PostgreSQL

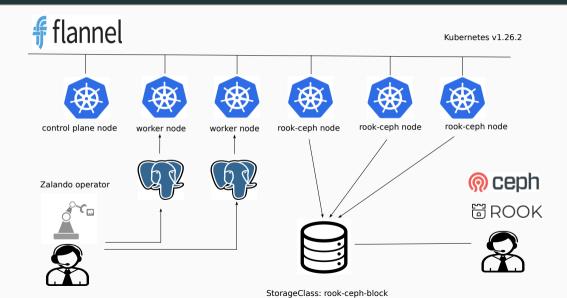
Déploiement de l'opérateur PostgreSQL de Zalando

Simon ELBAZ (selbaz@linagora.com)

27 février 2023

Installation

Architecture



Versions utilisées

- OS de déploiement : Debian 11 Bullseye
- Versions de Kubernetes : 1.26.x

Dimensionnement des serveurs

Dimensionnement du control plane :

- 8 CPU
- 8 Go RAM

Dimensionnement des workers :

- 2 CPU
- 2 Go RAM

Prérequis matériels

- etcd est la base de données clé-valuers centrale utilisée par Kubernetes
- etcd utilise de manière intensive les disques à disposition
- Pour une stabilité accrue du cluster, il est préférable d'utiliser des disques de type
 SSD

Déploiement du nœud control plane

- Kubernetes s'appuie sur un élément essentiel qui est le container runtime.
- La méthode de déploiement du container runtime s'appuie la méthode décrite dans le lien : https://docs.docker.com/engine/install/debian/

Désactivation permanente de la mémoire swap

Le process kubelet ne démarre pas en cas de mémoire swap activée. Pour désactiver l'utilisation de la swap, merci d'utiliser la commande suivante :

```
swapoff -a
```

Pour persister cet état et faire en sorte que la mémoire swap ne soit pas activée au prochain reboot, supprimer ou mettre en commentaires la ligne suivante dans /etc/fstab :

```
$ sudo cat /etc/fstab
/dev/mapper/dnumworker1--vg-root / ext4 errors=remount-ro 0 1
# /boot was on /dev/sdal during installation
UUID=ddd6fd9d-6ac3-4510-9156-22984bc82b67 /boot ext2 defaults 0 2
#/dev/mapper/dnumworker1--vg-swap_1 none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0
```

Installation du runtine container containerd

Mise à jour de l'index du paquet *apt* et installation des paquets nécessaires à l'utilisation des dépôts avec le protocole HTTPS :

```
sudo apt-get update
sudo apt-get install \
    ca-certificates \
    curl \
    gnupg
```

Ajout de la clef GPG officielle de Docker

```
  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings} \\  \  \, \text{curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo chmod a+r /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d /etc/apt/keyrings/docker.gpg} \\  \  \, \text{sudo install -m 0755 -d
```

Ajout du dépôt de Docker

```
echo \
"deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/debian \
"$(./etc/os-release & echo "$VERSION_CODENAME")" stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

Installation de Docker Engine

```
sudo apt-get update sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

Installation de kubectl, kubeadm et kubelet

sudo apt-get install -v kubelet

```
sudo apt-get install -y ca-certificates curl
sudo apt-get install -y apt-transport-https
sudo apt-get update
curl -fsSL https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo gpg --dearmor -o /etc/apt/keyrings/kubernetes-archive-keyring.gpg] \
https://apt.kubernetes.io/ kubernetes-xenial main" | \
sudo tee /etc/apt/sources.list.d/kubernetes.list

sudo apt-get update
sudo apt-get install -y kubectl
sudo apt-get install -v kubeadm
```

Activation des modules kernel *overlay* et *br_netfilter*

```
linagora@debian-cp:/etc/modules-load.d$ cat k8s.conf
overlay
br_netfilter
linagora@debian-cp:/etc/modules-load.d$ pwd
/etc/modules-load.d
```

Activation des fonctions bridge/iptables et forward du kernel

```
linagora@debian-cp:/etc/sysctl.d$ cat k8s.conf
inet.bridge.bridge-nf-call-iptables = 1
net.bridge.bridge-nf-call-ip6tables = 1
net.ipv4.ip_forward = 1
linagora@debian-cp:/etc/sysctl.d$ pwd
/etc/sysctl.d
```

Paramétrage de containerd

Génération du paramétrage par défaut de containerd :

 $\verb|root@debian-cp:"| containerd config | \textbf{default}| | dump > /etc/containerd/config.toml.dmp| \\$

Modifier la valeur à **true** pour le paramètre **SystemdCgroup** :

```
[plugins."io.containerd.grpc.vl.cri".containerd.runtimes.runc.options]
BinaryName = ""
CriuImagePath = ""
CriuWorkPath = ""
CriuWorkPath = ""
IoGid = 0
IoUid = 0
NoNewKeyring = false
NoPivotRoot = false
Root = ""
ShimCgroup = ""
SystemdCgroup = true
```

Paramétrage de containerd

Remplacer le paramétrage actuel par le paramétrage modifié :

```
\label{lem:contined} $$\operatorname{rootdebian-cp:^\# cp /etc/containerd/config.toml /etc/containerd/config.toml.bak rootdebian-cp:^\# cat /etc/containerd/config.toml.dmp > /etc/containerd/config.toml rootdebian-cp:^\# systemctl restart containerd
```

Initialisation du cluster Kubernetes

En tant que root, lancer la commande suivante :

```
# kubeadm init --control-plane-endpoint 10.10.10.30 \
   --skip-phases=addon/coredns,addon/kube-proxy \
   --v=5 \
   --ppd-network-cidr="10.244.0.0/16"
```

Si les phases *addon/coredns* et *addon/kube-proxy* ne sont pas évitées au 1^{er} lancement de kubeadm, l'erreur suivante est générée :

[kubelet-finalize]Updating "/etc/kubernetes/kubelet.conf" to point to a rotatable kubelet client certificate and key error execution phase addon/coredns: unable to fetch CoreDNS current installed version and ConfigMap.: rpc error: code = Unknown desc = malformed header: missing HTTP content-type To see the stack trace of this error execute with -v=5 or higher

Initialisation du cluster Kubernetes

Le résultat de la commande d'init est le suivant :

10315 01 :06 :38.342010 34405 kubeletfinalize.go :134][kubelet-finalize]Restarting the kubelet to enable client certificate rotation

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user :

```
mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -a) $HOME/.kube/config
```

Alternatively, if you are the root user, you can run:

```
export KUBECONFIG=/etc/kubernetes/admin.conf
```

You should now deploy a pod network to the cluster.

Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at :

https://kubernetes.io/docs/concepts/cluster-administration/addons/

You can now join any number of control-plane nodes by copying certificate authorities and service account keys on each node and then running the following as root:

```
kubeadm join 10.10.10.30:6443 --token 6pia7c.n6u8phm7y]16nnr8 \
--discovery-token-ca-cert-hash sha256:f6d45602ea75c7659dc91f661d19e97e6817e2847e4e5d0047880b871317a145 \
--control-plane
```

Then you can join any number of worker nodes by running the following on each as root :

```
kubeadm join 10.10.10.30:6443 --token 6pia7c.n6u8pbm7yj16nnr8 \
--discovery-token-ca-cert-hash sha256:f6d45602ea75c7659dc91f661d19e97e6817e2847e4e5d0047880b871317a145
```

Paramétrage de kubectl

L'utilisation de kubectl nécessite l'action suivante :

```
\label{local_model} $$ \mbox{mkdir -p $HOME/.kube} $$ \mbox{sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config} $$ \mbox{sudo chown $(id -u):$(id -g) $HOME/.kube/config} $$
```

Déploiement de l'addon CoreDNS

Comme indiqué précédemment, les addons CoreDNS et Kube-Proxy n'ont pas été déployés au $\mathbf{1}^{er}$ lancement de kubeadm.

CoreDNS peut maintenant être déployé sans erreur :

linagora@debian-cp:~\$ sudo kubeadm init phase addon coredns [addons] Applied essential addon: CoreDNS

Déploiement de l'addon Kube-Proxy

linagora@debian-cp:~\$ sudo kubeadm init phase addon kube-proxy [addons] Applied essential addon: kube-proxy

Choix de la couche réseau - Container Network Interface

Il existe différentes addons Kubernetes implémentant l'interface CNI.

Ces addons sont listés dans l'URL suivante :

https://kubernetes.io/docs/concepts/cluster-administration/addons Pour le POC, l'addon sélectionné est Flannel car il semble être le plus simple et le plus

basique des addons CNI.

Déploiement de l'addon Flannel

L'addon Flannel s'installe de plusieurs manières

(https://github.com/flannel-io/flannel#deploying-flannel-manually La méthode utilisée pour le POC est kubectl:

kubectl apply -f https://github.com/flannel-io/flannel/releases/latest/download/kube-flannel.yml

Installation de k9s

Un outil pratique de visualisation d'un cluster kubernetes est : k9s (https://k9scli.io/)

Le lien suivant permet de télécharger l'archive incluant le binaire :

https://github.com/derailed/k9s/releases/download/v0.27.3/k9s_Lin

Liste des namespaces

```
linagora@debian-cp:~$ kubectl get namespaces
NAME
                 STATUS
                          AGE
default
                 Active
                          40d
kube-flannel
                 Active
                         39d
kube-node-lease
                Active
                         40d
kube-public
                 Active
                          40d
kube-system
                 Active
                         40d
minio-operator
                 Active
                         32d
rook-ceph
                 Active
                          32d
```



Pods du namespace default

```
linagora@debian-cp:~$ kubectl get pods
NAME
                                        READY
                                                STATUS
                                                          RESTARTS
                                                                     AGE
acid-test-cluster-0
                                        1/1
                                                Running
                                                                     27d
acid-test-cluster-1
                                        1/1
                                                Running
                                                                      27d
postgres-operator-fcbd7cc96-ndpj8
                                        1/1
                                                Running
                                                                     40d
postgres-operator-ui-5579cc7779-86rgk
                                        1/1
                                                Running
                                                                      40d
```



Pods du namespace kube-system

linagora@debian-cp:~\$ kubectl get	pods -n	kube-system		
NAME	READY	STATUS	RESTARTS	AGE
coredns-787d4945fb-8ph9v	1/1	Running	0	40d
coredns-787d4945fb-9jrzs	1/1	Running	0	40d
etcd-debian-cp	1/1	Running	158	41d
kube-apiserver-debian-cp	0/1	Running	4968 (13m ago)	41d
kube-controller-manager-debian-cp	1/1	Running	4161 (8m26s ago)	41d
kube-proxy-4mfn8	1/1	Running	0	33d
kube-proxy-9h4c6	1/1	Running	0	27d
kube-proxy-9j47t	1/1	Running	0	33d
kube-proxy-s78vx	1/1	Running	0	33d
kube-proxy-wpwt4	1/1	Running	0	40d
kube-proxy-xjs5q	1/1	Running	1 (33d ago)	41d
kube-scheduler-debian-cp	1/1	Running	2848 (6m20s ago)	41d



Pods du namespace kube-flannel

```
linagora@debian-cp:~$ kubectl get pods -n kube-flannel
NAME
                        READY
                                STATUS
                                          RESTARTS
                                                         AGE
kube-flannel-ds-5nw2j
                        1/1
                                Running
                                          0
kube-flannel-ds-5xwsm
                        1/1
                                Running
                                                         40d
kube-flannel-ds-8vkg9
                        1/1
                                Running
                                          1 (33d ago)
                                                         40d
kube-flannel-ds-pv6ss
                        1/1
                                Running
                                                         27d
kube-flannel-ds-trbz9
                        1/1
                                Running
kube-flannel-ds-wmzz2
                        1/1
                                Running
```



