ALIBABA CLOUD

阿里云

阿里云Elasticsearch

Logstash

文档版本：20220616

阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。

4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。

5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。

6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
`Courier字体`	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.什么是阿里云Logstash

阿里云Logstash作为服务器端的数据处理管道，提供了100%兼容开源Logstash的能力。Logstash能够动态地从多个来源采集数据、转换数据，并且将数据存储到所选位置。通过输入、过滤和输出插件，Logstash可以加工和转换任何类型的事件。

为什么选择阿里云Logstash

阿里云Logstash除了支持所有官方预置插件外，还致力于打造包含logstash-input-sls、logstash-input-oss、logstash-output-oss等适用各类场景的插件中心，为您提供更为强大的数据处理和搬迁能力，实现云上数据生态打通。

在阿里云ELK（Elasticsearch、Logstash、Kibana）生态下，Elasticsearch作为实时分布式搜索和分析引擎，Logstash提供了数据采集、转换、优化和输出的能力，Kibana提供了强大的可视化界面，可以被广泛应用于实时日志处理、全文搜索和数据分析等领域。

Logstash数据传输原理

1. 数据采集与输入：Logstash支持各种输入选择，能够以连续的流式传输方式，轻松地从日志、指标、Web应用以及数据存储中采集数据。
2. 实时解析和数据转换：通过Logstash过滤器解析各个事件，识别已命名的字段来构建结构，并将它们转换成通用格式，最终将数据从源端传输到存储库中。
3. 存储与数据导出：Logstash提供多种输出选择，可以将数据发送到指定的地方。

产品特性

特性	相关文档
快速部署、轻松管理、简化复杂的运维操作，支持灵活扩容。	• 快速入门 • 升配集群
部署在逻辑隔离的专有网络中，提高产品的安全性。	创建阿里云Logstash实例
支持上传自定义扩展文件，提供对所有扩展文件进行管理的能力。	配置扩展文件
开放灵活的插件中心，集成官方全部Input、Output、Filter插件。同时支持日志服务SLS、OSS等阿里云产品输入或输出插件。	插件配置概述
支持通过配置NAT网关，实现与公网的连通。	配置NAT公网数据传输
通过配置文件集中式管理管道。支持在管道配置完成后，直接在控制台上查看管道配置的输出结果。	• 通过配置文件管理管道 • 使用Logstash管道配置调试功能

相关文档

数据同步

- MySQL数据同步：通过Logstash将RDS MySQL数据同步至Elasticsearch
- MaxCompute数据同步：通过阿里云Logstash将MaxCompute数据同步至Elasticsearch

- PolarDB-X（DRDS）数据同步：通过Logstash将PolarDB-X（DRDS）数据同步至Elasticsearch

数据迁移

- 自建Elasticsearch数据迁移：通过阿里云Logstash将自建Elasticsearch数据迁移至阿里云
- 腾讯云Elasticsearch数据迁移：腾讯云Elasticsearch数据迁移至阿里云

日志分析

使用Filebeat+Kafka+Logstash+Elasticsearch构建日志分析系统

2.快速入门

本文为您介绍如何创建一个阿里云Logstash实例，并通过Logstash的管道配置，完成阿里云Elasticsearch实例间的数据同步。

背景信息

在开始本文操作前，请先了解以下背景信息：

- 什么是阿里云Elasticsearch
- 什么是阿里云Logstash

前提条件

- 注册阿里云账号。

  具体操作，请参见账号注册。
- 创建专有网络和虚拟交换机。

  具体操作，请参见搭建IPv4专有网络。

使用限制

- 源Elasticsearch、Logstash和目标Elasticsearch实例在同一专有网络。如果不在同一专有网络，需要通过配置NAT网关实现与公网的连通，详细信息请参见配置NAT公网数据传输。
- 源Elasticsearch、Logstash和目标Elasticsearch实例版本需满足兼容性要求，详细信息请参见产品兼容性。

操作流程

1. 准备工作

   创建源和目标Elasticsearch实例、开启目标Elasticsearch实例的自动创建索引功能、准备测试数据。
2. 步骤一：创建阿里云Logstash实例

   创建阿里云Logstash实例，等待实例状态变为正常后，才可以创建并运行管道任务。
3. 步骤二：创建并运行管道任务

   创建并配置阿里云Logstash管道任务，运行任务完成数据同步。
4. 步骤三：查看数据同步结果

   通过目标Elasticsearch实例的Kibana控制台，查看数据同步结果。

准备工作

1. 创建阿里云Elasticsearch实例。
   i. 登录阿里云Elasticsearch控制台。
   ii. 在左侧导航栏，单击**Elasticsearch实例**。

iii. 在**Elasticsearch实例**页面，创建2个阿里云Elasticsearch实例。

创建的2个阿里云Elasticsearch实例，分别作为Logstash的input和output，具体操作请参见创建阿里云Elasticsearch实例。本文创建的实例版本为通用商业版6.7，使用的数据迁移方案为：阿里云Elasticsearch 6.7.0 > 阿里云Logstash 6.7.0 > 阿里云Elasticsearch 6.7.0，提供的脚本仅适用于该数据迁移方案，其他方案不保证兼容。创建的阿里云Elasticsearch实例的具体配置如下。

> **说明** 如果您使用的是其他方案，可参见产品兼容性判断是否存在兼容性问题。如果存在，可升级实例版本或新购实例。

> **说明** 访问阿里云Elasticsearch实例的账号默认为elastic（本文以此为例），如果需要使用自建用户，要给予自建用户相应的角色和权限，详细信息请参见通过Elasticsearch X-Pack角色管理实现用户权限管控。

2. 开启目标阿里云Elasticsearch实例的自动创建索引功能。

具体操作，请参见配置YML参数。

> **说明** 阿里云Elasticsearch为了保证用户操作数据的安全性，默认将**自动创建索引**配置设置为**不允许**。阿里云Logstash在传输数据的时候，使用提交数据的方式创建索引，而不是Create index API的方式。所以在使用阿里云Logstash上传数据之前，需要先把集群的**自动创建索引**设置为**允许**，或提前创建好索引和Mapping。

3. 准备测试数据。

进入源阿里云Elasticsearch实例的Kibana控制台，在**Dev Tools**页面的**Console**页签下，执行如下命令创建待同步的索引和文档。

> **注意**
>
> - 进入Kibana控制台的具体步骤，请参见登录Kibana控制台。
> - 以下脚本以Elasticsearch 6.7版本为例，仅供测试。7.0及以上版本的示例脚本，请参见Elasticsearch快速入门。

i. 创建名称为my_index，类型为my_type的索引。

```
PUT /my_index
{
    "settings" : {
      "index" : {
        "number_of_shards" : "5",
        "number_of_replicas" : "1"
      }
    },
    "mappings" : {
        "my_type" : {
            "properties" : {
              "post_date": {
                    "type": "date"
               },
              "tags": {
                    "type": "keyword"
               },
                "title" : {
                     "type" : "text"
                }
            }
        }
    }
}
```

ii. 在my_index索引中插入一个名称为1的文档。

```
PUT /my_index/my_type/1?pretty
{
  "title": "One",
  "tags": ["ruby"],
  "post_date":"2009-11-15T13:00:00"
}
```

iii. 在my_index索引中插入一个名称为2的文档。

```
PUT /my_index/my_type/2?pretty
{
  "title": "Two",
  "tags": ["ruby"],
  "post_date":"2009-11-15T14:00:00"
}
```

步骤一：创建阿里云Logstash实例

1. 进入Logstash实例页面。
   i. 登录阿里云Elasticsearch控制台。
   ii. 在顶部菜单栏，选择与目标阿里云Elasticsearch实例相同的地域。
   iii. 在左侧导航栏，单击**Logstash实例**。
2. 在**Logstash实例**页面，单击**创建**。
3. 在购买页面的前三个配置页面，完成实例启动配置。

本文选择实例的付费模式为按量付费，版本为6.7，其余配置均保持默认。更多配置信息，请参见创建阿里云Logstash实例。

> 说明
> - 在前期程序研发或功能测试期间，建议购买**按量付费**实例测试。
> - 购买**包年包月**实例，可以享受优惠条件。

4. 单击**下一步：确认订单**，预览实例配置。

   配置不符合预期时，可单击图标修改。

   本文的实例配置预览如下图。

5. 选中服务协议，单击**立即购买**。
6. 提示开通成功后，单击**管理控制台**。
7. 在顶部菜单栏，选择实例所在地域。在左侧导航栏，单击**Logstash实例**，进入**Logstash实例**页面，查看创建成功的实例。

步骤二：创建并运行管道任务

等到创建的Logstash实例状态变为**正常**后，您可以创建并运行管道任务同步数据。

1. 在**Logstash实例**页面，单击目标实例右侧**操作**列下的**管道管理**。
2. 在**管道列表**区域，单击**创建管道**。
3. 输入**管道ID**和**Config配置**。

   本文使用的Config配置如下。

```
input {
    elasticsearch {
        hosts => ["http://es-cn-0pp1f1y5g000h****.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "your_password"
        index => "*,-.monitoring*,-.security*,-.kibana*"
        docinfo => true
    }
}
filter {}
output {
    elasticsearch {
        hosts => ["http://es-cn-mp91cbxsm000c****.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "your_password"
        index => "%{[@metadata][_index]}"
        document_type => "%{[@metadata][_type]}"
        document_id => "%{[@metadata][_id]}"
    }
    file_extend {
        path => "/ssd/1/ls-cn-v0h1kzca****/logstash/logs/debug/test"
    }
}
```

参数	说明
hosts	阿里云Elasticsearch服务的访问地址。input中为 `http://<源实例ID>.elasticsearch.aliyuncs.com:9200` ；output中为 `http://<目标实例ID>.elasticsearch.aliyuncs.com:9200` 。
user	访问阿里云Elasticsearch服务的用户名，默认为elastic。
password	对应用户的密码。elastic用户的密码在创建实例时设定，如果忘记可进行重置，重置密码的注意事项和操作步骤请参见重置实例访问密码。
index	指定同步索引名。设置为 `*,-.monitoring*,-.security*,-.kibana*` ，表示同步除了 `.` 开头的系统索引外的所有索引。 `%{[@metadata][_index]}` ，表示匹配元数据中的index，即同步后索引的名称和源索引名称相同。 **说明** 系统索引一般用来存储Elasticsearch集群的监控日志，无需同步。
docinfo	设置为true，将会提取Elasticsearch文档的元信息，例如index、type和id。
document_type	指定同步后索引的类型。设置为 `%{[@metadata][_type]}` ，表示匹配元数据中的type，即同步后索引的类型和源索引类型相同。
document_id	指定同步后文档的ID。设置为 `%{[@metadata][_id]}` ，表示匹配元数据中的id，即同步后文档的ID和源文档ID相同。

参数	说明
file_extend	可选，用来开启调试日志功能，并通过path参数配置调试日志的输出路径。建议您配置该参数，配置后，可直接在控制台上查看输出结果。如果未配置，需要去目标端确认输出结果，再返回控制台修改，这样会耗费大量的时间和人力。详细信息，请参见使用Logstash管道配置调试功能。 **注意** 使用file_extend参数前，需要先安装logstash-output-file_extend插件。具体操作，请参见安装或卸载插件。其中的path参数默认为系统指定路径，请勿修改。您也可以单击**开启配置调试**获取path路径。

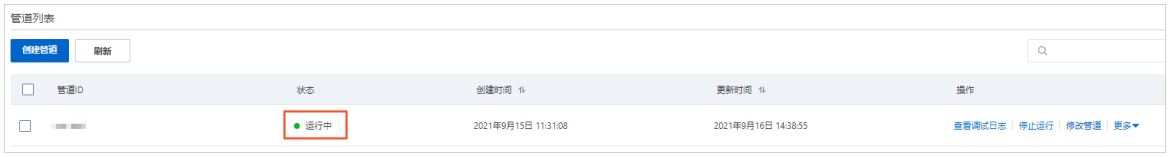
Config配置的结构及支持的数据类型的详细信息（不同版本支持的数据类型可能不同），请参见Structure of a Config File。

4. 单击**下一步**，配置管道参数。

   在配置的管道参数中，**管道工作线程**配置为实例的CPU核数，其他参数均为默认值。详细参数说明，请参见通过配置文件管理管道。

5. 单击**保存**或者**保存并部署**。
   - **保存**：将管道信息保存在Logstash里并触发实例变更，配置不会生效。保存后，系统会返回**管道管理**页面。可在**管道列表**区域，单击**操作**列下的**立即部署**，触发实例重启，使配置生效。
   - **保存并部署**：保存并且部署后，会触发实例重启，使配置生效。

6. 在创建成功提示框中，单击**确认**。
   确认后，可在**管道列表**中查看创建成功的管道。等待实例变更完成，并且管道的**状态**显示为**运行中**时，表示阿里云Logstash开始执行同步任务。



步骤三：查看数据同步结果

数据同步任务配置完成并开始运行后，您可以通过目标阿里云Elasticsearch的Kibana控制台，查看数据同步结果。

1. 登录目标阿里云Elasticsearch实例的Kibana控制台，根据页面提示进入Kibana主页。

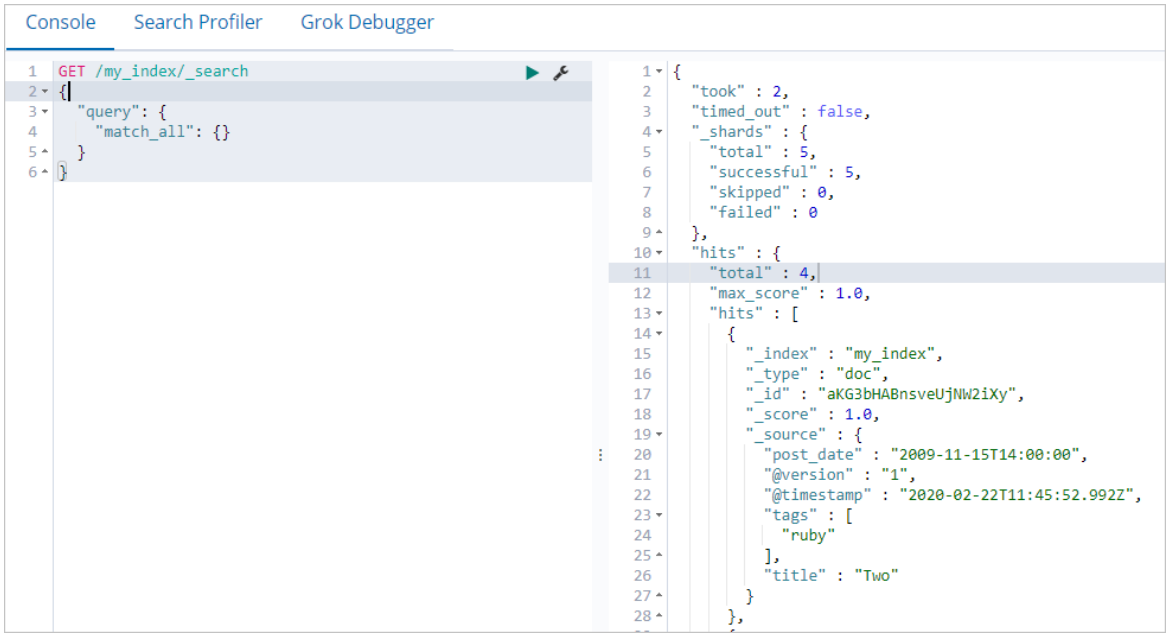
   登录Kibana控制台的具体操作，请参见登录Kibana控制台。

   说明 本文以阿里云Elasticsearch 6.7.0版本为例，其他版本操作可能略有差别，请以实际界面为准。

2. 在左侧导航栏，单击**Dev Tools**。
3. 在**Console**中，执行如下命令查看数据同步结果。

```
GET /my_index/_search
{
  "query": {
    "match_all": {}
  }
}
```

预期结果如下。



如果源端和目标端数据一致，表示数据同步成功。您也可以通过 `GET _cat/indice?v` 命令，查看源端和目标端相同索引的大小是否一致，来判断数据是否同步成功。

相关文档

- 了解如何配置集群监控：
  - 配置自定义报警策略
  - 配置X-Pack监控
- 了解如何将第三方Elasticsearch数据迁移至阿里云：
  - 通过阿里云Logstash将自建Elasticsearch数据迁移至阿里云
  - 通过Logstash将自建Elasticsearch数据全量或增量迁移至阿里云
  - 腾讯云Elasticsearch数据迁移至阿里云
- 了解如何将PolarDB-X数据同步至阿里云Elasticsearch：通过Logstash将PolarDB-X（DRDS）数据同步至Elasticsearch。
- 了解如何将MaxCompute数据同步至阿里云Elasticsearch：通过阿里云Logstash将MaxCompute数据同步至Elasticsearch。

常见问题

- Logstash FAQ
- Logstash数据写入问题排查方案

3.实例管理
3.1. 创建阿里云Logstash实例

本文介绍如何创建阿里云Logstash实例，并提供创建实例时的详细参数说明。

前提条件

您已完成以下操作：

- 注册阿里云账号。

  具体操作，请参见账号注册。
- 开通专有网络和虚拟交换机。

  具体操作，请参见搭建IPv4专有网络。

操作步骤

1. 前往实例创建页面。
2. 选择基础配置。

参数	说明
付费模式	支持包年包月和按量付费两种购买方式，请根据需求选择合适的方式： ○ 按量付费：在前期程序研发或功能测试期间，建议购买按量付费类型的实例进行测试。 支持在控制台手动释放实例。 ○ 包年包月：目前在购买包年包月类型的实例时，可以享受优惠条件。购买后，支持5天内退余款。超过5天后，将不再支持退款。 支持手动续费和自动续费，详情请参见Logstash续费。不支持在控制台手动释放实例。
Logstash版本	支持7.4和6.7版本。

3. 单击**下一步：集群配置**，选择集群配置。
   - **地域和可用区**

     阿里云Logstash支持的地域和可用区如下。

国家	地域	可用区
	华北 2（北京）	可用区C、可用区D、可用区E、可用区F、可用区G、可用区H、可用区I、可用区J、可用区K
	华东1（杭州）	可用区E、可用区F、可用区G、可用区H、可用区I、可用区J、可用区K
	华北1（青岛）	可用区B、可用区C

中国国家	地域	可用区
	华东 2（上海）	可用区B、可用区D、可用区E、可用区F、可用区G、可用区L
	华南 1（深圳）	可用区A、可用区B、可用区C、可用区D、可用区E、可用区F
	华北3（张家口）	可用区A、可用区B、可用区C
	中国（香港）	可用区B、可用区C、可用区D
亚太	新加坡	可用区A、可用区B、可用区C
	澳大利亚（悉尼）	可用区A、可用区B
	马来西亚（吉隆坡）	可用区A、可用区B
	印度尼西亚（雅加达）	可用区A、可用区B
	日本（东京）	可用区A、可用区B
欧洲与美洲	美国（弗吉尼亚）	可用区A、可用区B
	美国（硅谷）	可用区A、可用区B
	德国（法兰克福）	可用区A、可用区B
	英国（伦敦）	可用区A、可用区B
中东与印度	印度（孟买）	可用区A、可用区B

