



Red Hat CodeReady Workspaces 2.2

Release Notes and Known Issues

Release Notes and Known Issues for Red Hat CodeReady Workspaces 2.2

Red Hat CodeReady Workspaces 2.2 Release Notes and Known Issues

Release Notes and Known Issues for Red Hat CodeReady Workspaces 2.2

Robert Kratky
rkratky@redhat.com

Michal Maléř
mmaler@redhat.com

Fabrice Flore-Thébault
ffloreth@redhat.com

Yana Hontyk
yhontyk@redhat.com

Legal Notice

Copyright © 2020 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

Information about new and noteworthy features as well as known issues in Red Hat CodeReady Workspaces 2.2.

Table of Contents

CHAPTER 1. RELEASE NOTES	4
1.1. ABOUT RED HAT CODEREADY WORKSPACES	4
1.2. NOTABLE ENHANCEMENTS	5
1.2.1. Operator available in OSD 4.3	5
1.2.2. Update to RHEL 8.2, SSO 7.4, EAP 7.3.1, and MongoDB 3.6	5
1.2.2.1. Added the ability to configure custom multiple devfile registries	5
1.2.3. Update devfile editor with code assist from registries	5
1.2.4. Support for mounting Kubernetes secret into a workspace container	5
1.2.5. Option to run workspace in debug mode	5
1.2.6. Providing LogWatchers metrics in standard API	5
1.2.7. Added ability to choose in which namespace a workspace is going to be created	5
1.3. OTHER ENHANCEMENTS	6
1.3.1. Devfile updates to use new samples	6
1.3.2. Update Java-based devfiles	6
1.3.3. RAM and CPU limits improvements	6
1.3.4. Installing CodeReady Workspaces operator from the crwctl command line tool	6
1.3.5. Keybinding switching	6
1.4. SUPPORTED PLATFORMS	6
1.4.1. Supported platforms and installation methods	6
1.4.2. Installing and deploying CodeReady Workspaces	7
1.4.3. Support policy	7
1.5. DIFFERENCE BETWEEN ECLIPSE CHE AND RED HAT CODEREADY WORKSPACES	7
CHAPTER 2. KNOWN ISSUES	9
2.1. OPENSIFT CONTAINER PLATFORM DEVELOPER TOPOLOGY PERSPECTIVE VIEW IS NOT WORKING AS EXPECTED	9
2.2. CANNOT INSTALL CRW ON A CLUSTER CONFIGURED WITH AN AUTHENTICATED PROXY	9
2.3. AIRGAP INSTALLATION OF CODEREADY WORKSPACES 2.2 DOES NOT WORK ON OPENSIFT 4.4	9
2.4. INSTALLING VIA CRWCTL IN OPERATOR MODE IN SOME CASES FAILS	9
2.5. DASHBOARD SPLASH SCREEN REFERS TO AN INCORRECT LOCATION ON OPENSIFT 3.11	10
2.6. THE GETTING STARTED EXAMPLES ARE FAILING IN THE AIRGAP MODE INSTALLATIONS	10
2.7. THE TERMINATE TASK IDE FUNCTION DOES NOT WORK PROPERLY ON OPENSIFT 4.4 AND AWS	10
2.8. THE GO TO DEFINITION AND DEBUG FUNCTIONS DO NOT WORK IN THE GO WORKSPACE	10
2.9. SYNTAX HIGHLIGHTING MARKS VALID IMPORTS AS ERRONEOUS IN PHP-DI DEVFILE	10
2.10. THE CRWCTL WORKSPACE:INJECT COMMAND DOES NOT WORK IN WORKSPACES WITH OPENSIFT OAUTH SUPPORT	11
2.11. OPENSIFT PROJECTS REMAIN PRESENT AFTER A NAMESPACE DELETION IN CODEREADY WORKSPACES 2.2	11
2.12. THE CRWCTL SERVER:DELETE COMMAND DOES NOT REMOVE THE OPENSIFT PROJECT	11
2.13. THE WORKSPACE SHARING DOES NOT WORK	12
2.14. THE CRWCTL WORKSPACE:DELETE COMMAND CANNOT FIND ALREADY STARTED WORKSPACE	12
2.15. THE CRWCTL SERVER:DELETE COMMAND BREAKS EXISTING CODEREADY WORKSPACES DEPLOYMENTS ON THE SAME OPENSIFT CLUSTER	12
2.16. THE CRWCTL SERVER:DELETE COMMAND INCORRECTLY REMOVES CHECLUSTER FROM ALL CODEREADY WORKSPACES PROJECTS ON OPENSIFT CONTAINER PLATFORM	13
2.17. DELETING A CHECLUSTER CUSTOM RESOURCE CAUSES CODEREADY WORKSPACES OPERATOR ERRORS	13
2.18. OPENSIFT CONNECTOR PLUG-IN REQUIRES MANUALLY CONNECTING TO THE TARGET CLUSTER	13
2.19. AN EMBEDDED APPLICATION OF THE JAVA-EAP-MAVEN STACK TENDS TO FAIL TO LAUNCH IN THE DEBUG MODE	13
2.20. ERROR HIGHLIGHTING AND CODE COMPLETION DO NOT WORK IN A GO DEVFILE	14

