Leveraging Unstructured Data - Lab 1 : Creating a Dataproc Cluster v1.3

## Overview

Structured data has a useful organization or schema. Unstructured data includes not only data that is without a schema, but also data that has some structure, but that structure is not useful for the intended analysis or query.

In this lab, you will create a Dataproc cluster and explore Hadoop operations. The cluster is customized for secure access, uses a bucket for initialization, and customized to use the Google Cloud AP

## Objectives

In this lab, you will perform the following tasks:

* Prepare a bucket for cluster initialization
* Create a Dataproc Hadoop Cluster customized to use the Google Cloud API
* Enable secure access to the Dataproc cluster
* Explore Hadoop operations

## Task 1. Prepare Environment Variables

1. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **Compute Engine** > **VM instances**.
2. Locate the line with the instance called **training\_vm**.
3. On the far right, under 'connect', click on **SSH** to open a terminal window.
4. In this lab you will enter CLI commands on the **training\_vm**.

Why isn't Cloud Shell used in this lab?

Unlike Compute Engine, Cloud Shell has no SLA. Therefore, the availability of the Cloud Shell VM cannot be guaranteed during the lab.

<https://cloud.google.com/shell/sla>

<https://cloud.google.com/compute/sla>

The activities you will perform in this lab will require a number of values, such as the name of a Cloud Storage bucket for staging, hosting, and output files, and the IP address of your computer/browser needed for Firewall Rules.

As a best practice, you will set the environment variables in a file, **myenv**, where you can easily recover them if needed using the command: **source myenv**

### **Create the source file for setting and resetting environment variables**

1. In the **training\_vm** SSH terminal window, using your preferred command line editor, create and edit the file to hold your environment variables. For example:
2. cd ~
3. nano myenv

### **Identify a project**

One environment variable that you will set is \_\_$PROJECT\_ID\_\_ that contains the Google Cloud project ID required to access billable resources.

1. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **Home**. In the panel with Project Info, the **Project ID** is listed. You can also find this information in the Qwiklabs tab under Connection Details, where it is labeled **GCP Project ID**.
2. Add the environment variable to **myenv** for easy reference.
3. PROJECT\_ID=<project ID>

### **Verify a bucket for Dataproc staging**

Dataproc can use a Cloud Storage bucket to stage its files during initialization. You can use this bucket to stage application programs or data for use by Dataproc. The bucket can also host Dataproc initialization scripts and output. The bucket name must be globally unique. Qwiklabs has already created a bucket for you that has the same name as the Project ID, which is already globally unique.

1. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **Storage > Browser**. Verify that the bucket exists. Notice the default storage class and the location (region) of this bucket. You will be using this region information next.
2. Add the line to **myenv** to create an environment variable named "BUCKET".
3. BUCKET=<project ID>

You can use $BUCKET in CLI commands. And if you need to enter the bucket name <your-bucket> in a text field in Console, you can quickly retrieve the name with **echo $BUCKET**.

### **Identify a region and zone**

You will be creating a Dataproc cluster in a specific region. The Dataproc cluster and the bucket it will use for staging must be in the same region. Since the bucket you are using already exists, you will need to match the environment variable **$MYREGION** to the bucket region.

1. You can find the region on the Qwiklabs tab under Connection Details, labeled **QL Region**.

The zone must be in the same region.

1. You can find the zone on the Qwiklabs tab under Connection Details, labeled **QL Zone**.
2. Add the environment variables to **myenv** for easy reference.
3. MYREGION=<region>
4. MYZONE=<zone>

### **Identify the browser IP address**

You will use the browser IP address to enable your local browser to reach the Dataproc cluster.

1. Find your computer's browser IP address by opening a browser window and viewing <http://ip4.me/> Copy the IP address.
2. Add the line to **myenv** to create an environment variable named **BROWSER\_IP**.
3. BROWSER\_IP=<your-browser-ip>
4. After you have added all three definitions to **myenv**, and saved the file, use the source command to create the environment variables.

