



浪潮存储系统 Kubernetes FlexVolume 插件用户手册

文档版本 **1.2**

发布日期 **2019-08-23**

适用版本 **K8sPlugin_V1.0.0**

尊敬的存储系统用户：

衷心感谢您选用了浪潮存储系统！

本手册介绍了本存储系统的主机插件相关配置管理等信息，有助于您更详细地了解 and 便捷地使用本款存储系统。

浪潮拥有本手册的版权。

未经浪潮许可，任何单位和个人不得以任何形式复制本用户手册。浪潮保留随时修改本手册的权利。

本手册中的内容如有变动恕不另行通知。

如果您对本手册有疑问或建议，请向浪潮电子信息产业股份有限公司垂询。

技术服务电话： 4008600011

地 址： 中国济南市浪潮路 1036 号
浪潮电子信息产业股份有限公司

邮 编： 250101

声明

在您正式使用本存储系统之前，请您先阅读以下声明。只有您阅读了以下声明并且同意以下各条款后，方可正式开始使用本存储系统；如果您对以下条款有任何疑问，请和您的供货商联系或直接与我们联系。如您未向我们就以下条款提出疑问并开始使用本系统，则是默认您已经同意了以下各条款。

1. 在您使用的存储系统出现任何硬件故障或您希望对硬件进行任何升级时，请您将机器的详细硬件配置反映给我们的客户服务中心；您不要自行拆卸存储系统机箱及机箱内任何硬件设备。
2. 本存储系统的内存、CPU、CPU 散热片、风扇、硬盘托架、硬盘等都是特殊规格的，请您不要将它们和任何其他型号机器的相应设备混用。
3. 您在使用存储系统过程中遇到的任何软件问题，我们希望您首先和相应软件的提供商联系，由他和我们联系，以方便我们沟通、共同解决您遇到的问题。对于如数据库、网络管理软件或其他网络产品等的安装、运行问题，我们尤其希望您能够这样处理。
4. 如果上架安装本存储系统，请先仔细阅读相关产品手册中的快速安装指南。浪潮致力于产品功能和性能的持续提升，这可能导致部分功能及操作与手册描述有所差异，但不会影响使用，如果您有任何使用疑难问题，请与我们的客户服务中心联系。
5. **我们特别提醒您：在使用过程中，注意对您的数据进行必要的备份。**
6. 此为 A 级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，可能需要用户对其干扰采取切实可行的措施。
7. 请仔细阅读并遵守本手册的安全细则。
8. 本手册中涉及的各项、硬件产品的标识、名称版权归产品的相应公司拥有。
9. 以上声明中，“我们”指代浪潮电子信息产业股份有限公司；浪潮电子信息产业股份有限公司拥有对以上声明的最终解释权。

安全细则

1. 本系统中的电源设备可能会产生高电压和危险电能，从而导致人身伤害。请勿自行卸下主机盖以拆装、更换系统内部的任何组件，除非另外得到浪潮的通知，否则只有经过浪潮培训的维修技术人员才有权拆开主机盖及拆装、更换内部组件。
2. 请将设备连接到适当的电源，仅可使用额定输入标签上指明的外部电源为设备供电，为保护您的设备免受电压瞬间升高或降低所导致的损坏，请使用相关的稳压设备或不间断电源设备。
3. 如果必须使用延长线缆，请使用配有正确接地插头的三芯线缆，并查看延长线缆的额定值，确保插入延长线缆的所有产品的额定电流总和不超过延长线缆额定电流限制的百分之八十。
4. 请务必使用随机配备的供电组件如电源线、电源插座（如果随机配备）等，为了设备及使用者的安全，不要随意更换电源线缆或插头。
5. 为防止系统漏电造成电击危险，务必将系统和外围设备的电源电缆插入已正确接地的电源插座。在未安装接地导线及不确定是否已有适当接地保护的情况下，请勿操作使用本设备，请与电工联系。
6. 切勿将任何物体塞入系统的开孔处。如果塞入物体，可能会导致内部组件短路而引起火灾或电击。
7. 请将系统置于远离散热片和有热源的地方，切勿堵塞通风孔。
8. 切勿让食物或液体散落在系统内部或其它组件上，不要在高潮湿、高灰尘的环境中使用产品。
9. 用错误型号的电池更换会有爆炸危险，需要更换电池时，请先向制造商咨询并使用与制造商推荐型号相同或相近的电池，切勿拆开、挤压、刺戳电池或使其外部接点短路，不要将其丢入火中或水中，也不要暴露在温度超过 60 摄氏度的环境中，请勿尝试打开或维修电池，务必合理处置用完的电池，不要将用完的电池及可能包含电池的电路板及其它组件与其它废品放在一起，有关电池回收请与当地废品回收处理机构联系。