Pods du namespace rook-ceph

linagora@debian-cp:-S kubectl get pods -n rook-ceph				
NAME	READY	STATUS	RESTARTS	AGE
csi-cephfsplugin-9nbts	2/2	Running	1 (27d ago)	27d
csi-cephisplugin-bpxlw	2/2	Running	0 (27d ago)	33d
	2/2	Running	0	33d
csi-cephfsplugin-jd5x8				
csi-cephfsplugin-mddkf	2/2	Running	0	33d
csi-cephfsplugin-nrmfz	2/2	Running	0	33d
csi-cephfsplugin-provisioner-84cc595b78-9mm14	5/5	Running	6008 (2m44s ago)	33d
csi-cephfsplugin-provisioner-84cc595b78-9twnq	5/5	Running	2171	33d
csi-rbdplugin-92zlq	2/2	Running	0	33d
csi-rbdplugin-c95w7	2/2	Running	0	33d
csi-rbdplugin-pk57s	2/2	Running	1 (27d ago)	27d
csi-rbdplugin-provisioner-6f6b6b8cd6-4c8jd	1/5	CreateContainerError	1344	33d
csi-rbdplugin-provisioner-6f6b6b8cd6-gw6bm	1/5	CreateContainerError	4465	33d
csi-rbdplugin-srtfz	2/2	Running	0	33d
csi-rbdplugin-v6gqm	2/2	Running	0	33d
rook-ceph-crashcollector-dnumcephworker1-7845bb8ff-vs9fx	1/1	Running	0	32d
rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-n5xsz	1/1	Running	0	33d
rook-ceph-crashcollector-dnumcephworker3-6fddb6cd9-x45w5	1/1	Running	1 (8d ago)	32d
rook-ceph-mgr-a-c5db58dff-hvsp9	3/3	Running	1487 (6d6h ago)	33d
rook-ceph-mgr-b-7bbfd88c8b-wh4ww	2/3	CreateContainerError	944	22d
rook-ceph-mon-a-75cf9ccddc-b2jgc	2/2	Running	1163	33d
rook-ceph-mon-b-78d6586d5-qss4z	1/2	CreateContainerError	701 (19d ago)	19d
rook-ceph-mon-c-64dcb4c86c-wz8sg	2/2	Running	1755	33d
rook-ceph-operator-cf4f7dfd4-6tm6p	1/1	Running	0	32d
rook-ceph-osd-0-57d9b8db4d-d6dhr	1/2	CreateContainerError	484	32d
rook-ceph-osd-1-74698f77fd-6n2mh	1/2	Running	529	32d
rook-ceph-osd-2-5cc486467c-1hm47	1/2	Running	1116 (49m ago)	32d
rook-ceph-osd-prepare-dnumcephworker1-rnk78	0/1	Completed	0	21d
rook-ceph-osd-prepare-dnumcephworker3-42rxv	0/1	Completed	0	21d
rook-ceph-tools-7c4b8bb9b5-pxk67	1/1	Running	0	33d



Déploiement du nœud worker

Sur chacun des 2 workers, il est nécessaire de déployer :

- le runtime containerd de Docker
- les commandes kubectl, kubeadm et kubelet
- l'activation des modules kernel overlay et br_netfilter
- l'activation des fonctions bridge/iptables et forward du kernel
- le paramétrage de containerd

Ajout du nœud worker dans le cluster k8s - join

L'opération qui permet au nœud worker de rejoindre le cluster s'appelle le join. La syntaxe de cette commande est obtenue en lançant la commande suivante sur le control plane avec l'utilisateur root :

```
# kubeadm token create --print-join-command
kubeadm join 10.10.10.30:6443 \
--token ilfbgc.8xco4svm5pnxkfbj \
--discovery-token-ca-cert-hash sha256:73bf45619ae0051d4ff810328d1dadc18e6a5966c95d3c4ec76275b89a934595
```

Lancement du join sur chacun des workers

Sur chacun des workers, le lancement de la commande join produit le résultat suivant :

```
# kubeadm join 10.10.10.30:6443 \
--token 6pia7c.n6u8pbm7vil6nnr8 \
--discovery-token-ca-cert-hash sha256:f6d45602ea75c7659dc91f661d19e97e6817e2847e4e5d0047880b871317a145
[preflight] Running pre-flight checks
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system \
get cm kubeadm-config -o vaml'
W0315 16:31:41.445771 6266 configset.go:781 Warning: No kubeproxy.config.k8s.io/vlalphal config is loaded.
Continuing without it: configmaps "kube-proxy" is forbidden: User "system:bootstrap:6pia7c"
cannot get resource "configmaps" in API group "" in the namespace "kube-system"
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.vaml"
[kubelet-start] Writing kubelet environment file with flags to file "/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...
This node has joined the cluster:
* Certificate signing request was sent to apiserver and a response was received.
* The Kubelet was informed of the new secure connection details.
Run 'kubectl get nodes' on the control-plane to see this node join the cluster.
```

Lancement du join sur chacun des workers

La commande suivante permet de vérifier le résultat du join :

```
        $ kubect1 get
        nodes

        NAME
        STATUS
        ROLES
        AGE
        VERSION

        debian-cp
        NotReady
        control-plane
        15h
        v1.26.2

        dnumworker1
        NotReady
        <none>
        53s
        v1.26.2
```

Terminologie du stockage dans k8s

- Le stockage permanent des données s'appuie les volumes persistants (PV)
 (https://kubernetes.io/docs/concepts/storage/persistent-volumes/)
- Un PV est un espace de stockage mis à disposition par k8s.
- Il peut être alloué manuellement ou dynamiquement par l'intermédiaire des storage class (https://kubernetes.io/docs/concepts/storage/storage-classes/)
- Les PV sont l'équivalent d'un node dans un cluster.
- Les persistentVolumeClaim (PVC) sont l'équivalent d'un pod.

Déploiement du stockage - Rook Ceph

- Le storage class sur lequel s'appuie l'opérateur PostgreSQL est Ceph
- L'opérateur k8s Rook Ceph facilite le déploiement de Ceph
- Le déploiement s'appuie sur le lien https://rook.io/docs/rook/v1.9/quickstart.html
- La version de l'opérateur utilisée est la v1.9
- Elle supporte les versions k8s v1.17+

Prérequis au déploiement de l'opérateur - Rook Ceph

- Le déploiement de l'opérateur scanne l'ensemble des noeuds de stockage pour vérifier la présence de :
 - des devices bruts (sans partitions ou filesystems formattés)
 - des partitions brutes (sans filesystems formattés)
 - les volumes physiques initialisés par LVM

L'exemple ci-dessous indique comment vérifier la disponibilité d'espace pour l'opérateur Rook Ceph :

```
lsblk -f
    NAME
                          ESTYPE
                                       LABEL UUITD
                                                                                     MOUNTPOINT
    vda
    I-vda1
                          LVM2 member
                                             >eSO50t-GkUV-YKTH-WsGg-hNJY-eKNf-3i07IB
    |-ubuntu--vg-root
                         ext 4
                                            c2366f76-6e21-4f10-a8f3-6776212e2fe4
                                            9492a3dc-ad75-47cd-9596-678e8cf17ff9
    I-ubuntu--vg-swap 1 swap
                                                                                    [SWAP]
    vdb
```

Prérequis au déploiement de l'opérateur - Rook Ceph

- Dans l'exemple précédent, si la colonne FSTYPE est renseignée, cela indique la présence d'un filesystem
- La partition vdb n'est pas formatée avec un filesystem : elle est donc utilisable par l'opérateur Rook Ceph
- Le paquet lvm2 est une dépendance importante de Rook Ceph

Sélection des nœuds sur lesquels Ceph sera déployé

L'opérateur Rook Ceph offre la possibilité de sélectionner les nœuds sur lesquels le stockage Ceph est déployé.

Pour cela, il s'appuie sur la notion de label.

Dans le cadre du POC, les 3 nœuds suivants sont sélectionnés pour porter le stockage :

- dnumcephworker1
- dnumcephworker2
- dnumcephworker3

Affectation des labels sur les nœuds de stockage

Depuis le control plane, lancer les commandes suivantes pour marquer les nœuds :

```
$ kubectl label nodes dnumcephworker1 role=storage-node
node/dnumcephworker1 labeled
$ kubectl label nodes dnumcephworker2 role=storage-node
node/dnumcephworker2 labeled
$ kubectl label nodes dnumcephworker3 role=storage-node
node/dnumcephworker3 labeled
```

Affichage du label des nœuds :

\$ kubectl get nodesshow-labels					
NAME	STATUS	ROLES	LABELS		
dnumcephworkerl	Ready	<none></none>	kubernetes.io/hostname=dnumcephworker1, kubernetes.io/os=linux, role=storage-node		
dnumcephworker2	Ready	<none></none>	kubernetes.io/hostname=dnumcephworker2, kubernetes.io/os=linux, role=storage-node		
dnumcephworker3	Ready	<none></none>	$\verb kubernetes.io/hostname=dnumcephworker3 , \verb kubernetes.io/os=linux , \verb role=storage-node $		

Paramétrage pour la répartition du stockage Ceph sur les nœuds labelisés

```
diff --git a/deploy/examples/cluster.vaml b/deploy/examples/cluster.vaml
index 9bd50ec97..fef3f777f 100644
--- a/denlov/evamples/cluster vaml
+++ h/denlow/evamples/cluster vaml
88 -154,22 +154,22 88 spec:
   # To control where various services will be scheduled by kubernetes, use the placement configuration sections below.
   # The example under 'all' would have all services scheduled on kubernetes nodes labeled with 'role-storage-node' and
   # tolerate taints with a key of 'storage-node'
- # nlacement:
- # all:
         nodeAffinity:
              - key: role
                operator: In
                - storage-node
         podAffinity:
         podAntiAffinity:
         topologySpreadConstraints:
         - kev: storage-node
+ placement;
           nodeSelectorTerms:
           - matchExpressions:
           - kev: role
              operator: In
               values:
              - storage-node
      nodaffinity:
      podAntiAffinity:
      topologySpreadConstraints:
      - key: storage-node
         operator: Exists
   # The above placement information can also be specified for mon. osd, and mor components
   # Monitor deployments may contain an anti-affinity rule for avoiding monitor
```

Paramétrage pour la répartition du stockage Ceph sur les nœuds labelisés

La directive *nodeSelectorTerms* permet de sélectionner les noeuds portant la storageclass Ceph

```
+ nodeSelectorTerms:

+ - storage-node

+ podAffinity:
+ podAntiAffinity:
```

Déploiement de l'opérateur Rook Ceph

Comme indiqué dans le lien

https://rook.io/docs/rook/v1.9/quickstart.html, l'application des commandes ci-dessous amorce le déploiement de l'opérateur :

```
$ git clone --single-branch --branch v1.9.2 https://github.com/rook/rook.git
cd rook/deploy/examples
kubectl create -f crds.yaml -f common.yaml -f operator.yaml
kubectl create -f cluster.yaml
```

- Une fois le cluster opérationnel, il devient possible de créer :
 - stockage bloc
 - stockage objet
 - stockage fichier