### **Set and verify environment variables**

1. Set the environment variables.
2. source myenv
3. Verify the values are set with echo.
4. echo $PROJECT\_ID
5. echo $MYREGION $MYZONE
6. echo $BUCKET
7. echo $BROWSER\_IP

You can use the environment variables in CLI commands and scripts. However, Console does not use the environment variables. For example, if a text box in Console required a bucket name, you could use echo $BUCKET to recall the value, but you would need to copy and paste it into the text box in Console. If you entered $BUCKET into the text box it would be used literally and generate an error.

## Task 2. Create a Dataproc Cluster

### **Create a Dataproc Hadoop Cluster customized to use the Google Cloud API**

This lab makes several changes from the default values. It uses one master and three worker nodes. The master is an n1-standard-2 node with 100 GB storage, and the worker nodes are n1-standard-1 instances with 50 GB of storage on each node. In production you would select vCPUs and data sizes to reflect your application requirements. Generally, more resources produce results faster and fewer resources cost less.

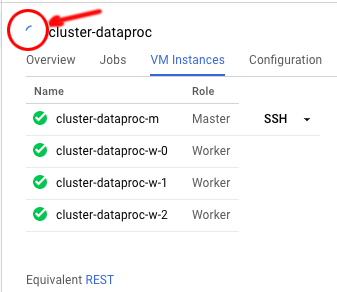
1. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **Dataproc > Clusters**.
2. Click **Create Cluster**.
3. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | cluster-dataproc |
| **Region** | <your region> |
| **Zone** | <your zone> |
| **Cluster mode** | Standard (1 Master, N workers) |
| **(Master node) Machine type** | n1-standard-2 |
| **(Master node) Primary disk size** | 100 GB |
| **(Worker nodes) Machine types** | n1-standard-1 |
| **(Worker nodes) Primary disk size** | 50 GB |
| **Nodes (minimum 2)** | 3 |

1. Click on **Advanced options**.
2. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Network tags** | hadoopaccess |
| **Cloud Storage staging bucket** | <your-bucket> |
| **Image** | 1.3 (Default) |
| **Project access** | [x] Allow API access to all Google Cloud services in the same project |

1. Click **Create**.
2. The cluster will take several minutes to become operational. In the Console, on the **Navigation menu** (7a91d354499ac9f1.png) click **Dataproc > Clusters**.
3. Click on your cluster, **cluster-dataproc**. Then click on the **VM Instances** tab. The instances will become operational before the hadoop software has completed initialization. When a checkmark in a green circle appears next to the name of the cluster, it is operational.



1. It will take a few minutes longer for the initialization to complete.

## Task 3. Enable secure access to the Dataproc cluster

### **Create a restrictive firewall rule using Target tags, IP address, and protocol**

Create a firewall rule that allows access only to the Master Node from your computer's IP address. Only ports 8088 (Hadoop Job Interface) and 9870 (Hadoop Admin interface) will be permitted.

Port 8042 is the web UI for the node manager on the worker nodes and port 8080 is the default port for Datalab. Datalab is a notebook-based integrated development environment derived from Jupyter notebooks. It is a common tool for developing Dataproc applications. The Serverless Machine Learning on GCP course uses Datalab extensively.

1. Recall your computer's browser IP address for use in Console.
2. echo $BROWSER\_IP
3. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **VPC Network** > **Firewall rules**.
4. Click **Create Firewall Rule**.
5. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | allow-hadoop |
| **Network** | default |
| **Priority** | 1000 |
| **Direction of traffic** | Ingress |
| **Action on match** | Allow |
| **Targets** | Specified target tags |
| **Target tags** | hadoopaccess |
| **Source IP ranges** | <your-IP>/32 |
| **Specified protocols and ports** | Check tcp and enter port number 9870,8088 |

1. Click **Create**.

It will take a few minutes for the firewall rule to become active.

