用户指南 V2.2.13

Kubernetes CSI

文档版本 01

发布日期 2021-08-23





版权所有 © 华为技术有限公司 2021。 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默示的声明或保证。

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: https://www.huawei.com

客户服务邮箱: support@huawei.com

客户服务电话: 4008302118

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	用于警示紧急的危险情形,若不避免,将会导致人员死亡或严重 的人身伤害。
▲警告	用于警示潜在的危险情形,若不避免,可能会导致人员死亡或严 重的人身伤害。
▲注意	用于警示潜在的危险情形,若不避免,可能会导致中度或轻微的 人身伤害。
注意	用于传递设备或环境安全警示信息,若不避免,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。 "注意"不涉及人身伤害。
□ 说明	用于突出重要/关键信息、最佳实践和小窍门等。 "说明"不是安全警示信息,不涉及人身、设备及环境伤害。

目录

前言	ii
 1 概述	
2 环境要求	2
3 限制说明	4
4 安装部署	7
4.1 软件包获取	7
4.2 软件包组件	7
4.3 制作华为 CSI 镜像	8
4.4 对接企业存储配置	<u>9</u>
4.4.1 通过 ISCSI 对接企业存储 SAN 配置	g
4.4.2 通过 FC 对接企业存储 SAN 配置	10
4.4.3 通过 NFS 对接企业存储 NAS 配置	11
4.4.4 通过 NVMe over RoCE 对接企业存储 SAN 配置	12
4.5 对接分布存储配置	13
4.5.1 通过 SCSI 对接分布存储 SAN 配置	13
4.5.2 通过 ISCSI 对接分布存储 SAN 配置	15
4.5.3 通过 NFS 对接分布存储 NAS 配置	16
4.6 启动 huawei-csi 服务	17
5 升级操作	20
5.1 卸载原 CSI	20
5.2 安装新 CSI	21
5.3 更新 CSI	21
6 使用说明	23
6.1 创建 StorageClass	23
6.1.1 创建 LUN StorageClass	23
6.1.2 创建文件系统 StorageClass	24
6.2 创建 PersistentVolumeClaim	
6.3 创建 Pod	26
6.4 创建快照	26
6.4.1 安装 Snapshot 依赖组件服务	26
6.4.2 创建 VolumeSnapshotClass	27

6.4.3 创建 volumeSnapshot	20
6.5 扩容 PVC	
6.5.1 安装扩容依赖组件服务	
6.5.2 扩容 PVC	
6.6 从源创建 PVC	
6.6.1 Clone	
6.6.2 Snapshot Restore	
7 高级特性	32
7.1 配置多个后端	32
7.2 指定后端创建卷	33
7.3 指定存储池创建卷	33
7.4 配置 ALUA 特性	33
7.4.1 配置企业存储 ALUA	33
7.4.2 配置 Dorado V6 ALUA	34
7.4.3 配置分布存储 ALUA	35
7.5 企业存储高级特性	36
7.5.1 配置 QoS	36
7.5.2 配置租户	37
7.5.3 配置 SAN 远程复制	38
7.5.4 配置 SAN 双活	39
7.5.5 配置 NAS 远程复制	40
7.5.6 配置 NAS 双活	41
7.6 分布式存储高级特性	42
7.6.1 配置 QoS	42
8 FAQ	43
8.1 查看日志信息	
8.2 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	
8.3 启动 huawei-csi-node 失败,提示错误为: "/var/lib/iscsi is not a directory"	
8.4 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	

■ 概述

本文档主要介绍Kubernetes配套华为企业存储和云存储产品的CSI插件安装部署和使用,达到利用华为存储向Kubernetes提供持久化卷存储能力的目的。

2 环境要求

- Kubernetes环境部署完成并正常运行。
- 华为存储正常运行。
- 需要提前在宿主机上安装扫盘和文件挂载必须的工具(如果由于系统工具欠缺导 致容器和服务无法正常运行,可先参考8.1 查看日志信息进行日志查看,并自行安 装主机侧工具)

表 2-1 Kubernetes 版本和企业存储产品版本要求

Kubernetes版本	企业存储产品版本
1.13/1.14/1.15/1.16/1.17/1.18/	OceanStor Dorado V6 6.0.0/6.0.1/6.1.0/6.1.2
1.19/1.20/1.21	OceanStor Dorado V3 V300R002
	OceanStor F V5/V5 V500R007/V500R007 Kunpeng
	OceanStor F V3/V3 V300R006

表 2-2 Kubernetes 版本和分布式存储产品版本要求

Kubernetes版本	分布式存储产品版本
1.13/1.14/1.15/1.16/1.17/1.18/ 1.19/1.20/1.21	FusionStorage V100R006C30 FusionStorage Block 8.0.0/8.0.1 OceanStor Pacific 8.1.0

表 2-3 Huawei CSI 特性支持说明 (√: 支持, x: 不支持)

特性	1.13	1.14	1.15	1.16	1.17	1.18	1.19	1.20	1.21
Create PVC	V	V	V	V	V	V	V	V	V

特性	1.13	1.14	1.15	1.16	1.17	1.18	1.19	1.20	1.21
Delete PVC	V	V	V	V	V	V	V	V	V
Create POD	$\sqrt{}$	$\sqrt{}$	V	V	V	V	V	V	\checkmark
Delete POD	V	V	V	V	V	V	V	V	V
Offline Resize	х	х	х	V	V	V	V	V	V
Online Resize	х	х	х	V	V	V	V	V	V
Create Snapsh ot	х	х	х	х	V	V	V	V	V
Delete Snapsh ot	х	х	х	х	V	V	V	V	V
Restore	х	х	х	х	V	V	V	V	V
Clone	х	х	х	х	$\sqrt{}$	$\sqrt{}$	$\sqrt{}$	\vee	$\sqrt{}$

3 限制说明

本章节用于说明CSI对接存储时的限制。

表 3-1 限制说明

场景描述	限制描述	限制存储	其他
PVC的访问模式	ReadWriteOnce: SAN/NAS ReadWriteMany: NAS ReadWriteOnly: SAN/NAS 如果SAN需要使用 ReadWriteMany模式,需要由Pod业 务保证数据一致 性。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0 NAS: OceanStor V3/V5, Dorado V6, OceanStor Pacific	
PersistentVolume Claim	每批次最多可批量 创建/删除100个。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0 NAS: OceanStor V3/V5, Dorado V6, OceanStor Pacific	存储限制RESTful 请求并发量为100
VolumeSnapshot	不涉及	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0 NAS: OceanStor V3/V5, Dorado V6	

