Red Hat

Upgrade

Ansible Automation Platform 2.6

May 12, 2026

Contents

1 Upgrade

Plan your upgrade to Ansible Automation Platform 2.6

Ansible Automation Platform 2.6 includes changes that improve the overall platform upgrade experience.

- **Upgrading from 2.5 to 2.6**
- **Upgrading from 2.4 to 2.6**

NOTE:
You must be on the latest version of 2.4 or 2.5 before you upgrade to 2.6.

Upgrading from 2.5 to 2.6

Upgrading from 2.5 to 2.6 does not involve changes to the platform infrastructure requirements, architecture, or services. The improvements described in the 2.4 to 2.6 upgrade path are also present in the 2.5 to 2.6 upgrade path; however, the platform gateway service is already in place in 2.5.

Additionally, note the following:

- If you upgraded from 2.4 to 2.5, you must migrate your authentication methods and users before upgrading to 2.6 as that legacy authenticator functionality was removed.
- When you upgrade to 2.6, the system removes any users that the 2.4 to 2.5 upgrade did not fully migrate. The users that have previously merged their user records while on 2.5 will remain to function as is for 2.6.
- Upgrading to 2.6 prevents 2.4 automation controller users who never successfully logged into 2.5 from logging into the platform-gateway. These users retain backwards compatibility for direct Automation Execution access but cannot access the full platform. Ensure all users planning to use 2.6 have successfully logged into 2.5 before upgrading.
- Unified RBAC management across Ansible Automation Platform components: All Ansible Automation Platform collections, which support the Configuration-as-Code (CaC) approach, now use a standard global environment variable name and module variable name across Ansible Automation Platform components. For more details, see the **Release notes** for whats new around RBAC in 2.6, what's changed around RBAC for users moving from 2.5 to 2.6

For more information about upgrading, see the upgrade document for your deployment type:

- Containerized
- RPM
- OpenShift Container Platform

> **NOTE:**
> Upgrades from the latest 2.5 version to 2.6 are supported with all deployment types: RPM, containerized, and OpenShift Container Platform deployments.

Upgrading from 2.4 to 2.6

Note the following when upgrading from 2.4 to 2.6:

- **Upgrades from 2.4**: Ansible Automation Platform supports upgrading directly from the latest 2.4 version to 2.6. Directly upgrading to 2.6 is the recommended upgrade path from 2.4, as several improvements in 2.6 simplify and improve the upgrade experience.

  > **NOTE:**
  > You can upgrade directly from the latest 2.4 version to 2.6 with RPM and OpenShift Container Platform deployments. However, upgrading Event-Driven Ansible 2.4 or from the 2.4 containerized deployment is not supported, as both features were Tech Preview in 2.4.

  For more information, see the upgrade document for your deployment type. Either RPM, or OpenShift Container Platform.

- **Infrastructure changes**: Ansible Automation Platform RPM deployments require additional infrastructure compared with 2.4, due to the addition of the platform gateway service. Infrastructure needs vary depending on factors such as whether you implement a growth or an enterprise deployment.

- **Authentication changes**: Enterprise authentication configuration and mappings (for example, SAML, LDAP, OIDC) move from automation controller 2.4 to platform gateway 2.6 as part of the upgrade process. You do not need to manually reconfigure these authentication methods after you upgrade.
  See **Access management and authentication** for information about authentication options in general.

  > **NOTE:**
  > Authentication upgrade improvements apply to RPM and OpenShift Container Platform deployments. Upgrades from the 2.4 containerized deployment Tech Preview release are not supported. Additionally, upgrading Event-Driven Ansible 2.4 is not supported.

- **Identify access management changes**: All automation controller Identity Access Management (IAM) data moves from automation controller 2.4 to the platform gateway in 2.6 as part of the upgrade process. With automation controller 2.4 as the default source of IAM data for the platform gateway in 2.6, users retain their memberships and are assigned appropriate platform-level roles in 2.6.
  As part of the upgrade process:

- Users, teams, organizations, their memberships, and common roles in 2.4 move from automation controller 2.4 to the platform gateway in 2.6.
- Administrators in automation controller 2.4 become platform gateway administrators in 2.6.
- Controller admins in 2.4 become platform gateway admins in 2.6.
  The more organizations, teams, and users being migrated during an upgrade, the longer the upgrade takes. As an example, upgrading and migrating 4,000 users, 400 teams, and 40 organizations can take close to two hours.

> **NOTE:**
> Identity access management changes apply to RPM and OpenShift Container Platform deployments. Upgrades from the 2.4 containerized deployment Tech Preview release are not supported.

See **Identity access management data movement** for more information.

- **API changes**: Some APIs are being deprecated in 2.6. See **API changes in Ansible Automation Platform 2.6** for more information.
- **Unified RBAC management across Ansible Automation Platform components**: All Ansible Automation Platform collections, which support the Configuration-as-Code (CaC) approach, now use a standard global environment variable name and module variable name across Ansible Automation Platform components. For more details, see the **Release notes** for what's new around RBAC in 2.6, or what's changed around RBAC for users moving from 2.5 to 2.6

Related information
Configure central authentication for Ansible Automation Platform
Manage access with role-based access control
ansible.platform
API changes in 2.6
Infrastructure changes for container-based deployments
Identity access management changes when upgrading to 2.6
Infrastructure changes for Operator-based deployments
Infrastructure changes for container-based deployments
Infrastructure changes for RPM deployments
Supported Operator-based upgrade and migration scenarios
Release notes
Supported RPM-based upgrade and migration scenarios

FAQs on upgrading to 2.6

Find concise answers to frequently asked questions about upgrading your system to quickly troubleshoot common issues and plan your migration effectively.

What are the supported installation topologies and operating systems for Ansible Automation Platform 2.6?

Red Hat has adopted a more definitive approach to installation topologies, categorizing them as "growth" or "enterprise" for production-ready setups:

- Growth topology: is intended for organizations that are getting started with Ansible Automation Platform and do not require redundancy or higher compute for large volumes of automation.
- Enterprise topology: is intended for organizations that require Ansible Automation Platform to be deployed with redundancy or higher compute for large volumes of automation.

For more information, see **Tested deployment models**.

Why did my authentication settings that were present in automation controller 2.4 not get imported?

If an authentication method is missing in Ansible Automation Platform 2.6, review the setup_log for migration warnings. These warnings indicate the reason authentication settings were not successfully created during the upgrade to version 2.6.

Why did my upgrade from Ansible Automation Platform 2.4 to 2.6 fail or encounter an error during the SAML authenticator migration?

A SAML authenticator migration fails if the configuration has an encrypted private key. While automation controller in Ansible Automation Platform 2.4 allowed input of an encrypted private key without error, Ansible Automation Platform 2.6 does not support this feature for authenticators and prevents the migration of any SAML configuration containing one.

To prevent migration failure and service disruption for SSO users, you must:

- **Before upgrading**: Replace the encrypted private key with an unencrypted one in the SAML authenticator settings on your Ansible Automation Platform 2.4 environment.
- **If the upgrade already failed**: Platform gateway did not migrate the authenticator. A local administrator must manually re-create the SAML authenticator in the new Ansible Automation Platform 2.6 environment to restore SSO functionality.

Which Red Hat Enterprise Linux versions are supported for RPM and containerized installations?

RPM installations will continue to be supported exclusively on Red Hat Enterprise Linux 9. Containerized installations support both Red Hat Enterprise Linux 9.4 or later versions of Red Hat Enterprise Linux 9 and Red Hat Enterprise Linux 10 or later versions of Red Hat Enterprise Linux 10 for enterprise topologies.

What is the difference between "upgrade" and "migration" in Ansible Automation Platform 2.6?

An upgrade is an application action, such as updating Ansible Automation Platform 2.5 to 2.6. A migration involves moving data, such as from an RPM-based 2.6 installation to a container-based 2.6 installation. New service components are not explicitly required between versions 2.5 and 2.6.

Can I upgrade from 2.4 to 2.6 or must I upgrade to 2.5 first?

Yes you can upgrade directly to 2.6. However note that there might be new system requirements you must update before upgrading.

How will managed cloud customers be upgraded to Ansible Automation Platform 2.6?

All managed cloud customers on Microsoft Azure and Amazon Web Services will be upgraded to 2.6. Two upgrade window options will be available following the platform upgrade: non-production or production environments. Communications will be ongoing from mid-July until late September.

Will migrations be fully supported in Ansible Automation Platform 2.6?

Yes, migrations are fully supported, enabling customers to move from RPM installations to containerized or OpenShift Container Platform environments, or to the managed offering. Customers must be on the latest version of Ansible Automation Platform for their current installation before migrating. The installation program manages new components introduced in version 2.5 for direct 2.4 to 2.6 upgrades.

When upgrading from 2.4 to 2.6 (applies only to RPM or OpenShift Container Platform), what is different about the upgrade process compared with the 2.4 to 2.5 process?

See the **Overview of upgrade improvements** section.

I'm using Event-Driven Ansible in 2.4, can I upgrade Event-Driven Ansible to 2.6?

If you are using Ansible Automation Platform 2.4 with the technical preview of Event-Driven Ansible controller but want to upgrade to the Ansible Automation Platform 2.6 with Event-Driven Ansible, you must install a new instance of Ansible Automation Platform 2.6 and manually re-create your Event-Driven Ansible configurations in the new, fully integrated environment.

Will my existing 2.4 or 2.5 OAuth Applications/Tokens, Credentials/Customer Credentials, and Personal Access Tokens still work after upgrading to 2.6?

For upgrades from Ansible Automation Platform 2.4 or 2.5 to Ansible Automation Platform 2.6, some manual configuration is required:

- OAuth applications:
  - Automation controller: You can view and edit existing automation controller applications, but you cannot create new ones. They still function, but they might be removed in a future release. You should plan to migrate to platform OAuth applications.
  - Ansible Automation Platform: Platform OAuth applications provide an updated interface and are the standard for future use. You will move to these applications.
- Tokens:
  - Automation controller: Automation controller personal access tokens (PATs) are deprecated. You will move to platform gateway PATs.
  - Ansible Automation Platform: Platform tokens provide an updated interface and are the standard for future use. You will move to these tokens.
- Authenticator configurations:
  - Ansible Automation Platform 2.4 to Ansible Automation Platform 2.6: The migration of all authenticator configurations from the automation controller to the platform gateway is automated. This includes third-party authentication

configurations and sensitive data such as SAML private keys or OAuth secret keys. If you use custom LDAP certificates, you must manually migrate them.

- Ansible Automation Platform 2.5 to Ansible Automation Platform 2.6: Authenticator configurations are not automatically migrated. LDAP settings configured in Ansible Automation Platform 2.5 remain as they were after upgrading to 2.6.

What RBAC (platform gateway and/or automation controller) control permissions will be missing in version 2.6?

- **For 2.4 to 2.6 upgrades:** During the upgrade, authenticators and their mappings from the controller are imported into the gateway; therefore, you don't need to manually migrate authenticators.
- **For 2.5 to 2.6 upgrades:** Authenticators and their mappings in the platform gateway continue to function as is, because no changes are imported.

Which version of PostgreSQL does Ansible Automation Platform 2.6 support?

Ansible Automation Platform 2.6 supports PostgreSQL 15 for its managed databases and additionally supports PostgreSQL 15, 16, and 17 for external databases.

Related information
Tested deployment models
Plan your upgrade to Ansible Automation Platform 2.6

Supported upgrade and migration scenarios

Use these reference tables to find the supported upgrade paths for your Ansible Automation Platform deployment. Review Red Hat Enterprise Linux version compatibility and step-by-step processes for RPM, container, and OpenShift Container Platform deployment types.

NOTE:
In-place upgrades of major Red Hat Enterprise Linux versions are not supported. You must migrate your existing deployment of Ansible Automation Platform to a new Red Hat Enterprise Linux environment.

WARNING: To upgrade to Ansible Automation Platform 2.7, you must be running a containerized or OpenShift Container Platform deployment. RPM-based deployments are not supported as an upgrade path to 2.7. If you are running an RPM-based deployment, migrate to a containerized or OpenShift Container Platform deployment before you upgrade.

.

Supported RHEL versions by deployment type

Supported RHEL versions differ among deployment types, as shown in the following table.

Deployment type and version	Supported RHEL version
RPM 2.6	RHEL 9
Containerized 2.6	RHEL 9, RHEL 10
OpenShift Container Platform 2.6	For RHEL versions included with OpenShift Container Platform, see Red Hat OpenShift Container Platform Life Cycle Policy.

Supported container-based upgrade and migration scenarios

Find the supported upgrade paths for your container-based Ansible Automation Platform deployment. This helps you plan the necessary steps for a smooth upgrade.

Container-based Ansible Automation Platform 2.5 on RHEL 9

Source	Target	Process
Container-based Ansible Automation Platform 2.5 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 9	• Upgrade your container deployment from 2.5 to 2.6.
Container-based Ansible Automation Platform 2.5 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 10	1. Backup your container deployment of 2.5 on a RHEL 9 environment, then restore to a RHEL 10 environment

Source	Target	Process
		running a fresh container installation 2.5. 2. Upgrade your container deployment from 2.5 to 2.6.
Container-based Ansible Automation Platform 2.5 on RHEL 9	Ansible Automation Platform on OpenShift Container Platform 2.6	1. Upgrade your container deployment from 2.5 to 2.6. 2. Migrate your container deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

Container-based Ansible Automation Platform 2.5 on RHEL 10

Source	Target	Process
Container-based Ansible Automation Platform 2.5 on RHEL 10	Container-based Ansible Automation Platform 2.6 on RHEL 10	• Upgrade your container deployment from 2.5 to 2.6.
Container-based Ansible Automation Platform 2.5 on RHEL 10	Ansible Automation Platform on OpenShift Container Platform 2.6	1. Upgrade your container deployment from 2.5 to 2.6. 2. Migrate your container deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

Container-based Ansible Automation Platform 2.6 on RHEL 9

Source	Target	Process
Container-based Ansible Automation Platform 2.6 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 10	• Backup your deployment of container 2.6 on RHEL 9, then restore to a RHEL 10 environment running a fresh container installation 2.6.
Container-based Ansible Automation Platform 2.6 on RHEL 9	Ansible Automation Platform on OpenShift Container Platform 2.6	• Migrate your container deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

Container-based Ansible Automation Platform 2.6 on RHEL 10

Source	Target	Process
Container-based Ansible Automation Platform 2.6 on RHEL 10	Ansible Automation Platform on OpenShift Container Platform 2.6	• Migrate your container deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

Supported Operator-based upgrade and migration scenarios

Find the supported upgrade paths for Ansible Automation Platform deployments that use OpenShift Container Platform. This helps you plan the necessary steps for a smooth upgrade.

Ansible Automation Platform on OpenShift Container Platform 2.4

Source	Target	Process
Ansible Automation Platform on OpenShift Container Platform 2.4	Ansible Automation Platform on OpenShift Container Platform 2.6	• Upgrading the Ansible Automation Platform on OpenShift Container Platform 2.4 to 2.6.

Ansible Automation Platform on OpenShift Container Platform 2.5

Source	Target	Process
Ansible Automation Platform on OpenShift Container Platform 2.5	Ansible Automation Platform on OpenShift Container Platform 2.6	• Upgrading the Ansible Automation Platform on OpenShift Container Platform 2.5 to 2.6.

Supported RPM-based upgrade and migration scenarios

Find the supported upgrade paths for your RPM-based Ansible Automation Platform deployment. This helps you plan the necessary steps for a smooth upgrade.

WARNING: Ansible Automation Platform 2.6 is the last version that supports RPM-based deployments. To upgrade to Ansible Automation Platform 2.7, migrate to a containerized or OpenShift Container Platform deployment first.

RPM-based Ansible Automation Platform 2.4 on RHEL 8

Source	Target	Process
RPM-based Ansible Automation Platform 2.4 on RHEL 8	RPM-based Ansible Automation Platform 2.6 on RHEL 9	1. Backup your deployment of 2.4 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.4. 2. Upgrade your RPM deployment from 2.4 to 2.6.
RPM-based Ansible Automation Platform 2.4 on RHEL 8	Container-based Ansible Automation Platform 2.6 on RHEL 9	1. Backup your deployment of 2.4 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.4. 2. Upgrade your RPM deployment from 2.4 to 2.6. 3. Migrate your RPM deployment 2.6 to a container deployment 2.6.
RPM-based Ansible Automation Platform 2.4 on RHEL 8	Container-based Ansible Automation Platform 2.6 on RHEL 10	1. Backup your deployment of 2.4 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.4. 2. Upgrade your RPM deployment from 2.4 to 2.6. 3. Migrate your RPM deployment 2.6 to a container deployment 2.6. 4. Backup your deployment of container 2.6 on RHEL 9, then restore to a RHEL 10 environment running a fresh container installation 2.6.
RPM-based Ansible Automation Platform 2.4 on RHEL 8	Ansible Automation Platform on OpenShift Container Platform 2.6	1. Backup your deployment of 2.4 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.4.

Source	Target	Process
		2. Upgrade your RPM deployment from 2.4 to 2.6. 3. Migrate your RPM deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

RPM-based Ansible Automation Platform 2.4 on RHEL 9

Source	Target	Process
RPM-based Ansible Automation Platform 2.4 on RHEL 9	RPM-based Ansible Automation Platform 2.6 on RHEL 9	1. Upgrade your RPM deployment from 2.4 to 2.6.
RPM-based Ansible Automation Platform 2.4 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 9	1. Upgrade your RPM deployment from 2.4 to 2.6. 2. Migrate your RPM deployment 2.6 to a container deployment 2.6.
RPM-based Ansible Automation Platform 2.4 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 10	1. Upgrade your RPM deployment from 2.4 to 2.6. 2. Migrate your RPM deployment 2.6 to a container deployment 2.6. 3. Backup your deployment of container 2.6 on RHEL 9, then restore to a RHEL 10 environment running a fresh container installation 2.6.
RPM-based Ansible Automation Platform 2.4 on RHEL 9	Ansible Automation Platform on OpenShift Container Platform 2.6	1. Upgrade your RPM deployment from 2.4 to 2.6. 2. Migrate your RPM deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

RPM-based Ansible Automation Platform 2.5 on RHEL 8

Source	Target	Process
RPM-based Ansible Automation Platform 2.5 on RHEL 8	RPM-based Ansible Automation Platform 2.6 on RHEL 9	1. Backup your deployment of 2.5 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.5. 2. Upgrade your RPM deployment from 2.5 to 2.6.
RPM-based Ansible Automation Platform 2.5 on RHEL 8	Container-based Ansible Automation Platform 2.6 on RHEL 9	1. Backup your deployment of 2.5 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.5. 2. Upgrade your RPM deployment from 2.5 to 2.6. 3. Migrate your RPM deployment 2.6 to a container deployment 2.6.
RPM-based Ansible Automation Platform 2.5 on RHEL 8	Container-based Ansible Automation Platform 2.6 on RHEL 10	1. Backup your deployment of 2.5 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.5. 2. Upgrade your RPM deployment from 2.5 to 2.6. 3. Migrate your RPM deployment 2.6 to a container deployment 2.6. 4. Backup your deployment of container 2.6 on RHEL 9, then restore to a RHEL 10 environment running a fresh container installation 2.6.
RPM-based Ansible Automation Platform 2.5 on RHEL 8	Ansible Automation Platform on OpenShift Container Platform 2.6	1. Backup your deployment of 2.5 RPM on RHEL 8, then restore to a RHEL 9 environment running a fresh installation of RPM 2.5.

Source	Target	Process
		2. Upgrade your RPM deployment from 2.5 to 2.6. 3. Migrate your RPM deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

RPM-based Ansible Automation Platform 2.5 on RHEL 9

Source	Target	Process
RPM-based Ansible Automation Platform 2.5 on RHEL 9	RPM-based Ansible Automation Platform 2.6 on RHEL 9	• Upgrade your RPM deployment from 2.5 to 2.6.
RPM-based Ansible Automation Platform 2.5 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 9	1. Upgrade your RPM deployment from 2.5 to 2.6. 2. Migrate your RPM deployment 2.6 to a container deployment 2.6.
RPM-based Ansible Automation Platform 2.5 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 10	1. Upgrade your RPM deployment from 2.5 to 2.6. 2. Migrate your RPM deployment 2.6 to a container deployment 2.6. 3. Backup your deployment of container 2.6 on RHEL 9, then restore to a RHEL 10 environment running a fresh container installation 2.6.
RPM-based Ansible Automation Platform 2.5 on RHEL 9	Ansible Automation Platform on OpenShift Container Platform 2.6	1. Upgrade your RPM deployment from 2.5 to 2.6. 2. Migrate your RPM deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

RPM-based Ansible Automation Platform 2.6 on RHEL 9

Source	Target	Process
RPM-based Ansible Automation Platform 2.6 on RHEL 9	Container-based Ansible Automation Platform 2.6 on RHEL 10	1. Migrate your RPM deployment 2.6 to a container deployment 2.6. 2. Backup your deployment of container 2.6 on RHEL 9, then restore to a RHEL 10 environment running a fresh container installation 2.6.
RPM-based Ansible Automation Platform 2.6 on RHEL 9	Ansible Automation Platform on OpenShift Container Platform 2.6	• Migrate your RPM deployment 2.6 to Ansible Automation Platform on OpenShift Container Platform 2.6.

Infrastructure changes for container-based deployments

Container-based deployments require specific infrastructure changes during upgrade.

NOTE:
Upgrades from containerized deployments that were Technology Preview are not supported. You must perform a fresh installation.

Upgrade a 2.5 growth topology to a 2.6 growth topology

You can upgrade your 2.5 container-based growth topology to a 2.6 container-based growth topology. The topologies are the same between Ansible Automation Platform 2.5 and 2.6.

For more information about the growth topology infrastructure requirements and configuration details, see the **Container growth topology** section of *Tested deployment models*.

Related information
Operator growth topology

Upgrade an enterprise topology on OpenShift Container Platform

You can upgrade your enterprise topology to the latest version of Ansible Automation Platform on OpenShift Container Platform.

For more information about the enterprise topology infrastructure requirements and configuration details, see the **Operator enterprise topology** section of *Tested deployment models*.

Related information
Operator enterprise topology

Infrastructure changes for Operator-based deployments

Tested infrastructure changes are available for Operator-based deployments. To perform an upgrade, see **Upgrading Red Hat Ansible Automation Platform Operator on Red Hat OpenShift Container Platform**.

