BIG DATA CYBERSECURITY

LAB 9 INSTALL APACHE METRON IN AWS

INSTALLING APACHE HADOOP AND APACHE SPOT IN AN AMAZON WEB SERVICES VIRTUAL MACHINE

Lab Description: Installing open-sources software is not always a straightforward task due to the common lack of software configuration management. This lab assignment will walk you through the steps needed to create a virtual machine in the Amazon Web Services cloud, set up needed dependencies and parameters, configure Apache Hadoop, Apache Spot and the rest of the Hadoop ecosystem.

It is recommended for the lab instructor to apply for the AWS Educate program providing students and instructors with free credits, which can be applied towards multiple AWS resources [1].

Lab Files that are Needed: localrepo folder, elasticsearch_mpack-0.4.3.0.tar.gz, metron_mpack-0.4.3.0.tar.gz, metron-cluster.ppk, metron-cluster.pem, metron-cluster-pem-public, metron-cluster-public.pub.

LAB EXERCISE/STEP 1

Obtain a public and private key pair from the lab instructor.

LAB EXERCISE/STEP 2

Create and activate an AWS account and login to the AWS Console as described at [2].

LAB EXERCISE/STEP 3

In the top right corner of the AWS Console, click your name, then click My Billing Dashboard. Next, click Credits in the left-hand-side menu. Enter a promo code given by your lab instructor and security check characters displayed as an image. Then click the Redeem button.

LAB EXERCISE/STEP 4

In the top right corner, click the region name to display a menu of available AWS data center locations. It is recommended to select a data center close to your physical location for better data transfer and response time.

In the top left corner, click Services then click EC2. In the menu on the left, locate and click Key Pairs under Network & Security. Click Import Key Pair, give your key pair a name and upload the public key file or paste its contents, and then click the Import button.

LAB EXERCISE/STEP 5

Navigate to Services → EC2 → Instances and click the Launch Instance button. In Step 1: Choose an Amazon Machine Image (AMI) we need to pick an image of a virtual machine with pre-installed operating system (OS). Hadoop can be used with several OSs. In this assignment we will use CentOS 7.

In the search box type centos and press Enter. Click the Select button in the line saying CentOS 7 (x86_64) - with Updates HVM. The next window will display a description of this image, its hourly fees and other information. Click Continue.

LAB EXERCISE/STEP 6

In Step 2. Choose and Instance Type, a virtual machine configuration should be selected. We recommend to use r4.2xlarge instance, since it contains enough resources to run Apache Hadoop with Metron and its hourly cost allows to extend a budget for a number of hours. Click Next: Configure Instance Details.

Make a screenshot and store it in the submission document.

LAB EXERCISE/STEP 7

In Step 3: Configure Instance Details screen shows additional configurable options, which in this case may be left unchanged. Click Next: Add Storage. It is recommended to select at least 32 GB or 64 GB storage size and volume type either General Purpose SSD or Provisioned IOPS SSD. Solid state drives accommodate disk-intensive Hadoop installation and operation.

Make a screenshot and store it in the submission document.

Click Next: Add Tags.

LAB EXERCISE/STEP 8

In Step 5: Add Tags, it is possible to add a tag with Key set to Name and Value set to a label of your choice, e.g. Metron Server. This label will be displayed later in the Running Instances view. Click Next: Configure Security Groups.

LAB EXERCISE/STEP 9

Apache Metron and other application from the Hadoop ecosystem, which are used with Metron or separately for big data processing, require a number of ports to be open for connecting to their interfaces. As a tradeoff between opening a number of custom ports and leaving the system unprotected, it is possible to open ports, which will be accessible only from the IP address of a client you are performing this assignment from. Thus, in Step 6: Configure Security Ports choose Create a new security group, then select All TCP in the dropdown menu of the Type column, and pick My IP in the Source column. This will make all the ports open to only your client computer.

Click Review and Launch. A pop-up window titled Select an existing key pair or create a new key pair will be displayed. Choose your existing key pair, check the I acknowledge … box and click Launch Instances.

A new Launch Status window will show Your instances are now launching message.

LAB EXERCISE/STEP 10

Click the View Instances button in the bottom right corner. A table will show your single running instance. Locate the IPv4 Public IP column and copy the IP address. Open a terminal on your local computer and establish an SSH connection using *centos* as a username and the private key from the key pair created in Step 1. This process was described in a previous lab assignment.

To get additional information on setting up a connection, click the Connect button located above the table.

