Claudio Fahey's Big Data Blog

Real-Time Global Anomaly Detection in IoT with EMC Elastic Cloud Storage (ECS) - Part 3

Posted by Claudio Fahey in Claudio Fahey's Big Data Blog on Apr 15, 2016 4:41:50 PM

Part 1

See Part 1 of this blog series.

Part 2

See Part 2 of this blog series.

Part 3

Installation Procedure

In this section, I'll provide a high-level overview of the installation procedure that I used to build this system, as well as some important details.

EMC Elastic Cloud Storage (ECS)

The system has two ECS U300 clusters (one for each site) running ECS version 2.2. Complete documentation can be found here.

Hadoop

The system has two independent installations of Hortonworks Data Platform (HDP) 2.4 (one for each site). At a minimum, the following components should be installed.

- YARN + MapReduce2
- Hive
- ZooKeeper
- Flume
- Kafka
- Spark

Anaconda Python

To install Anaconda Python, download Python 2.7 for Linux 64-bit. On each host in your Hadoop clusters, install it with the following commands.

```
# bash ./Anaconda2-*.sh -b -p /opt/anaconda
# /opt/anaconda/bin/pip install pykafka
```

Zeppelin Notebook

Zeppelin needs to be installed at just one site. It can be easily deployed using Ambari using the procedure described here.

To configure Zeppelin to use Anaconda Python:

1. In the Zeppelin UI, click Interpreter.
2. Find the parameter zeppelin.pyspark.python and set it to "/opt/anaconda/bin/python".

To make PySpark the default for paragraphs typed into Zeppelin notebooks:

1. In Ambari, click Zeppelin Notebook, then Configs.
2. Under Advanced zeppelin-config, find zeppelin.interpreters and move "org.apache.zeppelin.spark.PySparkInterpreter" to the beginning of the list. For example: org.apache.zeppelin.spark.PySparkInterpreter,org.apache.zeppelin.spark.SparkInterpreter,...
3. Save and restart Zeppelin.

Kafka

See Part 2 for how to create the topics with the appropriate number of partitions and replicas. This must be done on each.

Flume

See Part 2 for the flume.conf configuration to use. This must be done on each.

KDD Cup 99 Data

Download kddcup.data.gz from KDD Cup 1999 Data. Place the file on the server that you will use to run the data generator script (streaming_data_generator.py) at each site.

Anomaly Detection Demo Application

It will be convenient to used a shared NFS drive that is accessible from hosts at each site. If one is not available, then repeat this procedure at each site.

First, clone the Git repository.

```
$ git clone https://github.com/claudiofahey/global_anomaly_detection_demo.git
```

Edit the file config.sh as appropriate for your environment.

Import the Zeppelin*.json files into Zeppelin.

You may want to edit and use the script start_all_streaming_jobs.sh to automatically start the data generator and Spark Streaming jobs at site 1 and site 2.

Run batch_model_builder.sh on site 1 to run the Spark job to build the model.

Questions? Problems?

If you have any questions or problems, leave a comment below.

 Tags: zeppelin , streaming , spark , machine_learning , kafka , iot , hadoop , flume , ecs , analytics

0 Comments

There are no comments on this post