

Tutorial: cómo crear un montaje de volumen Azure Files en Azure Container Apps

En este tutorial, aprenderá a:

- Creación de un entorno de Container Apps
- Creación de una cuenta de Azure Storage
- Definición de un recurso compartido de archivos en la cuenta de almacenamiento
- Vinculación del entorno al recurso compartido de archivos de almacenamiento
- Montaje del recurso compartido de almacenamiento en un contenedor individual
- Compruebe el montaje de almacenamiento mediante la visualización del registro de acceso al sitio web

Configuración del entorno

Los siguientes comandos definen variables de entorno y garantizan que la extensión Container Apps esté actualizada.

1. Inicie sesión en la CLI de Azure.

`az login`

2. Configure las variables de entorno que se usan en varios comandos que se deben seguir.

`RESOURCE_GROUP="my-container-apps-group"`

`ENVIRONMENT_NAME="my-storage-environment"`

`LOCATION="canadacentral"`

3. Asegúrese de que tiene la versión más reciente de la extensión de la CLI de Azure Container Apps.

`az extension add -n containerapp --upgrade`

4. Registre el espacio de nombres Microsoft.App.

`az provider register --namespace Microsoft.App`

5. Registre el proveedor de Microsoft.Operationallnsights para el área de trabajo de Log Analytics de Azure Monitor si no lo ha usado antes.

`az provider register --namespace Microsoft.Operationallnsights`

Creación de un entorno

En los pasos siguientes se crea un grupo de recursos y un entorno de Container Apps.

1. Cree un grupo de recursos.

`az group create \`

`--name $RESOURCE_GROUP \`

`--location $LOCATION \`

```
--query "properties.provisioningState"
```

Una vez creado, el comando devuelve un mensaje "Correcto".

Al final de este tutorial, puede eliminar el grupo de recursos para quitar todos los servicios creados durante este artículo.

2. Creación de un entorno de Container Apps.

```
az containerapp env create \  
--name $ENVIRONMENT_NAME \  
--resource-group $RESOURCE_GROUP \  
--location "$LOCATION" \  
--query "properties.provisioningState"
```

Una vez creado, el comando devuelve un mensaje "Correcto".

Los montajes de almacenamiento están asociados a un entorno de Container Apps y se configuran dentro de aplicaciones de contenedor individuales.

Configuración de una cuenta de almacenamiento

A continuación, cree una cuenta de almacenamiento y establezca un recurso compartido de archivos para montarlo en la aplicación contenedora.

1. Defina un nombre de cuenta de almacenamiento.

Este comando genera un sufijo aleatorio en el nombre de la cuenta de almacenamiento para garantizar la unicidad.

```
STORAGE_ACCOUNT_NAME="myacastorageaccount$RANDOM"
```

2. Cree una cuenta de Azure Storage.

```
az storage account create \  
--resource-group $RESOURCE_GROUP \  
--name $STORAGE_ACCOUNT_NAME \  
--location "$LOCATION" \  
--kind StorageV2 \  
--sku Standard_LRS \  
--enable-large-file-share \  
--query provisioningState
```

Una vez creado, el comando devuelve un mensaje "Correcto".

3. Definir un nombre de recurso compartido de archivos.

```
STORAGE_SHARE_NAME="myfileshare"
```

4. Cree el recurso compartido de archivos de Azure Storage.

```
az storage share-rm create \  
  
--resource-group $RESOURCE_GROUP \  
  
--storage-account $STORAGE_ACCOUNT_NAME \  
  
--name $STORAGE_SHARE_NAME \  
  
--quota 1024 \  
  
--enabled-protocols SMB \  
  
--output table
```

5. Obtenga la clave de la cuenta de almacenamiento.

```
STORAGE_ACCOUNT_KEY=`az storage account keys list -n $STORAGE_ACCOUNT_NAME  
--query "[0].value" -o tsv`
```

La clave de la cuenta de almacenamiento es necesaria para crear el vínculo de almacenamiento en el entorno de Container Apps. Container Apps no admite el acceso basado en identidades a recursos compartidos de archivos de Azure.

