

# 红帽系统管理 III：LINUX自动化

学员练习册

**Internal use only - Not For Sale**

RH294-RHEL8.0-zh-CN-1-20190531  
MAN-RH294SKC

海量视频题库 myitpub.com QQ:5565462  
www.52myit.com

海量视频题库 myitpub.com QQ:5565462  
www.52myit.com

加入探索者、建设者和个人的队伍，一起为老问题大胆提供新解决方案。开源技术的创新离不开幕后人员的贡献。



# Red Hat

## Training and Certification

### 学员练习册

红帽企业 Linux 8.0 RH294

红帽系统管理 III：LINUX 自动化

版 1



Red Hat

myitpub.com  
www.52myit.com

QQ:5565462

海量视频题库 myitpub.com QQ:5565462  
www.52myit.com

# 红帽系统管理 III： LINUX 自动化



海量视频题库  
[www.myitpub.com](http://www.myitpub.com) [www.52myit.com](http://www.52myit.com) QQ:5565462

**红帽企业 Linux 8.0 RH294  
红帽系统管理 III：Linux 自动化  
版 1 20190531  
出版日期 20190531**

作者: Trey Feagle, Herve Quatremain, Dallas Spohn, Adolfo Vazquez,  
Morgan Weetman  
编者: Philip Sweany, Seth Kenlon, Jeff Tyson, Nicole Muller

Copyright © 2019 Red Hat, Inc.

The contents of this course and all its modules and related materials, including handouts to audience members, are  
Copyright © 2019 Red Hat, Inc.

No part of this publication may be stored in a retrieval system, transmitted or reproduced in any way, including, but  
not limited to, photocopy, photograph, magnetic, electronic or other record, without the prior written permission of  
Red Hat, Inc.

This instructional program, including all material provided herein, is supplied without any guarantees from Red Hat,  
Inc. Red Hat, Inc. assumes no liability for damages or legal action arising from the use or misuse of contents or details  
contained herein.

If you believe Red Hat training materials are being used, copied, or otherwise improperly distributed please e-mail  
[training@redhat.com](mailto:training@redhat.com) or phone toll-free (USA) +1 (866) 626-2994 or +1 (919) 754-3700.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, Hibernate, Fedora, the Infinity Logo, and RHCE are  
trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a registered trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or  
other countries.

The OpenStack® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/  
service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack  
Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the  
OpenStack community.

All other trademarks are the property of their respective owners.

本课程的部分内容改编自 Ansible Lightbulb 项目。该项目的资料可依据 MIT 许可证从 <https://github.com/ansible/lightbulb> 获取。

海量视频题库 myitpub.com www.52myit.com QQ:5565462

<b>文档规范</b>	<b>ix</b>
<b>简介</b>	<b>xi</b>
红帽系统管理 III：Linux 自动化 .....	xi
课堂环境介绍 .....	xii
国际化 .....	xiv
<b>1. 介绍 Ansible</b>	<b>1</b>
使用 Ansible 自动化 Linux 管理任务 .....	2
小测验：使用 Ansible 自动化 Linux 管理任务 .....	8
安装 Ansible .....	10
指导练习：安装 Ansible .....	14
总结 .....	16
<b>2. 部署 Ansible</b>	<b>17</b>
构建 Ansible 清单 .....	18
指导练习：构建 Ansible 清单 .....	22
管理 Ansible 配置文件 .....	26
指导练习：管理 Ansible 配置文件 .....	33
运行临时命令 .....	37
指导练习：运行临时命令 .....	44
开放研究实验：部署 Ansible .....	49
总结 .....	58
<b>3. 实施 Playbook</b>	<b>59</b>
编写和运行 Playbook .....	60
指导练习：编写和运行 Playbook .....	66
实施多个 Play .....	71
指导练习：实施多个 Play .....	80
开放研究实验：实施 Playbook .....	86
总结 .....	92
<b>4. 管理变量和事实</b>	<b>93</b>
管理变量 .....	94
指导练习：管理变量 .....	101
管理机密 .....	106
指导练习：管理机密 .....	111
管理事实 .....	114
指导练习：管理事实 .....	123
开放研究实验：管理变量和事实 .....	128
总结 .....	140
<b>5. 实施任务控制</b>	<b>141</b>
编写循环和条件任务 .....	142
指导练习：编写循环和条件任务 .....	152
实施处理程序 .....	155
指导练习：实施处理程序 .....	157
处理任务失败 .....	162
指导练习：处理任务失败 .....	166
开放研究实验：实施任务控制 .....	171
总结 .....	182
<b>6. 在被管理节点上创建文件或目录</b>	<b>183</b>
修改文件并将其复制到主机 .....	184
指导练习：修改文件并将其复制到主机 .....	189
使用 Jinja2 模板部署自定义文件 .....	197
指导练习：使用 Jinja2 模板部署自定义文件 .....	202
开放研究实验：将文件部署到受管主机 .....	205

