



Red Hat Ansible Automation Platform 2.1

Red Hat Ansible Automation Platform Operator Installation Guide

This guide provides procedures and reference information for the supported installation scenarios for the Red Hat Ansible Automation Platform operator on OpenShift Container Platform

Red Hat Ansible Automation Platform 2.1 Red Hat Ansible Automation Platform Operator Installation Guide

This guide provides procedures and reference information for the supported installation scenarios for the Red Hat Ansible Automation Platform operator on OpenShift Container Platform

Legal Notice

Copyright © 2022 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

Providing Feedback: If you have a suggestion to improve this documentation, or find an error, please contact technical support at to create an issue on the Ansible Automation Platform Jira project using the Docs component.

Table of Contents

PREFACE	3
CHAPTER 1. PLANNING YOUR RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR INSTALLATION ON RED HAT OPENSIFT CONTAINER PLATFORM	4
1.1. ABOUT ANSIBLE AUTOMATION PLATFORM OPERATOR	4
1.2. SUPPORTED INSTALLATION SCENARIOS FOR RED HAT OPENSIFT CONTAINER PLATFORM	4
1.3. CUSTOM RESOURCES	5
1.4. ADDITIONAL RESOURCES	5
CHAPTER 2. INSTALLING THE RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR ON RED HAT OPENSIFT CONTAINER PLATFORM	6
CHAPTER 3. INSTALLING AND CONFIGURING AUTOMATION CONTROLLER ON RED HAT OPENSIFT CONTAINER PLATFORM WEB CONSOLE	7
3.1. PREREQUISITES	7
3.2. INSTALLING THE AUTOMATION CONTROLLER OPERATOR	7
3.2.1. Configure your automation controller operator route options	7
3.2.2. Configure the Ingress type for your automation hub operator	7
3.3. CONFIGURING AN EXTERNAL DATABASE FOR AUTOMATION CONTROLLER ON RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR	8
3.4. ADDITIONAL RESOURCES	9
CHAPTER 4. INSTALLING AND CONFIGURING AUTOMATION HUB ON RED HAT OPENSIFT CONTAINER PLATFORM WEB CONSOLE	10
4.1. PREREQUISITES	10
4.2. INSTALLING THE AUTOMATION HUB OPERATOR	10
4.2.1. Configure your automation hub operator route options	10
4.2.2. Configure the Ingress type for your automation hub operator	10
4.3. ACCESSING THE AUTOMATION HUB USER INTERFACE	10
4.4. CONFIGURING AN EXTERNAL DATABASE FOR AUTOMATION HUB ON RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR	11
4.5. ADDITIONAL RESOURCES	12
CHAPTER 5. INSTALLING ANSIBLE AUTOMATION PLATFORM OPERATOR FROM THE OPENSIFT CONTAINER PLATFORM CLI	13
5.1. PREREQUISITES	13
5.2. SUBSCRIBING A NAMESPACE TO AN OPERATOR USING THE OPENSIFT CONTAINER PLATFORM CLI	13
5.3. FETCHING AUTOMATION CONTROLLER LOGIN DETAILS FROM THE OPENSIFT CONTAINER PLATFORM CLI	15
5.3.1. Fetching the automation controller web address	15
5.3.2. Fetching the automation controller password	15
5.4. ADDITIONAL RESOURCES	16

PREFACE

Thank you for your interest in Red Hat Ansible Automation Platform. Ansible Automation Platform is a commercial offering that helps teams manage complex multi-tier deployments by adding control, knowledge, and delegation to Ansible-powered environments.

This guide helps you to understand the installation requirements and processes behind installing the Ansible Automation Platform operator on OpenShift Container Platform.

CHAPTER 1. PLANNING YOUR RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR INSTALLATION ON RED HAT OPENSIFT CONTAINER PLATFORM

Red Hat Ansible Automation Platform is supported on both Red Hat Enterprise Linux 8 and Red Hat OpenShift.

OpenShift operators help install and automate day-2 operations of complex, distributed software on Red Hat OpenShift Container Platform. The Ansible Automation Platform Operator enables you to deploy and manage Ansible Automation Platform components on Red Hat OpenShift Container Platform.

You can use this section to help plan your Red Hat Ansible Automation Platform installation on your Red Hat OpenShift Container Platform environment. Before installing, review the supported installation scenarios to determine which meets your requirements.

1.1. ABOUT ANSIBLE AUTOMATION PLATFORM OPERATOR

The Ansible Automation Platform Operator provides cloud-native, push-button deployment of new Ansible Automation Platform instances in your OpenShift environment. The Ansible Automation Platform Operator includes resource types to deploy and manage instances of Automation controller and Private Automation hub. It also includes automation controller job resources for defining and launching jobs inside your automation controller deployments.

