Introduction to Unix/Linux

Daniel Lucio

Training Event at UPRM
October 27, 2016
Overview

- Origins
- Features
- What is it?
- The Unix philosophy
- Architecture
- What is a terminal?
- Terminal Emulators
- Why learn it?
- Where to use it?
- What commands are available?
- The Unix shell
- The Unix Filesystem Tree
- GNU == Linux?
- More Information
Origins

ENIAC - 1946
What is an Operating System?

An operating system (O.S.) is software that manages computer hardware resources and provides common services for computer programs.
Examples of Operating Systems
What is an Operating System?

Basic services offered by an Operating System are:

- Kernel
- Program Execution
- Interrupts
- Modes
- Memory Management
- Virtual Memory
- Multitasking
- Multiuser
- Disk access and File system
- Device Drivers
- Networking
- Security
- User Interface
Introduction to Unix

http://bellard.org/jslinux/

TCP: Hash tables configured (established 1024 bind 512)
TCP reno registered
Total HugeTLB memory allocated, 0
io scheduler noop registered
io scheduler anticipatory registered
io scheduler deadline registered
io scheduler cfq registered (default)
Real Time Clock Driver v1.12ac
JS clipboard: I/O at 0x03c0
Serial: 8250/16550 driver $Revision: 1.90 $ 4 ports, IRQ sharing disabled
serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a XScale
RAMDISK driver initialized: 16 RAM disks of 4096K size 1024 blocksize
loop: loaded (max 8 devices)
Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2
ide: Assuming 50MHz system bus speed for PIO modes; override with idebus=xx
hda: JSLinux HARDDISK, ATA DISK drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
hda: max request size: 128KiB
hda: 116736 sectors (59 MB) w/256KiB Cache, CHS=115/16/63
hda: unknown partition table
TCP cubic registered
NET: Registered protocol family 1
NET: Registered protocol family 17
Using IPI Shortcut mode
Time: pit clocksource has been installed.
VFS: Mounted root (ext2 filesystem) readonly.
Freeing unused kernel memory: 128k freed
Booted in 5.344 s
Welcome to JS/Linux
/var/root #
Origins

- Unix (officially trademarked as UNIX) is a multitasking, multi-user computer operating system originally developed in 1969 by a group of AT&T employees at Bell Labs.

- The Open Group, an industry standards consortium, owns the UNIX trademark.

- The term Unix is often used informally to denote any operating system that closely resembles the trademarked system, (Ex: Linux, BSD).
What is it?

- Unix is an operating system that was designed to be portable, multi-tasking and multi-user in a time-sharing configuration.

- Under Unix, the operating system consists of many utilities along with the master control program: the kernel.

- The kernel provides services to start and stop programs, handles the file system and other common "low level" tasks that most programs share, and schedules access to avoid conflicts when programs try to access the same resource or device simultaneously.
The Unix Philosophy

vs
Simple and Elegant Design

- Unix is simple: Only implements a few hundred system calls and all are well defined.
- Everything is a “file”: Provides a “universal” interface.
- Is written in C: Makes it accessible and portable.
- Has fast process creation through the fork() operation whereby a process creates a copy of itself.
- Provides a simple yet robust interprocess communication (IPC) primitives.
- Supports all modern features like multitasking, multithreading, virtual memory, demand paging, shared libraries, TCP/IP networking,…
User’s view

Supercomputer

users

terminal

compute nodes

login nodes
What is a Terminal?

• In the past, a computer terminal was an electronic or electromechanical hardware device that was used for entering data into, and display data from a computer or a computing system.

• The function of a terminal is confined to display and input of data.

• In the present, a personal computer can run terminal emulator software that replicates the function of a terminal, sometimes allowing concurrent use of local programs and access to a distant terminal host system.
What is a Terminal?

It includes all forms of keyboard/screen computer communication devices, including personal computers, diskless workstations, network computers, thin clients, and X terminals, the term dumb terminal is sometimes used to refer to any type of traditional computer terminal that communicates serially over a RS-232 connection that does not locally process data or execute user programs.

Dumb terminal
MacOS Emulator

Terminal Emulators

MacOS Terminal App
Linux computer
Windows Client

http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

[Images of the PuTTY configuration window and the Windows file security warning window]
Is there a GUI for Unix?

Example of a graphical user interface using X11 and KDE.
Why, learn it?

As of June 2013, 95.2% of the Top500 computers use a Linux-like O.S.

Source: http://www.top500.org/statistics/list/
Why, learn it?

Have you noticed that in the movies when the super-hacker wants to break into a super-ultra secure computer the only way to really get it done is by typing on a keyboard??

