



Red Hat OpenShift AI Self-Managed 2-latest

Release notes

Features, enhancements, resolved issues, and known issues associated with this release

Red Hat OpenShift AI Self-Managed 2-latest Release notes

Features, enhancements, resolved issues, and known issues associated with this release

Legal Notice

Copyright © 2024 Red Hat, Inc.

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

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

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

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

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

Abstract

These release notes provide an overview of new features, enhancements, resolved issues, and known issues in version 2-latest of Red Hat OpenShift AI.

Table of Contents

CHAPTER 1. OVERVIEW OF OPENSIFT AI	3
CHAPTER 2. NEW FEATURES AND ENHANCEMENTS	4
2.1. NEW FEATURES	4
2.2. ENHANCEMENTS	4
CHAPTER 3. TECHNOLOGY PREVIEW FEATURES	5
CHAPTER 4. SUPPORT REMOVALS	6
4.1. EMBEDDED SUBSCRIPTION CHANNEL DEPRECATED	6
4.2. REMOVAL OF BIAS DETECTION (TRUSTYAI)	6
4.3. UPCOMING DEPRECATION OF DATA SCIENCE PIPELINES V1	6
4.4. VERSION 1.2 NOTEBOOK CONTAINER IMAGES FOR WORKBENCHES ARE NO LONGER SUPPORTED	6
4.5. BETA SUBSCRIPTION CHANNEL DEPRECATED	6
CHAPTER 5. RESOLVED ISSUES	7
CHAPTER 6. KNOWN ISSUES	9
CHAPTER 7. PRODUCT FEATURES	29

CHAPTER 1. OVERVIEW OF OPENSIFT AI

Red Hat OpenShift AI is a platform for data scientists and developers of artificial intelligence and machine learning (AI/ML) applications.

OpenShift AI provides an environment to develop, train, serve, test, and monitor AI/ML models and applications on-premises or in the cloud.

For data scientists, OpenShift AI includes Jupyter and a collection of default notebook images optimized with the tools and libraries required for model development, and the TensorFlow and PyTorch frameworks. Deploy and host your models, integrate models into external applications, and export models to host them in any hybrid cloud environment. You can also accelerate your data science experiments through the use of graphics processing units (GPUs) and Habana Gaudi devices.

For administrators, OpenShift AI enables data science workloads in an existing Red Hat OpenShift or ROSA environment. Manage users with your existing OpenShift identity provider, and manage the resources available to notebook servers to ensure data scientists have what they require to create, train, and host models. Use accelerators to reduce costs and allow your data scientists to enhance the performance of their end-to-end data science workflows using graphics processing units (GPUs) and Habana Gaudi devices.

OpenShift AI offers two distributions:

- A **managed cloud service add-on** for Red Hat OpenShift Dedicated (with a Customer Cloud Subscription for AWS or GCP) or for Red Hat OpenShift Service on Amazon Web Services (ROSA).
For information about OpenShift AI on a Red Hat managed environment, see [Product Documentation for Red Hat OpenShift AI](#).
- **Self-managed software** that you can install on-premise or on the public cloud in a self-managed environment, such as OpenShift Container Platform.
For information about OpenShift AI as self-managed software on your OpenShift cluster in a connected or a disconnected environment, see [Product Documentation for Red Hat OpenShift AI Self-Managed](#).

For information about OpenShift AI supported software platforms, components, and dependencies, see [Supported configurations](#).

CHAPTER 2. NEW FEATURES AND ENHANCEMENTS

This section describes new features and enhancements in Red Hat OpenShift AI 2-latest.

2.1. NEW FEATURES

Support for self-signed certificates

You can now use self-signed certificates in your Red Hat OpenShift AI deployments and Data Science Projects in an OpenShift Container Platform cluster.

Some OpenShift AI components have additional options or required configuration for self-signed certificates, as described in [Working with certificates](#) (for disconnected environments, see [Working with certificates](#)).

2.2. ENHANCEMENTS

Upgraded OpenVINO Model Server

The OpenVINO Model Server has been upgraded to version 2023.3. For information on the changes and enhancements, see [OpenVINO™ Model Server 2023.3](#).

Support for gRPC protocol on single-model serving platform

The single-model serving platform now supports the gRPC API protocol in addition to REST. This support means that when you add a custom model serving runtime to the platform, you can specify which protocol the runtime uses.

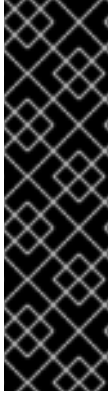
Extended support with new release channels

Starting with OpenShift AI 2.8, Red Hat provides production updates and support for the Red Hat OpenShift AI Operator in two new channels, in addition to the **fast**, **stable**, and **alpha** channels:

- The **stable-2.8** channel allows you to stay on the latest 2.8.x release with full support for seven months.
- The **eus-2.8** channel allows you to stay on the latest 2.8.x release with full support for seven months, followed by Extended Update Support for eleven months.

For more information about subscription channels, see [Installing the Red Hat OpenShift AI Operator](#).

CHAPTER 3. TECHNOLOGY PREVIEW FEATURES



IMPORTANT

This section describes Technology Preview features in Red Hat OpenShift AI 2-latest. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

For more information about the support scope of Red Hat Technology Preview features, see [Technology Preview Features Support Scope](#).

Distributed workloads

Distributed workloads enable data scientists to use multiple cluster nodes in parallel for faster, more efficient data processing and model training. The CodeFlare framework simplifies task orchestration and monitoring, and offers seamless integration for automated resource scaling and optimal node utilization with advanced GPU support.

Designed for data scientists, the CodeFlare framework enables direct workload configuration from Jupyter Notebooks or Python code, ensuring a low barrier of adoption, and streamlined, uninterrupted workflows. Distributed workloads significantly reduce task completion time, and enable the use of larger datasets and more complex models. The distributed workloads feature is currently available in Red Hat OpenShift AI 2-latest as a Technology Preview feature. This feature was first introduced in OpenShift AI 2.4.

code-server notebook image

Red Hat OpenShift AI now includes the code-server notebook image. See [code-server in GitHub](#) for more information.

With the code-server notebook image, you can customize your notebook environment to meet your needs using a variety of extensions to add new languages, themes, debuggers, and connect to additional services. Enhance the efficiency of your data science work with syntax highlighting, auto-indentation, and bracket matching.



NOTE

Elyra-based pipelines are not available with the code-server notebook image.

The code-server notebook image is currently available in Red Hat OpenShift AI 2-latest as a Technology Preview feature. This feature was first introduced in OpenShift AI 2.6.

CHAPTER 4. SUPPORT REMOVALS

This section describes major changes in support for user-facing features in Red Hat OpenShift AI.