Related information
Upgrade Red Hat Ansible Automation Platform Operator on Red Hat OpenShift Container Platform

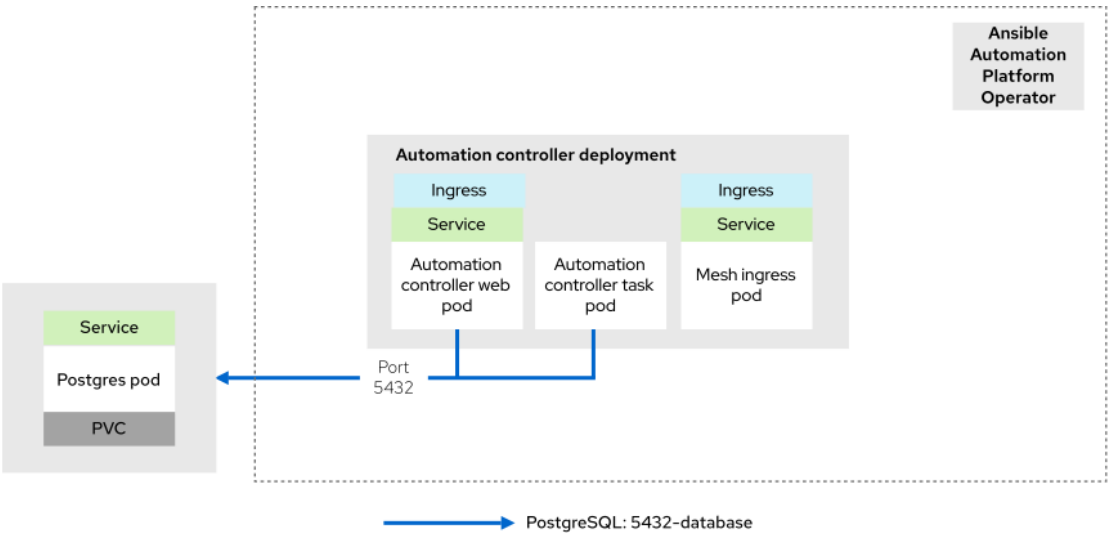
2.4 single automation controller node deployment to a 2.6 growth topology

Plan your upgrade from a 2.4 single automation controller node setup to a 2.6 growth topology. Review the required infrastructure changes and requirements for a successful upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

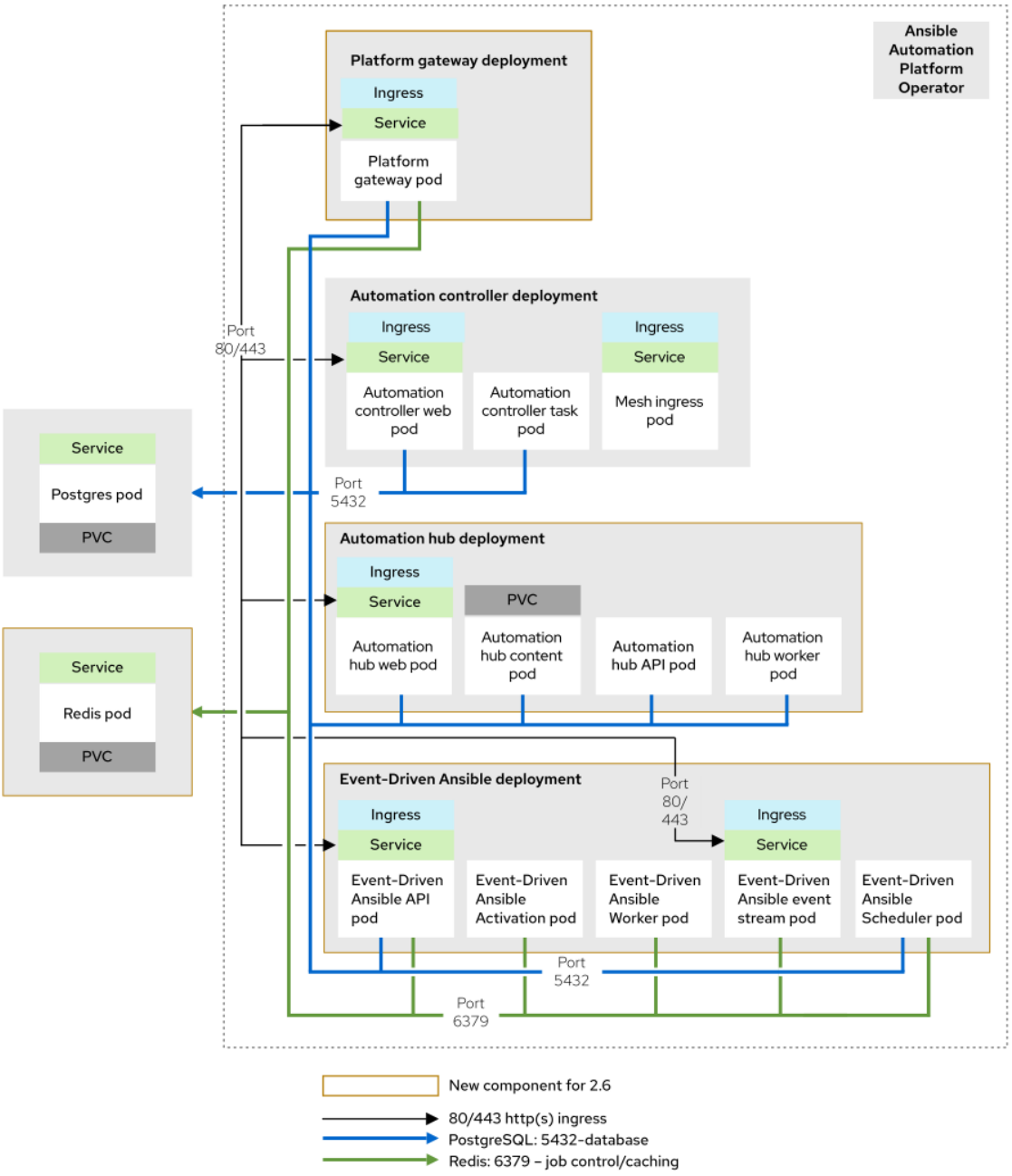
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a single automation controller node deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
Non-redundant automation controller-only deployment: • One automation controller web pod • One automation controller task pod • One database pod	Growth topology: • One automation controller web pod • One automation controller task pod • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • One automation hub Redis pod • One Event-Driven Ansible controller API pod • One Event-Driven Ansible controller activation worker pod • One Event-Driven Ansible controller default worker pod • One Event-Driven Ansible controller event stream pod • One Event-Driven Ansible controller scheduler pod • One platform gateway pod • One database pod • One Redis pod	See the **Operator growth topology** section of *Tested deployment models*.

Related information
Operator growth topology

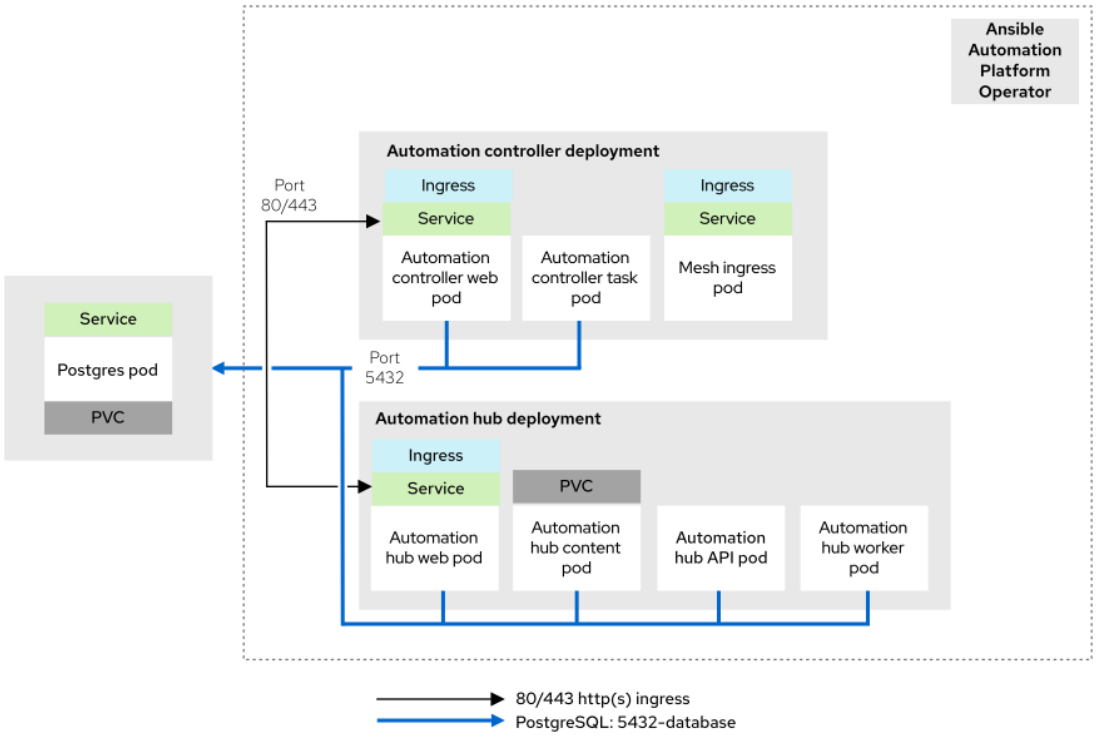
2.4 single automation controller and automation hub deployment to a 2.6 growth topology

Upgrade your 2.4 single-node deployment (automation controller and automation hub) to a 2.6 growth topology. Review the infrastructure changes and requirements needed to successfully plan your upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

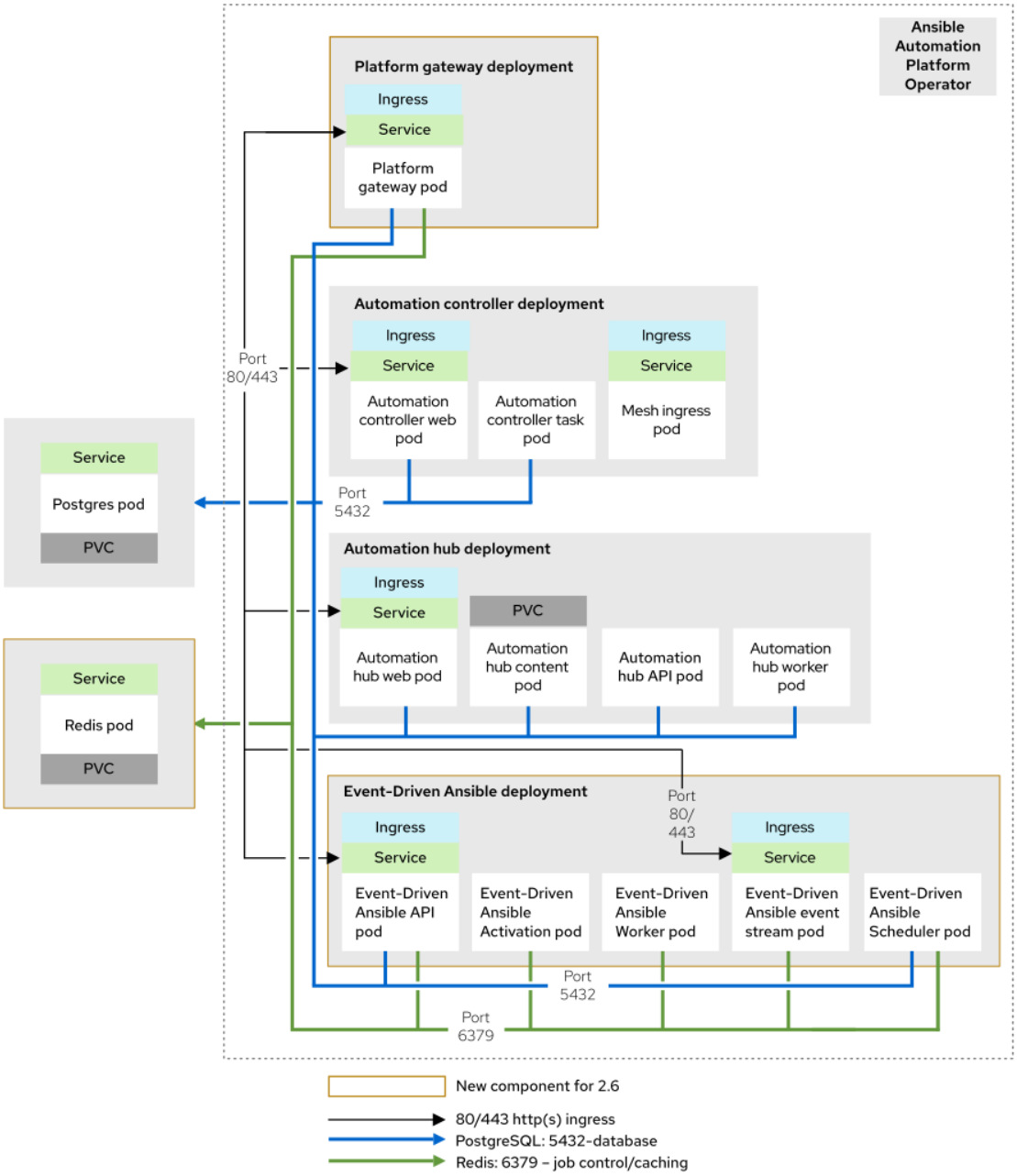
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a single automation controller and automation hub deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
Non-redundant automation controller and automation hub deployment: • One automation controller web pod • One automation controller task pod • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • One database pod	Growth topology: • One automation controller web pod • One automation controller task pod • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • One automation hub Redis pod • One Event-Driven Ansible controller API pod • One Event-Driven Ansible controller activation worker pod • One Event-Driven Ansible controller default worker pod • One Event-Driven Ansible controller event stream pod • One Event-Driven Ansible controller scheduler pod • One platform gateway pod • One database pod • One Redis pod	See the **Operator growth topology** section of *Tested deployment models*.

Related information
Operator growth topology

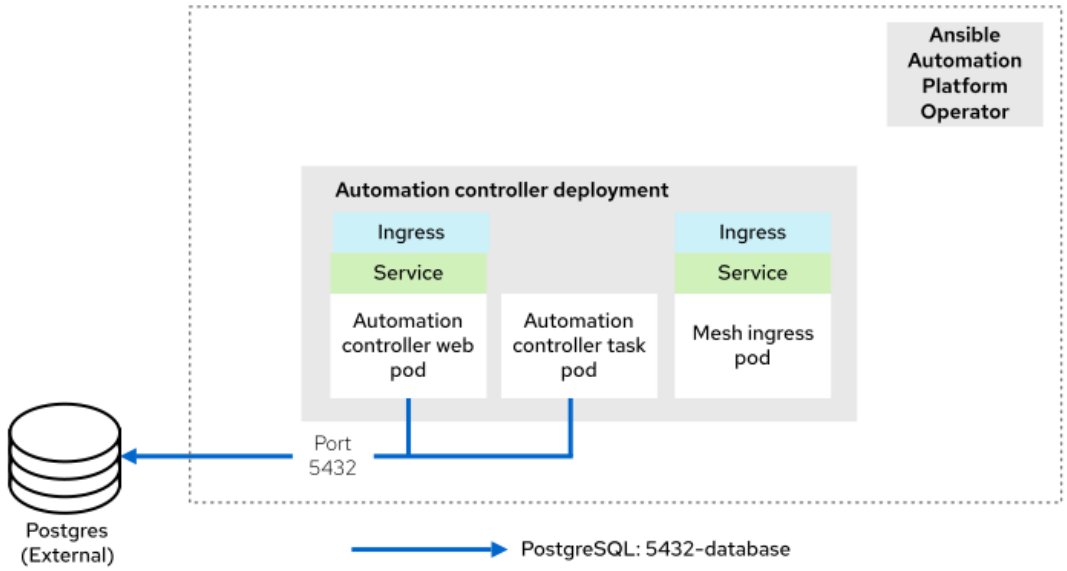
2.4 multi node automation controller deployment to a 2.6 enterprise topology

Upgrade your 2.4 multi-node automation controller setup to a 2.6 enterprise topology. Review the required infrastructure changes and requirements needed to successfully plan the upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

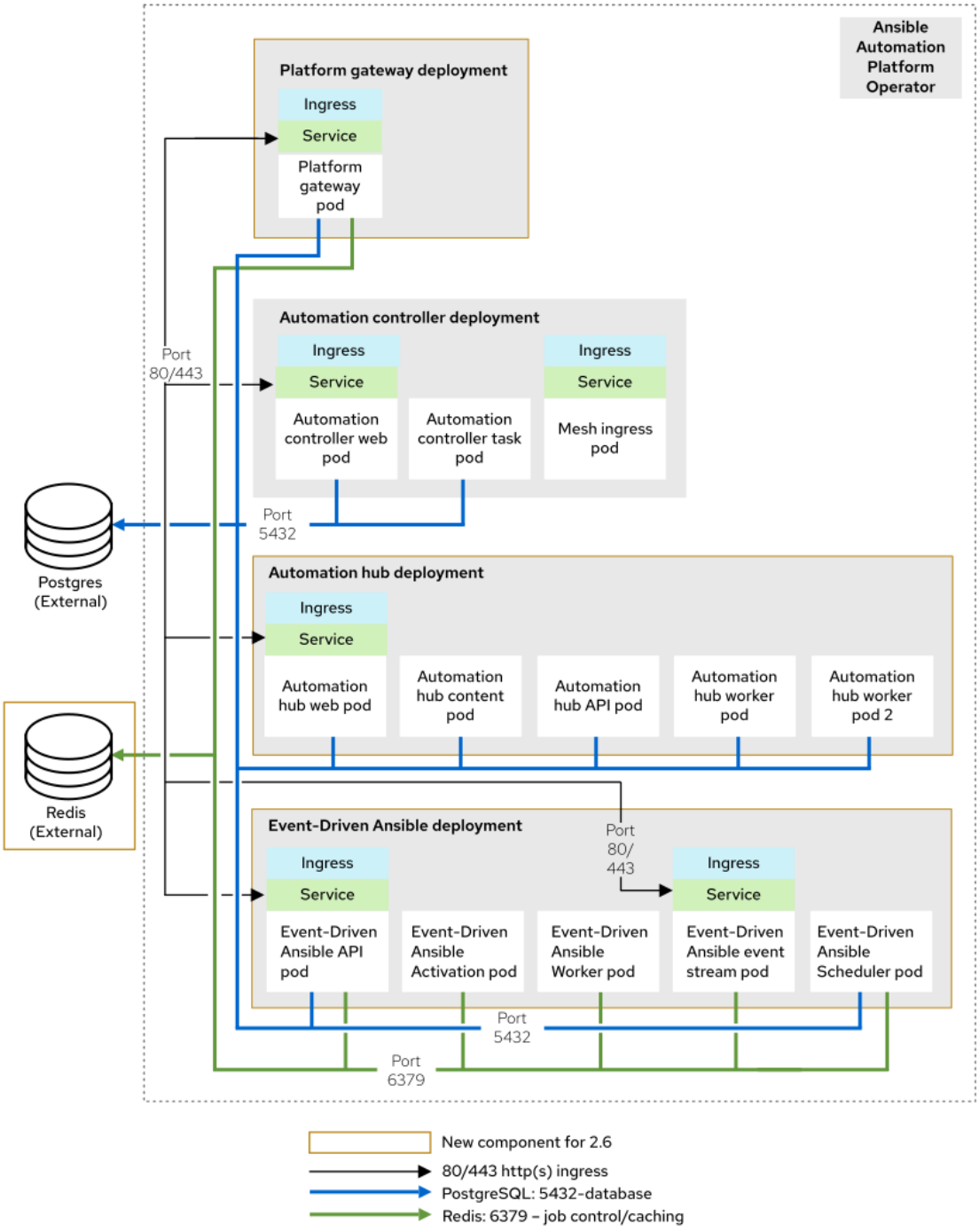
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a multi node automation controller deployment on OpenShift Container Platform

The following table highlights the requirements for upgrading from Ansible Automation Platform 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
Redundant automation controller-only deployment: • One automation controller web pod • One automation controller task pod • Two automation mesh ingress pods • Externally managed database service	Enterprise topology: • One automation controller web pod • One automation controller task pod • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • One automation hub Redis pod • One Event-Driven Ansible controller API pod • Two Event-Driven Ansible controller activation worker pods • Two Event-Driven Ansible controller default worker pods • Two Event-Driven Ansible controller event stream pods • One Event-Driven Ansible controller scheduler pod • One platform gateway pod • Two automation mesh ingress pods • Externally managed database service	See the **Operator enterprise topology** section of *Tested deployment models*.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
	• Externally managed Redis • Externally managed object storage service (for automation hub)	

Related information
Operator enterprise topology

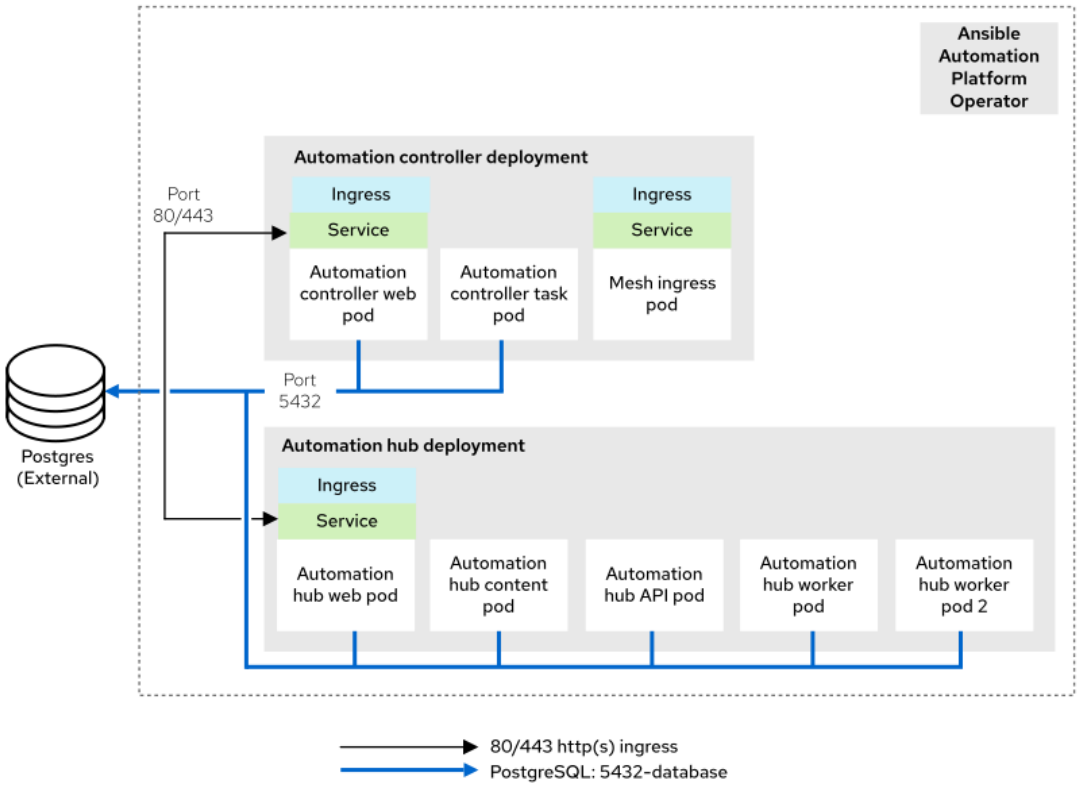
2.4 multi node automation controller and automation hub deployment to a 2.6 enterprise topology

