

# 淺談 Windows PowerShell 管理技術

彭耀東 / 財金資訊公司安控部資源控管組組長

陳美惠 / 財金資訊公司安控部資源控管組高級工程師

## 一、前言

一般使用者接觸 Windows 作業系統的第一印象，通常是具親和力的圖形化操作介面，藉由簡易的滑鼠點選，就可以操作某項功能或完成某項工作。雖然微軟一直致力於提升圖形化操作介面的親和力，但它對於指令化作業環境功能之增強也從未停止，投入的資源甚或與日俱增。

對於單一電腦的操作，圖形化操作介面其實已經相當方便，但是 Windows 系統管理人員所面對的，卻不僅是單一電腦，而是一群用戶端的個人電腦，或是二十四小時不中斷的營運伺服器，單純的圖形化操作介面已無法滿足系統管理的需求與目標，尚需要指令化作業環境協助完成各項系統管理工作。

作業系統的管理是既複雜又具重複性的作業，而營運伺服器的作業環境特別講究系統效能及管理效率，圖形化操作介面並不是最佳解決方案。歷來，各種 UNIX 作業系統都具有功能強大的 Shell 環境及 Script 語言，可以協助系統管理人員完成日常各項系統管理工作。相對而言，Windows 作業系統卻一直缺乏完整的指令化作業環境可供系統管理人員使用，直到 Windows PowerShell 出現，Windows 作業系

統終於有了可以與 UNIX 系統 Shell 環境相提並論的指令化作業環境。

## 二、Windows PowerShell 功能概述

Windows PowerShell 是微軟為 Windows 環境所開發的 Shell 及 Script 語言技術，Script 語言具有易學易用的特性，進入之門檻不高，經常是輔助系統管理人員進行系統管理作業的工具。Script 語言大多採用直譯的方式執行，效能雖然比經過編譯的系統程式語言差，但隨著電腦硬體的快速發展，這種差異已逐漸縮小到不易察覺。

Windows PowerShell 以 .NET 技術為基礎，並向上相容於原有之 Windows Script Host (WSH) 環境。WSH 是 Script 語言在 Windows 作業系統中的執行環境，以微軟的元件物件模型 (Component Object Model, 簡稱 COM) 技術為基礎，在 Windows PowerShell 出現之前，系統管理人員常會利用 WSH 來完成許多管理工作。Windows PowerShell 的出現將原本 Windows 環境下各自為政的 Script 語言 (WSH) 和操作指令重新緊密黏合，讓 Windows 作業系統擁有類似 UNIX BASH 的 Shell 環境，更適用於大型、複雜、重要的電腦運算環境。

Windows PowerShell 具備以下特性：

(一) 具 130 餘種的命令列工具( 稱為 Cmdlet )  
可用來執行系統管理工作

命令列工具，如：服務管理、事件日誌管理、Process 管理、憑證管理、登錄( Registry ) 管理及使用 Windows Management Instrumen-

tation (WMI) 等。只要利用簡單的 Get-Command 指令，即可列出 Windows PowerShell 環境支援的所有 Cmdlet，如圖一。

如欲進一步了解 Cmdlet 的使用方式及簡單範例，可使用 Get-Help CmdLet -Full ( 或 man CmdLet -Full ) 來取得完整的指令說明，如圖二。

