

SOFAStack

微服务 部署指南

产品版本：AntStack Plus 1.11.0

文档版本：20220929




法律声明

蚂蚁集团版权所有©2022，并保留一切权利。

未经蚂蚁集团事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

商标声明

 蚂蚁集团
ANT GROUP 及其他蚂蚁集团相关的商标均为蚂蚁集团所有。本文档涉及的第三方的注册商标，依法由权利人所有。

免责声明

由于产品版本升级、调整或其他原因，本文档内容有可能变更。蚂蚁集团保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在蚂蚁集团授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过蚂蚁集团授权渠道下载、获取最新版的用户文档。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <code>Instance_ID</code>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1.前置条件	05
2.准备工作	07
3.产品发布	11
3.1. 发布 MS	11
3.2. 发布 CloudMesh	13

1. 前置条件

产品部署依赖

- [核心产品+IAAS 产品]依赖产品列表：数据库、ALB、IAM、OP（以现场环境版本为准，一般没特殊版本要求）
- 本文档主要是解决 Mesh 的部署问题，请提前完成底座、中枢集群、云游的部署。部署方式请参见《微服务部署指南》。

解决方案和部署顺序

服务网格部署涉及以下产品，部署顺序按表格从上往下。

产品码	所属产品集	部署拓扑	部署规格	版本	是否必选
REGISTRY	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	2.1.1 及以上	是
ACVIP	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	0.18.0 及以上	是
OSP	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	1.18.0 及以上	是
INTELLIPROXY	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	1.10.0 及以上	是
PORTAL	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	1.13.0 及以上	是
MS	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	2.28.1 及以上	是
DST	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房	5.2.0 及以上	否
ANT MONITOR	共享中间件 1.9.0 及以上	标准单机房部署	标准生产-单机房		否
CLOUDMESH	共享中间件 1.9.0 及以上	标准单机房-公 专合一	标准生产-单机房	1.7.3	是

② 说明

- 建议根据对应环境发布对应产品版本。
- 若不部署 DST，会缺少服务拓扑产品，前端可以通过配置隐藏掉该菜单。
- 若不部署 ANTMONITOR，会缺少实时监控产品，前端可以通过配置隐藏掉菜单。

2.准备工作

准备机器资源

类型	应用	规格	数量
服务器	meshopenapi	4C/8G/100G	2
	meshpilot	8C/16G/100G	4
	operator	2C/4G/100G	3
	demotest	2C/4G/100G	5
数据库	rds4openapi	4C/2G/100G	1
存储	osscloudmesh		1
SLB	meshpilot		1
	galley		1
	meshapi		1
	meshopenapi		1
	operatorvmbbox		1

获取镜像下载地址

您可以在 Global 控制台获取上述产品的镜像，当 Docker 安装完成后，您可以通过 `docker pull` 命令下载这些镜像。

1. 登录云游 Global 控制台。
2. 进入目标环境，然后在左侧导航栏单击 交付升级。
3. 单击 规划，然后单击 产品 页签。

4. 选择要下载镜像的产品名，然后单击 **下载镜像** 获取下载链接。



5. 在 **镜像下载地址** 对话框单击 **复制**。



镜像清单

产品名称	镜像名称
MS	after-antscheduler-console
	after-confreg-console
	after-ddcs-console
	after-registry-console
	antscheduler
	antschedulerconsole

	drmdata
	dsrconsole
	mstest
CLOUDMESH	meshopenapi
	meshpilot
	operator
	mosn
应用镜像（CAFE 场景需要将这两个 demo 镜像上传到 CAFE 的镜像仓库）	sofa-server
	sofa-client

tar 包清单

除镜像文件以外的其他文件：[osscloudmesh-osscloudmesh.zip](#)