Upgrade your 2.4 multi-node deployment (automation controller and automation hub) to a 2.6 enterprise topology. Review the infrastructure changes and requirements needed to successfully plan your upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

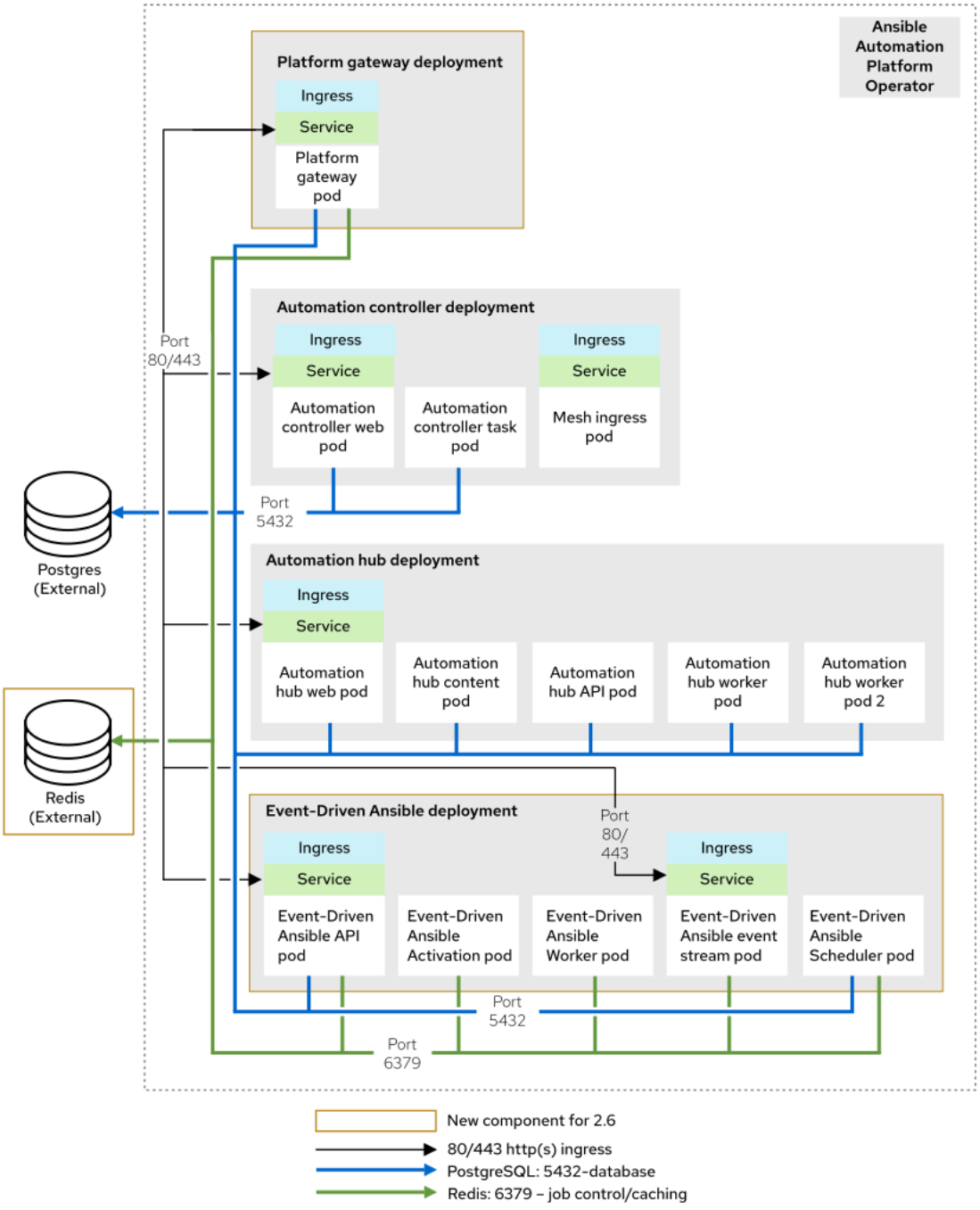
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a multi node automation controller and automation hub deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
Redundant deployment with automation controller and automation hub: • One automation controller web pod • One automation controller task pod • Two automation mesh ingress pods • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • Externally managed database service	Enterprise topology: • One automation controller web pod • One automation controller task pod • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • One automation hub Redis pod • One Event-Driven Ansible controller API pod • Two Event-Driven Ansible controller activation worker pods • Two Event-Driven Ansible controller default worker pods • Two Event-Driven Ansible controller event stream pods • One Event-Driven Ansible controller scheduler pod • One platform gateway pod • Two automation mesh ingress pods • Externally managed database service	See the **Operator enterprise topology** section of *Tested deployment models*.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
	• Externally managed Redis • Externally managed object storage service (for automation hub)	

Related information
Operator enterprise topology

Upgrade a growth topology on OpenShift Container Platform

You can upgrade your growth topology to the latest version of Ansible Automation Platform on OpenShift Container Platform.

For more information about the growth topology infrastructure requirements and configuration details, see the **Operator growth topology** section of *Tested deployment models*.

Related information
Operator growth topology

Upgrade an enterprise topology on OpenShift Container Platform

You can upgrade your enterprise topology to the latest version of Ansible Automation Platform on OpenShift Container Platform.

For more information about the enterprise topology infrastructure requirements and configuration details, see the **Operator enterprise topology** section of *Tested deployment models*.

Related information
Operator enterprise topology

Infrastructure changes for RPM deployments

The following sections describe the tested infrastructure changes for RPM-based deployments.

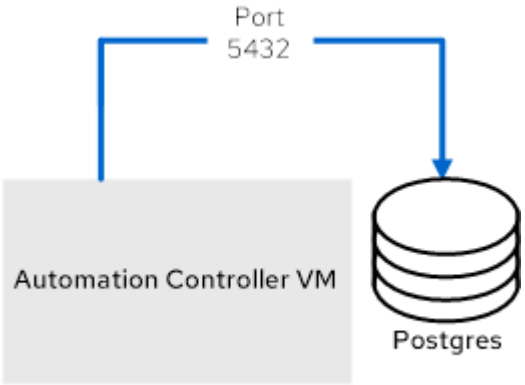
2.4 single automation controller node deployment to a 2.6 growth topology

Successfully upgrade your single automation controller node deployment (version 2.4) to a 2.6 growth topology. Use the provided infrastructure changes, requirements, and example inventory file to plan your upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

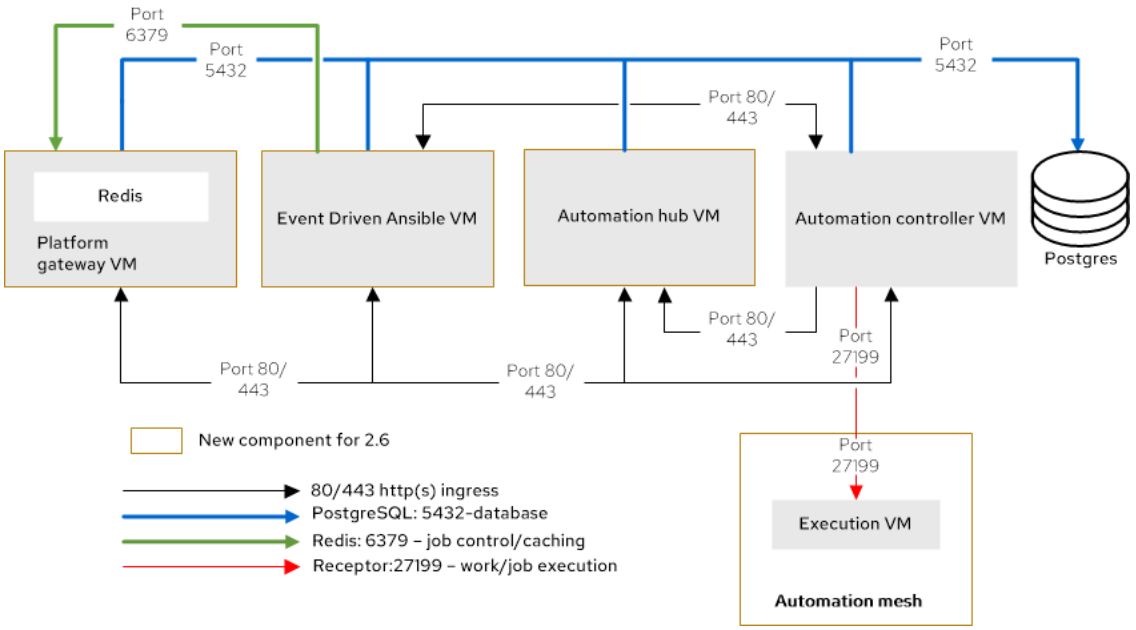
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a single automation controller node deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform version 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each VM
Non-redundant automation controller-only deployment: • One automation controller virtual machine (VM) • One Ansible Automation Platform managed PostgreSQL 15 database	Growth topology: • One platform gateway with colocated Redis VM • One automation controller VM • One private automation hub VM • One Event-Driven Ansible controller VM • One automation mesh execution node • One Ansible Automation Platform managed	For more information, see RPM growth topology.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each VM
	PostgreSQL 15 database	

Example inventory file

The following inventory file has been updated with the necessary changes to upgrade to the 2.6 growth topology.

```
# This is the RPM-based Ansible Automation Platform installer inventory file
intended for upgrading from a 2.4 single automation controller deployment to a 2.6
growth deployment.

# Consult the Ansible Automation Platform product documentation about this
topology's tested hardware configuration.

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/tested_deployment_models/rpm-topologies

# For all optional variables consult the Ansible Automation Platform documentation:

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation


# This section is for your platform gateway hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[automationgateway]

gateway.example.org


# This section is for your automation controller hosts from your 2.4 inventory

# -----------------------------------------------------

[automationcontroller]

controller.example.org


[automationcontroller:vars]

peers=execution_nodes


# This section is for your execution hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[execution_nodes]

exec.example.org
```

```
# This section is for your automation hub hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[automationhub]

hub.example.org


# This section is for your Event-Driven Ansible hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[automationedacontroller]

eda.example.org


# This section is for the Ansible Automation Platform database from your 2.4
inventory file

# -----------------------------------------------------

[database]

db.example.org


[all:vars]


# Common variables from your 2.4 inventory file

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#general-variables

# -----------------------------------------------------

registry_username=<your RHN username>

registry_password=<your RHN password>


# Common variables - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#general-variables

# -----------------------------------------------------

redis_mode=standalone


# Platform gateway - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#platform-gateway-variables

# -----------------------------------------------------

automationgateway_admin_password=<set your own>

automationgateway_pg_host=db.example.org

automationgateway_pg_password=<set your own>
```

```
# Automation controller variables from your 2.4 inventory file

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#controller-variables

# ------------------------------------------------------

admin_password=<set your own>

pg_host=db.example.org

pg_password=<set your own>


# Automation hub - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#hub-variables

# ------------------------------------------------------

automationhub_admin_password=<set your own>

automationhub_pg_host=db.example.org

automationhub_pg_password=<set your own>


# Event-Driven Ansible - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#event-driven-ansible-variables

# ------------------------------------------------------

automationedacontroller_admin_password=<set your own>

automationedacontroller_pg_host=db.example.org

automationedacontroller_pg_password=<set your own>
```

Related information
RPM growth topology

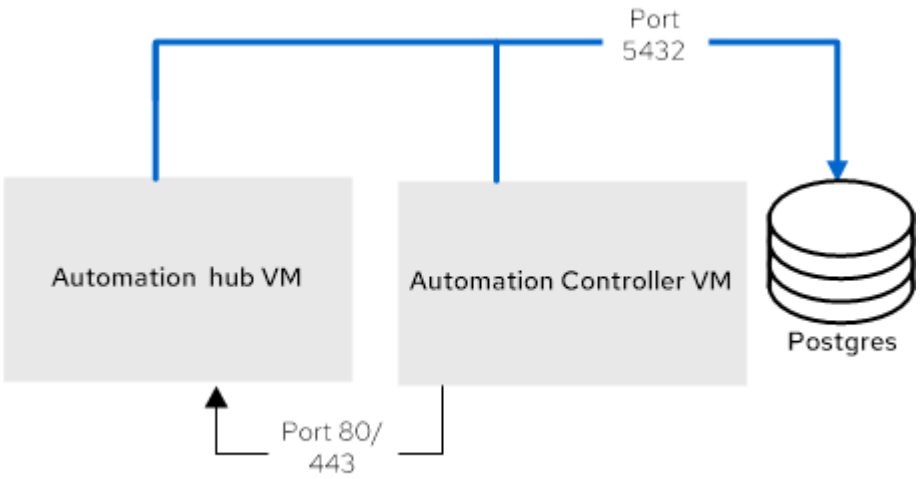
2.4 single node automation controller and automation hub deployment to a 2.6 growth topology

Upgrade your 2.4 single-node deployment (automation controller and automation hub) to a 2.6 growth topology. Review the infrastructure changes and requirements needed to successfully plan your upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

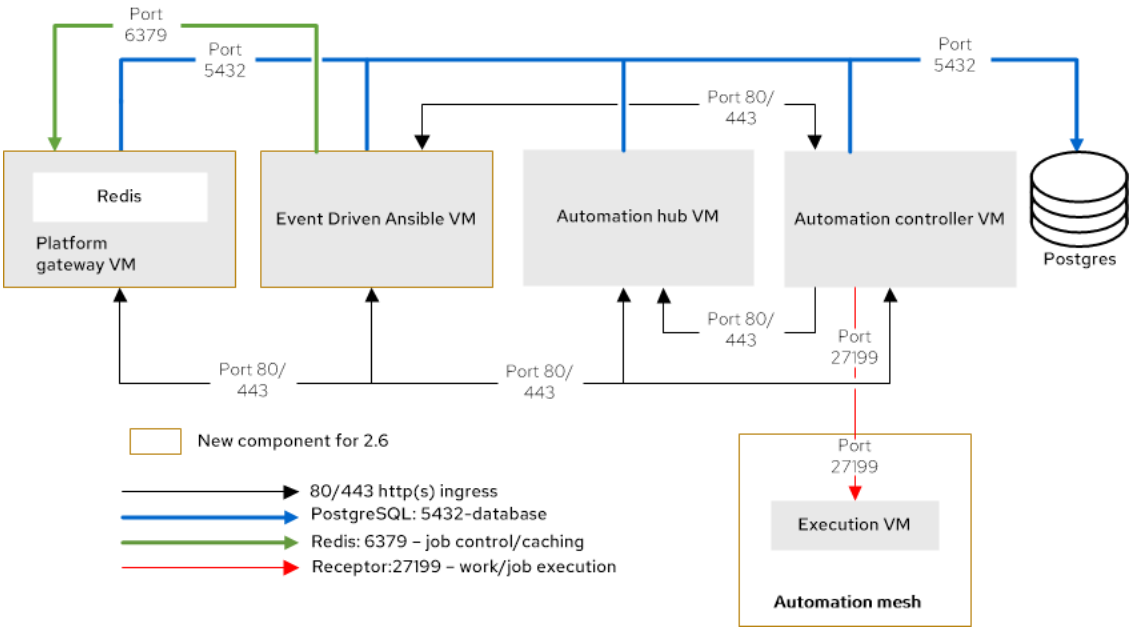
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a single automation controller node and automation hub deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform version 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each VM
Non-redundant deployment with automation controller and automation hub: • One automation controller VM • One automation hub VM • One Ansible Automation Platform managed PostgreSQL 15 database	Growth topology: • One platform gateway with colocated Redis VM • One automation controller VM • One private automation hub VM • One Event-Driven Ansible controller VM • One automation mesh execution node • One Ansible Automation Platform managed PostgreSQL 15 database	See *RPM growth topology*.

Example inventory file

The following inventory file has been updated with the necessary changes to upgrade to the 2.6 growth topology.

```
# This is the RPM-based Ansible Automation Platform installer inventory file
intended for upgrading from a 2.4 single automation controller and automation hub
deployment to a 2.6 growth deployment.


# Consult the Ansible Automation Platform product documentation about this
topology's tested hardware configuration.
```

```
# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/tested_deployment_models/rpm-topologies

# For all optional variables consult the Ansible Automation Platform documentation:

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation


# This section is for your platform gateway hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[automationgateway]

gateway.example.org


# This section is for your automation controller hosts from your 2.4 inventory

# -----------------------------------------------------

[automationcontroller]

controller.example.org


[automationcontroller:vars]

peers=execution_nodes


# This section is for your execution hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[execution_nodes]

exec.example.org


# This section is for your automation hub hosts from your 2.4 inventory

# -----------------------------------------------------

[automationhub]

hub.example.org


# This section is for your Event-Driven Ansible hosts - NEW for 2.6 growth topology

# -----------------------------------------------------

[automationedacontroller]

eda.example.org


# This section is for the Ansible Automation Platform database from your 2.4
inventory file

# -----------------------------------------------------

[database]
```

```
db.example.org


[all:vars]


# Common variables from your 2.4 inventory file

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#general-variables

# ------------------------------------------------------

registry_username=<your RHN username>

registry_password=<your RHN password>


# Common variables - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#general-variables

# ------------------------------------------------------

redis_mode=standalone


# Platform gateway - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#platform-gateway-variables

# ------------------------------------------------------

automationgateway_admin_password=<set your own>

automationgateway_pg_host=db.example.org

automationgateway_pg_password=<set your own>


# Automation controller variables from your 2.4 inventory file

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#controller-variables

# ------------------------------------------------------

admin_password=<set your own>

pg_host=db.example.org

pg_password=<set your own>


# Automation hub - NEW for 2.6 growth topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#hub-variables

# ------------------------------------------------------

automationhub_admin_password=<set your own>

automationhub_pg_host=db.example.org
```

```
automationhub_pg_password=<set your own>


# Event-Driven Ansible - NEW for 2.6 growth topology
# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/html/rpm_installation/appendix-inventory-files-vars#event-driven-ansible-variables
# ------------------------------------------------------
automationedacontroller_admin_password=<set your own>
automationedacontroller_pg_host=db.example.org
automationedacontroller_pg_password=<set your own>
```

Related information
RPM growth topology

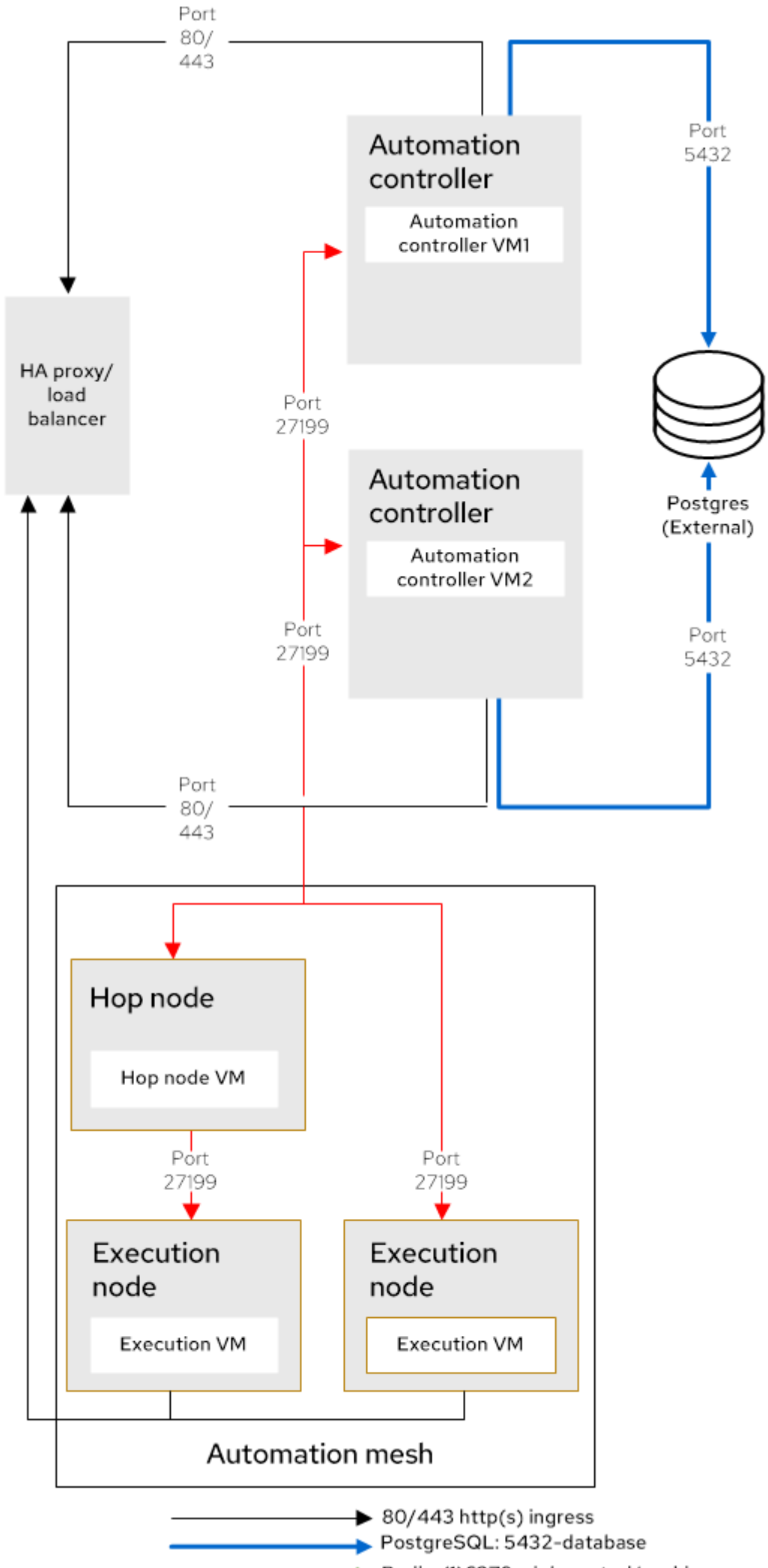
2.4 multi node automation controller deployment to a 2.6 enterprise topology

