

1. 先拉取对应的镜像

`docker pull elasticsearch:7.14.0`

`docker pull kibana:7.14.0`

`docker pull logstash:7.14.0`

2.配置对应的docker 容器的rancher配置

附录:

问题1: 有一个要注意的点, 就是部署了elasticsearch的挂载目录后要给挂载目录 `chmod 777` 权限

问题2: ES启动时出现异常, 报错bootstrap checks failed

先上传本地的elasticsearch.yml, 再去进行相应的挂载

问题X: 日志没有写入

从你提供的日志中, 我没有看到任何明显的错误或警告, 这表明 Logstash 已经成功启动, 并且能够连接到 Elasticsearch。然而, 你提到日志没有写入到 Elasticsearch 中, 这可能是由以下几个原因导致的: 1. **数据源问题**: 请检查你的数据源是否正在发送数据到 Logstash。你可以通过在 Logstash 配置文件中添加一个 stdout 输出来测试这一点。例如:

```
output { stdout { codec => rubydebug } elasticsearch { hosts => ["http://101.40.70.165:30757"] ... }
```

然后, 你可以查看 Logstash 的控制台输出, 看看是否有数据正在流入。

2. **配置问题**: 请检查你的 Logstash 配置文件, 确保你正确配置了 Elasticsearch 输出。你需要确保 `elasticsearch.url` 配置项的值是你的 Elasticsearch 实例的 URL。3. **网络问题**: 请检查 Logstash 服务器和 Elasticsearch 服务器之间的网络连接。你可以尝试在 Logstash 服务器上使用 `curl` 或其他工具直接访问 Elasticsearch 的 URL, 看看是否可以成功连接。4. **Elasticsearch 问题**: 请检查你的 Elasticsearch 实例是否正在运行, 并且能够正常处理请求。你可以通过访问 Elasticsearch 的 `/_status` API 来查看当前的索引状态。如果以上都没有问题, 那么可能需要更详细的日志来进一步排查问题。你可以尝试增加 Logstash 的日志级别, 以获取更多的信息。在 `logstash.yml` 文件中, 将 `log.level` 配置项的值设置为 `debug`, 然后重启 Logstash。例如: `log.level: debug` 希望这些信息能帮助你解决问题。如果你还有其他问题, 或者需要更详细的帮助, 欢迎随时向我提问。

能详细说明如何检测 Elasticsearch 实例是否正在运行吗

USED

Rerun without context

No context当然可以。检查 Elasticsearch 实例是否正在运行, 你可以通过以下几种方式: 1. **使用 curl 命令**: 在命令行中, 你可以使用 `curl` 命令来发送 HTTP 请求到 Elasticsearch。例如, 如果你的 Elasticsearch 实例运行在 `http://101.40.70.165:30757`, 你可以运行以下命令: `curl http://101.40.70.165:30757/_status` 如果 Elasticsearch 正在运行, 你应该会看到一个包含版本号和其他信息的 JSON 响应。

2. 访问 Elasticsearch 的 cat API: Elasticsearch 提供了一个名为 cat 的 API, 可以用来获取集群和索引的实时信息。例如, 你可以访问 `curl http://101.40.70.165:30757/_cat/health` API 来获取集群的健康状态: `curl http://101.40.70.165:30757/_cat/health` 如果 Elasticsearch 正在运行, 你应该会看到一个包含集群健康状态的响应。

3. 使用 Elasticsearch 的客户端库: 如果你在应用程序中使用 Elasticsearch, 你可能会使用 Elasticsearch 的客户端库, 如 Java 客户端、Python 客户端等。这些客户端库通常提供了检查 Elasticsearch 是否可用的方法。例如, 如果你使用的是 Python 客户端, 你可以使用以下代码来检查 Elasticsearch 是否可用: `from elasticsearch import Elasticsearch; es = Elasticsearch(["http://101.40.70.165:30757"]); print(es.ping())` 如果 Elasticsearch 正在运行, `es.ping()` 方法应该会返回 `True`。以上就是检查 Elasticsearch 实例是否正在运行的一些方法。希望这些信息对你有所帮助。如果你还有其他问题, 欢迎随时向我提问。

```
curl http://101.40.70.165:30757/_cat/health 返回 1687917747 02:02:27 docker-cluster yellow 1 1 8 8 0 0 1 0 - 88.9%
```

USED

Rerun without context

No context 你的 Elasticsearch 集群状态显示为 "yellow", 这意味着一些副本尚未被分配到集群中的节点。这可能是由于你的集群中只有一个节点, 而你的索引配置为有一个或多个副本。在这种情况下, 副本无法分配到其他节点, 因此状态为 "yellow"。这不一定意味着有问题, 特别是在开发环境中, 通常只有一个节点。然而, 在生产环境中, 你可能希望有多个节点来提供冗余和提高可用性。你可以通过以下命令查看更多关于未分配副本的信息: `curl http://101.40.70.165:30757/_cat/shards?h=index,shard,primary,state,unassigned,reason | grep UNASSIGNED`

如果你想要将集群状态变为 "green", 你需要添加更多的节点到你的集群, 或者将你的索引的副本数减少到 0。你可以通过以下命令来更新索引的副本数: `curl -XPUT 'http://101.40.70.165:30757/_all/_settings' -d '{`

```
"index" : { "number_of_replicas" : 0 } }'
```

请注意, 这将会将所有索引的副本数设置为 0, 这可能会增加数据丢失的风险。在生产环境中, 你应该确保你有足够的节点来存储所有的副本。希望这些信息能帮助你解决问题。如果你还有其他问题, 欢迎随时向我提问。

报错 "error": "Content-Type header [application/x-www-form-urlencoded] is not supported", "status": 406 [root@host-192-168-128-146 /]#

USED

Rerun without context

Current file 这个错误是因为 Elasticsearch 6.0 以后的版本默认不再接受 `application/x-www-form-urlencoded` 类型的请求。你需要在请求头中指定 `Content-Type: application/json`。你在使用 `curl` 命令时, 如果要发送 JSON 数据, 需要添加 `-H 'Content-Type: application/json'` 参数, 并且使用 `-d` 或 `-XPUT` 参数来提供 JSON 数据。例如: `curl -XPUT 'http://101.40.70.165:30757/_all/_settings' -H 'Content-Type: application/json' -d '{`

```
"index" : { "number_of_replicas" : 0 } }'
```

在这个例子中, `-d` 参数告诉 Elasticsearch 你将发送 JSON 格式的数据, `-H` 参数后面跟着的是你要发送的 JSON 数据。希望这些信息能帮助你解决问题。如果你还有其他问题, 欢迎随时向我提问。