Requisitos

Kubernetes clusters

NetApp
April 16, 2024

Tabla de contenidos

Requisitos

Requisitos para clústeres de Kubernetes en AWS

Puede añadir clústeres gestionados de Amazon Elastic Kubernetes Service (EKS) o clústeres de Kubernetes autogestionados en AWS a BlueXP. Antes de poder añadir los clústeres a BlueXP, debe asegurarse de que se cumplen los siguientes requisitos.

En este tema se utiliza *Kubernetes cluster*, donde la configuración es la misma para EKS y clústeres de Kubernetes autogestionados. El tipo de clúster se especifica dónde difiere la configuración.

Requisitos

Astra Trident

Se requiere una de las cuatro versiones más recientes de Astra Trident. Puede instalar o actualizar Astra Trident directamente desde BlueXP. Usted debe "revise los requisitos previos" Antes de instalar Astra Trident.

Cloud Volumes ONTAP

Cloud Volumes ONTAP para AWS debe configurarse como almacenamiento back-end para el clúster. "Vaya a los documentos de Astra Trident para ver los pasos de configuración".

Conector BlueXP

Debe ejecutarse un conector en AWS con los permisos necesarios. Más información a continuación.

Conectividad de la red

Se requiere conectividad de red entre el clúster de Kubernetes y el conector y entre el clúster de Kubernetes y Cloud Volumes ONTAP. Más información a continuación.

Autorización de RBAC

Debe autorizarse el rol BlueXP Connector en cada clúster de Kubernetes. Más información a continuación.

Prepare un conector

Se requiere un conector BlueXP en AWS para detectar y gestionar clústeres de Kubernetes. Tendrá que crear un conector nuevo o utilizar un conector existente que tenga los permisos necesarios.

Cree un conector nuevo

Siga los pasos de uno de los siguientes enlaces.

- "Cree un conector desde BlueXP" (recomendado)
- "Cree un conector desde AWS Marketplace"
- "Instale el conector en un host Linux existente en AWS"

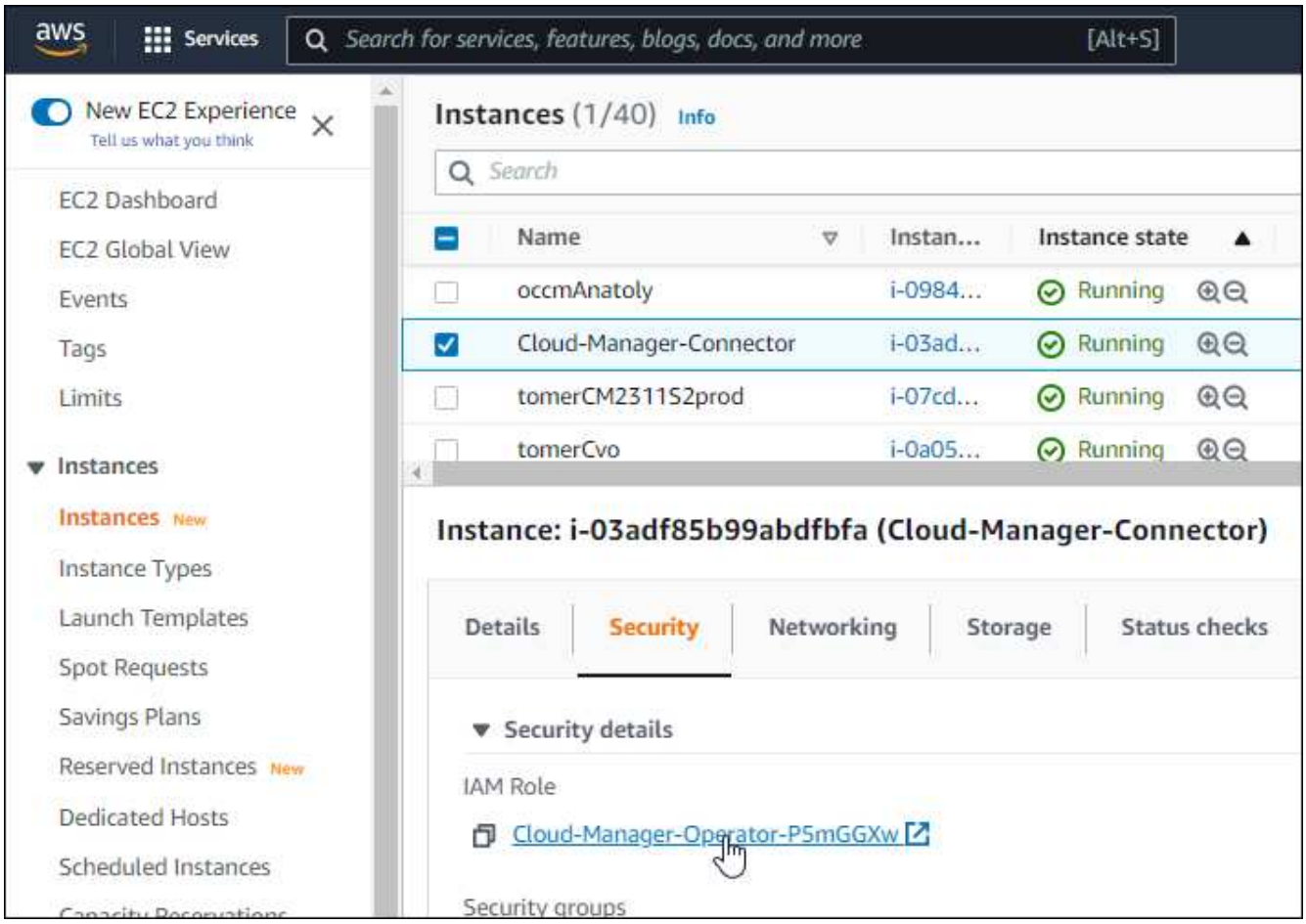
Agregue los permisos necesarios a un conector existente

A partir de la versión 3.9.13, todos los conectores _recién creados incluyen tres nuevos permisos de AWS que