场景描述	限制描述	限制存储	其他
VolumeSnapshotR estore	快照恢复,从快照 创建一个新的 PVC,而不是快照 回滚到原PVC。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0 NAS: OceanStor V3/V5	
VolumeExpansion	存储资源的扩容,不支持缩容。RWO模式的PVC,不支持扩容。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0 NAS: OceanStor V3/V5, Dorado V6	
VolumeClone	源PVC和目标PVC 的StorageClass需 要相同。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0 NAS: OceanStor V3/V5	
双活	● 双活只能存储正常情况下使用,一旦存储故障, 只能 保证已经下发业务的双活正常,不能下发新业务。 ● 与远程复制不能在一个 StorageClass同时配置。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6 NAS: OceanStor V3/V5	
远程复制	与双活不能在一个 StorageClass同时 配置。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6 NAS: OceanStor V3/V5, Dorado V6	

场景描述	限制描述	限制存储	其他
盘符残留	由于节点故障,容 器应用漂移到其 方点,节点恢复 后,节点上。需要 所有,节点上。需要 对清理盘符。需要 详细请参考8.4 集 群中worker节点 机并恢复后,Pod 完成failover,但 是Pod所在源主机 出现盘符残留。	SAN: OceanStor V3/V5, Dorado V3/Dorado V6, FusionStorage 8.0	条件: iSCSI/FC + Multipath

4 安装部署

- 4.1 软件包获取
- 4.2 软件包组件
- 4.3 制作华为CSI镜像
- 4.4 对接企业存储配置
- 4.5 对接分布存储配置
- 4.6 启动huawei-csi服务

4.1 软件包获取

通过华为自有Kubernetes CSI仓库获取对应的插件包。

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 选择对应的版本包,下载*eSDK_EnterPrise_Storage_Plugin_.*.***.zip,**.* *** 表示发布版本号。

步骤3 解压该压缩包。

步骤4 在解压出的目录中找到对应的包和资料。

----结束

4.2 软件包组件

通过解压*eSDK_EnterPrise_Storage_Plugin_.*.***.zip,获取到CSI安装和使用需要的软件包和示例文件。软件包组件结构如下表所示:

表 4-1 软件包组件描述

组件	组件描述
bin/huawei-csi	实现CSI规范接口的服务组件。

组件	组件描述
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。
yamls	后续部署过程中用到的yaml示例文件集合。

4.3 制作华为 CSI 镜像

Huawei CSI在运行时,是以容器的形态存在。目前Huawei CSI只提供二进制包(bin/huawei-csi),无法直接使用,因此我们需要根据二进制文件制作CSI镜像,用于启动Huawei CSI服务。

步骤1 准备一台Linux主机,要求该主机上安装了docker并且能够访问外网。

步骤2 登录该Linux主机。

步骤3 在该主机上新建一个目录,进入该目录。

mkdir image; cd image

步骤4 拷贝huawei-csi组件到当前目录下。

步骤5 创建Dockerfile文件。

vi Dockerfile

步骤6 在Dockerfile文件中输入以下内容,并保存。

FROM ***:***

ADD ["huawei-csi", "/"]

RUN ["chmod", "+x", "/huawei-csi"]

ENTRYPOINT ["/huawei-csi"]

须知

:* 表示的是基础镜像及其对应的TAG,需要保证基础镜像中包含**glibc**的运行环境,例如: busybox:stable-glibc

步骤7 制作镜像。

docker build -f Dockerfile -t huawei-csi:*.*.* .

□ 说明

..*对应于软件包名的插件版本号。如果环境上已经存在相同的镜像,请使用docker image rm <image-id>。

步骤8 导出镜像。

docker save huawei-csi:*.*.* -o huawei-csi.tar

步骤9 拷贝huawei-csi.tar镜像文件到Kubernetes集群所有worker节点上

scp huawei-csi.tar <user>@<ip>:/<path>

步骤10 导入镜像。

docker load -i huawei-csi.tar

----结束

4.4 对接企业存储配置

本章节描述如何配置huawei-csi插件对接华为企业存储。

4.4.1 通过 ISCSI 对接企业存储 SAN 配置

须知

部署前请保证如下事项:

- 所有worker节点已安装iSCSI客户端。
- 所有worker节点与后端存储管理IP连通。
- 所有worker节点与后端存储业务IP连通。
- 如果是多路径组网,要求所有worker节点上安装multipath多路径软件。

以下步骤仅需要在任一Kubernetes的master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-oceanstor-iscsi.yaml示例文件)。

表 4-2 配置项描述

配置项	参数格 式	描述	备注
name	字符串	必填,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
storage	字符串	必填,存储后端类 型	对接企业存储SAN场景固定填写 "oceanstor-san"
pools	列表	必填,使用的存储 池名称	支持一个或多个存储池, 用逗号分割

配置项	参数格 式	描述	备注
urls	列表	必填,存储后端管 理URL	支持一个或多个管理URL,用逗号分割,当前仅支持IPv4
paramete rs	字典	必填,ISCSI场景的 可变参数	ISCSI场景protocol参数固定填写iscsi, portals参数填写后端存储ISCSI业务 IP,用逗号分割

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.4.2 通过 FC 对接企业存储 SAN 配置

须知

部署前请保证如下事项:

- 所有worker节点与后端存储管理IP连通。
- 所有worker节点通过FC链路与后端存储连通。
- 如果是多路径组网,要求所有worker节点上安装multipath多路径软件。
- 需要安装systool工具。

以下步骤仅需要在任一master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-oceanstor-fc.yaml示例文件)。

表 4-3 配置项描述

配置项	参数格 式	描述	备注
name	字符串	必填,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
storage	字符串	必填,存储后端类 型	对接企业存储SAN场景固定填写 "oceanstor-san"
pools	列表	必填,使用的存储 池名称	支持一个或多个存储池
urls	列表	必填,存储后端管 理URL	支持一个或多个管理URL,当前仅支持 IPv4
paramete rs	字典	必填,FC场景的可 变参数	FC场景protocol参数固定填写fc

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.4.3 通过 NFS 对接企业存储 NAS 配置

须知

部署前请保证如下事项:

- 所有worker节点已安装NFS客户端工具。
- 所有worker节点与后端存储管理IP连通。
- 所有worker节点与后端存储NFS逻辑端口IP连通。

以下步骤仅需要在任一master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-oceanstor-nfs.yaml示例文件)。

表 4-4 配置项描述

配置项	参数格 式	描述	备注
name	字符串	必填,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
storage	字符串	必填,存储后端类 型	对接企业存储NAS场景固定填写 "oceanstor-nas"
pools	列表	必填,使用的存储 池名称	支持一个或多个存储池
urls	列表	必填,存储后端管 理URL	支持一个或多个管理URL,当前仅支持 IPv4
paramete rs	字典	必填,NAS场景的 可变参数	portals: 指定存储的逻辑端口IP或者 DNS Zone, 只支持配置一个

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.4.4 通过 NVMe over RoCE 对接企业存储 SAN 配置

须知

部署前请保证如下事项:

- 所有worker节点已安装nvme客户端。
- 所有worker节点与后端存储管理IP连通。
- 所有worker节点与后端存储业务IP连通。
- 和NVMe相关的存储,主机,多路径等兼容性,请参考https://supportopen.huawei.com/zh/pages/user/compatibility/support-matrix.jsf。