全局参数核对

以下参数基本按照现场安装环境渲染，不用做特殊修改。

key	value	备注
cloudAccessKey	\${中枢账号的 AK}	当前站点的 AccessKey。 本演示环境设置为： antCloudAdmin
cloudAccessSecret	\${中枢账号的 AKS}	当前站点的 AccessSecret。 本演示环境设置为：antCloud123
yunyou_access_key	yunyou_access_key	固定值
yunyou_access_secret	yunyou_access_secret	固定值

iaas_type	AntStack	IaaS 类型
com_alipay_instanceid	\${prod.MS.com_alipay_instanceid}	中间件实例 ID（默认值000001）
cell	default	双机房部署需要修改，单机房保持默认。双机房示例： defaultA defaultB
datacenter	default	双机房部署需要修改，单机房保持默认。双机房示例： cn-shenzhen cn-wuhan
protocol	http://	保持默认值，直接确认即可。
deploy_type	private	保持默认值，直接确认即可。
log_level	info	保持默认值，直接确认即可。
zmode	false	保持默认值，直接确认即可。
com_alipay_env	shared	保持默认值，直接确认即可。
dbmode	prod	保持默认值，直接确认即可。
site	antfin	保持默认值，直接确认即可。
timezone	Asia/Shanghai	国内站点默认即可。
TZ	\${env.info.timezone}	云游 2.8.10 支持，解决监控国际化问题。
zone	gz00a	双机房部署需要修改，单机房保持默认。双机房示例： gz00a gz00b

3. 产品发布

3.1. 发布 MS

操作步骤

1. 登录云游 Local。
2. 单击 **交付升级**，然后单击 **MS 的解决方案**。
3. 单击 **解决方案元数据**，然后单击 **参数**。
4. 修改 `console.mesh.enabled` 参数。
找到 `console.mesh.enabled` 参数，然后单击 **编辑**，将 **值模板** 修改为 `true`。
您可以通过搜索栏快速查找 `new_site` 参数。
5. 单击 **解决方案部署**，然后单击 **批量发布**。
6. 选择所有产品后，单击 **确定**。
7. 发布完成后，单击 **测试**。
8. 单击 **开始测试**，然后单击 **确定**。
完成测试后，可单击 **测试单**，查看测试详情。

后续操作

1. 登录微服务控制台。
租户选择 **ALIPAYCN**，工作空间选择 **middleware**。
2. 选择 **微服务 > 动态配置**。
3. 单击 **新增配置类**，然后配置如下参数：

参数	说明
域	配置为：Alipay。
所属应用	配置为：dsrconsole。
类标识	配置为：com.alipay.antcloud.dsrconsole.endpoint.webui。
描述（可选）	配置类的描述信息，可根据需要配置。

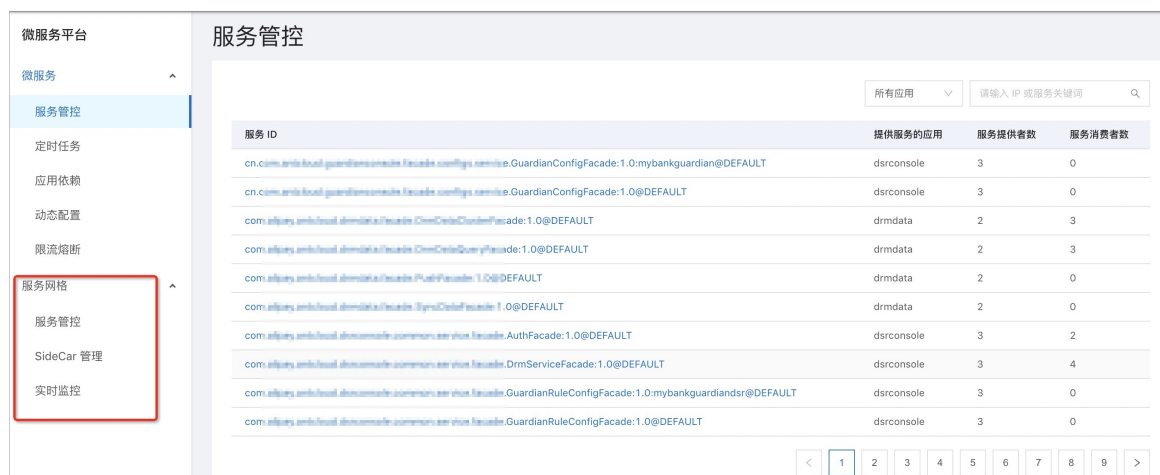
4. 在配置类列表中，单击目标配置类右侧的 **新增属性**。
5. 将属性名配置为 **menus**，然后单击 **确定**。
6. 单击配置类左侧的加号（+），然后单击 **menus** 属性名称。
7. 配置 **推送值** 后，单击 **推送配置**，然后单击 **确定**。