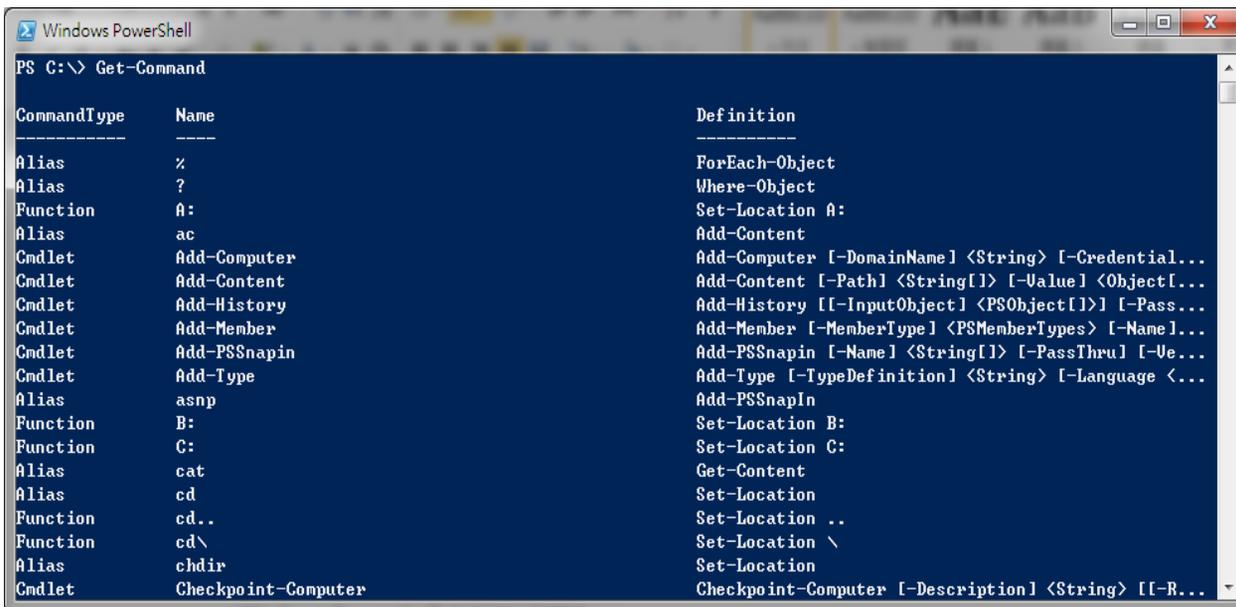


圖 一

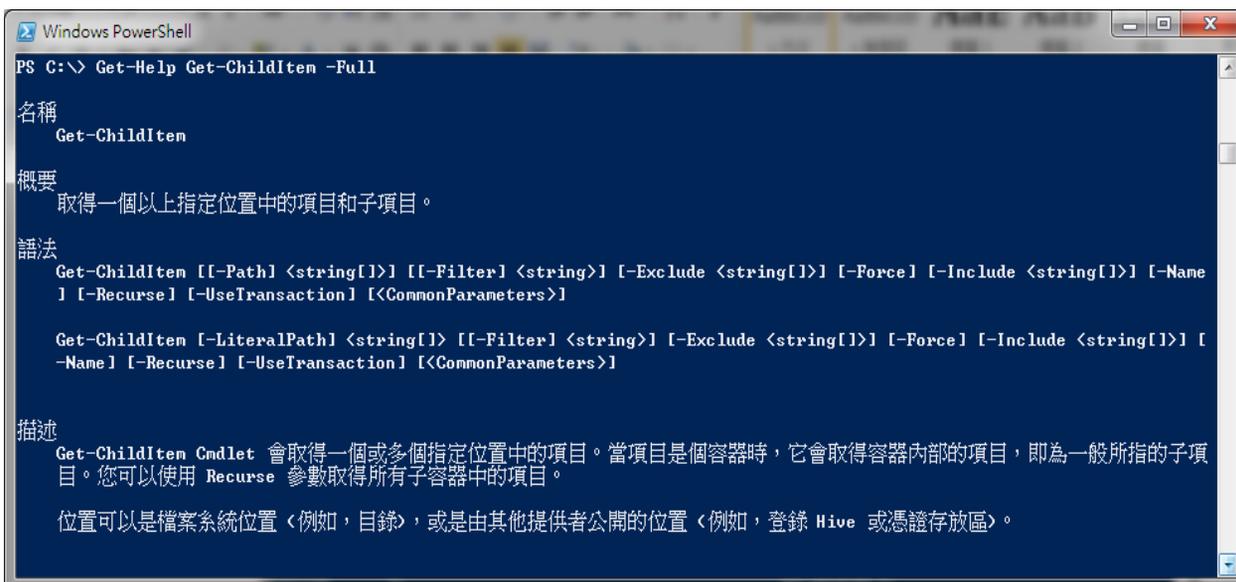


圖 二

(二) 支援現有的 Script 語言、命令列工具及多種 Windows 作業系統版本

可支援包括 Windows XP、Windows Server 2003、Windows Server 2008 及 Windows 7 等作業系統。

(三) 提供使用者以瀏覽檔案系統的方式瀏覽資料 (如登錄、憑證) 存放區的功能

以往系統管理人員要查詢登錄值，必須執行 regedit.exe 以進入圖形化管理介面或以命令列 reg 指令進行管理。在 Windows PowerShell 環境中，登錄被視同檔案系統，使用簡單的 dir 指令即可瀏覽登錄項目，並利用簡易指令取得登錄值之列表，如圖三。

