



Red Hat Ansible Automation Platform 2.4

Red Hat Ansible Automation Platform 计划指南

计划安装 Ansible Automation Platform

Red Hat Ansible Automation Platform 2.4 Red Hat Ansible Automation Platform 计划指南

计划安装 Ansible Automation Platform

法律通告

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, 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 trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

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.

摘要

本指南提供安装 Red Hat Ansible Automation Platform 的要求、选项和建议。

目录

前言	3
对红帽文档提供反馈	4
第 1 章 规划 RED HAT ANSIBLE AUTOMATION PLATFORM 安装	5
第 2 章 RED HAT ANSIBLE AUTOMATION PLATFORM 架构	6
2.1. ANSIBLE AUTOMATION PLATFORM 架构示例	6
第 3 章 RED HAT ANSIBLE AUTOMATION PLATFORM 组件	7
3.1. ANSIBLE AUTOMATION HUB	7
3.2. 私有自动化中心	7
3.3. 高可用性自动化中心	7
3.4. EVENT-DRIVEN ANSIBLE 控制器	7
3.5. AUTOMATION MESH	7
3.6. 自动执行环境	8
3.7. ANSIBLE GALAXY	8
3.8. 自动化内容导航器	8
第 4 章 系统要求	9
4.1. RED HAT ANSIBLE AUTOMATION PLATFORM 系统要求	9
4.2. 自动化控制器系统要求	10
4.3. AUTOMATION HUB 系统要求	12
4.4. 事件驱动 ANSIBLE 控制器系统要求	14
4.5. POSTGRESQL 要求	14
第 5 章 网络端口和协议	19
第 6 章 附加 RED HAT ANSIBLE AUTOMATION PLATFORM 订阅	24
第 7 章 选择并获取 RED HAT ANSIBLE AUTOMATION PLATFORM 安装程序	26
7.1. 使用互联网访问进行安装	26
7.2. 在没有互联网访问的情况下安装	26
第 8 章 关于安装程序清单文件	28
8.1. 主机和组指南	29
8.2. 取消置备节点或组	30
8.3. 清单变量	31
8.4. 在清单文件中声明变量的规则	31
8.5. 保护清单文件中的 SECRET	31
8.6. 其他清单文件变量	32
第 9 章 支持的安装场景	33
9.1. 带有同一节点上的数据库的独立自动化控制器， 或者一个非安装程序管理的数据库	33
9.2. 带有外部管理数据库的独立自动化控制器	33
9.3. 带有内部数据库的单个 EVENT-DRIVEN ANSIBLE 控制器节点	33
9.4. 带有同一节点上的数据库的独立自动化中心， 或者一个非安装程序管理的数据库	33
9.5. 带有外部管理数据库的独立自动化中心	33
9.6. 在自动化控制器节点或非安装程序管理数据库上使用数据库进行平台安装	33
9.7. 带有外部管理数据库的平台安装	34
9.8. 使用外部管理的数据库安装多机器集群	34

前言

感谢您对 Red Hat Ansible Automation Platform 的关注。Ansible Automation Platform 是一个商业产品，它可以帮助团队通过增加控制、知识、协调基于 Ansible 的环境来更好地管理多层次部署。

使用本指南中的信息来规划 Red Hat Ansible Automation Platform 安装。

对红帽文档提供反馈

如果您对本文档有任何改进建议，或发现了任何错误，请通过 <https://access.redhat.com> 联系技术支持，以使用 **docs-product** 组件在 Ansible Automation Platform JIRA 项目中创建一个问题。

第 1 章 规划 RED HAT ANSIBLE AUTOMATION PLATFORM 安装

Red Hat Ansible Automation Platform 在 Red Hat Enterprise Linux 和 Red Hat OpenShift 上都被支持。使用本指南规划在 Red Hat Enterprise Linux 上安装 Red Hat Ansible Automation Platform。

要在 Red Hat OpenShift Container Platform 环境中安装 Red Hat Ansible Automation Platform，请参阅在 OpenShift Container Platform 上部署 [Red Hat Ansible Automation Platform Operator](#)。

第 2 章 RED HAT ANSIBLE AUTOMATION PLATFORM 架构

作为模块化平台，Ansible Automation Platform 提供了轻松集成组件和自定义部署的灵活性，以更好地满足您的自动化要求。以下章节提供了 Ansible Automation Platform 部署的综合架构示例。

2.1. ANSIBLE AUTOMATION PLATFORM 架构示例

Red Hat Ansible Automation Platform 2.4 参考架构提供了在 Red Hat Enterprise Linux 上使用自动化网络部署 Ansible Automation Platform 的标准部署的示例设置。显示的部署利用以下组件，提供简单、安全且灵活的方法来处理自动化工作负载、内容集合的中央位置以及 IT 请求的自动化解析。

自动化控制器

通过其 UI、Restful API、RBAC 工作流和 CI/CD 集成提供控制平面来实现自动化。

Automation mesh

是一个覆盖网络，它可以通过利用现有网络相互建立对等连接的节点，简化了在大型和分散的 worker 集合中工作分布的功能。

私有自动化中心

为自动化开发人员提供协作和发布自己的自动化内容的能力，并简化组织内部的 Ansible 代码交付。

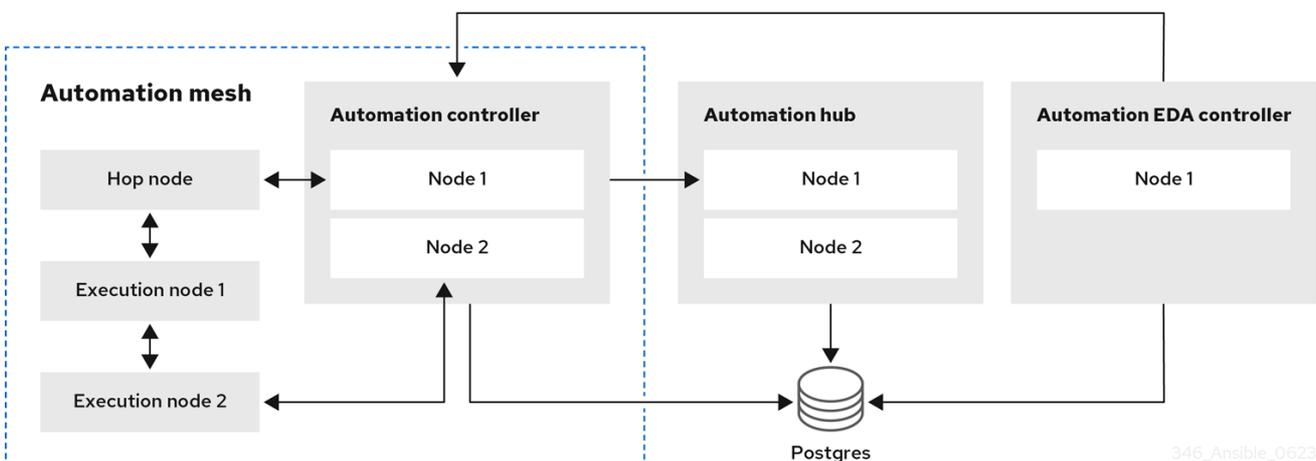
Event-Driven Ansible

提供自动化耗时的任务和响应任何 IT 域中条件所需的事件处理功能。

这个示例的架构包括：

- 两个节点自动化控制器集群
- 将自动化控制器连接到执行节点的可选跃点节点
- 两个节点自动化中心集群
- 单一节点 Event-Driven Ansible 控制器集群
- 连接到自动化控制器、自动化中心和 Event-Driven Ansible 控制器集群的单个 PostgreSQL 数据库
- 每个自动化控制器集群有两个执行节点