Deploying Ansible Automation Platform instances with a Kubernetes native operator offers several advantages over launching instances from a playbook deployed on Red Hat OpenShift Container Platform, including upgrades and full lifecycle support for your Red Hat Ansible Automation Platform deployments.

You can install the Ansible Automation Platform Operator from the Red Hat Operators catalog in OperatorHub.

1.2. SUPPORTED INSTALLATION SCENARIOS FOR RED HAT OPENSIFT CONTAINER PLATFORM

You can use the OperatorHub on the Red Hat OpenShift Container Platform web console to install Ansible Automation Platform Operator.

Alternatively, you can install Ansible Automation Platform Operator from the OpenShift Container Platform command-line interface (CLI), **oc**.

Follow one of the workflows below to install the Ansible Automation Platform Operator and use it to install the components of Ansible Automation Platform that you require.

- Automation controller and customer resources first, then automation hub and customer resources;
- Automation hub and customer resources first, then automation controller and customer resources;
- Automation controller and customer resources;
- Automation hub and custom resources.

1.3. CUSTOM RESOURCES

You can define custom resources for each primary installation workflows.

1.4. ADDITIONAL RESOURCES

- See [Understanding OperatorHub](#) to learn more about OpenShift Container Platform OperatorHub.

CHAPTER 2. INSTALLING THE RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR ON RED HAT OPENSIFT CONTAINER PLATFORM

Prerequisites

- You have installed the Red Hat Ansible Automation Platform catalog in Operator Hub.



NOTE

Red Hat OpenShift Container Platform clusters running on AWS do not support ReadWriteMany without adding NFS or other storage.

Procedure

1. Log in to Red Hat OpenShift Container Platform.
2. Navigate to **Operators** → **OperatorHub**.
3. Search for the Red Hat Ansible Automation Platform operator and click **Install**.
4. Select an **Installation Mode**, **Installed Namespace**, and **Approval Strategy**.
5. Click **Install**.

The installation process will begin. When installation is complete, a modal will appear notifying you that the Red Hat Ansible Automation Platform operator is installed in the specified namespace.

- Click **View Operator** to view your newly installed Red Hat Ansible Automation Platform operator.

CHAPTER 3. INSTALLING AND CONFIGURING AUTOMATION CONTROLLER ON RED HAT OPENSIFT CONTAINER PLATFORM WEB CONSOLE

You can use these instructions to install the automation controller operator on Red Hat OpenShift Container Platform, specify custom resources, and deploy Ansible Automation Platform with an external database.

3.1. PREREQUISITES

- You have installed the Red Hat Ansible Automation Platform catalog in Operator Hub.

3.2. INSTALLING THE AUTOMATION CONTROLLER OPERATOR

1. Navigate to **Operators > Installed Operators**, then click on the **Ansible Automation Platform** operator.
2. Locate the **Automation controller** tab, then click **Create instance**.

You can proceed with configuring the instance using either the Form View or YAML view.

3.2.1. Configure your automation controller operator route options

The Red Hat Ansible Automation Platform operator installation form allows you to further configure your automation controller operator route options under **Advanced configuration**.

1. Click **Advanced configuration**.
2. Under **Tower Ingress type**, click the drop-down menu and select **Route**.
3. Under **Route DNS host**, enter a common host name that the route answers to.
4. Under **Route TLS termination mechanism**, click the drop-down menu and select **Edge** or **Passthrough**.
5. Under **Route TLS credential secret**, click the drop-down menu and select a secret from the list.

3.2.2. Configure the Ingress type for your automation hub operator

The Red Hat Ansible Automation Platform operator installation form allows you to further configure your automation hub operator Ingress under **Advanced configuration**.

1. Click **Advanced Configuration**.
2. Under **Ingress type**, click the drop-down menu and select **Ingress**.
3. Under **Ingress annotations**, enter any annotations to add to the ingress.
4. Under **Ingress TLS secret**, click the drop-down menu and select a secret from the list.

Once you have configured your automation controller operator, click **Create** at the bottom of the form view. Red Hat OpenShift Container Platform will now create the pods. This may take a few minutes.

- View progress by navigating to **Workloads → Pods** and locating the newly created instance.

3.3. CONFIGURING AN EXTERNAL DATABASE FOR AUTOMATION CONTROLLER ON RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR

For users who prefer to deploy Ansible Automation Platform with an external database, they can do so by configuring a secret with instance credentials and connection information, then applying it to their cluster using the **oc create** command.

