

Introduction to Linux/Unix

Tinghua Chen

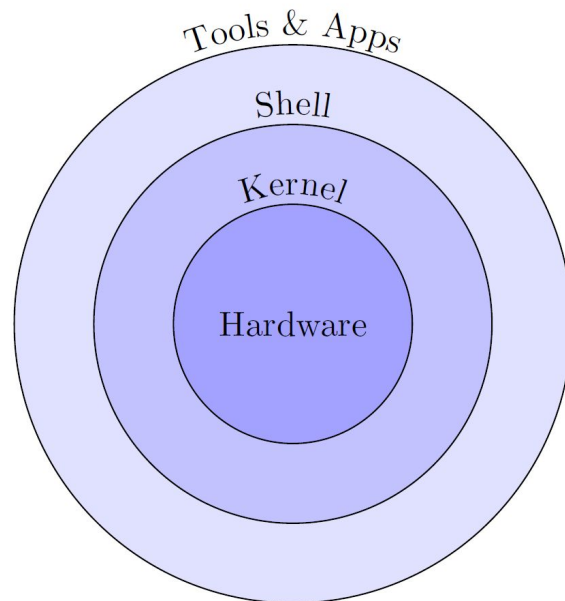
IT Research CyberInfrastructure (IT-RCI)

Workshop Overview

- Linux/Unix Operating System
- Command-line Interface
- Files and Directories
- Input/Output Redirection
- File Permissions
- Basic Data Transfer
- Compress and Decompress Files
- Environment Variables and PATH
- Basic Shell Scripting

Linux/Unix Operating System

- Kernel:
 - Core part of Linux OS
 - Manage resources
 - Between hardware and shell
- Shell:
 - Interface between user and kernel
 - Interpret commands and execute them
 - Type of Shell
 - Bash, zsh, csh, etc.



1969
1971 to 1973
1974 to 1975
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001 to 2002
2003
2004
2005 to 2007
2008 to 2009
2010
2011 to 2018
2019 to 2023

1969
1971 to 1973
1974 to 1975
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001 to 2002
2003
2004
2005 to 2007
2008 to 2009
2010
2011 to 2018
2019 to 2023

Legend:

- Open source (Green box)
- Mixed/shared source (Blue box)
- Closed source (Red box)

