Installation & Integration Guidelines
On
Spark, R, and H2O

Saurabh Chauhan

Contents

List of Figures

1 Introduction

The purpose of the document is to provide users with quick recipe on Installation & Integration of Spark, R, and H2O. The installation and integration of Spark, R, and H2O are seems to confusing for the beginner. The document provides step-by-step recipe to enable the integration of R, and H2O on top of Spark cluster/framework. Apart from the installation and integration, the document also highlights the working prototype of the components. The document targets the Ubuntu (14.04 or later version) and Linux (CentOS, Red Hat etc.) flavors operating systems for the installation and integration. The following sections describe the step-by-step installation procedure to enable the working prototype.

2 Software Installation before Installing Spark

2.1 Installation of Java

Java needs to be installed[1] before deploying Spark framework/cluster (pre-requisites for Spark). To install Java open the terminal and run the following commands:

- Update Java PPA repository in the system using the following command:
  `$ sudo add-apt-repository ppa:webupd8team/java`
- Get updated versions of available packages using the following command:
  `$ sudo apt-get update`
- Install Java version 1.8 using the following command:
  `$ sudo apt-get install oracle-java8-installer`

2.2 Verify Installation of Java Version

After successful installation of Java 8 using above steps, verify the installed version using following command:

- `$ java -version`

If it displays output similar to following than it indicates that installation procedure gets successfully completed and Java is up and running.

```
openjdk version "1.8.0_111"
OpenJDK Runtime Environment (build 1.8.0_111-8u111-b14-3 14.04.1-b14)
OpenJDK 64-Bit Server VM (build 25.111-b14, mixed mode)
```

2.3 Installation of Scala

Spark is written in Scala, so the installation of Scala is mandatory to built Spark. To install, Scala following steps need to be performed:

- Download the 2.11.4 version of the scala from `http://www.scala-lang.org/download/`
- Go to downloads folder using the following command:
  `$ cd Downloads`

[1] http://tecadmin.net/install-oracle-java-8-jdk-8-ubuntu-via-ppa/

- Extract Scala tar file using the following command:
  `$ sudo tar xvf scala-2.11.4.tgz`
- Open .bashrc file (setting up the environment for Scala) using the following command:
  `$ sudo gedit /.bashrc`
- Edit .bashrc file by adding the following path in the end of the file. The following lines add the location, where the Scala software files are located to the PATH variable:
  `export SCALA_HOME= path-where-scala-file-is-located`
  `export PATH=$SCALA_HOME/bin:$PATH`
- Source the modified .bashrc file using the following command:
  `source /.bashrc`

2.4 Verify Scala Installation

After installation of Scala using above steps, it's good to verify the version to make sure whether Scala is installed properly or not. Verify the Scala installation using the following command:

- `$ scala -version`

If output of above command display that `Scala code runner version 2.11.4 -- Copyright 2002-2013, LAMP/EPFL` means Scala is up and running in the system.

3 Installation of Spark

Install Spark in standalone mode i.e. single node cluster- To install Spark single node cluster[2], simply download Spark setup file, extract and configure it. To install and configure Spark following steps need to be performed:

- Download the 2.0.0 version of Spark pre-built for Hadoop 2.7 and later from the Apache Spark website `http://spark.apache.org/downloads.html`
- Go to downloads folder using the following command:
  `$ cd Downloads`
- Extract Spark tar file using the following command:
  `$ sudo tar xvf spark-2.0.0-bin-hadoop2.7.tgz`
- Open .bashrc file (setting up the environment for Scala) using the following command:
  `$ sudo gedit /.bashrc`
- Edit .bashrc file by adding the following path in the end of the file. The following lines add the location, where the Spark software files are located to the PATH variable:
  `export SPARK_HOME= path-where-spark-file-is-located`
  `export PATH=$SPARK_HOME/bin:$PATH`
- Source the modified .bashrc file using the following command:
  `source /.bashrc`

[2] http://data-flair.training/blogs/install-configure-run-apache-spark-2-x-single-node/

4 Installation of R and packages

To install R following steps need to be performed:

- Setting up APT (advanced packaging tool)[3]: In order to get the most recent version of R, we need to add the correct repository to the list of sources using the following command:
  `$ sudo sh -c 'echo "deb http://cran.rstudio.com/bin/linux/ubuntu trusty/" >> /etc/apt/sources.list'`
- Authenticate package downloaded using APT and add public key of CRAN using the following command:
  `$ gpg --keyserver keyserver.ubuntu.com --recv-key E084DAB9`
- Add the above key to apt using the following command:
  `$ gpg -a --export E084DAB9 | sudo apt-key add -`
- Update the list of available packages as we updated the sources list using the following command:
  `$ sudo apt-get update`
- Install R package using the following command:
  `$ sudo apt-get -y install r-base`

