



Hadoop Architecture Labs

Lab 1: Examining Nodes On The Cluster

At this point, YARN and the Hadoop Distributed File System (HDFS) are already running on the control node of your cluster. Both of these services have command line and Web interfaces. Let's look at both.

First, connect to the command line on your edge node using `ssh`. However, because your virtual machine is running under vagrant, you'll use the `vagrant ssh` command instead of ordinary `ssh`.

```
vagrant ssh edge
```

You should see output that looks like the following:

```
Last login: Sun Aug 20 18:06:02 2017 from 10.0.2.2
[vagrant@edge ~]$
```

From this point on, we'll simply show the commands we type (as indicated by the `[vagrant@edge vagrant]$` prompt) and the result in the same code block. See what version of yarn you are running using the command `yarn version` and then `yarn application -list` to list all the applications.

```
[vagrant@edge ~]$ yarn version
Hadoop 2.6.0-cdh5.4.2
Subversion http://github.com/cloudera/hadoop -r 15b703c8725733b7b2813d2325659eb7d57e7a3f
Compiled by jenkins on 2015-05-20T00:03Z
Compiled with protoc 2.5.0
From source with checksum de74f1adb3744f8ee85d9a5b98f90d
This command was run using /usr/lib/hadoop/hadoop-common-2.6.0-cdh5.4.2.jar
[vagrant@edge ~]$ yarn application -list
17/08/22 17:42:51 INFO client.RMProxy: Connecting to ResourceManager at control/192.168.33.10:8020
Total number of applications (application-types: [] and states: [SUBMITTED, ACCEPTED, RUNNING])
Application-Id      Application-Name      Application-Type      User
[vagrant@edge ~]$
```

We can get help with all the yarn commands by typing `yarn -help`

```
[vagrant@edge ~]$ yarn -help
Usage: yarn [--config confdir] COMMAND
where COMMAND is one of:
    resourcemanager -format-state-store  deletes the RMStateStore
    resourcemanager                        run the ResourceManager
    nodemanager                          run a nodemanager on each slave
    timelineserver                       run the timeline server
    rmadmin                             admin tools
    version                             print the version
    jar <jar>                           run a jar file
    application                          prints application(s)
                                         report/kill application
    applicationattempt                   prints applicationattempt(s)
                                         report
    container                           prints container(s) report
    node                                prints node report(s)
    queue                               prints queue information
    logs                                dump container logs
    classpath                           prints the class path needed to
                                         get the Hadoop jar and the
                                         required libraries
    daemonlog                           get/set the log level for each
                                         daemon

or
    CLASSNAME                           run the class named CLASSNAME
Most commands print help when invoked w/o parameters.
[vagrant@edge ~]$
```


We can also list the nodes in our cluster with `yarn node -list -all` and query the status of a node using `yarn node -status nodename` as shown below.

```
[vagrant@edge ~]$ yarn node -list -all
17/08/23 15:42:17 INFO client.RMProxy: Connecting to ResourceManager at control/192.168.33.10:8088
Total Nodes:1
      Node-Id      Node-State Node-Http-Address  Number-of-Running-Containers
control:44668      RUNNING    control:8042              0
[vagrant@edge ~]$ yarn node -status control:44668
17/08/23 15:42:35 INFO client.RMProxy: Connecting to ResourceManager at control/192.168.33.10:8088
Node Report :
  Node-Id : control:44668
  Rack : /default-rack
  Node-State : RUNNING
  Node-Http-Address : control:8042
  Last-Health-Update : Wed 23/Aug/17 03:40:52:695UTC
  Health-Report :
  Containers : 0
  Memory-Used : 0MB
  Memory-Capacity : 4096MB
  CPU-Used : 0 vcores
  CPU-Capacity : 4 vcores
  Node-Labels :
```

[vagrant@edge ~]\$

Now we'll look at the same information using the Web interface by going to `http://192.168.33.10:8088` or `http://control:8088` . We reference the control node because that's where the Resource Manager is located. In fact, the `yarn` commands we issued above connected to the control node to get their results.

This is what your output should look like on a brand new cluster.



Cluster Metrics

Cluster

About Nodes ApplicationsNEW NEW SAVING SUBMITTED ACCEPTED RUNNING FINISHED FAILED KILLED Scheduler

Tools

Apps		Apps		Apps		Apps		Containers	Memory	Memory	Memory	VCores	VCores	VCores	Active	Decommissioned	Lost	Unhealthy	Rebooted
Submitted	Pending	Running	Completed	Running	Used	Total	Reserved	Used	Total	Reserved	Used	Total	Reserved	Used	Nodes	Nodes	Nodes	Nodes	Nodes
0	0	0	0	0	0 B	4 GB	0 B	0	4	0	1	0	0	0	0	0	0	0	0

User Metrics for dr.who

Apps		Apps		Apps		Containers		Containers	Containers	Memory	Memory	Memory	VCores	VCores	VCores
Submitted	Pending	Running	Completed	Running	Pending	Reserved	Used	Pending	Reserved	Used	Pending	Reserved	Used	Pending	Reserved
0	0	0	0	0	0	0	0 B	0 B	0 B	0	0	0	0	0	0

Show 20 entries

ID

User

Name

Application Type

Queue

StartTime

FinishTime

State

FinalStatus

Progress


Tracking UI

No data available in table

Showing 0 to 0 of 0 entries

First Previous Next Last

If we look under the **Cluster** menu on the left and click on **Nodes** , we should see the nodes that are running, just as we did above on the command line



Cluster

About

Nodes

Applications

NEW

NEW SAVING

SUBMITTED

ACCEPTED

RUNNING

FINISHED

FAILED

KILLED

Scheduler

Tools

Nodes of the cluster

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
0	0	0	0	0	0 B	4 GB	0 B	0	4	0	1	0	0	0	0

User Metrics for dr.who

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Containers Pending	Containers Reserved	Memory Used	Memory Pending	Memory Reserved	VCores Used	VCores Pending	VCores Reserved
0	0	0	0	0	0	0	0 B	0 B	0 B	0	0	0

Show 20 entries

Search:

Node Labels	Rack	Node State	Node Address	Node HTTP Address	Last health-update	Health-report	Containers	Mem Used	Mem Avail	VCores Used	VCores Avail	Version
/default-rack		RUNNING	control:44668	control:8042	Wed Aug 23 16:22:52 +0000 2017		0	0 B	4 GB	0	4	2.6.0-cdh5.4.2

Showing 1 to 1 of 1 entries

First Previous 1 Next Last

Finally, if we click on the entry control:8042 under **Node HTTP Address** , we'll get detailed information of our one control node:



ResourceManager

NodeManager

Node Information

List of Applications

List of Containers

Tools

NodeManager information

Total Vmem allocated for Containers8.40 GB

Vmem enforcement enabledfalse

Total Pmem allocated for Container4 GB

Pmem enforcement enabledtrue

Total VCoers allocated for Containers4

NodeHealthyStatustrue

LastNodeHealthTimeWed Aug 23 16:24:52 UTC 2017

NodeHealthReport

Node Manager Version:2.6.0-cdh5.4.2 from 15b703c8725733b7b2813d2325659eb7d57e7a3f by jenkins source checksum e7a085479aa1989b5cecfabea403549 on 2015-05-20T00:09Z

Hadoop Version:2.6.0-cdh5.4.2 from 15b703c8725733b7b2813d2325659eb7d57e7a3f by jenkins source checksum de74f1adb3744f8ee85d9a5b98f90d on 2015-05-20T00:03Z

This step concludes this lab.