4.1. EMBEDDED SUBSCRIPTION CHANNEL DEPRECATED

Starting with OpenShift AI 2.8, the **embedded** subscription channel has been removed. You can no longer select the **embedded** channel for a new installation of the Operator. For more information about subscription channels, see [Installing the Red Hat OpenShift AI Operator](#).

4.2. REMOVAL OF BIAS DETECTION (TRUSTYAI)

Starting with OpenShift AI 2.7, the bias detection (TrustyAI) functionality has been removed. If you previously had this functionality enabled, upgrading to OpenShift AI 2.7 or later will remove the feature. The default TrustyAI notebook image remains supported.

4.3. UPCOMING DEPRECATION OF DATA SCIENCE PIPELINES V1

Currently, data science pipelines in OpenShift AI are based on KubeFlow Pipelines v1. See [Working with data science pipelines](#) for more information.

Data science pipelines in upcoming releases will be based on KubeFlow Pipelines v2, using a different engine. OpenShift AI 2.8 is a stable release that will be supported for 7 months. We recommend that current data science pipeline users stay on OpenShift AI 2.8 until you are ready to migrate to the new pipelines solution.

For a detailed view of the 2.8 release lifecycle, including its full support phase window, see [Red Hat OpenShift AI Self-Managed Life Cycle](#).

4.4. VERSION 1.2 NOTEBOOK CONTAINER IMAGES FOR WORKBENCHES ARE NO LONGER SUPPORTED

When you create a workbench, you specify a notebook container image to use with the workbench. Starting with OpenShift AI 2.5, when you create a new workbench, version 1.2 notebook container images are not available to select. Workbenches that are already running with a version 1.2 notebook image continue to work normally. However, Red Hat recommends that you update your workbench to use the latest notebook container image.

4.5. BETA SUBSCRIPTION CHANNEL DEPRECATED

Starting with OpenShift AI 2.5, the **beta** subscription channel has been removed. You can no longer select the **beta** channel for a new installation of the Operator. For more information about subscription channels, see [Installing the Red Hat OpenShift AI Operator](#).

CHAPTER 5. RESOLVED ISSUES

This section describes notable issues that have been resolved in Red Hat OpenShift AI 2-latest.

[RHOAIENG-3355](#) - OVMS on KServe does not use accelerators correctly

Previously, when you deployed a model using the single-model serving platform and selected the **OpenVINO Model Server** serving runtime, if you requested an accelerator to be attached to your model server, the accelerator hardware was detected but was not used by the model when responding to queries. This issue is now resolved.

[RHOAIENG-2869](#) - Cannot edit existing model framework and model path in a multi-model project

Previously, when you tried to edit a model in a multi-model project using the **Deploy model** dialog, the **Model framework** and **Path** values did not update. This issue is now resolved.

[RHOAIENG-2724](#) - Model deployment fails because fields automatically reset in dialog

Previously, when you deployed a model or edited a deployed model, the **Model servers** and **Model framework** fields in the "Deploy model" dialog might have reset to the default state. The **Deploy** button might have remained enabled even though these mandatory fields no longer contained valid values. This issue is now resolved.

[RHOAIENG-2099](#) - Data science pipeline server fails to deploy in fresh cluster

Previously, when you created a data science pipeline server on a fresh cluster, the user interface remained in a loading state and the pipeline server did not start. This issue is now resolved.

[RHOAIENG-1199](#) (previously documented as [ODH-DASHBOARD-1928](#)) - Custom serving runtime creation error message is unhelpful

Previously, when you tried to create or edit a custom model-serving runtime and an error occurred, the error message did not indicate the cause of the error. The error messages have been improved.

[RHOAIENG-675](#) (previously documented as [RHODS-12906](#)) - Cannot use ModelMesh with object storage that uses private certificates

Previously, when you stored models in an object storage provider that used a private TLS certificate, the model serving pods failed to pull files from the object storage, and the **signed by unknown authority** error message was shown. This issue is now resolved.

[RHOAIENG-556](#) - ServingRuntime for KServe model is created regardless of error

Previously, when you tried to deploy a KServe model and an error occurred, the **InferenceService** custom resource (CR) was still created and the model was shown in the **Data Science Project** page, but the status would always remain unknown. The KServe deploy process has been updated so that the ServingRuntime is not created if an error occurs.

[RHOAIENG-548](#) (previously documented as [ODH-DASHBOARD-1776](#)) - Error messages when user does not have project administrator permission

Previously, if you did not have administrator permission for a project, you could not access some features, and the error messages did not explain why. For example, when you created a model server in an environment where you only had access to a single namespace, an **Error creating model server** error message appeared. However, the model server is still successfully created. This issue is now resolved.

RHOAIENG-66 - Ray dashboard route deployed by CodeFlare SDK exposes self-signed certs instead of cluster cert

Previously, when you deployed a Ray cluster by using the CodeFlare SDK with the **openshift_oauth=True** option, the resulting route for the Ray cluster was secured by using the **passthrough** method and as a result, the self-signed certificate used by the OAuth proxy was exposed. This issue is now resolved.

RHOAIENG-12 - Cannot access Ray dashboard from some browsers

In some browsers, users of the distributed workloads feature might not have been able to access the Ray dashboard because the browser automatically changed the prefix of the dashboard URL from **http** to **https**. This issue is now resolved.

RHODS-6216 - The ModelMesh oauth-proxy container is intermittently unstable

Previously, ModelMesh pods did not deploy correctly due to a failure of the ModelMesh **oauth-proxy** container. This issue occurred intermittently and only if authentication was enabled in the ModelMesh runtime environment. This issue is now resolved.

CHAPTER 6. KNOWN ISSUES

This section describes known issues in Red Hat OpenShift AI 2-latest and any known methods of working around these issues.

[RHOAIENG-4572](#) - Unable to run data science pipelines after install and upgrade in certain circumstances

You are unable to run data science pipelines after installing or upgrading OpenShift AI in the following circumstances:

- You have installed OpenShift AI and you have a valid CA certificate. Within the **default-dsci** object, you have changed the **managementState** field for the **trustedCABundle** field to **Removed** post-installation.
- You have upgraded OpenShift AI from version 2.6 to version 2.8 and you have a valid CA certificate.
- You have upgraded OpenShift AI from version 2.7 to version 2.8 and you have a valid CA certificate.

Workaround

As a workaround, perform the following steps:

1. In the OpenShift Container Platform web console, click **Operators** → **Installed Operators** and then click the **Red Hat OpenShift AI Operator**.
2. Click the **DSC Initialization** tab.
3. Click the **default-dsci** object.
4. Click the **YAML** tab.
5. In the **spec** section, change the value of the **managementState** field for **trustedCABundle** to **Managed**, as shown:

```
spec:
  trustedCABundle:
    managementState: Managed
```



NOTE

If you upgraded from OpenShift AI version 2.6 or 2.7 to version 2-latest, you must manually add the **trustedCABundle** field and the **managementState** field as they are not present in the YAML code. In addition, you do not need to enter a value in the **customCABundle** field.

