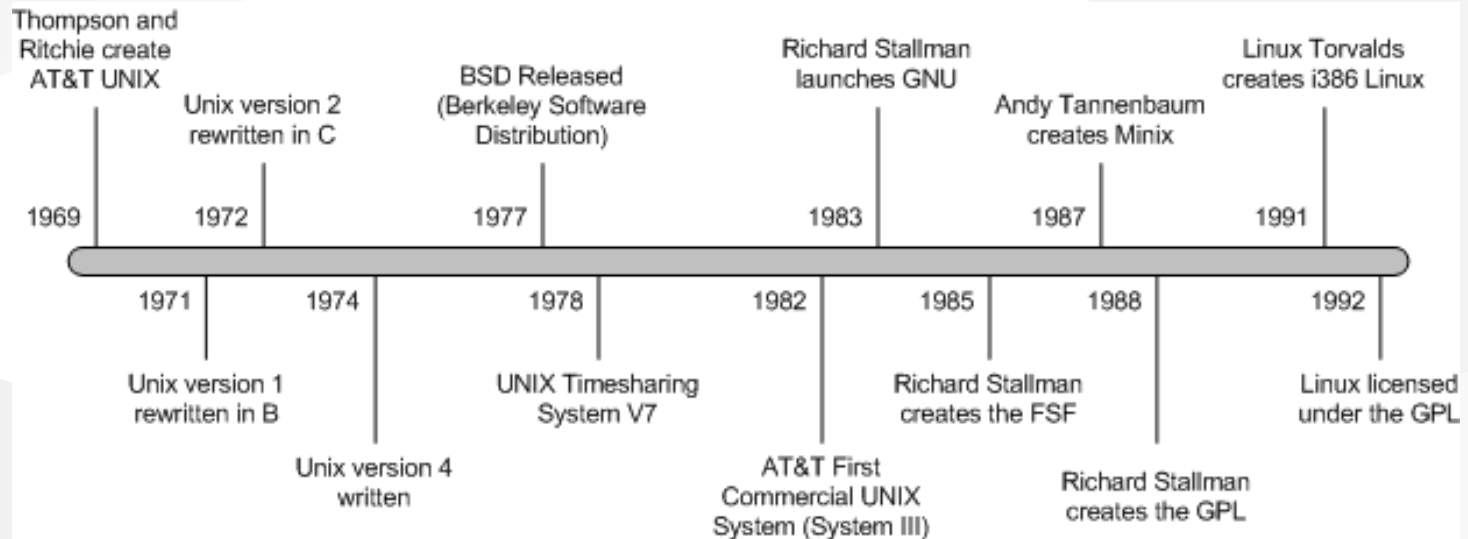
- An operating system (OS) is software that manages the resources of a computer.
- OS is the program which starts up when you turn on your computer and runs underneath all other programs.
- It manages all the available resources.
  - Manage Process
  - Control Hardware
  - Run Applications
  - Manage Data and Files

# Unix History

- 1969 : Developed at AT&T Bell Lab by Ken Thompson and Dennis Ritchie. Ken Thompson developed a new language 'B'.
- 1971 : Denis Ritchie developed 'C' language from 'B'.
- 1972 : Unix was re-written in 'C' language.



# Unix History



# History of Linux

Richard Stallman : author of Emacs, and many other utilities, ls, cat, ...

- 1983 Develop of free UNIX-like operating system
- 1985 Free Software Foundation
  - Run the program, for any purpose.
  - Study how the program works and adapt it.
  - Redistribute copies so you can help others.
  - Improve the program and release your improvements to the public, so that everyone benefits.

