时速云微服务平台 TMF

用户手册

北京云思畅想科技有限公司

产品版本：微服务平台 TMF 5.4.0

资料版本：P100V005R30

北京云思畅想科技有限公司（以下简称“时速云”）为客户提供全方位的技术支持和服务。购买时速云产品的用户，如果在使用过程中有任何问题，可与时速云各地办事处或用户服务中心联系，也可直接与公司总部联系。

读者如有任何关于时速云产品的问题，或者有意进一步了解公司其他相关产品，可通过下列方式与我们联系。

公司网址：https://www.tenxcloud.com

联系方式：400-626-1876

前　言

时速云微服务平台 TMF 用户手册总共包含 6 个章节，用于快速引导用户了解并使用时速云微服务平台以及帮助用户解决使用过程中遇到的问题。

产品版本

与本文档相对应的产品版本如下所示。

产品名称	产品版本
微服务治理	V5.4.0

符号约定

符号	说明
	以本标志开始的文本表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
	以本标志开始的文本是正文的附加信息，是对正文的强调和补充。
	以本标志开始的文本能帮助用户解决某个问题或节省用户的时间。

修订记录

编号	修订内容简述	修订日期	修订人	审阅人
1	v5.4.0 主要是升级 nacos 至 2.0.4，客户端适配 1.4.x 和 2.0.x	2022-5-9	LW	
2	6.3.1.9 服务详情中设置服务跨集群访问方式调整为直接设置“跨集群访问地址”，去掉虚拟实例概念	2022-5-9	LW	

3	适配 nacos2.0.4，配置中心关联服务设计调整，更新 6.4.3.2	2022-5-9	LW	
4	新增常见问题章节	2022-5-10	LW	

目录

1 产品介绍

1.1 概述

时速云微服务平台 TMF（TenxCloud Microservice Framework）是一个托管式的企业级微服务应用流量统一管理平台。以开源框架 Spring Cloud、Nacos 为核心基础组件，支持主流微服务框架 SpringCloud、Dubbo，轻松完成原生 SpringCloud，Dubbo 上云。平台完全基于开源技术构建，不绑定特定开发框架和平台，实现与企业内 IT 基础设施的快速集成。帮助企业轻松部署、更新、监控、运维以及治理其微服务应用。

1.2 产品功能

TMF v4.1 及以前版本采用 Spring Cloud 的高可用分布式集成方案，使用服务注册中心解决服务注册与发现，使用代码仓库级配置中心解决配置更新与发布问题，提供方便的应用集成。

TMF v5.1 全面更新注册中心、配置中心，支持 Nacos 核心组件，简化了使用流程，增强了服务注册与发现、动态配置管理的功能和易用性。

TMF v5.2 对网关智能路由、服务限流（服务级、接口级）、熔断策略等进行了功能和性能增强；全面更新了链路追踪，在全链路拓扑中提供多维度监控数据，更方便对服务进行横向、纵向对比，提升快速发现定位问题的能力；新增了主备服务故障切换，服务维度提供自动、手动切换策略，保证服务可用性；提供一站式解决方案，简化微服务架构应用开发、部署、测试、上线、运维流程，大大减少管理成本。

TMF v5.3 在原有基础上支持了跨集群服务注册发现、多集群、多租户链路追踪信息查询、链路告警、方法追踪、方法诊断等新功能，进一步丰富了微服务治理平台的功能。

微服务平台主要有、配置中心、认证中心、链路追踪、微服务运维。

TMF v5.4 主要是升级 nacos 至 2.0.4，客户端适配 1.4.x 和 2.0.x。

表 1-1 功能说明列表

功能	说明

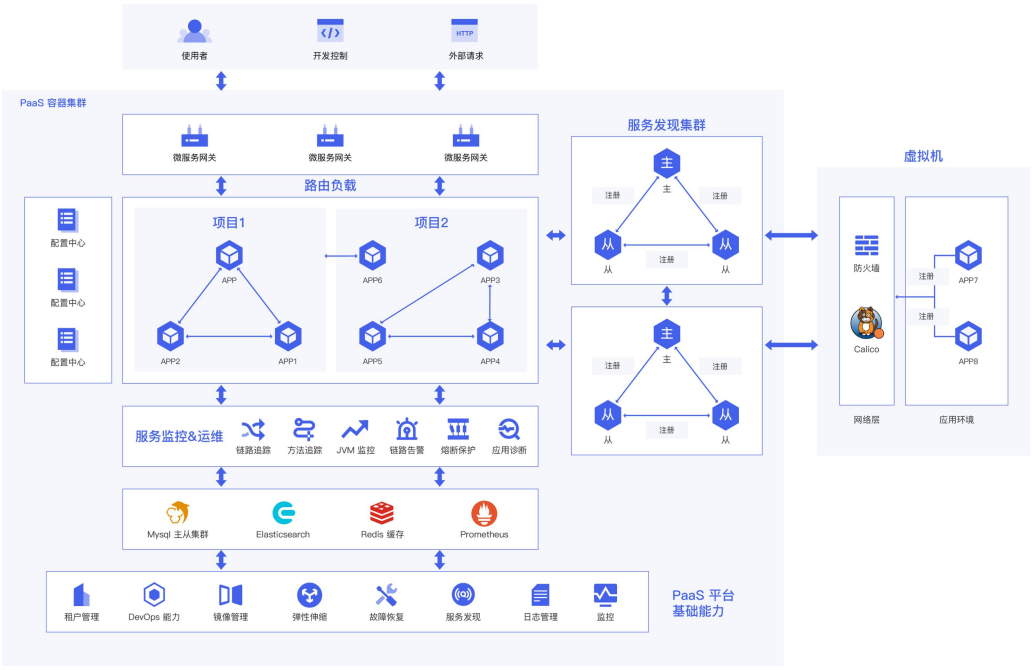
注册中心	● 实现微服务生命周期管理初始环节的自动化，让每个服务之间相互感知，相互通信 ● Nacos 注册中心支持高可用
配置中心	● 集中配置服务配置文件，统一管理，实时更新 ● 以中心化、外部化和动态化的方式管理所有环境的服务配置
主备服务故障切换	● 支持为服务设置备用服务，提供自动、手动切换策略，当策略触发时，注册中心将上游请求切换到备服务，保证其正常访问
链路追踪	● 分布式链路调用监控系统，能帮助用户收集时序数据用以定位微服务中的延迟问题，同时管理追踪数据的收集和查询 ● 追踪微服务、微服务组件、中间件（MySQL、RabbitMQ、Redis、Kafka、RocketMQ）之间的调用及数据统计展示 ● 全链路拓扑多维度展示服务监控数据，自由组合面板完成服务的横向、纵向对比，快速发现、定位问题
微服务网关	● 实现智能路由，通过配置权重路由、路由转发匹配条件、超时时间、失败重试次数等属性进一步扩展网关的路由功能。 ● 精细化流量管理，解决微服务之间的流量分配问题，保证微服务在自己的资源池运行，互不影响；限制请求次数确保不受网络攻击
熔断降级	● 服务熔断：当我们发现由于某些原因导致服务出现了过载现象，为避免造成整个系统故障，可采用熔断来进行保护 ● 服务降级：主要是用于对下游出现超时的非核心服务提供者进行低优先级调用，确保上游核心应用（服务消费者）不被影响
服务限流	● 服务级限流，支持基于平均 RT 策略、基于入口 QPS 策略、基于并发线程数策略、基于 CPU 使用率策略，保证微服务整体的稳定性 ● 接口级限流，监控微服务某接口 QPS 或线程数指标，当超出设定阈值时，微服务可立即截断流量或匀速排队处理流量，避免被瞬时的流量高峰冲垮，从而保障微服务可用性
访问控制	● 根据服务调用来源判断请求是否允许放行，通过设置黑白名单，保证服务访问来源的正确性，防止服务被恶意调用
认证中心	● 支持基于 OAuth2 协议，实现多个微服务的统一认证授权。 ● 提供独立的用户身份认证账户中心，用于管理用户的 OAuth 应用、账户信息和授权管理等。 ● 支持对用户、客户端应用的统一管理。认证服务支持多种模式，

	适合不同微服务场景
运维管理	● 微服务配置：支持租户管理员安装、卸载微服务组件 ● 微服务组件：支持租户管理员对微服务组件进行管理，包括对组件的重启、停止、启动、重新部署等操作 ● 监控运维：系统提供微服务的链路告警和应用诊断相关功能。

2 架构说明

2.1 系统架构

图 2-1 系统架构示意图



2.2 功能架构

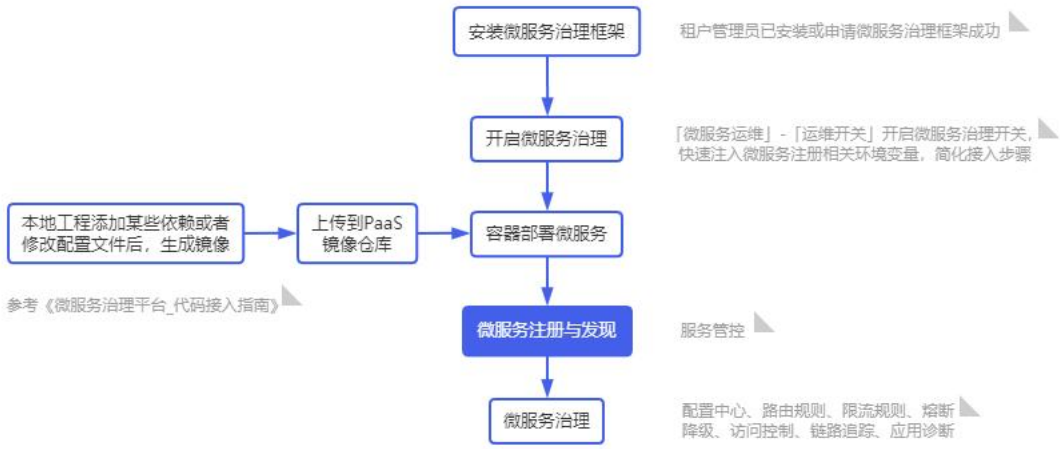
图 2-2 功能架构示意图


开发者
运营商
合作伙伴
微服务治理平台（TMF）
网关
智能路由
服务限流
负载均衡
服务注册与发现
跨集群服务发现
服务故障切换
健康检查
实例上/下线
跨集群全链路拓扑
多维度调用链查询
方法追踪
JVM监控
服务、接口及限流
熔断降级
访问控制
认证鉴权
配置中心
动态更新
实例回滚
导入/导出
运维中心
微服务框架管理
链路告警
应用诊断
多租户日志查询
容器化部署和运维（PaaS）
监控告警
租户管理（PaaS）
集群管理（PaaS）

3 快速入门

3.1 平台使用流程

图 3-1 平台使用流程示意图



部署微服务环境流程：

- 微服务地址设置->微服务安装与运维，详细参考 5 管理员指南。
- 微服务治理框架安装成功后，需到运维开关中开启微服务治理(容器部署自动注入环境变量)，接入服务后对其进行管理、治理及运维等。

3.2 微服务接入

微服务支持容器部署、主机部署方式，接入步骤详见《微服务治理平台_代码接入指南》

TMF_v5.4.0_时速云微服务治理平台

其中容器部署方式支持自动注入环境变量（前提：需到[微服务运维/运维开关]中开启“微服务治理”开关）。

- SpringCloud 服务接入请参看章节 6。
- Dubbo 服务接入请参看章节 6。
- 配置中心接入请参看章节 6。
- 动态接入链路追踪接入请参看章节 6。
- 服务治理接入（即支持服务限流、熔断降级、访问控制等治理能力）请参看章节 6。

3.3 平台管理自身组件快速升级

平台支持界面对自身组件的升级，平台安装成功后，系统、平台管理员登录后，进入租户平台管理，项目微服务组件中，可查看应用 msa-services 中的四个服务，分别是：initializr-api，msa-portal，spring-cloud-api，uaa。

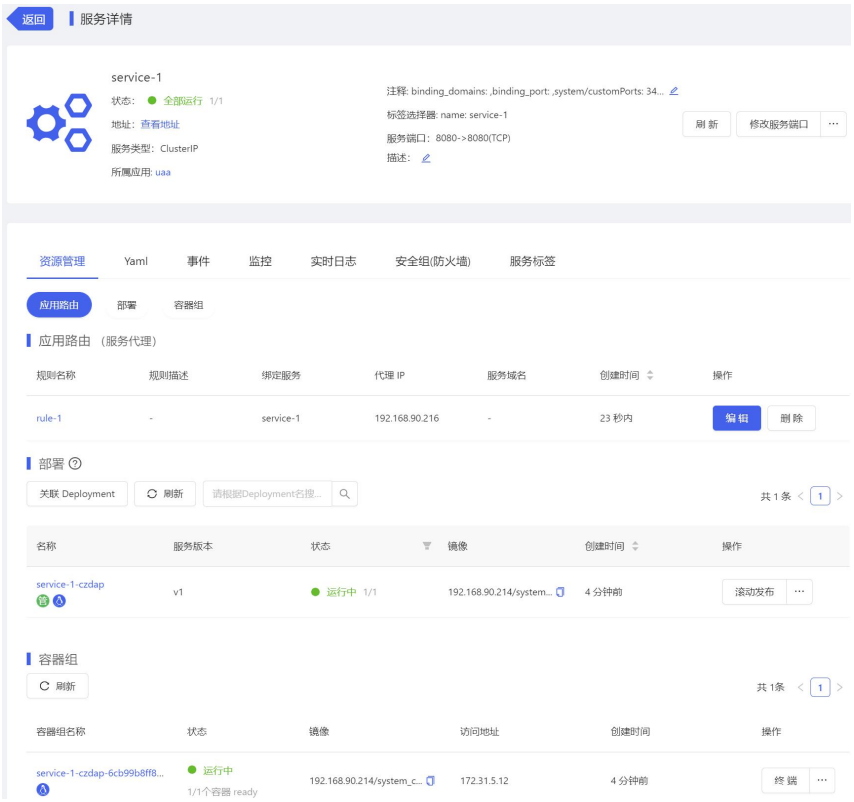
主要用于对这些平台自身组件的升级(跨版本升级会出现兼容问题，请谨慎操作)，以升级 uaa 组件为例，具体操作步骤如下：

(1) 以系统或平台管理员账户登录平台，在[租户列表/平台管理]选择租户，进入平台管理租户下。下拉上部导航“所有产品”，在菜单中选择[容器应用/服务]，进入服务列表页面。

图 3-2 所有产品导航展示

(2) 在该页面可见四个运行中的服务 initializr-api，msa-portal，spring-cloud-api，uaa，找到需要升级的 uaa 组件，点击 uaa 进入详情页，最右侧的操作列下拉，点击<滚动发布>按钮。

图 3-3 容器服务 uaa 的服务详情页面



(3) 在弹出框中可见当前运行的镜像版本 5.2.1，切换“目标版本”5.4.0，点击<确定>按钮，即可完成滚动发布。

图 3-4 服务滚动发布弹框设置

4 最佳实践

4.1 微服务开发

创建本地工程一键发布到平台代码仓库，结合 DevOps 产品开启敏捷开发，创建流水线，在线开发联调。

(1) 进入[微服务治理平台/微服务开发]页面，创建本地工程。

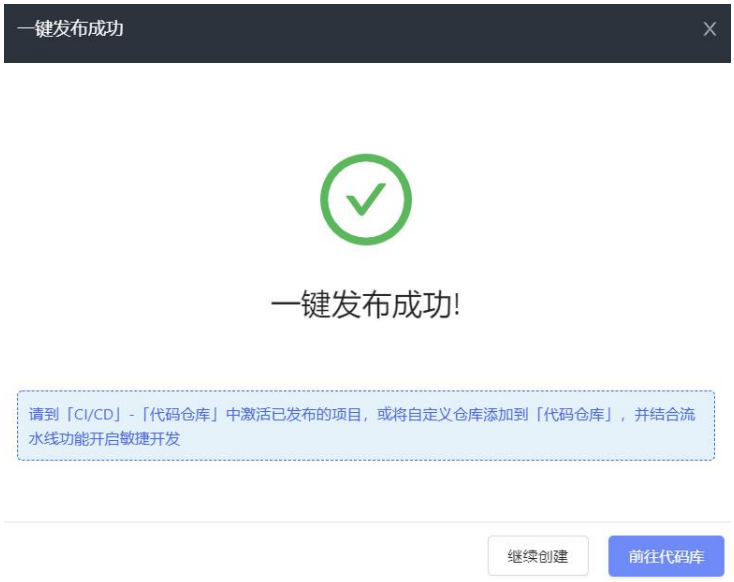
图 4-1 创建工程页面

(2) 配置工程对应的参数之后，单击<一键发布>按钮，进入仓库配置页面。

图 4-2 一键发布配置页面

(3) 配置代码仓库对应的参数，选择已经配置好的代码仓库，单击<确定>按钮。

图 4-3 一键发布成功



(4) 按照系统提示，在[CI/CD/代码仓库]对应的仓库类型中，找到已发布的工程，激活该代码源。

(5) 用户可以使用已激活的代码源创建流水线，如下图所示。

图 4-4 流水线新建页面

4.2 服务注册发现

4.2.1 部署示例服务

为方便快速体验微服务治理平台各功能，TMF 发布时提供测试 Demo，安装 TMF 将自动上传 Demo 镜像到 harbor 仓库。也可根据《TMF_v5.4.0_时速云微服务治理平台_代码接入指南》体验 Demo 从功能接入到镜像生成。Demo 相关说明如下：

Springcloud Demo 适配 Nacos 1.4.2

- tmf-demo-1.4.2：包含所有不涉及中间件和 dubbo 的功能，直接部署即可用（推荐）。

镜像：172.22.96.120/tmf-registry_v5.4.0/tmf-demo-1.4.2:v5.4.0-20220512.112641.59

镜像 ID：41e27628c9c8

Springcloud Demo 适配 Nacos 2.0.4

- tmf-demo-2.0.4：包含所有不涉及中间件和 dubbo 的功能，直接部署即可用（推荐）。

镜像：172.22.96.120/tmf-registry_v5.4.0/tmf-demo-2.0.4:v5.4.0-20220512.112656.38

镜像 ID：3ff9cb5b0316

- opentracing-demo：包含中间件调用，需提前部署中间件，方便查看拓扑图中间件面板指标。

镜像：172.22.96.120/tmf-registry_v5.4.0/opentracing-demo:v5.3.0-20220509.163649.50

镜像 ID：12facef14728

Dubbo Demo：

- provider-sample：包含中间件及所有微服务框架支持的 dubbo 功能（推荐）。

镜像：172.22.96.120/tmf-registry_v5.4.0/provider-sample:2.7.10-nacos-20211125015701

镜像 ID：19d9a6643386

- provider 不包含中间件

镜像：172.22.96.120/tmf-registry_v5.4.0/provider:2.7.10-nacos-20211122073349

镜像 ID：8d1ced8ac02e

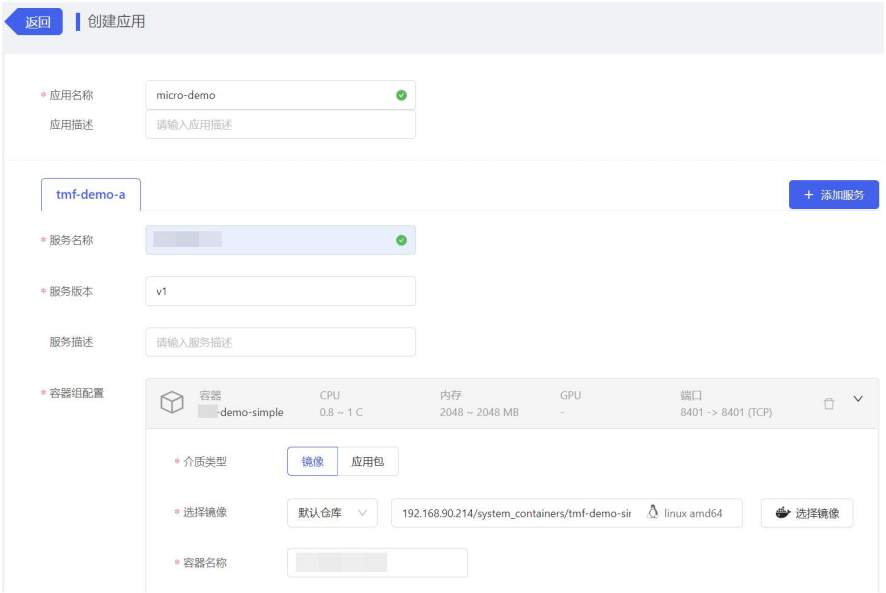
- tmf-dubbo-consumer：与服务提供者搭配使用。

镜像：172.22.96.120/tmf-registry_v5.4.0/tmf-dubbo-consumer:v5.3.0-20211129.175735.67

镜像 ID：4f00ca7b4edc

在[容器服务/容器应用/应用]中创建应用与服务，应用名称：micro-demo，服务名称：tmf-demo-a，选择镜像，填写容器名，点击创建。

图 4-5 创建服务

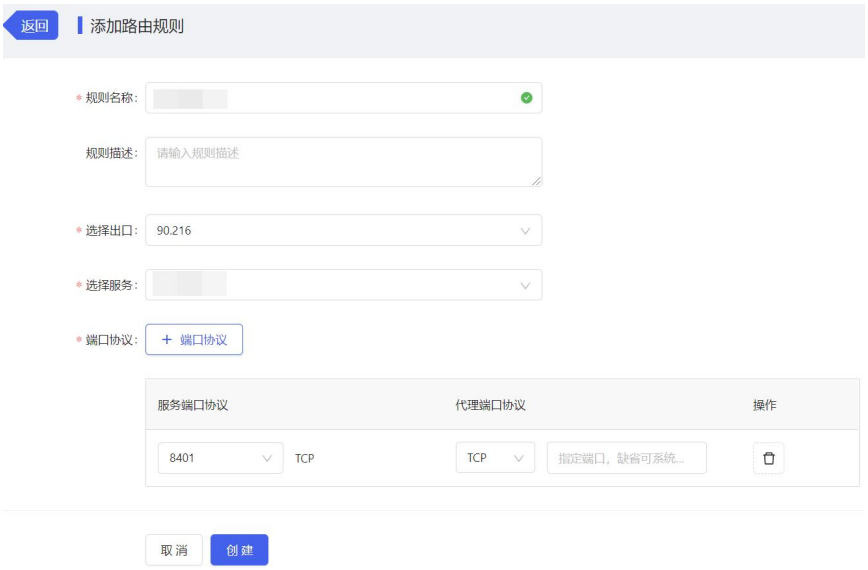


服务成功启动后，在[微服务治理平台/微服务/服务管控]列表中可查看已经注册成功。

4.2.2 通过平台暴露服务地址

在[容器服务/网络管理/应用路由]中点击“添加路由规则”，填写路由规则->创建。

图 4-6 应用路由-添加路由规则



添加成功后，在[容器服务/容器应用/应用]查看应用 micro-demo 中 tmf-demo-a 服务访问地址（http://192.168.90.216:58163）。

图 4-7 查看平台访问地址

复制访问地址，在浏览器访问/hello 接口，访问成功。

图 4-8 验证服务地址结果

4.3 微服务配置中心

4.3.1 添加配置

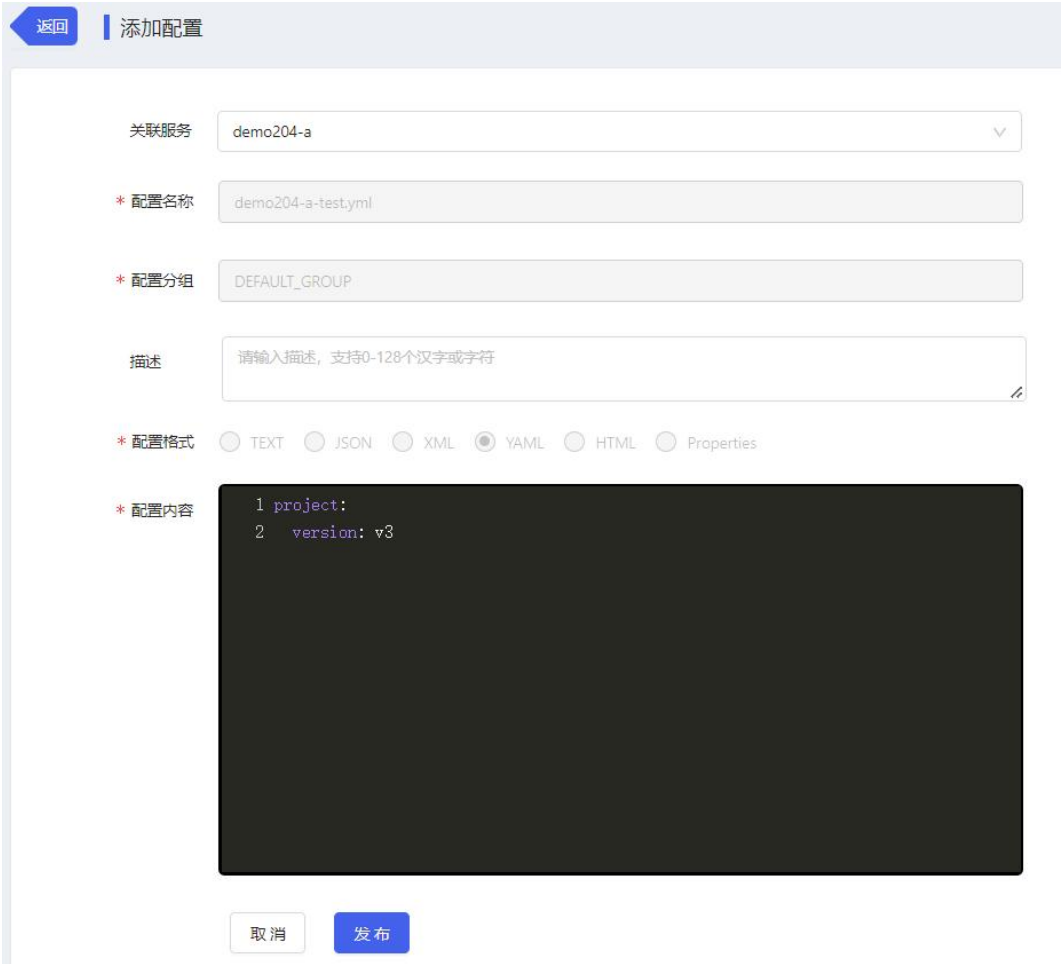
添加配置时配置名称和配置分组满足以下条件时才能和注册中心的微服务进行关联。

(1) 配置名称应为：{discovery-service-name}-{profile}，profile 值默认是 test，可以在 paas 平台的应用详情查看相应服务的环境变量 PROFILE，对应上即可。

(2) 配置分组：可自定义，建议和注册中心的分组保持一致。

(3) 关联服务：选择微服务，如果按接入文档规范配置完成，会自动回显关联的配置名称、配置分组。

图 4-6 添加配置



配置内容如下图所示。

```
project:
  version: v3
```

4.3.2 验证配置

图 4-6 配置中心验证结果

4.4 微服务网关

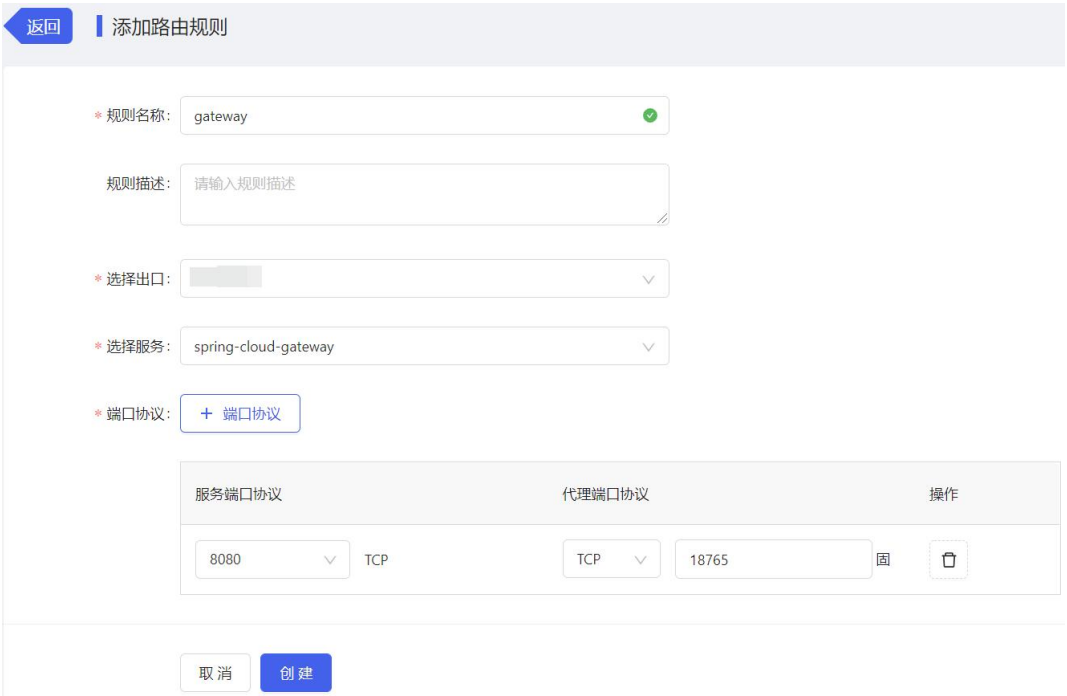
4.4.1 查看微服务网关地址

进入[微服务治理平台/微服务/服务管控]页面，选择组件列表，查看 spring-cloud-gateway 的内网 or 外网访问地址（需要到[容器服务/网络管理/应用路由]中为 spring-cloud-gateway 网关服务设置服务代理）。

图 4-9 服务列表页面

PaaS 平台应用路由配置说明：切换到微服务治理框架所在的项目，为网关添加路由规则。

图 4-10 添加应用路由页面



4.4.2 配置智能路由

在[微服务治理平台/微服务/服务管控]页面，选择某个服务，进入服务详情，选择[路由规则]tab 页，点击“添加路由”，填写路由路径/version，默认选中微服务 tmf-demo-a，点击<确认>后，添加成功。

图 4-11 添加路由规则页面

路由管理列表中显示添加成功的路由，添加完后开启该路由即可生效：

图 4-12 路由列表页面

4.4.3 验证路由

(1) 微服务网关内网地址+路由路径。

图 4-13 微服务网关+路由路径访问服务成功页面

4.4.4 配置限流规则(网关限流)

在[微服务治理平台/微服务/服务管控]页面，选择某个服务，进入服务详情，选择[限流规则/网关限流]tab 页，点击“添加限流规则”，为微服务 tmf-demo-a 设置限流规则，限流类型选择 URL，限流阈值 2 次，窗口 10 秒。

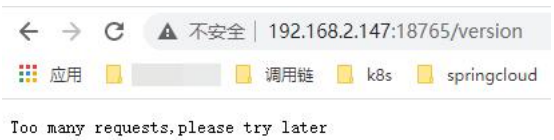
图 4-14 添加限流规则弹框

限流列表中显示添加成功的限流，添加完后开启该限流即可生效。

4.4.5 验证限流

(1) 通过微服务网关访问 tmf-demo-a，10 秒内超过 2 次结果。

图 4-15 超出限流阈值后访问结果



4.5 微服务主备切换

部署服务 tmf-demo-b，设置 tmf-demo-a 的备服务为 tmf-demo-b，在设置之前通过 tmf-

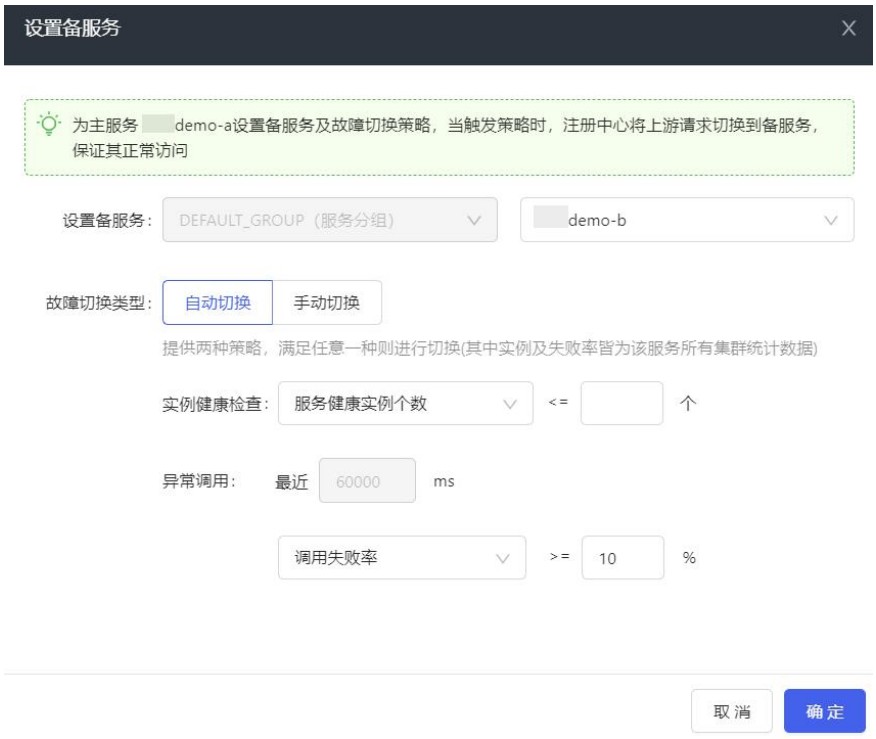
demo-b 服务的访问地址去调用 tmf-demo-a 服务正常，如下图所示。

