ES EQL Integration correctness tests

Description

Python EQL runs a series of queries against a specific dataset and the output of those queries (including results and timing) becomes the `queries.toml` file of this module.

The dataset is stored as a snapshot on a bucket in gcs. This module starts up an ES node, restores these data, executes the queries and asserts the results that are provided along with the query statement in the `queries.toml` file.

Running the tests

To be able to run the tests locally, one should set the environmental variable `eql_test_credentials_file` pointing to a local file holding the service account credentials which allow access to the gcs bucket where the dataset resides. E.g.:

```
export eql_test_credentials_file=/Users/username/credentials.gcs.json
```

To run the tests you can issue:

```
./gradlew -p x-pack/plugin/eql/qa/correctness check
```

or simply run:

```
./gradlew -p x-pack/plugin/eql check
```

If the `eql_test_credentials_file` environmental variable is not set the correctness tests will not be executed.

For every query you will get an `INFO` line logged that shows the response time for the query, e.g.:

```
org.elasticsearch.xpack.eql.EsEQLCorrectnessIT > test {2} STANDARD_OUT
    [2020-10-15T11:55:02,870][INFO ][o.e.x.e.EsEQLCorrectnessIT] [test] [2] before
test
    [2020-10-15T11:55:03,070][INFO ][o.e.x.e.EsEQLCorrectnessIT] [test] QueryNo: 2,
took: 169ms
    [2020-10-15T11:55:03,083][INFO ][o.e.x.e.EsEQLCorrectnessIT] [test] [2] after test
```

At the end of a successful run an `INFO` line is logged by the tests that shows the total response time for all the queries executed, e.g.:

```
[2020-10-15T06:39:55,826][INFO ][o.e.x.e.EsEQLCorrectnessIT] [suite] Total time: 24563
ms
```

Run a specific query

If one wants to run just one query from the set, needs to do it with following command by replacing `<queryNo>` (which can be found in queries.toml file) with the desired number of the query:

```
./gradlew ':x-pack:plugin:eql:qa:correctness:javaRestTest' --tests
"org.elasticsearch.xpack.eql.EsEQLCorrectnessIT.test {<queryNo>}"
```

Debug queries

If one wants to check that the filtering subqueries of a sequence query yields the same results (to pinpoint that the possible failure is in the sequence algorithm), needs to enable this debug mode with the use of a parameter:

```
./gradlew -p x-pack/plugin/eql/qa/correctness check -
Dtests.eql_correctness_debug=true
```

or

```
./gradlew ':x-pack:plugin:eql:qa:correctness:javaRestTest' --tests
"org.elasticsearch.xpack.eql.EsEQLCorrectnessIT.test {<queryNo>}" -
Dtests.eql_correctness_debug=true
```

Preserve data across node restarts

If you'd like to preserve the restored index and avoid the network download and delay of restoring them on every run of the node, you can set the `eql.test.preserve.data` system property, e.g.:

```
./gradlew :x-pack:plugin:eql:qa:correctness:javaRestTest -
Deql.test.preserve.data=true
```

Run an ES node manually and run the tests against it

If one wants to run an ES node manually (most probably to be able to debug the server), needs to run the following:

```
./gradlew :x-pack:plugin:eql:qa:correctness:runEqlCorrectnessNode --debug-jvm
```

Set the `eql_test_credentials_file` environmental variable correctly in the shell before running the command above,

Once the ES node is up and running, the data can be restored from the snapshot by running the `main` of the `EqlDataLoader` class.

Once the data is loaded, a specific query can be run against the running ES node with:

```
./gradlew ':x-pack:plugin:eql:qa:correctness:javaRestTest' --tests
"org.elasticsearch.xpack.eql.EsEQLCorrectnessIT.test {<queryNo>}" -
Dtests.rest.cluster=localhost:9200 -Dtests.cluster=localhost:9200 -
Dtests.clustername=runTask-0
```

Set the `eql_test_credentials_file` environmental variable correctly in the shell before running the command above,

Preserve data across node restarts

If you'd like to preserve the restored index and avoid the network download and delay of restoring them on every run of the node, you can start the node with `--preserve-data`:

```
./gradlew :x-pack:plugin:eql:qa:correctness:runEqlCorrectnessNode --debug-jvm --
preserve-data
```