Upgrade your 2.4 multi-node automation controller setup to a 2.6 enterprise topology. Review the required infrastructure changes and requirements needed to successfully plan the upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

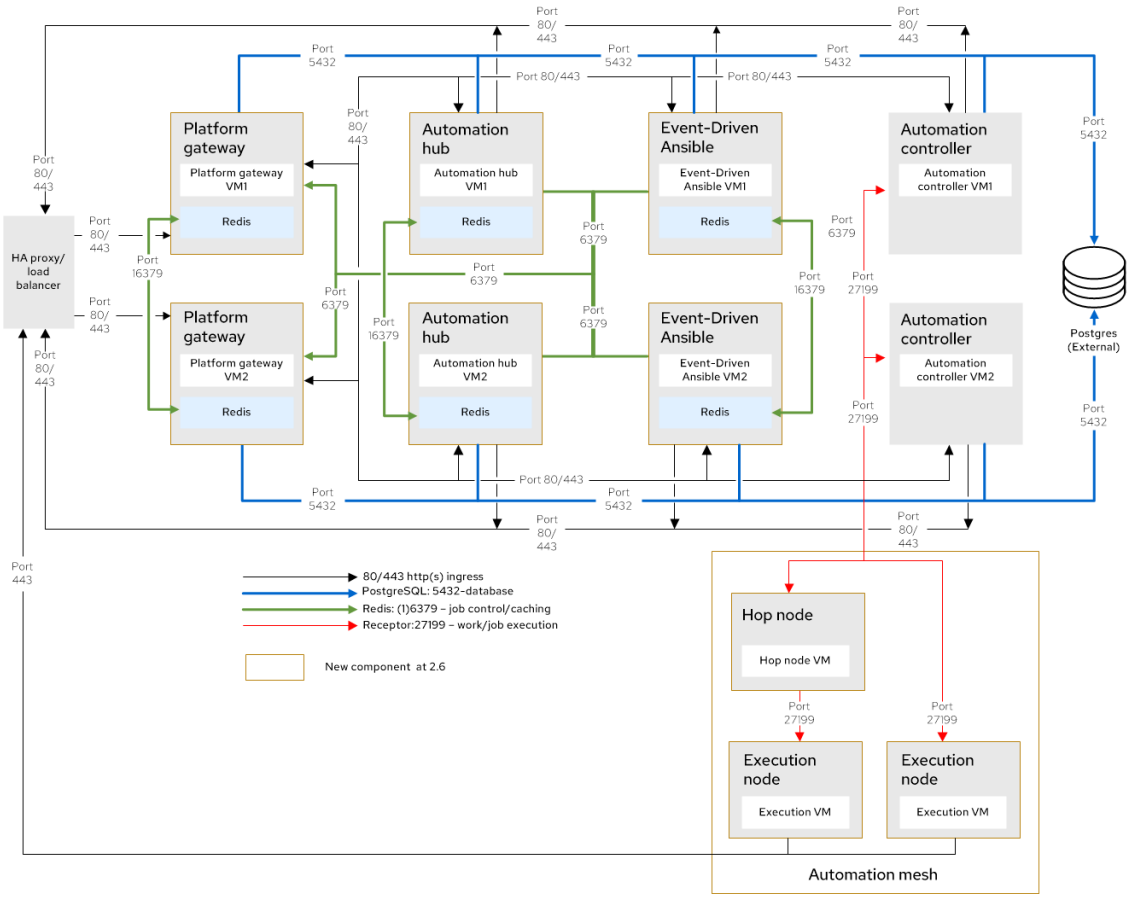
Figure: 2.4 infrastructure topology diagram


Port
80/
443
Automation
controller
Automation
controller VM1
Port
5432
HA proxy/
load
balancer
Port
27199
Automation
controller
Automation
controller VM2
Postgres
(External)
Port
27199
Port
5432
Port
80/
443
Hop node
Hop node VM
Port
27199
Port
27199
Execution
node
Execution VM
Execution
node
Execution VM
Automation mesh
80/443 http(s) ingress
PostgreSQL: 5432-database

2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a multi automation controller node deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform version 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each VM
Redundant automation controller-only deployment: • Two automation controller VMs	Enterprise topology: • Two platform gateway with colocated Redis VMs	See *RPM enterprise topology*.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each VM
• One automation mesh hop node VM • Two automation mesh execution node VMs • One customer-provided (external) PostgreSQL 15 database • One HA proxy load balancer in front of automation controller	• Two automation controller VMs • Two private automation hub with colocated Redis VMs • Two Event-Driven Ansible controller with colocated Redis VMs • One automation mesh hop node VM • Two automation mesh execution node VMs • One customer-provided (external) PostgreSQL 15 database • One HA proxy load balancer in front of platform gateway **Note**: Redis high availability requires 6 VMs. Redis can be colocated with automation hub, platform gateway, or Event-Driven Ansible components, but it cannot be colocated with automation controller, execution nodes, or the PostgreSQL database.	

Example inventory file

The following inventory file has been updated with the necessary changes to upgrade to the 2.6 enterprise topology.

```
# This is the RPM-based Ansible Automation Platform installer inventory file
intended for upgrading from a 2.4 multi node automation controller deployment to a
2.6 enterprise deployment.


# For all optional variables consult the Red Hat documentation:
```

```
# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation


# This section is for your platform gateway hosts - NEW for 2.6 enterprise topology

# -----------------------------------------------------

[automationgateway]

gateway1.example.org

gateway2.example.org


# This section is for your automation controller hosts from your 2.4 inventory

# -----------------------------------------------------

[automationcontroller]

controller1.example.org

controller2.example.org


[automationcontroller:vars]

peers=execution_nodes


# This section is for your execution hosts from your 2.4 inventory

# -----------------------------------------------------

[execution_nodes]

hop1.example.org node_type='hop'

exec1.example.org

exec2.example.org


# This section is for your automation hub hosts - NEW for 2.6 enterprise topology

# -----------------------------------------------------

[automationhub]

hub1.example.org

hub2.example.org


# This section is for your Event-Driven Ansible hosts - NEW for 2.6 enterprise
topology

# -----------------------------------------------------

[automationedacontroller]

eda1.example.org

eda2.example.org
```

```
# This section is for your Redis hosts - NEW for 2.6 enterprise topology
# -----------------------------------------------------
[redis]
gateway1.example.org
gateway2.example.org
hub1.example.org
hub2.example.org
eda1.example.org
eda2.example.org


[all:vars]
# Common variables from your 2.4 inventory
# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/html/rpm_installation/appendix-inventory-files-vars#general-variables
# -----------------------------------------------------
registry_username=<your RHN username>
registry_password=<your RHN password>


# Platform gateway - NEW for 2.6 enterprise topology
# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/html/rpm_installation/appendix-inventory-files-vars#platform-gateway-variables
# -----------------------------------------------------
automationgateway_admin_password=<set your own>
automationgateway_main_url=<set your own> #Set to the URL of the load balancer
automationgateway_pg_host=<set your own>
automationgateway_pg_database=<set your own>
automationgateway_pg_username=<set your own>
automationgateway_pg_password=<set your own>


# Automation controller variables from your 2.4 inventory
# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/html/rpm_installation/appendix-inventory-files-vars#controller-variables
# -----------------------------------------------------
admin_password=<set your own>
pg_host=<set your own>
pg_database=<set your own>
pg_username=<set your own>
```

```
pg_password=<set your own>


# Automation hub - NEW for 2.6 enterprise topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#hub-variables

# ------------------------------------------------------

automationhub_admin_password=<set your own>

automationhub_pg_host=<set your own>

automationhub_pg_database=<set your own>

automationhub_pg_username=<set your own>

automationhub_pg_password=<set your own>


# Event-Driven Ansible - NEW for 2.6 enterprise topology

# https://docs.redhat.com/en/documentation/red_hat_ansible_automation_platform/2.6/
html/rpm_installation/appendix-inventory-files-vars#event-driven-ansible-variables

# ------------------------------------------------------

automationedacontroller_admin_password=<set your own>

automationedacontroller_pg_host=<set your own>

automationedacontroller_pg_database=<set your own>

automationedacontroller_pg_username=<set your own>

automationedacontroller_pg_password=<set your own>
```

Related information
RPM enterprise topology

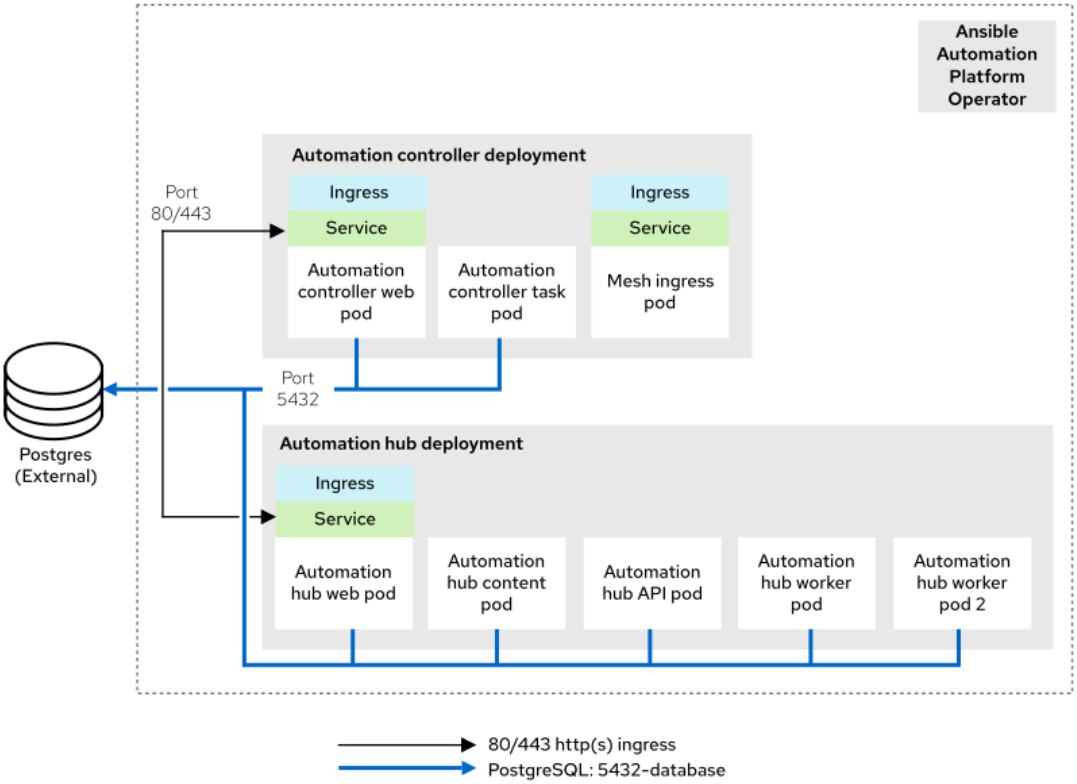
2.4 multi node automation controller and automation hub deployment to a 2.6 enterprise topology

Upgrade your 2.4 multi-node deployment (automation controller and automation hub) to a 2.6 enterprise topology. Review the infrastructure changes and requirements needed to successfully plan your upgrade.

2.4 infrastructure topology diagram

This diagram outlines the 2.4 infrastructure topology for this deployment model.

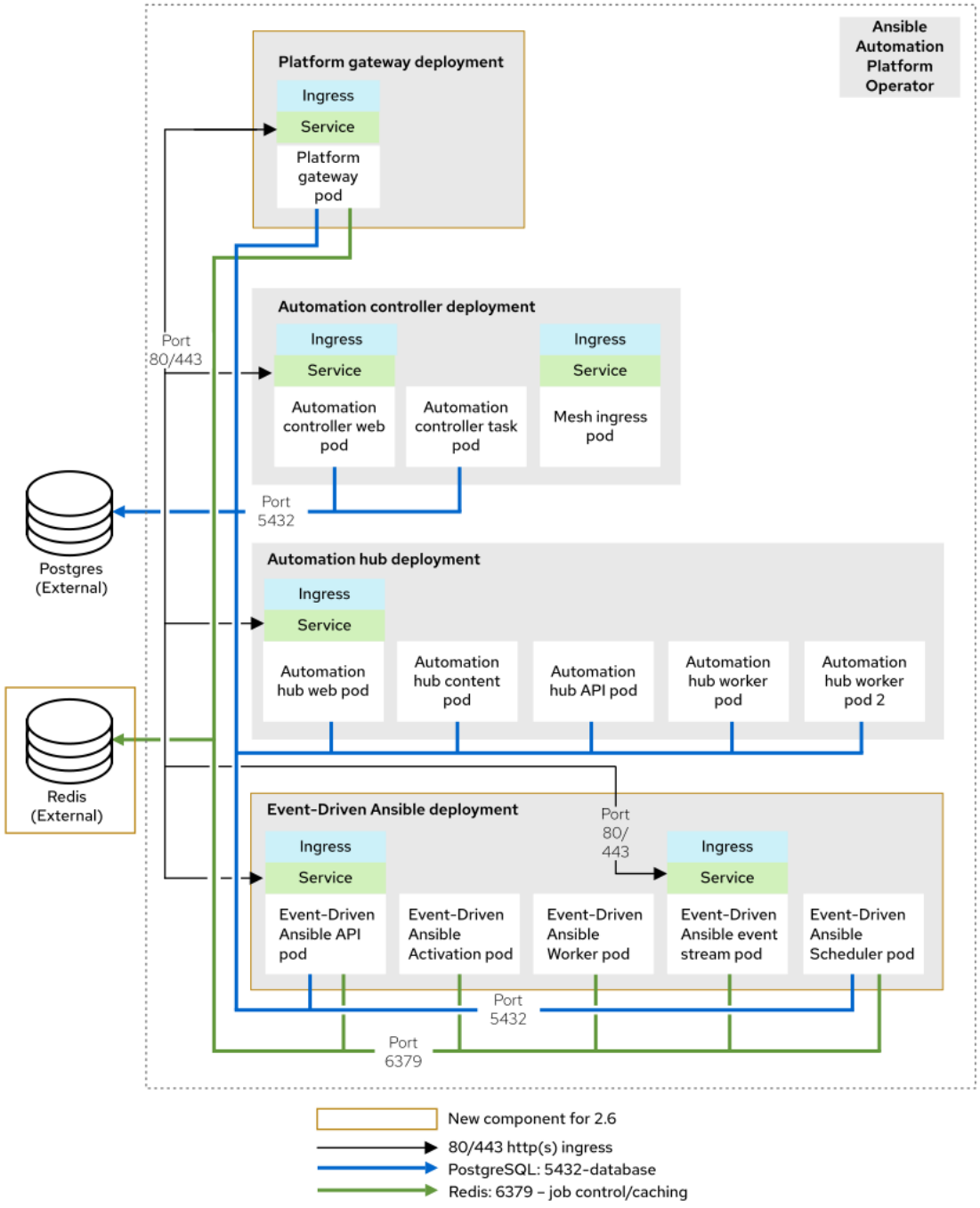
Figure: 2.4 infrastructure topology diagram



2.6 infrastructure topology diagram

This diagram outlines the 2.6 infrastructure topology that Red Hat has tested with this deployment model.

Figure: 2.6 infrastructure topology diagram



Requirements for upgrading a multi node automation controller and automation hub deployment

The following table highlights the requirements for upgrading from Ansible Automation Platform 2.4 to 2.6.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
Redundant deployment with automation controller and automation hub: • One automation controller web pod • One automation controller task pod • Two automation mesh ingress pods • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • Externally managed database service	Enterprise topology: • One automation controller web pod • One automation controller task pod • One automation hub web pod • One automation hub API pod • Two automation hub content pods • Two automation hub worker pods • One automation hub Redis pod • One Event-Driven Ansible controller API pod • Two Event-Driven Ansible controller activation worker pods • Two Event-Driven Ansible controller default worker pods • Two Event-Driven Ansible controller event stream pods • One Event-Driven Ansible controller scheduler pod • One platform gateway pod • Two automation mesh ingress pods • Externally managed database service	See the **Operator enterprise topology** section of *Tested deployment models*.

Existing 2.4 topology	Tested 2.6 topology	Requirements for each pod
	• Externally managed Redis • Externally managed object storage service (for automation hub)	

Related information
Operator enterprise topology

Upgrade an RPM-based growth topology

You can upgrade your RPM-based growth topology to the latest version of Ansible Automation Platform.

For more information about the growth topology infrastructure requirements and configuration details, see the **RPM growth topology** section of *Tested deployment models*.

Related information
RPM growth topology

Upgrade an RPM-based enterprise topology

You can upgrade your RPM-based enterprise topology to the latest version of Ansible Automation Platform.

Review the specific infrastructure requirements and configuration details to ensure your deployment model aligns with supported performance standards.

Related information
RPM enterprise topology

Identity and access management migration during upgrade

When upgrading from a version of Ansible Automation Platform that predates the platform gateway, Identity Access Management (IAM) data, including users, teams, organizations, their

memberships, and associated roles, is migrated from automation controller and automation hub to platform gateway.

This migration establishes automation controller as the primary source of IAM data for platform gateway, ensuring continuity of user memberships and appropriate platform-level role assignments.

NOTE:
If your current version is more than one minor release behind the target version, upgrade directly to the target version rather than performing intermediate upgrades. A direct upgrade is less complex.

Upgrade from 2.4 to 2.6

It is possible for customers to upgrade directly from the latest 2.4 version to 2.6. On startup, 2.6 platform services rename their service-specific roles to platform-wide roles, as shown in the following table.

2.4 role	2.6 equivalent
Automation controller auditor	Platform auditor
Automation controller superuser/ administrator (flag)	Platform superuser/administrator (flag)
Automation controller organization admin	Organization Administrator
Automation controller organization member	Organization member
Automation controller team admin	Team administrator
Automation hub administrator	Team member (user)
Automation controller team member (user)	Team member (user)
Automation hub team member (user)	Team member (user)

Additionally, note the following behavior when upgrading to 2.6:

- After upgrading, automation controller entity relationships, such as user associations with organizations, teams, or role sets, remain consistent in platform gateway. This applies to both existing entities moved during the upgrade and any new entities (users, teams, organizations) created in automation controller and subsequently moved in a 2.6 environment.
- Automation controller user types (normal, superuser and auditor) are mapped to platform gateway user types during the upgrade process.

- Where team names match between automation hub and platform gateway, for example, "Team A" exists in both, user memberships from automation hub are transferred to the corresponding team in platform gateway. This reduces the need to manually re-create memberships.
- If users exist only in automation hub, they are not moved to [Gateway]. You must manually re-create these users after upgrading. However, if a user has the same username in both automation controller and automation hub, the automation controller account is part of the regular data movement. Users who are not migrated need to have their passwords reset, but should keep the same permissions.
- Data movement also moves private automation hub but excludes Event-Driven Ansible data.
- Event-Driven Ansible users are not moved to platform gateway and must be recreated manually after upgrading.
- Automation hub users must be recreated if they cannot be moved as part of the upgrade. A user might not be migrated for the following reasons:
    - The user's account exists only in automation hub and not in automation controller.
    - Duplicate usernames exist in both automation hub and automation controller, but they belong to different people.
    - A discrepancy in the username, email, or other identifying attributes exists between the two services, which prevents the system from correctly merging the accounts.
- Automation hub admins are converted to normal users if they are able to be merged with automation controller users.
- The automation controller UI is updated to reflect the automation controller data moved as platform-level entities along with their roles.
- The automation controller setting **Organization Admins Can Manage Users and Teams** applies to organization admins in 2.6.

Nested team behavior changes

Ansible Automation Platform 2.5 and later versions no longer support nested team structures. This affects the UI, API, and collections.

In version 2.4, users could inherit permissions from multiple teams simultaneously through nested team structures created using REST APIs and collections. For example, a user on Team A could inherit permissions for Team B if Team B was nested under Team A (User → Team A → Team B). The user interface in 2.4 did not expose this capability; it was only possible through the API and collections.

During an upgrade from Ansible Automation Platform 2.4 to 2.6, nested teams are converted to a direct user-to-teams mapping. This means that instead of inheriting permissions through a nested structure, users are directly assigned to each team they have permissions for. For instance, if a

user previously had permissions through "User → Team A → Team B," after the upgrade, this becomes "User → Team A" and "User → Team B".

Impact and planning

- Users can still belong to one or more teams and simultaneously inherit permissions from those teams.
- Organizations that use integrations or automations with nested teams in their 2.4 deployment must plan to change this structure to a direct user-to-teams mapping.

IMPORTANT:
Before upgrading from Ansible Automation Platform 2.4, change any integrations or automations that implement nested teams to a direct user-to-teams mapping to avoid unexpected behavior in 2.5 and later.

Upgrade from 2.5 to 2.6

When upgrading from Ansible Automation Platform 2.5 to 2.6, existing authenticators and their mappings in platform gateway continue to function as they are, with no changes being imported.

This is because the core authentication service in 2.5 is already platform gateway, so a migration of this data is not needed.

Automation controller

Customers upgrading from 2.5 to 2.6 must also begin moving away from using nested teams in automation controller APIs, as future releases will disable direct access to service APIs.

After the upgrade, user data is synchronized between automation controller and the platform-wide authentication gateway.

Automation controller users, teams, roles, and organizations should become platform entities upon upgrade without the need to run additional "merge" processes. Customers that first upgraded from 2.4 to 2.5 will have teams that existed in 2.4 merged into platform gateway when they upgrade from 2.5 to 2.6.

Roles should apply the permission model for non-admin access to execution, content, and event services.

