FUJITSU

FUJITSU Enterprise Postgres 12 for Kubernetes

Manual Set

October 2022

FUJITSU

FUJITSU Enterprise Postgres 12 for Kubernetes

Release Notes

Linux

J2UL-NOTE-10ENZ0(00)
October 2022

Preface

Purpose of this document

This document provides release information for FUJITSU Enterprise Postgres for Kubernetes.

Structure of this document

This document is structured as follows:

Chapter 1 New Features and Improvements

Explains the new features and improvements in this version.

Abbreviations

The following abbreviations are used in this manual:

Full Name	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes FUJITSU Software Enterprise Postgres	FEP or FUJITSU Enterprise Postgres
Custom Resource	CR
Universal Base Image	UBI
OpenShift Container Platform	OCP
Mutual TLS	MTLS

Abbreviations of manual titles

The following abbreviations are used in this manual as manual titles:

Full Manual Title	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes Release Notes	Release Notes
FUJITSU Software Enterprise Postgres for Kubernetes Overview	Overview
FUJITSU Software Enterprise Postgres for Kubernetes User's Guide	User's Guide
FUJITSU Software Enterprise Postgres for Kubernetes Reference	Reference

Trademarks

- Linux is a registered trademark or trademark of Mr. Linus Torvalds in the U.S. and other countries.
- Red Hat and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.
- S/390 is a registered trademark of International Business Machines Corporation in the United States or other countries or both.

Other product and company names mentioned in this manual are the trademarks or registered trademarks of their respective owners.

Export restrictions

If this document is to be exported or provided overseas, confirm legal requirements for the Foreign Exchange and Foreign Trade Act as well as other laws and regulations, including U.S. Export Administration Regulations, and follow the required procedures.

Issue date and version

```
Edition 10.0: October 2022
Edition  9.0: September 2022
Edition  8.0: August 2022
Edition  7.0: July 2022
Edition  6.0: June 2022
Edition  5.0: May 2022
Edition  4.0: March 2022
Edition  3.0: December 2021
Edition  2.0: November 2021
Edition  1.0: August 2021
```

Copyright

Contents

Chapter 1 New Features and Improvements

This chapter explains FUJITSU Enterprise Postgres for Kubernetes new features and improvements added in this version.

Table 1.1 New features and improvements

Version and level	Classification	Feature
Operator image tag:v2.2.11 Container image tag:ubi8-12-2.10	OSS	PostgreSQL Rebase
	Platform enhancement	Patching the Container's Base OS Image
Operator image tag:v2.2.10 Container image tag:ubi8-12-2.9	OSS	Patching the PostgreSQL JDBC Driver
	Platform enhancement	Patching the Container's Base OS Image
Operator image tag:v2.2.9 Container image tag:ubi8-12-2.8	Platform enhancement	Patching the Container's Base OS Image
Operator image tag:v2.2.8 Container image tag:ubi8-12-2.7	Platform enhancement	Patching the Container's Base OS Image
Operator image tag:v2.2.7 Container image tag:ubi8-12-2.6	OSS	PostgreSQL Rebase
	Platform enhancement	Additional OCP Support
Operator image tag:v2.2.6 Container image tag:ubi8-12-2.5	Platform enhancement	Patching the Container's Base OS Image
Operator image tag:v2.2.5 Container image tag:ubi8-12-2.4	OSS	PostgreSQL Rebase
	Platform enhancement	Additional OCP Support
Operator image tag:v2.2.4 Container image tag:ubi8-12-2.3	OSS	PostgreSQL Rebase
Operator image tag:v2.2.3 Container image tag:ubi8-12-2.2	Platform enhancement	Additional OCP Support
Operator image tag:v2.2.0 Container image tag:ubi8-12-2.0	OSS	PostgreSQL Rebase
	Platform enhancement	Additional OCP Support
		Modifying Container's Base OS Image
	Operation	Additional Custom Annotations/Labels
		Additional Installation Support
		Additional Feature of Secure Connection
		Additional Storage Support

1.1 Features Added in v2.2.11

This section explains the improvement in FUJITSU Enterprise Postgres for Kubernetes v2.2.11.

1.1.1 OSS

This section explains the improvement related to OSS:

- PostgreSQL rebase

1.1.1.1 PostgreSQL Rebase

The PostgreSQL version that FUJITSU Enterprise Postgres is based on is 12.12.

Refer to "A OSS Supported by FUJITSU Enterprise Postgres for Kubernetes" in the Overview.

1.1.2 Platform Enhancement

This section explains the improvement related to platform enhancement:

- Patching the Container's Base OS image

1.1.2.1 Patching the Container's Base OS Image

Patched the Container's Base OS image.

1.2 Features Added in v2.2.10

This section explains the improvement in FUJITSU Enterprise Postgres for Kubernetes v2.2.10.

1.2.1 OSS

This section explains the improvement related to OSS:

- Patching the PostgreSQL JDBC driver

1.2.1.1 Patching the PostgreSQL JDBC Driver

Patched the PostgreSQL JDBC driver.

1.2.2 Platform Enhancement

This section explains the improvement related to platform enhancement:

- Patching the Container's Base OS image

1.2.2.1 Patching the Container's Base OS Image

Patched the Container's Base OS image.

1.2.3 OSS

This section explains the improvement related to OSS:

- Patching the PostgreSQL JDBC driver

1.2.3.1 Patching the PostgreSQL JDBC Driver

Patched the PostgreSQL JDBC driver.

1.3 Features Added in v2.2.9

This section explains the improvement in FUJITSU Enterprise Postgres for Kubernetes v2.2.9.

1.3.1 Platform Enhancement

This section explains the improvement related to platform enhancement:

- Patching the Container's Base OS image

1.3.1.1 Patching the Container's Base OS Image

Patched the Container's Base OS image.

1.4 Features Added in v2.2.8

This section explains the improvement in FUJITSU Enterprise Postgres for Kubernetes v2.2.8.

1.4.1 Platform Enhancement

This section explains the improvement related to platform enhancement:

- Patching the Container's Base OS image

1.4.1.1 Patching the Container's Base OS Image

Patched the Container's Base OS image.

1.5 Features Added in v2.2.7

This section explains new features and improvements in FUJITSU Enterprise Postgres for Kubernetes v2.2.7.

1.5.1 OSS

This section explains the new feature related to OSS:

- PostgreSQL rebase

1.5.1.1 PostgreSQL Rebase

The PostgreSQL version that FUJITSU Enterprise Postgres is based on is 12.11.

See

Refer to "A OSS Supported by FUJITSU Enterprise Postgres for Kubernetes" in the Overview.

1.5.2 Platform Enhancement

This section explains the new features related to platform enhancement:

- Additional OCP support

1.5.2.1 Additional OCP Support

The following additional OCP is supported:

- OCP 4.10

Refer to "Supported Platform" in the User's Guide for details.

1.6 Features Added in v2.2.6

This section explains the improvement in FUJITSU Enterprise Postgres for Kubernetes v2.2.6.

1.6.1 Platform Enhancement

This section explains the improvement related to platform enhancement:

- Patching the Container's Base OS image

1.6.1.1 Patching the Container's Base OS Image

Patched the Container's Base OS image.

1.7 Features Added in v2.2.5

This section explains new features and improvements in FUJITSU Enterprise Postgres for Kubernetes v2.2.5.

1.7.1 OSS

This section explains the new feature related to OSS:

- PostgreSQL rebase

1.7.1.1 PostgreSQL Rebase

The PostgreSQL version that FUJITSU Enterprise Postgres is based on is 12.10.

Refer to "A OSS Supported by FUJITSU Enterprise Postgres for Kubernetes" in the Overview.

1.7.2 Platform Enhancement

This section explains the new features related to platform enhancement:

- Additional OCP support

1.7.2.1 Additional OCP Support

The following additional OCP is supported:

- OCP 4.9

Refer to "Supported Platform" in the User's Guide for details.

1.8 Features Added in v2.2.4

This section explains new features and improvements in FUJITSU Enterprise Postgres for Kubernetes v2.2.4.

1.8.1 OSS

This section explains the new feature related to OSS:

- PostgreSQL rebase

1.8.1.1 PostgreSQL Rebase

The PostgreSQL version that FUJITSU Enterprise Postgres is based on is 12.9.

Refer to "A OSS Supported by FUJITSU Enterprise Postgres for Kubernetes" in the Overview.

1.9 Features Added in v2.2.3

This section explains new features and improvements in FUJITSU Enterprise Postgres for Kubernetes v2.2.3.

1.9.1 Platform Enhancement

This section explains the new features related to platform enhancement:

- Additional OCP support

1.9.1.1 Additional OCP Support

The following additional OCP is supported:

- OCP 4.7
- OCP 4.8

Refer to "Supported Platform" in the User's Guide for details.

1.10 Features Added in v2.2.0

This section explains new features and improvements in FUJITSU Enterprise Postgres for Kubernetes v2.2.0.

1.10.1 OSS

This section explains the new feature related to OSS:

- PostgreSQL rebase

1.10.1.1 PostgreSQL Rebase

The PostgreSQL version that FUJITSU Enterprise Postgres is based on is 12.5.

See

Refer to "A OSS Supported by FUJITSU Enterprise Postgres for Kubernetes" in the Overview.

1.10.2 Platform Enhancement

This section explains the new features related to platform enhancement:

- Additional OCP support
- Modifying container's base OS image

1.10.2.1 Additional OCP Support

The following additional OCP is supported:

- OCP 4.6

See

Refer to "Supported Platform" in the User's Guide for details.

1.10.2.2 Modifying Container's Base OS Image

Changed container's base OS image to the following image:

- Red Hat UBI minimal 8

See

Refer to "Components Embedded" in the User's Guide for details.

1.10.3 Operation

This section explains the new feature related to OSS:

- Additional custom annotations/labels
- Additional installation support
- Additional feature of secure connection
- Additional storage support

1.10.3.1 Additional Custom Annotations/Labels

Custom annotations/labels is added to parameter of FEPCluster CR.

See

Refer to "Adding Custom Annotations to FEPCluster Pods using Operator" in the User's Guide.

1.10.3.2 Airgap Installation Support

Additional support for the following installation methods:

- Airgap(offline) instllation

1.10.3.3 Additional Feature of Secure Connection

The following additional feature is supported:

- MTLS

See

Refer to "Configure FEP to Perform MTLS" in the User's Guide for details.

1.10.3.4 Additional Storage Support

The following additional storage is supported:

- IBM spectrum virtualize

FUJITSU

FUJITSU Enterprise Postgres 12 for Kubernetes

Overview

Linux

J2UL-OVER-06ENZ0(00)
October 2022

Preface

Purpose of this document

This document explains the FUJITSU Enterprise Postgres for Kubernetes concepts to those who are to operate databases using it.

This document explains the features of FUJITSU Enterprise Postgres for Kubernetes.

Intended readers

This document is intended for people who are:

- Considering installing FUJITSU Enterprise Postgres for Kubernetes
- Using FUJITSU Enterprise Postgres for Kubernetes for the first time
- Wanting to learn about the concept of FUJITSU Enterprise Postgres for Kubernetes
- Wanting to see a functional overview of FUJITSU Enterprise Postgres for Kubernetes

Readers of this document are also assumed to have general knowledge of:

- Linux
- Kubernetes
- Containers
- Operators

Structure of this document

This document is structured as follows:

Abbreviations

The following abbreviations are used in this manual:

Full Name	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes FUJITSU Software Enterprise Postgres	FEP or FUJITSU Enterprise Postgres
Custom Resource	CR
Custom Resource Definition	CRD
Persistent Volume	PV

Abbreviations of manual titles

The following abbreviations are used in this manual as manual titles:

Full Manual Title	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes User's Guide	User's Guide

Trademarks

- Linux is a registered trademark or trademark of Mr. Linus Torvalds in the U.S. and other countries.
- Red Hat and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.
- S/390 is a registered trademark of International Business Machines Corporation ("IBM") in the U.S. and other countries.

Other product and company names mentioned in this manual are the trademarks or registered trademarks of their respective owners.

Export restrictions

If this document is to be exported or provided overseas, confirm legal requirements for the Foreign Exchange and Foreign Trade Act as well as other laws and regulations, including U.S. Export Administration Regulations, and follow the required procedures.

Issue date and version


```
Edition 6.0: October 2022
Edition 5.0: June 2022
Edition 4.0: March 2022
Edition 3.0: December 2021
Edition 2.0: April 2021
Edition 1.0: March 2021
```


Copyright

Contents

Chapter 1 Know about the Product

This chapter explains the features of FUJITSU Enterprise Postgres for Kubernetes.

1.1 What is FUJITSU Software Enterprise Postgres for Kubernetes?

FUJITSU Software Enterprise Postgres for Kubernetes provides an enterprise platform for installing and managing your FUJITSU Enterprise Postgres 12 on OpenShift Container Platform.

There are multiple components in the solution.

FEP operator: Manages the lifecycle of FEP server container, including deployment, configuration update, backup and recovery of FEP database.

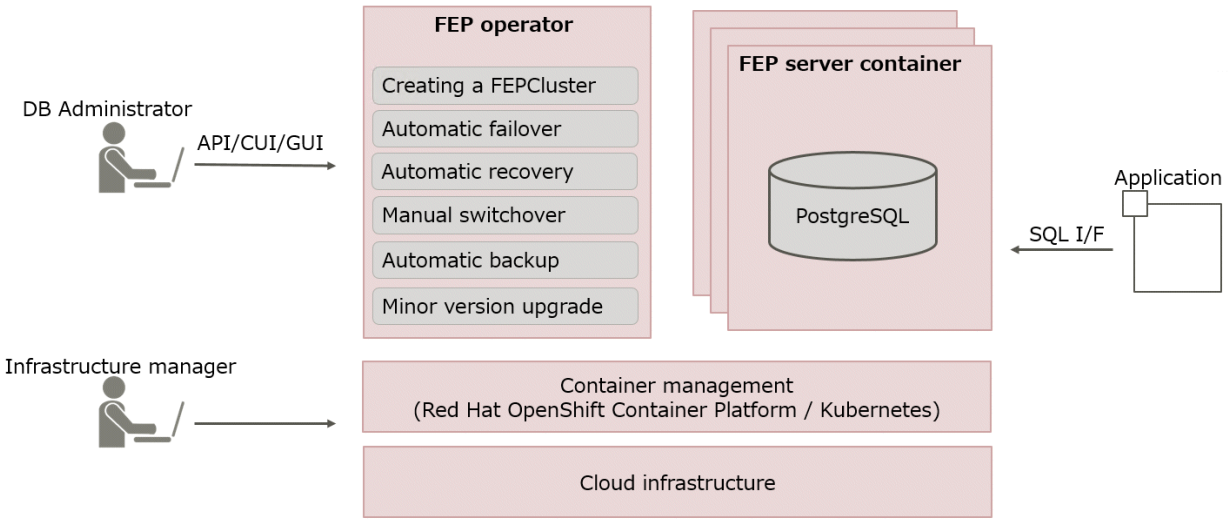
FEP server container: Contains the FEP server software to run the Postgres engine.

FEP backup container: Contains the FEP server software to perform scheduled backup operations.

FEP restore container: Contains the FEP server software to perform the restore operation.

FEP pgpool2 container: Contains the FEP server software to use Pgpool-II to provide load balancing and connection pooling.

Up and running in minutes, the operator provides the features required to maximise the benefits of this enterprise PostgreSQL solution.



This operator will deploy a standalone as well as highly available FUJITSU Enterprise Postgres cluster with pre-defined configuration to get started with small workload. User can adjust the configuration parameters at the time of deployment and after to make the instance suitable for the workload.

As the name implies, the FEP server container is intended to incorporate the FUJITSU Enterprise Postgres server component.

In principle, a running FEP server container is considered as equivalent to a FUJITSU Enterprise Postgres Server instance.

1.2 Operator Features

This product provides operator services to automate the construction and operation of databases on the customer's container management infrastructure. The features of the operator are as follows:

- Cluster Deployment
  - Creating a FEPCluster
  - Creating a FEP Pgpool2 Container

- Highly Available Feature
  - Automatic Failover
  - Automatic Recovery
  - Manual Switchover
- Backup Recovery
  - Automatic Backup
  - Point-in-time Recovery
- Configuration Change
  - Parameter Change
  - Resource Change
- Minor Version Upgrade
- FEP Features

1.2.1 Cluster Deployment

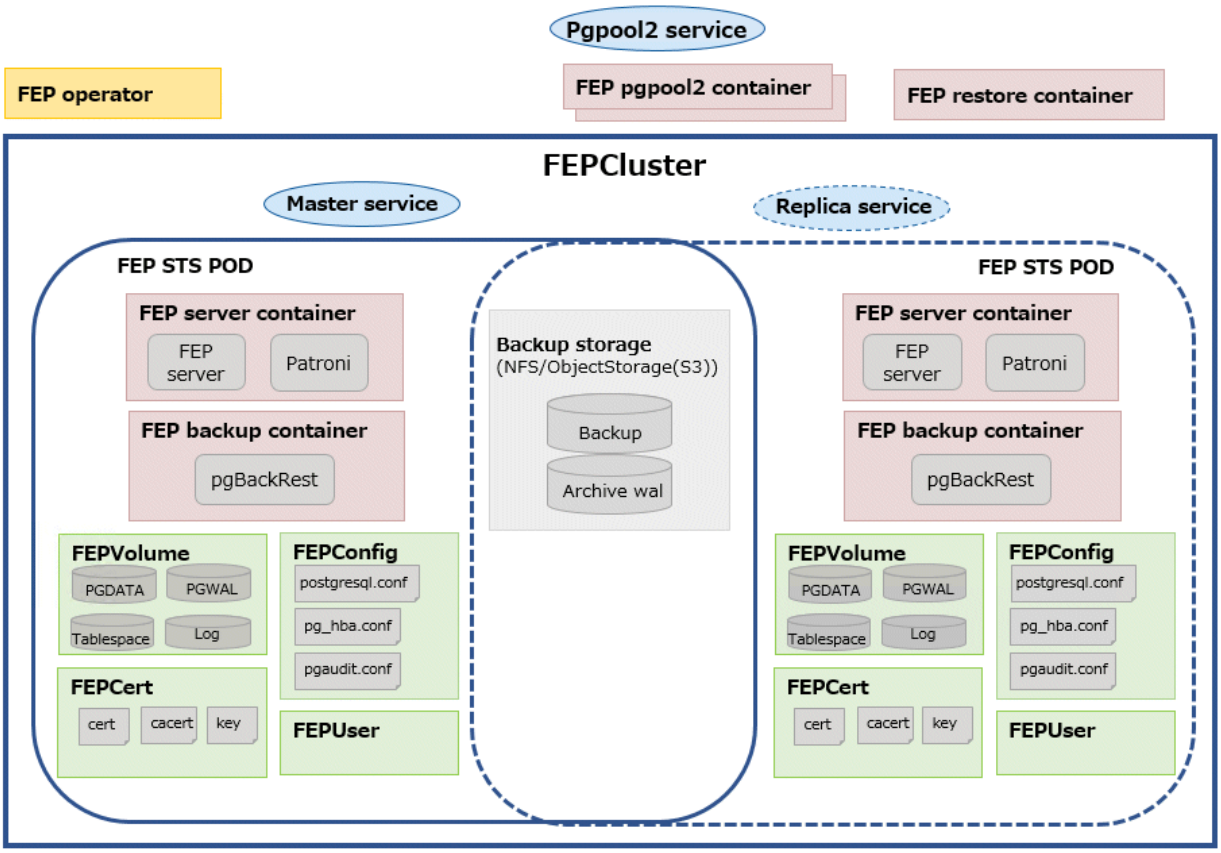
1.2.1.1 Creating a FEPCluster

Users can instruct the operator to build a system that includes the provisioning of containers and volumes with FEP installed, and network resources. The resulting system is called a FEPCluster. The FEPCluster can be created a single master server or multi-servers with one master and two replicas. You can choose between synchronous and asynchronous replication replica servers. The default is synchronous replication.

FEPCluster is composed of the following components:

- FEP server container
  - FEP server
  - Patroni
- FEP backup container
- CR FEPVolume for volumes
- CR FEPUser for database users
- CR FEPConfig for Postgres configuration
- CR FEPCert for secrets such as TLS certificate, keystore passphrase

The Below diagram depicts a FEPCluster with one Master and one Replica POD.



1.2.1.2 Creating a FEP Pgpool2 Container

Users can deploy Pgpool-II for load balancing and connection pooling with FEP pgpool2 container.

Users can deploy multiple FEP pgpool2 Pods in a single deployment to increase availability.

1.2.2 Highly Available Feature

1.2.2.1 Automatic Failover

When an error is detected in the container or POD of the master server, the cluster will perform an automatic failover by promoting one of the replicas to become the new master, and the connection destination of the database is switched. The database connection is broken, but you can reconnect by establishing a connection from the application again.

1.2.2.2 Automatic Recovery

If an error occurs on the master server and an automatic failover occurs, the POD or container of the failed old master server is automatically restarted and reincorporated into the cluster as a replica server.

If a replica server fails, it automatically restarts and rejoins the cluster as a replica server.

1.2.2.3 Manual Switchover

You can manually switch any replica server to the master server. In this case, the original master server becomes the replica server.

1.2.3 Backup Recovery

1.2.3.1 Automatic Backup

Users can set an arbitrary schedule for automatic backup. The backup type can be a full backup or an incremental backup. You can back up the database to shared storage such as NFS persistent volume or AWS S3 compatible storage. Backups can be automatically deleted by setting a retention period of your choice.

1.2.3.2 Point-in-time Recovery

Allows point-in-time recovery from automated backup data to restore the cluster. You can choose between restoring data to an existing cluster and a new cluster. You can also choose to restore to the most recent data or to any time you specify.

1.2.4 Configuration Change

1.2.4.1 Parameter Change

You can change the parameters that make up the FEP. PostgreSQL provides two types of parameters: those that take effect immediately, and those that take effect after restarting FEP server process.

- postgresql.conf
- pg_hba.conf
- pgaudit.conf

For parameters that take effect immediately, operator will apply the change to all FEP Pods and reload the FEP server process automatically. There is no outage on the cluster.

For parameters that take effect after restarting FEP server process, operator will update the configuration files on all FEP Pods. However, users have to initiate a manual restart of FEP process on all the FEP Pods using the FEPAction CR. There is a momentary outage on the cluster and the users should perform this action at a time that has least disruption to the service.

1.2.4.2 Resource Change

You can change the amount of CPU and memory resources allocated to FEP server containers, FEP backup containers, or FEP pgpool2 containers by changing the FEPCluster CR. The operator will apply the change to the Statefulset. However, the users have to perform a manual recreation of statefulsets for the new resource allocation to take effect.

Changing resource allocation will not take effect immediately. The users have to recreate statefulsets for new resource allocation to take effect. There is a momentary outage on the cluster and the users should perform this action at a time that has least disruption to the service.

1.2.5 Minor Version Upgrade

1.2.5.1 Minor Version Upgrade

New and patched FEP releases are made available as new container image. When the latest container image is provided, the user can perform a minor version upgrade by changing the FEPCluster CR. The operator will perform a rolling update to enable the minor version upgrade with minimal system disruption.

Note

The minor version upgrade will take effect immediately. There is a momentary outage on the cluster and the users should perform this action at a time that has least disruption to the service.

1.2.6 FEP Features

1.2.6.1 Scope of FEP Feature Support

Note

These features also require the FEP Client ISO.

The FEPCluster that is created supports the following features in addition to the PostgreSQL features of OSS. Details of each feature can be found in the FEP documentation.

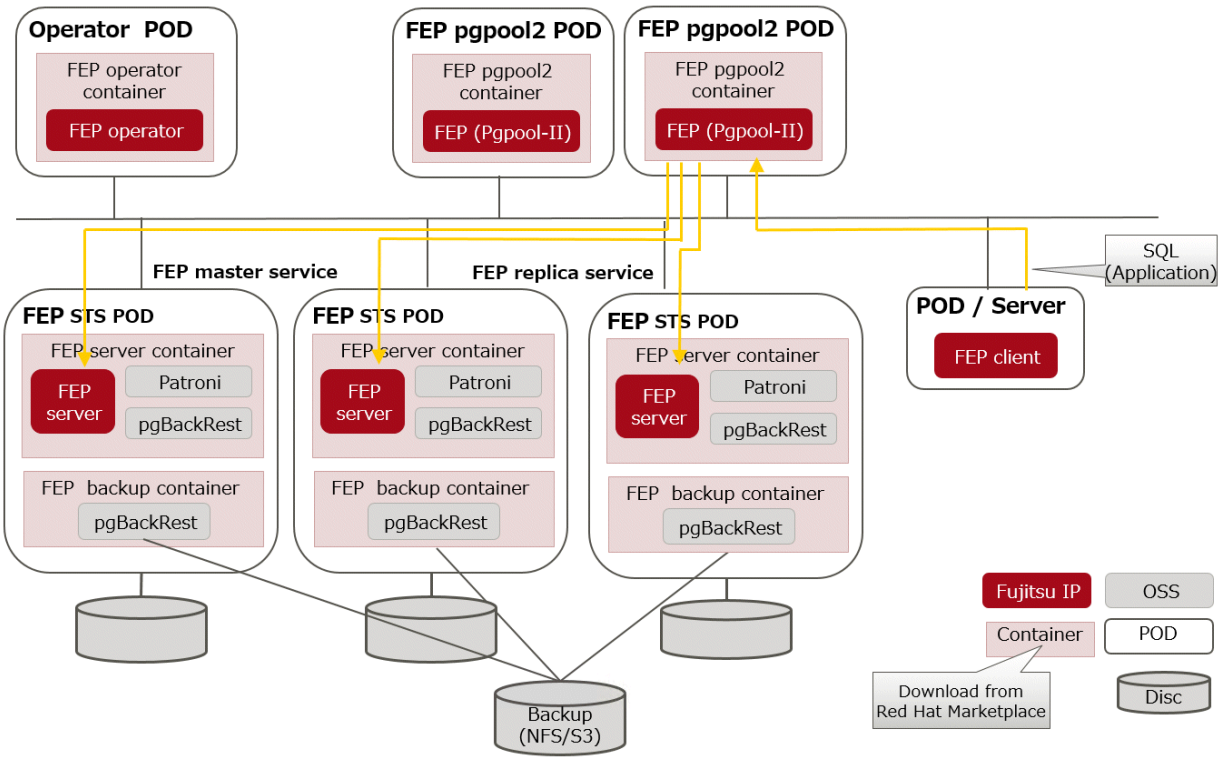
Category	Feature
Operation	pgAdmin
	Global Meta Cache
Data masking	Transparent Data Encryption
	Audit Log
	Secrecy
High Performance	In-memory feature
	High-speed data load
Application Interface	Java Integration
	ODBC Integration
	.NET Framework Integration
	Embedded SQL Integration (C language)
	Embedded SQL Integration (COBOL)

1.3 Operator System Configuration

The basic relationships among POD, containers and services are as follows.

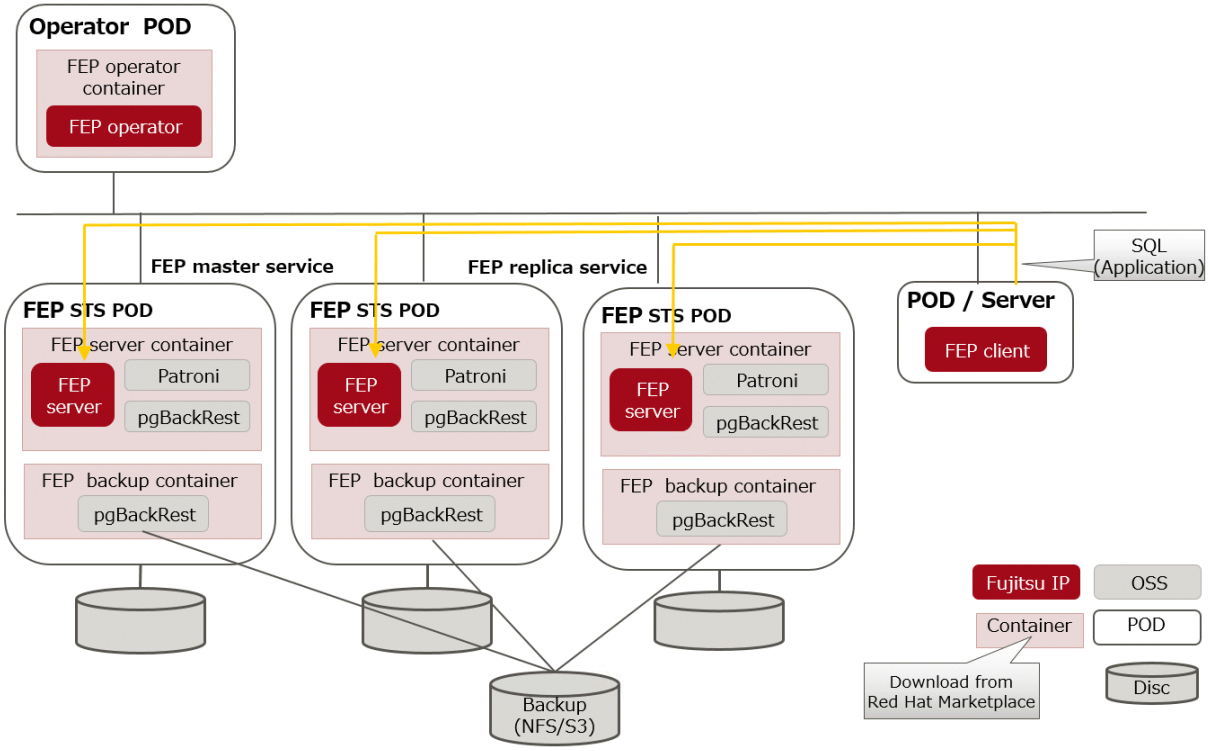
Example) Deployment with Pgpool-II

In this deployment scenario, Pgpool-II is used to provide connection pooling and load balancing. End user application will point its connection to Pgpool service. Depending on the transaction type, Pgpool will forward the connection to either the Master Pod or the Replica Pod. If a failover/switchover occurs, the FEP pgpool2 will direct traffic to the new FEP master Pod. This is transparent to the end user application.



Example) Deployment without Pgpool-II

Users can also run applications such as SQL directly against the FEPCluster without configuring Pgpool-II. In this deployment scenario, end user application will point its connection to the FEP master service. If a failover/switchover occurs, the FEP master service will point to the new FEP master Pod automatically. The end user application will experience a disconnection. When it re-establishes the connection, it will be connected to the new FEP master Pod. There is no need to reconfigure the application connection string.



Chapter 2 Know What it does

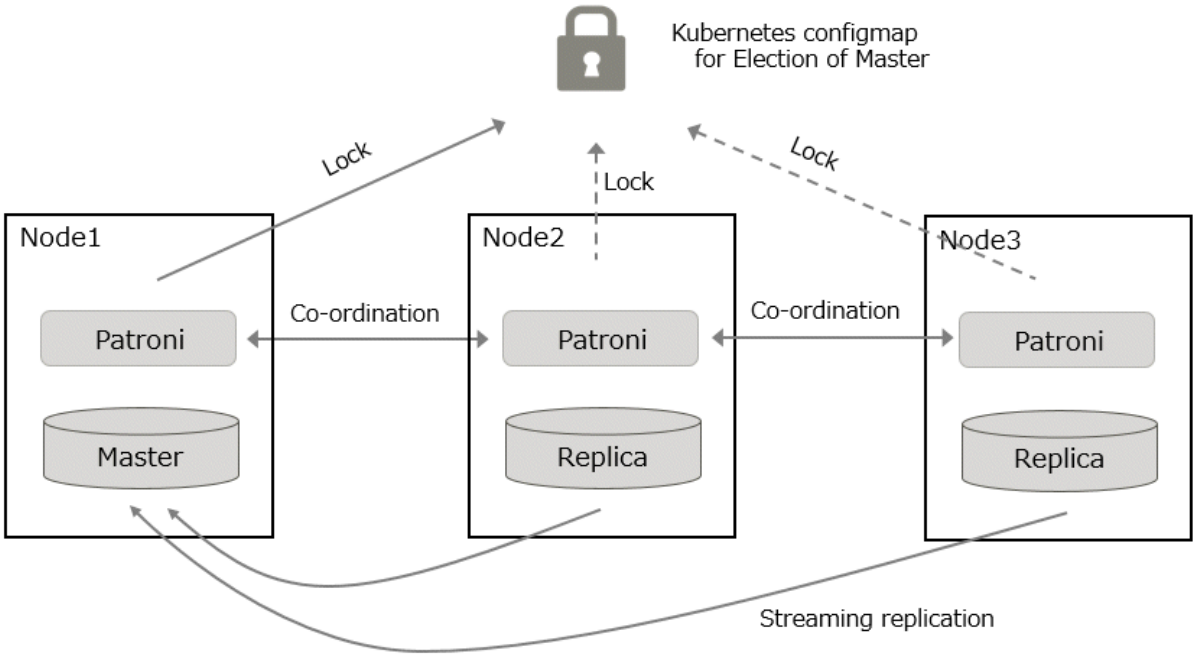
This chapter explains what you need to do.

2.1 Deployment

FEP operator is responsible for the lifecycle of FEPCluster. The operator will deploy a HA FEPCluster, together with all the associated containers such as backup container.

2.2 High Availability (Automatic failover and recovery)

The high availability and failover management of FEP is provided by Patroni. Both Patroni and FEP will be installed on the same container image. Patroni will then initialize and start an FEP instance. Patroni will then acquire a lock on a shared resource. In our case, it is a Kubernetes configmap. Whichever POD that can acquire the lock will become the Master. When subsequent FEP server container starts, Patroni will initialize that POD as a Replica with streaming replication.



If Patroni detects a failure in the cluster, either because the Postgres process crashed or the container where Postgres is running dies, Patroni will initiate a failover automatically.

2.3 Configuration Change

Traditionally, changing FEP configurations such as postgresql.conf, pg_hba.conf, TLS certificates and keystore passphrase will require a redeployment of FEP server container. That causes an outage in a Highly Available environment.

A new CRD FEPConfig is defined to encapsulate those configurations. The operator will monitor the CR with this CRD definition and perform action accordingly to minimize outages. For example, operator will reload FEP daemon, instead of redeploying the FEP server container when a reloadable postgresql.conf parameter is changed. If a parameter change requires restart of FEP (e.g. max_connections), the operator will update the configuration file but defer the restart. End user can follow a defined procedure to restart the cluster manually at a scheduled maintenance time.

2.4 Minor Version Upgrade

FEP version Minor upgrade is done by updating the Custom Resource with a new FEP image name. The POD will be redeployed with new image in a controlled manner. First, replica servers are upgraded, restarted and waited to be ready, one server at a time. When all replicas are upgraded, a controlled switchover is performed to pick a new master. Once that is done, the old master is upgraded as well.

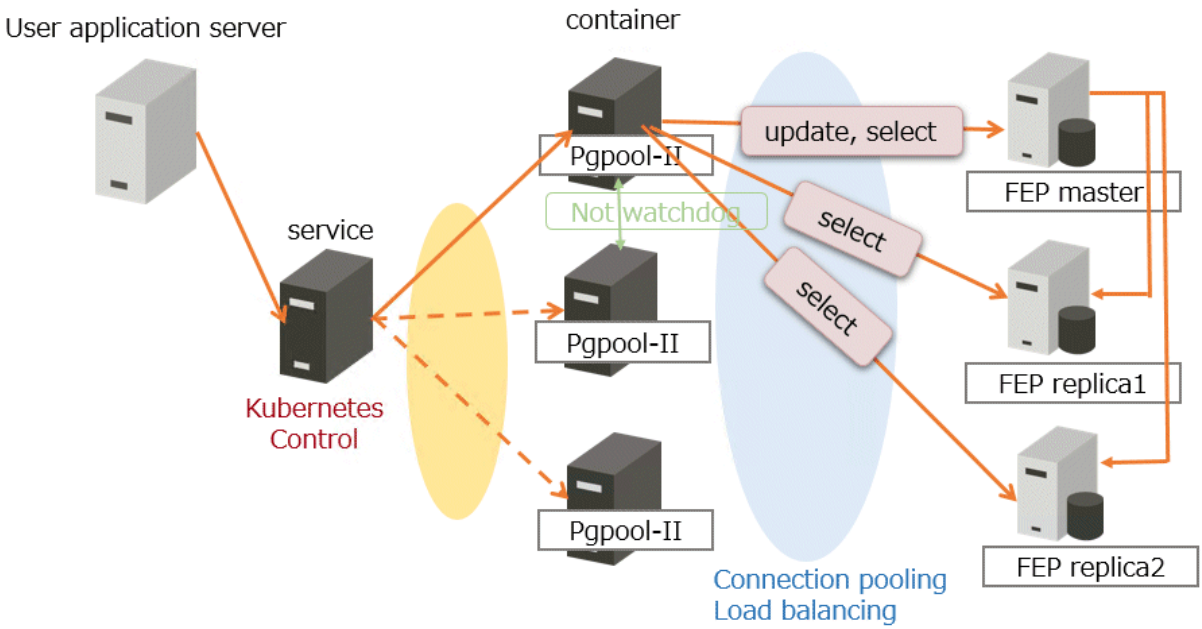
2.5 Configurable Volume Per Cluster

To improve performance, may want to separate the volume storing database files and WAL files. Similarly, one may want to use a dedicated volume for a new tablespace. The operator gives the end user the flexibility to create a FEPCluster with multiple PVs and select a suitable storage class for the PV. For example, one can create a FEPCluster with data volume, wal volume on a storage class backed up by SSD and a log volume on a storage class backed up by HDD.