总结 .....	211
<b>7. 管理大项目</b>	<b>213</b>
利用主机模式选择主机 .....	214
指导练习: 利用主机模式选择主机 .....	222
管理动态清单 .....	229
指导练习: 管理动态清单 .....	234
配置并行 .....	238
指导练习: 配置并行 .....	241
包含和导入文件 .....	246
指导练习: 包含和导入文件 .....	251
开放研究实验: 管理大项目 .....	256
总结 .....	265
<b>8. 利用角色简化 Playbook</b>	<b>267</b>
描述角色结构 .....	268
小测验: 描述角色结构 .....	273
利用系统角色重用内容 .....	275
指导练习: 利用系统角色重用内容 .....	282
创建角色 .....	288
指导练习: 创建角色 .....	294
使用 Ansible Galaxy 部署角色 .....	300
指导练习: 使用 Ansible Galaxy 部署角色 .....	307
开放研究实验: 利用角色简化 Playbook .....	314
总结 .....	325
<b>9. 对 Ansible 进行故障排除</b>	<b>327</b>
对 playbook 进行故障排除 .....	328
指导练习: 对 playbook 进行故障排除 .....	331
对 Ansible 受管主机进行故障排除 .....	339
指导练习: 对 Ansible 受管主机进行故障排除 .....	343
开放研究实验: 对 Ansible 进行故障排除 .....	347
总结 .....	355
<b>10. 自动执行 Linux 管理任务</b>	<b>357</b>
管理软件和订阅 .....	358
指导练习: 管理软件和订阅 .....	366
管理用户和身份验证 .....	373
指导练习: 管理用户和身份验证 .....	376
管理引导过程和调度的进程 .....	383
指导练习: 管理引导过程和调度的进程 .....	387
管理存储 .....	396
指导练习: 管理存储 .....	404
管理网络配置 .....	412
指导练习: 管理网络配置 .....	427
开放研究实验: 自动执行 Linux 管理任务 .....	440
总结 .....	440
<b>11. 总复习: 使用 Ansible 实现自动化</b>	<b>441</b>
总复习 .....	442
开放研究实验: 部署 Ansible .....	445
开放研究实验: 创建 Playbook .....	449
开放研究实验: 创建角色和使用动态清单 .....	457
<b>A. 补充主题</b>	<b>469</b>
检查 Ansible 配置选项 .....	470
<b>B. Ansible Lightbulb 许可</b>	<b>473</b>



海量视频题库 myitpub.com QQ:5565462  
www.52myit.com

# 文档规范



## 参考文献

“参考资料”介绍了查找与主题相关的外部文档的位置。



## 注意

“注意事项”包括针对执行手头任务的提示、快捷方式或者替代方法。忽略注意事项对结果不会有负面影响，但您可能错失某个可简化操作的技巧。



## 重要

“重要信息”框详细介绍容易忽略的内容：仅适用于当前会话的配置更改，或在应用更新前需要重新启动的服务。忽略标有“重要信息”的框不会导致数据丢失，但可能引起不便和困扰。



## 警告

“警告”不可忽略。忽略警告将很可能导致数据丢失。

海量视频题库 myitpub.com QQ:5565462  
www.52myit.com

# 简介

## 红帽系统管理 III：LINUX 自动化

红帽系统管理 III：Linux 自动化 (RH294) 适用于需要实现调配、配置、应用部署和编排自动化的 Linux 系统管理员和开发人员。

学员将了解更多如何在管理工作站上安装和配置 Ansible，并为自动化准备受管主机。学员将编写 Ansible Playbook 以自动执行任务，并运行这些任务以确保正确部署和配置服务器。将探索自动执行常见 Linux 系统管理任务的方法示例。

