云搜索服务（ES）

用户操作指南

天翼云科技有限公司

目 录

1 简介

1.1 什么是云搜索服务

云搜索服务（ES），为您提供托管的分布式搜索引擎服务，完全兼容开源 Elasticsearch 搜索引擎，支持结构化、非结构化文本的多条件检索、统计、报表。云搜索服务的使用流程和数据库类似。

云搜索服务会自动部署，快速创建 Elasticsearch 集群。免运维，内置搜索最佳调优实践；拥有完善的监控体系，提供一系列系统、集群以及查询性能等关键指标，让用户更专注于业务逻辑的实现。

Elasticsearch 搜索引擎相关内容的深入介绍可参见《Elasticsearch：权威指南》。

优势

云搜索服务是公有云提供的搜索服务，其具备如下优势：

- 高效易用

  TB 级数据毫秒级返回检索结果，提供可视化平台方便数据展示和分析。

- 弹性灵活

  按需申请，在线扩容，零业务中断，快速应对业务增长。

- 自主词库

  支持用户自定义行业词库，词库修改，无需重启实例。

- 无忧运维

  全托管服务，开箱即用，主要操作一键可达，专业团队贴身看护。

- 数据可靠

  支持用户触发以及定时触发的快照备份，支持恢复到本集群以及其他集群的能力，通过快照恢复支持集群的数据迁移。

功能

云搜索服务具备如下功能：

- 专业的集群管理平台

管理控制台提供了丰富的功能菜单，能够让您通过浏览器即可安全、方便地进行集群管理和维护，包括集群管理、运行监控等。

- 完善的监控体系

  通过管理控制台提供的仪表盘（Dashboard）和集群列表，您可以直观看到已创建集群的各种不同状态，可通过指标监控视图了解集群当前运行状况。

- 支持 Elasticsearch 搜索引擎

  提供 Elasticsearch 搜索引擎，Elasticsearch 是基于 Lucene 的当前流行的企业级搜索服务器，具备分布式多用户的能力。其主要功能包括全文检索、结构化搜索、分析、聚合、高亮显示等。能为用户提供实时搜索、稳定可靠的服务。

1.2 Elasticsearch 集群应用场景

云搜索服务 Elasticsearch 类型集群适用于日志分析、站内搜索等场景。

日志分析

对 IT 设备进行运维分析与故障定位、对业务指标分析运营效果。

- 统计分析：20 余种统计分析方法、近 10 种划分维度。
- 实时高效：从入库到能够被检索到，时间差在数秒到数分钟之间。
- 可视化：表格、折线图、热图、云图等多种图表呈现方式。

站内搜索

对网站内容进行关键字检索、对电商网站商品进行检索与推荐。

- 实时检索：站内资料或商品信息更新数秒至数分钟内即可被检索。
- 分类统计：检索同时可以将符合条件的商品进行分类统计。
- 高亮提示：提供高亮能力，页面可自定义高亮显示方式。

1.3 基本概念

集群

云搜索服务是以集群为单位进行组织，一个集群代表一个独立运行的搜索服务，由多个节点构成。

索引

用于存储 Elasticsearch 的数据，类似关系型数据库的 Database。是一个或多个分片分组在一起的逻辑空间。

表1-1 Elasticsearch 与关系型数据库的类比对应关系

Elasticsearch	索引（Index）	文档类型（Types）	文档（Document）	字段（Field）	映射（Mapping）
关系型数据库	Database	Table	Row	Column	Schema

Shard

索引可以存储数据量超过 1 个节点硬件限制的数据。为满足这样的需求，Elasticsearch 提供了一个能力，将一个索引拆分为多个，称为 Shard。当您创建一个索引时，您可以根据实际情况指定 Shard 的数量。每个 Shard 托管在集群中的任一节点中，且每个 Shard 本身是一个独立的、全功能的“索引”。

Shard 的数量只能在创建索引前指定，且在索引创建成功后无法修改。

Replica

Shard 下的实际存储索引的一个副本。可以理解为备份 Shard。副本的存在可以预防单节点故障。使用过程中，您可以根据业务情况增加或减少 Replica 数量。

文档

Elasticsearch 存储的实体，是可以被索引的基本单位，相当于关系型数据库中的行。

文档类型

类似关系型数据库中的表，用于区分不同的数据，1 个索引里面可以包含若干个文档类型。每个文档必须设定它的文档类型。

映射

用来约束字段的类型，可以根据数据自动创建。相当于数据库中的 Schema。

字段

组成文档的最小单位。相当于数据库中的 Column。

1.4 什么是 Kibana

Kibana 是一个开源的数据分析与可视化平台，与 Elasticsearch 搜索引擎一起使用。您可以用 Kibana 搜索、查看、交互存放在 Elasticsearch 索引中的数据，也可以使用 Kibana 以图表、表格、地图等方式展示数据。

Kibana 的官方文档：https://www.elastic.co/guide/en/kibana/current/index.html

一键访问 Kibana

云搜索服务的集群默认提供 Kibana，无需安装部署，一键访问 Kibana。

登录云搜索服务管理控制台。在左侧导航栏，单击“集群管理”进入集群管理列表。在对应集群的“操作”列，单击“Kibana”，即可打开 Kibana 界面。

Kibana 功能

完全兼容开源 Kibana 可视化展现和 Elasticsearch 统计分析能力。

- 支持 10 余种数据呈现方式
- 支持近 20 种数据统计方式
- 支持时间、标签等各种维度分类

1.5 什么是 Cerebro

Cerebro 是使用 Scala、Play Framework、AngularJS 和 Bootstrap 构建的开源的基于 Elasticsearch Web 可视化管理工具。您可以通过 Cerebro 对集群进行 web 可视化管理，如执行 rest 请求、修改 Elasticsearch 配置、监控实时的磁盘，集群负载，内存使用率等。

一键访问 Cerebro

云搜索服务的集群默认提供 Cerebro，无需安装部署，一键访问 Cerebro。

登录云搜索服务管理控制台。在左侧导航栏，单击“集群管理”进入集群管理列表。在对应集群的“操作”列，单击“Cerebro”，即可打开 Cerebro 界面。

打开 Cerebro 后，需要输入集群的内网访问地址，选择其中的一个内网访问地址即可。

- 非安全模式登录时，输入 **http://ip:9200**。
- 安全模式登录时，输入 **https://ip:9200**，并且输入登录安全模式的账号和密码。

Cerebro 功能

完全兼容开源 Cerebro，适配最新 0.8.4 版本

- 支持 Elasticsearch 可视化实时负载监控
- 支持 Elasticsearch 可视化数据管理

1.6 安全模式集群简介

当前我们提供的 Elasticsearch 6.5.4 及之后版本集群为您增加了安全模式功能，当您开启后，安全模式将会为您提供身份验证、授权以及加密等功能。

以下功能介绍以 kibana 可视化界面操作为例。

说明

安全模式只能在创建集群时开启。集群创建成功后，不支持开启或者关闭安全模式。

基本名词解释

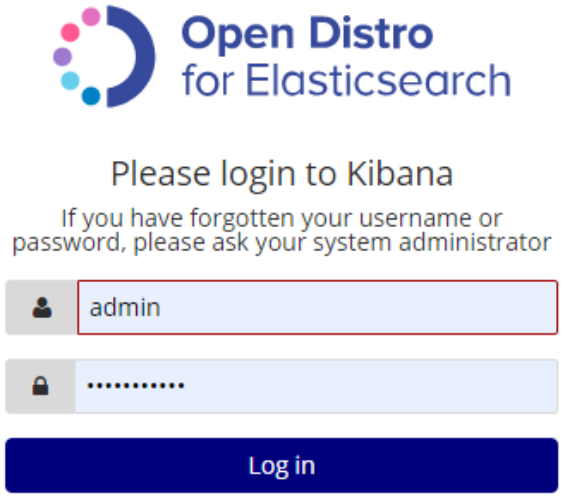
表1-2 安全模式名词解释

名词	描述
Permission	单个动作，例如创建索引（例如 indices:admin/create）。
Action group 操作组	一组权限。例如，预定义的 SEARCH 操作组授权角色使用_search 和_msearchAPI。
Role 角色	角色定义为权限或操作组的组合，包括对集群，索引，文档或字段的操作权限。
Backend role 后端角色	（可选）来自授权后端的其他外部角色（例如 LDAP / Active Directory）。
User 用户	用户可以向 Elasticsearch 集群发出操作请求。用户具有凭证（例如，用户名和密码）、零个或多个后端角色以及零个或多个自定义属性。
Role mapping 角色映射	用户在成功进行身份验证后会担任角色。角色映射，就是将角色映射到用户（或后端角色）。例如，kibana_user（角色）到 jdoe（用户）的映射意味着 John Doe 在获得 kibana_user 身份验证后获得了所有权限。同样，all_access（角色）到 admin（后端角色）的映射意味着具有后端角色 admin（来自 LDAP / Active Directory 服务器）的任何用户都获得了 all_access 身份验证后的所有权限。您可以将每个角色映射到许多用户和/或后端角色。

身份验证

安全模式开启后，需要使用您创建集群时设置的用户名和密码进行登录操作，登录集群后才能进行其他操作。

图1-1 身份验证登录

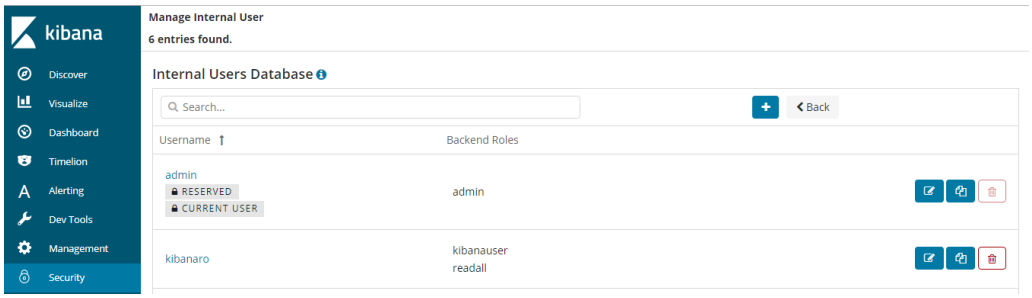


授权

在 kibana 使用界面您可以在 Security 菜单中控制用户在 ES 集群中的权限，并且可以针对集群、索引、文档和字段四个级别进行分层权限设置。

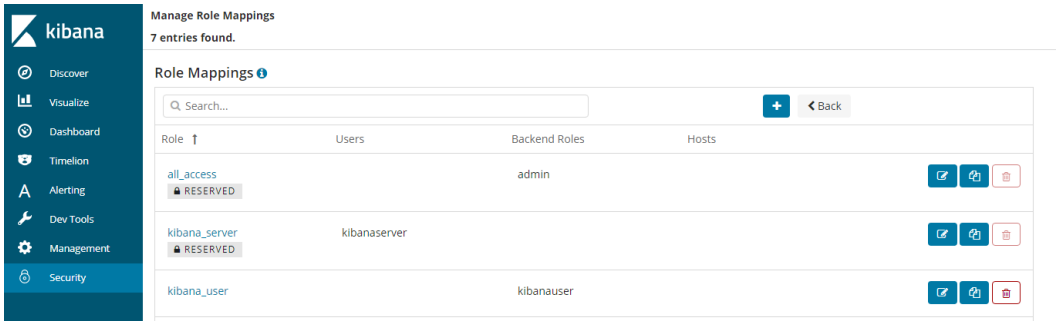
您可以增删用户，并将用户映射到角色类型设置权限。

图1-2 用户设置



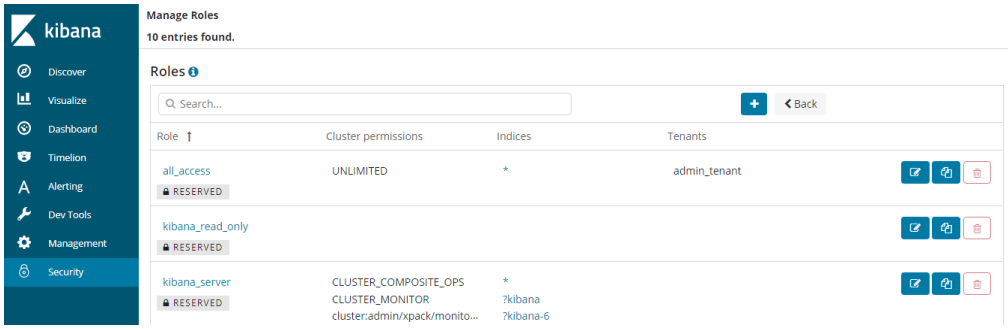
可以使用角色映射配置角色成员，可使用用户名、后端角色和主机名将用户分配给角色。

图1-3 角色映射



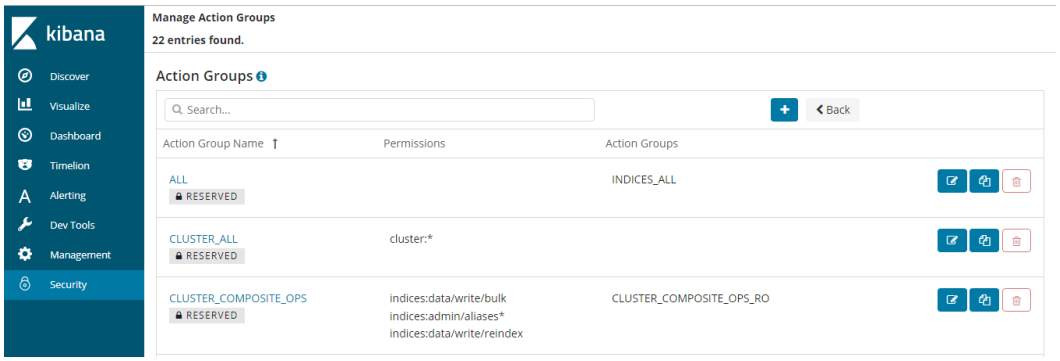
可以设置每种角色的集群访问权限、索引和文档访问权限以及 kibana 租户。

图1-4 角色权限设置



可以设置操作组，并将操作组分配给角色配置角色对索引和文档类型的访问权限。

图1-5 操作组设置

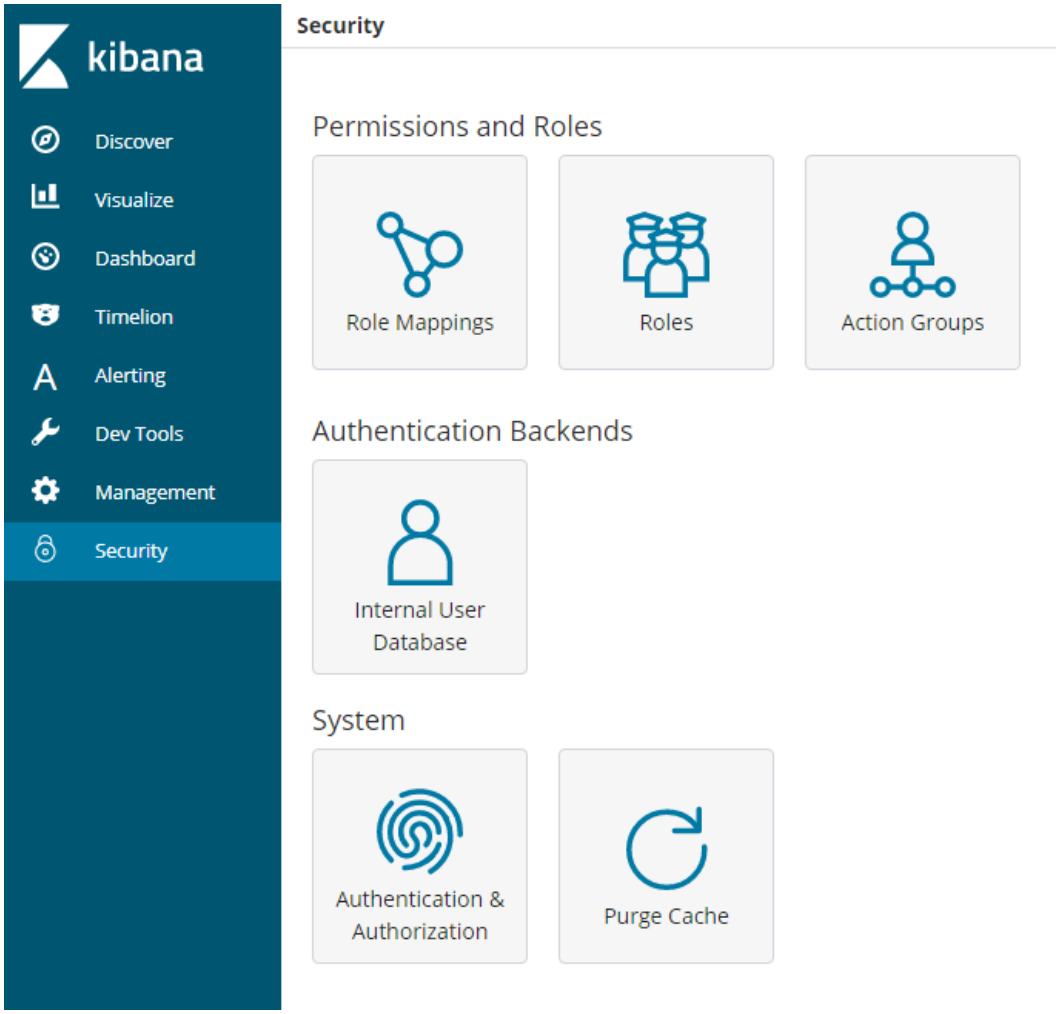


可以查询集群当前设置的身份验证及授权模块的参数。使用 securityadmin 命令行可修改相关配置。

图1-6 集群参数查看

最后，安全模块还为您提供了清除所有安全缓存的功能。

图1-7 安全缓存清除



加密

当您使用节点对节点传输或者 HTTP 传输方式传输关键数据时，可以借助 SSL/TLS 加密，对数据安全进行保护。

以上功能除了可以使用 Kibana 可视化界面操作，还可以使用.yml 文件（不推荐）和 REST API 操作，更多安全模式相关内容可以查看安全模式官方介绍。

重置密码

当您想要更换安全模式集群的登录密码，或者忘记密码时，可以对集群密码进行重置。

1. 在集群管理控制页面，选择需要重置密码的集群，单击集群名称，进入集群基本信息页面。
2. 在“基本信息”页面，单击“重置密码”后的“重置”，重置新密码。

说明

- 可输入的字符串长度为 8-32 个字符。
- 密码至少包含大写字母，小写字母，数字和特殊字符中的三类，不能包含空格。
- 不能与用户名或倒序的用户名相同。
- 建议定期修改密码。

1.7 跨 AZ 高可用性介绍

为了防止数据丢失并在服务中断时最大限度地减少集群停机时间，您可以在创建集群时，选择同一个区域中的两个或三个可用区，系统将在选择的可用区之间分配节点。

关于节点数的选择

当创建集群，可用区选择了两个或者三个时，云搜索服务自动为您开启跨 AZ 高可用特性，节点将会均衡的分布在不同的 AZ。

关于节点的数量分布您可以参考下表：

集群节点个数	单 AZ	两 AZ		三 AZ		
	AZ1	AZ1	AZ2	AZ1	AZ2	AZ3
1 节点	1	不支持		不支持		
2 节点	2	1	1	不支持		
3 节点	3	2	1	1	1	1
4 节点	4	2	2	2	1	1
…	…	…	…	…	…	…

说明

- 云搜索服务不强制要求节点个数要为 AZ 数量的倍数。
- 创建集群时，选择的节点数量要大于等于 AZ 数量。
- 各个 AZ 之间节点数量差小于等于 1。

关于副本设置

设置副本能最大程度的利用 AZ 的高可用能力。

- 在跨两个可用区的部署中，当其中一个 AZ 不可用时，剩下的 AZ 需要继续提供服务，因此索引的副本个数至少为 1 个。由于 Elasticsearch 默认副本数为 1 个，因此如果您对读性能没有特殊要求，可以直接使用默认值。
- 在跨三个可用区部署中，为了保证其中任意一个 AZ 不可用时，剩余的 AZ 需要继续提供服务，因此索引的副本数至少要为 1 个。当然，为了提高集群的查询能力，也可以设置更多的副本。由于 Elasticsearch 默认的副本数为 1 个，因此需要用户修改 setting 配置来实现修改索引副本个数。

可以通过如下命令修改索引的副本个数，如：

curl -XPUT http://ip:9200/{index_name}/_settings -d '{"number_of_replicas":2}'

也可以通过在模板中指定所有索引的副本个数，如：

curl -XPUT http://ip:9200/ _template/templatename -d '{ "template": "*", "settings": {"number_of_replicas": 2}}'

说明

- ip：表示内网访问地址。
- number_of_replicas：修改后的索引副本个数。命令中的取值表示修改为 2 个索引副本。

关于独立 Master 节点

创建集群时，如果选择了“启用 Master 节点”，选择多个 AZ 后，Master 节点也会均匀的分布在不同的 AZ 上面。

关于可用区中断

当创建集群时，选择两个或三个 AZ，如果一个 AZ 故障，业务故障行为分析如表 1-3 所示。

表1-3 一个 AZ 故障的业务故障行为分析

选择的 AZ 数量	开启主节点个数	业务中断行为
2	0	• 如果节点个数为 2 的倍数： – 一半的数据节点故障，需要替换故障可用区中的一个节点，才能继续选择主节点。 • 如果节点数为奇数： – 故障 AZ 含多一个节点，需要替换故障可用区中一个节

选择的AZ数量	开启主节点个数	业务中断行为
		点，才能继续选择主节点。相关替换请联系技术支持。 – 故障 AZ 含少一个节点，不中断业务，能够继续选主。
2	3	有 50%机会的停机时间。当两个专用主节点分配到一个可用区中，一个主节点分配到另一个可用区中时： • 如果具有一个专用主节点的可用区遇到中断，则剩余可用区具有两个专用主节点，这两个专用主节点可以选择出主节点。 • 如果具有两个专用主节点的可用区遇到中断，剩余可用区只有一个专用主节点，无法选择出主节点，业务中断，需要联系技术支持。
3	0	当您选择 3 个可用区，节点个数为 4，三个可用区的节点分布数为 2，1，1，如果节点个数为 2 的可用区故障，那么此时业务中断，建议您选择三个可用区时避免选择 4 个节点。 一般不会出现业务中断时间。
3	3	无业务中断时间。

1.8 与其他服务之间的关系

介绍了云搜索服务与其他服务的关系。

- 虚拟私有云（Virtual Private Cloud，简称 VPC）

  云搜索服务的集群创建在虚拟私有云（VPC）的子网内，VPC 通过逻辑方式进行网络隔离，为用户的集群提供安全、隔离的网络环境。

- 弹性云服务器（Elastic Cloud Server，简称 ECS）

  云搜索服务的集群中每个节点为一台弹性云服务器（ECS）。创建集群时将自动创建弹性云服务器作为节点。

- 云硬盘（Elastic Volume Service，简称 EVS）

  云搜索服务使用云硬盘（EVS）存储索引数据。创建集群时，将自动创建云硬盘用于集群存储。

- 对象存储服务（Object Storage Service，简称 OBS）

  云搜索服务的集群快照存储在对象存储服务（OBS）的桶中。

- 统一身份认证服务（Identity and Access Management，简称 IAM）

  云搜索服务使用统一身份认证服务（IAM）进行鉴权。

- 云监控服务（Cloud Eye）

  云搜索服务使用云监控服务实时监测集群的指标信息，保障服务正常运行。云搜索服务当前支持的监控指标为磁盘使用率和集群健康状态。用户通过磁盘使用率

指标可以及时了解集群的磁盘使用情况。通过集群健康状态指标，用户可以了解集群的健康状态。

- 云审计服务（Cloud Trace Service，简称 CTS）

  云审计服务（CTS）可以记录与云搜索服务相关的操作事件，便于日后的查询、审计和回溯。

1.9 权限管理

在使用云搜索服务（Cloud Search Service）的过程中，如果需要给企业中的员工设置不同的访问权限，以达到不同员工之间的权限隔离，您可以使用统一身份认证服务（Identity and Access Management，简称 IAM）进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能，可以帮助您安全控制云服务资源的访问。

如果当前帐号已经能满足您的需求，不需要创建独立的 IAM 用户进行权限管理，您可以跳过本章节，不影响您使用 ES 服务的其它功能。

通过 IAM，您可以在帐号中给员工创建 IAM 用户，并授权控制 IAM 用户对资源的访问范围，而且无需付费即可使用，您只需要为使用中的资源进行付费。关于 IAM 的详细介绍，请参见《IAM 产品介绍》。

ES 权限

默认情况下，ES 服务管理员创建的 IAM 用户没有任何权限，需要将其加入用户组，并给用户组授予策略或角色，才能使得用户组中的用户获得对应的权限，这一过程称为**授权**。授权后，用户就可以基于被授予的权限对云服务进行操作。

ES 是项目级服务，部署时通过物理区域划分，需要在各区域对应的项目中设置策略，并且该策略仅对此项目生效，如果需要所有区域都生效，则需要在所有项目都设置策略。访问 ES 时，需要先切换至授权区域。

根据授权精细程度分为角色和策略。

- 角色：IAM 最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。该机制以服务为粒度，提供有限的服务相关角色用于授权。由于云各服务之间存在业务依赖关系，因此给用户授予角色时，可能需要一并授予依赖的其他角色，才能正确完成业务。角色并不能满足用户对精细化授权的要求，无法完全达到企业对权限最小化的安全管控要求。

- 策略：IAM 最新提供的一种细粒度授权的能力，可以精确到具体服务的操作、资源以及请求条件等。基于策略的授权是一种更加灵活的授权方式，能够满足企业对权限最小化的安全管控要求。例如：ES 服务管理员能够控制 IAM 用户仅能对某一类云服务器资源进行指定的管理操作。多数细粒度策略以 API 接口为粒度进行权限拆分，ES 服务支持的 API 授权项请参见权限策略和授权。

如表 1-4 所示，包括了 ES 的所有系统权限。

- 对于 Elasticsearch Administrator，由于各服务之间存在业务交互关系，ES 的角色依赖其他服务的角色实现功能。因此给用户授予 ES 的角色时，需要同时授予依赖的角色，ES 的权限才能生效。

- 对于 ES FullAccess 和 ES ReadOnlyAccess，可使用这些策略来控制对云服务资源的访问范围。例如，您的员工中有负责软件开发的人员，您希望软件开发人员拥有 ES 的使用权限，但是不希望软件开发人员拥有删除 ES 等高危操作的权限，那么您可以使用 IAM 为开发人员创建 IAM 用户，通过授予仅能使用 ES 但不允许删除 ES 的权限，控制对 ES 资源的使用范围。

表1-4 ES 系统权限

系统角色/策略名称	类别	权限描述	依赖关系
Elasticsearch Administrator	系统角色	ES 服务的所有执行权限。 该角色有依赖，需要在同项目中勾选依赖的 Tenant Guest 和 Server Administrator 角色。	• Tenant Guest：全局级角色，在全局项目中勾选。 • Server Administrator：项目级角色，在同项目中勾选。
ES FullAccess	系统策略	基于策略授权的 ES 服务的所有权限，拥有该权限的用户可以完成基于策略授权的 ES 服务的所有执行权限。	无
ES ReadOnlyAccess	系统策略	ES 服务的只读权限，拥有该权限的用户仅能查看 ES 服务数据。	无

如表 1-5 所示列出了 ES 常用操作与系统权限的授权关系，您可以参照该表选择合适的系统权限。

表1-5 常用操作与系统权限的关系

操作	ES FullAccess	ES ReadOnlyAccess	Elasticsearch Administrator	备注
创建集群	√	x	√	-
查询集群列表	√	√	√	-
查询集群详情	√	√	√	-
删除集群	√	x	√	-
重启集群	√	x	√	-
扩容集群	√	x	√	-

操作	ES FullAccess	ES ReadOnlyAccess	Elasticsearch Administrator	备注
扩容实例的数量和存储容量	√	x	√	-
查询指定集群的标签	√	√	√	-
查询所有标签	√	√	√	-
加载自定义词库	√	x	√	依赖 OBS 和 IAM 权限
查询自定义词库状态	√	√	√	-
删除自定义词库	√	x	√	-
自动设置集群快照的基础配置	√	x	√	依赖 OBS 和 IAM 权限
修改集群快照的基础配置	√	x	√	依赖 OBS 和 IAM 权限
设置自动创建快照策略	√	x	√	-
查询集群的自动创建快照策略	√	√	√	-
手动创建快照	√	x	√	-
查询快照列表	√	√	√	-
恢复快照	√	x	√	-
删除快照	√	x	√	-
停用快照功能	√	x	√	-
更改规格	√	x	√	-
缩容集群	√	x	√	-

1.10 约束与限制

集群和节点限制

下表显示了云搜索服务的集群和节点的限制。

表1-6 集群和节点限制

集群和节点	限制
每个集群的最大节点数（节点数量）	默认值 32，最大支持 200 个节点，如果需要更改默认值，请联系技术支持。
每个集群的最小节点数（节点数量）	1

浏览器限制

- 访问云搜索服务管理控制台，建议使用如下版本浏览器
  - Google Chrome：36.0 及更高版本
  - Mozilla FireFox：35.0 及更高版本
  - Internet Explorer：9.0 及更高版本

    当使用 Internet Explorer 9.0 时可能无法登录云搜索服务管理控制台，原因是某些 Windows 系统例如 Win7 旗舰版，默认禁止 Administrator 用户，Internet Explorer 在安装时自动选择其他用户如 System 用户安装，从而导致 Internet Explorer 无法打开登录页面。请使用管理员身份重新安装 Internet Explorer 9.0 或更高版本（建议），或尝试使用管理员身份运行 Internet Explorer 9.0。
- 访问云搜索服务中 Kibana，建议使用如下版本浏览器
  - Google Chrome：36.0 及更高版本
  - Mozilla FireFox：35.0 及更高版本
  - Internet Explorer：11.0 及更高版本，不支持 IE9

2 入门

2.1 快速开始使用 Elasticsearch 搜索引擎

关于云搜索服务的概念、优势、功能和应用场景等，请参见 1.1 什么是云搜索服务。

本章节提供了一个简单示例，详情如场景描述所示。您可以参考此场景示例数据，使用云搜索服务的 Elasticsearch 搜索引擎搜索数据，基本操作流程如下所示：

- 步骤 1：创建集群
- 步骤 2：导入数据
- 步骤 3：搜索数据
- 步骤 4：删除集群

场景描述

某女装品牌在网上经营电商业务，其以前是使用传统数据库来为用户提供商品搜索功能，但随着用户数量和业务的增长，使用传统数据库的弊端愈来愈明显。主要问题表现为：响应速度慢、准确性低。为了改善用户体验从而避免用户流失，该电商网站开始使用 Elasticsearch 搜索引擎来为用户提供商品搜索功能，使用了一段时间后，不仅解决了之前使用传统数据库产生的问题，而且实现了用户数量的增长。

本章节将介绍如何使用 Elasticsearch 搜索引擎为用户提供搜索功能。

假设该电商网站经营商品的数据如下所示：

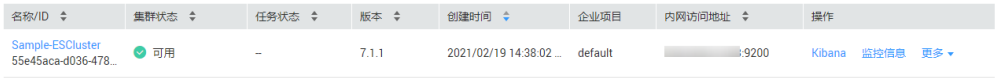
```
{
"products":[
{"productName":"2017 秋装新款文艺衬衫女装","size":"L"}
{"productName":"2017 秋装新款文艺衬衫女装","size":"M"}
{"productName":"2017 秋装新款文艺衬衫女装","size":"S"}
{"productName":"2018 春装新款牛仔裤女装","size":"M"}
{"productName":"2018 春装新款牛仔裤女装","size":"S"}
{"productName":"2017 春装新款休闲裤女装","size":"L"}
{"productName":"2017 春装新款休闲裤女装","size":"S"}
]
}
```

步骤 1：创建集群

在开始搜索数据之前，您需要创建一个集群，其搜索引擎为 Elasticsearch。例如，您可以创建一个名称为“Sample-ESCluster”的集群。此集群仅用于入门指导使用，建议选用“节点规格”为“ess.spec-4u8g”，“节点存储”为“高 I/O”，“节点存储容量”为“40GB”。详细操作步骤请参见 4.2 创建 Elasticsearch 类型集群（非安全模式）。

