Redis Enterprise for Kubernetes
Release Notes 6.0.6-6 (June 2020)

Redis Enterprise for Kubernetes 6.0.6-6 is now available. This release includes Redis Enterprise (RS) version 6.0 and introduces new features, improvements, and bug fixes.

Overview

This release of the operator provides:

- Support for the Redis Enterprise Software release 6.0
- Includes the new database and admission controllers
- Various improvements and bug fixes

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

Note: If you are running Active-Active (CRBD) databases on a previous release, do not upgrade to 6.0.6-6 at this time. There is an issue with the upgrade process that is currently being investigated (RED43635). For more information and support please contact Redis support.

Images

- **Redis Enterprise** - redislabs/redis:6.0.6-6 or redislabs/redis:6.0.6-6.rhel7-openshift
- **Operator** - redislabs/operator:6.0.6-6 or redislabs/operator:6.0.6-6.rhel7
- **Services Rigger** - redislabs/k8s-controller:6.0.6-6 or redislabs/k8s-controller:6.0.6-6.rhel7

New features and improvements

- Database controller - A new database controller in the operator provides the ability to create and manage databases on a Redis Enterprise cluster via a custom resource (RED36516).
- Admission controller - A new admission controller in the operator provides validation of database custom resources (RED36458).
- Pod tolerations - Support for specifying Redis Enterprise cluster node pod tolerations of node taints has been added to the cluster CR (see podTolerations) (RED33069).
- Pod annotations - Support for specifying Redis Enterprise cluster node pod annotations has been added to the cluster CR (see podAnnotations) (RED35613).
- Kubernetes versions - Support for Kubernetes 1.17 was added and versions 1.9 and 1.10 (previously deprecated) are no longer supported (RED41049).
- Improved OLM Experience - The overall user experience and documentation in the OLM (OperatorHub) has been improved (RED37008).
- Resource limits - Resource limits have been added to the recommended operator configuration (RED39572).
- LoadBalancer service type added - The LoadBalancer value has been added to the databaseServiceType option in servicesRiggerSpec (RED43215):
Important fixes

- Service creation failure causes cluster setup failure (RED37197)
- UI service update failure (RED37198)
- Error shown in OLM deployment: "The field status.state is invalid" (RED40278)
- OLM: StatefulSet not listed as an object owned by the Redis Enterprise Cluster (RED39296)
- Setting extraLabels in the cluster CR did not label pods on OpenShift (RED39763)
- log_collector failed to get the pods logs when a namespace wasn’t given (RED39292)
- Role and RoleBinding created or updated using an existing ServiceAccount in REC spec (RED42912)

Known limitations

- CrashLoopBackOff pod status and cluster recovery - When a pod status is CrashLoopBackOff and we run the cluster recovery, the process will not complete. The solution is to delete the crashing pods manually and recovery process will continue (RED33713).

- Active-Active (CRDB) - limitation on cluster name length - A cluster name longer than 20 characters will result in a rejected route configuration as the host part of the domain name exceeds 63 characters. Cluster names must be limited to 20 characters or less (RED25871).

- Active-Active (CRDB) service broker cleanup - The service broker doesn’t clean up database service bindings in case of failures. These bindings must be removed manually (RED25825).

- Service broker deployment error - The service broker deployment results in an error when two types of service naming schemes are set. Choosing one of the methods will resolve this error (redis-port is the recommended default) (RED25547).

- Cluster spec invalid errors not reported - A cluster CR specification error is not reported if two or more invalid CR resources are updated in sequence (RED25542).

- Unreachable cluster does not produce an error - When a cluster is in an unreachable state the state is still running instead of being reported as an error (RED32805).

- Readiness probe ignores radmin failure - STS Readiness probe doesn’t mark a node as not ready when radmin status nodes fails (RED39300).

- Missing permission for role - The redis-enterprise-operator role is missing permission on replicasets (RED39002).

- Openshift 3.11 doesn’t support DockerHub private registry - Openshift 3.11 doesn’t support DockerHub private registry. This is a known OpenShift issue and not addressable by the Redis Enterprise operator (RED38579).

- Possible DNS conflicts within cluster nodes - DNS conflicts are possible between the cluster mdns_server and the K8s DNS. This only impacts DNS resolution from within cluster node and while using the full fqdn *.cluster.local (RED37462).

- Coexistence of 5.4.10 and 5.4.6 clusters - K8s clusters with Redis Enterprise 5.4.6 clusters are negatively affected by installing a Redis Enterprise 5.4.10 cluster due to changes in CRD (CustomeResourceDefinition) (RED37233).

- Redis Enterprise CPU utilization metric reports at K8s node level rather than at pod level - In Kubernetes, the node CPU usage we report on is the usage of the K8s worker node hosting the REC pod (RED-36884). Pod resource utilization should be measured by K8s-native means rather than through the application.

- Cluster name is limited on OpenShift via OLM - 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"). Naming the cluster "rec" resolves the issue (RED39825).

Coming Soon

The following lists features, fixes and changes the Redis team is currently investing in:

- Redis Enterprise and Kubernetes Container Artifacts - starting from the next release of Redis Enterprise for K8s, new container artifacts will be published using different base images:
  - **Redis Enterprise** - A UBI (RHEL 7) base image will replace the Ubuntu and the RHEL7 base images
  - **Operator** - An Image built from scratch containing the Golang executable will replace the Ubuntu and the RHEL7 base images
  - **Services Rigger** - A UBI (RHEL 7) base image will replace the Ubuntu and the RHEL7 base images
- Deprecation notice - the service broker solution is deprecated and will not be supported starting from the next release of the Redis Enterprise Operator for Kubernetes.
- Additional Redis Enterprise Database configuration options in the Database Controller - We're currently investing in the following additional capabilities of the Database Controller:
  - Support for loading database modules
  - Support for setting up Alerts, expressed as K8s events
  - Support for configuring database backup options
  - Support for Kubernetes 1.18
  - Support for Rancher K8s distribution

Updated: January 11, 2022