

## Azure Video on Demand (VOD) Solution: Wowza Streaming Engine, Aspera Faspex and Azure Media Services

# Hands on Lab Manual

*This hands-on lab session will give you a brief introduction of Video on Demand solution stack on Azure built with services from Aspera, Wowza and Azure Media Services.*



## I. Introduction

Welcome to the Launch and Learn hands on lab session at Ignite. In this session, you will be able to understand how a fully integrated Video on Demand (VoD) stack is put together on Azure and use the solution tacks to take a video file through the workflow.

The Quickstart solution is comprised of Aspera's Faspex file transfer service, Wowza's media streaming engine and Azure Media Services all working together to provide a Video on Demand environment. Quickstart templates make it very simple and fast for enterprises to pilot (and validate) fully baked solutions before putting into production. The hands-on lab will demonstrate how:

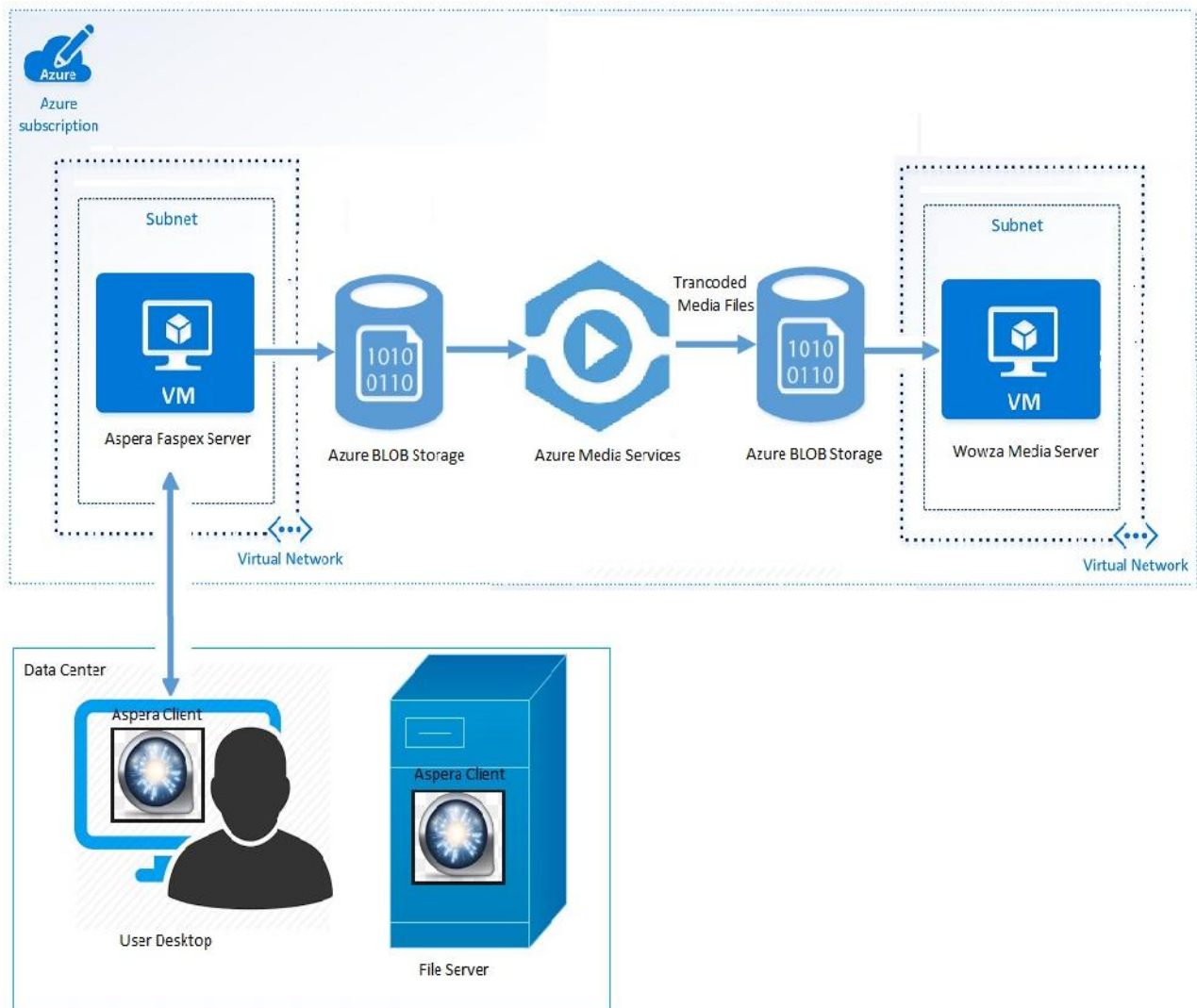
- Aspera Faspex, deployed in Azure is used to move the media files from the client machine to the Azure storage.
- Azure Media Services is used to run transcoding jobs on the inputted media files and output them to azure storage.
- And, Wowza streaming engine to pick up the transcoded media files and deliver/stream them to the clients.

The deployment of the Quickstart solution takes around 15 to 20 minutes' start to finish. For the purposes of this lab, the stack is already pre-deployed.

## II. What you will learn in this lab session?

1. How ARM templates are integrated with Aspera, Azure media services and Wowza Streaming Engine.
2. Transfer files to Azure storage using Aspera's Faspex client and server.
3. Transcoding of video file(s) with Azure media services API
4. Streaming video content through Wowza streaming engine.

### III. Solution Overview



## Components of the Quickstart Solution

---

The diagram above shows the overall deployment architecture for the Video on Demand solution stack. As a part of deployment, the template launches the following:

- Aspera Faspex Server
- Azure Storage Accounts
- Azure Media Services
- Wowza Streaming Server

### Aspera Faspex Server

Aspera Faspex will be running in a Linux Centos Machine. The size of VM is Standard DS1. Aspera's transfer service can move terabytes of data in and out of Azure Blobs, as well as local storage, up to 100x faster than FTP. The Aspera application platform supports a variety of Aspera or custom client options for desktops, web and mobile transfers. For more information on Aspera, please visit <https://www.aspera.com>

### Azure Storage Account

"End Users" will transfer the video files from their desktop to an Azure storage account "through" the Aspera connect client and are saved inside Azure Blob's storage container(s). For more information on Azure Blobs, please visit <https://azure.microsoft.com/en-us/>

### Azure Media Services

Azure Media Services is used for encoding/transcoding content for delivery across multiple formats and devices. In this lab user will transcode to multiple bitrate MP4 files and deliver them dynamically to the latest adaptive bitrate streaming protocols. For more information please visit <https://azure.microsoft.com/en-us/services/media-services/>

## Wowza Streaming Engine Server

Wowza Streaming Engine server is used for streaming of on-demand video over IP networks to desktops, laptops, tablets and mobile devices. Wowza Streaming Engine can stream to multiple types of playback clients and devices simultaneously, including the Adobe Flash player, Microsoft Silverlight player and Apple QuickTime Player. For more information, please visit <https://www.wowza.com/>

## IV. Tools needed for the hands-on lab

All the tools needed to complete the lab are pre-installed on the jump-host VM.

1. Azure Portal with access will be **provided by the instructor**
  - Username: [\\*\\*\\*\\*@2016ignite.onmicrosoft.com](mailto:****@2016ignite.onmicrosoft.com)
    - Actual user id will be provided in the class.
  - Password: Instructor will provide the password.
  - Azure Portal access at : <https://portal.azure.com>

## V. Hands on lab steps

### Access and key information:

The following information is needed to connect to each server. Please keep referring to them as needed.

- Customer ID: 89280462-5bf1-4371-95cc-ab4cb2f33a5e
- Entitlement key: cc924817-0621-457e-95ce-8dcb6813b648
- **Desktop login: duser/Admin@123**  
Provided as parameters while deploying and displayed in output
- Aspera server ssh login: auser/Admin@123  
Provided as parameters while deploying and displayed in output
- Wowza server ssh login: wuser/Admin@123  
Provided as parameters while deploying and displayed in output
- Wowza web portal login: wowza/Ignite@2016  
Deployed as part of the ARM script.

### 1. Access the azure portal

Your Azure Video on Demand stack/instance is already deployed for you.

The ARM template for this deployment is at

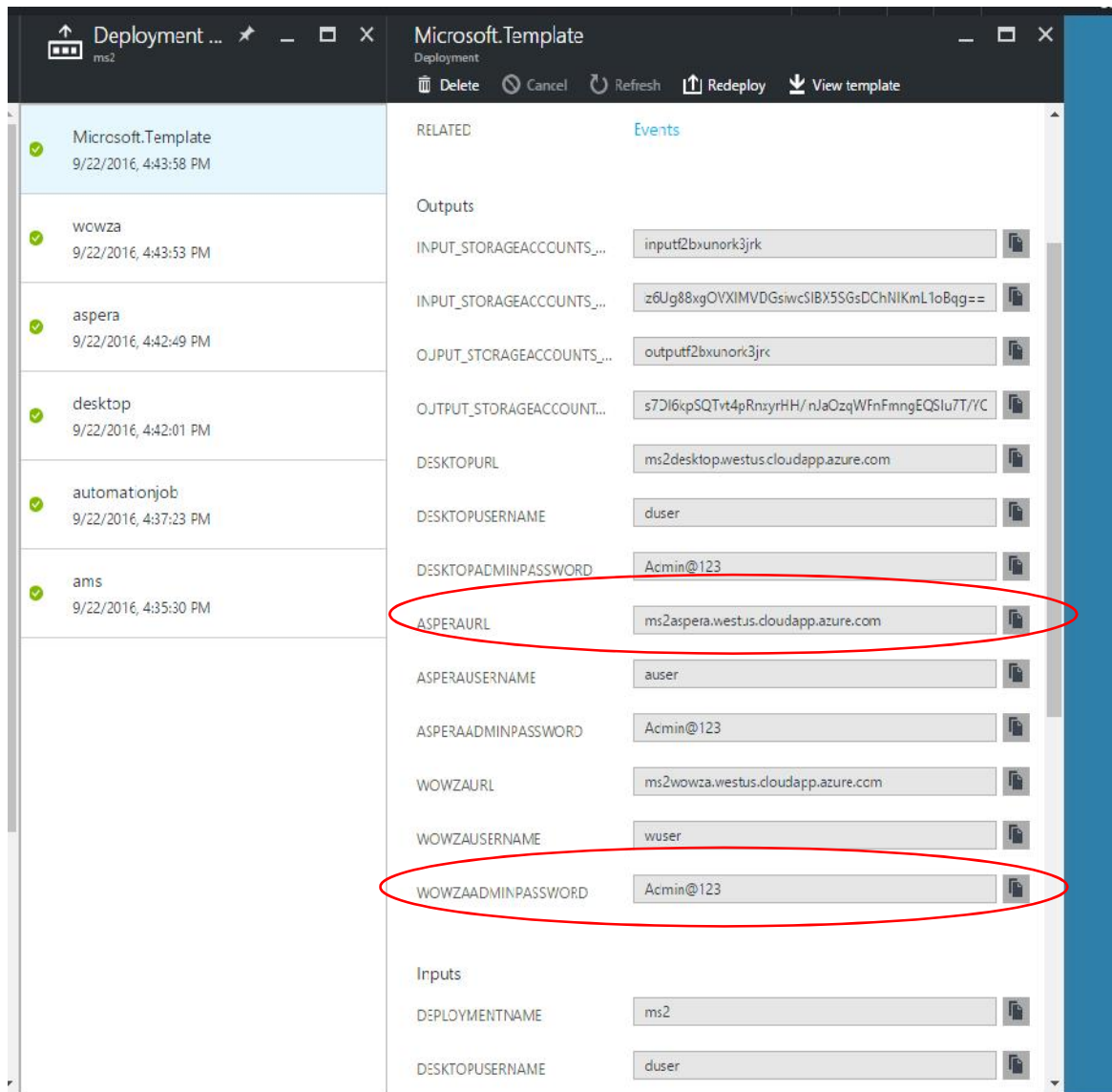
<https://github.com/sysgain/azurequickstarts/tree/master/VOD-Aspera-Wowza-AzureMediaServices>

