HW3: Elasticsearch Project

Big Picture

Stakeholder[1]

- Internal
  - Project Manager
    - Shay Banon
  - Project Team
    - GitHub Developer Contributors
    - Elastic employees
      - Developers
      - Product managers
      - OA Engineers
      - Designers

      …...
  - Company: Elastic
    - CEO & Key Developer: Shay Banon
    - Other board members
    - Leadership
    - Engineering

    ……
  - Funders
    - Shareholders of U.S. stock market
- External
  - Partner Companies
    - Alibaba Cloud
    - Tencent Cloud Computing (Beijing) Co., Ltd
    - Cisco
    - Google Cloud Platform
    - Microsoft

    …...
  - Customer & Users
  - Suppliers
  - Government

Functionality[2]

- Overall Domain:
    - Cloud Computing
    - Search Engine
    - Big Data
- Essential functional aspects:
    - Store index data distributedly
    - Return searching result for queries quickly
    - Scalable for various data scale and machine numbers

- Essential non-functional aspects:
    - Understandable and Usable Interface
    - Fault-tolerant
    - Security: IP filter, User Authorization
    - Expandable
    - Operational persistence
- Unique aspects:
    - Distributing computing
    - Rest API
    - Realtime data and analysis
    - Document-based full-text search

Key Developers

- Core maintainers, Developers
    - #1 Kimchy (Shay Banon): kimchy@gmail.com
    - #2 s1monw (Simon Willnauer): simonw@apache.org
    - #3 martijnvg (Martijn van Groningen): simonw@apache.org
    - #4 jasontedor (Jason Tedor): jason@tedor.me
    - #5 javanna (Luca Cavanna): N/A
    - #6 rjernst (Ryan Ernst): ryan@iernst.net
    - #7 nik9000 (Nik Everett): nik9000@gmail.com
    - #8 dadoonet (David Pilato): david@pilato.fr
    - #9 jpountz (Adrien Grand): N/A
    - #10 bleskes (Boaz Leskes): N/A

      …...
- Testers
- Documentation writers

  …...

Issues

1.Simplify object creating in ConstructingObjectParser #52499

This issue describes that currently, the constructor of ConstructingObjectParser takes in a function as a parameter whose functionality is to build an object from an array of objects. This causes us to write lots of unnecessary codes such as

```
                int i = 0;
                boolean completed = (boolean) a[i++];
                TaskInfo task = (TaskInfo) a[i++];
                BytesReference error = (BytesReference) a[i++];
                BytesReference response = (BytesReference) a[i++];
                return new TaskResult(completed, task, error, response);
```

This issue is located org.elasticsearch.common.xcontent.ConstructingObjectParse. The issue reporter also came up with two plausible solutions:

1. Wrap the object with an iterator
2. Use maps instead of arrays.

2. SQL: Validate params for next page requests #52207

We can specify fetch_size in Elasticsearch SQL request, and it will return the cursor of the next patch of response. This issue suggests that we should add validation for sql next page request. Currently, next page requests can have the same params as the initial requests, although most of the params will be ignored, it is safer to add a validation to prevent the misuse of those params. For example if the time_zone parameter is different from the initial request it may cause a problem.

We think this can be achieved by edit the validate() method in SqlQueryRequest of package org.elasticsearch.xpack.sql.action, and add validation for the parameters

3. Prevent misuse of IP filtering #51178

This issue talks about that if the user uses the IP filter and set it to deny all of the IPs, he/she will fail to control the system since he can not do any operations on it without the access permission to the host.

In this case, we think we could add a condition in IPfilter.java to check whether the user setting of enabling the IP filter. If the ip filter is set to be unenabled, it will allow users to access the host, otherwise, it denies the request based on the ip filter rules. Our IP filter could read the user settings from the settings file.

4. EQL: Verification tests need improving #51873

The issue reporter suggests that the current test cases for the Verifier are too simple. He says we should use more complex datasets from both mapping and query point of view.

Since we have learned several ways to write efficient test cases to cover the methods in the class, we think we may be able to handle this issue. After doing some research on the Verifier related classes (especially the mapping and query part) and understanding how it works, we may apply Pairwise (2-way) Testing or Exhaustive combinations Testing to enrich the testing datasets.

5. Introduce first query #52482

The issue reporter claims that because of the removal of the funciont_score query, a useful and unique functionality disappears and she wants it back as a dedicated query type.

The function she is asking for is a type of query that evaluates all of the child queries in order and returns the score of the first matching one. The basic function of evaluating the ordered child queries and calculating the score of the query already exists. Therefore the issue fixer only needs to combine the functions and provide an interface for this kind of query, which is the reason why we think we may be able to address it.

Reference:

[1] https://www.geeksforgeeks.org/software-engineering-stakeholder/
[2] https://github.com/elastic/elasticsearch