6. Click **Save**.
7. Restart the dashboard replicaset.
 - a. In the OpenShift Container Platform web console, switch to the **Administrator** perspective.
 - b. Click **Workloads** → **Deployments**.

- c. Set the **Project** to **All Projects** or **redhat-ods-applications** to ensure you can see the appropriate deployment.
- d. Search for the **rhods-dashboard** deployment.
- e. Click the action menu (⋮) and select **Restart Rollout** from the list.
- f. Wait until the **Status** column indicates that all pods in the rollout have fully restarted.

RHOAIENG-4524 - BuildConfig definitions for RStudio images contain occurrences of incorrect branch

The BuildConfig definitions for the **RStudio** and **CUDA - RStudio** workbench images point to the wrong branch in OpenShift AI. The BuildConfig definitions incorrectly point to the **main** branch instead of the **rhoai-2.8** branch.

Workaround

To use the **RStudio** and **CUDA - RStudio** workbench images in OpenShift AI, follow the steps in the [Branch workaround for RStudio image BuildConfig definition](#) knowledgebase article.

RHOAIENG-4497 - Models on the multi-model serving platform with self-signed certificates stop working after upgrading to 2.8

In previous versions, if you wanted to use a self-signed certificate when serving models on the multi-model serving platform, you had to manually configure the **storage-config** secret used by your data connection to specify a certificate authority (CA) bundle.

If you upgrade a previous version of OpenShift AI that used that workaround to the latest version, the multi-model serving platform can no longer serve models.

Workaround

To use a self-signed certificate with both the multi- and single-model serving platforms, follow the steps in [Adding a CA bundle](#).

RHOAIENG-4430 - CA Bundle does not work for KServe without a data connection

If you have installed a certificate authority (CA) bundle on your OpenShift cluster to use self-signed certificates and then use the OpenShift AI dashboard to create a data connection to serve a model, OpenShift AI automatically stores the certificate in a secret called **storage-config**. However, if you bypass the OpenShift AI dashboard and configure the underlying **InferenceService** resource to specify a different secret name or a service account, OpenShift AI fails to validate SSL connections to the model and the model status includes **[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed: self signed certificate**.

Workaround

Use the OpenShift AI dashboard to create the data connection for your model. Do not manually modify the **InferenceService** resource to specify a different secret name or a service account.

RHOAIENG-4327 - Workbenches do not use the self-signed certificates from centrally configured bundle automatically

There are two bundle options to include self-signed certificates in OpenShift AI, **ca-bundle.crt** and **odh-ca-bundle.crt**. Self-signed certificates should apply to workbenches that you create after configuring self-signed certificates centrally. Workbenches do not use the self-signed certificates from the centrally configured bundle automatically.

Workaround

After configuring self-signed certificates centrally, they apply to any new workbenches and are available at **/etc/pki/tls/certs/** with the **custom** prefix. You can force the tools in your workbench to use these certificates by setting a known environment variable that points to your certificate path.

- If you used **ca-bundle.crt** when you configured certificates centrally, your path is **/etc/pki/tls/certs/custom-ca-bundle.crt**.
- If you used **odh-ca-bundle.crt** when you configured certificates centrally, your path is **/etc/pki/tls/certs/custom-odh-ca-bundle.crt**.

Set a known environment variable:

1. From the OpenShift AI dashboard, go to **Data Science Projects** and select the name of the project containing your workbench.
2. In the **Workbenches** section, click the action menu (:) beside the workbench that you want to update, and click **Edit workbench**.
3. Click the **Environment variables** tab.
4. Click **Add variable**.
5. From the **Select environment variable type** dropdown list, select **ConfigMap**.
6. In the **Key** field, enter **SSL_CERT_FILE**.
7. In the **Value** field, enter the path to your certificate file. For example, **/etc/pki/tls/certs/custom-ca-bundle.crt**.
8. Click **Update workbench**.

For more information, see [How to execute a pipeline from a Jupyter notebook in a disconnected environment](#).

RHOAIENG-4252 - Data science pipeline server deletion process fails to remove **ScheduledWorkFlow** resource

The pipeline server deletion process does not remove the **ScheduledWorkFlow** resource. As a result, new DataSciencePipelinesApplications (DSPAs) do not recognize the redundant **ScheduledWorkFlow** resource.

Workaround

1. Delete the pipeline server. For more information, see [Deleting a pipeline server](#).
2. In the OpenShift command-line interface (CLI), log in to your cluster as a cluster administrator and perform the following command to delete the redundant **ScheduledWorkFlow** resource.

```
$ oc -n <data science project name> delete scheduledworkflows --all
```

RHOAIENG-4240 - Jobs fail to submit to Ray cluster in unsecured environment

When running distributed data science workloads from notebooks in an unsecured OpenShift cluster, a **ConnectionError: Failed to connect to Ray** error message might be shown.

Workaround

In the **ClusterConfiguration** section of the notebook, set the **openshift_oauth** option to **True**.

[RHOAIENG-3981](#) - In unsecured environment, the functionality to wait for Ray cluster to be ready gets stuck

When running distributed data science workloads from notebooks in an unsecured OpenShift cluster, the functionality to wait for the Ray cluster to be ready before proceeding (**cluster.wait_ready()**) gets stuck even when the Ray cluster is ready.

Workaround

Perform one of the following actions:

- In the **ClusterConfiguration** section of the notebook, set the **openshift_oauth** option to **True**.
- Instead of using the **cluster.wait_ready()**, functionality, you can manually check the Ray cluster availability by opening the Ray cluster Route URL. When the Ray dashboard is available on the URL, then the cluster is ready.

[RHOAIENG-3963](#) - Unnecessary managed resource warning

When you edit and save the **OdhDashboardConfig** custom resource for the **redhat-ods-applications** project, the system incorrectly displays the following **Managed resource** warning message.

This resource is managed by DSC default-doc and any modifications may be overwritten. Edit the managing resource to preserve changes.

You can safely ignore this message.

Workaround

Click **Save** to close the warning message and apply your edits.

[RHOAIENG-1825](#) - After setting up self-signed certificates, executing pipelines might fail with workbenches that contain Elyra

After configuring self-signed certificates centrally, executing pipelines with workbenches that contain Elyra might fail.

Workaround

See the following knowledgebase articles for workaround steps:

- [Workbench workaround for executing a pipeline using Elyra](#)
- [Workbench workaround for an object storage connection with a self-signed certificate](#)
- [How to execute a pipeline from a Jupyter notebook in a disconnected environment](#)

[RHOAIENG-3025](#) - OVMS expected directory layout conflicts with the KServe StoragePuller layout

When you use the OpenVINO Model Server (OVMS) runtime to deploy a model on the single model serving platform (which uses KServe), there is a mismatch between the directory layout expected by OVMS and that of the model-pulling logic used by KServe. Specifically, OVMS requires the model files to be in the `/<mnt>/models/1/` directory, while KServe places them in the `/<mnt>/models/` directory.

