



开始使用

红帽 Ansible 自动化平台

目录

简介

第1章： 了解基本知识

- 1.1 平台安装
- 1.2 关键平台组件
- 1.3 Ansible Playbook
- 1.4 Ansible 角色
- 1.5 清单
- 1.6 Ansible 内容集

第2章： 开始使用常见用例

- 2.1 置备云资源
- 2.2 大规模部署虚拟机
- 2.3 管理服务
- 2.4 执行合规性审核
- 2.5 管理系统配置
- 2.6 部署应用
- 2.7 配置网络设备
- 2.8 升级操作系统

第3章： 探索高级自动化用例

- 3.1 事件驱动型自动化
- 3.2 边缘自动化
- 3.3 端到端云工作流
- 3.4 企业级自动化

资源和信息



简介

自动化已成为管理当今日益复杂的 IT 环境的关键技术。

借助 IT 自动化，您可以节省员工时间、提高产品质量、鼓励跨团队合作、提升员工满意度并降低整个企业组织的成本。

红帽® Ansible® 自动化平台是大规模构建和运维自动化服务的基石，提供所有的必要工具和功能来帮助您实现企业级自动化。该平台为部署端到端自动化工作流提供灵活、稳定且以安全为重的基础，包括从 IT 流程到混合云，再到边缘位置。借助 Ansible 自动化平台，您可以在整个企业组织内创建、管理和扩展自动化。

本电子书提供有关开始使用 Ansible 自动化平台的指导。我们将回顾关键的自动化概念、产品特性和功能、快速获得价值的用例，以及在整个企业组织范围内实施自动化的策略。

自动化价值的表现

红帽 Ansible 自动化平台将人员和流程统一在一个灵活的基础之上，从而为您的企业组织创造价值：

36%

开发团队工作效率
提升幅度¹

68%

新计算资源部署速度
提升幅度¹

23%

新产品或服务的上市速度
提升幅度¹

61%

计划外停机时间减少
幅度¹

668%

3 年投资回报率 (ROI) ¹

854 万美元

每个企业组织的年收入提升金额¹

第1章：

了解基本知识

在开始使用红帽 Ansible 自动化平台之前，务必要了解该平台包括哪些内容、其组件如何协同工作，以及一些关键的自动化概念，如 playbook、角色和清单。

本章内容：

- 1.1 平台安装
- 1.2 关键平台组件
- 1.3 Ansible Playbook
- 1.4 Ansible 角色
- 1.5 清单
- 1.6 Ansible 内容集

安装红帽 Ansible 自动化平台

红帽 Ansible 自动化平台的安装与设置简单且快捷。安装程序提供多种支持的安装场景，让您能够灵活地安装 Ansible 自动化平台以及可选的事件驱动的 Ansible 控制器。请参阅以下安装指南，了解每种场景的具体详情：

- ▶ [传统虚拟机](#)
- ▶ [红帽 OpenShift®](#)
- ▶ [云环境](#)
- ▶ [容器化环境](#)

无论选择哪种安装场景，安装 Ansible 自动化平台都需要执行以下步骤：

1. **编辑 Ansible 自动化平台安装程序清单文件**，以指定安装场景，并向 Ansible 描述主机部署情况。
2. **运行 Ansible 自动化平台安装程序设置脚本**，并使用安装程序清单文件中定义的参数安装私有自动化中心。
3. 登录自动化控制器、自动化中心和可选的事件驱动的 Ansible 控制器，**验证是否安装成功**。

了解平台的关键组件

红帽 Ansible 自动化平台包含多个关键组件，当您在整个企业组织中创建、管理和扩展自动化时，将与这些组件进行交互。

Ansible 自动化中心

Ansible 自动化中心随 Ansible 自动化平台订阅提供，它是一个中央存储库，可用于发现、下载和管理 **Ansible 内容集**和**验证内容**。**私有自动化中心**是一个位于企业本地的存储库，允许企业在未联网的环境中管理、共享和整理自己的自动化内容，并控制用户对红帽和合作伙伴所创建的认证内容和验证内容的访问权限。



有关 Ansible 内容集的更多信息，请参见第 10 页。

自动化控制器

自动化控制器是 Ansible 自动化平台的控制平面。通过该平面，您可以在整个企业组织内部署、启动、委派和审核自动化。自动化控制器包括用户界面、基于角色的访问控制（RBAC）和工作流，可帮助您高效灵活地扩展自动化。管理清单、启动和调度工作流、跟踪更改，以及生成报告，这一切均可通过集中式用户界面和 **RESTful 应用编程接口（API）** 来完成。



阅读**入门指南**、**管理指南**和**用户指南**，了解如何配置和使用自动化控制器。

自动化网络

自动化网络是一种覆盖网络，可使用现有连接来简化在一组节点之间分发自动化任务。执行节点运行自动化执行环境，其中执行 Ansible playbook 中定义的任务。自动化网络可在这些执行节点之间建立点对点连接，提高自动化工作负载在网络延迟和连接中断情况下的恢复能力。这还允许采用更灵活的架构，并提供快速、独立的控制和执行能力扩展。

事件驱动的 Ansible

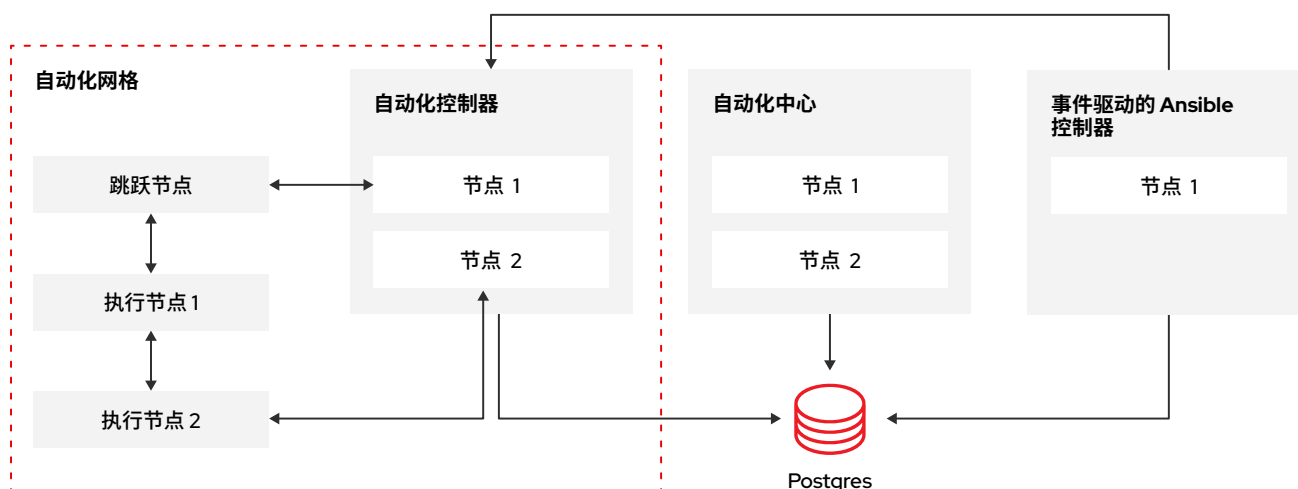
事件驱动的 Ansible 提供事件处理功能，可对任何 IT 领域中不断变化的 IT 条件做出自动响应。它可以处理包含有关 IT 环境条件中离散智能的事件，判断出合适的应对措施，然后自动执行操作以解决或修复事件。IT 服务管理任务（例如优化工单支持、问题修复和用户管理等）是理想的起点，但事件驱动的 Ansible 足够灵活，可自动执行整个 IT 环境中的各种任务。



