



ugr

Universidad
de Granada

Máster en Ciencia de Datos e Ingeniería de Computadores

Big Data I: Cloud Computing y Almacenamiento Masivo de Datos

Parte III: Diseña un experimento ETL con Impala a tu medida

Autora

Lidia Sánchez Mérida

Contacto

lidiasm96@correo.ugr.es



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

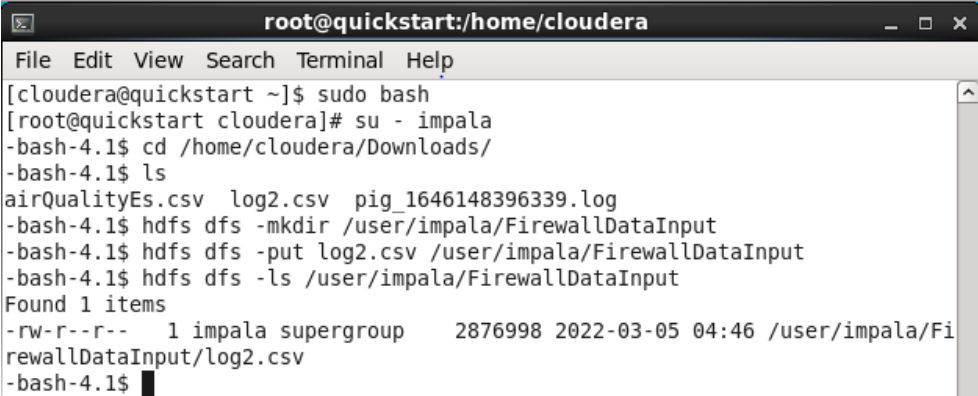
Granada, Marzo de 2022

Descripción de la base de datos

La base de datos seleccionada se denomina **Internet Firewall** y se encuentra disponible en el repositorio *UCI Machine Learning*, junto con un amplio conjunto de datasets que permiten realizar diversos experimentos para resolver problemas de regresión y clasificación. Este conjunto de datos procede de los registros generados por un *firewall* instalado en equipos informáticos de una universidad y data del año 2019. Dispone de un total de **65.532 registros y 12 columnas**, mayormente numéricas a excepción de la variable dependiente que se compone de cuatro clases: *allow*, *deny*, *drop*, *reset-both*. Una de las aplicaciones más interesantes consiste en **modelar el comportamiento del tráfico online** utilizando esta información para mejorar la toma de decisiones por parte de programas de seguridad, como los *firewalls*, a través de un clasificador que sea capaz de identificar qué acción es la más apropiada para cada situación. A partir de esta idea ha surgido un nuevo concepto conocido como NGFW (*Next-generation firewalls*) que hace referencia al nuevo estándar de seguridad que se lleva desarrollando e implementando en los últimos años. El principal objetivo consiste en construir **programas de seguridad proactivos** introduciendo técnicas de Aprendizaje Automático que les permitan aprender nuevas amenazas y comportamientos sospechosos para reducir el tiempo de respuesta en aplicar una solución automática y acorde a diversos ataques de ciberseguridad.