- **实例规格**

  单击**修改**，可展开Logstash节点配置，并根据需求修改。

参数	说明
规格族	根据节点的CPU和内存配比，阿里云Logstash提供了1:1、1:2、1:4和1:8四种比例的规格族，各规格族支持的规格如下（实际以界面为准）： ■ 1:1规格族：4核4 GB、8核8 GB、12核12 GB、16核16 GB ■ 1:2规格族：2核4 GB、16核32 GB、32核64 GB ■ 1:4规格族：2核8 GB、4核16 GB、8核32 GB、16核64 GB ■ 1:8规格族：2核16 GB、4核32 GB、8核64 GB
存储类型	支持**SSD云盘**和**高效云盘**： ■ **SSD云盘**（默认）：支持最大2 TB的存储空间，适合拥有高IOPS，数据响应度较高的在线分析和搜索场景。 ■ **高效云盘**：支持最大5 TB的存储空间，提供较为低廉的存储能力，适合大规模数据量的日志及分析场景。

参数	说明
单节点存储空间	单节点存储空间与节点的存储类型有关： ■ **SSD云盘**：最大支持2048 GB（2 TB），最小支持20 GB。 ■ **高效云盘**：最大支持5120 GB（5 TB）。 **注意** 高效云盘扩容时最大支持扩容到2 TB。2.5 TB以上的高效云盘通过磁盘阵列及RAID 0的方式提供服务，不支持扩容。
数量	表示需要购买几个数据节点，可选范围为1~20个。

4. 单击**下一步：网络与系统配置**，选择网络配置。

参数	说明
网络类型	目前仅支持**专有网络**。
专有网络	选择对应区域下的专有网络。 **注意** 如果您需要通过ECS访问Logstash实例，且该ECS位于专有网络下，则Logstash实例与ECS实例必须在同一个专有网络下。
虚拟交换机	选择对应专有网络下，与Logstash实例在相同可用区下的虚拟交换机。 **注意** 所选交换机下的可用IP数必须大于等于50。
计费周期	仅当**付费模式**为**包年包月**时显示。默认购买时长为一个月。可以自定义选择购买时长（单位：1~9月，1~3年）。
到期自动续费	仅当**付费模式**为**包年包月**时显示。选中后后，可开启自动续费功能。 ○ 按月购买：自动续费周期为1个月。 ○ 按年购买：自动续费周期为1年。

5. 单击**下一步：确认订单**，预览实例配置。

   配置不符合预期时，可单击图标进行修改。

6. 勾选服务协议，单击**立即购买**。
7. 提示开通成功后，单击**管理控制台**。
8. 在左侧导航栏， 单击**Logstash实例**，在实例列表中查看创建成功的实例。

相关文档

CreateLogstash

3.2. 实例列表概览

logstash实例列表

阿里云Logstash的实例列表展示了实例的基本信息，并提供了创建实例、刷新实例状态、管理管道、管理实例等功能的入口。

登录阿里云Elasticsearch控制台，在左侧导航栏单击**Logstash实例**，系统直接进入Logstash的实例列表页面。实例列表页面展示了当前地域您账号下的所有阿里云Logstash实例，并提供了以下操作功能。

功能	说明
创建实例	单击**创建**，可在购买页面购买实例。详细信息，请参见创建阿里云Logstash实例。
刷新实例	单击**刷新**，可获取实例的实时状态。实例创建后，默认为**待生效**状态，可单击**刷新**查看实例的最新状态，当**状态**变为**正常**时，即可正常使用实例。
搜索实例	当实例太多时，您可以通过搜索功能快速定位到目标实例。阿里云Logstash支持按照**实例ID**和**实例名称**搜索实例，搜索时支持模糊匹配。
查看实例的列表信息	包括**实例ID/名称**、**状态**、**版本**、**数据节点数**、**规格**、**可用区**、**付费类型**、**网络类型**和**创建时间**等。
查看实例的基本信息	单击实例ID，在**基本信息**页面查看实例的基本信息。详细信息，请参见查看实例的基本信息。
管理管道	单击右侧**操作**列下的**管道管理**，可在**管道管理**页面创建并配置管道。详细信息，请参见通过配置文件管理管道。
管理实例	单击右侧**操作**列下的**实例管理**，可在实例管理页面进行集群配置、插件配置、网络与安全配置、集群监控、日志查询和管道管理操作。
转包年包月	此功能仅适用于按量付费类型的实例。选择右侧**操作**列下的 > **转包年包月**，可在确认订单页面变更付费类型。详细信息，请参见按量付费转包年包月。
升配	选择右侧**操作**列下的 > **升配**，可在变配页面修改集群的配置。详细信息，请参见升配集群。
释放实例	选择右侧**操作**列下的 > **释放实例**，在**释放实例**页面确认后，即可释放实例。 详细信息，请参见释放实例。 警告 实例释放后，数据将不可恢复，请谨慎操作。

3.3. 查看实例的基本信息

当您需要使用阿里云Logstash实例的访问地址、端口号、版本、状态等信息时，可在实例的基本信息页面获取。本文为您介绍实例基本信息页面的参数。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在**基本信息**页面，查看实例的基本信息和端口信息。

类别	名称	描述
基本信息	**实例ID**	实例的唯一标识。
	创建时间	实例创建的时间。
	名称	实例的名称默认与实例ID相同。支持自定义实例名称，也支持按照名称搜索实例。
	状态	实例的状态。支持正常（绿色）、生效中（黄色）、变更中断（红色）和失效（灰色）。
	地域	实例所在的地域。
	可用区	实例所在的可用区。
	版本	支持6.7.0和7.4.0版本。
	付费类型	支持**包年包月**和**后付费**类型。
	专有网络	实例所属的专有网络。
	虚拟交换机ID	实例所属的交换机。
	节点规格	实例中包含的数据节点的CPU和内存配比。
	节点数量	实例中包含的数据节点的数量。
	存储规格	实例中包含的数据节点的存储类型，支持**高效云盘**和**SSD云盘**： ○ **高效云盘**：支持最大5 TB的存储空间，提供较为低廉的存储能力，适合大规模数据量的日志及分析场景。 ○ **SSD云盘**：支持最大2 TB的存储空间，适合拥有高IOPS，数据响应度较高的在线分析和搜索场景。
	节点容量	实例中包含的单个数据节点的存储空间。
	可维护时间段	允许阿里云进行维护操作的时间段，默认为关闭状态。

类别	名称	描述
端口信息	**访问地址**	实例的私网IP地址。当您需要配置公网访问Logstash实例时，可能会用到该地址，详细信息请参见配置NAT公网数据传输。 **注意** • 只有同一VPC下的机器，才可以通过该地址访问实例。 • 该地址默认禁止ping，可以使用telnet测试。
	私网端口	实例的私网端口，固定为**9600**。 **注意** 该端口为Logstash实例的服务端口，可在访问实例时使用。管道配置时，如果需要在input中配置port，请使用8000~9000端口，详细信息请参见通过配置文件管理管道。

相关文档

获取Logstash实例基本信息的API文档：DescribeLogstash

3.4. 释放实例

释放实例功能仅支持释放按量付费和已到期的包年包月实例。包年包月实例到期前，需申请退款后再释放。本文介绍释放按量付费实例的相关操作。

注意事项

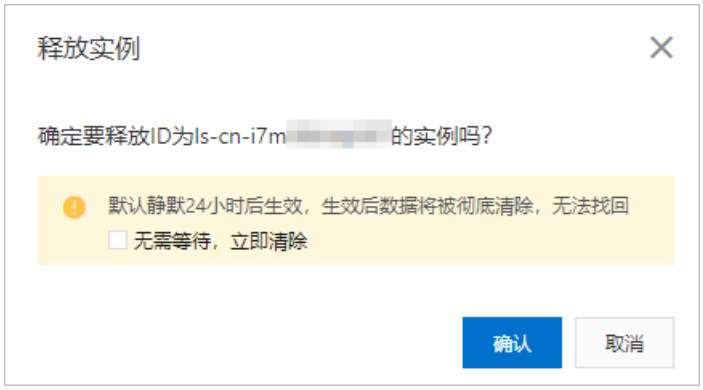
实例释放后数据无法恢复，实例中所包含的管道会被删除，请谨慎操作。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 在顶部菜单栏处，选择地域。
3. 在左侧导航栏，单击**Logstash实例**。
4. 在实例列表中，选择待释放实例右侧**操作**列下的**更多 > 释放实例**。

> **注意** 阿里云Logstash不支持释放变更中的实例。如果要释放，请先停止变更，或等实例变更完成后再释放。

5. 在弹出的对话框中，根据需求选择是否要选中**无需等待，立即清除**，单击**确认**。



- 不选中

  该实例会被冻结24小时后，再彻底清除数据，期间实例仍在实例列表中显示，您可以选择恢复实例或立即释放：

  - 选择**更多 > 恢复实例**，确认后即可恢复实例服务，并继续计费。
  - 选择**更多 > 立即清除**，确认后实例被释放，数据被彻底清除，同时在**实例列表**中清除该实例信息。

- 选中

  彻底清除所有数据，且实例不再显示在**实例列表**中。

相关文档

删除Logstash实例的API文档：DeleteLogstash

4.集群变更

4.1. 重启实例或节点

重启logstash

当您修改了实例或节点的配置、异常问题导致服务不可用或出现其他状况时，可能需要重启阿里云Logstash实例或节点才能生效。本文介绍如何通过控制台重启实例或节点。

前提条件

实例的**状态**为**正常**（绿色），且资源使用率不是很高。

> **说明** 资源使用率可在**集群监控**页面查看，例如**节点磁盘使用率**、**节点HeapMemory使用率**、**节点CPU使用率**和**节点load_1m**。详细信息，请参见配置自定义报警策略。

操作说明

重启分为实例级别重启和节点级别重启，实例级别重启是指重启实例中所包含的所有节点，节点级别重启是指重启所选的单个节点。阿里云Logstash的重启方式和相关注意事项与阿里云Elasticsearch类似。详细信息，请参见重启实例或节点。

相关文档

重启Logstash实例或节点的API文档：RestartLogstash

4.2. 升配集群

随着业务的发展，您可能对集群配置的要求越来越高。当集群的配置无法满足业务需求时，可通过集群升配功能，升级集群配置。本文介绍阿里云Logstash集群升配的注意事项及操作方法。

注意事项

升配集群会触发集群重启，重启时间与集群规格、数据结构和大小等因素有关，建议在业务低峰期操作。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 在顶部菜单栏处，选择地域。
3. 在左侧导航栏，单击**Logstash实例**，然后在Logstash实例列表中单击目标实例ID。
4. 在**基本信息**页面，单击右侧的**集群升配**。
5. 在**变配**页面，单击**变更规格**区域中**Logstash节点**下的**修改**。

   > **说明** 除了以上操作方式外，您还可以在Logstash实例列表页面，单击目标实例右侧**操作**列下的**更多 > 升配**，进入集群**变配**页面。

6. 在展开的节点配置中，按需修改**规格族**、**存储类型**、**单节点存储空间**、**数量**等配置。

   参数详情，请参见创建阿里云Logstash实例。

   变配页面的**当前配置**区域，展示了当前实例的配置信息，便于您在执行升配操作时参考。

7. 选中**服务协议**，单击**立即购买**。

购买后，集群会重启，重启成功后即可完成集群升配。

4.3. 查看实例任务进度详情

您可以通过任务列表查看正在进行中的任务信息，例如实例的创建进度和重启进度。

前提条件

确保实例处于**生效中**状态。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 在顶部菜单栏处，选择地域。
3. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
4. 单击右上角的≡图标。
5. 在**任务列表**页面，单击**展开详情**，查看各任务的进度详情。

6. （可选）如果您需要查看日志信息，单击**查看日志**，会跳转到**日志查询**页面，即可查看实例的操作日志。
7. （可选）如果您需要暂停变更任务，单击**中断变更**。变更中断后，可单击**恢复变更**，继续完成之前的实例变更任务。

注意

- 实例处于**变更中断**状态时，可能会导致集群服务受到影响，此时可通过二次变更或手动操作恢复变更 。二次变更支持集群升配和插件管理。
- 触发**恢复变更**操作后，整个重启流程会重新执行一遍，集群中的节点会再进行一次重启，请耐心等待。
- 集群变更操作会触发集群重启，建议在业务低峰期操作。

5.集群配置

5.1. 配置扩展文件

logstash驱动文件

当您需要在阿里云Logstash的配置文件中定义驱动文件时，可通过扩展文件配置功能，上传所需的驱动文件。同时扩展文件配置功能也提供了对所有扩展文件进行管理的能力。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**集群配置**。
4. 在**扩展文件配置**区域，单击**上传扩展文件**右侧的**管理**。

5. 在**修改配置**页面，单击下方的**配置**。
6. 单击**上传文件**，在弹出框中选择本地文件进行上传。

   阿里云Logstash支持批量上传，且上传前会对文件进行文件名及md5值校验（文件后缀必须是.jar，文件名不支持中文，且长度不超过100个字符），校验失败会进行提示，无法上传。

   目前，阿里云Logstash支持MySQL JDBC、PolarDB O引擎JDBC和PostgreSQL JDBC三种类型的驱动文件，下载地址请参见下表。

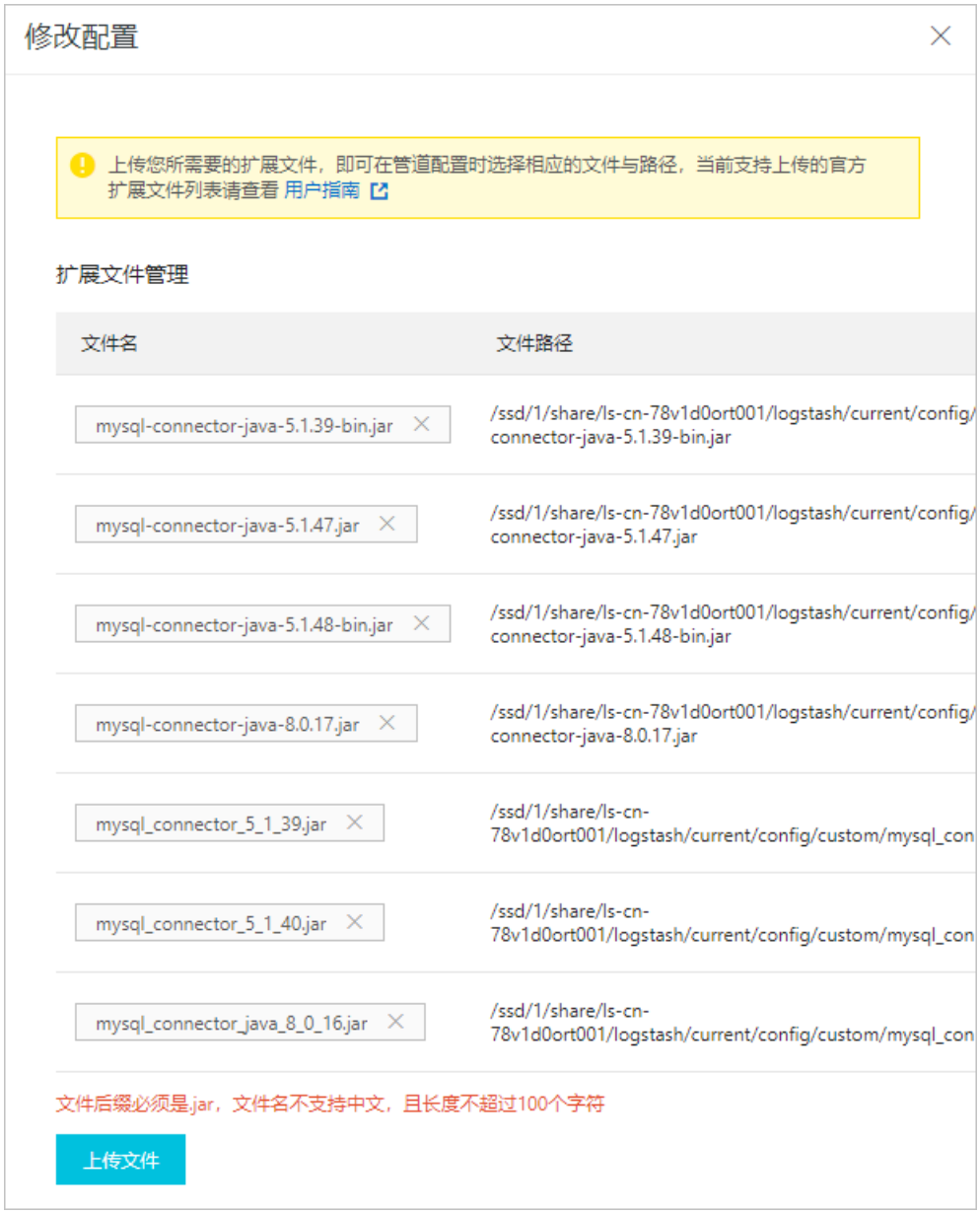
驱动文件类型	驱动文件

驱动文件类型	驱动文件
MySQL JDBC driver	○ mysql-connector-java-5.1.27.jar ○ mysql-connector-java-5.1.35.jar ○ mysql-connector-java-5.1.39-bin.jar ○ mysql-connector-java-5.1.39.jar ○ mysql-connector-java-5.1.43.jar ○ mysql-connector-java-5.1.47.jar ○ mysql-connector-java-5.1.48.jar ○ mysql-connector-java-5.1.9.jar ○ mysql-connector-java-6.0.2.jar ○ mysql-connector-java-6.0.6.jar ○ mysql-connector-java-8.0.11.jar ○ mysql-connector-java-8.0.17.jar ○ mysql-connector-java-8.0.18.jar
PolarDB O引擎JDBC driver	PolarDB O引擎JDBC.zip **说明** 阿里云PolarDB O引擎提供了兼容Java 6、Java 7和Java 8三个Java版本的JDBC驱动，分别对应以下三个JAR包： ○ polardb-jdbc16.jar ○ polardb-jdbc17.jar ○ polardb-jdbc18.jar 您可以根据应用使用的JDK版本选择合适的JDBC。
PostgreSQL JDBC driver	○ postgresql-42.0.0.jar ○ postgresql-42.1.4.jar ○ postgresql-42.2.0.jar ○ postgresql-42.2.1.jar ○ postgresql-42.2.8.jar ○ postgresql-42.2.10.jar ○ postgresql-42.2.13.jar

警告 修改扩展文件会触发实例重启，请在不影响业务的情况下继续执行以下步骤。

7. 单击**保存**。
   保存后，系统返回**扩展文件配置**页面，并触发集群重启。重启完成后，即可完成扩展文件的添加。
8. （可选）再次单击**上传扩展文件**右侧的**管理**，在**修改配置**页面查看已上传的扩展文件信息。

   扩展文件信息包括**文件名**和**文件路径**。单击文件右侧的×图标，可移除对应文件。



注意

- 为了提升安全性，如果在配置管道时使用了JDBC驱动，需要在 `jdbc_connection_string` 参数后面添加 `allowLoadLocalInfile=false&autoDeserialize=false` ，否则在添加Logstash配置文件时，调度系统会抛出校验失败的提示，例如 `jdbc_connection_string => "jdbc:mysql://xxx.drds.aliyuncs.com:3306/test-database?allowLoadLocalInfile=false&autoDeserialize=false"` 。
- 如果不再使用扩展文件，可在**修改配置**页面，单击下方的**配置**，再单击扩展文件右侧的 图标，移除对应的扩展文件。

相关文档

- 获取Logstash实例扩展文件配置的API：ListExtendfiles
- 更新Logstash实例扩展文件配置的API：UpdateExtendfiles

