Serverless 应用中心

快速入门

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

快速入门
安装 Serverless Cloud Framework
快速部署函数模板
快速创建应用模板
控制台开发指南

版权所有：腾讯云计算（北京）有限责任公司

快速入门
安装 Serverless Cloud Framework

最近更新时间：2024-07-18 14:09:11

您可以通过 NPM 安装 Serverless Cloud Framework。

注意事项

为了提升产品体验，云函数 SCF 和 Serverless 应用已于2024年7月1日进行了 CLI 版本升级，以支持函数 URL。使用方式如下：

升级到新版本 CLI

您可以升级到新版本 CLI（serverless-cloud-framework@1.2.0），并在 YAML 文件中配置函数 URL 相关参数后部署函数或应用。
参数配置参考（详情请参见组件 YAML 配置）：

```
events:
  - http:
      parameters:
        netConfig:
          enableIntranet: true
          enableExtranet: true
        qualifier: $DEFAULT
        authType: NONE
```

继续使用老版本 CLI

如果您不需要使用函数 URL 功能，可以继续使用老版本 CLI（serverless-cloud-framework@1.2.0 以下版本）。

> ⚠ **注意：**
> 如果您此前已通过控制台或云 API 创建过函数 URL，但在使用新版本 CLI 部署时未在 YAML 中配置函数 URL，将视为不使用函数 URL 功能，原有的函数 URL 将不可用。

通过 NPM 安装

安装前提

使用 npm 安装前，需要确保您的环境中已安装好了 Node（**版本需要 > 12**）以及 npm（查看 Node.js 安装指南）。

```
$ node -v
v12.18.0

$ npm -v
7.0.10
```

注意：

- 为保证安装速度和稳定性，建议您使用 cnpm 来完成安装：先下载安装 cnpm，然后将下面所有使用的 npm 命令替换为 cnpm 即可。
- `serverless-cloud-framework` 命令的简写为 `scf` 。

安装步骤

在命令行中运行如下命令：

```
npm i -g serverless-cloud-framework
```

说明：
如 MacOS 提示无权限，则需要运行 `sudo npm i -g serverless-cloud-framework` 进行安装。

如果之前您已经安装过 Serverless Cloud Framework，可以通过以下命令升级到最新版。

```
npm update -g serverless-cloud-framework
```

查看版本信息

安装完毕后，通过运行 `scf -v` 命令，查看 Serverless Cloud Framework 的版本信息：

```
scf -v
```

相关操作

下一步：快速开始

- 快速部署函数模板
- 快速创建应用模板

快速部署函数模板

最近更新时间：2023-07-10 12:28:57

操作场景

该任务指导您通过 Serverless Cloud Framework，在腾讯云上快速创建、配置和部署一个 SCF 云函数应用。

前提条件

已经 注册腾讯云账号 并完成 实名认证 。

> **说明**
> 如果您的账号为**腾讯云子账号**，请先联系主账号，参考 账号和权限配置 进行授权。

操作步骤

快速部署

在**空文件夹**目录下，执行如下指令：

```
serverless-cloud-framework
```

接下来按照交互提示，完成项目初始化，应用请选择 `scf-starter` 模板，并选择您希望用的运行时（此处以 Node.js 为例）：

```
serverless-cloud-framework: 当前未检测到 Serverless 项目，是否希望新建一个项
目? Yes
serverless-cloud-framework: 请选择您希望创建的 Serverless 应用 scf-starter -
快速部署一个云函数

  react-starter - 快速部署一个 React.js 应用
  restful-api - 快速部署一个 REST API 使用 python + API gateway
› scf-starter - 快速部署一个云函数
  vue-starter - 快速部署一个 Vue.js 基础应用
  website-starter - 快速部署一个静态网站
  eggjs-starter - 快速部署一个Egg.js 基础应用
  express-starter - 快速部署一个 Express.js 基础应用

serverless-cloud-framework: 请选择应用的运行时 scf-nodejs - 快速部署一个
nodejs 云函数
```

```
  scf-golang - 快速部署一个 golang 云函数
❯ scf-nodejs - 快速部署一个 nodejs 云函数
  scf-php - 快速部署一个 PHP 云函数
  scf-python - 快速部署一个 python 云函数

serverless-cloud-framework: 请输入项目名称 demo
serverless-cloud-framework: 正在安装 scf-nodejs 应用...

scf-nodejs › Created


demo 项目已成功创建!
```