Workaround

Perform the following actions:

1. In your S3-compatible storage bucket, place your model files in a directory called **1/**, for example, `/<s3_storage_bucket>/models/1/<model_files>`.
2. To use the OVMS runtime to deploy a model on the single model serving platform, choose one of the following options to specify the path to your model files:
 - If you are using the OpenShift AI dashboard to deploy your model, in the **Path** field for your data connection, use the `/<s3_storage_bucket>/models/` format to specify the path to your model files. Do not specify the **1/** directory as part of the path.
 - If you are creating your own **InferenceService** custom resource to deploy your model, configure the value of the **storageURI** field as `/<s3_storage_bucket>/models/`. Do not specify the **1/** directory as part of the path.

KServe pulls model files from the subdirectory in the path that you specified. In this case, KServe correctly pulls model files from the `/<s3_storage_bucket>/models/1/` directory in your S3-compatible storage.

RHOAIENG-3018 - OVMS on KServe does not expose the correct endpoint in the dashboard

When you use the OpenVINO Model Server (OVMS) runtime to deploy a model on the single-model serving platform, the URL shown in the **Inference endpoint** field for the deployed model is not complete. To send queries to the model, you must add the `/v2/models/_<model-name>_infer` string to the end of the URL. Replace `_<model-name>_` with the name of your deployed model.

Workaround

None.

RHOAIENG-2542 - Inference service pod does not always get an Istio sidecar

When you deploy a model using the single model serving platform (which uses KServe), the **istio-proxy** container might be missing in the resulting pod, even if the inference service has the **sidecar.istio.io/inject=true** annotation.

In OpenShift AI 2.7, the missing **istio-proxy** container might not present a problem. However, if the pod experiences connectivity issues, they might be caused by the missing container.

Workaround

Delete the faulty pod. OpenShift AI automatically creates a new pod, which should have the missing container.

RHOAIENG-3378 - Internal Image Registry is an undeclared hard dependency for Jupyter notebooks spawn process

Before you can start OpenShift AI notebooks and workbenches, you must first enable the internal, integrated container image registry in OpenShift Container Platform. Attempts to start notebooks or workbenches without first enabling the image registry will fail with an "InvalidImageName" error.

You can confirm whether the image registry is enabled for a cluster by using the following command:

```
$ oc get pods -n openshift-image-registry
```

Workaround

Enable the internal, integrated container image registry in OpenShift Container Platform.

See [Image Registry Operator in OpenShift Container Platform](#) for more information about how to set up and configure the image registry.

RHOAIENG-2759 - Model deployment fails when both secured and regular model servers are present in a project

When you create a second model server in a project where one server is using token authentication, and the other server does not use authentication, the deployment of the second model might fail to start.

Workaround

None available.

RHOAIENG-2602 - "Average response time" server metric graph shows multiple lines due to ModelMesh pod restart

The **Average response time** server metric graph shows multiple lines if the ModelMesh pod is restarted.

Workaround

None available.

RHOAIENG-2585 - UI does not display an error/warning when UWM is not enabled in the cluster

Red Hat OpenShift AI does not correctly warn users if User Workload Monitoring (UWM) is **disabled** in the cluster. UWM is necessary for the correct functionality of model metrics.

Workaround

Manually ensure that UWM is enabled in your cluster, as described in [Enabling monitoring for user-defined projects](#).

RHOAIENG-2555 - Model framework selector does not reset when changing Serving Runtime in form

When you use the **Deploy model** dialog to deploy a model on the single model serving platform, if you select a runtime and a supported framework, but then switch to a different runtime, the existing framework selection is not reset. This means that it is possible to deploy the model with a framework that is not supported for the selected runtime.

Workaround

While deploying a model, if you change your selected runtime, click the **Select a framework** list again and select a supported framework.

RHOAIENG-2479 - ModelMesh monitoring stack are not deleted during upgrade from 2.4 or 2.5

If you upgrade the Red Hat OpenShift AI operator from version 2.4 to 2.5, and then update the operator to version 2.6, 2.7, or 2.8, all components related to hardware resource-consuming model monitoring are removed from the cluster. Some residual model-monitoring resources, which do not consume hardware resources, will still be present.

Workaround

To delete these resources, execute the following **oc delete** commands with cluster-admin privileges:

```
$ oc delete service rhods-model-monitoring -n redhat-ods-monitoring
$ oc delete service prometheus-operated -n redhat-ods-monitoring
$ oc delete sa prometheus-custom -n redhat-ods-monitoring
$ oc delete sa rhods-prometheus-operator -n redhat-ods-monitoring
$ oc delete prometheus rhods-model-monitoring -n redhat-ods-monitoring
$ oc delete route rhods-model-monitoring -n redhat-ods-monitoring
```

RHOAIENG-2468 - Services in the same project as KServe might become inaccessible in OpenShift

If you deploy a non-OpenShift AI service in a data science project that contains models deployed on the single model serving platform (which uses KServe), the accessibility of the service might be affected by the network configuration of your OpenShift cluster. This is particularly likely if you are using the [OVN-Kubernetes network plugin](#) in combination with host network namespaces.

Workaround

Perform one of the following actions:

- Deploy the service in another data science project that does not contain models deployed on the single model serving platform. Or, deploy the service in another OpenShift project.
- In the data science project where the service is, add a [network policy](#) to accept ingress traffic to your application pods, as shown in the following example:

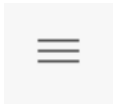
```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-ingress-to-myapp
spec:
  podSelector:
    matchLabels:
      app: myapp
  ingress:
    - {}
```

RHOAIENG-2312 - Importing numpy fails in code-server workbench

Importing **numpy** in your code-server workbench fails.

Workaround



1. In your code-server workbench, from the **Activity bar**, select the menu icon() > **View > Command Palette** to open the Command Palette.
In Firefox, you can use the F1 keyboard shortcut to open the command palette.
2. Enter **python: s**.
3. From the drop-down list, select the **Python: Select interpreter** action.
4. In the **Select Interpreter** dialog, select **Enter interpreter path...**

5. Enter **/opt/app-root/bin/python3** as the interpreter path and press **Enter**.
6. From the drop-down list, select the new Python interpreter.
7. Confirm that the new interpreter (**app-root**) appears on the **Status bar**. The selected interpreter persists if the workbench is stopped and started again, so the workaround should need to be performed only once for each workbench.

RHOAIENG-2228 - The performance metrics graph changes constantly when the interval is set to 15 seconds

On the **Endpoint performance** tab of the model metrics screen, if you set the **Refresh interval** to 15 seconds and the **Time range** to 1 hour, the graph results change continuously.

