

Day3 感知微服务和CSE的交互

1 打卡任务

作业：

1、从ServiceStage工具下载页面下载LocalCSE包，在本地安装运行。

基于Day2的demo，修改provider服务和consumer服务的microservice.yaml配置，令两个微服务注册到LocalCSE，并能够正常相互调用。打卡：

- 1、调用consumer服务的greeting方法成功，并截图
- 2、截取consumer服务的日志图片，要求包含consumer服务实例注册成功的日志、没有连接cc/monitor服务报错的日志
- 2、

打卡任务基于Day2的demo项目：



2 准备工作

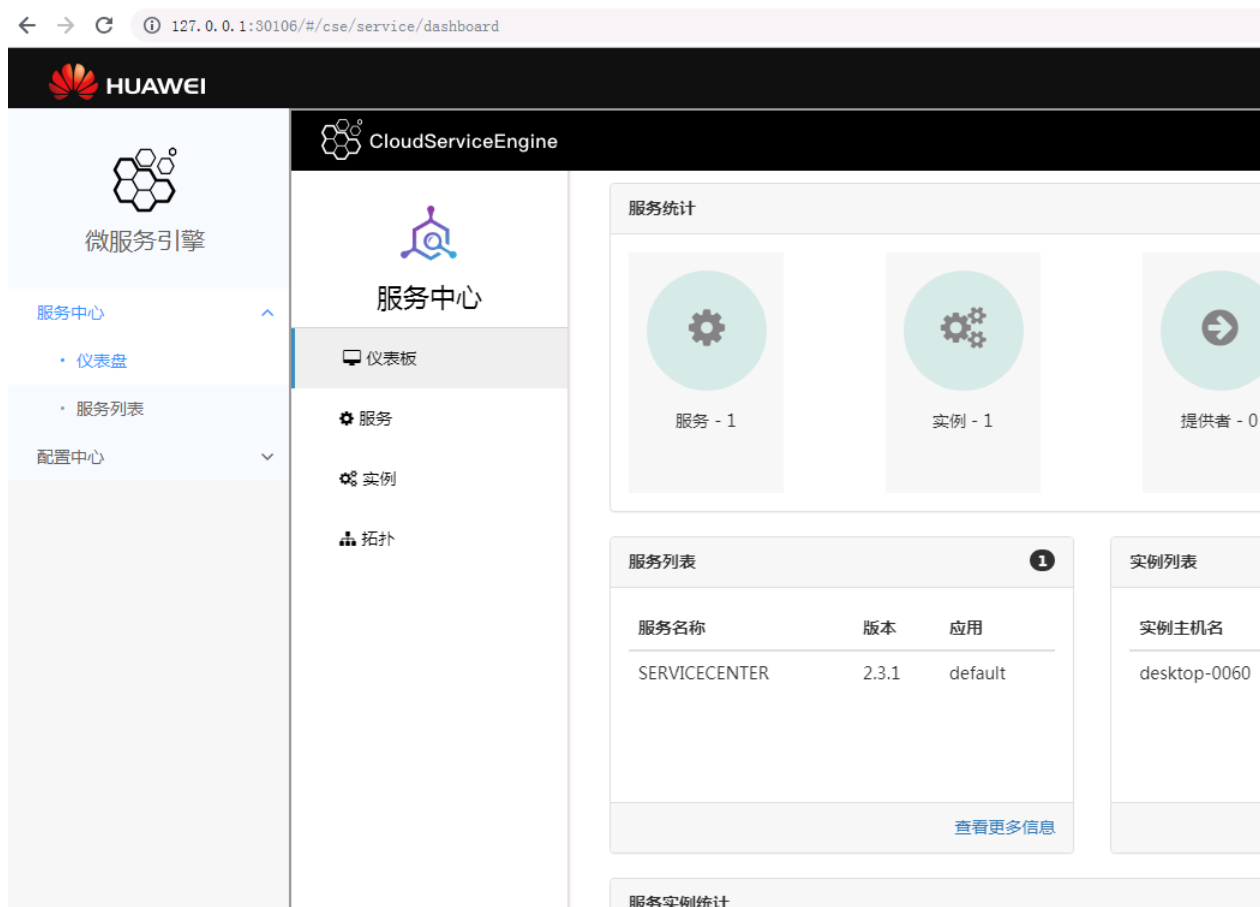
- 1、登录华为云ServiceStage页面，点击进入应用开发 -> 微服务开发 -> 工具下载 -> 本地轻量化微服务引擎，点击下载最新版本。



