COMP 175

System Administration and Security

COMMANDS

```
% cat "food in cans"
cat: can't open food in cans
```

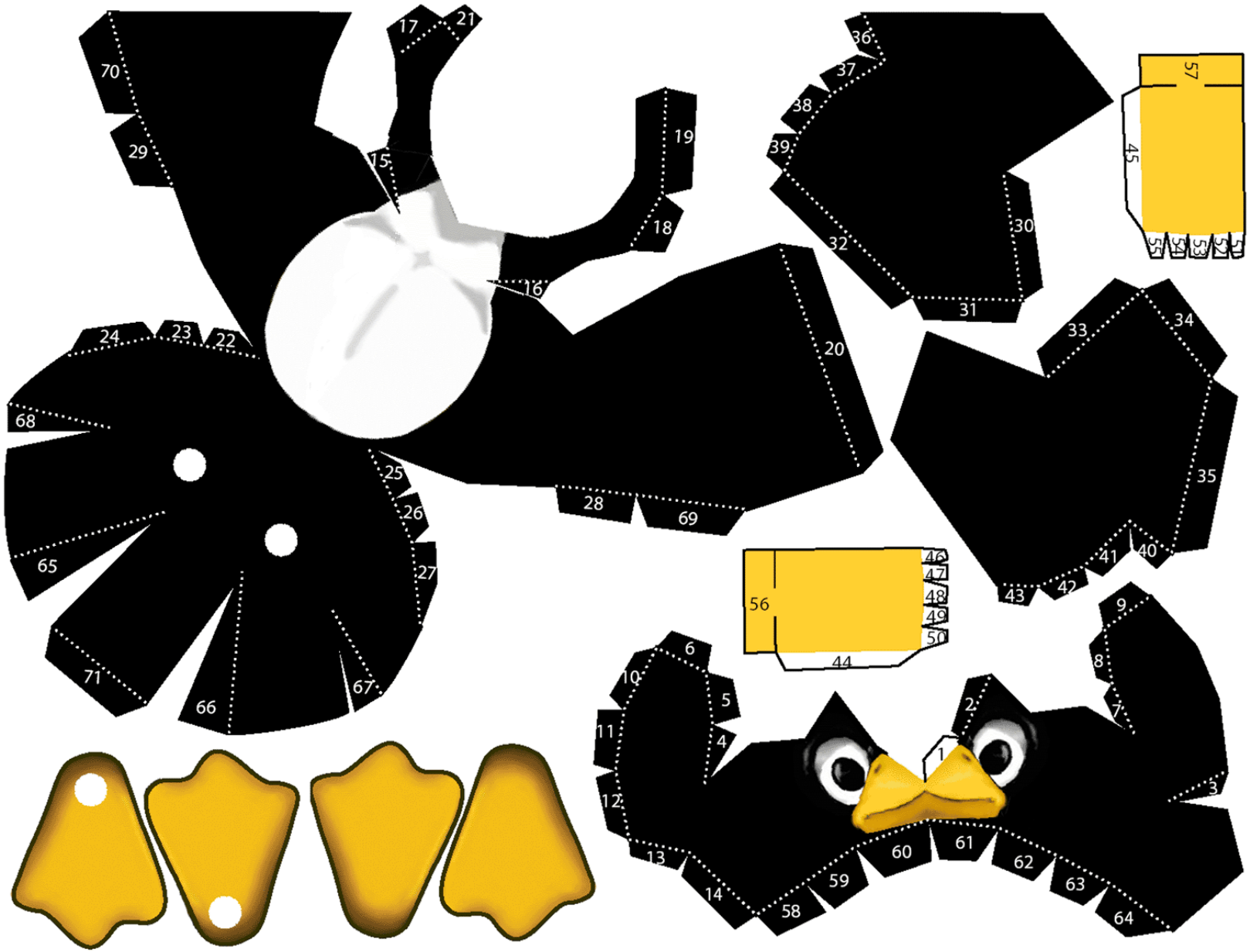
UNIX Philosophy

- Multi-user
  - User needs an account, must log in
  - Complete separation of different users' files and configuration settings
- Small components
  - Each should perform a single task
  - Multiple components can be combined and chained together for more complex tasks
  - An individual component can be substituted for another, without affecting other components

Linux

- Linux kernel (Linus Torvalds)
- Associated utilities
  - Many from the GNU project (Richard Stallman)
  - Recompiled legacy code
- Applications & Desktop
  - GNOME – GTK+ (GIMP Toolkit)
  - KDE – Qt (*cute* framework) (Autodesk Maya, Google Earth, Photoshop Elements, Skype)