以下步骤仅需要在任一master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-oceanstor-roce.yaml示例文件)。

```
"parameters": {"protocol": "roce", "portals": ["*.*.**", "*.*.*"]}
}
]
}
```

表 4-5 配置项描述

配置项	参数格 式	描述	备注
name	字符串	必填,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
storage	字符串	必填,存储后端类 型	对接企业存储SAN场景固定填写 "oceanstor-san"
pools	列表	必填,使用的存储 池名称	支持一个或多个存储池, 用逗号分割
urls	列表	必填,存储后端管 理URL	支持一个或多个管理URL,用逗号分割,当前仅支持IPv4
paramete rs	字典	必填,RoCE场景的 可变参数	RoCE场景protocol参数固定填写 roce,portals参数填写后端存储NVMe 业务IP,用逗号分割

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.5 对接分布存储配置

本章节描述如何配置huawei-csi插件对接华为云存储。

4.5.1 通过 SCSI 对接分布存储 SAN 配置

须知

部署前请保证如下事项:

- 所有worker节点与后端存储管理IP连通。
- 所有worker节点已安装分布存储VBS客户端。
- 所有worker节点已加入分布存储客户端。

以下步骤仅需要在任一master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-fusionstorage-scsi.yaml示例文件)。

kind: ConfigMap apiVersion: v1 metadata:

name: huawei-csi-configmap

表 4-6 配置项描述

配置项	参数格 式	描述	备注
name	字符串	必填,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
storage	字符串	必填,存储后端类 型	对接FusionStorage SAN场景固定填写 "fusionstorage-san"
pools	列表	必填,使用的存储 池名称	支持一个或多个存储池
urls	列表	必填,存储后端管 理URL	FusionStorage的管理URL, 只支持配置 一个
paramete rs	字典	必填,可变参数	protocol参数固定填写scsi,portals参数填写主机和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,"*.*.*."为FusionStorage上块客户端对应的管理IP; 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

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.5.2 通过 ISCSI 对接分布存储 SAN 配置

须知

部署前请保证如下事项:

- 所有worker节点已安装iscsi客户端。
- 所有worker节点与后端存储管理IP连通。
- 所有worker节点与后端存储业务IP连通。
- 如果是多路径组网,要求所有worker节点上安装multipath多路径软件。
- 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及其以后版本支持ISCSI组网配置

以下步骤仅需要在任一master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-fusionstorage-iscsi.yaml示例文件)。

表 4-7 配置项描述

配置项	参数格式	描述	备注
name	字符串	必填,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
storage	字符串	必填,存储后端类 型	对接FusionStorage SAN场景固定填写 "fusionstorage-san"
pools	列表	必填,使用的存储 池名称	支持一个或多个存储池
urls	列表	必填,存储后端管 理URL	FusionStorage的管理URL

配置项	参数格 式	描述	备注
paramete rs	字典	必填,可变参数	ISCSI场景protocol参数固定填写iscsi, portals参数填写后端存储ISCSI业务 IP,用逗号分割

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.5.3 通过 NFS 对接分布存储 NAS 配置

须知

部署前请保证如下事项:

- 所有worker节点已安装NFS客户端工具。
- 所有worker节点与后端存储管理IP连通。
- 所有worker节点与后端存储NFS逻辑端口IP连通。

以下步骤仅需要在任一master节点上执行。

步骤1 创建huawei-csi-configmap.yaml文件,并写入以下内容(可参考软件包中yamls/huawei-csi-configmap-fusionstorage-nfs.yaml示例文件)。

表 4-8 配置项描述

配置项	参数格式	描述	备注
name	字符串	必填,存储后 端名	自定义,支持大小写字母、数字、中 划线组合
storage	字符串	必填,存储后 端类型	对接FusionStorage NAS场景固定填写 "fusionstorage-nas"

配置项	参数格式	描述	备注
pools	列表	必填,使用的 存储池名称	支持一个或多个存储池
urls	列表	必填,存储后 端管理URL	FusionStorage的管理URL
paramete rs	字典	必填,NAS场景 的可变参数	portals: 指定存储的逻辑端口IP, 只支持配置一个

步骤2 创建huawei-csi-configmap。

kubectl create -f huawei-csi-configmap.yaml

----结束

4.6 启动 huawei-csi 服务

须知

步骤过程中可能涉及到镜像下载,需要Kubernetes集群的worker节点能够访问外网。如果是内网环境,请通过其他方式获取相关镜像包并手动导入到所有worker节点。

以下步骤仅需要在任一master节点上执行。

步骤1 使用加密工具填写存储用户名和密码

● 初次配置

chmod +x secretGenerate

./secretGenerate

Current backend name is: <backend-1-name>

Current backend url is: [<backend-1-url>]

Enter backend <backend-1-name>'s user: #请填写存储1的用户名
Enter backend <backend-1-name>'s password: #请填写存储1的密码
Please Enter the password again: #再次填写存储1的密码

• 非初次配置

chmod +x secretGenerate

./secretGenerate

The secret info is exist. Do you force to update it? Y/N

Current backend name is: <backend-1-name>

Current backend url is: [<backend-1-url>]

Enter backend <backend-1-name>'s user: #请填写存储1的用户名
Enter backend <backend-1-name>'s password: #请填写存储1的密码
Please Enter the password again: #再次填写存储1的密码

配置多个后端(如需配置多后端,参考7.1 配置多个后端)

chmod +x secretGenerate

./secretGenerate

Current backend name is: <backend-1-name> Current backend url is: [<backend-1-url>]

Enter backend <backend-1-name>'s user: #请填写存储1的用户名
Enter backend <backend-1-name>'s password: #请填写存储1的密码
Please Enter the password again: #再次填写存储1的密码

Current backend name is: <backend-2-name>
Current backend url is: [<backend-2-url>]

Enter backend <backend-2-name>'s user: #请填写存储2的用户名
Enter backend <backend-2-name>'s password: #请填写存储2的密码
Please Enter the password again: #再次填写存储2的密码

完成后,可以使用如下命令检查

kubectl get secret -n kube-system | grep huawei-csi-secret huawei-csi-secret Opaque

步骤2 编写huawei-csi-rbac.yaml文件(可参考软件包中yamls/huawei-csi-rbac.yaml示例文件,resize和snapshot详见 **附:** resize和snapshot功能说明)。

创建RBAC权限。

kubectl create -f huawei-csi-rbac.yaml

步骤3 编写huawei-csi-controller.yaml文件(可参考软件包中yamls/huawei-csi-controller.yaml示例文件,resize和snapshot详见•附: resize和snapshot功能说明)。

启动controller服务。

kubectl create -f huawei-csi-controller.yaml

□ 说明

示例yaml文件中huawei-csi:*.*.*配置项, *.*.*应替换为前面制作的华为CSI镜像版本号;

