

云应用 独立开发者指南



腾讯云

【 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

独立开发者指南

应用管理

制作安装包

独立开发者指南

应用管理

最近更新时间：2025-02-27 10:27:32

本文将介绍独立开发者自主制作应用并上架的整体流程。

术语及解释

术语	解释说明
云应用	云应用是一个选购、安装和管理应用的平台。客户在云应用平台能够自助采购软件和依赖的云资源，并能自动完成安装和部署。同时，还能通过看板形式管理应用。
应用	软件开发者可以在云应用平台上架应用，供客户选购。一个应用包括两部分，其一是软件部分，其二是软件所依赖的云资源部分。例如底层计算、存储、网络、数据库等。
安装包	安装包是开发者提交符合云应用标准的自动化部署交付物，整合了云资源编排、安装参数和输出声明等内容。用户通过云应用一键部署，即可自动完成资源创建、应用配置和进程启动，实现全流程自动化安装。
应用实例	客户在云应用平台安装一个应用，会在自己的账号下形成一个应用实例。
云资源	软件运行所需要的腾讯云资源，例如 TKE 容器集群。
独立开发者	在云应用平台制作/上架软件应用的提供方。

应用管理整体流程



一、创建应用

1. 进入 [开发者中心](#) > [我的应用](#)，单击**创建应用**。

云应用

应用实例

开发者中心

我的应用

安装包管理

开发者信息

我的应用

开发者上架应用的流程介绍 [查看操作指引 >](#)

1. 创建应用

设置应用名称生成应用

创建应用

全部 (30) 待上架 (12) 已上架 已下架

应用名称 安装包信息

简易 SPU_ pkg_

2. 填写应用名称（必填，限制50个字符）和应用简介（选填，限制200个字），单击确定即可完成创建。

创建应用

应用名称* 请输入应用名称，不超过50个字符 0/50

应用名称将在应用商店展示，客户可见

应用简介 请输入应用介绍内容 0/200

取消 确定

3. 创建成功后，系统将自动跳转到应用指引页面，您可按照分步指引说明进行操作。

二、安装包制作

您需要制作应用运行所需的安装包，并与应用完成关联：

- 若您未制作安装包，您可前往 [安装包管理](#) 进行制作。具体操作请参见 [制作安装包](#)。
- 若您已完成安装包制作，您可单击[立即关联](#)选择安装包。

说明:

仅支持选择有线上版本的安装包。

Step1: 制作安装包并关联 未关联

安装包指应用运行依赖的程序，开发者需要准备应用镜像、描述资源栈、描述服务编排等，并进行安装包的推送、测试、审核。

[立即关联](#)

[去制作安装包>](#)

Step1: 制作安装包并关联 未关联

安装包指应用运行依赖的程序，开发者需要准备应用镜像、描述资源栈、描述服务编排等，并进行安装包的推送、测试、审核。

关联安装包*

请选择安装包



[去制作安装包](#)

取消

确认

三、配置应用信息

应用信息包含基础信息、应用规格、计费信息和应用详情。

1. 基础信息

基础信息

应用名称*

安装包1111

0/50

应用名称将在应用商店展示，客户可见

应用简介*

请输入内容

0/200

应用图标*



选择图标

应用图片*

第一张为商品主图，一共可上传 10张 照片，鼠标拖拽图片可调整顺序



- 应用名称：显示创建应用时填写的名称，最多不超过50字，支持修改。
- 应用简介：显示创建应用时填写的简介，最多不超过200字，支持修改。
- 应用图标：正方形图标（64 × 64px），图形需简洁清晰，分辨率 > 150dpi。
- 应用图片：建议使用横向比例4 : 3左右的图片，分辨率 > 150dpi。最多可上传10张，首张为应用主图，可拖拽调整图片顺序，鼠标悬停可删除图片。

2. 应用规格

独立开发者暂不支持设置应用规格，规格默认为标准版。

应用规格

❗ 暂不支持设置应用规格，规格默认为标准版

版本 标准版

3. 计费方式

独立开发者暂不支持软件定价，软件默认免费，云资源费用按照软件实际消耗收取。

计费方式

应用价格 免费 ❗ 默认免费，暂不支持设置价格

云资源价格 根据应用实际消耗的费用收取

4. 应用详情

应用详情

应用详情

应用详情* 请输入文本

定价说明 自动生成

使用指南

服务支持

客户案例

自定义模板

+ 添加自定义模板 (1/3)

