Create and Use a Storage Class

Lab 30

What Are You Learning?

In this lab we'll quickly create a storage class for a cloud provider and use it with a workload.

Why Is It Important?

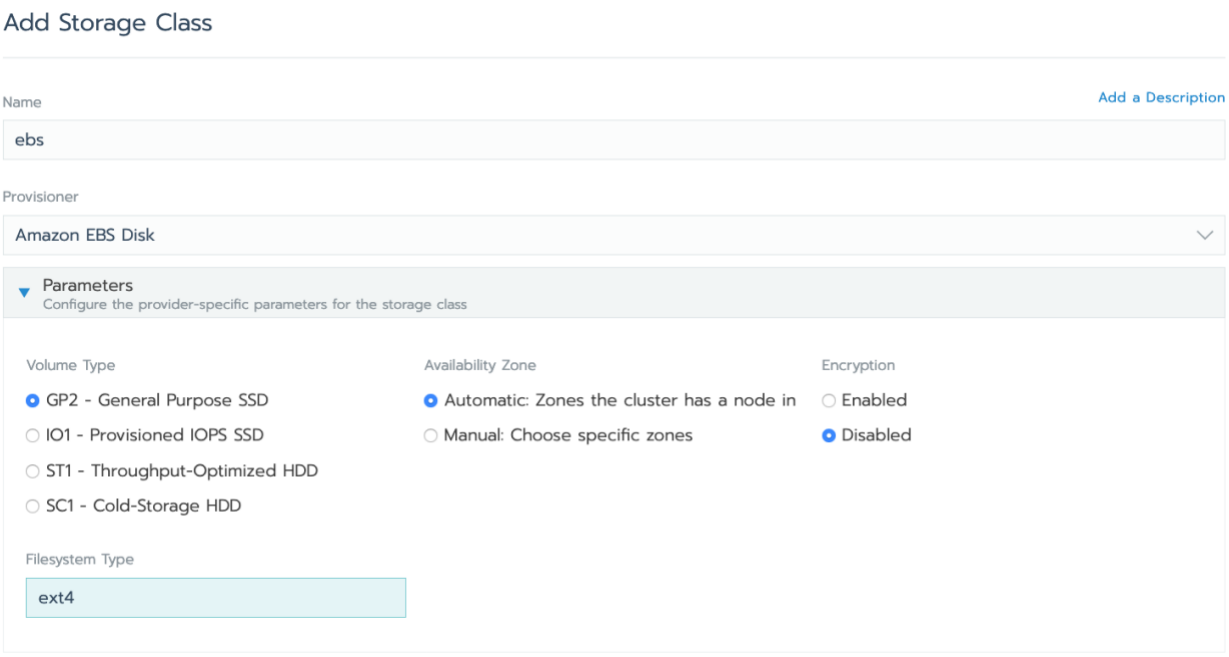
Storage classes enable workloads to request storage from the cluster. This ensures that users can receive the resources they need for their workloads, while operations staff can maintain control over the actual backend resources themselves. An operator can define multiple classes of storage, such as fast SSD storage, slower rotational storage, NFS, iSCSI, and other classes. The consumer of the storage doesn't know where or what it is, only that it exists for consumption.

Create a Storage Class

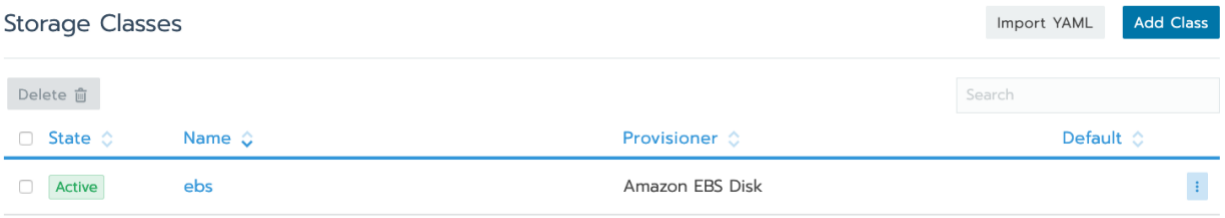
This example shows how to create a StorageClass from the available options within Rancher. These are for cloud providers or Custom providers where you know the configuration. There are also Apps that you can install which will create a StorageClass during their installation.

For example, if you have an NFS server, you can install the "nfs-client-provisioner" app and configure it with a large NFS volume. This will create a StorageClass that makes that volume available for consumption.

