Amazon Managed Streaming for Apache Kafka

开发人员指南

亚马逊云科技

Amazon Managed Streaming for Apache Kafka: 开发人员指南

Table of Contents

欢迎使用 Amazon MSK 开发人员指南

欢迎使用Amazon MSK 开发人员指南. 以下主题可帮助您以尝试执行的操作为基础开始使用本指南。

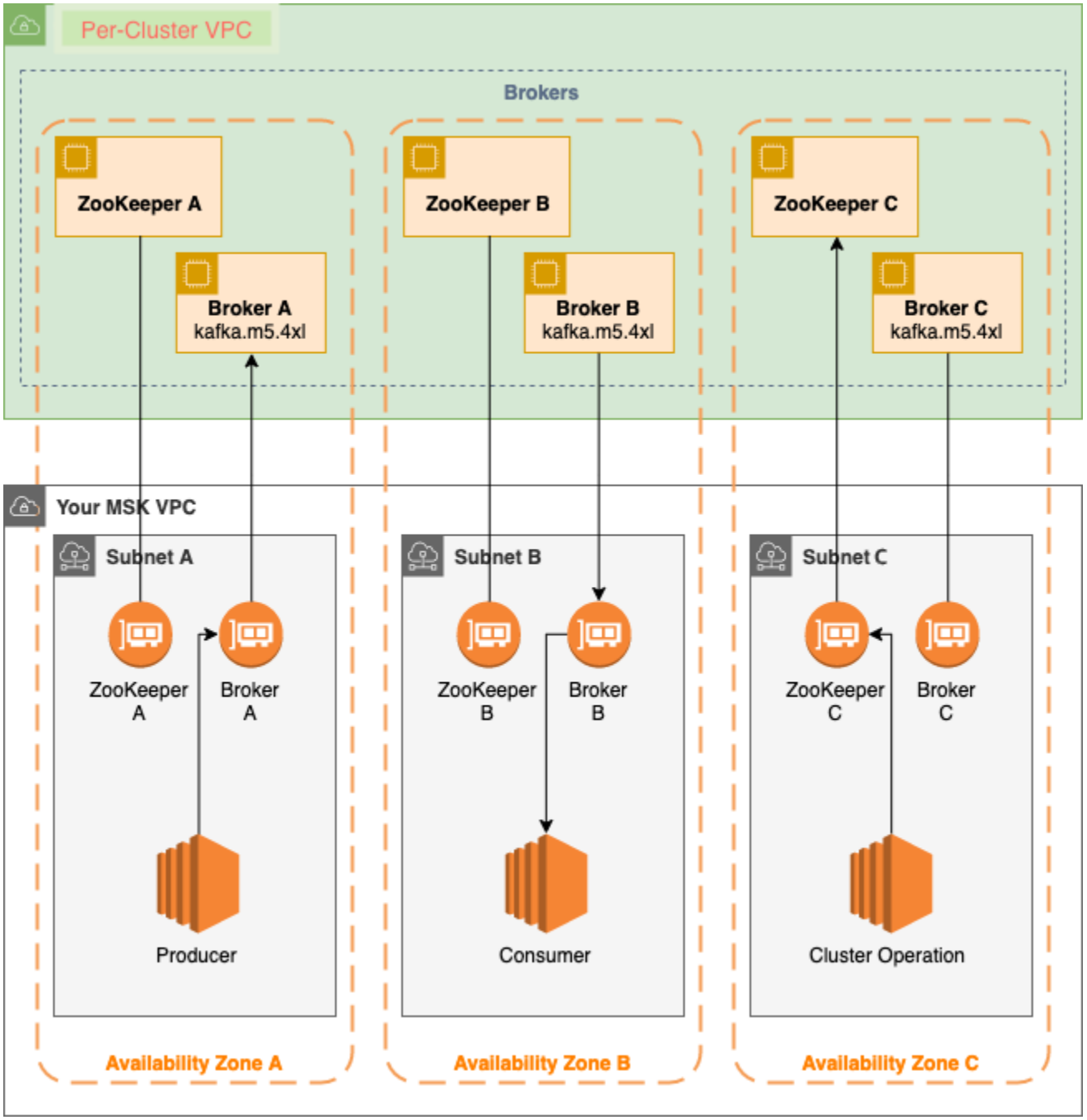
- 按照以下步骤创建 Amazon MSK 集群Amazon MSK 入门 (p. 5)教程。
- 在中深入了解 Amazon MSK 的功能Amazon MSK：工作方式 (p. 9).
- 无需管理和扩展集群容量即可运行 Apache KafkaMSK 无服务器 (p. 42).

有关亮点、产品详细信息和定价，请参阅 Service 页面以获取Amazon MSK.

什么是 Amazon MSK？

Amazon Managed Streaming for Apache Kafka（Amazon MSK）是一项完全托管式服务，让您能够构建并运行使用 Apache Kafka 来处理串流数据的应用程序。Amazon MSK 提供控制层面操作，例如，用于创建、更新和删除集群的操作。它允许您使用 Apache Kafka 数据层面操作，例如，用于生成和使用数据的操作。它运行 Apache Kafka 的开源版本。这意味着支持来自合作伙伴和 Apache Kafka 社区的现有应用程序、工具和插件，而无需更改应用程序代码。您可以使用 Amazon MSK 创建使用以下列出的任何 Apache Kafka 版本的集群。the section called “支持的 Apache Kafka 版本” (p. 111).

下图概述了 Amazon MSK 的工作原理。



该图演示了以下各个组件之间的交互：

- 代理节点— 创建 Amazon MSK 集群时，您可以指定希望 Amazon MSK 在每个可用区中创建的代理节点数。在此图显示的示例集群中，每个可用区有一个代理。每个可用区都有自己的 Virtual Private Cloud (VPC) 子网。
- ZooKeeper 节点— 亚马逊 MSK 还创建了 Apache ZooKeeper 节点给你。阿帕奇 ZooKeeper 是一个开源服务器，可实现高度可靠的分布式协调。
- 制作者、消费者和主题创作者— Amazon MSK 允许您使用 Apache Kafka 数据层面操作来创建主题以及生成和使用数据。

- 集群操作您可以使用Amazon Web Services Management Console，Amazon Command Line Interface(Amazon CLI) 或软件开发工具包中的 API 来执行控制层面操作。例如，您可以创建或删除 Amazon MSK 集群、列出账户中的所有集群、查看集群属性以及更新集群中代理的数量和类型。

Amazon MSK 检测集群最常见的故障情形并自动恢复，这样，生成器和使用器应用程序能够继续执行其写入和读取操作，而产生最小影响。当 Amazon MSK 检测到代理故障时，它会解决故障或用新的代理替换不正常或无法访问的代理。此外，如果可能，它会重用旧代理的存储来减少 Apache Kafka 需要复制的数据。可用性影响将仅限于 Amazon MSK 完成检测和恢复所需的时间。恢复后，生成器和使用器应用程序可以继续与发生故障前使用的相同代理 IP 地址进行通信。

设置 Amazon MSK

首次使用 Amazon MSK 前，请完成以下任务。

任务

- 注册Amazon (p. 4)
- 下载库和工具 (p. 4)

注册Amazon

当你注册时Amazon，将在中为您的 Amazon Web Services 账户自动注册所有服务Amazon，包括亚马逊 MSK。您只需为使用的服务付费。

如果您已有 Amazon 账户，请跳到下一个任务。如果您还没有 Amazon 账户，请使用以下步骤创建。

注册 Amazon Web Services 账户

1. 打开 https://portal.aws.amazon.com/billing/signup。
2. 按照屏幕上的说明进行操作。

   在注册时，您将接到一通电话，要求您使用电话键盘输入一个验证码。

下载库和工具

以下库和工具可帮助您使用 Amazon MSK：

- 这些区域有：Amazon Command Line Interface(Amazon CLI) 支持 Amazon MSK。这些区域有：Amazon CLI利用，您可以从命令行控制多个 Amazon Web Services 并通过脚本自动执行这些服务。升级 Amazon CLI将升级到最新版本，确保它支持本用户指南中介绍的 Amazon MSK 功能。有关如何升级的详细说明Amazon CLI，请参阅安装Amazon Command Line Interface. 安装完Amazon CLI，你必须对其进行配置。有关如何配置Amazon CLI，请参阅Amazon 配置.
- 这些区域有：Amazon Managed Streaming for Kafka API 参考记录 Amazon MSK 支持的 API 操作。
- 用于的 Amazon Web Services SDK转到、Java、JavaScript、.NET、Node.js、PHP、Python, 和红宝石包括亚马逊 MSK 支持和示例。

Amazon MSK 入门

本教程向您展示了如何创建 MSK 集群、生成和使用数据以及使用指标监控集群运行状况的示例。此示例并不代表您在创建 MSK 集群时可以选择的所有选项。为了简单起见，我们在本教程的各个部分均选择默认选项。这并不意味着它们是设置 MSK 集群或客户端实例的唯一选项。

主题

第 1 步：创建Amazon MSK 集群

在此步骤中开始使用亚马逊 MSK (p. 5)，您创建一个 Amazon MSK 集群。

要创建 Amazon MSK 集群Amazon Web Services Management Console

1. 登录Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home?region=us-east-1#/home/.
2. 选择创建集群。
3. 对于创建方法，离开快速创建已选择选项。这些区域有：快速创建选项允许您使用默认设置创建集群。
4. 对于集群名称，请输入集群的描述性计划名称。例如，**`MSKTutorialCluster`**。
5. 对于集群常规属性，选择预置作为集群类型.
6. 从下面的桌子上看所有集群设置，请复制以下设置的值并保存它们，因为您将在本教程的后面部分中使用它们：
   - VPC
   - 子网
   - 与 VPC 关联的安全组
7. 选择创建集群。
8. 检查集群状态在集群摘要页面。状态从创建到处于活动状态因为亚马逊 MSK 正在配置集群。当状态为处于活动状态，您可以连接到集群。有关集群状态的更多信息，请参阅集群状态 (p. 49).

下一步

第 2 步：创建客户端计算机 (p. 5)

第 2 步：创建客户端计算机

在此步骤中开始使用亚马逊 MSK (p. 5)，你创建一台客户端计算机。可以使用此客户端计算机创建生成和使用数据的主题。为简单起见，您将在与 MSK 集群关联的 VPC 中创建此客户端计算机，以便该客户端可以轻松连接到集群。

创建客户端计算机

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 选择 Launch instances。
3. 输入 a名称用于您的客户端计算机，例如**`MSKTutorialClient`**.
4. 离开亚马逊 Linux 2 AMI (HVM)-内核 5.10，固态硬盘卷类型SELECTAmazon Machine Machine（.
5. 离开t2.micro已选择实例类型。
6. 下面密钥对（登录），选择创建新的key pair. ENTER**`MSKKeyPair`**为了密钥对名称，然后选择下载密钥对. 此外，您还可使用现有密钥对。
7. 选择 Launch instance (启动实例)。
8. 选择查看实例。然后，在安全组列中，选择与新实例关联的安全组。复制安全组的 ID，然后保存，以供稍后使用。
9. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
10. 在导航窗格中，选择 Security Groups（安全组）。找到您保存 ID 的安全组the section called “第 1 步：创建 集群” (p. 5).
11. 在里面入站规则选项卡，选择编辑入站规则.
12. 选择 Add rule。
13. 在新规则中，选择所有流量在里面类型列。在第二个字段中源列中，选择您的客户端计算机的安全组。这是您在启动客户端计算机实例后保存其名称的组。
14. 选择 Save rules（保存规则）。现在，集群的安全组可以接受来自客户端计算机安全组的流量。

下一步

第 3 步：创建主题 (p. 6)

第 3 步：创建主题

在此步骤中开始使用亚马逊 MSK (p. 5)，你在客户端计算机上安装 Apache Kafka 客户端库和工具，然后创建一个主题。

在客户端计算机上创建主题

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在导航窗格中，选择 Instances (实例)。然后选中您在中创建的客户端计算机名称旁边的复选框第 2 步：创建客户端计算机 (p. 5).
3. 选择 Actions (操作)，然后选择 Connect (连接)。按照控制台中的说明连接到客户端计算机。
4. 通过运行以下命令在客户端计算机上安装 Java：

```
sudo yum install java-1.8.0
```

5. 运行以下命令以下载 Apache Kafka。

```
wget https://archive.apache.org/dist/kafka/2.6.2/kafka_2.12-2.6.2.tgz
```

   Note

   如果您希望使用此命令中使用的镜像站点之外的镜像站点，则可在 Apache 网站上选择其他镜像站点。

6. 在上一步中将 TAR 文件下载到的目录中运行以下命令。

```
tar -xzf kafka_2.12-2.6.2.tgz
```

7. 转到ka_2.12-2.6.2目录。
8. 在以下位置打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
9. 等待集群的状态变为处于活动状态. 这可能需要花几分钟的时间。状态变为之后处于活动状态，请选择集群名称。这将带您进入包含集群摘要的页面。
10. 选择查看客户信息.
11. 复制用于明文身份验证的连接字符串。
12. 运行以下命令，替换*BootstrapServerString*使用您在上一步中获得的连接字符串。

```
<path-to-your-kafka-installation>/bin/kafka-topics.sh --create --bootstrap-
server BootstrapServerString --replication-factor 3 --partitions 1 --topic
 MSKTutorialTopic
```

如果此命令成功，您将看到以下消息：`Created topic MSKTutorialTopic.`

下一步

第 4 步：生成和使用数据 (p. 7)

第 4 步：生成和使用数据

在处登录开始使用亚马逊 MSK (p. 5)，您生成和使用数据。

生成和使用消息

1. 转到`bin`在客户端计算机上安装 Apache Kafka 的文件夹，然后创建一个名为`client.properties`具有以下内容。

```
security.protocol=PLAINTEXT
```

2. 运行以下命令以启动控制台生成器。Replace（替换）*BootstrapServerString*使用您在中获得的纯文本连接字符串the section called “第 3 步：创建主题” (p. 6). 有关如何检索此连接字符串的说明，请参阅获取 Amazon MSK 集群的引导代理 (p. 14).

```
<path-to-your-kafka-installation>/bin/kafka-console-producer.sh --broker-
list BootstrapServerString --producer.config client.properties --topic MSKTutorialTopic
```

3. 输入所需的任何消息，然后按 Enter。重复执行此步骤两次或三次。每次输入一行并按 Enter 时，该行会作为单独的消息发送到您的 Apache Kafka 集群。
4. 将与客户端计算机的连接保持打开状态，然后在新窗口中打开与该计算机的第二个单独连接。
5. 在以下命令中，替换*BootstrapServerString*使用您之前保存的纯文本连接字符串。然后，要创建控制台消费者，请在第二次连接到客户端计算机时运行以下命令。

```
<path-to-your-kafka-installation>/bin/kafka-console-consumer.sh --bootstrap-
server BootstrapServerString --consumer.config client.properties --topic
 MSKTutorialTopic --from-beginning
```

您开始看到之前使用控制台生成器命令时输入的消息。

6. 在生成器窗口中输入更多消息，并观察消息显示在使用器窗口中。

下一步

第 5 步：使用 Amazon CloudWatch 查看亚马逊 MSK 指标 (p. 8)

第 5 步：使用 Amazon CloudWatch 查看亚马逊 MSK 指标

在处登录开始使用亚马逊 MSK (p. 5)，你看看亚马逊中的亚马逊 MSK 指标 CloudWatch.

要查看亚马逊 MSK 指标，请访问 CloudWatch

1. 打开 CloudWatch 控制台https://console.aws.amazon.com/cloudwatch/.
2. 在导航窗格中，选择 Metrics（指标）。
3. 选择所有指标选项卡，然后选择AWS/Kafka.
4. 要查看代理级别的指标，请选择 Broker ID, Cluster Name (代理 ID，集群名称)。对于集群级别的指标，请选择 Cluster Name (集群名称)。
5. （可选）在图表窗格中，选择统计数据和时间段，然后创建一个 CloudWatch 使用这些设置发出警报。

下一步

第 6 步：删除Amazon为本教程创建的资源 (p. 8)

第 6 步：删除Amazon为本教程创建的资源

在最后一步中开始使用亚马逊 MSK (p. 5)，则删除您为本教程创建的 MSK 集群和客户端计算机。

要删除资源，请使用Amazon Web Services Management Console

1. 打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择集群的名称。例如，MSKTutorialCluster.
3. 选择 Actions (操作)，然后选择 Delete (删除)。
4. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
5. 选择您为客户端计算机创建的实例，例如，MSKTutorialClient.
6. 选择实例状态，然后选择终止实例.

Amazon MSK：工作方式

Amazon MSK 集群是您可以在账户中创建的主要 Amazon MSK 资源。本节中的主题介绍如何执行常见的 Amazon MSK 操作。有关可以在 MSK 集群上执行的所有操作的列表，请参阅以下内容：

- 这些区域有： Amazon Web Services Management Console
- 这些区域有：Amazon MSK 参考
- 这些区域有：Amazon MSK CLI 命令参考

主题

创建 Amazon MSK 集群

> **Important**
>
> 创建集群之后无法更改 Amazon MSK 集群的 VPC。

在创建 Amazon MSK 集群之前，您需要拥有一个Amazon Virtual Private Cloud(VPC)，并在该 VPC 内设置子网。

在美国西部（加利福尼亚北部）区域中，您需要两个不同可用区中的两个子网。在提供 Amazon MSK 的所有其他区域中，您可以指定两个或三个子网。您的子网必须位于不同的可用区中。创建集群时，Amazon MSK 将代理节点平均分布到您指定的子网中。

代理代理代理引擎

在创建 Amazon MSK 集群时，您可以指定您希望它具有的代理类型。亚马逊 MSK 支持以下代理类型：

- kafka.t3.small
- ka.m4x4xka.m4xka.m4xka.m4xka.m4xka.m4xmarge、ka.m4x4xka.m4xka.m4xxa.m4xxarge、ka.m4x4xka.m4x4xka.m

M5 代理具有比 T3 代理更高的基准吞吐量性能，建议用于生产工作负载。M5 代理还可具有比 T3 代理更多的每代理分区。如果您正在运行较大的生产级工作负载或需要更多的分区，请使用 M5 代理。要了解有关 M5 实例类型的更多信息，请参阅 Amazon EC2 M5 实例。

T3 代理可以使用 CPU 积分来临时提高性能。如果您正在测试中小型流式处理工作负载，或者您的低吞吐量流式处理工作负载会临时出现吞吐量高峰，则可以使用 T3 代理进行低成本开发。建议运行 proof-of-concept 测试来确定 T3 代理是否足以应对生产或关键工作负载。要了解有关 T3 实例类型的更多信息，请参阅 Amazon EC2 T3 实例。

有关如何选择代理类型的更多信息，请参阅最佳实践 (p. 123).

使用创建群集Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择创建集群。
3. 指定集群的名称。
4. 在 VPC 列表中，选择要用于集群的 VPC。您还可以指定您希望 Amazon MSK 用于创建集群的 Apache Ka.MSK。
5. 如果您使用以下区域之一，请指定两个子网：南美洲 (圣保罗)、加拿大 (中部) 和美国西部 (加利福尼亚北部)。在提供 Amazon MSK 的其它区域中，您可以指定两个或三个子网。指定的子网必须位于不同的可用区中。
6. 选择所需配置类型。有关 MSK 配置的信息，请参阅 配置 (p. 31)。
7. 指定您希望 MSK 在每个可用区中创建的代理类型和数量。每个可用区最少一个代理，最多30个代理。
8. （可选）为您的集群分配标签。标签是可选的。有关更多信息，请参阅 the section called “为集群添加标签” (p. 28)。
9. 您可以调整每个代理的存储量。创建集群后，您可以增加每个代理的存储量，但不能减少它。
10. 选择对传输中的数据进行加密所需的设置。默认情况下，MSK 会在集群中的代理之间传输数据时对数据进行加密。如果您希望在代理之间传输数据时不要对数据进行加密，请清除带有 Enable encryption within the cluster (在集群内启用加密) 标签的复选框。
11. 选择用于在客户端和代理之间传输数据时加密数据的三种设置之一。有关更多信息，请参阅 the section called “传输中加密” (p. 51)。
12. 选择要用于静态加密数据的 KMS 密钥类型。有关更多信息，请参阅 the section called “静态加密” (p. 51)。
13. 如果要验证客户端身份，请通过选择 Enable TLS client authentication (启用 TLS 客户端身份验证) 旁边的方框来选择该选项。有关身份验证的更多信息，请参阅the section called “双向 TLS 身份验证” (p. 72)。
14. 选择所需的监控级别。此选择决定您获得的指标集。有关更多信息，请参阅 监控集群 (p. 95)。
15. (可选) 选择高级设置，然后选择自定义设置. 您可以指定要向其授予对集群的访问权限的一个或多个安全组（例如，客户端计算机的安全组）。如果您指定与您共享的安全组，则必须确保您拥有对它们的权限。具体来说，您需要 `ec2:DescribeSecurityGroups` 权限。有关示例，请参阅 。Amazon EC2：允许以编程方式和在控制台中管理关联至特定 VPC 的 EC2 安全组.
16. 选择创建集群。
17. 检查集群状态在集群摘要页. 状态从更改创建到处于活动状态因为 Amazon MSK 配置了集群。当状态为处于活动状态的更多信息，您可以连接到集群。有关集群状态的更多信息，请参阅集群状态 (p. 49).

使用创建群集Amazon CLI

1. 复制以下 JSON 并将其保存到文件中。将文件命名为 `brokernodegroupinfo.json`。将 JSON 中的子网 ID 替换为与子网对应的值。这些子网必须位于不同的可用区中。将 *`"Security-Group-ID"`* 替换为客户端 VPC 的一个或多个安全组的 ID。与这些安全组关联的客户端可以访问集群。如果您指定与您共享的安全组，则必须确保您拥有对它们的权限。具体来说，您需要

ec2:DescribeSecurityGroups 权限。有关示例，请参阅 。Amazon EC2：允许以编程方式和在控制台中管理关联至特定 VPC 的 EC2 安全组. 最后，将更新的 JSON 文件保存在已安装 Amazon CLI 的计算机上。

```
{
  "InstanceType": "kafka.m5.large",
  "ClientSubnets": [
    "Subnet-1-ID",
    "Subnet-2-ID"
  ],
  "SecurityGroups": [
    "Security-Group-ID"
  ]
}
```

Important

如果您使用以下区域之一，请恰好指定两个子网：南美洲 (圣保罗)、加拿大 (中部) 和美国西部 (加利福尼亚北部)。对于提供 Amazon MSK 的其它区域，您可以指定两个或三个子网。指定的子网必须位于不同的可用区中。在创建集群时，Amazon MSK 在您指定的子网之间平均分配代理节点。

2. 在保存 brokernodegroupinfo.json 文件的目录中运行以下 Amazon CLI 命令，并将 *"Your-Cluster-Name"* 替换为您选择的名称。对于 *"Monitoring-Level"*，您可以指定以下三个值之一：DEFAULT、PER_BROKER 或 PER_TOPIC_PER_BROKER。有关这三个不同监控级别的信息，请参阅??? (p. 95)。enhanced-monitoring 参数是可选的。如果未在 create-cluster 命令中指定该参数，监控级别即为 DEFAULT。

```
aws kafka create-cluster --cluster-name "Your-Cluster-Name" --broker-node-group-info
 file://brokernodegroupinfo.json --kafka-version "2.2.1" --number-of-broker-nodes 3 --
enhanced-monitoring "Monitoring-Level"
```

该命令的输出如以下 JSON 所示：

```
{
    "ClusterArn": "...",
    "ClusterName": "AWSKafkaTutorialCluster",
    "State": "CREATING"
}
```

Note

create-cluster 命令可能会返回错误，指示一个或多个子网所属的可用区不受支持。发生此种情况时，该错误会指示不受支持的可用区。请创建不使用不受支持的可用区的子网，然后重试 create-cluster 命令。

3. 保存 ClusterArn 键的值，因为您需要该键才能对集群执行其他操作。
4. 运行以下命令以检查集群STATE. 这些区域有：STATE值从CREATING到ACTIVE因为 Amazon MSK 配置了集群。当状态为ACTIVE的更多信息，您可以连接到集群。有关集群状态的更多信息，请参阅集群状态 (p. 49).

```
aws kafka describe-cluster --cluster-arn <your-cluster-ARN>
```

使用自定义 MSK 配置创建集群Amazon CLI

有关自定义 MSK 配置以及如何创建这些配置的信息，请参阅 配置 (p. 31)。

1. 将以下 JSON 保存到文件中，并将 *configuration-arn* 替换为创建集群要使用的配置的 ARN。

```
{
    "Arn": configuration-arn,
    "Revision": 1
}
```

2. 运行 `create-cluster` 命令并使用 `configuration-info` 选项指向您在上一步中保存的 JSON 文件。以下是示例。

```
aws kafka create-cluster --cluster-name ExampleClusterName --broker-node-group-info
 file://brokernodegroupinfo.json --kafka-version "1.1.1" --number-of-broker-nodes 3 --
enhanced-monitoring PER_TOPIC_PER_BROKER --configuration-info file://configuration.json
```

以下是运行此命令后的成功响应示例。

```
{
    "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/
CustomConfigExampleCluster/abcd1234-abcd-dcba-4321-a1b2abcd9f9f-2",
    "ClusterName": "CustomConfigExampleCluster",
    "State": "CREATING"
}
```

使用 API 创建群集

要使用 API 创建集群，请参阅CreateCluster.

删除 Amazon MSK 集群

Note

如果您的集群具有auto-scaling 策略，我们建议您在删除集群之前移除该策略。有关更多信息，请参阅 自动扩展 (p. 18)。

使用删除群集Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 通过选择要删除的 MSK 集群旁边的方框来选择该集群。
3. 选择Delete，然后确认删除。

使用删除群集Amazon CLI

运行以下命令，将*ClusterArn*和创建集群时所获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

```
aws kafka delete-cluster --cluster-arn ClusterArn
```

使用 API 删除集群

要使用 API 删除集群，请参阅DeleteCluster.

获得 Apache ZooKeeper Amazon MSK 集群的连接字符串

获得 Apache ZooKeeper 连接字符串使用Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 该表显示了此账户下当前区域的所有集群。选择集群名称以查看其说明。
3. 在存储库的集群摘要页面上，选择查看客户端信息. 这将显示引导代理，和 Apache ZooKeeper 连接字符串。

获得 Apache ZooKeeper 连接字符串使用Amazon CLI

1. 如果您不知道集群的 Amazon 资源名称 (ARN)，您可以通过列出您账户中的所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。
2. 为了得到 Apache ZooKeeper 连接字符串和有关集群的其他信息，运行以下命令，并将*ClusterArn*与集群的 ARN。

```
aws kafka describe-cluster --cluster-arn ClusterArn
```

该 `describe-cluster` 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterInfo": {
        "BrokerNodeGroupInfo": {
            "BrokerAZDistribution": "DEFAULT",
            "ClientSubnets": [
                "subnet-0123456789abcdef0",
                "subnet-2468013579abcdef1",
                "subnet-1357902468abcdef2"
            ],
            "InstanceType": "kafka.m5.large",
            "StorageInfo": {
                "EbsStorageInfo": {
                    "VolumeSize": 1000
                }
            }
        },
        "ClusterArn": "arn:aws:kafka:us-east-1:111122223333:cluster/
testcluster/12345678-abcd-4567-2345-abcdef123456-2",
        "ClusterName": "testcluster",
        "CreationTime": "2018-12-02T17:38:36.75Z",
        "CurrentBrokerSoftwareInfo": {
            "KafkaVersion": "2.2.1"
        },
        "CurrentVersion": "K13V1IB3VIYZZH",
        "EncryptionInfo": {
            "EncryptionAtRest": {
                "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:555555555555:key/12345678-
abcd-2345-ef01-abcdef123456"
            }
        },
        "EnhancedMonitoring": "DEFAULT",
        "NumberOfBrokerNodes": 3,
```

```
        "State": "ACTIVE",
        "ZookeeperConnectString": "10.0.1.101:2018,10.0.2.101:2018,10.0.3.101:2018"
    }
}
```

上一 JSON 示例在 `describe-cluster` 命令输出中显示 `ZookeeperConnectString` 键。复制与此键对应的值，并保存它以用于在集群上创建主题。

> **Important**
>
> 您的 Amazon MSK 集群必须位于`ACTIVE`状态，你才能获得 Apache ZooKeeper 连接字符串。当集群仍处于 `CREATING` 状态时，`describe-cluster` 命令的输出不包含 `ZookeeperConnectString`。如果发生这种情况，请等待几分钟，然后在集群进入 `ACTIVE` 状态后再次运行 `describe-cluster`。

获得 Apache ZooKeeper 使用 API 的连接字符串

为了得到 Apache ZooKeeper 使用 API 的连接字符串，请参阅DescribeCluster.

获取 Amazon MSK 集群的引导代理

使用获取引导程序经纪人Amazon Web Services Management Console

术语引导代理代理是指 Apache Kafka 客户端可以用作连接到集群的起点的代理列表。此列表不一定包含集群中的所有代理。

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 该表显示了此账户下当前区域的所有集群。选择集群名称以查看其说明。
3. 在存储库的集群摘要页面上，选择查看客户端信息. 这将显示引导代理，和 Apache ZooKeeper 连接字符串。

使用获取引导程序经纪人Amazon CLI

运行以下命令，将*`ClusterArn`*和创建集群时所获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

```
aws kafka get-bootstrap-brokers --cluster-arn ClusterArn
```

对于使用的 MSK 集群the section called “IAM 访问控制” (p. 65)，该命令的输出如以下 JSON 示例所示。

```
{
    "BootstrapBrokerStringSaslIam": "b-1.myTestCluster.123z8u.c2.kafka.us-
west-1.amazonaws.com:9098,b-2.myTestCluster.123z8u.c2.kafka.us-west-1.amazonaws.com:9098"
}
```

以下示例显示了已开启公共访问权限的集群的引导代理。使用`BootstrapBrokerStringPublicSaslIam`用于公共访问，以及`BootstrapBrokerStringSaslIam`从中访问的字符串Amazon.