Tux


17
21
70
29
15
19
18
16
20
36
37
38
39
30
32
31
57
45
24
23
22
68
25
26
27
65
71
66
67
28
69
33
34
35
41
40
43
42
56
46
47
48
49
50
44
9
8
7
6
10
5
11
4
2
1
3
12
13
14
58
59
60
61
62
63
64

Help

UNIX Help

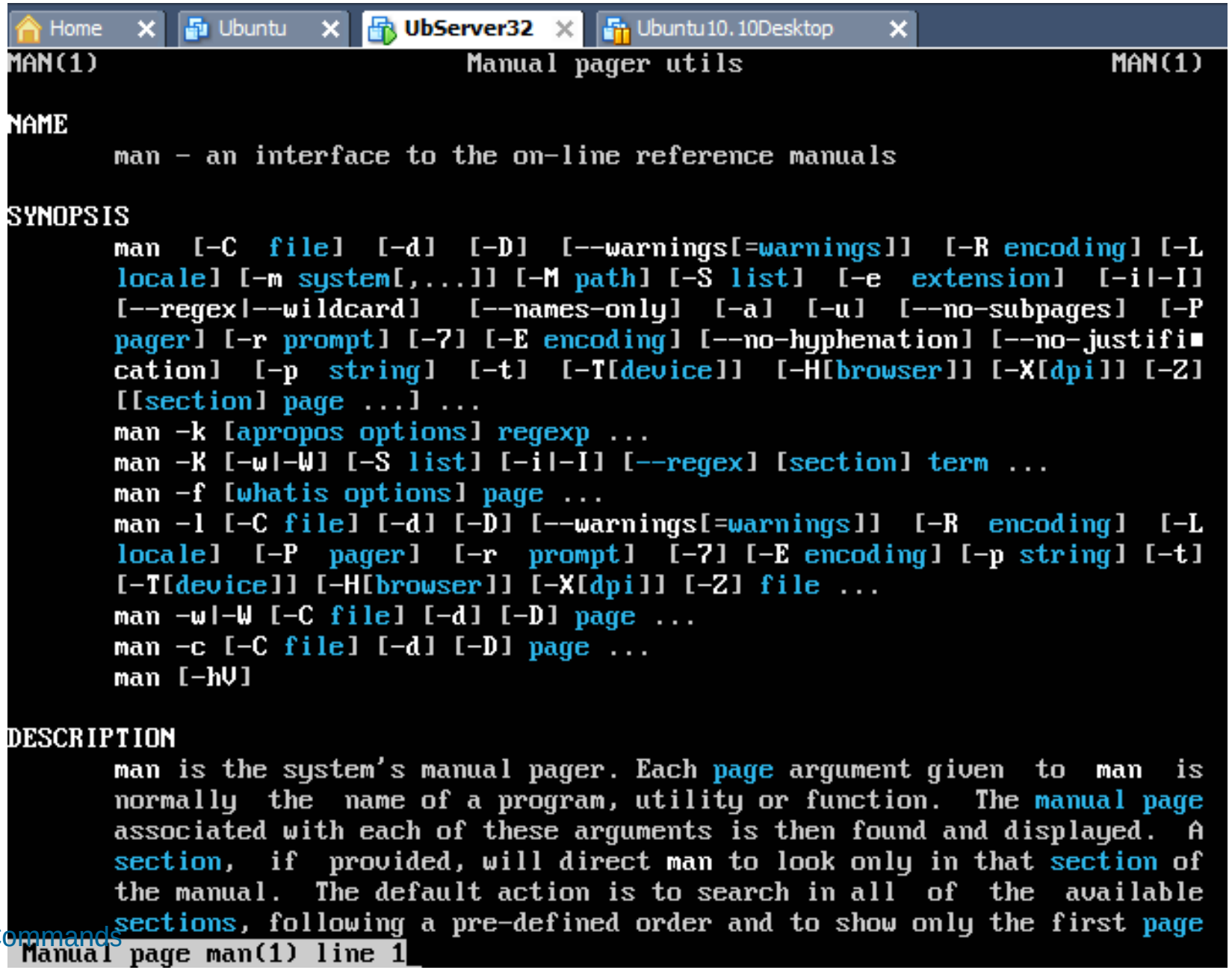
Man pages (manual pages):