Workaround

None available.

RHOAIENG-2183 - Endpoint performance graphs might show incorrect labels

In the **Endpoint performance** tab of the model metrics screen, the graph tooltip might show incorrect labels.

Workaround

None available.

RHOAIENG-1919 - Model Serving page fails to fetch or report the model route URL soon after its deployment

When deploying a model from the OpenShift AI dashboard, the system displays the following warning message while the **Status** column of your model indicates success with an **OK**/green checkmark.

Failed to get endpoint for this deployed model. routes.rout.openshift.io"<model_name>" not found

Workaround

Refresh your browser page.

RHOAIENG-1452 - The Red Hat OpenShift AI Add-on gets stuck

The Red Hat OpenShift AI Add-on uninstall does not delete OpenShift AI components after being triggered via OCM APIs.

Workaround

Manually delete the remaining OpenShift AI resources as follows:

1. Delete the **DataScienceCluster** CR.
2. Wait until all pods are deleted from the **redhat-ods-applications** namespace.
3. If Serverless was set to **Managed** in the **DataScienceCluster** CR, wait until all pods are deleted from the **knative-serving** namespace.
4. Delete the **DSCInitialization** CR.
5. If Service Mesh was set to **Managed** in the **DSCInitialization** CR, wait until all pods are deleted from the **istio-system** namespace.

6. Uninstall the Red Hat OpenShift AI Operator.
7. Wait until all pods are deleted from the **redhat-ods-operator** namespace and the **redhat-ods-monitoring** namespace.

RHOAIENG-880 - Default pipelines service account is unable to create Ray clusters

You cannot create Ray clusters using the default pipelines Service Account.

Workaround

Authenticate using the CodeFlare SDK, by adding the following lines to the pipeline code:

```
from codeflare_sdk.cluster.auth import TokenAuthentication
auth = TokenAuthentication(
    token=openshift_token, server=openshift_server, skip_tls=True
)
auth_return = auth.login()
```

RHOAIENG-404 - No Components Found page randomly appears instead of Enabled page in OpenShift AI dashboard

A No Components Found page might appear when you access the Red Hat OpenShift AI dashboard.

Workaround

Refresh the browser page.

RHOAIENG-234 - Unable to view .ipynb files in VSCode in Insecured cluster

When you use the code-server notebook image on Google Chrome in an insecure cluster, you cannot view .ipynb files.

Workaround

Use a different browser.

RHOAIENG-2541 - KServe controller pod experiences OOM because of too many secrets in the cluster

If your OpenShift cluster has a large number of secrets, the KServe controller pod might continually crash due to an out-of-memory (OOM) error.

Workaround

Reduce the number of secrets in the OpenShift cluster until the KServe controller pod becomes stable.

RHOAIENG-1128 - Unclear error message displays when attempting to increase the size of a Persistent Volume (PV) that is not connected to a workbench

When attempting to increase the size of a Persistent Volume (PV) that is not connected to a workbench, an unclear error message is displayed.

Workaround

Verify that your PV is connected to a workbench before attempting to increase the size.

RHOAIENG-545 - Cannot specify a generic default node runtime image in JupyterLab pipeline editor

When you edit an Elyra pipeline in the JupyterLab IDE pipeline editor, and you click the **PIPELINE PROPERTIES** tab, and scroll to the **Generic Node Defaults** section and edit the **Runtime Image** field, your changes are not saved.

Workaround

Define the required runtime image explicitly for each node. Click the **NODE PROPERTIES** tab, and specify the required image in the **Runtime Image** field.

RHOAIENG-497 - Removing DSCI Results In OpenShift Service Mesh CR Being Deleted Without User Notification

If you delete the **DSCInitialization** resource, the OpenShift Service Mesh CR is also deleted. A warning message is not shown.

RHOAIENG-307 - Removing the DataScienceCluster deletes all OpenShift Serverless CRs

If you delete the **DataScienceCluster** custom resource (CR), all OpenShift Serverless CRs (including knative-serving, deployments, gateways, and pods) are also deleted. A warning message is not shown.

RHOAIENG-282 - Workload should not be dispatched if required resources are not available

Sometimes a workload is dispatched even though a single machine instance does not have sufficient resources to provision the RayCluster successfully. The **AppWrapper** CRD remains in a **Running** state and related pods are stuck in a **Pending** state indefinitely.

Workaround

Add extra resources to the cluster.

RHOAIENG-131 - gRPC endpoint not responding properly after the InferenceService reports as Loaded

When numerous **InferenceService** instances are generated and directed requests, Service Mesh Control Plane (SMCP) becomes unresponsive. The status of the **InferenceService** instance is **Loaded**, but the call to the gRPC endpoint returns with errors.

Workaround

Edit the **ServiceMeshControlPlane** custom resource (CR) to increase the memory limit of the Istio egress and ingress pods.

RHOAIENG-130 - Synchronization issue when the model is just launched

When the status of the KServe container is **Ready**, a request is accepted even though the TGIS container is not ready.

Workaround

Wait a few seconds to ensure that all initialization has completed and the TGIS container is actually ready, and then review the request output.

RHOAIENG-88 - Cannot log in to Red Hat OpenShift AI dashboard

Sometimes, when you try to log in to Red Hat OpenShift AI, the **500 internal error** error message is shown.

Workaround

Disable and re-enable the **dashboard** component in the **DataScienceCluster** object.

RHOAIENG-3115 - Model cannot be queried for a few seconds after it is shown as ready

Models deployed using the multi-model serving platform might be unresponsive to queries despite appearing as **Ready** in the dashboard. You might see an "Application is not available" response when querying the model endpoint.

Workaround

Wait 30-40 seconds and then refresh the page in your browser.

RHOAIENG-1619 (previously documented as DATA-SCIENCE-PIPELINES-165) - Poor error message when S3 bucket is not writable

When you set up a data connection and the S3 bucket is not writable, and you try to upload a pipeline, the error message **Failed to store pipelines** is not helpful.

Workaround

Verify that your data connection credentials are correct and that you have write access to the bucket you specified.

RHOAIENG-1207 (previously documented as ODH-DASHBOARD-1758) - Error duplicating OOTB custom serving runtimes several times

If you duplicate a model-serving runtime several times, the duplication fails with the **Serving runtime name "<name>" already exists** error message.

Workaround

Change the **metadata.name** field to a unique value.

RHOAIENG-1204 (previously documented as ODH-DASHBOARD-1771) - JavaScript error during Pipeline step initializing

Sometimes the pipeline **Run details** page stops working when the run starts.

Workaround

Refresh the page.

RHOAIENG-1203 (previously documented as ODH-DASHBOARD-1781) - Missing tooltip for Started Run status

Data science pipeline runs sometimes don't show the tooltip text for the status icon shown.