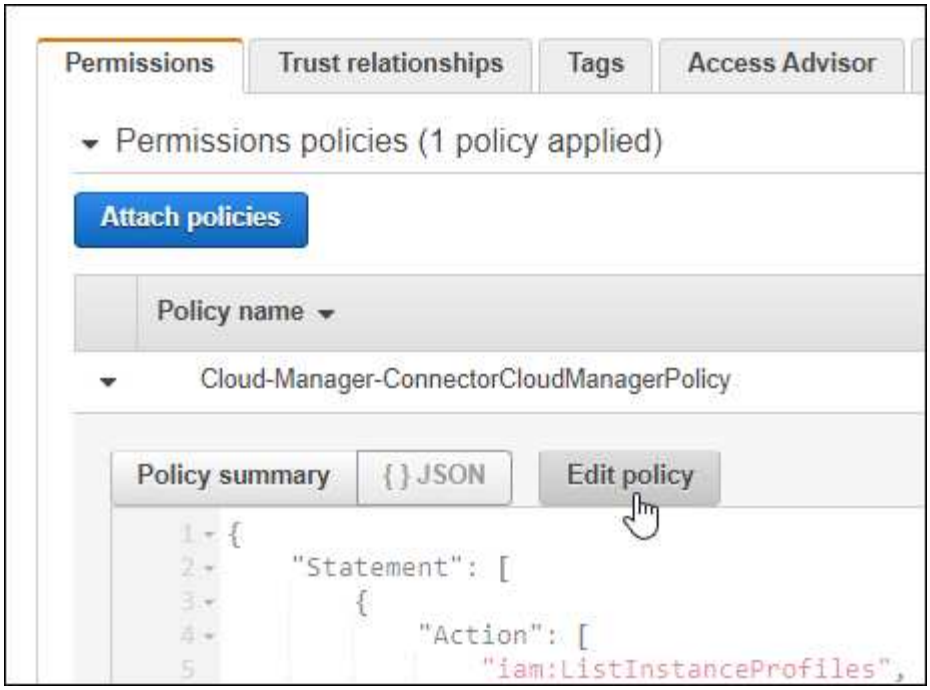
permiten la detección y la gestión de los clústeres de Kubernetes. Si ha creado un conector antes de esta versión, deberá modificar la directiva existente para el rol IAM del conector para proporcionar los permisos.

Pasos

1. Acceda a la consola de AWS y abra el servicio EC2.
2. Seleccione la instancia de conector, haga clic en **Seguridad** y haga clic en el nombre de la función IAM para ver el rol en el servicio IAM.



3. En la ficha **permisos**, expanda la directiva y haga clic en **Editar directiva**.



4. Haga clic en **JSON** y agregue los siguientes permisos en el primer conjunto de acciones:
   - ec2:regiones describidas
   - eks:ListClusters
   - eks:DescribeCluster
   - iam:GetInstanceProfile

   "Vea el formato JSON completo para la política"

5. Haga clic en **revisar directiva** y luego en **Guardar cambios**.

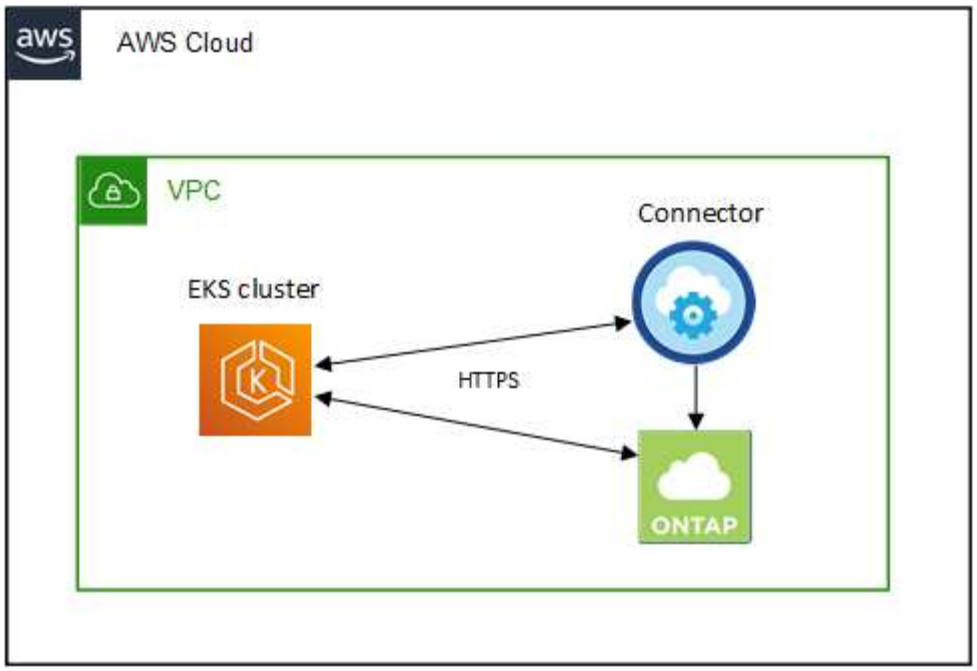
Revise los requisitos de red

Debe proporcionar conectividad de red entre el clúster de Kubernetes y el conector, y entre el clúster de Kubernetes y el sistema Cloud Volumes ONTAP que proporciona almacenamiento de back-end al clúster.

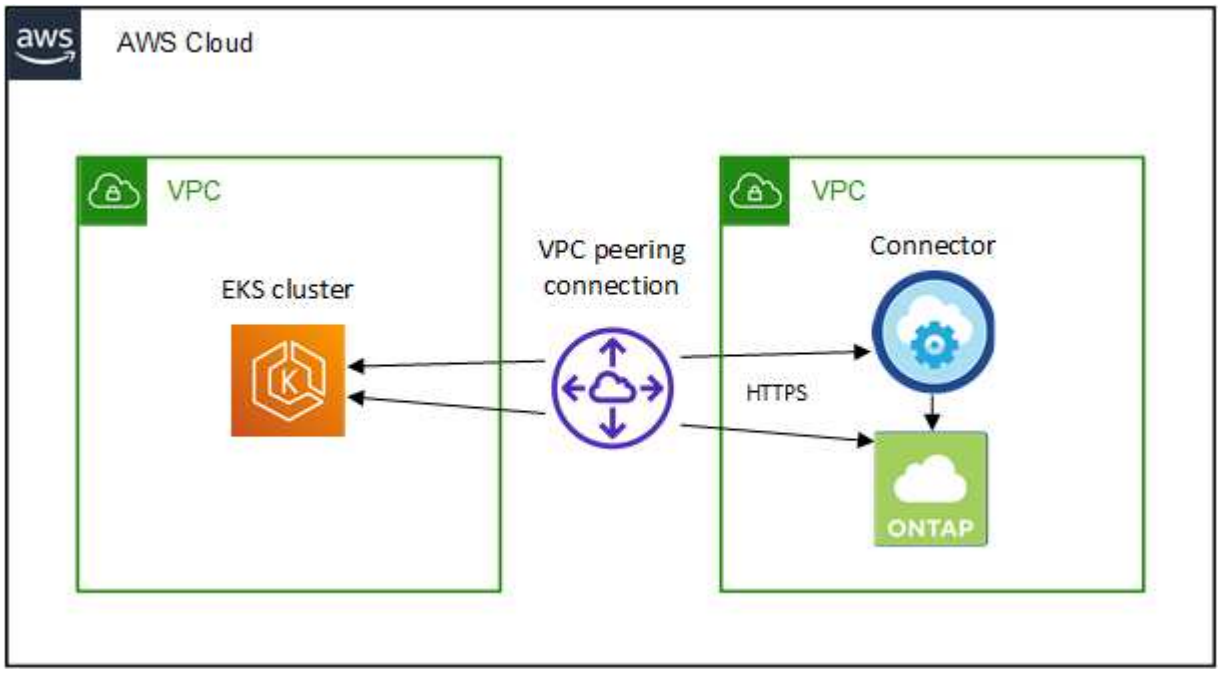
- Cada clúster de Kubernetes debe tener una conexión entrante desde el conector
- El conector debe tener una conexión de salida a cada clúster de Kubernetes a través del puerto 443

La forma más sencilla de proporcionar esta conectividad es poner en marcha el conector y Cloud Volumes ONTAP en el mismo VPC que el clúster de Kubernetes. De lo contrario, deberá configurar una conexión VPC peering entre los distintos VPC.