2.6 Deploying Pgpool-II and Connect to FEPCluster from Operator

Users can deploy the FEP pgpool2 container and access the database via Pgpool-II to use load-balancing and connection pooling features.

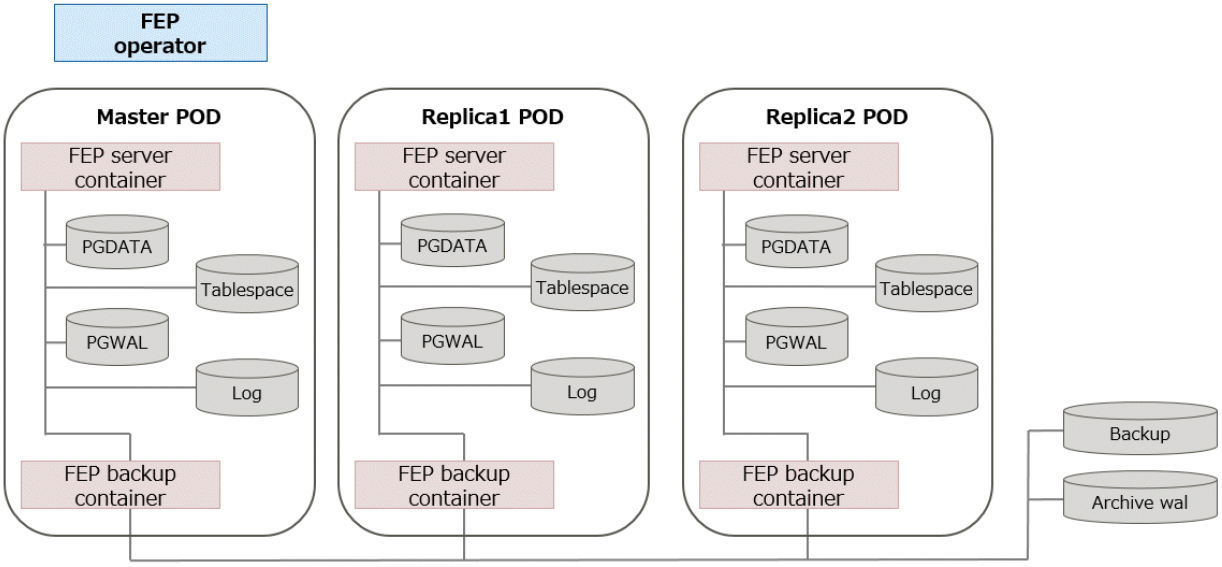
Multiple FEP pgpool2 containers can be deployed for load-share and high availability. Users can request a Kubernetes service to distribute their work across multiple FEP pgpool2 containers.



2.7 Scheduling Backup from Operator

The FEP backup container is deployed as a sidecar to each FEP server POD. The backup is performed at scheduled time set by the user (like crontab). The FEP backup container determines if the FEP server in the POD is a master or replica, and will perform the backup process only on the master POD. The volume storing backup and archived WAL files must be on a shared storage such as NFS or AWS S3.

Backup and WAL archiving is accomplished with pgBackRest.

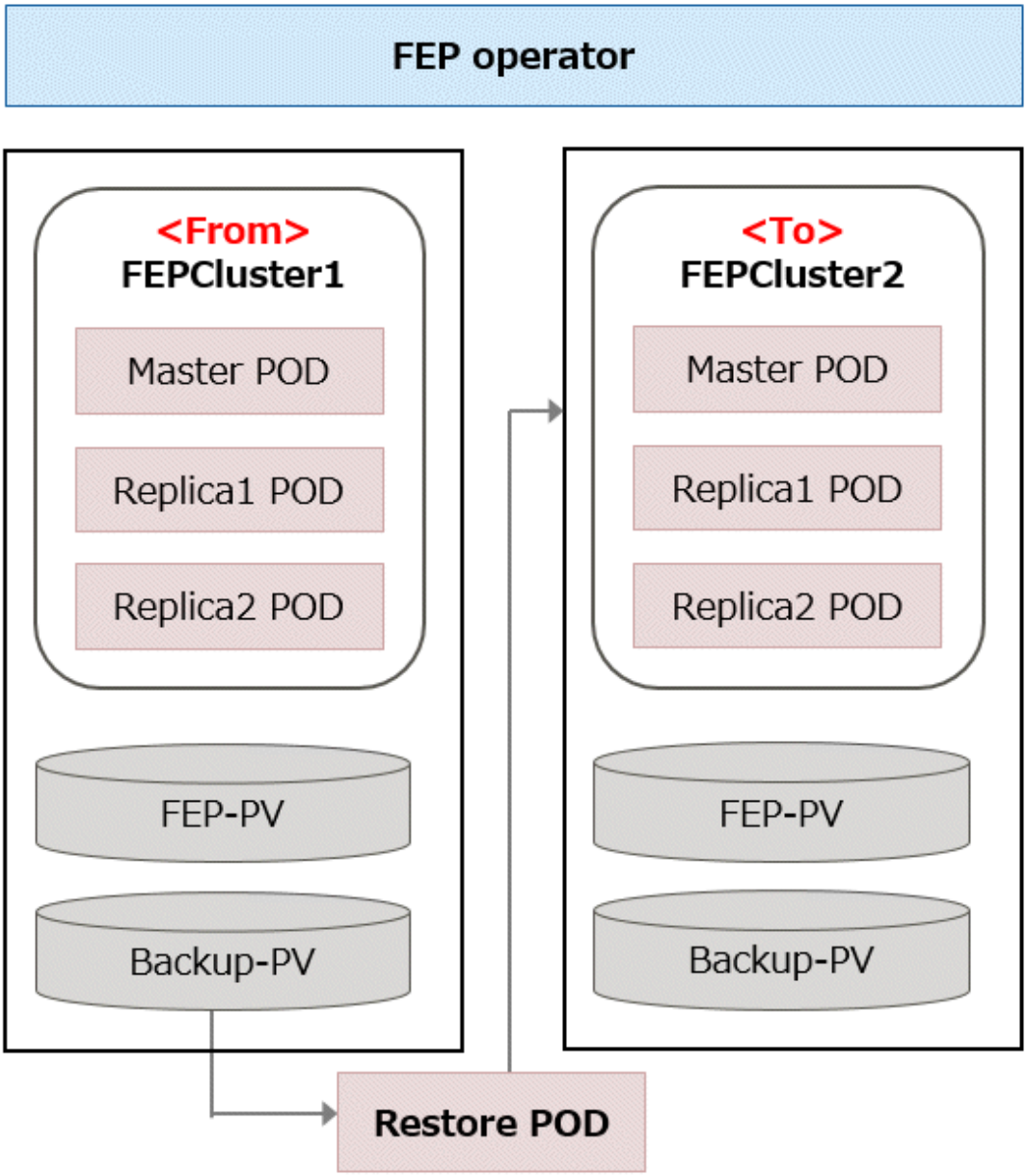


2.8 Perform PITR and Latest Backup Restore from Operator

There are two types of restore: one is to restore backup data to an existing FEP cluster, and the other is to create a new FEP cluster and restore backup data.

The former retains the attributes of the FEP cluster, such as IP address and name, while the latter is created from scratch.

The restore process deploys a restore container. The restore container performs the pgBackRest restore operation from the backup data to be restored to the master server of the FEP cluster. After the data is restored to the master server, the FEP cluster is created by synchronizing the data to two replica servers.


FEP operator
<From>
FEPCluster1
Master POD
Replica1 POD
Replica2 POD
FEP-PV
Backup-PV
<To>
FEPCluster2
Master POD
Replica1 POD
Replica2 POD
FEP-PV
Backup-PV
Restore POD

Appendix A OSS Supported by FUJITSU Enterprise Postgres for Kubernetes

The OSS supported by FUJITSU Enterprise Postgres for Kubernetes is listed below.

OSS name	Version and level	Description	Reference
PostgreSQL	12.12	Database management system	PostgreSQL Documentation
orafce	3.18.1	Oracle-compatible SQL features	"Compatibility with Oracle Databases" in the FUJITSU Enterprise Postgres Application Development Guide
Pgpool-II	4.1.11	Failover, connection pooling, load balancing, etc.	"Pgpool-II" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server
pg_statsinfo	12.0	Collection and accumulation of statistics	"pg_statsinfo" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server
pg_hint_plan	12.1.3.7	Tuning (statistics management, query tuning)	- "pg_hint_plan" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server - "Optimizer Hints" in the FUJITSU Enterprise Postgres Application Development Guide
pg_dbms_stats	1.3.11		- "pg_dbms_stats" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server - "Locked Statistics" in the FUJITSU Enterprise Postgres Application Development Guide
pg_repack	1.4.6	Table reorganization	"pg_repack" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server
pg_rman	1.3.13	Backup and restore management	"pg_rman" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server
pgBadger	11.7	Log analysis	"pgBadger" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server
pg_bigm	1.2	Full-text search (multibyte)	"pg_bigm" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server
PostgreSQL JDBC driver	42.2.8	JDBC driver	"JDBC Driver" in the FUJITSU Enterprise Postgres Application Development Guide
psqlODBC	12.01.0000	ODBC driver	"ODBC Driver" in the FUJITSU Enterprise Postgres Application Development Guide
pgBackRest	2.30	Backup and restore management	"Scheduling Backup from Operator" in the User's Guide
patroni	2.0.2	Postgres cluster management	"High Availability" in the User's Guide
Python	3.7.5	PL/Python	

FUJITSU

FUJITSU Enterprise Postgres 12 for Kubernetes

User's Guide

Linux

J2UL-USER-08ENZ0(00)
October 2022

Preface

Purpose of this document

This document describes system configuration, design, installation, setup, and operational procedures of the FUJITSU Enterprise Postgres for Kubernetes.

Intended readers

This document is intended for people who are:

- Considering installing FUJITSU Enterprise Postgres for Kubernetes
- Using FUJITSU Enterprise Postgres for Kubernetes for the first time
- Wanting to learn about the concept of FUJITSU Enterprise Postgres for Kubernetes
- Wanting to see a functional overview of FUJITSU Enterprise Postgres for Kubernetes

Readers of this document are also assumed to have general knowledge of:

- Linux
- Kubernetes
- Containers
- Operators

Structure of this document

This document is structured as follows:

Chapter 1 Overview of Operator Design

Describes an overview of the operator design.

Chapter 2 System Requirements

Describes the system requirements.

Chapter 3 Operator Installation

Describes the installation of the FEP operator.

Chapter 4 Deployment Container

Describes container deployment.

Chapter 5 Post-Deployment Operations

Describes the operation after deploying the container.

Chapter 6 Abnormality

Describes the actions to take when an error occurs in the database or an application.

Appendix A Quantitative Values and Limitations

Describes the quantitative values and limitations.

Abbreviations

The following abbreviations are used in this manual:

Full Name	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes FUJITSU Software Enterprise Postgres	FEP or FUJITSU Enterprise Postgres

Full Name	Abbreviations
Vertical Clustered Index	VCI
Transparent Data Encryption	TDE
Point in time recovery	PITR
Custom Resource	CR
Custom Resource Definition	CRD
Persistent Volume	PV
Universal Base Image	UBI
OpenShift Container Platform	OCP
Mutual TLS	MTLS

Abbreviations of manual titles

The following abbreviations are used in this manual as manual titles:

Full Manual Title	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes Release Notes	Release Notes
FUJITSU Software Enterprise Postgres for Kubernetes Overview	Overview
FUJITSU Software Enterprise Postgres for Kubernetes User's Guide	User's Guide
FUJITSU Software Enterprise Postgres for Kubernetes Reference	Reference

Trademarks

- Linux is a registered trademark or trademark of Mr. Linus Torvalds in the U.S. and other countries.
- Red Hat and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.
- S/390 is a registered trademark of International Business Machines Corporation in the United States or other countries or both.

Other product and company names mentioned in this manual are the trademarks or registered trademarks of their respective owners.

Export restrictions

If this document is to be exported or provided overseas, confirm legal requirements for the Foreign Exchange and Foreign Trade Act as well as other laws and regulations, including U.S. Export Administration Regulations, and follow the required procedures.

Issue date and version


```
Edition 8.0: October 2022
Edition 7.0: September 2022
Edition 6.0: June 2022
Edition 5.0: March 2022
Edition 4.0: December 2021
Edition 3.0: November 2021
Edition 2.0: April 2021
Edition 1.0: March 2021
```


Copyright

Contents

Chapter 1 Overview of Operator Design

This chapter describes an overview of the operator design.

1.1 Design Task

This section describes the operation of FEP.

First, determine the configuration. You then design each feature and deploy the container. You can use FEP features immediately after deployment.

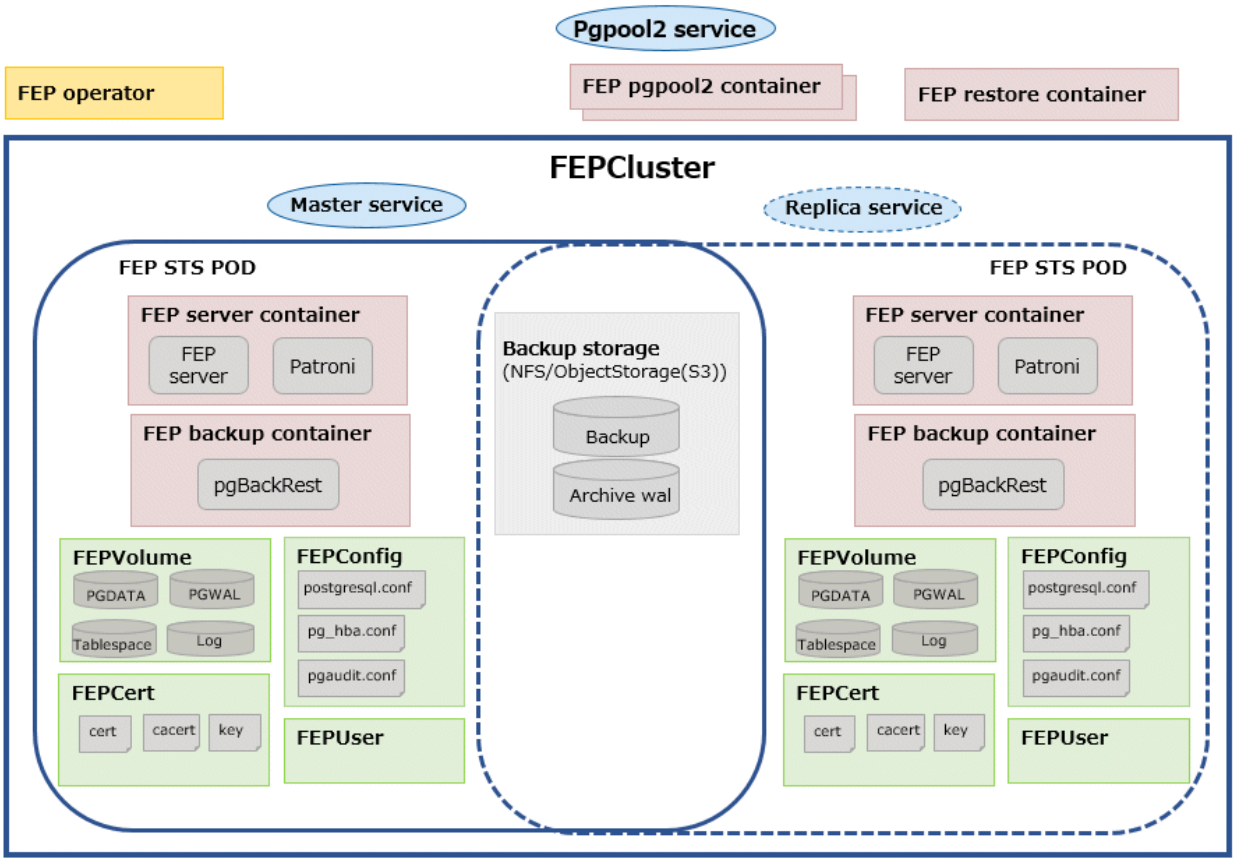
Task	Design required to operate FEP	Where to find
FEP setup	Required.	1.3.1 Deployment
High availability configuration	May be necessary. (When checking or changing the behavior of high availability. However, even by default, constant high availability operation is possible.)	1.3.2 High Availability
Volume settings	May be necessary. (When setting the volume. However, even by default, allocate a fixed volume.)	1.3.3 Configurable Volume per Cluster
Pgpool-II setup	May be necessary. (When using Pgpool-II.)	1.3.4 Deploying Pgpool-II and Connect to FEPCluster from Operator
Backup/restore settings	May be necessary. (When using a backup and restore.)	1.3.5 Scheduling Backup from Operator 1.3.6 Perform PITR and Latest Backup Restore from Operator

1.2 System Configuration Design

This section describes the system configuration.

1.2.1 Server Configuration

The following is an overview diagram of the server configuration:



System component

Describes various system resources.

Configuration server type	Description
FEP operator	A container that accepts user requests and is responsible for automating database construction and operational operations.
FEP server container	A container for the FEP server.
FEP backup container	A container that performs scheduled backup operations. Created on the same POD as the FEP server container.
FEP restore container	A container that performs the restore operation. Temporarily created during a restore operation.
FEP pgpool2 container	A container that uses Pgpool-II to provide load balancing and connection pooling. If you do not use it, you do not need to create it.
Backup storage	Storage where backup data is stored. If you do not need to obtain a backup, you do not need to create one.
FEPCluster	Parent CR for FEP Cluster definition and configuration
FEPBackup	Child CR for backup configuration
FEPVolume	Child CR for volumes.
FEPConfig	Child CR for FEP configurations.
FEPCert	Child CR for system certificates.
FEPUser	Child CR for database users.
FEPAction	CR for performing actions.

Configuration server type	Description
Master service	A service to connect to the master FEP server.
Replica service	A service to connect to the replica FEP server.
Pgpool2 service	A service for connecting to Pgpool-II.

1.2.2 User Account

The user accounts used by this product are as follows.

User type	User name	Description
Infrastructure administrator	Mandatory	A system administrator (superuser) who has root privileges on all the servers that make up this product.
Database administrator	Mandatory	Install, set up, start, stop, and perform operation and maintenance of this product.
Application developer	Mandatory	Develops and executes database applications.

1.2.3 Basic Information of the Container

This section describes the basic information of the container.

FEP server container

The naming convention for the FEP server container is as below.

fujitsu-enterprise-postgres-server:*OS-FEPBaseVersion-MajorVersion.MinorVersion-ARCH*

For each *Version*, specify the following:

Field	Values	Description
OS	ubi8	
FEPBaseVersion	12	
MajorVersion	1,2, ...	To be used when major change in image, including server patch application
MinorVersion	0,1,2 ...	To be used when minor changes in image, e.g bug fix in container script

The first publishing will expect following names / tagging (Manifest and Child images).

- fujitsu-enterprise-postgres-server: latest
  - fujitsu-enterprise-postgres-server:ubi8-12-1.0-amd64
  - fujitsu-enterprise-postgres-server:ubi8-12-1.0-s390x

FEP backup container

Use the same naming convention for FEP backup containers as for FEP server containers.

fujitsu-enterprise-postgres-backup:*OS-FEPBaseVersion-MajorVersion.MinorVersion-ARCH*

For each *Version*, specify the following:

Field	Values	Description
OS	ubi8	

Field	Values	Description
FEPBaseVersion	12	
MajorVersion	1,2, ...	To be used when major change in image, including server patch application
MinorVersion	0,1,2 ...	To be used when minor changes in image, e.g bug fix in container script

The first publishing will expect following names / tagging (Manifest and Child images)

- fujitsu-enterprise-postgres-backup: latest
  - fujitsu-enterprise-postgres-backup:ubi8-12-1.0-amd64
  - fujitsu-enterprise-postgres-backup:ubi8-12-1.0-s390x

FEP restore container

Use the same naming convention for FEP restore containers as for FEP server containers.

fujitsu-enterprise-postgres-restore:*OS-FEPBaseVersion-MajorVersion.MinorVersion-ARCH*

For each *Version*, specify the following:

Field	Values	Description
OS	ubi8	
FEPBaseVersion	12	
MajorVersion	1,2, ...	To be used when major change in image, including server patch application
MinorVersion	0,1,2 ...	To be used when minor changes in image, e.g bug fix in container script

The first publishing will expect following names / tagging (Manifest and Child images)

- fujitsu-enterprise-postgres-restore: latest
  - fujitsu-enterprise-postgres-restore:ubi8-12-1.0-amd64
  - fujitsu-enterprise-postgres-restore:ubi8-12-1.0-s390x

FEP pgpool2 container

Use the same naming convention for FEP pgpool2 containers as for FEP server containers.

fujitsu-enterprise-postgres-pgpool2: *OS-FEPBaseVersion-MajorVersion.MinorVersion-ARCH*

For each *Version*, specify the following:

Field	Values	Description
OS	ubi8	
FEPBaseVersion	12	
MajorVersion	1,2, ...	To be used when major change in image, including server patch application
MinorVersion	0,1,2 ...	To be used when minor changes in image, e.g bug fix in container script

The first publishing will expect following names / tagging (Manifest and Child images)

- fujitsu-enterprise-postgres-pgpool2: latest
  - fujitsu-enterprise-postgres-pgpool2:ubi8-12-1.0-amd64
  - fujitsu-enterprise-postgres-pgpool2:ubi8-12-1.0-s390x

1.3 Parameter Information for the Custom Resource

This section describes the parameter information for custom resources.

postgresql-cfg format

A postgresql-cfg represent ConfigMap for containing postgresql parameters. The file is used to contain the parameters which need to be reflected in postgresql.conf of the instance. Since patroni ignores all parameters which are not known by OSS postgresql.conf, an approach is defined to treat FEP Parameters in a special way.

The content of the ConfigMap is defined by key=value format. The following table shows the detail:

Spec	Example	Comment
The content may have multiple key/value pairs	foo=bar foo1=bar1	-
The value cannot have space unless quoted.	foo=bar bar2	Invalid
The quoted value cannot have another value after	foo='bar bar2' something	Invalid
The key value pair must have a '=' sign	-	-
White spaces are allowed before/after/between the key value pair	foo = bar	-

Any content after '#' will be ignored	# this is a comment
	foo=bar #this is a comment
		-

The value may be quoted by single quotes	foo='bar bar2'	-
Single quote can be escaped by two single quotes	foo='It''s ok'	Note: single quotes are not supported by Patroni edit-config command
Backslash '\' will be replaced by '\\' when invoking patronictl edit-config command	-	To avoid command line escape
When a key value pair is invalid, it will be ignored. the update continue to process next pair	foobar foo2=bar2	The 'foobar' will be ignored
The container script does not validate the key and value as long as they are in correct format.	-	-

It is recommended to use the psql's show command to verify parameter is setting correctly.

1.3.1 Deployment

Information for the FEPCluster

Equivalent Kubernetes command: kubectl apply -f FEPClusterCR.yaml

This operation will create a FEPCluster with supplied information in FEPClusterCR.yaml.

Refer to "FEPCluster parameter" in the Reference for details.

1.3.2 High Availability

Describes the settings for using the highly available features.

Arbitration

Patroni is used to control and monitor FEP instance startup, shutdown, status and trigger failover should the master instance fails. It plays a significant role in the solution. If the Patroni process dies, especially on master POD, without notice, the POD will not update the Patroni cluster lock. This may trigger an unwanted failover to one of the Replica, without corresponding corrective action on the running master. This can create a split brain issue. It is important to monitor Patroni's status to make sure it is running. This is done using liveliness probe. Important to note that this is not configurable.

```
livenessProbe:
  httpGet:
    scheme: HTTP
    path: /liveness
    port: 25001
  initialDelaySeconds: 30
  periodSeconds: 6
  timeoutSeconds: 5
  successThreshold: 1
  failureThreshold: 3
```

1.3.3 Configurable Volume per Cluster

Cluster node (Pod) volumes are created according to the values set in the storage section of fepChildCrVal in the FEPCluster custom resource.

Note

- After you create the FEPCluster for the first time, you cannot add new volumes later or modify the storageClass or accessModes.
- You can resize the initially created volume only if the underlying storageClass supports dynamic resizing.

The following is the schema for the storage section of the FEPCluster customer resource:

Field	Mandatory	Sub-Field	Default	Description
archivewalVol	No	size	1Gi	Volume size of the archive log. Refer to "Estimating Database Disk Space Requirements" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server to help you design the size.
		storageClass	Defaults to platform default if omitted	SC is only set at start

Field	Mandatory	Sub-Field	Default	Description
		accessModes	Defaults to ReadWriteOnce if omitted	Access mode is only set at start
backupVol	No	size	2Gi	Volume size of the backup. Estimate based on the following formula: (full backup generations + incr backup generations + 1) * dataVol size
		storageClass	Defaults to platform default if omitted	SC is only set at start
		accessModes	Defaults to ReadWriteOnce if omitted	Access mode is only set at start
dataVol	Yes	size	2Gi	Volume size of the data. Refer to "Estimating Database Disk Space Requirements" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server and base the design on table/index size.
		storageClass	Defaults to platform default if omitted	SC is only set at start
		accessModes	Defaults to ReadWriteOnce if omitted	Access mode is only set at start
logVol	No	size	1Gi	Volume size of the log. If you change the log output level (default: WARNING), measure the actual amount of log output in a test environment.
		storageClass	Defaults to platform default if omitted	SC is only set at start
		accessModes	Defaults to ReadWriteOnce if omitted	Access mode is only set at start
tablespaceVol	No	size	512Mi	Volume size of the tablespace. When using tablespaces, as with dataVol, you should refer to "Estimating Database Disk Space Requirements" in the FUJITSU Enterprise Postgres Installation and

Field	Mandatory	Sub-Field	Default	Description
				Setup Guide for Server for information on sizing.
		storageClass	Defaults to platform default if omitted	SC is only set at start
		accessModes	Defaults to ReadWriteOnce if omitted	Access mode is only set at start
walVol	Yes	size	1200Mi	Volume size of the transaction log. Refer to "Estimating Database Disk Space Requirements" in the FUJITSU Enterprise Postgres Installation and Setup Guide for Server to help you design the size. Note that the default value for max_wal_size is 1 GB.
		storageClass	Defaults to platform default if omitted	SC is only set at start
		accessModes	Defaults to ReadWriteOnce if omitted	Access mode is only set at start

The 'accessMode' is been incorporated for the inclusion of pgBadger layer later. Giving it a shared volume capability will allow pgBadger Container to read logs from multiple server instance (master / replica) and expose it via a WebServer.

1.3.4 Deploying Pgpool-II and Connect to FEPCluster from Operator

Equivalent Kubernetes command: kubectl create FEPpgpool2

This operation will create a FEP pgpool2 container with supplied information.

Field	Default	Details
apiVersion	fep.fujitsu.io/v1	Fixed
kind	FEPPgpool2	Fixed
metadata.name	-	List the name of the FEP pgpool2 container.
metadata.namespace	-	Specify the namespace of the environment where you want to deploy the operator.
spec.image	-	Specifies the FEP pgpool2 container image to provide.
spec.count	2	List the number of FEP pgpool2 containers to create.
spec.serviceport	9999	Describes the TCP port for connecting to the FEP pgpool2 container.
spec.statusport	9898	Identifies the TCP port for connecting to the PCP process.
spec.limits.cpu	400m	List the number of CPUs (restriction) to allocate to resources.limits.cpu.
spec.limits.memory	512Mi	Specifies the memory size (restriction) to allocate to resources.limits.memory.

Field	Default	Details
spec.requests.cpu	200m	List the number of CPUs (request) to allocate to resources.requests.cpu.
spec.requests.memory	256Mi	Specifies the memory size (request) to allocate to resources.requests.memory.
spec.fepclustername	new-fep	Enter the FEPCluster name to connect to.
spec.customhba	-	If you want to use pool_hba.conf, describe what pool_hba.conf should contain from the line below.
spec.customparams	"|"	"|" and the Pgpool-II parameters. Refer to "Pgpool-II parameters" for detail.
spec.custompcp	"|"	If you use the pcp command, "|" and the contents of pcp.conf from the line below.
spec.customsslkey	"|"	If you want to do it, "|" and the Beethoven key content in the line below.
spec.customsslcert	"|"	If you want to do it, "|" and the contents of the public x 509 certificate from the line below.
spec.customsslcacert	"|"	If you want to do it, "|" and the following lines describe the contents of the CA root certificate in PEM format.
spec.customlogsize	100 Mi	Specifies the persistent volume size for log output.
spec.storageclassname	-	Specifies the storage class for log output.

Pgpool-II parameters

The parameters that can be specified are shown in the table below. For details on the parameters, refer to the Pgpool-II manual.

Category	Parameter name (Specified format)	Restart required after change
Connection settings	listen_addresses (string)	Y
	pcp_listen_addresses (string)	Y
	num_init_children (integer)	Y
	reserved_connections (integer)	Y
Authentication settings	enable_pool_hba (boolean)	
	allow_clear_text_frontend_auth (boolean)	
	authentication_timeout (integer)	
Backend settings	backend_weight0 (floating point)	
	backend_weight1 (floating point)	
	backend_flag0	
	backend_flag1	
Connection pooling	connection_cache (boolean)	Y
	max_pool (integer)	Y
	listen_backlog_multiplier (integer)	Y
	serialize_accept (boolean)	Y
	child_life_time (integer)	Y
	client_idle_limit (integer)	
	child_max_connections (integer)	Y

Category	Parameter name (Specified format)	Restart required after change
	connection_life_time (integer)	Y
	reset_query_list (string)	
Error reporting and log acquisition	client_min_messages (enum)	
	log_min_messages (enum)	
	log_statement (boolean)	
	log_per_node_statement (boolean)	
	log_client_messages (boolean)	
	log_hostname (boolean)	
	log_connections (boolean)	
	log_error_verbosity (enum)	
	log_line_prefix (string)	
Load sharing settings	load_balance_mode (boolean)	Y
	ignore_leading_white_space (boolean)	
	white_function_list (string)	
	black_function_list (string)	
	black_query_pattern_list (string)	
	database_redirect_preference_list (string)	
	app_name_redirect_preference_list (string)	
	allow_sql_comments (boolean)	
	disable_load_balance_on_write (string)	Y
	statement_level_load_balance (boolean)	
Health check	connect_timeout (integer)	
Streaming replication check	sr_check_period (integer)	
	sr_check_user (string)	
	sr_check_password (string)	
	sr_check_database (string)	
	delay_threshold (integer)	
	log_standby_delay (string)	
Secure Sockect Layer (SSL)	ssl (boolean)	Y
	ssl_ciphers (string)	Y
	ssl_prefer_server_ciphers (boolean)	Y
	ssl_ecdh_curve (string)	Y
	ssl_dh_params_file (string)	Y
Other parameters	relcache_expire (integer)	Y
	relcache_size (integer)	Y
	enable_shared_relcache (boolean)	Y
	relcache_query_target (enum)	
	check_temp_table (enum)	
	check_unlogged_table (boolean)	

1.3.5 Scheduling Backup from Operator

When creating a FEPCluster, users can obtain scheduled backups by setting up backup definitions. Users can also modify the backup schedule by modifying the Backup custom resource that was created.

A backup definition includes the following:

- Acquisition time (Specify in crontab format)
- Backup type (Full or incremental backups)

Backup is taken on master POD only.

Backup processing is performed by pgBackRest.

Parameter can be set to pgbackrestParams in CR definition.

The maximum number of backup schedules is 5.

See the pgBackRest User's Guide for details on the parameters.

However, some parameters are limited. Details are given below.

- 1.3.5.1 Important Setting Items
- 1.3.5.2 Parameters that cannot be Set
- 1.3.5.3 Restricted Parameters
- 1.3.5.4 About Sections in the Config File

1.3.5.1 Important Setting Items

Here are the important parameters for setting pgBackRest. This parameter sets the retention period of backup information. If automatic backup is set and this parameter is not set, the risk of overflowing the backup area increases.

Parameter	Overview of parameters	Setting value
Full Retention Option (repo retention -full)	Specify number of full backups to keep No default (should be set according to user backup policy)	natural number
Full Retention Type Option (repo retention-full-type)	spec.retention -full Specifies whether the setting is a number of retention days (time) or a number of retention times (count) No default (should be set according to user backup policy)	time/count

The following is a sample CR example of changing the backup retention period (How long the PITR is valid) to 30 days after a FEPCluster deployment by setting the above parameters.

```
apiVersion: fep.fujitsu.io/v1
kind: FEPBackup
metadata:
  name: fepcluster-backup
spec:
  pgBackrestParams: |
    # define custom pgbackrest.conf parameters below to override defaults.
    [global]
    repo-retention-full = 30
    repo-retention-full-type = time
...
```

1.3.5.2 Parameters that cannot be Set

The following parameters in the pgBackRest Configuration Reference are not configurable.

Parameter	Overview of parameters	Reason
Copy Archive Option (--archive -copy)	Copy the WAL segments needed for consistency to the backup	To use internal fixed values
Backup from Standby Option (--backup-standby)	Back up from the standby cluster	Limited to backup from master
Stop Auto Option (--stop-auto)	Stops a previously failed backup on a new backup.	Because they are 9.6 not supported in
SSH client command Option (--cmd-ssh)	Path to ssh client executable	Not using ssh
Compress Option (--compress)	Use File Compression	For obsolete options (Use compresse-type option instead)
Delta Option (--delta)	Restore or Backup with Checksum	For new restores only
Lock Path Option (--lock-path)	Path where the lock file is stored	To use internal fixed values
Keep Alive Option (--sck -keep-alive)	Enable keep-alive messages on socket connections	To use internal fixed values
Spool Path Option (--spool-path)	Path to store temporary data for asynchronous archive-push and archive-get commands	For automatic determination from FEPCluster CR values
Console Log Level Option (--log-level-console)	Console Log Level	It is not expected to operate on POD.
Std Error Log Level Option (--log-level-stderr)	Stderr log level	It is not expected to operate on POD.
Log Path Option (--log-path)	Log File Destination	For automatic determination from FEPCluster CR values
Azure Repository Account Option (--repo-azure-account)	Azure account used to store the repository	Azure storage is not supported
Azure Repository TLS CA File Option (--repo-azure-ca-file)	Use a non-default CA file for the Azure Repository TLS CA file system	
Azure Respository TLS CA Path Option (--repo-azure-ca-path)	Use non-default CA path for Azure Repository TLS CA path system	
Azure Repository Container Option (--repo-azure-container)	Azure repository container.Azure container used to store the repository.	
Azure Repository Host Option (--repo-azure-host)	Azure Repository Host	
Azure Repository Key Option (--repo-azure-key)	Azure Repository Shared Key or Shared Access Signature	
Azure Repository Key Type Option (--repo-azure-key-type)	Azure Repository Key Type	
Azure Repository Server Port Option (--repo-azure-port)	Azure Repository Server Port	
Azure Repository Server Certificate Verify Option (--repo-azure-verify-tls)	Validate Azure Repository Server Certificate.	

Parameter	Overview of parameters	Reason
Repository Host Option (--repo-host)	Repository host for remote operations via SSH	Repository Host is not used
Repository Host Command Option (--repo-host-cmd)	Path of pgBackRest on Repository Host	
Repository Host Configuration Option (--repo-host-config)	Repository Host Configuration File Path	
Repository Host Configuration Include Path Option (--repo-host-config-include-path)	Repository hosts configuring include path	
Repository Host Configuration Path Option (--repo-host-config-path)	Repository Host Configuration Path	
Repository Host Port Option (--repo-host-port)	Repository host port when "repo-host" is configured	
Repository Host User Option (--repo-host-user)	Repository host user when "repo-host" is configured	
Repository Path Option (--repo-path)	Path where backups and archives are stored	For automatic determination from FEPCluster CR values
Archive Retention Option (--repo-retention-archive)	The number of consecutive WAL backups to keep.	This option is not recommended, and WAL retention is controlled by the Full Retention Option and Full Retention Type Option.
Archive Retention Type Option (--repo-retention-archive-type)	Backup Type for WAL Retention	It is recommended not to change from the default.
Differential Retention Option (--repo-retention-diff)	Number of incremental backups to keep	No incremental backups
Archive Mode Option (--archive-mode)	Retains or disables the archive for the restored cluster.	To use internal fixed values
Include Database Option (--db-include)	Restore only the specified database	To restore the entire FEP cluster, including all databases
Link All Option (--link-all)	Restore all symbolic links.	To use internal fixed values
Link Map Option (--link-map)	Changes the destination of a symbolic link.	To use internal fixed values
Recovery Option Option (--recovery-option)	Setting options in postgreSQL recovery.conf	To use internal fixed values
Tablespace Map Option (--tablespace-map)	Restoring tablespace to a specified directory	For automatic determination from FEPCluster CR values
Map All Tablespaces Option (--tablespace-map-all)	Restores all tablespaces to the specified directory	No tablespace required because there is only one tablespace per FEPCluster
PostgreSQL Host Option (--pg-host)	PostgreSQL host for remote operations via SSH	No SSH connection required
PostgreSQL Host Command Option (--pg-host-cmd)	Path of pgBackRest exe on the PostgreSQL host	To use internal fixed values
PostgreSQL Host Configuration Option (--pg-host-config)	Path of the pgBackRest configuration file	To use internal fixed values

Parameter	Overview of parameters	Reason
PostgreSQL Host Configuration Include Path Option (--pg-host-config-include-path)	Setting pgBackRest on PostgreSQL host include path	To use internal fixed values
PostgreSQL Host Configuration Path Option (--pg-host-config-path)	Path to configure pgBackRest on the PostgreSQL host	To use internal fixed values
PostgreSQL Host Port Option (--pg-host-port)	SSH Port Specification	No SSH connection required
PostgreSQL Host User Option (--pg-host-user)	The logon user when hosting PostgreSQL, if pg-host is set.	No SSH connection required
PostgreSQL Path Option (--pg-path)	PostgreSQL data directory.	For automatic determination from FEPCluster CR values
PostgreSQL Port Option (--pg-port)	PostgreSQL Ports	For automatic determination from FEPCluster CR values
PostgreSQL Socket Path Option (--pg-socket-path)	PostgreSQL Unix socket path	For automatic determination from FEPCluster CR values
PostgreSQL Database User Option (--pg-user)	PostgreSQL database user	To use internal fixed values

1.3.5.3 Restricted Parameters

Of the parameters in the pgBackRest Configuration Reference, the following parameters limit the configurable values.