Make a screenshot and store it in the submission document.

LAB EXERCISE/STEP 11

The rest of the steps should be completed in the SSH terminal window connected to the virtual machine [3]. Execute the commands:

```
su sudo
mkdir /home/metroninst
cd /home/metroninst
pip install --upgrade setuptools
npm install @angular/cli
cp -rp /home/metroninst/metron/metron-deployment/packaging/docker/rpm-docker/RPMS/noarch/* /localrepo/
sudo yum install -y ntp
sudo systemctl enable ntpd
```

Edit file /etc/hosts:

Remove the following lines:

```
ipv4 'localhost.localdomain'
ipv6 'localhost.localdomain'
```

And add this line:

```
127.0.0.1   metronserver metronserver.localdomain localhost
```

Execute this command in the terminal to set the host name:

```
hostnamectl set-hostname metronserver.localdomain
```

Download Ambari repository:

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos7/2.x/updates/2.6.1.5/ambari.repo -O /etc/yum.repos.d/ambari.repo
```

Install Ambari server:

```
yum install ambari-server -y
```

Install MariaDB by executing the command below:

```
mysql_secure_installation
```

Login to MariaDB and execute the lines below to create users, privileges, database and tables needed for the installation. Notice that the script below creates usernames and assigns passwords, which will be used later. The passwords may be customized.

```
mysql -u root -pMetronsailing1
create database ambaridb;
use ambaridb;
CREATE USER 'ambariu'@'localhost' IDENTIFIED BY 'Metronsailing1';
GRANT ALL PRIVILEGES ON ambaridb.* TO 'ambariu'@'localhost';
CREATE USER 'ambariu'@'%' IDENTIFIED BY 'Metronsailing1';
GRANT ALL PRIVILEGES ON ambaridb.* TO 'ambariu'@'%';
FLUSH PRIVILEGES;
exit

cd /var/lib/ambari-server/resources/
mysql -u ambariu -pMetronsailing1
USE ambaridb;
SOURCE Ambari-DDL-MySQL-CREATE.sql;
quit

# MySQL for Metron REST:

mysql -u root -pMetronsailing1
create database metronrest;
use metronrest;
CREATE USER 'metron'@'localhost' IDENTIFIED BY 'Smoothmetron2';
GRANT ALL PRIVILEGES ON metronrest.* TO 'metron'@'localhost';
CREATE USER 'metron'@'%' IDENTIFIED BY 'Smoothmetron2';
GRANT ALL PRIVILEGES ON metronrest.* TO 'metron'@'%';
FLUSH PRIVILEGES;

create table if not exists users(
 username varchar(50) not null primary key,
 password varchar(50) not null,
 enabled boolean not null
);
create table authorities (
 username varchar(50) not null,
 authority varchar(50) not null,
 constraint fk_authorities_users foreign key(username) references
 users(username)
);
create unique index ix_auth_username on authorities
(username,authority);
insert into users (username, password, enabled) values ('metron',
'Smoothmetron2',1);
insert into authorities (username, authority) values ('metron',
'ROLE_USER');
commit;
exit
mysql -u root -pMetronsailing1
create database hive;
use hive;
CREATE USER 'hive'@'localhost' IDENTIFIED BY 'Hivesailing1';
GRANT ALL PRIVILEGES ON hive.* TO 'hive'@'localhost';
CREATE USER 'hive'@'%' IDENTIFIED BY 'Hivesailing1';
GRANT ALL PRIVILEGES ON hive.* TO 'hive'@'%';
FLUSH PRIVILEGES;
exit
```

Make a screenshot and store it in the submission document.

Start the Ambari Server setup:

```
ambari-server setup
```

During the setup, the following parameters are recommended:

Select y when prompted to enter advanced database configuration. Then enter:

3 to choosef MySQL/MariaDB

Host: localhost

Port: 3306

DB: ambaridb

User: ambariu

Password: Metronsailing1

LAB EXERCISE/STEP 12

Install Ambari extension packs for Metron and Elasticsearch. Copy the two files and directory into the /home/centos/ directory:

elasticsearch_mpack-0.4.3.0.tar.gz

elasticsearch_mpack-0.4.3.0.tar.gz

localrepo

Install the extension packs using these commands:

```
ambari-server install-mpack --mpack=metron_mpack-0.4.3.0.tar.gz --verbose
ambari-server install-mpack --mpack=elasticsearch_mpack-0.4.3.0.tar.gz --verbose
```

Make a screenshot and store it in the submission document.

Provide Ambari Server with a path to Java MySQL connector:

```
ambari-server setup --jdbc-db=mysql --jdbc-driver=/usr/share/java/mysql-connector-java.jar
```

Start Ambari Server:

```
ambari-server start
```

LAB EXERCISE/STEP 13

Start a web browser and navigate it to

http://ipaddress:8080

where ipaddress should be substituted with IPv4 of your AWS virtual machine. It will open Apache Ambari web interface. Ambari is used to setup, monitor and maintain an Apache Hadoop cluster.

Use admin as a username and a password for Ambari. Enter metron as a cluster name. Pick 2.5 version of the stack. Select "Perform manual registration on hosts and do not use SSH."

Switch to the SSH terminal to install and start an Ambari agent:

```
yum install ambari-agent
ambari-agent start
```

LAB EXERCISE/STEP 14

Then switch back to Ambari web interface and click the Register and Confirm button.

In the "Choose Services" page, select the following required services for a working Metron deployment. You may select additional services as needed: HDFS, YARN + MapReduce2, HBase, Tez, Hive, Pig, Sqoop, Oozie, Zookeeper, Storm, Spark, Kafka, Zeppelin, Elasticsearch, Kibana, Metron, Slider, and Ambari Metrics.

Settings of several applications must be tuned as follows.

Increase the number of Storm supervisor slots from 2 to 15. Storm requires one port per worker. If Storm does not have enough ports, it will leave topologies inoperable without workers. Provide the port numbers for Storm:

```
supervisor.slots.ports = [6700, 6701, 6702, 6703, 6704, 6705, 6706,
6707, 6708, 6709, 6710, 6711, 6712, 6713, 6714]
```

Suggested HDFS settings:

NameNode Java Heap Size: 0.5 GB

DataNode failed disk tolerance: 0

NameNode Server threads: 50

DataNode maximum Java heap size: 0.5 GB

Suggested YARN settings:

Memory Node: 512 MB

Memory Container: 512 MB

Maximum Container Size (Memory): 512 MB

yarn.app.mapreduce.am.resource.mb = 1024 MB

Suggested Hive settings:

Hive Server2 Heap Size: 512 MB

Metastore Heap Size: 512 MB

Client Heap Size: 512 MB

Memory for Map Join, per Map memory: 136 MB

Data per Reducer: 64 MB

Suggested HBase settings:

HBase Master Maximum Memory: 0.5 GB

HBase RegionServer Maximum Memory: 0.5 GB

Suggested Ambari Infra Solr settings:

Minimum Heap Size: 0.5 GB

Maximum Heap Size: 0.5 GB

Settings for Elasticsearch:

zen_discovery_ping_unicast_hosts (specify the location where Elasticsearch master is installed): metronserver.localdomain

In Advanced elastic-site (for single-node cluster):

masters_also_are_datanodes = "true"

gateway_recover_after_data_nodes = 1

expected_data_nodes = 0

network_host = 0.0.0.0

index_number_of_shards = 4

index_number_of_replicas = 0

Settings for Kibana

The Elasticsearch instance to use for all your queries: http://metronserver.localdomain:9200

Settings for Metron:

Elasticsearch Hosts: metronserver.localdomain

Metron REST port: 8082

JDBC URL: jdbc:mysql://metronserver.localdomain:3306/metronrest

JDBC Driver: com.mysql.jdbc.Driver

JDBC Username: metron

JDBC Password: Smoothmetron2

Metron JDBC platform: mysql

Metron JDBC client path: /usr/share/java/mysql-connector-java.jar

Make a screenshot and store it in the submission document.

Then click Next, then a warning pops up that memory values are less than recommended. Click Proceed Anyway. Complete the setup process by following the installation wizard.

LAB EXERCISE/STEP 15

After the setup process completes, perform these additional steps. In Ambari web interface locate Metron and click Service Actions. Then perform Elasticsearch template install and Kibana dashboard install.

Switch to the SSH terminal window, and edit the elasticsearch.yml:

```
vi /etc/elasticsearch/elasticsearch.yml
```

Perform the following edits:

```
node:
  data: true
  master: true
  name: metron
index:
  number_of_shards: 1
  number_of_replicas: 0
```

Save and exit the file. Restart Elasticsearch in the command line:

```
service elasticsearch restart
curl -XDELETE http://localhost:9200/.kibana
curl -XPUT "http://127.0.0.1:9200/_settings?pretty" \
   -d '{ "index": {"number_of_replicas":0}}'
curl -XPUT "http://127.0.0.1:9200/_settings?pretty" \
   -d '{ "node": {"master":true}}'
# Then for future indices:
curl -XPUT localhost:9200/_template/template_1 -d '
{
    "index" : {
        "number_of_replicas" : 0
    }
}'
```

Make a screenshot and store it in the submission document.

LAB EXERCISE/STEP 16

After the assignment is complete or should be stopped for any other reason, it is very important to terminate the virtual machine you have been using. Otherwise, the budget will keep depleting. If a credit card was connected to your account, the charge will be made against the card.

To stop the virtual machine, navigate to Services → EC2 → 1 Running Instances. Check the box in the instance line of the table, click Actions → Instance State → Terminate.

Next, click Volumes under Elastic Block Store in the menu on the left side. The storage volume, which was attached to the terminated virtual machine, should be in the Available state. Check the box next to the Volume ID and click Actions → Delete Volume → Yes, Delete.

LAB EXERCISE/STEP 17

Be sure to terminate your AWS Metron VM to avoid budget depletion. Resources created in a cloud environment under your account have associated costs. AWS offers more than 60 products at a free tier with associated free tier usage limits [4].

PUZZLER

Sometimes, it is needed to manipulate services from Hadoop ecosystem using Ambari command line interface. For example, you need to delete the SmartSense service. Execute the commands given below to achieve the described effect.

First, make sure it exists by running the command in the SSH terminal:

```
curl -u admin:admin -H "X-Requested-By: ambari" -X GET http://metronserver.localdomain:8080/api/v1/clusters/metron/services/SMARTSENSE
```

Note text in italics. “admin:admin” stand for Ambari username and password pair. “metron” stands for the cluster name. Second, stop the required service:

```
curl -u admin:admin -H "X-Requested-By: ambari" -X PUT -d '{"RequestInfo":{"context":"Stop Service"},"Body":{"ServiceInfo":{"state":"INSTALLED"}}}' http://metronserver.localdomain:8080/api/v1/clusters/metron/services/SMARTSENSE
```

Note that SMARTSENSE is a service name. Third, delete the service:

```
curl -u admin:admin -H "X-Requested-By: ambari" -X DELETE
http://metronserver.localdomain:8080/api/v1/clusters/metron/services/SMARTSENSE
```

Sometimes, it is needed to restart all services using Ambari. This can be done via pressing a button in Ambari user interface. On the other hand, the commands below can be used from a script. Stop all services in Ambari:

```
curl -i -u admin:admin -H "X-Requested-By: ambari"  -X PUT  -d
'{"RequestInfo":{"context":"_PARSE_.STOP.ALL_SERVICES","operation_level":{"level":"CLUSTER","cluster_name":"metron"}},"Body":{"ServiceInfo":{"state":"STARTED"}}}'
http://localhost:8080/api/v1/clusters/metron/services
```

Start all services in Ambari from command line:

```
curl -i -u admin:admin -H "X-Requested-By: ambari"  -X PUT  -d
'{"RequestInfo":{"context":"_PARSE_.START.ALL_SERVICES","operation_level":{"level":"CLUSTER","cluster_name":"metron"}},"Body":{"ServiceInfo":{"state":"STARTED"}}}'
http://localhost:8080/api/v1/clusters/metron/services
```

What to submit

Submit a Word (or other text editor) document with embedded screenshots made as requested in the assignment and a brief description for each screenshot.

References

[1] AWS, "AWS Educate. Teach Tomorrow's Cloud Workforce Today," [Online]. Available: https://aws.amazon.com/education/awseducate/. [Accessed Aug 8, 2019].

[2] AWS, "How do I create and activate a new Amazon Web Services account?" [Online]. Available: https://aws.amazon.com/premiumsupport/knowledge-center/create-and-activate-aws-account/[Accessed Aug 8, 2019].

[3] Apache, "Metron 0.4.1 with HDP 2.5 bare-metal install on Centos 7 with MariaDB for Metron REST," [Online]. Available: https://cwiki.apache.org/confluence/display/METRON/Metron+0.4.1+with+HDP+2.5+bare-metal+install+on+Centos+7+with+MariaDB+for+Metron+REST. [Accessed Aug 8, 2019].

[4] AWS, “AWS Free Tier,” [Online]. Available: https://aws.amazon.com/free/. [Accessed Aug 8, 2019].