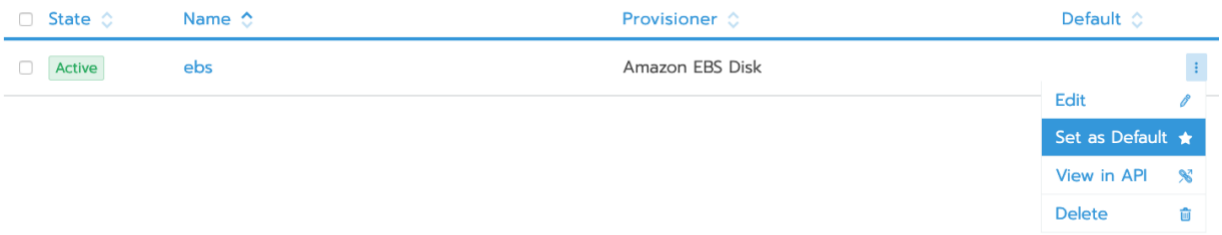
1. From the cluster overview screen, choose Storage >> Storage Classes from the menu at the top.
2. Click the Add Class button in the top right.
3. Give your storage class a name, like "ebs".
4. Select the provider. Our cluster is in Amazon, so I'll choose "Amazon EBS Disk".
5. Set the corresponding options in the section below. I'll choose GP2 disks with automatic placement and no encryption.



6. Review the options under **Customize** but leave them at their defaults.
7. Click **Save**
8. The StorageClass has been created.



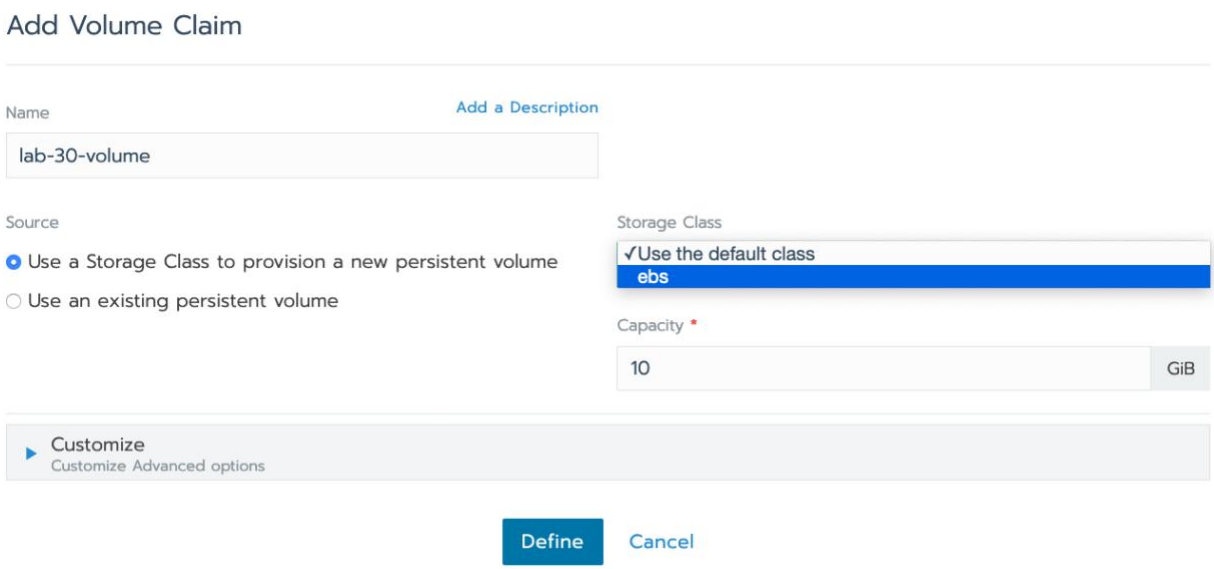
9. If you wish, you can set this StorageClass to be the default for storage requests that don't specify a class.



Use the Storage Class

For StorageClasses in cloud providers, the worker nodes must have a policy attached to them that enables them to create storage resources. This is what enables anyone to request storage without needing their own credentials with the cloud provider. If your hosts do not have this, the following steps will not succeed.

1. From a Project, choose **Resources >> Workloads** and then click the **Deploy** button in the top right.
2. Set the workload to launch the nginx image. Configure other values as needed and then twirl open the **Volumes** section.
3. Click **Add Volume** and choose **Add a new persistent volume (claim)**
4. Give your volume a name and capacity and select your StorageClass from the dropdown. Click **Define** to close the dialog.



5. Set a mountpoint where the storage will be connected inside the Pod. When ready, click **Launch**.

Testing That It Works

1. When your Pod is running, you can shell into it and look at the output of "**df -h**" to see that an external volume is attached and mounted at the specified location.
2. You can also connect to the console of your cloud provider and see that a volume was created.
3. If you left the default options when creating the storage class, then you can delete your workload and then delete the PersistentVolume from **Storage >> Persistent Volumes** at the cluster level. This will delete the actual volume from the cloud provider, but it's always a good idea to verify that no orphaned volumes exist in the provider..

References

- Persistent Volumes and Storage Classes - https://rancher.com/docs/rancher/v2.x/en/cluster-admin/volumes-and-storage/
- Storage Classes - https://kubernetes.io/docs/concepts/storage/storage-classes/