



Update to Puppet 3
Cookbook

Quick answers to common problems

Puppet Cookbook

Third Edition

Jump-start your Puppet deployment using engaging and practical recipes

Thomas Uphill
John Arundel

[PACKT] open source*
PUBLISHING community experience distilled

Table of Contents

[Puppet Cookbook Third Edition](#)

[Credits](#)

[About the Authors](#)

[About the Reviewers](#)

[www.PacktPub.com](#)

[Support files, eBooks, discount offers, and more](#)

[Why Subscribe?](#)

[Free Access for Packt account holders](#)

[Preface](#)

[What this book covers](#)

[What you need for this book](#)

[Who this book is for](#)

[Conventions](#)

[Reader feedback](#)

[Customer support](#)

[Downloading the example code](#)

[Errata](#)

[Piracy](#)

[Questions](#)

[1. Puppet Language and Style](#)

[Introduction](#)

[Adding a resource to a node](#)

[How to do it...](#)

[How it works...](#)

[Using Facter to describe a node](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Variables](#)

[Scope](#)

[Installing a package before starting a service](#)

[How to do it...](#)

[How it works...](#)

[Capitalization](#)

[Learning metaparameters and ordering](#)

[Trifecta](#)

[Idempotency](#)

[Installing, configuring, and starting a service](#)

[How to do it...](#)

[How it works...](#)

[Using community Puppet style](#)

[How to do it...](#)

- [Indentation](#)
- [Quoting](#)
- [False](#)
- [Variables](#)
- [Parameters](#)
- [Symlinks](#)
- [Creating a manifest](#)
 - [How to do it...](#)
 - [There's more...](#)
- [Checking your manifests with Puppet-lint](#)
 - [Getting ready](#)
 - [How to do it...](#)
 - [There's more...](#)
 - [See also](#)
- [Using modules](#)
 - [How to do it...](#)
 - [How it works...](#)
 - [There's more...](#)
 - [Templates](#)
 - [Facts, functions, types, and providers](#)
 - [Third-party modules](#)
 - [Module organization](#)
 - [See also](#)
- [Using standard naming conventions](#)
 - [How to do it...](#)
 - [There's more...](#)
- [Using inline templates](#)
 - [How to do it...](#)
 - [How it works...](#)
 - [There's more...](#)
 - [See also](#)
- [Iterating over multiple items](#)
 - [How to do it...](#)
 - [How it works...](#)
 - [There's more...](#)
 - [Using hashes](#)
 - [Creating arrays with the split function](#)
- [Writing powerful conditional statements](#)
 - [How to do it...](#)
 - [How it works...](#)
 - [There's more...](#)
 - [Elseif branches](#)
 - [Comparisons](#)
 - [Combining expressions](#)
 - [See also](#)
- [Using regular expressions in if statements](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Capturing patterns](#)

[Regular expression syntax](#)

[See also](#)

[Using selectors and case statements](#)

[How to do it...](#)

[How it works...](#)

[Selector](#)

[Case statement](#)

[There's more...](#)

[Regular expressions](#)

[Defaults](#)

[Using the in operator](#)

[How to do it...](#)

[There's more...](#)

[Using regular expression substitutions](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using the future parser](#)

[Getting ready](#)

[How to do it...](#)

[Appending to and concatenating arrays](#)

[Lambda functions](#)

[Reduce](#)

[Filter](#)

[Map](#)

[Slice](#)

[Each](#)

[Other features](#)

[2. Puppet Infrastructure](#)

[Introduction](#)

[Installing Puppet](#)

[Getting ready](#)

[How to do it...](#)

[Managing your manifests with Git](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Creating a decentralized Puppet architecture](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)
[Writing a puppet script](#)
[How to do it...](#)
[How it works...](#)
[Running Puppet from cron](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[There's more...](#)
[Bootstrapping Puppet with bash](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[There's more...](#)
[Creating a centralized Puppet infrastructure](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[There's more...](#)
[Creating certificates with multiple DNS names](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[Running Puppet from passenger](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[There's more...](#)
[Setting up the environment](#)
[Getting ready](#)
[How to do it...](#)
[There's more...](#)
[Configuring PuppetDB](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[There's more...](#)
[Configuring Hieradata](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)
[There's more...](#)
[Setting node-specific data with Hieradata](#)
[Getting ready](#)
[How to do it...](#)
[How it works...](#)