4.1 Verify Installation of R

To verify the installed R, open terminal and type `R`. If the R is running is properly then we can see output similar to the following:

```
R version 3.3.0 beta (2016-03-30 r70404) -- "Supposedly Educational"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform:  x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

[3] https://www.digitalocean.com/community/tutorials/how-to-set-up-r-on-ubuntu-14-04

4.2 Installation of RStudio (GUI for R)

- Go to RStudio website i.e. `https://www.rstudio.com/products/rstudio/download/`, choose and download the right package (download zip file with .deb extension) for your system
- Open this file in Ubuntu Software Center
- Click install and you're done with RStudio installation

4.3 Installation of SparkR package using RStudio from Github

The RStudio provides GUI interface to interact with R. To install SparkR package[4] using RStudio following steps need to be performed:

- Open RStudio
- Install package `devtools` to install SpakR package from Github repository using the following command:
  `if (!require('devtools')) install.packages('devtools')`
- Install SparkR package with version 2.0.0 compatible using the following command:
  `devtools::install_github('apache/spark@v2.0.0', subdir='R/pkg')`

4.4 Verify Installation of SparkR package

In RStudio type the `library(SparkR)` and if it displays output similar to the following means SparkR package is correctly installed in RStudio:

```
Attaching package: 'SparkR'
The following objects are masked from 'package:stats':
cov, filter, lag, na.omit, predict, sd, var, window
The following objects are masked from 'package:base':
as.data.frame, colnames, colnames<-, drop, endsWith, intersect, rank, rbind,
sample, startsWith, subset, summary, transform, union
```

4.5 Installation of H2O package on top of R using RStudio

- Install H2O package from CRAN[5] using the following command:
  `install.packages("h2o")`
- To verify the installation of H2O, type following commands in RStudio:
  `library(h2o)`
  `h2o.init()`
- If the output of the above commands display the following text means H2O cluster is up and running on top of R.
  `Starting H2O JVM and connecting: ..... Connection successful!`

  `R is connected to the H2O cluster:`

[4] http://stackoverflow.com/questions/31184918/installing-of-sparkr
[5] http://h2o-release.s3.amazonaws.com/h2o/rel-lambert/5/docs-website/Ruser/Rinstall.html