图 4-16 配置页面

4.5.1 自动切换

异常调用策略选择调用响应时间，说明：关于这里的调用失败相关策略，不是我们理解的返回 5xx，而是网络不通情况下异常或者连接超时异常之类的才算是调用失败，比如 SocketException 或 SocketTimeoutException，这种场景不太好模拟，所以这里不用这种策略测试；简单模拟的话就使用调用响应时间策略演示。

图 4-17 配置页面



使用 tmf-demo-b 服务去调用 tmf-demo-a 的/sleep/200 接口。

图 4-18 配置页面

再去正常调用 tmf-demo-a 服务，返回的是 tmf-demo-b，测试通过。

图 4-19 配置页面

4.5.2 手动切换

图 4-20 配置页面

设置之后，去正常调用 tmf-demo-a 服务，返回的是 tmf-demo-b，测试通过。

图 4-21 配置页面

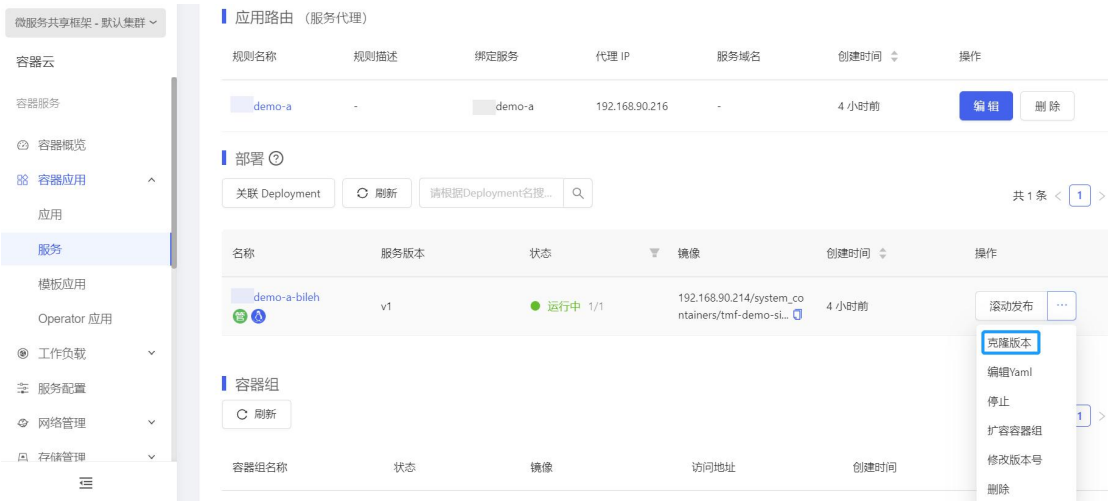
4.6 服务滚动升级/下线流量无损

背景：在 PaaS 平台对服务进行滚动升级，由于 nacos 有 15s 的服务健康->不健康的间隔，导致滚动过程中请求流量出错，请求被分发到了已删除的 ip 上，解决方案如下。

4.6.1 服务手动下线

(1) 在 PaaS 平台[容器服务/容器应用/服务详情/版本控制器]中对服务克隆一个版本，镜像为新版本的镜像，这时相当于环境中同时存在新旧版本的服务实例。

图 4-22 配置页面



(2) 在微服务平台[服务管控/服务详情/实例列表]中对就服务实例进行下线操作(隐藏实例)，此时消费者就不会收到已下线的实例列表，收到的就会是新版本的实例列表，虽然旧版本状态已下线，但真实还是在运行的，所以此时处理中的请求并不会出错。

图 4-23 配置页面

(3) 在窗口期（发版期，or 用户量少，or 已确认没有流量进到旧版本实例中）在 paas 平台将旧版本的服务 deployment 停止，此时环境中运行的就都是新版本的服务，即实现了下线流量无损。

4.6.2 服务自动下线

(1) 准备服务下线脚本，需要将脚本放置到服务镜像中（无损下线脚本实现参考，具体配置视现场情况而定，如遇问题需联系相关技术人员指导）。

```
#!/bin/bash
#当服务类型是 dubbo 时，以下脚本适用于 springcloud alibaba 的 dubbo，注册的服务名就是 application name，若 apache 的 dubbo springboot starter，则是以接口当服务，此时服务会比较多，做下线操作也会比较麻烦
#若只有一个 java 端口，可通过下面方式获取端口，若无法正常获取端口，
则需要通过某种方式告知脚本端口
# 1. 设置环境变量
export PORT=$(netstat -tunlp|grep java|awk '{print $4}'|awk -F":" '{print $2}')
export SERVICE_NAME=${PAAS_NAME}
export NAMESPACE_ID=${NACOS_NAMESPACE}
export POD_IP=$(cat /etc/hosts | awk 'END{print $1}')
export NACOS_SVC_HOST="spring-cloud-
discovery.${COMPONENT_NS}.svc.cluster.local:8848"
#export NACOS_SVC_HOST='192.168.2.120'
#export COMPONENT_NS=${COMPONENT_NS}
#export NACOS_GROUP=${NACOS_GROUP}
#declare -a RET
RET=$(curl -s -L -X PUT "http://${NACOS_SVC_HOST}/nacos/v1/ns/instance?" -
H 'Content-Type: application/x-www-form-urlencoded; charset=UTF-8' \
  --
data "serviceName=${SERVICE_NAME}&clusterName=DEFAULT&groupName=${N
ACOS_GROUP}&namespaceId=${NAMESPACE_ID}&ip=${POD_IP}&port=${PORT}
&ephemeral=true&weight=1&enabled=false")
if [ $RET == "ok" ];then
   sleep 60s
   exit 0
fi
## do other thing
exit 1
```

(2) 在 PAAS 平台[容器服务/工作负载/部署详情/容器配置/启动配置]中设置“容器钩子”为 PreStop（容器停止前执行）。

图 4-24 配置页面

(3) 在 PAAS 平台[容器服务/服务详情/版本控制器]中即可对服务执行滚动发布操作，此时滚动发布可实现流量无损。

4.7 微服务限流

服务级限流，支持基于平均 RT 策略、基于入口 QPS 策略、基于并发线程数策略、基于 CPU 使用率策略，保证微服务整体的稳定性。

接口级限流，监控微服务某接口 QPS 或线程数指标，当超出设定阈值时，微服务可立即截断流量或匀速排队处理流量，避免被瞬时的流量高峰冲垮，从而保障微服务可用性。

下面根据不同的业务场景来介绍限流的使用流程：

4.7.1 请求流量具有波峰波谷的特点(接口级限流）

流控原理：将前面的峰值流量延迟（匀速排队）到后面再处理，既能最大化满足所有请求，又能保证用户体验。

背景介绍：在实际微服务中，收到的请求是没有规律的。例如：某微服务的处理请求的能力是每秒 10 个。在某一秒，突然到来了 30 个请求，而接下来两秒，都没有请求到达。在这种情况下，如果直接拒绝 20 个请求，应用在接下来的两秒就会空闲。所以，需要把骤增的请求平均到一段时间内，让系统负载保持在请求处理水位之内，同时处理更多请求。

流控规则设置：验证流控规则前，为避免造成影响，应像停用其他限流、熔断、降价规则

(1) 进入微服务详情页面，选择[限流规则/流控规则]页签。

(2) 点击“添加流控规则”，弹框中输入接口名称，选择“QPS”流控策略，设置限流阈值 2；选择直接排队，点击确定，并开启规则。

图 4-25 设置流控规则（QPS-直接决绝）

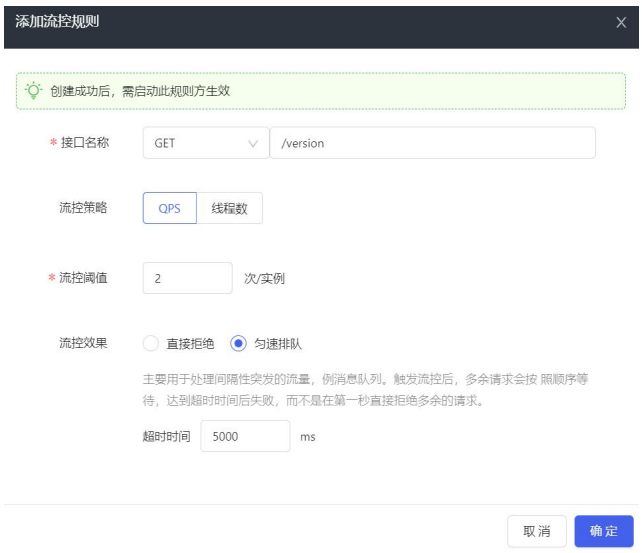
(3) 超过两次访问接口，请求直接被拒绝。

图 4-26 配置页面


```
Blocked by Sentinel (FlowException)
```

(4) 编辑流控规则，然后选择“匀速排队”流控效果，设置超时时间 5000ms。即微服务则每 100 ms 处理一条请求，多余的处理任务将排队；预计排队时长超过 5s 的处理任务将快速失败，直接返回默认流控信息。

图 4-27 设置流控规则（QPS-匀速排队）



4.7.2 用于保护业务线程池不被慢调用耗尽

背景介绍：例如，当应用所依赖的下游应用由于某种原因导致服务不稳定、响应延迟增加，对于调用者来说，意味着吞吐量下降和更多的线程数占用，极端情况下甚至导致线程池耗尽。为应对太多线程占用的情况，业内有使用隔离的方案，比如通过不同业务逻辑使用不同线程池来隔离业务自身之间的资源争抢（线程池隔离）。这种隔离方案虽然隔离性比较好，但是代价就是线程数目太多，线程上下文切换的 overhead 比较大，特别是对低延时的调用有比较大的影响。Sentinel 并发控制不负责创建和管理线程池，而是简单统计当前请求上下文的线程数目（正在执行的调用数目），如果超出阈值，新的请求会被立即拒绝，效果类似于信号量隔离。

流控规则设置：

(1) 进入微服务详情页面，选择[限流规则/流控规则]页签。

(2) 点击“添加流控规则”，弹框中输入接口名称，选择“线程数”，设置阈值 5，当处理请求的线程数达到 5 时，剩余请求将直接拒绝。

(3) 点击<确定>，规则创建成功，默认是停用状态。

(4) 列表中启用此规则即可。

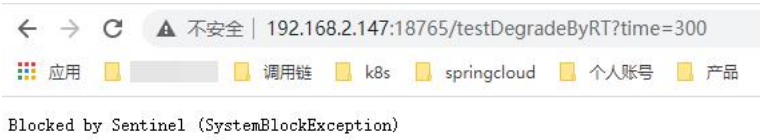
4.7.3 系统规则(服务级限流)

(1) 进入微服务详情页面，选择[限流规则/系统规则]页签，编辑限流策略。

(2) 启用“基于平均 RT 策略”，设置阈值 100。

(3) 连续访问接口，/testDegradeByRT?time=300，第一次请求的响应时间超过了设置的值，所以开启了限流策略，后面的请求被限流。

图 4-28 系统规则基于平均 RT 策略访问结果



4.8 微服务熔断

4.8.1 配置熔断规则(慢调用比例)

在[微服务治理平台/微服务/服务管控]页面，选择某个服务，进入服务详情，选择[熔断降级/熔断规则] tab 页，点击“添加熔断规则”，为微服务 tmf-demo-a 设置熔断规则，添加完成后启用该规则即可生效。(需要按照 4.4.2 为接口” /testDegradeByRT”设置网关路由。

图 4-29 添加熔断规则(满调用比例)弹框

4.8.2 验证熔断规则(慢调用比例)

访问五次以上 testDegradeByRT 接口，即可触发熔断。

图 4-7 超出熔断阈值后访问结果

4.9 链路追踪

4.9.1 链路配置

在[微服务治理平台/链路追踪/链路配置]页，点击“接入链路”弹框中选择服务 tmf-demo-a，为其安装 Agent，采集配置默认开启。

4.9.2 查看调用链及拓扑图

重复 4.7 及 4.8 章节接口调用，即可产生调用数据。

在[链路追踪/调用链查询]中可查看请求的调用链路。

在[链路追踪/全链路拓扑]中可查看服务间拓扑关系图及监控指标数据。

（链路追踪详细功能请参考 6.5 章节）

图 4-30 调用链查询图

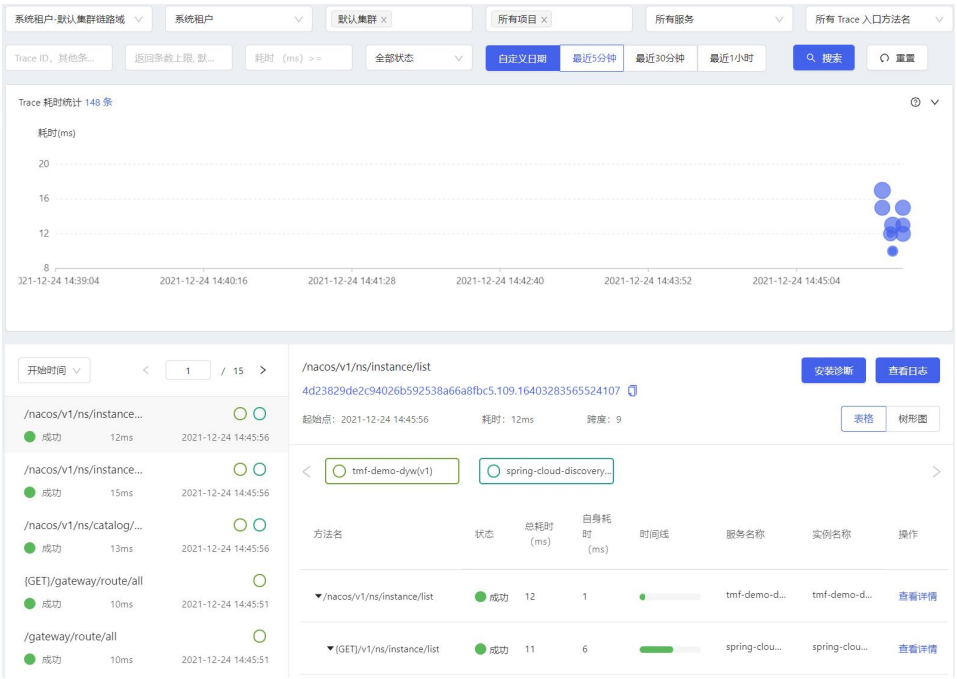
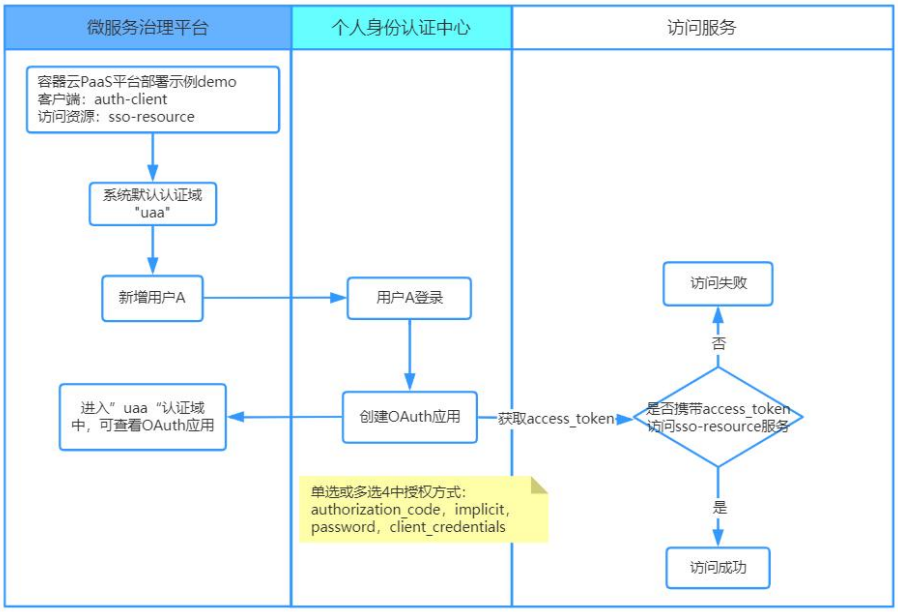


图 4-31 全链路拓扑图

4.10 UAA 授权认证

系统默认提供 uaa 认证域，也支持添加、编辑自定义认证域，来满足不同客户不同场景的需求。基本使用流程。

图 4-32 UAA 基本使用流程图



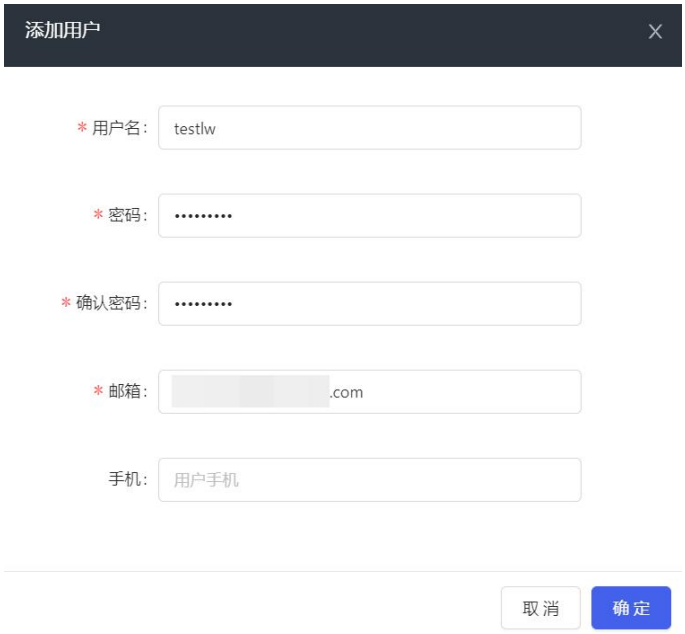
4.10.1 部署示例服务

参考 3.2 快速接入微服务注册中心，将示例服务注册到微服务平台。

4.10.2 创建 OAuth 应用

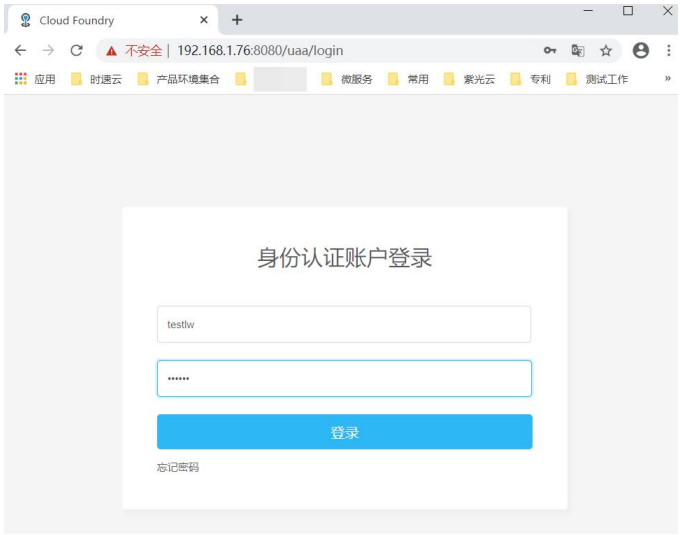
(1) 默认认证域“uaa”下，创建用户 testlw。

图 4-33 认证域管理-添加用户弹框



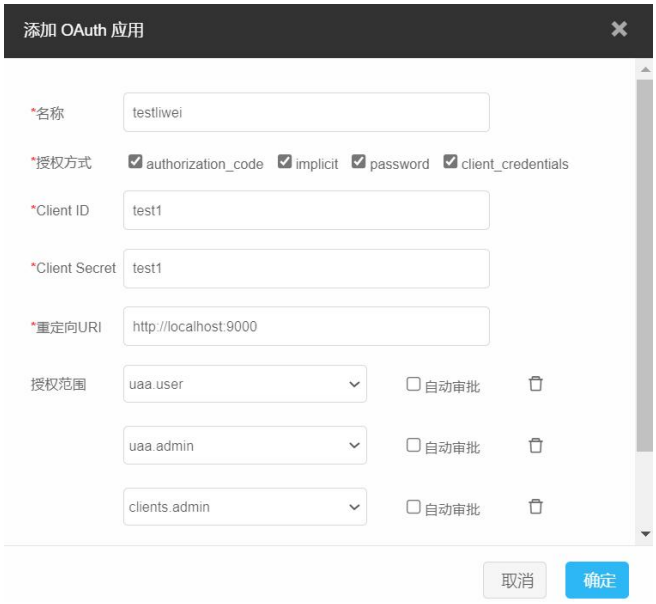
(2) 登录“身份认证中心”

图 4-34 个人身份认证中心登录页面



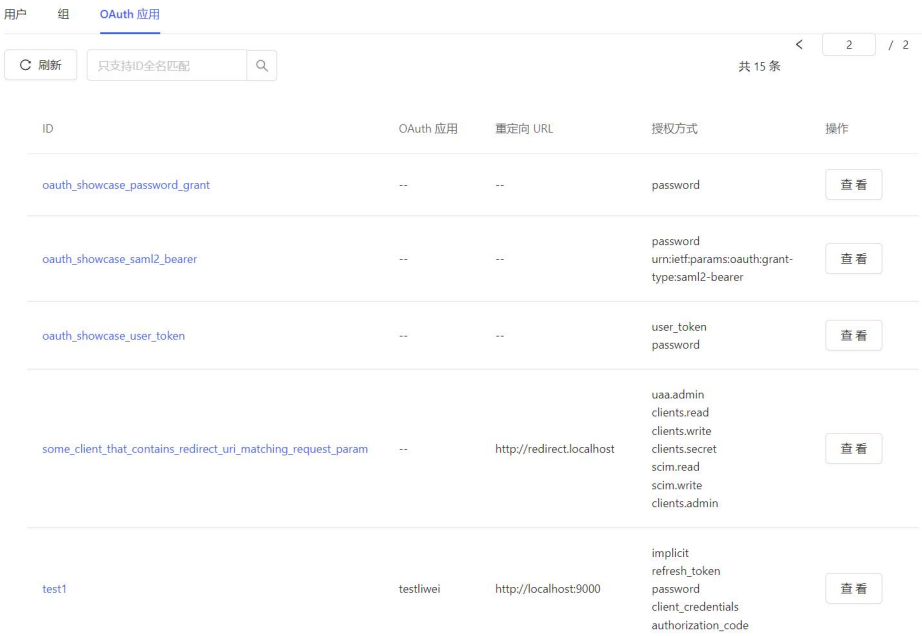
(3) 登录成功后，创建 OAuth 应用。

图 4-35 身份认证中心-添加 OAuth 应用弹框



(4) 微服务平台查看已创建成功的 OAuth 应用：test1。

图 4-36 认证域管理-OAuth 应用页面



4.10.3 UAA 授权认证示例（authorization code）

标准服务端授权模式说明：

标准的 Server 授权模式，非常适合 Server 端的 Web 应用。一旦资源的拥有者授权访问他们的数据之后，他们将会被重定向到 Web 应用并在 URL 的查询参数中附带一个授权码

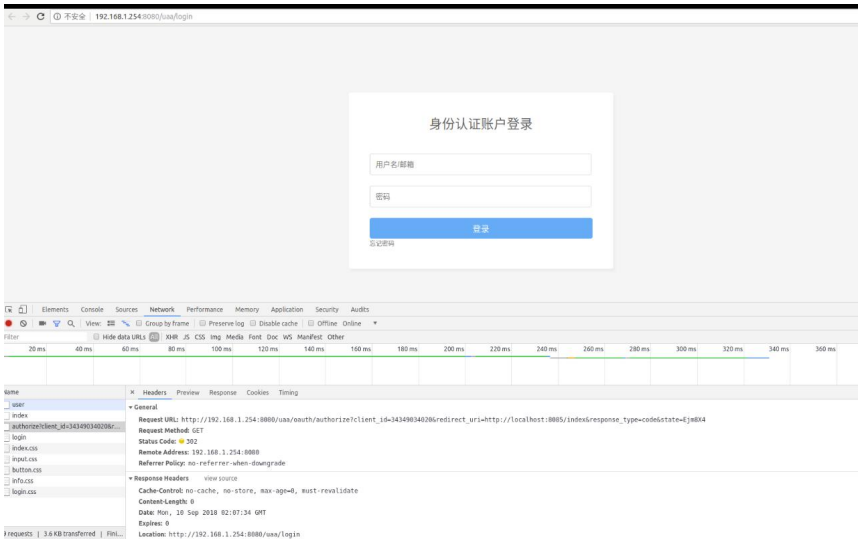
（code）。

在客户端里，该 code 用于请求访问令牌（access_token）。并且该令牌交换的过程是两个服务端之前完成的，防止其他人甚至是资源拥有者本人得到该令牌。另外，在该授权模式下可以通过 refresh_token 来刷新令牌以延长访问授权时间，也是最为复杂的一种方式。

4.10.3.1 获取 access_token

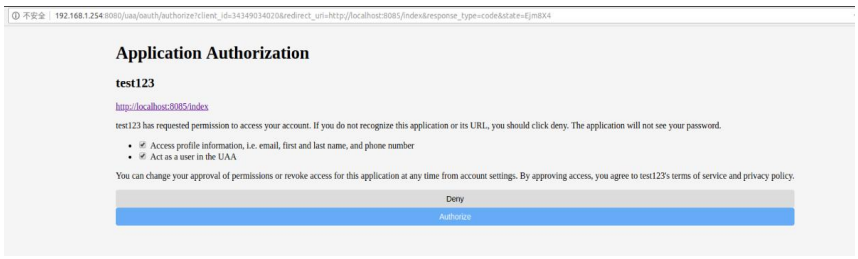
(1) 浏览器中输入如下地址：http://localhost:8085/user，自动跳到授权登录页。

图 4-37 身份认证授权登录页面



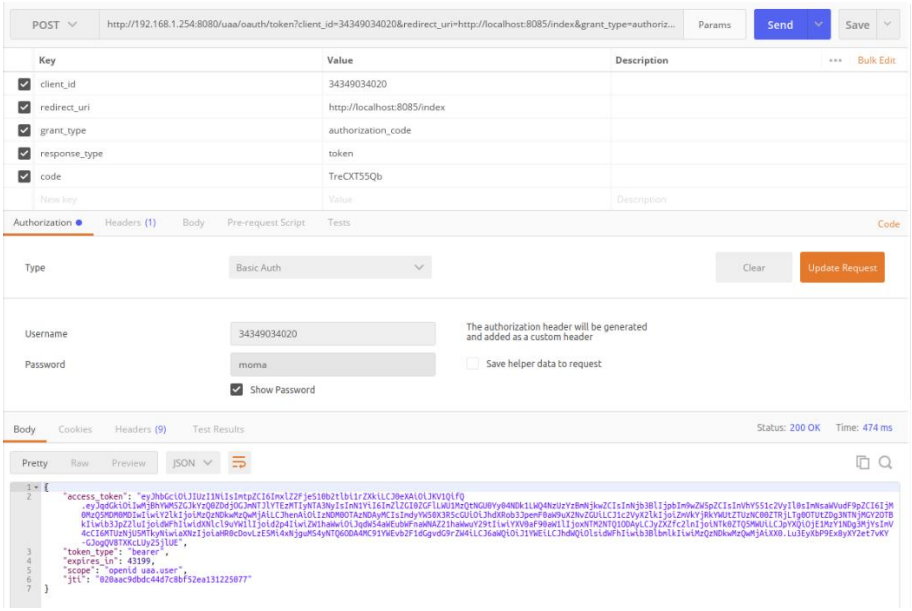
(2) 输入已注册的 UAA 用户账号和密码进行登录，进入授权页面。

图 4-38 身份认证授权页面



(3) 点击“Authorize”授权按钮，完成授权，重定向到已设置的 redirect_uri 地址，返回获取的 code 信息(code 为一次性），根据 code 码获取 access_token（所需参数：client_id=34349034020&redirect_uri=http://localhost:8085/index&grant_type=authorization_code&response_type=token&code=TreCXT55Qb）。

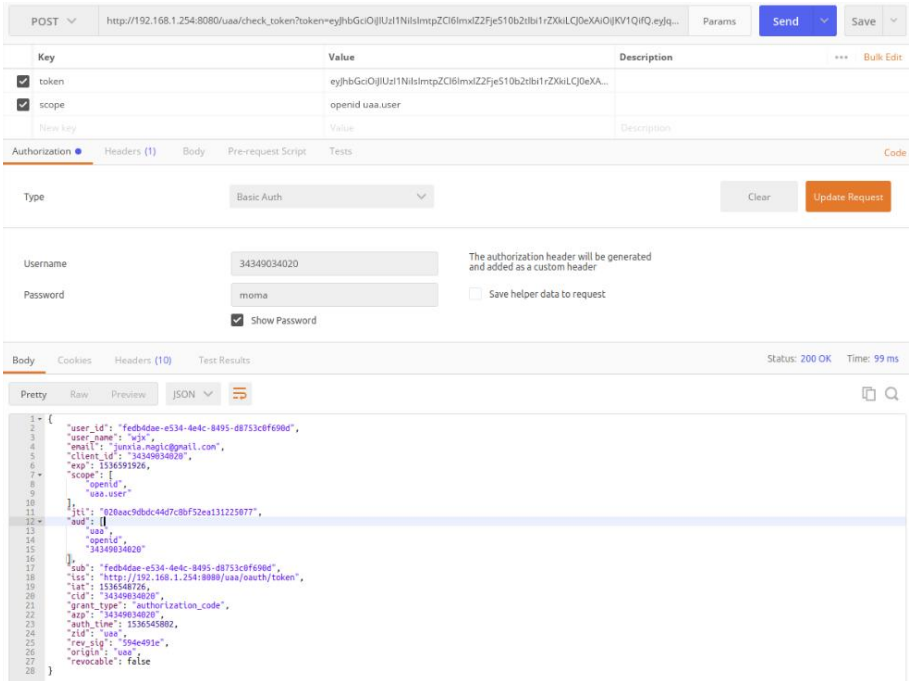
图 4-39 authorization code 授权方式获取 access_token 页面



4.10.3.2 访问资源

验证 access_token，获取资源信息，将 access_token 带入到下游服务请求，下游服务检查 access_token。

图 4-40 authorization code 授权方式访问资源页面



4.10.4 UAA 授权认证示例（password）

- 密码模式说明：这种模式要求用户提供用户名和密码来交换访问令牌 access_token。用户必须把自己的用户名和密码给客户端，但是客户端不得储存密码。因为初始验证之后，只需将 OAuth 的令牌记录下来即可。如果用户希望取消授权，因为其真实密码并没有被记录，因此无需修改密码就可以立即取消授权。token 本身也只是得到有限的授权，因此相比最传统的 username/password 授权，该模式依然更为安全。
- 示例中 uaa 账号名：testlw 密码：testlw。
- 示例中客户端应用：client_id: test1、 client_secret : test1。

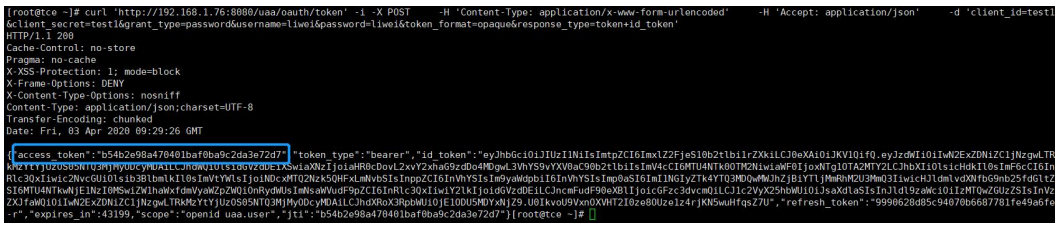
4.10.4.1 获取 access_token

uaa 认证服务器地址：http://192.168.1.76:8080/uaa/。

获取 token 接口地址：http://192.168.1.76:8080/uaa/oauth/token。

```
curl 'http://192.168.1.76:8080/uaa/oauth/token' -i -X POST      -H 'Content-Type: application/x-www-form-urlencoded'    -H 'Accept: application/json' -d 'client_id=test1&client_secret=test1&grant_type=password&username=testlw&password=testlw&token_format=opaque&response_type=token+id_token'
```