步骤4 编写huawei-csi-node.yaml文件(可参考软件包中yamls/huawei-csi-node.yaml示例文件)。

启动node服务。

kubectl create -f huawei-csi-node.yaml

□ 说明

- 示例yaml文件中huawei-csi:*.*.*配置项,*.*.*应替换为前面制作的华为CSI镜像版本号。
- 示例yaml文件中huawei-csi-driver配置参数args中: --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"

步骤5 以上步骤完成后,容器化的huawei-csi服务即部署完成。可以使用如下命令检查

kubectl get pod -A | grep huawei kube-system huawei-csi-controller-695b84b4d8-tg64l 3/3 **Running** 0 14s kube-system huawei-csi-node-g6f7z 2/2 **Running** 0 14s

----结束

• 附: resize和snapshot功能说明

□ 说明

csi-resizer服务从Kubernetes v1.16开始支持

- huawei-csi-rbac.yaml文件请参考软件包中yamls/huawei-csi-resize-rbac.yaml示例文件
- huawei-csi-controller.yaml文件请参考软件包中yamls/huawei-csi-resize-controller.yaml示例文件

csi-snapshotter服务从Kubernetes v1.17开始支持

- huawei-csi-rbac.yaml文件请参考软件包中yamls/huawei-csi-resize-snapshot-rbac.yaml示例文件
- huawei-csi-controller.yaml文件请参考软件包中yamls/huawei-csi-resize-snapshot-controller.yaml示例文件

5 升级操作

须知

- CSI升级过程中对已经下发的PVC/Snapshot/Pod等资源没有影响
- 在升级过程中,不能使用CSI下发新的资源
- 5.1 卸载原CSI
- 5.2 安装新CSI
- 5.3 更新CSI

5.1 卸载原 CSI

以下步骤仅需要在任一Kubernetes的master节点上执行

步骤1 删除node服务(huawei-csi-node.yaml为步骤4中的配置信息)。

kubectl delete -f huawei-csi-node.yaml

使用如下命令查看容器化的huawei-csi-node服务,如果没有回显,表示删除完成。

kubectl get pod -A | grep huawei-csi

kube-system huawei-csi-node-g6f7z 2/2 **Running** 0 14s

步骤2 删除controller服务(huawei-csi-controller.yaml为步骤3中配置信息)。

kubectl delete -f huawei-csi-controller.yaml

可以使用如下命令查看容器化的huawei-csi-controller服务,如果没有回显,表示删除 完成

kubectl get pod -A | grep huawei-csi

kube-system huawei-csi-controller-695b84b4d8-tg64l 3/3 Running 0 14

步骤3 删除RBAC权限(huawei-csi-rbac.yaml为步骤2中配置信息)。

kubectl delete -f huawei-csi-rbac.yaml

步骤4 删除configmap对象。

kubectl delete configmap huawei-csi-configmap -n <namespace-name>

使用如下命令查看huawei-csi的configmap信息,如果没有回显,表示删除完成。

kubectl get configmap -A | grep huawei-csi-configmap

kube-system huawei-csi-configmap 1 129m

步骤5 删除secret对象。

kubectl delete secret huawei-csi-secret -n <namespace-name>

使用如下命令查看huawei-csi的secret信息,如果没有回显,表示删除完成。

```
# kubectl get secret -A | grep huawei-csi-secret kube-system huawei-csi-secret Opaque 1 20m
```

----结束

5.2 安装新 CSI

新的CSI安装,请参考4 安装部署

须知

• 如果是升级场景,请保证storage, name, pools参数与原配置文件保持一致

● 如果是升级场景,urls和parameters参数需要手动改动,具体差异请查看4.4 对接企业存储配置或4.5 对接分布存储配置

5.3 更新 CSI

须知

- 更新CSI时,由Huawei CSI创建的资源业务能够正常运行。
- 如果需要CSI在更新后继续管理原有的资源,需要保证huawei-csi-configmap(4.4 对接企业存储配置或4.5 对接分布存储配置)不变。

以下步骤仅需要在任一Kubernetes的master节点上执行

•

更新 configmap 对象

步骤1 查看configmap对象

kubectl get configmap huawei-csi-configmap -n <namespace-name>

步骤2 删除huawei-csi-node和huawei-csi-controller对象

kubectl delete -f huawei-csi-controller.yaml && kubectl delete -f huawei-csi-node.yaml

步骤3 删除configmap对象

kubectl delete configmap huawei-csi-configmap -n <namespace-name>

步骤4 按照4.4 对接企业存储配置或4.5 对接分布存储配置编辑huawei-csi-configmap.yaml

步骤5 创建configmap对象

kubectl create -f huawei-csi-configmap.yaml

步骤6 创建huawei-csi-node和huawei-csi-controller对象

kubectl create -f huawei-csi-controller.yaml && kubectl create -f huawei-csi-node.yaml

----结束

更新 secret 对象

步骤1 执行以下命令,添加secretUpdate(4.2 软件包组件)的可执行权限。

chmod +x secretUpdate

步骤2 运行secretUpdate工具,按照提示信息进行输入。

The 1 backend name is: <backend-1-name> backend url is: [<backend-1-url>]

Do you want to update it? Y/N #请填写是否更新 Enter backend <backend-1-name>'s user: #请填写存储1的用户名

Enter backend <backend-1-name>'s user: #请填与存储1的用户名
Enter backend <backend-1-name>'s password: #请填写存储1的密码
Please Enter the password again: #再次填写存储1的密码

The 2 backend name is: <backend-2-name> backend url is: [<backend-2-url>]

Do you want to update it? Y/N #请填写是否更新

Enter backend <backend-2-name>'s user: #请填写存储2的用户名
Enter backend <backend-2-name>'s password: #请填写存储2的密码
Please Enter the password again: #再次填写存储2的密码

----结束



本章节主要介绍如何配套华为存储提供PersistentVolume给Kubernetes使用。

- 6.1 创建StorageClass
- 6.2 创建PersistentVolumeClaim
- 6.3 创建Pod
- 6.4 创建快照
- 6.5 扩容PVC
- 6.6 从源创建PVC

6.1 创建 StorageClass

6.1.1 创建 LUN StorageClass

以下步骤仅需要在任一master节点上执行。

步骤1 配置StorageClass yaml文件。

kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: "***" provisioner: "csi.huawei.com" parameters: volumeType: "lun" allocType: "thin" cloneFrom: "**" cloneSpeed: "**" fsType: "**"

表 6-1 参数说明

参数	说明	备注
name	自定义的 StorageClass对 象名称	

参数	说明	备注
provisioner	provisioner标识	固定填写"csi.huawei.com"
volumeType	待创建卷类型	固定填写"lun"
allocType	待创建卷的分配 类型	可选,支持thin/thick,默认为thin
cloneFrom	指定克隆原卷	可选,格式为"存储后端名.原卷名"
cloneSpeed	指定克隆速度	可选,默认值3,支持1~4,4速度最快
fsType	指定文件系统类 型	可选,支持ext2/ext3/ext4,默认为ext4

