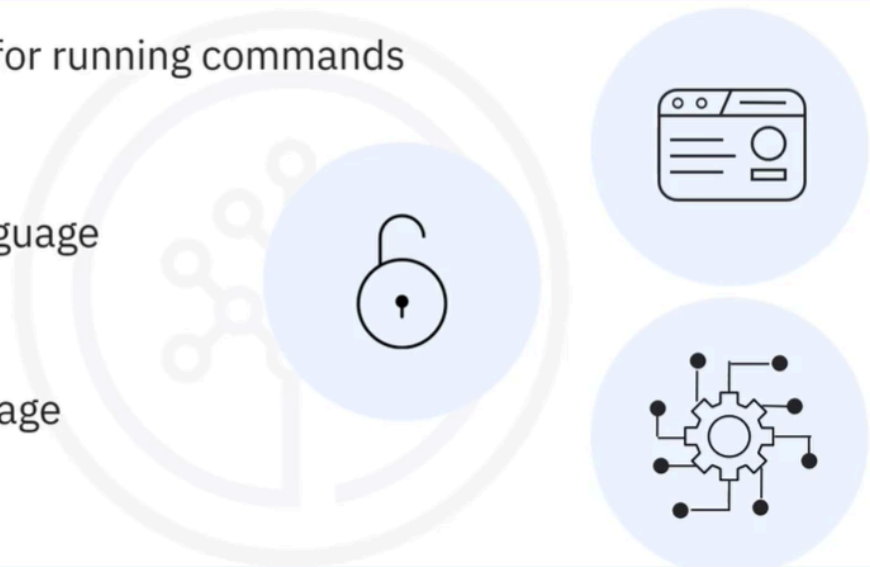


# Overview of Common Linux Shell Commands

## What is a shell?

---

- User interface for running commands
- Interactive language
- Scripting language



- A shell is a powerful user interface for Unix-like operating systems. It can interpret commands and run other programs. A shell, which enables access to files, utilities, and applications, is also an interactive language. A shell is also a scripting language. And it can be used to automate tasks.
- A shell is like a friendly translator between you and your computer. Imagine you want to ask a waiter for a dish at a restaurant. You tell the waiter what you want, and they communicate your order to the kitchen. Similarly, when you type commands into the shell, it interprets those commands and tells the computer what to do. The shell allows you to access files, run programs, and even automate tasks, making it a powerful tool for interacting with your operating system.
  - For example, if you want to see what files are in a folder, you can use the command `ls`, and the shell will show you a list of those files, just like the

waiter bringing you a menu. This makes it easier to navigate and manage your computer's resources.

## A sea of shells

---

- Default shell is usually Bash
- Many other shells, including sh, ksh, tcsh, zsh, and fish
- This course uses Bash

```
$ printenv SHELL
/bin/bash
$ bash
```

```
$
```

```
>
```

- The default shell on Linux systems is usually Bash. Other shells include 'the Bourne shell, S H,' 'the Korn shell,' K S H, 'T C shell,' 'Z shell,' and 'fish'. For this course, we will only use the Bash shell, which stands for 'bourne again shell.' To find out what the default is, enter 'printenv SHELL' on the command line. This returns the path to the default shell program, which in this case is bash. If your default shell is not Bash, you can always switch to it, simply by entering 'bash' on the command line. We will use the dollar sign to represent the command prompt throughout this course. Outside of this course, you may encounter the 'greater than' symbol being used for the same purpose.

# Shell command applications

---

- Getting information
- Navigating and working with files and directories
- Printing file and string contents
- Compression and archiving
- Performing network operations
- Monitoring performance and status
- Running batch jobs



- Applications of shell commands include: Getting information, Navigating and working with files and directories, Printing file and string contents, File compression and archiving, Performing network operations, Monitoring performance and status of the system, its components and applications, and Running batch jobs, such as ETL operations.

