

Operating Systems

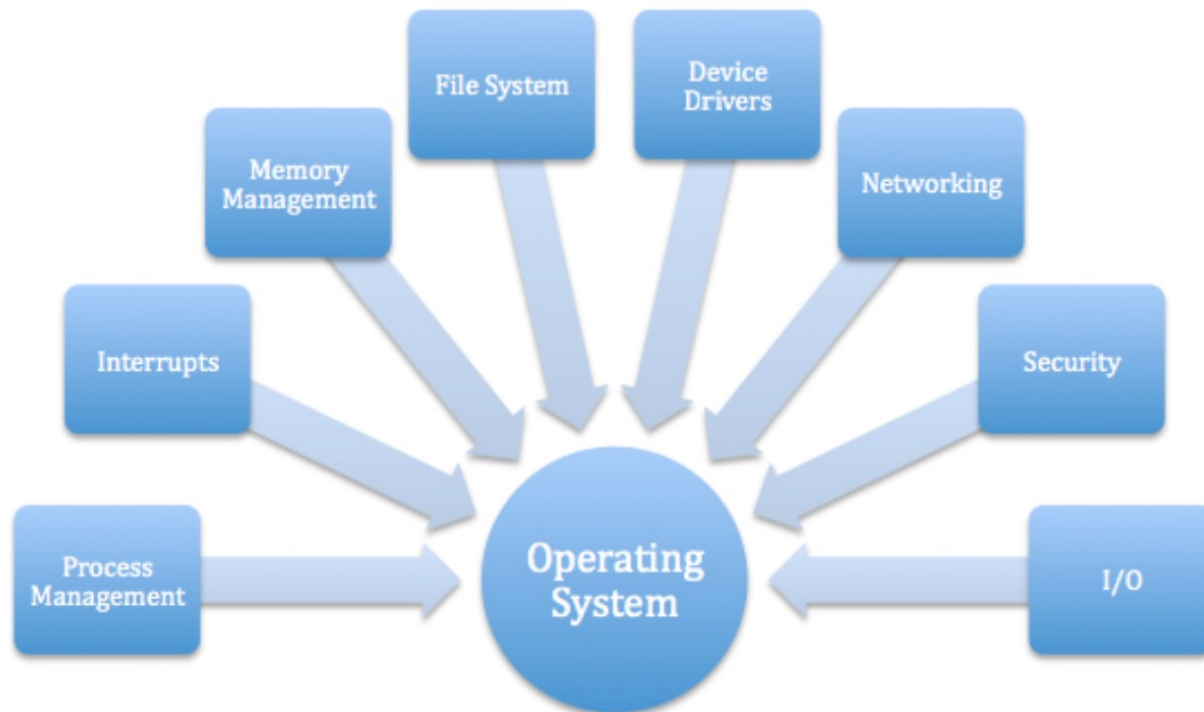
Operating Systems

- Definition: *the main program in a computer that controls the way the computer works and makes it possible for other programs to function.*
- Consists of software that:
 - Controls the use of computer hardware and software resources
 - Enables user interaction via applications
 - Gives direct access to various functions outside of applications (such as copying/deleting files)
 - Updates the OS itself



Operating System Functions

- Gives applications easy but “safe” access to hardware (safe meaning it performs the desired actions without crashing the system)
- Manages sharing of data and security
- Enables use of resources, such as memory, storage, and networking



Operating Systems

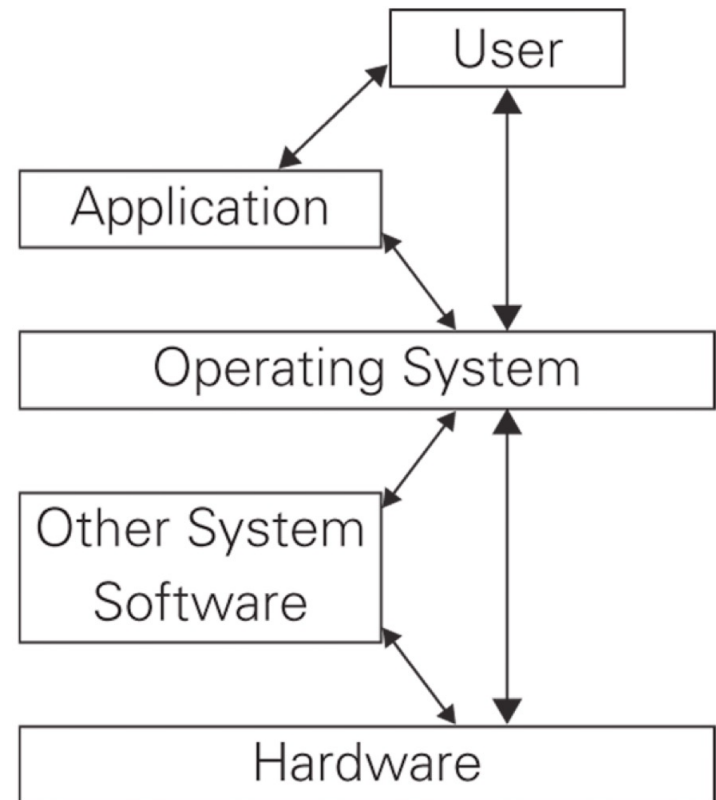
- The biggest and fastest computer in the world is of no use if it cannot efficiently provide beneficial services to its users.
- Users see the computer through their application programs. These programs are ultimately executed by computer hardware.
- System software, in the form of operating systems and middleware, is the glue that holds everything together.
- The evolution of operating systems has paralleled the evolution of computer hardware.
- As hardware became more powerful, operating systems allowed people to more easily manage the power of the machine.

