Data Storage using Azure Data Lake Store gen1 & Blob Storage

Lab 2 – Data Ingestion & storage using Azure Data Lake Store gen1 & Blob.

Overview

In this lab, you will understand the various process to transfer data with the AzCopy/Azure Storage Explorer/Data explorer/Import & Export Tool utility.

What You’ll Need

To complete the labs, you will need the following:

* A web browser
* A Microsoft account
* A Microsoft Azure subscription
* A Microsoft Windows, Linux, or Apple Mac OS X computer on which the Azure CLI has been installed.
* Azure PowerShell

The lab files for this course

## Prerequisites:

* For AzCopyV10, no installation is required & can start directly from command line with azcopy.exe executable is located. You can add the AzCopy folder location to your system path.

## Commands / Scripts

* First, use the following command to sign in using Azure AD . The user should have “Storage Blob Data Contributor” role assigned to write to blob storage & using Azure AD authentication.

.\azcopy login

### .\azcopy <command> <arguments> --<flag-name>=<flag-value>

### # Examples if you have **logged into the Azure Active Directory**:

### .\azcopy copy <source path> <destination path> --<flag-name>=<flag-value>

### .\azcopy cp "C:\local\path" "https://account.blob.core.windows.net/container" --recursive=true

### .\azcopy cp "C:\local\path\myfile" "https://account.blob.core.windows.net/container/myfile"

### .\azcopy cp "C:\local\path\\*" "https://account.blob.core.windows.net/container"

### # Examples if you are using **SAS tokens to authenticate**:

### .\azcopy cp "C:\local\path" "https://account.blob.core.windows.net/container?sastoken" --recursive=true

### .\azcopy cp "C:\local\path\myfile" <https://account.blob.core.windows.net/container/myfile?sastoken>

* **Copy data to Azure Storage –** First using AzCopy create the Blob storage & File share account & then start transferring data from on-premise/another Cloud based storage.

.\azcopy copy <source path> <destination path> --<flag-name>=<flag-value>

# Using alias instead

.\azcopy cp <source path> <destination path> --<flag-name>=<flag-value>

.\azcopy cp "C:\local\path" "https://account.blob.core.windows.net/mycontainer1<sastoken>" --recursive=true

To upload all files from on-premise C:\local\path directory.

.\azcopy cp "C:\local\path\\*" [https://account.blob.core.windows.net/mycontainer1<sastoken>](https://account.blob.core.windows.net/mycontainer1%3csastoken%3e)

* Upload blobs to Azure Blob Storage
* **Upload a single blob**

AzCopy /Source:C:\yourfolder /Dest:https://youraccount.blob.core.windows.net/yourcontainer /DestKey:key /Pattern:"abc.txt" (Note: AzCopy will create the blob container in case the specified Blob container doesn’t exist.

* **Upload all blobs in a folder**

AzCopy /Source:C:\yourfolder /Dest:https://youraccount.blob.core.windows.net/mycontainer /DestKey:key /S

Note: specifying /s uploads the contents of the specified directory to blob storage recursively.

* Upload blobs matching a specific pattern

AzCopy /Source:C:\myfolder /Dest:https://myaccount.blob.core.windows.net/mycontainer /DestKey:key /Pattern:a\* /S

## **Use Azure Import/Export services to transfer files to Azure Blob storage**

* Enable Bitlocker on the Wndows system. <http://thesolving.com/storage/how-to-enable-bitlocker-on-windows-server-2012-r2/>
* Download the WAImportExport version 1 on the Windows system. <https://aka.ms/waiev1>
* Open the PowerShell or cmd with admin privileges.

cd C:\WaImportExportV1

* Retrieve bitlocker key of the drive , execute the following command:

manage-bde -protectors -get <DriveLetter>:

* In order to prepare the disk, run the following command, depending on the data size, it may consume from hours to days for shifting data to Azure.