Unix-like systems

Linux first released in 1991

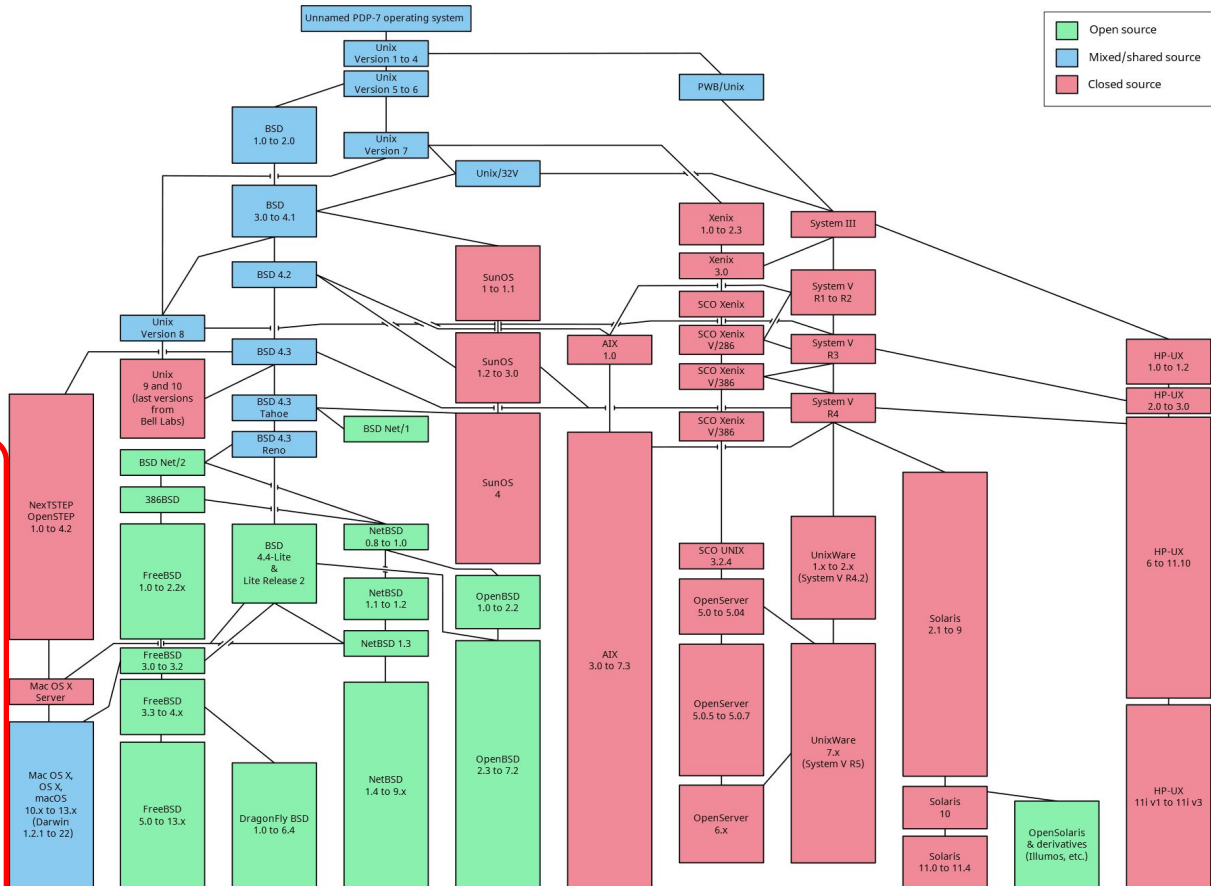
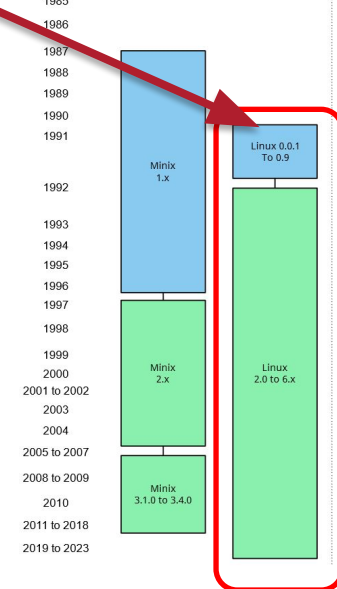
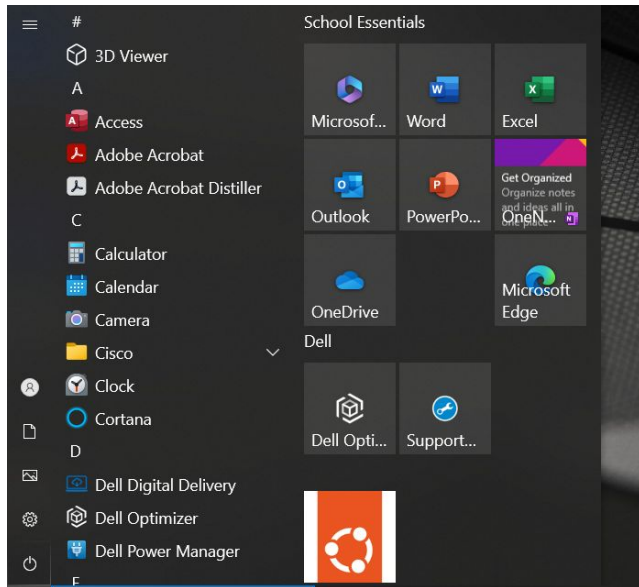


Image by Eraserhead1, Infinity0, Sav_vas.
https://commons.wikimedia.org/wiki/File:Uix_history-simple.svg

Graphical User Interface (GUI)

