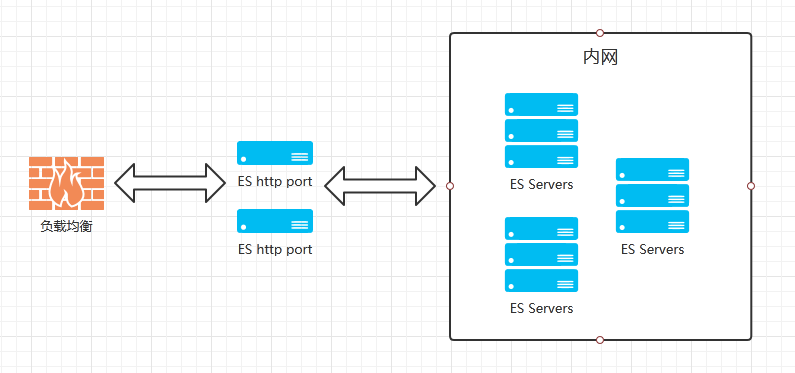
大数据平台ES权限功能使用手册

# 文档说明

大数据平台集群组件通常为各个项目或厂商共用，在未加权限控制和规范的情况下，往往会因为某个项目对集群资源占用过高，导致集群无法稳定运行。大数据平台ES权限功能是为了维护ES共用集群健康、稳定运行而设计的一项操作控制型功能。本文将介绍如何使用大数据平台ES权限功能。

# 架构简介



大数据集群包含三个部分，负载均衡服务器、ES http port、ES数据节点。负载均衡服务器负责接收并分发客户端请求，ES http port 负责处理http请求并控制权限、ES数据节点负责存储数据，处理查询。

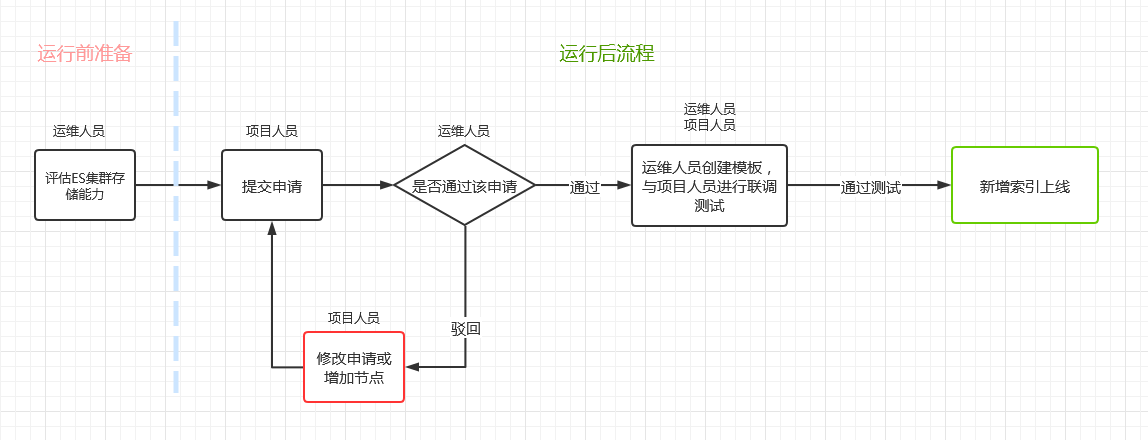
# 流程简介

## 角色说明

**运维人员**：负责维护集群正常平稳运行。主要工作包含检查集群健康状态、健康日报月报、审核索引增加申请、 配合项目人员测试新增索引等。

**项目人员**：对索引进行实际的增删改查操作。主要工作包含设计提供索引template脚本、提交索引新增申请、对已申请的索引进行日常维护等。

## 使用流程



一个完整的新增索引流程包含如下几个步骤，相应的人员可根据每一步的详细说明来确认该步骤的操作：

1. [运维人员评估ES集群存储能力](#_运维人员评估ES集群存储能力)。
2. [项目人员申请创建索引模板](#_项目人员申请创建索引模板)。
3. [运维人员审批新增索引模板申请](#_运维人员审批新增索引模板申请)。
4. [运维人员通过申请后的创建模板和赋权操作](#_运维人员通过申请后的创建模板和赋权操作)。
5. [运维人员和项目人员测试索引模板是否创建并赋权成功](#_运维人员和项目人员测试索引模板是否创建并赋权成功)。