目 录

声 明	ii
安全细则.....	iii
1 功能描述	1
1.1 基本介绍.....	1
1.2 约束与限制.....	2
1.3 应用场景.....	2
2 安装与部署	3
2.1 浪潮 Kubernetes FlexVolume 插件安装.....	3
2.2 Kubernetes 集群搭建.....	4
3 功能配置与管理	5
3.1 网络配置.....	5
3.2 启用多路径.....	6
3.3 完善浪潮存储插件配置文件.....	8
4 Kubernetes 中使用存储.....	10
4.1 直接使用卷.....	10
4.2 通过 PVC 方式使用卷.....	12
5 故障分析与解决	16
6 术语&缩略语	17
附录	19

1 功能描述

1.1 基本介绍

浪潮 Kubernetes FlexVolume 主机插件使得浪潮 InStorage 存储可以为 Kubernetes 集群中的应用提供持久化存储。

该插件实现了所有的 Kubernetes FlexVolume 接口，通过这些接口，插件可以实现卷在主机端的挂载、卸载等相关功能。当 Kubernetes 集群中的一个服务需要使用浪潮存储提供的 PV（Persistent Volume）时，Kubernetes 会通过 FlexVolume 接口来调用插件，插件自动完成卷的挂载动作，在需要时，会自动对卷进行格式化，并将卷上的文件系统挂载到 Kubernetes 指定的目录上，随后，Kubernetes 可以将该目录映射到服务的容器中，供服务使用。

当前该插件主要实现了主机端的相关功能。详细功能如下：

表 1-1 Kubernetes FlexVolume 实现的功能

序号	功能模块	操作
1	卷-主机操作	挂载卷到主机
2		mount 卷上的文件系统到主机
3		从主机上 unmount 卷上的文件系统
4		从主机上卸载卷
5	存储接口协议	iSCSI
6		FC
7	多路径	支持多路径设备

1.2 约束与限制

表 1-2 约束与限制

支持 Kubernetes 版本:	1.10 及以上
支持功能:	卷的自动挂载、卸载, 支持 FC/iSCSI, 支持多路径
支持 Linux 内核版本:	4.4 及以上

适用存储产品类型

AS2150G2&AS2200G2&AS2600G2&AS5300G2&AS5500G2&AS5600G2&AS5800G2&AS6800G2

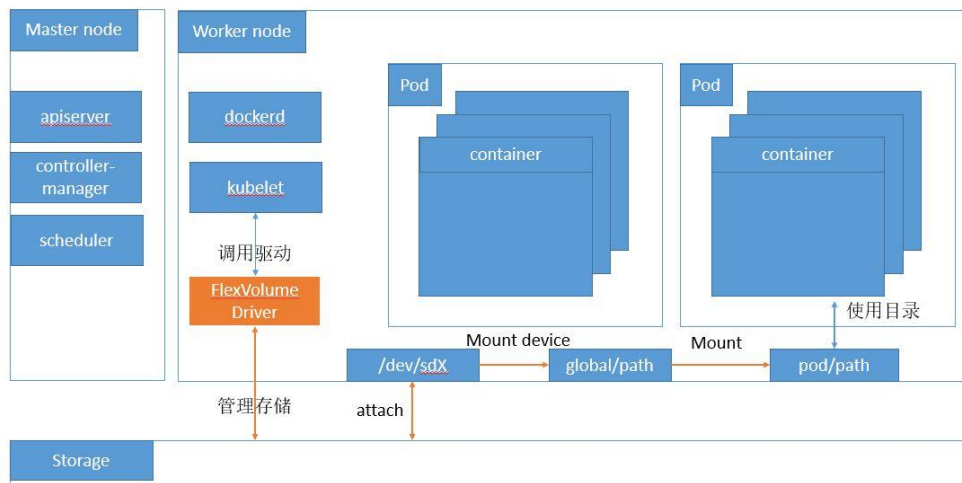
AS2600G2-F&AS5300G2-F&AS5500G2-F&AS5600G2-F&AS5800G2-F&AS6800G2-F