集群创建完成后，在集群列表查看已创建的集群，集群状态为“可用”表示集群创建成功。如图 2-1 所示。

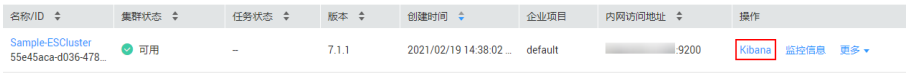
图2-1 创建集群



步骤 2：导入数据

云搜索服务支持通过云数据迁移（简称 CDM）、数据接入服务（简称 DIS）、Logstash、Kibana 或 API 将数据导入到 Elasticsearch。其中 Kibana 是 Elasticsearch 的图形化界面，便于交互验证，因此，这里以 Kibana 为例介绍将数据导入到 Elasticsearch 的操作流程。

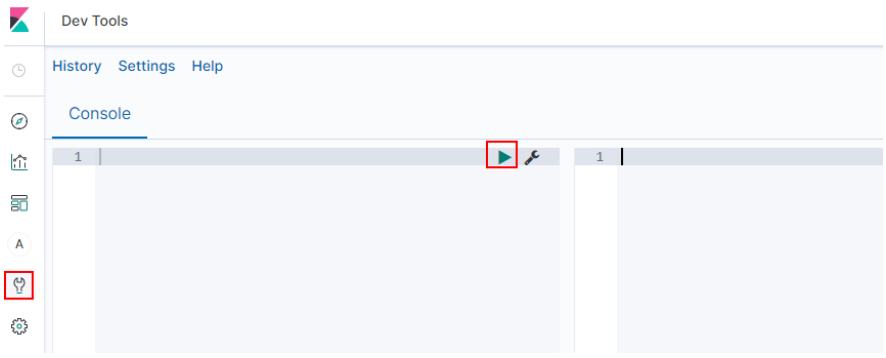
1. 在云搜索服务的“集群管理”页面上，单击集群“操作”列的“Kibana”访问集群。



2. 在 Kibana 的左侧导航中选择“Dev Tools”，单击“Get to work”，进入 Console 界面，如图 2-2 所示。

   Console 左侧区域为输入框，右侧为结果输出区域，  为执行命令按钮。

   图2-2 Console 界面



3. 在 Console 界面，执行如下命令创建索引“my_store”。

   （7.x 之前版本）

```
PUT /my_store
{
  "settings": {
    "number_of_shards": 1
  },
  "mappings": {
    "products": {
      "properties": {
        "productName": {
          "type": "text",
          "analyzer": "ik_smart"
        },
        "size": {
          "type": "keyword"
        }
      }
    }
  }
}
```

（7.x 之后版本）

```
PUT /my_store
{
  "settings": {
    "number_of_shards": 1
  },
  "mappings": {
        "properties": {
      "productName": {
        "type": "text",
        "analyzer": "ik smart"
      },
      "size": {
        "type": "keyword"
      }
    }
  }
}
```

返回结果如下所示。

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : "my_store"
}
```

4. 在 Console 界面，执行如下命令，将数据导入到“my_store”索引中。

（7.x 之前版本）

```
POST /my_store/products/_bulk
{"index":{}}
{"productName":"2017 秋装新款文艺衬衫女装","size":"L"}
{"index":{}}
{"productName":"2017 秋装新款文艺衬衫女装","size":"M"}
{"index":{}}
{"productName":"2017 秋装新款文艺衬衫女装","size":"S"}
{"index":{}}
```

```
{"productName":"2018 春装新款牛仔裤女装","size":"M"}
{"index":{}}
{"productName":"2018 春装新款牛仔裤女装","size":"S"}
{"index":{}}
{"productName":"2017 春装新款休闲裤女装","size":"L"}
{"index":{}}
{"productName":"2017 春装新款休闲裤女装","size":"S"}
```

（7.x 之后版本）

```
POST /my_store/_doc/_bulk
{"index":{}}
{"productName":"2017 秋装新款文艺衬衫女装","size":"L"}
{"index":{}}
{"productName":"2017 秋装新款文艺衬衫女装","size":"M"}
{"index":{}}
{"productName":"2017 秋装新款文艺衬衫女装","size":"S"}
{"index":{}}
{"productName":"2018 春装新款牛仔裤女装","size":"M"}
{"index":{}}
{"productName":"2018 春装新款牛仔裤女装","size":"S"}
{"index":{}}
{"productName":"2017 春装新款休闲裤女装","size":"L"}
{"index":{}}
{"productName":"2017 春装新款休闲裤女装","size":"S"}
```

当返回结果信息中“errors”字段的值为“false”时，表示导入数据成功。

步骤 3：搜索数据

- **全文检索**

假设用户进入该电商网站，她想要查找名称包含“春装牛仔裤”的商品信息，可以搜索“春装牛仔裤”。这里使用 Kibana 演示用户搜索数据在后台的执行命令和返回结果。

执行命令如下所示。

（7.x 之前版本）

```
GET /my_store/products/_search
{
  "query": {"match": {
    "productName": "春装牛仔裤"
  }}
}
```

（7.x 之后版本）

```
GET /my store/ search
{
  "query": {"match": {
    "productName": "春装牛仔裤"
  }}
}
```

返回结果如下所示。

```
{
  "took" : 3,
  "timed_out" : false,
```

```
"_shards" : {
  "total" : 1,
  "successful" : 1,
  "skipped" : 0,
  "failed" : 0
},
"hits" : {
  "total" : {
    "value" : 4,
    "relation" : "eq"
  },
  "max_score" : 1.7965372,
  "hits" : [
    {
      "_index" : "my_store",
      "_type" : "_doc",
      "_id" : "9xf6VHIBfClt6SDjw7H5",
      "_score" : 1.7965372,
      "_source" : {
        "productName" : "2018 春装新款牛仔裤女装",
        "size" : "M"
      }
    },
    {
      "_index" : "my_store",
      "_type" : "_doc",
      "_id" : "-Bf6VHIBfClt6SDjw7H5",
      "_score" : 1.7965372,
      "_source" : {
        "productName" : "2018 春装新款牛仔裤女装",
        "size" : "S"
      }
    },
    {
      "_index" : "my_store",
      "_type" : "_doc",
      "_id" : "-Rf6VHIBfClt6SDjw7H5",
      "_score" : 0.5945667,
      "_source" : {
        "productName" : "2017 春装新款休闲裤女装",
        "size" : "L"
      }
    },
    {
      "_index" : "my_store",
      "_type" : "_doc",
      "_id" : "-hf6VHIBfClt6SDjw7H5",
      "_score" : 0.5945667,
      "_source" : {
        "productName" : "2017 春装新款休闲裤女装",
        "size" : "S"
      }
    }
  ]
}
```

```
}
```

- Elasticsearch 支持分词，上面执行命令会将“春装牛仔裤”分词为“春装”和“牛仔裤”。
- Elasticsearch 支持全文检索，上面执行命令会在所有商品信息中搜索包含“春装”或“牛仔裤”的商品信息。
- Elasticsearch 与传统数据库不同，它能借助倒排索引在毫秒级返回结果。
- Elasticsearch 支持评分排序，在上面返回结果中，前两条商品信息中同时出现了“春装”和“牛仔裤”，后两条商品信息中只出现了“春装”，所以前两条比后两条与检索关键词的匹配度更高，分数更高，排序也更靠前。

- **聚合结果显示**

该电商网站可以提供聚合结果显示功能，例如： 对“春装”对应的产品按照尺码分类，统计不同尺码的数量。这里使用 Kibana 演示聚合结果显示功能在后台的执行命令和返回结果。

执行命令如下所示。

（7.x 之前版本）

```
GET /my_store/products/_search
{
"query": {
"match": { "productName": "春装" }
},
"size": 0,
"aggs": {
"sizes": {
"terms": { "field": "size" }
}
}
}
```

（7.x 之后版本）

```
GET /my_store/_search
{
"query": {
"match": { "productName": "春装" }
},
"size": 0,
"aggs": {
"sizes": {
"terms": { "field": "size" }
}
}
}
```

返回结果如下所示。

（7.x 之前版本）

```
{
  "took" : 31,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
```

```
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : 4,
    "max_score" : 0.0,
    "hits" : [ ]
  },
  "aggregations" : {
    "sizes" : {
      "doc_count_error_upper_bound" : 0,
      "sum_other_doc_count" : 0,
      "buckets" : [
        {
          "key" : "S",
          "doc_count" : 2
        },
        {
          "key" : "L",
          "doc_count" : 1
        },
        {
          "key" : "M",
          "doc_count" : 1
        }
      ]
    }
  }
}
```


（7.x 之后版本 ）


```
{
  "took" : 3,
  "timed out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : null,
    "hits" : [ ]
  },
  "aggregations" : {
    "sizes" : {
      "doc_count_error_upper_bound" : 0,
      "sum_other_doc_count" : 0,
      "buckets" : [
        {
          "key" : "S",
          "doc_count" : 2
```

```
      },
      {
        "key" : "L",
        "doc_count" : 1
      },
      {
        "key" : "M",
        "doc_count" : 1
      }
    ]
  }
 }
}
```

步骤 4：删除集群

当您已完全了解 Elasticsearch 搜索引擎的使用流程和方法后，您可以参考如下步骤，删除示例集群以及示例数据，避免造成资源浪费。

由于集群删除后，数据无法恢复，请谨慎操作。

1. 登录云搜索服务管理控制台。在左侧菜单栏选择“集群管理”。
2. 进入集群管理页面，选中“Sample-ESCluster”集群所在行，在操作列单击“更多”>“删除”。
3. 在弹出的确认对话框中，单击“确定”完成操作。

3 权限管理

3.1 创建用户并授权使用 ES

本章节介绍创建 ES 用户操作，将 ES 服务的策略授予用户组，并将用户添加至用户组中（一个用户组下面的用户具有相同的权限），从而使用户拥有对应的 ES 权限，操作流程如图 3-1 所示。

ES 具有两种类型用户权限（ES 管理员权限和只读权限），在权限规划的时候请规划这两种类型的用户组。

前提条件

给用户组授权之前，请您了解用户组可以添加的 ES 系统策略，请参见 1.9 权限管理权限管理。

示例流程

图3-1 给用户授权 ES 权限流程

1. IAM 用户指南中创建用户组授权
   在 IAM 控制台创建用户组，并授予云搜索服务权限。
2. IAM 用户指南中创建用户并加入用户组
   在 IAM 控制台创建用户，并将其加入 1.创建用户组并授权中创建的用户组。
3. 用户登录并验证权限
   新创建的用户登录控制台，验证云搜索服务的权限。

3.2 ES 自定义策略

如果系统预置的 ES 权限，不满足您的授权要求，可以创建自定义策略。自定义策略中可以添加的授权项（Action）请参考云搜索服务 API 参考中的权限策略和授权项。

目前支持以下两种方式创建自定义策略：

- 可视化视图创建自定义策略：无需了解策略语法，按可视化视图导航栏选择云服务、操作、资源、条件等策略内容，可自动生成策略。
- JSON 视图创建自定义策略：可以在选择策略模板后，根据具体需求编辑策略内容；也可以直接在编辑框内编写 JSON 格式的策略内容。
- 具体创建步骤请参见：IAM 用户指南中的创建自定义策略。本章为您介绍常用的 ES 自定义策略样例。

ES 系统策略样例

示例 1：授权用户 ES FullAccess 权限，即给用户配置 ES 的所有权限。

开启 ES FullAccess 权限，需要依赖 OBS 和 IAM 权限，除了配 ES FullAccess 还要加上 IAM ReadOnlyAccess 和 OBS 的所有权限。如果用户需要查看集群监控信息，则需要依赖 CES 的只读权限。如果用户需要使用终端节服务，除了配置 VPCEndpoint Administrator 权限，还需要同时配置 VPC Administrator、Server Administrator 以及 DNS Administrator 权限。如果用户需要使用标签功能，需要配置 TMS Administrator 权限。

说明

如果是子账户，需要同时设置 GetBucketStoragePolicy、GetBucketLocation、ListBucket 权限，才能看到 OBS 桶。

1. 授权用户 ES FullAccess 权限。

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "ES:*:*",
                "vpc:securityGroups:get",
                "vpc:securityGroups:create",
                "vpc:securityGroups:delete",
                "vpc:securityGroupRules:get",
                "vpc:securityGroupRules:create",
                "vpc:securityGroupRules:delete",
                "vpc:vpcs:list",
                "vpc:privateIps:list",
                "vpc:ports:get",
                "vpc:ports:create",
                "vpc:ports:update",
                "vpc:ports:delete",
                "vpc:quotas:list",
                "vpc:subnets:get",
                "ecs:cloudServerFlavors:get",
                "ecs:serverInterfaces:use",
                "ecs:cloudServers:addNics",
                "ecs:quotas:get",
                "evs:types:get",
                "evs:quotas:get"
            ],
            "Effect": "Allow"
        }
    ]
}
```

2. 授权用户 IAM ReadOnlyAccess 自定义策略。

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "iam:*:get*",
```

```
            "iam:*:list*",
            "iam:*:check*"
        ],
        "Effect": "Allow"
    }
  ]
}
```

3. 授权用户 OBS 的所有权限。

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "OBS:*:*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

4. （可选）授权用户查看集群监控信息权限。

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "ces:*:get*",
        "ces:*:list*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

5. （可选）授权用户 VPCEndpoint Administrator 权限。

```
{
  "Version": "1.0",
  "Statement": [
    {
      "Action": [
        "VPCEP:endpoint_services:*"
      ],
      "Effect": "Allow"
    }
  ],
  "Depends": [
    {
      "catalog": "BASE",
      "display_name": "Server Administrator"
    },
    {
      "catalog": "VPC",
      "display_name": "VPC Administrator"
    },
    {
```

```
            "catalog": "DNS",
            "display_name": "DNS Administrator"
        }
    ]
}
```

6. （可选）授权 VPC Administrator 权限。

```
{
    "Version": "1.0",
    "Statement": [
        {
            "Action": [
                "vpc:vpcs:*",
                "vpc:routers:*",
                "vpc:networks:*",
                "vpc:subnets:*",
                "vpc:ports:*",
                "vpc:privateIps:*",
                "vpc:peerings:*",
                "vpc:routes:*",
                "vpc:lbaas:*",
                "vpc:vpns:*",
                "ecs:*:get",
                "ecs:*:list",
                "elb:*:get",
                "elb:*:list"
            ],
            "Effect": "Allow"
        }
    ]
}
```

7. （可选）授权 Server Administrator 权限。

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "ecs:*:*",
                "evs:*:get",
                "evs:*:list",
                "evs:volumes:create",
                "evs:volumes:delete",
                "evs:volumes:attach",
                "evs:volumes:detach",
                "evs:volumes:manage",
                "evs:volumes:update",
                "evs:volumes:uploadImage",
                "evs:snapshots:create",
                "vpc:*:get",
                "vpc:*:list",
                "vpc:networks:create",
                "vpc:networks:update",
                "vpc:subnets:update",
                "vpc:subnets:create",
                "vpc:routers:get",
```

```
            "vpc:routers:update",
            "vpc:ports:*",
            "vpc:privateIps:*",
            "vpc:securityGroups:*",
            "vpc:securityGroupRules:*",
            "vpc:floatingIps:*",
            "vpc:publicIps:*",
            "vpc:bandwidths:*",
            "vpc:firewalls:*",
            "ims:images:create",
            "ims:images:delete",
            "ims:images:get",
            "ims:images:list",
            "ims:images:update",
            "ims:images:upload"
        ],
        "Effect": "Allow"
    }
  ],
  "Depends": [
    {
        "catalog": "BASE",
        "display_name": "Tenant Guest"
    },
    {
        "catalog": "BASE",
        "display_name": "BSS Administrator"
    },
    {
        "catalog": "BASE",
        "display_name": "VPC Administrator"
    },
    {
        "catalog": "BASE",
        "display_name": "IMS Administrator"
    }
  ]
}
```

8. （可选）授权 DNS Administrator 权限。

```
{
  "Version": "1.0",
  "Statement": [
    {
        "Action": [
            "DNS:Zone:*",
            "DNS:RecordSet:*",
            "DNS:PTRRecord:*"
        ],
        "Effect": "Allow"
    }
  ],
  "Depends": [
    {
        "catalog": "BASE",
        "display_name": "Tenant Guest"
```

```
        },
        {
            "catalog": "VPC",
            "display_name": "VPC Administrator"
        }
    ]
}
```

9. （可选）授权 TMS Administrator 权限。

```
{
    "Version": "1.0",
    "Statement": [
        {
            "Action": [
                "TMS:predefine_tag:*",
                "TMS:resource_tag:*"
            ],
            "Effect": "Allow"
        }
    ],
    "Depends": [
        {
            "catalog": "BASE",
            "display_name": "Tenant Guest"
        },
        {
            "catalog": "BASE",
            "display name": "Server Administrator"
        },
        {
            "catalog": "IMS",
            "display name": "IMS Administrator"
        },
        {
            "catalog": "Auto Scaling",
            "display_name": "AutoScaling Administrator"
        },
        {
            "catalog": "VPC",
            "display_name": "VPC Administrator"
        },
        {
            "catalog": "VBS",
            "display_name": "VBS Administrator"
        },
        {
            "catalog": "OBS",
            "display_name": "Tenant Administrator"
        },
        {
            "catalog": "OBS",
            "display_name": "Tenant Guest"
        }
    ]
}
```

说明

如果用户账号开通了企业项目：

- 当该账号配置 ES FullAccess 权限时，即使给单个企业项目只配了 ES ReadOnlyAccess 权限，但所有企业项目都拥有 ES FullAccess 权限。
- 如果给单个企业项目开通了 ES FullAccess 权限，则该企业项目下的所有子用户都可以拥有该权限。如：企业项目 default 配了 ES FullAccess 权限，那么子用户可以读到 default 企业项目下的集群，且进行读写操作。

示例 2：授权用户 ES ReadOnlyAccess 权限，即给用户配置 ES 的只读权限。如果用户需要查看集群监控信息，则需要依赖 CES 的只读权限。

1. 授权用户 ES ReadOnlyAccess 权限。

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "ES:*:get*",
                "ES:*:list*",
                "vpc:securityGroups:get",
                "vpc:securityGroupRules:get",
                "vpc:vpcs:list",
                "vpc:privateIps:list",
                "vpc:ports:get",
                "vpc:quotas:list",
                "vpc:subnets:get",
                "ecs:cloudServerFlavors:get",
                "ecs:quotas:get",
                "evs:types:get",
                "evs:quotas:get"
            ],
            "Effect": "Allow"
        }
    ]
}
```

2. （可选）授权用户查看集群监控信息权限。

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "ces:*:get*",
                "ces:*:list*"
            ],
            "Effect": "Allow"
        }
    ]
}
```

说明

如果用户账号开通了企业项目：

- 在统一认证服务中给该账号开通了 ES ReadOnlyAccess 权限，但是给某个企业项目配置了 ES FullAccess 权限，那么子用户可以读到所有企业项目下的集群，但是只对开通 ES FullAccess 权限下的集群进行写操作。例如，给企业项目 default 配置了 ES FullAccess 权限，那么子用户可以读到所有企业项目下的集群，但是只对 default 下的集群进行写操作。
- 在统一认证服务中给该账号开通了 ES ReadOnlyAccess 权限，但是没有给单个企业项目授权，则子账户只能读取该项目下的集群，不能进行写操作。例如，给企业项目 default 配了 ES ReadOnlyAccess，那么子用户可以读到 default 企业项目下的集群，不可进行写操作。

ES 自定义策略样例

示例 1：授权用户创建集群。

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Action": [
                "ES:cluster:create",
                "vpc:securityGroups:get",
                "vpc:securityGroups:create",
                "vpc:securityGroups:delete",
                "vpc:securityGroupRules:get",
                "vpc:securityGroupRules:create",
                "vpc:securityGroupRules:delete",
                "vpc:vpcs:list",
                "vpc:privateIps:list",
                "vpc:ports:get",
                "vpc:ports:create",
                "vpc:ports:update",
                "vpc:ports:delete",
                "vpc:quotas:list",
                "vpc:subnets:get",
                "ecs:cloudServerFlavors:get",
                "ecs:serverInterfaces:use",
                "ecs:cloudServers:addNics",
                "ecs:quotas:get",
                "evs:types:get",
                "evs:quotas:get"
            ],
            "Effect": "Allow"
        }
    ]
}
```

示例 2：拒绝用户删除集群。

拒绝策略需要同时配合其他策略使用，否则没有实际作用。用户被授予的策略中，一个授权项的作用如果同时存在 Allow 和 Deny，则遵循 **Deny 优先原则**。

如果您给用户授予 ES Admin 的系统策略，但不希望用户拥有 ES admin 中定义的删除云服务器权限，您可以创建一条拒绝删除云服务的自定义策略，然后同时将 ES Admin 和拒绝策略授予用户，根据 Deny 优先原则，则用户可以对 ES 执行除了删除集群外的所有操作。拒绝策略示例如下：

```
{
     "Version": "1.1",
     "Statement": [
          {
           "Effect": "Deny",
                "Action": [
                     "ES:cluster:delete"
                ]
          }
     ]
}
```

示例 3：多个授权项策略。

一个自定义策略中可以包含多个授权项，且除了可以包含本服务的授权项外，还可以包含其他服务的授权项，可以包含的其他服务必须跟本服务同属性，即都是项目级服务或都是全局级服务。多个授权语句策略描述如下：

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Action": [
            "ecs:cloudServers:resize",
            "ecs:cloudServers:delete",
            "ecs:cloudServers:delete",
            "ES:cluster:restart",
            "ES:*:get*",
            "ES:*:list*"
         ],
         "Effect": "Allow"
      }
   ]
}
```

4 创建并接入集群

4.1 创建 Elasticsearch 类型集群（安全模式）

在开始使用云搜索服务时，您必须创建一个集群。云搜索服务 6.5.4 及之后的版本支持开启安全模式，安全模式开启之后，不能更改。

说明

- 安全模式只能在创建集群时开启。集群创建成功后，不支持开启安全模式。
- 公网访问和 Kibana 公网访问需要开启安全模式才能使用。

操作步骤

1. 登录云搜索服务管理控制台。
2. 在“总览”或者“集群管理”页面，单击“创建集群”，进入“创建集群”页面。
3. 选择“当前区域”和“可用区”。

表4-1 区域和可用区参数说明

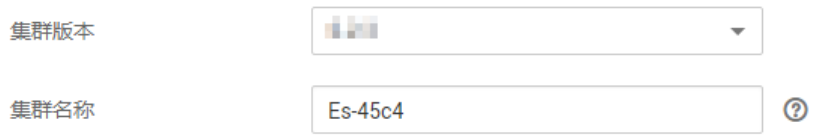
参数	说明
当前区域	集群工作区域在右侧下拉框中选择。
可用区	选择集群工作区域下关联的可用区。详细描述可参考 14.2 什么是区域和可用区。 云搜索服务支持配置多个“可用区”，详细请参考 1.7 跨 AZ 高可用性介绍。

4. 指定集群基本信息，选择“集群版本”，并输入“集群名称”。

表4-2 基本参数说明

参数	说明
集群版本	当前支持 6.5.4、7.1.1、7.6.2、7.9.3。
集群名称	自定义的集群名称，可输入的字符范围为 4～32 个字符，只能包含数字、字母、中划线和下划线，且必须以字母开头。 说明 当集群创建成功后，您可以根据需求修改集群名称。单击需要修改的集群名称，进入集群基本信息页面，单击“集群名称”后面的，修改完成后，单击，进行保存。如果需要取消修改，可单击进行取消。

图4-1 基本信息配置



5. 指定集群的主机规格相关参数。

表4-3 参数说明

参数	说明
节点数量	集群中的节点个数。 • 如果未启用 Master 节点和 Client 节点时，此参数指定的节点将被作为 Master 节点和 Client 节点，同时具备集群管理、存储数据、提供接入集群和分析数据的服务。此时，为保证集群中数据的稳定性，建议设置节点数量大于等于 3 个。 • 如果启用 Master 节点，且未启用 Client 节点，此参数指定的节点将用于存储数据并提供 Client 节点功能。 • 如果已启用 Master 节点和 Client 节点，此参数指定的节点将仅用于存储数据。 • 如果启用 Client 节点，且未启用 Master 节点，此参数指定的节点将用于存储数据并提供 Master 节点功能。
节点规格	集群中的节点规格。您可以根据需求，选择对应的规格。
节点存储	当前支持三种存储类型，普通 I/O、高 I/O、超高 I/O。
节点存储容量	存储空间大小，其取值范围与节点规格关联，不同的规格允许

参数	说明
	的取值范围不同。
启用 Master 节点	Master 节点用于管理集群中的所有节点。当需要存储和分析的数据量大，所需节点数量大于 20 个节点时，建议启用 Master 节点，保证集群的稳定性。反之，建议购买节点同时作为 Master 和 Client 节点即可，即仅设置集群的“节点数量”参数。 启用 Master 节点后，在下方选择对应的“节点规格”、“节点数量”和“节点存储”。“节点数量”必须是大于 3 的奇数，最多设置 9 个节点。其中“节点存储”的存储容量为固定值，存储类型可以根据实际情况选择，默认为存储容量为 40GB 的高 I/O 磁盘。 **说明** **只有开通大集群权限的用户才能配置“启用 Master 节点”和“启用 Client 节点”参数。**
启用 Client 节点	Client 节点用于提供客户端接入集群和分析数据的服务。当需要存储和分析的数据量大，所需节点数量大于 20 个节点时，建议启用 Client 节点，保证集群的稳定性。反之，建议购买节点同时作为 Master 和 Client 节点即可，即仅设置集群的“节点数量”参数。 启用 Client 节点后，在下方选择对应的“节点规格”、“节点数量”和“节点存储”。“节点数量”可设置为 1~32 任意数值。其中“节点存储”的存储容量为固定值，存储类型可以根据实际情况选择，默认为存储容量为 40GB 的高 I/O 磁盘。
启用冷数据节点	冷数据节点用于存放对于历史数据要求分钟级别的返回。当用户对历史数据返回时间要求不是很高的话，可以将这部分数据存储在冷数据节点上，从而降低成本。 冷数据节点为可选节点，支持的节点个数为 1-32 个。 开启冷数据节点之后，云搜索服务将会自动的给相关节点打上冷热标签，相关的集群参数，配置详情请参见 https://www.elastic.co/guide/en/elasticsearch/reference/master/allocation-awareness.html。

图4-2 设置主机规格

6. 指定集群的网络规格相关参数。

表4-4 参数说明

参数	说明
虚拟私有云	VPC 即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。 选择创建集群需要的 VPC，单击“查看虚拟私有云”进入 VPC 服务查看已创建的 VPC 名称和 ID。如果没有 VPC，需要创建一个新的 VPC。 说明 此处您选择的 VPC 必须包含网段（CIDR），否则集群将无法创建成功。新建的 VPC 默认包含网段（CIDR）。
子网	通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。 选择创建集群需要的子网，可进入 VPC 服务查看 VPC 下已创建的子网名称和 ID。
安全组	安全组是一个逻辑上的分组，为同一个 VPC 内具有相同安全保护需求并相互信任的弹性云服务器提供访问策略。单击“查看安全组”可了解安全组详情。 说明 请确保安全组的“端口范围/ICMP 类型”为“Any”或者包含端口 9200 的端口范围。
安全模式	在 6.5.4 以及之后版本中提供了安全模式的选择，开启之后将对集群进行通讯加密和安全认证，管理员账户名默认为 admin，需要设置管理员密码并确认密码。参考 1.6 安全模式集群简介

参数	说明
	可了解安全模式详情。
HTTPS 访问	当集群开启安全模式时，HTTPS 访问默认开启。安全集群使用 HTTPS 通信，相比非安全集群使用 HTTP 通信在读取性能上会慢很多。如果客户想要读取性能快，又想要使用安全集群所提供的用户权限隔离资源（索引、文档、字段等）的功能，则可以关闭 HTTPS 访问。关闭 HTTPS 访问后，会使用 HTTP 协议与集群通信，无法保证数据安全性，并且无法开启公网访问功能。 集群创建成功后，无法修改“HTTPS 访问”状态。
公网访问	只有开启“安全模式”的集群，才能配置公网访问。开启公网访问后，用户可以获得一个公网访问的 IP，通过这个 IP，可以在公网上面访问该集群。详细配置请参考 9.10 公网访问。
企业项目	如果您开通了企业项目，在创建云搜索服务集群时，您可以给集群绑定一个企业项目。您可以在右侧下拉框中选择当前用户下已创建的企业项目，也可以通过单击“查看项目管理”按钮，前往“企业项目管理”管理控制台，新建企业项目和查看已有的企业项目。
集群快照开关	• 基础配置 – OBS 桶：快照存储的 OBS 桶的名称。 – 备份路径：快照在 OBS 桶中的存放路径。 – IAM 委托：指当前账户授权云搜索服务访问或维护存储在 OBS 中数据。 具体详细信息，请参考《云搜索服务用户指南》中“备份与恢复索引”章节。 • 自动创建快照 默认情况下，系统打开了“自动快照创建”开关，您可以根据自己的需求设置“快照名称前缀”、“备份开始时间”和“保留时间（天）”。如果您不需要启用自动快照，可以在“自动快照创建”右侧，单击开关关闭自动创建快照功能。 – “快照名称前缀”：快照名称由快照名称前缀加上时间戳组成，例如自动生成的快照名称 snapshot-1566921603720。快照名称前缀的长度为 1～32 个字符，只能包含小写字母、数字、中划线和下划线，且必须以小写字母开头。 – “备份开始时间”：指每天自动开始备份的时间，只能指定整点时间，如 00:00、01:00，取值范围为 00:00～23:00。请在下拉框中选择备份时间。 – “保留时间（天）”：指备份的快照在 OBS 的保留时间，以天为单位，取值范围为 1～90，您可以根据自己的需求进行设置。系统在半点时刻会自动删除超过保留时间的快照。

参数	说明
	例如：自动快照创建的策略设置如图 4-5 所示，则系统会在 35 天后的 00:30 自动删除 35 天前 00:00 自动开始备份的快照。

图4-3 设置网络规格

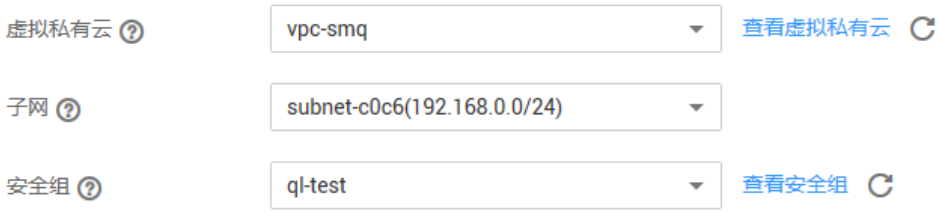


图4-4 设置网络规格（安全模式）

图4-5 设置自动快照创建的相关参数

7. 配置集群高级配置功能。

   “高级配置”：选择“默认配置”，则默认关闭“终端节点服务”、“Kibana 公网访问”和“标签”功能。如果需要配置“终端节点服务”、“标签”和“自动快照创建”功能，选择“自定义”。

表4-5 高级配置参数

参数	说明
终端节点服务	开启终端节点服务后，用户可以获得一个内网访问的域名，通过这个域名，可以在同一个 vpc 内访问该集群。详细配置请参考 9.15 终端节点服务。
Kibana 公网访问	只有开启“安全模式”的集群，才能配置 Kibana 公网访问。开启 Kibana 公网访问后，用户可以获得一个 Kibana 公网访问地址，通过这个地址，可以在公网上面访问该集群。详细配置请参考 9.16 Kibana 公网访问。
标签	为集群添加标签，可以方便用户识别和管理拥有的集群资源。此处您可以选择“标签管理服务”中已定义好的“预定义标签”，也可以自己定义标签。详细标签使用请参考 9.9 标签管理。

8. 单击“立即申请”，进入规格确认界面。
9. 规格确认完成后，单击“提交申请”开始创建集群。
10. 单击“返回集群列表”，系统将跳转到“集群管理”页面。您创建的集群将展现在集群列表中，且集群状态为“创建中”，创建成功后集群状态会变为“可用”。