Try http://geektyper.com/
Why, learn it?

- A command line interface can be a powerful expressive way of communicating with a computer.
- Graphical user interfaces (GUI) make easy tasks easy, while command line interfaces make difficult tasks possible!
- Learning the command line can be challenging and takes real effort. **BUT,** not that it’s so hard, **but, rather** it’s so vast.
- Unlike many other computer skills, knowledge of the command line is long lasting. The skills learned today will be useful 10 years from now.
Why, learn it?

• Android is a Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers.

• The iOS kernel is based on Darwin OS. Darwin forms the core set of components upon which OS X and iOS are based. It is compatible with the Single UNIX Specification version 3 (SUSv3) and POSIX UNIX applications and utilities.
Why, learn it?

According to indeed.com, there are 40 available positions in P.R. today for someone who knows Unix/Linux!
Where to use it?

- Login to a Unix system like ’kraken’ or any other NICS/UT/XSEDE resource.
- Download and boot from a Linux LiveCD either from a CD/DVD or USB drive.
  - http://www.puppylinux.com/
  - http://www.ubuntu.com/
Where to use it?
Where to use it?

- Install Cygwin: a collection of tools which provide a Linux look and feel environment for Windows.

- Online terminal emulator
  - [http://bellard.org/jslinux/](http://bellard.org/jslinux/)
Where to use it?
What type of commands are available?

- Communication
- Comparison
- File Management
- Printing
- Programming
- Searching
- Shells
- Shell programming
- Storage
- System status
- Text processing
Architecture

Computer System

Computer

Hardware

Software

Development Environment

Documentation

utilities (commands)

UNIX

terminal

Users

Introduction to Unix
Bash commands (Linux)
https://goo.gl/5gK1td

 RELATED ARTICLES

UNIX For Dummies Cheat Sheet

UNIX For Dummies, 5th Edition

Logging In to UNIX

General UNIX Tips and Lost-and-Found Insight

UNIX Shell Commands

How to Print with UNIX

UNIX Filenames and Pathnames

UNIX SHELL COMMANDS

Whether your UNIX uses a Bourne-style shell or the C shell, you can do some simple typing to execute a range of commands. The following list shows a list of commands and what to type to execute them:

<table>
<thead>
<tr>
<th>To Do This</th>
<th>Type This</th>
<th>To Do This</th>
<th>Type This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel a command</td>
<td>Ctrl+U, Ctrl+X, or @</td>
<td>List filenames with sizes and dates</td>
<td>ls -l</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Change the working directory to /usr</td>
<td>cd /usr</td>
<td>List hidden files too</td>
<td>is -al</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Change the working directory back to your home directory</td>
<td>cd</td>
<td>List files and pause when screen is full</td>
<td>ls</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Copy a file</td>
<td>cp oldfile newfile</td>
<td>Look at a text file</td>
<td>cat letter</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Copy a file to</td>
<td>cp oldfile</td>
<td>Look at a more letter</td>
<td></td>
</tr>
</tbody>
</table>

http://www.dummies.com/computers/operating-systems/unix/unix-for-dummies-cheat-sheet/
Communication

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ftp</td>
<td>Insecure interactive file transfer program</td>
</tr>
<tr>
<td>login</td>
<td>Sign on to Unix</td>
</tr>
<tr>
<td>mailx</td>
<td>Read and send email</td>
</tr>
<tr>
<td>scp</td>
<td>Secure file transfer</td>
</tr>
<tr>
<td>sftp</td>
<td>Secure interactive file transfer</td>
</tr>
<tr>
<td>slogin</td>
<td>Sign on to remote Unix using secure shell</td>
</tr>
<tr>
<td>ssh</td>
<td>Connect to another system, securely</td>
</tr>
<tr>
<td>telnet</td>
<td>Connect to another system, INSECURELY</td>
</tr>
</tbody>
</table>
Comparisons

cmp  Compare two files, byte by byte
comm Compare items in two sorted files
diff  Compare two files, line by line
diff3 Compare three files
dircmp Compare directories
sdiff  Compare two files, side by side
File Management

