

Установка Ceph CSI driver для подключения CephFS

Все работы проводим под root'ом

Заходим на `master-1` и клонируем репозиторий наш

```
cd ~  
git clone git@gitlab.slurm.io:edu/slurm.git  
cd ~/slurm/practice/7.datastorage/cephfs/
```

Далее будем работать на двух серверах: `node-1` и `master-1` Все команды `kubectl` выполняются на `master-1` Команды `ceph` и `rbd` выполняются на `node-1`

На серверах `ingress-1`, `node-1` и `node-2` развернут кластер `ceph`, более подробно установка рассмотрена в курсе по `ceph`.

На сервере `node-1` настроен доступ в `ceph` с правами администратора.

Получаем файл с настройками chart'a

1) Заходим на `master-1` и получаем переменные chart'a `ceph-csi-cephfs`

(ставить будем версию 2.1.2, потому что в версиях 3.x.x опять сломали изменение размера тома)

```
helm repo add ceph-csi https://ceph.github.io/csi-charts  
helm inspect values ceph-csi/ceph-csi-cephfs --version 2.1.2 >cephfs.yml
```

Заполняем переменные в `cephfs.yml`

2) Выполняем на `node-1`, чтобы узнать необходимые параметры:

Получаем `clusterID`

```
ceph fsid
```

Получаем список мониторов кластера `Ceph`

```
ceph mon dump
```

3) Возвращаемся на `master-1` и правим файл `cephfs.yml`

Заносим свои значение `clusterID`, и адреса мониторов, включаем создание политик PSP.

!! Внимание, адреса мониторов указываются в простой форме адрес:порт, потому что они передаются в модуль ядра для монтирования cephfs на узел.

!! А модуль ядра еще не умеет работать с протоколом мониторов v2

Список изменений в файле `cephfs.yml`. Опции в разделах `nodeplugin` и `provisioner` уже есть в файле, их надо исправить так как показано ниже.

`csiConfig:`

```
- clusterID: "bcd0d202-fba8-4352-b25d-75c89258d5ab"
  monitors:
    - "172.X.X.5:6789"
    - "172.X.X.6:6789"
    - "172.X.X.7:6789"
```

`nodeplugin:`

```
  podSecurityPolicy:
    enabled: true
```

`provisioner:`

```
  replicaCount: 1
  podSecurityPolicy:
    enabled: true
```

При необходимости можно свериться с файлом

`cephfs/cephfs-values-example.yml`

Устанавливаем чарт

4) Запускаем на `master-1`

```
helm upgrade -i ceph-csi-cephfs ceph-csi/ceph-csi-cephfs -f cephfs.yml -n
ceph-csi-cephfs --create-namespace --version 2.1.2
```

Создаем Secret и StorageClass

В документации указано, что provisioner'y CephFS необходимы права доступа администратора кластера. Но мы создадим отдельного пользователя fs, с более ограниченными правами

5) Создаем пользователя для cephfs

Возвращаемся на node-1

```
ceph auth get-or-create client.fs mon 'allow r' mgr 'allow rw' mds 'allow rws'
osd 'allow rw pool=cephfs_data, allow rw pool=cephfs_metadata'
```

6) Посмотреть ключ доступа для пользователя fs

```
ceph auth get-key client.fs
```

Вывод:

```
AQC09NJbhYipKRAAMqZsnqqS/T80YQX20xIa9A==
```

7) Заполняем манифест secret.yaml

Выполняем на master-1

```
cd ~/slurm/practice/7.datastorage/cephfs/
vi secret.yaml
```

Заносим имя пользователя fs и значение ключа в

```
adminID: fs
adminKey: AQC09NJbhYipKRAAMqZsnqqS/T80YQX20xIa9A==
```

8) Создаем секрет

```
kubectl apply -f secret.yaml
```

9) Получаем id кластера ceph

Выполняем на node-1

```
ceph fsid
```

10) Заносим clusterid в storageclass.yaml

Выполняем на master-1

```
vi storageclass.yaml
```

```
clusterID: bcd0d202-fba8-4352-b25d-75c89258d5ab
```

11) Создаем storageclass

```
kubectl apply -f storageclass.yaml
```

Проверяем как работает

12) Создаем pvc, и проверяем статус и наличие pv

```
kubectl apply -f pvc.yaml
```

```
kubectl get pvc
```

```
kubectl get pv
```

+2

Проверяем создание каталога в cephfs

13) монтируем CephFS на node-1

Выполняем на node-1

Точка монтирования

```
mkdir -p /mnt/cephfs
```

Создаем файл с ключом администратора

```
ceph auth get-key client.admin >/etc/ceph/secret.key
```

Добавляем запись в /etc/fstab

!!! Изменяем ip адрес на адрес узла node-1

```
echo "172.<xx>.<yy>.6:6789:/ /mnt/cephfs ceph
```

```
name=admin,secretfile=/etc/ceph/secret.key,noatime,_netdev 0 2">>/etc/fstab
```

```
mount /mnt/cephfs
```

14) Идем в каталог `/mnt/cephfs` и смотрим что там есть

```
cd /mnt/cephfs
```

Пробуем как работает `resize`

15) Изменяем размер тома в манифесте `pvc.yaml`

Возвращаемся на `master-1`

```
vi pvc.yaml
```

```
resources:
  requests:
    storage: 7Gi
```

```
kubectl apply -f pvc.yaml
```

16) Проверяем на `node-1`

```
yum install -y attr
```

```
getfattr -n ceph.quota.max_bytes <каталог-с-данными>
```

Пример команды

```
getfattr -n ceph.quota.max_bytes
/mnt/cephfs/volumes/csi/csi-vol-c7400bc4-19ef-11eb-98f6-56dd9a1d0fab/d6a7
99be-43e1-4980-aa5b-e38168ab6f7f/ +2
```

Вывод команды

```
getfattr: Removing leading '/' from absolute path names
# file:
mnt/cephfs/volumes/csi/csi-vol-c7400bc4-19ef-11eb-98f6-56dd9a1d0fab/d6a7
99be-43e1-4980-aa5b-e38168ab6f7f/
ceph.quota.max_bytes="7516192768"
```