    如果集群创建失败，请根据界面提示，重新创建集群。

4.2 创建 Elasticsearch 类型集群（非安全模式）

在开始使用云搜索服务时，您必须创建一个集群。

操作步骤

1. 登录云搜索服务管理控制台。
2. 在“总览”或者“集群管理”页面，单击“创建集群”，进入“创建集群”页面。
3. 选择“当前区域”和“可用区”。

表4-6 区域和可用区参数说明

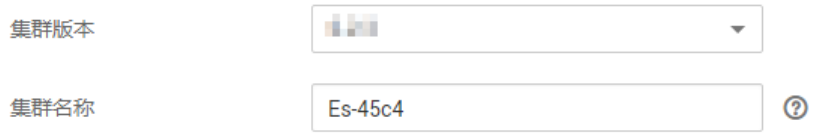
参数	说明
当前区域	集群工作区域在右侧下拉框中选择。
可用区	选择集群工作区域下关联的可用区。详细描述可参考14.2 什么是区域和可用区。 云搜索服务支持配置多个“可用区”，详细请参考1.7 跨 AZ 高可用性介绍。

4. 指定集群基本信息，选择“集群版本”，并输入“集群名称”。

表4-7 基本参数说明

参数	说明
集群版本	当前支持 5.5.1、6.2.3、6.5.4、7.1.1、7.6.2、7.9.3。
集群名称	自定义的集群名称，可输入的字符范围为 4～32 个字符，只能包含数字、字母、中划线和下划线，且必须以字母开头。 说明 当集群创建成功后，您可以根据需求修改集群名称。单击需要修改的集群名称，进入集群基本信息页面，单击“集群名称”后面的，修改完成后，单击，进行保存。如果需要取消修改，可单击进行取消。

图4-6 基本信息配置



5. 指定集群的主机规格相关参数。

表4-8 参数说明

参数	说明
节点数量	集群中的节点个数。 • 如果未启用 Master 节点和 Client 节点时，此参数指定的节点将被作为 Master 节点和 Client 节点，同时具备集群管理、存储数据、提供接入集群和分析数据的服务。此时，为保证集群中数据的稳定性，建议设置节点数量大于等于 3 个。 • 如果启用 Master 节点，且未启用 Client 节点，此参数指定的节点将用于存储数据并提供 Client 节点功能。 • 如果已启用 Master 节点和 Client 节点，此参数指定的节点将仅用于存储数据。 • 如果启用 Client 节点，且未启用 Master 节点，此参数指定的节点将用于存储数据并提供 Master 节点功能。
节点规格	集群中的节点规格。您可以根据需求，选择对应的规格。
节点存储	当前支持三种存储类型，普通 I/O、高 I/O、超高 I/O。
节点存储容量	存储空间大小，其取值范围与节点规格关联，不同的规格允许的取值范围不同。
启用 Master 节点	Master 节点用于管理集群中的所有节点。当需要存储和分析的数据量大，所需节点数量大于 20 个节点时，建议启用 Master 节点，保证集群的稳定性。反之，建议购买节点同时作为 Master 和 Client 节点即可，即仅设置集群的“节点数量”参数。 启用 Master 节点后，在下方选择对应的“节点规格”、“节点数量”和“节点存储”。“节点数量”必须是大于 3 的奇数，最多设置 9 个节点。其中“节点存储”的存储容量为固定值，存储类型可以根据实际情况选择，默认为存储容量为 40GB 的高 I/O 磁盘。 说明 只有开通大集群权限的用户才能配置“启用 Master 节点”和“启用 Client 节点”参数。
启用 Client 节点	Client 节点用于提供客户端接入集群和分析数据的服务。当需要存储和分析的数据量大，所需节点数量大于 20 个节点时，建议启用 Client 节点，保证集群的稳定性。反之，建议购买节点同时作为 Master 和 Client 节点即可，即仅设置集群的“节点数量”参数。 启用 Client 节点后，在下方选择对应的“节点规格”、“节点数量”和“节点存储”。“节点数量”可设置为 1~32 任意数值。其中“节点存储”的存储容量为固定值，存储类型可以根据实际情况选择，默认为存储容量为 40GB 的高 I/O 磁盘。
启用冷数据节点	冷数据节点用于存放对于历史数据要求分钟级别的返回。当用

参数	说明
	户对历史数据返回时间要求不是很高的话，可以将这部分数据存储在冷数据节点上，从而降低成本。 冷数据节点为可选节点，支持的节点个数为 1-32 个。 开启冷数据节点之后，云搜索服务将会自动的给相关节点打上冷热标签，相关的集群参数，配置详情请参见 https://www.elastic.co/guide/en/elasticsearch/reference/master/allocation-awareness.html。

图4-7 设置主机规格

6. 指定集群的网络规格相关参数。

表4-9 参数说明

参数	说明
虚拟私有云	VPC 即虚拟私有云，是通过逻辑方式进行网络隔离，提供安全、隔离的网络环境。 选择创建集群需要的 VPC，单击“查看虚拟私有云”进入 VPC 服务查看已创建的 VPC 名称和 ID。如果没有 VPC，需要创建一个新的 VPC。 说明 此处您选择的 VPC 必须包含网段（CIDR），否则集群将无法创建成功。新建的 VPC 默认包含网段（CIDR）。
子网	通过子网提供与其他网络隔离的、可以独享的网络资源，以提高网络安全。 选择创建集群需要的子网，可进入 VPC 服务查看 VPC 下已创

参数	说明
	建的子网名称和 ID。
安全组	安全组是一个逻辑上的分组，为同一个 VPC 内具有相同安全保护需求并相互信任的弹性云服务器提供访问策略。单击“查看安全组”可了解安全组详情。 **说明** **请确保安全组的“端口范围/ICMP 类型”为“Any”或者包含端口 9200 的端口范围。**
安全模式	关闭安全模式。
企业项目	如果您开通了企业项目，在创建云搜索服务集群时，您可以给集群绑定一个企业项目。您可以在右侧下拉框中选择当前用户下已创建的企业项目，也可以通过单击“查看项目管理”按钮，前往“企业项目管理”管理控制台，新建企业项目和查看已有的企业项目。
集群快照开关	• 基础配置 – OBS 桶：快照存储的 OBS 桶的名称。 – 备份路径：快照在 OBS 桶中的存放路径。 – IAM 委托：指当前账户授权云搜索服务访问或维护存储在 OBS 中数据。 具体详细信息，请参考《云搜索服务用户指南》中“备份与恢复索引”章节。 • 自动创建快照 默认情况下，系统打开了“自动快照创建”开关，您可以根据自己的需求设置“快照名称前缀”、“备份开始时间”和“保留时间（天）”。如果您不需要启用自动快照，可以在“自动快照创建”右侧，单击开关关闭自动创建快照功能。 – “快照名称前缀”：快照名称由快照名称前缀加上时间戳组成，例如自动生成的快照名称 snapshot-1566921603720。快照名称前缀的长度为 1～32 个字符，只能包含小写字母、数字、中划线和下划线，且必须以小写字母开头。 – “备份开始时间”：指每天自动开始备份的时间，只能指定整点时间，如 00:00、01:00，取值范围为 00:00～23:00。请在下拉框中选择备份时间。 – “保留时间（天）”：指备份的快照在 OBS 的保留时间，以天为单位，取值范围为 1～90，您可以根据自己的需求进行设置。系统在半点时刻会自动删除超过保留时间的快照。 例如：自动快照创建的策略设置如图 4-9 所示，则系统会在 35 天后的 00:30 自动删除 35 天前 00:00 自动开始备份的快照。

图4-8 设置网络规格

图4-9 设置自动快照创建的相关参数

7. 配置集群高级配置功能。

   “高级配置”：选择“默认配置”，则默认关闭“终端节点服务”和“标签”功能。如果需要配置“终端节点服务”、“标签”功能，选择“自定义”。

表4-10 高级配置参数

参数	说明
终端节点服务	开启终端节点服务后，用户可以获得一个内网访问的域名，通过这个域名，可以在同一个 vpc 内访问该集群。详细配置请参考 9.15 终端节点服务。
标签	为集群添加标签，可以方便用户识别和管理拥有的集群资源。此处您可以选择“标签管理服务”中已定义好的“预定义标签”，也可以自己定义标签。详细标签使用请参考 9.9 标签管理。

8. 单击“立即申请”，进入规格确认界面。
9. 规格确认完成后，单击“提交申请”开始创建集群。
10. 单击“返回集群列表”，系统将跳转到“集群管理”页面。您创建的集群将展现在集群列表中，且集群状态为“创建中”，创建成功后集群状态会变为“可用”。

如果集群创建失败，请根据界面提示，重新创建集群。

4.3 接入集群

集群创建完成后，可以接入集群开始使用 Elasticsearch 搜索引擎，如定义索引数据、导入数据、搜索数据等，Elasticsearch 搜索引擎相关内容的深入介绍可参见《Elasticsearch：权威指南》。接入方式有 4 种：

- 在管理控制台通过 Kibana 接入集群
- 在同一 VPC 内的弹性云服务器，直接调用 Elasticsearch API
- 非安全模式使用 Java API 接入集群
- Elasticsearch 安全模式 Java API 接入集群

在管理控制台通过 Kibana 接入集群

1. 登录云搜索服务管理控制台。
2. 在左侧导航栏，单击“集群管理”。
3. 在集群对应的“操作”列，单击“Kibana”，即可打开 Kibana 界面。

说明

单击“Kibana”将会在浏览器中打开一个新窗口，而弹出的新窗口有可能被浏览器拦截。如果单击“Kibana”后无新窗口打开，表示已被浏览器拦截。建议在浏览器中查看拦截信息，允许浏览器访问已拦截的弹出式窗口，即 kibana 的访问地址。

4. 在打开的 Kibana 页面中，您可以创建索引、查询索引和文档、对文档字段进行分析。 其中导入数据到 Elasticsearch 的操作指导请参见如下章节：
   - 5.1 **错误!未找到引用源。**
   - 5.1 **错误!未找到引用源。**
   - 5.1 使用 Logstash 导入数据到 Elasticsearch
   - 5.2 使用 Kibana 或 API 导入数据到 Elasticsearch

在同一 VPC 内的弹性云服务器，直接调用 Elasticsearch API

直接调用 Elasticsearch API 的弹性云服务器，需满足如下要求。创建并登录弹性云服务器的操作指导请参见弹性云服务器的“登录 Linux 弹性云服务器”或“登录 Windows 弹性云服务器”。

- 为弹性云服务分配足够的磁盘空间。
- 此弹性云服务器的 VPC 需要与集群在同一个 VPC 中。
- 此弹性云服务器的安全组需要和集群的安全组相同。

  如果不同，请修改弹性云服务器安全组或配置弹性云服务器安全组的出入规则允许集群所有安全组的访问。修改操作请参见《虚拟私有云用户指南》。

- 待接入的 ES 集群，其安全组的出方向和入方向需允许 TCP 协议及 9200 端口，或者允许端口范围包含 9200 端口。

操作步骤如下所示：

1. 创建并登录满足要求的弹性云服务器。
2. 接入集群时需要使用内网访问地址和端口号，您可以在集群列表中的“内网访问地址”列获取节点内网访问地址。如果集群只有 1 个节点，此处仅显示 1 个节点的内网访问地址和端口号，如果集群有多个节点，此处显示所有节点的内网访问地址和端口号。

   例如，集群中有 2 个节点，集群列表显示的值为“10.62.179.32:9200 10.62.179.33:9200”，表示 2 个节点的内网访问地址分别为“10.62.179.32”和“10.62.179.33”，访问节点需使用端口都为“9200”。
3. 在此弹性云服务器中，直接通过 cURL 执行 API 或者开发程序调用 API 并执行程序即可使用集群。Elasticsearch 操作和接口请参见《Elasticsearch：权威指南》。

   例如，使用 cURL 执行如下命令，查看集群中的索引信息，集群中某一个节点的内网访问地址为“10.62.179.32”，端口为“9200”。

   - 如果接入集群未启用安全模式，接入方式为：

     ```
     curl 'http://10.62.179.32:9200/_cat/indices'
     ```

   - 如果接入集群已启用安全模式，则需要使用 https 方式访问，并附加用户名和密码，在 curl 命令中添加-u 选项。

     ```
     curl -u username:password -k 'https://10.62.179.32:9200/_cat/indices'
     ```

说明

此处仅使用集群中某一个节点内网访问地址和端口号，当该节点出现故障时，将导致命令执行失败。如果集群包含多个节点，可以将节点内网访问地址和端口号替换为集群中另一节点的内网访问地址和端口号；如果集群只包含一个节点，则需要将该节点修复之后再次执行命令。

以接入未设置通信加密的集群为例，其结果如下图所示。

图4-10 执行结果

```
SZX1000355659:/home/elasticsearch-5.5.2/bin # curl 'http://10.62.179.32:9200
/_cat/indices'
green open new_twitter BSVY8wt0SIWSXGzZ5U5mzw 5 1  3 0 21.3kb 10.6kb
green open .kibana     ks7lz4ggTUCy2UDWkXqEgw 1 1  2 0 22.8kb 11.4kb
green open tweets_1    FXOn8ykvQrmvBISyKbFRHA 5 1  0 0  1.5kb   810b
green open my_store    AWybpSpLQPK_2T4cWDN-TQ 5 1 20 0 41.2kb 20.6kb
green open my_index    QF5ARy2VQ6G0t8BzZEI86g 5 1  1 0    9kb  4.5kb
green open tweets_2    uLdSGZ0BS7uamlQfxDlEsQ 5 1  0 0  1.5kb   810b
green open twitter     lzPIdrmRQpeBglI76SAYGA 5 2  3 0 40.5kb 13.5kb
green open my_index2   oLjbtIBPSNqeVIXgHOXQHg 5 1  0 0  1.8kb   955b
```

非安全模式使用 Java API 接入集群

非安全模式即 6.5.4 及之后版本集群未开启安全模式状态和其他版本的集群状态，我们为您提供两种方式：TransportClient 类和 RestHighLevelClient 类建立客户端，并且在 6.2.3 版本和 5.5.1 版本中建议您使用 TransportClient，6.5.4 及之后版本中建议您使用 RestHighLevelClient。

- 使用 TransportClient 类默认方法建立客户端：

  ```
  Settings settings =
  ImmutableSettings.settingsBuilder().put("client.transport.sniff",false).build()
  ;
  ```

```
TransportClient client = new TransportClient(settings) .addTransportAddress(new
InetSocketTransportAddress("host1", 9300));
```

- 使用 RestHighLevelClient 类默认方法建立客户端：

```
RestHighLevelClient client = new RestHighLevelClient(
      RestClient.builder(
            new HttpHost("localhost", 9200, "http")));
```

Elasticsearch 安全模式 Java API 接入集群

当您在 Elasticsearch 6.5.4 及之后版本中开启安全模式后，需要使用 https 方式连接集群，同时还需要提供用户名和密码进行身份验证。

对于使用 Java API 连接方式，旧版本提供的 TransportClient 不能实现使用用户名和密码连接集群，所以需要使用 6.5.4 及之后版本配套的相关 API 进行相关开发。

这里提供两种方式进行连接：TransportClient 类和 RestHighLevelClient 类建立客户端。但是推荐使用 RestHighLevelClient 方式。

- **使用 TransportClient 类建立客户端**

首先，使用以下步骤在客户端使用命令行分别生成 keystore 和 truststore 文件，其中使用到从集群管理界面下载的证书（CloudSearchService.cer）。证书下载可参考图 4-11 进行下载。

```
keytool -genkeypair -alias certificatekey -keyalg RSA -keystore transport-
keystore.jks
keytool -import -alias certificatekey -file CloudSearchService.cer  -keystore
truststore.jks
```

然后，使用生成的 keystore、truststore 文件，放入集群连接设置，使用 PreBuiltTransportClient 方法建立 TransportClient 类，并将连接设置放入客户端线程中。

图4-11 下载证书

关键代码如下所示：

```
String userPw = "username:password";
String path =
Paths.get(SecurityTransportClientDemo.class.getClassLoader().getResource(".").t
oURI()).toString();

Settings settings = Settings.builder()
              .put("opendistro_security.ssl.transport.enforce_hostname_verifica
tion", false)
              .put("opendistro_security.ssl.transport.keystore_filepath", path
+ "/transport-keystore.jks")
              .put("opendistro_security.ssl.transport.keystore_password",
"tscpass")
              .put("opendistro_security.ssl.transport.truststore_filepath",
path + "/truststore.jks")
              .put("client.transport.ignore_cluster_name", "true")
              .put("client.transport.sniff", false).build();

TransportClient client = (new PreBuiltTransportClient(settings, new
Class[]{OpenDistroSecurityPlugin.class})).addTransportAddress(new
                  TransportAddress(InetAddress.getByName(ip), 9300));

String base64UserPw = Base64.getEncoder().encodeToString(userPw.getBytes("utf-
8"));
            client.threadPool().getThreadContext().putHeader("Authorization",
"Basic " + base64UserPw);
```

- **使用 RestHighLevelClient 建立客户端**

  其中使用 HttpHost 类负责 http 请求，并在 HttpHost 类中将 CredentialsProvider 和 SSLIOSessionStrategy 配置参数类封装在自定义的 SecuredHttpClientConfigCallback 类配置请求连接参数。

  SecuredHttpClientConfigCallback：封装所有自定义的连接参数。

  CredentialsProvider：Elasticsearch API，主要使用 Elasticsearch 提供的方法封装用户名和密码。

  SSLIOSessionStrategy：配置 SSL 相关参数，包括 SSL 域名验证方式、证书处理方式等。主要使用 SSLContext 类封装相关设置。

  有两种方式连接集群：忽略证书方式和使用证书方式。

  - 忽略所有证书，跳过证书校验环节进行连接

    构造 TrustManager，使用默认 X509TrustManager，不重写任何方法，相当于忽略所有相关操作。

    构造 SSLContext：使用第一步的 TrustManager 为参数，默认方法构造 SSLContext。

```
static TrustManager[] trustAllCerts = new TrustManager[] { new
X509TrustManager() {
      @Override
      public void checkClientTrusted(X509Certificate[] chain, String
authType) throws CertificateException {

      }
      @Override
      public void checkServerTrusted(X509Certificate[] chain, String
authType) throws CertificateException {
```

```
        }
        @Override
        public X509Certificate[] getAcceptedIssuers() {
            return null;
        }
    }};
 final CredentialsProvider credentialsProvider = new
BasicCredentialsProvider();
        credentialsProvider.setCredentials(AuthScope.ANY,
                new UsernamePasswordCredentials(userName, password));
        SSLContext sc = null;
        try{
            sc = SSLContext.getInstance("SSL");
            sc.init(null, trustAllCerts, new SecureRandom());
        }catch(KeyManagementException e){
                e.printStackTrace();
        }catch(NoSuchAlgorithmException e){
                e.printStackTrace();
        }
        SSLIOSessionStrategy sessionStrategy = new SSLIOSessionStrategy(sc,
new NullHostNameVerifier());

        SecuredHttpClientConfigCallback httpClientConfigCallback = new
SecuredHttpClientConfigCallback(sessionStrategy,credentialsProvider);

        RestClientBuilder builder = RestClient.builder(new
HttpHost(clusterAddress, 9200,

"https")).setHttpClientConfigCallback(httpClientConfigCallback);

        RestHighLevelClient client = new RestHighLevelClient(builder);
```

- 使用下载的证书（CloudSearchService.cer），加载证书进行连接。

  上传证书到客户端，在命令行中使用 keytool 工具将证书转换成 Java 可以读取的证书格式：（keytool 默认密码为 changeit）

```
keytool -import -alias 自命名 -keystore 输出的证书路径和重命名名字 -file 上传证书的路径
```

  自定义 TrustManager 类，继承于 X509TrustManager，读取上一步输出的证书，将其加入信任证书里，重写 X509TrustManager 接口的三个方法；

  构造 SSLContext：使用第一步的 TrustManager 为参数，默认方法构造 SSLContext。

```
public static class MyX509TrustManager implements X509TrustManager {

        X509TrustManager sunJSSEX509TrustManager;
        MyX509TrustManager() throws Exception {
            File file = new File("certification file path");
            if (file.isFile() == false) {
                throw new Exception("Wrong Certification Path");
            }
            System.out.println("Loading KeyStore " + file + "...");
            InputStream in = new FileInputStream(file);
            KeyStore ks = KeyStore.getInstance("JKS");
            ks.load(in, "changeit".toCharArray());
```

```
            TrustManagerFactory tmf =
                  TrustManagerFactory.getInstance("SunX509", "SunJSSE");
            tmf.init(ks);
            TrustManager tms [] = tmf.getTrustManagers();
            for (int i = 0; i < tms.length; i++) {
                if (tms[i] instanceof X509TrustManager) {
                    sunJSSEX509TrustManager = (X509TrustManager) tms[i];
                    return;
                }
            }
            throw new Exception("Couldn't initialize");
        }

final CredentialsProvider credentialsProvider = new
BasicCredentialsProvider();
        credentialsProvider.setCredentials(AuthScope.ANY,
              new UsernamePasswordCredentials(userName, password));

        SSLContext sc = null;
        try{
            TrustManager[] tm = {new MyX509TrustManager()};
            sc = SSLContext.getInstance("SSL", "SunJSSE");
            sc.init(null, tm, new SecureRandom());
        }catch (Exception e) {
            e.printStackTrace();
        }

        SSLIOSessionStrategy sessionStrategy = new SSLIOSessionStrategy(sc,
new NullHostNameVerifier());

        SecuredHttpClientConfigCallback httpClientConfigCallback = new
SecuredHttpClientConfigCallback(sessionStrategy,credentialsProvider);
        RestClientBuilder builder = RestClient.builder(new
HttpHost(clusterAddress, 9200, "https"))
              .setHttpClientConfigCallback(httpClientConfigCallback);
        RestHighLevelClient client = new RestHighLevelClient(builder);
```

- 代码示例

  代码运行时，传入 3 个参数，分别是**连接地址**，**集群登录用户名**和**密码**，示例实现的请求是 GET /_search{"query": {"match_all": {}}}。

说明

安全集群的链接地址一般是以 https 开头。

ESSecuredClient 类（忽略证书方式）

```
package securitymode;

import org.apache.http.auth.AuthScope;
import org.apache.http.auth.UsernamePasswordCredentials;
import org.apache.http.client.CredentialsProvider;
import org.apache.http.impl.client.BasicCredentialsProvider;
import org.apache.http.HttpHost;
import org.apache.http.nio.conn.ssl.SSLIOSessionStrategy;
import org.elasticsearch.action.search.SearchRequest;
import org.elasticsearch.action.search.SearchResponse;
```

```
import org.elasticsearch.client.RequestOptions;
import org.elasticsearch.client.RestClient;
import org.elasticsearch.client.RestClientBuilder;
import org.elasticsearch.client.RestHighLevelClient;
import org.elasticsearch.index.query.QueryBuilders;
import org.elasticsearch.search.SearchHit;
import org.elasticsearch.search.SearchHits;
import org.elasticsearch.search.builder.SearchSourceBuilder;

import java.io.IOException;
import java.security.KeyManagementException;
import java.security.NoSuchAlgorithmException;
import java.security.SecureRandom;
import java.security.cert.CertificateException;
import java.security.cert.X509Certificate;

import javax.net.ssl.HostnameVerifier;
import javax.net.ssl.SSLContext;
import javax.net.ssl.SSLSession;
import javax.net.ssl.TrustManager;
import javax.net.ssl.X509TrustManager;

public class ESSecuredClientIgnoreCerDemo {

   public static void main(String[] args) {
      String clusterAddress = args[0];
      String userName = args[1];
      String password = args[2];
      // 建立客户端
      RestHighLevelClient client = initESClient(clusterAddress, userName,
password);
      try {
         // 查询 match_all，相当于 {\"query\": {\"match_all\": {}}}
         SearchRequest searchRequest = new SearchRequest();
         SearchSourceBuilder searchSourceBuilder = new
SearchSourceBuilder();
         searchSourceBuilder.query(QueryBuilders.matchAllQuery());
         searchRequest.source(searchSourceBuilder);

         SearchResponse searchResponse = client.search(searchRequest,
RequestOptions.DEFAULT);
         System.out.println("query result: " +
searchResponse.toString());
         SearchHits hits = searchResponse.getHits();
         for (SearchHit hit : hits) {
            System.out.println(hit.getSourceAsString());
         }
         System.out.println("query success");
         Thread.sleep(2000L);
      } catch (InterruptedException | IOException e) {
         e.printStackTrace();
      } finally {
         try {
            client.close();
            System.out.println("close client");
```

```
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }

    private static RestHighLevelClient initESClient(String clusterAddress,
String userName, String password) {
        final CredentialsProvider credentialsProvider = new
BasicCredentialsProvider();
        credentialsProvider.setCredentials(AuthScope.ANY, new
UsernamePasswordCredentials(userName, password));
        SSLContext sc = null;
        try {
            sc = SSLContext.getInstance("SSL");
            sc.init(null, trustAllCerts, new SecureRandom());
        } catch (KeyManagementException | NoSuchAlgorithmException e) {
            e.printStackTrace();
        }
        SSLIOSessionStrategy sessionStrategy = new SSLIOSessionStrategy(sc,
new NullHostNameVerifier());
        SecuredHttpClientConfigCallback httpClientConfigCallback = new
SecuredHttpClientConfigCallback(sessionStrategy,
            credentialsProvider);
        RestClientBuilder builder = RestClient.builder(new
HttpHost(clusterAddress, 9200, "https"))
            .setHttpClientConfigCallback(httpClientConfigCallback);
        RestHighLevelClient client = new RestHighLevelClient(builder);
        return client;
    }

    static TrustManager[] trustAllCerts = new TrustManager[] {
        new X509TrustManager() {
            @Override
            public void checkClientTrusted(X509Certificate[] chain, String
authType) throws CertificateException {
            }

            @Override
            public void checkServerTrusted(X509Certificate[] chain, String
authType) throws CertificateException {
            }

            @Override
            public X509Certificate[] getAcceptedIssuers() {
                return null;
            }
        }
    };

    public static class NullHostNameVerifier implements HostnameVerifier {
        @Override
        public boolean verify(String arg0, SSLSession arg1) {
            return true;
        }
```

```
    }

}
```

ESSecuredClient 类（使用证书方式）

```
package securitymode;

import org.apache.http.auth.AuthScope;
import org.apache.http.auth.UsernamePasswordCredentials;
import org.apache.http.client.CredentialsProvider;
import org.apache.http.impl.client.BasicCredentialsProvider;
import org.apache.http.HttpHost;
import org.apache.http.nio.conn.ssl.SSLIOSessionStrategy;
import org.elasticsearch.action.search.SearchRequest;
import org.elasticsearch.action.search.SearchResponse;
import org.elasticsearch.client.RequestOptions;
import org.elasticsearch.client.RestClient;
import org.elasticsearch.client.RestClientBuilder;
import org.elasticsearch.client.RestHighLevelClient;
import org.elasticsearch.index.query.QueryBuilders;
import org.elasticsearch.search.SearchHit;
import org.elasticsearch.search.SearchHits;
import org.elasticsearch.search.builder.SearchSourceBuilder;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.security.KeyStore;
import java.security.SecureRandom;
import java.security.cert.CertificateException;
import java.security.cert.X509Certificate;

import javax.net.ssl.HostnameVerifier;
import javax.net.ssl.SSLContext;
import javax.net.ssl.SSLSession;
import javax.net.ssl.TrustManager;
import javax.net.ssl.TrustManagerFactory;
import javax.net.ssl.X509TrustManager;

public class ESSecuredClientWithCerDemo {

    public static void main(String[] args) {
        String clusterAddress = args[0];
        String userName = args[1];
        String password = args[2];
        String cerFilePath = args[3];
        String cerPassword = args[4];
        // 建立客户端
        RestHighLevelClient client = initESClient(clusterAddress, userName,
password, cerFilePath, cerPassword);
        try {
            // 查询 match_all，相当于 {\"query\": {\"match_all\": {}}}
            SearchRequest searchRequest = new SearchRequest();
            SearchSourceBuilder searchSourceBuilder = new
SearchSourceBuilder();
```

```
            searchSourceBuilder.query(QueryBuilders.matchAllQuery());
            searchRequest.source(searchSourceBuilder);

            // query
            SearchResponse searchResponse = client.search(searchRequest,
RequestOptions.DEFAULT);
            System.out.println("query result: " +
searchResponse.toString());
            SearchHits hits = searchResponse.getHits();
            for (SearchHit hit : hits) {
               System.out.println(hit.getSourceAsString());
            }
            System.out.println("query success");
            Thread.sleep(2000L);
         } catch (InterruptedException | IOException e) {
            e.printStackTrace();
         } finally {
            try {
               client.close();
               System.out.println("close client");
            } catch (IOException e) {
               e.printStackTrace();
            }
         }
      }

      private static RestHighLevelClient initESClient(String clusterAddress,
String userName, String password,
         String cerFilePath, String cerPassword) {
         final CredentialsProvider credentialsProvider = new
BasicCredentialsProvider();
         credentialsProvider.setCredentials(AuthScope.ANY, new
UsernamePasswordCredentials(userName, password));
         SSLContext sc = null;
         try {
            TrustManager[] tm = {new MyX509TrustManager(cerFilePath,
cerPassword)};
            sc = SSLContext.getInstance("SSL", "SunJSSE");
            sc.init(null, tm, new SecureRandom());
         } catch (Exception e) {
            e.printStackTrace();
         }

         SSLIOSessionStrategy sessionStrategy = new SSLIOSessionStrategy(sc,
new NullHostNameVerifier());
         SecuredHttpClientConfigCallback httpClientConfigCallback = new
SecuredHttpClientConfigCallback(sessionStrategy,
            credentialsProvider);
         RestClientBuilder builder = RestClient.builder(new
HttpHost(clusterAddress, 9200, "https"))
            .setHttpClientConfigCallback(httpClientConfigCallback);
         RestHighLevelClient client = new RestHighLevelClient(builder);
         return client;
      }
```

```
   public static class MyX509TrustManager implements X509TrustManager {
      X509TrustManager sunJSSEX509TrustManager;

      MyX509TrustManager(String cerFilePath, String cerPassword) throws
Exception {
         File file = new File(cerFilePath);
         if (!file.isFile()) {
            throw new Exception("Wrong Certification Path");
         }
         System.out.println("Loading KeyStore " + file + "...");
         InputStream in = new FileInputStream(file);
         KeyStore ks = KeyStore.getInstance("JKS");
         ks.load(in, cerPassword.toCharArray());
         TrustManagerFactory tmf =
TrustManagerFactory.getInstance("SunX509", "SunJSSE");
         tmf.init(ks);
         TrustManager[] tms = tmf.getTrustManagers();
         for (TrustManager tm : tms) {
            if (tm instanceof X509TrustManager) {
               sunJSSEX509TrustManager = (X509TrustManager) tm;
               return;
            }
         }
         throw new Exception("Couldn't initialize");
      }

      @Override
      public void checkClientTrusted(X509Certificate[] chain, String
authType) throws CertificateException {

      }

      @Override
      public void checkServerTrusted(X509Certificate[] chain, String
authType) throws CertificateException {

      }

      @Override
      public X509Certificate[] getAcceptedIssuers() {
         return new X509Certificate[0];
      }
   }

   public static class NullHostNameVerifier implements HostnameVerifier {
      @Override
      public boolean verify(String arg0, SSLSession arg1) {
         return true;
      }
   }

}
```