Aquí hay un ejemplo que muestra cada componente en el mismo VPC.



En este otro ejemplo se muestra un clúster de EKS que se ejecuta en un VPC diferente. En este ejemplo, VPC peering proporciona una conexión entre el VPC del clúster de EKS y el VPC del conector y Cloud Volumes ONTAP.



Configure la autorización de RBAC

Debe autorizar el rol de conector en cada clúster de Kubernetes para que el conector pueda detectar y gestionar un clúster.

Se requiere una autorización diferente para habilitar diferentes funciones.

Backup y restauración

El backup y la restauración solo necesitan una autorización básica.

Añada clases de almacenamiento

Se requiere una autorización ampliada para añadir clases de almacenamiento mediante BlueXP y supervisar el clúster en busca de cambios en el back-end.

Instale la trident

Debe proporcionar una autorización completa para que BlueXP instale Astra Trident.

> Cuando se instala Astra Trident, BlueXP instala el secreto de Kubernetes y back-end de Astra Trident que contiene las credenciales que Astra Trident necesita para comunicarse con el clúster de almacenamiento.

Pasos

1. Cree una función y un enlace de roles del clúster.
   a. Puede personalizar la autorización en función de sus requisitos.

Backup/restauración

Añada una autorización básica para habilitar el backup y la restauración para los clústeres de Kubernetes.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
        - ''
      resources:
        - namespaces
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumes
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - pods
        - pods/exec
      verbs:
        - get
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumeclaims
      verbs:
        - list
        - create
        - watch
    - apiGroups:
        - storage.k8s.io
      resources:
        - storageclasses
      verbs:
```

```
            - list
    - apiGroups:
            - trident.netapp.io
      resources:
            - tridentbackends
      verbs:
            - list
            - watch
    - apiGroups:
            - trident.netapp.io
      resources:
            - tridentorchestrators
      verbs:
            - get
            - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: Group
      name: cloudmanager-access-group
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

Clases de almacenamiento

Agregue autorización expandida para agregar clases de almacenamiento con BlueXP.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
            - ''
      resources:
            - secrets
            - namespaces
            - persistentvolumeclaims
            - persistentvolumes
            - pods
```

```
            - pods/exec
      verbs:
          - get
          - list
          - watch
          - create
          - delete
          - watch
    - apiGroups:
          - storage.k8s.io
      resources:
          - storageclasses
      verbs:
          - get
          - create
          - list
          - watch
          - delete
          - patch
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentbackends
          - tridentorchestrators
          - tridentbackendconfigs
      verbs:
          - get
          - list
          - watch
          - create
          - delete
          - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: Group
      name: cloudmanager-access-group
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

Instalación de Trident

Utilice la línea de comandos para proporcionar autorización completa y habilitar BlueXP para instalar Astra Trident.

```
eksctl create iamidentitymapping --cluster < > --region < > --arn
< > --group "system:masters" --username
system:node:{{EC2PrivateDNSName}}
```

b. Aplique la configuración a un clúster.

```
kubectl apply -f <file-name>
```

2. Cree una asignación de identidad al grupo de permisos.

Utilice eksctl

Utilice eksctl para crear una asignación de identidad IAM entre un clúster y la función IAM del conector BlueXP.

"Vaya a la documentación de eksctl para obtener instrucciones completas".

A continuación se muestra un ejemplo.

```
eksctl create iamidentitymapping --cluster <eksCluster> --region
<us-east-2> --arn <ARN of the Connector IAM role> --group
cloudmanager-access-group --username
system:node:{{EC2PrivateDNSName}}
```

Editar autenticación de aws

Edite directamente ConfigMap de AWS-auth para agregar acceso de RBAC a la función IAM para el conector BlueXP.

"Vaya a la documentación de AWS EKS para obtener instrucciones completas".

A continuación se muestra un ejemplo.

```
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - cloudmanager-access-group
      rolearn: <ARN of the Connector IAM role>
     username: system:node:{{EC2PrivateDNSName}}
kind: ConfigMap
metadata:
  creationTimestamp: "2021-09-30T21:09:18Z"
  name: aws-auth
  namespace: kube-system
  resourceVersion: "1021"
  selfLink: /api/v1/namespaces/kube-system/configmaps/aws-auth
  uid: dcc31de5-3838-11e8-af26-02e00430057c
```

Requisitos para clústeres de Kubernetes en Azure

Puede añadir y gestionar clústeres de Kubernetes de Azure (AKS) gestionados y clústeres de Kubernetes autogestionados en Azure usando BlueXP. Antes de poder añadir los clústeres a BlueXP, asegúrese de que se cumplan los siguientes requisitos.

En este tema se utiliza *Kubernetes cluster*, donde la configuración es la misma para AKS y clústeres de Kubernetes autogestionados. El tipo de clúster se especifica dónde difiere la configuración.

Requisitos

Astra Trident

Se requiere una de las cuatro versiones más recientes de Astra Trident. Puede instalar o actualizar Astra Trident directamente desde BlueXP. Usted debe "revise los requisitos previos" Antes de instalar Astra Trident.

Cloud Volumes ONTAP

Cloud Volumes ONTAP debe configurarse como almacenamiento back-end para el clúster. "Vaya a los documentos de Astra Trident para ver los pasos de configuración".

Conector BlueXP

Un conector debe ejecutarse en Azure con los permisos necesarios. Más información a continuación.

Conectividad de la red

Se requiere conectividad de red entre el clúster de Kubernetes y el conector y entre el clúster de Kubernetes y Cloud Volumes ONTAP. Más información a continuación.

Autorización de RBAC

BlueXP admite clústeres habilitados para RBAC con y sin Active Directory. Debe autorizarse el rol BlueXP Connector en cada clúster de Azure. Más información a continuación.

Prepare un conector

Se necesita un conector BlueXP en Azure para detectar y gestionar clústeres de Kubernetes. Tendrá que crear un conector nuevo o utilizar un conector existente que tenga los permisos necesarios.

Cree un conector nuevo

Siga los pasos de uno de los siguientes enlaces.