```
{
```

```
    "BootstrapBrokerStringPublicSaslIam": "b-2-public.myTestCluster.v4ni96.c2.kafka-
beta.us-east-1.amazonaws.com:9198,b-1-public.myTestCluster.v4ni96.c2.kafka-
beta.us-east-1.amazonaws.com:9198,b-3-public.myTestCluster.v4ni96.c2.kafka-beta.us-
east-1.amazonaws.com:9198",
    "BootstrapBrokerStringSaslIam": "b-2.myTestCluster.v4ni96.c2.kafka-
beta.us-east-1.amazonaws.com:9098,b-1.myTestCluster.v4ni96.c2.kafka-
beta.us-east-1.amazonaws.com:9098,b-3.myTestCluster.v4ni96.c2.kafka-beta.us-
east-1.amazonaws.com:9098"
}
```

bootstrap brokers 字符串应包含来自部署 MSK 集群的可用区域中的三个代理（除非只有两个代理可用）。

使用 API 获取引导代理

要使用 API 获取引导代理，请参阅GetBootstrapBrokers.

列出 Amazon MSK 集群

使用列出群集Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 该表显示了此账户下当前区域的所有集群。选择要查看其详细信息的集群的名称。

使用列出群集Amazon CLI

运行以下命令。

```
aws kafka list-clusters
```

使用 API 列出集群

要使用 API 列出集群，请参阅ListClusters.

预配置存储吞吐量

Amazon MSK 代理将数据保存在存储卷上。当创建者向集群写入数据、在代理之间复制数据以及使用者读取不在内存中的数据时，会消耗存储 I/O。卷存储吞吐量是指可以向存储卷写入和读取数据的速率。预配置存储吞吐量是指为集群中的代理指定该速率的能力。

您可以为代理类型为 MiB 的集群指定预配置吞吐率（以 MiB /秒为单位）`kafka.m5.4xlarge`或更大，且存储卷大于或等于 10 GiB。可以在集群创建期间指定预配置吞吐量。您还可以为位于中的集群启用或禁用预配置吞吐量`ACTIVE`状态。

吞吐量瓶颈

代理吞吐量瓶颈有多种原因：卷吞吐量、EC2-EBS 网络吞吐量和 EC2 出口吞吐量。您可以启用预配置存储吞吐量来调整卷吞吐量。但是，代理吞吐量限制可能是由 EC2-EBS 网络吞吐量和 EC2 出口吞吐量造成的。

EC2 出口吞吐量受使用者组和每个使用者组的使用者数量的影响。此外，对于较大的代理类型，EC2-EBS 网络吞吐量和 EC2 出口吞吐量都更高，如下表所示。

代理代理引擎	EC2-EBS 网络吞吐量 (Mbps)
kafka.m5.4xlarge	593.75
kafka.m5.8xlarge	850
kafka.m5.12xlarge	1187.5
kafka.m5.16xlarge	1700
kafka.m5.24xlarge	2375

测量存储吞吐量

您可以使用VolumeReadBytes和VolumeWriteBytes指标来衡量集群的平均存储吞吐量。这两个指标的总和得出平均存储吞吐量（以字节为单位）。要获取集群的平均存储吞吐量，请将这两个指标设置为 SUM，将周期设置为 1 分钟，然后使用以下公式。

```
Average storage throughput in MiB/s = (Sum(VolumeReadBytes) + Sum(VolumeWriteBytes)) / (60
 * 1024 * 1024)
```

有关的信息VolumeReadBytes和VolumeWriteBytes指标，请参阅the section called “PER_BROKER级数监控” (p. 99).

配置更新

您可以在启用预配置吞吐量之前或之后更新您的 Amazon MSK 配置。但是，在执行以下两个操作之前，您将看不到所需的吞吐量：更新num.replica.fetchers配置参数并打开预配置吞吐量。

在默认的亚马逊 MSK 配置中，num.replica.fetchers值为 2。更新您的num.replica.fetchers，您可使用下表中的建议值。这些值仅供参考。我们建议您基于自己的使用案例调整这些值。

代理代理引擎	num.replica.fetchers
kafka.m5.4xlarge	4
kafka.m5.8xlarge	8
kafka.m5.12xlarge	14
kafka.m5.16xlarge	16
kafka.m5.24xlarge	16

更新后的配置可能在 24 小时内不会生效，如果源卷未得到充分利用，则可能需要更长的时间。但是，在迁移期间，过渡卷的性能至少等于源存储卷的性能。一个完全利用的 1 TiB 卷通常需要大约 6 小时才能迁移到更新的配置。

使用预配置存储吞吐量Amazon Web Services Management Console

1. 登录到Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home?region=us-east-1#/home/.

2. 选择创建集群。
3. 选择自定义创建.
4. 指定集群的名称。
5. 在存储部分，选择启用.
6. 为每个代理的存储吞吐量选择一个值。
7. 选择 VPC、区域和子网以及安全组。
8. 选择 Next（下一步）。
9. 在的底部安全步骤，选择下一步.
10. 在的底部监控和标记步骤，选择下一步.
11. 审核集群的设置，然后选择创建集群.

使用预配置存储吞吐量Amazon CLI

此部分显示的示例说明如何使用Amazon CLI创建启用了预配置吞吐量的集群。

1. 将下面的 JSON 复制并粘贴到文件中。将子网 ID 和安全组 ID 占位符替换为您账户中的值。将该文件命名为`cluster-creation.json`然后保存。

```
{
    "Provisioned": {
        "BrokerNodeGroupInfo":{
            "InstanceType":"kafka.m5.4xlarge",
            "ClientSubnets":[
                "Subnet-1-ID",
                "Subnet-2-ID"
            ],
            "SecurityGroups":[
                "Security-Group-ID"
            ],
            "StorageInfo": {
                "EbsStorageInfo": {
                    "VolumeSize": 10,
                    "ProvisionedThroughput": {
                        "Enabled": true,
                        "VolumeThroughput": 250
                    }
                }
            }
        },
        "EncryptionInfo": {
            "EncryptionInTransit": {
                "InCluster": false,
                "ClientBroker": "PLAINTEXT"
            }
        },
        "KafkaVersion":"2.2.1",
        "NumberOfBrokerNodes": 2
    },
    "ClusterName": "provisioned-throughput-example"
}
```

2. 运行以下命令Amazon CLI命令从您在上一步中保存 JSON 文件的目录中。

```
aws kafka create-cluster-v2 --cli-input-json file://cluster-creation.json
```

使用 API 配置存储吞吐量

要在创建集群时配置预配置存储吞吐量，请使用CreateClusterV2.

增大代理存储空间

您可以增加每个代理的 EBS 存储空间。您无法减少存储。

在此扩展操作期间，存储卷仍然可用。

> **Important**
>
> 当为 MSK 群集扩展存储时，附加的存储会立即变为可用。但是，集群在每次存储扩展事件后都需要一个冷却期。Amazon MSK 使用此冷却时间对集群进行优化，然后才能再次进行扩展。此时间段从最少 6 小时到 24 小时不等，具体取决于集群的存储大小和利用率以及流量。这适用于auto 伸缩事件和使用UpdateBrokerStorageoperation. 有关调整存储大小的信息，请参阅最佳实践 (p. 123).

主题

自动扩展

要自动扩展集群的存储空间以响应使用量的增加，您可以为 Amazon MSK 配置应用程序自动扩展策略。在 auto-scaling 策略中，您可以设置目标磁盘利用率和最大扩展容量。

在 Amazon MSK 使用自动扩展之前，您应考虑以下几点：

- > **Important**
  >
  > 存储扩展操作每六小时只能执行一次。

  我们建议您从一个大小合适的存储卷开始，以满足您的存储需求。有关调整集群大小的指导，请参阅 将集群设置为正确大小：每个集群的代理数 (p. 123).
- Amazon MSK 不会因使用量减少而减少集群存储。Amazon MSK 不支持减小存储卷的大小。如果需要减小集群存储的大小，则必须将现有集群迁移到存储空间较小的集群。有关迁移集群的信息，请参阅。迁移 (p. 92).
- Amazon MSK 不支持在亚太地区（大阪）和非洲（开普敦）区域推出。
- 如果将auto-scaling 策略与集群关联到集群，Amazon EC2 Auto Scaling 自动创建弹性伸缩策略。CloudWatch 针对目标跟踪的警报。如果您使用auto-scaling 策略删除集群，则 CloudWatch 警报仍然存在。删除 CloudWatch 警报，您应该在删除集群之前从集群中移除auto-scaling 策略。要了解有关目标跟踪的更多信息，请参阅。Amazon EC2 Auto Scaling 的目标跟踪扩缩策略中的Amazon EC2 Auto Scaling 用户指南.

自动扩展策略详情

auto-scaling 策略为您的集群定义集群定义集群定义集群定义集群定义集群定义的集群

- 存储利用率目标：Amazon MSK 使用此阈值触发auto-scaling 操作。您可以将此使用率目标设置为当前存储容量的 10% 到 80% 之间。我们建议您将存储利用率目标设置在 50% 到 60% 之间。

- 最大存储容量：Amazon MSK 可以为您的代理存储设置的最大扩展限制。您可以将每个代理的最大存储容量设置为 16 TiB。有关更多信息，请参阅 Amazon MSK 配额 (p. 108)。

当亚马逊 MSK 检测到您的`Maximum Disk Utilization`指标等于或大于`Storage Utilization Target`设置，它将存储容量增加的量等于两个数字中较大的一个：10 GiB 或当前存储空间的 10%。例如，如果您有 1000 GiB，则该数量为 100 GiB。该服务每分钟检查一次存储利用率。进一步的扩展操作会继续增加存储量，其数量等于两个数字中较大的一个：10 GiB 或当前存储空间的 10%。

要确定是否发生了auto-scaling 操作，请使用 ListClusterOperationsoperation.

为您的 Amazon MSK 集群设置自动扩展

您可以使用亚马逊 MSK 控制台、亚马逊 MSK API 或Amazon CloudFormation以实现存储的自动扩展。CloudFormation 可通过以下方式获得支持Application Auto Scaling.

> Note
>
> 在创建集群时，无法实现自动扩展。您必须先创建集群，然后为其创建并启用auto-scaling 策略。但是，您可以在 Amazon MSK 服务创建集群的同时创建策略。

使用设置自动缩放Amazon Web Services Management Console

1. 登录到。Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home?region=us-east-1#/home/.
2. 请在集群列表中，选择集群。这将带您进入一个列出集群详细信息的页面。
3. 在自动扩展存储部分中，选择。配置.
4. 创建自动扩展策略并命名auto-scaling 策略。指定存储利用率目标、最大存储容量和目标指标。
5. 选择 `Save changes`。

保存并启用新策略后，集群的策略将变为活动状态。然后，当达到存储利用率目标时，Amazon MSK 会扩展集群的存储。

使用 CLI 设置自动扩展

1. 使用 RegisterScalableTarget命令注册存储使用率目标。
2. 使用 PutScalingPolicy命令创建自动扩展策略。

使用 API 设置自动扩展

1. 使用 RegisterScalableTarget用于注册存储利用率目标的 API。
2. 使用 PutScalingPolicy用于创建自动扩展策略的 API。

手动扩展

要增加存储空间，请等待集群进入 `ACTIVE` 状态。存储扩展在两次事件之间至少有六个小时的冷却时间。尽管该操作会立即提供额外的存储空间，但该服务会在您的集群上执行长达 24 小时或更长时间的优化。这些优化的持续时间与您的存储大小成正比。

使用扩展代理存储Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择要更新代理存储的 MSK 集群。

3. 在存储部分，选择编辑.
4. 指定所需存储量。您只能增加存储量，不能减少存储量。
5. 选择Save changes（保存更改）。

使用扩展代理存储Amazon CLI

运行以下命令，将*ClusterArn*和创建集群时所获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

将 *Current-Cluster-Version* 替换为集群的当前版本。

> **Important**
>
> 集群版本不是简单的整数。要查找集群的当前版本，请使用DescribeClusteroperationdescribes Amazon CLI命令。示例版本是 `KTVPDKIKX0DER`。

Target-Volume-in-GiB 参数表示您希望每个代理具备的存储量。只能更新所有代理的存储。您不能指定要更新存储的单个代理。您为 *Target-Volume-in-GiB* 指定的值必须是大于 100 GiB 的整数。更新操作后每个代理的存储不能超过 16384 GiB。

```
aws kafka update-broker-storage --cluster-arn ClusterArn --current-version Current-
Cluster-Version --target-broker-ebs-volume-info '{"KafkaBrokerNodeId": "All",
 "VolumeSizeGB": Target-Volume-in-GiB}'
```

使用 API 扩展代理存储

要使用 API 更新代理存储，请参阅UpdateBrokerStorage.

更新代理类型

您可以通过更改代理的类型（大小或系列）按需扩展 MSK 集群，而无需重新分配 Apache Kafka 分区。通过更改代理的类型，您可以根据工作负载的变化灵活调整 MSK 集群的计算容量，而不会中断集群 I/O。Amazon MSK 对给定集群中的所有代理使用相同的代理类型。本节介绍如何更新 MSK 集群的代理类型。当集群启动并运行时，代理类型更新会以滚动方式进行。这意味着Amazon MSK一次关闭一个经纪商来执行经纪人类型的更新。有关如何在代理类型更新期间使集群高度可用的信息，请参阅the section called “构建高度可用的集群” (p. 123). 为了进一步减少对工作效率的任何潜在影响，您可以在流量较低的时期执行经纪人类型的更新。

在代理类型更新期间，您可以继续生成和使用数据。但是，您必须等到更新完成后才能重新启动 broker 或调用下面列出的任何更新操作Amazon MSK 运营.

如果您想将集群更新为较小的代理类型，我们建议您先在测试集群上尝试更新，以了解它如何影响您的场景。

> **Important**
>
> 如果每个代理的分区数超过中指定的最大数量，则无法将集群更新为较小的代理类型the section called “ 将集群设置为正确大小：每个代理的分区数” (p. 123).

使用更新代理类型Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.

2. 选择要更新代理类型的 MSK 集群。
3. 在集群的详细信息页面上，找到代理摘要部分，然后选择编辑代理代理代理代理引擎.
4. 从列表中选择所需的代理类型。
5. 保存更改。

使用更新代理类型Amazon CLI

1. 运行以下命令，将*ClusterArn*和创建集群时所获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

   Replace******集群的当前版本和*TargetType*使用您希望代理采用的新类型。要了解有关代理类型的更多信息，请参阅the section called “代理代理代理引擎” (p. 9).

   ```
   aws kafka update-broker-type --cluster-arn ClusterArn --current-version Current-
   Cluster-Version --target-instance-type TargetType
   ```

   下面的示例说明如何使用此命令：

   ```
   aws kafka update-broker-type --cluster-arn "arn:aws:kafka:us-
   east-1:0123456789012:cluster/exampleName/abcd1234-0123-abcd-5678-1234abcd-1" --current-
   version "K1X5R6FKA87" --target-instance-type kafka.m5.large
   ```

   该 命令的输出如以下 JSON 示例所示。

   ```
   {
       "ClusterArn": "arn:aws:kafka:us-east-1:0123456789012:cluster/exampleName/
   abcd1234-0123-abcd-5678-1234abcd-1",
       "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
   operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
   abcd-4f7f-1234-9876543210ef"
   }
   ```

2. 要得到update-broker-type操作时，运行以下命令，将*ClusterOperationArn*使用您在输出中获得的 ARNupdate-broker-type命令。

   ```
   aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
   ```

   该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

   ```
   {
     "ClusterOperationInfo": {
       "ClientRequestId": "982168a3-939f-11e9-8a62-538df00285db",
       "ClusterArn": "arn:aws:kafka:us-east-1:0123456789012:cluster/exampleName/
   abcd1234-0123-abcd-5678-1234abcd-1",
       "CreationTime": "2021-01-09T02:24:22.198000+00:00",
       "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-operation/
   exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
   abcd-4f7f-1234-9876543210ef",
       "OperationState": "UPDATE_COMPLETE",
       "OperationType": "UPDATE_BROKER_TYPE",
       "SourceClusterInfo": {
         "InstanceType": "t3.small"
       },
       "TargetClusterInfo": {
         "InstanceType": "m5.large"
   ```

```
    }
  }
}
```

如果 OperationState 的值为 UPDATE_IN_PROGRESS，请等待一段时间，然后再次运行 describe-cluster-operation 命令。

使用 API 更新代理类型

要使用 API 更新代理类型，请参阅UpdateBrokerType.

更新 Amazon MSK 集群的配置

要更新集群配置，请确保集群处于 ACTIVE 状态。您还必须确保 MSK 集群上每个代理的分区数量低于中所述的限制。the section called “ 将集群设置为正确大小：每个代理的分区数” (p. 123). 您无法更新超过这些限制的群集的配置。

有关 MSK 配置的信息，包括如何创建自定义配置、可以更新哪些属性以及更新现有集群的配置时会发生什么情况，请参阅 配置 (p. 31)。

使用更新集群的配置Amazon CLI

1. 复制以下 JSON 并将其保存到文件中。将文件命名为 configuration-info.json。Replace*ConfigurationArn*和要用于更新集群的 Amazon 资源名称 (ARN)。在以下 JSON 中，ARN 字符串必须使用引号引起来。

   将 *Configuration-Revision* 替换为要使用的配置的修订版本。配置修订版本是从 1 开始的整数。在以下 JSON 中，该整数不能使用引号引起来。

   ```
   {
       "Arn": ConfigurationArn,
       "Revision": Configuration-Revision
   }
   ```

2. 运行以下命令，将*ClusterArn*使用您在创建集群时获取的 ARN。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

   将 *Path-to-Config-Info-File* 替换为您的配置信息文件的路径。如果您将上一步中创建的文件命名为 configuration-info.json，并将其保存在当前目录中，*Path-to-Config-Info-File* 即为 configuration-info.json。

   将 *Current-Cluster-Version* 替换为集群的当前版本。

   > **Important**
   >
   > 集群版本不是简单的整数。要查找集群的当前版本，请使用DescribeClusteroperationdescribes Amazon CLI命令。示例版本是 KTVPDKIKX0DER。

   ```
   aws kafka update-cluster-configuration --cluster-arn ClusterArn --configuration-info
    file://Path-to-Config-Info-File --current-version Current-Cluster-Version
   ```

   下面的示例说明如何使用此命令：

```
aws kafka update-cluster-configuration --cluster-arn "arn:aws:kafka:us-
east-1:0123456789012:cluster/exampleName/abcd1234-0123-abcd-5678-1234abcd-1" --
configuration-info file://c:\users\tester\msk\configuration-info.json --current-version
 "K1X5R6FKA87"
```

该 update-cluster-configuration 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
    "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef"
}
```

3. 要得到update-cluster-configuration操作时，运行以下命令，将*ClusterOperationArn*使用您在输出中获得的 ARNupdate-cluster-configuration命令。

```
aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
```

该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterOperationInfo": {
        "ClientRequestId": "982168a3-939f-11e9-8a62-538df00285db",
        "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
        "CreationTime": "2019-06-20T21:08:57.735Z",
        "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef",
        "OperationState": "UPDATE_COMPLETE",
        "OperationType": "UPDATE_CLUSTER_CONFIGURATION",
        "SourceClusterInfo": {},
        "TargetClusterInfo": {
            "ConfigurationInfo": {
                "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/
ExampleConfigurationName/abcdabcd-abcd-1234-abcd-abcd123e8e8e-1",
                "Revision": 1
            }
        }
    }
}
```

在此输出中， OperationType 是 UPDATE_CLUSTER_CONFIGURATION。如果 OperationState 的值为 UPDATE_IN_PROGRESS，请等待一段时间，然后再次运行 describe-cluster-operation 命令。

使用 API 更新集群的配置

要使用 API 更新集群的配置，请参阅UpdateClusterConfiguration.

扩展 Amazon MSK 集群

如果要增加 MSK 集群中的代理数量，请使用此 Amazon MSK 操作。要扩展集群，请确保集群处于 ACTIVE 状态。

Important

如果要扩展 MSK 集群，请确保使用此 Amazon MSK 操作。切勿尝试在未使用此操作的情况下向集群添加代理。

有关在将代理添加到集群后如何重新平衡分区的信息，请参阅the section called “重新分配分区” (p. 126)。

使用扩展集群Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择要为其增加代理数量的 MSK 集群。
3. 在集群详细信息页面上，选择编辑按钮旁边的集群级代理详细信息标题。
4. 输入您希望集群在每个可用区具有的代理数量，然后选择保存更改.

使用扩展集群Amazon CLI

1. 运行以下命令，将*ClusterArn*和创建集群时所获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

   将 *Current-Cluster-Version* 替换为集群的当前版本。

   Important

   集群版本不是简单的整数。要查找集群的当前版本，请使用DescribeClusteroperationdescribes Amazon CLI命令。示例版本是 KTVPDKIKX0DER。

   Target-Number-of-Brokers 参数表示在此操作成功完成时您希望集群具有的代理节点的总数。您为 *Target-Number-of-Brokers* 指定的值必须是大于集群中当前代理数量的整数。它还必须是可用区数目的倍数。

   ```
   aws kafka update-broker-count --cluster-arn ClusterArn --current-version Current-
   Cluster-Version --target-number-of-broker-nodes Target-Number-of-Brokers
   ```

   该 update-broker-count 操作的输出如以下 JSON 所示：

   ```
   {

       "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
   abcdefab-1234-abcd-5678-cdef0123ab01-2",
       "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
   operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
   abcd-4f7f-1234-9876543210ef"
   }
   ```

2. 要得到update-broker-count操作时，运行以下命令，将*ClusterOperationArn*使用您在输出中获得的 ARNupdate-broker-count命令。

```
aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
```

该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterOperationInfo": {
        "ClientRequestId": "c0b7af47-8591-45b5-9c0c-909a1a2c99ea",
        "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
        "CreationTime": "2019-09-25T23:48:04.794Z",
        "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef",
        "OperationState": "UPDATE_COMPLETE",
        "OperationType": "INCREASE_BROKER_COUNT",
        "SourceClusterInfo": {
            "NumberOfBrokerNodes": 9
        },
        "TargetClusterInfo": {
            "NumberOfBrokerNodes": 12
        }
    }
}
```

在此输出中，OperationType 是 INCREASE_BROKER_COUNT。如果 OperationState 的值为 UPDATE_IN_PROGRESS，请等待一段时间，然后再次运行 describe-cluster-operation 命令。

使用 API 扩展集群

要使用 API 增加集群中的代理数量，请参阅UpdateBrokerCount.

更新集群的安全设置

使用此 Amazon MSK 操作更新您的 MSK 集群的身份验证和客户端代理加密设置。您还可以更新用于签署双向 TLS 身份验证的证书的私有安全机构。您不能更改集群中 (broker-to-broker) 加密设置。

集群必须位于ACTIVE状态，以便您更新其安全设置。

如果您使用 IAM、SASL 或 TLS 启用身份验证，则还必须启用客户端和代理之间的加密。下表显示了可能的组合。

身份验证	客户端代理加密选项	经纪商加密
未经身份验证的	TLS、明文、TLS_PLAINTEXT	可以打开或关闭。
MTL	TLS，TLS_PLAINTEXT	必须打开。
SASL/SCRAM	TLS	必须打开。
SASLUSTERS	TLS	必须打开。

当客户端代理加密设置为TLS_PLAINTEXT并且客户端身份验证设置为mTLS中，Amazon MSK 创建了两种类型的侦听器供客户端连接：一种监听器供客户端使用 MTLS 身份验证和 TLS 加密进行连接，另一种监听器供客户端在不进行身份验证或加密的情况下进行连接（明文）。

有关安全设置的更多信息，请参阅安全性 (p. 50).

使用更新群集的安全设置Amazon Web Services Management Console

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择要更新的 MSK 集群。
3. 在安全设置部分，选择编辑.
4. 为群集选择所需的身份验证和加密设置，然后选择保存更改.

使用更新群集的安全设置Amazon CLI

1. 创建包含您希望集群采用的加密设置的 JSON 文件。以下是示例。

   Note

   您只能更新客户端代理加密设置。您无法更新集群中 (broker-to-broker) 加密设置。

   ```
   {"EncryptionInTransit":{"ClientBroker": "TLS"}}
   ```

2. 创建包含您希望集群采用的身份验证设置的 JSON 文件。以下是示例。

   ```
   {"Sasl":{"Scram":{"Enabled":true}}}
   ```

3. 运行以下 Amazon CLI 命令：

   ```
   aws kafka update-security --cluster-arn ClusterArn --current-version Current-Cluster-
   Version --client-authentication file://Path-to-Authentication-Settings-JSON-File --
   encryption-info file://Path-to-Encryption-Settings-JSON-File
   ```

   该 update-security 操作的输出如以下 JSON 所示：

   ```
   {
   
       "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
   abcdefab-1234-abcd-5678-cdef0123ab01-2",
       "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
   operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
   abcd-4f7f-1234-9876543210ef"
   }
   ```

4. 查看的状态update-security操作时，运行以下命令，将*ClusterOperationArn*使用您在输出中获得的 ARNupdate-security命令。

   ```
   aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
   ```

   该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

   ```
   {
       "ClusterOperationInfo": {
           "ClientRequestId": "c0b7af47-8591-45b5-9c0c-909a1a2c99ea",
           "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
   abcdefab-1234-abcd-5678-cdef0123ab01-2",
           "CreationTime": "2021-09-17T02:35:47.753000+00:00",
   ```

```
        "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef",
        "OperationState": "PENDING",
        "OperationType": "UPDATE_SECURITY",
        "SourceClusterInfo": {},
        "TargetClusterInfo": {}
    }
}
```

如果`OperationState`有价值`PENDING`要么`UPDATE_IN_PROGRESS`，请等待一段时间，然后运行`describe-cluster-operation`再次命令。

使用 API 更新集群的安全设置

要使用 API 更新集群的安全设置，请参阅UpdateSecurity.

Note

这些区域有：Amazon CLI和用于更新集群安全设置的 API 操作是幂等的。这意味着，如果您调用安全更新操作并指定与集群当前设置相同的身份验证或加密设置，则该设置不会更改。

重启 Amazon MSK 集群代理

当您想为 MSK 集群重新启动经纪商时，请使用此 Amazon MSK 操作。要重启集群代理，请确保集群`ACTIVE`状态。

Amazon MSK 服务可能会在系统维护（例如修补或版本升级）期间为 MSK 集群重新启动代理商。手动重启经纪人可以让你测试 Kafka 客户端的弹性，以确定他们如何应对系统维护。

使用重启代理Amazon Web Services Management Console

1. 在打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择要重启代理的 MSK 集群。
3. 向下滚动到代理详情部分，然后选择要重启的代理。
4. 选择重启代理按钮。

使用重启代理Amazon CLI

1. 运行以下命令，替换以下命令*`ClusterArn`*使用您在创建集群时获取的 Amazon 资源名称 (ARN)，*`BrokerId`*使用您要重启的代理的 ID。

   Note

   这些区域有：`reboot-broker`操作一次只支持重启一个代理。

   如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

   如果您没有集群的代理 ID，可以通过列出代理节点来找到它们。有关更多信息，请参阅 。列表节点.

```
aws kafka reboot-broker --cluster-arn ClusterArn --broker-ids BrokerId
```

该 reboot-broker 操作的输出如以下 JSON 所示：

```
{

    "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
    "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef"
}
```

2. 要获得结果reboot-broker操作，运行以下命令，替换*ClusterOperationArn*使用你在输出中获得的 ARNreboot-broker命令。

```
aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
```

该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterOperationInfo": {
        "ClientRequestId": "c0b7af47-8591-45b5-9c0c-909a1a2c99ea",
        "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
        "CreationTime": "2019-09-25T23:48:04.794Z",
        "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef",
        "OperationState": "REBOOT_IN_PROGRESS",
        "OperationType": "REBOOT_NODE",
        "SourceClusterInfo": {},
        "TargetClusterInfo": {}
    }
}
```

在重启操作完成后，OperationState是REBOOT_COMPLETE.

使用 API 重启代理

要使用 API 重启集群中的代理，请参阅RebootBroker.

为 Amazon MSK 集群添加标签

您可以以下形式分配自己的元数据标签迁移到 Amazon MSK 资源，例如 MSK 群集。标签是您为资源定义的键值对。使用标签是管理 Amazon 资源和组织数据（包括账单数据）的一种简单却强有力的方式。

主题

有关标签的基本知识

您可以使用Amazon MSK API 完成以下任务：

- 向 Amazon MSK 资源添加标签。
- 列出 Amazon MSK 资源的标签。
- 从 Amazon MSK 资源中删除标签。

您可以使用标签对 Amazon MSK 资源进行分类。例如，您可以按用途、拥有者或环境对 Amazon MSK 集群进行分类。由于您定义每个标签的键和值，因此您可以创建一组自定义类别来满足您的特定需求。例如，您可以定义一组标签来帮助您按所有者和关联应用程序跟踪集群。

以下是标签的多个示例：

- `Project: Project name`
- `Owner: Name`
- `Purpose: Load testing`
- `Environment: Production`

使用标签跟踪成本

您可以使用标签对 Amazon 成本进行分类和跟踪。当您将标签应用于Amazon资源，包括 Amazon MSK 集群，Amazon成本分配报告包括按标签汇总的使用率和成本。您可通过应用代表业务类别（如成本中心、应用程序名称或拥有者）的标签来整理多种服务的成本。有关更多信息，请参阅 Amazon Billing 用户指南中的对自定义账单报告使用成本分配标签。

标签限制

以下限制适用于Amazon MSK 中的标签。

基本限制

- 每个资源的最大标签数是 50。
- 标签键和值区分大小写。
- 无法更改或编辑已删除的资源的标签。

标签键限制

- 每个标签键必须是唯一的。如果您添加的标签具有已使用的键，则您的新标签将覆盖现有键值对。
- 标签键不能以 `aws:` 开头，因为此前缀将预留以供 Amazon 使用。Amazon 将代表您创建以此前缀开头的标签，但您不能编辑或删除这些标签。
- 标签键的长度必须介于 1 和 128 个 Unicode 字符之间。
- 标签密钥必须包含以下字符：Unicode 字母、数字、空格和以下特殊字符：`_ . / = + - @`.

标签值限制

- 标签值的长度必须介于 0 和 255 个 Unicode 字符之间。
- 标签值可以为空。否则，它们必须包含以下字符：Unicode 字母、数字、空格和以下任意特殊字符：`_ . / = + - @`.

使用亚马逊 MSK API 标记资源

您可以使用以下操作来标记或取消标记Amazon MSK 资源，或者列出资源的当前标记集：

- ListTagsForResource
- TagResource
- UntagResource

Amazon MSK 配置

Amazon MSK 为代理、主题和 Apache 提供默认配置 ZooKeeper节点。您还可以创建自定义配置，并使用这些配置来创建新的 MSK 集群或更新现有集群。MSK 配置由一组属性及其相应的值构成。

主题

自定义 MSK 配置

Amazon MSK 允许您创建自定义 MSK 配置，您可以在其中设置以下属性。未显式设置的属性将获得其在 the section called “默认配置” (p. 35) 中具有的值。有关配置属性的更多信息，请参阅 Apache Kafka 配置。

你可以设置的 Apache Kafka 配置属性

名称	描述
allow.everyone.if.no.acl.found	如果您想将此属性设置为 false，请首先确保为集群定义 Apache Kafka ACL。如果您在未事先定义 Apache Kafka ACL 的情况下将此属性设置为 false，则您将失去对集群的访问权限。如果发生这种情况，您可以再次更新配置并将此属性设置为 true 以重新获得对集群的访问权限。
auto.create.topics.enable	在服务器上启用主题自动创建。
compression.type	给定主题的最终压缩类型。可以将此属性设置为标准压缩编解码器（`gzip`、`snappy`、`lz4` 和 `zstd`）。此外，它还接受 `uncompressed` 和 `producer`，前者相当于没有压缩，后者意味着保留由创建器设置的原始压缩编解码器。
connections.max.idle.ms	空闲连接超时（以毫秒为单位）服务器套接字处理器线程关闭空闲时间超过您为此属性设置的值的连接。
default.replication.factor	自动创建的主题的默认复制因子。
delete.topic.enable	启用删除主题操作。如果此配置已被禁用，则无法通过管理工具删除主题。
group.initial.rebalance.delay.ms	在执行第一次重新平衡之前，组协调器等待更多使用器加入新组的时间。更长的延迟时间意味着重新平衡可能会更少，但会增加处理开始之前的时间。
group.max.session.timeout.ms	注册使用器的最长会话超时时间。超时时间越长，可供使用器用来处理检测信号之间的消息的时间就越多，但这会导致需要花费更多时间来检测故障。
group.min.session.timeout.ms	注册使用器的最短会话超时时间。超时时间越短，故障检测的速度就会越快，但需要更频繁的使用器检测信号，这会耗尽代理资源。

名称	描述
leader.imbalance.per.broker.percentage	各代理允许的领导节点不平衡比率。如果各代理上超过了此值，则控制器将触发领导节点平衡操作。此值以百分比的形式指定。
log.cleaner.delete.retention.ms	您希望 Apache Kafka 保留已删除的记录的时间量。最小值为 0。
log.cleaner.min.cleanable.ratio	此配置属性的值可以在 0 和 1 之间。它决定了日志压缩器尝试清理日志的频率（假设启用了日志压缩）。默认情况下，Apache Kafka 会避免清理已压缩 50% 以上的日志的日志。此比率限制了重复项在日志中浪费的最大空间（为 50%，这意味着最多有 50% 的日志可能是重复的）。更高的比率意味着更少、更高效的清洁，但会浪费更多的日志空间。
log.cleanup.policy	超出保留时段的分段的默认清除策略。有效策略的逗号分隔列表。有效策略为 `delete` 和 `compact`。
log.flush.interval.messages	将消息刷新到磁盘之前，日志分区上累积的消息的数量。
log.flush.interval.ms	任何主题中的消息在刷新到磁盘之前保存在内存中的最长时间（以毫秒为单位）。如果未设置，则使用 log.flush.scheduler.interval.ms 中的值。最小值为 0。
log.message.timestamp.diffence.max.ms	代理收到消息时的时间戳与消息中指定的时间戳之间允许的最大差异。如果 log.message.timestamp.type=CreateTime，如果时间戳的差值超过此阈值，则消息将被拒绝。如果 log.message.timestamp.type= 则忽略此配置 LogAppendTime.
log.message.timestamp.typ	指定消息中的时间戳是消息创建时间还是日志追加时间。允许的值是 `CreateTime` 和 `LogAppendTime`。
log.retention.bytes	删除日志前的最大日志大小。
log.retention.hours	删除日志文件前保留日志文件的小时数，它是 log.retention.ms 属性的三级属性。
log.retention.minutes	删除日志文件前保留日志文件的分钟数，它是 log.retention.ms 属性的二级属性。如果未设置，则使用 log.retention.hours 中的值。
log.retention.ms	删除日志文件前保留日志文件的毫秒数（以毫秒为单位）。如果未设置，则使用 log.retention.minutes 中的值。
log.roll.ms	推出新日志段之前的最长时间（以毫秒为单位）。如果您未设置此属性，则使用 log.roll.hours 中的值。此属性的最小可能值为 1。
log.segment.bytes	单个日志文件的最大大小。
max.incremental.fetch.session.cache.slots	维护的增量提取会话的最大数量。

名称	描述
message.max.bytes	Kafka 允许的最大记录批处理大小。如果增加此数量，并且存在大于 0.10.2 的使用器，则使用器的提取大小也必须增加，以便它们能够提取如此大的记录批处理。 在最新的消息格式版本中，总是将记录分组到批处理中来提高效率。在以前的消息格式版本中，未压缩的记录不会分组到批处理中，在此情况下，此限制仅适用于单条记录。 可使用主题级别 max.message.bytes 配置为每个主题进行此设置。
min.insync.replicas	当创建器将 acks 设置为 "all"（或 "-1"）时，min.insync.replicas 指定为使写入被视为成功而必须确认写入的最小副本数。如果无法达到这个最小值，则生产者会引发异常（ NotEnoughReplicas 要么 NotEnoughReplicasAfterAppend)。 通过将 min.insync.replicas 和 acks 结合使用，您可以增强耐用性保证。一个典型的场景是，创建复制因子为 3 的主题，将 min.insync.replicas 设置为 2，并在 acks 为 "all" 的情况下进行创建。这可确保在大多数副本未收到写操作时，创建器将引发异常。
num.io.threads	服务器用于处理请求的线程的数目，其中可能包括磁盘 I/O。
num.network.threads	服务器用于接收来自网络的请求并向其发送响应的线程的数目。
num.partitions	每个主题的默认日志分区数。
num.recovery.threads.per.data.dir	在启动时用于日志恢复以及在关闭时用于刷新的每个数据目录的线程数。
num.replica.fetchers	用于从源代理复制消息的提取器线程数。增大此值会增加跟踪器代理中的 I/O 并行度。
offsets.retention.minutes	当一个使用器组丢失其所有使用器（即变空）后，其偏移量将在此保留期内保留，然后被丢弃。对于独立使用器（即，使用手动分配），偏移量会在最后一次提交时间加上此保留期后过期。
offsets.topic.replication.factor	偏移量主题的复制因子（设置为较高的值可确保可用性）。内部主题创建失败，直到集群大小满足此复制因子要求。
replica.fetch.max.bytes	尝试为每个分区提取的消息的字节数。这不是绝对最大值。如果提取的第一个非空分区中的第一个记录批处理大于此值，则将返回该记录批处理以确保可以取得进展。代理接受的最大记录批处理大小通过 message.max.bytes（代理配置）或 max.message.bytes（主题配置）进行定义。

名称	描述
replica.fetch.response.max.bytes	整个提取响应预期的最大字节数。记录是分批提取的，如果提取的第一个非空分区中的第一个记录批处理大于此值，则将返回该记录批处理以确保可以取得进展。这不是绝对最大值。message.max.bytes（代理配置）或 max.message.bytes（主题配置）属性指定代理接受的最大记录批处理大小。
replica.lag.time.max.ms	如果跟踪器没有发送任何提取请求，或者至少在此毫秒数内没有使用到领导的日志结束偏移量，则领导会从 ISR 中删除追随者。 MinValue：10000 MaxValue（含）= 30000
replica.selector.class	实现的完全限定类名 ReplicaSelector. 代理使用此名称来查找首选读取副本。如果您使用的是 Apache Kafka 版本 2.4.1 或更高版本，并且希望允许使用器从最近的副本提取，请将此属性设置为 `org.apache.kafka.common.replica.RackAwareReplicaS` 有关更多信息，请参阅 the section called “Apache Kafka 版本 2.4.1（改为使用 2.4.1）” (p. 113)。
replica.socket.receive.buffer.bytes	网络请求的套接字接收缓冲区。
socket.receive.buffer.bytes	套接字服务器套接字的 SO_RCVBUF 缓冲区。可以将此属性设置为的最小值为 -1。如果值为 -1，则亚马逊 MSK 使用操作系统的默认值。
socket.request.max.bytes	套接字请求中的最大字节数。
socket.send.buffer.bytes	套接字服务器套接字的 SO_SNDBUF 缓冲区。可以将此属性设置为的最小值为 -1。如果值为 -1，则亚马逊 MSK 使用操作系统的默认值。
transaction.max.timeout.ms	事务的最大超时时间。如果客户请求的交易时间超过此值，则经纪人会返回错误 InitProducerIdRequest. 这可防止客户端的超时时间过长，此情况可能会导致使用器无法阅读事务中包含的主题。
transaction.state.log.min.isr	覆盖事务主题的 min.insync.replicas 配置。
transaction.state.log.replication.factor	事务主题的复制因子。将它设置为较高的值可提高可用性。内部主题创建失败，直到集群大小满足此复制因子要求。
transactional.id.expiration.ms	事务协调器在交易 ID 到期之前没有收到当前事务的任何事务状态更新的等待时间（以毫秒为单位）。此设置还会影响生产者 ID 的到期：如果在最后一次使用给定的生产者 ID 写入之后经过这段时间，生产者 ID 即过期。如果由于主题的保留设置而删除了生产者 ID 的最后一次写入，则生产者 ID 可能会更快过期。此属性的最小值为 1 毫秒。

名称	描述
unclean.leader.election.enable	指示是否允许选择不在 ISR 集中的副本作为领导（作为最后的手段），即使这样做可能导致数据丢失。
zookeeper.connection.timeout.ms	客户端等待与之建立连接的最长时间 ZooKeeper. 如果未设置，则使用 zookeeper.session.timeout.ms 中的值。
zookeeper.session.timeout.ms	Apache ZooKeeper 会话超时（以毫秒为单位） MinValue = 6000 MaxValue （含）= 18000

要了解如何创建自定义 MSK 配置、列出所有配置或描述它们，请参阅 the section called “配置操作” (p. 37)。要使用自定义 MSK 配置创建 MSK 集群或使用新的自定义配置更新集群，请参阅 工作方式 (p. 9)。

当您使用自定义 MSK 配置更新现有 MSK 集群时，Amazon MSK 会在必要时进行滚动重启，使用最佳实践最大限度地减少客户停机时间。例如，在 Amazon MSK 重启每个代理之后，它会尝试让代理在转移到下一个代理之前catch 上代理在配置更新期间可能丢失的数据。

动态配置

除了 Amazon MSK 提供的配置属性外，您还可以动态设置不需要重启代理的集群和代理级配置属性。您可以动态设置在 Apache Kafka 文档中的代理配置下的表中未标记为只读的配置属性。有关动态配置和示例命令的信息，请参阅 Apache Kafka 文档中的更新代理配置。

Note

您可以设置 `advertised.listeners` 属性，但不能设置 `listeners` 属性。

主题级配置

您可以使用 Apache Kafka 命令为新主题和现有主题设置或修改主题级别的配置属性。有关主题级别的配置属性以及如何使用这些属性的示例的更多信息，请参阅 Apache Kafka 文档中的主题级别的配置。

配置状态

亚马逊 MSK 配置可能处于以下状态。要对配置执行操作，该配置必须位于`ACTIVE`要么`DELETE_FAILED`状态：

- `ACTIVE`
- `DELETING`
- `DELETE_FAILED`

Amazon MSK 默认配置

当您在未指定自定义 MSK 配置的情况下创建 MSK 集群时，Amazon MSK 会创建并使用默认配置，其值如下表所示。对于不在此表中的属性，亚马逊 MSK 使用与您的 Apache Kafka 版本相关的默认值。有关这些默认值的列表，请参阅 Apache Kafka 配置。

默认配置值

名称	描述	默认值
allow.everyone.if.no.acl.found	如果没有与特定资源匹配的资源模式，则该资源没有关联的 ACL。在此情况下，如果将此属性设置为 true，则将允许所有人（而不仅仅是超级用户）访问该资源。	`true`
auto.create.topics.enable	在服务器上启用主题的自动创建。	`false`
auto.leader.rebalance.enable	启用自动领导平衡。如果需要，后台线程会定期检查并触发领导平衡。	`true`
default.replication.factor	自动创建的主题的默认复制因子。	3 个用于 3 AZ 群集，2 个用于 2-AZ 群集
min.insync.replicas	当创建器将 acks 设置为 `"all"`（或 `"-1"`）时，min.insync.replicas 指定为使写入被视为成功而必须确认写入的最小副本数。如果无法达到这个最小值，则生产者会引发异常（ NotEnoughReplicas要么 NotEnoughReplicasAfterAppend)。 通过将 min.insync.replicas 和 acks 结合使用，您可以增强耐用性保证。一个典型的场景是，创建复制因子为 3 的主题，将 min.insync.replicas 设置为 2，并在 acks 为 `"all"` 的情况下进行创建。这可确保在大多数副本未收到写操作时，创建器将引发异常。	2 个用于 3 AZ 群集，1 个用于 2-AZ 群集
num.io.threads	服务器用于处理请求的线程的数目，其中可能包括磁盘 I/O。	8
num.network.threads	服务器用于接收来自网络的请求并向网络发送响应的线程的数目。	5
num.partitions	每个主题的默认日志分区数。	1
num.replica.fetchers	用于从源代理复制消息的提取器线程数。增大此值会增加跟踪器代理中的 I/O 并行度。	2
replica.lag.time.max.ms	如果跟踪器没有发送任何提取请求，或者至少在此毫秒数内没有使用到领导的日志结束偏移量，则领导会从 ISR 中删除追随者。	30000

名称	描述	默认值
socket.receive.buffer.bytes	套接字服务器套接字的 SO_RCVBUF 缓冲区。如果值为 -1，则使用操作系统默认值。	102400
socket.request.max.bytes	套接字请求中的最大字节数。	104857600
socket.send.buffer.bytes	套接字服务器套接字的 SO_SNDBUF 缓冲区。如果值为 -1，则使用操作系统默认值。	102400
unclean.leader.election.enable	指示是否允许选择不在 ISR 集中的副本作为领导（作为最后的手段），即使这样做可能导致数据丢失。	`true`
zookeeper.session.timeout.ms	Apache ZooKeeper 会话超时（以毫秒为单位）	18000
zookeeper.set.acl	将客户端设置为使用安全 ACL。	`false`

有关如何指定自定义配置值的信息，请参阅the section called “自定义 配置” (p. 31).

Amazon MSK 配置操作

本主题说明如何创建自定义 MSK 配置以及如何对这些配置执行操作。有关如何使用 MSK 配置创建或更新集群的信息，请参阅 工作方式 (p. 9)。

本主题包含下列部分：

创建 MSK 配置

1. 创建一个文件，可在其中指定要设置的配置属性以及要分配给这些属性的值。以下是示例配置文件的内容。