- Extensive documentation that comes preinstalled with almost every UNIX/Linux OS.
- Command to display them is: `man <command>`
- Each page is a self-contained document
  - Name
  - Synopsis
  - Description
  - Examples
  - See Also

man

- man <cmd> retrieves detailed info about <cmd>
- man –k <keyword> searches the man page summaries (faster, and will probably give better results)
- man –K <keyword> searches the full text of the man pages

man man


Home
Ubuntu
UbServer32
Ubuntu10.10Desktop
MAN(1) Manual pager utils MAN(1)
NAME
man - an interface to the on-line reference manuals
SYNOPSIS
man [-C file] [-d] [-D] [--warnings[=warnings]] [-R encoding] [-L locale] [-m system[,...]] [-M path] [-S list] [-e extension] [-i|-I] [--regex|--wildcard] [--names-only] [-a] [-u] [--no-subpages] [-P pager] [-r prompt] [-7] [-E encoding] [--no-hyphenation] [--no-justification] [-p string] [-t] [-T[device]] [-H[browser]] [-X[dpi]] [-Z] [[section] page ...] ...
man -k [apropos options] regexp ...
man -K [-w|-W] [-S list] [-i|-I] [--regex] [section] term ...
man -f [whatis options] page ...
man -l [-C file] [-d] [-D] [--warnings[=warnings]] [-R encoding] [-L locale] [-P pager] [-r prompt] [-7] [-E encoding] [-p string] [-t] [-T[device]] [-H[browser]] [-X[dpi]] [-Z] file ...
man -w|-W [-C file] [-d] [-D] page ...
man -c [-C file] [-d] [-D] page ...
man [-hV]
DESCRIPTION
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 default action is to search in all of the available sections, following a pre-defined order and to show only the first page
Manual page man(1) line 1

UNIX Help

- GUI's may provide HTML versions (man2html)
- Generally split into eight numbered sections:

1. General commands
2. System calls
3. C library functions
4. Special files (devices in /dev) and drivers
5. File formats and conventions
6. Games and screensavers
7. Miscellanea
8. System administration commands and daemons
9. Kernel routines (Ubuntu - non-standard)

RTFM

RTFM - SysAdmin acronym for:

- Read The ... Man page
- (or Read The ... Manual)

Similar to:

- GIYF ("Google is your friend")
- LMGTFY ("let me google that for you").

- Also the Ubuntu Desktop Guide (Help)
- Users often struggle instead of reading the man pages. Learn how to use and decipher them.
- Apps lacking man pages are *challenged*

root@localhost:/usr/share/doc

File Edit View Terminal Tabs Help

```
File: dir       Node: Top       This is the top of the INFO tree

  This (the Directory node) gives a menu of major topics.
  Typing "q" exits, "?" lists all Info commands, "d" returns here,
  "h" gives a primer for first-timers,
  "mEmacs<Return>" visits the Emacs topic, etc.

  In Emacs, you can click mouse button 2 on a menu item or cross reference
  to select it.

* Menu:

Texinfo documentation system
* Info: (info).                 Documentation browsing system.
* Pinfo: (pinfo).           curses based lynx-style info browser.
* Texinfo: (texinfo).           The GNU documentation format.
* info standalone: (info-stnd).            Read Info documents without Emacs.
* infokey: (info-stnd)Invoking infokey.    Compile Info customizations.
* install-info: (texinfo)Invoking install-info. Update info/dir entries.
* makeinfo: (texinfo)Invoking makeinfo.         Translate Texinfo source.
* texi2dvi: (texinfo)Format with texi2dvi.      Print Texinfo documents.
* texi2pdf: (texinfo)PDF Output.                PDF output for Texinfo.
* texindex: (texinfo)Format with tex/texindex.  Sort Texinfo index files.

Miscellaneous
* As: (as).                     The GNU assembler.
* Binutils: (binutils).         The GNU binary utilities.
* Gas: (as).                    The GNU assembler.
* Gpm: (gpm).   A server wich hands mouse events to non-X programs.
* Groff: (groff).               The GNU troff document formatting system.
* Ld: (ld).                       The GNU linker.
-----Info: (dir)Top, 1963 lines --Top-----------------------------------------------------
Welcome to Info version 4.8. Type ? for help, m for menu item.
```