5.2. 配置YML文件

配置logstash yml文件

当您需要通过参数设置来控制阿里云Logstash执行的任务时，可通过Logstash的配置YML文件功能，修改YML文件的参数。

注意事项

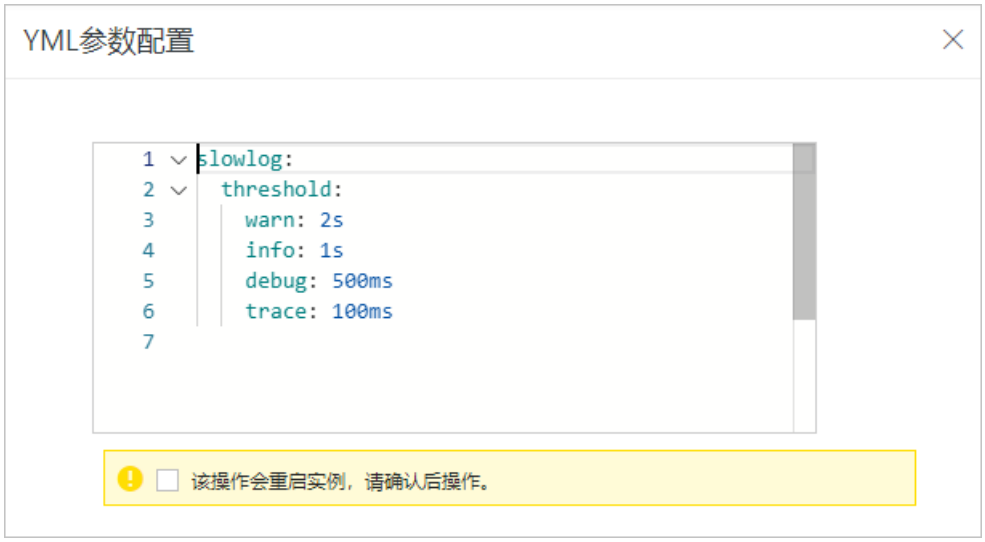
在修改YML参数配置时，请注意：

- 为了方便后续排查与定位阿里云Logstash问题，YML文件配置中默认开启了慢日志，请不要移除该慢日志配置。
- 为了保证服务运行的稳定性，阿里云Logstash不支持修改以下参数值。

```
node.name
path.data
path.config
http.host
http.port
log.level
path.logs
path.plugins
log.format
path.settings
pipeline.workers
xpack.management.enabled
xpack.management.pipeline.id
xpack.management.elasticsearch.username
xpack.management.elasticsearch.password
xpack.management.elasticsearch.hosts
xpack.monitoring.enabled
xpack.monitoring.elasticsearch.username
xpack.monitoring.elasticsearch.password
xpack.monitoring.elasticsearch.hosts
```

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**集群配置**。
4. 在**集群配置**页面，单击**YML文件配置**右侧的**修改配置**。
5. 在**YML参数配置**页面，根据实际业务场景需求修改YML参数配置。



配置参数的详细信息，请参见官方Logstash 6.7.0参考文档。

> **警告** 修改YML文件需要重启阿里云Logstash实例才能生效，为保证您的业务不受影响，请确认后再执行以下步骤。

6. 勾选**该操作会重启实例，请确认后操作**，单击**确定**。
   确定后，阿里云Logstash实例会重启。重启时，可在任务列表中查看任务进度，具体操作请参见查看实例任务进度详情。重启成功后，即可完成YML文件的配置。

相关文档

UpdateLogstashSettings

6.插件配置

6.1. 插件配置概述

阿里云Logstash支持100余款系统默认插件，包含开源和自研Logstash插件，能够提升集群在数据传输和处理、日志调试等各方面的能力。本文介绍阿里云Logstash支持的系统默认插件。

阿里云Logstash仅支持系统默认插件，不支持自定义插件。系统默认插件为阿里云Logstash预置的插件，您可以根据需求卸载或安装。具体操作，请参见安装或卸载插件。

阿里云Logstash支持的插件如下：

- 自研插件

类别	名称	说明	介绍
input	logstash-input-datahub	从阿里云流式数据服务DataHub读取数据。	logstash-input-datahub插件使用说明
	logstash-input-maxcompute	从阿里云大数据计算服务MaxCompute读取数据。	logstash-input-maxcompute插件使用说明
	logstash-input-oss	从阿里云对象存储服务OSS读取数据。	logstash-input-oss插件使用说明
	logstash-input-sls	从阿里云日志服务SLS读取日志。	logstash-input-sls插件使用说明
output	logstash-output-datahub	传输数据至阿里云流式数据服务DataHub。	logstash-output-datahub插件使用说明
	logstash-output-file_extend	直接在控制台上查看管道配置的输出结果。	使用Logstash管道配置调试功能
	logstash-output-oss	批量传输数据至阿里云对象存储服务OSS。	logstash-output-oss插件使用说明

- 开源插件

类别	名称	说明	介绍
	logstash-input-azure_event_hubs	使用Azure事件中心中的事件。	Azure Event Hubs plugin
	logstash-input-beats	从Elastic Beats框架中接收事件。	Beats input plugin
	logstash-input-dead_letter_queue	从Logstash的死信队列中读取事件。	Dead_letter_queue input plugin

类别	名称	说明	介绍
input	logstash-input-elasticsearch	从Elasticsearch集群中读取数据。	Elasticsearch input plugin
	logstash-input-exec	定期运行Shell命令，将Shell命令的全部输出作为事件捕获。	Exec input plugin
	logstash-input-ganglia	通过用户数据协议UDP（User Datagram Protocol）从网络读取Ganglia包。	Ganglia input plugin
	logstash-input-gelf	在网络上将GELF格式信息作为事件读取。	Gelf input plugin
	logstash-input-generator	生成随机日志事件。	Generator input plugin
	logstash-input-graphite	从Graphite工具读取指标。	Graphite input plugin
	logstash-input-heartbeat	生成心跳消息。	Heartbeat input plugin
	logstash-input-http	通过HTTP或HTTPS接收单行或多行事件。	Http input plugin
	logstash-input-http_poller	调用HTTP API，将输出解码为事件，并发送事件。	Http_poller input plugin
	logstash-input-imap	从IMAP服务器读取邮件。	Imap input plugin
	logstash-input-jdbc	通过JDBC程序界面，将任一数据库数据读取到Logstash中。	Jdbc input plugin
	logstash-input-kafka	从Kafka主题读取事件。	Kafka input plugin
	logstash-input-pipe	从长时间运行的管道命令中流式读取事件。	Pipe input plugin
	logstash-input-rabbitmq	从RabbitMQ队列中读取事件。	Rabbitmq input plugin
	logstash-input-redis	从Redis实例中读取事件。	Redis input plugin
	logstash-input-s3	从S3 Bucket中的文件流式读取事件。	S3 input plugin
	logstash-input-snmp	使用简单网络管理协议（SNMP）轮询网络设备，获取当前设备操作状态信息。	SNMP input plugin
	logstash-input-snmptrap	将SNMP trap消息作为事件读取。	Snmptrap input plugin

类别	名称	说明	介绍
	logstash-input-sqs	从Amazon Web Services简单队列服务（Simple Queue Service,SQS）队列中读取事件。	Sqs input plugin
	logstash-input-stdin	从标准输入读取事件。	Stdin input plugin
	logstash-input-syslog	在网络上将syslog消息作为事件读取。	Syslog input plugin
	logstash-input-tcp	通过TCP套接字读取事件。	Tcp input plugin
	logstash-input-twitter	从Twitter Streaming API接收事件。	Twitter input plugin
	logstash-input-udp	在网络上通过UDP，将消息作为事件读取。	Udp input plugin
	logstash-input-unix	通过UNIX套接字读取事件。	Unix input plugin
	logstash-output-elasticsearch	从Elasticsearch读取数据。	Elasticsearch output plugin
	logstash-output-kafka	向Kafka主题写入事件。	Kafka output plugin
	logstash-output-lumberjack	使用lumberjack协议发送事件。	Lumberjack output plugin
	logstash-output-nagios	通过Nagios命令文件，向Nagios发送被动检查结果。	Nagios output plugin
	logstash-output-pagerduty	根据预先配置的服务和升级政策发送通知。	Pagerduty output plugin
	logstash-output-pipe	将事件输送到另一个程序的标准输入。	Pipe output plugin
	logstash-output-rabbitmq	将事件推送到RabbitMQ exchange。	Rabbitmq output plugin
	logstash-output-redis	使用RPUSH命令将事件发送到Redis队列。	Redis output plugin
	logstash-output-s3	向亚马逊简单存储服务（Amazon Simple Storage Service, Amazon S3）批量上传Logstash事件。	S3 output plugin
	logstash-output-sns	向采用托管pub/sub框架的亚马逊简单通知服务（Amazon Simple Notification Service）发送事件。	Sns output plugin

类别	名称	说明	介绍
output			
	logstash-output-sqs	将事件推送到Amazon Web Services（AWS）SQS队列。	Sqs output plugin
	logstash-output-stdout	将事件打印到运行Shell命令的Logstash标准输出。	Stdout output plugin
	logstash-output-tcp	通过TCP套接字写入事件。	Tcp output plugin
	logstash-output-udp	通过UDP发送事件。	Udp output plugin
	logstash-output-webhdfs	通过webhdfs REST API向HDFS中的文件发送Logstash事件。	Webhdfs output plugin
	logstash-output-cloudwatch	聚合并发送指标数据到AWS CloudWatch。	Cloudwatch output plugin
	logstash-output-csv	以逗号分隔（CSV）或其他分隔的格式，将事件写入磁盘。基于文件输出共享配置值。内部使用Ruby CSV库。	Csv output plugin
	logstash-output-elastic_app_search	向Elastic App Search解决方案发送事件。	App Search output plugin
	logstash-output-email	收到输出后发送电子邮件。 您也可以使用条件来包含，或者排除电子邮件输出执行。	Email output plugin
	logstash-output-file	向磁盘文件写入事件。您可以将事件中的字段作为文件名和/或路径的一部分。	File output plugin
	logstash-output-graphite	从日志中读取指标数据，并将它们发送到Graphite工具。Graphite是一个用于存储和绘制指标的开源工具。	Graphite output plugin
	logstash-output-http	向通用HTTP或HTTPS端点发送事件。	Http output plugin
	logstash-filter-aggregate	聚合单个任务下多个事件（通常为日志记录）的信息，并将聚合信息推送到最后的任务事件。	Aggregate filter plugin
	logstash-filter-anonymize	将字段值替换为一致性哈希值，以实现字段匿名化。	Anonymize filter plugin
	logstash-filter-cidr	根据网络块列表检查事件中的IP地址。	Cidr filter plugin

类别	名称	说明	介绍
filter	logstash-filter-prune	基于黑名单或白名单的字段列表来精简事件数据。	Prune filter plugin
	logstash-filter-clone	检查重复事件。将为克隆列表中的每个类型创建克隆。	Clone filter plugin
	logstash-filter-csv	解析包含CSV数据的事件字段，并将其作为单独字段存储（名字也可指定）。 该过滤器还可以解析带有任何分隔符（不只是逗号）的数据。	Csv filter plugin
	logstash-filter-date	解析字段中的日期，然后使用该日期或时间戳作为事件的logstash时间戳。	Date filter plugin
	logstash-filter-de_dot	将点（.）字符替换为其他分隔符，以重命名字段。 在实际应用中，该过滤器的代价较大。它必须将源字段内容拷贝到新目的字段，该新字段的名字中不再包含点（.），然后移除相应源字段。	De_dot filter plugin
	logstash-filter-dissect	Dissect过滤器是一种拆分操作。	Dissect filter plugin
	logstash-filter-dns	对reverse阵列下的各个或指定记录执行DNS查找（A记录或CNAME记录查找，或PTR记录的反向查找）。	Dns filter plugin
	logstash-filter-drop	删除满足此过滤器的所有事件。	Drop filter plugin
	logstash-filter-elasticsearch	搜索Elasticsearch中的过往日志事件，并将其部分字段复制到当前事件。	Elasticsearch filter plugin
	logstash-filter-fingerprint	创建一个或多个字段的一致性哈希值（指纹），并将结果存储在新字段。	Fingerprint filter plugin
	logstash-filter-geoip	根据Maxmind GeoLite2数据库的数据添加关于IP地址的地理位置信息。	Geoip filter plugin
	logstash-filter-grok	解析任意非结构化文本并将其结构化。	Grok filter plugin
	logstash-filter-http	整合外部网络服务或多个REST API。	HTTP filter plugin
	logstash-filter-jdbc_static	使用预先从远程数据库加载的数据丰富事件。	Jdbc_static filter plugin
	logstash-filter-jdbc_streaming	执行SQL查询，并将结果集存储到目标字段。 将结果缓存到具有有效期的本地最近最少使用（LRU）缓存。	Jdbc_streaming filter plugin

类别	名称	说明	介绍
	logstash-filter-json	JSON解析过滤器，将包含JSON的已有字段扩展为Logstash事件中的实际数据结构。	JSON filter plugin
	logstash-filter-kv	自动解析各种foo=bar消息（或特定事件字段）。	Kv filter plugin
	logstash-filter-memcached	将外部数据整合到Memcached。	Memcached filter plugin
	logstash-filter-metrics	聚合指标。	Metrics filter plugin
	logstash-filter-mutate	在字段上执行转变。您可以重命名、删除、更换并修改您事件中的字段。	Mutate filter plugin
	logstash-filter-ruby	执行Ruby代码。 该过滤器接受内联Ruby代码或文件。 这两个方案互斥，在工作方式方面略有不同。	Ruby filter plugin
	logstash-filter-sleep	按照指定的休眠时间长度休眠。 在休眠时间内，Logstash将停止。这有助于限流。	Sleep filter plugin
	logstash-filter-split	通过分解事件的一个字段，并将产生的每个值嵌入原始事件的克隆版，从而克隆事件。被分解的字段可以是字符串或字符串数组。	Split filter plugin
	logstash-filter-syslog_pri	解析Syslog（RFC3164）消息前部的PRI字段。 如果没有设置优先级，它会默认为13（每个请求注解）。	Syslog_pri filter plugin
	logstash-filter-throttle	限制事件的数量。	Throttle filter plugin
	logstash-filter-translate	一款普通搜索和替换工具，基于配置的哈希和/或文件决定替换值。	Translate filter plugin
	logstash-filter-truncate	截断超过一定长度的字段。	Truncate filter plugin
	logstash-filter-urldecode	解码URL编码字段。	Urldecode filter plugin
	logstash-filter-useragent	基于BrowserScope数据，将用户代理字符串解析为结构化数据。	Useragent filter plugin
	logstash-filter-xml	XML过滤器。将包含XML的字段，扩展为实际数据结构。	Xml filter plugin
	logstash-codec-cef	根据《实施 ArcSight 通用事件格式》第二十次修订版（2013 年 6 月 5 日），使用Logstash编解码器处理ArcSight通用事件格式（CEF）。	Cef codec plugin

类别	名称	说明	介绍
codec	logstash-codec-collectd	在网络上通过UDP从collectd二进制协议中读取事件。	Collectd codec plugin
	logstash-codec-dots	该编解码器生成一个点（.），代表它处理的一个事件。	Dots codec plugin
	logstash-codec-edn	读取并产生EDN格式数据。	Edn codec plugin
	logstash-codec-edn_lines	读取并产生以新行分隔的EDN格式数据。	Edn_lines codec plugin
	logstash-codec-es_bulk	将Elasticsearch bulk格式解码到单独的事件中，并将元数据解码到[@metadata](/metadata)字段。	Es_bulk codec plugin
	logstash-codec-fluent	处理fluentd msgpack模式。	Fluent codec plugin
	logstash-codec-graphite	编码并解码Graphite格式的行。	Graphite codec plugin
	logstash-codec-json	解码（通过输入）和编码（通过输出）完整的JSON消息。	Json codec plugin
	logstash-codec-json_lines	解码以新行分隔的JSON流。	Json_lines codec plugin
	logstash-codec-line	读取面向行的文本数据。	Line codec plugin
	logstash-codec-msgpack	读取并产生MessagePack编码内容。	Msgpack codec plugin
	logstash-codec-multiline	将多行消息合并到单个事件中。	Multiline codec plugin
	logstash-codec-netflow	解码Netflow v5、v9和v10（IPFIX）数据。	Netflow codec plugin
	logstash-codec-plain	处理事件之间没有分隔的明文。	Plain codec plugin
	logstash-codec-rubydebug	使用Ruby Awesome Print库输出Logstash事件数据。	Rubydebug codec plugin

6.2. 安装或卸载插件

安装或卸载Logstash插件

当您购买了阿里云Logstash实例后，系统会在默认插件列表中显示预置的插件，您可以根据需求安装或卸载这些插件。本文介绍如何安装或卸载系统默认插件。

前提条件

已创建阿里云Logstash实例。具体操作，请参见步骤一：创建阿里云Logstash实例。

注意事项

- 阿里云Logstash不支持logstash-input-file 插件，如果您需要将本地文件数据采集到阿里云Logstash中，可以使用logstash-input-beats插件，将Filebeat作为Logstash input，并使用8000～9000之间的端口。
- 安装或卸载插件都会触发集群重启，并且卸载插件时会删除当前选中的插件，因此建议在业务低峰期操作。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**插件配置**。
4. 在**系统默认插件列表**中，单击目标插件右侧的**安装**或**卸载**。
5. 在弹出的对话框中，阅读注意事项，确认无误后单击**确认**。
   确认后，实例会重启。重启时，可在任务列表中查看任务进度，具体操作请参见查看实例任务进度详情。重启成功后，即可完成插件的安装或卸载。

相关文档

- 获取所有或指定插件详细信息的API文档：ListLogstashPlugins
- 安装插件的API文档：InstallLogstashSystemPlugin
- 卸载插件的API文档：UninstallLogstashPlugin

6.3. logstash-input-sls插件使用说明

logstash-input-sls插件

logstash-input-sls插件是阿里云Logstash自带的默认插件。作为Logstash的input插件，logstash-input-sls插件提供了从日志服务获取日志的功能。

> **说明** logstash-input-sls是阿里云维护的开源插件，详情请参见logstash-input-logservice。

功能特性

- 支持分布式协同消费：可配置多台服务器同时消费一个Logstore服务。

  > **说明** 多台Logstash服务器进行分布式协同消费时，由于logstash-input-sls插件限制，需保证各个服务器仅部署一个input sls管道。如果单个服务器中存在多个input sls管道，输出端可能会出现数据重复的异常情况。

- 高性能：基于Java ConsumerGroup实现，单核消费速度可达20 MB/s。
- 高可靠：消费进度会被保存到服务端，宕机恢复时，会从上一次checkpoint处自动恢复。
- 自动负载均衡：根据消费者数量自动分配shard，消费者增加或退出后会自动进行负载均衡。

前提条件

您已完成以下操作：

- 安装logstash-input-sls插件。

  具体操作步骤请参见安装或卸载插件。

- 创建日志服务项目和Logstore，并采集数据。

  具体操作步骤请参见日志服务快速入门教程。

使用logstash-input-sls插件

满足以上前提条件后，您可以通过配置文件管理管道的方式创建管道任务。在创建管道任务时，按照以下说明配置管道参数。配置完成后进行保存与部署，即可触发Logstash从日志服务获取日志。

以使用阿里云Logstash消费某一个Logstore，并将日志输出到阿里云Elasticsearch为例，配置示例如下。

```
input {
 logservice{
  endpoint => "your project endpoint"
  access_id => "your access id"
  access_key => "your access key"
  project => "your project name"
  logstore => "your logstore name"
  consumer_group => "consumer group name"
  consumer_name => "consumer name"
  position => "end"
  checkpoint_second => 30
  include_meta => true
  consumer_name_with_ip => true
  }
}
output {
  elasticsearch {
    hosts => ["http://es-cn-***.elasticsearch.aliyuncs.com:9200"]
    index => "<your_index>"
    user => "elastic"
    password => "changeme"
  }
}
```

