Red Hat Ansible Automation Platform 2.4

Red Hat Ansible Automation Platform 强化指南

以安全的方式安装、配置和维护在 Red Hat Enterprise Linux 上运行的 Ansible Automation Platform。

Last Updated: 2024-06-17

Red Hat Ansible Automation Platform 2.4 Red Hat Ansible Automation Platform 强化指南

以安全的方式安装、配置和维护在 Red Hat Enterprise Linux 上运行的 Ansible Automation Platform。

法律通告

摘要

本指南为在 Red Hat Enterprise Linux 上安装、配置和维护 Ansible Automation Platform 所需的各种进程提供推荐的实践。

目录

前言

本指南为在 Red Hat Enterprise Linux 上安装、配置和维护 Ansible Automation Platform 所需的各种进程提供推荐的实践。

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 强化 ANSIBLE AUTOMATION PLATFORM 简介

本文档提供了有关在 Red Hat Enterprise Linux 上部署 Red Hat Ansible Automation Platform 部署过程中改进 Red Hat Ansible Automation Platform 部署的安全形象（称为"强化"）。

其他部署目标（如 OpenShift）目前不在本指南范围内。通过云服务供应商市场提供的 Ansible Automation Platform 托管服务也不在本指南范围内。

本指南采取一种实用的方法来强化 Ansible Automation Platform 安全，从部署的规划和架构阶段开始，然后介绍安装、初始配置和第 2 天操作的具体指导。由于本指南专门涵盖在 Red Hat Enterprise Linux 上运行的 Ansible Automation Platform，Red Hat Enterprise Linux 的强化指导将包括在影响自动化平台组件的地方。针对国防信息系统局(DISA)安全技术实施指南(STIG)的额外注意事项是为那些集成 DISA STIG 作为其整体安全策略一部分的机构。

注意

这些建议不保证部署 Ansible Automation Platform 的安全性或合规性。您必须评估机构的唯一要求中的安全性，以解决特定的威胁和风险，并根据实施因素实现平衡。

1.1. 受众

本指南面向在 Red Hat Enterprise Linux 上部署时负责安装、配置和维护 Ansible Automation Platform 2.4 的人员。为安全操作、合规评估以及与相关安全流程关联的其他功能提供了其他信息。

1.2. ANSIBLE AUTOMATION PLATFORM 概述

Ansible 是一个使用 Python 编写的开源命令行 IT 自动化软件应用程序。您可以使用 Ansible Automation Platform 配置系统、部署软件并编配高级工作流来支持应用程序部署、系统更新等。Ansible 的主要优势是简洁且易于使用。它还非常注重安全性和可靠性，包含最少的移动部分。它使用安全、知名的通信协议（如 SSH、HTTPS 和 WinRM）进行传输，并使用人类可读的语言，这些语言可在无需大量培训的情况下快速开始。

Ansible Automation Platform 使用企业级功能（如基于角色的访问控制(RBAC)、集中日志记录和审计、凭据管理、作业调度和复杂自动化工作流）增强了 Ansible 语言。使用 Ansible Automation Platform，您可以从我们强大的合作伙伴生态系统获取认证内容；添加安全、报告和分析；以及生命周期技术支持，以跨机构扩展自动化。Ansible Automation Platform 简化了自动化工作负载的开发和操作，用于管理企业应用程序基础架构生命周期。它可在跨多个 IT 域间工作，包括操作、网络、安全、开发以及跨混合环境。

1.2.1. Ansible Automation Platform 组件

Ansible Automation Platform 是一个模块化平台，包括自动化控制器、自动化中心、Event-Driven Ansible 控制器和 Insights for Ansible Automation Platform。

其他资源

如需有关 Ansible Automation Platform 中提供的组件的更多信息，请参阅 Red Hat Ansible Automation Platform *计划指南中的 Red Hat Ansible Automation Platform 组件*。

第 2 章 强化 ANSIBLE AUTOMATION PLATFORM

本指南采取一种实用的方法来强化 Ansible Automation Platform 安全，从部署的规划和架构阶段开始，然后介绍安装阶段的特定指导。由于本指南专门涵盖在 Red Hat Enterprise Linux 上运行的 Ansible Automation Platform，Red Hat Enterprise Linux 的强化指导将包括在影响自动化平台组件的地方。

2.1. 规划注意事项

在规划 Ansible Automation Platform 安装时，请确保包含以下组件：

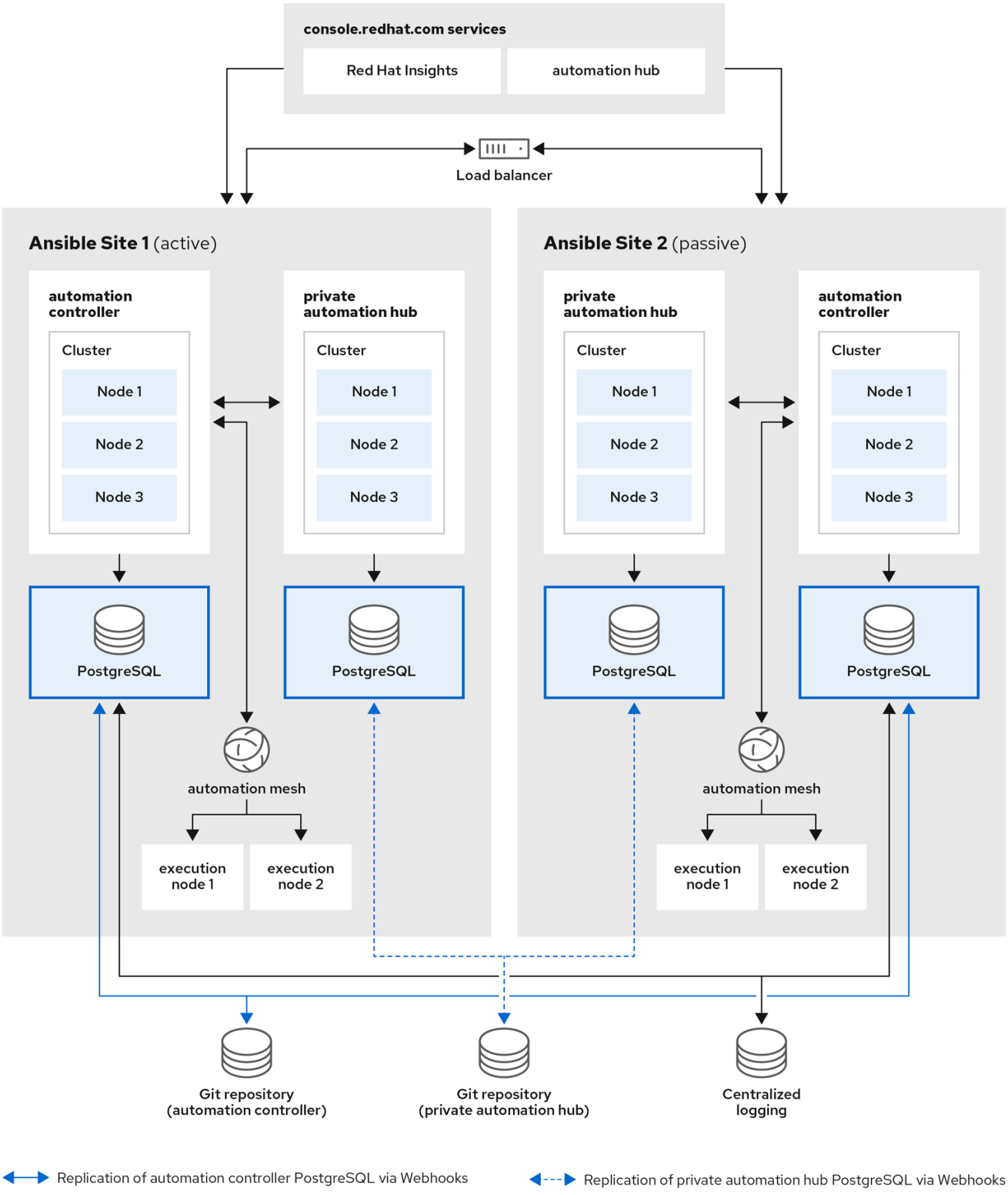
- 安装程序多个组件
  - 自动化控制器
  - Event-Driven Ansible 控制器
  - 私有自动化中心
- PostgreSQL 数据库（如果不是外部）
  - 外部服务
  - Red Hat Insights for Red Hat Ansible Automation Platform
  - Automation hub
  - **registry.redhat.io** （默认执行环境容器 registry）

如需更多信息，请参阅 *Red Hat Ansible Automation Platform 规划指南* 中的系统要求部分。https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.4/html/red_hat system-requirements

