密钥管理

用户使用指南

天翼云科技有限公司

目 录

1. 产品简介

1.1. 产品定义

密钥管理服务（Key Management Service，KMS）是一站式密钥管理和数据加密服务平台，提供安全合规、可靠易用的资源托管及密码运算服务。同时与天翼云云硬盘、对象存储、弹性文件、关系型数据库 MYSQL 等云产品无缝集成，实现云上原生数据的加密保护。

业务组件

业务组件	说明	参考文档
密钥管理	密钥管理组件提供密钥安全托管存储、生命周期管理以及密码运算能力。您可以在自建应用程序中，通过 KMS 提供的云原生接口实现数据加解密、签名验签等运算，同时 KMS 已对接天翼云云硬盘、对象存储、弹性文件、关系数据库 MySQL 版，为云服务提供服务端加密能力。	密钥管理概述
证书管理	证书管理组件提供高可用、高安全的密钥和证书托管能力，您可以通过 KMS 提供的云原生接口实现签名验签运算。	证书管理概述

1.2. 产品优势

密钥管理服务（KMS）与传统的密钥管理设施相比具有安全合规、弹性高效、广泛集成以及稳定可用等优势。

安全合规

- 通过国家密码管理局安全性审查，符合国家密码行业标准（GM/T） 相关技术规范要求，获得由国家密码管理局商用密码检测中心颁发的《商用密码产品认证证书》。
- 采用由国家密码管理局批准的硬件密码设备，通过更高安全的保护机制确保密钥的保密性、完整性和可用性。

弹性高效

- 支持自动化开通，按需扩容，弹性灵活。
- 提供密码基础设施的完全托管，可轻松创建密钥等资源，并通过极简 API/SDK 实现应用的快速集成。

广泛集成

- 与云硬盘、对象存储、弹性文件、数据库等天翼云产品无缝集成，实现云上资源原生数据的加密保护。
- 云产品服务端加密功能可一键开启，加密过程透明无感知，用户体验极好。

稳定可靠

- 采用分布式部署，在每个资源池构建了冗余的密码计算能力，有效保证服务可靠性。
- 支持 VPC 内应用通过私密的链接通道访问 KMS 服务，保证数据安全性的同时，大幅度提高了访问效率，减少延时。

1.3. 功能特性

密钥生命周期管理

提供密钥全生命周期管理，包括密钥创建、自带密钥导入（BYOK）、启用/禁用、别名设置、轮转策略设置、版本设置、计划删除、取消删除等。

密钥算法

- 支持对称密钥算法类型为 AES_256、SM4；
- 支持非对称密钥算法类型为 RSA_2048、SM2。

硬件保护

通过部署托管密码机，采用由国家密码管理局批准的密码设备硬件，满足监管合规需求。

提供更高安全等级的硬件保护机制保护密钥，确保密钥的保密性、完整性和可用性。

密钥轮转

支持通过定期自动轮转或手动创建密钥版本，以加强密钥使用的安全性。

- 对于对称密钥，密钥版本可通过设置轮转策略，由系统根据轮转周期自动生成；
- 对于非对称密钥，可人工创建新的密钥版本。

密钥轮转或人工创建产生新的主版本后，KMS 不会删除或禁用非主版本，使得经非主版本加密的密文仍可以正常解密。

自带密钥导入

支持导入用户自带密钥。当用户希望使用自己的密钥材料时，可通过 KMS 管理控制台的导入密钥功能创建密钥材料为空的用户主密钥，并将自己的密钥材料导入该用户主密钥中。

别名管理

别名是用户主密钥的可选标识，同一个用户在一个地域中的别名具有唯一性。每个别名只能指向同地域的一个用户主密钥，但是每个用户主密钥可以绑定多个别名。

用户可通过控制台创建别名、删除别名，还可以通过 API 进行别名的创建、更新、删除等。

在线加密

在线加密是对称密钥加密的场景，适用于保护小型敏感数据（小于 6KB），如口令、证书、身份信息、后台配置文件等。通过密钥管理服务 KMS 的在线加密 API，使用用户主密钥（CMK）直接加密敏感数据信息，而非直接将明文存储，确保敏感数据安全。

信封加密

信封加密是对称密钥加密的场景，是一种应对海量数据的高性能加解密方案。这种技术不再使用用户主密钥（CMK）直接加密和解密数据，而是通过生成加密数据的数据密钥（DEK），将其封入信封中（即通过 CMK 加密）存储、传递和使用，由 KMS 确保数据密钥的随机性和安全性。

实际使用时，用户无需将大量业务数据上传至 KMS 服务端，直接通过离线的数据密钥在本地实现加解密，有效避免安全隐患，保证了业务加密性能的要求。

签名验签

数字签名技术是非对称加密算法的另一种典型应用。用户可在 KMS 中创建非对称用户主密钥（CMK），其由一对关联的公钥和私钥构成。公钥可以被分发给任何人，而私钥由 KMS 确保安全性，不提供任何接口导出非对称密钥的私钥。 使用者仅能通过接口调用私钥进行签名运算。

实际使用时，签名者将验签公钥分发给消息接收者，签名者使用签名私钥，对数据产生签名，签名者将数据以及签名传递给消息接收者，消息接收者获得数据和签名后，使用公钥针对数据验证签名的合法性。

非对称数据加解密

非对称密钥加密通信的过程类似于对称加密，区别在于需要使用公钥进行数据加密，使用私钥进行数据解密。由于 KMS 中用户私钥不支持导出，使用者仅能通过接口调用私钥进行数据解密。

实际使用时，信息接收者将加密公钥分发给信息传送者，信息传送者使用公钥对敏感信息进行加密保护，信息传送者将敏感信息的密文传递给信息接收者，信息接收者使用私钥将敏感信息的密文解密。

云产品服务端加密

与天翼云产品联动，提供对云硬盘、对象存储、弹性文件、数据库等产品中的数据进行服务端加密，保证云上数据的安全性。用户只需通过云产品控制台一键勾选 KMS 加密功能，加解密过程透明无感知。

完整性保护

提供基于国密算法的完整性保护能力，通过 SM3 算法计算 HMAC 值以进行对比验证，实现完整性校验。

1.4. 相关术语解释

- **对称密钥加密**：又称单密钥加密，即采用一个密钥进行信息的加密和解密。
- **非对称密钥加密**：非对称密钥由一对互相关联的公钥和私钥组成，其中的公钥可以被分发给任何人，而私钥必须被安全保护起来，只有受信任者可以使用。非对称密钥通常用于在信任程度不对等的系统之间，实现数字签名验签或者加密传递敏感信息。
- **用户主密钥（Customer Master Key，CMK）**：用户主密钥包括对称密钥及非对称密钥，主要用于加密保护数据密钥，也可直接用于加密少量的数据。用户可以通过 KMS 产品控制台或调用 KMS 的 CreateKey 接口创建一个用户主密钥。
- **默认主密钥（Default CMK）**：用户使用云产品加密功能时，由云产品触发 KMS 系统自动生成的并托管在用户账号下的服务密钥。
- **信封加密（Envelope Encryption）**：信封加密是类似数字信封技术的一种加密手段。这种技术将加密数据的数据密钥封入信封中存储、传递和使用，不再使用用户主密钥（CMK）直接加密和解密数据。当需要加密业务数据时，可以调用 KMS 的 GenerateDataKey 或 GenerateDataKeyWithoutPlaintext 接口生成一个对称密钥，同时使用指定的用户主密钥加密该对称密钥（被密封的信封保护）。

- **数据加密密钥（Data Encryption Key，DEK）**：信封加密技术中用于加密业务数据的密钥，受用户主密钥 CMK 加密保护。
- **硬件安全模块（Hardware Security Module，HSM）**：硬件安全模块也称为密码机，是一种执行密码运算、安全生成和存储密钥的硬件设备。KMS 提供的托管密码机可以满足监管机构的检测认证要求，为用户在 KMS 托管的密钥提供更高的安全等级保证。
- **密钥导入（Bring Your Own Key，BYOK）**：指用户可以自行导入密钥材料至用户主密钥中，KMS 不会为创建的用户主密钥（CMK）生成密钥材料。
- **应用接入点：**是一种身份验证和访问控制机制，当用户 VPC 内的自建应用需要访问 KMS 服务时，需要创建应用接入点实现私网通道打通，同时在应用接入点内完成访问权限策略配置以及身份凭证的生成。

1.5. 应用场景

密钥管理服务 KMS（Key Management Service）具有广泛的应用场景，以下时 KMS 常见的应用场景。

场景一：敏感数据加密

通过调用密钥管理服务（KMS）的密码运算 API 实现数据的在线运算，直接使用用户主密钥进行数据的加解密。

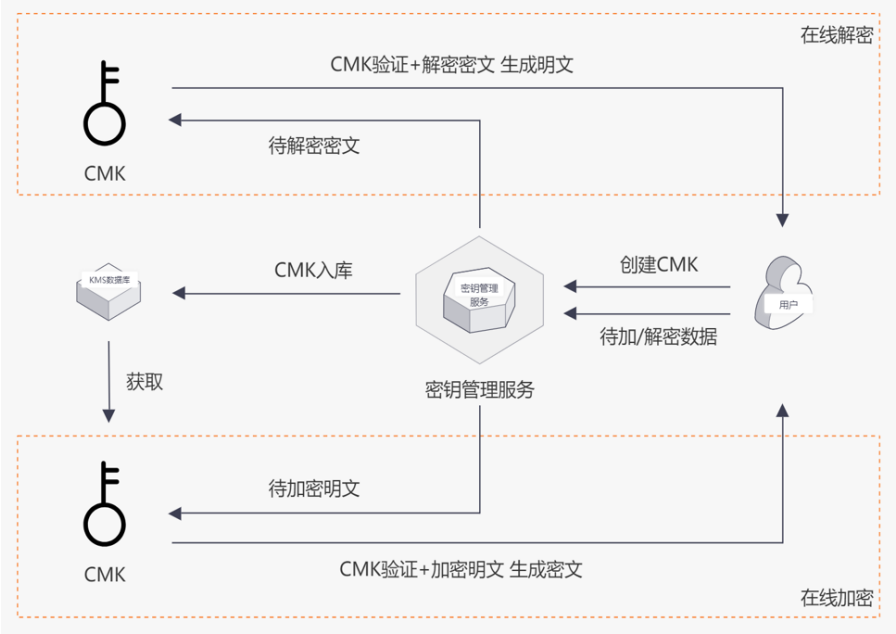
场景特点

用于少量数据（例如：口令、证书、配置文件等）的加密保护，有效避免敏感信息泄露。

优势

- 轻松加密：通过密钥管理服务的密码运算 API，在线对数据直接加解密；
- 安全可靠：直接通过主密钥进行数据加解密保护，保证明文数据不落盘。

场景示意图



场景二：信封加密

通过调用密钥管理服务（KMS）的密码运算 API 在线生成数据密钥，数据密钥通过用户主密钥加密并支持安全导出，通过导出的数据密钥在本地进行大规模数据的加解密。

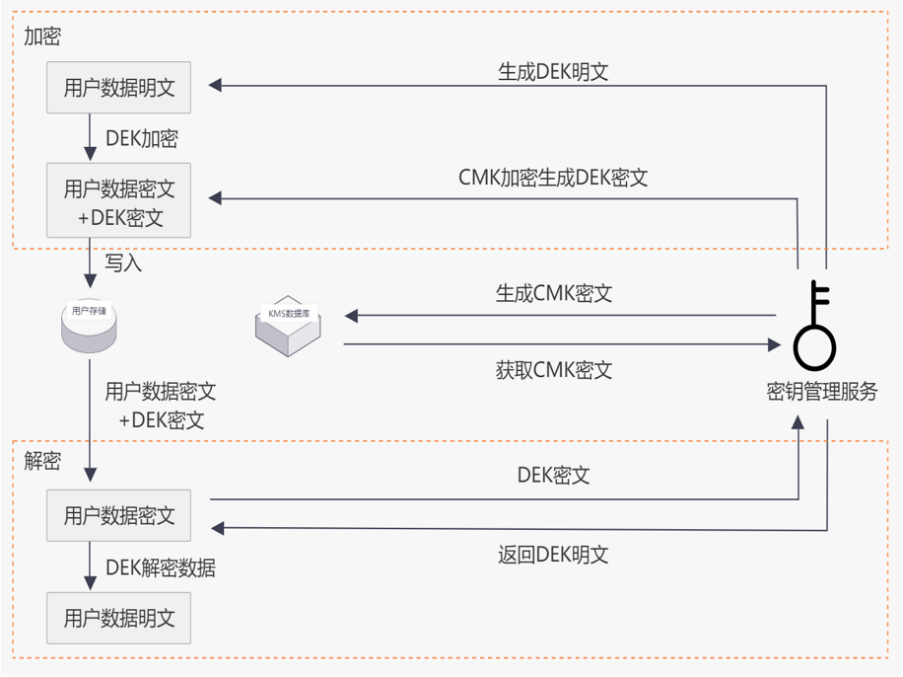
场景特点

用于海量大型数据或对性能敏感数据的加密保护，保证业务访问体验

优势

- 高效易用：通过创建数据密钥，实现本地数据的离线加密，避免移动大量数据产生安全隐患；
- 双重加密：通过主密钥和数据密钥两级密钥结构，确保数据密钥的随机性和安全性，保证数据加密性能。

场景示意图



场景三：签名验签

通过密钥管理服务（KMS）创建非对称密钥，签名者通过调用密码运算 API 使用私钥计算消息签名，同时获取公钥并分发至消息接收者，接收者使用公钥对消息进行签名验证。

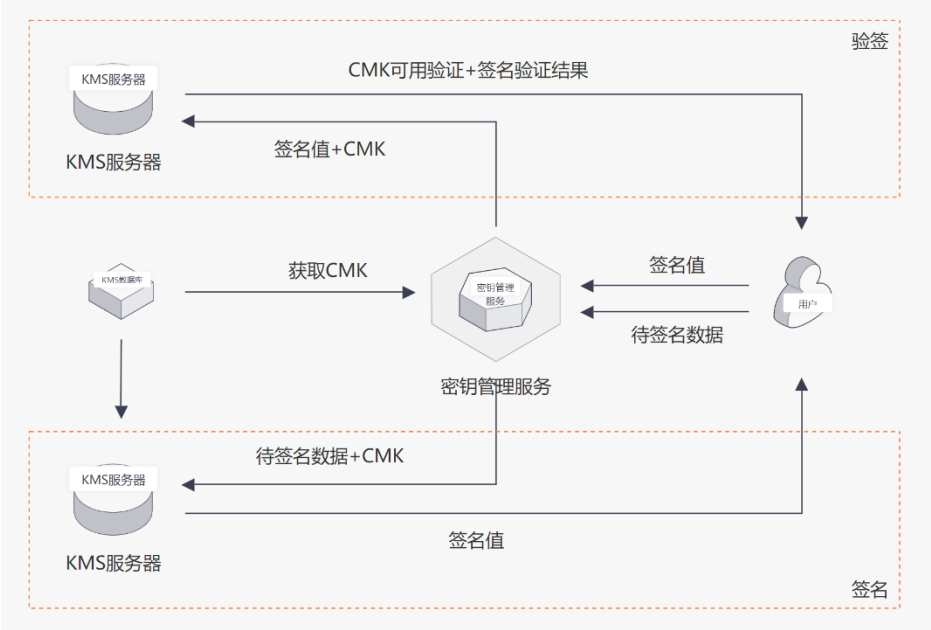
场景特点

用于信任程度不对等的系统之间，实现敏感信息的安全传递

优势

- 应用广泛：通过非对称密钥实现签名验签，广泛用于数据防篡改、身份认证等相关技术领域；
- 安全保障：支持主流的非对称密钥算法并且提供足够的安全强度，保证数字签名的安全性。

场景示意图



1.6. 产品规格

本章为您介绍密钥管理服务 KMS（Key Management Service）的服务模式及规格说明。

说明：自 2024 年 9 月 13 日起，KMS 全新升级上线包周期版，升级后不再支持按需版本的开通，新用户需购买包周期版 KMS（基础版、企业版），原已开通按需版 KMS 的用户仍可继续使用按需密钥，并依据按需版计费标准计费。

包周期服务

KMS 包周期版本提供不同规格的服务版本，包括基础版、企业版，您可以根据本章节列出的版本对比信息，选择合适的服务版本。同时，KMS 包周期服务提供免费的默认密钥，支持用于云产品加密功能。

√表示支持，×表示不支持。

对比项	子项	默认密钥	基础版	企业版
计费模式		免费	包周期	包周期
应用场景	云产品透明加密	√	√	√
	用户自建应用加密	×	√	√
	密评合规	×	×	√
	证书管理	×	√	√
配额	默认计算性能	750QPS	2000QPS	2000QPS
	密钥数量	每个天翼云账号在每个资源池，可为每个云产品创建 1 个默认密钥	0-2000 个	0-2000 个

	证书数量	×	0-1000 个	0-1000 个
接入网络类型	_	公网（由 KMS 管理）	openapi（公网） VPC 间调用（私网）	openapi（公网） VPC 间调用（私网）
密钥管理	密钥规格	AES_256	对称密钥： AES_256 非对称密钥： RSA_2048	对称密钥： AES_256 SM4 非对称密钥： RSA_2048 SM2
	导入外部密钥材料（BYOK）	×	√ 对称密钥支持	√ 对称密钥支持
	密钥自动轮转	×	√ 对称密钥支持	√ 对称密钥支持
	计划删除密钥	×	√	√
	密钥删除保护	×	√	√
	密钥别名管理	√（系统默认别名）	√	√
	密钥标签管理	√	√	√
密码运算	数据加解密	√（云产品）	√	√
	签名验签	×	√	√
	完整性校验	×	√	√
证书管理	创建证书	×	√	√
	导入证书	×	√	√
	吊销证书	×	√	√
	删除证书	×	√	√

按需服务

对于前期已开通按需版本的客户，仍可继续使用按需服务。按需服务无产品规格上的区分，根据业务情况创建所需的密钥或证书资源即可。

KMS 服务会根据所创建的密钥类型及个数、API 调用次数进行统计并计费。

- 按需按服务支持密钥规格

密钥类型	算法类型	保护级别	是否支持加解密	是否支持签名验签
对称密钥	AES_256	Software/HSM	√	×
	SM4	HSM	√	×
非对称密钥	RSA_2048	Software/HSM	√	√
	SM2	HSM	√	√

- 按需版本资源配额
  - ✓ 默认主密钥：同一云产品在同一资源池仅有一个默认主密钥。
  - ✓ 用户主密钥-对称密钥：暂不限制创建个数。
  - ✓ 用户主密钥-非对称密钥：限制密钥的版本数量，同一用户在同一资源池最多创建 50 个版本。