- 应用详情：必填，建议详细描述应用的定位、功能特性、适用客户范围等信息。

- 定价说明：系统自动生成，展示规格对应的价格。
- 其余模块信息开发者可根据实际情况填写，若内容为空，将不会在客户侧展示该模块。
- 最多支持自定义3个模块信息。

5. 提交审核

- 完善应用信息后，您需要将应用信息提交审核，审核通过才可上架应用。
- 审核期间不支持编辑应用信息；如需编辑，可通过**撤回审核**操作撤回后重新修改提交。
- 若云应用团队正在审核应用信息，则不支持撤回，建议开发者确认无误后再提交，避免等待过长时间。
- 您可通过**我的应用列表**中“审核状态”字段来了解审核进度，若审核不通过，系统将反馈驳回原因，建议您根据驳回原因修改后重新提交。

ⓘ 说明：

云应用平台将在2~3个工作日内完成审核处理，请开发者耐心等待。

四、上架应用

完成**安装包关联**和**应用信息审核**通过后，您可单击**立即上架**发布应用。仅 AI 类应用有机会在云应用市场曝光。

Step3: 应用上架

安装包和应用信息审核通过后可发布应用。发布后，您可将应用链接发送至客户进行安装，同时，您的AI类应用将有机会在应用市场曝光。

立即上架

我的应用列表操作项说明

云应用

应用实例

开发者中心

我的应用

安装包管理

开发者信息

我的应用 操作指引

开发者上架应用的流程介绍 [查看操作指引](#)

1. 创建应用

设置应用名称生成应用

2. 安装包制作

制作应用运行的安装包, 去制作

3. 配置应用信息

填写应用基本信息、介绍等

4. 上架应用

上架后客户可购买安装

[创建应用](#)

全部 (30)	待上架 (12)	已上架	已下架	审核中/审核不通过
应用名称	安装包信息	审核状态	上下架状态	操作
简影 SPU_	简影 pkg_	● 审核通过	● 已上架 访问应用	编辑 下架 获取应用链接 客户权限设置 更换安装包
科技 SPU_	科技 SPU_	● 审核通过	● 已上架 访问应用	编辑 下架 获取应用链接 客户权限设置 更换安装包
安装 SPU_	安装 SPU_	● 审核不通过	● 待上架	编辑
小店 SPU_	小店 SPU_	● 审核通过	● 已下架	编辑 上架
特卖 SPU_	特卖 SPU_	● 审核中	● 待上架	编辑
安装 SPU_	安装 SPU_	● 未提交审核	● 待上架	编辑
网久 SPU_	网久 SPU_	● 审核中	● 待上架	编辑

共 7 项 10 条 / 页 1 / 100 页

状态说明

支持快捷筛选查看“待上架”、“已上架”、“已下架”和“审核中/审核不通过”状态下的应用。

● 审核状态说明:

审核状态	说明
未提交审核	填写应用信息后保存至草稿, 暂未提交审核。
审核中	已提交审核, 并由云应用审核中。
审核不通过	应用信息填写有误, 系统将反馈审核驳回原因, 单击“审核不通过”状态即可查看驳回原因。
审核通过	应用信息填写无误, 开发者可上架应用。

● 上下架状态说明:

上下架状态	说明
待上架	应用未上架至云应用商城, 客户不可访问/安装/购买。
已上架	应用已上架至云应用商城, 客户可访问/安装/购买 (前提: 应用对所有客户开放权限)。

已下架	应用从云应用商城已下架，客户不可访问/安装/购买，已购买客户可正常使用。
-----	--------------------------------------

编辑应用信息

关键逻辑说明：

- 应用信息最多有两个版本，分为线上版和草稿版。
 - 线上版定义：已上架、上架后已下架的应用。
 - 草稿版定义：未上架的应用或者“已上架”/“已下架”的应用重新进行编辑后保存至草稿/提交审核但未出审核结果。
 - 草稿版审核通过后，将直接更新线上版应用信息。
 - 草稿版审核驳回后，不影响线上版本，伙伴可根据驳回建议修改后重新提交。
 - 多次对草稿版编辑将更新草稿版信息（始终只有一个草稿版）。