- "Cree un conector desde BlueXP" (recomendado)
- "Cree un conector desde Azure Marketplace"
- "Instale el conector en un host Linux existente"

Agregar los permisos necesarios a un conector existente (para detectar un clúster AKS gestionado)

Si desea detectar un clúster AKS gestionado, puede que necesite modificar la función personalizada para que Connector proporcione los permisos.

Pasos

1. Identifique la función asignada a la máquina virtual conector:
    a. En el portal de Azure, abra el servicio Virtual Machines.
    b. Seleccione la máquina virtual conector.
    c. En Configuración, seleccione **identidad**.

d. Haga clic en **asignaciones de roles de Azure**.

e. Anote la función personalizada asignada a la máquina virtual conector.

2. Actualice el rol personalizado:

a. En el portal de Azure, abra su suscripción a Azure.

b. Haga clic en **Control de acceso (IAM) > roles**.

c. Haga clic en los puntos suspensivos (…). Para la función personalizada y, a continuación, haga clic en **Editar**.

d. Haga clic en JSON y añada los siguientes permisos:

```
"Microsoft.ContainerService/managedClusters/listClusterUserCredential
/action"
"Microsoft.ContainerService/managedClusters/read"
```

e. Haga clic en **revisar + actualizar** y, a continuación, haga clic en **Actualizar**.

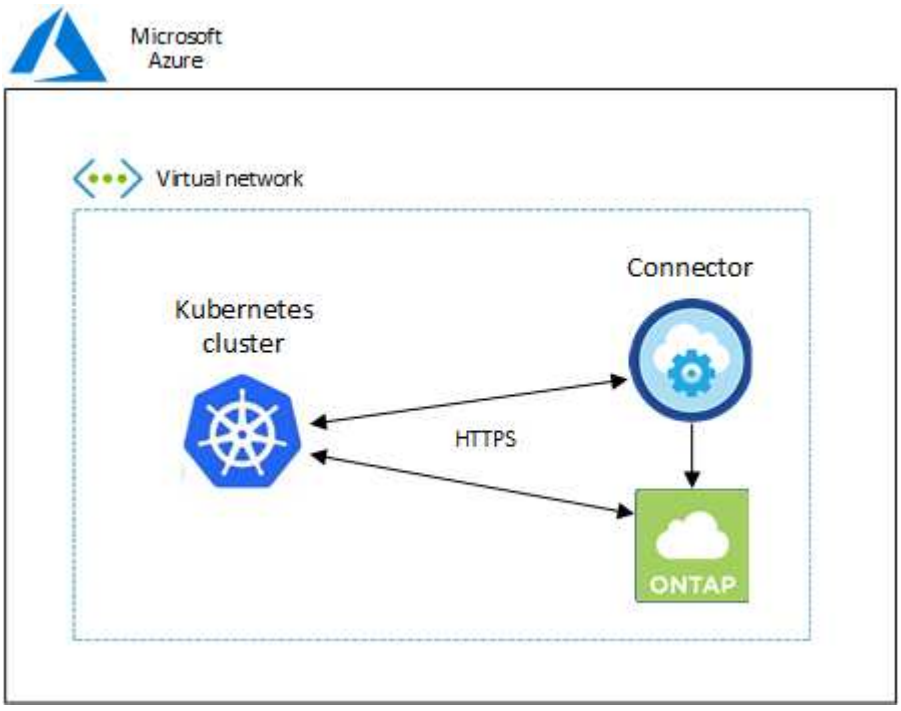
Revise los requisitos de red

Debe proporcionar conectividad de red entre el clúster de Kubernetes y el conector, y entre el clúster de Kubernetes y el sistema Cloud Volumes ONTAP que proporciona almacenamiento de back-end al clúster.

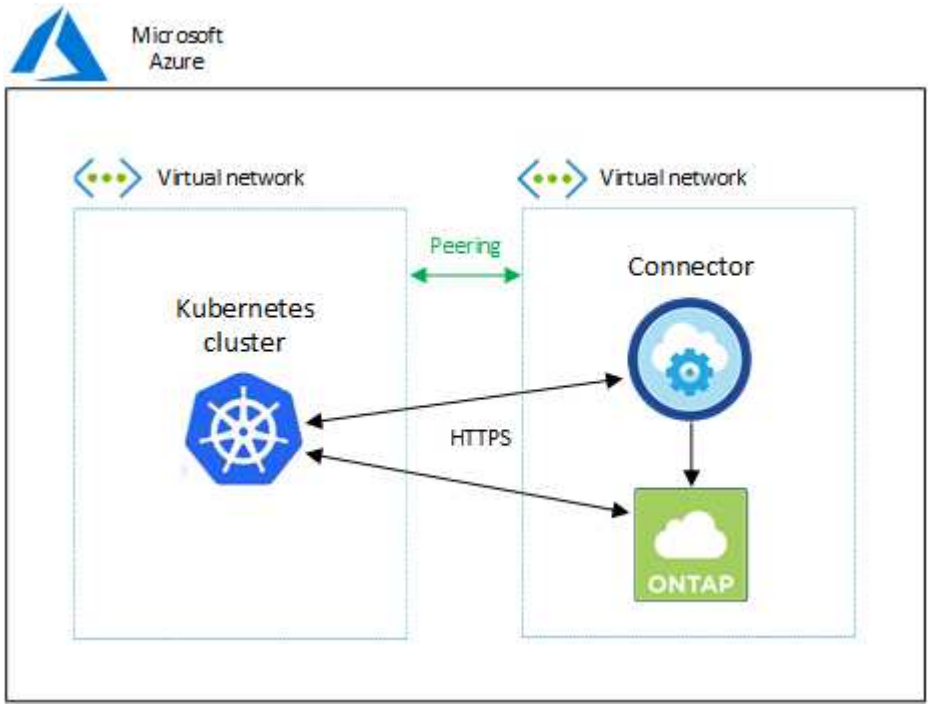
- Cada clúster de Kubernetes debe tener una conexión entrante desde el conector
- El conector debe tener una conexión de salida a cada clúster de Kubernetes a través del puerto 443

La forma más sencilla de proporcionar esta conectividad es poner en marcha el conector y Cloud Volumes ONTAP en la misma red que el clúster de Kubernetes. De lo contrario, debe configurar una conexión de interconexión entre los distintos VNets.

A continuación se muestra un ejemplo que muestra cada componente en el mismo vnet.



Y este es otro ejemplo que muestra un clúster Kubernetes que se ejecuta en un vnet diferente. En este ejemplo, peering proporciona una conexión entre el vnet del clúster de Kubernetes y el vnet del conector y Cloud Volumes ONTAP.



Configure la autorización de RBAC

La validación de RBAC solo se produce en clústeres de Kubernetes con Active Directory (AD) habilitado. Los clústeres de Kubernetes sin AD pasarán la validación automáticamente.

