异步图书
www.epubit.com

Packt

Hands-On
Enterprise Automation with Python

Python
自动化运维实战

[美] 巴塞姆 · 阿利（Bassem Aly）著
王文峰 袁洪艳 译

系统介绍如何通过Python模块、库与工具自动配置和管理大量服务器，提高运维的效率

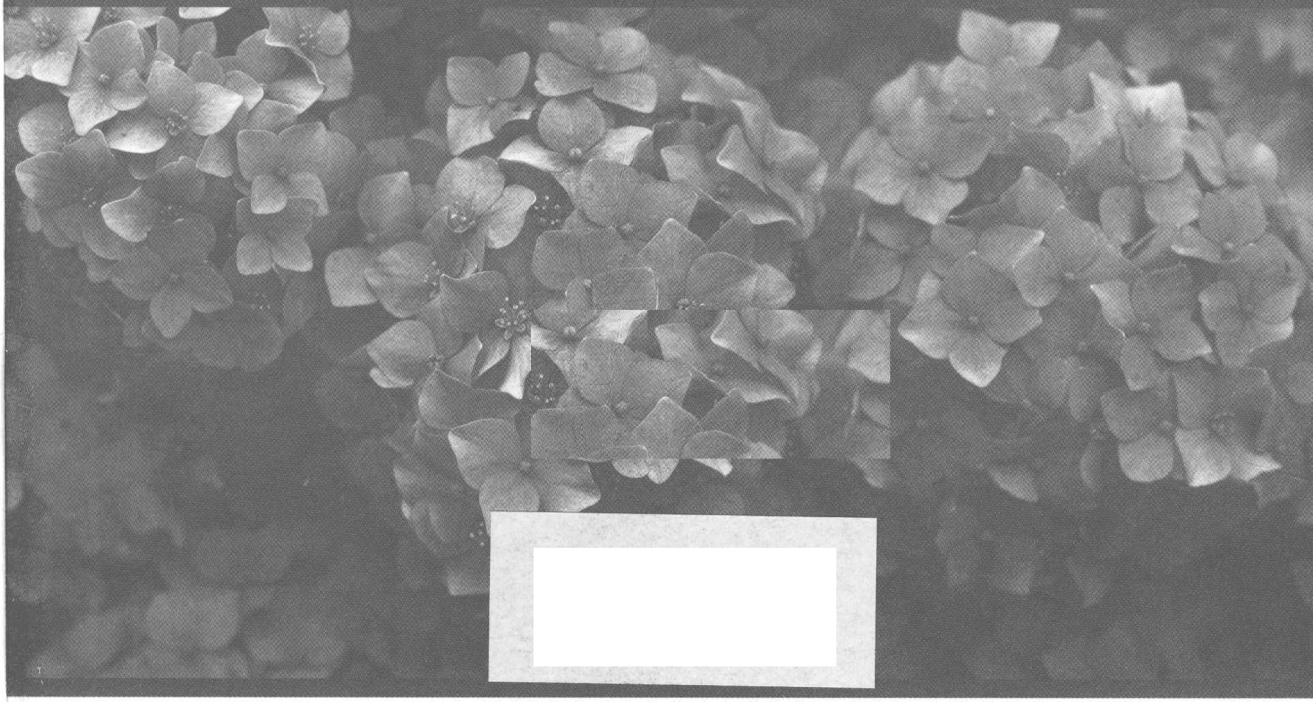


中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

Packt>

Python 自动化运维实战

[美] 巴塞姆 · 阿利（Bassem Aly）著
王文峰 袁洪艳 译



人民邮电出版社
北京

内 容 提 要

本书介绍了如何通过 Python 来自动完成服务器的配置与管理，自动完成系统的管理任务（如用户管理、数据库管理和进程管理），以及完成这些工作所需的模块、库和工具。此外，本书还讲述了如何使用 Python 脚本自动执行测试，如何通过 Python 在云基础设施和虚拟机上自动执行任务，如何使用基于 Python 的安全工具自动完成与安全相关的任务。

本书适合运维人员和开发人员阅读，也可作为相关专业人士的参考书。

作 者 简 介

Bassem Aly 是 Juniper Networks 公司经验丰富的 SDN/NFV 解决方案顾问，过去 9 年来一直在电信行业工作。他擅长使用不同的自动化工具以及 DevOps 框架设计和实现下一代自动化解决方案。此外，他在使用 OpenStack 构建和部署电信应用程序方面拥有丰富的经验，同时他还负责网络自动化和网络编程方面的企业培训。