Automation hub

Understand identity changes for automation hub users when upgrading from 2.5 to 2.6. Review automatically merged teams, manually reassign permissions for removed admins, and reconfigure SSO to restore user access.

The following apply:

- A private automation hub admin (Automation Content Administrator) in 2.5 will be removed in the upgraded version and for this user the permissions must be reassigned manually as part of the data movement process.

IMPORTANT:
If teams with the same name exist in both automation hub and within the platform-wide authentication gateway, users from automation hub will be automatically added to corresponding teams within the platform-wide authentication gateway, and new teams will be created if they do not exist. This approach aims to retain team memberships, but requires careful review of permissions post-upgrade.

- If you rely on automation hub *Single Sign-On* (SSO) to access the automation hub user interface (UI), automation hub SSO logins will no longer function after the upgrade. However, API tokens will remain active. Therefore, automated processes or systems that use API tokens for authentication will continue to operate without interruption. If your workflows predominantly rely on API access, the impact might be minimal. However, if users primarily access the UI through SSO, they will need to take action post-upgrade.
- To restore UI access for users who previously relied on automation hub SSO, you need to reconfigure SSO within Ansible Automation Platform to be able to login. For further information, see Configuring Ansible Automation Platform Central Authentication Generic OIDC Settings and Red Hat SSO/KEYCLOAK for Red Hat SSO and Ansible Automation Platform.
- Automation controller admins will become platform admins and can administer automation hub.
- If you upgraded from 2.4 to 2.6 with both automation controller and private automation hub, then a dialog is displayed in the product post upgrade that informs you that there are steps to take to reconfigure private automation hub. This dialog can either display information in-product, or link to a product doc or Knowledge Base article. In either case, you will be guided to take action from within the product and not be expected to find that information unprompted.
- If you upgraded from 2.4 to 2.6 from an automation controller-only environment, then the addition of private automation hub and Event-Driven Ansible services involves adding the necessary roles to a normal user account to grant access to those services.

Event-Driven Ansible

When upgrading Event-Driven Ansible from version 2.5 to 2.6, users must reset their password to log in unless they use SSO. Administrators must manually reassign permissions for the former Automation Decisions Administrator role

The following apply:

- An Event-Driven Ansible administrator (Automation Decisions Administrator) in 2.5 will be removed in the upgraded version and for this user the permissions must be reassigned manually as part of the movement process.
- For Event-Driven Ansible, you must reset your password to log in to Ansible Automation Platform. You can still use your Event-Driven Ansible username but will require new passwords.
- If an Event-Driven Ansible user with SSO exists, then they will not have to reset password and should have their permissions moved over as part of the SSO migration.

Verify assigned permissions after upgrading

It is imperative that administrators verify the assigned permissions for all teams in the platform-wide authentication gateway immediately after the upgrade:

- Ensure the transferred team members have the correct access rights in the Ansible Automation Platform environment based on the filesystem.
- Make sure all members that have been merged are, in fact, the same member. Incorrect permissions could lead to access issues or security vulnerabilities.
- When the upgrade is complete, user accounts that exist in both the automation hub and automation controller systems will be unified, and platform gateway IAM will be the source of truth for users after the data movement.
- Automation hub and Event-Driven Ansible users must either be recreated or the users that moved from automation controller given permission to use those services.

After the upgrade is complete, verify that you can log in to the upgraded platform with your existing automation controller credentials (username and password).

NOTE:
To do this, you must have an automation controller account on Ansible Automation Platform 2.4 or 2.5 with administrative privileges.

The following table provides next steps for each type of user after they have upgraded.

If you are this type of user before upgrading:	Then take these actions after the upgrade:
An automation controller administrator (no automation hub account)	
An automation controller normal user (no automation hub account)	Log in with your automation controller username and password; you are now a platform gateway normal user.
An automation hub user (no automation controller account)	Request a password reset from your administrator. When you log in with your new password you will be a platform gateway normal user. You will retain your hub-related permissions.
An automation controller and automation hub user (with the same username in both services)	Log in with your automation controller username and password; your previous two accounts will be merged and you are now a platform gateway normal user.
An automation hub user with SSO (no automation controller account)	Log in with your SSO credentials; you are now a platform gateway normal user.

The `MANAGE_ORGANIZATION_AUTH` setting

The automation controller setting previously called **Organization Admins Can Manage Users and Teams** in the UI (or `MANAGE_ORGANIZATION_AUTH` in the API) controls whether an organization administrator can create users and teams.

This setting now exists in both platform gateway and automation controller in Ansible Automation Platform 2.6. During an upgrade the value from automation controller is imported into the platform gateway server. If you decide to change the value of this setting ensure that you change it to the same values in both the platform gateway and automation controller.

IMPORTANT:
For environments with automation running directly against automation controller, maintain a consistent value for `MANAGE_ORGANIZATION_AUTH` across both automation controller and platform gateway to avoid unexpected behavior.

Authentication provider migration behavior

During an upgrade from a version of Ansible Automation Platform that predates the platform gateway, only complete authentication provider configurations are migrated to the platform gateway.

A configuration is considered complete when it meets the following criteria:

- **LDAP**: You must specify a server URL.
- **GitHub and Microsoft Azure AD**: You must specify both a key and a secret.
- **OIDC**: You must define a key, a secret, and an OIDC endpoint.
- **RADIUS** and **TACACS+**: You must specify the host.

Before proceeding with the upgrade, ensure that you complete the following steps:

- **Create a local administrator account** and verify that you can log in to the environment using local authentication. You can also use the default administrator account from the inventory file.
- **Enable the local authenticator** in the target environment to ensure a fallback login method is available.
- **Perform a full backup** of your existing environment.

> **IMPORTANT:**
> This is a critical step for data recovery in case any issues occur during the migration process.

Post upgrade

- **Update the callback URLs** in your *Identity Provider* (IdP) configurations after the migration. This is necessary for OAuth and SSO providers to function correctly with the platform gateway architecture.
- **Reestablish custom certificates for LDAPS** if your LDAP authentication uses custom certificates in the system's truststore. This configuration is not automatically migrated and you must manually reestablish it.

The migration of existing authentication configurations from automation controller to the platform gateway is automated. The following tables show how settings and mappings from the automation controller schema are transformed to fit the platform gateway API schema.

Related information
Update callback URLs for OAuth and SSO providers

Authentication type: OIDC

Review the general settings and mappings for OpenID Connect (OIDC) authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6

`SOCIAL_AUTH_OIDC_KEY: "client-id"`
 	
`SOCIAL_AUTH_OIDC_SECRET: “client-secret"`
 	
`SOCIAL_AUTH_OIDC_OIDC_ENDPOINT: "https://idp.example.com"`
 	
`SOCIAL_AUTH_OIDC_VERIFY_SSL: true`
	`"configuration": {`
	
	`"OIDC_ENDPOINT": "https://idp.example.com",`
	
	`"KEY": "client-id",`
	
	`"SECRET": "client-secret",`
	
	`"VERIFY_SSL": true`
	
	`}`

Mappings

Automation controller 2.4	Platform gateway 2.6

`AUTH_LDAP_ORGANIZATION_MAP:`
 	
`"LDAP Organization":`
 	
`users: true`
	`"name": "Default - Users (users)",`
	
	`"map_type": "organization",`
	
	`"order": 1,`
	
	`"authenticator": -1,`
	
	`"triggers": {`
	
	`"users": true`
	
	`},`
	
	`"organization": "Default",`
	
	`"team": null,`
	
	`"role": "Organization Member",`
	
	`"revoke": true`
	
	`}`

Automation controller 2.4	Platform gateway 2.6

SOCIAL_AUTH_SAML_USER_FLAGS_BY_ATTR
:
is_superuser_attr: "is_superuser"
is_superuser_value: "true"
	{
	"name": "is_superuser - role",
	"authenticator": -1,
	"revoke": true,
	"map_type": "is_superuser",
	"team": null,
	"organization": null,
	"triggers": {
	"attributes": {
	"is_superuser": {
	"has_or": [
	"true"
	]
	}
	}
	},
	"order": 2
	}

Authentication type: LDAP

Review the general settings and mappings for LDAP authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6

`AUTH_LDAP_SERVER_URI: "ldap://ldap.example.com:389"`
`AUTH_LDAP_BIND_DN: "cn=admin,dc=example,dc=org"`
`AUTH_LDAP_BIND_PASSWORD: "password"`
`AUTH_LDAP_START_TLS: false`
`AUTH_LDAP_USER_SEARCH: [`
`"ou=users,dc=example,dc=org",`
`"SCOPE_SUBTREE", "(cn=%(user)s)"`
`]`
`AUTH_LDAP_USER_ATTR_MAP: {`
`"first_name": "givenName",`
`"last_name": "sn",`
`"email": "mail"`
`}`
	`"configuration": {`
	`"SERVER_URI": "ldap://ldap.example.com:389",`
	`"BIND_DN": "cn=admin,dc=example,dc=org",`
	`"BIND_PASSWORD": "password",`
	`"START_TLS": false,`
	`"USER_SEARCH": [`
	`"ou=users,dc=example,dc=org",`
	`"SCOPE_SUBTREE",`
	`"(cn=%(user)s)"`
	`],`
	`"USER_ATTR_MAP": {`
	`"first_name": "givenName",`
	`"last_name": "sn",`
	`"email": "mail"`
	`}`
	`}`

Mappings

Automation controller 2.4	Platform gateway 2.6

AUTH_LDAP_ORGANIZATION_MAP:
 	
"LDAP Organization":
 	
users: true
 	
admins:
 	
-
"cn=awx_org_admins,ou=groups,dc=example,dc=org"
	{
	
	"name": "LDAP Organization - Admins cn=awx_org_admins,ou=groups,dc=example,dc=org",
	
	"map_type": "organization",
	
	"order": 1,
	
	"authenticator": -1,
	
	"triggers": {
	
	"groups": {
	
	"has_or": [
	
	"cn=awx_org_admins,ou=groups,dc=example,dc=org"
	
	]
	
	}
	
	},
	
	"organization": "LDAP Organization",
	
	"team": null,
	
	"role": "Organization Admin",
	
	"revoke": false
	
	}

Automation controller 2.4	Platform gateway 2.6

`AUTH_LDAP_USER_FLAGS_BY_GROUP:`
`  is_superuser:`
`  - 'cn=awx_admins,ou=groups,dc=example,dc=org'`
	`{`
	`  "name": "is_superuser - role",`
	`  "authenticator": -1,`
	`  "revoke": true,`
	`  "map_type": "is_superuser",`
	`  "team": null,`
	`  "organization": null,`
	`  "triggers": {`
	`  "groups": {`
	`  "has_or": [`
	`"cn=awx_admins,ou=groups,dc=example,dc=org"`
	`  ]`
	`  }`
	`  },`
	`  "order": 2`
	`}`

Authentication type: SAML

Understand how Security Assertion Markup Language (SAML) provides secure, token-based authentication for your upgraded system. Implement SAML to ensure seamless single sign-on (SSO) and centralized identity management.

IMPORTANT:
Automation controller in Ansible Automation Platform 2.4 allowed customers to enter an encrypted private key in SAML configuration without raising an error. If request signing was not enabled in the authenticator and the SAML IdP, then the Ansible Automation Platform administrator would not know that encrypted keys were not supported. Encrypted keys not supported in Ansible Automation Platform 2.6 authenticators. The platform alerts users that encrypted keys are not supported. However, when upgrading from Ansible Automation Platform 2.4 to 2.6, customers must replace encrypted private keys with unencrypted private keys in their SAML authenticators to prevent migration errors for the authenticator to platform gateway. If you skip this step, the authenticator is not migrated as part of the upgrade. The SAML authenticator must then be recreated manually by a local administrator to re-enable authentication. This might delay SSO users from logging back into the platform after the upgrade.

General settings

Automation controller 2.4

```
SOCIAL_AUTH_SAML_ENABLED_IDPS:

  Keycloak: null

  entity_id: 'https://
idp.example.com/auth/realms/awx'

  url: 'https://idp.example.com/
auth/realms/awx/protocol/saml'

  x509cert: MIICert...

  attr_username: username

  attr_email: email

SOCIAL_AUTH_SAML_SP_ENTITY_ID:
'https://
controller.example.com:8043'

SOCIAL_AUTH_SAML_SP_PUBLIC_CERT:
MIICertPublic...

SOCIAL_AUTH_SAML_SP_PRIVATE_KEY:
MIIKeyPrivate...
```

Platform gateway 2.6

```
"configuration": {

  "IDP_URL": "https://
idp.example.com/auth/realms/awx/
protocol/saml",

  "IDP_X509_CERT": "-----BEGIN
CERTIFICATE-----\nMIICert...\n-----
END CERTIFICATE-----",

  "IDP_ENTITY_ID": "https://
idp.example.com/auth/realms/awx",

  "IDP_ATTR_EMAIL": "email",

  "IDP_ATTR_USERNAME": "username",

  "SP_ENTITY_ID": "https://
controller.example.com:8043",

  "SP_PUBLIC_CERT":
"MIICertPublic...",

  "SP_PRIVATE_KEY":
"MIIKeyPrivate..."

}
```

Mappings

Automation controller 2.4	Platform gateway 2.6

SOCIAL_AUTH_SAML_ORGANIZATION_MAP:
"Default":
users: true
	{
	"name": "Default - Users (users)",
	"map_type": "organization",
	"order": 1,
	"authenticator": -1,
	"triggers": {
	"users": true
	},
	"organization": "Default",
	"team": null,
	"role": "Organization Member",
	"revoke": true
	}

Automation controller 2.4	Platform gateway 2.6
SOCIAL_AUTH_SAML_USER_FLAGS_BY_ATTR : is_superuser_attr: "is_superuser" is_superuser_value: "true"	{ "name": "is_superuser - role", "authenticator": -1, "revoke": true, "map_type": "is_superuser", "team": null, "organization": null, "triggers": { "attributes": { "is_superuser": { "has_or": ["true"] } } }, "order": 2 }

Authentication type: Github

Review the general settings and mappings for Github authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6
SOCIAL_AUTH_GITHUB_KEY: client-id SOCIAL_AUTH_GITHUB_SECRET: client-secret SOCIAL_AUTH_GITHUB_SCOPE: - 'user:email' - 'read:org'	{ "configuration": { "KEY": "client-id", "SECRET": "client-secret", "SCOPE": ["user:email", "read:org"] } }

Mappings

Automation controller 2.4	Platform gateway 2.6
SOCIAL_AUTH_GITHUB_ORGANIZATION_MAP : "MyOrg": users: true admins: - "admin-team"	{ "name": "MyOrg - Admins admin-team", "map_type": "organization", "order": 1, "authenticator": -1, "triggers": { "users": { "has_or": ["admin-team"] } }, "organization": "MyOrg", "team": null, "role": "Organization Admin", "revoke": false }

Automation controller 2.4	Platform gateway 2.6

`SOCIAL_AUTH_GITHUB_TEAM_MAP:`
`"Developers":`
`organization: "MyOrg"`
`users:`
`- "dev-team"`
	`{`
	`"name": "MyOrg - Developers dev-team",`
	`"map_type": "team",`
	`"order": 2,`
	`"authenticator": -1,`
	`"triggers": {`
	`"users": {`
	`"has_or": [`
	`"dev-team"`
	`]`
	`}`
	`},`
	`"organization": "MyOrg",`
	`"team": "Developers",`
	`"role": "Team Member",`
	`"revoke": false`
	`}`

Authentication type: Azure AD

Review the general settings and mappings for Azure AD authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6

SOCIAL_AUTH_AZUREAD_OAUTH2_KEY: "application-id"
 	
SOCIAL_AUTH_AZUREAD_OAUTH2_SECRET: "client-secret"
	"configuration": {
	
	"KEY": "application-id",
	
	"SECRET": "client-secret",
	
	"GROUPS_CLAIM": "groups"
	
	}

Mappings

Automation controller 2.4	Platform gateway 2.6

SOCIAL_AUTH_AZUREAD_OAUTH2_ORGANIZATION_MAP:
 	
"Azure Organization":
 	
users: true
	{
	
	"name": "Azure Organization - Users (users)",
	
	"map_type": "organization",
	
	"order": 1,
	
	"authenticator": -1,
	
	"triggers": {
	
	"users": true
	
	},
	
	"organization": "Azure Organization",
	
	"team": null,
	
	"role": "Organization Member",
	
	"revoke": false
	
	}

Automation controller 2.4	Platform gateway 2.6

SOCIAL_AUTH_AZUREAD_OAUTH2_TEAM_MAP :
 	
"Admin Team":
 	
organization: "Azure Organization"
 	
users:
 	
- "admin@company.com"
	{
	
	"name": "Azure Organization - Admin Team admin@company.com",
	
	"map_type": "team",
	
	"order": 2,
	
	"authenticator": -1,
	
	"triggers": {
	
	"emails": {
	
	"has_or": [
	
	"admin@company.com"
	
	]
	
	}
	
	},
	
	"organization": "Azure Organization",
	
	"team": "Admin Team",
	
	"role": "Team Member",
	
	"revoke": false
	
	}

Authentication type: RADIUS

Review the general settings and mappings for RADIUS authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6

`RADIUS_SERVER: "radius.example.com"`
`RADIUS_PORT: 1812`
`RADIUS_SECRET: "shared-secret"`
	`"configuration": {`
	`"SERVER": "radius.example.com",`
	`"PORT": 1812,`
	`"SECRET": "shared-secret"`
	`}`

Mappings

RADIUS authentication does not support user mappings in either automation controller 2.4 or Platform gateway 2.6.

Authentication type: TACACS+

Review the general settings and mappings for TACACS+ authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6

`TACACSPLUS_HOST: "tacacs.example.com"`
`TACACSPLUS_PORT: 49`
`TACACSPLUS_SECRET: "shared-secret"`
`TACACSPLUS_SESSION_TIMEOUT: 5`
`TACACSPLUS_AUTH_PROTOCOL: "ascii"`
`TACACSPLUS_REM_ADDR: false`
	`"configuration": {`
	`"HOST": "tacacs.example.com",`
	`"PORT": 49,`
	`"SECRET": "shared-secret",`
	`"SESSION_TIMEOUT": 5,`
	`"AUTH_PROTOCOL": "ascii",`
	`"REM_ADDR": false`
	`}`

Mappings

TACACS+ authentication does not support user mappings in either automation controller 2.4 or Platform gateway 2.6.

Authentication type: Google OAuth2

Review the general settings and mappings for Google OAuth2 authentication. Compare how configurations transform from automation controller 2.4 to the platform gateway 2.6.

General settings

Automation controller 2.4	Platform gateway 2.6

`SOCIAL_AUTH_GOOGLE_OAUTH2_KEY: "client-id"`
 	
`SOCIAL_AUTH_GOOGLE_OAUTH2_SECRET: "client-secret"`
 	
`SOCIAL_AUTH_GOOGLE_OAUTH2_SCOPE: ["profile", "email"]`
	`{`
	`  "configuration": {`
	`  "KEY": "client-id",`
	`  "SECRET": "client-secret",`
	`  "REDIRECT_STATE": true,`
	`  "SCOPE": [`
	`  "profile",`
	`  "email"`
	`  ]`
	`  }`
	`}`

Mappings

Automation controller 2.4	Platform gateway 2.6

SOCIAL_AUTH_GOOGLE_OAUTH2_ORGANIZATION_MAP:
"Google Org":
users: true
	{
	"name": "Google Org - Users (users)",
	"map_type": "organization",
	"order": 1,
	"authenticator": -1,
	"triggers": {
	"users": true
	},
	"organization": "Google Org",
	"team": null,
	"role": "Organization Member",
	"revoke": false
	}

SOCIAL_AUTH_GOOGLE_OAUTH2_TEAM_MAP:
"Engineers":
organization: "Google Org"
users: true
	{
	"name": "Google Org - Engineers (users)",
	"map_type": "team",
	"order": 2,
	"authenticator": -1,
	"triggers": {
	"users": true
	},
	"organization": "Google Org",
	"team": "Engineers",
	"role": "Team Member",
	"revoke": false
	}

API changes for platform gateway

Ansible Automation Platform uses a platform gateway that provides centralized API access to all services. While APIs for automation controller, automation hub, and Event-Driven Ansible remain accessible directly for backward compatibility, this direct access will be removed in a future release.

These changes impact your organization if you have API calls implemented directly with automation controller or private automation hub, or if you are integrating directly with automation controller or private automation hub hosts. You must migrate these integrations to the API endpoints exposed through the platform gateway to ensure they are not disrupted when direct service API access is removed in a future Ansible Automation Platform release.

For detailed API reference information, see the following sources:

- For platform gateway APIs, see the browsable API at `https://<gateway server name>/api/gateway/v1`.
- For automation controller APIs, see the browsable API at `https://<gateway server name>/api/controller/v2`.
- For automation hub APIs, see **Automation Hub API** in *API Catalog and Documentation*.
- For Event-Driven Ansible APIs, see the browsable API at `https://<gateway server name>/api/eda/v1`.

Related information
Automation Hub API

General changes

In Ansible Automation Platform 2.5 and later, API endpoints across components changed with the addition of platform gateway.

Component	2.4 and earlier endpoints start with...	2.5 and later endpoints start with...	Notes
Automation controller	`/api/v2/`	`/api/controller/v2/`	
Automation hub	`/api/automation-hub`	`/api/galaxy/v1`	This is the default path, but this path can be changed. For example: `https://<local_hub_URL>/api/`

Component	2.4 and earlier endpoints start with...	2.5 and later endpoints start with...	Notes
Platform gateway	Not applicable	`/api/gateway/v1/`	
Event-Driven Ansible	Not applicable	`/api/eda/v1/`	

Specific API changes

Specific API mappings for functionality that was centralized through the platform gateway are listed in the following table.

Component	2.4 and earlier endpoints start with...	2.5 and 2.6 API endpoints	Action needed and notes
Automation controller	`/api/v2/o`	`/api/gateway/v1/tokens/`	Token authentication has moved to the platform gateway. The 2.4 API endpoint is deprecated; it still works in 2.6, but it will not work in a future release.
Automation controller	`/api/v2/organizations`	`/api/gateway/v1/organizations/`	Moved to the platform gateway. The 2.4 API endpoint is deprecated; it still works in 2.6, but it will not work in a future release.
Automation controller	`/api/v2/teams`	`/api/gateway/v1/teams/`	Moved to the platform gateway. The 2.4 API endpoint is deprecated; it still works in 2.6, but it will not work in a future release.