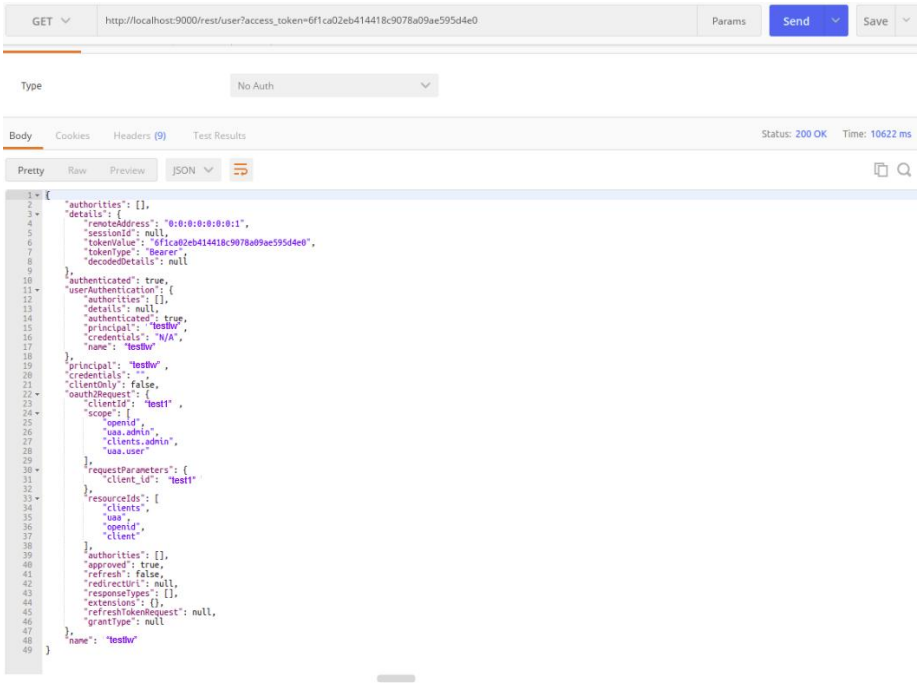
图 4-41 password 授权方式获取 access_token 页面



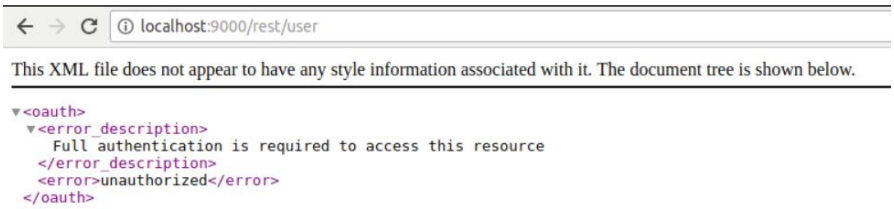
4.10.4.2 访问资源

(1) 根据 access_token 获取资源服务中的信息，访问资源服务接口/rest/user，获取成功。

图 4-42 password 授权方式访问资源页面



(2) 不使用 access_token，进行非法请求资源。



4.10.5 UAA 授权认证示例（implicit）

标准服务端授权简化模式：该模式是所有授权模式中最简单的一种。当用户访问该应用时，不通过第三方应用程序的服务器，直接在浏览器中向认证服务器申请令牌 access_token，服务端会立即生成一个新的访问令牌并通过 URL 的#hash 段传回客户端。这时，客户端就可以利用 JavaScript 等将其取出然后请求 API 接口。该模式不需要授权码（code），当然也不会提供 refresh token 以获得长期访问的入口。

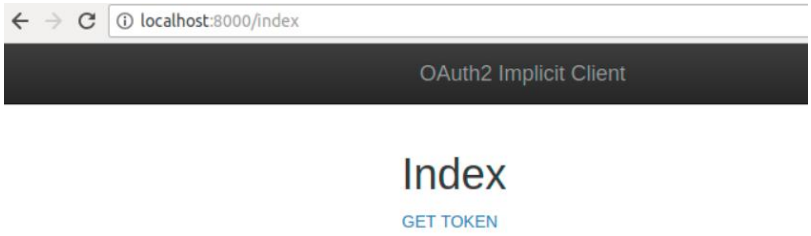
- 示例中 uaa 账号名：weiwei。
- 密码：weiwei。
- 示例中客户端应用：client_id: client。
- uaa 认证服务器地址：http://192.168.1.254:8080/uaa/。

- 客户端服务：sso-client-implicit，端口：8000。
- 资源服务器：sso-resource，端口：9000。

4.10.5.1 获取 access_token

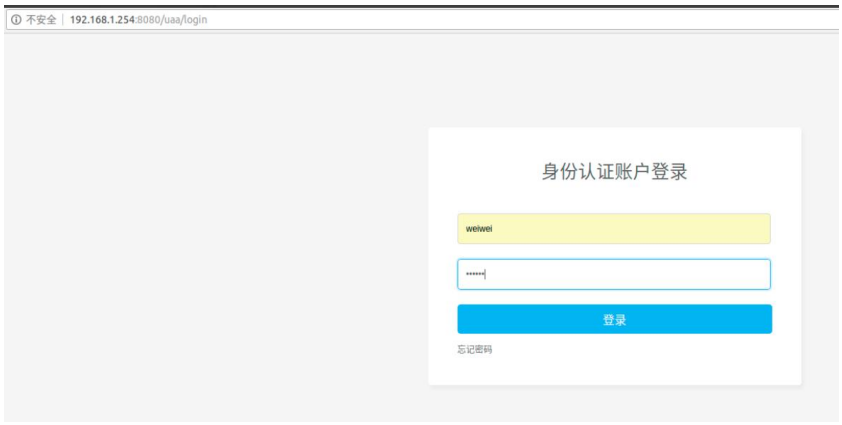
(1) 启动 sso-client-implicit ，启动后访问/index 接口，进入 index 页。

图 4-43 implicit 授权方式获取 access_token 页面 1



(2) 点击“GET TOKEN”链接，进入授权认证服务登录页。

图 4-44 implicit 授权方式获取 access_token 页面 2



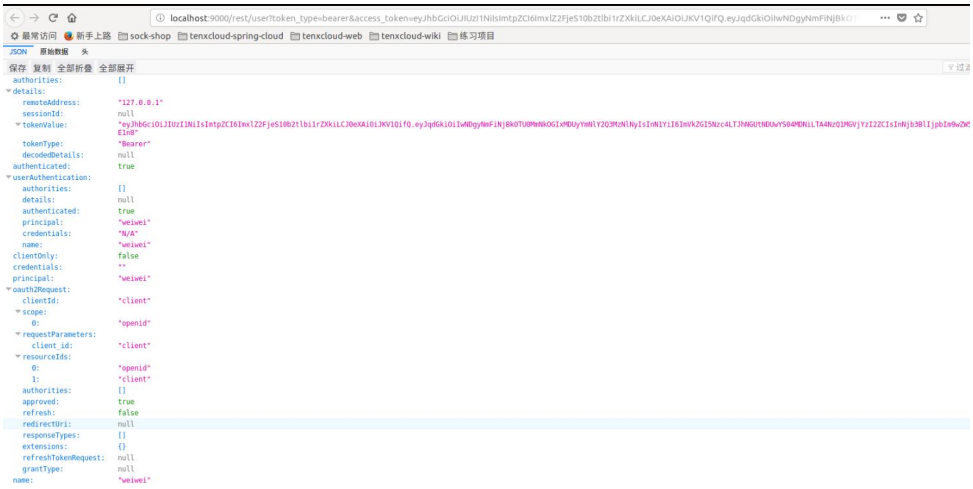
(3) 输入 uaa 账号密码，获取 access_token。

图 4-45 implicit 授权方式获取 access_token 页面 3

4.10.5.2 访问资源

(1) 点击“GET USERS”链接，获取资源服务服务信息。

图 4-46 implicit 授权方式访问资源页面



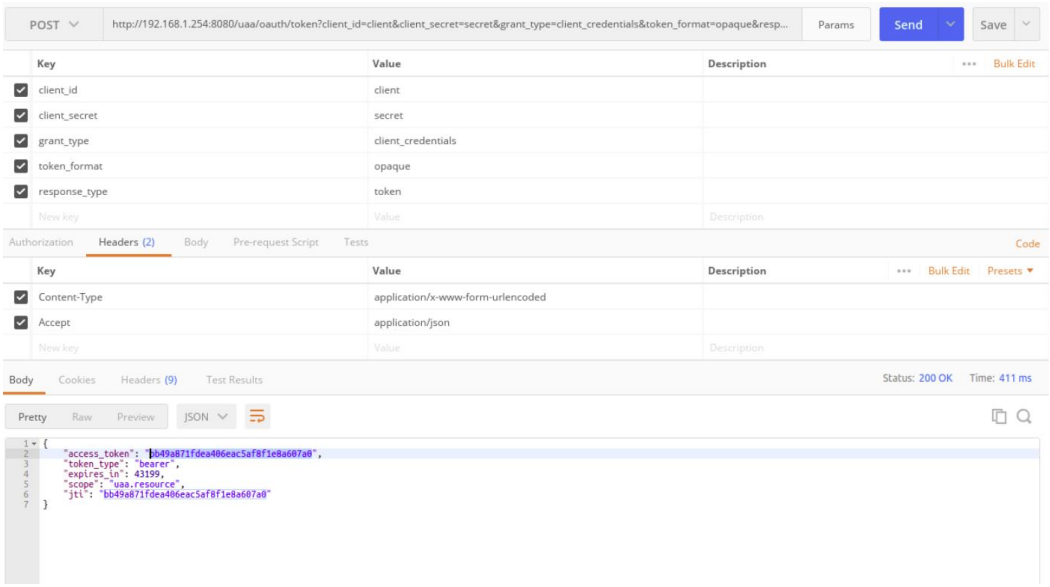
4.10.6 UAA 授权认证示例（client_credentials）

- 应用端密钥授权模式：客户端以自己的名义，而不是以用户的名义，要求“服务提供商”提供服务，不存在授权问题。
- 示例中客户端应用：client_id: client 、 client_secret : secret。
- uaa 认证服务器地址：http://192.168.1.254:8080/uaa/。
- 获取 token 接口地址：http://192.168.1.254:8080/uaa/oauth/token。

4.10.6.1 获取 access_token

客户端提供 client_id 和 client_secret，向认证服务索取获取服务资源的 access_token。

图 4-47 client_credentials 授权方式获取 access_token 页面



4.10.6.2 访问资源

根据 access_token 获取服务资源。

图 4-48 client_credentials 授权方式访问资源页面

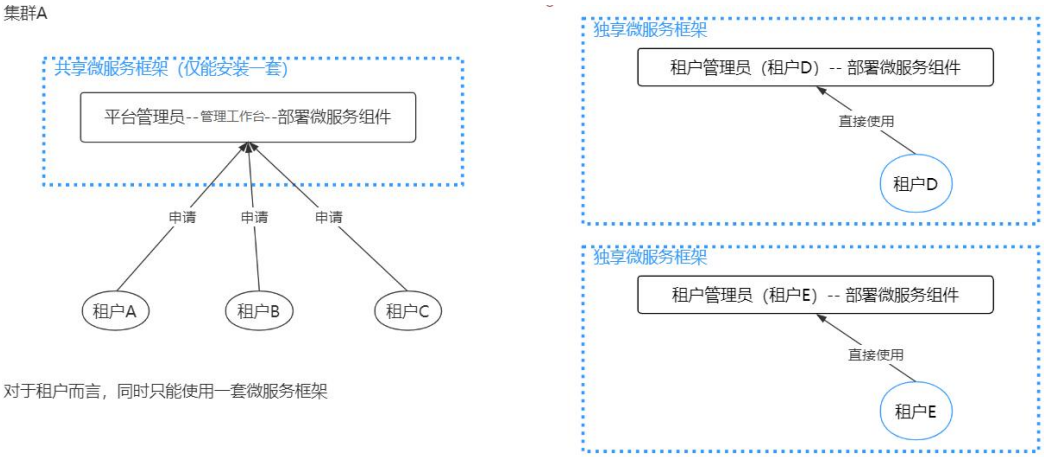
5 管理员指南

为满足不同用户需求，提供两种微服务治理框架部署方式。

- 管理工作台及系统租户中创建共享微服务治理框架，供平台其他租户申请使用。
- 每个租户也可自行创建独享微服务治理框架，供自己租户使用。

但同一时间同一集群，租户只能选择一种框架，如下图所示。

图 5-1 微服务部署方式



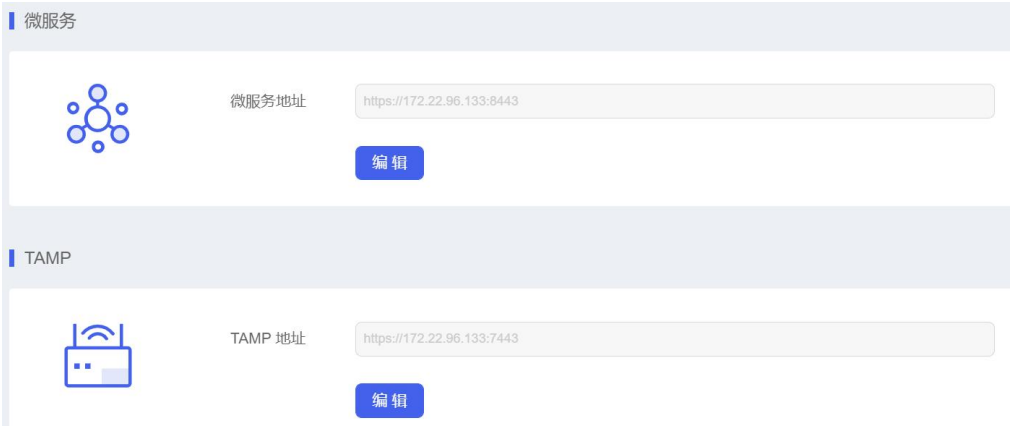
5.1 微服务地址设置

位置说明：[管理工作台/平台设置/全局配置]，查看微服务地址。

内容说明：此处为微服务治理平台界面地址，正确填写后才能从 TCE 跳转到微服务治理页面（安装好后微服务地址会自动配置）。

权限说明：系统管理员、基础设施管理员可见，可操作。

图 5-2 微服务地址设置



5.2 微服务环境配置

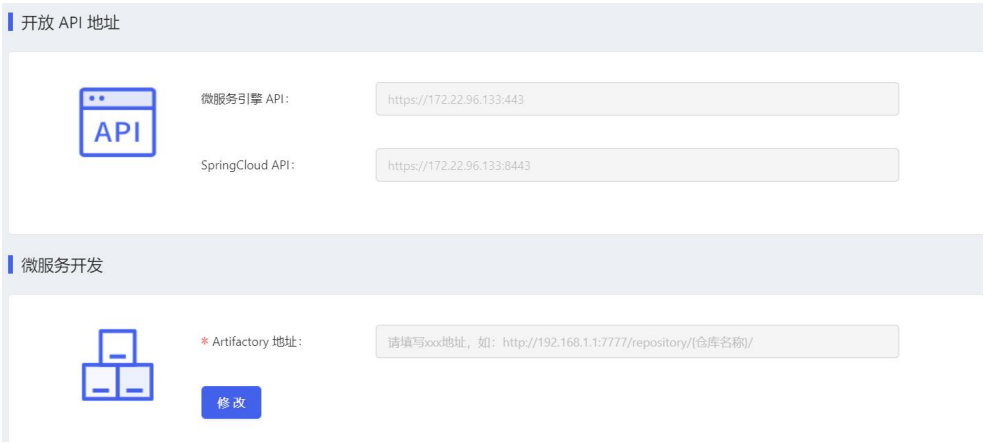
位置说明：[管理工作台/微服务配置]。

内容说明：

- 支持显示开放的 API 地址(只读)，包括微服务引擎 API、Spring Cloud API。
- 可配置微服务开发 Artifactory 地址，即微服务开发中依赖组件所在的私服地址。

权限说明：系统管理员、平台管理员、基础设施管理员可见，可操作。

图 5-3 微服务环境设置



5.3 微服务治理框架部署方式

结合多租户使用方式和资源使用分配两方面来考虑使用共享还是独享微服务治理框架。用

户在使用平台时，租户具体是对应业务域、子公司、公司部门还是项目组，使用方式不同，选择结果也会不同。

前提条件：需要准备所需中间件 MySQL、ES、Redis。

- MySQL、ES、Redis 用户自定义配置，可使用第三方服务，也可使用平台中间件。
- 使用第三方服务，版本兼容说明
  a. ES 版本支持 6.4.0、6.8.x
  b. MySQL 版本支持 5.7 及以上
  c. Redis 版本支持 5.x
- 使用平台中间件，请到[服务目录/应用]中选择对应的入口进行部署。详细说明请参考<TCE _V5.3.0_时速云容器云用户手册>。使用平台中间件的注意事项如下：
  a. 平台中间件 MySQL：暂不支持，建议单独部署。
  b. 使用平台中间件 Redis 5.0 版本：
  - 集群模式支持一主多从和 Cluster 模式。
  - 若微服务组件中 spring-cloud-api 服务和 redis 部署同一个集群中，两种方式（可集群外访问（不支持哨兵）、仅在集群内访问）都支持。
  - 若微服务组件中 spring-cloud-api 服务和 redis 未部署到同一个集群中，仅支持可集群外访问，并确保网络互通。

  c. ES 版本支持 6.8.x。

图 5-4 服务目录-应用页面

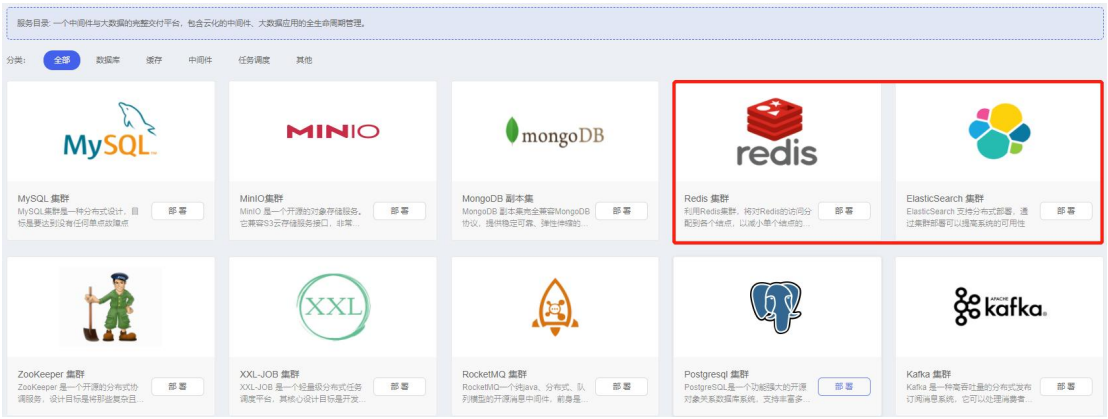


图 5-5 ES 部署页面，

5.3.1 共享微服务治理框架

平台管理员可为某个集群创建共享微服务治理框架，配置其基本信息，并对基础服务组件的安装与卸载操作控制；其他租户的租户管理员申请对应集群的共享微服务治理框架，审批通过后可进行微服务治理。

安装入口：

- 管理工作台，进入[管理和审批/微服务配置/微服务治理框架管理]页面，选择集群进行安装。
- 系统租户，进入[安全和运维/微服务运维/微服务治理框架管理]页面，选择集群进行安装。

因为共享微服务治理框架安装时选择租户、项目，默认租户为“系统租户”，项目需要平台管理员自行创建，其命名空间定义为 system-mf，项目名称可自定义。故这两个入口安装成功后，都是共享微服务治理框架。

5.3.1.1 安装基础服务组件

安装基础服务组件的操作步骤如下：

(1) 进入[微服务治理框架管理]页面，选择需要安装的集群。

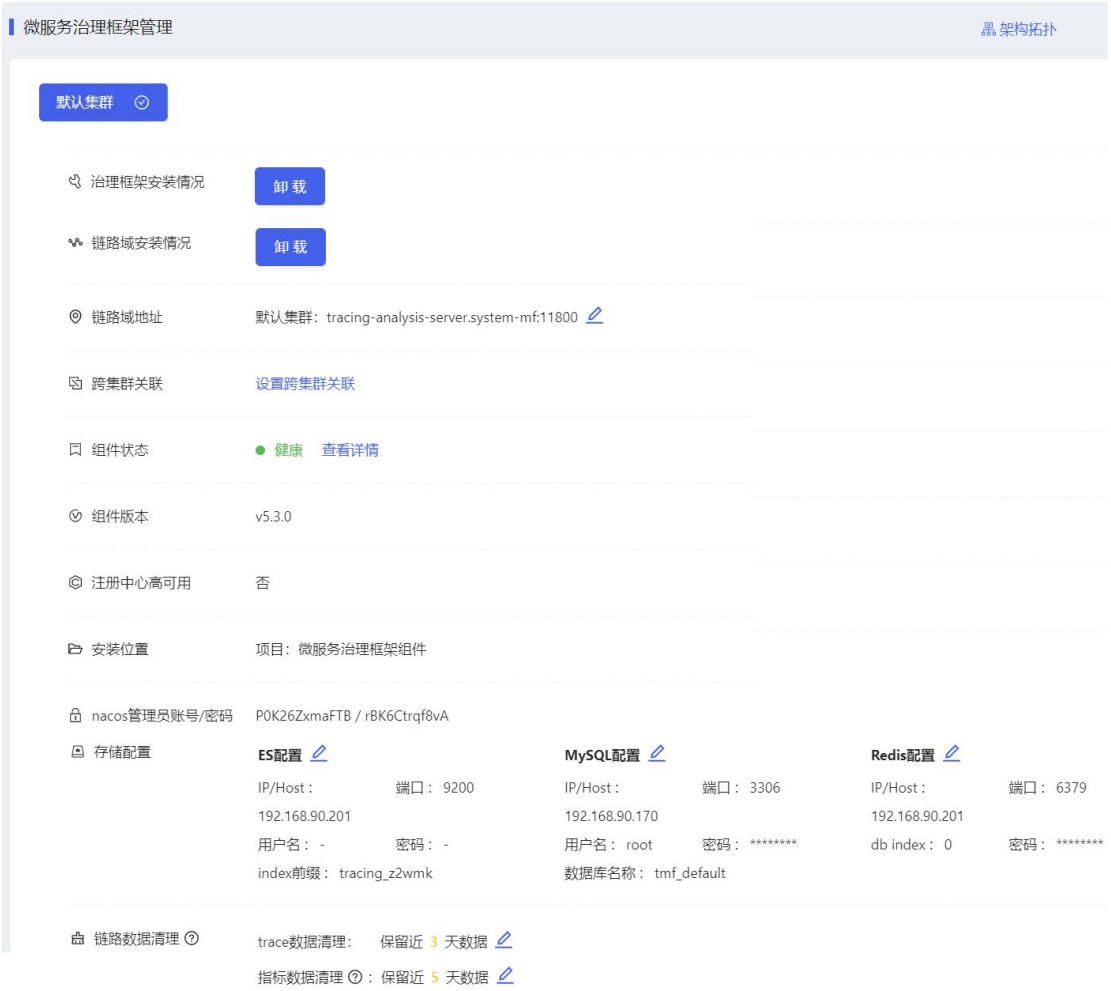
(2) 若该集群暂未安装基础服务组件，治理框架安装情况显示“安装”，点击安装，进入组件安装页面，如下图所示。

图 5-6 微服务治理框架配置弹框

- 选择注册中心是否高可用。
- 选择微服务治理框架组件所安装的项目。
- 链路域设置：选择是否安装链路域(skywalking)，
  - 选择是则在框架内安装链路域，链路追踪数据默认上报至本集群链路域（内网）。
  - 选择否则不安装链路域，可选择其他集群链路域地址（外网）上报链路追踪数据。
- 配置 ES 的 IP/host、端口（选填，默认 9200）、用户名、密码、索引前缀，其中地址配置信息支持服务名输入形式；该数据库主要为链路追踪相关功能提供存储。
- 配置 MySQL 的 IP/host、端口（选填，默认 3306）、用户名、密码、数据库名称，该数据库主要用于存储微服务治理相关的数据。
- 配置 Redis 数据库的 IP/host、端口（选填，默认 6379）、密码、db index（范围：0-15），该数据库主要用于缓存微服务治理相关的数据。

基础组件配置完成后，单击<确定>按钮，基础组件安装操作完成，进入组件启动页面，启动成功后，示例页面如下图所示。

图 5-6 组件启动页面



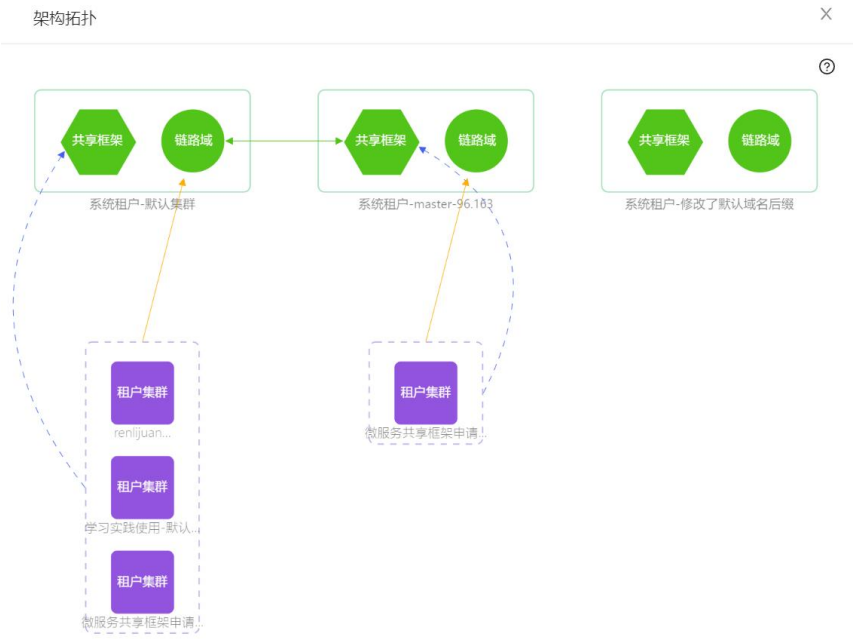
- 治理框架安装情况：框架整体安装情况，根据状态展示安装、卸载、重启等操作按钮
- 链路域安装情况(治理框架安装后显示)：治理框架存在的基础上可独立安装卸载
- 链路域地址：链路追踪数据上报的地址，支持修改。
- 跨集群关联：设置多个集群互相关联(需暴露注册中心外网地址)，关联后的集群可支持跨集群服务发现。

  一个项目授权给不同的集群，用户在每个集群的项目内创建相同的服务，并为该服务设置跨集群访问地址，当其中一个集群的服务出现异常，可调用到另一个集群的相同服务，从而实现跨集群的服务调用。
- nacos 管理员账号/密码：可查看 nacos 所有 namespace 内数据的账号密码。
- 安装成功后，还可分别查看整体框架组件的状态与版本、注册中心是否高可用、安装位置、存储相关配置。
- 用户可按业务需要配置链路数据清理，包括 trace 数据清理、指标数据清理周期。

- 若框架安装失败安装情况，会显示“重启”按钮及 job 初始化状态图标，可根据提示信息及日志定位问题后，点击重启重新修改配置，启动框架组件。
- 架构拓扑：展示集群间框架及链路域拓扑关系；

  若服务跨集群部署，则展示实例真实的情况，例如：服务 1，跨集群 A、B 部署，则拓扑图中，集群 A 内，展示服务 1 及 A 集群所有实例，集群 B 内也展示服务 1 及 B 集群所有实例，示例页面如下图所示。

图 5-7 架构拓扑



说明

- 微服务组件会安装到此项目“容器服务”中，如果对组件服务重启、删除等敏感操作，会影响所有注册到此框架的微服务，影响用户业务，用户需关注权限设置。
- 系统管理员在系统租户选择微服务框架所在项目,在[容器服务/服务配置/普通配置]可编辑框架配置 sc-components-env 相关环境变量。
- IP/host 支持数字、小写字母、“.”、“-”随意组合。支持使用集群 service 名称进行通信。
- 当不同微服务治理框架使用同一个 ES 或者 MySQL 时，ES 的 index 前缀，MySQL 数据库的名称，需要唯一。
- 链路数据保留天数修改后，需手动重启微服务组件 tracing-analysis-server。
- 编辑 ES\MySQL\Redis 配置时，需要重启相关微服务组件才生效。
  - ES 编辑成功后，需手动重启微服务组件（tracing-analysis-server）方可生效。
  - MySQL 编辑成功后，需要手动重启微服务组件（spring-cloud-discovery，spring-cloud-gateway，spring-cloud-config-extension）。
  - Redis 编辑成功后，需要手动重启微服务组件（spring-cloud-gateway）。

5.3.1.2 卸载基础服务组件

步骤如下：安装成功后，点击<卸载>，弹出框中显示卸载提示信息，卸载前可更换框架链路域，若无需更换输入“卸载”再次确认后，卸载成功。

图 5-8 微服务框架卸载弹框提示

5.3.1.3 微服务组件

支持平台管理员对微服务组件进行管理，包括对组件的重启、停止、启动、重新部署、路由等操作。通过切换选择集群，来管理不同集群下的微服务组件。

微服务组件包含 6 个，如下表所示。

表 5-1 服务组件说明

功能	组件
注册与配置中心	spring-cloud-discovery
微服务网关中心	spring-cloud-gateway
配置中心扩展	spring-cloud-config-extension
链路追踪	tracing-analysis-server
链路监控	tracing-analysis-web
应用诊断	spring-cloud-arthas

操作步骤如下所示。

(1) 进入[微服务配置/微服务治理框架管理]，点击查看组件，选择需要管理的组件，右侧下拉框中可分别点击重启组件、停止组件、启动组件、重新部署按钮。

(2) 在弹出的提示框中，点击<确定>即可分别完成以上操作。

(3) 水平扩展功能可在[容器服务/容器应用/服务]中进行操作。

(4) 设置组件代理可在系统租户下，选择共享微服务框架所在项目，在[容器服务/网络管理/应用路由]中为 spring-cloud-gateway 或其他组件设置服务代理。

说明

- 微服务治理框架安装后，默认给 spring-cloud-discovery、spring-cloud-gateway 两个微服务组件已完成链路配置，从而支持链路追踪相关功能。
- 暂不支持在平台上对此两个微服务组件进行链路移除、开启 or 关闭采集操作，支持修改 Apdex 阈值和采样率，请谨慎操作。

5.3.2 独享微服务治理框架

为满足不同用户需求，租户管理员可创建独享微服务治理框架，仅供租户内成员使用。

- 位置说明：[微服务运维/微服务治理框架管理]。
- 权限说明：租户管理员、平台管理员、系统管理员可见、可操作。

租户管理员可为某个集群创建独享微服务治理框架，配置其基本信息，并对基础服务组件的安装与卸载操作控制。

前提条件：判断是否申请共享微服务治理框架，如果申请，需要申请或解除授权后才可创建独享微服务治理框架。

入口：[微服务运维/微服务治理框架管理]

安装及卸载均和共享框架一致，不再赘述。

说明

- 选择项目，可选择当前租户下已创建并授权该集群的项目。
- 其他说明，详见共享微服务治理框架-安装基础服务组件处的说明。
- 租户管理员在该租户微服务框架所在项目，[容器服务/服务配置/普通配置]可编辑框架配置 sc-components-env 相关环境变量。

5.4 开启微服务治理

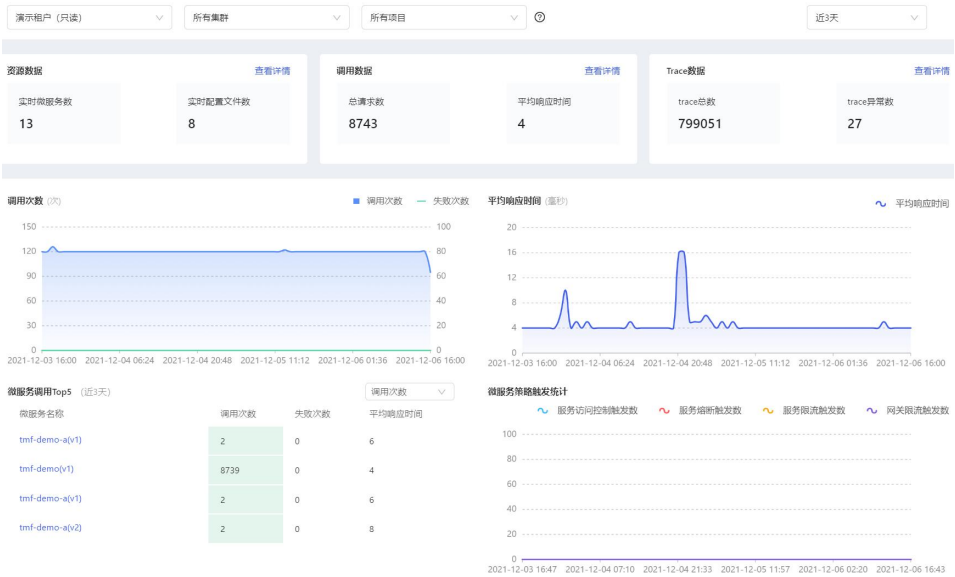
项目管理员到[微服务运维/运维开关]中开启微服务治理开关，开启后，平台容器部署微服务注册时，会自动注入相关环境变量（管理员在[容器服务/服务配置/普通配置]，tmf-env-cm 可编辑），快速接入微服务治理。若卸载微服务框架后重新部署，则开关需要手动关闭，并重新打开。

6 用户指南

6.1 微服务概览

进入[微服务治理平台/微服务概览]页签。根据用户角色筛选可选租户、集群、项目。默认选中当前租户，所有数据，除微服务数、配置文件数外，都受时间选择的控制，默认选择“近一天”，支持近一天、近三天、近一周。统计信息说明如下所示。

图 6-1 微服务概览页面



- 实时微服务数、实时配置文件数（配置文件不区分配置分组，统计所有分组的文件总数即可）。仅统计当前时刻的数据，不支持按时间段查询。
- 总请求数、平均响应时间（统计所有请求，包括来自微服务网关的以及服务之前的调用，以下两项亦如此）
- trace 总数/异常数。
- 微服务调用 Top5：根据所选时段，按调用次数、失败次数、平均时延三个维度排序展示 Top5 服务。

