



# Unlocking the secrets of PowerShell Help

# Sources of help

## Order of precedence

### .NET reflection

Gives basic syntax and parameter information when no help available

### Comment-based help for scripts

Documentation lives in the source code of the script

### MAML-based help

Static XML files that ship with the module

Can be updateable if you design for it

Only way to document cmdlets in compiled modules

Difficult to create

# Features of Help



## Search and filter

by topic,  
parameter,  
component, role,  
functionality,  
category



## Control display verbosity

-Examples  
-Detailed  
-Full



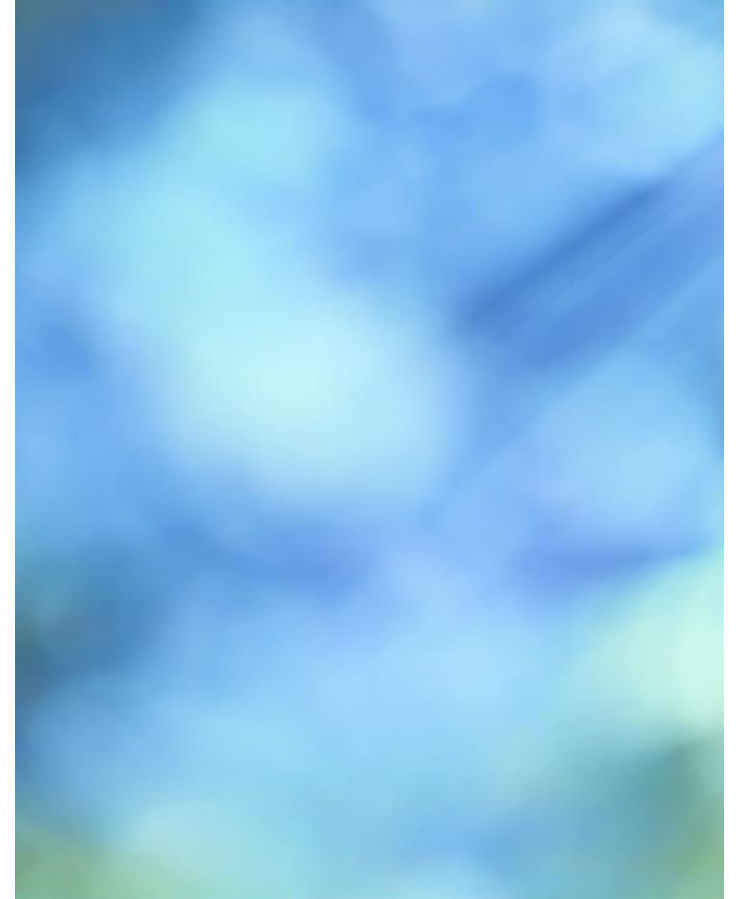
## Open in a browser

-Online



## Get new content

Update-Help





A close-up, shallow depth-of-field photograph of a computer keyboard. The central focus is on a dark grey key with a white 'X' symbol. To its left, a portion of a key with blue markings is visible. Below the 'X' key, another dark grey key is partially in focus. The background is a soft, out-of-focus grey, suggesting the rest of the keyboard and the desk surface.

# Supporting –Online

- Defined by the HelpUri property of the CmdletBinding attribute for your function
  - Value is a URL that points to the webpage containing the documentation
  - Becomes metadata in the markdown documentation
  - Compiled into the MAML
- See [about Functions CmdletBindingAttribute](#)