SecuredHttpClientConfigCallback 类

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

```
import org.apache.http.nio.conn.ssl.SSLIOSessionStrategy;
import org.elasticsearch.client.RestClientBuilder;
import org.elasticsearch.common.Nullable;
import java.util.Objects;
class SecuredHttpClientConfigCallback implements
RestClientBuilder.HttpClientConfigCallback {
   @Nullable
   private final CredentialsProvider credentialsProvider;
   /**
    * The {@link SSLIOSessionStrategy} for all requests to enable SSL /
TLS encryption.
    */
   private final SSLIOSessionStrategy sslStrategy;
   /**
    * Create a new {@link SecuredHttpClientConfigCallback}.
    *
    * @param credentialsProvider The credential provider, if a
username/password have been supplied
    * @param sslStrategy       The SSL strategy, if SSL / TLS have been
supplied
    * @throws NullPointerException if {@code sslStrategy} is {@code null}
    */
   SecuredHttpClientConfigCallback(final SSLIOSessionStrategy sslStrategy,
                                   @Nullable final CredentialsProvider
credentialsProvider) {
      this.sslStrategy = Objects.requireNonNull(sslStrategy);
      this.credentialsProvider = credentialsProvider;
   }
   /**
    * Get the {@link CredentialsProvider} that will be added to the HTTP
client.
    *
    * @return Can be {@code null}.
    */
   @Nullable
   CredentialsProvider getCredentialsProvider() {
      return credentialsProvider;
   }
   /**
    * Get the {@link SSLIOSessionStrategy} that will be added to the HTTP
client.
    *
    * @return Never {@code null}.
    */
   SSLIOSessionStrategy getSSLStrategy() {
      return sslStrategy;
   }
   /**
    * Sets the {@linkplain
HttpAsyncClientBuilder#setDefaultCredentialsProvider(CredentialsProvider)
credential provider},
    *
    * @param httpClientBuilder The client to configure.
    * @return Always {@code httpClientBuilder}.
    */
```

```
    @Override
    public HttpAsyncClientBuilder customizeHttpClient(final
HttpAsyncClientBuilder httpClientBuilder) {
        // enable SSL / TLS
        httpClientBuilder.setSSLStrategy(sslStrategy);
        // enable user authentication
        if (credentialsProvider != null) {

httpClientBuilder.setDefaultCredentialsProvider(credentialsProvider);
        }
        return httpClientBuilder;
    }
}
```

pom.xml 文件

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>1</groupId>
    <artifactId>ESClient</artifactId>
    <version>1.0-SNAPSHOT</version>
    <name>ESClient</name>
    <!-- FIXME change it to the project's website -->
    <url>http://www.example.com</url>
    <properties>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
        <maven.compiler.source>1.7</maven.compiler.source>
        <maven.compiler.target>1.7</maven.compiler.target>
    </properties>
    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.11</version>
            <scope>test</scope>
        </dependency>
        <dependency>
            <groupId>org.elasticsearch.client</groupId>
            <artifactId>transport</artifactId>
            <version>6.5.4</version>
        </dependency>
        <dependency>
            <groupId>org.elasticsearch</groupId>
            <artifactId>elasticsearch</artifactId>
            <version>6.5.4</version>
        </dependency>
        <dependency>
            <groupId>org.elasticsearch.client</groupId>
            <artifactId>elasticsearch-rest-high-level-client</artifactId>
            <version>6.5.4</version>
        </dependency>
        <dependency>
            <groupId>org.apache.logging.log4j</groupId>
```

```
            <artifactId>log4j-api</artifactId>
            <version>2.7</version>
        </dependency>
        <dependency>
            <groupId>org.apache.logging.log4j</groupId>
            <artifactId>log4j-core</artifactId>
            <version>2.7</version>
        </dependency>
    </dependencies>
    <build>
        <pluginManagement><!-- lock down plugins versions to avoid using
Maven defaults (may be moved to parent pom) -->
            <plugins>
                <!-- clean lifecycle, see
https://maven.apache.org/ref/current/maven-
core/lifecycles.html#clean_Lifecycle -->
                <plugin>
                    <artifactId>maven-clean-plugin</artifactId>
                    <version>3.1.0</version>
                </plugin>
                <!-- default lifecycle, jar packaging: see
https://maven.apache.org/ref/current/maven-core/default-
bindings.html#Plugin_bindings_for_jar_packaging -->
                <plugin>
                    <artifactId>maven-resources-plugin</artifactId>
                    <version>3.0.2</version>
                </plugin>
                <plugin>
                    <artifactId>maven-compiler-plugin</artifactId>
                    <version>3.8.0</version>
                </plugin>
                <plugin>
                    <artifactId>maven-surefire-plugin</artifactId>
                    <version>2.22.1</version>
                </plugin>
                <plugin>
                    <artifactId>maven-jar-plugin</artifactId>
                    <version>3.0.2</version>
                </plugin>
                <plugin>
                    <artifactId>maven-install-plugin</artifactId>
                    <version>2.5.2</version>
                </plugin>
                <plugin>
                    <artifactId>maven-deploy-plugin</artifactId>
                    <version>2.8.2</version>
                </plugin>
                <!-- site lifecycle, see
https://maven.apache.org/ref/current/maven-
core/lifecycles.html#site_Lifecycle -->
                <plugin>
                    <artifactId>maven-site-plugin</artifactId>
                    <version>3.7.1</version>
                </plugin>
                <plugin>
```

```
                <artifactId>maven-project-info-reports-plugin</artifactId>
                <version>3.0.0</version>
            </plugin>
        </plugins>
    </pluginManagement>
</build>
</project>
```

5 导入数据到 Elasticsearch

5.1 使用 Logstash 导入数据到 Elasticsearch

云搜索服务支持使用 Logstash 将其收集的数据迁移到 Elasticsearch 中，方便用户通过 Elasticsearch 搜索引擎高效管理和获取数据。数据文件支持 JSON、CSV 等格式。

Logstash 是开源的服务器端数据处理管道，能够同时从多个来源采集数据、转换数据，然后将数据发送到 Elasticsearch 中。Logstash 的官方文档请参见：https://www.elastic.co/guide/en/logstash/current/getting-started-with-logstash.html。

数据导入分为如下 2 种场景：

- Logstash 部署在外网时导入数据
- Logstash 部署在弹性云服务器上时导入数据

前提条件

- 为方便操作，建议采用 Linux 操作系统的机器部署 Logstash。
- Logstash 的下载路径为：https://www.elastic.co/cn/downloads/logstash-oss

说明

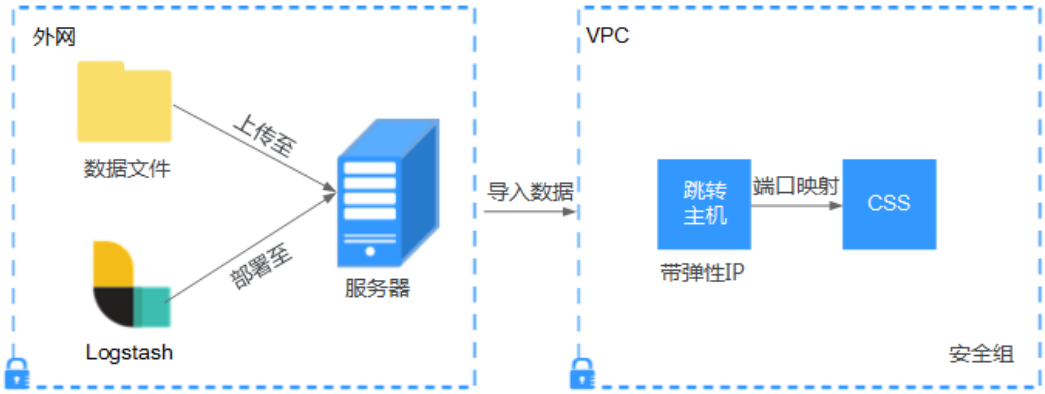
Logstash 要求使用 OSS 版本，选择和 ES 一致版本。

- 安装完 Logstash 后，再根据如下步骤导入数据。安装 Logstash 的操作指导，请参见：https://www.elastic.co/guide/en/logstash/current/installing-logstash.html
- 安装 Logstash 之前，需要先安装 JDK。在 Linux 操作系统中，您可以执行 **yum -y install java-1.8.0** 命令直接安装 1.8.0 版本 JDK。在 Windows 操作系统中，您可以访问 JDK 官网，下载符合操作系统版本的 JDK，并根据指导安装。
- 在“Logstash 部署在弹性云服务器上时导入数据”场景中，请确保此弹性云服务器与接入的 Elasticsearch 集群在同一个 VPC 下。

Logstash 部署在外网时导入数据

当 Logstash 部署在外网时，导入数据的流程说明如图 5-1 所示。

图5-1 Logstash 部署在外网时导入数据示意图



1. 创建一个跳转主机，并按如下要求进行配置。
   - 跳转主机为一台 Linux 操作系统的弹性云服务器，且已绑定弹性 IP。
   - 跳转主机与 ES 集群在同一虚拟私有云下。
   - 已开放跳转主机的本地端口，用于 SSH 转发，能够从本地端口转发至 ES 集群某一节点的 9200 端口。
   - 关于跳转主机的本地端口转发配置，请参见 SSH 官方文档。
2. 使用 PuTTY，通过弹性 IP 登录已创建的跳转主机。
3. 执行如下命令进行端口映射，将发往跳转主机对外开放端口的请求转发到待导入数据的集群中。

   ```
   ssh -g -L <跳转主机的本地端口:节点的内网访问地址和端口号> -N -f root@<跳转主机的私网 IP 地址>
   ```

   说明

   - *<跳转主机的本地端口>*：为步骤 1 中的端口。
   - *<节点的内网访问地址和端口号>*：为集群中某一节点的内网访问地址和端口号。当该节点出现故障时，将导致命令执行失败。如果集群包含多个节点，可以将*<节点的内网访问地址和端口号>*替换为集群中另一节点的内网访问地址和端口号；如果集群只包含一个节点，则需要将该节点修复之后再次执行命令进行端口映射。
   - *<跳转主机的私网 IP 地址>*：打开弹性云服务器管理控制台，从“IP 地址”列中获取标有“私网”对应的 IP 地址。

   例如：跳转主机对外开放的端口号为 9200，节点的内网访问地址和端口号为 192.168.0.81:9200，跳转主机的私网 IP 地址为 192.168.0.227，需要执行如下命令进行端口映射。

   ```
   ssh -g -L 9200:192.168.0.81:9200 -N -f root@192.168.0.227
   ```

4. 登录部署了 Logstash 的服务器，将需要进行操作的数据文件存储至此服务器中。

   例如，需要导入的数据文件“access_20181029_log”，文件存储路径为“/tmp/access_log/”，此数据文件中包含的数据如下所示：

   ```
   |   All |            Heap used for segments |                   |    18.6403 |
   MB |
   |   All |          Heap used for doc values |                   |   0.119289
   ```

```
|     MB |
|   All |               Heap used for terms |                    |    17.4095 |
MB |
|   All |               Heap used for norms |                    |   0.0767822 |
MB |
|   All |              Heap used for points |                    |    0.225246 |
MB |
|   All |        Heap used for stored fields |                    |    0.809448
|     MB |
|   All |                    Segment count |                    |        101 |
|
|   All |                   Min Throughput |         index-append |     66232.6
|  docs/s |
|   All |                Median Throughput |         index-append |     66735.3
|  docs/s |
|   All |                   Max Throughput |         index-append |     67745.6
|  docs/s |
|   All |           50th percentile latency |         index-append |
510.261 |     ms |
```

5. 在部署 Logstash 的服务器中，执行如下命令在 Logstash 的安装目录下新建配置文件 logstash-simple.conf。

```
cd /<Logstash 的安装目录>/
vi logstash-simple.conf
```

6. 在配置文件 logstash-simple.conf 中输入如下内容。

```
input {
数据所在的位置
}
filter {
数据的相关处理
}
output {
   elasticsearch {
      hosts => "<跳转主机的公网 IP 地址>:<跳转主机对外开放的端口号>"
   }
}
```

- input：指明了数据的来源。实际请根据用户的具体情况来设置。input 参数的详细解释和使用介绍请参见 https://www.elastic.co/guide/en/logstash/current/input-plugins.html。
- filter：指定对数据进行处理的方式。例如，对日志进行了提取和处理，将非结构化信息转换为结构化信息。filter 参数的详细解释和使用介绍请参见 https://www.elastic.co/guide/en/logstash/current/filter-plugins.html。
- output：指明了数据的目的地址。output 参数的详细解释和使用介绍请参见 https://www.elastic.co/guide/en/logstash/current/output-plugins.html。*<跳转主机的公网 IP 地址>*请从弹性云服务器管理控制台的“IP 地址”列中获取标有“弹性公网”对应的 IP 地址。*<跳转主机对外开放的端口号>*即为步骤 1 中的端口，例如：9200。

以步骤 4 中“/tmp/access_log/”的数据文件为例，输入数据文件从首行开始，且过滤条件保持为空，即不做任何数据处理操作。跳转主机的公网 IP 和端口号为“192.168.0.227:9200”。导入数据的索引名称为“myindex”。配置文件的示例如下所示，配置文件按实际数据情况修改完成后，输入“:wq”保存。

```
input {
    file{
      path => "/tmp/access_log/*"
      start_position => "beginning"
    }
}
filter {
}
output {
    elasticsearch {
      hosts => "192.168.0.227:9200"
      index => "myindex"
      document_type => "mytype"
    }
}
```

说明

如果在使用中出现 license 相关的报错，可以尝试设置 ilm_enabled => false。

如果集群开启了安全模式，则需要先下载证书。

a. 在集群基本信息页面下载证书。

图5-2 下载证书

b. 将下载的证书存放到部署 logstash 服务器中。

c. 修改配置文件 logstash-simple.conf。

以步骤 4 中“/tmp/access_log/”的数据文件为例，输入数据文件从首行开始，且过滤条件保持为空，即不做任何数据处理操作。跳转主机的公网 IP 和端口号为“192.168.0.227:9200”。导入数据的索引名称为“myindex”，证书存放路径为“/logstash/logstash6.8/config/CloudSearchService.cer”。配置文件的示例如下所示，配置文件按实际数据情况修改完成后，输入“:wq”保存。

```
input{
   file {
      path => "/tmp/access_log/*"
```

```
        start_position => "beginning"
    }
}
filter {
    }
output{
    elasticsearch{
        hosts => ["https://192.168.0.227:9200"]
        index => "myindex"
        user => "admin"
        password => "******"
        cacert => "/logstash/logstash6.8/config/CloudSearchService.cer"
    }
}
```

说明

password：**登录安全集群的密码。**

7. 执行如下命令将 Logstash 收集的数据导入到集群中。

```
./bin/logstash -f logstash-simple.conf
```

8. 登录云搜索服务管理控制台。
9. 在左侧导航栏中，选择“集群管理”，进入集群列表页面。
10. 在集群列表页面中，单击待导入数据的集群“操作”列的“Kibana”。
11. 在 Kibana 的左侧导航中选择“Dev Tools”，单击“Get to work”，进入 Console 界面。
12. 在已打开的 Kibana 的 Console 界面，通过搜索获取已导入的数据。

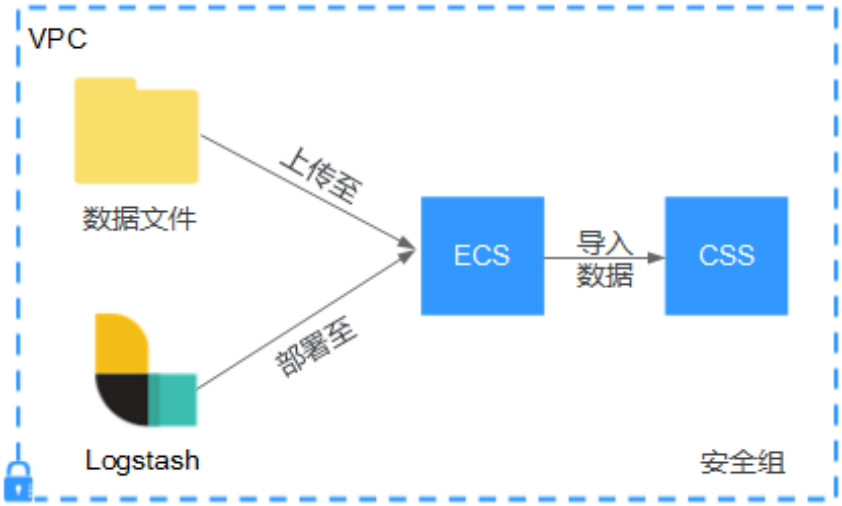
    在 Kibana 控制台，输入如下命令，搜索数据。查看搜索结果，如果数据与导入数据一致，表示数据文件的数据已导入成功。

```
GET myindex/_search
```

Logstash 部署在弹性云服务器上时导入数据

当 Logstash 部署在同一 VPC 的弹性云服务时，导入数据的流程说明如图 5-3 所示。

图5-3 Logstash 部署在弹性云服务器上时导入数据示意图



1. 确保已部署 Logstash 的弹性云服务器与待导入数据的集群在同一虚拟私有云下，已开放安全组的 9200 端口的外网访问权限，且弹性云服务器已绑定弹性 IP。

   说明

   - 如果同一个 VPC 内有多台服务器，只要其中一台绑定了弹性 IP，其他的服务器可以不需要绑定弹性 IP。通过绑定弹性 IP 的节点跳转到部署 Logstash 的节点即可。
   - 如果有专线或者 VPN，也不需要绑定弹性 IP。

2. 使用 PuTTY 登录弹性云服务器。

   例如此服务器中存储了需要导入的数据文件“access_20181029_log”，文件存储路径为“/tmp/access_log/”，此数据文件中包含的数据如下所示：

```
|   All |             Heap used for segments |                    |    18.6403 |
MB |
|   All |           Heap used for doc values |                    |   0.119289
|     MB |
|   All |               Heap used for terms |                    |    17.4095 |
MB |
|   All |               Heap used for norms |                    |   0.0767822 |
MB |
|   All |              Heap used for points |                    |   0.225246 |
MB |
|   All |        Heap used for stored fields |                    |   0.809448
|     MB |
|   All |                    Segment count |                    |        101 |
|
|   All |                   Min Throughput |         index-append |    66232.6
|  docs/s |
|   All |                Median Throughput |         index-append |    66735.3
|  docs/s |
|   All |                   Max Throughput |         index-append |    67745.6
|  docs/s |
|   All |            50th percentile latency |         index-append |
510.261 |     ms |
```

3. 执行如下命令在 Logstash 的安装目录下新建配置文件 logstash-simple.conf。

```
cd /<Logstash 的安装目录>/
vi logstash-simple.conf
```

在配置文件 logstash-simple.conf 中输入如下内容。

```
input {
数据所在的位置
}
filter {
数据的相关处理
}
output {
   elasticsearch{
      hosts => "<节点的内网访问地址和端口号>"}
}
```

- input：指明了数据的来源。实际请根据用户的具体情况来设置。input 参数的详细解释和使用介绍请参见 https://www.elastic.co/guide/en/logstash/current/input-plugins.html。
- filter：对日志进行了提取和处理，将非结构化信息转换为结构化信息。filter 参数的详细解释和使用介绍请参见 https://www.elastic.co/guide/en/logstash/current/filter-plugins.html。
- output：指明了数据的目的地址。output 参数的详细解释和使用介绍请参见 https://www.elastic.co/guide/en/logstash/current/output-plugins.html。<*节点的内网访问地址和端口号*>为集群中节点的内网访问地址和端口号。

  当集群包含多个节点时，为了避免节点故障，建议将上述命令中<*节点的内网访问地址和端口号*>替换为该集群中多个节点的内网访问地址和端口号，多个节点的内网访问地址和端口号之间用英文逗号隔开，填写格式请参见如下示例。

  ```
  hosts => ["192.168.0.81:9200","192.168.0.24:9200"]
  ```

  当集群只包含一个节点时，填写格式请参见如下示例。

  ```
  hosts => "192.168.0.81:9200"
  ```

以步骤 2 中“/tmp/access_log/”的数据文件为例，输入数据文件从首行开始，且过滤条件保持为空，即不做任何数据处理操作。需导入数据的集群，其节点内网访问地址和端口号为“192.168.0.81:9200”。导入数据的索引名称为“myindex”。配置文件的示例如下所示，配置文件按实际数据情况修改完成后，输入“:wq”保存。

```
input {
   file{
     path => "/tmp/access_log/*"
     start_position => "beginning"
   }
}
filter {
}
output {
   elasticsearch {
     hosts => "192.168.0.81:9200"
     index => "myindex"
     document_type => "mytype"
```

```
    }
}
```

如果集群开启了安全模式，则需要先下载证书。

a. 在集群基本信息页面下载证书。

图5-4 下载证书

b. 将下载的证书存放到部署 logstash 服务器中。

c. 修改配置文件 logstash-simple.conf。

以步骤 2 中“/tmp/access_log/”的数据文件为例，输入数据文件从首行开始，且过滤条件保持为空，即不做任何数据处理操作。跳转主机的公网 IP 和端口号为“192.168.0.227:9200”。导入数据的索引名称为“myindex”，证书存放路径为“/logstash/logstash6.8/config/CloudSearchService.cer”。配置文件的示例如下所示，配置文件按实际数据情况修改完成后，输入“:wq”保存。

```
input{
    file {
        path => "/tmp/access_log/*"
        start_position => "beginning"
    }
}
filter {
    }
output{
    elasticsearch{
        hosts => ["https://192.168.0.227:9200"]
        index => "myindex"
        user => "admin"
        password => "******"
        cacert => "/logstash/logstash6.8/config/CloudSearchService.cer"
    }
}
```

说明

password：登录安全集群的密码。

4. 执行如下命令将 Logstash 收集的弹性云服务器的数据导入到集群中。

```
./bin/logstash -f logstash-simple.conf
```

5. 登录云搜索服务管理控制台。
6. 在左侧导航栏中，选择“集群管理”，进入集群列表页面。
7. 在集群列表页面中，单击待导入数据的集群“操作”列的“Kibana”。
8. 在 Kibana 的左侧导航中选择“Dev Tools”，单击“Get to work”，进入 Console 界面。
9. 在已打开的 Kibana 的 Console 界面，通过搜索获取已导入的数据。

   在 Kibana 控制台，输入如下命令，搜索数据。查看搜索结果，如果数据与导入数据一致，表示数据文件的数据已导入成功。

```
GET myindex/_search
```

5.2 使用 Kibana 或 API 导入数据到 Elasticsearch

云搜索服务支持使用 Kibana 或者 API 将数据导入到 Elasticsearch 中，数据文件支持 JSON、CSV 等格式。

使用 Kibana 导入数据

在导入数据之前，您可以使用 Kibana 接入集群。如下操作步骤介绍如何使用 POST 命令导入数据。

1. 登录 Kibana Console 页面，详细操作请参见在管理控制台通过 Kibana 接入集群。

   首次登录时，需要在 Kibana 的左侧导航中选择“Dev Tools”，单击“Get to work”，进入 Console 界面。非首次登录可单击“Dev Tools”直接进入 Kibana Console 页面。
2. （可选）在 Console 界面，执行命令创建待存储数据的索引，并指定自定义映射来定义数据类型。

   如果待导入数据的集群已存在可用的索引，则不需要再创建索引；如果待导入数据的集群不存在可用的索引，则需要参考如下示例创建索引。

   例如：在 Console 界面，执行如下命令，创建索引“my_store”，并指定自定义映射来定义数据类型。

   7.x 之前版本

```
PUT /my_store
{
   "settings": {
      "number_of_shards": 1
   },
   "mappings": {
      "products": {
         "properties": {
            "productName": {
               "type": "text"
            },
            "size": {
```

```
                "type": "keyword"
            }
        }
    }
}
```

7.x 之后版本

```
PUT /my_store
{
    "settings": {
        "number_of_shards": 1
    },
    "mappings": {
        "properties": {
            "productName": {
                "type": "text"
            },
            "size": {
                "type": "keyword"
            }
        }
    }
}
```

3. 在 Console 界面的右侧文本框中输入要导入数据的 POST 命令，以导入一条数据为例，执行如下命令。

   7.x 之前版本

   ```
   POST /my_store/products/_bulk
   {"index":{}}
   {"productName":"Latest art shirts for women in 2017 autumn","size":"L"}
   ```

   7.x 之后版本

   ```
   POST /my_store/_bulk
   {"index":{}}
   {"productName":"Latest art shirts for women in 2017 autumn","size":"L"}
   ```

   返回结果如图 5-5 所示，当返回结果信息中“errors”字段的值为“false”时，表示导入数据成功。

图5-5 返回消息

使用 API 导入数据

使用 bulk API 通过 cURL 命令导入数据文件，如下操作以 JSON 数据文件为例。

说明

使用 API 导入数据文件时，建议导入的数据文件大小不能超过 50MB。

1. 登录即将接入集群的弹性云服务器。

   接入集群的详细操作指导请参见在同一 VPC 内的弹性云服务器，直接调用 Elasticsearch API。

2. 执行如下命令，导入 JSON 数据。

   其中，***{Private network address and port number of the node}***需替换为集群中节点的内网访问地址和端口号，当该节点出现故障时，将导致命令执行失败。如果集群包含多个节点，可以将***{Private network address and port number of the node}***替换为集群中另一节点的内网访问地址和端口号；如果集群只包含一个节点，则需要将该节点修复之后再次执行命令进行导入数据。**test.json** 为导入数据的 json 文件。

   ```
   curl -X PUT "http://{Private network address and port number of the node}
   /_bulk" -H 'Content-Type: application/json' --data-binary @test.json
   ```

说明

其中，-X 参数的参数值为命令，如“-X PUT”，-H 参数的参数值为消息头，如“-H 'Content-Type: application/json' --data-binary @test.json”。添加的-k 参数时，请勿将-k 参数放置在参数与参数值之间。

示例：将“testdata.json”数据文件中的数据导入至 Elasticsearch 集群，此集群未进行通信加密，其中一个节点内网访问地址为“192.168.0.90”，端口号为“9200”。其中 testdata.json 文件中的数据如下所示：

```
{"index": {"_index":"my_store","_type":"products"}}
{"productName": "2019 秋装新款文艺衬衫女装","size": "M"}
{"index": {"_index":"my_store","_type":"products"}}
{"productName": "2019 秋装新款文艺衬衫女装","size": "L"}
```

导入数据的操作步骤如下所示：

a. 可执行以下命令，创建 my_store 索引。

7.x 之前版本

```
curl -X PUT http://192.168.0.90:9200/my_store -H 'Content-Type:
application/json' -d '
 {
   "settings": {
     "number_of_shards": 1
   },
   "mappings": {
     "products": {
       "properties": {
         "productName": {
           "type": "text"
           },
         "size": {
           "type": "keyword"
         }
       }
     }
   }
 }'
```

7.x 之后版本

```
curl -X PUT http://192.168.0.90:9200/my_store -H 'Content-Type:
application/json' -d '
{
    "settings": {
        "number_of_shards": 1
    },
    "mappings": {
        "properties": {
            "productName": {
                "type": "text"
            },
            "size": {
                "type": "keyword"
            }
        }
    }
}'
```

b. 执行以下命令，导入 testdata.json 文件中的数据。

```
curl -X PUT "http://192.168.0.90:9200/_bulk" -H 'Content-Type:
application/json' --data-binary @testdata.json
```

6 Elasticsearch 使用建议

Elasticsearch 是开源搜索引擎，在深入使用 Elasticsearch 搜索引擎过程中，积累了一些经验和技巧，建议用户在使用云搜索服务时，作为参考。

提高索引效率

- **使用多进程或多线程发送数据到 Elasticsearch**

  一个单线程发送 bulk 请求不能够发挥一个集群的索引能力。为了更好地利用集群的资源，应该使用多线程或多进程来发送数据，提升数据处理效率。

  对于相同大小的 bulk 请求，通过测试可以得到最优的线程数量。可以逐步增加线程数量直至到集群中的机器 Load 或 CPU 饱和。建议使用“nodes stats”接口查看节点中的 cpu 和 load 状态，您可以通过“os.cpu.percent”、“os.cpu.load_average.1m”、“os.cpu.load_average.5m”和“os.cpu.load_average.15m”参数信息了解详细信息。“nodes stats”接口的使用指导和参数解释请参见：https://www.elastic.co/guide/en/elasticsearch/reference/6.2/cluster-nodes-stats.html#os-stats

  例如，在执行 bulk 请求时，使用的线程数量为 2 个，观察 Load 和 CPU 的情况，如果未饱和，可再增加线程数量。当线程数量增加到 N 个时，此时 Load 和 CPU 已饱和，建议就采用 N 个线程去执行 bulk 请求提高索引效率。通过测试获得最优的线程数量。

- **增加 refresh_interval 刷新的间隔时间**

  默认情况下，每个分片每秒自动刷新一次。但并不是所有场景都需要每秒刷新。在使用 Elasticsearch 索引大量的日志文件，想优化索引速度而不是近实时搜索，可以通过设置，降低每个索引的刷新频率。

```
PUT /my_logs
{
  "settings": {
    "refresh_interval": "30s"
  }
}
```

- **在初始化索引时，可以禁用 refresh 和 replicas 数量**

  如果需要一次导入较大数据量的数据进 index 里面时，可以先禁用 refresh，把“refresh_interval”设置成为“-1”，把“number_of_replicas”设置成“0”。当数据导入完成后，将 refresh_interval 和 number_of_replicas 设置回原来的值。

选择合适的分片数和副本数

在创建索引数据时，建议指定相关的分片数和副本数，否则会使用服务器中的默认配置参数“shards=5, replicas=1”，即分片数为 5，副本数为 1。

分片数，与检索速度非常相关的的指标，如果分片数过少或过多都会导致检索比较慢。分片数过多会导致检索时打开比较多的文件，且会导致多台服务器之间通讯慢。而分片数过少会导致单个分片索引过大，所以检索速度慢。

根据机器数、磁盘数、索引大小等设置分片数，建议单个分片不要超过 30GB。总数据量除以分片数，则为分片的大小。

```
PUT /my_index
{
  "settings": {
    "number_of_shards":   1,
    "number_of_replicas":  0
  }
}
```

将数据存放在不同的索引

Elasticsearch 是基于 Lucene 进行索引和存储数据的，主要的工作方式是密集的数据，即所有的 document 拥有相同的字段。

- **避免把无关联的数据放在同一个 index**

  不要把完全不同的数据结构 document 放在同一个 index 里。可以考虑创建一些较小的 index，用较少的 shard 去存储。

- **避免不同的 type 放在同一个 index**

  多个 type 放在单个 index 看起来是个简单的方法，但是 Elasticsearch 并不是基于 type 来存储的，不同的 type 在单个 index 会影响效率。如果 type 没有非常相似的 mapping，建议放到一个单独的 index。

- **同一个 index 里面不同 type 之间字段不能冲突**

  如果有两个不同的 type，每个 type 都有同名的字段，但映射不同，这在 Elasticsearch 是不允许的。

按照时间范围创建索引

在 Elasticsearch 用于存储跟时间相关的数据时，如日志数据，建议按照时间范围创建索引，而不是把所有数据都存放到一个超级大的索引里面。

基于时间范围索引。可以开始于一个按年的索引（logs_2014）或按月的索引（logs_2014-10）。当数据量变得非常庞大的时候切换到一个按天的索引（logs_2014-10-24）。

按照时间范围创建索引具有如下优势：

- **扩容的时候根据当前数据量选择合适的 shard 和 Replica**

  针对时间范围创建的每个索引都可以灵活的设置 Shard 数和 Replica 数，从而可以避免在一开始设置一个很大的 shard 来考虑扩容的情况。在集群扩容之后也可以方便的设置时间范围周期来适配集群规模。

- **删除旧数据只需要删除旧的索引**

```
DELETE /logs_2014-09
```

- **利用 alias 机制可以在索引间灵活切换**

例如，将 logs_current 的 alias 机制中的 logs_2014-09 索引删除，并在此 alias 机制中新增 logs_2014-10 索引。

```
POST /_aliases
{
  "actions": [
    { "add":    { "alias": "logs current",  "index": "logs 2014-10" }},
    { "remove": { "alias": "logs current",  "index": "logs 2014-09" }}
  ]
}
```

- **针对不再更新的索引，如上周或者上月的索引，进行索引优化以提高查询效率**

将 logs_2014-09-30 索引下多个小 segment 合并成一个大的分片，以提高查询效率。

7.x 之前版本

```
PUT /logs 2014-09-30/ settings
{ "number of replicas": 0 }

POST /logs 2014-09-30/ forcemerge?max num segments=1

PUT /logs_2014-09-30/_settings
{ "number_of_replicas": 1 }
```

7.x 之后版本

```
PUT /logs_2014-09-30/_settings
{ "number_of_replicas": 0 }

POST /logs_2014-09-30/_forcemerge
{
  "max_num_segments":1
}

PUT /logs_2014-09-30/_settings
{ "number_of_replicas": 1 }
```

优化索引配置

- **区分 text 和 keyword**

在 Elasticsearch 中 string 字段被拆分成两种新的数据类型：text 用于全文搜索的，而 keyword 用于关键词搜索。

对于不需要分词的字符串精确值字段，如标签或枚举，建议配置为 keyword 类型。

7.x 之前版本

```
PUT my_index1
{
  "mappings": {
    "my_type": {
      "properties": {
        "tags": {
```