Es necesario autorizar el rol de conector en cada clúster de Kubernetes para que el conector pueda detectar y gestionar un clúster.

Backup y restauración

El backup y la restauración solo necesitan una autorización básica.

Añada clases de almacenamiento

Se requiere una autorización ampliada para añadir clases de almacenamiento mediante BlueXP y supervisar el clúster en busca de cambios en el back-end.

Instale la trident

Debe proporcionar una autorización completa para que BlueXP instale Astra Trident.

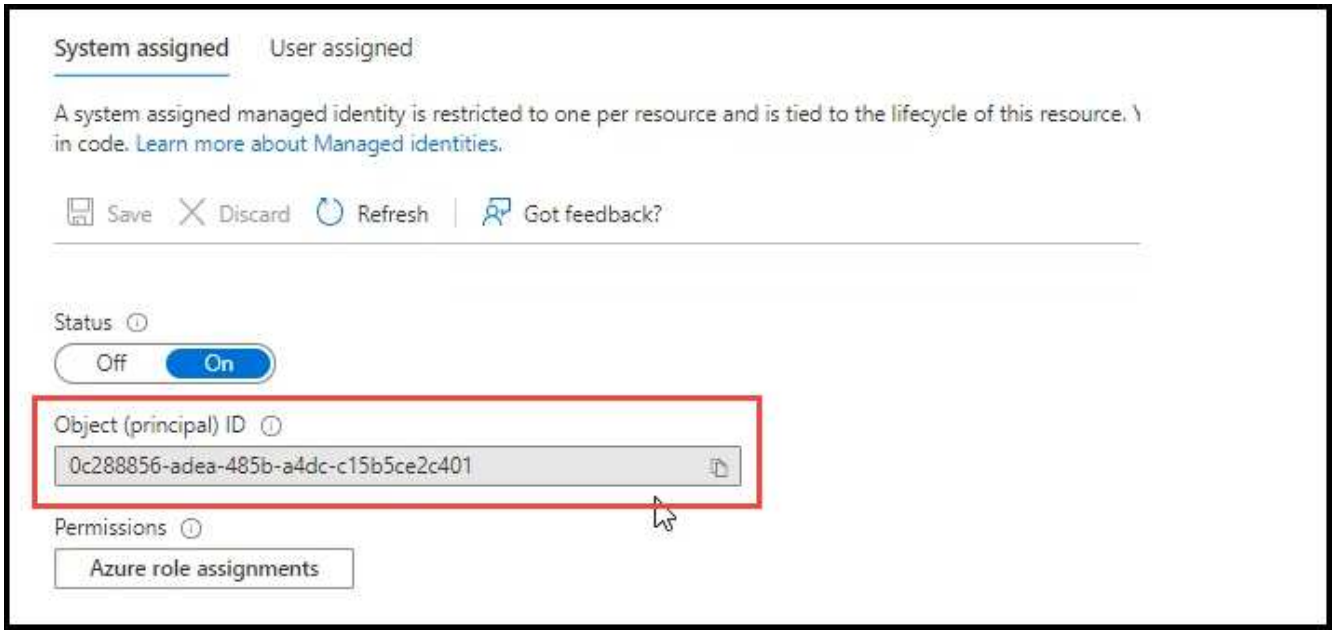
Cuando se instala Astra Trident, BlueXP instala el secreto de Kubernetes y back-end de Astra Trident que contiene las credenciales que Astra Trident necesita para comunicarse con el clúster de almacenamiento.

Antes de empezar

Su RBAC `subjects: name:` La configuración varía ligeramente según el tipo de clúster de Kubernetes.

- Si va a implementar un clúster **AKS gestionado**, necesita el identificador de objeto para la identidad administrada asignada por el sistema para el conector. Este ID está disponible en el portal de gestión de

Azure.



- Si va a implementar un **clúster Kubernetes autogestionado**, necesita el nombre de usuario de cualquier usuario autorizado.

Pasos

Cree una función y un enlace de roles del clúster.

1. Puede personalizar la autorización en función de sus requisitos.

Backup/restauración

Añada una autorización básica para habilitar el backup y la restauración para los clústeres de Kubernetes.

Sustituya el `subjects: kind:` variable con su nombre de usuario y. `subjects: name:` Con el ID de objeto para la identidad administrada asignada por el sistema o el nombre de usuario de cualquier usuario autorizado como se ha descrito anteriormente.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
        - ''
      resources:
        - namespaces
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumes
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - pods
        - pods/exec
      verbs:
        - get
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumeclaims
      verbs:
        - list
        - create
        - watch
    - apiGroups:
        - storage.k8s.io
```

```
      resources:
          - storageclasses
      verbs:
          - list
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentbackends
      verbs:
          - list
          - watch
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentorchestrators
      verbs:
          - get
          - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: User
      name:
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

Clases de almacenamiento

Agregue autorización expandida para agregar clases de almacenamiento con BlueXP.

Sustituya el `subjects: kind:` variable con su nombre de usuario y. `subjects: user:` Con el ID de objeto para la identidad administrada asignada por el sistema o el nombre de usuario de cualquier usuario autorizado como se ha descrito anteriormente.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
          - ''
```

```
      resources:
          - secrets
          - namespaces
          - persistentvolumeclaims
          - persistentvolumes
          - pods
          - pods/exec
      verbs:
          - get
          - list
          - watch
          - create
          - delete
          - watch
    - apiGroups:
          - storage.k8s.io
      resources:
          - storageclasses
      verbs:
          - get
          - create
          - list
          - watch
          - delete
          - patch
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentbackends
          - tridentorchestrators
          - tridentbackendconfigs
      verbs:
          - get
          - list
          - watch
          - create
          - delete
          - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: User
      name:
```

```
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

Instalación de Trident

Utilice la línea de comandos para proporcionar autorización completa y habilitar BlueXP para instalar Astra Trident.

```
eksctl create iamidentitymapping --cluster < > --region < > --arn <
> --group "system:masters" --username
system:node:{{EC2PrivateDNSName}}
```

2. Aplique la configuración a un clúster.

```
kubectl apply -f <file-name>
```

Requisitos para los clústeres de Kubernetes en Google Cloud

Puede añadir y gestionar clústeres gestionados de Google Kubernetes Engine (GKE) y clústeres de Kubernetes autogestionados en Google mediante BlueXP. Antes de poder añadir los clústeres a BlueXP, asegúrese de que se cumplan los siguientes requisitos.

En este tema se utiliza *Kubernetes cluster*, donde la configuración es la misma para los clústeres GKE y Kubernetes autogestionados. El tipo de clúster se especifica dónde difiere la configuración.

Requisitos

Astra Trident

Se requiere una de las cuatro versiones más recientes de Astra Trident. Puede instalar o actualizar Astra Trident directamente desde BlueXP. Usted debe "revise los requisitos previos" Antes de instalar Astra Trident

Cloud Volumes ONTAP

Cloud Volumes ONTAP debe estar en BlueXP con la misma cuenta de seguridad, espacio de trabajo y conector que el clúster de Kubernetes. "Vaya a los documentos de Astra Trident para ver los pasos de configuración".

