



The next generation command line scripting

Presented by
Bob McCoy, CISSP/ISSAP, MCSE
Microsoft Services
Cyber Security Forum 07/18/2007

Windows PowerShell

- cmd.exe and command.com
 - Lack of scriptable functionality
 - Lack of documentation
- Windows Script Host (WSH)
 - Not integrated with the shell
 - Vulnerable
- Cscript
 - Not integrated with the shell
- Administrative tasks and automation too dependent on proprietary GUI based tools
- Scriptable wrappers difficult to implement

Command Line Interface

- Gottfried Leibniz's monadology (1704)
 - Universe is a composite of fundamental elements (cmdlets) integrated in a pre-established harmony
- In place of CMD and WSH
- Microsoft GUI wrappers
- Supported on XP, Server 2003, Vista and Longhorn Server (x86, x64, IA64)
- Foundation for Exchange Server 2007
- Built on .NET Framework

Background

- Object Orientation
- .NET Based (*requires .NET 2.0*)
- CLI & Scripting Language (C# like, Perl like)
- Cmdlets
- Platform Independent Utilities
- Users
 - Administrators
 - Scripters
 - System integrators

Windows PowerShell

- Strong guidelines
- Aliasing
- Tab-completion & partial parameter usage
- Command line editing
- Pipelining
- Object utilities
- Better documentation
- Security

Windows PowerShell

- Cmdlets
 - Object oriented
 - Extensible
 - Easily discoverable
- Access to objects
 - ADO
 - .NET
 - WMI
 - COM
 - Etc...
- PSDrives

Feature Highlights

- Loose
 - `$d="7/18/2007"`
 - `Function foo() {$args[0]}`
- "Strong"
 - `[datetime]$d="4/20/2005"`
 - `function foo([datetime]$date) {$date}`
 - `[int] [char] "a"`
- Extensible
 - `add-member $d Note Description "CSF Briefing"`
 - `$d.Description`
- PowerShell works on any .NET type - not a fixed set of "scripting types"

Typing

- Basic Shell usage
 - Discovering PS commands and functionality
 - Get-Command
 - Get-Alias
 - Get-Help
 - Get-Member
- Other Useful cmdlets not in Demo
 - Select-Object
 - ForEach-Object
 - Where-Object
 - Sort-Object

DEMO – Basic Usage

- Access WMI data
- Use of alternate credentials
- Get-WmiObject win32_share
- Get-WmiObject win32_share | Format-List [a-z]*
- GWmi win32_share | Format-Table Name,Path -hide
- Binds to root\cimV2 namespace on the local host by default, params

Demo – WMI

- A data store location
- Access it just like a file system drive
- New-Psdrive
- Get-Psdrive
- Set-Location (aliased to cd)
- Help About_Provider

Demo - PSDrive

- Debug
 - Programmer-level information
- ErrorAction
 - SilentlyContinue* | *Stop* | *Continue* | *Inquire*
 - What to do if an error occurs?
- ErrorVariable *VariableName*
 - Assign errors to a variable
- Verbose
 - Additional information about the activities being performed

Ubiquitous Parameters

- Commands with side-effects support:
 - Whatif
 - `gps | where {$_.handles -ge 500} | stop-process -WhatIf`
 - Confirm
 - `Stop-process S* -Confirm`
 - Verbose
 - `Stop-Process [a-x]*[q]*[r-t] -Verbose`

Interactivity

- Get-ExecutionPolicy
- Set-ExecutionPolicy
 - Restricted | AllSigned | RemoteSigned | Unrestricted*
 - Policy can only be set by Admin
- Use of code signing
- Self-signing cert
 - <http://www.hanselman.com/blog/SigningPowerShellScripts.aspx>
 - Root ► Personal Cert ► Trusted Publisher
- Does not implicitly execute scripts from current directory