Experimentación con Impala

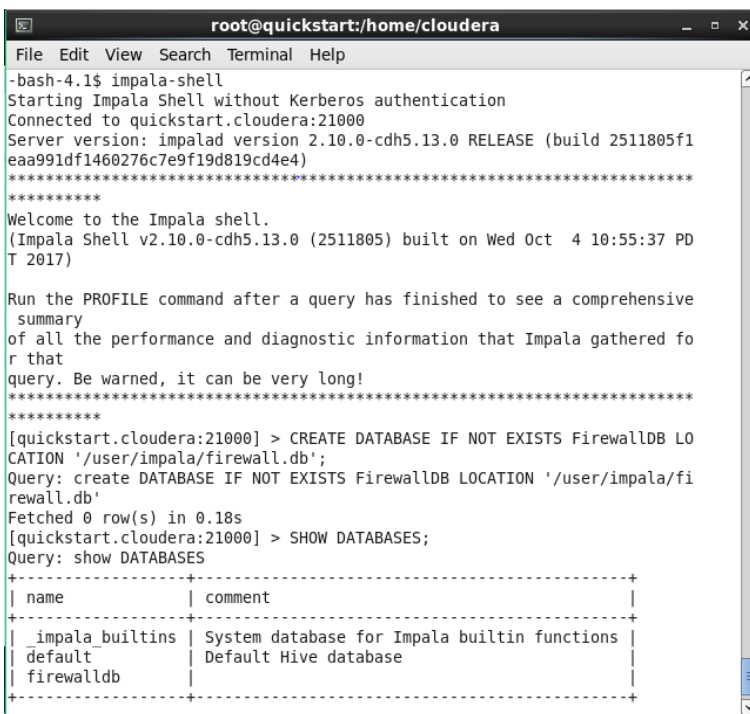
En primer lugar **descargamos el fichero log2.csv** que contiene el conjunto de datos explicado anteriormente para almacenarlo dentro de la carpeta *Downloads*. A continuación **iniciamos sesión con el usuario impala** tal y como se muestra en la Figura 1. Para operar con el dataset seleccionado procedemos a **crear una carpeta en HDFS** denominada *FirewallDataInput* y subimos una copia del archivo de datos mencionado anteriormente.



```
root@quickstart:/home/cloudera
File Edit View Search Terminal Help
[cloudera@quickstart ~]$ sudo bash
[root@quickstart cloudera]# su - impala
-bash-4.1$ cd /home/cloudera/Downloads/
-bash-4.1$ ls
airQualityEs.csv log2.csv pig_1646148396339.log
-bash-4.1$ hdfs dfs -mkdir /user/impala/FirewallDataInput
-bash-4.1$ hdfs dfs -put log2.csv /user/impala/FirewallDataInput
-bash-4.1$ hdfs dfs -ls /user/impala/FirewallDataInput
Found 1 items
-rw-r--r--  1 impala supergroup    2876998 2022-03-05 04:46 /user/impala/FirewallDataInput/log2.csv
-bash-4.1$
```

Figura 1. Creación de una carpeta en HDFS para trasladar el fichero de datos desde el usuario impala.

A continuación iniciamos la *shell* de Impala y **creamos una base de datos** denominada *FirewallDB* ubicándola en el directorio de este usuario dentro de un nuevo fichero generado bajo el nombre *firewall.db*. Como podemos apreciar en la Figura 2, la base de datos se ha creado correctamente y se encuentra disponible para su uso.



```
root@quickstart:/home/cloudera
File Edit View Search Terminal Help
-bash-4.1$ impala-shell
Starting Impala Shell without Kerberos authentication
Connected to quickstart.cloudera:21000
Server version: impalad version 2.10.0-cdh5.13.0 RELEASE (build 2511805f1
eaa991df1460276c7e9f19d819cd4e4)
*****
Welcome to the Impala shell.
(Impala Shell v2.10.0-cdh5.13.0 (2511805) built on Wed Oct 4 10:55:37 PD
T 2017)

Run the PROFILE command after a query has finished to see a comprehensive
summary
of all the performance and diagnostic information that Impala gathered fo
r that
query. Be warned, it can be very long!
*****
[quickstart.cloudera:21000] > CREATE DATABASE IF NOT EXISTS FirewallDB LO
CATION '/user/impala/firewall.db';
Query: create DATABASE IF NOT EXISTS FirewallDB LOCATION '/user/impala/fi
rewall.db'
Fetched 0 row(s) in 0.18s
[quickstart.cloudera:21000] > SHOW DATABASES;
Query: show DATABASES
+-----+-----+
| name          | comment                                     |
+-----+-----+
| impala_builtins | System database for Impala builtin functions |
| default        | Default Hive database                     |
| firewallldb    |                                             |
+-----+-----+
```

Figura 2. Creación de una base de datos desde la consola de Impala.