不同操作路径对应相关状态可参考下表：

操作路径	应用版本	审核状态	上下架状态	备注
创建应用	草稿版	未提交审核	待上架	-
创建应用->关联安装包	草稿版	未提交审核	待上架	-
创建应用->关联安装包->填写应用信息->保存至草稿	草稿版	未提交审核	待上架	-
创建应用->关联安装包->填写应用信息->提交审核	草稿版	审核中	待上架	-
提交审核->云应用审核通过	草稿版	审核通过	待上架	-
提交审核->云应用审核驳回	草稿版	审核不通过	待上架	-
应用审核通过->上架应用	线上版	审核通过	已上架	-
应用已上架->下架应用	线上版	审核通过	已下架	-
应用已上架->编辑应用基础信息->保存至草稿	线上版	审核通过	已上架	-
	草稿版	未提交审核	-	-
应用已上架->编辑应用基础信息->提交审核	线上版	审核通过	已上架	-

(同时有两个版本)	草稿版	审核中	-	-
应用已上架->编辑应用基础信息->提交审核->云应用审核通过	线上版	审核通过	已上架	草稿版审核通过将自动更新线上版应用信息
应用已上架->编辑应用基础信息->提交审核->云应用审核驳回	线上版	审核通过	已上架	草稿版审核驳回对线上版无影响
	草稿版	审核不通过	-	

查看应用信息

- 单击“应用名称”即可跳转到应用详情页面。
- 若应用信息同时存在线上版和草稿版，默认仅展示草稿版应用信息。

列表其他操作项

操作项	操作前置条件	说明
上架	应用审核通过且已关联安装包	上架后即可获取应用安装链接发送至客户进行安装。
下架	应用已上架	下架后客户无法访问应用链接。
访问应用	应用已上架	可通过“访问应用”查看应用的应用商城的展示效果。
获取应用链接	应用已上架	单击后链接将自动复制至剪贴板，可发送给目标客户。
更换安装包	应用已关联安装包	单击后回显当前关联的安装包，可调整为任意有线上版本的安装包。
客户权限设置	应用已上架且应用等级为 B/C	操作权限： <ul style="list-style-type: none"> • 所有用户可访问&安装（默认选项）。 • 所有用户可访问&部分用户可安装。 • 白名单用户可访问&部分用户可安装。 若涉及白名单用户，则需要输入用户 UIN。

制作安装包

最近更新时间：2025-02-27 14:37:52

术语及解释

术语	解释说明
安装包	安装包是伙伴提交符合云应用标准的自动化部署交付物，整合了云资源编排、安装参数和输出声明等内容。 用户通过云应用一键部署，即可自动完成资源创建、应用配置和进程启动，实现全流程自动化安装。

创建安装包

当前，安装包支持两种创建方式，说明如下：

	模板创建	普通创建
优势	简单、快捷，通过官方提供的固定模板创建。 只需输入少量模板参数，即可快速创建安装包并提交开发版。	灵活性高，可创建复杂应用。 可完全自定义各种类型的资源，以及资源的各种参数。
不足	灵活性不足。 仅支持部分参数自定义。	需要有一定开发基础。 需要安装 Node.js 环境和 CLI 开发者工具。

具体创建步骤如下：

模板快速创建

1. 在 [安装包管理](#) 单击**新建安装包**，选择**通过模板创建**，填写安装包名称、部分模板参数后即可完成创建。

新建安装包
✕

安装包名称 *

创建方式 普通创建 通过模板创建

选择模板 CVM通用模板

模板参数

选择CVM镜像 请选择

服务名	端口 ?	主端口	操作
<input style="width: 90%;" type="text" value="例：API服务"/>	<input style="width: 90%;" type="text" value="端口范围：1-65535，例：80"/>	<input type="checkbox"/>	删除
新增			

2. 安装包创建完成后，单击目标应用操作栏下的**版本管理**即可进入安装包版本管理界面。

安装包名称	安装包 ID	状态	线上版本	已关联商品	操作
测试	pkg-	● 已上线	0.0.1	1	安装包详情 制品管理 版本管理 删除
测试	pkg-	● 已上线	0.0.4	10	安装包详情 制品管理 版本管理 删除