Parameter	Overview of parameters	Possible Values
Repository Type Option (--repo-type)	Type of storage to use for the repository	posix/s3

1.3.5.4 About Sections in the Config File

In FEPBackup CR, you can write the contents of pgbackrest.conf, but the setting for stanza (Backup space for pgBackRest) is specified internally.

The following sections are not allowed;

[stanza: command] , [stanza]

1.3.6 Perform PITR and Latest Backup Restore from Operator

There are two types of restore: one is to restore backup data to an existing FEPCluster, and the other is to create a new FEPCluster and restore backup data.

The former retains the attributes of the FEPCluster, such as IP address and name, while the latter is created from scratch.

The restore process deploys a FEP restore container. The FEP restore container performs the pgBackRest restore operation from the backup data to be restored to the master server of the FEPCluster. After the data is restored to the master server, the FEPCluster is created by synchronizing the data to two replica servers.

If user create a new FEPCluster, the newly created FEPCluster will inherit the settings of the source cluster, unless otherwise specified

User can also create a cluster with different settings from the source cluster by including the settings in FEPRestore CR.

Switching connections to the new cluster

The restore creates a new FEPCluster. If necessary, you need to set up Pgpool-II and change the access point of the application to the new cluster or the new Pgpool-II.

About recovering a failed FEPCluster

Even if the existing FEPCluster fails and the FEP is not running, if the volume of the backup area is safe, it is possible to restore from the backup data.

1.3.7 FEP Unique Feature Enabled by Default

Enable the following FEP features:

- Vertical Clustered Index (VCI)
- Data masking
- pgaudit
- Transparent Data Encryption (TDE)

VCI, Data masking and pgaudit

The VCI, Data masking and pgaudit are enabled by default. The postgresql.conf in container contains the following parameters:

```
shared_preload_libraries = 'pgx_datamasking,vci,pg_prewarm,pgaudit'
session_preload_libraries = 'vci,pg_prewarm'
max_worker_processes= 20
```

The user can overwrite these values in config map.

The value of max_worker_processes needs to be tuned in case VCI is used. Value of vci.control_max_workers and vci.max_parallel_degree should be added in max_worker_processes; refer to the FUJITSU Software Enterprise Postgres Operation Guide for further details.

TDE

TDE is enabled by default. For details on how to specify the passphrase, refer to "FEPCluster parameter" in the Reference.

Chapter 2 System Requirements

This chapter describes the system requirements.

2.1 Components Embedded

The FEP Server container embeds following components. However it is understood that these components are bound to be upgraded in the maintenance phase.

No	Component	Version	Description
1	Red Hat UBI minimal	8	Meant to provide base OS image for the container
2	FUJITSU Enterprise Postgres Server	12.12	To provide server capabilities
3	Patroni	2.0.2	To provide HA capabilities and other management to the Cluster

2.2 CPU

It should be noted that it provides supports to both the following CPU Architectures to meet the scope of work.

No	CPU architecture
1	x86
2	s390x

2.3 Supported Platform

It supports running on the following platforms.

No	Platform	Version
1	OpenShift Container Platform	4.6, 4.8, 4.9, 4.10

Supports storage supported by OpenShift.

However, backup and archive WAL volumes require shared storage, such as NFS.

Chapter 3 Operator Installation

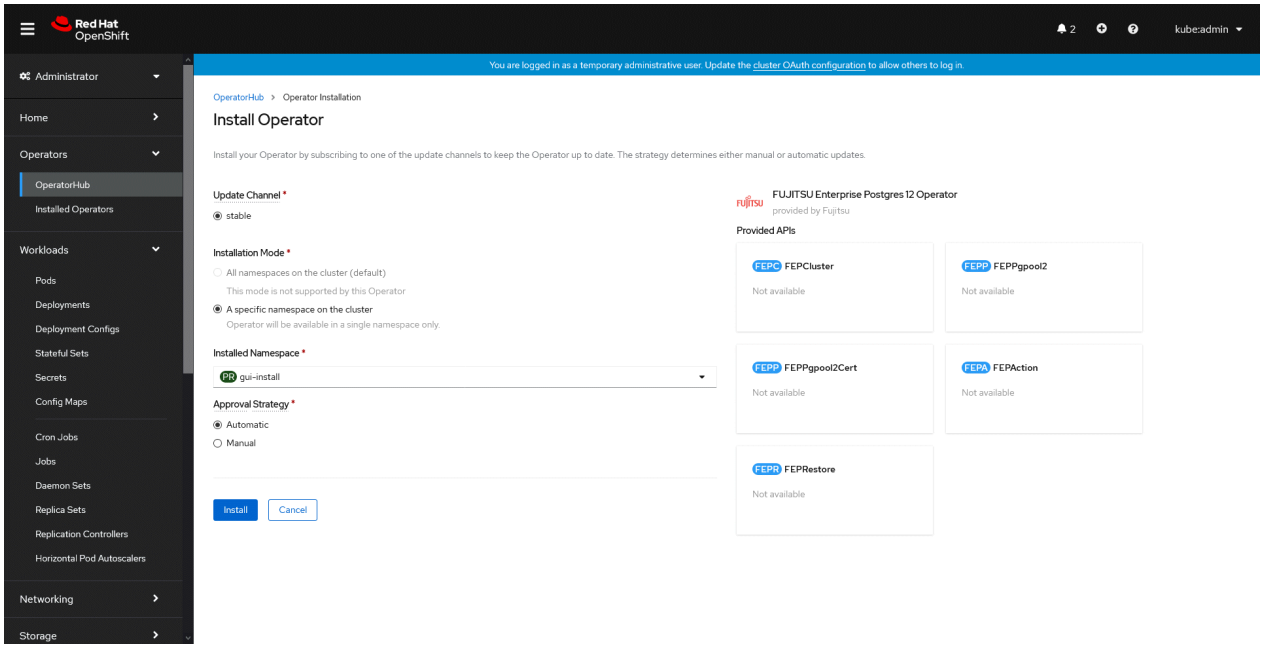
This chapter describes the installation of the FEP operator.

Refer to "5.9 Assigned Resources for Operator Containers" for more information about the resources assigned to installed operator containers and how to change them.

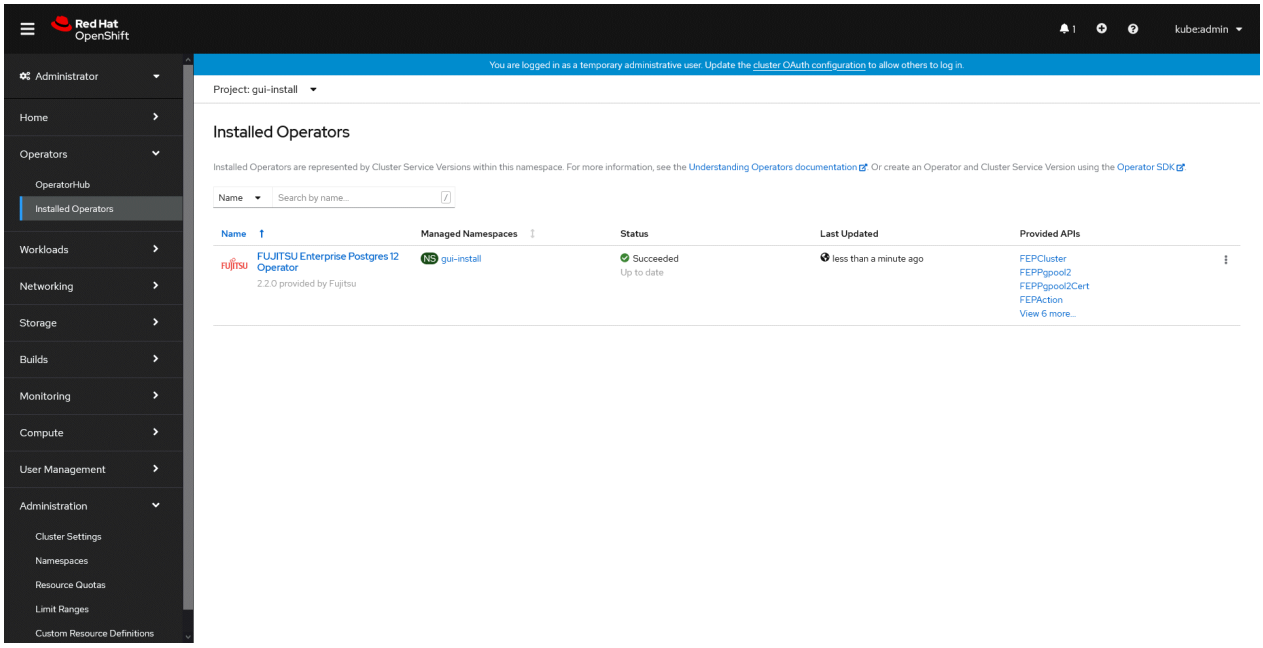
3.1 Installation from RedHat OperatorHub

Once operator is certified by RedHat, it is made available on OperatorHub on all RedHat OpenShift container platform.

1. On OpenShift platform, logon with credentials that has privileges to install operator. Click on OperatorHub on menu item under Operators and type filter keyword Fujitsu to find FUJITSU Enterprise Postgres 12 Operator.
2. Click on FEP Operator to install operator. It will bring up details page with **install** button as below.
3. Click on "**Install**" button, to bring up following screen to choose namespace and approval strategy. Select "**A specific namespace on the cluster**" and choose desired namespace. Leave everything else to default and click install.



4. Wait till installation is complete and status changes to "**Succeeded**".



Chapter 4 Deployment Container

This chapter describes container deployment.

Note

Each volume of a Pod created by a FEPCluster deployment is sized by default for the following operations:

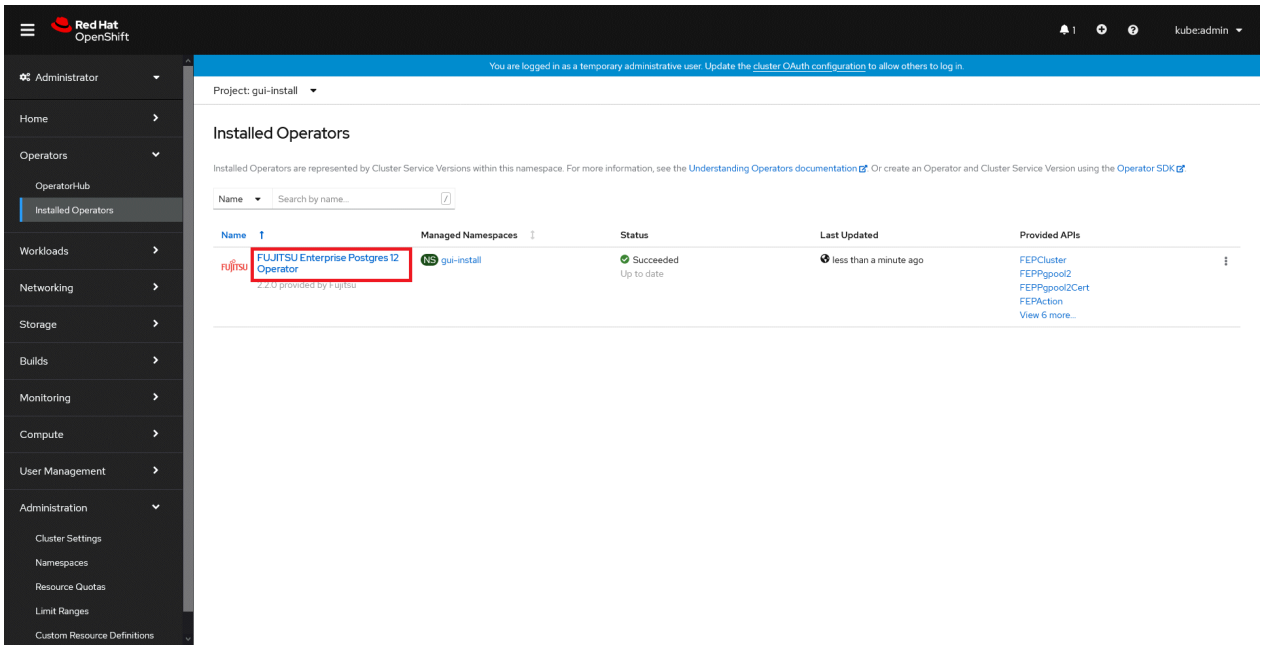
- Data size: 1 GB
- Daily update: about 50 MB

Refer to "1.3.3 Configurable Volume per Cluster" to design each volume size according to actual operation.

4.1 Deploying FEPCluster using Operator

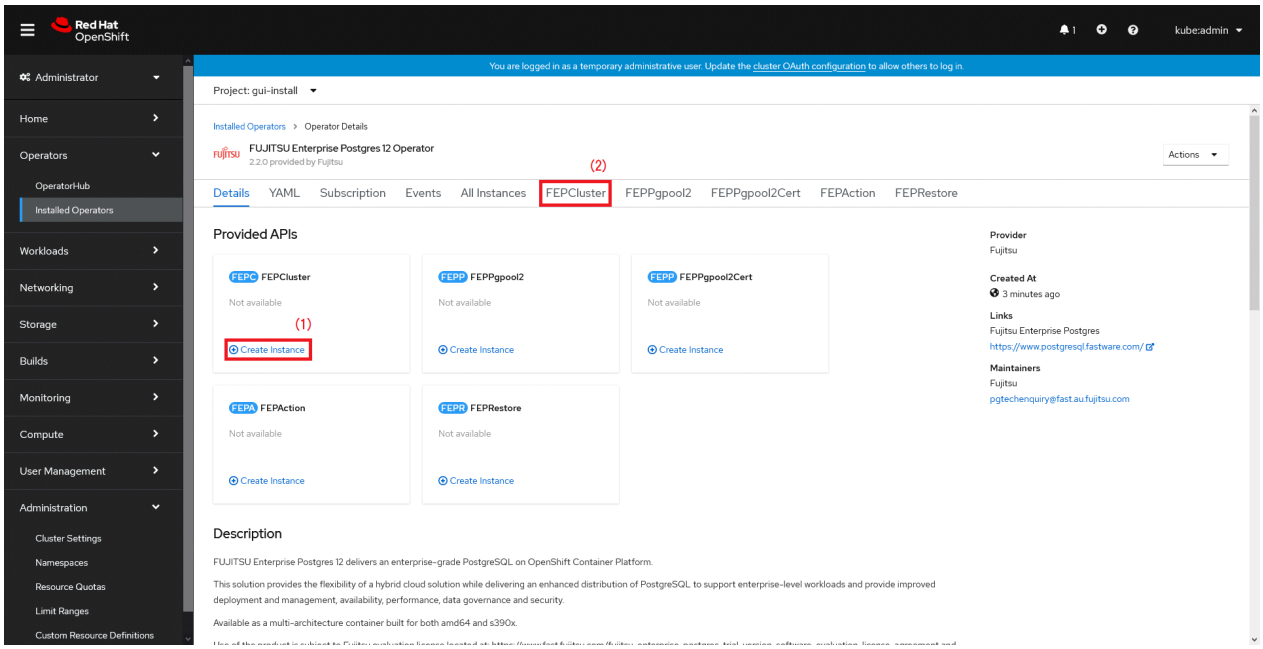
To deploy a FEPCluster in given namespace, follow these steps:

1. Under "Operators" menu item, click on "**Installed Operators**". You would see the installed FEP operator deployed in "Chapter 3 Operator Installation". Click on the name of operator.

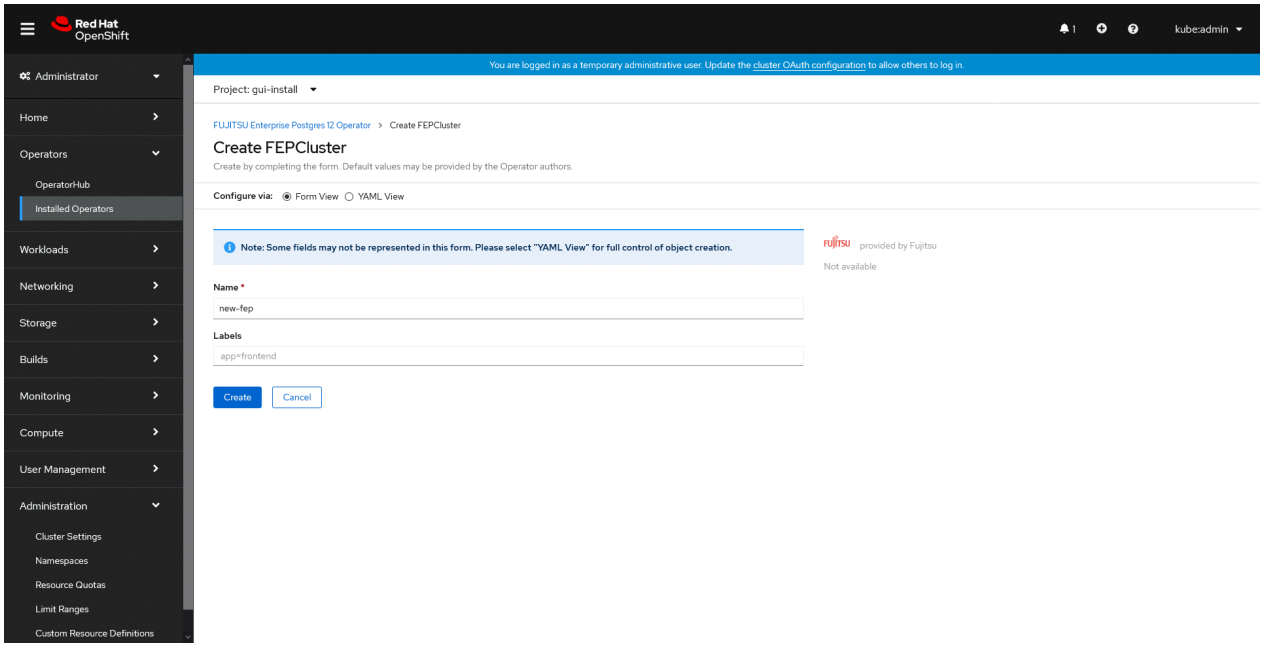


2. It will display a page with all CRs this operator supports. FEPCluster is the main CR and all others are child CR. We would create main CR and all other CRs will be created automatically by Operator.
   To create Cluster CR, either
   (1) Click on "**Create Instance**" under FEPCluster.

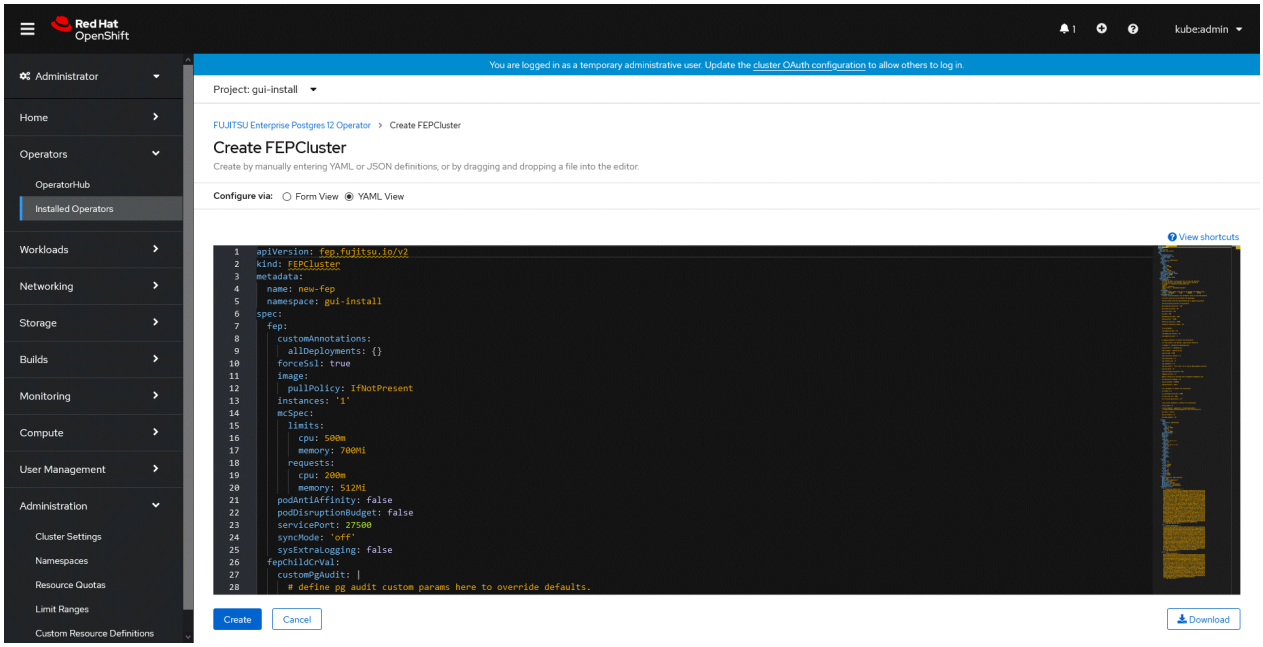
OR
(2) Click on "**FEPCluster**" on top and then click on "**Create FEPCluster**" on next page.



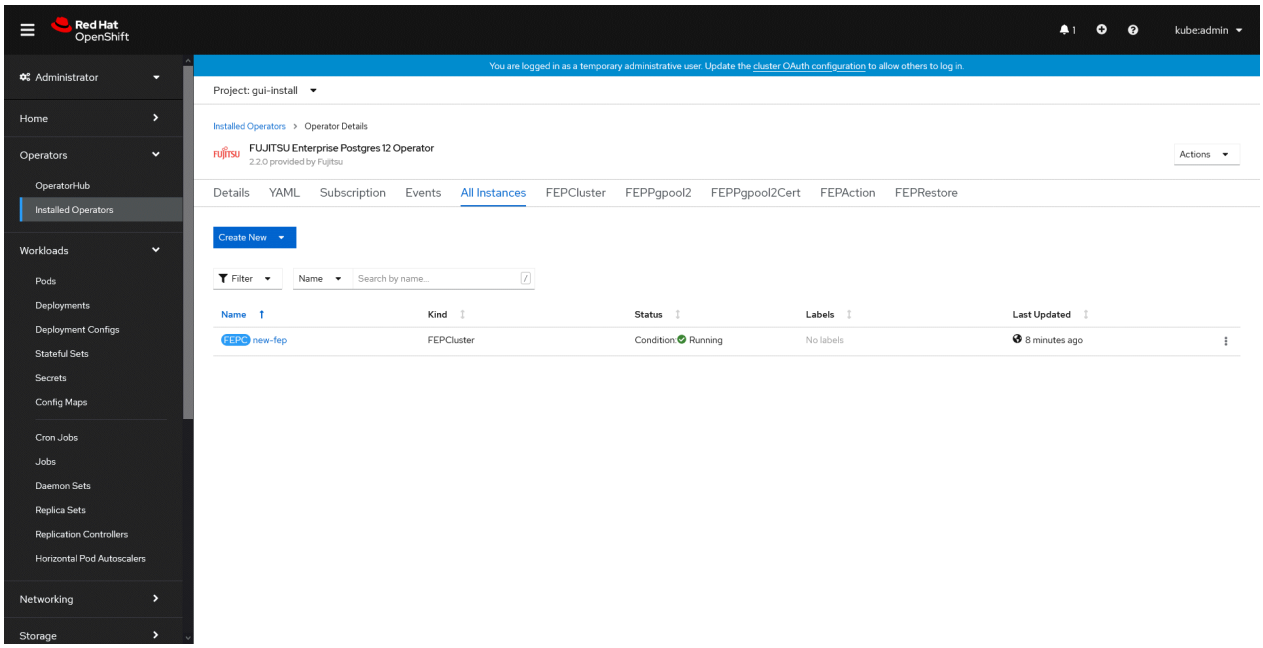
3. This will bring to "Create FEPCluster" page. Here you have two options to configure. The first one is Form View. At the moment, in Form View , one can change only the name of cluster being deployed. Default name is "new-fep".
This name must be unique within a namespace.



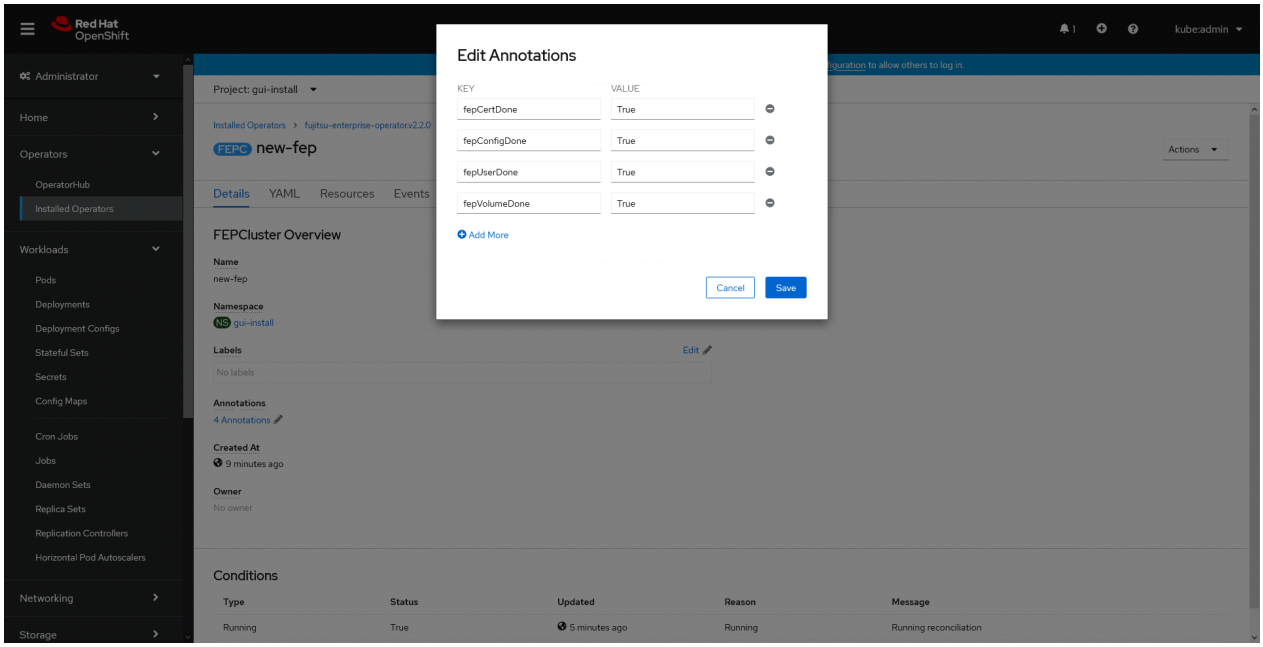
4. In YAML View, starting value of CR is visible and one can choose to modify parameters before creating CR. Refer to the Reference for details of parameters.



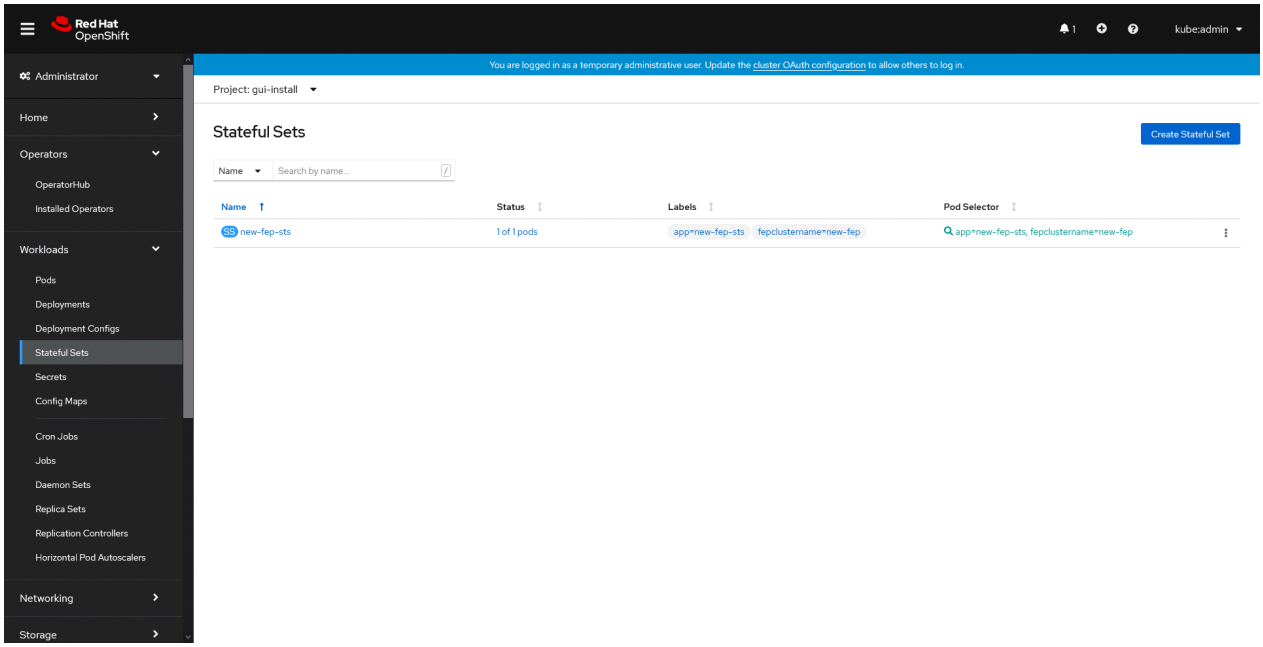
5. When "**Create**" is clicked on either of two pages above, operator creates FEPCluster CR and there after one by one FEPBackup, FEPConfig, FEPVolume, FEPUser and FEPCert child CRs are created automatically.
The starting values for child CRs are taken from "fepChildCrVal" section of FEPCluster CR yaml file. Once child CRs are created, respective values are managed through child CRs only. Modifying value in FEPCluster "fepChildCrVal" section. Operator reflects changes from FEPCluster parent CR to respective child CRs. Only allowable changes are reflected in child CRs. Child CRs are marked internal objects and hence will not be visible on OCP console. However, you can check child CRs using command line tools.



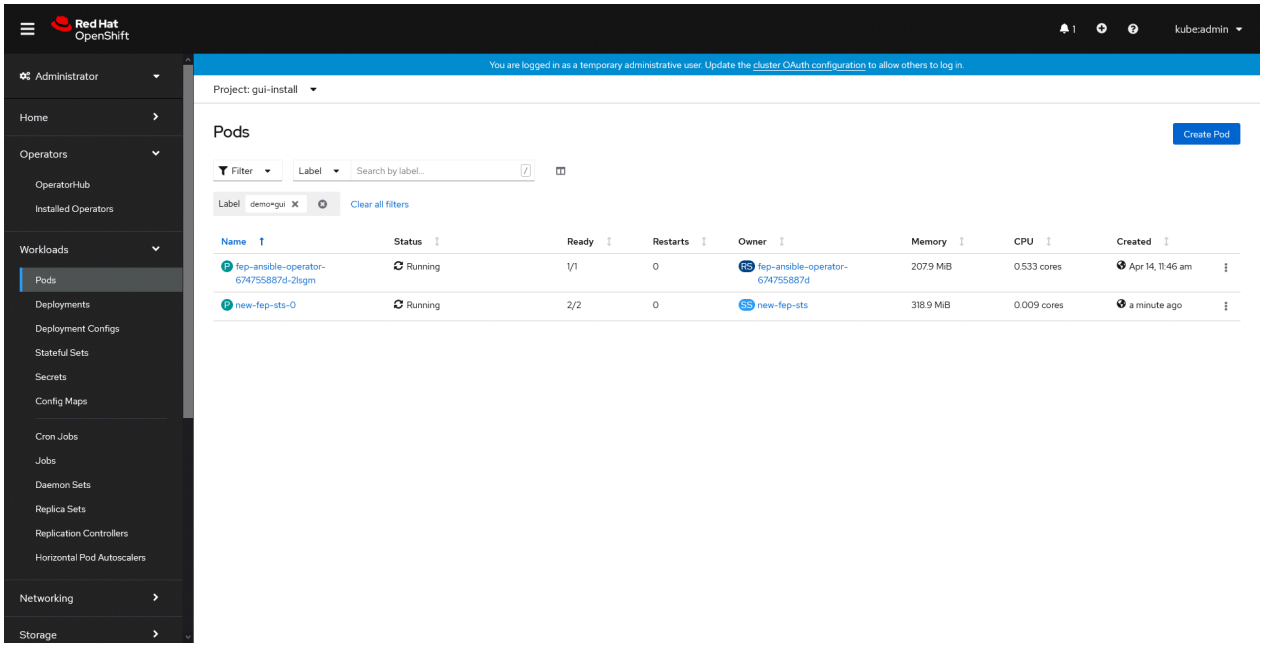
6. In FEPCluster CR, annotations are added to indicate that child CRs are created successfully and has initialised properly. It may take some time to complete.



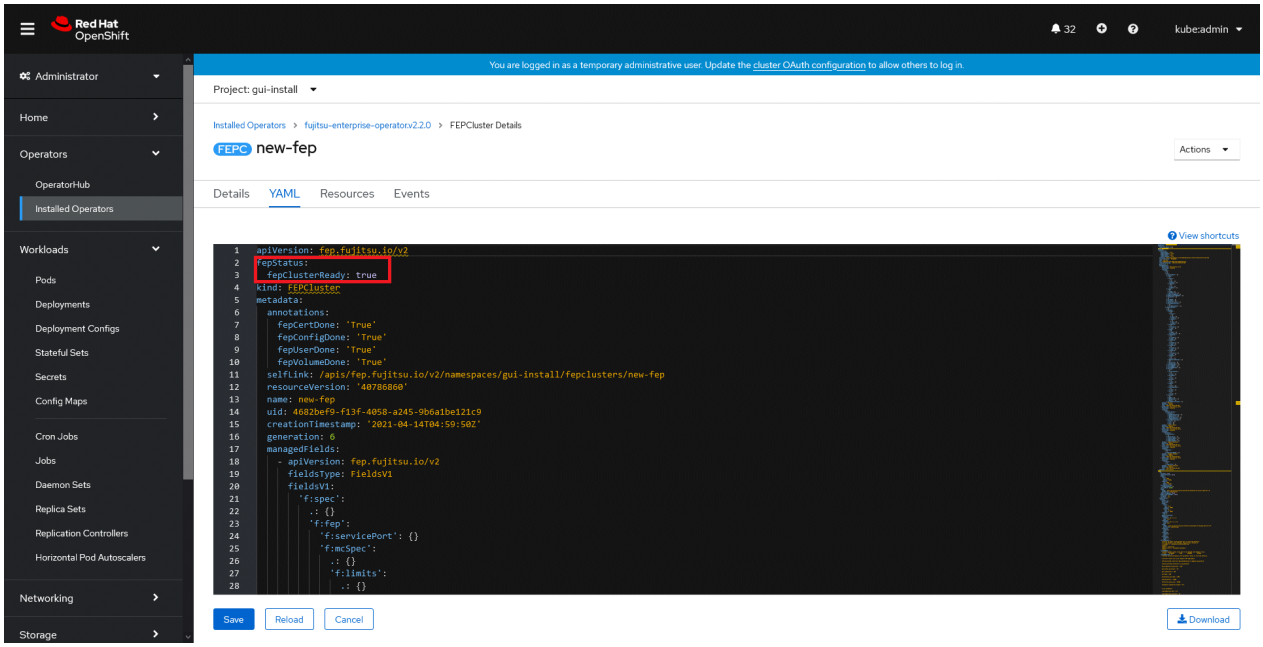
7. Once all four child CRs are marked done in annotations, operator creates StatefulSet for the cluster.