Vérification de l'opérateur Rook Ceph

verify the rook-ceph-operator is in the 'Running' state before proceeding kubectl -n rook-ceph get pod

cal - htdpluid-provisioner - 6f6b6b8cd-gw6bm 1/5 CreateContaineTErro 465 69d cal - htdpluid-nertfs 2/2 Running 0 69d cal - htdpluid-nertfg 2/2 Running 0 69d cook-ceph-crashcollector-dnumcephworker1-7845b8ff-wa9fx 1/1 Pending 0 4d10 rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-13cd 1/1 Terrainting 0 69d rook-ceph-arashcollector-dnumcephworker2-75cdf95dcd-n3cd 1/1 Terrainting 2 69d rook-ceph-arashcollector-dnumcephworker2-75cdf95dcd-n3cd 1/1 Running 1 69d rook-ceph-arashcollector-dnumcephworker2-75cdf95dcd-n4sb 1/1 Running 1 69d rook-ceph-aras-acd5b8dff-fpg7s 2/2 CrashcopBackoff 146 (28d apc) 3dd rook-ceph-aras-acd5b8dff-frups9 2/3 Terninating 150d (31d apc) 69d rook-ceph-aras-bc-5bdffedbe-bdyg 2/2 Running 1500 (31d apc) 69d rook-ceph-arc-6bdcodfc-b2jgc 2/2 Running 1500 (31d apc) 69d </th <th></th> <th></th> <th></th> <th></th> <th></th>					
cai-cephfaplugin-bpklw 2/2 Running 0 69d cai-cephfaplugin-indx8 2/2 Running 0 69d cai-cephfaplugin-mdx8 2/2 Running 0 69d ci-cephfaplugin-motors 2/2 Running 652 28d ago 69d ci-cephfaplugin-provisioner-84cc59b78-9ml4 5/5 Running 652 28d ago 69d ci-cephfaplugin-provisioner-84cc59b78-9ml4 5/5 Running 0 69d ci-chplugin-952ml 2/2 Running 1 63d ago 69d ci-chplugin-provisioner-6f6b88d6-4c8jd 5/5 Terminating 1 (3d ago) 69d ci-chplugin-provisioner-6f6b88d6-4c8jd 5/5 Terminating 1 (3d ago) 69d ci-chplugin-provisioner-6f6b88d6-4c8jd 5/5 Terminating 1 (3d ago) 69d ci-chplugin-provisioner-6f6b88d6-4c9bd 1/5 CreateContaineError 40c 64d ci-chplugin-provisioner-6f6b88d6-4c9bd 1/5 Running 0 60d 64d 64d 64d-1-chplugin	NAME	READY	STATUS	RESTARTS	AGE
cai-cephfaplugin-jdx8 2/2 Running 0 69d cai-cephfaplugin-mdxf 2/2 Running 0 69d cai-cephfaplugin-mcmfz 2/2 Running 0 69d ci-cephfaplugin-provisioner-84cc595b78-9ml4 5/5 Running 398 (30d ago) 69d ci-cephfaplugin-provisioner-84cc595b78-9twnq 2/2 Running 0 69d ci-rhdplugin-provisioner-8fc6b6b8cd 2/2 Running 0 69d ci-rhdplugin-provisioner-6fc6b6b8cd6-4ct3d 5/5 Terminating 2919 (29d ago) 60d ci-rhdplugin-provisioner-6fc6b6b8cd6-4ct5d 1/3 Running 0 60d ci-rhdplugin-provisioner-6fc6b6b8cd6-4ct5d 1/3 Running 0 60d ci-rhdplugin-provisioner-6fc6			Running	1 (63d ago)	
cai-cephfaplugin-mdakf 2/2 Running 0 69d cai-cephfaplugin-mrmfa 2/2 Running 652 (28d ag) 69d cai-cephfaplugin-provisioner-84cc59b78-9ml 5/5 Running 652 (28d ag) 69d ci-cephfaplugin-provisioner-84cc59b78-9ml 5/5 Running 0 69d ci-cephfaplugin-provisioner-84cc59b78-9fwq 2/2 Running 1 63d ag) 69d ci-r-chplugin-pS78 2/2 Running 1 63d ag) 63d ci-r-chplugin-pF0vfsioner-6f6b88d64-de5jd 5/5 Ferninating 19/ 20d ag) 69d ci-r-chplugin-pF0vfsioner-6f6b88d64-de5jd 5/5 Ferninating 10 (30d ag) 63d ci-r-chplugin-pF0vfsioner-6f6b88d64-de5jd 5/5 Ferninating 10 (30d ag) 69d ci-r-chplugin-pF0vfsioner-6f6b88d6-4c96bm 1/5 CreateContaineError 46 60d ci-r-chplugin-pF0vfsioner-6f6b88d6-4c96bm 1/5 Running 0 60d ci-r-chplugin-pF0vfsioner-6f6b88d6-4c96bm 1/5 Running 0 60d	csi-cephfsplugin-bpxlw	2/2	Running	0	69d
cai-ephfaplugin-mrff 2/2 Running 0 69 cai-ephfaplugin-provisioner-84cc595b78-9ml4 5/5 Running 3908 (30d ago) 69d cai-ephfaplugin-provisioner-84cc595b78-9ml4 5/5 Running 3908 (30d ago) 69d cai-rhchglugin-921q 2/2 Running 0 69d cai-rhchglugin-p857s 2/2 Running 163 (30 ago) 69d cai-rhchglugin-p867s 2/2 Running 291 (29d ago) 69d cai-rhchglugin-provisioner-6f6b88d6-det5d 6/5 Terninating 291 (29d ago) 69d cai-rhchglugin-provisioner-6f6b88d6-q4t5d 1/5 CreateContainerError 465 69d cai-rhchglugin-provisioner-6f6b88d6-q4t5d 1/5 Runding 0 69d cai-rhchglugin-provisioner-6f6b88d6-q4t5d 1/5 Runding 0 69d cai-rhchglugin-provisioner-6f6b88d6-q4tbm 1/5 Runding 0 69d cai-rhchglugin-provisioner-6f6b88d6-q4tbm 1/5 Runding 0 69d cai-rhchglugin-provisioner-6f6b8d6d7-q4tbm <	csi-cephfsplugin-jd5x8	2/2	Running	0	69d
cai-caphfaplugin-provisioner-84cc595b78-9mal4 5/5 Running 6523 (28d ago) 69d cai-caphfaplugin-provisioner-84cc595b78-9twnq 5/5 Running 308 (30d ago) 69d cai-caphfaplugin-provisioner-84cc595b78-9twnq 2/2 Running 0 69d cai-rhchplugin-pSy3 2/2 Running 1 (5d ago) 69d cai-rhchplugin-pFovisioner-6f6bb8ed6-4c8jd 5/5 Feminating 219/ (29d ago) 60d cai-rhchplugin-provisioner-6f6bb8ed6-4c8jd 1/5 CreateContainerError 465 60d cai-rhchplugin-provisioner-6f6bb8ed6-4c96bm 1/5 CreateContainerError 465 60d cai-rhchplugin-provisioner-6f6bb8ed6-4c96bm 1/5 Running 0 60d cai-rhchplugin-provisioner-6f6bb8ed6-4c96bm 1/5 Running 0 60d cai-rhchplugin-provisioner-6f6bb8ed6-4c96bm 1/5 Running 0 60d cai-rhchplugin-brovisioner-6f6bb8ed6-4c96bm 1/5 Running 0 60d cai-rhchplugin-brovisioner-6f6bb8ed6-4c96bm 1/2 Running 0 60d	csi-cephfsplugin-mddkf	2/2	Running	0	69d
cai-ephfaplugin-provisioner-84c595b78-9twnq 5/5 Running 390 (304 ag) 69d cai-rhdplugin-92tiq 2/2 Running 0 69d cai-rhdplugin-p857s 2/2 Running 0 60d cai-rhdplugin-p857s 2/2 Running 1 (3d ago) 69d cai-rhdplugin-provisioner-6f6b8b8cd6-4c9jd 5/5 Terninating 291 (29d ago) 69d cai-rhdplugin-provisioner-6f6b8b8cd6-4c9tbm 1/5 CreateContainerError 465 69d cai-rhdplugin-provisioner-6f6b8b8cd6-qwbm 1/5 Running 0 69d cai-rhdplugin-provisioner-6f6b8b8cd6-qwbm 1/5 Running 0 69d cai-rhdplugin-provisioner-6f6b8b8cd6-qwbm 1/2 Running 0 69d cai-rhdplugin-provisioner-6f6b8b8cd6-qwbm 1/2 Running 0 69d cai-rhdplugin-provisioner-6f6b8b8cd6-qwbm 1/2 Running 0 69d cai-rhdplugin-provisioner-6f6b8cd6-qwbm 1/2 Running 0 69d cai-rhdplugin-provisioner-6f6b8cd6-qwbm 1/2	csi-cephfsplugin-nrmfz	2/2	Running	0	69d
cai - hopluyin-921q 2/2 Running 0 69d cai - hopluyin-059W 2/2 Running 1 (5d ago) 69d cai - hopluyin-pK97s 2/2 Running 1 (5d ago) 69d cai - hopluyin-pF0v1sioner-6f6bb86d6-4c8jd 5/5 Foreiniating 2919 (29d ago) 60d cai - hopluyin-pF0v1sioner-6f6bb86d6-4c4b56 1/5 Pending 0 60d ci - hopluyin-pF0v1sioner-6f6bb86d6-4c4b66 1/5 CreateContainerErro 465 69d ci - hopluyin-ertris 2/2 Running 0 69d cob-ceph-erashoollector-d-muscephworker2-75cdf95dcd-hyd 1/1 Running 0 69d rook-ceph-eras	csi-cephfsplugin-provisioner-84cc595b78-9mm14	5/5	Running	6523 (28d ago)	69d
cal-hcplugin-098w 2/2 Running 0 69 cal-hcplugin-p68/Ts 2/2 Running 1 (3 dag) 64 cal-hcplugin-pcv4:ioner-6f6bb8cd6-409d 5/5 Terminating 2919 (29d ago) 69d cal-hcplugin-provisioner-6f6bb8cd6-q6b8 1/5 Pending 0 645 69d cal-hcplugin-provisioner-6f6bb8cd6-q6b8 1/2 Running 0 69d 69d cal-hcplugin-ertfa 2/2 Running 0 69d 69d cook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-13vq 1/1 Pending 0 68d rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-13vq 1/1 Pending 0 69d rook-ceph-arashcollector-dnumcephworker2-75cdf95dcd-13vq 1/1 Pending 0 69d rook-ceph-arashcollector-dnumcephworker2-75cdf95dcd-13vq 1/1 Pending 0 69d rook-ceph-arashcollector-dnumcephworker2-75cdf95dcd-13vq 1/1 Pending 1 69d rook-ceph-arashcollector-dnumcephworker3-75cdf95dcd-13vq 1/2 Tersinating 146 (28d a	csi-cephfsplugin-provisioner-84cc595b78-9twnq	5/5	Running	3908 (30d ago)	69d
cai-r-bqluqin-pK57a 2/2 Running 1 (5d ago) 63 cai-r-bqluqin-provisioner-eff6bb8ede-det5d 5/5 Foreiniating 2919 (294 ago) 64 cai-r-bqluqin-provisioner-eff6bb8ede-q46ba 1/5 Pending 0 60 ci-r-bqluqin-provisioner-eff6bb8ede-q46ba 1/5 CreateContaineError 465 69d ci-r-bqluqin-ertfz 2/2 Running 0 69d ci-r-bqluqin-ertfz 2/2 Running 0 69d rook-eeph-crashcollector-d-muscephworkerl-78db95dd-1ydg 1/1 Running 0 60 rook-eeph-crashcollector-d-muscephworkerl-75cdf95dd-1ydg 1/1 Pending 0 69d rook-eeph-crashcollector-d-muscephworkerl-75cdf95dd-4ydg 1/1 Terninating 0 69d rook-eeph-arg-a-d5b8fdf-fyp7z 2/3 CrashcopBacoff 16 (28d ag) 30d rook-eeph-arg-a-d5b8fdf-fyp7z 2/3 Terninating 10 401 rook-eeph-arg-a-d5b8fdf-fyp6p 2/3 Terninating 10 40 rook-eeph-arg-a-d5b8fdf-fyp8p2 <t< td=""><td>csi-rbdplugin-92zlq</td><td>2/2</td><td>Running</td><td>0</td><td>69d</td></t<>	csi-rbdplugin-92zlq	2/2	Running	0	69d
