Configuration

NetApp Solutions

NetApp
November 24, 2021

目录

Configuration

对于任何多租户解决方案，任何用户都无法访问比所需更多的集群资源。因此，要在多租户配置中配置的整个资源集将在集群管理员，存储管理员和处理每个项目的开发人员之间进行划分。

下表概括了不同用户要执行的不同任务：

Role	任务
* 集群管理 *	为不同的应用程序或工作负载创建项目
	为 storage-admin 创建 ClusterRoles 和 RoleBindings
	为分配对特定项目的访问权限的开发人员创建角色和角色绑定
	[可选] 配置项目以在特定节点上计划 Pod
* 存储管理 *	在 NetApp ONTAP 上创建 SVM
	创建 Trident 后端
	创建 StorageClasses
	创建存储 ResourceQuotas
* 开发人员 *	验证对已分配项目中的 PVC 或 Pod 的创建或修补访问权限
	验证对在其他项目中创建或修补 PVC 或 Pod 的访问权限
	验证对查看或编辑项目， ResourceQuotas 和 StorageClasses 的访问权限

"下一步：前提条件。"

Configuration

前提条件

- NetApp ONTAP 集群。
- Red Hat OpenShift 集群
- 集群上安装的 Trident 。
- 安装了 tridentctl 和 oc 工具并将其添加到 $path 中的管理工作站。
- 对 ONTAP 的管理员访问权限。
- 对 OpenShift 集群的集群管理员访问。
- 集群已与身份提供程序集成。
- 身份提供程序经过配置，可以有效区分不同团队中的用户。

"下一步：集群管理员任务。"

配置： cluster-admin 任务

Red Hat OpenShift cluster-admin 执行以下任务：

1. 以 cluster-admin 身份登录到 Red Hat OpenShift 集群。
2. 创建两个与不同项目对应的项目。

```
oc create namespace project-1
oc create namespace project-2
```

3. 为 project-1 创建开发人员角色。

```
cat << EOF | oc create -f -
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: project-1
  name: developer-project-1
rules:
  - verbs:
      - '*'
    apiGroups:
      - apps
      - batch
      - autoscaling
      - extensions
      - networking.k8s.io
      - policy
      - apps.openshift.io
      - build.openshift.io
      - image.openshift.io
      - ingress.operator.openshift.io
      - route.openshift.io
      - snapshot.storage.k8s.io
      - template.openshift.io
    resources:
      - '*'
  - verbs:
      - '*'
    apiGroups:
      - ''
    resources:
      - bindings
      - configmaps
```

```
      - endpoints
      - events
      - persistentvolumeclaims
      - pods
      - pods/log
      - pods/attach
      - podtemplates
      - replicationcontrollers
      - services
      - limitranges
      - namespaces
      - componentstatuses
      - nodes
  - verbs:
      - '*'
    apiGroups:
      - trident.netapp.io
    resources:
      - tridentsnapshots
EOF
```

本节中提供的角色定义只是一个示例。必须根据最终用户要求定义开发人员角色。

1. 同样，为 project-2 创建开发人员角色。
2. 所有 OpenShift 和 NetApp 存储资源通常由存储管理员管理。存储管理员的访问由安装 Trident 时创建的 Trident 操作员角色控制。此外，存储管理员还需要访问 ResourceQuotas 来控制存储的使用方式。
3. 在集群中的所有项目中创建一个用于管理 ResourceQuotas 的角色，以将其连接到存储管理员：

```
cat << EOF | oc create -f -
kind: ClusterRole
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: resource-quotas-role
rules:
  - verbs:
      - '*'
    apiGroups:
      - ''
    resources:
      - resourcequotas
  - verbs:
      - '*'
    apiGroups:
      - quota.openshift.io
    resources:
      - '*'
EOF
```

4. 确保集群与组织的身份提供程序集成，并且用户组与集群组同步。以下示例显示身份提供程序已与集群集成并与用户组同步。

```
$ oc get groups
NAME                          USERS
ocp-netapp-storage-admins     ocp-netapp-storage-admin
ocp-project-1                 ocp-project-1-user
ocp-project-2                 ocp-project-2-user
```

1. 为存储管理员配置 ClusterRoleBindings 。

```
cat << EOF | oc create -f -
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: netapp-storage-admin-trident-operator
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-netapp-storage-admins
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: trident-operator
---
kind: ClusterRoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: netapp-storage-admin-resource-quotas-cr
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-netapp-storage-admins
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: resource-quotas-role
EOF
```

对于存储管理员，必须绑定两个角色： Trident 操作员和资源配额。

1. 为开发人员创建 RoleBindings ，将开发人员项目 1 角色绑定到项目 1 中的相应组（ OCP-project-1 ）。

```
cat << EOF | oc create -f -
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: project-1-developer
  namespace: project-1
subjects:
  - kind: Group
    apiGroup: rbac.authorization.k8s.io
    name: ocp-project-1
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: developer-project-1
EOF
```

