

Keyvault Class

Reference



Manages secrets stored in the Azure Key Vault associated with an Azure Machine Learning workspace.

Each Azure Machine Learning workspace has an associated [Azure Key Vault](#). The Keyvault class is a simplified wrapper of the Azure Key Vault that allows you to manage secrets in the key vault including setting, retrieving, deleting, and listing secrets. Use the Keyvault class to pass secrets to remote runs securely without exposing sensitive information in cleartext.

For more information, see [Using secrets in training runs](#).

In this article

- [Constructor](#)
- [Remarks](#)
- [Methods](#)

Inheritance `builtins.object` → `Keyvault`

Constructor

Python
<code>Keyvault(workspace)</code>

Parameters

workspace [Workspace](#)

Required

The Azure Machine Learning Workspace associated with this key vault.

Remarks

In submitted runs on local and remote compute, you can use the [get_secret](#) method of the Run instance to get the secret value from Key Vault. To get multiple secrets, use the [get_secrets](#) method of the Run instance.

These Run methods gives you a simple shortcut because the Run instance is aware of its Workspace and Keyvault, and can directly obtain the secret without the need to instantiate the Workspace and Keyvault within the remote run.

The following example shows how to access the default key vault associated with a workspace and set a secret.

Python

```
import uuid

local_secret = os.environ.get("LOCAL_SECRET", default =
str(uuid.uuid4())) # Use random UUID as a substitute for real secret.
keyvault = ws.get_default_keyvault()
keyvault.set_secret(name="secret-name", value = local_secret)
```

Full sample is available from <https://github.com/Azure/MachineLearningNotebooks/blob/master/how-to-use-azureml/manage-azureml-service/authentication-in-azureml/authentication-in-azureml.ipynb>

Methods

delete_secret	Delete the secret with the specified name.
delete_secrets	Delete a list of secrets from the Azure Key Vault associated with the workspace.
get_secret	Return the secret value for a given secret name.
get_secret_content_type	Return the secret's content type for a given secret name.
get_secrets	Return the secret values for a given list of secret names.
list_secrets	Return the list of secret names from the Azure Key Vault associated with the workspace. This method does not return the secret values.

<code>set_secret</code>	Add a secret to the Azure Key Vault associated with the workspace.
<code>set_secrets</code>	Add the dictionary of secrets to the Azure Key Vault associated with the workspace.

delete_secret

Delete the secret with the specified name.

Python
<code>delete_secret(name)</code>

Parameters

name `str`

Required

The name of the secret to delete.

Return type

`None`

delete_secrets

Delete a list of secrets from the Azure Key Vault associated with the workspace.

Python
<code>delete_secrets(secrets)</code>

Parameters

secrets_batch `list[str]`

Required

The list of secrets to delete.

secrets

Required

Return type

None

get_secret

Return the secret value for a given secret name.

Python
<code>get_secret(name)</code>

Parameters

name `str`

Required

The secret name to return the value for.

Returns

The secret value for a specified secret name.

Return type

`str`

get_secret_content_type

Return the secret's content type for a given secret name.

Python
<code>get_secret_content_type(name)</code>

Parameters

name `str`

Required

The secret name to return the content type for.

Returns

The secret content type for a specified secret name.

Return type

`str`

get_secrets

Return the secret values for a given list of secret names.

Python
<code>get_secrets(secrets)</code>

Parameters

secrets `list[str]`

Required

The list of secret names to retrieve values for.

Returns

A dictionary of found and not found secrets.

Return type

`dict(<xref:str: str>)`

list_secrets

Return the list of secret names from the Azure Key Vault associated with the workspace.

This method does not return the secret values.

Python
<code>list_secrets()</code>

Returns

A list of dictionary of secret names with format {name : "secretName"}

Return type

`dict(<xref:str:str>)`

set_secret

Add a secret to the Azure Key Vault associated with the workspace.

Python
<code>set_secret(name, value, content_type=<KeyVaultContentType.not_provided: 'application/NotProvided'>)</code>

Parameters

name `str`

Required

The name of the secret to add.

value `<xref:azureml.core.azureml._restclient.models.KeyVaultContentType>`

Required

The value of the secret to add.

value

Required

The content type of the secret to add.

content_type

default value: KeyVaultContentType.not_provided

Return type

None

set_secrets

Add the dictionary of secrets to the Azure Key Vault associated with the workspace.

Python
<code>set_secrets(secrets_batch)</code>

Parameters

secrets_batch dict(<xref:str:str>)

Required

A dictionary of secret names and values to add.

Return type

None