



Exercise - Create a key vault and store secrets

7 minutes

Sandbox activated! Time remaining: **3 min**

You have used 1 of 10 sandboxes for today. More sandboxes will be available tomorrow.

Creating Key Vaults for your applications

Good practice is to create a separate vault for each deployment environment of each of your applications, such as development, test, and production. You can use a single vault to store secrets for multiple apps and environments, but the impact of an attacker gaining read access to a vault increases with the number of secrets in the vault.

Tip

If you use the same names for secrets across different environments for an application, the only environment-specific configuration that has to change in your app is the vault URL.

Creating a vault requires no initial configuration. Your user identity is automatically granted the full set of secret management permissions and you can start adding secrets immediately. Once you have a vault, adding and managing secrets can be done from any Azure administrative interface, including the Azure portal, the Azure CLI, and Azure PowerShell. When you set up your application to use the vault, you'll need to assign the correct permissions to it; we'll see that in the next unit.

Create the vault and store the secret in it

Given all the trouble the company's been having with application secrets, management has asked you to create a small starter app to set the other developers on the right path. The app needs to demonstrate best practices for managing secrets as simply and securely as possible.

To start, you'll create a vault and store one secret in it.

Create the vault

Key Vault names must be globally unique, so you'll need to pick a unique name. Vault names must be 3-24 characters long and contain only alphanumeric characters and dashes. Make a note of the vault name you choose, as you'll need it throughout this exercise.

Run the following command in the Cloud Shell to create your vault.

Azure CLI

 Copy

```
az keyvault create \  
  --resource-group learn-76815a55-2a7e-4698-98b8-1da305fa359e \  
  --location centralus \  
  --name <your-unique-vault-name>
```

When it finishes, you'll see JSON output describing the new vault.

Tip

The command used the pre-created resource group named **learn-76815a55-2a7e-4698-98b8-1da305fa359e**. When working with your own subscription, you would want to either create a new resource group, or use an existing one you have previously created.

Add the secret

Now add the secret: our secret will be named **SecretPassword** with a value of **reindeer_flotilla**.

Azure CLI


 Copy

```
az keyvault secret set \  
  --name SecretPassword \  
  --value reindeer_flotilla \  
  --vault-name <your-unique-vault-name>
```

We'll write the code for our application shortly, but first we need to learn a little bit about how our app is going to authenticate to a vault.

Next unit: Vault authentication with managed identities for Azure resources

Continue >

 English (United States)

[Previous Version Docs](#) • [Blog](#) • [Contribute](#) • [Privacy & Cookies](#) • [Terms of Use](#) • [Trademarks](#) •

© Microsoft 2020

 Azure Cloud Shell

```
"active": true,  
"author": "N/A",  
"author_email": "N/A",  
"complete": true,  
"deployer": "Push-Deployer",  
"end_time": "2020-04-01T12:27:44.9113677Z",  
"id": "69f8737de4d14157a48cc1e817197519",  
"is_readonly": true,
```