Please log into the Azure Portal at <https://portal.azure.com> with the user id/password provided.

Access the portal and review the resources deployed.

- Click on the resource group.

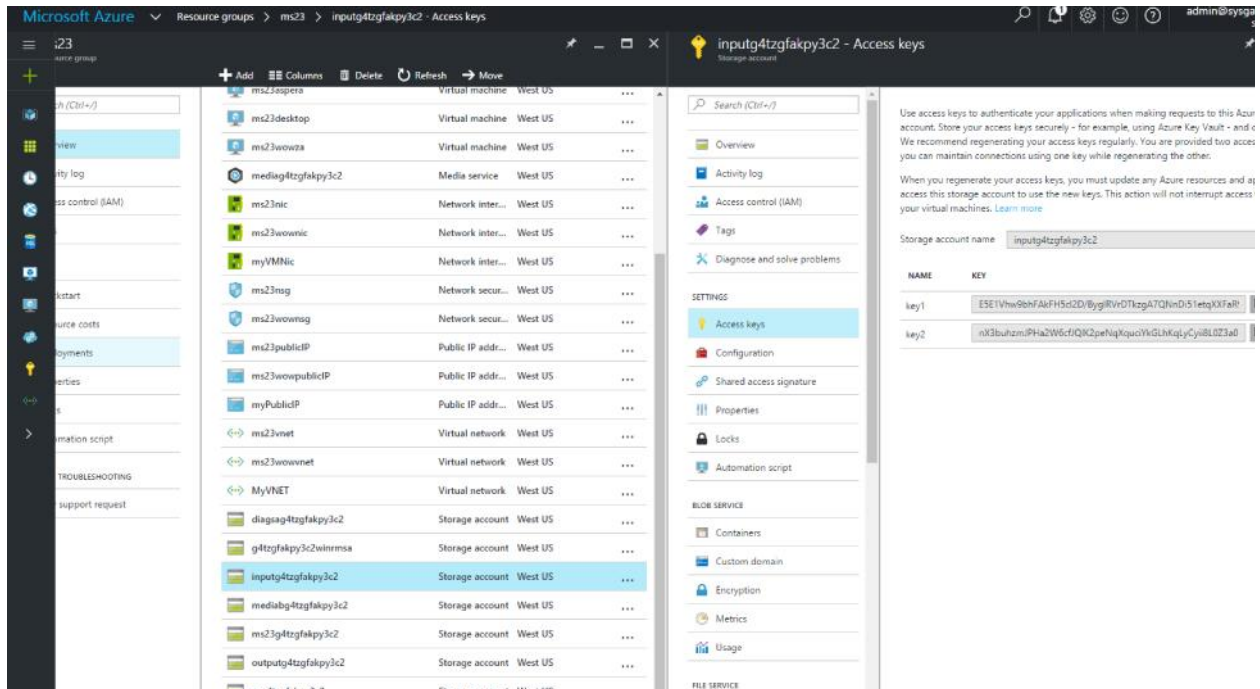
- Click the last deployment and review the outputs.



- Note down the Server name/IP name for **desktop**, Aspera and Wowza VMs which are deployed. They will have the name of \*aspera and \*wowza respectively.
  - Example:**      `ams21aspera.centralus.cloudapp.azure.com`      and  
                          `ams21wowza.centralus.cloudapp.azure.com`

- **Note the actual server names from your deployment from the Azure portal.**
- Note down the input storage account name and their access key. The input storage account will have a name of input\* under storage account type. Click on the name and go to Settings/Access keys to get the key 1. Review the screenshot below to how to get to storage account and access keys.
  - **Example:** Storage account name: inputyizo6zxgqmgd2 Storage account access key1: RjbLVcWfbLWDxRR/muEVQsKhTCs07/lc2k0K3QxnyrQ/OM1BlpUEHX+A t6FnIBuia3mPI9r2Evd1PbOKN80lyg==
  - **Note: Please make sure you get the actual values from your deployment using the azure portal**

We need both information to move the data from the client to the input storage.



The screenshot shows the Microsoft Azure portal interface. On the left, the navigation pane is visible with 'Storage accounts' selected. The main pane displays a list of resources in the 'ms23' resource group. The resource 'inputg4tzgfkpy3c2' is highlighted, and its details are shown on the right. The 'Access keys' tab is selected, displaying a table with two keys: 'key1' and 'key2'.

NAME	KEY
key1	ESE1Vhw9bhfAxFH5c2D/BygIRVdTKzgA7QInD151etqXXfaR
key2	nX3buhzmPHa2W6cFQK2pehlgKqucYhGLHkqlyCyJbL0Z3a0



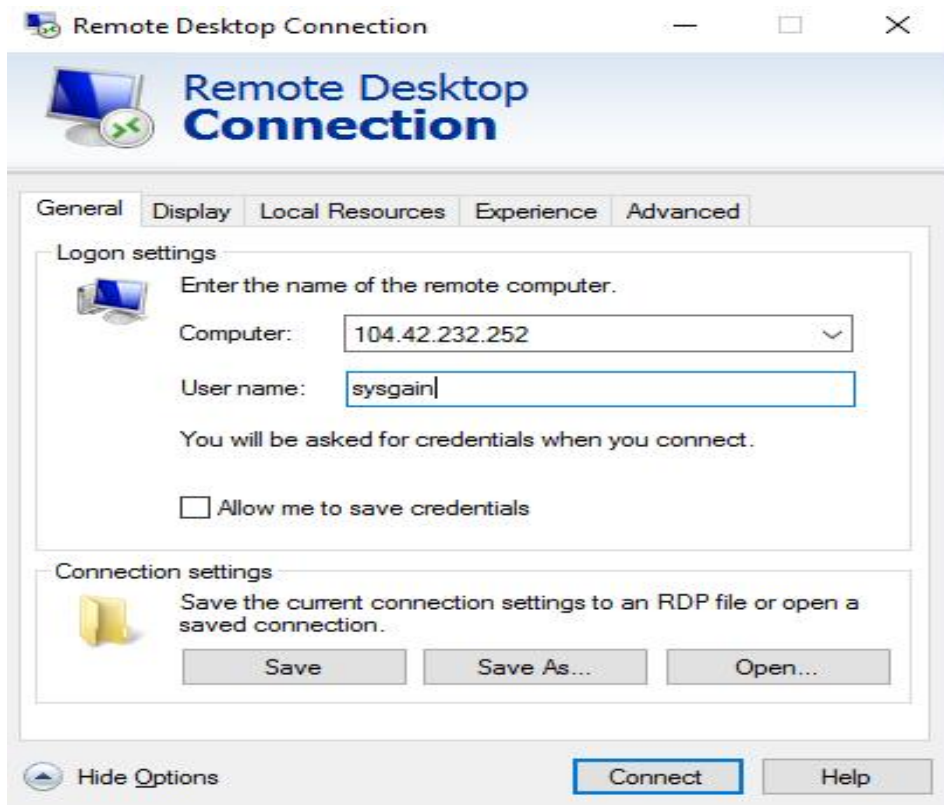
## 2. Remote Desktop into the Jump Host

The jump host is already configured with all necessary software/tools. Open remote desktop and connect to the windows server. Please review the following software shortcuts on desktop.