```
auto.create.topics.enable = true

zookeeper.connection.timeout.ms = 1000

log.roll.ms = 604800000
```

2. 运行以下命令Amazon CLI命令，替换*config-file-path*使用您在上一步中保存配置的文件的路径。

   Note

   您为配置选择的名称必须与以下正则表达式相匹配：“A----------

```
aws kafka create-configuration --name "ExampleConfigurationName" --description
 "Example configuration description." --kafka-versions "1.1.1" --server-properties
 fileb://config-file-path
```

以下是运行此命令后的成功响应示例。

```
{
    "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/abcdabcd-1234-
abcd-1234-abcd123e8e8e-1",
    "CreationTime": "2019-05-21T19:37:40.626Z",
    "LatestRevision": {
        "CreationTime": "2019-05-21T19:37:40.626Z",
        "Description": "Example configuration description.",
        "Revision": 1
    },
    "Name": "ExampleConfigurationName"
}
```

3. 上一条命令将返回新创建的配置的 Amazon 资源名称 (ARN)。保存此 ARN，因为您需要使用它来在其他命令中引用此配置。如果您丢失了配置 ARN，则可通过列出您账户中的所有配置来重新找到它。

更新 MSK 配置

1. 创建一个文件，在其中指定要更新的配置属性和要分配给它们的值。以下是示例配置文件的内容。

```
auto.create.topics.enable = true

zookeeper.connection.timeout.ms = 1000

min.insync.replicas = 2
```

2. 运行以下命令Amazon CLI命令，替换*config-file-path*使用您在上一步中保存配置的文件的路径。

   Replace（替换）*## arn*使用您在创建配置时获得的 ARN。如果您在创建配置时未保存 ARN，则可使用 `list-configurations` 命令列出您账户中的所有配置，并在响应中显示的列表中找到所需的配置。配置的 ARN 也将显示在该列表中。

```
aws kafka update-configuration --arn configuration-arn --description "Example
 configuration revision description." --server-properties fileb://config-file-path
```

3. 以下是运行此命令后的成功响应示例。

```
{
    "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/abcdabcd-1234-
abcd-1234-abcd123e8e8e-1",
    "LatestRevision": {
        "CreationTime": "2020-08-27T19:37:40.626Z",
        "Description": "Example configuration revision description.",
        "Revision": 2
    }
}
```

删除 MSK 配置

以下过程介绍如何删除未附加到集群的配置。您无法删除连接到集群的配置。

1. 要运行此示例，请将 *configuration-arn* 替换为您在创建配置时获取的 ARN。如果您在创建配置时未保存 ARN，则可使用 `list-configurations` 命令列出您账户中的所有配置，并在响应中显示的列表中找到所需的配置。配置的 ARN 也将显示在该列表中。

```
aws kafka delete-configuration --arn configuration-arn
```

2. 以下是运行此命令后的成功响应示例。

```
{
    "arn": " arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/abcdabcd-1234-abcd-1234-abcd123e8e8e-1",
    "state": "DELETING"
}
```

描述 MSK 配置

1. 此命令将返回有关配置的元数据。要获取配置的详细说明，请运行 `describe-configuration-revision`。

   要运行此示例，请将 *configuration-arn* 替换为您在创建配置时获取的 ARN。如果您在创建配置时未保存 ARN，则可使用 `list-configurations` 命令列出您账户中的所有配置，并在响应中显示的列表中找到所需的配置。配置的 ARN 也将显示在该列表中。

```
aws kafka describe-configuration --arn configuration-arn
```

2. 以下是运行此命令后的成功响应示例。

```
{
    "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/abcdabcd-abcd-1234-abcd-abcd123e8e8e-1",
    "CreationTime": "2019-05-21T00:54:23.591Z",
    "Description": "Example configuration description.",
    "KafkaVersions": [
        "1.1.1"
    ],
    "LatestRevision": {
        "CreationTime": "2019-05-21T00:54:23.591Z",
        "Description": "Example configuration description.",
        "Revision": 1
    },
    "Name": "SomeTest"
}
```

描述 MSK 配置修订

通过使用 `describe-configuration` 命令描述 MSK 配置，您将获得配置的元数据。要查看配置的描述，请改用 `describe-configuration-revision` 命令。

- 运行以下命令，并将 *configuration-arn* 替换为您在创建配置时获取的 ARN。如果您在创建配置时未保存 ARN，则可使用 `list-configurations` 命令列出您账户中的所有配置，并在响应中显示的列表中找到所需的配置。配置的 ARN 也将显示在该列表中。

```
aws kafka describe-configuration-revision --arn configuration-arn --revision 1
```

以下是运行此命令后的成功响应示例。

```
{
    "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/abcdabcd-
abcd-1234-abcd-abcd123e8e8e-1",
    "CreationTime": "2019-05-21T00:54:23.591Z",
    "Description": "Example configuration description.",
    "Revision": 1,
    "ServerProperties":
 "YXV0by5jcmVhdGUudG9waWNzLmVuYWJsZSA9IHRydWUKCgp6b29rZWVwZXIuY29ubmVjdGlvbi50aW1lb3V0Lm1zID0gMTAwM
}
```

的值ServerProperties使用 base64 进行编码。如果您使用 base64 解码器（例如 https://www.base64decode.org/）对其进行手动解码，则会获得用于创建自定义配置的原始配置文件的内容。在此情况下，您将获得以下内容：

```
auto.create.topics.enable = true

zookeeper.connection.timeout.ms = 1000

log.roll.ms = 604800000
```

列出您的账户中当前区域的所有 MSK 配置

- 运行以下命令。

```
aws kafka list-configurations
```

以下是运行此命令后的成功响应示例。

```
{
    "Configurations": [
        {
            "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/
abcdabcd-abcd-1234-abcd-abcd123e8e8e-1",
            "CreationTime": "2019-05-21T00:54:23.591Z",
            "Description": "Example configuration description.",
            "KafkaVersions": [
                "1.1.1"
            ],
            "LatestRevision": {
                "CreationTime": "2019-05-21T00:54:23.591Z",
                "Description": "Example configuration description.",
                "Revision": 1
            },
            "Name": "SomeTest"
        },
        {
            "Arn": "arn:aws:kafka:us-east-1:123456789012:configuration/SomeTest/
abcdabcd-1234-abcd-1234-abcd123e8e8e-1",
            "CreationTime": "2019-05-03T23:08:29.446Z",
            "Description": "Example configuration description.",
            "KafkaVersions": [
                "1.1.1"
            ],
            "LatestRevision": {
                "CreationTime": "2019-05-03T23:08:29.446Z",
                "Description": "Example configuration description.",
```

```
                "Revision": 1
            },
            "Name": "ExampleConfigurationName"
        }
    ]
}
```

MSK 无服务器

> **Note**
>
> MSK Serverless Serverless Serverless Serverless Serverses Serverses Serverless Serverless Serverse Serverse Serverse Serverse Serverse Serverse Serverse Serverse Serverse Serverse Ser

MSK 无服务器是 Amazon MSK 的一种集群类型，它使您无需管理和扩展集群容量即可运行 Apache Kafka。它在管理主题中的分区的同时自动预置和扩展容量，因此您可以流式传输数据，而无需考虑调整集群规模或扩展集群。MSK Serverless Serverless Serverse Serverse Server Server Servers 如果您的应用程序需要可自动向上和向下扩展的按需流式处理容量，请考虑使用无服务器集群。

MSK 无服务器与 Apache Kafka 完全兼容，因此您可以使用任何兼容的客户端应用程序来生成和使用数据。它还与以下服务集成在一起：

- Amazon PrivateLink提供私有连接
- Amazon Identity and Access Management(IAM) 进行身份验证和授权
- Amazon Glue架构管理的架构注册表
- Amazon Kinesis Data Analytics
- Amazon Lambda用于事件处理

MSK 无服务器要求对所有群集进行 IAM 访问控制。有关更多信息，请参阅 the section called “IAM 访问控制” (p. 65)。

有关适用于 MSK 无服务器的服务配额的信息，请参阅the section called “无服务器集群的配额” (p. 108).

要帮助您开始使用无服务器集群，并详细了解无服务器群集的配置和监控选项，请参阅以下内容。

主题

开始使用 MSK 无服务器群集

本教程向您展示了如何创建 MSK Serverless 集群、创建可以访问该集群的客户端计算机以及使用该客户端在群集上创建主题并将数据写入这些主题的示例。本练习并未提供您在创建无服务器集群时可以选择的所有选项。为了简单起见，我们在本练习的各个部分均选择默认选项。这并不意味着它们是可用于设置无服务器集群的唯一选项。您也可以使用Amazon CLI或Amazon MSK API。有关更多信息，请参阅 。亚马逊 MSK API 参考 2.0.

主题

第 1 步：创建 MSK 无服务器集群

在这一步，您将执行下面两个任务。首先，使用默认设置创建 MSK 无服务器集群。其次，收集集群的信息。这是您在后续步骤中创建可向集群发送数据的客户端时需要的信息。

创建无服务器集群

1. 登录到Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home.
2. 选择创建集群。
3. 适用于创建方法，离开快速创建选项已选择。这些区域有：快速创建选项允许您使用缺省设置创建无服务器群集。
4. 适用于集群名称，输入一个描述性名称，例如**msk-serverless-tutorial-cluster**.
5. 适用于常规集群属性，选择Serverless (无服务器)作为集群类型. 将其余使用默认值常规集群属性。
6. 注意下面的表格所有集群设置. 此表列出了网络连接和可用性等重要设置的默认值，并指明了在创建集群后是否可以更改每个设置。要在创建集群之前更改设置，应选择自定义创建选项位于创建方法.

   Note

   您可以通过 MSK 无服务器群集连接来自多达五个不同 VPC 的客户端。要帮助客户端应用程序在中断时切换到另一个可用区，您必须在每个 VPC 中至少指定两个子网。
7. 选择创建集群。

收集集群的信息

1. 在集群摘要部分中，选择。查看客户端信息. 在 Amazon MSK 完成集群创建之前，此按钮将一直显示为灰色。您可能需要等待几分钟，直到按钮变为活动状态，然后才能使用它。
2. 复制标签下的字符串端点. 这是你的引导服务器字符串。
3. 选择 Properties（属性）选项卡。
4. 在联网设置部分中，复制子网和安全组的 ID 并将其保存，因为稍后需要这些信息来创建客户端计算机。
5. 选择任意子网。这将打开亚马逊 VPC 控制台。查找与子网关联的 Amazon VPC 的 ID。保存此 Amazon VPC ID 以供将来使用。

下一步

第 2 步：创建 IAM 角色 (p. 43)

第 2 步：创建 IAM 角色

在这一步，您将执行下面两个任务。第一个任务是创建一个 IAM 策略，该策略授予在集群上创建主题以及向这些主题发送数据的权限。第二个任务是创建 IAM 角色并将此策略与其关联。在后面的步骤中，我们将创建一个担任此角色的客户端计算机，并使用它在集群上创建主题并向该主题发送数据。

创建允许创建主题并写入主题的 IAM 策略

1. 通过以下网址打开 IAM 控制台：https://console.aws.amazon.com/iam/。
2. 在导航窗格中，选择。策略.
3. 请选择Create Policy（创建策略）。
4. 选择JSON选项卡，然后将编辑器窗口中的 JSON 替换为以下 JSON。

   Replace*##*使用了Amazon Web Services 区域您在其中创建集群。Replace*## ID*使用您的账户 ID。Replace*msk-serverless-tutorial-cluster*使用您的无服务器集群的名称。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:Connect",
                "kafka-cluster:AlterCluster",
                "kafka-cluster:DescribeCluster"
            ],
            "Resource": [
                "arn:aws:kafka:region:Account-ID:cluster/msk-serverless-tutorial-
cluster/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:*Topic*",
                "kafka-cluster:WriteData",
                "kafka-cluster:ReadData"
            ],
            "Resource": [
                "arn:aws:kafka:region:Account-ID:topic/msk-serverless-tutorial-cluster/
*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:AlterGroup",
                "kafka-cluster:DescribeGroup"
            ],
            "Resource": [
                "arn:aws:kafka:region:Account-ID:group/msk-serverless-tutorial-cluster/
*"
            ]
        }
    ]
}
```

有关如何编写安全策略的说明，请参阅the section called “IAM 访问控制” (p. 65).

5. 选择 Next:。标签。
6. 选择 Next:。审核。
7. 对于 Policy name (策略名称)，输入一个描述性名称，例如**msk-serverless-tutorial-policy**.
8. 选择 Create policy（创建策略）。

创建 IAM 角色并向其附加此策略

1. 在导航窗格中，选择。角色.
2. 选择Create role（创建角色）。
3. UNDER常见用例，选择EC2，然后选择。后续：Permissions (下一步：权限)。
4. 在搜索框中，输入先前为本教程创建的策略名称。然后，选中策略左侧的框。
5. 选择 Next:。标签。
6. 选择 Next:。审核。
7. 对于 Role name (角色名称)，输入一个描述性名称，例如**msk-serverless-tutorial-role**.
8. 选择Create role（创建角色）。

下一步

第 3 步：创建客户端计算机 (p. 45)

第 3 步：创建客户端计算机

在步骤中，您将执行下面两个任务。第一个任务是创建一个用作 Apache Kafka 客户端计算机的 Amazon EC2 实例。第二个任务是在机器上安装 Java 和 Apache Kafka 工具。

创建客户端计算机

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 选择 Launch instance (启动实例)。
3. 输入 Desption名称用于您的客户端计算机，例如**msk-serverless-tutorial-client**.
4. 离开亚马逊 Linux 2 AMI (HVM)-内核 5.10, 固态硬盘类型)-内核 5.10,已选择Amazon 系统映像 (AMI) 类型.
5. 离开t2.micro已选择实例类型。
6. UNDER密钥对（登录），选择创建新key pair. Enter**MSKServerlessKeyPair**为了密钥对名称. 然后，选择 Download Key Pair (下载密钥对)。此外，您还可使用现有密钥对。
7. 适用于Network settings (网络设置)，选择编辑.
8. UNDERVPC中，输入无服务器集群的虚拟私有云 (VPC) 的 ID。这是基于 Amazon VPC 服务的 VPC，您在创建集群后保存了该 VPC 的 ID。
9. 适用于子网，选择您在创建集群后保存其 ID 的子网。
10. 适用于防火墙（安全组）中，选择与集群关联的安全组。如果该安全组具有允许从安全组到自身的流量的入站规则，则此值有效。有了这样的规则，同一个安全组的成员可以相互通信。有关更多信息，请参阅 。安全组规则（在 Amazon VPC 开发人员指南中）。
11. 展开高级详细信息部分并选择您在中创建的 IAM 角色第 2 步：创建 IAM 角色 (p. 43).
12. 选择启动。
13. 在左侧导航窗格中，选择 Instances (实例)。然后选中代表您新创建的 Amazon EC2 实例的行中的复选框。从现在开始，我们将这个实例称为客户端计算机.
14. 选择Connect (连接)，然后按照说明执行操作，连接到客户端计算机。

在客户端计算机上设置 Apache Kafka 客户端工具

1. 要安装 Java，请在客户端计算机上运行以下命令：

```
sudo yum -y install java-11
```

2. 要获取创建主题和发送数据所需的 Apache Kafka 工具，请运行以下命令：

```
wget https://archive.apache.org/dist/kafka/2.8.1/kafka_2.12-2.8.1.tgz
```

```
tar -xzf kafka_2.12-2.8.1.tgz
```

3. 转至`kafka_2.12-2.8.1/libs`目录，然后运行以下命令以下载 Amazon MSK IAM JAR 文件。借助 Amazon MSK IAM JAR，客户端计算机可以访问集群。

```
wget https://github.com/aws/aws-msk-iam-auth/releases/download/v1.1.1/aws-msk-iam-auth-1.1.1-all.jar
```

4. 转至`kafka_2.12-2.8.1/bin`目录。复制以下属性设置并将其粘贴到新文件中。将该文件命名为`client.properties`然后保存。

```
security.protocol=SASL_SSL
sasl.mechanism=AWS_MSK_IAM
sasl.jaas.config=software.amazon.msk.auth.iam.IAMLoginModule required;
sasl.client.callback.handler.class=software.amazon.msk.auth.iam.IAMClientCallbackHandler
```

下一步

第 4 步：创建 Apache Kafka 主题 (p. 46)

第 4 步：创建 Apache Kafka 主题

在此步骤中，您将使用之前创建的客户端计算机在无服务器集群上创建主题。

创建主题并向其中写入数据

1. 在以下export命令，替换*my-endol*使用您在创建群集后保存的引导服务器字符串。然后，转至kafka_2.12-2.8.1/bin目录并运行export命令。

```
export BS=my-endpoint
```

2. 运行以下命令以创建名为的主题msk-serverless-tutorial.

```
<path-to-your-kafka-installation>/bin/kafka-topics.sh --bootstrap-server $BS --command-
config client.properties --create --topic msk-serverless-tutorial --partitions 6
```

下一步

第 5 步：生成和使用数据 (p. 46)

第 5 步：生成和使用数据

在此步骤中，您将使用在上一步中创建的主题生成和使用数据。

生成和使用消息

1. 运行以下命令以创建控制台制作者。

```
<path-to-your-kafka-installation>/bin/kafka-console-producer.sh --broker-list $BS --
producer.config client.properties --topic msk-serverless-tutorial
```

2. 输入所需的任何消息，然后按 Enter。重复执行此步骤两次或三次。每次你输入一行并按Enter，该行会作为单独的消息发送到集群。
3. 将与客户端计算机的连接保持打开状态，然后在新窗口中打开与该计算机的第二个单独连接。
4. 使用与客户端计算机的第二个连接，使用以下命令以创建控制台使用者。Replace*my-endol*使用您在创建群集后保存的引导服务器字符串。

```
<path-to-your-kafka-installation>/bin/kafka-console-consumer.sh --bootstrap-server my-
endpoint --consumer.config client.properties --topic msk-serverless-tutorial --from-
beginning
```

   您开始看到之前使用控制台生成器命令时输入的消息。
5. 在生成器窗口中输入更多消息，并观察消息显示在使用器窗口中。

下一步

第 6 步：Delete resources (p. 47)

第 6 步：Delete resources

在此步骤中，您将删除您按照本教程中的说明创建的资源。

删除集群

1. 从打开 Amazon MSK 控制台https://console.aws.amazon.com/msk/home.
2. 在集群列表中，选择为本教程创建的集群。
3. 适用于操作，选择删除集群.
4. Enter`delete`在字段中，选择。Delete.

停止客户端计算机

1. 通过以下网址打开 Amazon EC2 控制台：https://console.aws.amazon.com/ec2/。
2. 在 Amazon EC2 实例列表中，选择为本教程创建的客户端计算机。
3. 选择实例状态，然后选择。终止实例.
4. 选择 Terminate (终止)。

删除 IAM 策略和角色

1. 通过以下网址打开 IAM 控制台：https://console.aws.amazon.com/iam/。
2. 在导航窗格中，选择。角色.
3. 在搜索框中，输入为本教程创建的 IAM 角色名称。
4. 选择角色。然后选择 。删除角色，并确认删除。
5. 在导航窗格中，选择。策略.
6. 在搜索框中，输入为本教程创建的策略名称。
7. 选择策略以打开其摘要页面。关于保单的摘要页面上，选择。删除策略.
8. 选择 Delete（删除）。

无服务器群集的配置

Amazon MSK 为无服务器集群设置代理配置属性。您无法更改这些代理配置属性设置。但是，您可以设置以下主题配置属性。

配置属性	默认值	editable	允许的最大值
清理。政策	删除	是，但仅限于主题创建时间	
compression.type	创建者	是	
max.message.bytes	1048588	是	8 MiB
message.timestamp.difference.max.ms	long.max	是	
message.timestamp.type	CreateTime	是	

配置属性	默认值	editable	允许的最大值
保留 .bytes	250 GiB	是	250 GiB
保留 .ms	1 天	是	1 天

您还可以使用 Apache Kafka 命令为新主题或现有主题设置或修改主题级别的配置属性。有关主题级别的配置属性以及如何设置这些属性的示例的更多信息，请参阅主题级别的配置在官方 Apache Kafka 文档中。

监控无服务器集群

亚马逊 MSK 与亚马逊集成 CloudWatch 以便您可以为您的收集、查看和分析指标。下表中显示的指标适用于所有无服务器集群。由于这些指标是作为主题中每个分区的单独数据点发布的，因此我们建议将它们作为“SUM” 统计数据进行查看，以获得主题级视图。

Amazon MSK 发布`PerSec`指标到 CloudWatch 频率为每分钟一次。这意味着一分钟的 “SUM” 统计数据准确地表示了每秒的数据`PerSec`指标。要收集时间超过一分钟的每秒数据，请使用以下 CloudWatch 数学表达式：`m1 * 60/PERIOD(m1)`.

在默认监控级别可用的指标

名称	可见时间	Dimensions	描述
`BytesInPerSec`	在制作人写话题之后	集群名称、主题	每秒从客户端接收的字节数。此指标适用于每个经纪商，也适用于每个主题。
`BytesOutPerSec`	消费者组从某个主题消费后	集群名称、主题	每秒发送到客户端的字节数。此指标适用于每个经纪商，也适用于每个主题。
`FetchMessageConversionsPerSec`	消费者组从某个主题消费后	集群名称、主题	代理每秒提取消息转换的次数。
`MaxEstimatedTimeLag`	消费者组从某个主题消费后	集群名称、使用组、主题	的时间估计 MaxOffsetLag 指标。
`MaxOffsetLag`	消费者组从某个主题消费后	集群名称、使用组、主题	主题中所有分区的最大偏移滞后。
`MessagesInPerSec`	在制作人写话题之后	集群名称、主题	代理每秒传入消息数。
`ProduceMessageConversionsPerSec`	在制作人写话题之后	集群名称、主题	代理每秒生成的消息转换数。
`SumOffsetLag`	消费者组从某个主题消费后	集群名称、使用组、主题	主题中所有分区的聚合偏移滞后。

查看 MSK 无服务器指标

1. 登录到Amazon Web Services Management Console然后打开 CloudWatch 控制台位于https://console.aws.amazon.com/cloudwatch/.
2. 在导航窗格中的下指标，选择所有指标.
3. 在指标中搜索该词**`kafka`**.
4. 选择AWS/kafka/集群名称、主题要么AWS/kafka/集群名称、使用者组、主题查看不同的指标。

集群状态

下表显示了集群可能状态并描述了它们的含义。它还描述了当群集处于这些状态之一时，您可以和不能执行哪些操作。要了解集群的状态，你可以访问Amazon Web Services Management Console. 您也可以使用describe-cluster-v2命令或DescribeClusterV2操作来描述集群。群集的描述包括其状态。

集群状态	含义和可能的行动
ACTIVE (处于活动状态)	您可以生成和使用数据。您还可以执行亚马逊 MSK API 和Amazon CLI群集上的操作。
CREATING	亚马逊 MSK 正在设置集群。您必须等待集群达到 ACTIVE 状态，然后才能使用集群生成或使用数据，或者执行 Amazon MSK API 或Amazon CLI对其进行操作。
DELETING	正在删除集群。你不能用它来生成或消费数据。您也无法执行亚马逊 MSK API 或Amazon CLI对其进行操作。
FAILED	集群创建或删除过程失败。您不能使用集群来生成或使用数据。您可以删除集群但无法执行 Amazon MSK API 或Amazon CLI更新它的操作。
愈合	亚马逊 MSK 正在运行内部操作，比如更换不健康的经纪商。例如，经纪商可能无响应。您仍可以使用集群来生成和使用数据。但是，您无法执行亚马逊 MSK API 或Amazon CLI在集群上更新操作，直到集群返回 ACTIVE 状态。
保养	Amazon MSK 正在对群集执行例行维护操作。此类维护操作包括安全修补。您仍可以使用集群来生成和使用数据。但是，您无法执行亚马逊 MSK API 或Amazon CLI在集群上更新操作，直到集群返回 ACTIVE 状态。
重新启动 _ 经纪人	亚马逊 MSK 正在重新启动经纪商。您仍可以使用集群来生成和使用数据。但是，您无法执行亚马逊 MSK API 或Amazon CLI在集群上更新操作，直到集群返回 ACTIVE 状态。
UPDATING	用户启动的亚马逊 MSK API 或Amazon CLI操作正在更新集群。您仍可以使用集群来生成和使用数据。但是，您无法执行任何额外的亚马逊 MSK API 或Amazon CLI在集群上更新操作，直到集群返回 ACTIVE 状态。

Apache KaManaged Streaming for Apache Kafka

Amazon 十分重视云安全性。作为 Amazon 客户，您将从专为满足大多数安全敏感型企业的要求而打造的数据中心和网络架构中受益。

安全性是Amazon和您的共同责任。责任共担模型将其描述为云的安全性和云中的安全性：

- 云的安全性 – Amazon 负责保护在 Amazon 云中运行 Amazon 服务的基础设施。Amazon 还向您提供可安全使用的服务。第三方审核员定期测试和验证我们的安全性的有效性，作为 Amazon Compliance Programs 的一部分. 要了解适用于Amazon Managed Streaming for Apache Kafka 的合规性计划，请参阅合规性计划计划计划计划计划计划计划计划计划计划计划计划范围内的 Amazon Web Services.
- 云中的安全性 - 您的责任由您使用的 Amazon 服务决定。您还需要对其它因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。

此文档帮助您了解如何在使用 Amazon MSK 时应用责任共担模式。以下主题说明如何配置 Amazon MSK 以实现您的安全性和合规性目标。您还会了解如何使用其他Amazon Web Services 以帮助您监控和保护 Amazon MSK 资源。

主题

Amazon Managed Managed Streaming for Apache Kafka

这些区域有：Amazon 责任担担模式适用于Amazon Managed Streaming for Apache Kafka 中的数据保护。如该模式中所述，Amazon 负责保护运行所有 Amazon Web Services 云 的全球基础设施。您负责维护对托管在此基础设施上的内容的控制。此内容包括您所使用的 Amazon Web Services 的安全配置和管理任务。有关数据隐私的更多信息，请参阅数据隐私常见问题。