- **cd**: Change directory
- **chgrp**: Change file group
- **chmod**: Change access modes on files
- **chown**: Change file owner
- **cksum**: Print a file checksum
- **cp**: Copy files
- **csplit**: Break files at specific locations
- **file**: Determine a file’s type
- **head**: Show the first few lines of a file
- **less**: Advanced file viewer
- **ln**: Create symbolic links
- **locate**: Locate a given file using a database
File Management (cont)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ls</code></td>
<td>List files or contents of directories</td>
</tr>
<tr>
<td><code>md5sum</code></td>
<td>Print a file checksum using MD5 algorithm</td>
</tr>
<tr>
<td><code>mkdir</code></td>
<td>Create a directory</td>
</tr>
<tr>
<td><code>more</code></td>
<td>Display contents of files by screen</td>
</tr>
<tr>
<td><code>mv</code></td>
<td>Move or rename files</td>
</tr>
<tr>
<td><code>pwd</code></td>
<td>Print working (current) directory</td>
</tr>
<tr>
<td><code>rm</code></td>
<td>Remove files</td>
</tr>
<tr>
<td><code>rmdir</code></td>
<td>Remove (empty) directories</td>
</tr>
<tr>
<td><code>split</code></td>
<td>Split files evenly</td>
</tr>
<tr>
<td><code>tail</code></td>
<td>Show the last few lines of a file</td>
</tr>
<tr>
<td><code>wc</code></td>
<td>Count lines, words and characters</td>
</tr>
</tbody>
</table>
## Printing Commands

<table>
<thead>
<tr>
<th>BSD</th>
<th>BSD</th>
<th>System V</th>
<th>BSD</th>
<th>BSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>lpr</td>
<td>lpr</td>
<td>cancel</td>
<td>lpr</td>
<td>lpr</td>
</tr>
<tr>
<td>lpq</td>
<td>lpq</td>
<td>lp</td>
<td>lpq</td>
<td>lpq</td>
</tr>
<tr>
<td>lprm</td>
<td>lprm</td>
<td>lpstat</td>
<td>lprm</td>
<td>lprm</td>
</tr>
<tr>
<td>pr</td>
<td>pr</td>
<td>pr</td>
<td>pr</td>
<td>pr</td>
</tr>
</tbody>
</table>

- **lpr**: Send to the printer
- **lpq**: Get printer status
- **lprm**: Cancel a printer request
- **pr**: Format and paginate for printing
- **cancel**: Cancel a printer request
- **lp**: Send to printer
- **lpstat**: Get printer status
- **pr**: Format and paginate for printing
## Programming

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cc</td>
<td>C compiler</td>
</tr>
<tr>
<td>ctags</td>
<td>C function references (for vi)</td>
</tr>
<tr>
<td>ld</td>
<td>Linker</td>
</tr>
<tr>
<td>lex</td>
<td>Lexical analyzer generator</td>
</tr>
<tr>
<td>make</td>
<td>Execute commands in a specific order</td>
</tr>
<tr>
<td>od</td>
<td>Dump input in various formats</td>
</tr>
<tr>
<td>splint</td>
<td>C program analyzer</td>
</tr>
<tr>
<td>strace</td>
<td>Trace signals and system calls</td>
</tr>
<tr>
<td>strip</td>
<td>Remove data from an object file</td>
</tr>
<tr>
<td>truss</td>
<td>Trace signals and system calls</td>
</tr>
<tr>
<td>yacc</td>
<td>Parser generator. Can be used with lex.</td>
</tr>
</tbody>
</table>
Searching

egrep  Extended version of grep
fgrep  Search files for literal words
find   Search filenames or directories
grep   Search contents of files for a pattern
strings Display text strings found in binary files
Shells
(Command line interpreters)

<table>
<thead>
<tr>
<th>Bourne Family</th>
<th>bash</th>
<th>GNU’s Bourne Again Shell</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ksh</td>
<td>The Korn shell</td>
</tr>
<tr>
<td></td>
<td>pdksh</td>
<td>Public domain Korn shell</td>
</tr>
<tr>
<td></td>
<td>sh</td>
<td>Original Bourne shell</td>
</tr>
<tr>
<td></td>
<td>zsh</td>
<td>The Z-shell</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>C family</th>
<th>csh</th>
<th>The original BSD C shell</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>tcshe</td>
<td>Tenex shell (csh on steroids)</td>
</tr>
</tbody>
</table>
**Shell Programming**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>basename</td>
<td>Return filename of a pathname.</td>
</tr>
<tr>
<td>dirname</td>
<td>Return directory portion of a pathname</td>
</tr>
<tr>
<td>echo</td>
<td>Write arguments to the standard output</td>
</tr>
<tr>
<td>expr</td>
<td>Evaluate expression</td>
</tr>
<tr>
<td>id</td>
<td>Return user identity</td>
</tr>
<tr>
<td>line</td>
<td>Read a line of input</td>
</tr>
<tr>
<td>printf</td>
<td>Formatted output</td>
</tr>
<tr>
<td>sleep</td>
<td>Suspend execution for an interval of time</td>
</tr>
<tr>
<td>test</td>
<td>Test a condition</td>
</tr>
</tbody>
</table>
Storage

