Use persistent volumes in Redis Enterprise clusters

To deploy a Redis Enterprise cluster with Redis Enterprise operator the spec should include a persistentSpec section, in the redis-enterprise-cluster.yaml file:

```
spec:
  nodes: 3
  persistentSpec:
    enabled: true
    storageClassName: "standard"
    volumeSize: "23Gi" #optional
```

Persistence storage is a requirement for this deployment type.

**Note:** For production deployments of Redis Enterprise Cluster on Kubernetes, the Redis Enterprise Cluster (REC) must be deployed with persistence enabled. The REC deployment files in the Kubernetes documentation contain this declaration by default.

### Volume size

`volumeSize` is an optional definition. By default, if the definition is omitted, Operator allocates five times (5x) the amount of memory (RAM) defined for nodes (see example below), which is the recommended persistent storage size as described in the Hardware requirements article.

To explicitly specify the persistent storage size, use the `volumeSize` property as described in the example above.

**Warning** - Be aware the persistent volume size cannot be changed after deployment. Trying to change this value after deployment could result in unexpected and potentially damaging behavior. Please be sure your specified `volumeSize` is correct at the time of creation.

**Note:** We recommend that you omit the `volumeSize` definition from the REC declaration so that the Redis Enterprise Cluster deployment on Kubernetes use the default volume size.

### Storage class name

`storageClassName` determines the Storage Class resource, which is defined by the Kubernetes cluster administrator, to be used for persistent storage.

Different Kubernetes distributions and different deployments use different Storage Class resources.

In order to determine the Storage Class resources available for your K8s deployment, use the following command:

```
kubectl get StorageClass
```

Typically, AWS provides "gp2" as the Storage Class name while GKE uses "standard." Azure provides two Storage Classes: "default" using HDDs, and "managed-premium" using SSDs.

Below is an example of a response to the command.
<table>
<thead>
<tr>
<th><strong>Name:</strong></th>
<th>gp2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IsDefaultClass:</strong></td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Annotations:</strong></td>
<td>storageclass.beta.kubernetes.io/is-default-class=true</td>
</tr>
<tr>
<td><strong>Provisioner:</strong></td>
<td>kubernetes.io/aws-ebs</td>
</tr>
<tr>
<td><strong>Parameters:</strong></td>
<td>encrypted=false, kmsKeyId=, type=gp2</td>
</tr>
<tr>
<td><strong>AllowVolumeExpansion:</strong></td>
<td>&lt;unset&gt;</td>
</tr>
<tr>
<td><strong>MountOptions:</strong></td>
<td>&lt;none&gt;</td>
</tr>
<tr>
<td><strong>ReclaimPolicy:</strong></td>
<td>Delete</td>
</tr>
<tr>
<td><strong>VolumeBindingMode:</strong></td>
<td>Immediate</td>
</tr>
<tr>
<td><strong>Events:</strong></td>
<td>&lt;none&gt;</td>
</tr>
</tbody>
</table>

**Note:** storageClassName must be specified for this deployment type.

Example of the `redisEnterpriseNodeResources` definition:

```yaml
redisEnterpriseNodeResources:
  limits:
    cpu: "4000m"
    memory: 4Gi
  requests:
    cpu: "4000m"
    memory: 4Gi
```

**Updated:** October 6, 2021