# Supporting Update-Help

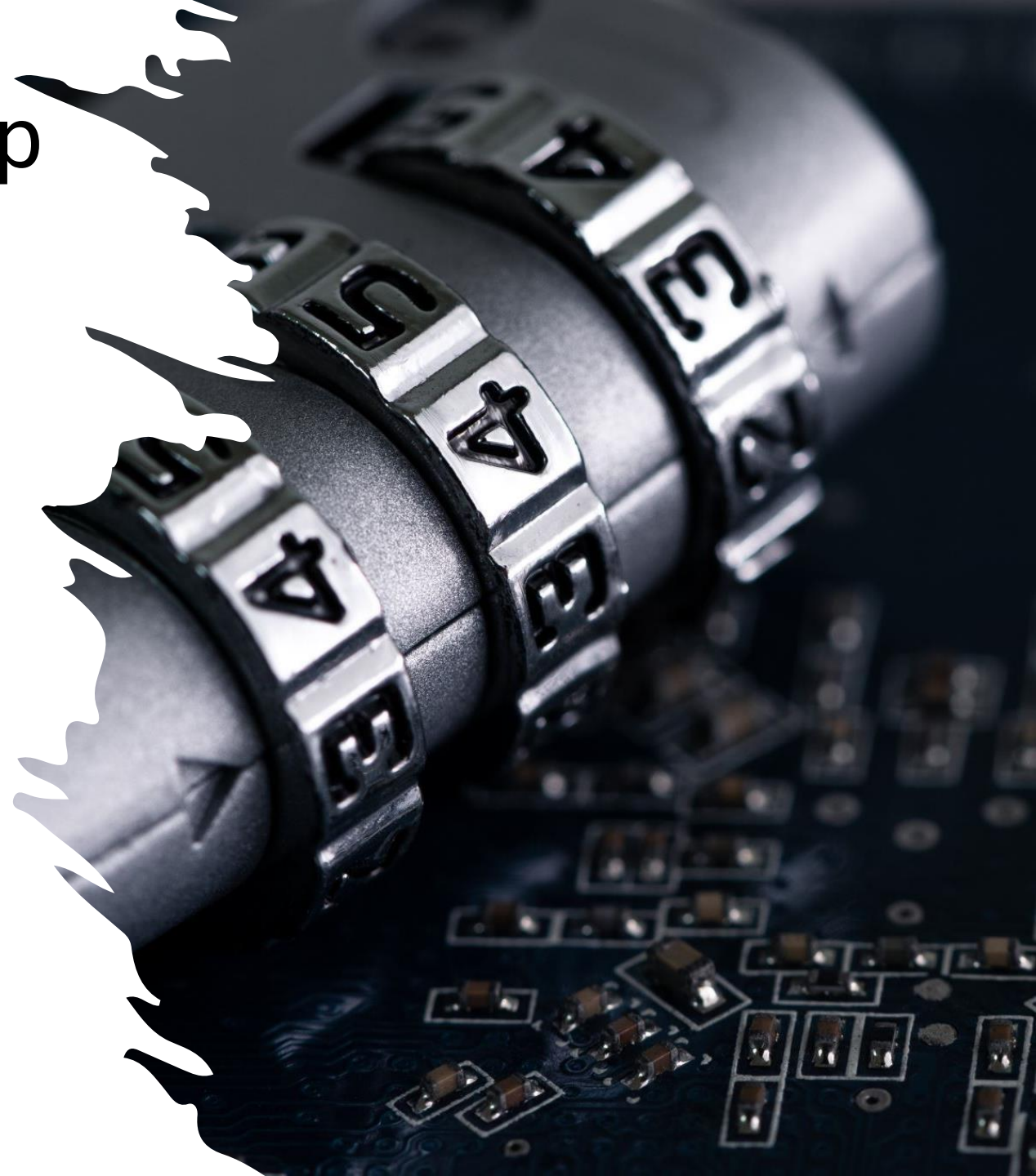
Got to have the right combination

## Difficult to create assets

- MAML files containing help content and metadata
- Package MAML files in a CAB/ZIP
- Create a HelpInfo Xml file to match CAB/ZIP package