有关事件驱动的 Ansible 的更多信息，请参见[第 23 页](#)。

红帽 Ansible 自动化平台 2.4 架构

这些组件均在整个 Ansible 自动化平台架构中协同工作。



了解 Ansible Playbook

Playbook 提供通过红帽 Ansible 自动化平台来配置、部署和编排 IT 资产所需的指令。它们由称为 play 的命令集构成，这些命令定义主机清单中的自动化。每个 play 包括一个或多个针对清单中的一个、多个或所有主机的任务。每个任务调用一个**模块**，模块执行特定的功能，如收集信息、管理配置或验证连接。playbook 可以由多个团队共享和重用，以创建可重复的自动化。

本例显示 Ansible Playbook 中的常见组成部分。

```
1  ---
2  - name: Add VLANs
3    hosts: arista
4    gather_facts: false
5
6    vars:
7      vlans:
8        - name: desktops
9          vlan_id: 20
10         - name: servers
11           vlan_id: 30
12         - name: DMZ
13           vlan_id: 50
14
15    tasks:
16      - name: Add VLAN configuration
17        arista.eos.eos_vlans:
18          state: merged
19          config: "{{ vlans }}"
```

表示 playbook 的开端

调用一个名为 **arista** 的设备或设备组

用于检索指定设备相关信息的可选参数

变量定义

在本 playbook 中，我们会直接定义变量值。

通过自动化控制器，您还可以**创建调查**，以便在用户运行您的 playbook 时提醒用户变量值。

任务

任务与 Ansible 模块之间具有一对一相关性。这部分会调用模块，为 **vars** 部分定义的两个变量分别配置虚拟局域网（VLAN）。

什么是 YAML?

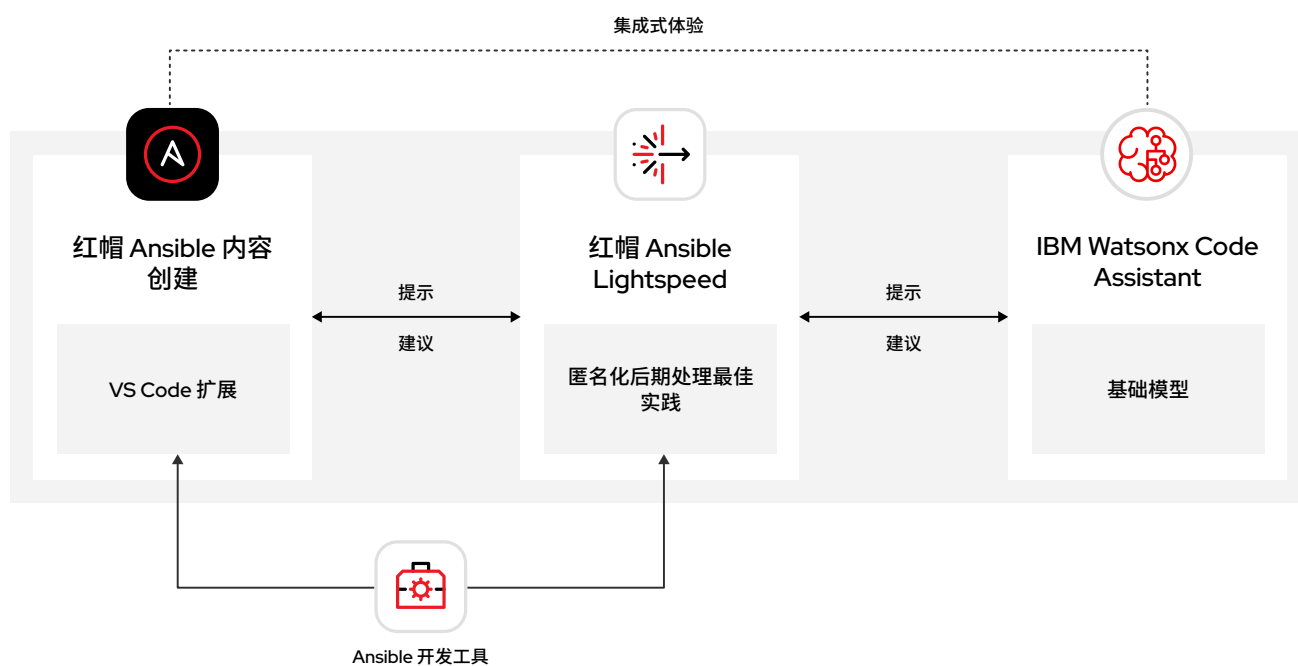
YAML 是易于读取、编写和理解的人性化数据序列化语言。与许多编程语言不同，该语言没有常见的格式符号，如大括号、方括号、结束标记或引号。YAML 文件更容易阅读，因为其使用空格字符缩进来确定结构和表示嵌套。而且由于 YAML 具有出色的灵活性和可访问性，Ansible Playbook、Rulebook 和清单均使用它来完全定义所有自动化内容。

简化 playbook 创建

虽然用 YAML 编写 playbook 很简单，但有两个工具可以进一步简化 playbook 的创建。

Ansible VS Code 扩展针对在支持 `ansible` 和 `ansible-lint` 的操作系统上运行的 Visual Studio Code 和 OpenVSX 兼容编辑器，添加了对 Ansible 的语言支持。系统会识别并高亮显示 Ansible 关键字、模块名称、模块选项以及标准 YAML 元素。在您输入的同时，系统还会验证 Ansible 脚本的语法，并即时提供反馈。打开和保存文档时，`ansible-lint` 会在后台执行，并报告任何潜在问题。智能自动完成功能会检测光标是否位于 `play`、区块或任务上，并提供相关建议。