2.21. MOUNTING A KUBERNETES SECRET INTO A WORKSPACE CONTAINER DOES NOT WORK ON OPENSIFT 3.11	14
2.22. CREATING A NAMESPACE FAILS WHEN A LOGIN NAME WITH UNSUPPORTED CHARACTERS IS USED	14
2.23. USING OF NFS MOUNTS MAY CAUSE INSTALLATION OR WORKSPACE STARTUP FAILURES	14
2.24. CODEREADY WORKSPACES DEPLOYED WITHOUT TLS SUPPORT CAUSES SOME FEATURES TO NOT WORK PROPERLY	14
2.25. CODEREADY WORKSPACES FAILS TO SHUT DOWN AFTER EXECUTING CRWCTL SERVER:STOP	15
CHAPTER 3. FAQ	16

CHAPTER 1. RELEASE NOTES

Red Hat CodeReady Workspaces is a web-based integrated development environment (IDE). CodeReady Workspaces runs in OpenShift and is well-suited for container-based development.

This section documents the most important features and bug fixes in Red Hat CodeReady Workspaces. For the list of CodeReady Workspaces 2.2 release issues, see the [Chapter 2, *Known issues*](#) section.

1.1. ABOUT RED HAT CODEREADY WORKSPACES

Having multiple CodeReady Workspaces deployments on the same cluster is not recommended, and the ability to do so may be removed in a future release.

Red Hat CodeReady Workspaces 2.2 provides an enterprise-level cloud developer workspace server and browser-based IDE. CodeReady Workspaces includes ready-to-use developer stacks for some of the most popular programming languages, frameworks, and Red Hat technologies.

This minor release of Red Hat CodeReady Workspaces is based on Eclipse Che 7.14 and offers a number of enhancements and new features, including:

- **Support for OpenShift Container Platform 4.4**

Using the CodeReady Workspaces Operator and `crwctl`, CodeReady Workspaces can be installed on OpenShift Container Platform versions 4.4, 4.3, and 3.11. Users can then benefit from all Red Hat-managed container application platform features.

See [OpenShift 4.4 - New features and enhancements](#). See [Supported platforms for deploying Red Hat CodeReady Workspaces](#) for a table of supported platforms and installation methods.

- **CodeReady Workspaces Air Gap with OpenShift 4.4**

On OpenShift 4.4, it is now possible to follow [Configuring OperatorHub for restricted networks using OpenShift Container Platform](#) and configure the CodeReady Workspaces Operator to be used this way.

- **Languages updates**

Provided versions:

- Python - CodeReady Workspaces registries were updated with the **`vscode-python`** version **2020.5.86806**.

- **Improvements to workspace startup and overall performance**

- **New chapters in the documentation**

Installation Guide

- Configuring CodeReady Workspaces Persistent Volumes strategy
- Configuring CodeReady Workspaces with and without RH-SSO
- Mounting a secret as a file or an environment variable into a workspace container

CodeReady Workspaces 2.2 is available in the [Red Hat Container Catalog](#). Install it on OpenShift Container Platform, starting at version 3.11, by following the instructions in the [Installing CodeReady Workspaces](#) chapter of the Installation Guide.