假设某Logstore有10个shard，每个shard的数据流量1 MB/s；每台阿里云Logstash机器处理的能力为3 MB/s，可分配5台阿里云Logstash服务器，每个服务器创建一个input sls管道；并且每个服务器管道设置相同的consumer_group和consumer_name，将consumer_name_with_ip字段设置为true。

这种情况每台服务器会分配到2个shard，分别处理2 MB/s的数据。

参数说明

logstash-input-sls支持的参数如下。

参数名	参数类型	是否必填	说明

参数名	参数类型	是否必填	说明
endpoint	String	是	VPC网络下的日志服务项目的Endpoint，详情请参见经典网络及VPC网络服务入口。
access_id	String	是	阿里云Access Key ID，需要具备ConsumerGroup相关权限，详情请参见使用消费组消费。
access_key	String	是	阿里云Access Key Secret，需要具备ConsumerGroup相关权限，详情请参见使用消费组消费。
project	String	是	日志服务项目名。
logstore	String	是	日志服务日志库名。
consumer_group	String	是	自定义消费组名。
consumer_name	String	是	自定义消费者名。同一个消费组内消费者名不能重复，否则会出现未定义行为。
position	String	是	消费位置，可选： • begin：从日志库写入的第一条数据开始消费。 • end：从当前时间点开始消费。 • yyyy-MM-dd HH:mm:ss：从指定时间点开始消费。
checkpoint_second	Number	否	每隔几秒checkpoint一次，建议10~60秒，不能低于10秒，默认30秒。
include_meta	Boolean	否	传入日志是否包含Meta，Meta包括日志source、time、tag以及topic，默认为true。
consumer_name_with_ip	Boolean	否	消费者名是否包含IP地址，默认为true。分布式协同消费下必须设置为true。

性能基准测试信息

- 测试环境
  - 处理器：Intel(R) Xeon(R) Platinum 8163 CPU @ 2.50GHz，4 Core
  - 内存：8 GB
  - 环境：Linux
- 阿里云Logstash配置

```
input {
  logservice{
  endpoint => "cn-hangzhou-intranet.log.aliyuncs.com"
  access_id => "***"
  access_key => "***"
  project => "test-project"
  logstore => "logstore1"
  consumer_group => "consumer_group1"
  consumer_name => "consumer1"
  position => "end"
  checkpoint_second => 30
  include_meta => true
  consumer_name_with_ip => true
  }
}
output {
  elasticsearch {
    hosts => ["http://es-cn-***.elasticsearch.aliyuncs.com:9200"]
    index => "myindex"
    user => "elastic"
    password => "changeme"
  }
}
```

- 测试过程
  i. 使用Java Producer向Logstore发送数据，每秒分别发送2 MB、4 MB、8 MB、16 MB、32 MB数据。

     每条日志约500字节，包括10个Key和Value对。

  ii. 启动阿里云Logstash消费Logstore中的数据，并确保消费延迟没有上涨（消费速度能够跟上生产的速度）。
- 测试结果

流量（MB/S）	CPU使用率（%）	内存占用量（GB）
32	170.3	1.3
16	83.3	1.3
8	41.5	1.3
4	21.0	1.3
2	11.3	1.3

6.4. logstash-input-oss插件使用说明

logstash-input-oss插件

logstash-input-oss插件基于阿里云消息服务MNS（Message Notification Service），实现了当关联的对象存储服务OSS（Object Storage Service）文件变化时，触发MNS通知阿里云Logstash从OSS文件系统中获取最新的数据。您可以在OSS的事件通知区域，配置当文件发生变化时，自动发送消息给MNS。

> **说明** logstash-input-oss是阿里云维护的开源插件，详情请参见logstash-input-oss。

注意事项

- 当logstash-input-oss插件接收到MNS通知消息后，阿里云Logstash会全量同步关联的文件。
- 如果OSS存储的是.gz或.gzip结尾的文本文件，阿里云Logstash会以.gzip的文件格式对其进行处理，其他格式的文件以文本文件进行处理。
- 文件是以文本文件的方式读取的，如果您的文件是不可解析的格式（例如.jar、.bin等格式），有可能读取出来是乱码。

前提条件

您已完成以下操作：

- 安装logstash-input-oss插件。

  具体操作，请参见安装或卸载插件。
- 开通阿里云OSS服务和阿里云MNS服务，且两者在相同地域。

  具体操作，请参见开通阿里云OSS服务和开通消息服务MNS并授权。
- 在OSS中配置事件通知。

  具体操作，请参见设置事件通知规则。

使用logstash-input-oss插件

参见通过配置文件管理管道创建管道任务，在创建管道任务时，需要按照以下说明配置管道参数。配置完成后进行保存与部署，即可触发阿里云Logstash从OSS中获取数据。

以从OSS中获取数据，然后写入到阿里云Elasticsearch为例，配置示例如下。

```
input {
  oss {
    endpoint => "oss-cn-hangzhou-internal.aliyuncs.com"
    bucket => "zl-ossou****"
    access_key_id => "******"
    access_key_secret => "*********"
    prefix => "file-sample-prefix"
    mns_settings => {
       endpoint => "******.mns.cn-hangzhou-internal.aliyuncs.com"
       queue => "aliyun-es-sample-mns"
    }
    codec => json {
      charset => "UTF-8"
    }
  }
}
output {
  elasticsearch {
    hosts => ["http://es-cn-***.elasticsearch.aliyuncs.com:9200"]
    index => "aliyun-es-sample"
    user => "elastic"
    password => "changeme"
  }
}
```

注意 MNS Endpoint不能以HTTP为前缀，并且需要internal域名，否则会报错。

参数说明

logstash-input-oss插件支持的参数如下。

参数	类型	是否必选	说明
endpoint	string	是	OSS对外服务的访问域名，获取方式请参见访问域名和数据中心。
bucket	string	是	OSS的Bucket名称。
access_key_id	string	是	阿里云账号的AccessKey ID。
access_key_secret	string	是	阿里云账号的Access Key Secret。
prefix	string	否	如果指定了该参数，则Bucket中目录或文件名的前缀必须与之匹配（不是正则表达式）。通过配置该参数，您可以读取指定Bucket下的某一个或者几个目录。

参数	类型	是否必选	说明
additional_oss_settings	hash	否	附加的OSS客户端配置。可选值 • *secure_connection_enabled*：是否启用安全连接。 • *max_connections_to_oss*：OSS的最大连接数。
delete	boolean	否	是否从原始Bucket中删除已处理的文件： • *true*：是 • *false*（默认）：否
backup_to_bucket	string	否	用来备份已处理过的文件的OSS Bucket名称。
backup_to_dir	string	否	用来备份已经处理过的文件的本地目录路径。
backup_add_prefix	string	否	文件处理后，为key（OSS中包含文件名的完整路径）附加一个前缀。当您将数据备份到另一个（或同一个）Bucket时，这个参数将有效地让您选择一个新的文件夹来放置文件。
include_object_properties	boolean	否	是否在[@metadata][oss]中包含OSS对象的属性（last_modified，content_type，metadata）： • *true*：是 • *false*：否 如果不设置此参数，[@metadata][oss][key]将始终存在。
exclude_pattern	string	否	要从Bucket中排除的key的ruby正则表达式。
mns_settings	hash	是	消息服务（MNS）配置，可选值： • *endpoint*：MNS端口链接。不能以HTTP为前缀，并且需要internal域名，否则会报错。 • *queue*：队列名。 • *poll_interval_seconds*：当队列中没有消息时，针对该队列的ReceiveMessage请求最长的等待时间，默认为10秒。 • *wait_seconds*：本次ReceiveMessage请求最长的Polling等待时间，单位为秒。 ReceiveMessage的详细信息请参见ReceiveMessage。

常见问题

- Q：为什么基于MNS设计logstash-input-oss插件？

  A：因为OSS文件的变更需要有一种机制通知客户端，而目前OSS文件事件变更可以无缝的写入到MNS中。

- Q：为什么不使用OSS的ListObjects API获取变更的文件？

  A：OSS在记录未处理的文件及已经处理的文件时会增加本地存储，当本地存储较大时，ListObjects API性能会降低。目前其他文件存储系统，如S3开源社区，也将ListObjects API改为了消息通知机制。

相关文档

logstash-input-oss插件使用的最佳实践文档：基于Logstash迁移OSS数据

6.5. logstash-output-oss插件使用说明

logstash-output-oss插件

通过logstash-output-oss插件，您可以将数据批量传送到阿里云对象存储服务OSS（Object Storage Service）中。本文介绍如何使用logstash-output-oss插件。

> **说明** logstash-output-oss是阿里云维护的开源插件，详细信息，请参见logstash-output-oss。

前提条件

您已完成以下操作：

- 安装logstash-output-oss插件。

  具体操作，请参见安装logstash-output-oss插件。

- 开通阿里云OSS服务。

  具体操作，请参见开通阿里云OSS服务。

- 创建可读写的OSS Bucket，并且获取拥有该Bucket写权限的Accesskey ID和Accesskey Secret。

  具体操作，请参见创建可读写的OSS Bucket。

- 准备输入数据源。

  输入数据源可以为input支持的所有输入源插件中的数据，详细信息，请参见input插件。

使用logstash-output-oss插件

满足以上前提条件后，您可以通过配置文件管理管道的方式创建管道任务。在创建管道任务时，按照以下说明配置Pipeline参数，保存并部署后，即可触发阿里云Logstash向OSS传送数据。

以将Beats采集文件中的数据传送到OSS为例。

```
input {
  beats {
    port => "8044"
    codec => json {
      charset => "UTF-8"
  }
}
output {
   oss {
    endpoint => "http://oss-cn-hangzhou-internal.aliyuncs.com"
    bucket => "zl-log-output-test"
    access_key_id => "LTAIaxxxxx******"
    access_key_secret => "zuxxxx8hBpXs3e6i******"
    prefix => "oss/database"
    recover => true
    rotation_strategy => "size_and_time"
    time_rotate => 1
    size_rotate => 1000
    temporary_directory => "/ssd/1/<Logstash实例ID>/logstash/data/22"

    encoding => "gzip"
    additional_oss_settings => {
      max_connections_to_oss => 1024
      secure_connection_enabled => false
    }
    codec => json {
      charset => "UTF-8"
    }
  }
}
```

说明

- 阿里云Logstash目前只支持在同一专有网络下进行数据传输，如果源端数据在公网下，请参见配置NAT公网数据传输，在公网环境下进行数据传输。
- logstash-output-oss插件的具体应用，请参见基于Logstash迁移OSS数据。

参数说明

logstash-output-oss插件支持的参数如下。

参数	类型	是否必选	说明
endpoint	string	是	OSS对外服务的访问域名。详细信息，请参见访问域名和数据中心。
bucket	string	是	OSS的Bucket名称。
access_key_id	string	是	拥有对应Bucket写权限的Accesskey ID。

参数	类型	是否必选	说明
access_key_secret	string	是	拥有对应Bucket写权限的Accesskey Secret。
prefix	string	否	指定文件名前缀，不指定默认为空。 警告 此选项支持字符串，因此可能会创建很多临时本地文件。
recover	Boolean	否	程序出现异常退出时，保存在本地的数据是否可以继续上传。默认为true。
additional_oss_settings	hash	否	附加的OSS客户端配置。可选值： • server_side_encryption_algorithm：服务端加密方式，只支持AES256。 • secure_connection_enabled：是否开启https，默认false。 • max_connections_to_oss：最大连接数，默认1024。
temporary_directory	string	是	数据上传到OSS之前的临时目录路径定义，必须设置为/ssd/1/<Logstash实例ID>/logstash/data/。任务结束后，一般会在秒级被删除。
rotation_strategy	string	否	文件滚动更新策略。可选值：size、time、size_and_time（默认）。
size_rotate	number	否	如果文件大小大于等于size_rotate，OSS将滚动更新文件（依赖rotation_strategy）。默认为31457280 Bytes。
time_rotate	number	否	如果文件的生存时长大于等于time_rotate，OSS将滚动更新文件（依赖rotation_strategy）。默认为15分钟。
upload_workers_count	number	否	上传线程并发数。
upload_queue_size	number	否	上载队列大小。
encoding	string	否	消息在上传文件到OSS之前，支持纯压缩和gzip压缩。可选值：gzip、none（默认）。

临时文件说明

logstash-output-oss在传送数据到OSS时，会在Logstash本地创建一个临时文件。数据临时存储在该文件下，logstash-output-oss插件定期推送数据到OSS。可通过设置temporary_directory参数，设置该临时文件的地址。如果您对输出数据保存的路径有要求，可以设置该临时文件路径。

临时文件路径示例如下。

```
/ssd/1/<Logstash实例ID>/logstash/data/eaced620-e972-0136-2a14-02b7449b****/logstash/1/ls.oss
.eaced620-e972-0136-2a14-02b7449b****.2018-12-24T14.27.part-0.data
```

路径	说明
/ssd/1/<Logstash实例ID>/logstash/data/	由temporary_directory指定的临时目录。
eaced620-e972-0136-2a14-02b7449b****	随机UUID。
logstash/1	OSS对象前缀。
ls.oss	临时文件，表示由logstash-output-oss插件生成。
2018-12-24T14.27	临时文件创建的时间。
part-0	临时文件的前缀。
.data	临时文件的后缀。如果设置 `encoding` 为 `gzip` ，将会以 `.gz` 结尾，其他以 `.data` 结尾。

6.6. logstash-input-maxcompute插件使用说明

logstash-input-maxcompute插件

通过logstash-input-maxcompute插件，您可以读取MaxCompute离线表的数据到其他数据源中。

前提条件

您已完成以下操作：

- 安装logstash-input-maxcompute插件。

  详情请参见安装或卸载插件。
- 开通阿里云MaxCompute产品，并创建项目、创建表和导入数据。

  详情请参见MaxCompute官方文档的准备工作和快速入门章节。

使用logstash-input-maxcompute插件

满足以上前提条件后，您可以通过配置文件管理管道的方式创建管道任务。在创建管道任务时，按照以下说明配置Pipeline参数，保存并部署后，即可触发阿里云Logstash读取MaxCompute的数据到目标数据源中。

配置脚本如下，相关参数说明请参见参数说明。

```
input {
    maxcompute {
        access_id => "Your accessId"
        access_key => "Your accessKey"
        endpoint => "maxcompute service endpoint"
        project_name => "Your project"
        table_name => "Your table name"
        partition => "pt='p1',dt='d1'"
        thread_num => 1
        dirty_data_file => "/ssd/1/tmp/xxxxx"
    }
}
output {
    stdout {
        codec => rubydebug
    }
}
```

注意

- 目前阿里云Logstash只支持同一专有网络VPC（Virtual Private Cloud）下的数据传输，如果源端数据在公网环境下，请参见配置NAT公网数据传输，通过公网访问Logstash。
- logstash-input-maxcompute插件会全量同步数据到目标数据源中。

参数说明

logstash-input-maxcompute插件支持的参数如下。

参数	类型	是否必选	说明
endpoint	string	是	MaxCompute对外服务的访问域名，详情请参见各地域Endpoint对照表（外网连接方式）。
access_id	string	是	阿里云账号的AccessKey ID。
access_key	string	是	阿里云账号的Access Key Secret。
project_name	string	是	MaxCompute的项目名称。
table_name	string	是	MaxCompute的表名称。
partition	string	是	分区字段。分区表按照字段来定义，例如：`sale_date='201911'`，`region='hangzhou'`。
thread_num	number	是	线程数，默认为1。
retry_interval	number	否	重试的间隔，单位为秒。

参数	类型	是否必选	说明
dirty_data_file	string	是	指定文件，用于记录处理失败的日志。

6.7. logstash-input-datahub插件使用说明

logstash-input-datahub插件

通过logstash-input-datahub插件，您可以读取DataHub中的数据到其他数据源中。本文介绍如何使用logstash-input-datahub插件。

前提条件

您已完成以下操作：

- 安装logstash-input-datahub插件。

  具体操作，请参见安装或卸载插件。
- 开通DataHub产品，并完成创建项目、创建Topic和导入数据。

  具体操作，请参见快速入门。

使用logstash-input-datahub插件

参见通过配置文件管理管道，在创建管道任务时，按照以下说明配置Pipeline参数，保存并部署后，即可触发阿里云Logstash读取DataHub的数据到目标数据库中。

Logstash的Pipeline配置如下，相关参数说明请参见参数说明。

```
input {
    datahub {
        access_id => "Your accessId"
        access_key => "Your accessKey"
        endpoint => "Endpoint"
        project_name => "test_project"
        topic_name => "test_topic"
        interval => 5
        #cursor => {
        #    "0"=>"20000000000000000000000003110091"
        #    "2"=>"20000000000000000000000003110091"
        #    "1"=>"20000000000000000000000003110091"
        #    "4"=>"20000000000000000000000003110091"
        #    "3"=>"20000000000000000000000003110000"
        #}
        shard_ids => []
        pos_file => "/ssd/1/<Logstash实例ID>/logstash/data/文件名"
    }
}
output {
    elasticsearch {
      hosts => ["http://es-cn-mp91cbxsm000c****.elasticsearch.aliyuncs.com:9200"]
      user => "elastic"
      password => "your_password"
      index => "datahubtest"
      document_type => "_doc"
  }
}
```

注意 目前阿里云Logstash只支持同一专有网络下的数据传输，如果源端数据在公网环境下，请参见配置NAT公网数据传输，通过公网访问Logstash。

参数说明

logstash-input-datahub插件支持的参数如下。

参数	类型	是否必选	说明
endpoint	string	是	DataHub对外服务的访问域名，详细信息请参见域名列表。
access_id	string	是	阿里云账号的AccessKey ID。
access_key	string	是	阿里云账号的Access Key Secret。
project_name	string	是	DataHub的项目名称。
topic_name	string	是	DataHub的Topic名称。
retry_times	number	否	重试次数。-1表示无限重试（默认）、0表示不重试、大于0表示按照设置的次数重试。

参数	类型	是否必选	说明
retry_interval	number	否	重试的间隔，单位为秒。
shard_ids	array	否	需要消费的shard列表。默认为空，表示全部消费。
cursor	string	否	消费起点。默认为空，表示从头开始消费。
pos_file	string	是	Checkpoint记录文件，必须配置，优先使用checkpoint恢复消费。
enable_pb	boolean	否	是否使用pb传输，默认为true。如果不支持pb传输，请将该参数设置为false。
compress_method	string	否	网络传输的压缩算法，默认不压缩。可选项：lz4、deflate。
print_debug_info	boolean	否	是否打印DataHub的Debug信息，默认为false。设置为true时，会打印大量信息，这些信息仅用来进行脚本调试。