步骤2 执行以下命令,基于该yaml文件创建StorageClass

kubectl create -f /path/to/yaml/file

----结束

6.1.2 创建文件系统 StorageClass

以下步骤仅需要在任一master节点上执行。

步骤1 配置StorageClass yaml文件。

kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: "***" provisioner: "csi.huawei.com" parameters: volumeType: "fs" allocType: "thin" authClient: "*" cloneFrom: "**"

表 6-2 参数说明

cloneSpeed: "**"

参数	说明	备注
name	自定义的StorageClass对象 名称	
provisioner	provisioner标识	固定填写"csi.huawei.com"
volumeTyp e	待创建卷类型	固定填写"fs"
authClient	指定可访问该FS卷的客户 端	必选,支持输入客户端主机名称、客户端IP地址、客户端IP地址段或使用"*"表示全部客户端IP地址 支持指定多个客户端,以";"分号分隔
allocType	待创建卷的分配类型	可选,支持thin/thick,默认为thin

参数	说明	备注
cloneFrom	指定克隆原卷	可选,格式为"存储后端名.原卷名"
cloneSpee d	指定克隆速度	可选,默认值3,支持1~4,4速度最快

步骤2 执行以下命令,基于该yaml文件创建StorageClass。

kubectl create -f /path/to/yaml/file

----结束

6.2 创建 PersistentVolumeClaim

以下步骤仅需要在任一master节点上执行。

步骤1 配置PersistentVolumeClaim yaml文件。

kind: PersistentVolumeClaim apiVersion: v1 metadata: name: "***" spec: accessModes: - ReadWriteMany storageClassName: "***" resources: requests: storage: ***Gi

表 6-3 参数说明

参数	说明	备注
name	自定义的 PersistentVolumeClaim对象 名称	
storageClas sName	StorageClass对象名称	填写 6.1 创建StorageClass 创建的 StorageClass对象名称
storage	指定待创建卷大小	格式为***Gi,单位为GB
accessMode s	指定卷访问模式	lun卷支持ReadWriteOnce, ReadOnlyMany和 ReadWriteMany,如果使用 ReadWriteMany模式且有多个Pod同 时访问该卷,需要由Pod业务保证数 据一致性 fs卷支持ReadWriteOnce, ReadOnlyMany和ReadWriteMany

步骤2 执行以下命令,基于该yaml文件创建PersistentVolumeClaim

kubectl create -f /path/to/yaml/file

----结束

6.3 创建 Pod

以下步骤仅需要在任一master节点上执行。

步骤1 配置Pod yaml文件。

```
kind: Pod
apiVersion: v1
metadata:
name: "***"
spec:
containers:
- name: "***"
image: "***"
volumeMounts:
- name: mypv
mountPath: "***"
volumes:
- name: mypv
persistentVolumeClaim:
claimName: "***"
```

表 6-4 参数说明

参数	说明	备注
metadata:name	自定义的Pod对象名称	
spec:containers:name	自定义的容器名称	
spec:containers:image	指定容器镜像	
spec:containers:image:volumeMo unts:mountPath	容器内持久化卷的挂载 路径	
spec:volumes:persistentVolumeCl aim:claimName	PersistentVolumeClai m对象名称	填写6.2 创建 PersistentVolumeCl aim创建的PVC对象 名称

步骤2 执行以下命令,基于该yaml文件创建Pod.

kubectl create -f /path/to/yaml/file

----结束

6.4 创建快照

从Kubernetes 的v1.17 版本开始,CSI支持快照的v1beta1版本。详情请查看: https://kubernetes-csi.github.io/docs/external-snapshotter.html

6.4.1 安装 Snapshot 依赖组件服务

以下步骤仅需要在任一master节点上执行。

前提条件

安装前需要确认是否已经安装snapshot相关资源服务,在master上使用kubectl apiresources | grep snapshot | awk '{print \$1}', 如果回显如下,则不需要再次安装

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

如果结果不包含上述服务,请按照以下指导进行安装

步骤1 缺省volumesnapshotclasses,编写volumesnapshotclasses.yaml文件(可参考软件包中yamls/snapshot.storage.k8s.io_volumesnapshotclasses.yaml示例文件)

kubectl create -f volumesnapshotclasses.yaml

步骤2 缺省volumesnapshotcontents,编写volumesnapshotcontents.yaml文件(可参考软件包中yamls/snapshot.storage.k8s.io_volumesnapshotcontents.yaml示例文件)

kubectl create -f volumesnapshotcontents.yaml

步骤3 缺省volumesnapshots,编写volumesnapshots.yaml文件(可参考软件包中yamls/snapshot.storage.k8s.io_volumesnapshots.yaml示例文件)

kubectl create -f volumesnapshots.yaml

步骤4 按照4.6 启动huawei-csi服务进行配置

步骤5 snapshot详见•附: resize和snapshot功能说明。如果使用huawei-csi-resize-snapshot-controller.yaml进行部署,容器化的huawei-csi服务显示如下:

kubectl get pod -A | grep huawei kube-system huawei-csi-controller-fd5f97768-qlldc 6/6 Running 0 16s kube-system huawei-csi-node-25txd 2/2 Running 0 15s

----结束

6.4.2 创建 VolumeSnapshotClass

步骤1 配置VolumeSnapshotClass yaml文件(可参考软件包中yamls/snapshotclass.yaml示例文件)。

apiVersion: snapshot.storage.k8s.io/v1beta1 kind: VolumeSnapshotClass metadata: name: *** driver: csi.huawei.com deletionPolicy: Delete

表 6-5 参数说明

参数	说明	备注
name	自定义的 VolumeSnapshotClass对 象名称	
driver	driver标识	固定填写"csi.huawei.com"
deletionPo licy	VolumeSnapshot删除时, 处理 VolumeSnapshotContent 策略	必填, 可选值为Delete或Retain

步骤2 执行以下命令,基于该yaml文件创建VolumeSnapshotClass

kubectl create -f /path/to/yaml/file

----结束

6.4.3 创建 volumeSnapshot

步骤1 配置VolumeSnapshot yaml文件(可参考软件包中yamls/snapshot.yaml示例文件)。

apiVersion: snapshot.storage.k8s.io/v1beta1 kind: VolumeSnapshot metadata: name: *** spec: volumeSnapshotClassName: *** source: persistentVolumeClaimName: ***

表 6-6 参数说明

参数	说明	备注
name	自定义的VolumeSnapshot 对象名称	
volumeSnapsh otClassName	VolumeSnapshotClass对 象名称	填写 6.4.2 创建 VolumeSnapshotClass创建的 VolumeSnapshotClass对象名称
persistentVolu meClaimName	源PVC对象名称	填写 6.2 创建 PersistentVolumeClaim创建的 PersistentVolumeClaim对象名称