CodeReady Workspaces 2.2 is available from the OperatorHub in OpenShift 4.3 and beyond. CodeReady Workspaces 2.2 is based on a new Operator that uses the Operator Lifecycle Manager. This makes the CodeReady Workspaces installation flow simpler and doable without leaving the OpenShift Console.

To install CodeReady Workspaces for OpenShift 4.3 or later, get CodeReady Workspaces from the OperatorHub and follow the [Installing CodeReady Workspaces on OpenShift 4 from OperatorHub](#) chapter of the Installation Guide.

1.2. NOTABLE ENHANCEMENTS

1.2.1. Operator available in OSD 4.3

1.2.2. Update to RHEL 8.2, SSO 7.4, EAP 7.3.1, and MongoDB 3.6

- Security improvements and bug fixes

1.2.2.1. Added the ability to configure custom multiple devfile registries

Users with administration rights can now configure custom multiple devfile registries by editing the CustomResource to list more than one registry.

1.2.3. Update devfile editor with code assist from registries

- Added defaultSnippets support for the devfile editor.
- Provide code assist for the **chePlugins** from a registry.

1.2.4. Support for mounting Kubernetes secret into a workspace container

Users can mount a secret that contains sensitive data in a workspace container. This reapplies the stored data from the secret automatically for every newly created workspace. As a result, the user does not have to provide these credentials and configuration settings manually.

Suitable for:

- Maven configuration, the **settings.xml** file
- SSH key pairs
- AWS authorization tokens

1.2.5. Option to run workspace in debug mode

To help diagnose issues with workspace start, users can start a workspace with additional debugging logs enabled.

1.2.6. Providing LogWatchers metrics in standard API

Using the LogWatchers metrics that are collected in the debug mode, users can track the load of all containers over a threads of logs simultaneously.

1.2.7. Added ability to choose in which namespace a workspace is going to be created

CodeReady Workspaces server can be configured to honor the user selection of a namespace when a workspace is created. This feature is disabled by default. To allow user-defined workspace namespaces:

- For Operator deployments, set the **allowUserDefinedWorkspaceNamespaces** field in the CheCluster Custom Resource.

1.3. OTHER ENHANCEMENTS

1.3.1. Devfile updates to use new samples

Devfiles have been updated to provide more interesting and up to date sample code.

1.3.2. Update Java-based devfiles

Java-based devfiles have been migrated to use JDK 11 where possible, remove EAP if not required, and continue to use JDK 8 where needed. Since Maven 3.6 was moved into the JDK sidecar to support Quarkus, it is possible to use Quarkus with JDK 8 and 11 if a user sets up a custom devfile.

In addition, a new **plugin-java8-rhel8** container replaces the old **stacks-java-rhel8** container and also provides support for node and python in the same container. This reduces the install footprint for most devfiles and provides a single container for the most widely used languages.

1.3.3. RAM and CPU limits improvements

Pre-configured limits could exceed cluster-wide resources settings causing workspaces to be unable to start. This enhancement disables any kind of default limits/requests by setting them into -1 value by default.

- Applying for **chePlugin** and Other components, a user is allowed to set or disable any default limits and requests.

1.3.4. Installing CodeReady Workspaces operator from the crwctl command line tool

Only supported for OCP 3.11. For OCP 4, please use the OperatorHub UI method.

1.3.5. Keybinding switching

Using a set of plug-ins, CodeReady Workspaces adds the ability for the Che-Theia editor to switch key bindings between Default, Vi, and Emacs options.

1.4. SUPPORTED PLATFORMS

1.4.1. Supported platforms and installation methods

The following section provides information about the availability of CodeReady Workspaces 2.2 on OpenShift Container Platform, OpenShift Dedicated, and about their supported installation methods.

Red Hat CodeReady Workspaces can be installed on OpenShift Container Platform and OpenShift Dedicated starting at version 3.11.