2.1.1. Ansible Automation Platform 参考架构

对于具有可用性要求的大型生产环境，本指南建议使用 Red Hat Enterprise Linux 上的 Red Hat Ansible Automation Platform 参考架构 文档中的说明部署本指南第 2.1 节中所述的组件。虽然某些变化可能对您的特定技术要求有意义，但遵循参考架构会导致受支持的生产环境就绪环境。

图 2.1. 参考架构概述



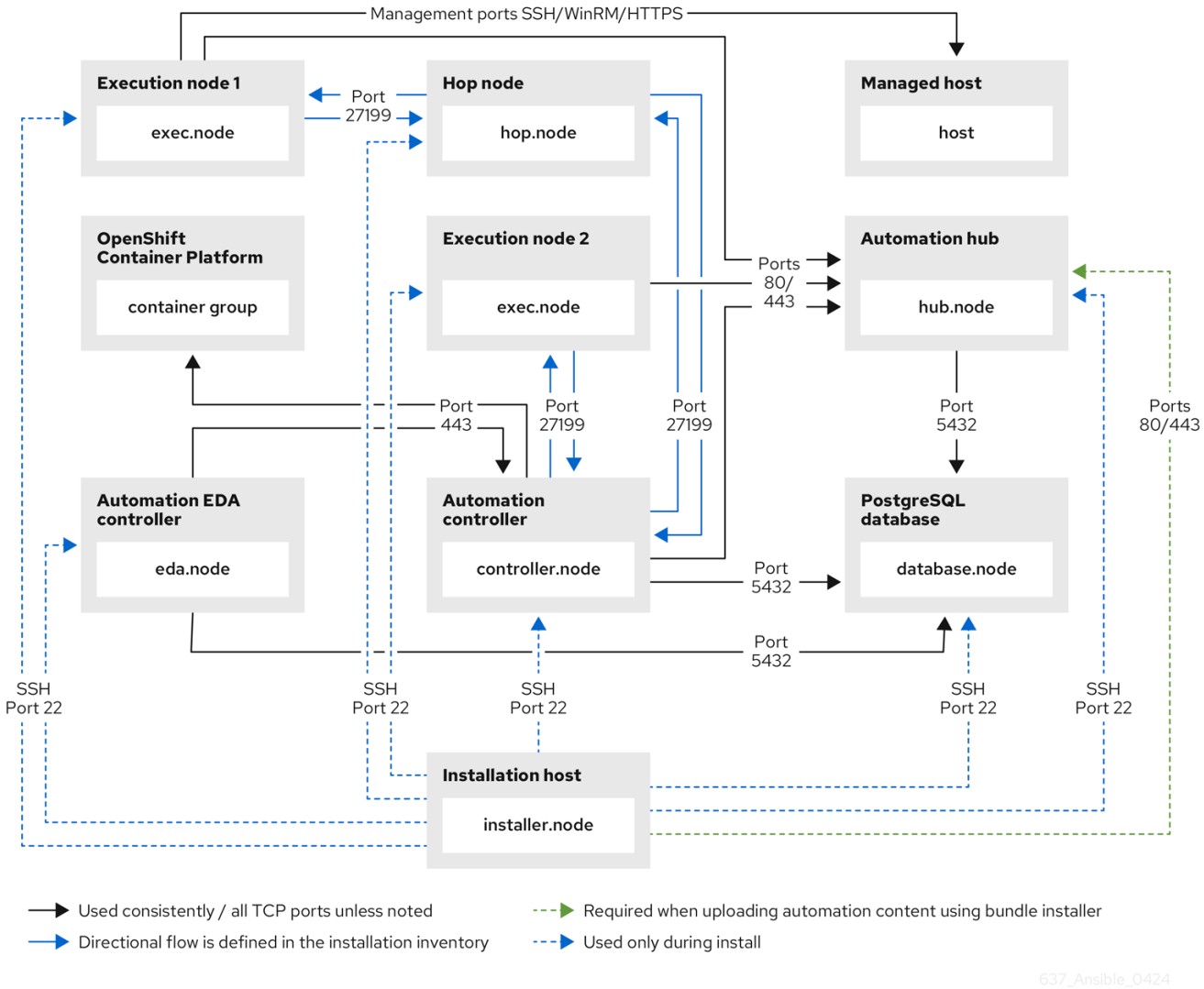
event-Driven Ansible 是 Ansible Automation Platform 2.4 的一个新功能，在图 1 中详述的参考架构时不可用：最初编写了参考架构概述。目前，支持的配置是单个自动化控制器、单一自动化中心和带有外部（安装程序管理的）数据库的 Ansible 控制器节点。对于对 Event-Driven Ansible 感兴趣的机构，建议根据 Ansible Automation Platform 安装指南中的配置 来安装。当需要 Event-Driven Ansible 特定强化配置时，本文档提供了额外的说明。

对于可能不需要完整参考架构的较小的生产环境，本指南建议使用专用 PostgreSQL 数据库服务器部署 Ansible Automation Platform，无论是由安装程序管理还是在外部提供。

2.1.2. 用于 Ansible Automation Platform 的网络、防火墙和网络服务规划

Ansible Automation Platform 需要访问网络以集成外部辅助服务，并管理目标环境和资源，如主机、其他网络设备、应用程序、云服务。Ansible Automation Platform 计划指南中的网络 端口和协议 部分描述了 Ansible Automation Platform 组件如何与网络进行交互，以及使用哪些端口和协议，如下图所示：

图 2.2. Ansible Automation Platform 网络端口和协议



在规划与 Ansible Automation Platform 相关的防火墙或云网络安全组配置时，请参阅 Ansible Automation Platform 规划指南中的网络 端口和协议 部分，以了解防火墙或安全组中需要打开哪些网络端口。

有关使用负载均衡器的更多信息，以及与 Ansible Automation Platform 兼容的服务的传出流量要求。请参阅红帽知识库文章 Ansible Automation Platform 2 服务防火墙中需要打开哪些端口？对于互联网连接的系统，本文档还定义了 Ansible Automation Platform 可以配置为使用的服务的传出流量要求，如 Ansible Automation hub、Red Hat Insights for Red Hat Ansible Automation Platform、Ansible Galaxy、registry.redhat.io 容器镜像 registry 等。

对于互联网连接的系统，本文档还定义了 Ansible Automation Platform 可以配置为使用的服务的传出流量要求，如 Red Hat Automation hub、Insights for Ansible Automation Platform、Ansible Galaxy、registry.redhat.io 容器镜像 registry 等。

将 Ansible Automation Platform 组件使用的端口的访问限制为受保护的网络和客户端。强烈推荐以下限制：

- 限制数据库服务器上的 PostgreSQL 数据库端口(5432)，以便只允许其他 Ansible Automation Platform 组件服务器（自动化控制器、自动化中心、Event-Driven Ansible 控制器）。
- 限制从 安装主机和其他 用于维护对 Ansible Automation Platform 服务器的访问的其他可信系统对 Ansible Automation Platform 服务器的 SSH 访问。
- 限制从可信网络和客户端对自动化控制器、自动化中心和 Event-Driven Ansible 控制器的 HTTPS 访问。

2.1.3. DNS、NTP 和服务规划

2.1.3.1. DNS

安装 Ansible Automation Platform 时，安装程序脚本会检查某些基础架构服务器是否使用安装程序清单中的完全限定域名(FQDN)定义。本指南建议所有 Ansible Automation Platform 基础架构节点在 DNS 中定义有效的 FQDN，它解析为可路由的 IP 地址，且这些 FQDN 用于安装程序清单文件中。

2.1.3.2. DNS 和负载均衡

当将负载均衡器与 Ansible Automation Platform 搭配使用时，如参考架构中所述，每个负载均衡组件（自动化控制器和私有自动化中心）都需要一个额外的 FQDN。

例如，如果在 Ansible Automation Platform 安装程序清单文件中定义了以下主机：

```
[automationcontroller]
controller0.example.com
controller1.example.com
controller2.example.com

[automationhub]
hub0.example.com
hub1.example.com
hub2.example.com
```

然后，负载均衡器可以使用 FQDN **controller.example.com** 和 **hub.example.com** 作为面向用户的这些 Ansible Automation Platform 服务的名称。

当在私有自动化中心前面使用负载均衡器时，安装程序必须了解负载均衡器 FQDN。在安装 Ansible Automation Platform 之前，在安装清单文件中将 **automationhub_main_url** 变量设置为负载均衡器的 FQDN。例如，若要与上例匹配，您要将变量设置为 **automationhub_main_url = hub.example.com**。

2.1.3.3. NTP

配置 Ansible Automation Platform 基础架构中的每个服务器，使时间与 NTP 池或机构的 NTP 服务同步。这样可确保由 Ansible Automation Platform 生成的日志记录和审计事件具有准确的时间戳，并且从自动化控制器执行的任何调度作业都正确执行。