AS5300G5&AS5500G5&HF5000G5

1.3 应用场景

当前浪潮 Kubernetes FlexVolume 插件实现了 FlexVolume 插件接口, 用于自动完成存储卷在主机端的挂载, 卸载功能。拥有了该插件, 就可以在 Kubernetes 集群中使用浪潮存储。该插件需要部署在 Kubernetes 集群中的所有节点。拓扑图如图 1-1 所示。

图 1-1 Kubernetes FlexVolume 主机插件应用拓扑图



2 安装与部署

为了在 Kubernetes 集群中使用浪潮存储，需要完成存储的初始化，并将浪潮存储的插件部署在集群中的各个节点的指定目录，以及根据存储的访问信息，对插件的配置文件进行修改。在部署插件前，需要确保 Kubernetes 集群状态正常。

说明：部署插件，无需重启 Kubernetes 服务。

2.1 浪潮 Kubernetes FlexVolume 插件安装

浪潮 Kubernetes FlexVolume 插件是一个可执行文件，通过将插件部署到 FlexVolume 指定的目录中即可。具体过程如下：

1. 打开浪潮存储随机光盘中的 Kubernetes FlexVolume 驱动安装包。

解压并查看浪潮 Kubernetes FlexVolume 插件安装包中包含的文件：

图 2-1 安装包中包含的文件

```
root@k8s-master:~# ls -l
total 1992
-rw-r--r-- 1 root root 2039442 Sep 17 11:04 K8sPlugin_V1.0.0.Build20180917.zip
root@k8s-master:~# unzip K8sPlugin_V1.0.0.Build20180917.zip
Archive:  K8sPlugin_V1.0.0.Build20180917.zip
  creating: K8sPlugin_V1.0.0.Build20180917/
  creating: K8sPlugin_V1.0.0.Build20180917/inspur-instorage/
  creating: K8sPlugin_V1.0.0.Build20180917/inspur-instorage/config/
  inflating: K8sPlugin_V1.0.0.Build20180917/inspur-instorage/config/instorage.yaml
  inflating: K8sPlugin_V1.0.0.Build20180917/inspur-instorage/instorage
  creating: K8sPlugin_V1.0.0.Build20180917/inspur-instorage/log/
root@k8s-master:~# ls -Rl
.:
total 1996
drwxr-xr-x 3 root root 4096 Sep 17 11:02 K8sPlugin_V1.0.0.Build20180917
-rw-r--r-- 1 root root 2039442 Sep 17 11:04 K8sPlugin_V1.0.0.Build20180917.zip

./K8sPlugin_V1.0.0.Build20180917:
total 4
drwxr-xr-x 4 root root 4096 Aug 6 13:46 inspur-instorage

./K8sPlugin_V1.0.0.Build20180917/inspur-instorage:
total 5800
drwxr-xr-x 2 root root 4096 Sep 17 11:00 config
-rw-r--r-- 1 root root 5930623 Sep 17 10:58 instorage
drwxr-xr-x 2 root root 4096 Aug 29 13:46 log

./K8sPlugin_V1.0.0.Build20180917/inspur-instorage/config:
total 4
-rw-r--r-- 1 root root 165 Sep 17 11:00 instorage.yaml

./K8sPlugin_V1.0.0.Build20180917/inspur-instorage/log:
total 0
root@k8s-master:~#
```