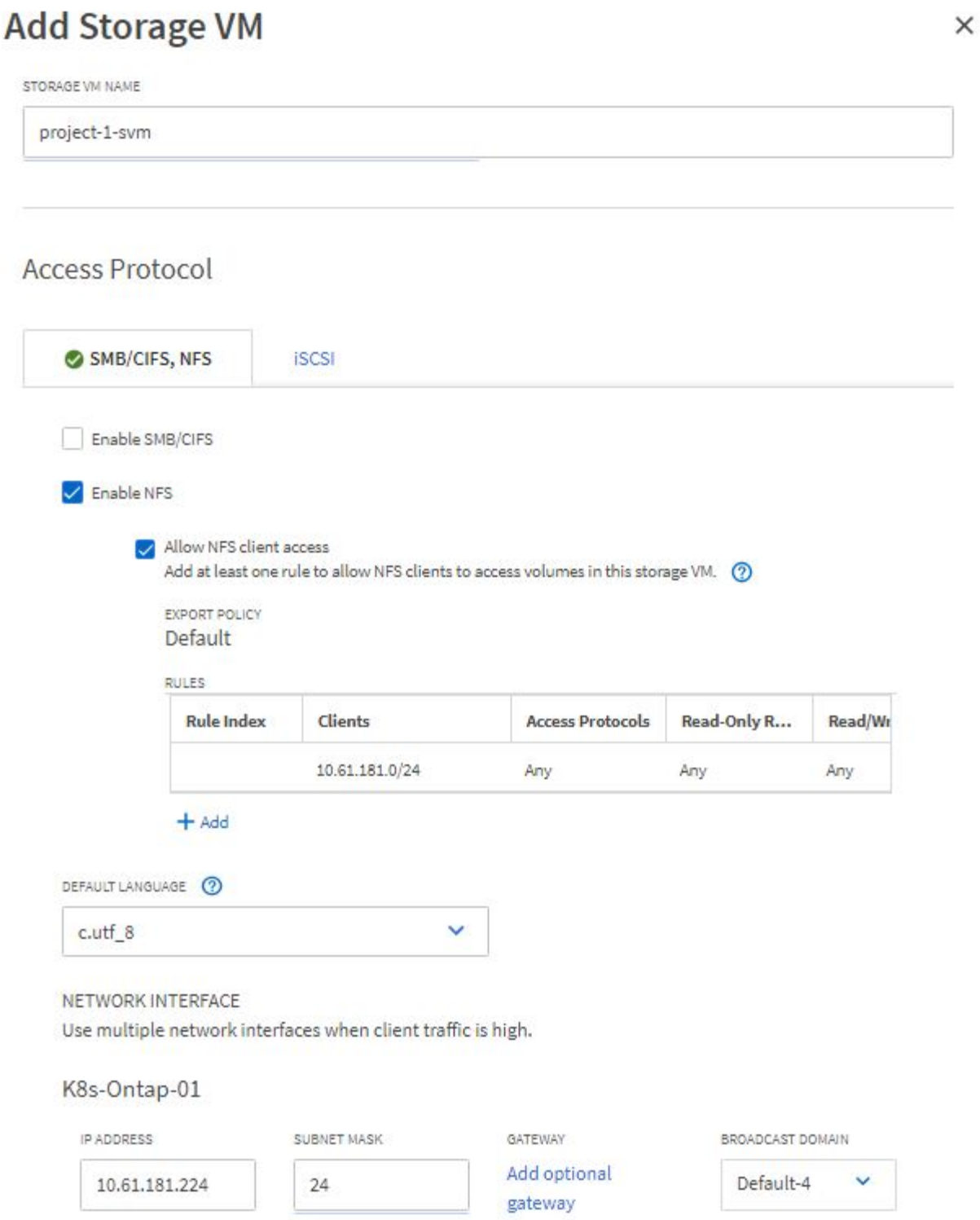
2. 同样，为开发人员创建 RoleBindings ，将开发人员角色绑定到 project-2 中的相应用户组。

"下一步：存储管理员任务。"

配置： storage-admin 任务

存储管理员必须配置以下资源：

1. 以管理员身份登录到 NetApp ONTAP 集群。
2. 导航到存储 > Storage VM ，然后单击添加。通过提供所需的详细信息，创建两个 SVM ，一个用于 project-1 ，另一个用于 project-2 。此外，还可以创建 vsadmin 帐户来管理 SVM 及其资源。



1. 以存储管理员身份登录到 Red Hat OpenShift 集群。
2. 为 project-1 创建后端，并将其映射到专用于该项目的 SVM 。NetApp 建议使用 SVM 的 vsadmin 帐户将后端连接到 SVM ，而不是使用 ONTAP 集群管理员。

```
cat << EOF | tridentctl -n trident create backend -f
{
    "version": 1,
    "storageDriverName": "ontap-nas",
    "backendName": "nfs_project_1",
    "managementLIF": "172.21.224.210",
    "dataLIF": "10.61.181.224",
    "svm": "project-1-svm",
    "username": "vsadmin",
    "password": "NetApp123"
}
EOF
```

在此示例中，我们使用的是 ontap-NAS 驱动程序。根据使用情形创建后端时，请使用相应的驱动程序。

我们假定 Trident 已安装在 Trident 项目中。

1. 同样，为 project-2 创建 Trident 后端，并将其映射到专用于 project-2 的 SVM 。
2. 接下来，创建存储类。为 project-1 创建存储类，并通过设置 storagePools 参数将其配置为使用后端专用于 project-1 的存储池。

```
cat << EOF | oc create -f -
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
  name: project-1-sc
provisioner: csi.trident.netapp.io
parameters:
  backendType: ontap-nas
  storagePools: "nfs_project_1:.*"
EOF
```

3. 同样，为 project-2 创建一个存储类，并将其配置为使用专用于 project-2 的后端存储池。
4. 创建 ResourceQuota 以限制 project-1 中的资源，从而从专用于其他项目的存储库请求存储。

```
cat << EOF | oc create -f -
kind: ResourceQuota
apiVersion: v1
metadata:
  name: project-1-sc-rq
  namespace: project-1
spec:
  hard:
    project-2-sc.storageclass.storage.k8s.io/persistentvolumeclaims: 0
EOF
```

5. 同样，也可以创建 ResourceQuota 来限制项目 2 中的资源，以便从专用于其他项目的存储库请求存储。

"下一步：验证。"