选择**立即部署**，将已经初始化好的项目快速部署到云函数控制台：

```
serverless-cloud-framework: 是否希望立即将该项目部署到云端? Yes

xxxxxxxx
x  QR  x
x CODE x
xxxxxxxx
请使用微信扫描上方二维码或者点击下方链接登录
https://slslogin.qcloud.com/********
登录成功!

serverless-cloud-framework
Action: "deploy" - Stage: "dev" - App: "scfApp" - Name: "scf-nodejs"

code:
  bucket: sls-cloudfunction-ap-guangzhou-code
  object: /scf_component_******-**********.zip
description:           This is a function in scfApp application
functionName:          scf-nodejs-dev-scfApp
handler:               index.main_handler
lastVersion:           $LATEST
memorySize:            128
namespace:             default
runtime:               Nodejs10.15
sourceCodeDownloadUrl: https://sls-app-code-prod-**********.cos.ap-
guangzhou.myqcloud.com/***
triggers:
  -
    NeedCreate:  true
```

```
    apiList:
      -
        apiId:           api-********
        apiName:         index
        authType:        NONE
        businessType:    NORMAL
        created:         true
        isBase64Encoded: false
        method:          GET
        path:            /
    created:     true
    environment: release
    protocols:   http
    serviceId:   service-********
    serviceName: serverless
    subDomain:   service-********-**********.gz.apigw.tencentcs.com
    urls:
      - http://service-********-
**********.gz.apigw.tencentcs.com/release/
type:                 event

应用控制台：https://serverless.cloud.tencent.com/apps/scfApp/scf-
nodejs/dev

184s »scf-nodejs» 执行成功
```

部署完毕后，通过以下指令，完成函数的远程调用：

```
scf invoke --inputs function=helloworld
```

> **说明**
> `serverless-cloud-framework` **命令的简写为** `scf` **。**

查看部署信息

如果希望再次查看应用的部署状态和资源，可以进入到部署成功的文件夹，运行如下命令，查看对应信息：

```
cd demo #进入项目目录，此处请改为您的项目目录名称
scf info
```

查看目录结构

在初始化的项目目录下，可以看到一个 Serverless 函数项目的最基本结构：

```
.
├── serverless.yml   # 配置文件
├── index.js     # 入口函数
└── .env # 环境变量文件
```

- serverless.yml 配置文件实现了函数基本信息的快速配置，函数控制台支持的配置项都支持在 yml 文件里配置（查看 云函数的全量配置信息 ）。
- index.js 为项目的入口函数，此处为 helloworld 模板。
- .env 文件里存放了用户登录的鉴权信息，您也可以在里面配置其它环境变量。

重新部署

在本地项目目录下，您可以对函数模板项目内容与配置文件进行修改，并通过以下指令进行重新部署：

```
scf deploy
```

> **说明**
> 如需查看移除过程中的详细信息，可以增加 `--debug` 参数进行查看。

持续开发

部署完成后，Serverless Cloud Framework 支持通过不同指令，帮助您完成项目的持续开发部署、灰度发布等能力，您也可以结合其它组件一起使用，完成多组件应用的部署管理。
详情请参考文档 应用管理 与 支持命令列表 。

常见问题

- **问题1：输入 `serverless-cloud-framework` 时没有默认弹出中文引导。**
  解决方案： 在 .env 文件中增加配置 SERVERLESS_PLATFORM_VENDOR=tencent 即可。
- **问题2：在境外网络环境，输入 `scf deploy` 后部署十分缓慢。**
  解决方案：在 .env 文件中增加配置 `GLOBAL_ACCELERATOR_NA=true` 则开启境外加速 。
- **问题3：输入 `scf deploy` 后部署报网络错误。**
  解决方案：在 .env 文件中增加以下代理配置。

```
HTTP_PROXY=http://127.0.0.1:12345 #请将'12345'替换为您的代理端口
HTTPS_PROXY=http://127.0.0.1:12345 #请将'12345'替换为您的代理端口
```

