# Red Hat OpenShift Dev Spaces 3.14

## Release notes and known issues

Release notes and known issues for Red Hat OpenShift Dev Spaces 3.14

Last Updated: 2024-06-17

# Red Hat OpenShift Dev Spaces 3.14 Release notes and known issues

Release notes and known issues for Red Hat OpenShift Dev Spaces 3.14

Jana Vrbkova
jvrbkova@redhat.com

## Legal Notice



## Abstract

Information about new and noteworthy features as well as known issues in Red Hat OpenShift Dev Spaces 3.14.

# Table of Contents

# MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message .

# CHAPTER 1. ABOUT RED HAT OPENSHIFT DEV SPACES

Red Hat OpenShift Dev Spaces provides web-based development environments on Red Hat OpenShift with an enterprise-level setup:

- Cloud Development Environments (CDE) server
- IDEs such as Microsoft Visual Studio Code - Open Source and JetBrains IntelliJ IDEA Community (Technology Preview)
- Containerized environments with popular programming languages, frameworks, and Red Hat technologies

Red Hat OpenShift Dev Spaces is well-suited for container-based development.

Red Hat OpenShift Dev Spaces 3.14 is based on Eclipse Che 7.86.

## 1.1. SUPPORTED PLATFORMS

OpenShift Dev Spaces runs on OpenShift 4.12–4.15 on the following CPU architectures:

- AMD64 and Intel 64 (**x86_64**)
- IBM Power (**ppc64le**) and IBM Z ( **s390x**)

**Additional resources**

- OpenShift Documentation
- Red Hat OpenShift Dev Spaces administration guide

## 1.2. SUPPORT POLICY

For Red Hat OpenShift Dev Spaces 3.14, Red Hat will provide support for deployment, configuration, and use of the product.

**Additional resources**

- OpenShift Dev Spaces life-cycle and support policy .

## 1.3. DIFFERENCES BETWEEN RED HAT OPENSHIFT DEV SPACES AND ECLIPSE CHE

There are some differences between Red Hat OpenShift Dev Spaces and the upstream project on which it is based, Eclipse Che:

- OpenShift Dev Spaces is supported only on Red Hat OpenShift.
- OpenShift Dev Spaces is based on Red Hat Enterprise Linux and is regularly updated to include the latest security fixes.
- OpenShift Dev Spaces provides devfiles for working with languages and technologies such as Quarkus, Lombok, NodeJS, Python, DotNet, Golang, C/C++, and PHP. You can find the latest sample projects in the devspaces-devfileregistry container image sources.

- OpenShift Dev Spaces uses OpenShift OAuth for user login and management.

Red Hat provides licensing and packaging to ensure enterprise-level support for OpenShift Dev Spaces.

# CHAPTER 2. NEW FEATURES AND ENHANCEMENTS

## 2.1. EXPOSED URL PARAMETERS ON THE 'GIT REPO OPTIONS' PANEL ON THE USER DASHBOARD

Git branch, remotes, and path to devfile URL parameters are now exposed on the 'Git Repo Options' panel of the User Dashboard.

### Additional resources

- CRW-6527

## 2.2. SUPPORT RAW DEVFILE URLS WITHOUT YAML EXTENSION

Starting from this release, you can create workspaces from any URL that serves a valid devfile, such as in the following cases:

- https://registry.devfile.io/devfiles/ollama/1.0.0
- https://registry.devfile.io/devfiles/dotnet60/1.0.3

### Additional resources

- CRW-6528

## 2.3. ADD INITCONTAINER FOR INITIALIZING PERSISTENTHOME WHEN $HOME PERSISTENCE IS ENABLED

With this release, if you have $HOME persistence enabled, there is an **initContainer** that runs the **entrypoint** script and initializes persistentHome. This prevents a race condition from occurring.

Previously, the persistent user home setup occurred in the entrypoint of the tooling container. Notably, the **entrypoint** was used to run **stow** to make symbolic links for the home directory to be saved on the PVC. If a postStart event depends on files or edits files located in the home directory, there is a risk of a race condition caused by the post-start event running before the **stow** execution is finished. To prevent this, the **entrypoint** script must be completed before the postStart events run.

You can find more details about this enhancement in the demo video.

