应用运维管理

最佳实践

文档版本 01

发布日期 2024-05-27

华为云计算技术有限公司

商标声明

HUAWEI和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址： 贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址： https://www.huaweicloud.com/

目 录

1 最佳实践概述

本章汇总呈现应用运维管理 AOM的最佳实践。

- 建设完整指标体系，实现立体化监控
- 通过告警降噪清除告警风暴
- 通过多账号聚合Prometheus实例实现指标数据统一监控
- 自建中间件接入（CCE容器场景）
  - PostgreSQL Exporter接入
  - MySQL Exporter接入
  - Kafka Exporter接入
  - Memcached Exporter接入
  - MongoDB Exporter接入
  - ElasticSearch Exporter接入
  - Redis Exporter接入
  - 其他Exporter接入

2 建设完整指标体系，实现立体化监控

本文档介绍如何建设完整的指标体系和统一监控大盘，实现资源和应用的全方位、立体化、可视化监控。

实践场景

用户体验至上的互联网时代，页面的响应速度、访问时延和页面的访问成功率常常会影响用户的体验，如果无法及时获知，就会导致流失大量用户，某商城的运维人员使用开源的监控软件，虽然能采集很多指标，但却分散在各处，无法统一展示。

解决方案

AOM能够实现云上应用的一站式立体化运维管理，在接入中心中可以接入需要监控的业务层、应用层、中间件层、基础设施层指标，在仪表盘中实现个性化监控，以及通过统一告警入口配置告警规则，实现业务的日常巡检，保障业务的正常运行。

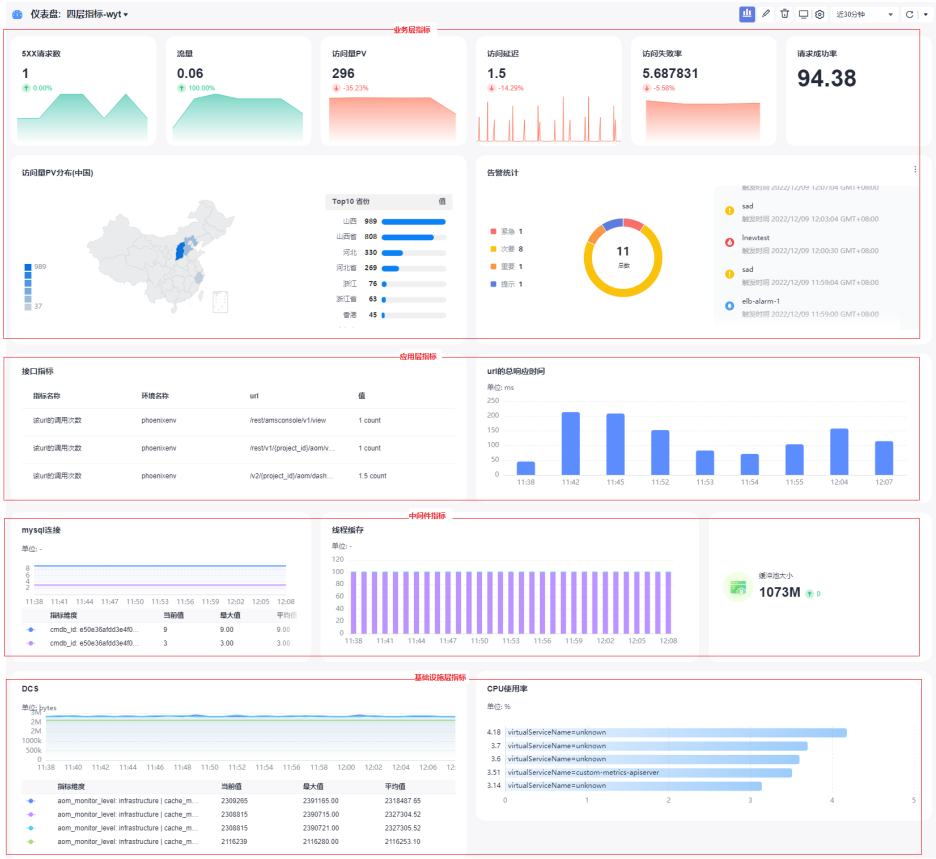
AOM提供多场景、多层次、多维度指标数据的监控能力，建立了从基础设施层指标、中间件层指标、应用层指标到业务层指标的四层指标体系，将1000+种指标数据全方位呈现，数据丰富全面。

表 2-1 AOM 支持的四层指标体系

类型	来源	指标举例	如何接入
业务层指标	通常来源于端侧日志SDK、提取的ELB日志。	访问UV、访问PV、访问延时、访问失败率、访问流量情况等	接入业务层指标
	通常来源于事务监控或上报的自定义指标。	URL的调用次数、URL的最大并发数、URL的最大响应时间等	
应用层指标	通常来源于组件性能图表或接口性能数据。	接口调用次数、请求平均时延、错误调用次数、请求吞吐量等	接入应用层指标
中间件指标	通常来源于原生中间件或云中间件数据。	文件系统容量、文件系统使用率等	接入中间件指标

类型	来源	指标举例	如何接入
基础设施层指标	通常来源于容器或云服务相关数据，例如计算、存储、网络、数据库等。	CPU使用率、内存使用率、健康状态等	接入基础设施层指标 • 接入容器指标 • 接入云服务指标

图 2-1 AOM 四层指标体系



前提条件

- 已将ELB日志接入LTS。
- 已为环境关联ECS资源。

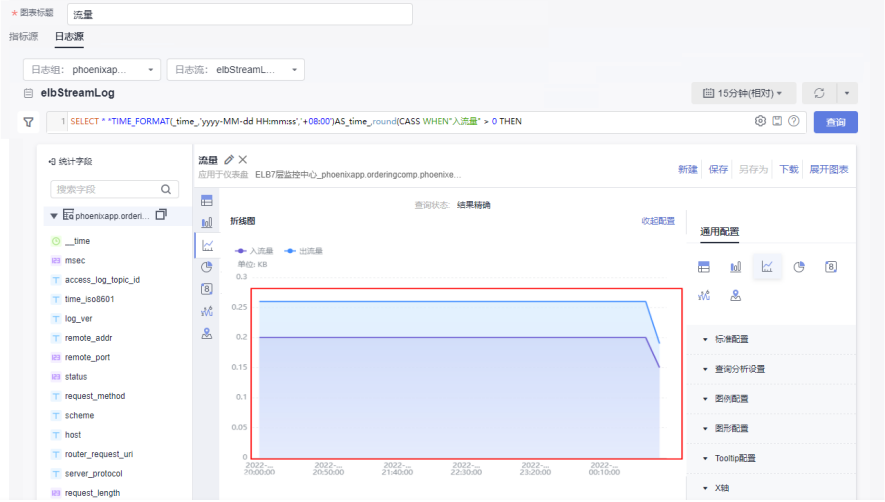
步骤一：建设四层指标体系

步骤1 接入业务层指标。

1. 登录AOM 2.0控制台。
2. 在左侧导航栏中选择“接入中心”。
3. 在右侧“业务层”面板单击需要接入的指标卡片。
   - 接入ELB 日志指标
     系统可自动接入，无需用户手动操作。

在左侧导航栏，选择已创建的仪表盘，单击页面右上角的  ，输入对应SQL语句，即可在仪表盘中查看该日志指标。以查看流量指标为例，输入对应SQL语句，单击“查询”，如**图2-2**所示。

图 2-2 查看流量指标



– 接入APM事务指标

i. 为工作负载安装APM探针，具体操作请参见**安装APM探针**。

ii. 安装完成后，请登录安装探针的服务对应的控制台界面，执行操作触发APM事务指标的采集。以本实践场景中的商城服务为例，可以在商城操作界面将对应商品添加到购物车。

iii. 登录AOM 2.0控制台。

iv. 在左侧导航栏选择“指标浏览”。在右侧区域通过选择指标的方式查看接入的APM指标。

步骤2 接入应用层指标。

1. 为工作负载安装APM探针，具体操作如下：
   a. 登录CCE控制台，单击集群名称进入集群。
   b. 在左侧导航栏中选择“工作负载 ”，选择需要上报到AOM的工作负载类型。
   c. 单击工作负载名称，选择“性能管理配置”，单击右下角“编辑”，修改“性能管理配置”相关信息。
   d. 选择“APM 2.0探针”，设置“探针版本”为“latest-x86”，“APM环境”为“phoenixenv1”，从“APM应用”的下拉列表中选择创建的“phoenixapp1”应用。
   e. 设置完成后，单击“保存”。
2. 安装完成后，请登录安装探针的服务对应的控制台界面，执行操作触发应用层指标的采集。以本实践场景中的商城服务为例，可以在商城操作界面将对应商品添加到购物车。
3. 登录AOM 2.0控制台。
4. 在左侧导航栏选择“指标浏览”。在右侧区域通过选择指标的方式查看接入的应用层指标。

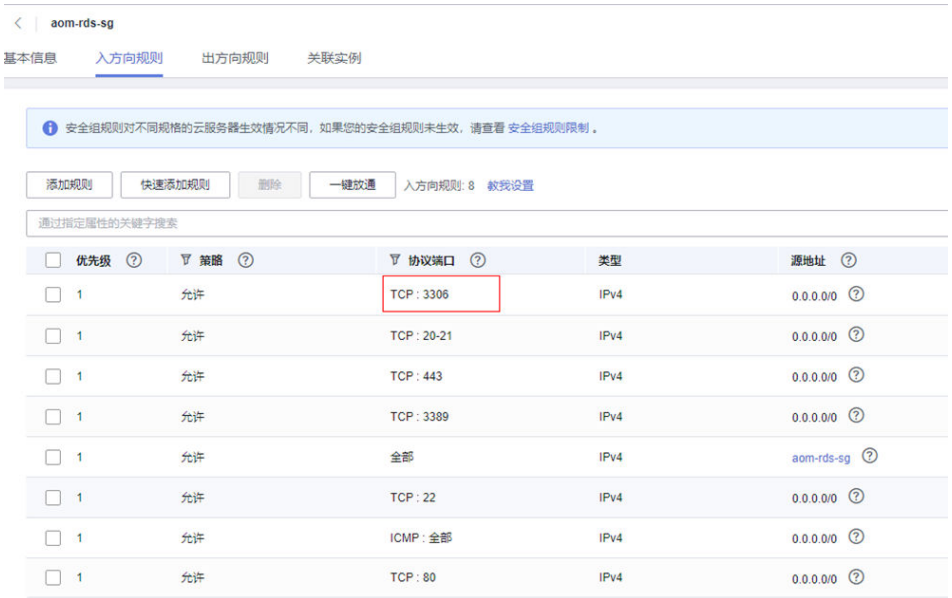
步骤3 接入中间件指标。

1. 将数据上传到ECS服务器。
   a. 下载mysqld_exporter-0.14.0.linux-amd64.tar.gz软件包，下载地址：**https://prometheus.io/download/**。
   b. 以root用户登录ECS服务器，将下载的Exporter软件包上传到ECS服务器并解压。
   c. 登录RDS 控制台，在“实例管理”界面实例列表中单击一个RDS实例名。在“基本信息”界面查看RDS安全组。

图 2-3 查看 RDS 安全组

   d. 检查RDS的安全组是否已开放3306端口。

图 2-4 检查 RDS 端口是否开放



   e. 执行以下命令，进入解压文件夹，并在ECS服务器上配置mysql.cnf文件。

```
cd mysqld_exporter-0.14.0.linux-amd64
vi mysql.cnf
```

   例如，在mysql.cnf文件中添加如下内容：

[client]

user=*root（rds用户名）*

password=*****（rds密码）*

host=*192.168.0.198（rds公网IP）*

port=*3306（端口）*

f. 执行以下命令，启动mysqld_exporter工具。

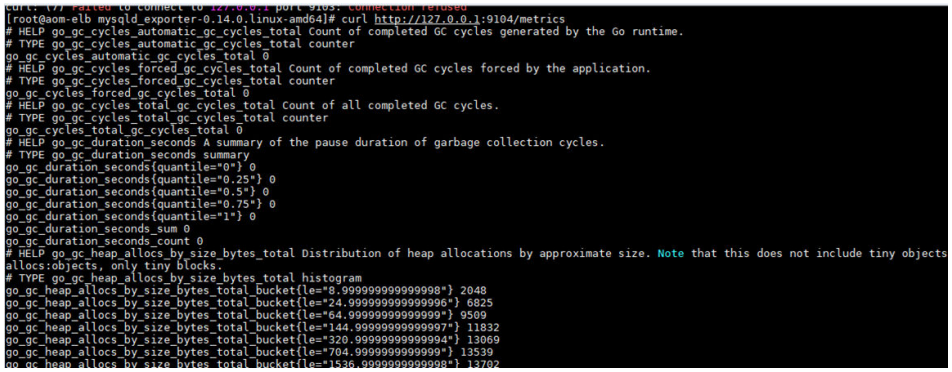
```
nohup ./mysqld_exporter --config.my-cnf="mysql.cnf" --collect.global_status --
collect.global_variables &
```

g. 执行以下命令，确认工具是否正常启动。

```
curl http://127.0.0.1:9104/metrics
```

如果回显信息如图2-5所示，能够查看到指标则说明工具启动正常。

图 2-5 查看指标



2. 通过虚机接入方式接入中间件指标。

a. 登录AOM 2.0控制台。

b. 在左侧导航栏中选择“接入中心”，在右侧“中间件”面板单击需要接入的指标卡片。

c. 在“虚机接入”界面为ECS服务器安装UniAgent采集工具，具体操作请参见**手动安装UniAgent**。

d. 为创建的“phoenixenv1”环境创建中间件采集任务。在左侧导航栏中，选择“采集任务”，单击“新增采集任务”。

图 2-6 创建采集任务

e. 在创建采集任务页面中，配置相关参数信息。

说明

关键配置项说明如下：

- Exporter地址/REDIS_Exporter地址/MySQL_Exporter地址：安装Exporter/REDIS_Exporter/MySQL_Exporter的主机IP地址和端口号。格式为IP:Port，例如：127.0.0.1:9104。
- 中间件地址/REDIS 地址/MySQL 地址：Exporter/REDIS_Exporter/MySQL_Exporter启动实例监控的主机/REDIS/MySQL，通常填写主机/REDIS/MySQL的IP地址。
- 指标：待采集的指标。默认为：''（单引号）表示输出原本的所有指标。如果需对采集指标进行过滤，则按如下格式进行填写，例如：'metric1, metric2'。
- 指标维度：新增采集指标维度。单击 + Tag ，输入指标维度名和指标维度值，字符长度不超过20个字符。最多可添加10个标签。例如：指标维度名为label1，指标维度值为label2，则添加成功后为label1:"label2"。