步骤2 执行以下命令,基于该yaml文件创建VolumeSnapshot

kubectl create -f /path/to/yaml/file

----结束

6.5 扩容 PVC

6.5.1 安装扩容依赖组件服务

扩容详见**•附: resize和snapshot功能说明**。如果使用huawei-csi-resize-snapshot-controller.yaml进行部署,容器化的huawei-csi服务显示如下:

kubectl get pod -A | grep huawei

kube-system huawei-csi-controller-fd5f97768-qlldc 6/6 Running 0 16s kube-system huawei-csi-node-25txd 2/2 Running 0 15s

6.5.2 扩容 PVC

步骤1 根据5.1章节创建StorageClass配置StorageClass yaml文件,在配置项中添加 allowVolumeExpansion配置项,示例如下:

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

```
metadata:
name: "***"
provisioner: "csi.huawei.com"
parameters:
...
allowVolumeExpansion: true
```

步骤2 根据5.2章节创建PersistentVolumeClaim

步骤3 扩容操作支持在线扩容和离线扩容,操作命令如下

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

----结束

6.6 从源创建 PVC

当创建PVC时,可以通过指定源对象,通过存储功能创建一个新的PVC,目前支持的源有PVC和VolumeSnapshot。详情请查看: https://kubernetes-csi.github.io/docs/volume-datasources.html

6.6.1 Clone

步骤1 配置PersistentVolumeClaim yaml文件。

```
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
name: ***
spec:
storageClassName: ***
dataSource:
name: ***
kind: PersistentVolumeClaim
accessModes:
- ReadWriteMany
resources:
requests:
storage: ***Gi
```

表 6-7 参数说明

参数	说明	备注
metadata/ name	自定义的新建 PersistentVolumeClaim对象 名称	
spec/ storageClas sName	StorageClass对象名称	填写 6.1 创建StorageClass 创建的 StorageClass对象名称,与 dataSource中的对象的StorageClass 需一致
spec/ dataSource/ name	源PersistentVolumeClaim对 象名称	

参数	说明	备注
spec/ resources/ requests/ storage	指定待创建新卷大小	不小于源PersistentVolumeClaim, 格式为***Gi,单位为GB

步骤2 执行以下命令,基于该yaml文件创建PersistentVolumeClaim。

kubectl create -f /path/to/yaml/file

----结束

6.6.2 Snapshot Restore

步骤1 配置PersistentVolumeClaim yaml文件。

kind: PersistentVolumeClaim
apiVersion: v1
metadata:
name: ***
spec:
storageClassName: ***
dataSource:
name: ***
kind: VolumeSnapshot
apiGroup: snapshot.storage.k8s.io
accessModes:
- ReadWriteMany
resources:
requests:
storage: ***Gi

表 6-8 参数说明

参数	说明	备注
metadata/ name	自定义的新建 PersistentVolumeClaim对象 名称	
spec/ storageClas sName	StorageClass对象名称	填写 6.1 创建StorageClass 创建的 StorageClass对象名称,与 dataSource中快照的原PVC的 StorageClass需一致
spec/ dataSource/ name	源VolumeSnapshot对象名称	
spec/ resources/ requests/ storage	指定待创建新卷大小	不小于源VolumeSnapshot,格式为 ***Gi,单位为GB

步骤2 执行以下命令,基于该yaml文件创建PersistentVolumeClaim

kubectl create -f /path/to/yaml/file

----结束

了 高级特性

本章节介绍如何配置使用华为存储的高级特性。

- 7.1 配置多个后端
- 7.2 指定后端创建卷
- 7.3 指定存储池创建卷
- 7.4 配置ALUA特性
- 7.5 企业存储高级特性
- 7.6 分布式存储高级特性

7.1 配置多个后端

Huawei CSI是可以支持配置多个后端的,多个后端间使用","进行分割(最后一个后端不需要","),每个后端内容可以参考**4.4 对接企业存储配置**或**4.5 对接分布存储配置**

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

7.2 指定后端创建卷

针对配置了多后端的场景,可以通过以下方式指定卷在哪个后端上创建。

配置StorageClass yaml文件,在parameters配置项下添加backend配置项,示例如下:

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

backend参数值是huawei-csi-configmap.yaml后端配置中的某个后端名称。

使用该StorageClass创建的卷即会在指定后端存储上创建。

7.3 指定存储池创建卷

针对配置了多存储池的场景,可以通过以下方式指定卷在哪个存储池上创建。

配置StorageClass yaml文件,在parameters配置项下添加pool配置项,示例如下:

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

pool参数值是某个存储池名称。

□ 说明

使用该StorageClass新创建的卷即会在指定存储池上创建,已经存在的PVC不会更改存储池信息。

7.4 配置 ALUA 特性

在对接SAN/Block存储的场景下,如果有使用多路径,可以配置ALUA。

7.4.1 配置企业存储 ALUA

修改huawei-csi-configmap.yaml文件,在后端parameters参数中增加ALUA配置。

```
{
    "backends": [
    {
        "storage": "oceanstor-san",
        ...
        "parameters": {..., "ALUA": {"<HostName>": {"MULTIPATHTYPE": "*", "FAILOVERMODE": "*",
"SPECIALMODETYPE": "*", "PATHTYPE": "*"}, "<HostName>": {...}}}
}
```

] }

表 7-1 ALUA 参数说明

参数名	参数描述	备注
<hostname></hostname>	主机名称的正则表达式,匹 配此正则表达式的worker节 点主机将使用对应的ALUA配 置	如果配置为*号,表示默认ALUA配置。 置。 如果worker节点主机无法匹配到其 他主机名称正则表达式,则使用默 认配置。
MULTIPATHT YPE	多路径类型 0: 默认 1: 第三方多路径	
FAILOVERMO DE	启动器的切换模式 0: 旧版本ALUA 1: 通用ALUA 2: 不使用ALUA 3: 特殊模式	当开启第三方多路径时该参数才需要下发,旧版本ALUA在V5所有型号不支持
SPECIALMOD ETYPE	启动器的特殊模式类型 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时 该参数才需要下发
PATHTYPE	启动器的路径类型 0:优选路径 1:非优选路径	当开启第三方多路径时该参数才需 要下发。

山 说明

- 1. 针对不同的操作系统,ALUA配置可能有所不同。进入https://support.huawei.com/enterprise/zh/index.html,在搜索输入框中输入"主机连通性指南",点击搜索。在搜索结果中,选择对应操作系统的主机连通性指南,根据指南的推荐配置进行ALUA配置。
- 2. 企业存储V3/V5系列,以及Dorado V3系列使用此种配置方式。
- 3. 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。

7.4.2 配置 Dorado V6 ALUA

修改huawei-csi-configmap.yaml文件,在后端parameters参数中增加ALUA配置。

```
{
    "backends": [
    {
      "storage": "oceanstor-san",
      ...
```

```
"parameters": {..., "ALUA": {"<HostName>": {"accessMode": "*", "hyperMetroPathOptimized": "*"}, "<HostName>": {...}}}
}
]
]
```