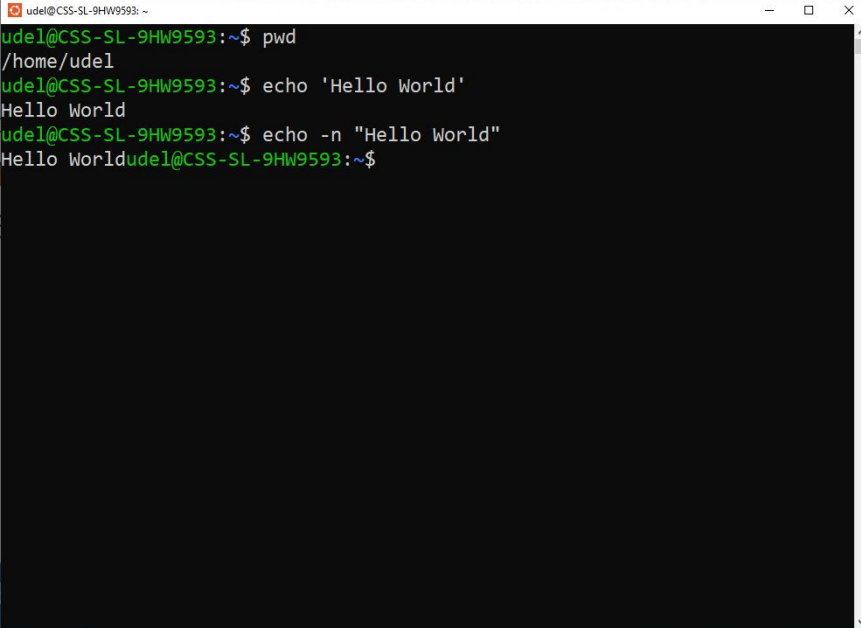


Command-line Interface (CLI)



General Command Syntax

- `command` `option(s)` `argument(s)`
- There may be zero or more options
- Example commands:
 - `pwd`
 - `echo 'Hello World'`
 - `echo -n "Hello World"`

A terminal window titled 'udel@CSS-SL-9HW9593: ~' with standard window controls. It shows a sequence of commands and their outputs: 'pwd' returns '/home/udel', 'echo 'Hello World'' outputs 'Hello World', and 'echo -n "Hello World"' outputs 'Hello World' without a trailing newline.

```
udel@CSS-SL-9HW9593:~$ pwd
/home/udel
udel@CSS-SL-9HW9593:~$ echo 'Hello World'
Hello World
udel@CSS-SL-9HW9593:~$ echo -n "Hello World"
Hello Worldudel@CSS-SL-9HW9593:~$
```

Finding Help for Commands

- `man <command>`
 - Open the manual pages
- `<command> --help`
- `info <command>`
 - Display information in the document format
- `apropos <keyword>`
 - Search the descriptions for the installed command
- Google

```
udel@CSS-SL-9HW9593: ~
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.

  Mandatory arguments to long options are mandatory for short
  options too.

  -a, --all
        do not ignore entries starting with .

  -A, --almost-all
        do not list implied . and ..

Manual page ls(1) line 1 (press h for help or q to quit)
```

man ls


```
udel@CSS-SL-9HW9593: ~$ man ls
udel@CSS-SL-9HW9593: ~$ ls --help
Usage: ls [OPTION]... [FILE]...
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.

Mandatory arguments to long options are mandatory for short options too.
-a, --all                do not ignore entries starting with .
-A, --almost-all       do not list implied . and ..
--author                with -l, print the author of each file
-b, --escape            print C-style escapes for nongraphic characters
--block-size=SIZE      with -l, scale sizes by SIZE when printing them;
                       e.g., '--block-size=M'; see SIZE format below
-B, --ignore-backups    do not list implied entries ending with ~
-c                      with -lt: sort by, and show, ctime (time of last
                       modification of file status information);
                       with -l: show ctime and sort by name;
                       otherwise: sort by ctime, newest first
-C                      list entries by columns
--color[=WHEN]         colorize the output; WHEN can be 'always' (default
                       if omitted), 'auto', or 'never'; more info below
-d, --directory        list directories themselves, not their contents
```

