Tech Mahindra

# Unix Fundamentals & Commands

Day 2

TLS

Rewards and Recognition

2012 ASTD BEST AWARD WINNER

training TOP 125 Winner - 2013

Tata Institute of Social Sciences CLO Awards India Best Virtual Learning Award

# Objectives

- At the end of the session, you will be able to:
  - Use general purpose Unix commands

Copyright © 2015 Tech Mahindra. All rights reserved.

# Agenda: Day 2

- File comparison Commands
- File Access Permission Commands
- Pipes & Filters
- Shell Basics
- Unix Environment

Copyright © 2015 Tech Mahindra. All rights reserved.

# File Comparison

- cmp command compares two files and reports the location of first mismatch

```
$ cmp file1 file2
file1 file2 differ: byte 1, line 1
```

Copyright © 2015 Tech Mahindra. All rights reserved.

# File Comparison

- comm compares two sorted files and produces result in three columns:
  Lines: Unique to file1  Unique to file2  Common

```
$ comm file1 file2
                Anil 12 2000
Ajit 12 2000
        Dinesh 27 1400
Sunil 12 2000
```

Copyright © 2015 Tech Mahindra. All rights reserved.

# File Comparison

- diff command compares two files and proposes the changes in first file to make two files identical

| | | |
|---|---|---|
| `$ cat file1`<br>`Anil 12 2000`<br>`Ajit 12 2000`<br>`Sunil 12 2000` | `$ cat file2`<br>`Anil 12 2000`<br>`Dinesh 27 1400` | `$diff file1 file2`<br>`2,3c2`<br>`< Ajit 12 2000`<br>`< Sunil 12 2000`<br>`---`<br>`> Dinesh 27 1400` |

Copyright © 2015 Tech Mahindra. All rights reserved.

# File Access Permissions

- chmod command is used to change the permissions on a file for owner, group and others
  chmod <permission><filename>

- Symbolic Method:

| Code | Meaning |
|---|---|
| a | all |
| u | user |
| g | group |
| o | other |
| + | add |
| - | remove |
| = | assign |

chmod u+x,g-w,o+r,o-wx sample

Copyright © 2015 Tech Mahindra. All rights reserved.

# File Access Permissions

- Absolute value Method:

| Code | Meaning |
|---|---|
| 4 | Read |
| 2 | Write |
| 1 | Execute |

- $ chmod 754 sample

Copyright © 2015 Tech Mahindra. All rights reserved.

# File Access Permissions

- umask command changes initial permission of newly created file
- The value of argument can be calculated by subtracting the mode you want as default from the current default mode
- Assume that the current default mode is 0666 and you want it as 0644 then 666 – 644 = 022 will be the parameter which we have to pass with “umask” command
- $umask 0 – sets default mode which is 0666

Copyright © 2015 Tech Mahindra. All rights reserved.

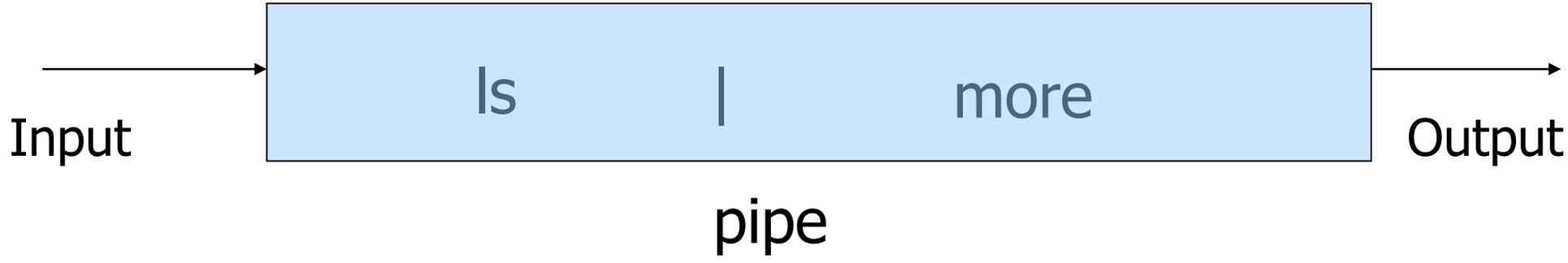
# Pipes

- It is used to pass the output of one Unix command as an input to another Unix command

  e.g. $ ls >temp

  $ more temp

$ ls | more




Copyright © 2015 Tech Mahindra. All rights reserved.

# Text Manipulation Commands: Filters

- A filter is a SHELL command which takes an input ( either from a standard input device or a file ) , processes it and send the processed out put to a standard output device or a file

- At run time, the system supplies data to the filter as standard input. This standard input file can not be altered by the program.

Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- wc: counts lines, words and characters

  $ wc -[wlc] [filename]

- head: displays first ‘n’ lines, horizontal slicing

  $ head -[n] [filename]

- tail: displays last ‘n’ lines, horizontal slicing

  $ tail -[n] [filename]

- split: divides files horizontally

  $ split -[l] [filename]

Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- cut: cuts file vertically either column wise/field wise

- $ cut -[cfd] [filename]
  - -c  columns/characters
  - -f  field number
  - -d  field delimiter/separator

- $ cut -c2-5 sample
  - cuts columns 2 to 5 from the file sample

Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- $ cut -c1,2 file
  - Cuts 1st and 2nd characters from file
- $ cut  -d” “ -f1-2 file
  - Cuts first 2 fields of the file
- $ cut -c2- names
  - Starts cutting from 2nd char to the end of line

Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- paste: merges lines of specified files and display onto standard output
  - $ paste -d[field separator] [list of files]

- sort: ordering text files
  - $ sort filename

e.g. $ sort +3 -4 emp

Sorts emp file as per 4th column

$ sort –t”:” +3 -4 emp

Sorts emp file as per 4th field

$ sort –t”:” +3.3 -4.5 file

Sorts emp file as per 4th and 5th columns of 4th field

Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- uniq: removes duplicate lines from a sorted file
  - $ uniq –[dcf] [file]

  -d  only print duplicate lines
  -c  prefix lines with number of occurences
  -f2  avoid comparing first two fields

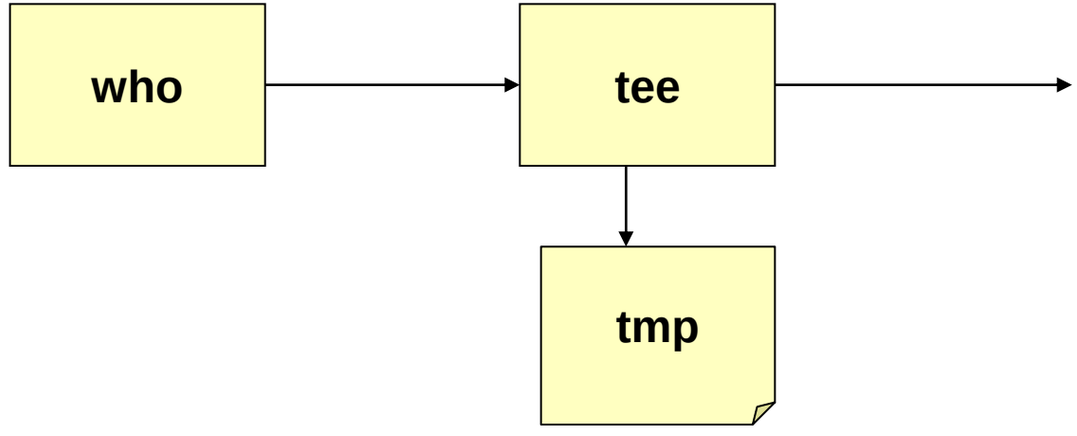
- nl: Displays file contents along with line nos.
  - $ nl –[options] [files]

Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- tee: takes input from some command and generates two outputs. One is redirected to a file and other to standard output or next command

  e.g. $who | tee tmp




Copyright © 2015 Tech Mahindra. All rights reserved.

# Filters

- tr: translating Characters

  $ tr  [options] < [file]

  -d: deletes specified characters

  -cd: do not delete specified characters

  -s: substitute multiple occurrences of a character by single occurrence.

- $ tr “abc” “ABC” < samp: replaces all occurrences of a with A, b with B, c with C

Copyright © 2015 Tech Mahindra. All rights reserved.

# Command Substitution

- One command can be used as an argument to another command

  - $echo The date today is `date`
  - $echo “There are `ls |wc –l` files”

Copyright © 2015 Tech Mahindra. All rights reserved.

# UNIX Shell

TLS

Rewards and Recognition




Copyright © 2015 Tech Mahindra. All rights reserved.

# UNIX Shells

- Bourne Shell
- Korn Shell
- C Shell
- tc Shell
- Bourne Again Shell

Copyright © 2015 Tech Mahindra. All rights reserved.

# Bourne Shell

- Original UNIX shell program
- Widely used, written by Steve Bourne
- Start Bourne Shell by typing “sh” or “/bin/sh” in the command prompt
  - This does not spawn a new shell window
  - This just changes the current shell to Bourne shell
- Supports conditional branching in the form of if / then / else statements
- Supports case statements and loops (for, while, until)
- Uses “$” as the prompt

Copyright © 2015 Tech Mahindra. All rights reserved.

# Korn Shell

- A newer variation of Bourne shell
- Originally written by David Korn; copyrighted by AT&T
- Programming structure similar to Bourne, but more interactive
- Adds features not available in Bourne shell
- Korn shell not a standard UNIX offering; type ksh or /bin/ksh at the shell prompt
- A public domain version of Korn shell – pdksh

Copyright © 2015 Tech Mahindra. All rights reserved.