# 任务说明

## 运维人员评估ES集群存储能力

当ES集群安装配置完毕后，运维人员可使用《ES集群存储量评估工具》来评估ES集群的存储性能。



*《ES集群存储量评估工具》样例*

通过填写左侧的各类节点的属性和数量，右侧黄色部分将会自动算出集群可容纳的ES分片数量和ES可用存储。左侧表格只需要填写用于存储数据的节点信息，集群用于接受客户端请求的节点不必填写。

## 项目人员申请创建索引模板

项目人员在提交申请创建新的索引模板前，需要收集以下的信息提交给运维人员。

**索引种类名称**： 索引种类的名称，需要创建的索引模板的名称，例：t\_maker\_black\_full\_data\_yyyyMM。

**存储周期**：索引生成周期，例如，年、月、日。

**单索引数据量（单位GB）**：每个索引的实际数据量大小，单位GB。

**周期内索引数**：集群内需要同时存储多少个该类索引。

**单索引分片数（包含副本）**：每个索引需要划分为多少分片，包含副本分片。

**归属项目**：归属于哪个项目。

**索引模板：**包含索引信息的索引template

项目人员根据模板提交《大数据平台ES集群索引增加申请-yyyyMMdd-项目名称》 和《索引模板脚本》给现场运维，运维人员根据当前集群情况判断是否通过或驳回修改后再提交该申请。若申请通过，则等运维人员反馈进行下一步测试；若申请被驳回，项目人员可以通过调整所属项目的索引留存数量，或申请增加新数据节点来确保申请通过。



*《大数据平台ES集群索引增加申请-yyyyMMdd-项目名称》样例*



*《索引模板脚本》样例*

## 运维人员审批新增索引模板申请

当接收到项目人员发来的新增索引模板申请后，运维人员可根据项目人员提供的信息，使用《ES集群存储量评估工具》的右侧表格来评估是否可以通过该申请。



*《ES集群存储量评估工具》样例*

当运维人员准确填写需要申请的索引信息之后，评估计算集群性能的绿色区域会自动算出当前的分片使用率和存储使用率。当新增的索引会导致分片使用率或存储使用率超过100%时，运维人员需要驳回该申请，可根据《ES集群使用规范》给出相应的建议。若集群可以容纳新增的索引，则运维人员根据项目人员提供的《索引模板脚本》来进行索引模板新增操作，并在ES 鉴权配置中进行配置。

## 运维人员通过申请后的创建模板和赋权操作

运维人员首先应使用项目人员提供的《索引模板脚本》以及管理员账号来创建索引模板。索引模板创建成功后，对ES鉴权的配置文件进行修改添加。只需修改用于接收客户端请求的节点中的配置文件。配置文件的路径为ES安装路径下的 {ES\_HOME}/config/readonlyrest.yml 文件。



*《readonlyrest.yml》文件样例*

readonlyrest.yml文件是由不同的配置块和其子模块组成的：

access\_control\_rules: 表示访问控制规则，每个相应的规则都有唯一的name，name可以根据需要自由定义。样例中的第一个子模块是允许该账号进行所有的操作，第二三四个子模块分别代表了允许不同的用户组访问相应的索引类别。例如子模块’ Accept requests from users in group team1 on index1’ 便代表了允许组team1中的账号访问所有index开头的索引。

users: 表示用户组配置。配置可接受加密的key值，但如果无特殊要求，可以明文配置更节省cpu开销。group代表了相应配置的组。auth\_key代表‘账号：密码’。一个用户可以对应多个组，一个组也可以包含多个用户。

比如，上面FZ808项目申请的t\_detector\_grey\_listyyyyMM和t\_marker\_grey\_full\_datayyyyMMdd索引模板。就可以进行如下配置来赋权：

1. 增加一个FZ808的用户组，并允许改组成员访问起所申请的索引类别。将其添加到访问控制规则’ access\_control\_rules’模块下

- name: Accept requests from users in group FZ808

groups: ["FZ808"]

indices: ["t\_detector\_grey\_list\*", "t\_marker\_grey\_full\_data\*"]