cal - htdplujn-provisioner - offobbedde-debid 5/5 Terminating 291 (294 ago) 69d cal - htdplujn-provisioner - offobbedde-deb5 0/5 Pending 0 445 69d cal - htdplujn-provisioner - offobbedde-gw6bm 1/5 CreateContainerError 465 69d cal - htdplujn-retfx 2/2 Running 0 0 69d cal - htdplujn-retfx 2/2 Running 0 0 69d rob-teph-reashcollector-dnumeephworker1-7845bb8ffvw8fx 1/1 Running 0 0 69d rob-ceph-cashcollector-dnumeephworker2-75cdff55dd-11xq 1/1 Pending 0 0 69d rob-ceph-agn-a-do5b68ff-fygg 1/1 Running 2 0 69d rob-ceph-agn-a-do5b8ff-fybp3 2/3 CreathcoBackoff 146 (28d ago) 30d rob-ceph-agn-b-Tbbfd8cb-Jdgfp 2/3 Terninating 146 (28d ago) 30d rob-ceph-agn-b-Tbbfd8cb-Jdgfp 2/3 Running 150 (31d ago) 69d rob-ceph-agn-b-Tbbfd8cb-Jdgfp 2/3 Running <td>csi-rbdplugin-c95w7</td> <td>2/2</td> <td>Running</td> <td>0</td> <td>69d</td>	csi-rbdplugin-c95w7	2/2	Running	0	69d
cal-rhqluqin-provisioner-ef6b6b8cd6-q4t56 0/5 Pending 0 4d10 cal-rhqluqin-provisioner-ef6b6b8cd6-q4t6bm 1/5 CreateContaineErro 465 69d cal-rhqluqin-artfa 2/2 Running 0 69d cal-rhqluqin-artfa 2/2 Running 0 69d cal-rhqluqin-artfa 2/2 Running 0 69d rook-ceph-crashcollector-d-muscephwocker2-75cdf95dcd-1xg 1/1 Running 0 69d rook-ceph-crashcollector-d-muscephwocker2-75cdf95dcd-nxsz 1/1 Terninating 0 69d rook-ceph-archaeloclector-d-muscephwocker2-75cdf95dcd-nxsz 1/1 Running 0 69d rook-ceph-arg-a-dxb65dff-tpp7z 2/3 CrashcopBackff 16 (28d ag) 30d rook-ceph-arg-a-dxb65dff-thvap9 2/3 Terninating 0 401 rook-ceph-arg-a-dxb65dff-thvap9 2/3 Terninating 10 40 rook-ceph-arg-b-b7bdfd8c8b-jdg9 2/3 Terninating 10 41 rook-ceph-arg-b-b7bdfd8c8b-jdg9 2/3 <td< td=""><td>csi-rbdplugin-pk57s</td><td>2/2</td><td>Running</td><td>1 (63d ago)</td><td>63d</td></td<>	csi-rbdplugin-pk57s	2/2	Running	1 (63d ago)	63d
cal - hqlujun-provisioner - 6f8b68dc6-qw6bm 1/5 CreateContaineTETro 4455 69d cal - hqlujun-artfa 2/2 Running 0 69d cal - hqlujun-wfqqm 1/2 Running 0 69d coh-ceph-crashcollector-duuscephworker2-75cdf55dcd-lykd 1/1 Running 0 64d rook-ceph-crashcollector-duuscephworker2-75cdf55dcd-lykd 1/1 Running 0 64d rook-ceph-crashcollector-duuscephworker2-75cdf55dcd-lykd 1/1 Running 2 68d rook-ceph-arahcollector-duuscephworker2-75cdf55dcd-nykd 1/1 Running 2 68d rook-ceph-arahcollector-duuscephworker2-75cdf55dcd-nykd 1/1 Running 2 68d rook-ceph-ara-a-du5b8dff-fpg7s 2/2 CrashcopBackoff 146 (28d apc) 30d rook-ceph-ag-a-a-du5b8dff-fuysp³ 2/3 Terslinating 315 (30d ago) 69d rook-ceph-ag-b-Drbfd8deb-hdsp 2/2 Running 150 (31d ago) 69d rook-ceph-ag-c-6fddcbd6dc-b23g 2/2 Running 150 (31d ago) 69d	csi-rbdplugin-provisioner-6f6b6b8cd6-4c8jd	5/5	Terminating	2919 (29d ago)	69d
caihcpluyin-artf 2/2 Running 0 69d caihcpluyin-v6gm 2/2 Running 0 69d caihcpluyin-v6gm 2/2 Running 0 68d cob-ceph-crashcollector-dnumcephworker2-75cdf95dcd-1xgd 1/1 Pending 0 68d rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-nxsz 1/1 Terninating 0 69d rook-ceph-archaelollector-dnumcephworker2-75cdf95dcd-nxsz 1/1 Terninating 0 69d rook-ceph-archaelollector-dnumcephworker2-75cdf95dcd-x45xz 1/1 Terninating 0 69d rook-ceph-archaelollector-dnumcephworker3-6fdddcd9-x45xz 1/1 Terninating 10 69d rook-ceph-arc-a-dsb8dff-f-vpp7z 2/3 CrashcopBackoff 16 (28d ag) 30d rook-ceph-arc-b-Tbdfd8c8b-Jdg5p 2/3 Terninating 0 4d10 rook-ceph-arc-b-Tbdfd8c8b-Jdg5p 2/3 Running 18 (28d ag) 58d rook-ceph-arcc-fdcdcddc-b2jgc 2/2 Running 1500 (31d ag) 69d rook-ceph-arcfdf7fdd4-6	csi-rbdplugin-provisioner-6f6b6b8cd6-d4t56	0/5	Pending	0	4d10h
cal - help luglan-wiggm 2/2 Running 0 69d rook-eeph-crashcollector-dnumcephworker1-7845hb8ff-wa9fx 1/1 Running 0 68d rook-eeph-crashcollector-dnumcephworker2-75cdf95adcd-1kgd 0/1 Pending 0 4d10 rook-eeph-crashcollector-dnumcephworker2-75cdf95adcd-nkg 1/1 Terminating 0 69d rook-eeph-arcashcollector-dnumcephworker2-75cdf95adcd-nkg 1/1 Running 2 69d rook-eeph-arcashcollector-dnumcephworker2-75cdf95adcd-nkg 1/1 Running 2 69d rook-eeph-ag-a-cdb85adff-fpg7s 2/2 CrashLoopBackOff 146 (28d apc) 30d rook-eeph-ag-a-cdb85adff-frivapg 2/3 Terminating 3115 (30d agc) 69d rook-eeph-ag-bDrbdf8debb-sh4ww 2/3 Running 1500 (31d agc) 69d rook-eeph-agCdb5adff-deb-dkg 2/2 Running 1500 (31d agc) 69d rook-eeph-agfd6debcdeb-w1g 2/2 Running 1500 (31d agc) 69d rook-eeph-ag	csi-rbdplugin-provisioner-6f6b6b8cd6-gw6bm	1/5	CreateContainerError	4465	69d
rook-ceph-crashcollector-dnumcephworker1-7845b8ff-vs9fx 1/1 Running 0 68d rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-1jkqd 0/1 Pending 0 4d10l rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-1jkqd 1/1 Terminating 0 69d rook-ceph-agr-a-cdsb65dff-ft-pp7a 2/3 CrashLoopBackOff 146 (28d ago) 30d rook-ceph-agr-a-cdsb65dff-ft-pp7a 2/3 Terminating 3115 (30d ago) 69d rook-ceph-agr-a-cdsb65dff-ft-pp7a 2/3 Pending 3115 (30d ago) 69d rook-ceph-agr-a-cdsb65dff-bd9dp 0/3 Pending 283 (28d ago) 8d rook-ceph-agr-b-7bbfd8e8b-bd4ww 2/2 Running 1500 (31d ago) 69d rook-ceph-agr-b-7bdd8e8b-bd-bdg 2/2 Running 1500 (31d ago) 69d rook-ceph-agr-b-7bdd8e8b-bd-bdg 2/2 Running 1800 (32d ago) 69d rook-ceph-agr-b-7bdd8e8b-bd-wh4w 2/2 Running 1800 (32d ago) 69d rook-ceph-agr-b-7bdd8e8b-bd-wh4w 2/2 Running 1800 (32d ago) <t< td=""><td>csi-rbdplugin-srtfz</td><td>2/2</td><td>Running</td><td>0</td><td>69d</td></t<>	csi-rbdplugin-srtfz	2/2	Running	0	69d
rook-ceph-crashcol lector-deumcephvorker2-75cdf55dcd-1½cd 0/1 Pending 0 4410 rook-ceph-crashcol lector-deumcephvorker3-6fdb6cd9-x45bs 1/1 Running 2 68d rook-ceph-mgra-a-c5db58dff-lysp* 2/3 Terminating 3115 (30d ago) 69d rook-ceph-mgra-a-c5db58dff-lysp* 2/3 Terminating 3115 (30d ago) 69d rook-ceph-mgra-a-c5db58dff-lysp* 2/3 Terminating 3115 (30d ago) 69d rook-ceph-mgra-b-7bbfd8cBb-Jdqfp 2/3 Running 288 (28d ago) 58d rook-ceph-more-6d6cbed8cb-d2fgc 2/2 Running 1500 (31d ago) 69d rook-ceph-more-6d6cbed8cb-2fgc 2/2 Running 1500 (31d ago) 69d rook-ceph-more-6d6cbed8cb-2fgc 1/2 Running 1500 (31d ago) 69d rook-ceph-more-6d6cbed8cb-2fgc 1/2 Running 1 68d rook-ceph-more-6d7cbed8cb-4cdbr 1/2 Running 1 68d rook-ceph-more-6-57d9b8dbd-4-wtjp 1/2 Running 116 (30d ago) 68d rook-c	csi-rbdplugin-v6gqm	2/2	Running	0	69d
rook-eeph-crashoollector-dnumcephworker2-75cdf95dcd-n5xz 1/1 Terminating 0 69d rook-eeph-crashoollector-dnumcephworker3-6fddb6cd9-x45w3 1/1 Running 2 68d rook-eeph-mgr-a-c5db58dff-f-tpp7z 2/3 CrashicopBackoff 115 (30d ago) 69d rook-eeph-mgr-a-c5db58dff-thvap9 2/3 Pending 0 40t rook-eeph-mgr-b-Tbbfd8c8b-jdg5p 2/3 Pending 282 (38d ago) 58d rook-eeph-mgr-b-Tbbfd8c8b-jdg5p 2/2 Running 1500 (31d ago) 69d rook-eeph-more-64dbc46bc-b23g 2/2 Running 1500 (31d ago) 69d rook-eeph-more-64dbc46bc-w82g 2/2 Running 0 68d rook-eeph-od-05-7db8dbd4d-dsdhr 1/2 Terninating 31 (28d ago) 68d rook-eeph-od-05-7db8dbd4d-wstjp 1/2 Running 10 (30d ago) 68d rook-eeph-od-05-7db8dbd4d-wstjp 1/2 Running 117 (30d ago) 68d rook-eeph-od-05-7db8dbd4d-wstjp 1/2 Running 117 (30d ago) 68d rook-eeph-od-	rook-ceph-crashcollector-dnumcephworker1-7845bb8ff-vs9fx	1/1	Running	0	68d