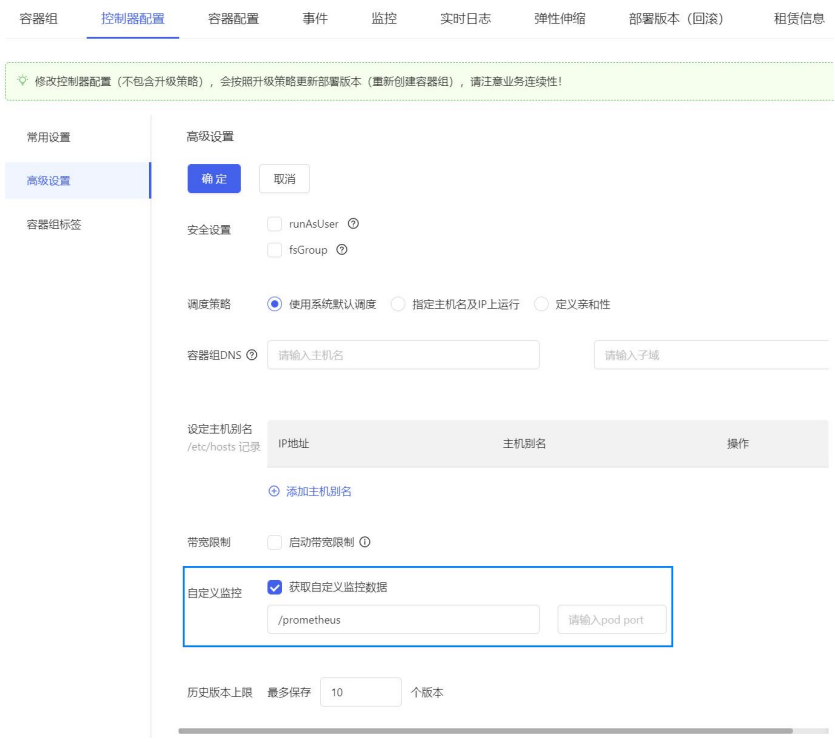
- 微服务治理统计：统计所选时段内本项目所有服务发生的短路熔断数、触发限流数。
- 微服务策略触发统计：统计服务访问控制触发数、服务熔断触发数、服务限流触发数、网关限流触发数。

说明

- 平台部署后，集群监控数据清理时效，默认是保留近 1 天的监控数据；可联系基础设施管理员按需调整。
- 当平台、系统管理员，选择“所有租户”时，点击查看详情，横坐标按租户/集群/项目分别统计数据，对比查看。
- 当租户管理员，选择某个租户/集群，项目为“所有项目”时，点击查看详情，横坐标按项目分别统计数据，对比查看。

PaaS 容器服务部署时需要在其“高级配置”中添加自定义监控配置，用来采集策略触发数据，如下图所示。

图 6-2 容器服务部署应用-高级设置入口



6.2 微服务开发

支持直接创建微服务工程并下载，简化微服务工程的初始搭建，省去样板化配置步骤，直接快速进入业务代码开发阶段。

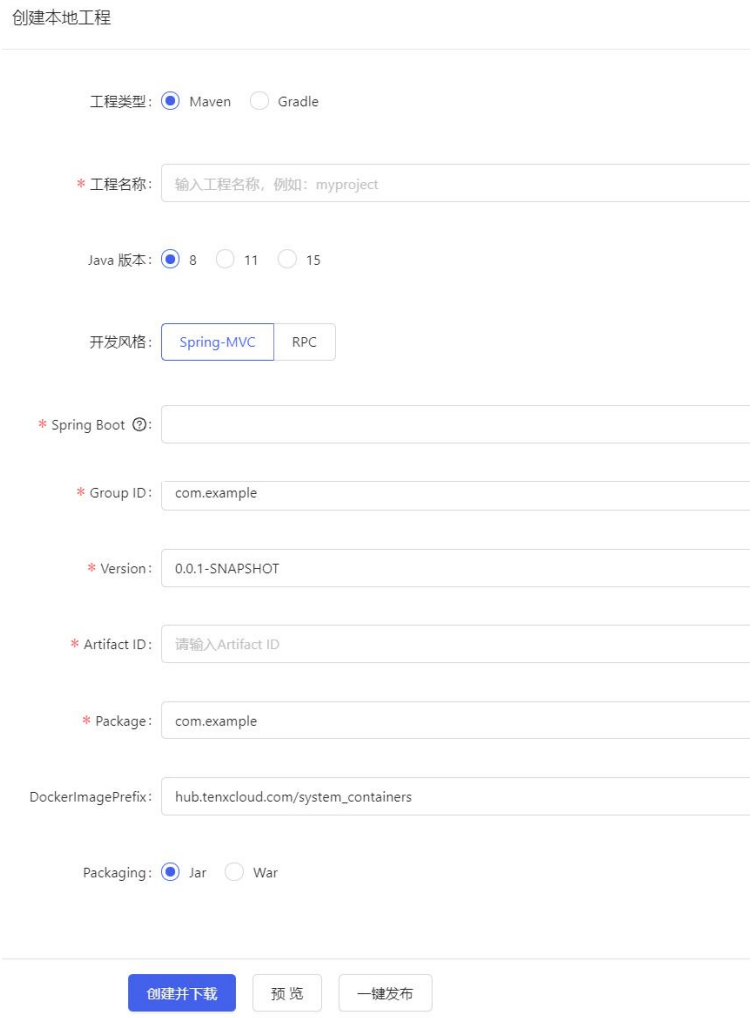
前提条件：确保配置了全局 Artifactory 地址，参考 5.3 微服务环境配置。

6.2.1 创建工程

创建工程的操作步骤如下所示。

(1) 进入[微服务治理平台/微服务开发]页面，创建本地工程。

图 6-3 微服务开发页面



(2) 生成工程配置项说明如下所示。

- 工程类型：用户可选择不用的项目类型，支持 Maven、Gradle。
- 工程名称：工程名称，支持由 1-24 个字符的小写字母、数字、中划线-组成，以字母

开头。

- JAVA 版本：支持 JAVA 8、JAVA 11、JAVA 15，缺省配置 JAVA 8。
- 开发风格：系统提供了 RPC、Spring-MVC 两种开发模式供选择，缺省配置 Spring-MVC。
- Spring Boot Version：支持 2.1.5，2.3.7 两种版本，缺省配置 2.1.5。
- 添加依赖：支持多种依赖，包括微服务组件依赖与其他依赖。
- GroupID：对应源码 maven 工程的 Group ID。
- Version：工程版本号。
- Artifact ID：对应源码 maven 工程的 Artifact ID。
- Package：对应源码 maven 工程中的 package 名称。
- Packaging：可选择 packing 的类型，可选 Jar、War。
- DockerimagePrefix：Docker 镜像前缀。

(3) 点击<预览>，可查看工程代码结构和文件内容。选择的工程类型不同，预览页面也存在差异。

(4) 用户填写完工程相关信息，支持一键发布到代码仓库，实现快速结合 Devops 产品使用，单击<一键发布>按钮，进入代码仓库配置页面，如下图所示，配置已有的 GitLab 代码仓库。用户也可选择自定义的 GitLab 代码仓库，配置对应的仓库地址以及仓库的 Access Token。

图 6-4 一键发布配置页面

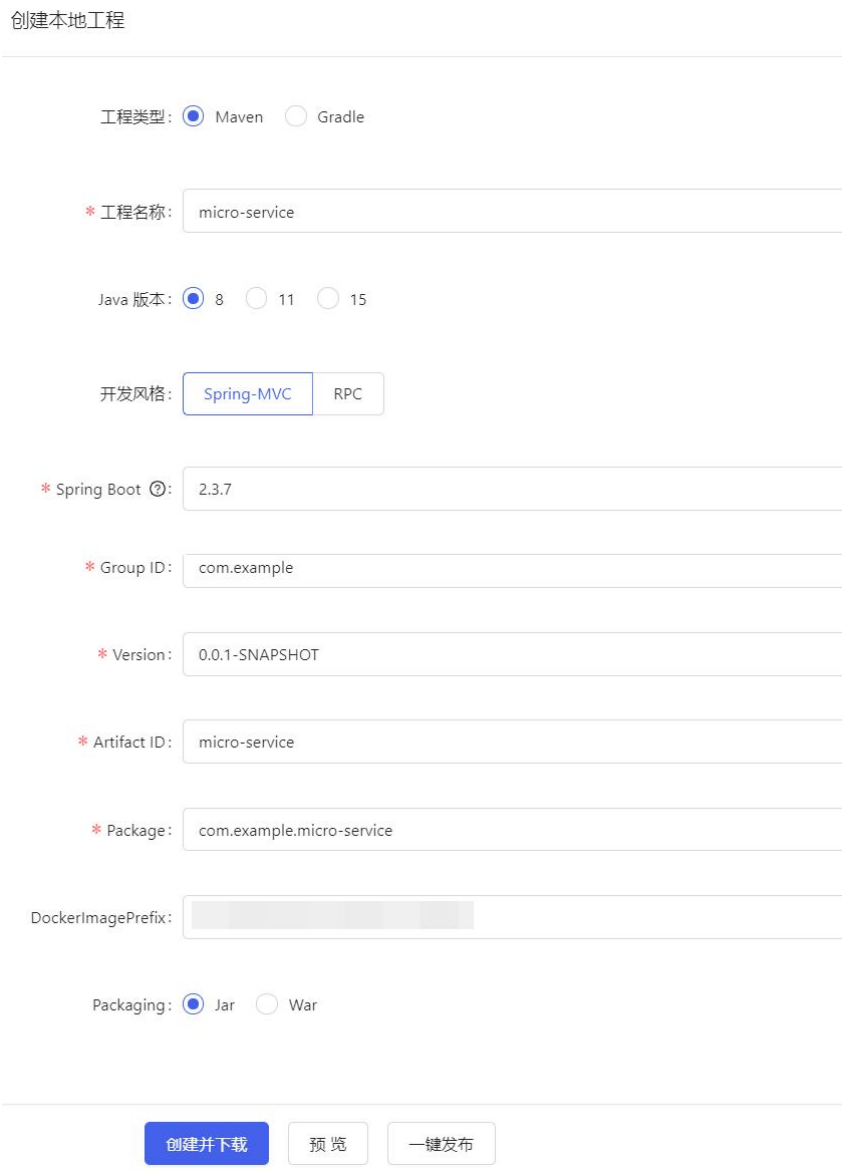
(5) 填写好相关配置信息后单击“创建并下载”按钮，即可完成工程创建。

6.2.2 MVC 开发风格工程示例

操作步骤如下：

(1) 创建 Spring-MVC 开发风格工程，开发风格中选择“Spring-MVC”，如下图所示。

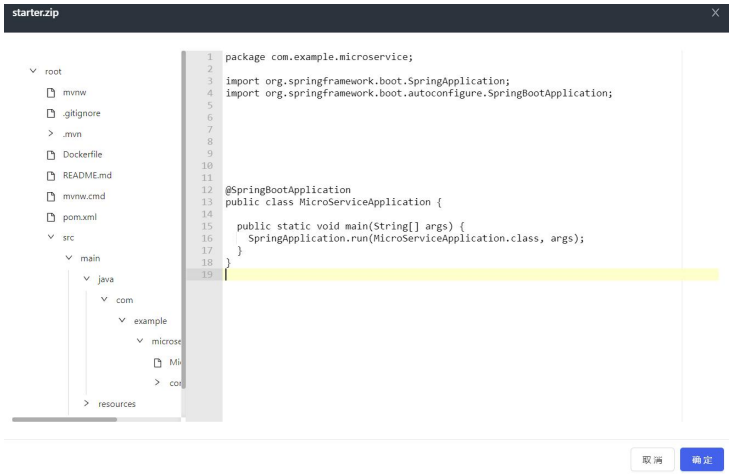
图 6-5 微服务开发-Spring-MVC 页面



(1) 点击“创建并下载”按钮，下载生成工程的 zip 压缩包。

(2) 点击<预览>，查看工程结构。

图 6-6 下载本地工程结构示意图



(3) 默认生成配置文件 application.yml，文件内容如下，用户可根据本地实际开发环境修改配置内容。

```
server:
port: 8080
spring:
application:
name: demo
profiles:
active: ${profiles.active:dev}
   datasource:
url:jdbc:mysql://${mysql_host:localhost}:${mysql.port:3306}/${mysql_db:test}?useUnicode=true&characterEncoding=utf-8
      username: root
      password: root
```

(4) 单击<一键发布>按钮，进入代码仓库配置页面，把该工程发布到代码仓库。

(5) 容器部署此服务(操作参考 3.2 快速接入微服务注册中心)，本工程在创建时增加了“注册中心”微服务依赖组件，在[微服务治理平台/微服务/服务管控]页面可看到刚刚注册的 demo 服务。

6.2.3 RPC 开发风格工程示例

6.3.3.1 RPC 开发风格工程示例-服务提供者

(1) 创建 RPC 开发风格工程，开发风格中选择“RPC”，服务类型选择“服务提供者”，如下图所示。

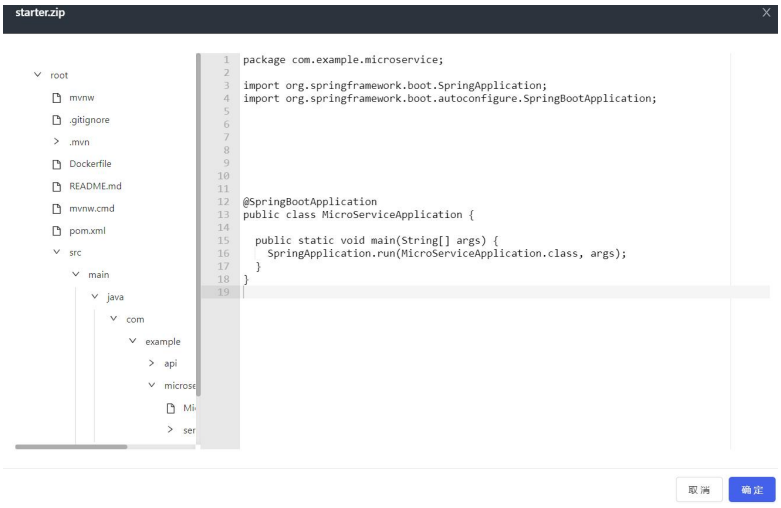
图 6-7 微服务开发-RPC 页面

(2) 配置工程对应的参数，单击<创建并下载>创建工程。

(3) 单击<预览>按钮，预览工程，工程页面显示如下图所示。

(4) 单击<一键发布>按钮，进入代码仓库配置页面，把该工程发布到代码仓库。

图 6-8 RPC 工程预览页面



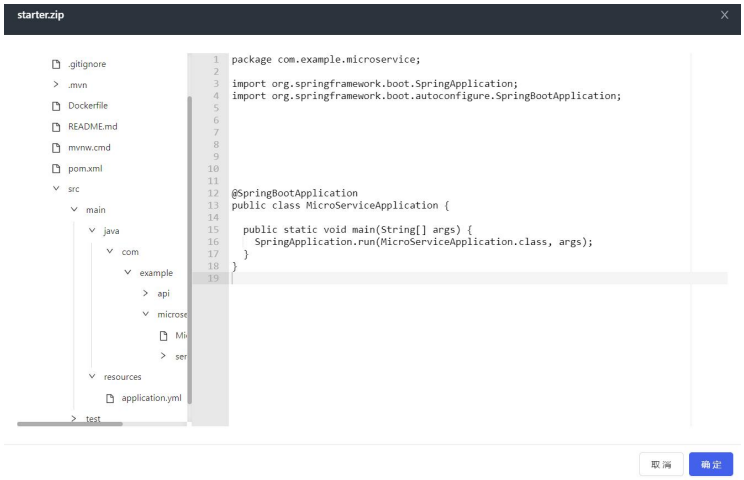
6.3.3.2 RPC 开发风格工程示例-服务消费者

(1) 创建 RPC 开发风格工程，开发风格中选择“RPC”，服务类型选择“服务消费者”。

(2) 配置工程对应的参数，单击<创建并下载>创建工程。

(3) 单击<预览>按钮，预览工程，工程页面显示如下图所示。

图 6-9 RPC 工程预览页面



(4) 单击<一键发布>按钮，进入代码仓库配置页面，把该工程发布到代码仓库。

(5) Paas 平台部署生产者示例服务（操作步骤略）。

6.2.3.1 隐藏服务与取消隐藏

操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，服务列表页找到需要隐藏的服务，在列表右侧操作列中，点击隐藏服务。

(2) 在弹出的确认框中点击<确定>，即可隐藏成功，此时服务状态更新为“不可被发现”。

(3) 同理，不可被发现状态的服务点击取消隐藏按钮，服务将被取消隐藏，服务状态更新为“可被发现”。

说明

隐藏服务即隐藏其全部实例，此时详情页面的实例对应操作按钮置变为取消隐藏。

6.3 微服务管理与治理

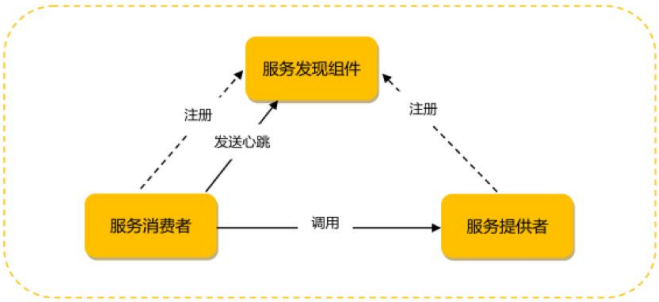
6.3.1 服务管控

只需要在代码中引用注册中心即可实现自动注册，并可以在服务管控列表设置服务是否允许发现或隐藏，并通过心跳来监测服务的状态。接入方法可参考 3.2 快速接入微服务注册中心。

若服务注册后未能在服务管控页面正常展示，管理员可参考 6.5.1 章节为注册中心组件设置路由，通过 nacos 原生页面，查看服务是否注册到 nacos 上。

本节主要介绍基于 Nacos 的微服务注册与发现，Nacos 致力于帮助您发现、配置和管理微服务。Nacos 提供了一组简单易用的特性集，帮助您快速实现动态服务发现、服务配置管理。

图 6-10 Nacos 微服务注册与发现关系图



图中服务提供者、服务消费者及服务发现组件这三者之间的关系如下：

- 各个微服务在启动时，将自己的网络地址等信息注册到服务发现组件中，服务发现组件会存储这些信息。
- 服务消费者可从服务发现组件查询服务提供者的网络地址，并使用该地址调用服务提供者的接口。
- 各个微服务与服务发现组件使用一定机制（例如心跳）通信。服务发现组件如长时间无法与某微服务实例通信，就会注销该实例。
- 微服务网络地址变更时，会重新注册到服务发现组件。使用这种方式，服务消费者就无需人工修改提供者的网络地址了。

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，服务管控列表页签中，支持空服务隐藏与展示、全局路由配置、服务检索、环境变量相关说明查看，列表显示微服务的服务名称、服务分组、跨集群实例状态、当前集群实例状态、服务类型、服务状态、注册方式、访问地址、操作。

  - 服务名称：微服务名称。
  - 服务分组：分组内服务可互相访问；分组间服务隔离，不可互相访问。
  - 跨集群实例状态：跨集群同名服务实例状态展示。
  - 当前集群实例状态：当前集群实例状态， 运行实例/总实例
  - 服务类型：SpringCloud、Dubbo、其他类型。
  - 服务状态：可被发现、不可被发现，状态变化与隐藏服务和取消隐藏对应。
  - 注册方式：当前服务是手动注册还是自动注册的。
  - 访问地址：展示配置的路由地址，未设置时会有提示“去设置”，点击会跳转路由管理页面。
  - 操作，详见 6.3.1.1 隐藏服务与取消隐藏。

图 6-11 服务管控列表页面

(2) 进入[微服务治理平台/微服务/服务管控]页面，点击组件列表页签，显示微服务所有组件、名称、服务地址、状态、实例数量、安装时间。

图 6-12 微服务组件列表页面

6.3.1.1 手动注册

手动注册的服务一般是普通的 HTTP 服务，无法依赖注册中心的服务，非 SpringCloud、Dubbo 服务。注册后的服务可被注册中心其他服务发现，可以通过网关进行路由/限流/负载均衡等配置。

手动注册服务的操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，在当前页面，单击<添加服务>按钮，进入服务配置页面。

(2) 配置服务的服务分组，服务名称，添加实例，配置健康检查方式，各参数说明如下所示。

- 服务分组：必填，由 3-63 个英文字母、数字或特殊符号"-"、"."、":"、"_"组成，且以字母或数字开头或结尾。
- 服务名称：必填，由 3-63 个英文字母、数字或特殊符号"-"、"."、":"、"_"组成，且以字母或数字开头或结尾。
- 添加实例：可添加多组（服务地址、服务端口），服务地址+服务端口做为一个服务内唯一校验。
- 健康检查地址：默认 TCP 探测，其中 TCP 探测默认使用各实例的 IP 加端口，示例 IP:port。HTTP 探测默认使用各实例的 IP 加端口，示例 https://ip:port/info，输入检查路径，例/info。

图 6-13 手动注册服务

(3) 服务参数配置完成后，单击<确定>按钮，手动注册服务创建成功。

说明

- 手动注册时，也可以不添加实例，添加成功后到详情页面“添加实例”。
- 手动注册的服务目前支持普通 HTTP 服务。
- 手动注册的服务目前只支持隐藏服务、隐藏实例，删除服务/实例，网关上的负载均衡/限流/路由。
- 自动注册服务的实例列表中，显示“心跳时间”，手动注册服务的实例列表中，显示“更新时间”。

(4) 进入服务详情页，进入实例列表，单击<添加实例>按钮，可添加多个实例。

实例列表展示内容是：实例 ID、实例状态、服务地址、服务端口、心跳时间/更新时间、当前状态、操作（隐藏实例/取消隐藏、删除实例）等服务属性。

6.3.1.2 全局路由配置

全局配置默认敏感 header，配置之后系统将不向下游的服务传递以下敏感 Header，若未添加敏感 Header 代表向下游服务传递所有 Header。配置的操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，在当前页面，单击<添加服务>按钮，进入服务配置页面。

(2) 单击<全局路由配置>按钮，进入敏感 header 配置页面，如下图所示。用户根据业务需要可配置多个 header。

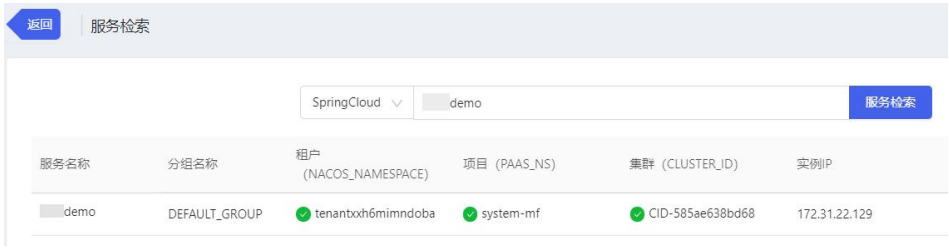
图 6-14 路由配置页面

(3) 配置完成后，单击<确定>按钮，配置全局路由操作完成。

6.3.1.3 服务检索

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，在当前页面，单击<服务检索>按钮，进入服务检索页面，选择要检索的服务类型，系统支持 SpringCloud、Dubbo 类型的服务。本例中选择检索 SpringCloud 类型服务，输入完整的服务名称，单击<服务检索>按钮，检索服务操作完成。示例页面如下图所示。

图 6-15 服务检索页面



6.3.1.4 查看元数据

系统支持查看服务的元数据，查看的操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，找到要查看元数据是的服务，点击服务名称链接，进入服务详情页面。

(2) 点击“查看服务元数据”、或者点击实例列表操作列的“查看元数据”即可查看对应的元数据。

6.3.1.5 查看自动接入引导

系统提供微服务自动接入引导，用户可根据引导做对应的配置，查看自动接入引导的步骤如下所示。

进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，在当前页面，单击“查看自动接入引导”链接，进入查看页面，如下图所示

图 6-16 接入引导

6.3.1.6 主备服务故障切换

当服务发生故障不能正常响应业务请求时，通过注册中心进行主备服务设置，当触发故障策略时，注册中心将上游请求切换到备服务（Dubbo 服务暂不支持主备切换）；同时支持手动切换主备，实现由主到备，或由备到主，正常提供服务，不间断业务。设置备服务的操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，找到要设置备服务的服务，单击<设置备服务>按钮，进入备服务配置页面。

(2) 配置备服务对应的参数，各参数说明如下所示。

- 选择备服务：服务分组默认选中主服务所在分组，暂不支持修改；选择此分组下的服务（过滤主服务）。
- 故障切换类型：自动切换和手动切换，用户可按需配置，当选择自动切换时配置属性如下所示。
  - 实例健康检查：服务健康实例个数≤N 个时，进行故障切换
  - 异常调用：最近 1 分钟满足下面某一条件时，进行故障切换（同时只能设置一个维度）。调用失败率≥N%、连续调用失败次数≥N 次、调用失败总次数≥N 次、调用响应时间≥N ms。
  - 两种策略都设置时，触发任一个就进行切换，两种策略不可都为空。
- 当故障切换类型选择手动切换时，参数配置说明如下所示。
  - 手动切换：选择主服务 or 备服务，切换原因选填。
  - 开启手动切换，上游请求全部发送到设置的服务商，自动切换策略不再生效。

图 6-17 设置备服务

(3) 备服务参数配置完成后，单击确定按钮，主备服务故障切换设置成功。

6.3.1.7 删除空实例服务

当服务不存在任何实例后，可以通过操作“删除服务”操作来删除空实例服务，手动注册的服务需要删除全部实例后方能删除服务。

图 6-18 删除服务入口

6.3.1.8 服务管控详情

进入[微服务治理平台/微服务/服务管控]，在服务列表页点击服务名称，进入服务详情页查看服务详情、环境信息、JVM 监控、资源监控、日志信息、配置信息、路由规则、限流规则、熔断降级、访问控制、链路配置、应用诊断等信息。

服务详情

(1) 进入[服务治理平台/微服务/服务管控]，服务列表页找到要查看实例列表的服务，点击服务名称，进入服务详情页面，示例页面如下图所示。

图 6-19 服务详情页面

(2) 默认展示服务详情页面，在该页面可以查看服务类型、服务分组、注册方式、访问地址、备服务、服务版本数、自动切换策略、配置仅网关访问服务时的负载均衡策略、查看服务的元数据、服务跨集群访问。

查看服务的实例列表，对于每个实例，用户可执行“查看元数据”、“隐藏实例”等操作。

(3) 服务跨集群访问：用于实现跨集群的服务发现。当管理员在微服务治理框架处配置了关联集群后，需要为服务设置跨集群访问地址，关联集群中的服务才能访问到本集群服务实例。

- ✓ 输入域名（默认端口 80）或 IP:Port
- ✓ 域名规则：支持数字、小写字母、“.”、“-”随意组合。支持"aa-aa.tmf-middle"等。

图 6-20 编辑服务跨集群访问

环境信息

(1) 进入[微服务治理平台/微服务/服务管控]，服务列表页找到要查看环境信息的服务，点击服务名称，进入服务详情页面。

(2) 默认展示实例列表页面，切换至环境信息页面。

JVM 监控

服务详情页显示“JVM 监控”tab，监控容器组 JVM 数据，用户可根据业务需要配置对应的进行查看。

- 内存统计内容：堆内存的最大内存和使用内存、Old Gen、Survivor、Space、Eden Space；非堆内存的最大内存和使用内存、Code Cache、Metaspace、Permgen；垃圾收集的 YoungGC 次数、OldGC 次数、YoungGC 耗时、OldGC 耗时。
- 线程统计图：活跃线程数、死锁线程数、守护线程数、峰值线程数。

图 6-21 JVM 监控页面-内存监控

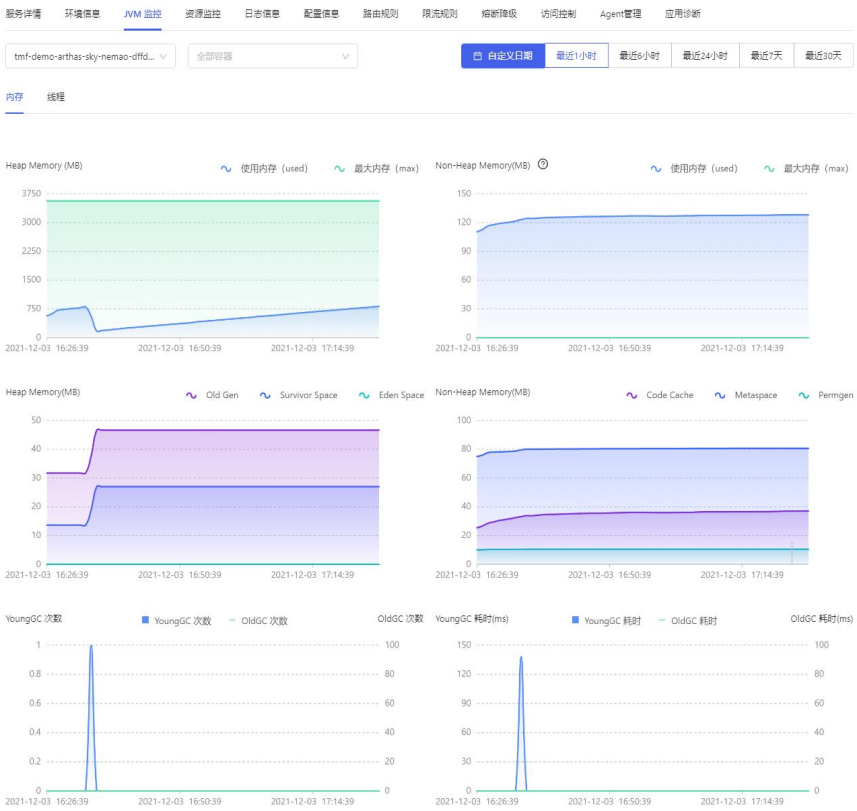
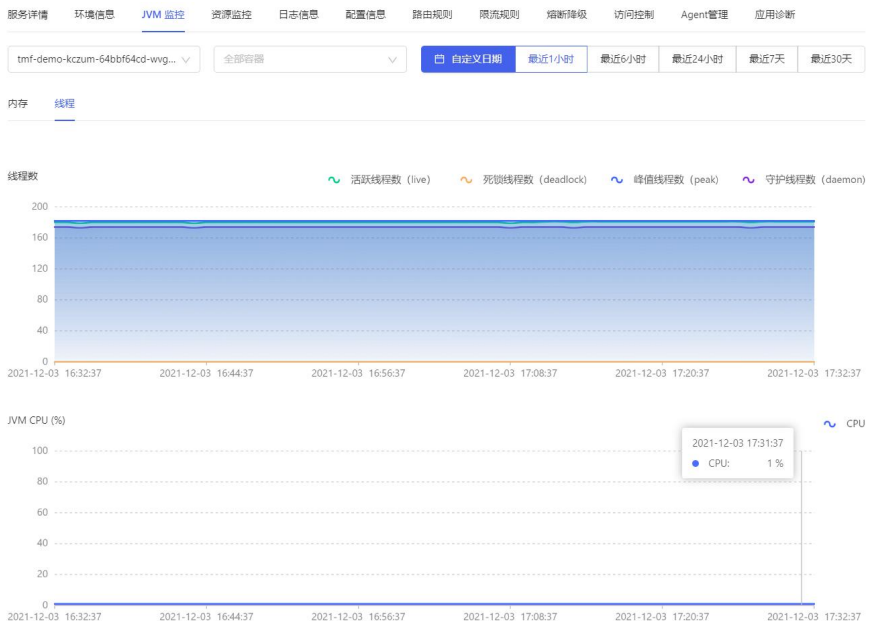