Table 1.1. Availability of CodeReady Workspaces 2.2 on OpenShift Container Platform and OpenShift Dedicated

	3.11	4.3	4.4	4.5
OpenShift Container Platform	✓	✓	✓	Technical Preview
OpenShift Dedicated	✓	✓	Technical Preview	Technical Preview

Table 1.2. Supported installation method for CodeReady Workspaces 2.2 on OpenShift Container Platform and OpenShift Dedicated

	3.11	4.3	4.4
OpenShift Container Platform	crwctl	OperatorHub	OperatorHub
OpenShift Dedicated	crwctl	OperatorHub	N/A

It is possible to use the `crwctl` utility script for deploying CodeReady Workspaces 2.2 on OpenShift Container Platform versions 4.3, 4.4, and OpenShift Dedicated version 4.3. This method is considered unofficial and serves as a backup installation method for situations where the installation method using OperatorHub is not available.

1.4.2. Installing and deploying CodeReady Workspaces

For OpenShift 3.11, see the [Installing CodeReady Workspaces](#) chapter of the Administrator Guide.

For OpenShift 4.4, see the [Installing CodeReady Workspaces from Operator Hub](#) chapter of the Installation Guide.

1.4.3. Support policy

For Red Hat CodeReady Workspaces 2.2, Red Hat will provide support for deployment, configuration, and use of the product.

CodeReady Workspaces 2.2 has been tested on Chrome version 83.0.4103.97 (Official Build) (64-bit).

For more information, see [CodeReady Workspaces life-cycle and support policy](#).

1.5. DIFFERENCE BETWEEN ECLIPSE CHE AND RED HAT CODEREADY WORKSPACES

The main differences between CodeReady Workspaces and Eclipse Che are:

- CodeReady Workspaces is built on RHEL8 to ensure the latest security fixes are included, vs. Alpine distributions that take a longer time to update.
- CodeReady Workspaces uses Red Hat Single Sign-On (SSO) instead of the upstream project Keycloak.

- CodeReady Workspaces provides a smaller supported subset of plugins compared to Che. CodeReady Workspaces provides devfiles for working with other Red Hat technologies such as EAP and Fuse.
- CodeReady Workspaces is supported on OpenShift Container Platform and OpenShift Dedicated; Che can also run on other Kubernetes clusters.

Red Hat also provides licensing, packaging, and support, so CodeReady Workspaces is considered a more stable product than the upstream Eclipse Che project.

CHAPTER 2. KNOWN ISSUES

This section lists known issues with Red Hat CodeReady Workspaces 2.2. Where available, workaround suggestions are provided.

2.1. OPENSIFT CONTAINER PLATFORM DEVELOPER TOPOLOGY PERSPECTIVE VIEW IS NOT WORKING AS EXPECTED

The **Edit source code** icon does not work properly from Developer Topology Perspective.

After the an CodeReady Workspaces application deploys, the topology view should display a clickable icon that loads a CodeReady Workspaces workspace using the devfile that is in the source repository. Instead, it displays the GitHub icon that brings the user to the GitHub source code repository.

- [CRW-1078](#)

2.2. CANNOT INSTALL CRW ON A CLUSTER CONFIGURED WITH AN AUTHENTICATED PROXY

When installing CRW on a cluster configured with an authenticated proxy, the installation fails because the CodeReady Workspaces server cannot contact the RH-SSO OIDC endpoint.

- [CRW-1043](#)

2.3. AIRGAP INSTALLATION OF CODEREADY WORKSPACES 2.2 DOES NOT WORK ON OPENSIFT 4.4

On OpenShift 4.4, the catalog bundle image is corrupted and does not appear in OperatorHub. To workaround this issue, update to OpenShift 4.5.

- [CRW-1013](#)

2.4. INSTALLING VIA CRWCTL IN OPERATOR MODE IN SOME CASES FAILS

Installation of CodeReady Workspaces with OpenShift OAuth using `crwctl` in **operator** mode in some cases fails or does not deploy resources if existing resources from another `crwctl` exist in the cluster.