rook-eeph-crashcollector-dnuncephworker3-6fdubécd9-x4595 1/1 Running 2 68 68 rook-eeph-mgr-a-c-dubbádff-fhvpp 2/3 CrashLoopBackOff 146 284 ago 90 rook-eeph-mgr-a-c-dubbádff-fhvpp 2/3 Terninating 3115 304 ago 90 rook-eeph-mgr-b-Thbfda8c8b-Jdgp 0/3 Preding 2283 (284 ago) 50 rook-eeph-mgr-b-Tbbfd8c8b-h4wu 2/3 Running 150 (314 ago) 69 rook-eeph-mgr-b-Tbbfd8c8b-h4wu 2/2 Running 150 (314 ago) 69 rook-eeph-mgr-b-Tbbfd8c8b-h4wu 2/2 Running 150 (314 ago) 69 rook-eeph-mgr-b-G4dcb4c6bcr-wBag 2/2 Running 0 68 68 rook-eeph-mgr-b-57db8dbd4-whtj 1/2 Fending 0 400 684 rook-eeph-mgr-d-1746982ff7d-62mh 1/2 Running 112 (34 ago) 68 rook-eeph-mgr-d-2-5ccd866f7c-1hm3 1/2 Running 117 (38 ago) 68 rook-eeph-	rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-1jkqd	0/1	Pending	0	4d10h
rook-ceph-mgr-a-c5db58dfr-fpp7r 2/3 CrashLoopBackOff 146 (28d ago) 30d rook-ceph-mgr-a-c5db58dfr-lvap9 2/3 Terminating 3115 (30d ago) 69d rook-ceph-mgr-b-7bbfd8eBb-Jdgp 0/3 Pending 0 4d10 rook-ceph-mora-b-75df9cdb-b-12gc 2/2 Running 1500 (31d ago) 69d rook-ceph-mora-375df9cddb-b4g6c-w28ag 2/2 Running 1608 (28d ago) 69d rook-ceph-mora-ce-64db-k266c-w28ag 1/1 Running 0 68d rook-ceph-od-0-57db9dbd-d-d6dhr 1/2 Pendinating 731 (28d ago) 68d rook-ceph-od-0-57db9dbd-d-d6dhr 1/2 Pending 71 71 71 72	rook-ceph-crashcollector-dnumcephworker2-75cdf95dcd-n5xsz	1/1	Terminating	0	69d
rook-ceph-mgr-a-c-dub5ddf-fnVapp 2/3 Terminating 3115 (30d ago) 69d rook-ceph-mgr-b-Thbfd8cBb-hdaye 0/3 Pending 0 40t rook-ceph-mgr-b-Thbfd8cBb-hdaye 2/2 Terminating 2283 (28d ago) 58d rook-ceph-mon-a-75cf9ccddc-b2jc 2/2 Running 1500 (31 dago) 69d rook-ceph-mon-a-64dcbdc8c-wa8g 2/2 Running 100 (32d ago) 68d rook-ceph-os-0-57db9bddv4-dsdhr 1/2 Terminating 71 (28d ago) 68d rook-ceph-osd-0-57db9bddv4-wrij 1/2 Pending 0 4d10 rook-ceph-osd-0-57db9bddv4-wrij 1/2 Running 116 (30d ago) 68d rook-ceph-osd-0-57db9bddv4-wrij 1/2 Running 116 (30d ago) 68d rook-ceph-osd-2-5ccd8667c-lmaf 1/2 Running 117 (28d ago) 68d rook-ceph-osd-2-5ccd8667c-lmaf 0/1 Completed 0 57d rook-ceph-osd-preparae-dnuncephvorker3-42rxv 0/1 Completed 0 57d rook-ceph-osl-osl-7c4b8bb95r2-betr <	rook-ceph-crashcollector-dnumcephworker3-6fddb6cd9-x45w5	1/1	Running	2	68d
rook-ceph-mgr-b-Tbbfd88cBb-Jqdp 0/3 Pending 0 4d10* rook-ceph-mgr-b-Tbbfd8cBb-Jqdp 2/3 Treminating 228 (28d ago) 58d rook-ceph-mgr-b-Tbbfd8cBb-Jqdp 2/2 Running 1500 (31d ago) 69d rook-ceph-mon-a-75cf9cddb-26f6c-w28ag 2/2 Running 1800 (28d ago) 69d rook-ceph-more-c-6dcbc46f6c-w28ag 1/1 Running 731 (28d ago) 68d rook-ceph-od-0-57d9b8db4d-4dcMr 1/2 Pending 731 (28d ago) 68d rook-ceph-od-0-57d9b8db4d-4dcMr 1/2 Pending 716 (30d ago) 68d rook-ceph-od-1-7469b8d77fd-6n2mh 1/2 Running 716 (30d ago) 68d rook-ceph-od-2-5ccd86467c-lm47 1/2 Running 712 (28d ago) 68d rook-ceph-od-p-repare-dnuncephvorker1-rnk78 0/1 Completed 0 57d rook-ceph-od-p-repare-dnuncephvorker3-42rxv 0/1 Completed 0 57d rook-ceph-od-1-7-64b8bb3-8-tfsr 0/1 Pending 0 400	rook-ceph-mgr-a-c5db58dff-fpp7z	2/3	CrashLoopBackOff	146 (28d ago)	30d
rook-ceph-megr-b-Tubtd8e8b-wh4ww 2/3 Terminating 2283 (28d ago) 8d rook-ceph-mon-a-75cfgcddch-bdjgc 2/2 Running 150 (31d ago) 6d rook-ceph-mon-c-64dcb4e86c-wz8sg 2/2 Running 1808 (28d ago) 69d rook-ceph-oc-0-57db8ddw4-dsdhr 1/2 Terminating 71 (28d ago) 68d rook-ceph-od-0-57db8ddw4-wntjp 0/2 Pending 71 (30d ago) 68d rook-ceph-od-1-746982f74-6nzh 1/2 Running 716 (30d ago) 68d rook-ceph-od-2-5ccd866f7c-lmsf 1/2 Running 117 (28d ago) 68d rook-ceph-od-2-5ccd866f7c-lmsf 0/1 Completed 0 57d rook-ceph-od-prepare-dnuncephvorker3-42rw 0/1 Completed 0 57d rook-ceph-od-1-7-6d88bb58-8cf8r 0/1 Pending 0 400	rook-ceph-mgr-a-c5db58dff-hvsp9	2/3	Terminating	3115 (30d ago)	69d
rook-ceph-mon-a-7st/9ccddc-b2/gc 2/2 Running 1500 (31d ago) 69d rook-ceph-mon-c-64dcb486c-w288g 2/2 Running 1808 (28d ago) 69d rook-ceph-oser-c-64f7df4d-61mfp 1/1 Running 731 (28d ago) 68d rook-ceph-od-0-57d9b8db4d-d6dhr 1/2 Pending 731 (28d ago) 68d rook-ceph-od-0-57d9b8db4d-wttp 0/2 Pending 716 (30d ago) 68d rook-ceph-od-2-57d9b8db4d-wttp 1/2 Running 716 (30d ago) 68d rook-ceph-od-2-5cc486d5f-Clm4r 1/2 Running 1172 (28d ago) 68d rook-ceph-od-prepare-dnumcephworker1-rnk7s 0/1 Completed 0 57d rook-ceph-od-prepare-dnumcephworker3-42xw 0/1 Completed 0 57d rook-ceph-od-10-7cdb8bb3b5-8cffr 0/1 Pending 0 40d	rook-ceph-mgr-b-7bbfd88c8b-jdg4p	0/3	Pending	0	4d10h
rook-ceph-mon-o-64dabde6c-wz8sg 2/2 Running 1808 (28d ago) 69d rook-ceph-poerator-cftf7df3d4-6tmfp 1/1 Running 0 68d rook-ceph-osd-0-57db8db4d-4ddhr 1/2 Terminating 731 (28d ago) 68d rook-ceph-osd-0-57db8db4d-wmt.fp 0/2 Pending 0 4d10 rook-ceph-osd-0-57db8db4d-wmt.fp 1/2 Running 716 (30d ago) 68d rook-ceph-osd-2-5ccd86467c-lhms7 1/2 Running 1172 (28d ago) 68d rook-ceph-osd-2-5ccd86467c-lhms7 0/1 Completed 0 57d rook-ceph-osd-prepare-dnuncephvorker3-42rxv 0/1 Completed 0 57d rook-ceph-osd-prepare-dnuncephvorker3-42rxv 0/1 Cephected 0 57d rook-ceph-osd-prepare-dnuncephvorker3-42rxv 0/1 Pending 0 400	rook-ceph-mgr-b-7bbfd88c8b-wh4ww	2/3	Terminating	2283 (28d ago)	58d
rook-ceph-operator-cf47df44-6tm6p 1/1 Running 0 68d rook-ceph-oad-0-57d9b8db44-d6dhr 1/2 Terminating 731 (28d ago) 68d rook-ceph-oad-0-57d9b8db44-wttp 0/2 Pending 0 4d10 rook-ceph-oad-1-7469b8777d-6n2mh 1/2 Running 716 (30d ago) 68d rook-ceph-oad-2-5cc468647-1m47 1/2 Running 1172 (28d ago) 68d rook-ceph-oad-prepare-dnumcephvorkeri-rnk78 0/1 Completed 0 57d rook-ceph-oad-prepare-dnumcephvorker3-42xw 0/1 Completed 0 57d rook-ceph-oad-7-6ab8bb58-bc.ffr 0/1 Pending 0 4d10	rook-ceph-mon-a-75cf9ccddc-b2jgc	2/2	Running	1500 (31d ago)	69d
rook-ceph-osd-0-57db8db4d-4ddhr 1/2 Terminating 731 (28d ago) 68d rook-ceph-osd-0-57db8db4d-vmt.jp 0/2 Pending 0 4d10 rook-ceph-osd-1-746982T7f4-652mh 1/2 Running 716 (30d ago) 68d rook-ceph-osd-2-5ccd86467c-lhm47 1/2 Running 1172 (28d ago) 68d rook-ceph-osd-prepare-dnuncephworker1-rnk78 0/1 Completed 0 57d rook-ceph-osd-prepare-dnuncephworker3-42rxv 0/1 Completed 0 57d rook-ceph-osd-prepare-dnuncephworker3-42rxv 0/1 Pending 0 4d10	rook-ceph-mon-c-64dcb4c86c-wz8sg	2/2	Running	1808 (28d ago)	69d
rook-ceph-osd-0-57d9b8db4d-vmtjp 0/2 Pending 0 4d10t rook-ceph-osd-1-7469b8d77fd-6n2mh 1/2 Running 716 (30d ago) 68d rook-ceph-osd-2-5cc4846f7-lmaf7 1/2 Running 1172 (28d ago) 68d rook-ceph-osd-prepare-dnumcephvorkerl-rnk78 0/1 Completed 0 57d rook-ceph-osd-prepare-dnumcephvorker3-42rv 0/1 Completed 0 57d rook-ceph-colo-7-cobsbb95b-8-tfsr 0/1 Pending 0 4d10t	rook-ceph-operator-cf4f7dfd4-6tm6p	1/1	Running	0	68d
rook-ceph-oad-1-746987776-6a2mh 1/2 Running 716 (30d ago) 68d rook-ceph-oad-2-5cc486467c-lhm47 1/2 Running 1172 (28d ago) 68d rook-ceph-oad-prepare-dnumcephworker1-rnk78 0/1 Completed 0 57d rook-ceph-oad-prepare-dnumcephworker3-42xw 0/1 Completed 0 57d rook-ceph-to-01-7-C48bbb396-8-tfsr 0/1 Pending 0 4d10	rook-ceph-osd-0-57d9b8db4d-d6dhr	1/2	Terminating	731 (28d ago)	68d
rook-ceph-osd-2-5cc486467c-lnm47 1/2 Running 1172 (28d ago) 68d rook-ceph-osd-prepare-dnumcephworker1-rnk78 0/1 Completed 0 57d rook-ceph-osd-prepare-dnumcephworker3-42rv 0/1 Completed 0 57d rook-ceph-cools-7c4b8bb95-8tf8r 0/1 Pending 0 4dl0	rook-ceph-osd-0-57d9b8db4d-vmtjp	0/2	Pending	0	4d10h
rook-ceph-osd-prepare-dnumcephworkeri-rnk78 0/1 Completed 0 57d rook-ceph-osd-prepare-dnumcephworker3-42rwv 0/1 Completed 0 57d rook-ceph-tool-7-db8bbb5-8-ffr 0/1 Pending 0 4d100	rook-ceph-osd-1-74698f77fd-6n2mh	1/2	Running	716 (30d ago)	68d
rook-ceph-osd-prepare-dnumcephworker3-42rxv 0/1 Completed 0 57d rook-ceph-tools-7c4b8bb9b5-8tf8r 0/1 Pending 0 4d10h	rook-ceph-osd-2-5cc486467c-1hm47	1/2	Running	1172 (28d ago)	68d
rook-ceph-tools-7c4b8bb9b5-8tf8r	rook-ceph-osd-prepare-dnumcephworker1-rnk78	0/1	Completed	0	57d
	rook-ceph-osd-prepare-dnumcephworker3-42rxv	0/1	Completed	0	57d
rook-ceph-tools-7c4b8bb9b5-pxk67 1/1 Terminating 0 68d	rook-ceph-tools-7c4b8bb9b5-8tf8r	0/1	Pending	0	4d10h
	rook-ceph-tools-7c4b8bb9b5-pxk67	1/1	Terminating	0	68d