Conector BlueXP

Un conector debe estar en ejecución en Google con los permisos necesarios. Más información a continuación.

Conectividad de la red

Se requiere conectividad de red entre el clúster de Kubernetes y el conector y entre el clúster de Kubernetes y Cloud Volumes ONTAP. Más información a continuación.

Autorización de RBAC

BlueXP admite clústeres habilitados para RBAC con y sin Active Directory. La función conector BlueXP debe estar autorizada en cada clúster GKE. Más información a continuación.

Prepare un conector

Se necesita un conector BlueXP en Google para detectar y gestionar clústeres de Kubernetes. Tendrá que crear un conector nuevo o utilizar un conector existente que tenga los permisos necesarios.

Cree un conector nuevo

Siga los pasos de uno de los siguientes enlaces.

- "Cree un conector desde BlueXP" (recomendado)
- "Instale el conector en un host Linux existente"

Agregar los permisos necesarios a un conector existente (para detectar un clúster GKE administrado)

Si desea detectar un clúster GKE administrado, es posible que deba modificar la función personalizada para que Connector proporcione los permisos.

Pasos

1. Pulg "Consola de cloud", Vaya a la página **roles**.
2. Mediante la lista desplegable situada en la parte superior de la página, seleccione el proyecto o la organización que contiene la función que desea editar.
3. Haga clic en una función personalizada.
4. Haga clic en **Editar rol** para actualizar los permisos del rol.
5. Haga clic en **Agregar permisos** para agregar los siguientes permisos nuevos a la función.

```
container.clusters.get
container.clusters.list
```

6. Haga clic en **Actualizar** para guardar la función editada.

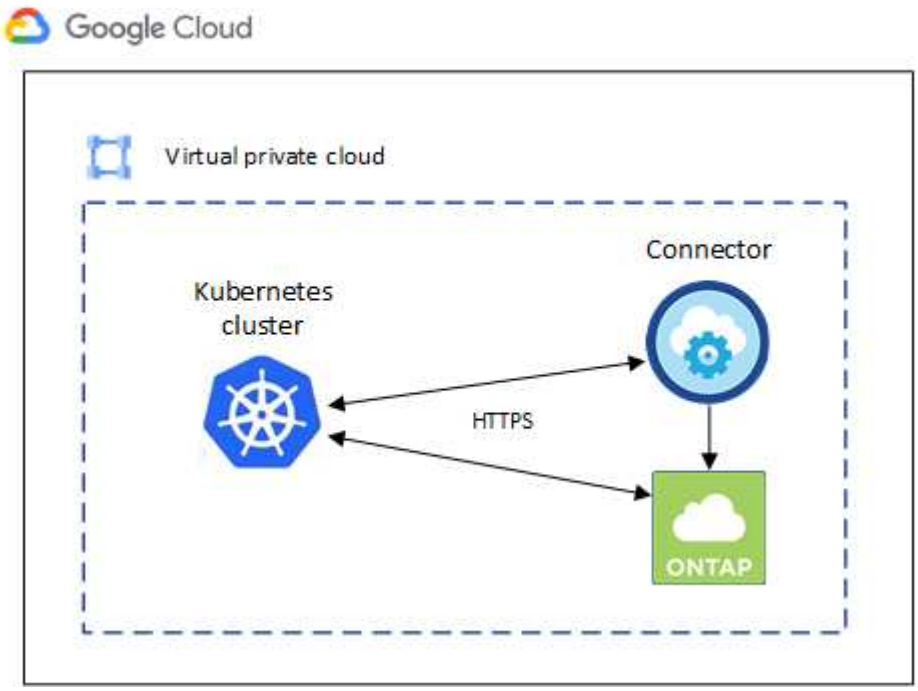
Revise los requisitos de red

Debe proporcionar conectividad de red entre el clúster de Kubernetes y el conector, y entre el clúster de Kubernetes y el sistema Cloud Volumes ONTAP que proporciona almacenamiento de back-end al clúster.

- Cada clúster de Kubernetes debe tener una conexión entrante desde el conector
- El conector debe tener una conexión de salida a cada clúster de Kubernetes a través del puerto 443

La forma más sencilla de proporcionar esta conectividad es poner en marcha el conector y Cloud Volumes ONTAP en el mismo VPC que el clúster de Kubernetes. De lo contrario, deberá configurar una conexión entre iguales entre el VPC diferente.

Aquí hay un ejemplo que muestra cada componente en el mismo VPC.



Configure la autorización de RBAC

La validación de RBAC solo se produce en clústeres de Kubernetes con Active Directory (AD) habilitado. Los clústeres de Kubernetes sin AD pasarán la validación automáticamente.

Es necesario autorizar el rol de conector en cada clúster de Kubernetes para que el conector pueda detectar y gestionar un clúster.

Backup y restauración

El backup y la restauración solo necesitan una autorización básica.

Añada clases de almacenamiento

Se requiere una autorización ampliada para añadir clases de almacenamiento mediante BlueXP y supervisar el clúster en busca de cambios en el back-end.

Instale la trident

Debe proporcionar una autorización completa para que BlueXP instale Astra Trident.

Cuando se instala Astra Trident, BlueXP instala el secreto de Kubernetes y back-end de Astra Trident que contiene las credenciales que Astra Trident necesita para comunicarse con el clúster de almacenamiento.

