

Agent Runtime

实践教程



腾讯云

【 版权声明 】

©2013–2026 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

实践教程

在 AgentSandbox 沙箱中使用 AgentCFS

实践教程

在 AgentSandbox 沙箱中使用 AgentCFS

最近更新时间：2026-10-16 16:30:00

概述

AgentCFS 是腾讯云文件存储（CFS）面向 Agent 沙箱场景推出的共享存储能力，可为大量 Agent 沙箱提供可共享、安全隔离、休眠态可达的数据底座。本文介绍如何从零创建 AgentCFS 文件系统、创建并启动挂载了该文件系统的 Agent 沙箱，并在沙箱内验证存储挂载。

① 说明：

- 本文示例中的 `AKIDxxxx`、`SKxxxx`、`cfs-xxxxxxxx`、`subnet-xxx`、`sg-xxx`、`<UIN>`、`e2b-xxxxxxxx` 等均为占位符，请替换为您自己的真实值。
- AgentCFS 为保证沙箱间的数据隔离性，仅支持在沙箱启动时通过配置文件指定挂载。不支持通过云服务器、容器等方式挂载访问。如果需要通过非沙箱的方式访问 AgentCFS，请基于 S3 接口访问，具体请参见 [TurboS3接入点操作指南](#)。

前提条件

- 已 [注册腾讯云账号](#)，并具备文件存储（CFS）操作权限，授权方式请参见 [访问管理](#)。
- 已获取腾讯云 API 密钥（`SecretId` / `SecretKey`），用于调用创建模板、启动沙箱等 API，获取方式请参见 [访问密钥管理](#)。
- 若通过 E2B SDK 启动沙箱，需额外准备 `e2b_` 前缀的 API Key（与腾讯云 API 密钥不同，请勿混用），获取方式请参见 [创建与查看 API Key](#)。
- 文件系统、沙箱模板与沙箱实例的部署地域需保持一致。

操作步骤

步骤1：创建 AgentCFS 文件系统

- 登录 [文件存储控制台](#)，单击新建，进入 [新建文件系统](#) 页面。
- 配置基础信息与存储类型。

新建文件系统

基础配置

文件系统名称
请输入64位以内的中文、字母、数字、_、-。

地域
广州

可用区
广州六区

为了降低访问延时，建议文件系统与您的 CVM 在同一个区域。

存储类型

架构类型
全部 通用文件系统 并行文件系统 Agent文件系统

文件系统类型	容量	吞吐	最大IOPS	平均单流4K读时延	平均单流4K写时延	支持协议	扩容方式	支持操作系统	适用场景	计费模式	产品利弊
Agent CFS 标准型	20TiB - 100PiB	最大100GB/s	读200W / 写100W	0.2ms	3ms	POSIX/MPI/S3	手动扩容/基于扩容策略自动扩容	Linux	延迟不敏感型 Agent 共享存储空间 适用于周期任务、离线分析、报表审核等离线 agent 业务	按存储部署容量计费	存储部署容量： "配合低昂存储" !
Agent CFS 性能型	10TiB - 100PiB	最大100GB/s 并发性 最高2TiB/s	读1000W / 写300W	0.2ms	0.8ms	POSIX/MPI/S3	手动扩容/基于扩容策略自动扩容	Linux	延迟敏感型 Agent 共享存储空间 适用于 VibeCoding、在线办公 Agent、客服助手等在线 agent 业务	按存储部署容量计费	存储部署容量： "配合低昂存储"

*假设80%数据存储在低昂存储中

已选择Agent CFS 标准型 (容量20TiB - 100PiB; 吞吐最大100GB/s; IOPS读200W / 写100W; 4K读时延0.2ms; 4K写时延3ms)

存储量
20 250 500 1000 10000 10 + TiB
Turbo标准型起步容量为10TiB

自动扩容策略
为文件系统配置自动扩容策略 开启
自动扩容策略可在达到容量阈值后，自动进行扩容，降低容量超额风险

其他配置
加密 开启
标签 标签键 标签值
+ 添加标签 管理标签 历史记录

费用
(实际消费以使用情况为准，此数据仅供参考)