普通创建

1. 在 **安装包管理** 单击**新建安装包**，填写安装包名称、选择**普通创建**后即可完成创建。

新建安装包
✕

安装包名称 *

创建方式 普通创建 通过模板创建

2. 安装包创建完成后，单击目标应用操作栏下的**版本管理**即可进入安装包版本管理界面。

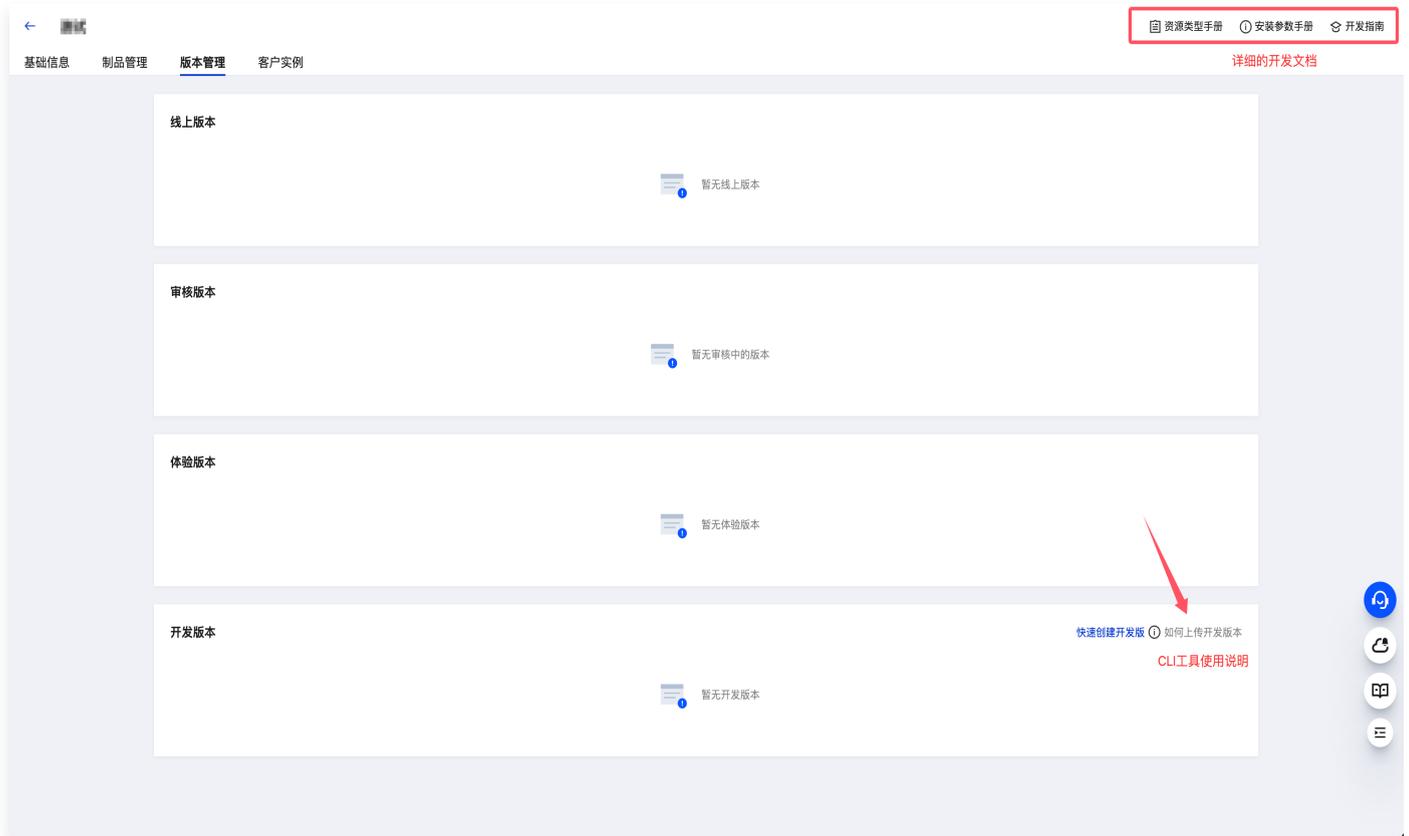
安装包名称	安装包 ID	状态	线上版本	已关联商品	操作
测试	pkg-	● 已上线	0.0.1	1	安装包详情 制品管理 版本管理 删除
测试	pkg-	● 已上线	0.0.4	10	安装包详情 制品管理 版本管理 删除