8. StatefulSet will start one FEP instance at one time and will wait it to be ready before starting next one.



9. Once all instances of FEP servers are started, operator marks a flag "fepClusterReady" in "fepStatus" section of CR to be **true**, indicating that FEPCluster is ready for use. Looking at YAML of FEPCluster CR, it would look like as below:



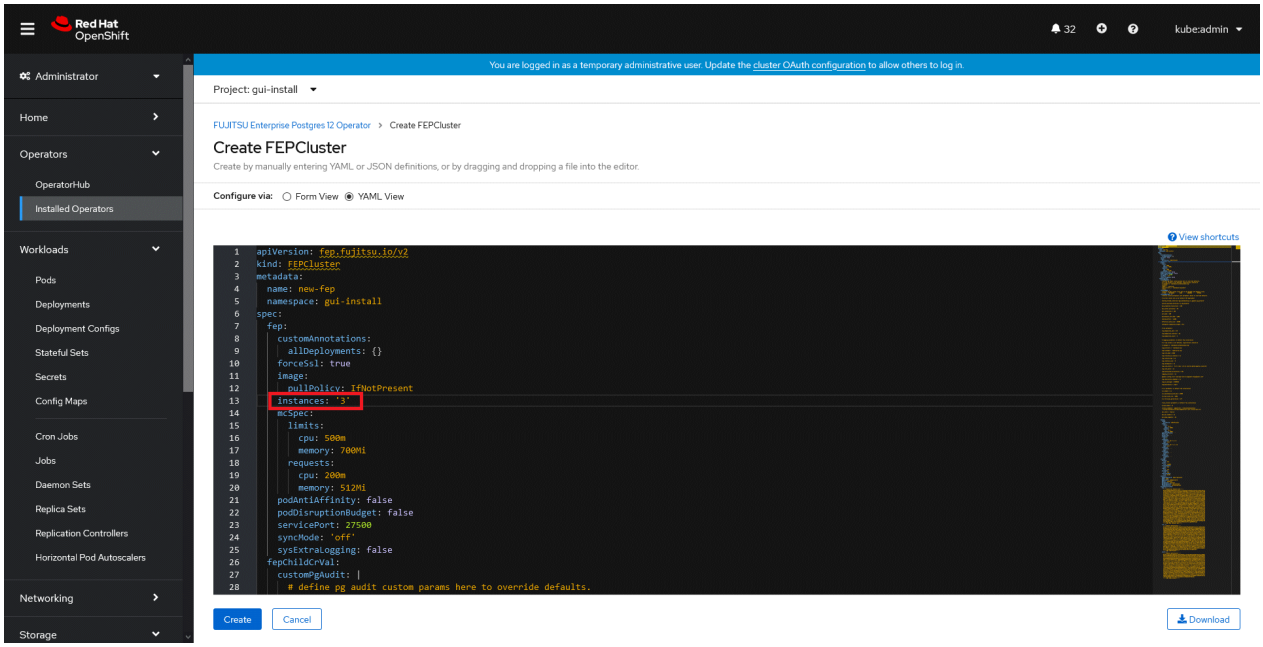
10. Operator also masks the sensitive fields like passwords, passphrase, certificates and keys in FEPCluster fepChildCrVal and also in child CRs.

4.2 Deploy a Highly Available FEPCluster

To deploy a highly available FEPCluster in given namespace, follow these steps:

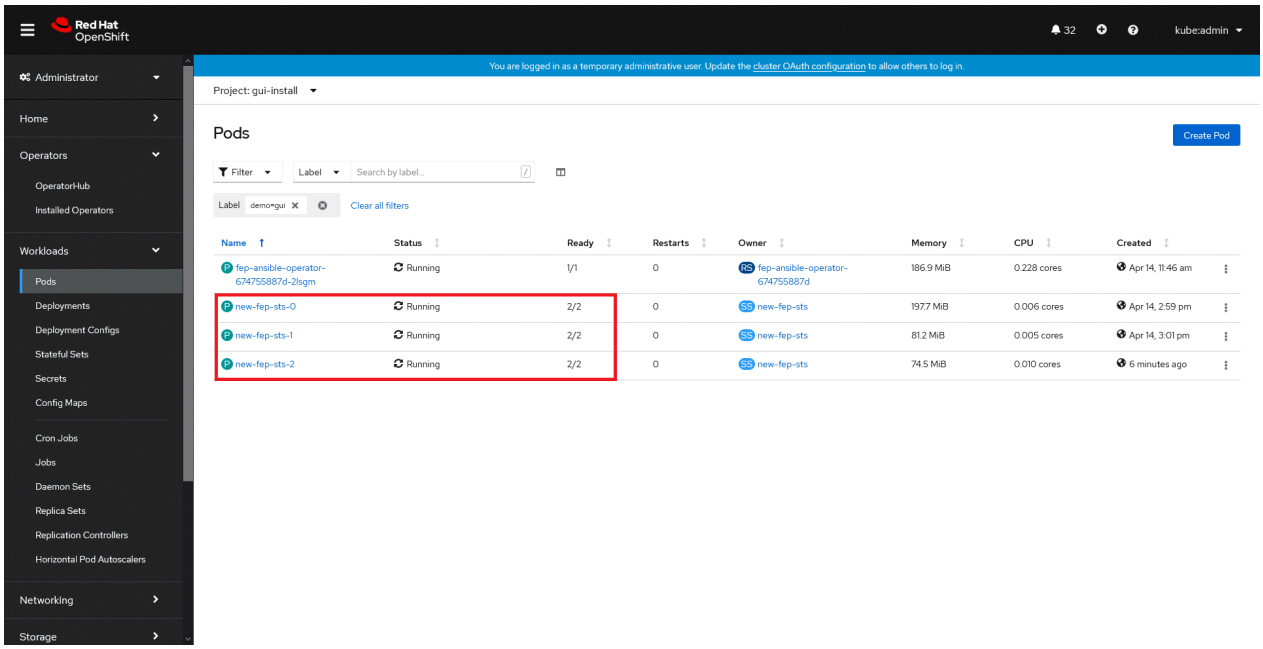
1. It is the same as the procedure from step 1 to step 3 in "4.1 Deploying FEPCluster using Operator".

2. Instead of step 4 in "4.1 Deploying FEPCluster using Operator", change to the yaml view and specify '3' for the "instances" parameter of "fep" in "spec".



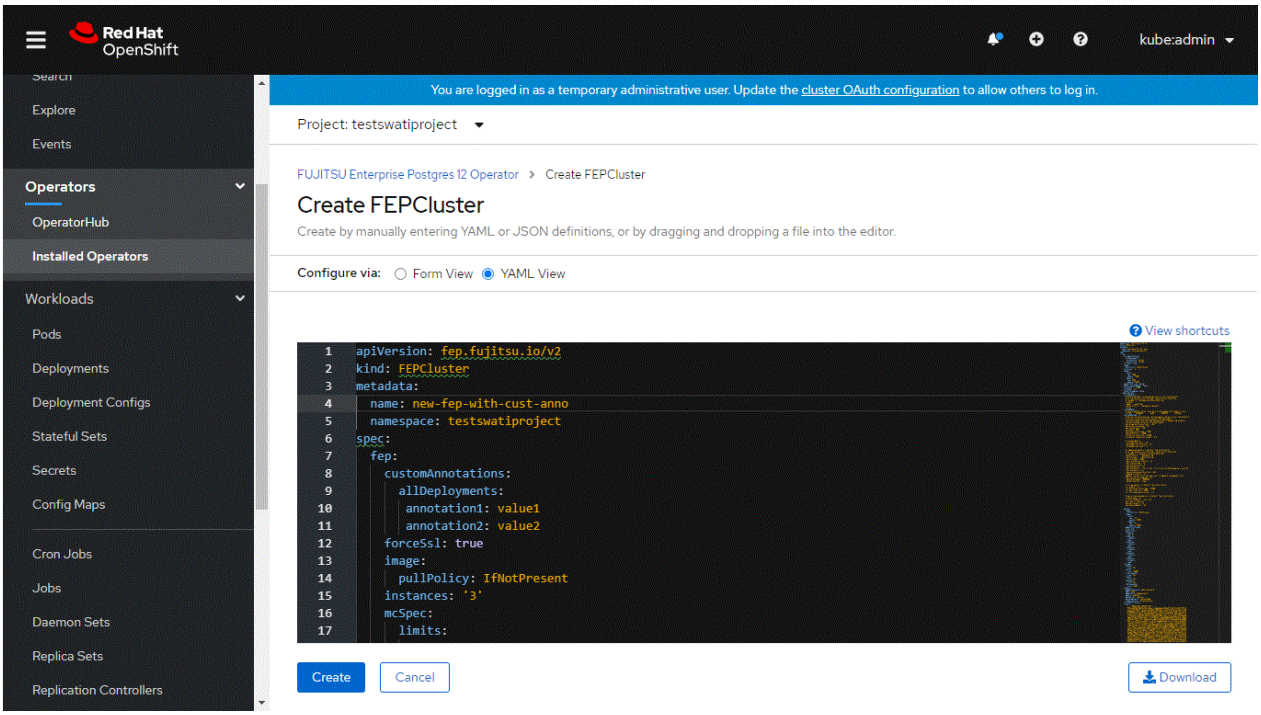
3. It is the same as the procedure from step 5 to step 10 in "4.1 Deploying FEPCluster using Operator".

4. Three pods deployed and ready for a highly available FEPCluster.

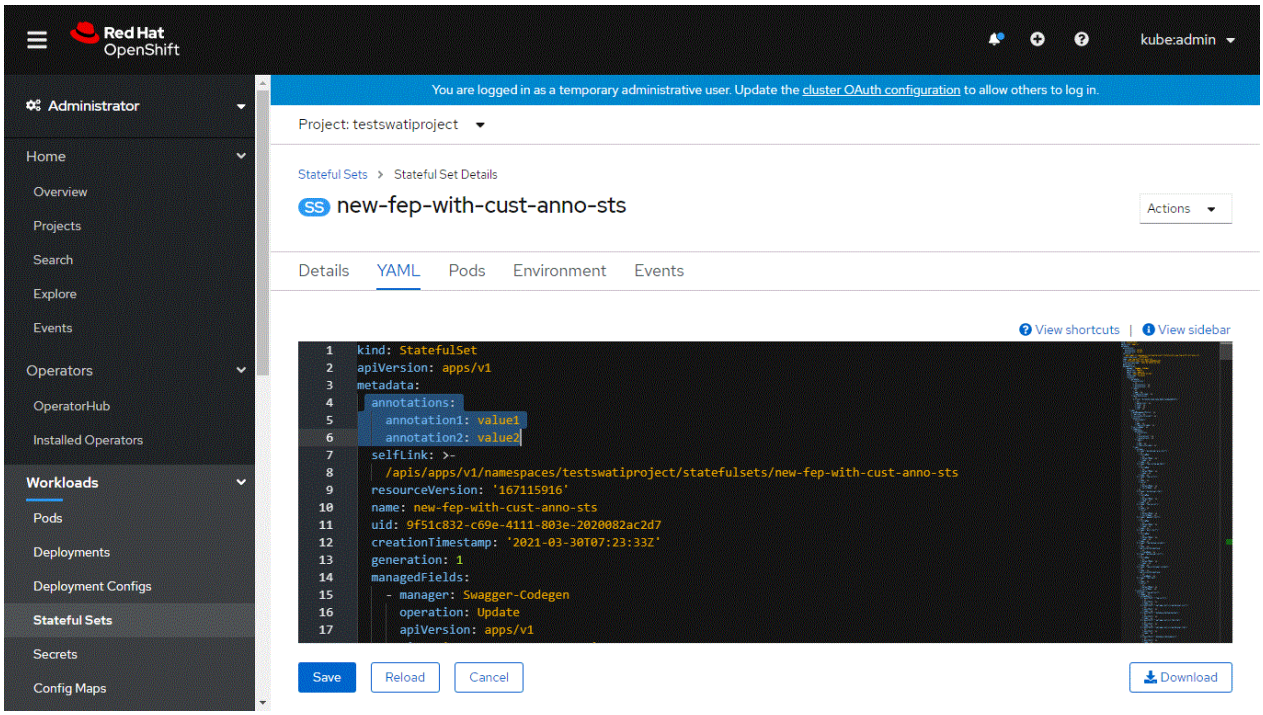


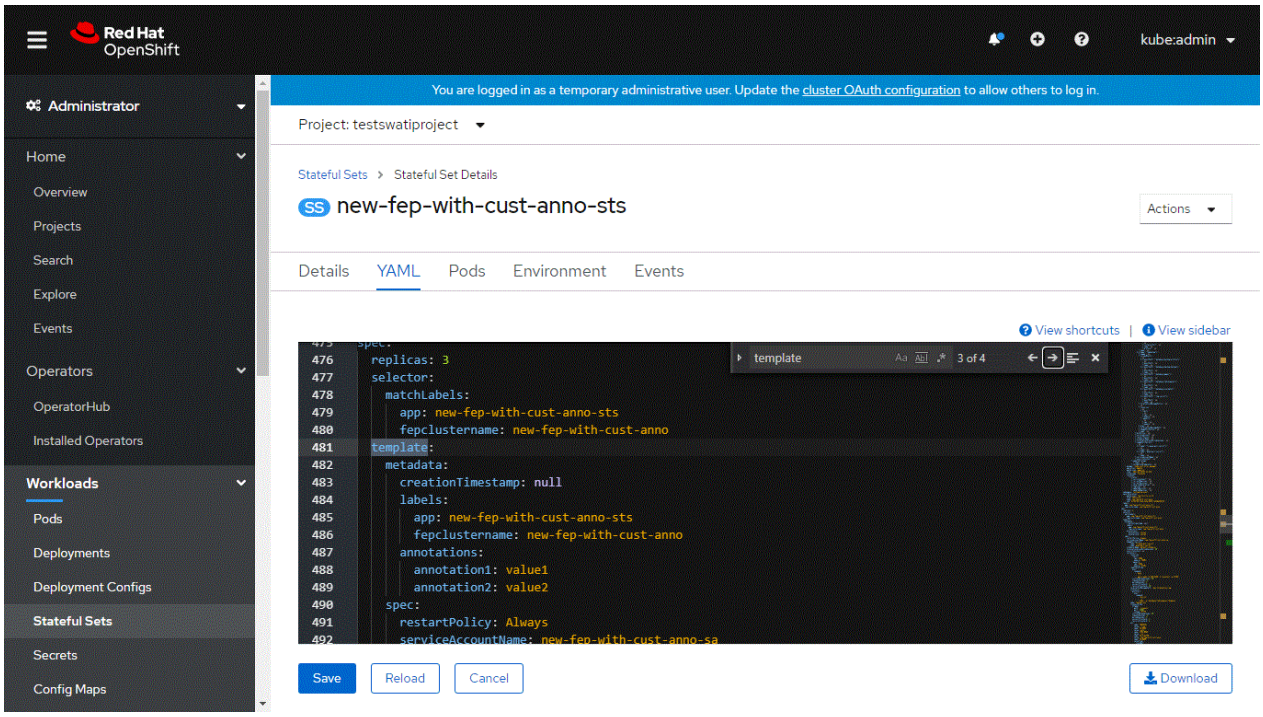
4.3 Adding Custom Annotations to FEPCluster Pods using Operator

1. In YAML view of the Create FEPCluster section, add custom annoatations as below and then click on Create.



2. Both the Statefulset and its resulting pods will be annotated with your prvided annotations: archivewalVol and backupVol must be ReadWriteMany.




Red Hat OpenShift
kube:admin
Administrator
Home
Overview
Projects
Search
Explore
Events
Operators
OperatorHub
Installed Operators
Workloads
Pods
Deployments
Deployment Configs
Stateful Sets
Secrets
Config Maps
You are logged in as a temporary administrative user. Update the cluster OAuth configuration to allow others to log in.
Project: testswatiproject
Stateful Sets > Stateful Set Details
SS new-fep-with-cust-anno-sts
Actions
Details
YAML
Pods
Environment
Events
View shortcuts | View sidebar
template
3 of 4
476 replicas: 3
477 selector:
478 matchLabels:
479 app: new-fep-with-cust-anno-sts
480 fepclustername: new-fep-with-cust-anno
481 template:
482 metadata:
483 creationTimestamp: null
484 labels:
485 app: new-fep-with-cust-anno-sts
486 fepclustername: new-fep-with-cust-anno
487 annotations:
488 annotation1: value1
489 annotation2: value2
490 spec:
491 restartPolicy: Always
492 serviceAccountName: new-fep-with-cust-anno-sa
Save
Reload
Cancel
Download

Chapter 5 Post-Deployment Operations

This chapter describes the operation after deploying the container.

5.1 Configuration Change

This section describes changes to the FEPCluster configuration.

List FEPCluster

Equivalent Kubernetes command: kubectl get FEPClusters (-A)

This operation will list all FEPClusters in a namespace, or if the -A option is specified, will list all FEPClusters in all namespace.

Default output format:

Field	Value	Details
NAME	.metadata.name	Name of Cluster
COUNT	.spec.fep.instances	Number of FEP nodes in the cluster

Example)

```
# kubectl get fepclusters -A

NAMESPACE       NAME            COUNT
namespace1      ns1fep1         3/3
namespace2      ns2fep2         5/5
```

Update FEPCluster

Equivalent Kubernetes command: kubectl apply -f <new_spec>

Operations that can be performed here.

Custom Resource spec	Change effect
.spec.fep.instances: *n*	Increase the number of nodes in the cluster to *n*.
.spec.fep.image.image: 'quay.io/fujitsu/fujitsu-enterprise-postgres-12-server:ubi8-12-1.1'	Minor upgrade of FEP image to tag ubi8-12-1.1.
spec.fepChildCrVal.backup.image.image: 'quay.io/fujitsu/fujitsu-enterprise-postgres-12-backup:ubi8-12-1.1'	Minor upgrade of Backup image to tag ubi8-12-1.1.

This will impact behaviour for values in fep section only.

Delete FEPCluster

Equivalent Kubernetes command: kubectl delete FEPCluster <cluster_name>

This operation will remove the FEPCluster by the cluster_name and all Child CRs (FEPVolume, FEPConfig, FEPCert & FEPUser) & resources associated with it.

Note

Deleting a FEPCluster will delete all PV associated with the cluster, including backup and archived WAL volumes (unless using AWS S3). This is an unrecoverable action.

When connecting from outside the OpenShift system

Automatically creating a service with ClusterIP to connect to the deployed container. You can connect to FEP or FEP pgpool2 services from the OpenShift system's internal network. To access from outside the OpenShift system, you need to know the address of the OpenShift node.

For example, "Access the FEP pgpool2 container from an application server that is running outside the OpenShift system but is part of the Internal network".

An example of how to check the node IP in OpenShift.

```
$ oc get nodes
 NAME STATUS ROLES AGE VERSION
 fepcontainercluster -qmb 95 -master-0 Ready master 44 d v 1.19. 0 + 7070803
 fepcontainercluster -qmb 95 -master-1 Ready master 44 d v 1.19 .0 + 7070803
 fepcontainercluster -qmb 95 -master-2 Ready master 44 d v 1.19 .0 + 7070803
$ oc describe nodes fepcontainercluster-qmb 95 -master-0 | grep IP
 InternalIP: 10.0.2.8
```

An example of verifying the service resource for the FEP pgpool2 container.

```
$ oc get all
```

Check where the resource type is Service (Begin with the "svc /").

You can also see this with the oc get svc command. The following is an example.

```
$ oc get svc
 NAME TYPE CLUSTER -IP EXTERNAL -IP PORT (S) AGE
 svc-feppgpool2-feppgpool2 NodePort 172.30.248.12 <none> 9999: 30537/TCP, 9998: 30489/TCP 2m5s
```

This is an example of accessing the FEP pgpool2 container.

```
$psql -h 10.0.2.8 -p 30537 -c "show pool_nodes"
```

5.2 FEPCluster Resource Change

5.2.1 Changing CPU and Memory Allocation Resources

Describes how to change the CPU and memory resources assigned to a pod created by a FEPCluster.

This allows you to scale the pod vertically through custom resources.

To modify CPU and memory resources, modify the spec.fep.mcSpec section(*1) of the FEPCluster custom resource and apply your changes.

When the changes are applied, restart the replica server with the new resource settings. If there are multiple replica servers, restart them one at a time. When all replica servers are restarted, one of them is promoted to the new master server due to a switchover. Then restart the container image on the original master server. This allows you to change resource settings for all servers with minimal disruption.

*1) Modifying this section scales up the FEP server container. For information about other container resource sections, refer to "FEPCluster Parameters" in the Reference.

5.2.2 Resizing PVCs

Describes how to resize a PVC assigned to a pod created by a FEPCluster.

This allows you to increase the size of the volume allocated to the pod through custom resources.

To change the PVC size, modify the size of each volume in the spec.fepChildCrVal.storage section of the FEPCluster custom resource and apply the change. These changes apply to all PVCs assigned to the pod created by the FEPCluster.

Note

- PVC resizing is extensible only.
- You can resize a PVC only if the StorageClass supports dynamic resizing.
- If the StorageClass does not support resizing PVCs, use the FEPRestore custom resource to create a new FEPCluster to resize the PVC. For more information, refer to "FEPRestore Custom Resource Parameters" in the Reference.

5.3 Minor Version Upgrade

Minor FEP version upgrade is done by replacing the image in FEPCluster customer resource with a new one. For the procedure, refer to "Minor Version Upgrade" in the Overview.

Note

The upgrade process will cause an outage on the cluster for the duration to upgrade both Master and Sync Replica. If there is no Sync Replica in the cluster, the outage is limited to the length of time to upgrade the Master (or actually the failover time required to take another replica been promoted by patroni).

5.4 Cluster Master Switchover

Specify "switchover" for the action type of the FEPAction CR to update FEPAction CR.

Equivalent Kubernetes command: kubectl apply -f <new_spec>

"switchover" action type expects users to specify the name of the current leader/primary pod that they want to switchover from. Specify the name in the args section under the FEPAction CR spec as below:

```
spec:
fepAction:
 args:
 - new-fep-sts-2
 type: switchover
 targetClusterName: new-fep
```

Here, new-fep-sts-2 is the current primary.

Refer to "FEPAction Custom Resource Parameters" in the Reference for more information on parameters.

5.5 FEPPGPool2 Configuration Change

This section describes changes to the FEPPGPool2 configuration.

List FEPPGPool2

Equivalent Kubernetes command: kubectl get FEPPGPool2 (-A)

This operation will list all FEPClusters in a namespace, or if the -A option is specified, will list all FEPClusters in all namespace.

Default output format:

Field	Value	Details
Name	.metadata.name	Name of pgpool2

Example)

```
# kubectl get feppgpool2 -A

NAMESPACE       NAME
namespace1      fep1-pgpool2
namespace2      fep2-pgpool2
```

Delete FEPPGPool2

Equivalent Kubernetes command: kubectl delete FEPPGPool2 <pgpool2_name>

This operation will remove the FEPPGPool2 by the pgpool2_name.

Update FEPPGPool2

Equivalent Kubernetes command: kubectl apply -f <new_spec>

Specify updated parameters in the format described in "1.3.4 Deploying Pgpool-II and Connect to FEPCluster from Operator". Only following parameters would change for Operations that can be performed here.

Custom Resource spec	Change Effect
.spec.count: n	Increase the number of nodes in the cluster to n.
.spec.serviceport	Change the TCP port for connecting to the Pgpool-II.
.spec.statusport	Change the TCP port for connecting to the PCP process.
.spec.limits.cpu	Change limits of cpus.
.spec.limits.memory	Change limits of memory.
.spec.requests.cpu	Change requests of cpus.
.spec.requests.memory	Change requests of memory.
.spec.fepclustername	Change fepcluster to connect.
.spec.customhba	Change pool_hba.conf file.
.spec.customparams	Change pgpool2 parameters
.spec.custompcp	Change pcp.conf file.
.spec.customsslkey	Change key content
.spec.customsslcert	Change the contents of the public x 509 certificate.
.spec.customsslcacert	Change the contents of the CA root certificate in PEM format.

Some of the customparams parameters, customhba and custompcp, require a restart of pgpool2.

Equivalent Kubernetes command: Kubectl apply -f <new_spec>

"pgpool2_restart" action type expects users to specify the name of the pgpool2 that they want to restart from.

Specify the metadata.Name of the FEPPGPool2 CR in the targetPgpool2Name section of the FEPAction CR, as below:

```
spec:
    targetPgpool2Name: fep1-pgpool2
    fepAction:
        type: pgpool2_restart
```

5.6 Scheduling Backup from Operator

Operational status confirm

Information about the backup can be found by running the command in the FEP backup container, as shown in the example below.

```
$ oc exec pod/fepserver-XXXXX -c FEPbackup - pgbackrest info
stanza: fepbackup
    status: ok
    cipher: none

    db (current)
        wal archive min/max (12-1): 000000010000000000000001/000000010000000000000005

        full backup: 20201125-025043F
            timestamp start/stop: 2020-11-25 02:50:43 / 2020-11-25 02:50:52
            wal start/stop: 000000010000000000000003 / 000000010000000000000003
            database size: 31.7MB, backup size: 31.7MB
            repository size: 3.9MB, repository backup size: 3.9MB

        incr backup: 20201125-025043F_20201125-025600I
            timestamp start/stop: 2020-11-25 02:56:00 / 2020-11-25 02:56:02
            wal start/stop: 000000010000000000000005 / 000000010000000000000005
            database size: 31.7MB, backup size: 24.3KB
            repository size: 3.9MB, repository backup size: 619B
            backup reference list: 20201125-025043F
```

Update FEPBackup

Equivalent Kubernetes command: kubectl apply -f <new_spec>

Specify updated parameters in the format described in "1.3.5 Scheduling Backup from Operator". Only following parameters would change for Operations that can be performed here.

Custom Resource spec	Change Effect
spec.schedule.num	Change the Number of Registered Backup Schedules
spec.scheduleN.schedule	Change the scheduled backup time
spec.scheduleN.type	Change the scheduled backup type
spec.pgBackrestParams	Change pgBackRest parameters

- Changes made during the backup are reflected from the next backup.
- Changes to the backup schedule do not affect the application.
- If you perform any of the following update operations, be sure to obtain a backup after the update.
  - When the master encryption key is updated with pgx_set_master_key
  - When the encryption passphrase for transparent data encryption is updated (can be updated by the tdeppassphrase parameter of FEPCluster CR)

5.7 Perform PITR and the Latest Backup Restore from Operator

Restore process can restore data by creating a CR (FEPRestore CR) for the restore as follows:

oc create -f [Custom Resource Files]

Example)

```
$oc create -f config/samples/postgres_v1_restore.yaml
```

There are two methods of restoring: restoring data to an existing FEPCluster or restoring data to a new FEPCluster.

When restoring to an existing FEPCluster, information such as the FEPCluster name, IP address, and various settings remain the same.

If you restore to a new FEPCluster, the FEPCluster name is the one you specified in CR and the new IP address is also given. If the setting value is not specified, the new cluster will inherit the settings from the restore source cluster, but you can change the settings to create a new cluster by specifying them in CR.

5.7.1 Setting Item

Refer to "FEP Restore Custom Resource Parameters" in the Reference for the items to be set in a custom resource file.

5.7.2 After Restore

Switching connections to the new cluster

The restore creates a new FEPCluster. If necessary, you need to set up Pgpool-II and change the access point of the application to the new cluster or the new Pgpool-II.

Backup data of the destination cluster

PITR restores to the pre-restore time are not possible, because the backup of the destination cluster begins after the restore completes.

5.8 Configure FEP to Perform MTLS

All three traffic can be secured by using TLS connection protected by certificates:

- Postgres traffic from Client Application to FEPCluster
- Patroni RESTAPI within FEPCluster
- Postgres traffic within FEPCluster (e.g. replication, rewind)

Here, we provide two methods to create certificates for securing the TLS connection and provide mutual authentication. The first method is to create and renew certificate manually. The second method is to use CertManager to create an automatically renew certificate.

5.8.1 Manual Certificate Management

Overview of Procedures

The procedures to enable MTLS communication are listed below:

1. Create a password for protecting CA private key (optional)
2. Create a self signed certificate as CA
3. Create Configmap to store CA certificate
4. Create a password for protecting FEP Server private key (optional)
5. Create FEP Server private key
6. Create FEP Server certificate signing request
7. Create FEP Server certificate signed by CA
8. Create TLS Secret to store FEP Server certificate and key
9. Create private key for Patroni
10. Create certificate signing request for Patroni
11. Create certificate signed by CA for Patroni
12. Create TLS secret to store Patroni certificate and key
13. Create private key for postgres user client certificate
14. Create certificate signing request for postgres user client certificate

15. Create client certificate for postgres user
16. Create TLS secret to store postgres certificate and key
17. Repeat step 14-17 for repluser and rewinduser

1. Create a password for protecting CA private key (optional)

```
oc create secret generic ca-private-key-password --from-literal=keypassword=0okm9ijn8uhb7ygv -n my-namespace
```

2. Create a self signed certificate as CA

```
openssl genrsa -aes256 -out myca.key 4096
Generating RSA private key, 4096 bit long modulus (2 primes)
.....................++++
.....................................................++++
e is 65537 (0x010001)
Enter pass phrase for myca.key: 0okm9ijn8uhb7ygv
Verifying - Enter pass phrase for myca.key: 0okm9ijn8uhb7ygv

cat << EOF > ca.cnf
[req]
distinguished_name=req_distinguished_name
x509_extensions=v3_ca
[v3_ca]
basicConstraints = critical, CA:true
keyUsage=critical,keyCertSign,digitalSignature,cRLSign
[req_distinguished_name]
commonName=Common Name
EOF

openssl req -x509 -new -nodes -key myca.key -days 3650 -out myca.pem -subj "/O=My Organization/OU=CA/CN=My Organization Certificate Authority" -config ca.cnf
Enter pass phrase for myca.key: abcdefghijk
```

3. Create Configmap to store CA certificate

```
oc create configmap cacert --from-file=ca.crt=myca.pem -n my-namespace
```

4. Create a password for protecting FEP Server private key (optional)

```
oc create secret generic mydb-fep-private-key-password --from-literal=keypassword=abcdefghijk -n my-namespace
```

5. Create FEP Server private key

```
openssl genrsa -aes256 -out fep.key 2048
Generating RSA private key, 2048 bit long modulus
..............................................+++
.......+++
e is 65537 (0x10001)
Enter pass phrase for fep.key: abcdefghijk
Verifying - Enter pass phrase for fep.key: abcdefghijk
```

6. Create FEP Server certificate signing request

```
cat << EOF > san.cnf
[SAN]
subjectAltName = @alt_names
[alt_names]
DNS.1 = *.my-namespace.pod
DNS.2 =  *.my-namespace.pod.cluster.local
DNS.3 = mydb-primary-svc
DNS.4 = mydb-primary-svc.my-namespace
DNS.5 = mydb-primary-svc.my-namespace.svc
DNS.6 = mydb-primary-svc.my-namespace.svc.cluster.local
DNS.7 = mydb-replica-svc
DNS.8 = mydb-replica-svc.my-namespace
DNS.9 = mydb-replica-svc.my-namespace.svc
DNS.10 = mydb-replica-svc.my-namespace.svc.cluster.local
EOF

openssl req -new -key fep.key -out fep.csr -subj "/CN=mydb-headless-svc" -reqexts SAN -config
<(cat /etc/pki/tls/openssl.cnf <(cat san.cnf))  # all in one line
```

7. Create FEP Server certificate signed by CA

```
openssl x509 -req -in fep.csr -CA myca.pem -CAkey myca.key -out fep.pem -days 365 -extfile
<(cat /etc/pki/tls/openssl.cnf <(cat san.cnf)) -extensions SAN -CAcreateserial  # all in one line
Signature ok
subject=/CN=mydb-headless-svc
Getting CA Private Key
Enter pass phrase for myca.key: 0okm9ijn8uhb7ygv
```

8. Create Secret to store FEP Server certificate and key

```
oc create secret generic mydb-fep-cert --from-file=tls.crt=fep.pem --from-file=tls.key=fep.key -n
my-namespace
```

9. Create private key for Patroni

At the moment, FEP container does not support password protected private key for Patroni.

```
openssl genrsa -out patroni.key 2048
Generating RSA private key, 2048 bit long modulus
.............................................+++
.......+++
e is 65537 (0x10001)
```

10. Create certificate signing request for Patroni

```
cat << EOF > san.cnf
[SAN]
subjectAltName = @alt_names
[alt_names]
DNS.1 = *.my-namespace.pod
DNS.2 = *.my-namespace.pod.cluster.local
DNS.3 = mydb-primary-svc
DNS.4 = mydb-primary-svc.my-namespace
DNS.5 = mydb-replica-svc
DNS.6 = mydb-replica-svc.my-namespace
DNS.7 = mydb-headless-svc
DNS.8 = mydb-headless-svc.my-namespace
```

```
EOF

openssl req -new -key patroni.key -out patroni.csr -subj "/CN=mydb-headless-svc" -reqexts SAN -
config <(cat /etc/pki/tls/openssl.cnf <(cat san.cnf))  # all in one line
```

11. Create certificate signed by CA for Patroni

```
openssl x509 -req -in patroni.csr -CA myca.pem -CAkey myca.key -out patroni.pem -days 365 -extfile
<(cat /etc/pki/tls/openssl.cnf <(cat san.cnf)) -extensions SAN -CAcreateserial  # all in one line
Signature ok
subject=/CN=mydb-headless-svc
Getting CA Private Key
Enter pass phrase for myca.key: 0okm9ijn8uhb7ygv
```

12. Create TLS secret to store Patroni certificate and key

```
oc create secret tls mydb-patroni-cert --cert=patroni.pem --key=patroni.key -n my-namespace
```

13. Create private key for postgres user client certificate

    At the moment, SQL client inside FEP server container does not support password protected certificate.

```
openssl genrsa -out postgres.key 2048
Generating RSA private key, 2048 bit long modulus
.............................................+++
.......+++
e is 65537 (0x10001)
```

14. Create certificate signing request for postgres user client certificate

```
openssl req -new -key postgres.key -out postgres.csr -subj "/CN=postgres"
```

15. Create client certificate for postgres user

```
openssl x509 -req -in postgres.csr -CA myca.pem -CAkey myca.key -out postgres.pem -days 365
```

16. Create TLS secret to store postgres certificate and key

```
oc create secret tls mydb-postgres-cert --cert=postgres.pem --key=postgres.key -n my-namespace
```

Repeat the same steps for repluser and rewinduser.

5.8.2 Automatic Certificate Management

There are many Certificate Management tools available in the public. In this example, we will use cert-manager for the purpose.

Note that certificates created in this example are not password protected.

Install cert-manager

```
oc create namespace cert-manager

oc apply -f https://github.com/jetstack/cert-manager/releases/download/v1.3.0/cert-manager.yaml
```

Create a Self Signed Issuer (This can be namespace specific or cluster wise)

This example creates an Issuer, that can create self signed certificate, in namespace my-namespace.

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Issuer
metadata:
  name: selfsigned-issuer
  namespace: my-namespace
spec:
  selfSigned: {}
EOF
```

Create a Self Signed CA certificate using selfsigned-issuer

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: cacert
  namespace: my-namespace
spec:
  subject:
    organizations:
      - My Organization
    organizationalUnits:
      - CA
  commonName: "My Organization Certificate Authority"
  duration: 87600h
  isCA: true
  secretName: cacert
  issuerRef:
    name: selfsigned-issuer
EOF
```

The above command will create a self signed Root certificate and private key stored in the Kubernetes secret "cacert" in namespace my-namespace.

Create a CA Issuer with above certificate

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Issuer
metadata:
  name: ca-issuer
  namespace: my-namespace
spec:
  ca:
    secretName: cacert
EOF
```

Create FEP Server certificate using above CA Issuer

Assuming FEPCluster name is mydb in namespace my-namespace.

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: mydb-fep-cert
  namespace: my-namespace
spec:
  subject:
  commonName: "mydb-headless-svc"
  dnsNames:
  - "*.my-namespace.pod"
  - "*.my-namespace.pod.cluster.local"
  - "mydb-primary-svc"
  - "mydb-primary-svc.my-namespace"
  - "mydb-primary-svc.my-namespace.svc"
  - "mydb-primary-svc.my-namespace.svc.cluster.local"
  - "mydb-replica-svc"
  - "mydb-replica-svc.my-namespace"
  - "mydb-replica-svc.my-namespace.svc"
  - "mydb-replica-svc.my-namespace.svc.cluster.local"
  duration: 8760h
  usages:
  - server auth
  secretName: mydb-fep-cert
  issuerRef:
    name: ca-issuer
EOF
```

Create Patroni certificate using above CA Issuer

Assuming FEPCluster name is mydb in namespace my-namespace.

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: mydb-patroni-cert
  namespace: my-namespace
spec:
  subject:
  commonName: "mydb-headless-svc"
  dnsNames:
  - "*.my-namespace.pod"
  - "*.my-namespace.pod.cluster.local"
  - "*.mydb-primary-svc"
  - "*.mydb-primary-svc.my-namespace"
  - "*.mydb-replica-svc "
  - "*.mydb-replica-svc.my-namespace"
  duration: 8760h
  usages:
  - server auth
  secretName: mydb-patroni-cert
  issuerRef:
    name: ca-issuer
EOF
```

Create postgres user client certificate

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: mydb-postgres-cert
  namespace: my-namespace
spec:
  subject:
  commonName: "postgres"
  duration: 8760h
  usages:
  - client auth
  secretName: mydb-postgres-cert
  issuerRef:
    name: ca-issuer
EOF
```

Create repluser user client certificate

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: mydb-repluser-cert
  namespace: my-namespace
spec:
  subject:
  commonName: "repluser"
  duration: 8760h
  usages:
  - client auth
  secretName: mydb-repluser-cert
  issuerRef:
    name: ca-issuer
EOF
```

Create rewinduser user client certificate

```
cat << EOF | oc apply -f -
apiVersion: cert-manager.io/v1
kind: Certificate
metadata:
  name: mydb-rewinduser-cert
  namespace: my-namespace
spec:
  subject:
  commonName: "rewinduser"
  duration: 8760h
  usages:
  - client auth
  secretName: mydb-rewinduser-cert
  issuerRef:
    name: ca-issuer
EOF
```

5.8.3 Deploy FEPCluster with MTLS support

Deploy FEPCluster with manual certificate management

Use the following yaml as an example to deploy a FEPCluster with Manual Certificate Management. MTLS related parameters are highlighted in Red.