Workaround

For more information, view the pipeline **Run details** page and see the run output.

RHOAIENG-1201 (previously documented as ODH-DASHBOARD-1908) - Cannot create workbench with an empty environment variable

When creating a workbench, if you click **Add variable** but do not select an environment variable type from the list, you cannot create the workbench. The field is not marked as required, and no error message is shown.

[RHOAIENG-1196](#) (previously documented as [ODH-DASHBOARD-2140](#)) - Package versions displayed in dashboard do not match installed versions

The dashboard might display inaccurate version numbers for packages such as JupyterLab and Notebook. The package version number can differ in the image if the packages are manually updated.

Workaround

To find the true version number for a package, run the **pip list** command and search for the package name, as shown in the following examples:

```
$ pip list | grep jupyterlab
jupyterlab          3.5.3
$ pip list | grep notebook
notebook            6.5.3
```

[RHOAIENG-582](#) (previously documented as [ODH-DASHBOARD-1335](#)) - Rename Edit permission to Contributor

The term *Edit* is not accurate:

- For *most* resources, users with the **Edit** permission can not only edit the resource, they can also create and delete the resource.
- Users with the **Edit** permission cannot edit the project.

The term *Contributor* more accurately describes the actions granted by this permission.

[RHOAIENG-432](#) (previously documented as [RHODS-12928](#)) - Using unsupported characters can generate Kubernetes resource names with multiple dashes

When you create a resource and you specify unsupported characters in the name, then each space is replaced with a dash and other unsupported characters are removed, which can result in an invalid resource name.

[RHOAIENG-226](#) (previously documented as [RHODS-12432](#)) - Deletion of the notebook-culler ConfigMap causes Permission Denied on dashboard

If you delete the **notebook-controller-culler-config** ConfigMap in the **redhat-ods-applications** namespace, you can no longer save changes to the **Cluster Settings** page on the OpenShift AI dashboard. The save operation fails with an **HTTP request has failed** error.

Workaround

Complete the following steps as a user with **cluster-admin** permissions:

1. Log in to your cluster by using the **oc** client.
2. Enter the following command to update the **OdhDashboardConfig** custom resource in the **redhat-ods-applications** application namespace:

```
$ oc patch OdhDashboardConfig odh-dashboard-config -n redhat-ods-applications --
type=merge -p '{"spec": {"dashboardConfig": {"notebookController.enabled": true}}}'
```

[RHOAIENG-133](#) - Existing workbench cannot run Elyra pipeline after notebook restart

If you use the Elyra JupyterLab extension to create and run data science pipelines within JupyterLab, and you configure the pipeline server *after* you created a workbench and specified a notebook image within the workbench, you cannot execute the pipeline, even after restarting the notebook.

Workaround

1. Stop the running notebook.
2. Edit the workbench to make a small modification. For example, add a new dummy environment variable, or delete an existing unnecessary environment variable. Save your changes.
3. Restart the notebook.
4. In the left sidebar of JupyterLab, click **Runtimes**.
5. Confirm that the default runtime is selected.

RHOAIENG-52 - Token authentication fails in clusters with self-signed certificates

If you use self-signed certificates, and you use the Python **codeflare-sdk** in a notebook or in a Python script as part of a pipeline, token authentication will fail.

RHOAIENG-11 - Separately installed instance of CodeFlare Operator not supported

In Red Hat OpenShift AI, the CodeFlare Operator is included in the base product and not in a separate Operator. Separately installed instances of the CodeFlare Operator from Red Hat or the community are not supported.

Workaround

Delete any installed CodeFlare Operators, and install and configure Red Hat OpenShift AI, as described in the Red Hat Knowledgebase solution [How to migrate from a separately installed CodeFlare Operator in your data science cluster](#).

RHODS-12986 - Potential reconciliation error after upgrade to Red Hat OpenShift AI 2-latest

After you upgrade to Red Hat OpenShift AI 2-latest, a reconciliation error might appear in the Red Hat OpenShift AI Operator pod logs and in the **DataScienceCluster** custom resource (CR) conditions.

Example error:

```
2023-11-23T09:45:37Z ERROR Reconciler error {"controller": "datasciencecluster",
"controllerGroup": "datasciencecluster.opendatahub.io", "controllerKind": "DataScienceCluster",
"DataScienceCluster": {"name": "default-dsc", "namespace": "", "name": "default-dsc", "reconcileID":
"0c1a32ca-7ffd-4310-8259-f6baabf3c868", "error": "1 error occurred:\n\t* Deployment.apps \"rhods-
prometheus-operator\" is invalid: spec.selector: Invalid value:
v1.LabelSelector{MatchLabels:map[string]string{\"app.kubernetes.io/part-of\":\"model-mesh\",
\"app.opendatahub.io/model-mesh\":\"true\", \"k8s-app\":\"rhods-prometheus-operator\"},
MatchExpressions:[v1.LabelSelectorRequirement(nil)]: field is immutable\n\n"}
```

Workaround

Restart the Red Hat OpenShift AI Operator pod.

RHODS-12798 - Pods fail with "unable to init seccomp" error

Pods fail with **CreateContainerError** status or **Pending** status instead of **Running** status, because of a known kernel bug that introduced a **seccomp** memory leak. When you check the events on the namespace where the pod is failing, or run the **oc describe pod** command, the following error appears:

```
runc create failed: unable to start container process: unable to init seccomp: error loading seccomp filter into kernel: error loading seccomp filter: errno 524
```

Workaround

Increase the value of **net.core.bpf_jit_limit** as described in the Red Hat Knowledgebase solution [Pods failing with error loading seccomp filter into kernel: errno 524 in OpenShift 4](#).

KUBEFLOW-177 - Bearer token from application not forwarded by OAuth-proxy

You cannot use an application as a custom workbench image if its internal authentication mechanism is based on a bearer token. The OAuth-proxy configuration removes the bearer token from the headers, and the application cannot work properly.

RHOAIENG-642 (previously documented as RHODS-12903) - Successfully-submitted Elyra pipeline fails to run

If you use a private TLS certificate, and you successfully submit an Elyra-generated pipeline against the data science pipeline server, the pipeline steps fail to execute, and the following error messages are shown:

```
File "/opt/app-root/src/bootstrapper.py", line 747, in <module>
main()
File "/opt/app-root/src/bootstrapper.py", line 730, in main
Actions
...
WARNING: Retrying (Retry (total=4, connect=None, read=None, redirect=None, status=None)) after
connection broken by 'NewConnectionError('<pip._vendor.urllib3.connection.HTTPSConnection obj
In this situation, a new runtime image should be created, to include the correct CA bundle, as well as
all the required pip packages.
```

Workaround

Contact Red Hat Support for detailed steps to resolve this issue.