[Storing secret data with hiera-gpg](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using MessagePack serialization](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[Automatic syntax checking with Git hooks](#)

[How to do it...](#)

[How it works...](#)

[Pushing code around with Git](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[Managing Environments with Git](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[3. Writing Better Manifests](#)

[Introduction](#)

[Using arrays of resources](#)

[How to do it...](#)

[How it works...](#)

[See also](#)

[Using resource defaults](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using defined types](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using tags](#)

[How to do it...](#)

[There's more...](#)

[Using run stages](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using roles and profiles](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Passing parameters to classes](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Specifying default values](#)

[Passing parameters from Hiera](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Writing reusable, cross-platform manifests](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Getting information about the environment](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Importing dynamic information](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Passing arguments to shell commands](#)

[How to do it...](#)

[How it works...](#)

[4. Working with Files and Packages](#)

[Introduction](#)

[Making quick edits to config files](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Editing INI style files with puppetlabs-inifile](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using Augeas to reliably edit config files](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Building config files using snippets](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[Using ERB templates](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using array iteration in templates](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using EPP templates](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using GnuPG to encrypt secrets](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Installing packages from a third-party repository](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Comparing package versions](#)

[How to do it...](#)

[How it works...](#)

[5. Users and Virtual Resources](#)

[Introduction](#)

[Using virtual resources](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Managing users with virtual resources](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Managing users' SSH access](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Managing users' customization files](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using exported resources](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[6. Managing Resources and Files](#)

[Introduction](#)

[Distributing cron jobs efficiently](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Scheduling when resources are applied](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using host resources](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using exported host resources](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using multiple file sources](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Distributing and merging directory trees](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Cleaning up old files](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Auditing resources](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Temporarily disabling resources](#)

[How to do it...](#)

[How it works...](#)

[7. Managing Applications](#)

[Introduction](#)

[Using public modules](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Managing Apache servers](#)

[How to do it...](#)

[How it works...](#)

[Creating Apache virtual hosts](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Creating nginx virtual hosts](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Managing MySQL](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Creating databases and users](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[8. Internode Coordination](#)

[Introduction](#)

[Managing firewalls with iptables](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Building high-availability services using Heartbeat](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Managing NFS servers and file shares](#)

[How to do it...](#)

[How it works...](#)

[Using HAProxy to load-balance multiple web servers](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Managing Docker with Puppet](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[9. External Tools and the Puppet Ecosystem](#)

[Introduction](#)

[Creating custom facts](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Adding external facts](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Debugging external facts](#)

[Using external facts in Puppet](#)

[See also](#)

[Setting facts as environment variables](#)

[How to do it...](#)

[Generating manifests with the Puppet resource command](#)

[How to do it...](#)

[There's more...](#)

[Generating manifests with other tools](#)

[Getting ready](#)

[How to do it...](#)

[There's more...](#)

[Using an external node classifier](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Creating your own resource types](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Documentation](#)

[Validation](#)

[Creating your own providers](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Creating custom functions](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Testing your puppet manifests with rspec-puppet](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Using librarian-puppet](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Using r10k](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[10. Monitoring, Reporting, and Troubleshooting](#)

[Introduction](#)

[Noop – the don't change anything option](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Logging command output](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Logging debug messages](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Printing out variable values](#)

[Resource ordering](#)

[Generating reports](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Other report types](#)

[See also](#)

[Producing automatic HTML documentation](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[Drawing dependency graphs](#)

[Getting ready](#)

[How to do it...](#)

[How it works...](#)

[There's more...](#)

[See also](#)

[Understanding Puppet errors](#)

[How to do it...](#)

[See also](#)

[Inspecting configuration settings](#)

[How to do it...](#)

[How it works...](#)

[See also](#)

[Index](#)

Puppet Cookbook Third Edition

Puppet Cookbook Third Edition

Copyright © 2015 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the authors, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: October 2011

Second edition: August 2013

Third edition: February 2015

Production reference: 1170215

Published by Packt Publishing Ltd.

Livery Place

35 Livery Street

Birmingham B3 2PB, UK.