搭载 IBM watsonx Code Assistant 的红帽 Ansible Lightspeed 是一种生成式人工智能（AI）服务，可协助开发人员更加高效地创建 Ansible 内容。它会接受用户输入的提示，然后与 IBM watsonx 基础模型交互，生成基于 Ansible 最佳实践的代码建议。由于具有上下文感知功能，搭载 watsonx Code Assistant 的 Ansible Lightspeed 还能根据 playbook 中的其他词语（包括 `play` 名称和其他模块）提供建议。



观看本视频，了解如何通过 Ansible VS Code 扩展启用搭载 watsonx Code Assistant 的 Ansible Lightspeed。

利用角色创建可重复使用的自动化内容

共享和重用自动化内容有助于提高团队工作效率并避免重复工作。红帽 Ansible 自动化平台提供了一些技术，可帮助您创建可重复使用的内容，并在整个企业组织内快速发布。

利用 **Ansible 角色**，您可以通过分组和封装相关的自动化工件（如配置文件、模板、任务和处理程序）来构建可重复使用的组件。由于角色隔离了这些组件，可以更轻松地进行重复使用并与其他人共享。您还可以通过公开用户在调用角色时可以设置的变量，使角色可供配置，从而让用户可以根据具体要求配置系统。



[阅读本文](#)，进一步了解如何使用角色创建可重复使用的 playbook。



了解清单

清单是一个主机集合，可使用 Ansible 命令和 playbook 进行操作。清单文件用于对主机分组，可用作您的 IT 资产的可信来源。这些文件可以采取简单的 INI 或 YAML 格式。许多企业组织选择以 YAML 格式编写清单，以与他们的 playbook 保持一致。使用清单文件，一个 playbook 可以通过一个命令维护数百个设备。

本部分将介绍如何构建清单文件。您还可在 GitHub 上找到[示例清单报告](#)。

创建基础的 INI 格式清单

首先，将您的清单按逻辑分组。最好按照**对象**（应用、堆栈或微服务）、**位置**（数据中心或区域）和**时间**（开发阶段）对服务器和设备进行分组。下面是一些示例：

- ▶ **对象**：db、web、leaf、spine
- ▶ **位置**：east、west、floor_19、building_A
- ▶ **时间**：dev、test、staging、prod

这是一个 INI 格式的示例代码，列出了一个非常小的数据中心的基本组结构。您可以使用语法 `[metagroupname:children]` 对组进行分组，并将组列为元组成员。

这里的 `network` 组包含所有的 leaf 和 spine。`datacenter` 组包含所有网络设备和所有网络服务器。

更多详情，请参阅 Ansible 文档中的[“构建您的清单”部分](#)。

```
1  [leafs]
2  leaf01
3  leaf02
4
5  [spines]
6  spine01
7  spine02
8
9  [network:children]
10 leafs
11 spines
12
13 [webserver]
14 webserver01
15 webserver02
16
17 [datacenter:children]
18 network
19 webserver
```

YAML 格式清单解析

```
1 ---
2 all:
3   vars:
4     ansible_user: admin
5     ansible_password: password123
6     ansible_become_pass: password123
7     ansible_become: True
8     ansible_become_method: enable
9     ansible_network_cli_ssh_type: libssh
10  children:
11    routers:
12      children:
13        arista:
14        cisco:
15        juniper:
16    arista:
17      hosts:
18        rtr2:
19          ansible_host: 172.16.100.2
20        rtr4:
21          ansible_host: 172.16.100.4
22      vars:
23        ansible_network_os: arista.eos.eos
24        ansible_connection: ansible.netcommon.network_cli
25    cisco:
26      hosts:
27        rtr1:
28          ansible_host: 172.16.100.1
29      vars:
30        ansible_network_os: cisco.ios.ios
31        ansible_connection: ansible.netcommon.network_cli
32    juniper:
33      hosts:
34        rtr3:
35          ansible_host: 172.16.100.3
36      vars:
37        ansible_network_os: junipernetworks.junos.junos
38        ansible_connection: ansible.netcommon.netconf
```

表示 playbook 的开端

定义适用于清单中所有主机的变量，不限组别

组的层级结构

第 10-15 行确定了该清单中的主机组。在本例中，**routers** 组包含三个子组：**arista**、**cisco** 和 **juniper**。

组定义

hosts 命令定义哪些主机属于哪个组。在本例中，**arista** 组包含 2 台通过 IP 地址识别的主机。

组变量

每个组都可以有自己的变量集。该清单定义每个组的操作系统和连接类型。所有这些组变量都指向内容集中包含的项目。

通过版本控制减少挫折

将 **playbook**、角色、清单和变量文件保存在**版本控制系统**中，并提供有意义的提交信息和注释，可以简化创建、审核和修改自动化内容的过程。您可以通过版本控制工具来跟踪对文件所做的修改，以便跟踪和审核一段时间内的内容开发，并在出现问题时快速恢复到以前的版本。此外，由于版本控制工具支持分支与合并，多个团队成员可以处理同一自动化内容，而且不会相互影响。

请阅读 [Git 版本控制新手指南](#)，详细了解这一热门的版本控制系统。

使用 Ansible 内容集更快上手

Ansible 内容集是自动化内容的标准化分发格式。内容集可以包含 **playbook**、**rulebook**、**角色**、**模块**以及其他**插件**、库存和文档。

您可以利用 2 种类型的即用型内容集：

- ▶ **红帽 Ansible 认证内容**包括由红帽和技术合作伙伴构建、支持和维护的内容集。这些内容集可直接从 Ansible 自动化中心获取，重点关注红帽与合作伙伴平台之间的集成，旨在帮助您跨不同的 IT 领域和技术实现自动化。
- ▶ **Ansible 验证内容**提供有关使用 Ansible 自动化平台在红帽或合作伙伴平台上执行运维或任务的专家指导。验证内容通常以角色和文档的形式交付，包含可自定义的独特用例，并且可能基于红帽 Ansible 认证内容提供这些信息。验证内容可提供与整个 Ansible 自动化平台相关的附加操作功能，包括配置自动化控制器和自动化中心以及创建执行环境。

您也可以开发**内部内容集**并将其上传到私有自动化中心，以便在自己的企业组织内进行分发。内容集发布之后，内部用户便可下载使用。



有关 **Ansible 自动化平台认证内容与验证内容**的更多信息，请访问红帽客户门户网站。

第 2 章：

开始使用常见用例

采用自动化是一趟旅程，而不是孤注一掷的提议。您可以从一个用例入手，然后按照适合企业组织的节奏逐渐扩展。

本章简要介绍常见的自动化用例，以帮助您在企业组织内探索自动化的优势。对于每个用例，实现自动化所需投入的时间和精力各不相同。建议您创建当前流程的流程图，以便了解自动化项目的复杂性以及如何最有效地应用这些用例。成功的自动化采用之旅通常都是循序渐进的：团队从小入手，展示价值，以迭代方式扩大范围，开展更加综合复杂的工作。