RHOAIENG-637 (previously documented as RHODS-12904) - Pipeline submitted from Elyra might fail when using private certificate

If you use a private TLS certificate, and you submit a pipeline from Elyra, the pipeline might fail with a **certificate verify failed** error message. This issue might be caused by either or both of the following situations:

- The object storage used for the pipeline server is using private TLS certificates.
- The data science pipeline server API endpoint is using private TLS certificates.

Workaround

Provide the workbench with the correct Certificate Authority (CA) bundle, and set various environment variables so that the correct CA bundle is recognized. Contact Red Hat Support for detailed steps to resolve this issue.

RHOAIENG-673 (previously documented as RHODS-12946) - Cannot install from PyPI mirror in disconnected environment or when using private certificates

In disconnected environments, Red Hat OpenShift AI cannot connect to the public-facing PyPI repositories, so you must specify a repository inside your network. If you are using private TLS certificates, and a data science pipeline is configured to install Python packages, the pipeline run fails.

Workaround

Add the required environment variables and certificates to your pipeline, as described in the Red Hat Knowledgebase solution [Install packages from PyPI Mirror fails on Data Science Pipelines in disconnected installation](#).

RHOAIENG-1666 (previously documented as **DATA-SCIENCE-PIPELINES-OPERATOR-349**) - The Import Pipeline button is prematurely accessible

When you import a pipeline to a workbench that belongs to a data science project, the **Import Pipeline** button is prematurely accessible before the pipeline server is fully available.

Workaround

Refresh your browser page and import the pipeline again.

RHOAIENG-5646 (previously documented as **NOTEBOOKS-218**) - Data science pipelines saved from the Elyra pipeline editor reference an incompatible runtime

When you save a pipeline in the Elyra pipeline editor with the format **.pipeline** in OpenShift AI version 1.31 or earlier, the pipeline references a runtime that is incompatible with OpenShift AI version 1.32 or later.

As a result, the pipeline fails to run after you upgrade OpenShift AI to version 1.32 or later.

Workaround

After you upgrade to OpenShift AI to version 1.32 or later, select the relevant runtime images again.

NOTEBOOKS-210 - A notebook fails to export as a PDF file in Jupyter

When you export a notebook as a PDF file in Jupyter, the export process fails with an error.

RHOAIENG-1210 (previously documented as **ODH-DASHBOARD-1699**) - Workbench does not automatically restart for all configuration changes

When you edit the configuration settings of a workbench, a warning message appears stating that the workbench will restart if you make any changes to its configuration settings. This warning is misleading because in the following cases, the workbench does not automatically restart:

- Edit name
- Edit description
- Edit, add, or remove keys and values of existing environment variables

Workaround

Manually restart the workbench.

RHOAIENG-1208 (previously documented as **ODH-DASHBOARD-1741**) - Cannot create a workbench whose name begins with a number

If you try to create a workbench whose name begins with a number, the workbench does not start.

Workaround

Delete the workbench and create a new one with a name that begins with a letter.

RHOAIENG-1205 (previously documented as RHODS-11791) - Usage data collection is enabled after upgrade

If you previously had the **Allow collection of usage data** option deselected (that is, disabled), this option becomes selected (that is, enabled) when you upgrade OpenShift AI.

Workaround

Manually reset the **Allow collection of usage data** option. To do this, perform the following actions:

1. In the OpenShift AI dashboard, in the left menu, click **Settings → Cluster settings**. The **Cluster Settings** page opens.
2. In the **Usage data collection** section, deselect **Allow collection of usage data**.
3. Click **Save changes**.

KUBEFLOW-157 - Logging out of JupyterLab does not work if you are already logged out of the OpenShift AI dashboard

If you log out of the OpenShift AI dashboard before you log out of JupyterLab, then logging out of JupyterLab is not successful. For example, if you know the URL for a Jupyter notebook, you are able to open this again in your browser.

Workaround

Log out of JupyterLab before you log out of the OpenShift AI dashboard.

RHODS-9789 - Pipeline servers fail to start if they contain a custom database that includes a dash in its database name or username field

When you create a pipeline server that uses a custom database, if the value that you set for the **dbname** field or **username** field includes a dash, the pipeline server fails to start.

Workaround

Edit the pipeline server to omit the dash from the affected fields.

RHOAIENG-580 (previously documented as **RHODS-9412**) - Elyra pipeline fails to run if workbench is created by a user with edit permissions

If a user who has been granted edit permissions for a project creates a project workbench, that user sees the following behavior:

- During the workbench creation process, the user sees an **Error creating workbench** message related to the creation of Kubernetes role bindings.
- Despite the preceding error message, OpenShift AI still creates the workbench. However, the error message means that the user will not be able to use the workbench to run Elyra data science pipelines.
- If the user tries to use the workbench to run an Elyra pipeline, Jupyter shows an **Error making request** message that describes failed initialization.

Workaround

A user with administrator permissions (for example, the project owner) must create the workbench on behalf of the user with edit permissions. That user can then use the workbench to run Elyra pipelines.

RHOAIENG-583 (previously documented as RHODS-8921 and RHODS-6373) - You cannot create a pipeline server or start a workbench when cumulative character limit is exceeded

When the cumulative character limit of a data science project name and a pipeline server name exceeds 62 characters, you are unable to successfully create a pipeline server. Similarly, when the cumulative character limit of a data science project name and a workbench name exceeds 62 characters, workbenches fail to start.

Workaround

Rename your data science project so that it does not exceed 30 characters.

RHODS-7718 - User without dashboard permissions is able to continue using their running notebooks and workbenches indefinitely

When a Red Hat OpenShift AI administrator revokes a user's permissions, the user can continue to use their running notebooks and workbenches indefinitely.

Workaround

When the OpenShift AI administrator revokes a user's permissions, the administrator should also stop any running notebooks and workbenches for that user.

RHOAIENG-1157 (previously documented as RHODS-6955) - An error can occur when trying to edit a workbench

When editing a workbench, an error similar to the following can occur:

Error creating workbench

Operation cannot be fulfilled on notebooks.kubeflow.org "workbench-name": the object has been modified; please apply your changes to the latest version and try again

RHOAIENG-1132 (previously documented as RHODS-6383) - An ImagePullBackOff error message is not displayed when required during the workbench creation process

Pods can experience issues pulling container images from the container registry. If an error occurs, the relevant pod enters into an **ImagePullBackOff** state. During the workbench creation process, if an **ImagePullBackOff** error occurs, an appropriate message is not displayed.

Workaround

Check the event log for further information on the **ImagePullBackOff** error. To do this, click on the workbench status when it is starting.