ISBN 978-1-78439-488-2

www.packtpub.com

Credits

Authors

Thomas Uphill

John Arundel

Reviewers

Dhruv Ahuja

James Fryman

Jeroen Hooyberghs

Pedro Morgado

Commissioning Editor

Kartikey Pandey

Acquisition Editor

Usha Iyer

Content Development Editor

Rahul Nair

Technical Editor

Mohita Vyas

Copy Editors

Merilyn Pereira

Adithi Shetty

Project Coordinator

Judie Jose

Proofreaders

Ameesha Green

Joanna McMahon

Indexer

Tejal Soni

Graphics

Disha Haria

Abhinash Sahu

Production Coordinator

Shantanu N. Zagade

Cover Work

Shantanu N. Zagade

About the Authors

Thomas Uphill is an RHCA who has been using Puppet since 0.24. He has worked as a system administrator for almost 20 years, most recently with RHEL systems. He recently wrote *Mastering Puppet*, Packt Publishing a book for managing Puppet in large deployments. He has given tutorials on Puppet at LISA and LOPSA-East. When not at the Puppet User Group of Seattle (PUGS), he can be found at <http://ramblings.narrabilis.com>.

Thank you to John Arundel for the previous editions of this book, I had a great foundation for this rewrite. I would like to thank my wife Priya Fernandes for her support and encouragement while I was updating this book. Thanks to my fellow PUGS Andy and Justin for their suggestions. Thanks to my reviewers, Jeroen Hooyberghs, James Fryman, and Dhruv Ahuja for taking the time to find all the errors they did.

John Arundel has worked in the IT industry for most of his life, and during that time has done wrong (or seen others do wrong) almost everything that you can do wrong with computers. That comprehensive knowledge of what not to do, he feels, is one of his greatest assets as a consultant. He is still adding to it.

He spent much of his career working in very large corporations and, as a result, now likes to work with very small corporations. They like working with him too, not only because he can tell them about things that should not be done, but also because he can confidently inform them that big companies don't know what they're doing either.

Off the clock, he enjoys gardening, competitive rifle shooting, and other gentle hobbies. You can follow him on Twitter [@bitfield](#). If your company is small enough, you can hire him there too.

About the Reviewers

Dhruv Ahuja is a senior DevOps engineer at a leading financial data vendor. He specializes in software delivery optimization and infrastructure automation. He also holds a master's degree in advanced software engineering from King's College London, and won the Red Hat UK Channel Consultant of the Year award in 2012 for delivering progressive solutions. A long history in software development and systems administration equip him with aptness in both areas. In this era of infrastructure as code, he believes that declarative abstraction and accurate interfaces are key to a scalable business software lifecycle process.

James Fryman is an information technologist who builds, designs, curates, and evangelizes automation in all layers of the IT stack. Over the last decade, James has held roles in information technology that includes the domains of information security, service delivery, IT operations, IT development, and IT management. He has learned through these experiences the importance of automation in all facets of information technology to accelerate delivery, and reduce human errors throughout an application lifecycle. He is also a frequent speaker on the topic of automation at conferences throughout the world.

Now a senior DevOps engineer at StackStorm, James most recently worked at GitHub assisting in the development and curation of systems scaling within the Operations group.

My wife Melanie is awesome and deserves much more than just the words printed here.

Jeroen Hooyberghs is an open source and Linux consultant, working for Open-Future in Belgium. In this position, as well as when he was involved in Linux system administration, he has built up technical expertise in a lot of open source solutions, such as Puppet. In 2014, he became a Puppet Certified Professional and official Puppet trainer. As a reviewer, he also contributed to *Mastering Puppet*, *Thomas Uphill*, *Packt Publishing*.

Pedro Morgado holds a master's degree in informatics and computing engineering at FEUP (Faculdade de Engenharia da Universidade do Porto) and did his master's thesis on object-oriented patterns and service-oriented patterns.

Since 2009, he has been working with several different programming languages, frameworks, and technologies, which includes the main object-oriented programming languages such as PHP, Python, C/C++, Java, and JavaScript, as well as web languages such as HTML, JSON, and XML. He has worked with different database technologies such as MySQL, PostgreSQL, Oracle SQL, and SQL Server, and also with different caching systems and search engines.

He has worked as an IT consultant in the banking field for a year, and has built a recommendation system (data mining and text mining) as a research assistant at INESC (Technology and Science-Associated Laboratory) where he worked for a period of 1 year. Finally, he focused on web projects as a technical lead at Rocket Internet AG, for which he built scalable systems for FoodPanda, CupoNation, Camudi, and Lamudi. Due to his experience, he has specialized in project management and product development based in an e-commerce area. For more information, take a look at his LinkedIn account (<https://www.linkedin.com/in/pedrombmorgado>).