



Linux Academy

# PowerShell Essentials



Linux Academy

# PowerShell Essentials

Welcome



## PowerShell Essentials at Linux Academy

My Name is

Shahan Karim

and I will be your instructor for this course

**WELCOME**



### Linux Academy Features

- Live Labs
- Small Courses called Nuggets, Learning Paths, Course Schedules
- Notecards
- Study Groups
- Community
- Blog and Announcements
- Support and Ticket System



Linux Academy

# PowerShell Essentials

Course Objectives



## PowerShell Essentials Rewind

- What is PowerShell?
- Where can I use PowerShell?
- How to find commands and get help
- What is the structure of how commands and syntax work
- Storing Variables
- Manipulation of Files
- Extensibility and Remote Administration
- Tools and their benefits
- Best Practices



Linux Academy

# PowerShell Essentials

PowerShell Background



## PowerShell Essentials (Background)

- What is PowerShell?
  - PowerShell is an object-oriented programming language and interactive command line shell
- Why Use PowerShell?
  - Open Source
  - Cross-Platform
  - Muscle-Memory
  - Scripting Language
  - Extensibility
  - Automation
  - Configuration Management





Microsoft  Linux



## Open Source and Microsoft

- Microsoft has Open-Sourced PowerShell
- New Microsoft approach for the last few years
  - .Net & PowerShell
- Currently running in Alpha version
- All source code is now on GitHub
- Microsoft does not want a final product sent to the community. Started with a baseline so then community builds and adds
- Launched under the MIT License
- Governance model is published on GitHub



Linux Academy

# PowerShell Essentials

PowerShell Clients



## PowerShell Clients

- Microsoft will be following the Azure model for which platforms PowerShell will be developed to run as a client
  - 12 platforms in the development pipeline as Azure and .NET increase their platform capability
- Currently supports (as listed on the GitHub repo)
  - Windows
  - Mac
  - Linux
  - Docker



Linux Academy

# PowerShell Essentials

Discover Commands



## Discover Commands in PowerShell

- Make sure you have the help system updated (`update-help -force`)
  - Reference section on help system
- `Get-help` will go through the entire help system on PowerShell
  - Can be used as discovering device
  - Thousands of Cmdlets will make it difficult to memorize all
- How to search
  - Example `get-help *process*`
  - `Get-help get-process`
  - `Get-help get-process -online`



Linux Academy

# PowerShell Essentials

PowerShell Structure & Syntax



## Types of Commands

- Four different types of commands can be run
  - Native Administrative Commands
    - ifconfig
  - Aliases
    - cls, ls
  - Scripts
  - Cmdlets
- Auto-complete engine or intellisense





## PowerShell Essentials

- Common concept in syntax:
  - Command, argument, switch
    - For example `ifconfig -a`
- PowerShell builds structure on Verb-Noun
  - `Get-childitem`
  - `Set-location c:\`
  - `Clear-host`
- Parameters always begin with a dash or hyphen
  - Arguments follow parameters and can be joined using a comma
  - Example: `verb-noun -param Arg , Arg`
- Use help to determine command's capability
  - Notice symbols that denote special things [`<servicename[]>`]



Linux Academy

# PowerShell Essentials

Cmdlets and Usage



## PowerShell Cmdlets

- What are Cmdlets
  - Pronounced “Command-let”
  - Does not exist in any other model (Unix/Linux)
  - Created by Jeffrey Snover, who is the original architect who created PowerShell
  - Cmdlets are small commands have a specific format and purpose
  - Format is verb-noun
    - Comes from VMS DCL
  - Get-command
  - Cmdlets can be controlled as they can be used with parameters and arguments



Linux Academy

# PowerShell Essentials

Positional & Terse Syntax



## PowerShell Positional and Terse Syntax

- Understand that Cmdlets are based on positional parameters and arguments
  - Basically means that for certain Cmdlets that accept parameters, you must, when required, place the parameters in the right sequence
- Terse Syntax
  - Essentially means using shortened commands and/or aliases to accomplish the same command with less typed letters/words
    - Caution: Easy to write quick commands, harder to diagnose/debug down the line so scripting should not be terse syntax
    - Example terse syntax
    - Stop-process → spps



Linux Academy

# PowerShell Essentials

Pipeline and Pipe Commands



## PowerShell Pipe Commands

- Used conceptually as an object-based environment
- For example get-process. For each row returned would be a single object
- Methods will describe what an object can do
- Allows flexibility and granular manipulation
- Used to string results and also for real-time maintenance of data by utilizing variables (more in later slides on this)
- Example
  - `Get-process -name Mail | stop-process`
    - Will result in pulling all process from machine, then find the Mail process, then send Mail process as an object to which then stop-process is applied to



