Redis Enterprise for Kubernetes
Release Notes 6.0.20-4 (May 2021)

The Redis Enterprise K8s 6.0.20-4 release is a major release on top of 6.0.8-20 providing support for the Redis Enterprise Software release 6.0.20-69 and includes several enhancements and bug fixes.

Overview

This release of the operator provides:

- New features
- Various bug fixes

To upgrade your deployment to this latest release, see "Upgrade a Redis Enterprise cluster (REC) on Kubernetes".

Images

This release includes the following container images:

- **Redis Enterprise**: redislabs/redis:6.0.20-69 or redislabs/redis:6.0.20-69.rhel7-openshift
- **Operator and Bootstrapper**: redislabs/operator:6.0.20-4
- **Services Rigger**: redislabs/k8s-controller:6.0.20-4 or redislabs/services-manager:6.0.20-4 (on the Red Hat registry)

New features

- Support for Openshift 4.7
- Support for Kubernetes 1.20

New preview features

- Hashicorp Vault integration - REC secret
- Hashicorp Vault integration - REDB secrets

Important fixes

- Fixed upgrade issue with custom container repositories specifying port numbers (RED-53192)
- REDB controller no longer performs reconciliation until Redis Enterprise software version complies with operator (RED-53194)
- Removed unused node.js package from Services Rigger image (RED-53536)
- Fixed operator crash on change of uiServiceType (RED-54621)
- Avoid excessive logging within RS pod (envoy_access.log) (RED-55525)
Known limitations

OpenShift 4.7 - v6.0.20-4 cannot be deployed by OLM

6.0.20-4 does not appear in OLM. Workaround - deploy manually. A future maintenance release will address this.

Hashicorp Vault integration - no support for Gesher (RED-55080)

There is no workaround at this time

Nodes down indefinitely after the redis-enterprise-node container of a REC pod is restarted (53042)

In some cases where the Redis Enterprise Cluster container in the Redis Enterprise Cluster(REC) pod is restarted, the REC node remains down. Workaround: restart the pod, while ensuring the majority of REC nodes are available.

CrashLoopBackOff causes cluster recovery to be incomplete (RED33713)

When a pod’s status is CrashLoopBackOff and we run the cluster recovery, the process will not complete. The workaround is to delete the crashing pods manually. The recovery process will then continue.

Long cluster names cause routes to be rejected (RED25871)

A cluster name longer than 20 characters will result in a rejected route configuration because the host part of the domain name will exceed 63 characters. The workaround is to limit cluster name to 20 characters or fewer.

Cluster CR (REC) errors are not reported after invalid updates (RED25542)

A cluster CR specification error is not reported if two or more invalid CR resources are updated in sequence.

An unreachable cluster has status running (RED32805)

When a cluster is in an unreachable state, the state is still running instead of being reported as an error.

Readiness probe incorrect on failures (RED39300)

STS Readiness probe does not mark a node as “not ready” when running rladmin status on the node fails.

Role missing on replica sets (RED39002)

The redis-enterprise-operator role is missing permission on replica sets.

Private registries are not supported on OpenShift 3.11 (RED38579)

Openshift 3.11 does not support DockerHub private registries. This is a known OpenShift issue.

Internal DNS and Kubernetes DNS may have conflicts (RED37462)

DNS conflicts are possible between the cluster mdns_server and the K8s DNS. This only impacts DNS resolution from within cluster nodes for Kubernetes DNS names.

5.4.10 negatively impacts 5.4.6 (RED37233)

Kubernetes-based 5.4.10 deployments seem to negatively impact existing 5.4.6 deployments that share a Kubernetes cluster.

Node CPU usage is reported instead of pod CPU usage (RED36884)
In Kubernetes, the node CPU usage we report on is the usage of the Kubernetes worker node hosting the REC pod.

**Clusters must be named “rec” in OLM-based deployments (RED39825)**

In OLM-deployed operators, the deployment of the cluster will fail if the name is not "rec". When the operator is deployed via the OLM, the security context constraints (scc) is bound to a specific service account name (i.e., “rec”). The workaround is to name the cluster "rec".

**Master pod label in Rancher (RED42896)**

The master pod is not always labeled in Rancher.

**REC clusters fail to start on Kubernetes clusters with unsynchronized clocks (RED47254)**

When REC clusters are deployed on Kubernetes clusters with unsynchronized clocks, the REC cluster does not start correctly. The fix is to use NTP to synchronize the underlying K8s nodes.

**Deleting an OpenShift project with an REC deployed may hang (RED47192)**

When a REC cluster is deployed in a project (namespace) and has REDB resources, the REDB resources must be deleted first before the REC can be deleted. As such, until the REDB resources are deleted, the project deletion will hang. The fix is to delete the REDB resources first and the REC second. Afterwards, you may delete the project.

**REC extraLabels are not applied to PVCs on K8s versions 1.15 or older (RED51921)**

In K8s 1.15 or older, the PVC labels come from the match selectors and not the PVC templates. As such, these versions cannot support PVC labels. If this feature is required, the only fix is to upgrade the K8s cluster to a newer version.

**Compatibility Notes**

- OpenShift 4.7 and Rancher/kOps 1.20 are now supported
- OpenShift 4.1, 4.2, 4.3 (previously deprecated) are no longer supported
- GKE K8s version 1.14 (previously deprecated) is no longer supported
- kOps (upstream K8s) 1.13, 1.14 (previously deprecated) are no longer supported

**Deprecation notice**

- OpenShift 4.4 (no longer supported by Red Hat) is deprecated
- GKE K8s versions 1.15, 1.16 (no longer supported by Google) are deprecated
- kOps (upstream K8s) 1.15 is deprecated

**Updated:** January 11, 2022