6.8. logstash-output-datahub插件使用说明

logstash-output-datahub插件

通过logstash-output-datahub插件，您可以将数据传输到DataHub中。本文介绍如何使用logstash-output-datahub插件。

前提条件

您已完成以下操作：

- 安装logstash-output-datahub插件。

  具体操作，请参见安装或卸载插件。
- 开通DataHub产品，并完成创建项目和创建Topic。

  具体操作，请参见快速入门。
- 准备输入数据源。

  输入数据源可以为input支持的所有输入源插件中的数据，本文以阿里云Elasticsearch为例，详细信息请参见input插件。

使用logstash-output-datahub插件

参见通过配置文件管理管道，在创建管道任务时，按照以下说明配置Pipeline参数，保存并部署后，即可触发阿里云Logstash向DataHub传送数据。

Logstash的Pipeline配置如下，相关参数说明请参见参数说明。

```
input {
    elasticsearch {
      hosts => ["http://es-cn-mp91cbxsm000c****.elasticsearch.aliyuncs.com:9200"]
      user => "elastic"
      index => "test"
      password => "your_password"
      docinfo => true
  }
}
filter{
}
output {
    datahub {
        access_id => "Your accessId"
        access_key => "Your accessKey"
        endpoint => "Endpoint"
        project_name => "project"
        topic_name => "topic"
        #shard_id => "0"
        #shard_keys => ["thread_id"]
        dirty_data_continue => true
        dirty_data_file => "/ssd/1/<Logstash实例ID>/logstash/data/文件名"
        dirty_data_file_max_size => 1000
    }
}
```

说明 阿里云Logstash目前只支持在同一专有网络下进行数据传输，如果源端数据在公网下，请参见配置NAT公网数据传输，在公网环境下进行数据传输。

参数说明

logstash-output-datahub插件支持的参数如下。

参数	类型	是否必选	说明
endpoint	string	是	DataHub对外服务的访问域名，详细信息请参见域名列表。
access_id	string	是	阿里云账号的AccessKey ID。
access_key	string	是	阿里云账号的Access Key Secret。
project_name	string	是	DataHub的项目名称。
topic_name	string	是	DataHub的Topic名称。
retry_times	number	否	重试次数。-1表示无限重试（默认）、0表示不重试、大于0表示按照设置的次数重试。
retry_interval	number	否	重试的间隔，单位为秒，默认为5。

参数	类型	是否必选	说明
skip_after_retry	boolean	否	当由DataHub异常导致的重试次数超过retry_times设置的值，是否跳过这一轮上传的数据。默认为false。
approximate_request_bytes	number	否	用于限制每次发送请求的字节数，是一个近似值，防止因Request body过大而被拒绝接收，默认为2048576（2MB）。
shard_keys	array	否	数据的字段名称，插件会根据这些字段的值计算Hash值，将每条数据写入到某个shard。 **注意** shard_keys和shard_ids都未指定，默认轮询写入各shard。
shard_ids	array	否	所有数据写入指定的shard。 **注意** shard_keys和shard_ids都未指定，默认轮询写入各shard。
dirty_data_continue	string	否	处理数据时遇到脏数据是否继续运行，默认为false。设置为true时，必须指定dirty_data_file文件，表示处理数据时忽略脏数据。
dirty_data_file	string	否	脏数据文件名称。当dirty_data_continue为true时，必须指定该参数值。 **注意** 处理数据时，脏数据文件会被分割成两个部分part1和part2，part1为原脏数据，part2为替换后的脏数据。
dirty_data_file_max_size	number	否	脏数据文件大小的最大值。
enable_pb	boolean	否	是否使用pb传输，默认为true。如果不支持pb传输，请将该参数值设置为false。

7.网络与安全

7.1. 配置NAT公网数据传输

logstash与公网连通

阿里云Logstash实例部署在专有网络VPC（Virtual Private Cloud）下，如果您需要通过Logstash采集公网网络中的数据，或者将Logstash采集的数据输出到公网网络中，则需要配置NAT网关，实现专有网络下的阿里云Logstash与公网连通。本文介绍具体的配置方法。

前提条件

您已完成以下操作：

- 创建专有网络和虚拟交换机。

  具体操作，请参见搭建IPv4专有网络。
- 创建阿里云Logstash实例。

  具体操作，请参见创建阿里云Logstash实例。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**网络与安全**。
4. 在**网络配置**区域，单击**前往配置NAT网关**。

   关于NAT网关的详细说明和配置流程，请参见创建和管理公网NAT网关实例。其中DNAT条目适用于公网服务向Logstash节点推送数据；SNAT条目适用于Logstash主动访问公网。
5. 在NAT网关配置页面，创建NAT网关。

   创建NAT网关时，所选的地域和VPC ID需要与阿里云Logstash保持一致。详细创建方法，请参见步骤一：创建公网NAT网关。
6. 绑定弹性公网IP。
   i. 单击NAT网关列表右侧**操作**列下的 > **绑定弹性公网IP**。
   ii. 在**绑定弹性公网IP**页面，选择**从已有弹性公网IP中选择**。

   如果还没有EIP，可选择**新购弹性公网IP并绑定**，按照页面提示完成绑定。
   iii. 选择可用的EIP，单击**确定**。

   > 注意 一个NAT网关最多可绑定20个EIP（最多可绑定10个按流量计费的EIP，每个按流量计费的EIP的最大峰值不能超过200 Mbps），您可以提交工单申请更多配额。
7. 创建DNAT条目。
   i. 在NAT网关列表中，单击对应网关右侧**操作**列下的**设置DNAT**。
   ii. 在**DNAT条目列表**区域，单击**创建DNAT条目**。

iii. 在**创建DNAT条目**页面，填写相关参数。

参数	说明
选择公网IP地址	选择一个可用的公网IP。 **说明** 用于创建SNAT条目的公网IP不能再用来创建DNAT条目。
选择私网IP地址	选择**通过手动输入**，输入Logstash的IP地址。可在Logstash的基本信息页面获取，获取方式请参见查看实例的基本信息。
端口设置	选择DNAT映射的方式： ■ **任意端口**：该方式属于IP映射，相当于为目标Logstash实例配置了一个弹性公网IP。任何访问该公网IP的请求，都将转发到目标Logstash实例上。 ■ **具体端口**：该方式属于端口映射，NAT网关会将指定协议和端口访问该公网IP的请求，转发到目标Logstash实例的指定端口上。 选择**具体端口**后，请根据业务需求输入**公网端口**（进行端口转发的外部端口）、**私网端口**（进行端口转发的内部端口）和**协议类型**（进行端口转发的协议类型）。
条目名称	输入DNAT条目的名称。 名称长度为2~128个字符，以大小写字母或中文开头， 可包含数字、下划线（_）和短横线（-）。

iv. 单击**确定创建**，完成创建。

8. 创建SNAT条目。

i. 返回NAT网关列表页面，单击对应网关右侧**操作**列下的**设置SNAT**。

ii. 在**SNAT条目列表**区域，单击**创建SNAT条目**。

iii. 在**创建SNAT条目**页面，单击**SNAT条目粒度**区域的**交换机粒度**，并填写相关参数。

参数	说明
选择交换机	选择Logstash所属的专有网络中的交换机。该交换机下所有ECS实例，都将通过SNAT功能进行公网访问。
选择公网IP地址	选择用来提供互联网访问的公网IP，支持选择多个公网IP，多个公网IP构建SNAT IP地址池。 当选择多个公网IP地址配置SNAT IP地址池时，请确保每个公网IP地址加入到一个共享带宽中。详细信息，请参见加入与移出共享带宽。

更多参数的详细信息，请参见创建和管理SNAT条目。

iv. 单击**确定创建**，完成创建。

9. 返回Logstash控制台，通过管道配置实现公网数据传输。

详细信息，请参见通过配置文件管理管道。

相关文档

最佳实践：腾讯云Elasticsearch数据迁移至阿里云

8.监控报警与日志查询

8.1. 监控报警

8.1.1. 集群监控概述

阿里云Logstash为您提供了集群监控功能，通过配置对应的监控功能，您可以实时了解集群健康状态。本文主要为您介绍阿里云Logstash集群涉及的监控功能以及使用场景。

功能	使用场景
配置自定义报警策略	为避免出现集群状态异常、节点磁盘使用率过高等问题而影响Logstash服务，建议您进行监控报警配置，实时监控集群状态、节点磁盘使用率等信息，及时查收报警短信，提前做好防御措施。 目前，Logstash支持在云监控中配置以下四种监控项： • Logstash实例节点磁盘使用率 • 节点内存使用量 • Logstash实例节点CPU使用率 • 节点1分钟负载
配置X-Pack监控	如果您需要在Kibana中监控Logstash服务，那么您可以通过开启Logstash的X-Pack监控功能，并关联目标阿里云Elasticsearch实例，进行相关指标的查看。

8.1.2. 配置自定义报警策略

配置logstash自定义报警策略

阿里云Logstash支持对实例进行监控，并支持设置自定义报警规则以及通知方式。为避免出现集群状态异常、节点磁盘使用率过高等问题而影响Logstash服务，建议您进行监控报警配置，实时监控集群状态、节点磁盘使用率等信息，及时查收报警短信，提前做好防御措施。本文介绍如何为Logstash实例配置自定义报警策略。

背景信息

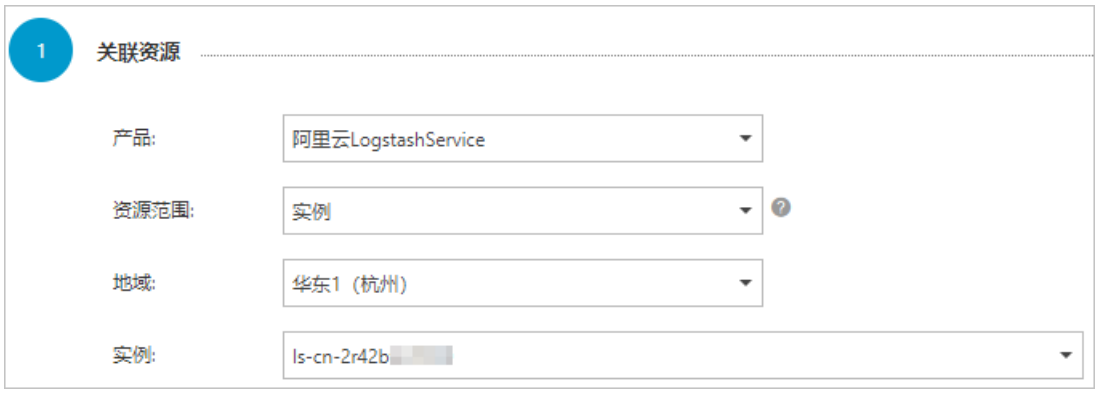
目前Logstash只支持在云监控中配置以下四种监控项。如果您在配置项中观察到其他项，请忽略。

监控项	说明
Logstash实例节点磁盘使用率	必选。报警阈值控制在75%以下。
节点内存使用量	必选。报警阈值控制在85%以下。
Logstash实例节点CPU使用率	可选。报警阈值控制在95%以下。
节点1分钟负载	可选。以CPU核数的80%为参考值。

操作步骤

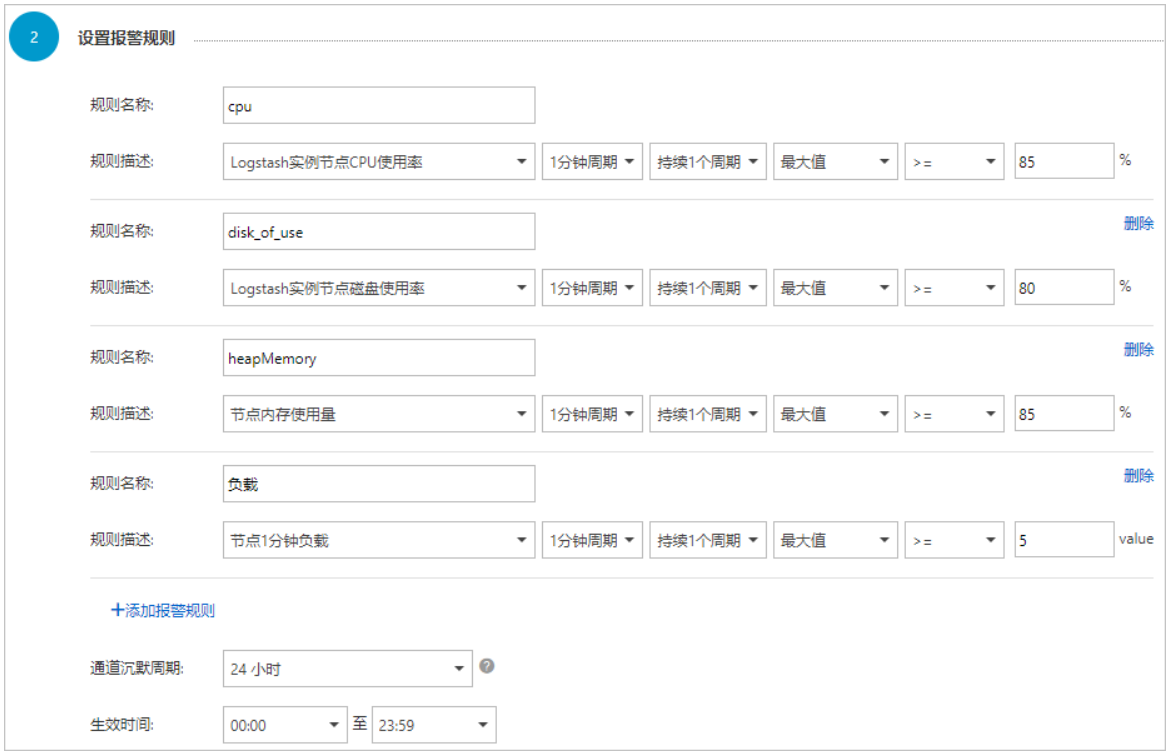
1. 进入云监控控制台。
   i. 登录阿里云Elasticsearch控制台。
   ii. 在顶部菜单栏处，选择地域。
   iii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
   iv. 在左侧导航栏，单击**集群监控**。
   v. 在**监控报警配置**页面，单击**前往云监控配置**，进入云监控控制台的**创建报警规则**页面。

2. 配置关联资源。



参数	说明
产品	选择**阿里云LogstashService**。
资源范围	按需选择，取值含义如下： ○ **全部资源**：选择全部资源，则产品下任何实例满足报警规则描述时，都会发送报警通知。 ○ **实例**：选择指定的实例，则选中的实例满足报警规则描述时，才会发送报警通知。 本文以选择**实例**为例。
地域	选择实例所在地域。
实例	选择待监控的实例。

3. 设置报警规则。



通道沉默时间指报警发生后如果未恢复正常，再次发送一次报警通知的间隔时间。

> **说明** 其他参数说明，请参见创建阈值报警规则。

4. 配置报警**通知方式**，选择**云账号报警联系人**。

   如果您还没有报警联系组，请单击**快速创建联系人组**，进行创建。

> **说明** 您可以在**报警回调**中填写可通过公网访问的URL，云监控会将报警信息通过POST请求推送到该地址，目前仅支持HTTP协议。

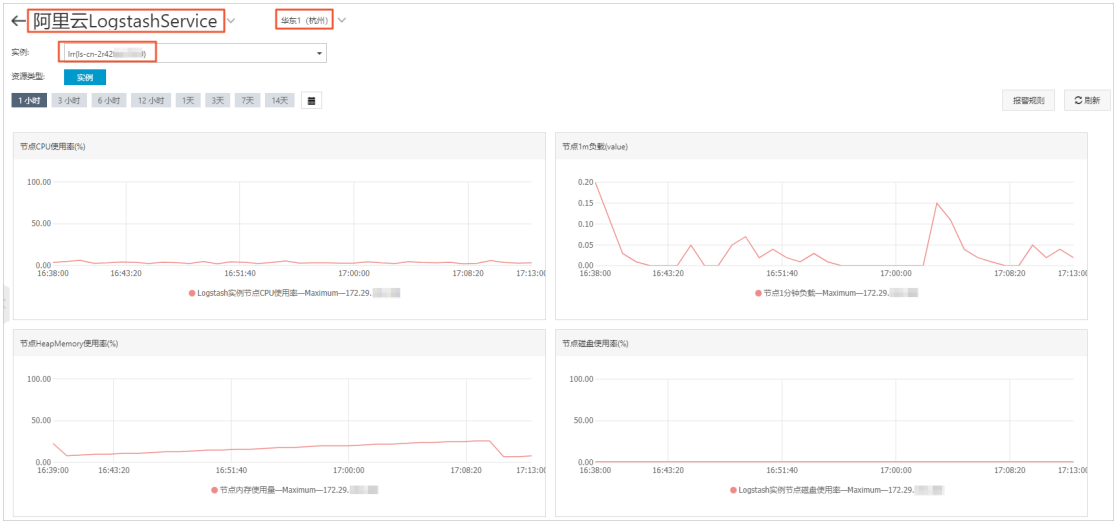
5. 单击**确认**。
6. 查看Logstash监控大屏。

   配置完成后，Logstash实例的监控信息将在实例正常运行后开始采集。当指标值超过您设置的报警阈值时，系统会为您发送报警通知。您可以通过以下方式查看Logstash监控大屏：

   i. 在云监控首页的左侧导航栏，选择**Dashboard > 云产品监控大盘**。

   ii. 选择**阿里云LogstashService**产品，并选择地域。

iii. 选择**实例**和监控时间段，即可查看该段时间内的监控大屏。



8.1.3. 配置X-Pack监控

本文介绍如何通过配置X-Pack来监控阿里云Logstash服务。开启X-Pack监控，并关联阿里云Elasticsearch实例后，即可在Kibana中监控Logstash服务。

前提条件

您已完成以下操作：

- 创建阿里云Logstash实例。详情请参见创建阿里云Logstash实例。
- 创建阿里云Elasticsearch实例。该实例要求满足以下条件：
  - 创建的Elasticsearch实例需要与Logstash实例处于同一专有网络下，且尽量保证大版本相同，如果大版本不相同，请务必保证版本间的兼容性。详情请参见创建阿里云Elasticsearch实例。
  - 创建Elasticsearch实例后，需要开通Kibana公网访问功能。详情请参见配置Kibana公网或私网访问白名单。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**集群监控**。
4. 在**监控报警配置**区域，单击**X-Pack监控**右侧的**修改**。

5. 在**修改配置**面板中，选中**开启**，并配置要关联的阿里云Elasticsearch实例。

参数	说明
Elasticsearch实例	选择需要关联的阿里云Elasticsearch实例，要求与Logstash实例在同一专有网络下，尽量保证大版本相同，如果大版本不相同，请务必保证版本间的兼容性。
用户名	访问阿里云Elasticsearch实例的用户名。
密码	访问阿里云Elasticsearch实例的密码。