### 课程目标

- 在控制节点上安装和配置 Ansible 或红帽 Ansible 引擎
- 创建和管理受管主机清单，并为 Ansible 自动化准备受管主机。
- 从命令行运行各个临时自动化任务。
- 编写 Ansible Playbook 以持续自动执行多个任务并将其应用于受管主机。
- 使用变量和事实参数化 playbook，并使用 Ansible Vault 保护敏感数据。
- 编写并重用现有的 Ansible 角色，以简化 playbook 创建和重用代码。
- 使用 Ansible 自动执行常见的红帽企业 Linux 系统管理任务。

### 培训对象

- 负责配置管理的自动化、一致且可重复的应用部署、调配和部署开发、测试和生产服务器以及与 DevOps CI/CD 工作流集成的 Linux 系统管理员、DevOps 工程师、基础架构自动化工程师和系统设计工程师。

### 入学条件

- 红帽认证系统管理员 (EX200 / RHCSA) 认证或同等的红帽企业 Linux 知识与经验。

# 课堂环境介绍

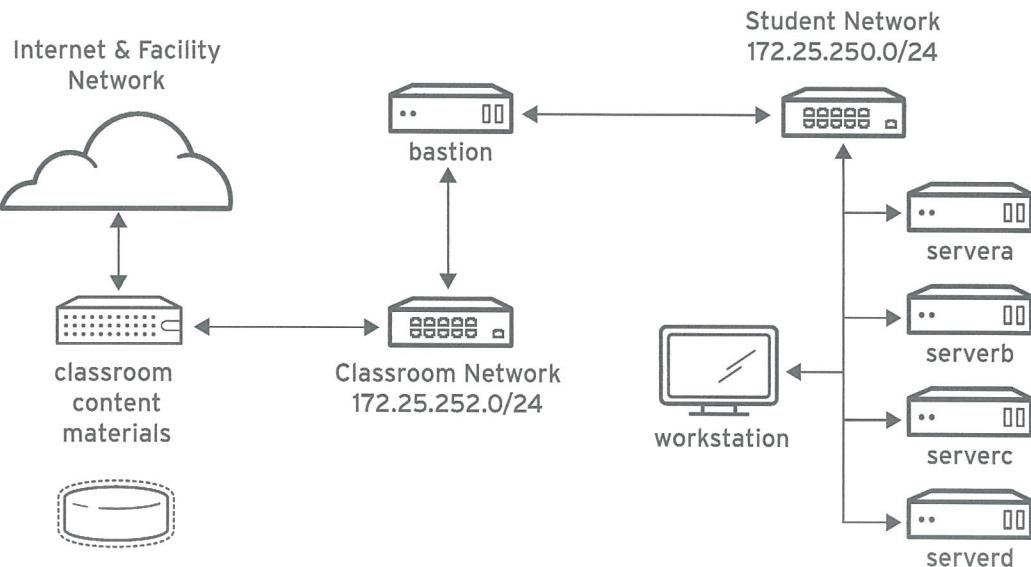


图 0.1: 课堂环境

在本课程中，用于实践学习活动的主要计算机系统是 **workstation**。另有四台计算机供学员用于这些活动：**servera**、**serverb**、**serverc** 和 **serverd**。所有这五个系统都在 `lab.example.com` DNS 域内。

所有学员计算机系统都有一个标准用户帐户 **student**，其密码为 **student**。所有学员系统的 **root** 密码都是 **redhat**。

## 教室计算机

计算机名称	IP 地址	角色
bastion.lab.example.com	172.25.250.254	用于将学员专用网络连接到课堂服务器的网关系统（必须始终处于运行状态）
workstation.lab.example.com	172.25.250.9	用于进行系统管理的图形工作站
servera.lab.example.com	172.25.250.10	通过 Ansible 管理的主机
serverb.lab.example.com	172.25.250.11	通过 Ansible 管理的主机
serverc.lab.example.com	172.25.250.12	通过 Ansible 管理的主机
serverd.lab.example.com	172.25.250.13	通过 Ansible 管理的主机

## 简介

**bastion** 的主要功能是充当连接学员计算机的网络和课堂网络之间的路由器。如果 **bastion** 客机，则其他学员计算机将只能访问各个学员网络上的系统。

课堂中有几个系统提供支持服务。**content.example.com** 和 **materials.example.com** 这两台服务器提供实践活动中使用的软件和实验材料。有关如何使用这些服务器的信息将在这些活动的说明中提供。这些信息由 **classroom.example.com** 虚拟机提供。为确保正确使用实验环境，**classroom** 和 **bastion** 应始终处于运行状态。