6. Defina el nombre del punto de montaje de almacenamiento.

```
STORAGE_MOUNT_NAME="mystoragemount"
```

Este valor es el nombre que se usa para definir el vínculo de montaje de almacenamiento desde el entorno de Container Apps a la cuenta de Azure Storage.

Crear montaje de almacenamiento

Ahora puede actualizar la configuración de la aplicación contenedora para admitir el montaje de almacenamiento.

1. Cree el vínculo de almacenamiento en el entorno.

```
az containerapp env storage set \  
  
--access-mode ReadWrite \  
  
--azure-file-account-name $STORAGE_ACCOUNT_NAME \  
  
--azure-file-account-key $STORAGE_ACCOUNT_KEY \  
  
--azure-file-share-name $STORAGE_SHARE_NAME \  
  
--storage-name $STORAGE_MOUNT_NAME \  
  
--name $ENVIRONMENT_NAME \  
  
--resource-group $RESOURCE_GROUP \  
  
--output table
```

Este comando crea un vínculo entre el entorno de la aplicación contenedora y el recurso compartido de archivos creado con el `az storage share-rm` comando .

Ahora que la cuenta de almacenamiento y el entorno están vinculados, puede crear una aplicación de contenedor que use el montaje de almacenamiento.

2. Defina el nombre de la aplicación contenedora.

```
CONTAINER_APP_NAME="my-container-app"
```

3. Cree la aplicación de contenedor.

```
az containerapp create \  
  --name $CONTAINER_APP_NAME \  
  --resource-group $RESOURCE_GROUP \  
  --environment $ENVIRONMENT_NAME \  
  --image nginx \  
  --min-replicas 1 \  
  --max-replicas 1 \  
  --target-port 80 \  
  --ingress external \  
  --query properties.configuration.ingress.fqdn
```

Este comando muestra la dirección URL de la nueva aplicación contenedora.

4. Copie la dirección URL y péguela en el explorador web para ir al sitio web.

Una vez que se cargue la página, verá el mensaje "Bienvenido a nginx!". Mantén abierta esta pestaña del explorador. Volverá al sitio web durante los pasos de comprobación del montaje de almacenamiento.

Ahora que ha confirmado que la aplicación contenedora está configurada, puede actualizar la aplicación con una definición de montaje de almacenamiento.

5. Exporte la configuración de la aplicación contenedora.

```
az containerapp show \  
  --name $CONTAINER_APP_NAME \  
  --resource-group $RESOURCE_GROUP \  
  --output yaml > app.yaml
```

Nota

Aunque esta aplicación no tiene secretos, muchas aplicaciones sí incluyen secretos. De forma predeterminada, al exportar la configuración de una aplicación, los valores de los secretos no se incluyen en el YAML generado.

Si no necesita cambiar los valores secretos, puede quitar la `secrets` sección y los secretos permanecen sin modificar. Como alternativa, si necesita cambiar el valor de un secreto, asegúrese de proporcionar el `name` y el `value` para todos los secretos del archivo antes de

intentar actualizar la aplicación. Si se omite un secreto de la secrets sección, se elimina el secreto.

6. Abra *app.yaml* en un editor de código.
7. Reemplace la `volumes: null` definición de la template sección por una `volumes:` definición que haga referencia al volumen de almacenamiento. La sección de plantilla debe ser similar a la siguiente:

template:

containers:

- image: nginx

imageType: ContainerImage

name: my-container-app

resources:

cpu: 0.5

ephemeralStorage: 2Gi

memory: 1Gi

volumeMounts:

- volumeName: my-azure-file-volume

mountPath: /var/log/nginx

initContainers: null

revisionSuffix: "

scale:

cooldownPeriod: 300

maxReplicas: 1

minReplicas: 1

pollingInterval: 30

rules: null

serviceBinds: null

terminationGracePeriodSeconds: null

volumes:

- name: my-azure-file-volume

storageName: mystoragemount

storageType: AzureFile

La nueva template.volumes sección incluye las siguientes propiedades.