Operating Systems

- In the 1960s, hardware has become powerful enough to accommodate multiprogramming, the concurrent execution of more than one task.
- Multiprogramming is achieved by allocating each process a given portion of CPU time (a timeslice).
- Interactive multiprogramming systems were called timesharing systems.
- When a process is taken from the CPU and replaced by another, we say that a context switch has occurred.

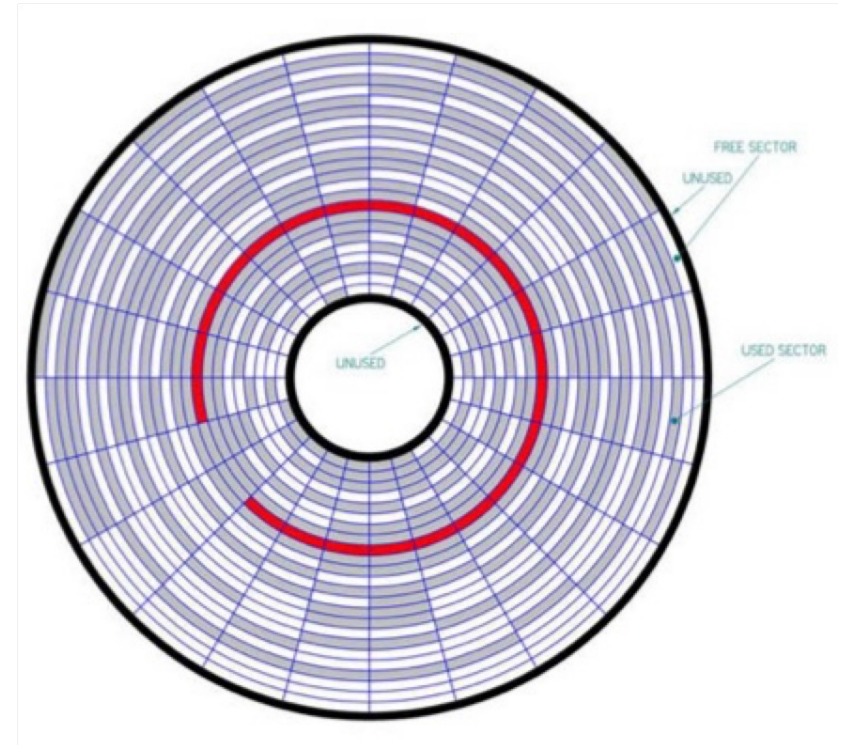
Operating Layers

- Users: input data, require steps to be executed, save data or generate output
- Application: responds to requests such as saving a file by passing it along to the OS
- OS layer: (“other system software” = e.g. device drivers - software that assists the OS)
- Hardware: the physical computer. Kernel performs/controls the actions on the hardware.