出于数据保护目的，我们建议您保护 Amazon Web Services 账户 凭证并使用 Amazon Identity and Access Management (IAM) 设置单独的用户账户。这仅向每个用户授予履行其工作职责所需的权限。我们还建议您通过以下方式保护您的数据：

- 对每个账户使用 Multi-Factor Authentication (MFA)。
- 使用 SSL/TLS 与 Amazon 资源进行通信。建议使用 TLS 1.2 或更高版本。
- 使用 Amazon CloudTrail 设置 API 和用户活动日志记录。

- 使用 Amazon 加密解决方案以及 Amazon 服务中的所有默认安全控制。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Simple Storage Service（Amazon S3）中的个人数据。
- 如果在通过命令行界面或 API 访问 Amazon 时需要经过 FIPS 140-2 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅《美国联邦信息处理标准 (FIPS) 第 140-2 版》。

我们强烈建议您切勿将机密信息或敏感信息（例如您客户的电子邮件地址）放入标签或自由格式字段（例如名称字段）。这包括您与Amazon MSK 或其他合作时Amazon使用控制台、API 的服务Amazon CLI，或AmazonSDK。您在用于名称的标签或自由格式字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供 URL，我们强烈建议您不要在 URL 中包含凭证信息来验证您对该服务器的请求。

主题

Amazon

Amazon MSK 提供数据加密选项，您可以使用这些选项来满足严格的数据管理要求。亚马逊 MSK 用于加密的证书必须每 13 个月更新一次。Amazon MSK 会自动为所有集群续订这些证书。在它启动证书更新操作时，它会将集群状态设置为 `MAINTENANCE`。待更新完成后，它会将集群状态重新设置为 `ACTIVE`。当集群处于 `MAINTENANCE` 状态时，您可以继续生成和使用数据，但无法对数据执行任何更新操作。

静态加密

Amazon MSK 与集成Amazon Key Management Service(KMS) 提供透明的服务器端加密。Amazon MSK 始终加密您的静态数据。在创建 MSK 集群时，您可以指定Amazon KMS keyAmazon MSK 可以用来加密您的静态数据。如果您未指定 KMS 密钥，则 Amazon MSK 会创建一个 KMS 密钥Amazon 托管式密钥代表您使用并代表您使用它。有关 KMS 密钥的更多信息，请参阅《Amazon Key Management Service 开发人员指南》中的 Amazon KMS keys。

传输中加密

Amazon MSK 使用 TLS 1.2。默认情况下，它加密 MSK 集群代理之间传输的数据。可以在创建集群时覆盖此默认值。

对于客户端和代理之间的通信，您必须指定下列三项设置之一：

- 仅允许 TLS 加密数据。这是默认设置。
- 同时允许明文数据和 TLS 加密数据。
- 仅允许明文数据。

亚马逊 MSK 经纪人公开使用Amazon Certificate Manager证书。因此，任何信任亚马逊信托服务的信任库也信任亚马逊 MSK 经纪人的证书。

虽然我们强烈建议启用传输中加密，但它可能会增加额外的 CPU 开销和几毫秒的延迟。但是，大多数使用案例对这些差异并不敏感，影响的程度取决于集群、客户端和使用情况配置文件的配置。

如何开始使用加密？

创建 MSK 集群时，您可以以 JSON 格式指定加密设置。以下是示例。

```
{
   "EncryptionAtRest": {
```

```
       "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/abcdabcd-1234-
abcd-1234-abcd123e8e8e"
    },
   "EncryptionInTransit": {
        "InCluster": true,
        "ClientBroker": "TLS"
    }
}
```

对于DataVolumeKMSKeyId，您可以指定客户托管密钥或者Amazon 托管式密钥在您的账户中使用 MSK (alias/aws/kafka)。如果您不指定EncryptionAtRest，Amazon MSK 仍将您的静态数据加密Amazon 托管式密钥. 要确定您的集群正在使用哪个密钥，请发送GET请求或调用DescribeClusterAPI 操作。

对于EncryptionInTransit，默认值为InCluster是真的，但是如果您不想让 Amazon MSK 在数据在经纪人之间传递时对其进行加密，则可以将其设置为 false。

要为客户端和代理之间传输的数据指定加密模式，请将 ClientBroker 设置为以下三个值之一：TLS、TLS_PLAINTEXT 或 PLAINTEXT。

创建集群时指定加密设置

1. 将上一示例的内容保存在文件中，并为该文件指定所需的任何名称。例如，将其命名为 encryption-settings.json。
2. 运行 create-cluster 命令并使用 encryption-info 选项指向您保存配置 JSON 的文件。以下是示例。

```
aws kafka create-cluster --cluster-name "ExampleClusterName" --broker-node-group-info
 file://brokernodegroupinfo.json --encryption-info file://encryptioninfo.json --kafka-
version "2.2.1" --number-of-broker-nodes 3
```

以下是运行此命令后的成功响应示例。

```
{
    "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/SecondTLSTest/
abcdabcd-1234-abcd-1234-abcd123e8e8e",
    "ClusterName": "ExampleClusterName",
    "State": "CREATING"
}
```

测试 TLS 加密

1. 按照the section called “第 2 步：创建客户端计算机” (p. 5)中的指导创建客户端计算机。
2. 在客户端计算机上安装 Apache Kafka。
3. 在已安装 Amazon CLI 的计算机上运行以下命令，并将 *ClussterARN* 替换为集群 ARN（如前面过程中的示例所示，将 ClientBroker 设置为 TLS 创建的集群）。

```
aws kafka describe-cluster --cluster-arn clusterARN
```

在结果中，查找 ZookeeperConnectString 的值并保存它，因为您需要在下一步中使用该值。

4. 在您的客户机上运行以下命令以创建主题。Replace（替换）*ZookeeperConnectString*使用您获得的值ZookeeperConnectString在上一步中。

```
<path-to-your-kafka-installation>/bin/kafka-topics.sh --create --
zookeeper ZookeeperConnectString --replication-factor 3 --partitions 1 --topic
 TLSTestTopic
```

5. 在此示例中，我们使用 JVM 信任库与 MSK 集群通信。为此，请首先在客户端计算机上创建一个名为 /tmp 的文件夹。然后，转到 Apache Kafka 安装的 bin 文件夹，并运行以下命令。（您的 JVM 路径可能不相同。）

```
cp /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.201.b09-0.amzn2.x86_64/jre/lib/security/
cacerts /tmp/kafka.client.truststore.jks
```

6. 仍在客户端计算机上的 Apache Kafka 安装的 bin 文件夹中，创建一个名为 client.properties 的文本文件，该文件包含以下内容。

```
security.protocol=SSL
ssl.truststore.location=/tmp/kafka.client.truststore.jks
```

7. 在已安装 Amazon CLI 的计算机上运行以下命令，将 *clusterARN* 替换为集群 ARN。

```
aws kafka get-bootstrap-brokers --cluster-arn clusterARN
```

成功结果如下所示。保存此结果，因为您需要在下一步中使用它。

```
{
    "BootstrapBrokerStringTls": "a-1.example.g7oein.c2.kafka.us-
east-1.amazonaws.com:0123,a-3.example.g7oein.c2.kafka.us-
east-1.amazonaws.com:0123,a-2.example.g7oein.c2.kafka.us-east-1.amazonaws.com:0123"
}
```

8. 运行以下命令可在客户端计算机上创建控制台制作器。Replace（替换）*BootstrapBrokerStringTls*使用您在上一步中获得的值。保持运行此生成器命令。

```
<path-to-your-kafka-installation>/bin/kafka-console-producer.sh --broker-
list BootstrapBrokerStringTls --producer.config client.properties --topic TLSTestTopic
```

9. 打开一个新的命令窗口并连接到同一台客户机。然后，运行以下命令以创建控制台消费者。

```
<path-to-your-kafka-installation>/bin/kafka-console-consumer.sh --bootstrap-
server BootstrapBrokerStringTls --consumer.config client.properties --topic
 TLSTestTopic
```

10. 在 producer 窗口中，键入一条短信，然后在消费者窗口中查找相同的消息。亚马逊 MSK 对传输中的此消息进行了加密。

有关配置 Apache Kafka 客户端以使用加密数据的更多信息，请参阅配置 Kafka 客户端。

Amazon MSK API 的身份验证和授权

Amazon Identity and Access Management (IAM) 是一项 Amazon Web Service，可以帮助管理员安全地控制对 Amazon 资源的访问。IAM 管理员控制谁可以成为身份验证（已登录）和授权（有权限）使用亚马逊 MSK 资源。IAM 是一项无需额外费用即可使用的 Amazon Web Service。

此页面介绍如何使用 IAM 控制可以执行此操作的人员Amazon在集群上。有关如何控制谁可以在您的集群上执行 Apache Kafka 操作的信息，请参阅the section called “Apache Kafka API 的身份验证和授权” (p. 65).

主题

Amazon MSK 如何与 IAM 协同工作

在使用 IAM 管理对 Amazon MSK 的访问权限之前，您应该了解哪些 IAM 功能可用于 Amazon MSK。大致了解Amazon MSK 和其他人如何使用 Amazon MSKAmazon服务与 IAM 配合使用，请参阅Amazon使用 IAM 的服务在里面IAM 用户指南.

主题

Amazon MSK 基于身份的策略

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。Amazon MSK 支持特定的操作、资源和条件键。要了解在 JSON 策略中使用的所有元素，请参阅 IAM 用户指南 中的 IAM JSON 策略元素参考。

操作

管理员可以使用 Amazon JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体 可以对什么资源 执行操作，以及在什么 条件 下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 Amazon API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行相关操作的权限。

Amazon MSK 中的策略操作在操作前面使用以下前缀：`kafka:`. 例如，授予某人使用亚马逊 MSK 描述一个 MSK 集群的权限`DescribeCluster`API 操作，你包括`kafka:DescribeCluster`在他们的政策中采取行动。策略语句必须包含 `Action` 或 `NotAction` 元素。Amazon MSK 定义了一组自己的操作，以描述您可以使用该服务执行的任务。

要在单个语句中指定多项操作，请使用逗号将它们隔开，如下所示：

```
"Action": ["kafka:action1", "kafka:action2"]
```

您也可以使用通配符 (*) 指定多个操作。例如，要指定以单词 `Describe` 开头的所有操作，包括以下操作：

```
"Action": "kafka:Describe*"
```

要查看Amazon MSK 操作的列表，请参阅Amazon Managed Streaming for Apache Kafka 的操作、资源和条件键在里面IAM 用户指南.

资源

管理员可以使用 Amazon JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体 可以对什么资源 执行操作，以及在什么条件 下执行。

`Resource` JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 `Resource` 或 `NotResource` 元素。作为最佳实践，请使用其 Amazon Resource Name (ARN) 指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符 (*) 指示语句应用于所有资源。

```
"Resource": "*"
```

Amazon MSK 实例资源拥有以下 ARN：

```
arn:${Partition}:kafka:${Region}:${Account}:cluster/${ClusterName}/${UUID}
```

有关 ARN 格式的更多信息，请参阅 Amazon Resource Name（ARN）和 Amazon 服务命名空间。

例如，要在语句中指定 `CustomerMessages` 实例，请使用以下 ARN：

```
"Resource": "arn:aws:kafka:us-east-1:123456789012:cluster/CustomerMessages/abcd1234-abcd-
dcba-4321-a1b2abcd9f9f-2"
```

要指定属于特定账户的所有实例，请使用通配符 (*)：

```
"Resource": "arn:aws:kafka:us-east-1:123456789012:cluster/*"
```

无法对特定资源执行某些 Amazon MSK 操作，例如，用于创建资源的操作。在这些情况下，您必须使用通配符 (*)。

```
"Resource": "*"
```

要在单个语句中指定多个资源，请使用逗号分隔 ARN。

```
"Resource": ["resource1", "resource2"]
```

要查看 Amazon MSK 资源类型及其 ARN 的列表，请参阅Amazon Managed Managed Streaming for Apache Kafka在里面IAM 用户指南. 要了解您可以使用哪些操作指定每个资源的 ARN，请参阅Amazon Managed Managed Streaming for Apache Kafka.

条件键

管理员可以使用 Amazon JSON 策略来指定谁有权访问什么内容。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 `Condition` 元素（或 `Condition` 块）中，可以指定语句生效的条件。`Condition` 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 `Condition` 元素，或在单个 `Condition` 元素中指定多个键，则 Amazon 使用逻辑 `AND` 运算评估它们。如果您为单个条件键指定多个值，则 Amazon 使用逻辑 `OR` 运算来评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅 IAM 用户指南 中的 IAM policy 元素：变量和标签。

Amazon支持全局条件键和特定于服务的条件键。要查看所有 Amazon 全局条件键，请参阅《IAM 用户指南》中的 Amazon 全局条件上下文键。

Amazon MSK 定义了自己的一组条件键，还支持使用一些全局条件键。要查看所有 Amazon 全局条件键，请参阅《IAM 用户指南》中的 Amazon 全局条件上下文键。

要查看Amazon MSK 条件键的列表，请参阅Amazon Managed Managed Streaming for Apache Kafka在里面IAM 用户指南. 要了解您可以对哪些操作和资源使用条件键，请参阅Amazon Managed Managed Streaming for Apache Kafka.

示例

要查看 Amazon MSK 基于身份的策略的示例，请参阅Amazon MSK 基于身份的策略示例 (p. 57).

Amazon MSK 基于资源的策略

Amazon MSK 不支持基于资源的策略。

Amazon 托管策略

基于 Amazon MSK 标签的授权

您可以将标签附加到 Amazon MSK 集群。要基于标签控制访问，您需要使用 `kafka:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。有关标记 Amazon MSK 资源的更多信息，请参阅the section called “为集群添加标签” (p. 28).

要查看基于身份的策略（用于基于集群上的标签来限制对该集群的访问）的示例，请参阅根据标签访问亚马逊 MSK 集群 (p. 58)。

Amazon

网络 ACL 和安全组都允许 (因此可到达您的实例) 的发起 ping 的IAM 角色是您的 Amazon Web Services 账户中具有特定权限的实体。

对 Amazon MSK 使用临时凭证

您可以使用临时凭证进行联合身份登录，担任 IAM 角色或担任跨账户角色。您可以通过调用获得临时安全证书Amazon STSAPI 操作，例如AssumeRole要么GetFederationToken.

Amazon MSK 支持使用临时凭证。

服务相关角色

服务相关角色允许 Amazon Web Services 访问其他服务中的资源以代表您完成操作。服务相关角色显示在您的 IAM 账户中，并归该服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

Amazon MSK 支持服务相关角色。有关创建或管理 Amazon MSK 服务相关角色的详细信息，请参阅the section called “服务相关角色” (p. 59).

Amazon MSK 基于身份的策略示例

预设情况下，IAM 用户和角色没有执行亚马逊 MSK API 操作的权限。IAM 管理员必须创建 IAM policy，以便为用户和角色授予权限以对所需的指定资源执行特定的 API 操作。然后，管理员必须将这些策略附加到需要这些权限的 IAM 用户或组。

要了解如何使用这些示例 JSON 策略文档创建 IAM 基于身份的策略，请参阅 IAM 用户指南 中的在 JSON 选项卡上创建策略。

主题

策略最佳实践

基于身份的策略确定某个人是否可以创建、访问或删除您账户中的 Amazon MSK 资源。这些操作可能会使 Amazon Web Services 账户 产生成本。创建或编辑基于身份的策略时，请遵循以下准则和建议：

- Amazon 托管策略及转向最低权限许可入门 - 要开始向用户和工作负载授予权限，请使用 Amazon 托管策略来为许多常见使用场景授予权限。您可以在 Amazon Web Services 账户 中找到这些策略。我们建议通过定义特定于您的使用场景的 Amazon 客户管理型策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的 Amazon 托管策略或工作职能的 Amazon 托管策略。
- 应用最低权限 – 在使用 IAM policy 设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM policy 中的条件进一步限制访问权限 – 您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果通过特定 Amazon Web Service（例如 Amazon CloudFormation）使用服务操作，您还可以使用条件来授予对服务操作的访问权限。有关更多信息，请参阅 。IAM JSON 策略元素：Condition在里面IAM 用户指南.
- 使用 IAM Access Analyzer 验证您的 IAM policy，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM policy语言 (JSON) 和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的 IAM Acess Analyzer 策略验证。
- 需要多重身份验证 (MFA) – 如果您的账户需要 IAM 用户或根用户，请启用 MFA 来提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的配置受 MFA 保护的 API 访问。

有关 IAM 中的最佳实践的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

允许用户查看他们自己的权限

该示例说明了您如何创建策略，以允许 IAM 用户查看附加到其用户身份的内联和托管式策略。此策略包括在控制台上完成此操作或者以编程方式使用 Amazon CLI 或 Amazon API 所需的权限。

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

访问一个 Amazon MSK 集群

在本示例中，您想要为您的 Amazon Web Services 账户中的 IAM 用户授予访问其中一个集群的权限，`purchaseQueriesCluster`. 此策略允许用户描述集群、获取其引导代理、列出其代理节点并更新它。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"UpdateCluster",
         "Effect":"Allow",
         "Action":[
            "kafka:Describe*",
            "kafka:Get*",
            "kafka:List*",
            "kafka:Update*"
         ],
         "Resource":"arn:aws:kafka:us-east-1:012345678012:cluster/purchaseQueriesCluster/
abcdefab-1234-abcd-5678-cdef0123ab01-2"
      }
   ]
}
```

根据标签访问亚马逊 MSK 集群

您可以在基于身份的策略中使用条件，以便基于标签控制对 Amazon MSK 资源的访问。此示例演示了如何创建允许用户描述集群、获取其引导代理、列出其代理节点、更新和删除集群的策略。但是，仅当集群标签 `Owner` 的值为该用户的用户名时，才能授予此权限。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessClusterIfOwner",
      "Effect": "Allow",
      "Action": [
        "kafka:Describe*",
        "kafka:Get*",
        "kafka:List*",
        "kafka:Update*",
        "kafka:Delete*"
      ],
      "Resource": "arn:aws:kafka:us-east-1:012345678012:cluster/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Owner": "${aws:username}"
        }
      }
    }
  ]
}
```

您可以将该策略附加到您账户中的 IAM 用户。如果用户的名字是`richard-roe`尝试更新 MSK 集群，则必须对该集群进行标记`Owner=richard-roe`要么`owner=richard-roe`. 否则，他将被拒绝访问。条件标签键 `Owner` 匹配 `Owner` 和 `owner`，因为条件键名称不区分大小写。有关更多信息，请参阅 。IAM JSON 策略元素：Condition在里面IAM 用户指南.

对 Amazon MSK 使用服务相关角色

Amazon MSK 使用Amazon Identity and Access Management(IAM)服务相关角色. 服务相关角色是一种独特类型的 IAM 角色，它与 Amazon MSK 直接相关。服务相关角色由 Amazon MSK 预定义，并包含服务调用其他Amazon代表您提供的服务

服务相关角色可让您更轻松地设置 Amazon MSK，因为您不必手动添加必要的权限。Amazon MSK 定义其服务相关角色的权限。除非另有定义，否则只有亚马逊 MSK 可以担任其角色。定义的权限包括信任策略和权限策略，以及不能附加到任何其它 IAM 实体的权限策略。

有关支持服务相关角色的其他服务的信息，请参阅与 IAM 配合使用的Amazon Web Services，然后寻找有以下服务的服务是在里面服务相关角色列。请选择 Yes 与查看该服务的服务相关角色文档的链接。

主题

Amazon MSK 的服务相关角色权限

Amazon MSK 使用名为的服务相关角色AWSServiceRoleForKafka— 允许亚马逊 MSK 访问Amazon代表您提供资源。

AWSServiceRoleForKafka 服务相关角色信任以下服务代入该角色：

- `kafka.amazonaws.com`

角色权限策略允许 Amazon MSK 对指定资源完成以下操作：

- 操作：* 上的 ec2:CreateNetworkInterface
- 操作：ec2:DescribeNetworkInterfaces 上的 *
- 操作：ec2:CreateNetworkInterfacePermission 上的 *
- 操作：ec2:AttachNetworkInterface 上的 *
- 操作：ec2:DeleteNetworkInterface 上的 *
- 操作：ec2:DetachNetworkInterface 上的 *
- 操作：acm-pca:GetCertificateAuthorityCertificate 上的 *
- 操作：secretsmanager:ListSecrets 上的 *
- 操作：secretsmanager:GetResourcePolicy在带有前缀的密钥上AmazonMSK_您为Amazon MSK 创建的
- 操作：secretsmanager:PutResourcePolicy在带有前缀的密钥上AmazonMSK_您为Amazon MSK 创建的
- 操作：secretsmanager:DeleteResourcePolicy在带有前缀的密钥上AmazonMSK_您为Amazon MSK 创建的
- 操作：secretsmanager:DescribeSecret在带有前缀的密钥上AmazonMSK_您为Amazon MSK 创建的

必须配置权限，允许 IAM 实体（如用户、组或角色）创建、编辑或删除服务相关角色。有关更多信息，请参阅 IAM 用户指南中的服务相关角色权限。

为 Amazon MSK 创建服务相关角色

您无需手动创建服务相关角色。当您在中创建 Amazon MSK 集群时Amazon Web Services Management Console，Amazon CLI，或者AmazonAPI，Amazon MSK 将为您创建服务相关角色。

如果删除此服务相关角色，然后需要再次创建，可以使用相同流程在账户中重新创建此角色。当您创建 Amazon MSK 集群时，Amazon MSK 将再次为您创建服务相关角色。

编辑Amazon MSK 的服务相关角色

Amazon MSK 不允许您编辑 MSK MSK 不允许您编辑AWSServiceRoleForKafka服务相关角色。创建服务相关角色后，将无法更改角色名称，因为可能有多个实体引用该角色。但是可以使用 IAM 编辑角色说明。有关更多信息，请参见 IAM 用户指南中的编辑服务相关角色。

Amazon MSK 服务相关角色支持的区域

Amazon MSK 支持在该服务可用的所有区域中使用服务相关角色。有关更多信息，请参阅 Amazon 区域和终端节点。

AmazonAmazon MSK 的托管策略

要向用户、组和角色添加权限，与自己编写策略相比，使用 Amazon 托管策略更简单。创建仅为团队提供所需权限的 IAM 客户托管策略需要时间和专业知识。要快速入门，您可以使用我们的 Amazon 托管式策略。这些策略涵盖常见使用案例，可在您的 Amazon Web Services 账户 中使用。有关 Amazon 托管式策略的更多信息，请参阅 IAM 用户指南中的Amazon 托管式策略。

Amazon Web Services 负责维护和更新 Amazon 托管式策略。您无法更改 Amazon 托管式策略中的权限。服务偶尔会向 Amazon 托管式策略添加额外权限以支持新功能。此类更新会影响附加策略的所有身份（用户、组和角色）。当启动新功能或新操作可用时，服务最有可能会更新 Amazon 托管式策略。服务不会从 Amazon 托管式策略中删除权限，因此策略更新不会破坏您的现有权限。

此外，Amazon 还支持跨多种服务的工作职能的托管式策略。例如，`ViewOnlyAccess` Amazon 托管式策略提供对许多 Amazon Web Services 服务和资源的只读访问权限。当服务启动新功能时，Amazon 会为新操作和资源添加只读权限。有关工作职能策略的列表和说明，请参阅 IAM 用户指南中的适用于工作职能的 Amazon 托管策略。

Amazon托管策略：AmazonMSKFullAccess

此策略授予管理权限，允许委托人完全访问所有 Amazon MSK 操作。此策略中的权限分组如下：

- 亚马逊 MSK 权限允许所有亚马逊 MSK 操作。
- 此策略中的某些 Amazon EC2 权限是验证 API 请求中传递的资源所必需的。这是为了确保 Amazon MSK 能够成功使用集群的资源。此策略中的其余Amazon EC2 权限允许亚马逊 MSK 创建Amazon使您能够连接到集群所需的资源。
- 这些区域有：Amazon KMS在 API 调用期间使用权限来验证请求中传递的资源。它们是亚马逊 MSK 能够在亚马逊 MSK 集群中使用传递的密钥所必需的。
- 这些区域有： CloudWatch Amazon MSK 需要日志、Amazon S3 和 Amazon Kinesis Data Firehose 权限，才能确保日志传输目标可达，并且它们可供代理日志使用。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kafka:*",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeRouteTables",
                "ec2:DescribeVpcEndpoints",
                "ec2:DescribeVpcAttribute",
                "kms:DescribeKey",
                "kms:CreateGrant",
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups",
                "S3:GetBucketPolicy",
                "firehose:TagDeliveryStream"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateVpcEndpoint"
            ],
            "Resource": [
                "arn:*:ec2:*:*:vpc/*",
                "arn:*:ec2:*:*:subnet/*",
                "arn:*:ec2:*:*:security-group/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateVpcEndpoint"
```

```
                ],
                "Resource": [
                    "arn:*:ec2:*:*:vpc-endpoint/*"
                ],
                "Condition": {
                    "StringEquals": {
                        "aws:RequestTag/AWSMSKManaged": "true"
                    },
                    "StringLike": {
                        "aws:RequestTag/ClusterArn": "*"
                    }
                }
            },
            {
                "Effect": "Allow",
                "Action": [
                    "ec2:CreateTags"
                ],
                "Resource": "arn:*:ec2:*:*:vpc-endpoint/*",
                "Condition": {
                    "StringEquals": {
                        "ec2:CreateAction": "CreateVpcEndpoint"
                    }
                }
            },
            {
                "Effect": "Allow",
                "Action": [
                    "ec2:DeleteVpcEndpoints"
                ],
                "Resource": "arn:*:ec2:*:*:vpc-endpoint/*",
                "Condition": {
                    "StringEquals": {
                        "ec2:ResourceTag/AWSMSKManaged": "true"
                    },
                    "StringLike": {
                        "ec2:ResourceTag/ClusterArn": "*"
                    }
                }
            },
            {
                "Effect": "Allow",
                "Action": "iam:CreateServiceLinkedRole",
                "Resource": "arn:aws:iam::*:role/aws-service-role/kafka.amazonaws.com/
AWSServiceRoleForKafka*",
                "Condition": {
                    "StringLike": {
                        "iam:AWSServiceName": "kafka.amazonaws.com"
                    }
                }
            },
            {
                "Effect": "Allow",
                "Action": [
                    "iam:AttachRolePolicy",
                    "iam:PutRolePolicy"
                ],
                "Resource": "arn:aws:iam::*:role/aws-service-role/kafka.amazonaws.com/
AWSServiceRoleForKafka*"
            },
            {
                "Effect": "Allow",
                "Action": "iam:CreateServiceLinkedRole",
                "Resource": "arn:aws:iam::*:role/aws-service-role/
delivery.logs.amazonaws.com/AWSServiceRoleForLogDelivery*",
                "Condition": {
```

```
                    "StringLike": {
                        "iam:AWSServiceName": "delivery.logs.amazonaws.com"
                    }
                }
            }

        ]
    }
```

Amazon托管策略：AmazonMSKReadOnlyAccess

此策略授予只读权限，允许用户在 Amazon MSK 中查看信息。附加此政策的委托人不能进行任何更新或删除现有资源，也不能创建新的 Amazon MSK 资源。例如，拥有这些权限的委托人可以查看集群列表以及与其账户关联的配置，但不能更改任何集群的配置或设置。此策略中的权限分组如下：

- 亚马逊 MSK 权限允许您列出亚马逊 MSK 资源、描述它们并获取有关它们的信息。
- Amazon EC2 权限用于描述与集群关联的 Amazon VPC、子网、安全组和 ENI。
- 这些区域有：Amazon KMS权限用于描述与集群关联的密钥。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "kafka:Describe*",
                "kafka:List*",
                "kafka:Get*",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "kms:DescribeKey"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Amazon托管策略： KafkaServiceRolePolicy

您不能附录 KafkaServiceRolePolicy 至您的 IAM 实体。此策略附加到允许 Amazon MSK 代表您执行操作的服务相关角色。有关更多信息，请参阅 the section called “服务相关角色” (p. 59)。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface",
                "ec2:DescribeNetworkInterfaces",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:AttachNetworkInterface",
                "ec2:DeleteNetworkInterface",
                "ec2:DetachNetworkInterface",
                "acm-pca:GetCertificateAuthorityCertificate",
                "secretsmanager:ListSecrets"
            ],
```

```
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetResourcePolicy",
                "secretsmanager:PutResourcePolicy",
                "secretsmanager:DeleteResourcePolicy",
                "secretsmanager:DescribeSecret"
            ],
            "Resource": "*",
            "Condition": {
                "ArnLike": {
                    "secretsmanager:SecretId":
 "arn:*:secretsmanager:*:*:secret:AmazonMSK_*"
                }
            }
        }
    ]
}
```

AmazonAmazon托管策略

查看有关更新的详细信息Amazon自从此服务开始跟踪这些更改以来，Amazon MSK 的托管策略。

更改	说明	日期
AmazonMSKFullAccess (p. 61)—对现有策略的更新	亚马逊 MSK 添加了新的 Amazon EC2 权限，使连接到集群成为可能。	2021 年 11 月 30 日
AmazonMSKFullAccess (p. 61)—对现有策略的更新	亚马逊 MSK 添加了一项新权限，允许其描述Amazon EC2 路由表。	2021 年 11 月 19 日
Amazon MSK 已开启跟踪更改	Amazon MSK 为其 MSK 开启了跟踪更改Amazon托管策略。	2021 年 11 月 19 日

Amazon MSK 身份和访问疑难解答

可以使用以下信息，以帮助您诊断和修复在使用 Amazon MSK 和 IAM 时可能遇到的常见问题。

主题

- 我无权在 Amazon MSK 中执行操作 (p. 64)

我无权在 Amazon MSK 中执行操作

如果 Amazon Web Services Management Console 告诉您，您无权执行某个操作，则必须联系您的管理员寻求帮助。您的管理员是指为您提供用户名和密码的那个人。

以下示例错误发生在`mateojackson`IAM 用户尝试使用控制台删除集群，但没有`kafka:`*`DeleteCluster`*权限。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 kafka:DeleteCluster on resource: purchaseQueriesCluster
```

在这种情况下，Mateo 请求他的管理员更新其策略，以允许他使用 `kafka:DeleteCluster` 操作访问 `purchaseQueriesCluster` 资源。

Apache Kafka API 的身份验证和授权

您可以使用 IAM 对客户端进行身份验证，并允许或拒绝 Apache Kafka 操作。或者，您可以使用 TLS 或 SASL/SCRAM 对客户端进行身份验证，以及使用 Apache Kafka ACL 允许或拒绝操作。

有关如何控制谁可以表演的信息Amazon在您的集群上，请参阅the section called “Amazon MSK API 的身份验证和授权” (p. 53).

主题

IAM 访问控制

Amazon MSK 的 IAM 访问控制允许您处理 MSK 集群的身份验证和授权。这样就不需要使用一种身份验证机制和另一种授权机制。例如，当客户端尝试写入您的集群时，Amazon MSK 使用 IAM 来检查该客户端是否是经过身份验证的身份，以及是否有权向您的集群生成数据。

Amazon MSK 会记录访问事件，以便您可以对其进行审计。有关更多信息，请参阅 the section called “CloudTrail 事件” (p. 83)。

为了使 IAM 访问控制成为可能，亚马逊 MSK 对 Apache Kafka 源代码进行了细微的修改。这些修改不会对你的 Apache Kafka 体验造成明显的差异。

> Important
>
> IAM 访问控制不适用于 Apache 访问控制 ZooKeeper 节点。有关如何控制对这些节点的访问权限的信息，请参阅the section called “控制对 Apache 的访问 ZooKeeper” (p. 80).

> Important
>
> 这些区域有：`allow.everyone.if.no.acl.found`如果您的集群使用 IAM 访问控制，则 Apache Kafka 设置不起作用。

> Important
>
> 你可以为使用 IAM 访问控制的 MSK 集群调用 Apache Kafka ACL API。但是，Apache Kafka ACL 存储在 Apache 中 ZooKeeper 对 IAM 角色的授权没有影响。您必须使用 IAM 策略来控制 IAM 角色的访问。

亚马逊 MSK 的 IAM 访问控制的工作原理

要使用 Amazon MSK 的 IAM 访问控制，请执行以下步骤，本节其余部分将详细介绍这些步骤。

- the section called “创建使用 IAM 访问控制的集群” (p. 66)
- the section called “将客户端配置为 IAM 访问控制” (p. 66)
- the section called “创建授权策略” (p. 67)

- the section called “获取 IAM 访问控制的引导代理” (p. 67)

创建使用 IAM 访问控制的集群

本节介绍如何使用Amazon Web Services Management Console、API 或Amazon CLI创建使用 IAM 访问控制的集群。有关如何为现有集群启用 IAM 访问控制的信息，请参阅the section called “更新安全性” (p. 25).

使用Amazon Web Services Management Console创建使用 IAM 访问控制的集群

1. 在以下位置打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择创建集群。
3. 选择使用自定义设置创建集群.
4. 在里面身份验证部分，选择IAM 访问控制.
5. 完成创建集群的剩余工作流程。

使用 API 或Amazon CLI创建使用 IAM 访问控制的集群

- 要创建启用了 IAM 访问控制的集群，请使用CreateClusterAPI 或创建集群CLI 命令，并将以下 JSON 传递给`ClientAuthentication`参数：`"ClientAuthentication": { "Sasl": { "Iam": { "Enabled": true } }`.

将客户端配置为 IAM 访问控制

要使客户端能够与使用 IAM 访问控制的 MSK 集群通信，请按照以下步骤进行配置。

1. 将以下命令添加到`client.properties`file。Replace（替换）*<PATH_TO_TRUST_STORE_FILE>*使用客户端上信任存储库文件的完全限定路径。

   Note

   如果您不想使用特定证书，则可以删除`ssl.truststore.location=`*<PATH_TO_TRUST_STORE_FILE>*来自你的`client.properties`file。当您不指定的值时`ssl.truststore.location`，Java 进程使用默认证书。

   ```
   ssl.truststore.location=<PATH_TO_TRUST_STORE_FILE>
   security.protocol=SASL_SSL
   sasl.mechanism=AWS_MSK_IAM
   sasl.jaas.config=software.amazon.msk.auth.iam.IAMLoginModule required;
   sasl.client.callback.handler.class=software.amazon.msk.auth.iam.IAMClientCallbackHandler
   ```

   使用您为之创建的已命名配置文件Amazon身份证明，包括`awsProfileName="`*your profile name*`";`在客户端配置文件中，在客户端配置文件中。有关命名配置文件的信息，请参阅命名配置文件在里面Amazon CLI文档中）。
2. 下载最新的稳定版aws-msk-iam-authJAR 文件，并将其放在类路径中。如果您使用 Maven，请添加以下依赖关系，根据需要调整版本号：

   ```
   <dependency>
       <groupId>software.amazon.msk</groupId>
       <artifactId>aws-msk-iam-auth</artifactId>
       <version>1.0.0</version>
   </dependency>
   ```

Amazon MSK 客户端插件在 Apache 2.0 许可证下是开源的。

创建授权策略

将授权策略附加到与客户端对应的 IAM 角色。在授权策略中，您可以指定允许或拒绝角色的操作。如果您的客户端使用的是 Amazon EC2 实例，请将授权策略与该 Amazon EC2 实例的 IAM 角色相关联。或者，您可以将您的客户端配置为使用命名配置文件，然后将授权策略与该命名配置文件的角色相关联。the section called “将客户端配置为 IAM 访问控制” (p. 66)介绍如何将客户机配置为使用命名配置文件。

有关如何创建 IAM 策略的信息，请参阅创建 IAM policy.

以下是示例集群 MyTestCluster. 要理解的语义`Action`和`Resource`元素，请参阅the section called “动作和资源的语义” (p. 68).

> **Important**
>
> 您对 IAM 策略所做的更改将反映在 IAM API 和Amazon CLI立即。但是，可能需要很长时间才能使策略更改生效。在大多数情况下，政策更改将在不到一分钟的时间内生效。网络状况有时可能会增加延迟。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:Connect",
                "kafka-cluster:AlterCluster",
                "kafka-cluster:DescribeCluster"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:0123456789012:cluster/MyTestCluster/abcd1234-0123-
abcd-5678-1234abcd-1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:*Topic*",
                "kafka-cluster:WriteData",
                "kafka-cluster:ReadData"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:0123456789012:topic/MyTestCluster/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kafka-cluster:AlterGroup",
                "kafka-cluster:DescribeGroup"
            ],
            "Resource": [
                "arn:aws:kafka:us-east-1:0123456789012:group/MyTestCluster/*"
            ]
        }
    ]
}
```

要了解如何使用与常见 Apache Kafka 用例（例如生成和使用数据）对应的操作元素创建策略，请参阅the section called “常见用例” (p. 71).

