Contents

Foreword . . . . . **i**

Preface . . . . . **iii**
Who is this book for? iv
Typographic conventions iv
Please help improve this book! v
Current Published Book Version Information vi
About the Author vi

Introduction . . . . . **vii**
In the beginning, there were sysadmins vii
Modern infrastructure management vii
Ansible and Red Hat viii
Ansible Examples x
Other resources x

Chapter 1 - Getting Started with Ansible **1**
Ansible and Infrastructure Management 1
On snowflakes and shell scripts 1
Configuration management 2
Installing Ansible 3
Creating a basic inventory file 6
Running your first Ad-Hoc Ansible command 7
Summary 8

Chapter 2 - Local Infrastructure Development: Ansible and Vagrant **9**
Prototyping and testing with local virtual machines 9
Your first local server: Setting up Vagrant 10

Using Ansible with Vagrant 11
Your first Ansible playbook 12
Cleaning Up 16
Summary 16

Chapter 3 - Ad-Hoc Commands 17
Conducting an orchestra 17
Build infrastructure with Vagrant for testing 18
Inventory file for multiple servers 20
Your first ad-hoc commands 22
Discover Ansible's parallel nature 22
Learning about your environment 24
Make changes using Ansible modules 27
Configure groups of servers, or individual servers 28
Configure the Application servers 29
Configure the Database servers 30
Make changes to just one server 31
Manage users and groups 32
Manage packages 33
Manage files and directories 34
Get information about a file 34
Copy a file to the servers 34
Retrieve a file from the servers 35
Create directories and files 35
Delete directories and files 36
Run operations in the background 36
Update servers asynchronously with asynchronous jobs 37
Check log files 38
Manage cron jobs 40
Deploy a version-controlled application 41
Ansible's SSH connection history 42
Paramiko 42
OpenSSH (default) 43
Faster OpenSSH with Pipelining 43
Summary 44

Chapter 4 - Ansible Playbooks **45**
Power plays 45
Running Playbooks with `ansible-playbook` 50
Limiting playbooks to particular hosts and groups 50
Setting user and sudo options with `ansible-playbook` 51
Other options for `ansible-playbook` 52
Real-world playbook: CentOS Node.js app server 53
Add extra repositories 55
Deploy a Node.js app 58
Launch a Node.js app 60
Node.js app server summary 61
Real-world playbook: Ubuntu LAMP server with Drupal 62
Include a variables file, and discover `pre_tasks` and `handlers` 62
Basic LAMP server setup 64
Configure Apache 66
Configure PHP with `lineinfile` 68
Configure MySQL 69
Install Composer and Drush 70
Install Drupal with Git and Drush 73
Drupal LAMP server summary 75
Real-world playbook: Ubuntu server with Solr 76
Include a variables file, and more `pre_tasks` 76
Install Java 8 77
Install Apache Solr 78
Apache Solr server summary 80
Summary 81

Chapter 5 - Ansible Playbooks - Beyond the Basics **82**
Handlers 82
Environment variables 84
Per-play environment variables 85
Variables 87
Playbook Variables 87
Inventory variables 89
Registered Variables 90
Accessing Variables 91

Host and Group variables . . . 93
Automatically-loaded `group_vars` and `host_vars` . . . 94
Magic variables with host and group variables and information . 95
Facts (Variables derived from system information) . . . 96
Local Facts (Facts.d) . . . 97
Ansible Vault - Keeping secrets secret . . . 99
Variable Precedence . . . 103
If/then/when - Conditionals . . . 104
Jinja Expressions, Python built-ins, and Logic . . . 104
`register` . . . 106
`when` . . . 107
`changed_when` and `failed_when` . . . 109
`ignore_errors` . . . 110
Delegation, Local Actions, and Pauses . . . 110
Pausing playbook execution with `wait_for` . . . 111
Running an entire playbook locally . . . 112
Prompts . . . 113
Tags . . . 114
Blocks . . . 116
Summary . . . 118