有关为 NTP 同步配置 chrony 服务的详情，请参考 Red Hat Enterprise Linux 文档中的使用 Chrony。

2.1.4. 用户身份验证计划

在计划访问 Ansible Automation Platform 用户界面或 API 时，请注意用户帐户可以是本地的，也可以映射到外部身份验证源，如 LDAP。本指南建议尽可能，所有主用户帐户都应映射到外部身份验证源。使用外部帐户源可消除在此上下文中使用权限时的错误源，并尽可能减少维护专门在 Ansible Automation

Platform 内一组完整的用户的时间。这包括分配给个人个人以及非个人实体的帐户，如用于外部应用程序集成的服务帐户。保留任何本地管理员帐户，如默认的"admin"帐户，用于紧急访问或"breakNote"场景，其中外部身份验证机制不可用。

注意

Event-Driven Ansible 控制器目前不支持外部身份验证，只有本地帐户。

对于运行 Ansible Automation Platform 服务的 Red Hat Enterprise Linux 服务器上的用户帐户，请遵循您的机构策略，以确定单个用户帐户是否是本地或来自外部身份验证源。只有有效需要对 Ansible Automation Platform 组件本身执行维护任务的用户才应被授予对底层 Red Hat Enterprise Linux 服务器的访问权限，因为服务器将具有包含加密密钥和服务密码的配置文件。由于这些个人必须具有维护 Ansible Automation Platform 服务的特权访问权限，因此尽量减少对底层 Red Hat Enterprise Linux 服务器的访问至关重要。不要向不受信任的用户授予对 root 帐户或本地 Ansible Automation Platform 服务帐户(awx、pulp、postgres)的 sudo 访问权限。

注意

本地 Ansible Automation Platform 服务帐户（如 awx、pulp 和 postgres）由 Ansible Automation Platform 安装程序创建和管理。底层 Red Hat Enterprise Linux 主机上的这些特定帐户不能来自外部身份验证源。

2.1.4.1. 自动化控制器身份验证

自动化控制器目前支持以下外部验证机制：

- Azure Activity Directory
- GitHub 单点登录
- Google OAuth2 单点登录
- LDAP
- RADIUS
- SAML
- TACACS+
- 通用 OIDC

选择遵循您机构验证策略的身份验证机制，并参阅 Controller 配置 - 身份验证 文档以了解相关身份验证机制的先决条件。使用的身份验证机制必须确保在 Ansible Automation Platform 和身份验证后端之间与身份验证相关的流量进行加密，当流量在公共或非安全网络上（例如，LDAPS 或 LDAP 通过 TLS，HTTPS 用于 OAuth2 和 SAML 提供程序等）时，会加密。

在自动化控制器中，任何"系统管理员"帐户都可以编辑、更改和更新任何清单或自动化定义。将这些帐户权限限制为对于低级自动化控制器配置和灾难恢复可能的最小用户集合。

2.1.4.2. 私有自动化中心身份验证

私有自动化中心目前支持以下外部验证机制：

- Ansible Automation Platform 中央身份验证（基于 RHSSO）

- LDAP

对于生产环境，LDAP 是私有自动化中心的首选外部身份验证机制。Ansible Automation Platform 中央身份验证是一个选项，可通过 Ansible Automation Platform 安装程序部署，但它仅部署一个中央身份验证服务器实例，使其成为潜在的单点故障。在生产环境中不推荐用于 Ansible Automation Platform 中央身份验证的独立模式。但是，如果您已在生产环境中部署了单独的 Red Hat Single Sign-On (RHSSO)产品，它可用作私有自动化中心的外部身份验证源。

Ansible Automation Platform 安装程序在安装过程中为私有自动化中心配置 LDAP 身份验证。如需更多信息，请参阅 私有自动化中心上的 LDAP 配置。

安装前必须填写以下安装程序清单文件变量：

表 2.1. 用于自动化中心 LDAP 设置的清单变量

变量	详情
automationhub_authentication_backend	设置为"ldap"以使用 LDAP 身份验证。
automationhub_ldap_server_uri	LDAP 服务器 URI，如 "ldap://ldap-server.example.com" 或 "ldaps://ldap-server.example.com:636"。
automationhub_ldap_bind_dn	用于连接到 LDAP 服务器的帐户。此帐户应该是有足够的特权来查询 LDAP 服务器以获取用户和组，但它不应是一个管理员帐户，或者能够修改 LDAP 记录的功能。
automationhub_ldap_bind_password	绑定帐户使用的密码来访问 LDAP 服务器。
automationhub_ldap_user_search_base_dn	用于搜索用户的基本 DN。
automationhub_ldap_group_search_base_dn	用于搜索组的基本 DN。

为确保 LDAP 流量在私有自动化中心和 LDAP 服务器之间加密，LDAP 服务器必须通过 SSL (LDAPS)支持 LDAP over TLS 或 LDAP。

2.1.5. Ansible Automation Platform 的凭证管理规划

自动化控制器使用凭证向作业验证对作业的请求，与清单源同步，并从版本控制系统中导入项目内容。自动化控制器管理三组 secret：

- **用于本地自动化控制器用户的** 用户密码。如需了解更多详细信息，请参阅本指南的 用户身份验证规划 部分。
- **用于自动化控制器的** secret （数据库密码、消息总线密码等）。
- 用于自动化的 secret (SSH 密钥、云凭证、外部密码 vault 凭证等)。

强烈建议您实施特权访问或凭证管理解决方案来保护凭证。组织应审计 的使用，并提供对、访问和特权升级的其他编程控制。

您可以通过确保自动化凭证是唯一的并只存储在自动化控制器中来进一步保护自动化凭证。OpenSSH 等服务可以配置为只允许来自特定地址的连接的凭证。从系统管理员用来登录服务器的用户，使用不同的凭据进行自动化。虽然直接访问应尽可能限制，但可用于灾难恢复或其他临时管理目的，从而可以更轻松地进行审核。

不同的自动化作业可能需要在不同级别上访问系统。例如，您可以有低级系统自动化来应用补丁并执行安全基线检查，同时具有更高级别的自动化部署应用程序。通过将不同的密钥或凭证用于不同的自动化部分，可以最小化任何一个关键漏洞的影响。这也允许轻松进行基准审核。

2.1.5.1. 自动化控制器操作 secret

自动化控制器包含以下在操作中使用的 secret：

表 2.2. 自动化控制器操作 secret

file	详情
/etc/tower/SECRET_KEY	用于加密数据库中自动化 secret 的 secret 密钥。如果 **SECRET_KEY** 发生变化或未知，则无法访问数据库中的加密字段。
/etc/tower/tower.cert /etc/tower/tower.key	自动化控制器 Web 服务的 SSL 证书和密钥。默认安装自签名证书 **/密钥** ；您可以提供本地适当证书和密钥（请参阅 使用用户提供的 PKI 证书安装 ）。
/etc/tower/conf.d/postgres.py	包含自动化控制器用于连接数据库的密码。
/etc/tower/conf.d/channels.py	包含自动化控制器用于 websocket 广播的 secret。

这些 secret 存储在 Automation 控制器服务器上，因为自动化控制器服务必须在启动时以自动的方式读取它们。所有文件都由 Unix 权限保护，仅限于 root 用户或自动化控制器服务用户 awx。通常应监控这些文件，以确保没有未经授权的访问或修改。

注意

自动化控制器以前被命名为 Ansible Tower。这些文件位置保留以前的产品名称。

2.1.5.2. 自动化使用 secret

自动化控制器在数据库中存储各种用于自动化的 secret 或自动化生成的 secret。自动化使用 secret 包括：

- 所有凭证类型的所有 secret 字段（密码、密钥、身份验证令牌、secret 云凭证）。
- 自动化控制器设置中定义的外部服务的 secret 令牌和密码。
- "密码"类型调查字段条目。

您可以向用户和团队授予使用这些凭证的权限，而无需实际向用户公开凭证。这意味着，如果用户移动到不同的团队或离开机构，您不必对所有系统重新密钥。

自动化控制器使用 SSH （或 Windows 等效）连接到远程主机。要将密钥从自动化控制器传递给 SSH，必须在将其写入命名管道前解密密钥。然后，自动化控制器使用该管道将密钥发送到 SSH （因此永远不会写入磁盘）。如果使用密码，自动化控制器会直接响应密码提示并解密密码，然后再将其写入提示。

作为具有超级用户访问权限的管理员，您可以使用类似于 YAML/JSON 的定义以标准格式定义自定义凭证类型，从而启用将新凭证类型分配给作业和清单更新。这可让您定义一个与现有凭证类型类似的自定义凭证类型。例如，您可以创建一个自定义凭证类型，将第三方 Web 服务的 API 令牌注入环境变量，您的 playbook 或自定义清单脚本可以使用它。