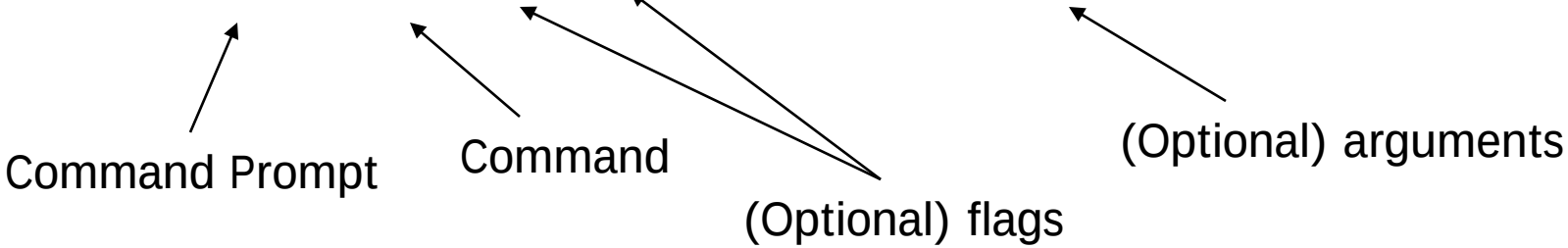
Commands

The Command Prompt

- Commands are the way to “do things” in Unix
- Consist of a command name and options called “flags”
- Commands are typed at the *command prompt*
- In Unix, *everything* (including commands) is case-sensitive

```
[prompt]$ <command> <flags> <args>
```

```
fiji:~$ ls –l -a unix-tutorial
```



Note: Many Unix commands will print a message only if something went wrong. Be careful with rm and mv.

Command Syntax

- Most commands take parameters
  - Some commands require them
  - Parameters are also known as arguments
- Commands are case-sensitive
  - Usually lower-case
- For example, echo simply displays its arguments:

```
$ echo

$ echo Hello there
Hello there
$ ECHO HELLO THERE
bash: ECHO: command not found
```

Command History

- Want to repeat a previously-executed command?
- The shell keeps a command history for this purpose
- Up and Down cursor keys scroll list of previous commands
- Press Enter to execute the displayed command
- Commands can also be edited before being run
- Useful for fixing a typo in the previous command
- Left and Right cursor keys navigate across a command
- Extra characters can be typed at any point
- Backspace deletes characters to the left of the cursor
- Del and Ctrl+D delete characters to the right
- Take care not to log out by holding down Ctrl+D too long

Permissions

- There are three such special permissions within Linux. They are:
- setuid — used only for applications, this permission indicates that the application is to run as the owner of the file and not as the user executing the application. It is indicated by the character s in place of the x in the owner category. If the owner of the file does not have execute permissions, the S is capitalized to reflect this fact.
- setgid — used primarily for applications, this permission indicates that the application is to run as the group owning the file and not as the group of the user executing the application. The setgid permission is indicated by the character s in place of the x in the group category. If the group owner of the file or directory does not have execute permissions, the S is capitalized to reflect this fact.
- sticky bit — used primarily on directories, this bit dictates that a file created in the directory can be removed only by the user that created the file. It is indicated by the character t in place of the x in the everyone category. If the everyone category does not have execute permissions, the T is capitalized to reflect this fact.

chown – change owner

- **Identities:**
  - u - the user who owns the file (the owner)
  - g - the group to which the user belongs
  - o - others (not owner or the owner's group)
  - a - everyone or all (u, g, and o)
- **Permissions:**
  - r - read access
  - w : write access
  - x : execute access
- **Actions:**
  - + : adds the permission
  - - : removes the permission
  - = : makes it the only permission

chmod examples

- g+w adds write access for the group
- o-rwx removes all permissions for others
- u+x allows file owner to execute the file
- a+rw allows all to read and write to the file
- ug+r allows owner and group to read the file
- g=rx allows only group to read and execute (not write)
- -R change permissions for directory trees
- Note: execute directory permission is whether directory search is/is not allowed

chmod - numerical

- Each permission setting has a numerical value:

  r = 4 w = 2 x = 1 - = 0

- When these values are added together, the total is used to set specific permissions. For example, if you want read and write permissions: 4 (read) + 2 (write) = 6.

Numerical Permission

- -rw------- (600) Only the owner has read and write permissions.
- -rw-r--r-- (644) Only the owner has read and write permissions; The group and others have read only.
- -rwx------ (700) Only the owner has read, write, and execute
- -rwxr-xr-x (755) The owner has read, write, and execute permissions; The group and others have only read and execute.
- -rwx--x--x (711) The owner has read, write, and execute permissions; The group and others have only execute.
- -rw-rw-rw- (666) Everyone can read and write to the file. (Be careful with these permissions.)
- -rwxrwxrwx (777) Everyone can read, write, and execute. (Again, this permissions setting can be hazardous)
- drwx------ (700) Only the user can read, write in this directory.
- drwxr-xr-x (755) Everyone can read the directory; users and groups have read and execute permissions.