图 2.1. Ansible Automation Platform 2.4 架构示例



346_Ansible_0623

第 3 章 RED HAT ANSIBLE AUTOMATION PLATFORM 组件

Ansible Automation Platform 是一个由单独的组件组成的模块平台，可一起连接以满足您的部署需求。Ansible Automation Platform 部署从自动化控制器开始，这是使用用户界面 (UI) 和 RESTful 应用程序编程接口 (API) 控制、保护和管理 Ansible 自动化的企业框架。然后，您可以在部署中添加以下自动化平台组件的任意组合：

3.1. ANSIBLE AUTOMATION HUB

Ansible Automation hub 是一个用于认证 Ansible 内容集合内容的存储库。它是红帽及其合作伙伴的集中存储库，用于发布内容，并允许客户发现经过认证、受支持的 Ansible 内容集合。Red Hat Ansible 认证的内容为用户提供已经过测试并受红帽支持的内容。

3.2. 私有自动化中心

私有自动化中心提供断开连接和内部解决方案，用于同步内容。您可以从 Red Hat 云自动化中心同步集合和执行环境镜像，存储和提供自己的自定义自动化集合和执行镜像。您还可以使用 Ansible Galaxy 或其他容器 registry 等其他源向私有自动化中心提供内容。私有自动化中心可以集成到您的企业目录和 CI/CD 管道中。

3.3. 高可用性自动化中心

高可用性(HA)配置提高了自动化中心部署的可靠性和可扩展性。

自动化中心的 HA 部署有多个节点，它们使用负载均衡器分布工作负载同时运行相同的服务（一个“主动-主动”配置）。此配置消除了单一故障点，以最小化服务停机时间，并可让您轻松添加或移除节点来满足工作负载需求。

3.4. EVENT-DRIVEN ANSIBLE 控制器

Event-Driven Ansible 控制器是事件驱动的自动化的接口，并引入对 IT 请求的自动解析。event-Driven Ansible 控制器可帮助您连接到事件的来源，并使用规则规则对这些事件执行操作。该技术提高了 IT 速度和灵活性，实现了一致性和弹性。使用 Event-Driven Ansible，您可以：

- 自动化决策
- 使用许多事件源
- 在很多 IT 用例中实施事件驱动的自动化

其他资源

- [Event-Driven Ansible 指南 入门](#).

3.5. AUTOMATION MESH

自动化网格是一个覆盖网络，旨在通过使用现有网络相互建立对等连接的节点，简化在大型且分散的 worker 集合中的工作分布。

Automation mesh 提供：

- 动态集群容量可独立扩展，允许您以最少的停机时间创建、注册、分组、分组和取消注册节点。

- 控制和执行平面分离，可让您独立于 control plane 容量来缩放 playbook 执行容量。
- 部署选择具有应对延迟、在没有中断中断的情况下可重新配置的部署选择，并在存在中断时动态重新路由以选择不同的路径。
- Mesh 路由更改。
- 包括符合联邦信息处理标准(FIPS)的双向多跃网格通信可能性的连接。

3.6. 自动执行环境

自动化执行环境是运行 Red Hat Ansible Automation Platform 中的所有自动化的容器镜像。它们提供了一个包括 Ansible 执行引擎和数百个模块的解决方案，可帮助用户自动化 IT 环境和流程的所有方面。自动化执行环境自动执行常用的操作系统、基础架构平台、网络设备和云。

3.7. ANSIBLE GALAXY

Ansible Galaxy 是一个用于查找、重复利用和共享 Ansible 内容的中心。社区提供的 Galaxy 内容采用预打包角色的形式，可帮助启动自动化项目。用于配置基础架构、部署应用和完成其他任务的角色可被丢弃到 Ansible Playbook 中，并立即应用到客户环境。

3.8. 自动化内容导航器

自动化内容导航器是一个文本用户界面(TUI)，它成为自动化平台的主要命令行界面，包括内容构建、在执行环境中运行自动化、在 Ansible Automation Platform 中运行自动化，并为未来 集成开发环境(IDE) 提供基础。

第 4 章 系统要求

在规划 Red Hat Ansible Automation Platform 安装并设计适合您的用例的自动化网络拓扑时，请使用此信息。

先决条件

- 您可以通过 `sudo` 命令或特权升级来获取 root 访问权限。如需有关特权升级的更多信息，[请参阅了解特权升级](#)。
- 您可以将特权从 root 降级到用户，例如：AWX、PostgreSQL、Event-Driven Ansible 或 Pulp。
- 您已在所有节点上配置了 NTP 客户端。如需更多信息，[请参阅使用 Chrony 配置 NTP 服务器](#)。

4.1. RED HAT ANSIBLE AUTOMATION PLATFORM 系统要求

您的系统必须满足以下最低系统要求才能安装和运行 Red Hat Ansible Automation Platform。

表 4.1. 基本系统

要求	必填	备注
Subscription	有效的 Red Hat Ansible Automation Platform	
OS	Red Hat Enterprise Linux 8.6 或更高的 64 位版本 (x86, ppc64le, s390x, aarch64)	OpenShift 也支持 Red Hat Ansible Automation Platform， 请参阅在 OpenShift Container Platform 上部署 Red Hat Ansible Automation Platform Operator 。
ansible-core	ansible-core 版本 2.14 或更高版本	Ansible Automation Platform 包含包含 ansible-core 2.15 的执行环境。
Python	3.9 或更高版本	
浏览器	当前支持的 Mozilla FireFox 或 Google Chrome 版本	
数据库	PostgreSQL 版本 13	

在使用项目更新和集合时，需要满足以下条件：

- 确保 [Table 5.9 中列出的网络端口和协议](#)。Automation Hub 可用于成功连接并从自动化中心或 Ansible Galaxy 服务器下载集合。
- 在使用自签名证书或红帽域时禁用 SSL 检查。