ls --help

```
udel@CSS-SL-9HW9593: ~
Next: What information is listed, Up: ls invocation

10.1.1 Which files are listed
-----

These options determine which files 'ls' lists information for. By
default, 'ls' lists files and the contents of any directories on the
command line, except that in directories it ignores files whose names
start with '.'.

'-a'
'--all'
    In directories, do not ignore file names that start with '.'.

'-A'
'--almost-all'
    In directories, do not ignore all file names that start with '.';
    ignore only '.' and '..'. The '--all' ('-a') option overrides this
    option.

'-B'
'--ignore-backups'
    In directories, ignore files that end with '~'. This option is
    equivalent to '--ignore='*~' --ignore='.*~'.'.

'-d'
'--directory'
    List just the names of directories, as with other types of files,
    rather than listing their contents. Do not follow symbolic links
    listed on the command line unless the '--dereference-command-line'
    ('-H'), '--dereference' ('-L'), or
    '--dereference-command-line-symlink-to-dir' options are specified.

-----Info: (coreutils)Which files are listed, 91 lines --Top-----
```

info ls

```
udel@CSS-SL-9HW9593:~$ apropos "list directory"
dir (1)          - list directory contents
ls (1)           - list directory contents
ntfsls (8)       - list directory contents on an NTFS filesystem
vdir (1)         - list directory contents
udel@CSS-SL-9HW9593:~$
```

apropos "list directory"

Work Effectively in Linux Command Line

- Linux is case-sensitive
- It is a good practice to avoid SPACES in filenames
- Tab completion
 - Automatically complete filenames, directory names, and commands

```
udel@CSS-SL-9HW9593:~/test-1$ ls
myfile.txt  Myfile.txt
udel@CSS-SL-9HW9593:~/test-1$ ls myFile.txt
ls: cannot access 'myFile.txt': No such file or directory
udel@CSS-SL-9HW9593:~/test-1$ ls myfile.txt
myfile.txt
udel@CSS-SL-9HW9593:~/test-1$
```

- Use command `history [option]`
 - i.e., `history 10`, displays the 10 previous commands in the history
 - Exclamation mark (!) followed by the command number in history to rerun the command
 - Double exclamation mark (!!) to rerun the last command

```
udel@CSS-SL-9HW9593:~$ history 10
1407 history 10
1408 clear
1409 pwd
1410 echo 'Hello World!'
1411 echo -n 'Hello World!'
1412 man ls
1413 info ls
1414 ls --help
1415 apropos "list directory"
1416 history 10
udel@CSS-SL-9HW9593:~$ !1409
pwd
/home/udel
udel@CSS-SL-9HW9593:~$ !!
pwd
/home/udel
udel@CSS-SL-9HW9593:~$
```

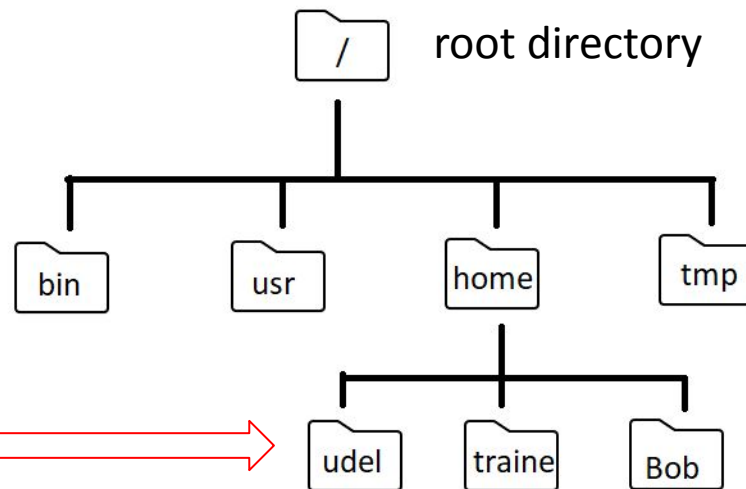
Keyboard Shortcuts

- Ctrl + a: Move to the beginning of the line.
- Ctrl + e: Move to the end of the line.
- Ctrl + c: Interrupt the current command.
- Ctrl + u: Delete text from cursor to line start.
- Ctrl + k: Delete text from cursor to line end.
- Up/Down Arrow: Recall the previous command.