# History of Linux

- Linus Torvalds, a student at the University of Helsinki
- 1991 Created a UNIX-like system for Intel 386 processor
  - UNIX was absent from Intel market
  - MINIX : miniture UNIX with more functionality
  - Command-driven interface
  - Brought UNIX features to small computer
- 1992 Released Linux kernel under GNU GPL (GNU General Public License)

# Why Linux?

- Distributions are available for free
- Reliable, Stable and Very Powerful
- Complete development environment
- Share hardware resources and use them fully
- GUIs are more powerful than Mac
- Users need stability, speed, ease of use
- System Admins needs development and networking
- Security is built into the system

# Linux Distributions





# Linux Shell

- Shell is a command interpreter
  - Bourne shell (sh); Steve Bourne, 1978
  - C shell (csh); Bill Joy, 1978
  - Tenex C shell (tcsh); Ken Greer, 1981
  - Korn shell (ksh); David Korn, 1983
  - Almquist shell (ash); Kenneth Almquist, 1989
  - Bourne-Again shell (bash); Brian Fox, 1989
  - Z shell (zsh); Paul Falstad, 1990
- Shell commands : Faster, Remote Access, Repeatable
- Editors : vi, pico, leafpad, etc.

# Linux Shell

```
% command [-option] [arguments]
```

- Command line entries are case sensitive
- Use forward slash (/) not backslash (\)
- Control files and user permissions
- Ordinary files and directories are case sensitive
- Links to documents and directories (shortcuts)
- Hidden files and directories

Command	Description
<code>man</code>	Manual documents
<code>ls</code>	List contents of current directory
<code>cd</code>	Change directory (., ../..)
<code>cp</code>	Copy file
<code>mv</code>	Move (rename) file
<code>rm</code>	Remove file
<code>cat</code>	Concatenate files
<code>more</code>	Page through text

Command	Description
<code>echo</code>	Display a line of text
<code>grep</code>	Pattern matching filter
<code>mkdir</code>	Make a new directory
<code>rmdir</code>	Remove an empty directory
<code>tar</code>	Archiving utility
<code>gzip</code> / <code>gunzip</code>	Compress and Expand files
<code>top</code>	Interactive list of running processes on a system
<code>ps</code>	Print a snapshot of the current processes running

Command	Description
<code>kill</code>	Terminate the process
<code>printenv</code>	Print the values of environment variables
<code>Ctrl-c</code>	Terminate the current command
<code>exit</code>	Exit the terminal
<code>Ctrl-d</code>	Exit the terminal
<code>clear</code>	Clear the terminal
<code>Ctrl-l</code>	Clear the terminal
<code>Ctrl-z</code>	Suspend the current command (Resume fg)

# File and Directory Permissions

```
ls -la
```

```
drwxr-x--- 2 mary users 4096 Dec 28 04:09 tmp
```

```
-rw-r--r-- 1 mary users 969 Dec 21 02:32 foo
```

```
-rwxr-xr-x 1 mary users 345 Sep 1 04:12 somefile
```

- Type of files listed
  - `d`, `-`, `t`
- Permissions are for
  - user (u), group (g), and others (o)
- Permissions are
  - `r` Read 4, `w` Write 2, and `x` Execute 1

## chown - ownership

```
-rw-r--r-- 1 mary users 969  Dec 21 02:32 foo
```

```
chown tom foo
```

```
chown :test foo
```

```
-rw-r--r-- 1 tom test 969  Dec 21 02:32 foo
```

```
chown mary:users foo
```

```
-rw-r--r-- 1 mary users 969  Dec 21 02:32 foo
```

## chmod - mode bits

```
chmod 655 foo
```

```
-rw-r-xr-x 1 mary users 969  Dec 21 02:32 foo
```

```
chmod g-x,o-x foo
```

```
-rw-r--r-- 1 mary users 969  Dec 21 02:32 foo
```

## Processes

- Processes are any programs running on the system  
`ps` and `kill` command can monitor and manage

## Standard Input, Output, and Error

- Keyboard input is standard input (stdin)
- Screen output is standard output (stdout)
- Error messages are standard error (stderr)
- Redirect
  - `>`, `>>`, `<`, `<<`, `|`

## Command Examples on terminal

- See examples on terminal



**Questions?**