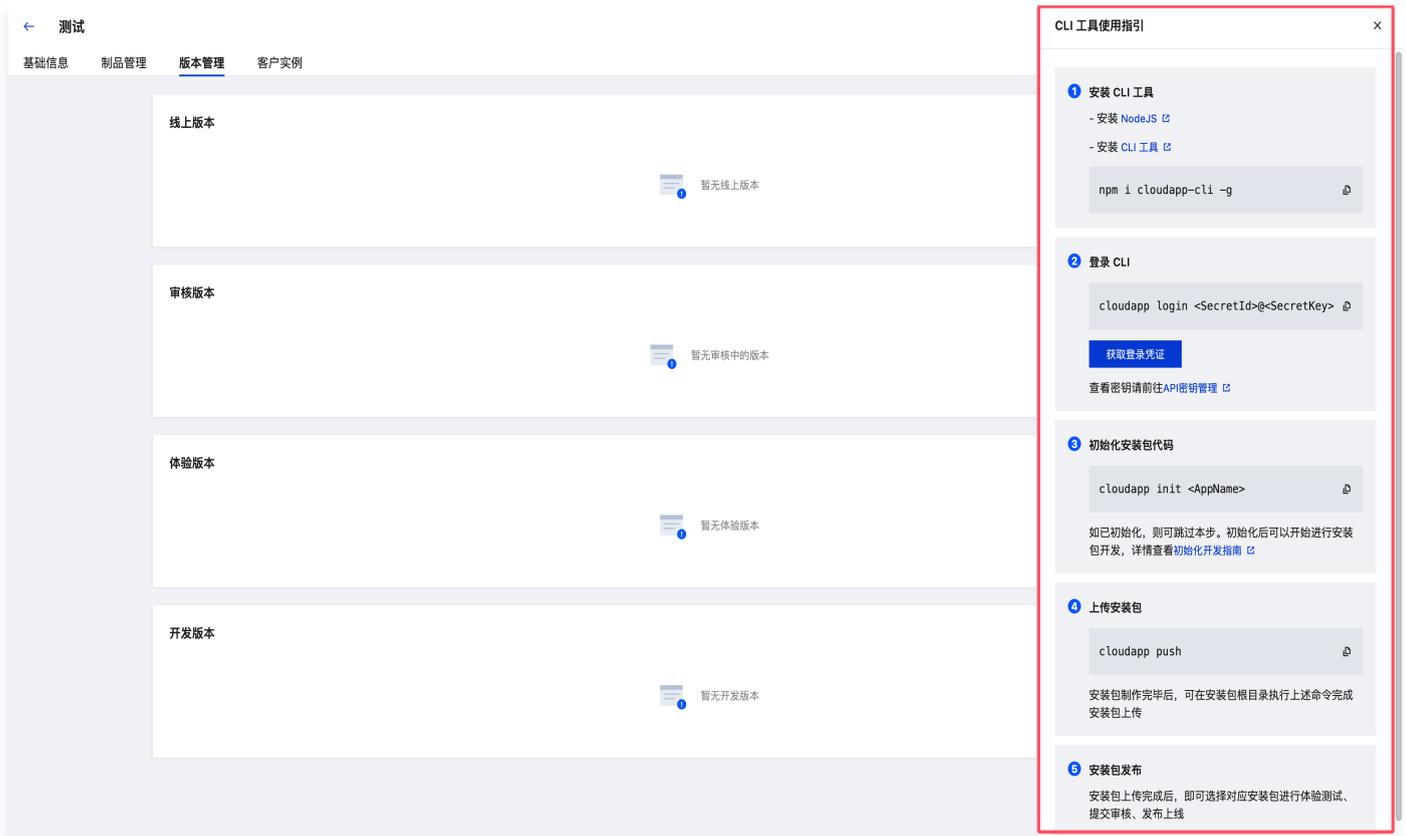
登录开发者工具（CLI）

1. 在版本管理界面中，单击[如何上传开发版本](#)，唤出工具安装和登录指引。

说明：

需要 Node.js 环境。云应用开发者工具基于 Node.js 开发，安装前请确认开发环境已安装 Node.js。如未安装，请前往 [Node.js 官网](#) 下载安装。





2. 根据指引安装好开发者工具后，使用 `cloudapp` 命令可以查看工具的使用说明：

```
> cloudapp
Usage: cloudapp [options] [command]

Options:
  -V, --version      output the version number
  -h, --help         display help for command

Commands:
  login              登记开发者身份
  init               初始化云应用项目
  docker            获取 Docker 仓库登录凭据
  push              提交一个新的应用包版本
  help [command]    display help for command
```

3. 安装好工具后，请使用 `cloudapp login` 命令在本地开发者中心登录，您需要使用开发者的 CAM 密钥进行登录。

```
cloudapp login $SECRET_ID@$SECRET_KEY
```