### ./WAImportExport.exe PrepImport /j:<journal file name> /id:session#<session number> /sk:<Storage account key> /t:<Drive letter> /bk:<BitLocker key> /srcdir:<Drive letter>:\ /dstdir:<Container name>/ /skipwrite

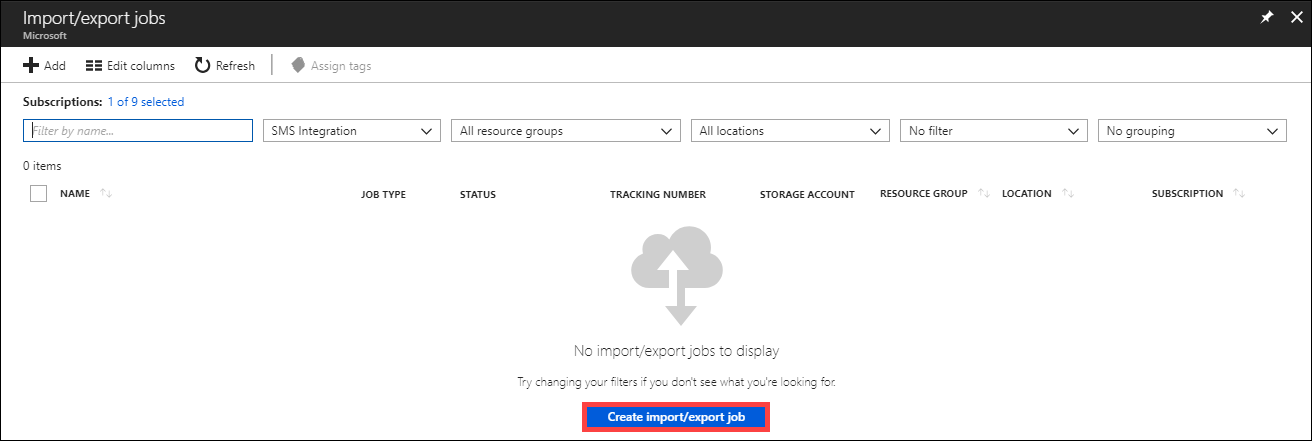
A journal file is created in the same folder where you ran the tool. Two other files are also created - an *.xml* file (folder where you run the tool) and a *drive-manifest.xml* file (folder where data resides).

The parameters used are described in the following table:

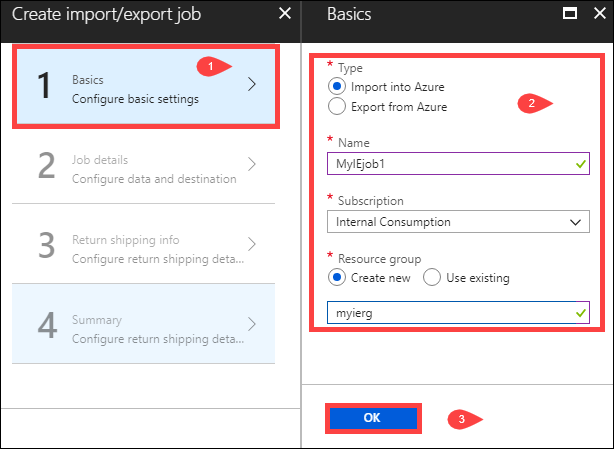
| **Option** | **Description** |
| --- | --- |
| /j: | The name of the journal file, with the .jrn extension. A journal file is generated per drive.  Microsoft recommends that you use the disk serial number as the journal file name. |
| /id: | The session ID. Use a unique session number for each instance of the command. |
| /sk: | The Azure Storage account key. |
| /t: | The drive letter of the disk to be shipped. For example, drive D. |
| /bk: | The BitLocker key for the drive. Its numerical password from output of manage-bde -protectors -get D: |
| /srcdir: | The drive letter of the disk to be shipped followed by :\. For example, D:\. |
| /dstdir: | The name of the destination container in Azure Storage. |
| /skipwrite: | The option that specifies that there is no new data required to be copied and existing data on the  disk is to be prepared. |