Component	2.4 and earlier endpoints start with...	2.5 and 2.6 API endpoints	Action needed and notes
Automation controller	`/api/v2/users`	`/api/gateway/v1/users/`	Moved to the platform gateway. The 2.4 API endpoint is deprecated; it still works in 2.6, but it will not work in a future release.
Automation controller	`/api/v2/roles`	`/api/gateway/v1/role_definitions/`	Moved to the platform gateway. This is a list of roles. In Ansible Automation Platform 2.6, this is a list of roles which can apply to all services, and includes custom roles. The 2.4 API endpoint is only a listing. It still works in 2.6, but it will not work in a future release.
Automation controller	• `/api/v2/roles/{id}/teams/` • `/api/gateway/v1/role_definitions/`	• `/api/gateway/v1/role_team_assignments/` • `/api/gateway/v1/role_user_assignments/`	A POST request gives a user a role to a resource. This is how to give user permissions. The 2.4 API endpoint is only a listing. It still works in 2.6, but it will not work in a future release.
Automation controller	The following roles list:	• `/api/gateway/v1/role_team_as`	List user and team permissions, and give new permissions.

Component	2.4 and earlier endpoints start with...	2.5 and 2.6 API endpoints	Action needed and notes
	• `/api/v2/teams/{id}/roles/` • `/api/v2/users/{id}/roles/`	`signments/?team={id}` • `/api/gateway/v1/role_user_assignments/?user={id}`	The 2.4 API endpoint is only a listing. It still works in 2.6, but it will not work in a future release.
Automation controller	The following object roles list: `/api/v2/{model_name}/{id}/object_roles/` Example: `/api/v2/credentials/42/`	`/api/gateway/v1/role_user_assignments/?content_type__api_slug={model_api_slug}&object_id={id}` Example: `/api/gateway/v1/role_user_assignments/?content_type__api_slug=awx.credential&object_id=42`	List the roles that apply to a resource.
Automation controller	The following resource access list: `/api/v2/{model_name}/{id}/access_list/` Example: `/api/v2/credentials/42/access_list/`	Replacement in 2.6: `/api/gateway/v1/role_user_access/{model_api_slug}/{id}/` Example: `/api/gateway/v1/role_user_access/awx.credential/42/`	List the users who have access to a resource.
Automation hub	`/api/v3/login/keycloak`	`/api/gateway/social/complete/<UID>/`	Moved to the platform gateway.
Automation hub	`/api/v3/auth/token`	`/api/gateway/v1/tokens/`	Token authentication used for pulling collections will migrate to the

Component	2.4 and earlier endpoints start with...	2.5 and 2.6 API endpoints	Action needed and notes
			platform gateway tokens.
Event-Driven Ansible	N/A	`/api/gateway/v1/organizations/`	No action needed, as upgrades from 2.4 are not supported.
Event-Driven Ansible	N/A	`/api/gateway/v1/teams/`	No action needed, as upgrades from 2.4 are not supported.
Event-Driven Ansible	N/A	`/api/gateway/v1/users/`	No action needed, as upgrades from 2.4 are not supported.
Event-Driven Ansible	N/A	• `/api/gateway/v1/role_definitions/` • `/api/gateway/v1/role_team_assignments/` • `/api/gateway/v1/role_user_assignments/`	New role capabilities included as part of the platform gateway API.

Upgrade your containerized deployment of Ansible Automation Platform

Perform an upgrade of containerized Ansible Automation Platform.

Before you begin

- You have reviewed the release notes for the associated release. For more information, see Release notes.
- You have a backup of your Ansible Automation Platform deployment. For more information, see Back up containerized Ansible Automation Platform.

Procedure

1. Log in to the Red Hat Enterprise Linux host as your dedicated non-root user.
2. Follow the steps in Download Ansible Automation Platform to download the latest version of containerized Ansible Automation Platform.
3. Copy the downloaded installation program to your Red Hat Enterprise Linux Host.
4. Edit the `inventory` file to match your required configuration. You can keep the same parameters from your existing Ansible Automation Platform deployment or you can change the parameters to match any modifications to your environment.
5. Run the `install` playbook:

```
$ ansible-playbook -i inventory ansible.containerized_installer.install
```

- If your privilege escalation requires a password to be entered, append `-K` to the command. You will then be prompted for the `BECOME` password.
- You can use increasing verbosity, up to 4 v's (`-vvvv`) to see the details of the installation process. However it is important to note that this can significantly increase installation time, so it is recommended that you use it only as needed or requested by Red Hat support.

Upgrade your Operator-based deployment of Ansible Automation Platform

The Ansible Automation Platform Operator simplifies the installation, upgrade, and deployment of new Red Hat Ansible Automation Platform instances in your OpenShift Container Platform environment.

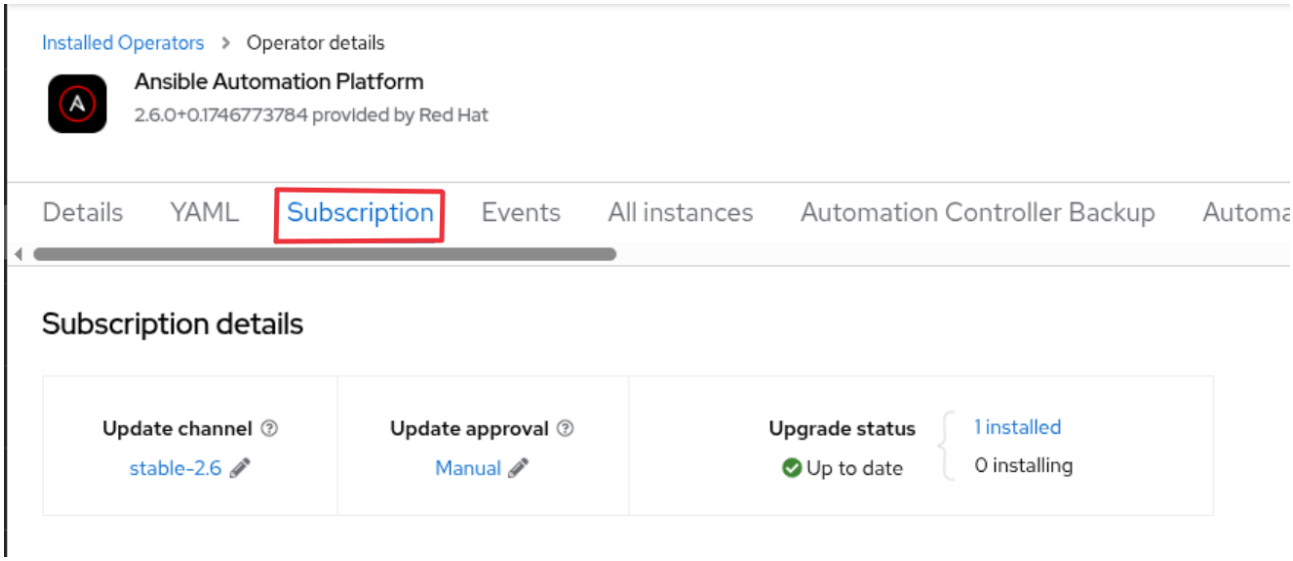
Overview

You can use this document for help with upgrading Ansible Automation Platform on Red Hat OpenShift Container Platform. This includes upgrades from supported previous versions and patch updates within your current version.

The Ansible Automation Platform Operator manages deployments, upgrades, backups, and restores of automation controller and automation hub. It also handles deployments of AnsibleJob and JobTemplate resources from the Ansible Automation Platform Resource Operator.

Each operator version has default automation controller and automation hub versions. When the operator is upgraded, it also upgrades the automation controller and automation hub deployments it manages, unless overridden in the spec.

OpenShift deployments of Ansible Automation Platform use the built-in Operator Lifecycle Management (OLM) functionality. For more information, see **Operator Lifecycle Manager concepts and resources**. OpenShift does this by using Subscription, CSV, InstallPlan, and OperatorGroup objects. Most users will not have to interact directly with these resources. They are created when the Ansible Automation Platform Operator is installed from **OperatorHub** and managed through the **Subscriptions** tab in the OpenShift console UI. For more information, refer to **Accessing the web console**.



Related information
Operator Lifecycle Manager concepts and resources
Accessing the web console

Upgrade considerations

If you are upgrading from version 2.4, continue to the **Upgrading the Ansible Automation Platform Operator**.

If your OpenShift Container Platform version is not supported by the Red Hat Ansible Automation Platform version you are upgrading to, you must upgrade your OpenShift Container Platform cluster to a supported version first.

Refer to the **Red Hat Ansible Automation Platform Life Cycle** to determine the OpenShift Container Platform version needed.

For information about upgrading your cluster, refer to **Updating clusters**.

Related information
Upgrade your Operator-based deployment of Ansible Automation Platform
Red Hat Ansible Automation Platform Life Cycle
Updating clusters

Prerequisites and channel upgrades

To upgrade to a newer version of Ansible Automation Platform Operator, you must:

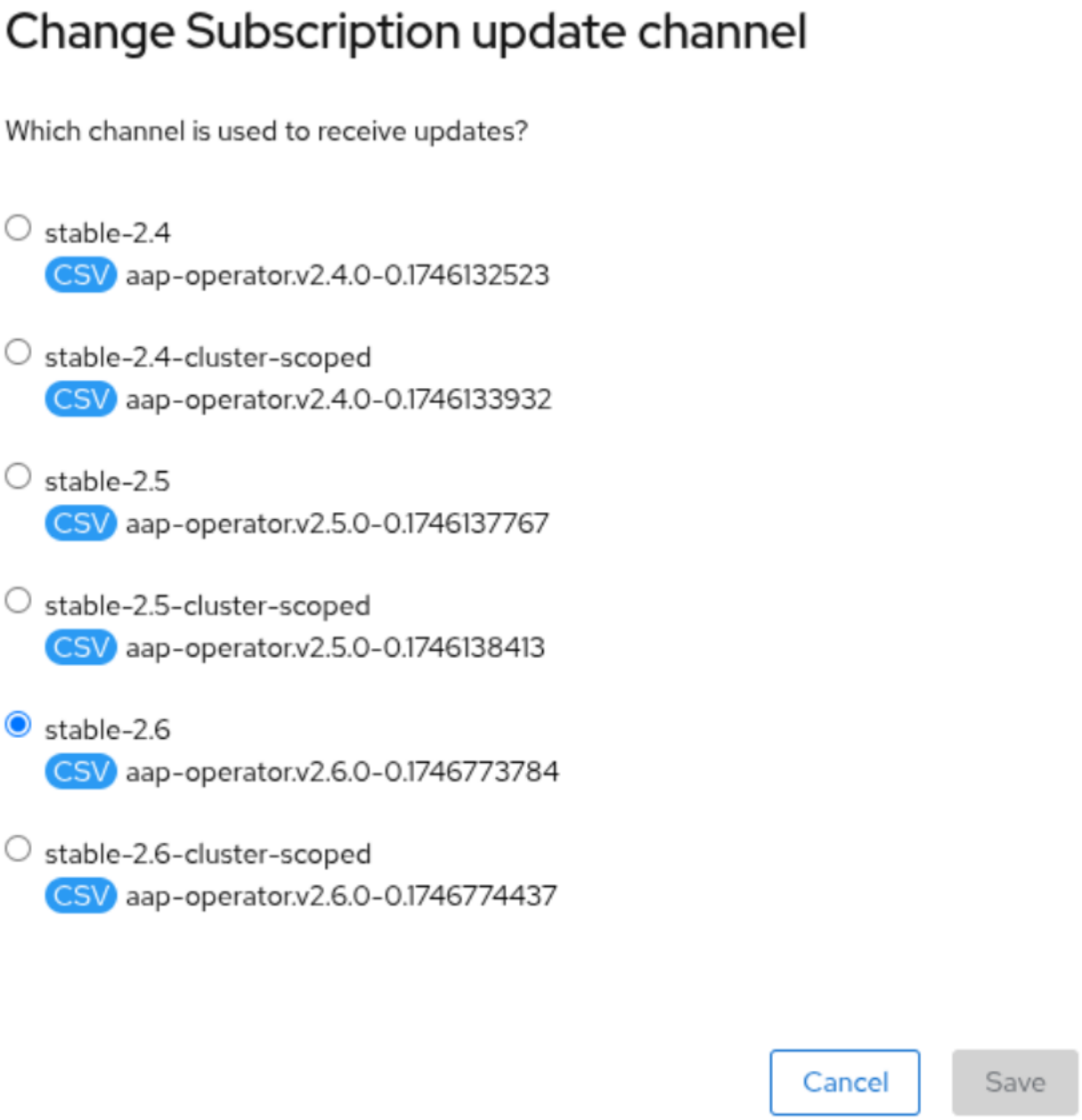
Ensure your system meets the system requirements detailed in the **Operator topologies** section of the *Tested deployment models* .

- Create **AutomationControllerBackup** and **AutomationHubBackup** objects. For help with this see **Backup and recovery for operator environments**
- Review the **Release notes** for the new Ansible Automation Platform version to which you are upgrading and any intermediate versions.
- Determine the type of upgrade you want to perform. See the **Channel Upgrades** section for more information.

Related information
Operator topologies
Backup and restore in an OpenShift environment
Release notes
Channel upgrades

Channel upgrades

Upgrading Ansible Automation Platform involves retrieving updates from a channel. A channel refers to a location where you can access your update from the OpenShift console UI.



In-channel upgrades

Most upgrades occur within a channel as follows:

1. A new update becomes available in the marketplace, through the redhat-operator CatalogSource.
2. The system automatically creates a new InstallPlan for your Ansible Automation Platform subscription.
    - If set to **Manual**, the InstallPlan needs manual approval in the OpenShift UI.
    - If set to **Automatic**, it upgrades as soon as the new version is available.

NOTE:
Set a manual install strategy on your Ansible Automation Platform Operator subscription during installation or upgrade. You will be prompted to approve upgrades when available in your chosen update channel. A stable channel is available for each X.Y release, following the naming pattern `stable-X.Y`.

3. A new subscription, CSV, and operator containers are created alongside the old ones. The old resources are cleaned up after a successful install.

Cross-channel upgrades

Upgrading between X.Y channels is always manual and intentional. Stable channels for major and minor versions are in the Operator Catalog. It is recommended to stay on the latest minor version channel for the latest patches.

If the subscription is set for manual upgrades, you must approve the upgrade in the UI. Then, the system upgrades the Operator to the latest version in that channel.

The containers provided in the latest channel are updated regularly for OS upgrades and critical fixes. This allows customers to receive critical patches and CVE fixes faster. Larger changes and new features are saved for minor and major releases.

For each major or minor version channel, there is a corresponding "cluster-scoped" channel available. Cluster-scoped channels deploy operators that can manage all namespaces, while non-cluster-scoped channels can only manage resources in their own namespace.

IMPORTANT:
Cluster-scoped bundles are not compatible with namespace-scoped bundles. Do not switch between namespace-scoped (`stable-X.Y`) channels and cluster-scoped (`stable-X.Y-cluster-scoped`) channels. This is not supported.

Upgrade the Ansible Automation Platform Operator

To upgrade to the latest version of Ansible Automation Platform Operator on OpenShift Container Platform, you can use the following procedure:

Before you begin

- Read the Release notes for your target version.

1. For existing deployments only: You must deploy your automation controller and automation hub instances to the same, single namespace before upgrading. For more information see, Migrating from one namespace to another.
2. Review the Backup and restore in an OpenShift environment section and backup your services:
    - AutomationControllerBackup
    - AutomationHubBackup
    - EDABackup

IMPORTANT:
Upgrading from Event-Driven Ansible 2.4 is not supported. If you are using Event-Driven Ansible 2.4 in production, contact Red Hat before you upgrade.

Procedure

1. Log in to OpenShift Container Platform.
2. Navigate to **Operators › Installed Operators**.
3. Select the Ansible Automation Platform Operator installed on your project namespace.
4. Select the **Subscriptions** tab.
5. Change the channel to the stable channel for your target version (for example, `stable-2.6`).
6. This creates an InstallPlan for the user. Click **Preview InstallPlan**.
7. Click **Approve**.
8. Create a Custom Resource (CR) using the Ansible Automation Platform UI.

NOTE:
The automation controller and automation hub UIs remain until all SSO configuration is supported in the platform gateway UI.

Result

You can confirm you have upgraded successfully by navigating to **Operators › Installed Operators**, here under Ansible Automation Platform you can verify the version number matches your target version.

Additionally, go to your Ansible Automation Platform Operator deployment and click **All instances** to verify if all instances upgraded correctly. All pods should display either a **Running** or **Completed status**, with no pods displaying an error status.

Create Ansible Automation Platform custom resources

After upgrading to the latest version of Ansible Automation Platform Operator on OpenShift Container Platform, you can create an Ansible Automation Platform custom resource (CR) that specifies the names of your existing deployments, in the same namespace.

The following example outlines the steps to deploy a new Event-Driven Ansible setup after upgrading to the latest version, with existing automation controller and automation hub deployments already in place.

The Appendix contains more examples of Ansible Automation Platform CRs for different deployments.

Procedure

1. Log in to Red Hat OpenShift Container Platform.
2. Navigate to **Operators › Installed Operators**.
3. Select your Ansible Automation Platform Operator deployment.
4. Select the **Details** tab.
5. On the **Ansible Automation Platform** tile click **Create instance**.
6. From the **Create Ansible Automation Platform** page enter a name for your instance in the **Name** field.
7. Click **YAML view** and paste the following YAML (aap-existing-controller-and-hub-new-eda.yml):

```
---
apiVersion: aap.ansible.com/v1alpha1
kind: AnsibleAutomationPlatform
metadata:
  name: myaap
spec:
  # Development purposes only
  no_log: false

  controller:
    name: existing-controller #obtain name from controller CR
    disabled: false

  eda:
    disabled: false

  hub:
    name: existing-hub
    disabled: false
```

8. Click **Create**.

> **NOTE:**
> You can override the operator's default image for automation controller, automation hub, or platform-resource app images by specifying the preferred image on the YAML spec. This enables upgrading a specific deployment, like a controller, without updating the operator.
>
> The recommended approach however, is to upgrade the operator and use the default image values.

Result

Navigate to your Ansible Automation Platform Operator deployment and click **All instances** to verify whether all instances have deployed correctly. You should see the **Ansible Automation Platform** instance and the deployed **AutomationController**, **EDA**, and **AutomationHub** instances here.

Alternatively, you can verify whether all instances deployed correctly by running `oc get route` in the command line.

Patch update for Operator-based Ansible Automation Platform

You can use an upgrade patch to update your operator-based Ansible Automation Platform.

Patch updating Ansible Automation Platform on OpenShift Container Platform

When you perform a patch update for an installation of Ansible Automation Platform on OpenShift Container Platform, most updates happen within a channel:

1. A new update becomes available in the marketplace (through the redhat-operator CatalogSource).
2. A new InstallPlan is automatically created for your Ansible Automation Platform subscription. If the subscription is set to Manual, the InstallPlan must be manually approved in the OpenShift UI. If the subscription is set to Automatic, it upgrades as soon as the new version is available.

   NOTE:
   It is recommended that you set a manual install strategy on your Ansible Automation Platform Operator subscription (set when installing or upgrading the Operator) and you will be prompted to approve an upgrade when it becomes available in your selected update channel. Stable channels for each X.Y release (for example, stable-2.5) are available.

3. A new Subscription, CSV, and Operator containers will be created alongside the old Subscription, CSV, and containers. Then the old resources will be cleaned up if the new install was successful.

Upgrade your RPM deployment of Ansible Automation Platform

You can upgrade your Ansible Automation Platform installation using the supported upgrade paths. Review the available upgrade paths and required steps to ensure a successful upgrade of your Ansible Automation Platform environment.

Before beginning your upgrade, review the prerequisites and upgrade planning sections of this guide.

Ansible Automation Platform supports upgrades from the two most recent minor releases. Check the release notes for your target version to confirm which upgrade paths are supported.

Supported upgrade path	Steps to upgrade
From the immediately previous minor release	1. Back up your Ansible Automation Platform instance. 2. Download the installation package for your target version. 3. Set up your inventory file to match your installation environment. 4. Run the installation program over your current Ansible Automation Platform instance.
From two minor releases back	1. Back up your Ansible Automation Platform instance. 2. Download the installation package for your target version. 3. Set up your inventory file to match your installation environment. 4. Check the release notes for any component-specific upgrade restrictions. Some components might require additional steps, such as removing a database before upgrading. 5. Run the installation program over your current Ansible Automation Platform instance.
Mixed-version environments	If your environment includes components at different minor versions, upgrade the older components first so that all services reach the same target version. Use the inventory file to specify only the components you are upgrading in each pass. After all components are at the same version, run a final upgrade on all services together.

IMPORTANT:
Not all components support upgrades that skip a minor release. Check the release notes for your target version to identify any components that require removal and reinstallation rather than a direct upgrade.

Related information
Back up your Ansible Automation Platform instance
Set up the inventory file
Tested deployment models
Run the installer setup script and verify platform accounts

Upgrade Ansible Automation Platform

To upgrade your deployment of Ansible Automation Platform, review the planning requirements to ensure a successful upgrade. You can then download the desired version of the installation program, configure the inventory file to reflect your environment, and then run the installation program.

IMPORTANT:
The automation controller API remains available for backward compatibility; however, you must use the platform gateway API for managing organizations, teams, and users. Using the legacy API introduces a delay of up to 15 minutes before changes are synchronized to all components, including Event-Driven Ansible controller.

System requirements

Before you begin the upgrade process, review the following considerations to plan and prepare your Ansible Automation Platform deployment.

WARNING: To upgrade to Ansible Automation Platform 2.7, you must be running a containerized or OpenShift Container Platform deployment. RPM-based deployments are not supported as an upgrade path to 2.7. If you are running an RPM-based deployment, migrate to a containerized or OpenShift Container Platform deployment before you upgrade.

Ansible Automation Platform requirements

- Verify that you have a valid subscription before upgrading from a previous version of Ansible Automation Platform. Existing subscriptions are carried over during the upgrade process.