6. 单击**测试连通性**。

   无报错即连通成功。

> **警告** 修改X-Pack配置会触发实例重启，请在不影响业务的情况下，继续执行以下步骤。

7. 单击**确定**。

   确定后，系统返回**集群监控**页面，并触发实例重启。

8. 等待重启完成后，查看Logstash监控信息。

   重启完成后，**X-Pack监控**显示为**开启**，且在当前页面显示所关联的阿里云Elasticsearch实例。

> **注意** 重启完成后，才可在Kibana控制台上查看到Logstash监控信息。

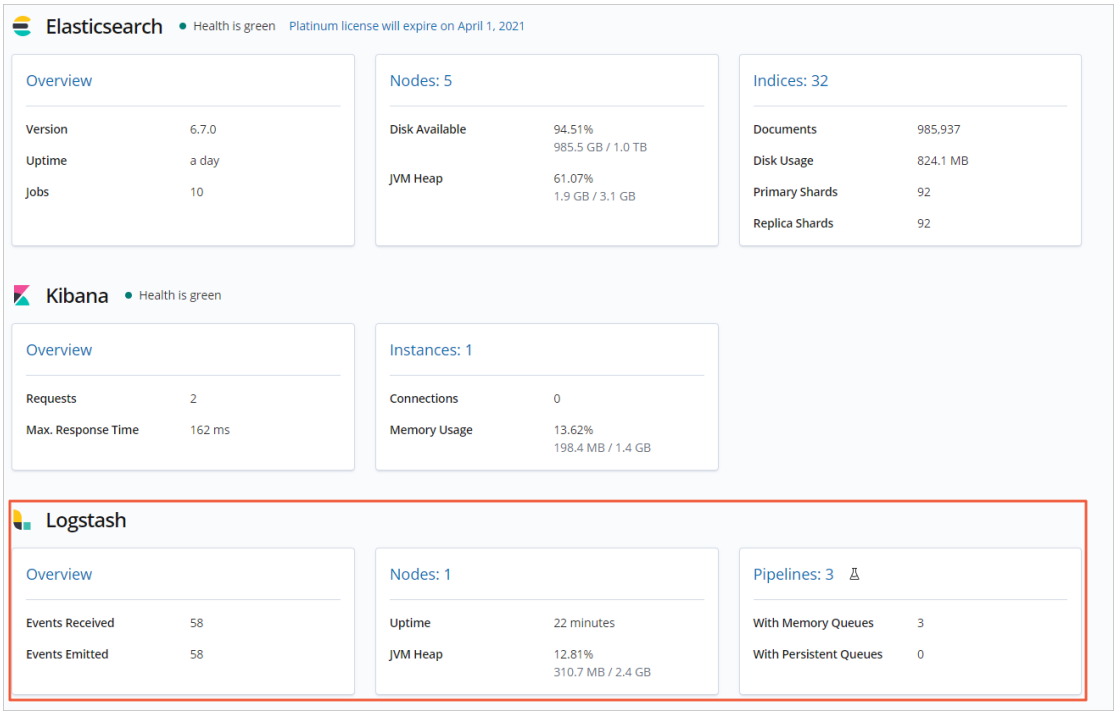
   i. 在**集群监控**页面，单击**前往Kibana控制台管理**。

   ii. 登录Kibana控制台。

   具体操作，请参见登录Kibana控制台。

   iii. 在左侧导航栏，单击**Monitoring**。

iv. 在**Logstash**区域，查看Logstash的监控信息。



8.2. 查询日志

logstash日志

通过阿里云Logstash的日志功能，您可以输入关键字和设置时间范围，锁定需要查询的日志内容，快速定位集群问题，辅助集群运维。本文为您介绍如何查询日志以及常见日志的使用说明。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**日志查询**，查看集群的运行日志。

   阿里云Logstash支持4种类型的日志：主日志、慢日志、GC日志和调试日志。各类日志的说明和使用场景如下，更多详细信息请参见日志说明。

日志类型	说明	使用场景

日志类型	说明	使用场景
主日志	集群的运行状态日志。	当您需要查看集群中各节点的运行状况及管道的运行情况，例如源端和目标端的连通性、创建或修改管道配置情况、管道运行的错误信息时，可查看主日志进行排查。 **注意** 如果您的业务侧出现问题，建议优先查看主日志和集群监控，排除集群自身的性能瓶颈或管道配置问题。
慢日志	耗时比较久的管道事件日志。当管道运行耗时超过指定阈值时，将在慢查询日志中打印相关信息。 **注意** 为了方便后续排查与定位阿里云Logstash问题，YML文件配置中默认开启了慢日志，请不要移除该慢日志配置，详细信息请参见配置YML文件。	当您的业务出现写入慢的问题时，可查看慢日志进行排查。常见原因如下： ○ 管道配置中的源端或目标端压力达到瓶颈：建议扩充源端或目标端资源。 ○ Logstash的管道工作线程数配置太小：增加Logstash的管道批大小和工作线程数，详细信息请参见通过配置文件管理管道。
GC日志	垃圾回收器日志。显示所有JVM堆内存占用触发的垃圾回收情况，通过GC日志可获取详细的垃圾回收信息，包括Old GC、CMS GC、Full GC以及Minor GC等回收机制。	当集群出现性能瓶颈时，可通过GC日志获取详细的GC回收信息，查看是否存在耗时长或操作频繁的GC。
调试日志	管道处理后的输出数据。默认关闭，需要安装logstash-output-file_extend插件，并在output中配置file_extend参数进行开启。	需要在控制台上查看管道配置的输出结果，调试管道配置的场景。

4. 在日志页面的搜索框中，输入查询条件，选择开始时间和结束时间，单击**搜索**。

   阿里云Logstash最多支持查询连续7天内的日志，日志默认按时间倒序展示。支持基于Lucene的日志查询语法，详情请参见Query string syntax。

   本文以查询content包含关键字running，level为info，host为172.16.xx.xx的Logstash主日志为例，示例查询条件为：`host:172.16.xx.xx AND level:info AND content:running`。

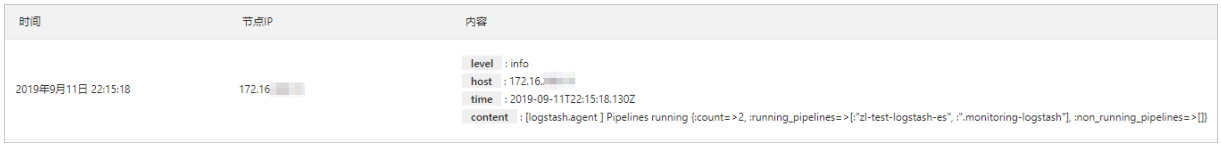
   注意
   - 查询条件中的`AND`必须为大写。
   - 如果结束时间为空，那么结束时间默认为当前时间。如果开始时间为空，那么开始时间默认为结束时间减去1小时。

   搜索成功后，阿里云Logstash会根据您的查询条件返回日志查询结果，并展示在日志查询页面。

日志说明

主日志

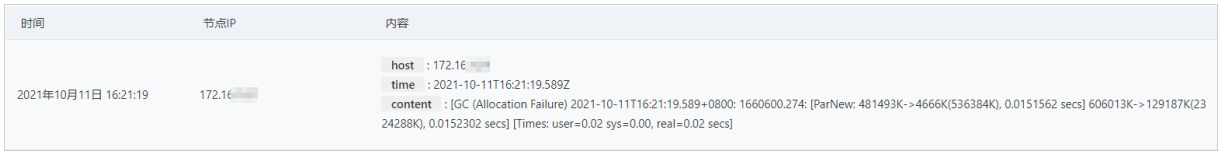
主日志主要展示集群的运行日志，包括日志产生的时间、日志所在的节点IP和日志的详细信息。



参数	说明
时间	日志产生时间。
节点IP	生成日志的节点的IP地址。
内容	日志的详细信息，主要由level、host、time和content组成： • level：日志级别。包括trace、debug、info、warn、error等。 **说明** GC日志没有level。 • host：生成日志的节点的IP地址。 • time：日志产生的时间。 • content：日志的主要内容。

GC日志

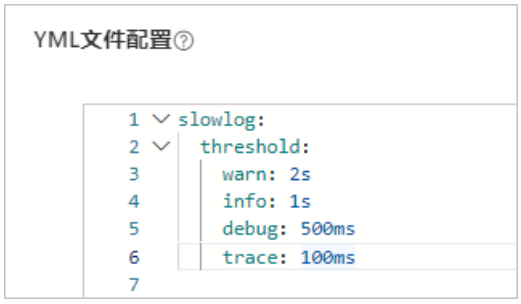
GC日志默认开启，包含日志产生的时间、所在节点的IP地址和日志内容。详细信息，请参见主日志。



慢日志

慢日志默认开启，您可在Logstash实例的YML配置中查看或修改默认的慢日志配置，详细信息请参见配置YML文件。

注意 为方便后续排查与定位阿里云Logstash问题，请不要移除该慢日志配置。



调试日志

当Logstash管道配置出现错误时，可能导致输出数据结果不符合预期，需要反复去目标端确认数据格式，再返回控制台修改配置，这样会耗费大量的时间和人力。对于这种情况，您可以通过阿里云Logstash提供的管道配置调试功能，在管道配置完成后，直接在控制台上通过调试日志查看管道配置的输出结果，降低调试成本。详细信息，请参见使用Logstash管道配置调试功能。

调试日志默认关闭，您需要通过以下方式开启：

1. 安装logstash-output-file_extend插件，具体操作请参见安装或卸载插件。
2. 在管道配置的output中配置file_extend参数，具体操作请参见通过配置文件管理管道。

开启调试日志后，您可以在调试日志页签中，获取管道处理后的输出数据。

相关文档

ListLogstashLog

9.管道任务管理

9.1. 通过配置文件管理管道

logstash管道配置

Logstash通过管道完成数据的采集与处理，管道配置中包含input、output和filter（可选）插件，input和output用来配置输入和输出数据源、filter用来对数据进行过滤或预处理。阿里云Logstash支持多管道并行运行，目前最多支持20个。本文介绍如何通过配置文件管理管道，包括创建管道、修改管道、复制管道和删除管道。

前提条件

您已完成以下操作：

- 创建阿里云Elasticsearch实例。

  具体操作，请参见创建阿里云Elasticsearch实例。

- 开启目标阿里云Elasticsearch实例的自动创建索引功能（本文以此为例），或提前在实例中创建索引和Mapping。

  开启自动创建索引功能的具体操作，请参见配置YML参数。创建索引和Mapping的具体操作，请参见步骤三：创建索引。

  > **说明** 阿里云Elasticsearch为了保证用户操作数据的安全性，默认将**自动创建索引**配置设置为**不允许**。阿里云Logstash在传输数据的时候，使用提交数据的方式创建索引，而不是Create index API的方式。所以在使用阿里云Logstash上传数据之前，需要先把集群的**自动创建索引**设置为**允许**，或提前创建好索引和Mapping。

- 创建阿里云Logstash实例。

  具体操作，请参见创建阿里云Logstash实例。

使用限制

- 阿里云Logstash最多支持20个管道并行运行。
- 如果output指定的数据源为阿里云Elasticsearch，需要提前开启自动创建索引，或创建目标索引和Mapping。
- 配置过程中涉及到阿里云系列产品时，需要在同一专有网络下，否则需要配置网络与安全。详细信息，请参见配置NAT公网数据传输。
- 如果在output中使用了file_extend参数，需要先安装logstash-output-file_extend插件。具体操作，请参见安装或卸载插件。

创建管道

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**管道管理**。
4. 单击**创建管道**。

5. 输入**管道ID**和**Config配置**。

   配置示例如下。

```
input {
    beats {
        port => 8000
        host => "118.11.xx.xx"
    }
}
filter {
}
output {
    elasticsearch {
        hosts => ["http://es-cn-o40xxxxxxxxx****.elasticsearch.aliyuncs.com:9200"]
        index => "logstash_test_1"
        password => "es_password"
        user => "elastic"
    }
    file_extend {
        path => "/ssd/1/ls-cn-v0h1kzca****/logstash/logs/debug/test"
    }
}
```

参数	说明
input	指定输入数据源。支持的数据源类型，请参见Input plugins。 **说明** ○ 当Input插件需要监听Logstash进程所在节点的端口时，请使用8000~9000范围内的端口。 ○ 如果您需要在input中定义插件、驱动或其他文件，可单击**查看扩展文件路径**，在**扩展文件管理**对话框中，单击**前往上传**，根据提示上传对应的文件。详细信息，请参见配置扩展文件。
filter	指定对输入数据进行过滤的插件。支持的插件类型，请参见Filter plugins。
output	指定目标数据源类型。支持的数据源类型，请参见Output plugins。 file_extend：可选，用来开启调试日志功能，并通过path参数配置调试日志的输出路径。建议您配置该参数，配置后，可直接在控制台上查看输出结果。如果未配置，需要去目标端确认输出结果，再返回控制台修改，这样会耗费大量的时间和人力。详细信息，请参见使用Logstash管道配置调试功能。 **注意** 使用file_extend参数前，需要先安装logstash-output-file_extend插件。具体操作，请参见安装或卸载插件。其中的path参数默认为系统指定路径，请勿修改。您也可以单击**开启配置调试**获取path路径。

Config配置的结构及支持的数据类型的详细信息（不同版本支持的数据类型可能不同），请参见Structure of a Config File。

> **注意**
>
> - 为了提升安全性，在使用JDBC驱动并配置管道时，需要在jdbc_connection_string参数后面添加 `allowLoadLocalInfile=false&autoDeserialize=false` ，否则当您在添加Logstash配置文件的时候，调度系统会抛出校验失败的提示，例如 `jdbc_connection_string => "jdbc:mysql://xxx.drds.aliyuncs.com:3306/<数据库名称>?allowLoadLocalInfile=false&autoDeserialize=false"` 。
> - 当Config配置中有类似last_run_metadata_path的参数时，需要阿里云Logstash服务提供文件路径。目前后端开放了*/ssd/1/<Logstash实例ID>/logstash/data/*路径供您测试使用，且该目录下的数据不会被删除。因此在使用时，请确保磁盘有充足的使用空间。
> - 由于阿里云Logstash创建在专有网络下，配置过程中涉及到阿里云系列产品时，建议使用同一专有网络下的实例。如果使用外网访问阿里云Logstash，需要配置网络与安全，详细信息，请参见配置NAT公网数据传输。
> - 建议使用file_extend打印日志进行测试，不要使用stdout。

6. 单击**下一步**，配置管道参数。

参数	说明
管道工作线程	并行执行管道的Filter和Output的工作线程数量。当事件出现积压或CPU未饱和时，请考虑增大线程数，更好地使用CPU处理能力。默认值：实例的CPU核数。
管道批大小	单个工作线程在尝试执行Filter和Output前，可以从Input收集的最大事件数目。较大的管道批大小可能会带来较大的内存开销。您可以设置LS_HEAP_SIZE变量，来增大JVM堆大小，从而有效使用该值。默认值：125。
管道批延迟	创建管道事件批时，将过小的批分派给管道工作线程之前，要等候每个事件的时长，单位为毫秒。默认值：50ms。
队列类型	用于事件缓冲的内部排队模型。可选值： - **MEMORY**：默认值。基于内存的传统队列。 - **PERSISTED**：基于磁盘的ACKed队列（持久队列）。
队列最大字节数	请确保该值小于您的磁盘总容量。默认值：1024 MB。
队列检查点写入数	启用持久性队列时，在强制执行检查点之前已写入事件的最大数目。设置为0，表示无限制。默认值：1024。

> 警告 配置完成后，需要保存并部署才能生效。保存并部署操作会触发实例重启，请在不影响业务的前提下，继续执行以下步骤。

7. 单击**保存**或者**保存并部署**。
   - **保存**：将管道信息保存在Logstash里并触发实例变更，配置不会生效。保存后，系统会返回**管道管理**页面。可在**管道列表**区域，单击**操作**列下的**立即部署**，触发实例重启，使配置生效。
   - **保存并部署**：保存并且部署后，会触发实例重启，使配置生效。
8. 在创建成功提示框中，单击**确认**，在**管道列表**中查看创建成功的管道。

   等待实例重启完成后，即可完成管道任务的创建。

修改管道

> 警告 修改管道后，在保存并部署时会触发实例重启，请在不影响业务的情况下执行操作。

1. 在**管道列表**区域，单击目标管道右侧**操作**列下的**修改管道**。
2. 在**修改管道任务**页面，修改管道的**Config配置**和**管道参数配置**（**管道ID**不可修改）。
3. 单击**保存**或**保存并部署**，等待实例重启完成后，即可完成管道修改。

复制管道

> 警告 复制管道后，在保存并部署时会触发实例重启，请在不影响业务的情况下执行操作。

1. 在**管道列表**区域，选择目标管道右侧**操作**列下的 **> 复制管道**。
2. 在**复制管道任务**页面，修改**管道ID**，其他配置保持不变。
3. 单击**保存**或**保存并部署**，等待实例重启完成后，即可完成管道复制。

删除管道

> 警告
> - 管道删除后无法恢复，正在运行的管道任务会被中断，请确认后操作。
> - 管道删除操作会触发实例变更，请在不影响业务的情况下执行操作。

1. 在**管道列表**区域，选择目标管道右侧**操作**列下的 **> 删除管道**。
2. 在**删除管道**对话框中，查看风险提示。
3. 单击**继续**，等待实例变更完成后，即可删除管道。

相关文档

- 创建管道的API文档：CreatePipelines
- 通过Logstash迁移数据的最佳实践文档：
  - 通过阿里云Logstash将自建Elasticsearch数据迁移至阿里云

- 腾讯云Elasticsearch数据迁移至阿里云

  > **说明** 该文档同样适用于跨账号、跨地域的阿里云Elasticsearch之间的数据迁移。

- 通过Logstash同步MySQL数据的最佳实践文档：通过Logstash将RDS MySQL数据同步至Elasticsearch
- 通过Logstash同步日志数据的文档：
  - logstash-input-sls插件使用说明
  - 使用Filebeat+Kafka+Logstash+Elasticsearch构建日志分析系统

常见问题

- 管道创建后，进程卡住了，实例变更进度不变，如何处理？
- Logstash数据写入问题排查方案

9.2. Logstash配置文件说明

Logstash通过管道完成数据的采集与处理，管道配置中包含input、output和filter（可选）插件，input和output用来配置输入和输出数据源、filter用来对数据进行过滤或预处理。本文为您介绍阿里云Logstash管道配置文件的详细说明。

本文内容参考了Logstash官方文档，最新内容请参见Structure of a Config File。

您可以通过配置文件管理方式修改Logstash的配置文件，完成数据的采集与处理。详细信息，请参见通过配置文件管理管道。

配置文件结构

Logstash的管道配置文件对每种类型的插件都提供了一个单独的配置部分，用于处理管道事件。

```
input {
  ...
}
filter {
  ...
}
output {
  ...
}
```

每个配置部分可以包含一个或多个插件。例如，指定多个filter插件，Logstash会按照它们在配置文件中出现的顺序进行处理。

注意

- 如果管道配置中有类似last_run_metadata_path的参数，您需要将其设置为Logstash服务的文件路径。目前阿里云Logstash后端开放了 `/ssd/1/<Logstash实例ID>/logstash/data/` 路径供您测试使用，且该目录下的数据不会被删除，因此在使用时，请确保磁盘有充足的使用空间。
- 为了提升安全性，如果在配置管道时使用了JDBC驱动，需要在jdbc_connection_string参数后面添加 `allowLoadLocalInfile=false&autoDeserialize=false` ，否则当您在添加Logstash配置文件的时候，调度系统会抛出校验失败的提示，例如 `jdbc_connection_string => "jdbc:mysql://xxx.drds.aliyuncs.com:3306/<数据库名称>?allowLoadLocalInfile=false&autoDeserialize=false"` 。

插件配置

插件的配置包括插件名称，以及名称中包含的一组插件配置属性。例如，以下input部分包含了两个beats插件，每个beats插件中都配置了port和host属性。

```
input {
  beats {
    port => 8000
    host => "118.11.xx.xx"
  }
  beats {
    port => 8001
    host => "192.168.xx.xx"
  }
}
```

插件支持的属性因插件类型而异，各插件的详细信息请参见输入插件、输出插件、过滤器插件和编解码器插件。

值类型

配置插件时，您可以设置插件的值类型，例如布尔值、列表、哈希等。插件支持的值类型如下。

数组

目前不推荐使用此类型，而建议使用标准类型（例如String），使用插件定义 `:list => true` 属性，以便更好地进行类型检查。数组类型目前被用于处理不需要类型检查的哈希表或混合类型列表，示例如下。

```
users => [ {id => 1, name => bob}, {id => 2, name => jane} ]
```