推送值设置为：

```
{
  "displayNameCN": "微服务",
  "type": "singular",
  "isChildrenSameApp": true,
  "hasChildren": true,
  "apps": [
    {
      "name": "ms",
      "entryHTML": "/index.local.html",
      "routerPrefix": "#/ms"
    }
  ],
  "menus": [
    {
      "appName": "dsrconsole",
      "displayNameCN": "经典微服务",
      "hasChildren": true,
      "children": [
        {
          "displayNameCN": "服务管控",
          "link": "#/ms/classic/service-governance"
        },
        {
          "displayNameCN": "应用依赖",
          "link": "#/ms/classic/app-dependency"
        },
        {
          "displayNameCN": "动态配置",
          "link": "#/ms/classic/dynamic-configuration"
        },
        {
          "displayNameCN": "服务限流",
          "link": "#/ms/classic/mybank-guardian"
        },
        {
          "displayNameCN": "服务治理",
          "link": "#/ms/classic/service-configuration"
        }
      ]
    }
  ],
  {
    "displayNameCN": "服务网格",
    "hasChildren": true,
    "children": [
      {
        "displayNameCN": "服务管控",
        "link": "#/ms/cloud/service-governance"
      },
      {
        "displayNameCN": "服务治理",
        "link": "#/ms/cloud/service-configuration"
      }
    ]
  }
}
```

```
{
  "displayNameCN": "SideCar 管理",
  "link": "#/ms/cloud/sideCarManage"
}
```

8. 查看 **微服务** 下方是否出现 **服务网格** 的菜单。



3.2. 发布 CloudMesh

发布 CloudMesh 前，您需要修改如下参数，步骤可以参考 [发布 MS](#)。

应用名	键	值模板
operator	RUN_MODE	<ul style="list-style-type: none">1.7.x 版本: cloud1.8.x 版本: white-config
meshcoreinit	global_skip	第一次部署需要改为: false。
cloudmeshtest	KUBECONFIG	环境的 kubeconfig 证书。 您可以在中枢集群中，通过如下命令获取： <pre>cat ~/.kube/config sed 's/^/ /g'</pre>
cloudmeshtest	com.antcloud.clusterName	环境存在 clustername 时，需要修改成对应值。
CLOUDMESH	mosn_image_address	需要修改为对应版本的 MOSN 镜像。

后续操作

1. 检查 Pod 状态。

在中枢集群中输入以下命令，检查三个应用 Pod 状态全部为 running，并且 ready 全部为 1/1

```
kubectl get pod -A | grep mesh
```

```
[root@master11111.wga ~]# kubectl get pod -A | grep mesh
asp-cloudmesh          asp-cloudmesh-meshopenapi-0      1/1    Running    0          8m24s
asp-cloudmesh          asp-cloudmesh-meshpilot-0        1/1    Running    0          7m51s
asp-cloudmesh          asp-cloudmesh-operator-0         1/1    Running    0          7m51s
```

2. 测试验证。

? 说明

CAFE Standard 必须进行该步骤，其他产品根据需要选择是否进行验证。

o 检查 ValidatingWebhookConfiguration

执行如下命令：

```
kubectl get ValidatingWebhookConfigurationgalley
```

执行后不能有输出，如果有输出，执行如下命令：

```
kubectl delete ValidatingWebhookConfigurationgalley
```

o 检查 MutatingWebhookConfiguration sidecar-operator-webhook

执行如下命令：

```
kubectl get MutatingWebhookConfiguration sidecar-operator-webhook
```

执行后不能有输出，如果有输出，执行如下命令：

```
kubectl delete MutatingWebhookConfiguration sidecar-operator-webhook
```

o 验证 Pod

执行以下命令确认 Pod 可以创建：

```
kubectl apply -f test-inject.yaml
```

```
test-inject.yaml  内容为：
```

```
apiVersion: v1
kind: Pod
metadata:
  name: test-inject
  annotations:
    aks.cafe.sofastack.io/mosn-inject: enabled
  labels:
    monotype: hard
spec:
  hostNetwork: true
  containers:
  - name: debug
    image: acs-reg.alipay.com/cloudnative/sidecar-operator-webhook:v0.1.3
    command: ["sleep", "100000"]
```

确认 Pod 可以成功创建，创建后有 1 个 container，并且都是 running 状态。验证完成后，执行以下命令删除测试 Pod。

```
kubectl delete -f test-inject.yaml
```