通过衡量和分析手动执行一项任务与自动化执行同一任务所需的时间和成本，您可以确定自动化将为整个企业组织节省的成本。**自动化计算器**会提供图表、指标和计算方法，帮助您确定自动化投资将节省的总成本。

您可以使用**自动化分析和用于 Ansible 自动化平台的红帽智能分析**，根据可操作的数据来规划、衡量、管理和扩展自动化。这两款工具包含在您的订阅中，可提供丰富的报告和可观测指标，帮助您跟踪自动化工作。借助自动化分析工具，您可以衡量 Ansible 自动化平台的业务影响，包括投资回报率（ROI）和自动化性能。借助红帽智能分析，您可以监控自动化基础架构，包括系统配置和运行状况。您可以及时发现并解决问题，如基础架构性能问题、系统可用性问题和安全漏洞。您还可以借助事件驱动的 Ansible，将红帽智能分析的通知作为数据源来启动自动修复。

本章内容：

- 2.1 置备云资源
- 2.2 大规模部署虚拟机
- 2.3 管理服务
- 2.4 执行合规性审核
- 2.5 管理系统配置
- 2.6 部署应用
- 2.7 配置网络设备
- 2.8 升级操作系统

用例：置备云资源

对许多企业组织而言，置备云资源是一个耗时且容易出错的过程。您可以使用红帽 Ansible 自动化平台来简化公共云环境中的虚拟机置备。使用 Ansible 认证内容创建 `playbook`，以分配存储空间、设置网络和子网并置备虚拟机实例。为实例类型、区域和安全组等配置选项添加变量，使 `playbook` 可重复使用，从而可以在任何位置部署虚拟机。

下面 3 个示例任务演示了在公共云环境中置备云资源。

示例：创建一个带磁盘和网络接口的 Google 云实例

```
1 ---
2 - name: create a instance
3   google.cloud.gcp_compute_instance:
4     name: test_object
5     machine_type: nl-standard-1
6     disks:
7       - auto_delete: 'true'
8         boot: 'true'
9         source: "{{ disk }}"
10    - auto_delete: 'true'
11      interface: NVME
12      type: SCRATCH
13      initialize_params:
14        disk_type: local-ssd
15      labels:
16        environment: production
17    network_interfaces:
18      - network: "{{ network }}"
19        access_configs:
20          - name: External NAT
21            nat_ip: "{{ address }}"
22            type: ONE_TO_ONE_NAT
23    zone: us-central1-a
24    project: test_project
25    auth_kind: serviceaccount
26    state: present
```



有关 **Google Cloud** 相关红帽认证内容的更多信息，请参阅红帽生态系统目录。

示例：创建一个具有公共 IP 地址的 AWS EC2 实例

```
1 ---
2 - name: Provision AWS EC2 instance
3   amazon.aws.ec2_instance:
4     name: "public-compute-instance"
5     key_name: "prod-ssh-key"
6     vpc_subnet_id: subnet-5calable
7     instance_type: c5.large
8     security_group: default
9     network:
10       assign_public_ip: true
11     image_id: ami-123456
12     tags:
13       Environment: Testing
```



有关 **AWS 相关红帽认证内容** 的更多信息，请参阅红帽生态系统目录。

示例：创建一个具有托管磁盘的 Microsoft Azure 虚拟机

```
1 ---
2 - name: Provision Microsoft Azure instance
3   azure_rm_virtualmachine:
4     name: vm-managed-disk
5     vm_size: Standard_D4
6     resource_group: myResourceGroup
7     admin_username: "{{ username }}"
8     availability_set: avs-managed-disk
9     managed_disk_type: Standard_LRS
10    image:
11      offer: RHEL
12      publisher: RedHat
13      sku: '8-lvm-gen2'
14      version: latest
```



有关 **Microsoft Azure 相关红帽认证内容** 的更多信息，请参阅红帽生态系统目录。

用例：大规模部署虚拟机

使用手动流程部署虚拟机可能会导致配置错误或意外问题，从而导致停机和服务中断。借助红帽 Ansible 自动化平台，您可以自动创建和维护标准化的虚拟机模板，从而在整个 VMware vSphere 环境中一致地置备虚拟机。使用 VMware 内容集中的 `vmware.vmware_rest.vcenter_vmtemplate_libraryitems` 模块，根据静态虚拟机镜像创建模板。然后，使用同一模块根据模板一致部署新的虚拟机。

示例：使用模板在 VMware 环境中部署虚拟机

```
1 ---
2 - name: Deploy a new VM based on the template
3   vmware.vmware_rest.vcenter_vmtemplate_libraryitems:
4     name: vm-from-template
5     library: "{{ nfs_lib.id }}"
6     template_library_item: "{{ my_template_item.id }}"
7     placement:
8       cluster: "{{ lookup('vmware.vmware_rest.cluster_moid',
9                             '/my_dc/host/my_cluster') }}"
10    folder: "{{ lookup('vmware.vmware_rest.folder_moid',
11                       '/my_dc/vm') }}"
12    resource_pool: "{{ lookup('vmware.vmware_rest.resource_pool_moid',
13                              '/my_dc/host/my_cluster/Resources') }}"
14    state: deploy
```



阅读[使用 Ansible 管理 VMware 模板生命周期](#)以了解更多信息。
访问红帽生态系统目录中的 [VMware 内容集](#)。

用例：管理服务

已知存在问题且需要频繁重新启动的服务可能很难管理。红帽 Ansible 自动化平台可帮助您快速应对应用和服务中反复出现的问题。利用内置模块（包括 `ansible.builtin.systemd` 和 `ansible.builtin.sysvinit`），您可以通过一系列服务管理器控制远程主机上的服务。`ansible.builtin.service` 模块充当服务管理器模块的代理，因此无需为每个服务管理器创建特定任务，即可管理各种环境。这样一来，您就可以创建简单的 `playbook`，以便自动收集受影响系统和应用层的信息，并在报告问题后立即重启服务。

示例：启动服务

```
1 ---
2 - name: Start service httpd, if not started
3   ansible.builtin.service:
4     name: httpd
5     state: started
```

示例：停止服务

```
1 ---
2 - name: Stop service httpd, if started
3   ansible.builtin.service:
4     name: httpd
5     state: stopped
```