### **Verify that the network tag is set on the Master Node**

Verify that the network tag "hadoopaccess" is set on the Master Node. That will apply the firewall rule to the Master Node, giving your laptop access to it.

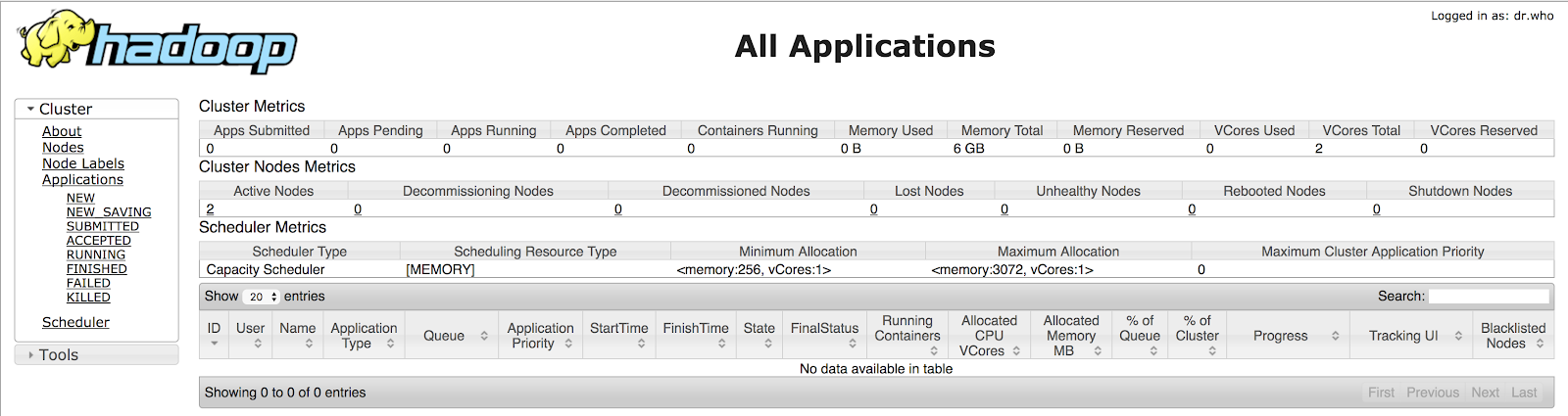
1. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **Compute Engine** > **VM instances**.
2. Click on the Master Node, **cluster-dataproc-m**.
3. Verify that under Network Tags it lists **hadoopaccess**.
4. If the tag is not there, click **EDIT**.
5. Under Network Tags add the tag: **hadoopaccess**
6. Click **Save**.

