# eSDK Huawei Storage Kubernetes CSI Plugins V3.2.0

## 用户指南

**文档版本** 01

发布日期 2022-12-30





#### 版权所有 © 华为技术有限公司 2023。 保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部,并不得以任何形式传播。

#### 商标声明



HUAWE和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标,由各自的所有人拥有。

#### 注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束,本文档中描述的全部或部分产品、服务或 特性可能不在您的购买或使用范围之内。除非合同另有约定,华为公司对本文档内容不做任何明示或暗示的声 明或保证。

由于产品版本升级或其他原因,本文档内容会不定期进行更新。除非另有约定,本文档仅作为使用指导,本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

## 华为技术有限公司

地址: 深圳市龙岗区坂田华为总部办公楼 邮编: 518129

网址: <a href="https://e.huawei.com">https://e.huawei.com</a>

## 前言

## 读者对象

本文档主要适用于以下读者对象:

- 技术支持工程师
- 运维工程师
- 具备存储和Kubernetes基础知识的工程师

## 符号约定

在本文中可能出现下列标志,它们所代表的含义如下。

符号	说明
▲ 危险	表示如不避免则将会导致死亡或严重伤害的具有高等级风险的危害。
▲ 警告	表示如不避免则可能导致死亡或严重伤害的具有中等级风险的危害。
▲ 注意	表示如不避免则可能导致轻微或中度伤害的具有低等级风险的危害。
须知	用于传递设备或环境安全警示信息。如不避免则可能会导致设备 损坏、数据丢失、设备性能降低或其它不可预知的结果。 "须知"不涉及人身伤害。
□ 说明	对正文中重点信息的补充说明。 "说明"不是安全警示信息,不涉及人身、设备及环境伤害信 息。

## 修改记录

文档版本	发布日期	修改说明
01	2022-12-30	第一次正式发布。

## 目录

則言	ii
1 概述	1
	2
<b>2.1 Kubernetes 及操作系统兼容性</b>	
2.2 Kubernetes 特性矩阵	
2.3 华为企业存储兼容性	
2.4 华为分布式存储兼容性	
3 安装前准备	
3.1 前提条件	
3.2 下载华为 CSI 软件包	
3.3 上传华为 CSI 镜像	
3.3.1 上传镜像到镜像仓库	
3.3.2 导入镜像到所有节点	
3.4 检查 CSI 依赖的镜像	
3.5 检查卷快照依赖组件	13
3.6 检查主机多路径配置	14
3.7 检查华为存储上的账号配置	15
3.8 检查主机依赖软件状态	15
3.9 通信矩阵	16
4 安装华为 CSI	17
4.1 使用 Helm 安装华为 CSI	
4.1.1 准备 values.yaml 文件	
- 4.1.2 安装华为 CSI	33
4.2 手动编写 ConfigMap 文件部署华为 CSI	35
4.2.1 创建华为 CSI 运行时需要的 ConfigMap	35
4.2.1.1 通过 iSCSI 对接企业存储 SAN	35
4.2.1.2 通过 FC 对接企业存储 SAN	37
4.2.1.3 通过 NFS 对接企业存储 NAS	39
4.2.1.4 通过 NVMe over RoCE 对接企业存储 SAN	41
4.2.1.5 通过 NVMe over FC 对接企业存储 SAN	43
4.2.1.6 通过 SCSI 对接分布式存储 SAN	
4.2.1.7 通过 iSCSI 对接分布式存储 SAN	47

707 7410	н 30
4.2.1.8 通过 NFS 对接分布式存储 NAS	50
4.2.1.9 通过 DPC 对接分布式存储 NAS	
4.2.1.10 通过租户/账户对接华为存储后端	
4.2.1.11 通过 CSI 在后端发放双活卷	
4.2.1.12 通过 CSI 对接多个后端存储	55
4.2.2 启动 huawei-csi 服务	56
5 卸载华为 CSI	61
5.1 使用 Helm 卸载 huawei-csi	
5.2 手动卸载 huawei-csi	62
5.2.1 卸载 huawei-csi-node 服务	
5.2.2 卸载 huawei-csi-controller 服务	62
5.2.3 删除 huawei-csi-configmap 对象	63
5.2.4 删除 huawei-csi-secret 对象	63
5.2.5 删除 RBAC 权限	63
5.2.6 删除老版本镜像	64
5.3 卸载 Snapshot 依赖组件服务	65
6 升级/回退操作	67
6.1 使用 Helm 升级/回退华为 CSI	67
6.1.1 升级华为 CSI	67
6.1.2 回退 CSI	68
6.2 手动编写 ConfigMap 文件升级/回退华为 CSI	69
6.2.1 卸载原 CSI	69
6.2.2 安装新 CSI	69
7 使用华为 CSI	71
7.1 PV/PVC 管理	71
7.1.1 创建 PVC	71
7.1.1.1 动态卷供应	71
7.1.1.1.1 配置 StorageClass	72
7.1.1.1.2 配置 PVC	82
7.1.1.2 静态卷供应	84
7.1.1.2.1 配置 PV	85
7.1.1.2.2 配置 PVC	88
7.1.2 扩容 PVC	90
7.1.3 克隆 PVC	91
7.1.4 从快照创建 PVC	92
7.2 创建 VolumeSnapshot	93
7.2.1 检查卷快照依赖组件信息	
7.2.2 配置 VolumeSnapshotClass	
7.2.3 配置 VolumeSnapshot	94
8 高级特性	97
8.1 配置 ALUA 特性	97

8.1.1 通过 Helm 配置 ALUA 特性	97
8.1.1.1 配置华为企业存储后端的 ALUA 参数	97
8.1.1.2 配置分布式存储后端的 ALUA 参数	100
8.1.2 手动配置 ALUA 特性	101
8.1.2.1 配置华为企业存储后端的 ALUA 参数	102
8.1.2.2 配置分布式存储后端的 ALUA 参数	105
8.2 配置存储拓扑感知	106
8.2.1 通过 Helm 配置存储拓扑感知	107
8.2.2 手动配置存储拓扑感知	109
9 常用操作	112
9.1 更新 CSI 上配置的存储用户名或密码	112
9.2 更新 huawei-csi 的 configmap 对象	113
9.2.1 使用 Helm 更新 configmap 对象	113
9.2.2 手动更新 configmap 对象	114
9.3 为 huawei-csi 新增后端	116
9.3.1 使用 Helm 新增后端	116
9.3.2 手动新增后端	117
9.4 更新 huawei-csi-controller 服务	117
9.4.1 使用 Helm 更新 controller 服务	118
9.4.2 手动更新 controller 服务	119
9.5 更新 huawei-csi-node 服务	119
9.5.1 使用 Helm 更新 node 服务	119
9.5.2 手动更新 node 服务	120
9.6 修改日志输出模式	121
9.6.1 使用 Helm 修改 controller 或 node 服务的日志输出模式	121
9.6.2 手动修改 huawei-csi-controller 服务的日志输出模式	121
9.6.3 手动修改 huawei-csi-node 服务的日志输出模式	123
9.7 开启 ReadWriteOncePod 功能门	124
9.8 配置非 root 用户访问 Kubernetes 集群	125
10 FAQ	127
10.1 如何查看华为 CSI 日志	127
10.2 Kubernetes 平台第一次搭建时, iscsi_tcp 服务没有正常启动,导致创建 Pod 失败	129
10.3 启动 huawei-csi-node 失败,提示错误为:"/var/lib/iscsi is not a directory"	129
10.4 集群中 worker 节点宕机并恢复后,Pod 完成 failover,但是 Pod 所在源主机出现盘符残留	130
10.5 启动 huawei-csi 服务时,服务启动异常, 状态显示 InvalidImageName	132
10.6 创建 PVC 时, PVC 的状态为 Pending	133
10.7 删除 PVC 前,PVC 的状态为 Pending	135
10.8 创建 Pod 时,Pod 的状态为 ContainerCreating	136
10.9 创建 Pod 时,Pod 的状态长时间处于 ContainerCreating 状态	137
10.10 创建 Pod 失败,日志显示执行 mount 命令超时	137
10.11 创建 Pod 失败,日志显示执行 mount 命令失败	138
10.12 如何下载容器镜像到本地	138

用尸指图	<u> 日 录</u>
10.13 如何获取 CSI 版本信息	139
10.14 对接 Tanzu Kubernetes 集群常见问题及解决方法	139
10.14.1 未创建 PSP 权限导致 Pod 无法创建	139
10.14.2 修改主机挂载点	140
10.14.3 修改 livenessprobe 容器的默认端口	
10.15 使用 Tanzu Kubernetes 集群时常见问题及解决方法	142
10.15.1 创建临时卷失败	142
10.16 通用临时卷扩容失败	143
10.17 PVC 扩容的目标容量超过存储池容量导致扩容失败	
11 附录	144
11.1 OceanStor V3/V5 系列和 OceanStor Dorado V3 系列 ALUA 特性配置策略样例	144
11.2 OceanStor Dorado 6.x ALUA 特性配置策略样例	
11.3 分布式存储 ALUA 特性配置策略样例	145
11.3 分布式存储 ALUA 特性配置策略样例11.4 安装 Helm 3	146
11.5 制作 CSI 镜像	

# ● 概述

Kubernetes,简称 K8s或者 "kube",是一个可移植、可扩展的开源平台,用于管理容器化的工作负载和服务。

容器存储接口(Container Storage Interface),简称 CSI,是一种行业标准,用于将块和文件存储系统暴露给 Kubernetes 等容器编排系统 (CO) 上的容器工作负载。华为CSI插件用于和华为企业存储和分布式存储产品进行通信,为Kubernetes的容器工作负载提供存储服务。是华为企业存储和分布式存储在Kubernetes环境中使用的必须插件。

Kubernetes与华为CSI以及华为存储的整体结构如下:

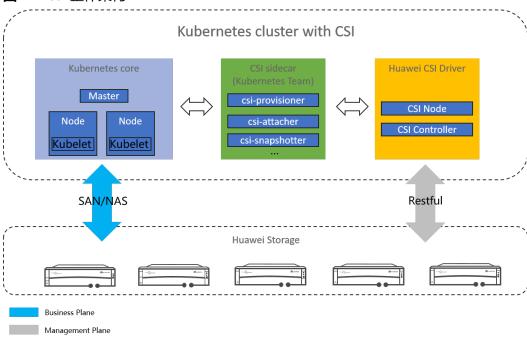


图 1-1 CSI 整体架构

- Kubernetes通过其官方维护的一系列sidecar组件负责注册监听Kubernetes对象资源,并在需要的时候发起对CSI Driver调用。
- 华为CSI Driver将sidecar发起的调用在华为存储上实施,如创建一个持久卷 (PersistentVolume, PV)的操作被实施为在华为存储上创建一个LUN/文件系统。

- CSI Controller:以Deployment方式单独运行的一个Pod,用于和存储交互,在存储侧对资源进行创建、删除等操作,如创建LUN/文件系统,扩容等。
- CSI Node:以DaemonSet方式运行在Kubernetes工作节点上的Pod,用于在工作节点上对华为存储提供的LUN/文件系统资源进行挂载和卸载、以及格式化LUN为本地文件系统等操作。
- 华为存储为Kubernetes工作节点通过多种协议提供SAN/NAS存储资源服务,华为存储和华为CSI驱动之间则使用RESTful进行通信。

本文档主要介绍华为CSI V3.2.0插件安装部署和使用。

本章节会详细说明华为CSI插件支持的容器管理平台、操作系统、多路径软件以及CSI 插件配合华为存储所提供的特性和功能。

- 2.1 Kubernetes及操作系统兼容性
- 2.2 Kubernetes特性矩阵
- 2.3 华为企业存储兼容性
- 2.4 华为分布式存储兼容性

## 2.1 Kubernetes 及操作系统兼容性

华为CSI插件支持如下容器管理平台:

表 2-1 支持的容器管理平台

容器管理平台	版本
Kubernetes	1.13~1.25
Red Hat OpenShift	4.6 EUS, 4.7, 4.8, 4.9, 4.10, 4.11
Tanzu Kubernetes	TKGI 1.14.1 + Kubernetes 1.23.7

#### 须知

- 华为CSI对接Red Hat OpenShift的相关操作请参见《eSDK Enterprise Storage Plugins 用户指南(Kubernetes CSI for Red Hat OpenShift)》。
- 华为CSI对接Tanzu Kubernetes的相关FAQ请参见10.14 对接Tanzu Kubernetes集群常见问题及解决方法。

华为CSI插件支持的操作系统以及多路径信息如表2-2所示。

表 2-2 支持的操作系统及多路径信息

操作系统名称	操作系统版本	原生DM- Multipath版本	华为UltraPath版本
CentOS x86_64	7.6, 7.7, 7.9	随OS自带,支 持FC/iSCSI	UltraPath 31.1.0,支持FC/iSCSI
CentOS x86_64	8.2	随OS自带,支 持FC/iSCSI	UltraPath 31.1.0,支持FC/iSCSI UltraPath-NVMe 31.1.RC8,支 持NVMe over RoCE/NVMe over FC
SUSE 15 x86_64	SP2, SP3	随OS自带,支持FC/iSCSI	UltraPath 31.1.0,支持FC/iSCSI UltraPath-NVMe 31.1.RC8,支 持NVMe over RoCE/NVMe over FC
Red Hat CoreOS x86_64	4.6, 4.7, 4.8, 4.9, 4.10, 4.11	随OS自带,支 持FC/iSCSI	不支持
Ubuntu x86_64	18.04, 20.04, 22.04	随OS自带,支 持FC/iSCSI	不支持
Kylin V10 x86_64	SP1, SP2	随OS自带,支 持FC/iSCSI	不支持
Kylin V10 ARM	SP1, SP2	随OS自带,支 持FC/iSCSI	不支持
Debian x86_64	11	随OS自带,支 持FC/iSCSI	不支持
EulerOS x86_64	V2R10	随OS自带,支 持FC/iSCSI	不支持
EulerOS ARM	V2R10	随OS自带,支 持FC/iSCSI	不支持

## 2.2 Kubernetes 特性矩阵

本章节说明华为CSI在不同Kubernetes版本下支持的特性。

表 2-3 Kubernetes 版本与支持的特性

特性	V1.1	V1.2	V1.2						
	3	4	5	6	7	8	9	0	1+
Static Provisionin g	V	V	V	V	V	V	V	V	V

特性	V1.1 3	V1.1 4	V1.1 5	V1.1 6	V1.1 7	V1.1 8	V1.1 9	V1.2 0	V1.2 1+
Dynamic Provisionin g	V	V	V	V	V	V	V	V	V
Expand Persistent Volume	×	×	×	V	V	V	V	V	V
Create VolumeSn apshot	×	×	×	×	V	V	V	V	V
Restore VolumeSn apshot	×	×	×	×	V	V	V	V	V
Delete VolumeSn apshot	×	×	×	×	V	V	V	V	V
Clone Persistent Volume	×	×	×	×	V	V	V	V	V
Raw Block Volume	×	V	V	V	V	V	V	V	V
Topology	×	V	V	$\sqrt{}$	V	V	V	V	$\checkmark$
Generic Ephemera I Volumes	×	×	×	×	×	×	×	×	V

## 2.3 华为企业存储兼容性

华为CSI插件兼容华为OceanStor系列的全闪存存储和混合闪存存储,具体支持的存储版本如下表所示:

表 2-4 支持的华为企业存储

存储产品	版本
OceanStor V3	V300R006
OceanStor V5	V500R007, V500R007 Kunpeng
OceanStor Dorado V3	V300R002
OceanStor V6	6.1.3, 6.1.5
OceanStor Dorado V6	6.0.0, 6.0.1, 6.1.0, 6.1.2, 6.1.3, 6.1.5

华为CSI插件针对华为企业存储支持如下特性。

表 2-5 华为企业存储支持的特性及约束

特性	OceanSt or V3	OceanStor V5	OceanStor Dorado V3	OceanStor V6	OceanStor Dorado V6		
Static Provisionin g	SAN: FC/ iSCSI <sup>1</sup> NAS:	SAN: FC/ iSCSI <sup>1</sup> NAS: NFS	SAN: FC/ iSCSI <sup>1</sup>	SAN: FC/ iSCSI/NVMe over RoCE/ NVMe over	SAN: FC/ iSCSI/NVMe over RoCE/ NVMe over		
Dynamic Provisionin g	NFS 3	3		FC <sup>1</sup> NAS: NFS 3/4.0/4.1	FC <sup>1</sup> NAS: NFS 3/4.0/4.1 <sup>2</sup>		
Expand Persistent Volume	仅支持使用	Dynamic Prov	visioning方式创	 別建的卷			
Create VolumeSna pshot	仅支持使用	Dynamic Prov	visioning方式包	刘建的非双活卷			
Delete VolumeSna pshot	支持	支持	支持	支持	支持		
Restore VolumeSna pshot	支持	支持	支持	SAN: 支持 NAS: 仅 6.1.5支持	SAN: 支持 NAS: 仅 6.1.5支持		
Clone Persistent Volume	仅支持使用 创建的非双	Dynamic Prov 活卷	SAN: 支持使 Provisioning 双活卷 NAS: 仅6.1.5 Dynamic Prov 创建的非双活	方式创建的非 支持使用 visioning方式			
Raw Block Volume	仅支持 SAN类型 的卷	仅支持SAN 类型的卷	仅支持SAN 类型的卷	仅支持SAN 类型的卷	仅支持SAN 类型的卷		
Topology	支持	支持	支持	支持	支持		
Generic Ephemeral Volumes	支持	支持	支持	支持	支持		
Access Mode	RWO/ROX/RWOP: 所有类型卷均支持,RWOP需Kubernetes 1.22版本以上支持。 RWX: 仅Raw Block卷和NFS类型的卷支持。						

特性	OceanSt or V3	OceanStor V5	OceanStor Dorado V3	OceanStor V6	OceanStor Dorado V6
QoS	支持	支持	支持	支持	支持
应用类型	不涉及	不涉及	不涉及	支持	支持
卷双活	不支持	不支持	不涉及	仅支持NAS类型的卷	
存储多租户	仅支持NAS类型的卷		不涉及	仅支持NAS类型的卷 <sup>3</sup>	

- 注释1 若用户的容器平台部署在虚拟化环境中,则仅支持iscsi组网;使用NVMe over RoCE或NVMe over FC时,worker节点nvme-cli工具版本不低于1.9,查询命令: nvme version。
- 注释2 仅OceanStor Dorado V6 6.1.0及以后版本支持NFS。仅OceanStor Dorado V6 6.1.3及以后版本NFS 4.1。
- 注释3 仅OceanStor Dorado V6 6.1.3及以后版本支持多租户。

## 2.4 华为分布式存储兼容性

华为CSI插件兼容华为OceanStor系列的分布式存储系统,具体支持的存储版本如下表所示:

表 2-6 支持的华为分布式存储

存储产品	版本
FusionStorage	V100R006C30
FusionStorage Block	8.0.0, 8.0.1
OceanStor Pacific系列	8.1.0, 8.1.1, 8.1.2, 8.1.3

华为CSI插件针对华为分布式存储支持如下特性。

表 2-7 华为分布式存储支持的特性及约束

特性	FusionStorage	FusionStorage Block	OceanStor Pacific 系列
Static Provisioning	SAN: iSCSI/	SAN: iSCSI/SCSI	SAN: iSCSI/SCSI
Dynamic Provisioning	SCSI		NAS: DPC <sup>1</sup> /NFS 3/4.1 <sup>2</sup>
Expand Persistent Volume	仅支持使用Dynan	nic Provisioning方式f	创建的卷

特性	FusionStorage	FusionStorage Block	OceanStor Pacific 系列		
Create VolumeSnapshot	仅支持使用Dynamic Provisioning方式创建的SAN类型卷				
Delete VolumeSnapshot	支持	支持	仅支持SAN类型的卷 快照		
Restore VolumeSnapshot	支持	支持	仅支持SAN类型的卷 快照		
Clone Persistent Volume	仅支持使用Dynamic Provisioning方式创建的SAN类型卷				
Raw Block Volume	仅支持SAN类型 的卷	仅支持SAN类型的 卷	仅支持SAN类型的卷		
Topology	支持	支持	支持		
Generic Ephemeral Volumes	支持	支持	支持		
Access Mode	RWO/ROX/RWOF 1.22及以上版本支		,RWOP在Kubernetes		
	RWX: 仅Raw Block卷和NFS类型的卷支持。				
QoS	支持	支持	仅支持SAN类型的卷		
软配额	不支持	不支持	仅支持NAS类型的卷		
存储多租户	不支持	不支持	仅支持NAS类型的卷		

- 注释1 仅OceanStor Pacific系列 8.1.2及以后版本支持DPC。华为CSI支持的操作系统对DPC的支持请参考对应产品版本兼容性文档。
- 注释2 仅OceanStor Pacific系列 8.1.2及以后版本支持NFS 4.1。

# **3** 安装前准备

在安装华为CSI插件到您的容器管理平台前,需要CSI镜像制作、上传/导入镜像以及必要的主机多路径环境配置工作。本章节将对安装前的准备工作进行详细说明。

- 3.1 前提条件
- 3.2 下载华为CSI软件包
- 3.3 上传华为CSI镜像
- 3.4 检查CSI依赖的镜像
- 3.5 检查卷快照依赖组件
- 3.6 检查主机多路径配置
- 3.7 检查华为存储上的账号配置
- 3.8 检查主机依赖软件状态
- 3.9 通信矩阵

## 3.1 前提条件

在进行本章节所说明的操作前,请确保如下条件已经具备:

- 容器管理平台已部署完成并正常运行,且满足**2.1 Kubernetes及操作系统兼容性** 章节的要求。
- 已完成对接华为企业存储初始化配置,包括存储池划分、端口配置等。且存储产品的版本满足**2.3 华为企业存储兼容性**章节的要求。
- 已完成对接华为分布式存储初始化配置,包括存储池划分、端口配置等。且存储 产品的版本满足**2.4 华为分布式存储兼容性**章节的要求。
- 完成华为存储和容器平台主机连通性配置,根据您的规划,请确保容器集群上工作节点已经安装对应协议所需要的软件客户端,如iSCSI客户端、NFS客户端等。
- 如果是多路径组网,请确保所有worker节点上已安装多路径软件,详细操作请参考表2-2。
- Kubernetes的所有worker节点与待接入的存储设备的业务IP地址通信正常, iSCSI 场景下允许使用ping命令进行连通性校验。
- 已安装Docker的Linux主机,且该主机可以访问用户的镜像仓库。

## 3.2 下载华为 CSI 软件包

本章节详细说明了下载方法以及软件包组件结构。

步骤1 打开浏览器,访问仓库地址: https://github.com/Huawei/eSDK\_K8S\_Plugin/releases。

步骤2 根据存储类型和CPU架构,下载对应的3.2.0版本软件包。

#### □□ 说明

软件包命名规范:存储类型+插件名称(Kubernetes\_CSI\_Plugin)+版本号+CPU架构例如:使用分布式存储对接x86架构的主机时,需要下载的软件包名为eSDK\_Huawei\_Storage\_Kubernetes\_CSI\_Plugin\_V3.2.0\_X86\_64.zip

步骤3 将下载的软件包解压。软件包组件结构如表3-1所示。

表 3-1 软件包组件描述

组件	组件描述	
image/huawei-csi-v3.2.0- arch.tar	huawei-csi镜像,"arch"为x86或arm。	
bin/huawei-csi	实现CSI规范接口的服务组件。	
bin/secretGenerate	明文密码加密工具,用于生产secret对象。	
bin/secretUpdate	明文密码加密工具,用于更新secret对象。	
helm	Helm组件,用于部署华为CSI。	
deploy	CSI部署过程中,yaml示例文件。	
examples	CSI使用过程中,yaml示例文件。	
tools	无镜像仓库时,用于上传镜像的脚本。	

----结束

## 3.3 上传华为 CSI 镜像

当前华为已经提供huawei-csi镜像供用户使用,镜像文件获取参见<mark>表3-1</mark>。如果用户需要重新制作镜像,请参见**11.5 制作CSI镜像**。

为了后续在容器管理平台中可以使用CSI镜像,需要按照以下方式中的一种提前将CSI 镜像导入到集群中:

- 使用Docker工具,将CSI镜像上传至镜像仓库(推荐)
- 使用镜像上传脚本,将CSI镜像导入到所有需要部署华为CSI的节点。

## 3.3.1 上传镜像到镜像仓库

#### 前提条件

已准备一台已安装Docker的Linux主机,且该主机支持访问镜像仓库。

#### 操作步骤

步骤1 执行docker load -i huawei-csi.tar命令,将CSI镜像导入当前节点。

# docker load -i huawei-csi.tar Loaded image: huawei-csi:3.2.0

**步骤2** 执行docker tag huawei-csi:3.2.0 repo.huawei.com/huawei-csi:3.2.0命令,添加镜像仓库地址到镜像标签。其中repo.huawei.com表示镜像仓库的地址。

# docker tag huawei-csi:3.2.0 repo.huawei.com/huawei-csi:3.2.0

步骤3 执行docker push repo.huawei.com/huawei-csi:3.2.0命令,将CSI镜像上传到镜像仓库。其中repo.huawei.com表示镜像仓库的地址。

# docker push repo.huawei.com/huawei-csi:3.2.0

----结束

## 3.3.2 导入镜像到所有节点

若镜像已上传至镜像仓库,则跳过本章节。

#### 前提条件

- 镜像所在的主机与需要导入镜像的所有主机之间能进行SSH通信。
- 镜像所在的主机上已经安装expect、sshpass和scp软件包。

### 操作步骤

步骤1 执行vi worker-list.txt命令,创建worker-list.txt配置文件。

# vi worker-list.txt

**步骤2** 配置worker-list.txt文件,worker-list.txt文件的模板如下所示。按l或lnsert进入编辑状态,添加节点信息。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

# ip 192.168.128.16 192.168.128.17

#### 步骤3 上传并导入镜像。

- 如果使用的是containerd作为容器运行时,执行**./containerd-upload.sh** *worker-list.txt huawei-csi.tar*命令,按提示输入用户名和密码,上传并导入镜像。 # ./containerd-upload.sh worker-list.txt huawei-csi.tar
- 如果使用的是Docker作为容器运行时,执行**./docker-upload.sh** *worker-list.txt huawei-csi.tar*命令,按提示输入用户名和密码,上传并导入镜像。
  # ./docker-upload.sh worker-list.txt huawei-csi.tar

#### 步骤4 检查导入镜像是否成功。

1. 如果脚本最后提示All images are uploaded successfully,则表示已成功将镜像导入到所有节点。

All images are uploaded successfully

2. 如果脚本最后回显是如下的列表,则表示列表中的节点镜像导入是失败的,这时 需要检查日志检查原因。

List of nodes to which the image fails to be imported: 192.168.128.16 192.168.128.17

#### ----结束

## 3.4 检查 CSI 依赖的镜像

华为CSI安装过程中需要依赖下表中的镜像,若集群中的所有worker节点已连接互联网能够在线拉取镜像,则可跳过本章节。若集群中的节点无法连接互联网,则请根据使用的Kubernetes版本,下载对应的镜像文件并上传到镜像仓库中或者导入Kubernetes集群的所有worker节点中。

huawei-csi-controller服务依赖的sidecar镜像: livenessprobe、csi-provisioner、csi-attacher、csi-resizer、csi-snapshotter、snapshot-controller和huawei-csi-driver。

huawei-csi-node服务依赖的sidecar镜像: livenessprobe、csi-node-driver-registrar和 huawei-csi-driver。

关于每个镜像的功能和详情,请参考下表。

表 3-2 Huawei CSI 依赖的镜像

容器名称	容器镜像	K8s 版本 要求	功能描述	官方说明
livenesspr obe	k8s.gcr.io/sig-storage/ livenessprobe:v2.5.0	v1.13 +	用于监控CSI的健康状态,并上报给Kubernetes,使 Kubernetes能够自动检测 CSI程序的问题并重启Pod 尝试修改该问题。	查看 详情
csi-resizer	k8s.gcr.io/sig-storage/ csi-resizer:v1.4.0	v1.13 +	在扩容PVC时,调用CSI给 PVC提供更多的存储容量 空间。	查看 详情
csi-node- driver- registrar	k8s.gcr.io/sig-storage/ csi-node-driver- registrar:v2.3.0	v1.13 +	用于获取CSI信息,并通过 kubelet的插件注册机制将 节点注册到kubelet中,从 而Kubernetes能够感知该 节点与华为存储的对接。	查看 详情
csi- snapshott er	k8s.gcr.io/sig-storage/ csi-snapshotter:v4.2.1	v1.17 +	在创建/删除 VolumeSnapshot时,调 用CSI在存储侧完成快照的 创建和删除。	查看 详情

容器名称	容器镜像	K8s 版本 要求	功能描述	官方说明
snapshot- controller	k8s.gcr.io/sig-storage/ snapshot- controller:v4.2.1	v1.17 +	在创建/删除 VolumeSnapshot时,监 听Kubernetes API中关于 VolumeSnapshot和 VolumeSnapshotContent 的对象,并触发csi- snapshotter在存储上完成 快照的创建。	查看 详情
csi- provisione	k8s.gcr.io/sig-storage/ csi-provisioner:v3.0.0	er:v3.0.0 + Controller服	• 在创建PVC时,调用CSI Controller服务在存储	查看 详情
r	quay.io/k8scsi/csi- provisioner:v1.4.0	v1.13. x- v1.16. x	上创建LUN/文件系统作为PV,并将PV绑定至PVC。  在删除PVC时,调用CSIController服务解除PV至PVC的绑定,然后在存储上删除该PV对应的LUN/文件系统	
csi- attacher	k8s.gcr.io/sig-storage/ csi-attacher:v3.4.0	v1.17 +	在创建/删除Pod时,调用 CSI Controller服务执行	查看 详情
	quay.io/k8scsi/csi- attacher:v1.2.1	v1.13. x- v1.16. x	Publish/Unpublish Volume操作。	

#### 山 说明

如何下载容器镜像到本地,请参考10.12 如何下载容器镜像到本地。

## 3.5 检查卷快照依赖组件

如果您需要在容器环境中使用卷快照以及卷快照关联的特性,请通过如下步骤检查您的环境是否部署了卷快照依赖组件以及卷快照Api资源版本信息。若当前Kubernetes版本不支持卷快照相关特性,请跳过本章节。支持卷快照的Kubernetes版本请参考表2-3。

#### 操作步骤

步骤1 执行命令,查看snapshot相关资源服务安装详情。

# kubectl api-resources | grep snapshot | awk '{print \$1}' volumesnapshotclasses volumesnapshotcontents volumesnapshots

- 如果回显如上所示,则说明快照依赖组件服务已安装,请进入步骤2,检查Api资源信息。
- 如果回显缺少如上所示的任意一项服务,请进入步骤3,安装Snapshot依赖组件服务。

#### 步骤2 执行命令,查看卷快照Api资源信息。

# kubectl api-versions | grep "snapshot.storage.k8s.io" snapshot.storage.k8s.io/v1 snapshot.storage.k8s.io/v1beta1

- 如果回显如上所示,则说明快照依赖组件服务支持v1和v1beta1版本的Api资源, 跳过本章节。
- 如果回显缺少如上所示的任意一项服务,请继续执行后续步骤进行安装。

步骤3 进入/helm/esdk/crds/snapshot-crds目录,执行以下命令,安装Snapshot依赖组件服务。组件包路径请参考表3-1。

# kubectl apply -f huawei-csi-snapshot-crd-v1.yaml --validate=false

#### ----结束

完成安装后,可以使用步骤1中所示的命令检查snapshot相关资源服务安装详情。

## 3.6 检查主机多路径配置

当您计划在容器环境中使用FC/iSCSI/NVMe over RoCE/NVMe over FC协议对华为存储进行访问时,推荐您使用主机多路径软件增强主机和存储的链路冗余和性能。如果您不准备使用多路径软件,请跳过本章节。

华为CSI软件支持对接的操作系统和多路径软件请参考表2-2。

#### □ 说明

- 如果您准备使用FC/iSCSI协议对接华为存储时,推荐使用操作系统自带的原生DM-Multipath。
- 如果您准备使用NVMe over RoCE/NVMe over FC协议对接华为存储时,推荐使用华为自研的UltraPath-NVMe。
- 如果您使用SCSI协议对接华为存储时,请关闭操作系统自带的DM-Multipath。

#### 前提条件

主机多路径软件已经被正确的安装在主机上。

- 如果您使用的是操作系统自带的原生DM-Multipath,请咨询您的主机或操作系统 提供商获取安装所需的资料和软件包。
- 如果您使用的是华为自研的UltraPath或者UltraPath-NVMe,请联系华为工程师获取UltraPath或者UltraPath-NVMe的资料和软件包。软件包版本请参考表2-2。

#### 检查步骤

步骤1 如果您使用iSCSI/FC协议对接华为企业存储,请参考OceanStor Dorado 6.x & OceanStor 6.x在Red Hat下的主机连通性指南文档的"配置多路径"->"配置多路径(非双活场景)"章节,对主机多路径进行配置和检查。

