This PowerShell script has three main parts: it retrieves a list of services with specific criteria, checks each service's path for spaces and not enclosed in quotes, and then displays the list of services that meet the criteria along with some additional properties. Let's break it down step-by-step:

PowerShell Script: "C:\Development\PowerShell\PowerShell-Script-Omatic\FindObjectsReplaceWithQuotes\_V7.ps1"

Text

Description automatically generated

**Get a list of services that match specific criteria:**

$services = Get-CimInstance Win32\_Service |

Where-Object {$\_.StartMode -eq 'Auto' -and $\_.PathName -notlike 'C:\Windows\*' -and $\_.Name -ne $null -and $\_.DisplayName -ne $null}



* Get-CimInstance Win32\_Service:
  + Retrieves all instances of the Win32\_Service class, which represents services on a Windows system.
* Where-Object {...}:
  + Filters the services based on specific criteria.
  + The $\_ variable represents the current object (service) in the pipeline.
* StartMode -eq 'Auto':
  + The service's start mode is set to 'Auto'.
* PathName -notlike 'C:\Windows\*':
  + The service's PathName doesn't start with 'C:\Windows'.
* Name -ne $null:
  + The service's Name property is not null (-ne stands for “not equal to”)
* DisplayName -ne $null:
  + The service's DisplayName property is not null.

**Check each service's path for spaces and not enclosed in quotes, and update if needed:**

foreach ($service in $services) {

if ($service.PathName -match '\s' -and $service.PathName -notmatch '^"(.\*)"') {

$quotedPath = '"' + $service.PathName + '"'

$service.PathName = $quotedPath

$service.Put()

Write-Host "The path $($service.PathName) is vulnerable and has been modified with quotes."

}

}

Text

Description automatically generated

* foreach ($service in $services):
  + Iterate through each service in the $services variable.
* if ($service.PathName -match '\s' -and $service.PathName -notmatch '^"(.\*)"'):
  + Check if the service's PathName contains spaces and is not already enclosed in quotes.
* $quotedPath = '"' + $service.PathName + '"':
  + Create a new variable $quotedPath and enclose the PathName with quotes.
* $service.PathName = $quotedPath:
  + Update the service's PathName with the new quoted path.
* $service.Put():
  + Save the changes made to the service.
* Write-Host "The path $($service.PathName) is vulnerable and has been modified with quotes.":
  + Print a message indicating that the service's path was vulnerable and has been updated with quotes.

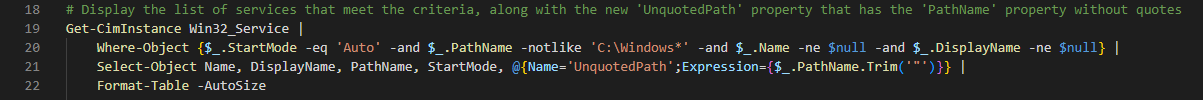
**Display the list of services that meet the criteria, along with additional properties:**

Get-CimInstance Win32\_Service |

Where-Object {$\_.StartMode -eq 'Auto' -and $\_.PathName -notlike 'C:\Windows\*' -and $\_.Name -ne $null -and $\_.DisplayName -ne $null} |

Select-Object Name, DisplayName, PathName, StartMode, @{Name='UnquotedPath';Expression={$\_.PathName.Trim('"')}} |

Format-Table -AutoSize



* Get-CimInstance Win32\_Service and Where-Object {...}:
  + Retrieves and filters the services as described in step the earlier bullet point
* Select-Object Name, DisplayName, PathName, StartMode, @{Name='UnquotedPath';Expression={$\_.PathName.Trim('"')}}:
  + Selects specific properties of the services to display, and adds a new property 'UnquotedPath'