注意

由 Ansible Automation Platform 管理的系统的要求与 Ansible 相同。请参阅 [Ansible 社区文档中的安装 Ansible](#)。

有关 Red Hat Ansible Automation Platform 要求的额外备注

- Red Hat Ansible Automation Platform 依赖于 Ansible Playbook，并且需要安装最新版本的 `ansible-core`。您可以手动下载 `ansible-core`，也可以作为 Red Hat Ansible Automation Platform 安装的一部分自动下载它。
- 对于新安装，自动化控制器会安装最新版本的 `ansible-core`。
- 如果执行捆绑的 Ansible Automation Platform 安装，安装 `setup.sh` 脚本会尝试从捆绑包中安装 `ansible-core`（及其依赖项）。
- 如果您手动安装 Ansible，Ansible Automation Platform 安装 `setup.sh` 脚本将检测到已安装了 Ansible，且不会尝试重新安装它。



注意

您必须使用软件包管理器（如 `dnf`）安装 Ansible，必须安装软件包管理器的最新稳定版本，以便 Red Hat Ansible Automation Platform 正常工作。版本 2.4 及更新的版本需要 Ansible 2.14 版本。

4.2. 自动化控制器系统要求

自动化控制器是一种分布式系统，不同的软件组件可以并置或部署到多个不同计算节点上。在安装程序中，提供四个节点类型作为抽象层，可帮助您设计适合您的用例的拓扑：控制、混合、执行和跃点节点。

对节点大小使用以下建议：



注意

在控制和混合节点上，为执行环境存储分配至少 20 GB 的 `/var/lib/awx`。

执行节点

执行节点运行自动化。增加内存和 CPU 以增加容量来运行更多分叉。



注意

- 执行节点上安装的软件包可能不需要 RAM 和 CPU 资源，但最低建议处理节点的作业负载，以便同时运行平均作业数量。
- 不提供推荐的 RAM 和 CPU 节点大小。所需的 RAM 或 CPU 取决于您在该环境中运行的作业数量。

有关所需 RAM 和 CPU 级别的更多信息，请参阅 [自动化控制器的性能调整](#)。

表 4.2. 执行节点

要求	最低要求
RAM	16 GB
CPU	4
本地磁盘	最小 40GB

控制节点

控制节点处理事件并运行集群作业，包括项目更新和清理作业。增加 CPU 和内存有助于处理作业事件。

表 4.3. 控制节点

要求	最低要求
RAM	16 GB
CPU	4
本地磁盘	<ul style="list-style-type: none"> ● 至少 40GB，在 /var/lib/awx 下至少有 20GB ● 存储卷的最低基础线必须被定做 1500 IOPS ● 项目存储在控制和混合节点上，作业持续时间也存储在执行节点上。如果集群有很多大型项目，请考虑在 /var/lib/awx/projects 中调整 GB，以避免磁盘空间错误。

hop 节点

hop 节点用于将流量从自动化网络的一个部分路由到另一个网络（例如，hop 节点可以是堡垒主机到另一网络）。RAM 可能会影响吞吐量，CPU 活动较低。网络带宽和延迟通常比 RAM 或 CPU 更重要。

表 4.4. 跃点 (hop) 节点

要求	最低要求
RAM	16 GB
CPU	4
本地磁盘	40 GB

- 实际 RAM 的要求取决于同时管理的主机自动化控制器数量（这由作业模板或系统 `ansible.cfg` 文件中的 `forks` 参数控制）。为避免可能的资源冲突，Ansible 建议每 10 个 fork 和 2 GB 保留内存用于自动化控制器。如需更多信息，[请参阅自动控制器容量确定和作业影响](#)。如果 `fork` 设为 400，则建议使用 42 GB 内存。

- 自动化控制器主机检查 **umask** 是否已设置为 0022。如果没有，则设置会失败。设置 **umask=0022** 以避免出现这个错误。
- 可以处理更多主机，但如果 fork 数量小于主机总数，则需要更多主机。您可以使用以下方法之一避免这些 RAM 限制：
 - 使用滚动更新。
 - 使用内置在自动化控制器中的置备回调系统，其中每个请求配置的系统都会进入队列，并尽快处理。
 - 如果自动化控制器正在生成或部署镜像，如 AMI。

其他资源

- 有关获取自动化控制器订阅的更多信息，请参阅 [导入订阅](#)。
- 如有疑问，请通过红帽客户门户网站联系 Ansible [支持](#)。

4.3. AUTOMATION HUB 系统要求

通过自动化中心，您可以从 Red Hat Ansible 和认证合作伙伴发现并使用新的认证自动化内容。在 Ansible Automation Hub 上，您可以发现和管理由红帽及其合作伙伴开发的自动化内容的 Ansible 集合，用于云自动化、网络自动化和安全自动化等用例。

自动化中心有以下系统要求：

要求	必填	备注
RAM	最小 8 GB	<ul style="list-style-type: none"> ● 8 GB RAM (Vagrant trial 版本安装的最小和推荐值) ● 8 GB RAM (外部独立 PostgreSQL 数据库最小值) ● 有关配置中基于 fork 的容量，请参阅 自动化控制器容量确定和作业影响。
CPU	最少 2 个	有关配置中基于 fork 的容量，请参阅 自动化控制器容量确定和作业影响 。
本地磁盘	60 GB 磁盘	至少指定 40GB 到 /var 进行集合存储。



注意

私有自动化中心

如果您从内部地址安装私有自动化中心，并且具有仅包含外部地址的证书，这可能会导致安装无法用作容器 registry，且没有证书问题。

要避免这种情况，请使用 `automationhub_main_url` 清单变量和值，如 `https://pah.example.com` 链接到安装清单文件中的私有自动化中心节点。

这会将外部地址添加到 `/etc/pulp/settings.py`。这代表，您只使用外部地址。

有关清单文件变量的详情，请参考 *Red Hat Ansible Automation Platform 安装指南中的清单文件变量*。https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.4/inventory-files-vars

4.3.1. 高可用性自动化中心要求

在部署高可用性 (HA) 自动化中心前，请确保在您的环境中安装了共享文件系统，并配置了网络存储系统（如果适用）。

4.3.1.1. 所需的共享文件系统

高可用性自动化中心需要您在环境中安装一个共享文件系统，如 NFS。在运行 Red Hat Ansible Automation Platform 安装程序前，请验证作为共享文件系统安装的一部分，在集群中安装了 `/var/lib/pulp` 目录。如果在其中一个节点中没有检测到 `/var/lib/pulp`，则 Red Hat Ansible Automation Platform 安装程序会返回错误，从而导致高可用性自动化中心设置失败。