技术审稿人简介

Jere Julian 是一名高级网络自动化工程师，在网络自动化方面拥有近 20 年的工作经验，当前的研究方向是自动化运维。过去几年，他还在 DevOps Days 和 Interop ITX 会议上发言，同时定期为网络计算技术的发展贡献自己的力量。他与妻子和两个儿子一起住在北卡罗来纳州。在 Twitter 上通过@julianje 能联系到他。

前　言

本书从 Python 安装环境开始，一直介绍到如何使用 Python 实现自动化任务，同时会介绍所用到的模块（module）、库（library）和工具。

本书不仅会介绍如何使用简单的 Python 程序自动管理系统和网络，还将分析如何使用 Python 库、模块、工具自动管理系统。另外，通过一些系统管理任务（例如，用户管理、数据库管理，以及进程管理），介绍如何配置和管理服务器。随着内容的深入，你将能够使用 Python 脚本自动完成一些测试服务，在虚拟机和云基础设施上使用 Python 完成自动化任务。在最后几章中，你将接触到基于 Python 的安全工具并学会如何自动完成安全任务。

读完本书，你将学到如何使用 Python 自动完成多种系统管理任务。



在 WordPress 网站上搜索 basimaly，即可查看作者的博客。

本书读者对象

本书适合那些希望替换主流自动化框架（如 Puppet 和 Chef）的系统管理员、开发人员、运维人员阅读。阅读本书需要具备 Python 以及 Linux shell 脚本编程基础的知识。

本书内容

第 1 章探讨如何下载、安装 Python 解释器和 Python 集成开发环境(Integrated Development Environment，IDE ）——JetBrains PyCharm。IDE 为我们提供智能的自动补全、智能的代码分析、强大的重构功能，并集成了 Git、virtualenv、Vagrant 和 Docker。这些功能和工具将帮助我们编写专业的、功能强大的 Python 代码。

第 2 章涵盖当前可用的与自动化相关的 Python 库。我们将根据用途（系统、网络和云）对它们进行分类，并分别介绍。随着内容的继续深入，你会深入了解每一个库，详细了解它们的用法。

第 3 章讨论网络自动化的优点以及网络运营商如何自动管理当前的设备。该章展示当前用来实现 Cisco、Juniper 和 Arista 自动化的网络节点的主流库。该章介绍如何搭建网络实验室，用来运行 Python 脚本。我们会用到一款开源网络仿真工具 EVE-NG。

第 4 章介绍如何使用 netmiko、Paramiko 和 telnetlib 通过 Telnet 与 SSH 建立连接、管理网络设备。该章讲述如何编写 Python 代码来访问交换机和路由器，并在终端上执行命令，然后返回输出。该章还讨论如何利用不同的 Python 技术进行备份和推送配置。该章结尾给出现代网络环境中的一些用例。

第 5 章首先介绍如何在 Python 中使用不同的工具和方法，从返回的输出结果中提取有用的数据并对数据进行操作，然后讨论如何使用 CiscoConfParse 库来审核配置，最后讲述如何使用 Matplotlib 可视化数据，生成图形和报告。

第 6 章介绍对拥有数百个网络节点的网络生成通用配置的方法，即创建模板，并使用模板语言 Jinja2 来生成一个配置模板。

第 7 章讲述如何实例化和并行执行 Python 代码。只要这些脚本之间没有相互依赖，并行运行能够更快地完成自动化工作流。

第 8 章介绍虚拟实验室环境的安装过程和准备工作。该章不仅介绍如何在 CentOS 或 Ubuntu 系统上使用不同的虚拟机软件安装自动化服务器，还讨论如何使用 Cobbler 自动安装操作系统。

第 9 章介绍如何将命令从 Python 脚本直接发送到操作系统 shell 并分析返回的输出内容。

第 10 章介绍如何使用 Fabric 执行系统管理任务。Fabric 是一个通过 SSH 执行系统管理任务的 Python 库。Fabric 还可以用来部署大型应用程序。该章还介绍如何用 Fabric 在远程服务器上执行任务。