图 6-22 JVM 监控页面-线程监控



说明

暂不支持容器级查看 JVM 数据。

资源监控

支持按容器组、容器筛选查看监控指标，便于发现问题及排查故障，用户可自定义查看时间段，开启实时查看，如下图所示。

图 6-23 资源监控页面（部分）

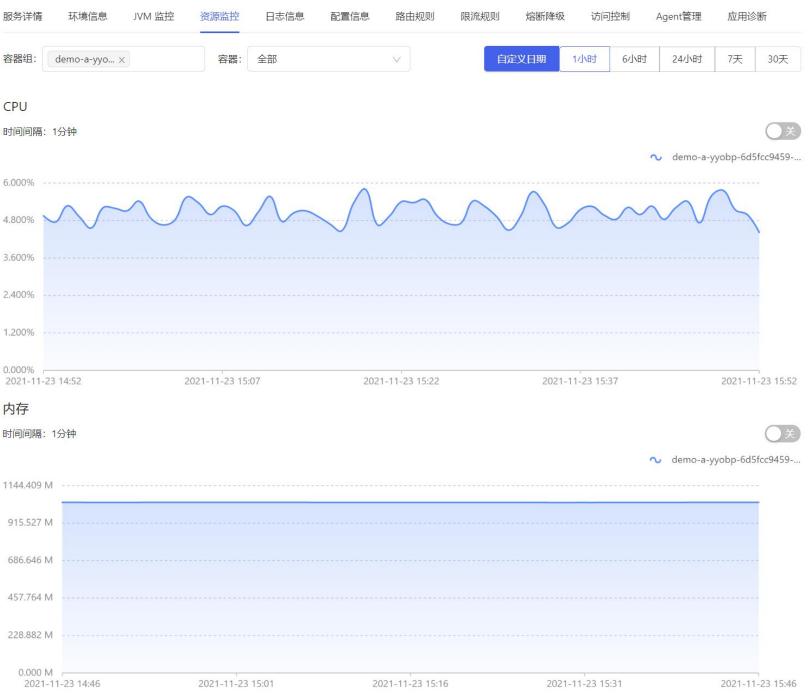
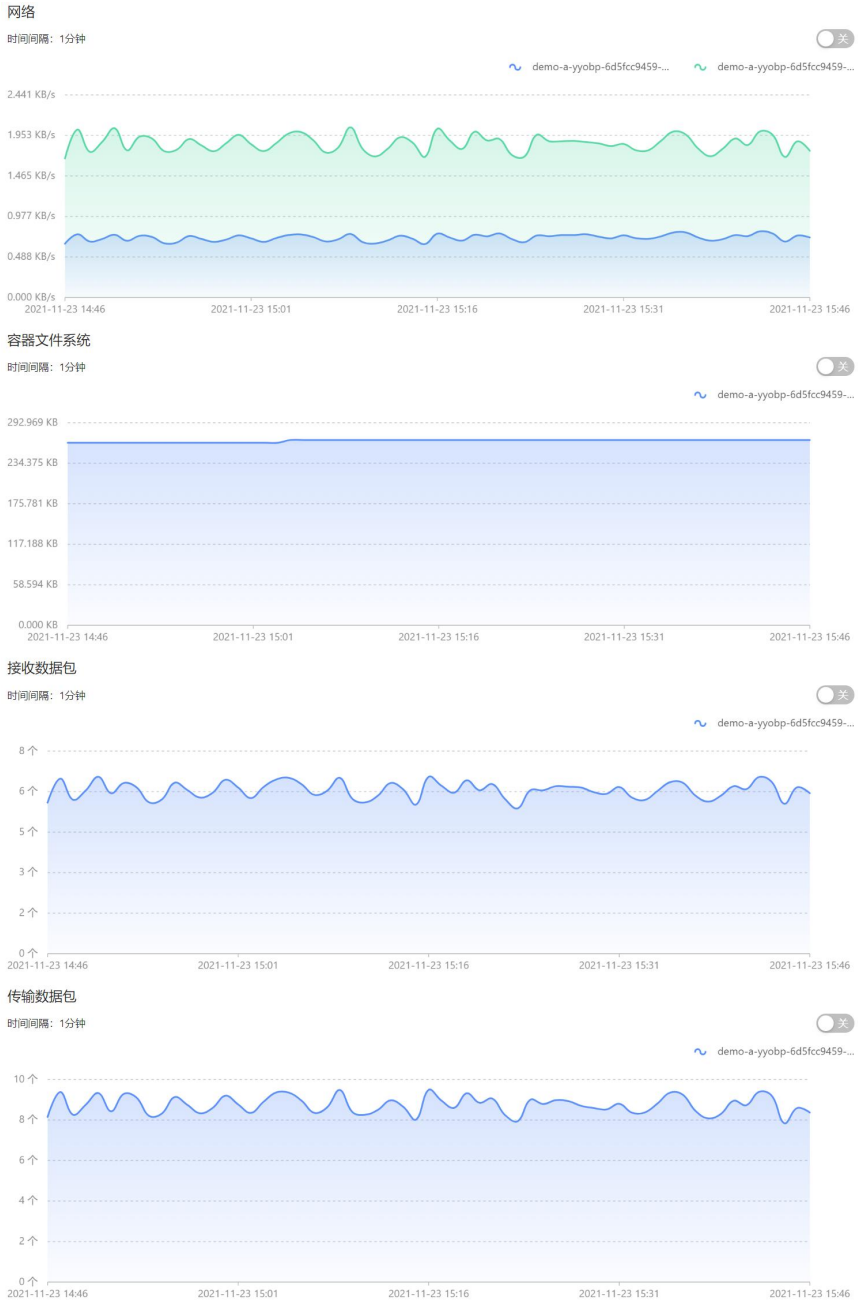


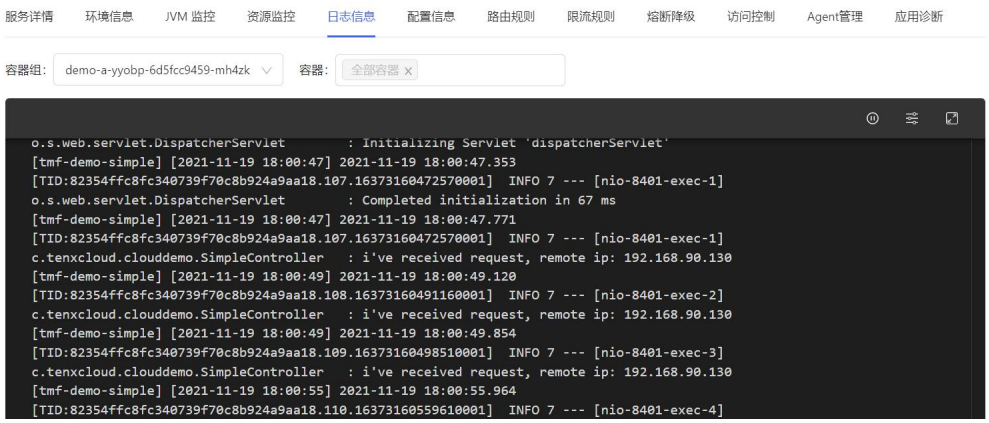
图 6-24 资源监控页面（接上图）



日志信息

可以筛选容器组、容器查看日志信息。日志查看页面支持全屏显示。

图 6-25 日志信息页面



配置信息

配置，引入了配置中心依赖的微服务，且添加或导入配置文件后，才能展示其配置文件内容。详细内容请参看 6.3.2 节所述。

路由规则

路由规则详细描述请参看 6.4 节相应内容。

限流规则

服务间限流指的是其他服务请求到本服务，超出限流阈值，本服务不再响应请求。

- 系统规则

系统规则是从微服务所有入口流量进行控制，提供四种维度的流控策略，保证微服务整体的稳定性。支持基于四种维度的服务限流策略，分别支持启用、停用功能，操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，找到要添加系统规则的服务。

(2) 点击对应服务的名称链接，进入服务详情页面，点击“系统规则”页签，勾选对应的系统规则，四种规则的配置互相不影响，用户可根据业务需要做对应的配置，各规则的详细说明如下所示，对应的配置页面如下所示。

- 基于平均 RT 策略：微服务所有入口流量的平均 RT 达到阈值即触发限流保护，单位 ms(统计每秒的平均 PT)。
- 基于入口 QPS 策略：微服务所有入口流量的 QPS 达到阈值即触发限流保护。
- 基于并发线程数策略：微服务所有入口流量的并发线程数达到阈值即触发限流保护(实时并发线程数)。
- 基于 CPU 使用率策略：当系统 CPU 使用率超过阈值即触发限流保护(一秒钟刷新一次，刷新的时候是获取的当前 CPU 利用率)。

图 6-26 配置页面

(3) 对应的规则配置完成后，单击<保存>按钮，系统规则配置成功。

- 流控规则

流控规则实现了接口级限流，对服务各实例内此接口的请求量进行限制设置的策略、阈值，在每个实例上生效，配置步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，找到要添加系统规则的服务。

(2) 点击对应服务的名称链接，进入服务详情页面，点击“流控规则”页签，进入流控规则配置页面，配置页参数说明如下所示，配置页面如下所示。

- 接口名称：例如 GET:/testA，仅支持填写一个接口，不支持特殊符号。
- 流控策略：QPS、线程数（线程数仅支持“直接拒绝”流控效果）。
- 流控阈值：整数（对于微服务拥有多个实例的情况，支持设置单个实例流控，例 2700QPS，那任一实例请求超过 2700QPS，则直接拒绝此请求）；流控策略选择“线程数”，统计的是实时线程数。
- 流控效果：支持两种效果“直接拒绝”、“匀速排队”。
  - 直接拒绝：对于超出请求的服务直接拒绝，返回默认的信息。
  - 匀速排队：对于请求服务设置超时时间，超过此时长，则直接拒绝排队请求。

图 6-27 流控规则配置页面

(3) 对应的规则配置完成后，单击<确定>按钮，流控规则配置完成。

用户可以根据业务要求编辑/停用/启用对应的流控规则策略。

- 网关限流

网关限流功能模块通过对经过网关的服务配置不同类型的限流规则实现对服务的限流，支持的类型有：URL、User、Origin。

(1) 进入[微服务治理平台/微服务/服务管控]，默认进入服务列表页面，找到要添加网关限流的服务。

(2) 点击对应服务的名称链接，进入服务详情页面，点击“限流规则”页签，进入限流规则配置页面，切换至“网关限流”页面，配置页参数说明如下所示，配置页面如下所示。

图 6-28 添加限流规则

- 限流类型：选择限流规则的限流类型，支持的限流类型有：User、URL、Origin。
- 限流阈值：配置限流规则的限流阈值，用户可按需进行设置。
- 窗口：配置限流规则窗口的时长。

熔断降级

前提条件：服务已配置熔断降级相关依赖，才可支持熔断/降级功能。

表 6-1 功能说明

功能	目的	适用场景
熔断	当 A 服务模块中的某块程序出现故障后，为了不影响其他客户端的请求而做出的及时回应。微服务应用中通常会存在多层服务调用，基础服务的故障可能导致级联故障，将不可用放大，拖垮整个系统，通过熔断机制解决这种服务提供者不可用导致服务调用者不可用的问题	熔断在服务请求处理满足触发条件时产生作用。进入熔断状态后，hystrix 会认为被请求的服务已经无法处理请求，在第一时间截断请求直接返回错误给调用者。 hystrix 会在熔断的持续时间内不再响应请求，时间过后，会再次尝试请求，如果服务未恢复正常，会继续熔断状态，若恢复正常可正常请求访问
降级	为了解决整体项目的压力，而牺牲掉某一服务模块而采取的措施。现有资源不够用时，使用降级机制，关掉部分不重要的服务，避免占用资源，保证主体业务不受影响	当出现服务吞吐量巨大，资源不够用等情况，可使用降级机制关掉部分不重要、性能较差的服务，避免占用资源，以保证主体业务功能可正常使用

说明

- 服务注销后，服务再次上线时保持设置的熔断降级策略不变。
- 若降级和熔断同时开启时，将执行降级策略。

- 添加熔断规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]，选择某服务，进入其服务详情页面，点击熔断规则页签。

图 6-29 微服务详情-熔断页面

(2) 点击“添加熔断规则”，输入接口名、熔断策略、触发条件、熔断时间窗。

- 接口名称、例如 GET:/testA，仅支持填写一个接口，不支持特殊符号。
- 熔断策略：支持三种熔断策略，包括慢调用比例、失败比例、失败数。

- 慢调用：某请求的 RT 超过设置的慢调用 RT 时，称为慢调用，在一个统计时间窗内，窗口请求数（需满足窗口最小请求数）的慢调用比例超出阈值时，该实例熔断，在熔断时间窗后恢复。
- 失败比例：在一个统计时间窗内，窗口请求数（需满足窗口最小请求数）的失败比例超出阈值时，该实例熔断，在熔断时间窗后恢复。
- 失败数：在一个统计时间窗内，窗口请求数（需满足窗口最小请求数）的失败次数超出阈值，该实例熔断，在熔断时间窗后恢复。

图 6-30 添加熔断策略

- 统计时间窗：用于统计熔断器关闭时的请求结果。
- 窗口最小请求数：配置熔断器可以计算错误率之前的最小请求数。
- 熔断时间窗：整数，单位 s，默认值 15s，需要大于统计时间窗。

(3) 熔断规则配置完后，单击<确定>按钮，熔断策略创建成功。

- 启用熔断策略

操作步骤如下：

进入[微服务治理平台/微服务/服务管控]，选择某服务，进入其服务详情页面，点击熔断规则页签，单击某条记录上的“启用”即可启用该规则。

- 停用熔断策略

操作步骤如下：

进入[微服务治理平台/微服务/服务管控]，选择某服务，进入其服务详情页面，点击熔断规则页签，单击某条记录上的“停用”即可停用该规则。

- 编辑熔断策略

操作步骤如下：

进入[微服务治理平台/微服务/服务管控]，选择某服务，进入其服务详情页面，点击熔断规则页签，单击某条记录上的“编辑”即可修改。

- 删除熔断策略

进入[微服务治理平台/微服务/服务管控]，选择某服务，进入其服务详情页面，点击熔断规则页签，单击某条记录操作下的“删除”即可删除规则。

- 开启/关闭降级策略

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]，选择某服务，进入其服务详情页面，点击降级页签，默认是关闭状态。

图 6-31 微服务详情-降级页面

(2) 点击开关，弹框提示是否开启，确认后开启成功。

访问控制

根据调用来源判断请求是否允许放行，设置微服务、IP 的黑白名单，保证服务访问来源的正确性，防止服务被恶意调用。微服务黑白名单控制哪些服务可以访问微服务；IP 黑白名单控制哪些 IP 地址可以访问微服务，配置页面如下所示。

- 黑名单：表示根据匹配规则匹配到的微服务都不允许访问当前服务。
- 白名单：表示根据匹配规则匹配到的微服务允许访问当前服务。

图 6-32 访问控制配置页面

用户可以根据业务要求编辑/停用/启用对应的访问控制策略。

链路配置

链路配置详细说明请参看 6.5.3 节所述。

说明

- 资源监控权限受工作负载中查看容器组监控、查看容器组详情控制，日志信息权限受[日志服务/查询分析/原始数据查询]控制。
- 手动注册服务详情 tab 页仅支持查看实例列表、路由规则、限流规则信息。
- Dubbo 服务仅支持服务注册与发现、配置中心管理功能，其他服务治理功能暂不支持。故 Dubbo 类型服务的详情页面仅展示实例列表、JVM 监控、日志信息、资源监控、配置信息、应用诊断、链路配置等页签。上方基本信息处不展示负载均衡设置。

应用诊断

链路配置详细说明请参看 6.6.3 节所述。

6.3.2 配置中心

Nacos 配置中心用于配置的管理和下发，可为用户程序提供配置查询、存储等功能，统一管理配置。

6.4.3.1 主要功能

- 配置集中管理：解决分布式环境下多个服务实例的配置统一管理问题。

- 系统运行期间可动态配置、更新：无需重启服务，支持动态配置文件并更新到用户程序。
- 查看示例代码：系统提供示例代码，用户可参考示例代码配置所需的代码。
- 导入导出：支持配置文件批量导入，导出。
- 历史版本：支持一键回滚到某一版本。

6.4.3.2 基本操作

发布配置

操作步骤如下：

(1) 进入[微服务治理平台/微服务/配置中心]页面，列表展示配置名称、配置分组、关联服务、操作（历史版本、编辑、删除）。

图 6-33 微服务配置中心页面

(2) 单击配置中心页的添加配置按钮，会进入到如下操作界面，输入项介绍如下所示。

a. 关联服务：可选项。

✧ 选择微服务（已按《微服务治理平台_代码接入指南》配置完成），配置名称、配置分组、配置格式均默认填充，置灰不可修改

✧ 选择微服务，若此服务没有配置正确，提示“暂无法找到关联的配置文件，请依据《微服务治理平台_代码接入指南》检查相关配置”

✧ 选择微服务，若如果此微服务已有创建关联的配置文件，此时提示“此微服务已创建配置文件”。 即一个微服务可关联多个配置文件，但同一时刻只能有一个生效，TMF 平台只展示生效的这个配置文件。故平台上微服务只能有一个配置文件。

b. 配置名称：可由 3-63 个英文字母、数字或特殊符号“-”、“.”、“:”、“_”组成，且以字母或数字开头或结尾，编辑时不可修改。

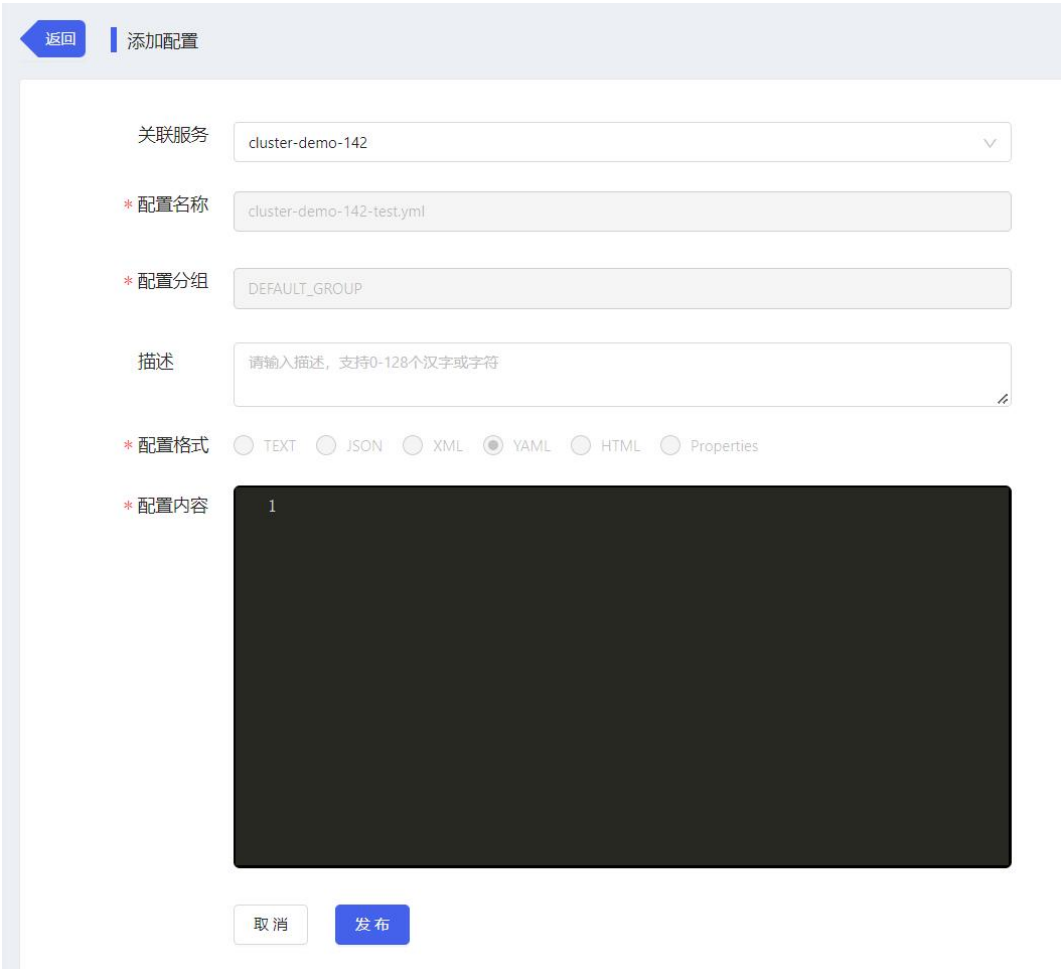
c. 配置分组：可由 3-63 个英文字母、数字或特殊符号“-”、“.”、“:”、“_”组成，且以字母或数字开头或结尾，编辑时不可修改。

d. 描述：支持 1-128 个汉字或字符。

e. 配置格式：可选 TEXT、JSON、XML、YAML、HTML、Properties。

f. 配置内容：用户配置的对应文件的内容。

图 6-34 微服务添加配置示例页面



(3) 填写好配置信息后单击发布，服务配置添加完成。首次发布后，即生成第一个版本，再次发布时，会做 diff 校验，查看当前编辑后与编辑前的配置内容对比，发布成功后，编辑后的为最新版本，其他版本支持回滚操作。

查看示例代码

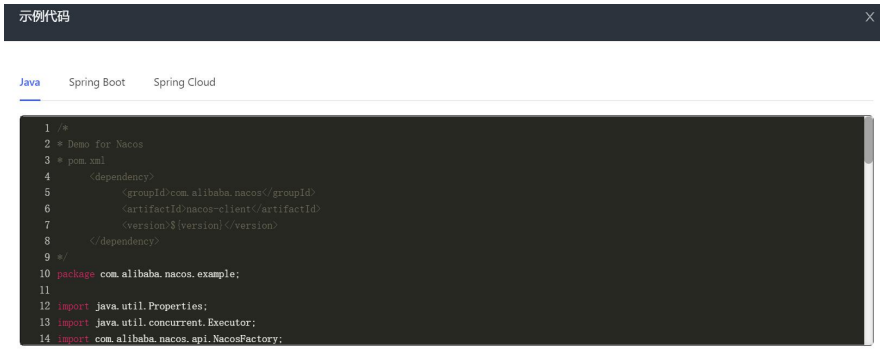
操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/配置中心]页面。

(2) 点击“示例代码”链接，进入代码查看页面，如下图所示。

系统提供多种类型的示例代码，例如：Java、Spring Boot、Spring Cloud 等类型的代码。

图 6-35 示例代码



查看配置详情

操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/配置中心]页面。

(2) 点击配置名称进入其详情页面。

a. 基本信息：配置名称、配置分组、配置格式、关联服务。

b. 配置详情：显示配置内容。

c. 历史版本：显示历史版本，历史版本支持查看详情和回滚操作。

版本管理

操作步骤如下：

(1) 进入[微服务治理平台/微服务/配置中心]页面，点击配置名称进入其详情页面。

(2) 单击历史版本页签：

a. 展示此文件的历史版本列表（版本号、配置分组、最后更新时间）。

b. 点击“配置名称”，查看历史版本配置内容。

c. 点击“回滚”，弹框显示版本配置内容，再次确定后，回滚成功。

导入/导出配置

操作步骤如下：

(1) 进入[微服务治理平台/微服务/配置中心]页面。

(2) 点击导入配置，弹框中。

a. 可导出配置样例查看导入的结构和格式。

b. 上传 zip 文件，仅支持 zip 文件，导入前校验格式和文件类型，不超过 100MB。

c. 选择遇到相同配置（配置分组和配置名称完全一致）时的处理方式：终止导入、跳过、覆盖。

d. 终止导入后，弹框提示失败的条目和未处理的条目。

(3) 勾选已经导入的配置，点击导出配置，即浏览器下载 zip 文件。

删除配置

操作步骤如下：

(1) 进入[微服务治理平台/微服务/配置中心]页面。

(2) 找到需要删除的配置文件，在列表右侧操作列中，点击删除按钮，二次弹框提醒，确定后删除成功。

说明

- 配置分组的常见场景：不同的应用或组件使用了相同的配置类型，如 database_url 配置和 MQ_topic 配置。
- 同一分组下的配置文件名称不能重复。

6.3.3 认证中心

认证中心基于 Cloud Foundry 的 UAA 认证开发的，是一个多租户身份管理服务，用于管理用户和 OAuth2 客户端。不同的认证域相当于部署不同的 UAA，但是使用更少的资源。主要用于服务之间的调用可通过第三方授权，服务将受到授权中心保护，需提供授权方可调用。

6.3.3.1 名词解释

- 认证域（Identity Zones）：旨在支持多租户架构，每个租户被称为认证域。身份区域使用 REST API 配置了 JSON 有效内容。请参阅 UAA API 文档（https://docs.cloudfoundry.org/api/uaa/version/4.19.0/index.html#overview）中的示例。身份区域是 UAA 管理的实体周围的逻辑边界。区域中的实体包括但不限于：客户注册、用户、组、组映射、用户批准、身份提供者和访问、ID 和刷新标记。
- 默认区域：UAA 部署始终将一个区域称为默认区域。默认区域使用 YAML 配置文件进行配置和引导。
- 用户：用户是 UAA 服务器的中央结构域对象。由于 UAA 既是一个账户存储又是一个授权服务器，许多不同类型的信息都与用户链接，并且可以通过以用户为中心的 API 调用访问。UAA 用户履行 OAuth2 协议的资源所有者角色。发给用户的访问令牌在请求客户端允许的范围和用户的组成员身份的交集处包含范围。
- 组：组限制实体具有的授权范围以及此客户端或用户可以授权的内容。
- OAuth 应用：UAA 是一个 OAuth2 授权服务器。在应用程序可以获得访问令牌之前，开发人员必须执行一次性注册过程以在 UAA 中创建 OAuth 应用。OAuth 应用通常代表具有自己的一组权限和配置的应用程序。OAuth 应用受到简单凭证（如客户端 ID 和密码）的保护，应用程序用它们向 UAA 验证自己以获取令牌。有两种类型的 OAuth 应用：
  - 访问资源并从 UAA 请求令牌的 OAuth 应用。
  - 代表资源并接受和验证访问令牌的 OAuth 应用。

6.3.3.2 认证域管理

存在一个默认的认证域 UAA，可在该认证域中进行管理用户、组、OAuth 应用等。

设置认证域登录地址

(1) 部署系统服务 UAA 时可同时配置 UAA 域名，若未配置，部署后，可到指定位置进行配置。

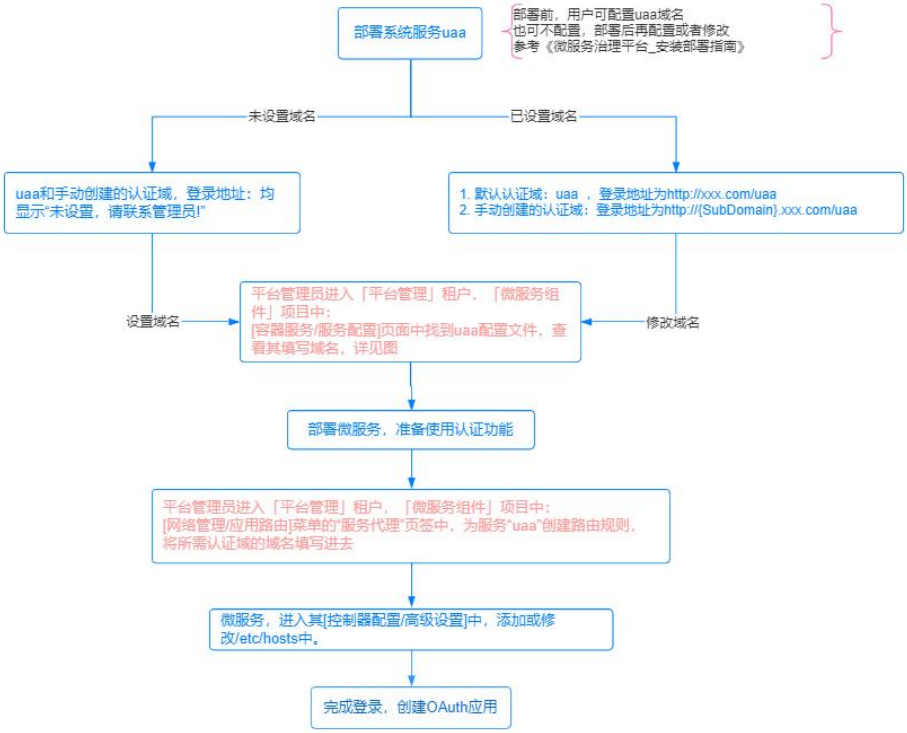
(2) 部署微服务，准备使用认证功能。

(3) 为 UAA 服务创建集群对外访问地址：在[容器服务/网络管理/应用路由]中为 UAA 创建路由规则。

(4) 进入服务的容器服务详情页面，进入[控制器配置/高级设置]中，添加或修改/etc/hosts。

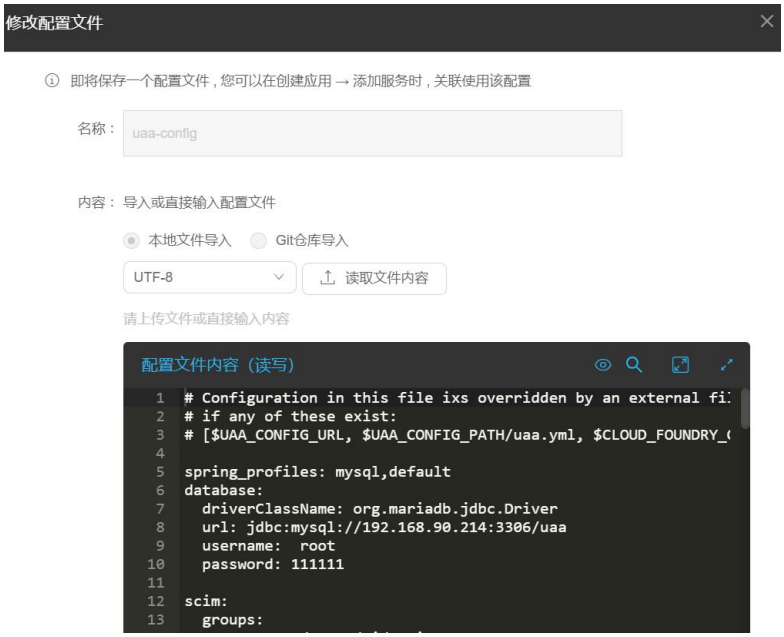
(5) 进入微服务治理平台-认证中心-认证域管理，选择 UAA 认证域，进入其详情页面，点击登录地址，输入用户名、密码，完成登录，创建 OAuth 应用。

图 6-36 认证域登录



备注：配置文件中的所有存在的域名是“服务代理中可配置的域名”的一个子集，UAA 配置文件中填写域名地址位置如下。入口：平台管理租户，微服务组件项目中，进入[容器服务/服务配置]可看到 UAA 配置文件。需平台、系统管理员才有此权限。

图 6-37 配置文件示例页面



添加认证域

操作步骤如下：

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 单击页面上部的添加认证域按钮，进入到如下界面。

图 6-38 添加认证域弹框

(1) 填写认证域配置信息，点击确定按钮，认证域添加完成，新的认证域信息会展现在管理列表中，如下图所示。

图 6-39 认证域管理页面

(2) 点击认证域名称，进入其详情页面，查看访问地址，可登录个人身份认证中心，维护 OAuth 应用。

图 6-40 认证详情

编辑认证域

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 找到需要删除的认证域，在列表右侧操作列下拉后点击编辑按钮。

(3) 修改认证域配置信息，点击确定按钮，认证域修改完成。

删除认证域

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 找到需要编辑的认证域，在列表右侧操作列下拉后点击<删除>按钮。

(3) 在弹出框中点击确定按钮即可删除。

删除认证域后，相应的用户、组、OAuth 应用也将被删除，请谨慎操作。

6.3.3.3 用户管理

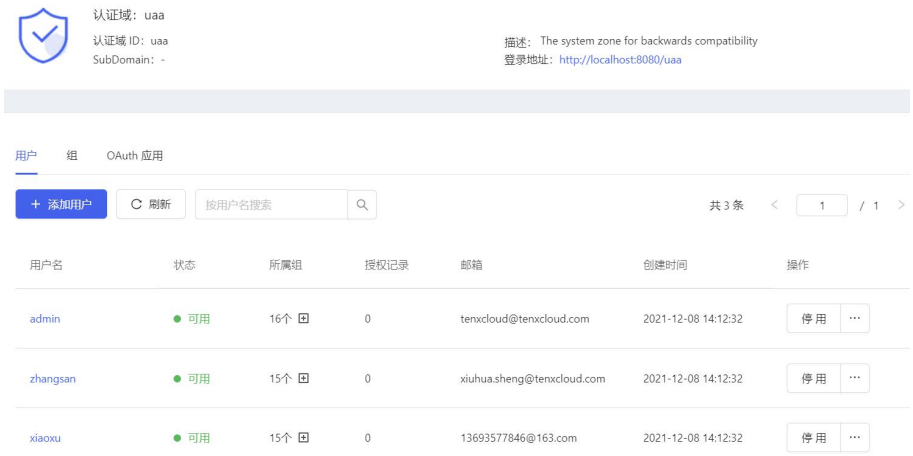
添加用户

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击<查看详情>按钮，点击用户进入到如下界面。

图 6-41 认证域管理详情页面



(3) 单击页面中的添加用户按钮，弹出弹框如下图所示。

图 6-42 认证域管理-添加用户弹框

(4) 输入基本信息，用户名、密码、再次输入密码、邮箱、手机后，点击<确定>按钮，添加用户到列表中。

编辑用户

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击用户。

(3) 在用户页面中找到需要编辑的用户，列表右侧操作列下拉后点击<编辑>按钮。

(4) 此处仅支持修改用户名、邮箱、手机，修改信息后，点击确定按钮。

修改密码

操作步骤如下：

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击用户。

(3) 在用户页面中找到需要修改密码的用户，列表右侧操作列下拉后点击修改密码按钮。

(4) 填写旧密码与新密码后，点击<确定>按钮。

删除用户

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击<查看详情>按钮，点击用户。

(3) 在用户页面中找到需要删除的用户，列表右侧操作列下拉后点击<删除>按钮。

(4) 在弹出框中点击<确定>按钮即可删除。

启用/停用用户

操作步骤如下：

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击用户。

(3) 在用户页面中找到需要启用／停用的用户，列表右侧操作列下拉后点击启用／停用按钮。

(4) 在弹出框中点击<确定>按钮即可成功改变状态。

查看用户详情

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

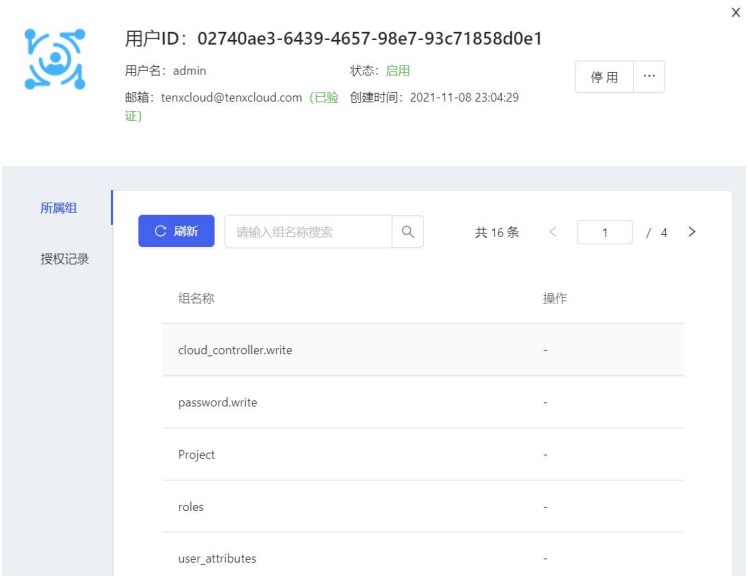
(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击用户。