# C Shell

- A commonly used shell
- Programming structure very similar to the C programming language
- Uses “%” as prompt
- Supports all features of Bourne shell; more natural syntax for programming
- More interactive than Bourne shell, with additional features that were not there in older shells
- Configuration controlled by the .rc and .login files

Copyright © 2015 Tech Mahindra. All rights reserved.

# tc Shell

- A modern variation of the C Shell
- Reads the same configuration files as C Shell
- tcsh contains command line editing keystrokes that the C Shell doesn’t
- *Modern* conveniences that the C Shell lacks

Copyright © 2015 Tech Mahindra. All rights reserved.

# Bourne-Again Shell

- A variation of Bourne Shell
- Commonly used in Linux; widely available in other UNIX distributions
- Uses “$” as the prompt
- Type bash at the shell prompt to start Bourne-Again shell
- Behavior and environment controlled by the .bashrc file, which is a hidden file in the Home directory

Copyright © 2015 Tech Mahindra. All rights reserved.

# UNIX Environment

TLS

Rewards and Recognition




Copyright © 2015 Tech Mahindra. All rights reserved.

# Environment Variables

- Variables set by the system to control the UNIX system – some during the boot sequence, and some after logging in
- Also called System Variables
- The set statement displays the complete list of System Variables
- Built-in variable names are defined in Upper Case

Copyright © 2015 Tech Mahindra. All rights reserved.

# Environment Variables (Contd..)

- PATH: instructs the Shell about the route it should follow to locate any executable command
- HOME: On login, UNIX normally places the user in a directory named after the login name
- MAIL: determines where all incoming mails addressed to a user are to be stored
- PS1 and PS2: PS1 is a command prompt and PS2 is a multi-line command string
- SHELL: determines the type of shell that a user sees on logging in

Note: There are other environment variables also. We have discussed only the important ones here

Copyright © 2015 Tech Mahindra. All rights reserved.

# Profile

- .profile (.bash_profile in Linux): the script executed during login time
- The .profile must be located in your home directory, and it is executed after /etc/profile, the universal profile for all users
- Universal environment settings are kept by the administrator in /etc/profile so that they are available to all users

Copyright © 2015 Tech Mahindra. All rights reserved.

# Setting Environment Variables

- Assigning value to a variable sets an environment variable temporarily
- For example:

  $ x = 50

  This sets the value 50 to variable x

Copyright © 2015 Tech Mahindra. All rights reserved.

# Setting Environment Variables (Contd..)

- Variables are lost when the user exits the Shell
- To set variables permanently, add them in configuration files (as discussed earlier)
- unset command erases the value of a variable
- For example:

  $ unset x

  This causes variable x to have no set value

Copyright © 2015 Tech Mahindra. All rights reserved.

# Export

- By default, the values stored in shell variables are local to the shell
  - Values of variables not available in a child-shell
- Shells can export these variable values recursively to all child processes, so that once defined, they are available globally
- Use the export command for this

Copyright © 2015 Tech Mahindra. All rights reserved.

# Export (Contd..)

```
$ x = hello
$ export x
$ sh
$ echo $x
hello
```

Copyright © 2015 Tech Mahindra. All rights reserved.

# Summary

- In this session, we have covered:
  - File comparison Commands
  - File Access Permission Commands
  - Pipes & Filters
  - Shell Basics
  - Unix Environment

Copyright © 2015 Tech Mahindra. All rights reserved.

# Unix Fundamentals & Commands

## Thank You

**Disclaimer**

Tech Mahindra Limited, herein referred to as TechM provide a wide array of presentations and reports, with the contributions of various professionals. These presentations and reports are for informational purposes and private circulation only and do not constitute an offer to buy or sell any securities mentioned therein. They do not purport to be a complete description of the markets conditions or developments referred to in the material. While utmost care has been taken in preparing the above, we claim no responsibility for their accuracy. We shall not be liable for any direct or indirect losses arising from the use thereof and the viewers are requested to use the information contained herein at their own risk. These presentations and reports should not be reproduced, re-circulated, published in any media, website or otherwise, in any form or manner, in part or as a whole, without the express consent in writing of TechM or its subsidiaries. Any unauthorized use, disclosure or public dissemination of information contained herein is prohibited. Unless specifically noted, TechM is not responsible for the content of these presentations and/or the opinions of the presenters. Individual situations and local practices and standards may vary, so viewers and others utilizing information contained within a presentation are free to adopt differing standards and approaches as they see fit. You may not repackage or sell the presentation. Products and names mentioned in materials or presentations are the property of their respective owners and the mention of them does not constitute an endorsement by TechM. Information contained in a presentation hosted or promoted by TechM is provided “as is” without warranty of any kind, either expressed or implied, including any warranty of merchantability or fitness for a particular purpose. TechM assumes no liability or responsibility for the contents of a presentation or the opinions expressed by the presenters. All expressions of opinion are subject to change without notice.

Rewards and Recognition

2012 ASTD BEST AWARD WINNER

Training TOP 125 Winner - 2013

Tata Institute of Social Sciences CLO Awards India Best Virtual Learning Award


Tech
Mahindra