立即创建

配置项	说明
文件系统名称	输入64位以内的名称，支持中文、字母、数字、_、-。
地域 / 可用区	选择与沙箱一致的地域和可用区。
架构类型	选择 Agent 文件系统。
存储类型	根据业务时延要求选择 Agent CFS 标准型（延迟不敏感的共享空间，适用周期任务、离线分析等）或 Agent CFS 性能型（延迟敏感场景，适用 VibeCoding、在线办公 Agent、客服助手等）。
存储量	按需设置起步容量：AgentCFS 标准型起步10TiB，AgentCFS 性能型起步20TiB。
自动扩容策略	可选。开启后达到容量阈值自动扩容，降低容量超额风险。
加密 / 标签	按需开启加密、添加标签。

3. 确认费用后单击**立即创建**。创建完成后，在文件系统列表中记录**文件系统 ID**（形如 `cfs-xxxxxxxx` ），后续创建沙箱模板挂载存储时需要使用。

步骤2：创建沙箱模板

创建沙箱工具（模板）时挂载步骤1创建的 AgentCFS 文件系统。您可以通过控制台或 API 两种方式创建。

版权所有：腾讯云计算（北京）有限责任公司

第5 共14页

方式一：控制台创建

1. 登录 [Agent Runtime 控制台](#)，进入 **Agent 沙箱 > 沙箱** 页面，单击**新建沙箱工具**。

沙箱 (原自定义沙箱)

Agent 沙箱服务商业化上线，自月6起！即日起至8月31日，限时优惠火热进行中，加速业务 Agent 应用落地，欢迎试用。

新建沙箱工具 批量编辑沙箱

多个沙箱标签页同时展示

☐	ID/名称	状态	类型	生命周期类型	镜像	网络策略	腾讯云服务器	创建时间	更新时间
☐	sdb- cfs-sandbox-fl	可用	自定义沙箱	常驻沙箱	ccr.ccs.tencentyun	VPC网络	☒	2026-06-24 12:01:58	2026-06-24 12:02:13
☐	sdb- cfs-sandbox	可用	自定义沙箱	限时沙箱	ccr.ccs.tencentyun	VPC网络	☒	2026-06-23 21:51:17	2026-06-23 21:51:33
☐	sdb- custom-2	可用	自定义沙箱	限时沙箱	ccr.ccs.tencentyun	VPC网络	☒	2026-06-23 19:30:31	2026-06-23 22:08:15

共 3 条

2. 配置沙箱工具信息。

创建自定义沙箱

基本信息

工具名称 *

test

最长50个字符，只能包含字母、数字及分隔符["-"], 且必须以字母开头

生命周期 *

限时沙箱

到期自动释放。适合短时 Agent、工具调用

常驻沙箱

持续运行至手动释放。适合OpenClaw等有状态场景

5分钟

后自动释放

更多 >

描述

资源与健康

资源规格 *

CPU 2 核 内存 4 Gi 系统盘大小 不设置 (使用... Gi)

健康检查 *

☒ 已使用推荐配置 查看配置项

更多 >

持久化存储配置-b...

持久化存储配置-e...

+ 添加存储配置

存储类型

COS CFS AgentBucket 镜像

CFS

名称

test

请输入名称,该名称将作为后续启动沙箱配置的索引

文件系统

沙箱agent测试 (cfs-)

CFS 路径

/

CFS 内的挂载路径, 默认为根目录 /

挂载路径

/mnt

以 / 开头的绝对路径, 不能为根路径 /, 不能以 / 结尾, 不能包含 .. 或 //

读写权限

读写 只读

请选择读写权限

沙箱支持对接多种存储类型作为持久化存储，可在沙箱工具端统一挂载同一存储路径，确保工具下所有沙箱实例共享数据；同时也支持按沙箱实例进行路径隔离，满足多实例并发运行下的数据安全与独立性需求,可参考[操作文档](#)

镜像与启动

镜像

ccr.ccs.tencentyun.com/cfs-sandbox

选择镜像

当前镜像来源支持容器镜像服务企业版、个人版。

镜像版本 (Tag)

不选默认为 latest

选择镜像版本

启动命令 *

/usr/bin

确定

取消

一键预检

配置项	说明
基本信息	<ul style="list-style-type: none"> 填写工具名称：最长50字符，仅支持字母、数字、分隔符 <code>-</code>，且以字母开头。 选择生命周期：选择限时沙箱或常驻沙箱。
资源与健康	<ul style="list-style-type: none"> 设置 CPU、内存等资源规格。 健康检查可使用推荐配置。 存储配置：单击添加存储配置，存储类型选择 CFS，填写以下信息： <ul style="list-style-type: none"> 存储类型：选择 CFS。