Previo a la carga de datos debemos **generar una tabla** con un esquema compatible con la estructura explicada anteriormente para el conjunto de datos elegido. Para ello, en primer lugar **nos situamos en la nueva base de datos creada** en el paso anterior y, mediante la sentencia *CREATE TABLE*, definimos las columnas que componen el dataset junto con el tipo de dato relativo a los valores que se pretenden almacenar. Adicionalmente especificamos los caracteres que actúan como **separadores** tanto de los propios datos a insertar como de los diferentes registros que componen el dataset, en particular son la coma y el salto de línea. En el primer experimento realizado pude apreciar que la cabecera del fichero que contiene los nombres de las columnas también formaba parte de la ingesta, pero al no disponer de los tipos de datos declarados para cada columna, **generaba una primera fila de valores nulos**. Para corregir este comportamiento se añade a la sentencia el comando *TBLPROPERTIES* con el que

conseguimos **ignorar el primer registro del archivo** con el que se va a realizar la ingesta de la información. Finalmente con *DESCRIBE* podemos visualizar el esquema definido tras crear la tabla, como se muestra en la Figura 3.

```

root@quickstart:/home/cloudera
[quickstart.cloudera:21000] > USE FirewallDB;
Query: use FirewallDB
[quickstart.cloudera:21000] > CREATE TABLE IF NOT EXISTS InternetFirewall (SourcePort INT, DestinationPort INT, NatSourcePort INT, NatDestinationPort INT, Action STRING, Bytes INT, BytesSent INT, BytesReceived INT, Packets INT, ElapsedTime INT, PktsSent INT, PktsReceived INT) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n' TBLPROPERTIES("skip.header.line.count"="1");
Query: create TABLE IF NOT EXISTS InternetFirewall (SourcePort INT, DestinationPort INT, NatSourcePort INT, NatDestinationPort INT, Action STRING, Bytes INT, BytesSent INT, BytesReceived INT, Packets INT, ElapsedTime INT, PktsSent INT, PktsReceived INT) ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n' TBLPROPERTIES("skip.header.line.count"="1")
Fetched 0 row(s) in 0.21s
[quickstart.cloudera:21000] > DESCRIBE InternetFirewall;
Query: describe InternetFirewall
+-----+-----+-----+
| name          | type   | comment |
+-----+-----+-----+
| sourceport    | int    |          |
| destinationport | int    |          |
| natsourceport  | int    |          |
| natdestinationport | int    |          |
| action        | string |          |
| bytes         | int    |          |
| bytessent     | int    |          |
| bytesreceived  | int    |          |
| packets       | int    |          |
| elapsedtime   | int    |          |
| pktsent       | int    |          |
| pktsreceived  | int    |          |
+-----+-----+-----+

```

Figura 3. Creación de una tabla específica para el conjunto de datos *Internet Firewall*.

Una vez disponemos de la estructura y el almacenamiento apropiado para el conjunto de datos, procedemos a su **ingesta a través del fichero** situado en el sistema de archivos HDFS. Mediante la consulta *SELECT COUNT(*)* que se puede visualizar en la Figura 4 podemos apreciar que la tabla creada en el paso anterior ahora contiene los 65.532 registros del archivo *log2.csv*.

```

[quickstart.cloudera:21000] > LOAD DATA INPATH '/user/impala/FirewallDataInput/log2.csv' OVERWRITE INTO TABLE InternetFirewall;
Query: load DATA INPATH '/user/impala/FirewallDataInput/log2.csv' OVERWRITE INTO TABLE InternetFirewall
+-----+-----+-----+
| summary |
+-----+-----+-----+
| Loaded 1 file(s). Total files in destination location: 1 |
+-----+-----+-----+
Fetched 1 row(s) in 0.38s
[quickstart.cloudera:21000] > SELECT COUNT(*) FROM InternetFirewall;
Query: select COUNT(*) FROM InternetFirewall
Query submitted at: 2022-03-05 05:20:17 (Coordinator: http://quickstart.cloudera:25000)
Query progress can be monitored at: http://quickstart.cloudera:25000/query_plan?query_id=694ccfa07aff8f76:2af8637000000000
+-----+-----+
| count(*) |
+-----+-----+
| 65532    |
+-----+-----+
Fetched 1 row(s) in 0.21s
[quickstart.cloudera:21000] >

```

Figura 4. Carga del dataset desde el fichero que contiene el conjunto de datos *Internet Firewall*.