1.7. 与其他云服务关系

KMS 集成天翼云产品提供服务端加密能力，实现云上原生数据提供加密保护，有效提升默认安全能力。在创建云产品资源时开启加密功能，您可以自定义加密密钥，支持选择默认密钥和您在 KMS 中自行创建的用户主密钥。

服务端加密优势

- 提升云产品内生安全性

KMS 面向天翼云产品提供内生的数据加密能力，有效提升云内数据的安全性，您可以在创建云产品时开启加密功能，也可以在后续使用中开启。

加密过程中使用的密钥由用户自定义选择，集中托管在 KMS 服务中，KMS 已通过国家密码管理局审查，获得商用密码产品认证，合规性得到有效保障。

- 降低研发成本

使用云产品服务端加密能力，您无需自行构建和维护密钥管理基础设施，无需考虑自研数据加密能力所涉及的密钥管理安全及合理性、加密算法的研发等一系列复杂的工程，大幅度降低开发成本

- 加密过程透明无感知

服务端加密为您提供内嵌至云服务中的加密方案，您无需关注底层数据加密的细节，只需一键开启加密功能，即可实现数据加密。

支持服务端加密的云产品

- 云硬盘
- 对象存储
- 弹性文件
- 关系型数据库 MYSQL 版

功能详情详见云产品服务端加密。

2. 计费说明

2.1. 计费概述

本章为您介绍密钥管理服务 KMS（Key Management Service）的计费模式、费用组成。

注意：自 2024 年 9 月 10 日起，KMS 全新升级上线包周期版，升级后不再支持按需版本的开通，新用户需购买包周期版 KMS（基础版、企业版），原已开通按需版 KMS 的用户仍可继续使用按需密钥，并依据按需版计费标准计费。

计费模式

KMS 产品当前支持包周期版本及按需版本，对应两种计费模式：

- 包年/包月计费：一种预付费模式，即先付费再使用。您可根据业务需要，选择合适的包周期服务版本（基础版、企业版），一次性支付一个月/多个月/一年/多年的费用，支付成功后，KMS 服务资源将被系统分配给用户使用，直到超过保留期后被系统回收。
- 按使用量计费 ：一种后付费模式，即先使用再付费。在结算时会按照您在按需版本中，实际资源使用量收取费用，如密钥数量、API 调用量等。

注意：KMS 包周期版本上线后，则不再允许新增用户开通按需版本，已开通按需版本账号可继续使用存量密钥等资源，若需长期使用，请注意保证账户余额充足，避免因账号欠费导致服务冻结、资源释放。

优惠政策

KMS 包周期版支持包年优惠政策，若您按年购买，可享受 8.5 折优惠。

账单和用量查询

您可以在天翼云管理中心-账单管理中的流水账单、账单详情页面查看计费详情，支持导出账单明细。

2.2. 计费项

本章为您介绍密钥管理服务 KMS（Key Management Service）的计费项、标准资费及折扣。

包周期版

- 包周期服务费

计费项	说明	标准资费（元/月）	计费周期
基础版	提供软件保护等级服务，支持客户构建专属的密钥库，具备高度可扩展性；同时提供极简的密码运算接口能力，满足应用的安全快速集成。	3099	1-11 个月、1年、2年、3年
企业版	提供硬件保护等级服务，底层对接使用经国家密码管理局认证的密码机硬件，提供更高安全与合规等级保证的资源管理及密码运算服务，满足监管机构的检测认证要求。	9699	1-11 个月、1年、2年、3年

关于不同服务版本的规格参数，请参见产品规格。

针对一次性包年付费，可享 8.5 折优惠，如一次性支付 2 年，费用=包月标准价格*24*85%。

按需版

- 密钥托管费

密钥创建后托管在 KMS 服务产生的费用，按照密钥类型、密钥个数以天为周期进行计费。

计费项	说明	标准资费（元/天）	计费周期
默认主密钥	在使用云产品加密功能时，由云产品为用户自动创建的主密钥，同时托管在用户主账号下。	免费	——
用户主密钥-软件密钥	由用户自主创建且保护级别为 Software 的主密钥。	0.014	天
用户主密钥-硬件密钥	由用户自主创建且保护级别为 Hsm 的主密钥。	0.237	天

说明：计划删除状态的密钥不收费。

- API 调用费

密钥创建后通过接口调用产生的请求费用，按照 API 请求次数计费，每个账户每月有 20000 次的免费请求次数，超过 20000 次后开始计费。

计费项	含义	标准资费（万次/月）	计费周期
API 调用	用户通过云原生接口调用密钥实现密码运算操作产生的调用量	0.014	月

说明：仅密码及证书运算类接口调用收费。

2.3. 续订说明

若您购买了包周期版 KMS 服务，为避免 KMS 服务到期后停服导致您的业务无法使用密钥等资源，需要在服务到期前为实例手动续订，或设置到期自动续订策略。

到期说明

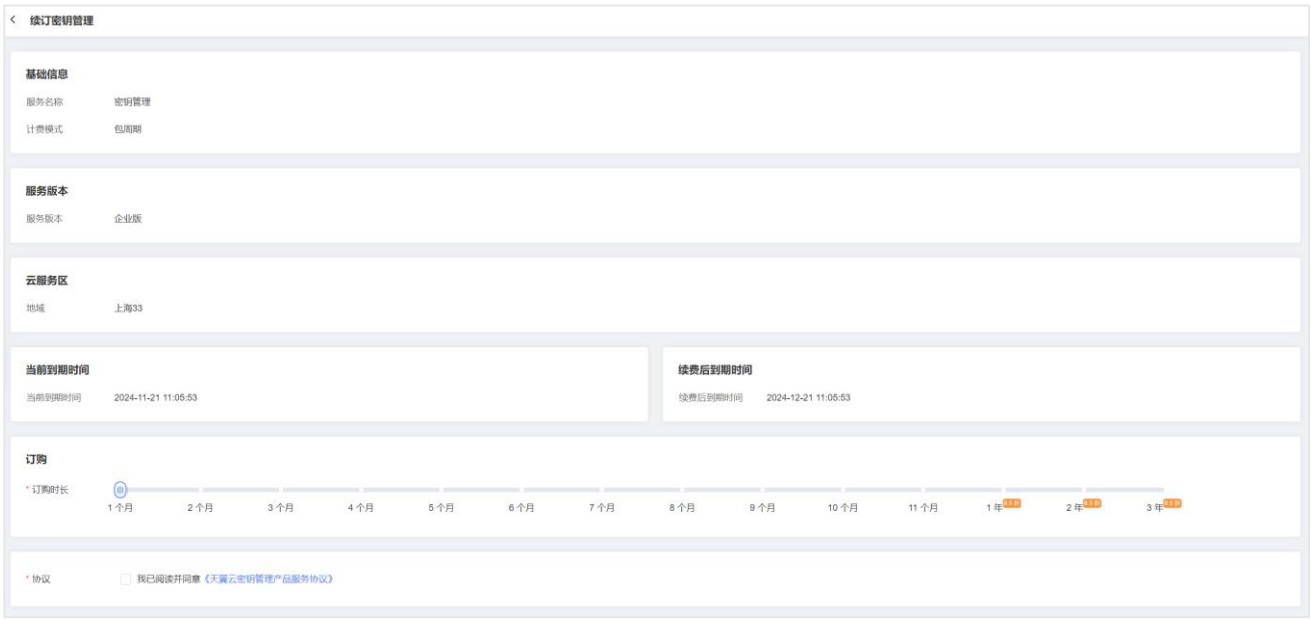
- 服务即将到期前，系统会以短信或邮件的形式提醒服务即将到期，并提醒用户续订。
- 服务到期后，如果没有按时续订，平台会冻结服务，但用户的资源及配置信息会提供 15 天的保留期。

> 说明：
> 保留期内，平台会冻结 KMS 服务，用户购买包周期版本后所创建的密钥等资源不可用，即无法正常进行加解密等运算操作。保留期满，用户若仍未续订，平台会清除服务资源，用户所创建的密钥等资源及其所有配置将会被删除且不可恢复。 如果您仍需继续使用 KMS 服务中的密钥等资源，请提前进行续订。

续订说明

- 服务支持手动续订，需要在服务到期前进入 KMS 产品控制台或天翼云续订管理页面操作。续订规则可参考续订规则说明。
- 在购买 KMS 服务时，支持勾选并同意“自动续订”，则在服务到期前，系统会自动按照默认的续订周期生成续费订单并进行续费，无须用户手动续订。自动续订规则可参考自动续订。

> 说明：
> 若购买 KMS 服务时勾选了“自动续订”，系统将会默认设置续费周期：
> 按月购买，自动续订周期默认为 3 个月。
> 按年购买，自动续订周期默认为 1 年。
> 如需要修改自动续费周期，可进入天翼云“费用中心 > 订单管理 > 续订管理”页面，在资源页面找到待修改自动续订的资源，单击操作列的“修改自动续订”，拖动“续订周期”可修改自动续订周期，当自动续订周期达 1 年或以上时，将可享受包年折扣。



2.4. 退订说明

KMS 服务支持退订，可通过 KMS 服务控制台界面、天翼云费用中心发起并完成退订操作。

退订说明

您可根据需要，在符合天翼云退订规则的前提下，灵活退订 KMS 服务。目前退订包含七天无理由全额退订和非七天无理由退订以及其他退订，退订规则详情见退订规则说明。

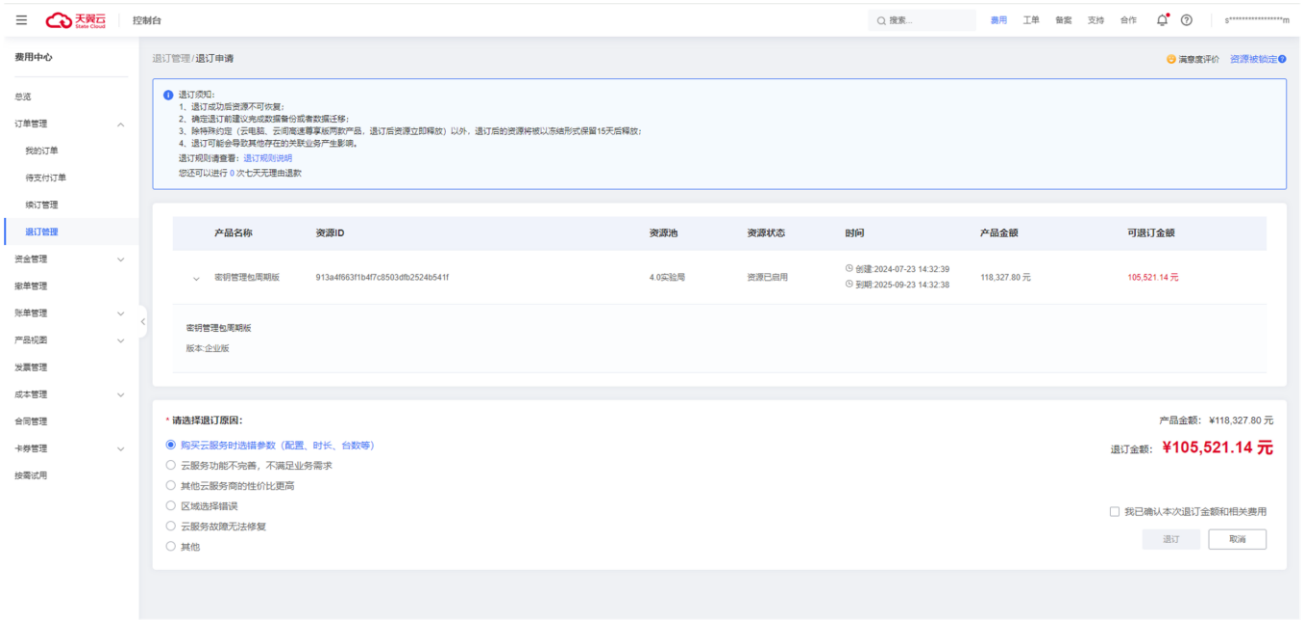
退订完成后，退款金额会退回账户余额，客户可根据需要进行提现。提现操作详情见余额提现。

> 说明：
> KMS 基础版、企业版服务支持退订；默认密钥由天翼云云服务创建，为免费资源，无须退订。
> 成功发起退订后，KMS 将转入冻结状态，冻结期 15 天。冻结期间，密钥等资源及配置会保留 15 天，15 天后资源被释放，释放后无法恢复。

操作步骤

1. 登录天翼云控制中心。
2. 单击页面顶部的区域选择框，选择区域。
3. 在产品服务列表页，选择“安全 > 密钥管理”。
4. 在左侧导航栏，选择“信息概览”，进入信息概览页面。

5. 在当前服务信息展示页面，点击“退订”。
6. 进入退订申请页面，确认退订信息，信息确认无误后选择退订原因，勾选“我已确认本次退订金额和相关费用”后，点击“退订”后即可进行退订。



7. 系统提示退订申请提交成功，可前往订单详情查看退订进度。

2.5. 到期与欠费说明

KMS 目前存在包周期、按需两种付费方式，关于到期、欠费后对 KMS 服务会进入冻结状态，期间服务将不可用。如需继续使用服务，您需要进行包周期服务续订、账户充值。

注意： 若您使用了云硬盘、对象存储、弹性文件、关系数据库 Mysql 版，并开启了 KMS 服务端加密功能，请特别注意 KMS 服务的到期或欠费状态。若 KMS 服务到期未及时续订、欠费未及时充值而进入冻结，云产品

加密所使用的密钥将无法正常使用，加密、解密等调用请求将被拒绝，云产品将会出现异常。

包周期到期说明

若您购买了包周期版 KMS 服务，为避免 KMS 服务到期后停服导致您的业务无法使用密钥等资源，需要在服务到期前为实例手动续订，或设置到期自动续订策略。

- 服务即将到期前，系统会以短信或邮件的形式提醒服务即将到期，并提醒用户续订。
- 服务到期后，如果没有按时续订，平台会冻结服务，但用户的资源及配置信息会提供 15 天的保留期。

说明：
保留期内，平台会冻结 KMS 服务，用户购买包周期版本后所创建的密钥等资源不可用，即无法正常进行加解密等运算操作。保留期满，用户若仍未续订，KMS 服务将被释放，用户所创建的密钥等资源及其所有配置将会被删除且不可恢复。 如果您仍需继续使用 KMS 服务中的密钥等资源，请提前进行续订。

按需欠费说明

当您已开通 KMS 按需版服务，若账户进入欠费状态，平台会冻结 KMS 按需版服务，服务将不可用，同时按需资源会进入保留期。

- KMS 按需版服务资源保留期为 15 天，保留期内 KMS 按需服务停止服务，用户对密钥管理系统的访问将被拒绝。
- 欠费期间，KMS 将不会再收取密钥托管费用，但当月进入冻结状态前所产生的 API 调用费，在本月底仍会生成账单，累计欠费。
- 若您在保留期内充值，充值后系统会自动扣减欠费金额。
- 若保留期到期您仍未充值，KMS 服务中的密钥等资源会被释放且不可恢复。
- 结清账单后，已欠费冻结的 KMS 按需服务会自动启动并进入可用状态。

3. 快速入门

3.1. 注册天翼云账号

在创建和使用密钥管理服务之前，您需要先注册天翼云门户的账号。本节将介绍如何进行账号注册，如果您拥有天翼云的账号，请跳转至开通密钥管理服务。

1. 登录天翼云门户 http://www.ctyun.cn，点击**注册**；

2. 在注册页面，请填写“邮箱地址”、“登录密码”、“手机号码”，并点击**同意协议并提交**，如 1 分钟内手机未收到验证码，请再次点击**免费获取短信验证码**；

3. 注册成功后，可到邮箱激活您的账号或立即体验天翼云。

3.2. 购买密钥管理服务

密钥管理服务（KMS）包周期版提供基础版、企业版两个规格，本章为您介绍如何购买 KMS 服务。

前提条件

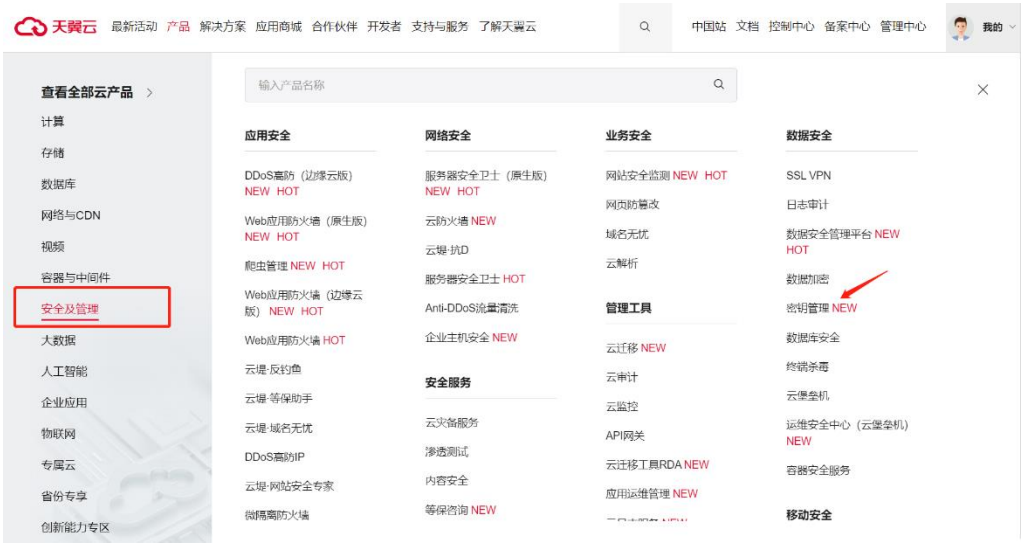
已经注册天翼云账号并完成实名认证。

规格限制

- 基础版提供软件保护等级服务，支持客户构建专属的密钥库，具备高度可扩展性；同时提供极简的密码运算接口能力，满足应用的安全快速集成。
- 企业版提供硬件保护等级服务，底层对接使用经国家密码管理局认证的密码机硬件，提供更高安全与合规等级保证的资源管理及密码运算服务，满足监管机构的检测认证要求。
- 基础版、企业版计算性能默认为 2000QPS，默认密钥的计算性能为 1000QPS.
- 基础版、企业版最多可创建 2000 个密钥、1000 个证书。