图 2-7 配置采集参数-1

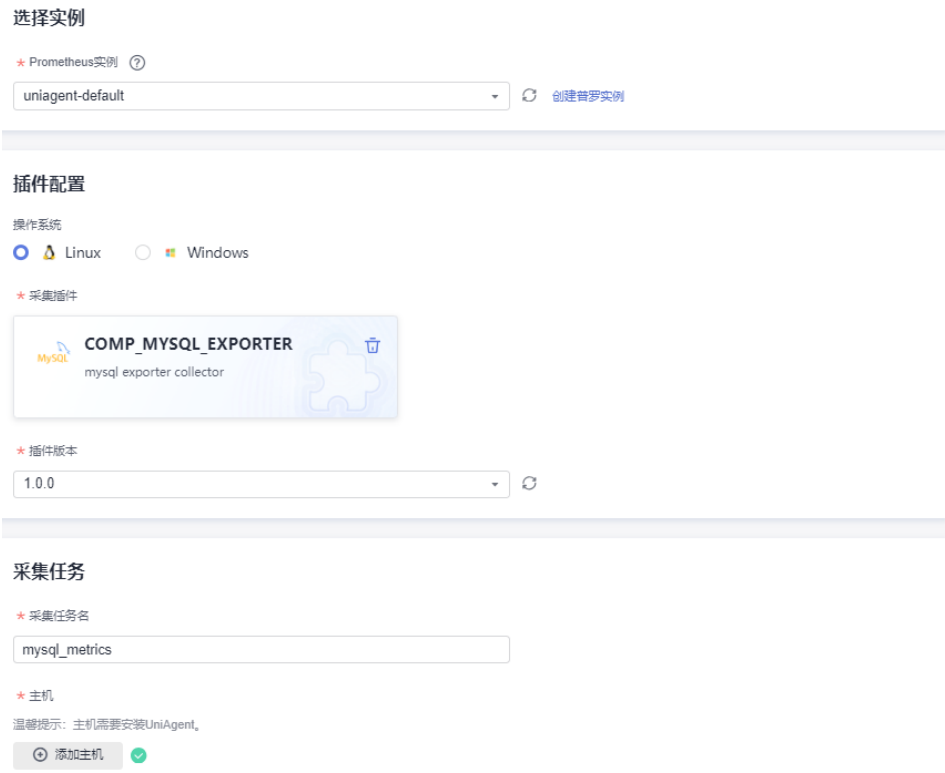
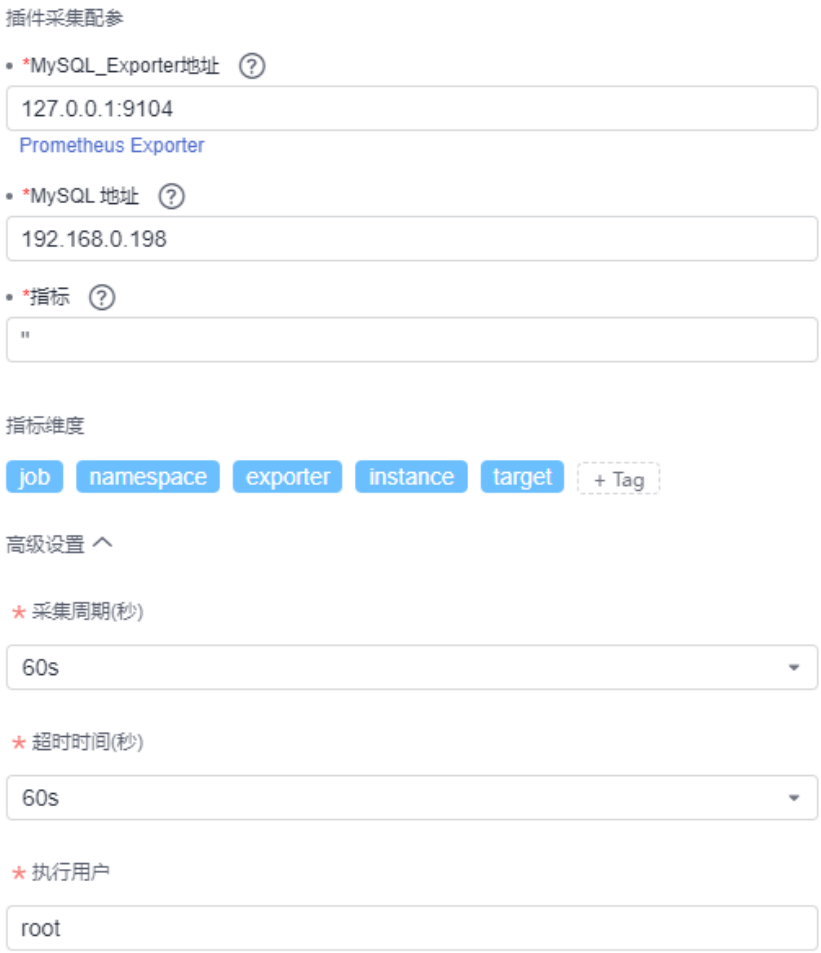


图 2-8 配置采集参数-2



f. 完成后，单击“立即创建”。

3. 接入完成后，在左侧导航栏，选择“指标浏览”。在右侧区域通过选择指标的方式查看接入的中间件指标。

步骤4 接入基础设施层指标。

1. 登录AOM 2.0控制台。
2. 在左侧导航栏中选择“接入中心”。
3. 在右侧“运行环境”与“云服务”面板单击需要接入的指标卡片。
   - 选择容器指标卡片：

     以选择“云容器引擎CCE”卡片为例，具体操作如下：

     i. 在“插件市场”界面搜索找到云原生监控插件，单击“安装”。

图 2-9 查找云原生监控插件

ii. 设置集群名称、规格信息。

iii. 开启对接第三方开关，将普罗数据上报至AOM，填写上报AOM的地址及token，并跳过证书认证。

图 2-10 设置对接第三方参数

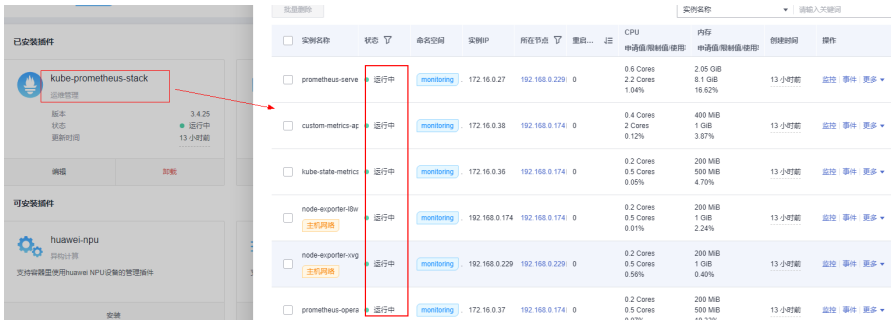
参数说明如下：

- 数据上报地址：https://aom-internal-access.{region_name}.myhuaweicloud.com:8443/v1/{project_id}/push，其中region_name为指定承载REST服务端点的服务器域名或IP，project_id 为项目的ID，您可以单击右上方的用户名称，在下拉列表中选择“我的凭证”。在“我的凭证”页面中选择“API凭证”页签。从右侧项目列表的“项目”和“项目ID”信息中获取。例如AOM服务在“华北-北京一”区域名称为“cn-north-1”。

○ Token：登录AOM 2.0控制台，在左侧导航栏选择“设置”，单击“认证管理”，从右侧AccessCode列表的“ID”信息中获取。

iv. 设置完成后，单击“安装”。安装完后单击插件查看安装状态，当状态都为“运行中”则表示插件安装成功。

图 2-11 安装 kube-prometheus-stack 插件



– 选择云服务监控指标卡片：

i. 在弹出的“云服务接入”对话框中选择需要监控的云服务。例如RDS或DCS服务。

ii. 单击“确定”完成接入。

接入完成后，系统自动跳转至“云服务监控”页面，即可查看已选择的云服务运行状态等信息。

4. 接入完成后，在左侧导航栏选择“指标浏览”。在右侧区域通过选择指标的方式查看接入的基础设施层指标。

----结束

步骤二：配置统一监控大盘

步骤1 创建指标告警规则。

通过指标告警规则可对资源的指标设置阈值条件。当指标数据满足阈值条件时产生阈值告警，当没有指标数据上报时产生数据不足事件。

按照配置方式的不同，创建指标告警规则可分为三种：**按资源类型创建**、**按全量指标创建**和**按Prometheus命令创建**。下面的操作以**按资源类型创建**为例说明。

1. 登录AOM 2.0控制台。
2. 在左侧导航栏中选择“告警管理 > 告警规则”。
3. 单击“创建告警规则”。
4. 设置告警规则的规则名称等基本信息。
5. 设置告警规则的详细信息。

   a. 选择“规则类型”为“指标告警规则”。

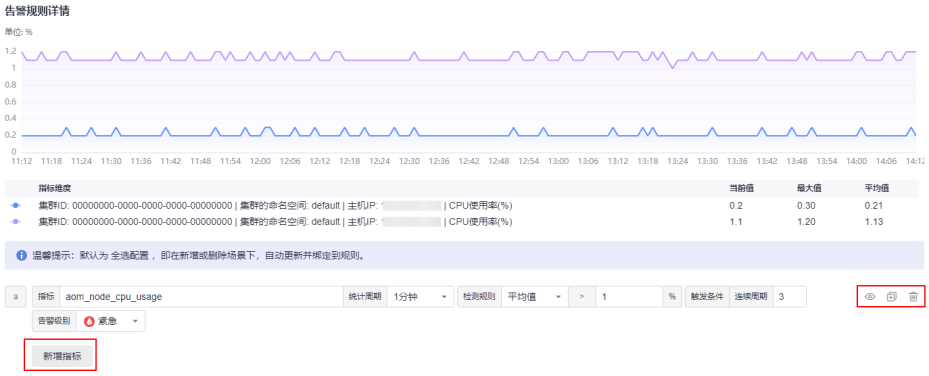
   b. 选择指标配置方式为“资源类型”，设置资源类型和监控对象信息。

   选择监控对象时，如果开启了“应用到所有”开关，将会针对应用或服务下的所有该类型指标创建一条告警规则。例如选择了“CCE / 主机 / 主机 / CPU使用率”指标，开启应用到所有开关，则会为CCE服务下所有主机创建一条告警规则。

   c. 设置告警规则详情。

指标告警的检测规则，由统计方式（平均值、最小值、最大值、总计、样本个数）、判断条件（>=、<=、>、<）和阈值组成。例如，检测规则设置为“平均值>1”，表示指标的平均值大于已设置的阈值1时，生成指标告警。

图 2-12 设置告警详细信息



d. 单击“高级设置”，设置检查频率、告警恢复等信息。

6. 设置告警通知策略。告警通知策略有两种方式，此处选择直接告警方式。

   直接告警：满足告警条件，直接发送告警。选择直接告警方式，需要设置通知频率和是否启用告警行动规则。

   a. 设置发送告警通知的频率，请根据需要从下拉列表中选择。
   b. 设置是否启用告警行动规则。启用告警行动规则后，系统根据关联SMN主题与消息模板来发送告警通知。
   c. 启用告警行动规则后，需要设置是否开启告警恢复通知。开启告警恢复通知后，当满足告警条件中设置的告警恢复条件，则按照选择的告警行动规则发送告警恢复通知。

图 2-13 设置直接告警方式

7. 单击“立即创建”，完成创建。创建完成后，单击“返回告警规则列表”可查看已创建的告警规则。

如下图所示，创建了一条指标告警规则，单击规则名称前的>，可查看该告警规则的详细信息。

在展开的列表中，只要某个主机的CPU使用率满足设置的告警条件时，在告警界面就会生成一条指标类告警，您可在左侧导航栏中选择“告警管理 > 告警列表”，在告警列表中查看该告警。只要某个主机满足已设的通知策略，系统就会以邮件、短信或企业微信等方式发送告警通知给指定人员。

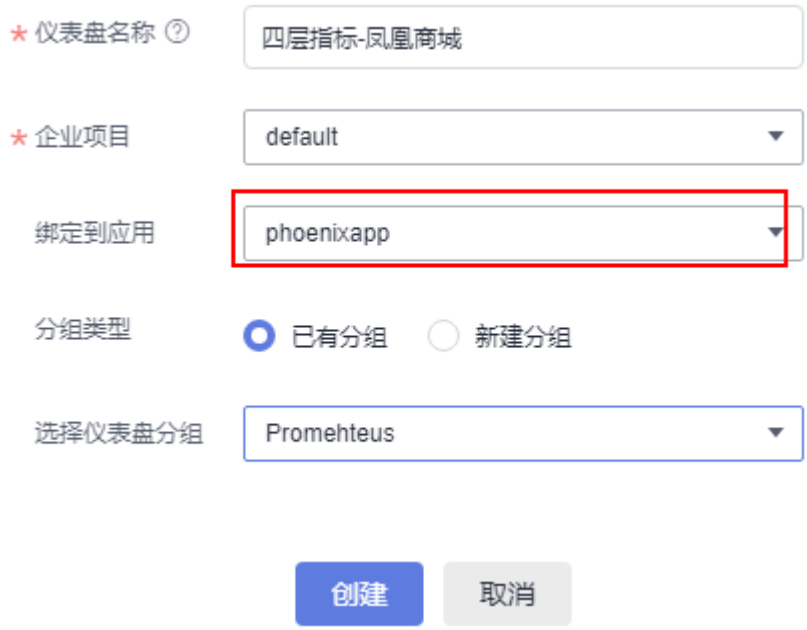
图 2-14 创建指标告警规则

步骤2 创建仪表盘。

1. 新建仪表盘。
   a. 登录AOM 2.0控制台。
   b. 在左侧导航栏选择“仪表盘”。
   c. 单击列表左上角的“创建仪表盘”。
   d. 在弹出的“新建仪表盘”对话框中，设置相关参数。

      将仪表盘绑定到事先创建的应用，后续可以在“应用监控”页面可视化监控应用的关键指标。

      图 2-15 新建仪表盘



   e. 设置完成，单击“创建”。
2. 为仪表盘添加可视化图表。

a. 在仪表盘列表中，单击已创建的仪表盘。

b. 进入对应仪表盘页面，单击页面右上角的，为该仪表盘添加图表。请根据需要，选择合适的图表。

表 2-2 添加图表

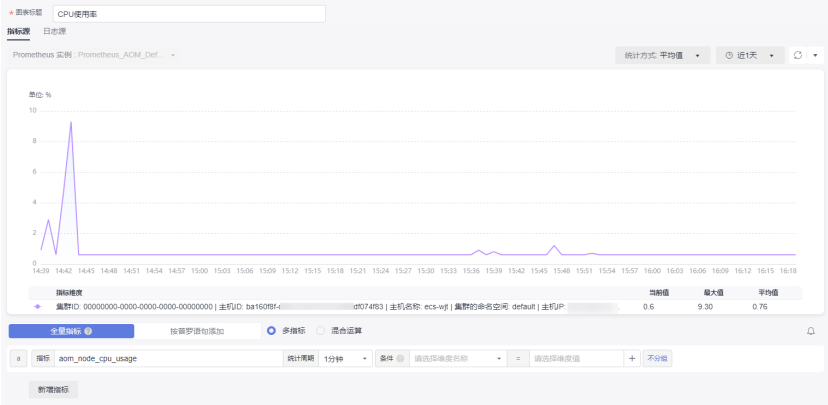
添加图表类型	数据来源	使用场景
请添加指标图表	指标数据	监控基础设施层、中间件、应用层和业务层指标。
请添加日志图表	日志数据	监控业务指标或其他日志指标，如基于ELB日志清洗出来的接口黄金指标（时延、吞吐和错误）。

