

## Create an Empty Volume

### Header Section

1. Set the Volume `Name` .
2. (Optional) Provide a `Description` for the Volume.

### Basics Tab

1. Choose `New` in `Source` .
2. Select an existing `StorageClass` .
3. Configure the `Size` of the volume.



create-empty-volume

### Oversized Volumes

In Harvester v1.5.0, which uses Longhorn v1.8.1, oversized volumes (for example, 999999 Gi in size) are marked **Not Ready** and cannot be deleted.

To resolve this issue, perform the following steps:

1. Temporarily remove the PVC webhook rule.

```
kubectl patch validatingwebhookconfiguration longhorn-webhook-validator \
  --type='json' \
  -p='[{"op": "remove", "path": "/webhooks/0/rules/17"}]'
```

2. Wait for the related PVC to be deleted.
3. Restore the PVC webhook rule to re-enable validation.

```
kubectl patch validatingwebhookconfiguration longhorn-webhook-validator \
  --type='json' \
  -p='[{"op": "add", "path": "/webhooks/0/rules/-", "value": {"apiGroups":
[""], "apiVersions": ["v1"], "operations": ["UPDATE"], "resources":
["persistentvolumeclaims"], "scope": "Namespaced"}}]'
```

The issue will be addressed in Longhorn v1.8.2, which will likely be included in Harvester v1.5.1.

Related issues:

- Harvester: [Issue #8096](#)
- Longhorn: [Issue #10741](#)

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  annotations:
    volume.beta.kubernetes.io/storage-provisioner: driver.longhorn.io
    volume.kubernetes.io/storage-provisioner: driver.longhorn.io
  name: my-vol
```

```

    namespace: default
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 10Gi
  volumeMode: Block
  volumeName: pvc-my-vol

```

To create an empty volume on Harvester with Terraform using the [Harvester Terraform Provider](#), define a `harvester_volume` resource block:

```

resource "harvester_volume" "empty-volume" {
  name      = "empty-volume"
  namespace = "default"

  size = "10Gi"
}

```

## Create an Image Volume

### Header Section

1. Set the Volume Name .
2. (Optional) Provide a Description for the Volume.

### Basics Tab

1. Choose VM Image in Source .
2. Select an existing Image .
3. Configure the Size of the volume.

:::info important

When creating volumes from a VM image, ensure that the volume size is greater than or equal to the image size. The volume may become corrupted if the configured volume size is less than the size of the underlying image. This is particularly important for qcow2 images because the virtual size is typically greater than the physical size.

By default, Harvester will set the volume size to the virtual size of the image.

:::

 create-image-volume

Create a volume, initialized with the contents of the image `image-8rb2z` from the namespace `default` :

```

apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  annotations:
    harvesterhci.io/imageId: default/image-8rb2z

```

```
  volume.beta.kubernetes.io/storage-provisioner: driver.longhorn.io
  volume.kubernetes.io/storage-provisioner: driver.longhorn.io
name: foobar
namespace: default
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 5Gi
  storageClassName: longhorn-image-8rb2z
  volumeMode: Block
  volumeName: pvc-foobar
```

To create a volume on Harvester using Terraform and initialize it with the contents of an image, define a `harvester_volume` resource block and set the `image` property:

```
resource "harvester_volume" "opensuse154-image-disk" {
  name      = "opensuse154-image-disk"
  namespace = "default"

  size = "10Gi"
  image = harvester_image.opensuse154.id
}
```