To workaround this issue, delete old resources and perform a fresh installation. For instructions, see step 6 of the [Installing CodeReady Workspaces on OpenShift 3 using the Operator](#) Installation Guide chapter.

Having multiple CodeReady Workspaces deployments on the same cluster is not recommended, and the ability to do so may be removed in a future release.

- [Issue 17187](#)
- [CRW-987](#)
- [CRW-967](#)

2.5. DASHBOARD SPLASH SCREEN REFERS TO AN INCORRECT LOCATION ON OPENSIFT 3.11

When CodeReady Workspaces deployed on OpenShift 3.11 encounters a certificate error, the error notification links to the wrong remediation instructions.

To import the certificates, see: [Installing Codeready Workspaces in TLS mode with self-signed certificates](#)

- [CRW-1062](#)

2.6. THE GETTING STARTED EXAMPLES ARE FAILING IN THE AIRGAP MODE INSTALLATIONS

Some of the sample projects included in the Getting Started devfiles are not designed for offline or airgapped use, so some commands may not work. To resolve this, user may have to talk to a organization's administrator to get access to internal mirrors, such as NMP, Maven, and PIP.

The base functions of the **Getting started** ZIP-archived samples embedded in the offline devfile registry do not work.

Commands that require internet access to run: **Run**, **Simple build** `, `Outline

- [CRW-1059](#)

2.7. THE TERMINATE TASK IDE FUNCTION DOES NOT WORK PROPERLY ON OPENSIFT 4.4 AND AWS

Tasks terminated with the **Terminate Task** IDE function do not end properly. As a consequence, reactivating the task returns an ``The task is already running active`` error with the option to use the **Restart task** function. However, using this function causes a port conflict and the task fails.

- [CRW-827](#)

2.8. THE GO TO DEFINITION AND DEBUG FUNCTIONS DO NOT WORK IN THE go WORKSPACE

Some devfiles are provided as Technology Preview and are not guaranteed to work for all implementations.

- [CRW-1019](#)
- [CRW-778](#)

2.9. SYNTAX HIGHLIGHTING MARKS VALID IMPORTS AS ERRONEOUS IN PHP-DI DEVFILE

Workspaces created from the **PHP-DI** devfile incorrectly highlight valid code as erroneous.

To work around this issue:

1. Install the PHP dependencies using the **Install dependencies** IDE command.

2. If not done automatically after the dependencies installation, restart the workspace manually.
3. Refresh the browser.
 - [CRW-981](#)

2.10. THE `crwctl workspace:inject` COMMAND DOES NOT WORK IN WORKSPACES WITH OPENSIFT OAUTH SUPPORT

The `crwctl workspace:inject` command causes the following error during the workspace Pod localization in workspaces created with OpenShift OAuth support.

```
$ crwctl workspace:inject -n codeready-tls-oauth -k
✓ Verify if namespace codeready-tls-oauth exists
✗ Verify if the workspaces is running
  → No workspace pod is found
  Injecting configurations
  › Error: No workspace pod is found
```

To work around the issue, use the `oc login` command inside the affected container instead.

- [CRW-803](#)

2.11. OPENSIFT PROJECTS REMAIN PRESENT AFTER A NAMESPACE DELETION IN CODEREADY WORKSPACES 2.2

When a workspace is created in a dedicated namespace and the namespace is later entirely deleted, the corresponding OpenShift project currently does not get deleted.

To complete the deletion process, delete the project manually from the OpenShift console:

```
$ oc delete project <projectname>
```

- [RHDEVDOCS-1821](#)

2.12. THE `crwctl server:delete` COMMAND DOES NOT REMOVE THE OPENSIFT PROJECT

After using the `crwctl server:delete` command, the OpenShift project hosted the CodeReady Workspaces instance remains. This makes it impossible to install a new CodeReady Workspaces instance into the still existing default namespace.

To uninstall CodeReady Workspaces completely, remove the namespace manually:

1. Stop the Red Hat CodeReady Workspaces Server:

```
$ crwctl server:stop
```

2. Obtain the name of the CodeReady Workspaces namespace:

```
$ oc get checluster --all-namespaces -o=jsonpath="{.items[*].metadata.namespace}"
```

- Remove CodeReady Workspaces from the cluster:

```
$ crwctl server:delete -n <namespace>
```

This removes all CodeReady Workspaces installations from the cluster.