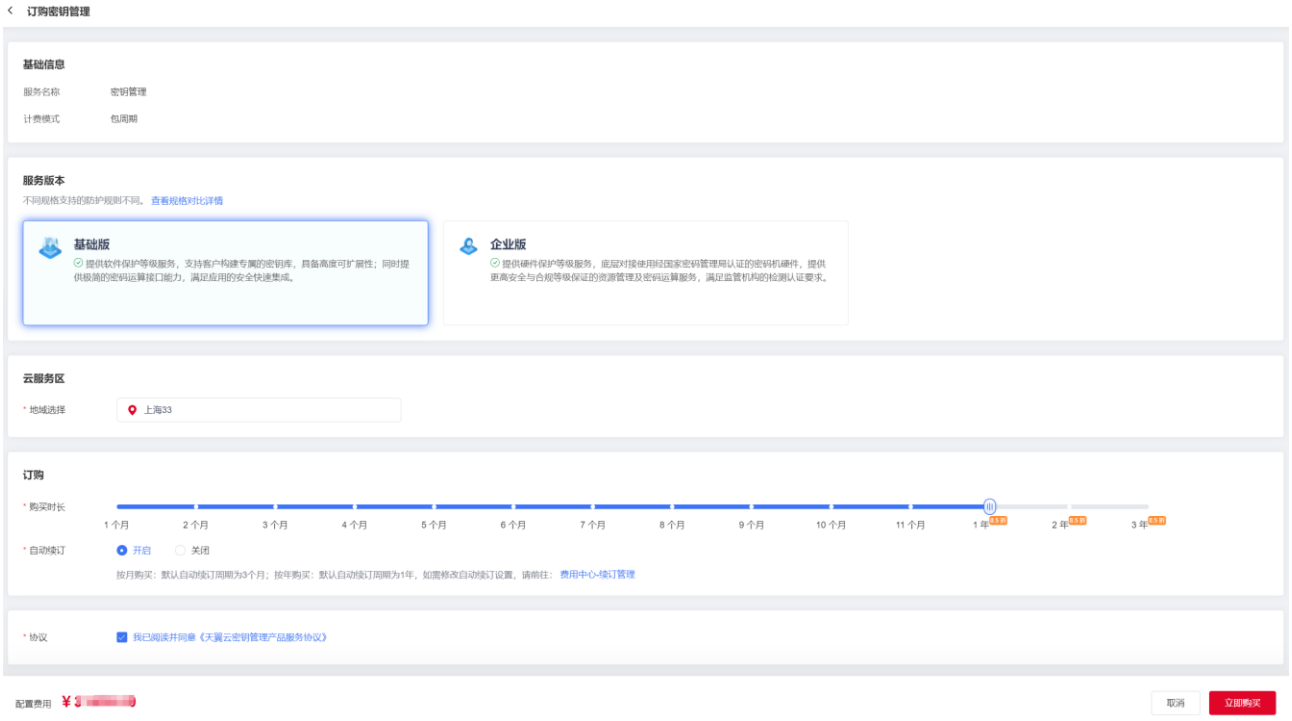
操作步骤

1. 进入天翼云门户并登录，在产品导航栏，定位到“安全与管理”分类，找到密钥管理，点击进入。

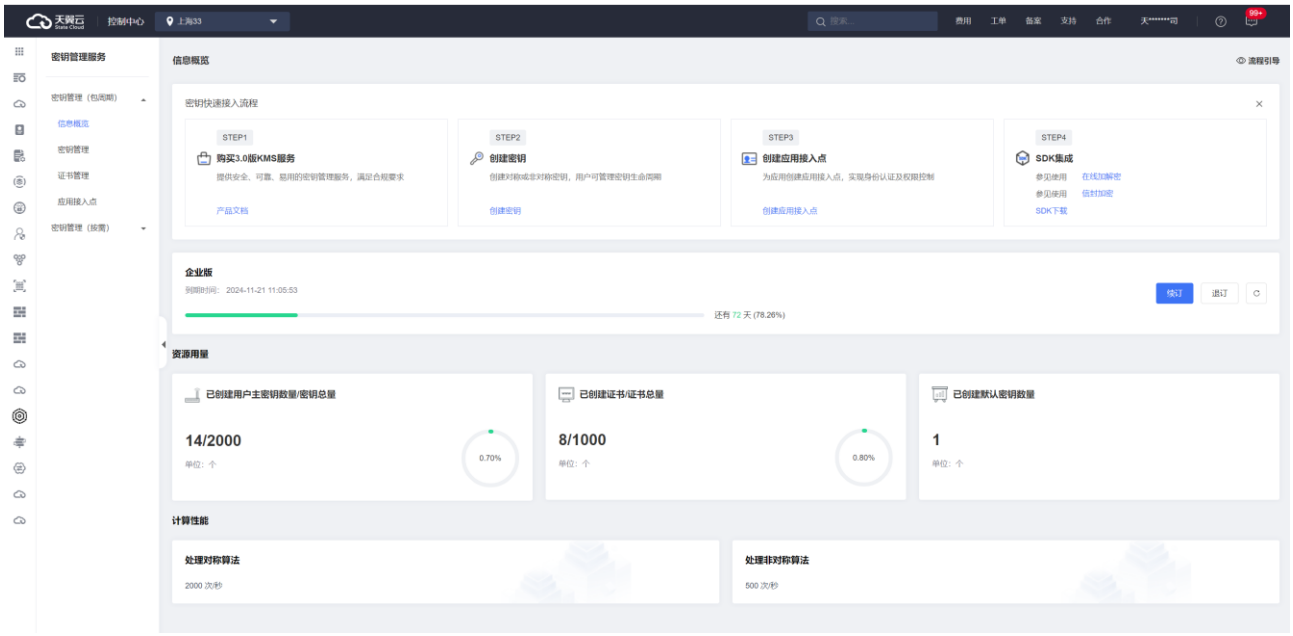


2. 进入密钥管理 产品详情页，点击**立即开通**。

3. 进入到**密钥管理服务订购**页面，选择服务版本、服务区域，设置订购时长，确认参数配置后，阅读《天翼云密钥管理服务协议》，并勾选“我已阅读并同意《天翼云密钥管理服务协议》”，点击“立即购买”。



4. 进入**订单详情**页面，确认支付金额，点击**立即支付**。
5. 完成订单支付，可在**订单详情页**查看订单状态，状态更新为“已完成”后代表服务已开通。
6. 服务开通后，您可进入密钥管理服务控制台创建资源。



3.3. 创建用户主密钥

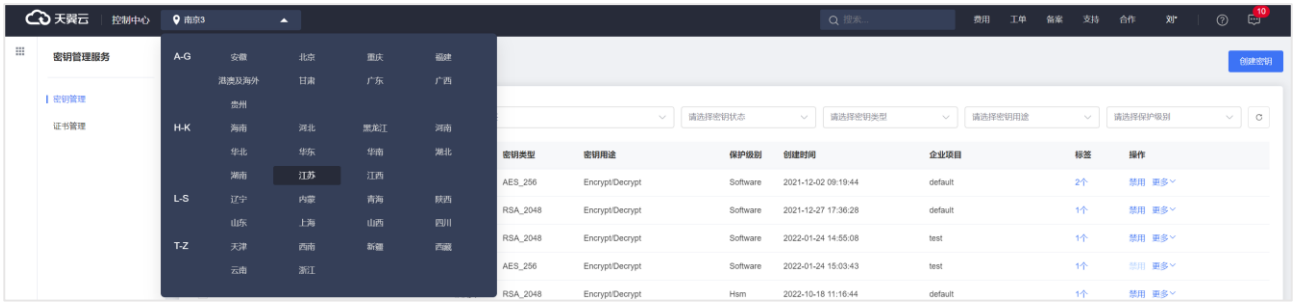
开通密钥管理服务后，您可以在控制台轻松地创建不同类型的密钥，以满足各类业务场景的需求。同时密钥集中托管在 KMS 服务中，便于统一管理，满足安全与合规要求。

前提条件

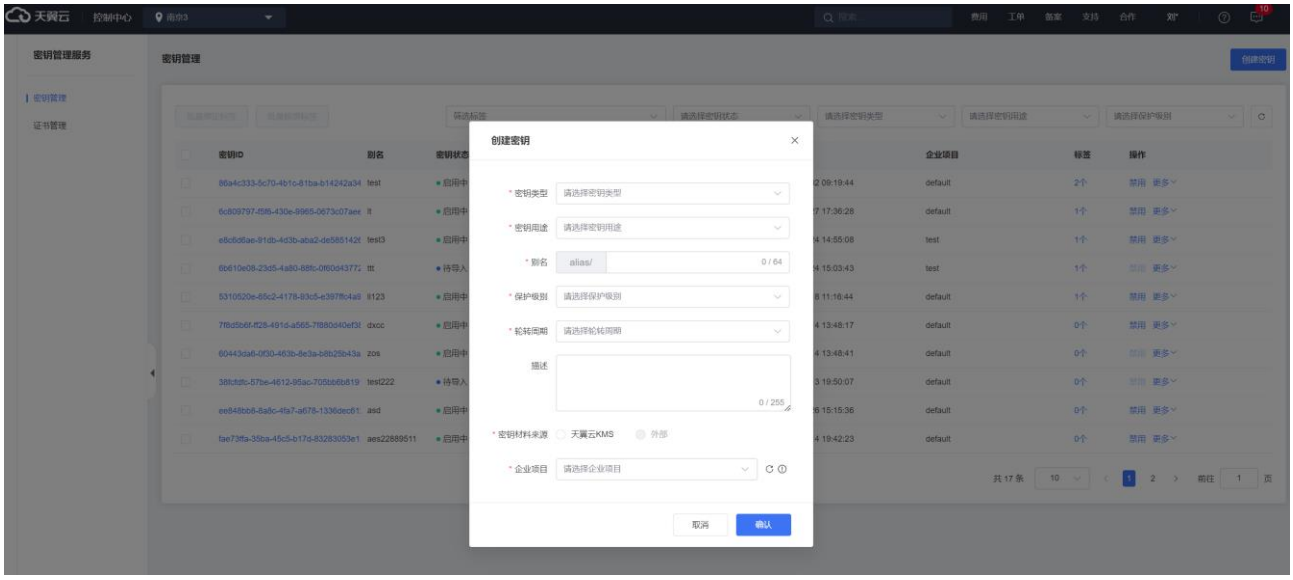
- 已开通密钥管理服务。

创建密钥

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；



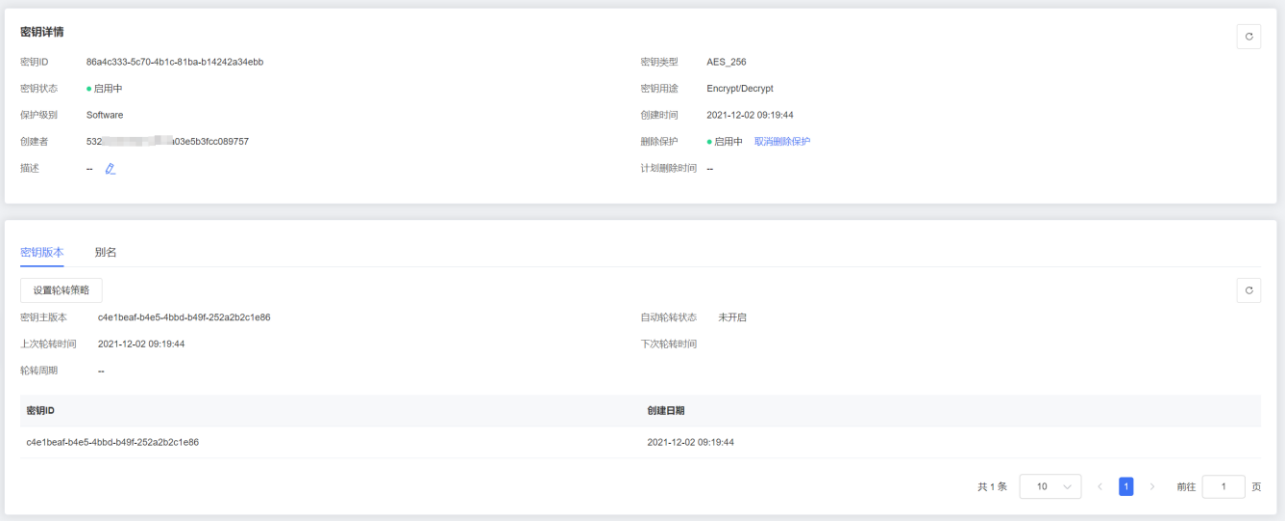
3. 单击**创建密钥**，在弹出的**创建密钥**对话框，根据页面提示进行配置；



配置项	说明
密钥类型	取值： 对称密钥类型： • AES_256 • Ctyun_SM4 非对称密钥类型： • RSA_2048 • Ctyun_SM2
密钥用途	取值： • Encrypt/Decrypt：数据加密和解密 • Sign/Verify：产生和验证数字签名 说明：对称密钥不支持 Sign/Verify 用途。
别名	用户主密钥的可选标识。 更多操作，请参见别名管理。
保护级别	取值： • Software：通过软件模块对密钥进行保护。 • Hsm：将密钥托管在密码机中，使密钥获得高安全等级的专用硬件的保护。
描述	密钥的描述信息。
轮转周期	自动轮转的时间周期。取值： • 不开启：不开启轮转 • 30 天

配置项	说明
	• 90 天 • 180 天 • 自定义：7~730 天 **说明**：仅对称密钥（AES_256、Ctyun_SM4）支持设置自动轮转周期。
密钥材料来源	取值： • 天翼云 KMS：密钥材料将由 KMS 生成。 • 外部：KMS 将不会生成密钥材料，您需要将自己的密钥材料导入 KMS。更多信息，请参见导入密钥材料。 **说明**：仅对称密钥的 AES_256 支持设置导入密钥材料。
企业项目	选择密钥归属的企业项目。默认为 default。

4. 单击确定，完成密钥创建。您可以在密钥列表查看密钥 ID、密钥状态、密钥类型、密钥用途、密钥保护级别等信息。



使用密钥

您可以将创建的密钥集成到自建应用中，实现应用层的密码技术改造。同时可用于已集成 KMS 服务的云产品中，满足云产品服务端加密。

- 自建应用集成 KMS 实现密码技术改造

  KMS 服务提供极简的 OpenAPI，您可以轻松实现调用，用于数据加解密、签名验签等场景；具体调用方式，请参见 API 参考。

- 云产品集成 KMS 密钥实现服务端加密

  当前 KMS 服务已为云硬盘、对象存储、弹性文件、关系数据库 MySQL 版产品提供服务端加密能力，您在创建云资源时，可一键开启加密，加密过程透明无感知。更多信息，请参见服务端加密。

4. 用户指南

4.1. 总览

总览页面帮助您快速了解 KMS 服务的实际使用情况，包含 KMS 服务规格、到期时间、资源占用情况等指标。您可以根据使用数据，选择是否需要进行服务续订、资源扩容等。

服务版本

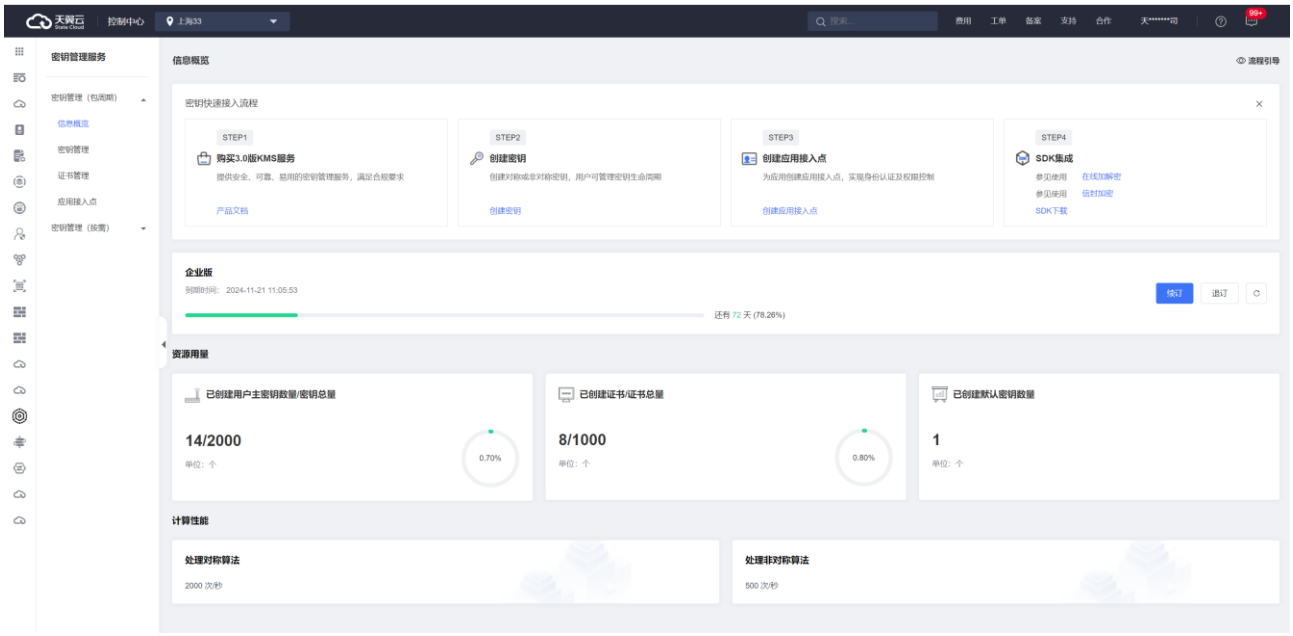
- 服务规格：当前账号已开通的服务规格，包括基础版、企业版，规格描述详见产品规格。
- 到期时间：当前包周期服务的到期时间。若您根据业务需求判断是否需要持续使用 KMS 服务，若需要继续使用，请在到期前及时续订，否则 KMS 服务将在到期后进入冻结状态，服务将不可用。

资源用量

- 用户主密钥数量：用户开通包周期服务后，通过控制台/API 接口自主创建的用户主密钥个数，以及当前已创建数量占用服务版本对应配额的百分比。
- 证书数量：用户开通包周期服务后，通过控制台/API 接口自主创建的证书个数，以及当前已创建数量占用服务版本对应配额的百分比。
- 默认密钥数量：用户在使用云产品加密功能时，由云产品触发创建的默认密钥数量。

计算性能

- 处理对称算法：对称算法所对应接口的请求配额（上限），超过 API 请求配额后，KMS 回限制请求（即拒绝访问）。
- 处理非对称算法：非对称算法所对应接口的请求配额（上限），超过 API 请求配额后，KMS 回限制请求（即拒绝访问）。



4.2. 密钥管理

4.2.1. 密钥管理概述

密钥管理服务提供密钥的全托管的生命周期管理能力，支持基于 API 接口的数据加解密和数字签名验签。

KMS 支持的密钥类型说明

KMS 对加密算法、保护级别以及应用场景的支持情况请参见如下表格。

密码算法大类	密码算法子类	保护级别	是否支持加解密	是否支持签名验签
对称密钥	AES_256	• Software • HSM	支持	不支持
	SM4	• HSM	支持	不支持
非对称密钥	RSA_2048	• Software • HSM	支持	支持
	SM2	• HSM	支持	支持