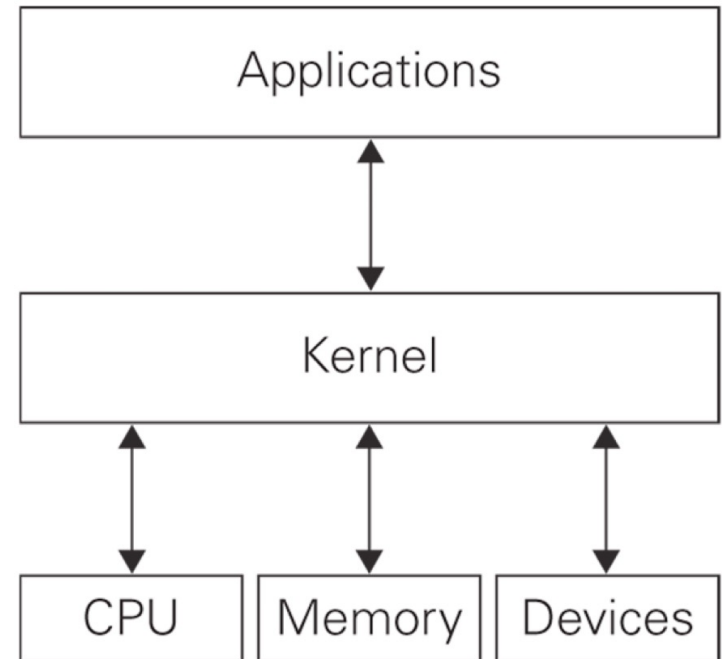
File Systems

- Different schemes of organizing and storing files on a storage device
- FAT32 (File Allocation Table)
- NTFS (New Technology File System)
- HFS+ (Hierarchical File System)
- APFS (Apple File System)
- ext4:
 - Journaling file system for Linux
 - Fourth extended filesystem



The Kernel

- Two operating system components are crucial: kernel and the system programs
- Kernel is software that oversees and exerts basic control for a computer's hardware, memory access, central processing unit (CPU), storage devices and file systems
- As the core of the operating system, the kernel performs scheduling, synchronization, memory management, interrupt handling and it provides security and protection
- Provides the necessary interface for applications to use the computer's hardware



Command Line Interface

- Operating systems have a powerful utility that allows commands to be entered and executed on the command line
- This Command Line Interface provides the user with access to operating system functions
- All major operating systems have a command line interface:
 - Windows Command Prompt / PowerShell
 - macOS Terminal
 - Unix Terminal
 - Linux Terminal

A terminal window with a dark background and light green text. The window title bar shows three window control buttons and the text '~ — pi@raspberrypi: ~ — ssh pi@raspberrypi'. The prompt is 'pi@raspberrypi:~ \$'. The command 'cat /etc/os-release' has been entered, and the output is displayed line by line: 'PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"', 'NAME="Debian GNU/Linux"', 'VERSION_ID="11"', 'VERSION="11 (bullseye)"', 'VERSION_CODENAME=bullseye', 'ID=debian', 'HOME_URL="https://www.debian.org/"', 'SUPPORT_URL="https://www.debian.org/support"', and 'BUG_REPORT_URL="https://bugs.debian.org/"'. The prompt 'pi@raspberrypi:~ \$' is shown again with a green cursor block.

```
~ — pi@raspberrypi: ~ — ssh pi@raspberrypi
pi@raspberrypi:~ $ cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
pi@raspberrypi:~ $ █
```

Command Line Interface

- Operating systems have a powerful utility that allows commands to be entered and executed on the command line
- This Command Line Interface provides the user with access to operating system functions
- All major operating systems have a command line interface:
 - Windows Command Prompt / PowerShell
 - macOS Terminal
 - Unix Terminal
 - Linux Terminal

A terminal window with a dark background and light green text. The window title bar shows three window control buttons and the text '~ — pi@raspberrypi: ~ — ssh pi@raspberrypi'. The prompt is 'pi@raspberrypi:~ \$'. The command 'cat /etc/os-release' has been entered, and the output is displayed line by line: 'PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"', 'NAME="Debian GNU/Linux"', 'VERSION_ID="11"', 'VERSION="11 (bullseye)"', 'VERSION_CODENAME=bullseye', 'ID=debian', 'HOME_URL="https://www.debian.org/"', 'SUPPORT_URL="https://www.debian.org/support"', and 'BUG_REPORT_URL="https://bugs.debian.org/"'. The prompt 'pi@raspberrypi:~ \$' is followed by a green cursor block.

```
pi@raspberrypi:~ $ cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
pi@raspberrypi:~ $
```