- Putty ssh client
- Aspera client
- Azure storage explorer

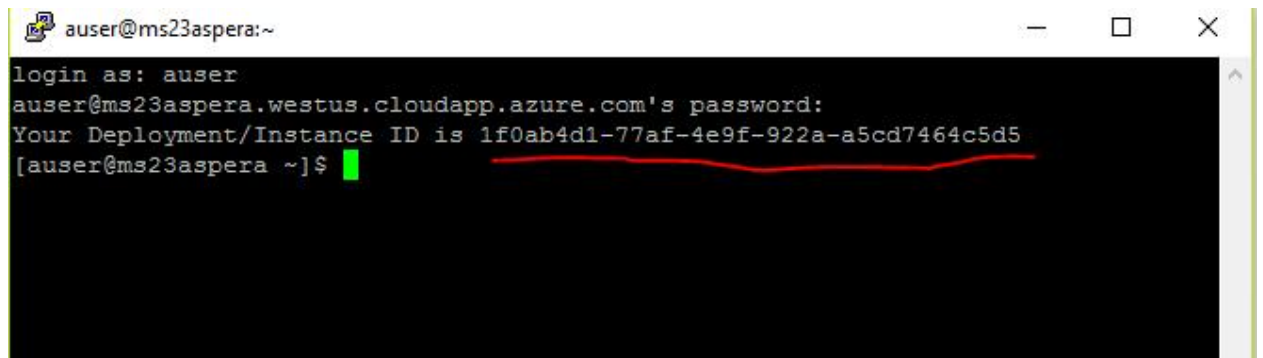
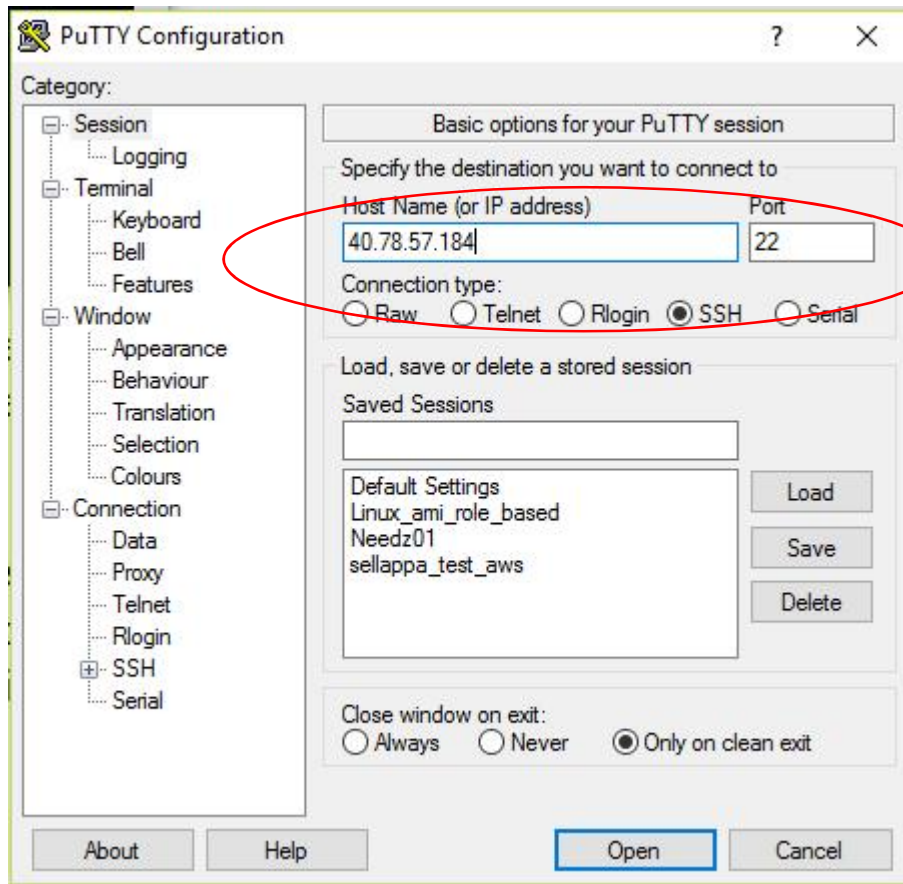
From the location machine search for Remote Desktop Connection, select the application and click. It will show a popup as below.

Provide the computer IP address and username/password as provided by the instructor.



### 3. Access Aspera Faspex Server using SSH

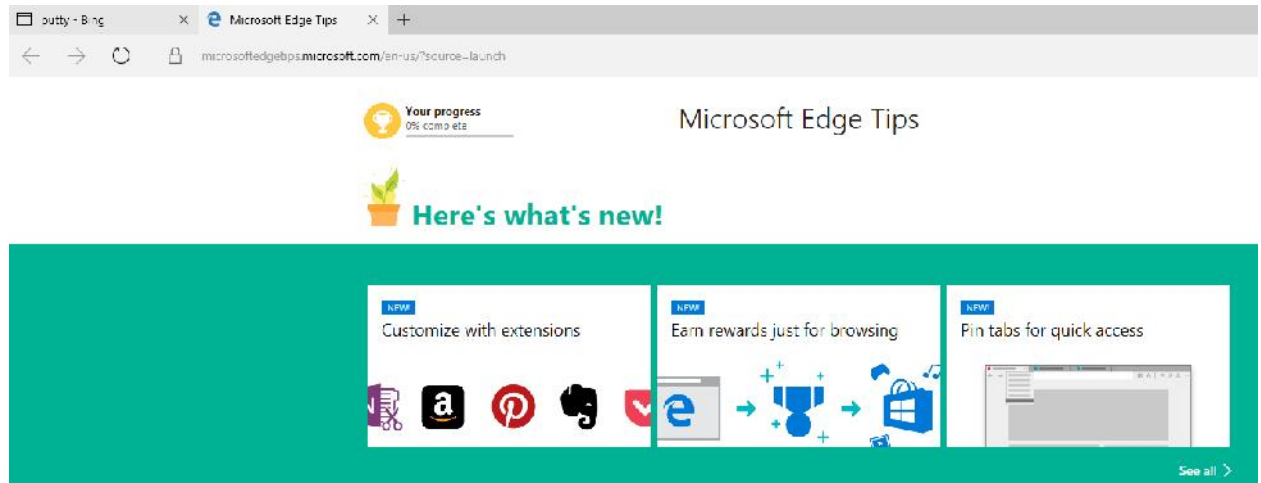
Open the putty application and open an ssh connection to the aspera server. On prompt accept and provide login information which you collected from the output window.



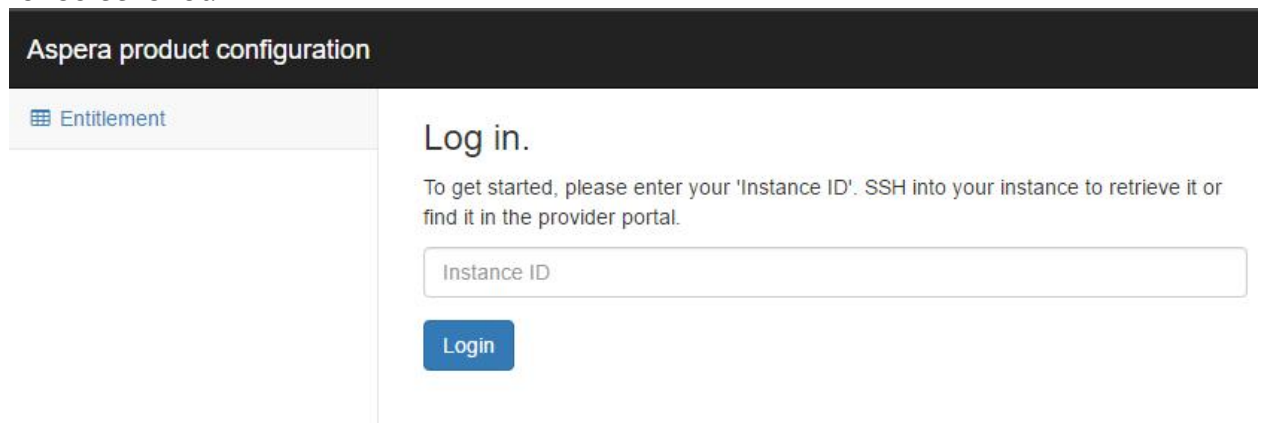
Note down the **instance id by double clicking** on it. It will be copied to the clipboard.

## 4. Access Aspera web app to configure

- Open a web browser from the desktop



- Login to the Aspera web application by pointing the browser to <https://<asperaServerName>/setup>
- Enter instance Id you copied from the previous step and submit. See below for screenshot.



- The interface would ask for customer id and entitlement id.
- Enter the following info and click "install" button.
  - o Customer ID: 89280462-5bf1-4371-95cc-ab4cb2f33a5e
  - o Entitlement key: cc924817-0621-457e-95ce-8dcb6813b648

**Aspera product configuration**

Entitlement

### Register for new entitlement

Start registration

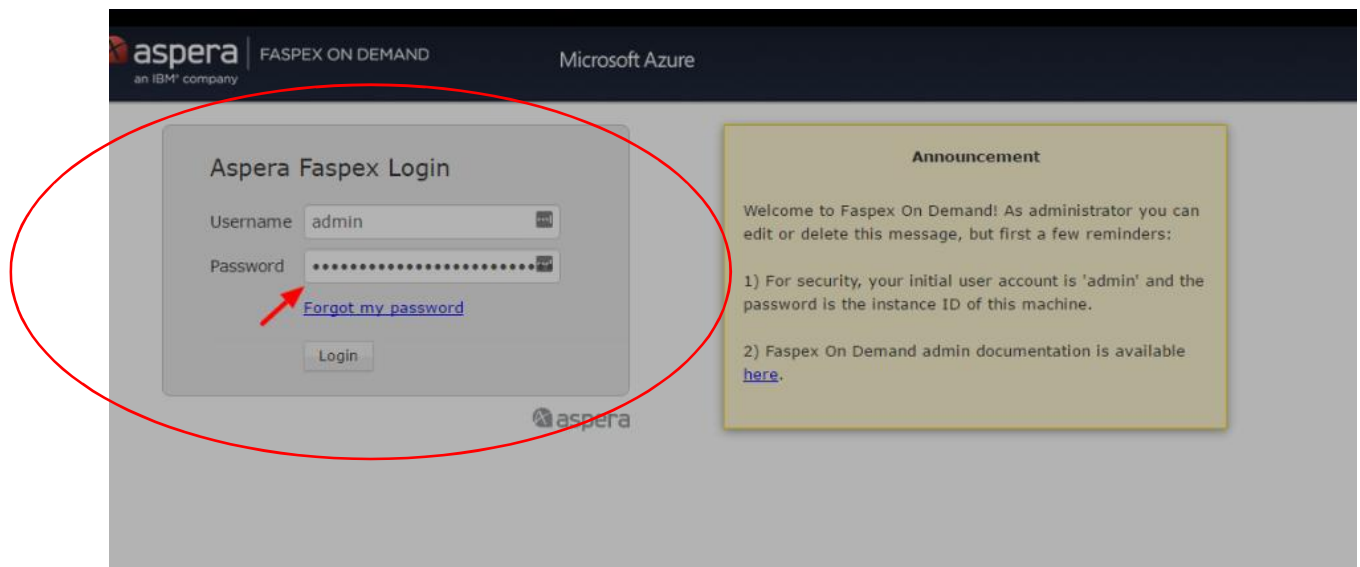
### Install entitlement

Customer ID:

Entitlement key:

Install

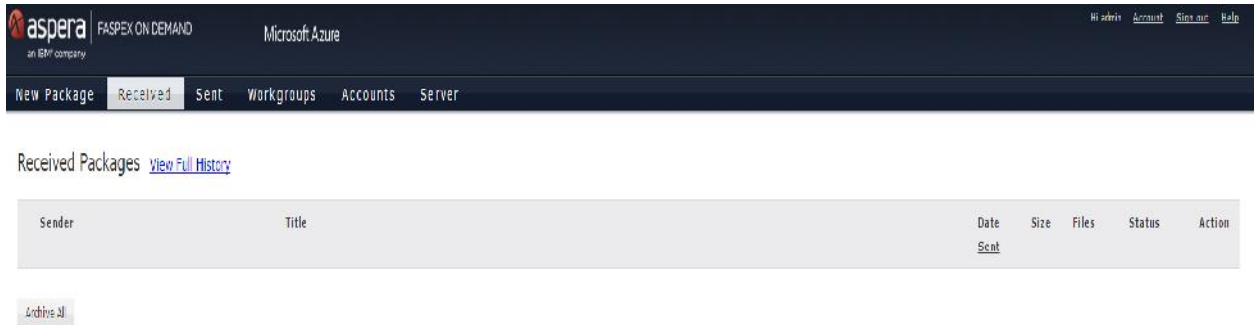
- After a few seconds you will be prompted with a login screen
- Enter "admin" for username and the instance id from the previous step for password. On login change the password to "Admin@123"



The screenshot shows the Aspera Faspex Login interface. The login form is highlighted with a red oval. It includes fields for Username (set to 'admin') and Password (masked with dots). Below the password field is a link for 'Forgot my password' with a red arrow pointing to it. A 'Login' button is at the bottom of the form. To the right, an 'Announcement' box contains a welcome message and two reminders: 1) The initial user account is 'admin' and the password is the instance ID. 2) Admin documentation is available at a link labeled 'here'.