如果您收到一个节点上没有检测到 `/var/lib/pulp` 的错误，请确保 `/var/lib/pulp` 已在所有服务器中正确挂载并重新运行安装程序。

4.3.1.2. 为网络存储安装 firewalld

如果要使用自动化 hub 节点本身的网络存储安装 HA 自动化中心，您必须首先安装和使用 `firewalld`，以根据共享存储系统的要求打开共享存储系统所需的端口，然后才能运行 Ansible Automation Platform 安装程序。

执行以下命令安装和配置 `firewalld`：

1. 安装 `firewalld` 守护进程：

```
$ dnf install firewalld
```

2. 使用以下命令在 `<service>` 下添加网络存储：

```
$ firewall-cmd --permanent --add-service=<service>
```



注意

如需支持的服务列表，请使用 `$ firewall-cmd --get-services` 命令

3. 重新载入以应用配置：

```
$ firewall-cmd --reload
```

4.4. 事件驱动 ANSIBLE 控制器系统要求

Event-Driven Ansible 控制器是一个可以处理长时间运行的进程数（如规则手册激活）的单节点系统，具体取决于 CPU 内核数。默认情况下，使用以下最低要求来运行，最多 12 个并发激活：

要求	必填
RAM	16 GB
CPU	4
本地磁盘	最小 40 GB

重要

- 如果您正在运行 Red Hat Enterprise Linux 8 并希望设置内存限值，则必须在安装 Event-Driven Ansible 前启用 cgroup v2。具体步骤请查看 [知识库支持\(KCS\)文章，Red Hat Enterprise Linux 8 的 Ansible Automation Platform Event-Driven Ansible 控制器需要 cgroupv2。](#)
- 当您在标准条件下激活 Event-Driven Ansible 规则手册时，它使用大约 250 MB 内存。但是，根据规则的复杂性以及处理事件的卷和大小，实际内存消耗可能会很大不同。在预计大量事件或规则手册复杂性很高的情况下，对暂存环境中的资源使用情况进行初步评估。这样可确保您的激活的最大数量取决于您的资源容量。如需设置 [Event-Driven Ansible 控制器最大运行激活的示例](#)，请参阅[单一自动化控制器、单一自动化中心和带有外部（安装程序管理的）数据库](#)的单个 Event-Driven Ansible 控制器节点。

4.5. POSTGRESQL 要求

Red Hat Ansible Automation Platform 使用 PostgreSQL 13。在将 PostgreSQL 用户密码保存到数据库前，会使用 SCRAM-SHA-256 安全散列算法对其进行处理。

要确定您的自动化控制器实例是否可以访问数据库，您可以使用 `awx-manage check_db` 命令。

表 4.5. 数据库

Service	必填	备注
---------	----	----

Service	必填	备注
数据库	<ul style="list-style-type: none"> ● 20 GB 专用硬盘空间 ● 4 个 CPU ● 16 GB RAM 	<ul style="list-style-type: none"> ● 建议大于 150 GB ● 存储卷必须为高基线 IOPS 进行评级 (1500 或更多)。 ● 所有自动化控制器数据都存储在数据库中。通过管理的主机数量、作业运行数量、事实缓存中存储的 fact 数量以及单个作业中的任务数量，数据库存储会增加。例如，一个 playbook 在 250 个主机中每小时运行一次（一天 24 次），20 个任务每周会在数据库中存储超过 800000 个事件。 ● 如果数据库中没有足够的空间，则必须定期清理旧作业运行和事实。如需更多信息，请参阅 自动化控制器 管理指南中的管理作业。

PostgreSQL 配置

另外，您可以将 PostgreSQL 数据库配置为不由 Red Hat Ansible Automation Platform 安装程序管理的独立节点。当 Ansible Automation Platform 安装程序管理数据库服务器时，它会使用通常为大多数工作负载推荐的默认值配置服务器。有关可以用来提高数据库性能的设置的信息，请参阅 [数据库设置](#)。

其他资源

有关调整 PostgreSQL 服务器的更多信息，请参阅 [PostgreSQL 文档](#)。

4.5.1. 设置外部（客户支持）数据库



重要

红帽不支持使用外部（客户支持）数据库，但客户使用它。以下有关 initial 配置的指导只从产品安装角度提供，以避免相关的支持请求。

要在外部 PostgreSQL 兼容数据库中创建用于自动化控制器的数据库、用户和密码，请使用以下步骤。

流程

1. 安装，然后连接到具有超级用户权限的 PostgreSQL 兼容数据库服务器。

```
# psql -h <db.example.com> -U superuser -p 5432 -d postgres <Password for user superuser>:
```

其中：

```
-h hostname
--host=hostname
```

指定运行服务器的机器的主机名。如果该值以斜杠开头，它将用作 Unix-domain 套接字的目录。

```
-d dbname
--dbname=dbname
```

指定要连接到的数据库的名称。这等同于将 **dbname** 指定为命令行中的第一个非选项参数。**dbname** 可以是连接字符串。如果是，则连接字符串参数会覆盖任何冲突的命令行选项。

```
-U username
--username=username
```

以用户 **用户名** 而非默认值连接到数据库。（您必须有权限才能这样做。）

2. 使用分配给用户的 **createDB** 或管理员角色，创建用户、数据库和密码。如需更多信息，请参阅 [数据库角色](#)。
3. 将数据库凭据和主机详情作为外部数据库添加到自动化控制器清单文件。以下示例中使用了默认值。

```
[database]
pg_host='db.example.com'
pg_port=5432
pg_database='awx'
pg_username='awx'
pg_password='redhat'
```

4. 运行安装程序。
如果您将 PostgreSQL 数据库与自动化控制器搭配使用，则数据库由 connect 用户所有，且必须为其分配 **createDB** 或管理员角色。
5. 检查您是否能够使用用户、密码和数据库名称连接到创建的数据库。
6. 检查用户的权限，该用户应具有 **createDB** 或 administrator 角色。



注意

在此过程中，您必须检查外部数据库覆盖。如需更多信息，请参阅 <https://access.redhat.com/articles/4010491>

4.5.2. 为自动化中心 PostgreSQL 数据库启用 hstore 扩展

在 Ansible Automation Platform 2.4 中，数据库迁移脚本使用 **hstore** 字段来存储信息，因此必须启用对自动化中心 PostgreSQL 数据库的 **hstore** 扩展。

使用 Ansible Automation Platform 安装程序和受管 PostgreSQL 服务器时，此过程是自动的。

如果 PostgreSQL 数据库是外部的，则必须在自动化中心安装前手动为自动化中心 PostgreSQL 数据库启用 **hstore** 扩展。

如果在自动化中心安装前没有启用 **hstore** 扩展，在数据库迁移过程中会引发故障。

流程

1. 检查 PostgreSQL 服务器上是否有扩展（自动化 hub 数据库）。

```
$ psql -d <automation hub database> -c "SELECT * FROM pg_available_extensions WHERE name='hstore'"
```