By default, the Red Hat Ansible Automation Platform operator automatically creates and configures a managed PostgreSQL pod in the same namespace as your Ansible Automation Platform deployment. A user may instead choose to use an external database if they prefer to use a dedicated node to ensure dedicated resources or to manually manage backups, upgrades, or performance tweaks. The following section outlines the steps to configure an external database for your automation controller on a Ansible Automation Platform operator.

Prerequisite

The external database must be a PostgreSQL database that is the version supported by the current release of Ansible Automation Platform.



NOTE

Ansible Automation Platform 2.0 and 2.1 supports PostgreSQL 12.

Procedure

The external postgres instance credentials and connection information will need to be stored in a secret, which will then be set on the automation controller spec.

1. Create a **postgres_configuration_secret** .yaml file, following the template below:

```
apiVersion: v1
kind: Secret
metadata:
  name: external-postgres-configuration
  namespace: <target_namespace> 1
stringData:
  host: <external_ip_or_url_resolvable_by_the_cluster> 2
  port: <external_port> 3
  database: <desired_database_name>
  username: <username_to_connect_as>
  password: <password_to_connect_with> 4
  sslmode: prefer 5
  type: unmanaged
type: Opaque
```

- 1 Namespace to create the secret in. This should be the same namespace you wish to deploy to.
- 2 The resolvable hostname for your database node.
- 3 External port defaults to **5432**.
- 4

Value for variable **password** should not contain single or double quotes (' ') or backslashes (\) to avoid any issues during deployment, backup or restoration.

- 5 The variable **sslmode** is valid for **external** databases only. The allowed values are: **prefer**, **disable**, **allow**, **require**, **verify-ca**, and **verify-full**.

2. Apply **external-postgres-configuration-secret.yml** to your cluster using the **oc create** command.

```
$ oc create -f external-postgres-configuration-secret.yml
```

3. When creating your **AutomationController** custom resource object, specify the secret on your spec, following the example below:

```
apiVersion: awx.ansible.com/v1beta1
kind: AutomationController
metadata:
  name: controller-dev
spec:
  postgres_configuration_secret: external-postgres-configuration
```

3.4. ADDITIONAL RESOURCES

- For more information on running operators on OpenShift Container Platform, navigate to the [OpenShift Container Platform product documentation](#) and click the *Operators - Working with Operators in OpenShift Container Platform* guide.

CHAPTER 4. INSTALLING AND CONFIGURING AUTOMATION HUB ON RED HAT OPENSIFT CONTAINER PLATFORM WEB CONSOLE

You can use these instructions to install the automation hub operator on Red Hat OpenShift Container Platform, specify custom resources, and deploy Ansible Automation Platform with an external database.

4.1. PREREQUISITES

- You have installed the Red Hat Ansible Automation Platform operator in Operator Hub.

4.2. INSTALLING THE AUTOMATION HUB OPERATOR

1. Navigate to **Operators > Installed Operators**.
2. Locate the **Automation hub** entry, then click **Create instance**.

4.2.1. Configure your automation hub operator route options

The Red Hat Ansible Automation Platform operator installation form allows you to further configure your automation hub operator route options under **Advanced configuration**.

1. Click **Advanced configuration**.
2. Under **Ingress type**, click the drop-down menu and select **Route**.
3. Under **Route DNS host**, enter a common host name that the route answers to.
4. Under **Route TLS termination mechanism**, click the drop-down menu and select **Edge** or **Passthrough**.
5. Under **Route TLS credential secret**, click the drop-down menu and select a secret from the list.

4.2.2. Configure the Ingress type for your automation hub operator

The Red Hat Ansible Automation Platform operator installation form allows you to further configure your automation hub operator Ingress under **Advanced configuration**.

1. Click **Advanced Configuration**.
2. Under **Ingress type**, click the drop-down menu and select **Ingress**.
3. Under **Ingress annotations**, enter any annotations to add to the ingress.
4. Under **Ingress TLS secret**, click the drop-down menu and select a secret from the list.

Once you have configured your automation hub operator, click **Create** at the bottom of the form view. Red Hat OpenShift Container Platform will now create the pods. This may take a few minutes.

- View progress by navigating to **Workloads → Pods** and locating the newly created instance.

4.3. ACCESSING THE AUTOMATION HUB USER INTERFACE

You can access the automation hub interface once all pods have successfully launched.

1. Navigate to **Networking** → **Routes**.
2. Under **Location**, click on the URL for your automation hub instance.

The automation hub user interface will launch, You can sign in with the admin credentials specified during the operator configuration process.