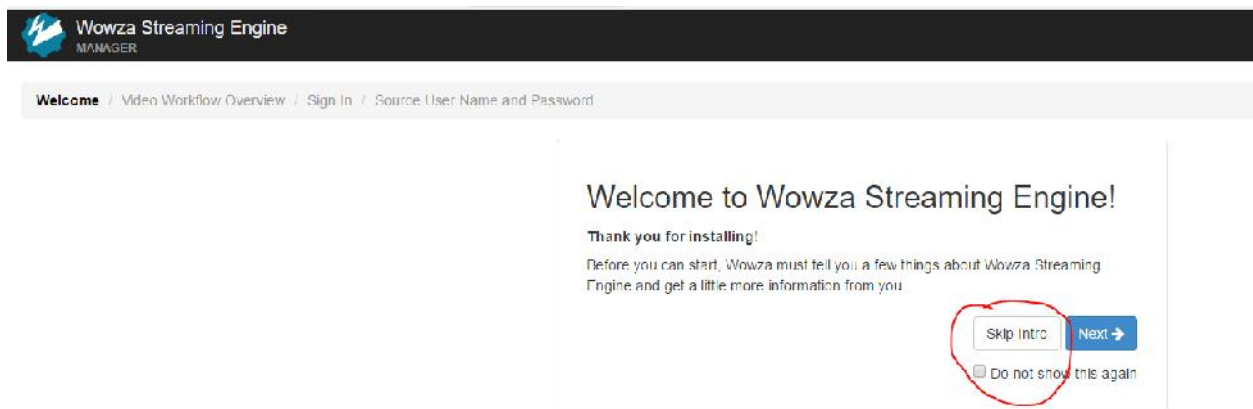
After password change, you will be prompted with a login screen. Upon logging in with Admin and new password that you gave in the previous step ( Admin@123),

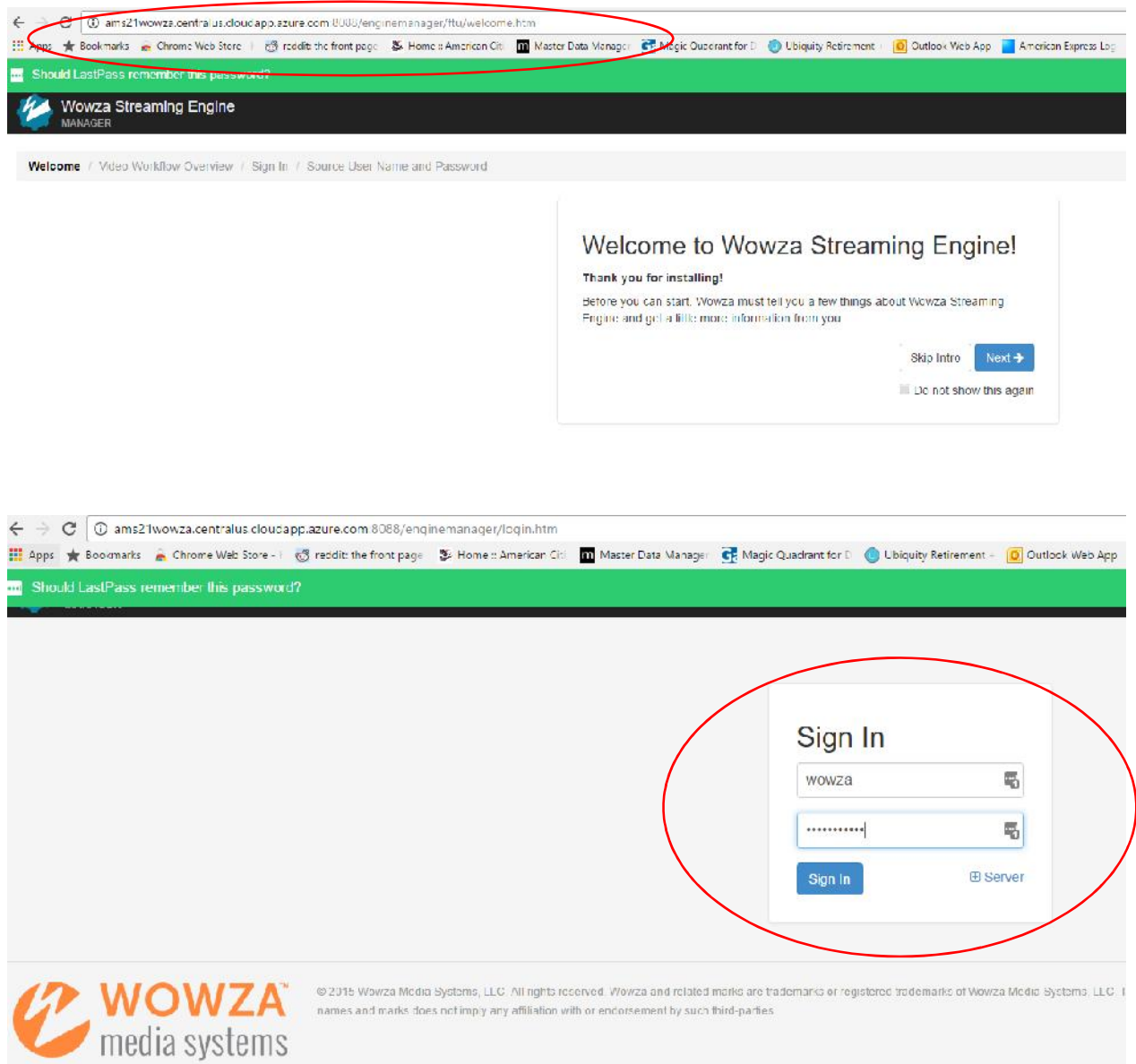
the below portal home page is displayed. This completes the setup of Aspera Server.




## 5. Connect to Wowza Streaming engine

- Login to the Wowza web application by opening the browser to `http://<wowzaServerName>:8088/enginemanager`
- Login with username and password **wowza/ignite@2016**
- Click Skip Intro.






- Once logged in, you will be presented with the welcome screen shown below.



Wowza Streaming Engine

MANAGER

[Home](#)
[Server](#)
[Applications](#)



[Help](#)
[Sign Out](#)

# Welcome to Wowza Streaming Engine!

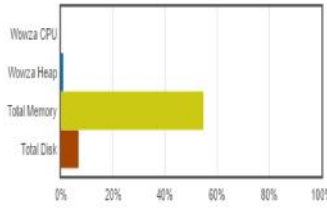
for Microsoft Azure 4.2.0 build15089 License

## Status

**Connections** incoming and outgoing



**Usage** CPU, Memory, Heap and Disk



**Server Uptime**

Since 22 Sep 2016 04:48:17 PM

**Features**

Transcoder: **Not Licensed** [Learn more](#)

DRM: **Not Licensed** [Learn more](#)

nDVR: **Not Licensed** [Learn more](#)

## Test Video

To play a video on demand test video, click **Test Players**.

[▶ Test Players...](#)

### Application Connection Settings

Use the following settings to publish a stream to Wowza Streaming Engine:

<b>Host - Server</b>	<code>\${com.wowza.cloud.platform.PLATFORM_METADATA_EXTERNAL_IP}</code>
<b>Host - Port</b>	1935
<b>Application</b>	A live application name on this server
<b>Stream Name</b>	The stream name you want to use
<b>Login</b>	A valid source user name and password

### Getting Started With Applications

Wowza Streaming Engine uses **applications** to deliver streaming content. An application is a set of settings for live or video on demand (VOD) streaming. Either use the preinstalled default applications or go to the [Add Application](#) page to easily create and configure new applications.

#### Live Applications

A **live** streaming application is preinstalled to allow you to easily publish video directly from a video encoder or IP camera to Wowza Streaming Engine. Visit the [Wowza Forums](#) for instructions on how to work with common encoders and cameras.

#### VOD Applications

A **vod** streaming application is also preinstalled with Wowza Streaming Engine. Simply copy video files to the Wowza Streaming Engine **content** directory to stream out via a **VOD** application.

#### Testing Playback

Test players are available for every application to help troubleshoot potential issues. Go to the [Applications](#) page, click the application name in the contents panel, and look for the **Test Players** button.