PowerShell

The screenshot shows the Microsoft Learn website for PowerShell documentation. The browser address bar displays the URL: <https://learn.microsoft.com/en-us/powershell/scripting/discover-powershell?view=powershell-7.4>. The page features a dark theme with a top navigation bar containing the Microsoft logo, 'Learn' link, and various documentation categories like 'Documentation', 'Training', 'Credentials', 'Q&A', 'Code Samples', 'Assessments', and 'Shows'. A search bar and a 'Sign in' link are also present. Below the navigation bar, a breadcrumb trail shows 'PowerShell' > 'Overview' > 'DSC' > 'PowerShellGet' > 'Utility modules' > 'Module Browser' > 'API Browser' > 'Resources'. A 'Download PowerShell' button is visible in the top right. The main content area is titled 'Discover PowerShell' and includes a 'Version' dropdown set to 'PowerShell 7.4 (LTS)'. The left sidebar contains a 'Filter by title' box and a list of topics under 'Overview', with 'Discover PowerShell' selected. The main text area explains that PowerShell is a command-line shell and scripting language, started on Windows to help automate administrative tasks. It also mentions that PowerShell is unique for accepting and returning .NET objects. Below this, a section titled 'What can PowerShell be used for?' lists three key uses: cloud management, CI/CD, and automating tasks for Active Directory and Exchange. The right sidebar, titled 'Additional resources', includes links to 'Training' (Introduction to PowerShell - Training) and 'Documentation' (What is a PowerShell command? - PowerShell, How to use the PowerShell documentation - PowerShell, What is PowerShell? - PowerShell). A 'Feedback' link is also present near the top of the main content area.

Microsoft | Learn Documentation Training Credentials Q&A Code Samples Assessments Shows

PowerShell Overview DSC PowerShellGet Utility modules Module Browser API Browser Resources

Download PowerShell

Version

PowerShell 7.4 (LTS)

Filter by title

How to use this documentation

Overview

What is PowerShell?

What is a PowerShell command?

Discover PowerShell

Install

Learning PowerShell

What's New in PowerShell

Windows PowerShell

Desired State Configuration (DSC)

PowerShell Gallery

Community

Scripting and development

Docs Contributor's Guide

PowerShell support lifecycle

Reference

Download PDF

Learn / PowerShell /

Discover PowerShell

Article • 12/18/2023 • 6 contributors

Feedback

In this article

- What can PowerShell be used for?
- Who uses PowerShell?
- PowerShell cmdlets
- Find commands with Get-Command
- Explore objects with Get-Member

PowerShell is a command-line shell and a scripting language in one. PowerShell started out on Windows to help automate administrative tasks. Now, it runs cross platform and can be used for various tasks.

The thing that makes PowerShell unique is that it accepts and returns .NET objects, rather than text. This feature makes it easier to connect different commands in a *pipeline*.

What can PowerShell be used for?

Usage of PowerShell has grown since the days when it was Windows-only. It's still used for Windows task automation, but today, you can use it for tasks like:

- Cloud management.** PowerShell can be used to manage cloud resources. For example, you can retrieve information about cloud resources, as well as update or deploy new resources.
- CI/CD.** It can also be used as part of a Continuous Integration/Continuous Deployment pipeline.
- Automate tasks for Active Directory and Exchange.** You can use it to automate almost any task on Windows like creating users in Active Directory and mailboxes in Exchange.

Additional resources

Training

Module

Introduction to PowerShell - Training

This module introduces PowerShell, a cross-platform command-line shell and scripting language built for task automation and...

Documentation

What is a PowerShell command? - PowerShell

Commands for PowerShell are known as cmdlets (pronounced command-lets)

How to use the PowerShell documentation - PowerShell

This articles explains how to use the features of this site including search filtering and version selection.

What is PowerShell? - PowerShell

This article is an introduction to the PowerShell scripting environment and its features.

Show 5 more

<https://learn.microsoft.com/en-us/powershell/scripting/discover-powershell?view=powershell-7.4>