- 对称密钥主要用于数据的加密保护场景，可通过接口调用进行在线加密或者信封加密。更多信息，请参见对称加密概述。
- 非对称密钥可用于数据加密和数字签名。在 KMS 创建的非对称用户主密钥（CMK），由一对关联的公钥和私钥构成。公钥可以被分发给任何人，而私钥由 KMS 确保安全性，不提供任何接口导出非对称密钥的私钥。使用者仅能通过接口调用私钥进行签名运算或者数据解密。更多信息，请参见非对称加密概述。

密钥生命周期管理

KMS 提供集中托管的密钥全生命周期管理，您可以轻松创建并使用密钥。

功能	说明	参考文档
密钥生命周期管理	通过 KMS 可创建用户主密钥 CMK（Customer Master Key），支持对 CMK 进行启用、禁用、删除等生命周期管理。 密钥支持软件或硬件的密钥保护级别，硬件密钥通过硬件安全模块（HSM）的保护，满足更高的安全性。 支持导入自带密钥材料到 KMS 中（BYOK），满足一些特定的安全需求。	创建密钥 导入密钥材料 启用禁用密钥 计划删除密钥
密钥版本管理	支持通过密钥版本化或定期轮转来加强密钥使用的安全性，实现数据保护的安全策略。	密钥版本管理
密钥别名管理	支持设置密钥别名，更方便的使用密钥。	别名管理

密码运算

KMS 提供了云原生的密码运算 API，快速满足数据加密解密、数字签名验签等多样性需求。

功能	说明	参考文档
对称密钥运算	在线加密：适用于少量信息（6KB）的加密，直接通过用户主密钥 CMK 对数据进行加解密的操作。	在线加密
	信封加密：适用于海量数据的高性能加密，通过生成数据密钥 DEK，在本地实现数据的高效对称加解密处理。	信封加密
非对称密钥运算	签名验签：适用于敏感信息的传递，信息发送者通过发送签名和数据提供身份证明，信息接收者进行签名验证，校验数据的安全性。	签名验签

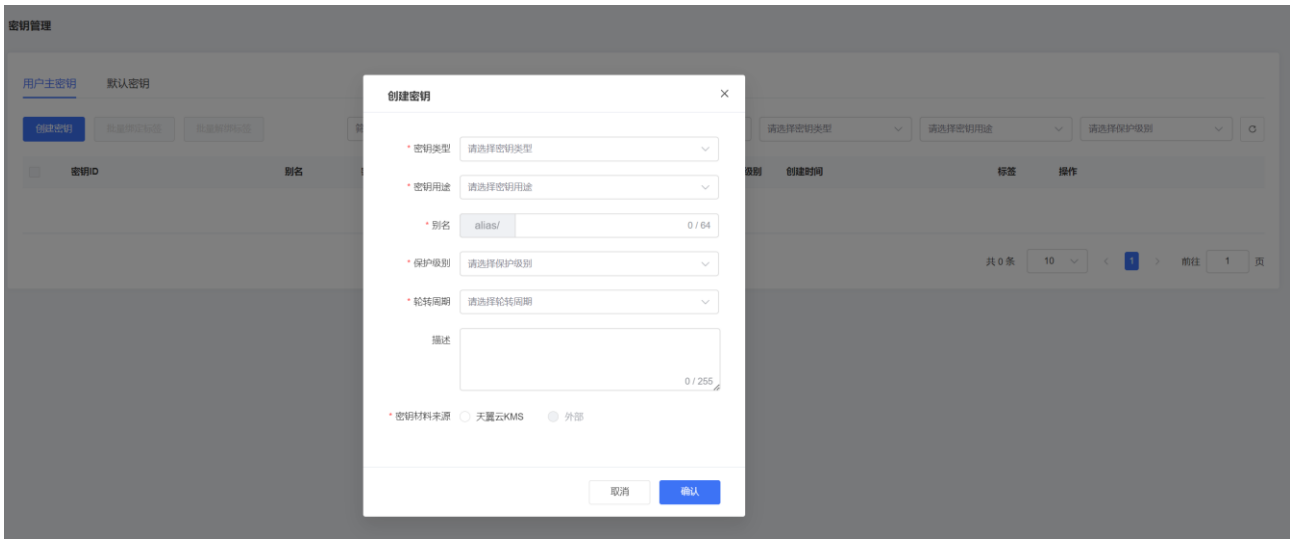
功能	说明	参考文档
	非对称密钥加解密：适用对敏感信息加密后进行传递，通过使用非对称密钥公钥对数据进行加密、私钥进行解密处理。	非对称密钥加解密

4.2.2. 创建密钥

本文为您介绍在控制台创建密钥的操作步骤。

操作步骤

1. 登录密钥管理服务控制台。
2. 在页面最上方的导航栏的资源池下拉列表，选择服务所在的区域。
3. 单击“创建密钥”，在弹出的创建密钥对话框，根据页面提示进行配置；



配置项说明

配置项	说明
密钥类型	取值： 对称密钥类型： • AES_256 • Ctyun_SM4（企业版支持） 非对称密钥类型： • RSA_2048 • Ctyun_SM2（企业版支持）

配置项	说明
密钥用途	取值： • Encrypt/Decrypt：数据加密和解密 • Sign/Verify：产生和验证数字签名 说明：对称密钥不支持 Sign/Verify 用途。
别名	用户主密钥的可选标识。 更多操作，请参见别名管理。
保护级别	取值： • Software：通过软件模块对密钥进行保护。 • Hsm：将密钥托管在密码机中，使密钥获得高安全等级的专用硬件的保护。
描述	密钥的说明信息。
轮转周期	自动轮转的时间周期。取值： • 不开启：不开启轮转 • 30 天 • 90 天 • 180 天 • 自定义：7~730 天 说明：仅对称密钥（AES_256、Ctyun_SM4）支持设置自动轮转周期。
密钥材料来源	取值： • 天翼云 KMS：密钥材料将由 KMS 生成。 • 外部：KMS 将不会生成密钥材料，您需要将自己的密钥材料导入 KMS。更多信息，请参见导入密钥材料。 说明：仅对称密钥的 AES_256 类型支持设置导入密钥材料。
企业项目	选择密钥归属的企业项目。默认为 default。 注：当前仅按需版本支持企业项目功能。

4. 单击**确定**，完成密钥创建。您可以在密钥列表查看密钥 ID、密钥状态、密钥类型、密钥用途、密钥保护级别等信息。

4.2.3. 导入密钥材料

用户主密钥包含密钥元数据（密钥 ID、密钥别名、描述、密钥状态与创建日期）和用于加解密数据的密钥材料。

- 当用户使用 KMS 管理控制台创建用户主密钥时，KMS 系统会自动为该用户主密钥生成密钥材料。
- 当用户希望使用自己的密钥材料时，可通过 KMS 管理控制台的导入密钥功能创建密钥材料为空的用户主密钥，并将自己的密钥材料导入该用户主密钥中。

注意事项

当您选择密钥材料来源为外部，使用您自己导入的密钥材料时，需要注意以下几点：

- 请确保您使用了符合安全要求的随机源生成密钥材料；
- 用户在使用导入密钥时，需要对自己密钥材料的可靠性负责；
- 请保存密钥材料的原始备份，以便在意外删除密钥材料时，能及时将备份的密钥材料重新导入 KMS。

导入密钥材料的功能特性

- **可用性与持久性**

在将密钥材料导入 KMS 之前，用户需要确保密钥材料的可用性和持久性。

导入的密钥材料与通过 KMS 创建密钥时自动生成的密钥材料的区别，如下表所示。

密钥材料来源	说明
外部导入	• 支持手动删除密钥材料，但该主密钥及其元数据仍然保留。 • 导入密钥材料时，可以设置密钥材料过期时间，密钥材料过期后，KMS 将自动删除密钥材料，但该主密钥及其元数据仍然保留。 • 导入的密钥材料被删除后，可以再次导入相同的密钥材料使得 CMK 再次可用。用户需自行备份密钥材料，以便密钥材料失效或误删除时重新导入该密钥材料。
KMS 创建	• 不能手动删除密钥材料，不能设置密钥材料过期时间。 • 密钥材料只能通过设置 CMK 计划删除时间后，到期后随 CMK 一并删除。

- **关联性**

  当您将密钥材料导入 CMK 时，该 CMK 与该密钥材料永久关联，不能将其他密钥材料导入该 CMK 中，即便密钥材料已经过期或者被删除。

- **独立性**

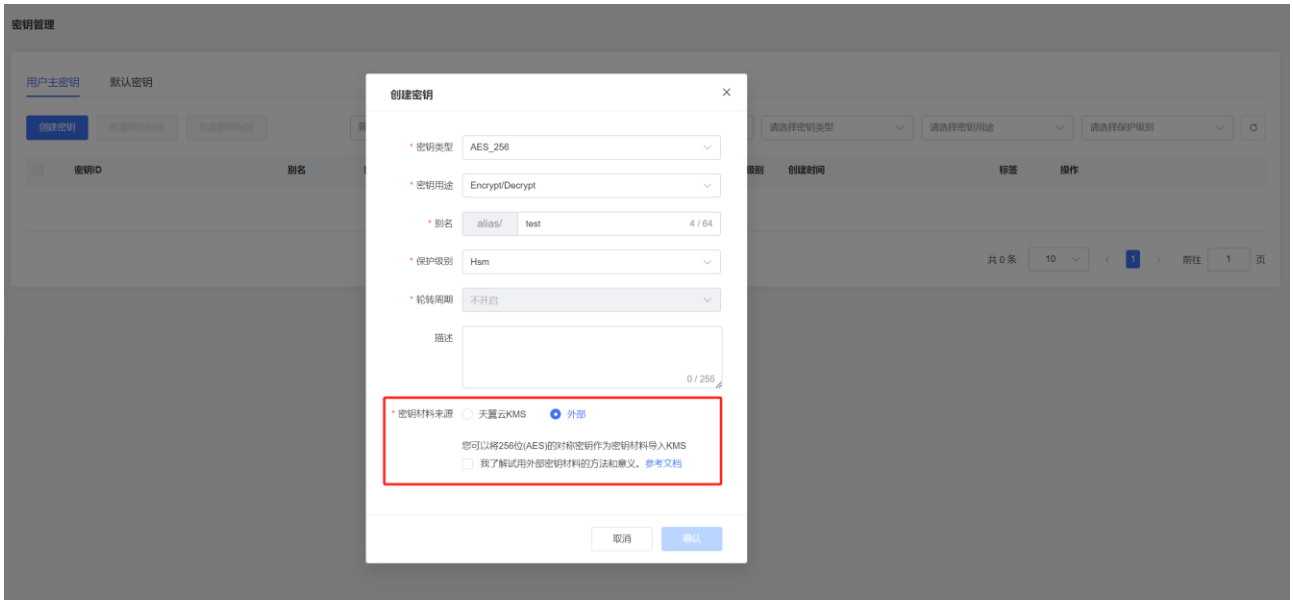
CMK 具有唯一性，即您使用 CMK 加密的数据，无法使用其他 CMK 进行解密，即便这些 CMK 都使用相同的密钥材料。

限制条件

- AES_256 类型的 CMK 需导入 256 位对称密钥作为密钥材料。
- 从 KMS 获取到的导入令牌与加密密钥材料的公钥具有绑定关系，一个令牌只能为其生成时指定的主密钥导入密钥材料。导入令牌的有效期为 24 小时，在有效期内可以重复使用，失效以后需要获取新的导入令牌和加密公钥。

操作步骤-导入密钥材料

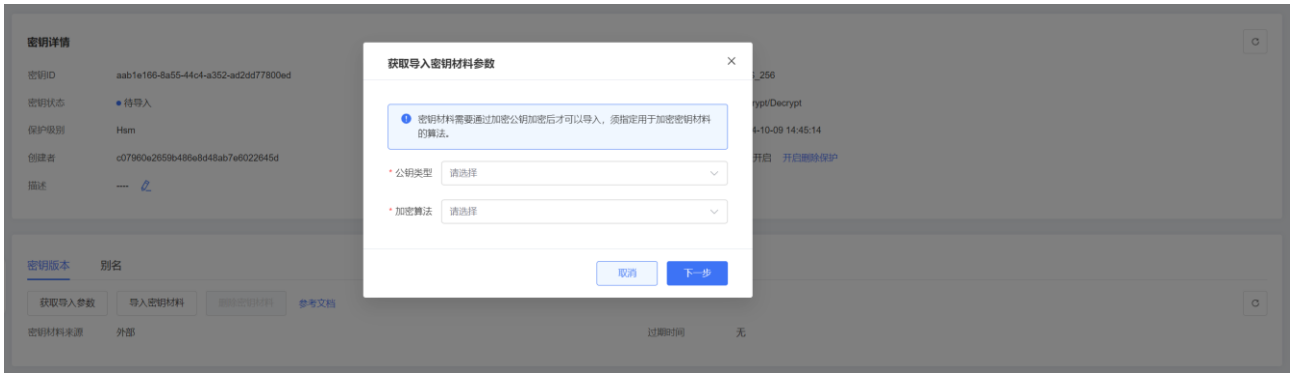
1. 创建用户主密钥，其中**密钥材料来源**选择**外部**，并勾选**“我了解使用外部密钥材料的方法和意义”**；



2. 获取导入密钥材料参数。

1）在密钥列表，点击**密钥 ID**，进入**密钥详情**，在**密钥材料**区域，单击**获取导入密钥材料参数**。

2）在**获取导入密钥材料参数**对话框，选择**公钥类型**、**加密算法**，单击**确定**。



配置项说明

配置项	说明
公钥类型	取值： • RSA_2048（默认）
加密算法	取值： • RSAES_PKCS1_V1_5 • RSAES_OAEP_SHA_1 • RSAES_OAEP_SHA_256

3）在获取导入密钥材料参数对话框，下载**加密公钥**和**导入令牌**，然后单击**确定**。

注意：导入令牌存在过期时间，请关注过期时间，及时进行导入。

3. 使用 OPENSSL 加密密钥材料。

加密公钥是一个 2048 比特的 RSA 公钥，使用的加密算法需要与获取导入密钥材料参数时指定的一致。由于加密公钥经过 Base64 编码，因此在使用时需要先进行 Base64 解码。您可以使用 OPENSSL 通过以下步骤获取加密的密钥材料。

1）创建一个密钥材料，使用 OPENSSL 产生一个随机数。

2）将加密公钥进行 Base64 解码。

3）根据指定的加密算法（以 RSAES_OAEP_SHA_1 为例）加密密钥材料。

4）将加密后的密钥材料进行 Base64 编码，保存为文本文件。

代码示例：

```
openssl rand -out KeyMaterial.bin 32
openssl enc -d -base64 -A -in PublicKey_base64.txt -out PublicKey.bin
openssl rsautl -encrypt -in KeyMaterial.bin -oaep -inkey PublicKey.bin  -keyform DER  -pubin -out EncryptedKeyMaterial.bin
openssl enc -e -base64 -A -in EncryptedKeyMaterial.bin -out EncryptedKeyMaterial_base64.txt
```

采用 OpenSSL 加密密钥材料，支持 RSAES_OAEP_SHA_256，RSAES_PKCS1_V1_5 和 RSAES_OAEP_SHA_1 三种密钥算法。OpenSSL 命令代码示例如下表所示：

密钥算法	OpenSSL 加密生成密钥材料命令代码示例
RSAES_OAEP_SHA_256	openssl pkeyutl -in PlaintextKeyMaterial.bin -inkey PublicKey.bin -out EncryptedKeyMaterial.bin -keyform der -pubin -encrypt -pkeyopt rsa_padding_mode:oaep -pkeyopt rsa_oaep_md:sha256

密钥算法	OpenSSL 加密生成密钥材料命令代码示例
RSAES_PKCS1_V1_5	openssl rsautl -encrypt -in PlaintextKeyMaterial.bin -pkcs -inkey PublicKey.bin -keyform der -pubin -out EncryptedKeyMaterial.bin
RSAES_OAEP_SHA_1	openssl rsautl -encrypt -in KeyMaterial.bin -oaep -inkey PublicKey.bin -keyform DER -pubin -out EncryptedKeyMaterial.bin

4. 导入密钥材料。

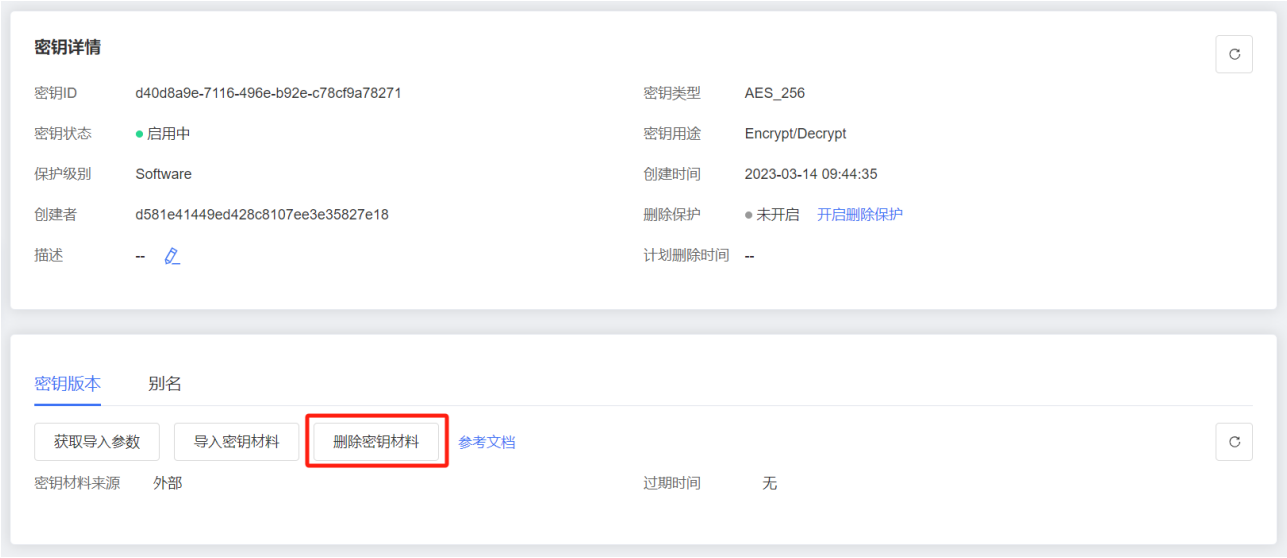
1）在密钥列表，点击**密钥 ID**，进入**密钥详情**，在**密钥材料**区域，单击**导入密钥材料**。