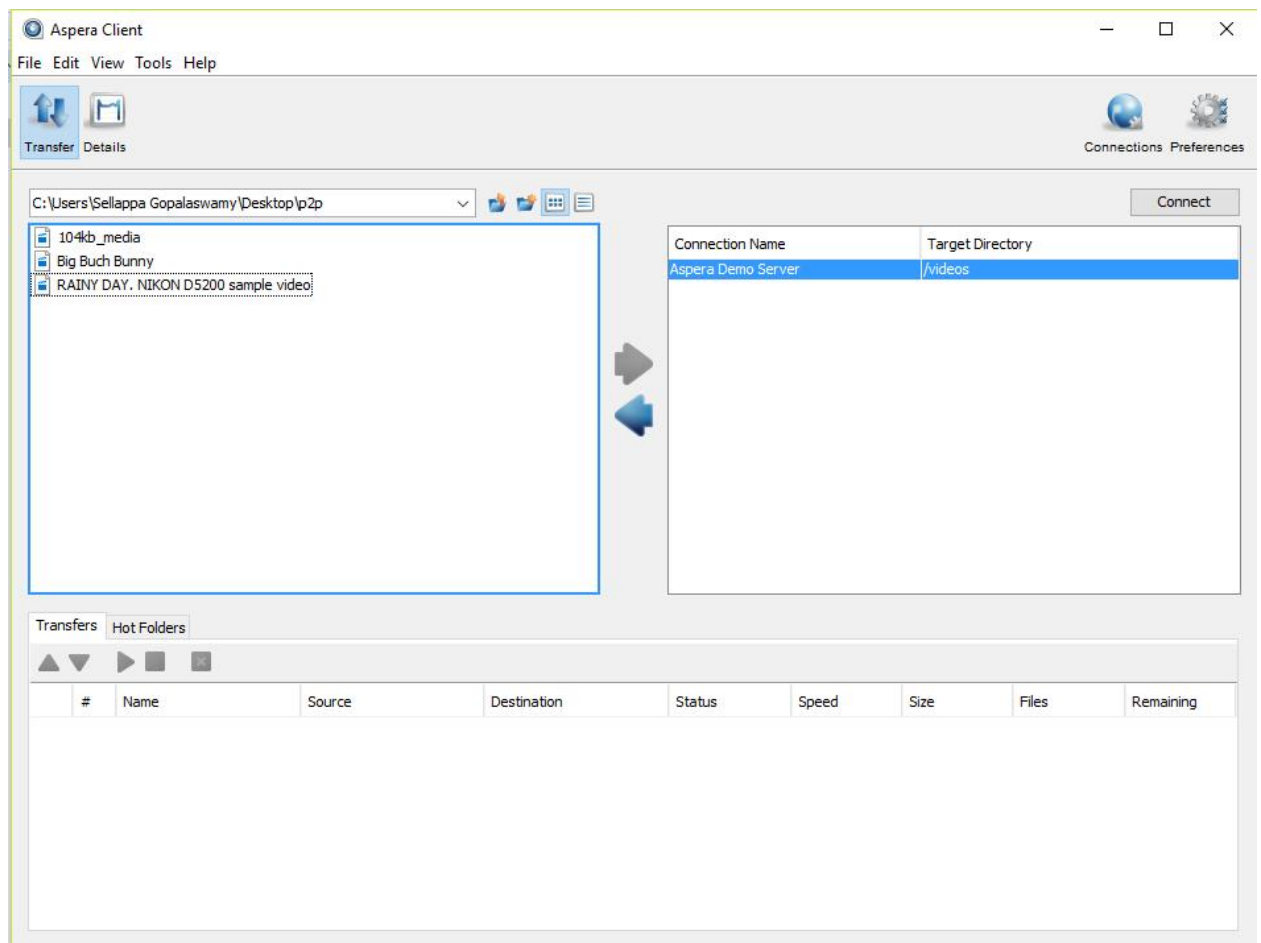
### Server Setup and Monitoring

Wowza Streaming Engine's default settings can be used for many streaming scenarios. Go to the [Server](#) page to change settings, create and manage accounts for users and live media sources, and to access statistics.



## 6. Connect to Aspera Faspex server using the Aspera client

Choose the remote desktop session and open the Aspera client installed on the jump host. You will find the Aspera client icon on the Desktop.

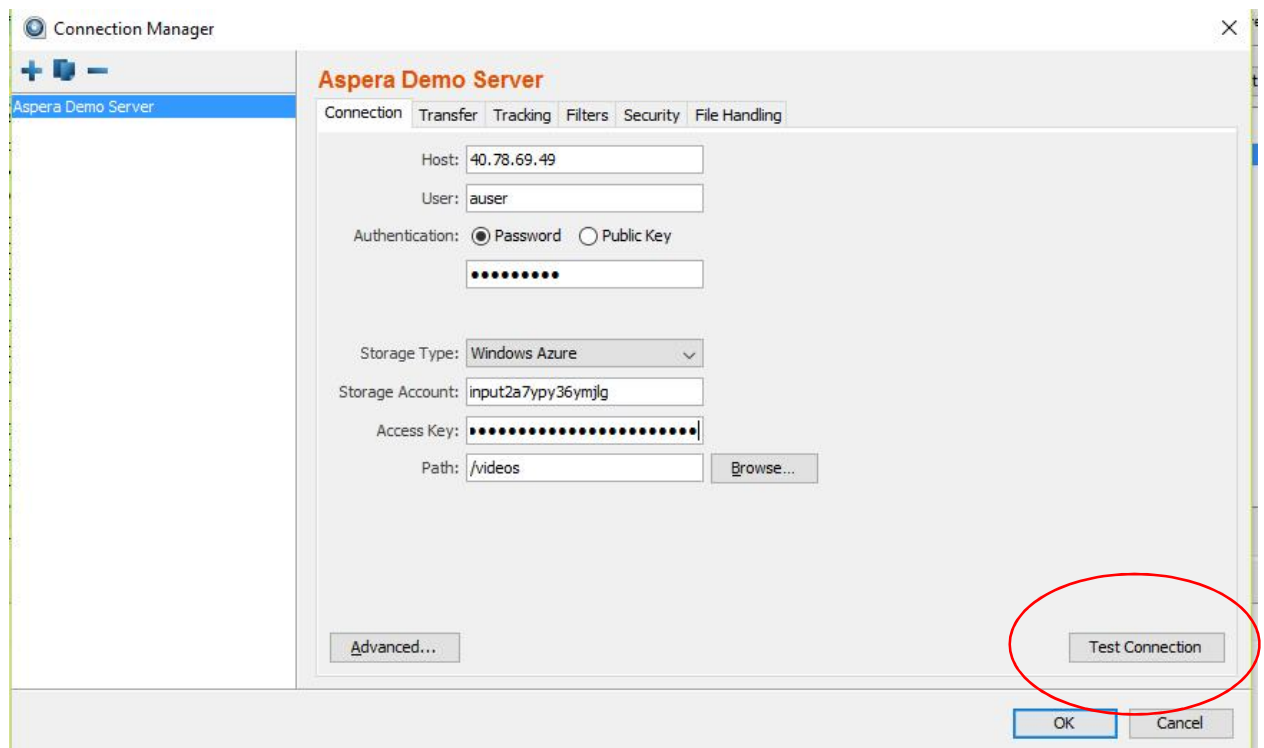


To setup Aspera client to connect to the Aspera Server, collect the Aspera server name, user id and password and azure storage details.

- Aspera Server name collected from output on azure portal.
- User id: auser

- Password : Admin@123
- Storage account and access key from the first step

The media file will land in the above storage account.



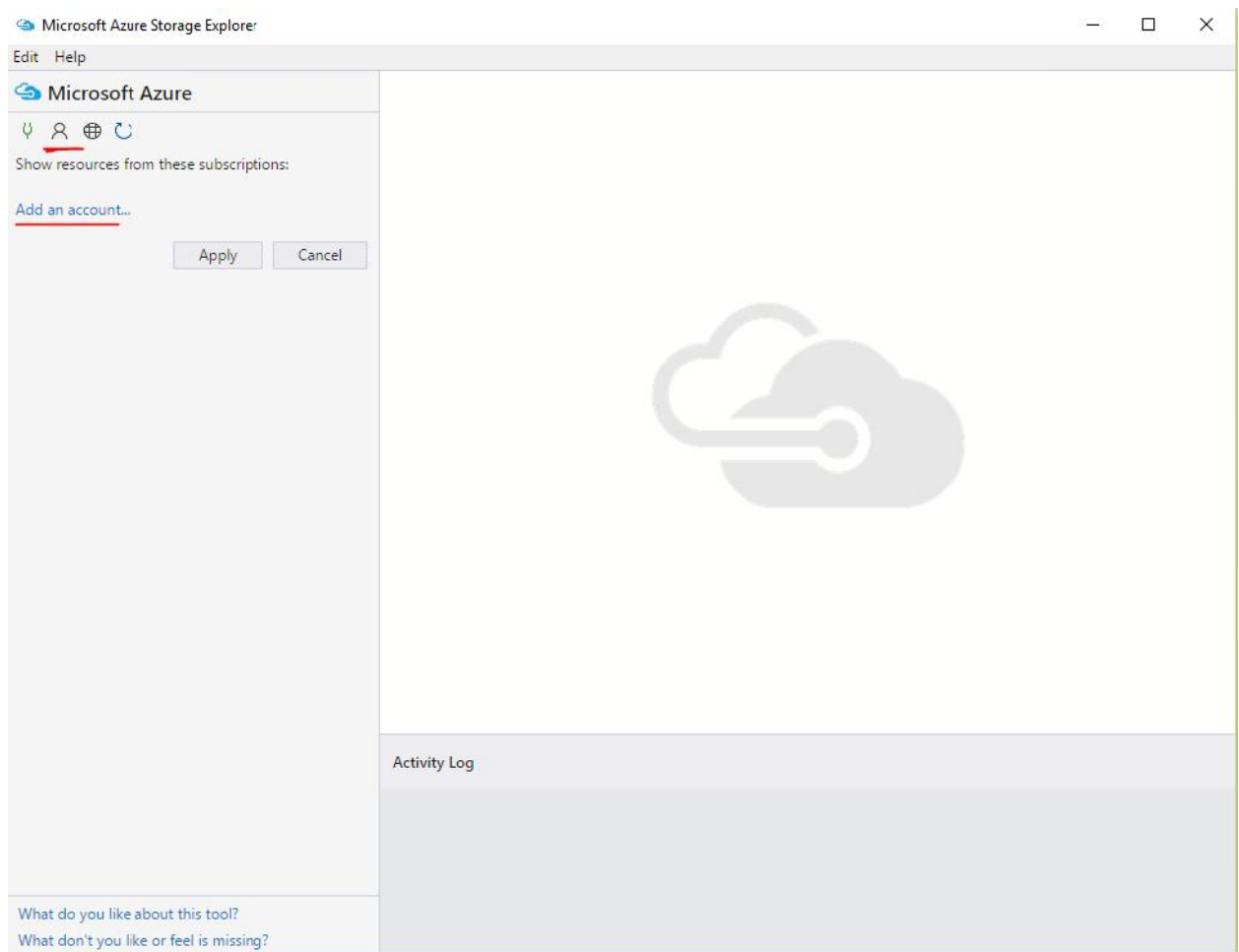
Fill in all the information and hit "Test Connection" to receive a success message.