	<ul style="list-style-type: none"> ○ 名称：存储配置名称，将作为后续启动沙箱配置的索引。 ○ 文件系统：选择 步骤1 创建的 AgentCFS 文件系统（ <code>cfs-xxxxxxxx</code> ）。 ○ CFS 路径：CFS 内的挂载源路径，默认根目录 <code>/</code>。 ○ 挂载路径：以 <code>/</code> 开头的绝对路径，不能为根路径 <code>/</code>，不能以 <code>/</code> 结尾（如 <code>/workspace/data</code>）。 ○ 读写权限：选择读写或只读。
镜像与启动	填写容器镜像地址、镜像版本（Tag，不填默认 <code>latest</code> ）与启动命令。

3. 单击**确定**完成创建。

方式二：API 创建

调用 `CreateSandboxTool` 接口创建沙箱模板。与标准模板相比，仅需在 `StorageMounts` 中配置 CFS 挂载：

- `StorageSource.Cfs.FileSystemId`：步骤一创建的文件系统 ID。
- `StorageSource.Cfs.Path`：CFS 内的挂载源路径，该路径必须在文件系统内已存在。
- `MountPath`：挂载到沙箱内的目标路径。

```
import json
from tencentcloud.common import credential
from tencentcloud.common.profile.client_profile import ClientProfile
from tencentcloud.common.profile.http_profile import HttpProfile
from tencentcloud.agv20250920 import ags_client, models

# 1) 凭证（使用腾讯云 API 密钥）
cred = credential.Credential("AKIDxxxx", "SKxxxx")
http = HttpProfile(); http.endpoint = "ags.tencentcloudapi.com"
cp = ClientProfile(); cp.httpProfile = http
client = ags_client.AgsClient(cred, "ap-beijing", cp)

# 2) 组装请求
req = models.CreateSandboxToolRequest()
req.from_json_string(json.dumps({
    "ToolName": "cfs-sandbox-demo",
    "ToolType": "custom",
    "Persistent": True,
```

```
"RoleArn": "qcs::cam::uin/<UIN>:roleName/<ROLE>",
"CustomConfiguration": {
    "Image": "ccr.ccs.tencentyun.com/qcloud/centos:latest",
    "ImageRegistryType": "personal",
    "Command": ["/envd"],
    "Ports": [{ "Name": "envd", "Port": 49983, "Protocol":
"TCP" }],
    "Resources": { "CPU": "2", "Memory": "4Gi" },
    "Probe": {
        "HttpGet": { "Path": "/health", "Port": 49983, "Scheme":
"HTTP" },
        "ReadyTimeoutMs": 30000, "ProbeTimeoutMs": 2000,
        "ProbePeriodMs": 1000, "SuccessThreshold": 1,
"FailureThreshold": 30
    }
},
"NetworkConfiguration": {
    "NetworkMode": "VPC",
    "VpcConfig": { "SubnetIds": ["subnet-xxx"],
"SecurityGroupIds": ["sg-xxx"] }
},
"StorageMounts": [
    {
        "Name": "brucetest",
        "StorageSource": { "Cfs": { "FileSystemId": "cfs-
xxxxxxxxxx", "Path": "/" } },
        "MountPath": "/workspace/data"
    }
]
}))

# 3) 调用, 拿到 ToolId
resp = client.CreateSandboxTool(req)
print(resp.to_json_string()) # {"ToolId": "sdt-xxxxxxxx",
"RequestId": "..."}

```

① 说明:

记录返回的 `ToolId` 与 `ToolName`，下一步启动沙箱时需指定带 CFS 挂载的 `ToolName`。

步骤3：启动沙箱

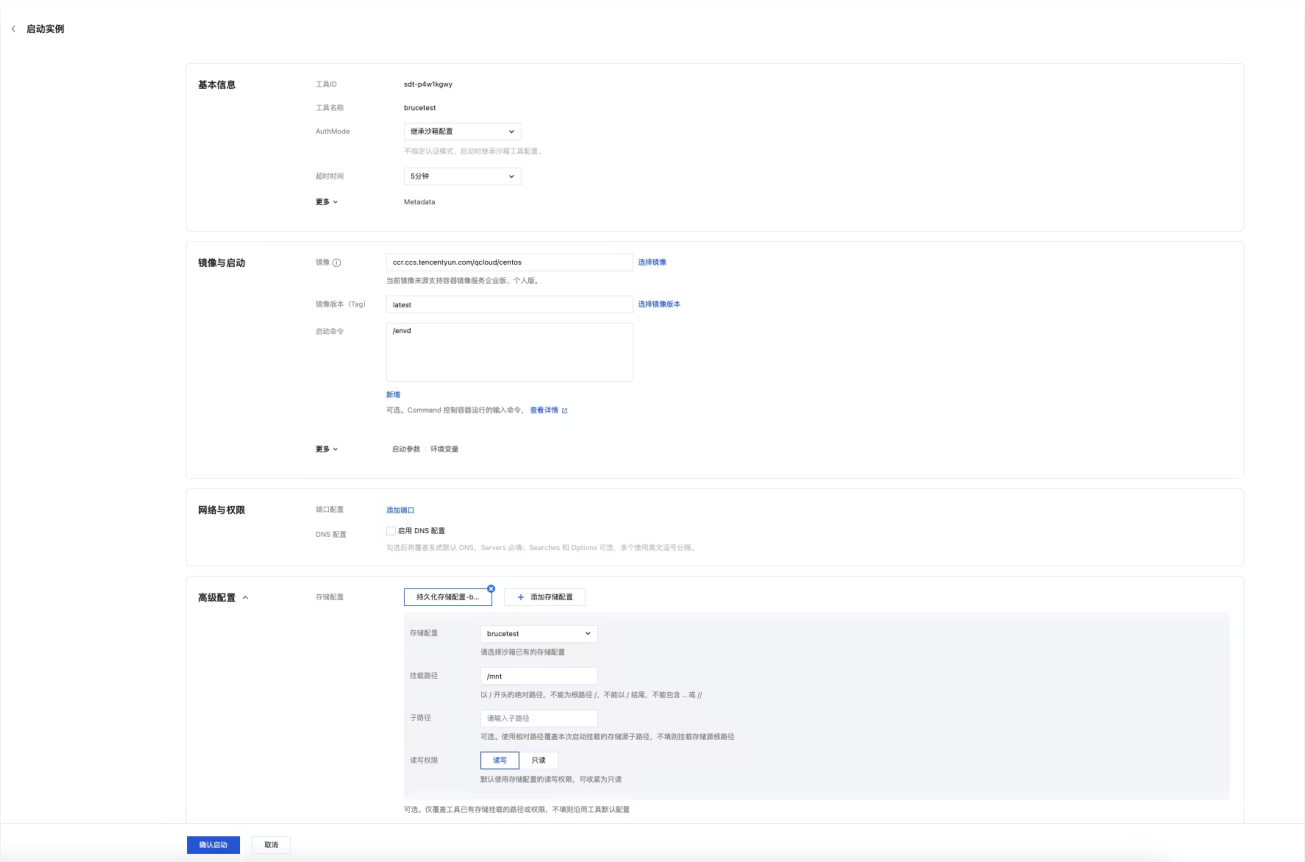
基于步骤2创建的模板启动沙箱实例。您可以通过控制台或 API 两种方式启动。

方式一：控制台启动

1. 在沙箱工具详情页的实例列表页签，单击启动实例。



2. 配置启动参数。



配置项	说明
基本信息	确认工具 ID / 名称，设置 AuthMode 与超时时间。
镜像与启动	确认镜像、版本与启动命令。
网络与权限	<ul style="list-style-type: none">端口配置：放开指定端口的网络通信。DNS 配置：支持配置指定 DNS 服务的地址，覆盖沙箱默认的 DNS 配置。