获取 IAM 访问控制的引导代理

请参阅 the section called “获取引导代理” (p. 14)。

动作和资源的语义

本部分说明可在 IAM 授权策略中使用的操作和资源元素的语义。有关策略示例，请参阅 the section called "创建授权策略" (p. 67)。

操作

下表列出了您在使用 Amazon MSK 的 IAM 访问控制时可以在授权策略中包含的操作。当您在授权策略中包含来自的操作时操作表中的列，您还必须包含以下中的相应操作所需的操作列。

操作	描述	所需的操作	所需的 资源	适用于无服务器集群
`kafka-cluster:Connect`	授予连接和验证集群的权限。	None（无）	集群	是
`kafka-cluster:DescribeClusterV2`	授予描述集群各个方面的权限，相当于 Apache Kafka 的 DESCRIBE CLUSTER ACL	`kafka-cluster:Connect`	集群	是

`kafka-cluster:AlterCluster`	授予更改集群各个方面的权限，相当于 Apache Kafka 的 ALTER CLUSTER ACL。	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeCluster`
			集群	否

`kafka-cluster:DescribeClusterDynamicConfiguration`	授予描述集群动态配置的权限，相当于 Apache Kafka 的 DESCRIBE_CONFIGS CLUSTER ACL	`kafka-cluster:Connect`	集群	否

`kafka-cluster:AlterClusterDynamicConfiguration`	授予更改集群动态配置的权限，相当于 Apache Kafka 的 ALTER_CONFIGS CLUSTER ACL	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeClusterDynamicConfiguration`
			集群	否

`kafka-cluster:WriteDataIdempotently`	授予在集群上以幂等方式写入数据的权限，相当于 Apache Kafka 的 IDEMPOTENT_WRITE CLUSTER ACL	`kafka-cluster:Connect`
		
		`kafka-cluster:WriteData`
			集群	是

`kafka-cluster:CreateTopic`	授予在集群上创建主题的权限，相当于 Apache Kafka 的 CREATE CLUSTER/TOPIC ACL	`kafka-cluster:Connect`	topic	是
`kafka-cluster:DescribeTopic`	授予描述集群上的主题的权限，相当	`kafka-cluster:Connect`	topic	是

操作	描述	所需的操作	所需的 资源	适用于无服务器集群
	于 Apache Kafka 的 DESCRIBE TOPIC ACL			

`kafka-cluster:AlterTopic`	授予更改集群上主题的权限，相当于 Apache Kafka 的 ALTER TOPIC ACL。	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeTopic`
			topic	是

`kafka-cluster:DeleteTopic`	授予删除集群上主题的权限，相当于 Apache Kafka 的 DELETE TOPIC ACL。	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeTopic`
			topic	是

`kafka-cluster:DescribeTopicDynamicConfiguration`	授予描述集群上的主题动态配置的权限，相当于 Apache Kafka 的 DESCRIBE_CONFIGS TOPIC ACL	`kafka-cluster:Connect`	topic	是

`kafka-cluster:AlterTopicDynamicConfiguration`	授予更改集群上主题的动态配置的权限，相当于 Apache Kafka 的 ALTER_CONFIGS TOPIC ACL	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeTopicDynamicConfiguration`
			topic	是

`kafka-cluster:ReadData`	授予从集群上的主题中读取数据的权限，相当于 Apache Kafka 的 READ TOPIC ACL。	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeTopic`
		
		`kafka-cluster:AlterGroup`
			topic	是

`kafka-cluster:WriteData`	授予向集群上的主题写入数据的权限，相当于 Apache Kafka 的 WRITE TOPIC ACL	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeTopic`
			topic	是

`kafka-cluster:DescribeGroup`	授予描述集群上的群组的权限，相当于 Apache Kafka 的 DESCRIBE GROUP ACL。	`kafka-cluster:Connect`	group	是

操作	描述	所需的操作	所需的 资源	适用于无服务器集群

`kafka-cluster:AlterGroup`	授予加入集群上群组的权限，相当于 Apache Kafka 的 READ GROUP ACL。	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeGroup`
			group	是

`kafka-cluster:DeleteGroup`	授予删除集群上的群组的权限，相当于 Apache Kafka 的 DELETE GROUP ACL。	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeGroup`
			group	是

`kafka-cluster:DescribeTransactionalId`	授予描述集群上的事务 ID 的权限，相当于 Apache Kafka 的 DESCRIBE TRANSACTIONAL_ID ACL	`kafka-cluster:Connect`	交易编号	是

`kafka-cluster:AlterTransactionalId`	授予更改集群上事务 ID 的权限，相当于 Apache Kafka 的 WRITE TRANSACTIONAL_ID ACL	`kafka-cluster:Connect`
		
		`kafka-cluster:DescribeTransactionalId`
		
		`kafka-cluster:WriteData`
			交易编号	是

您可以在冒号后面的操作中多次使用星号 (*) 通配符。示例如下。

- `kafka-cluster:*Topic`代表`kafka-cluster:CreateTopic`,`kafka-cluster:DescribeTopic`,`kafka-cluster:AlterTopic`，以及`kafka-cluster:DeleteTopic`. 它不包括`kafka-cluster:DescribeTopicDynamicConfiguration`要么`kafka-cluster:AlterTopicDynamicConfiguration`.
- `kafka-cluster:*`代表所有权限。

资源

下表显示了您在使用 Amazon MSK 的 IAM 访问控制时可以在授权策略中使用的四种类型的资源。您可以从中获取集群的 Amazon Resource (ARN)Amazon Web Services Management Console或者使用DescribeClusterAPI 或描述集群 Amazon CLI命令。然后，您可以使用集群 ARN 来构造主题、群组和事务 ID ARN。要在授权策略中指定资源，请使用该资源的 ARN。

资源	ARN 格式
Cluster	arn: aws: kafka:*##* : *## id* : cluster/*####*/*## uid*
主题	arn: aws: kafka:*##* : *## id* : topic/*####*/*## uid*/*####*
组	arn: aws: kafka:*##* : *## id* : group/*####*/*## uid*/*###*
事务 ID	arn: aws: kafka:*##* : *## id*:transactional-id/*####*/*## uid*/*####*

您可以在 ARN 之后的 ARN 部分中的任意位置多次使用星号 (*) 通配符`:cluster/`,`:topic/`,`:group/` ，以及`:transaction-id/`. 以下示例说明如何使用星号 (*) 通配符指向多个资源：

- `arn:aws:kafka:us-east-1:0123456789012:topic/MyTestCluster/*`: 名为的任意集群中的所有主题 MyTestCluster，不管集群的 UUID 如何。
- `arn:aws:kafka:us-east-1:0123456789012:topic/MyTestCluster/abcd1234-0123-abcd-5678-1234abcd-1/*_test`: 名称为 “_test” 的集群中名称以 “_test” 结尾的所有主题 MyTestCluster 而且谁的 UUID 是 abcd1234-0123-abcd-5678-1234abcd-1。
- `arn:aws:kafka:us-east-1:0123456789012:transactional-id/MyTestCluster/*/5555abcd-1111-abcd-1234-abcd1234-1`: 交易 ID 为 5555abcd-1111-abcd-1234-abcd1234-1 的所有交易，涵盖名为的集群的所有化身 MyTestCluster 在账户中。这意味着，如果你创建一个名为的集群 MyTestCluster，然后将其删除，然后使用相同名称创建另一个集群，您可以使用此资源 ARN 代表两个集群上的相同事务 ID。但是，已删除的集群无法访问。

常见用例

下表中的第一列显示了一些常见用例。要授权客户端执行给定用例，请在客户端的授权策略中包含该用例所需的操作，然后设置`Effect`到`Allow`.

有关属于 Amazon MSK 的 IAM 访问控制的所有操作的信息，请参阅the section called “动作和资源的语义” (p. 68).

> **Note**
>
> 默认情况下，操作将被拒绝。您必须明确允许您要授权客户端执行的每项操作。

使用案例	所需的操作
管理员	`kafka-cluster:*`
创建主题	`kafka-cluster:Connect` `kafka-cluster:CreateTopic`

生成数据	`kafka-cluster:Connect`
	`kafka-cluster:DescribeTopic`
	`kafka-cluster:WriteData`

消耗数据	`kafka-cluster:Connect`
	`kafka-cluster:DescribeTopic`
	`kafka-cluster:DescribeGroup`
	`kafka-cluster:AlterGroup`
	`kafka-cluster:ReadData`

以等效方式生成数据	`kafka-cluster:Connect`
	`kafka-cluster:DescribeTopic`
	`kafka-cluster:WriteData`
	`kafka-cluster:WriteDataIdempotently`

以交易方式生成数据	`kafka-cluster:Connect`

使用案例	所需的操作
	kafka-cluster:DescribeTopic kafka-cluster:WriteData kafka-cluster:DescribeTransactionalId kafka-cluster:AlterTransactionalId
描述集群的配置	kafka-cluster:Connect kafka-cluster:DescribeClusterDynamicConfiguration
更新集群的配置	kafka-cluster:Connect kafka-cluster:DescribeClusterDynamicConfiguration kafka-cluster:AlterClusterDynamicConfiguration
描述主题的配置	kafka-cluster:Connect kafka-cluster:DescribeTopicDynamicConfiguration
更新主题的配置	kafka-cluster:Connect kafka-cluster:DescribeTopicDynamicConfiguration kafka-cluster:AlterTopicDynamicConfiguration
更改主题	kafka-cluster:Connect kafka-cluster:DescribeTopic kafka-cluster:AlterTopic

双向 TLS 身份验证

您可以使用 TLS 为应用程序与 Amazon MSK 代理之间的连接启用客户端身份验证，以及 ZooKeeper 节点。要使用客户端身份验证，您需要一个 ACM 私有 CA。ACM 私有 CA 可以是相同的Amazon Web Services 账户作为集群或使用不同的账户。有关私有 CA 的信息，请参阅创建和管理私有 CA。

> **Note**
>
> TLS 身份验证目前不在北京和宁夏区域提供。

Amazon MSK 不支持证书吊销列表 (CRL)。要控制对集群主题的访问权限或屏蔽受损证书，请使用 Apache Kafka ACL 和Amazon安全组。有关使用 Apache Kafka ACL 的信息，请参阅the section called “Apache Kafka ACL” (p. 78).

本主题包含下列部分：

- 使用身份验证生成和使用消息 (p. 75)

创建支持客户端身份验证的集群

此过程向您展示如何使用由 ACM 托管的 CA 启用客户端身份验证。

Note

当您使用双向 TLS 控制访问时，我们强烈建议为每个 MSK 集群使用独立的 ACM PCA。这样做将确保由 PCA 签名的 TLS 证书仅对单个 MSK 集群进行身份验证。

1. 使用以下内容创建名为 `clientauthinfo.json` 的文件。将 *`Private-CA-ARN`* 替换为您的 PCA 的 ARN。

```
{
   "Tls": {
       "CertificateAuthorityArnList": ["Private-CA-ARN"]
    }
}
```

2. 创建一个名为 `brokernodegroupinfo.json` 的文件，如the section called “使用创建群集Amazon CLI” (p. 10)中所述。
3. 客户端身份验证还要求您启用客户端和代理之间的传输中加密。使用以下内容创建名为 `encryptioninfo.json` 的文件。将 *`KMS-Key-ARN`* 替换为您的 KMS 密钥的 ARN。可以将 `ClientBroker` 设置为 `TLS` 或 `TLS_PLAINTEXT`。

```
{
   "EncryptionAtRest": {
       "DataVolumeKMSKeyId": "KMS-Key-ARN"
    },
   "EncryptionInTransit": {
        "InCluster": true,
        "ClientBroker": "TLS"
    }
}
```

有关加密的更多信息，请参阅 the section called “加密” (p. 51)。

4. 在已安装 Amazon CLI 的计算机上，运行以下命令以创建启用了身份验证和传输加密的集群。保存响应中提供的集群 ARN。

```
aws kafka create-cluster --cluster-name "AuthenticationTest" --broker-node-group-info
 file://brokernodegroupinfo.json --encryption-info file://encryptioninfo.json --client-
authentication file://clientauthinfo.json --kafka-version "2.2.1" --number-of-broker-
nodes 3
```

将客户端设置为使用身份验证

1. 创建一个用作客户端计算机的 Amazon EC2 实例。为简单起见，请在用于集群的同一 VPC 中创建此实例。有关如何创建此类客户端计算机的示例，请参阅the section called “第 2 步：创建客户端计算机” (p. 5)。
2. 创建主题。有关示例，请参阅the section called “第 3 步：创建主题” (p. 6)下的说明。
3. 在已安装 Amazon CLI 的计算机上，运行以下命令以获取集群的引导代理。将 *`Cluster-ARN`* 替换为您的集群的 ARN。

```
aws kafka get-bootstrap-brokers --cluster-arn Cluster-ARN
```

保存与响应中的 BootstrapBrokerStringTls 关联的字符串。

4. 在客户端计算机上，运行以下命令以使用 JVM 信任存储来创建客户端信任存储。如果您的 JVM 路径不同，请相应地调整命令。

```
cp /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.201.b09-0.amzn2.x86_64/jre/lib/security/
cacerts kafka.client.truststore.jks
```

5. 在客户端计算机上，运行以下命令为客户端创建私有密钥。将 *Distinguished-Name*、*Example-Alias*、*Your-Store-Pass* 和 *Your-Key-Pass* 替换为所选字符串。

```
keytool -genkey -keystore kafka.client.keystore.jks -validity 300 -storepass Your-
Store-Pass -keypass Your-Key-Pass -dname "CN=Distinguished-Name" -alias Example-Alias -
storetype pkcs12
```

6. 在客户端计算机上，运行以下命令以使用您在上一步中创建的私有密钥创建证书请求。

```
keytool -keystore kafka.client.keystore.jks -certreq -file client-cert-sign-request -
alias Example-Alias -storepass Your-Store-Pass -keypass Your-Key-Pass
```

7. 打开 client-cert-sign-request 文件，并确保该文件的开头为 -----BEGIN CERTIFICATE REQUEST----- 且结尾为 -----END CERTIFICATE REQUEST-----。如果该文件的开头为 -----BEGIN NEW CERTIFICATE REQUEST-----，请从文件的开头和结尾处删除单词 NEW（及其后面的单个空格）。
8. 在已安装 Amazon CLI 的计算机上，运行以下命令以对证书请求进行签名。将 *Private-CA-ARN* 替换为您的 PCA 的 ARN。如果需要，您可以更改有效性值。在这里，我们以 300 为例。

```
aws acm-pca issue-certificate --certificate-authority-arn Private-CA-ARN --csr
 fileb://client-cert-sign-request --signing-algorithm "SHA256WITHRSA" --validity
 Value=300,Type="DAYS"
```

保存响应中提供的证书 ARN。

> **Note**
>
> 要检索您的客户端证书，请使用acm-pca get-certificate命令并指定您的证书 ARN。有关更多信息，请参阅 。gectificate在里面Amazon CLI命令参考.

9. 运行以下命令获取 ACM 为您签署的证书。将 *Certificate-ARN* 替换为您从上一条命令的响应中获取的 ARN。

```
aws acm-pca get-certificate --certificate-authority-arn Private-CA-ARN --certificate-
arn Certificate-ARN
```

10. 从运行上一条命令所获得的 JSON 结果中，复制与 Certificate 和 CertificateChain 关联的字符串。将这两个字符串粘贴到名为的新文件中 signed-certificate-from-acm. 先粘贴与 Certificate 关联的字符串，然后粘贴与 CertificateChain 关联的字符串。将 \n 字符替换为新行。以下是将证书和证书链粘贴到其中之后的文件结构。

```
-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----
```

11. 在客户端计算机上运行以下命令将此证书添加到您的密钥库中，以便能在与 MSK 代理交流时出示此证书。

```
keytool -keystore kafka.client.keystore.jks -import -file signed-certificate-from-acm -
alias Example-Alias -storepass Your-Store-Pass -keypass Your-Key-Pass
```

12. 使用以下内容创建名为 `client.properties` 的文件。将信任存储和密钥库位置调整为您将 `kafka.client.truststore.jks` 保存到的路径。

```
security.protocol=SSL
ssl.truststore.location=/tmp/kafka_2.12-2.2.1/kafka.client.truststore.jks
ssl.keystore.location=/tmp/kafka_2.12-2.2.1/kafka.client.keystore.jks
ssl.keystore.password=Your-Store-Pass
ssl.key.password=Your-Key-Pass
```

使用身份验证生成和使用消息

1. 运行以下命令以创建主题。

```
<path-to-your-kafka-installation>/bin/kafka-topics.sh --create --zookeeper ZooKeeper-
Connection-String --replication-factor 3 --partitions 1 --topic ExampleTopic
```

2. 运行以下命令以启动控制台生成器。名为 `client.properties` 的文件是您在上一过程中创建的文件。

```
<path-to-your-kafka-installation>/bin/kafka-console-producer.sh --broker-
list BootstrapBroker-String --topic ExampleTopic --producer.config client.properties
```

3. 在客户端计算机上的新命令窗口中，运行以下命令以启动控制台使用器。

```
<path-to-your-kafka-installation>/bin/kafka-console-consumer.sh --bootstrap-
server BootstrapBroker-String --topic ExampleTopic --consumer.config client.properties
```

4. 在生成器窗口中键入消息，并观察消息显示在使用器窗口中。

使用用户名和密码进行身份验证AmazonSecrets Manager

您可以使用来存储和保护用户名和密码，从而控制使用这些用户名和密码访问 Amazon MSK 集群的权限AmazonSecrets Manager。在 Secrets Manager 中存储用户凭证可以减少审计、更新和轮换凭证等群集身份验证的开销。Secrets Manager 还允许您在集群之间共享用户证书。

本主题包含下列部分：

工作原理

Amazon MSK 的用户名和密码身份验证使用 SASL/SCRAM（简单身份验证和安全层/Salted Challenge Response Authentication 要为集群设置用户名和密码身份验证，请在中创建密钥资源AmazonSecrets Manager，并将用户名和密码与该密钥相关联。

SASL/SCRAM 定义于RFC 5802. SCRAM 使用安全哈希算法，不会在客户端和服务器之间传输纯文本密码。

> **Note**
>
> 当您为集群设置 SASL/SCRAM 身份验证时，Amazon MSK 会为客户端和代理之间的所有流量开启 TLS 加密。

为亚马逊 MSK 集群设置 SASL/SCRAM 身份验证

在中设置密钥AmazonSecrets Manager，关注创建和检索密钥教程AmazonSecrets Manager 用户指南.

为 Amazon MSK 集群创建密钥时，请注意以下要求：

- 选择其他密钥类型（例如 API 密钥）对于密钥类型。
- 您的密钥名称必须以前缀开头amazonmsk_.
- 您必须使用现有的自定义Amazon KMS键入或创建新的自定义Amazon KMS你的秘密的钥匙。Secrets Manager 使用默认设置Amazon KMS默认情况下是密钥的密钥。

  > **Important**
  >
  > 使用默认值创建的密钥Amazon KMS密钥不能与 Amazon MSK 集群结合使用。
- 您的用户和密码数据必须采用以下格式才能使用键值对输入纯文本选项。

```
{
  "username": "alice",
  "password": "alice-secret"
}
```

- 记录您的密钥的 ARN（Amazon 资源名称）值。
- > **Important**
  >
  > 您不能将 Secrets Manager 密钥与超过中所述限制的集群相关联the section called “ 将集群设置为正确大小：每个代理的分区数” (p. 123).
- 如果您将Amazon CLI要创建密钥，请为指定密钥 ID 或 ARN`kms-key-id`参数。不要指定别名。
- 要将密钥与您的集群关联起来，请使用 Amazon MSK 控制台或 BatchAssociateScramSecret操作。

  > **Important**
  >
  > 当您将密钥与集群关联时，Amazon MSK 会向密钥附加资源策略，允许您的集群访问和读取您定义的密钥值。您不应修改此资源策略。这样做可以防止您的集群访问您的密钥。

  以下示例 JSON 输入`BatchAssociateScramSecret`操作将密钥与集群相关联：

```
{
  "clusterArn" : "arn:aws:kafka:us-west-2:0123456789019:cluster/SalesCluster/abcd1234-
abcd-cafe-abab-9876543210ab-4",
  "secretArnList": [
    "arn:aws:secretsmanager:us-west-2:0123456789019:secret:AmazonMSK_MyClusterSecret"
  ]
}
```

使用用户名和密码连接到集群

创建密钥并将其与集群相关联后，您可以将客户端连接到集群。以下示例步骤演示如何将客户端连接到使用 SASL/SCRAM 身份验证的集群，以及如何从示例主题中生成和使用。

1. 使用以下命令检索集群详细信息。Replace（替换）*ClusterArn*使用集群的 Amazon 资源名称 (ARN)：

```
aws kafka describe-cluster --cluster-arn "ClusterArn"
```

从命令的 JSON 结果中，保存与名为的字符串关联的值`ZookeeperConnectString`.

2. 要创建示例主题，请在您的客户端计算机上运行以下命令。Replace（替换）*ZookeeperConnectString*使用您在上一步中记录的字符串。

```
<path-to-your-kafka-installation>/bin/kafka-topics.sh --create --
zookeeper ZookeeperConnectString --replication-factor 3 --partitions 1 --
topic ExampleTopicName
```

3. 在您的客户端计算机上，创建一个 JAAS 配置文件，其中包含存储在您的密钥中的用户证书。例如，对于用户爱丽丝，创建一个名为`users_jaas.conf`具有以下内容。

```
KafkaClient {
   org.apache.kafka.common.security.scram.ScramLoginModule required
   username="alice"
   password="alice-secret";
};
```

4. 使用以下命令将您的 JAS 配置文件导出为`KAFKA_OPTS`环境参数。

```
export KAFKA_OPTS=-Djava.security.auth.login.config=<path-to-jaas-file>/users_jaas.conf
```

5. 创建一个名为的文件`kafka.client.truststore.jks`在一个`./tmp`目录。
6. 使用以下命令可从 JVM 复制 JDK 密钥存储文件`cacerts`文件夹转录`kafka.client.truststore.jks`您在上一步中创建的文件。Replace（替换）*jdk ## #*使用您的实例上的 JDK 文件夹的名称。例如，您的 JDK 文件夹可能被命名为`java-1.8.0-openjdk-1.8.0.201.b09-0.amzn2.x86_64`.

```
cp /usr/lib/jvm/JDKFolder/jre/lib/security/cacerts /tmp/kafka.client.truststore.jks
```

7. 在里面`bin`你的 Apache Kafka 安装目录，创建一个名为`client_sasl.properties`具有以下内容。此文件定义了 SASL 机制和协议。

```
security.protocol=SASL_SSL
sasl.mechanism=SCRAM-SHA-512
ssl.truststore.location=<path-to-keystore-file>/kafka.client.truststore.jks
```

8. 使用以下命令检索您的引导代理字符串。Replace（替换）*ClusterArn*使用集群的 Amazon 资源名称 (ARN)：

```
aws kafka get-bootstrap-brokers --cluster-arn ClusterArn
```

从命令的 JSON 结果中，保存与名为的字符串关联的值`BootstrapBrokerStringSaslScram`.

9. 要生成您创建的示例主题，请在您的客户端计算机上运行以下命令。Replace（替换）*BootstrapBrokerStringSaslScram*使用您在上一步中检索的值。

```
<path-to-your-kafka-installation>/bin/kafka-console-producer.sh --broker-
list BootstrapBrokerStringSaslScram --topic ExampleTopicName --producer.config
 client_sasl.properties
```

10. 要使用您创建的主题中的内容，请在您的客户端计算机上运行以下命令。Replace（替换）*BootstrapBrokerStringSaslScram*使用您之前获得的值。

```
<path-to-your-kafka-installation>/bin/kafka-console-consumer.sh --bootstrap-
server BootstrapBrokerStringSaslScram --topic ExampleTopicName --from-beginning --
consumer.config client_sasl.properties
```

使用用户

创建用户：您在密钥中创建用户。当您使用以下应用程序时：纯文本选项在 Secrets Manager 控制台中，应按以下格式指定用户名和密码数据。

```
{
  "username": "alice",
  "password": "alice-secret"
}
```

撤消用户访问权限：要撤消用户访问集群的证书，我们建议您首先删除或强制使用集群上的 ACL，然后取消关联密钥。这是因为以下原因：

- 删除用户不会关闭现有连接。
- 对您的密钥所做的更改最多需要 10 分钟才能传播。

有关结合 Amazon MSK 使用 ACL 的更多信息，请参阅Apache Kafka ACL (p. 78).

我们建议您限制对 zookeeper 节点的访问，以防止用户修改 ACL。有关更多信息，请参阅 控制对 Apache 的访问 ZooKeeper (p. 80)。

限制

使用 SCRAM 密钥时，请注意以下限制：

- Amazon MSK 仅支持 SCRAM-SHA-512 身份验证。
- 一个 Amazon MSK 集群最多可拥有 1000 个用户。
- 您必须使用Amazon KMS key用你的秘密。您不能将使用默认 Secrets Manager 加密密钥的密钥与 Amazon MSK 一起使用。有关创建 KMS 密钥的信息，请参阅创建对称加密 KMS 密钥.
- 您不能在 Secrets Managed 中使用非对称 KMS 密钥。
- 您可以一次将最多 10 个密钥与集群关联 BatchAssociateScramSecret操作。
- 与 Amazon MSK 集群关联的密钥的名称必须具有前缀amazonmsk_.
- 与 Amazon MSK 集群关联的密钥必须位于同一Amazon Web Services 账户中，并且Amazon区域作为集群。

Apache Kafka ACL

Apache Kafka 有一个可插拔的授权器，并附带了 out-of-box 使用 Apache 的授权器实现 ZooKeeper 存储所有 ACL。Amazon MSK 在代理上的 `server.properties` 文件中启用此授权方。对于 Apache Kafka 版本 2.4.1，授权方是 AclAuthorizer. 对于 Apache Kafka 的早期版本来说，确实如此 SimpleAclAuthorizer.

Apache Kafka ACL 的格式为 “主体 P 是 [允许/拒绝] 来自主机 H 对任何匹配的资源 R 的操作 O ResourcePattern RP”。如果 RP 与特定资源 R 不匹配，则 R 没有关联的 ACL，因此不允许除超级用户之外的用户访问 R。若要更改此 Apache Kafka 行为，请将属性 `allow.everyone.if.no.acl.found` 设为 true。默认情况下，Amazon MSK 会将其设置为 true。这意味着，对于 Amazon MSK 集群，如果您没有在资源上显式设置 ACL，则所有委托人都可以访问此资源。如果在资源上启用 ACL，则只有授权的委托人才能访问它。如果要限制对主题的访问并使用 TLS 相互身份验证授权客户端，请使用 Apache Kafka 授权方 CLI 添加 ACL。有关添加、删除和列出 ACL 的更多信息，请参阅 Kafka 授权命令行界面。

除客户端之外，您还需要授予所有代理访问主题的权限，以便代理可以从主分区复制消息。如果代理无权访问某个主题，则该主题的复制将失败。

添加或删除对主题的读写访问权

1. 将代理添加到 ACL 表中，以允许它们读取具有 ACL 的所有主题。要向您的代理授予对某个主题的读取权限，请在可以与 MSK 集群通信的客户端计算机上运行以下命令。

   Replace（替换）*ZooKeeper-#####*用您的 Apache ZooKeeper 连接字符串。有关如何获取此字符串的信息，请参阅the section called “获得 Apache ZooKeeper 连接字符串” (p. 13)。

   用任何集群引导代理的 DNS 替换 *Distinguished-Name*，然后用星号 (*) 替换此可分辨名称中第一个句点之前的字符串。例如，如果您的集群的引导代理之一具有 DNS `b-6.mytestcluster.67281x.c4.kafka.us-east-1.amazonaws.com`，请将以下命令中的 *Distinguished-Name* 替换为 `*.mytestcluster.67281x.c4.kafka.us-east-1.amazonaws.com`。有关如何获取引导代理的信息，请参阅the section called “获取引导代理” (p. 14)。

   ```
   <path-to-your-kafka-installation>/bin/kafka-acls.sh --authorizer-properties
    zookeeper.connect=ZooKeeper-Connection-String --add --allow-principal
    "User:CN=Distinguished-Name" --operation Read --group=* --topic Topic-Name
   ```

2. 要授予对主题的读访问权，请在客户端计算机上运行以下命令。如果您使用双向 TLS 身份验证，请使用相同的*####*你在创建私钥时使用的。

   ```
   <path-to-your-kafka-installation>/bin/kafka-acls.sh --authorizer-properties
    zookeeper.connect=ZooKeeper-Connection-String --add --allow-principal
    "User:CN=Distinguished-Name" --operation Read --group=* --topic Topic-Name
   ```

   要删除读访问权，您可以运行相同的命令，并将 `--add` 替换为 `--remove`。

3. 要授予对主题的写访问权，请在客户端计算机上运行以下命令。如果您使用双向 TLS 身份验证，请使用相同的*####*你在创建私钥时使用的。

   ```
   <path-to-your-kafka-installation>/bin/kafka-acls.sh --authorizer-properties
    zookeeper.connect=ZooKeeper-Connection-String --add --allow-principal
    "User:CN=Distinguished-Name" --operation Write --topic Topic-Name
   ```

   要删除写访问权，您可以运行相同的命令，并将 `--add` 替换为 `--remove`。

更改 Amazon MSK 集群的安全组

本页介绍如何更改现有 MSK 集群的安全组。您可能需要更改集群的安全组，以便为一组特定用户提供访问权限或限制对集群的访问权限。有关安全组的信息，请参阅您的 VPC 的安全组Amazon VPC 用户指南中。

1. 使用ListNodesAPI 或列表节点命令在Amazon CLI获取集群中的服务。此操作的结果包括与代理关联的弹性网络接口 (ENI) 的 ID。
2. 登录到 Amazon Web Services Management Console 并打开 Amazon EC2 控制台（https://console.aws.amazon.com/ec2/）。

3. 使用屏幕右上角附近的下拉列表，选择部署集群的区域。
4. 在左窗格中，在下面的网络与安全，选择网络接口.
5. 选择您在第一步中获得的第一个 ENI。选择操作屏幕顶部的菜单，然后选择更改安全组. 将新安全组分配给此 ENI。对您在第一步中获得的每个 ENI 重复此步骤。

   Note

   您使用 Amazon EC2 控制台对集群安全组所做的更改不会反映在下面的 MSK 控制台中网络设置.
6. 配置新安全组的规则，确保您的客户端可以访问代理。有关设置安全组规则的信息，请参阅添加、删除和更新规则Amazon VPC 用户指南中。

   Important

   如果您更改与集群代理关联的安全组，然后向该集群添加新代理，则 Amazon MSK 会将新代理与创建集群时与集群关联的原始安全组相关联。但是，要使集群正常运行，其所有代理必须与同一个安全组相关联。因此，如果您在更改安全组后添加了新的经纪人，则必须再次执行之前的步骤并更新新经纪人的ENI。

