

Table of Contents

PowerShell 3

PowerShell

PowerShell is a task automation and configuration management framework from Microsoft, consisting of a command-line shell and associated scripting language.

In PowerShell, administrative tasks are generally performed by cmdlets (pronounced command-lets), which are specialized .

What is PowerShell?

PowerShell is a task automation and configuration management framework from Microsoft. It is a command-line shell and scripting language that allows you to automate administrative tasks and manage system configurations across Windows, Linux, and macOS operating systems.

What can you do with PowerShell?

With PowerShell, you can automate repetitive administrative tasks, manage system configurations, and create custom tools and scripts to improve productivity. It provides access to a wide range of system management features, such as file system navigation, process management, and network configuration.

How is PowerShell different from traditional command-line shells?

PowerShell is designed to provide a more flexible and powerful command-line environment than traditional shells like Command Prompt or Bash. It uses a consistent object-oriented syntax and provides a comprehensive set of cmdlets (commands) that can be used to perform a wide range of tasks. It also supports remote management and scripting, making it well-suited for automating administrative tasks in large-scale environments.

Can PowerShell be used for scripting?

Yes, PowerShell is a powerful scripting language that allows you to automate administrative tasks and create custom tools and scripts. It provides access to a wide range of system management features, and its object-oriented syntax makes it easy to manipulate data and perform complex operations.

What resources are available to learn PowerShell?

There are many resources available to learn PowerShell, including online tutorials, books, and video courses. Microsoft provides comprehensive documentation for PowerShell, which includes a quick start guide, reference material, and examples of how to use PowerShell to perform common administrative tasks. There are also many community-driven resources, such as forums and blogs, where you can find tips and best practices for using PowerShell.

Snippet from [Wikipedia](#): **PowerShell**

PowerShell is a task automation and configuration management program from Microsoft, consisting of a command-line shell and the associated scripting language. Initially a Windows component only, known as **Windows PowerShell**, it was made open-source and cross-platform on August 18, 2016, with the introduction of **PowerShell Core**. The former is built on the .NET Framework, the latter on .NET (previously .NET Core).

PowerShell is bundled with all currently supported Windows versions, and can also be installed on MacOS and Linux. Since Windows 10 build 14971, PowerShell replaced Command Prompt (cmd.exe) and became the default command shell for File Explorer.

In PowerShell, administrative tasks are generally performed via *cmdlets* (pronounced *command-lets*), which are specialized .NET classes implementing a particular operation. These work by accessing data in different data stores, like the file system or Windows Registry, which are made available to PowerShell via *providers*. Third-party developers can add cmdlets and providers to PowerShell. Cmdlets may be used by scripts, which may in turn be packaged into modules. Cmdlets work in tandem with the .NET API.

PowerShell's support for .NET Remoting, WS-Management, CIM, and SSH enables administrators to perform administrative tasks on both local and remote Windows systems. PowerShell also provides a hosting API with which the PowerShell runtime can be embedded inside other applications. These applications can then use PowerShell functionality to implement certain operations, including those exposed via the graphical interface. This capability has been used by Microsoft Exchange Server 2007 to expose its management functionality as PowerShell cmdlets and providers and implement the graphical management tools as PowerShell hosts which invoke the necessary cmdlets. Other Microsoft applications including Microsoft SQL Server 2008 also expose their management interface via PowerShell cmdlets.

PowerShell includes its own extensive, console-based help (similar to man pages in Unix shells) accessible via the `Get-He lp` cmdlet. Updated local help contents can be retrieved from the Internet via the `Update-He lp` cmdlet. Alternatively, help from the web can be acquired on a case-by-case basis via the `-on line` switch to `Get-He lp`.

[Creative Commons Attribution-Share Alike 4.0](#)

External links:

- <https://powershellposse.com/devops-and-powershell/>
- <https://powershell.org/>
- <https://www.powershellgallery.com/>

[tool](#), [programming](#), [maintenance](#), [release](#), [automation](#)

From:

<https://almbok.com/> - **ALMBoK.com**

Permanent link:

<https://almbok.com/tools/powershell>

Last update: **2023/03/25 18:32**