## Getting information

---

Some common shell commands for getting information include:

- `whoami` – username
- `id` – user ID and group ID
- `uname` – operating system name
- `ps` – running processes
- `top` – resource usage
- `df` – mounted file systems
- `man` – reference manual
- `date` – today's date

- Some common shell commands for getting information include: `whoami` – which returns the user's username, `id` – which returns the current user, and group IDs, `uname` – returns the operating system name, `ps` – displays running processes and their IDs, `top` – displays running processes and resource usage including memory, CPU, and IO, `df` – shows information about mounted file systems, `man` – fetches the reference manual for any shell command, and `date` – prints today's date.

## Working with files

---

Some common shell commands for working with files include:

- `cp` – copy file
  - `mv` – change file name or path
  - `rm` – remove file
  - `touch` – create empty file, update file timestamp
  - `chmod` – change/modify file permissions
  - `wc` – get count of lines, words, characters in file
  - `grep` – return lines in file matching pattern
- 
- Some common shell commands for working with files include: `cp` – copy file, `mv` – change file name or path, `rm` – remove file, `touch` – create empty file, update file timestamp, `chmod` – change or modify file permissions, `wc` – get count of lines, words, characters in file, and `grep` – return lines in file matching pattern.

## Navigating and working with directories

---

Common shell commands for navigating and working with directories include:

- `ls` – list files and directories
  - `find` – find files in directory tree
  - `pwd` – get present working directory
  - `mkdir` – make directory
  - `cd` – change directory
  - `rmdir` – remove directory
- 
- Very common shell commands for navigating and working with directories include: `LS` – lists the files and directories in the current directory, `find` – used to find files matching a pattern in the current directory tree, `pwd` – prints the current, or 'present working,' directory, `mkdir` – makes a new directory, `cd` – changes the current directory to another directory, and `rmdir` – removes an entire directory.

## Printing file and string contents

---

For printing file contents or strings, common commands include:

- `cat` – print file contents
  - `more` – print file contents page-by-page
  - `head` – print first N lines of file
  - `tail` – print last N lines of file
  - `echo` – print string or variable value
- 
- For printing file contents or strings, common commands include: `cat` – which prints the entire contents of a file, `more` – used to print file contents one page at a time, `head` – for printing just the first 'N' lines of a file, `tail` – for printing the

last 'N' lines of a file, and the very common echo command – which 'echoes' an input string by printing it. It can also 'echo' the value of a variable.

## Compression and archiving

---

Shell commands related to file compression and archiving applications include:

- `tar` – archive a set of files
  - `zip` – compress a set of files
  - `unzip` – extract files from a compressed zip archive
- 
- Shell commands related to file compression and archiving applications include: `tar` – which is used to archive a set of files, `zip` – which compresses a set of files, and `unzip` – which extracts files from a compressed or zipped archive.

## Networking

---

Networking applications include the following:

- `hostname` – print hostname
  - `ping` – send packets to URL and print response
  - `ifconfig` – display or configure system network interfaces
  - `curl` – display contents of file at a URL
  - `wget` – download file from URL
- 
- Networking applications include the following: `hostname` – prints the host name, `ping` – sends packets to a URL and prints the response, `ifconfig` –

displays or configures network interfaces on the system, curl – displays the contents of a file located at a URL, and the wget command can be used to download a file from a URL.

## Running Linux on a Windows machine

---

- Dual boot with a partition
- Install Linux on a virtual machine
- Use a Linux emulator
- Windows Subsystem for Linux (WSL)



- We should mention that if you happen to be running a Windows machine, and you want to run Linux on it, you can do so in various ways. Linux can be installed on a separate drive partition. Switching between the two operating systems requires a reboot. Alternatively, you can install Linux on a virtual machine, or you can install a Linux emulator such as CygWin, or use the Windows Subsystem for Linux, a compatibility layer for running Linux binary executables natively on Windows.

# Recap

---

In this video, you learned that:

- A shell is an interactive user interface for running commands, a scripting language, and an interactive language.
- Shell commands are used for navigating and working with files and directories.
- Shell commands can be used for file compression.
- The curl and wget commands, respectively, can be used to display and download files from URLs.
- The echo command prints string or variable values.
- The cat and tail commands are used to display file contents.