```
        "type":  "keyword"
      },
      "full_name": {
        "type":  "text"
      }
    }
  }
}
```

7.x 之后版本

```
PUT my_index1
{
  "mappings": {
        "properties": {
      "tags": {
        "type":  "keyword"
      },
      "full_name": {
        "type":  "text"
      }
    }
  }
}
```

- **基于 text 字段的聚合统计**

  分词字段的聚合统计不是一种常见的需求。在 Elasticsearch 对于分词字段的聚合统计需要用到 fielddata，默认是禁用的，开启 fielddata 会带来较大的内存负担。

  建议的做法是分词字符串进行多字段映射，映射为一个 text 字段用于全文检索，和一个 keyword 字段用于聚合统计。

  7.x 之前版本

```
PUT my_index2
{
  "mappings": {
    "my_type": {
      "properties": {
        "full_name": {
          "type": "text",
          "fields": {
            "raw": {
              "type":  "keyword"
            }
          }
        }
      }
    }
  }
}
```

  7.x 之后版本

```
PUT my_index2
{
    "mappings": {
          "properties": {
```

```
            "full_name": {
               "type": "text",
               "fields": {
                  "raw": {
                     "type": "keyword"
                  }
               }
            }
         }
      }
}
```

使用索引模板

Elasticsearch 支持通过索引模板控制一些新建索引的设置（settings）和映射（mappings），如限制分片数为 1，并且禁用_all 域。索引模板可以用于控制何种设置（settings）应当被应用于新创建的索引：

- 索引模板可以通过 template 字段指定通配符。
- 多个索引模板可以通过 order 指定覆盖顺序。数值越大，优先级越高。

如下示例表示，logstash-*匹配的索引采用 my_logs 模板，且 my_logs 模板的优先级数值为 1。

7.x 之前版本

```
PUT /_template/my_logs
{
  "template": "logstash-*",
  "order":    1,
  "settings": {
    "number_of_shards": 1
  },
  "mappings": {
    "_default_": {
      "_all": {
        "enabled": false
      }
    }
  },
  "aliases": {
    "last_3_months": {}
  }
}
```

7.x 之后版本

```
PUT /_template/my_logsa
{
  "index_patterns": ["logstasaah-*"],
  "order": 1,
  "settings": {
    "number_of_shards": 1
  },
  "mappings": {
    "properties": {
```

```
    "_all": {
      "enabled": false
    }
   }
  },
  "aliases": {
   "last_3_months": {}
  }
}
```

数据备份和恢复

Elasticsearch 副本提供了高可靠性，让您可以容忍零星的节点丢失而不会中断服务。

但是，副本并不提供对灾难性故障的保护。对这种情况，您需要的是对集群真正的备份，在某些东西确实出问题的时候有一个完整的拷贝。

备份集群，您可以使用创建快照的功能，将集群的数据保存到 OBS 桶中。其备份过程是智能的。第一个快照建议是数据的完整拷贝，后续的快照会保留的是已存快照和新数据之间的差异。随着您不时的对数据进行快照，备份也在增量的添加和删除。这意味着后续备份会相当快速，因为它们只传输很小的数据量。

用过滤提高查询效率

过滤器的执行速度非常快，不会计算相关度（直接跳过了整个评分阶段），而且很容易被缓存。

通常当查找一个精确值的时候，我们不希望对查询进行评分计算。只希望对文档进行包括或排除的计算，所以我们会使用 constant_score 查询以非评分模式来执行 term 查询并以一作为统一评分。

```
GET /my_store/products/_search
{
   "query" : {
      "constant_score" : {
         "filter" : {
            "term" : {
               "city" : "London"
            }
         }
      }
   }
}
```

采用 scroll API 返回大量数据

当返回大量数据时，先查后取的过程支持用 from 和 size 参数分页，但有限制。结果集在返回之前需要在每个分片上先进行排序，然后合并之后再排序输出。使用足够大的 from 值，排序过程可能会变得非常沉重，使用大量的 CPU、内存和带宽。因此，强烈建议不要使用深分页。

为了避免深度翻页，推荐采用 scroll 查询返回大量数据。

scroll 查询可以用来对 Elasticsearch 有效地执行大批量的文档查询，而又不用付出深度分页那种代价。scroll 查询允许我们先做查询初始化，然后再批量地拉取结果。

查询（query）与过滤（filter）的区别

性能差异：一般情况下，一次过滤会比一次评分的查询性能更优异，并且表现更稳定。

当使用过滤情况时，查询被设置为一个“不评分”或“过滤”查询。即这个查询只是去判断是否匹配，结果是 yes 或 no。

例如，以下情况是典型的过滤情况：

- created 时间是否在 2013 与 2014 这个区间？
- status 字段是否包含 published 这个单词？
- lat_lon 字段表示的位置是否在指定点的 10km 范围内？

当使用查询情况时，查询会变成一个“评分”的查询。和不评分的查询类似，也要去判断这个文档是否匹配，同时还需要判断这个文档匹配的程度如何。此查询的典型用法是用于查找以下文档：

- 查找与 full text search 这个词语匹配的文档。
- 包含 run 这个词，也能匹配 runs、running、jog 或者 sprint。
- 包含 quick、brown 和 fox 这几个词，词之间离的越近，文档相关性越高。
- 标有 lucene、search 或 java 标签，标签越多，相关性越高。

验证查询是否合法

查询在和不同的分析器与不同的字段映射相结合时，会比较难理解，可以用 **validate-query API** 来验证查询是否合法。

示例：在 Kibana 的 Console 界面中，执行如下命令。validate 请求会告诉您这个查询不合法。

7.x 之前版本

```
GET /gb/tweet/_validate/query
{
 "query": {
 "tweet" : {
   "match" : "really powerful"
  }
}
}
```

7.x 之后版本

```
GET /gb/tweet/_validate/query
{
"query": {
   "productName" : {
  "match" : "really powerful"
  }
```

```
  }
 }
```

为了找出查询不合法的原因，可以把 explain 参数加到查询字符串中，执行如下命令。

7.x 之前版本

```
GET /gb/tweet/_validate/query?explain
{
"query": {
   "tweet" : {
  "match" : "really powerful"
  }
  }
 }
```

7.x 之后版本

```
GET /gb/tweet/_validate/query?explain
{
 "query": {
 "productName" : {
   "match" : "really powerful"
  }
}
}
```

返回结果如下所示，可以从返回结果看出查询类型（match）与字段名称（tweet）搞混了。

```
{
  "valid": false,
  "error": "org.elasticsearch.common.ParsingException: no [query] registered for
[tweet]"
}
```

因此，对于合法查询，使用 explain 参数将返回可读的描述，这对准确理解云搜索服务是如何解析 query 是非常有帮助的。

7 自定义词库

7.1 使用示例

分词器说明

在 Elasticsearch 搜索引擎中使用分词词库时，有两种用法：

- ik_max_word：会将文本做最细粒度的拆分，比如会将“昨夜西风吹折千林梢”拆分为“昨夜西风,昨夜,西风,吹折千林梢,吹折,千林梢,千,林,折千林,千林,吹”，会穷尽各种可能的组合。
- ik_smart：会做最粗粒度的拆分，比如会将“昨夜西风吹折千林梢”拆分为“昨夜西风,吹折千林梢”。

示例指导

假设，用户将“智能手机”设置为主词，“是”设置为停词。然后，使用 Elasticsearch 引擎对“智能手机是很好用”文本内容进行分词。

1. 在云搜索服务管理控制台，选择需要使用的集群，单击集群名称进入集群“基本信息”页面。
2. 准备主词库文件、停词词库文件和同义词词库文件，主词库文件中包含“智能手机”词语，停词文件中包含“是”词语，同义词词库文件中包含一组同义词：“开心”和“高兴”。然后将 UTF-8 无 BOM 格式编码的文件存储到对应 OBS 路径下，如“obs-b8ed”桶。

   说明

   由于系统默认词库的停用词包含了“是”、“的”等常用词，此处您也可以不上传示例中的停用词。

3. 参考配置自定义词库步骤描述，选择对应 OBS 路径，并选择对应的“主词库文件”、“停词词库文件”和“同义词词库文件”。单击“保存”，等待词库配置信息生效。
4. 待词库状态变更为“成功”后，进入集群管理页面，在对应的集群列，单击“操作”列的“Kibana”接入集群。
5. 在 Dev Tools 的 Console 中，输入如下代码，并单击  运行，您可以在右侧查看分词效果。

- 使用 ik_smart 分词器，对“智能手机是很好用”文本内容进行分词。

  示例代码：

```
POST /_analyze
{
  "analyzer":"ik_smart",
  "text":"智能手机是很好用"
}
```

  运行结束后，查看分词效果：

```
{
  "tokens": [
    {
      "token": "智能手机",
      "start_offset": 0,
      "end_offset": 4,
      "type": "CN_WORD",
      "position": 0
    },
    {
      "token": "很好用",
      "start offset": 5,
      "end offset": 8,
      "type": "CN WORD",
      "position": 1
    }
  ]
}
```

- 使用 ik_max_word 分词器，对“智能手机是很好用”文本内容进行分词。

  示例代码：

```
POST /_analyze
{
  "analyzer":"ik_max_word",
  "text":"智能手机是很好用"
}
```

  运行结束后，查看分词效果：

```
{
  "tokens" : [
    {
      "token" : "智能手机",
      "start_offset" : 0,
      "end_offset" : 4,
      "type" : "CN_WORD",
      "position" : 0
    },
    {
      "token" : "智能",
      "start_offset" : 0,
      "end_offset" : 2,
      "type" : "CN_WORD",
      "position" : 1
    },
    {
      "token" : "智",
```

```
    "start_offset" : 0,
    "end_offset" : 1,
    "type" : "CN_WORD",
    "position" : 2
  },
  {
    "token" : "能手",
    "start_offset" : 1,
    "end_offset" : 3,
    "type" : "CN_WORD",
    "position" : 3
  },
  {
    "token" : "手机",
    "start_offset" : 2,
    "end_offset" : 4,
    "type" : "CN_WORD",
    "position" : 4
  },
  {
    "token" : "机",
    "start_offset" : 3,
    "end_offset" : 4,
    "type" : "CN_WORD",
    "position" : 5
  },
  {
    "token" : "很好用",
    "start_offset" : 5,
    "end_offset" : 8,
    "type" : "CN_WORD",
    "position" : 6
  },
  {
    "token" : "很好",
    "start_offset" : 5,
    "end_offset" : 7,
    "type" : "CN_WORD",
    "position" : 7
  },
  {
    "token" : "好用",
    "start_offset" : 6,
    "end_offset" : 8,
    "type" : "CN_WORD",
    "position" : 8
  },
  {
    "token" : "用",
    "start_offset" : 7,
    "end_offset" : 8,
    "type" : "CN_WORD",
    "position" : 9
  }
```

```
  ]
}
```

6. 参考如下示例，创建索引、导入数据、并使用关键字进行搜索和查看搜索结果。

   a. 创建索引“book”。如下所示，示例中“analyzer”和“search_analyzer”选择“ik_max_word”，您也可以选择“ik_smart”分词器。

   （7.x 之前版本）

```
PUT /book
{
   "settings": {
      "number of shards": 2,
      "number_of_replicas": 1
   },
   "mappings": {
      "type1": {
         "properties": {
            "content": {
               "type": "text",
               "analyzer": "ik_max_word",
               "search_analyzer": "ik_max_word"
            }
         }
      }
   }
}
```

   （7.x 及之后版本）

```
PUT /book
{
   "settings": {
      "number_of_shards": 2,
      "number_of_replicas": 1
   },
   "mappings": {
      "properties": {
         "content": {
            "type": "text",
            "analyzer": "ik max word",
            "search analyzer": "ik max word"
         }
      }
   }
}
```

   b. 导入数据。将文本信息导入至“book”索引中。

   （7.x 之前版本）

```
PUT /book/type1/1
{
  "content":"智能手机是很好用"
}
```

   （7.x 及之后版本）

```
PUT /book/_doc/1
{
```

```
  "content":"智能手机是很好用"
}
```

c. 使用关键字“智能手机”进行搜索，并查看搜索结果。

（7.x 之前版本）

```
GET /book/type1/_search
{
  "query": {
    "match": {
      "content": "智能手机"
    }
  }
}
```

（7.x 及之后版本）

```
GET /book/_doc/_search
{
  "query": {
    "match": {
      "content": "智能手机"
    }
  }
}
```

搜索结果：

（7.x 之前版本）

```
{
  "took" : 12,
  "timed_out" : false,
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : 1,
    "max_score" : 1.7260926,
    "hits" : [
      {
        "_index" : "book",
        "_type" : "type1",
        "_id" : "1",
        "_score" : 1.7260926,
        "_source" : {
          "content" : "智能手机是很好用"
        }
      }
    ]
  }
}
```

（7.x 及之后版本）

```
{
  "took" : 16,
```

```
  "timed_out" : false,
  "_shards" : {
    "total" : 2,
    "successful" : 2,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 1.7260926,
    "hits" : [
      {
        "_index" : "book",
        "_type" : "_doc",
        "_id" : "1",
        "_score" : 1.7260926,
        "_source" : {
          "content" : "智能手机是很好用"
        }
      }
    ]
  }
}
```

7. 参考如下示例，创建索引、导入数据、并使用同义词进行搜索查看搜索结果。

   a. 创建索引。

   （7.x 之前版本如下所示）

```
PUT myindex
{
  "settings": {
    "analysis": {
      "filter": {
        "my_synonym": {
          "type": "dynamic_synonym"
        }
      },
      "analyzer": {
        "ik_synonym": {
          "filter": [
            "my_synonym"
          ],
          "type": "custom",
          "tokenizer": "ik_smart"
        }
      }
    }
  },
  "mappings": {
    "mytype" :{
      "properties": {
        "desc": {
          "type": "text",
```

```
        "analyzer": "ik_synonym"
      }
     }
   }
  }
}
```

（7.x 及之后版本如下所示）

```
PUT myindex
{
   "settings": {
      "analysis": {
         "filter": {
            "my_synonym": {
               "type": "dynamic_synonym"
            }
         },
         "analyzer": {
            "ik_synonym": {
               "filter": [
                  "my_synonym"
               ],
               "type": "custom",
               "tokenizer": "ik_smart"
            }
         }
      }
   },
   "mappings": {
      "properties": {
         "desc": {
            "type": "text",
            "analyzer": "ik synonym"
         }
      }
   }
}
```

b. 导入数据。将文本信息导入至“myindex”索引中。

（7.x 之前版本）

```
PUT /myindex/mytype/1
{
   "desc": "我今天获奖了我很开心"
}
```

（7.x 及之后版本）

```
PUT /myindex/_doc/1
{
   "desc": "我今天获奖了我很开心"
}
```

c. 使用同义词“高兴”进行搜索，并查看搜索结果。

执行如下命令搜索“高兴”。

```
GET /myindex/_search
{
```

```
  "query": {
    "match": {
      "desc": "高兴"
    }
  }
}
```

搜索结果：

（7.x 之前版本）

```
{
  "took": 12,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 0.41048482,
    "hits": [
      {
        "_index": "myindex",
        " type": "mytype",
        " id": "1",
        " score": 0.41048482,
        " source": {
          "desc": "我今天获奖了我很开心"
        }
      }
    ]
  }
}
```

（7.x 及之后版本）

```
{
  "took" : 1,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 1,
      "relation" : "eq"
    },
    "max_score" : 0.1519955,
    "hits" : [
      {
        " index" : "myindex",
        " type" : " doc",
        "_id" : "1",
```

```
      "_score" : 0.1519955,
      "_source" : {
        "desc" : "我今天获奖了我很开心"
      }
    }
  ]
 }
}
```

8 简繁体转换插件

默认情况下云搜索服务安装了简繁体转换插件，用户无需自行安装。简繁体转换插件是一款可以使中文简体和中文繁体相互转换的插件。通过该插件的转换，用户可以使用中文繁体关键字搜索出包含对应中文简体的索引数据，也可以使用中文简体关键字搜索出包含对应中文繁体的索引数据。

简繁体转换插件通常可以当做 analyzer、tokenizer、token-filter 或 char-filter 来使用。

简繁体转换插件的转换类型包含如下两种：

- s2t：将中文简体转换为中文繁体。
- t2s：将中文繁体转换为中文简体。

示例指导

1. 登录云搜索服务管理控制台。
2. 在左侧导航栏中，选择“集群管理”，进入集群列表页面。
3. 在集群列表中，单击需要使用的集群对应“操作”列的“Kibana”。

   如果开启了安全模式，需要输入创建集群时设置的用户名和密码。
4. 在 Kibana 的左侧导航中选择“Dev Tools”，单击“Get to work”，进入 Console 界面。
5. 在 Console 界面，执行如下命令，创建索引“stconvert”，并指定自定义映射来定义数据类型。

   7.x 之前版本

```
PUT /stconvert
{
   "settings": {
      "number_of_shards": 1,
      "number_of_replicas": 0,
      "analysis": {
         "analyzer": {
            "ts_ik": {
               "tokenizer": "ik_smart",
               "char_filter": [
                  "tsconvert",
                  "stconvert"
               ]
```

```
            }
        },
        "char_filter": {
            "tsconvert": {
                "type": "stconvert",
                "convert_type": "t2s"
            },
            "stconvert": {
                "type": "stconvert",
                "convert_type": "s2t"
            }
        }
    },
    "mappings": {
        "type": {
            "properties": {
                "desc": {
                    "type": "text",
                    "analyzer": "ts_ik"
                }
            }
        }
    }
}
```

7.x 之后版本

```
PUT /stconvert
{
    "settings": {
        "number of shards": 1,
        "number of replicas": 0,
        "analysis": {
            "analyzer": {
                "ts_ik": {
                    "tokenizer": "ik_smart",
                    "char_filter": [
                        "tsconvert",
                        "stconvert"
                    ]
                }
            },
            "char_filter": {
                "tsconvert": {
                    "type": "stconvert",
                    "convert_type": "t2s"
                },
                "stconvert": {
                    "type": "stconvert",
                    "convert_type": "s2t"
                }
            }
        }
    },
    "mappings": {
              "properties": {
```

```
            "desc": {
               "type": "text",
               "analyzer": "ts_ik"
            }
         }
      }
}
```

返回结果如下所示。

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : "stconvert"
}
```

6. 在 Console 界面，执行如下命令，导入数据到“stconvert”索引中。

   7.x 之前版本

```
POST /stconvert/type/1
{
   "desc": "國際電視臺"
}
```

   7.x 之后版本

```
POST /stconvert/_doc/1
{
   "desc": "國際電視臺"
}
```

   当返回结果信息中“failed”字段的值为“0”时，表示数据导入成功。

7. 在 Console 界面，执行如下命令，搜索关键字“国际”，并查看搜索结果。

```
GET /stconvert/_search
{
   "query": {
      "match": {
         "desc": "国际"
      }
   }
}
```

   搜索结果如下所示。

```
{
  "took" : 15,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : 1,
    "max_score" : 0.5753642,
    "hits" : [
      {
        "_index" : "stconvert",
        "_type" : "type",
```

```
      "_id" : "1",
      "_score" : 0.5753642,
      "_source" : {
        "desc" : "國際電視臺"
      }
    }
  ]
 }
}
```

8.1 配置自定义词库

用户在使用搜索引擎时，针对中文，一些特殊的词语在分词的时候也能够被识别。

例如，想根据公司名称来查询；或者想根据网络上的某个流行词来查询，如“喜大普奔”。

针对如上场景，您可以使用云搜索服务的自定义词库功能完成分词。且支持热更新，不需要重启集群即可生效。

说明

自定义词库功能上线之前（即 2018 年 3 月 10 日之前）创建的集群，无法使用自定义词库功能。

基本概念

- 主词库：主词为用户希望进行分词的特殊词语，如上文场景中的“智能手机”和“喜大普奔”。主词库则是这些特殊词语的集合。主词库文件必须是 UTF-8 无 BOM 格式编码的文本文件，且文件中每一行为一个分词。主词库文件最大支持 100M。
- 停词词库：停词为用户不希望进行分词或者关注的词语，如“的”、“什么”、“怎么”等。停词词库是停词词语的集合。停词词库文件必须是 UTF-8 无 BOM 格式编码的文本文件，且文件中每一行为一个分词。停词词库文件最大支持 20M。
- 同义词词库：同义词为意义相同的一组词语，如“开心”和“高兴”。同义词词库是同义词词语的集合。同义词词库文件必须是 UTF-8 无 BOM 格式编码的文本文件，且文件中每一行为一组同义词，同义词之间用英文逗号隔开。同义词词库文件最大支持 20MB。

前提条件

登录云搜索服务管理控制台的账号或 IAM 用户必须同时具备如下两个权限才能使用自定义词库功能。

- “全局服务”中“对象存储服务”项目的“OBS Administrator”权限。
- 当前所属区域的“Elasticsearch Administrator”权限。

配置自定义词库

1. 在云搜索服务管理控制台，单击左侧导航栏的“集群管理”。
2. 在“集群管理”页面，单击需要配置自定义词库的集群名称，进入集群信息页面。
3. 在集群信息页面，单击“自定义词库”页签。
4. 在“自定义词库”页面，您可以单击开关来开启或关闭自定义词库功能。
   - OBS 桶：主词库、停词词库和同义词词库文件存储的 OBS 桶位置。如果当前没有可用 OBS 桶，您可以单击“创建桶”进行创建，详细操作步骤请参见《对象存储服务用户指南》。创建的 OBS 桶必须和集群所在“区域”相同。
   - 主词库对象：主词库的文件，当前只支持编码为 UTF-8 无 BOM 格式编码的文本文件。必须先把文件存储到对应 OBS 路径下。

- 停词词库对象：停词词库的文件，当前只支持 UTF-8 无 BOM 格式编码的文本文件。必须先把文件存储到对应 OBS 路径下。
- 同义词词库对象：同义词词库的文件，当前只支持 UTF-8 无 BOM 格式编码的文本文件。必须先把文件存储到对应 OBS 路径下。

5. 单击“保存”，在“确认”对话框中，单击“确定”。词库信息在下方呈现，此时词库状态为“更新中”。请耐心等待 1 分钟左右，当词库配置完成后，词库状态变更为“成功”，此时，配置的词库信息已在此集群中生效。

修改自定义词库

如果您需要修改已配置的自定义词库的参数，您可以修改自定义词库。请提前将词库文件上传至对应的 OBS 桶中。

在“自定义词库”页面，直接修改“OBS 桶”、“主词库对象”、“停词词库对象”或“同义词词库对象”参数，单击“保存”。在弹出的提示框中单击“确定”。当词库修改完成后，词库状态变更为“成功”。

删除自定义词库

如果您的词库已不再需要，您可以删除词库释放资源。

在自定义词库页面，单击  按钮，在弹出的提示框中单击“确定”。词库信息删除后，自定义词库界面如下所示。

9 管理 Elasticsearch 类型集群

9.1 集群状态和存储容量状态说明

在云搜索服务管理控制台中，直接展现当前云搜索服务中已有集群的状态以及集群存储容量状态。

表9-1 集群状态说明

状态	说明
可用	表示集群服务正常运行中，并为用户提供服务。

状态	说明
异常	表示集群创建失败或不可用。 如果此集群处于“不可用”状态，支持删除集群操作或将集群正常状态时创建的快照恢复至其他集群。无法执行扩容集群、访问 Kibana、创建快照或将快照恢复至此集群的操作。建议不要执行导入数据的操作，避免数据丢失。您可以查看监控或重启集群，但根据集群故障情况不同这些操作可能执行失败，当执行失败时，请及时联系管理员。
处理中	表示集群正处在重启中、扩容中、备份中或恢复中。
创建中	表示集群正处在创建过程中。

表9-2 集群存储容量状态

状态	说明
空闲	表示集群中所有节点存储容量使用率小于 50%。
警告	表示集群中任一节点存储容量使用率大于 等于 50%，小于 80%。
危险	表示集群中任一节点存储容量使用率大于等于 80%。建议增加集群的存储容量，以便能够正常使用集群进行数据搜索或分析。
异常	表示未能查询到集群的存储容量信息。例如，集群运行故障，状态为“异常”时，此集群的存储容量状态为“异常”。

9.2 集群列表简介

集群列表显示云搜索服务所有的集群，集群数量较多时，可采用翻页显示，您可以查看任何状态下的集群。

集群列表默认按时间顺序排列，时间最近的集群显示在最前端。在集群列表的表头，您可以单击对应参数的 按钮，修改集群的排序。集群列表参数说明如表 9-3 所示。

在集群列表右上角，您可以指定集群名称或集群 ID，然后单击 进行查找。也可以单击右上角的 ，刷新集群列表。

表9-3 集群列表说明

参数	描述
名称/ID	表示集群的名称和 ID。单击集群名称可进入集群“基本信息”页面，展现了集群的基本信息。集群 ID 是系统自动生成的，是集群在服务中的唯一标示。
集群状态	展示集群当前的状态。集群状态说明请参见 9.1 集群状态和存储容量状态说明。
任务状态	展示重启集群、扩容集群、备份集群、恢复集群等任务的状态。
版本	表示此集群中 Elasticsearch 的版本号。
创建时间	表示集群的创建时间。
内网访问地址	集群的内网访问地址和端口号，您可以使用此参数接入集群。集群有多个节点时，此处显示多个节点的内网访问地址和端口号。
操作	展示集群可执行的操作入口，包含 Kibana、监控信息、更改规格、重启、删除、自定义词库、备份与恢复、集群迁移、Cerebro。当某一操作无法执行时，显示为灰色链接。

9.3 备份与恢复索引

为避免数据丢失，您可以将集群的索引数据进行备份，当数据发生丢失或者想找回某一时间段数据时，您可以通过恢复索引操作快速获得数据。索引的备份是通过创建集群快照实现。第一次备份时，建议将所有索引数据进行备份。

说明

备份与恢复索引功能上线之前（即 2018 年 3 月 10 日之前）创建的集群，无法创建快照。

- 管理自动创建快照：自动创建快照指按照设置的规则，每天在指定时间自动创建快照。您可以开启自动创建功能、设置自动创建的策略、和关闭自动创建功能。
- 手动创建快照：在任意时间，您通过手动创建快照的方式，针对当时的数据或某几个索引创建快照进行备份。
- 恢复数据：将已有的快照，通过恢复快照功能，将备份的索引数据恢复到指定的集群中。
- 删除快照：对于已失效的快照，建议删除以释放存储资源。

说明

- 创建快照之前，您需要进行基础配置，包含存储快照的 OBS 桶、快照的备份路径及安全认证使用的 IAM 委托。
- 集群快照存储的 OBS 桶，在首次设置后，不管自动创建快照还是手动创建快照，如果快照列表中已有可用的快照，则 OBS 桶将无法再变更，请谨慎选择存储 OBS 桶。

- 如果 OBS 桶已经存储了快照，OBS 无法变更，您可以使用这个方法修改：首先关闭快照功能，然后再开启快照功能，指定新的 OBS 桶。一旦关闭快照功能，之前创建的快照将无法用于恢复集群。
- 当集群处于“不可用”状态时，快照功能中，除了恢复快照功能外，其他快照信息或功能只能查看，无法进行编辑。
- 备份与恢复过程中，支持集群扩容、访问 Kibana、查看监控、删除其他快照的操作。不支持重启此集群、删除此集群、删除正在创建或恢复的快照、再次创建或恢复快照的操作。补充说明，当此集群正在进行创建快照或者恢复快照时，此时，自动创建快照任务将被取消。

前提条件

登录云搜索服务管理控制台的账号或 IAM 用户必须同时具备如下权限才能使用创建或恢复快照功能。

- “全局服务”中“对象存储服务”项目的“OBS Administrator”权限。
- 当前所属区域的“Elasticsearch Administrator”权限。

管理自动创建快照

1. 在云搜索服务管理控制台，单击左侧导航栏的“集群管理”。
2. 在“集群管理”页面，单击需要进行备份的集群名称，进入集群信息页面。在集群信息页面，单击“集群快照”页签。

   或者，在“集群管理”页面，单击对应集群“操作”列的“更多>备份与恢复”，进入“集群快照”管理页面。
3. 在“集群快照”管理页面，在“集群快照开关”右侧单击开关，打开集群快照功能。

    表示关闭集群快照功能，  表示打开集群快照功能。
4. （可选）打开集群快照功能后，配置 OBS 桶和 IAM 委托路径。

   在基础配置弹出框中，您可以在下拉框中选择您账户下已有的 OBS 桶和 IAM 委托，或者通过“创建桶”和“创建委托”链接重新创建。

表9-4 参数说明

参数	说明	注意事项
“OBS 桶”	快照存储的 OBS 桶的名称。	创建或者已存在的 OBS 桶需满足如下条件： • “存储类别”为“标准存储”。 • 创建桶时不能开启加密功能，加密桶不可用。 • “区域”须与创建的集群所在区域相同。
“备份路径”	快照在 OBS 桶中的存放路径。	备份路径配置规则： • 备份路径不能包括下列符号：\:*?"<>|

参数	说明	注意事项
		• 备份路径不能以“/”开头。 • 备份路径不能以“.”开头或结尾。 • 备份路径的总长度不能超过 1023 个字符。
“IAM 委托”	指当前账户授权云搜索服务访问或维护存储在 OBS 中数据。	创建或者已存在的 IAM 需满足如下条件： • “委托类型”选择“云服务”。 • “云服务”选择“Elasticsearch”。 • 设置当前委托具备“全局服务”中“对象存储服务”项目的“OBS Administrator”权限。

说明

如果是子账户，需要同时设置 GetBucketStoragePolicy、GetBucketLocation、ListBucket 权限，才能看到 OBS 桶。

图9-1 修改基础配置

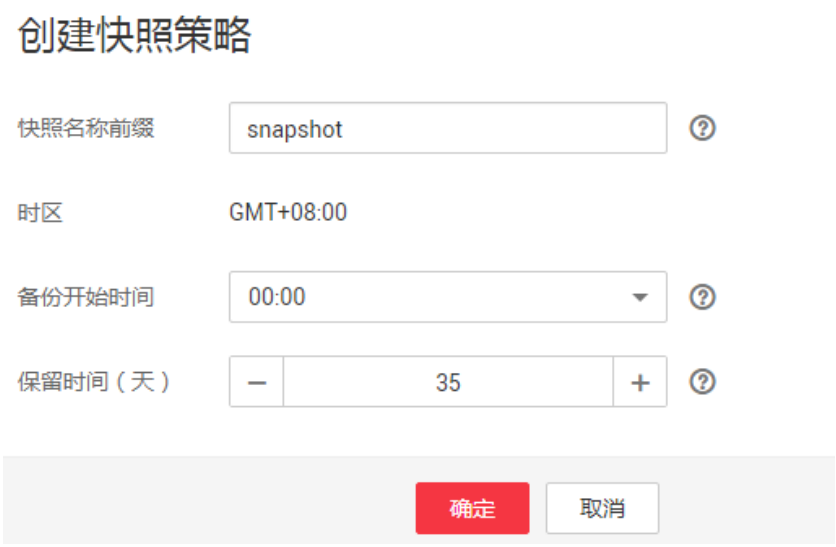
5. 在“自动创建快照”右侧，单击开关开启自动创建快照功能。

    表示打开自动创建快照功能，  表示关闭自动创建快照功能。

6. 在“创建快照策略”页面进行策略设置。
   - “快照名称前缀”：快照名称由快照名称前缀加上时间组成，例如自动生成的快照名称 snapshot-2018022405925。快照名称前缀的长度为 1～31 个字符，只能包含小写字母、数字、中划线和下划线，且必须以字母开头。
   - “时区”：指备份时间对应的时区。请基于此时区选择“备份开始时间”。

- “备份开始时间”：指每天自动开始备份的时间，只能指定整点时间，如00:00、01:00，取值范围为00:00～23:00。请在下拉框中选择备份时间。
- “保留时间（天）”：指备份的快照在OBS的保留时间，以天为单位，取值范围为1～90，您可以根据自己的需求进行设置。系统在半点时刻会自动删除超过保留时间的快照，例如：创建快照的策略设置如图9-2所示，则系统会在35天后的00:30自动删除35天前00:00自动开始备份的快照。