I/O Redirection

- "Glue" UNIX utilities together to use them effectively
- Use the sort filter and related commands to operate on files
- Use the sed, grep, and awk commands to search files and select desired fields
- Standard input = issuing a command that the OS reads and processes
- Standard output = first stream
- Standard error = second stream

CLI Redirection

Output Redirection

- > Redirect standard output to a file. Creates the file if not present, otherwise overwrites it.
- >> Append standard output to a file.
- 2> redirect standard error to a file
- 2>>Appends standard error to a file
- &> redirect standard output & error to a file

Input Redirection

- < redirect file to standard input

Pipe

- | Chain processes together

Linux Commands

Linux Commands exist for:

- File Management and Viewing
- Filesystem Management
- Help, Job and Process Management
- Network Management
- System Management
- User Management
- Printing and Programming
- Document Preparation
- Miscellaneous

File Management and Viewing

- Copy and rename files
  - cp: copies files
  - mv: moves and renames files and directories
  - rm: deletes files

File and Directory Management

- cat View a file

  Ex: cat filename

  cat student_list

  Mike

  Sue

  Control D after a line break

- cmp Compare two files.
- cut Remove sections from each line of files.
- diff Show the differences between files.

  Ex: diff file1 file2 : Find differences between file1 & file2

- echo Display a line of text.

Filesystem Management

- more: views text files page by page
- less: similar to more, allows backward movement
- tac: prints file contents in reverse order
- head and tail: views first/last few lines of a file
- nl: numbers the lines of a file
- wc: displays count of lines, words, and characters
  - wc –l filename
- diff: reports differences between files
- od: displays a binary file in human-readable form
- strings: finds printable characters in a binary file

File and Directory Management

- chown Change owner

  Ex: **`chown <owner1> <filename>`**

  Change ownership of a file to owner1.
- chgrp Change group.

  Ex: **`chgrp <group1> <filename>`**

  Change group of a file to group1.
- cp Copy a file from one location to another

  Ex: **`cp file1 file2`** Copy file1 to file2

  Ex: **`cp –R dir1 dir2`** Copy dir1 to dir2
- md5sum Prints the MD5 Checksum

File and Directory Management

File compression, backing up and restoring

- compress Compress data
- uncompress Expand data
- gzip - zip a file to a gz file
- gunzip - unzip a gz file
- tar Archives files and directories

  Ex: tar -zcvf <destination> <files/directories>
- tar –zxvf <compressed file> to uncompress
- zip – Compresses a file to a .zip file
- unzip – Uncompresses a file with .zip extension

Misc. Commands

- date shows and sets time and date
- w lists logon information about users
- cal provides a monthly calendar
- bc runs a calculator utility

File and Directory Management

- cd <directory path> Change directory
- With no arguments changes to users home directory
- chmod Change the file permissions.
- Ex: chmod 751 myfile : change the file permissions to rwx for owner, rx for group and x for others
- Ex: chmod go=+r myfile : Add read permission for the group and others (character meanings u-user, g-group, o-other, + add permission,-remove,r-read,w-write,x-exe)
- Ex: chmod +s myfile - Setuid bit on the file which allows the program to run with user or group privileges of the file

File and Directory Management

- grep List all files with the specified expression. (grep pattern <filename/directorypath>)

  Ex: ls –l |grep sidbi : List all lines with a sidbi in them.

  Ex: grep " R " : Search for R with a space on each side
- sleep Delay for a specified amount of time.
- sort Sort a file alphabetically.
- uniq Remove duplicate lines from a sorted file.
- wc Count lines, words, characters in a file.

  Ex. wc –c/w/l <filename> (counts/words/lines)

File Manipulation

- sort - Sorts files
- grep - Searches for patterns
- awk - Processes its own programming language
- sed - Allows editing file contents without opening

lsof

- list open files
  - in Unix just about everything is a file
  - (including a network socket)

```
# lsof –i        (sudo or root shell)      (but not on our desktops)
COMMAND    PID   USER   FD   TYPE  DEVICE SIZE/OFF NODE NAME
inetd     1773   root   4u  IPv4    5018      0t0  TCP *:time (LISTEN)
inetd     1773   root   5u  IPv4    5019      0t0  UDP *:time
inetd     1773   root   6u  IPv4    5020      0t0  TCP *:telnet (LISTEN)
inetd     1773   root   7u  IPv4    5021      0t0  UDP *:biff
inetd     1773   root   8u  IPv4    5022      0t0  TCP *:pop3 (LISTEN)
inetd     1773   root   9u  IPv4    5023      0t0  TCP *:auth (LISTEN)
sshd      1778   root   3u  IPv4    5045      0t0  TCP *:ssh (LISTEN)
named     1781   root  20u  IPv4    5067      0t0  TCP localhost:domain (LISTEN)
named     1781   root  21u  IPv4    5069      0t0  TCP hatter.treacle.com:domain (LISTEN)
named     1781   root  22u  IPv4    5071      0t0  TCP tea.treacle.com:domain (LISTEN)
```