2）在**导入密钥材料**对话框，上传**加密密钥材料**和**导入令牌**，单击**确定**。

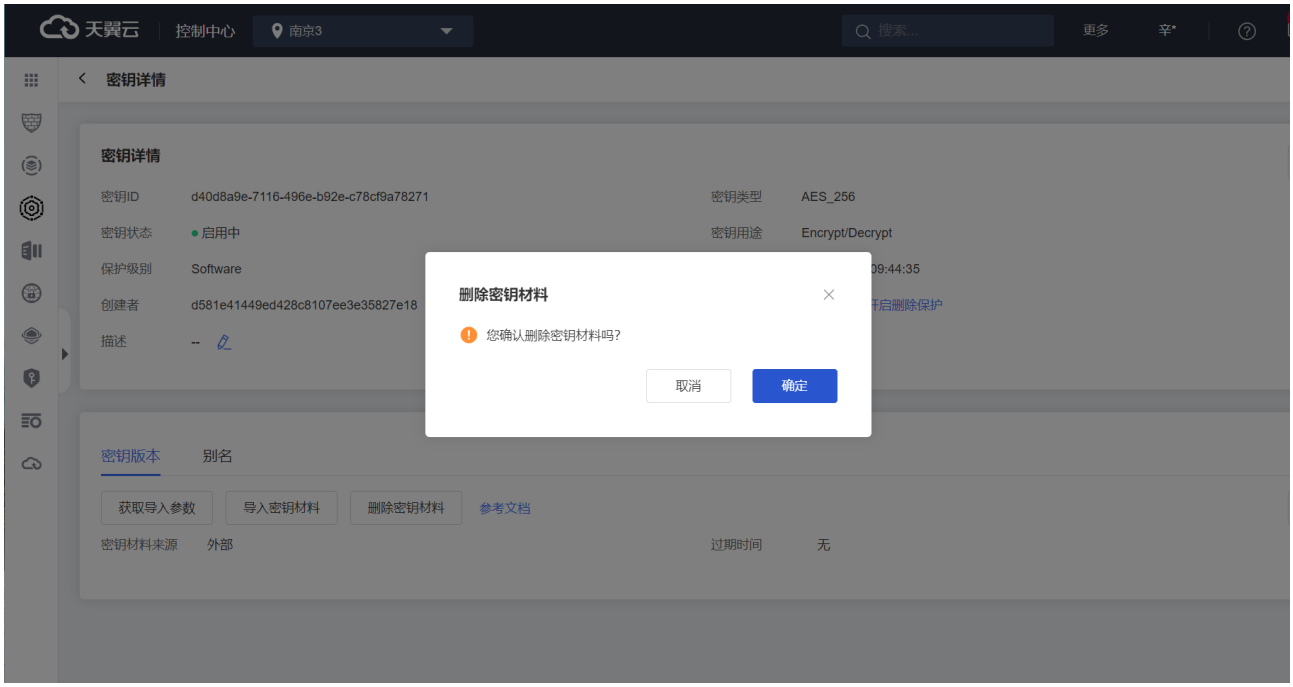
3）设置**密钥材料过期时间**，单击**确定**。导入密钥材料成功后，密钥状态从**待导入**更新为**启用中**。

操作步骤-删除密钥材料

1. 登录密钥管理服务控制台。
2. 在页面最上方的导航栏选择密钥所在的区域。
3. 在密钥列表，点击**更多**，进入**密钥详情**，在**密钥材料**区域，单击**删除密钥材料**。



4. 在删除密钥材料对话框，单击**确定**。密钥材料删除成功后，密钥状态从**启用中**更新为**待导入**。

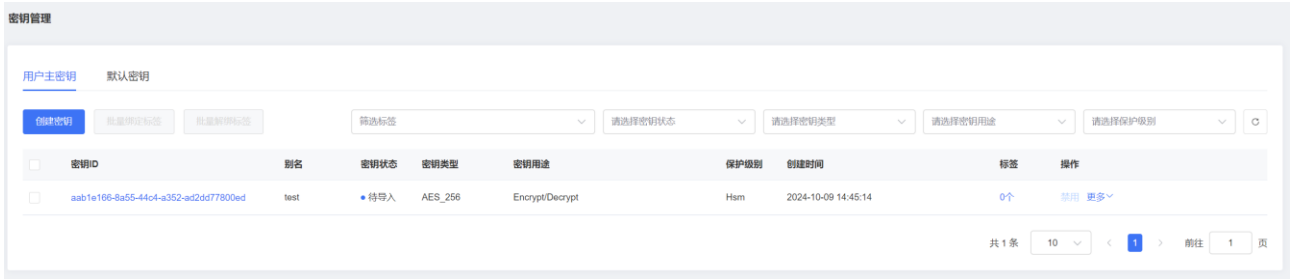


4.2.4. 查看密钥

成功创建了用户主密钥之后，您可以通过控制台查看密钥列表以及密钥详情信息。

操作步骤

1. 登录密钥管理服务控制台。
2. 在页面最上方的导航栏选择密钥所在的区域。
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**。



4. 在密钥列表中，查看密钥信息。密钥列表参数说明如下表所示；

参数	说明
密钥 ID	创建密钥时自动生成的密钥 ID。可点击进入密钥详情。

参数	说明
别名	密钥的别名。
密钥状态	密钥的状态，包含： • 启用中 • 已禁用 • 待删除 • 待导入
密钥类型	创建密钥时选择的算法类型，包含： 对称密钥： • AES_256 • Ctyun_SM4（企业版支持） 非对称密钥： • RSA_2048 • Ctyun_SM2（企业版支持）
密钥用途	创建密钥时选择的用途，包含： • Encrypt/Decrypt • Sign/Verify，仅非对称密钥支持
保护级别	创建密钥时选择的保护级别，包含： • Software • Hsm
创建时间	创建该密钥的时间。
标签	密钥绑定的标签个数，点击可查看标签键值信息。
操作	用户可以对密钥进行启用/禁用、计划删除密钥/取消计划删除密钥、编辑标签操作。

5. 在密钥列表点击密钥 ID，进入密钥详情页。

1）在**密钥详情**区域可查看当前密钥的详细信息；

密钥详情参数说明

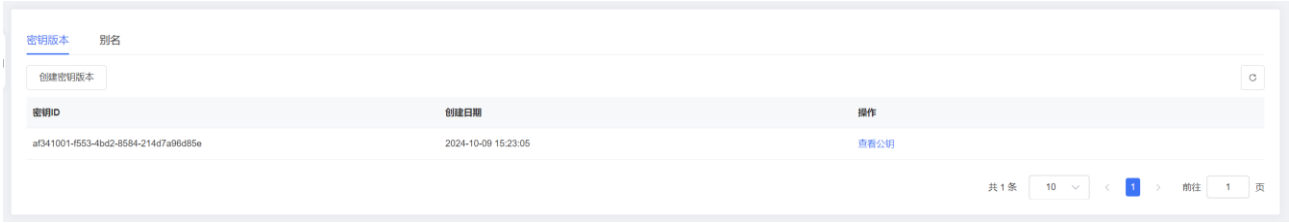
参数	说明
密钥类型	创建密钥时选择的算法类型，包含： 对称密钥： • AES_256 • Ctyun_SM4 非对称密钥： • RSA_2048 • Ctyun_SM2
密钥用途	创建密钥时选择的用途，包含： • Encrypt/Decrypt • Sign/Verify，仅非对称密钥支持
创建者	即 User_id。
密钥状态	密钥的状态，包含： • 启用中 • 已禁用 • 待删除 • 待导入
保护级别	创建密钥时选择的保护级别，包含： • Software • Hsm
创建时间	创建该密钥的时间。
删除保护	状态： • 未开启 • 启用中 说明：开启删除保护状态的密钥，将无法直接删除该密钥，从而避免误删除密钥。若确认要将密钥删除，需要将删除保护关闭。
描述	描述信息，可修改。

2）在**密钥详情页**的**别名管理**区域，可为密钥创建别名，同时可删除不需要的别名。详情请参见别名管理；

3）在用户主密钥详情页的**密钥版本**区域，可为对称密钥**设置轮转策略**，为非对称密钥手动**更新密钥版本**，同时可查看当前密钥的版本列表。详情请参见密钥版本管理。

对称密钥，设置轮转策略：

非对称密钥，创建密钥版本：



4.2.5. 别名管理

别名是用户主密钥的可选标识，同一个用户在一个地域中的别名具有唯一性。每个别名只能指向同地域的一个用户主密钥，但是每个用户主密钥可以绑定多个别名。

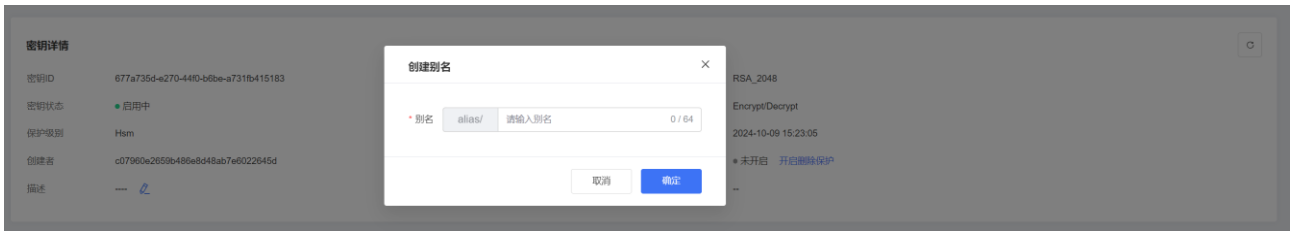
别名必须依附于用户主密钥存在。其特点如下：

- 一个用户主密钥下可以绑定多个别名，删除别名不会删除其关联的用户主密钥。
- 别名不可修改。您可以通过为一个用户主密钥创建新的别名，并且删除旧的别名来达到修改主密钥别名的目的。
- 可以调用 UpdateAlias 接口更改别名绑定的用户主密钥，而不会影响用户主密钥。
- 默认主密钥的别名不能删除和添加。

操作步骤-创建别名

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**。在密钥列表点击**密钥 ID**，进入**密钥详情页**。
4. 在**别名**区域，点击**创建别名**。

5. 填写别名，单击**确定**。



操作步骤-删除别名

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 在密钥列表点击**密钥 ID**，进入**密钥详情页**；
5. 在**别名**区域的别名列表，选择对应别名，点击**删除别名**，单击**确定**。

别名管理相关 API 接口

您可以通过调用别名管理的相关接口，实现别名的创建、删除、更新、查询等操作。

功能	API	描述
密钥别名管理	createAlias	创建密钥别名。
	updateAlias	更新密钥别名。
	deleteAlias	删除密钥别名。
	listAlias	列出云账号在本地域的所有别名。
	listAliasByUuid	列出与指定用户主密钥绑定的别名。

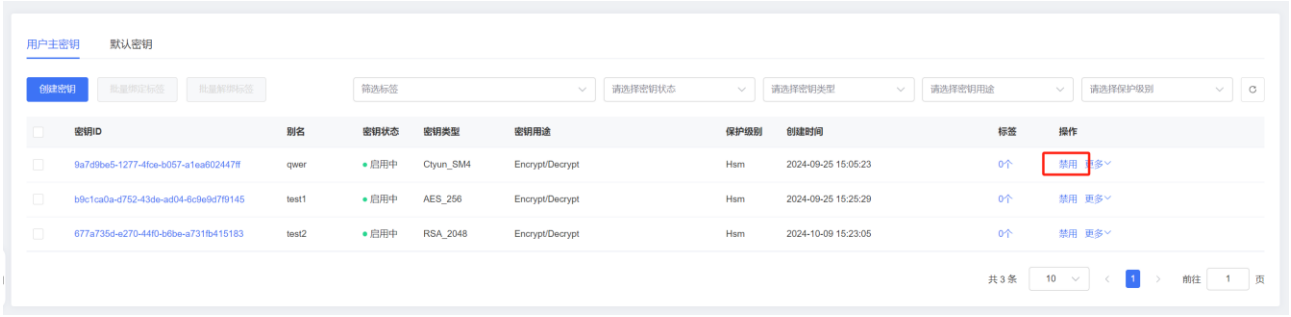
4.2.6. 启用禁用密钥

密钥创建完成后，默认为启用状态。您可以禁用密钥，被禁用的密钥无法用于加密和解密。

操作步骤

- **禁用**

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 定位待禁用的密钥，单击右侧操作列的**禁用**；



5. 在弹出的禁用密钥对话框，单击**确定**。

- **启用**

1. 找到待启用的密钥，单击右侧操作列的**启用**；

2. 在弹出的启用密钥对话框，单击**确定**。

4.2.7. 密钥版本管理

密钥常用于保护特定的数据，因此，数据的安全依赖于密钥的安全。您可以通过密钥版本化和定期轮转来加强密钥使用的安全性，实现数据保护的安全策略和最佳实践。

密钥版本概述

KMS 中的用户 CMK 支持多个密钥版本。每一个密钥版本是一个独立生成的密钥，同一个 CMK 下的多个密钥版本在密码学上互不相关。

- 对于对称密钥，密钥版本可通过自动轮转策略，由系统自动生成；
- 对于非对称密钥，可人工创建新的密钥版本。

设置自动轮转

对称密钥支持设置自动轮转，生成新的密钥版本。对称密钥版本分为主版本和非主版本：

主版本（Primary Key Version）

- 系统根据自动轮转策略，定期生成新的密钥版本，并自动设为主版本。
- 主版本是 CMK 的活跃加密密钥（Active Encryption Key）。每个 CMK 在任何时间点上有且仅有一个主版本。
- 调用 GenerateDataKey、Encrypt 等加密 API 接口时，KMS 使用指定 CMK 的主版本对明文进行加密。

非主版本（Non-primary Key Version）

- 非主版本是 CMK 的非活跃加密密钥（Inactive Encryption Key）。每个 CMK 可以有零到多个非主版本。非主版本历史上曾经是主版本，在当时被用作活跃加密密钥。

- 密钥轮转产生新的主版本后，KMS 不会删除或禁用非主版本，它们需要被用作解密数据。

创建密钥版本

由于公私钥使用场景的特殊性，KMS 不支持对非对称的用户主密钥进行自动轮转。可在指定用户主密钥中人工创建新的密钥版本，生成全新的一对公钥和私钥。

除此之外，和对称类型的用户主密钥不同，非对称的用于主密钥没有主版本（PrimaryKeyVersion）的概念，因此使用非对称密码运算的接口除需指定用户主密钥标志符（或别名）之外，还需指定密钥版本。

不适用范围

KMS 管理的以下类型的密钥不支持多个版本：

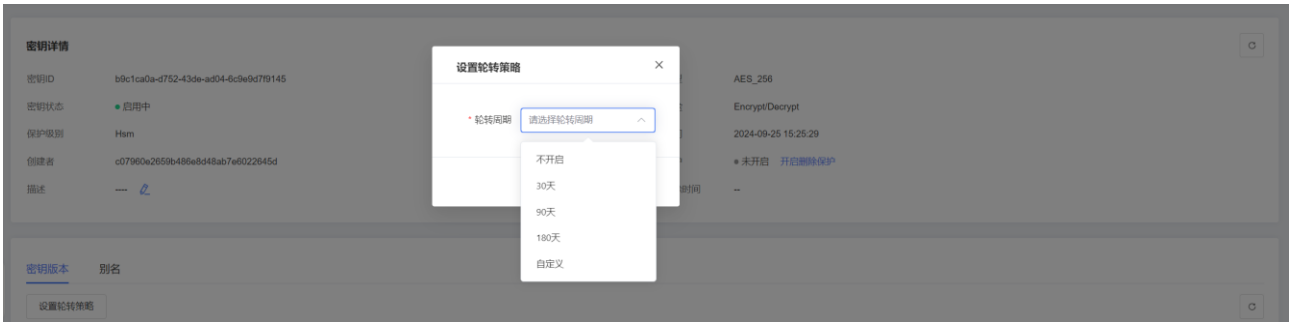
- 云产品的默认密钥：特定云产品托管在 KMS 上的、用于加密保护您的数据的默认密钥。这类密钥由特定云产品为用户代为管理，为您的数据提供最基本的加密保护。
- 用户自带密钥（BYOK）：您导入到 KMS 中的密钥。这类 CMK 的 Origin 属性为 External，KMS 不负责为用户生成密钥材料，无法自动发起轮转行为。更多信息，请参见导入密钥材料。

操作步骤

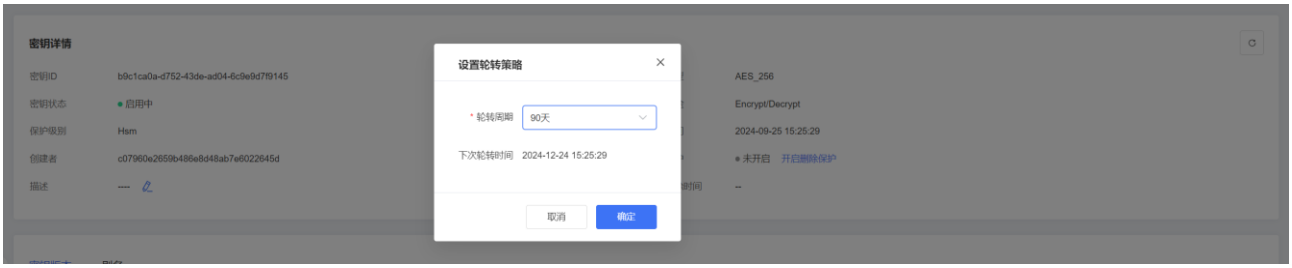
设置自动轮转（对称密钥）

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 定位待设置的对称密钥，点击**密钥 ID**，进入**密钥详情页**；
5. 在**密钥版本**区域，点击**设置轮转策略**；