图9-2 自动创建快照



7. 设置完成后，单击“保存”。

自动创建策略配置完成后，策略信息将直接呈现在集群快照页面，如图9-3所示。当业务发生变更，需要修改策略时，您可以单击进行重新设置。

按照策略自动创建的快照将呈现在快照管理列表中。快照列表同时展示自动创建和手动创建的快照，您可以通过快照类型参数进行区分。在快照列表右上角，您可以输入快照名称或快照ID的关键字进行查找。您还可以通过快照列表表头的按钮根据不同参数进行排序，以便进行快速查找。

图9-3 自动创建快照策略

8. （可选）关闭自动创建快照功能。

   关闭自动创建快照功能后，系统将停止继续自动创建快照。如果系统正在根据策略自动创建快照，而快照列表还未呈现正在创建的快照时，无法关闭自动创建快照功能。如果您单击了关闭按钮，系统将提示您无法关闭。建议等快照自动创建成功后，即快照列表已出现最新创建的快照时，再单击关闭按钮，关闭自动创建快照功能。

   关闭自动创建快照功能时，您可以在弹出窗口中通过“删除自动创建的快照”选项，选择是否立即删除之前已自动创建的快照，默认不勾选。

   - 不勾选：表示不会删除关闭此功能前已自动创建的快照。如果不删除，后续还可以在快照列表中通过删除按钮手动删除，详细操作指导请参见删除快照。如果未手动删除，且之后用户又重新开启了自动创建快照功能，那么此集群中所有“快照类型”为自动创建的快照（包含开启自动创建快照功能前已存在的自动创建的快照）都无法手动删除，只会被系统自动删除。系统会基于重新开启自动创建快照功能时的配置策略进行自动删除，例如此策略中定义的保留时间为 10 天，那么系统中超过 10 天的快照将被系统自动删除。
   - 勾选：表示删除此集群快照列表中所有“快照类型”为自动创建的快照。

手动创建快照

1. 在云搜索服务管理控制台，单击左侧导航栏的“集群管理”。
2. 在“集群管理”页面，单击需要进行备份的集群名称，进入集群信息页面。在集群信息页面，单击“集群快照”页签。

   或者，在“集群管理”页面，单击对应集群“操作”列的“更多>备份与恢复”，进入“集群快照”管理页面。
3. 在“集群快照”管理页面，单击“集群快照开关”右侧的开关，打开集群快照功能。

    表示关闭集群快照功能，  表示打开集群快照功能。
4. （可选）打开集群快照功能后，配置 OBS 桶和 IAM 委托路径。基础配置的详细配置指导请参见 4。
5. 完成基础配置后，单击“创建快照”可手动创建。
   - “快照名称”：手动创建的快照名称，4～64 个字符，只能包含小写字母、数字、中划线和下划线，且必须以字母开头。与自动创建不同，手动创建的快照名称按照用户设置的名称，不会自动加上时间信息。
   - “索引”：手动创建快照可针对集群中某几个索引进行备份，此处填写索引名称。0～1024 个字符，不能包含空格和大写字母，且不能包含"\<|>/?特殊字符。多个索引之间使用英文逗号隔开。如果不填写，则默认备份集群中所有索引。支持使用“*”匹配多个索引，例如：2018-06*，表示备份名称前缀是 2018-06 的所有索引的数据。

     您可以在 Kibana 中使用 **GET /_cat/indices** 命令，查询集群存在的所有索引名称。在了解了集群的索引后，再填写需要对哪些索引数据创建快照进行备份。
   - “快照描述”：创建的快照描述信息。0～256 个字符，不能包含“<>”字符。

图9-4 手动创建快照

6. 单击“确定”开始创建快照。

   快照创建完成后，将直接呈现在快照管理列表中，快照状态为“可用”表示快照创建成功。快照列表同时展示自动创建和手动创建的快照，您可以通过快照类型参数进行区分。在快照列表右上角，您可以输入快照名称或快照 ID 的关键字进行查找。您还可以通过快照列表表头的 按钮根据不同参数进行排序，以便进行快速查找。

恢复数据

快照管理列表中“快照状态”为“可用”的快照，您可以使用快照恢复集群中的数据。已存储快照数据可恢复至其他集群。

恢复数据将覆盖集群中当前的数据，请谨慎操作。

1. 在快照管理页面中，选择需要恢复的快照，单击“操作”列的“恢复”。
2. 在恢复弹出框中，填写如下信息。

   “索引”：指定需要进行恢复的索引名称，默认为空。如保持默认值，即不指定索引名称，则表示恢复所有的索引数据。0～1024 个字符，不能包含空格和大写字母，且不能包含"\<|>/?特殊字符。

   “索引名称匹配模式”：在恢复时，可以根据文本框中定义的过滤条件去恢复符合条件的索引，过滤条件请使用正则表达式。默认值“index_(.+)”表示所有的索引。0～1024 个字符，不能包含空格和大写字母，且不能包含"\<|>/?,特殊字符。

   “索引名称替换模式”：索引重命名的规则。默认值“restored_index_$1”表示在所有恢复的索引名称前面加上“restored_”。0～1024 个字符，不能包含空格和大写字母，且不能包含"\<|>/?,特殊字符。在设置“索引名称替换模式”时，“索引名称匹配模式”和“索引名称替换模式”需要同时设置才会生效。

“集群”：选择需要进行恢复的集群名称，可选择当前集群或者其他集群。只能选择处于“可用”状态的集群，如果快照所属的集群处于“不可用”状态，那么也无法将快照恢复到本集群。恢复到其他集群时，目标集群中的 Elasticsearch 版本不低于本集群。如果已选择其他集群，且该集群中存在同名的索引，则恢复完成后，该同名的索引中的数据将会被覆盖，请谨慎操作。

图9-5 恢复快照

3. 单击“确定”开始恢复。恢复成功，快照列表中“任务状态”将变更为“恢复成功”，索引数据将根据快照信息重新生成。

   快照列表中，“任务状态”列显示快照最近一次恢复的状态。所有最近一次恢复成功的快照才会显示为“恢复成功”。

图9-6 恢复成功

删除快照

当快照信息不需要使用时，您可以删除快照释放存储资源。当自动创建快照功能开启时，自动创建的快照无法手动删除，系统会按照设置的策略在半点时刻自动删除超过“保留时间”的快照。当自动创建快照功能关闭，且之前已自动创建的快照并未同步删除时，快照列表中自动创建的快照，可通过删除按钮手动删除。如果未手动删除，且之后用户又重新开启了自动创建快照功能，那么此集群中所有“快照类型”为自动创建的快照（包含开启自动创建快照功能前已存在的自动创建的快照）都无法手动删除，只会被系统自动删除。

说明

快照信息删除后，数据将无法恢复，请谨慎操作。

1. 在快照管理页面中，选择需要删除的快照，单击“操作”列的“删除”。

2. 在“删除快照”框中单击“确定”删除。

9.4 更改规格

当创建的集群规格不能满足业务需求时，您可以通过更改规格，对已经创建的集群规格进行修改，以提高存储和使用效率。您可以根据业务需求，对集群节点进行缩减，以使集群拥有更优的存储，降低运维成本。

扩容集群

1. 登录云搜索服务管理控制台。
2. 单击“集群管理”，进入集群列表页面，选择需要更改规格的集群，在其“操作”列中单击“更多>更改规格”。
3. 在“更改集群规格”页面，设置所需“修改后的节点数量”、“节点存储容量”。

   如果集群未启用 Master 节点或 Client 节点，此时更改集群节点数量或节点存储容量，请至少增加 1 个节点，且节点数量最大为 32 个节点。

   如果此集群已启用 Master 节点或 Client 节点，您还可以更改 Master 节点数量和 Client 节点数量或节点存储容量。请至少增加 1 个节点，且集群节点数量最大为 200 个节点，Master 节点数量最大为 9 个节点，Client 节点数量最大为 32 个节点。

说明

- 如果只新增节点数量，则新增节点的“节点规格”、“节点存储容量”与创建集群时相同。
- 如果既新增节点数量，又增加节点存储容量，则新增节点的“节点规格”与创建集群时相同，集群的所有节点的存储容量将统一更改为变更后的节点存储容量。
- 如果只更改“节点存储容量”，则集群的所有节点的存储容量将统一更改为变更后的节点存储容量。
- “节点存储容量”参数最多只能修改 6 次。
- 当集群扩容时，业务不会中断。

图9-7 更改集群规格

4. 单击“立即创建”。
5. 在“集群详情”页面，确认更改的规格后，单击“提交申请”。
6. 单击“返回集群列表”跳转到集群管理页面。“任务状态”列中显示为“扩容”，表示集群正在扩容。当集群状态变为“可用”，则表示规格更改成功。

缩容集群

1. 登录云搜索服务管理控制台。
2. 单击“集群管理”，进入集群列表页面，选择需要更改规格的集群，在其“操作”列中单击“更多>更改规格”。
3. 在“更改集群规格”页面，设置所需“修改后的节点数量”。

说明

- 每次缩容节点数要小于集群节点的一半。
- 缩容之后的节点数要大于索引副本数。
- 缩容之后的磁盘使用量要小于 80%。
- 当集群缩容时，业务不会中断。
- 缩容时需要将要下线节点的数据迁移到其他节点，这个数据迁移的超时时间为 5 小时，如果超过 5 小时数据还未迁移完，那么缩容将会失败。集群数据量较大的情况下，建议可以分多次缩容。

4. 单击“立即创建”。
5. 在“集群详情”页面，确认更改的规格后，单击“提交申请”。
6. 单击“返回集群列表”跳转到集群管理页面。“任务状态”列中显示为“缩容中”，表示集群正在缩容。当集群状态变为“可用”，则表示规格更改成功。

更改集群节点规格

说明

- 只支持拥有 3 个及以上节点的集群更改节点规格。
- 只支持集群节点规格向大更改。

- 此功能上线之前创建的集群不支持更改节点规格。
- 更改规格过程中，Kibana 不可用。
- 变更节点规格不能与“修改后的节点数量”和“节点存储容量”同时操作。
- 如果数据量比较大的情况下，更改节点规格耗时会比较长。
- 更改节点规格时，为了不中断业务，业务数据必须都有副本。

1. 登录云搜索服务管理控制台。
2. 单击“集群管理”，进入集群列表页面，选择需要更改规格的集群，在其“操作”列中单击“更多>更改规格”。
3. 在“更改集群规格”页面，设置所需的“变更后节点规格”。
4. 单击“立即创建”。
5. 在“集群详情”页面，确认更改的规格后，单击“提交申请”。
6. 单击“返回集群列表”跳转到集群管理页面。“任务状态”列中显示为“规格修改”，表示集群正在更改规格。当集群状态变为“可用”，则表示规格更改成功。

9.5 绑定企业项目

每个集群必须设置其对应的企业项目，如果不需要此参数进行区分时，您可以将集群绑定至“default”项目。对于绑定企业项目特性上线前创建的集群，集群的企业项目将被绑定至“default”项目。您可以根据实际情况修改绑定企业项目。

绑定企业项目

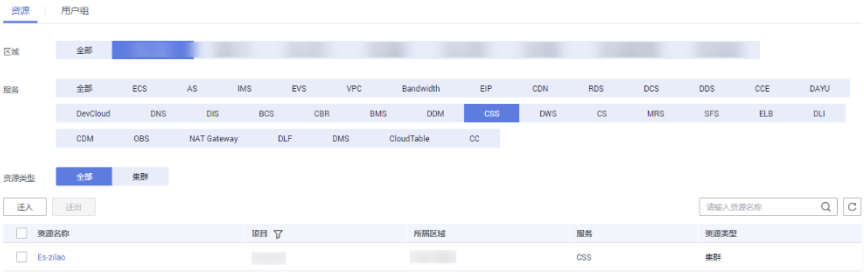
在创建集群时，您可以在企业项目参数中绑定，详细操作步骤及参数解释请参见 4.1 创建 Elasticsearch 类型集群（安全模式）。

修改企业项目

针对之前已创建的集群，其绑定的企业项目可根据实际情况进行修改。

1. 在“云搜索服务”管理控制台，单击“集群管理”进入集群列表。
2. 在集群列表中，单击集群名称进入集群“基本信息”页面。
3. 在集群“基本信息”页面中，单击“企业项目”参数右侧的参数值，进入“资源管理”的“项目管理”页面。
4. 在资源页签下，选择对应“区域”，然后在“服务”中选择“ES”服务。此时，资源列表将筛选出对应的 ES 集群。

图9-8 筛选 ES 集群



5. 勾选需要修改企业项目的集群，然后单击“迁出”。
6. 在“迁出资源”页面，选择“迁出方式”，再选择“请选择要迁入的企业项目”，然后单击“确定”。

图9-9 迁出资源

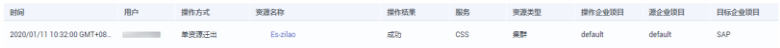
7. 迁出完成后，在原始的项目管理页面中，无法再获取此集群信息。您可以通过两种方式查看此集群绑定的企业项目。
   - 进入云搜索服务集群列表页面，此集群对应的“企业项目”列的值，将更换为修改后的企业项目。

     图9-10 查看集群对应的企业项目

   - 在资源管理服务中，在左侧导航栏选择“项目管理” ，在企业项目管理页面，单击“查看迁出迁入事件”，可获取到此集群的信息。

图9-11 查看资源



9.6 重启集群

集群停止工作时，您可通过重启集群恢复运行。处于“可用”或“异常”状态集群能执行重启操作，“处理中”或“创建中”的集群不能进行重启操作。

快速重启

- 确保集群处于“可用”或“异常”状态。
- 确认集群无执行中的任务，如导入数据、搜索数据等。

须知

- 快速重启集群过程中会导致集群不可用。如果快速重启失败，将有可能会丢失数据或导致集群不可用，请谨慎操作。
- 当集群处于可用状态时，请排查集群没有正在处理业务数据的任务，再执行快速重启操作。如果集群正在处理业务数据，如导入数据、搜索数据时，一旦快速重启集群，有可能导致数据丢失，如传输的数据丢失等。所以，建议停止所有集群任务后，再快速重启集群。

1. 登录云搜索服务管理控制台。
2. 单击“集群管理”进入集群列表界面，在对应集群的“操作”列中单击“更多>重启>快速重启”。

   快速重启过程中，集群不可用。
3. 重启集群后，请刷新页面，观察集群状态。重启过程中，集群状态为“处理中”，任务状态为“重启中”。如果集群状态变更为“可用”，表示集群已重启成功。如果集群状态变更为“异常”，建议联系管理员排查故障。

滚动重启

须知

- 滚动重启过程中可能出现数据丢失，请谨慎操作。建议您在业务空闲时操作。
- 只有当集群节点数大于等于 3 个时，才能支持滚动重启。
- 当数据量比较大时，滚动重启耗时较长。

1. 登录云搜索服务管理控制台。
2. 单击“集群管理”进入集群列表界面，在对应集群的“操作”列中单击“更多>重启>滚动重启”。

   滚动重启过程中，集群可以提供服务，只有正在重启的节点不可用。如果客户端访问某个节点失败，可以访问其他节点。
3. 重启集群后，请刷新页面，观察集群状态。重启过程中，集群状态为“处理中”，任务状态为“重启中”。如果集群状态变更为“可用”，表示集群已重启成功。如果集群状态变更为“异常”，建议联系管理员排查故障。

9.7 迁移集群

将一个集群的数据迁移到另一个集群，我们称之为集群迁移。集群迁移的应用场景很多，如当业务数据不断增加时，无法直接修改当前集群的规格以便满足需求时，可以选择创建一个规格较高的集群，然后通过集群迁移的操作，快速将数据全部迁移至新集群中，以满足业务需求。另一个场景，如通过集群迁移可将两个集群的索引合并到一个集群中，以满足业务的需要。在云搜索服务中，通过备份与恢复索引功能可实现集群迁移，即将一个集群的快照恢复到另一个集群。

迁移条件

- 原集群和目标集群在同一个 region 下。
- 目标集群的版本等于或高于原集群。
- 目标集群节点数要大于原集群节点数的一半。

迁移建议

- 目标集群的节点数不少于原集群的 shard 副本数。
- 目标集群的 CPU、MEM 和 Disk 配置大于等于原集群，使迁移后业务受损最小化。

本文以将集群“Es-1”中的数据迁移到集群“Es-2”为例。其中“Es-2”集群的版本高于“Es-1”集群，且节点数要高于“Es-1”节点数的 1/2。

操作步骤

1. 在集群管理界面中，单击集群名称“Es-1”进入集群“基本信息”页面。然后，选择“集群快照”页签。
2. 单击“创建快照”手动创建快照，在弹出框中输入快照名称并单击“提交”，等待快照创建完成。

首次使用备份与恢复索引功能，需要先进行基础配置，详见请参见手动创建快照。

图9-12 创建快照

快照创建完成后，如图 9-13 所示。

图9-13 快照创建成功

3. 在快照管理页面，单击该快照操作列的“恢复”按钮，将数据恢复至 Es-2 集群。
   - 在“索引”的文本框中输入“*”，表示对集群“Es-1”的全部索引进行恢复。
   - 在“集群”的下拉框中选择“Es-2”，将该快照恢复到集群“Es-2”中。

最后单击“确定”按钮开始恢复。当然还可以进行对恢复之后的索引重命名等操作，详情请参见 9.3 备份与恢复索引。

图9-14 恢复数据

4. 恢复完成后，即完成了集群“Es-1”中的数据到集群“Es-2”的迁移。

9.8 删除集群

当用户已完成数据搜索业务，无需继续使用某一集群时，可删除集群。删除集群时，集群创建的快照将保存在 OBS 桶中，不会被删除。

说明

由于集群删除后，数据无法恢复，请您谨慎操作。

操作步骤

1. 登录云搜索服务管理控制台。
2. 单击“集群管理”进入集群列表界面，在对应集群的“操作”列中单击“更多>删除”。
3. 在弹出的确认提示框中，单击“确定”完成集群删除。

9.9 标签管理

标签是集群的标识。为集群添加标签，可以方便用户识别和管理拥有的集群资源。

您可以在创建集群时添加标签，也可以在集群创建完成后，在集群的详情页添加标签。

为集群添加标签

1. 登录云搜索服务管理控制台。
2. 在创建集群页面，“高级配置”选择“自定义”后，为集群添加标签。

   您可以选择预定义标签，并为此标签设置“标签值”。您可以单击“查看预定义标签”，进入“标签管理服务”，了解此用户下已有的标签。

   您也可以自定义“标签键”和“标签值”。

   云搜索服务的每个集群最多可以设置 10 个标签。当设置不正确时，可单击标签右侧的“删除”按钮，删除此标签。当不设置标签时，可保持为空。

   表9-5 标签命名规则

参数	说明
标签键	不能为空。 对于同一个集群，标签键值唯一。 长度不超过 36 个字符。 只能包含数字、英文字母、下划线、中划线和中文。
标签值	长度不超过 43 个字符。 只能包含数字、英文字母、下划线、中划线和中文。 不能为空。

通过标签搜索集群

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击集群列表右上角的“标签搜索”。
3. 输入需要搜索的标签键和标签值。

   标签键和标签值仅支持从下拉列表中选择，当标签键和标签值全匹配时，系统可以自动查询到目标集群。当有多个标签条件时，会取各个标签的交集，进行集群查询。

   系统最多支持 10 个不同标签的组合搜索。
4. 单击“搜索”。

   系统根据标签键和标签值搜索目标集群。

已有集群标签管理

您可以对已经创建的集群的标签进行修改，删除，也可以添加标签。

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击待管理标签的集群名称。

   系统跳转至该集群“基本信息”页面。
3. 选择“标签”，在此可以对集群标签进行添加，修改，删除操作。
   - 查看

在“标签”页，可以查看当前集群的标签详情，包括标签个数，以及每个标签的键和值。

- 添加

  单击左上角的“添加标签”，在弹出的“添加标签”窗口，输入新添加标签的键和值，并单击“确定”。

- 修改

  只能修改已有标签的标签值。

  单击标签所在行“操作”列下的“编辑”，在弹出的“编辑标签”窗口，输入修改后标签值，并单击“确定”。

- 删除

  单击标签所在行“操作”列下的“删除”，如果确认删除，在弹出的“删除标签”窗口，单击“确定”。

9.10 公网访问

针对安全集群，云搜索服务的集群支持配置公网访问，配置完成后，通过提供的公网IP，您可以在外网接入安全集群。

创建集群时配置公网访问

1. 登录云搜索服务管理控制台。
2. 在创建集群页面，开启“安全模式”。

   6.5.4 及之后版本的集群支持开启“安全模式”。
3. “公网访问”选择“自动绑定”，配置公网访问相关参数。

表9-6 公网访问参数说明

参数	说明
带宽	设置公网访问的带宽。
访问控制开关	如果关闭访问控制开关，则允许任何 IP 通过公网 IP 访问集群。如果开启访问控制开关，则只允许白名单列表中的 IP 通过公网 IP 访问集群。
白名单	设置允许访问的 IP 地址或网段，中间用英文逗号隔开。

已有集群公网访问管理

您可以对已经创建集群的公网访问进行修改，查看，解绑，也可以配置公网访问。

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击需要配置公网访问的集群名称，进入集群基本信息页面。
   - 配置公网访问

如果创建集群时，未配置公网访问，集群配置成功后，可以在集群基本信息页面配置公网访问。

单击“公网访问”参数右侧的“绑定”，设置访问带宽后，单击“确定”。

如果绑定失败，用户可以等待几分钟后，再次尝试重新绑定公网访问。

- 修改

  对已经配置了公网访问的集群，可以通过单击“带宽”参数右侧的“修改”，修改带宽大小，也可以通过单击“访问控制”右侧的“设置”，设置访问控制开关和访问白名单。

- 查看

  在“基本信息”页面，可以查看当前集群绑定的公网 IP 地址。

- 解绑

  对于已经绑定的公网 IP，可以通过单击“公网访问”参数右侧的“解绑”，解绑公网 IP。

通过公网 IP 接入集群

公网访问配置完成后，集群将会获得一个公网 IP，用户可以通过公网 IP 和端口接入集群。接入方式为 ***https://公网 IP:9200/接口 URL***。

9.11 日志管理

为了方便用户使用日志定位问题，云搜索服务提供了日志备份和日志查询功能。用户可以将集群的日志备份在 OBS 桶中，然后通过 OBS 可以直接下载需要的日志文件，进行问题分析定位。

开启日志管理

1. 登录云搜索服务管理控制台。
2. 在“集群管理”页面，单击需要配置日志备份的集群名称，进入集群基本信息页面。
3. 选择“日志管理”，在“日志管理开关”右侧单击开关，打开集群的日志管理功能。

    表示关闭日志管理功能，  表示打开日志管理功能。

4. （可选）打开日志管理功能后，云搜索服务会自动为客户创建 OBS 桶、备份路径和 IAM 委托，用于日志备份。自动创建的 OBS 桶、备份路径和 IAM 委托将直接展示在界面中。如果您不希望使用自动创建的 OBS 桶、备份路径和 IAM 委托，您可以在“日志备份配置”右侧单击  进行配置。

   在修改基础配置弹出框中，您可以在下拉框中选择您账户下已有的 OBS 桶和 IAM 委托，或者通过“创建桶”和“创建委托”链接重新创建。

表9-7 参数说明

参数	说明	注意事项
“OBS 桶”	日志存储的 OBS 桶的名称。	创建或者已存在的 OBS 桶需满足“区域”必须跟创建集群的区域保持一致。
“备份路径”	日志在 OBS 桶中的存放路径。	备份路径配置规则： • 备份路径不能包括下列符号：\:*?"<>| • 备份路径不能以“/”开头。 • 备份路径不能以“.”开头或结尾。 • 备份路径的总长度不能超过 1023 个字符。
“IAM 委托”	指当前账户授权云搜索服务访问或维护存储在 OBS 中数据。	创建或者已存在的 IAM 需满足如下条件： • “委托类型”选择“云服务” • “云服务”选择“Elasticsearch”。 • 设置当前委托具备“全局服务”中“对象存储服务”项目的“OBS Administrator”权限。

说明

如果是子账户，需要同时设置 GetBucketStoragePolicy、GetBucketLocation、ListBucket 权限，才能看到 OBS 桶。

5. 日志备份。
   a. 自动备份日志。

      在“自动备份开关”右侧，单击开关，开启自动备份日志功能。

       表示打开自动备份日志功能，  表示关闭自动备份日志功能。

      开启“自动备份开关”后，在“修改日志备份策略”页面修改“备份开始时间”。设置成功后，系统会按照设置的时间进行自动备份。

   b. 手动备份日志。

      单击“日志备份”下面的“开始备份”，在弹出的确认提示框中，单击“确定”，开始备份日志。

      日志备份列表中的“任务状态”为“SUCCESS”时，表示日志备份成功。

说明

云搜索服务会把集群中当前的所有日志全部复制到指定的 OBS 路径中，用户可以在自己的 OBS 桶对应的路径中直接查看或者下载日志文件。

6. 日志查询。

   用户可以根据集群的节点，日志类型，日志级别信息查询集群各个节点的日志信息。可查询的日志类型包括：运行日志、慢索引日志、慢查询日志、废弃操作日

志。查询日志时，是从最近时刻的 1 万条日志中进行匹配，查询结果最多显示 100 条。

在“日志查询”页面，选择需要查询的节点，日志类型，日志级别信息后，单击  ，显示查询结果。

日志信息

日志备份成功后，用户可以单击“备份路径”，进入到 OBS 控制台，查看备份的日志信息。

图9-15 查看日志信息

云搜索服务备份的日志信息主要包括废弃操作日志、运行日志、慢索引日志、慢查询日志。在 OBS 桶中的存储类型如表 9-8 所示。

表9-8 日志类型信息

日志名称	描述
clustername_deprecation.log	弃用操作的日志记录。
clustername_index_indexing_slowlog.log	慢索引日志。
clustername_index_search_slowlog.log	慢索引查询日志。
clustername.log	Elasticsearch 运行日志。
clustername_access.log	接入日志。
clustername_audit.log	审计日志。

9.12 插件管理

云搜索服务支持查看系统默认插件功能。

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击需要安装插件的集群名称。
   系统跳转至该集群基本信息页面。
3. 在集群基本信息页面，选择“插件管理”。
4. 查看系统默认插件信息。
   在“系统默认插件列表”页查看当前版本支持的系统默认插件信息。

表9-9 系统默认插件支持的集群版本号

插件名称	支持的集群版本号
analysis-dynamic-synonym	5.5.1、6.2.3、6.5.4、7.1.1、7.6.2、7.9.3
analysis-icu	6.2.3、6.5.4、7.1.1、7.6.2、7.9.3
analysis-ik	5.5.1、6.2.3、6.5.4、7.1.1、7.6.2、7.9.3
analysis-pinyin	5.5.1、6.2.3、6.5.4、7.1.1、7.6.2、7.9.3
analysis-poisson	6.2.3、7.1.1
analysis-stconvert	5.5.1、6.2.3、6.5.4、7.1.1、7.6.2
lasthit	5.5.1、6.2.3、7.1.1
repository-obs	5.5.1、6.2.3、6.5.4、7.1.1、7.6.2、7.9.3
vector-search	6.2.3
opendistro_security	6.5.4、7.1.1、7.6.2、7.9.3
opendistro_sql	6.5.4、7.1.1、7.6.2、7.9.3
analysis-kuromoji	6.5.4、7.1.1、7.6.2、7.9.3
analysis-nori	6.5.4、7.1.1、7.6.2、7.9.3
ingest-attachment	6.5.4、7.1.1、7.6.2、7.9.3

9.13 冷热数据存储

云搜索服务提供了冷数据节点供企业选择，企业可以将部分现查要求秒级返回的数据放在高性能机器上面，对于历史数据要求分钟级别返回的数据放在大容量低规格节点。

说明

- 创建集群时，数据节点为必选，只有当选择了冷数据节点后，数据节点才会变成热节点。
- 选择冷数据节点的同时，支持独立选择 Master 和 Client 节点。
- 冷数据节点支持节点和磁盘扩容，前提是冷节点规格支持（本地盘不支持磁盘扩容）。

冷热数据切换

选择冷数据节点后，冷数据节点将会打上 cold”标签，用来表示冷节点。同时，数据节点将会上升为热节点，会被打上“hot”标签。用户可以通过配置指定索引，将数据分配到冷热节点。

通过设置 template，可以通过模板将相应的 index 存储到指定冷热节点。

如下，登录集群的 Kibana Console 页面，配置 myindex 开头的索引，储存在冷节点上面。 这样可以通过模板在创建的时候把 myindex*的数据存储在冷数据节点上面。

```
PUT _template/test
{
   "order": 1,
   "template": "myindex*",
   "settings": {
      "index": {
         "refresh_interval": "30s",
         "number_of_shards": "3",
         "number_of_replicas": "1",
         "routing.allocation.require.box_type": "cold"
      }
   }
}
```

同时也可以单独对已经创建好的索引进行操作。

```
PUT myindex/_settings
 {
       "index.routing.allocation.require.box_type": "cold"
   }
```

也可以去掉冷热数据配置，不受冷热数据标签影响。

```
PUT myindex/_settings
{
       "index.routing.allocation.require.box_type": null
   }
```

9.14 参数配置

云搜索服务支持用户通过集群管理页面修改 elasticsearch.yml 文件的某些配置，修改配置完后需要重启集群才能够使这些配置生效。

修改参数配置

1. 登录云搜索服务管理控制台。

2. 在“集群管理”页面，单击需要修改参数配置的集群名称，进入集群基本信息页面。
3. 选择“参数配置”，根据需求，修改对应模块的参数取值。

表9-10 模块参数信息说明

模块名称	参数名称	说明
跨域访问	http.cors.allow-credentials	跨域访问是否返回头部的 Access-Control-Allow-Credentials。 取值范围：true、false。 默认值：false。
	http.cors.allow-origin	允许跨域访问的 IP，配置样例如 122.122.122.122:9200。
	http.cors.max-age	浏览器默认缓存时间。如果超过设置的时间后，缓存将自动清除。 单位：秒。 默认值：1728000。
	http.cors.allow-headers	跨域访问允许的 headers，包括 X-Requested-With, Content-Type, Content-Length，中间用英文逗号和空格分开。
	http.cors.enabled	是否允许跨域访问。 取值范围：true、false。 默认值：false。
	http.cors.allow-methods	跨域访问允许的方法，包括 OPTIONS, HEAD, GET, POST, PUT, DELETE，中间用英文逗号和空格分开。
集群索引重建	reindex.remote.whitelist	配置该参数可以将本集群数据通过 reindex 接口迁移到配置的集群，配置样例如 122.122.122.122:9200。
自定义缓存	indices.queries.cache.size	查询阶段的缓存大小。 取值范围：1-100。 单位：%。 默认值：10%。
线程池队列大小	thread_pool.bulk.queue_size	Bulk 请求的队列大小。输入的参数值为整数类型。 默认值：200。 当集群版本为 7.x 之前版本时，显示此参数。
	thread_pool.write.queue_size	线程池写入队列大小。输入的参数值为整数类型。

模块名称	参数名称	说明
		默认值：200。 当集群版本为 7.x 之后版本时，显示此参数。
	thread_pool.force_merge.size	用来做 forcemerge 的队列大小。输入的参数值为整数类型。 默认值：1。
自定义	用户可以根据实际情况，添加相关参数名称。	自定义参数的取值。 说明 • 如果自定义参数有多个取值，则取值的输入格式为[value1, value1, value1...]。 • 取值之间用英文逗号和空格隔开。 • 自定义参数值中不能包含冒号。

4. 修改完成后，单击“确认修改”。

   系统弹出确认提示，勾选“参数配置后需要手动重启才能生效”后，单击“确定”。

   您可以在当前页面查看修改记录，系统最多显示 20 条修改记录。

说明

如果修改了参数配置，未重启集群，则在“集群管理”页面的“任务状态”栏显示为“配置未更新”。

如果修改后重启集群，“任务状态”显示“配置错误”，则表示修改参数配置文件失败。

9.15 终端节点服务

云搜索服务提供了终端节点服务，用户开启了此服务后，可以通过内网域名访问集群。在开启终端节点服务时，系统会默认给用户创建一个终端节点。创建终端节点需要有相关的权限，请参考 VPC 终端节点权限管理。

注意

公网访问和终端节点服务功能使用的是同一个负载均衡。如果开启了公网访问白名单，由于白名单是作用在负载均衡上面，会同时限制公网访问集群和内网通过 VPCEP 访问集群的 IP。此时需要在公网访问白名单中添加一个网络白名单 198.19.128.0/20，该白名单用来放通经过 VPCEP 的流量。