Para asegurarnos de que la cabecera del fichero no se ha considerado durante el proceso de la ingesta, en la Figura 5 se muestran los **tres primeros registros en el orden en el que han sido insertados**. Tal y como podemos comprobar, no se han generado valores nulos y los datos que se visualizan se corresponden con los tipos de valores que se declararon en el esquema de la tabla.

```
[quickstart.cloudera:21000] > SELECT * FROM InternetFirewall LIMIT 3;
Query: select * FROM InternetFirewall LIMIT 3
Query submitted at: 2022-03-05 05:24:26 (Coordinator: http://quickstart.cloudera:25000)
Query progress can be monitored at: http://quickstart.cloudera:25000/query_plan?query_id=1a4a713a53b2381f:ff92b59b0000000000
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+
| sourceport | destinationport | natsourceport | natdestinationport | action |
| bytes | bytessent | bytesreceived | packets | elapsedtime | pktssent | pkts |
| received |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 57222 | 94 | 53 | 83 | 54587 | 2 | 53 | 30 | 1 | allow |
| 177 | | | | | | | | | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 56258 | 1600 | 3389 | 3168 | 56258 | 19 | 3389 | 17 | 10 | allow |
| 4768 | | | | | | | | | 9 |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 6881 | 118 | 50321 | 120 | 43265 | 2 | 50321 | 1199 | 1 | allow |
| 238 | | | | | | | | | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+
+-----+
Fetched 3 row(s) in 0.07s
```

Figura 5. Comprobación de la ingesta correcta del dataset ignorando la cabecera del archivo que contiene las columnas.

Una vez disponemos del conjunto de datos almacenado en una tabla particular a su estructura, podemos realizar cualquier tipo de consulta. Para ejemplificar su funcionamiento, en la Figura 6 se realiza una para recuperar aquellos **registros asociados a la clase *deny* visualizando únicamente los puertos origen y destino, además del número de bytes y paquetes enviados** durante cada transacción. De este modo solo recuperamos un conjunto determinado de instancias que cumplen la condición especificada, filtrando además las columnas que deseamos visualizar en el resultado. Al disponer de tanta información, podemos agilizar la consulta y facilitar su captura de imagen añadiendo la sentencia *LIMIT* para **obtener únicamente los veinte primeros registros** resultantes.

```
root@quickstart:/home/cloudera
File Edit View Search Terminal Help
[quickstart.cloudera:21000] > SELECT SourcePort, DestinationPort, BytesSent, Packets
s FROM InternetFirewall WHERE Action='deny' LIMIT 20;
Query: select SourcePort, DestinationPort, BytesSent, Packets FROM InternetFirewall
WHERE Action='deny' LIMIT 20
Query submitted at: 2022-03-05 11:33:17 (Coordinator: http://quickstart.cloudera:25
000)
Query progress can be monitored at: http://quickstart.cloudera:25000/query_plan?que
ry_id=8e45c8bae54ec9b4:7b3aafb5000000000
+-----+
| sourceport | destinationport | bytessent | packets |
+-----+
| 13394      | 23              | 60        | 1        |
| 61078      | 57470           | 62        | 1        |
| 62776      | 62413           | 146       | 1        |
| 46448      | 30170           | 159       | 1        |
| 10688      | 25174           | 146       | 1        |
| 34086      | 25174           | 62        | 1        |
| 64605      | 25174           | 62        | 1        |
| 56688      | 25174           | 62        | 1        |
| 57015      | 42707           | 62        | 1        |
| 41213      | 25174           | 62        | 1        |
| 57131      | 8055            | 145       | 1        |
| 45855      | 25174           | 62        | 1        |
| 54483      | 25174           | 66        | 1        |
| 45855      | 25174           | 146       | 1        |
| 49990      | 62413           | 66        | 1        |
| 1582       | 56205           | 62        | 1        |
| 65003      | 57470           | 78        | 1        |
| 63486      | 51505           | 66        | 1        |
| 54100      | 37965           | 62        | 1        |
| 1815       | 22114           | 66        | 1        |
+-----+
```

Figura 6. Consulta para recuperar los registros clasificados como *deny* seleccionando un subconjunto de columnas.