6. 在**设置轮转策略**对话框，选择轮转周期，**30 天**、**90 天**、**180 天**，或**自定义天数**；



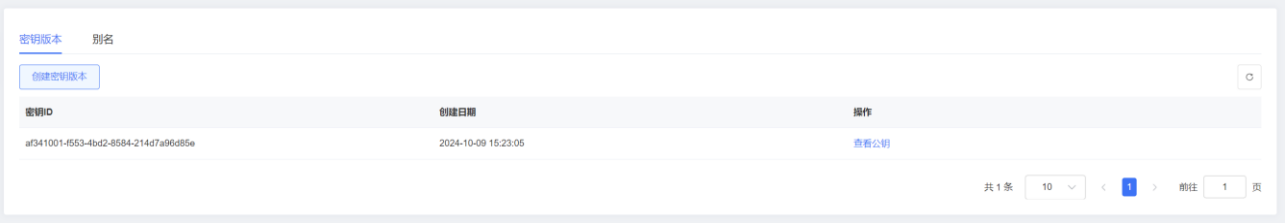
7. 设置了自动轮转策略后，将显示密钥下次轮转时间。点击**确定**完成设置；



8. 可通过相同的步骤更改轮转周期，也可取消轮转策略。

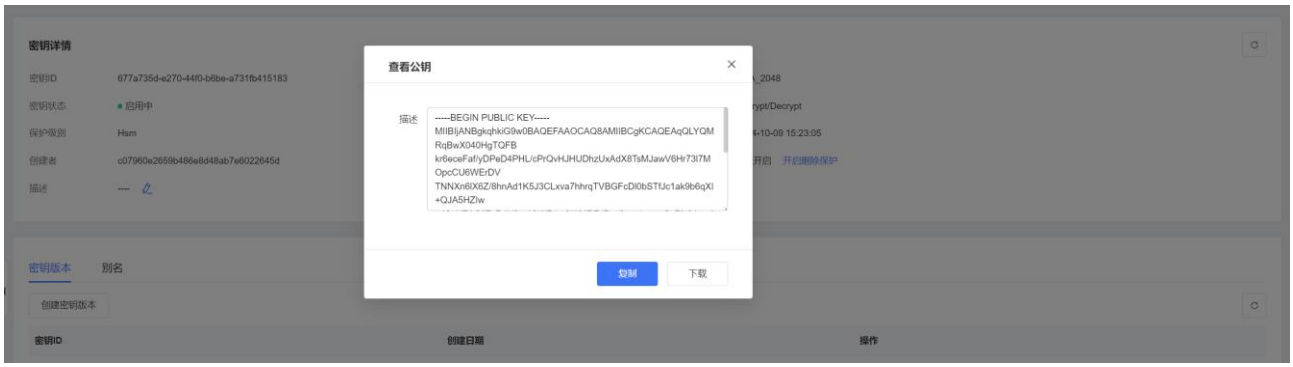
创建密钥版本（非对称密钥）

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 定位待设置的非对称密钥，点击**密钥 ID**，进入**密钥详情页**；
5. 在**密钥版本**区域，点击**创建密钥版本**；



6. 在弹出的对话框内，点击**确定**；

7. 在密钥版本列表，可查看密钥版本 ID、创建日期。点击**查看公钥**，在弹出的对话框，可**复制**或**下载**公钥。

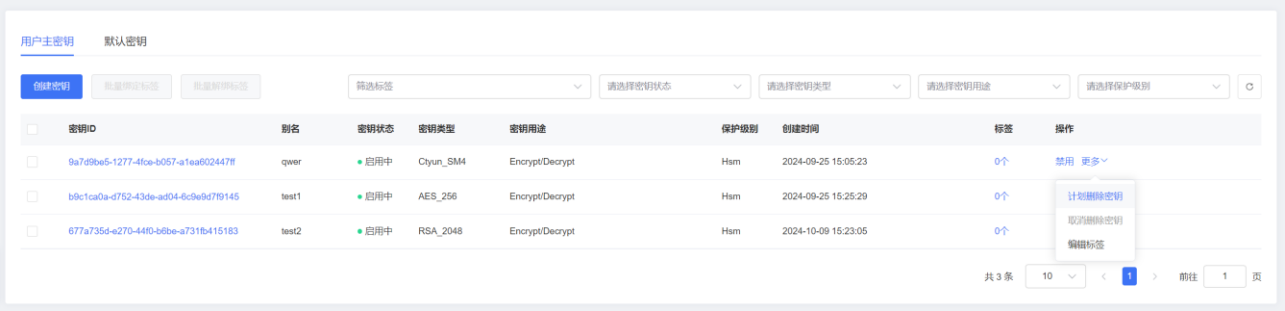


4.2.8. 删除密钥

用户主密钥（CMK）一旦删除，将无法恢复，使用该 CMK 加密的内容及产生的数据密钥也将无法解密。因此，对于 CMK 的删除，KMS 只提供计划删除的方式，而不提供直接删除的方式。如果不再使用 CMK，推荐您使用禁用密钥功能。

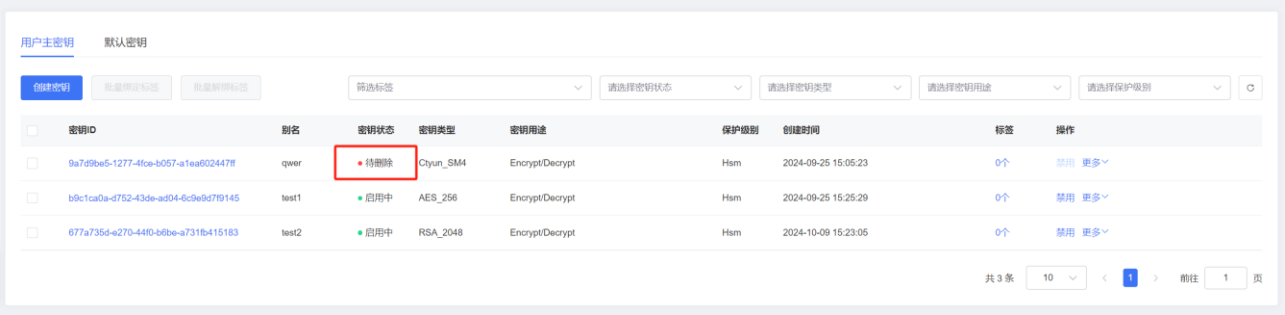
操作步骤

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 定位计划删除的密钥，在右侧操作列选择**更多>计划删除密钥**；

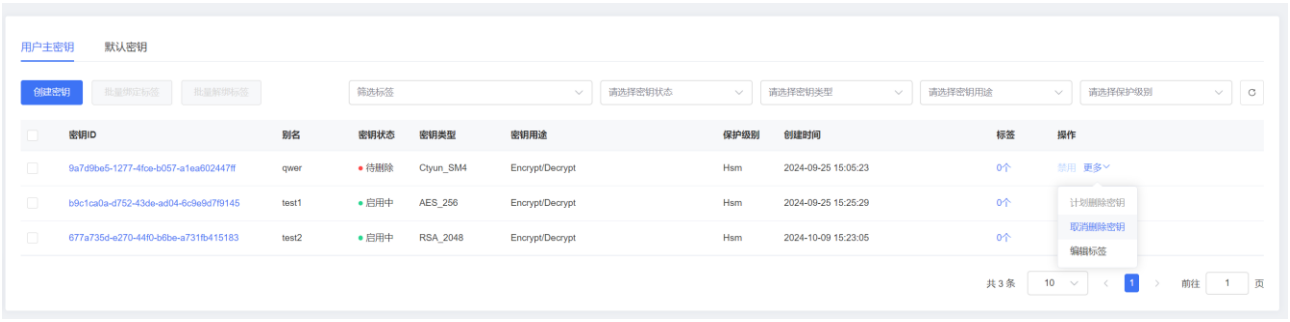


5. 在计划删除密钥对话框，填写预删除周期，点击**确定**。预删除周期可选值为：7~30 天；

6. 此时密钥状态由启用中变为**待删除**。处于待删除状态的密钥无法用于加密、解密和产生数据密钥；



7. 处于待删除状态的密钥，您可以通过在右侧操作列选择**更多 > 取消计划删除密钥**，撤销删除密钥的申请；



8. 在弹出的对话框，点击**确定**，即可取消计划删除，密钥恢复可用状态。

4.3. 对称密钥运算

4.3.1. 对称加密概述

对称加密是最常用的数据加密保护方式。KMS 提供了简单易用的接口，方便您在云上轻松实现数据加解密功能。

密钥管理服务支持主流的对称密钥算法并且提供足够的安全强度，保证数据加密的安全性。

KMS 支持的对称密钥类型

算法	密钥长度	密钥规格	保护级别
AES	256 比特	AES_256	• Software • HSM
SM4	128 比特	Ctyun_SM4	• HSM

对称密钥功能特性

KMS 生成的对称主密钥支持多个密钥版本，同时支持用户主密钥基于密钥版本进行自动轮转，您可以自定义密钥轮转的策略。为了满足特殊的安全合规要求，KMS 支持您使用自带密钥（BYOK）进行数据的加密保护。

功能	功能描述
自动轮转	• 支持设置自动轮转策略，生成新的密钥版本，并自动设为主版本（primaryKeyVersion），KMS 会使用主版本密钥实现加解密 • 密钥轮转产生新的主版本后，KMS 不会删除或禁用非主版本，他们需要被用作解密操作。
导入密钥材料（BYOK）	• 默认情况下，当创建 CMK 时，会由 KMS 生成密钥材料。也可以选择创建密钥材料来源为外部的密钥，将自带密钥材料导入到 CMK 中。 • 导入的密钥材料可以进行删除，也可以设置过期时间，在密钥材料过期后进行删除（CMK 不会被删除）。导入的密钥材料被删除后，可以再次导入相同的密钥材料使得 CMK 再次可用，因此您需要自行保存密钥材料的副本。 • 每个 CMK 只能拥有一个导入密钥材料。当您将一个密钥材料导入 CMK 时，CMK 将与密钥材料绑定，即便密钥材料已经过期或者被删除，也不能导入其他密钥材料。如果您需要轮换使用外部密钥材料的 CMK，只能创建一个新的 CMK 然后导入新的密钥材料。

对称密钥应用场景

KMS 生成的对称密钥支持如下数据加密方式，满足多样化的数据保护场景。

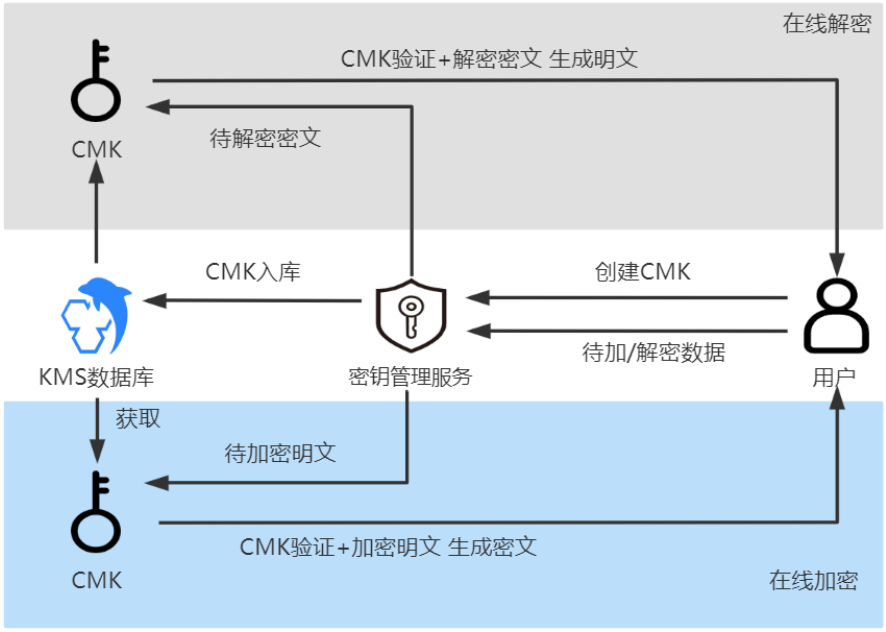
场景	场景描述
在线加密	• 适用于保护小型敏感数据（小于 6KB）的加解密，如密钥、证书、配置文件等。 • 用户的数据会通过安全信道传递到 KMS 服务端，服务端通过指定 CMK 完成加密和解密后，操作结果通过安全信道返回给用户。

场景	场景描述
信封加密	• 适用于海量数据的高性能加解密，如规模较大的对性能敏感的本地文件。 • 通过 KMS 生成数据密钥 DEK，并返回 DEK 明文及经指定 CMK 加密的 DEK 密文。用户使用数据密钥 DEK 明文在本地进行高效的加解密处理，然后将内存中的 DEK 明文销毁，将 DEK 密文及密文文件落盘存储。

4.3.2. 在线加密

敏感信息加密是密钥管理系统 KMS 核心的能力，适用于保护小型敏感数据（小于 6KB），如口令、证书、配置文件等。通过密钥管理服务 KMS 的在线加密 API，使用 用户主密钥（CMK）直接加密敏感数据信息，而非直接将明文存储，确保敏感数据安全。

场景示意图



操作流程（以证书加密为例）

1. 通过 KMS 控制台或者调用 CreateKey 接口，创建一个用户主密钥（CMK）；
2. 调用 KMS 服务的 Encrypt 接口，将明文证书加密为密文证书；
3. 将密文证书部署在服务器上；
4. 当服务器启动需要使用证书时，调用 KMS 服务的 Decrypt 接口将密文证书解密为明文证书。

相关 API

您可以调用以下 KMS API，轻松完成对数据的加密或解密操作。

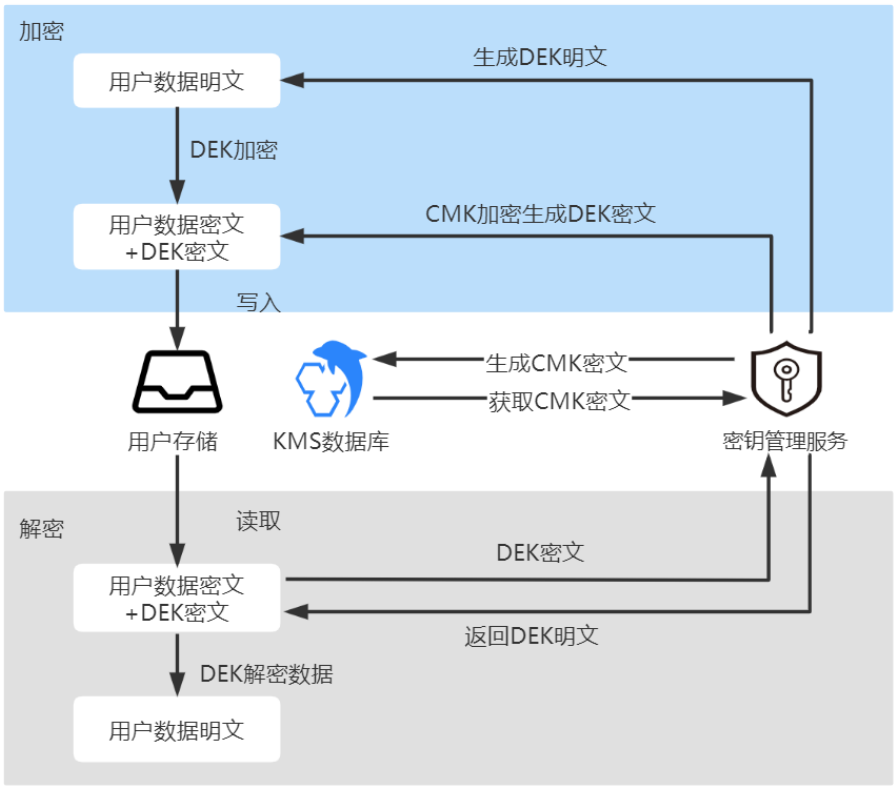
API 名称	说明
createKey	创建用户主密钥（CMK）。
encrypt	指定 CMK，直接输入明文数据，由 KMS 在线加密数据。
decrypt	解密由 encrypt 接口加密的数据，不需要指定 CMK 即可完成在线解密。

4.3.3. 信封加密

信封加密（Envelope Encryption）是一种应对海量数据的高性能加解密方案。这种技术不再使用用户主密钥（CMK）直接加密和解密数据，而是通过生成加密数据的数据密钥（DEK），将其封入信封中（即通过 CMK 加密）存储、传递和使用，由 KMS 确保数据密钥的随机性和安全性。

实际使用时，用户无需将大量业务数据上传至 KMS 服务端，直接通过离线的数据密钥在本地实现加解密，有效避免安全隐患，保证了业务加密性能的要求。

场景示意图



操作流程

信封加密

1. 通过 KMS 控制台或者调用 CreateKey 接口，创建一个用户主密钥（CMK）；
2. 调用 GenerateDataKey 接口创建一个数据密钥。KMS 会返回一个明文的数据密钥和一个经用户主密钥（CMK）加密的密文数据密钥；
3. 使用明文的数据密钥加密本地文件，产生密文文件，然后销毁内存中的明文数据密钥；
4. 用户将密文数据密钥和密文文件一同存储到持久化存储设备或服务中。

信封解密

1. 从本地文件中读取密文数据密钥；
2. 调用 KMS 服务的 Decrypt 接口，将密文数据密钥解密为明文数据密钥；
3. 用明文数据密钥为本地密文文件解密，再销毁内存中的明文密钥。

相关 API

您可以调用以下 KMS API，实现对本地数据的加密或解密操作。

API 名称	说明
createKey	创建用户主密钥（CMK）
generateDataKey	生成信封加密的数据密钥，返回数据密钥的明文和经过指定用户主密钥加密的密文
decrypt	解密由 generateDataKey 接口生成的数据密钥密文，不需要指定 CMK

4.4. 非对称密钥运算

4.4.1. 非对称密钥概述

相比对称加密，非对称密钥通常用于在信任程度不对等的系统之间，实现数字签名验签或者加密传递敏感信息。

非对称密钥由一对密钥组成，分别是公开密钥（public key，简称公钥）和私有密钥（private key，简称私钥）。公钥可以任意对外发布，私钥必须由用户自行严格秘密保管。非对称密钥具有双向性，即公钥和私钥中的任一个均可用作加密，此时另一个则用作解密。

密钥管理服务（KMS）支持主流的非对称密钥算法并且提供足够的安全强度，保证数据加密和数字签名的安全性。

KMS 支持的非对称密钥类型

算法	密钥规格	保护级别
RSA	RSA_2048	• Software • HSM
SM2	Ctyun_SM2	• HSM

非对称密钥功能特性

由于非对称密钥公、私钥使用场景的特殊性，KMS 不支持对非对称的用户主密钥进行自动轮转。您可以自主在指定用户主密钥中创建新的密钥版本，生成全新的一对公钥和私钥。

非对称密钥区分公钥运算和私钥运算，公钥主要用于数据加密和验签，私钥主要用于数字签名和数据解密。

功能	功能描述
创建密钥版本	• 支持自主创建新密钥版本，不支持设置自动轮转策略。 • 区别于对称密钥，非对称密钥无密钥主版本概念，则在调用非对称密码运算 API 接口时，在指定使用的用户主密钥（CMK）的同时，还需指定使用的密钥版本（keyVersion）。
公钥运算	• 大多数情况下，您可以调用 GetPublicKey 接口获取公钥，之后分发给公钥使用者。使用者在业务端通过 OpenSSL、Java JCE 等常用的密码运算库在本地进行加密、验签处理。 • 密钥管理服务（KMS）也提供公钥运算的非对称密钥加密接口（asymmetricEncrypt）和数字签名验签接口（asymmetricVerify），满足特定的业务需求。
私钥运算	• 由于私钥的不公开性，用户仅能通过调用 KMS 提供的私钥运算的产生数字签名接口（asymmetricSign）和非对称密钥解密接口（asymmetricDecrypt），实现签名、解密处理。