下面以添加**“CPU使用率”**的指标图表和**“延迟”**的日志图表为例说明。

- 添加“CPU使用率”的指标图表。

  选择“CPU使用率”指标，设置完成后，添加的指标图表如**图2-16**所示。

  图 2-16 添加指标图表

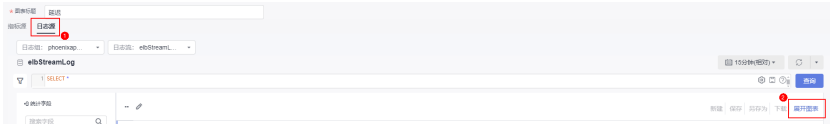
  

- 添加“延迟”的日志图表。单击“日志源”，设置日志图表的相关参数。

  可直接从图表中获取SQL查询语句：

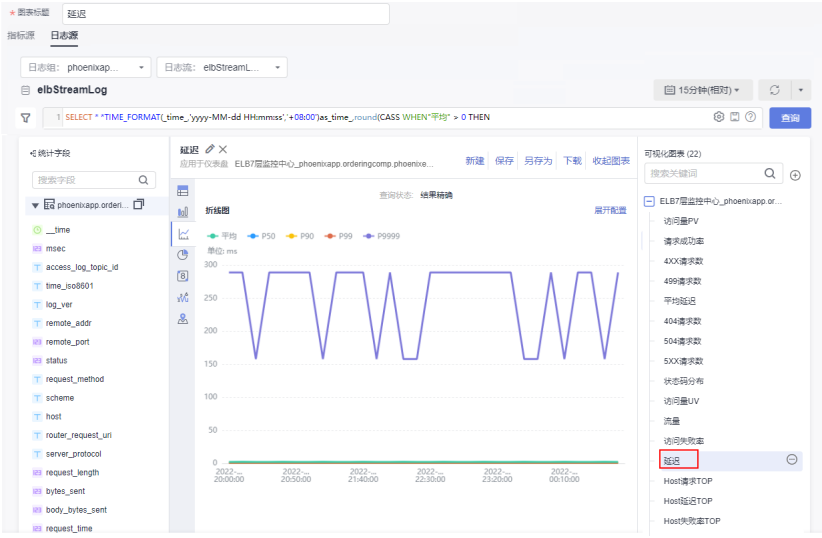
  1) 在图表展示区右上方单击“展开图表”。

  图 2-17 展开图表

  2) 在“可视化图表”列表中选择需要监控的日志指标，例如“延迟”，如**图2-18**所示。

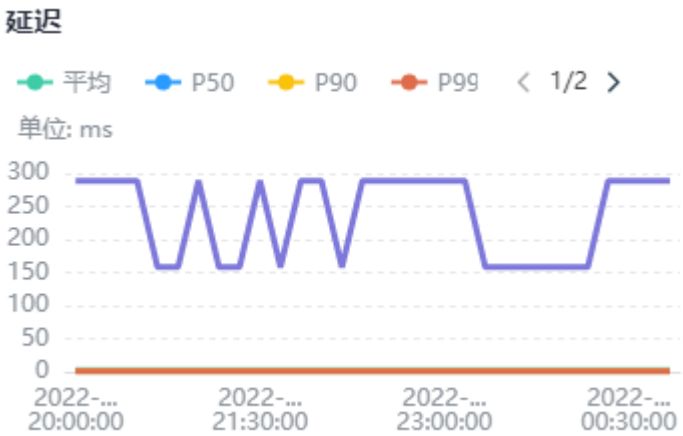
图 2-18 选择日志指标



3) 该指标对应的查询语句会自动填充到SQL语句设置区。

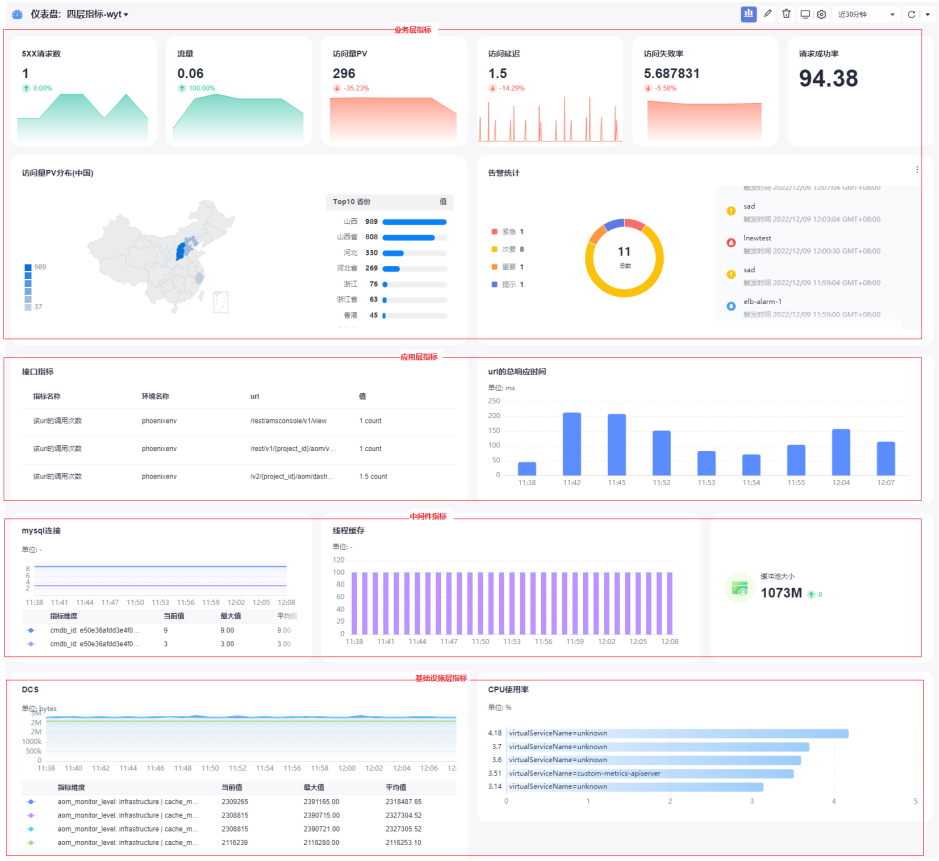
参数设置完成后，单击“添加至仪表盘”。添加的日志图表如图2-19所示。

图 2-19 添加日志图表



c. 可重复上面的操作为仪表盘添加多个可视化图表。添加完成后，单击，保存仪表盘，如图2-20所示。

图 2-20 添加可视化图表



----结束

3 通过告警降噪清除告警风暴

本文档介绍如何为告警规则配置告警降噪功能，在发送告警通知前按告警降噪规则对告警进行处理，处理完成后再发送通知，避免产生告警风暴。

实践场景

某电商运维人员在定位分析应用、资源及业务的实时运行状况时，发现系统上报的告警数量过大，重复性告警过多，需要从众多告警中快速及时发现故障，全面掌握应用。

解决方案

AOM通过设置告警规则，实时监控环境中主机、组件等资源使用情况。当产品自身或外部服务存在异常情况时，立即触发告警。并提供告警降噪功能，支持发送告警通知前按告警降噪规则对告警进行处理，处理完成后再发送通知，帮助用户快速识别重点问题，避免产生告警风暴。

告警降噪功能分为分组、去重、抑制、静默四部分：

- 使用分组规则，您可以从告警中筛选出满足条件的告警子集，然后按分组条件对告警子集分组，告警触发时同组告警会被汇聚在一起发送一条通知。
- 使用抑制规则，您可以抑制或阻止与某些特定告警相关的其他告警通知。例如：当严重级别的告警产生时，可以抑制与其相关的低级别的告警。或当节点故障发生时，抑制节点上的进程或者容器的所有其他告警。
- 使用静默规则，您可以在指定时间段屏蔽告警通知，静默规则一旦创建完成，即刻生效。
- 去重为内置策略，服务后台会自动检验告警内容是否一致实现去重的效果，用户无需手动创建规则。

下面以监控ELB业务层全量指标为例说明。

前提条件

已创建告警行动规则。

步骤一：创建分组规则

创建一个分组规则，当产生AOM的紧急、重要告警时，触发“Monitor_host”行动规则，且告警按照告警源合并分组。

步骤1 登录AOM 2.0控制台。

步骤2 在左侧导航栏中选择“告警管理 > 告警降噪”。

步骤3 在“分组规则”页签下单击“创建分组规则”，设置规则名称、分组条件等信息。

图 3-1 创建分组规则

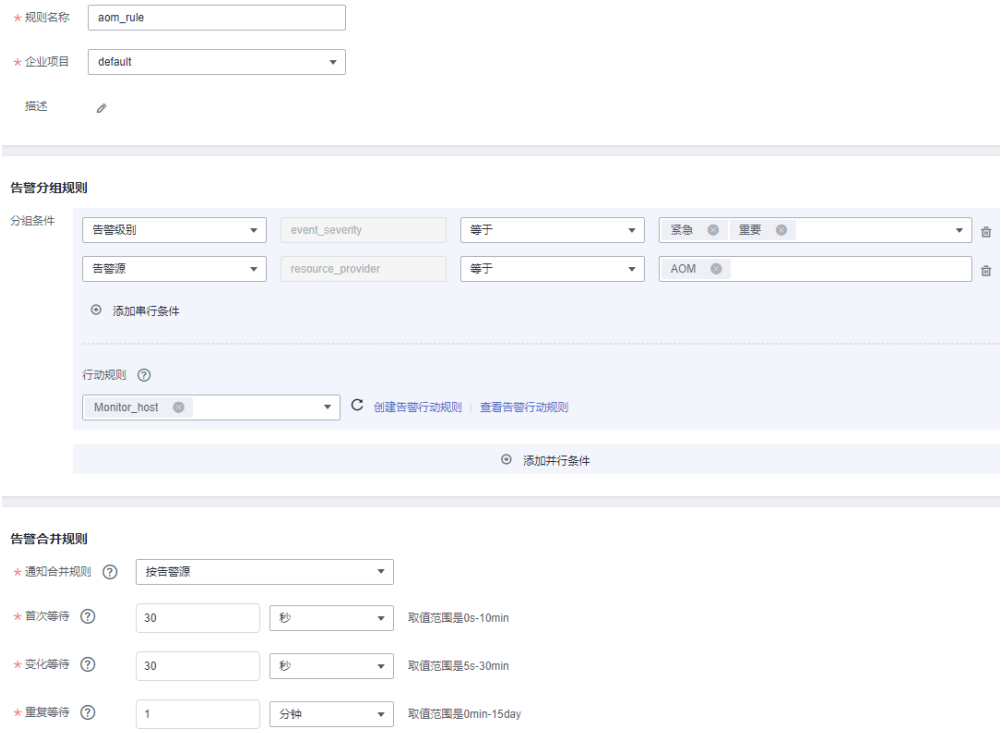


表 3-1 告警合并规则说明

通知合并方式	根据指定字段对分组后的告警合并 。合并在一组的告警会被汇聚在一起发送一条通知。 合并方式包括： ● 按告警源：由相同告警源触发的告警，合并为一组发送告警通知。 ● 按告警源 + 严重度：由相同告警源触发的告警，且其严重度相同时，合并为一组发送告警通知。 ● 按告警源 + 所有标签：由相同告警源触发的告警，且其标签相同时，合并为一组发送告警通知。
首次等待	首次创建告警合并集合后，等待多久发送第一次告警通知。通常设置为秒级别的时间，便于告警合并后再发送，避免告警风暴。 取值范围：0s-10min，推荐设置为 15s。
变化等待	合并集合内的告警数据发生变化后，等待多久发送告警通知。通常设置为分钟级别的时间。如果您需要尽快收到告警通知，也可设置为秒级时间。 此处的变化是指新增告警或告警状态改变。 取值范围：5s-30min，推荐设置为60s。

重复等待	合并集合内的告警数据重复后，等待多久发送告警通知。通常设置为小时级别的时间。 此处的重复是指无新增告警和状态变化，仅其他属性（例如标题、内容等）改变。 取值范围：0min-15day，推荐设置为1h。

----结束

步骤二：创建全量指标告警规则

通过指标告警规则可对资源的指标设置阈值条件。当指标数据满足阈值条件时产生阈值告警，当没有指标数据上报时产生数据不足事件。

按照配置方式的不同，创建指标告警规则可分为三种：**按资源类型创建**、**按全量指标创建**和**按Prometheus命令创建**。下面的操作以**按全量指标创建**为例说明，创建一个监控ELB业务层全量指标的告警规则。

步骤1 登录AOM 2.0控制台。

步骤2 在左侧导航栏中选择“告警管理 > 告警规则”。

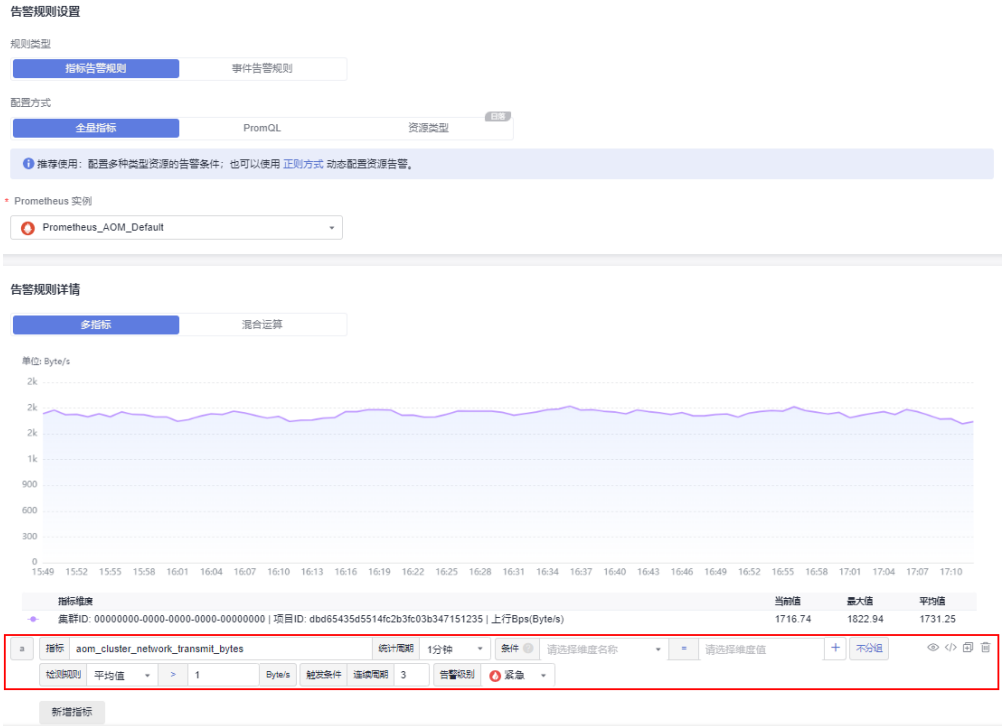
步骤3 在“指标或事件告警规则”页签下单击“创建告警规则”。

步骤4 设置告警规则的规则名称等基本信息。

步骤5 设置告警规则的详细信息。

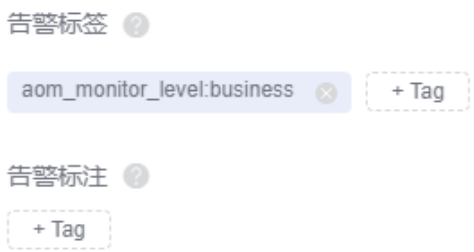
1. 选择“规则类型”为“指标告警规则”，“配置方式”为“按全量指标”。
2. 设置指标、环境、检查频率等告警条件参数。

图 3-2 设置告警规则详细信息



3. 根据需要设置告警标签和告警标注信息，为告警匹配分组，后续可关联告警降噪策略来发送告警通知。**步骤5.2**选择的是业务层指标，所以此处标签设置为“aom_monitor_level:business”。

图 3-3 自定义标签信息



说明

全量指标的标签为key:value键值对格式，key通常设置为“aom_monitor_level”，value的设置说明如下：

- 全量指标为基础设施层指标：infrastructure
- 全量指标为中间件指标：middleware
- 全量指标为应用层指标：application
- 全量指标为业务层指标：business

步骤6 设置告警通知策略。告警通知策略有两种方式，此处选择告警降噪方式。

告警降噪：对告警信息自动匹配告警降噪分组规则后再发送告警，防止产生告警风暴。

图 3-4 设置告警降噪方式

步骤7 单击“立即创建”，完成创建。创建完成后，单击“返回告警规则列表”可查看已创建的告警规则。

如下图所示，创建了一条指标告警规则，单击规则名称前的∨，可查看该告警规则的详细信息。

图 3-5 创建指标告警规则

在展开的列表中，只要指标数据满足设置的告警条件时，在告警界面就会生成一条指标类告警，您可在左侧导航栏中选择“告警管理 > 告警列表”，在告警列表中查看该告警。

只要该告警满足已设的通知策略，系统就会以邮件、短信或企业微信等方式发送告警通知给指定人员。

图 3-6 接收告警通知

----结束

4 通过多账号聚合 Prometheus 实例实现指标数据统一监控

本文档介绍通过配置统一监控告警，同时监控不同账号下的指标数据。

实践场景

某电商平台运维人员在监控指标时，只能实时监控一个账号下的指标数据，无法同时监控其他账号。

解决方案

AOM通过Prometheus监控功能，创建多账号聚合实例，并接入账号、云服务与云服务相关指标，支持在“指标浏览”界面同时监控多个成员账号的指标数据并为这些指标设置告警规则。当指标存在异常情况时，立即触发告警，发送通知。