其中 **<automation hub database>** 的默认值为 **automationhub**。

带有 **hstore** 可用的输出示例：

```
name | default_version | installed_version | comment
-----+-----+-----+-----
hstore | 1.7            |                  | data type for storing sets of (key, value) pairs
(1 row)
```

带有 **hstore** 不可用的输出示例：

```
name | default_version | installed_version | comment
-----+-----+-----+-----
(0 rows)
```

2. 在基于 RHEL 的服务器上，**hstore** 扩展包含在 **postgresql-contrib** RPM 软件包中，该软件包在安装 PostgreSQL 服务器 RPM 软件包时不会自动安装。
要安装 RPM 软件包，请使用以下命令：

```
dnf install postgresql-contrib
```

3. 使用以下命令，在自动化中心数据库上创建 **hstore** PostgreSQL 扩展：

```
$ psql -d <automation hub database> -c "CREATE EXTENSION hstore;"
```

输出：

```
CREATE EXTENSION
```

4. 在以下输出中，**installed_version** 字段包含使用的 **hstore** 扩展，表示启用了 **hstore**。

```
name | default_version | installed_version | comment
-----+-----+-----+-----
hstore | 1.7            | 1.7              | data type for storing sets of (key, value) pairs
(1 row)
```

4.5.3. Ansible Automation Platform PostgreSQL 数据库的基准测试存储性能

使用 Flexible I/O Tester (FIO) 工具检查是否满足最低 Ansible Automation Platform PostgreSQL 数据库要求。FIO 是一个用于对存储系统的读取和写入 IOPS 性能进行基准测试的工具。

先决条件

- 您已安装了 Flexible I/O Tester (**fio**) 存储性能基准工具。
要安装 **fio**，请以 root 用户身份运行以下命令：

```
# yum -y install fio
```

- 您有足够的磁盘空间来存储 **fio** 测试数据日志文件。
该流程中显示的示例至少需要 60GB 磁盘空间在 **/tmp** 目录中：
 - **numjobs** 设置由命令运行的作业数量。
 - **size=10G** 设置每个作业生成的文件大小。
- 您已调整了 **size** 参数的值。调整此值可减少测试数据量。

流程

1. 运行随机写入测试：

```
$ fio --name=write_iops --directory=/tmp --numjobs=3 --size=10G \  
--time_based --runtime=60s --ramp_time=2s --ioengine=libaio --direct=1 \  
--verify=0 --bs=4K --iodepth=64 --rw=randwrite \  
--group_reporting=1 > /tmp/fio_benchmark_write_iops.log \  
2>> /tmp/fio_write_iops_error.log
```

2. 运行随机读测试：

```
$ fio --name=read_iops --directory=/tmp \  
--numjobs=3 --size=10G --time_based --runtime=60s --ramp_time=2s \  
--ioengine=libaio --direct=1 --verify=0 --bs=4K --iodepth=64 --rw=randread \  
--group_reporting=1 > /tmp/fio_benchmark_read_iops.log \  
2>> /tmp/fio_read_iops_error.log
```

3. 查看结果：

在基准命令编写的日志文件中，搜索以 **iops** 开头的行。此行显示测试的最小、最大值和平均值。

以下示例显示了日志文件中随机读取测试的行：

```
$ cat /tmp/fio_benchmark_read_iops.log  
read_iops: (g=0): rw=randread, bs=(R) 4096B-4096B, (W) 4096B-4096B,  
ioengine=libaio, iodepth=64  
[...]  
iops      : min=50879, max=61603, avg=56221.33, stdev=679.97, samples=360  
[...]
```

您必须根据您的业务需求、应用程序工作负载和新需求审核、监控和重新查看日志文件。

端口	协议	Service	Source	目的地	需要的目的	安装程序清单变量
22	TCP	SSH	安装程序节点	Automation hub	安装（临时）	ansible_port
22	TCP	SSH	安装程序节点	控制器节点	安装（临时）	ansible_port
22	TCP	SSH	安装程序节点	EDA 节点	安装（临时）	ansible_port
22	TCP	SSH	安装程序节点	执行节点	安装（临时）	ansible_port
22	TCP	SSH	安装程序节点	hop 节点	安装（临时）	ansible_port
22	TCP	SSH	安装程序节点	混合节点	安装（临时）	ansible_port
22	TCP	SSH	安装程序节点	PostgreSQL 数据库	在安装过程中远程访问（临时）	pg_port
80/443	TCP	HTTP/HTTPS	安装程序节点	Automation hub	在使用捆绑包安装程序时，允许安装程序节点将执行环境镜像推送到自动化中心。	固定的值
80/443	TCP	HTTP/HTTPS	执行节点	Automation hub	允许执行节点从自动化中心拉取执行环境镜像。	固定的值
443	TCP	HTTPS	控制器节点	客户端	Web UI/API	nginx_https_port
443	TCP	HTTPS	控制器节点	OpenShift Container Platform	仅在使用容器组运行作业时才需要。	OpenShift API 服务器的主机名
5432	TCP	PostgreSQL	控制器节点	PostgreSQL 数据库	仅在将内部数据库与另一个组件一起使用时才打开。否则，不应打开此端口。	automationcontroller_pg_port

端口	协议	Service	Source	目的地	需要的目的	安装程序清单变量
5432	TCP	PostgreSQL	EDA 节点	PostgreSQL 数据库	仅在将内部数据库与另一个组件一起使用时才打开。否则，不应打开此端口。	automationedac ontroller_pg_port
5432	TCP	PostgreSQL	Automation hub	PostgreSQL 数据库	仅在将内部数据库与另一个组件一起使用时才打开。否则，不应打开此端口。	automationhub_ pg_port
27199	TCP	Receptor	控制器节点	执行节点	可配置 与控制器直接对等的 Mesh 节点。 涉及的直接节点。27199 通信可以是同时用于执行节点的方法（取决于安装清单）	receptor_listene r_port peers
27199	TCP	Receptor	控制器节点	hop 节点	可配置 如果通过跃点节点转发，从跃点节点到 Receptor 端口的ENABLE 连接。	receptor_listene r_port peers
27199	TCP	Receptor	控制器节点	混合节点	可配置 如果通过非连接的节点转发，从控制器到 Receptor 端口的ENABLE 连接。	receptor_listene r_port peers
27199	TCP	Receptor	执行节点	hop 节点	可配置 Mesh 27199 通信可以是这两个方法（取决于安装清单）用于执行节点 从控制器到 receptor 端口的ALLOW 连接	receptor_listene r_port peers