(3) 在用户页面中找到需要查看详情的用户，点击用户名。

(4) 右侧划出用户详情页面，页面中支持查看用户的所属组和授权记录。

(5) 在所属组页面点击右侧操作列的移出该组按钮，支持将用户移出所属组，如下图所示。

图 6-43 认证域管理-移除用户面版



(6) 在授权页面可查看该用户的授权记录，展示每条授权的详细信息，包括授权客户端、授权范围、授权状态、授权时间、授权过期时间。

6.3.3.4 组管理

添加组

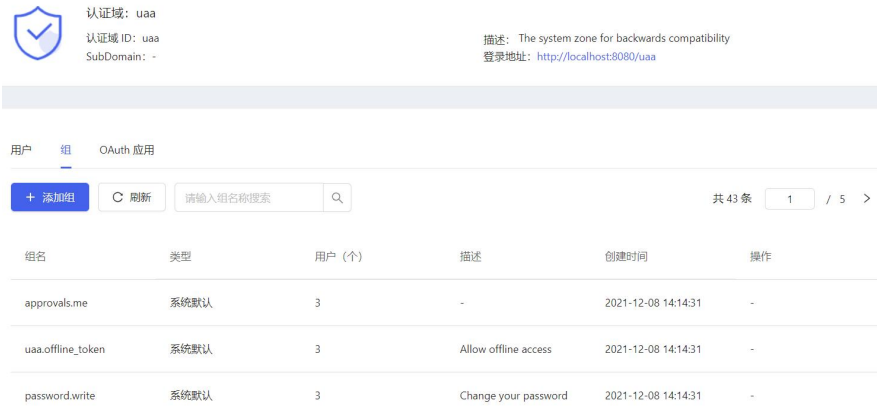
系统默认的组，不支持删除、编辑、管理组用户等操作；添加用户后，将自动添加到 14 个默认的组中。

操作步骤如下：

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击组进入到如下界面。

图 6-44 认证域管理-组页面



(3) 单击页面中的添加组按钮，输入组名称、描述，点击确定按钮，添加组到列表中。

编辑组

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或操作列"查看详情"按钮，进入认证域详情页面，点击“组”页签。

(3) 在组页签中找到需要编辑的组，操作列下拉后点击编辑按钮。

(4) 此处修改信息后，点击<确定>按钮。

管理组用户

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击组。

(3) 在组页面中找到需要管理用户的组，列表右侧操作列下拉后点击管理组用户按钮。

(4) 穿梭框左侧选择用户，点击<添加>按钮；或穿梭框右侧选择用户，点击<移除>按钮；最终确定后点击<确定>按钮。

删除组

操作步骤如下所示。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击“组”。

(3) 在组页面中找到需要删除的组，列表右侧操作列下拉后点击<删除>按钮。

(4) 在弹出框中点击<确定>按钮即可删除。

查看组详情

查看组详情的操作步骤如下。

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击组。

(3) 在组页面中找到需要查看详情的组，点击组名。

(4) 右侧划出组详情页面，页面中支持查看组的组用户。

(5) 在组用户页面点击右侧操作列的移除用户按钮，支持将用户移出该组。

6.3.3.5 OAuth 应用管理

认证域详情页处只对 OAuth 应用有查看的权限，具体的管理 OAuth 应用操作请在个人身

份认证中心中查看。

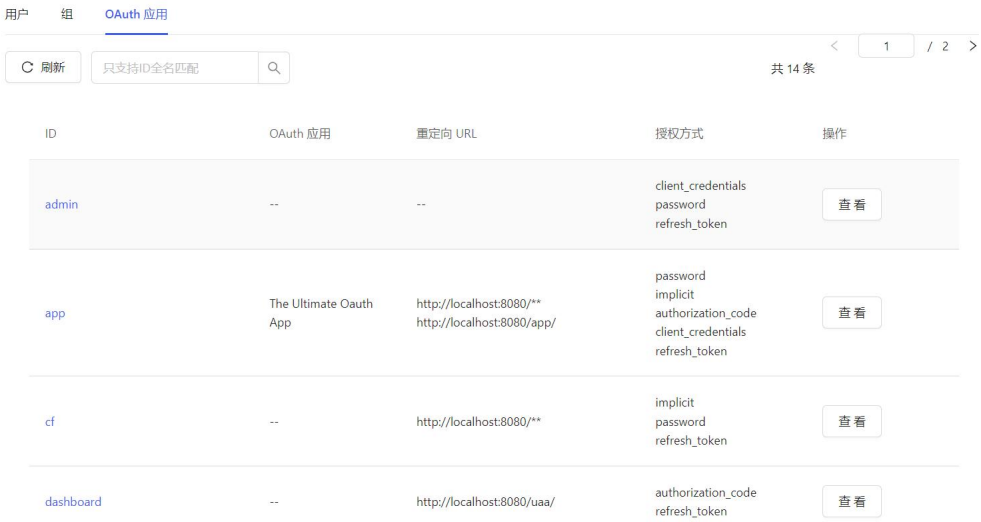
操作步骤如下：

(1) 进入[微服务治理平台/认证中心]进入认证域管理界面。

(2) 在页面中找到需要添加用户的认证域，点击名字或列表右侧操作列下拉后点击查看详情按钮，点击 OAuth 应用。

(3) 在 OAuth 应用页面中可查看该用户创建的所有 OAuth 应用，如下图所示。

图 6-45 认证域管理-OAuth 应用页面



(4) 支持查看 OAuth 应用的详情，在页面中找到需要查看详情的 OAuth 应用，点击列表右侧操作列中的查看按钮。

(5) 弹出框中可查看具体的 OAuth 应用设置，包括名称、授权方式、ClientID、授权范围等。

6.3.3.6 授权方式支持

表 6-2 授权方式说明列表

授权方式	说明
authorization_code	标准服务端授权模式
implicit	标准服务端授权简化模式
password	用户密码授权模式
client_credentials	应用端密钥授权模式

6.3.3.7 授权范围

表 6-3 授权范围说明列表

授权范围	说明
uaa.user	用户都拥有的权限
uaa.none	此 OAuth 应用将不会代表用户执行操作
uaa.admin	超级用户权限
scim.write	对整个跨域身份管理系统授予添加、更新、删除用户和组的权限
scim.read	对整个跨域身份管理系统授予读用户和组的权限
scim.create	仅授予添加用户的权限，无删、改、查用户的权限
scim.userids	用于用户名和用户 ID 之间转换
scim.invite	用于/invite_users 的 endpoint 参加邀请
groups.update	提供更新组的能力，scim.write 也包含该能力
password.write	提供修改用户密码的能力
openid	获取用户信息，适用于 OpenID clients
idps.read	读取、检索 identity providers 的能力
idps.write	添加、更新 identity providers 的能力
clients.admin	添加、修改和删除 OAuth 应用的能力
clients.write	添加、修改 OAuth 应用能力
clients.read	获取客户端注册信息的能力
clients.secret	修改自己 OAuth 应用密钥的能力
zones.read	用于获取认证域信息
zones.write	用于添加和修改认证域信息
scim.zones	限制仅能在某个认证域组内添加、删除用户
oauth.approval	默认授权，需要允许或拒绝 OAuth 应用代表用户行事
oauth.login	用于登录应用程序（例如外部登录服务器，可以执行可信操作，或添加的用户未进行身份验证）
approvals.me	尚未使用
uaa.resource	用于检查 token 的资源服务
zones.ZONE-ID.admin	允许在指定认证域内操作
zones.ZONE-ID.read	允许获取特定的认证域信息

授权范围	说明
zones.ZONE-ID.clients.admin	切换区域后与 clients.admin 一致
zones.ZONE-ID.clients.read	切换区域后与 clients.read 一致
zones.ZONE-ID.clients.write	切换区域后与 clients.write 一致
zones.ZONE-ID.clients.scim.read	切换区域后与 scim.read 一致
zones.ZONE-ID.clients.scim.create	切换区域后与 scim.create 一致
zones.ZONE-ID.clients.scim.write	切换区域后与 scim.write 一致
zones.ZONE-ID.idps.read	切换区域后与 idps.read 一致

6.3.4 个人身份认证中心

前置条件：需要在某个认证域中添加用户，设置用户名、密码等。然后使用用户名、密码登录身份认证中心，访问地址请到认证域详情页面进行查询，参见 6.4.3.2 添加认证域。

功能说明如下所示。

- 作为 OAuth 应用的管理中心。
- 作为个人 UAA 用户的账号管理中心。
- 作为 OAuth 授权管理中心。

6.3.4.1 登录个人身份认证中心

操作步骤如下：

(1) 访问单独部署的身份认证中心平台，输入微服务平台中创建好的 UAA 用户名、密码。

(2) 点击<登录>按钮，如下图所示。

图 6-46 个人身份认证中心登录页面

忘记密码

操作步骤如下所示。

(1) 访问单独部署的身份认证中心平台，若忘记密码。

(2) 点击左下方的忘记密码按钮，输入邮箱后，点击发送重置密码链接，如下图所示。

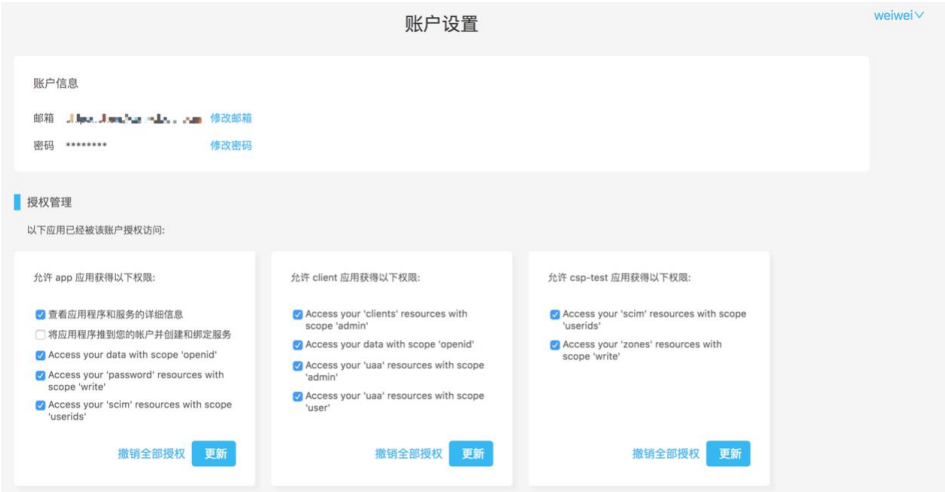
图 6-47 个人身份认证中心-密码重置页面

(3) 可在邮箱中查收到重置密码的邮件，点击验证链接，根据步骤，即可重置密码。

账户设置

正常登录身份认证中心平台后，点击下拉右上角的账户名称，单击账户设置按钮，可进入账户设置界面，如下图所示。

图 6-48 个人身份认证中心-账户设置页面



授权管理

- 更新：支持更新某个 OAuth 应用的授权。通过勾选修改权限后，点击<更新>按钮，即可更新成功。
- 撤销全部授权：支持撤销某个 OAuth 应用的全部授权，撤销后将停止向应用授权，如有需要需重新授权，点击<撤销全部授权>按钮。

修改邮箱、密码

分别点击修改邮箱和修改密码按钮，弹出下图所示的页面。

修改密码操作可输入当前密码、新密码、再次输入新密码后，点击修改密码按钮。

修改邮箱操作可输入新邮箱，发送链接至新邮箱验证后，可修改成功。

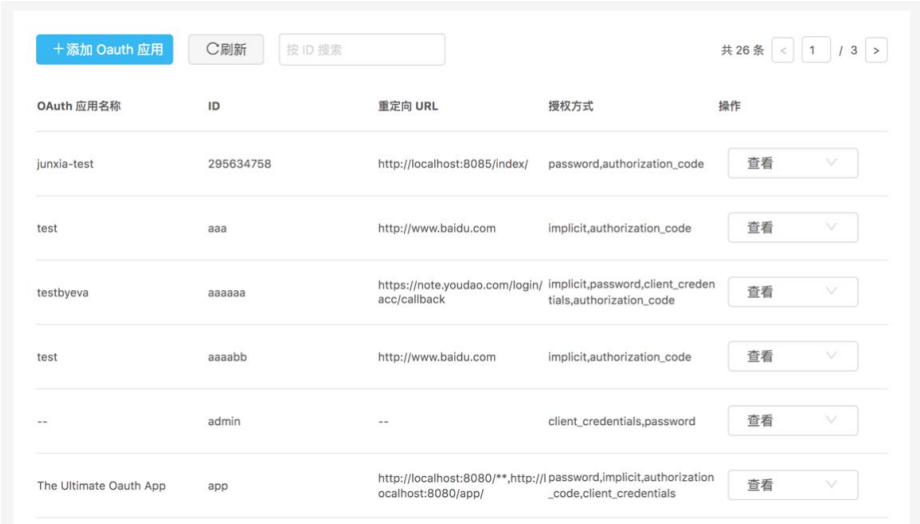
图 6-49 个人身份认证中心-修改密码页面

图 6-50 个人身份认证中心-修改邮箱页面

6.3.4.2 OAuth 应用管理

正常登录身份认证中心平台后，点击下拉右上角的账户名称，单击 OAuth 应用管理按钮，可进入界面，如下图所示。

图 6-51 个人身份认证中心-OAuth 应用列表页面



添加 OAuth 应用

操作步骤如下所示。

(1) 登录单独部署的身份认证中心平台，点击下拉右上角的账户名称，单击 OAuth 应用管理按钮。

(2) 单击添加 OAuth 应用，在弹出框中编辑信息。

图 6-52 添加 OAuth 应用弹框

- 授权方式：支持四种授权方式，可多选，包含：authorization_code、implicit、password、client_credentials。
- Client ID：client 的 credentials，支持自定义 Client ID。
- Client Secret：client 的 credentials。支持自定义 Client Secret。
- 授权范围：支持添加多种授权范围，每个授权范围可选择是否自动审批，若勾选自动审批则自动允许的客户端，用户无需显示的被询问是否授权。

(3) 点击<确定>按钮，完成添加 OAuth 应用。

说明

client_credentials：直接由 Client 向 Authorization Server 即 uaa，请求 access_token（提供 client 的 credentials，即：client_id, cilent_secret），无需用户（Resource Owner）的授权。

查看 OAuth 应用

操作步骤如下所示。

(1) 登录单独部署的身份认证中心平台，点击下拉右上角的账户名称，单击 OAuth 应用管理按钮。

(2) 在列表中找到需要查看的 OAuth 应用，右侧操作列中下拉点击查看按钮。

编辑 OAuth 应用

操作步骤如下所示。

(1) 登录单独部署的身份认证中心平台，点击下拉右上角的账户名称，单击 OAuth 应用管理按钮。

(2) 在列表中找到需要编辑的 OAuth 应用，右侧操作列中下拉点击<编辑>按钮。

(3) 在弹出框中可修改名称、授权方式、重定向 URI、授权范围，如需修改 Client Secret，请参考下方的修改 Client Secret。

(4) 修改信息后，单击<确定>按钮即可。

删除 OAuth 应用

操作步骤如下所示。

(1) 登录单独部署的身份认证中心平台，点击下拉右上角的账户名称，单击 OAuth 应用管理按钮。

(2) 在列表中找到需要删除的 OAuth 应用，右侧操作列中下拉点击<删除>按钮。

(3) 在弹出框中点击<确定>按钮，方可删除成功。

修改 Client Secret

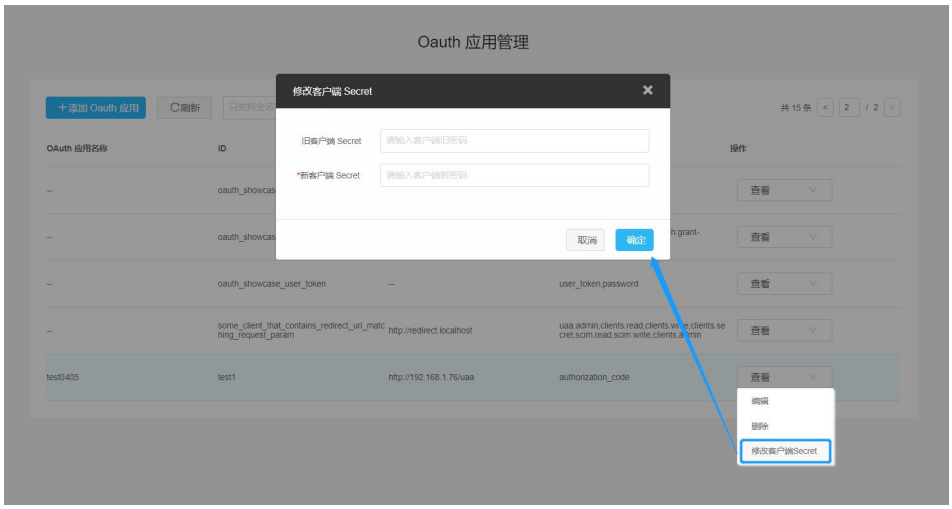
操作步骤如下所示。

(1) 登录单独部署的身份认证中心平台，点击下拉右上角的账户名称，单击 OAuth 应用管理按钮。

(2) 在列表中找到需要修改 Client Secret 的 OAuth 应用，右侧操作列中下拉点击修改 Client Secret 按钮。

(3) 在弹出框中输入旧的 Client Secret，和新的 Client Secret，点击<确定>按钮即可。

图 6-53 修改客户端 Secret 弹框



6.4 微服务网关

微服务网关支持智能路由、限流、负载均衡等功能。

6.4.1 设置网关服务代理

共享微服务框架：系统租户下选择共享微服务治理框架所在项目,在[容器服务/网络管理/应用路由]中为 spring-cloud-gateway 网关服务设置服务代理。

独享微服务框架：在目标租户下，选择独享微服务治理框架所在项目,在[容器服务/网络管理/应用路由]中为 spring-cloud-gateway 网关服务设置服务代理。

6.4.2 网关路由管理

路由将请求通过网关映射到具体的服务端，无需考虑服务地址变更问题，根据请求的来源、目标服务、敏感 Header 将服务访问请求分发到不同版本的微服务实例中。

微服务网关支持权重路由、设置路由转发匹配条件、超时设置、失败重试次数设置等。

若没有服务网关，多个服务提供给前端调用地址管理错综复杂，增加了客户端的复杂性，认证也相对麻烦，每个服务都需要编写相同的认证。

6.4.2.1 添加路由规则

操作步骤如下：

(1) 选择[微服务治理平台/微服务/服务管控]菜单项，进入微服务详情页面，点击路由规则页签。

(2) 单击<添加路由>按钮，进入如下图中操作界面。

图 6-54 微服务添加路由页面

参数介绍：

- 路由名称：可由3~50位字母、数字、中划线组成，以字母开头，字母或者数字结尾。
- 路由路径：主要补充该访问路径中的http://网关地址：端口/路由路径，路由路径部分。
  a. 精确匹配（/demo）：路径必须精确匹配/demo。
  b. 单级目录（/demo/*）：路由路径可匹配单级目录。
  c. 多级目录（/demo/**）：路由路径可匹配多级目录。
- 目标服务：可添加一个或多个服务；多个服务时，设置各个服务的权重百分比。

当某路由规则中有多个服务时，在[运维中心/治理策略管理]中统计时，此路由规则属于主服务。

- 匹配规则：设置路由过滤条件，可以不设置、也可设置多条。
  - Cookie：基于 key-value，value 支持完全匹配、正则匹配的方式匹配路由。
  - Header：基于 key-value，value 支持完全匹配、正则匹配的方式匹配路由。
  - Query：基于 key-value，value 支持完全匹配、正则匹配的方式匹配路由。
  - IP：基于 IP 地址匹配路由，支持用逗号分隔，例如 192.168.0.1,192.168.0.2。
  - Host：基于 host 域名进行路由匹配，支持固定匹配，通配符匹配。
  - Method：支持路由获取方式进行匹配，支持方式有 GET、HEAD、POST、PUT、PATCH、DELETE、OPTIONS 等方式，支持多选。
  - datetime：支持路由请求时间段的选择，支持 before、between、after，需单选，时间支持年月日时分秒选择。
- 去掉路径前缀：开启后表示转发前去掉路径中前缀。
- 失败重试机制：开启后表示开启失败重试机制，支持设置重试次数。失败包括超时、请求5xx。。
- 超时设置：设置路由请求时长，时长取值是1~600000ms。超时后也会触发重试。
- 敏感Header：若设置敏感Header，将不向下游服务传递设置的敏感Header；若未设置任何敏感Header，则向下游服务传递所有Header。
- 全局Header：选择全局敏感Header，表示使用全局统一配置；自定义敏感Header:选择自定义敏感Header，表示微服务添加特定的敏感Header。

(3) 填写好路由配置信息后，点击<确认>按钮，路由添加成功，新添加的路由规则后，点击确定按钮即可添加完成。

图 6-55 查看页面

6.4.2.2 设置全局敏感 Header

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，进入服务列表页面。

(2) 单击<全局路由设置>按钮，进入如下图所示的操作界面。

图 6-56 微服务设置全局敏感 header 页面

(3) 通过增加输入名称或减少 Header 后，单击<确定>按钮即可设置成功。

6.4.2.3 启用规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，进入微服务详情页面，点击路由规则页签。

(2) 单击路由列表页找到要启用的规则，在列表右侧操作列中，点击<启用>操作。可在弹出框中单击<确定>按钮即可。

6.4.2.4 停用规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，进入微服务详情页面，点击路由规则页签。

(2) 单击路由列表页找到需要编辑的规则，在列表右侧操作列中，点击“停用”操作。

(3) 单击确定按钮，停用成功。

6.4.2.5 编辑路由规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，进入微服务详情页面，点击路由规则页签。

(2) 单击路由列表页找到需要编辑的规则，在列表右侧操作列中，点击<修改>按钮。

(3) 编辑信息后，单击<确定>按钮，编辑成功。

6.4.2.6 删除路由规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，进入微服务详情页面，点击路由规则页签。

(2) 单击路由列表页找到需要删除的规则，在列表右侧操作列中，点击<删除>按钮。

(3) 可在弹出框中单击<确定>按钮即可。

6.4.3 网关负载均衡

新增负载均衡功能，用户可在在微服务详情页进行配置。

(1) 进入[微服务治理平台/微服务/服务管控]页面

(2) 点击服务名称链接，进入服务详情页面，配置负载均衡策略，如下图所示。

图 6-57 负载均衡算法页面

说明

- Dubbo 类服务不支持网关负载均衡的配置，其服务详情页面也不显示该配置项。
- 服务配置负载均衡后，对应服务通过网关直接访问时才生效，服务间访问不生效。

6.4.4 网关限流

一个服务实例限制别的服务访问自身，防止过载崩溃的机制，防止瞬时流量过大造成系统崩溃或服务不可用。

当限流对象对当前服务实例的每秒请求数量超过设定的值时，当前服务实例将不在接受该

对象的请求。

微服务网关是每个请求的必经入口，非常适合做一些 API 限流，防止网络攻击，限制客户端的请求速率能在一定程度上抵御攻击，降低损失。

限流类型主要包括 url、origin、user 三种。

```
if (types.contains(URL)) {
      joiner.add(route.getPath());
  }
  if (types.contains(ORIGIN)) {
      joiner.add(getRemoteAddr(request));
  }
  if (types.contains(USER)) {
      joiner.add(request.getUserPrincipal() != null ? request.getUserPrincipal().getNam
e() : ANONYMOUS);
  }
```

- url 类型的限流就是通过请求路径区分。
- origin 是通过客户端 IP 地址区分。
- user 是通过登录用户名进行区分，也包括匿名用户。
- 也可以多个限流类型结合使用。
- 如果不配置限流类型，就不做以上区分。

6.4.4.1 添加限流规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，选择目标服务进入其详情页面。

(2) 单击“限流规则”-“网关限流”页签，点击添加限流规则按钮，进入如下界面。

图 6-58 微服务添加限流规则页面

参数介绍：

- 限流类型：支持 url、origin、user 三种限流。
- 限流阈值：窗口时间内请求高于阈值次数时将触发限流，表示当前服务将不再接受请求。
- 窗口：该时间内的请求高于限流阈值，将触发限流。

(3) 填写限流规则信息后，点击<确定>按钮，规则添加成功，新添加的规则会展现在规则列表中。

6.4.4.2 编辑限流规则

操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]页面，选择目标服务进入其详情页面。

(2) 单击“限流规则/网关限流”页签，找到需要编辑的规则，在列表右侧操作列中，点击编辑按钮。

(3) 可编辑限流阈值、时间窗口等信息后，单击确定按钮，编辑成功。

6.4.4.3 停用 / 启用限流规则

操作步骤如下所示。

(1) 进入[微服务治理平台/微服务/服务管控]页面，选择目标服务进入其详情页面

(2) 单击“限流规则/网关限流”页签，找到需要启用/停用的规则，在列表右侧操作列中，点击启用 / 停用按钮。

(3) 可在弹出框中单击<确定>按钮即可。

6.4.4.4 删除限流规则

操作步骤如下：

(1) 进入[微服务治理平台/微服务/服务管控]页面，选择目标服务进入其详情页面。

(2) 单击“限流规则/网关限流”页签，找到需要删除的规则，在列表右侧操作列中，点击<删除>按钮。

(3) 可在弹出框中单击<确定>按钮即可。

说明

若选择的服务已存在启用状态的限流规则，开启后将停用该服务的其他限流规则。

6.5 链路追踪

前提条件：

- 租户管理员已为授权集群安装或申请微服务治理框架。
- 项目管理员已为授权集群开启微服务治理[微服务运维/运维开关]。
- 服务已注册到微服务注册中心，并已接入链路，详见链路配置。

链路追踪提供了完整的调用链路拓扑、trace 查询、方法追踪等功能，可以帮助开发者快速发现和定位分布式应用架构下的性能瓶颈，提高微服务时代下的开发诊断效率。主要功能如下：

- 调用链查询和诊断：追踪分布式架构中的所有微服务用户请求，并将它们汇总成分布式调用链。
- 全链路拓扑动态发现：用户的所有微服务应用、中间件可以通过链路追踪收集到的分布式调用信息。
- 多语言开发程序接入。
- 方法级别。

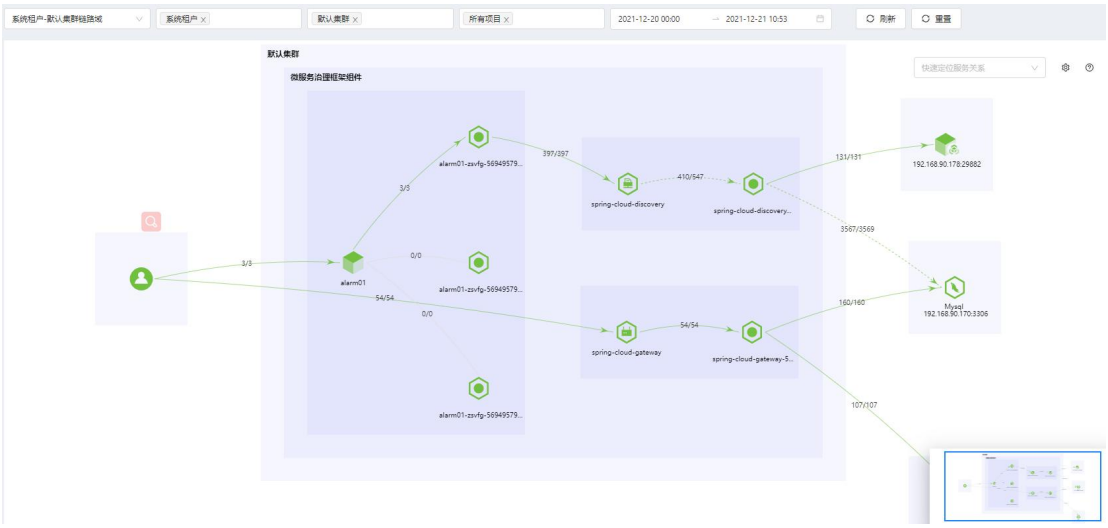
6.5.1 全链路拓扑

全链路拓扑选择链路域后，支持跨集群、租户、项目查看服务之间的调用关系，点击服务展开多维度面板，可快速获取资源、JVM、中间件、流量、详情等方面的监控数据，同时还支持实例级别调用数据展示，有助于及时发现问题、排查问题，以及在资源利用方面给用户决策提供数据支撑。

进入[微服务治理平台/链路追踪/全链路拓扑]页面。为方便运维人员协助定位问题。

服务增加详情面板、流量面板、资源面板、JVM 面板、中间件面板、链路（跳转新的标签页）、展开（实例级数据展开/收起）展示。

图 6-59 微服务拓扑页面



- 查询条件：可通过链路域、租户、集群、项目、时间多维度筛选，查看服务调用拓扑关系。
- 时间支持自定义日期，默认展示近 1 天时间内，当前链路域&租户&集群（列表中第一个）下所有项目中微服务调用拓扑图。
- 拓扑图支持按集群、项目划分服务所属关系，同时自持跨租户、集群、项目调用关系展示。
- 拓扑图支持实例级别调用关系展示。
- 对于不属于当前所选集群的服务独立展示，属于第三方服务。
- 对于从浏览器或控制台命令发出的服务调用，统一归属为客户端调用。
- 所有中间件服务独立展示
- 拓扑支持按服务名称搜索，高亮显示某服务所有拓扑关系，从繁杂的拓扑图中快速定位服务关系
- 拓扑图帮助面板：
  - 服务/实例健康状态：按 ApdexScore，显示不同的颜色。

图 6-60 参数说明

ApdexScore	服务健康状态颜色	用户满意度
[0.75,1]	绿色	表示应用被调用时响应很快，用户体验较满意。
[0.3,0.75)	黄色	表示应用被调用时响应较慢，用户体验一般。
[0，0.3)	红色	表示应用被调用时响应极慢，用户体验较差。

  - 服务下方的依次表示：服务名称、项目名称、租户名称。

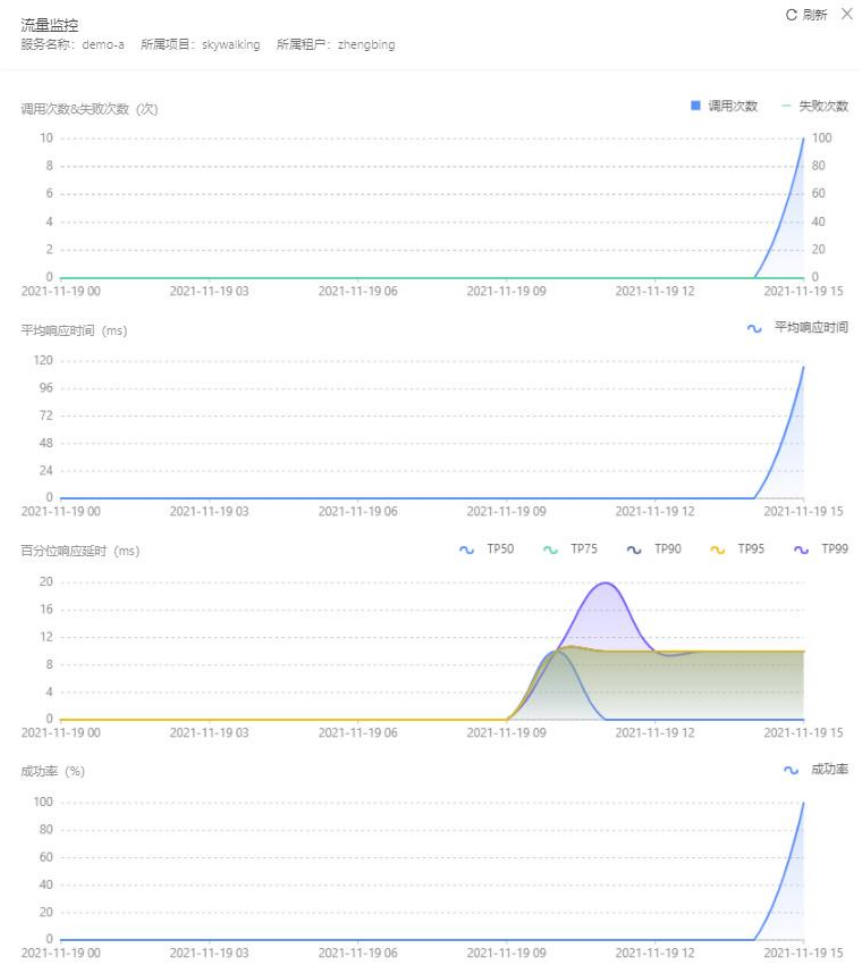
- 线上的数值表示：成功调用量/总调用量。
- 绿色实线代表发出的调用全部成功。
- 绿色虚线代表发出的调用部分成功/失败。
- 红色实线代表发出的调用全部失败。

- 点击服务弹出面板显示监控信息。
  - 详情：展示服务详情，包括服务名称（服务分组、所属项目、所属租户）、当前实例数、调用次数、失败次数、平均相应时间、ApdexScore、Service ApdexScore。
  - JVM 监控：展示对应容器组(支持历史容器组筛选）统计的 JVM 信息，默认选中第一个容器组，仅支持单选，统计展示页面如下图所示。

图 6-61 JVM 统计页面

  - 流量监控：统计并展示对应服务的调用次数&失败次数，平均相应时间，百分位响应时延，调用成功率，统计页面如下图所示。

图 6-62 流量监控页面



- 中间件：中间件面板显示支持分页。

Redis 展示 IP 地址、端口，版本信息，分别展示调用次数&错误次数、平均响应时间 ms、成功率曲线图；Operation 统计，分别展示每个 Redis 操作的类型、平均响应时间 ms、调用次数/失败次数，并支持关键字查询；