lsof

- Show all networking related to a given port

  lsof -i :port
- Show connections to a specific host

  lsof -i@host_ip
- Show connections to a specific host and port

  lsof -i@host_ip:port

```
root@hatter:/etc# lsof -i :22
COMMAND   PID USER   FD   TYPE  DEVICE SIZE/OFF NODE NAME
sshd     1778 root    3u  IPv4    5045      0t0  TCP *:ssh (LISTEN)
sshd     1778 root    4u  IPv6    5047      0t0  TCP *:ssh (LISTEN)
sshd    10662 root    3r  IPv4 2172611      0t0  TCP hatter.treacle.com:ssh->10.0.0.5:52490 (ESTABLISHED)
root@hatter:/etc#
```

lsof

- lsof -i| grep LISTEN
- lsof -i| grep ESTABLISHED

```
root@hatter:/etc# lsof -i| grep LISTEN
inetd      1773   root    4u  IPv4    5018      0t0  TCP *:time (LISTEN)
inetd      1773   root    6u  IPv4    5020      0t0  TCP *:telnet (LISTEN)
inetd      1773   root    8u  IPv4    5022      0t0  TCP *:pop3 (LISTEN)
inetd      1773   root    9u  IPv4    5023      0t0  TCP *:auth (LISTEN)
sshd       1778   root    3u  IPv4    5045      0t0  TCP *:ssh (LISTEN)
sshd       1778   root    4u  IPv6    5047      0t0  TCP *:ssh (LISTEN)
named      1781   root   20u  IPv4    5067      0t0  TCP localhost:domain (LISTEN)
named      1781   root   21u  IPv4    5069      0t0  TCP hatter.treacle.com:domain (LISTEN)
named      1781   root   22u  IPv4    5071      0t0  TCP tea.treacle.com:domain (LISTEN)
```

- Show what a user has open lsof –u user_name
- Show files and connections a command is using lsof –c program

```
root@hatter:/etc# lsof -c syslog
COMMAND  PID USER   FD   TYPE     DEVICE SIZE/OFF    NODE NAME
syslogd 1401 root  cwd    DIR        8,1     4096       2 /
syslogd 1401 root  rtd    DIR        8,1     4096       2 /
syslogd 1401 root  txt    REG        8,1    31508 1055783 /usr/sbin/syslogd
syslogd 1401 root  mem    REG        8,1  1651695 3801163 /lib/libc-2.13.so
syslogd 1401 root  mem    REG        8,1    49949 3801172 /lib/libnss_files-2.13.so
syslogd 1401 root  mem    REG        8,1   136521 3801205 /lib/ld-2.13.so
syslogd 1401 root    0u  unix 0xf5b35b80      0t0    4372 /dev/log
syslogd 1401 root    2w   REG        8,1    19723 2503068 /var/log/messages
syslogd 1401 root    3w   REG        8,1        0 2503074 /var/log/syslog
syslogd 1401 root    4w   REG        8,1        0 2503062 /var/log/debug
syslogd 1401 root    5w   REG        8,1     8179 2503072 /var/log/secure
syslogd 1401 root    6w   REG        8,1        0 2503061 /var/log/cron
syslogd 1401 root    7w   REG        8,1    65245 2503067 /var/log/maillog
syslogd 1401 root    8w   REG        8,1        0 2503073 /var/log/spooler
```

lsof

- Kill everything Wayne has open
- kill -9 `lsof -t -u wayne`

- lsof +L1 shows you all open files that have a link count less than 1, often indicative of a cracker trying to hide something
- lsof +L1

Commands

- ls LiSts the contents of a specified files or directories (or the current directory if no files are specified)
  - Syntax: ls [<args> ...]
  - Example: ls backups/
  - ls –al all – include files starting with .
  - ls –alh human readable file size
  - man ls Why so many options?
- pwd Print Working Directory (where am I?)