端口	协议	Service	Source	目的地	需要的目的	安装程序清单变量
27199	TCP	Receptor	执行节点	控制器节点	可配置 Mesh 27199 通信可以是这两个方法（取决于安装清单）用于执行节点 从控制器到 receptor 端口的 ALLOW 连接	receptor_listener_port peers



注意

- 混合节点充当控制和执行节点的组合，因此混合节点共享这两者的连接。
- 如果定义了 **receptor_listener_port**，机器还需要一个可用的开放端口来建立入站 TCP 连接，例如 27199。

表 5.2. Red Hat Insights for Red Hat Ansible Automation Platform

URL	需要的目的
https://api.access.redhat.com:443	通用帐户服务、订阅
https://cert-api.access.redhat.com:443	Insights 数据上传
https://cert.console.redhat.com:443	清单上传和云连接器连接
https://console.redhat.com:443	访问 Insights 仪表盘

表 5.3. Automation Hub

URL	需要的目的
https://console.redhat.com:443	通用帐户服务、订阅
https://catalog.redhat.com:443	索引执行环境
https://sso.redhat.com:443	TCP
https://automation-hub-prd.s3.amazonaws.com:443 https://automation-hub-prd.s3.us-east-2.amazonaws.com:443	防火墙访问
https://galaxy.ansible.com:443	Ansible 社区策展的 Ansible 内容

URL	需要的目的
https://ansible-galaxy-ng.s3.dualstack.us-east-1.amazonaws.com:443	为社区策展的 Ansible 内容存储库的双堆栈 IPv6 端点
https://registry.redhat.io:443	访问由红帽和合作伙伴提供的容器镜像
https://cert.console.redhat.com:443	红帽和合作伙伴策展的 Ansible 集合

表 5.4. 执行环境 (EE)

URL	需要的目的
https://registry.redhat.io:443	访问由红帽和合作伙伴提供的容器镜像
cdn.quay.io:443	访问由红帽和合作伙伴提供的容器镜像
cdn01.quay.io:443	访问由红帽和合作伙伴提供的容器镜像
cdn02.quay.io:443	访问由红帽和合作伙伴提供的容器镜像
cdn03.quay.io:443	访问由红帽和合作伙伴提供的容器镜像

重要

镜像清单和文件系统 Blob 直接从 **registry.redhat.io** 提供。但是，从 2023 年 5 月 1 日开始，文件系统 Blob 是从 **quay.io** 提供的。为了避免拉取容器镜像出现问题，您必须启用到列出的 **quay.io** 主机名的出站连接。

此更改应该对专门用于启用到 **registry.redhat.io** 的出站连接的任何防火墙配置进行。

在配置防火墙规则时使用主机名而不是 IP 地址。

完成此更改后，您可以继续从 **registry.redhat.io** 拉取镜像。您不需要 **quay.io** 登录，或以任何方式直接与 **quay.io** registry 交互，以继续拉取红帽容器镜像。

如需更多信息，[请参阅容器镜像拉取的防火墙更改](#)。

第 6 章 附加 RED HAT ANSIBLE AUTOMATION PLATFORM 订阅

在安装 Red Hat Ansible Automation Platform 前，您**必须**在所有节点上附加了有效的订阅。附加 Ansible Automation Platform 订阅可让您访问继续安装所需的仅订阅资源。



注意

如果您在红帽帐户中启用了简单内容访问模式，则不需要附加订阅。启用后，您需要在安装 Ansible Automation Platform 前将您的系统注册到 Red Hat Subscription Management(RHSM)或 Satellite。如需更多信息，请参阅[简单内容访问](#)。

流程

1. 获取 Red Hat Ansible Automation Platform 订阅的 **pool_id** :

```
# subscription-manager list --available --all | grep "Ansible Automation Platform" -B 3 -A 6
```



注意

不要将 MCT4022 用作订阅的 **pool_id**，因为它可能会导致 Ansible Automation Platform 订阅附加失败。

示例

subscription-manager list 命令的输出示例。获取 **Pool ID:** 部分中的 **pool_id** :

```
Subscription Name: Red Hat Ansible Automation, Premium (5000 Managed Nodes)
Provides: Red Hat Ansible Engine
Red Hat Ansible Automation Platform
SKU: MCT3695
Contract: ````
Pool ID: <pool_id>
Provides Management: No
Available: 4999
Suggested: 1
```

2. 附加订阅 :

```
# subscription-manager attach --pool=<pool_id>
```

现在，您已将 Red Hat Ansible Automation Platform 订阅附加到所有节点。

验证

- 验证订阅是否已成功附加 :

```
# subscription-manager list --consumed
```

故障排除

- 如果您无法找到与 Ansible Automation Platform 安装程序捆绑的特定软件包，或者如果您看到**配置消息禁用了 Repositories**，请尝试使用以下命令启用存储库 :

Red Hat Ansible Automation Platform 2.4 for RHEL 8

```
subscription-manager repos --enable ansible-automation-platform-2.4-for-rhel-8-x86_64-rpms
```

Red Hat Ansible Automation Platform 2.4 for RHEL 9

```
subscription-manager repos --enable ansible-automation-platform-2.4-for-rhel-9-x86_64-rpms
```

第 7 章 选择并获取 RED HAT ANSIBLE AUTOMATION PLATFORM 安装程序

根据您的 Red Hat Enterprise Linux 环境互联网连接，选择您需要的 Red Hat Ansible Automation Platform 安装程序。查看以下场景，以确定哪个 Red Hat Ansible Automation Platform 安装程序满足您的需要。

7.1. 使用互联网访问进行安装

如果您的 Red Hat Enterprise Linux 环境连接到互联网，请选择 Red Hat Ansible Automation Platform 安装程序。使用互联网访问进行安装会检索最新的软件仓库、软件包和依赖项。选择以下方法之一来设置 Ansible Automation Platform 安装程序。

Tarball 安装

1. 进入 [Red Hat Ansible Automation Platform 下载](#) 页面。
2. 为 **Ansible Automation Platform <latest-version> Setup** 点 **Download Now**。
3. 解压文件：

```
$ tar xvzf ansible-automation-platform-setup-<latest-version>.tar.gz
```

RPM 安装

1. 安装 Ansible Automation Platform 安装程序软件包
v.2.4 for RHEL 8 for x86_64

```
$ sudo dnf install --enablerepo=ansible-automation-platform-2.4-for-rhel-8-x86_64-rpms  
ansible-automation-platform-installer
```

v.2.4 for RHEL 9 for x86-64

```
$ sudo dnf install --enablerepo=ansible-automation-platform-2.4-for-rhel-9-x86_64-rpms  
ansible-automation-platform-installer
```