前提条件

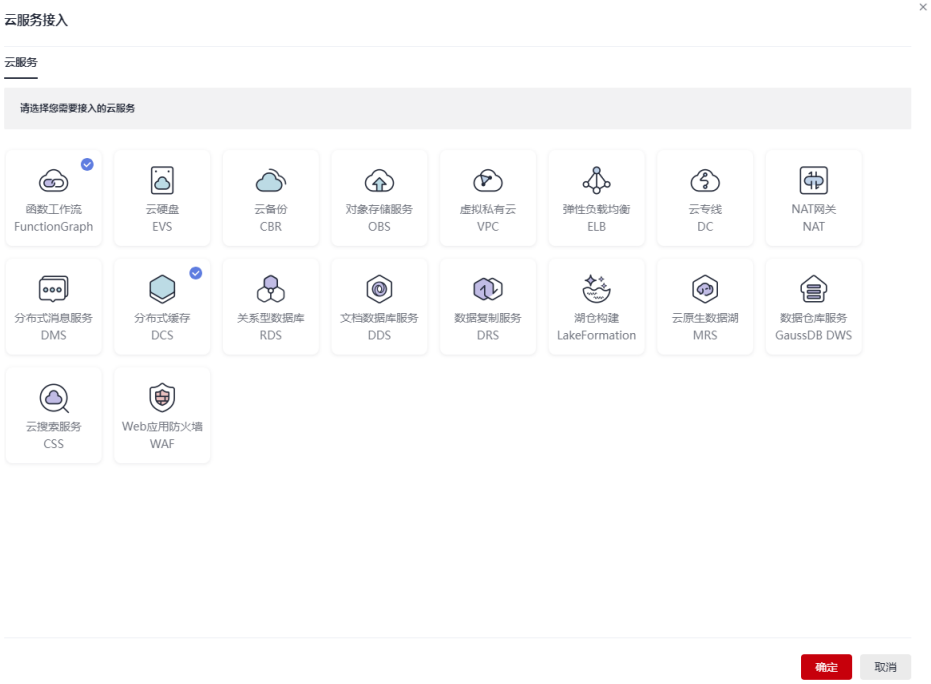
- 监控账号与被监控账号均已**加入组织**。监控账号需为组织管理员，非组织管理员的组织成员需进行**步骤二**，授权委托管理员身份。
- 被监控账号当前支持汇聚的包括“Prometheus for 云服务”可接入的18个云服务指标（FunctionGraph，EVS，CBR，OBS，VPC，ELB，DC，NAT，DMS，DCS，RDS，DDS，DRS，LakeFormation，MRS，GaussDB DWS，CSS，WAF）以及ICAgent采集的CCE和ECS指标。

步骤一：被监控账号接入云服务资源

下面的操作以接入**FunctionGraph**、**DCS**、**ECS**为例说明。接入CCE与接入ECS类似，但当前CCE购买时默认自动安装ICAgent。接入其他云服务资源的操作与接入FunctionGraph、DCS类似。

- 接入FunctionGraph、DCS云服务资源。
  a. 登录AOM 2.0控制台。
  b. 在左侧导航栏中选择“接入中心”。
  c. 在“云服务”下单击需要接入的云服务资源。
  d. 选择云服务FunctionGraph、DCS后，单击“确定”，即可接入。

图 4-1 云服务接入



- 接入ECS资源。
  a. 将鼠标移动到右上方的用户名称，并在下拉列表中选择“我的凭证”。

  图 4-2 我的凭证

  b. 在“我的凭证”页面中选择“访问密钥”页签。
  c. 在列表上方单击“新增访问密钥”，输入验证码或密码。

图 4-3 新增访问密钥

d. 单击“确定”，生成并下载AK/SK。

创建访问密钥成功后，您可以在访问密钥列表中查看访问密钥ID（AK），在下载的.csv文件中查看秘密访问密钥（SK）。

e. 返回AOM 2.0控制台页面，在左侧导航栏中选择“采集管理”，进入“采集管理”界面。

f. 在左侧导航栏中，选择“UniAgent管理 > 虚机接入”。

g. 在虚机接入中，选择待安装ICAgent的主机，单击“插件批量操作”。

图 4-4 安装 ICAgent

h. 在弹出的对话框中，操作类型选择“安装”，选择插件为“ICAgent”，插件版本选择“5.12.163”，在“ak”、“sk”中输入d获取的AK/SK。

i. 设置完成后，单击“确认”，安装ICAgent。

步骤二：开启 AOM 可信服务并设置委托管理员（若进行监控的账号为组织管理员，可跳过此步骤）

步骤1 使用组织中的管理员账号登录组织Organizations控制台。

步骤2 在左侧导航栏选择“可信服务”。

步骤3 在可信服务列表中，单击“应用运维管理服务（AOM）”操作列的“启用”，开启AOM可信服务。

步骤4 单击“应用运维管理服务（AOM）”操作列的“设置委托管理员”，选择需要设置为委托管理员的账号，单击“确定”。如图4-5所示，将paas_aom设置为委托管理员。

图 4-5 设置委托管理员

----结束

步骤三：配置多账号聚合实例

步骤1 使用组织中身份为管理员或委托管理员的监控账号登录AOM 2.0 控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”，单击“创建Prometheus实例”。

步骤3 填写实例名称，选择实例类型为“Prometheus for 多账号聚合实例”。

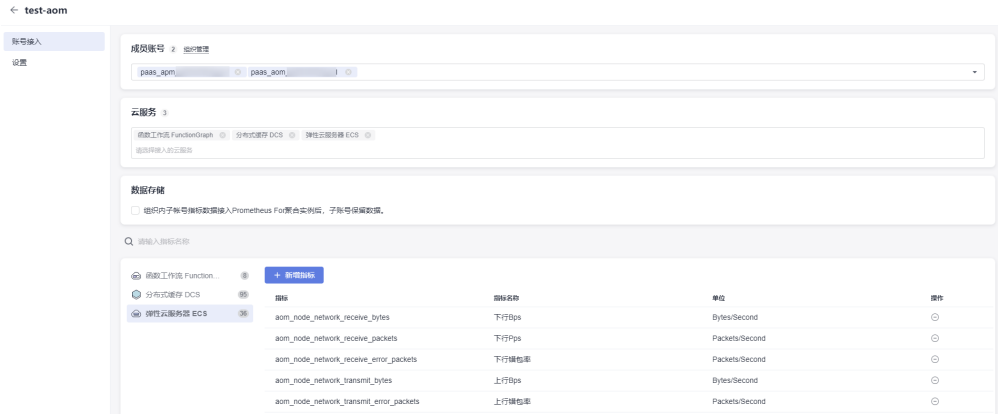
步骤4 单击“确定”完成创建。如图4-6所示，创建了一个名为“test-aom”的多账号聚合实例。

图 4-6 Prometheus 实例列表

步骤5 在“Prometheus实例”列表中单击创建的多账号聚合实例的名称，进入多账号聚合实例的“账号接入”页面，选择需要接入的账号，云服务及云服务指标。

例如，成员账号接入“paas_apm、paas_aom”。云服务选择接入“函数工作流FunctionGraph、分布式缓存 DCS、弹性云服务器 ECS”。在云服务列表中选择云服务后，单击“新增指标”，可以在新增指标弹框里勾选任意需要接入的指标。

图 4-7 账号接入



接入后，等待2-3min在指标浏览处即可查看接入的指标数据。

----结束

步骤四：监控账号配置统一监控告警

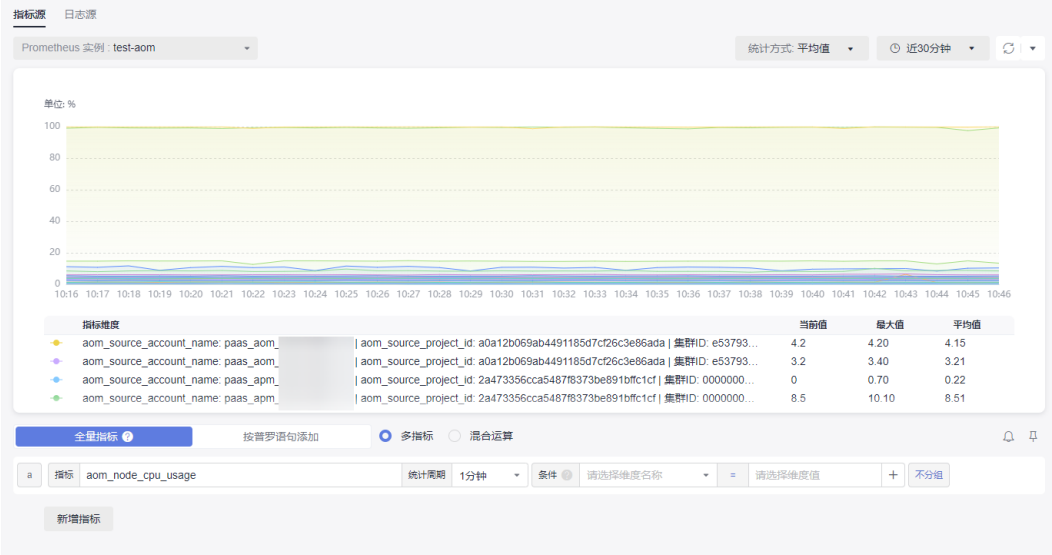
步骤1 验证多账号聚合实例的指标是否接入。

1. 在左侧导航栏选择“指标浏览”，在“Prometheus实例”下拉列表中选择步骤三配置的多账号聚合实例“test-aom”。
2. 单击“全量指标”，选择一个指标并复制指标名称。
3. 单击“按普罗语句添加”，输入普罗表达式：sum(指标名称) by (aom_source_account_name)，即可查看指标是否接入。

图 4-8 查看指标

步骤2 单击“全量指标”，选择需要监控的指标，即可查看该账号下的指标。如图4-9所示，选择指标“aom_node_cpu_usage”，即可在图表中实时监控“paas_apm”与“paas_aom”账号下该指标的指标值与趋势。

图 4-9 查看指标

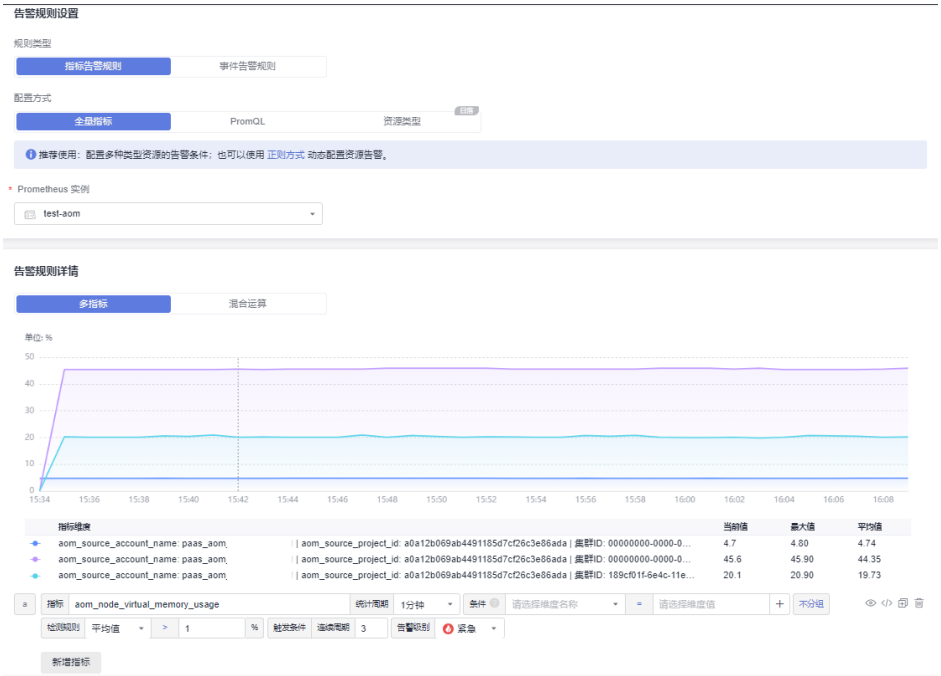


步骤3 单击指标列表右上角的 ，为选择的指标新增告警规则。

1. 设置告警规则的规则名称等基本信息。
2. 设置告警规则的详细信息。
   a. 告警规则设置中的规则类型、配置方式、Prometheus 实例默认选择为指标浏览处的配置。
   b. 设置告警规则详情。监控的指标自动选择为指标浏览处选择的指标。

      指标的详细设置由统计周期、条件、检测规则、触发条件以及告警级别组成。指标告警的检测规则，由统计方式（平均值、最小值、最大值、总计、样本个数）、判断条件（>=、<=、>、<）和阈值组成。例如，统计周期为“1分钟”，检测规则设置为“平均值>1”，触发条件为连续周期“3”，告警级别为“紧急”，表示连续三个统计周期，指标的平均值大于已设置的阈值1时，生成紧急告警。

图 4-10 设置告警规则



c. 单击“高级设置”，设置检查频率、告警恢复等信息。

d. 设置告警通知策略。告警通知策略有两种方式，如图4-11所示，此处选择直接告警方式。

直接告警：满足告警条件，直接发送告警。选择直接告警方式，需要设置通知频率和是否启用告警行动规则。

i. 设置发送告警通知的频率，请根据需要从下拉列表中选择。

ii. 设置是否启用告警行动规则。启用告警行动规则后，系统根据关联SMN主题与消息模板来发送告警通知。

图 4-11 告警通知

e. 单击“立即创建”，完成创建。创建完成后，单击“返回告警规则列表”可查看已创建的告警规则。

如图4-12所示，单击规则名称前的∨，可查看该告警规则的详细信息。

在展开的列表中，只要监控对象满足设置的告警条件时，在告警界面就会生成一条指标类告警，您可在左侧导航栏中选择“告警管理 > 告警列表”，在告警列表中查看该告警。只要某个主机满足已设的通知策略，系统就会以邮件、短信或企业微信等方式发送告警通知给指定人员。

图 4-12 告警规则

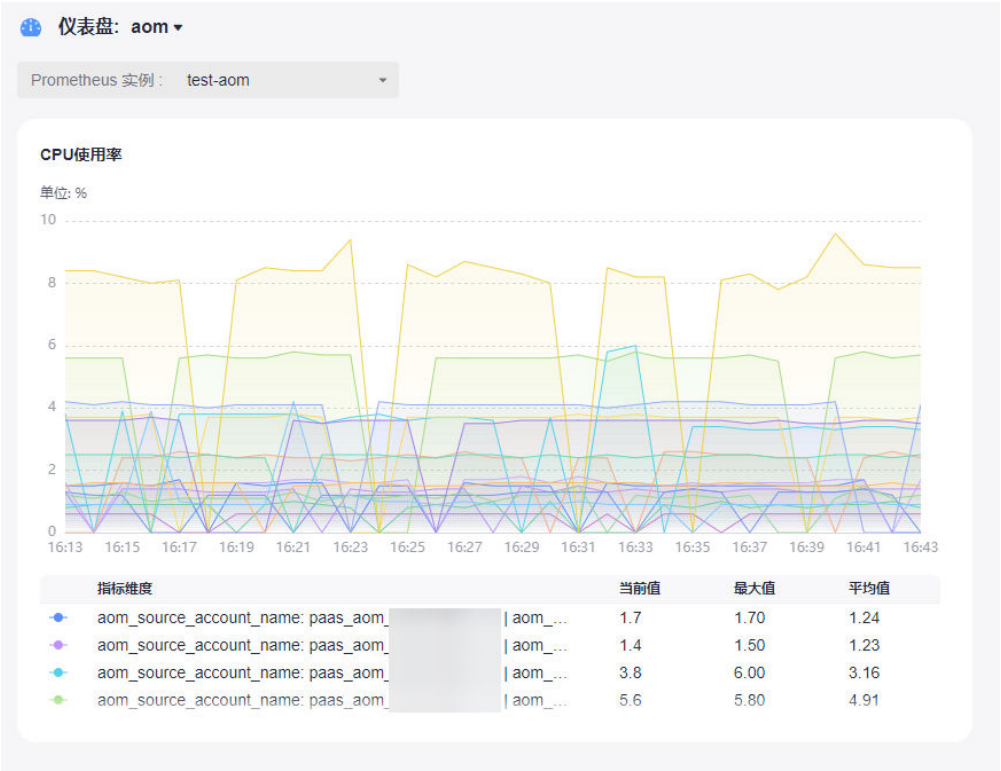
步骤4 单击指标列表右上角的📌，将图表添加至仪表盘。

1. 在下拉列表中选择仪表盘并输入图表名称。如果现有列表中的仪表盘无法满足需要，可单击“创建新的仪表盘”，新建仪表盘的操作详见创建仪表盘。

图 4-13 添加到仪表盘

2. 单击“确定”，自动跳转至仪表盘界面查看创建的图表。如图4-14所示，在仪表盘“aom”下创建了“CPU使用率”的图表，可以实时监控“paas_apm”与“paas_aom”账号下“CPU使用率”的指标值与趋势。