Antes de empezar

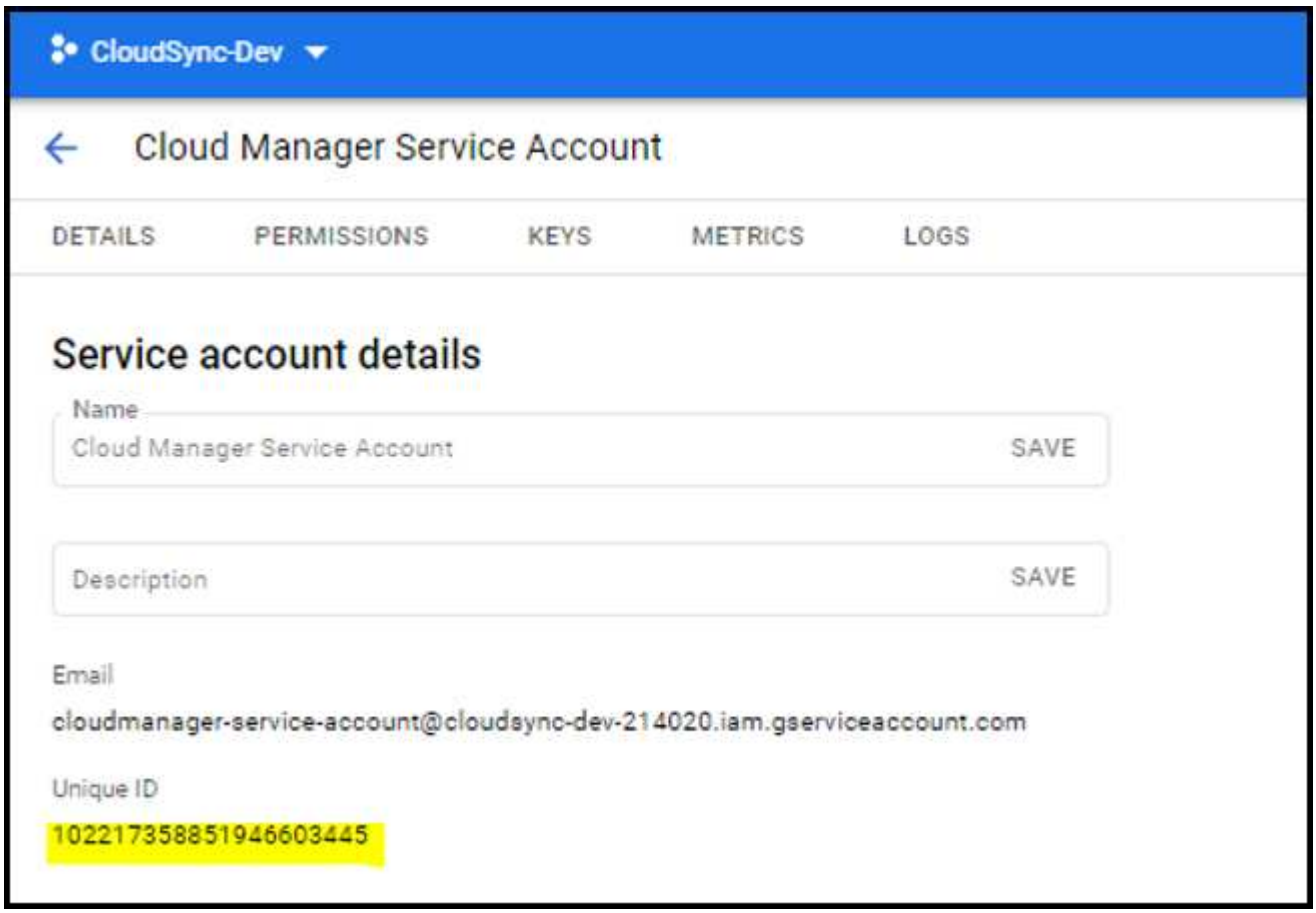
Para configurar `subjects: name:` En el archivo YAML, debe conocer el ID único de BlueXP.

Puede encontrar el ID único de dos maneras:

- Con el comando:

```
gcloud iam service-accounts list
gcloud iam service-accounts describe <service-account-email>
```

- En Detalles de la cuenta de servicio en la "Consola de cloud".



Pasos

Cree una función y un enlace de roles del clúster.

1. Puede personalizar la autorización en función de sus requisitos.

Backup/restauración

Añada una autorización básica para habilitar el backup y la restauración para los clústeres de Kubernetes.

Sustituya el `subjects: kind:` variable con su nombre de usuario y. `subjects: name:` Con el ID exclusivo para la cuenta de servicio autorizada.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
        - ''
      resources:
        - namespaces
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumes
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - pods
        - pods/exec
      verbs:
        - get
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumeclaims
      verbs:
        - list
        - create
        - watch
    - apiGroups:
        - storage.k8s.io
```

```
      resources:
          - storageclasses
      verbs:
          - list
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentbackends
      verbs:
          - list
          - watch
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentorchestrators
      verbs:
          - get
          - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: User
      name:
      apiGroup: rbac.authorization.k8s.io
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

Clases de almacenamiento

Agregue autorización expandida para agregar clases de almacenamiento con BlueXP.

Sustituya el `subjects: kind:` variable con su nombre de usuario y. `subjects: user:` Con el ID exclusivo para la cuenta de servicio autorizada.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
          - ''
      resources:
```

```
            - secrets
            - namespaces
            - persistentvolumeclaims
            - persistentvolumes
            - pods
            - pods/exec
        verbs:
            - get
            - list
            - watch
            - create
            - delete
            - watch
    - apiGroups:
            - storage.k8s.io
        resources:
            - storageclasses
        verbs:
            - get
            - create
            - list
            - watch
            - delete
            - patch
    - apiGroups:
            - trident.netapp.io
        resources:
            - tridentbackends
            - tridentorchestrators
            - tridentbackendconfigs
        verbs:
            - get
            - list
            - watch
            - create
            - delete
            - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
subjects:
    - kind: User
      name:
      apiGroup: rbac.authorization.k8s.io
```

```
roleRef:
    kind: ClusterRole
    name: cloudmanager-access-clusterrole
    apiGroup: rbac.authorization.k8s.io
```

Instalación de Trident

Utilice la línea de comandos para proporcionar autorización completa y habilitar BlueXP para instalar Astra Trident.

```
kubectl create clusterrolebinding test --clusterrole cluster-admin
--user <Unique ID>
```

2. Aplique la configuración a un clúster.

```
kubectl apply -f <file-name>
```

Requisitos para clústeres de Kubernetes en OpenShift

Puede añadir y gestionar clústeres de Kubernetes de OpenShift autogestionados mediante BlueXP. Antes de poder añadir los clústeres a BlueXP, asegúrese de que se cumplan los siguientes requisitos.

Requisitos

Astra Trident

Se requiere una de las cuatro versiones más recientes de Astra Trident. Puede instalar o actualizar Astra Trident directamente desde BlueXP. Usted debe "revise los requisitos previos" Antes de instalar Astra Trident.

Cloud Volumes ONTAP

Cloud Volumes ONTAP debe configurarse como almacenamiento back-end para el clúster. "Vaya a los documentos de Astra Trident para ver los pasos de configuración".

Conector BlueXP

Se necesita un conector BlueXP para importar y gestionar clústeres de Kubernetes. Tendrá que crear un nuevo conector o utilizar un conector existente que tenga los permisos necesarios para su proveedor de cloud:

- "Conector AWS"
- "Conector de Azure"
- "Conector de Google Cloud"

Conectividad de la red

Se requiere conectividad de red entre el clúster de Kubernetes y el conector y entre el clúster de

Kubernetes y Cloud Volumes ONTAP.

Archivo de configuración de Kubernetes (kubeconfig) con autorización de RBAC

Para importar clústeres OpenShift, necesita un archivo kubeconfig con la autorización RBAC necesaria para habilitar diferentes funcionalidades. Cree un archivo kubeconfig.

- Backup y restauración: El backup y la restauración solo requieren una autorización básica.
- Añadir clases de almacenamiento: Se requiere una autorización ampliada para agregar clases de almacenamiento con BlueXP y supervisar el clúster para realizar cambios en el back-end.
- Instalar Astra Trident: Necesita proporcionar una autorización completa para que BlueXP instale Astra Trident.

> Cuando se instala Astra Trident, BlueXP instala el secreto de Kubernetes y back-end de Astra Trident que contiene las credenciales que Astra Trident necesita para comunicarse con el clúster de almacenamiento.

Cree un archivo kubeconfig

Con la CLI de OpenShift, cree un archivo kubeconfig para importarlo a BlueXP.

Pasos

1. Inicie sesión en la CLI de OpenShift mediante `oc login` En una URL pública con un usuario administrativo.
2. Cree una cuenta de servicio del siguiente modo:
   a. Cree un archivo de cuenta de servicio llamado `oc-service-account.yaml`.

      Ajuste el nombre y el espacio de nombres según sea necesario. Si se realizan cambios aquí, debe aplicar los mismos cambios en los pasos siguientes.

