Pro Windows PowerShell

Hristo Deshev

Apress®

Pro Windows PowerShell

ISBN-13 (pbk): 978-1-59059-940-2

ISBN-10 (pbk): 1-59059-940-3

ISBN-13 (electronic): 978-1-4302-0546-3

ISBN-10 (electronic): 1-4302-0546-6

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Lead Editor: Tony Campbell
Technical Reviewer: Jon Rolfe
Editorial Board: Clay Andres, Steve Anglin, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Kevin Goff, Matthew Moodie, Joseph Ottinger, Jeffrey Pepper, Frank Pohlmann, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Project Manager: Sofia Marchant
Copy Editor: Heather Lang
Associate Production Director: Kari Brooks-Copony
Production Editor: Laura Cheu
Compositor: Susan Glinert Stevens
Proofreader: Linda Seifert
Indexer: Julie Grady
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit http://www.springeronline.com.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at http://www.apress.com/info/bulksales.

The source code for this book is available to readers at http://www.apress.com. You will need to answer questions pertaining to this book in order to successfully download the code.

To my wife, Yana, for all her love and support

Contents at a Glance

Contents

About the Author

HRISTO DESHEV has been a software developer, a team leader, and a product manager for Telerik, the leading vendor of user interface components (`http://www.telerik.com`). For the past several years, his focus has been on creating rich and, at the same time, robust and reliable web and desktop components running on the Microsoft .NET platform. A major goal for him and his team is to predictably deliver working software—applying agile software engineering practices to create rock-solid solutions targeting all modern web browsers and supporting multiple Visual Studio .NET versions.

Hristo is a reformed UNIX fanatic who now runs Windows and tries to apply his scripting skills and experiences to the Microsoft platform. He is also a tool freak: anything that can help automate a task—be it a scriptable utility or a full-blown scripting language—is a welcome addition to his toolbox. PowerShell has been a passion of his since its early unofficial releases due to its ease of use and almost incredible ability to automate all Windows-related tasks without needing other tools.

You can reach Hristo through his blog at `http://weblogs.asp.net/hristodeshev/`.

About the Technical Reviewer

JON ROLFE has been an IT professional for more than 12 years and is currently a senior solutions architect for one of the world's largest IT services companies. He specializes in designing secure Microsoft-based enterprise architectures for clients in the United Kingdom's public sector and large blue-chip companies. During the course of his career, he has gained extensive experience at all stages of the IT life cycle, including enterprise architecture design, software development, rapid system deployment, and large enterprise systems support. Although he specializes in Microsoft technologies, Jon also has an interest in other technologies including Linux and wrote the book *Using Ubuntu Linux* (Bernard Babani, 2007).

When not working, Jon has an active interest in digital photography, follows motor sports, and enjoys mountain biking and skiing. He can be contacted through his web site at `http://JonRolfe.com`.

Acknowledgments

I still find it hard to believe that I wrote an entire book! It has been an enjoyable project that was also quite formidable and downright hard. I would never have done it all on my own, without the huge amounts of help I got from many people.

First, I would like to thank the entire Apress team that guided me through the process: Tony Campbell for providing important input and valuable feedback on the content, Sofia Marchant for being so flexible with my sometimes late chapters, Heather Lang for her extraordinary ability to turn my often unreadable writings into well-thought-out text. Jon Rolfe deserves a special mention here; he has been an awesome technical reviewer. He not only spotted a host of problems with the sample scripts but was keen to provide solutions and improvement ideas; thus his efforts became one of the most important factors in getting this book done.

I would also like to thank the Microsoft PowerShell team for creating such a wonderful product. Being a long-time UNIX user, I have been disappointed by the primitive Windows command line shell. With PowerShell, this is no longer the case.

The PowerShell community deserves special recognition too. The wealth of information and thoughtful discussions on blogs and online forums has been an incredible inspiration when writing this book.

Last, but not least, I am personally thankful to the people behind the open source tools covered in the book: Keith Hill and the numerous contributors to the PowerShell Community Extensions project; Marc van Orsouw, the creator of the PowerTab tab completion extension; and Oisin Grehan, the man behind the PowerShell Eventing Library and many other PowerShell-related open source projects. You guys rock!

Introduction

I remember the first time I got to play with Windows PowerShell—I had some spare time on my hands, and after reading a blog post, I quickly set off to download the unofficial pre-release bits that were available at the time. I originally had very low expectations about the tool—all I wanted was a shell that is up to par with its UNIX cousins. Boy, was I wrong! The moment I learned about object pipelines, I knew this tool was going to change the way I think about automating tasks on Windows. Fast forward a couple of months, and here I am, so much in love with PowerShell that I even wrote a whopping 400-page book about it.

Why Another Book on Windows PowerShell?

As much as I like PowerShell, its plain-console looks make it very easy for people to dismiss it without really seeing what lies beneath the surface. I feel it is still largely undiscovered—I know many professionals who have yet to try it out. When administrators see PowerShell for the first time, many simply shrug, assuming it's just another version of the primitive command prompt we inherited from DOS.

The most successful way for me to recruit a new person into the ranks of happy PowerShell users is by example. When presented with an automation problem, my first suggestion is, "Let's try to do it in PowerShell." A couple of lines later, the usual reaction is, "Wow, can it really do that?" A short and sweet command can win many hearts for PowerShell. Hopefully, the examples in this book will win your heart too—it is my goal to share with you the joy of working with a modern automation environment that will change your professional life forever.

What's in This Book?

When working on the book, I separated it into sections, and this distinction may be helpful to you too.

The first section includes Chapters 1 through 14. It introduces the shell's object-oriented features and its means for providing basic abstraction like script blocks, functions, scripts, aliases, and providers. It is best to read those chapters in sequence to get a solid understanding of the advanced techniques that allow you to work with the shell's type system, get to know its security infrastructure, learn about the documentation facilities, and start using important debugging techniques.

The second section spans Chapters 15 to 21. The text uses advanced scripting techniques to work with .NET, COM, and WMI objects. This section's goal is twofold. On one hand, you'll learn how to work with external code and employ those techniques to automate almost any task on a Windows system. On the other hand, you are solving real-world problems that are very likely to land on your plate some time soon. The chapters describe how to start and stop processes

and services, work with text-based I/O, use performance counters to monitor your system, download and upload files to the web, send e-mail, automate programs via their COM interfaces, and get information and manage operating system objects through WMI. The chapters can be read in random order, and you can easily use them for reference.

The last section of the book is about tools and add-ons that you can use to extend the shell and become even more productive. Many programs can get much easier to use after extending them with a tool that provides an important missing feature. The book shows how you can use freely available open source tools to boost your productivity. An important part of my secret in winning over new people for PowerShell lies in the fact that I always do my demonstrations with PowerShell Community Extensions and PowerTab installed. Those two tools are described in Chapters 21 and 23 respectively.

Do I Need Prior Programming Experience?

The book is written by a programmer—does that mean you need to be a programmer to read it? Not at all! I have done my best to make the text easy to digest for just about anyone with a little prior scripting experience. The PowerShell scripting language bears a lot of similarities to other scripting languages, and users will be able to draw parallels between it and VBScript, JScript, batch scripts, UNIX shell scripts, and maybe even Perl or Python. People new to scripting should be able to just use any of the working code samples as they are, without needing to learn the details or any of the inner workings of the code.