Directory Structure

Linux's principle: Everything is a file

- absolute path
 - begin from / of the file system
- relative path
 - begin from the current working directory



Basic File Operations

- **mkdir/rmdir**
 - Create/delete directories
 - Syntax: `mkdir [option] <directory>`
 - '-p': create parent directory if needed, e.g., `mkdir -p train_data/data`
 - Syntax: `rmdir <directory>`, e.g., `rmdir train_data`

```
udel@CSS-SL-9HW9593:~/test-1$ mkdir dir1
udel@CSS-SL-9HW9593:~/test-1$ ls dir1/
udel@CSS-SL-9HW9593:~/test-1$ rmdir dir1/
udel@CSS-SL-9HW9593:~/test-1$ ls dir1
ls: cannot access 'dir1': No such file or directory
```



```
udel@CSS-SL-9HW9593:~$ pwd
/home/udel
udel@CSS-SL-9HW9593:~$ cd test/test-a/
udel@CSS-SL-9HW9593:~/test/test-a$ pwd
/home/udel/test/test-a
udel@CSS-SL-9HW9593:~/test/test-a$ cd ..
udel@CSS-SL-9HW9593:~/test$ pwd
/home/udel/test
udel@CSS-SL-9HW9593:~/test$ cd /home/udel/test/test-a
udel@CSS-SL-9HW9593:~/test/test-a$ pwd
/home/udel/test/test-a
udel@CSS-SL-9HW9593:~/test/test-a$ cd
udel@CSS-SL-9HW9593:~$ pwd
/home/udel
```

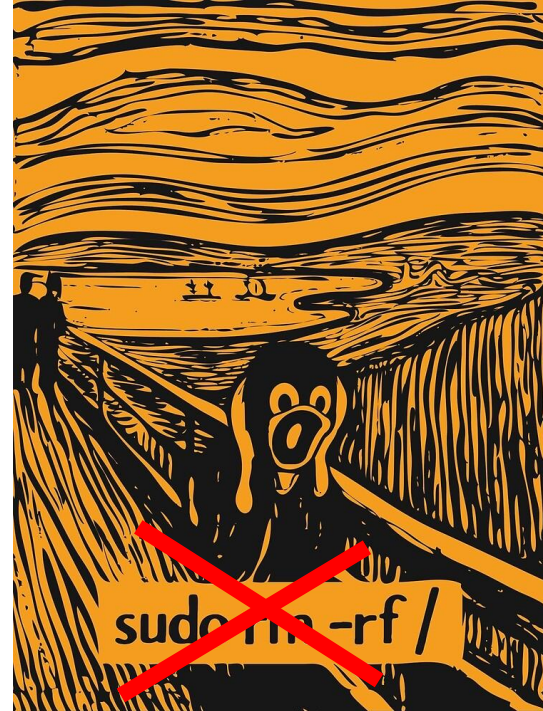
relative path

absolute path

- **cd**
 - Change directories
 - Syntax: `cd <directory>`, e.g., `cd /home/udel`

- **cp**
 - Copy files and directories
 - Syntax: `cp [option] <source> <destination>`
 - e.g., `cp sample.txt sample_dir`
 - '-a': copy files recursively and preserve symbolic links and date/time stamps
 - e.g., `cp -a <source> <destination>`
- **mv**
 - Move files/directories to a new location
 - Syntax: `mv [option] <file1> <destination>`
 - e.g., `mv sample.txt /home/udel`
 - Rename files/directories
 - e.g., `mv sample.txt sample_1.txt`