控制对 Apache 的访问 ZooKeeper

出于安全考虑，您可以限制对 Apache 的访问 ZooKeeper 属于您的亚马逊 MSK 集群的节点。要限制对节点的访问，您可以为节点分配单独的安全组。然后，您可以决定有权访问该安全组的人员。

本主题包含下列部分：

放置你的 Apache ZooKeeper 单独安全组中的节点

1. 获取 Apache ZooKeeper 集群的连接字符串。要了解如何操作，请参阅the section called “获得 Apache ZooKeeper 连接字符串” (p. 13)。连接字符串包含您的 Apache 的 DNS 名称 ZooKeeper 节点。
2. 使用 `host` 或 `ping` 等工具将您在上一步中获得的 DNS 名称转换为 IP 地址。稍后您需要在此过程中使用这些 IP 地址，因此请保存这些地址。
3. 登录到 Amazon Web Services Management Console 并打开 Amazon EC2 控制台（ https://console.aws.amazon.com/ec2/ ）。
4. 在左侧窗格的 Network & Security (网络与安全性) 下，选择 Network Interfaces (网络接口)。
5. 在网络接口表上方的搜索字段中，键入集群名称，然后键入 return。这会将表中显示的网络接口数限制为与您的集群关联的接口。
6. 选中与列表中的第一个网络接口对应的行开头处的复选框。
7. 在页面底部的详细信息窗格中，查找 Primary private IPv4 IP (主要私有 IPv4 IP)。如果此 IP 地址与您在此过程的第一步中获得的 IP 地址之一相匹配，则表示该网络接口已分配给 Apache ZooKeeper 属于您的集群的节点。否则，取消选中此网络接口旁边的复选框，然后选择列表中的下一个网络接口。选择网络接口的顺序无关紧要。在接下来的步骤中，您将在分配给 Apache 的所有网络接口上执行相同的操作 ZooKeeper 节点，一个接一个。
8. 当您选择与 Apache 对应的网络接口时 ZooKeeper 节点，选择操作页面顶部的菜单，然后选择更改安全组. 将新安全组分配给此网络接口。有关创建安全组的信息，请参阅创建安全组Amazon VPC 文档中，请查看
9. 重复前面的步骤，为与 Apache 关联的所有网络接口分配相同的新安全组 ZooKeeper 集群的节点。
10. 现在，您可以选择有权访问此新安全组的人员。有关设置安全组规则的信息，请参阅添加、删除和更新规则Amazon VPC 文档中，请查看

在 Apache 中使用 TLS 安全性 ZooKeeper

在客户端和 Apache 之间的传输过程中，您可以使用 TLS 安全进行加密 ZooKeeper 节点。使用你的 Apache 实现 TLS 安全性 ZooKeeper 节点，执行以下操作：

- 集群必须使用 Apache Kafka 版本 2.5.1 或更高版本才能在 Apache 中使用 TLS 安全性 ZooKeeper.
- 在创建或配置集群时启用 TLS 安全。使用 Apache Kafka 版本 2.5.1 或更高版本创建且启用了 TLS 的集群会自动使用 Apache 的 TLS 安全性 ZooKeeper 终端节点。有关设置 TLS 安全的信息，请参阅如何开始使用加密？ (p. 51).
- 检索 TLS Apache ZooKeeper 端点使用DescribeCluster 操作。
- 创建一个 Apache ZooKeeper 与... 一起使用的配置文件`kafka-configs.sh`和`kafka-acls.sh`工具，或者使用 ZooKeeper Shell。对于每种工具，您都使用`--zk-tls-config-file`用于指定你的 Apache 的参数 ZooKeeper COMig。

  以下示例显示了典型的 Apache。 ZooKeeper 配置文件：

  ```
  zookeeper.ssl.client.enable=true
  zookeeper.clientCnxnSocket=org.apache.zookeeper.ClientCnxnSocketNetty
  zookeeper.ssl.keystore.location=kafka.jks
  zookeeper.ssl.keystore.password=test1234
  zookeeper.ssl.truststore.location=truststore.jks
  zookeeper.ssl.truststore.password=test1234
  ```

- 用于其他命令（例如`kafka-topics`），您必须使用`KAFKA_OPTS`用于配置 Apache 的环境变量 ZooKeeper参数。以下示例说明了如何配置`KAFKA_OPTS`传递 Apache 的环境变量 ZooKeeper将参数转换为其他命令：

  ```
  export KAFKA_OPTS="
  -Dzookeeper.clientCnxnSocket=org.apache.zookeeper.ClientCnxnSocketNetty
  -Dzookeeper.client.secure=true
  -Dzookeeper.ssl.trustStore.location=/home/ec2-user/kafka.client.truststore.jks
  -Dzookeeper.ssl.trustStore.password=changeit"
  ```

  在您配置之后`KAFKA_OPTS`环境变量，你可以正常使用 CLI 命令。以下示例使用 Apache 创建 Apache Kafka 主题 ZooKeeper 配置来自`KAFKA_OPTS`环境变量：

  ```
  <path-to-your-kafka-installation>/bin/kafka-topics.sh --create --
  zookeeper ZooKeeperTLSConnectString --replication-factor 3 --partitions 1 --topic
   AWSKafkaTutorialTopic
  ```

  Note

  你在 Apache 中使用的参数的名称 ZooKeeper 配置文件和你在你的`KAFKA_OPTS`环境变量不一致。注意你在配置文件中使用了哪些名称和哪些参数`KAFKA_OPTS`环境变量。

有关访问您的 Apache 的更多信息 ZooKeeper 带有 TLS 的节点，请参阅KIP-515：允许 ZK 客户端使用新的 TLS 支持身份验证.

日志记录

您可以将 Apache Kafka 代理日志传送到以下一种或多种目标类型：亚马逊 CloudWatch 日志、Amazon S3、Amazon Kinesis Data Firehose 您也可以使用以下方式记录亚马逊 MSK API 调用Amazon CloudTrail.

代理日志

代理日志使您能够对 Apache Kafka 应用程序进行故障排除，并分析它们与 MSK 集群的通信。您可以配置新的或现有的 MSK 集群，将信息级别的代理日志传送到以下一种或多种类型的目标资源：a CloudWatch 日志组、S3 存储桶、Kinesis Data Firehose 传输流。然后，您可以通过 Kinesis Data Firehose 将日志数据从传输流传输到 OpenSearch 服务 。必须先创建目标资源，然后才能将群集配置为向其传送代理日志。Amazon MSK 不会为您创建这些目标资源。有关这三种类型的目标资源以及如何创建这些资源的信息，请参阅以下文档：

- 亚马逊 CloudWatch 日志
- Amazon S3
- Amazon Kinesis Data Firehose

> **Note**
>
> Amazon MSK 不支持将代理日志传输到亚太（大阪）区域的 Kinesis Data Firehose。

所需权限

要为 Amazon MSK 代理日志配置目的地，您用于 Amazon MSK 操作的 IAM 身份必须具有中所述的权限Amazon托管策略：AmazonMSKFullAccess (p. 61)策略。

要将代理日志流式传输到 S3 存储桶，您还需要`s3:PutBucketPolicy`权限。有关 S3 存储桶策略的信息，请参阅如何添加 S3 存储桶策略？Amazon S3 控制台用户指南中《Amazon S 有关 IAM policy 的一般信息，请参阅访问管理IAM 用户指南中。

与 SSE-KMS 存储桶结合使用时必需的 KMS 密钥策略

如果您使用为 S3 存储桶启用了服务器端加密Amazon KMS-托管密钥 (SSE-KMS) 使用客户托管密钥，将以下内容添加到 KMS 密钥的密钥策略中，以便Amazon MSK 可以将代理文件写入存储桶。

```
{
  "Sid": "Allow Amazon MSK to use the key.",
  "Effect": "Allow",
  "Principal": {
    "Service": [
      "delivery.logs.amazonaws.com"
    ]
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

使用 Amazon Web Services Management Console配置代理日志

如果您正在创建新集群，请查找代理日志传送正在前进监控部分。您可以指定您希望 Amazon MSK 将您的代理日志传送到的目的地。

对于现有集群，从集群列表中选择集群，然后选择属性选项卡。向下滚动到日志传输部分，然后选择它的编辑按钮。您可以指定您希望 Amazon MSK 将您的代理日志传送到的目的地。

使用 Amazon CLI配置代理日志

使用 `create-cluster` 或 `update-monitoring` 命令时，您可以选择指定 `logging-info` 参数并将类似如下的 JSON 结构传递给该参数。在此 JSON 中，所有三种目标类型都是可选的。

```
{
  "BrokerLogs": {
    "S3": {
      "Bucket": "ExampleBucketName",
      "Prefix": "ExamplePrefix",
      "Enabled": true
    },
    "Firehose": {
      "DeliveryStream": "ExampleDeliveryStreamName",
      "Enabled": true
    },
    "CloudWatchLogs": {
      "Enabled": true,
      "LogGroup": "ExampleLogGroupName"
    }
  }
}
```

使用 API 配置代理日志

可以指定可选的`loggingInfo`你传递给的 JSON 中的结构CreateCluster要么UpdateMonitoring操作。

> **Note**
>
> 默认情况下，启用代理日志记录时，Amazon MSK 会日志`INFO`到指定目的地的级别日志。但是，Apache Kafka 2.4.X 及更高版本的用户可以动态地将代理日志级别设置为任何log4j 日志级别. 有关动态设置代理日志级别的信息，请参阅KIP-412：扩展管理员 API 以支持动态应用程序日志级别. 如果您将日志级别动态设置为`DEBUG`要么`TRACE`，我们建议使用 Amazon S3 或 Kinesis Data Firehose 作为日志目标。如果您使用 CloudWatch 将日志作为日志目的地，您可以动态启用`DEBUG`要么`TRACE`级别日志，Amazon MSK 可能会持续提供日志样本。这可能会显著影响经纪商的表现，只能在经纪人业绩时使用`INFO`日志级别不够详细，不足以确定问题的根本原因。

使用 Amazon CloudTrail 记录 API 调用

> **Note**
>
> Amazon CloudTrail只有当您使用时，日志才可用于亚马逊 MSKIAM 访问控制 (p. 65).

亚马逊 MSK 已集成到Amazon CloudTrail，一种提供用户、角色或角色所采取操作的记录的服务Amazon亚马逊 MSK 中的服务。 CloudTrail 将的 API 调用作为事件捕获 API 调用。这些捕获包括通过 Amazon MSK 控制台的调用和对 Amazon MSK API 操作的代码调用。它还捕获 Apache Kafka 的操作，例如创建和修改主题和群组。

如果您创建了跟踪，则可以使持续传输 CloudTrail Amazon S3 存储桶的事件，包括Amazon MSK 的事件。如果您不配置跟踪，则仍可在 CloudTrail 控制台输入事件历史. 使用收集的信息 CloudTrail，您可以确定向 Amazon MSK 发出了什么请求、何人发出的请求、请求的发出时间以及其他详细信息。

了解相关更多信息 CloudTrail，包括如何对其进行配置和启用，请参阅Amazon CloudTrail用户指南.

Amazon MSK 信息位于 CloudTrail

CloudTrail 在您创建Amazon Web Services 账户时，将在该账户上启用。当 MSK 集群中发生受支持的事件活动时，该活动将记录在 CloudTrail 活动以及其他Amazon中的服务事件事件历史. 您可以在 Amazon Web

Services 账户中查看、搜索和下载最新事件。有关更多信息，请参阅使用 CloudTrail 事件历史记录查看事件。

要持续记录 Amazon Web Services 账户中的事件（包括 Amazon MSK 的事件），请创建跟踪记录。一个踪迹启用 CloudTrail 将日志文件传送至 Amazon S3 存储桶。默认情况下，在控制台中创建跟踪时，此跟踪应用于所有区域。此跟踪记录在 Amazon 分区中记录所有区域中的事件，并将日志文件传送至您指定的 Simple Storage Service（Amazon S3）存储桶。此外，您可以配置其他 Amazon 服务，进一步分析在中收集的事件数据并采取操作 CloudTrail 日志。有关更多信息，请参阅下列内容：

- 创建跟踪记录概述
- CloudTrail 支持的服务和集成
- 为 配置 Amazon SNS 通知 CloudTrail
- 接收 CloudTrail 多个区域的日志文件和接收 CloudTrail 多个账户中的日志文件

AmazonAmazon作为事件在 CloudTrail 日志文件。此外，它还会记录以下 Apache Kafka 操作。

- kafka 集群：DescribeClusterDynamicConfiguration
- kafka 集群：AlterClusterDynamicConfiguration
- kafka 集群：CreateTopic
- kafka 集群：DescribeTopicDynamicConfiguration
- kafka 集群：AlterTopic
- kafka 集群：AlterTopicDynamicConfiguration
- kafka 集群：DeleteTopic

每个事件或日志条目都包含有关生成请求的人员信息。身份信息可帮助您确定以下内容：

- 请求是使用根用户凭证还是 Amazon Identity and Access Management（IAM）用户凭证发出的。
- 请求是使用角色还是联合身份用户的临时安全凭证发出的。
- 请求是否由其它 Amazon 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

示例：Amazon MSK 日志文件条目

跟踪是一种配置，可用于将事件作为日志文件传送到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。一个事件表示来自任何源的一个请求，包括有关所请求的操作、操作的日期和时间、请求参数等方面的信息。 CloudTrail 日志文件不是公用 API 调用和 Apache Kafka 操作的有序堆栈跟踪，因此它们不会按任何特定顺序显示。

下面的示例演示如何操作 CloudTrail 演示以下内容的日志条目`DescribeCluster`和`DeleteCluster`Amazon MSK 操作。

```
{
  "Records": [
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEF0123456789ABCDE",
        "arn": "arn:aws:iam::012345678901:user/Joe",
        "accountId": "012345678901",
        "accessKeyId": "AIDACKCEVSQ6C2EXAMPLE",
        "userName": "Joe"
      },
```

```
      "eventTime": "2018-12-12T02:29:24Z",
      "eventSource": "kafka.amazonaws.com",
      "eventName": "DescribeCluster",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "aws-cli/1.14.67 Python/3.6.0 Windows/10 botocore/1.9.20",
      "requestParameters": {
        "clusterArn": "arn%3Aaws%3Akafka%3Aus-east-1%3A012345678901%3Acluster
%2Fexamplecluster%2F01234567-abcd-0123-abcd-abcd0123efa-2"
      },
      "responseElements": null,
      "requestID": "bd83f636-fdb5-abcd-0123-157e2fbf2bde",
      "eventID": "60052aba-0123-4511-bcde-3e18dbd42aa4",
      "readOnly": true,
      "eventType": "AwsApiCall",
      "recipientAccountId": "012345678901"
    },
    {
      "eventVersion": "1.05",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEF0123456789ABCDE",
        "arn": "arn:aws:iam::012345678901:user/Joe",
        "accountId": "012345678901",
        "accessKeyId": "AIDACKCEVSQ6C2EXAMPLE",
        "userName": "Joe"
      },
      "eventTime": "2018-12-12T02:29:40Z",
      "eventSource": "kafka.amazonaws.com",
      "eventName": "DeleteCluster",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "aws-cli/1.14.67 Python/3.6.0 Windows/10 botocore/1.9.20",
      "requestParameters": {
        "clusterArn": "arn%3Aaws%3Akafka%3Aus-east-1%3A012345678901%3Acluster
%2Fexamplecluster%2F01234567-abcd-0123-abcd-abcd0123efa-2"
      },
      "responseElements": {
        "clusterArn": "arn:aws:kafka:us-east-1:012345678901:cluster/
examplecluster/01234567-abcd-0123-abcd-abcd0123efa-2",
        "state": "DELETING"
      },
      "requestID": "c6bfb3f7-abcd-0123-afa5-293519897703",
      "eventID": "8a7f1fcf-0123-abcd-9bdb-1ebf0663a75c",
      "readOnly": false,
      "eventType": "AwsApiCall",
      "recipientAccountId": "012345678901"
    }
  ]
}
```

下面的示例介绍一个 CloudTrail 演示以下内容的日志条目`kafka-cluster:CreateTopic`操作。

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "ABCDEFGH1IJKLMN2P34Q5",
    "arn": "arn:aws:iam::111122223333:user/Admin",
    "accountId": "111122223333",
    "accessKeyId": "CDEFAB1C2UUUUU3AB4TT",
    "userName": "Admin"
  },
  "eventTime": "2021-03-01T12:51:19Z",
  "eventSource": "kafka-cluster.amazonaws.com",
```

```
  "eventName": "CreateTopic",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "198.51.100.0/24",
  "userAgent": "aws-msk-iam-auth/unknown-version/aws-internal/3 aws-sdk-java/1.11.970
 Linux/4.14.214-160.339.amzn2.x86_64 OpenJDK_64-Bit_Server_VM/25.272-b10 java/1.8.0_272
 scala/2.12.8 vendor/Red_Hat,_Inc.",
  "requestParameters": {
    "kafkaAPI": "CreateTopics",
    "resourceARN": "arn:aws:kafka:us-east-1:111122223333:topic/IamAuthCluster/3ebafd8e-
dae9-440d-85db-4ef52679674d-1/Topic9"
  },
  "responseElements": null,
  "requestID": "e7c5e49f-6aac-4c9a-a1d1-c2c46599f5e4",
  "eventID": "be1f93fd-4f14-4634-ab02-b5a79cb833d2",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333"
}
```


Amazon Managed Managed Streaming for Apache Kafka

作为其中的一部分，第三方审计员将评估 Amazon Managed fManaged Streaming for Apache Kafka 的 Amazon ManagedAmazon合规性计划。其中包括 PCI 和 HIPAA BAA。

有关以下内容的清单Amazon特定合规性计划范围内的服务，请参阅合规性计划计划计划计划计划计划计划计划计划计划计划计划范围内的 Amazon Web Services. 有关常规信息，请参阅Amazon合规性计划。

您可以使用 Amazon Artifact 下载第三方审计报告。有关更多信息，请参阅下载 Amazon Artifact 中的报告。

您在使用 Amazon MSK 时的合规性责任由您的数据的敏感性、您公司的合规性目标以及适用的法律法规决定。Amazon提供以下资源来帮助实现合规性：

- 安全性与合规性 Quick Start 指南安全性与合规性 Quick Start 指南 - 这些部署指南讨论了架构注意事项，并提供了在 Amazon 上部署基于安全性和合规性的基准环境的步骤。
- 《设计符合 HIPAA 安全性和合规性要求的架构》白皮书 – 此白皮书介绍公司如何使用Amazon创建符合 HIPAA 标准的应用程序。
- Amazon合规性资源 – 此业务手册和指南集合可能适用于您的行业和位置。
- Amazon Config 开发人员指南中的使用规则评估资源 – 此 Amazon Config 服务评估您的资源配置对内部实践、行业指南和法规的遵循情况。
- Amazon Security Hub – 此 Amazon 服务提供了 Amazon 中安全状态的全面视图，可帮助您检查是否符合安全行业标准和最佳实践。

Amazon MManaged Streaming for Apache Kafka

Amazon全球基础设施围绕Amazon区域和可用区构建。Amazon区域提供多个在物理上独立且隔离的可用区，这些可用区通过延迟低、吞吐量高且冗余性高的网络连接在一起。利用可用区，您可以设计和操作在可用区之间无中断地自动实现故障转移的应用程序和数据库。与传统的单个或多个数据中心基础设施相比，可用区具有更高的可用性、容错性和可扩展性。

有关 Amazon 区域和可用区的更多信息，请参阅 Amazon 全球基础设施。

Amazon Managed Streaming for Apache Kafka 中的基础设施安全

作为一项托管服务，Amazon Managed Streaming for Apache Kafka 受到Amazon全局网络安全程序，如中所述Amazon Web Services：安全过程概述白皮书。

你用Amazon发布的 API 调用通过网络访问 Amazon MSK。客户端必须支持传输层安全性 (TLS) 1.0 或更高版本。建议使用 TLS 1.2 或更高版本。客户端还必须支持具有完全向前保密 (PFS) 的密码套件，例如 Ephemeral Diffie-Hellman (DHE) 或 Elliptic Curve Ephemeral Diffie-Hellman (ECDHE)。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 委托人关联的秘密访问密钥来对请求进行签名。或者，您可以使用 Amazon Security Token Service (Amazon STS) 生成临时安全凭证来对请求进行签名。

连接到一个 Amazon MSK 集群

默认情况下，客户端只有在与集群位于同一 VPC 中时，才能访问该集群。要从与位于同一 VPC 中的客户端连接到该集群，请确保集群的安全组具有接受来自客户端安全组的流量的入站规则。有关设置这些规则的信息，请参阅安全组规则。有关如何从与位于同一 VPC 中的 Amazon EC2 实例访问该集群的示例，请参阅开始使用 (p. 5).

要从群集 VPC 外部的客户端连接到 MSK 集群，请参阅以下主题：

主题

公有访问权限

通过 Amazon MSK，您可以选择对运行 Apache Kafka 2.6.0 或更高版本的 MSK 集群的代理开启公共访问权限。出于安全原因，您无法在创建 MSK 集群时启用公有访问。但是，您可以更新现有集群以使其可公开访问。您还可以创建新集群，然后对其进行更新以使其可公开访问。

您可以开启对 MSK 群集的公共访问权限，无需额外费用，但标准配置Amazon数据传输费用适用于进出集群的数据传输。有关定价的信息，请参阅Amazon EC2 按需定价.

要启用集群的公共访问权限，请首先确保集群满足以下所有条件：

- 与群集关联的子网必须是公有的。这意味着子网必须有一个连接了 Internet 网关的关联路由表。有关如何创建和连接 Internet 网关的信息，请参阅Internet 网关（在 Amazon VPC 用户指南中）。
- 必须关闭未经身份验证的访问控制，并且必须至少打开以下访问控制方法之一：SASL/IAM、SASL/SCRAM、mtls。有关如何更新集群的访问控制方法的信息，请参阅the section called “更新安全性” (p. 25).
- 必须开启群集内的加密。on 设置是创建集群时的默认设置。对于在关闭加密功能的情况下创建的集群，无法在集群内启用加密。因此，无法为在群集内使用加密功能创建的群集开启公共访问权限。
- 经纪人和客户之间的明文流量必须关闭。有关如何在开启时将其关闭的信息，请参阅the section called “更新安全性” (p. 25).
- 如果您使用的是 SASL/SCRAM 或 mTLS 访问控制方法，则必须为集群设置 Apache Kafka ACL。为集群设置 Apache Kafka ACL 后，请更新集群的配置以使用属性`allow.everyone.if.no.acl.found`集群的值为 false。有关如何更新集群配置的信息，请参阅the section called “配置操作” (p. 37). 如果您正在使用 IAM 访问控制，并且想要应用授权策略或更新授权策略，请参阅the section called “IAM 访问控制” (p. 65). 有关 Apache Kafka ACL 的信息，请参阅the section called “Apache Kafka ACL” (p. 78).

确保 MSK 群集满足上面列出的条件后，您可以使用Amazon Web Services Management Console，Amazon CLI，或者使用亚马逊 MSK API 来开启公共访问权限。开启对集群的公共访问后，您可以为其获取公共引导代理字符串。有关获取集群的引导代理的信息，请参阅the section called “获取引导代理” (p. 14).

> **Important**
>
> 除了启用公共访问外，还要确保集群的安全组具有允许从您的 IP 地址进行公共访问的入站 TCP 规则。因此，建议您使这些规则尽可能具有限制性。有关安全组和入站规则的信息，请参阅您的 VPC 的安全组（在 Amazon VPC 用户指南 中）。有关端口号，请参阅the section called “端口信息” (p. 91). 有关如何更改集群的安全组的说明，请参阅the section called “更改安全组” (p. 79).

Note

如果您使用以下说明开启公有访问权限，但仍然无法访问集群，请参阅the section called “无法访问已开启公共访问权限的群集” (p. 120).

使用控制台启用公有访问

1. 登录到Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home?region=us-east-1#/home/.
2. 在集群列表中，选择要启用公有访问的集群。
3. 选择属性选项卡，然后找到Network settings (网络设置)部分。
4. 选择编辑公有访问.

使用启用公有访问Amazon CLI

1. 运行以下命令Amazon CLI命令，替换*ClusterArn*和*######*的 ARN 和集群的当前版本。要查找集群的当前版本，请使用DescribeClusteroperation或describe- Amazon CLI命令。示例版本是KTVPDKIKX0DER。

```
aws kafka update-connectivity --cluster-arn ClusterArn --current-
version Current-Cluster-Version --connectivity-info '{"PublicAccess": {"Type":
 "SERVICE_PROVIDED_EIPS"}}'
```

该 update-connectivity 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
    "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef"
}
```

Note

要关闭公共访问权限，请使用类似Amazon CLI命令，但改为使用以下连接信息：

```
'{"PublicAccess": {"Type": "DISABLED"}}'
```

2. 要得出结果update-connectivityoperation，运行以下命令，替换*ClusterOperationArn*使用您在输出中获得的 ARNupdate-connectivity命令。

```
aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
```

该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterOperationInfo": {
        "ClientRequestId": "982168a3-939f-11e9-8a62-538df00285db",
        "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
        "CreationTime": "2019-06-20T21:08:57.735Z",
        "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef",
        "OperationState": "UPDATE_COMPLETE",
```

```
        "OperationType": "UPDATE_CONNECTIVITY",
        "SourceClusterInfo": {
            "ConnectivityInfo": {
                "PublicAccess": {
                    "Type": "DISABLED"
                }
            }
        },
        "TargetClusterInfo": {
            "ConnectivityInfo": {
                "PublicAccess": {
                    "Type": "SERVICE_PROVIDED_EIPS"
                }
            }
        }
    }
}
```

如果 `OperationState` 的值为 `UPDATE_IN_PROGRESS`，请等待一段时间，然后再次运行 `describe-cluster-operation` 命令。

使用亚马逊 MSK API 开启公共访问权限

- 要使用 API 打开或关闭集群的公共访问权限，请参阅UpdateConnectivity.

  Note

  出于安全原因，亚马逊 MSK 不允许公开访问 Apache ZooKeeper 节点。有关如何控制对 Apache 的访问的信息 ZooKeeper 来自内部的 MSK 集群的节点Amazon，请参阅the section called “控制对 Apache 的访问 ZooKeeper” (p. 80).

从内部进入Amazon但在集群的 VPC 之外

从内部连接到 MSK 集群Amazon但在集群的 Amazon VPC 之外，存在以下几种方式。

Amazon VPC 等连接

要从不同于所在的 VPC 连接到该集群，您可以在这两个 VPC 之间建立对等连接。有关 VPC 对等连接的信息，请参阅 Amazon VPC 对等连接指南。

Amazon Direct Connect

Amazon Direct Connect将您的本地网络链接到Amazon通过标准的 1 Gb 或 10 Gb 以太网光纤电缆进行连接。电缆的一端接到您的路由器，另一端接到 Amazon Direct Connect 路由器。有了此连接后，您就可以创建直接连接到Amazoncloud 和 Amazon VPC，从而绕过您的网络路径中的互联网服务提供商。有关更多信息，请参阅 Amazon Direct Connect。

Amazon Transit Gateway

Amazon Transit Gateway 是一项服务，通过此服务，您可以将您的 VPC 和本地网络连接到单个网关。有关如何使用 Amazon Transit Gateway 的信息，请参阅Amazon Transit Gateway。

VPN 连接

您可以使用以下主题中介绍的 VPN 选项将 MSK 集群的 VPC 连接到远程网络和用户：VPN 连接.

REST 代理

您可以在集群的 Amazon VPC 中运行的实例上安装 REST 代理。利用 REST 代理，创建器和使用器将能够通过 HTTP API 请求与集群通信。

多区域多 VPC 连接

以下文档介绍了位于不同区域的多个 VPC 的连接选项：多区域多 VPC 连接.

EC2-Classic

按照以下过程从 EC2-Classic 实例连接到您的集群。

1. 按照中的指导进行操作ClassicLink，将 EC2-Classic 实例连接到集群的 VPC。
2. 查找并复制与您的 EC2-Classic 实例关联的私有 IP。
3. 使用Amazon CLI，运行以下命令，替换*ClusterArn*，以及您的 MSK 集群的 Amazon 资源名称 (ARN)。

```
aws kafka describe-cluster --cluster-arn "ClusterArn"
```

4. 在的输出中`describe-cluster`命令，寻找`SecurityGroups`，然后保存 MSK 集群的安全组 ID。
5. 通过以下网址打开 Amazon VPC 控制台：https://console.aws.amazon.com/vpc/。
6. 在左侧窗格中，选择 Security Groups (安全组)。
7. 选择您在运行 `describe-cluster` 命令后保存其 ID 的安全组。选中与此安全组对应的行开头的框。
8. 在页面的下半部分中，选择 Inbound Rules (入站规则)。
9. 选择 Edit rules (编辑规则)，然后选择 Add Rule (添加规则)。
10. 对于 Type (类型) 字段，选择下拉列表中的 All traffic (所有流量)。
11. 将 Source (源) 设置为 Custom (自定义)，然后输入 EC2-Classic 实例的私有 ID，后紧跟 **`/32`**，无中间空格。
12. 选择 Save rules（保存规则）。

端口信息

以下列表提供了 Amazon MSK 用于与客户端计算机进行通信的端口号。

- 要以明文形式与代理进行通信，请使用端口 9092。
- 要使用 TLS 加密与经纪商通信，请使用端口 9094 从内部进行访问Amazon和用于公共访问的端口 9194。
- 要使用SASL/SCRAM与经纪商进行通信，请使用端口为9096进行内部访问Amazon和用于公共访问的端口 9196。
- 与设置为使用的集群中的代理进行通信the section called “IAM 访问控制” (p. 65)，使用端口 9098 从内部访问Amazon和用于公共访问的端口 9198。
- 阿帕奇 ZooKeeper 节点默认使用端口 2181。与 Apache 通信 ZooKeeper 通过使用 TLS 加密, 请使用端口 2182。

使用 Apache Kafka 迁移集群 MirrorMaker

您可以使用镜像或迁移集群 MirrorMaker，它是 Apache Kafka 的一部分。例如，可以使用它将 Apache Kafka 集群迁移到 Amazon MSK 或从一个 MSK 集群迁移到另一个 MSK 集群。有关如何使用的信息 MirrorMaker，请参阅在集群之间镜像数据在 Apache Kafka 文档中。我们建议您设置 MirrorMaker 在高可用性配置中。

使用时要执行的步骤的概述 MirrorMaker 要迁移到 MSK 集群

1. 创建目标 MSK 集群
2. 启动 MirrorMaker 从目标集群所在的同一个 Amazon VPC 中的 Amazon EC2 实例进行的。
3. 检查 MirrorMaker 滞后。
4. 晚于 MirrorMaker 使用 MSK 集群引导代理将创建器和使用器重定向到新的集群。
5. 关闭 MirrorMaker.

将您的 Apache Kafka 集群迁移到亚马逊 MSK

假定您有一个名为 CLUSTER_ONPREM 的 Apache Kafka 集群。该集群中已填充主题和数据。如果您想将该集群迁移到新创建的名为的 Amazon MSK 集群CLUSTER_AWSMSK，此过程提供您需要执行的步骤的高级视图。

将现有的 Apache Kafka 集群迁移到 Amazon MSK

1. 在 CLUSTER_AWSMSK 中，创建要迁移的所有主题。

   你不能使用 MirrorMaker 因为此步骤不会自动使用正确的复制级别重新创建要迁移的主题。您可以在 Amazon MSK 中创建具有与中相同的复制因子和分区数的主题CLUSTER_ONPREM. 也可以创建具有不同的复制因子和分区数的主题。
2. 启动 MirrorMaker 从具有有读取访问权的实例CLUSTER_ONPREM和的写入访问权限CLUSTER_AWSMSK.
3. 运行以下命令以镜像所有主题：

   ```
   <path-to-your-kafka-installation>/bin/kafka-mirror-maker.sh --consumer.config
    config/mirrormaker-consumer.properties --producer.config config/mirrormaker-
   producer.properties --whitelist '.*'
   ```

   在此命令中，config/mirrormaker-consumer.properties 指向 CLUSTER_ONPREM 中的引导代理；例如，bootstrap.servers=localhost:9092。Andconfig/mirrormaker-producer.properties指向 CLUSTER_ 中的引导代理AWSMSK；例如，bootstrap.servers=10.0.0.237:9092,10.0.2.196:9092,10.0.1.233:9092.
4. 保持 MirrorMaker 在后台运行，并继续使用CLUSTER_ONPREM. MirrorMaker 将镜像所有新数据。
5. 通过检查每个主题的最后一个偏移量与当前偏移量之间的滞后来检查镜像的进度 MirrorMaker 正在消耗。

   请记住 MirrorMaker 仅使用创建器和使用器。因此，您可以使用 kafka-consumer-groups.sh 工具检查滞后。要查找使用器组名称，请在 mirrormaker-consumer.properties 文件中查找 group.id，然后使用其值。如果文件中没有此类密钥，您可以创建它。例如，设置 group.id=mirrormaker-consumer-group。

6. 晚于 MirrorMaker 镜像完所有主题，停止所有创建器和使用器，然后停止 MirrorMaker. 然后，将创建器和使用器重定向到 `CLUSTER_AWSMSK` 集群，方式是更改该集群的创建器和使用器引导代理值。在 `CLUSTER_AWSMSK` 上重新启动所有创建器和使用器。

从一个 Amazon MSK 群集迁移到另一个

你可以使用 Apache MirrorMaker 将 MSK 集群迁移到另一个集群。例如，您可以从一个版本的 Apache Kafka 迁移到另一个版本的 Apache Kafka。有关如何使用的示例Amazon CloudFormation要执行此操作，请参阅AWS::MSK::Cluster 示例（搜索标题为的示例`Create Two MSK Clusters To Use With Apache MirrorMaker`.