## 控制您的系统

在讲师主导的培训课堂上，为学员分配物理计算机 (**foundationX.ilt.example.com**)，用以访问他们在该主机上运行的虚拟机。学员以 **kiosk** 用户身份并使用密码 **redhat** 自动登录物理计算机。您操作的课堂系统就是在该主机上运行的虚拟机。

在 **foundationX** 上，使用一个名为 **rht-vmctl** 的特殊命令来操作虚拟机。下表中的命令应当以 **kiosk** 用户身份在 **foundationX** 上运行，可以用于这些虚拟机中的任何一台。

### **rht-vmctl** 命令

操作	命令
启动 <b>server</b> 计算机	<b>rht-vmctl start server</b>
查看“物理控制台”，以进行登录并操作 <b>server</b> 计算机	<b>rht-vmview view server</b>
将 <b>server</b> 计算机重置为之前的状态，并重新启动虚拟机	<b>rht-vmctl reset server</b>

开始实验室练习时，如果显示“reset your servera”的说明，则表示您应当在 **foundationX** 系统上以 **kiosk** 用户身份运行 **rht-vmctl reset servera** 命令。类似地，如果显示“reset your workstation”说明，则表示您应该在 **foundationX** 上以 **kiosk** 用户身份运行 **rht-vmctl reset workstation** 命令。

# 国际化

---

## 语言支持

红帽企业 Linux 8 官方支持 22 种语言：英语、阿萨姆语、孟加拉语、简体中文、繁体中文、法语、德语、古吉拉特语、印地语、意大利语、日语、卡纳达语、韩语、马拉雅拉姆语、马拉地语、奥里亚语、葡萄牙语（巴西）、旁遮普语、俄语、西班牙语、泰米尔语和泰卢固语。

## 按用户选择语言

用户可能希望使用不同于系统范围默认值的语言显示其桌面环境。他们也可能想要将帐户设置为使用不同的键盘布局或输入法。

## 语言设置

在 GNOME 桌面环境中，用户可能会在第一次登录时看到有关设置首选语言和输入法的提示。如果没有，则个人用户调整首选语言和输入法设置的最简单方式为使用 **Region & Language** 应用。运行 **gnome-control-center region** 命令，或者从顶栏中选择（用户）→ Settings。在打开的窗口中，选择 Region & Language。用户可以单击 Language 框，然后从出现的列表中选择他们的首选语言。这也会将 Formats 设置更新为该语言的默认值。用户下一次登录时，这些更改将全部生效。

这些设置会影响 GNOME 桌面环境以及在其中启动的任何应用，如 **gnome-terminal** 等。但是，从远程系统通过 **ssh** 登录访问该帐户时，或者从本地文本控制台（如 **tty2**）访问时，将不应用这些设置。



### 注意

用户可以使其 shell 环境使用与图形环境相同的 **LANG** 设置，即使通过文本控制台或 **ssh** 登录时也如此。其中一种实现方式是在用户的 **~/.bashrc** 文件中放入类似如下的代码。此示例代码会将文本登录时所用的语言设置为与当前针对用户 GNOME 桌面环境设置的语言相匹配：

```
i=$(grep 'Language=' /var/lib/AccountService/users/${USER} \
    | sed 's/Language=/'')
if [ "$i" != "" ]; then
    export LANG=$i
fi
```

本地文本控制台中可能无法正确显示日语、韩语、中文或其他包含非拉丁字符集的语言。

在命令行上设置 **LANG** 变量，可使个别命令使用其他语言：

```
[user@host ~]$ LANG=fr_FR.utf8 date
jeu. avril 24 17:55:01 CDT 2014
```

后续命令将恢复为使用系统的默认语言进行输出。能够使用 **locale** 命令检查 **LANG** 和其他相关环境变量的当前值。

## 输入法设置

红帽企业 Linux 8 中的 GNOME 3 自动使用 **IBus** 输入法选择系统，用户可以轻松、快速地更改键盘布局和输入法。

**Region & Language** 应用也可用于启用其他输入法。在 **Region & Language** 应用的窗口中，**Input Sources** 框显示当前可用的输入法。默认情况下，**English (US)** 可能是唯一可用的输入法。突出显示 **English (US)**，然后单击 **键盘** 图标，查看当前的键盘布局。