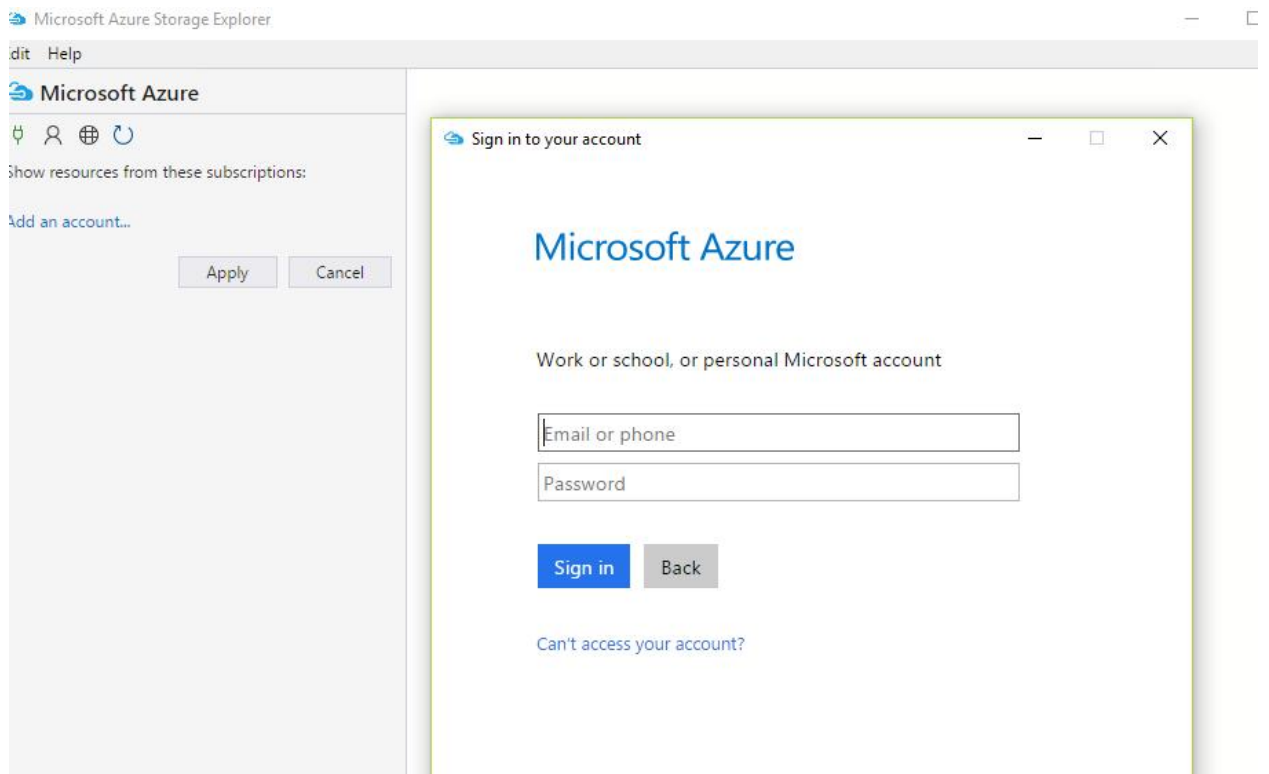
Note: Storage account here is the input storage account which starts as input\* that you have seen in the earlier section.

## 7. Open and connect to Azure storage using Storage Explorer

From the desktop in jump host open the Azure storage explorer and provide the Azure credentials which you used in the first step to login to the portal.

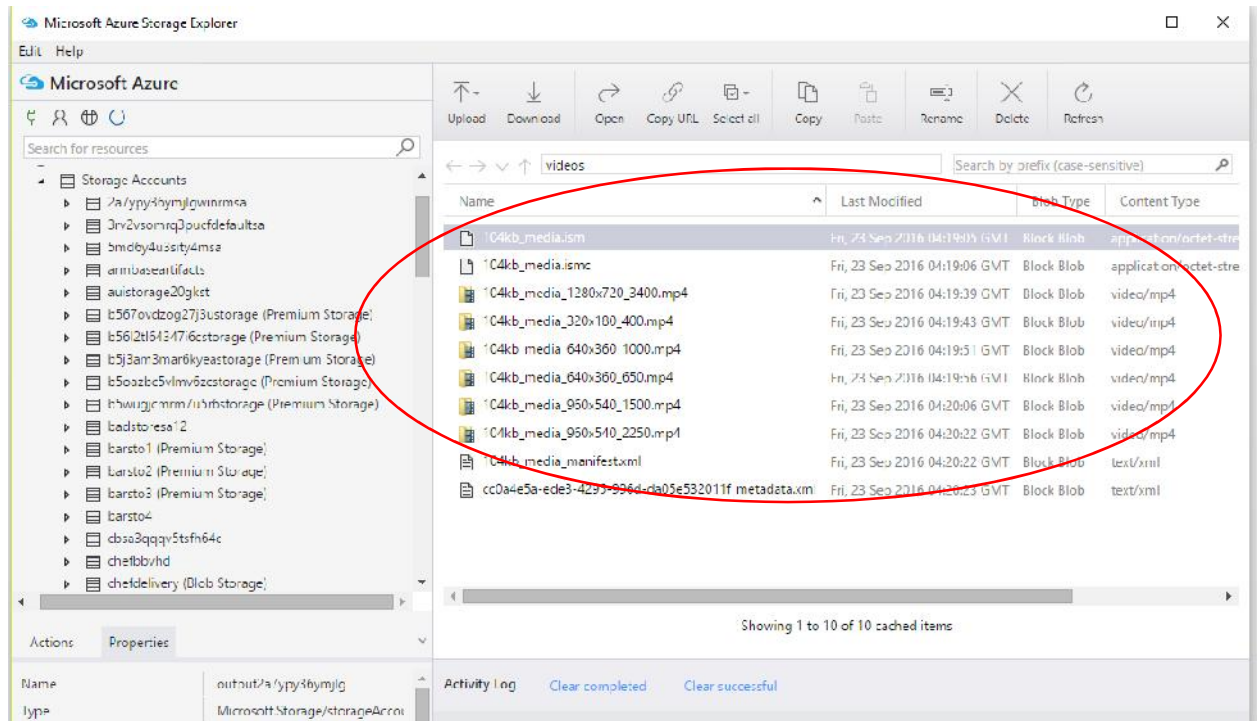
To connect to your account, click the azure account settings and provide the azure account details.





Browse to Azure storage accounts and view the input and output storage accounts which are deployed as part of your deployment.

**This below step will show you where and how the azure blobs are organized and how the media files are moving through the infrastructure.**



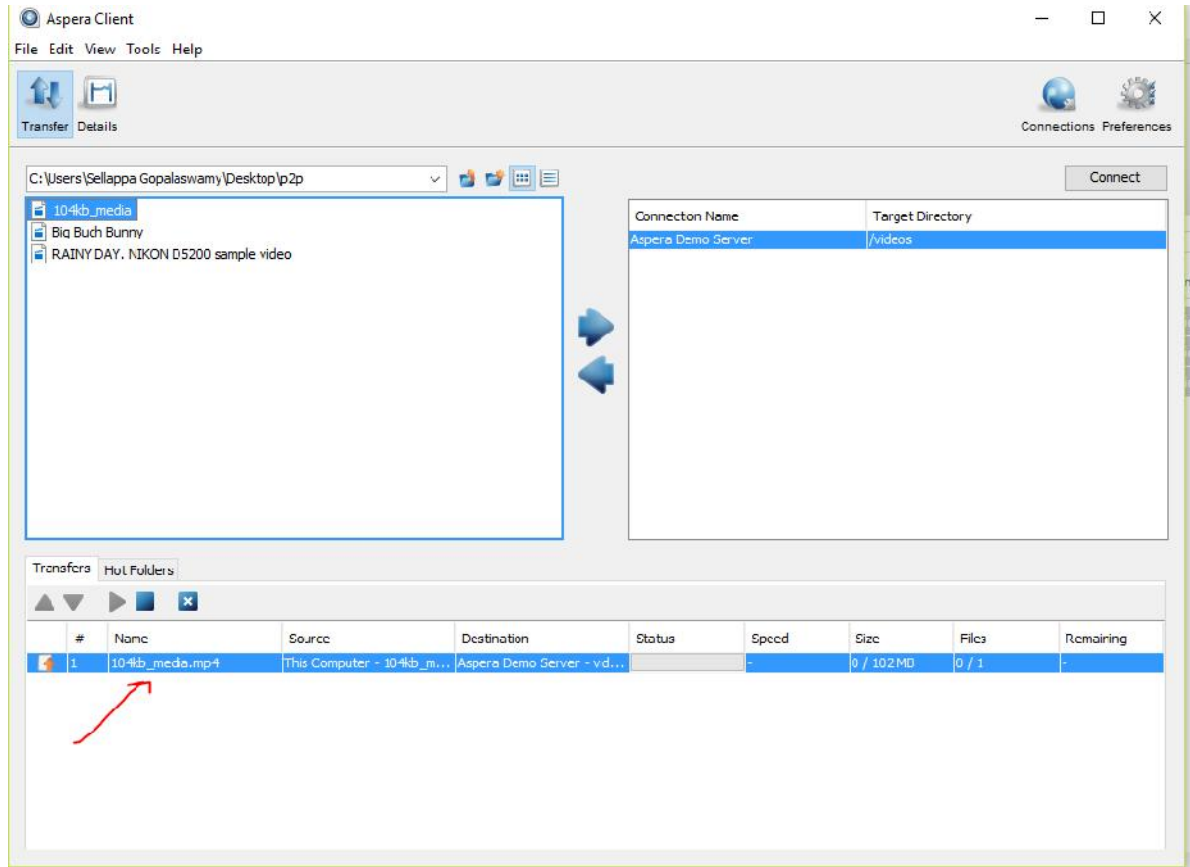
The screenshot shows the Microsoft Azure Storage Explorer interface. On the left, a list of storage accounts is visible. The main pane displays a folder named 'videos' containing several files. A red circle highlights a group of files, including two .ism files, one .ismc file, and several .mp4 files of different resolutions. The bottom pane shows the properties of the selected file.