创建集群时开启终端节点服务

1. 登录云搜索服务管理控制台。
2. 在创建集群页面，“高级配置”选择“自定义”后，开启终端节点服务。
   - “创建内网域名”：如果开启，系统将会自动为用户创建一个内网域名，可以通过内网域名访问集群。
   - “终端节点服务白名单”：您可以在“终端节点服务白名单”中添加需要授权的账号 ID，只要其账号 ID 被添加到终端节点服务白名单中，，就可以通过内网域名或者节点 IP 访问集群。

     如果需要添加多个账号，可以通过单击  进行添加。也可以通过单击“操作”列下面的“删除”，进行删除不允许访问的账号。

说明

- 授权账号 ID 配置成*，则表示允许全部用户访问该集群。
- 需要授权的账号 ID 可在“我的凭证”中进行查看。
- 集群开启终端节点服务之后，终端节点将按需进行收费，终端节点的费用将由用户进行支付，详细的计费方式请参考终端节点计费说明。

已有集群终端节点服务管理

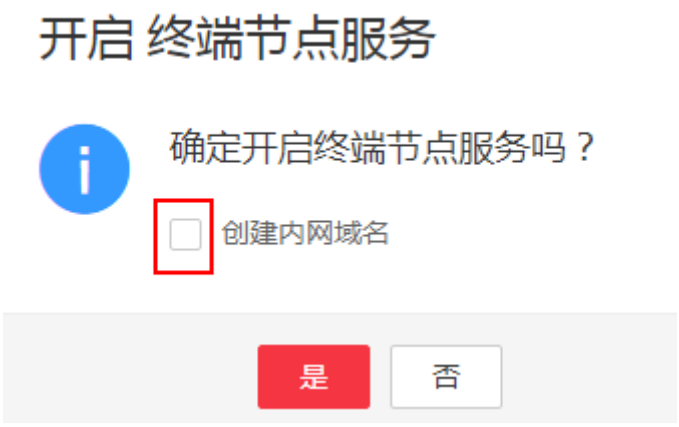
如果创建集群时未开启终端节点服务，集群创建成功后，可以通过如下步骤进行开启。

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击需要开启终端节点服务的集群名称，进入集群基本信息页面。
3. 选择“终端节点服务”，在“终端节点服务”右侧单击开关，打开集群的终端节点服务功能。

    表示关闭终端节点服务功能，  表示打开终端节点服务功能。

   在弹出的提示框中，您可以根据需求，选择是否开启内网域名。选择创建内网域名后，可以通过内网域名访问集群。

图9-16 开启终端节点服务



说明

- 开启终端节点服务后，您可以通过终端节点产生的“内网域名”或者“节点 IP”访问此集群。详细请参考通过内网域名或节点 IP 访问集群。
- 关闭终端节点服务功能后，所有的用户将不能通过内网域名访问此集群。

4. 单击“是”，开启终端节点服务。
5. （可选）打开终端节点服务后，您可以单击“终端节点服务白名单”后面的“更新”，更新已有的白名单。
6. 连接管理。

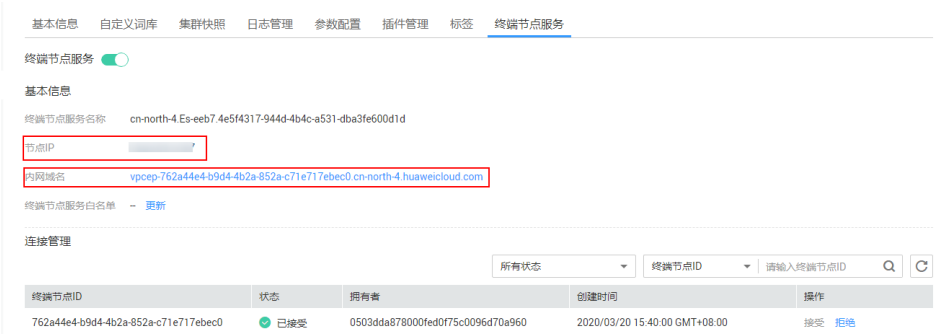
   在终端节点服务页面下，显示所有连接当前终端节点服务的终端节点。您可以对这些终端节点的“连接状态”进行“接受”或者“拒绝”操作。如果对某个终端节点“拒绝”操作之后，其生成的内网域名将不能再访问到当前集群。

通过内网域名或节点 IP 访问集群

1. 获取内网域名或者节点 IP。
   - 当前用户

     登录 console 控制台，单击集群名称，进入集群“基本信息”页面，选择“终端节点服务”，查看内网域名，如图 9-17。

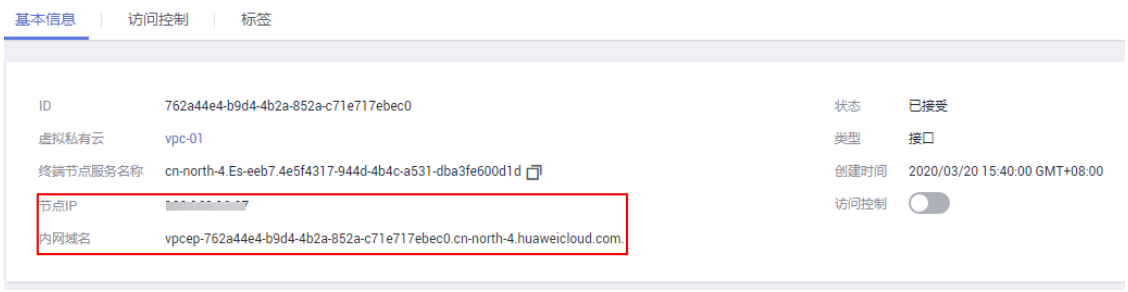
图9-17 查看内网域名和节点 IP 信息（1）



- 其他用户

  如果您申请了终端节点，登录终端节点控制台，单击“ID”，进入终端节点基本信息页面查看内网域名。如图 9-18。

图9-18 查看内网域名和节点 IP 信息（2）



2. 在弹性云服务器中，直接通过 cURL 执行 API 或者开发程序调用 API 并执行程序即可使用集群。Elasticsearch 操作和接口请参见《Elasticsearch：权威指南》。

   弹性云服务器需要满足如下要求：

   - 为弹性云服务分配足够的磁盘空间。
   - 此弹性云服务器的 VPC 需要与集群在同一个 VPC 中，开通终端节点服务后，可以实现跨 VPC 访问。
   - 此弹性云服务器的安全组需要和集群的安全组相同。

     如果不同，请修改弹性云服务器安全组或配置弹性云服务器安全组的出入规则允许集群所有安全组的访问。修改操作请参见《虚拟私有云用户指南》。
   - 待接入的 ES 集群，其安全组的出方向和入方向需允许 TCP 协议及 9200 端口，或者允许端口范围包含 9200 端口。

   例如，使用 cURL 执行如下命令，查看集群中的索引信息，集群中的内网访问地址为“vpcep-7439f7f6-2c66-47d4-b5f3-790db4204b8d.region01.xxxx.com”，端口为“9200”。

   - 如果接入集群未启用安全模式，接入方式为：

     ```
     curl 'http://vpcep-7439f7f6-2c66-47d4-b5f3-
     790db4204b8d.region01.xxxx.com:9200/_cat/indices'
     ```

- 如果接入集群已启用安全模式，则需要使用 https 方式访问，并附加用户名和密码，在 curl 命令中添加-u 选项。

```
curl -u username:password -k 'https://vpcep-7439f7f6-2c66-47d4-b5f3-790db4204b8d.region01.xxxx.com:9200/_cat/indices'
```

9.16 Kibana 公网访问

云搜索服务支持安全模式的集群通过公网访问云上 Kibana 服务。针对安全模式集群，云搜索服务支持配置 Kibana 开启公网访问，选择 Kibana 公网访问带宽，配置完成后，对应集群将会获得一个 Kibana 公网访问地址，通过这个地址可以在公网上面访问集群的 Kibana。

对于安全模式集群来说，支持在创建的时候配置 Kibana 公网访问，同时也支持安全模式集群创建完之后，开启 Kibana 公网访问。

说明

如果在该特性上线之前购买的安全模式的集群，不支持此功能。

创建集群时配置 Kibana 公网访问

1. 登录云搜索服务管理控制台。
2. 在创建集群页面，开启“安全模式”。

   6.5.4 及之后版本的集群支持开启“安全模式”。
3. “高级配置”选择“自定义”后，开启 Kibana 公网访问，配置相关参数。

表9-11 Kibana 公网访问参数说明

参数	说明
带宽	设置公网访问的带宽。 取值范围：1-100。 单位：Mbit/s。
访问控制开关	如果关闭访问控制开关，则允许任何 IP 通过公网 IP 访问集群 Kibana。如果开启访问控制开关，则只允许白名单列表中的 IP 通过公网 IP 访问集群 Kibana。
白名单	设置允许访问的 IP 地址或网段，中间用英文逗号隔开。 建议开启白名单。

集群创建成功后，单击集群名称，进入集群基本信息页面，在“Kibana 公网访问”页签，可以查看 kibana 公网访问地址。

已有集群开启 Kibana 公网访问

您可以对已经创建的安全模式集群的 Kibana 公网访问进行开启，关闭，修改，查看等操作。

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击需要配置 Kibana 公网访问的集群名称，进入集群基本信息页面。
3. 选择“Kibana 公网访问”页签，在 **Kibana 公网访问**右侧单击开关，打开 Kibana 公网访问功能。

    表示关闭 Kibana 公网访问功能，  表示打开 Kibana 公网访问功能。
4. 在开启 Kibana 公网访问页面，配置相关参数。

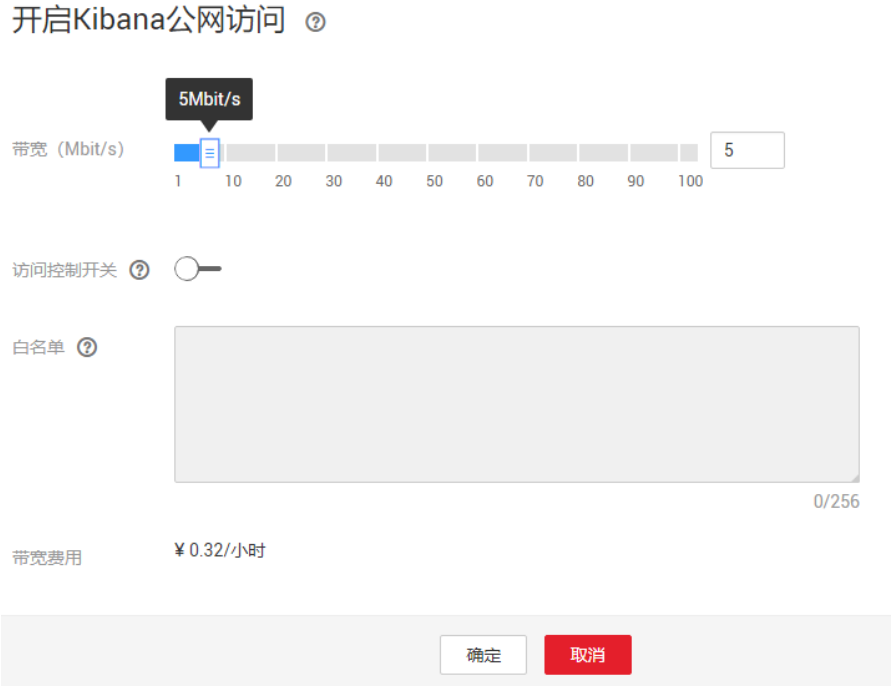


表9-12 Kibana 公网访问参数说明

参数	说明
带宽	设置公网访问的带宽。 取值范围：1-100。 单位：Mbit/s。
访问控制开关	如果关闭访问控制开关，则允许任何 IP 通过公网 IP 访问集群 Kibana。如果开启访问控制开关，则只允许白名单列表中的 IP 通过公网 IP 访问集群 Kibana。
白名单	设置允许访问的 IP 地址或网段，中间用英文逗号隔开。 建议开启白名单。

5. 配置完成后，单击“确定”。

修改 Kibana 公网访问

对已经配置了 Kibana 公网访问的集群，云搜索服务支持修改带宽和访问控制功能。

- 修改带宽

  单击“带宽”参数右侧的“修改”，在“修改 Kibana 公网访问带宽”页面修改带宽大小，修改完成后，单击“确定”。

- 修改访问控制

  单击“访问控制开关”右侧的“修改”，在“修改 Kibana 公网访问控制”页面设置“访问控制开关”和访问“白名单”，修改完成后，单击“确定”。

通过公网 IP 访问 Kibana

Kibana 公网访问配置完成后，将会获得一个 kibana 公网访问地址，用户可以通过此 IP 访问集群的 Kibana。

1. 登录云搜索服务管理控制台。
2. 在集群管理页面，单击需要配置 Kibana 公网访问的集群名称，进入集群基本信息页面。
3. 选择“Kibana 公网访问”页签，获取 kibana 公网访问地址。
4. 通过该地址，就可以在公网上面访问云搜索服务集群的 Kibana。

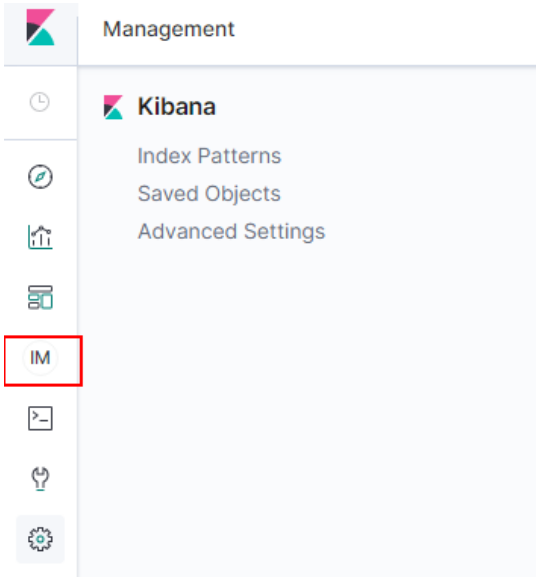
10 索引状态管理

10.1 创建及管理索引

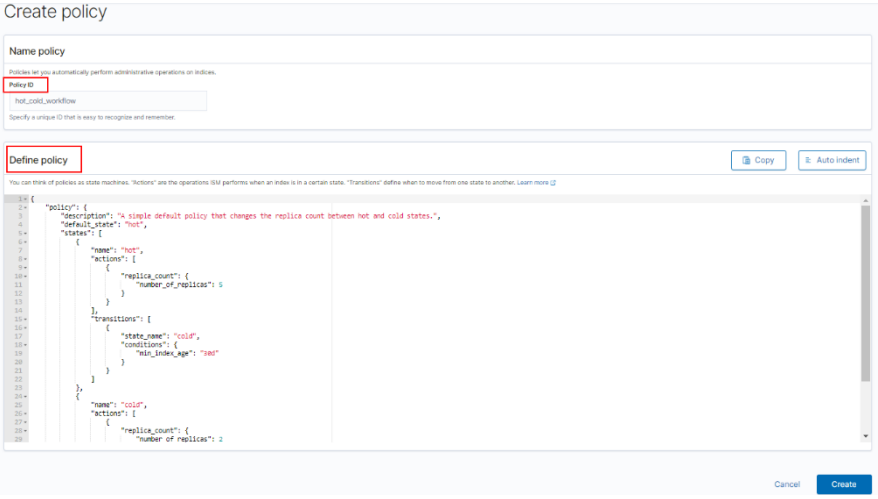
云搜索服务 7.6.2 版本的集群支持索引状态管理。索引状态管理（ISM）是一个插件，通过该插件，您可以根据索引使用期限，索引大小或文档数的变化触发这些定期的管理操作，从而使它们自动化。使用 ISM 插件时，您可以根据需要定义自动处理索引滚动或删除的策略。

创建索引策略

1. 登录 Kibana，在左侧选择 IM，进入索引管理页面 。



2. 单击 **Create policy**，创建索引策略。
3. 在 **Policy ID** 部分输入策略 ID，**Define policy** 部分输入您的策略。



4. 单击 **Create**。

   创建策略后，下一步将此策略附加到一个或多个索引。您还可以将 policy_id 在索引模板中包含，因此当创建与索引模板模式匹配的索引时，该索引将附加有策略。

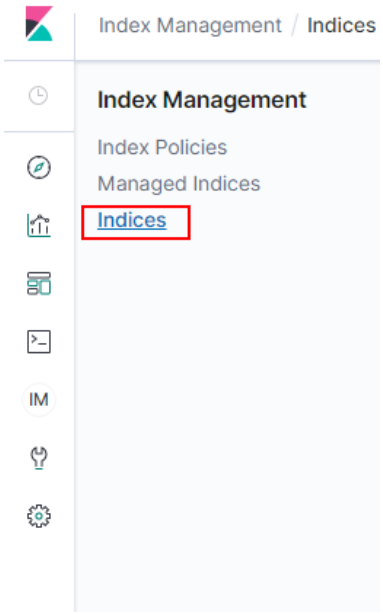
   创建索引模板可参考索引模板。

```
PUT _template/<template_name>
{
    "index_patterns": [
        "index_name-*"
    ],
    "settings": {
        "opendistro.index_state_management.policy_id": "policy_id"
    }
}
```

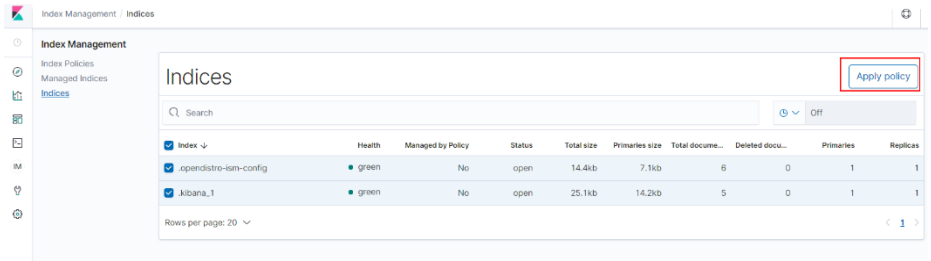
   - <template_name>：需要替换为创建的索引模板名。
   - policy_id：需要替换为 3 创建的 Policy ID。

将策略附加到索引

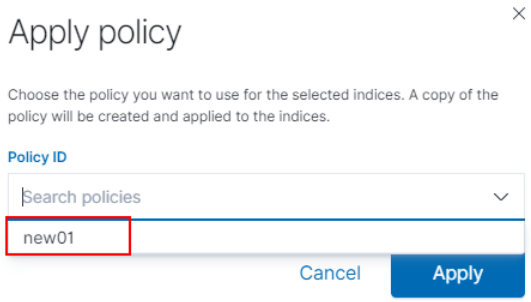
1. 选择 **Indices**。



2. 在 **Indices** 列表中选择您要附加策略的一个或多个索引。
3. 单击右上角的 **Apply policy**，添加应用策略。



4. 从 **Policy ID** 菜单中，选择您创建的策略。



5. 单击 **Apply**。

   将策略附加到索引后，ISM 会默认创建每 5 分钟运行一次的作业，以执行策略操作，检查条件并将索引转换为不同的状态。

管理索引

1. 选择 **Managed Indices**。
2. 如果您要更改策略，可以选择 **Change policy**，详情请参考**变更策略**。
3. 如果您要删除策略，请选择您的策略，然后选择 **Remove policy**。
4. 如果您要重试策略，请选择您的策略，然后选择 **Retry policy**。

具体使用可参考索引管理官方介绍。

10.2 变更策略

您可以更改任何托管索引策略，但是 ISM 有一些约束条件可以确保策略更改不会破坏索引。

如果索引卡在其当前状态，永不进行，并且您想立即更新其策略，请确保新策略包括与旧策略相同的状态（名称，操作，顺序相同）。在这种情况下，即使策略处于执行操作中，ISM 也会应用新策略。

如果在不包含相同状态的情况下更新策略，则 ISM 仅在当前状态下的所有操作执行完成后才更新策略。或者，您可以在旧策略中选择特定状态，然后让新策略生效。

在 Kibana 中更改更改政策，操作步骤如下：

1. 在 **Managed indices** 下，选择需要更换新策略的索引。
2. 单击右上角的 **Change policy**，进入 **Choose managed indices** 页面，选择更换新策略的相关信息 。

表10-1 更换索引策略参数信息

参数	说明
Managed indices	选择需要更换策略的索引名称。支持选择多个索引。
State filters	选择索引状态。选择后，会将新策略附加到处于特定状态的索引。
New policy	选择新策略。

3. 选择完成后，单击 **Change**。

11 监控集群

11.1 支持的监控指标

功能说明

本节定义了云搜索服务上报云监控服务的监控指标的命名空间，监控指标列表和维度定义。用户可以通过云监控服务提供管理控制台或 API 接口来检索云搜索服务产生的监控指标和告警信息。

命名空间

SYS.ES

监控指标

表11-1 监控指标说明

指标 ID	指标名称	含义	取值范围	测量对象&维度	监控周期（原始指标）
status	集群健康状态	该指标用于统计测量监控对象的状态。	0,1,2,3 0：集群是100%可用的。 1：数据是完整的，部分副本缺失。高可用性在某种程度上弱化，存在风险，请及时关注集群情况。 2：数据缺失，集群使用时将出现异常。	ES 集群	1 分钟

指标 ID	指标名称	含义	取值范围	测量对象&维度	监控周期（原始指标）
			3：没有获取到集群状态。		
disk_util	磁盘使用率	该指标用于统计测量对象的磁盘使用率。 单位：百分比	0-100%	ES 集群	1 分钟
max_jvm_heap_usage	最大 JVM 堆使用率	ES 集群节点中最大的 JVM 堆使用率。 单位：百分比。	0-100%	ES 集群	1 分钟
max_jvm_young_gc_time	最大 JVM Young GC 耗时	ES 集群节点中最大的 JVM Young GC 耗时。 单位：ms	≥ 0 ms	ES 集群	1 分钟
max_jvm_young_gc_count	最大 JVM Young GC 次数	ES 集群节点中最大的 JVM Young GC 次数。	≥ 0	ES 集群	1 分钟
max_jvm_old_gc_time	最大 JVM Old GC 耗时	ES 集群节点中最大的 JVM Old GC 耗时。 单位：ms	≥ 0 ms	ES 集群	1 分钟
max_jvm_old_gc_count	最大 JVM Old GC 次数	ES 集群节点中最大的 JVM Old GC 次数。	≥ 0	ES 集群	1 分钟
total_fs_size	文件系统总大小	ES 集群的文件系统总大小。 单位：byte	≥ 0 bytes	ES 集群	1 分钟
free_fs_size	文件系统可用大小	ES 集群的文件系统可用大小。 单位：byte	≥ 0 bytes	ES 集群	1 分钟

指标 ID	指标名称	含义	取值范围	测量对象&维度	监控周期（原始指标）
max_cpu_usage	最大 CPU 利用率	ES 集群节点中最大的 CPU 利用率。 单位：百分比	0-100%	ES 集群	1 分钟
max_cpu_time_of_jvm_process	最大 JVM 进程使用的 CPU 时间	ES 集群节点中最大的 JVM 进程使用的 CPU 时间。 单位：ms	≥ 0 ms	ES 集群	1 分钟
max_virtual_memory_size_of_jvm_process	最大 JVM 进程使用的虚拟内存大小	ES 集群节点中最大的 JVM 进程使用的虚拟内存大小。 单位：byte	≥ 0 bytes	ES 集群	1 分钟
max_current_opened_http_count	最大当前打开的 Http 连接数	ES 集群节点中最大的当前打开的 Http 连接数。	≥ 0	ES 集群	1 分钟
max_total_opened_http_count	最大全部打开的 Http 连接数	ES 集群节点中最大的全部打开的 Http 连接数。	≥ 0	ES 集群	1 分钟
indices_count	索引数量	ES 集群的索引数量。	≥ 0	ES 集群	1 分钟
total_shards_count	分片数量	ES 集群的分片数量。	≥ 0	ES 集群	1 分钟
primary_shards_count	主分片数量	ES 集群的主分片数量。	≥ 0	ES 集群	1 分钟
docs_count	文档数量	ES 集群的文档数量。	≥ 0	ES 集群	1 分钟
docs_deleted_count	被删除的文档数量	ES 集群的被删除的文档数量。	≥ 0	ES 集群	1 分钟
nodes_count	节点数量	ES 集群的节	≥ 0	ES 集群	1 分钟

指标 ID	指标名称	含义	取值范围	测量对象&维度	监控周期（原始指标）
		点数量。			
data_nodes_count	数据节点数量	ES 集群的数据节点数量。	≥ 0	ES 集群	1 分钟
coordinating_nodes_count	协调节点数量	ES 集群的协调节点数量。	≥ 0	ES 集群	1 分钟
master_nodes_count	Master 节点数量	ES 集群的 Master 节点数量。	≥ 0	ES 集群	1 分钟
ingest_nodes_count	Client 节点数量	ES 集群的 Client 节点数量。	≥ 0	ES 集群	1 分钟

维度

表11-2 维度说明

Key	Value
cluster_id	ES 集群

11.2 创建告警规则

通过在云监控服务管理控制台创建集群指标的告警规则，当监控指标达到用户设置的告警规则触发告警时，用户可以及时了解集群的异常状况并采取措施，以免造成业务损失。

操作步骤

1. 登录管理控制台。
2. 选择“服务列表>管理与部署 > 云监控服务”。
3. 在左侧导航树栏，选择“告警 > 告警规则”。
4. 在“告警规则”界面，单击“创建告警规则”。
5. 在“创建告警规则”对话框中，根据界面提示配置参数。

   云监控服务支持对某个特定的监控指标设置自定义告警规则，同时也支持使用告警模板为多个实例或者云服务批量创建告警规则。这里以使用告警模板创建云搜索服务的集群的告警规则为例，介绍如何创建告警规则。

a. 配置告警名称和描述信息。

图11-1 配置规则信息

表11-3 参数说明

参数	说明	示例
名称	系统会随机产生一个名称，用户也可以进行修改。	alarm-p8v9
描述	告警规则描述（此参数非必填项）。	-

b. 选择监控对象，配置告警内容参数。

图11-2 配置告警内容

表11-4 配置告警内容

参数	说明	示例
资源类型	配置告警规则监控的服务名称。	云搜索服务
维度	用于指定告警规则对应指标的维度名称。目前支持的维度包括： • ES 集群：以集群维度指定告警规则。 • ES 集群 - 云服务节点：以集群中的某个	ES 集群

参数	说明	示例
	节点维度指定告警规则。	
监控范围	告警规则适用的资源范围，可选择资源分组或指定资源。 说明： • 当选择资源分组时，该分组下任何资源满足告警策略时，都会触发告警通知。 • 选择指定资源时，勾选具体的监控对象，单击  将监控对象同步到右侧对话框。	指定资源

c. 配置“选择类型”、“模板”和“发送通知”等参数。

表11-5 参数说明

参数	说明	示例
选择类型	根据需要可选择从模板导入或自定义创建。 当监控范围为指定资源时可选择从模板导入。	自定义创建
模板	选择需要导入的模板。	-
发送通知	如果已开启发送通知功能，则需要配置生效时间、通知对象和触发条件。	-

d. 配置完成后，单击“立即创建”，完成告警规则的创建。

创建成功后，告警规则列表中将会显示已创建的告警规则。

11.3 配置监控对象

云搜索服务集群创建成功后，有时候想要查看集群或者单节点的监控指标时，可以通过云监控服务进行配置。

1. 登录云监控服务管理控制台。
2. 在“总览 > 监控面板”区域右侧，单击“创建监控面板”。
3. 监控面板创建成功后，然后单击“添加监控视图”。

表11-6 参数说明

参数	说明	示例
标题	自定义关注指标组件的标题名称，该名称只能由中文、英文字母、数字、下划线、中划线组成。	widget-axaj
资源类型	所关注指标对应的服务名称。	云搜索服务
维度	所关注指标的维度名称。 • ES 集群：以集群维度监控。 • ES 集群 - 云服务节点：以集群中的某个节点维度监控。	ES 集群
监控对象	所关注指标对应的监控对象。 可支持一次勾选多个监控对象。	-
监控指标	所关注指标的名称。	最大 CPU 利用率

4. 单击“下一步：配置图例名称”。

   图例名称是显示在监控视图指标变化曲线上的名称，您可以自定义图例名称，例如 ES01-CPU 使用率。这种情况下，假设当 CPU 利用率为 10%时，监控视图会展示：ES01-CPU 使用率：10%。

   您可以选择不配置图例名称，那么系统默认展示：监控对象 (资源类型) - 监控指标: 数据。

5. 单击“确定”，完成监控视图的添加。

11.4 查看监控指标

提供的云监控服务可以对云搜索服务集群的核心指标进行日常监控。您可以通过云监控服务管理控制台直观地查看集群的监控指标数据。

云监控服务只支持实时监控已创建成功的集群。

前提条件

- 集群状态为“可用”或“处理中”。

说明

“异常”、“创建中”状态的集群或者已删除的集群，无法在云监控服务管理控制台中查看其监控指标。当集群状态由“异常”或“创建中”变为“可用”时，该集群需要正常运行一段时间（约 10 分钟），才可以实时查看其监控指标。

- 集群已正常运行一段时间（约 10 分钟）。
- 已创建告警规则。

操作步骤

1. 登录管理控制台。
2. 选择“管理与部署 > 云监控服务”。
3. 在左侧导航树栏，选择“云服务监控 > 云搜索服务”。
4. 在监控列表中待查看的集群所在行的“操作”列，单击“查看监控指标”。
5. 选择待查看的时间段页签。
6. 查看监控指标数据。

12 查询 Elasticsearch SQL

在 6.5.4 及之后版本中我们提供 Open Distro for Elasticsearch SQL 插件允许您使用 SQL 而不是 Elasticsearch 查询域特定语言（DSL）编写查询。

如果您已经熟悉 SQL 并且不想学习 DSL 查询，那么此功能是一个很好的选择。

基本操作

要使用该功能，需要将请求发送到_opendistro/_sqlURI。您可以使用请求参数或请求正文（推荐）。

```
GET https://<host>:<port>/_opendistro/_sql?sql=select * from my-index limit 50
POST https://<host>:<port>/_opendistro/_sql
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

您还可以使用 curl 命令：

```
curl -XPOST https://localhost:9200/_opendistro/_sql -u username:password -k -d
'{"query": "SELECT * FROM kibana_sample_data_flights LIMIT 10"}' -H 'Content-Type:
application/json'
```

默认情况下，查询返回 JSON，但您也可以选择 CSV 格式返回数据，需要对 format 参数进行设置：

```
POST _opendistro/_sql?format=csv
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

CSV 格式返回数据时，每行对应一个文档，每列对应一个字段。

支持操作

我们支持的 SQL 操作包括声明、条件、聚合函数、Include 和 Exclude、常用函数、连接 join 和展示等操作。

- 声明 statements

表12-1 声明 statements

Statement	Example
Select	SELECT * FROM my-index
Delete	DELETE FROM my-index WHERE _id=1
Where	SELECT * FROM my-index WHERE ['field']='value'
Order by	SELECT * FROM my-index ORDER BY _id asc
Group by	SELECT * FROM my-index GROUP BY range(age, 20,30,39)
Limit	SELECT * FROM my-index LIMIT 50 (default is 200)
Union	SELECT * FROM my-index1 UNION SELECT * FROM my-index2
Minus	SELECT * FROM my-index1 MINUS SELECT * FROM my-index2

说明

与任何复杂查询一样，大型 UNION 和 MINUS 语句可能会使集群资源紧张甚至崩溃。

- 条件 Conditions

表12-2 条件 Conditions

Condition	Example
Like	SELECT * FROM my-index WHERE name LIKE 'j%'
And	SELECT * FROM my-index WHERE name LIKE 'j%' AND age > 21
Or	SELECT * FROM my-index WHERE name LIKE 'j%' OR age > 21
Count distinct	SELECT count(distinct age) FROM my-index
In	SELECT * FROM my-index WHERE name IN ('alejandro', 'carolina')
Not	SELECT * FROM my-index WHERE name NOT IN ('jane')
Between	SELECT * FROM my-index WHERE age BETWEEN 20 AND 30
Aliases	SELECT avg(age) AS Average_Age FROM my-index
Date	SELECT * FROM my-index WHERE birthday='1990-11-15'
Null	SELECT * FROM my-index WHERE name IS NULL

