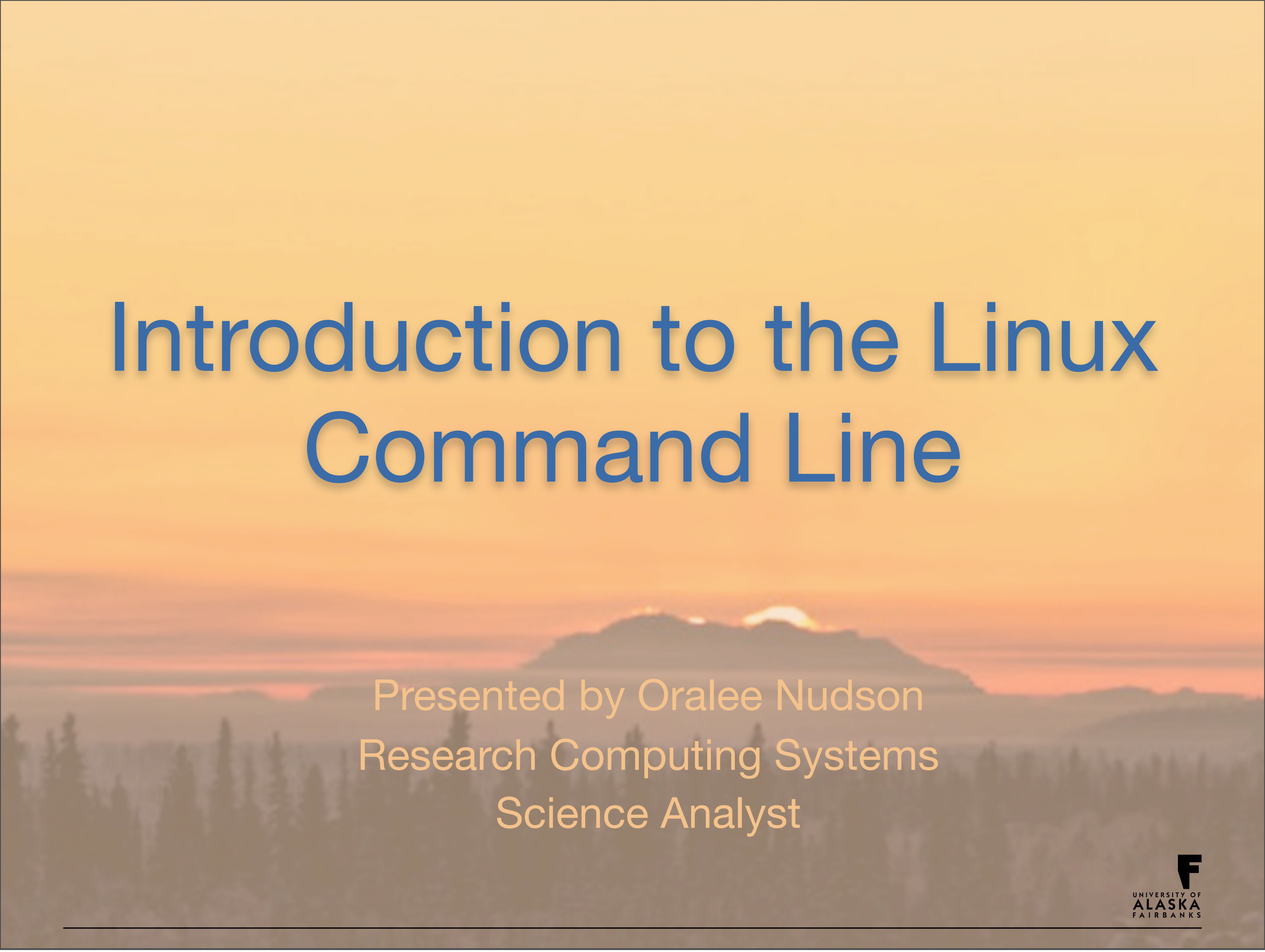

Introduction to the Linux Command Line

Presented by Oralee Nudson
Research Computing Systems
Science Analyst

UNIVERSITY OF ALASKA FAIRBANKS

Overview

- Connecting to Remote Systems
- Working with Files
- File and Directory Permissions
- Modules
- Intro to Linux Command Line Training

Linux Overview

- Unix-like OS developed by Linus Torvalds in 1991
- Open Source Software
- “Runs on more computer hardware platforms than any other OS” (wikipedia.org)
- Runs on Supercomputers, …, embedded systems
- The shell is a command line interface to the OS
  - ○ Open a “terminal” window
  - ○ Edit files
  - ○ Launch processes or jobs
  - ○ Check the status of running processes
  - ○ Send signals to processes
  - ○ Common shells: bash, ksh, tcsh, csh

Open a Terminal

- On the mac, click on the Finder icon, select "Applications" then "Utilities", and double click on "Terminal".