```
H2O cluster uptime:  6 seconds 270 milliseconds
H2O cluster version:  3.10.0.8
H2O cluster version age:  2 months and 7 days
H2O cluster name:  H2O_started_from_R_saurabh_rhf483
H2O cluster total nodes:  1
H2O cluster total memory:  5.33 GB
H2O cluster total cores:  4
H2O cluster allowed cores:  4
H2O cluster healthy:  TRUE
H2O Connection ip:  localhost
H2O Connection port:  54321
H2O Connection proxy:  NA
R Version:  R version 3.3.0 beta (2016-03-30 r70404)
```

5 Verify Spark Installation/Start Spark Services

5.1 Starting Spark Cluster

- Open the terminal and go to path where spark files are located
- Start spark master server by executing the following command:
  `$./sbin/start-master.sh`
- To verify whether Spark master is running or not, open URL `localhost:8080` and it should display the following screen (refer Figure 1):

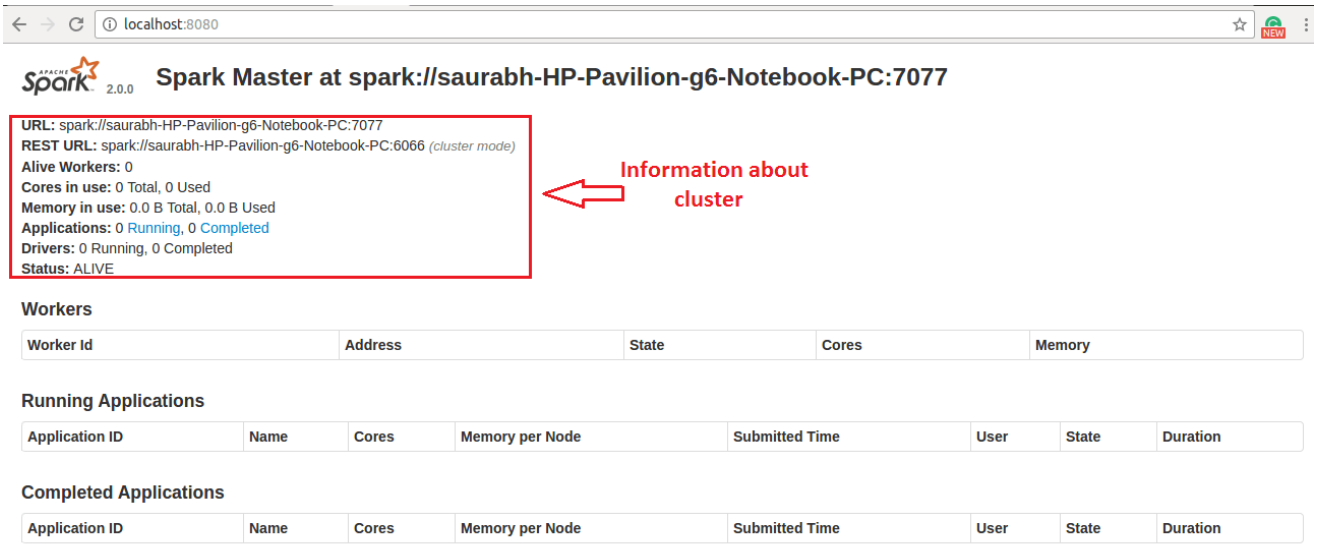

Figure 1: Spark cluster console

- Similarly, we can start one or more workers node connect them to the master node using the following command:

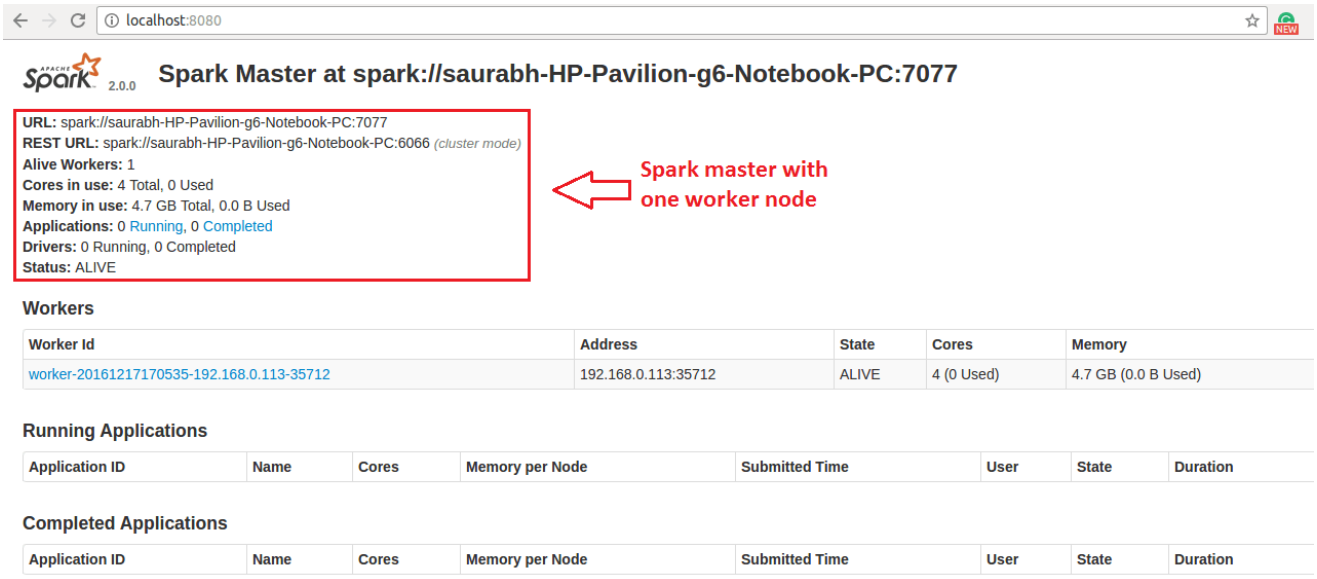

Figure 2: Spark cluster console with alive worker

```
$ ./sbin/start-slave.sh spark://ip address of master:7077 i.e.
spark://saurabh-HP-Pavilion=g6-Notebook-PC:7077
```

- Status of Spark cluster console with alive worker is shown in Figure 2.
- Check the status of Spark daemons using the following command:
  `jps`
- The execution of above command should produce the following output. It ensures the working of master, slave, h2o, and spark context for R (SparkSubmit):

```
835 Worker
28597 h2o.jar
348 Master
1132 SparkSubmit
1213 CoarseGrainedExecutorBackend
1279 Jps
```

Now, Spark cluster integrated with R and H2O package is up and running. We can access it either through SparkR console or RStudio. The RStudio is the better for the ease of programming as it provides the GUI to access integrated R and H2O on top of Spark cluster.