Linux Academy

# PowerShell Essentials

## Storing Variables





## PowerShell Variables and Storing Data

- Allows manipulation of data when needed
- Data can be stored to variables
  - Variables are temporary memory storage
  - Type of data agnostic and size of data
- Find all commands for variables
  - Get-help \*variable\*
- Not necessary for specific cmdlets to be used for variables
- Variables are associated with names
  - Variable needs to start with “\$” which will not be part of the name but will be used as a variable initiator
- Will only last with session
- Able to store results from commands while setting variable
  - Example: \$variable= get-process -name Mail



Linux Academy

# PowerShell Essentials

Customization of Results



## PowerShell Results and Customization

- Understanding parameters and arguments is the core of this section
  - Capabilities include items such as filtering and sorting
- Expressions can be used for further file/data manipulation, such as comparison
- For example compare Hello to Ello using
  - `"Hello" -eq "Ello"`



Linux Academy

# PowerShell Essentials

No News is Good News



## PowerShell Errors and Error Handling

- First point
  - Key item difference between clients: case-sensitivity
- Different types of errors
  - Syntax errors
  - Positional/parameters errors
  - Connectivity errors
  - Translation errors
- Color coded display to enhance intellisense and code diagnostics
- Error handling is completely possible, although out-of-scope for this course. We do want to mention it so you understand the toolset for PowerShell



Linux Academy

# PowerShell Essentials

Modules & Snap-Ins



## PowerShell Modules and Snap-Ins

- What is a module in PowerShell?
  - Set of related Windows PowerShell functionalities, grouped together as a convenient unit (can be saved in a single directory).
  - By defining a set of related script files, assemblies and related resources as a module, you can reference, load, persist, and share your code much easier than you would otherwise.



## What Does a Module Consist of?

- Modules consist of
  - Code file – either a PS script or a managed cmdlet assembly.
  - Anything else that the above code file may need, such as additional assemblies, help files or scripts.
  - A manifest file that describes the above files, as well as stores metadata such as author and versioning information.
  - A directory that contains all of the above content, and located where PowerShell can find it.

\* NOTE \* None of these are required. You can use script stored in .psm1 file





## Example of Module

- To administer Azure from a Mac machine using PowerShell:
  - Import module using this command
    - `"Install-Package -Name AzureRM.NetCore.Preview -Source https://www.powershellgallery.com/api/v2 -ProviderName NuGet -ExcludeVersion -Destination "`
  - Note how the code includes reaching out to URL repository to download the module. This increases its flexibility to maintain distributed code



Linux Academy

# PowerShell Essentials

Files in PowerShell



## PowerShell File and File Manipulation

- Files can be written and read in various formats
  - For example: txt, XML, JSON, Pearl, CSV
- Pipe commands are your best friend to accomplish various tasks
- Sample code to achieve a written file for processes on a Mac machine
  - `Get-process | out-file --FilePath /users/username/process.csv`
- Sample code to read a file a Mac machine
  - `Cat /users/username/process.csv`
- Pipe commands can take file parameters to enable the ability to have scripts access several files during the script process when needed throughout code execution



Linux Academy

# PowerShell Essentials

Framework Extensibility



## PowerShell Framework Extensibility

- Having built in binaries
  - Calculator
  - Notepad
  - Printer
  - File Writer
- File analysis for example
  - Have the ability to compare two files and generate a result
- For example, write perl scripts and curl scripts natively
- Intellisense for commands execution
- Automation and configuration management



Linux Academy

# PowerShell Essentials

Remote Connectivity



## PowerShell Essentials

- Remote connectivity is possible in three flavors and growing
  - Implicit Remoting
    - Creates remote session and imports remote session with modules into local shell. Doesn't require local module installation
  - Direct Remote Control
    - Direct remote PowerShell sessions are very similar to using a terminal to access a virtual machine, for example
  - OpenSSH (ported and alpha)
    - As of 2015, Microsoft has joined the OpenSSH community and are developing the OpenSSH product to be built into PowerShell



Linux Academy

# PowerShell Essentials

Rewind of the Essentials





Linux Academy

# PowerShell Essentials

Next Steps



## PowerShell Essentials Next Steps

- Practice, Practice, Practice (Use our Labs)
- Create small projects for fun
- Explore our Certifications Courses
- Begin a Learning Path
- Try our Course Nuggets out for some hands on quick learning!
- Explore other Azure courses such as 70-533 exam or Azure ML
- Learn a new piece of technology (try our new Chef course or Docker)