- **bunzip2**: Expand compressed files .bz2
- **bzip2**: Compression program
- **cpio**: Copy archives in or out
- **gunzip**: Expand compressed files .gz and .Z
- **gzcat**: Uncompress files on the fly
- **gzip**: File compression program
- **tar**: File/tree directory archiver
- **zcat**: Uncompress files on the fly
## System Status

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>at</td>
<td>Execute commands later</td>
</tr>
<tr>
<td>crontab</td>
<td>Execute commands (periodically) at certain time</td>
</tr>
<tr>
<td>date</td>
<td>Display or set date</td>
</tr>
<tr>
<td>df</td>
<td>Show free disk space and mounted disks</td>
</tr>
<tr>
<td>du</td>
<td>Show disk usage</td>
</tr>
<tr>
<td>env</td>
<td>Show environment variables</td>
</tr>
<tr>
<td>finger</td>
<td>Display information about users</td>
</tr>
<tr>
<td>kill</td>
<td>Terminate a running program</td>
</tr>
<tr>
<td>ps</td>
<td>Show processes</td>
</tr>
<tr>
<td>stty</td>
<td>Set or display terminal settings</td>
</tr>
<tr>
<td>who</td>
<td>Show who is logged on</td>
</tr>
</tbody>
</table>
# Text Processing

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>awk</td>
<td>Pattern-directed scanning and processing language</td>
</tr>
<tr>
<td>cat</td>
<td>Concatenate files or display them</td>
</tr>
<tr>
<td>cut</td>
<td>Select columns for display</td>
</tr>
<tr>
<td>ex</td>
<td>Line editor (underlying vi)</td>
</tr>
<tr>
<td>fmt</td>
<td>Simple text formatter</td>
</tr>
<tr>
<td>iconv</td>
<td>Character set conversion</td>
</tr>
<tr>
<td>join</td>
<td>Merge different columns into a database</td>
</tr>
<tr>
<td>paste</td>
<td>Merge columns or switch order</td>
</tr>
<tr>
<td>sed</td>
<td>Non-interactive text editor</td>
</tr>
<tr>
<td>sort</td>
<td>Sort or merge files</td>
</tr>
<tr>
<td>tr</td>
<td>Translate characters</td>
</tr>
<tr>
<td>uniq</td>
<td>Find repeated or unique lines in a file</td>
</tr>
<tr>
<td>vi</td>
<td>Visual text editor</td>
</tr>
<tr>
<td>xargs</td>
<td>Process many arguments in manageable portions</td>
</tr>
</tbody>
</table>
Text Editing on Unix

- There are screen text editors like vi, vim, ex, emacs, pico, nano,..
- There are graphical text editors like gVim, gEdit, Eclipse, emacs,..
- You can process text files via utilities or custom programs with tools like: sed, awk, gawk, troff, nroff, ...
The Unix Shell

- The shell is the user interface to Unix.
- There is always a default one, but you can choose from different ones with different features.
- The shell is simply a program that allows the system to understand your commands. That is why is called a **Command Line Interpreter**.
The Unix Shell Features

- Cursor movement with keyboard arrow keys.
- Command history
- Automatic completion
Command line session

Result from command

Prompt

Terminal Window

Command
Command line session

user004@sshell  ~ $  
user004@sshell  ~ $ fdsfdsf  
bash: fdsfdsf: command not found  
user004@sshell  ~ $ date  
Thu Oct 24 14:38:15 UTC 2013  
user004@sshell  ~ $ cal  
  October 2013  
Su Mo Tu We Th Fr Sa  
   1 2 3 4 5  
   6 7 8 9 10 11 12  
13 14 15 16 17 18 19  
20 21 22 23 24 25 26  
27 28 29 30 31  
user004@sshell  ~ $ whoami  
user004  
user004@sshell  ~ $ exit
Purpose of the Command-line interface?

• Is a program that takes keyboard commands and passes them to the operating system to carry out.

• Customization of your Unix session: You can setup variables, run initialization files, run startup commands, etc.

• Programming: Shells allow you to create small programs called shells scripts that help automate tasks.
More information

http://www.gnu.org/
http://www.linux.org/
Unix man pages
http://www.ubuntu.com/
http://linuxcommand.org
Operating Systems Evolution

https://www.betterbuys.com/evolution-of-operating-systems/
The Mythical Man-Month
by Frederick P Brooks Jr