Name	Last Modified	Block Type	Content Type
*C4kb_media.ism	Fri, 23 Sep 2016 04:19:05 GMT	Block Blob	application/octet-stream
*C4kb_media.ismc	Fri, 23 Sep 2016 04:19:06 GMT	Block Blob	application/octet-stream
*C4kb_media_1280x720_3400.mp4	Fri, 23 Sep 2016 04:19:39 GMT	Block Blob	video/mp4
*C4kb_media_320x100_400.mp4	Fri, 23 Sep 2016 04:19:43 GMT	Block Blob	video/mp4
*C4kb_media_640x360_1000.mp4	Fri, 23 Sep 2016 04:19:51 GMT	Block Blob	video/mp4
*C4kb_media_640x360_650.mp4	Fri, 23 Sep 2016 04:19:56 GMT	Block Blob	video/mp4
*C4kb_media_950x540_1500.mp4	Fri, 23 Sep 2016 04:20:06 GMT	Block Blob	video/mp4
*C4kb_media_950x540_2250.mp4	Fri, 23 Sep 2016 04:20:22 GMT	Block Blob	video/mp4
*C4kb_media_manifest.xml	Fri, 23 Sep 2016 04:20:22 GMT	Block Blob	text/xml
cc0ae5a-ede3-4293-996d-d805e532011f metadata.xml	Fri, 23 Sep 2016 04:20:23 GMT	Block Blob	text/xml

Showing 1 to 10 of 10 cached items

Activity Log: Clear completed, Clear successful

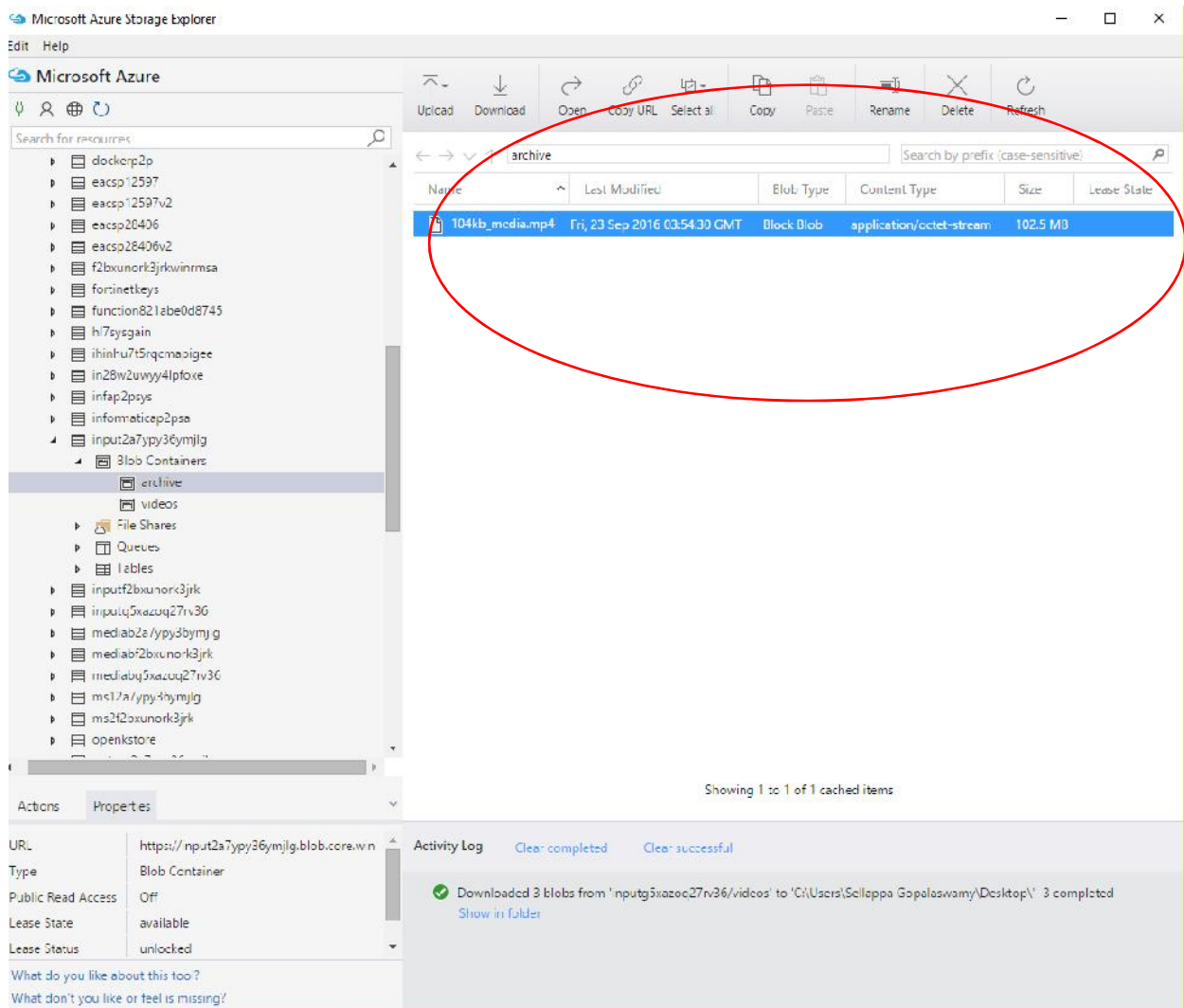
## 8. Upload a video using the Aspera client



From the left panel, **select desktop and select the mp4 file** and then press right arrow to transfer the file to the Aspera server. The status of the transfer can be seen in the bottom window.

## 9. Review media file(s) are in Azure Blobs

Check the input storage account and you will see the file transferred to the Azure Blob storage. Aspera will transfer large amounts of data into Azure very quickly, which is a typical use-case scenario in enterprises.



The screenshot displays the Microsoft Azure Storage Explorer interface. On the left, the 'archive' container is selected under the '3blob Containers' folder. The main pane shows a table of files in the 'archive' container. A single file, '104kb\_media.mp4', is listed and highlighted with a red circle. The file's properties are as follows:

Name	Last Modified	Blob Type	Content Type	Size	Lease State
104kb_media.mp4	Fri, 23 Sep 2016 03:54:30 GMT	Block Blob	application/octet-stream	102.5 MB	

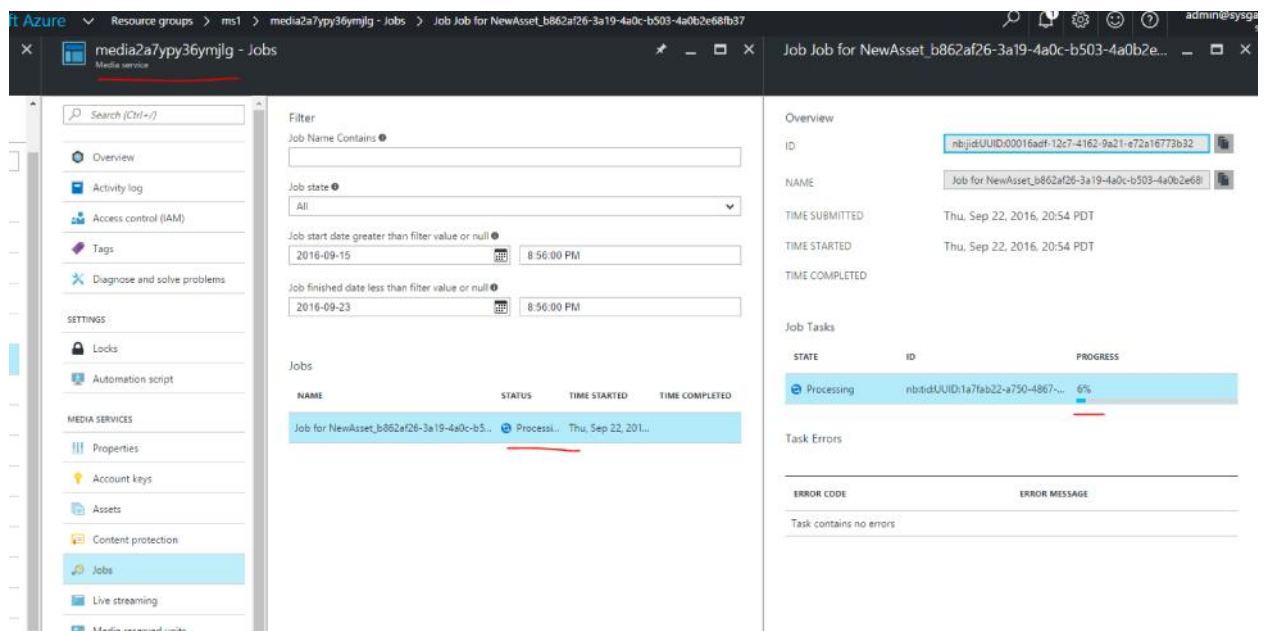
Below the file list, the 'Properties' tab is active, showing the following details:

- URL: https://input2a7yp36ymjlg.blob.core.wi...
- Type: Blob Container
- Public Read Access: Off
- Lease State: available
- Lease Status: unlocked

The Activity Log at the bottom indicates a successful download of 3 blobs from the 'input2a7yp36ymjlg/videos' container to the local desktop.

## 10. Review Azure media services job

The server job deployed as part of the solution stack and running in the background will up the file from input blob, create assets and run **"H264 Multiple Bitrate 720p"** transcoding process which will generate medial files in different formats.



The screenshot shows the Azure portal interface for Media Services. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, SETTINGS (Locks, Automation script), MEDIA SERVICES (Properties, Account keys, Assets, Content protection, Jobs, Live streaming, Machine parameters), and Machine parameters. The main area displays the 'Jobs' page for a specific job. The 'Filter' section allows filtering by Job Name, Job state (All), Job start date, and Job finished date. The 'Jobs' table shows a single job in progress. The right sidebar provides an 'Overview' of the job, including its ID, Name, Time Submitted, Time Started, and Time Completed. Below the overview, the 'Job Tasks' section shows a single task in progress with a progress bar. The 'Task Errors' section is empty, indicating no errors.