步骤2 如果您使用NVMe over RoCE/NVMe over FC协议对接华为企业存储,请参考
OceanStor Dorado 6.x & OceanStor 6.x在Red Hat下的主机连通性指南文档的"配

置多路径"->"配置多路径(非双活场景)"->"UltraPath多路径"章节,对主机多路径进行配置和检查。

- 步骤3 如果您使用iSCSI协议对接华为分布式存储,请参考《FusionStorage 8.0.1 块存储基础业务配置指南》中的"应用服务器配置多路径"章节,对主机多路径进行配置和检查。
- **步骤4** 如果您使用了操作系统原生多路径时,需要检查/etc/multipath.conf文件,检查文件是否存在如下配置:

```
defaults {
    user_friendly_names yes
    find_multipaths no
}
```

如果配置不存在,请在/etc/multipath.conf文件开始处增加该配置项。

----结束

## 3.7 检查华为存储上的账号配置

当华为存储接入容器平台后,华为CSI需要在华为存储上根据业务要求,管理存储资源,如创建卷、映射卷等操作。此时,华为CSI需要使用华为存储上已经创建的账号和华为存储进行通信。针对不同存储设备所需要的账号信息如表3-3所示。

丰	3-3	左徐对接	CSI	时使用的账号要求
æ	<b>J</b> -J	1丁  沿入1] 女	CJI	

存储类型	用户类型	角色	级别	类型
OceanStor V3/V5	系统用户	管理员	管理员	本地用户
	租户用户	租户管理员	管理员	本地用户
OceanStor Dorado V3	系统用户	管理员	管理员	本地用户
OceanStor 6.1	系统用户	管理员	N/A	本地用户
OceanStor	系统用户	管理员	N/A	本地用户
Dorado 6.1.3/6.1.5	租户用户	租户管理员	N/A	本地用户
OceanStor Pacific 系列	系统用户	管理员	N/A	本地用户

## 3.8 检查主机依赖软件状态

本章节介绍如何检查集群中工作节点上主机依赖软件状态是否正常。本例中主机操作系统为CentOS 7.9 x86\_64。

- 检查iSCSI客户端状态。# systemctl status iscsi iscsid
- 检查NFS客户端状态。# systemctl status rpcbind

- 检查DM-Multipath多路径软件状态。# systemctl status multipathd.socket multipathd
- 检查UltraPath多路径软件状态。# systemctl status nxup
- 检查UltraPath-NVMe多路径软件状态。# systemctl status upudev upService\_plus

## 3.9 通信矩阵

本章节用于描述CSI/存储/kubelet之间的通信关系,包括:通信使用的端口、协议、IP地址、认证方式、端口说明信息等。具体如表3-4所示。

表 3-4 通信矩阵

源设备	CSI所在主机	CSI所在主机
源IP地址	CSI所在主机IP地址	CSI所在主机IP地址
源端口	1024~65535	1024~65535
目的设备	存储设备	CSI所在主机
目的IP地址	存储设备管理IP地址	CSI所在主机IP地址
目的端口	8088	9800/9808
协议	HTTPS	НТТР
认证方式	用户名/密码	无
端口说明	<ul> <li>1024~65535:该端口用于CSI和存储建立连接,其端口分配由Linux临时端口号范围决定。</li> <li>8088:该端口由存储提供,用于CSI请求卷创建/管理/删除等一系列动作。</li> </ul>	<ul> <li>1024~65535: 该端口分配由Linux临时端口号范围决定。</li> <li>9800/9808: 该端口由CSI提供,用于kubelet检测CSI健康状态。</li> </ul>

## 4 安装华为 CSI

本章节介绍如何安装部署华为CSI,推荐使用Helm安装华为CSI,手动安装华为CSI的方式当前版本仍然保留,但是会在后续版本删除。

华为CSI的安装支持root用户和非root用户。使用非root用户安装华为CSI时,需要保证当前用户能够访问Kubernetes集群的API Server,配置非root用户访问Kubernetes集群请参考9.8 配置非root用户访问Kubernetes集群。

华为CSI必须在root用户权限下运行。

- 4.1 使用Helm安装华为CSI
- 4.2 手动编写ConfigMap文件部署华为CSI

## 4.1 使用 Helm 安装华为 CSI

本章节介绍如何使用Helm 3安装华为CSI。

Helm是Kubernetes生态系统中的一个软件包管理工具,类似Ubuntu的APT、CentOS的YUM、或Python的pip一样,专门负责管理Kubernetes的应用资源。使用Helm可以对Kubernetes应用进行统一打包、分发、安装、升级以及回退等操作。

- Helm的获取、安装请参考: https://helm.sh/docs/intro/install/
- Helm的其他信息请参考: https://github.com/helm/helm

Helm在安装huawei-csi-controller时,将在指定命名空间的Deployment类型的工作负载中部署以下组件:

- huawei-csi-driver: 华为CSI驱动。
- Kubernetes External Provisioner: 用于提供卷。
- Kubernetes External Attacher: 用于挂载卷。
- (可选)Kubernetes External Snapshotter:提供快照支持(作为CRD安装)。
- Kubernetes External Resizer: 用于扩容卷。

Helm在安装huawei-csi-node时,将在指定命名空间的DaemonSet类型的工作负载中部署以下组件:

huawei-csi-driver: 华为CSI驱动。

Kubernetes Node Registrar: 处理驱动程序注册。

## 4.1.1 准备 values.yaml 文件

在使用Helm安装CSI时,需要您根据部署时对接的华为存储以及需要使用的特性准备 values.yaml文件。华为CSI已经在软件包的helm/esdk目录下提供了values.yaml模板文件。本章节将详细说明values.yaml中的配置项以及典型场景下的后端配置示例。

在values.yaml需要配置如下内容:

- backends配置项说明,典型场景的backends配置请参考如下示例
  - 配置多个不同的华为存储后端
  - 配置多个不同的租户/账户的华为存储后端
  - 配置iSCSI协议类型的存储后端
  - 配置FC协议类型的存储后端
  - 配置NVMe over RoCE协议类型的存储后端
  - 配置NVMe over FC协议类型的存储后端
  - 配置NFS协议类型的存储后端
  - 配置SCSI协议类型的存储后端
  - 配置DPC协议类型的存储后端
  - 配置双活类型的存储后端
- images配置项说明
- sidecar配置项说明
- csi\_driver配置项说明
- Kubernetes配置项说明
- sidecarParameters配置项说明( 可选 )
- 其他配置项

#### Kubernetes 配置项说明

values.yaml中的Kubernetes配置项主要配置华为CSI运行时的上下文信息。需要配置的参数如下:

表 4-1 Kubernetes 配置项说明

参数	描述	必选参 数	默认值
kubernetes.names pace	华为CSI运行时所在 Kubernetes命名空间,支持 用户自定义。名称必须由小 写字母、数字和"-"组成, 例如: my-name、123- abc。	是	huawei-csi

#### 须知

我们强烈建议为华为CSI使用单独的命名空间,并且不推荐使用系统自带的"default"、"kube-system"和"kube-public"命名空间。

请确保此处填入的命名空间在Kubernetes上已经存在,如果不存在请使用如下命令创建对应的命名空间。本例中,华为CSI运行的命名空间为"huawei-csi"。

# kubectl create namespace huawei-csi

## images 配置项说明

values.yaml中的images配置项主要配置华为CSI运行时依赖的组件镜像信息。需要配置的参数如下:

表 4-2 images 配置项说明

参数	描述	必选参 数	默认值
images.sidecar.live nessProbe	<b>livenessprobe</b> sidecar镜像。	是	k8s.gcr.io/sig- storage/ livenessprobe:v2.5. 0
images.sidecar.pro visioner	<b>csi-provisioner</b> sidecar镜 像。	是	k8s.gcr.io/sig- storage/csi- provisioner:v3.0.0
images.sidecar.att acher	<b>csi-attacher</b> sidecar镜像。	是	k8s.gcr.io/sig- storage/csi- attacher:v3.4.0
images.sidecar.resi zer	<b>csi-resizer</b> sidecar镜像。	是	k8s.gcr.io/sig- storage/csi- resizer:v1.4.0
images.sidecar.sna pshotter	<b>csi-snapshotter</b> sidecar镜 像。	是	k8s.gcr.io/sig- storage/csi- snapshotter:v4.2.1
images.sidecar.sna pshotController	<b>snapshot-controller</b> sidecar 镜像。	是	k8s.gcr.io/sig- storage/snapshot- controller:v4.2.1
images.sidecar.reg istrar	<b>csi-node-driver-registrar</b> sidecar镜像。	是	k8s.gcr.io/sig- storage/csi-node- driver- registrar:v2.3.0
images.huaweiCSI Service	huawei-csi镜像。	是	-

huaweiCSIService参数的值,请参考**3.3 上传华为CSI镜像**章节的说明,使用您的最终生成镜像的名称和版本。如本例中huawei-csi镜像名称为"huawei-csi:3.2.0"。

其他sidecar镜像参数,请参考**3.4 检查CSI依赖的镜像**章节的说明,使用您的最终上传的镜像的名称和版本。

#### sidecar 配置项说明

本章节详细描述在不同的Kubernetes版本下的sidecar如何配置。

- 如果您使用的Kubernetes版本高于或等于V1.17.0,请跳过本章节。
- 如果您使用的Kubernetes版本低于V1.17.0,可以根据以下步骤完成配置。
- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- **步骤2** 执行**kubectl get node**命令,查看Kubernetes版本信息。若Kubernetes版本高于或等于V1.17.0,请跳过本章节。

# kubectl get node

NAME STATUS ROLES AGE VERSION master Ready control-plane,master 23d v1.16.0

**步骤3** 执行**cd** *helm/esdk* 命令,进入到helm/esdk的工作目录下,执行**vi** *values.yaml* 命令 修改values.yaml文件中sidecar配置,可以参考如下配置示例:

images:

# The image name and tag for the Huawei CSI Service container

# Replace the appropriate tag name huaweiCSIService: huawei-csi:3.2.0

# The image name and tag for the sidecars. These must match the appropriate Kubernetes version. sidecar:

attacher: quay.io/k8scsi/csi-attacher:v1.2.1 provisioner: quay.io/k8scsi/csi-provisioner:v1.4.0

resizer: k8s.gcr.io/sig-storage/csi-resizer:v1.4.0

registrar: k8s.gcr.io/sig-storage/csi-node-driver-registrar:v2.3.0 livenessProbe: k8s.gcr.io/sig-storage/livenessprobe:v2.5.0

**步骤4** 执行**vi** *values.yaml* 命令修改values.yaml文件,将snapshot.enable设置为false,可以参考如下配置示例:

# The kubernetes version is lower than 1.17, please set it to false snapshot:
enable: false

**步骤5** 执行**cd** *helm/esdk* 命令,进入到helm/esdk的工作目录下,移除crds文件夹,命令如下:

# rm -r ./crds

步骤6 在helm/esdk的工作目录下,执行ls命令,如果目录结构和下面一致,则修改完成。

# ls

Chart.yaml templates values.yaml

----结束

#### csi\_driver 配置项说明

csi\_driver配置项包括了华为CSI运行时的基本配置,如华为驱动名称、多路径类型等配置信息。需要配置的参数如下:

表 4-3 csi\_driver 配置项说明

参数	描述	必选参数	默认值	配置建议
csi_driver.driv erName	注册的驱动名 称。	是	csi.huawei.com	直接使用默认 值。
csi_driver.end point	通信端点。	是	/csi/csi.sock	直接使用默认 值。
csi_driver.con nectorThread s	最大并发扫盘/ 卸盘数。参数 格式为整型, 支持范围为 1~10。	是	4	该值设置越大,同一时间单个节点中的针对多路径的扫盘、卸盘并发操作就越多。在使用DM-Multipath时,并发数过大可能会导致未知问题,影响整体时间。
csi_driver.vol umeUseMulti path	是否使用多路 径软件。参数 格式为布尔 值。	是	true	强烈建议开启多 路径软件,以增 强存储链路的冗 余度和性能。
csi_driver.scsi MultipathTyp e	存储协议为fc/iscsi时,使用的多路径软件。支持配置如下参数: DM-multipath HW-UltraPath HW-UltraPathNVMe	当 volumeUs eMultipat h为TRUE 时必填。	DM-multipath	建议使用DM-multipath取值。
csi_driver.nv meMultipath Type	存储协议为 roce/fc-nvme 时,使用的多 路径软件。仅 支持配置HW- UltraPath- NVMe。	当 volumeUs eMultipat h为TRUE 时必填。	HW-UltraPath- NVMe	-

参数	描述	必选参数	默认值	配置建议
csi_driver.sca nVolumeTim eout	在主机上使用 DM-Multipath 多路径时,等 待多路径聚合 的超时时间, 支持范围为 1~600,单位 秒。	是	3	-
csi_driver.bac kendUpdateI nterval	后端能力的更 新时间间隔, 支持范围 60~600,单位 秒。	是	60	-
csi_driver.con trollerLoggin g.module	controller日志 记录类型。支 持配置如下参 数: • file • console	是	file	使用file选项时,日志将被保留在节点指定的目录下,当CSI所在的Pod被销毁时,日志任然被保留。使用console选项时,日志将被保留在CSI所在Pod的临时空间中,当CSI所在的Pod被销毁时,日志也随之被销毁。
csi_driver.con trollerLoggin g.level	controller日志 输出级别。支 持配置如下参 数: • debug • info • warning • error • fatal	是	info	-
csi_driver.con trollerLoggin g.fileDir	controller日志 在file输出模式 下的日志目 录。	是	/var/log/huawei	请确保该目录下 有足够的空间保 留日志。空间大 小建议不小于 200 MB。
csi_driver.con trollerLoggin g.fileSize	controller日志 在file输出模式 下单个日志文 件大小。	是	20M	-

参数	描述	必选参数	默认值	配置建议
csi_driver.con trollerLoggin g.maxBackup s	controller日志 在file输出模式 下日志文件备 份上限。	是	9	-
csi_driver.nod eLogging.mo dule	node日志记录 类型。支持配 置如下参数: • file • console	是	file	使用file选项时, 日志将被保留录 下,当CSI所在 Pod被销毁时, 日志任然 留。 使用console选保 留。 使用console选保 可由志将在Pod 的临时空间的中, 当CSI所在Pod 被销毁时,日志 的临时空间的日志 被销毁时,
csi_driver.nod eLogging.lev el	node日志输出 级别。支持配 置如下参数: • debug • info • warning • error • fatal	是	info	-
csi_driver.nod eLogging.file Dir	node日志在file 输出模式下的 日志目录。	是	/var/log/huawei	请确保该目录下 有足够的空间保 留日志。空间大 小建议不小于 200 MB。
csi_driver.nod eLogging.file Size	node日志在file 输出模式下单 个日志文件大 小。	是	20M	-
csi_driver.nod eLogging.ma xBackups	node日志在file 输出模式下日 志文件备份上 限。	是	9	-

## <u> 注意</u>

如果您的容器环境已经部署了华为CSI,请确保csi\_driver.driverName的设置和之前部署时的配置保持一致。否则会导致系统中已存在的有华为CSI发放的卷/快照无法被新部署的华为CSI管理。

#### backends 配置项说明

backends配置项包括了CSI插件需要管理的存储后端信息,以及对接这些存储后端时使用的协议、端口信息。存储后端可以为Kubernetes容器平台提供存储资源,如通过SAN或者NAS等方式为Kubernetes提供PV。

华为CSI支持接入多个存储后端,也支持在存储后端中配置如ALUA、双活等高级特性。本章节将详细对backends配置项进行说明。

backends需要配置的参数如下:

表 4-4 backends 配置项说明

参数	描述	必选参数	配置建议
backends.s torage	存储服务类型。      企业存储提供SAN存储时填写oceanstor-san。      企业存储提供NAS存储时填写oceanstor-nas。      分布式存储提供SAN存储时填写fusionstorage-san。      分布式存储提供NAS存储时填写fusionstorage-nas。	是	一个后端只允许提供一种存储服务。如果单套华为存储系统可以同时提供SAN和NAS的存储服务时,可以配置创建多个后端,每个后端使用不同的存储服务类型。
backends.n ame	存储后端名称。支持大小写字 母、数字和[]。 如果需要配置多个存储后端,请 保证存储后端名唯一。	是	-
backends.v storeName	存储侧的租户名称。当对接后端 是OceanStor V3/V5存储时,需 要在指定租户下发放资源时,需 要指定该参数。	条件必选	仅对接后端是OceanStor V3/V5且需要支持租户时, 需要指定该参数。
backends.a ccountNa me	存储侧的账户名称,当对接资源 时OceanStor Pacific NAS存储 时,需要在指定账户下发放NAS 资源时,需要指定该参数。	条件必选	仅对接后端是OceanStor Pacific NAS存储且需要支持 账号时,需要指定该参数。

参数	描述	必选参数	配置建议
backends.u rls	存储设备的管理URL。参数格式 为列表。支持按照域名或者IP +端口的方式进行配置。仅支持 IPv4。	是	当对接后端是OceanStor V6 或OceanStor Dorado V6存 储,需要在指定租户下发放 资源时,该参数配置为指定 租户的逻辑管理端口URL。
backends.p ools	存储设备的存储池。参数格式为列表。	是	-
backends.p arameters. protocol	存储协议。参数格式为字符串。     iscsi     fc     roce     fc-nvme     nfs     dpc     scsi	是	<ul> <li>使用iscsi时,请确保对接的计算节点已安装iSCSI客户端。</li> <li>使用nfs时,请确保对接的计算节点已安装NFS客户端工具。</li> <li>使用fc-nvme/roce时,请确保对接的计算节点已安装nvme-cli工具,且版本不低于1.9。</li> <li>使用dpc时,请确保对接的计算节点已安装DPC客户端,并已在待接入序储上添加为DPC计算节点。</li> <li>使用scsi时,请确保对接的计算节点已安装分布式存储VBS客户端。</li> </ul>
backends.p arameters. portals	业务访问端口。节点会使用该端口对存储资源进行读写访问。参数格式为字符串。 iscsi,roce协议支持配置多个端口,nfs协议仅支持配置一个端口,fc、fc-nvme、dpc协议无需配置业务端口,scsi协议的端口形式为字典格式,key为主机名称,value为IP地址,仅支持IPv4。	条件必选	使用租户/账户对接后端时, 此时portals必须配置为租 户/账户所拥有的逻辑端口信 息。
backends.p arameters. ALUA	存储后端ALUA参数配置。当工作节点使用操作系统原生多路径,且启用了ALUA时,需要进行配置。	条件必选	如果主机多路径配置启用了ALUA,请确保后端ALUA配置和主机的ALUA配置一致。 ALUA详细配置请参考8.1.1 通过Helm配置ALUA特性。

参数	描述	必选参数	配置建议
backends. metrovStor ePairID	双活租户Pair ID。 当需要创建PV在存储侧支持 NAS双活特性时,该字段必填。 此时需要填入待创建的PV所归 属的存储侧双活租户Pair ID。	条件必选	双活租户Pair ID请到 DeviceManager界面查询。 如果对接存储是OceanStor V3/V5系列时,还需要同时 填入 <b>vstoreName</b> 信息。
backends. metroBack end	双活对端的后端名称。参数格式为字符串。 当需要创建PV在存储侧支持 NAS双活特性时,该字段必填。 此时需要填入准备和当前后端组 成双活的另一个后端名称。	条件必选	组对的两个后端都必须将对 方名称填入。这两个后端组 成双活关系后,不允许再和 其他后端组成双活关系。
backends.s upportedT opologies	存储拓扑感知配置。参数格式为 列表类型的JSON。	条件必选	如果启用存储拓扑感知,需要配置该参数。具体请参考8.2.1 通过Helm配置存储拓扑感知。

## sidecarParameters 配置项说明 (可选)

values.yaml中的sidecarParameters配置项主要配置sidecar的参数,例如PV名称的前缀。支持配置的参数如下:

表 4-5 sidecarParameters 配置项说明

参数	描述	必选参 数	默认 值	备注
sidecarParamete r.provisioner.volu meNamePrefix	PV名称的前缀,默认值为pvc,即创建的PV名称为: pvc- <uuid>。前缀必须满足DNS 子域名的命名规则,且PV名称总长度不得超过253个字符。</uuid>	否	pvc	对应的provisioner 参数名称为: volume-name- prefix 详细配置请参考配置PV名称前缀。  • 对接后端键是 OceanStor V3/V5时,过5个 字符。  • 对接后端键是 OceanStor V3/V5 NAS可能 时,写及及时,写及及时,所写及是 一个。 一个,以后端包。 一个,对接后就是一个。 一个,对接后就是一个。 一个,对接后就是一个,对接后就是一个。 一个,对接后就是一个,可以是一个。 一个,可以是一个,可以是一个,可以是一个。

## 配置多个不同的华为存储后端

如果在一个Kubernetes集群中使用多套华为存储,或者同一套华为存储提供多种存储服务类型时,可以参考如下配置示例。

```
# An array of storages with the access info
backends:
 - storage: "oceanstor-nas"
  name: "nfs-155"
   - "https://192.168.129.155:8088"
  pools:
   - "StoragePool001"
  parameters:
   protocol: "nfs"
   portals:
     - "192.168.128.155"
 - storage: "oceanstor-san"
  name: "iscsi-155"
  urls:
    - "https://192.168.129.155:8088"
  pools:
    - "StoragePool001"
  parameters:
   protocol: "iscsi"
   portals:
     - "192.168.128.156"
     - "192.168.128.157"
```

```
- storage: "fusionstorage-san"
 name: "iscsi-156"
 urls:
  - "https://192.168.129.156:8088"
  - "StoragePool001"
 parameters:
  protocol: "iscsi"
  portals:
    - "192.168.128.160"
   - "192.168.128.161"
- storage: "fusionstorage-nas"
 name: "nfs-156"
 urls:
  - "https://192.168.129.156:8088"
 pools:
  - "StoragePool001"
 parameters:
  protocol: "nfs"
  portals:
   - "192.168.128.170"
```

## 配置多个不同的租户/账户的华为存储后端

如果对接的华为存储是OceanStor V3/V5系列存储,且需要使用租户来对接存储做资源隔离时,需要为每个后端配置"vstoreName"信息。本示例中的场景为两个存储后端实际上是同一套华为OceanStor Dorado V5存储,但是使用了不同的租户对接。此时portals必须配置为租户所拥有的逻辑端口信息。

```
# An array of storages with the access info
backends:
 - storage: "oceanstor-nas"
  name: "nfs-vstore001"
  vstoreName: "vstore001"
    - "https://192.168.129.155:8088"
  pools:
    "StoragePool001"
  parameters:
   protocol: "nfs"
    portals:
     - "10.168.129.100"
 - storage: "oceanstor-nas"
  name: "nfs-vstore002"
  vstoreName: "vstore002"
  urls:
    - "https://192.168.129.155:8088"
  pools:
    - "StoragePool001"
  parameters:
    protocol: "nfs"
   portals:
     - "10.168.129.101"
```

如果对接的华为存储是OceanStor V6或OceanStor Dorado V6系列存储,且需要使用租户来对接存储做资源隔离时,需要为每个后端配置对应租户的指定租户的逻辑管理端口URL。本示例中的场景为两个存储后端实际上是同一套华为OceanStor Dorado V6存储,但是使用了不同的租户对接。此时portals必须配置为租户所拥有的逻辑端口信息。

```
# An array of storages with the access info backends:
- storage: "oceanstor-nas"
name: "nfs-vstore001"
```

```
- "https://192.168.129.155:8088"
  - "StoragePool001"
 parameters:
  protocol: "nfs"
  portals:
   - "10.168.129.100"
- storage: "oceanstor-nas"
name: "nfs-vstore002"
urls:
  - "https://192.168.129.156:8088"
pools:
  - "StoragePool001"
parameters:
  protocol: "nfs"
  portals:
   - "10.168.129.101"
```

如果对接的华为存储是OceanStor Pacific系列存储,且需要使用账号来对接存储做资源隔离时,需要为每个后端配置"accountName"信息。本示例中的场景为两个存储后端实际上是同一套华为OceanStor Pacific存储,但是使用了不同的账号对接。此时portals必须配置为账户所拥有的逻辑端口信息。

```
# An array of storages with the access info
backends:
 - storage: "fusionstorage-nas"
  name: "nfs-account001"
  accountName: "***"
  urls:
   - "https://192.168.129.155:8088"
  pools:
    - "StoragePool001"
  parameters:
   protocol: "nfs"
    portals:
     - "10.168.129.100"
 - storage: "fusionstorage-nas"
  name: "nfs-account002"
  accountName: "***"
    - "https://192.168.129.156:8088"
    - "StoragePool001"
  parameters:
    protocol: "nfs"
   portals:
     - "10.168.129.101"
```

## 配置 iSCSI 协议类型的存储后端

本示例展示了如何为华为企业存储和分布式存储配置iSCSI协议类型的后端。

```
backends:
- storage: "oceanstor-san"
name: "dorado-iscsi-155"
urls:
- "https://192.168.129.155:8088"
- "https://192.168.129.156:8088"
pools:
- "StoragePool001"
parameters:
protocol: "iscsi"
portals:
- "192.168.128.120"
- "192.168.128.121"
```

```
- storage: "fusionstorage-san"
name: "pacific-iscsi-125"
urls:
- "https://192.168.129.125:28443"
- "https://192.168.129.126:28443"
pools:
- "StoragePool001"
parameters:
protocol: "iscsi"
portals:
- "192.168.128.122"
- "192.168.128.123"
```

## 配置 FC 协议类型的存储后端

本示例展示了如何为华为企业存储配置FC协议类型的后端。

```
backends:
- storage: "oceanstor-san"
name: "fc-155"
urls:
- "https://192.168.129.155:8088"
- "https://192.168.129.156:8088"
pools:
- "StoragePool001"
parameters:
protocol: "fc"
```

## 配置 NVMe over RoCE 协议类型的存储后端

本示例展示了如何为华为企业存储配置NVMe over RoCE协议类型的后端。

```
backends:
- storage: "oceanstor-san"
name: "roce-155"
urls:
- "https://192.168.129.155:8088"
- "https://192.168.129.156:8088"
pools:
- "StoragePool001"
parameters:
protocol: "roce"
portals:
- "192.168.128.120"
- "192.168.128.121"
```

## 配置 NVMe over FC 协议类型的存储后端

本示例展示了如何为华为企业存储配置NVMe over FC协议类型的后端。

```
backends:
- storage: "oceanstor-san"
# support upper&lower characters, numeric and [-_].
name: "fc-nvme-155"
urls:
- "https://192.168.129.155:8088"
- "https://192.168.129.156:8088"
pools:
- "StoragePool001"
parameters:
protocol: "fc-nvme"
```

## 配置 NFS 协议类型的存储后端

本示例展示了如何为华为企业存储和分布式存储配置NFS协议类型的后端。

```
backends:
 - storage: "oceanstor-nas"
  name: "nfs-155"
  urls:
    - "https://192.168.129.155:8088"
   - "https://192.168.129.156:8088"
  pools:
    - "StoragePool001"
  parameters:
   protocol: "nfs"
   portals:
     - "192.168.128.155"
 - storage: "fusionstorage-nas"
  name: "nfs-126"
  urls:
    - "https://192.168.129.125:28443"
    - "https://192.168.129.126:28443"
  pools:
    - "StoragePool001"
  parameters:
    protocol: "nfs"
   portals:
     - "192.168.128.123"
```

## 配置 SCSI 协议类型的存储后端

本示例展示了如何为华为分布式存储配置SCSI协议类型的后端。

```
backends:
- storage: "fusionstorage-san"
name: "scsi-155"
urls:
- "https://192.168.129.155:28443"
pools:
- "StoragePool001"
parameters:
protocol: "scsi"
portals:
- {"hostname01": "192.168.125.21"}
- {"hostname02": "192.168.125.22"}
```

## 配置 DPC 协议类型的存储后端

本示例展示了如何为华为分布式存储配置DPC协议类型的后端。

```
backends:
- storage: "fusionstorage-nas"
name: "dpc-155"
urls:
- "https://192.168.129.155:28443"
- "https://192.168.129.156:28443"
pools:
- "StoragePool001"
parameters:
protocol: "dpc"
```

## 配置双活类型的存储后端

CSI支持在对接OceanStor V6或OceanStor Dorado V6时,在存储侧发放NFS类型的双活卷。本示例展示了如何为华为OceanStor V6或OceanStor Dorado V6存储配置双活类型的后端。

```
backends:
- storage: "oceanstor-nas"
name: "nfs-hypermetro-155"
```

```
- "https://192.168.129.155:8088"
  - "https://192.168.129.156:8088"
pools:
 - "StoragePool001"
metrovStorePairID: "f09838237b93c000"
metroBackend: "nfs-hypermetro-157"
parameters:
 protocol: "nfs"
  portals:
   - "192.168.129.155"
- storage: "oceanstor-nas"
name: "nfs-hypermetro-157"
 - "https://192.168.129.157:8088"
  - "https://192.168.129.158:8088"
pools:
 - "StoragePool001"
metrovStorePairID: "f09838237b93c000"
metroBackend: "nfs-hypermetro-155"
parameters:
  protocol: "nfs"
  portals:
   - "192.168.129.157"
```

#### 须知

- 配置NAS双活前,需要在两台存储设备之间配置双活关系,包含远端设备、双活域等,仅支持文件系统双活域工作模式为双活模式,配置操作请参考对应存储型号的产品文档。
- 对接NAS双活后端的账号必须为存储租户的租户管理员账号。

## 其他配置项

其他配置项包括了CSI插件某些特性的开关或者镜像获取策略。

#### 表 4-6 其他配置项说明

参数	描述	必选参数	默认值
sidecarImagePullP olicy	sidecar镜像的拉取 策略。	是	IfNotPresent
huaweiImagePullP olicy	huawei-csi镜像的 拉取策略。	是	IfNotPresent
snapshot.enable	是否开启快照特 性。要求 Kubernetes版本高 于v1.17。	是	true

#### 须知

当开启snapshot.enable后,该参数配置为true。这种情况下,在执行**helm install**命令时,将自动读取"helm/crd"目录下的卷快照CRD,并将安装快照CRD资源。

## 4.1.2 安装华为 CSI

## 前提条件

- 华为CSI镜像已制作完成,并且按照**3.3 上传华为CSI镜像**章节说明,上传到镜像仓库或者导入到所有节点。
- 华为CSI安装运行所依赖的组件镜像都已经上传到镜像仓库或者导入到所有节点。 具体信息请参考**3.4 检查CSI依赖的镜像**章节说明。
- 华为CSI运行所依赖的卷快照组件CRD已经安装。具体信息请参考3.5 检查卷快照 依赖组件章节说明。
- 如果您准备使用多路径联通华为存储,请确保所有计算节点上已安装多路径软件,具体信息请参考3.6 检查主机多路径配置章节说明。
- 容器管理平台已经安装部署了Helm 3。
- 已经准备安装CSI所需的values.yaml文件。具体信息请参考4.1.1 准备 values.yaml文件章节说明。
- Kubernetes的所有worker节点与待接入的存储设备的业务IP地址通信正常,iSCSI 场景下允许使用ping命令进行连通性校验。
- Kubernetes的所有worker节点已安装对应协议所需要的软件客户端,如iSCSI客户端、NFS客户端等。
- 已经在需要对接的华为存储上创建了对接华为CSI所需要的账号,具体信息请参考 3.7 检查华为存储上的账号配置章节说明。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 将Kubernetes CSI组件包中的"helm"目录拷贝到master节点的任意目录下。Helm工具路径请参见表3-1。
- 步骤3 进入到helm/esdk的工作目录下。

# cd helm/esdk

**步骤4** 执行**helm install** *helm-huawei-csi* ./ **-n** *huawei-csi* --**create-namespace**命令安装 华为CSI。

其中,helm-huawei-csi为自定义的Helm Chart名称,./表示使用当前目录下的Helm工程,huawei-csi为自定义的Helm Chart命名空间。

# helm install helm-huawei-csi ./ -n huawei-csi --create-namespace

NAME: helm-huawei-csi

LAST DEPLOYED: Wed Jun 8 11:50:28 2022

NAMESPACE: huawei-csi-helm-chart

STATUS: deployed REVISION: 1

TEST SUITE: None

**步骤5** 将华为CSI组件包中的secretGenerate工具拷贝到master节点的任意目录下,工具路径参见表3-1。

步骤6 使用加密工具填写存储设备的用户名和密码。

#### 须知

- 如果您对接的存储是OceanStor Dorado V6或OceanStor V6,且使用租户对接华为存储后端时,请使用租户管理员账号信息作为存储设备的用户名和密码。
- 华为CSI将存储设备的用户名及密码等信息存放在Kubernetes的secret资源中,secret资源的安全性提升建议参考Kubernetes的官方文档: encrypt-data
- 华为CSI中所有Kubernetes资源均使用声明式对象配置方式进行创建,此操作会在每个对象上设置 kubectl.kubernetes.io/last-applied-configuration: '{...}' 注解,该注解值中会包含用来创建对象的配置文件的内容,社区说明: https://github.com/pulumi/pulumi-kubernetes/issues/1118
- 1. 执行**chmod +x secretGenerate**命令,给secretGenerate添加可执行权限。 # chmod +x secretGenerate
- 2. 执行**./secretGenerate** --namespace=huawei-csi --logFileDir=/var/log/huawei 命令,运行secretGenerate工具,namespace的参数可修改为实际使用的命名空间,不使用该参数则默认使用huawei-csi命名空间,logFileDir参数可修改为实际使用的日志目录,不使用该参数则默认使用/var/log/huawei目录,根据界面提示输入需要配置backend的编号。Configured为false时表示这个backend尚未配置,为true时表示这个backend已配置。

3. 根据界面提示输入用户名和密码来创建secret对象。

Name:strage-backend-03

Urls:[https://192.168.125.27:8088]

Please enter this backend user name: admin

Please enter this backend password:

Verifying user name and password. Please wait.....