示例：重启服务

```
1 ---
2 - name: Restart service httpd
3   ansible.builtin.service:
4     name: httpd
5     sleep: 60
6     state: restarted
```



有关使用 `ansible.builtin.service` 模块管理服务的更多信息，请参阅 Ansible 自动化平台文档。

用例：执行合规审核

大多数环境都包含许多不同的平台和设备，因此手动进行合规审核既困难又耗时。红帽 Ansible 自动化平台可简化并标准化在整个 IT 环境中审核资源的方式。您可以使用 Ansible 认证内容编写 playbook，用于查询、存储和报告系统配置，从而减少手动操作。如果系统配置并非处于预期状态，Ansible 自动化平台会自动记录服务工单，并有选择地修复配置。

示例：收集网络信息

```
1  ---
2  - name: Use Cisco IOS facts module
3    hosts: cisco
4    gather_facts: false # this is not the cisco facts module
5
6    tasks:
7      - name: retrieve facts
8        cisco.ios.ios_facts:
9
10     - name: display version
11       ansible.builtin.debug:
12         msg: "{{ ansible_net_version }}"
13
14     - name: display serial number
15       ansible.builtin.debug:
16         msg: "{{ ansible_net_serialnum }}"
```

示例：检索网络资源信息

```
1  ---
2  - name: Retrieve interface information
3    hosts: cisco
4    gather_facts: false # this is not the cisco facts module
5
6    tasks:
7      - name: use state gathered
8        cisco.ios.ios_interfaces:
9          state: gathered
10         register: interfaces_info
11
12      - name: print interface information
13        ansible.builtin.debug:
14          msg: "{{ interfaces_info }}"
```

示例：备份网络配置

```
1  ---
2  - hosts: cisco
3    gather_facts: false
4
5    tasks:
6      - name: Back up config
7        cisco.ios.ios_config:
8          backup: yes
```

用例：管理系统配置

使资源符合最新的安全标准有助于保护系统和减少漏洞。红帽企业 Linux® 系统角色是 Ansible 认证内容的集合，提供稳定且一致的配置界面，以自动化操作并管理多个版本的红帽企业 Linux。您可以使用这些角色创建和审查 playbook，以便在安全标准发生变化时自动更新系统配置。



如需了解更多详情，请阅读 Ansible 自动化平台文档中的**红帽企业 Linux 系统角色简介**。还可从红帽混合云控制台（console.redhat.com）访问系统角色和文档。

示例：更新内核设置

```
1 ---
2 - name: Manage kernel settings
3   hosts: all
4   vars:
5     kernel_settings_sysctl:
6       - name: fs.epoll.max_user_watches
7         value: 785592
8       - name: fs.file-max
9         value: 379724
10      - name: kernel.threads-max
11        state: absent
12
13     kernel_settings_sysfs:
14       - name: /sys/kernel/debug/x86/pti_enabled
15         value: 0
16       - name: /sys/kernel/debug/x86/retp_enabled
17         value: 0
18       - name: /sys/kernel/debug/x86/ibrs_enabled
19         value: 0
20
21     kernel_settings_systemd_cpu_affinity: "1,3,5,7"
22     kernel_settings_transparent_hugepages: madvise
23     kernel_settings_transparent_hugepages_defrag: defer
24   roles:
25     - linux-system-roles.kernel_settings
```

用例：部署应用

手动应用部署过程可能容易出错，并导致安全风险增加和应用性能下降。红帽 Ansible 自动化平台包含内置模块，可让您编写可重复使用的 playbook，用于在整个环境中简单、一致地安装和配置应用。使用认证模块可以在一个简单易读的 playbook 中通过 YUM（Yellowdog Updater Modified）或 DNF（Dandified YUM）安装 Web 服务器，设置默认主页，启动服务器和配置防火墙。

示例：部署 Web 服务器

```
1  ---
2  - name: Setup the web server
3    hosts: "{{ hosts }}"
4    become: true
5    tasks:
6      - name: httpd installed
7        ansible.builtin.yum:
8          name: httpd
9          state: latest
10
11     - name: custom index.html
12       ansible.builtin.copy:
13         dest: /var/www/html/index.html
14         content: | Custom Web Page
15
16     - name: httpd service enabled
17       ansible.builtin.service:
18         name: httpd
19         enabled: true
20         state: started
21
22     - name: open firewall
23       ansible.posix.firewalld:
24         service: http
25         state: enabled
26         immediate: true
27         permanent: true
```

用例：配置网络设备

手动网络配置和更新方法过于缓慢，无法有效支持现代应用和数据传输要求。红帽 Ansible 认证内容可帮助您实现混合云中许多常见网络任务的自动化。您可以编写 `playbook` 以配置路由器主机名和域名系统（DNS）服务器，并在整个环境中创建和传播虚拟局域网（VLAN）配置。

示例：配置路由器

```
1  ---
2  - name: configure cisco routers
3    hosts: routers
4    connection: ansible.netcommon.network_cli
5    gather_facts: false
6    vars:
7      dns: "8.8.8.8 8.8.4.4"
8
9    tasks:
10     - name: configure hostname
11       cisco.ios.ios_config:
12         lines: hostname {{ inventory_hostname }}
13
14     - name: configure DNS
15       cisco.ios.ios_config:
16         lines: ip name-server {{dns}}
```



阅读 Ansible 自动化平台文档中的[使用 Ansible 网络角色](#)，了解有关实现网络自动化的更多信息。

示例：添加 VLAN

```
1  ---
2  - name: add vlans
3    hosts: arista
4    gather_facts: false
5
6    vars:
7      vlans:
8        - name: desktops
9          vlan_id: 20
10       - name: servers
11         vlan_id: 30
12       - name: DMZ
13         vlan_id: 50
14
15    tasks:
16      - name: add VLAN configuration
17        arista.eos.eos_vlans:
18          state: merged
19          config: "{{ vlans }}"
```

用例：升级操作系统

要完成操作系统升级等基础架构维护任务，往往需要庞大的 IT 人员团队在正常工作时间之外展开工作。有了红帽 Ansible 自动化平台，您可以创建复杂的自动化工作流，以便在环境中执行红帽企业 Linux 操作系统升级。您可以编写 `playbook`，用于下载和安装新的操作系统版本，有条件地重启虚拟机，并自动创建描述已安装服务和软件包的报告。