- Delete the **checluster** object and the **codeready-workspaces** resource:

```
$ oc delete checluster codeready-workspaces --namespace=<openshift_namespace>
```

<openshift_namespace> is the name of the OpenShift project where CodeReady Workspaces is deployed.

- Delete the OpenShift namespace:

```
$ oc delete project <openshift_namespace>
```

- [CRW-499](#)
- [CRW-714](#)

2.13. THE WORKSPACE SHARING DOES NOT WORK

The **File > Share** IDE command currently launches the **Workspace** tab, but the **Share** tab is missing.

- [CRW-1041](#)

2.14. THE CRWCTL WORKSPACE:DELETE COMMAND CANNOT FIND ALREADY STARTED WORKSPACE

To work around the issue:

- Use **crwctl server:stop**
- Use **crwctl workspace:delete** again

- [CRW-1017](#)

2.15. THE CRWCTL SERVER:DELETE COMMAND BREAKS EXISTING CODEREADY WORKSPACES DEPLOYMENTS ON THE SAME OPENSIFT CLUSTER

The **crwctl server:delete** command removes certain cluster-scoped objects, which causes all other CodeReady Workspaces deployments to terminate unexpectedly.

To work around the issue, patch the Custom Resource Definition:

+

```
$ oc patch customresourcedefinition/checlusters.org.eclipse.che -p \
  '{"metadata": {"finalizers": null }}' --type merge
```


Having multiple CodeReady Workspaces deployments on the same cluster is not recommended, and the ability to do so may be removed in a future release.

- [Issue 17187](#)
- [CRW-1039](#)
- [Issue 16000](#)

2.16. THE CRWCTL SERVER:DELETE COMMAND INCORRECTLY REMOVES CHECLUSTER FROM ALL CODEREADY WORKSPACES PROJECTS ON OPENSIFT CONTAINER PLATFORM

Instead of removing **checluster** from the current CodeReady Workspaces project, the **checluster** is deleted from all projects in the OpenShift Container Platform cluster.

- [CRW-467](#)

2.17. DELETING A CHECLUSTER CUSTOM RESOURCE CAUSES CODEREADY WORKSPACES OPERATOR ERRORS

Uninstalling the CodeReady Workspaces manually by deleting the **checluster** custom resource in the OperatorHub causes errors in the CodeReady Workspaces Operator. As a consequence, attempting to re-install CodeReady Workspaces in OperatorHub fails.

- [CRW-763](#)

2.18. OPENSIFT CONNECTOR PLUG-IN REQUIRES MANUALLY CONNECTING TO THE TARGET CLUSTER

By default, the OpenShift Connector plug-in logs in to the cluster as **inClusterUser**, which in some cases does not have the **manage project** permission. This causes an error message to be displayed when creating a new project using OpenShift Application Explorer:

```
Failed to create Project with error 'Error: Command failed: "/tmp/vscode-unpacked/redhat.vscode-openshift -connector.latest.qvkozqtkba.openshift-connector-0.1.4-523.vsix/extension/out/tools/linux/odo" project create test-project X projectrequests.project.openshift.io is forbidden
```

To work around this issue, log out from the local cluster and relog in to OpenShift cluster using the OpenShift user's credentials.

- [RHDEVDOCS-1806](#)

2.19. AN EMBEDDED APPLICATION OF THE JAVA-EAP-MAVEN STACK TENDS TO FAIL TO LAUNCH IN THE DEBUG MODE

When launching an embedded application of the **java-eap-maven** stack in the debug mode, the launch often fails. The dialog window with the application URL is already displayed, but the terminal output states that the application is still starting. Using the URL link displayed causes an error.

- [CRW-617](#)

2.20. ERROR HIGHLIGHTING AND CODE COMPLETION DO NOT WORK IN A GO DEVFILE

To work around this issue, update the Go language server plug-in to the latest version.