要添加其他输入法，请单击 **Input Sources** 窗口左下方的 **+** 按钮。此时将打开 **Add an Input Source** 窗口。选择语言，然后选择首选的输入法或键盘布局。

配置了多种输入法后，用户可以通过按 **Super+Space**（有时称为 **Windows+Space**）快速切换。状态指示器也将显示在 GNOME 顶栏中，该顶栏有两个功能：其指示哪个输入方法处于活动状态；并且充当一个菜单，用于切换输入方法或者选择更复杂输入方法的高级功能。

一些输入法有齿轮，这表示这些输入法具有高级配置选项和功能。例如，日语的 **Japanese (Kana Kanji)** 输入法允许用户预先用拉丁字母编辑文本，然后用 **向下箭头** 和 **向上箭头** 键选择要使用的正确字符。

美国英语用户也可利用此功能。例如，**English (United States)** 下的键盘布局是 **English (international AltGr dead keys)**，它将 PC 104/105 键版本键盘上的 **AltGr**（或右侧 **Alt**）视为“第二换挡”修饰符键和死键激活键，用于输入其他字符。另外，也有 Dvorak 和其他布局供用户选用。



### 注意

如果用户知道 Unicode 字符的 Unicode 代码点，只需按 **Ctrl+Shift+U**，再输入代码点，即可在 GNOME 桌面环境中输入任意 Unicode 字符。按 **Ctrl+Shift+U** 后将显示带有下划线的 **u**，表示系统正在等待 Unicode 代码点输入。

例如，小写希腊字母 lambda 的代码点为 U+03BB，可以通过按 **Ctrl+Shift+U**，然后按 **03bb**，再按 **Enter** 来输入。

## 系统范围的默认语言设置

系统的默认语言设为美国英语，使用 Unicode 的 UTF-8 编码作为其字符集 (**en\_US.utf8**)，但可以在安装期间或之后更改此设置。

在命令行中，root 用户可以通过 **localectl** 命令更改系统范围的区域设置。如果不带参数运行 **localectl** 命令，它将显示当前的系统范围区域设置。

要设置系统范围的语言，请运行 **localectl set-locale LANG=locale** 命令，其中 **locale** 是本章“语言代码参考”表中相应的 **\$LANG**。此更改存储在 **/etc/locale.conf** 中，将在用户下一次登录时生效。

```
[root@host ~]# localectl set-locale LANG=fr_FR.utf8
```

在 GNOME 中，管理员用户通过单击窗口右上角的 **Login Screen** 按钮，即可在 **Region & Language** 中更改此设置。更改登录屏幕的 **Language**，也会调整存储在 **/etc/locale.conf** 配置文件中的系统范围默认语言设置。

**重要**

**tty2** 等本地文本控制台可显示的字体比 **gnome-terminal** 和 **ssh** 会话受到的局限多。例如，本地文本控制台中的日语、韩语和中文字符显示可能会不正常。因此，最好将英语或其他包含拉丁语字符集的语言用于系统的文本控制台。

类似地，本地文本控制台支持的输入法受到的限制比较多，而且其管理也与图形桌面环境分离。可以通过 **localectl** 为本地文本虚拟控制台和 X11 图形环境配置可用的全局输入设置。有关更多信息，请参见 **localectl(1)**、**kbd(4)** 和 **vconsole.conf(5)** man page。

## 语言包

使用非英语语言时，您可能要安装额外的“语言包”，以提供额外的翻译和字典等。要查看可用的语言包列表，请运行 **yum langavailable**。要查看系统中当前安装的语言包列表，请运行 **yum langlist**。要向系统添加其他语言包，请运行 **yum langinstall code**，其中 code 是 **yum langavailable** 命令输出中语言名称之后用方括号括起的代码。

**参考文献**

**locale(7)**、**localectl(1)**、**kbd(4)**、**locale.conf(5)**、**vconsole.conf(5)**、**unicode(7)**、**utf-8(7)** 和 **yum-langpacks(8)** man page

有关图形桌面环境 X11 布局的名称和 **localectl** 中对应名称之间的转换，请参见 **/usr/share/X11/xkb/rules/base.lst** 文件。

## 语言代码参考

### 语言代码

语言	\$LANG 值
英语（美国）	en_US.utf8
阿萨姆语	as_IN.utf8
孟加拉语	bn_IN.utf8
简体中文	zh_CN.utf8
繁体中文	zh_TW.utf8
法语	fr_FR.utf8
德语	de_DE.utf8
古吉拉特语	gu_IN.utf8
印地语	hi_IN.utf8
意大利语	it_IT.utf8