非对称密钥应用场景

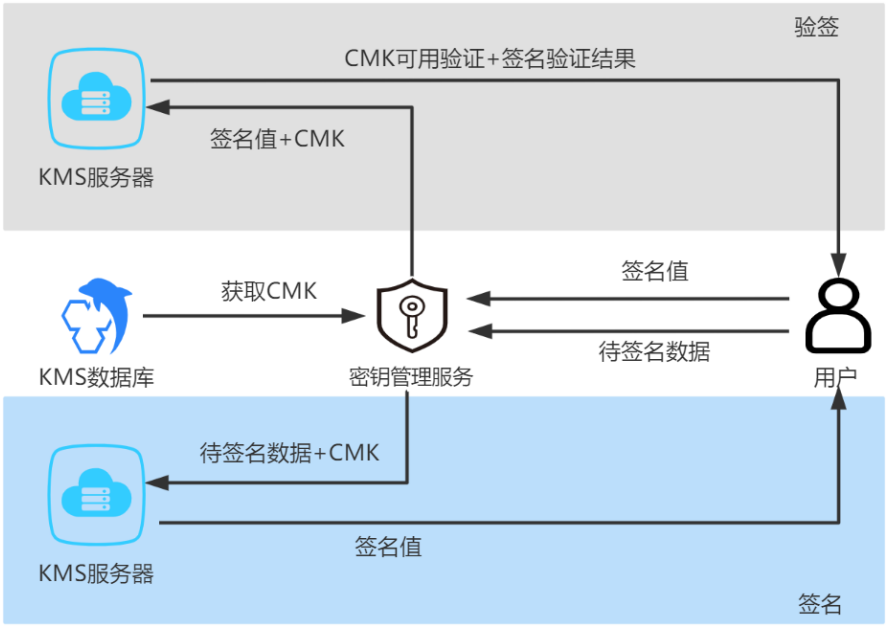
场景	场景描述
签名验签	• 数字签名技术是非对称加密算法的另一种典型应用。数字签名分为签名和验证两个过程，消息发送者使用私钥对数据签名，消息接收者使用公钥进行签名验证。 • 由于签名是使用私钥加密产生，而私钥不公开，这使得签名具有唯一的特征，广泛用于数据防篡改、身份认证等相关技术领域。
数据加解密	• 非对称密钥加密通信的过程类似于对称加密，区别在于需要使用公钥进行数据加密，使用私钥进行数据解密。 • 由于密文只有通过私钥才可以解密，而私钥是不公开的，所以即使由于传输介质的安全性比较低而导致信息泄露，拿到密文的人也无法将其破译，从而保证了敏感信息的安全。这种敏感信息传递的方式，被广泛用于各类密钥交换场景。

4.4.2. 签名验签

数字签名技术是非对称加密算法的另一种典型应用。数字签名分为签名和验证两个过程，消息发送者使用私钥对数据签名，消息接收者使用公钥进行签名验证。

由于签名是使用私钥加密产生，而私钥不公开，这使得签名具有唯一的特征，广泛用于数据防篡改、身份认证等相关技术领域。

场景拓扑图



操作流程

1. 信息发送者通过 KMS 控制台或者调用 CreateKey 接口，创建一个非对称的用户主密钥（CMK）；
2. 信息发送者通过调用 KMS 的 getPublicKey 接口获取到公钥，并将公钥分发给消息接收者；
3. 信息发送者通过调用 KMS 的 asymmetricSign 接口，使用创建的 CMK 私钥对需要传输的数据生成签名；
4. 信息发送者将签名和数据传递给信息接收者；
5. 信息接收者拿到签名和数据之后，在本地通过 gmssl、openssl、密码库、KMS 的国密 Encryption SDK 等验签方法，使用信息发送者分发的公钥进行验证。特殊需求场景下，也可调用 KMS 的 asymmetricVerify 接口，使用 CMK 进行签名校验。

相关 API

您可以调用以下 KMS API，完成对数据的签名验签处理。

API 名称	说明
createKey	创建用户主密钥（CMK）。
getPublicKey	获取非对称密钥的公钥，可用于离线验证数字签名，或者加密数据。
asymmetricSign	非对称密钥的私钥运算：产生数字签名。
asymmetricVerify	非对称密钥的公钥运算：验证私钥产生的数字签名。

4.4.3. 非对称密钥加解密

非对称密钥加密通信的过程类似于对称加密，区别在于需要使用公钥进行数据加密，使用私钥进行数据解密。

由于密文只有通过私钥才可以解密，而私钥是不公开的，所以即使由于传输介质的安全性比较低而导致信息泄露，拿到密文的人也无法将其破译，从而保证了敏感信息的安全。这种敏感信息传递的方式，被广泛用于各类密钥交换场景。

操作流程

1. 信息接收者通过 KMS 控制台或者调用 KMS 的 CreateKey 接口，创建一个非对称的用户主密钥（CMK）；
2. 信息接收者通过调用 KMS 的 getPublicKey 接口获取到公钥，并将公钥分发给消息发送者；
3. 信息发送者使用公钥在本地通过 OpenSSL 等方式对数据进行加密。特殊需求场景下，也可通过调用 KMS 的 asymmetricEncrypt 接口，使用 CMK 进行加密；
4. 信息发送者将密文数据传递给信息接收者；
5. 信息接收者拿到密文数据之后，可调用 KMS 的 asymmetricDecrypt 接口，使用私钥进行数据解密。

相关 API

您可以调用以下 KMS API，完成对敏感数据传输中的加解密处理。

API 名称	说明
createKey	创建用户主密钥（CMK）。
getPublicKey	获取非对称密钥的公钥，可用于离线验证数字签名，或者加密数据。
asymmetricEncrypt	非对称密钥的公钥运算：加密数据。
asymmetricDecrypt	非对称密钥的私钥运算：解密公钥加密的数据。

4.5. 应用接入点

4.5.1. 应用接入点概述

您开通密钥管理包周期服务后，需要访问 KMS 的 API 接口以进行密码运算等能力调用，完成数据加解密、签名验签、完整性校验等业务场景。密钥管理服务提供两种服务调用方式：

- OpenAPI（公网调用）
- 应用接入点（内网调用）

若要实现租户 VPC 内应用通过内网私密网络通道访问 KMS 服务，需要创建应用接入点，KMS 通过应用接入点完成网络通道的建立。

应用接入点提供访问控制机制，需要为使用该应用接入点的调用方创建身份凭证，在调用 KMS 服务时依据身份凭证完成身份校验。

> 说明：
> 1、当您的自建应用部署在同一地域但分布在多个 VPC 时，您需要为每个需要集成 KMS 的服务创建应用接入点，完成多个 VPC 内应用与 KMS 服务之间网络通道的建立。
> 2、当您的多个自建应用部署在同一个 VPC，若需要实现访问控制的独立，可以为每个应用单独创建应用接入点。
> 3、当前 KMS 提供 3 个免费的应用接入点额度。

建立网络通道

您需要创建应用接入点，建立应用所在的 VPC 与 KMS 服务端之间的网络通道。

- 创建应用接入点时需要指定 VPC，请明确调用 KMS 服务的应用所在 VPC。
- 创建应用接入点后，KMS 会生成 endpoint 地址，您需要通过该地址访问 KMS 服务。

> 说明：
> KMS 产品为 region 级服务，支持同一地域下的 VPC 内应用通过应用接入点访问 KMS 服务，暂不支持跨地域访问。

访问控制

应用接入点中提供访问控制机制，当您的自建应用需要访问 KMS 包周期版服务时，需要对其进行身份认证，您需要为调用方创建访问凭证（AK/SK）。

- 身份凭证用于对 KMS 资源访问者进行身份认证和行为鉴权。
- 当前 KMS 支持通过 AK/SK 的身份验证方式，其中包含 AK（AccessKey）和 SK（SecretKey）。

- 您可以通过 KMS 提供的 SDK 快速集成 KMS 服务，并在初始化 SDK 时导入所创建的 AK/SK。

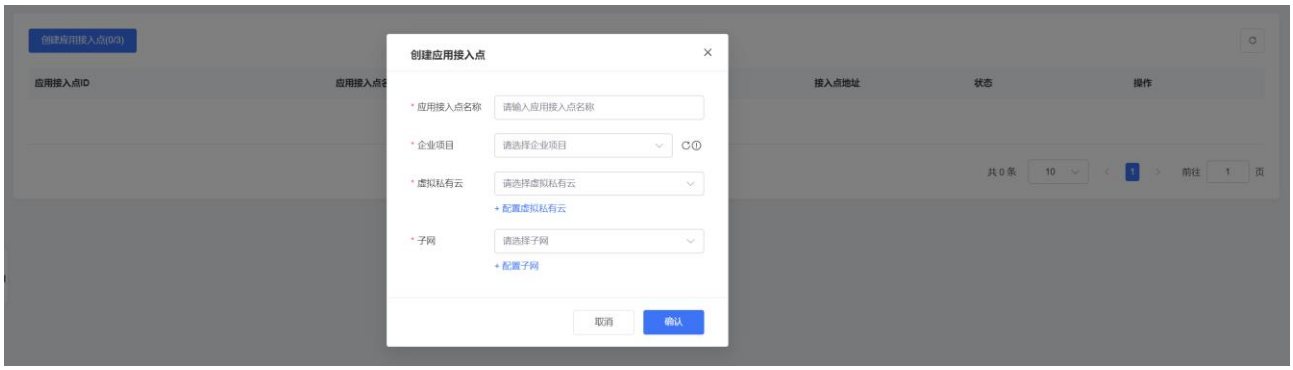
> 注意：
> 1、生成访问凭证（AK/SK）后，您需要立即在弹窗中复制或下载文件，关闭后不再支持下载。若您未能成功保存，可删除后重新创建；
> 2、如果访问凭证（AK/SK）泄露，会带来数据泄露风险，建议妥善保管。

4.5.2. 管理应用接入点

本文为您介绍如何通过 KMS 产品控制台管理应用接入点。

创建应用接入点

1. 登录密钥管理服务控制台。
2. 在页面最上方的导航栏选择服务所在的区域。
3. 进入“应用接入点”页面，点击**创建应用接入点**。



4. 在弹出的创建对话框，根据页面提示进行配置。

   配置项说明

配置项	说明
应用接入点名称	自定义名称。
企业项目	选择所属的企业项目。
虚拟私有云	选择当前地域下的虚拟私有云（Virtual Private Cloud，VPC）。 说明：此处对应的是您应用所在的虚拟私有云（VPC），即需要联通 KMS 服务的虚拟私有云（VPC）。

配置项	说明
子网	选择当前 VPC 内子网。

5. 创建成功后，您可以在应用接入点列表查看对应信息。

删除应用接入点

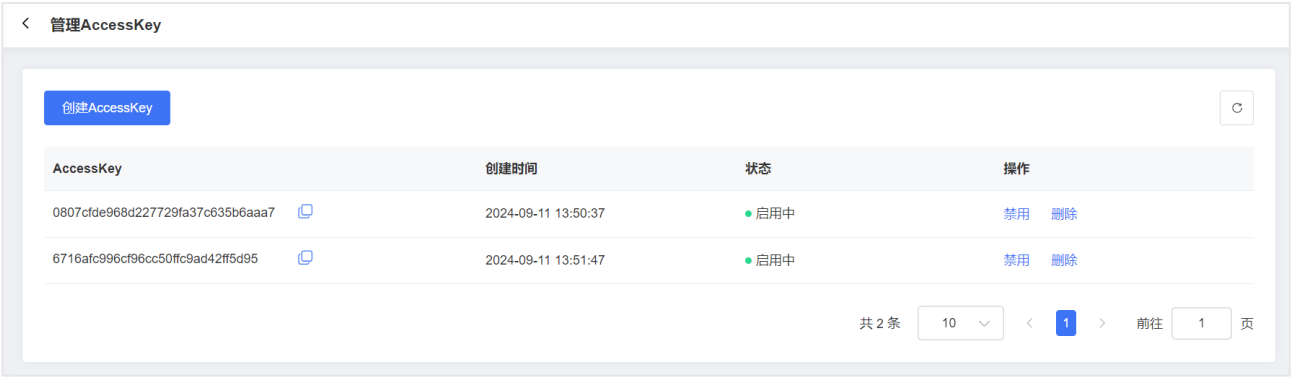
若您不再需要使用应用接入点资源，您可以在列表操作列，对该应用接入点进行删除。

创建访问凭证（AK/SK）

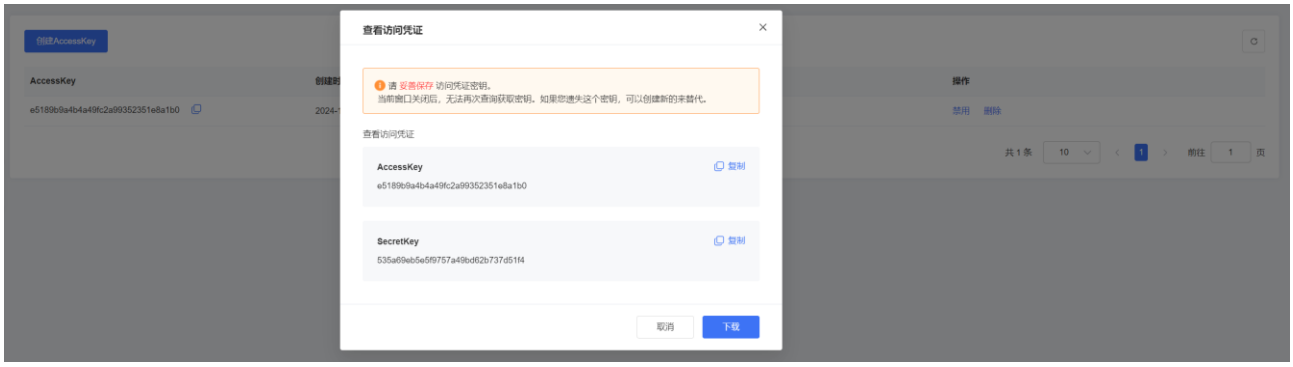
成功创建应用接入点后，您需要在该应用接入点下创建访问凭证（AK/SK），在调用 KMS 服务时，需要导入访问凭证（AK/SK），进行身份验证。

1. 在应用接入点列表页的操作列，点击“管理 AccessKey”。

2. 进入 AccessKey 列表页，点击“创建 AccessKey”。



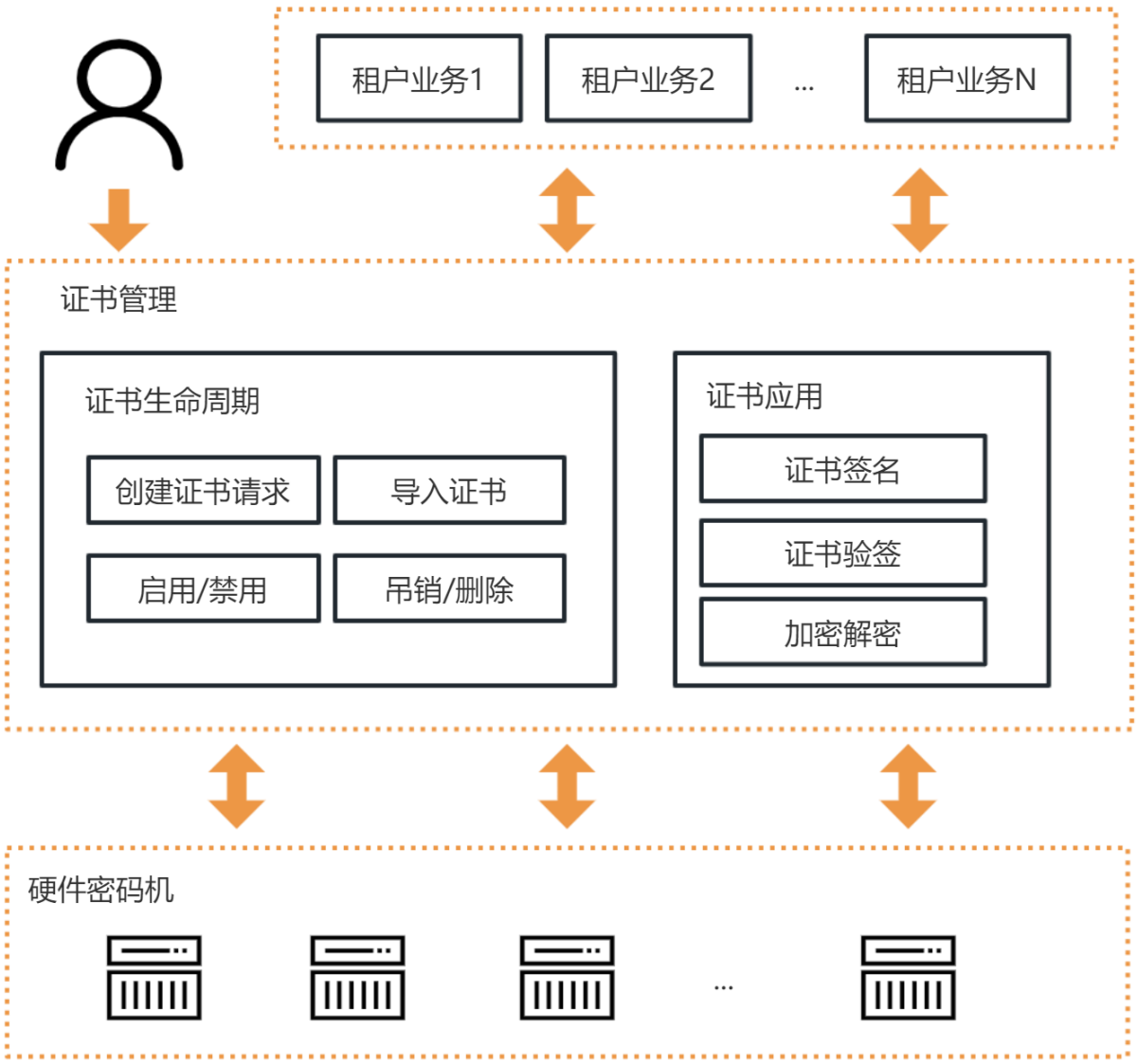
3. 访问凭证创建成功后，请在弹窗中复制或下载该访问凭证。



4.6. 证书管理

4.6.1. 证书管理概述

证书管理组件为您提供高可用、高安全的密钥和证书托管能力，您可以通过 KMS 提供的云原生接口实现签名验签运算。



证书生命周期管理

证书管理模块提供高可用、高安全的密钥和证书托管能力，您可以通过控制台或 API 集中管理证书。

功能	说明
证书托管	支持管理密钥和证书，可以生成证书请求、导入证书和证书链、启用/禁用证书、吊销或删除证书等。
密钥安全存储	证书管家使用托管密码机保障证书密钥的产生、存储安全。
API 便于集成	支持多个 API 接口，帮助在开发环境高效集成证书服务，快速进行产品部署，为您提供快速开发上线证书相关功能的能力。