高级配置 > 存储配置

选择 **步骤2** 配置的存储，可覆盖挂载路径、子路径（可选，用于挂载到文件系统内的子目录，需保证该子目录已存在）与读写权限。

3. 单击确认启动。**方式二：E2B SDK 启动**

E2B SDK 通用、轻量，适合大多数 Agent 运行时集成场景。

```
import os, json
from e2b_code_interpreter import Sandbox

# 1) 环境变量（按 Tool 所在地域填写 domain，使用 e2b_ 前缀的 key）
os.environ["E2B_DOMAIN"] = "ap-beijing.tencenttags.com"
os.environ["E2B_API_KEY"] = "e2b_xxxxxxxx"

# 2) template = 步骤二创建的带 CFS 挂载的 ToolName
sbx = Sandbox.create(
    template="cfs-sandbox-demo",
    timeout=3600, # 沙箱存活秒数，到期自动回收
    metadata={
        "x-mounts": json.dumps([
            {
                "name": "brucetest", # 必需：匹配工具配置的
                "mountPath": "/workspace/data", # 可选：沙箱内挂载路径
                "readOnly": False, # 可选：读写权限
                "subPath": "user-123" # 可选：文件系统内子路
            }
        ])
    },
)

print("sandbox_id =", sbx.sandbox_id)

# 3) 在沙箱内执行命令，务必 user="root" 才能访问挂载点
res = sbx.commands.run(
    "df -hT /workspace/data; touch /workspace/data/hello && echo"
    "WRITE-OK",
```

```
user="root",
)
print(res.stdout)
```

⚠ 注意:

- `template` 需填写 Tool 的 **ToolName**, 而非 **ToolId** (`sdt-xxxxxxx`)。
- 连接已存在的持久化沙箱请使用 `Sandbox.connect("<sandbox_id>")`, 不要重复 `create`。

方式三: 腾讯云 SDK 启动

腾讯云 Python SDK 走 AGS 原生接口 `CreateSandbox`, 与创建模板使用同一套凭证/SDK, 适合已深度使用腾讯云 SDK 的平台侧集成。

```
import json
from tencentcloud.common import credential
from tencentcloud.common.profile.client_profile import ClientProfile
from tencentcloud.common.profile.http_profile import HttpProfile
from tencentcloud.ags.v20250920 import ags_client, models

# 1) 凭证 (与创建模板同一套腾讯云 API 密钥)
cred = credential.Credential("AKIDxxxx", "SKxxxx")
http = HttpProfile(); http.endpoint = "ags.tencentcloudapi.com"
cp = ClientProfile(); cp.httpProfile = http
client = ags_client.AgsClient(cred, "ap-beijing", cp)

# 2) 基于步骤二创建的 Tool 启动沙箱
req = models.CreateSandboxRequest()
req.from_json_string(json.dumps({
    "ToolName": "cfs-sandbox-demo",
    "TimeoutSeconds": 3600,
    "StorageMounts": [
        {
            "Name": "brucetest",
            "MountPath": "/workspace/data",
            "SubPath": "user-123",
            "ReadOnly": False
```

```
}  
]  
}))  
  
# 3) 调用, 拿到 SandboxId  
resp = client.CreateSandbox(req)  
print(resp.to_json_string()) # {"SandboxId": "...", "RequestId":  
"..."}
```

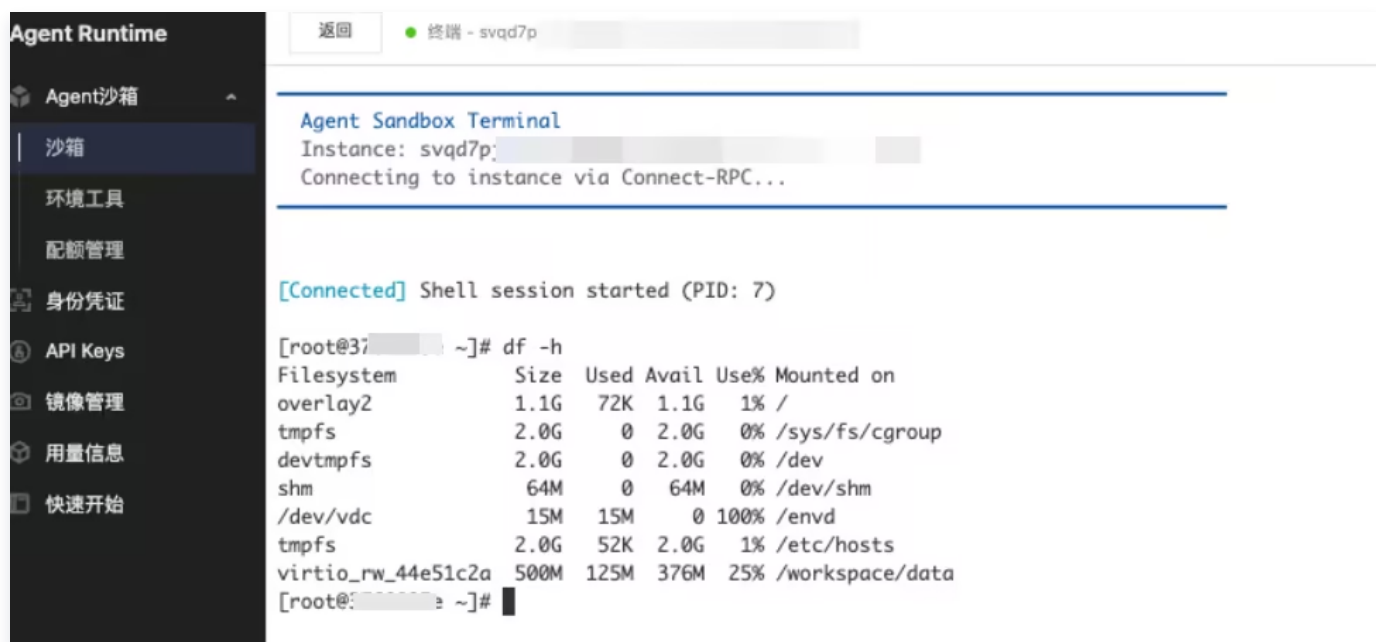
① 说明:

拿到 `SandboxId` 后, 后续在沙箱内执行命令仍走 E2B 通道 (`Sandbox.connect(SandboxId)`), 命令执行同样需 `user="root"` 才能访问挂载点。

步骤4: 登录沙箱验证存储

沙箱启动后, 可登录实例验证 AgentCFS 是否挂载成功。

1. 在实例列表中找到运行中的实例, 单击**登录实例**进入沙箱终端。
2. 执行 `df -h` 查看挂载情况, 可看到 AgentCFS 已挂载到指定路径 (如 `/workspace/data`)。



3. 在挂载路径下执行读写操作 (如 `touch /workspace/data/hello`), 确认存储可正常读写。

(可选) 通过 TurboS3 在休眠态访问数据

当沙箱处于休眠 / 暂停等非运行状态时, 若仍需查看、上传、下载对应数据, 可为文件系统开启 TurboS3接入点, 通过标准 S3接口访问数据, 使数据访问与沙箱生命周期解耦。

以下为在 VPC 内使用 boto3访问 TurboS3的最小连通性示例:

```
import boto3
from botocore.client import Config
import urllib3; urllib3.disable_warnings()

s3 = boto3.client(
    "s3",
    endpoint_url="https://<您的S3域名>", # 创建 S3 服务后获取的内网域名
    region_name="ap-beijing",
    aws_access_key_id="AKIDxxxx", # 腾讯云 API 密钥
    aws_secret_access_key="SKxxxx",
    config=Config(signature_version="s3v4",
                   s3={"addressing_style": "virtual"}), # 必须用 virtual 模式, hostpath 有域名风险
)

# 连通性自检与基本读写
print([b["Name"] for b in s3.list_buckets()["Buckets"]]) # 列出桶
s3.put_object(Bucket="cfs-xxxxxxx", Key="demo/hello.txt", Body=b"hello turbos3\n") # 写
print(s3.get_object(Bucket="cfs-xxxxxxx", Key="demo/hello.txt")["Body"].read()) # 读
```

⚠ 注意:

- TurboS3域名仅在您指定的 VPC 内生效，须在该 VPC 内的沙箱或主机上访问，并放通 TCP 443。
- boto3客户端务必使用 virtualhost (`addressing_style="virtual"`)

TurboS3 接入点的创建、完整接口支持范围与更多用法，请参见 [TurboS3接入点操作指南](#)、[TurboS3使用限制](#)。

常见问题

创建模板时提示挂载路径不存在

`StorageMounts` 中的 `Path` 必须是文件系统内已存在的路径。请先确认该目录已创建，或使用根目录 `/`。

接口鉴权失败

创建模板与启动沙箱（腾讯云 SDK 方式）使用腾讯云 API 密钥；通过 E2B SDK 启动沙箱使用 `e2b_` 前缀的 key，两者请勿混用。

沙箱内访问挂载点权限不足

在沙箱内执行命令时须指定 `user="root"`，例如 `sbx.commands.run(..., user="root")`。

沙箱内 pip 安装依赖超时

该 VPC 通常无公网出口，请改用内网源：`pip install -i http://mirrors.tencentyun.com/pypi/simple --trusted-host mirrors.tencentyun.com <包名>`。

SDK 执行长任务超时（`deadline_exceeded`）

对于耗时较长的命令，使用 `sbx.commands.run(..., background=True)` 后台执行，再轮询日志文件获取结果。