2. 确定 Kubernetes 集群中各节点 FlexVolume 插件部署路径。

Kubernetes FlexVolume 插件默认部署路径为 “/usr/libexec/kubernetes/kubelet-plugins/volume/exec”。具体路径需要根据集群的配置进行确定。

3. 将插件安装包中的 inspur~instorage 目录拷贝到第 2 步中的插件部署路径中。

拷贝完成后，结果如图 2-2 所示。

图 2-2 拷贝完成

```
root@k8s-master:/usr/libexec/kubernetes/kubelet-plugins/volume/exec# ls -lR
.:
total 4
drwxr-xr-x 4 root root 4096 Aug  6 13:46 inspur~instorage

./inspur~instorage:
total 8764
drwxr-xr-x 2 root root  4096 Aug 29 13:46 config
-rwxr-xr-x 1 root root 5925604 Aug 21 14:20 instorage
drwxr-xr-x 2 root root 3039232 Aug 29 13:46 log

./inspur~instorage/config:
total 4
-rw-r--r-- 1 root root 298 Aug 29 13:46 instorage.yaml

./inspur~instorage/log:
total 0
```

inspur~instorage 目录为插件目录，目录下 instorage 为插件本身，config 目录用于保存配置文件，log 目录用于保存运行日志。config 目录下的 instorage.yaml 文件为插件的配置文件。

4. 至此，浪潮 Kubernetes FlexVolume 插件部署完毕，后续可参考功能配置与管理部分对插件配置文件进行合理配置。

2.2 Kubernetes 集群搭建

Kubernetes 是一个开源的容器编排管理平台。Kubernetes 集群的搭建过程，使用方法，请参考官方网站 <https://kubernetes.io/> 中的介绍。同时作为一个开源系统，各厂商可以基于 Kubernetes 发行自己的版本，针对厂商版本的使用方法，请参考各厂商提供的使用手册及相关文档。

3 功能配置与管理

为了在 Kubernetes 集群中使用浪潮存储，首先需要确保浪潮存储本身已经完成了初始化，并且完成存储池的创建。存储可以在集群各节点被访问，存储数据层通道（iSCSI、FC）可以正常使用，然后根据集群的具体信息对配置文件进行修改。

