



链滴

ElasticSearch7.x 升级后 SpringBoot 连不上?

作者: [jianzh5](#)

原文链接: <https://ld246.com/article/1579312855978>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

前言

原来项目中使用的ElasticSearch版本6.8升级到了7.5.1，且开启了x-pack安全认证，导致原来的查询口全部使用不了。

注：原SpringBoot版本为2.1.x。

连接ElasticSearch使用的是[spring-boot-starter-data-elasticsearch](#)

本文主要讲述使用ElasticSearch6.x 升级到 ElasticSearch7.x后所需要的修改以及案例代码。

配置修改

SpringBoot版本升级

在SpringBoot2.2.x版本才支持ElasticSearch7.x，所以我们先需要升级SpringBoot版本。

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>2.2.2.RELEASE</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>
```

ES连接配置修改

原来[spring-boot-starter-data-elasticsearch](#)连接ES的配置如下：

```
spring.data.elasticsearch.cluster-nodes = 172.31.0.207:9300
spring.data.elasticsearch.cluster-name = elk-cluster
spring.data.elasticsearch.repositories.enabled = true
```

升级后主要使用rest接口进行连接，由于开启了x-pack认证，所以需要修改es的账号密码，修改后配如下：

```
spring.elasticsearch.rest.uris = 172.31.0.207:9300
spring.elasticsearch.rest.username = elastic
spring.elasticsearch.rest.password = changeme
```

ElasticsearchTemplate升级

原来[spring-boot-starter-data-elasticsearch](#)连接ES主要使用[ElasticsearchTemplate](#)进行操作，新本主要使用[ElasticsearchRestTemplate](#)

```
@Autowired
private ElasticsearchRestTemplate elasticsearchRestTemplate;
```

案例源码

- [pom文件](#)

```

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/
MLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xs
/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.2.2.RELEASE</version>
  </parent>
  <groupId>com.jianzh5</groupId>
  <artifactId>esalarm</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>esalarm</name>

  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <optional>true</optional>
    </dependency>

    <!--ES-->
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-data-elasticsearch</artifactId>
    </dependency>
  </dependencies>
</project>

```

- 建立ElasticSearch实体类LogDoc

```

@Data
@Document(indexName = "alarm-*",type = "_doc")
public class LogDoc implements Serializable {
    private static final long serialVersionUID = 6320548148250372657L;
    @Id
    private String id;
    private String logLevel;
    private String module;
    private String system;
    private String refCode;
    private String response;
    private String parm;
    private String message;
    private long logTime;
}

```

- 建立LogRepository, 操作底层查询

```
@Repository
public interface LogRepository extends ElasticsearchRepository<LogDoc,String> {

}
```

- 建立接口层LogService

```
public interface LogService {
    /**
     * 根据ID获取对应的实体
     * @param id 日志id
     * @return
     */
    LogDoc getById(String id);

    /**
     * 根据时间范围查找指定格式的日志
     * @author javadaily
     * @date 2019/9/21 9:24
     * @param minRange 开始时间
     * @param maxRange 结束时间
     * @param logLevel 日志级别
     * @return 日志列表
     */
    List<LogDoc> findRangeLogByLevel(DateTime minRange,DateTime maxRange,String logLevel);
}
```

- 建立接口实现层LogServiceImpl

```
/**
 * @author javadaily
 * @date 2019/9/10 15:12
 */
@Service
public class LogServiceImpl implements LogService {
    @Autowired
    private LogRepository logRepository;

    @Autowired
    private ElasticsearchRestTemplate elasticsearchRestTemplate;

    @Override
    public LogDoc getById(String id) {
        Optional<LogDoc> logOptional = logRepository.findById(id);
        return logOptional.orElse(null);
    }

    @Override
    public List<LogDoc> findRangeLogByLevel(DateTime minRange, DateTime maxRange, String logLevel) {
        //需要强制转换成小写
        logLevel = logLevel.toLowerCase();
    }
}
```

```
SearchQuery searchQuery = new NativeSearchQueryBuilder()
    .withQuery(boolQuery()
        //module,system 必须有值才能告警
        .must(existsQuery("module"))
        .must(existsQuery("system"))
        .must(termQuery("logLevel", logLevel))
        .must(rangeQuery("logTime")
            .from(minRange.getMillis())
            .to(maxRange.getMillis()))
    ).build();

return elasticsearchRestTemplate.queryForList(searchQuery, LogDoc.class);
}
}
```

使用[ElasticsearchRepository](#)与[ElasticsearchRestTemplate](#)对ES进行操作，[ElasticsearchRestTemplate](#)的功能比较强大，能支持更为复杂的查询，大家可以根据具体情况选择。

好了，各位朋友们，本期的内容到此就全部结束啦，能看到这里的同学都是优秀的同学，下一个升职加薪的就是你了！

如果觉得这篇文章对你有所帮助的话请扫描下面二维码加个关注。

"转发" 加 "在看"，养成好习惯！咱们下期再见！