The acount information of the backend strage-backend-03 has been configured successfully.

4. 配置完成后,输入exit退出并保存配置。

Please enter the backend number to configure (Enter 'exit' to exit): *exit* Saving configuration. Please wait............
The configuration is saved successfully.

5. 使用kubectl get secret -n huawei-csi | grep huawei-csi-secret命令检查 secret对象是否已创建成功。

```
# kubectl -n huawei-csi get secret huawei-csi-secret
NAME TYPE DATA AGE
huawei-csi-secret Opaque 1 8d
```

步骤7 完成huawei-csi服务部署后,可执行kubectl get pod -n huawei-csi命令检查服务是否启动。

```
# kubectl get pod -n huawei-csi
NAME READY STATUS RESTARTS AGE
huawei-csi-controller-764bd64c97-kr2nm 7/7 Running 0 144m
huawei-csi-node-7m48s 3/3 Running 0 144m
```

#### ----结束

## 4.2 手动编写 ConfigMap 文件部署华为 CSI

本章节主要介绍在没有Helm的情况下,如何通过手动编写所有资源配置文件(如 Config.yaml等)完成对华为CSI的安装。如果您以及根据**4.1 使用Helm安装华为CSI**的说明安装了CSI,请跳过本章节。

## **/ 注意**

强烈建议您使用Helm对华为CSI进行安装部署和后续的升级。手动编写资源配置文件部署华为CSI将在后续版本中不再支持。

## 4.2.1 创建华为 CSI 运行时需要的 ConfigMap

ConfigMap 是一种 API 对象,用来将非机密性的数据保存到键值对中。华为CSI启动运行时,需要使用在本章节创建的ConfigMap中包含的信息作为重要的运行参数。

## 4.2.1.1 通过 iSCSI 对接企业存储 SAN

当您需要通过iSCSI对接企业存储SAN配置时,执行此操作。

## 前提条件

- Kubernetes的所有worker节点已安装iSCSI客户端。
- Kubernetes的所有worker节点与待接入的存储设备业务IP通信正常(华为CSI使用ping命令判断)。
- 如果是多路径组网,请确保所有worker节点上已安装多路径软件,详细操作请参考3.6 检查主机多路径配置。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-oceanstor-iscsi.yaml示例文件 )。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-7。

```
kind: ConfigMap
apiVersion: v1
metadata:
name: huawei-csi-configmap
namespace: huawei-csi
data:
csi.json: |
{
    "backends": [
```

## 表 4-7 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入 后端存储设备 列表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-8。

## 表 4-8 后端存储设备配置项描述

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存储 设备的类型。	对接企业存储SAN场 景,固定填写 "oceanstor-san"。
name	字符串	存储后端名。	自定义字符串,支持大小写字母、数字、中划线组合。 说明 如果需要配置多个存储后端,请保证存储后端名唯一。
urls	列表	必填,待接入的存储设备的管理URL。	支持同一存储设备的一个或多个管理URL,用逗号分割,当前仅支持IPv4。例如:https://192.168.125.20:8088说明 1个存储设备有多个控制器,每个控制器有一个管理URL,所以1个存储设备存在多个管理URL。
pools	列表	必填,被使用的待 接入存储设备上的 存储池名称。	支持同一存储设备上的 一个或多个存储池, 用逗 号分割。 可通过登录 DeviceManager获取支 持块存储服务的存储 池。

配置项	参数格式	描述	备注
parameters	字典	必填,iSCSI场景的 可变参数。	iSCSI场景protocol参数 固定填写"iscsi"。
			portals参数填写待接入 存储设备的iSCSI业务IP 地址,多个业务IP地址 请用逗号分割。
			iSCSI业务IP地址可通过登录DeviceManager获取。以OceanStorDorado 6.x系列为例:在DeviceManager管理界面,选择"服务>网络>逻辑端口",获取数据协议类型为iSCSI的IP(其它系列请参照相应的操作说明进行获取)。

# kubectl create -f huawei-csi-configmap.yaml

步骤5 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi | configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.2 通过 FC 对接企业存储 SAN

当您需要通过FC对接企业存储SAN配置时,请执行此操作。

## 限制条件

当您需要通过FC对接企业存储SAN配置时,需要保证主机侧没有盘符残留,如果有盘符残留,请参考10.4 集群中worker节点宕机并恢复后,Pod完成failover,但是Pod所在源主机出现盘符残留进行盘符清理。

## 前提条件

- 如果是多路径组网,请确保Kubernetes的所有worker节点上已安装多路径软件, 详细请参见3.6 检查主机多路径配置。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

- 步骤2 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。
  # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-oceanstor-fc.yaml示例文件 )。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-9。

#### 表 4-9 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入后 端存储设备列 表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-10。

## 表 4-10 后端存储设备配置项描述

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存储设 备的类型。	对接企业存储SAN场 景,固定填写 "oceanstor-san"。
name	字符串	存储后端名。	自定义字符串,支持大 小写字母、数字、中划 线组合。 <b>说明</b> 如果需要配置多个存储后 端,请保证存储后端名唯 一。

配置项	参数格式	描述	备注
urls	列表	必填,待接入的存储 设备的管理URL。	支持同一存储设备的一 个或多个管理URL,用 逗号分割,当前仅支持 IPv4。例如:https:// 192.168.125.20:8088
			说明 1个存储设备有多个控制 器,每个控制器有一个管 理URL,所以1个存储设 备存在多个管理URL。
pools	列表	必填,被使用的待接 入存储设备上的存储 池名称。	支持同一存储设备上的 一个或多个存储池, 用逗 号分割。 可通过登录 DeviceManager获取支 持块存储服务的存储 池。
parameters	字典	必填,FC场景的可 变参数。	FC场景protocol参数固定填写"fc"。

# kubectl create -f huawei-csi-configmap.yaml

步骤5 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi | configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.3 通过 NFS 对接企业存储 NAS

当您需要通过NFS对接企业存储NAS配置时,执行此操作。

#### 前提条件

- Kubernetes的所有worker节点已安装NFS客户端工具。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

**步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。
# vi huawei-csi-configmap.yaml

步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-

configmap-oceanstor-nfs.yaml示例文件 )。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考<mark>表4-11</mark>。

## 表 4-11 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入后 端存储设备列 表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-12。

## 表 4-12 后端存储设备配置项描述

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存储 设备的类型。	对接企业存储NAS场景 固定填写"oceanstor- nas"。
name	字符串	存储后端名。	自定义字符串,支持大 小写字母、数字、中划 线组合。 <b>说明</b> 如果需要配置多个存储后 端,请保证存储后端名唯 一。

配置项	参数格式	描述	备注
urls	列表	必填,待接入的存储设备的管理URL。	支持同一存储设备的一个或多个管理URL,用逗号分割,当前仅支持IPv4。例如: https://192.168.125.20:8088说明 1个存储设备有多个控制器,每个控制器有一个管理URL,所以1个存储设备存在多个管理URL。
pools	列表	必填,被使用的待 接入存储设备上的 存储池名称。	支持同一存储设备上的 一个或多个存储池, 用逗 号分割。 可通过登录 DeviceManager获取支 持文件存储服务的存储 池。
parameters	字典	必填,NFS场景的可变参数。	protocol参数固定填写 "nfs"。 portals: 指定存储的逻辑端口IP或者DNS Zone,只支持配置一个。 逻辑端口IP地址可通过登录DeviceManager获取。以OceanStor Dorado 6.x系列为例: 在DeviceManager管理界面,选择"服务 > 网络 > 逻辑端口",获取数据协议类型为NFS的IP(其它系列请参照相应的操作说明进行获取)。

# kubectl create -f huawei-csi-configmap.yaml

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.4 通过 NVMe over RoCE 对接企业存储 SAN

当您需要通过NVMe over RoCE对接企业存储SAN配置时,执行此操作。

## 前提条件

- Kubernetes的所有worker节点与待接入的存储设备业务IP通信正常(华为CSI使用ping命令判断)。
- Kubernetes的所有worker节点已安装nvme-cli工具,且版本不低于1.9。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。
- 如果是多路径组网,请确保Kubernetes的所有worker节点上已安装多路径软件, 详细请参见3.6 检查主机多路径配置。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-oceanstor-roce.yaml示例文件 )。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-13。

#### 表 4-13 配置项描述

配置项	参数格式	描述	备注
data."csi.json".b ackends	列表	必填,待接入 后端存储设备 列表。	配置的后端存储设备数量没有限制。 单个后端存储设备支持配置的字段,请参考表4-14。

表 4-14 后端存储设备配置项描述

置项	参数格式	描述	备注	
stor age	字符串	必填,待接入存 储设备的类型。	对接企业存储SAN场景固定填写"oceanstor- san"。	
na me	字符串	存储后端名。	自定义字符串,支持大小写字母、数字、中划线组合。 <b>说明</b> 如果需要配置多个存储后端,请保证存储后端名唯一。	
urls	列表	必填,待接入的 存储设备的管理 URL。	支持同一存储设备的一个或多个管理URL,用逗号分割,当前仅支持IPv4。例如: https://192.168.125.20:8088  说明 1个存储设备有多个控制器,每个控制器有一个管理URL,所以1个存储设备存在多个管理URL。	
poo ls	列表	必填,被使用的 待接入存储设备 上的存储池名 称。	支持同一存储设备上的一个或多个存储池, 用逗号分割。 可通过登录DeviceManager获取存储池。	
par am eter s	字典	必填,NVMe over RoCE场景 的可变参数。	NVMe over RoCE场景protocol参数固定填写 "roce"。 portals参数填写存储的数据协议类型为NVMe over RoCE的逻辑端口IP,用逗号分割。逻辑端口IP地址可通过登录DeviceManager获取。以OceanStor Dorado 6.x系列为例:在 DeviceManager管理界面,选择"服务 > 网络 > 逻辑端口",获取数据协议类型为NVMe over RoCE的IP(其它系列请参照相应的操作说明进行获取)。	

# kubectl create -f huawei-csi-configmap.yaml

步骤5 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi | configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.5 通过 NVMe over FC 对接企业存储 SAN

当您需要通过NVMe over FC对接企业存储SAN配置时,请执行此操作。

## 限制条件

当您需要通过NVMe over FC对接企业存储SAN配置时,需要保证主机侧没有盘符残留,如果有盘符残留,请参考10.4 集群中worker节点宕机并恢复后,Pod完成failover,但是Pod所在源主机出现盘符残留进行盘符清理。

## 前提条件

- Kubernetes的所有worker节点已安装nvme-cli工具,且版本不低于1.9。
- 如果是多路径组网,请确保Kubernetes的所有worker节点上已安装多路径软件, 详细请参见3.6 检查主机多路径配置。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-oceanstor-fc-nvme.yaml示例文件)。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-15。

#### 表 4-15 配置项描述

配置项	参数 格式	描述	备注
data."csi.js	列表	必填,待接入	配置的后端存储设备数量没有限制。
on".backe		后端存储设备	单个后端存储设备支持配置的字段,请参考 <mark>表</mark>
nds		列表。	4-16。

表 4-16 后端存储设备配置项
------------------

配置 项	参数格式	描述	备注
stor age	字符串	必填,待接入存储 设备的类型。	对接企业存储SAN场景,固定填写"oceanstor-san"。
nam e	字符串	存储后端名。	自定义字符串,支持大小写字母、数字、中划线组合。 <b>说明</b> 如果需要配置多个存储后端,请保证存储后端名唯一。
urls	列表	必填,待接入的存储设备的管理URL。	支持同一存储设备的一个或多个管理URL,用逗号分割,当前仅支持IPv4。例如: https://192.168.125.20:8088  说明 1个存储设备有多个控制器,每个控制器有一个管理URL,所以1个存储设备存在多个管理URL。
pool s	列 表	必填,被使用的待 接入存储设备上的 存储池名称。	支持同一存储设备上的一个或多个存储池, 用逗号分割。 可通过登录DeviceManager获取支持块存储服务的存储池。
para met ers	字典	必填,NVMe over FC场景的可变参 数。	NVMe over FC场景protocol参数固定填写"fc-nvme"。

# kubectl create -f huawei-csi-configmap.yaml

步骤5 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi | configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.6 通过 SCSI 对接分布式存储 SAN

当您需要通过SCSI对接分布式存储SAN配置,执行此操作。

## 前提条件

- Kubernetes的所有worker节点已安装分布式存储VBS客户端。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件,huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-fusionstorage-scsi.yaml示例文件)。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-17。

#### 表 4-17 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入后端 存储设备列表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-18。

## 表 4-18 后端存储设备配置项描述

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存 储设备的类型。	对接分布式存储SAN场景固定填写"fusionstorage-san"。
name	字符串	存储后端名。	自定义字符串,支持大小写字 母、数字、中划线组合。 <b>说明</b>
			如果需要配置多个存储后端,请保 证存储后端名唯一。

配置项	参数格式	描述	备注
urls	列表	必填,待接入的 存储设备的管理 URL。	FusionStorage的管理URL, 只支持配置一个。
pools	列表	必填,被使用的 待接入存储设备 上的存储池名 称。	支持同一存储设备上的一个或多 个存储池, 用逗号分割。 可通过登录DeviceManager获 取存储池。
parameters	字典	必填,SCSI场景 的可变参数。	protocol参数固定填写 "scsi"。
			portals参数填写主机名称和VBS 节点IP Pair列表,参数格式为 [{"hostname":"*.*.*.*"}],其中 hostname为对应的worker节点 主机名称,"*.*.*."为分布式存储 块客户端对应的管理IP地址(当 前仅支持IPv4)。
			如果存在多个worker节点,则 相应地以字典格式配置多个,以 逗号隔开。
			例如示例中hostname01为 Kubernetes中Worker节点的主 机名称,192.168.125.21为该 Kubernetes中Worker节点创建 VBS后,在VBS中呈现的管理 IP。

# kubectl create -f huawei-csi-configmap.yaml

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.7 通过 iSCSI 对接分布式存储 SAN

当您需要通过iSCSI对接分布式存储SAN配置时,请执行此操作。

## 前提条件

- Kubernetes的所有worker节点已安装iSCSI客户端。
- Kubernetes的所有worker节点与待接入的存储设备业务IP通信正常(华为CSI使用ping命令判断)。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

● 如果是多路径组网,请确保Kubernetes的所有worker节点上已安装多路径软件, 详细请参见**3.6 检查主机多路径配置**。

## 注意事项

- Kubernetes的worker节点主机名称由数字、字母、"\_"、"-"、"-"、"."和":"组成,首字符只能是数字、字母或"\_",名称长度不超过31个字符。
- 仅FusionStorage 8.0.0及其以后版本支持iSCSI组网配置。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-fusionstorage-iscsi.yaml示例文件)。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-19。

#### 表 4-19 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入 后端存储设备 列表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-20。

表 4-20 后端存储设备配置项描述

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存储 设备的类型。	对接分布式存储SAN场 景固定填写 "fusionstorage-san"。
name	字符串	存储后端名。	自定义字符串,支持大 小写字母、数字、中划 线组合。 说明 如果需要配置多个存储后 端,请保证存储后端名唯 一。
urls	列表	必填,待接入的存 储设备的管理URL。	FusionStorage的管理 URL, 只支持配置一个。
pools	列表	必填,被使用的待 接入存储设备上的 存储池名称。	支持同一存储设备上的 一个或多个存储池, 用逗 号分割。 可通过登录 DeviceManager获取存 储池。
parameters	字典	必填,iSCSI场景的可变参数。	iSCSI场景protocol参数 固定填写"iscsi"。 portals参数填写指定存储设备的iSCSI业务IP地址,用逗号分割,可通过登录DeviceManager获取。 iSCSI业务IP地址可通过登录DeviceManager获取。以OceanStor Pacific系列为例:在DeviceManager管理界面,选择"资源 > 访问 > 业务网络"获取(其它系列请参照相应的操作说明进行获取)。

# kubectl create -f huawei-csi-configmap.yaml

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

-----结束

## 4.2.1.8 通过 NFS 对接分布式存储 NAS

当您需要通过NFS对接分布式存储NAS配置时,请执行此步骤。

## 前提条件

- Kubernetes的所有worker节点已安装NFS客户端工具。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-fusionstorage-nfs.yaml示例文件)。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细请参考表4-21。

#### 表 4-21 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入后端 存储设备列表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-22。

表 4-22 后端存储设备配置项描述

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存储 设备的类型。	对接分布式存储NAS场 景固定填写 "fusionstorage-nas"。
name	字符串	存储后端名。	自定义字符串,支持大小写字母、数字、中划线组合。 <b>说明</b> 如果需要配置多个存储后端,请保证存储后端名唯一。
urls	列表	必填,待接入的存 储设备的管理URL。	FusionStorage的管理 URL, 只支持配置一个。
pools	列表	必填,被使用的待 接入存储设备上的 存储池名称。	支持同一存储设备上的 一个或多个存储池, 用逗 号分割。 可通过登录 DeviceManager获取存 储池。
parameters	字典	必填,NFS场景的可变参数。	portals: 指定存储的逻辑 端口IP,可通过登录 DeviceManager获取, 当前只支持配置一个。 逻辑端口IP地址可通过 登录DeviceManager获取。以OceanStor Pacific系列为例: 在 DeviceManager管理界面,选择"资源>定好管理界面,选择"资源>访问>业务网络,单击一个zone的名称。在弹出的页面,单击"IP地址/掩码"列即为逻辑端口IP地址。其它系列请参照相应的操作说明进行获取。

# kubectl create -f huawei-csi-configmap.yaml

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.9 通过 DPC 对接分布式存储 NAS

当您需要通过DPC对接分布式存储NAS配置时,请执行此步骤。

## 前提条件

- Kubernetes的所有worker节点均已在待接入存储上添加为DPC计算节点。
- 已向管理员获取Kubernetes集群中任意master节点的IP地址、登录帐号和密码。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- 步骤2 执行**vi** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap.yaml文件。 # vi huawei-csi-configmap.yaml
- 步骤3 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示(您也可以参考软件包中deploy/huawei-csi-configmap/huawei-csi-configmap-fusionstorage-nfs.yaml示例文件)。请根据实际情况设置相关参数,并保存为yaml格式的文件,详细信息请参考表4-23。

#### 表 4-23 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backends	列表	必填,待接入后端 存储设备列表。	配置的后端存储设备数量没有限制。单个后端存储设备支持配置的字段,请参考表4-24。

表 4-24 后端存储设备配置项描	沭	
-------------------	---	--

配置项	参数格式	描述	备注
storage	字符串	必填,待接入存储 设备的类型。	对接分布式存储NAS场 景固定填写 "fusionstorage-nas"。
name	字符串	存储后端名。	自定义字符串,支持大小写字母、数字、中划线组合。 <b>说明</b> 如果需要配置多个存储后端,请保证存储后端名唯一。
urls	列表	必填,待接入的存 储设备的管理URL。	FusionStorage的管理 URL, 只支持配置一个。
pools	列表	必填,被使用的待 接入存储设备上的 存储池名称。	支持同一存储设备上的 一个或多个存储池, 用逗 号分割。 可通过登录 DeviceManager获取存 储池名称。
parameters	字典	必填,DPC场景的 可变参数。	DPC场景protocol参数固 定填写为"dpc"。

# kubectl create -f huawei-csi-configmap.yaml

步骤5 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi | configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

----结束

## 4.2.1.10 通过租户/账户对接华为存储后端

如果对接的华为存储时需要使用租户/账户来对接存储做资源隔离时,需要为每个后端配置租户/账户信息。如果未配置租户/账户信息时,CSI系统将使用存储系统默认的租户/账户创建资源。

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

**步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,修改yaml文件。

如果租户使用的后端是OceanStor V3/V5存储,则在该后端配置中增加 "vstoreName"参数,参数值填写存储设备上的租户名称。

```
{
    "backends": [
    {
        ...
    "vstoreName": "***"
    }]
}
```

- 如果租户使用的后端是OceanStor 6.1或OceanStor Dorado 6.x存储,需要为该后端配置对应租户的指定租户的逻辑管理端口URL。请注意"urls"为租户的逻辑管理端口,"pools"和"portals"需要确保是当前租户可用的存储池和数据逻辑端口。
- 如果租户使用的后端是OceanStor Pacific存储,则在该后端配置中增加 "accountName"配置项,参数值为需要指定的帐户名称,此时该后端的portals 必须配置为帐户所拥有的逻辑端口。

```
{
    "backends":[
    {
        ...
        "accountName": "***"
    }]
}
```

# kubectl create -f huawei-csi-configmap.yaml

步骤4 执行kubectl get configmap -n huawei-csi | grep huawei-csi-configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

```
# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s
```

----结束

## 4.2.1.11 通过 CSI 在后端发放双活卷

CSI支持在对接华为OceanStor Dorado V6或OceanStor V6时,在存储侧发放NFS类型的双活卷。当您需要配置双活后端时,执行此操作。

## 须知

- 配置NAS双活前,需要在两台存储设备之间配置双活关系,包含远端设备、双活域等,仅支持文件系统双活域工作模式为双活模式,配置操作请参考对应存储型号的产品文档。
- 对接NAS双活后端的账号必须为存储租户的租户管理员账号。
- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,修改yaml文件。按**l**或**Insert**进入编辑状态,修改相关参数。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

在huawei-csi-configmap.yaml文件的backends配置段中,添加构成双活关系的两个后端,每个后端存储设备配置项可以参考**4.2.1.3 通过NFS对接企业存储NAS**。还需要额外增加双活场景下一些配置参数,详情请参考**表4-25**。

```
kind: ConfigMap
apiVersion: v1
```

#### 表 4-25 后端存储设备双活配置项描述

配置项	参数格式	描述	备注
vstoreName	字符串	条件必填 (OceanStor V3/V5存储必 填),租户名称。	仅租户用户支持配置NAS双活。
metrovStor ePairID	字符串	必填,租户归属的 双活租户Pair ID。	以OceanStor Dorado 6.x或 OceanStor 6.1为例,该值在存 储设备DeviceManager界面显示 为"ID"。
metroBacke nd	字符串	必填,双活对端的 后端名,两个双活 后端相互构成双活 关系。	后端hyperMetro1的双活对端是 hyperMetro2,相应的, hyperMetro2的双活对端是 hyperMetro1。

## **步骤3** 执行**kubectl create -f** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap。

# kubectl create -f huawei-csi-configmap.yaml

# 步骤4 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi-configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

```
# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap
huawei-csi-configmap 1 5s
```

----结束

## 4.2.1.12 通过 CSI 对接多个后端存储

Huawei CSI是可以支持配置多个后端。当您需要配置多个后端时,执行此操作。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 配置huawei-csi-configmap.yaml文件, huawei-csi-configmap.yaml文件的模板如回显所示。请根据实际情况设置相关参数,并保存为yaml格式的文件。

多个后端间使用","进行分割,每个后端内容可以参考**4.2.1 创建华为CSI运行时需要的ConfigMap**章节的说明。

```
kind: ConfigMap
apiVersion: v1
metadata:
 name: huawei-csi-configmap
 namespace: huawei-csi
data:
 csi.json: |
     "backends": [
        {
          "storage": "***",
          "name": "backend1",
        },
          "storage": "***",
           "name": "backend2",
        }
     ]
```

**步骤3** 执行**kubectl create -f** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap。

# kubectl create -f huawei-csi-configmap.yaml

步骤4 创建完成后,执行kubectl get configmap -n huawei-csi | grep huawei-csi | configmap命令,查看是否创建成功。显示如下回显时,则表示创建成功。

```
# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap
huawei-csi-configmap 1 5s
```

----结束

## 4.2.2 启动 huawei-csi 服务

本章节介绍如何启动huawei-csi服务。

## 前提条件

- 华为CSI镜像已制作完成,并且按照3.3 上传华为CSI镜像章节说明,上传到镜像仓库或者导入到所有节点。
- 华为CSI安装运行所依赖的组件镜像都已经上传到镜像仓库或者导入到所有节点。
   具体信息请参考3.4 检查CSI依赖的镜像章节说明。
- 华为CSI运行所依赖的卷快照组件CRD已经安装。具体信息请参考3.5 检查卷快照 依赖组件章节说明。
- 如果您准备使用多路径联通华为存储,请确保所有计算节点上已安装多路径软件,具体信息请参考3.6 检查主机多路径配置章节说明。
- Kubernetes的所有worker节点与待接入的存储设备的业务IP地址通信正常,iSCSI 场景下允许使用ping命令进行连通性校验。

- Kubernetes的所有worker节点已安装对应协议所需要的软件客户端,如iSCSI客户端、NFS客户端等。
- 已经在需要对接的华为存储上创建了对接华为CSI所需要的账号,具体信息请参考 3.7 检查华为存储上的账号配置章节说明。

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 将Kubernetes CSI组件包中的secretGenerate工具拷贝到master节点的任意目录下,工具路径参见表3-1。
- 步骤3 使用加密工具填写存储设备的用户名和密码。

#### 须知

- 如果您对接的存储是OceanStor Dorado V6或OceanStor V6,且使用租户对接华为存储后端时,请使用租户管理员账号信息作为存储设备的用户名和密码。
- 华为CSI将存储设备的用户名及密码等信息存放在Kubernetes的secret资源中,secret资源的安全性提升建议参考Kubernetes的官方文档: encrypt-data
- 华为CSI中所有Kubernetes资源均使用声明式对象配置方式进行创建,此操作会在每个对象上设置 kubectl.kubernetes.io/last-applied-configuration: '{...}' 注解,该注解值中会包含用来创建对象的配置文件的内容,社区说明: https://github.com/pulumi/pulumi-kubernetes/issues/1118
- 执行chmod +x secretGenerate命令,给secretGenerate添加可执行权限。 # chmod +x secretGenerate
- 2. 执行./secretGenerate --namespace=huawei-csi --logFileDir=/var/log/huawei 命令,运行secretGenerate工具,namespace的参数可修改为实际使用的命名空间,不使用该参数则默认使用huawei-csi命名空间,logFileDir参数可修改为实际使用的日志目录,不使用该参数则默认使用/var/log/huawei目录,根据界面提示输入需要配置backend的编号。Configured为false时表示这个backend尚未配置,为true时表示这个backend已配置。

```
# ./secretGenerate huawei-csi
Getting backend configuration information.....
Number Configured BackendName
                                        Urls
     false
              strage-backend [https://192.168.125.25:8088]
              strage-backend-02 [https://192.168.125.26:8088]
     false
     false
              strage-backend-03 [https://192.168.125.27:8088]
3
              strage-backend-04 [https://192.168.125.28:8088]
     false
              strage-backend-05 [https://192.168.125.29:28443]
     false
              strage-backend-06 [https://192.168.125.30:28443]
Please enter the backend number to configure (Enter 'exit' to exit):3
```

3. 根据界面提示输入用户名和密码来创建secret对象。

4. 配置完成后,输入exit退出并保存配置。

Please enter the backend number to configure (Enter 'exit' to exit): *exit* Saving configuration. Please wait............
The configuration is saved successfully.

5. 使用kubectl get secret -n huawei-csi | grep huawei-csi-secret命令检查 secret对象是否已创建成功。

# kubectl -n huawei-csi get secret huawei-csi-secret NAME TYPE DATA AGE huawei-csi-secret Opaque 1 8d

#### 步骤4 执行以下命令,创建RBAC权限。

# kubectl apply -f huawei-csi-rbac.yaml

#### 步骤5 启动controller服务。

1. 如果您使用的Kubernetes版本高于或等于V1.17.0,请执行**vi huawei-csi-controller-snapshot-v1.yaml**命令。如果您使用的Kubernetes版本低于V1.17.0,请执行**vi huawei-csi-controller.yaml**命令,修改yaml文件。按**I**或**Insert**进入编辑状态,修改以下相关参数。修改完成后,按**Esc**,并输入**:wq!**,保存修改。

#### □ 说明

(必选)示例yaml文件中huawei-csi-driver的参数image配置项,修改huawei-csi:\*.\*\*
 为3.3 上传华为CSI镜像章节中上传的华为CSI镜像<名称>:<版本号>。
 containers:

- name: huawei-csi-driver image: repo.huawei.com/huawei-csi:3.2.0

- (可选)示例yaml文件中metadata的参数namespace配置项表示huawei-csi-controller服务安装的命名空间,如需修改,请确保config.yaml、rbac.yaml、node.yaml文件中命名空间保持一致,修改方式请参考如下。metadata:

name: huawei-csi-controller namespace: huawei-csi

- 2. 如果Kubernetes版本低于V1.17.0,执行以下命令,启动controller服务。 # kubectl apply -f huawei-csi-controller.yaml
- 3. 如果Kubernetes版本高于或等于V1.17.0,执行以下命令,启动controller服务。 # kubectl apply -f huawei-csi-controller-snapshot-v1.yaml

#### 步骤6 启动node服务。

1. 执行**vi** *huawei-csi-node.yaml*命令,修改yaml文件。按**l**或**lnsert**进入编辑状态, 修改相关参数。修改完成后,按**Esc**,并输入**:wq!**,保存修改。

#### □ 说明

(必选)示例yaml文件中huawei-csi-driver的参数image配置项,修改huawei-csi:\*.\*.\*
 为3.3 上传华为CSI镜像章节中上传的华为CSI镜像<名称>:<版本号>。

#### containers:

- name: huawei-csi-driver image: huawei-csi:3.2.0
- (可选)示例yaml文件中metadata的参数namespace配置项表示huawei-csi-node服 务安装的命名空间,如需修改,请确保config.yaml、rbac.yaml、controller.yaml文件 中命名空间保持一致,修改方式请参考如下。

#### metadata:

name: huawei-csi-controller namespace: huawei-csi

- (可选)示例yaml文件中huawei-csi-driver配置参数args:--volume-use-multipath 表示默认开启多路径,如需修改,请参考如下。

#### args:

- "--endpoint=/csi/csi.sock"
- "--containerized"
- "--driver-name=csi.huawei.com"
- "--volume-use-multipath=false"
- (可选)示例yaml文件中huawei-csi-driver配置参数args: --connector-threads 表示 在主机上对盘符操作的并发量,整型,默认值为4,支持范围为1~10,如需修改,请参 考如下。

#### args:

- "--endpoint=/csi/csi.sock"
- "--containerized"
- "--driver-name=csi.huawei.com"
- "--volume-use-multipath=true"
- "--connector-threads=5"
- (可选)示例yaml文件中huawei-csi-driver配置参数args: --scan-volume-timeout 表示在主机上使用DM-Multipath多路径时,等待多路径聚合的超时时间,整型,默认值为3,支持范围为1~600,如需修改,请参考如下。

#### args:

- "--endpoint=/csi/csi.sock"
- "--containerized"
- "--driver-name=csi.huawei.com"
- "--volume-use-multipath=true"
- "--connector-threads=4"
- "--scan-volume-timeout=3"
- (可选)示例yaml文件中huawei-csi-driver配置参数args:对于企业存储,当--volume-use-multipath设置为true时,可针对不同组网进行多路径类型配置,详情可参考表4-26。

#### args:

- "--endpoint=/csi/csi.sock"
- "--containerized"
- "--driver-name=csi.huawei.com"
- "--connector-threads=4"
- "--volume-use-multipath=true"
- "--scsi-multipath-type=DM-multipath"
- "--nvme-multipath-type=HW-UltraPath-NVMe"

表 4-26 企业存储多路径配置参数说明

存储协议	参数名称	说明	备注
iSCSI/FC	scsi-multipath- type	取值可为: - DM-multipath - HW-UltraPath - HW- UltraPath- NVMe 默认值为DM- multipath。	<ul> <li>DM- multipath:操 作系统原生多 路径软件</li> <li>HW- UltraPath:华 为UltraPath多 路径软件</li> <li>HW-</li> </ul>
NVMe over RoCE/NVMe over FC	nvme- multipath-type	默认值为HW- UltraPath- NVMe, 仅支持 配置HW- UltraPath- NVMe。	UltraPath- NVMe:华为 UltraPath- NVMe多路径 软件

2. 执行以下命令,启动node服务。 # kubectl apply -f huawei-csi-node.yaml

# **步骤7** 完成huawei-csi服务部署后,可执行**kubectl get pod -A | grep huawei**命令检查服务是否启动。

# kubectl get pod -A | grep huawei huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 **Running** 0 14s huawei-csi huawei-csi-node-g6f7z 3/3 **Running** 0 14s