3.1 网络配置

管理 IP

驱动需要访问浪潮存储的管理接口，驱动使用 SSH 的方式与管理接口通信。驱动需要配置浪潮存储系统的 IP、SSH 端口（默认为 22）。



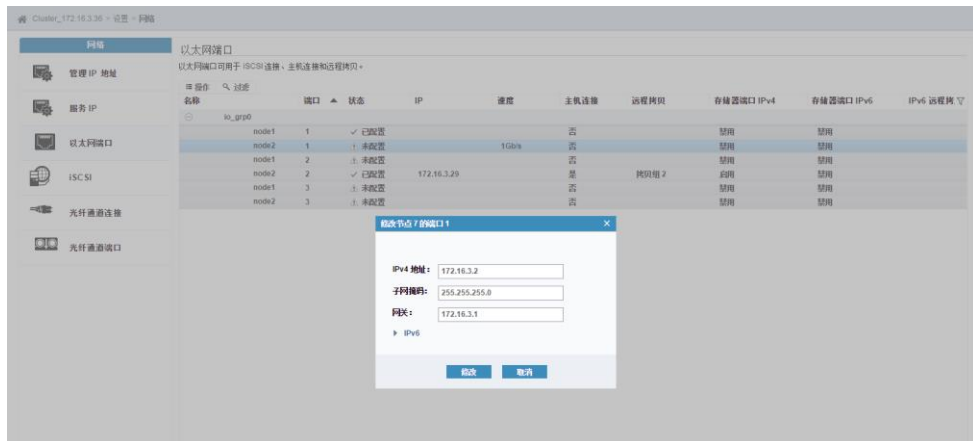
注意

- 确保插件所在节点具有存储系统的 SSH 网络访问权限。
- 浪潮存储设备必须配置有 iSCSI、FC，两者至少有一种，或者两者兼有。

iSCSI 网络配置

如果使用 iSCSI，则需要每个浪潮存储节点至少有一个 iSCSI 的 IP 地址。浪潮 Kubernetes FlexVolume 插件会优先使用卷的优先节点（如果存在）的 iSCSI 的 IP 来挂载卷到实例，否则会使用系统的第一个可用的 iSCSI 的 IP。插件会直接从存储系统中获取 iSCSI IP，用户不需要给插件单独提供 iSCSI IP。虽然不需要单独为驱动分配 iSCSI IP，但是要在存储管理系统中设定 iSCSI 端口的 IP，在“设置 > 网络 > 以太网端口”中，选择已经连通的端口，单击鼠标右键，选择“修改 IP”，填写有效的 IP、子网掩码、网关。如图 3-1 所示。

图 3-1 添加 iSCSI 端口 IP



注意

如果使用 iSCSI，确保各工作节点与存储系统有畅通的 iSCSI 网络可供访问。

FC 网络配置

如果使用 FC，则需要每个浪潮存储节点至少配置有一个 WWPN 端口。插件会使用所有可用的 WWPN 端口将卷挂载目标主机。插件将直接从存储系统中获取 WWPN，用户无需为驱动单独提供 WWPN。



注意

如果使用 FC，确保各工作节点与存储系统有 FC 连接。

3.2 启用多路径

为了提升 SAN 存储卷的可靠性，在生产环境中，通常会启用多路径。

在 Kubernetes 环境中，如果多路径相关配置设置不恰当，在实际使用中有可能使用到单路径设备，如果该路径出现损坏，则可能会产生 I/O 错误，导致虚拟机文件系统变为只读或损坏。

在 Linux 环境下，浪潮存储利用 Linux 系统自带的 multipathd 服务进行多路径聚合，如果需要启用多路径，首先需要保证 Kubernetes 中各工作节点均按照要求部署安装 multipathd 服务。然后对 Kubernetes 环境中的各工作节点多路径相关的配置，请参考下面的描述，以正确使用多路径。

1. 在 `multipath.conf` 配置中正确设置设备黑名单。

在 Kubernetes 环境中，一个工作节点上，会同时挂载非常多的卷，每个卷又会有多条路径，从而使得节点上的 `sdX` 设备会非常多。多路径的配置文件中 `blacklist` 配置组中的参数会过滤符合 `blacklist` 条件的路径。如果该配置过滤的设备不恰当，可能会导致部分路径无法进行多路径聚合。

例如，如果 `devnode` 参数配置成类似“`^sda`”时，会导致 `sdaa`，`sdab` 等路径无法进行路径聚合，以致路径缺失，正确的应该是“`^sda$`”，即仅将 `sda` 设备屏蔽掉。

针对 `multipath.conf` 配置文件的具体用法，Linux 用户可以通过 `man 5 multipath.conf` 获取帮助。请确保不会将浪潮存储提供的设备路径加入设备黑名单。



注意

`multipath.conf` 配置文件的具体路径，请咨询 Kubernetes 容器平台提供商，linux 系统下该文件默认路径为 `/etc/multipath/multipath.conf`。

2. 在 `multipath.conf` 配置文件中，正确设置浪潮存储推荐的设备配置参数。

目前浪潮存储的推荐多路径配置已经合入到多路径工具的社区版本中，针对使用旧版本多路径工具的场景，需要在配置文件中加入浪潮推荐的多路径配置。即在 `devices` 配置组中增加浪潮存储的 `device` 配置内容。通常配置信息如下：

```
devices{
    device {
        vendor            "INSPUR"
        product           "MCS"
        path_grouping_policy group_by_prio
        path_selector     "round-robin 0"
        path_checker      tur
        features          "1 queue_if_no_path"
        hardware_handler  "0"
        prio              alua
        failback          immediate
        rr_weight         uniform
        rr_min_io         1000
    }
}
```

```

    }
}