### Additional resources

- CRW-6529

# CHAPTER 3. BUG FIXES

## 3.1. EXTENSION SECRETS ARE LOST ON WORKSPACE RESTART

Extension secrets for Visual Studio Code - Open Source ("Code - OSS") are no longer lost after workspace restart, but encrypted and persisted in the browser's local storage. This allows extensions like Ansible that use Visual Studio Code Secrets Storage API to persist the data between workspace restarts in the same browser.

### Additional resources

- CRW-5942

## 3.2. UNABLE TO RESOLVE PARENT DEVFILE WHEN USING SELF-SIGNED CERTS IN DISCONNECTED CLUSTERS

Previously, when you used self-signed certificates on an air-gapped cluster, starting a workspace that referenced a parent devfile by URI would fail with **x509: certificate signed by unknown authority** error. The defect has been fixed in this release and you can now reference a parent devfile in disconnected clusters.

### Additional resources

- CRW-6001

PROJECTS_ROOT environment variable being set incorrectly at workspace startup

Previously, PROJECTS_ROOT environment variable was set incorrectly to /**projects**/**projects** after workspace startup. The defect has been fixed in this release and the environment variable correctly points to the /**projects** directory.

### Additional resources

- CRW-6025

## 3.3. THE WORKSPACE STATUS CHANGED UNEXPECTEDLY TO 'STARTING'

Previously during a workspace startup, the status could have unexpectedly changed to 'Starting'

The defect has been fixed in this release, and status changes (except 'Failed' and 'Terminating') are ignored during workspace startup.

### Additional resources

- CRW-6281

## 3.4. THE DASHBOARD POD FREQUENTLY RESTARTS WITH EXITCODE: 137

Previously, the dashboard pod might have been frequently restarting with **exitCode: 137** due to a memory leak which has been fixed in this release.

**Additional resources**

- CRW-6292

## 3.5. DASHBOARD URL IS UNAVAILABLE FOR A FEW SECONDS WHEN POD IS DELETED AND RESTARTED

Previously, when a pod was restarted, the Dashboard URL could become unavailable for a short period of time during the operator update. The problem has been fixed in this release by adding the appropriate **LivenessProbe`and `ReadinessProbe** to the Gateway.

**Additional resources**

- CRW-6524

## 3.6. AFTER REVOKING THE OAUTH APPLICATION THE 'AUTHORIZATION' INDICATOR IS STILL ACTIVE IN THE 'USER PREFERENCES' DASHBOARD

The defect related to the misleading status of the 'Authorization' indicator after OAuth revocation from the Dashboard has been fixed in this release.

**Additional resources**

- CRW-6525

## 3.7. DASHBOARD PAGE IS BLANK IF DEVWORKSPACE IS MISSING CONTROLLER.DEVFILE.IO/CREATOR LABEL

Previously, if a DevWorkspace object was missing the **controller.devfile.io/creator** label the User Dashboard displayed a blank page. The defect has been fixed in this release.

**Additional resources**

- CRW-6526

## 3.8. PROJECTS ARE NOT CLONED AFTER RESTARTING THE WORKSPACE USING THE 'RESTART WORKSPACE FROM LOCAL DEVFILE' COMMAND

Before this release, extra projects added to the devfile from a workspace were not cloned if you restarted the workspace with the 'Restart Workspace from Local Devfile' command. With this release, the issue is fixed.

**Additional resources**

- CRW-6531

## 3.9. WORKSPACE ACTION MENU REMAINS OPEN

Before this release, the action menu items such as 'Open' and 'Stop Workspace' remained open on the User Dashboard after you clicked them. With this release, the issue is fixed.

**Additional resources**

- CRW-6534

## 3.10. ERROR WHEN STARTING A WORKSPACE WITH DF AND OVERRIDE.DEVFILEFILENAME URL PARAMETERS FROM THE DASHBOARD

The defect related to the errors during a workspace startup with **df**, **override.devfileFilename** parameters has been fixed in this release.

**Additional resources**

- CRW-6535

## 3.11. TOOLING CONTAINER $PATH IS OVERRIDDEN

Before this release, **process.env.PATH** was overridden by **userShellEnv.PATH** environment variable. With this release, the values of the **process.env.PATH** and **userShellEnv.PATH** environment variables are merged.