1. 增加一个用户，属于FZ808组，将账号和密码返还给项目人员。

- username: fzUser

auth\_key: fzUser:fzPwd

groups: ["FZ808"]

配置结束后，将更新后的readonlyrest.yml拷贝到所有的请求接收服务器上，轮流重启服务器。

## 运维人员和项目人员测试索引模板是否创建并赋权成功

以FZ808所创建的索引模板为例，运维人员使用项目人员提供的《索引模板脚本》创建索引模板后，可使用以下命令来查看模板是否创建成功：

curl -u fzUser: fzPwd -XGET {esHost:port}/\_template/ t\_detector\_grey\_list?pretty

注：‘{}’中是需要根据集群实际情况来修改的地方。

如配置成功，该命令会返回相应的所欲模板。

可以通过以下命令来创建一个测试索引：

curl -u fzUser: fzPwd -XPUT {esHost:port}/ t\_detector\_grey\_list\_test?pretty

如配置成功，会返回{“Acknowledge” : “true”}。

可以通过以下命令来查看是否能够查看相关的索引信息：

curl -u fzUser: fzPwd -XGET {esHost:port}/ t\_detector\_grey\_list\_test?pretty

curl -u fzUser: fzPwd -XGET {esHost:port}/\_cat/ t\_detector\_grey\_list\*?pretty

# 客户端开发

大数据平台目前采用ES 5.6.0版本。为保障数据安全和合理使用，大数据ES集群**仅开放rest端口**。本文档主要介绍如何使用ES REST Client 接入大数据ES集群。更加详细的文档关于如何使用Java REST Client可参考以下链接：

<https://www.elastic.co/guide/en/elasticsearch/client/java-rest/5.6/index.html>

## 项目构建

### Java Low Level REST Client

#### Maven依赖

<dependency>

<groupId>org.elasticsearch.client</groupId>

<artifactId>elasticsearch-rest-client</artifactId>

<version>5.6.16</version>

</dependency>

#### Gradle依赖

dependencies {

compile 'org.elasticsearch.client:elasticsearch-rest-client:5.6.16'

}

### Java High Level REST Client

#### Maven依赖

<dependency>

<groupId>org.elasticsearch.client</groupId>

<artifactId>elasticsearch-rest-high-level-client</artifactId>

<version>5.6.16</version>

</dependency>

#### Gradle依赖

dependencies {

compile 'org.elasticsearch.client:elasticsearch-rest-high-level-client:5.6.16'

}

## 代码样例

### Java Low Level REST Client

#### 类引用：

import org.apache.http.Header;

import org.apache.http.HttpHost;

import org.apache.http.auth.AuthScope;

import org.apache.http.auth.UsernamePasswordCredentials;

import org.apache.http.client.CredentialsProvider;

import org.apache.http.impl.client.BasicCredentialsProvider;

import org.apache.http.impl.nio.client.HttpAsyncClientBuilder;

import org.apache.http.util.EntityUtils;

import org.elasticsearch.client.Response;

import org.elasticsearch.client.RestClient;

import org.elasticsearch.client.RestClientBuilder;

#### 配置认证用户名密码

CredentialsProvider credentialsProvider = new BasicCredentialsProvider();

credentialsProvider.setCredentials(AuthScope.ANY, new UsernamePasswordCredentials("username","password"));//在此配置用户名密码

#### 创建Client实例

RestClientBuilder builder = RestClient.builder(new HttpHost("localhost", 19211)).setHttpClientConfigCallback(

new RestClientBuilder.HttpClientConfigCallback() {

@Override

public HttpAsyncClientBuilder customizeHttpClient(HttpAsyncClientBuilder httpClientBuilder) {

return httpClientBuilder.setDefaultCredentialsProvider(credentialsProvider);

}

});

RestClient restClient = builder.build();

### Java High Level REST Client

Java High Level REST Client 的构建基于Java Low Level REST Client,只是在此基础之上做了一些常用查询的封装。

#### 类引用：

import org.elasticsearch.client.RestHighLevelClient;

其他与Java Low Level REST Client相同

#### 配置认证用户名密码

与Java Low Level REST Client相同

#### 创建Client实例

RestHighLevelClient restHighLevelClient =

new RestHighLevelClient(restLowLevelClient);

其他与Java Low Level REST Client相同