Le contrôleur d'admission (Admission Controller) - Rook Ceph

- Il est recommandé de déployer le contrôleur d'admission : il permet de vérifier que Rook est correctement paramétré grâce aux réglages des Customer Resources (CR)
- L'Admission Controller intercepte les requêtes à destination de l'API k8s avant l'objet persistant après les phases d'authentification et d'autorisation
- Pour installer l'Admission Controller, lancer les requêtes suivantes : kubectl apply -f https://github.com/jetstack/cert-manager/releases/download/v1.7.1/cert-manager.yaml

Affichage des storage class déployés

Le storageclass déployé a pour nom rook-ceph-block.

linagora@debian-cp:~\$ kubectl get storageclass						
NAME	PROVISIONER	RECLAIMPOLICY	VOLUMEBINDINGMODE	ALLOWVOLUMEEXPANSION	AGE	
local-storage	kubernetes.io/no-provisioner	Delete	WaitForFirstConsumer	false	12d	
rook-ceph-block	rook-ceph.rbd.csi.ceph.com	Delete	Immediate	true	5d23h	

Sélection des nœuds PostgreSQL

De manière similaire à l'opérateur Rook Ceph, il est possible de sélectionner les nœuds portant le pod PostgreSQL en se basant sur les labels Kubernetes.

Marquage des nœuds PostgreSQL

<none>

<none>

Ready

Readv

dnumworker1

dnumworker2

Les commandes ci-dessous marquent les nœuds destinés à porter les pods PostgreSQL :

kubernetes.io/hostname=dnumworker1, kubernetes.io/os=linux, postgres-operator=enabled kubernetes.io/hostname=dnumworker2, kubernetes.io/os=linux, postgres-operator=enabled

Répartitions des pods PostgreSQL sur les nœuds worker et choix du storageClass

```
S git diff
diff --git a/manifests/complete-postgres-manifest.vaml b/manifests/complete-postgres-manifest.vaml
index 8d197a75..56b32c34 100644
--- a/manifests/complete-postgres-manifest.vaml
+++ b/manifests/complete-postgres-manifest.vaml
88 -57,7 +57,7 88 spec:
   volume:
     size: 1Gi
    storageClass: mv-sc
     storageClass: rook-ceph-block
     iops: 1000 # for EBS gp3
     throughput: 250 # in MB/s for EBS gp3
     selector.
00 -203,14 +203,14 00 spec:
 # Add node affinity support by allowing postgres pods to schedule only on nodes that
 # have label: "postgres-operator:enabled" set.
-# nodeAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
        nodeSelectorTerms:
          - matchExpressions:
              - kev: postgres-operator
                operator: In
                values.
+ nodeAffinity:
     requiredDuringSchedulingIgnoredDuringExecution:
       nodeSelectorTerms:
         - matchExpressions:
             - key: postgres-operator
               operator: In
               values:
                - enabled
 # Enables change data capture streams for defined database tables
 # streams:
```

Déploiement de l'opérateur PostgreSQL de Zalando

Le storage class est maintenant déployé.

Il devient possible d'appliquer l'opérateur PostgreSQL.

Le lien suivant décrit les commandes à appliquer :

Clonage du dépôt de l'opérateur

git clone $\label{local_postgres-operator} \begin{subarray}{ll} \tt git clone https://github.com/zalando/postgres-operator.git cd postgres-operator \end{subarray}$

Application des différents manifestes

```
kubectl create -f manifests/configmap.yaml # configuration
kubectl create -f manifests/operator-service-account-rbac.yaml # identity and permissions
kubectl create -f manifests/postgres-operator.yaml # deployment
kubectl create -f manifests/api-service.yaml # operator API to be used by UI
```

Pour information, il existe également des chart Helm pour facilier le déploiement.

Accès à l'interface web

Pour activer l'accès à l'interface web de l'opérateur PostgreSQL, veuillez la commande suivante sur le nœud control plane :

```
$ kubectl port-forward svc/postgres-operator-ui 8081:80 Forwarding from 127.0.0.1:8081 -> 8081 Forwarding from [::1]:8081 -> 8081
```

Elle redirige le flux TCP du port 80 du control plane vers le port TCP 8081 du service postgres-operator-ui

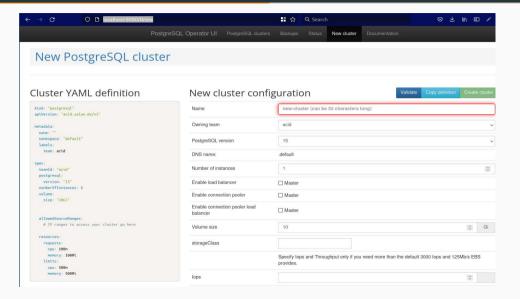
Accès à l'interface web

Pour accéder à l'interface web de l'opérateur PostgreSQL depuis le PC de l'utilisateur, il est possible de passer par une redirection SSH :

```
ssh -L 9090:10.106.57.137:80 dgfip-k8s
```

Lancer le navigateur pour accéder à l'URL http://localhost:9090/#new

Interface web de l'opérateur PostgreSQL



Fonctionnalités proposées par l'interface web de l'opérateur PostgreSQL

L'UI permet de :

- choisir la version PostgreSQL (jusquà la version 15 actuellement)
- le nombre d'instances
- activation du load-balancer
- activation du pool de connexions à la base
- activation du load-balancer pour le pool de connexions à la base
- taille du volume persistent alloué à la base de données
- choix du storageClass
- performances IO
- choix des ressources (demandées et limites) CPU et RAM allouées

Utilisation de la commande en ligne pour la création d'un cluster PostgreSQL

- Les fonctionnalités proposées par l'Ul sont également disponibles par la commande en ligne.
- Le manifeste manifests/complete-postgresql-manifest.yaml permet de préciser l'ensemble des paramètres proposés par l'UI.
- Pour appliquer ce manifeste *manifests/complete-postgresql-manifest.yaml*, la commande suivante est lancée sur le nœud :

kubectl create -f manifests/complete-postgresql-manifest.yaml

Vérification de l'état du cluster PostgreSQL

```
$ kubectl get pods -l application=spilo -L spilo-role
NAME
                      READY
                              STATUS
                                        RESTARTS
                                                   AGE
                                                         SPILO-ROLE
acid-test-cluster-0
                              Running
                                                   12m
acid-test-cluster-1 1/1
                              Running
                                                   10m
$ kubectl get postgresgl
NAME
                    TEAM
                           VERSION
                                     PODS
                                            VOLUME
                                                     CPU-REOUEST
                                                                   MEMORY-REQUEST
                                                                                          STATUS
                                                                                     AGE
acid-test-cluster
                    acid
                          15
                                                     10m
                                                                   100Mi
                                                                                     68d
                                                                                          Running
$ kubectl get pods -l application=spilo -L spilo-role
NAME
                      READY
                              STATUS
                                        RESTARTS
                                                   ACE
                                                         SPILO-ROLE
acid-test-cluster-0 1/1
                              Running
                                        Λ
                                                   1 3 m
acid-test-cluster-1 1/1
                              Running
                                                   1 0 m
$ kubectl get svc -l application=spilo -L spilo-role
NAME
                           TYPE
                                       CLUSTER-IP
                                                       EXTERNAL-IP
                                                                     PORT(S)
                                                                                       SPILO-ROLE
                                                                                 ACE
acid-test-cluster
                                       10.103.247.68
                           ClusterIP
                                                                     5432/TCP
                                                                                68d
                                                                                      master
                                                       <none>
acid-test-cluster-config
                           ClusterIP
                                                                                68d
                                       None
                                                       <none>
                                                                     <none>
acid-test-cluster-repl
                           ClusterIP
                                       10.100.95.205
                                                       <none>
                                                                     5432/TCP
                                                                                 68d
                                                                                       replica
```

Déploiement de krew

```
set -x; cd "$(mktemp -d)" &
OS="$(uname | tr '[:upper:]' '[:lower:]')" &
ARCH="$(uname -m | sed -e 's/x86_64/amd64/' -e 's/\(arm\)\(64\)\?.*/\1\2/' -e 's/aarch64$/arm64/')" &
KREW="krew-${OS}_${ARCH}" &
curl -fsSLO "https://github.com/kubernetes-sigs/krew/releases/latest/download/${KREW}.tar.gz" &
tar zxvf "${KREW}.tar.gz" &
./"${KREW}" install krew
}
```

Déploiement de krew

Ajout du répertoire des binaires du paquet krew dans .bashrc ou .zshrc :

export PATH="\${KREW_ROOT:-\$HOME/.krew}/bin:\$PATH"

Redémarrer le shell.

Pour vérifier le déploiement correct de krew, lancer la commande suivante :

kubectl krew

Déploiement de l'opérateur Minio

Le déploiement de Minio s'appuie sur l'opérateur Minio. Son déploiement est décrit dans le lien suivant https://operator.min.io/#architecture.

kubectl krew update
kubectl krew install minio

Vérification de l'état de l'opérateur Minio

\$ Kubecti get pods -n minio-opera	tor			
NAME	READY	STATUS	RESTARTS	AGE
console-56f9795d5c-59fsx	1/1	Running	1 (33d ago)	820
minio-operator-7cd6784f59-5c52w	1/1	Running	5 (23h ago)	17c
minio-operator-7cd6784f59-m7h8x	1/1	Running	2320 (3d16h ago)	82d

Accès à la console Minio

La commande suivante ouvre un accès de type proxy à la console Minio :

```
$ kubectl minio proxy -n minio-operator
Starting port forward of the Console UI.
To connect open a browser and go to http://localhost:9090
Current JWT to login: *****
Forwarding from 0.0.0.0:9090 -> 9090
Handling connection for 9090
```

Depuis le terminal de l'utilisateur, lancer la commande suivante :

```
$ ssh -L 9090:localhost:9090 dgfip-k8s
```

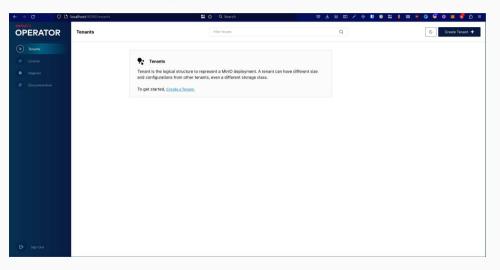
Accès à la console Minio

Dans le champ *Enter JWT*, renseigner la valeur du token JWT renvoyé par la commande précédente :

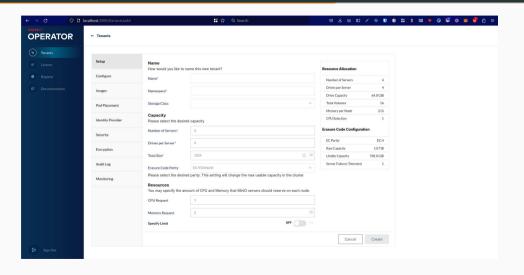


Dashboard Minio

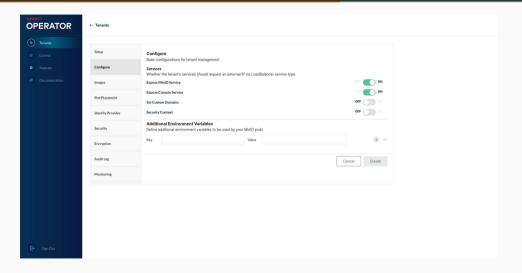
Le tableau de bord de Minio ressemble à ceci :