- [Issue 16113](#)
- [CRW-778](#)

2.21. MOUNTING A KUBERNETES SECRET INTO A WORKSPACE CONTAINER DOES NOT WORK ON OPENSIFT 3.11

The **subPath** in **VolumeMount** property, necessary for mounting secrets, is supported on Kubernetes version 1.15, where OpenShift 3.11 is based on Kubernetes version 1.11. As a consequence, mounting a Kubernetes secret into a workspace container does not work.

- [Issue 17213](#)

2.22. CREATING A NAMESPACE FAILS WHEN A LOGIN NAME WITH UNSUPPORTED CHARACTERS IS USED

If the **<username>-che** namespace strategy is used, a problem can occur with the user names that contain characters incompatible with the Kubernetes namespace name pattern.

The specified namespace `example@redhat.com-codeready` is invalid: a DNS-1123 label must consist of lower case alphanumeric characters or '-', and must start and end with an alphanumeric character (e.g. 'my-name', or '123-abc', regex used for validation is '[a-z0-9]([-a-z0-9]*[a-z0-9])?')

To work around this issue, do one of the following:

- Use the **CHE_INFRA_KUBERNETES_NAMESPACE_DEFAULT** property with `<userid>` or `<workspaceid>` instead of `<username>`.
- Use the **CHE_INFRA_KUBERNETES_NAMESPACE_ALLOW_USER_DEFINED** property to allow users to select namespaces by themselves.
 - [CRW-1032](#)

2.23. USING OF NFS MOUNTS MAY CAUSE INSTALLATION OR WORKSPACE STARTUP FAILURES

For best performance, Red Hat recommends using block storage for Persistent Volumes used with CodeReady Workspaces. The use of NFS is not recommended.

- [RHDEVDOCS-2039](#)

2.24. CODEREADY WORKSPACES DEPLOYED WITHOUT TLS SUPPORT CAUSES SOME FEATURES TO NOT WORK PROPERLY

In CodeReady Workspaces 2.1 and later, secure HTTPS is required to use the most recent Theia IDE, and therefore TLS mode is enabled by default. Disabling the TLS support will cause user experience to suffer and some UI will not work as expected or at all.

For example, the welcome page may be blank or broken, images may be missing, and other functionality may not work properly.

- [Issue 17012](#)
- [Issue 15318](#)

2.25. CODEREADY WORKSPACES FAILS TO SHUT DOWN AFTER EXECUTING `CRWCTL SERVER:STOP`

The `crwctl server:stop` command is unable to shut down the CodeReady Workspaces server and instead fails with a timeout and displays the following error message:

```
> Error: E_SHUTDOWN_CHE_SERVER_FAIL - Failed to shutdown CodeReady Workspaces
server. E_CHE_API_NO_RESPONSE - Endpoint: http://codeready-ndp-
test.apps.crw.codereadyqe.com/api/system/stop?shutdown=true - Error message: timeout of
> 3000ms exceeded
```

To work around the issue, execute `crwctl server:stop` again.

- [CRW-519](#)

CHAPTER 3. FAQ

1. Can I install CodeReady Workspaces **offline** (that is, disconnected from the internet)?
Yes, you can. For detailed instructions, see [Installing CodeReady Workspaces in restricted environments](#) chapter of the *Installation Guide*.
2. Can I use non-default certificates with CodeReady Workspaces?
Yes, you can use self-signed or public certificates. See [Installing CodeReady Workspaces on OpenShift v3](#) chapter of the *Installation Guide*.
3. Can I run multiple workspaces simultaneously?
Yes. The following two conditions must be met to run multiple workspaces simultaneously:
 - CodeReady Workspaces must use the **per-workspace** Persistent Volume Claim (PVC) strategy (default), and
 - Persistent volumes (PVs) must use **ReadWriteMany** (RWX) access mode.
Thus to run multiple workspaces simultaneously, ensure the following configuration is set:
 - Set **ReadWriteMany** (RWX) access mode for PVs.
 - Use the **per-workspace** PVC strategy (default in CodeReady Workspaces), or optionally, the **unique** strategy.