## ----结束

# **5** 卸载华为 CSI

本章节介绍如何卸载华为CSI。根据您安装时的方式,请使用不同的方式进行卸载。

- 如果安装华为CSI时使用的是Helm方式部署,请参考5.1 使用Helm卸载huaweicsi章节。
- 如果安装华为CSI时使用的是手动编写资源配置文件方式部署,请参考5.2 手动卸载huawei-csi章节。

## <u> 注意</u>

如果您不是出于升级的目的卸载华为CSI,请确保卸载华为CSI前已经在您的容器平台中将华为CSI发放的资源(PV、PVC、快照等)全部清理。否则一旦您卸载华为CSI后,这些资源将无法被自动调度、管理或者清理。

- 5.1 使用Helm卸载huawei-csi
- 5.2 手动卸载huawei-csi
- 5.3 卸载Snapshot依赖组件服务

## 5.1 使用 Helm 卸载 huawei-csi

如果安装华为CSI时使用的是Helm方式部署,可以参考本章节卸载华为CSI。

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

**步骤2** 执行helm uninstall *helm-huawei-csi* -n *huawei-csi*命令卸载华为CSI,*helm-huawei-csi是*自定义的Helm Chart名称,*huawei-csi是*该Helm Chart所在的命名空间。该卸载命令将会删除华为CSI的huawei-csi-controller、huawei-csi-node、huawei-csi-configmap和RBAC资源。

# helm uninstall helm-huawei-csi -n huawei-csi release "helm-huawei-csi" uninstalled

删除命令执行后,还需要检查卸载是否成功。

# helm list -n huawei-csi

NAME NAMESPACE REVISION UPDATED STATUS CHART APP VERSION

其中,huawei-csi为chart所在的命名空间。

如果回显为空,则表示服务删除成功。

#### 步骤3 删除secret对象。

# kubectl delete secret huawei-csi-secret -n huawei-csi

其中huawei-csi-secret是secret对象的名称,*huawei-csi*是secret对象所在的命名空间。

删除命令执行后,还需要检查删除是否成功。

# kubectl get secret huawei-csi-secret -n huawei-csi Error from server (NotFound): secrets "huawei-csi-secret" not found

如果命令回显提示"NotFound"表示huawei-csi-secret对象已成功删除。

步骤4 (可选)卸载快照依赖组件服务,请参考5.3 卸载Snapshot依赖组件服务进行操作。

步骤5 (可选)删除华为CSI镜像。

----结束

## 5.2 手动卸载 huawei-csi

如果安装华为CSI时使用的是手动编写资源配置文件方式,可以参考本章节卸载华为CSI。

## 5.2.1 卸载 huawei-csi-node 服务

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

**步骤2** 执行**kubectl delete daemonset huawei-csi-node -n** *huawei-csi*命令卸载huawei-csi-node服务,huawei-csi替换为华为CSI所在的命名空间。

# kubectl delete daemonset huawei-csi-node -n huawei-csi

步骤3 执行以下命令检查服务是否已成功卸载(如果提示NotFound错误,表示已成功卸载)。

# kubectl get daemonset huawei-csi-node -n huawei-csi

----结束

## 5.2.2 卸载 huawei-csi-controller 服务

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行kubectl delete deployment huawei-csi-controller -n huawei-csi命令卸载 huawei-csi-controller服务,huawei-csi替换为华为CSI所在的命名空间。

# kubectl delete deployment huawei-csi-controller -n huawei-csi

步骤3 执行以下命令检查服务是否已成功卸载(如果提示NotFound错误,表示已成功卸载)。

# kubectl get deployment huawei-csi-controller -n huawei-csi

-----结束

## 5.2.3 删除 huawei-csi-configmap 对象

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行kubectl delete configmap huawei-csi-configmap -n huawei-csi命令删除 configmap对象,其中huawei-csi-configmap是configmap的名称,huawei-csi是 configmap对象所在的命名空间。

# kubectl delete configmap huawei-csi-configmap -n huawei-csi

**步骤3** 执行以下命令检查对象是否已成功删除(如果提示NotFound错误,表示已成功删除)。

# kubectl get configmap huawei-csi-configmap -n huawei-csi

----结束

## 5.2.4 删除 huawei-csi-secret 对象

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例 ),通过管理IP地址,登录Kubernetes集群的任意 master节点 。

**步骤2** 执行**kubectl delete secret** *huawei-csi-secret* **-n** *huawei-csi*命令删除secret对象,其中huawei-csi-secret是secret对象的名称,huawei-csi是secret对象所在的命名空间。
# kubectl delete secret huawei-csi-secret -n huawei-csi

**步骤3** 执行以下命令检查secret对象是否删除成功,如果命令回显提示"NotFound"表示huawei-csi-secret对象已成功删除。

# kubectl get secret huawei-csi-secret -n huawei-csi
Error from server (NotFound): secrets "huawei-csi-secret" not found

----结束

## 5.2.5 删除 RBAC 权限

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 删除RBAC权限。

● 当huawei-csi版本号高于2.2.15,执行以下命令进行删除,-n参数为命名空间,请按实际修改。

# kubectl -n huawei-csi -l provisioner=csi.huawei.com delete ServiceAccount,role,rolebinding,ClusterRole,ClusterRoleBinding

- 当huawei-csi版本号低于或等于2.2.15,按照以下步骤进行删除。
  - a. 执行以下命令创建delete-huawei-csi-rbac.sh文件,-n的参数为命名空间,请 按实际修改,V3.0之前的CSI默认命名空间为kube-system。

# cat <<EOF > delete-huawei-csi-rbac.sh

kubectl delete ServiceAccount huawei-csi-controller -n huawei-csi

kubectl delete ServiceAccount huawei-csi-node -n huawei-csi

kubectl delete ClusterRole huawei-csi-attacher-runner -n huawei-csi

kubectl delete ClusterRole huawei-csi-driver-registrar-runner -n huawei-csi

kubectl delete ClusterRole huawei-csi-provisioner-runner -n huawei-csi

kubectl delete ClusterRole huawei-csi-resizer-runner -n huawei-csi

kubectl delete ClusterRole huawei-csi-snapshotter-runner -n huawei-csi

kubectl delete ClusterRole snapshot-controller-runner -n huawei-csi

kubectl delete ClusterRoleBinding huawei-csi-attacher-role -n huawei-csi

kubectl delete ClusterRoleBinding huawei-csi-driver-registrar-role -n huawei-csi

kubectl delete ClusterRoleBinding huawei-csi-provisioner-role -n huawei-csi

kubectl delete ClusterRoleBinding huawei-csi-resizer-role -n huawei-csi

kubectl delete ClusterRoleBinding huawei-csi-snapshotter-role -n huawei-csi

kubectl delete ClusterRoleBinding snapshot-controller-role -n huawei-csi

kubectl delete Role huawei-csi-resizer-cfg -n huawei-csi

kubectl delete Role huawei-csi-snapshotter-leaderelection -n huawei-csi

kubectl delete Role snapshot-controller-leaderelection -n huawei-csi

kubectl delete RoleBinding huawei-csi-resizer-role-cfg -n huawei-csi

kubectl delete RoleBinding huawei-csi-snapshotter-leaderelection -n huawei-csi

kubectl delete RoleBinding snapshot-controller-leaderelection -n huawei-csi

EOF

b. 执行以下命令删除RBAC权限。如果报NotFound错误,请忽略。

# sh delete-huawei-csi-rbac.sh

## 步骤3 按照以下步骤检查RBAC权限是否已成功删除。

当huawei-csi版本号高于2.2.15,执行以下命令,-n参数为命名空间,请按实际修改。如果提示No resources found,则表示已成功删除。

# kubectl -n huawei-csi -l provisioner=csi.huawei.com get ServiceAccount,role,rolebinding,ClusterRole,ClusterRoleBinding

- 当huawei-csi版本号低于或等于2.2.15,按照以下步骤检查RBAC权限是否已成功删除。
  - a. 执行以下命令创建check-huawei-csi-rbac.sh文件,-n参数为命名空间,请按实际修改。

# cat <<EOF > check-huawei-csi-rbac.sh

kubectl get ServiceAccount -n huawei-csi | grep huawei-csi

kubectl get ClusterRole -n huawei-csi | grep huawei-csi

kubectl get ClusterRoleBinding -n huawei-csi | grep huawei-csi

kubectl get Role -n huawei-csi | grep huawei-csi

kubectl get RoleBinding -n huawei-csi | grep huawei-csi

kubectl get ClusterRole snapshot-controller-runner -n huawei-csi --ignore-not-found=true

kubectl get ClusterRoleBinding snapshot-controller-role -n huawei-csi --ignore-not-found=true

kubectl get Role snapshot-controller-leaderelection -n huawei-csi --ignore-not-found=true

kubectl get RoleBinding snapshot-controller-leaderelection -n huawei-csi --ignore-not-

found=true

FOI

b. 执行以下命令,如果没有回显,表示RBAC权限已成功删除。

# sh check-huawei-csi-rbac.sh

#### ----结束

## 5.2.6 删除老版本镜像

如果需要在集群中删除huawei-csi镜像,需要在所有的worker节点都执行一遍删除操作。

以删除单个节点的镜像为例,操作步骤如下。

## 前提条件

依赖该镜像的容器服务已经被停止,否则无法删除。

## 操作步骤

步骤1 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录该worker节点。

步骤2 执行以下命令,查看现存的所有版本。

• 如果使用的docker,执行**docker image ls | grep huawei-csi**命令。

# docker image ls | grep huawei-csi
REPOSITORY TAG IMAGE ID CREATED SIZE
huawei-csi 2.2.15 b30b3a8b5959 2 weeks ago 79.7MB
huawei-csi 3.2.0 14b854dba227 2 weeks ago 79.6MB

• 如果使用的containerd,执行crictl image ls | grep huawei-csi命令。

# crictl image ls | grep huawei-csi
REPOSITORY TAG IMAGE ID CREATED SIZE
docker.io/library/huawei-csi 2.2.15 b30b3a8b5959 2 weeks ago 79.7MB
docker.io/library/huawei-csi 3.2.0 14b854dba227 2 weeks ago 79.6MB

## 步骤3 执行以下命令,删除老版本镜像:

- 如果使用的docker, 执行docker rmi <REPOSITORY>:<TAG>命令。 # docker rmi huawei-csi:2.2.15
- 如果使用的containerd,执行crictl rmi <REPOSITORY>:<TAG>命令。
   # crictl rmi huawei-csi:2.2.15

**步骤4** 再次执行以下命令,检查镜像是否已成功删除。如果已没有要删除版本的记录,表示已成功删除该版本。

如果使用的docker, 执行docker image ls | grep huawei-csi命令。
 # docker image ls | grep huawei-csi

huawei-csi 3.2.0 14b854dba227 10 minutes ago 80MB

如果使用的containerd,执行**crictl image ls | grep huawei-csi**命令。 # crictl image ls | grep huawei-csi docker.io/library/huawei-csi 3.2.0 14b854dba2273 93.1MB

----结束

## 5.3 卸载 Snapshot 依赖组件服务

## 

- 请勿在存在快照时卸载Snapshot依赖组件服务,否则Kubernetes会自动删除所有的用户快照且无法恢复,请谨慎操作。详细说明请参见删除
   CustomResourceDefinition。
- 请勿在CSI升级时卸载Snapshot依赖组件服务。

## 场景说明

- 当前华为CSI使用了快照特性。
- 当前Kubernetes集群仅存在华为CSI,且不再使用华为CSI。
- 在卸载前请确保在Kubernetes集群中已经没有华为CSI管理的VolumeSnapshot资源。

#### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

步骤2 执行以下命令,卸载Snapshot依赖组件服务。

# kubectl delete crd volumesnapshotclasses.snapshot.storage.k8s.io volumesnapshotcontents.snapshot.storage.k8s.io volumesnapshots.snapshot.storage.k8s.io

步骤3 执行以下命令,检查服务是否已成功卸载。

如果结果为空,表示已成功卸载。 # kubectl get crd | grep snapshot.storage.k8s.io

----结束

# 6 升级/回退操作

本章节介绍如何升级/回退华为CSI。根据您安装时的方式,请使用不同的方式进行升级/回退。

- 如果安装华为CSI时使用的是Helm方式部署,请参考6.1 使用Helm升级/回退华为 CSI章节。
- 如果安装华为CSI时使用的是手动编写资源配置文件方式部署,请参考6.2 手动编写ConfigMap文件升级/回退华为CSI章节。

当完成升级/回退过程中,已经存在的PVC/快照/Pod等资源会正常运行,不会影响您的业务访问。

#### **/ 注意**

- 在升级/回退过程中,不能使用华为CSI创建新的资源,或者对已有的PVC做挂载/卸载操作。
- 在升级/回退过程中,请勿卸载Snapshot依赖组件服务。
- 6.1 使用Helm升级/回退华为CSI
- 6.2 手动编写ConfigMap文件升级/回退华为CSI

## 6.1 使用 Helm 升级/回退华为 CSI

如果华为CSI使用Helm方式部署,请参考本章节进行升级/回退操作。

## 6.1.1 升级华为 CSI

#### 前提条件

在对CSI进行升级操作时,请确保values.yaml文件中backends字段的参数与之前安装华为CSI时配置的huawei-csi-configmap.yaml中的参数保持一致,否则将会导致升级后的CSI无法对之前发放的PVC、Pod等资源进行管理。

#### 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 备份安装CSI时使用的values.yaml文件,可通过**helm get values** *helm-huawei-csi* **-n** *huawei-csi* **-a** > values.yaml.bak 命令备份,其中,helm-huawei-csi为安装时自定义的Helm Chart名称,huawei-csi为安装时自定义的Helm Chart命名空间。

# helm get values helm-huawei-csi -n huawei-csi -a > values.yaml.bak

- 步骤3 如果您需要在容器环境中使用卷快照以及卷快照关联的特性,请通过3.5 检查卷快照依赖组件检查您的环境是否部署了卷快照依赖组件以及卷快照Api资源版本信息。其中支持快照创建PVC的存储请参考表2-5和表2-7,支持快照创建PVC的Kubernetes版本请参考表2-3,如不使用快照相关特性,或者使用的存储和Kubernetes版本不支持快照特性,请跳过本步骤。
- 步骤4 检查升级所需sidecar镜像,如果您镜像仓库中已经存在此次升级所需sidecar镜像,可跳过本步骤。如果您环境中缺少本次升级所需sidecar镜像,请上传对应kubernetes版本所需sidecar镜像,其中sidecar镜像与kubernetes对应关系请参考3.4 检查CSI依赖的镜像。
- 步骤5 上传目标版本华为CSI镜像,具体步骤请参考3.3 上传华为CSI镜像。
- 步骤6 进入升级包 /helm/esdk 目录,执行vi values.yaml命令打开文件,修改华为CSI镜像版本,修改完成后,按Esc,并输入:wq!,保存修改。 其中组件包路径请参考表3-1,修改华为CSI镜像请参考images配置项说明。

# vi values.yaml
images:

# The image name and tag for the Huawei CSI Service container

# Replace the appropriate tag name
huaweiCSIService: huawei-csi:3.2.0

步骤7 进入升级包 /helm/esdk 目录,执行helm upgrade helm-huawei-csi ./ -n huawei-csi -f values.yaml命令升级Helm chart。升级命令将更新Deployment、Daemonset和 RBAC资源,但是secret资源不会更新,如果后端信息发生更改,则必须手动更新 secret,更新操作请参考9.1 更新CSI上配置的存储用户名或密码。

其中,helm-huawei-csi为自定义的chart名称,./表示使用当前目录下的Helm工程,huawei-csi为自定义的命名空间,-f values.yaml表示使用指定values.yaml文件升级。

# helm upgrade helm-huawei-csi ./ -n huawei-csi -f values.yaml Release "helm-huawei-csi" has been upgraded. Happy Helming! NAME: helm-huawei-csi LAST DEPLOYED: Thu Jun 9 07:58:15 2022 NAMESPACE: huawei-csi

STATUS: deployed REVISION: 2 TEST SUITE: None

----结束

## 6.1.2 回退 CSI

Helm回滚命令将当前安装的版本回滚到指定的版本。如果未指定版本号,它将回滚到上一个版本。

#### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 查看当前使用Helm安装的华为CSI版本,使用**helm list -A**命令可以查询所有命名空间中的所有Helm版本;使用**helm list -n** *huawei-csi*可以查询出指定命名空间中的Helm版本。

# helm list -A NAMESPACE **REVISION** NAME **UPDATED STATUS CHART** APP VERSION 2022-06-08 08:48:30.038729177 +0000 UTC deployed helm-test huawei-csi 1 esdk-1.0.0 3.2.0 # helm list -n huawei-csi NAME NAMESPACE **REVISION UPDATED STATUS CHART** APP VERSION 2022-06-08 08:48:30.038729177 +0000 UTC deployed helm-test huawei-csi esdk-1.0.0 3.2.0

步骤3 执行helm rollback helm-huawei-csi revision-number -n huawei-csi命令将华为CSI 回退到指定版本。如下示例中将回退到"1"版本。

其中,helm-huawei-csi为需要回退的chart名称,revision-number为需要回退的目标版本,huawei-csi为chart所在的命名空间。

# helm rollback helm-huawei-csi 1 -n huawei-csi Rollback was a success! Happy Helming!

----结束

## 6.2 手动编写 ConfigMap 文件升级/回退华为 CSI

如果安装华为CSI时使用的是手动编写ConfigMap文件方式部署,请参考本章节进行升级/回退操作。升级/回退操作需要完成如下两个操作:

- 卸载原CSI
- 安装新CSI

## 6.2.1 卸载原 CSI

在升级/回退华为CSI前,首先要卸载原有的华为CSI。

在卸载CSI前,请先执行**kubectl get configmap huawei-csi-configmap -n huawei-csi -o yaml > huawei-csi-configmap.yaml.bak**命令,备份huawei-csi-configmap的内容(在升级CSI时huawei-csi-configmap.yaml中的backends参数需与现有的configmap配置保持一致)。

## 

请在升级/回退前备份CSI的所有配置,以防止升级失败造成您的配置信息丢失。

备份完成后,请参考5.2 手动卸载huawei-csi章节说明,卸载原CSI。

## 6.2.2 安装新 CSI

卸载完成后需要重新安装CSI。

#### 前提条件

已执行**6.2.1 卸载原CSI**章节描述的操作,并确保原CSI中的huawei-csi-configmap.yaml完成备份。

#### 操作步骤

步骤1 完成安装前准备工作。请参考3 安装前准备章节说明。

步骤2 安装新的华为CSI。

- 如果准备部署的华为CSI版本不低于3.0.0版本,强烈建议使用Helm方式对华为CSI 进行安装部署。请参考**4.1 使用Helm安装华为CSI**章节部署新版本的CSI插件。
- 如果准备部署的华为CSI版本低于3.0.0版本,请参考4.2 手动编写ConfigMap文件 部署华为CSI章节部署新版本的CSI插件。

#### **注意**

请确保以下参数配置与升级前一致,**否则会导致huawei-csi服务无法启动,并且无法管理已创建资源**。

- **"storage","name","pools"**参数与已备份的huawei-csi-configmap.yaml.bak文件保持一致。
- "urls"和"parameters"参数请参考已备份的huawei-csi-configmap.yaml.bak文件中的信息,根据您的需要进行设置。

#### ----结束

## **7** 使用华为 CSI

本章节主要介绍如何使用华为CSI对PV、快照的生命周期进行管理。

7.1 PV/PVC管理

7.2 创建VolumeSnapshot

## 7.1 PV/PVC 管理

根据业务的需求,容器中的文件需要在磁盘上进行持久化。当容器被重建或者重新分配至新的节点时,可以继续使用这些持久化数据。

为了可以将数据持久化到存储设备上,您需要在发放容器时使用**持久卷** (PersistentVolume, PV)以及**持久卷申领**(PersistentVolumeClaim, PVC)。

- PV:是Kubernetes集群中的一块存储,可以由管理员事先制备,或者使用存储类 (StorageClass)来动态制备。
- PVC: 是用户对存储的请求。PVC会耗用 PV 资源。PVC可以请求特定的大小和访问模式 (例如,可以要求 PV能够以 ReadWriteOnce、ReadOnlyMany 或 ReadWriteMany 模式之一来挂载,参见访问模式)。

本章将介绍如何使用华为CSI对PV/PVC进行创建、扩容、克隆以及从快照创建PVC。

## 7.1.1 创建 PVC

华为CSI支持在华为存储上创建存储资源(LUN/文件系统),并根据用户的设置供给容器使用。具体支持的特性请参考表2-5或者表2-7。

创建PVC的方式分为动态卷供应和静态卷供应。

- 动态卷供应不需要事先创建PV,华为CSI会根据StorageClass自动在存储设备上创建PV所需要的资源。并且可以在创建PVC时同时创建PV。
- 静态卷供应需要管理员事先在存储设备上创建好所需要的资源,通过创建PV的方式使用已存在的资源。并且可以在创建PVC时指定关联的PV。

#### 7.1.1.1 动态卷供应

动态卷供应(Dynamic Volume Provisioning)允许按需创建存储卷。动态卷供应依赖 StorageClass对象。 集群管理员可以根据需要定义多个StorageClass对象,在声明PV 或者PVC时,指定满足业务要求的StorageClass。华为CSI在从华为存储设备上申请资源时,会根据StorageClass的预置定义,创建满足业务要求的存储资源。

为了完成动态卷供应,需要完成如下两步:

- 配置StorageClass
- 配置PVC

#### 7.1.1.1.1 配置 StorageClass

**存储类(StorageClass**)为管理员提供了描述存储 "类" 的方法。 不同的类型可能会映射到一组不同的能力定义。Kubernetes集群用户可基于StorageClass进行动态卷制备。

StorageClass支持配置如下参数信息。

使用SAN存储时可参考示例文件/examples/sc-lun.yaml,使用NAS存储时可参考示例文件/examples/sc-fs.yaml。

表 7-1 StorageClass 配置参数说明

参数	说明	备注
metadata.name	自定义的StorageClass 对象名称。必选参数。	以Kubernetes v1.22.1为例,支持数字、小写字母、中划线(-)和点 (.)的组合,并且必须以字母数字字符开头和结尾。
provisioner	制备器名称。必选参 数。	该字段需要指定为安装华为CSI时设置的驱动名。 默认的驱动名为 "csi.huawei.com"。
reclaimPolicy	回收策略。可选参数。 支持如下类型:  Delete: 自动回收资源。  Retain: 手动回收资源。	<ul><li>Delete: 删除PV/PVC时会关联删除存储上的资源。</li><li>Retain: 删除PV/PVC时不会删除存储上的资源。</li></ul>
allowVolumeExp ansion	是否允许卷扩展。可选 参数。参数设置为true 时,使用该 StorageClass的PV可以 进行扩容操作。	此功能仅可用于扩容PV,不能用于缩容PV。 扩容PV功能在Kubernetes 1.14 (alpha)后才支持。
parameters.backe nd	待创建资源所在的后端 名称。可选参数。	如果不设置,华为CSI随机选择一个满足容量要求的后端创建资源。 建议指定后端,确保创建的资源在预期的后端上。

参数	说明	备注
parameters.pool	待创建资源所在的存储 资源池名称。可选设 置。如果设置,则必须 设置 parameters.backend。	如果不设置,华为CSI会在所选后端上随机选择一个满足容量要求的存储池创建资源。建议指定存储池,确保创建的资源在预期的存储池上。
parameters.volu meType	待创建卷类型。支持如下类型:  • lun: 存储侧发放的资源是LUN。  • fs: 存储侧发放的资源是文件系统。	
parameters.alloc Type	待创建卷的分配类型。可选参数。支持如下类型:  thin: 创建时不会分配所有需要的空间,而是根据使用情况动态分配。  thick: 创建时分配所有需要的空间。	如果不设置,默认为thin。
parameters.fsTyp e	主机文件系统类型。必 选参数。支持类型为:     ext2     ext3     ext4     xfs	如果不设置,默认为ext4。仅当 StorageClass的volumeType设置为 "lun",且PVC的volumeMode配置 为"Filesystem"时生效。
parameters.auth Client	可访问该卷的NFS客户 端IP地址信息,在指定 volumeType为"fs" 时必选。 支持输入客户端主机名 称(建议使用全称域 名)、客户端IP地址、 客户端IP地址段。	可以使用"*"表示任意客户端。当您不确定访问客户端IP信息时,建议使用"*"防止客户端访问被存储拒绝。当使用客户端主机名称时建议使用全称域名。 IP地址支持IPv4、IPv6地址或两者的混合IP地址。可以同时输入多个主机名称、IP地址或IP地址段,以英文分号,空格或按回车键隔开。如示例: "192.168.0.10;192.168.0.0/24;myserver1.test"
parameters.clone Speed	克隆速度。可选参数。 支持配置为1~4。	如果不设置,默认值3。4速度最快。配置克隆PVC或从快照创建PVC时生效,参考7.1.3 克隆PVC或7.1.4 从快照创建PVC。

参数	说明	备注
parameters.applic ationType	后端为OceanStor Dorado V6存储时,指 定创建LUN/NAS时的 应用类型名称。	<ul> <li>"volumeType"为"lun"时,在DeviceManager管理界面,选择"服务 &gt; 块服务 &gt; LUN组 &gt; LUN &gt; 创建 &gt; 应用类型",获取应用类型名称。</li> <li>"volumeType"为"fs"时,在DeviceManager管理界面,选择"服务 &gt; 文件服务 &gt; 文件系统 &gt; 创建 &gt; 应用类型",获取应用类型名称。</li> </ul>
parameters.qos	PV在存储侧的 LUN/NAS的QoS设置。 可选参数。 配置项值是字典格式的 JSON字符串(字符串 两边由单引号修饰,字 典key由双引号修 饰)。如: '{"maxMBPS": 999, "maxIOPS": 999}'	支持的QoS配置请参考 <mark>表7-2</mark> 说明。
parameters.stora geQuota	PV在存储侧配额设置。 可选参数,仅在对接 OceanStor Pacific系列 存储使用NAS时生效。 配置项值是字典格式的 JSON字符串(字符串 两边由单引号修饰,字 典key由双引号修 饰)。如: '{"spaceQuota": "softQuota", "gracePeriod": 100}'	支持的配额配置请参考表7-3说明。
parameters.hyper Metro	是否创建双活卷。条件可选,当使用的后端是双活类型的后端需要配置。  • "true": 创建的卷为双活卷。  • "false": 创建的卷为普通卷。	当使用的后端是双活类型的后端,且需要发放双活卷时,设置该参数。默认值为"false"。
parameters.fsPer mission	挂载到容器内的目录权 限。可选参数。	配置格式参考Linux权限设置,如 "777"、"755"等。 支持所有的SAN存储,NAS存储仅支 持OceanStor Dorado V6、 OceanStor V6、OceanStor Pacific 8.1.2及之后版本的存储设备。

参数	说明	备注
parameters.rootS quash	用于设置是否允许客户端的root权限。可选参数。可选值:  root_squash:表示不允许客户端以root用户访问,客户端使用root用户访问时映射为匿名用户。  no_root_squash:表示允许客户端以root用户访问,保留root用户的权限。	仅支持NAS存储。
parameters.allSq uash	用于设置是否保留共享目录的UID和GID。可选参数。可选值:  all_squash:表示共享目录的UID和GID映射为匿名用户。  no_all_squash:表示保留共享目录的UID和GID。	仅支持NAS存储。
parameters.snaps hotDirectoryVisib ility	用于设置快照目录是否可见。可选参数。可选值: • visible:表示快照目录可见。 • invisible:表示快照目录不可见。	仅支持NAS存储。
parameters.reser vedSnapshotSpac eRatio	用于配置快照预留空间。可选参数。参数类型:字符串取值范围:"0"~"50"	支持OceanStor Dorado 6.1.5+、 OceanStor 6.1.5+的NAS存储。
parameters.descri ption	用于配置创建的文件系统/LUN的描述信息。可选参数。参数类型:字符串长度限制:0-255	仅支持企业存储文件系统及LUN。

参数	说明	备注
mountOptions.nf svers	主机侧NFS挂载选项。 支持如下挂载选项:	在主机执行mount命令时-o参数后的 可选选项。列表格式。
	nfsvers: 挂载NFS时的 协议版本。支持配置的 参数值为"3", "4","4.0"和 "4.1"。	指定NFS版本挂载时,当前支持NFS 3/4.0/4.1协议(需存储设备支持且开 启)。当配置参数为nfsvers=4时,因 为操作系统配置的不同,实际挂载可 能为NFS 4的最高版本协议,如4.1, 当需要使用4.0协议时,建议配置 nfsvers=4.0。
mountOptions.ac	DPC命名空间支持ACL 功能。DPC客户端支持 POSIX ACL、NFSv4 ACL、NT ACL的鉴权行 为。	acl、aclonlyposix、cnflush、cflush 参数描述仅供参考,详细参数说明请 参考《OceanStor Pacific系列 产品 文档》> 配置 > 文件服务基础业务配 置指南 > 配置基础业务(DPC场景) > 客户端访问DPC共享 > 步骤2。
mountOptions.ac lonlyposix	DPC命名空间支持 POSIX ACL功能,DPC 客户端支持POSIX ACL 的鉴权行为。	aclonlyposix与acl参数同时使用时, 仅acl参数生效,即命名空间支持ACL 功能。
	支持POSIX ACL的协议 有: DPC、NFSv3、 HDFS。如使用NFSv4 ACL或NT ACL,会导致 DPC客户端无法识别该 类型的ACL,从而导致 该类型的ACL不会生 效。	
mountOptions.cnf lush	异步刷盘模式,即关闭 命名空间下的文件时不 会立即刷盘。	异步刷盘模式,当文件关闭时不会同步将Cache的数据持久化到存储介质中,而是通过Cache异步刷盘的方式将数据写入存储介质,Cache的后台刷盘将在写业务完成后根据刷盘周期定时刷盘。在多客户端场景下,对同一文件进行并行操作,文件Size的更新会受刷盘周期的影响,即当刷盘动作完成后才会更新文件的Size,更新通常会在数秒内完成。同步I/O不受刷盘周期影响。
mountOptions.cfl ush	同步刷盘模式,即关闭 命名空间下的文件时立 即刷盘。	默认使用同步刷盘模式。

表 7-2 支持的 QoS 配置

存储类型	参数名	参数描述	备注	
OceanSt or V3/ OceanSt	IOTYPE	控制读写类型。	可选参数(未明确指定将使用 后端存储默认值,具体参考相 关存储资料)。	
or V5			有效值如下:	
			● 0: 读I/O	
			• 1: 写I/O	
			● 2: 读写I/O	
	MAXBAN DWIDTH	最大带宽限制策略。	单位MB/s,有效值为>0的整 数。	
	MINBAND WIDTH	最小带宽保护策略。	单位MB/s,有效值为>0的整 数。	
	MAXIOPS	最大IOPS限制策略。	有效值为>0的整数。	
	MINIOPS	最小IOPS保护策略。	有效值为>0的整数。	
	LATENCY	最大时延保护策略。	单位ms,有效值为>0的整数。	
OceanSt	IOTYPE	控制读写类型。	有效值如下:	
or Dorado			● 2: 读写I/O	
V3	MAXBAN DWIDTH	最大带宽限制策略。	单位MB/s,整数, 范围 1~999999999。	
	MAXIOPS	最大IOPS限制策略。	类型为整数, 范围 100~999999999。	
OceanSt	IOTYPE	控制读写类型。	有效值如下:	
or Dorado			● 2: 读写I/O	
V6/ OceanSt	MAXBAN DWIDTH	最大带宽限制策略。	单位MB/s,类型为整数, 范围 1~999999999。	
or V6	MINBAND WIDTH	最小带宽保护策略。	单位MB/s,类型为整数, 范围 1~999999999。	
	MAXIOPS	最大IOPS限制策略。	类型为整数, 范围 100~999999999。	
	MINIOPS	最小IOPS保护策略。	类型为整数, 范围 100~999999999。	
	LATENCY	最大时延保护策略。	单位ms,仅支持配置0.5或 1.5。	
FusionSt orage/ OceanSt	maxMBPS	最大带宽限制策略。	必填。有效值为>0的整数,单 位MB/s。	

存储类型	参数名	参数描述	备注
or Pacific系 列	maxIOPS	最大IOPS限制策略。	必填。有效值为>0的整数。

#### 表 7-3 支持的配额配置

参数名	参数描述	备注
spaceQuota	文件配额类型。	必选。仅支持配置 "softQuota"或者 "hardQuota"。
gracePeriod	配置软配额时,允许的超 限天数。	条件可选,当 "spaceQuota"配置为 "softQuota"时可选。 类型为整数,支持范围为0 ~4294967294。

#### 典型场景下StorageClass配置请参考如下示例:

- StorageClass中设置后端和存储池
- StorageClass中设置NFS访问方式
- StorageClass中设置本地文件系统访问方式
- StorageClass中设置DPC访问方式
- StorageClass中设置应用类型
- StorageClass中设置软配额
- StorageClass中设置QoS
- StorageClass中设置双活
- StorageClass中设置挂载目录权限