快速创建应用模板

最近更新时间：2023-07-13 11:02:56

操作场景

该任务指导您通过 Serverless Cloud Framework，在腾讯云上快速创建、配置和部署一个 Web 框架应用。

前提条件

已经 注册腾讯云账号 并完成 实名认证 。

> **说明**
> 如果您的账号为**腾讯云子账号**，请先联系主账号，参考 账号和权限配置 进行授权。

操作步骤

快速部署

在**空文件夹**目录下，执行如下指令：

```
serverless-cloud-framework
```

接下来按照交互提示，完成项目初始化，应用请选择您希望部署的应用框架模板（此处以 Express 为例）：

```
serverless-cloud-framework: 当前未检测到 Serverless 项目，是否希望新建一个项
目? Yes
serverless-cloud-framework: 请选择您希望创建的 Serverless 应用 express-
starter

  eggjs-starter - Egg.js 项目模版
❯ express-starter - Express.js 项目模版
  flask-starter - Flask 项目模版
  fullstack - 快速部署一个 Full Stack 应用, vuejs + express + postgres
  koa-starter - Koa.js 项目模版
  laravel-starter - Laravel 项目模版
  nextjs-starter -  nextjs 函数项目模版

serverless-cloud-framework: 请输入项目名称 demo
serverless-cloud-framework: 正在安装 express-starter 应用...
```

```
express-starter › Created


demo 项目已成功创建!
```

选择立即部署，将已经初始化好的项目快速部署腾讯云平台：

```
Serverless: 是否希望立即将该项目部署到云端?  Yes

xxxxxxxx
x  QR  x
x CODE x
xxxxxxxx
请使用微信扫描上方二维码或者点击下方链接登录
https://slslogin.qcloud.com/********
腾讯云 登录成功!

serverless-cloud-framework
Action: "deploy" - Stage: "dev" - App: "demo" - Name: "expressDemo"

apigw:
  apiList:
    -
      apiId:            api-********
      apiName:          http_api
      authType:         NONE
      businessType:     NORMAL
      created:          true
      isBase64Encoded: false
      method:           ANY
      path:             /
  created:     true
  environment: release
  id:          service-********
  subDomain:   service-********.gz.apigw.tencentcs.com
  url:         https://service-********.gz.apigw.tencentcs.com/release/
faas:
  name:      expressDemo
  namespace: default
  runtime:   Nodejs12.16
  type:      web
region:                ap-guangzhou
```

```
sourceCodeDownloadUrl: https://sls-app-code-prod-**********.cos.ap-guangzhou.myqcloud.com/***
应用控制台：https://serverless.cloud.tencent.com/apps/demo/expressDemo/dev

32s »expressDemo» 执行成功
```

部署完毕后，单击命令行输出的 API 网关链接，即可快速访问已部署好的 Web 框架应用。

查看部署信息

如果希望再次查看应用的部署状态和资源，可以进入到部署成功的文件夹，运行如下命令，查看对应信息：

```
cd demo #进入项目目录，此处请改为您的项目目录名称
scf info
```

> **说明**
> `serverless-cloud-framework` 命令的简写为 `scf` 。

查看目录结构

在初始化的项目目录下，可以看到一个 Express 项目的最基本结构：

```
.
├── serverless.yml   # 配置文件
├── index.js     # 入口函数
├── package.json # 项目依赖
└── .env # 环境变量文件
```

- serverless.yml 配置文件实现了函数基本信息的快速配置，函数控制台支持的配置项都支持在 yml 文件里配置（查看 云函数的全量配置信息 ）。
- index.js 为项目的入口函数，此处为 helloworld 模板。
- package 为项目依赖文件，记录了该 Node.js 框架项目需要安装的依赖包。
- .env 文件里存放了用户登录的鉴权信息，您也可以在里面配置其它环境变量。