语言	\$LANG 值
日语	ja_JP.utf8
埃纳德语	kn_IN.utf8
韩语	ko_KR.utf8
马拉雅拉姆语	ml_IN.utf8
马拉地语	mr_IN.utf8
奥里亚语	or_IN.utf8
葡萄牙语（巴西）	pt_BR.utf8
旁遮普语	pa_IN.utf8
俄语	ru_RU.utf8
西班牙语	es_ES.utf8
泰米尔语	ta_IN.utf8
泰卢固语	te_IN.utf8

海量视频题库 myitpub.com QQ:5565462  
www.52myit.com

## 章 1

# 介绍 ANSIBLE

### 目标

描述 Ansible 基本概念和使用方法，并安装红帽 Ansible 引擎。

### 培训目标

- 描述使用 Ansible 自动化 Linux 管理任务的动机、基本的 Ansible 概念，以及 Ansible 的基本架构。
- 在控制节点上安装 Ansible 并描述社区 Ansible 和红帽 Ansible 引擎之间的区别。

### 章节

- 使用 Ansible 自动化 Linux 管理任务（和测验）
- 安装 Ansible（及引导式练习）

# 使用 ANSIBLE 自动化 LINUX 管理任务

## 培训目标

学完本节后，您应能够描述使用 Ansible 自动化 Linux 管理任务的动机、基本的 Ansible 概念，以及 Ansible 的基本架构。

## 自动化与 LINUX 系统管理

多年来，大多数系统管理和基础架构管理都依赖于通过图形或命令行用户界面执行的手动任务。系统管理员通常使用检查清单、其他文档或记忆的例程来执行标准任务。

这样的做法容易出错。系统管理员很容易跳过某个步骤或错误地执行某个步骤。对步骤执行是否正确或产生预期的结果的验证通常有限。

另外，如果以手动方式单独管理每一服务器，对于应该要在配置上完全一致的许多服务器而言，它们很容易会出现微小（或重大）的差异。这可能会加大维护的难度，并给 IT 环境带来错误或不稳定性。

自动化可以帮助您避免手动管理系统和基础架构造成的问题。作为系统管理员，您可以使用它来确保快速、正确地部署和配置所有系统。这样，您可以自动执行日常计划中的重复性任务，从而空出时间并专注于更重要的事情。对组织而言，这意味着您可以更快地推出下一版本的应用或服务更新。

## 基础架构即代码

良好的自动化系统允许您实施基础架构即代码方法。基础架构即代码意味着您可以使用机器可读的自动化语言来定义和描述您希望 IT 基础架构所处的状态。理想情况下，这种自动化语言也应该非常便于人类阅读，因为这样您就可以轻松了解所处的状态并对其进行更改。然后，此代码可应用于您的基础架构，确保它真正处于该状态。

如果自动化语言表示为简单文本文件，则可以在软件代码等版本控制系统中轻松管理。这样做的好处是每个更改都可以签入到版本控制系统中，因此您可以获得随时间所做更改的历史记录。如果您要恢复到更早的已知良好配置，只需签出这一版本的代码并将其应用到您的基础架构。

这样就奠定了一个基础，帮助您遵循 DevOps 中的最佳实践。开发人员可以在自动化语言中定义所需的配置。操作员可以更轻松地查看这些更改以提供反馈，并使用该自动化可重复地确保系统处于开发人员期望的状态。

## 减少人为错误

通过使用任务自动化和基础架构即代码方法来减少在服务器上手动执行的任务，您的服务将更频繁地处于一致的配置中。这意味着您需要习惯于通过更新自动化代码进行更改，而不是手动将其应用到服务器。否则，在下次通过自动化应用更改时您将面临丢失之前手动应用的更改的风险。

自动化允许您使用代码审查、多个主题专家的同行评审，以及自动化本身的程序文档，以降低您的操作风险。

最终，您可以强制必须通过自动化对 IT 基础架构进行更改，以减少人为错误。

## 什么是 ANSIBLE?

Ansible 是一款开源自动化平台。它是一种简单的自动化语言，能够在 Ansible Playbook 中完美地描述 IT 应用基础架构。它也是一个自动化引擎，可运行 Ansible Playbook。