```

针对 `multipath.conf` 配置文件的具体用法，Linux 用户可以通过 `man 5 multipath.conf` 获取帮助。



注意

`multipath.conf` 配置文件的具体路径，请咨询 Kubernetes 容器平台提供商，linux 系统下该文件默认路径为 `/etc/multipath/multipath.conf`。

`devices` 配置组中可能会存在多个存储厂商的 `device` 配置。

3.3 完善浪潮存储插件配置文件

浪潮存储插件配置文件为插件目录中的 `config/instorage.yaml` 文件。文件为 `yaml` 格式。具体配置如下：

```

log:
  enabled: <false>
  logdir: <log>
  level: <0>
host:
  link: <iscsi or fc>
storage:
- name: <storage-01>
  host: <10.0.0.1:22>
  username: <username>
  password: <password>

```

配置说明如下：

表 3-1 配置说明

配置名称	说明
<code>log.enabled</code>	是否打开插件日志。 <code>true</code> 打开， <code>false</code> 不打开。
<code>log.logdir</code>	日志输出目录。
<code>Log.level</code>	日志输出级别。0-4，数值越高，输出日志越详细。
<code>Host.link</code>	数据通道连接类型。

	iscsi 使用 iSCSI 连接方式。 fc 使用 FC 连接方式。
storage[].name	存储名称。配置文件范围内唯一，区分多个存储。当前只支持一个存储。
storage[].host	存储 SSH 访问路径。 格式为 IP:Port。
storage[].username	存储 SSH 访问时的用户名。
storage[].password	存储 SSH 访问时的密码。

4 Kubernetes 中使用存储

通过浪潮 Kubernetes FlexVolume 插件，可以实现在 Kubernetes 集群中使用浪潮存储。Kubernetes 中使用持久化存储的方式包括直接使用卷和通过 PVC（Persistent Volume Claim）使用卷等两种方式。

4.1 直接使用卷

直接使用卷，即在创建 Pod 或者 Pod 模板的时候，通过 Volume 部分直接指定目标卷的方式。过程如下：

1. 确存储端已经创建了待使用的卷。

图 4-1 卷信息



Name	State	Pool	UID	Host Mappings
instorage-120-test04	✓ Online	Pool0	60050760008989C0D00000000002C4B5	No

2. 通过 kubectl 创建使用该卷的 Pod。

```
k8s@dev:~/k8s$ cat deploy-use-volume-direct.yaml
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: deploy-nginx-02
  labels:
    app: nginx-02
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx-02
  template:
    metadata:
      name: nginx-02
      labels:
        app: nginx-02
    spec:
```

```

containers:
  - name: nginx
    image: nginx
    ports:
      - containerPort: 80
    volumeMounts:
      - name: instorage-120-test04
        mountPath: /mnt
volumes:
  - name: instorage-120-test04
    flexVolume:
      driver: "inspur/instorage"
      fsType: "ext4"
k8s@dev:~/k8s$ kubectl create -f deploy-use-volume-direct.yaml

```



注意

spec.template.spec.volumes[].name 所使用的卷名称需要与存储上的卷名称保持一致。

- 查看 Pod 已经处于 Running 状态。

```

k8s@dev:~/k8s $ kubectl get pod
NAME                                READY   STATUS    RESTARTS
AGE
deploy-nginx-02-79c4d4644f-5pww8    1/1    Running   0
22s
deploy-nginx-02-79c4d4644f-bmscf    1/1    Running   0
22s

```

- 查看存储端卷已经映射到主机。

图 4-2 卷映射信息

Name	Status	Host Type	# of Ports	Host Mappings
iscsi-k8s-worker-01	✓ Online	Generic	1	Yes
iscsi-k8s-worker-02	✓ Online	Generic	1	Yes

SCS...	Name	UID	Cach...
0	instorage-120-test04	60050760008989C0D00000000002C4B5	0