Files

- cp CoPies a file, preserving the original
  - Syntax: cp <sources> <destination>
  - Example: cp tutorial.txt tutorial.txt.bak
- mv MoVes/renames a file, destroying the original
  - Syntax: mv <sources> <destination>
  - Examples:
    - mv tutorial.txt tutorial.txt.bak
    - mv tutorial.txt tutorial-slides.ppt backups/

Note: Both of these commands will over-write existing files without warning you!

Commands

- cd Change Directory

  (to home directory if unspecified)
  - Syntax: cd <directory>
  - Examples:
  - cd usr/local/bin
  - cd ../sbin
- mkdir MaKe DIRectory
  - Syntax: mkdir <directories>
  - Example: mkdir backups-notes
- rmdir ReMove DIRectory, which must be empty
  - Syntax: rmdir <directories>
  - Example: rmdir backups-notes

Two Ways To Find Files

find (find <start directory> -name <file name> -print)
Ex: find /home –name readme -print

(Search for readme starting at home and output full path.)
"/home" = Search starting at the home directory and proceed through all its subdirectories

- "-name readme" = Search for a file named readme
- "-print" = Output the full path to that file

locate File locating program that uses the locate database.

- Ex: locate –u to create the database,
- locate <file/directory> to find file/directory

Files

- pwd Print/list present working directory with full path
- touch Change file timestamps to the current time. Make the file if it doesn't exist. (touch <filename>)
- whereis Locate the binary and man page files for a command. (whereis <program/command>)
- which Show full path of commands where given commands reside. (which <command>)

Text Manipulation

File viewing and editing

- emacs Full screen editor.
- pico Simple text editor.
- vi Editor with a command mode and text mode. Starts in command mode.
- gedit GUI Text Editor
- tail Look at the last 10 lines of a file.
- Ex: tail –f <filename>
- Ex: tail -100 <filename>
- head Look at the first 10 lines of a file.
  - Ex. head <filename>

Compression

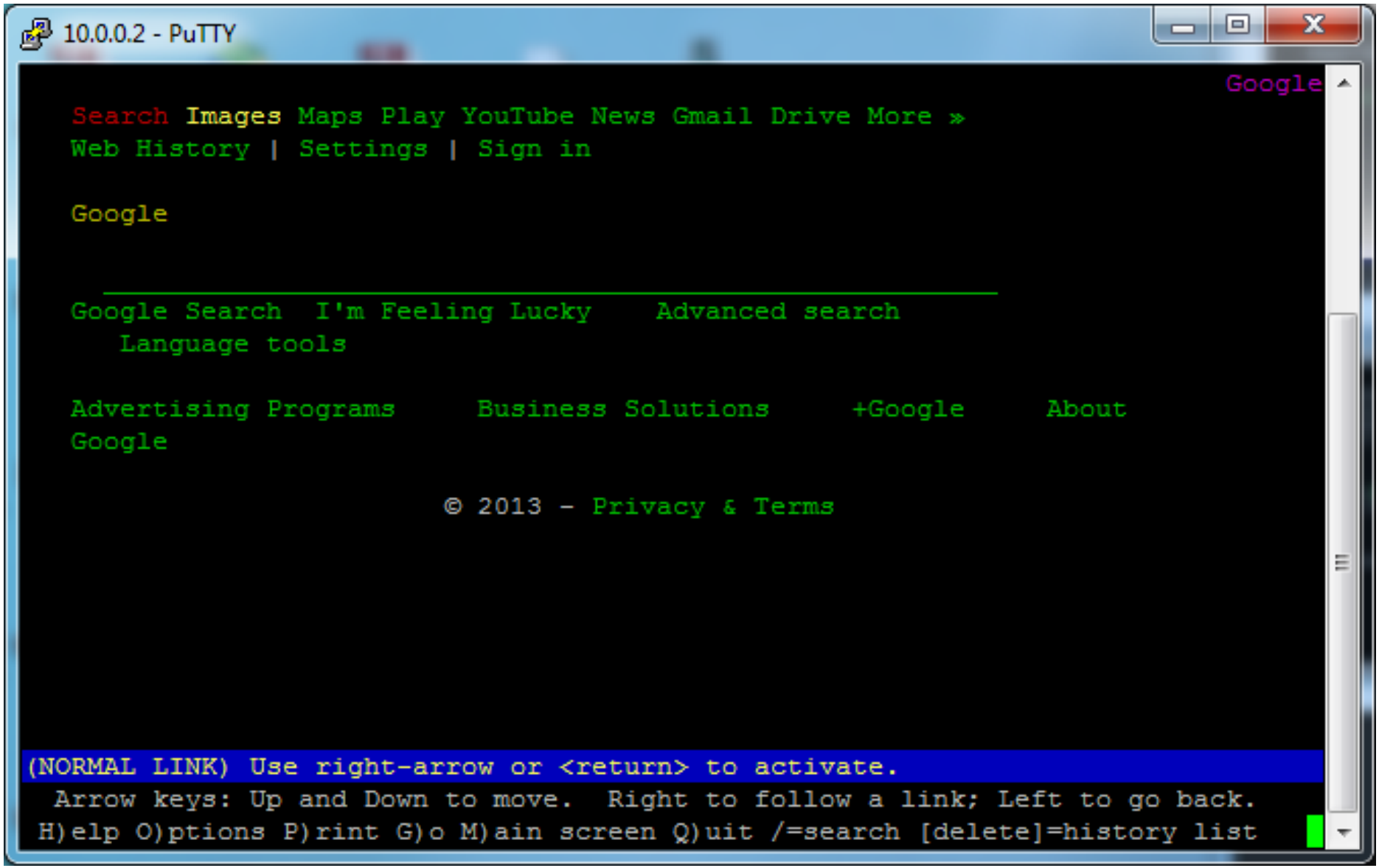
File compression, backing up and restoring