要加密 secret 字段，Ansible Automation Platform 使用 CBC 模式的 AES，以及 256 位密钥进行加密、PKCS7 padding 和 HMAC 使用 SHA256 进行身份验证。加密/解密过程从 SECRET_KEY、模型字段的字段名称和数据库分配的记录 ID 生成 AES-256 位加密密钥。因此，如果密钥生成过程中使用的任何属性发生变化，Ansible Automation Platform 无法正确解密 secret。Ansible Automation Platform 设计为，在 playbook Ansible Automation Platform 启动时，SECRET_KEY 永远不会可读，因此 Ansible Automation Platform 用户永远不会读取这些 secret，并且没有通过 Ansible Automation Platform REST API 提供的 secret 字段值。如果 playbook 中使用了 secret 值，则必须对任务使用 no_log，使其不会被意外记录。如需更多信息，请参阅在没有日志的情况下保护敏感数据。

2.1.6. 日志记录和日志捕获

可见性和分析是企业安全和零信任架构的重要支密。日志记录是捕获操作和审核的关键。您可以使用 Red Hat Enterprise Linux 安全强化指南中的内置的 审计 支持来管理日志记录和审核。控制器的内置日志记录和活动流支持自动化控制器和自动日志中的所有更改，以满足审计目的。如需更多信息，请参阅自动化控制器文档中的 日志记录和聚合 部分。

本指南建议您配置 Ansible Automation Platform 和底层 Red Hat Enterprise Linux 系统，以集中收集日志和审核，而不是在本地系统中查看它。自动化控制器必须配置为使用外部日志记录来从控制器服务器中的多个组件编译日志记录。发生的事件必须与时间相关，才能进行准确的诊断分析。这意味着，控制器服务器必须配置有 NTP 服务器，服务器也由日志记录聚合器服务以及控制器的目标。相关性必须满足某些行业容错要求。换句话说，可能有不同的要求，不同日志事件的时间戳不得因 X 秒大于 X 秒而不同。此功能应该在外部日志记录服务中可用。

日志记录的另一个关键功能是使用加密来保护日志工具的完整性。日志数据包括成功记录系统活动所需的所有信息（如日志记录、日志设置和日志报告）。攻击者通常会替换日志工具，或将代码注入现有工具，以便从日志中隐藏或擦除系统活动。要解决这个风险，日志工具必须经过加密签名，以便您可以识别何时修改、操作或替换日志工具。例如，验证日志工具没有被修改、操作或替换的方法是对工具文件使用 checksum 哈希。这样可确保工具的完整性没有被破坏。

2.1.7. 审计和事件检测

在常见用例中应用 NIST Cybersecurity Framework，需要使用 Ansible Automation Platform 来满足安全策略要求，例如：

- 为 Red Hat Enterprise Linux 上的 Web 服务器需要 HTTPS.
- 为 Red Hat Enterprise Linux 上的 Web 服务器和数据库服务器之间的内部通信需要 TLS 加密。
- 生成报告，显示策略已正确部署。
- 监控违反了策略的偏移。
- 自动更正任何策略违反情况。

这可以通过网络安全框架的 5 个步骤完成：

识别

定义根据安全策略实施的要求。

保护

实施并应用要求作为 Ansible Playbook。

DETECT

监控偏移并生成审计报告。

响应

探索在检测到事件时可以执行的操作。

RECOVER

使用 Ansible 将系统恢复到已知的良好配置。

2.1.8. Red Hat Enterprise Linux 主机规划

Ansible Automation Platform 的安全性依赖于底层 Red Hat Enterprise Linux 服务器的配置。因此，每个 Ansible Automation Platform 组件的底层 Red Hat Enterprise Linux 主机都必须根据您的机构使用的安全强化 Red Hat Enterprise Linux 8或 安全强化 （取决于使用什么操作系统）以及任何安全配置集要求(CIS、STIG、HIPAA 等)以及您的机构使用的任何安全配置集要求(CIS、STIG、HIPAA 等)。

请注意，应用 STIG 或其他安全配置集的某些安全控制可能会与 Ansible Automation Platform 支持要求冲突。虽然 Automation controller STIG considerations部分列出了一些示例，但这不是一个完整的列表。要维护受支持的配置，请务必讨论与安全审核员的任何此类冲突，以便了解并批准 Ansible Automation Platform 要求。

2.1.8.1. Ansible Automation Platform 和其他软件

在 Red Hat Enterprise Linux 服务器上安装 Ansible Automation Platform 组件时，Red Hat Enterprise Linux 服务器应该单独使用。除了 Ansible Automation Platform 外，不应安装额外的服务器功能，因为这是一个不受支持的配置，可能会影响 Ansible Automation Platform 软件的安全性和性能。

同样，当 Ansible Automation Platform 部署到 Red Hat Enterprise Linux 主机上时，它会安装如 nginx web 服务器、Pulp 软件存储库和 PostgreSQL 数据库服务器等软件。此软件不应以更通用的方式进行修改或使用（例如，不要使用 nginx 来服务器额外网站内容或 PostgreSQL 来托管其他数据库），因为这是一个不受支持的配置，并可能会影响 Ansible Automation Platform 的安全性和性能。此软件的配置由 Ansible Automation Platform 安装程序管理，在执行升级时可能会撤消任何手动更改。

2.2. 安装

有影响 Ansible Automation Platform 安全状况的安装时决定。安装过程包括设置多个变量，它们与 Ansible Automation Platform 基础架构强化相关。在安装 Ansible Automation Platform 前，请参考本指南的安装部分中的指导。

2.2.1. 从专用安装主机安装

Ansible Automation Platform 安装程序可以从其中一个基础架构服务器运行，如自动化控制器，或者从可通过 SSH 访问 Ansible Automation Platform 基础架构服务器的外部系统运行。Ansible Automation Platform 安装程序也不用于安装，而是用于后续第二天操作，如备份和恢复，以及升级。本指南建议从专用外部服务器执行安装和第二天操作，此处称为安装主机。这样做消除了登录其中一个基础架构服务器以运行这些功能的需求。安装主机必须仅用于管理 Ansible Automation Platform，且不得运行任何其他服务或软件。

安装主机必须是已安装和配置的 Red Hat Enterprise Linux 服务器，根据 Red Hat Enterprise Linux 的安全强化 以及与您的机构相关的任何安全配置文件(CIS、STIG 等)。获取 Ansible Automation Platform 安装程序，如 自动化平台规划指南中所述，按照 Automation Platform 安装指南中所述 创建安装程序清单

文件。此清单文件用于升级、添加基础架构组件和由安装程序执行第二天操作，因此在安装后保留 文件以备将来使用。

对安装主机的访问必须仅限于负责管理 Ansible Automation Platform 基础架构的人员。随着时间的推移，它将包含敏感信息，如安装程序清单（其中包含 Ansible Automation Platform 的初始登录凭据）、用户提供的 PKI 密钥和证书的副本、备份文件等。如果需要基础架构管理和维护，还必须使用安装主机通过 SSH 登录 Ansible Automation Platform 基础架构服务器。

2.2.2. 安装清单中的与安全相关的变量

安装清单文件定义了 Ansible Automation Platform 基础架构的架构，并提供了很多变量，可用于修改基础架构组件的初始配置。如需有关安装程序清单的更多信息，请参阅 Ansible Automation Platform 安装指南。

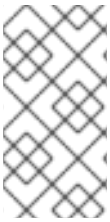
下表列出了一些与安全相关的变量，以及它们的建议值用于创建安装清单。

表 2.3. 与安全相关的清单变量

变量	推荐的值	详情
postgres_use_ssl	true	当设置了此变量时，安装程序会将安装程序管理的 Postgres 数据库配置为接受基于 SSL 的连接。
pg_sslmode	verify-full	默认情况下，当控制器连接到数据库时，它会尝试加密的连接，但不会强制执行它。将此变量设置为 "verify-full" 需要控制器和数据库之间的 mutual TLS 协商。**postgres_use_ssl** 变量还必须设置为 "true"，以便此 **pg_sslmode** 生效。 **注意** ：如果使用第三方数据库而不是安装程序管理的数据库，则必须单独设置第三方数据库以接受 mTLS 连接。
nginx_disable_https	false	如果设置为 "true"，则此变量禁用到控制器的 HTTPS 连接。默认值为 "false"，因此如果安装程序清单中没有此变量，它实际上与将变量明确定义为 "false"。
automationhub_disable_https	false	如果设置为 "true"，则此变量禁用到私有自动化中心的 HTTPS 连接。默认值为 "false"，因此如果安装程序清单中没有此变量，它实际上与将变量明确定义为 "false"。

automationedacontroller_disable_https	false	如果设置为 "true"，则此变量禁用到 Event-Driven Ansible 控制器的 HTTPS 连接。默认值为 "false"，因此如果安装程序清单中没有此变量，它实际上与将变量明确定义为 "false"。