第 11 章介绍如何生成系统报告、管理用户和监控系统。从系统中收集数据并定期生成报告是每一个系统管理员的基本任务。自动执行这项任务能够帮助我们尽早发现问题，及时准备解决方案。该章将用到一些经过实践检验的方法，从服务器上自动收集数据并生成正式的报告。该章还介绍如何使用 Python 与 Ansible 管理新用户和现有用户。此外，该章还深入讨论如何监控系统 KPI 和分析日志。同时，还可以定期运行监控脚本并将结果发送到指定的邮箱中。

第 12 章介绍如何与数据库进行交互。如果你是数据库管理员或数据库开发人员，那么 Python 提供了大量的库和模块，它们支持主流的数据库管理系统（Database Management System，DBMS），如 MySQL、Postgress 和 Oracle。该章介绍如何使用 Python 连接器与数据库管理系统进行交互。

第 13 章介绍一款强大的配置管理软件 Ansible。Ansible 在系统管理方面功能非常强大，能够同时准确地对数百甚至数千台服务器进行相同的配置。

第 14 章介绍如何在 VMware 虚拟机管理程序（hypervisor）上自动创建虚拟机。我们将在 ESXi 上使用 VMware 官方提供的库，使用几种不同的方法来创建和管理虚拟机。

第 15 章解释为什么使用 OpenStack 在私有云上创建私有 IaaS。我们将使用 Python 模块（如 requests）通过 REST 接口与 OpenStack 中的一些服务（如 nova、cinder 和 neutron）进行交互，并通过 OpenStack 创建资源。在该章的后半部分，我们将使用 Ansible playbook 再执行一次同样的操作。

第 16 章介绍如何使用亚马逊官方的 Boto3 自动处理常见的 AWS 服务，如 EC2 和 S3。Boto3 为我们访问亚马逊相关的服务提供了一个非常好用的 API。

第 17 章介绍一个功能强大的 Python 工具——Scapy，它能够根据需要组建数据包（packet）[①]，并通过网络将报文发送出去。使用 Scapy 可以创造出任意类型的数据包，并可将它发送到网络上。Scapy 还可以抓取报文并将在网络上重播。

第 18 章展示使用 Python 编写网络扫描程序的完整例子。使用这个例子，可以扫描整个子网，查找子网中开放的各种协议和端口，并对扫描到的每个主机生成一份报告。最后该章

① 除非特别强调，为行文方便，本书将不加区分地使用数据包、报文、网络数据报文这些名词。——译者注

展示如何使用 Git 在开源社区（GitHub）上共享代码。

如何充分利用本书

读者应该熟悉基本的 Python 编程范例，并且具备 Linux 系统和 Linux shell 脚本的基础知识。

下载示例代码

利用 packtpub 网站上的账户可以下载本书中所有示例的代码。如果你在其他地方购买了本书，访问 packtpub 网站，注册之后，示例代码将直接通过电子邮件发送给你。

按照下列步骤下载代码。

（1）在 packtpub 网站上登录或注册。

（2）选择 **SUPPORT** 选项卡。

（3）单击 **Code Downloads & Errata**。

（4）在 **Search** 框中输入本书的名字，然后根据屏幕提示进行操作。

从下列软件中选择一个，解压缩下载的文件。注意，使用以下最新版本的解压缩软件：

- Windows 版的 WinRAR/7-Zip；
- Mac 版的 Zipeg/iZip/UnRarX；
- Linux 版的 7-Zip/PeaZip。

本书示例代码也托管在 GitHub 上，在该网站上搜索 Hands-On Enterprise Automation with Python 即可查看源代码。任何更新都将同步到 GitHub 上。

版式约定

为方便阅读，本书中使用约定的版式来表示不同的内容。

整段代码通常以下列形式表示。

```
from netmiko import ConnectHandler
```

```
from devices import R1,SW1,SW2,SW3,SW4

nodes = [R1,SW1,SW2,SW3,SW4]

for device in nodes:
    net_connect = ConnectHandler(**device)
    output = net_connect.send_command("show run")
    print output
```

当我们希望读者注意代码段的某部分时，相应的行或文字将以粗体显示，如下所示。

```
hostname {{hostname}}
```

任何命令行的输入或输出都会使用下面的格式。

```
pip install jinja2
```



表示警告或重要说明。



表示提示和技巧。

保持联络

欢迎读者对我们提出宝贵意见和建议。

建议：发送电子邮件至 feedback@packtpub.com，并在邮件主题中注明书名。如果你对本书的任何方面有疑问，请发送电子邮件至 questions@packtpub.com。

