In Azure Storage, a **Shared Key** (also known as an **Account Key**) is a **way to authorize access** to your storage account. It works like a **password** that grants access to all data and services in the storage account, such as blobs, files, queues, and tables.

There are two shared keys (key1 and key2) for redundancy—you can rotate one without interrupting service.

**🔐 What is it used for?**

Shared Key authentication is used when:

* Accessing storage via **REST API**, **Azure SDKs**, or **Azure CLI**.
* You need to authenticate manually or from a script, app, or tool.
* Using tools like **AzCopy**, **Storage Explorer**, or custom applications.

az storage blob upload \

--account-name mystorageaccount \

--account-key <your-account-key> \

--container-name mycontainer \

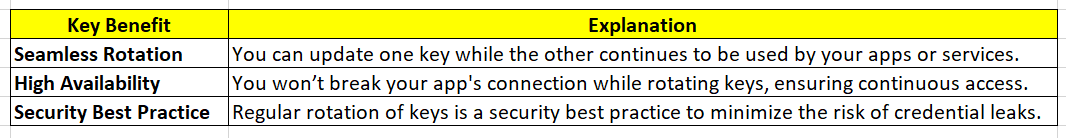
--name myfile.txt \

--file ./myfile.txt

In this example:

* --account-name is your Azure Storage Account.
* --account-key is the shared key (copied from the Azure Portal).
* You’re uploading a file (myfile.txt) into a blob container.

**Why Azure provides two shared keys for a Storage Account** Azure gives you **two shared keys (key1 and key2)** to support **key rotation** without downtime. Here's why that's important:



Conditional question: I have storage account which has 2 shared key, tell me best practice how I can use them in script, so if one is expired, I can use another. I am referring to azure storage account shared key

**✅ Best Practice: Use Fallback Logic in Your Script**

**🧪 Step-by-step:**

1. **Store both keys securely** (e.g., in Azure Key Vault or environment variables).
2. **Try key1** first.
3. If it **fails**, **catch the error** and **retry with key2**.
4. **Log failure** and alert if both fail.