SecretId 和 SecretKey 为注册应用开发方主体身份时，提交的开发者信息中，对应的 CAM 密钥。登录成功会显示开发方和开发者名称。如果您忘记了密钥信息，可以到 [CAM 控制台](#) 查询。

安装包初始化（CLI 工具）

🚨 快速理解安装包：

安装包在开发者的本地体现为一个文件系统目录，一般建议采用项目根目录下的 `.cloudapp` 目录作为安装包目录。

推荐通过阅读 [最小安装包的 Demo 源码](#)，来理解安装包的组成。

使用 `cloudapp init` 命令可以生成一个示例安装包：

```
# 初始化本地项目，可通过 -t 指定初始模板，支持的模板参考 Github 模板清单（默认模板：  
CVM实例 + 公网IP）
```

```
cloudapp init hello-world
```

```
# 示例
```

```
# cloudapp init hello-world -t CT_001
```

🔔 技术预览版功能：

`cloudapp init` 命令当前为预览版功能，后续可能会调整其用法和参数。当前 `cloudapp init` 只在本地下载模板代码，更多的模板代码可以在云应用的 [GitHub 仓库](#) 中查看。

安装包组成

安装包由开发方交付的内容组成（分为 CVM 部署、容器部署两种模式），具体差异见下表：

CVM 部署

🚨 说明：

如果您的应用采用的是「容器部署」，请切换到「容器部署」查看需要交付的内容。

内容	位置	说明
安装包元信息	<code>.cloudapp/package.yaml</code>	安装包的元信息描述文件。 包含安装包 ID、应用安装参数、CAM 权限策略等信息声明。
资源和变量声明	<code>.cloudapp/infrastructure/**/.tf</code>	资源和变量声明，采取 Terraform 语法。 资源采用资源块的方式声明；变量为资源部署时可使用的变量，使用前需声明。

CVM 镜像
(制品)

云应用制品托管中心

云应用提供 CVM 镜像托管

容器部署

! 说明:

如果您的应用采用的是「CVM 部署」，请切换到「CVM 部署」查看需要交付的内容。

内容	位置	说明
安装包元信息	.cloudapp/package.yaml	安装包的元信息描述文件。包含安装包 ID、应用安装参数、云 API 权限等信息声明。
资源和变量声明	.cloudapp/infrastructure/**/*.tf	资源和变量声明，采取 Terraform 语法。资源采用资源块的方式声明；变量为资源部署时可使用的变量，使用前需声明。
容器镜像 (制品)	云应用 TCR 仓库	云应用提供 TCR 仓库为需要托管容器镜像的应用提供镜像托管服务。
服务编排	.cloudapp/software/charts/**/*.yaml	服务编排，采用 Helm Chart 编排规范。编排容器集群中的 StatefulSet、Service、Job、Secret 等内容。

安装包元信息

安装包元信息在 `package.yaml` 中声明，包含安装包 ID、应用安装参数、云 API 权限等信息声明。

```
# 安装包 ID, 对应开发者中心中的应用 ID
id: pkg-81ve2gn1

# 应用参数, 定义允许客户在安装过程中设置的参数, 例如所属网络、应用规格、付费方式等
args:
  - name: app_target
    label: 目标网络
    widget: subnet-select

# 应用权限声明 (可选), 如果应用内需调用云 API, 需这里进行声明
```

```
role:
  policy:
    version: "2.0"
    statement:
      - action:
          - cloudapp:VerifyLicense
        resource: "*"
        effect: allow
```

⚠ 注意:

请提供正确的安装包 ID。开发者工具在推送应用的时候，会根据 package.yaml 中记录的安装包 ID 来鉴权并推送，如果应用未登记，或者当前登录的开发者，不是对应的应用开发者，将推送失败。

资源和变量声明

资源声明

安装包中的资源声明，是安装包的核心内容。云应用借用 [Terraform 语法](#) 来声明软件依赖的云资源。

```
resource "tencentcloud_eks_cluster" "eks" {
  availability_zone = var.app_target.subnet.zone
  k8s_version       = "1.20.6"
  vpc_id            = var.app_target.vpc.id
  subnet_ids       = [var.app_target.subnet.id]
  service_subnet_id = var.app_target.subnet.id
}
```

上面就是一个资源块的声明语法，`resource` 表示声明资源块，`"tencentcloud_eks_cluster"` 表示资源类型，`"eks"` 为资源名称。

📌 说明:

安装包支持声明的资源大致分为两类：

- 腾讯云 IaaS / SaaS 资源，资源类型前缀为 `tencentcloud`。
- 云应用封装的资源，资源前缀 `cloudapp`。

支持的资源类型及资源参数，请参考 [资源类型手册](#)。

变量声明

上面声明的 EKS 资源，其关联的 VPC、子网等信息，是从 **变量** 中获取的。安装包中，所有用到的变量，都需要进行声明才能使用：

使用 `variable` 关键字声明变量，`app_target` 为「变量名」，通过 `var.app_target` 可使用该变量

```
variable "app_target" {
  type = object({
    region      = string
    region_id   = string
    vpc = object({
      id          = string
      cidr_block = string
    })
    subnet = object({
      id   = string
      zone = string
    })
  })
}
```

跟资源块类似，我们使用变量块 `variable` 声明一个变量，上面的声明中 `app_target` 为变量名，在资源块中使用 `var.app_target` 即可引用。

变量可用于获取用户的安装参数，也可以用于获取系统的安装参数，详情请参考 [安装参数](#)、[变量和输出](#) 文档说明。

镜像托管

CVM 镜像托管

CVM 镜像

❗ 说明：

当您制作好 CVM 镜像后，需要将镜像托管到云应用制品管理中心，以便于在用户安装时，云应用自动将对应镜像共享给用户。

- 在云应用开发者中，选择应用的制品管理。

应用名称	应用 ID	状态	线上版本	已关联商品	操作
测试建的应用	pkg-ld- 	已上线	0.0.2	1	应用详情 制品管理 版本管理 删除

- 在制品管理中心选择当前应用需要托管的 CVM 镜像。

云应用

- 应用实例
- 开发者中心
- 数据看板

← 模板应用-CVM-云API

应用信息 **制品管理** 版本管理 客户实例

CVM 镜像

托管镜像

镜像 ID	可使用地域	镜像描述	创建时间	操作
暂无数据				

容器镜像

推送地址: cloudapp.tencentcloudcr.com/pkg-[name]:[tag]

登录凭证 ⓘ 获取凭证

名称	仓库地址	创建时间	操作
node-server	cloudapp.tencentcloudcr.com/pkg- [name] /node-server	2024-07-31 16:48:47	查看版本列表

托管镜像 ×

所属地域: ▼

镜像 ID: [name] (云应用-Node-云API调用)"/> ▼

镜像描述:

导入镜像

- 根据应用允许安装的地域，调整镜像可用地域。

CVM 镜像

托管镜像

镜像 ID	可使用地域	镜像描述	创建时间	操作
img-8	广州	共享镜像演示	2024-08-14 17:28:47	调整可用地域 查看镜像详情

容器镜像托管

容器镜像

❗ 说明：

云应用提供 TCR 仓库为需要托管容器镜像的应用提供镜像托管服务。该仓库客户在安装是从内网拉取，无需外网支持。

- 使用 `cloudapp docker` 命令获得仓库登录命令。

```
> cloudapp docker
docker login cloudapp-tcr.tencentcloudcr.com --username 100026272331 --password eyJhbGciOiJSUzI1NiIsImtpZCI6Iik5TT086WlFSQzo2VEZC0kxJSkM6WVhBTzplMjdQ0k9QVE86WFBaTdpRRFM30ldGRTQ6RVVENjpIWE5UIn0.eyJvd25lclVpbiI6IjQzOD
E2NzYxMyIsIm9wZXJhdG9yVWluIjoiYjg3Njk3NTMsImldhdCI6MTY3ODc2OTc1M30.CzTKkD7Rcwyp7SqUAFvpK8cJdZNKEQmB5KTWGLQVi-6FxmP_aZF0BBHv2YKHwMu1TeMThwM5Uv5ps2P9KhRlTeMrmM5yaaFk
fPmS22L0GEVw0Gv8PQ6CvCFxFCw6kDBaTsjne-3uQhpcm76J963
lLfD5RFRccIUVLgIRaLiMH3QxTQsIQc8boCCpbczfU9c4cMGoYpHwWSiaaxUmys_4CEf70zfW_ZJ5XsZMV3BV2g0ad5Pug0ZJfDoSsT23z
mHUPQy4dgsN6h5Y0B1V1A3vNlv7-ANug
```

- 使用 `cloudapp docker --url` 获得仓库推送地址。

```
> cloudapp docker --url
cloudapp-tcr.tencentcloudcr.com/techirdliu/[image_name]:[tag]
```