图 4-14 查看图表


仪表盘: aom
Prometheus 实例 : test-aom
CPU使用率
单位: %
10
8
6
4
2
0
16:13 16:15 16:17 16:19 16:21 16:23 16:25 16:27 16:29 16:31 16:33 16:35 16:37 16:39 16:41 16:43
指标维度 当前值 最大值 平均值
aom_source_account_name: paas_aom | aom_... 1.7 1.70 1.24
aom_source_account_name: paas_aom | aom_... 1.4 1.50 1.23
aom_source_account_name: paas_aom | aom_... 3.8 6.00 3.16
aom_source_account_name: paas_aom | aom_... 5.6 5.80 4.91

----结束

5 CCE 容器场景自建中间件接入

5.1 PostgreSQL Exporter 接入

操作场景

使用PostgreSQL过程中需要对PostgreSQL运行状态进行监控，以便了解PostgreSQL服务是否运行正常，及时排查PostgreSQL故障问题原因。Prometheus监控服务提供了CCE容器场景下基于Exporter的方式来监控PostgreSQL运行状态。本文介绍如何部署Exporter以及实现PostgreSQL Exporter告警接入等操作。

前提条件

- CCE服务已拥有CCE集群并已安装PostgreSQL。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将**postgres_exporter**镜像上传到SWR，具体操作请参见**使用容器引擎客户端上传镜像**。

PostgreSQL Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的集群名称，进入该集群的管理页面。

步骤3 执行以下操作完成Exporter部署。

1. 使用Secret管理PostgreSQL密码。

   在左侧导航栏中选择“工作负载”，在右上角单击“YAML创建”完成YAML配置。YAML配置说明：使用Kubernetes的Secret来管理密码并对密码进行加密处理，在启动PostgreSQL Exporter的时候直接使用Secret Key，需要调整对应的password。

   YAML 配置示例如下：

```
apiVersion: v1
kind: Secret
metadata:
    name: postgres-test
type: Opaque
stringData:
    username: postgres
    password: you-guess #对应 PostgreSQL 密码
```

2. 部署PostgreSQL Exporter。

   在左侧导航栏中选择“工作负载”，在右上角单击“YAML创建”，以YAML的方式部署Exporter。

   YAML配置示例如下（请直接复制下面的内容，根据实际业务调整相应的参数）：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: postgres-test # 根据业务需要调整成对应的名称，建议加上 PG 实例的信息
  namespace: default #需要和 postgres 的 service 在同一命名空间
  labels:
    app: postgres
    app.kubernetes.io/name: postgresql
spec:
  replicas: 1
  selector:
    matchLabels:
      app: postgres
      app.kubernetes.io/name: postgresql
  template:
    metadata:
      labels:
        app: postgres
        app.kubernetes.io/name: postgresql
    spec:
      containers:
      - name: postgres-exporter
        image: swr.cn-north-4.myhuaweicloud.com/aom-exporter/postgres-exporter:v0.8.0 # 上传至 SWR
的 postgres-exporter 镜像
        args:
          - "--web.listen-address=:9187" # Exporter 开启的端口
          - "--log.level=debug" # 日志级别
        env:
          - name: DATA_SOURCE_USER
            valueFrom:
              secretKeyRef:
                name: postgres-test # 对应上一步中的 Secret 的名称
                key: username  # 对应上一步中的 Secret Key
          - name: DATA_SOURCE_PASS
            valueFrom:
              secretKeyRef:
                name: postgres-test # 对应上一步中的 Secret 的名称
                key: password  # 对应上一步中的 Secret Key
          - name: DATA_SOURCE_URI
            value: "x.x.x.x:5432/postgres?sslmode=disable"  # 对应的连接信息
        ports:
        - name: http-metrics
          containerPort: 9187
```

3. 获取指标。

   通过“curl http://exporter:9187/metrics”无法获取Postgres实例运行时间，可以通过自定义一个queries.yaml来获取该指标。

   a. 创建一个包含queries.yaml的配置。

   b. 将配置作为Volume挂载到Exporter某个目录下。

   c. 通过extend.query-path来使用配置，将上述的Secret以及Deployment进行汇总，汇总后的YAML如下所示：

```
# 以下 document 创建一个包含自定义指标的 queries.yaml
---
apiVersion: v1
kind: ConfigMap
metadata:
  name: postgres-test-configmap
  namespace: default
data:
  queries.yaml: |
```

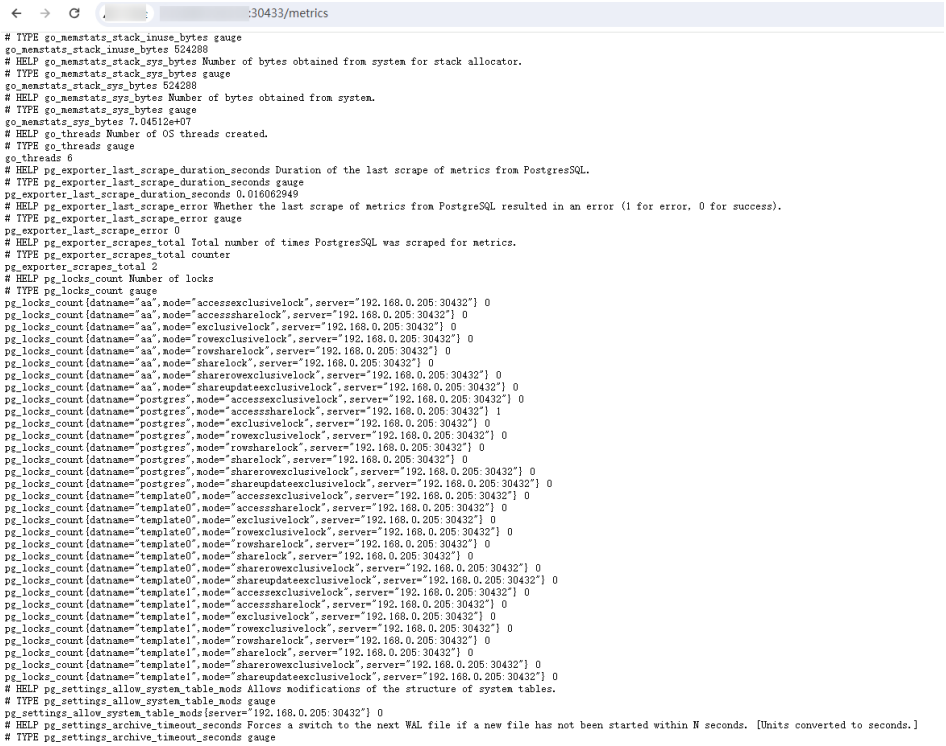
```
    pg_postmaster:
      query: "SELECT pg_postmaster_start_time as start_time_seconds from
pg_postmaster_start_time()"
      master: true
      metrics:
        - start_time_seconds:
            usage: "GAUGE"
            description: "Time at which postmaster started"

# 以下 document 挂载了 Secret 和 ConfigMap ，定义了部署 Exporter 相关的镜像等参数
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: postgres-test
  namespace: default
  labels:
    app: postgres
    app.kubernetes.io/name: postgresql
spec:
  replicas: 1
  selector:
    matchLabels:
      app: postgres
      app.kubernetes.io/name: postgresql
  template:
    metadata:
      labels:
        app: postgres
        app.kubernetes.io/name: postgresql
    spec:
      containers:
        - name: postgres-exporter
          image: wrouesnel/postgres_exporter:latest
          args:
            - "--web.listen-address=:9187"
            - "--extend.query-path=/etc/config/queries.yaml"
            - "--log.level=debug"
          env:
            - name: DATA_SOURCE_USER
              valueFrom:
                secretKeyRef:
                  name: postgres-test-secret
                  key: username
            - name: DATA_SOURCE_PASS
              valueFrom:
                secretKeyRef:
                  name: postgres-test-secret
                  key: password
            - name: DATA_SOURCE_URI
              value: "x.x.x.x:5432/postgres?sslmode=disable"
          ports:
            - name: http-metrics
              containerPort: 9187
          volumeMounts:
            - name: config-volume
              mountPath: /etc/config
      volumes:
        - name: config-volume
          configMap:
            name: postgres-test-configmap
---
apiVersion: v1
kind: Service
metadata:
  name: postgres
spec:
  type: NodePort
  selector:
```

```
    app: postgres
    app.kubernetes.io/name: postgresql
  ports:
    - protocol: TCP
      nodePort: 30433
      port: 9187
      targetPort: 9187
```

d. 访问地址：

http://{集群任意节点的公网IP}:30433/metrics，即可通过自定义的queries.yaml查询到Postgres实例启动时间指标。

图 5-1 访问地址



----结束

添加采集任务

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

说明

如下指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: postgres-exporter
  namespace: default
spec:
  namespaceSelector:
    matchNames:
      - default # exporter 所在的命名空间
  podMetricsEndpoints:
  - interval: 30s
    path: /metrics
```

```
  port: http-metrics
selector:
  matchLabels:
    app: postgres
```

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

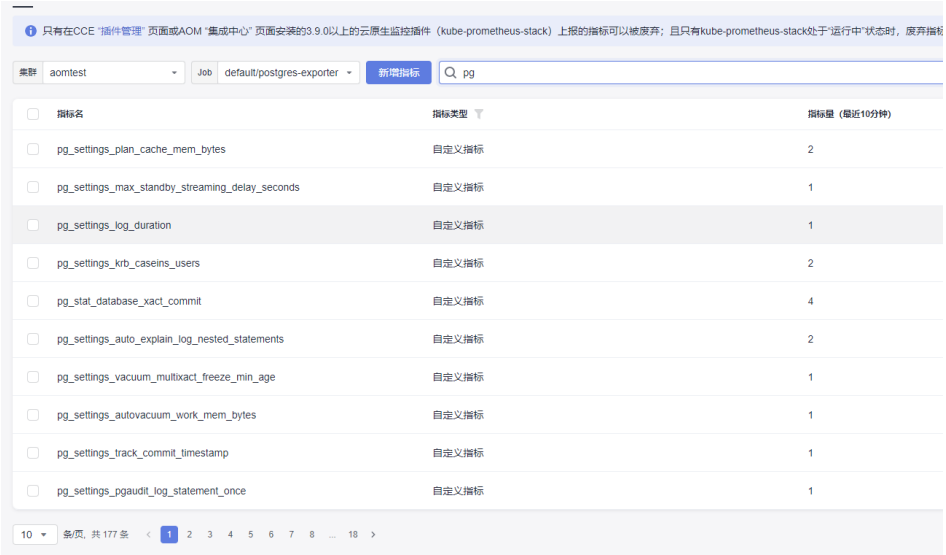
步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择对应集群。

步骤5 选择Job：{namespace}/postgres-exporter，可以查询到pg开头的postgresql指标。

图 5-2 搜索指标



----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见**创建仪表盘**。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见**添加图表至仪表盘**。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见**创建指标告警规则**。

5.2 MySQL Exporter 接入

操作场景

MySQL Exporter专门为采集MySQL数据库监控指标而设计开发，通过Exporter上报核心的数据库指标，用于异常报警和监控大盘展示。目前，Exporter支持5.6版本或以上版本的MySQL。在MySQL低于5.6版本时，部分监控指标可能无法被采集。

说明

为了方便安装管理Exporter，推荐使用CCE进行统一管理。

前提条件

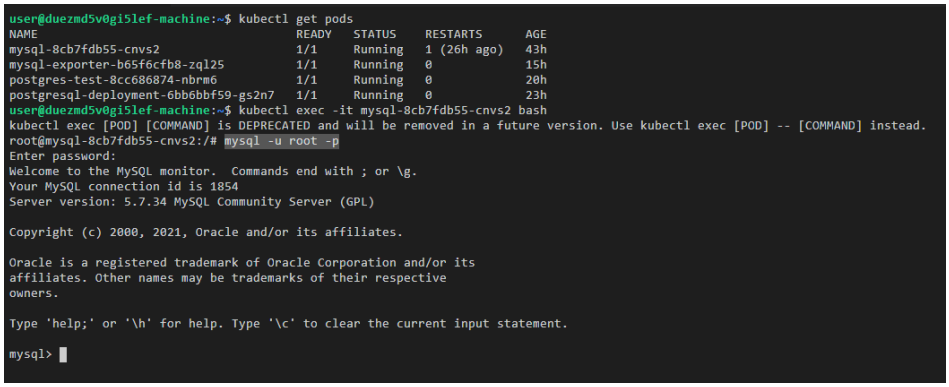
- CCE服务已拥有CCE集群并已安装MySQL。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将对应**mysql_exporter**镜像上传到SWR，具体操作请参见**使用容器引擎客户端上传镜像**。

数据库授权

步骤1 登录集群执行以下命令：

```
kubectl exec -it ${mysql_podname} bash
mysql -u root -p
```

图 5-3 执行命令



步骤2 登录数据库，执行以下命令：

```
CREATE USER 'exporter'@'x.x.x.x(hostip)' IDENTIFIED BY 'xxxx(password)' WITH MAX_USER_CONNECTIONS
3;
GRANT PROCESS, REPLICATION CLIENT, SELECT ON *.* TO 'exporter'@'x.x.x.x(hostip)';
```

步骤3 验证授权是否成功。

输入以下命令查询sql，查看是否有exporter的数据，host为mysql所在节点的IP。

```
select user,host from mysql.user;
```

图 5-4 查询 sql

```
mysql> select user,host from mysql.user;
+---------------+---------------+
| user          | host          |
+---------------+---------------+
| root          | %             |
| exporter      | 192.168.0.205 |
| mysql.session | localhost     |
| mysql.sys     | localhost     |
| root          | localhost     |
+---------------+---------------+
5 rows in set (0.00 sec)

mysql>
```

----结束

MySQL Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的集群名称，进入该集群的管理页面。

步骤3 执行以下操作完成Exporter部署。

1. 使用Secret管理MySQL连接串：

   在左侧导航栏中选择“配置与密钥”，在右上角单击“YAML创建”，输入以下yml文件，密码是按照Opaque加密过的。

```
apiVersion: v1
kind: Secret
metadata:
    name: mysql-secret
    namespace: default
type: Opaque
stringData:
    datasource: "user:password@tcp(ip:port)/"  #对应 MySQL 连接串信息，需要加密
```