RHOAIENG-1152 (previously documented as RHODS-6356) - The notebook creation process fails for users who have never logged in to the dashboard

The dashboard's notebook **Administration** page displays users belonging to the user group and admin group in OpenShift. However, if an administrator attempts to start a notebook server on behalf of a user who has never logged in to the dashboard, the server creation process fails and displays the following error message:

Request invalid against a username that does not exist.

Workaround

Request that the relevant user logs into the dashboard.

RHODS-5906 - The NVIDIA GPU Operator is incompatible with OpenShift 4.11.12

Provisioning a GPU node on a OpenShift 4.11.12 cluster results in the **nvidia-driver-daemonset** pod getting stuck in a CrashLoopBackOff state. The NVIDIA GPU Operator is compatible with OpenShift 4.11.9 and 4.11.13. In addition, the minimum version of OpenShift required for an installation of OpenShift AI is 4.12.

RHODS-5763 - Incorrect package version displayed during notebook selection

The **Start a notebook server** page displays an incorrect version number for the Anaconda notebook image.

RHODS-5543 - When using the NVIDIA GPU Operator, more nodes than needed are created by the Node Autoscaler

When a pod cannot be scheduled due to insufficient available resources, the Node Autoscaler creates a new node. There is a delay until the newly created node receives the relevant GPU workload. Consequently, the pod cannot be scheduled and the Node Autoscaler's continuously creates additional new nodes until one of the nodes is ready to receive the GPU workload. For more information about this issue, see the Red Hat Knowledgebase solution [When using the NVIDIA GPU Operator, more nodes than needed are created by the Node Autoscaler](#).

Workaround

Apply the **cluster-api/accelerator** label in **machineset.spec.template.spec.metadata**. This causes the autoscaler to consider those nodes as unready until the GPU driver has been deployed.

RHOAIENG-1149 (previously documented RHODS-5216) - The application launcher menu incorrectly displays a link to OpenShift Cluster Manager

Red Hat OpenShift AI incorrectly displays a link to the OpenShift Cluster Manager from the application launcher menu. Clicking this link results in a "Page Not Found" error because the URL is not valid.

RHOAIENG-1137 (previously documented as RHODS-5251) - Notebook server administration page shows users who have lost permission access

If a user who previously started a notebook server in Jupyter loses their permissions to do so (for example, if an OpenShift AI administrator changes the user's group settings or removes the user from a permitted group), administrators continue to see the user's notebook servers on the server **Administration** page. As a consequence, an administrator is able to restart notebook servers that belong to the user who's permissions were revoked.

RHODS-4799 - Tensorboard requires manual steps to view

When a user has TensorFlow or PyTorch notebook images and wants to use TensorBoard to display data, manual steps are necessary to include environment variables in the notebook environment, and to import those variables for use in your code.

Workaround

When you start your notebook server, use the following code to set the value for the **TENSORBOARD_PROXY_URL** environment variable to use your OpenShift AI user ID.

```
import os
os.environ["TENSORBOARD_PROXY_URL"] = os.environ["NB_PREFIX"] + "/proxy/6006/"
```

RHODS-4718 - The Intel® oneAPI AI Analytics Toolkits quick start references nonexistent sample notebooks

The Intel® oneAPI AI Analytics Toolkits quick start, located on the **Resources** page on the dashboard, requires the user to load sample notebooks as part of the instruction steps, but refers to notebooks that do not exist in the associated repository.

RHODS-4627 - The CronJob responsible for validating Anaconda Professional Edition's license is suspended and does not run daily

The CronJob responsible for validating Anaconda Professional Edition's license is automatically suspended by the OpenShift AI operator. As a result, the CronJob does not run daily as scheduled. In addition, when Anaconda Professional Edition's license expires, Anaconda Professional Edition is not indicated as disabled on the OpenShift AI dashboard.

RHOAIENG-1141 (previously documented as RHODS-4502) - The NVIDIA GPU Operator tile on the dashboard displays button unnecessarily

GPUs are automatically available in Jupyter after the NVIDIA GPU Operator is installed. The **Enable** button, located on the NVIDIA GPU Operator tile on the **Explore** page, is therefore redundant. In addition, clicking the **Enable** button moves the NVIDIA GPU Operator tile to the **Enabled** page, even if the Operator is not installed.

RHOAIENG-1135 (previously documented as RHODS-3985) - Dashboard does not display Enabled page content after ISV operator uninstall

After an ISV operator is uninstalled, no content is displayed on the **Enabled** page on the dashboard. Instead, the following error is displayed:

```
Error loading components
HTTP request failed
```

Workaround

Wait 30-40 seconds and then refresh the page in your browser.

RHODS-3984 - Incorrect package versions displayed during notebook selection

In the OpenShift AI interface, the **Start a notebook server page** displays incorrect version numbers for the JupyterLab and Notebook packages included in the oneAPI AI Analytics Toolkit notebook image. The page might also show an incorrect value for the Python version used by this image.

Workaround

When you start your oneAPI AI Analytics Toolkit notebook server, you can check which Python packages are installed on your notebook server and which version of the package you have by running the **!pip list** command in a notebook cell.

RHODS-2956 - Error can occur when creating a notebook instance

When creating a notebook instance in Jupyter, a **Directory not found** error appears intermittently. This error message can be ignored by clicking **Dismiss**.

RHOAIENG-1147 (previously documented as RHODS-2881) - Actions on dashboard not clearly visible

The dashboard actions to revalidate a disabled application license and to remove a disabled application tile are not clearly visible to the user. These actions appear when the user clicks on the application tile's **Disabled** label. As a result, the intended workflows might not be clear to the user.

RHOAIENG-1134 (previously documented as RHODS-2879) - License revalidation action appears unnecessarily

The dashboard action to revalidate a disabled application license appears unnecessarily for applications that do not have a license validation or activation system. In addition, when a user attempts to revalidate a license that cannot be revalidated, feedback is not displayed to state why the action cannot be completed.

RHOAIENG-2305 (previously documented as RHODS-2650) - Error can occur during Pachyderm deployment

When creating an instance of the Pachyderm operator, a webhook error appears intermittently, preventing the creation process from starting successfully. The webhook error is indicative that, either the Pachyderm operator failed a health check, causing it to restart, or that the operator process exceeded its container's allocated memory limit, triggering an Out of Memory (OOM) kill.

Workaround

Repeat the Pachyderm instance creation process until the error no longer appears.

RHODS-2096 - IBM Watson Studio not available in OpenShift AI

IBM Watson Studio is not available when OpenShift AI is installed on OpenShift Dedicated 4.9 or higher, because it is not compatible with these versions of OpenShift Dedicated. Contact [Marketplace support](#) for assistance manually configuring Watson Studio on OpenShift Dedicated 4.9 and higher.

CHAPTER 7. PRODUCT FEATURES

Red Hat OpenShift AI provides a rich set of features for data scientists and IT operations administrators. To learn more, see [Introduction to Red Hat OpenShift AI](#).