- Review *Plan your upgrade to Ansible Automation Platform 2.6* to understand the upgrade requirements and scenarios, and *Choose a deployment method and topology* for the RPM topologies and infrastructure.
    - Inspect all existing SAML authenticators in your automation controller environment before upgrading from Ansible Automation Platform 2.4 to 2.6. Encrypted private keys for SAML configurations are not supported in Ansible Automation Platform 2.6.
- Ensure that you are on Ansible Automation Platform 2.4 or 2.5 before upgrading to 2.6. You can only upgrade from Ansible Automation Platform 2.4 or 2.5 to 2.6.
- Upgrade to the latest version of Ansible Automation Platform 2.4 or 2.5 before upgrading to Red Hat Ansible Automation Platform 2.6.

IMPORTANT:

- When upgrading from Ansible Automation Platform 2.4 to 2.6, the API endpoints for the automation controller, automation hub, and Event-Driven Ansible controller are all available for use. These APIs are being deprecated and will be disabled in an upcoming release. This grace period is to allow for migration to the new APIs put in place with the platform gateway.
- If you upgraded from Ansible Automation Platform 2.4 to 2.5, you must migrate your authentication methods and users before upgrading to 2.6 as that legacy authenticator functionality was removed.

- Back up your Ansible Automation Platform environment before upgrading in case any issues occur.
- Capture your inventory or instance group details before upgrading.
- Review the platform gateway requirements:
    - Ansible Automation Platform 2.4 to 2.6 upgrades include the platform gateway. Ensure you review the 2.6 Network ports and protocols diagram for architectural changes.
    - Platform gateway has a number of associated inventory file variables, some of which are required.
    - When upgrading from Ansible Automation Platform 2.4 to 2.6, connections to the platform gateway URL might fail on the platform gateway UI if you are using the automation controller behind a load balancer. The following error message is displayed: `Error connecting to Controller API`
      To resolve this issue, for each automation controller host, add the platform gateway URL as a trusted source in the `CSRF_TRUSTED_ORIGIN` setting in the **settings.py** file for each automation controller host. You must then restart each automation controller host so that the URL changes are implemented.
- Review the centralized redis instance offered by Ansible Automation Platform for both standalone and clustered topologies.

    - Ansible Automation Platform 2.6 offers a centralized Redis instance in both standalone and clustered topologies.
    - 6 VMs are required for a Redis high availability (HA) compatible deployment. Redis can be colocated on each Ansible Automation Platform component VM except for automation controller, execution nodes, or the PostgreSQL database.
    - External Redis is not supported for RPM-based deployments of Ansible Automation Platform.

- Limitation:
    - Upgrade of Event-Driven Ansible 2.5 to 2.6 is supported, but upgrade from Event-Driven Ansible 2.4 to 2.6 is not supported. Database migrations between Event-Driven Ansible 2.4 and Event-Driven Ansible 2.6 are not compatible. If you are upgrading from Ansible Automation Platform 2.4 to 2.6 and you deployed Event-Driven Ansible, you must first remove the Event-Driven Ansible 2.4 database and then upgrade your platform to 2.6.

System requirements

Type	Description	Notes
Subscription	Valid Red Hat Ansible Automation Platform subscription	
Operating system	Red Hat Enterprise Linux 9.4 or later minor versions of Red Hat Enterprise Linux 9	Red Hat Ansible Automation Platform are also supported on OpenShift.
CPU architecture	x86_64, AArch64, s390x (IBM Z), ppc64le (IBM Power)	
Ansible-core	Ansible-core version 2.16 or later	Ansible Automation Platform uses the system-wide ansible-core package to install the platform, but uses ansible-core 2.16 for both its control plane and built-in execution environments.
Browser	A currently supported version of Mozilla Firefox or Google Chrome.	
Database	• For Ansible Automation Platform managed databases: PostgreSQL 15. • For customer provided (external) databases: PostgreSQL 15, 16, or 17.	• External (customer supported) databases require International Components for Unicode (ICU) support.

Type	Description	Notes
		• External databases using PostgreSQL 16 or 17 must rely on external backup and restore processes. Backup and restore functionality is dependent on utilities provided with PostgreSQL 15.

Database requirements

- Ansible Automation Platform can work with two varieties of database:
  - Database installed with Ansible Automation Platform - This database consists of a PostgreSQL installation done as part of an Ansible Automation Platform installation using PostgreSQL packages provided by Red Hat.
  - Customer provided or configured database - This is an external database that is provided by the customer, whether on bare metal, virtual machine, container, or cloud hosted service. Ansible Automation Platform requires customer provided (external) database to have ICU support.
- PostgreSQL user passwords are hashed with SCRAM-SHA-256 secure hashing algorithm before storing in the database.
- Ensure that you back up your Ansible Automation Platform environment before upgrading in case any issues occur.

User privileges

- Ensure a dedicated non-root user is configured on the Red Hat Enterprise Linux host.
  - This user requires sudo or other Ansible supported privilege escalation (sudo is recommended) to perform administrative tasks during the installation.
  - This user is responsible for the installation of RPM Ansible Automation Platform.
  - You can obtain root access either through the sudo command, or through privilege escalation. You can de-escalate privileges from root to users such as AWX, PostgreSQL, Event-Driven Ansible, or Pulp.
  - An NTP client is configured on each node.

Related information
Plan your upgrade to Ansible Automation Platform 2.6
Choose a deployment method and topology
Authentication type: SAML

Back up your Ansible Automation Platform instance

Back up an Ansible Automation Platform instance by running the `setup.sh` script with the `backup_dest` flag. You can also enable the compression flags `use_archive_compression` and `use_db_compression` to reduce the size of the backup artifacts.

Procedure

1. Navigate to your Ansible Automation Platform installation directory.
2. Run the `./setup.sh` script following the example below:

```
$ ./setup.sh -e 'backup_dest=/ansible/mybackup' -e
'use_archive_compression=true' 'use_db_compression=true @credentials.yml -b
```

Where:

- `backup_dest`: Specifies a directory to save your backup to.
- `use_archive_compression=true` and `use_db_compression=true`: Compresses the backup artifacts before they are sent to the host running the backup operation.
  You can use the following variables to customize the compression:
  - For global control of compression for filesystem related backup files: `use_archive_compression=true`
  - For component-level control of compression for filesystem related backup files: `<componentName>_use_archive_compression`
    For example:
    - `automationgateway_use_archive_compression=true`

- `automationcontroller_use_archive_compression=true`
- `automationhub_use_archive_compression=true`
- `automationedacontroller_use_archive_compression=true`

- For global control of compression for database related backup files: `use_db_compression=true`
- For component-level control of compression for database related backup files: `<componentName>_use_db_compression=true`
  For example:
  - `automationgateway_use_db_compression=true`
  - `automationcontroller_use_db_compression=true`
  - `automationhub_use_db_compression=true`
  - `automationedacontroller_use_db_compression=true`

Result

After a successful backup, a backup file is created at `/ansible/mybackup/automation-platform-backup-<date/time>.tar.gz`.

Install with internet access

Choose the Red Hat Ansible Automation Platform installation program if your Red Hat Enterprise Linux environment is connected to the internet. Installing with internet access retrieves the latest required repositories, packages, and dependencies.

Choose one of the following ways to set up your Ansible Automation Platform installation program.

- **Tarball install**
  a. Navigate to the Red Hat Ansible Automation Platform download page.
  b. In the **Product software** tab, click **Download Now** for the **Ansible Automation Platform <latest-version> Setup**.
  c. Extract the files:

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

- **RPM install**
  a. Install the Ansible Automation Platform Installer Package.
  v.2.6 for RHEL 9 for x86-64:

```
$ sudo dnf install --enablerepo=ansible-automation-platform-2.6-for-
rhel-9-x86_64-rpms ansible-automation-platform-installer
```

NOTE:
`dnf install` enables the repo as the repo is disabled by default.

When you use the RPM installer, the files are placed under the `/opt/ansible-automation-platform/installer` directory.

Install without internet access

Use the Ansible Automation Platform Bundle installation program for offline environments or to avoid installing dependencies from online repositories. RHEL repository access is still required; all other data is in the archive.

Procedure

1. Navigate to the Red Hat Ansible Automation Platform download page.
2. In the **Product software** tab, click **Download Now** for the **Ansible Automation Platform <latest-version> Setup Bundle**.
3. Extract the files:

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

Set up the inventory file

Before upgrading your Red Hat Ansible Automation Platform installation, edit the `inventory` file to match your required configuration. You can keep the same parameters from your existing deployment or you can update the parameters to match any changes to your environment.

You can find sample inventory files in the Test topologies GitHub repository, or in Choose a deployment method and topology.

Procedure

1. Navigate to the installation program directory.

Bundled installer

```
$ cd ansible-automation-platform-setup-bundle-2.6-4-x86_64
```

Online installer

```
$ cd ansible-automation-platform-setup-2.6-4
```

2. Open the `inventory` file for editing.
3. Modify the `inventory` file to provision new nodes, deprovision nodes or groups, and import or generate automation hub API tokens.

   You can use the same `inventory` file from an existing Ansible Automation Platform installation if there are no changes to the environment.

> **NOTE:**
> Provide a reachable IP address or fully qualified domain name (FQDN) for all hosts to ensure that users can synchronize and install content from Ansible automation hub from a different node. Do not use `localhost`. If `localhost` is used, the upgrade will be stopped as part of preflight checks.

4. Provision new nodes in a cluster, by adding new nodes alongside existing nodes in the `inventory` file as follows:

```
[automationcontroller]
clusternode1.example.com
clusternode2.example.com
clusternode3.example.com

[all:vars]
admin_password='password'

pg_host='<host_name>'

pg_database='<database_name>'
pg_username='<your_username>'
pg_password='<your_password>'
```

Remove the 2.4 database for Event-Driven Ansible

Ansible Automation Platform 2.6 supports upgrades from Ansible Automation Platform 2.4 environments for all components, except for Event-Driven Ansible. Database migrations between Event-Driven Ansible 2.4 and Event-Driven Ansible 2.6 are not compatible.

If you are upgrading from Ansible Automation Platform 2.4 to 2.6, you must first remove the Event-Driven Ansible 2.4 database. A new Event-Driven Ansible 2.6 database gets created automatically after the upgrade. You can then reconnect Automation Decisions (Event-Driven Ansible controller) to Automation Execution (automation controller) to run rulebook activations.

NOTE:
When upgrading from Ansible Automation Platform 2.5 to 2.6, the Event-driven Ansible component will be updated automatically. You are not required to delete the existing Event-Driven Ansible 2.5 database before upgrading your platform to 2.6.

Procedure

1. Shut down the old Event-Driven Ansible 2.4 host.
2. Log in to your database host with a user that has superuser privileges.

   ```
   # psql -h <hostname> -U <username>
   ```
3. When prompted, enter your password.
4. Delete the existing Event-Driven Ansible 2.4 database by using the following command:

   ```
   DROP DATABASE automationedacontroller
   ```
5. When prompted, reenter your password.

Next steps

1. Run the Ansible Automation Platform installer setup script.
2. After the upgrade is completed, reconnect Automation Decisions (Event-Driven Ansible controller) to Automation Execution (automation controller) to run rulebook activations successfully.

Run the installer setup script and verify platform accounts

Execute the Red Hat Ansible Automation Platform installer setup script after configuring your `inventory` file. This action initiates the installation or update process, applying all your custom settings to deploy the platform successfully.

- Run the `setup.sh` script:

```
$ ./setup.sh
```

Result

The installation will begin.

Verify user migration

During the upgrade to Ansible Automation Platform 2.6, controller user accounts are converted into platform user accounts. Controller administrators retain their administrative privileges, but they are converted into platform administrator privileges.

Other controller accounts become platform users, and their existing permissions are mapped over appropriately after an initial password reset. Users with existing accounts associated with other components (like private automation hub and Event-Driven Ansible) must have their passwords reset by their administrator before they can log in.

Authenticator configurations are automatically migrated. SSO and LDAP accounts do not require any manual migration steps, including password resets, with the exception of accounts that use a certificate from the trust store. See Configuring authentication in the Ansible Automation Platform for more information on migrating authentication configurations that use a custom certificate.

After your upgrade to Ansible Automation Platform 2.6 is complete, verify that you can log in to the upgraded platform.

- If you have a controller account that has been converted to a platform gateway account for Ansible Automation Platform 2.6:
    - Log into your upgraded platform instance with your controller credentials.
- If you have a component-level account (such as an account associated with private automation hub or Event-Driven Ansible):
    - Request a password reset from your administrator and log into the upgraded platform with your new password.

Mixed-version upgrades with pre-gateway components

Ansible Automation Platform supports upgrades from pre-gateway environments for all components except Event-Driven Ansible. You can configure a mixed environment with Event-Driven Ansible and the platform gateway connected to an existing pre-gateway cluster running automation controller and automation hub.

Combining installation methods (OpenShift Container Platform, RPM, containerized) within such a topology is not supported.

NOTE:
If you are running a pre-gateway version of Event-Driven Ansible in production, contact Red Hat support or your account representative before you upgrade for guidance on migrating to the current version.

Supported topologies described in this document assume that:

- Pre-gateway services include only automation controller and automation hub.
- Current-version services include Event-Driven Ansible and the platform gateway.
- Combining installation methods for these topologies is not supported.

Upgrade considerations

- You must have two inventory files as a starting point: one for the pre-gateway services and one for the current-version services.
- Before running the upgrade, you must merge the pre-gateway inventory into the current-version inventory. The platform gateway host must be included in the inventory for the installation program to run successfully.
- Run the upgrade on the merged current-version inventory file only.
- You must be using an external database for both inventories.
- If you are using managed database instances for either inventory, you must migrate to an external database before upgrading.

Use migration path for 2.4 instances with managed databases

Migrate Ansible Automation Platform 2.4 instances with managed databases to 2.6 by upgrading automation controller and automation hub before enabling unified UI and Event-Driven Ansible.

Before you begin

- An inventory from 2.4 for automation controller and automation hub and a 2.6 inventory for unified UI (platform gateway) and Event-Driven Ansible. You must merge both inventories into a single 2.6 inventory file before running the upgrade. The platform gateway host must be included in the inventory for the installation program to run successfully.

IMPORTANT:
Ensure you have upgraded to the latest version of Ansible Automation Platform2.4 before merging inventories and running the 2.6 upgrade.

- **For standalone node managed database**
  - Convert the database node to an external one, removing it from the inventory. The PostgreSQL node will continue working and will not lose the Ansible Automation Platform-provided setup, but you are responsible for managing its configuration afterward.
- **For collocated managed database**
  a. Back up
  b. Restore with standalone managed database node instead of collocated
  c. Unmanaged standalone database

Use migration path for 2.4 services with 2.6 services

If you installed Ansible Automation Platform 2.6 to use Event-Driven Ansible in a supported scenario, you can upgrade your Ansible Automation Platform 2.4 automation controller and automation hub to Ansible Automation Platform 2.6 by following this procedure.

Before you begin

- An inventory from 2.4 for automation controller and automation hub and a 2.6 inventory for unified UI (platform gateway) and Event-Driven Ansible. You must merge both inventories

into a single 2.6 inventory file before running the upgrade. The platform gateway host must be included in the inventory for the installation program to run successfully.

IMPORTANT:
Ensure you have upgraded to the latest version of Ansible Automation Platform 2.4 before merging inventories and running the 2.6 upgrade.

Procedure

1. Merge 2.4 inventory data into the 2.6 inventory.

   The example below shows the inventory file for automation controller and automation hub for 2.4 and the inventory file for Event-Driven Ansible and the unified UI (platform gateway) for 2.6, respectively, as the starting point, and what the merged inventory looks like.

Inventory files from 2.4

```
[automationcontroller]
controller-1
controller-2

[automationhub]
hub-1
hub-2

[all:vars]
# Here we have the admin passwd, db credentials, etc.
```

Inventory files from 2.6

```
[automationedacontroller]
eda-1
eda-2

[automationgateway]
gw-1
gw-2

[all:vars]
# Here we have admin passwd, db credentials etc.
```

Merged Inventory

```
[automationcontroller]
controller-1
controller-2

[automationhub]
hub-1
hub-2

[automationedacontroller]
eda-1
eda-2

[automationgateway]
gw-1
gw-2

[all:vars]
# Here we have admin passwd, db credentials etc from both inventories above
```

2. Run `setup.sh`

   The installation program upgrades all services to the latest version of Ansible Automation Platform 2.6 and connects them to the unified UI (platform gateway).

Result

- Verify that everything has upgraded to 2.6 and is working properly in one of two ways:
  - Perform an SSH to automation controller and Event-Driven Ansible.
  - In the unified UI, go to **Help > About** to verify the RPM versions are at 2.6.

Upgrade additional services for Ansible Automation Platform

Upgrading services that extend Ansible Automation Platform keeps your environment current with the latest features and bug fixes. Update these services to maintain compatibility with the platform and access new features as they become available.

Upgrading services that extend Ansible Automation Platform helps you to:

- **Access latest improvements**: Apply new features and bug fixes to your deployed services.

- **Support development teams**: Provide current tools and integrations for content development workflows.
- **Ensure continued compatibility**: Keep extended services synchronized with Ansible Automation Platform to maintain proper integration and functionality.

Upgrade the Ansible plug-ins with Helm

To upgrade the Ansible plug-ins, you must update the `plugin-registry` application with the latest Ansible plug-ins files.

Download the Ansible plug-ins files

Download the Ansible plug-ins for Red Hat Developer Hub **Setup Bundle** from the Red Hat Ansible Automation Platform Product Software downloads page.

Procedure

1. In a browser, navigate to the Red Hat Ansible Automation Platform Product Software downloads page and select the **Product Software** tab.
2. Click **Download now** next to **Ansible plug-ins for Red Hat Developer Hub Setup Bundle** to download the latest version of the plug-ins.

   The format of the filename is `ansible-rhdh-plugins-x.y.z.tar.gz`. Substitute the Ansible plug-ins release version, for example `2.0.0`, for `x.y.z`.
3. Create a directory on your local machine to store the `.tar` files.

```
$ mkdir /path/to/<ansible-backstage-plugins-local-dir-changeme>
```

4. Set an environment variable (`$DYNAMIC_PLUGIN_ROOT_DIR`) to represent the directory path.

```
$ export DYNAMIC_PLUGIN_ROOT_DIR=/path/to/<ansible-backstage-plugins-local-dir-changeme>
```

5. Extract the `ansible-rhdh-plugins-<version-number>.tar.gz` contents to `$DYNAMIC_PLUGIN_ROOT_DIR`.

```
$ tar --exclude='*code*' -xzf ansible-rhdh-plugins-x.y.z.tar.gz -C $DYNAMIC_PLUGIN_ROOT_DIR
```

Substitute the Ansible plug-ins release version, for example `2.0.0`, for `x.y.z`.

Result

Run `ls` to verify that the extracted files are in the `$DYNAMIC_PLUGIN_ROOT_DIR` directory:

```
$ ls $DYNAMIC_PLUGIN_ROOT_DIR
ansible-plugin-backstage-rhaap-dynamic-x.y.z.tgz
ansible-plugin-backstage-rhaap-dynamic-x.y.z.tgz.integrity
ansible-plugin-scaffolder-backend-module-backstage-rhaap-dynamic-x.y.z.tgz
ansible-plugin-scaffolder-backend-module-backstage-rhaap-dynamic-x.y.z.tgz.integrity
```

The files with the `.integrity` file type contain the plugin SHA value. The SHA value is used during the plug-in configuration.

Update the plug-in registry

Rebuild your plug-in registry application in your OpenShift cluster with the latest Ansible plug-ins files.

Before you begin

- You have downloaded the Ansible plug-ins files.
- You have set an environment variable, for example `$DYNAMIC_PLUGIN_ROOT_DIR`, to represent the path to the local directory where you have stored the `.tar` files.

Procedure

1. Log in to your OpenShift Container Platform instance with credentials to create a new application.
2. Open your Red Hat Developer Hub OpenShift project.

```
$ oc project <YOUR_DEVELOPER_HUB_PROJECT>
```

3. Run the following commands to update your plug-in registry build in the OpenShift cluster. The commands assume that `$DYNAMIC_PLUGIN_ROOT_DIR` represents the directory for your `.tar` files. Replace this in the command if you have chosen a different environment variable name.

```
$ oc start-build plugin-registry --from-dir=$DYNAMIC_PLUGIN_ROOT_DIR --wait
```

4. When the registry has started, the output displays the following message:

```
 Uploading directory "/path/to/dynamic_plugin_root" as binary input for the
build …

Uploading finished

build.build.openshift.io/plugin-registry-1 started
```

Result

Verify that the `plugin-registry` has been updated.

1. In the OpenShift UI, click **Topology**.
2. Click the **redhat-developer-hub** icon to view the pods for the plug-in registry.
3. Click **View logs** for the plug-in registry pod.
4. Open the **Terminal** tab and run `ls` to view the `.tar` files in the `plug-in registry`.
5. Verify that the new `.tar` file has been uploaded.

Update the Ansible plug-ins version numbers for a Helm installation

To upgrade the Ansible plug-ins, you must update the `imageTagInfo` parameter in the Helm chart configuration to the desired version. This triggers the Red Hat Developer Hub to pull the new container images directly from the Red Hat registry.

Procedure

1. Log in to your Red Hat OpenShift Container Platform instance.
2. In the OpenShift Developer UI, navigate to **Helm > developer-hub > Actions > Upgrade > Yaml view**.
3. Locate the `global` section.

```
...
global:
  # Ensure OCI mode is enabled
  pluginMode: oci

  # UPDATE this value to the new desired version
  imageTagInfo: "2.1"

  # Note: Do not manually update 'plugins' packages;
  # OCI mode handles the download automatically based on the tag above.
  dynamic:
    plugins: []
```

4. Click **Upgrade**.

   Step result:
   The Red Hat Developer Hub pods restart and pull the new plug-in versions.

Result

1. In the OpenShift UI, click **Topology**.
2. Make sure that the Red Hat Developer Hub instance is available.

Upgrade the Ansible plug-ins for an Operator environment

To upgrade the Ansible plug-ins, you must update the `plugin-registry` application with the latest Ansible plug-ins files.

Download the Ansible plug-ins files

Download the Ansible plug-ins for Red Hat Developer Hub **Setup Bundle** from the Red Hat Ansible Automation Platform Product Software downloads page.

Procedure

1. In a browser, navigate to the Red Hat Ansible Automation Platform Product Software downloads page and select the **Product Software** tab.
2. Click **Download now** next to **Ansible plug-ins for Red Hat Developer Hub Setup Bundle** to download the latest version of the plug-ins.