注意

dnf install 启用存储库，因为默认禁用存储库。

使用 RPM 安装程序时，文件位于 `/opt/ansible-automation-platform/installer` 目录下。

7.2. 在没有互联网访问的情况下安装

如果您无法访问互联网，或者不想从在线存储库安装独立的组件和依赖项，请使用 Red Hat Ansible Automation **PlatformBundle** 安装程序。仍然需要访问 Red Hat Enterprise Linux 软件仓库。所有其他依赖项都包含在 tar 归档中。

流程

1. 进入 [Red Hat Ansible Automation Platform 下载](#) 页面。

2. 为 **Ansible Automation Platform <latest-version> Setup Bundle**点 **Download Now**。

3. 解压文件：

```
$ tar xvzf ansible-automation-platform-setup-bundle-<latest-version>.tar.gz
```

第 8 章 关于安装程序清单文件

Red Hat Ansible Automation Platform 使用清单文件，根据您以逻辑方式组织的基础架构中的受管节点或主机列表进行工作。您可以使用 Red Hat Ansible Automation Platform 安装程序清单文件指定您的安装场景，并描述 Ansible 的主机部署。通过使用清单文件，Ansible 可以通过一个命令管理大量主机。清单还可以通过减少您指定的命令行选项数目来更有效地使用 Ansible。

根据您拥有的清单插件，清单文件可以采用多种格式。最常见的格式是 **INI** 和 **YAML**。本文档中列出的清单文件以 INI 格式显示。

清单文件的位置取决于您使用的安装程序。下表显示了可能的位置：

安装程序	位置
捆绑包 tar	<code>/ansible-automation-platform-setup-bundle-<latest-version></code>
非捆绑包 tar	<code>/ansible-automation-platform-setup-<latest-version></code>
RPM	<code>/opt/ansible-automation-platform/installer</code>

您可以使用以下命令验证清单中的主机：

```
ansible all -i <path-to-inventory-file> --list-hosts
```

清单文件示例

```
[automationcontroller]
host1.example.com
host2.example.com
Host4.example.com

[automationhub]
host3.example.com

[database]
Host5.example.com

[all:vars]
admin_password='<password>'

pg_host=""
pg_port=""

pg_database='awx'
pg_username='awx'
pg_password='<password>'

registry_url='registry.redhat.io'
registry_username='<registry username>'
registry_password='<registry password>'
```

清单文件的第一个部分指定 Ansible 可使用的主机或主机组。

8.1. 主机和组指南

数据库

- 使用外部数据库时，请确保正确设置清单文件的 **[database]** 部分。
- 要提高性能，请不要在同一服务器上并置数据库和自动化控制器。

Automation hub

- 如果有 **[automationhub]** 组，则必须包含变量 **automationhub_pg_host** 和 **automationhub_pg_port**。
- 在 **[automationhub]** 组中添加 Ansible Automation hub 信息。
- 不要在同一节点上安装 Ansible Automation hub 和自动化控制器。
- 为 **[automationhub]** 和 **[automationcontroller]** 主机提供可访问的 IP 地址或完全限定域名 (FQDN)，以确保用户可以从不同节点同步并安装 Ansible 自动化中心和自动化控制器的内容。FQDN 不得包含 - 或 _ 符号，因为它无法正确处理。

不要使用 **localhost**。

私有自动化中心

- 不要在同一节点上安装私有自动化中心和自动化控制器。
- 您可以使用相同的 PostgreSQL（数据库）实例，但它们必须使用不同的（数据库）名称。
- 如果您从内部地址安装私有自动化中心，并且有一个只包含外部地址的证书，可能会导致安装在没有证书问题的情况下作为容器 registry 使用。



重要

您必须分隔自动化控制器和 Ansible 自动化中心的安装，因为 **[database]** 组无法区分这两个（如果同时安装了这两个组）。

如果您在 **[database]** 和自动化控制器和 Ansible 自动化中心中使用一个值，则它们将使用相同的数据库。

自动化控制器

- 自动化控制器不会为它使用的数据库配置复制或故障转移。
- 自动化控制器可以和已有的复制一起工作。

Event-Driven Ansible 控制器

- event-Driven Ansible 控制器必须安装在单独的服务器上，且不能安装到与自动化中心和自动化控制器相同的主机上。

集群安装

- 在升级现有集群时，您还可以重新配置集群来忽略现有的实例或实例组。从清单文件中省略实例或实例组不足以将其从集群中移除。除了从清单文件中省略实例或实例组外，还必须在开始升级

前取消置备实例或实例组。如需更多信息，请参阅 [取消置备节点或组](#)。否则，忽略的实例或实例组会继续与集群通信，这可能会在升级过程中造成自动化控制器服务的问题。

- 如果要创建集群安装设置，则必须使用所有实例的主机名或 IP 地址替换 `[localhost]`。自动化控制器和自动化中心的安装程序不接受 `[localhost]` 所有节点，实例必须能够使用此主机名或地址访问任何其他节点。您不能在其中一个节点上使用 `localhost ansible_connection=local`。对所有节点的主机名使用相同的格式。

因此，这无法正常工作：

```
[automationhub]
localhost ansible_connection=local
hostA
hostB.example.com
172.27.0.4
```

需要使用以下格式：

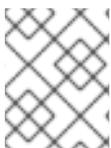
```
[automationhub]
hostA
hostB
hostC
```

或者

```
[automationhub]
hostA.example.com
hostB.example.com
hostC.example.com
```

8.2. 取消置备节点或组

您可以使用 Ansible Automation Platform 安装程序取消置备节点和实例组。运行安装程序将删除附加到组中节点的所有配置文件和日志。



注意

您可以取消置备清单中的任何主机，但 `[automationcontroller]` 组中指定的第一个主机除外。

要取消置备节点，请将 `node_state=deprovision` 附加到清单文件中的节点或组。

例如：

从部署中删除单个节点：

```
[automationcontroller]
host1.example.com
host2.example.com
host4.example.com node_state=deprovision
```

或者

从部署中删除整个实例组：

```
[instance_group_restrictedzone]
host4.example.com
host5.example.com

[instance_group_restrictedzone:vars]
node_state=deprovision
```

8.3. 清单变量

示例清单文件的第二部分（在 `[all:vars]` 后面）是安装程序使用的变量列表。使用 `all` 表示变量适用于所有主机。

要将变量应用到特定的主机，请使用 `[hostname:vars]`。例如，`[automationhub:vars]`。

8.4. 在清单文件中声明变量的规则

字符串变量的值在引号中声明。例如：

```
pg_database='awx'
pg_username='awx'
pg_password='<password>'
```