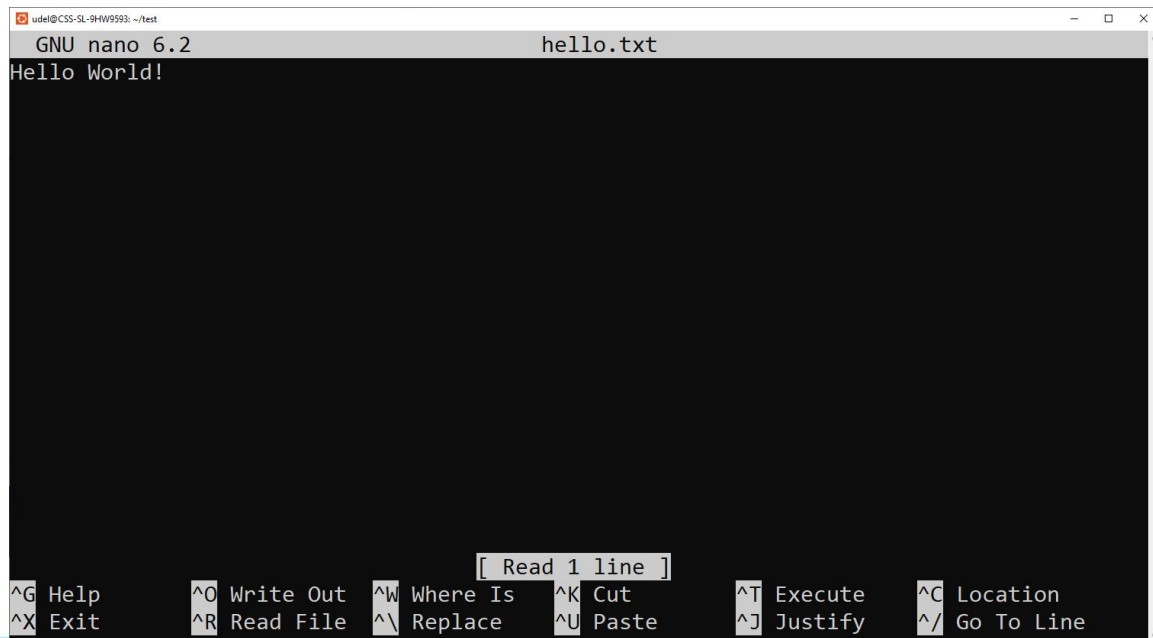
- **rm**
 - Removes files or directories, be caution to use
 - Syntax: `rm [option] <file1>`
 - e.g., `rm sample.txt`
 - '-i': add interactive flag for confirmation
 - '-r': remove file recursively, can be dangerous
 - e.g., use with '-i' option, `rm -ri <dir1>`



<http://www.redbubble.com/people/uman>

Text Editor: nano

- `nano <filename>`
 - Ctrl + x: exit
 - Ctrl + g: open the help menu
 - Ctrl + w: search the word
 - Ctrl + k: cut the text
 - Ctrl + u: paste the text



The screenshot shows a terminal window with the GNU nano 6.2 text editor. The window title is 'udel@CSS-SL-9HW9593: ~/test' and the file being edited is 'hello.txt'. The text 'Hello World!' is visible in the editor. At the bottom, a status bar shows the command '[Read 1 line]' and a list of keyboard shortcuts: ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Location, ^X Exit, ^R Read File, ^\ Replace, ^U Paste, ^J Justify, and ^_ Go To Line.

Wildcards

- *(asterisk)
 - Represent any number of characters
 - e.g., `ls a*.txt`
- ?(question mark)
 - Represent any single character, e.g., `ls a?.txt`
- [](square brackets)
 - Represent any single character within the specified range or set
 - e.g., `ls a[a-e].txt`
- {}(curly brackets)
 - Used for multiple matches, e.g., `mv file{1,2,3}.txt dir1/`

```
udel@CSS-SL-9HW9593:~/test-1$ touch {a,b}{1,2}.txt
udel@CSS-SL-9HW9593:~/test-1$ ls
a1.txt  a2.txt  b1.txt  b2.txt
udel@CSS-SL-9HW9593:~/test-1$ ls a?.txt
a1.txt  a2.txt
udel@CSS-SL-9HW9593:~/test-1$ ls *.txt
a1.txt  a2.txt  b1.txt  b2.txt
udel@CSS-SL-9HW9593:~/test-1$ ls b[1-2].txt
b1.txt  b2.txt
```

Input/Output Redirection

- > (Output Redirection)
 - Redirects standard output to a file
 - e.g., `ls > output.txt`
- < (Input Redirection)
 - Redirects standard input from a file
 - e.g., `sort < input.txt`
- >> (Append Output)
 - Appends standard output to a file
 - e.g., `ls >> output.txt`


```
udel@CSS-SL-9HW9593: ~/test/demo-1
udel@CSS-SL-9HW9593:~/test/demo-1$ ls > list.txt
udel@CSS-SL-9HW9593:~/test/demo-1$ cat list.txt
list.txt
sample1.txt
sample2.txt
sample3.txt
udel@CSS-SL-9HW9593:~/test/demo-1$ echo "Hello!" >> list.txt
udel@CSS-SL-9HW9593:~/test/demo-1$ cat list.txt
list.txt
sample1.txt
sample2.txt
sample3.txt
Hello!
udel@CSS-SL-9HW9593:~/test/demo-1$ sort < list.txt
Hello!
list.txt
sample1.txt
sample2.txt
sample3.txt
udel@CSS-SL-9HW9593:~/test/demo-1$
```