**Additional resources**

- CRW-6536

# CHAPTER 4. TECHNOLOGY PREVIEW

Technology Preview features provide early access to upcoming product innovations, enabling you to test functionality and provide feedback during the development process. However, these features are not fully supported under Red Hat Subscription Level Agreements, may not be functionally complete, and are not intended for production use. As Red Hat considers making future iterations of Technology Preview features generally available, we will attempt to resolve any issues that customers experience when using these features. See: Technology Preview support scope.

None.

# CHAPTER 5. DEPRECATED FUNCTIONALITIES

None.

# CHAPTER 6. REMOVED FUNCTIONALITIES

## 6.1. END OF DEVFILE V1 SUPPORT

With this release, devfile v1 is not supported anymore. Check Devfile.io documentation for supported devfile versions.

### Additional resources

- CRW-6538

# CHAPTER 7. KNOWN ISSUES

## 7.1. ANSIBLE LIGHTSPEED NOT CONNECTING TO ANSIBLE SERVER

There is a known issue with Ansible Lightspeed and connection to the Ansible server. If the OpenShift Dev Spaces environment is not under *.openshiftapps.com domain, Ansible Lightspeed can not connect to the Ansible server.

There is no workaround available.

**Additional resources**

- CRW-5691

## 7.2. FIPS COMPLIANCE UPDATE

There's a known issue with FIPS compliance that results in certain cryptographic modules not being FIPS-validated. Below is a list of requirements and limitations for using FIPS with OpenShift Dev Spaces:

### Required cluster and operator updates

Update your Red Hat OpenShift Container Platform installation to the latest z-stream update for 4.11, 4.12, or 4.13 as appropriate. If you do not already have FIPS enabled, you will need to uninstall and reinstall.

Once the cluster is up and running, install OpenShift Dev Spaces 3.7.1 (3.7-264) and verify that the latest DevWorkspace operator bundle 0.21.2 (0.21-7) or newer is also installed and updated. See https://catalog.redhat.com/software/containers/devworkspace/devworkspace-operator-bundle/60ec9f48744684587e2186a3

### Golang compiler in UDI image

The Universal Developer Image (UDI) container includes a golang compiler, which was built without the **CGO_ENABLED=1** flag. The check-payload scanner ( https://github.com/openshift/check-payload ) will throw an error, but this can be safely ignored provided that anything you build with this compiler sets the correct flag **CGO_ENABLED=1** and does NOT use **extldflags -static** or **-tags no_openssl**.

The resulting binaries can be scanned and should pass without error.

### Statically linked binaries

You can find statically linked binaries not related to cryptography in these two containers:

- code-rhel8
- idea-rhel8.

As they are not related to cryptography, they do not affect FIPS compliance.

### Helm support for FIPS

The UDI container includes the **helm** binary, which was not compiled with FIPS support. If you are in a FIPS environment do not use **helm**.

**Additional resources**

- CRW-4598

## 7.3. DEBUGGER DOES NOT WORK IN THE .NET SAMPLE

Currently, the debugger in Microsoft Visual Studio Code - Open Source does not work in the .NET sample.

### Workaround

- Use a different image from the following sources:
  - Custom UBI-9 based Dockerfile
  - devfile.yaml

### Additional resources

- CRW-3563

# CHAPTER 8. FREQUENTLY ASKED QUESTIONS

**Is it possible to deploy applications from OpenShift Dev Spaces to an OpenShift cluster?**

The user must log in to the OpenShift cluster from their running workspace using **oc login**.

**For best performance, what is the recommended storage to use for Persistent Volumes used with OpenShift Dev Spaces?**

Use block storage.

**Is it possible to deploy more than one OpenShift Dev Spaces instance on the same cluster?**

Only one OpenShift Dev Spaces instance can be deployed per cluster.

**Is it possible to install OpenShift Dev Spacesoffline (that is, disconnected from the internet)?**

See Installing Red Hat OpenShift Dev Spaces in restricted environments on OpenShift .

**Is it possible to use non-default certificates with OpenShift Dev Spaces?**

You can use self-signed or public certificates. See Importing untrusted TLS certificates.

**Is it possible to run multiple workspaces simultaneously?**

See Enabling users to run multiple workspaces simultaneously .