2、解压下载包，运行其中的start.bat命令，启动LocalCSE

LocalCSE会占用多个端口号，详见 README.md 中的描述，运行时需要保证这些端口没有被其他服务占用。

3、在浏览器输入 <http://127.0.0.1:30106>，能够打开LocalCSE的页面



3 配置微服务连接 LocalCSE

- 1、修改microservice.yaml文件中sc/cc的地址

sc地址改为 <http://127.0.0.1:30100>

cc地址改为 <http://127.0.0.1:30113>

- 2、关闭monitor客户端

仅仅删除monitor的地址不能关闭monitor客户端，微服务会尝试从sc自动发现monitor服务的地址，进而报错。因此需要在 microservice.yaml 文件中配置 servicecomb.monitor.client.enabled=false 显式将其关闭。

- 3、配置完成的microservice.yaml文件如下图所示

```
cse:
  service:
    registry:
      address: http://127.0.0.1:30100
      instance:
        watch: false
    config:
      client:
        serverUri: http://127.0.0.1:30113
        refreshMode: 1
        refresh_interval: 5000
    monitor:
      client:
        serverUri: https://cse.cn-north-1.myhuaweicloud.com:443
        enabled: false
```

4、启动工程，可以看到LocalCSE的服务列表中出现provider和consumer服务的记录

The screenshot shows the Huawei CloudServiceEngine console. On the left is a navigation menu with '微服务引擎' (Microservice Engine) and '服务中心' (Service Center). The '服务中心' section is expanded, showing '仪表盘' (Dashboard), '服务' (Services), '实例' (Instances), and '拓扑' (Topology). The '服务' (Services) tab is selected, displaying a table of services.

| 名称 | 状态 | 应用 | 版本 | 创建时间 |
|---------------|----|---------------------------|-------|-----------------|
| consumer | UP | Training21Days-HelloWorld | 0.0.1 | 2019-2-14 16:5 |
| provider | UP | Training21Days-HelloWorld | 0.0.1 | 2019-2-14 16:0 |
| SERVICECENTER | UP | default | 2.3.1 | 2019-2-14 15:37 |

Page 1 of 1

4 打卡截图

1.调用consumer服务的greeting方法成功，并截图

2.截取consumer服务的日志图片，要求包含consumer服务实例注册成功的日志、没有连接cc/monitor服务报错的日志

调用consumer服务的结果如下：

127.0.0.1:9090/consumer/v0/greeting

POST 127.0.0.1:9090/consumer/v0/greeting

Params Authorization Headers (1) Body Pre-request Script Tests

● none ● form-data ● x-www-form-urlencoded ● raw ● binary JSON (application/json)

```
1 {
2   "name": "Wilson",
3   "gender": "FEMALE"
4 }
```

Body Cookies Headers (2) Test Results Status: 200 OK

Pretty Raw Preview JSON

```
1 {
2   "msg": "Hello, Ms.Wilson",
3   "timestamp": "2019-02-14T08:07:02.953Z"
4 }
```

consumer日志如下:

```
[INFO] read MicroserviceRegisterTask status is FINISHED org.apache.servicecomb.servicereg
[INFO] running microservice instance register task. org.apache.servicecomb.serviceregistr
[INFO] Register microservice instance success. MicroserviceId=134332a31760cefff9103781f67
[INFO] receive MicroserviceInstanceRegisterTask event, check instance Id... org.apache.se
[INFO] instance registry succeeds for the first time, will send AFTER_REGISTRY event. org
[WARN] keyStore [server.p12] file not exist, please check! org.apache.servicecomb.foundat
[WARN] trustStore [trust.jks] file not exist, please check! org.apache.servicecomb.founda
[INFO] Monitor data sender started. Configured data providers is {com.huawei.paas.monitor
[INFO] ServiceComb is ready. org.apache.servicecomb.core.SCBEEngine$1.afterRegistryInstanc
[INFO] read MicroserviceInstanceRegisterTask status is FINISHED org.apache.servicecomb.se
[INFO] Waiting for status up. timeout: 10000ms org.apache.servicecomb.core.SCBEEngine.wait
[INFO] Status already changed to up. org.apache.servicecomb.core.SCBEEngine.waitForStatusUp(S
[INFO] sc task interval changed from -1 to 30 org.apache.servicecomb.serviceregistry.task
[INFO] Found SPI service javax.ws.rs.core.Response$StatusType, count=0. org.apache.servic
[INFO] Found SPI service org.apache.servicecomb.core.tracing.TraceIdGenerator, count=1. o
[INFO] ... 0. org.apache.servicecomb.core.tracing.BraceTraceIdGenerator. org.apache.service
```

5 小提示

1、关于配置项前缀: 有些同学可能注意到了, 课程资料中写的sc/cc/monitor配置的前

缀是servicecomb，但是示例中的配置项前缀是cse。这是为了开源的ServiceComb-Java-Chassis和CSEJavaSDK相互兼容，框架内部自动将配置项做了映射，cse. 开头的配置项也会映射一份 servicecomb. 开头的。

参考答案：