在负载均衡器用于多个控制器或 hub 的参考架构中，SSL 客户端连接可以在负载均衡器终止，或传递给单独的 Ansible Automation Platform 服务器。如果 SSL 在负载均衡器终止，本指南建议流量从负载均衡器重新加密到单个 Ansible Automation Platform 服务器，以确保使用端到端加密。在这种情况下，表 2.3 中列出的 *_disable_https 变量会保留默认值 "false"。



注意

本指南建议在生产环境中使用外部数据库，但对于开发和测试场景，数据库可以并置到自动化控制器上。由于当前的 PostgreSQL 13 限制，当数据库位于自动化控制器上时，设置 pg_sslmode = verify-full 会导致在 TLS 协商过程中验证主机名的错误。在解决此问题前，必须使用外部数据库来确保自动化控制器和数据库之间的 mutual TLS 身份验证。

2.2.3. 使用用户提供的 PKI 证书安装

默认情况下，Ansible Automation Platform 为平台的基础架构组件创建自签名 PKI 证书。如果现有的 PKI 基础架构可用，必须为自动化控制器、私有自动化中心、Event-Driven Ansible 控制器和 postgres 数据库服务器生成证书。将证书文件复制到安装程序目录中，以及用于验证证书的 CA 证书。

使用以下清单变量，以使用新证书配置基础架构组件。

表 2.4. PKI 证书清单变量

变量	详情
custom_ca_cert	安装程序目录中的 CA 证书的文件名。
web_server_ssl_cert	自动化控制器 PKI 证书的文件名，位于安装程序目录中。
web_server_ssl_key	自动化控制器 PKI 密钥的文件名，位于安装程序目录中。
automationhub_ssl_cert	私有自动化中心 PKI 证书的文件名，位于安装程序目录中。
automationhub_ssl_key	私有自动化中心 PKI 密钥的文件名，位于安装程序目录中。
postgres_ssl_cert	数据库服务器 PKI 证书的文件名，位于安装程序目录中。只有在使用第三方数据库时，只有安装程序管理的数据库服务器才需要此变量。

postgres_ssl_key	数据库服务器 PKI 证书的文件名，位于安装程序目录中。只有在使用第三方数据库时，只有安装程序管理的数据库服务器才需要此变量。
automationedacontroller_ssl_cert	Event-Driven Ansible 控制器 PKI 证书的文件名，位于安装程序目录中。
automationedacontroller_ssl_key	Event-Driven Ansible 控制器 PKI 密钥的文件名，位于安装程序目录中。

当使用负载均衡器部署多个自动化控制器时，web_server_ssl_cert 和 web_server_ssl_key 会为每个控制器共享。要防止主机名不匹配，证书的通用名称(CN)必须与负载均衡器使用的 DNS FQDN 匹配。这在部署多个私有自动化中心和 automationhub_ssl_cert 和 automationhub_ssl_key 变量时也适用。如果您的机构策略为每个服务都需要唯一的证书，每个证书都需要一个 Subject Alt Name (SAN)，它与用于负载均衡服务的 DNS FQDN 匹配。要在每个自动化控制器上安装唯一的证书和密钥，安装清单文件中的证书和密钥变量必须定义为每个主机变量，而不是在 [all:vars] 部分中。例如：

```
[automationcontroller]
controller0.example.com web_server_ssl_cert=/path/to/cert0 web_server_ssl_key=/path/to/key0
controller1.example.com web_server_ssl_cert=/path/to/cert1 web_server_ssl_key=/path/to/key1
controller2.example.com web_server_ssl_cert=/path/to/cert2 web_server_ssl_key=/path/to/key2
```

```
[automationhub]
hub0.example.com automationhub_ssl_cert=/path/to/cert0 automationhub_ssl_key=/path/to/key0
hub1.example.com automationhub_ssl_cert=/path/to/cert1 automationhub_ssl_key=/path/to/key1
hub2.example.com automationhub_ssl_cert=/path/to/cert2 automationhub_ssl_key=/path/to/key2
```

2.2.4. 安装清单中的敏感变量

安装清单文件包含多个敏感变量，主要用于设置 Ansible Automation Platform 使用的初始密码，它们通常以纯文本形式保存在清单文件中。为防止未经授权的查看这些变量，您可以将这些变量保存在加密的 Ansible vault 中。要做到这一点，进入安装程序目录并创建 vault 文件：

- **cd /path/to/ansible-automation-platform-setup-bundle-2.4-1-x86_64**
- **ansible-vault create vault.yml**

系统将提示您输入新 Ansible vault 的密码。不要丢失 vault 密码，因为每次需要访问 vault 文件时都需要它，包括在第二天操作和执行备份过程中。您可以通过将 vault 密码存储在加密的密码管理器中或根据您的组织策略安全地存储密码来保护 vault 密码。

在密码库中添加敏感变量，例如：

```
admin_password: <secure_controller_password>
pg_password: <secure_db_password>
automationhub_admin_password: <secure_hub_password>
automationhub_pg_password: <secure_hub_db_password>
automationhub_ldap_bind_password: <ldap_bind_password>
automationedacontroller_admin_password: <secure_eda_password>
automationedacontroller_pg_password: <secure_eda_db_password>
```

确保安装清单文件中不存在这些变量。要将新的 Ansible 库与安装程序搭配使用，请使用命令 ./setup.sh -e @vault.ymlcategories-unmarshal--ask-vault-pass 来运行。

2.2.5. 自动化控制器 STIG 注意事项

对于使用 Defense Information Systems Agency (DISA)安全技术实施指南(STIG)作为其整体安全策略的一部分，现在提供了 用于 Ansible Automation Platform 自动化控制器的 STIG。STIG 目前仅涵盖 Ansible Automation Platform 的自动化控制器组件。将 STIG 应用到自动化控制器时，需要注意很多注意事项。

自动化控制器 STIG 概述文档指出它要与 Red Hat Enterprise Linux 8 的 STIG 一起使用。此版本的自动化控制器 STIG 在 Red Hat Enterprise Linux 9 可用前发布，因此在应用自动化控制器 STIG 时，Red Hat Enterprise Linux 8 应该用作底层主机操作系统。某些 Red Hat Enterprise Linux 8 STIG 控制将与 Ansible Automation Platform 安装和操作冲突，如以下部分所述。

2.2.5.1. fapolicyd

Red Hat Enterprise Linux 8 STIG 要求 fapolicyd 守护进程正在运行。但是，当 fapolicyd enforcing 策略时不支持 Ansible Automation Platform，因为这会导致 Ansible Automation Platform 的安装和操作失败。因此，安装程序运行一个 pre-flight 检查，如果发现 fapolicyd 是 enforcing 策略，它将停止安装。本指南推荐按照以下流程在自动化控制器中将 fapolicyd 设置为 permissive 模式：

1. 编辑文件 /etc/fapolicyd/fapolicyd.conf 并设置 "permissive = 1"。
2. 使用命令 sudo systemctl restart fapolicyd.service 重新启动该服务。

在常规审核 STIG 控制的环境中，讨论通过安全审核员实现与 fapolicy 相关的 STIG 控制。

注意

如果 Red Hat Enterprise Linux 8 STIG 也应用于安装主机，默认的 fapolicyd 配置会导致 Ansible Automation Platform 安装程序失败。在这种情况下，建议在安装主机上将 fapolicyd 设置为 permissive 模式。

2.2.5.2. 使用"noexec"挂载的文件系统

Red Hat Enterprise Linux 8 STIG 要求使用 noexec 选项挂载了大量文件系统，以防止执行这些文件系统中的二进制文件。Ansible Automation Platform 安装程序运行 preflight 检查，如果以下文件系统使用 noexec 选项挂载，该检查将失败：

- /tmp
- /var
- /var/tmp

要安装 Ansible Automation Platform，您必须重新安装这些文件系统，并删除了 noexec 选项。安装完成后，执行以下步骤：

1. 重新应用 /tmp 和 /var/tmp 文件系统的 noexec 选项。
2. 将自动化控制器作业执行路径从 /tmp 更改为没有启用 noexec 选项的替代目录。
3. 要进行此更改，请以管理员身份登录自动化控制器 UI，进入到 Settings 并选择 Jobs settings。
4. 将 "Job execution path" 设置改为备用目录。

在正常操作过程中，包含 /var/lib/awx 子目录（通常为 /var）的文件系统不能使用 noexec 选项挂载，或者自动化控制器无法在执行环境中运行自动化作业。

在常规审核 STIG 控制的环境中，通过安全审核员讨论与文件系统相关的 STIG 控制。

2.2.5.3. 用户命名空间

Red Hat Enterprise Linux 8 STIG 要求内核设置 user.max_user_namespaces 设置为 "0"，但只有 Linux 容器没有使用。因为 Ansible Automation Platform 使用容器作为其执行环境功能的一部分，所以这个 STIG 控制不适用于自动化控制器。