Navigating the File System

- Linux is a collection of files and directories (think of folders)
- The top directory is called the “root”.
- Some directories contain actual files, others provide access to hardware devices

- Common Commands:
  - pwd
  - cd
  - ls
  - ls –al
  - ls $HOME
  - mkdir
  - rm filename
  - rmdir
  - rm –rf directoryName

Connecting to Remote Systems

- RCS supports **ssh**, **sftp** and **scp** clients
- Linux and most operating systems come with native command line versions of **ssh**, **sftp** and **scp**.
- Windows requires downloading a terminal and file transfer program (and optionally an X11 server). RCS has the most experience supporting PuTTY, FileZilla, and Xming.
- Login with:
  - ssh –X –Y username@systemname.arsc.edu
  - Example: ssh –X –Y nudson@pacman3.arsc.edu
- Copy files with:
  - scp myfiles.tar.gz username@systemname.arsc.edu:~/phys608/
  - Example: scp myfiles.tar.gz nudson@pacman4.arsc.edu:~/phys608
  - Use a GUI: filezilla, fetch, winscp
- Set up ssh keys, see "`news pubkeys`"

Working with Files

- Common Linux Text Editors
  - vim or gvim
  - Emacs
  - nedit (X11 enabled only)
- Quickly view the contents of a file with:
  - cat
  - less
    - Exit with “q”
- Documentation for shell commands
  - “man” pages
  - Info
- View images with the “`display`” command

File and Directory Permissions

- Permissions control access to files and directories
  - Three categories of access:
    - user
    - group (type “`groups`” to determine which you belong to)
    - other
  - Three categories of permissions:
    - read
    - write
    - execute
  - Use “`chmod`” to modify access permissions
    - chmod u+r myDir (add read permissions for myself)
    - chmod g+rx myFile (add group read & execute permissions)
    - chmod go-rwx myFile (remove group and other permissions)

File and Directory Permissions

- Copy files to/from another user with scp:
  - scp username@pacman10:/scratch/ontester/introToLinux/introToLinux_fall2015.tar .

- Security Awareness:
  - World write permissions are discouraged.
  - Never share your login credentials (username & password) with others.
  - What else?

UNIVERSITY OF ALASKA FAIRBANKS

File Input/Output & Redirection

- Three forms of input/output:
  - “stdin” from keyboard or a file
  - “stdout” to screen or a file
  - “stderr” to screen or a file
- Redirect I/O with
  - Greater/Less Than Symbols, “>” or “>>” or “<“
  - Pipes, “|”
- Tie stdout and stderr together with “2>&1”
  - # In bash:
  - mpirun $CENTER/wrf.exe > wrf.mix.out 2>&1 &!

Special Shell Characters

- “*” matches anything
- “?” matches a single character
- “&” backgrounds a running process
  - View the process status with “ps”
  - Bring process back to foreground with “fg”
  - Try with CTRL+Z

Working with Active Processes

- “ps” allows you to view process statuses
    - Useful variations “ps –elf” and “ps –aux”
- “top” to view what’s eating up all the CPU resources!
    - Exit with “q”
- Send a signal:
    - CTRL+c (kill)
    - CTRL+z (suspend)
- Search with “grep”, then “sort”

Working with Active Processes

Try it!

```
$ sleep 1000
$ ctrl-z
$ ps
$ fg
$ ctrl-c
$ sleep 1000 &
$ ps
$ fg
```

Working with Active Processes

Try it!

```
# edit a new file called sleepyTime.sh
containing:
    #!/bin/bash
    echo “hello there.  I’m tired...”
    sleep 1005
    exit
$ chmod 700 sleepyTime.sh
$ ./sleepyTime.sh
```

Common Linux Commands

- “kill” to terminate processes
- “man kill”
- Send particular signals, e.g. “kill –KILL 3039”
- Try it!
  - ○ Sleep 2000
  - ○ ps
  - ○ kill <pid>
  - ○ ps

Customizing the User Environment

- Environment Variables store short strings of information
- Important variables: $PATH, $HOME, $CENTER
- The shell auto-expands variables
- Set with
  - bash: export CHUBBY_BUNNIES=funny
  - bash: export PATH=${PATH}:/u1/uaf/nudson/bin
  - csh/tcsh: setenv CHUBBY_BUNNIES funny
  - csh/tcsh: setenv PATH ${PATH}:/u1/uaf/nudson/bin
- View with echo $CHUBBY_BUNNIES

UNIVERSITY OF ALASKA FAIRBANKS

User Environment

- Customize your login by modifying your $HOME "." files
- Example for bash users:
  - Add the following to your ~/.profile file:

    export PS1="Good Morning!% "
  - Then source the file with ". ~/.profile"

Questions, Comments?

Oralee Nudson
onudson@alaska.edu
907-450-8637

Please complete the 3 minute Training Evaluation Form:

http://www.arsc.edu/support/training/ trainingevalform/ index.xml