```
apiVersion: fep.fujitsu.io/v2
kind: FEPCluster
metadata:
  name: mydb
  namespace: my-namespace
spec:
  fep:
    usePodName: true
    patroni:
      tls:
        certificateName: mydb-patroni-cert
        caName: cacert
    postgres:
      tls:
        certificateName: mydb-fep-cert
        caName: cacert
        privateKeyPassword: mydb-fep-private-key-password
    forceSsl: true
    podAntiAffinity: false
    mcSpec:
      limits:
        cpu: 500m
        memory: 700Mi
      requests:
        cpu: 200m
        memory: 512Mi
    customAnnotations:
      allDeployments: {}
    servicePort: 27500
    image:
      image: 'quay.io/fujitsu/fujitsu-enterprise-postgres-12-server:ubi8-12-1.1'
      pullPolicy: IfNotPresent
    sysExtraLogging: false
    podDisruptionBudget: false
    instances: '3'
    syncMode: 'on'
  fepChildCrVal:
    customPgAudit: |
      # define pg audit custom params here to override defaults.
      # if log volume is not defined, log_directory should be
      # changed to '/database/userdata/data/log'
      [output]
      logger = 'auditlog'
      log_directory = '/database/log/audit'
      [rule]
    customPgHba: |
      # define pg_hba custom rules here to be merged with default rules.
      # TYPE     DATABASE        USER        ADDRESS        METHOD
      hostssl    all             all         0.0.0.0/0      cert
      hostssl    replication     all         0.0.0.0/0      cert
    customPgParams: >+
      # define custom postgresql.conf parameters below to override defaults.

      # Current values are as per default FEP deployment


      shared_preload_libraries='pgx_datamasking,vci,pgaudit,pg_prewarm'

      session_preload_libraries='vci,pg_prewarm'
```

```
max_prepared_transactions = 100

max_worker_processes = 30

max_connections = 100

work_mem = 1MB

maintenance_work_mem = 12MB

shared_buffers = 128MB

effective_cache_size = 384MB

checkpoint_completion_target = 0.8


# tcp parameters

tcp_keepalives_idle = 30

tcp_keepalives_interval = 10

tcp_keepalives_count = 3


# logging parameters in default fep installation

# if log volume is not defined, log_directory should be

# changed to '/database/userdata/data/log'

log_directory = '/database/log'

log_filename = 'logfile-%a.log'

log_file_mode = 0600

log_truncate_on_rotation = on

log_rotation_age = 1d

log_rotation_size = 0

log_checkpoints = on

log_line_prefix = '%e %t [%p]: [%l-1] user=%u,db=%d,app=%a,client=%h'

log_lock_waits = on

log_autovacuum_min_duration = 60s

logging_collector = on

pgaudit.config_file='/opt/app-root/src/pgaudit-cfg/pgaudit.conf'

log_replication_commands = on

log_min_messages = WARNING

log_destination = stderr
```

```
    # vci parameters in default fep installation

    vci.enable = on

    vci.maintenance_work_mem = 256MB

    vci.max_local_ros = 64MB

    vci.force_max_parallelism = off


    # wal_archive parameters in default fep installation

    archive_mode = on

    archive_command = '/bin/true'

    wal_level = replica

    max_wal_senders = 12

    wal_keep_segments = 64

  storage:
    dataVol:
      size: 2Gi
      storageClass: nfs-client
    walVol:
      size: 1200Mi
      storageClass: nfs-client
    logVol:
      size: 1Gi
      storageClass: nfs-client
  sysUsers:
    pgAdminPassword: admin-password
    pgdb: mydb
    pgpassword: mydbpassword
    pguser: mydbuser
    pgrepluser: repluser
    pgreplpassword: repluserpwd
    pgAdminTls:
      certificateName: mydb-postgres-cert
      caName: cacert
      sslMode: prefer

    pgrepluserTls:
      certificateName: mydb-repluser-cert
      caName: cacert
      sslMode: prefer

    pgRewindUserTls:
      certificateName: mydb-rewinduser-cert
      caName: cacert
      sslMode: prefer

    tdepassphrase: tde-passphrase
  systemCertificates:
    key: |-
      -----BEGIN RSA PRIVATE KEY-----
      MIIEowIBAAKCAQEA0DFkImha8CIJiVcwXbBPlL+/DmS9/ipRhQQHxfO5x7jSOnse
      IHdFd6+Qx2GX8KAiAhVykf6kfacwBYTATU1xDgwWTm82KVRPh+kZDIj2wPcJr14m
      mTP6I6a2mavUgDhezHc9F8/dchYj3cw81X0kU6xamqrKQYlxQH48NkI0qcwhO6sK
```

```
  AHF4eWfCr8Ot44xADIA1JcU2CS1RKSZEtURZ+30Py+j907Enjp1YR33ZKUHw30pU
  9dpIneyfXBN/pT6cX3MetYwtgmpV/pHqY8pbxqGfoyRhgQDsSRCl4dtlecaZeZ4j
  uTOotcPkZELHP6eu8gaLtycG9lpbAMQl5w0r8QIDAQABAoIBACq213qPuoimExrQ
  fqXaNJmqNYK4fJqXCB6oUwf0F1u4ubkx5V532hLSPHwLs+a0lAWlbNozSoBVOu8G
  64VwrA9bv3/cJVqZZ6/UzUTbHPU+Ogh24qhwF5QU8kXZEUI1To3YsPoftalgjX9G
  Ff0fLcLVC8nL3K9RiaDXxXbEYpWrYu39M3FCpAXAzV2PrNxsP9PKyNWHnBPc08z5
  tFj45/bHn+j31AVVvgWtqz0pLks57hc4Q7yW/2RoRYq2md1KI709OLNwtkWEOVqb
  qnraorh2TwGnNaOB5oX5/lJvKtlq778fw96jGqykBr0+DKozj9rlr1OGgYOKDwlD
  nsZJPAECgYEA+Oqf/fxtPdsNGiaL2Z/heewvtaxjw/WoEVBFECb6/y4Ro7aux9nB
  16FcVi79Cwfp0UTJ7cnZvYSmBk5GWEObEIAeo61lvm/QeltM5+usAPd5/TcHXLYe
  92OnXmq7h3F4UXEkMayak8Lpu/TdmR5uOaL+m4aEu+XMY5tlxqDCnyECgYEA1h4X
  jCPi7Ja5CHK7a2Ud4TL2DNpIBE6GSK9iQ+0xFL6TsiK2Sfu6n8mx2sh+Jm0KHTiE
  /gWHdHQZSSWiuULfHoYEq3Rq8S6Av3GsGtRSpO03j7BE8C20Vpt0FnNTjZmdzf2/
  YZxc5KuYLh9qeY7Y7ceOsWA8JckDgMHPYzyLAtECgYBALD0TPgDr8Y1vMIDdmlqH
  FF04eTk/TBYIYKltgJ81KqthibeFzp4q+W7UyUhzj5a4XQOySlfYhFpJReTc3JEd
  r+o2SH3ymuEkqmUpZZjyptrMbWN4g3t4TDjaHqo6QQbD+GdcZyNy9M1Np9N5pl7E
  fUEm14dg6d3H0Ehs7QVAAQKBgQDRUx3mLXc9oKRINBIyDerGLJILQqLBQxtYl81T
  ZuFizGWL8w+PCIAMkpxDrVpWqqcGpiiuRi2ElbPapOaOg2epaY/LJscd/j5z6uc8
  W3JoNljpKoRa4fO578Pv5tM6TYHOzlF5Veoiy/a8sI3hRNuiqkM/+TsUHY5FJDRh
  aeDk4QKBgCOHIevvR+MWuwakzD6lNCbb8H6fvZ3WRAT8BYYz3wW9YfnV4J4uh/Bl
  moWYgIK2UpkrhA8scMUC790FoybQeParQ35x7Jl91bmTKkCqsX63fyqqYhx3SXRl
  JSktmH4E2cGmosZisjB7COKHR32w0J5JCgaGInQxjldbGrwhZQpn
  -----END RSA PRIVATE KEY-----
crt: |-
  -----BEGIN CERTIFICATE-----
  MIID2DCCAsCgAwIBAgIQDfFYteD4kZj4Sko2iy1IJTANBgkqhkiG9w0BAQsFADBX
  MRgwFgYDVQQKEw9NeSBPcmdhbml6YXRpb24xCzAJBgNVBAsTAkNBMS4wLAYDVQQD
  EyVNeSBPcmdhbml6YXRpb24gQ2VydGlmaWNhdGUgQXV0aG9yaXR5MB4XDTIxMDQy
  MDAwMDQ1OVoXDTIxMDQyMDAxMDQ1OVowGDEWMBQGA1UEAwwNKi5jaGctcHRjLnBv
  ZDCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANAxZCJoWvAiCYlXMF2w
  T5S/vw5kvf4qUYUEB8Xzuce40jp7HiB3RXevkMdhl/CgIgIVcpH+pH2nMAWEwE1N
  cQ4MFk5vNilUT4fpGQyI9sD3Ca9eJpkz+iOmtpmr1IA4Xsx3PRfP3XIWI93MPNV9
  JFOsWpqqykGJcUB+PDZCNKnMITurCgBxeHlnwq/DreOMQAyANSXFNgktUSkmRLVE
  Wft9D8vo/dOxJ46dWEd92SlB8N9KVPXaSJ3sn1wTf6U+nF9zHrWMLYJqVf6R6mPK
  W8ahn6MkYYEA7EkQpeHbZXnGmXmeI7kzqLXD5GRCxz+nrvIGi7cnBvZaWwDEJecN
  K/ECAwEAAaOB3jCB2zATBgNVHSUEDDAKBggrBgEFBQcDATAMBgNVHRMBAf8EAjAA
  MIG1BgNVHREEga0wgaqCCWxvY2FsaG9zdIIbKi5jaGctcHRjLnBvZC5jbHVzdGVy
  LmxvY2FsghMqLm15ZGItaGVhZGxlc3Mtc3Zjghsq Lm15ZGItaGVhZGxlc3Mtc3Zj
  LmNoZy1wdGOCHyoubXlkYi1oZWFkbGVzcy1zdmMuY2hnLXB0Yy5zdmOCLSoubXlk
  Yi1oZWFkbGVzcy1zdmMuY2hnLXB0Yy5zdmMuY2x1c3Rlci5sb2NhbDANBgkqhkiG
  9w0BAQsFAAOCAQEALnhliDflu+BHp5conq4dXBwD/Ti2YR5TWQixM/0a6OD4KecZ
  MmaLl0T+OJJvA/j2IufZpc7dzEx5mZDkR2CRmoq10qZXqCRTrBZSXm6ARQWoYpeg
  9c0l4f8roxrkMGUKVPTKUwAvbnNYhD2l6PlBPwMpkMUfqFaSEXMaPyQKhrtQxdpH
  WjuS54OP0lm0peYu/yiaD98LtrTXnb6jch84SKf6Vii4HAVQyMeJaW+dpkqcI2+V
  Q4fkwWYSJy8BNcmXCwvHDLDy+s4EXWvHafhusuUhcp4HyMb1A6hd5hJhgFSnEvLy
  kLA0L9LaScxee6V756Vt9TN1NGjwmwyQDOhnQQ==
  -----END CERTIFICATE-----
cacrt: |-
  -----BEGIN CERTIFICATE-----
  MIIDXDCCAkSgAwIBAgIRAMPzF3BNFxT9HWE+NXlFQjQwDQYJKoZIhvcNAQELBQAw
  VzEYMBYGA1UEChMPTXkgT3JnYW5pemF0aW9uMQswCQYDVQQLEwJDQTEuMCwGA1UE
  AxMlTXkgT3JnYW5pemF0aW9uIENlcnRpZmljYXRlIEF1dGhvcml0eTAeFw0yMTA0
  MTkwNDQ0MjNaFw0zMTA0MTcwNDQ0MjNaMFcxGDAWBgNVBAoTD015IE9yZ2FuaXph
  dGlvbjELMAkGA1UECxMCQ0ExLjAsBgNVBAMTJU15IE9yZ2FuaXphdGlvbiBDZXJ0
  aWZpY2F0ZSBBdXRob3JpdHkwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIB
  AQC5t6CS23Glk65YMw5e4i4xHldyxkCZS67w/6LWqeIlYKmFAaEl83WWy8MHUpOb
  4mahtUafEzDEOX6+URf72J8m0voldQ5FYr1AyUOyX8U90wGFqhbEgKRqt7vZEwIe
  2961fwqHh6917zI4xmt5W6ZJ5dBQVtkhzB+Pf7O6KBYjHoCnBBkfNVzsfZQ/1hnR
  0UzimfAc7Ze+UNwhXJhinFRJ3YuR+xiOTpPkl1GXPhLgFSQhekz4KepcbQEQKejb
  jg0dum1oBYIXZTSSbiO9rNmfUVLB5DcV0vZbSrGxLjWLBt5U8N2xf2d1bvkQW+bw
  Kklf9OG26bAi27tujurzN3r3AgMBAAGjIzAhMA4GA1UdDwEB/wQEAwICpDAPBgNV
  HRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwUAA4IBAQAM0CN3n5C/KOT4uZ4ewwKK
  rHmANBPVM9u6MJBO8U62HcqLeoCuDFeU8zmUjLHjsQaPX64mJZlR7T5y52gEKO5A
```

```
        0qsBz3pg/vJ5DJTv0698+1Q1hB9k3smQdksAim19FZqysB7J4zK/+8aJ/q2kIFvs
        Jk3ekwQdQ3xfggklBQVuf76gr1v0uYlPtfPffPlfcGZ06Im6mqbajenXoR1PxPB0
        +zyCS8DkgPtDulplruwvXCFMYw9TPbzXKlt7tlsqRXogYLnxWJDzM1nOYCnD+rDm
        qxenV9Ir8RqZ0XSYuUyzRka5N4dhIhrzTAiNdeU5gzynXOz67u/Iefz1iK9ZcdE3
        -----END CERTIFICATE-----
```

Deploy FEPCluster with automatic certificate management

Use the following yaml as an example to deploy a FEPCluster with Automatic Certificate Management. MTLS related parameters are highlighted in Red.

```
apiVersion: fep.fujitsu.io/v2
kind: FEPCluster
metadata:
  name: mydb
  namespace: my-namespace
spec:
  fep:
    usePodName: true
    patroni:
      tls:
        certificateName: mydb-patroni-cert
    postgres:
      tls:
        certificateName: mydb-fep-cert
    forceSsl: true
    podAntiAffinity: false
    mcSpec:
      limits:
        cpu: 500m
        memory: 700Mi
      requests:
        cpu: 200m
        memory: 512Mi
    customAnnotations:
      allDeployments: {}
    servicePort: 27500
    image:
      image: 'quay.io/fujitsu/fujitsu-enterprise-postgres-12-server:ubi8-12-1.1'
      pullPolicy: IfNotPresent
    sysExtraLogging: false
    podDisruptionBudget: false
    instances: '3'
    syncMode: 'on'
  fepChildCrVal:
    customPgAudit: |
      # define pg audit custom params here to override defaults.
      # if log volume is not defined, log_directory should be
      # changed to '/database/userdata/data/log'
      [output]
      logger = 'auditlog'
      log_directory = '/database/log/audit'
      [rule]
    customPgHba: |
      # define pg_hba custom rules here to be merged with default rules.
      # TYPE     DATABASE        USER        ADDRESS        METHOD
      hostssl    all             all         0.0.0.0/0      cert
      hostssl    replication     all         0.0.0.0/0      cert
    customPgParams: >+
      # define custom postgresql.conf parameters below to override defaults.

      # Current values are as per default FEP deployment
```

```
shared_preload_libraries='pgx_datamasking,vci,pgaudit,pg_prewarm'

session_preload_libraries='vci,pg_prewarm'

max_prepared_transactions = 100

max_worker_processes = 30

max_connections = 100

work_mem = 1MB

maintenance_work_mem = 12MB

shared_buffers = 128MB

effective_cache_size = 384MB

checkpoint_completion_target = 0.8


# tcp parameters

tcp_keepalives_idle = 30

tcp_keepalives_interval = 10

tcp_keepalives_count = 3


# logging parameters in default fep installation

# if log volume is not defined, log_directory should be

# changed to '/database/userdata/data/log'

log_directory = '/database/log'

log_filename = 'logfile-%a.log'

log_file_mode = 0600

log_truncate_on_rotation = on

log_rotation_age = 1d

log_rotation_size = 0

log_checkpoints = on

log_line_prefix = '%e %t [%p]: [%l-1] user=%u,db=%d,app=%a,client=%h'

log_lock_waits = on

log_autovacuum_min_duration = 60s

logging_collector = on

pgaudit.config_file='/opt/app-root/src/pgaudit-cfg/pgaudit.conf'

log_replication_commands = on
```

```
    log_min_messages = WARNING

    log_destination = stderr


    # vci parameters in default fep installation

    vci.enable = on

    vci.maintenance_work_mem = 256MB

    vci.max_local_ros = 64MB

    vci.force_max_parallelism = off


    # wal_archive parameters in default fep installation

    archive_mode = on

    archive_command = '/bin/true'

    wal_level = replica

    max_wal_senders = 12

    wal_keep_segments = 64

  storage:
    dataVol:
      size: 2Gi
      storageClass: nfs-client
    walVol:
      size: 1200Mi
      storageClass: nfs-client
    logVol:
      size: 1Gi
      storageClass: nfs-client
  sysUsers:
    pgAdminPassword: admin-password
    pgdb: mydb
    pgpassword: mydbpassword
    pguser: mydbuser
    pgrepluser: repluser
    pgreplpassword: repluserpwd
    pgAdminTls:
      certificateName: mydb-postgres-cert
      sslMode: verify-full

    pgrepluserTls:
      certificateName: mydb-repluser-cert
      sslMode: verify-full

    pgRewindUserTls:
      certificateName: mydb-rewinduser-cert
      sslMode: verify-full

    tdepassphrase: tde-passphrase
  systemCertificates:
    key: |-
      -----BEGIN RSA PRIVATE KEY-----
      MIIEowIBAAKCAQEA0DFkImha8CIJiVcwXbBPlL+/DmS9/ipRhQQHxfO5x7jSOnse
      IHdFd6+Qx2GX8KAiAhVykf6kfacwBYTATU1xDgwWTm82KVRPh+kZDIj2wPcJr14m
```

```
  mTP6I6a2mavUgDhezHc9F8/dchYj3cw81X0kU6xamqrKQYlxQH48NkI0qcwhO6sK
  AHF4eWfCr8Ot44xADIA1JcU2CS1RKSZEtURZ+30Py+j907Enjp1YR33ZKUHw30pU
  9dpIneyfXBN/pT6cX3MetYwtgmpV/pHqY8pbxqGfoyRhgQDsSRCl4dtlecaZeZ4j
  uTOotcPkZELHP6eu8gaLtycG9lpbAMQl5w0r8QIDAQABAoIBACq213qPuoimExrQ
  fqXaNJmqNYK4fJqXCB6oUwf0F1u4ubkx5V532hLSPHwLs+a0lAWlbNozSoBVOu8G
  64VwrA9bv3/cJVqZZ6/UzUTbHPU+Ogh24qhwF5QU8kXZEUI1To3YsPoftalgjX9G
  Ff0fLcLVC8nL3K9RiaDXxXbEYpWrYu39M3FCpAXAzV2PrNxsP9PKyNWHnBPc08z5
  tFj45/bHn+j31AVVvgWtqz0pLks57hc4Q7yW/2RoRYq2md1KI709OLNwtkWEOVqb
  qnraorh2TwGnNaOB5oX5/lJvKtlq778fw96jGqykBr0+DKozj9rlr1OGgYOKDwlD
  nsZJPAECgYEA+Oqf/fxtPdsNGiaL2Z/heewvtaxjw/WoEVBFECb6/y4Ro7aux9nB
  16FcVi79Cwfp0UTJ7cnZvYSmBk5GWEObEIAeo61lvm/QeltM5+usAPd5/TcHXLYe
  92OnXmq7h3F4UXEkMayak8Lpu/TdmR5uOaL+m4aEu+XMY5tlxqDCnyECgYEA1h4X
  jCPi7Ja5CHK7a2Ud4TL2DNpIBE6GSK9iQ+0xFL6TsiK2Sfu6n8mx2sh+Jm0KHTiE
  /gWHdHQZSSWiuULfHoYEq3Rq8S6Av3GsGtRSpO03j7BE8C20Vpt0FnNTjZmdzf2/
  YZxc5KuYLh9qeY7Y7ceOsWA8JckDgMHPYzyLAtECgYBALD0TPgDr8Y1vMIDdmlqH
  FF04eTk/TBYIYKltgJ81KqthibeFzp4q+W7UyUhzj5a4XQOySlfYhFpJReTc3JEd
  r+o2SH3ymuEkqmUpZZjyptrMbWN4g3t4TDjaHqo6QQbD+GdcZyNy9M1Np9N5pl7E
  fUEm14dg6d3H0Ehs7QVAAQKBgQDRUx3mLXc9oKRINBIyDerGLJILQqLBQxtYl81T
  ZuFizGWL8w+PCIAMkpxDrVpWqqcGpiiuRi2ElbPapOaOg2epaY/LJscd/j5z6uc8
  W3JoNljpKoRa4fO578Pv5tM6TYHOzlF5Veoiy/a8sI3hRNuiqkM/+TsUHY5FJDRh
  aeDk4QKBgCOHIevvR+MWuwakzD6lNCbb8H6fvZ3WRAT8BYYz3wW9YfnV4J4uh/Bl
  moWYgIK2UpkrhA8scMUC790FoybQeParQ35x7Jl91bmTKkCqsX63fyqqYhx3SXRl
  JSktmH4E2cGmosZisjB7COKHR32w0J5JCgaGInQxjldbGrwhZQpn
  -----END RSA PRIVATE KEY-----
crt: |-
  -----BEGIN CERTIFICATE-----
  MIID2DCCAsCgAwIBAgIQDfFYteD4kZj4Sko2iy1IJTANBgkqhkiG9w0BAQsFADBX
  MRgwFgYDVQQKEw9NeSBPcmdhbml6YXRpb24xCzAJBgNVBAsTAkNBMS4wLAYDVQQD
  EyVNeSBPcmdhbml6YXRpb24gQ2VydGlmaWNhdGUgQXV0aG9yaXR5MB4XDTIxMDQy
  MDAwMDQ1OVoXDTIxMDQyMDAxMDQ1OVowGDEWMBQGA1UEAwwNKi5jaGctcHRjLnBv
  ZDCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANAxZCJoWvAiCYlXMF2w
  T5S/vw5kvf4qUYUEB8Xzuce40jp7HiB3RXevkMdhl/CgIgIVcpH+pH2nMAWEwE1N
  cQ4MFk5vNilUT4fpGQyI9sD3Ca9eJpkz+iOmtpmr1IA4Xsx3PRfP3XIWI93MPNV9
  JFOsWpqqykGJcUB+PDZCNKnMITurCgBxeHlnwq/DreOMQAyANSXFNgktUSkmRLVE
  Wft9D8vo/dOxJ46dWEd92SlB8N9KVPXaSJ3sn1wTf6U+nF9zHrWMLYJqVf6R6mPK
  W8ahn6MkYYEA7EkQpeHbZXnGmXmeI7kzqLXD5GRCxz+nrvIGi7cnBvZaWwDEJecN
  K/ECAwEAAaOB3jCB2zATBgNVHSUEDDAKBggrBgEFBQcDATAMBgNVHRMBAf8EAjAA
  MIG1BgNVHREEga0wgaqCCWxvY2FsaG9zdIIbKi5jaGctcHRjLnBvZC5jbHVzdGVy
  LmxvY2FsghMqLm15ZGItaGVhZGxlc3Mtc3Zjghsq Lm15ZGItaGVhZGxlc3Mtc3Zj
  LmNoZy1wdGOCHyoubXlkYi1oZWFkbGVzcy1zdmMuY2hnLXB0Yy5zdmOCLSoubXlk
  Yi1oZWFkbGVzcy1zdmMuY2hnLXB0Yy5zdmMuY2x1c3Rlci5sb2NhbDANBgkqhkiG
  9w0BAQsFAAOCAQEALnhliDflu+BHp5conq4dXBwD/Ti2YR5TWQixM/0a6OD4KecZ
  MmaLl0T+OJJvA/j2IufZpc7dzEx5mZDkR2CRmoq10qZXqCRTrBZSXm6ARQWoYpeg
  9c0l4f8roxrkMGUKVPTKUwAvbnNYhD2l6PlBPwMpkMUfqFaSEXMaPyQKhrtQxdpH
  WjuS54OP0lm0peYu/yiaD98LtrTXnb6jch84SKf6Vii4HAVQyMeJaW+dpkqcI2+V
  Q4fkwWYSJy8BNcmXCwvHDLDy+s4EXWvHafhusuUhcp4HyMb1A6hd5hJhgFSnEvLy
  kLA0L9LaScxee6V756Vt9TN1NGjwmwyQDOhnQQ==
  -----END CERTIFICATE-----
cacrt: |-
  -----BEGIN CERTIFICATE-----
  MIIDXDCCAkSgAwIBAgIRAMPzF3BNFxT9HWE+NXlFQjQwDQYJKoZIhvcNAQELBQAw
  VzEYMBYGA1UEChMPTXkgT3JnYW5pemF0aW9uMQswCQYDVQQLEwJDQTEuMCwGA1UE
  AxMlTXkgT3JnYW5pemF0aW9uIENlcnRpZmljYXRlIEF1dGhvcml0eTAeFw0yMTA0
  MTkwNDQ0MjNaFw0zMTA0MTcwNDQ0MjNaMFcxGDAWBgNVBAoTD015IE9yZ2FuaXph
  dGlvbjELMAkGA1UECxMCQ0ExLjAsBgNVBAMTJU15IE9yZ2FuaXphdGlvbiBDZXJ0
  aWZpY2F0ZSBBdXRob3JpdHkwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIB
  AQC5t6CS23Glk65YMw5e4i4xHldyxkCZS67w/6LWqeIlYKmFAaEl83WWy8MHUpOb
  4mahtUafEzDEOX6+URf72J8m0voldQ5FYr1AyUOyX8U90wGFqhbEgKRqt7vZEwIe
  2961fwqHh6917zI4xmt5W6ZJ5dBQVtkhzB+Pf7O6KBYjHoCnBBkfNVzsfZQ/1hnR
  0UzimfAc7Ze+UNwhXJhinFRJ3YuR+xiOTpPkl1GXPhLgFSQhekz4KepcbQEQKejb
  jg0dum1oBYIXZTSSbiO9rNmfUVLB5DcV0vZbSrGxLjWLBt5U8N2xf2d1bvkQW+bw
  Kklf9OG26bAi27tujurzN3r3AgMBAAGjIzAhMA4GA1UdDwEB/wQEAwICpDAPBgNV
  HRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwUAA4IBAQAM0CN3n5C/KOT4uZ4ewwKK
```

```
rHmANBPVM9u6MJBO8U62HcqLeoCuDFeU8zmUjLHjsQaPX64mJZlR7T5y52gEKO5A
0qsBz3pg/vJ5DJTv0698+1Q1hB9k3smQdksAim19FZqysB7J4zK/+8aJ/q2kIFvs
Jk3ekwQdQ3xfggklBQVuf76gr1v0uYlPtfPffPlfcGZ06Im6mqbajenXoR1PxPB0
+zyCS8DkgPtDulplruwvXCFMYw9TPbzXKlt7tlsqRXogYLnxWJDzM1nOYCnD+rDm
qxenV9Ir8RqZ0XSYuUyzRka5N4dhIhrzTAiNdeU5gzynXOz67u/Iefz1iK9ZcdE3
-----END CERTIFICATE-----
```

5.8.4 Configurable Parameters

To enable MTLS, make changes to the following parameters.

Key	Value	Details
spec.fep.usePodName	True	For MTLS, this key must be defined and set to true. For TLS connection without MTLS, it can be omitted. However, it is recommended to set this to true as well.
spec.fep.patroni.tls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for Patroni REST API. For MTLS Patroni REST API communication, this key must be defined. The private key cannot be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.fep.patroni.tls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.fep.patroni.postgres.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for Postgres server. For MTLS Postgres communication, this key must be defined. The private key can be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.fep.postgres.tls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.fep.postgres.tls.privateKeyPassword	<secret-name>	Name of Kubernetes secret that contains the password for the private key for Postgres Server.
spec.fep.sysUsers.pgAdminTls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for

Key	Value	Details
		“postgres” user. For MTLS Postgres communication, this key must be defined. The private key cannot be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.fep.sysUsers.pgAdminTls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.fep.sysUsers.pgAdminTls.sslMode	verify-full	For MTLS, this value must be set to verify-full. If only TLS is required, this can be set to verify-ca or prefer.
spec.fep.sysUsers.pgrepluserTls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for “repluser” user. For MTLS Postgres communication, this key must be defined. The private key cannot be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.fep.sysUsers.pgrepluserTls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.fep.sysUsers.pgrepluserTls.sslMode	verify-full	For MTLS, this value must be set to verify-full. If only TLS is required, this can be set to verify-ca or prefer.
spec.fep.sysUsers.pgRewindUserTls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for “rewinduser” user. For MTLS Postgres communication, this key must be defined. The private key cannot be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.fep.sysUsers.pgRewindUserTls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.

Key	Value	Details
spec.fep.sysUsers.pgRewindUserTls.sslMode	verify-full	For MTLS, this value must be set to verify-full. If only TLS is required, this can be set to verify-ca or prefer.

It is also required to customize pg_hba.conf to perform MTLS. Below are two possible settings.

spec.fep.customPgHba	hostssl all all 0.0.0.0/0 cert hostssl replication all 0.0.0.0/0 cert

The above setting will force FEP server to perform certification authentication. At the same time verify the authenticity of client certificate.

spec.fep.customPgHba	hostssl all all 0.0.0.0/0 md5 clientcert=verify-full hostssl replication repluser 0.0.0.0/0 md5 clientcert=verify-full

The above setting will foce FEP server to perform md5 authentication as well as verifying the authenticity of client certificate.

5.8.5 Certification Rotation

All certificates are bounded by the time limit. At certain time, it needs to be renewed. We recommended to renew the certificate when it reach 3/4 of its life cycle or as soon as possible if it is compromised. When a certificate is renew, we need to rotate it inside the FEP server container. At the moment, FEP server container can does not support automatic certificate rotation. Depends on which certificate has renewed, there are different procedures to handle that.

Patroni Certificate Rotation

When Patroni certificate is renewed, we have to re-deploy each and every POD for FEP server container to pick up the new certificate. There is a down time on FEPCluster.

FEP Server Certificate Rotation

When FEP Server certificate is renewed, we can use FEPAction CR to trigger a reload of the database and FEP server will pick up the new certificate with no interruption to service.

Client certification Rotation

When any of the client certificate is renewed, FEP server container internally will use the new certificate next time it establishes a connection to FEP server. However, to avoid any unexpected interruption to service, it is recommended to re-deploy each and every POD as soon as possible.

5.9 Assigned Resources for Operator Containers

The following resources are allocated by default to the operator containers provided by this product.

```
resources:
limits :
  cpu: 2
  memory: 1536Mi
requests :
  cpu: 500m
  memory: 768Mi
```

If there is only one FEPCluster custom resource managed by an operator, it can be operated with the resource assigned by default. However, when deploying and operating multiple FEPCluster custom resources, change the assigned resource of the operator container.

Note

If you have changed the resource, the resource value will revert to the default value after the operator version upgrade. Therefore, change the resource again after upgrading the operator.

5.9.1 How to Change Assigned Resources

Describes how to change the resources assigned to an operator container.

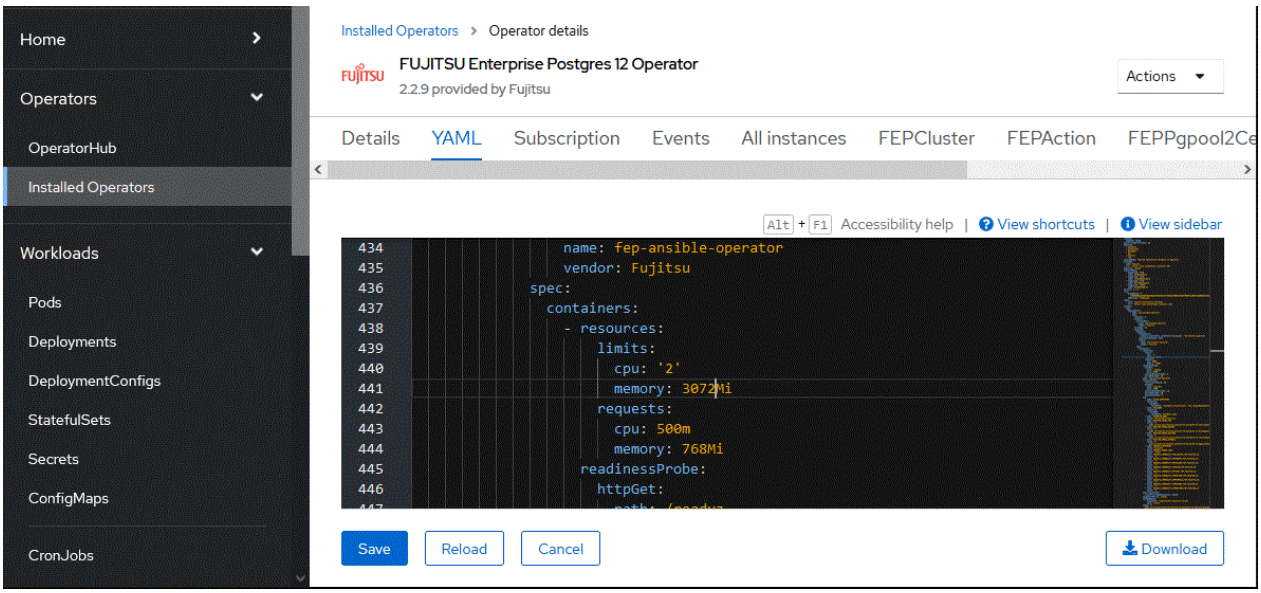
When updating resources assigned to an operator container, the operator container is recreated. At this time, the operation of already built containers such as FEPCluster will not stop.

Edit the ClusterServiceVersion (CSV) to change the resources assigned to the operator container.

Editing the CSV "spec.install.spec.deployments[0].spec.template.spec.containers[0].resources" will recreate the operator container and apply the specified resources.

When editing CSV from the OCP GUI console

Click [Installed Operators] in the menu item under Operators and select the installed operator. On the [YAML] tab, edit the specified part of the allocation resource and click [Save].



When editing CSV from the CUI console using the OC client

Check the CSV name of the installed operator with the "oc get" command.

```
$ oc get csv
NAME                           DISPLAY                                       VERSION   REPLACES   PHASE
fep-ansible-operator.v2.2.9    FUJITSU Enterprise Postgres 12 Operator   2.2.9               Succeeded
```

Edit the CSV with the "oc edit" command.

