



# Identify a queue

5 minutes

Now that we have a storage account, let's look at how we work with the queue that it will hold.

To access a queue, you need three pieces of information:

1. Storage account name
2. Queue name
3. Authorization token

This information is used by both applications that talk to the queue (the web front end that adds messages and the mid-tier that processes them).

## Queue identity

Every queue has a name that you assign during creation. The name must be unique within your storage account but doesn't need to be globally unique (unlike the storage account name).

The combination of your storage account name and your queue name uniquely identifies a queue.

## Access authorization

Every request to a queue must be authorized and there are several options to choose from.

| Authorization Type            | Description                                                                                                                                                                                                                                                                                                                                  |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Azure Active Directory</b> | You can use role-based authentication and identify specific clients based on AAD credentials.                                                                                                                                                                                                                                                |
| <b>Shared Key</b>             | Sometimes referred to as an <b>account key</b> , this is an encrypted key signature associated with the storage account. Every storage account has two of these keys that can be passed with each request to authenticate access. Using this approach is like using a root password - it provides <i>full access</i> to the storage account. |



| Authorization Type             | Description                                                                                                                                                                                                                                                                 |
|--------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Shared access signature</b> | A shared access signature (SAS) is a generated URI that grants limited access to objects in your storage account to clients. You can restrict access to specific resources, permissions, and scope to a data range to automatically turn off access after a period of time. |

### Note

We will use the account key authorization because it is the simplest way to get started working with queues, however it's recommended that you either use shared access signature (SAS) or Azure Active Directory (AAD) in production apps.

## Retrieving the account key

Your account key is available in the **Settings > Access keys** section of your storage account in the Azure portal, or you can retrieve it through the command line:

|                                               |                                                                                            |
|-----------------------------------------------|--------------------------------------------------------------------------------------------|
| Azure CLI                                     |  Copy |
| <code>az storage account keys list ...</code> |                                                                                            |
| PowerShell                                    |  Copy |
| <code>Get-AzStorageAccountKey ...</code>      |                                                                                            |

## Accessing queues

You access a queue using a REST API. To do this, you'll use a URL that combines the name you gave the storage account with the domain `queue.core.windows.net` and the path to the queue you want to work with. For example: `http://<storage account>.queue.core.windows.net/<queue name>`. An `Authorization` header must be included with every request. The value can be any of the three authorization styles.


## Using the Azure Storage Client Library for .NET

The Azure Storage Client Library for .NET is a library provided by Microsoft that formulates REST requests and parses REST responses for you. This greatly reduces the amount of code you need to write. Access using the client library still requires the same pieces of information

(storage account name, queue name, and account key); however, they are organized differently.

The client library uses a **connection string** to establish your connection. Your connection string is available in the **Settings** section of your Storage Account in the Azure portal, or through the Azure CLI and PowerShell.

A connection string is a string that combines a storage account name and account key and must be known to the application to access the storage account. The format looks like this:

|                                                                                                                                                                                 |                                                                                          |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|
| C#                                                                                                                                                                              |  Copy |
| <pre>string connectionString = "DefaultEndpointsProtocol=https;AccountName=&lt;your storage account name&gt;;AccountKey=&lt;your key&gt;;EndpointSuffix=core.windows.net"</pre> |                                                                                          |

### Warning

This string value should be stored in a secure location since anyone who has access to this connection string would be able to manipulate the queue.

Notice that the connection string doesn't include the queue name. The queue name is supplied in your code when you establish a connection to the queue.

Let's get our connection string from Azure and set up a new application to use it.

---

## Next unit: Exercise - Identify a queue

Continue >