(四) 可在命令列剖析複雜的運算式及操作 .NET Framework 物件

例如：利用管道 (Pipe，即“|”符號) 將物件及屬性(值)從一個 Windows PowerShell 命令輸送至另一個 Windows PowerShell 命令。管道是 Windows PowerShell 非常重要的功能，它可以將數個指令串接起來，以產出系統管理人員所要的結果。

以取得某一目錄的存取控制清單 (Access Control List，簡稱 ACL) 為例，如果直接以 Get-Acl 指令取得 ACL 列表，Access 值常會因過長而無法處理，如圖四。

此時，可將 Get-Acl 指令的輸出經由管道 (|) 轉為 Format-List 指令之輸入，就可產生較易閱讀的輸出格式，如圖五。

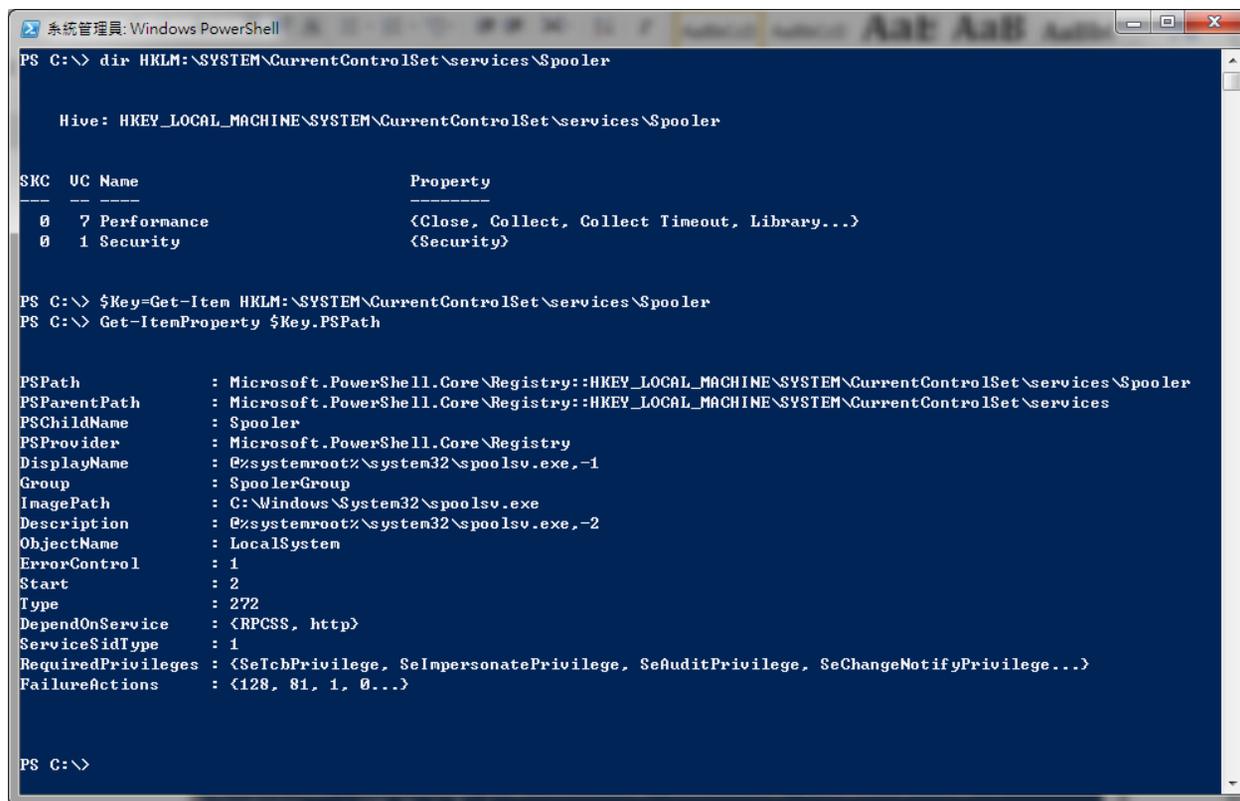
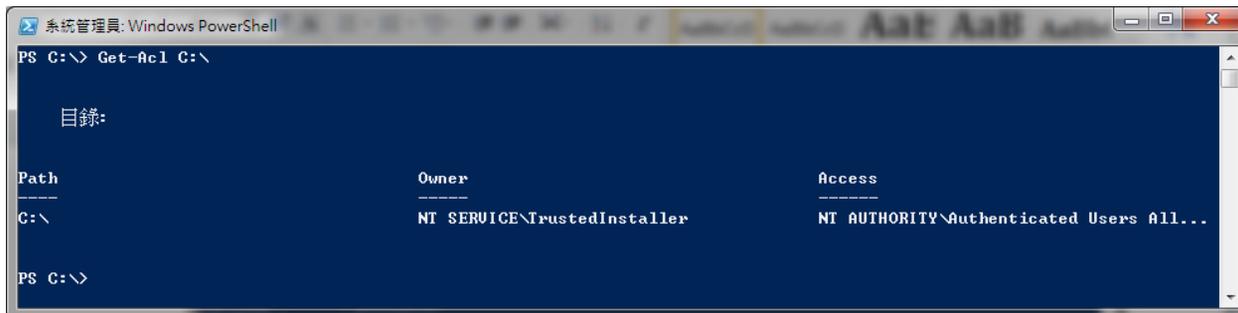