表 7-2 ALUA 参数说明

参数名	参数描述	备注
<hostname></hostname>	主机名称的正则表达式,匹 配此正则表达式的worker节	如果配置为*号,表示默认ALUA 配置。
	点主机将使用对应的ALUA配 置	如果worker节点主机无法匹配 到其他正则表达式,则使用默认 配置。
accessMode	主机访问模式	
	0:均衡模式	
	1: 非对称模式	
hyperMetroPath Optimized	双活场景下,主机在当前阵 列的路径是否优选	
	1: Yes	
	0: No	

山 说明

- 1. 针对不同的操作系统,ALUA配置可能有所不同。进入https://support.huawei.com/enterprise/zh/index.html,在搜索输入框中输入"主机连通性指南",点击搜索。在搜索结果中,选择对应操作系统的主机连通性指南,根据指南的推荐配置进行ALUA配置。
- 2. Dorado V6系列使用此种配置方式。
- 3. 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。

7.4.3 配置分布存储 ALUA

修改huawei-csi-configmap.yaml文件,在后端parameters参数中增加ALUA配置。

```
{
  "backends": [
  {
    "storage": "fusionstorage-san",
    ...
    "parameters": {..., "ALUA": {"<HostName>": {"switchoverMode": "*", "pathType": "*"},
  "<HostName>": {...}}}
  }
}
```

表 7-3 ALUA 参数说明

参数名	参数描述	备注
<hostname></hostname>	主机名称的正则表达式,匹 配此正则表达式的worker节	如果配置为*号,表示默认ALUA 配置。
	点主机将使用对应的ALUA配置	如果worker节点主机无法匹配 到其他正则表达式,则使用默认 配置。
switchoverMode	切换模式	
	Disable_alua: 禁用ALUA	
	Enable_alua: 启用ALUA	
pathType	路径类型	
	optimal_path: 优选路径	
	non_optimal_path: 非优选 路径	

□ 说明

- 1. 仅支持分布存储ISCSI场景。
- 2. 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。

7.5 企业存储高级特性

7.5.1 配置 QoS

山 说明

- 1. QoS特性不是Kubernetes的标准特性,由存储厂商自定义;
- 2. QoS策略只能在创建PVC时指定;
- 3. 针对已经创建好的PVC,Kubernetes测无法对其进行修改。

本章节介绍如何创建支持QoS的LUN/FS卷。

步骤1 配置StorageClass yaml文件,在parameters配置项下添加qos配置项,示例如下:

kind: StorageClass apiVersion: storage.k8s.io/v1 metadata: name: "***"

provisioner: "csi.huawei.com"

parameters:

qos: '{"IOTYPE": 2, "MINIOPS": 1000}'

qos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)。

步骤2 使用该StorageClass创建新的PVC

----结束

表 7-4 qos 支持的参数说明

参数名	参数描述	备注
IOTYPE	控制读写类型	可选(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)
		有效值如下:
		● 0: 读IO
		● 1: 写IO
		● 2: 读写IO
MAXBANDWIDTH	最大带宽限制策略	单位MB/s,有效值为>0的整数
MINBANDWIDTH	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

🔲 说明

- 1. 同属保护策略或者同属限制策略的多个参数可以同时指定,但是保护策略和限制策略参数不能同时指定。
- 2. Dorado存储仅支持IOTYPE设置为2(即读写IO),仅支持限制策略参数。
- 3. 租户用户不支持配置QoS策略。
- 4. 配置QoS后只能在新建的PVC上生效;对于同名StorageClass已经发放的PVC,不能自动添加QoS

7.5.2 配置租户

修改huawei-csi-configmap.yaml文件,在后端配置中增加vstoreName 配置项。

```
{
    "backends": [
    {
        ...
        "user": "***",
        "password": "***",
        "vstoreName": "***"
    }
    ]
}
```

□ 说明

- user、password和vstoreName对应于存储上事先配置好的租户用户名、租户用户密码以及租户名。
- 配置完成huawei-csi-configmap.yaml后,需要重启huawei-csi-controller和huawei-csi-node,否则不生效。

7.5.3 配置 SAN 远程复制

须知

使用SAN远程复制,需要事先在两台存储之间完成远程复制关系配置。具体请参考华 为存储相关配置指导。

在huawei-csi-configmap.yaml文件的backends配置段中,添加构成复制关系的两个后端,并且为每个后端增加replicaBackend配置项,如下所示:

```
{
    "backends":[
    {
        ...
        "name": "replica1",
        "replicaBackend": "replica2"
    },
    {
        ...
        "name": "replica2",
        "replicaBackend": "replica1"
    }
}
```

山 说明

● replicaBackend是远程复制对端的后端名,两个复制后端相互构成复制关系。如上所示,后端replica1的远程复制对端是replica2,相应的,replica2的远程复制对端是replica1。

在StorageClass yaml文件的parameters配置项下,添加以下配置项。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: "***"
provisioner: "csi.huawei.com"
parameters:
...
volumeType: lun
replication: "true"
replicationSyncPeriod: "3600"
backend: "***"
```

表 7-5 参数说明

参数名	参数描述	备注
replication	指定是否创建远程 复制卷	配置为true,表示需要创建远程复制卷; 不配置或配置为false,表示不创建远程复 制卷。
replicationSync Period	指定远程复制同步 时间间隔	可选,单位为秒,默认为3600s。
backend	指定创建远程复制 卷的主端名	对应于huawei-csi-configmap.yaml中配置的后端名。

使用该StorageClass创建的新卷即为具有SAN远程复制能力的卷。

7.5.4 配置 SAN 双活

须知

使用SAN双活,需要事先在两台存储之间完成双活关系配置。具体请参考华为存储相 关配置指导。

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,并且为每个后端增加hyperMetroDomain配置项,如下所示:

山 说明

- hyperMetroDomain是华为存储之间配置的双活域名。
- metroBackend是双活对端的后端名,两个双活后端相互构成双活关系。如上所示,后端 hyperMetro1的双活对端是hyperMetro2,相应的,hyperMetro2的双活对端是 hyperMetro1。

在StorageClass yaml文件的parameters配置项下,添加以下配置项。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: "***"
provisioner: "csi.huawei.com"
parameters:
...
volumeType: lun
hyperMetro: "true"
```

表 7-6 参数说明

参数名	参数描述	备注
hyperMetro	指定是否创建双活 卷	配置为true,表示需要创建双活卷;不配置 或配置为false,表示不创建双活卷。

使用该StorageClass创建的卷即为具有SAN双活能力的卷。

7.5.5 配置 NAS 远程复制

须知

使用NAS远程复制,需要事先在两台存储之间完成远程复制关系配置。具体请参考华 为存储相关配置指导。

在huawei-csi-configmap.yaml文件的backends配置段中,添加构成复制关系的两个后端,并且为每个后端增加replicaBackend配置项,如下所示:

```
"backends":[
{
    "name": "replica1",
    "replicaBackend": "replica2"
},
{
    ...
    "name": "replica2",
    "replicaBackend": "replica1"
}
]
```