MySQL 展示 IP 地址、端口，版本信息，分别展示调用次数/失败次数、平均响应时间 ms 等信息；SQL 统计，分别展示每条 SQL 的语句、平均响应时间 ms、调用次数、失败数，并支持关键字查询；

RabbitMQ 展示 IP 地址、端口，版本信息，分别展示名称(条数)、平均相应时间、调用次数&失败次数，并支持按名称查询；

RocketMQ：展示 IP 地址、端口，版本信息，分别展示 Topic 名称(条数)、平均相应时间、调用次数&失败次数；

Kafka：展示 IP 地址、端口，版本信息，分别展示 Topic 名称(条数)、平均相应时间、调用次数&失败次数。

- 链路：点击链路标签，新标签页跳转到“调用链查询”页面，并填充当前查询条件。

- 容器维度资源：选择容器组(支持历史容器组选择)、容器，下方节点级联选择；容器组支持多选，容器仅支持单选，统计项有 CPU、内存、硬盘、网络流量（入站/出站 KB/s）、网络数据包（接收/发送个）。
- 节点维度资源：主机负载（主机 load average 1 分钟平均值）、CPU、内存、硬盘、网络流量（入站/出站 KB/s）、网络数据包（接收/发送个）、容器数量（已分配，百分比）；CPU、内存、硬盘根据百分比展示红色（n≥75%)、黄色(75%>n≥50%)、绿色（50%>n）。

图 6-63 资源信息页面

- 支持拓扑设置，默认不勾选。默认连线展示“成功次数/总调用次数”，还支持选择“平均响应时间”，支持单选、多选。

图 6-64 拓扑设置

- 服务展开多实例显示详情说明如下所示。
  - 首选定位异常服务，当服务黄色或红色时，如果用户想进一步确定是哪个 pod 出现异常）。点击服务小花瓣，点击展开可显示各实例（即 pod）颜色状态，调用连线先到服务（service），服务再连向实例（多个相互调用的服务均展开）。
  - 实例显示实例名称（版本号），点击其图标，不展开小花瓣，
- 页面提供说明帮助，用户可通过帮助了解拓扑图各图标含义、连线颜色意义、服务健康说明等信息。

说明

- 中间件面板数据展示：Operation 统计/SQL 统计/推送&订阅/topic 统计中数据由于会涉及业务敏感信息，故仅展示有权限的项目&服务调用中间件的数据。
- 平台未安装 TDSF 时不展示“显示 mTLS 标记”，安装后显示“网格服务显示 mTLS 标记”。
- 容器服务指未注册到内置注册中心，未接入网格的在 PAAS 部署的容器服务，此服务接入链路追踪显示正常。
- 资源面板：如果单选容器组，容器下拉列表为“全部容器”（此时，列表中“容器”项，展示对应容器组的所有容器），列表中指标统计为容器组统计结果。点击展开，面板显示此容器组及所在节点的各指标曲线图。
- 资源面板：如果单选容器组，又单选容器（此时，列表中“容器”项，展示所选容器），列表中指标统计为此容器统计结果。点击展开，面板显示此容器及所在节点的各指标曲线图。
- 资源面板：如果多选容器组，此时容器下拉列表只展示“全部容器”。

6.5.2 调用链查询

随着业务的扩展服务增多，很难洞察数据如何在蛛网般复杂的服务结构中流转。因此引入服务调用链来跟踪服务调用关系，分析分布式系统的每一次调用，从而精确发现系统的瓶颈和隐患。此处可查询特定的项目集群下，调用链路的状态，定位某一次调用情况，帮助排查问题。

6.5.2.1 调用链原理

每一次服务调用追踪链路，相当于生成一个 TraceID，起始（根）服务生成的 TraceID，表示一个请求。每一个 Trace 由一组 Span 组成。

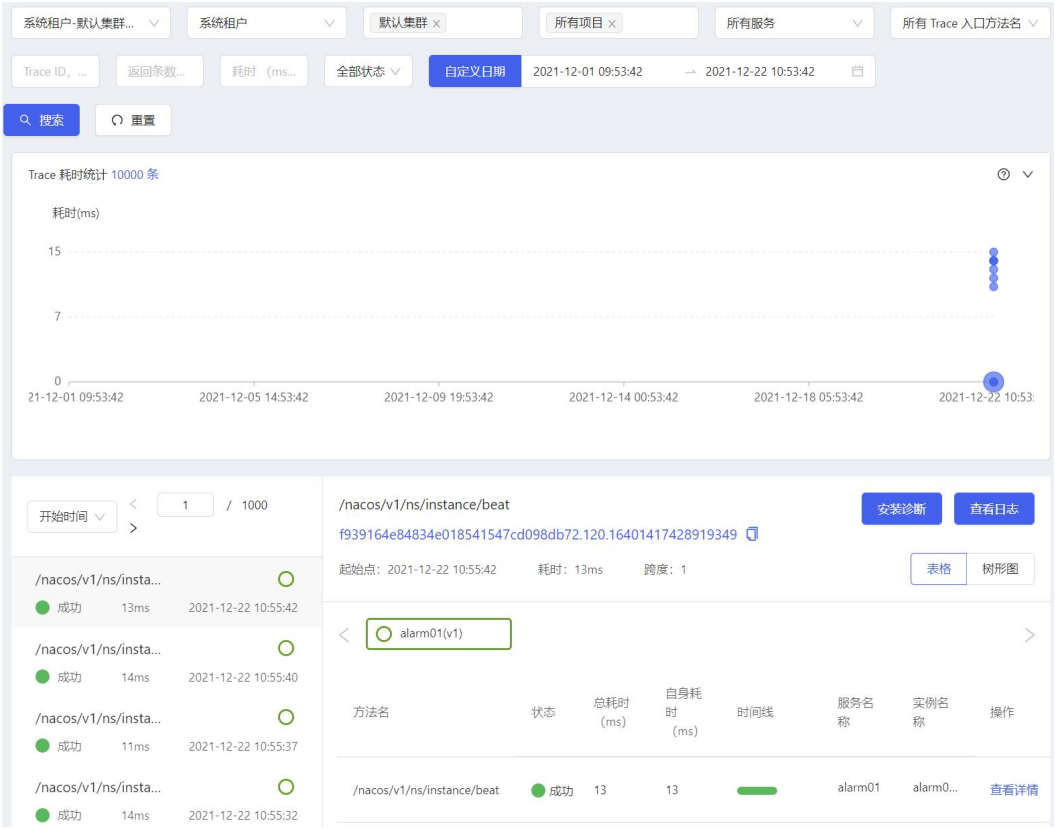
Span 表示一次完整 RPC 调用，是追踪服务调用的基本结构，多 span 形成树形结构组合成一次 Trace 追踪记录。Span 是有父子关系的，比如：Client A、Client A -> B、B ->C、C -> D、分别会产生 4 个 Span。Client A 接收到请求会时生成一个 Span A、Client A -> B 发请求时会再生成一个 Span A-B，并且 Span A 是 Span A-B 的父节点。

6.5.2.2 调用链查询与详情

操作步骤如下：

(1) 进入[微服务治理平台/链路追踪/调用链查询]即调用链查询界面。

图 6-65 微服务调用链查询页面



(2) 设置查询条件，单击<搜索>按钮。

- 选择链路域：仅支持单选，
- 选择集群：下拉框中选择租户，集群及对应的项目。
- 下拉选择服务分组、所有 Trace 入口方法名以及每条 trace 的耗时。
- 列表默认按开始时间倒序，每条 trace 展示入口方法名、开始时间、耗时、状态（成功 or 失败）、涉及服务。
- Trace ID：若使用 Trace ID 查询（完全匹配查询），则其他设置条件无效。
- 选择请求状态：所有状态、成功、失败。
- 返回条数：默认返回查询的所有结果，并做分页处理。
- trace 耗时气泡图：默认展开，支持收起，点击小气泡，可以选中下方列表中对应的 trace。
- 详情顶部展示“查看日志”入口，展示有权限服务的容器，详情顶部展示入口方法名、traceid（支持复制）、开始时间、耗时、span 数，下方表格展示各 span 信息，默认展开，支持收起、展开。
- Span 详情展示：方法名、状态、总耗时、自身耗时、时间线、微服务名称、实例名称、操作（查看详情）。
- Span 详情展示支持树形图展示，树结构展示 span 关系。

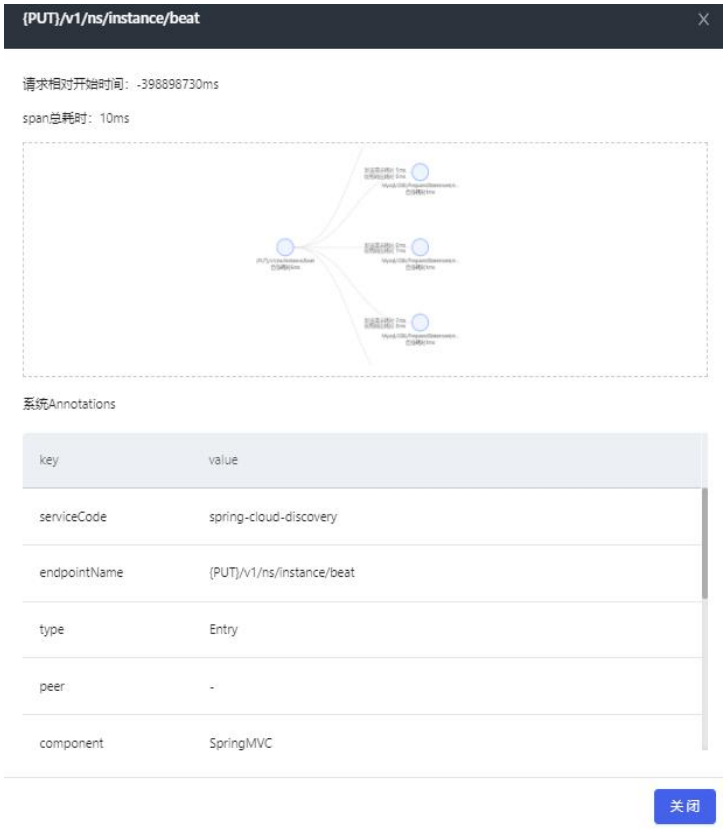
(3) 查询后数据如上图，可在上部图中根据圆点的颜色、大小、位置快速定位所需的调用。

- 大小表示该 Trace 包含的 Span 数量，越大表示 Span 越多。
- 颜色表示状态：蓝色表示成功，红色表示失败。
- 纵轴位置表示耗时时间，越往上表示耗时越大。
- 横轴位置表示 Trace 产生的时间。

(4) 点击列表右侧操作列的查看详情按钮，可查看该完整调用 Trace 详情，请求相对开始时间、span 总耗时、系统 annotation、自定义 annotationg 等信息。通过两个服务之间调用的耗时可大概定位问题产生的原因，如：若客户端发出请求耗时和客户端收到响应耗时较长，可考虑是否网络传输压力较大等；若服务端处理请求耗时较长，可考虑服务端本身是否出现问题。

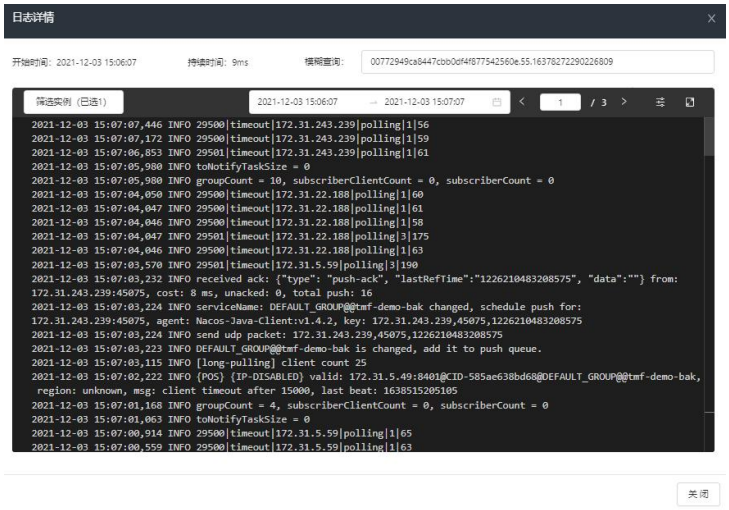
若时长出现负数，可能是节点时间不同步导致，同步节点时间请参考<TCE v5.2.0_时速云容器云 TCE 安装部署>及以上版本的相关章节。

图 6-66 微服务调用关系详情页面



(5) 点击服务调用链页面的“查看日志”页面，可根据 TracID 或者调用开始时间定位到容器日志相应位置，如下图所示。

图 6-67 微服务调用日志页面



6.5.3 链路配置

服务接入链路后方可支持链路追踪、链路拓扑等功能；同时为减少资源占用，您可以按需进行采集配置（采样率、Apdex 阈值等）。

6.5.3.1 接入链路

PAAS 部署服务后，服务均是未接入链路的状态，需要再此进行安装，安装成功后，默认开启采集，采样率 100%，Apdex 阈值 500ms。安装步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/链路配置]，进入链路配置列表页面。

(2) 单击<接入链路>按钮，进入接入链路页面，如下图所示。

图 6-68 接入链路页面

(3) 选择要接入链路的服务，点击<确定>按钮，对应服务接入链路成功。

说明

- 服务接入、移出后会自动重启。
- 手动注册的服务不支持接入链路。

6.5.3.2 移出链路

计入链路后支持移出，移出时，历史调用链相关数据不会被删除，移出步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/链路配置]，进入链路配置列表页面。

(2) 找到要移出链路的服务，单击<移出链路>按钮，如下图所示。

图 6-69 移出链路确认页面

(3) 确认卸载后，单击<确定>按钮，移出链路成功。

6.5.3.3 采集配置

单个服务采集配置

为减少资源占用，用户可以按需对服务“consumer-sample”进行采集配置，配置步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/链路配置]，进入链路配置列表页面。

(2) 找到要配置采集的服务，单击<采集配置>按钮，进入采集参数配置页面，如下图所示。

- Apdex 阈值：定义了服务响应时间的最优门槛为 T（即 Apdex 阈值，T 由性能评估人员根据预期性能要求确定）。

图 6-70 采集配置

(3) 配置完成后，单击<确定>按钮，采集配置成功。

服务统一采集设置

说明

独享框架租户管理员及以上权限、共享框架平台管理员及以上权限的用户可见统一采集配置按钮。

(1) 进入[微服务治理平台/链路追踪/链路配置]，进入链路配置列表页面。

(2) 单击<统一采集配置>按钮，进入采集参数配置页面，如下图所示。

- 采样率：设置链路采集信息的采样率，适当的采样率可以减少不必要的数据传输，降低对服务性能的影响，当采样率设置为 0%时，将停止采集。
- Apdex 阈值：定义了服务响应时间的最优门槛为 T（即 Apdex 阈值，T 由性能评估人员根据预期性能要求确定）。

图 6-71 统一采集配置

(3) 配置完成后，单击<确定>按钮，统一采集配置成功。

6.5.3.4 开启采集

(1) 进入[微服务治理平台/链路追踪/链路配置]，进入链路配置列表页面。

(2) 勾选要开启采集配置的服务，单击<开启采集>按钮，进入开启采集配置页面，如下图所示。

图 6-72 开启采集配置页面

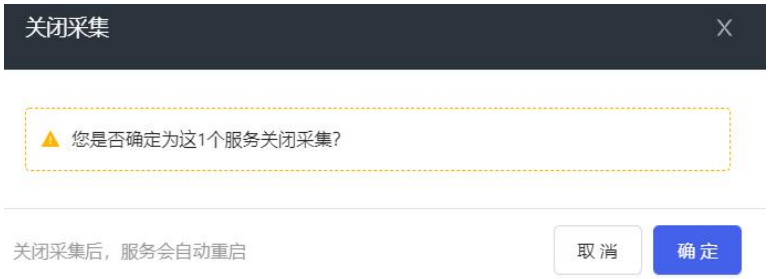
(3) 确认开启后，单击<确定>按钮，服务采集配置开启成功。

6.5.3.5 关闭采集

(1) 进入[微服务治理平台/链路追踪/链路配置]，进入链路配置列表页面。

(2) 找到要关闭采集的服务，单击<关闭采集>按钮，进入关闭采集配置页面，如下图所示。

图 6-73 关闭采集页面



(3) 确认关闭后，单击<确定>按钮，服务采集配置关闭成功。

6.5.4 方法追踪

方法追踪是对某个类的某个方法进行动态埋点，当这个类的方法被调用时，链路会按照用户配置的方法追踪规则对方法的调用数据进行采集，并将调用数据展现在调用链页面中。方法追踪主要用来帮助应用的开发人员在线定位方法级性能问题。

6.5.4.1 场景举例

- 运维或开发人员定位问题
  - 希望在调用链详情内看到链路追踪没有覆盖到的自身业务内部的方法，链路追踪

默认采集的内容粒度不够细，故增加方法级别的自定义埋点，将想采集的信息手动配置进去。

- 对于链路追踪默认没有支持采集的第三方框架，出现问题不好定位，通过自定义埋点可以扩展采集范围。对方法的埋点支持自定义标记参数信息，方便排错是检查方法参数值是否准确。

说明

- 服务在移出链路的时候，会将该服务下所有的埋点配置都删除，埋点列表中将不再展示这些埋点配置。
- 服务在关闭采集的时候，会将该服务下所有的埋点配置都停用，埋点列表中埋点状态将变为停用。
- PAAS 侧删除服务后，服务管控列表的服务直接将直接被删除，同时自定义埋点策略也直接删除。

6.5.4.2 添加埋点

- 前提条件

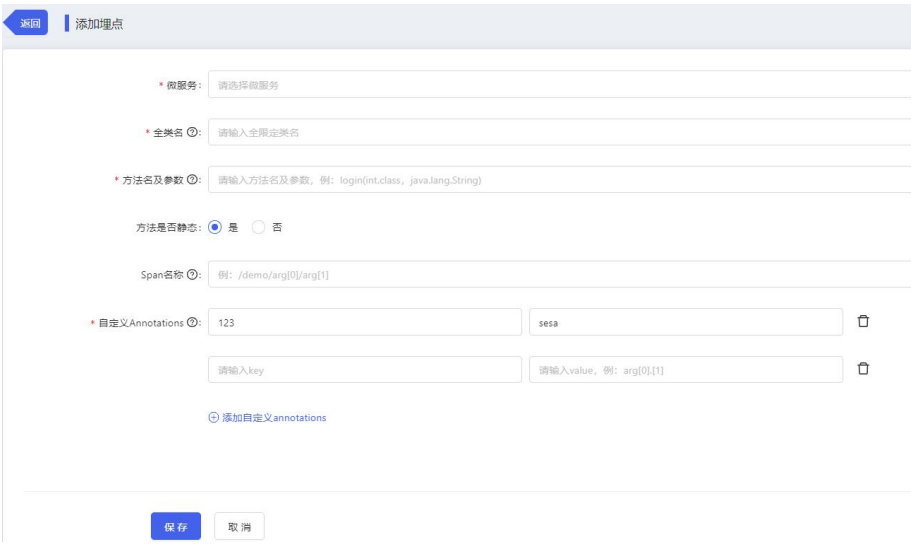
微服务框架安装成功(链路域设置正确)，且服务已接入链路。

添加埋点的操作步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/方法追踪]，进入埋点列表页面。

(2) 单击<添加埋点>按钮，进入埋点配置页面，如下图所示。

图 6-74 埋点配置页面



参数说明如下所示。

- 微服务：配置要埋点的服务。
- 全类名：服务对应的全类名，例如：包为 com.tenxcloud.user，类为 UserService，

则全类名为 com.tenxcloud.user.UserService。

- 方法名及参数：方法是代表功能的模块，传递给方法的参数，填写参数对应方法中参数的全类名,且必须和方法的参数顺序一致，并以英文逗号分隔，例如：方法为 public int login(int id, String name)，则方法名及参数为 login(int.class,java.lang.String)。
- Span 名称：Span 是分布式系统中的一个小的调用单元，可以是一个微服务中的 service，也可以是一次方法调用，甚至一个简单的代码块调用。Span 可以包含起始时间戳、log 等信息。每一个 Span 会有一个独有的 Span ID。
- 自定义 Annotation：程序的注解，用户可以配置注解为对象附加任意的非标识的元数据。服务程序能够获取元数据信息。

(3) 以上参数配置完成后，单击<确定>按钮，埋点设置配置成功。

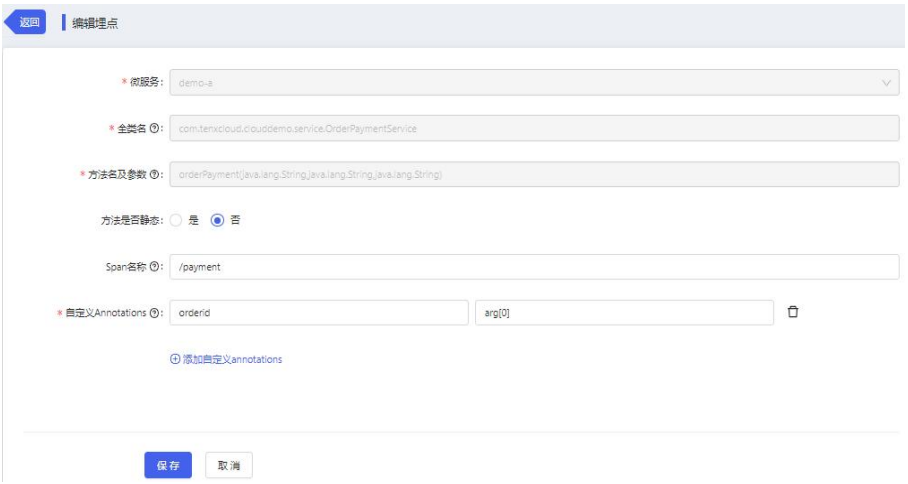
6.5.4.3 编辑埋点

编辑埋点的操作步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/方法追踪]，进入埋点列表页面。

(2) 找到要编辑的埋点，单击<编辑>按钮，进入埋点编辑页面，如下图所示。

图 6-75 编辑埋点页面



(3) 编辑埋点方法是否是静态、Span 名称、自定义 Annotations 等参数。

(4) 参数编辑完成后，单击<保存>按钮，编辑埋点操作完成。

6.5.4.4 查询埋点

查询埋点的操作步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/方法追踪]，进入埋点列表页面。

(2) 选择服务名/全类名/方法名及参数，单击<查询>按钮，查询埋点操作完成页面。

6.5.4.5 启用埋点

启用埋点的操作步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/方法追踪]，进入埋点列表页面。

(2) 找到要启用的埋点，单击<启用>按钮，进入启用操作页面。

(3) 根据页面提示执行操作后，单击<确定>按钮，启用埋点操作完成。

6.5.4.6 删除埋点

删除埋点的操作步骤如下所示。

(1) 进入[微服务治理平台/链路追踪/方法追踪]，进入埋点列表页面。

(2) 找到要删除的埋点，单击<删除>按钮，进入埋点删除确认页面，确认删除后，单击<确定>按钮，删除埋点操作完成。

6.6 微服务运维

6.6.1 策略管理

服务的治理策略包含网关路由、网关限流、服务限流（系统规则、流控规则）、熔断、手动降级、访问控制等，为了便于不同角色进行查询统计，快速了解平台或租户、项目中设置的策略情况，包含总数量、开启状态策略数量，主要支持以下功能：

- 多维度查询治理策略统计结果，根据登录用户角色权限选择租户、集群、项目，查询当前时刻网关路由、网关限流、服务限流（系统规则、流控规则）、熔断、手动降级、访问控制策略的数量。例网关路由显示 3/1，3 表示当前时刻此策略总数，1 表示开启状态的策略数量。
- 服务维度的各策略分别统计展示。

图 6-76 微服务策略实时统计

说明

- 微服务治理统计项部分：未安装或未申请框架时不进行提示，统计数据就是 0/0。下方微服务 tab 中列表显示暂无数据。
- 对于微服务策略-服务限流中的系统规则，其总数和启用数量保持一致。当全部启用时，显示 4/4；全部停用时，0/0；启用 2 个，显示 2/2。
- 微服务路由策略，当其中含有多个服务时，此策略仅统计为主服务的策略。

6.6.2 链路告警

为完善微服务的管理、治理能力，系统提供基于 skywalking 的告警。

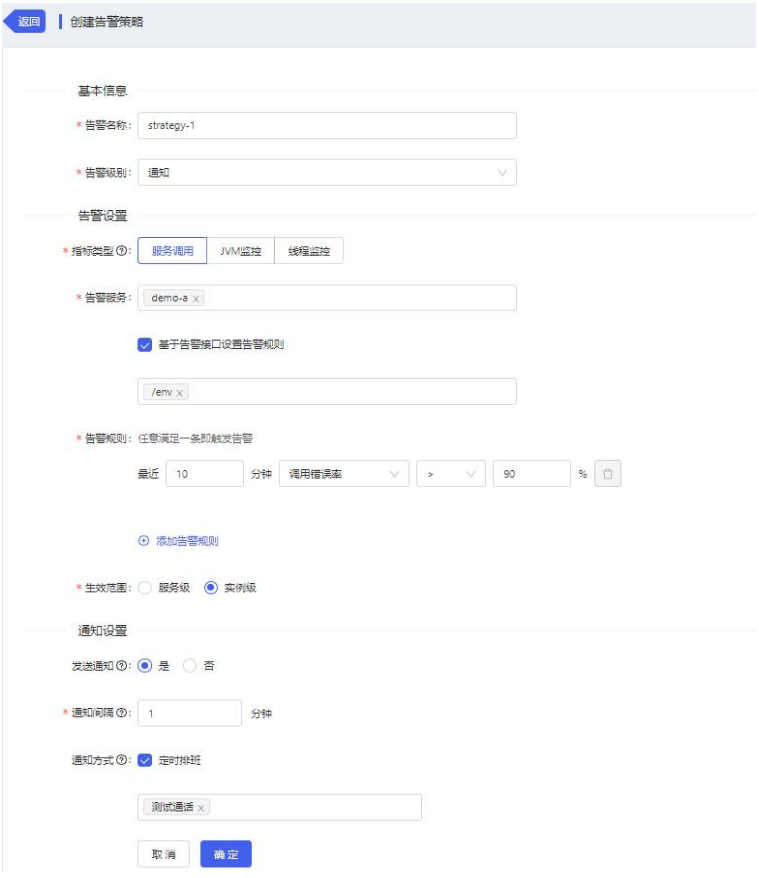
6.6.2.1 创建链路告警策略

告警规则其实就是配置的告警条件及检查周期，用户可根据业务需要进行配置，配置的操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，默认进入告警设置列表页面。

(2) 在告警设置页面单击<创建>按钮，进入链路告警策略配置页面，配置对应的策略参数，配置示例页面如下图所示。

图 6-77 链路告警配置示例页面



具体的参数说明如下图所示。

- 告警名称：输入 3-40 位字符，以英文字母或数字开头或结尾，支持下划线“_”、中划线“-”，修改时禁用不支持修改。
- 告警级别：配置当前告警策略的告警级别，支持设置的级别有：通知、告警、危险，用户可根据实际的业务需求进行配置。
- 指标类型：服务调用支持服务、实例、接口维度设置告警策略，JVM 监控、线程监控仅支持实例维度设置告警策略。
  - 线程监控：活跃线程总数 live（个）、死锁线程总数 deadlock（个）、峰值线程数 peak（个）。
  - JVM 监控：堆内存 used（MB）、非堆内存 used（MB）、OldGC 总次数（次）、YoungGC 总次数（次）、OldGC 总耗时（ms）、YoungGC 总耗时（ms）。
  - 服务调用：调用每分钟请求数（次）、调用总次数（次）、调用总错误次数（次）、调用错误率（%）、调用平均响应时间（ms）、调用 TP99（ms）、调用 TP95（ms）、调用 TP90（ms）、Apdex Score（最大值）。
- 告警服务：选择安装了 agent 的服务，如果指标类型为服务调用时支持配置“基于告警接口设置告警规则”，此处服务发生调用后才能读取到相关接口。
- 告警规则：支持最近“N”分钟，某一规则>（>=、<、<=、=）阈值（个/次/ms/MB）。
- 生效范围：配置链路告警策略的生效范围是服务级别还是示例级别。
- 通知配置：配置触发告警策略时是否发送通知，发送通知的时间间隔，选择是否定时排班，当选择定时排班时，选择相应的通知组。
- 不同集群的链路域推送告警信息到平台通知组需要配置集群地址，

方式一(不重启链路域)：在链路域所在项目[服务配置]选择 apm-dynamic-config 添加配置文件：名称 alarm.extended.paasUrl，内容为集群地址，例如：https://192.168.90.215。

方式二：如果已经确定地址不会变，可以直接在链路域服务端 tracing-analysis-server 容器配置/环境变量增加环境变量：SW_ALARM_PAAS_URL，变量值为集群地址

(3) 策略参数配置完成后，单击<确定>按钮，新建链路告警策略操作完成。

6.6.2.2 启用告警策略

启用告警策略操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，默认进入告警设置列表页面。

(2) 找到要启用的告警策略，单击<启用>按钮，确认启用后，启用告警策略操作完成。

6.6.2.3 停用告警策略

停用告警策略操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，默认进入告警设置列表页面。

(2) 找到要停用的告警策略，单击<停用>按钮，确认停用后，停用告警策略操作完成。

6.6.2.4 删除告警策略

删除告警策略操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，默认进入告警设置列表页面。

(2) 找到要删除的告警策略，单击<删除>按钮，确认删除后，删除告警策略操作完成。

6.6.2.5 忽略告警策略

删除告警策略操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，默认进入告警设置列表页面。

(2) 找到要忽略的告警策略，单击<忽略>按钮，确认后，忽略告警策略操作完成。

6.6.2.6 搜索告警策略

搜索告警策略操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，默认进入告警设置列表页面。

(2) 在搜索框内配置告警策略等参数，单击搜索图标，搜索告警策略操作完成。

6.6.2.7 告警记录

告警记录页面呈现了历史所有的告警策略产生的告警情况，这里可以通过筛选规则来定位查询历史的告警记录，来帮助运维开发同学来进行问题定位，应用运行的情况诊断。支持多维度监控、查询，支持清空告警记录。

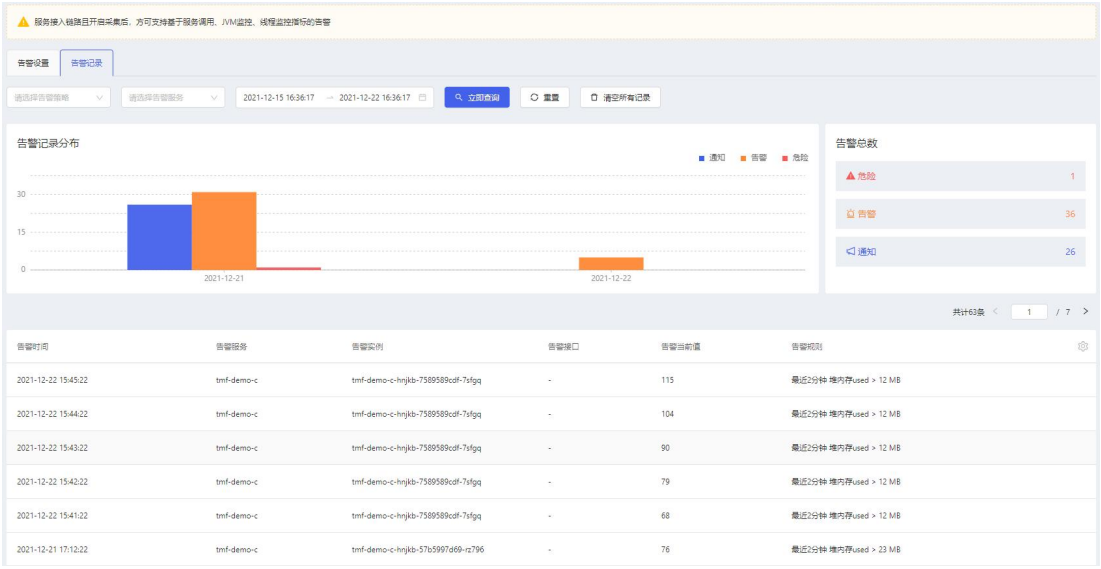
具体的操作步骤如下所示。

(1) 登录平台后，导航[安全和运维/微服务运维/链路告警]，切换至“告警记录”，进入日志告警页面。

(2) 选择或输入查询条件，用户可配置的告警记录属性有：告警策略、告警服务、告警时间段。

(3) 点击<立即查询>即可查询对应的告警记录。

图 6-78 告警记录查看页面



- 系统支持清除所有告警记录，该操作将会清空所有告警记录，并且已经清除的告警数据无法恢复。
- 告警记录分布：统计项目集群下的所有告警记录，告警记录类型有：通知、危险、告警。
- 告警总数：展示项目集群下的所有告警详情，用户可查看对应告警记录的告警时间、告警服务、告警实例、告警接口、告警当前值、告警规则等属性。
- 自定义列表项：系统支持自定义告警列表项，可选的配置列表项有：告警级别、策略名称、指标类型、是否发送通知。
- 重置：系统支持重置告警记录筛选条件。

6.6.3 应用诊断

Arthas 是诊断 Java 领域线上问题的利器，利用字节码增强技术，可以在不重启 JVM 进程的情况下，查看程序的运行情况。包括控制面板、线程列表、方法诊断和性能分析。功能特性如下所示。