Création d'un tenant - Setup - Minio



Création d'un tenant - Configure - Minio



Marquage des nœuds Minio

Depuis le control plane, lancer les commandes suivantes pour marquer les nœuds :

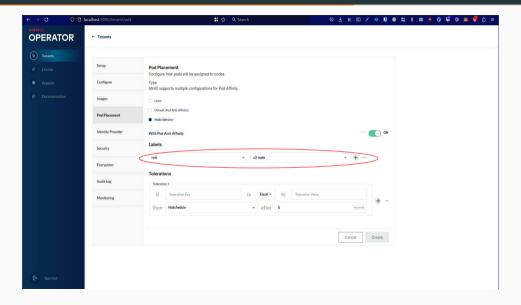
```
$ kubect1 label nodes dnumminioworker1 role=s3-node
node/dnumminioworker1 labeled
$ kubect1 label nodes dnumminioworker2 role=s3-node
node/dnumminioworker2 labeled
$ kubect1 label nodes dnumminioworker3 role=s3-node
node/dnumminioworker3 labeled
$ kubect1 label nodes dnumminioworker4 role=s3-node
node/dnumminioworker4 labeled
```

Affichage du label des nœuds :

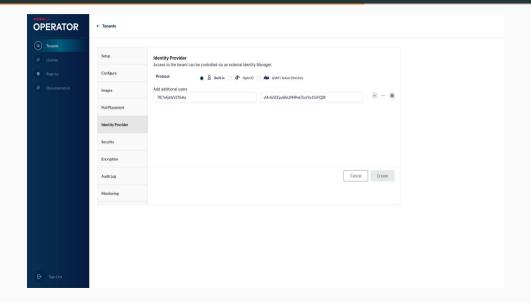
```
$ kubectl get nodes --show-labels
NAME
                   STATUS
                             ROLES
                                             AGE
                                                     VERSION
                                                               LABELS
dnumminioworker1
                   Readv
                                                    171 27 2
                                                                ....role=s3-node
                             <none>
                                              6h2m
dnumminioworker2
                                                    v1.27.2
                                                                ...role=s3-node
                   Readv
                             <none>
                                              142m
dnumminioworker3
                   Readv
                             <none>
                                              109m
                                                     v1.27.2
                                                                ....role=s3-node
dnumminioworker4
                   Readv
                                              83m
                                                     171 27 2
                                                                ... role=s3-node
                             < none>
```

67

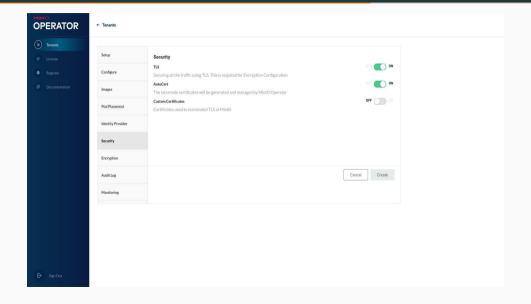
Création d'un tenant - Pod placement - Minio



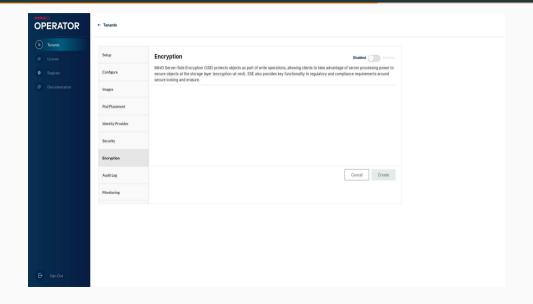
Création d'un tenant - Choix d'un fournisseur d'identité - Minio



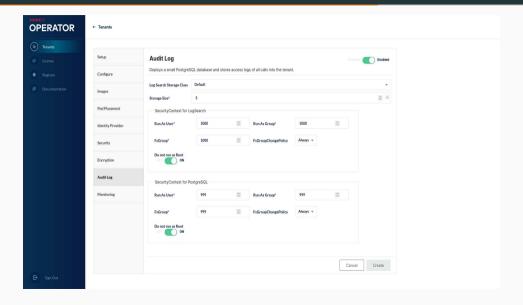
Création d'un tenant - Sécurité - Minio



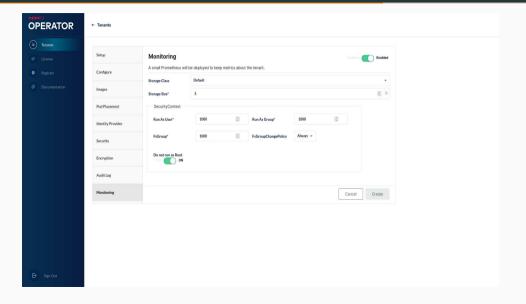
Création d'un tenant - Chiffrement - Minio



Création d'un tenant - Log d'audit - Minio



Création d'un tenant - Supervision - Minio



Clé d'accès et secret - Minio

Image Docker utilisée par l'opérateur PostgreSQL

L'image déployée par l'opérateur PostgreSQL de Zalando s'appuie sur Spilo https://github.com/zalando/spilo

Cette information se trouve dans le script manifests/complete-postgres-manifest.yaml :

dockerImage: ghcr.io/zalando/spilo-15:3.0-p1

Utilisation de container en mode rootless dans OpenShift

L'URL suivante décrit la problèmatique d'utilisation de container dans un environnement rootless :

https://docs.bitnami.com/tutorials/running-non-root-containers-on-openshift

Cette page fournit également un lien intéressant sur la sécurisation d'un cluster Kubernetes en se basant sur les **Pod Security Policies**.

Le lien est le suivant :

 $\verb|https://docs.bitnami.com/tutorials/secure-kubernetes-cluster-psp/.\\$

Pod Security Policies

Comme indiqué dans

https://kubernetes.io/docs/concepts/security/pod-security-policy/les PSP sont dépréciés.

Ils sont maintenant remplacés par Pod Security Admission

https://kubernetes.io/docs/concepts/security/pod-security-admissi

La norme

Pod Security Admission définit la notion de Security Context décrite dans le lien suivant :

https://kubernetes.io/docs/tasks/configure-pod-container/security

C'est cette notion qui va permet d'approcher au plus près les condictions de run d'un cluster Openshift

Opérateur - Security Context

Par défaut, l'opérateur PostgreSQL de Zalando applique les mesures de sécurité suivantes.

Extrait de manifests/postgres-operator.yaml :

```
securityContext:
    runAsUser: 1000
    runAsNonRoot: true
    readOnlyRootFilesystem: true
    allowPrivilegeEscalation: false
```

- Il ne tourne pas avec un identifiant privilégié ni le compte root
- Il s'appuie sur des filesystems en lecture seule

Pods PostgreSQL - Security Context

Il est possible d'affecter :

- un utilisateur non privilégié
- un groupe non privilégié au pod.
- un groupe de filesystem défini

${\sf Extrait\ de\ manifests/complete-postgres-manifest.yaml:}$

```
spiloRunAsUser: 101
spiloRunAsGroup: 103
spiloFSGroup: 103
```

Installation d'ArgoCD

Le lien suivant décrit l'installation d'ArgoCD :

https://argo-cd.readthedocs.io/en/stable/#quick-start

kubectl create namespace argocd
kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml

Installation de la CLI ArgoCD

Le lien suivant décrit l'installation de la CLI ArgoCD :

https://argo-cd.readthedocs.io/en/stable/cli_installation/#downlc

 $\hbox{curl -sSL -o argocd-linux-amd64 https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64 sudo install -m 555 argocd-linux-amd64 /usr/local/bin/argocd \\ \hbox{rm argocd-linux-amd64}$

Activation de l'accès au serveur d'API d'ArgoCD

service/argood-server patched

Le lien suivant décrit les différentes méthodes d'accès au serveur d'API d'ArgoCD : https://argo-cd.readthedocs.io/en/stable/getting_started/#3-acces

```
$ kubectl patch svc argood-server -n argood -p '{"spec": {"type": "LoadBalancer"}}'
```

Build de l'image Docker Spilo

Le Dockerfile définissant l'image Spilo est disponible à l'URL suivante :

https://github.com/zalando/spilo

La méthode de génération de l'image est décrite dans :

https://github.com/zalando/spilo#how-to-build-this-docker-image

```
$:-/spilo/postgres-appliance$ docker build --tag dnum-test .
[+] Building 9762.1s (33/33) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 3.01kB
=> [internal] load .dockerignore
=> => transferring context: 2B
...
=> => exporting layers
=> => writing image sha256:52afd69aff9414333220ec408283a8ff20e2162e703c6ab5afd5090d9d62e4e0
=> => naming to docker.io/library/dnum-test
```

Le build dure un peu moins de **2h43min**.

Login avec la CLI d'ArgoCD

La commande suivante permet de récupérer le mot de passe initial de l'admin d'ArgoCD :

This password must be only used for first time login. We strongly recommend you update the password using 'argood accounts and account to the password strongly recommend you update the password using 'argood accounts and the password strongly recommend you update the password using 'argood accounts and the password strongly recommend you update the password using 'argood accounts and the password strongly recommend you update the password using 'argood accounts and the password using 'argond accounts and the password using 'argond accounts and the password using 'argond accounts and 'argond accounts are also account and 'argond accounts are also accounts and 'argond accounts are also accounts and 'argond accounts are also accounts and 'argond accounts are also account and 'argond accounts are also account and 'argond accounts are also accounts and 'argond accounts are also account and 'argond accounts are also accounts and 'argond accounts are also account and 'argond accounts are also accounts are also account and 'argond accoun

Login avec la CLI d'ArgoCD

<pre>\$ kubectl get svc -n argocd</pre>					
NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	Ac
argocd-applicationset-controller	ClusterIP	10.109.27.242	<none></none>	7000/TCP,8080/TCP	60
argocd-dex-server	ClusterIP	10.100.63.110	<none></none>	5556/TCP, 5557/TCP, 5558/TCP	60
argocd-metrics	ClusterIP	10.102.19.237	<none></none>	8082/TCP	60
argocd-notifications-controller-metrics	ClusterIP	10.102.70.229	<none></none>	9001/TCP	60
argocd-redis	ClusterIP	10.104.253.209	<none></none>	6379/TCP	60
argocd-repo-server	ClusterIP	10.111.28.234	<none></none>	8081/TCP,8084/TCP	60
argocd-server	LoadBalancer	10.109.47.144	<pending></pending>	80:30235/TCP,443:30885/TCP	60
argocd-server-metrics	ClusterIP	10.98.167.31	<none></none>	8083/TCP	60
34	0 47 144				

linagora@debian-cp:~\$ argood login 10.109.47.144

WARNING: server certificate had error: x509: cannot validate certificate for 10.109.47.144 because it doesn't contain a Username: admin

Password:

Context '10.109.47.144' updated

 $[\]verb|'admin:login'| logged in successfully \\$

Ajout de l'application postgres-operator dans ArgoCD

- ArgoCD se met à l'écoute des changements d'un projet de déploiement dans un dépôt.
- L'étape suivante "abonne" ArgoCD au dépôt Github sur lequel est enregistré le déploiement de l'opérateur postgres
- La commande qui lie ArgoCD au dépôt de l'opérateur est la suivante :

```
argocd app create postgres-operator \
--repo https://github.com/simonelbaz/postgres-operator.git \
--path manifests \
--dest-server https://kubernetes.default.svc \
--revision poc-argocd \
--dest-namespace default
```

- Cette commande est décrite en détail dans la documentation officielle : https://argo-cd.readthedocs.io/en/stable/user-guide/commands/a
- Pour information, le dépôt ci-dessus est un fork du dépôt officiel
- L'ensemble des modifications réalisées pour le POC sont tracées dans la branche poc-argocd