4.4. CONFIGURING AN EXTERNAL DATABASE FOR AUTOMATION HUB ON RED HAT ANSIBLE AUTOMATION PLATFORM OPERATOR

For users who prefer to deploy Ansible Automation Platform with an external database, they can do so by configuring a secret with instance credentials and connection information, then applying it to their cluster using the **oc create** command.

By default, the Red Hat Ansible Automation Platform operator automatically creates and configures a managed PostgreSQL pod in the same namespace as your Ansible Automation Platform deployment. A user may instead choose to use an external database if they prefer to use a dedicated node to ensure dedicated resources or to manually manage backups, upgrades, or performance tweaks. The following section outlines the steps to configure an external database for your automation hub on a Ansible Automation Platform operator.

Prerequisite

The external database must be a PostgreSQL database that is the version supported by the current release of Ansible Automation Platform.



NOTE

Ansible Automation Platform 2.0 and 2.1 supports PostgreSQL 12.

Procedure

The external postgres instance credentials and connection information will need to be stored in a secret, which will then be set on the automation hub spec.

1. Create a **postgres_configuration_secret**.yaml file, following the template below:

```
apiVersion: v1
kind: Secret
metadata:
  name: external-postgres-configuration
  namespace: <target_namespace> 1
stringData:
  host: <external_ip_or_url_resolvable_by_the_cluster> 2
  port: <external_port> 3
  database: <desired_database_name>
  username: <username_to_connect_as>
  password: <password_to_connect_with> 4
  sslmode: prefer 5
  type: unmanaged
type: Opaque
```

- 1 Namespace to create the secret in. This should be the same namespace you wish to deploy to.
 - 2 The resolvable hostname for your database node.
 - 3 External port defaults to **5432**.
 - 4 Value for variable **password** should not contain single or double quotes (' ') or backslashes (\) to avoid any issues during deployment, backup or restoration.
 - 5 The variable **sslmode** is valid for **external** databases only. The allowed values are: **prefer**, **disable**, **allow**, **require**, **verify-ca**, and **verify-full**.
2. Apply **external-postgres-configuration-secret.yml** to your cluster using the **oc create** command.

```
$ oc create -f external-postgres-configuration-secret.yml
```

3. When creating your **AutomationHub** custom resource object, specify the secret on your spec, following the example below:

```
apiVersion: awx.ansible.com/v1beta1
kind: AutomationHub
metadata:
  name: hub-dev
spec:
  postgres_configuration_secret: external-postgres-configuration
```

4.5. ADDITIONAL RESOURCES

- For more information on running operators on OpenShift Container Platform, navigate to the [OpenShift Container Platform product documentation](#) and click the *Operators - Working with Operators in OpenShift Container Platform* guide.

CHAPTER 5. INSTALLING ANSIBLE AUTOMATION PLATFORM OPERATOR FROM THE OPENSIFT CONTAINER PLATFORM CLI

Use these instructions to install the Ansible Automation Platform Operator on Red Hat OpenShift Container Platform from the OpenShift Container Platform command-line interface (CLI) using the **oc** command.

5.1. PREREQUISITES

- Access to Red Hat OpenShift Container Platform using an account with operator installation permissions.
- The OpenShift Container Platform CLI **oc** command is installed on your local system. Refer to [Installing the OpenShift CLI](#) in the Red Hat OpenShift Container Platform product documentation for further information.

5.2. SUBSCRIBING A NAMESPACE TO AN OPERATOR USING THE OPENSIFT CONTAINER PLATFORM CLI

1. Create a project for the operator

```
oc new-project ansible-automation-platform
```

2. Create a file called **sub.yaml**.
3. Add the following YAML code to the **sub.yaml** file.

```
---
apiVersion: v1
kind: Namespace
metadata:
  labels:
    openshift.io/cluster-monitoring: "true"
  name: ansible-automation-platform
---
apiVersion: operators.coreos.com/v1
kind: OperatorGroup
metadata:
  name: ansible-automation-platform-operator
  namespace: ansible-automation-platform
spec:
  targetNamespaces:
    - ansible-automation-platform
---
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: ansible-automation-platform
  namespace: ansible-automation-platform
spec:
  channel: 'stable-2.1'
  installPlanApproval: Automatic
```

```

name: ansible-automation-platform-operator
source: redhat-operators
sourceNamespace: openshift-marketplace
---
apiVersion: automationcontroller.ansible.com/v1beta1
kind: AutomationController
metadata:
  name: example
  namespace: ansible-automation-platform
spec:
  create_preload_data: true
  route_tls_termination_mechanism: Edge
  garbage_collect_secrets: false
  loadbalancer_port: 80
  image_pull_policy: IfNotPresent
  projects_storage_size: 8Gi
  task_privileged: false
  projects_storage_access_mode: ReadWriteMany
  projects_persistence: false
  replicas: 1
  admin_user: admin
  loadbalancer_protocol: http
  nodeport_port: 30080

```

This file creates a **Subscription** object called *ansible-automation-platform* that subscribes the **ansible-automation-platform** namespace to the **ansible-automation-platform-operator** operator.