Terminal Commands

Unix/Linux Command Reference

FOSSwire.com

File Commands	System Info
ls - directory listing ls -al - formatted listing with hidden files cd <i>dir</i> - change directory to <i>dir</i> cd - change to home pwd - show current directory mkdir <i>dir</i> - create a directory <i>dir</i> rm <i>file</i> - delete <i>file</i> rm -r <i>dir</i> - delete directory <i>dir</i> rm -f <i>file</i> - force remove <i>file</i> rm -rf <i>dir</i> - force remove directory <i>dir</i> * cp <i>file1 file2</i> - copy <i>file1</i> to <i>file2</i> cp -r <i>dir1 dir2</i> - copy <i>dir1</i> to <i>dir2</i> ; create <i>dir2</i> if it doesn't exist mv <i>file1 file2</i> - rename or move <i>file1</i> to <i>file2</i> if <i>file2</i> is an existing directory, moves <i>file1</i> into directory <i>file2</i> ln -s <i>file link</i> - create symbolic link <i>link</i> to <i>file</i> touch <i>file</i> - create or update <i>file</i> cat > <i>file</i> - places standard input into <i>file</i> more <i>file</i> - output the contents of <i>file</i> head <i>file</i> - output the first 10 lines of <i>file</i> tail <i>file</i> - output the last 10 lines of <i>file</i> tail -f <i>file</i> - output the contents of <i>file</i> as it grows, starting with the last 10 lines	date - show the current date and time cal - show this month's calendar uptime - show current uptime w - display who is online whoami - who you are logged in as finger <i>user</i> - display information about <i>user</i> uname -a - show kernel information cat /proc/cpuinfo - cpu information cat /proc/meminfo - memory information man <i>command</i> - show the manual for <i>command</i> df - show disk usage du - show directory space usage free - show memory and swap usage whereis <i>app</i> - show possible locations of <i>app</i> which <i>app</i> - show which <i>app</i> will be run by default
Process Management	Compression
ps - display your currently active processes top - display all running processes kill <i>pid</i> - kill process id <i>pid</i> killall <i>proc</i> - kill all processes named <i>proc</i> * bg - lists stopped or background jobs; resume a stopped job in the background fg - brings the most recent job to foreground fg <i>n</i> - brings job <i>n</i> to the foreground	tar cf <i>file.tar files</i> - create a tar named <i>file.tar</i> containing <i>files</i> tar xf <i>file.tar</i> - extract the files from <i>file.tar</i> tar czf <i>file.tar.gz files</i> - create a tar with Gzip compression tar xzf <i>file.tar.gz</i> - extract a tar using Gzip tar cjf <i>file.tar.bz2</i> - create a tar with Bzip2 compression tar xjf <i>file.tar.bz2</i> - extract a tar using Bzip2 gzip <i>file</i> - compresses <i>file</i> and renames it to <i>file.gz</i> gzip -d <i>file.gz</i> - decompresses <i>file.gz</i> back to <i>file</i>
File Permissions	Network
chmod <i>octal file</i> - change the permissions of <i>file</i> to <i>octal</i> , which can be found separately for user, group, and world by adding: <ul style="list-style-type: none">4 - read (r)2 - write (w)1 - execute (x) Examples: chmod 777 - read, write, execute for all chmod 755 - rwx for owner, rx for group and world For more options, see man chmod .	ping <i>host</i> - ping <i>host</i> and output results whois <i>domain</i> - get whois information for <i>domain</i> dig <i>domain</i> - get DNS information for <i>domain</i> dig -x <i>host</i> - reverse lookup <i>host</i> wget <i>file</i> - download <i>file</i> wget -c <i>file</i> - continue a stopped download
SSH	Installation
ssh <i>user@host</i> - connect to <i>host</i> as <i>user</i> ssh -p <i>port user@host</i> - connect to <i>host</i> on port <i>port</i> as <i>user</i> ssh-copy-id <i>user@host</i> - add your key to <i>host</i> for <i>user</i> to enable a keyed or passwordless login	Install from source: ./configure make make install dpkg -i <i>pkg.deb</i> - install a package (Debian) rpm -Uvh <i>pkg.rpm</i> - install a package (RPM)
Searching	Shortcuts
grep <i>pattern files</i> - search for <i>pattern</i> in <i>files</i> grep -r <i>pattern dir</i> - search recursively for <i>pattern</i> in <i>dir</i> <i>command</i> grep <i>pattern</i> - search for <i>pattern</i> in the output of <i>command</i> locate <i>file</i> - find all instances of <i>file</i>	Ctrl+C - halts the current command Ctrl+Z - stops the current command, resume with fg in the foreground or bg in the background Ctrl+D - log out of current session, similar to exit Ctrl+W - erases one word in the current line Ctrl+U - erases the whole line Ctrl+R - type to bring up a recent command !! - repeats the last command exit - log out of current session
* use with extreme caution.	

```
kali@kali:~$ sudo su
[sudo] password for kali:
root@kali:/home/kali# useradd SomeUser
root@kali:/home/kali# chown SomeUser:SomeUser ./Devin/myFile
root@kali:/home/kali# ls -l ./Devin/
total 4
-rw-r--r-- 1 SomeUser SomeUser 10 Feb  5 01:13 myFile
root@kali:/home/kali# chmod 770 ./Devin/myFile
root@kali:/home/kali# ls -l ./De
Desktop/ Devin/
root@kali:/home/kali# ls -l ./Devin/
total 4
-rwxrwx--- 1 SomeUser SomeUser 10 Feb  5 01:13 myFile
root@kali:/home/kali#
```