重新部署

在本地项目目录下，您可以对函数模板项目内容与配置文件进行修改，重新安装依赖后，通过以下指令进行重新部署：

```
npm install && scf deploy
```

> **说明**
> 如需查看移除过程中的详细信息，可以在 `scf deploy` 后增加 `--debug` 参数进行查看。

持续开发

部署完成后，登录 Serverless 应用控制台，查看项目部署后输出的基本信息、项目请求次数、项目报错统计等多项监控指标，并实现项目持续开发与部署。详情请参考 控制台开发文档 。
Serverless Cloud Framework 支持通过不同指令，帮助您完成项目的持续开发部署、灰度发布等能力，您也可以结合**层**、**自定义域名**等其它高级能力一起使用，实现应用的高级能力配置。

常见问题

- **问题1：输入 `scf` 时没有默认弹出中文引导。**
  解决方案： 在 .env 文件中增加配置 SERVERLESS_PLATFORM_VENDOR=tencent 即可。
- **问题2：在境外网络环境，输入 `scf deploy` 后部署十分缓慢。**
  解决方案：在 .env 文件中增加配置 `GLOBAL_ACCELERATOR_NA=true` 则开启境外加速 。
- **问题3：输入 `scf deploy` 后部署报网络错误。**
  解决方案：在 .env 文件中增加以下代理配置。

```
HTTP_PROXY=http://127.0.0.1:12345 #请将'12345'替换为您的代理端口
HTTPS_PROXY=http://127.0.0.1:12345 #请将'12345'替换为您的代理端口
```

控制台开发指南

最近更新时间：2025-06-12 15:25:41

操作场景

针对常用框架组件，您可以通过 Serverless 应用控制台 快速实现完整的应用开发部署流程。

前提条件

在使用控制台部署前，您需要先完成以下权限配置：

主账号授权

1. 登录 Serverless 应用控制台 ，单击**前往授权**进入访问管理控制台。
2. 在访问管理控制台的**角色**列表页，查看 **SLS_QcsRole** 和 **CODING_QCSRole** 服务角色是否创建成功。

> ⚠ **注意：**
> 如果您已经创建过 CODING_QCSRole，请检查角色拥有权限是否完整，该角色需要基本策略列表如下：QcloudSLSFullAccess、QcloudSSLFullAccess、QcloudAccessForCODINGRole，如有缺失，请手动添加。

3. 确定角色与权限都符合要求后，即可开始使用服务。

子账号授权

如果未开通 **Serverless 应用中心** 和 **Coding DevOps** 的服务，请先与主账号联系，完成服务开通与角色创建。同时，用户腾讯云子账号需完成与 CODING 账号的关联。

操作步骤

步骤1：创建应用

1. 登录 Serverless 应用控制台 。
2. 单击**新建应用**，进入项目创建页面。
3. 选择应用创建方式，支持**应用市场创建**和 **Web 应用创建**两种方式，您可以根据自己的实际情况，选择相应的创建方案。