Security – Execution Policy

"hello" | get-member

```
# SIG # Begin signature block
# MIIEMwYJKoZIhvcNAQcCoIIIEJDCCBCACAQExCzAJBgUrDgMCGGUAMGkGCisGAQQB
# gjcCAQSGWzBZMDQGCisGAQQBgjcCAR4wJgIDAQAABBAfzDtgWUsITrck0sYpfvNR
# AgEAAgEAAgEAAgEAAgEAMCEwCQYFKw4DAhoFAAQUN0zmm7MbQLHr3Ay6MaRZ30DF
# ywCgggI9MIICOTCCAaagAwIBAgIQ/OlnVY3d169DGW1clWeFqjAJBgUrDgMCHQUA
# MCwxKjAoBgNVBAMTIVBvd2VyU2h1bGwgTG9jYWwgQ2VydG1maWNhdGUgUm9vdDAe
# Fw0wNzA3MTYxMzM1NDZaFw0zOTEyMzEyMzU5NTlaMBoxGDAWBgNVBAMTD1Bvd2Vy
# U2h1bGwgVXNlcjCBnzANBgkqhkiG9w0BAQEFAAOBjQAwGyKCGYEAoMd3ZkVZ0cAc
# 4kB2z5zIq7ttKsb+nK1Zv+WnuoIL89aWlXzZqqrXU5sAbBrLulaq84I7lnTR/LxH
# 0Ts3EwaxXXdzWx45u4Dnhnp7IR6pdoVaOnOaKu6SU68sQD5CV5HKbBzbRyxBuDxX
# 57QgWHhsLwHN05+mG3G41XMgfLLI0WcCAwEAAAN2MHQwEwYDVR0lBAwwCgYIKwYB
# BQUHAWMwXQYDVR0BBFYwVIAQwH7/oqgShF/e6q4m27AtX6EuMCwxKjAoBgNVBAMT
# IVBvd2VyU2h1bGwgTG9jYWwgQ2VydG1maWNhdGUgUm9vdIIQCztDjdeS64tDd5vS
# 3g4NNTAJBgUrDgMCHQUAA4GBALdwm6EP3CoT4qjG5yNBvtNtO447duj7LlDjWBio
# np5C3CpV3uxmFlOh2ejhQRa8kfqfe7t+S0dchQv1rTLf7m65TwZeRf5yLLvOaKI5
# myZIDymYGBLE2Hl9iqWm8IdiVDaVCRi/H2XB/BtbkkfS7iekce3tF/4SL/vNSRMW
# 1Tj6MYIBYDCCAVwCAQEwQDAsMSowKAYDVQQDEyFQb3dlclNoZWxsIExvY2FsIENl
# cnRpZmljYXRlIFJvb3QCEPzpZ1WN3devQxltXJVnhaowCQYFKw4DAhoFAKB4MBGg
# CisGAQQBgjcCAQwxCjAIOAKAAKECgAAwGQYJKoZIhvcNAQkDMQwGCisGAQQBgjcC
# AQQwHAYKKwYBBAGCNwIBCzEOMAwGCisGAQQBgjcCARUwIwYJKoZIhvcNAQkEMRYE
# FITgShAtGf7rHeEmHnBR40TpbGerMA0GCSqGSIsb3DQEBAQUABIGAoiR6OP3ixsjE
# Lks/xqcSHeYwjXfgIODFUGUnzBJr8nm/tsFvbfGEkgWYu0M8MqCvt7W90vIQT1Mq
# iF8eEr+8e17mj0puGvForlwRR3aP9kAcPhVC946t87ymtx/+RLbmxkbKM5/iJOagW
# EDkfvHuVHbXzChe6ZS8j3HbjNV0HKLA=
# SIG # End signature block
```

Code Signing Sample

- Read-Host -assecurestring
- ConvertTo-SecureString
- ConvertFrom-SecureString
- Encryption
 - No Key – uses Triple-DES
 - With Key – uses Rijndael
 - Windows Data Protection (DPAPI)
<http://msdn2.microsoft.com/en-us/library/ms995355.aspx>
 - <http://www.leeholmes.com/blog/SecureStringsInPowerShell.aspx>
 - get-help ConvertFrom-SecureString

Security – Passwords

```
C:\Program Files\Microsoft\Exchange Server>bin\setup /q /S:M.E.C./sourceDir:C:\Program Files\Microsoft\Exchange Server\bin\setup
```

Welcome to Microsoft Exchange Server 2007 Unattended Setup

The following Server Roles will be installed:

Mailbox, ClientAccess, Edgeguard

Performing Microsoft Exchange Server Pre-Setup Checks

Communication Channels	COMPLETED
Mailbox Role Checks	COMPLETED
Client Access Role Checks	COMPLETED
Edgeguard Role Checks	COMPLETED

Installing Microsoft Exchange Server

Copying Exchange Files	COMPLETED
Organization Preparation	COMPLETED
Mailbox Role	21

[illegible]