Ajout de l'application postgres-operator dans ArgoCD

• Pour vérifier que l'application a correctement été ajoutée dans ArgoCD :

```
$ argood app list

NAME CLUSTER NAMESPACE PROJECT STATUS HEALTH SYNCPOLICY COND.

argood/postgres-operator https://kubernetes.default.svc default default Unknown Healthy <none> Compa
```

Selon le lien suivant.

```
https://argo-cd.readthedocs.io/en/stable/user-guide/tool_deteclescript kustomization.yaml est automatiquement détecté par ArgoCD
```

Kustomize & ArgoCD

- ArgoCD est capable de s'interfacer avec Kustomize https://kustomize.io/.
- Zalando met à disposition un script Yaml manifests/kustomization.yaml

```
$ cat kustomization.yaml
apiVersion: kustomize.config.k8s.io/vlbetal
kind: Kustomization
resources:
- configmap.yaml
- operator-service-account-rbac.yaml
- postgres-operator.yaml
- ani-service.yaml
```

- Ce dernier facilite le déploiement de l'opérateur dans l'environnement k8s
- Une page synthétisant l'utilisation de Kustomize est disponible à l'URL suivante : https://kubectl.docs.kubernetes.io/guides/introduction/kustomi

Intégration CI/CD & ArgoCD

- Le lien suivant décrit l'intégration d'un outil de CI/CD et ArgoCD. https://argo-cd.readthedocs.io/en/stable/user-guide/ci_automat
- Globalement la mise à jour d'un cluster k8s par l'intermédiaire d'ArgoCD se déroule en 2 phases :
 - récupération du dépôt git
 - la phase de patch
 - la phase de sync

Phase de patch & ArgoCD

cd postgres-operator

■ La 1^{re} étape de la phase de patch est de récupérer le dépôt du projet :

git clone https://github.com/simonelbaz/postgres-operator.git

- Il devient possible de patcher avec la commande kustomize ou la commande kubectl
 - Patch avec kustomize (nécessite l'installation de kustomize) :

Phase de patch avec kustomize

- L'URL suivante décrit l'installation de kustomize : https://kubectl.docs.kubernetes.io/installation/kustomize/bina
 - Pour installer kustomize, merci de lancer la commande suivante : curl -s "https://raw.githubusercontent.com/kubernetes-sigs/kustomize/master/hack/install kustomize.sh" | bash
 - kustomize peut servir à l'édition de kustomization.yaml.
 - Dans le cadre du POC, cela n'a pas été nécessaire

Phase de patch avec kubectl

On liste dans un 1^{er} temps l'ensemble des ConfigMaps :

```
$ kubectl get configmaps

NAME DATA AGE
kube-root-ca.crt 1 98d
postgres-operator 59 97d
```

Pour lister la ConfigMap qui définit l'image spilo déployée :

```
$ kubectl get configmap postgres-operator -o yaml
apiVersion: v1
data:
    api_port: "8080"
...
    docker_image: ghcr.io/zalando/spilo-15:2.1-p9
...
kind: ConfigMap
metadata:
    creationTimestamp: "2023-03-16T22:20:28Z"
    name: postgres-operator
    namespace: default
...
```

Phase de patch avec kubectl

La version de l'image spilo est modifiée :

kubectl patch --local -f configmap.yaml -p '"data":"docker_image":"ghcr.io/zalando/spilo-15:3.0-p1"' -o yaml >
mv configmap.yaml.new configmap.yaml

Phase de sync & ArgoCD

 Une fois la modification réalisée, il est nécessaire de la pusher vers le dépôt sur lequel se synchronise ArgoCD

```
git add . -m "Mise à jour de l'image" git push
```

Lancer ensuite les commandes suivantes dans le pipeline de la CI :

```
export ARGOCD_SERVER=argocd.mycompany.com
export ARGOCD_AUTH_TOKEN=sJWT token generated from project>
curl -sSL -o /usr/local/bin/argocd https://$ARGOCD_SERVER/download/argocd-linux-amd64
argocd app sync nom application
argocd app wait nom application
```

ArgoCD - Etat du cluster avant le lancement du sync

argocd/postgres-operator

commit 90ad8c7aed92c6430bcb36ab23228a2d57c2715d
\$ argood app get argood/postgres-operator

default

rbac.authorization.k8s.io ClusterRoleBinding

\$ git log | head -1

Name: Project:

Server. https://kubernetes.default.svc Namespace: default URL: https://10.109.47.144/applications/postgres-operator Repo: https://github.com/simonelbaz/postgres-operator.git Target: poc-argood Path: manifests SyncWindow: Sync Allowed Sync Policy: <none> Sync Status: OutOfSync from poc-argocd (90ad8c7) Health Status: Healthy GROUP KIND NAMESPACE NAME STATHS HEALTH MESSAGE HOOK ConfigMap default postgres-operator Service default OutOfSvnc Healthy postgres-operator ServiceAccount default postgres-operator OutOfSvnc acid zalan do postgresgl default acid-test-cluster OutOfSync Deployment default postgres-operator OutOfSync Healthy apps rhac authorization.k8s.io ClusterRole postgres-operator OutOfSync rhac authorization kge in ClusterRole postgres-pod OutOfSync

postgres-operator

ArgoCD - Etat du cluster avant le lancement du sync

- L'information renvoyée par la commande argocd get est cohérente.
 - Il y a bien un décalage entre l'état du cluster et le dernier commit de la branche poc-argocd
- Vérification de la version de l'image spilo déployée :

```
$ kubectl describe pod acid-test-cluster-1 | grep spilo
Labels: application=spilo

Image: ghcr.io/zalando/spilo-15:2.1-p9

KUBERNETES_ROLE_LABEL: spilo-role

KUBERNETES_LABELS: "application": "spilo"
```

ArgoCD - Lancement de la synchronisation

```
NAME
2023-06-23T16:49:22+02:00 rbac.authorization.k8s.io ClusterRoleBinding
                                                    ConfigMap
                                                                          default
2023-06-23T16-49-22+02-00 anno
                                ServiceAccount default nostgres-operator Sunced
                                 ConfigMap default postgres-operator Synced
                                               de fault
2023-06-23T16:50:00+02:00 apps Deployment
figured, Warning; resource postgresgls/acid-test-cluster is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on
                                                    ConfidMan
                                                                                                                                          confirman/postgres-operator configured Warnin
g: resource configmaps/postgres-operator is missing the kubectl, kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resources creat
2023-06-23T16:50:01+02:00 rbac,authorization,k8s.io ClusterRole
                                                                                          postgres-pod Running Synced
-pod reconciled, reconciliation required update, clusterrole, rbac, authorization, k8s, io/postgres-pod configured, Marning; resource clusterroles/postgres-pod is missing the kubectl, kubernetes,
ic/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either create --save-config or apply. The missing ann
2023-06-23T16:50:01+02:00 rbac.authorization.k8s.io ClusterRole
                                                                          default postgres-operator Running Synced
-operator reconciled, reconciliation required update, clusterrole, that authorization, kRs. is/postgres-operator configured. Warning: resource clusterroles/postgres-operator is missing the kube
ctl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either create --save-config or apply.
2023-06-23T16:50:01+02:00 rhar authorization k8s in ClusterRoleRinding default postgres-operator Running Synced
ostgres-operator reconciled, reconciliation required update, clusterrolebinding.rbac.authorization.k8s.io/postgres-operator configured. Warning: resource clusterrolebindings/postgres-operator
r is missing the kubecti kubernates in/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either create --sa
ve-config or apply. The missing apportation will be patched automatically.
                                                   ServiceAccount
                                                                          default postgres-operator Synced
arming, resource servicescounts/nostgres-operator is missing the kubertl. kubernetes in/last-applied-configuration apportation which is required by apply, apply should only be used on resource.
rces created declaratively by either create --save-config or apply. The missing annotation will be batched automatically.
                                                      Service
                                                                          default nostgres-operator Synced Healthy
 resource services/nostares-onerator is missing the kubertl-kubernetes in/last-annied-configuration annotation which is required by anniv anniv should only be used on resources created d
eclaratively by either create --eave-config or apply. The missing appotation will be natched automatically.
2023-06-23T16+50+01402+00 appo
                                                                          default most gree-operator Synced Progressing
                                                                                                                                          deployment anno/nostgres-operator configured.
Warning: resource deployments/postgres-operator is missing the kubertl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resource
a created declaratively by either create --save-config or apply. The missing apportation will be natched automatically.
2023-06-23T16:50:02402:00 anno
                                        Deployment default postgres-operator Supred Healthy
                                                                                                                  deployment appointance-operator configured. Marning, recourse deployment
ents/postgres-operator is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively b
y either create --save-config or apply. The missing annotation will be patched automatically.
2023-06-23T16:50:02+02:00 acid.zalan.do postgresgl default acid-test-cluster Synced
                                                                                                                  postgresgl.acid.zalan.do/acid-test-cluster configured. Warning: resourc
e postgresgls/acid-test-cluster is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declar
atively by either create --save-config or apply. The missing annotation will be patched automatically.
```

ArgoCD - Lancement de la synchronisation

```
https://kubernetes.default.svc
HRL:
                   https://10.109.47.144/applications/postgres-operator
Repo:
                   manifests
                                                                          STATUS HEALTH HOOK MESSAGE
                                                                                                  services count (nost gree-operator, configured, Warning, resource, services counts (nost gree-o
perator is missing the kubectl kubernetes to/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either creat
e -- save-config or apply. The missing annotation will be patched automatically.
                          ConfidMan
                                             default postgres-operator Synced
                                                                                                  configmap/postgres-operator configured. Warning: resource configmaps/postgres-operator is
 missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by apply should only be used on resources created declaratively by either create --save-c
rbac.authorization,k8s.io ClusterRole default postgres-pod
date, clusterrole.rbac.authorization.k8s.io/postgres-pod configured. Warning: resource clusterroles/postgres-pod is missing the kubectl.kubernetes.io/last-applied-configuration annotation wh
ich is required by apply, apply should only be used on resources created declaratively by either create --save-config or apply. The missing annotation will be patched automatically.
                                         default postgres-operator Running Synced
n annotation which is required by apply, apply should only be used on resources created declaratively by either create --save-config or apply. The missing annotation will be patched auto
rbac.authorization.k8s.io ClusterRoleBinding default postgres-operator Running Synced
required update, clusterrolebinding.rbac.authorization, k8s.io/postgres-operator configured. Warning: resource clusterrolebindings/postgres-operator is missing the kubectl.kubernetes.io/last
-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either create --save-config or apply. The missing annotation
                                             default postgres-operator Synced Healthy
                                                                                                  service/postgres-operator configured. Warning: resource services/postgres-operator is mis
sing the kubertl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either create --save-confi
g or apply. The missing annotation will be patched automatically.
                          Deployment
                                             default postgres-operator Synced Healthy
                                                                                                  deployment.apps/postgres-operator configured. Warning: resource deployments/postgres-oper
ator is missing the kubertl kubernetes io/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either create -
-save-config or apply. The missing annotation will be patched automatically.
                                             default acid-test-cluster Synced
                                                                                                  postgresql.acid.zalan.do/acid-test-cluster configured. Warning: resource postgresqls/acid
-test-cluster is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by apply, apply should only be used on resources created declaratively by either
create -- save-config or apply. The missing annotation will be patched automatically.
rbac.authorization.k8s.io ClusterRole
rbac.authorization.k8s.io ClusterRole
rhan authorization k8s in ClusterBoleBinding
```

Accès client à la base de données

Point-in-time recovery PITR

Failover - Changement du nombre de pods

Mise à jour mineure de PostgreSQL

Mise à jour majeure de PostgreSQL

Bibliographie



Webographie

Sommaire

Sommaire

Installation

Conclusion