## Lab 2 - Accessing and Using Azure Storage

* Learning Objectives:
  1. Understand how to access Azure Storage for your container using HDFS CLI
  2. Ingress sample data into HDFS from local filesystem
* Prerequisites:
  1. Create HDInsight Cluster as per *Lab 1*
  2. [cURL](https://curl.haxx.se/) - used to communicate with web-based services
     + sudo apt-get install libcurl4-openssl-dev
  3. [jq](https://stedolan.github.io/jq/) - used to parse JSON documents
     + sudo apt-get install jq

### Querying Ambari REST API using curl

The fully qualified domain name (FQDN) to use when connecting to the cluster from the internet is **<clustername>.azurehdinsight.net** or (for SSH only) **<clustername-ssh>.azurehdinsight.net**.

Internally, each node in the cluster has a name that is assigned during cluster configuration. To find the cluster names, you can visit the **Hosts** page on the Ambari Web UI, or use the following to return a list of hosts from the Ambari REST API:

curl -u admin:PASSWORD -G "https://CLUSTERNAME.azurehdinsight.net/api/v1/clusters/CLUSTERNAME/hosts" | jq '.items[].Hosts.host\_name'

Replace **PASSWORD** with the password of the admin account, and **CLUSTERNAME** with the name of your cluster. This will return a JSON document that contains a list of the hosts in the cluster, then jq pulls out the host\_name element value for each host in the cluster.

If you need to find the name of the node for a specific service, you can query Ambari for that component. For example, to find the hosts for the HDFS name node, use the following.

curl -u admin:PASSWORD -G "https://CLUSTERNAME.azurehdinsight.net/api/v1/clusters/CLUSTERNAME/services/HDFS/components/NAMENODE" | jq '.host\_components[].HostRoles.host\_name'

This returns a JSON document describing the service, and then jq pulls out only the host\_name value for the hosts.

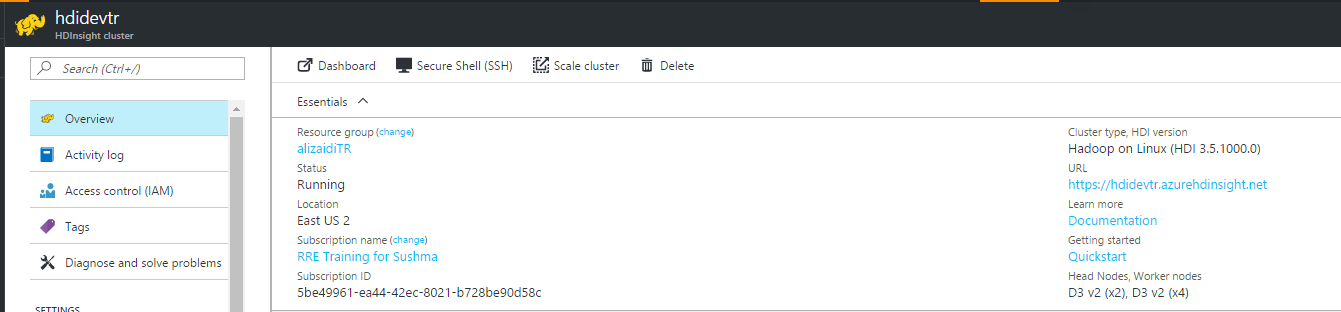
### Quering HDFS with HDFS CLI

Hadoop-related files can be found on the cluster nodes at /usr/hdp. This directory contains the following subdirectories:

1. SSH into your cluster:

ssh <username>@<clustername>-ssh.azurehdinsight.net

Can retrieve this by going to your cluster's dashboard on the Azure portal and clicking on the *Secure Shell (SSH)* button as well:



1. Examine your cluster's hdfs storage:

hdfs dfs -ls /

or equivalently, specify

hdfs dfs -ls adl://<adla\_account>.azuredatalakestore.net/

If you're using Azure blob storage as the primary storage for your cluster, then

hdfs dfs -ls wasb://<container\_name>@<stroage\_account\_name>.blob.core.windows.net/

Using the "adl://" or "wasb://" syntax is very useful when you're trying to query storage accounts that are not attached to your cluster but you have read access to.

1. Examine your local filesystem:

ls -lh

or alternatively,

hdfs dfs -ls file:///

The latter might be useful if you're working with an application that defaults to the bin/hdfs script rather than builtin bash, or if you're writing scripts programatically.

1. Make a new directory:

hdfs dfs -mkdir /TRLabs

1. Make sure it's there

hdfs dfs -ls /

1. Download some data online:

wget https://alizaidi.blob.core.windows.net/training/sample\_taxi.csv

1. Move it your new TRLabs folder:

hdfs dfs -copyFromLocal sample\_taxi.sv /TRLabs/

1. Check the contents of the TRLabs directory

hdfs dfs -ls /TRLabs/

#### HDFS vs Hadoop Shell Commands

hadoop fs and hadoop dfs shell commands have been deprecated in favor of hdfs shell commands since Hadoop 1.0. However, you may still use hadoop fs shell commands to query your cluster's storage container: hadoop fs -ls /example/data

### What Blob storage is the cluster using?

During cluster creation, you selected to either use an existing Azure Storage account and container, or create a new one. Then, you probably forgot about it. You can find the default storage account and container by using the Ambari REST API.

1. Use the following command to retrieve HDFS configuration information using curl, and filter it using [jq](https://stedolan.github.io/jq/):

* curl -u admin:PASSWORD -G "https://CLUSTERNAME.azurehdinsight.net/api/v1/clusters/CLUSTERNAME/configurations/service\_config\_versions?service\_name=HDFS&service\_config\_version=1" | jq '.items[].configurations[].properties["fs.defaultFS"] | select(. != null)'