MirrorMaker 1.0 最佳实践

此列表中的最佳实践适用于 MirrorMaker 1.0。

- 运行 MirrorMaker 在目标集群上。这样一来，如果发生网络问题，消息仍在源集群中可用。如果你跑了 MirrorMaker 在源集群上，在创建器中缓冲事件且存在网络问题，事件可能会丢失。
- 如果传输过程中需要加密，请在源集群中运行 MirrorMaker。
- 对于使用器，设置 auto.commit.enabled=false
- 对于创建器，设置
  - max.in.flight.requests.per.connection=1
  - retries=Int.Max_Value
  - acks=all
  - max.block.ms = Long.Max_Value
- 对于较高的创建器吞吐量：
  - 缓冲区消息和填充消息批处理 — 调整 buffer.memory、batch.size、linger.ms
  - 调整套接字缓冲区 — receive.buffer.bytes、send.buffer.bytes
- 为了避免数据丢失，请在源上关闭auto 提交，以便 MirrorMaker 可以控制提交，这通常是在从目标集群收到 ack 后进行的。如果创建器的 acks=all 且目标集群的 min.insync.replicas 设置为大于 1，则这些消息将在目标上的多个代理上持久保存 MirrorMaker 使用者在源位置提交偏移量。
- 如果顺序很重要，则可将重试次数设置为 0。或者，对于生产环境，将最大传输中连接数设置为 1，以确保在批处理中途失败时，不会无序提交发出的批处理。这样一来，将重试发送的每个批处理，直到发出下一个批处理为止。如果 max.block.ms 未设置为最大值，并且如果创建器缓冲区已满，则可能会丢失数据（具体取决于其他一些设置）。这可以阻止和反压使用器。
- 对于高吞吐量
  - 增加 buffer.memory。
  - 增大批处理大小。
  - 调整 linger.ms 以允许填充批处理。这还可以实现更好的压缩、更少的网络带宽用量以及更少的集群存储。这会导致提高保留率。
  - 监控 CPU 和内存使用情况。
- 对于高使用器吞吐量
  - 增加每个的线程/使用器数 MirrorMaker 进程 — num.streams。
  - 增加数量 MirrorMaker 在增加线程数以实现高可用性之前，请先执行计算机之间的进程。
  - 增加数量 MirrorMaker 进程依次在同一台计算机和其他计算机（具有相同的组 ID）上进行的。
  - 隔离具有非常高的吞吐量的主题，并使用单独的 MirrorMaker 实例。
- 对于管理和配置
  - 使用 Amazon CloudFormation 和配置管理工具，如 Chef 和 Ansible。

    - 使用 Amazon EFS 装载以确保可从所有 Amazon EC2 实例访问所有配置文件。
    - 使用容器来轻松扩展和管理 MirrorMaker实例。
- 通常，要使创建器饱和，需要多个使用器 MirrorMaker. 因此，请设置多个使用器。首先，在不同的计算机上设置使用器以实现高可用性。然后，扩展各个计算机以使每个分区有一个使用器，并且使用器在各个计算机之间均匀分配。
- 对于高吞吐量提取和传输，请调整接收和发送缓冲区，因为它们的默认值可能太小了。要获得最高性能，请确保流的总数 (num.streams) 与符合以下条件的主题分区总数匹配 MirrorMaker 正在尝试复制到目标集群。

MirrorMaker 2.* 的优势

- 可以利用 Apache Kafka Connect 框架和生态系统。
- 可以检测新主题和分区。
- 可以在集群之间自动同步主题配置。
- 支持“主动/主动”集群对以及任意数量的主动集群。
- 提供新增指标，包括 end-to-end 跨多个数据中心和集群的复制延迟。
- 提供在集群之间迁移使用器所需的偏移量，并提供偏移量转换工具。
- 支持高级配置文件，以实现在一个位置指定多个集群和复制流，这与为每个创建器/使用器属性单独指定低级创建器/使用器属性不同 MirrorMaker 1.* 流程。

监控 Amazon MSK 集群

亚马逊 MSK 收集 Apache Kafka 指标并将其发送给亚马逊 CloudWatch 你可以在哪里查看它们。有关 Amazon Kafka 指标的更多信息，包括亚马逊 MSK 显示的指标，请参阅监控在 Apache Kafka 文档中。

您还可以使用 Prometheus（一种开源监控应用程序）来监控您的 MSK 集群。有关 Prometheus 的信息，请参阅 Prometheus 文档中的概述。要了解如何使用 Prometheus 监控您的集群，请参阅the section called “使用 Prometheus 开放监控” (p. 103)。

主题

用于监控的亚马逊 MSK 指标 CloudWatch

亚马逊 MSK 与亚马逊集成 CloudWatch 这样你就可以收集、查看和分析 CloudWatchAmazon MSK 集群的指标。系统会自动收集您为 MSK 集群配置的指标并将其推送到 CloudWatch. 您可以将 MSK 集群的监控级别设置为以下之一：`DEFAULT`,`PER_BROKER`,`PER_TOPIC_PER_BROKER`，或`PER_TOPIC_PER_PARTITION`. 以下部分中的表格显示了从每个监控级别开始的所有可用指标。

`DEFAULT`级级指标是免费的。其他指标的定价详见亚马逊 CloudWatch定价页面。

DEFAULT级数监控

下表中描述的指标在 `DEFAULT` 监控级别可用。这些指标是免费的。

`DEFAULT` 监控级别可用的指标

名称	可见性	Dimensio	描述
`ActiveControllerCoun`	在集群进入 ACTIVE 状态后。	集群名称	在任何给定时间，每个集群只能有一个控制器处于活动状态。
`BurstBalance`	在集群进入 ACTIVE 状态后。	集群名称、代理 ID	集群中 EBS 卷的输入输出突增积分的剩余余额。使用它来调查延迟或吞吐量下降的情况。 `BurstBalance`当卷的基准性能超过最大突发性能时，不会报告 EBS 卷的相关信息。有关更多信息，请参阅 。I/O 积分和突增性能.
`BytesInPerSec`	在创建主题后。	集群名称、代理 ID、主题	每秒从客户端接收的字节数。该指标适用于每个经纪商和每个主题。
`BytesOutPerSec`	在创建主题后。	集群名称、代理 ID、主题	每秒发送到客户端的字节数。该指标适用于每个经纪商和每个主题。

名称	可见性	Dimensio	描述
ClientConnectionCount	在集群进入 ACTIVE 状态后。	集群名称、代理 ID、客户端身份验证	通过身份验证的活动连接数。
ConnectionCount	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	有效的经过身份验证、未经身份验证和代理间连接的数量。
CPUCreditBalance	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	此指标可以帮助您监控经纪商的 CPU 积分余额。如果您的 CPU 使用率保持在 20% 的基准水平以上，则可能会耗尽 CPU 积分余额，这可能会对集群性能产生负面影响。您可以采取措施减少 CPU 负载。例如，您可以减少客户请求的数量或将代理类型更新为 M5 代理类型。
CpuIdle	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	CPU 空闲时间百分比。
CpuIoWait	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	待处理的磁盘操作期间 CPU 空闲时间的百分比。
CpuSystem	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	内核空间中的 CPU 百分比。
CpuUser	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	用户空间中的 CPU 百分比。
GlobalPartitionCount	在集群进入 ACTIVE 状态后。	集群名称	集群中所有主题的分区数量，不包括副本。因为GlobalPartitionCount不包括副本，总和PartitionCount值可以高于 GlobalPartitionCount 如果某个主题的重复因子大于 1。
GlobalTopicCount	在集群进入 ACTIVE 状态后。	集群名称	集群中所有代理的主题总数。
EstimatedMaxTimeLag	在消费者群体消费某个话题之后。	消费者团体，话题	预计耗尽时间（以秒为单位）MaxOffsetLag.
KafkaAppLogsDiskUsed	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	用于应用程序日志的磁盘空间的百分比。
KafkaDataLogsDiskUsed(集群Cluster Name, Broker ID维度维度)	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	用于数据日志的磁盘空间的百分比。

名称	可见性	Dimensio	描述
KafkaDataLogsDiskUsed(集群级别Name维度维度)	在集群进入 ACTIVE 状态后。	集群名称	用于数据日志的磁盘空间的百分比。
LeaderCount	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	每个代理的分区前导总数，不包括副本。
MaxOffsetLag	在消费者群体消费某个话题之后。	消费者团体，话题	主题中所有分区的最大偏移延迟。
MemoryBuffered	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的缓冲内存大小（以字节为单位）。
MemoryCached	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的缓存内存大小（以字节为单位）。
MemoryFree	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	可供代理使用的可用内存大小（以字节为单位）。
HeapMemoryAfterGC	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	垃圾收集后正在使用的堆内存总量的百分比。
MemoryUsed	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理正在使用的内存大小（以字节为单位）。
MessagesInPerSec	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理每秒传入消息数。
NetworkRxDropped	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	丢弃的接收包的数量。
NetworkRxErrors	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的网络接收错误数。
NetworkRxPackets	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理收到的数据包的数量。
NetworkTxDropped	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	丢弃的传输包的数量。
NetworkTxErrors	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的网络传输错误的数量。
NetworkTxPackets	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理传输的数据包的数量。

名称	可见性	Dimensio	描述
OfflinePartitionsCount	在集群进入 ACTIVE 状态后。	集群名称	集群中处于脱机状态的分区的总数。
PartitionCount	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	每个代理的主题分区总数，包括副本。
ProduceTotalTimeMsMean	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	平均生成时间（以毫秒为单位）。
RequestBytesMean	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的请求字节的平均数量。
RequestTime	在应用请求限制后。	集群名称，代理 ID	代理网络和 I/O 线程处理请求所花费的平均时间（以毫秒为单位）。
RootDiskUsed	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理所使用的根磁盘的百分比。
SumOffsetLag	在消费者群体消费某个话题之后。	消费者团体，话题	主题中所有分区的聚合偏移延迟。
SwapFree	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	对代理可用的交换内存的大小（以字节为单位）。
SwapUsed	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理正在使用的交换内存的大小（以字节为单位）。
TrafficShaping	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	高级指标表示由于超出网络分配量而形成（丢弃或排队）的数据包数量。PER_BROKER 指标提供了更详细的信息。
UnderMinIsrPartitionCount	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的未完全管理分区的数目。
UnderReplicatedPartitions	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	代理的未完全复制分区的数目。
ZooKeeperRequestLatencyMsMean	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	Apache 的平均延迟（以毫秒为单位）ZooKeeper 来自经纪人的请求。
ZooKeeperSessionState	在集群进入 ACTIVE 状态后。	集群名称，代理 ID	经纪商的连接状态 ZooKeeper 会话，可能是以下之一：未连接：'0.0'，关联：'0.1'，连接：'0.5'，连接只读：'0.8'，已连接：'1.0'，已关闭：'5.0'，AUTH_FAILED：'10.0'。

PER_BROKER级数监控

在将监控级别设置为 PER_BROKER 时，除了所有 DEFAULT 级别指标之外，您还将获得下表中描述的指标。您需要为下表中的指标付费，而 DEFAULT 级别指标仍免费。此表中的指标具有以下维度：集群名称、代理 ID。

在 **PER_BROKER** 监控级别开始提供的其他指标

名称	可见性	描述
BwInAllowanceExceeded	在集群进入 ACTIVE 状态后。	因入站聚合带宽超过代理的最大值而形成的数据包的数量。
BwOutAllowanceExceeded	在集群进入 ACTIVE 状态后。	因出站聚合带宽超过代理的最大值而形成的数据包的数量。
ConnTrackAllowanceExceeded	在集群进入 ACTIVE 状态后。	因连接跟踪超过代理的最大值而形成的数据包的数量。连接跟踪与安全组有关，以跟踪建立的每个连接，以确保返回数据包按预期交付。
ConnectionCloseRate	在集群进入 ACTIVE 状态后。	每个监听器每秒关闭的连接数。此数字是针对每个监听器汇总的，并针对客户端侦听器进行筛选。
ConnectionCreationRate	在集群进入 ACTIVE 状态后。	每个监听器每秒建立的新连接数。此数字是针对每个监听器汇总的，并针对客户端侦听器进行筛选。
CpuCreditUsage	在集群进入 ACTIVE 状态后。	此指标可以帮助您监控实例上的 CPU 积分使用情况。如果您的 CPU 使用率保持在 20% 的基准水平以上，则可能会耗尽 CPU 积分余额，这可能会对集群性能产生负面影响。您可以监控此指标并发出警报，以采取纠正措施。
FetchConsumerLocalTimeMsMean	在提供创建器/使用器后。	在领导处处理使用器请求所花费的平均时间（以毫秒为单位）。
FetchConsumerRequestQueueTimeMsMean	在提供创建器/使用器后。	使用器请求在请求队列中等待的平均时间（以毫秒为单位）。
FetchConsumerResponseQueueTimeMsMean	在提供创建器/使用器后。	使用器请求在响应队列中等待的平均时间（以毫秒为单位）。
FetchConsumerResponseSendTimeMsMean	在提供创建器/使用器后。	使用器发送响应所花费的平均时间（以毫秒为单位）。
FetchConsumerTotalTimeMsMean	在提供创建器/使用器后。	使用器从代理提取数据所花费的总平均时间（以毫秒为单位）。
FetchFollowerLocalTimeMsMean	在提供创建器/使用器后。	在领导处处理跟踪器请求所花费的平均时间（以毫秒为单位）。
FetchFollowerRequestQueueTimeMsMean	在提供创建器/使用器后。	跟踪器请求在请求队列中等待的平均时间（以毫秒为单位）。
FetchFollowerResponseQueueTimeMsMean	在提供创建器/使用器后。	跟踪器请求在响应队列中等待的平均时间（以毫秒为单位）。

名称	可见性	描述
FetchFollowerResponseSendTimeMsMean	在提供创建器/使用器后。	跟踪器发送响应所花费的平均时间（以毫秒为单位）。
FetchFollowerTotalTimeMsMean	在提供创建器/使用器后。	跟踪器从代理提取数据所花费的总平均时间（以毫秒为单位）。
FetchMessageConversionsPerSec	在创建主题后。	代理每秒提取消息转换的次数。
FetchThrottleByteRate	在应用带宽限制后。	每秒的限制字节数。
FetchThrottleQueueSize	在应用带宽限制后。	限制队列中的消息数。
FetchThrottleTime	在应用带宽限制后。	平均提取限制时间（以毫秒为单位）。
NetworkProcessorAvgIdlePercent	在集群进入 ACTIVE 状态后。	网络处理器处于空闲状态的时间的平均百分比。
PpsAllowanceExceeded	在集群进入 ACTIVE 状态后。	因双向 PPS 超过代理的最大值而形成的数据包的数量。
ProduceLocalTimeMsMean	在集群进入 ACTIVE 状态后。	领导层处理请求的平均时间（以毫秒为单位）。
ProduceMessageConversionsPerSec	在创建主题后。	代理每秒生成的消息转换数。
ProduceMessageConversionsTimeMsMean	在集群进入 ACTIVE 状态后。	消息格式转换所花费的平均时间（以毫秒为单位）。
ProduceRequestQueueTimeMsMean	在集群进入 ACTIVE 状态后。	请求消息在队列中所花费的平均时间（以毫秒为单位）。
ProduceResponseQueueTimeMsMean	在集群进入 ACTIVE 状态后。	响应消息在队列中所花费的平均时间（以毫秒为单位）。
ProduceResponseSendTimeMsMean	在集群进入 ACTIVE 状态后。	发送响应消息所花费的平均时间（以毫秒为单位）。
ProduceThrottleByteRate	在应用带宽限制后。	每秒的限制字节数。
ProduceThrottleQueueSize	在应用带宽限制后。	限制队列中的消息数。
ProduceThrottleTime	在应用带宽限制后。	平均生成限制时间（以毫秒为单位）。
ProduceTotalTimeMsMean	在集群进入 ACTIVE 状态后。	平均生成时间（以毫秒为单位）。
ReplicationBytesInPerSec	在创建主题后。	每秒接收的读读读读读读读读读读读读读读读读读读
ReplicationBytesOutPerSec	在创建主题后。	每秒发送给其他代理的字节数。
RequestExemptFromThrottleTime	在应用请求限制后。	代理网络和 I/O 线程处理免受限制的请求所花费的平均时间（以毫秒为单位）。
RequestHandlerAvgIdlePercent	在集群进入 ACTIVE 状态后。	请求处理程序线程处于空闲状态的时间的平均百分比。
RequestThrottleQueueSize	在应用请求限制后。	限制队列中的消息数。
RequestThrottleTime	在应用请求限制后。	平均请求限制时间（以毫秒为单位）。

名称	可见性	描述
TcpConnections	在集群进入 ACTIVE 状态后。	显示设置了 SYN 标志的传入和传出 TCP 数据段的数量。
TrafficBytes	在集群进入 ACTIVE 状态后。	显示客户端（生产者和消费者）和代理之间的网络流量（以总字节为单位）。未报告经纪人之间的流量。
VolumeQueueLength	在集群进入 ACTIVE 状态后。	指定时间段中等待完成的读取和写入操作请求的数量。
VolumeReadBytes	在集群进入 ACTIVE 状态后。	在指定时间段内读取的字节数。
VolumeReadOps	在集群进入 ACTIVE 状态后。	在指定时间的读取操作总数。
VolumeTotalReadTime	在集群进入 ACTIVE 状态后。	指定时间段中所有读取操作耗费的总秒数。
VolumeTotalWriteTime	在集群进入 ACTIVE 状态后。	指定时间段中所有写入操作耗费的总秒数。
VolumeWriteBytes	在集群进入 ACTIVE 状态后。	在指定时间的读入的时间段内写入的字节数。
VolumeWriteOps	在集群进入 ACTIVE 状态后。	在指定时间的写入读读读读读读读读读读读读读读读读读读

PER_TOPIC_PER_BROKER级数监控

在将监控级别设置为 PER_TOPIC_PER_BROKER 时，除了 PER_BROKER 和 DEFAULT 级别的所有指标之外，您还将获得下表中描述的指标。仅 DEFAULT 级别指标是免费的。此表中的指标具有以下维度：集群名称、代理 ID、主题。

> **Important**
>
> 对于使用 Apache Kafka 2.4.1 或更高版本的 Amazon MSK 集群，下表中的指标仅在其值首次变为非零时才会出现。例如，要查看 BytesInPerSec，一个或多个创建器必须先向集群发送数据。

在 **PER_TOPIC_PER_BROKER** 监控级别开始提供的其他指标

名称	可见性	描述
FetchMessageConversionsPerSec	在创建主题后。	每秒转换的已提取消息的数量。
MessagesInPerSec	在创建主题后。	每秒接收的消息的数量。
ProduceMessageConversionsPerSec	在创建主题后。	已生成消息的每秒转换次数。

PER_TOPIC_PER_PARTITION级数监控

当您将监控级别设置为PER_TOPIC_PER_PARTITION，除了来自下表中的所有指标外，您还将获得下表中描述的指标PER_TOPIC_PER_BROKER,PER_BROKER和默认级别。仅 DEFAULT 级别指标是免费的。此表中的指标具有以下维度：消费者组、主题、分区。

在 **PER_TOPIC_PER_PARTITION** 监控级别开始提供的其他指标

名称	可见性	描述
EstimatedTimeLag	在消费者群体消费某个话题之后。	消耗分区偏移延迟的估计时间（以秒为单位）。
OffsetLag	在消费者群体消费某个话题之后。	分区级消费者在偏移量方面滞后。

使用查看亚马逊 MSK 指标 CloudWatch

您可以使用监控Amazon MSK 的指标 CloudWatch 控制台、命令行或 CloudWatch API。以下过程介绍如何使用这些不同的方式访问指标。

使用访问指标 CloudWatch 控制台

登录Amazon Web Services Management Console然后打开 CloudWatch 控制台https://console.aws.amazon.com/cloudwatch/.

1. 在导航窗格中，选择 Metrics（指标）。
2. 选择所有指标选项卡，然后选择Amazon/Kafka.
3. 要查看主题级别的指标，请选择 Topic, Broker ID, Cluster Name (主题、代理 ID、集群名称)；对于代理级别的指标，请选择 Broker ID, Cluster Name (代理 ID、集群名称)；对于集群级别的指标，请选择 Cluster Name (集群名称)。
4. （可选）在图表窗格中，选择统计数据和时间段，然后创建一个 CloudWatch 使用这些设置发出警报。

使用 Amazon CLI 访问指标

使用列表指标和get-metric-statistics命令。

使用访问指标 CloudWatch CLI

使用 mon-list-metrics 和 mon-get-stats 命令。

使用访问指标 CloudWatch API

使用 ListMetrics 和 GetMetricStatistics 操作。

消费者延迟监控

通过监控消费者延迟，您可以识别速度缓慢或停滞不前的消费者，他们无法及时了解主题中可用的最新数据。必要时，您可以采取补救措施，例如扩展或重启这些消费者。要监控消费者延迟，您可以使用亚马逊 CloudWatch 或者使用Prometheus 进行开放监控。

消费者延迟指标量化写入到您的主题的最新数据与应用程序读取的数据之间的差异。亚马逊 MSK 提供以下消费者延迟指标，您可以通过亚马逊获得这些指标 CloudWatch 或者通过Prometheus 进行公开监控：EstimatedMaxTimeLag,EstimatedTimeLag,MaxOffsetLag,OffsetLag，以及SumOffsetLag. 有关这些指标的信息，请参阅 the section called “用于监控的亚马逊 MSK 指标 CloudWatch” (p. 95)。

亚马逊 MSK 支持使用 Apache Kafka 2.2.1 或更高版本的集群的消费者延迟指标。

Note

要为 2020 年 11 月 23 日之前创建的集群开启消费者延迟监控，请确保该集群运行的是 Apache Kafka 2.2.1 或更高版本，然后创建支持案例.

使用 Prometheus 开放监控

您可以使用 Prometheus（一种用于时间序列指标数据的开源监控系统）监控您的 MSK 集群。您可以使用 Prometheus 的远程写入功能将这些数据发布到 Prometheus 的亚马逊托管服务。您还可以使用与 Prometheus 格式的指标兼容的工具或与 Amazon MSK Open Monitoring 集成的工具，例如 Datadog、Lenses、New Relic 和 Sumo logic。开源监控系统可免费使用，但跨可用区传输数据需要付费。有关 Prometheus 的信息，请参阅 Prometheus 文档。

创建启用开放监控的 Amazon MSK 集群

使用 Amazon Web Services Management Console

1. 登录Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home?region=us-east-1#/home/.
2. 在 Monitoring (监控) 部分中，选中 Enable open monitoring with Prometheus (启用 Prometheus 开源监控系统) 旁边的复选框。
3. 在页面上的各部分中提供所需的信息，并查看所有可用的选项。
4. 选择创建集群。

使用 Amazon CLI

- 调用 create-cluster 命令并指定其 `open-monitoring` 选项。启用 `JmxExporter`、`NodeExporter` 或两者。如果指定了 `open-monitoring`，则不能同时禁用这两个导出器。

使用 API

- 调用CreateCluster操作并指定`OpenMonitoring`. 启用 `jmxExporter`、`nodeExporter` 或两者。如果指定了 `OpenMonitoring`，则不能同时禁用这两个导出器。

启用对现有 Amazon MSK 集群的开放式监控

要启用开源监控系统，请确保集群处于 `ACTIVE` 状态。

使用 Amazon Web Services Management Console

1. 登录Amazon Web Services Management Console，然后打开亚马逊 MSK 控制台https://console.aws.amazon.com/msk/home?region=us-east-1#/home/.
2. 选择要更新的集群的名称。这将带您进入一个包含集群详细信息的页面。
3. 在属性选项卡，向下滚动查找监控部分。
4. 选择 Edit（编辑）。
5. 选中 Enable open monitoring with Prometheus (启用 Prometheus 开源监控系统) 旁边的复选框。
6. 选择 Save changes（保存更改）。

使用 Amazon CLI

- 调用 update-monitoring 命令并指定其 `open-monitoring` 选项。启用 `JmxExporter`、`NodeExporter` 或两者。如果指定了 `open-monitoring`，则不能同时禁用这两个导出器。

使用 API

- 调用UpdateMonitoring操作并指定`OpenMonitoring`. 启用 `jmxExporter`、`nodeExporter` 或两者。如果指定了 `OpenMonitoring`，则不能同时禁用这两个导出器。

在 Amazon EC2 实例上设置 Prometheus 主机

1. 从以下网址下载Prometheus 服务器https://prometheus.io/download/#prometheus到您的 Amazon EC2 实例。
2. 将下载的文件解压缩到某个目录并转到该目录。
3. 使用以下内容创建名为 `prometheus.yml` 的文件。

```
# file: prometheus.yml
# my global config
global:
  scrape_interval:     60s

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from
 this config.
  - job_name: 'prometheus'
    static_configs:
    # 9090 is the prometheus server port
    - targets: ['localhost:9090']
  - job_name: 'broker'
    file_sd_configs:
    - files:
      - 'targets.json'
```

4. 使用ListNodes操作以获取集群的代理列表。
5. 利用以下 JSON 创建名为 `targets.json` 的文件。将 *broker_dns_1*、*broker_dns_2* 和其余代理 DNS 名称替换为您在上一步中获取的代理 DNS 名称。包括您在上一步中获得的所有经纪人。亚马逊 MSK 使用端口 11001 作为 JMX Exporter，端口 11002 用于Node Exporter。

```
[
  {
    "labels": {
      "job": "jmx"
    },
    "targets": [
      "broker_dns_1:11001",
      "broker_dns_2:11001",
      .
      .
      .
      "broker_dns_N:11001"
    ]
  },
  {
    "labels": {
```

```
      "job": "node"
    },
    "targets": [
      "broker_dns_1:11002",
      "broker_dns_2:11002",
      .
      .
      .
      "broker_dns_N:11002"
    ]
  }
]
```

6. 要在您的 Amazon EC2 实例上启动 Prometheus 服务器，请在提取并保存 Prometheus 文件的目录中运行以下命令`prometheus.yml`和`targets.json`.

```
./prometheus
```

7. 找到您在上一步中运行 Prometheus 的 Amazon EC2 实例的 IPv4 公有 IP 地址。您在以下步骤中需要使用此公有 IP 地址。
8. 要访问 Prometheus Web UI，请打开一个可以访问您的 Amazon EC2 实例的浏览器，然后转到*Prometheus-Instance-Public-IP*`:9090`，其中*Prometheus Instance-public*是您在上一步中获得的公有 IP 地址。

Prometheus 指标

由 Apache Kafka 发送给 JMX 的所有指标都可通过 Prometheus 的开源监控系统访问。有关 Apache Kafka 指标的信息，请参阅 Apache Kafka 文档中的监控。除了 Apache Kafka 指标外，消费者滞后指标还可以在端口 11001 上以 JMX MBean 的名义获得`kafka.consumer.group:type=ConsumerLagMetrics`. 您还可以使用 Prometheus Node Exporter 在端口 11002 上获取代理的 CPU 和磁盘指标。

将 Prometheus 指标存储在亚马逊 Prometheus 托管服务中

Amazon Managed Service for Prometheus 是一项与 Prometheus 兼容的监控和警报服务，您可以使用该服务来监控 Amazon MSK 集群。这是一项完全托管的服务，可自动扩展指标的提取、存储、查询和警报。它还与集成Amazon安全服务使您能够快速安全地访问您的数据。您可以使用开源 ProMQL 查询语言来查询指标并发出警报。

有关更多信息，请参阅开始使用 Amazon Managed Service Managed Service Prometheus。

使用 LinkedIn使用亚马逊 MSK 对 Apache Kafka 的巡航控制

您可以使用 LinkedIn的 Cruise Control 来重新平衡您的 Amazon MSK 群集、检测和修复异常情况以及监控集群的状态和运行状况。

下载和构建巡航控制

1. 在与 Amazon MSK 集群相同的 Amazon VPC 中创建 Amazon EC2 实例。
2. 在上一步中创建的 Amazon EC2 实例上安装 Prometheus。请注意私有 IP 和端口。默认端口号为 9090。有关如何配置 Prometheus 以聚合集群的指标的信息，请参阅the section called “使用 Prometheus 开放监控” (p. 103).
3. 下载控制巡航在 Amazon EC2 实例上。（或者，如果您愿意，您可以使用单独的 Amazon EC2 实例进行 Cruise Control。）对于具有 Apache Kafka 版本 2.4.* 的集群，请使用最新的 2.4.* 巡航控制版本。如果您的集群的 Apache Kafka 版本早于 2.4.*，请使用最新的 2.0.* 巡航控制版本。
4. 解压缩 Cruise Control 文件，然后转到解压缩的文件夹。
5. 运行以下命令安装 git。

```
sudo yum -y install git
```

6. 运行以下命令初始化本地仓库。Replace*********使用当前文件夹的名称（解压缩 Cruise Control 下载时获得的文件夹）的名称。

```
git init && git add . && git commit -m "Init local repo." && git tag -a Your-Cruise-Control-Folder -m "Init local version."
```

7. 运行以下命令构建源代码。

```
./gradlew jar copyDependantLibs
```

配置和运行巡航控制

1. 进行以下更新：config/cruisecontrol.properties文件。替换示例引导服务器和 Apache ZooKeeper 带有集群值的连接字符串。要获取集群的这些字符串，您可以在控制台中查看集群详细信息。此外，也可以使用GetBootstrapBrokers和DescribeClusterAPI 操作或其 CLI 等效操作。

```
# If using TLS encryption, use 9094; use 9092 if using plaintext
bootstrap.servers=b-1.test-cluster.2skv42.c1.kafka.us-
east-1.amazonaws.com:9094,b-2.test-cluster.2skv42.c1.kafka.us-
east-1.amazonaws.com:9094,b-3.test-cluster.2skv42.c1.kafka.us-east-1.amazonaws.com:9094
zookeeper.connect=z-1.test-cluster.2skv42.c1.kafka.us-
east-1.amazonaws.com:2181,z-2.test-cluster.2skv42.c1.kafka.us-
east-1.amazonaws.com:2181,z-3.test-cluster.2skv42.c1.kafka.us-east-1.amazonaws.com:2181

# SSL properties, needed if cluster is using TLS encryption
security.protocol=SSL
ssl.truststore.location=/home/ec2-user/kafka.client.truststore.jks

# Use the Prometheus Metric Sampler
metric.sampler.class=com.linkedin.kafka.cruisecontrol.monitor.sampling.prometheus.PrometheusMetricS
```

```
# Prometheus Metric Sampler specific configuration
prometheus.server.endpoint=1.2.3.4:9090 # Replace with your Prometheus IP and port

# Change the capacity config file and specify its path; details below
capacity.config.file=config/capacityCores.json
```

2. 编辑config/capacityCores.json文件来指定正确的磁盘大小和 CPU 内核以及网络输入/输出限制。您可以使用DescribeCluster用于获取磁盘大小的 API 操作（或等效的 CLI）。有关 CPU 内核和网络输入/输出限制，请参阅Amazon EC2 实例类型.

```
{
  "brokerCapacities": [
    {
      "brokerId": "-1",
      "capacity": {
        "DISK": "10000",
        "CPU": {
          "num.cores": "2"
        },
        "NW_IN": "5000000",
        "NW_OUT": "5000000"
      },
      "doc": "This is the default capacity. Capacity unit used for disk is in MB, cpu
 is in number of cores, network throughput is in KB."
    }
  ]
}
```

3. 您可以有选择性地安装巡航控制界面。要下载它，请转至设置巡航控制前端.
4. 运行以下命令启动巡航控制。考虑使用类似的工具screen要么tmux以保持长时间运行的会话开放。

```
<path-to-your-kafka-installation>/bin/kafka-cruise-control-start.sh config/
cruisecontrol.properties 9091
```

5. 使用 Cruise Control API 或 UI 确保 Cruise Control 具有集群负载数据并提出重新平衡建议。获取有效的指标窗口可能需要几分钟时间。

Amazon MSK 配额

Amazon MSK 配额

- 每个账户最多 90 个经纪人，每个集群最多 30 个经纪人。要申请更高的配额，创建支持案例.
- 每个代理至少有 1 GiB 的存储空间。
- 每个代理最多有 16384 GiB 的存储空间。
- 一个集群，它使用the section called “IAM 访问控制” (p. 65)在任何给定时间，每个代理最多可以有 3000 个 TCP 连接。要提高此限制，可以调整`listener.name.client_iam.max.connections`或者`listener.name.client_iam_public.max.connections`使用 Kafka 配置属性 AlterConfig API 或`kafka-configs.sh`工具。请务必注意，将任一房产增加到较高的值都可能导致不可用。
- 对 TCP 连接的限制。一个集群，它使用the section called “IAM 访问控制” (p. 65)除了 kafka.t3.small 类型外，所有代理类型都可以以每个代理每秒最多 20 个 TCP 连接的速率接受新连接。kafka.t3.small 类型的经纪商限制为每个代理每秒 4 个 TCP 连接。如果您在 2022 年 5 月 25 日之后创建集群，它也支持连接速率突发。如果你想让较旧的集群支持突发连接速率，你可以创建支持案例.

  要处理连接失败时的重试次数，可以设置`reconnect.backoff.ms`客户端配置参数。例如，如果您希望客户端在 1 秒钟后重试连接，请设置`reconnect.backoff.ms`到 1000。有关更多信息，请参阅 。重新连接.backoff.ms在 Apache Kafka 文档中。
- 每个账户最多 100 个配置。要申请更高的配额，创建支持案例.
- 每个配置最多 50 个修订版。
- 要更新 MSK 集群的配置或 Apache Kafka 版本，请首先确保每个代理的分区数量低于中所述的限制the section called “ 将集群设置为正确大小：每个代理的分区数” (p. 123).

MSK 无服务器配额

维度	配额
最大入口吞吐量	200 MBP
最大出口吞吐量	400 MBP
最长保留持续时间	24 小时。要请求配额调整，创建支持案例.
最大客户端连接数	1000
最大连接尝试次数	每秒 100 个
最大消息大小	8 MB
最大请求大小	100MB
最大请求速率	每秒 15,000 个
每个请求的最大提取字节数	55MB
最大消费组数	500
最大分区数	120. 要请求配额调整，创建支持案例.