   说明

   配置密钥的详细操作参见**创建密钥**。

2. 部署MySQL Exporter。

   在左侧导航栏中选择“工作负载”，在右上角单击“创建负载”，选择“负载类型”为无状态工作负载Deployment，选择需要的命名空间部署MySQL Exporter。如果以YAML的方式部署Exporter，YAML配置示例如下：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    k8s-app: mysql-exporter # 根据业务需要调整成对应的名称，建议加上MySQL实例的信息, 如
ckafka-2vrgx9fd-mysql-exporter
  name: mysql-exporter # 根据业务需要调整成对应的名称，建议加上MySQL实例的信息, 如
ckafka-2vrgx9fd-mysql-exporter
  namespace: default #需要和CCE集群中安装的MySQL命名空间一致
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: mysql-exporter # 根据业务需要调整成对应的名称，建议加上MySQL实例的信息, 如
```

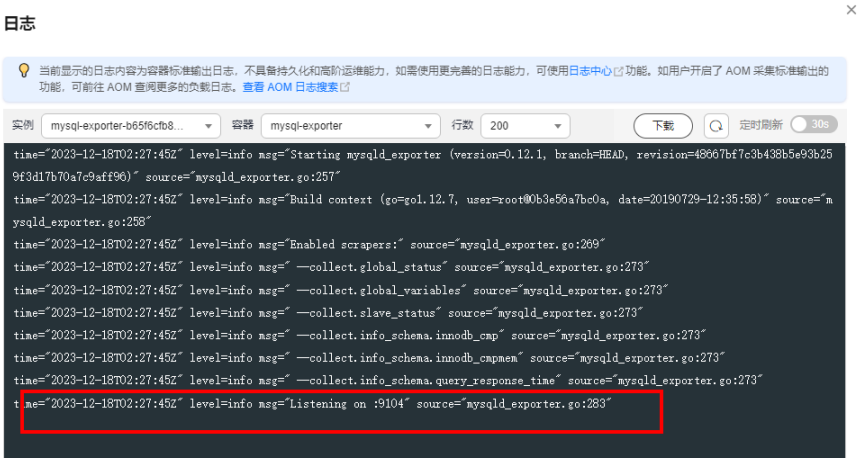
```
ckafka-2vrgx9fd-mysql-exporter
  template:
    metadata:
      labels:
        k8s-app: mysql-exporter # 根据业务需要调整成对应的名称，建议加上MySQL实例的信息, 如
ckafka-2vrgx9fd-mysql-exporter
    spec:
      containers:
      - env:
        - name: DATA_SOURCE_NAME
          valueFrom:
            secretKeyRef:
              name: mysql-secret
              key: datasource
        image: swr.cn-north-4.myhuaweicloud.com/aom-exporter/mysqld-exporter:v0.12.1
        imagePullPolicy: IfNotPresent
        name: mysql-exporter
        ports:
        - containerPort: 9104
          name: metric-port
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
    dnsPolicy: ClusterFirst
    imagePullSecrets:
    - name: default-secret
    restartPolicy: Always
    schedulerName: default-scheduler
    securityContext: {}
    terminationGracePeriodSeconds: 30
---
apiVersion: v1
kind: Service
metadata:
  name: mysql-exporter
spec:
  type: NodePort
  selector:
    k8s-app: mysql-exporter
  ports:
    - protocol: TCP
      nodePort: 30337
      port: 9104
      targetPort: 9104
```

说明

更多Exporter详细参数介绍请参见**mysql-exporter**。

3. 验证MySQL Exporter是否部署成功。
   a. 在工作负载列表中“无状态负载”页签下，单击**步骤3.2**创建的无状态工作负载的名称，在实例列表中单击操作列下的“更多 > 日志”，可以查看到Exporter成功启动并暴露对应的访问地址。

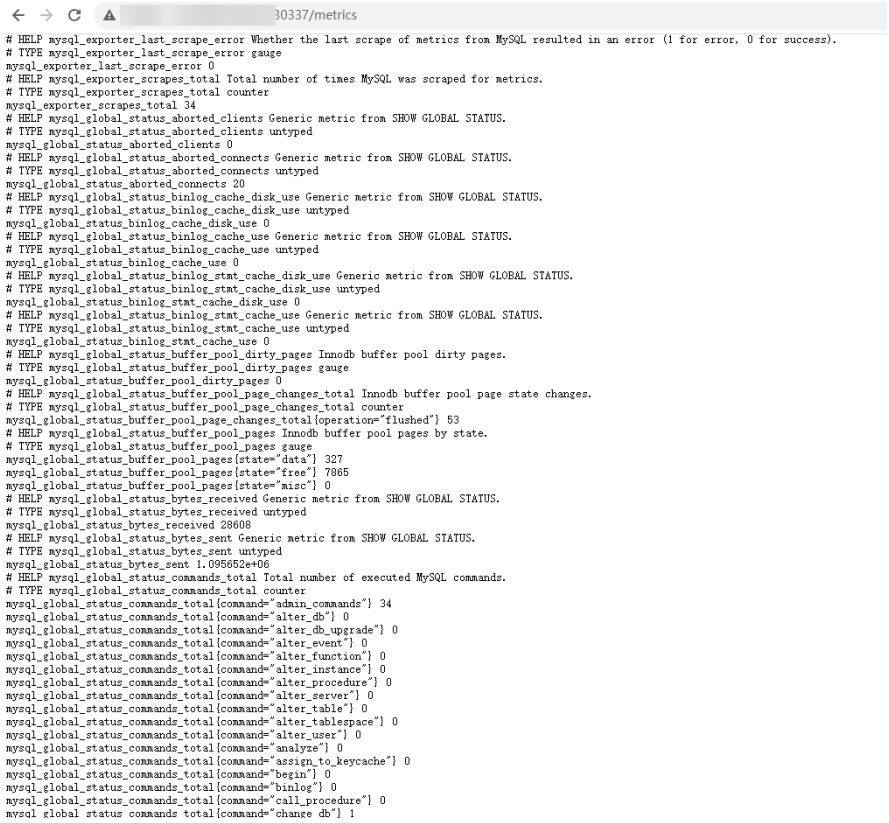
图 5-5 查看日志



b. 验证。有以下三种方法进行验证：

- 登录集群节点执行如下任意一种命令：
```
curl http://{集群IP}:9104/metrics
curl http://{集群任意节点私有IP}:30337/metrics
```
- 在实例列表中单击操作列下的“更多 > 远程登录”，执行如下命令：
```
curl http://localhost:9104/metric
```
- 访问：http://{集群任意节点的公网IP}:30337/metrics。

图 5-6 访问地址



----结束

采集 CCE 集群的业务数据

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

配置信息如下：

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: mysql-exporter
  namespace: default
spec:
  namespaceSelector:
    matchNames:
      - default # exporter 所在的命名空间
  podMetricsEndpoints:
  - interval: 30s
    path: /metrics
    port: metric-port
  selector:
    matchLabels:
      k8s-app: mysql-exporter
```

说明

指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择对应集群。

步骤5 选择Job：{namespace}/mysql-exporter，可以查询到mysql开头的自定义指标。

----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见**创建仪表盘**。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见**添加图表至仪表盘**。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见**创建指标告警规则**。

5.3 Kafka Exporter 接入

操作场景

使用Kafka过程中需要对Kafka运行状态进行监控，例如集群状态、消息消费情况是否有积压等。Prometheus监控服务提供了CCE容器场景下基于Exporter的方式来监控Kafka运行状态。本文介绍如何部署Kafka Exporter以及实现Kafka Exporter告警接入等操作。

说明

为了方便安装管理Exporter，推荐使用CCE进行统一管理。

前提条件

- CCE服务已拥有CCE集群并已安装Kafka。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将对应**kafka_exporter**镜像上传到SWR，具体操作请参见**使用容器引擎客户端上传镜像**。

Kafka Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的集群名称，进入该集群的管理页面。

步骤3 执行以下操作完成Exporter部署。

1. 部署Kafka Exporter。

   在左侧导航栏中选择“工作负载”，在右上角单击“创建负载”，选择“负载类型”为无状态工作负载Deployment，选择需要的命名空间部署Kafka Exporter。如果以YAML的方式部署Exporter，YAML配置示例如下：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    k8s-app: kafka-exporter # 根据业务需要调整成对应的名称，建议加上Kafka实例的信息, 如
ckafka-2vrgx9fd-kafka-exporter
  name: kafka-exporter # 根据业务需要调整成对应的名称，建议加上Kafka实例的信息, 如
ckafka-2vrgx9fd-kafka-exporter
  namespace: default #已存在集群的namespace
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: kafka-exporter # 根据业务需要调整成对应的名称，建议加上Kafka实例的信息, 如
ckafka-2vrgx9fd-kafka-exporter
  template:
    metadata:
      labels:
        k8s-app: kafka-exporter # 根据业务需要调整成对应的名称，建议加上Kafka实例的信息, 如
ckafka-2vrgx9fd-kafka-exporter
    spec:
      containers:
      - args:
        - --kafka.server=120.46.215.4:30092 # 对应Kafka实例的地址信息
```

```
      image: swr.cn-north-4.myhuaweicloud.com/mall-swarm-demo/kafka-exporter:latest
      imagePullPolicy: IfNotPresent
      name: kafka-exporter
      ports:
      - containerPort: 9308
        name: metric-port  # 这个名称在配置抓取任务的时候需要
      securityContext:
        privileged: false
      terminationMessagePath: /dev/termination-log
      terminationMessagePolicy: File
    dnsPolicy: ClusterFirst
    imagePullSecrets:
    - name: default-secret
    restartPolicy: Always
    schedulerName: default-scheduler
    securityContext: {}
    terminationGracePeriodSeconds: 30
---
apiVersion: v1
kind: Service
metadata:
  name: kafka-exporter
spec:
  type: NodePort
  selector:
    k8s-app: kafka-exporter
  ports:
    - protocol: TCP
      nodePort: 30091
      port: 9308
      targetPort: 9308
```

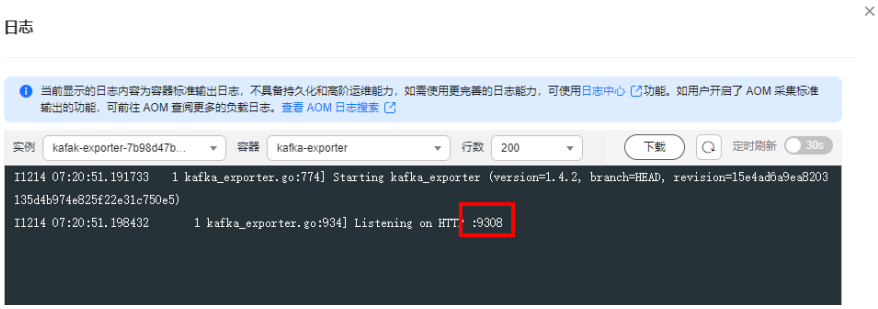
说明

更多 Exporter详细参数介绍请参见 **kafka-exporter**。

2. 验证Kafka Exporter是否部署成功。

   a. 在工作负载列表中“无状态负载”页签下，单击**步骤3.1**创建的无状态工作负载，在实例列表中单击操作列下的“更多 > 日志”，可以查看到Exporter成功启动并暴露对应的访问地址。

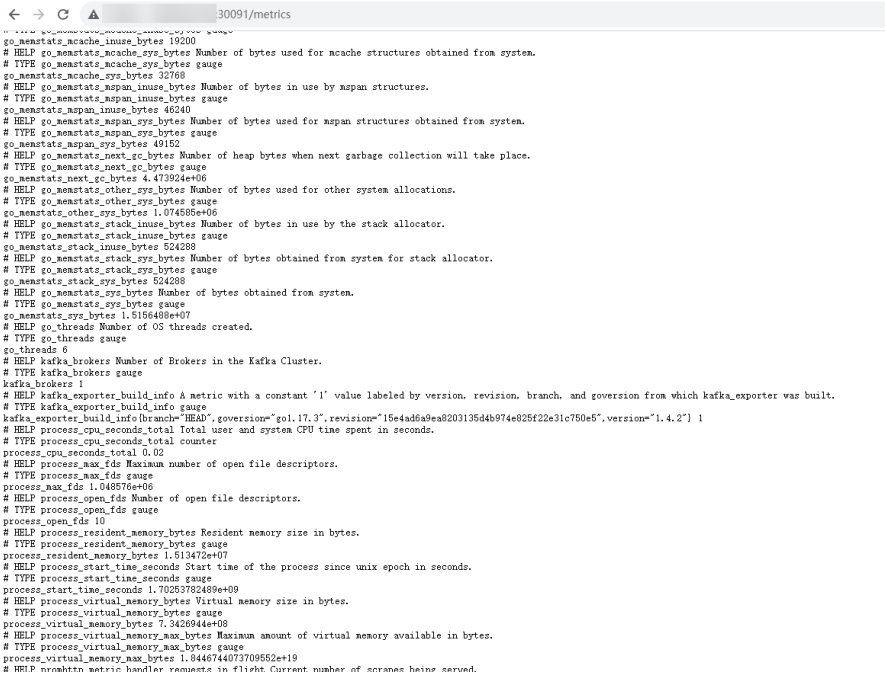
   图 5-7 查看日志



   b. 验证。有以下三种方法进行验证：

      - 登录集群节点执行如下任意一种命令：
        ```
        curl http://{集群IP}:9308/metrics
        curl http://{集群任意节点私有IP}:30091/metrics
        ```
      - 在实例列表中单击操作列下的“更多 > 远程登录”，执行如下命令：
        ```
        curl http://localhost:9308/metric
        ```
      - 访问：http://{集群任意节点的公网IP}:30091/metrics。

图 5-8 访问地址



----结束

采集 CCE 集群的业务数据

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

说明

如下指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

配置信息如下：

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: kafka-exporter
  namespace: default
spec:
  namespaceSelector:
    matchNames:
      - default # exporter 所在的命名空间
  podMetricsEndpoints:
  - interval: 30s
    path: /metrics
    port: metric-port
  selector:
    matchLabels:
      k8s-app: kafka-exporter
```

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择对应集群。

步骤5 选择Job：{namespace}/kafka-exporter,可以查询到kafka开头的自定义指标。

----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见创建仪表盘。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见添加图表至仪表盘。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见创建指标告警规则。

5.4 Memcached Exporter 接入

操作场景

使用Memcached过程中需要对Memcached运行状态进行监控，以便了解Memcached服务是否运行正常，排查Memcached故障等。Prometheus监控服务提供了CCE容器场景下基于Exporter的方式来监控Memcached运行状态。本文为您介绍如何使用Prometheus监控服务Memcached。

说明

为了方便安装管理Exporter，推荐使用CCE统一管理。

前提条件

- CCE服务已拥有CCE集群，已安装Memcached。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将**memcached_exporter**镜像上传到SWR，具体操作请参见使用容器引擎客户端上传镜像。

Memcached Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的CCE集群名称，进入该集群的管理页面。

步骤3 执行以下操作完成Exporter部署。

1. 配置密钥。

   在左侧导航栏中选择“配置与密钥”，单击页面右上角“YAML创建”。YAML配置示例如下：

```
apiVersion: v1
kind: Secret
metadata:
    name: memcached-exporter-secret
    namespace: default
type: Opaque
stringData:
    memcachedURI: 120.46.215.4:11211   # Memcached地址
```