Input/Output Redirection

- 2> (Error Redirection)
 - Redirects the standard error
 - e.g., `ls file1.txt 2> error.txt`

```
udel@CSS-SL-9HW9593:~/test-1$ ls file1.txt
ls: cannot access 'file1.txt': No such file or directory
udel@CSS-SL-9HW9593:~/test-1$ ls file1.txt > error.txt
ls: cannot access 'file1.txt': No such file or directory
udel@CSS-SL-9HW9593:~/test-1$ ls file1.txt 2> error.txt
udel@CSS-SL-9HW9593:~/test-1$ cat error.txt
ls: cannot access 'file1.txt': No such file or directory
```

```
udel@CSS-SL-9HW9593:~/test-1$ cat fruits.txt
Apple
Orange
Cherry
Peach
Lemon
Watermelon
Strawberry
Banana
Apple
udel@CSS-SL-9HW9593:~/test-1$ cat fruits.txt | grep "Apple" | wc -l
2
```

search for words

count the number of line

- **ls**
 - List files and directories
 - '`-l`': Long format, providing detailed information about each file or directory
 - '`-a`': Lists all files, including hidden ones (starting with a dot)
 - '`-h`': Human-readable format, displaying file sizes in a human-readable format

```
udel@CSS-SL-9HW9593:~/demo$ ls -lah
total 16K
drwxr-xr-x  3 udel udel 4.0K Apr  1 11:55 .
drwxr-x--- 19 udel udel 4.0K Apr  1 11:56 ..
drwxr-xr-x  2 udel udel 4.0K Apr  1 11:55 dir1
-rw-r--r--  1 udel udel   52 Apr  1 11:54 list.txt
lrwxrwxrwx  1 udel udel    8 Apr  1 11:55 sym_link -> list.txt
```

File Permissions

Group that owns the file, directory, et al.

Last modification timestamp

Target path for symbolic link

```
drwxr-xr-x 2 user group 4.0K Feb 20 11:40 directory_1
-rw-r--r-- 1 user group 52 Feb 22 11:40 Linux.txt
lrwxrwxrwx 1 user group 8 Feb 22 13:09 symbolic_link ->Linux.txt
```

Name of file, directory, et al.

Size of file, directory, et al. (in bytes here)

User that owns the file, directory, et al.

Permissions are indicated as:

[r]ead: see the contents of the item

[w]rite: modify the contents of the item

e[x]ecute:

a regular file is treated as an executable program

a directory can be entered with the 'cd' command

[d]: directory

[-]: regular file

[l]: symbolic link

Permissions for all users on the system

Permissions for the group that owns the file

Permissions for the user that owns the file

Manage File Permissions

- **chmod**

- Syntax: `chmod [permission] <file>`
- 'u' for users, 'g' for groups, 'o' for others, 'a' for all (u, g, and o)
- Modify permissions using '+' (add), '-' (remove), '=' (set)
- Absolute form: 4 for read, 2 for write, 1 for execute
- e.g., `chmod u+x file.sh` ← adds execute permission for the user (owner)
- e.g., `chmod 755 file.txt` ← the user (owner) can do anything, others can only read and execute