应用市场创建
如果选择**应用市场**创建，您可以通过选择控制台提供的项目模板，快速创建一个应用。当前支持模板如下图所示：

您也可以通过以下最佳实践文档了解更多：

- 部署融合媒体转码应用

Web 应用创建

如果选择Web 应用创建，您可以通过选择控制台提供的框架模板，快速创建一个 web 应用。当前支持框架如下：

支持框架	相关文档
Express 框架	快速部署 Express 框架

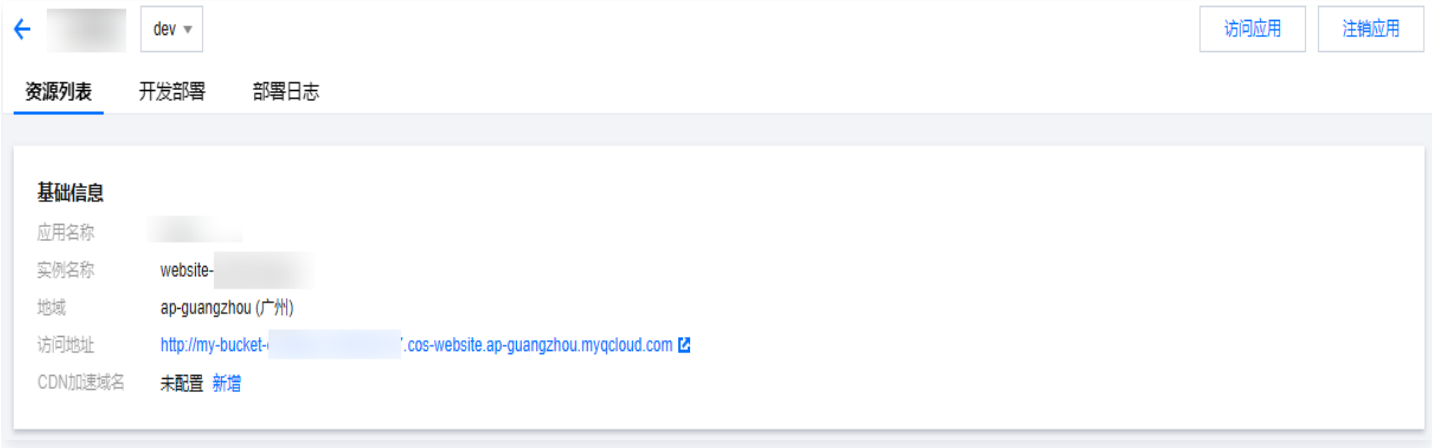
Koa 框架	快速部署 Koa 框架
Egg 框架	快速部署 Egg 框架
Next.js 框架	快速部署 Nextjs 框架
Nuxt.js 框架	快速部署 Nuxtjs 框架
Nest.js 框架	快速部署 Nestjs 框架
Flask 框架	快速部署 Flask 框架
Django 框架	快速部署 Django 框架
Laravel 框架	快速部署 Laravel 框架