It then creates an **AutomationController** object called *example* in the **ansible-automation-platform** namespace.

To change the Automation controller name from *example*, edit the *name* field in the **kind: AutomationController** section of **sub.yaml** and replace *<automation_controller_name>* with the name you want to use:

```

apiVersion: automationcontroller.ansible.com/v1beta1
kind: AutomationController
metadata:
  name: <automation_controller_name>
  namespace: ansible-automation-platform

```

4. Run the **oc apply** command to create the objects specified in the **sub.yaml** file:

```
oc apply -f sub.yaml
```

To verify that the namespace has been successfully subscribed to the **ansible-automation-platform-operator** operator, run the **oc get subs** command:

```
$ oc get subs -n ansible-automation-platform
```

For further information about subscribing namespaces to operators, see [Installing from OperatorHub using the CLI](#) in the Red Hat OpenShift Container Platform *Operators* guide.

You can use the OpenShift Container Platform CLI to fetch the web address and the password of the Automation controller that you created.

5.3. FETCHING AUTOMATION CONTROLLER LOGIN DETAILS FROM THE OPENSIFT CONTAINER PLATFORM CLI

To login to the Automation controller, you need the web address and the password.

5.3.1. Fetching the automation controller web address

A Red Hat OpenShift Container Platform route exposes a service at a host name, so that external clients can reach it by name. When you created the automation controller instance, a route was created for it. The route inherits the name that you assigned to the automation controller object in the YAML file.

Use the following command to fetch the routes:

```
oc get routes -n <controller_namespace>
```

In the following example, the **example** automation controller is running in the **ansible-automation-platform** namespace.

```
$ oc get routes -n ansible-automation-platform
```

NAME	HOST/PORT	PATH	SERVICES	PORT	TERMINATION
WILDCARD					
example	example-ansible-automation-platform.apps-crc.testing		example-service	http	
edge/Redirect	None				

The address for the automation controller instance is **example-ansible-automation-platform.apps-crc.testing**.

5.3.2. Fetching the automation controller password

The YAML block for the automation controller instance in **sub.yaml** assigns values to the *name* and *admin_user* keys. Use these values in the following command to fetch the password for the automation controller instance.

```
oc get secret/<controller_name>-<admin_user>-password -o yaml
```

The default value for *admin_user* is **admin**. Modify the command if you changed the admin username in **sub.yaml**.

The following example retrieves the password for an automation controller object called **example**:

```
oc get secret/example-admin-password -o yaml
```

The password for the automation controller instance is listed in the **metadata** field in the output:

```
$ oc get secret/example-admin-password -o yaml
```

```
apiVersion: v1
data:
  password: ODhLSzJVanByTXVtVEdmUmVQMzdxZXJXazByT3VYUDM=
kind: Secret
metadata:
  annotations:
```

```
kubectl.kubernetes.io/last-applied-configuration: '{"apiVersion":"v1","kind":"Secret","metadata":
{"labels":{"app.kubernetes.io/component":"automationcontroller","app.kubernetes.io/managed-
by":"automationcontroller-
operator","app.kubernetes.io/name":"example","app.kubernetes.io/operator-
version":"","app.kubernetes.io/part-of":"example"},"name":"example-admin-
password","namespace":"ansible-automation-platform"},"stringData":
{"password":"88KK2UjprMumTGfReP37qerWk0rOuXP3"}}'
creationTimestamp: "2021-12-03T00:02:24Z"
labels:
  app.kubernetes.io/component: automationcontroller
  app.kubernetes.io/managed-by: automationcontroller-operator
  app.kubernetes.io/name: example
  app.kubernetes.io/operator-version: ""
  app.kubernetes.io/part-of: example
name: example-admin-password
namespace: ansible-automation-platform
resourceVersion: "185013"
uid: 391b22f0-52a2-4240-b942-665f1f589359
```

For this example, the password is **88KK2UjprMumTGfReP37qerWk0rOuXP3**.

5.4. ADDITIONAL RESOURCES

- For more information on running operators on OpenShift Container Platform, navigate to the [OpenShift Container Platform product documentation](#) and click the *Operators - Working with Operators in OpenShift Container Platform* guide.