- 减少运维人员学习技术成本，增加线上诊断易用性。
- 避免线上问题现场日志记录缺失，反复的添加日志再部署。
- 避免线上问题出现，线下无法复现，难以定位调试。

6.6.3.1 诊断管理

服务安装 Arthas 后方可支持应用诊断，当前列表只管理自动安装的 Arthas。支持两种安装方式，如下所示。

- 自动安装，服务列表只管理自动安装的 Arthas。

手动安装（不重启服务，需参考《微服务治理平台_代码接入指南》）。

安装 Arthas

前提条件：tmf-operator-server 插件已经安装，injector 应用正常运行。

安装 arthas 的操作步骤如下所示。

(1) 登录平台后，选择[安全和运维/微服务运维/应用诊断/诊断管理]菜单项，进入服务列表页面。列表展示所有自动安装 Arthas 的服务及容器组，容器组列表项：容器组名称，容器组状态，Arthas 状态，诊断容器，诊断进程，操作时间，操作（停用/启用）。

图 6-79 列表页面

(2) 单击<安装 Arthas>按钮，进入该项目下可安装 Arthas 的列表页面，选择要安装 Arthas 的容器服务，示例页面如下图所示。

图 6-80 安装页面

(3) 容器服务选择完成后，单击确定按钮，安装 Arthas 操作执行完成。

卸载 Arthas

(1) 登录平台后，选择[安全和运维/微服务运维/应用诊断/诊断管理]菜单项，进入服务列表页面。

(2) 找到要卸载 Arthas 的服务，单击<卸载>按钮，卸载 Arthas 操作执行完成。

停用 Arthas

(1) 登录平台后，选择[安全和运维/微服务运维/应用诊断/诊断管理]菜单项，进入服务列表页面。

(2) 找到要停用 Arthas 的服务，单击<停用>按钮，停用 Arthas 操作执行完成。

6.6.3.2 诊断分析

通过该功能模块，用户可获取目标服务的运行情况；用户可通过“诊断引导”查看诊断场景指引，用户配置诊断分析的操作步骤如下所示。

前提条件：目标需要服务安装 Arthas 后方可进行诊断分析，且目标服务运行环境要求 JDK1.8 及以上版本。

开启诊断分析

开启诊断分析的的方式有两种，具体的说明如下所示。

- 自动安装：进程级开启/关闭 arthas，选择[服务/容器组/容器/进程]，开启诊断。
- 手动安装：选择[服务/容器组]，开始诊断，手动安装的 arthas 只支持手动关闭及卸载，手动安装的 arthas 只有开启后才能被监听到，状态只有开启。

(1) 登录平台后，选择[安全和运维/微服务运维/应用诊断/诊断分析]菜单项，进入诊断分析页面。

(2) 选择要开启诊断的服务-容器组-容器-进程，单击<开始诊断>按钮，进入诊断结果查看页面。

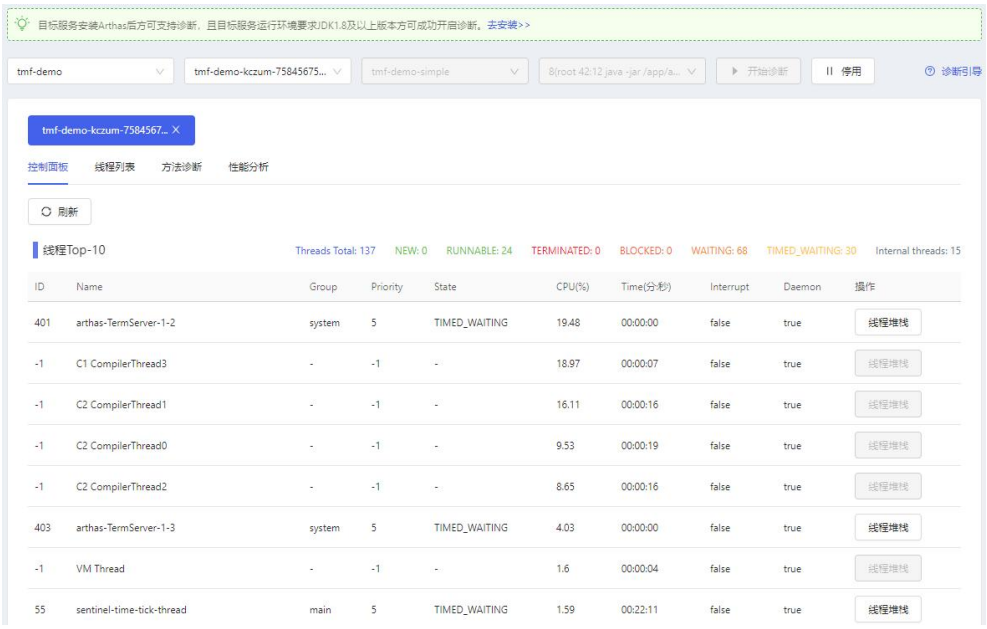
诊断分析详情

- 控制面板

总览系统实时数据，包括：线程 Top-10，JVM 信息，操作系统信息，变量信息。

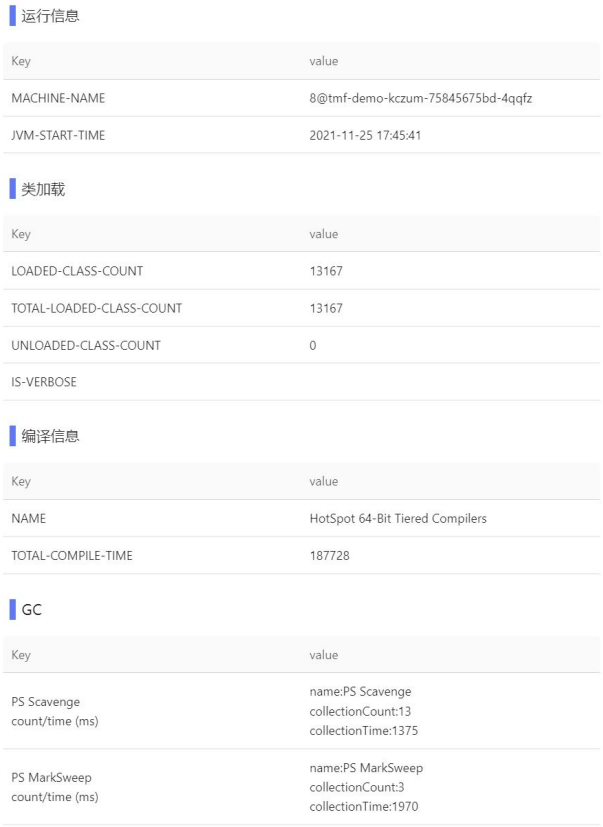
  - 线程 Top-10：统计 CPU 使用率排名前 10 的线程的相关信息，包括：线程 ID、螺纹名称、线程分组、优先级、线程状态、CPU 百分比、线程运行总 CPU 时间、线程是否中断、是否为守护线程、线程堆栈，线程堆栈等信息。

图 6-81 控制面板



- JVM 内存：统计并展示 JVM 相关信息，包括：内存类型（堆、非堆等）、已经使用的内存、总内存、最大内存、内存使用率；GC 类型（GC 次数、GC 耗时等）、统计数据，用户可查看 JVM 统计详情，包括运行信息、类加载、编译信息、GC、内存管理、内存使用、操作系统、线程统计、文件描述等；详情页面如下图所示。此外系统提供查看历史 JVM 内存的功能。

图 6-82 JVM 详情页面（部分）



运行信息

Key	value
MACHINE-NAME	8@tmf-demo-kczum-75845675bd-4qqfz
JVM-START-TIME	2021-11-25 17:45:41

类加载

Key	value
LOADED-CLASS-COUNT	13167
TOTAL-LOADED-CLASS-COUNT	13167
UNLOADED-CLASS-COUNT	0
IS-VERBOSE	

编译信息

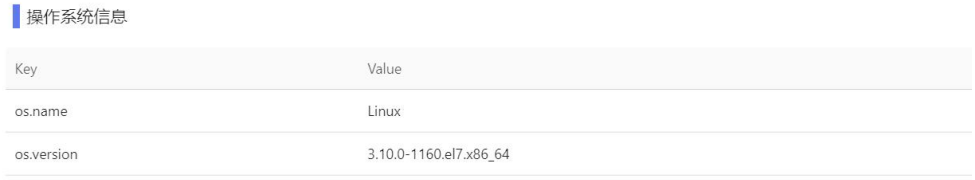
Key	value
NAME	HotSpot 64-Bit Tiered Compilers
TOTAL-COMPILE-TIME	187728

GC

Key	value
PS Scavenge count/time (ms)	name:PS Scavenge collectionCount:13 collectionTime:1375
PS MarkSweep count/time (ms)	name:PS MarkSweep collectionCount:3 collectionTime:1970

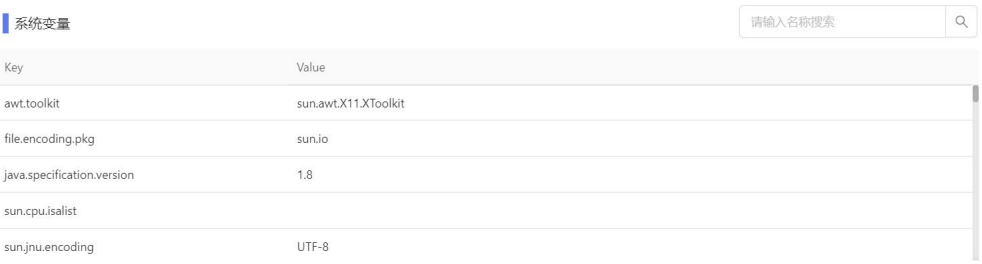
- 操作系统信息：用户可查看操作系统名称、操作系统版本等信息。

图 6-83 操作系统页面



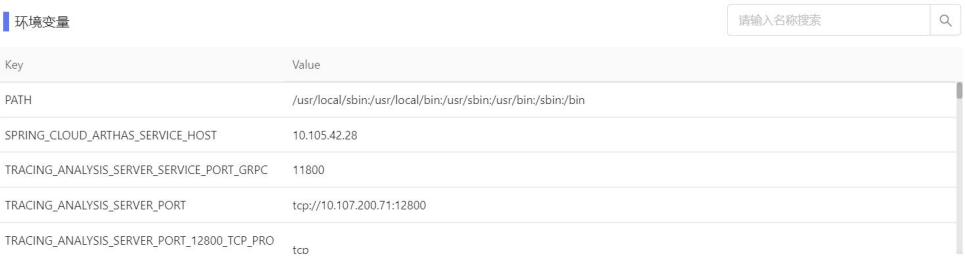
- 系统变量信息：当前服务涉及的系统变量信息，如下图所示。

图 6-84 操作系统变量



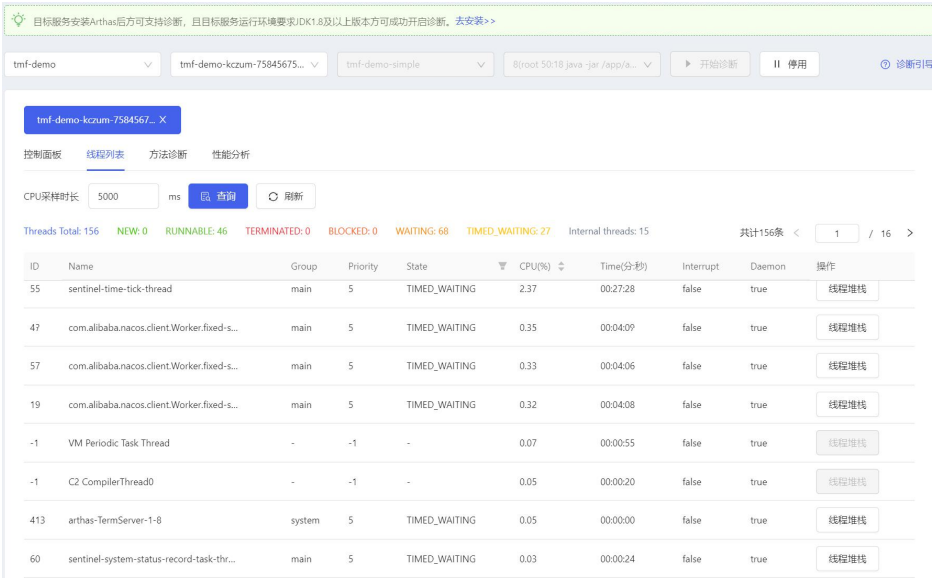
- 环境变量信息：用户当前服务的环境变量信息。

图 6-85 环境变量



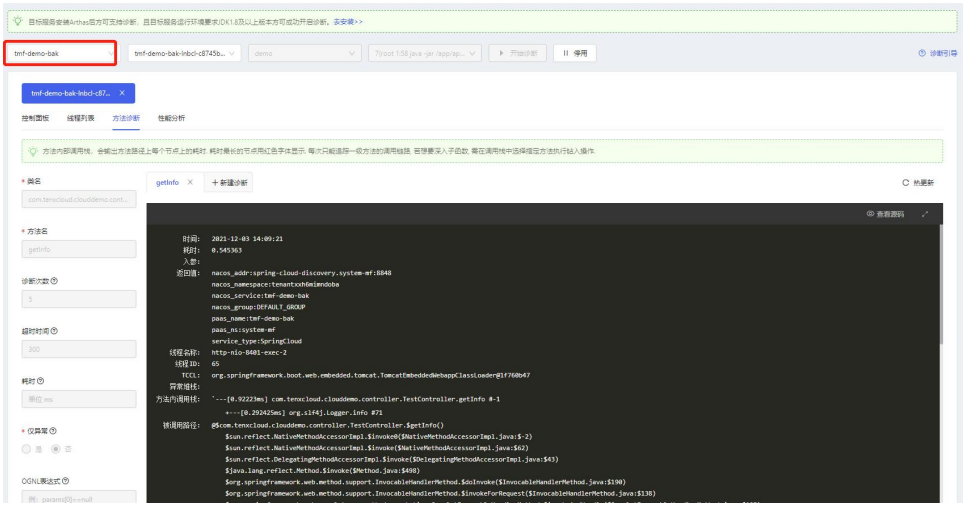
- 线程列表：根据 CPU 采样时长统计系统当前所有线程。统计信息有：线程 ID、螺纹名称、线程分组、优先级、线程状态、CPU 百分比、线程运行总 CPU 时间、线程是否中断、是否为守护线程、线程堆栈、线程堆栈详情等信息。线程列表如下图所示。

图 6-86 线程列表



- 方法诊断：观测指定方法的调用情况，包括：返回值，抛出异常，入参，方法内部调用路径，并输出方法路径上的每个节点上耗时，追溯当前方法被调用的路径。

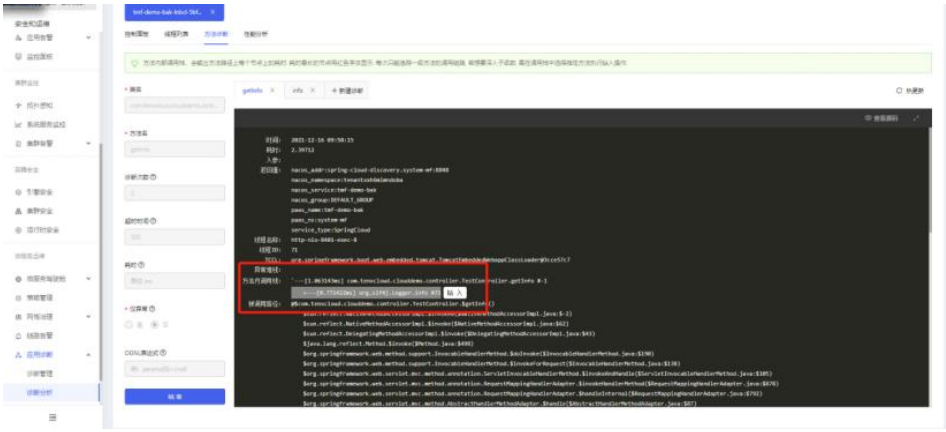
图 6-87 方法诊断



- 类名：服务对应的类名，例如：包为 com.tenxcloud.user，类为 UserService，全类名为 com.tenxcloud.user.UserService。
- 方法名：方法名及参数：方法是代表功能的模块，传递给方法的参数，填写参数对应方法中参数的全类名，且必须和方法的参数顺序一致，并以英文逗号分隔，例如：方法为 public int login(int id, String name)，则方法名及参数为 login(int.class,java.lang.String)。
- 诊断次数：配置抓取调用的次数，达到配置次数系统会自动结束诊断命令。
- 超时时间：配置抓取到调用结果的超时时间，超过配置时间系统自动结束观测命令。

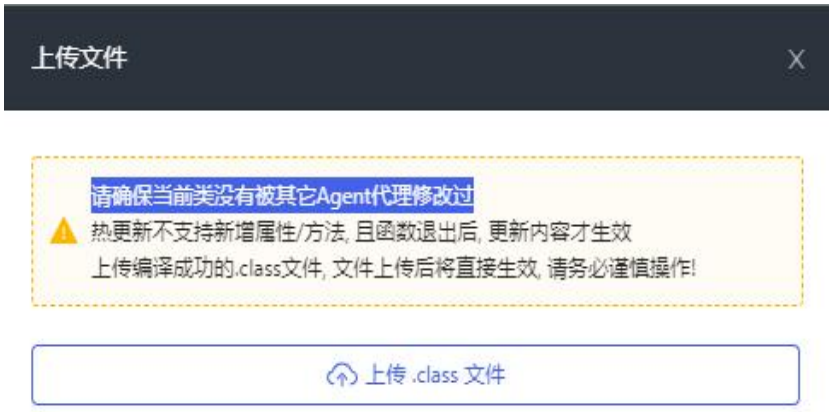
- 耗时：配置系统抓取大于多少耗时的调用。
- 仅异常：选择是否抓取抛出异常的方法。
- OGNL 表达式：对象图导航语言（Object Graphic Navigation Language），用户可根据业务需要配置 OGNL 表达式来筛选关注的代码。
- 钻入：在诊断结果方法内调用栈中点击钻入操作，可进一步对方法进行诊断。

图 6-88 查看页面



- 查看源码：用户可在诊断详情页面反编译的源码。
- 热更新：用户可通过上传编译后的 class 文件，动态更新代码，从而快速解决线上 bug。

图 6-89 热更新文件上传页面



- 性能分析：包括火焰图分析和堆快照分析。

新建火焰图：通过不断的采样，然后把收集到的采样结果生成应用热点火焰图。用户可查看“如何读懂火焰图”获取火焰图的解读方式。

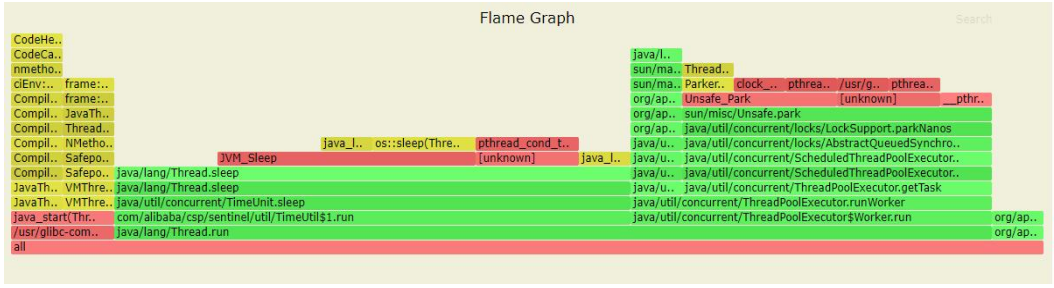
(1) 单击<新建>按钮，在新建火焰图页面配置火焰图的用途，火焰图类型，类型支持：CPU 耗时、内存分配、锁耗时、itimer；文件格式，文件格式支持：svg，html；配置火焰图的采样时长。

(2) 配置完成后，单击<确定>按钮，新建火焰图操作完成。配置示例页面如下图所示。

图 6-90 新建火焰图

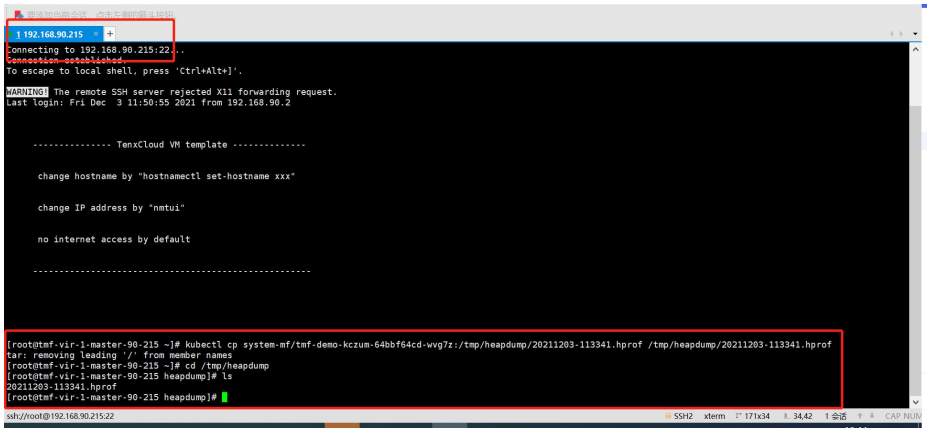
(3) 火焰图生成后，点击对应的文件链接，即可查看火焰图详情，示例页面如下图所示。

图 6-91 火焰图示例页面



转存储堆快照：生成文件包括：对象信息、类信息、GC 的根对象、线程栈及局部变量，借助性能分析工具定位问题(例：MAT），用户可复制下载命令，手动将文件从容器内下载到服务器临时目录。示例页面如下图所示。

图 6-92 示例页面



6.6.4 运维开关

为提高产品的易用性，系统提供运维开关，用户切换至相应的集群即可开启微服务治理的相关功能。配置页面如下图所示。

图 6-93 配置页面

7 常见问题

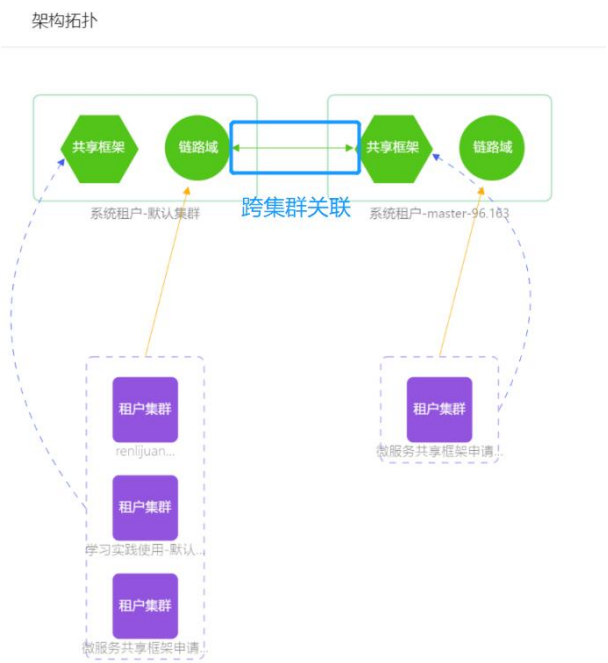
接入 TMF 后，Nacos 注册中心列表中可以查看到，但服务管控列表查询不到，怎么办？

- 进入「服务管控」列表，点击“服务检索”,输入服务名称或接口名称，进行完全匹配查询。一般是由于在接入时 NACOS_NAMESPACE、PAAS_NS、CLUSTER_ID 参数值未配置正确，通过此处查询，可以看到您配置的参数值，如果有误，会显示错误提示。
- 目前 TMF 支持自动注入环境变量，即安装微服务治理框架后，项目管理员开启“微服务治理”开关后，此时再接入 TMF，就自动注入了环境变量， 避免用户手动配置出现问题。
- 如需要获取 NACOS_NAMESPACE、PAAS_NS、CLUSTER_ID 这些值，在「服务管控」列表，点击“环境变量相关说明”可查看环境变量的含义及变量值。
- 按照《微服务治理平台_代码接入指南》检查您的相关配置是否

跨集群关联支持什么场景？

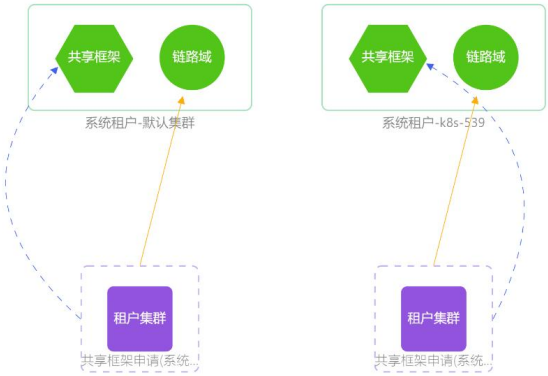
注册中心多集群部署及服务发现说明：

- 场景说明：一个项目授权给不同的集群，用户在每个集群的项目内创建相同的服务，并为该服务设置跨集群访问地址，当其中一个集群的服务出现故障时，可将请求转发到另一个集群的相同服务，从而实现跨集群的服务调用。
- 部署说明：部署两套微服务框架进行跨集群关联设置，前提需要为每个框架中的注册中心服务设置对外访问地址



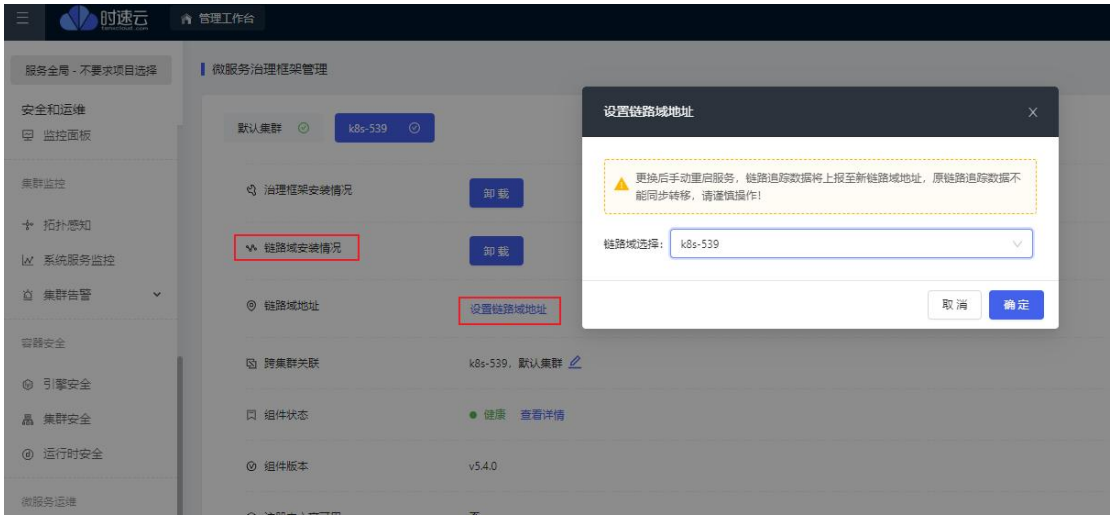
链路域是什么？

- 链路域就是链路追踪组件，可跟随微服务治理框架一起安装，也可独立安装。多个框架可以使用同一套链路域，即框架 A 安装链路追踪组件后，其他框架可选择使用框架 A 的链路追踪组件，将服务调用数据上报至此。（需为框架 A 的链路追踪组件配置外网地址，保证其访问可达）。
- 管理员可在「微服务治理框架管理」中查看架构拓扑。系统租户&默认集群，系统租户& k8s-539 集群，分别部署了共享框架和链路域，业务租户&默认集群申请使用系统租户&默认集群部署的框架及链路域；业务租户& k8s-539 集群申请使用系统租户& k8s-539 集群部署的框架及链路域。



单独安装链路域时的操作注意事项？

安装框架时未同步安装链路域，现需重新在此框架下安装链路域，并设置使用该链路域。且上述操作完成后，需重启该项目命名空间下的所有服务，才可保证服务上报数据到链路域。



调用链查询超时，怎么办？

一般是 ES 处理数据超时，联系管理员查看微服务系统组件的日志进一步判断。

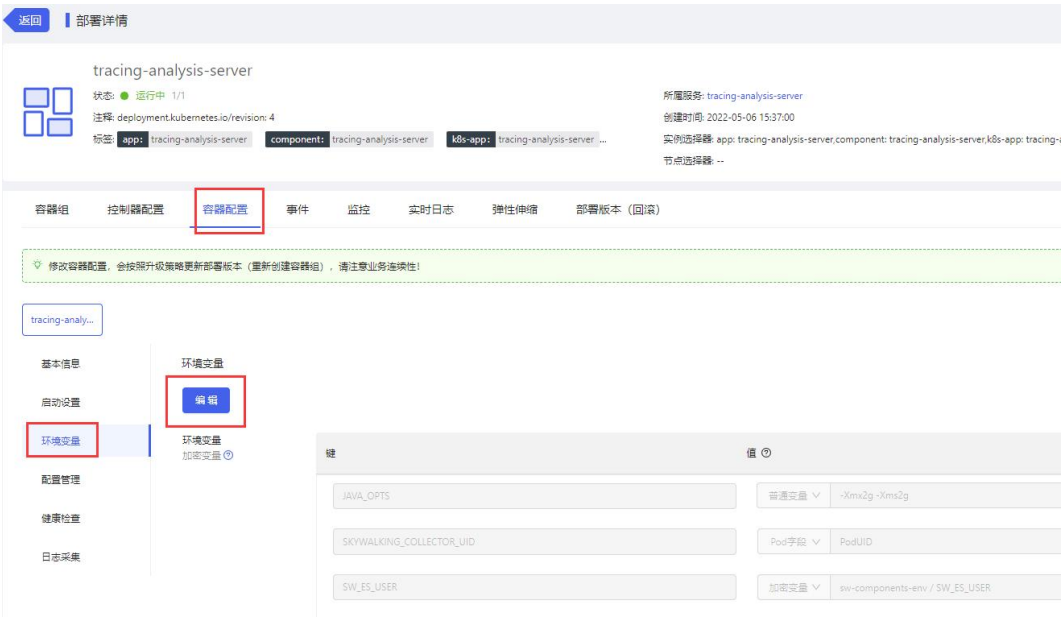
此外，还有网络原因，服务器硬件资源缺乏等。

链路追踪组件启动失败，怎么办？

- 报错内容：xxx does not exist. OAP is running in 'on-init' mode,waiting ... retry 3s later
- 此为 Skywalking 开源组件的 bug，已在版本 9 解决，但目前我们使用的是 8.7 版本，如遇到以上报错，需要修改链路追踪 server 环境变量 JAVA_OPTS 的值为 -Xmx2g -Xms2g。

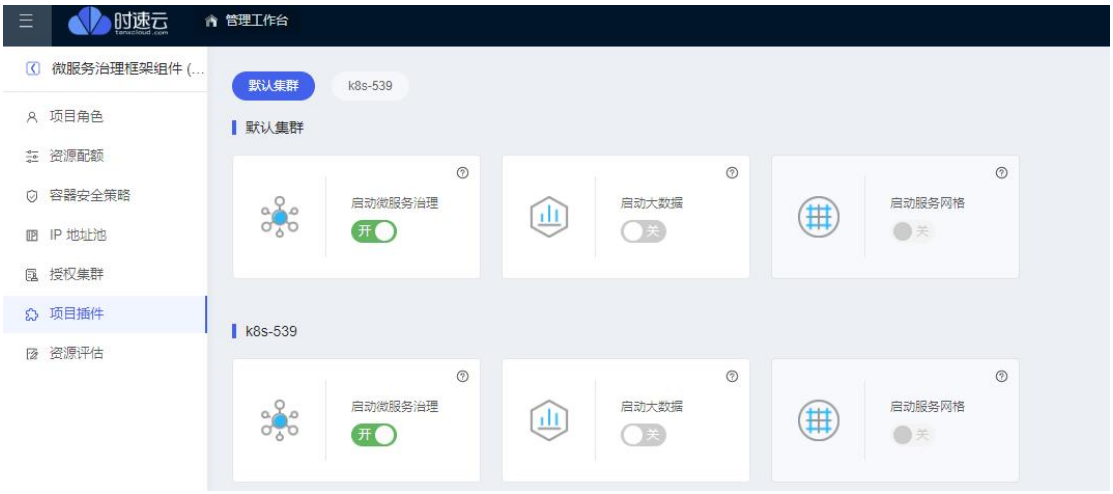
服务接入链路后正常调用，但是概览页面看不到数据，此时日志报错，怎么办？

- 日志报错：Grpc server thread pool is full, rejecting the task，该错误是线程池抛出来的，主要是在链路接入服务较多的情况下发生。需要修改配置如下：
  - SW_CORE_GRPC_POOL_QUEUE_SIZE，默认是 10000，可以改成 20000，这个是线程池队列的大小
  - SW_CORE_GRPC_THREAD_POOL_SIZE，默认是 CPU 核心数的 4 倍，可以改成 50 或者更多，表示线程池的核心线程数和最大线程数
  - 修改步骤：进入<系统租户>- 链路组件 tracing-analysis-server 所在的项目中，如果使用的是 TCE，到容器服务中的「工作负载」-「部署」中找到 tracing-analysis-server，进入其详情页，点击“容器配置”修改环境变量。



微服务治理框架卸载重新安装后，服务注册失败？

微服务治理开关在卸载框架时不自主关闭，需关闭再开启该项目&集群的“微服务治理”开关后，方可注册服务成功（下个版本优化）。



两个 Dubbo 服务有相同接口时，如何隔离？

如果在同一租户&集群中部署两个 Dubbo 服务，这两个服务有相同接口时，建议使用不同的 nacos group 做服务发现的隔离。 不然，如使用相同 nacos group，注册到 nacos 中，会标识为同一个接口服务的两个不同实例，这样无法进行隔离。