要检查 user.max_user_namespaces 内核设置，请完成以下步骤：

1. 在命令行中登录到您的自动化控制器。
2. 运行 sudo sysctl user.max_user_namespaces 命令。
3. 如果输出显示该值为零，请查看文件 /etc/sysctl.conf 和 /etc/sysctl.d/ 下的所有文件，编辑包含 user.max_user_namespaces 设置的文件，并将值设为 "65535"。
4. 要应用这个新值，请运行 sudo sysctl -p <file> 命令，其中 & lt;file& gt; 是刚才修改的文件。
5. 重新运行 sudo sysctl user.max_user_namespaces 命令，并验证该值现在是否设置为 "65535"。

2.2.5.4. sudo 和 NOPASSWD

Red Hat Enterprise Linux 8 STIG 要求具有 sudo 权限的所有用户都必须提供密码（即，sudoers 文件中不能使用"NOPASSWD"指令）。Ansible Automation Platform 安装程序以特权用户身份运行许多任务，默认情况下，它预期可以在无需密码的情况下提升特权。要为安装程序提升权限提供安装程序密码，请在启动安装程序脚本时附加以下选项： ./setup.sh <setup options>categories-ProductShortName--ask-become-pass。

这也适用于为备份和恢复等第二天操作运行安装程序脚本。

2.3. 初始配置

授予系统某些部分的访问权限会带来安全漏洞。应用以下实践来帮助安全访问：

- 最小化对系统管理帐户的访问。用户界面(web 接口)和自动化控制器在其中运行的操作系统之间有一个区别。系统管理员或 root 用户可以访问、编辑和破坏任何系统应用程序。对控制器具有 root 访问权限的人都可以解密这些凭证，因此尽量减少对系统管理帐户的访问对于维护安全系统至关重要。

- 最小化本地系统访问。除了管理目的外，自动化控制器不应该需要本地用户访问。非管理员用户不应该具有控制器系统的访问权限。

- 强制隔离任务。不同的自动化组件可能需要在不同级别上访问系统。为每个组件使用不同的密钥或凭证，以便最小化任何一个密钥或凭证漏洞的影响。

- 将自动化控制器限制为只能进行低级控制器配置和灾难恢复的最低用户集合。在控制器上下文中，任何控制器 '系统管理员' 或 'superuser' 帐户都可以编辑、更改和更新控制器中的任何清单或自动化定义。

2.3.1. 使用基础架构作为代码模式

Red Hat Community of practice 创建了一组通过集合提供的自动化内容，以管理 Ansible Automation Platform 基础架构和配置作为代码。这可通过 Infrastructure as Code (IaC)或 Configuration as Code (CaC)实现平台本身的自动化。虽然这种方法的许多优点很明显，但需要考虑一些关键的安全隐患。

以下 Ansible 内容集合可用于使用基础架构作为代码方法管理 Ansible Automation Platform 组件，它们都在 Ansible Automation Hub 中找到：

表 2.5. Ansible 内容集合

验证的集合	集合目的
infra.aap_utilities	用于自动化 Ansible Automation Platform 第 1 天和第 2 天操作的 Ansible 内容，包括安装、备份和恢复、证书管理等。
infra.controller_configuration	管理自动化控制器组件的角色集合，包括管理用户和组(RBAC)、项目、作业模板和工作流、凭证等。

infra.ah_configuration	用于与自动化中心交互的 Ansible 内容，包括用户和组 (RBAC)、集合上传和管理、集合批准、管理执行环境镜像 registry 等。
infra.ee_utilities	用于创建和管理执行环境镜像的角色集合，或者从旧的 Tower virtualenv 迁移到执行环境。

许多企业使用 CI/CD 平台来配置管道或其他方法来管理这类基础架构。但是，原生使用 Ansible Automation Platform，可以将 Webhook 配置为原生链接基于 Git 的存储库。这样，Ansible 可以直接响应 git 事件来触发作业模板。这从整个过程中删除了对外部 CI 组件的需要，从而减少了攻击面。

这些实践允许对所有基础架构和配置进行版本控制。应用 Git 最佳实践以确保在同步到 Ansible Automation Platform 前正确检查代码质量。相关的 Git 最佳实践包括：

- 创建拉取请求。
- 确保检查工具就位。
- 确保没有提交纯文本 secret。
- 确保遵循 pre-commit hook 和任何其他策略。

IaC 还鼓励使用外部 vault 系统，无需将任何敏感数据存储在存储库中，或者根据需要处理单独的 vault 文件。有关使用外部 vault 系统的更多信息，请参阅本指南中的 2.3.2.3 外部凭证库注意事项。

2.3.2. 控制器配置

2.3.2.1. 配置集中式日志记录

日志记录的一个关键功能是自动化控制器检测和采取措施来缓解故障，如达到存储容量，默认关闭控制器。本指南推荐应用服务器成为高可用性系统的一部分。在这种情况下，自动化控制器将执行以下步骤来缓解失败：

- 如果失败是由缺少日志记录存储容量导致的，则应用程序必须继续生成日志记录（如果需要，自动重启日志服务），以优先方式覆盖最旧的日志记录。

- 如果日志记录发送到集中式集合服务器，且与此服务器的通信丢失或服务器失败，则应用程序必须在本地队列记录，直到通信被恢复或直到手动检索日志记录为止。在恢复与集中式集合服务器的连接后，必须采取措施将本地日志数据与集合服务器同步。

要验证每个自动化控制器主机的 rsyslog 配置，请为每个自动化控制器完成以下步骤：

管理员必须检查每个自动化控制器主机的 rsyslog 配置，以针对组织定义的日志捕获大小验证日志滚动。要做到这一点，请使用以下步骤，并正确使用配置步骤：

1. 检查自动化控制器配置中的 LOG_AGGREGATOR_MAX_DISK_USAGE_GB 字段。在主机上执行：

```
awx-manage print_settings LOG_AGGREGATOR_MAX_DISK_USAGE_GB
```

如果此字段没有设置为组织定义的日志捕获大小，请按照配置步骤操作。

2. 检查自动化控制器配置中的 LOG_AGGREGATOR_MAX_DISK_USAGE_PATH 字段到 /var/lib/awx。在主机上执行：

```
awx-manage print_settings LOG_AGGREGATOR_MAX_DISK_USAGE_PATH
```

如果此字段没有设置为 /var/lib/awx，请按照以下步骤执行：

a. 打开 Web 浏览器，再进入到 https://<automation controller server>/api/v2/settings/logging/，其中 <automation controller server> 是自动化控制器的完全限定主机名。如果显示 Log In 选项，点它，以自动化控制器 adminstrator 帐户身份登录，然后继续。

b. 在 Content 部分中，修改以下值，然后点 PUT ：

- LOG_AGGREGATOR_MAX_DISK_USAGE_GB = <new log buffer in GB>
- LOG_AGGREGATOR_MAX_DISK_USAGE_PATH = /var/lib/awx

请注意，需要在负载均衡的场景中在每个自动化控制器上进行这个更改。

必须记录所有用户会话数据，以支持故障排除、调试和分析，以进行可见性和分析。如果没有来自控制器 Web 服务器的这些数据，则事件调查的重要审计和分析将会丢失。要验证系统是否已配置为确保记录用户会话数据，请使用以下步骤：

对于每个自动化控制器主机，进入 console Settings >> System >> Miscellaneous System。

1. 点 Edit。
2. 设置以下内容：
   - 启用活动流 = On
   - 为清单同步启用活动流 = On
   - 机构管理员可以管理用户和团队 = Off
   - 机构管理员可见所有用户 = On
3. 点 Save

要将日志记录设置为任何聚合器类型，请按照以下步骤读取 支持的日志聚合器 文档并配置日志聚合器：

1. 进入 Ansible Automation Platform。
2. 单击 Settings。

3. 在 System 选项列表中，选择 Logging settings。

4. 在 Logging 设置屏幕底部，点 Edit。

5. 从提供的字段中设置可配置选项：

- 启用外部日志记录 ：如果要将日志发送到外部日志聚合器，请单击切换按钮到 ON。在完成此操作前，UI 需要填写日志记录聚合器和日志记录聚合器端口字段。
- 日志记录聚合器 ：输入您要发送日志的主机名或 IP 地址。
- 日志记录聚合器端口：如果聚合器需要端口，请为其指定端口。
- 日志记录聚合器类型 ：从下拉菜单中选择聚合器服务：
  - Splunk
  - Loggly
  - Sumologic
  - 弹性堆栈（以前称为 ELK 堆栈）
- 日志记录聚合器用户名 ：如果需要，输入日志记录聚合器的用户名。
- 日志记录聚合器密码/令牌 ：如果需要，请输入日志记录聚合器的密码。
- 单独记录系统跟踪事实：点工具提示图标，单击工具提示图标，是否要将其打开，或者默认保持关闭。