Chapter 6 - Playbook Organization - Roles, Includes, and Imports . . . 120
Imports . . . 120
Includes . . . 122
Dynamic includes . . . 123
Handler imports and includes . . . 124
Playbook imports . . . 124
Complete includes example . . . 125
Roles . . . 128
Role scaffolding . . . 128
Building your first role . . . 129
More flexibility with role vars and defaults . . . 131
Other role parts: handlers, files, and templates . . . 134
Handlers . . . 134
Files and Templates . . . 134
Organizing more complex and cross-platform roles . . . 135

Ansible Galaxy . . . 138
Getting roles from Galaxy . . . 138
Using role requirements files to manage dependencies . . . 139
A LAMP server in nine lines of YAML . . . 140
A Solr server in seven lines of YAML . . . 142
Helpful Galaxy commands . . . 143
Contributing to Ansible Galaxy . . . 143
Summary . . . 143

Chapter 7 - Inventories . . . 145
A real-world web application server inventory . . . 146
Non-prod environments, separate inventory files . . . 150
Inventory variables . . . 151
`host_vars` . . . 153
`group_vars` . . . 154
Ephemeral infrastructure: Dynamic inventory . . . 155
Dynamic inventory with DigitalOcean . . . 156
DigitalOcean account prerequisites . . . 156
Connecting to your DigitalOcean account . . . 157
Creating a droplet with Ansible . . . 157
DigitalOcean dynamic inventory with `digital_ocean.py` . . . 163
Dynamic inventory with AWS . . . 164
Inventory on-the-fly: `add_host` and `group_by` . . . 165
Multiple inventory sources - mixing static and dynamic inventories . 166
Creating custom dynamic inventories . . . 166
Building a Custom Dynamic Inventory in Python . . . 168
Building a Custom Dynamic Inventory in PHP . . . 173
Managing a PaaS with a Custom Dynamic Inventory . . . 176
Summary . . . 177

Chapter 8 - Ansible Cookbooks . . . 178
Highly-Available Infrastructure with Ansible . . . 178
Directory Structure . . . 180
Individual Server Playbooks . . . 180
Main Playbook for Configuring All Servers . . . 192
Getting the required roles . . . 192

Vagrantfile for Local Infrastructure via VirtualBox . . . 193
Provisioner Configuration: DigitalOcean . . . 198
Provisioner Configuration: Amazon Web Services (EC2) . . . 203
Summary . . . 209
ELK Logging with Ansible . . . 210
ELK Playbook . . . 211
Forwarding Logs from Other Servers . . . 217
Summary . . . 223
GlusterFS Distributed File System Configuration with Ansible . . . 224
Configuring Gluster - Basic Overview . . . 225
Configuring Gluster with Ansible . . . 226
Summary . . . 233
Mac Provisioning with Ansible and Homebrew . . . 233
Running Ansible playbooks locally . . . 234
Automating Homebrew package and app management . . . 234
Configuring Mac OS X through dotfiles . . . 236
Summary . . . 238

Chapter 9 - Deployments with Ansible . . . 239
Deployment strategies . . . 239
Simple single-server deployments . . . 240
Provisioning a Ruby on Rails server . . . 241
Deploying a Rails app to the server . . . 244
Provisioning and Deploying the Rails App . . . 249
Deploying application updates . . . 251
Zero-downtime multi-server deployments . . . 254
Ensuring zero downtime with `serial` and integration tests . . . 263
Deploying to app servers behind a load balancer . . . 265
Capistrano-style and blue-green deployments . . . 272
Additional Deployment Features . . . 274
Summary . . . 275

Chapter 10 - Server Security and Ansible . . . 276
A brief history of SSH and remote access . . . 276
Telnet . . . 277
rlogin, rsh and rcp . . . 278

SSH 279
The evolution of SSH and the future of remote access 281
Use secure and encrypted communication 282
Disable root login and use `sudo` 284
Remove unused software, open only required ports 286
Use the principle of least privilege 286
User account configuration 287
File permissions 287
Update the OS and installed software 289
Automating updates 289
Automating updates for RHEL systems 290
Automating updates for Debian-based systems 290
Use a properly-configured firewall 292
Configuring a firewall with `ufw` on Debian or Ubuntu 292
Configuring a firewall with `firewalld` on RHEL, Fedora, or CentOS . 294
Make sure log files are populated and rotated 295
Monitor logins and block suspect IP addresses 296
Use SELinux (Security-Enhanced Linux) or AppArmor 297
Summary and further reading 299