## Host the files in a web service

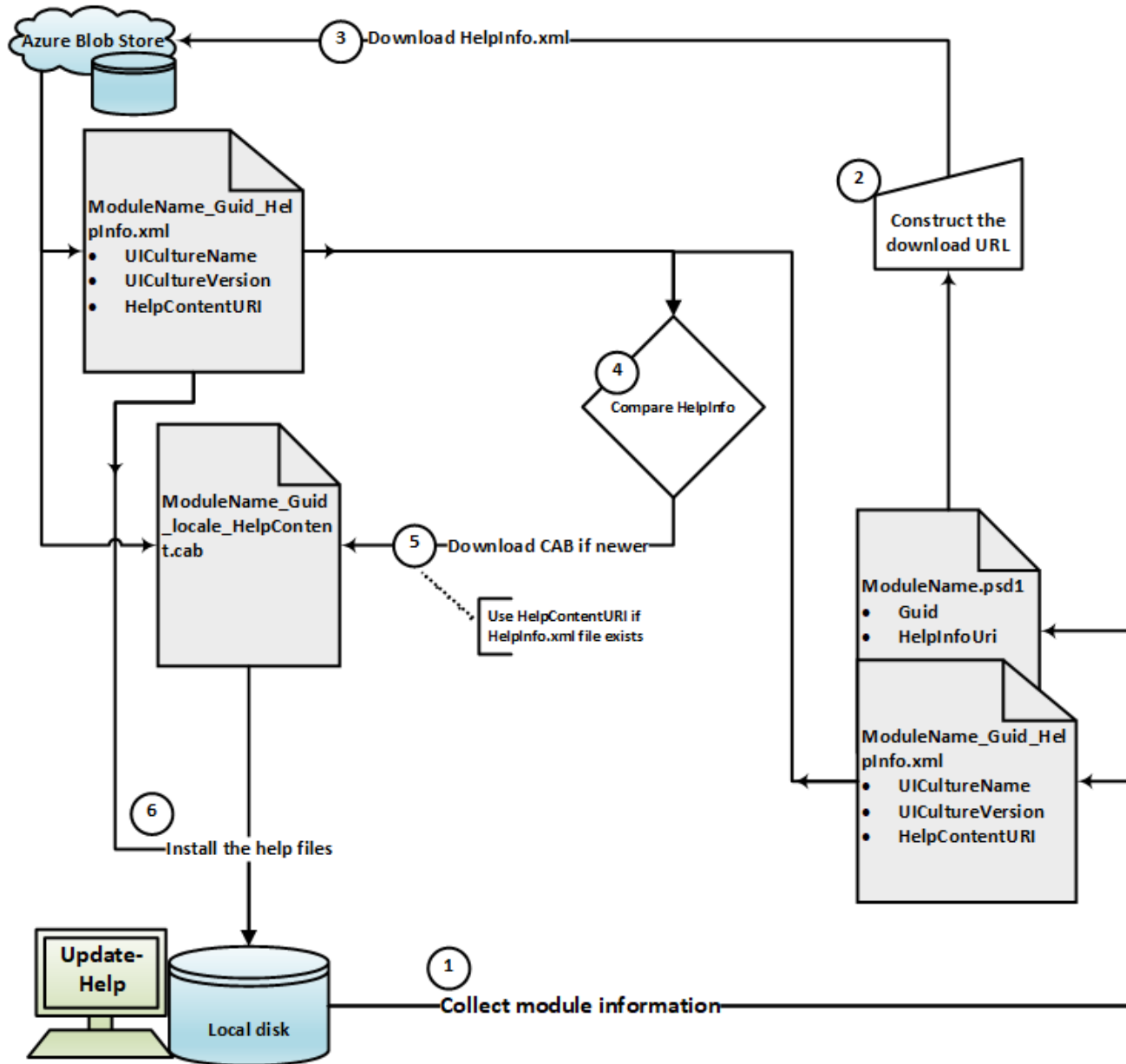
- Web service endpoint defined in HelpInfoUri property of the module manifest (.psd1)



# Update-Help process

## Update-Help Logic flow

1. Get module information from module manifest and HelpInfo.xml (if available).
2. Construct the download URLs from the module info.
3. Download the HelpInfo.xml
4. Compare local help version to downloaded version.
5. If local version is older, download the HelpContent.cab.
6. Unpack the CAB and install the help files





# Platypus – a strange looking mammal

---

- One of five monotremes - mammals that lay eggs instead of giving birth to live young
- One of the few species of venomous mammals - venomous spur on hind foot
- It senses prey through electrolocation





# PlatyPS – A tool for PowerShell MAML

- History
  - Started as a hackathon project
  - Used in production but no code owner for years
  - Lots of bugs and quirks
  - Working on a rewrite but progress is slow
- Why use it?
  - Easiest way to create and author help
  - Write in markdown – compile to MAML
  - Create package for Update-Help



## PlatyPS Workflow

1. Import-Module
2. Create new markdown files (cmdlet and module)
3. Edit markdown files to add descriptions, examples, links, notes, etc.
4. Compile markdown into MAML
5. Test the MAML
6. Create the module help CAB/ZIP and HelpInfo XML files.
7. Publish CAB and XML to web service