```
$ oc edit csv fujitsu-ansible-operator.v2.2.9
```

Chapter 6 Abnormality

This chapter describes the actions to take when an error occurs in the database or an application, while FEP is operating.

Depending on the type of error, recover from the backed-up material, reserve capacity, check the operator log, and check the FEP log.

6.1 Handling of Data Abnormalities

Recover the database cluster from the backup immediately prior to failure in any of the following cases:

- A hardware failure occurs on the data storage disk or the backup data storage disk.
- If the data on the disk is logically corrupted and the database does not work correctly
- Data corruption caused by user error

Refer to "5.7 Perform PITR and the Latest Backup Restore from Operator" for backup instructions.

6.2 Handling when the Capacity of the Data Storage Destination or Transaction Log Storage Destination is Insufficient

If you run out of space in the data storage location, first check if there are any unnecessary files on the disk, and then delete them so that you can continue working.

If deleting unnecessary files does not solve the problem, you may need to migrate the data to a larger disk.

Use a backup restore to migrate data.

6.3 What to do when the Capacity of the Backup Data Storage Area is Insufficient

If you run out of space in the backup data destination, first check the disk for unnecessary files, and then delete the unnecessary files. Or reduce the backup retention generation.

6.4 Handling Access Abnormalities When Instance Shutdown Fails

If an instance fails to start or stop, refer to the Operator log and the FEP log to determine the cause.

For checking the operator log and the FEP log, refer to Collecting Fault Investigation Information.

6.5 Collection of Failure Investigation Information

If the cause of the trouble that occurred during the construction or operation of the environment is not identified, information for the initial investigation is collected.

I will explain how to collect information for the initial investigation.

- Product log
- Operator log

Product log

FEP log

Get into the container and collect the log.

The log location is specified by log_directory in the custom resource FEP Clusterspec.startupValues.customPgParam parameter. The default is/database/log.

Pgpool-II log

Get into the container and collect the log.

The log location is/var/log/pgpool/pool.log.

Operator log

Check the operator log as follows.

Verification Example

```
$oc get po
NAME                                       READY   STATUS    RESTARTS   AGE
fep-ansible-operator-7dc5fd9bf7-4 smzk   1/1     Running   0          20m
```

How to check the log

```
$oc logs pod fep-ansible-operator-7dc5fd9bf7-4 smzk -c manager
```

The log will be output to the console. Please check the file output by redirection.

Appendix A Quantitative Values and Limitations

A.1 Quantitative Values

Refer to the FUJITSU Software Enterprise Postgres Installation and Setup Guide for Server.

A.2 Limitations

Note

If you log in to a container and edit the configuration file directly, restarting the container may undo your changes.

If you want to change the settings, modify the custom resource files as described in "5.1 Configuration Change" and reapply. Depending on the parameters to be changed, the container may be redeployed. Refer to "5.1 Configuration Change" for details of the parameters.

Unavailable FEP features

Since FEP server container is based on other components (like UBI and Patroni), there are certain limitations that doesn't allow it to be 100% functionally capable to VM based server instance. The known limitations are as below.

No	Limitation	Reason for Limitation	Description
1	No Support for JIT	Since UBI8 is not having requisite LLVM libraries	It is not possible to enable JIT in postgreql.conf. Impact for the customer is that they are not able to achieve maximum performance capabilities on given CPU and memory
2	FEP parallelism improvements	Since UBI8 is not hosting dstat binaries	FEP parallelism improvement is to restrict number of parallel workers in case the CPU is already busy because of other tasks/processes. It is unlikely to have too much impact on FEP container, since container is running only one process.
3	Crypto Express cards are not supported	IBM LinuxOne doesn't support CryptoExpress cards in Openshift container platform at this stage.	FEP TDEz extension cannot be used on LinuxOne Openshift environment. However, User can still use TDE on both LinuxOne Openshift environment as well as Azure (x86) Openshift environment.
4	No Support for Oracle foreign data wrapper	Oracle foreign data wrapper has dependency on Instant Client package, which is not available.	Oracle InstantClient package is not redistributed by FUJITSU Enterprise Postgres leading to this limitation. The functionality of Oracle Foreign data wrapper is not available to FUJITSU Enterprise Postgres on Openshift environment.

Fixed parameter

Some parameters cannot be changed. Refer to "1.3.5.2 Parameters that cannot be Set".

FEP features that needs to be set when using

Refer to "1.3.7 FEP Unique Feature Enabled by Default".

FUJITSU

FUJITSU Enterprise Postgres 12 for Kubernetes

Reference Guide

Linux

J2UL-REFE-04ENZ0(00)
October 2022

Preface

Purpose of this document

This document is a reference, and explains parameter.

Intended readers

This document is aimed at people who manage and operate.

Readers of this document are also assumed to have general knowledge of:

- Linux
- Kubernetes
- Containers
- Operators

Structure of this document

This document is structured as follows:

Chapter 1 Custom Resource Parameters

Explains the parameter.

Abbreviations

The following abbreviations are used in this manual:

Full Name	Abbreviations
FUJITSU Software Enterprise Postgres for Kubernetes FUJITSU Software Enterprise Postgres	FEP or FUJITSU Enterprise Postgres
Transparent Data Encryption	TDE
Custom Resource	CR
Custom Resource Definition	CRD
Persistent Volume	PV

Trademarks

- Linux is a registered trademark or trademark of Mr. Linus Torvalds in the U.S. and other countries.
- Red Hat and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.
- S/390 is a registered trademark of International Business Machines Corporation ("IBM") in the U.S. and other countries.

Other product and company names mentioned in this manual are the trademarks or registered trademarks of their respective owners.

Export restrictions

If this document is to be exported or provided overseas, confirm legal requirements for the Foreign Exchange and Foreign Trade Act as well as other laws and regulations, including U.S. Export Administration Regulations, and follow the required procedures.

Issue date and version


```
Edition 4.0: October 2022
Edition 3.0: November 2021
Edition 2.0: April 2021
Edition 1.0: March 2021
```

Copyright

Contents

Chapter 1 Custom Resource Parameters

This chapter explains the parameter.

1.1 FEPCluster Parameter

Equivalent Kubernetes command: kubectl apply -f FEPClusterCR.yaml

This operation will create a FEPCluster with supplied information in FEPClusterCR.yaml.

Initial configuration and subsequent changes to FEP Cluster are done through FEP Cluster CR.

Field	Default	Details
metadata.name	new-fep	Name for the Cluster. FEP server container will use this value for Patroni scope. e.g. new-fep
spec.fep.customAnnotation.allDeployments	{} (*)	Contents under this are optional. User can remove {} and add multiple key-value pairs. All of these pair will be added to annotations of FEP statefulSet and FEP Pods. If left at default, no annotation is added to Pods and statefulSets
spec.fep.image.image	<omitted>	FEP server container image to be used quay.io/fujitsu/fujitsu-enterprise-postgres-12-server:ubi8-12-1.1 It is optional. Image line is omitted by default. In such a case, it will pick up URL of image from operator container environment. If you specify the image, Operator will take that image to deploy fep container
spec.fep.image.pullPolicy	IfNotPresent	
spec.fep.mcSpec.limits	cpu: 500m memory: 700Mi	
spec.fep.mcSpec.requests	cpu: 200m memory: 512Mi	
spec.fep.sysExtraLogging	false	To turn extra debugging on, set value to true It can be turned on/off at any time
spec.fep.instances	1	Number of nodes in the cluster, including both Master and Replicas. In Example CR, it is kept at 1 for certification. However, user can change it to 3 for 1 master and 2 replicas.
spec.fep.servicePort	27500	TCP port for FEP master service
spec.fep.syncMode	off	Replication Mode:

Field	Default	Details
		off - async replication on - sync replication
spec.fep.forceSsl	true	Controls that the communication to the server should only be via SSL. Changes are reflected in pg_hba.conf
spec.fep.podAntiAffinity	false	Defines that all the pods should not run on same worker node
spec.fep.podDisruptionBudget	false	Allows to maintain minimum number of pods of an application even when some nodes are voluntarily drained for say, maintenance
spec.fep.usePodName		Setting this key to true will make internal POD communication, both Patroni and Postgres to use hostname, instead of IP address. This is important for TLS as the hostname of the POD is predictable and can be used to create Server Certificate, whereas IP address is unpredictable and cannot be used to create Certificate. There is no negative effect setting this key to true even if TLS (i.e. Server Certificate) is not used.
spec.fep.patroni.tls.certificateName		This points to Kubernetes TLS secret that contains the certificate for Patroni. The certificate itself is stored in the key tls.crt. This field is optional. When this key is set, the Operator will ignore the value in systemCertificates
spec.fep.patroni.tls.caName		This points to Kubernetes configmap that contains additional CA for Patroni to verify client. The CA is stored in the key ca.crt. This field is optional.
spec.fep.postgres.tls.certificateName		This points to Kubernetes TLS secret that contains the certificate for Postgres. The certificate itself is stored in the key tls.crt. This field is optional. When this value is set, Operator will ignore the value in systemCertificates
spec.fep.postgres.tls.caName		This points to Kubernetes configmap that contains additional CA for Postgres to verify client. The CA is stored in the key ca.crt. This field is optional.
spec.fepChildCrVal.backup.image.image	<omitted>	FEP backup container image to be used quay.io/fujitsu/fujitsu-enterprise-postgres-12-backup:ubi8-12-1.1' It is optional. Image line is omitted by default. In such a case, it will pick up URL of image from operator container environment.

Field	Default	Details
		If you specify the image, Operator will take that image to deploy backup container
spec.fepChildCrVal.backup.image.pullPolicy	IfNotPresent	
spec.fepChildCrVal.backup.mcSpec.limits	cpu: 0.2 memory: "300Mi"	
spec.fepChildCrVal.backup.mcSpec.requests	cpu: 0.1 memory: "200Mi"	
spec.fepChildCrVal.backup.pgbackrestParams	[global] repo1-retention-full=7 repo1-retention-full-type=time log-path=/database/log/backup	"|" When nothing is specified, and the parameter set in pgbackrest.conf is described from the line below.
spec.fepChildCrVal.backup.schedule.num	2	Number of schedules to set The maximum number of backup schedules is 5.
spec.fepChildCrVal.backup.schedule.schedule1.schedule	15 0 * * 0	Backup *schedule* in cron format. The date and time is UTC time.
spec.fepChildCrVal.backup.schedule.schedule1.type	full	full: Perform a full backup (Back up the contents of the database cluster). incr - Perform an incremental backup (Back up only the database cluster files that were changed to the last backup migration).
spec.fepChildCrVal.backup.schedule.schedule2.schedule	15 0 * * 1-6	Backup schedule in cron format. The date and time is UTC time.
spec.fepChildCrVal.backup.schedule.schedule2.type	incr	full: Perform a full backup (Back up the contents of the database cluster). incr - Perform an incremental backup (Back up only the database cluster files that were changed to the last backup migration).
spec.fepChildCrVal.customPgAudit	[output] logger = 'auditlog' log_directory = '/database/log/audit' log_truncate_on_rotation = on log_filename = 'pgaudit-%a.log' log_rotation_age = 1d log_rotation_size = 0 [rule]	PgAudit file content
spec.fepChildCrVal.customPgHba	# define pg_hba custom rules here to be merged with default rules.	Entries to be inserted into pg_hba.conf

Field	Default	Details
	# TYPE DATABASE USER ADDRESS METHOD	

spec.fepChildCrVal.customPgParams	# define custom postgresql.conf parameters below to override defaults.
	# Current values are as per default FEP deployment
	shared_preload_libraries='pgx_datamasking,pg_prewarm,pg_stat_statements'
	session_preload_libraries='pg_prewarm'
	max_prepared_transactions = 100
	max_worker_processes = 30
	max_connections = 100
	work_mem = 1MB
	maintenance_work_mem = 12MB
	shared_buffers = 128MB
	effective_cache_size = 384MB
	checkpoint_completion_target = 0.8
	# tcp parameters
	tcp_keepalives_idle = 30
	tcp_keepalives_interval = 10
	tcp_keepalives_count = 3
	# logging parameters in default fep installation
	# if log volume is not defined, log_directory should be
	# changed to '/database/userdata/data/log'
	log_directory = '/database/log'
	log_filename = 'logfile-%a.log'
	log_file_mode = 0600
	log_truncate_on_rotation = on
	log_rotation_age = 1d
	log_rotation_size = 0
	log_checkpoints = on
	log_line_prefix = '%e %t [%p]: [%l-1] user=%u,db=%d,app=%a,client=%h'
	log_lock_waits = on
	log_autovacuum_min_duration = 60s
	logging_collector = on
	pgaudit.config_file='/opt/app-root/src/pgaudit-cfg/pgaudit.conf'
	log_replication_commands = on
		Postgres configuration in postgresql.conf

Field	Default	Details

	log_min_messages = WARNING
	log_destination = stderr
	# wal_archive parameters in default fep installation
	archive_mode = on
	archive_command = 'pgbackrest --stanza=backupstanza --config=/database/userdata/pgbackrest.conf archive-push %p'
	wal_level = replica
	max_wal_senders = 12
	wal_keep_segments = 64
	track_activities = on
	track_counts = on	

spec.fepChildCrVal.storage.dataVol.size	2Gi (**)	Size of data volume. Data volume must be specified
spec.fepChildCrVal.storage.dataVol.storageClass	<omitted> (*)	StorageClass for data volume: When this line is omitted, the PV created will use default storage class in the Kubernetes cluster
spec.fepChildCrVal.storage.dataVol.accessModes	<omitted> (*)	accessModes for data volume: Specified as an array of accessModes e.g. [ReadWriteMany] If omitted, it will be treated as [ReadWriteOnce]
spec.fepChildCrVal.storage.walVol.size	1200Mi (**)	Size of WAL volume. WAL volume must be specified
spec.fepChildCrVal.storage.walVol.storageClass	<omitted> (*)	StorageClass for WAL volume: When this line is omitted, the PV created will use default storage class in the Kubernetes cluster
spec.fepChildCrVal.storage.walVol.accessModes	<omitted> (*)	accessModes for WAL volume: Specified as an array of accessModes e.g. [ReadWriteMany] If omitted, it will be treated as [ReadWriteOnce]
spec.fepChildCrVal.storage.tablespaceVol.size	512Mi (**)	Size of tablespace volume. This volume is optional and can be omitted
spec.fepChildCrVal.storage.tablespaceVol.storageClass	<omitted> (*)	StorageClass for tablespace volume: When this line is omitted, the PV created will use default storage class in the Kubernetes cluster

Field	Default	Details
spec.fepChildCrVal.storage.tablespaceVol.accessModes	<omitted> (*)	accessModes for tablespace volume: Specified as an array of accessModes e.g. [ReadWriteMany] If omitted, it will be treated as [ReadWriteOnce]
spec.fepChildCrVal.storage.archivewalVol.size	1Gi (**)	Size of archivewal volume. This volume is optional and can be omitted
spec.fepChildCrVal.storage.archivewalVol.storageClass	<omitted> (*)	StorageClass for Archived WAL volume: When this line is omitted, the PV created will use default storage class in the Kubernetes cluster When the number of instance is more than 1 and backup is not done on S3, both archivewalVol and backupVol must be hosted on Shared storage such as NFS with respective storageClass
spec.fepChildCrVal.archivewalVol.accessModes	<omitted> (*)	accessModes for Archived WAL volume: Specified as an array of accessModes e.g. [ReadWriteMany] If omitted, it will be treated as [ReadWriteOnce] When the number of instance is more than 1 and backup is not done on S3, both archivewalVol and backupVol must be hosted on Shared storage such as NFS with accessMode set to [ReadWriteMany]
spec.fepChildCrVal. logVol.size	1Gi (**)	Size of log volume. This volume is optional and can be omitted
spec.fepChildCrVal.logVol.storageClass	<omitted> (*)	StorageClass for log volume: When this line is omitted, the PV created will use default storage class in the Kubernetes cluster
spec.fepChildCrVal.logVol.accessModes	<omitted> (*)	accessModes for log volume: Specified as an array of accessModes e.g. [ReadWriteMany] If omitted, it will be treated as [ReadWriteOnce]
spec.fepChildCrVal.backupVol.size	2Gi (**)	Size of backup volume. This volume is optional and can be omitted
spec.fepChildCrVal.backupVol.storageClass	<omitted>	StorageClass for backup volume:

Field	Default	Details
	(*)	When this line is omitted, the PV created will use default storage class in the Kubernetes cluster When the number of instance is more than 1 and backup is not done on S3, both archivewalVol and backupVol must be hosted on Shared storage such as NFS with respective storageClass
spec.fepChildCrVal.backupVol.accessModes	<omitted> (*)	accessModes for backup volume: Specified as an array of accessModes e.g. [ReadWriteMany] If omitted, it will be treated as [ReadWriteOnce] When the number of instance is more than 1 and backup is not done on S3, both archivewalVol and backupVol must be hosted on Shared storage such as NFS with accessMode set to [ReadWriteMany]
spec.fepChildCrVal.sysUsers.pgAdminPassword	admin-password	Password for user "postgres"
spec.fepChildCrVal.sysUsers.pgdb	mydb (*)	Database to be created during provisioning
spec.fepChildCrVal.sysUsers.pguser	mydbuser (*)	Database user to be created during provisioning
spec.fepChildCrVal.sysUsers.pgpassword	mydbpassword	Password for database user pguser
spec.fepChildCrVal.sysUsers.pgrepluser	repluser (*)	Database user for replication
spec.fepChildCrVal.sysUsers.pgreplpassword	repluserpwd	Password for database user repluser
spec.fepChildCrVal.sysUsers.tdepassphrase	tde-passphrase	TDE keystore passphrase
spec.fepChildCrVal.sysUsers.pgRewindUser	rewind_user	Database user for Rewind
spec.fepChildCrVal.sysUsers.pgRewindUserPassword	rewind_password	Password for database user rewinduser
spec.fepChildCrVal.sysUsers.pgAdminTls.certificateName		This points to Kubernetes TLS secret that contains the certificate of Postgres user “postgres”. Patroni will use this for certificate authentication. The certificate itself is stored in the key tls.crt. This field is optional.
spec.fepChildCrVal.sysUsers.pgAdminTls.caName		This points to Kubernetes configmap that contains additional CA the client use to verify a server certificate. The CA is stored in the key ca.crt. This field is optional.

Field	Default	Details
spec.fepChildCrVal.sysUsers.pgAdminTls.sslMode	prefer	Specify the type of TLS negotiation with the server. - disable - allow - prefer - require - verify-ca - verify-full
spec.fepChildCrVal.sysUsers.pgreplUserTls.certificateName		This points to Kubernetes TLS secret that contains the certificate of Postgres user “repluser”. Patroni will use this for certificate authentication. The certificate itself is stored in the key tls.crt. This field is optional.
spec.fepChildCrVal.sysUsers.pgreplUserTls.caName		This points to Kubernetes configmap that contains additional CA the client use to verify a server certificate. The CA is stored in the key ca.crt. This field is optional.
spec.fepChildCrVal.sysUsers.pgreplUserTls.sslMode	prefer	Specify the type of TLS negotiation with the server. - disable - allow - prefer - require - verify-ca - verify-full
spec.fepChildCrVal.sysUsers.pgRewindUserTls.certificateName		This points to Kubernetes TLS secret that contains the certificate of Postgres user “rewinduser”. Patroni will use this for certificate authentication. The certificate itself is stored in the key tls.crt. This field is optional.
spec.fepChildCrVal.sysUsers.pgRewindUserTls.caName		This points to Kubernetes configmap that contains additional CA the client use to verify a server certificate. The CA is stored in the key ca.crt. This field is optional.
spec.fepChildCrVal.sysUsers.pgRewindUserTls.sslMode	prefer	Specify the type of TLS negotiation with the server. - disable - allow - prefer - require - verify-ca

Field	Default	Details
		- verify-full
spec.fepChildCrVal.systemCertificates.key		Use spec.fep.postgres.tls specification instead.
spec.fepChildCrVal.systemCertificates.crt		Use spec.fep.postgres.tls specification instead.
spec.fepChildCrVal.systemCertificates.cacrt		Use spec.fep.postgres.tls specification instead.

Note

- (*) - These parameters can be specified only at creation time and should not be changed. Any change to these parameters will be ignored and will not have any effect on FEP cluster functioning.
- (**) - The storage volumes size can be increased provided underlying storage supports the operation. Optional volumes can be specified only at initial FEP cluster creation. If an optional volume is added later, operator will ignore it and no action will be taken.
- User should do or remove unsupported CR changes manually.
- spec.fep.postgres.tls CR specification should be used instead of spec.fepChildCrVal.systemCertificates. The lateral spec can still be used, however spec.fep.postgres.tls gives better flexibility to control MTLS access of the cluster.
- Either spec.fep.postgres.tls specification (old specification) or spec.fepChildCrVal.systemCertificates should be used. They should not be used interchangeable.
- Server certificate specified under spec.fep.postgres.tls can be rotated by changing the secret and executing reload (e.g. using FEPAction); however for others specified in the CR, it is required to do restart of the PoDs

While in running state - following value will dynamically appear in the FEPCluster to reflect the cluster status

Field name	Details
fepClusterReady	Will be true or false to reflect if the whole cluster is ready. Kubernetes cluster information is fetched to check number of instances 'READY' & 'RUNNING' is equal to number of Configured instances.

1.2 Custom Resource Parameters

This section explains the Custom Resource Parameters.

1.2.1 FEPCluster Custom Resource Parameters

Category	Details
CRD Name	FEPCluster
Definition	apiVersion: apiextensions.k8s.io/v1beta1 kind: CustomResourceDefinition metadata: name: FEPClusters.fep.fujitsu.io spec: group: fep.fujitsu.io names: kind: FEPCluster

Category	Details
	listKind: FEPClusterList plural: fepclusters singular: fepcluster shortNames: - fac scope: Namespaced conversion: strategy: None subresources: status: {} versions: - name: v2 served: true storage: true
Operations	Create: kubectl create fepcluster (or kubectl create -f fepcluster.yaml) Delete: kubectl delete fepcluster (or kubectl delete fepcluster <clusername>) Update: kubectl patch fepcluster (or kubectl apply -f fepcluster.yaml) List: kubectl get fepcluster

FEPCluster CR Example

```
apiVersion: fep.fujitsu.io/v2
kind: FEPCluster
metadata:
  name: new-fep
  namespace: new-fep
spec:
  fep:
    forceSsl: true
    image:
      image: 'quay.io/fujitsu/fujitsu-enterprise-postgres-12-server:ubi8-12-1.1'
      pullPolicy: IfNotPresent
    mcSpec:
      limits:
        cpu: 500m
        memory: 700Mi
      requests:
        cpu: 200m
        memory: 512Mi
    podAntiAffinity: true
    podDisruptionBudget: true
    instances: '3'
    servicePort: 27500
    syncMode: 'on'
    sysExtraLogging: false
  fepChildCrVal:
    customPgAudit: |
      # define pg audit custom params here to override defaults.
      # if log volume is not defined, log_directory should be
      # changed to '/database/userdata/data/log'
```

```
      [output]
      logger = 'auditlog'
      log_directory = '/database/log/audit'
      log_truncate_on_rotation = on
      log_filename = 'pgaudit-%a.log'
      log_rotation_age = 1d
      log_rotation_size = 0
      [rule]

    customPgHba: |
      # define pg_hba custom rules here to be merged with default rules.
      # TYPE     DATABASE        USER        ADDRESS        METHOD

    customPgParams: |+
      # define custom postgresql.conf parameters below to override defaults.
      # Current values are as per default FEP deployment
      shared_preload_libraries='pgx_datamasking,vci,pgaudit,pg_prewarm'
      session_preload_libraries='vci,pg_prewarm'
      max_prepared_transactions = 100
      max_worker_processes = 30
      max_connections = 100
      work_mem = 1MB
      maintenance_work_mem = 12MB
      shared_buffers = 128MB
      effective_cache_size = 384MB
      checkpoint_completion_target = 0.8
      pgx_global_metacache = 10MB

      # tcp parameters
      tcp_keepalives_idle = 30
      tcp_keepalives_interval = 10
      tcp_keepalives_count = 3

      # logging parameters in default fep installation
      # if log volume is not defined, log_directory should be
      # changed to '/database/userdata/data/log'
      log_directory = '/database/log'
      log_filename = 'logfile-%a.log'
      log_file_mode = 0600
      log_truncate_on_rotation = on
      log_rotation_age = 1d
      log_rotation_size = 0
      log_checkpoints = on
      log_line_prefix = '%e %t [%p]: [%l-1] user=%u,db=%d,app=%a,client=%h'
      log_lock_waits = on
      log_autovacuum_min_duration = 60s
      logging_collector = on
      pgaudit.config_file='/opt/app-root/src/pgaudit-cfg/pgaudit.conf'
      log_replication_commands = on
      log_min_messages = WARNING
      log_destination = stderr

      # vci parameters in default fep installation
      vci.enable = on
      vci.maintenance_work_mem = 256MB
      vci.max_local_ros = 64MB
      vci.force_max_parallelism = off

      # wal_archive parameters in default fep installation
      archive_mode = on
     archive_command = 'pgbackrest --stanza=backupstanza --config=/database/userdata/pgbackrest.conf
archive-push %p'
      wal_level = replica
```

```
    max_wal_senders = 10
    wal_keep_segments = 64
    wal_sender_timeout = 60s

  backup:
    image:
      image: 'quay.io/fujitsu/fujitsu-enterprise-postgres-12-backup:ubi8-12-1.1'
      pullPolicy: IfNotPresent
    mcSpec:
      limits:
        cpu: 200m
        memory: 300Mi
      requests:
        cpu: 100m
        memory: 200Mi
    pgbackrestParams: |
      # define custom pgbackrest.conf parameters below to override defaults.
      [global]
      repo1-retention-full = 30
      repo1-retention-full-type = time
    preScript: " "
    postScript: " "
    schedule:
      num: 2
    schedule1:
      schedule: "15 0 * * 0"
      type: "full"
    schedule2:
      schedule: "15 0 * * 1-6"
      type: "incr"
    schedule3:
      schedule: " "
      type: " "
    schedule4:
      schedule: " "
      type: " "
    schedule5:
      schedule: " "
      type: " "

  storage:
    dataVol:
      size: 2Gi
    tablespaceVol:
      size: 512Mi
    walVol:
      size: 1200Mi
    archivewalVol:
      size: 1Gi
    backupVol:
      size: 2Gi
    logVol:
      size: 1Gi

  sysUsers:
    pgAdminPassword: admin-password
    pgdb: mydb
    pgpassword: mydbpassword
    pguser: mydbuser
    pgrepluser: repluser
    pgreplpassword: repluserpwd
    tdepassphrase: tde-passphrase
```

```
systemCertificates:
  key: |-
    -----BEGIN RSA PRIVATE KEY-----
    MIIEpAIBAAKCAQEAvhL4D/01Lmm/RY3nu+jgLOdLYEEg0wqMxhsyPRb43paWSFlp
    gX1CNAPzlEtns4LVGSd6n7TqV73MfZ4lNHpuVtjWVTS6wtf7dQj7bbKeWQCDF5bK
    QpekP5HAv/5uQ4Bx154FppJvmMX6CtUBm9ici8X7M1GrPQ5uir7kj8SrUkSpXdKp
    wqyuEufvbenaYCI8KapBcTAsRIMjWufWngriln4b8ZYiVh0mcHLrX8HWTmQJvqBh
    9laEwgn/KItpWQVp8dcZli1t+H6gBECd6n4q0/v1x0J2MoVK63Q+zZ7Y3ox5qSNN
    +/Kgacht916AcEzIoJ52pA4vneLwErKX6kJMRwIDAQABAoIBAF2vH9FRrlq4CGyR
    6vw1Zfj776z7rOAYPRaP5Q0zO2sKsfvrBhOq12yn3fDj0bMq8zm4ubnqA+9HP3lS
    72eUSLpJmirZGIxcxDYFPVfuSBn6JKMF0Z9M+snSXzzCfTqMHpB19LcLSjH0sq+Q
    GYDlHRPPe2bqBArOCDIesK0j9IVRNItWOzZCarjlzjlyNSS4vPaEjAySW/XxuRzi
    A5smx2zXVm55+FjJpc2+H4Q+Rd+0AdLKrRAOyGCLMG3X5iYgwBTjzRKmdHJoIsnI
    em+kJYxChSaJFK+2uzJ1+L1W9d+7CtEDxIyMKxvlTaF79agzJI7MvotGGvnvLaTP
    KTTODAECgYEA6h5h/OP9oB+1WM4xhQAmtnpwWOhQPKjMAYI0XZfzWMKBrzEKKk8p
    k1bZIM24xUzMSb/hKvcqcraxYN2lIUmGCDspbu0xMG6vNzqjAH0TtK8HhYOihhKl
    hGVYV3vToTeJns3SL39lIedhCOgVx1bPkHDS6lV97Hdd9WIanp/8RwECgYEAz9bC
    A+aMbe3+lxLaoQ2j+54QaE/TbP+bbuP6Rc1H8OP1C1ZEPT1p3I4+mAlTyMq8Reit
    4CmSvvpHWXpONnNVCyhHerWCySxyV5Mcp33RARX5xNl09TTJEgqoH2daTieM/KY6
    rakqerh7cwSGX0IXcB+N0OApBs7BJph2g3FNm0cCgYEAjneot2TiLTO+fmkTd1UN
    OdQuU9wjH5a0dmKOjTnvat8KXdrgzbhYm4GpJa4qt12xn1t1oVjBawdDz6dxWl1M
    g+vEne0XFtr0Iw66rIxwlm5AjH37Q81LDdNCPBJtSVjrlUi4lBDZMRWFVg8tWZ25
    N7OAlfsqYuCMu8tUWZ0PvwECgYA7xefGd2erteaiTCnUZ7fhhXPyjAKiNmDdY3N2
    37Lw9J5kxEqb0i2/4Kjf0M7n0GaVNHXNIieyBQjAEwyrXD/5FXx5LfqPINlBlm2H
    Pgf95/QNSPz6CFRLfAUeAvtVvsotXyFBEIXHBYd7bLG4c6mJ9YkzqUQjURL7pp1u
    8AcBwQKBgQCi149QxQsbnakltQsYG5e+vo49GBjrpA7HeZDGq5ojGShAIMqOJSuv
    t+dwgfloxAw65jkBH6hNqVLF4xdZPV6Ka2bHNCPGK7b0rwqQVyvjp1Ml4dgIhPN6
    tNZgz3cDtHwb3VCHN3APGLcIZDazZbjOMqLWBq/euhdcLnyuB9jBww==
    -----END RSA PRIVATE KEY-----
  crt: |-
    -----BEGIN CERTIFICATE-----
    MIIEHjCCAwagAwIBAgIJANroZLqsw8hNMA0GCSqGSIb3DQEBCwUAMBYxFDASBgNV
    BAMMCzEwLjEzMS4wLjk5MB4XDTIwMTEwODE0MzUyNloXDTQ4MDMyNjE0MzUyNlow
    aDELMAkGA1UEBhMCOTExFDASBgNVBAgMC01haGFyYXNodHJhMQ0wCwYDVQQHDARQ
    dW5lMRAwDgYDVQQKDAdGdWppdHN1MQwwCgYDVQQLDANDT0UxFDASBgNVBAMMCzEw
    LjEzMS4wLjk5MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvhL4D/01
    Lmm/RY3nu+jgLOdLYEEg0wqMxhsyPRb43paWSFlpgX1CNAPzlEtns4LVGSd6n7Tq
    V73MfZ4lNHpuVtjWVTS6wtf7dQj7bbKeWQCDF5bKQpekP5HAv/5uQ4Bx154FppJv
    mMX6CtUBm9ici8X7M1GrPQ5uir7kj8SrUkSpXdKpwqyuEufvbenaYCI8KapBcTAs
    RIMjWufWngriln4b8ZYiVh0mcHLrX8HWTmQJvqBh9laEwgn/KItpWQVp8dcZli1t
    +H6gBECd6n4q0/v1x0J2MoVK63Q+zZ7Y3ox5qSNN+/Kgacht916AcEzIoJ52pA4v
    neLwErKX6kJMRwIDAQABo4IBGzCCARcwRgYDVR0jBD8wPYAUtx0d+PZMXgb1RMKW
    4eoagmIUrjGhGqQYMBYxFDASBgNVBAMMCzEwLjEzMS4wLjk5ggkAtVepKtIe4D4w
    CQYDVR0TBAIwADALBgNVHQ8EBAMCBDAwHQYDVR0lBBYwFAYIKwYBBQUHAwEGCCsG
    AQUFBwMCMIGVBgNVHREEgY0wgYqCCmt1YmVybmV0ZXOCEmt1YmVybmV0ZXMuZGVm
    YXVsdIIWa3ViZXJuZXRlcy5kZWZhdWx0LnN2Y4Iea3ViZXJuZXRlcy5kZWZhdWx0
    LnN2Yy5jbHVzdGVygiRrdWJlcm5ldGVzLmRlZmF1bHQuc3ZjLmNsdXN0ZXIubG9j
    YWyHBAqDAGOHBKweVVkwDQYJKoZIhvcNAQELBQADggEBADBIpMWGoidMAriPoFE/
    f3Iwq2VMwr/NBg8ZgQd8l7+IIHooMP+1/nj2juy7enyrlFPiqRvhXADBkvZIhro2
    4c+1bDbEbW0HqdwUnRWwwBEei1lXq4m6voWXXIA+At7fdiK7Dr7fo2OX5nDgW1Tw
    btxCoqdUee/m9EgvHLmOLhuI3E1654zP6FVB2rlXN/oXeEzefPE18VqSvk7eZ/hR
    adqpK3yt3lLeFVQzqfXzcoxOCM7Bt0txVNN4a9NwBoF8abaHxVoKI3rZlxQFpkn+
    RCHx6QtaVDnLJ8jlykXCv8i7Qz+3Nwh0zszl3aM8Rt3Pd+PRjc9VGg5kXBMoRZi2
    xuo=
    -----END CERTIFICATE-----
  cacrt: |-
    -----BEGIN CERTIFICATE-----
    MIIC/zCCAeegAwIBAgIJALVXqSrSHuA+MA0GCSqGSIb3DQEBCwUAMBYxFDASBgNV
    BAMMCzEwLjEzMS4wLjk5MB4XDTIwMTEwODE0MjUyOVoXDTQ4MDMyNjE0MjUyOVow
    FjEUMBIGA1UEAwwLMTAuMTMxLjAuOTkwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
    ggEKAoIBAQDL7FlI69s5QY+NaZMEHVw4cksMdmxsNerSdBpZaHaTVLSGod5SeqK8
    EGL0NPua2KccjZRitDLdYx8FlTrPqTiE9N7QlEDoiLi2AMrP8DEykGA3O4JtOXzs
    yGGwmodgdqJORhcwmYEfiKej6483Ahy3bXORG4WbmouId1Ou7CmrH6VXTcP8sgmc
```

```
OLEEc9n33C/Ymw1lggJk6fM/ysZKSIc2wiePFPVo86tXJ5k8pRpGJZfqfGJ8OIdx
EEyW1r7GRnNm1ZQVD7A4meNarA14Bc/6b/uBtL+WySW7wvqUIua+e4Sp1X4mMbj8
IqZLEzsgvaKpDFT02+jQiVqMCD8OG2jHAgMBAAGjUDBOMB0GA1UdDgQWBBS3HR34
9kxeBvVEwpbh6hqCYhSuMTAfBgNVHSMEGDAWgBS3HR349kxeBvVEwpbh6hqCYhSu
MTAMBgNVHRMEBTADAQH/MA0GCSqGSIb3DQEBCwUAA4IBAQAiXOC/idXXeygT8UzH
k3biEs3iRWajDAlWVONOpnj8q75F4zIaGhCKvU/kfdOg9cwVy3GJq5+1LhR8qtcC
5o5iOHtS+XqyDXiv52Xe+GyY6GVtVUMd/KSHSInF2xgPUdInWdgqnFHC5bwNF2r8
yxHuNzUzEuu9xVzaqi7Wxk8t+uiktS4GgtcK94Zk8EkAxfnQe5PGa2ijcOF90whX
OCmhcT1CBXu4jgO3kfnuJ8E3A3gaN5I+VnqvvnPxpbg3GOmMhxr3pruTuCObGqFM
CUDBZqNBD5wezjJImdnvS50LGx1CKgelrxP2NAmzb1gMAmS7XZfKxa51Tszaeqna
wuC4
-----END CERTIFICATE-----
```

It should also be noted that all the passwords / passphrase and certificates will be masked after the creation of the CR. This includes

- Also, inital pgAdminPassword: admin-password
- pgpassword: mydbpassword
- pgreplpassword: repluserpwd
- tdepassphrase: tde-passphrase
- certificate.key
- certificate.crt
- certificate.cacrt

Values of child CRs at the time of initial deployment of cluster, are stored in FEPCluster under fepChildCrVals, e.g. for Server certificates, Configuration of FEP, User details.

All fields for FEPCluster CR and its child CRs should be managed through FEPCluster CR only. Operator will reflect the changes to respective child CR to be processed. The fields that not allowed to change will not be reflected from parent to child CR and hence will not have any affect.

1.2.2 FEP Cluster Configuration

Configuration of all aspects of FEP Cluster is done through FEPCluster CR only.

All fields for FEPCluster CR and its child CRs should be managed through FEPCluster CR only. Operator will reflect the changes to respective child CR to be processed. The fields that not allowed to change will not be reflected from parent to child CR and hence will not have any affect. Refer to "1.1 FEPCluster Parameter" for details.

All child CRs are marked as internal objects in RedHat OCP and will not appear on console. However, it can be checked on command line using oc or kubectl commands.

Following table shows Child CRs of FEPCluster CR and respective sections in parent CR related to given child CR.

Configuration changes are made in these sections will update allowable fields only in corresponding child CR.

Child CR Name	Relevant sections in FEP Cluster CR
FEPBackup	spec.fepChildCrVal.backup
FEPCert	spec.fepChildCrVal.systemCertificates
FEPConfig	spec.fepChildCrVal.customPgAudit spec.fepChildCrVal.customPgHba spec.fepChildCrVal.customPgParams
FEPUser	spec.fepChildCrVal.sysUsers
FEPVolume	spec.fepChildCrVal.storage

1.2.3 FEPConfig Child Custom Resource Parameters

Name	Details
CRD Name	FEPConfig
Definition	apiVersion: apiextensions.k8s.io/v1beta1 kind: CustomResourceDefinition metadata: name: fepconfigs.fep.fujitsu.io spec: group: fep.fujitsu.io names: kind: FEPConfig listKind: FEPConfigList plural: fepconfigs singular: fepconfig cope: Namespaced conversion: strategy: None subresources: status: {} versions: - name: v1 served: true storage: true
Operations	Create: will be done by FEPCluster CR Edit: kubectl edit fepconfig Update: kubectl patch fepconfig List: kubectl get fepcluster

Field	Default	Details
metadata.name	<same-as-in-FEPCluster>	This value is inherited from parent FEPCluster CR
metadata.namespace	<same-as-in-FEPCluster>	This value is inherited from parent FEPCluster CR
spec.customPgAudit	All line specified in spec.fepChildCrVal.customPgAudit of FEPCluster CR	Audit rules can be updated in this section. Requires restart. Note: initial values inherited once only at start. Changes to FEPConfig directly
spec.customPgHba	All line specified in spec.fepChildCrVal.customPgHba of FEPCluster CR	pg_hba rules can be added in this section Note: Inherited once at start. Changes to FEPConfig directly
spec.customPgParams	All line specified in	All postgres parameters are listed here to overwrite defaults.

Field	Default	Details
	spec.fepChildCrVal.customPgParams of FEPCluster CR	Note: Inherited once at start. Changes to FEPConfig directly

Example of FEPConfig CR created