## StorageClass 中设置后端和存储池

如果在一个Kubernetes集群中配置了多个华为后端,或者一个华为后端提供多个存储 池,建议在StorageClass中配置指定的后端和存储池信息,避免华为CSI随机选择后端 和存储池,导致卷所在的存储不符合规划。

SAN存储设置后端和存储池可以参考如下配置示例。

kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
allowVolumeExpansion: true
parameters:
backend: "iscsi\_dorado\_181"
pool: "pool001"

volumeType: lun allocType: thin

NAS存储设置后端和存储池可以参考如下配置示例。

kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: mysc provisioner: csi.huawei.com allowVolumeExpansion: true parameters: backend: "iscsi\_dorado\_181" pool: "pool001" volumeType: fs allocType: thin authClient: "\*"

#### StorageClass 中设置 NFS 访问方式

容器使用NFS文件系统作为存储资源时,可以参考如下配置示例。该示例中,NFS挂载时指定版本为4.1。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
parameters:
backend: nfs_dorado_181
pool: pool001
volumeType: fs
allocType: thin
authClient: "192.168.0.10;192.168.0.0/24;myserver1.test" #use * for all client
mountOptions:
- nfsvers=4.1
```

## StorageClass 中设置本地文件系统访问方式

容器使用企业存储或者分布式存储的LUN作为存储资源时,且需要格式化文件系统为本地文件系统时,可以参考如下示例。

kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
parameters:
backend: iscsi\_dorado\_181
pool: pool001
volumeType: lun
allocType: thin
fsType: xfs

## StorageClass 中设置 DPC 访问方式

当容器使用OceanStor Pacific系列存储,且存储支持DPC协议访问时,可以在 StorageClass中配置DPC访问的挂载参数。本例中设置挂载时使用"**acl**"做鉴权参 数,使用"**cnflush**"为设置异步刷盘模式。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
```

```
parameters:
backend: nfs_pacific_101
pool: pool001
volumeType: fs
allocType: thin
authClient: "*" #use * for all client
mountOptions:
- acl
- cnflush
```

## StorageClass 中设置应用类型

当容器使用OceanStor Dorado V6存储的LUN作为存储时,如果使用存储默认的应用类型无法满足某些业务的I/O模型要求(如容器对外提供数据库OLAP服务),可以在StorageClass中配置应用类型,提升存储性能。具体需要使用的应用类型请参考对应存储产品的产品文档说明。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
parameters:
backend: iscsi_dorado_181
pool: pool001
volumeType: lun
allocType: thin
fsType: xfs
applicationType: Oracle_OLAP
```

## StorageClass 中设置软配额

当容器使用OceanStor Pacific系列存储的文件系统作为存储时,可以在StorageClass中配置软配额信息,可以参考如下配置示例。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
parameters:
backend: nfs_pacific_101
pool: pool001
volumeType: fs
allocType: thin
authClient: "*"
storageQuota: '{"spaceQuota": "softQuota", "gracePeriod": 100}'
mountOptions:
- nfsvers=3
```

## StorageClass 中设置 QoS

容器使用企业存储或者分布式存储作为存储资源时,可以为容器使用的存储资源设置 QoS,从而保证这些容器对存储读写满足一定的服务等级。

不同型号或版本的存储支持的QoS设置不同,请参考表7-2找到对应存储的配置项。本示例中的后端是OceanStor Dorado V6存储,其他存储可以参考本例设置。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
parameters:
```

backend: iscsi\_dorado\_181
pool: pool001
volumeType: lun
allocType: thin
fsType: xfs
qos: '{"IOTYPE": 2, "MINIOPS": 1000}'

完成StorageClass配置后,进行如下步骤创建StorageClass。

步骤1 执行以下命令,基于该yaml文件创建StorageClass。

# kubectl create -f mysc.yaml

步骤2 执行以下命令,查看当前已经创建的StorageClass信息。

# kubectl get sc

NAME PROVISIONER RECLAIMPOLICY VOLUMEBINDINGMODE ALLOWVOLUMEEXPANSION AGE mysc csi.huawei.com Delete Immediate false 34s

创建StorageClass后,就可以使用该StorageClass进行创建PV或者PVC。

----结束

#### <u> 注意</u>

在StorageClass的使用中请注意如下事项:

- 请不要删除正在被PV使用的StorageClass。否则会导致PV的StorageClass信息缺失,导致在主机侧挂载时出现错误。
- 针对StorageClass进行的修改将不会在已经创建的PV上生效。您需要删除这些PV,并重新使用修改后的StorageClass创建才能应用修改的参数。

## StorageClass 中设置双活

容器使用NFS双活文件系统作为存储资源时,可以参考如下配置示例。该示例中,使用的后端是支持双活的后端,且配置"hyperMetro"参数为"true"。

kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: mysc
provisioner: csi.huawei.com
parameters:
backend: nfs\_hypermetro\_dorado\_181
pool: pool001
volumeType: fs
hyperMetro: "true"
allocType: thin
authClient: "\*"

#### 须知

发放NAS双活卷前,需要在两台存储设备之间配置双活关系,包含远端设备、双活域等,仅支持文件系统双活域工作模式为双活模式,配置操作请参考对应存储型号的产品文档。

## StorageClass 中设置挂载目录权限

当需要修改容器内挂载目录的权限时,可以在StorageClass中配置目录权限信息,可以 参考如下配置示例。

kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: mysc provisioner: csi.huawei.com allowVolumeExpansion: true parameters: volumeType: fs allocType: thin authClient: "\*" fsPermission: "777"

rootSquash: "no\_root\_squash" # 该参数仅支持NAS存储 all\_squash: "no\_all\_squash" # 该参数仅支持NAS存储

#### 7.1.1.1.2 配置 PVC

在完成配置StorageClass以后,就可以用该StorageClass来配置PVC。PVC的配置模板请参考华为CSI软件包中的examples目录下的pvc-xxxx.yaml文件示例。

表 7-4 pvc-xxxx.yaml 文件示例参数说明

参数	说明	备注
metadata. name	自定义的PVC对象名称。	以Kubernetes v1.22.1为例,支持数字、小写字母、中划线(-)和点(.)的组合,并且必须以字母数字字符开头和结尾。
spec.volu meMode	卷模式。可选参数。 当使用LUN类型的卷时,支持配置以下类型: • Filesystem: 本地文件系统。 • Block: 裸设备。	该参数在挂载PV时生效,默认为Filesystem。  Filesystem表示在容器通过一个本地文件系统访问PV,本地文件系统类型为指定StorageClass中的fsType字段指定。  Block表示使用裸卷的方式访问访问PV。
spec.stora geClassNa me	StorageClass对象名称。	业务需要的StorageClass对象名称。
spec.resou rces.reque sts.storage	指定待创建卷大小,格 式为***Gi,单位为GiB。 需要满足大小为512字节 的整数倍。	PVC容量的规格取决于存储规格限制和主机规格限制。以OceanStor Dorado 6.1.2/OceanStor Pacific系列 8.1.0对接CentOS 7为例,当使用的是ext4文件系统时,容量限制见表7-5;当使用的是XFS文件系统时,容量限制见表7-6。如果使用的是NFS或者课设备,容量需满足使用的华为存储设备型号和版本所要求的规格约束。如果PVC容量不在规格范围内,可能会由于存储规格限制或主机文件系统规格限制导致创建PVC或Pod失败。

参数	说明	备注
spec.acces sModes	指定卷访问模式。  RWO (ReadWriteOnce): 巻可以被注述中节点。 を写った。 を写った。 ROX (ReadOnlyMany): 以のでは表すでは、表すでは、表すでは、表すでは、表すでは、表すでは、表すでは、表すで	<ul> <li>RWO/ROX/RWOP: 所有类型卷均支持,RWOP需Kubernetes 1.22版本以上支持。请参考9.7 开启ReadWriteOncePod功能门章节,检查您的Kubernetes集群是否开启该特性。</li> <li>RWX: volumeMode设置为Block或者NFS类型的卷支持。</li> </ul>

#### 表 7-5 ext4 容量的规格

存储类型	存储规格限制	ext4规格限制	CSI规格限制
OceanStor Dorado 6.1.2	512Ki~256Ti	50Ti	512Ki~50Ti
OceanStor Pacific系列 8.1.0	64Mi~512Ti	50Ti	64Mi~50Ti

#### 表 7-6 XFS 容量的规格

存储类型	存储规格限制	XFS规格限制	CSI规格限制
OceanStor Dorado 6.1.2	512Ki~256Ti	500Ti	512Ki~500Ti
OceanStor Pacific系列 8.1.0	64Mi~512Ti	500Ti	64Mi~500Ti

步骤1 根据业务需要,参考本节描述和PVC配置文件示例,修改具体参数,生成本次需要创建的PVC配置文件,如本例中mypvc.yaml文件。

kind: PersistentVolumeClaim apiVersion: v1

```
metadata:
name: mypvc
spec:
accessModes:
- ReadWriteOnce
volumeMode: Filesystem
storageClassName: mysc
resources:
requests:
storage: 100Gi
```

#### 步骤2 执行命令,使用配置文件创建PVC。

# kubectl create -f mypvc.yaml

步骤3 等待一段时间后,执行以下命令,查看已经创建的PVC信息。如果PVC的状态是 "Bound"时,则说明该PVC已经创建成功,后续可以被Pod使用。

```
# kubectl get pvc mypvc
NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE mypvc Bound pvc-840054d3-1d5b-4153-b73f-826f980abf9e 100Gi RWO mysc 12s
```

#### **注意**

- 完成创建PVC操作后,如果长时间后(如一分钟后)PVC的状态是Pending,请参考 10.6 创建PVC时, PVC的状态为Pending。
- 建议每批次最多批量创建/删除100个PVC。

#### ----结束

在完成PVC创建后,就可以使用PVC来创建Pod。如下示例是一个简单的使用PVC示例,在该示例中,创建的Pod使用了刚刚创建的*mypvc*。

```
apiVersion: apps/v1
kind: Deployment
metadata:
name: nginx-deployment
spec:
 selector:
  matchLabels:
   app: nginx
 replicas: 2
 template:
  metadata:
   labels:
     app: nginx
  spec:
   containers:
    - image: nginx:alpine
     name: container-0
     volumeMounts:
     - mountPath: /tmp
      name: pvc-mypvc
   restartPolicy: Always
   volumes:
    - name: pvc-mypvc
     persistentVolumeClaim:
                                         # name of PVC
      claimName: mypvc
```

#### 7.1.1.2 静态卷供应

静态卷供应(Static Volume Provisioning)允许管理员使用已经在存储侧创建的资源做为PV,供集群中的容器使用。

为了完成静态卷供应,需要完成如下两步:

- 配置PV
- 配置PVC

#### 7.1.1.2.1 配置 PV

使用静态卷供应时,不需要配置StorageClass,直接创建PV。创建PV前需要配置PV。如下示例是一个静态卷供应的配置文件。

```
kind: PersistentVolume
apiVersion: v1
metadata:
name: mypv
spec:
volumeMode: Filesystem
storageClassName: ""
accessModes:
- ReadWriteOnce
csi:
driver: csi.huawei.com
volumeHandle: iscsi_dorado_181.lun0001
fsType: xfs
capacity:
storage: 100Gi
```

如上例所示,静态卷供应的配置文件中,storageClassName参数必须配置为 " " " ',如果不配置,否则Kubernetes会使用系统默认的StorageClass。其他参数及说明见表7-7。

表 7-7 静态卷供应参数

参数	说明	备注
metadata.nam e	自定义的PVC对象名称。	以Kubernetes v1.22.1为例,支持数字、小写字母、中划线(-)和点 (.)的组合,并且必须以字母数字字符开头和结尾。
spec.volumeM ode	卷模式。可选参数。 当使用LUN类型的卷时,支持配置以下类型:  Filesystem: 本地文件系统。 Block: 裸设备。	该参数在挂载PV时生效,默认为Filesystem。  Filesystem表示在容器通过一个本地文件系统访问PV,本地文件系统类型为指定StorageClass中的fsType字段指定。  Block表示使用裸卷的方式访问访问PV。
spec.storageCla ssName	StorageClass对象名称。必 选参数。	此处须设置为空字符串(即输入 "")。

参数	说明	备注
spec.accessMo des	指定卷访问模式。 RWO (ReadWriteOnce): 卷可以被一个节点以或也 允许运行在同一节点 允许运行在同一节点。 ROX (ReadOnlyMany): 卷可以被多个可式挂载。 RWX (ReadWriteMany): 卷可以被多个节点以读 写方式挂载。 RWOP (ReadWriteOncePod): 卷只允许是一个中的人员。	<ul> <li>RWO/ROX/RWOP: 所有类型卷均支持, RWOP需Kubernetes 1.22版本以上支持。请参考9.7 开启ReadWriteOncePod功能门章节,检查您的Kubernetes集群是否开启该特性。</li> <li>RWX: volumeMode设置为Block或者NFS类型的卷支持。</li> </ul>
spec.csi.driver	CSI驱动名称。	固定填写csi.huawei.com。
spec.csi.volume Handle	存储资源的唯一标志。必 选参数。 格式为: <backendname>.<volume -name&gt;</volume </backendname>	该参数值由以下两部分构成:  ■ 
spec.csi.fsType	指定主机文件系统类型。 可选参数。支持类型为: • ext2 • ext3 • ext4 • xfs	如果不设置,默认为ext4。仅当 volumeMode配置为"Filesystem" 时生效。
spec.capacity.st orage	指定卷大小。   	请确保与存储上对应资源的容量保持一致。Kubernetes并不会调用CSI检查此字段值的正确性,所以在PV容量与存储上对应资源的容量不一致也能被成功创建。

参数	说明	备注
mountOptions. nfsvers	主机侧NFS挂载选项。支持如下挂载选项: nfsvers: 挂载NFS时的协议版本。支持配置的参数值为"3","4","4.0"和"4.1"。	在主机执行mount命令时-o参数后的可选选项。列表格式。 指定NFS版本挂载时,当前支持NFS 3/4.0/4.1协议(需存储设备支持且 开启)。当配置参数为nfsvers=4 时,因为操作系统配置的不同,实 际挂载可能为NFS 4的最高版本协 议,如4.1,当需要使用4.0协议时, 建议配置nfsvers=4.0。
mountOptions. acl	DPC命名空间支持ACL功能。DPC客户端支持POSIXACL、NFSv4 ACL、NTACL的鉴权行为。	acl、aclonlyposix、cnflush、cflush 参数描述仅供参考,详细参数说明 请参考《OceanStor Pacific系列 产 品文档》> 配置 > 文件服务基础业 务配置指南 > 配置基础业务(DPC 场景) > 客户端访问DPC共享 > 步 骤2。
mountOptions. aclonlyposix	DPC命名空间支持POSIX ACL功能,DPC客户端支持 POSIX ACL的鉴权行为。 支持POSIX ACL的协议有: DPC、NFSv3、HDFS。如使用NFSv4 ACL或NT ACL,会导致DPC客户端无法识别该类型的ACL,从而导致该类型的ACL不会生效。	aclonlyposix与acl参数同时使用 时,仅acl参数生效,即命名空间支 持ACL功能。
mountOptions. cnflush	异步刷盘模式,即关闭命 名空间下的文件时不会立 即刷盘。	异步刷盘模式,当文件关闭时不会同步将Cache的数据持久化到存储介质中,而是通过Cache异步刷盘的方式将数据写入存储介质,Cache的后台刷盘将在写业务完成后根据刷盘周期定时刷盘。在多客户端场景下,对同一文件进行并行操作,文件Size的更新会受刷盘周期的影响,即当刷盘动作完成后才会更新文件的Size,更新通常会在数秒内完成。同步I/O不受刷盘周期影响。
mountOptions. cflush	同步刷盘模式,即关闭命 名空间下的文件时立即刷 盘。	默认使用同步刷盘模式。

## 前提条件

● 存储侧已经存在待创建PV所需要的存储资源,如LUN或者文件系统。如果存储资源是文件系统,还需要创建文件系统的共享和客户端信息。

● 已参考<mark>表7-7</mark>,完成PV的配置文件。

#### 操作步骤

步骤1 执行以下命令,基于准备好的yaml文件创建PV。

# kubectl create -f mypv.yaml

**步骤2** 等待一段时间后,执行以下命令,查看已经创建的PV信息。当PV状态为"Available"时,表明PV创建成功。

# kubectl get pv
NAME CAPACITY ACCESS MODES RECLAIM POLICY STATUS CLAIM STORAGECLASS
REASON AGE
mypv 100Gi RWO Retain Available 4s

----结束

#### 7.1.1.2.2 配置 PVC

当PV以静态卷供应的方式创建完成后,可以基于该PV创建PVC,从而供容器使用。如下示例是一个使用静态卷供应的PVC配置文件。

kind: PersistentVolumeClaim
apiVersion: v1
metadata:
name: mypvc
spec:
accessModes:
- ReadWriteOnce
volumeMode: Filesystem
resources:
requests:
storage: 100Gi
volumeName: mypv

如上例所示,通过设置PVC配置文件中"volumeName"参数为之前通过静态卷供应创建的PV。具体参数见表7-8。

#### 表 7-8 PVC 参数说明

参数	说明	备注
metadata. name	自定义的PVC对象名称。	以Kubernetes v1.22.1为例,支持数字、小写字母、中划线(-)和点(.) 的组合,并且必须以字母数字字符开头和结尾。

参数	说明	备注
spec.acces sModes	指定卷访问模式。 RWO (ReadWriteOnce): 卷可以被一个节点以读写方式挂载。该模式也允许运行在同一节点上的多个 Pod 访问卷。 ROX (ReadOnlyMany):卷可以被多个节点以只读方式挂载。 RWX (ReadWriteMany):卷可以被多个节点以读写方式挂载。 RWOP (ReadWriteOncePod):卷只能被单个 Pod 以读写方式挂载。该特性需要 Kubernetes 1.22 以上版本。	<ul> <li>RWO/ROX/RWOP: 所有类型卷均支持,RWOP需Kubernetes 1.22版本以上支持。请参考9.7 开启ReadWriteOncePod功能门章节,检查您的Kubernetes集群是否开启该特性。</li> <li>RWX: volumeMode设置为Block或者NFS类型的卷支持。</li> </ul>
spec.volu meMode	卷模式。	可选, 支持Filesystem或Block, 默认 为Filesystem。该参数在创建Pod时生 效,其中Filesystem表示在PVC上创建 一个文件系统访问存储, Block表示使 用裸卷的方式访问存储。
spec.resou rces.reque sts.storage	指定待创建卷大小。	指定待创建卷大小,格式为***Gi,单位为GiB。 PVC容量的规格取决于存储规格限制和主机规格限制。以OceanStor Dorado 6.1.2/OceanStor Pacific系列 8.1.0对接CentOS 7为例,当使用的是ext4文件系统时,容量限制见表7-5;当使用的是XFS文件系统时,容量限制见表7-6。如果使用的是NFS或者裸设备,容量需满足使用的华为存储设备型号和版本所要求的规格约束。 如果PVC容量不在规格范围内,可能会由于存储规格限制或主机文件系统规格限制导致创建PVC或Pod失败。 在通过静态PV创建PVC时,若PVC容量小于绑定PV容量,最终PVC容量大小为绑定PV容量,若PVC容量大于绑定PV容量,PVC将无法被创建。
spec.volu meName	PV对象名称。	静态创建PVC时必选。

#### 操作步骤

步骤1 执行以下命令,基于已配置的yaml文件创建PVC。

# kubectl create -f mypvc.yaml

步骤2 等待一段时间后,执行以下命令,查看已经创建的PVC信息。

```
# kubectl get pvc
NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE
mypvc Bound pvc-840054d3-1d5b-4153-b73f-826f980abf9e 100Gi RWX mysc 12s
```

#### 山 说明

- 完成创建PVC操作后,如果长时间后(如一分钟后)PVC的状态是Pending,请参考**10.6 创建PVC时,PVC的状态为Pending**。
- 建议每批次最多批量创建/删除100个PVC。

#### ----结束

在完成PVC创建后,就可以使用PVC来创建Pod。如下示例是一个简单的使用PVC示例,在该示例中,创建的Pod使用了刚刚创建的*mypvc*。

```
apiVersion: apps/v1
kind: Deployment
metadata:
name: nginx-deployment
spec:
 selector:
  matchLabels:
   app: nginx
 replicas: 2
 template:
  metadata:
   labels:
     app: nginx
  spec:
   containers:
    - image: nginx:alpine
     name: container-0
     volumeMounts:
     - mountPath: /tmp
      name: pvc-mypvc
   restartPolicy: Always
   volumes:
    - name: pvc-mypvc
     persistentVolumeClaim:
      claimName: mypvc
                                         # name of PVC
```

## 7.1.2 扩容 PVC

当容器使用的PVC容量不足时,需要对该PVC进行扩容操作。

#### 前提条件

- PVC已创建,所在的backend存在且支持扩容。
- 支持扩容的存储请参考表2-5和表2-7,支持扩容的Kubernetes版本请参考表2-3。
- huawei-csi-controller启用了csi-resizer服务。

```
# kubectl describe deploy huawei-csi-controller -n huawei-csi | grep csi-resizer csi-resizer:

Image: k8s.gcr.io/sig-storage/csi-resizer:v1.3.0
```

#### 操作步骤

**步骤1** 执行**kubectl get pvc** *mypvc*命令,查询PVC的StorageClass名称。其中,*mypvc*为需要扩容的PVC名称。

# kubectl get pvc mypvc

NAME STATUS VOLUME CAPACITY ACCESS MODES

STORAGECLASS AGE

mypvc Bound pvc-3383be36-537c-4cb1-8f32-a415fa6ba384 2Gi RWX

*mysc* 145m

**步骤2** 执行**kubectl get sc** *mysc*命令,查询StorageClass是否支持扩容。其中*,mysc*为需要 查看的StorageClass名称。

# kubectl get sc *mysc* 

NAME PROVISIONER RECLAIMPOLICY VOLUMEBINDINGMODE

ALLOWVOLUMEEXPANSION AGE

mysc csi.huawei.com Delete Immediate **true** 172m

如果ALLOWVOLUMEEXPANSION的值为true,表示当前StorageClass已经支持扩容,请跳转至步骤**步骤4**。

步骤3 执行以下命令,将"allowVolumeExpansion"的值修改为"true"。其中,*mysc*为需要修改的StorageClass名称。

# kubectl patch sc mysc --patch '{"allowVolumeExpansion":true}'

步骤4 执行以下命令进行扩容。

# kubectl patch pvc mypvc -p '{"spec":{"resources":{"requests":{"storage":"120Gi"}}}}'

其中,"*mypvc*"是需要扩容的PVC名称,"*120Gi*"是扩容后的容量大小。请根据实际情况进行替换。

#### □ 说明

- PVC容量的规格取决于存储规格限制和主机规格限制。以OceanStor Dorado 6.1.2/ OceanStor Pacific系列 8.1.0对接CentOS 7为例,当使用的是ext4文件系统时,容量限制见表7-5;当使用的是XFS文件系统时,容量限制见表7-6。如果使用的是NFS或者裸设备,容量需满足使用的华为存储设备型号和版本所要求的规格约束。
- 如果PVC容量不在规格范围内,可能会由于存储规格限制或主机文件系统规格限制导致创建 PVC或Pod失败。
- 如果扩容的目标容量超过存储池容量导致扩容失败,请参考10.17 PVC扩容的目标容量超过存储池容量导致扩容失败。

**步骤5** 执行以下命令,检查容量修改是否生效。

# kubectl get pvc

NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE mypvc Bound pvc-840054d3-1d5b-4153-b73f-826f980abf9e **120Gi** RWO mysc 24s

----结束

## 7.1.3 克隆 PVC

本章节描述如何克隆PVC。

在克隆PVC时,需要指定数据源。如下示例是一个简单的克隆PVC示例,在该示例中,使用"mypvc"作为数据源,新创建了一个名叫"myclone"的PVC。

kind: PersistentVolumeClaim

apiVersion: v1

metadata:

name: myclone

spec:

storageClassName: mysc

dataSource:

name: mypvc

kind: PersistentVolumeClaim

volumeMode: Filesystem

accessModes:

- ReadWriteOnce

storage: 2Gi

resources: requests:

#### <u> 注意</u>

- 指定的storageClassName必须和dataSource中的源卷的StorageClass需一致。
- 克隆卷的容量必须不小于源卷容量,建议和源卷容量保持一致。

#### 前提条件

系统中已经存在源PVC,且源PVC所在的backend存在支持克隆。支持克隆的存储请参考表2-5和表2-7,支持克隆的Kubernetes版本请参考表2-3。

#### 操作步骤

步骤1 执行以下命令,基于克隆卷的配置文件创建PVC。

# kubectl create -f myclone.yaml

----结束

## 7.1.4 从快照创建 PVC

本章节描述如何从快照创建PVC。

在创建这个PVC时,需要指定数据源。如下示例是一个简单的从快照创建PVC示例,在该示例中,使用快照"mysnapshot"作为数据源,新创建了一个名叫"myrestore"的PVC。

apiVersion: v1

kind: PersistentVolumeClaim

metadata:

name: myrestore

spec:

storageClassName: mysc

dataSource:

name: mysnapshot kind: VolumeSnapshot

apiGroup: snapshot.storage.k8s.io

volumeMode: Filesystem

accessModes:

- ReadWriteOnce

resources: requests:

storage: 100Gi

## **/** 注意

- 指定的storageClassName必须和dataSource中的快照源卷的StorageClass需一致。
- 克隆卷的容量必须不小于快照容量,建议和快照容量保持一致。

#### 前提条件

系统中已经存在快照,且快照所在的backend存在支持克隆。支持快照创建PVC的存储请参考表2-5和表2-7,支持快照创建PVC的Kubernetes版本请参考表2-3。

#### 操作步骤

**步骤1** 执行以下命令,基于从快照创建卷的配置文件创建PVC。

# kubectl create -f myrestore.yaml

----结束

## 7.2 创建 VolumeSnapshot

在Kubernetes中,**卷快照(VolumeSnapshot**)是一个存储系统上卷的快照。 VolumeSnapshot能力为Kubernetes用户提供了一种标准的方式来在指定时间点复制卷的内容,并且不需要创建全新的卷。 例如,这一功能使得数据库管理员能够在执行编辑或删除之类的修改之前对数据库执行备份。

本章将介绍如何使用华为CSI创建VolumeSnapshot。为了完成创建VolumeSnapshot,需要完成如下三步:

- 检查卷快照依赖组件信息
- 配置VolumeSnapshotClass
- 配置VolumeSnapshot

## 7.2.1 检查卷快照依赖组件信息

如果您需要在容器环境中使用卷快照以及卷快照关联的特性,请通过**3.5 检查卷快照依赖组件**检查您的环境是否部署了卷快照依赖组件以及卷快照Api资源版本信息。

## 7.2.2 配置 VolumeSnapshotClass

**卷快照类(VolumeSnapshotClass**)提供了一种在配置VolumeSnapshot时描述存储 "类"的方法。每个VolumeSnapshotClass都包含"driver"、"deletionPolicy"和 "parameters"字段,在需要动态配置属于该类的VolumeSnapshot时使用。

VolumeSnapshotClass对象的名称很重要,是用户可以请求特定类的方式。 管理员在首次创建VolumeSnapshotClass对象时设置类的名称和其他参数, 对象一旦创建就无法更新。

华为CSI使用的VolumeSnapshotClass示例如下:

● 如果您的环境中支持Api资源版本支持v1,请使用一下示例:

apiVersion: snapshot.storage.k8s.io/v1 kind: VolumeSnapshotClass metadata: name: mysnapclass driver: csi.huawei.com deletionPolicy: Delete

如果您的环境中支持Api资源版本支持v1beta1,请使用一下示例:

apiVersion: snapshot.storage.k8s.io/v1beta1 kind: VolumeSnapshotClass

metadata:

name: mysnapclass

driver: csi.huawei.com deletionPolicy: Delete

● 如果您的环境中支持Api资源版本同时支持v1beta1和v1,我们推荐您使用v1版本。

实际参数可以参考<mark>表7-9</mark>中的说明修改。由于当前华为CSI还不支持在 VolumeSnapshotClass中设置自定义参数(parameters),因此建议之创建一个 VolumeSnapshotClass,供所有快照使用。

#### 表 7-9 VolumeSnapshotClass 参数说明

参数	说明	备注
metadata.n ame	自定义的 VolumeSnapshotCla ss对象名称。	以Kubernetes v1.22.1为例,支持数字、小写字母、中划线(-)和点(.)的组合,并且必须以字母数字字符开头和结尾。
driver	driver标识。必填参 数。	该字段需要指定为安装华为CSI时设置的驱动名。默认的驱动名为"csi.huawei.com"。
deletionPoli cy	快照删除策略。必填 参数。可选值为: • Delete • Retain	<ul> <li>如果删除策略是 Delete,那么存储设备上的快照会和VolumeSnapshotContent对象一起删除。</li> <li>如果删除策略是Retain,那么存储设备上的快照和VolumeSnapshotContent对象都会被保留。</li> </ul>

#### 前提条件

华为CSI支持快照且运行所依赖的卷快照组件CRD已经安装。具体CRD信息请参考3.5 检查卷快照依赖组件章节说明,支持创建VolumeSnapshot的Kubernetes版本请参考表2-3。

#### 操作步骤

**步骤1** 执行以下命令,使用已经创建的VolumeSnapshotClass配置文件创建VolumeSnapshotClass。

# kubectl create -f mysnapclass.yaml

步骤2 执行以下命令,查看已创建的VolumeSnapshotClass信息。

# kubectl get volumesnapshotclass NAME DRIVER DELETIONPOLICY AGE mysnapclass csi.huawei.com Delete 25s

----结束

## 7.2.3 配置 VolumeSnapshot

VolumeSnapshot可以通过两种方式进行制备: 预制备或动态制备。华为CSI当前仅支持动态制备。本章节将说明如何使用华为CSI动态制备VolumeSnapshot。

VolumeSnapshot的配置文件示例如下:

● 如果您的环境中支持Api资源版本支持v1,请使用一下示例:

apiVersion: snapshot.storage.k8s.io/v1

kind: VolumeSnapshot

metadata:

name: mysnapshot

spec:

volumeSnapshotClassName: mysnapclass

source:

persistentVolumeClaimName: mypvc

• 如果您的环境中支持Api资源版本支持v1beta1,请使用一下示例:

apiVersion: snapshot.storage.k8s.io/v1beta1

kind: VolumeSnapshot

metadata:

name: mysnapshot

spec:

volumeSnapshotClassName: mysnapclass

source

persistentVolumeClaimName: mypvc

VolumeSnapshot版本信息请和创建VolumeSnapshotClass使用的版本保持一致。

实际参数可以参考表7-10中的说明修改。

表 7-10 VolumeSnapshot 参数说明

参数	说明	备注
metadata.name	自定义的 VolumeSnapshot对 象名称。	以Kubernetes v1.22.1为例,支持数字、小写字母、中划线(-)和点(.)的组合,并且必须以字母数字字符开头和结尾。
spec.volumeSnapshotCl assName	VolumeSnapshotCl ass对象名称。	
spec.source.persistentVo lumeClaimName	源PVC对象名称。	快照源PVC对应的名称

#### 前提条件

- 源PVC存在,且PVC所在的backend存在支持创建VolumeSnapshot。支持创建VolumeSnapshot的存储请参考表2-5和表2-7,支持创建VolumeSnapshot的Kubernetes版本请参考表2-3。
- 华为CSI运行所依赖的卷快照组件CRD已经安装。具体信息请参考**3.5 检查卷快照** 依赖组件章节说明。
- 系统中已经存在使用华为CSI的VolumeSnapshotClass。

#### 操作步骤

步骤1 执行以下命令,使用已经创建的VolumeSnapshot配置文件创建VolumeSnapshot。

# kubectl create -f mysnapshot.yaml

步骤2 执行以下命令,查看已创建的VolumeSnapshot信息。

# kubectl get volumesnapshot

NAME READYTOUSE SOURCEPVC SOURCESNAPSHOTCONTENT RESTORESIZE SNAPSHOTCLASS SNAPSHOTCONTENT CREATIONTIME AGE

mysnapshot **true** mypvc 100Gi mysnapclass snapcontent-1009af0a-24c2-4435-861c-516224503f2d <invalid> 78s

----结束

## **8** 高级特性

- 8.1 配置ALUA特性
- 8.2 配置存储拓扑感知

## 8.1 配置 ALUA 特性

ALUA(Asymmetric Logical Unit Access,非对称逻辑单元访问),是一种多目标器端口访问模型。在多路径状态下,ALUA标准提供了一种将卷的Active/Passive模型呈现给主机的方式。同时还提供了端口的可访问状态切换接口,可用来实现卷工作控制器切换等。例如,卷在一个控制器故障时,可以将该控制器的端口置为Unavailable,支持ALUA的主机多路径软件收到该状态后,会将I/O切换到另一端控制器。

## 8.1.1 通过 Helm 配置 ALUA 特性

#### 8.1.1.1 配置华为企业存储后端的 ALUA 参数

华为企业存储针对ALUA的配置请参考产品对应的主机连通性指南文档说明。

针对不同的操作系统,ALUA配置可能有所不同。进入**华为技术支持**,在搜索输入框中输入"主机连通性指南",单击搜索。在搜索结果中,选择对应操作系统的主机连通性指南。结合实际需要根据指南的说明进行ALUA配置。华为CSI将在华为存储上对该主机的启动器应用您设置的配置项。

#### □ 说明

已经发放的Pod的节点不会主动更改ALUA信息,需要通过在该节点重新发放Pod才会变更主机 ALUA配置。

## OceanStor V3/V5 系列和 OceanStor Dorado V3 系列存储后端的 ALUA 参数

华为CSI支持的OceanStor V3/V5系列和OceanStor Dorado V3系列存储的ALUA参数见表8-1。

表 8-1 华为 CSI 支持的 OceanStor V3/V5 系列和 OceanStor Dorado V3 系列存储的 ALUA 参数说明

参数名	参数描述	备注
HostName	主机名规则。必填,可 使用正则表达式。	主机名通常使用 cat /etc/hostname 可获取。支持正则表达式方式匹配,如当HostName= "*"时,该条配置对任意主机名的主机生效。可参考《正则表达式》。 当计算节点的主机名可已匹配多条ALUA配置选项,会根据匹配的精确度进行排序,使用第一条ALUA配置选项。排序规则见ALUA配置项匹配主机名的规则。
MULTIPATHTY PE	多路径类型。必填,取值为:	
FAILOVERMO DE	启动器的切换模式。条件必选,取值为:	当使用第三方多路径时该参数才需要指 定。请参考连通性指南的说明,配置启 动器的切换模式。
SPECIALMODE TYPE	启动器的特殊模式类型。条件必选,取值为:  • 0: 特殊模式0  • 1: 特殊模式1  • 2: 特殊模式2  • 3: 特殊模式3	当启动器的切换模式为"特殊模式ALUA"时该参数才需要指定。请参考连通性指南的说明,配置启动器的特殊模式类型。
PATHTYPE	启动器的路径类型。条件必选,取值为: • 0: 优选路径 • 1: 非优选路径	当使用第三方多路径时该参数才需要指 定。请参考连通性指南的说明,配置启 动器的路径类型。

以OceanStor 18500 V5存储对接Red Hat操作系统为例,主机连通性指南见《华为SAN存储在Red Hat系统下的主机连通性指南》。

如下ALUA设置示例,是非双活存储场景下,OceanStor 18500 V5存储的Red Hat操作系统的连通性指南的推荐设置。本例中假设Kubernetes集群中计算节点"myhost01"的操作系统是RHEL 5.x,其他计算节点操作系统均为RHEL 7.x。根据推荐,RHEL 5.x的切换模式应该为"不使用ALUA",RHEL 7.x的切换模式应该为"通用ALUA"。