- compress Compress data.
- uncompress Expand data.
- cpio Can store files on tapes. to/from archives.
- gzip - zip a file to a gz file.
- gunzip - unzip a gz file.
- tar Archives files and directories. Can store files and directories on tapes.
- Ex: tar -zcvf <destination> <files/directories> - Archive copy groups of files. tar –zxvf <compressed file> to uncompress
- zip – Compresses a file to a .zip file.
- unzip – Uncompresses a file with .zip extension.

Network Management

- Telnet
- FTP
- Lynx – A very handy tool

lynx


10.0.0.2 - PuTTY
Google
Search Images Maps Play YouTube News Gmail Drive More »
Web History | Settings | Sign in
Google
Google Search I'm Feeling Lucky Advanced search
Language tools
Advertising Programs Business Solutions +Google About
Google
© 2013 - Privacy & Terms
(NORMAL LINK) Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list

File Systems

- On UNIX systems, the **df** command reveals statistics concerning individual file systems in a system's file tree, whereas the **du** command produces a list of the space consumed by directories.
- On Windows systems this information is available using a graphical file tree viewer such as the explorer program or the web browser.

Dot Files

- .bash_logout - file executed by bash shell on logout
- .bash_profile - initialization of bash shell run only on login. Bash looks first for a .bash_profile file when started as a login shell or with the -login option. If it does not find .bash_profile, it looks for .bash_login. If it doesn't find that, it looks for .profile. System-wide functions and aliases go in /etc/bashrc and default environment variables go in /etc/profile.
- .bashrc - initialization command run when bash shell starts up as a non-login shell
- .cshrc - initialization commands that are run automatically (like autoexec.bat) when C shell is initiated
- .emacs - configuration file for emacs editor
- .fvwmrc - configuration file for fvwm window manager
- .fvwm2rc - configuration file for fvwm2 window manager
- .jedrc - configuration file for the jed text editor
- .lessrc - typically contains key bindings for cursor movement with the less command
- .login - initialization file when user logs in
- .logout - commands run when user logs out
- .wm_style - gives choice of default window manager if one is not specified in startx
- .Xdefaults - sets up X resources for individual user. The behavior of many different application programs can be changed by modifying this file.
- .xinitrc - initialization file when running startx. Can be used to activate applications, run a given window manager, and modify the appearance of the root window.
- .xsession - configuration file for xdm

UNIX Command References

- Cheatsheet: cb.vu/unixtoolbox.xhtml
- Tutorials: www.ee.surrey.ac.uk/Teaching/Unix/
- Shells: linuxcommand.org/learning_the_shell.php
- Reference: http://www.pixelbeat.org/cmdline.html
- Commands: http://www.oreillynet.com/linux/cmd/
- Cross ref: http://bhami.com/rosetta.html
- Oracle: http://www.oracle.com/technetwork/systems/index.html
- http://www.ibm.com/developerworks/aix/

Remember

Basic Utilities

- Directory/File management: cd, ls, pwd, mkdir, rmdir, cp, mv, rm, find, du, file
- File viewing/editing: touch, more, less, ed, vi, emacs
- User management: passwd, chmod, chown, su, who
- Process management: kill, killall, ps
- Documentation: man, info, /usr/share/doc
- Networking: lynx