错误反馈：我们将尽力确保本书内容的准确性，但仍然不能避免所有问题。如果你发现了本书的任何错误，请访问 packtpub 网站，选择图书名称，单击 Errata Submission Form 链接，然后输入详细信息。

盗版问题：如果在互联网上发现任何关于 Packt 图书的非法版本，欢迎提供地址或网站名称，我们将不胜感激。请通过 copyright@packtpub.com 与我们联系，告诉我们相关材料的链接。

投稿：如果你关于某个主题想写书，请访问 packtpub 网站。

评论

欢迎评论。阅读本书后，读者可以在网站上发表评论。其他读者可以查看并根据你的意见决定是否购买本书，同时 Packt 还可以了解你对相关图书的看法，作者也可以看到你的反馈。

有关 Packt 的更多信息，请访问 packtpub 网站。

致谢

感谢我的妻子 Sarah 和女儿 Mariam。为了撰写本书我很少陪伴她们，谢谢她们对我的包容。我希望有一天 Mariam 会读到这本书，并能够理解为什么我花了这么多时间伏案工作而没能陪她一起玩耍。感谢我父母的鼓励，正是这些鼓励成就了今天的我。最后，感谢我的导师 Ashraf Albasti，他通过各种方式在我的职业生涯中帮助我。

资源与支持

本书由异步社区出品，社区（https://www.epubit.com/）为您提供相关资源和后续服务。

配套资源

本书配套资源包括相关示例的源代码。

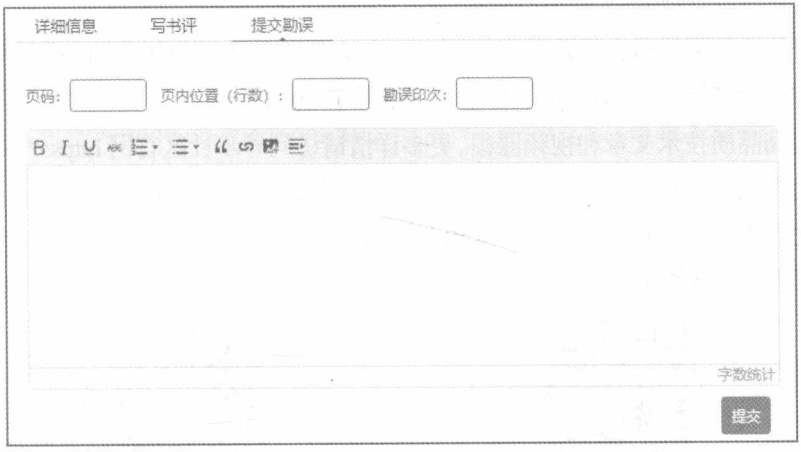
要获得源代码，请在异步社区本书页面中单击 配套资源，跳转到下载界面，按提示进行操作即可。注意，为保证购书读者的权益，该操作会给出相关提示，要求输入提取码进行验证。

如果您是教师，希望获得教学配套资源，请在社区本书页面中直接联系本书的责任编辑。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，按书名搜索，进入本书页面，单击“提交勘误”，输入勘误信息，单击“提交”按钮即可（见下图）。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的 100 积分。积分可用于在异步社区兑换优惠券、样书或奖品。



扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。



与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请您发邮件给我们，并请在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们；有意出版图书的作者也可以到异步社区在线提交投稿（直接访问 www.epubit.com/selfpublish/submission 即可）。

如果您所在学校、培训机构或企业想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接通过邮件发送给我们。您的这一举动是对作者权益的保护，也是我们持续为您提供有价值的内容的动力之源。

关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务。异步社区创办于 2015 年 8 月，提供大量精品 IT 技术图书和电子书，以及高品质技术文章和视频课程。更多详情请访问异步社区官网 https://www.epubit.com。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术等。