```
- storage: "oceanstor-san"
 name: "oceanstor-iscsi-155"
  - "https://192.168.129.155:8088"
  - "https://192.168.129.156:8088"
 pools:
   - "StoragePool001"
 parameters:
  protocol: "iscsi"
  portals:
    - "192.168.128.120"
   - "192.168.128.121"
  ALUA:
   ^myhost01$:
     MULTIPATHTYPE: 1
     FAILOVERMODE: 2
     PATHTYPE: 0
     MULTIPATHTYPE: 1
     FAILOVERMODE: 1
     PATHTYPE: 0
```

#### OceanStor V6 和 OceanStor Dorado V6 系列存储后端的 ALUA 参数

华为CSI支持的OceanStor V6和OceanStor Dorado V6系列存储的ALUA参数见表8-2。

#### □ 说明

OceanStor V6和OceanStor Dorado V6系列存储在默认情况下启动器主机访问模式即为"均衡模式",因此不建议对OceanStor V6和OceanStor Dorado V6系列存储配置ALUA参数。

表 8-2 OceanStor V6 和 OceanStor Dorado V6 系列存储的 ALUA 参数说明

参数名	参数描述	备注
HostName	主机名规则。必 填,可使用正则表 达式。	主机名通常使用 cat /etc/hostname 可获取。支持正则表达式方式匹配,如当HostName= "*"时,该条配置对任意主机名的主机生效。可参考《正则表达式》。  当计算节点的主机名可已匹配多条ALUA配置选项,会根据匹配的精确度进行排序,使用第一条ALUA配置选项。排序规则见ALUA配置项匹配主机名的规则。
accessMode	主机访问模式。必填,取值为:  • 0: 均衡模式  • 1: 非对称模式	非双活场景下建议使用均衡模式。当前 华为CSI未支持SAN双活场景,请谨慎使 用非对称模式。
hyperMetroPathO ptimized	双活场景下,主机 在当前阵列的路径 是否优选。取值 为: • 1: 是 • 0: 否	当主机访问模式设置为非对称模式时, 才需要配置该参数。 当前华为CSI未支持SAN双活场景,请谨 慎使用非对称模式。

以OceanStor Dorado 18500 V6存储对接Red Hat操作系统为例,主机连通性指南见《OceanStor Dorado 6.x & OceanStor 6.x在Red Hat下的主机连通性指南》。

如下ALUA设置示例,是非双活存储场景下,OceanStor Dorado 18500 V6存储的Red Hat操作系统的连通性指南的推荐设置。

```
backends:
- storage: "oceanstor-san"
name: "dorado-iscsi-155"
urls:
- "https://192.168.129.155:8088"
- "https://192.168.129.156:8088"
pools:
- "StoragePool001"
parameters:
protocol: "iscsi"
portals:
- "192.168.128.120"
- "192.168.128.121"
ALUA:
*:
accessMode: 0
```

#### ALUA 配置项匹配主机名的规则

● 如果设置的主机名规则精确匹配的业务节点主机名,则使用该主机名规则对应的 ALUA配置项。

如配置项1中主机名规则为"\*",配置项2中的主机名规则为"^myhost01\$"。 当计算节点的主机名是"myhost01"时,精确匹配配置项2,华为CSI将使用配置 项2中的配置应用到存储侧。

● 如果设置的主机名规则无法精确匹配的业务节点主机名,则直接使用正则匹配到的第一条ALUA配置项。

如配置项1中主机名规则为"myhost0[0-9]",配置项2中的主机名规则为"myhost0[5-9]",配置项1的优先级高于配置项2。当计算节点的主机名是"myhost06"时,两个配置项均可以匹配,此时华为CSI将使用配置项1中的配置应用到存储侧。

#### 8.1.1.2 配置分布式存储后端的 ALUA 参数

华为分布式存储针对ALUA的配置请参考产品对应的主机连通性指南文档说明。

针对不同的操作系统,ALUA配置可能有所不同。进入**华为技术支持**,在搜索输入框中输入"主机连通性指南",单击搜索。在搜索结果中,选择对应操作系统的主机连通性指南。结合实际需要根据指南的说明进行ALUA配置。华为CSI将在华为存储上对该主机的启动器应用您设置的配置项。

#### □说明

已经发放的Pod的节点不会主动更改ALUA信息,需要通过在该节点重新发放Pod才会变更主机 ALUA配置。

分布式存储非双活场景,存储系统自身为Active/Active模式,选择"启用ALUA"没有实际意义,建议选择存储默认的"禁用ALUA"。因此不建议对分布式存储配置ALUA参数。

华为CSI支持的分布式存储的ALUA参数见表8-3。

表 8-3 分布式存储 ALUA 参数说明

参数名	参数描述	备注
HostName	HostName的值为worker节 点的主机名,如 HostName1、 HostName2。	主机名通常使用 cat /etc/hostname 可获取。支持正则表达式方式匹配,如当HostName= "*"时,该条配置对任意主机名的主机生效。可参考《正则表达式》。 当计算节点的主机名可已匹配多条ALUA配置选项,会根据匹配的精确度进行排序,使用第一条ALUA配置选项。排序规则见ALUA配置项匹配主机名的规则。
switchoverMode	切换模式。必选,取值为: ● Disable_alua:禁用 ALUA ● Enable_alua:启用 ALUA	非双活场景,存储系统自身为Active/Active模式,选择"启用ALUA"没有实际意义,建议选择"禁用ALUA"。当前华为CSI未支持SAN双活场景,请谨慎启用ALUA。
pathType	路径类型。条件必选,取值为:  optimal_path: 优选路径  non_optimal_path: 非优选路径	切换模式为启动ALUA时需要 设置该选项。

#### ALUA 配置项匹配主机名的规则

如果设置的主机名规则精确匹配的业务节点主机名,则使用该主机名规则对应的 ALUA配置项。

如配置项1中主机名规则为 "\*",配置项2中的主机名规则为 "^myhost01\$"。 当计算节点的主机名是 "myhost01"时,精确匹配配置项2,华为CSI将使用配置 项2中的配置应用到存储侧。

● 如果设置的主机名规则无法精确匹配的业务节点主机名,则直接使用正则匹配到的第一条ALUA配置项。

如配置项1中主机名规则为"myhost0[0-9]",配置项2中的主机名规则为 "myhost0[5-9]",配置项1的优先级高于配置项2。当计算节点的主机名是 "myhost06"时,两个配置项均可以匹配,此时华为CSI将使用配置项1中的配置 应用到存储侧。

## 8.1.2 手动配置 ALUA 特性

# 8.1.2.1 配置华为企业存储后端的 ALUA 参数

华为企业存储针对ALUA的配置请参考产品对应的主机连通性指南文档说明。

针对不同的操作系统,ALUA配置可能有所不同。进入**华为技术支持**,在搜索输入框中输入"主机连通性指南",单击搜索。在搜索结果中,选择对应操作系统的主机连通性指南。结合实际需要根据指南的说明进行ALUA配置。华为CSI将在华为存储上对该主机的启动器应用您设置的配置项。

#### □ 说明

已经发放的Pod的节点不会主动更改ALUA信息,需要通过在该节点重新发放Pod才会变更主机 ALUA配置。

# OceanStor V3/V5 系列和 OceanStor Dorado V3 系列存储后端的 ALUA 参数

华为CSI支持的OceanStor V3/V5系列和OceanStor Dorado V3系列存储的ALUA参数见表8-4。

表 8-4 华为 CSI 支持的 OceanStor V3/V5 系列和 OceanStor Dorado V3 系列存储的 ALUA 参数说明

参数名	参数描述	备注
HostName	主机名规则。必填,可 使用正则表达式。	主机名通常使用 cat /etc/hostname 可获取。支持正则表达式方式匹配,如当HostName= "*"时,该条配置对任意主机名的主机生效。可参考《正则表达式》。
		当计算节点的主机名可已匹配多条ALUA配置选项,会根据匹配的精确度进行排序,使用第一条ALUA配置选项。排序规则见ALUA配置项匹配主机名的规则。
MULTIPATHTY PE	多路径类型。必填,取 值为:	
	● 0: 不使用第三方多 路径	
	● 1:使用第三方多路 径	
FAILOVERMO DE	启动器的切换模式。条 件必选,取值为:	当使用第三方多路径时该参数才需要指 定。请参考连通性指南的说明,配置启
	● 0: 旧版本ALUA	动器的切换模式。 
	● 1: 通用ALUA	
	● 2: 不使用ALUA	
	● 3:特殊模式ALUA	

参数名	参数描述	备注
SPECIALMODE TYPE	启动器的特殊模式类型。条件必选,取值为:  • 0: 特殊模式0  • 1: 特殊模式1  • 2: 特殊模式2  • 3: 特殊模式3	当启动器的切换模式为"特殊模式 ALUA"时该参数才需要指定。请参考连 通性指南的说明,配置启动器的特殊模 式类型。
PATHTYPE	启动器的路径类型。条件必选,取值为:  ①:优选路径  1:非优选路径	当使用第三方多路径时该参数才需要指 定。请参考连通性指南的说明,配置启 动器的路径类型。

以OceanStor 18500 V5存储对接Red Hat操作系统为例,主机连通性指南见《华为SAN存储在Red Hat系统下的主机连通性指南》。

如下ALUA设置示例,是非双活存储场景下,OceanStor 18500 V5存储的Red Hat操作系统的连通性指南的推荐设置。本例中假设Kubernetes集群中计算节点"myhost01"的操作系统是RHEL 5.x,其他计算节点操作系统均为RHEL 7.x。根据推荐,RHEL 5.x的切换模式应该为"不使用ALUA",RHEL 7.x的切换模式应该为"通用ALUA"。

```
{
  "backends": [
  {
    "storage": "oceanstor-san",
    ...
    "parameters": {..., "ALUA": {"^myhost01$": {"MULTIPATHTYPE": 1, "FAILOVERMODE": 2,
    "PATHTYPE": 0}, "*": {"MULTIPATHTYPE": 1, "FAILOVERMODE": 1, "PATHTYPE": 0}}}
  }
}
```

# OceanStor V6 和 OceanStor Dorado V6 系列存储后端的 ALUA 参数

华为CSI支持的OceanStor V6和OceanStor Dorado V6系列存储的ALUA参数见表8-5。

#### □ 说明

OceanStor V6和OceanStor Dorado V6系列存储在默认情况下启动器主机访问模式即为"均衡模式",因此不建议对OceanStor V6和OceanStor Dorado V6系列存储配置ALUA参数。

参数名	参数描述	备注
HostName	主机名规则。必 填,可使用正则表 达式。	主机名通常使用 cat /etc/hostname 可获取。支持正则表达式方式匹配,如当HostName= "*"时,该条配置对任意主机名的主机生效。可参考《正则表达式》。
		当计算节点的主机名可已匹配多条ALUA配置选项,会根据匹配的精确度进行排序,使用第一条ALUA配置选项。排序规则见ALUA配置项匹配主机名的规则。
accessMode	主机访问模式。必 填,取值为: • 0: 均衡模式 • 1: 非对称模式	非双活场景下建议使用均衡模式。当前 华为CSI未支持SAN双活场景,请谨慎使 用非对称模式。
hyperMetroPathO ptimized	双活场景下,主机 在当前阵列的路径 是否优选。取值 为: • 1: 是 • 0: 否	当主机访问模式设置为非对称模式时, 才需要配置该参数。 当前华为CSI未支持SAN双活场景,请谨 慎使用非对称模式。

表 8-5 OceanStor V6 和 OceanStor Dorado V6 系列存储的 ALUA 参数说明

以OceanStor Dorado 18500 V6存储对接Red Hat操作系统为例,主机连通性指南见《OceanStor Dorado 6.x & OceanStor 6.x在Red Hat下的主机连通性指南》。

如下ALUA设置示例,是非双活存储场景下,OceanStor Dorado 18500 V6存储的Red Hat操作系统的连通性指南的推荐设置。

```
{
  "backends": [
  {
    "storage": "oceanstor-san",
    ...
    "parameters": {..., "ALUA": {"*": {"accessMode": 0}}}
  }
}
```

## ALUA 配置项匹配主机名的规则

如果设置的主机名规则精确匹配的业务节点主机名,则使用该主机名规则对应的 ALUA配置项。

如配置项1中主机名规则为"\*",配置项2中的主机名规则为"^myhost01\$"。 当计算节点的主机名是"myhost01"时,精确匹配配置项2,华为CSI将使用配置 项2中的配置应用到存储侧。

如果设置的主机名规则无法精确匹配的业务节点主机名,则直接使用正则匹配到的第一条ALUA配置项。

如配置项1中主机名规则为"myhost0[0-9]",配置项2中的主机名规则为 "myhost0[5-9]",配置项1的优先级高于配置项2。当计算节点的主机名是 "myhost06"时,两个配置项均可以匹配,此时华为CSI将使用配置项1中的配置应用到存储侧。

### 8.1.2.2 配置分布式存储后端的 ALUA 参数

华为分布式存储针对ALUA的配置请参考产品对应的主机连通性指南文档说明。

针对不同的操作系统,ALUA配置可能有所不同。进入**华为技术支持**,在搜索输入框中输入"主机连通性指南",单击搜索。在搜索结果中,选择对应操作系统的主机连通性指南。结合实际需要根据指南的说明进行ALUA配置。华为CSI将在华为存储上对该主机的启动器应用您设置的配置项。

#### □ 说明

已经发放的Pod的节点不会主动更改ALUA信息,需要通过在该节点重新发放Pod才会变更主机 ALUA配置。

分布式存储非双活场景,存储系统自身为Active/Active模式,选择"启用ALUA"没有实际意义,建议选择存储默认的"禁用ALUA"。因此不建议对分布式存储配置ALUA参数。

华为CSI支持的分布式存储的ALUA参数见表8-6。

表 8-6 分布式存储 ALUA 参数说明

参数名	参数描述	备注
HostName	HostName的值为worker节 点的主机名,如 HostName1、 HostName2。	主机名通常使用 cat /etc/hostname 可获取。支持正则表达式方式匹配,如当HostName= "*"时,该条配置对任意主机名的主机生效。可参考《正则表达式》。 当计算节点的主机名可已匹配多条ALUA配置选项,会根据匹配的精确度进行排序,使用第一条ALUA配置选项。排序规则见ALUA配置项匹配主机名的规则。
switchoverMode	切换模式。必选,取值为:  • Disable_alua: 禁用 ALUA  • Enable_alua: 启用 ALUA	非双活场景,存储系统自身为Active/Active模式,选择"启用ALUA"没有实际意义,建议选择"禁用ALUA"。当前华为CSI未支持SAN双活场景,请谨慎启用ALUA。
pathType	路径类型。条件必选,取值为: • optimal_path: 优选路 径 • non_optimal_path: 非 优选路径	切换模式为启动ALUA时需要 设置该选项。

## ALUA 配置项匹配主机名的规则

● 如果设置的主机名规则精确匹配的业务节点主机名,则使用该主机名规则对应的 ALUA配置项。

如配置项1中主机名规则为 "\*",配置项2中的主机名规则为 "^myhost01\$"。 当计算节点的主机名是 "myhost01"时,精确匹配配置项2,华为CSI将使用配置 项2中的配置应用到存储侧。

● 如果设置的主机名规则无法精确匹配的业务节点主机名,则直接使用正则匹配到的第一条ALUA配置项。

如配置项1中主机名规则为"myhost0[0-9]",配置项2中的主机名规则为 "myhost0[5-9]",配置项1的优先级高于配置项2。当计算节点的主机名是 "myhost06"时,两个配置项均可以匹配,此时华为CSI将使用配置项1中的配置 应用到存储侧。

# 8.2 配置存储拓扑感知

在Kubernetes集群中,可以根据节点的拓扑标签以及存储后端支持的拓扑能力调度和 发放资源。

## 前提条件

需要在集群中的worker节点完成拓扑的标签配置,标签配置方法如下:

- 1. 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的 任意master节点。
- 2. 执行kubectl get node命令,查看当前集群中的worker节点信息。

```
# kubectl get node
NAME STATUS ROLES AGE VERSION
node01 Ready controlplane,etcd,worker 42d v1.22.3
node02 Ready worker 42d v1.22.3
node03 Ready worker 42d v1.22.3
```

# kubectl label node node01 topology.kubernetes.io/zone=ChengDu node/node01 labeled

#### 表 8-7 参数说明

参数名	参数描述	备注
<key></key>	拓扑标签的唯一标 识。	可支持配置: zone, region, protocol. <i><protocol></protocol></i> 其中 <i><protocol></protocol></i> 可支持配置iscsi, nfs, fc, roce。
<value></value>	拓扑标签的参数值。	"key"如果是"zone","region", "value"值为自定义参数。 "key"如果是protocol. <i><protocol></protocol></i> , "value"值固定为 "csi.huawei.com"。

#### □说明

- 拓扑标签必须以topology.kubernetes.io开头。拓扑标签示例:
  - 示例1: topology.kubernetes.io/region=China-west
  - 示例2: topology.kubernetes.io/zone=ChengDu
  - 示例3: topology.kubernetes.io/protocol.iscsi=csi.huawei.com
  - 示例4: topology.kubernetes.io/protocol.fc=csi.huawei.com
- 同一节点上拓扑标签中同一个key只能支持一个value值。
- 如果同一节点上拓扑标签中同时配置多个protocol,选择后端时,只需要满足其中一个 protocol即可。
- 如果同一节点上拓扑标签中同时配置region和zone,选择后端时,需要满足全部。
- 4. 执行kubectl get nodes -o=jsonpath='{range .items[\*]}[{.metadata.name}, {.metadata.labels}]{"\n"}{end}' | grep --color "topology.kubernetes.io"命令,查看当前集群中所有worker节点的标签信息。

# kubectl get nodes -o=jsonpath='{range .items[\*]}{{.metadata.name}, {.metadata.labels}]{"\n"}{end}' | grep --color "topology.kubernetes.io" [node01, {"beta.kubernetes.io/arch":"amd64","beta.kubernetes.io/os":"linux","kubernetes.io/arch":"amd64","kubernetes.io/hostname":"node01","kubernetes.io/os":"linux","node-role.kubernetes.io/controlplane":"true","node-role.kubernetes.io/etcd":"true","node-role.kubernetes.io/worker":"true","topology.kubernetes.io/zone":"ChengDu"}]

# 8.2.1 通过 Helm 配置存储拓扑感知

#### 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 进入Helm工程的目录下,如果无法找到之前的Helm工程,则将组件包中的helm目录 拷贝到master节点的任意目录下,组件包路径请参考表3-1。
- **步骤3** 备份安装CSI时使用的values.yaml文件,如果上次安装时使用的values.yaml文件无法 找到,可通过**helm get values** *helm-huawei-csi* **-n** *huawei-csi* **-a**命令查询。 # cp values.yaml values.yaml.bak
- **步骤4** 执行**vi** *values.yaml*命令打开文件,按需求配置多个后端,示例如下所示。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

```
backends:
- storage: "oceanstor-nas"
name: "storage"
urls:
- "https://*.*.*:8088"
- "https://*.*.*:8088"
pools:
- "***"
- "***"
parameters:
protocol: "nfs"
portals:
- "**.**"
supportedTopologies:
- {"topology.kubernetes.io/region": "China-west", "topology.kubernetes.io/zone": "ChengDu"}
- {"topology.kubernetes.io/region": "China-south", "topology.kubernetes.io/zone": "ShenZhen"}
```

步骤5 执行helm upgrade helm-huawei-csi ./ -n huawei-csi -f values.yaml命令升级Helm chart。升级命令将更新Deployment、daemonset和RBAC资源,但是secret资源不会 更新,如果后端信息发生更改,则必须手动更新secret,更新操作请参考9.1 更新CSI 上配置的存储用户名或密码。

其中,helm-huawei-csi为自定义的chart名称,huawei-csi为自定义的命名空间。

# helm upgrade helm-huawei-csi ./ -n huawei-csi

Release "helm-huawei-csi" has been upgraded. Happy Helming!

NAME: helm-huawei-csi

LAST DEPLOYED: Thu Jun 9 07:58:15 2022

NAMESPACE: huawei-csi STATUS: deployed **REVISION: 2** TEST SUITE: None

步骤6 执行vi StorageClass.yaml命令,修改yaml文件。按l或Insert进入编辑状态,在yaml 文件下增加相关参数,详细参数说明请参见表8-8。修改完成后,按Esc,并输 入:wq!,保存修改。

在StorageClass.yaml文件中添加以下配置项。

示例1: 在StorageClass中配置zone和region信息

kind: StorageClass

apiVersion: storage.k8s.io/v1

metadata:

name: example-storageclass

provisioner: csi.huawei.com

parameters: volumeType: lun allocType: thin

volumeBindingMode: WaitForFirstConsumer

allowedTopologies:

- matchLabelExpressions:
- key: topology.kubernetes.io/zone values:
- ChengDu
- key: topology.kubernetes.io/region
- values: - China-west
- 示例2: 在StorageClass中配置协议信息

kind: StorageClass

apiVersion: storage.k8s.io/v1

metadata:

name: protocol-example-storageclass

provisioner: csi.huawei.com

parameters:

volumeType: lun

allocType: thin

volumeBindingMode: WaitForFirstConsumer

allowedTopologies:

- matchLabelExpressions:
- key: topology.kubernetes.io/protocol.iscsi values:
  - csi.huawei.com

#### 表 8-8 参数说明

参数名	参数描述	备注
volumeBindin gMode	PersistentVolume 绑定方式,用于控 制何时进行 PersistentVolume 动态资源调配和绑 定。	可配置"WaitForFirstConsumer"或 "Immediate" "WaitForFirstConsumer":表示延迟 PersistentVolume的绑定和调配,直到创建 使用PVC的Pod。 "Immediate":表示创建PVC后,立即发 生PersistentVolume绑定和调配。

参数名	参数描述	备注
allowedTopol ogies.matchLa belExpression s	拓扑信息标签,用于过滤CSI后端和Kubernetes节点。如果匹配失败,会导致PVC或Pod无法创建。配置时需要同时按照固定格式配置"key"和"value".	"key":可支持配置 "topology.kubernetes.io/zone", "topology.kubernetes.io/region", topology.kubernetes.io/ protocol. <pre>/protocol&gt;,其中<pre>protocol&gt;为协议类型,例如: iscsi, fc, nfs等。  "value": "key"如果是"topology.kubernetes.io/zone", "topology.kubernetes.io/region", "value"值需要和前提条件中设置的拓扑标签保持一致。 "key"如果是topology.kubernetes.io/protocol.</pre>/ "value"值固定为 "csi.huawei.com"</pre>

步骤7 执行以下命令,基于该yaml文件创建StorageClass。

# kubectl create -f StorgeClass.yaml

步骤8 使用该StorageClass创建具有拓扑能力的PVC,详细操作请参考7.1.1.1.2 配置PVC。

----结束

# 8.2.2 手动配置存储拓扑感知

#### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

**步骤2** 执行**vi** *huawei-csi-configmap.yaml*命令,修改yaml文件。按**l**或**lnsert**进入编辑状态,修改相关参数。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

多个后端间使用","进行分割,每个后端内容可以参考**4.2.1 创建华为CSI运行时需要的ConfigMap**。

在huawei-csi-configmap.yaml文件的backends配置段中,添加**supportedTopologies** 字段,用于配置每个后端支持的拓扑信息。以下是后端示例。

```
{
    "backends":[
    {
        "storage": "oceanstor-san",
        ...
        "parameters": {"protocol": "iscsi", "portals": ["192.168.125.22", "192.168.125.23"]},
        "supportedTopologies": [
            {"topology.kubernetes.io/region": "China-west", "topology.kubernetes.io/zone": "ChengDu"},
            {"topology.kubernetes.io/region": "China-south", "topology.kubernetes.io/zone": "ShenZhen"}]
    }
}
```

#### □ 说明

- "supportedTopologies"是一个列表,列表中的每个元素是一个字典。

**步骤3** 执行**kubectl create -f** *huawei-csi-configmap.yaml*命令,创建huawei-csi-configmap。

# kubectl create -f huawei-csi-configmap.yaml

# kubectl get configmap -n huawei-csi | grep huawei-csi-configmap huawei-csi-configmap 1 5s

步骤5 启动huawei-csi服务,详细请参见4.2.2 启动huawei-csi服务。

**步骤6** 执行**vi** *StorageClass.yaml*命令,修改yaml文件。按**I**或**Insert**进入编辑状态,在yaml 文件下增加相关参数,详细参数说明请参见<mark>表8-9</mark>。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

在StorageClass.yaml文件中添加以下配置项。

• 示例1: 在StorageClass中配置zone和region信息

kind: StorageClass apiVersion: storage.k8s.io/v1 metadata:

name: example-storageclass provisioner: csi.huawei.com parameters:

volumeType: lun allocType: thin

volumeBindingMode: WaitForFirstConsumer

allowedTopologies:

- matchLabelExpressions:
- key: topology.kubernetes.io/zone values:
- ChengDu
- key: topology.kubernetes.io/region values:
- China-west
- 示例2: 在StorageClass中配置协议信息

kind: StorageClass

apiVersion: storage.k8s.io/v1

metadata:

name: protocol-example-storageclass

provisioner: csi.huawei.com

parameters: volumeType: lun allocType: thin

volumeBindingMode: WaitForFirstConsumer allowedTopologies:

- matchLabelExpressions:
- key: topology.kubernetes.io/protocol.iscsi values:
  - csi.huawei.com

表 8-9 参数说明

参数名	参数描述	备注
volumeBindin gMode	PersistentVolume 绑定方式,用于控 制何时进行 PersistentVolume 动态资源调配和绑 定。	可配置"WaitForFirstConsumer"或 "Immediate" "WaitForFirstConsumer":表示延迟 PersistentVolume的绑定和调配,直到创建 使用PVC的Pod。 "Immediate":表示创建PVC后,立即发 生PersistentVolume绑定和调配。
allowedTopol ogies.matchLa belExpression s	拓扑信息标签,用于过滤CSI后端和Kubernetes节点。如果匹配失败,会导致PVC或Pod无法创建。配置时需要同时按照固定格式配置"key"和"value".	"key":可支持配置 "topology.kubernetes.io/zone", "topology.kubernetes.io/region", topology.kubernetes.io/ protocol. <i><protocol></protocol></i> , 其中 <i><protocol></protocol></i> 为协议类型,例如:iscsi, fc, nfs等。
		"value": "key"如果是"topology.kubernetes.io/ zone","topology.kubernetes.io/ region","value"值需要和 <mark>前提条件中设</mark> 置的拓扑标签保持一致。 "key"如果是topology.kubernetes.io/ protocol. resi.huawei.com"

步骤7 执行以下命令,基于该yaml文件创建StorageClass。

# kubectl create -f StorgeClass.yaml

步骤8 使用该StorageClass创建具有拓扑能力的PVC,详细操作请参考7.1.1.1.2 配置PVC。

----结束

# 9 常用操作

- 9.1 更新CSI上配置的存储用户名或密码
- 9.2 更新huawei-csi的configmap对象
- 9.3 为huawei-csi新增后端
- 9.4 更新huawei-csi-controller服务
- 9.5 更新huawei-csi-node服务
- 9.6 修改日志输出模式
- 9.7 开启ReadWriteOncePod功能门
- 9.8 配置非root用户访问Kubernetes集群

# 9.1 更新 CSI 上配置的存储用户名或密码

当存储设备的用户名或密码发生变更时,需要更新CSI上的配置信息,否则huawei-csi服务无法正常工作。

# 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- **步骤2** 执行**chmod +x secretUpdate**命令,给secretUpdate添加可执行权限。

# chmod +x secretUpdate

步骤3 执行./secretUpdate -namespace=huawei-csi -logFileDir=/var/log/huawei/命令,其中huawei-csi 替换为华为CSI实际运行的命名空间,不添加该参数默认为huawei-csi,/var/log/huawei/替换为用户有读写权限的存放华为CSI日志的目录,不添加该参数则默认为/var/log/huawei/,运行secretUpdate工具后,根据界面提示输入需要配置backend的编号。Configured为false时表示这个backend尚未配置,为true时表示这个backend已配置。

```
# ./secretUpdate
Getting backend configuration information.......