```
apiVersion: fep.fujitsu.io/v1
kind: FEPConfig
metadata:
  name: new-fep-19ncfg
  namespace: cfg-expt
spec:
  sysExtraLogging: false
  customPgAudit: |
    # define pg audit custom params here to override defaults.
    # if log volume is not defined, log_directory should be
    # changed to '/database/userdata/data/log'
    [output]
    logger = 'auditlog'
    log_directory = '/database/log/audit'
    log_truncate_on_rotation = on
    log_filename = 'pgaudit-%a.log'
    log_rotation_age = 1d
    log_rotation_size = 0
    [rule]
  customPgHba: |
    # define pg_hba custom rules here to be merged with default rules.
    # TYPE     DATABASE        USER        ADDRESS        METHOD
  customPgParams: |+
    # define custom postgresql.conf parameters below to override defaults.
    # Current values are as per default FEP deployment
    shared_preload_libraries='pgx_datamasking,vci,pgaudit,pg_prewarm'
    session_preload_libraries='vci,pg_prewarm'
    max_prepared_transactions = 100
    max_worker_processes = 20
    max_connections = 100
    work_mem = 1MB
    maintenance_work_mem = 20MB
    shared_buffers = 128MB
    effective_cache_size = 384MB
    checkpoint_completion_target = 0.8
    pgx_global_metacache = 10MB
    temp_buffers = 10MB

    # tcp parameters
    tcp_keepalives_idle = 30
    tcp_keepalives_interval = 10
    tcp_keepalives_count = 3

    # logging parameters in default fep installation
    # if log volume is not defined, log_directory should be
    # changed to '/database/userdata/data/log'    log_directory = '/database/log'
    log_filename = 'logfile-%a.log'
    log_file_mode = 0600
    log_truncate_on_rotation = on
    log_rotation_age = 1d
    log_rotation_size = 0
    log_checkpoints = on
    log_line_prefix = '%e %t [%p]: [%l-1] user=%u,db=%d,app=%a,client=%h'
```

```
log_lock_waits = on
log_autovacuum_min_duration = 60s
logging_collector = on
pgaudit.config_file= '/opt/app-root/src/pgaudit-cfg/pgaudit.conf'
log_replication_commands = on
log_min_messages = WARNING
log_destination = stderr

# vci parameters in default fep installation
vci.enable = on
vci.maintenance_work_mem = 256MB
vci.max_local_ros = 64MB
vci.force_max_parallelism = off

# wal_archive parameters in default fep installation
archive_mode = on
wal_level = replica
max_wal_senders = 10
wal_keep_segments = 64
wal_sender_timeout = 60s
```

1.2.4 FEPUser Child Custom Resource Parameters

Name	Details
CRD Name	FEPUser
Definition	apiVersion: apiextensions.k8s.io/v1beta1 kind: CustomResourceDefinition metadata: name: fepusers.fep.fujitsu.io spec: group: fep.fujitsu.io names: kind: FEPUser listKind: FEPUserList plural: fepusers singular: fepuser scope: Namespaced conversion: strategy: None subresources: status: {} versions: - name: v1 served: true storage: true
Operations	Create: will be done by FEPCluster CR with data in fepChildCrVal Edit: kubectl edit fepuser Update: kubectl patch fepuser

Name	Details
	List: kubectl get fepuser

Field	Default	Details
metadata.name	<same-as-in-FEPCluster>	This value is inherited from parent FEPCluster CR
metadata.namespace	<same-as-in-FEPCluster>	This value is inherited from parent FEPCluster CR
spec.pgAdminPassword	spec.fepChildCrVal.users.pgAdminPassword of FEPCluster CR	postgres superuser password. Masked once secret is created/changed Note: initial values inherited once only at start. Changes to FEPUser directly
spec.pgdb	spec.fepChildCrVal.users.pgdb of FEPCluster CR	Name of a user database Note: Created once only at start. Cannot be changed
spec.pgpassword	spec.fepChildCrVal.users.pgpassword of FEPCluster CR	Password for superuser for user database pgdb. Masked once secret is created/changed Note: initial values inherited once only at start. Changes to FEPUser directly
spec.pguser	spec.fepChildCrVal.users.pguser of FEPCluster CR	Name of a user database Note: Created once only at start. Cannot be changed
spec.pgrepluser	spec.fepChildCrVal.users.pgrepluser of FEPCluster CR	Name of a database user for replication
spec.pgreplpassword	spec.fepChildCrVal.users.pgreplpassword of FEPCluster CR	Password for pgrepluser
spec.tdepassphrase	spec.fepChildCrVal.users.tdepassphrase of FEPCluster CR	Passphrase for encrypting/decrypting keystore file which contains the TDE encryption key
spec.pgRewindUser	rewind_user	Database user for Rewind
spec.pgRewindUserPassword	rewind_password	Password for database user rewinduser
spec.pgAdminTls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for “postgres” user. For MTLS Postgres communication, this key must be defined. The private key cannot be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.pgAdminTls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.pgAdminTls.sslMode	verify-full	For MTLS, this value must be set to verify-full. If only TLS is required, this can be set to verify-ca or prefer.
spec.pgrepluserTls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for “repluser” user. For MTLS Postgres communication, this key must be defined. The private key cannot be password protected. When using

Field	Default	Details
		cert-manager, the secret also contains the CA bundle in ca.crt.
spec.pgrepluserTls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.pgrepluserTls.sslMode	verify-full	For MTLS, this value must be set to verify-full. If only TLS is required, this can be set to verify-ca or prefer.
spec.pgRewindUserTls.certificateName	<secret-name>	Name of Kubernetes secret that contains the certificate in tls.crt and private key in tls.key for “rewinduser” user. For MTLS Postgres communication, this key must be defined. The private key cannot be password protected. When using cert-manager, the secret also contains the CA bundle in ca.crt.
spec.pgRewindUserTls.caName	<configmap-name>	Name of Kubernetes configmap that contains the CA bundle. If using cert-manager, the ca.crt is already included in the secret above. In this situation, this key can be omitted.
spec.pgRewindUserTls.sslMode	verify-full	For MTLS, this value must be set to verify-full. If only TLS is required, this can be set to verify-ca or prefer.

Example of FEPUser CR created

```
apiVersion: fep.fujitsu.io/v1
kind: FEPUser
metadata:
  name: new-fep-19n
  namespace: testswatiproject
spec:
  pgAdminPassword: '***************'
  pgdb: mydb
  pgpassword: '***************'
  pgreplpassword: '***************'
  pgrepluser: repluser
  pguser: mydbuser
  tdepassphrase: '***************'
  sysExtraLogging: false
  pgRewindUser: rewind_user
  pgRewindUserPassword: rewind_password
  pgAdminTls:
    certificateName: admin-client-certs-secret
    caName: admin-ssl-rootcert-configmap
    sslMode: prefer
  pgrepluserTls:
    certificateName: repluser-client-certs-secret
    caName: repluser-ca-name-configmap
    sslMode: prefer
  pgRewindUserTls:
    certificateName: rewinduser-client-certs-secret
    caName: rewinduser-ca-name-configmap
    sslMode: prefer
```

Note

- Password and Passphrase are masked in output from CR. The original values can still be found in the respective Kubernetes secrets and configmaps.

- TDE is enabled by default with given tdepassphrase and must have a value.
- TDE is enabled by using the key tdepassphrase with the desired passphrase.Do not remove this key once TDE is enabled. Otherwise, the database may go into a crash loop. If the Cluster is running on Async Replication and a failover/switchover occured during the crash loop, there could be data lost. The team is looking at preventing the deletion of this passphrase from Operator even if customer tries to remove it in customer resource.
- Database users and their passwords managed by the FEPUser CR should not be changed in the SQL interface. Inconsistencies with the information managed by the operator can cause problems with operator operation. If you make changes in the SQL interface, use the SQL interface again to restore the original state.

1.2.5 FEPVolume Child Custom Resource Parameters

FEPVolume CR created

```
spec:
  archivewalVol:
    size: 1Gi
  backupVol:
    size: 2Gi
  dataVol:
    size: 2Gi
  logVol:
    size: 1Gi
  tablespaceVol:
    size: 512Mi
  walVol:
    size: 1Gi
  selectedVolList:
  - name: data
  - name: tablespace
  - name: wal
  - name: log
  sysExtraLogging: false
```

1.2.6 FEPCert Child Custom Resource Parameters

Below is an example CR for certificates to be used by FEP server container

```
apiVersion: fep.fujitsu.io/v1
kind: FEPCert
metadata:
  name: new-fep
  namespace: ansible-operator-poc
spec:
  key: |-
    -----BEGIN RSA PRIVATE KEY-----
    MIIEowIBAAKCAQEA4AI33yvHZws+jta6qpV6wzJqF8odIfTIpCfbrVcUUtLFKJ1I
    2e4SceTKi6O3C/I1XuvWlpng5IO65+fQQLO06z1/AuQT78YUn/Wlm9x1aHVsv4AN
    B5JWWqDOjrRT3o7nRPGXfilabP0rGE2mJJcVR9nExJ3IeaktgT3sb8YlXvtchyYp
    mjdbfxabTz07ig0+6/cwKoRRxOK8Uf7f5euE0cI/490J6r5Rs4lgD8sIQNCUFlTF
    YvmAH7gcdssSFBt8NPlUATHEsoFmlW0DKCJWNhTLOht+s6L/1zwTHLjPG2pdkG6W
    dgmu5H2pDml8CDNLDv98Aj7i+I5SRKKcVPlnuQIDAQABAoIBAFPQYKlOzw/+BA0b
    yMIUpdctIMb/54CR/xR0mVw1DbSjigNVPjHUQvB8Y1B2FAITQObgJO06bAv0QdWN
    Rb0/v/yYiNJDFjaLjaIAHlO/2+oWrXbFaZqgpVDJhB+e1xaZr2x7XGxm+p925k30
    l6pvIRY+I8JRKvZiV1VZHwL/R3JOtPr++xMZtLVjVOI+f+ySqJ+TZHuAjm49EKxj
    cEmmJ28b7QcziXsvKy00f+zbqLIBKXQdZAFU5eEr1BsDRXdRW+Kf0XIvftuy4BJZ
    voKT+VGhEvF/qysswL4+6IAO6tpuYnnM0Y2d3sOGoWPkTcQK0MekYKzL/WmtCjNs
    9hodJtECgYEA5EWyhEOf4uOKe5TDp697UCUvXLoOR58FDe/S8XNvScn29jjOkqIg
    OMoqo9xAkJTNTzqn5UUdt1x/pgM2NxlPLFijrc0zQlX3SoOO2ryDd9WNi7YKtN16
    KJqa536WeZu2OEbuAZ+S3GALVy1RPeTNPnUOmKnF06DjDUGzLNCZy10CgYEA+zfw
    952DWuz1U0Z4wvAEqqcgUKXPKrkTXV/iUnjkDkrLYVr0ZofDNTXrdHl+UedFmaOC
```

```
    cieZn6DNhcdz5tKtyysGMH3g/qs9PfoGUngvcXsy0Egk04l3x1jc8TTCLqXZXYaQ
    HMsx51n+R58oncPtzYSUOr9qQ6PbC2CstTbFJA0CgYEAjGEsUliAB/jknfEzjXjG
    PdhQUxb8VyE864Az2lah9t/kJzFyIAziAeqZ5GE7t247AGFTBRTHHI8e1Qoemi3P
    Wbc9GVIbFs1lIYbcIDpUIyrKPEP8O5QEXtoNLxXTFgAjRGKiVY87spjCAJ+W2ZhO
    e/1it5GYXfgQCYQA2yuBmOUCgYANRkR2YR1axaCk+NlSu6oTdmdPu6M5x7PNQE7O
    OtMaKjua9lppvIzFGAdMDUtueoEEAE7ZR1xnwfB6PDLUpJdIYAqgr1YfPt8qkjaZ
    Tv56yZ7CwL0pbF8m6nwqRrZoDp1wwraEvvvxFKFKGY/k3kCHlpTakdjEoDjn3gDi
    RnWeVQKBgCEneMSzucei5LRppRtRaJw/Btll8qlPMlX3W7dxQ3cLwpmLOn0m51Fp
    PIZ44zYK8R6fu4+/sSrlfaIg86Ugeufp6YNxyNROKxUGza5vDIu5OftwWtBeg+UK
    Z8lLWNdX6pp7WMujmF3H1DrkBbauYMUKZ4UxUYtelgHERMePIxwb
    -----END RSA PRIVATE KEY-----
crt: |-
    -----BEGIN CERTIFICATE-----
    MIIDUTCCAjmgAwIBAgIRAMocW3qMoHrD6qRvMPppMkMwDQYJKoZIhvcNAQELBQAw
    NzEQMA4GA1UECgwHRnVqaXRzdTEjMCEGA1UEAwwaRkVQIFJvb3QgQ0EgZm9yIEt1
    YmVybmV0ZXMwHhcNMjEwMjA2MDQzMjM2WhcNMjYwMjA1MDQzMjM2WjA/MRAwDgYD
    VQQKEwdGdWppdHN1MSswKQYDVQQDEyJGVUpJVFNVIEVudGVycHJpc2UgUG9zdGdy
    ZXMgU2VydmVyMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA4AI33yvH
    Zws+jta6qpV6wzJqF8odIfTIpCfbrVcUUtLFKJ1I2e4SceTKi6O3C/I1XuvWlpng
    5IO65+fQQLO06z1/AuQT78YUn/Wlm9x1aHVsv4ANB5JWWqDOjrRT3o7nRPGXfila
    bP0rGE2mJJcVR9nExJ3IeaktgT3sb8YlXvtchyYpmjdbfxabTz07ig0+6/cwKoRR
    xOK8Uf7f5euE0cI/490J6r5Rs4lgD8sIQNCUFlTFYvmAH7gcdssSFBt8NPlUATHE
    soFmlW0DKCJWNhTLOht+s6L/1zwTHLjPG2pdkG6Wdgmu5H2pDml8CDNLDv98Aj7i
    +I5SRKKcVPlnuQIDAQABo1AwTjAdBgNVHSUEFjAUBggrBgEFBQcDAQYIKwYBBQUH
    AwIwDAYDVR0TAQH/BAIwADAfBgNVHSMEGDAWgBQcwrrUO0u+FhIUuVdrDRCQRsi6
    ZjANBgkqhkiG9w0BAQsFAAOCAQEAm5dxBoI9pScOCvRAchg4CprdRDSJb9K6yB3O
    nCAxnM47iHeXnY3WlnI388kHu8DU7O4ba1tJbGs3KY9KzioPk43pU12jWkO1onoF
    +mTDjx/Ef1cYWA9r5q/LtgTa6Q2sxV4O2x67QW82aAnaxO34dV5zWCPIvAoovZBV
    HRT+BgCg3r2vD1RGKK2nl1aYJtWhO1SZubam+VttdZ/vbM9oOJctxmImsEtBXjkY
    KteePdQtLL5o03JhyXWyRshCq+HMmKf2KgyY8gvydGcP4eLQdBWcW40LcnVq6UjT
    0kJycJEKngMVademq1ZWHGaiYB7hyT6GhgIcHUJ2cKrPgbEh1Q==
    -----END CERTIFICATE-----
cacrt: |-
    -----BEGIN CERTIFICATE-----
    MIIDTzCCAjegAwIBAgIUYssQ8I74US5g+1+Z7CHuaDgkZnEwDQYJKoZIhvcNAQEL
    BQAwNzEQMA4GA1UECgwHRnVqaXRzdTEjMCEGA1UEAwwaRkVQIFJvb3QgQ0EgZm9y
    IEt1YmVybmV0ZXMwHhcNMjEwMjA2MDM1MjI4WhcNMzEwMjA0MDM1MjI4WjA3MRAw
    DgYDVQQKDAdGdWppdHN1MSMwIQYDVQQDDBpGRVAgUm9vdCBDQSBmb3IgS3ViZXJu
    ZXRlczCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAMs97gUF0xkUzCgL
    7MiiDju9ySr/ziwjvcYU7jA9ML+SLmftMs3HtcYbAmSntqI+MDBSR/FAJTOoytuT
    pV+mCFcGj2YAjDpliHPeNcUpbryy4YMChF3+MovkIwGCksxo5rhiWhGmoBYpA48P
    4Xe8SPlzqMzhFvNeKzyiUhvjutS2Y1Ss38lsTaurFPx64vQ2PaC54XzdwMptXtpb
    tYmWSzCpJWwxZ6lF3vitdA2w0tnBWNyctAd0+RIM/fvArxiIqseAux9t0uogm5to
    lRIhvekuxOpXBPEqtIYQ4j9XUW2JH8vUDnzPkPvjrq+A3Ug8OyyfGVrW7+VYXozu
    c4aP7P0CAwEAAaNTMFEwHQYDVR0OBBYEFBzCutQ7S74WEhS5V2sNEJBGyLpmMB8G
    A1UdIwQYMBaAFBzCutQ7S74WEhS5V2sNEJBGyLpmMA8GA1UdEwEB/wQFMAMBAf8w
    DQYJKoZIhvcNAQELBQADggEBAMDwD85RAaWEBptFgLzKw+9xEUy1vcZaonAuA1qc
    T342XTueyAugxkC11HwdCGgGS34VyctfMGqj4AW6pA2ez4tLrbOps4DmV4sw8uBL
    8pgRDgfly3ob9FEg2wa0hmrwX9jH5Bt4vySUE2785uPAqaspT2UNtTBxS85BUi1T
    sKId2Rtil6an281Z81wyWVI6Jm2D4MG0mbsiGcTPlCtdg/UljvDYymXlAvd4vNhl
    k9hDa13TgDqJKgKdTIcmZoNQdpEVgFcO0h9AEUy5AuLqxHq60dLfZ6ESGPlMI7Lm
    i4PzYbCnBmOe+7TnHcPSyrnehs66Ik+oifRd82eYS7vKjFw=
    -----END CERTIFICATE-----
```

Note

This approach of specifying FEPCerts is getting deprecated. Should follow Secrets as referred in section to configure Certs for Server, Patroni and Users.

1.2.7 FEPBackup Child Custom Resource Parameters

Name	Details
CRD Name	FEPBackup
Definition	apiVersion: apiextensions.k8s.io/v1beta1 kind: CustomResourceDefinition metadata: name: fepbackup.fep.fujitsu.io spec: group: fep.fujitsu.io names: kind: FEPBackup listKind: FEPBackupList plural: fepbackups singular: fepbackup cope: Namespaced conversion: strategy: None subresources: status: {} versions: - name: v1 served: true storage: true
Operations	Create: will be done by FEPCluster CR Edit: kubectl edit fepbackup Update: kubectl patch fepbackup List: kubectl get fepbackup

Field	Default	Details
apiVersion	fep.fujitsu.io/v1	Fixed
kind	FEPBackup	Fixed
metadata.name	<clustername>	Enter the CR name.
spec.pgbackrestParams	"|"	"|" It is fixed, and the parameter set in pgbackrest.conf is described from the line below.
spec.schedule.num	Integer	Number of schedules to set The maximum number of backup schedules is 5.
spec.scheduleN.schedule	-	Write the date and time of the Nth schedule in cron format. The date and time is UTC time.
spec.scheduleN.type	full/incr	full: Perform a full backup (Back up the contents of the database cluster).

Field	Default	Details
		incr ー Perform an incremental backup (Back up only the database cluster files that were changed to the last backup migration).
spec.preScript	" "	This parameter must specify a default value.
spec.postScript	" "	This parameter must specify a default value.

Example of FEPBackup CR created

```
apiVersion: fep.fujitsu.io/v1
kind: FEPBackup
metadata:
  name: fepcluster-backup
spec:
  schedule:
    num : 2
  schedule1:
    schedule : "0 0 1 * *"
    type : "full"
  schedule2:
    schedule : "0 0 1-6 * *"
    type : "incr"
  preScript: " "
  postScript: " "
  pgbackrestParams: |
    # define custom pgbackrest.conf parameters below to override defaults.
    [global]
    repo1-retention-full = 30
    repo1-retention-full-type = time
...
```

1.2.8 FEPRestore Custom Resource Parameters

Name	Details
CRD Name	FEPRestore
Definition	apiVersion: apiextensions.k8s.io/v1beta1 kind: CustomResourceDefinition metadata: name: feprestore.fep.fujitsu.io spec: group: fep.fujitsu.io names: kind: FEPRestore listKind: FEPRestoreList plural: feprestores singular: feprestore cope: Namespaced conversion: strategy: None

Name	Details
	subresources: status: {} versions: - name: v1 served: true storage: true
Operations	Create: kubectl create feprestore Edit: kubectl edit feprestore Update: kubectl patch feprestore List: kubectl get feprestore

Field	Default	Details
apiVersion	fep.fujitsu.io/v1	Fixed
kind	FEPRestore	Fixed
metadata.name	-	Enter the CR name.
spec.image	<current-released-image>	FEP restore container image to be used quay.io/fujitsu/fujitsu-enterprise-postgres-12-restore:ubi8-12-1.1 It is optional. Image is left blank by default. In such a case, it will pick up URL of image from operator container environment. If you specify the image, Operator will take that image to deploy container
spec.imagePullPolicy	IfNotPresent	
spec.mcSpec.limits	cpu: 0.2 memory: "300Mi"	
spec.mcSpec.requests	cpu: 0.1 memory: "200Mi"	
spec.fromFEPcluster	<from_clustername>	The name of the FEPcluster from which to restore
spec.toFEPcluster	<to_clustername>	Name of the FEPcluster to restore to The exact restore destination volume is retrieved from FEPcluster
spec.restoretype	latest/PITR	latest - Restore Latest State PITR - Date-Time Restore
spec.restoredate	-	If spec.restoretype is PITR, specify the day of PITR (UTC) in YYYY-MM-DD format Example) 2020-11-25
spec.restoretime	-	If spec.restoretype is PITR, specifies the PITR time (UTC) in HH: MM: SS format Example) 02:50:43

Example of FEPRestore CR created

```
apiVersion: fep.fujitsu.io/v1
kind: FEPRestore
metadata:
  name: feprestore
spec:
  mcSpec:
    limits:
      cpu: 200m
      memory: 300Mi
    requests:
      cpu: 100m
      memory: 200Mi
  fromFEPcluster: fepcluster1
  toFEPcluster: fepcluster2
  restoretype: latest
  imagePullPolicy: IfNotPresent
```

Example of Point-In-Time-Recovery using FEPRestore CR

```
apiVersion: fep.fujitsu.io/v1
kind: FEPRestore
metadata:
  name: feprestore
spec:
  mcSpec:
    limits:
      cpu: 300m
      memory: 700Mi
    requests:
      cpu: 200m
      memory: 512Mi
  fromFEPcluster: fepclusterA
  toFEPcluster: fepclusterB
  restoretype: PITR
  restoredate: 2020-11-25
  restoretime: 02:50:43
  imagePullPolicy: IfNotPresent
```

1.2.9 FEPPgpool2 Custom Resource Parameters

Equivalent Kubernetes command: kubectl create FEPPgpool2

This operation will create a PGPool2 with supplied information.

Field	Default	Details
apiVersion	fep.fujitsu.io/v1	Fixed
kind	FEPPgpool2	Fixed
metadata.name	-	List the name of the FEP Pgpool2 container.
metadata.namespace	-	Specify the namespace of the environment where you want to deploy the operator.
spec.image	<current-released-image>	FEPPgpool2 container image to be used quay.io/fujitsu/fujitsu-enterprise-postgres-12-pgpool2:ubi8-12-1.1 It is optional.

Field	Default	Details
		Image is left blank by default. In such a case, it will pick up URL of image from operator container environment. If you specify the image, Operator will take that image to deploy container.
spec.count	2	List the number of FEP Pgpool2 containers to create.
spec.serviceport	9999	Describes the TCP port for connecting to the FEP Pgpool2 container.
spec.statusport	9898	Identifies the TCP port for connecting to the PCP process.
spec.limits.cpu	400m	List the number of CPUs (restriction) to allocate to resources.limits.cpu.
spec.limits.memory	512Mi	Specifies the memory size (restriction) to allocate to resources.limits.memory.
spec.requests.cpu	200m	List the number of CPUs (request) to allocate to resources.requests.cpu.
spec.requests.memory	256Mi	Specifies the memory size (request) to allocate to resources.requests.memory
spec.fepclustername	new-fep	Enter the FEPCluster name to connect to.
spec.customhba		If you want to use pool _ hba.conf, describe what pool _ hba.conf should contain from the line below.
spec.customparams	listen_addresses = '*' pcp_listen_addresses = '*' num_init_children = 32 reserved_connections = 0 enable_pool_hba = off allow_clear_text_frontend_auth = off authentication_timeout = 80 backend_weight0 = 1 backend_weight1 = 1 backend_flag0 = 'ALWAYS_MASTER' backend_flag1 = 'ALLOW_TO_FAILOVER' connection_cache = on max_pool = 4 listen_backlog_multiplier = 2 serialize_accept = off child_life_time = 300 client_idle_limit = 0 child_max_connections = 0 connection_life_time = 0 reset_query_list = 'ABORT; DISCARD ALL'	"|" and the Pgpool-II parameters. Refer to "Pgpool-II parameters" for detail.

Field	Default	Details
	client_min_messages = info log_min_messages = debug1 log_statement = on log_per_node_statement = on log_client_messages = on log_hostname = on log_connections = on log_line_prefix = '%t: pid %p: ' load_balance_mode = on ignore_leading_white_space = on white_function_list = '' black_function_list = 'currval,lastval,nextval,setval' black_query_pattern_list = '' database_redirect_preference_list = '' app_name_redirect_preference_list = '' allow_sql_comments = off disable_load_balance_on_write = 'transaction' statement_level_load_balance = on connect_timeout = 10000 sr_check_period = 0 sr_check_user = 'postgres' delay_threshold = 0 log_standby_delay = 'none' ssl = on ssl_ciphers = 'HIGH:MEDIUM:+3DES:!aNULL' ssl_prefer_server_ciphers = off ssl_ecdh_curve = 'prime256v1' ssl_dh_params_file = '' relcache_expire = 0 relcache_size = 256 check_temp_table = catalog check_unlogged_table = on enable_shared_relcache = on	

Field	Default	Details
	relcache_query_target = master	
spec.custompcp	"|"	If you use the pcp command, "|" and the contents of pcp.conf from the line below.
spec.customsslkey	"|"	If you want to do it, "|" and the Beethoven key content in the line below.
spec.customsslcert	"|"	If you want to do it, "|" and the contents of the public x 509 certificate from the line below.
spec.customsslcacert	"|"	If you want to do it, "|" and the following lines describe the contents of the CA root certificate in PEM format.
spec.customlogsize	100 Mi	Specifies the persistent volume size for log output.
spec.storageclassname		Specifies the storage class for log output.

Pgpool-II parameters

The parameters that can be specified are shown in the table below. For details on the parameters, refer to the Pgpool-II manual.

Category	Parameter name (Specified format)	Restart required after change
Connection settings	listen_addresses (string)	Y
	pcp_listen_addresses (string)	Y
	num_init_children (integer)	Y
	reserved_connections (integer)	Y
Authentication settings	enable_pool_hba (boolean)	
	allow_clear_text_frontend_auth (boolean)	
	authentication_timeout (integer)	
Backend settings	backend_weight0 (floating point)	
	backend_weight1 (floating point)	
	backend_flag0	
	backend_flag1	
Connection pooling	connection_cache (boolean)	Y
	max_pool (integer)	Y
	listen_backlog_multiplier (integer)	Y
	serialize_accept (boolean)	Y
	child_life_time (integer)	Y
	client_idle_limit (integer)	
	child_max_connections (integer)	Y
	connection_life_time (integer)	Y
	reset_query_list (string)	
Error reporting and log acquisition	client_min_messages (enum)	
	log_min_messages (enum)	
	log_statement (boolean)	
	log_per_node_statement (boolean)	

Category	Parameter name (Specified format)	Restart required after change
	log_client_messages (boolean)	
	log_hostname (boolean)	
	log_connections (boolean)	
	log_error_verbosity (enum)	
	log_line_prefix (string)	
Load sharing settings	load_balance_mode (boolean)	Y
	ignore_leading_white_space (boolean)	
	white_function_list (string)	
	black_function_list (string)	
	black_query_pattern_list (string)	
	database_redirect_preference_list (string)	
	app_name_redirect_preference_list (string)	
	allow_sql_comments (boolean)	
	disable_load_balance_on_write (string)	Y
	statement_level_load_balance (boolean)	
Health check	connect_timeout (integer)	
Streaming replication check	sr_check_period (integer)	
	sr_check_user (string)	
	sr_check_password (string)	
	sr_check_database (string)	
	delay_threshold (integer)	
	log_standby_delay (string)	
Secure Sockect Layer (SSL)	ssl (boolean)	Y
	ssl_ciphers (string)	Y
	ssl_prefer_server_ciphers (boolean)	Y
	ssl_ecdh_curve (string)	Y
	ssl_dh_params_file (string)	Y
Other parameters	relcache_expire (integer)	Y
	relcache_size (integer)	Y
	enable_shared_relcache (boolean)	Y
	relcache_query_target (enum)	
	check_temp_table (enum)	
	check_unlogged_table (boolean)	

1.2.10 FEPAction Custom Resource Parameters

Equivalent Kubernetes command: kubectl create -f <new_spec>

Specify parameters in the format described below.

Custom resource spec	Change effect
.spec.targetClusterName	Must specify target FEP Cluster name within namespace mentioned in metadata.
.spec.targetPgpool2Name	Must specify target FEPPgpool2 name within namespace mentioned in metadata when using pgpool2_restart.
.spec.fepAction.type	Must specify action type. Supported action types are: restart reload list switchover failover pgpool2_restart
.spec.fepAction.args	Must specify arguments needed for given action. For details of args corresponding to each action refer to "1.2.10.1 FEPAction Specific Operation Details".
.spec.sysExtraLogging	To turn extra debugging on, set value to true. It can be turned on/off at any time.

After execution of FEPAction CR, status is reflected in fepStatus field that is dynamically inserted in current FEPAction CR as needed.

fepStatus field used for FEPAction CR are described here

fepStatus (with possible values)	Remarks
fepActionStatus:	fepStatus is inserted at the top of FEPAction CR
fepActionCondition: Success | Failure	This flag is inserted in fepAction CR to reflect success or failure of requested action
fepActionResult: > “details”	The result contains verbose details corresponding to the specific action been executed. Should be noted that it is either plain text of HTTP output.
processedTimestamp: <time stamp>	Denotes time of action execution by the Operator

```
apiVersion: fep.fujitsu.io/v1
kind: FEPAction
fepActionStatus:
 fepActionCondition: Success
metadata:
 name: new-fep-reload-action
 namespace: myns
spec:
 fepAction:
  args:
  - new-fep-sts-0
  - new-fep-sts-1
  type: reload
 sysExtraLogging: false
 targetClusterName: new-fep
```

Note

Please do not use the FEPAction to perform a switchover or restart while executing backup. Failed to get the backup.

1.2.10.1 FEPAction Specific Operation Details

Action type - reload

The reload action will manually reload the FEP database on the targeted FEPCluster.

“reload” action type expects users to specify the name of individual FEP pods that they want to run the database reload operation on. They specify that in the args section under the FEPAction CR spec as below :

```
spec:
fepAction:
 args:
 - nf-131851-sts-0
 - nf-131851-sts-1
 type: reload
 targetClusterName: nf-131851
```

Action type - restart

The restart action will manually restart the FEP database on the targeted FEPCluster.

“restart” action type expects users to specify the name of individual FEP pods that they want to run the database restart operation on. They specify that in the args section under the FEPAction CR spec as below:

```
spec:
fepAction:
 args:
 - nf-131851-sts-0
 - nf-131851-sts-1
 type: restart
 targetClusterName: nf-131851
```

Action type - list

The list action will return the status of the targeted FEPCluster.

“list” action type expects users to specify just the target cluster name to list the details of the same. Looks like below:

```
spec:
fepAction:
 type: list
 targetClusterName: nf-131851
```

Action type - switchover

The switchover action performs a manually switchover of the current leader/primary database from one pod to another pod of the targeted FEPCluster.

“switchover” action type expects users to specify the name of the current leader/primary pod that they want to switchover from. They specify that in the args section under the FEPAction CR spec as below:

```
spec:
fepAction:
 args:
 - nf-131851-sts-2
 type: switchover
 targetClusterName: nf-131851
```

Here, nf-131851-sts-2 is the current primary.

Action type - failover

The failover action performs a manually failover of the current primary database from one pod to another pod of the targeted FEPCluster. The difference between switchover and failover is that, switchover expects the primary database is running at the time whereas failover can force switchover of primary role from a non-responding pod to another pod. Note that failover is a disruptive action and may cause data lost.

“failover” action type expects users to specify the names of the candidate pods that they want to failover to. They specify that in the args section under the FEPAction CR spec as below:

```
spec:
fepAction:
 args:
 - nf-131851-sts-1
 - nf-131851-sts-2
 type: failover
 targetClusterName: nf-131851
```

Here, nf-131851-sts-1 and nf-131851-2 are the candidate pods to failover to. In this example, the current primary pod would be nf-131851-sts-0.

Action type - pgpool2_restart

“pgpool2_restart” action type expects users to specify the name of individual FEPPgpool2 resource that they want to restart operation on. They specify that in the targetPgpool2Name section under the FEPAction CR spec as below:

```
spec:
fepAction:
 type: pgpool2_restart
 targetPgpool2Name: nf-131851-pgpool2
```

FUJITSU

FUJITSU Enterprise Postgres for Kubernetes

Quick Start Guide

J2UL-QUIK-05ENZ0(00)
September 2022

1. Prerequisites

- Registered OpenShift cluster with Red Hat Marketplace

  cf. https://marketplace.redhat.com/en-us/documentation/clusters#register-openshift-cluster-with-red-hat-marketplace

- Buy or try the product 'FUJITSU Enterprise Postgres for Kubernetes ' from Red Hat Marketplace

  cf. https://marketplace.redhat.com/en-us/documentation/operators

2. system requirements

2.1. CPU

It should be noted that it provides supports to both the following CPU Architectures to meet the scope of work.

No	CPU architecture
1	x86
2	s390x

2.2. Supported Platform

It supports running on the following platforms.

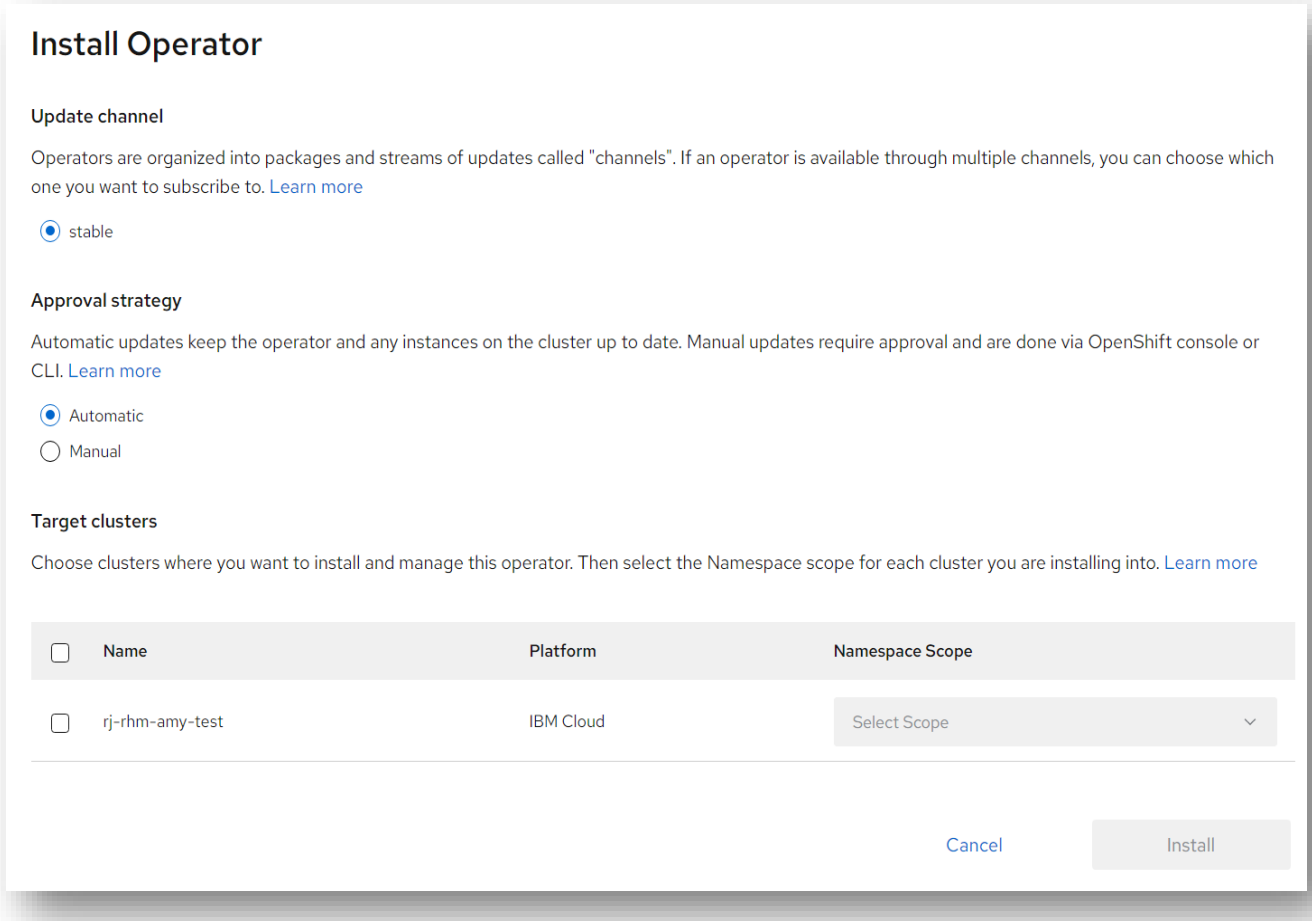
No	Platform	Version
1	OpenShift Container Platform	4.6, 4.8, 4.9, 4.10

3. Operator installation from Red Hat Marketplace

1. For information on registering your cluster and creating a namespace, see Red Hat Marketplace Docs. This must be done prior to operator install.
2. On the main menu, click **Workspace**, click **Software,** click on the **product box** of 'FUJITSU Enterprise Postgres for Kubernetes ', and then click Install Operator.
3. On the *Update Channel section*, select an option.
4. On the *Approval Strategy* section, select either *Automatic or Manual*. The approval strategy corresponds to how you want to process operator upgrades.
5. On the *Target Cluster* section:
   - Click the checkbox next to the clusters where you want to install the Operator.

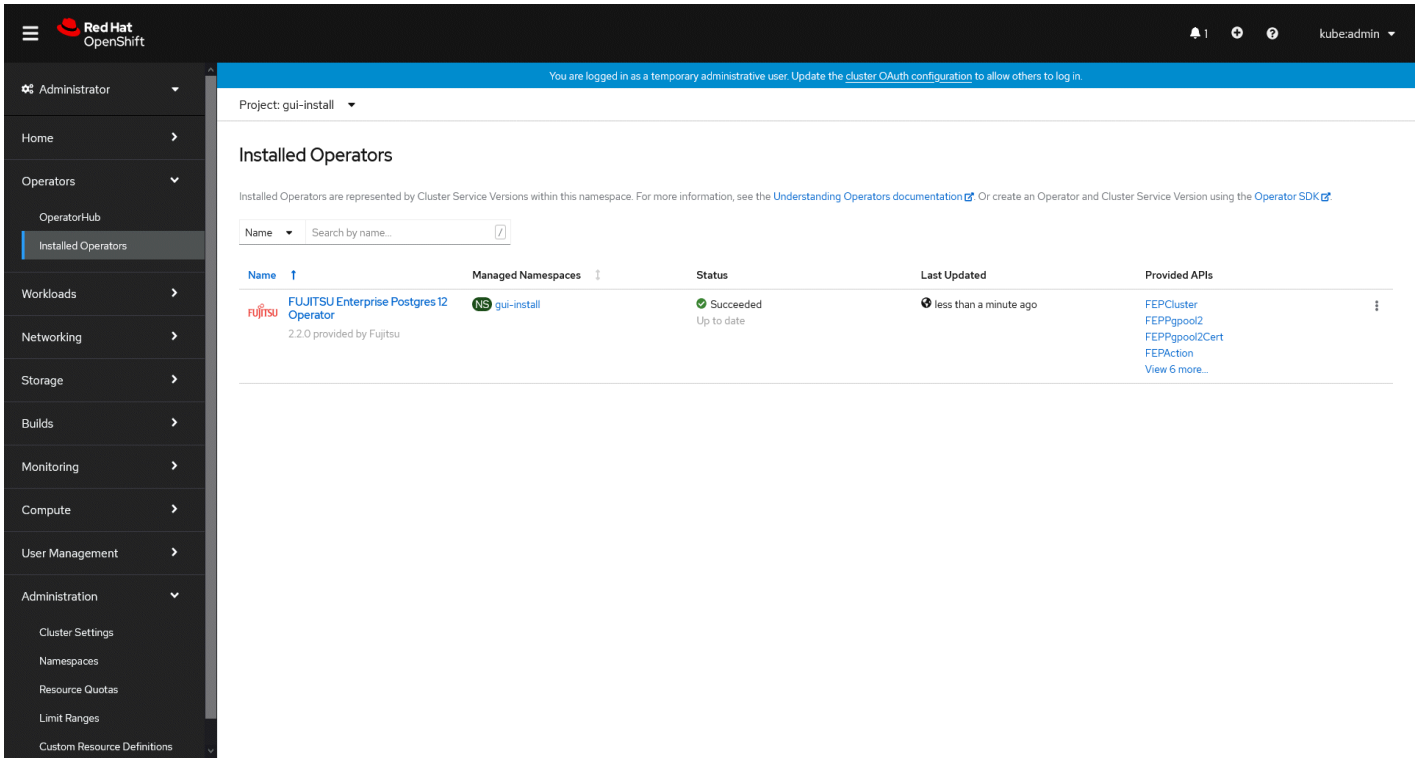
- For each cluster you selected, under **Namespace Scope**, on the **Select Scope** list, select an option.

6. Click **Install**. It may take several minutes for installation to complete.
7. Once installation is complete, the status will change from **installing** to **Up to date**.
8. For further information, see the [Red Hat Marketplace Operator documentation]



4. Verification of operator installation

1. Once status changes to Up to date, click the vertical ellipsis and select Cluster Console.
2. Open the cluster where you installed the product
3. Go to **Operators > Installed Operators**
4. Select the Namespace or Project you installed on
5. Verify status for product is **Succeeded**

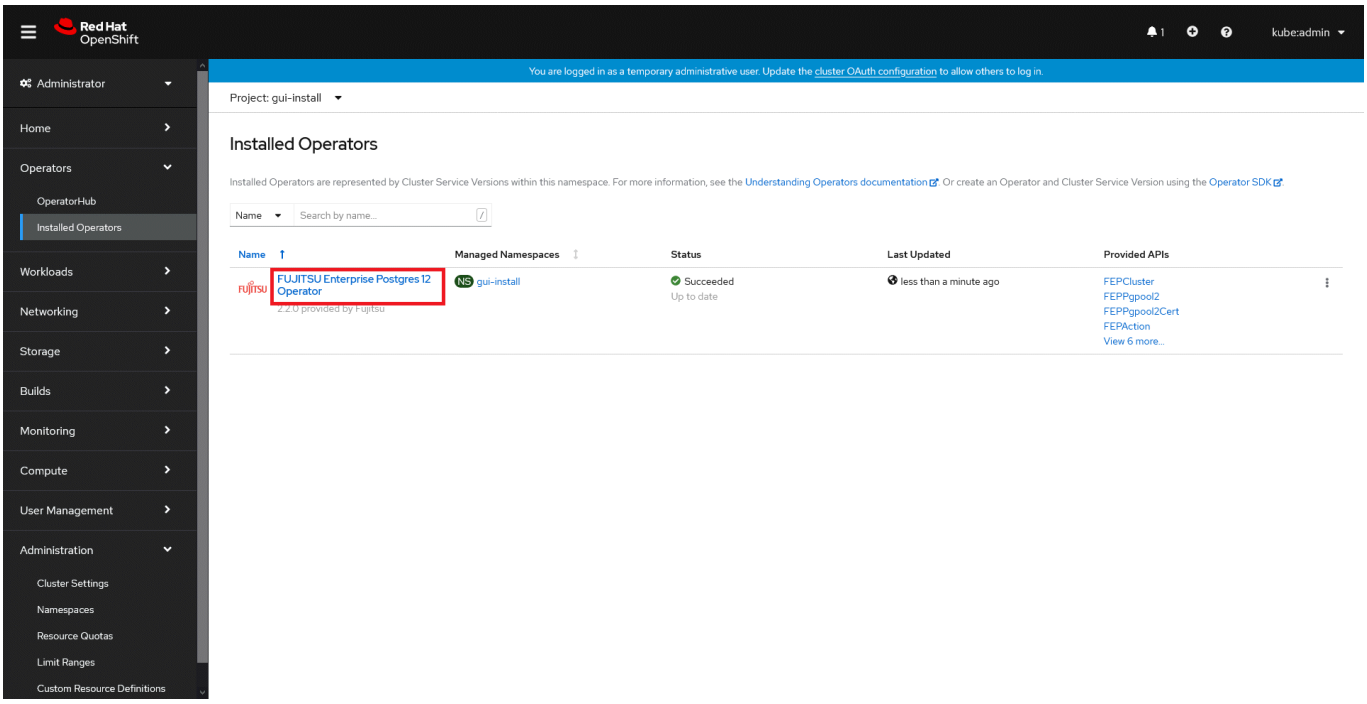


Installed operators status changes to “Succeeded”

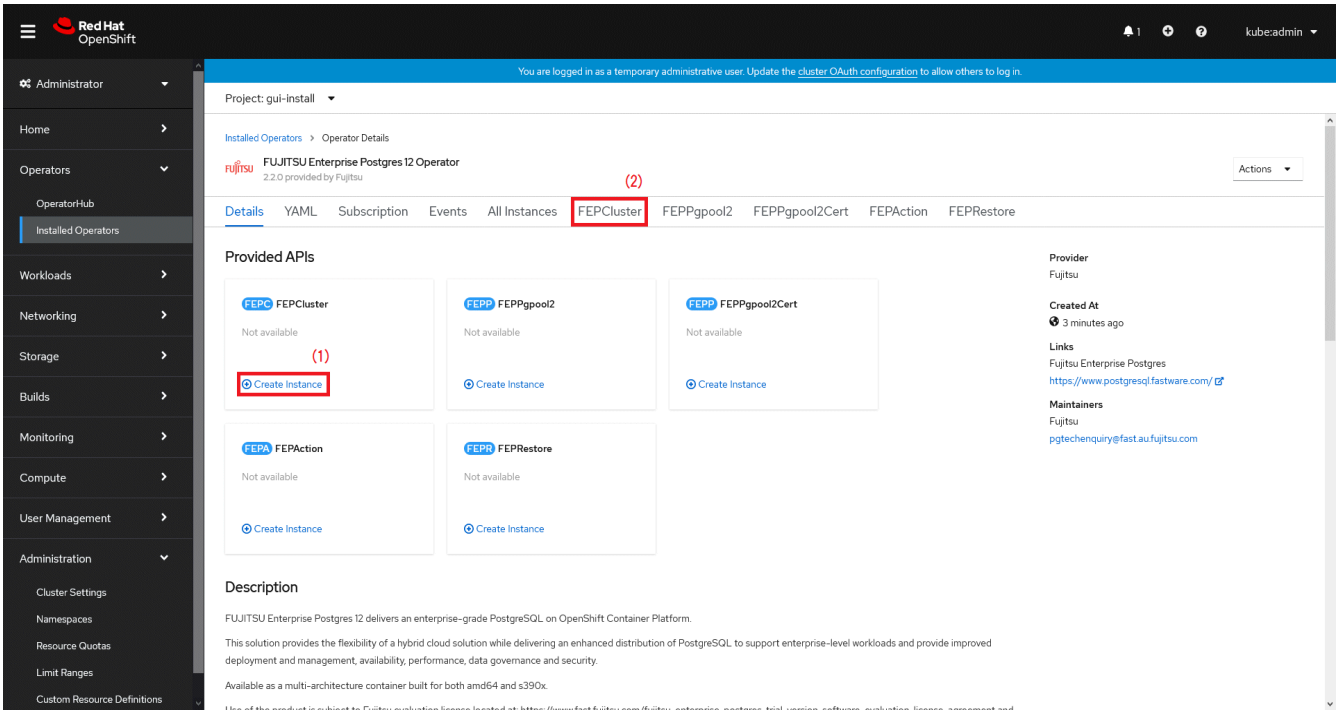
5. Deploying FEPCluster using Operator

To deploy a FEPCluster in given namespace, follow these steps:

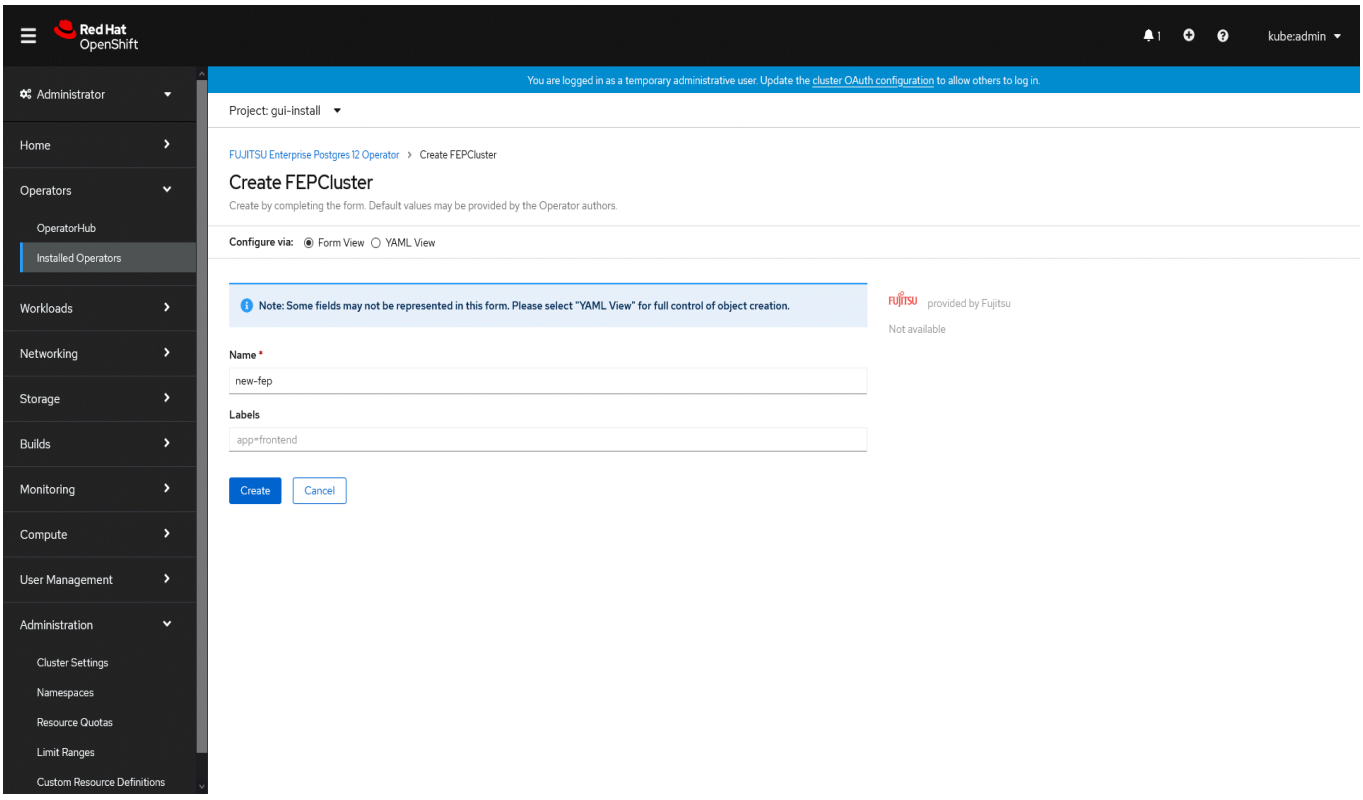
1. Under "Operators" menu item, click on "**Installed Operators**". You would see the installed FEP operator. Click on the name of operator.



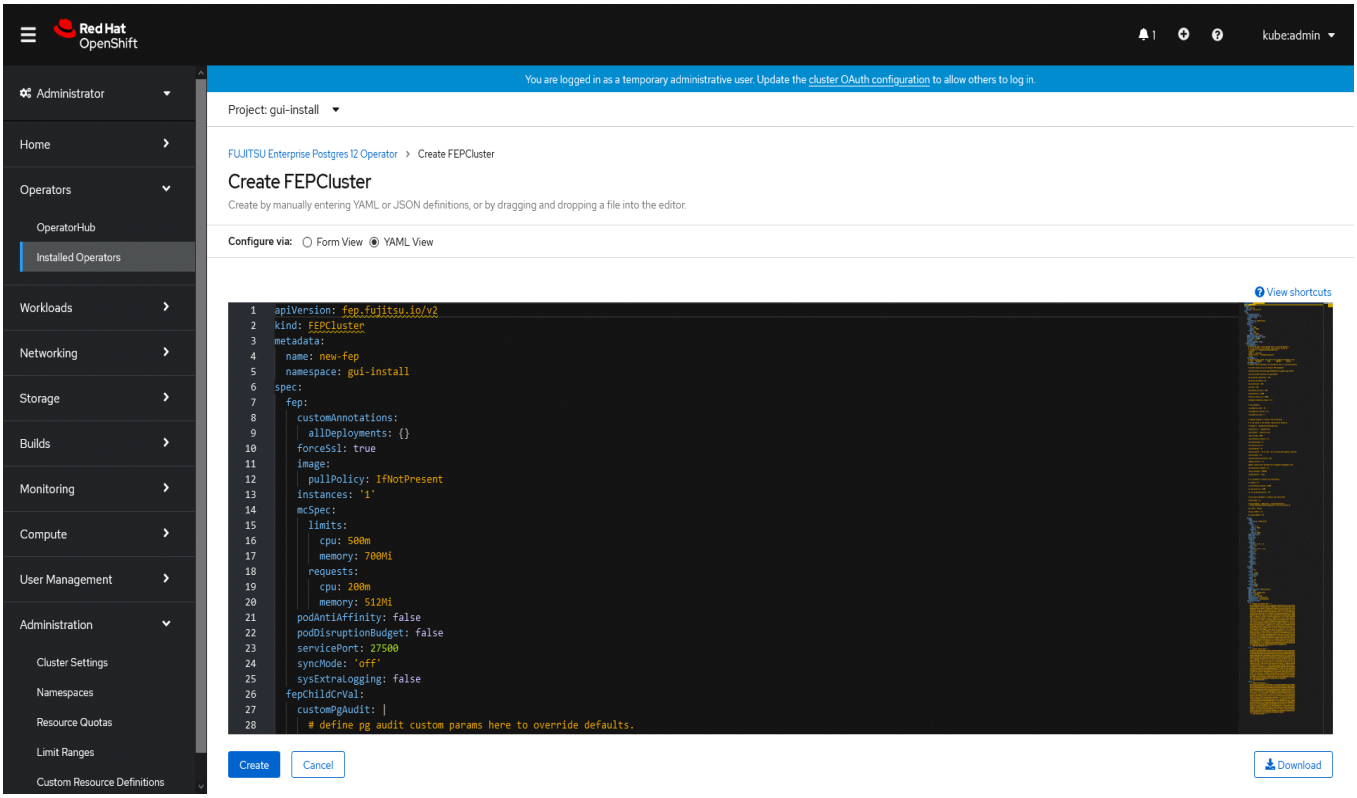
2. It will display a page with all CRs this operator supports. FEPCluster is the main CR and all others are child CR. We would create main CR and all other CRs will be created automatically by Operator.
   To create Cluster CR, either
   (1) Click on "**Create Instance**" under FEPCluster.
   OR
   (2) Click on "**FEPCluster**" on top and then click on "**Create FEPCluster**" on next page.



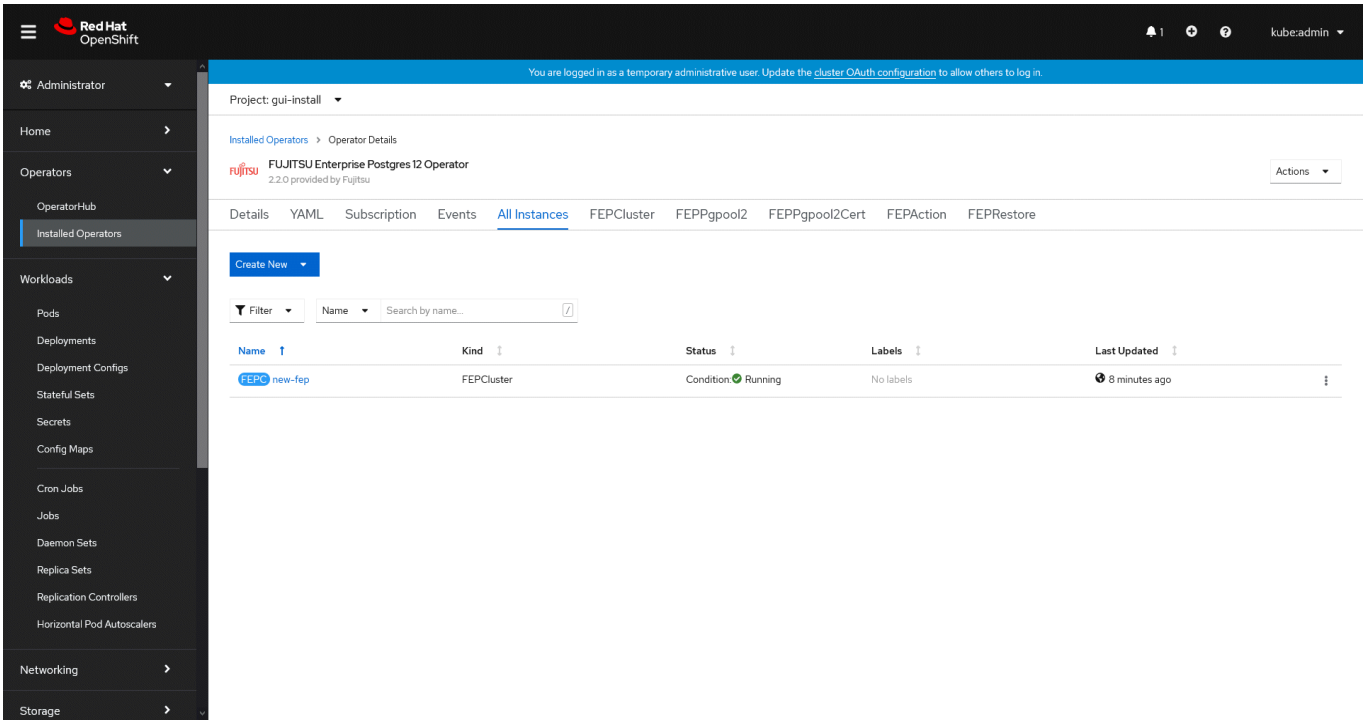
3. This will bring to "Create FEPCluster" page. Here you have two options to configure. The first one is Form View. At the moment, in Form View, one can change only the name of cluster being deployed. Default name is "new-fep".
   This name must be unique within a namespace.



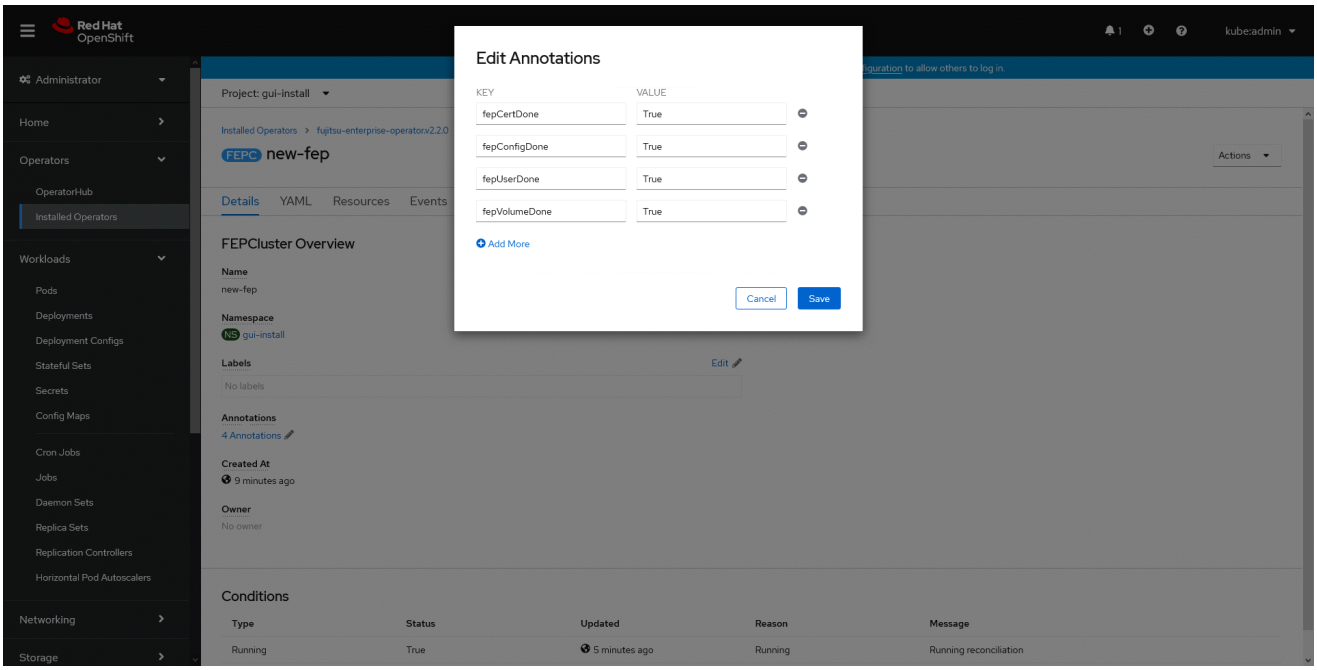
4. In YAML View, starting value of CR is visible and one can choose to modify parameters before creating CR. Refer to the Reference for details of parameters.For example, add a configuration value for the customPgHba parameter according to your environment.



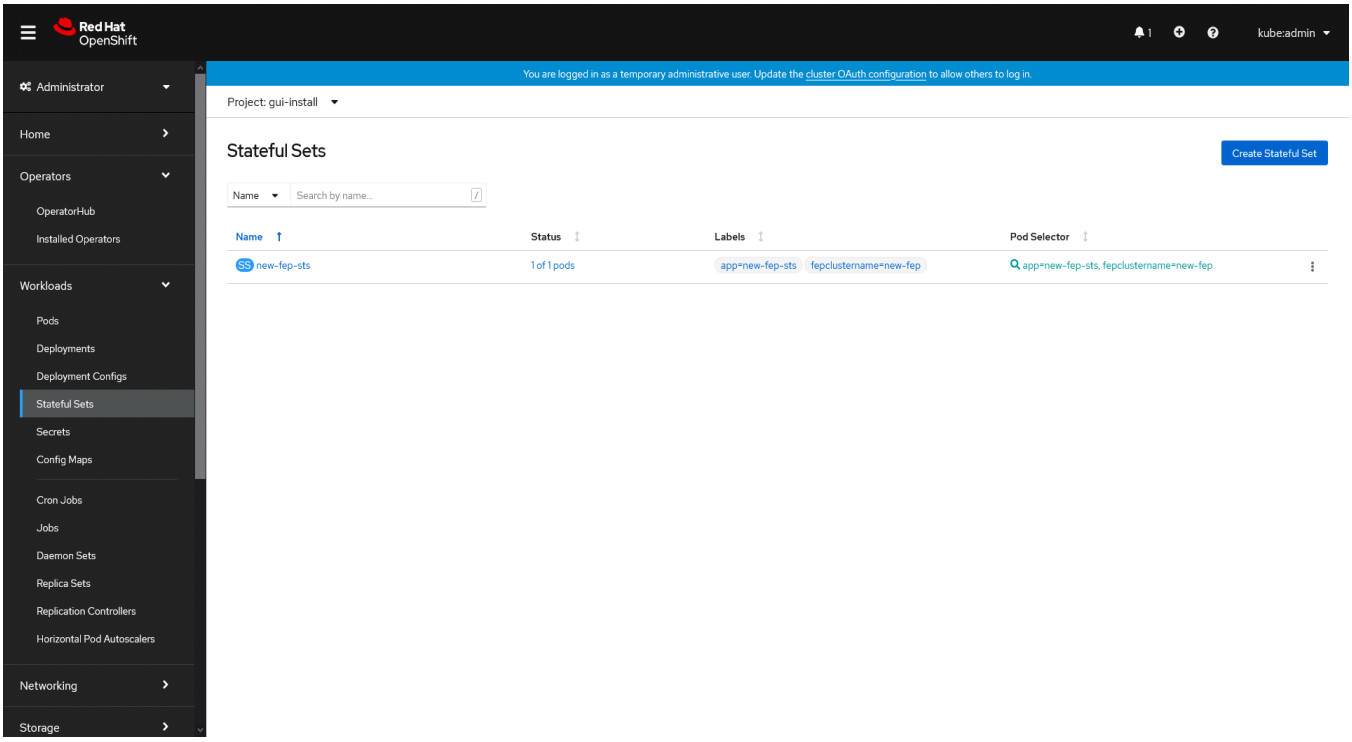
5. When "**Create**" is clicked on either of two pages above, operator creates FEPCluster CR and there after one by one FEPBackup, FEPConfig, FEPVolume, FEPUser and FEPCert child CRs are created automatically.
The starting values for child CRs are taken from "fepChildCrVal" section of FEPCluster CR yaml file. Once child CRs are created, respective values are managed through child CRs only. If you want to change the value, modify the value in FEPCluster "fepChildCrVal" section. Operator reflects changes from FEPCluster parent CR to respective child CRs. Only allowable changes are reflected in child CRs. Child CRs are marked internal objects and hence will not be visible on OCP console. However, you can check child CRs using command line tools.



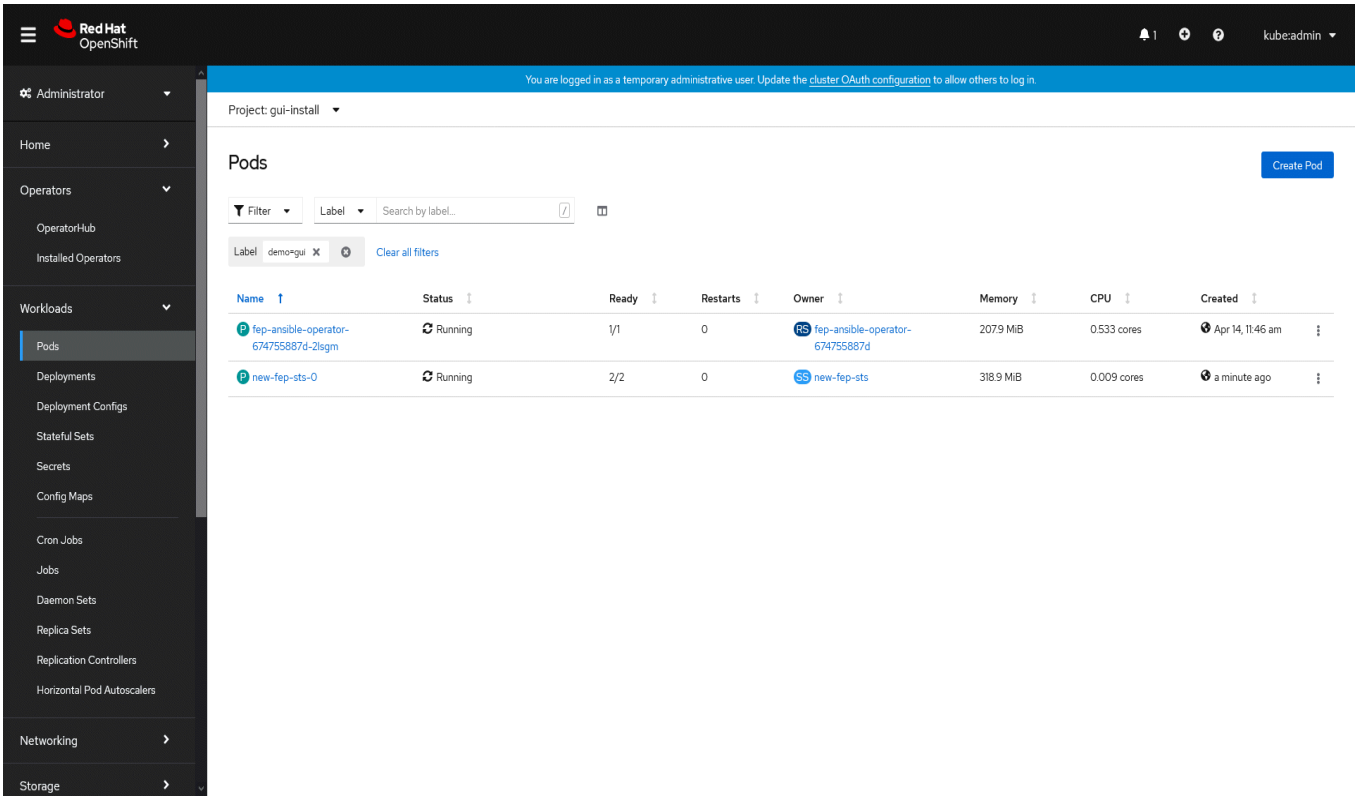
6. In FEPCluster CR, annotations are added to indicate that child CRs are created successfully and has initialized properly. It may take some time to complete.



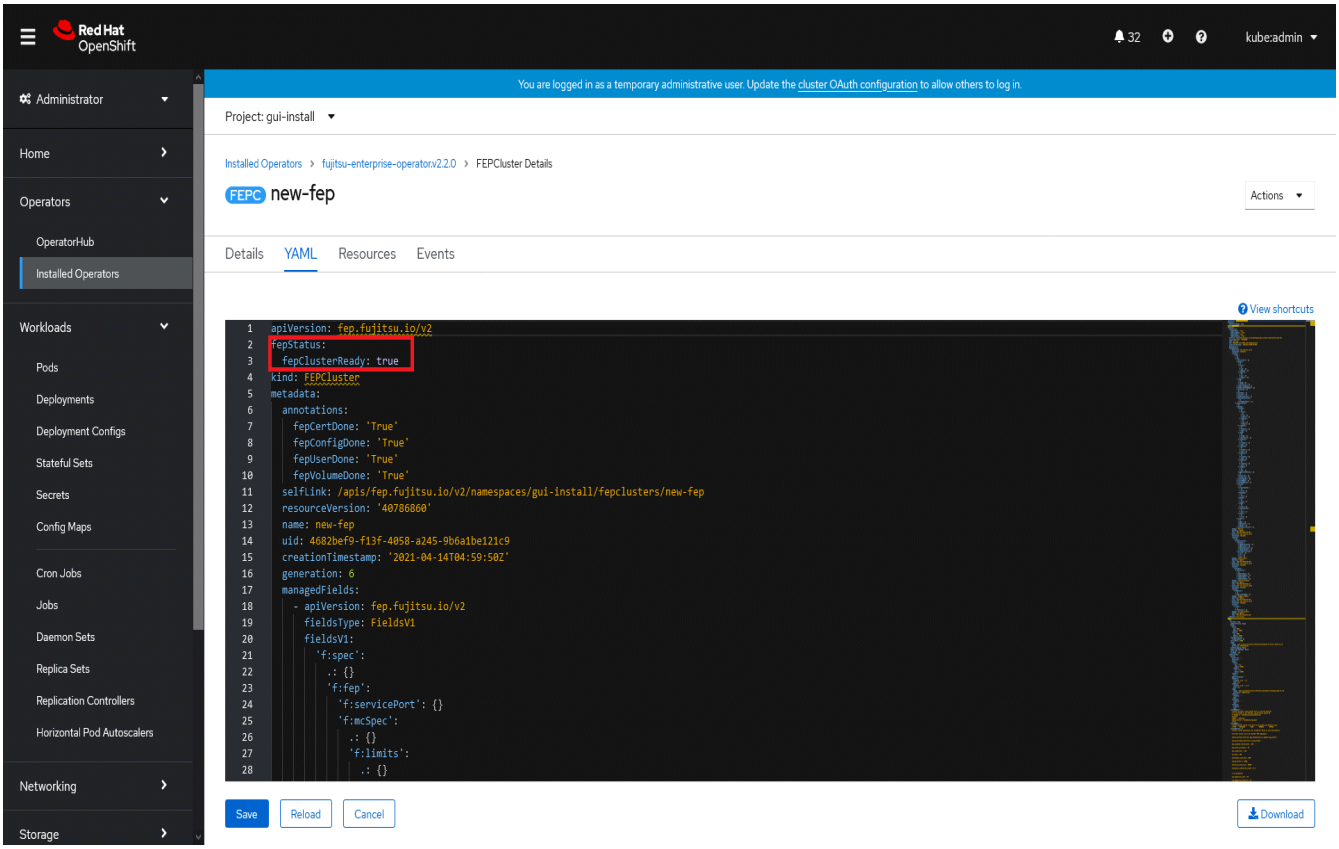
7. Once all four child CRs are marked done in annotations, operator creates StatefulSet for the cluster.



8. StatefulSet will start one FEP instance at one time and will wait it to be ready before starting next one.



9. Once all instances of FEP servers are started, operator marks a flag "fepClusterReady" in "fepStatus" section of CR to be **true**, indicating that FEPCluster is ready for use. Looking at YAML of FEPCluster CR, it would look like as below:



10. Operator also masks the sensitive fields like passwords, passphrase, certificates and keys in FEPCluster fepChildCrVal and also in child CRs.

11. For further information, see [the FUJITSU Enterprise Postgres 12 for Kubernetes Manuals]