异步社区



微信服务号

目　录

第 1 章　搭建 Python 环境 …… 1
1.1　Python 简介 …… 2
1.1.1　Python 版本 …… 3
1.1.2　安装 Python …… 4
1.2　安装 PyCharm IDE …… 6
1.3　PyCharm 的高级功能 …… 12
1.3.1　调试代码 …… 13
1.3.2　重构代码 …… 14
1.3.3　从 GUI 安装包 …… 16
1.4　小结 …… 17
第 2 章　常用的自动化库 …… 18
2.1　Python 包 …… 19
2.2　常用 Python 库 …… 20
2.2.1　与网络相关的 Python 库 … 21
2.2.2　与系统和云相关的 Python 库 …… 22
2.3　查看模块源代码 …… 23
2.4　小结 …… 28
第 3 章　搭建网络实验室环境 …… 29
3.1　技术要求 …… 30
3.2　需要自动化网络的时间和原因 …… 30
3.3　自动化的两种方式——屏幕抓取与 API …… 31
3.4　使用 Python 进行网络自动化的原因 …… 31
3.5　网络自动化的未来 …… 33
3.6　搭建网络实验室 …… 34
3.7　准备工作——安装 EVE-NG …… 34
3.7.1　在 VMware Workstation 上安装 …… 35
3.7.2　通过 VMware ESXi 安装 …… 36
3.7.3　通过 Red Hat KVM 安装 …… 38
3.7.4　访问 EVE-NG …… 39
3.7.5　安装 EVE-NG 客户端工具包 …… 42
3.7.6　在 EVE-NG 中加载网络镜像 …… 43

3.8 创建企业网络拓扑 …… 43
3.8.1 添加新节点 …… 44
3.8.2 连接节点 …… 45
3.9 小结 …… 47
第 4 章 使用 Python 管理网络设备 …… 48
4.1 技术要求 …… 49
4.1.1 Python 和 SSH …… 49
4.1.2 Paramiko 模块 …… 50
4.1.3 netmiko 模块 …… 52
4.2 在 Python 中使用 Telnet 协议 …… 59
4.3 使用 netaddr 处理 IP 地址和网络 …… 64
4.3.1 安装 netaddr …… 65
4.3.2 使用 netaddr 的方法 …… 65
4.4 简单的用例 …… 67
4.4.1 备份设备配置 …… 68
4.4.2 创建访问终端 …… 70
4.4.3 从 Excel 工作表中读取数据 …… 72
4.4.4 其他用例 …… 75
4.5 小结 …… 75
第 5 章 从网络设备中提取数据 …… 76
5.1 技术要求 …… 77
5.2 解释器 …… 77
5.3 正则表达式 …… 78
5.4 使用 CiscoConfParse 库校验配置 …… 86
5.4.1 CiscoConfParse 库 …… 86
5.4.2 支持的供应商 …… 87
5.4.3 安装 CiscoConfParse 库 …… 88
5.4.4 使用 CiscoConfParse 库 …… 88
5.5 使用 Matplotlib 库可视化返回的数据 …… 91
5.5.1 安装 Matplotlib 库 …… 91
5.5.2 使用 Matplotlib 库 …… 92
5.5.3 使用 Matplotlib 库可视化 SNMP …… 94
5.6 小结 …… 96
第 6 章 使用 Python 和 Jinja2 配置生成器 …… 97
6.1 YAML 简介 …… 98
6.2 使用 Jinja2 建立配置模板 …… 102
6.2.1 从文件系统中读取模板 …… 109
6.2.2 在 Jinja2 中使用循环和条件 …… 111
6.3 小结 …… 119
第 7 章 并行执行 Python 脚本 …… 120
7.1 Python 脚本在计算机中运行的方式 …… 121
7.2 multiprocessing 库 …… 123
7.2.1 开始使用 multiprocessing 库 …… 123
7.2.2 进程间的相互通信 …… 126
7.3 小结 …… 127
第 8 章 准备实验室环境 …… 128
8.1 获取 Linux 操作系统 …… 129
8.1.1 下载 CentOS …… 129
8.1.2 下载 Ubuntu …… 130
8.2 在虚拟机管理程序上创建自动化虚拟机 …… 131
8.2.1 在 VMware ESXi 上创建 Linux 虚拟机 …… 131