4. 单击**下一步**，根据页面提示，填写应用基本信息。

5. 单击**完成**，将为您自动部署应用，您可以查看项目的部署日志。

步骤2：资源管理

在 Serverless 应用 页面，单击目标应用进入应用详情页，查看项目部署后输出的基本信息、部署日志等，方便您轻松实现项目的管理运维。

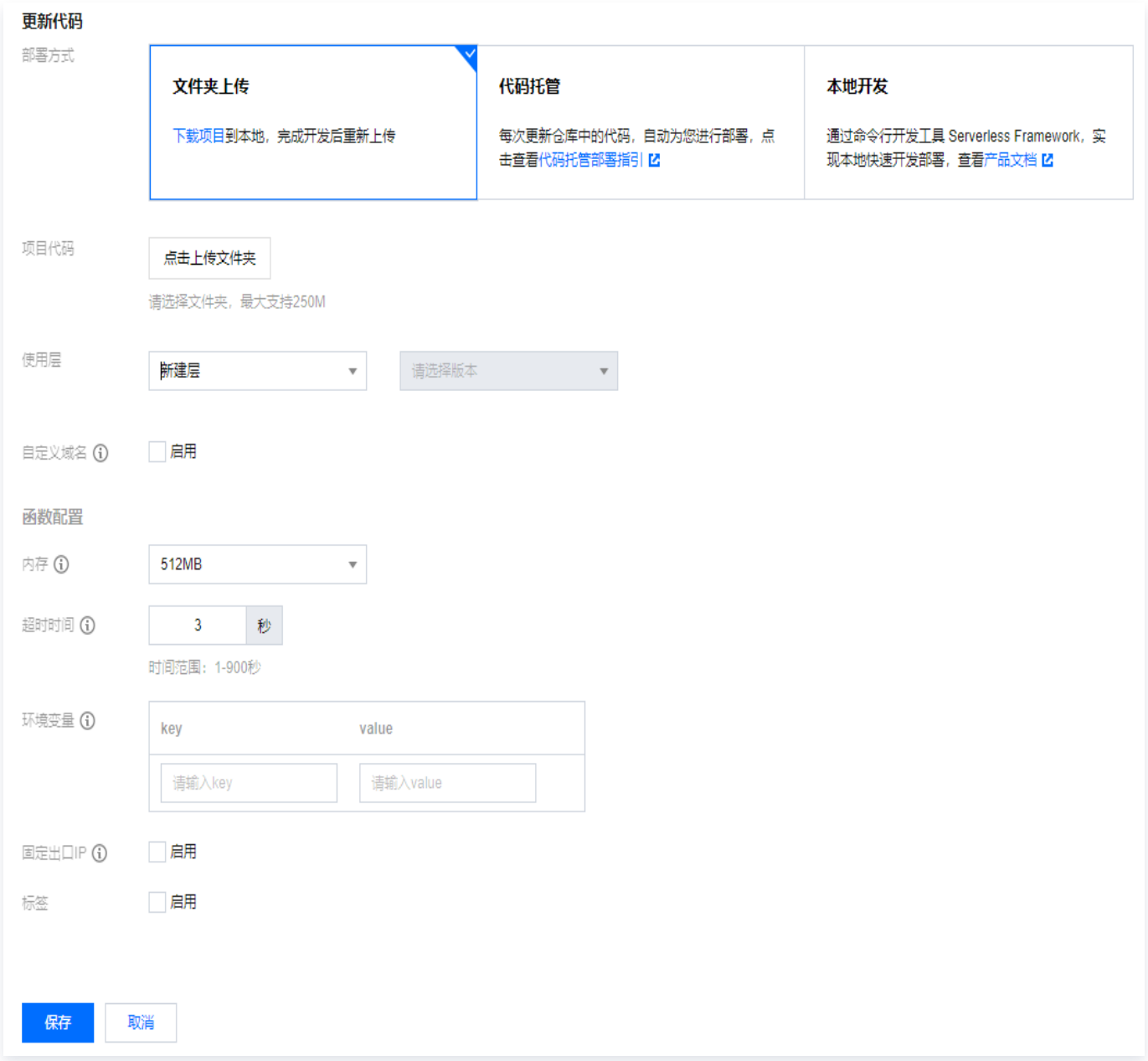


步骤3：开发部署

在应用详情页顶部，选择**开发部署 > 更新应用**，您可以轻松地实现应用的配置修改与二次部署上传，支持**文件夹上传、代码托管、本地开发**三种方式。

文件夹上传

您可以通过上传文件夹的方式直接导入本地项目，对于 Node.js 框架，Serverless Cloud Framework 将自动为您创建层，并将依赖包 node_modules 传入层中完成部署。



代码托管

目前支持 GitHub、GitLab、Gitee 和 CODING 的代码仓库地址，也支持公开的**自定义代码库**，您可以通过选择应用的触发方式，完成应用的自动更新，详情请参见 项目触发方式管理 。

更新代码
部署方式
文件夹上传
下载项目到本地，完成开发后重新上传
代码托管
每次更新仓库中的代码，自动为您进行部署，点击查看代码托管部署指引
本地开发
通过命令行开发工具 Serverless Framework，实现本地快速开发部署，查看产品文档
代码源
Github
Gitlab
Gitee
CODING
自定义仓库
选择当前代码源内项目及仓库，如当前账号尚未授权该代码源，可授权代码源
代码仓库
请选择仓库地址
请选择代码分支
触发规则
自动触发构建
手动触发构建
选择应用自动更新的触发规则，查看详细触发规则
使用层
新建层
请选择版本
自定义域名
启用
函数配置
内存
512MB
超时时间
3
秒
时间范围：1-900秒
环境变量
key
value
请输入key
请输入value
固定出口IP
启用
标签
启用
保存
取消

本地开发

您可以下载代码到本地，安装 Serverless Cloud Framework 并进行本地开发，构建项目并部署。