在 `:vars` 部分中声明时，INI 值将解释为字符串。例如，`var=FALSE` 创建一个等于 `FALSE` 的字符串。与主机行不同，`:vars` 部分每行只接受一个条目，因此 `=` 后的所有内容都必须是条目的值。主机行接受每行多个 `key=value` 参数。因此，它们需要使用一种方式来指出空格是值的一部分而不是分隔符。包含空格的值可以加上引号（单引号或双引号）。如需更多信息，请参阅 [Python shlex 解析规则](#)。

如果 INI 清单中设置的变量值必须是特定的类型（如字符串或布尔值），则始终在任务中使用过滤器指定类型。在消耗变量时，请勿依赖 INI 清单中设置的类型。



注意

考虑将 YAML 格式用于清单源，以避免在变量的实际类型上产生混淆。YAML 清单插件会一致且正确处理变量值。

如果 Ansible 清单文件中的参数值包含特殊字符，如 `#`、`{` 或 `}`，则必须双重转义（用单引号和双引号包括该值）。

例如，要将 `mypasswordwith#hashsigns` 用作变量 `pg_password` 的值，在 Ansible 主机清单文件中请将其声明为 `pg_password="mypasswordwith#hashsigns"`。

8.5. 保护清单文件中的 SECRET

您可以使用 Ansible Vault 加密敏感或机密的变量。但是，加密变量名称和变量值会难以查找值的来源。要绕过这个问题，您可以使用 `ansible-vault encrypt_string` 单独加密变量，或者加密包含变量的文件。

流程

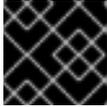
1. 创建一个标记为 `credentials.yml` 的文件来存储加密的凭证。

```
$ cat credentials.yml
```

```
admin_password: my_long_admin_pw
pg_password: my_long_pg_pw
registry_password: my_long_registry_pw
```

2. 使用 **ansible-vault** 加密 **credentials.yml** 文件。

```
$ ansible-vault encrypt credentials.yml
New Vault password:
Confirm New Vault password:
Encryption successful
```



重要

将加密的 vault 密码存储在安全的地方。

3. 验证 **credentials.yml** 文件是否加密。

```
$ cat credentials.yml
$ANSIBLE_VAULT;1.1;
AES256363836396535623865343163333339613833363064653364656138313534353135303
764646165393765393063303065323466663330646232363065316666310a37306230313337
633963383130303334313534383962613632303761636632623932653062343839613639653
6356433656162333133653636616639313864300a3532393734333133396134653263393130
356335653534643565386536316334643438353464323766386235336136663261363433323
131633436393939646132656164333634306335343039356462646330343839663362323033
65383763
```

4. 运行 **setup.sh** 以安装 Ansible Automation Platform 2.4，并传递 **credentials.yml** 和 **--ask-vault-pass** 选项。

```
$ ANSIBLE_BECOME_METHOD='sudo' ANSIBLE_BECOME=True
ANSIBLE_HOST_KEY_CHECKING=False ./setup.sh -e @credentials.yml -- --ask-vault-pass
```

8.6. 其他清单文件变量

您可以通过在清单文件中包含额外变量来进一步配置 Red Hat Ansible Automation Platform 安装。这些配置添加了用于管理 Red Hat Ansible Automation Platform 的各种可选功能。使用文本编辑器编辑清单文件来添加这些变量。

清单文件变量的预定义值表可在 *Red Hat Ansible Automation Platform 安装指南* 中的 [清单文件变量](#) 中找到。

第 9 章 支持的安装场景

红帽支持以下 Red Hat Ansible Automation Platform 安装场景：

其他资源

要编辑清单文件参数以指定受 [支持的安装场景](#)，请参阅 [Red Hat Ansible Automation Platform 安装指南中的基于安装场景的清单文件示例](#)。

9.1. 带有同一节点上的数据库的独立自动化控制器，或者一个非安装程序管理的数据库

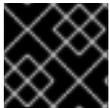
此场景包括安装自动化控制器，包括 Web 前端、REST API 后端和数据库。它安装 PostgreSQL，并将自动化控制器配置为将其用作其数据库。这被视为标准自动化控制器安装场景。

9.2. 带有外部管理数据库的独立自动化控制器

此场景包括在一台机器上安装自动化控制器服务器，并将与远程 PostgreSQL 实例的通信配置为其数据库。此远程 PostgreSQL 可以是您管理的服务器，也可以由 Amazon RDS 等云服务提供。

9.3. 带有内部数据库的单个 EVENT-DRIVEN ANSIBLE 控制器节点

此场景包括在具有内部数据库的单一机器上安装 Event-Driven Ansible 控制器。它安装一个与自动化控制器安装场景类似的安装程序管理的 PostgreSQL。



重要

在使用适当的 Event-Driven Ansible 变量填充清单文件前，必须安装自动化控制器。

其他资源

- [带有外部（安装程序管理的）数据库的单一自动化控制器、单一自动化中心和单个 Event-Driven Ansible 控制器节点](#)
- [附录 A.5.Event-Driven Ansible 控制器变量](#)

9.4. 带有同一节点上的数据库的独立自动化中心，或者一个非安装程序管理的数据库

此场景包括安装自动化中心，包括 Web 前端、REST API 后端和单一计算机上的数据库。它安装 PostgreSQL，并将自动化中心配置为将其用作其数据库。

9.5. 带有外部管理数据库的独立自动化中心

此场景包括在一台机器上安装自动化中心服务器，并安装由 Red Hat Ansible Automation Platform 安装程序管理的远程 PostgreSQL 数据库。

9.6. 在自动化控制器节点或非安装程序管理数据库上使用数据库进行平台安装

这个场景包括在自动化控制器节点上安装带有数据库的自动化控制器和自动化中心，或者一个非安装程序管理的数据库。

9.7. 带有外部管理数据库的平台安装

此场景包括安装自动化控制器和自动化中心，并将与远程 PostgreSQL 实例的通信配置为其数据库。此远程 PostgreSQL 可以是您管理的服务器，也可以由 Amazon RDS 等云服务提供。

9.8. 使用外部管理的数据库安装多机器集群

此场景包括安装多个自动化控制器节点和一个自动化中心实例，并将与远程 PostgreSQL 实例的通信配置为其数据库。此远程 PostgreSQL 可以是您管理的服务器，也可以由 Amazon RDS 等云服务提供。在这种情况下，所有自动化控制器都是活跃的，并可执行作业，任何节点都可以接收 HTTP 请求。



注意

- 在集群设置中运行需要自动化控制器使用外部的 PostgreSQL 数据库安装到不是主或从 tower 节点之一的机器上。在带有冗余功能的设置中，远程 PostgreSQL 版本要求为 **PostgreSQL 13**。
 - 有关配置集群设置的更多信息，请参阅[集群](#)。
- 为 **[automationhub]** 主机提供一个可访问的 IP 地址，以确保用户可以从不同节点的 Private Automation Hub 同步内容。