圖 三



```

PS C:\> Get-Acl C:\

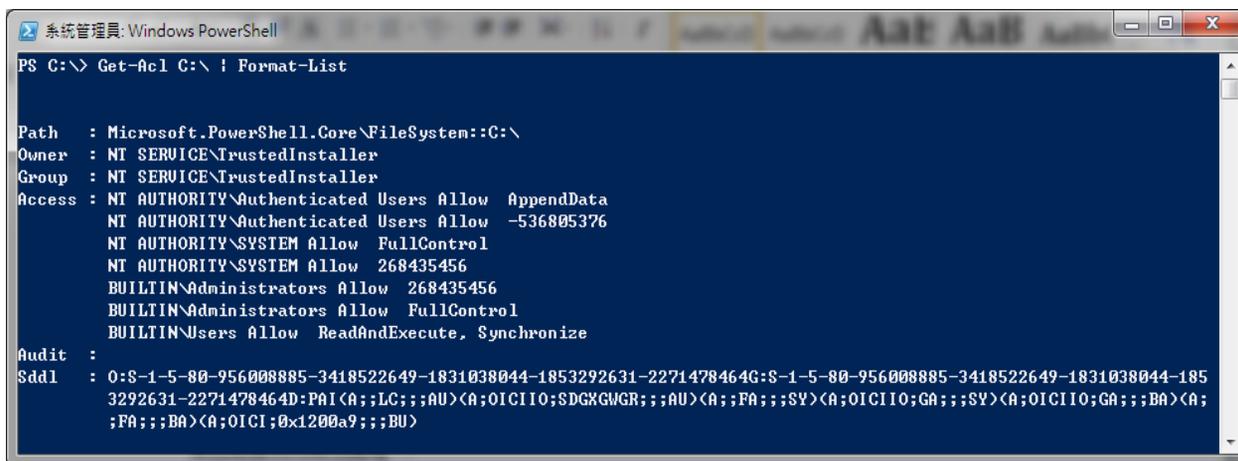
目錄:

Path                Owner                Access
----                -
C:\                  NT SERVICE\TrustedInstaller  NT AUTHORITY\Authenticated Users All...

PS C:\>

```

圖 四



```

PS C:\> Get-Acl C:\ | Format-List

Path      : Microsoft.PowerShell.Core\FileSystem::C:\
Owner     : NT SERVICE\TrustedInstaller
Group     : NT SERVICE\TrustedInstaller
Access    : NT AUTHORITY\Authenticated Users Allow AppendData
           NT AUTHORITY\Authenticated Users Allow -536805376
           NT AUTHORITY\SYSTEM Allow FullControl
           NT AUTHORITY\SYSTEM Allow 268435456
           BUILTIN\Administrators Allow 268435456
           BUILTIN\Administrators Allow FullControl
           BUILTIN\Users Allow ReadAndExecute, Synchronize
Audit     :
Sddl      : O:S-1-5-80-956008885-3418522649-1831038044-1853292631-2271478464G:S-1-5-80-956008885-3418522649-1831038044-1853292631-2271478464D:PAI(A;;LC;;;AU)(A;OICIIO;SDGKGWGR;;;AU)(A;;FA;;;SY)(A;OICIIO;GA;;;SY)(A;OICIIO;GA;;;BA)(A;;FA;;;BA)(A;OICI;0x1200a9;;;BU)