Propiedad	Descripción
name	Este valor coincide con el volumen creado llamando al az containerapp env storage set comando .
storageName	Este valor define el nombre que usan los contenedores del entorno para acceder al volumen de almacenamiento.
storageType	Este valor determina el tipo de volumen de almacenamiento definido para el entorno. En este caso, se declara un montaje de Azure Files.

La sección volumes define volúmenes a nivel de aplicación a los que el contenedor de aplicaciones o los contenedores sidecar pueden hacer referencia a través de una sección volumeMounts asociada a un contenedor.

8. Añada una sección volumeMounts al contenedor nginx en la sección containers.

containers:

- image: nginx

name: my-container-app

volumeMounts:

- volumeName: my-azure-file-volume

mountPath: /var/log/nginx

La nueva volumeMounts sección incluye las siguientes propiedades:

Propiedad	Descripción
volumeName	Este valor debe coincidir con el nombre definido en la volumes definición.
mountPath	Este valor define la ruta de acceso del contenedor donde se monta el almacenamiento.

9. Actualice la aplicación contenedora con la nueva configuración de montaje de almacenamiento.

az containerapp update \

--name \$CONTAINER_APP_NAME \

--resource-group \$RESOURCE_GROUP \

--yaml app.yaml \

--output table

Comprobación del montaje de almacenamiento

Ahora que se establece el montaje de almacenamiento, puede manipular archivos en Azure Storage desde el contenedor. Use los siguientes comandos para observar el montaje de almacenamiento en el trabajo.

1. Abra un shell interactivo dentro de la aplicación contenedora para ejecutar comandos dentro del contenedor en ejecución.

```
az containerapp exec \
```

```
--name $CONTAINER_APP_NAME \
```

```
--resource-group $RESOURCE_GROUP
```

Este comando puede tardar un momento en abrir el shell remoto. Una vez que el shell esté listo, puede interactuar con el montaje de almacenamiento a través de comandos del sistema de archivos. Para más información, consulte [Conexión a una consola de contenedor en Azure Container Apps](#).

2. Cambie a la carpeta `nginx /var/log/nginx`.

```
cd /var/log/nginx
```

3. Vuelva al explorador y vaya al sitio web y actualice la página varias veces.

Las solicitudes realizadas al sitio web crean una serie de entradas de flujo de registro.

4. Vuelva al terminal y enumere los valores de la `/var/log/nginx` carpeta.

```
ls
```

Observe cómo los archivos `access.log` y `error.log` aparecen en esta carpeta. Estos archivos se escriben en el montaje de Azure Files en el recurso compartido de Azure Storage creado en los pasos anteriores.

5. Vea el contenido del archivo `access.log`.

```
cat access.log
```

6. Salga del shell interactivo del contenedor para volver a la sesión del terminal local.

```
exit
```

7. Ahora, puede ver los archivos en Azure Portal para comprobar que existen en la cuenta de Azure Storage. Imprima el nombre de la cuenta de almacenamiento generada aleatoriamente.

```
echo $STORAGE_ACCOUNT_NAME
```

8. Vaya a Azure Portal y abra la cuenta de almacenamiento creada en este procedimiento.

9. En **Almacenamiento de datos**, seleccione **uso compartido de archivos**.

10. Seleccione **myshare** para ver los `archivos access.log` y `error.log`.

Limpieza de recursos

Si no va a seguir usando esta aplicación, ejecute el siguiente comando para eliminar el grupo de recursos junto con todos los recursos creados en este artículo.

```
az group delete \
```

```
--name $RESOURCE_GROUP
```