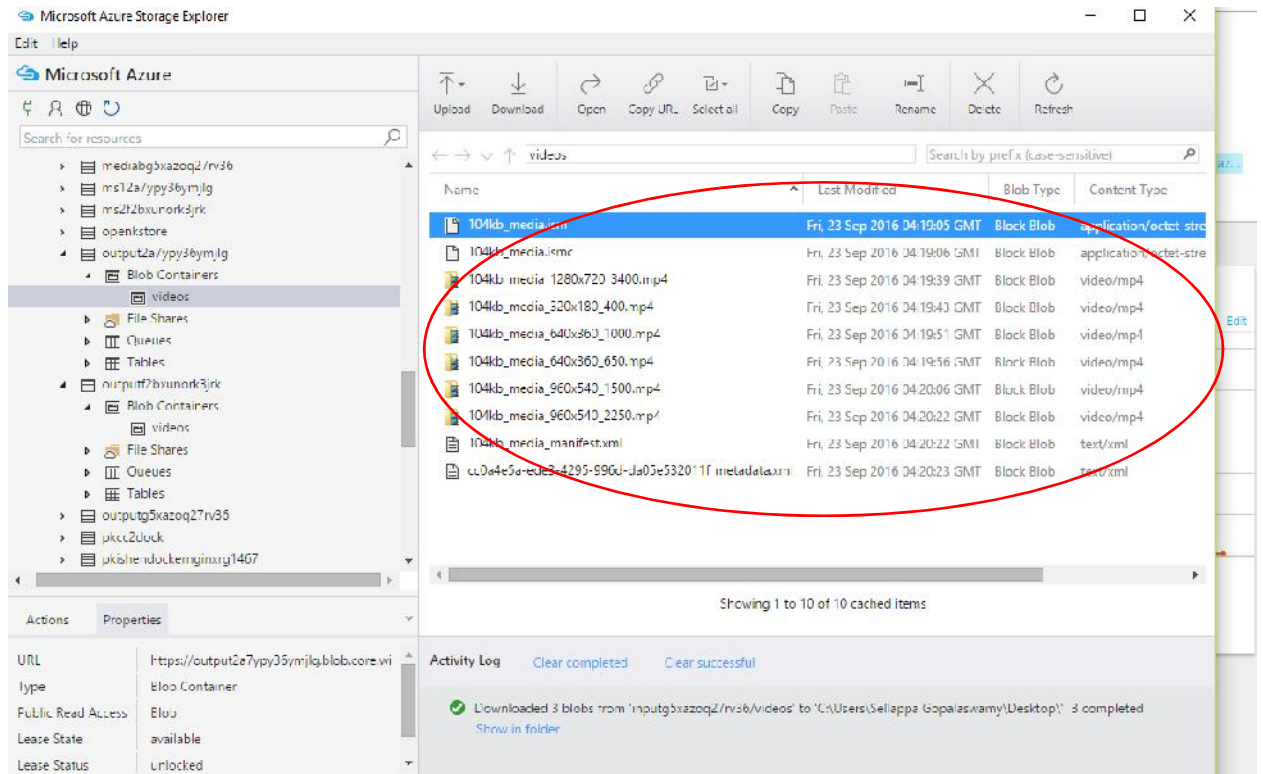
NAME	STATUS	TIME STARTED	TIME COMPLETED
Job for NewAsset_b862af26-3a19-4a0c-b503-4a0b2e68fb37	Processing	Thu, Sep 22, 2016	

Once completed, check the output Blob using the Azure Storage Explorer as shown below.

### Example outputs:

nature\_1280x720\_3400.mp4  
nature\_320x180\_400.mp4  
nature\_640x360\_1000.mp4  
nature\_640x360\_650.mp4  
nature\_960x540\_1500.mp4  
nature\_960x540\_2250.mp4





The screenshot shows the Microsoft Azure Storage Explorer interface. The left pane displays the hierarchy of storage resources, with the 'videos' container selected under a specific Blob Container. The main pane shows a list of files in the 'videos' container. A red circle highlights the first six items in the list.

Name	Last Modified	Blob Type	Content Type
104kb_media_10m	Fri, 23 Sep 2016 04:19:05 GMT	Block Blob	application/octet-stream
104kb_media_10m	Fri, 23 Sep 2016 04:19:06 GMT	Block Blob	application/octet-stream
104kb_media_1280x720_3400.mp4	Fri, 23 Sep 2016 04:19:39 GMT	Block Blob	video/mp4
104kb_media_320x180_400.mp4	Fri, 23 Sep 2016 04:19:43 GMT	Block Blob	video/mp4
104kb_media_640x360_1000.mp4	Fri, 23 Sep 2016 04:19:51 GMT	Block Blob	video/mp4
104kb_media_640x360_650.mp4	Fri, 23 Sep 2016 04:19:56 GMT	Block Blob	video/mp4
104kb_media_960x540_1500.mp4	Fri, 23 Sep 2016 04:20:06 GMT	Block Blob	video/mp4
104kb_media_960x540_2250.mp4	Fri, 23 Sep 2016 04:20:22 GMT	Block Blob	video/mp4
104kb_media_manifest.xml	Fri, 23 Sep 2016 04:20:22 GMT	Block Blob	text/xml
cc0e4e5a-e0e3-4295-596d-3a03e532011f_metadata.xml	Fri, 23 Sep 2016 04:20:23 GMT	Block Blob	text/xml

Showing 1 to 10 of 10 cached items

Activity Log: Clear completed, Clear successful

Downloaded 3 blobs from 'inputg5xazozq2/rv36/videos' to 'C:\Users\Sellappa Gopalaswamy\Desktop'. 3 completed. [Show in folder](#)

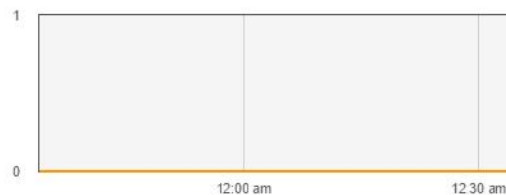
## 11. Stream the video through Wowza

Thus far, you have seen how the solution stack is used to upload media files using Aspera and transcode the file using Azure Media Services. In the next section we will use the Wowza portal to test the transcoded videos by playing them.

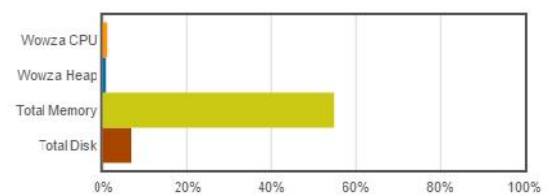
In the Wowza portal window click on “Test Players” button as shown below.

### Status

#### Connections Incoming and outgoing



#### Usage CPU, Memory, Heap and Disk



#### Server Uptime

Since 22 Sep 2016 04:48:17 PM

#### Features

Transcoder: **Not Licensed** [Learn more](#)  
 DRM: **Not Licensed** [Learn more](#)  
 nDVR: **Not Licensed** [Learn more](#)

### Test Video

To play a video on demand test video, click **Test Players**.

[▶ Test Players...](#)


In the final step, in the server section provide wowza server information and in the media file name provide one of the file names which was transcoded from the list below and click start. The wowza streaming player would play the transcoded file.

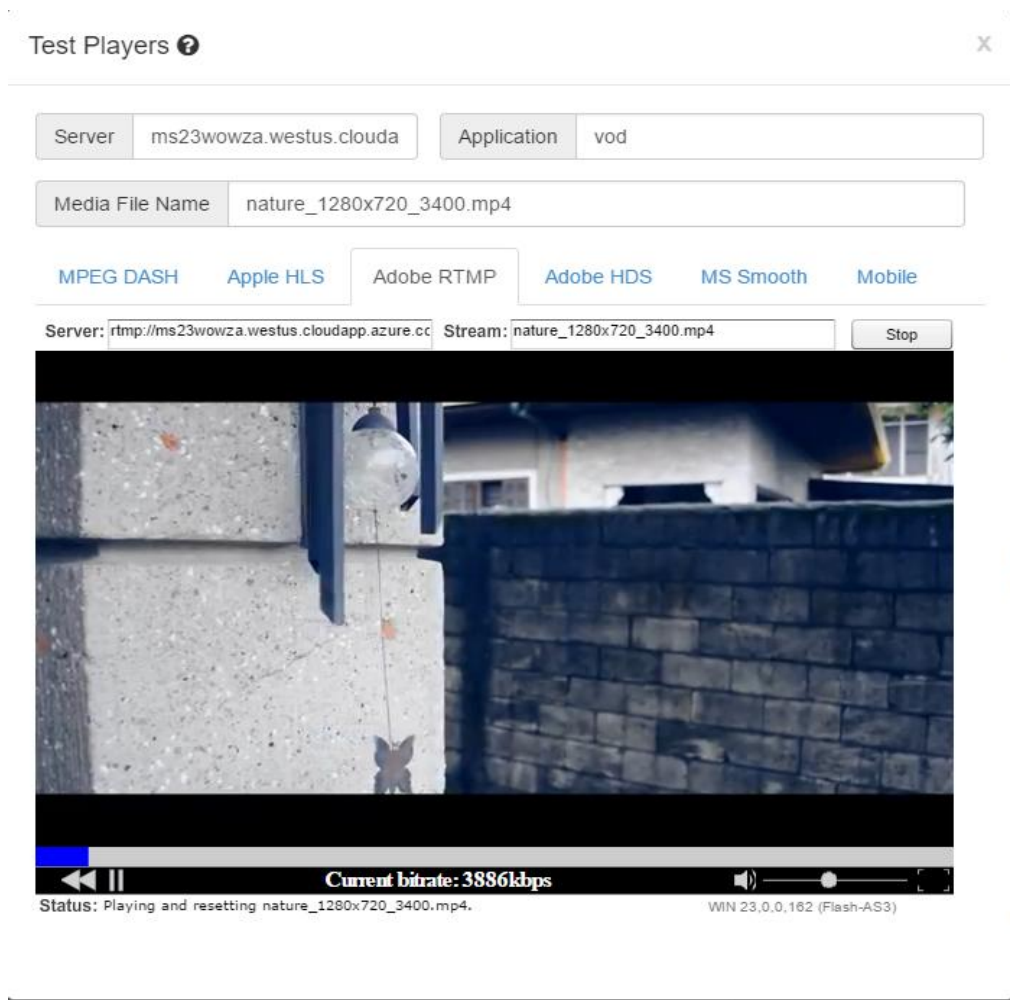
nature\_1280x720\_3400.mp4  
 nature\_320x180\_400.mp4  
 nature\_640x360\_1000.mp4  
 nature\_640x360\_650.mp4

nature\_960x540\_1500.mp4

nature\_960x540\_2250.mp4

## Test Players ?

Server	ms23wowza.westus.clouda	Application	vod
Media File Name	nature_1280x720_3400.mp4		
<div>MPEG DASH   Apple HLS   Adobe RTMP   Adobe HDS   MS Smooth   Mobile</div>			
Server:	rtmp://ms23wowza.westus.cloudapp.azure.cc		Stream: nature_1280x720_3400.mp4
			<b>Start</b>
			
Status:		WIN 23.0.0.162 (Flash-AS3)	



This is the end of this hands-on lab. Hope you gained more knowledge about Azure Quickstarts and the Media Service Solution Stack. If you have questions or comments, please reach out to us at [support@sysgain.com](mailto:support@sysgain.com).

Thank you.