- 聚合函数 Aggregation

表12-3 聚合函数 Aggregation

Aggregation	Example
avg()	SELECT avg(age) FROM my-index
count()	SELECT count(age) FROM my-index
max()	SELECT max(age) AS Highest_Age FROM my-index
min()	SELECT min(age) AS Lowest_Age FROM my-index
sum()	SELECT sum(age) AS Age_Sum FROM my-index

- Include 和 Exclude 字段

表12-4 Include 和 Exclude

Pattern	Example
include()	SELECT include('a*'), exclude('age') FROM my-index
exclude()	SELECT exclude('*name') FROM my-index

- 函数 Functions

表12-5 函数 Functions

Function	Example
floor	SELECT floor(number) AS Rounded_Down FROM my-index
trim	SELECT trim(name) FROM my-index
log	SELECT log(number) FROM my-index
log10	SELECT log10(number) FROM my-index
substring	SELECT substring(name, 2,5) FROM my-index
round	SELECT round(number) FROM my-index
sqrt	SELECT sqrt(number) FROM my-index
concat_ws	SELECT concat_ws(' ', age, height) AS combined FROM my-index
/	SELECT number / 100 FROM my-index
%	SELECT number % 100 FROM my-index

Function	Example
date_format	SELECT date_format(date, 'Y') FROM my-index

说明

必须在文档映射中启用 fielddata 才能使大多数字符串函数正常工作。

- 连接操作 Joins

表12-6 连接操作 Joins

Join	Example
Inner join	SELECT p.firstname, p.lastname, p.gender, dogs.name FROM people p JOIN dogs d ON d.holdersName = p.firstname WHERE p.age > 12 AND d.age > 1
Left outer join	SELECT p.firstname, p.lastname, p.gender, dogs.name FROM people p LEFT JOIN dogs d ON d.holdersName = p.firstname
Cross join	SELECT p.firstname, p.lastname, p.gender, dogs.name FROM people p CROSS JOIN dogs d

相关约束和限制，参考“连接操作 Joins”。

- 展示 Show

展示 show 操作与索引模式匹配的索引和映射。您可以使用*或%使用通配符。

表12-7 展示 show

Show	Example
Show tables like	SHOW TABLES LIKE logs-*

连接操作 Joins

Open Distro for Elasticsearch SQL 支持 inner joins, left outer joins,和 cross joins。Join 操作有许多约束：

- 您只能加入两个参数。
- 您必须为索引使用别名（例如 people p）。
- 在 ON 子句中，您只能使用 AND 条件。
- 在 WHERE 语句中，不要将包含多个索引的树组合在一起。例如，以下语句有效：

```
WHERE (a.type1 > 3 OR a.type1 < 0) AND (b.type2 > 4 OR b.type2 < -1)
```

以下声明无效：

```
WHERE (a.type1 > 3 OR b.type2 < 0) AND (a.type1 > 4 OR b.type2 < -1)
```

- 您不能使用 GROUP BY 或 ORDER BY 来获得结果。
- LIMIT 和 OFFSET 不支持一起使用（例如 LIMIT 25 OFFSET 25）。

JDBC 驱动

Java 数据库连接（JDBC）驱动程序允许您将 Open Distro for Elasticsearch 与您的商业智能（BI）应用程序集成。

有关下载和使用 JAR 文件的信息，请参阅 GitHub 仓库。

13 查看集群日志

13.1 支持云审计的关键操作

公有云平台提供了云审计服务。通过云审计服务，您可以记录与云搜索服务相关的操作事件，便于日后的查询、审计和回溯。

前提条件

已开通云审计服务。

支持审计的关键操作列表

表13-1 支持审计的关键操作列表

操作名称	资源类型	事件名称
创建集群	cluster	createCluster
删除集群	cluster	deleteCluster
扩容集群	cluster	growCluster
重启集群	cluster	rebootCluster
配置自定义词库	cluster	loadLexicon
删除自定义词库	cluster	deleteLexicon
设置集群快照的基础配置	cluster	updateSnapshotPolicy
设置自动创建快照策略	cluster	updateAutoSnapshotPolicy
手动创建快照	snapshot	createSnapshot
恢复快照	snapshot	restoreSnapshot
删除快照	snapshot	deleteSnapshot

13.2 查看审计日志

在您开启了云审计服务后，系统会记录云搜索服务的相关操作，且控制台保存最近 7 天的操作记录。本节介绍如何在云审计服务管理控制台查看最近 7 天的操作记录。

操作步骤

1. 登录云审计服务管理控制台。
2. 在管理控制台左上角单击  图标，选择区域。
3. 在左侧导航栏中，单击“事件列表”，进入“事件列表”页面。
4. 事件列表支持通过筛选来查询对应的操作事件。当前事件列表支持四个维度的组合查询，详细信息如下：
   - 事件来源、资源类型和筛选类型。

     在下拉框中选择查询条件。

     其中筛选类型选择事件名称时，还需选择某个具体的事件名称。

     选择资源 ID 时，还需输入某个具体的资源 ID。

     选择资源名称时，还需选择或手动输入某个具体的资源名称。
   - 操作用户：在下拉框中选择某一具体的操作用户，此操作用户指用户级别，而非租户级别。
   - 事件级别：可选项为“所有事件级别”、“normal”、“warning”、“incident”，只可选择其中一项。
   - 时间范围：可选择查询最近七天内任意时间段的操作事件。
5. 在需要查看的事件左侧，单击 ⌄ 展开该事件的详细信息。
6. 单击需要查看的事件“操作”列的“查看事件”，可以在弹窗中查看该操作事件结构的详细信息。

   更多关于云审计服务事件结构的信息，请参见《云审计服务用户指南》。

14 常见问题

14.1 什么是云搜索服务

云搜索服务（Cloud Search Service，简称 ES），为您提供托管的分布式搜索引擎服务，支持 Elasticsearch 搜索引擎，支持结构化、非结构化文本的多条件检索、统计、报表。

云搜索服务会自动部署，快速创建 Elasticsearch 集群。免运维，内置搜索调优实践；拥有完善的监控体系，提供一系列系统、集群以及查询性能等关键指标，让用户更专注于业务逻辑的实现。

14.2 什么是区域和可用区

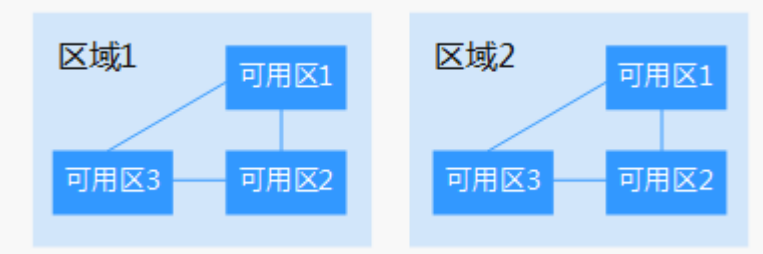
什么是区域、可用区？

我们用区域和可用区来描述数据中心的位置，您可以在特定的区域、可用区创建资源。

- 区域（Region）：从地理位置和网络时延维度划分，同一个 Region 内共享弹性计算、块存储、对象存储、VPC 网络、弹性公网 IP、镜像等公共服务。Region 分为通用 Region 和专属 Region，通用 Region 指面向公共租户提供通用云服务的 Region；专属 Region 指只承载同一类业务或只面向特定租户提供业务服务的专用 Region。
- 可用区（AZ，Availability Zone）：一个 AZ 是一个或多个物理数据中心的集合，有独立的风火水电，AZ 内逻辑上再将计算、网络、存储等资源划分成多个集群。一个 Region 中的多个 AZ 间通过高速光纤相连，以满足用户跨 AZ 构建高可用性系统的需求。

图 14-1 阐明了区域和可用区之间的关系。

图14-1 区域和可用区



如何选择区域？

建议就近选择靠近您或者您的目标用户的区域，这样可以减少网络时延，提高访问速度。

如何选择可用区？

是否将资源放在同一可用区内，主要取决于您对容灾能力和网络时延的要求。

- 如果您的应用需要较高的容灾能力，建议您将资源部署在同一区域的不同可用区内。
- 如果您的应用要求实例之间的网络延时较低，则建议您将资源创建在同一可用区内。

区域和终端节点

当您通过 API 使用资源时，您必须指定其区域终端节点。请向企业管理员获取区域和终端节点信息。

14.3 云搜索服务适用哪些场景

云搜索服务适用于日志搜索和分析、时空检索、时序检索和报表、智能搜索等场景。

14.4 云搜索服务如何保证数据和业务运行安全

云搜索服务主要从以下几个方面保障数据和业务运行安全：

- 网络隔离

  整个网络划分为 2 个平面，即业务平面和管理平面。两个平面采用物理隔离的方式进行部署，保证业务、管理各自网络的安全性。
  - 业务平面：主要是集群的网络平面，支持为用户提供业务通道，对外提供数据定义、索引、搜索能力。
  - 管理平面：主要是管理控制台，用于管理云搜索服务。
- 主机安全

  云搜索服务提供如下安全措施：
  - 通过 VPC 安全组来确保 VPC 内主机的安全。

- 通过网络访问控制列表（ACL），可以允许或拒绝进入和退出各个子网的网络流量。
- 内部安全基础设施（包括网络防火墙、入侵检测和防护系统）可以监视通过 IPsec VPN 连接进入或退出 VPC 的所有网络流量。

- 数据安全

  在云搜索服务中，通过多副本、集群跨 az 部署、索引数据第三方（OBS）备份功能保证用户的数据安全。

14.5 用户平时需要关注云搜索服务的哪些监控指标

用户需要关注的监控指标为磁盘使用率和集群健康状态。用户可以登录到云监控服务，根据实际应用场景配置告警提示，当收到告警，可采取相应措施消除告警。告警配置方法，请参见《云搜索服务用户指南》中的创建告警规则。

配置示例：

- 如果在某段时间内（如 5min），磁盘使用率出现多次（如 5 次）不低于某特定值（如 85%）的情况，则发出相应告警。
- 如果在某段时间内（如 5min），集群健康状态出现多次（如 5 次）大于 0 的情况，则发出相应告警。

采取措施：

- 收到与磁盘使用率有关的告警时，可以调查磁盘空间消耗，查看是否可以从集群节点中删除数据或是将数据存档到其他系统以释放空间，或者扩容磁盘。
- 收到与集群健康状态有关的告警时，可以查看集群的分片分配是否正常以及 Shard 是否已丢失，在 Cerebro 上查看进程是否发生重启。

14.6 云搜索服务有哪些存储选项

云搜索服务采用 EVS 和本地磁盘存储用户的索引。在集群创建过程中，用户可指定 EVS 的类型及规格（即卷大小）。

- 支持 EVS 类型有普通 I/O、高 I/O、超高 I/O。
- 针对不同的 ECS，其对应的 EVS 卷大小限制根据创建集群选择的节点规格而定。

14.7 云搜索服务存储容量的上限是多少

创建集群过程中，最少可创建 1 个节点，最多可创建 200 个节点，其中每个节点（对应一个 ECS）可挂载一定数量的 EVS。可参考不同 ECS 挂载 EVS 卷大小的不同，计算出云搜索服务存储容量的总大小，EVS 卷大小根据创建集群选择的节点规格而定。

14.8 申请的集群节点磁盘空间会有哪些开销

占用集群节点磁盘空间的日志及文件如下所示：

- 日志文件：Elasticsearch 日志
- 数据文件：Elasticsearch 索引文件
- 其他文件：集群配置文件
- 操作系统：默认余留 5%的存储空间

14.9 有哪些工具可以使用云搜索服务

管理云搜索服务，或使用其搜索引擎的 API，提供了如下三种方式。可以基于已构建好的请求消息发起请求。

- curl

  curl 是一个命令行工具，用来执行各种 URL 操作和信息传输。curl 充当的是 HTTP 客户端，可以发送 HTTP 请求给服务端，并接收响应消息。curl 适用于接口调试。关于 curl 详细信息请参见 https://curl.haxx.se/。

- 编码

  通过编码调用接口，组装请求消息，并发送处理请求消息。

- REST 客户端

  Mozilla Firefox、Google Chrome 都为 REST 提供了图形化的浏览器插件，发送处理请求消息。

  - 针对 Firefox，请参见 Firefox REST Client。
  - 针对 Chrome，请参见 Postman。

14.10 云搜索服务支持哪个 Elasticsearch 版本

云搜索服务中 Elasticsearch 搜索引擎目前支持 Elasticsearch 5.5.1、6.2.3、6.5.4、7.1.1、7.6.2 和 7.9.3 版本，Kibana 目前支持 5.5.1、6.2.3、6.5.4、7.1.1、7.6.2 和 7.9.3 版本。

14.11 云搜索服务支持哪些访问方式

云搜索服务支持以下访问方式：

- Restful API
- Transport Client
- Rest Client

  使用 Transport Client 访问云服务时，需要确保客户端与服务端的版本一致，否则可能出现无法访问的情况。

  Transport Client 不推荐使用，建议使用 High Level Rest Client。

14.12 云搜索服务中是否支持开源 Elasticsearch 的 API 或功能

支持的 Elasticsearch 版本为 5.5.1、6.2.3、6.5.4、7.1.1、7.6.2 和 7.9.3，其部署模式为 Elasticsearch Cloud 模式。支持 Kibana 工具与 Elasticsearch 的交互，功能与开源 Elasticsearch 5.5.1、6.2.3、6.5.4、7.1.1、7.6.2 和 7.9.3 相同。

14.13 云搜索服务是否支持 Logstash 对接

支持 Logstash 对接，Logstash 版本建议使用 5.5.1、6.2.3、6.5.4、7.1.1、7.6.2 和 7.9.3，且用户需要申请一台 ECS 来安装 Logstash 并进行配置。

14.14 ECS 无法连接到集群

遇到该问题，请按照如下操作步骤排查解决。

1. 先确认 ECS 实例和集群是否在同一个 VPC。
   - 如果在，执行步骤 2。
   - 如果不在，需要重新创建 ECS 实例，使之和集群在同一个 VPC 下。
2. 查看集群的安全组的出方向和入方向是否已允许 9200 端口（TCP 协议），或者允许的端口范围已包含 9200 端口（TCP 协议）。
   - 如果是，执行步骤 3。
   - 如果不是，请前往 VPC 页面，设置“安全组”的出方向和入方向已允许 9200 端口或允许的端口范围已包含 9200 端口。
3. 查看 ECS 实例是否添加安全组。
   - 如果有，检查安全组的配置规则是否满足要求，请参见“集群管理”中集群信息表格中“安全组”的描述，然后执行步骤 4。
   - 如果没有，从 ECS 的实例详情页面，进入 VPC 页面，选择“安全组”，添加安全组。
4. 在 ECS 实例上，测试是否可以正常连接到集群。

   ssh *< 节点的内网访问地址和端口号>*

   说明

   当集群包含多个节点时，需要逐个节点测试是否可以正常连接到该集群中的每个节点。

   - 如果可以通信，说明网络是正常的。
   - 如果端口不通，请联系技术支持协助排查。

14.15 云搜索服务支持哪些搜索功能

云搜索服务支持的搜索功能包括强大的全文检索，高亮显示，切面搜索，近实时索引，动态聚类，丰富的文档（如 Word、PDF 等格式）处理和地理信息搜索等。

Elasticsearch 搜索引擎支持的搜索功能请参见《Elasticsearch：权威指南》文档的“深入搜索”章节。

14.16 为什么集群创建失败

集群创建失败原因有如下 3 种：

- 资源配额不足，无法创建集群。建议申请足够的资源配额。
- 如果集群配置信息中，“安全组”的“端口范围/ICMP 类型”不包含“9200”端口，导致集群创建失败。请修改安全组信息或选择其他可用安全组。
- 7.6.2 以及 7.6.2 之后的版本，集群内通信端口 9300 默认开放在用户 VPC 的子网上面。创建集群时需要确认所选安全组是否放通子网内的 9300 通信端口，如果未放通，请修改安全组信息或选择其他可用安全组。

14.17 无法备份索引

索引的备份是通过创建集群快照实现的。遇到无法备份索引问题，请按照如下操作步骤排查解决。

排查集群的创建时间

1. 登录云搜索服务管理控制台。
2. 在左侧导航栏，单击“集群管理”。
3. 在“集群管理”页面上集群列表中的“创建时间”列，查看待备份索引的集群的创建时间。
   - 如果创建时间早于 2018 年 3 月 10 日，则创建该集群时备份与恢复索引功能尚未上线，当前无法为该集群备份索引。
   - 如果创建时间晚于 2018 年 3 月 10 日，则需要排查当前登录所用的账号或 IAM 用户是否具有备份索引的权限，具体操作请参见排查是否有权限。

排查是否有权限

1. 登录统一身份认证服务管理控制台。
2. 查看当前登录所用的账号或 IAM 用户所属的用户组。

   具体操作请参见《统一身份认证服务用户指南》中的“如何查看或修改用户信息”章节。
3. 查看用户组的权限中是否包含：“全局服务”中“对象存储服务”项目的“OBS Administrator”权限、当前所属区域的“Elasticsearch Administrator”权限。

具体操作请参见《统一身份认证服务用户指南》中的“如何查看或修改用户组”章节。

- 如果用户组的权限中不包含以上两个权限，请执行步骤 4。
- 如果用户组的权限中包含以上两个权限，请联系技术支持协助解决。

4. 为用户组添加：“全局服务”中“对象存储服务”项目的“OBS Administrator”权限、当前所属区域的“Elasticsearch Administrator”权限。

   具体操作请参见《统一身份认证服务用户指南》中的“如何查看或修改用户组”章节。

14.18 如何使用 Elasticsearch 自定义评分查询

通过 Elasticsearch 可以对匹配的文档进行评分。本节介绍如何使用 Elasticsearch 自定义评分查询。

操作步骤

1. 登录云搜索服务管理控制台。
2. 在左侧导航栏中，选择“集群管理”，进入集群列表页面。
3. 在集群列表页面中，单击集群“操作”列的“Kibana”。
4. 在 Kibana 的左侧导航中选择“Dev Tools”，单击“Get to work”，进入 Kibana 的 Console 界面。
5. 在 Kibana 的 Console 界面中，执行命令创建索引，并指定自定义映射来定义数据类型。

   例如：现在有“tv.json”数据文件，数据如下所示。

```
{
"tv":[
{ "name": "tv1", "description": "USB, DisplayPort", "vote": 0.98 }
{ "name": "tv2", "description": "USB, HDMI", "vote": 0.99 }
{ "name": "tv3", "description": "USB", "vote": 0.5 }
{ "name": "tv4", "description": "USB, HDMI, DisplayPort", "vote": 0.7 }
]
}
```

   可以执行如下命令，创建索引“mall”，并指定自定义映射来定义数据类型。

```
PUT /mall?pretty
{
  "mappings": {
    "tv": {
      "properties": {
        "description": {
          "type": "text",
          "fields": {
            "keyword": {
              "type": "keyword"
            }
          }
        },
        "name": {
```

```
        "type": "text",
        "fields": {
          "keyword": {
            "type": "keyword"
          }
        }
      },
      "vote": {
        "type": "float"
      }
    }
  }
}
```

6. 执行如下命令，将“tv.json”文件中的数据导入到“mall”索引中。

```
POST /mall/tv/_bulk?pretty
{ "index": {"_id": "1"}}
{ "name": "tv1", "description": "USB, DisplayPort", "vote": 0.98 }
{ "index": {"_id": "2"}}
{ "name": "tv2", "description": "USB, HDMI", "vote": 0.99 }
{ "index": {"_id": "3"}}
{ "name": "tv3", "description": "USB", "vote": 0.5 }
{ "index": {"_id": "4"}}
{ "name": "tv4", "description": "USB, HDMI, DisplayPort", "vote": 0.7 }
```

7. 自定义评分查询数据。
   - 用绝对好评率计算总分，按照总分由高到低的顺序排列搜索结果。

     假设用户想要查询有 USB 接口、HDMI 接口、DisplayPort 接口的电视机，接口出现，则记 1 分，不出现，则记 0 分，在前面的得分上乘以绝对好评率为总分。可以执行如下命令按照总分由高到低的顺序排列搜索结果。

```
GET /mall/tv/_search?pretty
{
  "query":{
    "function_score":{
      "query":{
        "bool":{
          "should":[
            {"constant_score":{
                "query":{"match":{"description":"USB"}}
            }},
            {"constant_score":{
                "query":{"match":{"description":"HDMI"}}
            }},
            {"constant_score":{
                "query":{"match":{"description":"DisplayPort"}}
            }}
          ]
        }
      },
      "field_value_factor":{
        "field":"vote",
        "factor":1
      },
      "boost_mode":"multiply",
```

```
    "max_boost":10
   }
  }
}
```

上面示例中总分计算公式：new_score = query_score * (factor * vote)

返回结果如下所示。

```
{
  "took": 13,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 4,
    "max_score": 2.1,
    "hits": [
      {
        "_index": "mall",
        "_type": "tv",
        "_id": "4",
        " score": 2.1,
        " source": {
          "name": "tv4",
          "description": "USB, HDMI, DisplayPort",
          "vote": 0.7
        }
      },
      {
        "_index": "mall",
        "_type": "tv",
        "_id": "2",
        "_score": 1.98,
        "_source": {
          "name": "tv2",
          "description": "USB, HDMI",
          "vote": 0.99
        }
      },
      {
        "_index": "mall",
        "_type": "tv",
        "_id": "1",
        "_score": 1.96,
        "_source": {
          "name": "tv1",
          "description": "USB, DisplayPort",
          "vote": 0.98
        }
      },
      {
        "_index": "mall",
        "_type": "tv",
```

```
      "_id": "3",
      "_score": 0.5,
      "_source": {
        "name": "tv3",
        "description": "USB",
        "vote": 0.5
      }
    }
   ]
  }
}
```

从上面返回结果可以看出 Elasticsearch 搜索引擎使用绝对好评率计算电视机的总分，并将电视机按照总分由高到低的顺序排列显示。

- 用相对好评率计算总分，按照总分由高到低的顺序排列搜索结果。

  假设用户想要查询有 USB 接口、HDMI 接口、DisplayPort 接口的电视机，接口出现，则记 1 分，不出现，则记 0 分，在前面的得分上乘以相对好评率为总分。为了避免不正常的好评率对查询结果的影响，不用绝对好评率作为评分因素，设定好评率的门限值为 0.8，高于门限值时，好评率按 1 计算，低于或等于门限值时，好评率按 0.5 计算。可以执行如下命令按照总分由高到低的顺序排列搜索结果。

```
GET /mall/tv/_search?pretty
{
  "query":{
    "function_score":{
      "query":{
        "bool":{
          "should":[
            {"constant_score":{
                "query":{"match":{"description":"USB"}}
            }},
            {"constant_score":{
                "query":{"match":{"description":"HDMI"}}
            }},
            {"constant_score":{
                "query":{"match":{"description":"DisplayPort"}}
            }}
          ]
        }
      },
      "script_score":{
        "script":{
          "params":{"threshold":0.8},
          "inline":"if (doc[\"vote\"].value > params.threshold) {return 1;}
return 0.5;"
        }
      },
      "boost_mode":"multiply",
      "max_boost":10
    }
  }
}
```

上面示例中总分计算公式：new_score = query_score * vote（当 vote > 0.8 时，取值为 1；当 vote ≤ 0.8 时，取值为 0.5。）

返回结果如下所示。


```
{
  "took": 634,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "failed": 0
  },
  "hits": {
    "total": 4,
    "max_score": 2,
    "hits": [
      {
        "_index": "mall",
        "_type": "tv",
        "_id": "2",
        "_score": 2,
        "_source": {
          "name": "tv2",
          "description": "USB, HDMI",
          "vote": 0.99
        }
      },
      {
        "_index": "mall",
        "_type": "tv",
        "_id": "1",
        "_score": 2,
        "_source": {
          "name": "tv1",
          "description": "USB, DisplayPort",
          "vote": 0.98
        }
      },
      {
        "_index": "mall",
        "_type": "tv",
        "_id": "4",
        "_score": 1.5,
        "_source": {
          "name": "tv4",
          "description": "USB, HDMI, DisplayPort",
          "vote": 0.7
        }
      },
      {
        " index": "mall",
        " type": "tv",
        " id": "3",
        " score": 0.5,
        "_source": {
```

```
        "name": "tv3",
        "description": "USB",
        "vote": 0.5
      }
    }
  ]
 }
}
```

从上面返回结果可以看出 Elasticsearch 搜索引擎使用相对好评率计算电视机的总分，并将电视机按照总分由高到低的顺序排列显示。

14.19 9200 端口访问失败

问题现象

通过 VPN 专线或 VPC 的对等连接访问 ES 集群的场景下，使用 curl 命令接入 ES 集群时，无返回结果。

例如，执行如下命令接入集群，无返回结果。

```
curl -s 'http://<节点内网访问地址>:9200'
```

原因分析

在“使用 VPN 专线访问 ES 集群”或“通过 VPC 的对等连接访问 ES 集群”场景下，其所在的客户端与 ES 不在同一 VPC 下。因此，要求 ES 集群的子网与其 VPC 具有不同的网段。

例如，某一 ES 集群，选用的 VPC 为 vpc-8e28，其网络配置为 192.168.0.0/16。选用了此 VPC 下的子网 subnet-4a81，subnet-4a81 子网的网段与 vpc-8e28 一致，均为 192.168.0.0/16。此时，如果使用 VPN 专线访问 ES 集群或通过 VPC 的对等连接访问 ES 集群，会导致此子网创建的机器内没有该 VPC 对应的网关，从而影响 ES 服务的默认路由的设置，最终导致 9200 端口访问失败。

处理步骤

当出现 9200 端口访问失败错误时，且 ES 集群状态为可用状态。执行步骤如下所示：

1. 进入 ES 服务管理控制台，在集群列表中，单击集群名称进入集群详情页面，查看此集群使用的 VPC 和子网。
2. 进入 VPC 服务管理控制台，在虚拟私有云列表中，单击 ES 集群使用的 VPC 名称，进入 VPC 详情页面。查看 VPC 和子网的网段信息。

   如图 14-2 所示，VPC 的网段信息，与子网的网段信息一致。在使用 VPN 专线访问或使用 VPC 对等连接访问时，会导致 9200 端口访问失败。

图14-2 查看网段信息

3. 如果出现上述错误，请重新创建集群，并选择一个网段与 VPC 不同的子网，如不存在这样的子网，请在 VPC 管理控制台重新创建一个子网。

   创建新的 ES 集群后，将旧集群的数据迁移至新集群中，然后再通过 VPN 专线访问或使用 VPC 对等连接访问使用。

说明

如果需要 VPN 专线访问或使用 VPC 对等连接访问 ES 集群时，请务必保证，新创建的 ES 集群，其 VPC 与子网，具备不同的网段信息。

14.20 Elasticsearch 针对 filebeat 配置调优

问题现象

filebeat 是性能非常出色的文件采集工具，绝大多数的业务日志可以很容易的在 1 秒内收集至 elasticsearch 内，但是个别日志量大的业务日志无法及时收集，按照官方的默认配置通常 1 核 CPU 分配给 filebeat 时，写 ES 的速率低于 1M/S，这里可以针对 filebeat.yml 配置文件做优化，提高写入 ES 的性能。

原因分析

filebeat.yml 的默认配置比较保守，在日志量很大的业务场景，需要修改 filebeat.yml 参数进行调优。

处理步骤

1. 针对 filebeat.yml 配置文件做参数优化，调整 input 端配置：

   #根据实际情况调大 harvester_buffer_size 参数（该参数是指每个 harvester 监控文件时，使用的 buffer 大小）。

   harvester_buffer_size:40960000

   #根据实际情况调大 filebeat.spool_size 参数（该参数是指 spooler 的大小，一次 Publish，上传多少行的日志量）。

   filebeat.spool_size:250000

   #根据实际情况调整 filebeat.idle_timeout 参数（该参数是指 spooler 的超时时间，如果到了超时时间，不论是否到达容量阈值，spooler 会清空发送出去）。

   filebeat.idle_timeout:1s

2. 针对 filebeat.yml 配置文件做参数优化，调整 output.elasticsearch 端配置：

   #根据实际情况将 worker 参数调整为跟 ES 个数一致（该参数是指对应 ES 的个数，默认 1）。

   worker:1

   #根据实际情况调大 bulk_max_size 参数（该参数是指单个 elasticsearch 批量 API 索引请求的最大事件数，默认是 50）。

   bulk_max_size:15000

   #根据实际情况调整 flush_interval 参数（该参数是指新事件两个批量 API 索引请求之间需要等待的秒数，如果 bulk_max_size 在该值之前到达，额外的批量索引请求生效）。

   flush_interval:1s

14.21 集群使用故障

问题现象

Es-event 集群单击进入 kibana 后，会出现一直卡在加载页面中，不能进入 kibana 控制台。

原因分析

浏览器缓存导致，清理缓存。

处理步骤

1. 登录云搜索服务管理控制台。
2. 在左侧导航栏，单击“集群管理”。
3. 在集群对应的“操作”列，单击“Kibana”，打开 Kibana 界面。
4. 在 kibana 页面按 F12。
5. 单击“Network”，选中“data:image”，右键选择“clear browser cache”，弹出对话弹框，单击确定，关闭 Kibana 界面。
6. 在集群对应的“操作”列，单击“Kibana”即可访问。

14.22 如何使用 NAT 网关实现云搜索服务公网访问

开通公网访问云搜索服务操作视图：

1.获取云搜索服务信息

2.配置 NAT 网关

3.修改云搜索服务安全组规则

4.通过公网访问云搜索服务

> **⚠ 注意**
>
> 如果非安全模式集群使用此功能，则会把集群数据直接暴露到公网，请禁用此功能。

获取云搜索服务信息

步骤 1 获取云搜索服务内网访问地址。

登录云搜索服务的 Console 控制台，在集群创建成功后，可在集群管理页面获取集群访问的“内网访问地址”。

步骤 2 获取 VPC 和子网信息。

单击集群“名称/ID”，进入集群“基本信息”页面，获取 VPC 和子网信息。

----结束

配置 NAT 网关

步骤 1 创建 NAT 网关。

1. 登录 Console 控制台，选择“服务列表”>“网络”>“NAT 网关”，进入网络控制台页面。
2. 单击“购买 NAT 网关”，配置 NAT 网关的相关信息。详细请参考《NAT 网关用户指南》**购买 NAT 网关**。

> **说明**
>
> “虚拟私有云”和“子网”配置为步骤 2 获取的信息。

3. 配置完成后，单击“立即购买”。

步骤 2 添加 DNAT 规则。

1. NAT 网关购买成功后，在 NAT 控制台，单击购买成功的 NAT 网关“名称”，进入 NAT 网关详情页面。
2. 选择“DNAT 规则”页签，单击“添加 DNAT 规则”。详细请参考《NAT 网关用户指南》**添加 DNAT 规则**。

说明

- 弹性公网 IP：可以根据自己业务在弹性公网 IP 页面创建。
- 公网端口：可以自定义。
- 私网 IP：云搜索服务的内网访问 IP，即步骤 1 获取的“内网访问地址”。
- 私网端口：9200
- 如果创建的集群包含多个“内网访问地址”，则需要添加多个 DNAT 规则。

3. 添加完成后，单击“确定”。

----结束

修改云搜索服务安全组规则

步骤 1 登录云搜索服务的 Console 控制台，选择对应的集群，单击集群“名称/ID”，进入集群“基本信息”页面。

步骤 2 在“基本信息”页面，单击“安全组”。

步骤 3 在“安全组”页面，选择“入方向规则”。

步骤 4 单击“添加规则”，添加 9200 端口入方向规则。

步骤 5 配置完成后，单击“确定”。

----结束

通过公网访问云搜索服务

步骤 1 在浏览器中输入 https://IP:port 或者 http://IP:port，访问云搜索服务。

- IP:port：弹性公网 IP:端口号，即创建 DNAT 规则设置的弹性公网 IP 和公网端口。
- 如果集群开启了**安全模式**，请输入 https://IP:port，并且输入安全模式的用户名和密码。
- 如果集群未开启**安全模式**，请输入 https://IP:port。

----结束

A 修订记录

表A-1 修订记录

发布日期	修订记录
2021-09-09	第一次正式发布。