- 使用 `cloudapp docker --auth` 单独获得 Docker 授权文件，适用于 CI 构建的场景。

```
> cloudapp docker --auth
{
  "auths": {
    "cloudapp-tcr.tencentcloudcr.com": {
      "auth": "MTAwMDI2MjcyMzMxOmV5SmhiR2NpT2lKU1V6STF0aUlzSW10cFpDSTZJazVUVDA4NldsRlNRem8yVkVa
Q09reEpTa002V1ZoQlR6cExNamRRT2s5UVZF0DZXrkJhVERwUljGTTNPbGRHUlRRNlJWVkw0anBJV0U1VUluMC5leUp2ZDI1bGNsV
nBiaUk2SWpRek9ER[REDACTED].pd2laWgh3SWpveE5qYzR0em
N6TmPvd0xDSnVZbVlpT2pFMk56ZzN0ekF3TlRbc0lTbGhkQ0k2TVRZM09EYzNNREExTUgwLlVLQXJwWDY3YTRXdDNrN2VPRtg2c1Z
rTEcyYnJjUFl1TGv[REDACTED]UtsUXd4Tjh0eUk4c20xYXh
VUlDVUkzRk5UZlZQVXlsekp0SjJTTTBfcVdSajQ5c0d6Q2NHTGFoNzNuS0dxwXIwTDRERVVSNOZ0ai1IQUhZUdd3cDNBeWtFTzdIQ
WlseHE5ak9tNGhaQU0wbVlVETzJKQXZMS2g4cHdybVJiVWlqLVBiamRlJUTVRck9wcXZtYXgwNjRqdDlkMmhCUllIVUg5ZXcyeFJwa0
NzM0FIRW9heDhfeVllRm5FZzZpVmVMZC00MTMxOVo2QUppc1RsX1hjak04RS1IZ0ptMVBWYnJfbTJkeThRUjleEdHJuU1k2dzVkNmJ
oZFdkREFqOHp4MzgtUWxiQQ=="
    }
  }
}
```

Docker 仓库登录后，就可以将镜像推送到托管的仓库中了。

🔔 仓库票据有效期和权限

cloudapp docker 命令拿到的仓库票据，只能用于应用方推送软件容器镜像。该票据有效期 2 个小时，所以不应该存储下来使用，建议 CI 流程构建完之后，再实时通过 cloudapp docker 命令获取票据。

客户安装时使用的安装票据是只读的，而 cloudapp docker 拿到的票据则可以操作应用开发方的软件镜像。

服务编排（容器部署）

通过声明 `cloudapp_helm_app` 类型的资源，可以声明容器服务编排。

```
resource "cloudapp_helm_app" "helm_charts" {
  cluster_id      = tencentcloud_eks_cluster.eks.id
  chart_src       = "../software/chart"
  chart_username  = var.cloudapp_repo_username
  chart_password  = var.cloudapp_repo_password

  chart_values = {
    SUBNET_ID = var.app_target.subnet.id
    IMAGE_CREDENTIALS = {
      REGISTRY = var.cloudapp_repo_server
      USERNAME = var.cloudapp_repo_username
      PASSWORD = var.cloudapp_repo_password
    }
  }
}
```

```
}  
}
```

需要指定编排的目标容器集群以及编排的 Helm Chart 源码位置（一般建议放在 `.cloudapp/software/chart` 上）。

通过 `chart_values` 将资源信息传递到 Helm Chart Values，这些信息在部署时，会传递给 Helm，因而编排的时候可以直接使用基础设施。参见上面的例子，将子网 ID 传给编排，在对 service 进行编排的时候，指定该子网 ID 从而生成内网的负载均衡类型。另一个常见的例子是，将 DB 实例化后的连接信息传入编排，从而让工作负载可以从环境变量中引用，或是生成到 configmap 中。

⚠ 注意：

将 TF 中声明的资源属性，传递到 Helm Chart Values，是个很重要的特性。利用该特性，让软件编排的时候可以直接编排实际部署的硬件资源，而无需二次设置。

进阶

继续阅读后续内容，学习进阶安装包特性的实现。

- [使用安装参数、变量和输出](#)
- [提供 HTTPS 访问](#)
- [调用云 API](#)
- [对接 License](#)

资源

前往 [Github 仓库](#) 查看更多示例源码。

下一步

完成安装包的生产 and 上传后，需进行安装包的推送、验证和上架，详情参考 [安装包管理](#)。