The screenshot shows the Exchange System Manager console. On the left, the 'Mailbox Servers' folder is expanded under 'Exchange Organization'. The main pane displays a table of Mailbox servers. The table has columns for 'Name', 'Assigned Type', 'Status', and 'Time'. The servers listed are: Exchange01, Exchange02, Exchange03, Exchange04, Exchange05, Exchange06, Exchange07, Exchange08, Exchange09, Exchange10, Exchange11, Exchange12, Exchange13, Exchange14, Exchange15, Exchange16, Exchange17, Exchange18, Exchange19, Exchange20, Exchange21, Exchange22, Exchange23, Exchange24, Exchange25, Exchange26, Exchange27, Exchange28, Exchange29, Exchange30, Exchange31, Exchange32, Exchange33, Exchange34, Exchange35, Exchange36, Exchange37, Exchange38, Exchange39, Exchange40, Exchange41, Exchange42, Exchange43, Exchange44, Exchange45, Exchange46, Exchange47, Exchange48, Exchange49, Exchange50, Exchange51, Exchange52, Exchange53, Exchange54, Exchange55, Exchange56, Exchange57, Exchange58, Exchange59, Exchange60, Exchange61, Exchange62, Exchange63, Exchange64, Exchange65, Exchange66, Exchange67, Exchange68, Exchange69, Exchange70, Exchange71, Exchange72, Exchange73, Exchange74, Exchange75, Exchange76, Exchange77, Exchange78, Exchange79, Exchange80, Exchange81, Exchange82, Exchange83, Exchange84, Exchange85, Exchange86, Exchange87, Exchange88, Exchange89, Exchange90, Exchange91, Exchange92, Exchange93, Exchange94, Exchange95, Exchange96, Exchange97, Exchange98, Exchange99, Exchange100.

Mailbox Database 1 Properties

General | Database | Limits

Mailbox Database 1

Default public store:
[C:\Program Files\Microsoft Exchange Server\Exchange \(64-bit\)\Mailbox Databases\](#) Browse...

Offline address list:
[C:\Program Files\Microsoft Exchange Server\Exchange \(64-bit\)\Mailbox Databases\](#) Browse...

☒ Client support S/MIME signatures
☐ Display plain text messages in fixed-size font

Maintenance interval: 14 days Customize...

OK Cancel Apply

E2007 Management Architecture

- PowerShell is extremely powerful (and very cool)
- The price is right for Windows users
- Documentation is extensive
- Scripting experience much richer at both design and execution time
- Strong community support (blogs, repositories, books)

Wrap Up

- Official PowerShell site
<http://www.microsoft.com/powershell>
- TechNet Scripting Center
<http://www.microsoft.com/technet/scriptcenter/hubs/msh.msp>
- PowerShell Team Blog
<http://blogs.msdn.com/PowerShell/>
- Sapien Press Blog
<http://blog.sapien.com/>
- The PowerShell Guy
<http://thepowershellguy.com/>
- Books
 - “PowerShell Step-by-Step” (MSPress)
 - “PowerShell TFM” (Sapien Press)
 - “Windows PowerShell in Action” (Manning Pub)
 - “Windows PowerShell Quick Reference” (O’Reilly)

Resources

Demo Script

Get-process

Ps | where { \$_.handles -gt 500 }

Ps | where { \$_.handles -gt 1000 } | sort ws -descending | select-object -first 5

Ps | gm

\$g = ps notepad; \$g.WaitForExit()

\$g = ps notepad; \$g.Kill()

Get-service

Get-service | where { \$_.status -eq "running" }

Get-service | format-table -groupby status

Get-service | sort status | format-table -groupby status

Get-service | sort status,name | format-table -groupby status

WMI

Get-WMIObject Win32_Share

Get-WMIObject -list | where { \$_.name -like "win32*" }

gwmi win32_share | format-table name,status,description -auto -hide

Regex

("foo" -match "o")

("foo" -match "^o")

("foo" -match "o\$")

("foo" -match "[a-l]")

("foo" -match "[a-l].*o\$")

("foo" -match "[a-l].*x\$")

```
[regex]$rx = "[a-l].*o$"
```

```
"foo" -match $rx      "fox" -match $rx
```

Date and Time

```
$d = get-date
```

```
$d.dayofweek
```

```
$d.adddays(90)
```

```
$d.getdatetimestrings()
```

```
$d.touniversaltime().tolocaltime()
```

```
$bd = get-date -year 2007 -month 6 -day 25
```

```
$d - $bd
```

```
($d - $bd).days
```

Registry

```
$r = get-item 'HKLM:\software\Microsoft\Windows\CurrentVersion\Run'
```

```
$r.getvaluenames()
```

```
$r.getvalue("Zune Launcher")
```

```
$reg = get-item "HKCU:\Software\Microsoft\Windows\CurrentVersion"
```

```
$r = $reg.OpenSubKey("run",$true)
```

```
$r.setvalue("foo","bar")
```