示例：修补红帽企业 Linux 安装

```
1  ---
2  - name: Upgrade all packages (yum)
3    ansible.builtin.yum:
4      name: '*'
5      state: latest
6      update_only: true
7  when: ansible_pkg_mgr == "yum"
8  register: patchingresult_yum
9
10
11 - name: Upgrade all packages (dnf)
12   ansible.builtin.dnf:
13     name: '*'
14     state: latest
15     update_only: true
16  when: ansible_pkg_mgr == "dnf"
17  register: patchingresult_dnf
18
19
20 - name: Check to see if we need a reboot
21   ansible.builtin.command: needs-restarting -r
22   register: result
23   changed_when: result.rc == 1
24   failed_when: result.rc > 1
25   check_mode: false
26
27
28 - name: Reboot Server if Necessary
29   ansible.builtin.reboot:
30     when:
31       - result.rc == 1
32       - allow_reboot == true
```

第 3 章：

探索高级自动化用例

一旦掌握常见自动化用例的经验，并开始通过自动化实现价值，您就可以将知识和流程扩展到整个企业组织中更高级的用例。

本章讨论使用红帽 Ansible 自动化平台推进自动化项目和目标的后续步骤。

本章内容：

- 3.1 事件驱动型自动化
- 3.2 边缘自动化
- 3.3 端到端云工作流
- 3.4 企业级自动化

利用事件驱动型自动化节省时间和精力

事件驱动型自动化是端到端 IT 自动化之旅向前迈进的下一步。当您的 IT 环境中发生特定事件或情况时，它会自动响应。事件驱动型自动化会从可观测性和其他工具接收信息，决定采取哪些操作，并根据条件规则启动预定义的操作。通过自动响应网络或系统速度减慢、配置偏移、不断变化的基础架构条件和新的服务工单条目等事件，您可以灵活地在整个环境中创建复杂的创新型工作流。

红帽 Ansible 自动化平台包含强大的事件驱动型自动化功能。**事件驱动的 Ansible** 让您以一种预先确定的方法来响应 IT 环境中观察到的事件和条件，而无需人为干预。只需在 **Ansible Rulebook** 中定义 **if-then** 规则、事件源和自动操作即可。该平台会将来自第三方监控和观测工具接收到的事件与相应的规则匹配，确定适当的操作，并执行该操作。

Rulebook 是什么？

Rulebook 是一组条件规则，事件驱动的 Ansible 使用这些规则来执行操作。它们定义一个或多个事件源、条件规则和相应的操作。Rulebook 使用 YAML 编写，并使用 **if-then** 规则将特定事件与自动化操作链接起来。

使用事件驱动的 Ansible 的 5 种方式

1. **主动修复问题。** 自动识别和修复性能下降、配置偏移或安全漏洞等潜在问题，以免影响运维和用户体验。
2. **加快故障排除速度。** 根据事件的类型和严重程度、类似事件的发生频率以及既定的企业政策等因素，自动执行初始响应操作，从而简化并加快故障排除活动。
3. **处理用户管理请求。** 根据用户角色和请求类型等信息，自动评估和响应用户管理请求，如恢复密码或管理访问权限。
4. **主动管理系统。** 注意配置偏移并自动应用更新，以维护整个基础架构中 IT 系统的预期状态。
5. **扩展和调优系统。** 根据基础架构监控工具报告的数据（包括网络带宽和延迟以及处理器和存储使用情况），自动扩展和调优基础架构，以满足用户和应用的需求。



阅读以下检查清单，了解有关如何使用事件驱动的 Ansible 的更多信息。

- ▶ **将事件驱动型自动化纳入 IT 策略的五大原因**
- ▶ **事件驱动的 Ansible 自动化帮助您取得更多成效的 5 种方法**

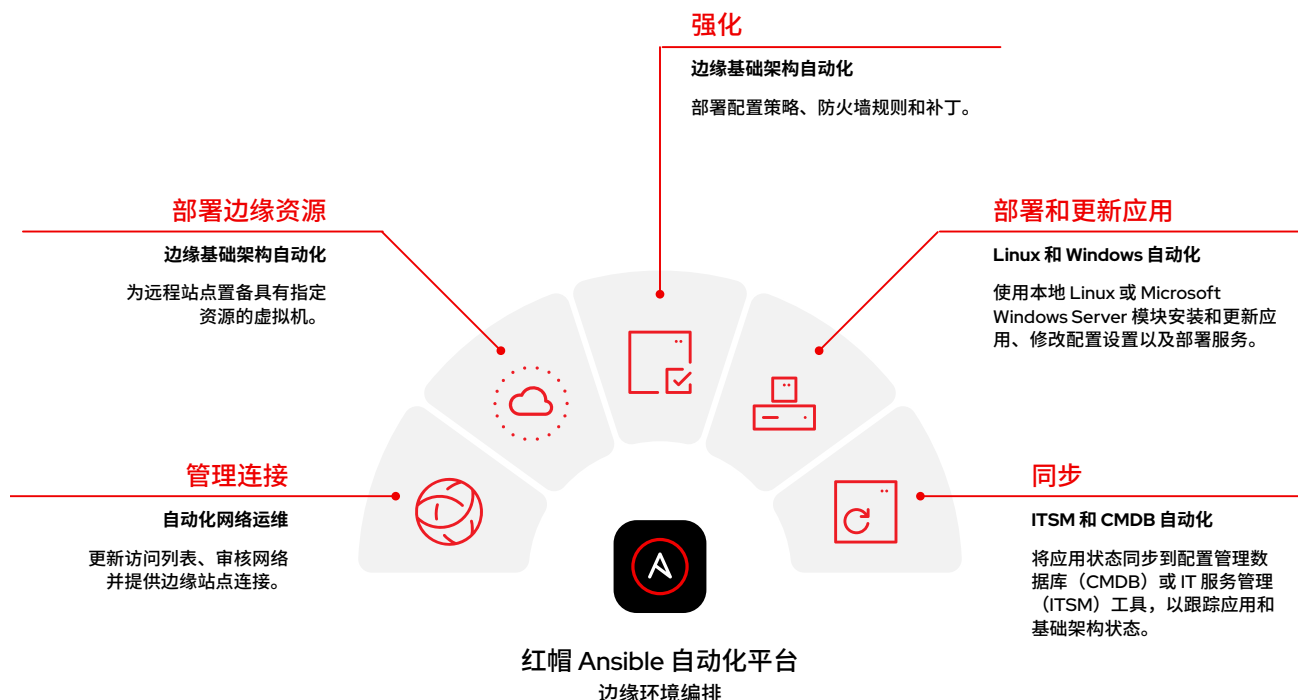
在边缘部署自动化

边缘计算将更多处理能力移到更靠近数据源的位置，让您能够部署延迟敏感型应用，从大量设备收集数据，并创建即使失去数据中心或云连接仍然能够工作的站点。实现[边缘自动化](#)可以帮助您以一致且标准化的方式管理、编排和维护整个 IT 环境。