列表

列表本身不具备类型特征，但其所包含的属性具有类型特征，这样就可以键入检查多个值。您可以通过列表的形式，在声明参数时指定启用检查 `:list => true` ，示例如下。

```
path => [ "/var/log/messages", "/var/log/*.log" ]
uris => [ "http://elastic.co", "http://example.net" ]
```

以上示例，将path配置为一个列表，该列表中包含2个字符串。uris也为一个列表，如果所包含的URL无效，会导致事件处理失败。

布尔类型

布尔类型的值必须为true或者false，且不需要引号标注，示例如下。

```
ssl_enable => true
```

字节类型

字节类型的字段，代表有效字节单位的字符串字段。这是在插件选项中，声明特定大小的便捷方法。字节类型支持十进制（k M G T P E Z Y）和二进制（Ki Mi Gi Ti Pi Ei Zi Yi）。二进制单位为base-1024，十进制单位为base-1000。该字段不区分大小写，并且支持值和单位之间的空格。如果未指定单位，则整数字符串表示字节数。示例如下。

```
 my_bytes => "1113"   # 1113 bytes
 my_bytes => "10MiB"  # 10485760 bytes
 my_bytes => "100kib" # 102400 bytes
 my_bytes => "180 mb" # 180000000 bytes
```

编解码器

编解码器是对数据进行编码或者解码后的目标数据类型，在输入和输出插件中都可以使用。输入编解码器提供了在数据输入之前对其进行解码的功能。输出编解码器，提供了在数据输出之前对其进行编码的功能。使用输入或输出编解码器后，您不需要在Logstash管道中单独使用过滤器。

您可以参考官方提供的编译解释器插件文档，查找可用的编解码器。示例如下。

```
codec => "json"
```

哈希

哈希格式是指定键值对的集合，例如 `"field1" => "value1"` 。

> **注意** 多个键值对使用空格进行分隔，而不是逗号。

示例如下。

```
match => {
  "field1" => "value1"
  "field2" => "value2"
  ...
}
# 单独一行，多个键值对使用空格进行分隔，而不是逗号。
match => { "field1" => "value1" "field2" => "value2" }
```

数值

必须是有效的数值（浮点数或整数）。示例如下。

```
port => 33
```

密码

密码是一个没有记录或打印的单值字符串。示例如下。

```
my_password => "password"
```

URI

URI可以是任何内容，例如完整的URL或者简单的标识符（如foobar）。如果URI中包含类似 `http://user:paas@example.net` 的密码，那么密码部分不会被记录或打印。示例如下。

```
 my_uri => "http://foo:bar@example.net"
```

路径

路径是代表有效操作系统路径的字符串。示例如下。

```
my_path => "/tmp/logstash"
```

字符串

字符串必须是单个字符序列，且必须用双引号或单引号括起来。

转义序列

默认情况下，Logstash不启用转义序列。如果您希望在带引号的字符串使用转义序列，需要在*logstash.yml*中设置 `config.support_escapes: true` 。设置后，字符串（双精度和单精度）将会按照下表进行转义。

转义字符	意义	ASCII码值（十进制）
\r	回车	013
\n	换行	010
\t	跳到下一个Tab位置	009
\\	反斜杠	092
\"	双引号	034
\'	单引号	039

示例如下。

```
name => 'It\'s a beautiful day'
```

注释

注释以 `#` 开头，不需要在行首。示例如下。

```
# 这是一条注释。
input { # 您也可以在行内添加注释。
  # ...
}
```

9.3. 使用Logstash管道配置调试功能

Logstash管道配置调试

当Logstash管道配置出现错误时，可能导致输出数据结果不符合预期，需要反复去目标端确认数据格式，再返回控制台修改配置，这样会耗费大量的时间和人力。对于这种情况，您可以通过阿里云Logstash提供的管道配置调试功能，在管道配置完成后，直接在控制台上查看管道配置的输出结果，降低调试成本。本文介绍具体的实现方法。

前提条件

已安装logstash-output-file_extend插件。如果还未安装，请先安装该插件，安装方法请参见安装或卸载插件。

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 在顶部菜单栏处，选择地域。
3.
4. 在左侧导航栏，单击**管道管理**。
5. 单击**创建管道**。
6. 配置并启动管道。
   i. 在**Config配置**中，填写**管道ID**和**Config配置**。

参数	说明
管道ID	自定义输入。输入后，管道ID会自动映射到file_extend的path路径下。
Config配置	**Config配置**由三部分组成： ■ input：指定待读取的数据源，支持Logstash自带的input plugins（除过file插件）。 ■ filter：进一步加工处理数据源采集到的数据，支持丰富的Filter plugins。 ■ output：将过滤后的数据发送到特定的目的端。阿里云Logstash不仅支持开源的 Logstash output plugins，还可通过配置特有的file_extend插件，开启调试日志功能，即可在管道部署完成后直接查看输出结果，并进行验证与调试。

config配置示例如下。

```
input {
     elasticsearch {
       hosts => "http://es-cn-0pp1jxv000****.elasticsearch.aliyuncs.com:9200"
       user  => "elastic"
       index => "twitter"
       password => "<your_password>"
       docinfo => true
       schedule => "* * * * *"
     }
}
filter {
}
output {
    elasticsearch {
    hosts => ["http://es-cn-000000000i****.elasticsearch.aliyuncs.com:9200"]
    user => "elastic"
    password => "<your_password>"
    index => "%{[@metadata][_index]}"
    document_id => "%{[@metadata][_id]}"
  }
   file_extend {
     path => "/ssd/1/ls-cn-v0h1kzca****/logstash/logs/debug/test"
   }
}
```

注意

- output中的file_extend配置默认为注释状态，如果需要使用调试功能，请先删除注释。
- file_extend中的path参数默认为系统指定路径，请勿修改。您也可以单击**开启配置调试**获取path路径。
- path中的{pipelineid}将自动映射为管道ID，请勿修改为其他名称，否则无法获取调试日志。

ii. 单击**下一步**，配置管道参数。

管道配置参数的详细信息，请参见通过配置文件管理管道。

iii. 保存并部署管道。

- **保存**：将管道信息保存在Logstash里并触发实例变更，配置不会生效。保存后，系统会返回**管道管理**页面。可在**管道列表**区域，单击**操作**列下的**立即部署**，触发实例重启，使配置生效。
- **保存并部署**：保存并且部署后，会触发实例重启，使配置生效。

iv. 在创建成功提示框中，单击**确认**。

7. 查看调试日志。

i. 等待实例重启完成后，在**管道列表**中，单击目标管道右侧**操作**列下的**查看调试日志**。

ii. 在**日志查询**页面的**调试日志**页签中，获取管道处理后的输出数据。

对于多个管道，您可在搜索框中输入pipelineId: <管道ID>过滤对应的日志。

10.最佳实践

10.1. 通过Logstash修改字段名

在某些业务的使用场景下，您可能需要对索引的一些字段进行重命名。例如，使用DataWorks在阿里云Elasticsearch集群间迁移数据时，由于源集群数据中包含了特殊符号（例如@），而DataWorks不支持特殊符号，因此需要修改字段名（去掉特殊符号）后再进行数据迁移。本文介绍如何通过Logstash修改字段名。

背景信息

您可以通过两种方式修改字段名：

- 使用Logstash的filter，对字段进行重命名。

  本文采用此方式，以去除源索引字段的@符号为例进行演示。即将源索引字段@ctxt_user_info，重命名为目标索引字段ctxt_user_info。

- 使用Reindex迁移时，对字段进行重命名。

前提条件

您已完成以下操作：

- 创建阿里云Elasticsearch实例。

  具体操作，请参见创建阿里云Elasticsearch实例，本文以7.10版本实例为例。

- 创建阿里云Logstash实例，需要与Elasticsearch实例在同一专有网络下。

  具体操作，请参见创建阿里云Logstash实例。

- 准备测试数据

  本文使用的测试数据如下。可以看到源索引字段@ctxt_user_info中包含@字符。

```
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 6,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "product_info",
        "_type" : "_doc",
        "_id" : "rpN7fn0BKQKHRO31rK6C",
        "_score" : 1.0,
        "_source" : {
          "@ctxt_user_info" : "test1"
        }
      },
      {
        "_index" : "product_info",
        "_type" : "_doc",
        "_id" : "r5N7fn0BKQKHRO31rK6C",
        "_score" : 1.0,
        "_source" : {
          "@ctxt_user_info" : "test2"
        }
      }
    ]
  }
}
```


操作流程

1. 步骤一：创建目标索引（可选）
2. 步骤二：创建并配置Logstash管道
3. 步骤三：验证结果

步骤一：创建目标索引（可选）

如果您开启了Elasticsearch的自动创建索引功能，可忽略此步骤。

> **说明** 自动创建的索引可能不符合您的预期，不建议开启。

1. 登录目标阿里云Elasticsearch实例的Kibana控制台。

   具体操作，请参见登录Kibana控制台。

> **说明** 本文以阿里云Elasticsearch 7.10版本为例，其他版本操作可能略有差别，请以实际界面为准。

2. 根据页面提示进入Kibana主页，单击右上角的**Dev tools**。
3. 在**Console**页签，执行以下脚本，创建目标索引product_info2。

```
PUT /product_info2
{
    "settings": {
        "number_of_shards": 5,
        "number_of_replicas": 1
    },
    "mappings": {
        "properties": {
            "ctxt_user_info": {
                "type": "text",
                "fields": {
                    "keyword": {
                        "type": "keyword",
                        "ignore_above": 256
                    }
                }
            }
        }
    }
}
```

步骤二：创建并配置Logstash管道

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**管道管理**。
4. 单击**创建管道**。
5. 在**创建管道任务**页面，输入管道ID并配置管道。

   本文使用的管道配置如下。

```
input {
    elasticsearch {
        hosts => ["http://es-cn-tl32gid**********.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "your_password"
        index => "product_info"
        docinfo => true
    }
}
filter {
    mutate {
        rename => { "@ctxt_user_info" => "ctxt_user_info" }
    }
}
output {
    elasticsearch {
        hosts => ["http://es-cn-tl32gid**********.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "your_password"
        index => "product_info2"
        document_type => "%{[@metadata][_type]}"
        document_id => "%{[@metadata][_id]}"
    }
}
```

以上管道配置中，通过Logstash的filter.mutate.rename参数实现索引的重命名。

更多管道配置说明，请参见通过配置文件管理管道和Logstash配置文件说明。

6. 单击**保存**或者**保存并部署**。
   - **保存**：将管道信息保存在Logstash里并触发实例变更，配置不会生效。保存后，系统会返回**管道管理**页面。可在**管道列表**区域，单击**操作**列下的**立即部署**，触发实例重启，使配置生效。
   - **保存并部署**：保存并且部署后，会触发实例重启，使配置生效。

步骤三：验证结果

1. 登录目标阿里云Elasticsearch的Kibana控制台。

   具体操作请参见登录Kibana控制台。
2. 根据页面提示进入Kibana主页，单击右上角的**Dev tools**。
3. 在**Console**页签，执行以下脚本，查询目标索引中的信息。

   ```
   GET product_info2/_search
   ```

   返回结果如下。

```
{
  "took" : 4,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 6,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "product_info2",
        "_type" : "_doc",
        "_id" : "r5N7fn0BKQKHRO31rK6C",
        "_score" : 1.0,
        "_source" : {
          "@timestamp" : "2021-12-03T04:14:26.872Z",
          "@version" : "1",
          "ctxt_user_info" : "test1"
        }
      },
      {
        "_index" : "product_info2",
        "_type" : "_doc",
        "_id" : "rpN7fn0BKQKHRO31rK6C",
        "_score" : 1.0,
        "_source" : {
          "@timestamp" : "2021-12-03T04:14:26.871Z",
          "@version" : "1",
          "ctxt_user_info" : "test2"
        }
      }
    ]
  }
}
```


根据结果可以看到，源索引字段@ctxt_user_info中的@已经去除，索引字段被重命名为ctxt_user_info。

10.2. 通过Logstash实现多字段数据整合

在使用Logstash传输数据时，如果您需要通过合并字段来整合数据，例如将a字段和b字段合并为一个新的c字段，然后移除a字段和b字段，可以通过logstash-filter-mutate插件的多个模块实现。此插件为系统默认安装插件，无须再安装，且不支持卸载。本文介绍如何通过logstash-filter-mutate插件实现多字段合并。

背景信息

logstash-filter-mutate插件支持对事件中的字段进行重命名、删除、替换和修改操作。配置文件中的mutate按照下表中的顺序执行，详细信息请参见Mutate filter plugin。

模块	输入类型
coerce	hash
rename	hash
update	hash
replace	hash
convert	hash
gsub	array
uppercase	array
capitalize	array
lowercase	array
strip	array
remove_field	array
split	hash
join	hash
merge	hash
copy	hash

前提条件

- 创建阿里云Elasticsearch实例。

  具体操作，请参见创建阿里云Elasticsearch实例，本文以7.10版本实例为例。

- 开启目标Elasticsearch实例的自动创建索引功能。

  具体操作请参见配置YML参数。

  > **说明** 自动创建的索引可能不符合您的预期，不建议开启，本文仅供测试。在实际业务中，建议您先在目标Elasticsearch实例中创建索引，再通过Logstash传输数据。创建索引的具体操作请参见快速入门。

- 创建阿里云Logstash实例，需要与Elasticsearch实例在同一专有网络下。

  具体操作，请参见创建阿里云Logstash实例。

- 在源阿里云Elasticsearch中准备测试数据。

  本文使用的测试数据如下。其中源索引的名称为yc_text，待合并的字段为app.name和message。

```
{
  "took" : 2,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 6,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "yc_text",
        "_type" : "_doc",
        "_id" : "HpIduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "app.name" : "app1",
          "annual_rate" : "31%",
          "describe" : "可以自助选择消息推送",
          "message" : "10000"
        }
      },
      {
        "_index" : "yc_text",
        "_type" : "_doc",
        "_id" : "H5IduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "app.name" : "app2",
          "annual_rate" : "35%",
          "describe" : "每天收益到账消息推送",
          "message" : "10001"
        }
      },
      {
        "_index" : "yc_text",
        "_type" : "_doc",
        "_id" : "IpIduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "app.name" : "app3",
          "annual rate" : "30",
          "describe" : "每天收益会消息推送",
          "message" : "10004"
        }
      },
      {
        "_index" : "yc_text",
```

```
        "_type" : "_doc",
        "_id" : "IJIduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "app.name" : "app4",
          "annual_rate" : "38%",
          "describe" : "每天收益立即到账消息推送",
          "message" : "10002"
        }
      },
      {
        "_index" : "yc_text",
        "_type" : "_doc",
        "_id" : "IZIduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "app.name" : "app5",
          "annual_rate" : "40%",
          "describe" : "每天收益到账消息推送",
          "message" : "10003"
        }
      },
      {
        "_index" : "yc_text",
        "_type" : "_doc",
        "_id" : "I5IduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "app.name" : "app6",
          "annual_rate" : "33%",
          "describe" : "通过短信提示获取收益消息",
          "message" : "10005"
        }
      }
    ]
  }
}
```


操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**管道管理**。
4. 单击**创建管道**。
5. 在**创建管道任务**页面，输入管道ID并配置管道。

   本文使用的管道配置如下。

```
input {
    elasticsearch {
        hosts => ["http://es-cn-tl3264bqv001d****.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "your_password"
        index => "yc_text"
        docinfo => true
    }
}
filter {
    mutate {
        merge =>  { "message" => "app.name" }
    }
    mutate {
        rename => [ "message","anger" ]
    }
    mutate {
        remove_field => [ "app.name" ]
    }
}
output {
    elasticsearch {
        hosts => ["http://es-cn-tl3264bqv001d****.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "your_password"
        index => "yc_text_new"
        document_type => "%{[@metadata][_type]}"
        document_id => "%{[@metadata][_id]}"
    }
}
```

以上管道配置的原理说明如下：

i. 通过Logstash的filter.mutate.merge参数合并源索引yc_text中的app.name和message两个字段。合并后，Logstash会使用message字段存放原message和app.name两个字段的数据。

ii. 通过filter.mutate.rename参数将合并后的message字段重命名为anger。

iii. 前两步完成后，app.name字段还会继续存在。为了避免出现重复数据，管道配置中使用filter.mutate.remove_field将app.name字段移除。

iv. 最后将合并后的字段anger字段传输到yc_text_new索引中。

> **说明** logstash-filter-mutate插件会按照优先级执行mutate块中的定义的操作，详细信息请参见本文的背景信息。但是您可以通过使用不同的mutate块来控制这个顺序，例如以上管道配置中使用了三个mutate块来控制执行顺序为rename、merge、remove_field。

更多管道配置说明，请参见通过配置文件管理管道和Logstash配置文件说明。

6. 单击**保存**或者**保存并部署**。
   - **保存**：将管道信息保存在Logstash里并触发实例变更，配置不会生效。保存后，系统会返回**管道管理**页面。可在**管道列表**区域，单击**操作**列下的**立即部署**，触发实例重启，使配置生效。
   - **保存并部署**：保存并且部署后，会触发实例重启，使配置生效。

验证结果

1. 登录目标阿里云Elasticsearch的Kibana控制台。

   具体操作请参见登录Kibana控制台。

2. 根据页面提示进入Kibana主页，单击右上角的**Dev tools**。

3. 在**Console**页签，执行以下脚本，查询目标索引中的信息。

```
GET yc_text_new/_search
{
  "query": {
    "match_all": {}
  }
}
```

返回结果如下。

```
{
  "took" : 1,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 6,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "yc_text_new",
        "_type" : "_doc",
        "_id" : "H5IduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "describe" : "每天收益到账消息推送",
          "@version" : "1",
          "anger" : [
            "10001",
            "app2"
          ],
          "@timestamp" : "2021-12-15T03:45:25.321Z",
          "annual_rate" : "35%"
        }
      },
      {
        "_index" : "yc_text_new",
        "_type" : "_doc",
        "_id" : "IZIduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
```

```
      "_source" : {
        "describe" : "每天收益到账消息推送",
        "@version" : "1",
        "anger" : [
          "10003",
          "app5"
        ],
        "@timestamp" : "2021-12-15T03:45:25.321Z",
        "annual_rate" : "40%"
      }
    },
    {
      "_index" : "yc_text_new",
      "_type" : "_doc",
      "_id" : "I5IduH0BWiRrY8Azn65i",
      "_score" : 1.0,
      "_source" : {
        "describe" : "通过短信提示获取收益消息",
        "@version" : "1",
        "anger" : [
          "10005",
          "app6"
        ],
        "@timestamp" : "2021-12-15T03:45:25.322Z",
        "annual_rate" : "33%"
      }
    },
    {
      "_index" : "yc_text_new",
      "_type" : "_doc",
      "_id" : "HpIduH0BWiRrY8Azn65i",
      "_score" : 1.0,
      "_source" : {
        "describe" : "可以自助选择消息推送",
        "@version" : "1",
        "anger" : [
          "10000",
          "app1"
        ],
        "@timestamp" : "2021-12-15T03:45:25.298Z",
        "annual_rate" : "31%"
      }
    },
    {
      "_index" : "yc_text_new",
      "_type" : "_doc",
      "_id" : "IJIduH0BWiRrY8Azn65i",
      "_score" : 1.0,
      "_source" : {
        "describe" : "每天收益立即到账消息推送",
        "@version" : "1",
        "anger" : [
          "10002",
          "app4"
        ],
```

```
          "@timestamp" : "2021-12-15T03:45:25.321Z",
          "annual_rate" : "38%"
        }
      },
      {
        "_index" : "yc_text_new",
        "_type" : "_doc",
        "_id" : "IpIduH0BWiRrY8Azn65i",
        "_score" : 1.0,
        "_source" : {
          "describe" : "每天收益会消息推送",
          "@version" : "1",
          "anger" : [
            "10004",
            "app3"
          ],
          "@timestamp" : "2021-12-15T03:45:25.321Z",
          "annual rate" : "30"
        }
      }
    ]
  }
}
```