## Task 4. Explore Hadoop operations

### **Browse to Master Node**

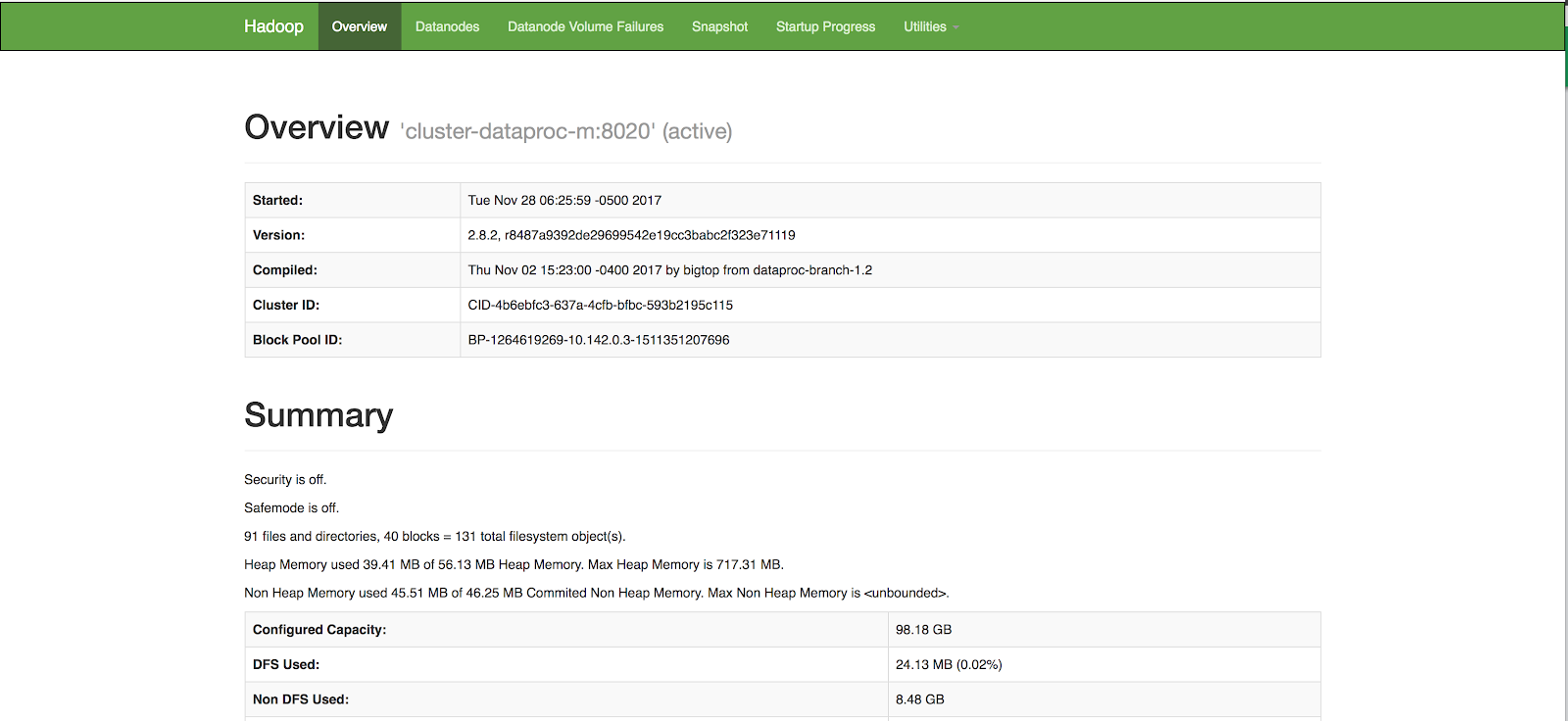
In this task, you will explore built-in Hadoop operation and administration services. This task and the next two will help you understand the relationship between Dataproc in the console, the Master Node instance, and Hadoop services. If you have worked with Hadoop before, the web UIs will be familiar.

1. In the Console, on the **Navigation menu** ( 7a91d354499ac9f1.png) click **Compute Engine** > **VM Instances**.
2. In the list of VM instances, in the row for **cluster-dataproc-m**, highlight the **External IP** and copy it.
3. Open a new browser tab or window and paste the External IP. Add ":8088" after the IP and press enter. Example: <External IP>:8088 The web page displayed is the Hadoop Applications interface and should look something like this:



You can view all jobs submitted to the cluster and their status from this page.

1. Open a new browser tab or window. Paste the External IP. Add ":9870" after the IP and press return. Example: <External IP>:9870 The webpage displayed is the Hadoop Administration Interface and should look something like this:



1. Click on the **Datanodes** tab. This will show you how much capacity is being used on the worker nodes HDFS (Hadoop Distributed File System) and how much capacity remains.
2. Click on **Utilities > Logs**. This shows you the Hadoop log files for each node in the cluster. This is where you can go to investigate problems with Hadoop. Use your browser's back button to return to the Hadoop Administration console.
3. Click on **Utilities > Browse the file system**. After a few moments the file system will appear in the browser page. You can use this to navigate the files system. In the row that says Owner is **hdfs** and Group is **hadoop**, click on the link that says **user**. Here you can see directories for all the hadoop services.