红帽 Ansible 自动化平台使用容器化技术跨环境分发和运行自动化，帮助运维团队将数据中心、云乃至边缘位置的所有配置和部署实现标准化。它提供一个覆盖整个 IT 环境的一致视图，以便您的团队能够可靠地管理数千个地点、网络设备和集群。这种统一的自动化方法可以帮助您降低运维支出，并在资源有限的边缘站点提供更顺畅的客户体验。



阅读[边缘自动化：7 个行业用例和示例](#)，了解关于边缘自动化的行业特定用例。



实现边缘自动化的六大优势

1. **注重安全性。**自动运行更新、补丁和必要维护，不必派遣技术人员到现场。
2. **减少停机时间。**简化网络管理，减少网络故障，并提高经营利润。
3. **提高效率。**利用自动化分析、监控和警报，改进性能并减少人为错误。
4. **增强可扩展性。**在基础架构中一致地应用配置，并快速扩展边缘设备。
5. **快速响应。**通过基于实时数据和事件的自动化工作流提供优化的用户体验。
6. **确保合规性。**确保系统和应用的运维符合安全防护和审核评估的规定，并满足监管要求和内部标准。

编排完整云工作流

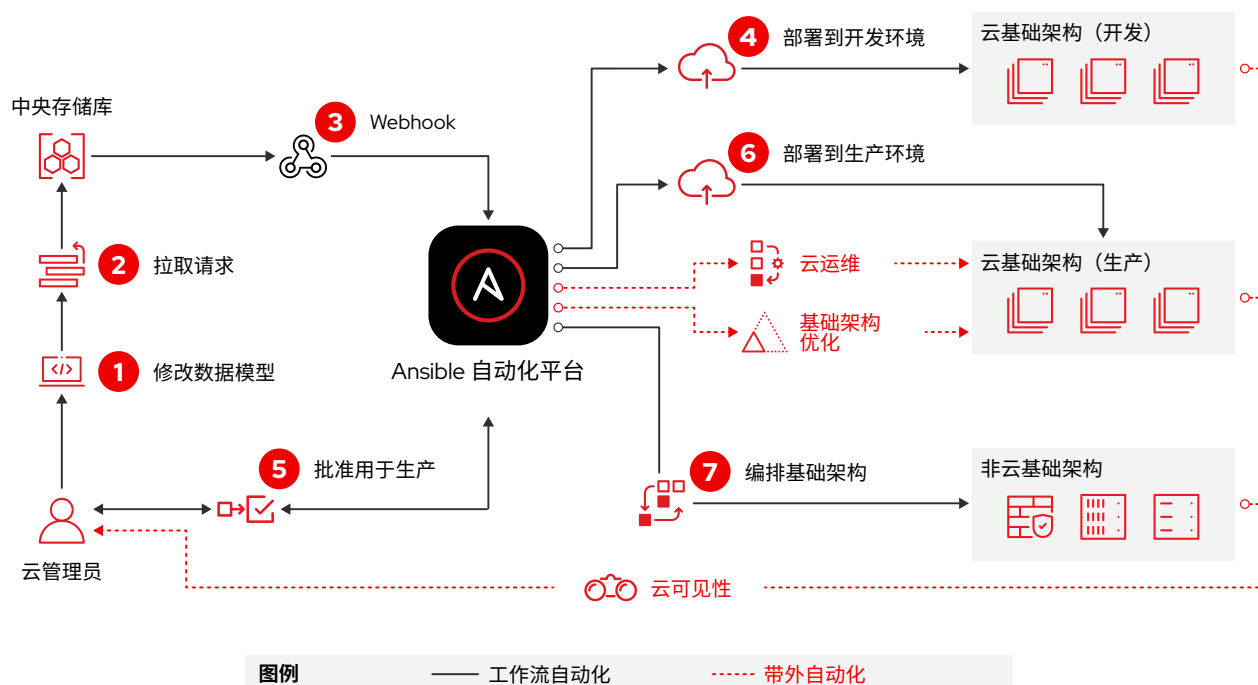
自动化可以通过简化的编排和工作流，帮助您实施整个混合和多云环境（从本地数据中心到公共云基础架构）。通过云自动化，您可以记录、评估和编写任务，使它们能够可靠且可重复地融入到工作流中，实现可预测的业务结果。云自动化还可以帮助您跨所有 IT 和云域创建一致的运维框架。

红帽 Ansible 自动化平台使您可以自动化和编排混合云环境的所有方面，从云资源和服务到操作系统、应用和安全等等。该平台用通用语言将您现有的自动化、配置和云工具以及流程连接起来。因此，您可以跨所有云域、流程和角色创建一致的运维框架，并使自动化更接近目标端点。



阅读**大规模地实现混合云自动化**，了解有关实现云环境自动化的更多信息。

该示例是一个完整的云自动化工作流，展示了如何使用 Ansible 自动化平台和 GitOps 方法来编排云资源和应用生命周期。



工作流自动化

1. 云管理员修改资源定义或 playbook。
2. 云管理员将修改好的定义或 playbook 提交到中央存储库。
3. Ansible 自动化平台中的 Webhook 集成会注意到所做的修改，并启动任何必要的自动化。
4. Ansible 自动化平台将云资源重新部署到开发环境。
5. 云管理员批准自动化生产请求。
6. Ansible 自动化平台将云资源部署到生产环境。
7. Ansible 自动化平台设置和编排生产部署所需的任何其他云外资源。

带外自动化

- **云运维：**Ansible 自动化平台根据需要执行 Day 1-2 运维，包括修改和更新。
- **基础架构优化：**Ansible 自动化平台根据需要优化基础架构和资源。
- **云可见性：**Ansible 自动化平台根据需要保存基础架构快照以用于可见性和洞察目的。

在企业组织范围内推动自动化

企业级自动化不是一蹴而就，也不适合孤注一掷。您需要可持续的自动化策略来指导您的自动化之旅。制定策略需要评估、规划和调整。

确定业务目标

将自动化工作与业务挑战和目标联系起来。这样可以帮助您确定自动化的内容，并提出成功实现自动化自上而下所要满足的要求。例如，您可以自动执行修补以提高系统的安全性和稳定性，同时满足企业对增加正常运行时间的需求。

鼓励跨团队合作

采用激励措施促进在整个企业组织内协同合作。协作使团队能够创建完整的自动化工作流，创造更多的价值。与他人的合作还有助于培养共同的自动化所有权和共担责任。

在整个企业组织内建立信任

为值得信赖的自动化内容构建一个中央存储库。每个团队都应在其专业领域内创建自动化内容并将其添加到该存储库中，以供其他团队使用。员工可以划定界限，让其他人能更放心地使用他们的内容。

