

Kamilimu cybersecurity track: Cohort 9

April 30, 2025

UnderTheWire Century Worksheet: Introduction to PowerShell

This worksheet provides introductory challenges from the UnderTheWire platform to help you get comfortable with basic PowerShell commands. PowerShell is a powerful scripting language used extensively in Windows environments, making it an important tool for security assessments on Windows systems.

You can find the UnderTheWire Century wargame here: <https://underthewire.tech/century/>

Instructions: For each challenge in **Part 1**, you will be given a goal. Use your PowerShell knowledge to find the answer.

You can complete this assignment on a Windows system, where PowerShell is already installed, or within the PowerShell terminal available in Kali Linux and Parrot OS. Open it by searching for "PowerShell" in the Start menu on Windows.

Part 1: Core Century Challenges (Recommended for Everyone)

Challenge 1: Century Level 0 -> Century Level 1

- **Goal:** The goal of this level is to execute your first PowerShell command. The instructions on the website will guide you to the command.
- **Task:** Open PowerShell and enter the command as shown on the website. This will reveal the answer for the next level.
- **Commands to explore:** The command given on the website.

Challenge 2: Century Level 1 -> Century Level 2

- **Goal:** This level introduces you to getting help in PowerShell. You need to find a specific parameter of a command.
- **Task:** Use PowerShell's help system to find the parameter of the specified command that will reveal the answer.
- **Commands to explore:** Get-Help

Challenge 3: Century Level 2 -> Century Level 3

- **Goal:** This level involves listing directory contents, similar to ls in Linux.
- **Task:** Use the appropriate PowerShell command to list the files and directories and find the one indicated.
- **Commands to explore:** Get-ChildItem

Challenge 4: Century Level 3 -> Century Level 4

- **Goal:** This level requires you to read the contents of a file.
- **Task:** Use the correct PowerShell command to display the contents of the specified file.
- **Commands to explore:** Get-Content

Challenge 5: Century Level 4 -> Century Level 5

- **Goal:** This level introduces you to filtering output.
- **Task:** Use a PowerShell command to find a specific piece of text within the output.
- **Commands to explore:** Select-String

Part 2: Century++ Challenges (Optional - For Those Seeking a Greater Challenge)

These additional challenges are for learners who are already comfortable with the basics and want to push their skills further.

Challenge 6: Century Level 5 -> Century Level 6

- **Goal:** This level introduces variables.
- **Task:** Use a variable to store a value and then use that variable to get the answer.
- **Commands to explore:** Variables (\$myVariable = "value")

Challenge 7: Century Level 6 -> Century Level 7

- **Goal:** This level requires you to use a loop.
- **Task:** Use a loop to iterate through a set of data and find the specific item.
- **Commands to explore:** foreach

Challenge 8: Century Level 7 -> Century Level 8

- **Goal:** This level introduces you to aliases.
- **Task:** Find the alias for a specific command and use it to get the answer.
- **Commands to explore:** Get-Alias

Challenge 9: Century Level 8 -> Century Level 9

- **Goal:** This level requires you to use a command to get system information.
- **Task:** Use the appropriate command to retrieve specific system information.
- **Commands to explore:** Get-Process

Challenge 10: Century Level 9 -> Century Level 10

- **Goal:** This level introduces remote execution.
- **Task:** Use a command to execute something remotely
- **Commands to Explore:** Invoke-Command

Bonus Scripting Challenge

- **Automating Century Level 4:** Write a PowerShell script (a .ps1 file) that retrieves the content of the file from level 4 and displays it on the console.
 - **Hint:** Create a new file with a .ps1 extension (e.g., level4.ps1) and use the Get-Content command. Then run the script in PowerShell using .\level4.ps1.
- **Potential Further Scripting:**
 - **Iterating through Century Levels:** Write a PowerShell script that attempts to automate connecting to and retrieving information from multiple Century levels. This will be more challenging in Century than it was in Bandit.
 - **Help File Reader:** Write a PowerShell script that takes a command name as input and displays the relevant sections of the help file for that command (e.g., the SYNOPSIS, DESCRIPTION, and EXAMPLES).
 - **System Information Gathering Script:** Create a PowerShell script that gathers and displays various system information, such as the operating system version, installed memory, and network adapters. Format the output in a user-friendly way.

Important Notes

- **Part 1 is recommended for all learners to build a solid foundation.**
- **Part 2 and the scripting challenges are optional and designed for those who want a more advanced experience.** Feel free to explore these at your own pace.
- **Use PowerShell to solve these challenges.**
- **Do not share the answers you find with others.** The goal is for everyone to learn by solving the challenges themselves.
- **Take your time and experiment.** If you get stuck, try using Get-Help <command> to learn more about a specific command. You can also search online for PowerShell tutorials and examples.
- **Remember, if you encounter any difficulties, please don't hesitate to reach out for assistance.**