```
oc-service-account.yaml
```

+

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: oc-service-account
  namespace: default
```

   a. Aplicar la cuenta de servicio:

```
kubectl apply -f oc-service-account.yaml
```

3. Cree un enlace de funciones personalizado en función de sus requisitos de autorización.

a. Cree un `ClusterRoleBinding` archivo llamado `oc-clusterrolebinding.yaml`.

```
oc-clusterrolebinding.yaml
```

b. Configure la autorización de RBAC según sea necesario para el clúster.

Backup/restauración

Añada una autorización básica para habilitar el backup y la restauración para los clústeres de Kubernetes.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
        - ''
      resources:
        - namespaces
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumes
      verbs:
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - pods
        - pods/exec
      verbs:
        - get
        - list
        - watch
    - apiGroups:
        - ''
      resources:
        - persistentvolumeclaims
      verbs:
        - list
        - create
        - watch
    - apiGroups:
        - storage.k8s.io
      resources:
        - storageclasses
      verbs:
```

```
            - list
    - apiGroups:
            - trident.netapp.io
      resources:
            - tridentbackends
      verbs:
            - list
            - watch
    - apiGroups:
            - trident.netapp.io
      resources:
            - tridentorchestrators
      verbs:
            - get
            - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cloudmanager-access-clusterrole
subjects:
    - kind: ServiceAccount
      name: oc-service-account
      namespace: default
```

Clases de almacenamiento

Agregue autorización expandida para agregar clases de almacenamiento con BlueXP.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
    name: cloudmanager-access-clusterrole
rules:
    - apiGroups:
            - ''
      resources:
            - secrets
            - namespaces
            - persistentvolumeclaims
            - persistentvolumes
            - pods
```

```
          - pods/exec
      verbs:
          - get
          - list
          - watch
          - create
          - delete
          - watch
    - apiGroups:
          - storage.k8s.io
      resources:
          - storageclasses
      verbs:
          - get
          - create
          - list
          - watch
          - delete
          - patch
    - apiGroups:
          - trident.netapp.io
      resources:
          - tridentbackends
          - tridentorchestrators
          - tridentbackendconfigs
      verbs:
          - get
          - list
          - watch
          - create
          - delete
          - watch
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
    name: k8s-access-binding
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cloudmanager-access-clusterrole
subjects:
    - kind: ServiceAccount
      name: oc-service-account
      namespace: default
```

Instalación de Trident

Conceda la autorización completa de administración y habilite BlueXP para instalar Astra Trident.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: cloudmanager-access-clusterrole
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: cluster-admin
subjects:
- kind: ServiceAccount
  name: oc-service-account
  namespace: default
```

c. Aplique el enlace de roles del clúster:

```
kubectl apply -f oc-clusterrolebinding.yaml
```

4. Enumere los secretos de la cuenta de servicio, reemplazando `<context>` con el contexto correcto para su instalación:

```
kubectl get serviceaccount oc-service-account --context <context>
--namespace default -o json
```

El final de la salida debe ser similar a lo siguiente:

```
"secrets": [
{ "name": "oc-service-account-dockercfg-vhz87"},
{ "name": "oc-service-account-token-r59kr"}
]
```

Los índices de cada elemento de la `secrets` la matriz comienza con 0. En el ejemplo anterior, el índice para `oc-service-account-dockercfg-vhz87` sería 0 y el índice para `oc-service-account-token-r59kr` sería 1. En la salida, anote el índice del nombre de la cuenta de servicio que contiene la palabra "token".

5. Genere la kubeconfig de la siguiente manera:

a. Cree un `create-kubeconfig.sh` archivo. Sustituya `TOKEN_INDEX` al principio de la secuencia de comandos siguiente con el valor correcto.

create-kubeconfig.sh

```
# Update these to match your environment.
# Replace TOKEN_INDEX with the correct value
# from the output in the previous step. If you
# didn't change anything else above, don't change
# anything else here.

SERVICE_ACCOUNT_NAME=oc-service-account
NAMESPACE=default
NEW_CONTEXT=oc
KUBECONFIG_FILE='kubeconfig-sa'

CONTEXT=$(kubectl config current-context)

SECRET_NAME=$(kubectl get serviceaccount ${SERVICE_ACCOUNT_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.secrets[TOKEN_INDEX].name}')
TOKEN_DATA=$(kubectl get secret ${SECRET_NAME} \
  --context ${CONTEXT} \
  --namespace ${NAMESPACE} \
  -o jsonpath='{.data.token}')

TOKEN=$(echo ${TOKEN_DATA} | base64 -d)

# Create dedicated kubeconfig
# Create a full copy
kubectl config view --raw > ${KUBECONFIG_FILE}.full.tmp

# Switch working context to correct context
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp config use-context
${CONTEXT}

# Minify
kubectl --kubeconfig ${KUBECONFIG_FILE}.full.tmp \
  config view --flatten --minify > ${KUBECONFIG_FILE}.tmp

# Rename context
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  rename-context ${CONTEXT} ${NEW_CONTEXT}

# Create token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
```

```
  set-credentials ${CONTEXT}-${NAMESPACE}-token-user \
  --token ${TOKEN}

# Set context to use token user
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --user ${CONTEXT}-${NAMESPACE}-token
-user

# Set context to correct namespace
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  set-context ${NEW_CONTEXT} --namespace ${NAMESPACE}

# Flatten/minify kubeconfig
kubectl config --kubeconfig ${KUBECONFIG_FILE}.tmp \
  view --flatten --minify > ${KUBECONFIG_FILE}

# Remove tmp
rm ${KUBECONFIG_FILE}.full.tmp
rm ${KUBECONFIG_FILE}.tmp
```

b. Origen de los comandos para aplicarlos al clúster de Kubernetes.

```
source create-kubeconfig.sh
```

Resultado

Utilizará el resultado `kubeconfig-sa` Archivo para agregar un clúster OpenShift a BlueXP.