根据结果可以看到，新字段anger已经整合了旧字段message和app.name的数据，且app.name字段已被移除。

10.3. 通过Logstash切分数据并提取到字段中

在使用Logstash传输数据时，在某些业务使用场景中，您可能需要切分源端数据并提取到字段中再写入目标端Elasticsearch集群。例如，源端Logs日志中存在以竖线（ | ）分隔的数据，此时您可以通过Logstash按照 | 切分数据并提取到字段中，再输出到目标端Elasticsearch集群。本文介绍如何通过Logstash切分数据并提取到字段中。

背景信息

logstash-filter-mutate插件（过滤器插件）支持对事件中的字段进行切分、重命名、删除、替换和修改等操作，详细信息请参见Mutate filter plugin。所有的过滤器插件都支持以下常见的可选配置项，详细信息请参见Common Option。

配置项	输入类型
add_field	hash
add_tag	array
enable_metric	boolean
id	string

配置项	输入类型
periodic_flush	boolean
remove_field	array
remove_tag	array

前提条件

您已完成以下操作：

- 创建阿里云Elasticsearch实例。

  具体操作，请参见创建阿里云Elasticsearch实例，本文以7.10版本实例为例。

- 开启目标Elasticsearch实例的自动创建索引功能。

  具体操作请参见配置YML参数。

  > **说明** 自动创建的索引可能不符合您的预期，不建议开启，本文仅供测试。在实际业务中，建议您先在目标Elasticsearch实例中创建索引，再通过Logstash传输数据。创建索引的具体操作请参见快速入门。

- 创建阿里云Logstash实例，需要与Elasticsearch实例在同一专有网络下。

  具体操作，请参见创建阿里云Logstash实例。

- 准备测试数据。

  本文以Beats采集的Logs中的某一条数据为例，关于Beats采集数据的详细信息请参见采集ECS服务日志。如下测试数据中的LogMessage以 `|` 分隔，并存在多个 `||||` 特殊符号。本文使用Logstash按照 `|` 切分数据，被切分的字段分别写入到对应的字段中：mobile、appName、type、timestamp、status、code、component、cid、serviceId、serviceName、serviceType、param，最后将字段输出到目标端Elasticsearch集群中。

  ```
  LogMessage: |1390000****|jop|byORP|2022-04-18T14:18:16.633|/log/cms/send|200|pluginNums=0
  ,pluginStatus=0||||||
  ```

操作步骤

1. 登录阿里云Elasticsearch控制台。
2. 进入目标实例。
   i. 在顶部菜单栏处，选择地域。
   ii. 在左侧导航栏，单击**Logstash实例**，然后在**Logstash实例**中单击目标实例ID。
3. 在左侧导航栏，单击**管道管理**。
4. 单击**创建管道**。
5. 在**创建管道任务**页面，输入管道ID并配置管道。

   本文使用的管道配置如下。

```
input {
    beats {
        port => 8001
    }
}
filter {
    mutate {
        gsub => ["message","\|","| "]
        split => ["message","|"]
        add_field => {
            "mobile" => "%{[message][1]}"
            "appName" => "%{[message][2]}"
            "type" => "%{[message][3]}"
            "timestamp" => "%{[message][4]}"
            "status" => "%{[message][5]}"
            "code" => "%{[message][6]}"
            "component" => "%{[message][7]}"
            "cid" => "%{[message][8]}"
            "serviceId" => "%{[message][9]}"
            "serviceName" => "%{[message][10]}"
            "serviceType" => "%{[message][11]}"
            "param" => "%{[message][12]}"
        }
    }
    mutate {
        strip => ["mobile","appName","type","timestamp","status","code","component","ci
d","serviceId","serviceName","serviceType","param"]
    }
}
output {
    elasticsearch {
        index => "<yourIndexName>"
        hosts => ["es-cn-7mz2mu1zp0006****.elasticsearch.aliyuncs.com:9200"]
        user => "elastic"
        password => "<yourPassword>"
    }
}
```

注意

- input.beat.port为Beats采集日志输入到当前Logstash管道的端口，需要使用8000~9000范围内的端口。
- 以上管道配置中的 `gsub => ["message","\|","| "]` ，第二个 `|` 后有一个空格。
- 以上管道配置中的index、hosts和password参数值需要替换为您实际业务的索引名称、Elasticsearch集群的访问地址和集群的elastic账号对应的密码。

以上管道配置的原理说明如下：

i. 通过Logstash的filter.mutate.gsub参数，使用正则表达式 `\|` 去匹配LogMessage中的 `|` ，将 `|` 替换为 `| ` （即 `|` +空格）。替换后的效果如下。

```
LogMessage: | 1390000****| jop| byORP| 2022-04-18T14:18:16.633| /log/cms/send| 200|
pluginNums=0,pluginStatus=0| | | | | |
```

ii. 通过filter.mutate.split参数将LogMessage按照 `|` 进行切分。

iii. 通过filter.mutate.add_field参数添加字段，即将切分后的LogMessage一一添加到对应的字段中。添加后的效果如下。

```
"mobile":" 1390000****",
"appName":" jop",
"type":" byORP",
"timestamp":" 2022-04-18T14:18:16.633",
"status":" /log/cms/sen",
"code":" 200",
"component":" pluginNums=0,pluginStatus=0",
"cid":" ",
"serviceId":" ",
"serviceName":" ",
"serviceType":" ",
"param":" "
```

iv. 通过filter.mutate.strip参数去除字段空格。由于添加后的每个字段前面都有一个空格，因此需要去除这些空格。

更多管道配置说明，请参见通过配置文件管理管道和Logstash配置文件说明。

6. 单击**保存**或者**保存并部署**。
   - **保存**：将管道信息保存在Logstash里并触发实例变更，配置不会生效。保存后，系统会返回**管道管理**页面。可在**管道列表**区域，单击**操作**列下的**立即部署**，触发实例重启，使配置生效。
   - **保存并部署**：保存并且部署后，会触发实例重启，使配置生效。

验证结果

1. 登录目标阿里云Elasticsearch实例的Kibana控制台，根据页面提示进入Kibana主页。

   登录Kibana控制台的具体操作，请参见登录Kibana控制台。

   > **说明** 本文以阿里云Elasticsearch 7.10.0版本为例，其他版本操作可能略有差别，请以实际界面为准。

2. 单击右上角的**Dev tools**。
3. 在**Console**中，执行以下脚本，查询目标索引中的信息。

```
GET <yourIndexName>/_search
{
  "query": {
    "match_all": {}
  }
}
```

> **说明** <yourIndexName>需要与管道配置中的index参数值保持一致。

预期结果如下。

```
{
  "took" : 1,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 1.0,
    "hits" : [
      {
        "_index" : "<yourIndexName>",
        "_type" : "_doc",
        "_id" : "Lb1UWoAB-6Zo6en4luDi",
        "_score" : 1.0,
        "_source" : {
          "mobile" : "1390000****",
          "appName" : "jop",
          "type" : "byORP",
          "timestamp" : "2022-04-18T14:18:16.633",
          "status" : "/log/cms/sen",
          "code" : "200",
          "component" : "pluginNums=0,pluginStatus=0",
          "cid" : "",
          "serviceId" : "",
          "serviceName" : "",
          "serviceType" : "",
          "param" : ""
        }
      }
    ]
  }
}
```

11.常见问题

11.1. Logstash FAQ

本文介绍使用阿里云Logstash的常见问题。

Logstash支持将数据源配置为DRDS吗？

支持。可参考RDS MySQL数据迁移方案进行配置，具体操作请参见通过Logstash将RDS MySQL数据同步至Elasticsearch。

如何将公网数据导入或导出到Logstash中？

Logstash实例部署在专有网络VPC（Virtual Private Cloud）下，可以通过配置NAT网关实现与公网的连通。具体操作，请参见配置NAT公网数据传输。

使用自建Kafka作为Logstash的输入或者输出时，出现错误日志，如何处理？

常见错误日志如下：

- No entry found for connection

  原因：Logstash节点无法解析到Kafka服务的hostname对应的IP地址。

  解决方法：在 `server.properties` 中添加配置信息，以Kafka服务运行在10.10.10.10的9092端口为例，配置信息如下。

  ```
  listeners=PLAINTEXT://10.10.10.10:9092
  advertised.listeners=PLAINTEXT://10.10.10.10:9092
  ```

  > **注意**
  > - 在配置信息时，请将 `10.10.10.10:9092` 替换为您Kafka集群的IP地址和端口号。
  > - 推荐您使用阿里云Kafka服务，并且保证Logstash所在节点的IP地址在Kafka的访问白名单内。

- could not be established. Broker may not be available

  原因：Kafka服务不存在或者无法连接。

  解决方法：检查Kafka服务是否正常运行，或者Logstash管道配置中的 `bootstrap_servers` 配置是否正确。

阿里云Logstash的JDBC支持MySQL数据库吗？

支持。需要上传mysql-connector-java驱动文件，具体操作请参见配置扩展文件。

Logstash支持节点监控吗？

支持。可通过配置X-Pack，关联阿里云Elasticsearch实例后，在Kibana中监控Logstash节点。具体操作，请参见配置X-Pack监控。

Logstash支持上传脚本文件吗？

不支持。目前，Logstash只支持通过Config配置文件配置管道，实现数据传输。详细信息，请参见通过配置文件管理管道。

Logstash支持配置HTTP采集协议吗？

支持。Logstash支持通过HTTP或HTTPS接收单行或多行事件，详细信息请参见Http input plugin。

> **说明** 阿里云Logstash默认不提供公网访问能力，如果您需要采集公网HTTP请求，可通过配置NAT网关实现。具体操作，请参见配置NAT公网数据传输。

如何通过Logstash，将日志服务数据同步到Elasticsearch上？

您可以通过logstash-input-sls插件实现，具体操作请参见logstash-input-sls插件使用说明。

Logstash能够实时同步数据吗？

Logstash是准实时同步工具。只要您不停止管道任务，且源端有数据，Logstash就会一直向目标端写入数据。

管道创建后，进程卡住了，实例变更进度不变，如何处理？

查看实例的主日志是否有报错，根据报错判断原因，具体操作请参见查询日志。常见的原因及解决方法如下。

原因	解决方法
管道配置错误。	中断变更，等到实例处于变更中断状态后，修改管道配置，触发重启恢复。具体操作，请参见查看实例任务进度详情。
集群磁盘使用率过高。	升级实例规格。具体操作，请参见升配集群。完成后，刷新实例，观察变更进度。
output为elasticsearch时，没有开启Elasticsearch实例的自动创建索引功能。	开启Elasticsearch实例的自动创建索引功能。具体操作，请参见配置YML参数。完成后，刷新实例，观察变更进度。
input为beats时，port没有使用8000~9000的端口。	中断变更，等到实例处于变更中断状态后，在管道配置中，修改port为8000~9000的端口，触发重启恢复。

原因	解决方法
源端或目标端的使用了外网地址。	选择以下任意一种方式处理： • 中断变更，等到实例处于变更中断状态后，在管道配置中，将外网地址修改为内网地址。 • 配置NAT网关实现公网数据传输。具体操作，请参见配置NAT公网数据传输。完成后，刷新实例，观察变更进度。
管道配置中包含了file_extend，但没有安装logstash-output-file_extend插件。	选择以下任意一种方式处理： • 安装logstash-output-file_extend插件。具体操作，请参见使用Logstash管道配置调试功能。完成后，刷新实例，观察变更进度。 • 中断变更，等到实例处于变更中断状态后，在管道配置中，去掉file_extend配置，触发重启恢复。

11.2. Logstash数据写入问题排查方案

在使用阿里云Logstash将数据写入阿里云Elasticsearch（output指定为阿里云Elasticsearch）时，您可能会遇到网络不通、管道配置错误、负载高、管道正常启动但无数据写入目标端以及服务正常但缺少数据等问题，此时您可以参考本文的排查方案进行排查解决。

网络不通

排查方案	常见错误案例	建议解决方案
分别检查Logstash是否与源端和目标端服务在同一网络下。 说明 阿里云Logstash和阿里云Elasticsearch服务部署在专有网络环境下，建议您将业务部署在相同的专有网络下。	源端服务在公网环境下，而Logstash在专有网络环境下。	选择以下任意一种方式处理： • 借助网络产品功能，打通网络环境。 • 参见配置NAT公网数据传输，配置NAT网关实现公网数据传输。 • 重新购买同一专有网络下的Logstash和Elasticsearch实例，并重新配置管道。

检查NAT配置是否错误。	• NAT条目地址或端口定义错误。
	• NAT网关类型不符合场景。
		根据具体情况，按照以下方式处理：
		• 检查NAT条目地址和端口，确保网络互通。
		• SNAT和DNAT使用场景不一样，结合业务场景选择正确的网关转换方式：
		  ○ SNAT：Logstash主动访问公网。
		  ○ DNAT：公网服务向Logstash节点推送数据。

检查是否上传了正确的JDBC驱动插件。	PolarDB数据同步场景中，使用高版本的JDBC驱动，日志无报错，但数据写不到目标端，换成低版本后正常。	选择正确版本的JDBC驱动，详细信息请参见配置扩展文件。

排查方案	常见错误案例	建议解决方案
检查白名单或安全组是否有限制。	通过Filebeat将数据采集到Logstash中处理，Filebeat部署在用户侧ECS上，但ECS未在安全组开放监听端口。	根据具体情况，按照以下方式处理： • 在对应服务的白名单中，加入Logstash节点的IP地址。 说明 获取Logstash的IP地址的具体操作，请参见查看实例的基本信息。 • 在安全组中开启服务监听端口，具体操作请参见添加安全组规则。
检查Logstash管道配置的源端或目标端是否涉及到RAM用户未授权，导致RAM用户无法访问对应服务。	• Logstash的output配置中指定了RAM用户访问Elasticsearch，但Elasticsearch上未对RAM用户设置索引权限。 • Logstash主日志报错401。	根据具体情况，按照以下方式处理： • 为RAM用户授予对应的权限，具体操作请参见为RAM用户授权。 • 使用正确的用户名和密码，并且密码中不能包含特殊字符。

管道配置错误

排查方案	常见错误案例	建议解决方案
参见查询日志，查看Logstash的主日志，检查日志是否存在报错。	未安装插件。例如日志中出现 `Couldn't find any output plugin named 'file_extend'` 错误提示时，说明集群中没有安装logstash-output-file_extend插件。	选择以下任意一种方式处理： • 安装插件。 • 在管道配置中，删除该插件的配置信息。
	配置中存在隐藏的特殊字符。	手动输入配置。
	filter过滤代码有误，例如ruby代码存在错误。	选择以下任意一种方式处理： • 将filter模块配置精简到原始配置，逐步增加过滤配置，找到根因。并根据实际情况处理。 • 借助第三方调试工具，调试正确后再上线。
	管道参数名或参数值写入错误。例如logstash-output-elasticsearch插件中的hosts写成了host、RDS实例名称不正确等。	参见Logstash官方文档或阿里云Elasticseatch官方最佳实践文档编写管道配置。

排查方案	常见错误案例	建议解决方案
	Logstash和源端或目标端连接超时。例如无法访问Elasticsearch时，会出现 `Elasticsearch Unreachable: [http://xxxx:9200/] [Manticore::ConnectTimeout] connect timed out` 的错误提示。	确保Logstash和Elasticsearch网络互通，并输入正确的源端和目标端地址。
	Elasticsearch开启了HTTPS协议，但Logstash管道配置时使用了http。	修改管道配置，使用与源端和目标端相同的访问协议。

负载问题

排查方案	常见错误案例	建议解决方案
参见集群监控章节，检查节点磁盘使用率是否过高。	• 在管道配置中，指定队列类型为永久型（PERSISTED），数据会永久存储在磁盘上，随着数据的积累，导致磁盘被打满。 • 管道output配置中指定了 `stdout{}` 。	根据具体情况，按照以下方式处理： • 将Logstash管道队列类型指定为默认的内存型（MEMORY），详细信息请参见通过配置文件管理管道。 **说明** 由于阿里云Logstash暂时没有清理磁盘的入口，因此当您遇到磁盘打满的问题时，需要技术人员在后端为您处理。 • 删除管道output配置中的 `stdout{}` 。 **说明** 管道output配置不支持定义 `stdout{}` ，否则会导致磁盘使用率过高的问题。
参见集群监控章节，检查节点内存是否溢出OOM（Out Of Memory）。	内存OOM，节点未拉起。	在控制台重启对应节点。
检查源端或目标端是否存在负载问题。	Elasticsearch集群不健康，影响写入。	暂停写入，优先恢复集群健康，建议扩容。

管道正常启动，但无数据写入目标端

排查方案	常见错误案例	建议解决方案
参见使用Logstash管道配置调试功能，开启Logsatsh的管道配置调试功能（需要安装logstash-output-file_extend插件），查看调试日志，判断是否有数据流入Logstash服务： • 无数据流入Logstash：检查源端配置信息是否正确。 • 有数据流入Logstash：检查目标端配置信息是否正确。	无数据流入Logstash： • 源端配置信息中存在阿里云AccessKey信息，但AccessKey信息已经失效。 • 源端无实时数据产生，例如Filebeat实时采集文件数据，文件无新数据产生。	根据具体情况，选择以下方式处理： • 检查配置信息，并修改不准确的信息。 • 如果Logstash使用了实时流插件，需要确保源端存在实时写入的数据。
	有数据流入Logstash： • 阿里云Elasticsearch实例未开启自动创建索引功能。 • 目标端禁止写入，例如Elasticsearch索引禁止写入。	根据具体情况，选择以下方式处理： • 开启阿里云Elasticsearch实例的自动创建索引功能。 • 确保目标端的可写性。

服务正常缺少数据

排查方案	常见错误案例	建议解决方案
根据管道配置场景，结合管道插件属性排查： • 检查JDBC查询语句是否正确。 • 检查管道配置中的logstash-input-elasticsearch插件是否存在实时写入的数据。	JDBC场景： • 通过查询语句查询的结果中缺少数据。 • 追踪字段为非递增数据，例如时间字段或ID。 • JDBC和Elasticsearch集群不在同一时区。	根据具体情况，选择以下方式处理： • 在源端调试查询语句。 • 检查追踪字段类型是否为官方建议字段类型，优先将字段类型设置为numeric或timestamp。 • 检查时区差异性，并根据检查结果进行相应处理。
	使用logstash-input-elasticsearch插件场景： • 源端数据存在实时写入。 • 管道配置中，定时时间设置较小，数据存在大量的写入，导致目标端数据堆积。	Logstash不适用于数据实时同步场景。如果源端存在实时写入，建议通过拉长定时查询时间，减少频繁在源端和目标端查询和写入。
参见查询日志，查看Logstash慢日志，检查是否存在写入慢的问题。	源端和目标端压力均未达到瓶颈，但Logstash的管道工作线程数使用了官方的默认值。	增加Logstash的管道批大小和工作线程数，详细信息请参见通过配置文件管理管道。