## Create an Import job

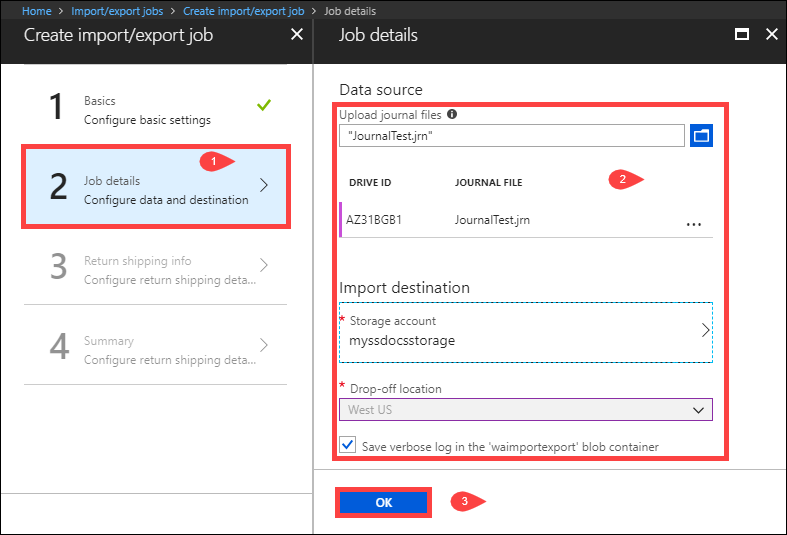
* Login into <https://portal.azure.com>
* Go to All services -> Storage -> Import/Export jobs
* Click into Import/Export job.



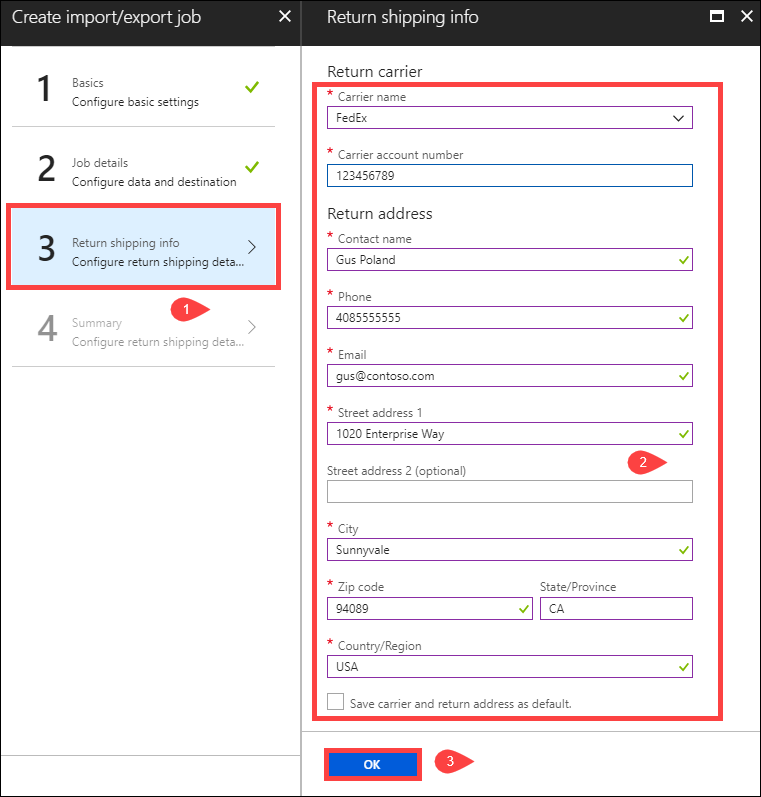
* In Basics: select Import into Azure.
* Enter a descriptive name for the import job, use the name to track the progress of your jobs.



* In job details, refer the drive journal files & upload one file for each drive that prepared. If journal file > 2 MB. Then use <Journal file name>\_DriveInfo\_<Drive serial ID>.xml.
* Select the destination storage account where data will reside. The dropoff location will be populated automatically on the storage account region.



* Select the carrier from the dropdown list.
* Enter a valid carrier account number that you have created with that carrier. Microsoft uses this account to ship the drives back to you once your import job is complete. If you do not have an account number, create a [FedEx](http://www.fedex.com/us/oadr/) or [DHL](http://www.dhl.com/) carrier account.



* Next step should be shipping the drives & update the job tracking information.
* Provide a valid FedEx, UPS, or DHL carrier account number that Microsoft will use to ship the drives back.
  + A FedEx, UPS, or DHL account number is required for shipping drives back from the US and Europe locations.
  + A DHL account number is preferred for shipping drives back from Asia and Australia locations.
  + If you do not have an account number, create a [FedEx](http://www.fedex.com/us/oadr/) or [DHL](http://www.dhl.com/) carrier account.
* When shipping your packages, you must follow the [Microsoft Azure Service Terms](https://azure.microsoft.com/support/legal/services-terms/).
* Properly package yours disks to avoid potential damage and delays in processing.