Chapter 11 - Automating Your Automation - Ansible Tower and CI/CD . . 301
Ansible Tower 301
Getting and Installing Ansible Tower 302
Using Ansible Tower 304
Other Tower Features of Note 307
Tower Alternatives 307
Jenkins CI 308
Build a local Jenkins server with Ansible 308
Create an Ansible playbook on the Jenkins server 311
Create a Jenkins job to run an Ansible Playbook 311
Unit, Integration, and Functional Testing 313
Debugging and Asserting 314
The `debug` module 314
The `fail` and `assert` modules 316
Checking syntax and performing dry runs 317
Automated testing on GitHub using Travis CI 318

Testing on multiple OSes with Docker . . . 319
Setting up the test . . . 320
Building Docker containers in Travis . . . 322
Testing the role's syntax . . . 325
Role success - first run . . . 326
Role idempotence . . . 326
Role success - final result . . . 327
Some notes about Travis CI . . . 328
Real-world examples . . . 329
Functional testing using serverspec . . . 329
Other server and role testing tools . . . 330
Summary . . . 331

Chapter 12 - Automating HTTPS and TLS Certificates . . . 332
Generating Self-Signed Certificates with Ansible . . . 332
Idempotent Nginx HTTPS playbook with a self-signed cert . . . 334
Automating Let's Encrypt with Ansible for free Certs . . . 342
Use Galaxy roles to get things done faster . . . 342
Create the playbook . . . 343
Create a server and configure DNS . . . 349
Point the playbook inventory at the server . . . 350
Access your server over HTTPS! . . . 350
Configuring Nginx to proxy HTTP traffic and serve it over HTTPS . . . 351
Modify the Nginx configuration to proxy traffic . . . 352
Summary . . . 355

Chapter 13 - Docker and Ansible . . . 356
A brief introduction to Docker containers . . . 356
Using Ansible to build and manage containers . . . 358
Building a Flask app with Ansible and Docker . . . 360
Data storage container . . . 366
Flask container . . . 367
MySQL container . . . 372
Ship it! . . . 373
Building containers with Ansible from the outside . . . 373
Build a Hubot Slack bot container with `ansible_connection: docker` 374

Hubot and Slack . . . 374
Building a Docker container with Ansible . . . 375
Building the `hubot-slack` role . . . 378
Building and running the Hubot Slack bot container . . . 380
Summary . . . 381
Summary . . . 382

Chapter 14 - Kubernetes and Ansible . . . 383
A bit of Kubernetes history . . . 383
Evaluating the need for Kubernetes . . . 384
Building a Kubernetes cluster with Ansible . . . 385
Managing Kubernetes with Ansible . . . 393
Ansible's `k8s` module . . . 394
Managing Kubernetes Applications with Helm . . . 400
Interacting with Pods using the `kubectl` connection plugin . . . 405
Summary . . . 407

Afterword . . . 409

Appendix A - Using Ansible on Windows workstations . . . 410
Method 1 - Use the Windows Subsystem for Linux / Bash on Ubuntu . . . 410
Installing Ansible inside Bash on Ubuntu . . . 411
Method 2 - When WSL is not an option . . . 412
Prerequisites . . . 413
Set up an Ubuntu Linux Virtual Machine . . . 413
Log into the Virtual Machine . . . 414
Install Ansible . . . 416
Summary . . . 417

Appendix B - Ansible Best Practices and Conventions . . . 418
Playbook Organization . . . 418
Write comments and use `name` liberally . . . 418
Include related variables and tasks . . . 419
Use Roles to bundle logical groupings of configuration . . . 420
Use role defaults and vars correctly . . . 421
YAML Conventions and Best Practices . . . 422
YAML for Ansible tasks . . . 423

Three ways to format Ansible tasks 424
Shorthand/one-line (`key=value`) 424
Structured map/multi-line (`key:value`) 425
Folded scalars/multi-line (`>`) 426
Using `|` to format multiline variables 427
Using `ansible-playbook` 427
Use Ansible Tower 428
Specify `--forks` for playbooks running on > 5 servers 428
Use Ansible's Configuration file 428
Summary 429