UNIVERSIDAD DE GRANADA

Big Data I

máster ciencia de datos e ingeniería de computadores

Procesamiento y Almacenamiento en Impala

Práctica sobre ETL

Autor
Ignacio Vellido Expósito
ignaciove@correo.ugr.es

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

Curso 2020-2021

1. Experimento

Dataset con medidas de peticiones en la red de una universidad, publicado en la página de la UCI (`https://archive.ics.uci.edu/ml/datasets/Internet+Firewall+Data`). Cuenta con 65.532 instancias y los siguientes 12 atributos:

- Source Port
- Destination Port
- NAT Source Port
- NAT Destination Port
- Action (cuatro tipos: allow, action, drop y reset-both)
- Bytes
- Bytes Sent
- Bytes Received
- Packets
- Elapsed Time (sec)
- pkts_sent
- pkts_received

1.1. Pasos para el desarrollo del experimento

Tras descargar el .csv, debemos cargarlo en el HDFS. Primero lo movemos a una carpeta compartida para todos los usuarios:

```
$ mv Downloads/log2.csv /var/tmp/materialImpala/log2.csv
```

Nos metemos ahora en el usuario de impala

```
$ sudo bash
# su - impala
```

Creamos el directorio en HDFS

```
$ hdfs dfs -mkdir /user/impala/input
```

Cargamos los datos en el directorio

```
$ hdfs dfs -put /var/tmp/materialImpala/log2.csv /user/impala/input
```

Entramos en Impala

```
$ impala-shell
```

Creamos la tabla donde ingestar los datos (Si algún campo de SMALLINT sobresale el límite de memoria Impala añade más espacio automáticamente)

```
> CREATE TABLE IF NOT EXISTS Firewall
    (
        SourcePort          SMALLINT,
        DestinationPort     SMALLINT,
        NATSourcePort       SMALLINT,
        NATDestinationPort  SMALLINT,
        Action              STRING,
        Bytes               INT,
        BytesSent           INT,
        BytesReceived       INT,
        Packets             INT,
        ElapsedTime         SMALLINT,
        pktsSent            INT,
        pktsReceived        INT
    )
    ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
    STORED AS TEXTFILE LOCATION '/user/impala/impalastore.db';
```

```
Query: describe Firewall
+--------------------+----------+---------+
| name               | type     | comment |
+--------------------+----------+---------+
| sourceport         | smallint |         |
| destinationport    | smallint |         |
| natsourceport      | smallint |         |
| natdestinationport | smallint |         |
| action             | string   |         |
| bytes              | int      |         |
| bytessent          | int      |         |
| bytesreceived      | int      |         |
| packets            | int      |         |
| elapsedtime        | smallint |         |
| pktssent           | int      |         |
| pktsreceived       | int      |         |
+--------------------+----------+---------+
```

Cargamos los datos en la tabla

```
> LOAD DATA INPATH '/user/impala/input/log2.csv'
  OVERWRITE INTO TABLE Firewall;
```

```
Query: select count(*) from firewall
Query submitted at: 2021-02-06 07:33:55
Query progress can be monitored at: htt
+----------+
| count(*) |
+----------+
| 65533    |
+----------+
```

Aplicamos la consulta: *Obtener los 3 puertos de destino con más peticiones permitidas, mostrando este número de peticiones y el tiempo medio implicado*

```
> SELECT DestinationPort, COUNT(*), AVG(ElapsedTime)
    FROM Firewall
    WHERE Action = "allow"
    GROUP BY DestinationPort
    ORDER BY 2 DESC
    LIMIT 3;
```

```
Query: select DestinationPort, COUNT(*), AVG(ElapsedTime) FROM Firewall
    WHERE Action = "allow"
    GROUP BY DestinationPort
    ORDER BY 2 DESC
    LIMIT 3
Query submitted at: 2021-02-06 07:56:51 (Coordinator: http://quickstart.c
Query progress can be monitored at: http://quickstart.cloudera:25000/quer
+-----------------+----------+-------------------+
| destinationport | count(*) | avg(elapsedtime)  |
+-----------------+----------+-------------------+
| 53              | 15385    | 31.3604809879753  |
| 443             | 11677    | 130.6371499528989 |
| 80              | 4028     | 73.56305858987091 |
+-----------------+----------+-------------------+
```

- En el SELECT indicamos los campos a proyectar.
- En el WHERE seleccionamos aquellas instancias de peticiones que el firewall ha permitido.
- Para poder aplicar el COUNT y el AVG, agrupamos por puertos de destino.
- Ordenamos en orden decreciente por la cuenta de peticiones y con el LIMIT nos quedamos con los 3 puertos pedidos.