维度	配额
分区创建和删除的最大速率	在 5 分钟
每个分区的最大入口吞吐量	5 Mbps
每个分区的最大出口吞吐量	10 Mbps
最大分区大小	250GB
每个无服务器集群的最大客户端 VPC 数	5
每个账户的最大无服务器集群数	3

MSK Connect 配额

- 最多 100 个自定义插件。
- 最多 100 个工作器配置。
- 最多 60 个连接工作线程。如果将连接器设置为具有auto 缩放容量，则将连接器设置为具有的最大工作线程数是 MSK Connect 用于计算账户配额计算账户配额的数量。
- 每个连接器最多 10 个工作人员。

要为 MSK Connect 申请更高的配额，创建支持案例.

Amazon MSK 资源

术语资源在 Amazon MSK 中有两种含义，具体视上下文而定。在 API 上下文中，资源是一种可以在其上调用操作的结构。有关这些资源以及可以在上调用的操作的列表，请参阅资源在 Amazon MSK API 参考中。在the section called “IAM 访问控制” (p. 65)，资源是您可以允许或拒绝访问的实体，如the section called “资源” (p. 70)部分。

Apache Kafka 版本

创建Amazon MSK 集群时，您可以指定您想要使用哪个 Apache Kafka 版本。您还可以更新现有集群的 Apache Kafka 版本。

主题

支持的 Apache Kafka 版本

Amazon Managed Streaming for Apache Kafka 和 Amazon MSK 版本。

主题

Apache Kafka 版本 3.2.0

有关 Apache Kafka 版本 3.2.0 的更多信息发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 3.1.1

有关 Apache Kafka 版本 3.1.1 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.8.1

有关 Apache Kafka 版本 2.8.1 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.8.0

有关 Apache Kafka 版本 2.8.0 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.7.2

有关 Apache Kafka 版本 2.7.2 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.7.1

有关 Apache Kafka 版本 2.7.1 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.6.3

有关 Apache Kafka 版本 2.6.3 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.6.2

有关 Apache Kafka 版本 2.6.2 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.7.0

有关 Apache Kafka 版本 2.7.0 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.6.1

有关 Apache Kafka 版本 2.6.1 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.6.0

有关 Apache Kafka 版本 2.6.0 的更多信息发布说明在 Apache Kafka 下载网站上。

Apache Kafka 版本 2.5.1

Apache Kafka 2.5.1 版本包含多个错误修复和新增功能，包括针对 Apache 的传输中加密 ZooKeeper 和管理客户端。Amazon MSK 提供 TLS ZooKeeper 终端节点，您可以使用DescribeCluster operation.

的输出 DescribeCluster操作包括`ZookeeperConnectStringTls`节点，其中列出了 TLS zookeeper 端点。

以下示例展示了`ZookeeperConnectStringTls`的响应节点`DescribeCluster`operate:

```
"ZookeeperConnectStringTls": "z-3.awskafkatutorialc.abcd123.c3.kafka.us-
east-1.amazonaws.com:2182,z-2.awskafkatutorialc.abcd123.c3.kafka.us-
east-1.amazonaws.com:2182,z-1.awskafkatutorialc.abcd123.c3.kafka.us-
east-1.amazonaws.com:2182"
```

有关将 TLS 加密用于动物园管理员的信息，请参阅在 Apache 中使用 TLS 安全性 ZooKeeper (p. 81).

有关 Apache Kafka 版本 2.5.1 的更多信息，请参阅发布说明在 Apache Kafka 下载网站上。

亚马逊 MSK 错误修复版本 2.4.1.1

此版本是 Apache Kafka 版本 2.4.1 的仅限Amazon MSK 的错误修复版本。此错误修复版本包含针对KAFKA-9752，这是一个罕见的问题，它会导致消费者群体不断重新平衡并停留在`PreparingRebalance`state. 此问题会影响运行 Apache Kafka 版本 2.3.1 和 2.4.1 的集群。此版本包含一个社区制作的修补程序，在 Apache Kafka 2.5.0 版本中可用。

> Note
>
> 运行版本 2.4.1.1 的 Amazon MSK 集群与任何与 Apache Kafka 2.4.1 版兼容的 Apache Kafka 客户端兼容。

如果您更喜欢使用 Apache Kafka 2.4.1，我们建议您对新的 Amazon MSK 集群使用 MSK 错误修复版本 2.4.1.1。您可以将运行 Apache Kafka 版本 2.4.1 的现有集群更新到此版本以包含此修复程序。有关升级现有集群的信息，请参阅更新 Apache Kafka 版本 (p. 114).

要在不将集群升级到 2.4.1.1 版本的情况下解决此问题，请参阅消费者群体陷入困境`PreparingRebalance`state (p. 117)的 部分Amazon MSK 集群故障排除 (p. 117)指南。

Apache Kafka 版本 2.4.1（改为使用 2.4.1）

> Note
>
> 您无法再使用 Apache Kafka 版本 2.4.1 来创建 MSK 集群。相反，您可以改为使用亚马逊 MSK 错误修复版本 2.4.1.1 (p. 113)将安装与 Apache Kafka 版本 2.4.1 兼容。而且，如果你已经有一个装有 Apache Kafka 2.4.1 版的 MSK 集群，我们建议你将其更新为使用 Apache Kafka 2.4.1.1 版。

KIP-392 是 Apache Kafka 2.4.1 版中包含的重要 Kafka 改进建议之一。此项改进允许使用器从最近的副本提取。要使用此功能，请将使用器属性中的 `client.rack` 设置为使用器可用区的 ID。例如 AZ ID 是`use1-az1`. Amazon MSK 套装`broker.rack`更新为代理可用区 ID。您还必须将 `replica.selector.class` 配置属性设置为 `org.apache.kafka.common.replica.RackAwareReplicaSelector`，这是 Apache Kafka 提供的 rack 感知的一种实现方式。

当您使用此版本的 Apache Kafka 时，`PER_TOPIC_PER_BROKER` 监控级别中的指标仅在其值首次变为非零后才会显示。有关此问题的更多信息，请参阅 the section called “`PER_TOPIC_PER_BROKER`级数监控” (p. 101)。

有关如何查找可用区 ID 的信息，请参阅您的资源的 AZ ID中的Amazon Resource Access Manager用户指南。

有关设置配置属性的信息，请参阅配置 (p. 31)。

有关 KIP-392 的更多信息，请参阅 Confluence 页面中的允许使用器从最近的副本提取。

有关 Apache Kafka 版本 2.4.1 的更多信息，请参阅 Apache Kafka 下载网站上的版本说明。

Apache Kafka 版本 2.3.1

有关 Apache Kafka 版本 2.3.1 的更多信息，请参阅 Apache Kafka 下载网站上的版本说明。

Apache Kafka 版本 2.2.1

有关 Apache Kafka 版本 2.2.1 的更多信息，请参阅 Apache Kafka 下载网站上的版本说明。

Apache Kafka 版本 1.1（仅适用于现有集群）

您无法再使用 Apache Kafka 版本 1.1 创建新 MSK 集群。您可以继续使用配置了 Apache Kafka 版本 1.1.1 的现有集群。有关 Apache Kafka 版本 1.1.1 的更多信息，请参阅 Apache Kafka 下载网站上的版本说明。

更新 Apache Kafka 版本

您可以将现有的 MSK 集群更新到较新版本的 Apache Kafka。您无法将它更新为较旧版本。在更新 MSK 集群的 Apache Kafka 版本时，还要检查您的客户端软件，以确保其版本允许您使用集群的新 Apache Kafka 版本的功能。亚马逊 MSK 仅更新服务器软件。它不会更新您的客户端。

有关如何在更新期间使集群高度可用的信息，请参阅the section called “构建高度可用的集群” (p. 123)。

> Important
>
> 您无法为超过中所述限制的 MSK 集群更新 Apache Kafka 版本the section called “ 将集群设置为正确大小：每个代理的分区数” (p. 123).

使用 Amazon Web Services Management Console 更新 Apache Kafka 版本

1. 从打开 Amazon MSK 控制台https://console.amazonaws.cn/msk/.
2. 选择要更新 Apache Kafka 版本的 MSK 集群。
3. 在存储库的属性选项卡选择升级中的Apache Kafka 版本部分。

使用 Amazon CLI 更新 Apache Kafka 版本

1. 运行以下命令，将*ClusterArn*将使用创建集群时获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

   ```
   aws kafka get-compatible-kafka-versions --cluster-arn ClusterArn
   ```

   此命令的输出包括您可以将集群更新到的 Apache Kafka 版本的列表。其内容类似于以下示例。

   ```
   {
       "CompatibleKafkaVersions": [
           {
               "SourceVersion": "2.2.1",
               "TargetVersions": [
                   "2.3.1",
                   "2.4.1",
                   "2.4.1.1",
                   "2.5.1"
               ]
           }
       ]
   }
   ```

2. 运行以下命令，将*ClusterArn*将使用创建集群时获取的 Amazon 资源名称 (ARN)。如果您没有该集群的 ARN，可以通过列出所有集群来找到它。有关更多信息，请参阅 the section called “列出集群” (p. 15)。

   将 *Current-Cluster-Version* 替换为集群的当前版本。适用于*TargetVersion*您可以指定上一个命令输出中的任何目标版本。

Important

集群版本不是简单的整数。要查找集群的当前版本，请使用DescribeClusteroperate或describe-Amazon CLI命令。示例版本是 KTVPDKIKX0DER。

```
aws kafka update-cluster-kafka-version --cluster-arn ClusterArn --current-
version Current-Cluster-Version --target-kafka-version TargetVersion
```

上一个命令的输出如以下 JSON 所示。

```
{

    "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
    "ClusterOperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef"
}
```

3. 要得到update-cluster-kafka-versionoperate的，运行以下命令，将*ClusterOperationArn*使用您在输出中获得的 ARNupdate-cluster-kafka-version命令。

```
aws kafka describe-cluster-operation --cluster-operation-arn ClusterOperationArn
```

该 describe-cluster-operation 命令的输出如以下 JSON 示例所示。

```
{
    "ClusterOperationInfo": {
        "ClientRequestId": "62cd41d2-1206-4ebf-85a8-dbb2ba0fe259",
        "ClusterArn": "arn:aws:kafka:us-east-1:012345678012:cluster/exampleClusterName/
abcdefab-1234-abcd-5678-cdef0123ab01-2",
        "CreationTime": "2021-03-11T20:34:59.648000+00:00",
        "OperationArn": "arn:aws:kafka:us-east-1:012345678012:cluster-
operation/exampleClusterName/abcdefab-1234-abcd-5678-cdef0123ab01-2/0123abcd-
abcd-4f7f-1234-9876543210ef",
        "OperationState": "UPDATE_IN_PROGRESS",
        "OperationSteps": [
            {
                "StepInfo": {
                    "StepStatus": "IN_PROGRESS"
                },
                "StepName": "INITIALIZE_UPDATE"
            },
            {
                "StepInfo": {
                    "StepStatus": "PENDING"
                },
                "StepName": "UPDATE_APACHE_KAFKA_BINARIES"
            },
            {
                "StepInfo": {
                    "StepStatus": "PENDING"
                },
                "StepName": "FINALIZE_UPDATE"
            }
        ],
        "OperationType": "UPDATE_CLUSTER_KAFKA_VERSION",
        "SourceClusterInfo": {
            "KafkaVersion": "2.4.1"
        },
        "TargetClusterInfo": {
```

```
            "KafkaVersion": "2.6.1"
        }
    }
}
```

如果 `OperationState` 的值为 `UPDATE_IN_PROGRESS`，请等待一段时间，然后再次运行 `describe-cluster-operation` 命令。操作完成后，`OperationState` 的值变为 `UPDATE_COMPLETE`。由于 Amazon MSK 完成操作所需的时间各不相同，因此您可能需要反复检查直到操作完成。

使用 API 更新 Apache Kafka 版本

1. InvokeGetCompatibleKafkaVersionsoperate获取您可以将集群更新到的 Apache Kafka 版本的列表。
2. InvokeUpdateClusterKafkaVersion操作将集群更新到兼容的 Apache Kafka 版本之一。

Amazon MSK 集群故障排除

以下信息可帮助您排查 Amazon MSK 集群存在的问题。您也可以将问题发布到Amazon Web Services re:Post.

主题

消费者群体陷入困境PreparingRebalancestate

如果您的一个或多个消费者组陷入永久再平衡状态，原因可能是 Apache Kafka 问题KAFKA-9752，该程序会影响 Apache Kafka 版本 2.3.1 和 2.4.1。

要解决此问题，建议您将集群升级到亚马逊 MSK 错误修复版本 2.4.1.1 (p. 113)，其中包含针对此问题的修复程序。有关将现有集群更新到 Amazon MSK 错误修复版本 2.4.1.1 的信息，请参阅更新 Apache Kafka 版本 (p. 114).

在不将集群升级到 Amazon MSK 错误修复版本 2.4.1.1 的情况下解决此问题的解决方法是将 Kafka 客户端设置为使用静态成员资协议 (p. 117)，或者识别并重启 (p. 118)卡住的消费者组的协调代理节点。

实现静态成员协议

要在您的客户端中实施静态成员资格协议，请执行以下操作：

1. 设置`group.instance.id`你的财产Kafka 消费者配置为标识组中使用者的静态字符串。
2. 确保配置的其他实例已更新为使用静态字符串。
3. 将更改部署到你的 Kafka 消费者。

如果将客户端配置中的会话超时设置为允许使用者在不过早触发使用者组重新平衡的情况下恢复的持续时间，则使用静态成员协议会更有效。例如，如果您的使用者应用程序可以容忍 5 分钟的不可用状态，则会话超时的合理值为 4 分钟，而不是默认值 10 秒。

Note

使用静态成员资格协议只会降低遇到此问题的可能性。即使使用静态成员协议，你仍然可能遇到此问题。

重启协调代理节点

要重启协调代理节点，请执行以下操作：

1. 使用标识小组协调员`kafka-consumer-groups.sh`命令。
2. 使用重新启动停滞的使用者组的组协调器 RebootBrokerAPI 操作。

向Amazon传送代理日志时出错 CloudWatch 日志

当您尝试将集群设置为将代理日志发送到 Amazon 时 CloudWatch 日志，可能会遇到两个异常之一。

如果遇到 `InvalidInput.LengthOfCloudWatchResourcePolicyLimitExceeded` 异常，请重试，但使用以 `/aws/vendedlogs/` 开头的日志组。有关更多信息，请参阅 。从特定Amazon Web Services 启用日志记录.

如果您`InvalidInput.NumberOfCloudWatchResourcePoliciesLimitExceeded`异常，请选择一个现有Amazon CloudWatch 在您的账户中记录策略，并将以下 JSON 附加到该策略。

```
{"Sid":"AWSLogDeliveryWrite","Effect":"Allow","Principal":
{"Service":"delivery.logs.amazonaws.com"},"Action":
["logs:CreateLogStream","logs:PutLogEvents"],"Resource":["*"]}
```

如果您尝试将上述 JSON 附加到现有策略，但收到一个错误，指出您已达到所选策略的最大长度，请尝试将 JSON 附加到另一个亚马逊中 CloudWatch 日志策略。将 JSON 附加到现有策略后，再次尝试设置将代理日志传送到Amazon CloudWatch 日志。

无默认安全组

如果您尝试创建集群，并收到错误指示没有默认安全组，则可能是因为您使用的是共享 VPC。请向管理员申请向您授予描述此 VPC 上的安全组的权限，然后重试。有关允许此操作的策略的示例，请参阅Amazon EC2：允许以编程方式在控制台中管理与特定 VPC 关联的 EC2 安全组.

集群显示卡在 CREATING 状态

有时，集群创建可能需要长达 30 分钟。请等待 30 分钟，然后再次检查集群的状态。

集群状态从 CREATING 变为 FAILED

请尝试再次创建集群。

集群状态为 ACTIVE，但生成器无法发送数据，或者使用器无法接收数据

- 如果集群创建成功（集群状态为 ACTIVE），但您无法发送或接收数据，请确保生成器和使用器应用程序有权访问集群。有关更多信息，请参阅the section called “第 2 步：创建客户端计算机” (p. 5)中的指南。

- 如果您的生产器和使用器有权访问集群，但仍出现生成和使用数据问题，原因可能是 KAFKA-7697，这会影响 Apache Kafka 2.1.0 版本，并可能导致一个或多个代理发生死锁。请考虑迁移到 Apache Kafka 2.2.1，该版本不受此错误影响。有关如何迁移的信息，请参阅迁移 (p. 92)。

Amazon CLI无法识别Amazon MSK

如果您Amazon CLI已安装，但是该程序无法识别 Amazon MSK 命令，请升级您的Amazon CLI到最新版本。有关如何升级Amazon CLI，请参阅安装Amazon Command Line Interface. 有关如何使用Amazon CLI要运行亚马逊 MSK 命令，请参阅工作方式 (p. 9).

分区脱机或副本不同步

这些可能是磁盘空间不足的症状。请参阅 the section called “磁盘空间不足” (p. 119)。

磁盘空间不足

请参阅以下有关管理磁盘空间的最佳实践：the section called “监控磁盘空间” (p. 124)和the section called “调整数据保留参数” (p. 125)。

内存不足

如果您发现 `MemoryUsed` 指标太高或 `MemoryFree` 太低，这并不意味着存在问题。Apache Kafka 的设计初衷是充分利用内存，并以最佳方式管理内存。

创建器获取 NotLeaderForPartitionException

这往往是临时错误。将生成器的 `retries` 配置参数设置为高于其当前值的值。

复制中的分区 (URP) 大于零

`UnderReplicatedPartitions` 指标是要监控的重要指标。在正常运行的 MSK 集群中，此指标的值为 0。如果它大于零，这可能是由以下某个原因所致。

- 如果 `UnderReplicatedPartitions` 是峰值，问题可能在于该集群的大小配置不合适，无法处理传入和传出流量。请参阅 最佳实践 (p. 123)。

- 如果`UnderReplicatedPartitions`始终大于 0（包括在低流量期间），问题可能在于您设置了限制性 ACL，该 ACL 未向代理授予主题访问权限。要复制分区，必须向代理授予 READ 和 DESCRIBE 主题的权限。默认情况下，将随 READ 授权一起授予 DESCRIBE 权限。有关设置 ACL 的信息，请参阅 Apache Kafka 文档中的授权和 ACL 。

集群有名为 __amazon_msk_canary 和 __amazon_msk_canary_state 的主题

您可能会看到您的 MSK 集群有一个名为的主题`__amazon_msk_canary`还有一个名字叫`__amazon_msk_canary_state`. 这些是 Amazon MSK 为集群运行状况和诊断指标创建和使用的内部主题。这些主题的大小可以忽略不计，不能删除。

分区复制失败

确保您尚未在 CLUSTER_ACTIONS 上设置 ACL。

无法访问已开启公共访问权限的群集

如果您的集群已开启公共访问权限，但仍无法从 Internet 访问它，请按照以下步骤操作：

1. 确保集群的安全组的入站规则允许您的 IP 地址和群集的端口。有关群集端口号的列表，请参阅the section called “端口信息” (p. 91). 还要确保安全组的出站规则允许出站通信。有关安全组及其入站和出站规则的更多信息，请参阅您的 VPC 的安全组（在 Amazon VPC 用户指南 中）。
2. 确保集群的 VPC 网络 ACL 的入站规则中允许您的 IP 地址和群集端口。与安全组不同，网络 ACL 是无状态的。这意味着您必须配置入站和出站规则。在出站规则中，允许所有流量（端口范围：0-65535）到您的 IP 地址。有关更多信息，请参阅 。添加和删除规则（在 Amazon VPC 用户指南 中）。
3. 确保您使用的是公共访问引导程序代理字符串来访问集群。开启了公共访问的 MSK 集群有两个不同的引导代理字符串，一个用于公共访问，另一个用于从内部访问Amazon. 有关更多信息，请参阅 the section called “使用获取引导程序经纪人Amazon Web Services Management Console” (p. 14)。

无法从内部访问集群Amazon：联网问题

如果您的 Apache Kafka 应用程序无法与 MSK 集群成功通信，可以先执行以下连接测试。

1. 使用the section called “获取引导代理” (p. 14)中介绍的方法之一获取引导代理的地址。
2. 在以下命令中，将 *`bootstrap-broker`* 替换为您在上一步中获得的某个代理地址。如果将集群设置为使用 TLS 身份验证，则将 *`port-number`* 替换为 9094。如果集群不使用 TLS 身份验证，请将 *`port-number`* 替换为 9092。从客户端计算机运行命令。

   ```
   telnet bootstrap-broker port-number
   ```

3. 对所有引导代理重复运行上面的命令。
4. 使用中介绍的方法之一the section called “获得 Apache ZooKeeper 连接字符串” (p. 13)获取集群的 Apache 的地址 ZooKeeper 节点。
5. 在客户端计算机上运行以下命令，替换*`Apache-ZooKeeper-##`*使用其中一个 Apache 的地址 ZooKeeper 您在上一步中获得的节点。数字 2181 是端口号。对所有 Apache 重复此操作 ZooKeeper节点。

```
telnet Apache-ZooKeeper-node 2181
```

客户端计算机是否能够访问代理和 Apache ZooKeeper 节点，这意味着没有连接问题。在这种情况下，可以运行以下命令来检查 Apache Kafka 客户端是否设置正确。要获取 *bootstrap-brokers*，可使用the section called “获取引导代理” (p. 14)中介绍的方法之一。将 *topic* 替换为您的主题名称。

```
<path-to-your-kafka-installation>/bin/kafka-console-producer.sh --broker-list bootstrap-brokers --producer.config client.properties –topic topic
```

如果上一个命令成功，则表示客户端设置正确。如果仍然无法从应用程序创建和使用，请在应用程序级别调试问题。

如果客户端计算机无法访问代理和 Apache ZooKeeper 节点的，请参阅以下几个小节，获得关于客户端计算机设置的指导。

Amazon EC2 客户端和 MSK 集群位于同一 VPC 中

如果客户端计算机与 MSK 集群位于同一 VPC 中，请确保集群的安全组具有接受来自客户端计算机安全组的流量的入站规则。有关设置这些规则的信息，请参阅安全组规则。有关如何从与集群位于同一 VPC 中的 Amazon EC2 实例访问集群的示例，请参阅开始使用 (p. 5).

不同 VPC 中的 Amazon EC2 客户端和 MSK 集群

如果客户端计算机和集群位于两个不同的 VPC 中，请确保满足以下条件：

- 这两个 VPC 是对等连接的。
- 对等连接处于活动状态。
- 这两个 VPC 的路由表已正确设置。

有关 VPC 对等连接的信息，请参阅使用 VPC 对等连接。

本地客户端

对于设置为使用连接到 MSK 集群的本地客户端Amazon VPN，请确保以下各项：

- VPN 连接状态为 `UP`。有关如何检查 VPN 连接状态的信息，请参阅如何检查 VPN 隧道的当前状态？。
- 集群 VPC 的路由表包含目标格式为 `Virtual private gateway(vgw-xxxxxxxx)` 的本地 CIDR 的路由。
- MSK 集群的安全组允许端口 2181、端口 9092（如果您的集群接受纯文本流量）和端口 9094（如果您的集群接受 TLS 加密的流量）上的流量传输。

有关 Amazon VPN 问题排查方面的更多指导，请参阅客户端 VPN 问题排查。

Amazon Direct Connect

如果客户端使用Amazon Direct Connect，请参阅问题排查Amazon Direct Connect.

如果上述问题排查指导未能解决此问题，请确保没有防火墙阻止网络流量。要进一步调试，请使用类似的工具`tcpdump`和`Wireshark`来分析流量，并确保流量到达 MSK 集群。

身份验证失败：连接过多

这些区域有：`Failed authentication ... Too many connects`error 表示代理正在保护自己，因为一个或多个 IAM 客户端正在尝试以激进的速率连接到它。为了帮助经纪商接受更高比率的新 IAM 连接，您可以提高`reconnect.backoff.ms`配置参数。

要了解有关每个代理的新连接的速率限制的更多信息，请参阅Amazon MSK 配额 (p. 108)页.

MSK 无服务器：创建集群失败

如果您尝试创建 MSK 无服务器集群但工作流程失败，则可能没有创建 VPC 终端节点的权限。验证您的管理员是否已为您授予创建 VPC 终端节点的权限，方法是允许`ec2:CreateVpcEndpoint`action.

有关执行所有 Amazon MSK 操作所需的权限的完整列表，请参阅Amazon托管策略：
AmazonMSKFullAccess (p. 61).

最佳实践

本主题概述了使用 Amazon MSK 时应遵循的一些最佳实践。

将集群设置为正确大小：每个代理的分区数

下表显示了您可以每个代理拥有的最大分区数（包括领导和关注副本）。

代理类型	每个代理的最大分区数（包括领导和关注副本）
`kafka.t3.small`	300
`kafka.m5.large` 或者 `kafka.m5.xlarge`	1000
`kafka.m5.2xlarge`	2000
`kafka.m5.4xlarge`、`kafka.m5.8xlarge`、`kafka.m5.12xlarge`、`kafka.m5.16xlarge`，或者`kafka.m5.24xlarge`	4000

如果每个 Broker 的分区数超过上表中指定的最大值，则无法在集群上执行以下任何操作：

- 更新集群配置
- 更新集群的 Apache Kafka 版本
- 将集群更新为较小的代理类型
- 关联一个Amazon Secrets Manager具有 SASL/SCRAM 身份验证的集群的密钥

有关选择分区数的指导，请参阅 Apache Kafka 支持每个集群 20 万个分区。我们还建议您自己执行测试，以确定适合您的代理的类型。有关不同代理类型的更多信息，请参阅the section called “代理代理代理引擎” (p. 9)。

将集群设置为正确大小：每个集群的代理数

要确定的正确代理数量并了解成本，请参阅MSK 规模和定价电子表格。此电子表格提供了与类似的、自我管理的基于EC2 的 Amazon Kafka 相比，估计的大小和相关 Amazon MSK 成本。有关电子表格中的输入参数的更多信息，请将鼠标指针悬停在参数描述的上方。本表提供的估计是保守的，为新集群提供了一个起点。集群性能、大小和成本取决于您的使用案例，我们建议您通过实际测试进行验证。

要了解底层基础架构对 Apache Kafka 性能的影响，请参阅调整您的 Apache Kafka 集群大小以优化性能和成本的最佳实践中的Amazon大数据博客。博客文章提供了有关如何调整集群大小以满足吞吐量、可用性和延迟要求的信息。它还提供了诸如何时应该扩展等问题的答案向上与比例出，以及有关如何持续验证生产集群大小的指导。

构建高度可用的集群

使用以下建议，以便在更新期间（例如在更新代理类型或 Amazon MSK 更换代理时），您可以高度可用 MSK。

- 设置一个三可用区集群。
- 确保重复因子 (RF) 至少为 3。请注意，在滚动更新期间，RF 为 1 可能会导致脱机分区；RF 为 1 可能会导致数据丢失。
- 将最小同步副本数 (minISR) 设置为最多 RF - 1。minISR 等于 RF 可能会阻止在滚动更新期间生成到集群。当一个副本处于脱机状态时，minISR 为 2 使三向复制主题可用。
- 确保客户端连接字符串至少包含来自每个可用区的一个 Broker。在客户端的连接字符串中具有多个代理允许在特定代理脱机进行更新时进行故障转移。有关如何获取具有多个代理的连接字符串的信息，请参阅the section called “获取引导代理” (p. 14)。

监控 CPU 使用率

Amazon MSK 强烈建议您保持代理的总 CPU 使用率（定义为`CPU User + CPU System`）低于 60%。当你有至少 40% 的集群总 CPU 可用时，Apache Kafka 可以在必要时在集群中的代理之间重新分配 CPU 负载。例如，当 Amazon MSK 检测到代理故障并从中恢复时，就说明了何时需要执行此操作；在这种情况下，Amazon MSK 会执行自动维护，例如修补。另一个例子是当用户请求代理类型更改或版本升级时；在这两种情况下，Amazon MSK 部署的滚动工作流程一次让一个代理下线。当具有潜在分区的代理下线时，Apache Kafka 会重新分配分区领导以将工作重新分配给集群中的其他代理。通过遵循此最佳实践，您可以确保集群中有足够的 CPU 余量来容忍此类操作事件。

您可以使用亚马逊 CloudWatch 指标数学创建复合指标是`CPU User + CPU System`. 设置当复合指标达到平均 CPU 利用率达到 60% 时触发的警报。触发此警报警报后，使用下列选项之一扩展集群：

- 选项 1（推荐）：更新您的经纪商类型到下一个较大的类型。例如，如果当前类型是`kafka.m5.large`，更新集群以使用`kafka.m5.xlarge`. 请记住，当您更新集群中的代理类型时，Amazon MSK 会以滚动方式使代理脱机，并暂时将分区领导重新分配给其他代理。每个经纪商的规模更新通常需要 10-15 分钟。
- 选项 2：如果有一些主题包含从使用循环写入的生产者那里摄取的所有消息（换句话说，消息没有键入，排序对消费者来说并不重要），扩展你的集群通过添加经纪人。同时向吞吐量最高的现有主题添加分区。接下来，使用`kafka-topics.sh --describe`以确保将新添加的分区分配给新代理。与前一个选项相比，此选项的主要好处是您可以更精细地管理资源和成本。此外，如果CPU负载明显超过60%，则可以使用此选项，因为这种形式的扩展通常不会导致现有代理的负载增加。
- 选项 3：通过添加 broker 扩展集群，然后使用名为的分区重新分配工具重新分配现有分区`kafka-reassign-partitions.sh`. 但是，如果您使用此选项，则在重新分配分区后，集群将需要花费资源将数据从代理复制到代理。与之前的两个选项相比，这可以显著增加集群的负载。因此，当 CPU 利用率高于 70% 时，Amazon MSK 不建议使用此选项，因为复制会导致额外的 CPU 负载和网络流量。仅当前两个选项不可行时，Amazon MSK 才建议使用此选项。

其他建议：

- 监视每个代理的总 CPU 利用率，作为负载分配的代理。如果 broker 的 CPU 利用率一直不均衡，则可能表明负载在集群内分布不均匀。Amazon MSK 建议使用巡航控制通过分区分配持续管理负载分配。
- 监视生产和使用延迟。生产和消费延迟会随着 CPU 利用率呈线性增长。

监控磁盘空间

要避免出现因磁盘空间不足而无法保存消息的情况，您可以创建一个 CloudWatch 警报器监视`KafkaDataLogsDiskUsed`指标。当此指标的值达到或超过 85% 时，请执行下列一项或多项操作：

- 使用 the section called “自动扩展” (p. 18)。您也可以手动增加代理存储空间，如中所述the section called “手动扩展” (p. 19).
- 缩短消息保留期或减小日志大小。有关如何做到这一点的信息，请参阅the section called “调整数据保留参数” (p. 125)。
- 删除未使用的主题。

有关如何设置和使用警报的信息，请参阅使用 Amazon CloudWatch Alarms. 有关 Amazon MSK 指标的完整列表，请参阅监控集群 (p. 95).

调整数据保留参数

使用消息不会将其从日志中删除。要定期释放磁盘空间，您可以明确指定一个保留时间段，即消息在日志中保留的时间。您也可以指定保留日志大小。当达到保留时间段或保留日志大小时，Apache Kafka 会开始从日志中删除非活动段。

要在集群级别指定保留策略，请设置以下一个或多个参数：`log.retention.hours`、`log.retention.minutes`、`log.retention.ms` 或 `log.retention.bytes`。有关更多信息，请参阅 the section called “自定义 配置” (p. 31)。

您也可以在主题级别指定保留参数：

- 要为每个主题指定一个保留时间段，请使用以下命令。

```
kafka-configs.sh --zookeeper ZooKeeperConnectionString --alter --entity-type topics --
entity-name TopicName --add-config retention.ms=DesiredRetentionTimePeriod
```

- 要为每个主题指定一个保留日志大小，请使用以下命令。

```
kafka-configs.sh --zookeeper ZooKeeperConnectionString --alter --entity-type topics --
entity-name TopicName --add-config retention.bytes=DesiredRetentionLogSize
```

您在主题级别指定的保留参数优先于集群级别参数。

监控 Amaze Kafka 内存

我们建议您监控 Apache Kafka 使用的内存。否则，可能会导致集群不可用。

要确定 Apache Kafka 使用了多少内存，您可以监视`HeapMemoryAfterGC`指标。`HeapMemoryAfterGC`是垃圾回收后使用中的总堆内存的百分比。建议您创建一个 CloudWatch 警报在以下情况下采取行动`HeapMemoryAfterGC`增加到 60% 以上。

您可以执行的减少内存使用率的步骤各不相同。它们取决于你配置 Apache Kafka 的方式。例如，如果您使用事务性消息传递，则可以减少`transactional.id.expiration.ms`你的 Apache Kafka 配置中的值来自`604800000`ms 到`86400000`毫秒（从 7 天到 1 天）。这减少了每个事务的内存占用。

请勿添加非 MSK 代理

如果你使用 Apache ZooKeeper 命令添加代理，这些代理将不会被添加到您的 MSK 集群和您的 Apache ZooKeeper 将包含有关集群的错误信息。这可能会导致丢失数据。有关受支持的集群操作，请参阅工作方式 (p. 9)。

启用传输中加密

有关传输中加密以及如何启用此加密的信息，请参阅the section called “传输中加密” (p. 51)。

重新分配分区

要将分区移动到同一集群上的不同代理，您可以使用名为 `kafka-reassign-partitions.sh` 的分区重新分配工具。例如，在添加新代理来扩展集群后，您可以通过将分区重新分配给新代理来重新使集群达到平衡。有关如何向集群添加代理的信息，请参阅the section called “扩展集群” (p. 24)。有关分区重新分配工具的信息，请参阅 Apache Kafka 文档中的扩展集群。

Amazon词汇表

有关最新Amazon术语，请参阅《Amazon一般参考》中的Amazon术语表。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。