   The format of the filename is `ansible-rhdh-plugins-x.y.z.tar.gz`. Substitute the Ansible plug-ins release version, for example `2.0.0`, for `x.y.z`.
3. Create a directory on your local machine to store the `.tar` files.

```
$ mkdir /path/to/<ansible-backstage-plugins-local-dir-changeme>
```

4. Set an environment variable (`$DYNAMIC_PLUGIN_ROOT_DIR`) to represent the directory path.

```
$ export DYNAMIC_PLUGIN_ROOT_DIR=/path/to/<ansible-backstage-plugins-local-
dir-changeme>
```

5. Extract the `ansible-rhdh-plugins-<version-number>.tar.gz` contents to `$DYNAMIC_PLUGIN_ROOT_DIR`.

```
$ tar --exclude='*code*' -xzf ansible-rhdh-plugins-x.y.z.tar.gz -C
$DYNAMIC_PLUGIN_ROOT_DIR
```

   Substitute the Ansible plug-ins release version, for example `2.0.0`, for `x.y.z`.

Result

Run `ls` to verify that the extracted files are in the `$DYNAMIC_PLUGIN_ROOT_DIR` directory:

```
$ ls $DYNAMIC_PLUGIN_ROOT_DIR
ansible-plugin-backstage-rhaap-dynamic-x.y.z.tgz
ansible-plugin-backstage-rhaap-dynamic-x.y.z.tgz.integrity
ansible-plugin-scaffolder-backend-module-backstage-rhaap-dynamic-x.y.z.tgz
ansible-plugin-scaffolder-backend-module-backstage-rhaap-dynamic-x.y.z.tgz.integrity
```

The files with the `.integrity` file type contain the plugin SHA value. The SHA value is used during the plug-in configuration.

Update the plug-in registry

Rebuild your plug-in registry application in your OpenShift cluster with the latest Ansible plug-ins files.

Before you begin

- You have downloaded the Ansible plug-ins files.
- You have set an environment variable, for example `$DYNAMIC_PLUGIN_ROOT_DIR`, to represent the path to the local directory where you have stored the `.tar` files.

Procedure

1. Log in to your OpenShift Container Platform instance with credentials to create a new application.
2. Open your Red Hat Developer Hub OpenShift project.

```
$ oc project <YOUR_DEVELOPER_HUB_PROJECT>
```

3. Run the following commands to update your plug-in registry build in the OpenShift cluster. The commands assume that `$DYNAMIC_PLUGIN_ROOT_DIR` represents the directory for your `.tar` files. Replace this in the command if you have chosen a different environment variable name.

```
$ oc start-build plugin-registry --from-dir=$DYNAMIC_PLUGIN_ROOT_DIR --wait
```

4. When the registry has started, the output displays the following message:

```
 Uploading directory "/path/to/dynamic_plugin_root" as binary input for the
build …

Uploading finished

build.build.openshift.io/plugin-registry-1 started
```

Result

Verify that the `plugin-registry` has been updated.

1. In the OpenShift UI, click **Topology**.
2. Click the **redhat-developer-hub** icon to view the pods for the plug-in registry.
3. Click **View logs** for the plug-in registry pod.
4. Open the **Terminal** tab and run `ls` to view the `.tar` files in the `plug-in registry`.

5. Verify that the new `.tar` file has been uploaded.

Update the Ansible plug-ins version numbers for an Operator installation

To upgrade the Ansible plug-ins, you must edit the `rhaap-dynamic-plugins-config` ConfigMap to reference the new OCI image tag.

Procedure

1. Log in to your Red Hat OpenShift Container Platform instance.
2. Navigate to ConfigMaps and select the `rhaap-dynamic-plugins-config` map.
3. Select the YAML tab to edit the file.
4. In the `plugins` list, update the version tag at the end of the `package` URL for both the frontend and backend plugins.

```
kind: ConfigMap

apiVersion: v1

metadata:

  name: rhaap-dynamic-plugins-config

data:

  dynamic-plugins.yaml: |

    includes:

      - dynamic-plugins.default.yaml

    plugins:

      # FRONTEND PLUGIN

      - disabled: false

        # UPDATE the version tag at the end of the URL (e.g., :2.1)

        package: 'oci:registry.redhat.io/ansible-automation-platform/
automation-portal:2.1'

        pluginConfig:

          dynamicPlugins:

            frontend:

              ansible.plugin-backstage-rhaap:

                appIcons:

                  - importName: AnsibleLogo

                    name: AnsibleLogo

                dynamicRoutes:

                  - importName: AnsiblePage

                    menuItem:

                      icon: AnsibleLogo

                      text: Ansible

                    path: /ansible


      # BACKEND PLUGIN

      - disabled: false

        # UPDATE the version tag at the end of the URL (e.g., :2.1)

        package: 'oci:registry.redhat.io/ansible-automation-platform/
automation-portal:2.1'

        pluginConfig:

          dynamicPlugins:

            backend:

              ansible.plugin-scaffolder-backend-module-backstage-rhaap: null
```

5. Click **Save**.

   Step result:
   The Red Hat Developer Hub detects the configuration change and reload the plug-ins automatically.

Result

1. In the OpenShift UI, click **Topology**.
2. Make sure that the Red Hat Developer Hub instance is available.

Upgrade self-service automation portal

To ensure that your self-service automation portal deployment has the latest features and fixes, you must upgrade the plug-in registry and Helm chart to the latest versions.

Self-service automation portal version compatibility

When upgrading self-service automation portal, ensure version compatibility between the Helm chart, plug-in bundle, and Ansible Automation Platform version to avoid installation or upgrade failures.

Version components

A self-service automation portal deployment consists of three version-dependent components:

Helm chart version

The version of the self-service automation portal Helm chart deployed in OpenShift Container Platform. Example: `2.1.0`

Plug-in bundle version

The version of the Ansible plug-ins setup bundle downloaded from the Red Hat Customer Portal. Example: `self-service-automation-portal-plugins-2.1.0.tar.gz`

Ansible Automation Platform version

The version of Ansible Automation Platform that self-service automation portal connects to. Example: `2.6`

Version alignment requirements

For a successful self-service automation portal upgrade:

- The Helm chart version and plug-in bundle version must have matching major and minor versions. The patch version can differ, but the plug-in bundle patch version must be equal to or greater than the Helm chart patch version.
  Example: Plug-in bundle version `2.1.1` is compatible with Helm chart version `2.1.0`, but plug-in bundle version `2.0.0` is not compatible with Helm chart version `2.1.0`.
- The self-service automation portal version must be compatible with your Ansible Automation Platform version.
  See the Ansible Automation Platform Life Cycle for version compatibility information.

Common version mismatch scenarios

The following scenarios can cause upgrade failures:

Plug-in bundle and Helm chart mismatch

If you download plug-in bundle version `2.0.0` but upgrade to Helm chart version `2.1.0`, the installation fails because the major.minor versions do not match. Similarly, if the plug-in bundle patch version is lower than the Helm chart patch version, you might encounter compatibility issues.

Stale plug-in bundle

If you download a plug-in bundle, and a new version is released before you complete the installation, you might install an outdated bundle with a newer Helm chart. This causes version mismatch errors during deployment.

Best practices for version management

To avoid version mismatch issues during upgrades:

- Identify your target version before starting the upgrade process. Check the Red Hat Ansible Automation Platform Product Software downloads page for the latest available versions.
- Download the plug-in bundle and upgrade the Helm chart in the same maintenance window to minimize the risk of version drift between download and installation.
- Verify the plug-in bundle version before extracting it. Check that the filename major.minor version (for example, `2.1` in `self-service-automation-portal-plugins-2.1.1.tar.gz`) matches your target Helm chart major.minor version. Ensure that the plug-in bundle patch version is equal to or greater than the Helm chart patch version.
- Keep a record of your current self-service automation portal version. Document the versions of all three components (Helm chart, plug-ins, Ansible Automation Platform) to simplify future upgrades and troubleshooting.

Related information
Ansible Automation Platform Life Cycle
Red Hat Ansible Automation Platform Product Software downloads page

Download the plug-in TAR files

Download the latest `.tar.gz` plug-in files for self-service automation portal from the Red Hat Customer Portal.

Procedure

1. Create a directory on your local machine to store the files.

```
$ mkdir /path/to/<automation-portal-plugins>
```

2. Set an environment variable (`$DYNAMIC_PLUGIN_ROOT_DIR`) to represent the directory path.

```
$ export DYNAMIC_PLUGIN_ROOT_DIR=/path/to/<automation-portal-plugins>
```

3. Download the setup bundle. In a browser, navigate to the Red Hat Ansible Automation Platform Product Software downloads page. and select the **Product Software** tab.
4. Click **Download now** next to **Ansible self-service automation portal Setup Bundle** to download the latest version of the plug-ins.

   The format of the filename is `self-service-automation-portal-plugins-x.y.z.tar.gz` .

   Substitute the Ansible plug-ins release version, for example `2.0.0` , for `x.y.z` .
5. Extract the `self-service-automation-portal-plugins-<version-number>.tar.gz` contents to `$DYNAMIC_PLUGIN_ROOT_DIR` .

```
$ tar --exclude='*code*' -xzf self-service-automation-portal-plugins-
x.y.z.tar.gz -C $DYNAMIC_PLUGIN_ROOT_DIR
```

   Substitute the Ansible plug-ins release version, for example `2.0.0` , for `x.y.z` .

Result

Run `ls` to verify that the extracted files are in the `$DYNAMIC_PLUGIN_ROOT_DIR` directory:

```
$ ls $DYNAMIC_PLUGIN_ROOT_DIR
ansible-plugin-backstage-rhaap-dynamic-x.y.z.tgz
ansible-plugin-backstage-rhaap-dynamic-x.y.z.tgz.integrity
ansible-plugin-scaffolder-backend-module-backstage-rhaap-dynamic-x.y.z.tgz
ansible-plugin-scaffolder-backend-module-backstage-rhaap-dynamic-x.y.z.tgz.integrity
```

The files with the `.integrity` file type contain the plugin SHA value.

Update the plug-in registry

To update the plug-in registry, you must upload your plug-in files to OpenShift, and start a new build of the registry.

Before you begin

- You have downloaded the plug-in TAR files for self-service automation portal.
- You have set an environment variable, for example `$DYNAMIC_PLUGIN_ROOT_DIR`, to represent the path to the local directory where you have stored the TAR files.

Procedure

1. In a terminal, log in to your OpenShift Container Platform instance.
2. Open your OpenShift project for self-service automation portal.

```
$ oc project <YOUR_SELF_SERVICE_AUTOMATION_PORTAL_PROJECT>
```

3. Find the name of your current plug-in registry build configuration:

```
$ oc get buildconfig
```

4. From the output, identify the correct build configuration name, for example `aap-self-service-plugins`.
5. Run the following command to start a new build in in your OpenShift project.

```
$ oc start-build <build_config_name> --from-dir=$DYNAMIC_PLUGIN_ROOT_DIR --
wait
```

- The command assumes that `$DYNAMIC_PLUGIN_ROOT_DIR` represents the directory for your TAR files. Replace this in the command if you have chosen a different environment variable name.
- Replace `<build_config_name>` with the build configuration name you identified.

When the build starts, the following message is displayed:

```
 Uploading directory "/path/to/dynamic_plugin_root" as binary input for the
build ...

 Uploading finished
```

Result

1. Open the **Topology** view in the **Developer** perspective for your project in the OpenShift web console.
2. Select the plugin registry icon to open the **plugin-registry** details pane.
3. In the **Pods** section of the **plugin-registry** details pane, select **View logs** for the new build pod. The format for the pod name is `<build_config_name>-<build_number>-build`.
4. Click the **terminal** tab and log in to the container.
5. In the terminal, run `ls` to view the TAR files in the plugin registry.
6. Verify that the new TAR files have been uploaded.

Update the self-service automation portal version numbers for a Helm installation

After you have updated your plug-in registry for your self-service automation portal project on your OpenShift Container Platform instance, you must update the Helm chart with the new versions of your plug-ins files.

You can update the Helm chart from the command line using `helm` commands, or from the OpenShift web console.

NOTE:
For upgrades in air-gapped or disconnected environments, the standard procedure cannot be used directly. You must first mirror the necessary container images to your local registry and prepare the Helm chart for offline use.

For detailed instructions on this process, see the Installing the self-service automation portal in an air-gapped environment

- **Update the Helm chart from the command line:**
  a. In a terminal, log in to your OpenShift instance.
  b. Open your OpenShift Project that has your self-service automation portal installation.
  c. Run the following command to ensure your Helm repository is up to date:

```
$ helm repo update
```

  d. Find the latest version of the Helm chart:

```
$ helm search repo openshift-helm-charts/redhat-rhaap-portal
```

  e. Upgrade the Helm release:

```
$ helm upgrade <release_name> openshift-helm-charts/redhat-rhaap-portal
--version <chart_version>
```

  Replace `<release_name>` with the name of your Helm release and `<chart_version>` with the new Helm chart version number you identified in the previous step.

- **Update the Helm chart using the OpenShift web console:**
  a. In a browser, log in to your OpenShift Container Platform web console.
  b. Switch to the **Developer** perspective.
  c. Ensure you are in the OpenShift Project that has your self-service automation portal Helm deployment.
  d. From the navigation menu, Select **Helm**.
  e. Find your existing self-service automation portal deployment in the list of **Helm releases** and click its name.
  f. Select **Actions › Upgrade**.
  g. In the **Upgrade** pane, select the version that you want to upgrade to from the **Chart Version** dropdown list.
  h. Review the YAML configuration to ensure your custom values are preserved.
  i. Click **Upgrade** to begin the upgrade.

Result

After the upgrade completes, verify that the updated self-service automation portal instance is running: . In the OpenShift Container Platform web console, navigate to the **Topology** view for your project. . Check that the self-service automation portal instance is available and that all associated pods are in a **Running** state.

Troubleshoot self-service automation portal upgrades

You might encounter issues during self-service automation portal upgrades. The following sections describe common problems and their solutions.

Plug-in version mismatch errors

Symptom: After you upgrade the Helm chart, self-service automation portal pods fail to start with errors indicating that plug-in files cannot be loaded or have incorrect versions.

Cause: The plug-in bundle major.minor version does not match the Helm chart major.minor version, or the plug-in bundle patch version is lower than the Helm chart patch version.

Solution:

1. Check your current plug-in bundle version:

   ```
   oc exec $(oc get pods -l deployment=plugin-registry -o
   jsonpath='{.items[0].metadata.name}') -- \

     ls -l /opt/app-root/src | grep ansible-plugin
   ```

   The version number appears in the plug-in file names.

2. Identify your Helm chart version:

   ```
   helm list -n <namespace>
   ```

   Look for your self-service automation portal release and note the chart version.

3. If the versions do not match:
   a. Download the correct plug-in bundle version from the Red Hat Ansible Automation Platform Product Software downloads page.
   b. Update your plug-in registry following the procedure in See "Download the plug-in TAR files" on page 1 119.
   c. Wait for the self-service automation portal pods to restart automatically.

Pods stuck in CrashLoopBackOff after upgrade

Symptom: After you upgrade, self-service automation portal pods repeatedly restart and show a status of `CrashLoopBackOff`.

Cause: Database schema migration failed, or the upgrade introduced configuration errors.

Solution:

1. Check the pod logs for specific error messages:

```
oc logs -n <namespace> <pod_name> --previous
```

2. If you see database migration errors:
   a. Verify that your database is accessible:

```
oc exec -it -n <namespace> <pod_name> -- pg_isready -h <database_host>
```

   b. Check database connection secrets:

```
oc get secret -n <namespace> | grep -i database
```

   c. If you use an external database, verify that the database user has the required permissions.
3. If you see configuration errors:
   a. Review your Helm values for syntax errors:

```
helm get values <release_name> -n <namespace>
```

   b. Compare with your previous working configuration to identify changes.
   c. Revert problematic configuration changes and upgrade again.

Self-service automation portal upgrade considerations for Ansible Automation Platform 2.4 to 2.6

Consider the following when you upgrade self-service automation portal from Ansible Automation Platform 2.4 to Ansible Automation Platform 2.6:

1. Upgrade your Ansible Automation Platform instance to version 2.6 before you upgrade self-service automation portal.

The self-service automation portal version that is compatible with Ansible Automation Platform 2.6 requires Ansible Automation Platform 2.6 features for full functionality.

2. Review the Ansible Automation Platform 2.6 release notes for breaking changes that affect self-service automation portal.
3. Back up your existing self-service automation portal configuration before you upgrade:

```
helm get values <release_name> -n <namespace> > backup-values.yaml
```

4. After you upgrade, verify that OAuth authentication still functions:
    a. Check that your OAuth application in Ansible Automation Platform is configured correctly.
    b. Test the sign-in functionality.
    c. If authentication fails, verify that the OAuth redirect URL matches your upgraded deployment URL.
5. Update any custom templates or configurations to use syntax that is compatible with the new self-service automation portal version.

Helm upgrade fails with a release not found error

Symptom: Running `helm upgrade` returns an error stating that the release cannot be found.

Cause: The Helm release name or namespace is incorrect.

Solution:

1. List all Helm releases in your cluster:

```
helm list --all-namespaces
```

2. Identify the correct release name and namespace for your self-service automation portal deployment.
3. Run the upgrade command with the correct parameters:

```
helm upgrade <correct_release_name> openshift-helm-charts/redhat-rhaap-portal
\
  --version <target_version> \
  -n <correct_namespace>
```

Custom values lost after upgrade

Symptom: After you upgrade, your custom configurations (such as custom CA certificates, OAuth settings, or RBAC configurations) are no longer applied.

Cause: The upgrade command did not include your custom values file, or the values were overwritten.

Solution:

1. Before you upgrade, export your current values:

```
helm get values <release_name> -n <namespace> > current-values.yaml
```

2. When you upgrade, specify your values file:

```
helm upgrade <release_name> openshift-helm-charts/redhat-rhaap-portal \
  --version <target_version> \
  -f current-values.yaml \
  -n <namespace>
```

3. After you upgrade, verify that your custom values are still applied:

```
helm get values <release_name> -n <namespace>
```

Related information
Self-service automation portal version compatibility

Upgrade automation dashboard

This procedure applies when upgrading from automation dashboard versions before 0.1 (which did not include Redis) to the current version. This process involves running the installation program and updating your cluster configuration with new authentication requirements.

NOTE: For more information about recent updates to automation dashboard, see What's new: Updates for automation dashboard.

Procedure

1. Download the latest installation bundle from access.redhat.com. Navigate to Downloads > Red Hat Ansible Automation Platform Product Software.

2. Extract the bundle to a new directory.
3. Copy the `inventory` file from your previous installation directory to the new directory.
4. Edit the `inventory` file to include the mandatory Redis configuration. You must add a `[redis]` group and define the `redis_mode` variable:

NOTE:
The inventory file lines prefixed with + must be added when upgrading your automation dashboard.

```
  [database]
host.example.com ansible_connection=local


+[redis]
+host.example.com ansible_connection=local
+
 [all:vars]
+redis_mode=standalone
+
 postgresql_admin_username=postgres
 postgresql_admin_password=TODO
 # registry_username=
```

5. Run the installation playbook from the new directory:

```
 ansible-galaxy collection install -r requirements.yml

ansible-playbook -i inventory
ansible.containerized_installer.dashboard_install --ask-become-pass
```

6. Update your `clusters.yaml` file. You must add the `refresh_token`, `client_id`, and `client_secret` variables to your existing cluster configurations.

NOTE:
For instructions on obtaining these values, see Integrating automation dashboard with your Ansible Automation Platform.

7. Apply the updated configuration to the dashboard. You must copy the configuration file to the container's `/tmp/` directory:

```
podman cp clusters.yaml automation-dashboard-web:/tmp/

podman exec -it automation-dashboard-web /venv/bin/python manage.py
setclusters /tmp/clusters.yaml
```

Result

1. Retrieve the current cluster configuration:

```
podman exec -it automation-dashboard-web /venv/bin/python ./manage.py
getclusters --decrypt
```

2. Verify that the output displays the content from your `clusters.yaml` file, including the `access_token`, `refresh_token`, `client_id`, and `client_secret` fields.

Troubleshoot synchronization failures

If new jobs from Ansible Automation Platform do not synchronize to the automation dashboard after an upgrade, an interrupted synchronization job might be blocking the process. This occurs if the automation dashboard service stops or restarts while a synchronization task is active.

Before you begin

- You have ssh access to the host machine.
- You have access to the PostgreSQL database, including the database user password defined in your inventory file (variable `dashboard_pg_password`).

To resolve this issue, you must manually remove the stuck jobs from the database.

Procedure

1. Connect to the automation dashboard database. You must replace `<password>` with your configured `dashboard_pg_password`. Replace 127.0.0.1 with database server address if external database is used.

```
POSTGRES_PASSWORD=<password> psql -h 127.0.0.1 -p 5432 -U aapdashboard -d
aapdashboard
```

2. Identify jobs that are in a pending or running state:

```
SELECT * FROM scheduler_syncjob WHERE status IN
('pending','waiting','running');
```

3. Wait approximately one minute and run the command again. If the same job IDs appear in the output, these jobs are stuck.
4. Delete the stuck jobs using their ID. Replace `<id>` with the ID returned in the previous step (for example, `20, 21`):

```
DELETE from scheduler_syncjob WHERE id IN (<id>);
```

Result

- Refresh the automation dashboard to confirm that synchronization has resumed.

Red Hat product documentation legal notices

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If

such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions. "This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code. The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component,

or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions. All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law. No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work’s users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies. You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions. You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so. A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms. You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

- c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms. "Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination. You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies. You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients. Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents. A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom. If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License. Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License. The Free Software Foundation may publish revised and/ or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty. THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES

SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16. If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
    <one line to give the program's name and a brief idea of what it does.>
    Copyright (C) <year>  <name of author>

    This program is free software: you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program.  If not, see <https://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
    <program>  Copyright (C) <year>  <name of author>
    This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see link:https://www.gnu.org/licenses/.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read link:https://www.gnu.org/licenses/why-not-lgpl.html.

Apache license

Version 2.0, January 2004

http://www.apache.org/licenses/

Terms and Conditions for use, reproduction, and distribution

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, **"control"** means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or **"Your"**) shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, **"submitted"** means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as **"Not a Contribution."**

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
4. If the Work includes a **"NOTICE"** text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications,

or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

www.redhat.com