Number Configured BackendName Urls
true strage-backend [https://192.168.125.25:8088]
true strage-backend-02 [https://192.168.125.26:8088]
```

```
        3
        true
        strage-backend-03
        [https://192.168.125.27:8088]

        4
        true
        strage-backend-04
        [https://192.168.125.28:8088]

        5
        true
        strage-backend-05
        [https://192.168.125.29:28443]

        6
        true
        strage-backend-06
        [https://192.168.125.30:28443]
```

Please enter the backend number to configure (Enter 'exit' to exit):3

#### 步骤4 根据界面提示输入用户名和密码来更新secret对象。

Name:strage-backend-03

Urls:[https://192.168.125.27:8088]

Please enter this backend user name: admin

Please enter this backend password:

Verifying user name and password. Please wait.....

The acount information of the backend strage-backend-03 has been configured successfully.

#### 步骤5 配置完成后,输入exit退出并保存配置。

Please enter the backend number to configure (Enter 'exit' to exit): exit Saving configuration. Please wait...........
The configuration is saved successfully.

#### 步骤6 执行以下命令,重启huawei-csi-controller服务。

# kubectl get deployment huawei-csi-controller -o yaml -n=huawei-csi | kubectl replace --force -f -

#### 步骤7 执行以下命令,重启huawei-csi-node服务。

# kubectl get daemonset huawei-csi-node -o yaml -n=huawei-csi | kubectl replace --force -f -

#### 步骤8 执行kubectl get pod -A | grep huawei命令检查服务是否重启成功。

```
# kubectl get pod -A | grep huawei
huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 Running 0 14s
huawei-csi huawei-csi-node-g6f7z 3/3 Running 0 14s
```

----结束

# 9.2 更新 huawei-csi 的 configmap 对象

当您需要更改已有的业务IP地址时,执行此操作。

# 9.2.1 使用 Helm 更新 configmap 对象

#### 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- **步骤2** 备份安装CSI时使用的values.yaml文件,可通过**helm get values** *helm-huawei-csi* **-n** *huawei-csi* **-a** > values.yaml.bak 命令备份,其中,helm-huawei-csi为安装时自定义的Helm Chart名称,huawei-csi为安装时自定义的Helm Chart命名空间。

# helm get values helm-huawei-csi -n huawei-csi -a > values.yaml.bak

步骤3 backends参数请谨慎修改,如需修改请进入/helm/esdk 目录,执行**vi** *values.yaml*命令打开文件,修改backends信息,具体修改请参考**Kubernetes配置项说明**,修改完成后,按**Esc**,并输入:**wq!**,保存修改。其中组件包路径请参考**表3-1** 

# vi values.yaml

步骤4 执行helm upgrade helm-huawei-csi./ -n huawei-csi-f values.yaml命令升级Helm chart。升级命令将更新Deployment、daemonset和RBAC资源,但是secret资源不会更新,如果后端信息发生更改,则必须手动更新secret,更新操作请参考9.1 更新CSI上配置的存储用户名或密码。

其中,helm-huawei-csi为自定义的chart名称,./表示使用当前目录下的Helm工程,huawei-csi为自定义的命名空间,-f values.yaml表示使用指定values.yaml文件升级。

```
# helm upgrade helm-huawei-csi / -n huawei-csi -f values.yaml
Release "helm-huawei-csi" has been upgraded. Happy Helming!
NAME: helm-huawei-csi
LAST DEPLOYED: Thu Jun 9 07:58:15 2022
NAMESPACE: huawei-csi
STATUS: deployed
REVISION: 2
TEST SUITE: None
```

步骤5 执行以下命令,重启huawei-csi-controller服务。

# kubectl get deployment huawei-csi-controller -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤6 执行以下命令,重启huawei-csi-node服务。

# kubectl get daemonset huawei-csi-node -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤7 执行kubectl get pod -n huawei-csi命令检查服务是否重启成功。

```
# kubectl get pod -n huawei-csi
huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 Running 0 14s
huawei-csi huawei-csi-node-g6f7z 3/3 Running 0 14s
```

----结束

# 9.2.2 手动更新 configmap 对象

#### 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 执行kubectl edit configmap huawei-csi-configmap -n huawei-csi命令,按l或 Insert进入编辑状态,修改相关参数。以iSCSI组网为例,详细参数说明请参见<mark>表9-1</mark>。 修改完成后,按Esc,并输入:wq!,保存修改。

#### 表 9-1 配置项描述

配置项	参数格式	描述	备注
data."csi.json".backe nds	列表	必填,待接 入后端存储 设备列表。	配置的后端存储设备数量没有限制。 单个后端存储设备支持配置的字段,请参考表9-2。

### 表 9-2 后端存储设备配置项描述

配置项	参数格式	描述	备注
	参数恰式	畑迩	<b>甘</b> 注
storage	字符串	必填,待接入存 储设备的类型。	对接分布式存储SAN场景固定填写"fusionstorage-san"。
name	字符串	存储后端名。	<ul><li>自定义字符串,支持大小写字母、数字、中划线组合。</li><li>该参数不支持修改。</li></ul>
urls	列表	必填,待接入的 存储设备的管理 URL。	支持同一存储设备的一个或多个管理URL,用逗号分割,当前仅支持IPv4。例如: https://192.168.125.20:8088  说明 1个存储设备有多个控制器,每个控制器有1个管理URL,所以1个存储设备可能存在多个管理URL。
pools	列表	必填,被使用的 待接入存储设备 上的存储池名 称。	<ul> <li>支持同一存储设备上的一个或多个存储池,用逗号分割。</li> <li>当前只支持新增存储池。</li> <li>可通过登录DeviceManager获取支持块存储服务的存储池。</li> </ul>
parameters	字典	iSCSI场景的可变 参数。	iSCSI场景protocol参数固定填写 "iscsi"。 portals参数填写待接入存储设备的iSCSI业务IP地址,多个业务IP地址请用逗号分割。 iSCSI业务IP地址可通过登录DeviceManager获取。以OceanStor Dorado 6.x系列为例:在DeviceManager管理界面,选择"服务>网络>逻辑端口",获取数据协议类型为iSCSI的IP(其它系列请参照相应的操作说明进行获取)。

- 步骤3 若后端参数storage、name或urls有修改,需要更新存储的用户名或密码。详细操作请参见9.1 更新CSI上配置的存储用户名或密码。
- 步骤4 执行以下命令,重启huawei-csi-controller服务。
  # kubectl get deployment huawei-csi-controller -o yaml -n=huawei-csi | kubectl replace --force -f -
- **步骤5** 执行以下命令,重启huawei-csi-node服务。

# kubectl get daemonset huawei-csi-node -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤6 执行kubectl get pod -A | grep huawei命令,检查服务是否重启成功。

```
# kubectl get pod -A | grep huawei
huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 Running 0 14s
huawei-csi huawei-csi-node-g6f7z 3/3 Running 0 14s
```

----结束

# 9.3 为 huawei-csi 新增后端

当您需要新增存储设备或存储池当做独立的后端时,执行此操作。

# 9.3.1 使用 Helm 新增后端

#### 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- **步骤2** 备份安装CSI时使用的values.yaml文件,可通过**helm get values** *helm-huawei-csi* **-n** *huawei-csi* **-a** > values.yaml.bak 命令备份,其中,helm-huawei-csi为安装时自定义的Helm Chart名称,huawei-csi为安装时自定义的Helm Chart命名空间。

# helm get values helm-huawei-csi -n huawei-csi -a > values.yaml.bak

步骤3 进入/helm/esdk 目录,执行**vi** *values.yaml*命令打开配置文件,在backends下新增后端配置,示例如下所示,修改完成后,按**Esc**,并输入:**wq!**,保存修改。其中组件包路径请参考表3-1。

```
backends:
  - storage: "oceanstor-san"
  name: "***"
  urls:
   - "https://*.*.*:8088"
   pools:
  parameters:
   protocol: "iscsi"
    portals:
     - "*.*.*"
  - storage: "oceanstor-nas"
  name: "***"
  urls:
    - "https://*.*.*:8088"
  pools:
  parameters:
    protocol: "nfs"
    portals:
     - "*.*.*
```

**步骤4** 执行**helm upgrade** *helm-huawei-csi* ./ **-n** *huawei-csi* **-f** *values.yaml*命令升级Helm chart。升级命令将更新Deployment、daemonset和RBAC资源,但是secret资源不会

更新,如果后端信息发生更改,则必须手动更新secret,更新操作请参考**9.1 更新CSI**上配置的存储用户名或密码。

其中,helm-huawei-csi为自定义的chart名称,./表示使用当前目录下的Helm工程,huawei-csi为自定义的命名空间,-f values.yaml表示使用指定values.yaml文件升级。

# helm upgrade helm-huawei-csi / -n huawei-csi -f values.yaml Release "helm-huawei-csi" has been upgraded. Happy Helming!

NAME: helm-huawei-csi

LAST DEPLOYED: Thu Jun 9 07:58:15 2022

NAMESPACE: huawei-csi STATUS: deployed REVISION: 2 TEST SUITE: None

**步骤5** 执行以下命令,重启huawei-csi-controller服务。

# kubectl get deployment huawei-csi-controller -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤6 执行以下命令,重启huawei-csi-node服务。

# kubectl get daemonset huawei-csi-node -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤7 执行kubectl get pod -n huawei-csi命令检查服务是否重启成功。

# kubectl get pod -n huawei-csi huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 **Running** 0 14s huawei-csi huawei-csi-node-g6f7z 3/3 **Running** 0 14s

----结束

# 9.3.2 手动新增后端

# 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 配置多个后端,详细操作请参见4.2.1.12 通过CSI对接多个后端存储。

步骤3 为新增的后端配置账号,详细操作请参见9.1 更新CSI上配置的存储用户名或密码。

**步骤4** 执行以下命令,重启huawei-csi-controller服务。

# kubectl get deployment huawei-csi-controller -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤5 执行以下命令,重启huawei-csi-node服务。

# kubectl get daemonset huawei-csi-node -o yaml -n=huawei-csi | kubectl replace --force -f -

步骤6 执行kubectl get pod -A | grep huawei命令,检查服务是否重启成功。

# kubectl get pod -A | grep huawei huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 **Running** 0 14s huawei-csi huawei-csi-node-g6f7z 3/3 **Running** 0 14s

----结束

# 9.4 更新 huawei-csi-controller 服务

当您需要更新huawei-csi-controller服务时, 例如添加快照功能或者扩容功能,执行此操作。

# 9.4.1 使用 Helm 更新 controller 服务

#### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

**步骤2** 备份安装CSI时使用的values.yaml文件,可通过**helm get values** *helm-huawei-csi* **-n** *huawei-csi* **-a** > values.yaml.bak 命令备份,其中,helm-huawei-csi为安装时自定义的Helm Chart名称,huawei-csi为安装时自定义的Helm Chart命名空间。

# helm get values helm-huawei-csi -n huawei-csi -a > values.yaml.bak

**步骤3** 进入/helm/esdk 目录,执行**vi** *values.yaml*命令打开文件,修改controller相关参数,示例如下所示。修改完成后,按**Esc**,并输入:**wq!**,保存修改。其中组件包路径请参考表3-1。

表3-1。 kubernetes: namespace: huawei-csi # The image name and tag for the attacher, provisioner and registrar sidecars. These must match the appropriate Kubernetes version. sidecar: csiAttacher: k8s.gcr.io/sig-storage/csi-attacher:v3.3.0 csiProvisioner: k8s.gcr.io/sig-storage/csi-provisioner:v3.0.0 csiResizer: k8s.gcr.io/sig-storage/csi-resizer:v1.3.0 livenessProbe: k8s.gcr.io/sig-storage/livenessprobe:v2.5.0 csiSnapshotter: k8s.gcr.io/sig-storage/csi-snapshotter:v4.2.1 snapshotController: k8s.gcr.io/sig-storage/snapshot-controller:v4.2.1 # The image name and tag for the Huawei CSI Service container # Replace the appropriate tag name huaweiCSIService: huawei-csi:3.2.0 # The CSI driver parameter configuration csi\_driver: driverName: csi.huawei.com endpoint: /csi/csi.sock backendUpdateInterval: 60 controllerLogging: module: file level: info fileDir: /var/log/huawei fileSize: 20M maxBackups: 9 huaweiCsiController: replicas: 1 # Default number of controller replicas # Default image pull policy for sidecar container images sidecarImagePullPolicy: "IfNotPresent" # Default image pull policy for Huawei plugin container images huaweiImagePullPolicy: "IfNotPresent" # Flag to enable or disable snapshot (Optional) snapshot: enable: true # Flag to enable or disable resize (Optional) resizer: enable: true

步骤4 执行helm upgrade helm-huawei-csi // -n huawei-csi命令升级Helm chart。

其中,helm-huawei-csi为需要升级的chart名称,./为使用当前目录下Helm工程,huawei-csi为chart所在的命名空间。

# helm upgrade helm-huawei-csi ./ -n huawei-csi Release "helm-huawei-csi" has been upgraded. Happy Helming! NAME: helm-huawei-csi LAST DEPLOYED: Thu Jun 9 07:58:15 2022 NAMESPACE: huawei-csi

STATUS: deployed REVISION: 2 TEST SUITE: None

----结束

# 9.4.2 手动更新 controller 服务

### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

步骤2 卸载huawei-csi-controller服务,请参见5.2.2 卸载huawei-csi-controller服务。

步骤3 删除RBAC权限,请参见5.2.5 删除RBAC权限。

步骤4 创建RBAC权限,请参见步骤4。

步骤5 启动controller服务,请参见步骤5。

步骤6 完成huawei-csi服务部署后,可执行kubectl get pod -A | grep huawei-csi-controller命令检查服务是否启动。

# kubectl get pod -A | grep huawei-csi-controller huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 Running 0 14s

----结束

# 9.5 更新 huawei-csi-node 服务

当您需要更新huawei-csi-node服务时,执行此操作。

# 9.5.1 使用 Helm 更新 node 服务

#### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

**步骤2** 备份安装CSI时使用的values.yaml文件,可通过**helm get values** *helm-huawei-csi* **-n** *huawei-csi* **-a** > values.yaml.bak 命令备份,其中,helm-huawei-csi为安装时自定义的Helm Chart名称,huawei-csi为安装时自定义的Helm Chart命名空间。

# helm get values helm-huawei-csi -n huawei-csi -a > values.yaml.bak

步骤3 进入/helm/esdk 目录,执行vi values.yaml命令打开文件,修改node相关参数,示例如下所示。修改完成后,按Esc,并输入:wq!,保存修改。其中组件包路径请参考表3-1。

kubernetes:

namespace: huawei-csi

images:

# The image name and tag for the attacher, provisioner and registrar sidecars. These must match the appropriate Kubernetes version.

```
livenessProbe: k8s.gcr.io/sig-storage/livenessprobe:v2.5.0
  registrar: k8s.gcr.io/sig-storage/csi-node-driver-registrar:v2.3.0
 # The image name and tag for the Huawei CSI Service container
 # Replace the appropriate tag name
 huaweiCSIService: huawei-csi:3.2.0
# The CSI driver parameter configuration
csi_driver:
 driverName: csi.huawei.com
 endpoint: /csi/csi.sock
 connectorThreads: 4
 volumeUseMultipath: true # Flag to enable or disable volume multipath access
 scsiMultipathType: DM-multipath #Required, if volume-use-multipath is set to TRUE
 nvmeMultipathType: HW-UltraPath-NVMe #Required, if volume-use-multipath is set to TRUE
 scanVolumeTimeout: 3
 backendUpdateInterval: 60
 nodeLogging:
  module: file
  level: info
  fileDir: /var/log/huawei
  fileSize: 20M
  maxBackups: 9
# Default image pull policy for sidecar container images
sidecarImagePullPolicy: "IfNotPresent"
# Default image pull policy for Huawei plugin container images
huaweiImagePullPolicy: "IfNotPresent"
```

#### 步骤4 执行helm upgrade helm-huawei-csi./ -n huawei-csi命令升级Helm chart。

其中,helm-huawei-csi为需要升级的chart名称,./为使用当前目录下Helm工程,huawei-csi为chart所在的命名空间。

# helm upgrade helm-huawei-csi ./ -n huawei-csi Release "helm-huawei-csi" has been upgraded. Happy Helming! NAME: helm-huawei-csi LAST DEPLOYED: Thu Jun 9 07:58:15 2022 NAMESPACE: huawei-csi STATUS: deployed REVISION: 2 TEST SUITE: None

#### ----结束

# 9.5.2 手动更新 node 服务

#### 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- 步骤2 卸载huawei-csi-node服务,请参见5.2.1 卸载huawei-csi-node服务。
- 步骤3 安装新的huawei-csi-node服务,请参见步骤6。
- 步骤4 完成huawei-csi服务部署后,可执行kubectl get pod -A | grep huawei-csi-node命令检查服务是否启动。

# kubectl get pod -A | grep huawei-csi-node huawei-csi huawei-csi-node-g6f7z 3/3 Running 0 14s

#### ----结束

# 9.6 修改日志输出模式

huawei-csi支持两种日志输出模式,分别是file和console。file指的是输出到固定的日志目录(例如:/var/log/huawei);console指的是输出到容器标准目录。用户可以根据自身需求自行设置日志输出模式,默认为file.

# 9.6.1 使用 Helm 修改 controller 或 node 服务的日志输出模式

## 操作步骤

- **步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。
- 步骤2 进入Helm工程的目录下,如果无法找到之前的Helm工程,则将组件包中的helm目录 拷贝到master节点的任意目录下,组件包路径请参考表3-1。
- 步骤3 备份安装CSI时使用的values.yaml文件,如果上次安装时使用的values.yaml文件无法找到,可通过helm get values helm-huawei-csi -n huawei-csi -a命令查询。

# cp values.yaml values.yaml.bak

**步骤4** 执行**vi** *values.yaml*命令打开文件,修改controller或node的日志相关参数,示例如下所示。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

```
# The CSI driver parameter configuration
csi_driver:
controllerLogging:
    module: file
    level: info
    fileDir: /var/log/huawei
    fileSize: 20M
    maxBackups: 9
nodeLogging:
    module: file
    level: info
    fileDir: /var/log/huawei
    fileSize: 20M
    maxBackups: 9
```

步骤5 执行helm upgrade helm-huawei-csi / -n huawei-csi命令升级Helm chart。

其中,helm-huawei-csi为需要升级的chart名称,./为使用当前目录下Helm工程,huawei-csi为chart所在的命名空间。

```
# helm upgrade helm-huawei-csi ./ -n huawei-csi
Release "helm-huawei-csi" has been upgraded. Happy Helming!
NAME: helm-huawei-csi
LAST DEPLOYED: Thu Jun 9 07:58:15 2022
NAMESPACE: huawei-csi
STATUS: deployed
REVISION: 2
TEST SUITE: None
```

----结束

# 9.6.2 手动修改 huawei-csi-controller 服务的日志输出模式

当您需要为huawei-csi-controller服务设置日志输出模式,执行此操作。

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 卸载huawei-csi-controller服务,请参见5.2.2 卸载huawei-csi-controller服务。

步骤3 进入"deploy"目录下,组件包路径参考表3-1。

**步骤4** 执行**vi** *huawei-csi-controller.yaml*命令,修改yaml文件。按**I**或**Insert**进入编辑状态, 修改以下相关参数,详细请参考<mark>表</mark>9-3。修改完成后,按**Esc**,并输入:**wq!**,保存修 改。

#### args:

- "--endpoint=\$(CSI\_ENDPOINT)"
- "--controller"
- "--containerized"
- "--driver-name=csi.huawei.com"
- "--loggingModule=file"
- "--logLevel=info"
- "--logFileDir=/var/log/huawei"
- "--logFileSize=20M"
- "--maxBackups=9"

#### 表 9-3 日志输出参数说明

配置项	描述	备注
loggingModule	huawei-csi日志输出 模式。	当前可以支持"file"和 "console",默认为"file"。
logLevel	huawei-csi日志输出 级别。	支持的级别有:debug,info, warning,error,fatal。默认的级 别为info。
logFileDir	huawei-csi日志在file 输出模式下的日志目 录。	仅当"loggingModule"为"file" 时生效,默认的日志目录为 "/var/log/huawei"。
logFileSize	huawei-csi日志在file 输出模式下单个日志 文件大小。	仅当"loggingModule"为"file" 时生效,默认的日志文件大小为 20MiB。
maxBackups	huawei-csi日志在file 输出模式下日志文件 备份上限。	仅当"loggingModule"为"file" 时生效,默认的日志文件备份数量 为9个。

#### 步骤5 执行以下命令,启动controller服务。

# kubectl create -f huawei-csi-controller-snapshot-v1.yaml

步骤6 完成huawei-csi服务部署后,可执行kubectl get pod -A -o wide | grep huawei命令检查服务是否启动。

# kubectl get pod -A -o wide | grep huawei huawei-csi huawei-csi-controller-b59577886-qqzm8 7/7 Running 0 18h 10.244.1.67 node <none> <none>

#### 步骤7 查看huawei-csi-controller服务的日志。

● 如果设置"loggingModule"为"file",请登录到具体的node节点,然后进入到logFileDir设置的日志目录,执行以下命令,查看huawei-csi-controller日志。

# tail -f huawei-csi-controller

● 如果设置"loggingModule"为"console",请执行以下命令,查看huawei-csi-controller日志

# kubectl logs *huawei-csi-controller* -c huawei-csi-driver -n huawei-csi

#### ----结束

# 9.6.3 手动修改 huawei-csi-node 服务的日志输出模式

当您需要为huawei-csi-node服务设置日志输出模式,执行此操作。

#### 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 卸载huawei-csi-node服务,请参见5.2.1 卸载huawei-csi-node服务。

步骤3 执行vi huawei-csi-node.yaml命令,修改yaml文件。按l或Insert进入编辑状态,修改相关参数。修改完成后,按Esc,并输入:wq!,保存修改。编写huawei-csi-node.yaml文件(可参考软件包中deploy/huawei-csi-node.yaml示例文件)。参数详情可参考表9-4。

#### args:

- "--endpoint=/csi/csi.sock"
- "--containerized"
- "--driver-name=csi.huawei.com"
- "--volume-use-multipath=false"
- "--loggingModule=file"
- "--logLevel=info"
- "--logFileDir=/var/log/huawei"
- "--logFileSize=20M"
- "--maxBackups=9"

#### 表 9-4 日志输出参数说明

配置项	描述	备注
loggingModule	huawei-csi日志输出模 式	当前可以支持"file"和 "console",默认为"file"。
logLevel	huawei-csi日志输出级 别	支持的级别有:debug,info, warning,error,fatal。默认的级 别为info。
logFileDir	huawei-csi日志在file输 出模式下的日志目录	仅当"loggingModule"为 "file"时生效,默认的日志目录 为/var/log/huawei。
logFileSize	huawei-csi日志在file输 出模式下单个日志文件 大小	仅当"loggingModule"为 "file"时生效,默认的日志文件 大小为20MiB。
maxBackups	huawei-csi日志在file输 出模式下日志文件备份 上限	仅当"loggingModule"为 "file"时生效,默认的日志文件 备份数量为9个。

#### 步骤4 执行以下命令,启动node服务。

# kubectl create -f huawei-csi-node.yaml

# 步骤5 完成huawei-csi服务部署后,可执行kubectl get pod -A -o wide | grep huawei-csi-node命令检查服务是否启动。

# kubectl get pod -A | grep huawei-csi-node huawei-csi huawei-csi-node-4sfwr 3/3 Running 0 18h 10.244.1.68 node <none> <none>

#### 步骤6 查看huawei-csi-node服务的日志。

- 如果设置"loggingModule"为"file",请登录到具体的node节点,然后进入到 logFileDir设置的日志目录,执行以下命令,查看huawei-csi-node日志。
  # tail -f huawei-csi-node
- 如果设置 "loggingModule"为 "console",请执行以下命令,查看huawei-csi-node日志
   # kubectl logs huawei-csi-node -c huawei-csi-driver -n huawei-csi

----结束

# 9.7 开启 ReadWriteOncePod 功能门

ReadWriteOnce访问模式是Kubernetes v1.22版本为PV和PVC引入的第四种访问模式。如果您使用ReadWriteOncePod访问模式的PVC创建一个Pod,Kubernetes会确保该Pod是整个集群中唯一可以读取或写入该PVC的Pod。

由于ReadWriteOncePod访问模式在当前已发布的Kubernetes v1.22/1.23/1.24版本中是alpha特性,需要先在kube-apiserver、kube-scheduler和kubelet的feature-gates中开启ReadWriteOncePod特性才能使用。

### 操作步骤

步骤1 为kube-apiserver启用ReadWriteOncePod功能门。

- 1. 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- 2. 执行**vi /etc/kubernetes/manifests/kube-apiserver.yaml**命令,按**l**或**Insert**进入编辑状态,为 kube-apiserver容器添加参数--feature-gates=ReadWriteOncePod=true。修改完成后,按**Esc**,并输入**:wq!**,保存修改。

...
spec:
containers:
- command:
- kube-apiserver
- --feature-gates=ReadWriteOncePod=true

#### □ 说明

在编辑完成后,Kubernetes会自动应用更新,不需要手动更新。

#### 步骤2 为kube-scheduler启用ReadWriteOncePod功能门。

- 1. 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。
- 2. 执行vi /etc/kubernetes/manifests/kube-scheduler.yaml命令,按l或Insert进入编辑状态,为kube-scheduler容器添加参数--feature-

gates=ReadWriteOncePod=true。修改完成后,按**Esc**,并输入**:wq!**,保存修 改。

spec:

containers:

- command:
- kube-scheduler
- -- feature-gates=ReadWriteOncePod=true
- □□ 说明

在编辑完成后,Kubernetes会自动应用更新,不需要手动更新。

步骤3 为kubelet启用ReadWriteOncePod功能门。

#### 须知

由于动态Kubelet配置功能在v1.22中已弃用,并且在v1.24中删除,因此集群中每个worker节点上的kubelet都需要执行以下操作。

- 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意worker节点。
- 2. 执行**vi /var/lib/kubelet/config.yaml**命令,按**l**或**Insert**进入编辑状态,为 KubeletConfiguration对象的feature-gates字段添加ReadWriteOncePod: true, 如果没有feature-gates字段请一并添加。修改完成后,按**Esc**,并输入**:wq!**,保存 修改。

apiVersion: kubelet.config.k8s.io/v1beta1 featureGates:

ReadWriteOncePod: true

□□说明

kubelet配置文件的默认路径为/var/lib/kubelet/config.yaml,请根据实际情况填写。

- 3. 在配置完成后,执行systemctl restart kubelet命令重启kubelet。
- ----结束

# 9.8 配置非 root 用户访问 Kubernetes 集群

#### 操作步骤

**步骤1** 拷贝Kubernetes集群的认证文件,/etc/kubernetes/admin.conf修改为实际使用的认证文件。

\$ mkdir -p \$HOME/.kube

\$ sudo cp -i /etc/kubernetes/admin.conf \$HOME/.kube/config

步骤2 修改认证文件的用户与用户组。

\$ sudo chown \$(id -u):\$(id -g) \$HOME/.kube/config

步骤3 配置当前用户的KUBECONFIG环境变量,以Ubuntu 20.04举例如下。

----结束

# **10** FAQ

- 10.1 如何查看华为CSI日志
- 10.2 Kubernetes平台第一次搭建时, iscsi\_tcp服务没有正常启动,导致创建Pod失败
- 10.3 启动huawei-csi-node失败,提示错误为: "/var/lib/iscsi is not a directory"
- 10.4 集群中worker节点宕机并恢复后,Pod完成failover,但是Pod所在源主机出现盘符残留
- 10.5 启动huawei-csi服务时,服务启动异常, 状态显示InvalidImageName
- 10.6 创建PVC时, PVC的状态为Pending
- 10.7 删除PVC前,PVC的状态为Pending
- 10.8 创建Pod时,Pod的状态为ContainerCreating
- 10.9 创建Pod时,Pod的状态长时间处于ContainerCreating状态
- 10.10 创建Pod失败,日志显示执行mount命令超时
- 10.11 创建Pod失败,日志显示执行mount命令失败
- 10.12 如何下载容器镜像到本地
- 10.13 如何获取CSI版本信息
- 10.14 对接Tanzu Kubernetes集群常见问题及解决方法
- 10.15 使用Tanzu Kubernetes集群时常见问题及解决方法
- 10.16 通用临时卷扩容失败
- 10.17 PVC扩容的目标容量超过存储池容量导致扩容失败

# 10.1 如何查看华为 CSI 日志

## 查看配置 secret 时的日志

步骤1 执行cd /var/log/huawei命令,进入日志目录

# cd /var/log/huawei

#### 步骤2 执行以下命令,查看huawei-csi-install日志

# vi huawei-csi-install

#### ----结束

## 查看 huawei-csi-controller 服务的日志

步骤1 执行以下命令,获取huawei-csi-controller所在的节点

# kubectl get pod -A -o wide | grep huawei huawei-csi huawei-csi-controller-695b84b4d8-tg64l 7/7 **Running** 0 14s <host1-ip> <host1-name> <none>

步骤2 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的huawei-csi-controller节点

步骤3 执行cd /var/log/huawei命令,进入日志目录

# cd /var/log/huawei

步骤4 执行以下命令,查看容器自定义输出日志

# vi huawei-csi-controller

步骤5 执行cd /var/log/containers命令, 进入到容器目录

# cd /var/log/containers

步骤6 执行以下命令, 查看容器标准输出日志

# vi huawei-csi-controller-<name>\_huawei-csi\_huawei-csi-driver-<contrainer-id>.log

----结束

# 查看 huawei-csi-node 服务的日志

步骤1 执行以下命令,获取huawei-csi-node所在的节点

# kubectl get pod -A -o wide | grep huawei huawei-csi huawei-csi-node-g6f7z 3/3 **Running** 0 14s <host2-ip> <host2name> <none>

步骤2 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的huawei-csi-node节点

步骤3 执行cd /var/log/huawei命令,进入日志目录

# cd /var/log/huawei

步骤4 执行以下命令,查看容器自定义输出日志

# vi huawei-csi-node

步骤5 执行cd /var/log/containers命令, 进入到容器目录

# cd /var/log/containers

步骤6 执行以下命令, 查看容器标准输出日志

# vi huawei-csi-node-<name>\_huawei-csi\_huawei-csi-driver-<contrainer-id>.log

----结束

# 10.2 Kubernetes 平台第一次搭建时, iscsi\_tcp 服务没有正常启动,导致创建 Pod 失败

## 现象描述

创建Pod时报错,在/var/log/huawei-csi-node日志中报错" Cannot connect ISCSI portal \*.\*.\*.\*: libkmod: kmod\_module\_insert\_module: could not find module by name='iscsi tcp'。

## 根因分析

搭建Kubernete和安装iSCSI服务后, iscsi\_tcp服务可能会被停掉,可通过执行**lsmod | grep iscsi | grep iscsi\_tcp**查看服务是否被停掉。

# Ismod | grep iscsi | grep iscsi\_tcp
iscsi\_tcp 18333 6
libiscsi\_tcp 25146 1 iscsi\_tcp
libiscsi 57233 2 libiscsi\_tcp,iscsi\_tcp
scsi\_transport\_iscsi 99909 3 iscsi\_tcp,libiscsi

#### 解决措施或规避方法

执行以下命令,手动加载iscsi\_tcp服务。

# 10.3 启动 huawei-csi-node 失败,提示错误为: "/var/lib/iscsi is not a directory"

# 现象描述

启动huawei-csi-node时,无法启动huawei-csi-node服务, 使用**kubectl describe daemonset huawei-csi-node -n huawei-csi**命令查看,提示错误为: "/var/lib/iscsi is not a directory"

# 根因分析

huawei-csi-node中容器内部无/var/lib/iscsi目录

## 解决措施或规避方法

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行以下命令,删除huawei-csi-node服务(huawei-csi-node.yaml为步骤6中的配置文件)。

# kubectl delete -f huawei-csi-node.yaml

**步骤3** 执行以下命令,查看huawei-csi-node服务,如果没有回显,表示删除完成。 # kubectl get pod -A | grep huawei-csi-node

步骤4 执行vi huawei-csi-node.yaml命令,修改yaml文件。按l或Insert进入编辑状态,将huawei-csi-node.yaml -> volumes -> iscsi-dir -> hostPath中"path"设置为"/var/lib/iscsi",删除"type"行。修改完成后,按Esc,并输入:wq!,保存修改。编写huawei-csi-node.yaml文件(可参考软件包中deploy/huawei-csi-node.yaml示例文件)。

步骤5 执行以下命令,启动node服务。

# kubectl create -f huawei-csi-node.yaml

步骤6 完成huawei-csi服务部署后,可执行kubectl get pod -A | grep huawei-csi-node命 令检查服务是否启动。

# kubectl get pod -A | grep huawei-csi-node huawei-csi huawei-csi-node-g6f7z 3/3 Running 0 14s

----结束

# 10.4 集群中 worker 节点宕机并恢复后,Pod 完成failover,但是 Pod 所在源主机出现盘符残留

#### 现象描述

worker节点 A上运行Pod, 并通过CSI挂载外置块设备到该Pod; 异常掉电节点worker节点A; Kubernetes平台会在感知到节点故障后,将Pod切换至worker节点B; 恢复worker节点A, 节点A上的盘符会从正常变为故障。

#### 环境配置

Kubernetes版本: 1.18及以上

存储类型: 块存储

#### 根因分析

worker节点A恢复后,Kubernetes会向存储发起解除映射操作,但是不会发起主机侧的移除盘符操作。在Kubernetes解除映射后,worker节点A上就会出现盘符残留。

#### 解决措施或规避方法

目前的解决方法只能人工介入,手动清理掉主机的残留盘符(或者再次重启主机,利用主机重启过程中扫盘机制,清理掉残留盘符)。具体方法如下:

#### 步骤1 排查主机的残留盘符。

1. 执行multipath -ll命令,判断是否存在多路径状态异常的DM多路径设备:

如下图:路径状态为failed faulty running表示异常,对应的DM多路径设备为dm-12,关联的SCSI磁盘为sdi和sdj,在配置多条路径时,会有多个SCSI磁盘。记录这些SCSI磁盘。

# multipath -ll mpathb (3618cf24100f8f457014a764c000001f6) dm-12 HUAWEI ,XSG1 size=100G features='0' hwhandler='0' wp=rw `-+- policy='service-time 0' prio=-1 status=active

- 是 => 步骤1.2。
- **否** => 不涉及。
- 2. 判断残留的DM多路径设备是否可读。

执行dd if=/dev/dm-xx of=/dev/null count=1 bs=1M iflag=direct命令 (dm-xx为步骤1.1查到的设备号)

如果返回结果为: Input/output error,且读取数据为"0 bytes (0 B) copied",表示该设备不可读。

#dd if=/dev/dm-12 of=/dev/null count=1 bs=1M iflag=direct dd: error reading '/dev/dm-12': Input/output error 0+0 records in 0+0 records out

- 是 => 记录残留的dm-xx设备以及关联磁盘号(见<mark>步骤1.1</mark>),执行清理步
- 命令卡死 => 步骤1.3
- **其他** => 联系技术支持。

0 bytes (0 B) copied, 0.0236862 s, 0.0 kB/s

- 3. 在另一窗口再次登录该节点。
  - a. 执行以下命令, 查看卡死的进程。

# ps -ef | grep dm-12 | grep -w dd root 21725 9748 0 10:33 pts/10 00:00:00 dd if=/dev/dm-12 of=/dev/null count=1 bs=10M iflag=direct

b. 将该pid杀死。 # kill -9 *pid* 

c. 记录残留的dm-xx设备以及关联磁盘号(见**步骤1.1**),执行清理步骤。

#### 步骤2 清理主机的残留盘符。

1. 根据**步骤1**获取的DM多路径设备,执行**multipath -f /dev/***dm-\**命令,清理残留的多路径聚合设备信息。

# multipath -f /dev/dm-12

如果执行报错,请联系技术支持。

2. 清理残留的SCSI磁盘,根据"排查方法"获取的残留磁盘的盘符,依次执行命令:

echo 1 > /sys/block/xxxx/device/delete

配置多条多路径时,依次根据盘符清除,本次残留路径为sdi/sdj:

# echo 1 > /sys/block/sdi/device/delete # echo 1 > /sys/block/sdj/device/delete

如果执行报错,请联系技术支持。

3. 确认DM多路径设备和SCSI磁盘信息是否已经清理干净。

依次执行"multipath -ll"、"ls -l /sys/block/"、"ls -l /dev/disk/by-id/" 命令,查询的多路径和磁盘信息显示,残留的dm-12和SCSI磁盘sdi/sdj均已消 失,则证明清理完成。

# multipath -ll

#### mpathb (3618cf24100f8f457014a764c000001f6) dm-3 HUAWEI ,XSG1

mpathn (3618cf24100f8f457315a764c000001f6) dm-5 HUAWEI ,XSG1

size=100G features='0' hwhandler='0' wp=rw
`-+- policy='service-time 0' prio=-1 status=active