- 日志记录聚合器协议 ：选择一个连接类型（协议）来与日志聚合器通信。后续选项因所选协议而异。

- Logging Aggregator Level Threshold ：选择希望日志处理器报告的严重性级别。

- TCP Connection Timeout：指定连接超时（以秒为单位）。这个选项仅适用于 HTTPS 和 TCP 日志聚合器协议。

- 启用/禁用 HTTPS 证书验证： HTTPS 日志协议默认启用证书验证。如果您不希望日志处理程序在建立连接前验证外部日志聚合器发送的 HTTPS 证书，点切换按钮变为 OFF。

- 将数据发送到日志聚合器表单的日志记录器：默认情况下，所有四种类型数据都会预先填充。点字段旁的工具提示图标可查看每种数据类型的附加信息。删除您不想的数据类型。

- 日志格式 for API 4XX 错误：配置特定的错误消息。

6. 点 Save 应用设置，或点 Cancel 来取消更改。

7. 要验证您的配置是否已正确设置，请首先 保存，然后单击 Test。这会使用自动化控制器中的当前日志记录配置向日志聚合器发送测试日志消息。您应该检查以确保外部日志聚合器收到了此测试信息。

为使用 LDAP 用户名和密码登录的任何用户自动创建自动化控制器帐户。这些用户可以作为常规用户或机构管理员自动放入机构中。这意味着，当使用 LDAP 集成时，应打开日志记录。您可以像为 LDAP 启用日志记录一样，为 SAML 适配器启用日志信息。

以下步骤启用 LDAP 日志记录：

要为 LDAP 启用日志记录，您必须在 Settings 配置窗口中将 level 设置为 DEBUG。

1. 点击左侧导航栏中的 Settings，然后从 System 选项列表中选择 Logging settings。

2. 点 Edit。

3. 将 Logging Aggregator Level Threshold 字段设置为 Debug。

4. 点 Save 保存您的更改。

2.3.2.2. 配置外部身份验证源

如 用户身份验证计划 一节中所述，建议使用外部身份验证来访问自动化控制器。选择最适合您的需要的身份验证类型后，进入 Settings →] 并选择自动化控制器 UI 中的 Authentication → ，点身份验证后端的相关链接 → 并按照链接：https://docs.ansible.com/automation-controller/latest/html/administration/configure_tower_in_tower.html#authentication[configuring 身份验证连接的相关说明进行操作。

当在自动化控制器中使用 LDAP 进行外部身份验证时，进入 Settings 并选择 Authentication，然后在自动化控制器中选择 LDAP 设置，并确保配置了以下之一：

- 对于 LDAP over SSL，LDAP Server URI 设置必须以 ldaps:// 开头并使用端口 636，如 ldaps://ldap-server.example.com:636。

- 对于 LDAP over TLS，LDAP Start TLS 设置必须设置为 "On"。

2.3.2.3. 外部凭证库注意事项

Secret 管理是维护安全自动化平台的重要组件。我们推荐以下 secret 管理实践：

- 确保没有未授权用户对系统的访问权限，并确保只授予需要访问权限的用户。自动化控制器加密敏感信息，如密码和 API 令牌，但也存储要解密的密钥。授权用户可能有权访问所有内容。

- 使用外部系统来管理机密。如果需要更新凭证，外部系统就可以检索更新的凭证，其复杂性比内部系统要低。用于管理 secret 的外部系统包括 CyberArk、HashiCorp Vault、Microsoft Azure Key Management 等。如需更多信息，请参阅自动化控制器用户指南 v4.4 中的 Secret 管理系统 部分。

2.4. 第二天操作

第 2 天操作包括 Cluster Health 和 Scaling Checks，包括主机、项目和环境级别 Sustainment。您应该持续分析配置和安全偏移。

2.4.1. RBAC 注意事项

作为管理员，您可以使用内置到自动化控制器的基于角色的访问控制(RBAC)来委派对服务器清单、机构等的访问权限。管理员也可以集中管理各种凭证，允许最终用户利用所需的 secret，而无需向最终用户公开该 secret。RBAC 控制允许您控制器帮助您提高安全性并简化管理。

RBAC 是向用户或团队授予角色的方法。RBAC 易于认为是角色(Role)，它精确定义了谁、更改或删除要为其设置特定功能的"对象"。

您应该熟悉自动化控制器的 RBAC 设计的一些主要概念，包括角色、资源和用户。用户可以是角色的成员，授予他们对与该角色关联的任何资源或与"子代"角色关联的任何资源的某些访问权限。

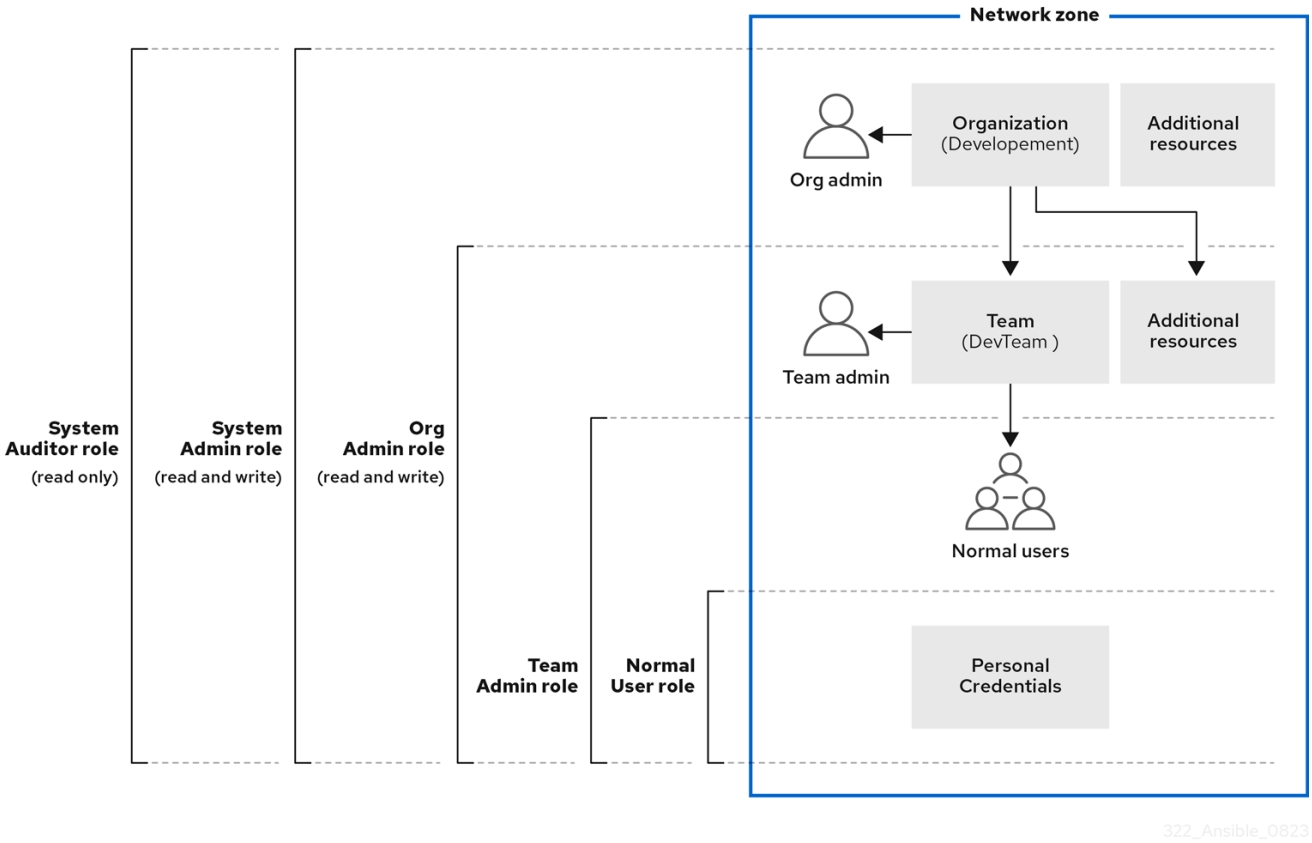
角色基本上是一个能力集合。通过为其分配的角色或通过通过角色层次结构继承的角色，授予用户对这些功能和控制器资源的访问权限。

角色将一组能力与一组用户相关联。所有功能都源自角色内的成员资格。用户仅通过为其分配的角色或通过角色层次结构继承的角色获得权限。角色的所有成员都具有授予该角色的所有权限。在一个机构中，角色相对稳定，而用户和能力有很多且可能会快速变化。用户可以有许多角色。

有关角色层次结构、访问权限继承、角色、权限、用户角色、角色创建等的更多详情，请参阅基于角色的访问控制。

以下是具有角色和资源权限的机构示例：

图 2.3. 自动化控制器中的 RBAC 角色范围



用户访问权限基于管理系统对象（用户、组、命名空间）的权限，而不是单独为特定用户分配权限。您可以为您创建的组分配权限。然后，您可以为这些组分配用户。这意味着，组中的每个用户都有分配给该组的权限。

