

Updated 5 July 2011

This document contains errata for *Learn Windows PowerShell in a Month of Lunches*. While tremendous effort was made to ensure the accuracy of the book, errors and typos always seem to find a way to sneak in during the publishing process. The author takes full responsibility for these, and appreciates readers bringing them to light so that they can be corrected in a future edition.

Updates to this errata are posted at least quarterly, when new errata is available. You can report typos and other errors at <http://www.manning-sandbox.com/forum.jspa?forumID=723&start=0>. Please be sure to post typos and other errata in the appropriate “errata” thread. If you have a technical question related to the book or its subject, please start a new thread with your question.

This document is posted as a companion resource to the book on <http://MoreLunches.com>, and is available on the book’s main page at <http://manning.com/jones>.

Chapter 1

Section 1.4

While the book is entirely accurate in its “Setting up your lab environment” section, there’s often confusion around Windows Server 2008 R2. This product is available from Microsoft *only in a 64-bit edition*. As of this writing, Microsoft’s Virtual PC product is incapable of running 64-bit guest operating systems; you will need to either use a VMware product like VMware Player, or use Microsoft’s Hyper-V product to run Windows Server 2008 R2 in a virtual machine. Windows Server 2008 **R2** is *not* the same as Windows Server 2008; if you think you’ve found a 32-bit version of R2, you have in fact found the older, non-R2 version, and it will not work identically to the R2 version. You will need a computer containing a 64-bit processor in order to run Windows Server 2008 R2, even if you are running it inside a virtual machine.

Chapter 8

Section 8.2

On page 86, the second PowerShell command should be:

```
PS C:\>notepad dotnettypes.format.ps1xml
```

The book incorrectly lists the file with a “.ps1ml” filename extension.

Chapter 13

Section 13.2

This section includes the following:

```
Get-Service -name BITS -computer Server1,Server2,Server3 | Start-Service -  
passthru | Get-Service
```

Unfortunately, this won't work exactly as-is unless the `-computer` parameter only specifies the *local* computer. That's because the `Start-Service` (and `Stop-Service`) cmdlets don't support remote computers (they don't have a `-computerName` parameter). This would be functional if written as follows:

```
Get-Service -name BITS | Start-Service -passthru | Get-Service
```