Ansible 可以管理强大的自动化任务，而且能够适应许多不同的工作流和环境。同时，Ansible 新用户可以非常快速地利用它来提高工作效率。

## Ansible 简单明了

Ansible Playbook 提供人类可读的自动化。这表示，playbook 不仅是自动化工具，而且易于阅读、理解和更改。不需要掌握特别的编码技能就能编写它们。Playbook 按顺序执行任务。Playbook 设计的简易性使其可供每个团队使用，让 Ansible 入门者也能够快速获得成效。

## Ansible 功能强大

您可以利用 Ansible 部署应用，也可将它用于配置管理、工作流自动化和网络自动化。Ansible 可用于编排整个应用生命周期。

## Ansible 无需代理

Ansible 围绕无代理架构构建。通常而言，Ansible 通过 OpenSSH 或 WinRM 连接它所管理的主机并且运行任务，方法通常是（但不总是）将称为 Ansible 模块的小程序推送至这些主机。这些程序用于将系统置于需要的特定状态。在 Ansible 运行完其任务后，推送的所有模块都会被删除。您几乎可以立即开始使用 Ansible，因为不需要批准使用任何特殊代理，然后再部署到受管主机上。由于没有代理，也不需要额外的自定义安全基础架构，Ansible 要比其他备选方案更加高效和安全。

Ansible 具有多个重要的优点：

- 跨平台支持：Ansible 提供 Linux、Windows、UNIX 和网络设备的无代理支持，适用于物理、虚拟、云和容器环境。
- 人类可读的自动化：Ansible Playbook 采用 YAML 文本文件编写，易于阅读，有助于确保所有人都能理解它们的用途。
- 完美描述应用：可以通过 Ansible Playbook 进行每一种更改，并描述和记录应用环境的每一个方面。
- 轻松管理版本控制：Ansible Playbook 和项目是纯文本。它们可以视作源代码，放在您的现有版本控制系统中。
- 支持动态清单：可以从外部来源动态更新 Ansible 管理的计算机的列表，随时获取所有受管服务器的当前正确列表，不受基础架构或位置的影响。
- 编排可与其他系统轻松集成：能够利用环境中现有的 HP SA、Puppet、Jenkins、红帽卫星和其他系统，并且集成到您的 Ansible 工作流中。

## ANSIBLE: DEVOPS 的语言



图 1.1: Ansible 用于应用生命周期

沟通是 DevOps 的关键。Ansible 是第一款可以在整个 IT 范围读取和编写的自动化语言。它也是唯一能够从头至尾自动化应用生命周期和持续交付管道的自动化引擎。

## ANSIBLE 概念和架构

Ansible 架构中有两种计算机类型，即控制节点和受管主机。Ansible 在控制节点上安装和运行，该计算机上也含有 Ansible 项目文件的副本。控制节点可以是管理员的笔记本电脑、多个管理员共享的系统，或者运行红帽 Ansible Tower 的服务器。

受管主机列在清单中，清单还可以将这些系统组织到组中，以便于集中管理。清单可以在静态文本文件中定义，或者通过从外部来源获取信息的脚本来动态确定。

Ansible 用户无需编写复杂的脚本，而只要创建高级别 play 即可确保主机或主机组处于特定状态。Play 按该 play 指定的顺序对主机执行一系列任务。这些 play 通过采用 YAML 格式的文本文件来表达。包含一个或多个 play 的文件称为 playbook。

每个任务运行一个模块，即（使用 Python、PowerShell 或某种其他语言编写的）一小段代码。各个模块基本上是您的工具包中的一个工具。Ansible 随附了数百个实用模块，它们能够执行许多不同的自动化任务。它们可以作用于系统文件，安装软件或者进行 API 调用。

在任务中使用时，模块通常确保计算机的某一特定方面处于特定的状态。例如，使用某一模块的任务可以确保某一文件存在且具有特定的权限和内容，而使用另一不同模块的任务可确保已挂载特定的文件系统。如果系统不处于指定的状态，任务应将它置于该状态。如果系统已处于该状态，则不执行任何操作。如果任务失败，Ansible 的默认行为是对发生了错误的主机中止 playbook 的其余部分。

任务、play 和 playbook 设计为具有幂等性。这意味着，您可以在相同主机上多次安全地运行一个 playbook。当您的系统处于正确状态时，playbook 在运行时不会进行任何更改。这意味着，您应该能够在相同主机上多次安全地运行一个 playbook。当您的系统处于正确状态时，playbook 在运行时不应进行任何更改。您可以使用多个模块来运行任意命令。但是，您必须小心使用这些模块，以确保它们以幂等方式运行。