在 Automation Hub 中创建的组，其范围可以包括系统管理员所需进行的任务，包括负责管理内部集合、配置用户访问和存储库管理。这些组可具有组织并上传内部开发的内容至 Automation Hub 的权限。如需更多信息，请参阅 Automation hub 权限 以实现一致性。

可以启用只查看访问，以进一步锁定私有自动化中心。通过启用只查看访问权限，您可以授予用户查看私有自动化 hub 上的集合或命名空间的访问权限，而无需进行登录。View-only 访问权限允许您与未授权用户共享内容，同时限制他们只能查看或下载源代码的权限，而无需编辑私有自动化中心上的任何内容。编辑 Red Hat Ansible Automation Platform 安装程序中的清单文件，为您的私有自动化中心启用只查看访问。

2.4.2. 更新和升级

所有升级的系统版本不能比当前要升级到的版本低两个主要版本。例如，要升级到自动化控制器 4.3，您必须首先在版本 4.1.x 上，因为没有从 3.8.x 或更早版本直接升级路径。如需更多信息，请参阅 升级到 Ansible Automation Platform。要运行自动化控制器 4.3，还必须有 Ansible 2.12 或更高版本。

2.4.2.1. 自动化控制器 STIG 注意事项

自动化控制器必须在您的机构策略指定的时间周期内安装与安全相关的软件更新，以及维护系统及其普通资产的完整性和保密性所需的安全配置集。

每天发现软件应用程序的安全漏洞。红帽不断更新和补丁自动化控制器，以解决新发现的安全漏洞。需要组织（包括组织的任何承包商）来快速安装与安全相关的软件更新（例如，补丁、服务包和热修复）。在安全评估、持续监控、事件响应活动或信息系统错误处理过程中发现的漏洞还必须非常迅速地解决。

作为每个自动化控制器主机的系统管理员，执行以下操作：

1. 检查 DNF Automatic 计时器的状态：

   ```
   systemctl status dnf-automatic.timer
   ```

2. 如果输出中没有包括 Active: active，则这是一个发现。

3. 检查 DNF Automatic 的配置：

   ```
   grep apply_updates /etc/dnf/automatic.conf
   ```

4. 如果没有显示 apply_updates = yes，则会出现一个查找。

5. 安装并启用 DNF Automatic：

   ```
   dnf install dnf-automatic （运行安装）systemctl enable --now dnf-automatic.timer
   ```

6. 修改 /etc/dnf/automatic.conf 并设置 apply_updates = yes。

在成为生产 Web 服务器的一部分之前，所有自动化控制器 nginx 前端 Web 服务器文件都必须验证其完整性（如校验和和哈希）。验证添加到 web 服务器的补丁、升级、证书等与文件的生成者没有变化，对于文件验证和信息不涉及是至关重要的。自动化控制器 nginx web 服务器主机必须具有机制来验证文件在安装前是否有效。

作为系统管理员，对于每个自动化控制器 nginx web 服务器主机，请执行以下操作：

1. 验证自动化控制器 nginx web 服务器托管文件的完整性：

   aide --check

2. 验证根据之前保留的高级入侵检测环境(AIDE)数据库校验和的校验和。

3. 如果对之前校验和有任何未授权或未解释的更改，则这是发现的。

作为系统管理员，对于每个自动化控制器 nginx web 服务器主机，请执行以下操作：

1. 检查现有或安装 AIDE：

   yum install -y aide

2. 在每个自动化控制器 nginx web 服务器主机初始安装后，立即创建或更新 AIDE 数据库：

   aide --init && mv /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz

3. 通过更新 AIDE 数据库接受对主机的任何预期更改：

   aide --update

4. 输出将为 AIDE 数据库提供校验和。保存在受保护的位置。

所有自动化控制器 nginx web 服务器帐户没有被安装的功能（如工具、实用程序、特定服务等）没有被创建，且必须在卸载 web 服务器功能时删除。如果没有使用 Web 服务器帐户，必须在卸载 Web 服务器时删除它们。这是因为帐户随着时间的推移过时，且不会被认为是过时的。另外，如果不会使用帐户，则不能因为相同原因创建帐户。这两种情况都造成了 Web 服务器利用的机会。

当创建用于 Web 服务器功能的帐户时，如文档、示例代码、示例应用程序、教程、实用程序和服务，

即使没有安装该功能，也会成为一个 Web 服务器的易受威胁。这些帐户不再处于非活跃状态，且不会通过常规使用监控，帐户的密码不会被创建或更新。攻击者可以使用这些帐户获得 Web 服务器的访问权限，并开始调查提高帐户特权的方法。

不得创建用于所有自动化控制器 nginx web 服务器功能的帐户，且必须在卸载这些功能时删除。

作为每个自动化控制器 nginx web 服务器的系统管理员，请执行以下操作：

1. 检查 /etc/passwd 中的 nginx 用户。

2. 使用以下命令验证单个用户 nginx 是否存在：

   ```
   [ grep -c nginx /etc/passwd == 1 ] || echo FAILED
   ```

3. 如果显示 FAILED，则会出现一个查找。

作为每个自动化控制器 nginx web 服务器的系统管理员，请执行以下操作：

1. 如果 /etc/passwd中没有 nginx 用户，则重新安装自动化控制器

2. 查看 /etc/passwd 中枚举的所有用户，并删除任何不面向 Red Hat Enterprise Linux 或自动化控制器和/或组织禁止的任何用户。

自动化控制器 nginx web 服务器被配置为检查并从机构标识的时间段内从机构标识的时间段内安装安全相关的软件更新。默认情况下，这个时间段将每 24 小时。

作为每个自动化控制器 nginx web 服务器主机的系统管理员，请执行以下操作：

1. 验证系统是否已配置为从组织定义的源接收权威系统更新：

   ```
   yum -v repolist
   ```

2. 如果每个 URL 无效，且与组织定义的要求一致，则这是发现的。

3. 如果根据机构定义的要求没有启用每个软件仓库，则这是发现的。

4. 如果系统没有配置为至少从此源自动接收和应用系统更新，或者至少每 30 天手动接收并应用更新，这将会发现。

作为系统管理员，对于每个自动化控制器 nginx web 服务器主机，请执行以下操作：

1. 根据机构定义的要求配置更新存储库，或订阅红帽底层操作系统的更新存储库。

2. 从这些仓库执行更新：

```
$ yum update -y
```

3. 执行以下之一：

   a. 计划每 30 天或根据机构定义的策略进行一次更新：

```
$ yum install -y dnf-automatic && sed -i '/apply_updates/s/no/yes/'
/etc/dnf/automatic.conf && sed -i '/OnCalendar/s/^OnCalendar\s*=./OnCalendar=-14.10
6:00/lib/systemd/system/dnf-automatic.timer && enable --now-automatic.timer.timer.
```

   b. 计划至少每 30 天或根据组织定义的策略进行一次手动更新。

4. 重启自动化控制器 nginx web 服务器主机。

2.4.2.2. 灾难恢复和连续性操作

进行常规的 Ansible Automation Platform 备份是灾难恢复计划的关键部分。备份和恢复都是使用安装程序执行的，因此应从本文档前面所述的专用安装主机执行这些操作。有关如何执行这些操作的详情，请参阅自动化控制器文档中的 备份和恢复 部分。

备份的一个重要方面是它们包含数据库的副本，以及用于解密存储在数据库中的凭据的机密密钥，因此备份文件应存储在安全加密位置。这意味着，对端点凭证的访问被正确保护。对备份的访问应仅限于具有对自动化控制器和专用安装主机的 root shell 访问权限的 Ansible Automation Platform 管理员。

Ansible Automation Platform 管理员需要备份其 Ansible Automation Platform 环境的主要原因包括：

- 要从 Ansible Automation Platform 环境中保存数据副本，以便在需要时恢复数据。
- 如果您要创建新的 Ansible Automation Platform 集群或准备升级，请使用备份将环境恢复到不同的服务器集合。

在所有情况下，推荐的和最安全的流程都是始终使用同一版本的 PostgreSQL 和 Ansible Automation Platform 来备份和恢复环境。

强烈建议在系统中使用一些冗余。如果 secrets 系统停机，则自动化控制器无法获取信息，并可能会导致失败，在恢复服务后可以恢复。如果您认为为您生成的 SECRET_KEY 自动化控制器已被破坏，并且必须重新生成，您可以从安装程序运行一个工具，它的行为与自动化控制器备份和恢复工具非常相似。

要生成新的 secret 密钥，请执行以下步骤：

1. 在做任何其他操作前，备份 Ansible Automation Platform 数据库！按照 备份和恢复 Controller 部分所述步骤操作。
2. 使用安装中的清单（与运行备份/恢复的清单相同），运行 setup.sh -k。

之前密钥的备份副本保存在 /etc/tower/ 中。