分享知识和成功

打造一个由利益相关者组成的核心团队（通常称为实践社区（CoP）或卓越中心（CoE）），以便在整个企业组织内分享自动化的最佳实践、经验和成就。这些团队还应在自动化之旅中积极帮助他人。

集中管理您的自动化内容

选择一个自动化平台，为整个企业组织内的协作、工具和内容提供统一的基础。通过在一个值得信赖的地方共享工具和内容，可以让团队更有效地实现自动化并避免重复工作。

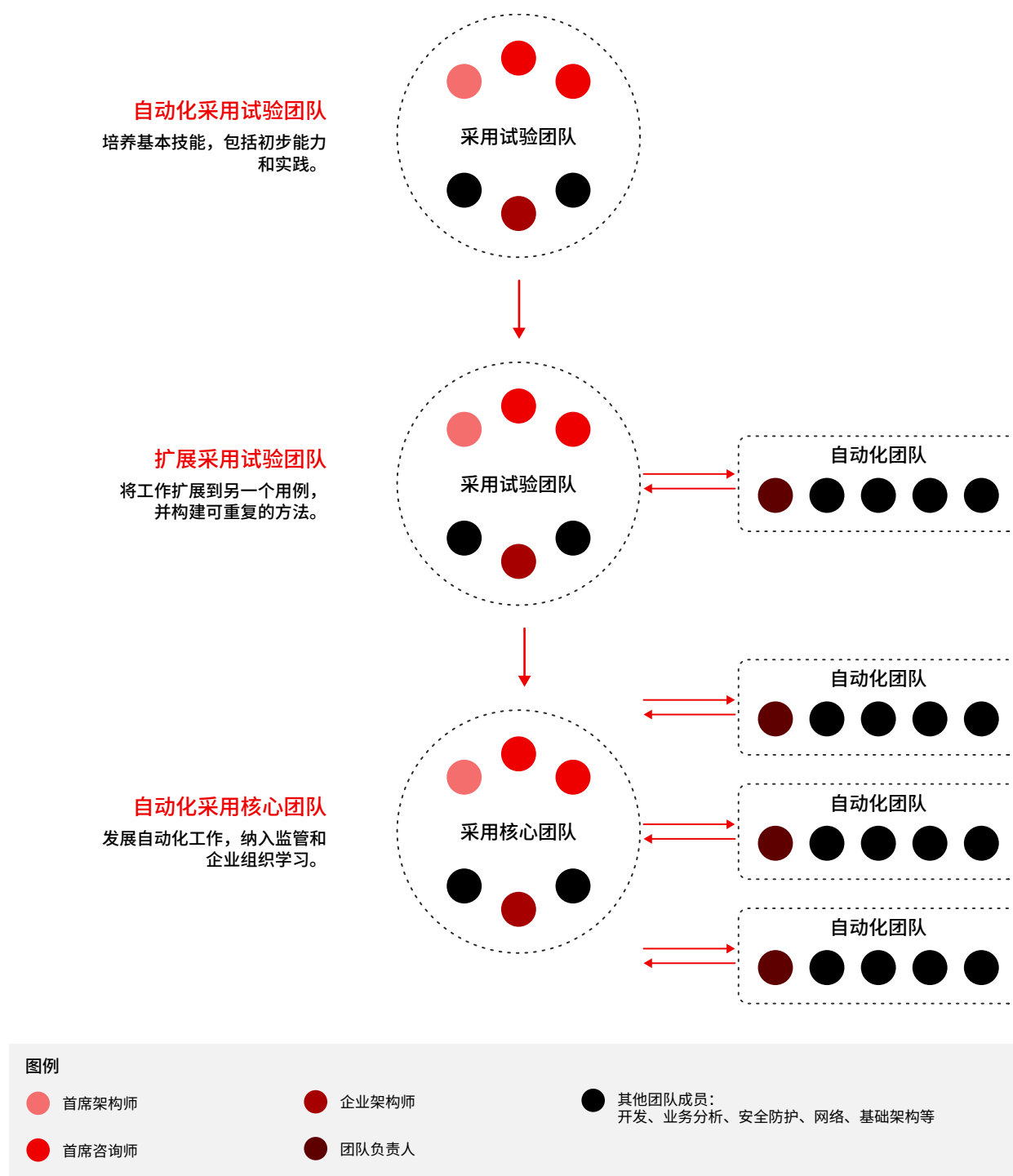
定义成功标准

没有任何一种单独的方法可以衡量自动化的成功程度，每个团队都有自己的特点和目标。应设定符合您团队当前技能水平的现实目标，同时鼓励员工不断提升能力。长期取得自动化成功的示例包括：

- ▶ **采用**，企业完成从愿景到执行的整个过程，并将重点放在简易性和共享知识上。
- ▶ **责任**，每个员工都尽职尽责实现自己的个人目标。
- ▶ **监管**，通过规范的流程进行监管，实现自动化目标并产生可重复的结果。
- ▶ **安全**，借助一个简化的管道，运用可重复和可重用的实践，利用前瞻性的漏洞解决方案，自动进行事故调查和响应。
- ▶ **标准**，提供必要的基础和扩展能力，以实现企业和团队目标。

阅读[自动化企业](#)以了解更多信息。

人员始终是任何企业级计划的核心所在，自动化也不例外。要想在整个企业组织内采用自动化，所有团队（包括业务线、网络、安全、运维、开发和基础架构）都必须积极加入进来，并准备好学习新概念和技能。创建 COP 或 COE 团队，支持整个企业组织采用自动化最佳实践。



资源和信息

红帽提供了许多资源，帮助您开启并推进自动化之旅。

试用免费的互动实验室

这些实训学习场景将为您提供预配置的环境，以便您进行试验、实践，并了解自动化的运作方式。

阅读产品文档

查找 Ansible 自动化平台相关文档，包括发布说明、安装指南和操作信息。

学习良好实践

Ansible 良好实践指南收集了来自红帽 Ansible 从业者、顾问、开发人员及其他人员的建议。

学习免费在线课程

“Ansible 基础知识：自动化技术概述”课程包含一系列点播视频，为您详细介绍 Ansible 自动化平台。

查找所需资源

红帽提供了许多资源，以帮助您轻松入门并不断学习。这些信息都收集在 Ansible 入门网页上。

加入 Ansible 社区

加入 Ansible 社区，讨论面临的 IT 挑战、合作制定解决方案、结识业内同行并在社交网络上建立联系。