Ansible 也使用插件。插件是您可以添加到 Ansible 中的代码，以对它进行扩展并使它适合新的用途和平台。

Ansible 架构是无代理的。通常，当管理员运行 Ansible Playbook 或临时命令时，控制节点使用 SSH（默认）或 WinRM 连接受管主机。这意味着客户端无需在受管主机上安装特定于 Ansible 的代理，也不需要允许将特殊的网络流量传输到某一非标准端口。

红帽 AnsibleTower 是一种企业框架，可帮助您规模化控制、保护和管理 Ansible 自动化。您可以用它控制谁有权在哪些主机上运行 playbook，共享使用 SSH 凭据而不必允许用户传输或查看其内容，记录您的所有 Ansible 作业，以及管理清单，等等。它提供基于 Web 的用户界面 (web UI) 和 RESTful API。它不是 Ansible 的核心部分，而是单独的产品，能够帮助您更加有效地以团队或更大规模使用 Ansible。

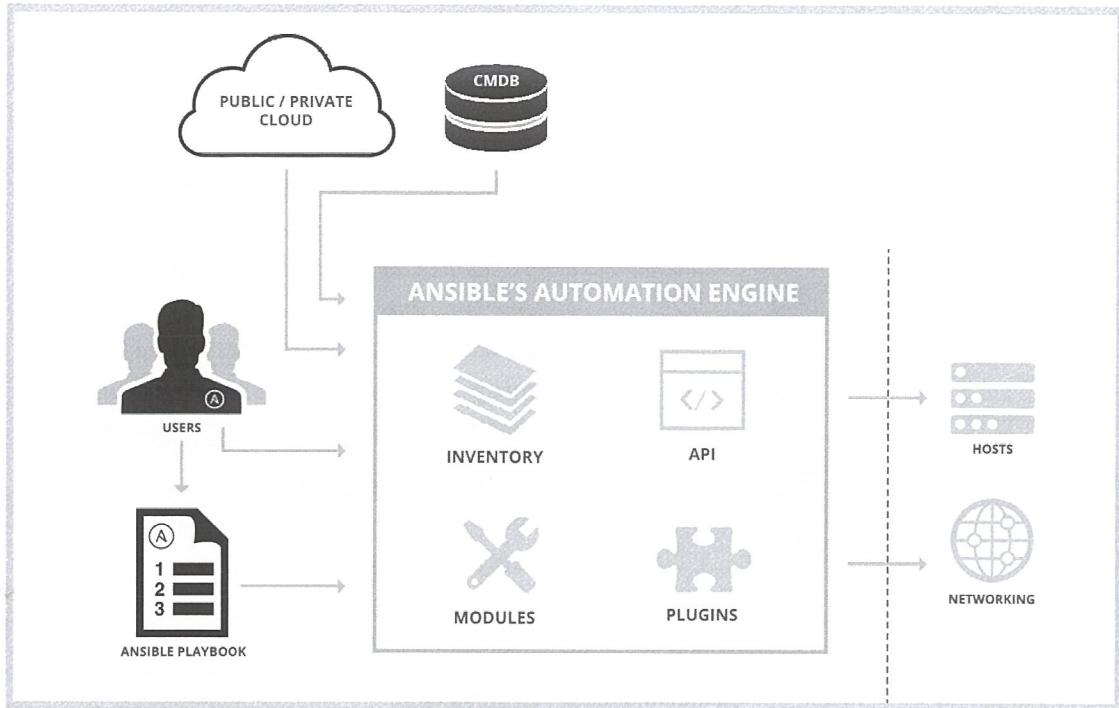


图 1.2: Ansible 架构

## ANSIBLE 方式 复杂性会破坏效率

越简单越好。Ansible 的设计宗旨是工具易用，自动化易写易读。您应利用这一特点在创建自动化时追求简单化。

### 专为易读性优化

Ansible 自动化语言围绕简单易读的声明性文本文件来构建。正确编写的 Ansible Playbook 可以清楚地记录您的工作流自动化。

### 声明式思维

Ansible 是一种要求状态引擎。它通过表达您希望系统处于何种状态来解决如何自动化 IT 部署的问题。Ansible 的目标是通过仅执行必要的更改，使您的系统处于所需的状态。试图将 Ansible 视为脚本语言并非正确的做法。