   说明

   - Memcached 连接串的格式为：http://{ip}:{port}。
   - 配置密钥的详细操作参见创建密钥。

2. 部署Memcached Exporter。

   在左侧导航栏中选择“工作负载”，选择“无状态负载”页签，单击右上角的“YAML创建”，以YAML的方式部署Exporter。

   YAML配置示例如下：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    k8s-app: memcached-exporter # 根据业务需要调整
  name: memcached-exporter # 根据业务需要调整
  namespace: default
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: memcached-exporter # 根据业务需要调整
  template:
    metadata:
      labels:
        k8s-app: memcached-exporter # 根据业务需要调整
    spec:
      containers:
      - env:
          - name: Memcached_Url
            valueFrom:
              secretKeyRef:
                name: memcached-exporter-secret # 对应上一步中的 Secret 的名称
                key: memcachedURI # 对应上一步中的 Secret Key
          - name: Memcached_ALL
            value: "true"
        image: swr.cn-east-3.myhuaweicloud.com/aom-org/bitnami/memcached-exporter:0.13.0 #前提条件中上传到swr中的镜像
        imagePullPolicy: IfNotPresent
        name: memcached-exporter
        ports:
        - containerPort: 9150
          name: metric-port
        securityContext:
          privileged: false
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
    dnsPolicy: ClusterFirst
    imagePullSecrets:
    - name: default-secret
    restartPolicy: Always
    schedulerName: default-scheduler
    securityContext: {}
    terminationGracePeriodSeconds: 30
```

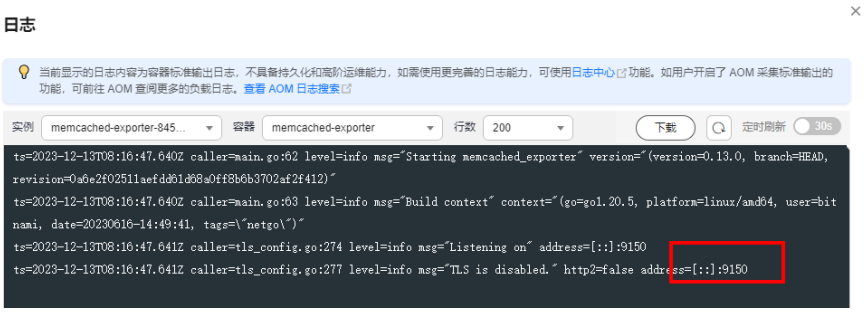
```
---
apiVersion: v1
kind: Service
metadata:
  name: memcached-exporter
spec:
  type: NodePort
  selector:
    k8s-app: memcached-exporter
  ports:
    - protocol: TCP
      nodePort: 30122
      port: 9150
      targetPort: 9150
```

说明

更多Exporter详细参数介绍请参见 **memcached_exporter**。

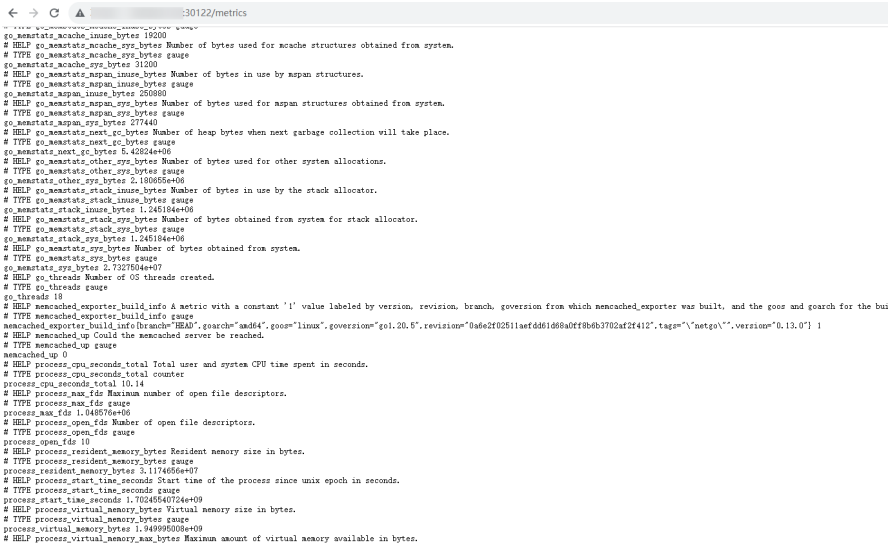
3. 验证Memcached Exporter是否部署成功。
   a. 在工作负载列表中“无状态负载”页签下，单击**步骤3.2**创建的无状态工作负载的名称，在实例列表中单击操作列下的“更多 > 日志”，可以查看到Exporter成功启动并暴露访问地址。

图 5-9 查看日志



   b. 验证。有以下三种方法进行验证：
      - 登录集群节点执行如下任意一种命令：
        ```
        curl http://{集群IP}:9150/metrics
        curl http://{集群任意节点私有IP}:30122/metrics
        ```
      - 访问地址：http://{集群任意节点的公网IP}:30122/metrics。

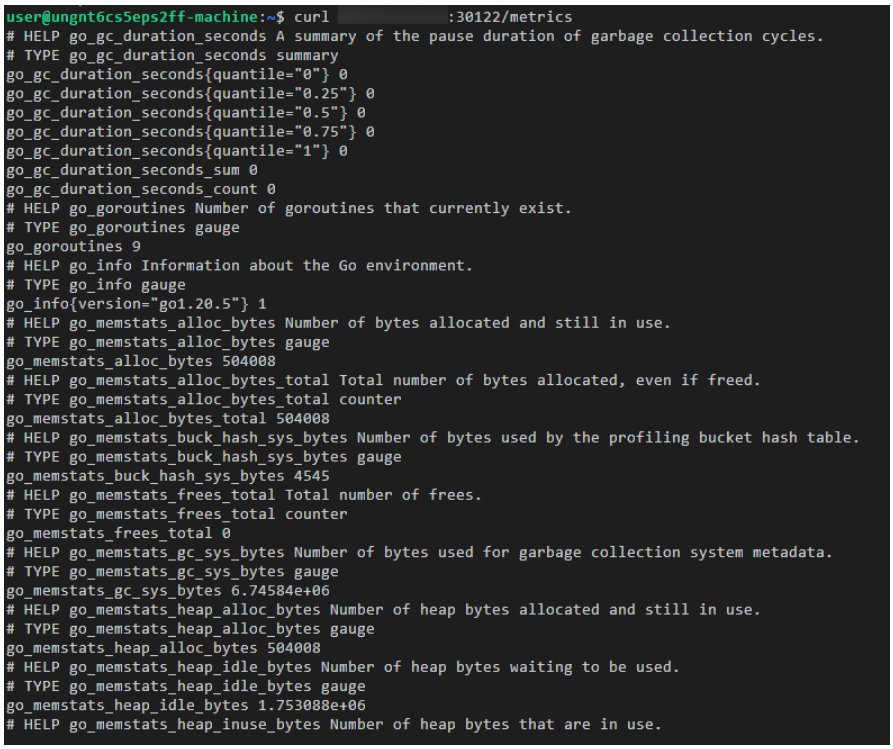
图 5-10 访问地址



- 在实例列表中单击操作列下的“更多 > 远程登录”，执行如下命令。

```
curl http://localhost:9150/metric
```

图 5-11 执行命令



----结束

添加采集任务

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

说明

如下示例中指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: memcached-exporter
  namespace: default
spec:
  namespaceSelector:
    matchNames:
      - default # exporter所在的命名空间
  podMetricsEndpoints:
  - interval: 30s
    path: /metrics
    port: metric-port
  selector:
    matchLabels:
      k8s-app: memcached-exporter
```

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择集群。

步骤5 选择Job：{namespace}/memcached-exporter，可以查询到go_memstats开头的memcached指标。

----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见创建仪表盘。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见添加图表至仪表盘。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见创建指标告警规则。

5.5 MongoDB Exporter 接入

操作场景

使用MongoDB过程中需要对MongoDB运行状态进行监控，以便了解MongoDB服务是否运行正常，排查MongoDB故障问题原因。Prometheus监控服务提供了CCE容器场景下基于Exporter的方式来监控MongoDB运行状态。本文介绍如何部署Exporter以及实现MongoDB Exporter告警接入等操作。

说明

为了方便安装管理Exporter，推荐使用CCE进行统一管理。

前提条件

- CCE服务已拥有CCE集群，已安装MongoDB。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将**mongodb_exporter**镜像上传到SWR，具体操作请参见**使用容器引擎客户端上传镜像**。

MongoDB Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的CCE集群名称，进入该集群的管理页面。

步骤3 执行以下操作完成Exporter部署。

1. 配置密钥。

   在左侧导航栏中选择“配置与密钥”，在页面右上角单击“YAML创建”。YAML配置示例如下：

```
apiVersion: v1
kind: Secret
metadata:
    name: mongodb-secret-test
    namespace: default
type: Opaque
stringData:
    datasource: "mongodb://{user}:{passwd}@{host1}:{port1},{host2}:{port2},{host3}:{port3}/admin"  #
对应连接URI
```

   说明

   - 密码已按照Opaque加密。
   - 配置密钥的详细操作参见**创建密钥**。

2. 部署MongoDB Exporter。

   在左侧导航栏中选择“工作负载”，在右上角单击“创建负载”，选择“负载类型”为无状态工作负载Deployment，选择需要的命名空间部署MongoDB Exporter。如果以YAML的方式部署Exporter，YAML配置示例如下：

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
```

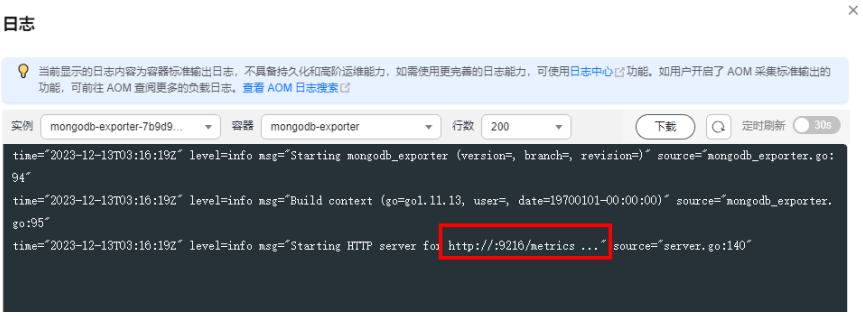
```
    k8s-app: mongodb-exporter # 根据业务需要调整，建议加上MongoDB实例的信息
  name: mongodb-exporter # 根据业务需要调整，建议加上MongoDB实例的信息
  namespace: default #需要和CCE集群中安装的MongoDB命名空间一致
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: mongodb-exporter # 根据业务需要调整，建议加上MongoDB实例的信息
  template:
    metadata:
      labels:
        k8s-app: mongodb-exporter # 根据业务需要调整，建议加上MongoDB实例的信息
    spec:
      containers:
        - args:
            - --collect.database       # 启用数据库指标采集
            - --collect.collection     # 启用集合指标采集
            - --collect.topmetrics     # 启用数据库表头指标信息采集
            - --collect.indexusage     # 启用索引使用统计信息采集
            - --collect.connpoolstats  # 启动MongoDB连接池统计信息采集
          env:
            - name: MONGODB_URI
              valueFrom:
                secretKeyRef:
                  name: mongodb-secret-test
                  key: datasource
          image: swr.cn-north-4.myhuaweicloud.com/mall-swarm-demo/mongodb-exporter:0.10.0
          imagePullPolicy: IfNotPresent
          name: mongodb-exporter
          ports:
            - containerPort: 9216
              name: metric-port  # 这个名称在配置抓取任务的时候需要
          securityContext:
            privileged: false
          terminationMessagePath: /dev/termination-log
          terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      imagePullSecrets:
        - name: default-secret
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: { }
      terminationGracePeriodSeconds: 30
---
apiVersion: v1
kind: Service
metadata:
  name: mongodb-exporter
spec:
  type: NodePort
  selector:
    k8s-app: mongodb-exporter
  ports:
    - protocol: TCP
      nodePort: 30003
      port: 9216
      targetPort: 9216
```

说明

更多Exporter详细参数介绍请参见**mongodb_exporter**。

3. 验证MongoDB Exporter是否部署成功。
   a. 在工作负载列表中“无状态负载”页签下，单击**步骤3.2**创建的无状态工作负载的名称，在实例列表中单击操作列下的“更多 > 日志”，可以查看到Exporter成功启动并暴露访问地址。

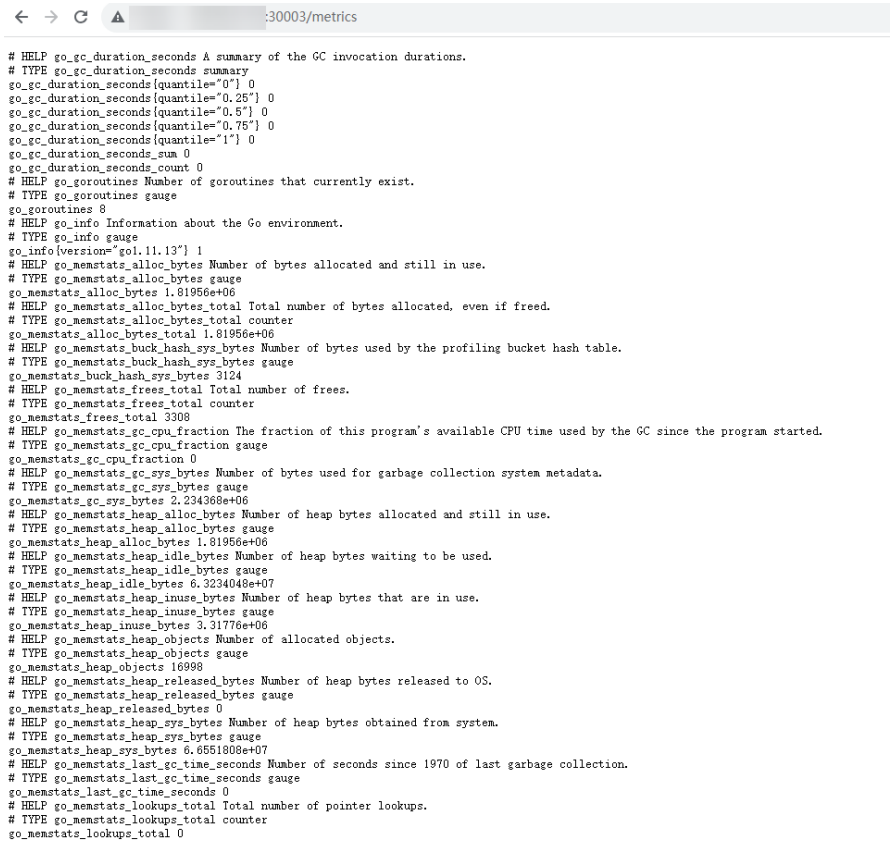
图 5-12 查看日志



b. 验证。有以下三种方法进行验证：

- 登录集群节点执行如下任意一种命令：
```
curl http://{集群IP}:9216/metrics
curl http://{集群任意节点私有IP}:30003/metrics
```
- 访问地址：http://{集群任意节点的公网IP}:30003/metrics。

图 5-13 访问地址



- 在实例列表中单击操作列下的“更多 > 远程登录”，执行如下命令。
```
curl http://localhost:9216/metric
```

----结束

采集 CCE 集群的业务数据

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

说明

如下示例中指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: mongodb-exporter
  namespace: default
spec:
  namespaceSelector:
    matchNames:
      - default # exporter所在的命名空间
  podMetricsEndpoints:
  - interval: 30s
    path: /metrics
    port: metric-port
  selector:
    matchLabels:
      k8s-app: mongodb-exporter
```

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择集群。

步骤5 选择job：{namespace}/MongoDB-exporter,可以查询到mongodb开头的自定义指标。

----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见**创建仪表盘**。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见**添加图表至仪表盘**。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见**创建指标告警规则**。

5.6 ElasticSearch Exporter 接入

操作场景

使用ElasticSearch过程中需要对ElasticSearch运行状态进行监控，例如集群及索引状态等。Prometheus监控服务提供了CCE容器场景下基于Exporter的方式来监控ElasticSearch运行状态。本文介绍如何部署ElasticSearch Exporter以及实现ElasticSearch Exporter告警接入等操作。

说明

为了方便安装管理Exporter，推荐使用CCE进行统一管理。

前提条件

- CCE服务已拥有CCE集群，已安装ElasticSearch。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将**elasticsearch_exporter**镜像上传到SWR，具体操作请参见**使用容器引擎客户端上传镜像**。

ElasticSearch Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的CCE集群名称，进入该集群的管理页面。

步骤3 执行以下操作完成Exporter部署。

1. 配置密钥。

   在左侧导航栏中选择“配置与密钥”，单击页面右上角“YAML创建”，YAML配置示例如下：

```
apiVersion: v1
kind: Secret
metadata:
  name: es-secret-test
  namespace: default
type: Opaque
stringData:
  esURI: http://124.70.14.51:30920  #对应 ElasticSearch 的 URI，IP为集群IP或集群任意节点IP
```

   说明

   - ElasticSearch连接串的格式为 <proto>://<user>:<password>@<host>:<port>，例如http://admin:pass@localhost:9200。也可以不设置密码，例如设置为：http://10.247.43.50:9200。
   - 密码已按照Opaque加密。
   - 配置密钥的详细操作参见**创建密钥**。

2. 部署ElasticSearch Exporter。

   在左侧导航栏中选择“工作负载”，在右上角单击“创建负载”，选择“负载类型”为无状态工作负载Deployment，选择需要的命名空间部署ElasticSearch Exporter。如果以YAML的方式部署Exporter，YAML配置示例如下：

```
apiVersion: apps/v1
kind: Deployment
```

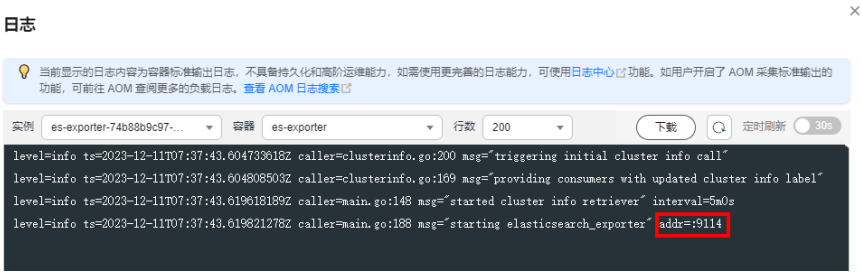
```
metadata:
  labels:
    k8s-app: es-exporter # 根据业务需要调整
  name: es-exporter # 根据业务需要调整
  namespace: default # 选择一个适合的 namespace 来部署 Exporter，如果没有需要新建一个
spec:
  replicas: 1
  selector:
    matchLabels:
      k8s-app: es-exporter # 根据业务需要调整
  template:
    metadata:
      labels:
        k8s-app: es-exporter # 根据业务需要调整
    spec:
      containers:
      - env:
          - name: ES_URI
            valueFrom:
              secretKeyRef:
                name: es-secret-test # 对应上一步中的 Secret 的名称
                key: esURI # 对应上一步中的 Secret Key
          - name: ES_ALL
            value: "true"
        image: swr.cn-north-4.myhuaweicloud.com/mall-swarm-demo/es-exporter:1.1.0
        imagePullPolicy: IfNotPresent
        name: es-exporter
        ports:
        - containerPort: 9114
          name: metric-port
        securityContext:
          privileged: false
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      imagePullSecrets:
      - name: default-secret
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
---
apiVersion: v1
kind: Service
metadata:
  name: es-exporter
  name-space: default # 与Exporter部署的namespace相同
spec:
  type: NodePort
  selector:
    k8s-app: es-exporter
  ports:
    - protocol: TCP
      nodePort: 30921
      port: 9114
      targetPort: 9114
```