- 查看工作节点上，卷被正确的映射和挂载。

图 4-3 卷的映射和挂载信息

```
root@k8s-worker-01:~# ll /dev/disk/by-path/ | grep iscsi
lrwxrwxrwx 1 root root 9 Aug 29 16:28 ip-10.180.210.123:3260-iscsi-iqn.2004-12.com.inspur:mcs.cluster10.180.210.120.node1-lun-0 -> ../../sdb
root@k8s-worker-01:~# mount | grep flex
/dev/sdb on /var/lib/kubelet/plugins/kubernetes.io/flexvolume/inspur/instorage/mounts/instorage-120-test04 type ext4 (rw,relatime,stripe=8,data=ordered)
```

6. 当应用不在需要时，可以通过 kubelet 将其删除。

```
k8s@dev:~/k8s$ kubectl delete -f deploy-use-volume-direct.yaml
```

7. 当 Pod 结束后，卷会重主机端卸载，并且映射删除。在存储端可以看到卷已经与主机解映射。

4.2 通过 PVC 方式使用卷

通过 PVC 方式使用卷时，需要首先创建 StorageClass，作为 PV 和 PVC 之间关联的信息。然后管理员在存储端创建待使用卷，并在 Kubernetes 集群中创建对应的 PV 来表示该卷。用户在使用时，创建 PVC 请求 PV 资源，然后创建 Pod 来使用 PVC。过程如下：

1. 在 Kubernetes 集群中创建 StorageClass。

```
k8s@dev:~/k8s$ cat sc.yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: instorage120
provisioner: inspur/instorage
reclaimPolicy: Retain
k8s@dev:~/k8s$ kubectl create -f sc.yaml
storageclass.storage.k8s.io "instorage120" created
k8s@dev:~/k8s$ kubectl get sc
NAME          PROVISIONER      AGE
instorage120  inspur/instorage 10s
```

2. 确存储端已经创建了待使用的卷。

图 4-4 卷信息

Name	State	Pool	UID	Host Mappings
instorage-120-test04	✓ Online	Pool0	60050760008989C0D0000000002C4B5	No

3. 为存储上的待使用的卷在 Kubernetes 集群中创建 PV。

```
k8s@dev:~/k8s$ cat pv.yaml
```

```

apiVersion: v1
kind: PersistentVolume
metadata:
  name: instorage-120-test01
spec:
  accessModes:
    - ReadWriteOnce
  capacity:
    storage: 15Gi
  volumeMode: Filesystem
  persistentVolumeReclaimPolicy: Retain
  storageClassName: instorage120
  flexVolume:
    driver: "inspur/instorage"
    fsType: "ext4"
k8s@dev:~/k8s$ kubectl create -f pv.yaml
persistentvolume "instorage-120-test01" created
k8s@dev:~/k8s$ kubectl get pv

```

NAME	CAPACITY	ACCESS MODES	RECLAIM
POLICY	STATUS	CLAIM	STORAGECLASS
AGE	REASON		
instorage-120-test01	15Gi	RWO	Retain
Available	instorage120	5s	

4. 在 Kubernetes 集群中创建 PVC，来声明使用存储资源。

```

k8s@dev:~/k8s$ cat pvc.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: pvc-data
spec:
  accessModes:
    - ReadWriteOnce
  volumeMode: Filesystem
  resources:
    requests:
      storage: 13Gi
  storageClassName: instorage120
k8s@dev:~/k8s$ kubectl create -f pvc.yaml
persistentvolumeclaim "pvc-data" created
k8s@dev:~/k8s$ kubectl get pvc

```

NAME	STATUS	VOLUME	CAPACITY
ACCESS MODES	STORAGECLASS	AGE	

pvc-data	Bound	instorage-120-test01	15Gi	RWO
instorage120	9s			
k8s@dev:~/k8s\$ kubectl get pv				
NAME		CAPACITY	ACCESS MODES	RECLAIM
POLICY	STATUS	CLAIM		STORAGECLASS
REASON	AGE			
instorage-120-test01	15Gi		RWO	Retain
Bound	default/pvc-data	instorage120		3m