```
|- 39:0:0:2
               sdc 8:32 active ready running
 - 38:0:0:2
               sdb 8:16 active ready running
# ls -l /sys/block/
total 0
lrwxrwxrwx 1 root root 0 Aug 11 19:56 dm-0 -> ../devices/virtual/block/dm-0
lrwxrwxrwx 1 root root 0 Aug 11 19:56 dm-1 -> ../devices/virtual/block/dm-1
lrwxrwxrwx 1 root root 0 Aug 11 19:56 dm-2 -> ../devices/virtual/block/dm-2
lrwxrwxrwx 1 root root 0 Aug 11 19:56 dm-3 -> ../devices/virtual/block/dm-3
lrwxrwxrwx 1 root root 0 Aug 11 19:56 sdb -> ../devices/platform/host35/session2/
target35:0:0/35:0:0:1/block/sdb
lrwxrwxrwx 1 root root 0 Aug 11 19:56 sdc -> ../devices/platform/host34/
target34:65535:5692/34:65535:5692:0/block/sdc
lrwxrwxrwx 1 root root 0 Aug 11 19:56 sdd -> ../devices/platform/host39/session6/
target39:0:0/39:0:0:1/block/sdd
lrwxrwxrwx 1 root root 0 Aug 11 19:56 sde -> ../devices/platform/host38/session5/
target38:0:0/38:0:0:1/block/sde
lrwxrwxrwx 1 root root 0 Aug 11 19:56 sdh -> ../devices/platform/host39/session6/
target39:0:0/39:0:0:3/block/sdh
lrwxrwxrwx 1 root root 0 Aug 11 19:56 sdi -> ../devices/platform/host38/session5/target38:0:0/38:0:0:3/
block/sdi
ls -l /dev/disk/by-id/
total 0
lrwxrwxrwx 1 root root 10 Aug 11 19:57 dm-name-mpathb -> ../../dm-3
lrwxrwxrwx 1 root root 10 Aug 11 19:58 dm-name-mpathn -> ../../dm-5
lrwxrwxrwx 1 root root 10 Aug 11 19:57 dm-uuid-mpath-3618cf24100f8f457014a764c000001f6 -> ../../
lrwxrwxrwx 1 root root 10 Aug 11 19:58 dm-uuid-mpath-3618cf24100f8f457315a764c000001f6 -> ../../
dm-5
lrwxrwxrwx 1 root root 9 Aug 11 19:57 scsi-3618cf24100f8f457014a764c000001f6 -> ../../sdd
lrwxrwxrwx 1 root root 9 Aug 11 19:57 scsi-3618cf24100f8f45712345678000103e8 -> ../../sdi
lrwxrwxrwx 1 root root 9 Aug 3 15:17 scsi-3648435a10058805278654321ffffffff -> ../../sdb
lrwxrwxrwx 1 root root 9 Aug 2 14:49 scsi-36888603000020aff44cc0d060c987f1 -> ../../sdc
lrwxrwxrwx 1 root root 9 Aug 11 19:57 wwn-0x618cf24100f8f457014a764c000001f6 -> ../../sdd
lrwxrwxrwx 1 root root 9 Aug 11 19:57 wwn-0x618cf24100f8f45712345678000103e8 -> ../../sdi
lrwxrwxrwx 1 root root 9 Aug 3 15:17 wwn-0x648435a10058805278654321ffffffff -> ../../sdb
lrwxrwxrwx 1 root root 9 Aug 2 14:49 wwn-0x68886030000020aff44cc0d060c987f1 -> ../../sdc
```

#### ----结束

# 10.5 启动 huawei-csi 服务时,服务启动异常, 状态显示 InvalidImageName

## 现象描述

启动huawei-csi时,无法启动huawei-csi服务(huawei-csi-controller服务或者 huawei-csi-node服务),使用kubectl get pod -A | grep huawei命令查看,显示状态 为InvalidImageName

```
# kubectl get pod -A | grep huawei
huawei-csi huawei-csi-controller-fd5f97768-qlldc 6/7 InvalidImageName 0 16s
huawei-csi huawei-csi-node-25txd 2/3 InvalidImageName 0 15s
```

# 根因分析

controller和node的yaml配置文件中,配置Huawei CSI的镜像版本号错误。例如:

```
...
- name: huawei-csi-driver
image: huawei-csi:3.2.0
...
```

## 解决措施或规避方法

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

**步骤2** 执行以下命令,修改huawei-csi-node服务的配置文件。按l或l**nsert**进入编辑状态,修改相关参数。修改完成后,按**Esc**,并输入:**wg!**,保存修改。

# kubectl edit daemonset huawei-csi-node -o yaml -n=huawei-csi

#### □ 说明

示例yaml文件中huawei-csi-driver的参数image配置项,修改huawei-csi:\*.\*.\*应替换为前面制作的华为CSI镜像

containers:

- name: huawei-csi-driver image: huawei-csi:3.2.0

**步骤3** 执行以下命令,修改huawei-csi-controller服务的配置文件。按**I**或**Insert**进入编辑状态,修改相关参数。修改完成后,按**Esc**,并输入:**wq!**,保存修改。

# kubectl edit deployment huawei-csi-controller -o yaml -n=huawei-csi

#### □说明

示例yaml文件中huawei-csi-driver的参数image配置项,修改huawei-csi:\*.\*.\*应替换为前面制作的华为CSI镜像<*名称*>:<版本号>。

containers:

 name: huawei-csi-driver image: huawei-csi:3.2.0

步骤4 等待huawei-csi-node和huawei-csi-controller服务启动。

步骤5 执行以下命令,查看huawei csi服务是否启动。

# kubectl get pod -A | grep huawei

huawei-csi huawei-csi-controller-58799449cf-zvhmv 7/7 Running 0 2m29s huawei-csi huawei-csi-node-7fxh6 3/3 Running 0 12m

----结束

# 10.6 创建 PVC 时, PVC 的状态为 Pending

### 现象描述

执行完成PVC的创建操作,一段时间后,PVC的状态仍然处于Pending。

# 根因分析

原因1:由于没有提前创建指定名称的StorageClass,导致Kubernetes在创建PVC时无法找到指定StorageClass名称。

原因2:由于存储池能力和StorageClass能力不匹配,导致huawei-csi选择存储池失败。

原因3:由于存储RESTful接口执行返回具体错误码(例如:50331651),导致huawei-csi在执行创建PVC时失败。

原因4:由于存储在huawei-csi设定的超时时间内没有返回,huawei-csi向Kubernetes返回超时错误。

原因5: 其他原因。

## 解决措施或规避方法

创建PVC时,如果PVC处于Pending状态,需要根据以下不同的原因采取不同的解决措施。

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行以下命令,查看PVC的详细信息。

# kubectl describe pvc mypvc

步骤3 根据PVC详细信息中Events信息,执行相应操作。

● 如果由原因1导致PVC处于Pending状态,执行以下步骤。

Events:			_				
Type	Reason	Age	From	Message			
Warni	ng Provisio	ningFailed Os (x1	5 over 3m24s)	persistentvolume-controller			
storageclass.storage.k8s.jo " <i>mysc</i> " not found							

- a. 删除PVC。
- b. 创建StorageClass,可参考7.1.1.1 配置StorageClass。
- c. 创建新的PVC,可参考**7.1.1.1.2 配置PVC**。
- 如果由原因2导致PVC处于Pending状态,执行以下步骤。

Events:		_	
Type	Reason	Age	
From		J	Message
Norma	al Provisioning	63s (x3 over 64	s) csi.huawei.com_huawei-csi-controller-b59577886-
_	E0E00 4 004	4 = 5 00 0 0 0 = 1 00=	

qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 External provisioner is provisioning volume for claim "default/mypvc"

Warning ProvisioningFailed 63s (x3 over 64s) csi.huawei.com\_huawei-csi-controller-b59577886-

qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 failed to provision volume with StorageClass "mysc": rpc error: code = Internal desc = **failed to select pool**, the capability filter failed, error: failed to select pool, the final filter field: **replication**, parameters map[allocType:thin replication:True size: 1099511627776 volumeType:lun]. please check your storage class

- a. 删除PVC。
- b. 删除StorageClass。
- c. 根据Events信息修改StorageClass.yaml文件。
- d. 创建StorageClass,详细请参考7.1.1.1 配置StorageClass。
- e. 创建新的PVC,详情请参考**7.1.1.1.2 配置PVC**。
- 如果由原因3导致PVC处于Pending状态,请联系华为工程师处理。

Events:			
Type	Reason	Age	
From		J	Message
Norma	l Provisioning	63s (x4 over 68s)	csi.huawei.com_huawei-csi-controller-b59577886-
qqzm8_5	8533e4a-884c-4c	7f-92c3-6e8a7b32751	5 External provisioner is provisioning volume for
claim "d	efault/mynyc"		

Warning ProvisioningFailed 62s (x4 over 68s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 failed to provision volume with StorageClass "mysc": rpc error: code = Internal desc = Create volume map[ALLOCTYPE:1 CAPACITY:20 DESCRIPTION:Created from Kubernetes CSI NAME:pvc-63ebfda5-4cf0-458e-83bd-ecc PARENTID:0]

error: **50331651** 

如果由原因4导致PVC处于Pending状态,执行以下步骤。

Events:
Type Reason Age
From Message
---- -----

Normal Provisioning 63s (x3 over 52s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 External provisioner is provisioning volume for claim "default/mypvc"

Warning ProvisioningFailed 63s (x3 over 52s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 failed to provision volume with StorageClass "mysc": rpc error: code = Internal desc = context deadline exceeded (Client.Timeout exceeded while awaiting headers)

- a. 请先等待10分钟,参考本章节再次检查PVC详细信息
- b. 如果还处于Pending状态,请联系华为工程师处理。
- 如果由原因5导致PVC处于Pending状态,请联系华为工程师处理。

#### ----结束

# 10.7 删除 PVC 前,PVC 的状态为 Pending

#### 现象描述

在执行删除PVC前,PVC的状态处于Pending。

## 根因分析

原因1:由于没有提前创建指定名称的StorageClass,导致Kubernetes在创建PVC时无法找到指定StorageClass名称。

原因2:由于存储池能力和StorageClass能力不匹配,导致huawei-csi选择存储池失败。

原因3:由于存储RESTful接口执行返回具体错误码(例如:50331651),导致huawei-csi在执行创建PVC时失败。

原因4:由于存储在huawei-csi设定的超时时间内没有返回,huawei-csi向Kubernetes返回超时错误。

原因5: 其他原因。

# 解决措施或规避方法

删除Pending状态下的PVC,需要根据以下不同的原因采取不同的解决措施。

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意master节点。

步骤2 执行以下命令,查看PVC的详细信息。

# kubectl describe pvc mypvc

步骤3 根据PVC详细信息中Events信息,执行相应操作。

如果由原因1导致PVC处于Pending状态,可以执行kubectl delete pvc mypvc命令,删除PVC。

Events:	3,5,5,1, 1, 0,0			
Type	Reason	Age	From	Message

Warning ProvisioningFailed 0s (x15 over 3m24s) persistentvolume-controller storageclass.storage.k8s.io "mysc" not found

● 如果由原因2导致PVC处于Pending状态,可以执行**kubectl delete pvc** *mypvc*命令,删除PVC。

Events:
Type Reason Age
From Message
---- ----Normal Provisioning 63s (x3 over 64s) csi huawei com huawei-csi-controller-h59577886

Normal Provisioning 63s (x3 over 64s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 External provisioner is provisioning volume for claim "default/mypvc"

Warning ProvisioningFailed 63s (x3 over 64s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 failed to provision volume with StorageClass "mysc": rpc error: code = Internal desc = **failed to select pool**, the capability filter failed, error: failed to select pool, the final filter field: *replication*, parameters map[allocType:thin replication:True size: 1099511627776 volumeType:lun]. please check your storage class

如果由原因3导致PVC处于Pending状态,可以执行kubectl delete pvc mypvc命令,删除PVC。

Events:
Type Reason Age
From Message
---- ------

Normal Provisioning 63s (x4 over 68s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 External provisioner is provisioning volume for claim "default/mypvc"

Warning ProvisioningFailed 62s (x4 over 68s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 failed to provision volume with StorageClass "mysc": rpc error: code = Internal desc = Create volume map[ALLOCTYPE:1 CAPACITY:20 DESCRIPTION:Created from Kubernetes CSI NAME:pvc-63ebfda5-4cf0-458e-83bd-ecc PARENTID:0] error: 50331651

• 如果由原因4导致PVC处于Pending状态,请联系华为工程师处理。

Events:
Type Reason Age
From Message
---- ---- ----

Normal Provisioning 63s (x3 over 52s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 External provisioner is provisioning volume for claim "default/mypvc"

Warning ProvisioningFailed 63s (x3 over 52s) csi.huawei.com\_huawei-csi-controller-b59577886-qqzm8\_58533e4a-884c-4c7f-92c3-6e8a7b327515 failed to provision volume with StorageClass "mysc": rpc error: code = Internal desc = context deadline exceeded (Client.Timeout exceeded while awaiting headers)

● 如果由原因5导致PVC处于Pending状态,请联系华为工程师处理。

-----结束

# 10.8 创建 Pod 时,Pod 的状态为 ContainerCreating

# 现象描述

执行完成Pod的创建操作,一段时间后,Pod的状态仍然处于ContainerCreating,查看具体日志信息(详情请参考**10.1 如何查看华为CSI日志**),报错"Fibre Channel volume device not found"。

# 根因分析

该问题是因为在主机节点有磁盘残留,导致下次创建Pod时,查找磁盘失败。

#### 解决措施或规避方法

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行以下命令,查看Pod所在节点信息。

# kubectl get pod -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE
READINESS GATES
mypod 0/1 ContainerCreating 0 51s 10.244.1.224 node1 <none> <none>

步骤3 删除Pod。

**步骤4** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的*node1* 节点。*node1*节点为**步骤2**中查询的节点。

步骤5 移除盘符残留,详情请参考解决措施或规避方法。

----结束

# 10.9 创建 Pod 时,Pod 的状态长时间处于 ContainerCreating 状态

#### 现象描述

创建Pod时,Pod长时间处于ContainerCreating状态,此时查看huawei-csi-node的日志信息(详情请参考10.1 如何查看华为CSI日志),huawei-csi-node的日志中无创建Pod的日志记录,执行kubectl get volumeattachment命令后,PV列无该Pod使用的PV名称。在等待较长时间后(超过十分钟),Pod正常创建,Pod状态变为Running状态。

### 根因分析

该问题是因为Kubernetes的kube-controller-manager组件服务异常导致。

#### 解决措施或规避方法

请联系容器平台侧工程师解决。

# 10.10 创建 Pod 失败,日志显示执行 mount 命令超时

#### 现象描述

创建Pod时,Pod一直处于ContainerCreating状态,此时查看huawei-csi-node的日志信息(详情请参考**10.1 如何查看华为CSI日志**),日志显示执行mount命令超时。

# 根因分析

- 1. 该问题可能由于配置的业务IP网络不通,导致mount命令执行超时失败。
- 2. 对于部分操作系统,如Kylin V10 SP2,使用NFSv3从容器内执行mount命令耗时较长,可能导致mount命令超时。

## 解决措施或规避方法

- 1. 请配置可用的业务IP地址。
- 2. 建议使用NFSv4.0或NFSv4.1协议。

# 10.11 创建 Pod 失败,日志显示执行 mount 命令失败

## 现象描述

NAS场景下,创建Pod时,Pod一直处于ContainerCreating状态,此时查看huaweicsi-node的日志信息(详情请参考**10.1 如何查看华为CSI日志**),日志显示执行mount 命令失败。

## 根因分析

该问题可能由于存储侧未开启NFS 4.0/4.1协议,主机在使用NFS v4协议挂载失败后,未进行协商使用NFS v3协议挂载。

### 解决措施或规避方法

- 开启存储侧的NFS 3/4.0/4.1协议, 重新尝试默认挂载。
- 直接指定可用的NFS协议进行挂载,参考7.1.1.1.1 配置StorageClass。

# 10.12 如何下载容器镜像到本地

## 使用 containerd 下载容器镜像

**步骤1** 执行**ctr image pull** *image:tag*命令,下载镜像到本地。其中*image:tag*表示需要拉取的镜像及其标签。

# ctr image pull k8s.gcr.io/sig-storage/livenessprobe:v2.5.0

**步骤2** 执行ctr image export *image*.tar *image:tag* 以下命令,导出镜像到文件。其中 *image:tag*表示需要导出的镜像,*image*.tar表示镜像导出后的文件名称。

# ctr image export livenessprobe.tar k8s.gcr.io/sig-storage/livenessprobe:v2.5.0

----结束

# 使用 Docker 下载容器镜像

**步骤1** 执行**docker pull** *image:tag*命令,下载镜像到本地。其中*image:tag*表示需要拉去的镜像。

# docker pull k8s.gcr.io/sig-storage/livenessprobe:v2.5.0

**步骤2** 执行**docker save** *image:tag* **-o** *image.tar* 以下命令,导出镜像到文件。其中 *image:tag*表示需要导出的镜像,*image.*tar表示镜像导出后的文件名称。

# docker save k8s.gcr.io/sig-storage/livenessprobe:v2.5.0 -o livenessprobe.tar

----结束

## 使用 Podman 下载容器镜像

**步骤1** 执行**podman pull** *image:tag*命令,下载镜像到本地。其中*image:tag*表示需要拉去的镜像。

# podman pull k8s.gcr.io/sig-storage/livenessprobe:v2.5.0

**步骤2** 执行**podman save** *image:tag* **-o** *image.tar* 以下命令,导出镜像到文件。其中 *image:tag*表示需要导出的镜像,*image*.tar表示镜像导出后的文件名称。

# podman save k8s.gcr.io/sig-storage/livenessprobe:v2.5.0 -o livenessprobe.tar

----结束

# 10.13 如何获取 CSI 版本信息

本章节指导用户如何查看CSI版本信息。

## 操作步骤

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行以下命令,查看huawei-csi-node所在节点信息。

# kubectl get pod -A -owide | grep huawei-csi-node READY STATUS RESTARTS NAMESPACE NAME AGE IP NOMINATED NODE READINESS GATES NODE huawei-csi huawei-csi-node-87mss 3/3 Running 0 6m41s 8.44.128.33 node-1 <none> huawei-csi huawei-csi-node-xp8cc 3/3 Running 0 6m41s 8.44.128.32 <none>

**步骤3** 使用远程访问工具(以PuTTY为例),通过节点IP地址,登录任意huawei-csi-node所在节点。

步骤4 执行以下命令,查看CSI版本信息。

# cat /var/lib/kubelet/plugins/csi.huawei.com/version 3.2.0

-----结束

# 10.14 对接 Tanzu Kubernetes 集群常见问题及解决方法

本章节用于说明对接Tanzu Kubernetes集群时常见问题及解决办法,目前对接Tanzu Kubernetes集群时主要有以下三个问题:

- 未创建PSP权限导致Pod无法创建
- 主机挂载点与原生Kubernetes不同导致挂载卷失败
- livenessprobe容器端口与Tanzu vSphere端口冲突导致容器不断重启

# 10.14.1 未创建 PSP 权限导致 Pod 无法创建

## 现象描述

创建huawei-csi-controller和huawei-csi-node时,仅Deployment和DaemonSet资源创建成功,controller和node的Pod未创建。

## 根因分析

创建资源使用的service account没有PSP策略的"use"权限。

## 解决措施或规避方法

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

**步骤2** 执行**vi** *psp-use.yaml*命令,创建psp-use.yaml文件。

# vi psp-use.yaml

#### 步骤3 配置psp-use.yaml文件。

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRole

metadata:

name: huawei-csi-psp-role

rules:

- apiGroups: ['policy']

resources: ['podsecuritypolicies']

verbs: ['use']

---

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRoleBinding

metadata:

name: huawei-csi-psp-role-cfg

roleRef:

kind: ClusterRole

name: huawei-csi-psp-role

apiGroup: rbac.authorization.k8s.io

subjects:

- kind: Group

 $api Group: \ \dot{r} bac. authorization. k8s. io$ 

name: system:serviceaccounts:huawei-csi

- kind: Group

apiGroup: rbac.authorization.k8s.io name: system:serviceaccounts:default

#### **步骤4** 执行kubectl create -f psp-use.yaml命令,创建PSP权限。

# kubectl create -f psp-use.yaml

----结束

# 10.14.2 修改主机挂载点

# 现象描述

创建Pod时失败,华为CSI日志中报错"mount point does not exist"。

## 根因分析

huawei-csi-node中的"pods-dir"目录原生Kubernetes集群与Tanzu Kubernetes集群不一致。

#### 解决措施或规避方法

步骤1 进入helm/esdk/templates目录,执行vi huawei-csi-node.yaml命令打开node配置文件。

# vi huawei-csi-node.yaml

步骤2 将"huawei-csi-node.yaml"文件中部分"/var/lib/kubelet/"目录替换为"/var/vcap/data/kubelet/",共计3处需要修改。

#### 第1处:

```
// 替换前
- hostPath:
    path: /var/lib/kubelet/plugins_registry
    type: Directory
    name: registration-dir
// 替换后
- hostPath:
    path: /var/vcap/data/kubelet/plugins_registry
    type: Directory
    name: registration-dir
```

#### 第2处:

```
// 替换前
- hostPath:
    path: /var/lib/kubelet
    type: Directory
    name: pods-dir

// 替换后
- hostPath:
    path: /var/vcap/data/kubelet
    type: Directory
    name: pods-dir
```

#### 第3处:

```
// 替换前
- mountPath: /var/lib/kubelet
mountPropagation: Bidirectional
name: pods-dir
// 替换后
- mountPath: /var/vcap/data/kubelet
mountPropagation: Bidirectional
name: pods-dir
```

#### ----结束

# 10.14.3 修改 livenessprobe 容器的默认端口

## 现象描述

huawei-csi-controller组件中livenessprobe容器一直重启。

# 根因分析

huawei-csi-controller的livenessprobe容器的默认端口(9808)与已有的Tanzu的vSphere CSI端口冲突。

# 解决措施或规避方法

将livenessprobe容器的默认端口修改为未占用端口。

步骤1 进入"helm/esdk/templates"目录,执行vi huawei-csi-controller.yaml命令打开 controller配置文件。

# vi huawei-csi-controller.yaml

步骤2 将livenessprobe容器的默认端口9808修改为其他未占用端口。

----结束

# 10.15 使用 Tanzu Kubernetes 集群时常见问题及解决方法

本章节用于说明在使用Tanzu Kubernetes集群时常见问题及解决办法

# 10.15.1 创建临时卷失败

## 现象描述

创建**通用临时卷**失败,报错PodSecurityPolicy: unable to admit pod: [spec.volumes[0]: Invalid value: "ephemeral": ephemeral volumes are not allowed to be used spec.volumes[0]

# 根因分析

当前使用的PSP策略中没有使用"ephemeral"卷的权限。

# 解决措施或规避方法

在默认PSP "pks-privileged"和"pks-restricted"中增加使用"ephemeral"卷的权限,以修改"pks-privileged"举例:

**步骤1** 使用远程访问工具(以PuTTY为例),通过管理IP地址,登录Kubernetes集群的任意 master节点。

步骤2 执行kubectl edit psp pks-privileged命令,修改pks-privileged的配置。

# kubectl edit psp pks-privileged

#### 步骤3 在spec.volumes中增加 "ephemeral",示例如下

```
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
 annotations:
  apparmor.security.beta.kubernetes.io/allowedProfileName: '*'
  seccomp.security.alpha.kubernetes.io/allowedProfileNames: '*'
 creationTimestamp: "2022-10-11T08:07:00Z"
 name: pks-privileged
 resourceVersion: "1227763"
 uid: 2f39c44a-2ce7-49fd-87ca-2c5dc3bfc0c6
 allowPrivilegeEscalation: true
 allowedCapabilities:
 supplementalGroups:
  rule: RunAsAny
 volumes:
 - glusterfs
 - hostPath
 - iscsi
 - persistentVolumeClaim
```

## 步骤4 执行kubectl get psp pks-privileged -o yaml命令,确认是否添加成功。

# kubectl get psp pks-privileged -o yaml

----结束

# 10.16 通用临时卷扩容失败

## 现象描述

在Kubernetes版本低于1.25环境中,对LUN类型的**通用临时卷**扩容失败,显示PV已经扩容,但PVC未成功更新容量。

## 根因分析

该问题是由Kubernetes的bug导致,Kubernetes在1.25版本中修复了该问题。

# 10.17 PVC 扩容的目标容量超过存储池容量导致扩容失败

### 现象描述

在低于1.23版本的Kubernetes环境中,对PVC进行扩容,当目标容量超过存储池容量时,扩容失败。

## 根因分析

社区已知问题,详情请参考处理扩充卷过程中的失败。

# 解决措施或规避方法

参考处理扩充卷过程中的失败。

11 1 附录

- 11.1 OceanStor V3/V5系列和OceanStor Dorado V3系列ALUA特性配置策略样例
- 11.2 OceanStor Dorado 6.x ALUA特性配置策略样例
- 11.3 分布式存储ALUA特性配置策略样例
- 11.4 安装Helm 3
- 11.5 制作CSI镜像

# 11.1 OceanStor V3/V5 系列和 OceanStor Dorado V3 系列ALUA 特性配置策略样例

#### 例1.配置文件如下:

对于主机名为"node1",上述ALUA配置段都能用于配置启动器。根据**8.1.2.1 配置华 为企业存储后端的ALUA参数**中的配置策略规则,优先级顺序为第2条配置段 (HostName为"node1")高于第1条配置段(HostName为"\*")。

#### 例2.配置文件如下:

对于主机名为"node6"的主机,上述ALUA配置段都能用于配置启动器。根据**8.1.2.1 配置华为企业存储后端的ALUA参数**中的配置策略规则,选择第一条ALUA配置段来配置启动器。

#### 例3.配置文件如下:

根据**8.1.2.1 配置华为企业存储后端的ALUA参数**中的配置策略规则,对于主机名为"node1"的主机,选择第一条ALUA配置段来配置启动器;对于主机名为"node10"的主机,选择第二条ALUA配置段来配置启动器。^表示匹配字符串的开头,\$表示匹配字符串的结尾。

# 11.2 OceanStor Dorado 6.x ALUA 特性配置策略样例

#### 例1.配置文件如下:

```
...
"parameters": {..., "ALUA": {
    "*": {"accessMode": "1", "hyperMetroPathOptimized": "1"},
    "node1": {"accessMode": "1", "hyperMetroPathOptimized": "0"}}}
...
```

对于主机名为"node1",上述ALUA配置段都能用于配置启动器。根据**8.1.2.1 配置华 为企业存储后端的ALUA参数**中的配置策略规则,优先级顺序为第2条配置段 (HostName为"node1")高于第1条配置段(HostName为"\*")。

#### 例2.配置文件如下:

对于主机名为"node6"的主机,上述ALUA配置段都能用于配置启动器。根据**8.1.2.1 配置华为企业存储后端的ALUA参数**中的配置策略规则,选择第一条ALUA配置段来配置启动器。

#### 例3.配置文件如下:

根据**8.1.2.1 配置华为企业存储后端的ALUA参数**中的配置策略规则,对于主机名为"node1"的主机,选择第一条ALUA配置段来配置启动器;对于主机名为"node10"的主机,选择第二条ALUA配置段来配置启动器。^表示匹配字符串的开头,\$表示匹配字符串的结尾。

# 11.3 分布式存储 ALUA 特性配置策略样例

#### 例1.配置文件如下:

对于主机名为"node1",上述ALUA配置段都能用于配置启动器。根据**8.1.2.2 配置分布式存储后端的ALUA参数**中的配置策略规则,优先级顺序为第2条配置段 (HostName为"node1")高于第1条配置段(HostName为"\*")。

#### 例2.配置文件如下:

对于主机名为"node6"的主机,上述ALUA配置段都能用于配置启动器。根据**8.1.2.2 配置分布式存储后端的ALUA参数**中的配置策略规则,选择第一条ALUA配置段来配置启动器。

#### 例3.配置文件如下:

根据8.1.2.2 配置分布式存储后端的ALUA参数中的配置策略规则,对于主机名为 "node1"的主机,选择第一条ALUA配置段来配置启动器;对于主机名为"node10"的主机,选择第二条ALUA配置段来配置启动器。^表示匹配字符串的开头,\$表示匹配字符串的结尾。

# 11.4 安装 Helm 3

本章节指导用户如何安装Helm 3。

参考: https://helm.sh/docs/intro/install/

## 前提条件

确保Kubernetes集群中的master节点可以访问Internet。

#### 操作步骤

步骤1 执行以下命令,下载Helm 3的安装脚本。

# curl -fsSL -o get helm.sh https://raw.githubusercontent.com/helm/helm/master/scripts/get-helm-3

步骤2 执行以下命令,修改Helm 3的安装脚本权限。

# chmod 700 get helm.sh

步骤3 根据Helm与Kubernetes版本配套关系确认需要安装的Helm版本,配套关系请参考 Helm Version Support Policy,执行以下命令,修改DESIRED\_VERSION环境变量为 需要安装的Helm版本,并执行安装命令。

# DESIRED\_VERSION=v3.9.0 ./get\_helm.sh

**步骤4** 执行以下命令,查看指定版本的Helm 3是否安装成功。

# helm version version.BuildInfo{Version:"v3.9.0", GitCommit:"7ceeda6c585217a19a1131663d8cd1f7d641b2a7", GitTreeState:"clean", GoVersion:"go1.17.5"}

----结束

# 11.5 制作 CSI 镜像

本章节指导用户如何制作CSI镜像。

## 前提条件

已准备好一台**已安装Docker的Linux主机**,且该主机支持访问互联网(仅用于下载镜像包)。

#### 操作步骤

步骤1 登录该Linux主机。

步骤2 执行mkdir image命令,在该主机上新建一个目录(例如:"image")。

# mkdir image

步骤3 执行cd image命令进入"image"目录。

# cd image

步骤4 拷贝huawei-csi二进制文件到"image"目录下。

步骤5 执行以下命令,创建Dockerfile文件。

# cat <<EOF > ./Dockerfile FROM busybox:stable-glibc

LABEL maintainers="The Huawei CSI Team"
LABEL description="Huawei Storage CSI Driver."

COPY huawei-csi /

ENTRYPOINT ["/huawei-csi"]

EOF

#### 须知

*busybox:stable-glibc*是基础镜像及其对应的TAG,此处仅为示例,请根据实际情况进行替换。

步骤6 执行docker build -f Dockerfile -t huawei-csi:3.2.0 .命令制作镜像。

# docker build -f Dockerfile -t huawei-csi:3.2.0 .

#### 山 说明

3.2.0为对应软件包名的插件版本号,此处仅为示例,请根据实际情况替换。如果环境上已经存在相同的镜像,请使用docker image rm <image-id>。

**步骤7** 执行docker image ls | grep huawei-csi命令,检查镜像是否制作完成。显示如下回显,则表示制作完成。

# docker image ls | grep huawei-csi huawei-csi 3.2.0

c8b5726118ac

About a minute ago 39 MB

步骤8 执行docker save huawei-csi:3.2.0 -o huawei-csi.tar命令,导出镜像。

# docker save huawei-csi:3.2.0 -o huawei-csi.tar

# 🗀 说明

3.2.0为对应软件包名的插件版本号,此处仅为示例,请根据实际情况替换。

#### ----结束