```

圖 五

### (五) 命令列工具的設計利於系統管理人員學習

除搭配標準命名慣例和一般參數，亦可使用排序、篩選和格式化資料及物件等簡易工具。

### (六) 具備完整的擴充功能

獨立軟體廠商或系統管理人員都可利用 Windows PowerShell 提供的簡易語法及豐富元件，自行擴充開發，建立符合應用程式和系統管理獨特需求的 Cmdlet。

Windows PowerShell 還利用別名 ( Alias ) 的方式，讓許多 Cmdlet 指令與對應的 UNIX 指令具有相同名稱，UNIX 使用者即使在

Windows PowerShell 環境下，仍可使用與 UNIX 環境相同的指令，大幅降低系統管理人員在作業系統轉換時，適應不同作業系統指令所需的時間。例如：在 UNIX 環境下，ls 指令可以列出目錄及檔案清單，而 Windows PowerShell 環境對應的 Cmdlet 指令是 Get-ChildItem，利用別名與 ls 連結後，UNIX 使用者在 Windows PowerShell 環境下也可使用 ls 指令達到相同目的。

系統管理人員當然可以繼續使用傳統工具，但是 Windows PowerShell 具有傳統工具無法提供的彈性及廣度，讓系統管理人員不需編譯及部署程式碼，就可以擁有近乎程式開發人員的掌控能力。

### 三、Windows PowerShell 的應用

Windows PowerShell 問世之後，微軟已逐步在各項產品中導入 Windows PowerShell 技術應用，讓管理者可以藉由 Windows PowerShell 以指令方式進行系統管理。

在作業系統面，Windows Server 2008 Server Core、Windows Server 2008、Windows 7 等作業系統上都已內建 Windows PowerShell 功能，且會隨著作業系統版本更新，提供更多 Cmdlets 供系統管理人員使用。Windows XP 及 Windows Server 2003 雖未內建 Windows PowerShell，仍可另行安裝。顯而易見，微軟企圖利用 Windows PowerShell 技術，達成統一 Windows 作業系統指令化環境的目標。

以 Windows Server 2008 Server Core 為例，微軟繼 Windows Vista 作業系統華麗的使用者介面及 3D Aero 的視窗切換功能之後，推出了顛覆 Windows 作業系統使用者傳統印象的 Windows Server 2008 Server Core，一種完全以指令化操作介面為基礎的作業系統，僅提供 Windows Server 2008 核心功能的安裝選項，系統管理人員只要搭配使用 Windows PowerShell 這項威力強大的工具，就可有效管理伺服器，不僅降低伺服器硬體本身的需求及負擔，若配合以 Windows Server 2008 建置的伺服器角色，更可提高整體資訊環境的安全性。

Windows PowerShell 也應用於微軟的系統管理軟體，例如 System Center 產品家族的 System Center Operations Manager (SCOM)、System Center Configurations Manager (SCCM)、System Center Virtual Machine Manager (SCVMM)、System Center Data Protection

Manager (DPM) 及 System Center Service Manager (SCSM) 等產品均搭配 Windows PowerShell 管理介面。

微軟陸續推出的新產品，如企業商務共同作業平台 Sharepoint 2010、雲端辦公室解決方案 Office 365 等，也都搭配 Windows PowerShell 技術應用。

### 四、結論

在 Windows 作業系統的世界裡，經過千呼萬喚，終於出現功能可與 UNIX Shell 匹敵的 Windows PowerShell 指令化環境，不僅整合了 .NET 技術，還與先前的 WSH Script 語言環境相容。目前 Windows PowerShell 版本仍持續不斷更新，所支援的功能及指令亦不斷增加。當舊版 Windows 作業系統及產品逐步淘汰，以 Windows PowerShell 為預設功能的新版作業系統及產品相繼問世，Windows PowerShell 未來勢必成為系統管理人員執行日常管理作業不可或缺的工具，系統管理人員應及早深入瞭解 Windows PowerShell 技術，以強化管理技術，提升管理效率。

●參考文獻/ 資料來源：

1. [www.microsoft.com](http://www.microsoft.com)
2. [technet.microsoft.com](http://technet.microsoft.com)
3. [support.microsoft.com](http://support.microsoft.com)