部署Operator Hub 相关组件

简介

本文用于描述 Operator Hub产品所依赖的组件部署。

环境要求

Kubernetes 集群

步骤

部署OLM

功能: Operator 生命周期管理

OLM 是通过 yaml 方式提供安装，步骤如下；

```
#  release  install.sh
# https://github.com/operator-framework/operator-lifecycle-manager/releases
./install.sh <version>
```

```
customresourcedefinition.apiextensions.k8s.io/clusterserviceversions.operators.coreos.com configured
customresourcedefinition.apiextensions.k8s.io/installplans.operators.coreos.com configured
customresourcedefinition.apiextensions.k8s.io/subscriptions.operators.coreos.com configured
customresourcedefinition.apiextensions.k8s.io/catalogsources.operators.coreos.com configured
customresourcedefinition.apiextensions.k8s.io/operatorgroups.operators.coreos.com configured
namespace/olm created
namespace/operators created
serviceaccount/olm-operator-serviceaccount created
clusterrole.rbac.authorization.k8s.io/system:controller:operator-lifecycle-manager created
clusterrolebinding.rbac.authorization.k8s.io/olm-operator-binding-olm created
deployment.apps/olm-operator created
deployment.apps/catalog-operator created
clusterrole.rbac.authorization.k8s.io/aggregate-olm-edit created
clusterrole.rbac.authorization.k8s.io/aggregate-olm-view created
operatorgroup.operators.coreos.com/global-operators created
operatorgroup.operators.coreos.com/olm-operators created
clusterserviceversion.operators.coreos.com/packageserver created
catalogsource.operators.coreos.com/operatorhubio-catalog created
Waiting for deployment "olm-operator" rollout to finish: 0 out of 1 new replicas have been updated...
Waiting for deployment "olm-operator" rollout to finish: 0 of 1 updated replicas are available...
deployment "olm-operator" successfully rolled out
Waiting for deployment "catalog-operator" rollout to finish: 0 of 1 updated replicas are available...
deployment "catalog-operator" successfully rolled out
Package server phase: Installing
Package server phase: Succeeded
deployment "packageserver" successfully rolled out
```

此步骤会安装:

1. CRD: CSV, InstallPlan,Subscriptions,CatalogSource,OperatorGroup
2. Namespace: olm/operators

3. Operator: olm-operator/catalog-opeartor

另一种安装方式是使用 operator sdk 提供的命令

```
operator-sdk olm
```

此处暂未测试

部署 Alauda Operator 仓库

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: harbor
  namespace: olm
spec:
  sourceType: grpc
  image: harbor-b.alauda.cn/operator-test/index:v0.0.1
  publisher: harbor.alauda.cn
  displayName: Alauda Test Operators
```

注意其中的 image 部分，此image会包含 Alauda 发布的 Operators. CatalogSource创建之后， OLM 会负责生成 Operator 仓库

(Optional) 部署 Operator Marketplace

说明: 此步骤暂时我们不需要

功能: Operator 市场管理

通过 YAML方式安装

```
# from source code
kubectl apply -f deploy/upstream
```

```
namespace/marketplace unchanged
customresourcedefinition.apiextensions.k8s.io/operatorsources.operators.coreos.com created
serviceaccount/marketplace-operator unchanged
clusterrole.rbac.authorization.k8s.io/marketplace-operator unchanged
role.rbac.authorization.k8s.io/marketplace-operator unchanged
clusterrolebinding.rbac.authorization.k8s.io/marketplace-operator unchanged
rolebinding.rbac.authorization.k8s.io/marketplace-operator unchanged
operatorsource.operators.coreos.com/upstream-community-operators created
deployment.apps/marketplace-operator unchanged
```

主要资源：

- Namespace: marketplace
- CRD: OperatorSource
- Operator: marketplace-operator