说明

上述示例通过ES_ALL采集了所有ElasticSearch的监控项，可以通过对应的参数进行调整，Exporter更多详细的参数请参见 **elasticsearch_exporter**。

3. 验证ElasticSearch Exporter是否部署成功。
   a. 在工作负载列表中“无状态负载”页签下，单击**步骤3.2**创建的无状态工作负载的名称，在实例列表中单击操作列下的“更多 > 日志”，可以查看到Exporter成功启动并暴露访问地址。

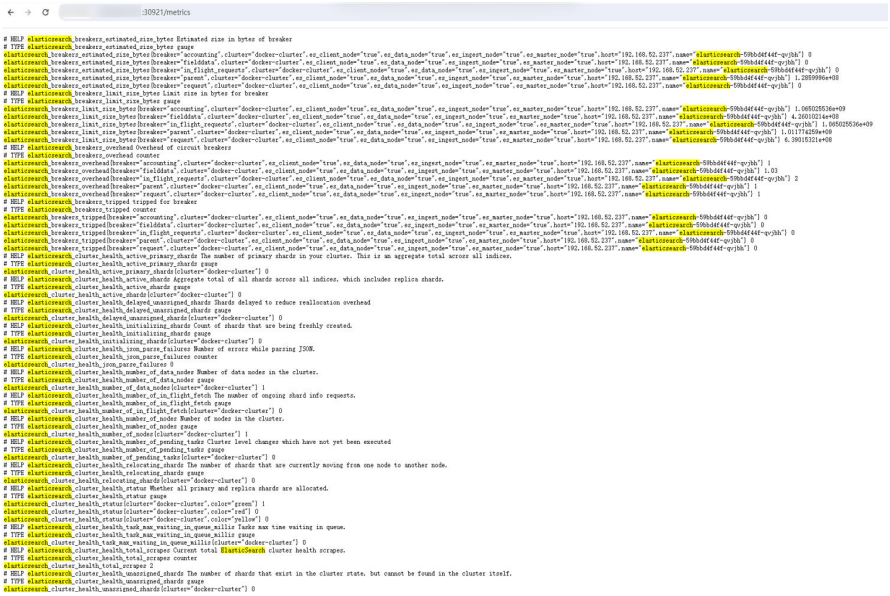
图 5-14 查看日志



b. 验证。有以下三种方法进行验证：

- 登录集群节点执行如下任意一种命令：
  ```
  curl http://{集群IP}:9114/metrics
  curl http://{集群任意节点私有IP}:30921/metrics
  ```
- 访问地址：http://{集群任意节点的公网IP}:30921/metrics。

  图 5-15 访问地址

- 在实例列表中单击操作列下的“更多 > 远程登录”，执行如下命令。
  ```
  curl http://localhost:9114/metric
  ```

----结束

采集 CCE 集群的业务数据

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

说明

如下示例中指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
```

```
  name: elasticSearch-exporter
  namespace: default
spec:
  namespaceSelector: # 选择监控Exporter部署所在的namespace
    matchNames:
      - default # exporter所在的命名空间
  podMetricsEndpoints:
  - interval: 30s # 设置指标采集周期
    path: /metrics # 填写Prometheus Exporter对应的Path的值，默认/metrics
    port: metric-port # 填写Prometheus Exporter对应YAML的ports的name
  selector: # 填写要监控Exporter Pod的Label标签，以定位目标Exporter
    matchLabels:
      k8s-app: elasticSearch-exporter
```

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择集群。

步骤5 选择Job：{namespace}/elasticsearch-exporter,可以查询到elasticsearch开头的自定义指标。

----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见创建仪表盘。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见添加图表至仪表盘。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见创建指标告警规则。

5.7 Redis Exporter 接入

操作场景

使用数据库Redis过程中需要对Redis运行状态进行监控，以便了解Redis服务是否运行正常，及时排查Redis故障等。Prometheus监控服务提供了CCE容器场景下基于Exporter的方式来监控Redis运行状态。本文为您介绍如何使用Prometheus监控Redis。

说明

为了方便安装管理Exporter，推荐使用云容器引擎CCE进行统一管理。

前提条件

- CCE服务已拥有CCE集群，已安装Redis。
- 服务已接入可观测Prometheus监控并接入CCE集群，具体请参见**Prometheus实例 for CCE**。
- 已将**redis_exporter**镜像上传到SWR，具体操作请参见**使用容器引擎客户端上传镜像**。

Redis Exporter 部署

步骤1 登录CCE控制台。

步骤2 单击已接入的CCE集群名称，进入该集群的管理页面。

步骤3 执行以下步骤完成Exporter部署。

1. 在左侧导航栏中选择“配置与密钥”，选择“密钥”页签，单击页面右上角“YAML创建”，YAML配置示例如下：

```
apiVersion: v1
kind: Secret
metadata:
  name: redis-secret-test
  namespace: default # 与Exporter部署的namespace相同
type: Opaque
stringData:
  password: redis123  #对应 Redis 密码
```

说明

- 密码已按照Opaque加密。
- 配置密钥的详细操作参见**创建密钥**。

2. 部署Redis Exporter。

   在左侧菜单栏中选择“工作负载”，选择“无状态负载”页签，单击页面右上角“YAML创建”，选择命名空间来进行部署服务。可以通过控制台的方式创建，如果以YAML的方式部署Exporter，YAML配置示例如下：

```
apiVersion: apps/v1
kind: Deployment
metadata:
 labels:
  k8s-app: redis-exporter # 根据业务需要调整，建议加上 Redis 实例的信息，如crs-66e112fp-redis-exporter
 name: redis-exporter # 根据业务需要调整，建议加上 Redis 实例的信息，如crs-66e112fp-redis-exporter
 namespace: default # 选择一个适合的 namespace 来部署 Exporter，如果没有需要新建一个 namespace
spec:
 replicas: 1
 selector:
  matchLabels:
   k8s-app: redis-exporter # 根据业务需要调整成对应的名称，建议加上 Redis 实例的信息，如crs-66e112fp-redis-exporter
 template:
  metadata:
   labels:
    k8s-app: redis-exporter # 根据业务需要调整成对应的名称，建议加上 Redis 实例的信息，如crs-66e112fp-redis-exporter
  spec:
```

```
      containers:
      - env:
        - name: REDIS_ADDR
          value: 120.46.215.4:30379 # 对应 Redis 的 ip:port
        - name: REDIS_PASSWORD
          valueFrom:
            secretKeyRef:
              name: redis-secret-test # 对应上一步的 Secret 的名称
              key: password # 对应上一步中的 Secret Key
        image: swr.cn-north-4.myhuaweicloud.com/mall-swarm-demo/redis-exporter:v1.32.0  # 替换为您
上传到 SWR 的镜像地址
        imagePullPolicy: IfNotPresent
        name: redis-exporter
        ports:
        - containerPort: 9121
          name: metric-port  # 这个名称在配置采集任务的时候需要
        securityContext:
          privileged: false
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      imagePullSecrets:
      - name: default-secret
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
---
apiVersion: v1
kind: Service
metadata:
  name: redis-exporter
  name-space: default # 与Exporter部署的namespace相同
spec:
  type: NodePort
  selector:
    k8s-app: redis-exporter
  ports:
    - protocol: TCP
      nodePort: 30378
      port: 9121
      targetPort: 9121
```

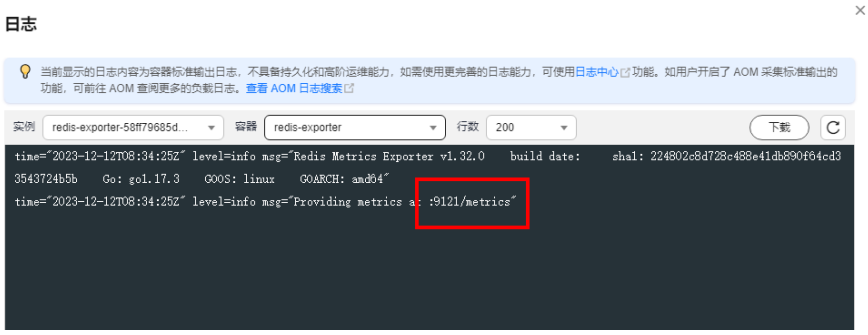
说明

更多Exporter详细参数介绍请参见 **redis_exporter**。

3. 验证Redis Exporter是否部署成功。

   a. 在工作负载列表中“无状态负载”页签下，单击**步骤3.2**创建的无状态工作负载的名称，在实例列表中单击操作列下的“更多 > 日志”，可以查看到Exporter成功启动并暴露访问地址。

图 5-16 查看日志

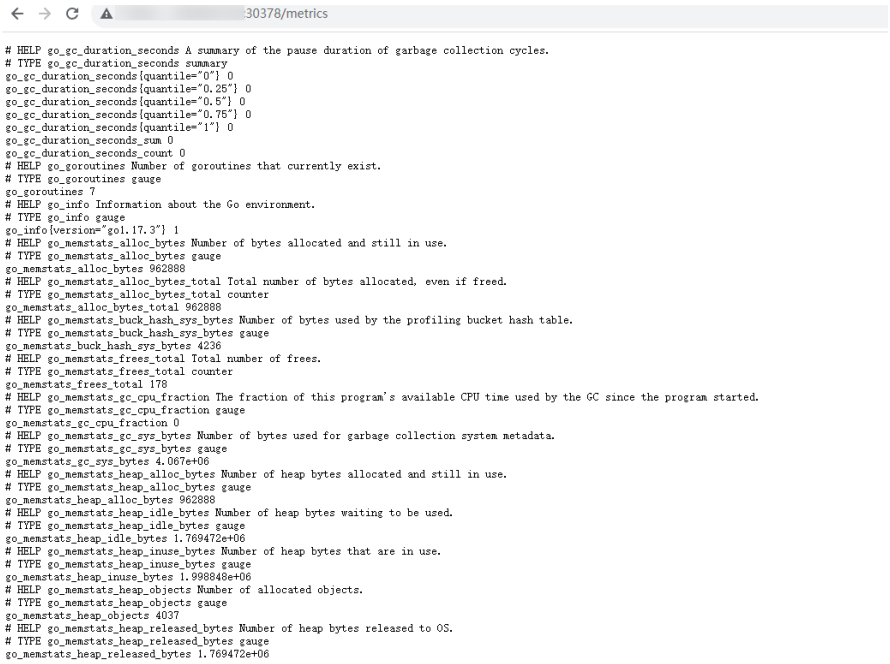


b. 验证。有以下三种方法进行验证：

- 登录集群节点执行如下任意一种命令：
```
curl http://{集群IP}:9121/metrics
curl http://{集群任意节点私有IP}:30378/metrics
```

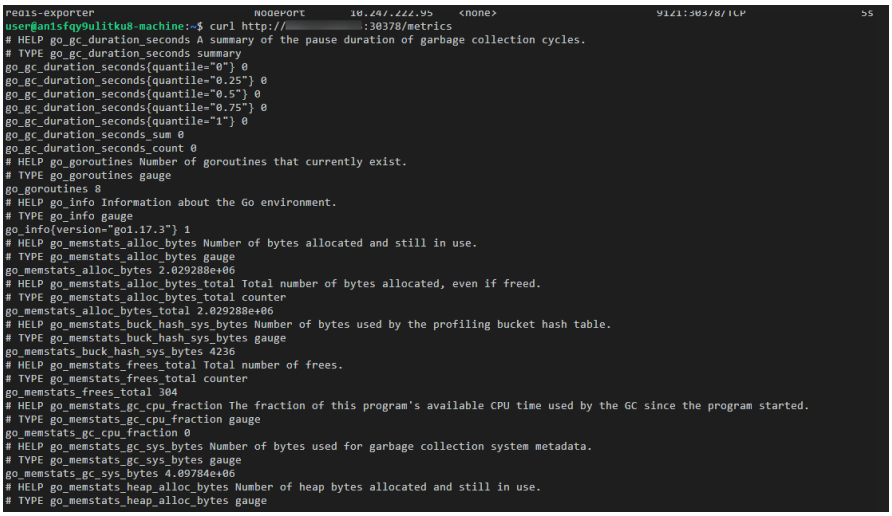
- 访问地址：http://{集群任意节点的公网IP}:30378/metrics
  如发现未能得到数据，请检查一下**部署Redis Exporter**时YAML中的**REDIS_ADDR**和**REDIS_PASSWORD**是否正确，示例如下：

图 5-17 访问地址



- 在实例列表中单击操作列下的“更多 > 远程登录”，在弹出的控制台中执行如下命令。
```
curl http://localhost:9121/metrics
```

图 5-18 执行命令



----结束

添加采集任务

通过**新增PodMonitor**方式为应用配置可观测监控Prometheus版的采集规则，监控部署在CCE集群内的应用的业务数据。

说明

如下指标采集的周期是30秒，所以等待大概30秒后才能在AOM的界面上查看到上报的指标。

```
apiVersion: monitoring.coreos.com/v1
kind: PodMonitor
metadata:
  name: redis-exporter
  namespace: default
spec:
  namespaceSelector: #选择要监控 Exporter Pod 所在的namespace
    matchNames:
      - default # exporter所在的命名空间
  podMetricsEndpoints:
  - interval: 30s # 设置指标采集周期
    path: /metrics # 填写 Prometheus Exporter 对应的 path 的值，默认/metrics
    port: metric-port# 填写 Prometheus Exporter 对应的 YAML 的 ports 的 name
  selector: # 填写要监控 Exporter Pod 的 Label 标签，以定位目标 Exporter
    matchLabels:
      k8s-app: redis-exporter
```

验证指标上报到 AOM

步骤1 登录AOM 2.0控制台。

步骤2 在左侧菜单栏中选择“Prometheus监控 > 实例列表”。

步骤3 单击接入了该CCE集群的“prometheus for CCE”实例名称，进入实例详情页面。

步骤4 在“服务发现”页面的“指标”页签下，选择集群。

步骤5 在搜索框输入redis，能够搜索出redis开头的指标，即可证明指标成功接入AOM。

----结束

在 AOM 上配置仪表盘和告警

通过仪表盘功能可视化监控CCE集群数据，通过告警规则功能，在集群发生故障时能够及时发现并预警。

- 配置仪表盘图表
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“仪表盘”，单击“创建仪表盘”新建一个仪表盘，详情可参见**创建仪表盘**。
  c. 在仪表盘页面选择实例类型为“Prometheus for CCE”的实例并单击“添加图表”，详情请参见**添加图表至仪表盘**。
- 配置告警
  a. 登录AOM 2.0控制台。
  b. 在左侧菜单栏中选择“告警管理 > 告警规则”。
  c. 单击“创建告警规则”配置告警，详情请参见**创建指标告警规则**。

5.8 其他 Exporter 接入

操作场景

Prometheus监控服务目前已经提供了常用中间件exporter接入操作指导，由于AOM兼容原生Prometheus，所以您也可以安装社区其他的Exporter。

操作方式

如果您所使用的基础组件还没有提供相应的集成方式，可以参考如下方式进行集成，以及自定义监控大屏来满足相应的监控需求。

1. **开源社区Exporter列表。**
2. 在**容器场景自建中间件接入**，已经提供部分常用中间件exporter接入操作指导，可以根据操作安装其他的Exporter。