山 说明

- replicaBackend是远程复制对端的后端名,两个复制后端相互构成复制关系。如上所示,后端replica1的远程复制对端是replica2,相应的,replica2的远程复制对端是replica1。
- 支持租户,可参照5.3.2章节配置远程复制后端使用租户。

在StorageClass yaml文件的parameters配置项下,添加以下配置项。

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
name: "***"
provisioner: "csi.huawei.com"
parameters:
...
volumeType: fs
replication: "true"
replicationSyncPeriod: "3600"
backend: "***"
```

表 7-7 参数说明

参数名	参数描述	备注
replication	指定是否创建远程 复制卷	配置为true,表示需要创建远程复制卷; 不配置或配置为false,表示不创建远程复 制卷。
replicationSync Period	指定远程复制同步 时间间隔	可选,单位为秒,默认为3600s。

参数名	参数描述	备注
backend	指定创建远程复制 卷的主端名	条件必选。在使用租户的场景下,如果租户属于远程复制租户pair,远程复制卷只能在租户pair主端创建,所以需要通过backend参数指定创建复制卷的主端名,对应于huawei-csi-configmap.yaml中配置的后端名。

使用该StorageClass创建的新卷即为具有NAS远程复制能力的卷。

7.5.6 配置 NAS 双活

须知

使用NAS双活,需要事先在两台存储之间完成双活关系配置,并创建双活租户pair。具体请参考华为存储相关配置指导。

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,并且为每个后端增加metrovStorePairID配置项,如下所示:

```
"backends":[

"name": "hyperMetro1",

"vstoreName": "***",

"metroBackend": "hyperMetro2"
},

{
...

"name": "hyperMetro2",

"vstoreName": "***",

"metrovStorePairID": "***",

"metrovStorePairID": "***",

"metrovStorePairID": "***",

"metrovStorePairID": "***",

"metrovStorePairID": "***",

"metrobackend": "hyperMetro1"
}

}
```

山 说明

- NAS双活仅支持租户。
- metrovStorePairID是租户归属的双活租户pair ID。
- metroBackend是双活对端的后端名,两个双活后端相互构成双活关系。如上所示,后端 hyperMetro1的双活对端是hyperMetro2,相应的,hyperMetro2的双活对端是 hyperMetro1。

在StorageClass yaml文件的parameters配置项下,添加以下配置项。

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

volumeType: fs hyperMetro: "true"

表 7-8 参数说明

参数名	参数描述	备注
hyperMetro	指定是否创建双活 卷	配置为true,表示需要创建双活卷;不配 置或配置为false,表示不创建双活卷。

使用该StorageClass创建的新卷即为具有NAS双活能力的卷。

7.6 分布式存储高级特性

7.6.1 配置 QoS

本章节介绍如何使用分布式存储创建支持QoS的LUN卷。

步骤1 配置StorageClass yaml文件,在parameters配置项下添加qos配置项,示例如下:

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

... qos: '{"maxMBPS": 999, "maxIOPS": 999}'

qos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)。

步骤2 使用该StorageClass创建新的PVC

----结束

表 7-9 qos 支持的参数说明

参数名	参数描述	备注
maxMBPS	最大带宽	必填。有效值为>0的整数,单位 MB/s
maxIOPS	最大IOPS	必填。有效值为>0的整数

8 FAQ

- 8.1 查看日志信息
- 8.2 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
- 8.3 启动huawei-csi-node失败,提示错误为: "/var/lib/iscsi is not a directory"
- 8.4 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

8.1 查看日志信息

huawei-csi的运行日志目录为/var/log/huawei,csi-controller的日志文件是huawei-csi-controller,csi-node的日志文件是huawei-csi-node,配置secret的日志文件为huawei-csi-install。可通过如下命令查询csi-controller和csi-node所在节点。

```
# kubectl get pod -A -o wide | grep huawei kube-system huawei-csi-controller-695b84b4d8-tg64l 3/3 Running 0 14s <host1-ip> <host1-name> <none> <none> kube-system huawei-csi-node-g6f7z 2/2 Running 0 14s <host2-ip> <host2-name> <none>
```

8.2 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

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'",这是由于搭建Kubernete和安装iscsi服务后, iscsi_tcp服务可能会被停掉,可以使用"lsmod | grep iscsi"查看iscsi_tcp服务是否启动,如果没启动,需要手动使用"modprobe iscsi_tcp"启动。

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

启动huawei-csi-node时, 无法启动huawei-csi-node服务。这是因为容器内部 无/var/lib/iscsi目录,只需将huawei-csi-node.yaml中hostpath为/var/lib/iscsi的type 设置为 " 或者删除type这一行

8.4 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: 1.13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:

排查方法:

步骤1 执行命令: multipath -ll, 判断是否存在多路径状态异常的dm设备:

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

#mpathb (3618cf24100f8f457014a764c000001f6) dm-12 HUAWEI ,XSG1 size=100G features='0' hwhandler='0' wp=rw `-+- policy='service-time 0' prio=-1 status=active |- 39:0:0:1 sdi 8:48 failed faulty running

`- 38:0:0:1 sdj 8:64 failed faulty running

l 是 => 步骤2。

l 否 => 不涉及。

步骤2 判断残留的dm设备是否可读

执行命令: dd if=/dev/dm-xx of=/dev/null count=1 bs=10M iflag=direct

(dm-xx为步骤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=10M iflag=direct dd: error reading '/dev/dm-12': Input/output error 0+0 records in 0+0 records out 0 bytes (0 B) copied, 0.0236862 s, 0.0 kB/s

l 是 => 记录残留的dm-xx设备以及关联磁盘号(见步骤一),执行清理步骤。

l 否 => 联系技术支持。

----结束

清理步骤:

步骤1 根据"排查方法"获取的dm设备,执行命令: multipath -f /dev/*dm-xx*,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。

步骤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

步骤3 确认dm设备和scsi磁盘信息是否已经清理干净

依次执行命令"multipath -ll"、"ls -l /sys/block/"、"ls -l /dev/disk/by-id/",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
size=100G features='0' hwhandler='0' wp=rw
-+- policy='service-time 0' prio=-1 status=active
 |- 39:0:0:1
                sdd 8:48 active ready running
 `- 38:0:0:1
                sde 8:64 active ready running
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

`- 38:0:0:2
                sdc 8:32 active ready running
                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
lrwxrwxrux 1 root root 0 Aug 11 19:56 sde -> ../devices/platform/host38/session5/target38:0:0/38:0:0:1/
block/sde
lrwxrwxrux 1 root root 0 Aug 11 19:56 sdh -> ../devices/platform/host39/session6/target39:0:0/39:0:0:3/
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 -> ../../dm-3
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-368886030000020aff44cc0d060c987f1 -> ../../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