Basic Data Transfer

- **scp (Secure Copy):**
 - `scp <local_file>`
`remote_user@remote_host:/remote/path/`
 - `scp remote_user@remote_host:/path/to/file`
`/local/path/`
- **wget/curl**
 - Download files from internet
 - Syntax: `wget -O [file_name] <link_address_for_archive>`
 - Syntax: `curl -o [file_name] <link_address_for_archive>`


```
udel@CSS-SL-9HW9593: ~/demo/dir1
udel@CSS-SL-9HW9593:~/demo/dir1$ wget -O iris-1.zip https://archive.ics.uci.edu/static/public/53/iris.zip
--2024-04-05 15:24:35-- https://archive.ics.uci.edu/static/public/53/iris.zip
Resolving archive.ics.uci.edu (archive.ics.uci.edu)... 128.195.10.252
Connecting to archive.ics.uci.edu (archive.ics.uci.edu)|128.195.10.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified
Saving to: 'iris-1.zip'

iris-1.zip          [ <=> ] 3.65K  --.-KB/s  in 0s

2024-04-05 15:24:35 (158 MB/s) - 'iris-1.zip' saved [3738]

udel@CSS-SL-9HW9593:~/demo/dir1$ curl -o iris-2.zip https://archive.ics.uci.edu/static/public/53/iris.zip
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
100 3738    0 3738    0     0 13757      0  --:--:--  --:--:--  --:--:-- 13793

udel@CSS-SL-9HW9593:~/demo/dir1$ ls
iris-1.zip  iris-2.zip

udel@CSS-SL-9HW9593:~/demo/dir1$ scp iris-1.zip traine@caviness.hpc.udel.edu:/home/trainee
```

upload local file to remote server

```
udel@CSS-SL-9HW9593:~/demo/dir1$ scp traine@caviness.hpc.udel.edu:/home/trainee/iris.zip /home/udel
```

download file from remote server to local

Compress and Decompress Files

- **zip/unzip**

- Syntax: `zip [option] <file.zip> <file>`
- Syntax: `unzip <file.zip>`
- Use '-r' to compress the directory, e.g., `zip -r file.zip dir1/`
- Extract the zip file to a directory, e.g., `unzip file.zip -d dir1/`

- **tar**

- Syntax: `tar [option] <archive> <file>`
- '-c': create archive file, '-x': extract archive file
- '-f': specify the name of archive file, '-v' print verbose information
- '-t': list contents in archive file
- '-z': use gzip compression, for .tar.gz archive

Compress and Decompress Files

```
udel@CSS-SL-9HW9593:~/test-1/dir1$ tar xf iris.tar.gz
udel@CSS-SL-9HW9593:~/test-1/dir1$ ls
iris  iris.tar.gz
udel@CSS-SL-9HW9593:~/test-1/dir1$ tar zcf iris-data.tar.gz iris/
udel@CSS-SL-9HW9593:~/test-1/dir1$ ls
iris  iris-data.tar.gz  iris.tar.gz
udel@CSS-SL-9HW9593:~/test-1/dir1$
```

– Extract the zip file to a directory, e.g., `unzip FILE.zip -d DIR1/`

- **tar**
 - Syntax: `tar [option] <archive> <file>`
 - '-c': create archive file, '-x': extract archive file
 - '-f': specify the name of archive file, '-v' print verbose information
 - '-t': list contents in archive file
 - '-z': use gzip compression, for .tar.gz archive

Environment Variables and PATH

- Environment variables are in uppercase letters
 - HOME: user's home directory
 - USER: username of the current user
 - PATH: Specifies directories where executable files are located
 - SHELL: shell for current user
- To display the value of an environment variable, use
 - `echo $VARIABLE_NAME` or `printenv VARIABLE_NAME`
- To set an environment variable, use: `export VARIABLE_NAME=value`
- To add to the PATH: `export PATH="$PATH:/path/to/file"`

Shell Scripts

- Series of commands in a script file
- Bash script has file extension of `.sh` by conventions
- Starts with a shebang: `#!/bin/bash`
- Define variables: `variable_name=value`
- To get the value of variables: `variable_name`
- Comments begin with `#`
- Don't forget to change the permissions, e.g., `chmod +x example.sh`

For loops and If statements

```
for variable in {range}; do
    statement
done
```

```
if [ condition 1 ]
then
    statement 1
elif [ condition 2 ]
then
    statement 2
else
    statement 3
fi
```


Need Help?

The Unix Shell

<https://swcarpentry.github.io/shell-novice/>

The Linux Command Line for Beginners

<https://ubuntu.com/tutorials/command-line-for-beginners#1-overview>

Linux/Unix Tutorial

<https://www.geeksforgeeks.org/linux-tutorial/>

HPC wiki

<https://docs.hpc.udel.edu/>

Google group: hpc-ask

Research Computing Help Request



Thank you!

For more information, contact:

askit@udel.edu