8.2.2 使用 KVM 创建 Linux 虚拟机 …… 135
8.3 开始使用 Cobbler …… 139
8.3.1 Cobbler 的工作原理 …… 139
8.3.2 在自动化服务器上安装 Cobbler …… 141
8.3.3 通过 Cobbler 检查服务器硬件 …… 144
8.4 小结 …… 149
第 9 章 使用 subprocess 库 …… 150
9.1 subprocess 库中的 Popen() …… 151
9.2 stdin、stdout 和 stderr …… 154
9.3 subprocess 库中的 call()函数 …… 156
9.4 小结 …… 157
第 10 章 使用 Fabric 运行系统管理任务 …… 158
10.1 技术要求 …… 159
10.2 Fabric 库 …… 159
10.2.1 安装 Fabric 库 …… 160
10.2.2 Fabric 库中的操作 …… 161
10.3 运行第一个 Fabric 文件 …… 164
10.3.1 有关 fab 工具的更多信息 …… 167
10.3.2 使用 Fabric 检查系统健康状态 …… 168
10.4 其他有用的 Fabric 特性 …… 173
10.4.1 Fabric 角色 …… 173
10.4.2 Fabric 上下文管理器 …… 175
10.5 小结 …… 176
第 11 章 生成系统报告和监控系统 …… 177
11.1 从 Linux 系统中收集数据 …… 178
11.1.1 通过邮件发送收集的数据 …… 183
11.1.2 使用 time 和 date 模块 …… 186
11.1.3 定期运行脚本 …… 187
11.2 在 Ansible 中管理用户 …… 188
11.2.1 在 Linux 系统中通过 Ansible 管理用户 …… 188
11.2.2 在 Windows 系统中通过 Ansible 管理用户 …… 190
11.3 小结 …… 190
第 12 章 与数据库交互 …… 191
12.1 在自动化服务器上安装 MySQL …… 192
12.1.1 安装后的安全问题 …… 193
12.1.2 验证数据库的安装 …… 194
12.2 从 Python 中访问 MySQL 数据库 …… 195
12.2.1 查询数据库 …… 197
12.2.2 向数据库中插入数据 …… 198
12.3 小结 …… 201
第 13 章 使用 Ansible 管理系统 …… 202
13.1 Ansible 术语 …… 203
13.2 在 Linux 系统上安装 Ansible …… 205
13.2.1 在 RHEL 系统和 CentOS 上安装 Ansible …… 205
13.2.2 在 Ubuntu 系统上安装 Ansible …… 205
13.3 在即席模式下使用 Ansible …… 206
13.4 创建第一个 playbook …… 210

13.5 Ansible 的条件、处理程序和循环……212
13.5.1 设计条件……213
13.5.2 在 Ansible 中创建循环……215
13.5.3 使用处理程序触发任务……216
13.6 使用事实数据……218
13.7 使用 Ansible 模板……219
13.8 小结……221
第 14 章 创建和管理 VMware 虚拟机……222
14.1 设置环境……223
14.2 使用 Jinja2 生成 VMX 文件…225
14.2.1 创建 VMX 模板……226
14.2.2 处理 Excel 工作表中的数据……229
14.2.3 生成 VMX 文件……231
14.3 VMware Python 客户端……238
14.3.1 安装 PyVmomi 库……239
14.3.2 使用 PyVmomi 库的第一步……240
14.3.3 更改虚拟机的状态…244
14.3.4 更多内容……246
14.4 使用 playbook 管理实例……246
14.5 小结……249
第 15 章 和 OpenStack API 交互……250
15.1 RESTful Web 服务……251
15.2 设置环境……253
15.2.1 安装 rdo-OpenStack 包……253
15.2.2 生成 answer 文件……254
15.2.3 编辑 answer 文件……254
15.2.4 运行 packstack……255
15.2.5 访问 OpenStack GUI…255
15.3 向 OpenStack keystone 发送请求……256
15.4 用 Python 创建实例……259
15.4.1 创建镜像……259
15.4.2 分配类型模板……261
15.4.3 创建网络和子网……262
15.4.4 启动实例……265
15.5 使用 Ansible 管理 OpenStack 实例……266
15.5.1 Shade 和 Ansible 的安装……266
15.5.2 创建 Ansible playbook……267
15.6 小结……270
第 16 章 使用 Python 和 Boto3 自动化 AWS……271
16.1 AWS Python 模块……272
16.2 管理 AWS 实例……274
16.3 自动化 AWS S3 服务……277
16.3.1 创建存储桶……277
16.3.2 上传文件到存储桶…278
16.3.3 删除存储桶……278
16.4 小结……279
第 17 章 使用 Scapy 框架……280
17.1 Scapy……281
17.2 安装 Scapy……282
17.2.1 在基于 UNIX 的系统上安装 Scapy……282