5. 利用 PVC 创建 Pod 来使用 PVC（通过 Deployment 创建 Pod）。

```
k8s@dev:~/k8s$ cat deploy.yaml
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: deploy-nginx-01
  labels:
    app: nginx-01
spec:
  replicas: 1
  selector:
    matchLabels:
      app: nginx-01
  template:
    metadata:
      name: nginx-01
      labels:
        app: nginx-01
    spec:
      containers:
        - name: nginx
          image: nginx
          ports:
            - containerPort: 80
          volumeMounts:
            - name: data
              mountPath: /mnt
      volumes:
        - name: data
          persistentVolumeClaim:
            claimName: pvc-data
k8s@dev:~/k8s$ kubectl create -f deploy.yaml
deployment.extensions "deploy-nginx-01" created
k8s@dev:~/k8s$ kubectl get pod
```

NAME	READY	STATUS
RESTARTS AGE		
deploy-nginx-01-6bd55685d5-5n4wg 0/1	ContainerCreating	0
14s		
k8s@dev:~/k8s\$ kubectrl get pod		
NAME	READY	STATUS
RESTARTS AGE		
deploy-nginx-01-6bd55685d5-5n4wg 1/1	Running	0
3m		

6. 查看工作节点上，卷被正确的映射和挂载。

图 4-5 卷的映射和挂载信息

```

root@k8s-worker-01:~# ll /dev/disk/by-path/ | grep iscsi
lrwxrwxrwx 1 root root 9 Aug 29 18:08 ip-10.180.210.123:3260-iscsi-iqn.2004-12.com.inspur:mcs.cluster10.180.210.120.node1-lun-0 -> ../../sdb
root@k8s-worker-01:~# mount | grep flex
/dev/sdb on /var/lib/kubelet/plugins/kubernetes.io/flexvolume/inspur/instorage/mounts/instorage-120-test01 type ext4 (rw,relatime,stripe=8,data=ordered)

```

5 故障分析与解决

1. 网络不通。Kubernetes 安装浪潮存储 Kubernetes FlexVolume 的各节点需要与存储的管理网络互通，Kubernetes FlexVolume 插件需要通过 ssh 连接到存储上执行存储端的相关命令来完成卷操作相关动作。
2. 认证失败，无法连接到浪潮存储设备，请检查用户名、密码是否配置正确。

6 术语&缩略语

A		
API	Application Program Interface	应用程序接口
D		
Deployment	-	Kubernetes 集群中部署业务的一种形式
F		
FC	Fiber Channel	光纤通道
-	FlexVolume	Kubernetes 集群存储管理的一种插件接口
I		
iSCSI	Internet Small Computer System Interface	互联网小型计算机接口
InStorage	Inspur Storage	浪潮存储
K		
-	Kubernetes	Kubernetes 容器管理平台
K8s	Kubernetes	Kubernetes 缩写
K8s Plugin	Kubernetes Plugin	浪潮存储 Kubernetes 存储管理平台插件
P		
-	Pod	Kubernetes 集群中的最小业务单元
PV	Persistent Volume	Kubernetes 集群中持久卷资源

PVC	Persistent Volume Claim	Kubernetes 集群中持久卷声明资源
S		
SC	StorageClass	Kubernetes 集群中的存储类型资源
SSH	Secure Shell Protocol	一种安全协议
W		
WWPN	World Wide Port Name	全球端口名称

附录

浪潮所开发的产品属于 Kubernetes FlexVolume 的插件。Kubernetes 中的相关概念，部署配置方法，使用方法等知识，请参考 Kubernetes 社区中的相关资料，或通过 Kubernetes 提供商获取相关使用资料。以下提供了部分社区公开资料：

1. Kubernetes 社区官方网站：

<https://kubernetes.io/>

2. Kubernetes 社区部署安装资料：

<https://kubernetes.io/docs/setup/>

3. Kubernetes 社区存储相关资料：

<https://kubernetes.io/docs/concepts/storage/volumes/>

4. Kubernetes 开发设计等过程即资料均托管在 GitHub 上：

<https://github.com/kubernetes/>

5. Kubernetes 存储相关开发信息

<https://github.com/kubernetes/community/tree/master/sig-storage>