证书运算 API

KMS 提供云原生的证书运算类 API，帮助在开发环境高效集成证书服务，快速实现证书调用。

功能	说明	参考文档
加密解密	证书公钥运算：使用指定证书加密数据。 证书私钥运算：使用指定证书解密数据。	证书公钥加密 证书私钥解密
签名验签	证书私钥运算：使用指定证书生成数字签名。 证书私钥运算：使用指定证书验证数字签名。	证书私钥签名 证书公钥验签

功能优势

- **密钥安全存储**

证书管家使用托管密码机保障证书密钥的产生、存储安全。

- **生命周期管理**

支持管理密钥和证书，可以生成证书请求、导入证书和证书链、检查证书链签名有效性，并检查证书有效性。

- **API 便于集成**

支持多个 API 接口，帮助您在开发环境高效集成证书服务，快速进行产品部署，为您提供快速开发上线证书相关功能的能力。

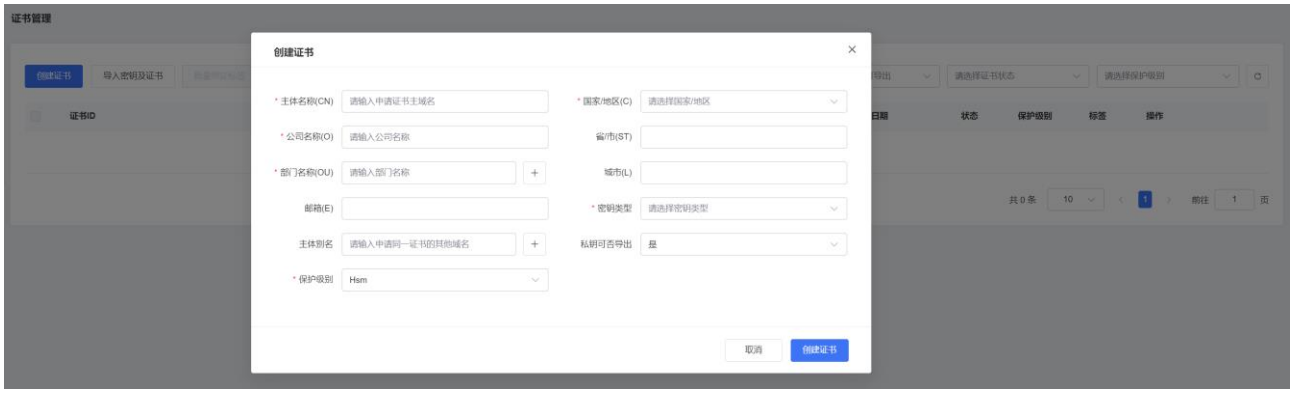
4.6.2. 创建证书

本文为您介绍在控制台创建密钥的操作步骤。

操作步骤

1. 登录密钥管理服务控制台。
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，点击**证书管理**，在证书列表页，点击创建证书；

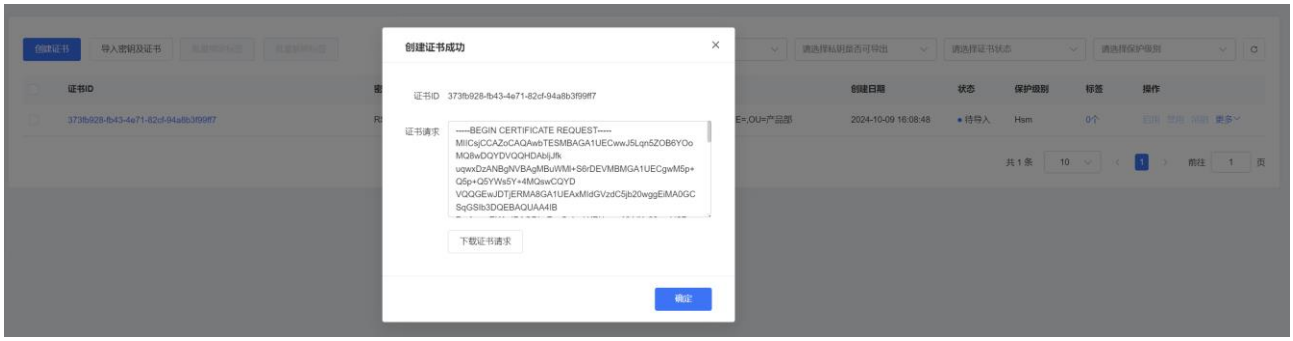
4. 在弹出的创建证书对话框，根据页面提示进行配置信息填写；



配置项说明

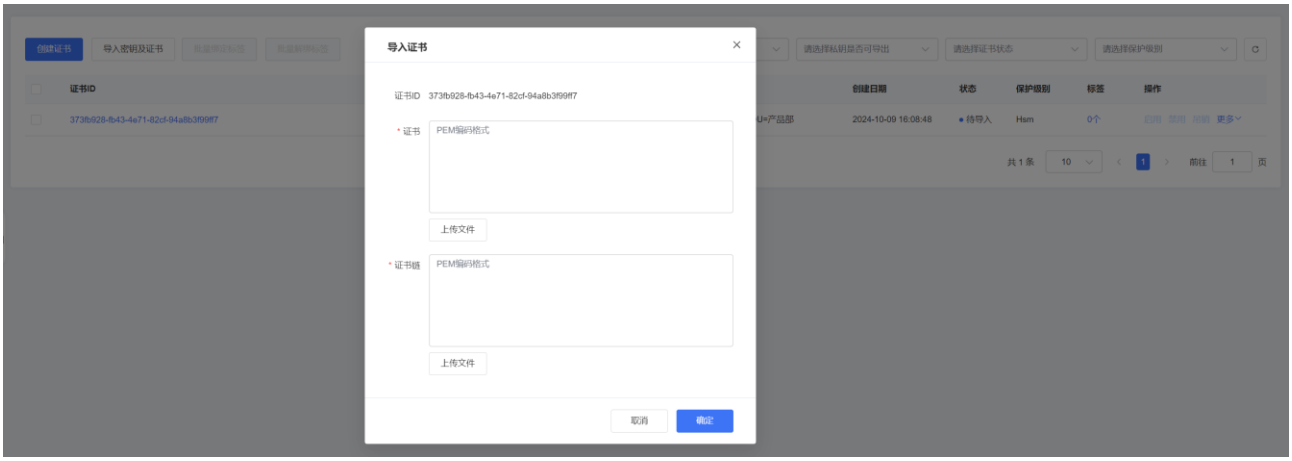
配置项	说明
主体名称（CN）	证书使用的主体名称。
国家/地区（C）	使用 ISO 3166-1 的二位国家代码，例如：CN 代表中国。
省/市（ST）	省、直辖市、自治区或特别行政区名称。
城市（L）	城市名称。
公司名称（O）	企业、单位、组织或机构的法定名称。
部门名称（OU）	部门名称。 单击右侧加号，可以添加多个部门名称，最多可添加 5 个。
邮箱（E）	证书持有者或管理者邮箱。
主体别名	当证书为 DV 证书时，可使用主体别名生成多域名证书请求。最多可添加 10 个。
密钥类型	取值：RSA_2048、SM2（企业版支持）
私钥可否导出	证书私钥是否需要导出使用。取值： 是：证书私钥需要导出使用。 否：证书私钥不需要导出使用。建议选择否，以便使用更高安全级别的密钥保护。
企业项目	选择证书归属的企业项目。默认为 default。 注：当前仅按需版本支持企业项目功能。

5. 证书信息填写完成后，点击**创建证书**，系统会返回证书 ID 及证书请求，点击**下载证书请求**，下载完成点击**确定**。



6. 下载 csr 格式的证书请求文件后，将其提交给 CA 机构，获取正式的证书和证书链；
7. 将证书和证书链导入证书管理服务，在证书列表页面，找到目标证书，点击**更多-导入证书**；

8. 在导入证书对话框，输入或上传 CA 机构颁发的证书和证书链，点击**确定**；



9. 导入证书成功后，证书状态为**启用中**，您可以使用证书进行签名验签等操作。

4.6.3. 导入密钥和证书

当您需要将其他证书系统的证书迁移并托管至天翼云证书管理服务中，需要先从其他证书系统导出PFX/PKCS12 格式的密钥文件，并将其导入到密钥管理服务-证书管理控制台。

操作步骤

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，点击**证书管理**，在证书列表上方，点击**导入密钥及证书**；

4. 在导入密钥及证书对话框，输入 PKCS12 文件保护口令，输入或上传 PKCS12 格式文件；

5. 完成填写后，点击确定。

4.6.4. 禁用、吊销或删除证书

证书成功导入证书管理控制台后，您可以调用实现签名验签。当您不再使用该证书时，可以通过控制台禁用、吊销或删除证书。

禁用证书

当您暂时无需使用证书时，可以禁用证书。禁用后的证书信息将保留，后续您可以根据需求再次启用证书。

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；
3. 在左侧导航栏，点击**证书管理**，在证书列表找到目标证书，点击操作列的**禁用**；

4. 在禁用证书对话框，点击确定。

吊销证书

当 CA 机构作废了证书时，您可以将证书管理控制台中证书的状态设置为已吊销。吊销后的证书信息将保留，后续您仅可以查看证书信息，但不可以启用或禁用证书。

1. 单击目标证书右侧操作列的**吊销**；
2. 在吊销证书对话框，单击确定。

删除证书

当您无需使用证书管家管理证书时，可以删除证书。删除证书前，请确保证书没有用于签名验签。

1. 在目标证书右侧操作列的**删除证书**；
2. 在删除证书对话框，单击**确定**。

4.7. 云产品服务端加密

4.7.1. 云产品集成 KMS 加密概述

密钥管理服务（KMS）与云硬盘、对象存储、弹性文件、关系型数据库 MYSQL 版等产品实现了服务端集成，在使用这些云服务时，可通过密钥管理服务实现对数据的加解密，并集中使用密钥管理服务（KMS）对密钥进行管理。

支持服务端加密的密钥类型

服务端加密支持选择默认密钥及用户自行创建的用户主密钥，具体可选择的密钥类型如下。

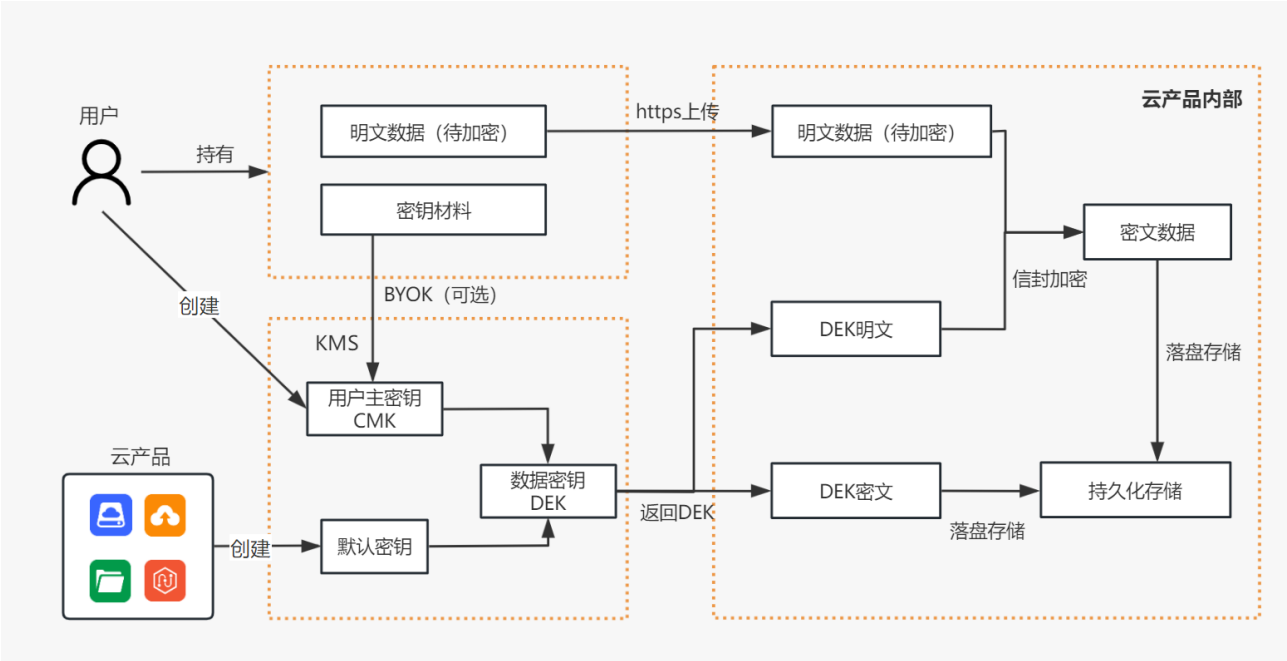
密钥创建者	密钥类型	密钥算法	服务版本
云产品	默认密钥	AES_256（默认）	按需版&包周期版
用户自行创建	用户主密钥-软件	AES_256	按需版
	用户主密钥-硬件	AES_256 SM4	按需版

- **默认密钥**
  - ✓ 系统为云产品自动创建的用于服务端加密的默认密钥，默认密钥与云产品对应，每个天翼云账号下的每个云产品在每个资源支持创建 1 个默认密钥。
  - ✓ 默认密钥的别名定义为 alias_<云产品代码>，例如 alias_ecs。
  - ✓ 默认密钥的密钥材料由 KMS 生成，不支持导入外部密钥材料，同时不支持自动轮转、启用/禁用、计划删除、自定义别名等操作。
- **用户主密钥**
  - ✓ 云产品加密时，可选用户在 KMS 服务中自建的用户主密钥，密钥类型为对称密钥，算法支持 AES_256、SM4，保护级别可选软件保护、硬件保护。
  - ✓ 用户主密钥按照 KMS 服务标准资费进行计费，请您确保账户余额充足，避免因 KMS 服务冻结导致云产品无法正常调用 KMS 服务进行加解密操作，云产品可能会出现异常
  - ✓ 用户主密钥支持计划删除，操作计划删除前请确保该密钥非未被用于云产品加密，避免删除后导致云产品无法正常加解密而出现异常。为避免误删，您可以为密钥开启删除保护功能。

注意：当前云产品加密仅支持选择按需版本中的自建用户主密钥，当前包周期版本中的用户主密钥暂不支持做云产品加密使用。若您为 2024 年 9 月 10 日之后购买了 KMS 包周期服务，您可选择使用默认密钥进行云产品加密。

云产品服务端加密的流程

通常情况下，云产品采用信封加密的机制实现对云产品数据的加密，即通过 KMS 生成数据密钥，并应用数据密钥在云产品服务端完成加解密操作，并将数据密钥密文及数据密文落盘存储，实现流程如下示意图。



1. 用户在 KMS 中创建一个用户主密钥，或由云产品触发创建默认密钥。
2. 云产品调用 KMS 的 GenerateDataKey 接口请求数据密钥。
3. KMS 返回数据密钥，包含数据密钥明文和数据密钥密文。其中数据密钥密文，是由指定的密钥加密数据密钥明文生成的。
4. 云产品使用数据密钥明文加密数据明文，并将数据密钥密文（由 KMS 使用密钥加密）与数据密文（由云产品使用数据密钥加密）一同写入持久化存储介质中。

4.7.2. 支持 KMS 服务端加密的云产品

当前天翼云 KMS 已与部分云产品集成，为云产品提供服务端加密功能，您可以在云产品控制台一键开启加密功能，加解密过程透明无感知。

存储

产品名称	描述	相关文档
云硬盘	当您的业务因为等保合规或安全要求等原因，需要对存储在云硬盘上的数据进行加密保护时，您可以在创建云硬盘时勾选加密选项，即可对新创建的云硬盘进行加密。 在创建加密云硬盘并将其挂载到实例后，以下数据都将关联此密钥并进行加密： • 云硬盘中的静态数据 • 云硬盘和实例间传输的数据（实例操作系统内的数据不加密） • 通过加密云硬盘创建的快照	管理加密云硬盘
对象存储	对象存储（简称 ZOS）在数据写入数据中心内的磁盘之前，支持在对象级别上应用数据加密的保护策略，并在访问数据时自动解密。加密和解密这一操作过程都是在服务端完成，这种服务端加密功能可以有效保护静态数据。	服务端加密
弹性文件	在创建文件系统时可以根据实际需要选择是否开启加密服务，无须授权，选择开启即可对新创建的文件系统进行加密。	加密

数据库

产品名称	描述	相关文档
关系数据库 MySQL 版	关系数据库 MySQL 版服务支持设置透明数据加密 TDE，在数据落盘时对数据进行加密，从而保证数据的安全性，用户业务对此加密过程无感知。	设置透明数据加密 TDE

4.8. 权限管理

天翼云提供统一身份认证（Identity and Access Management，简称 IAM）服务，是提供用户进行权限管理的基础服务，可以帮助您安全的控制云服务和资源的访问及操作权限。您通过 IAM 服务定义企业项目、创建子用户，轻松实现 IAM 子用户对 KMS 资源的访问控制、权限分配等。

KMS 权限管理

默认情况下，主账号创建的 IAM 用户没有任何权限，需要将其加入特定的用户组，并给用户组授予 KMS 产品的权限策略（包括系统策略和自定义策略），授权后 IAM 用户就能获得策略中定义的 KMS 产品的使用权限。

KMS 产品支持企业项目管理，若您需要对 KMS 服务中的资源进行分组和管理，形成逻辑隔离，您可以创建企业项目，并将资源划分至不同的企业项目中，不同的企业项目可以绑定不同的用户组，并给用户组授予 KMS 产品的权限策略（包括系统策略和自定义策略），从而实现对特定资源的授权。

KMS 权限配置

IAM 权限管理：进入天翼云 IAM 权限配置界面（https://iam.ctyun.cn/），可以进行 IAM 用户及用户组的创建，并在用户组列表中点击“授权”，为用户组添加 KMS 产品对应的权限策略。KMS 权限策略分为系统策略和自定义策略，系统策略默认提供，您也可以在“策略管理”界面创建自定义策略。用户组授权记录均可在“授权管理”界面查看并管理。

企业项目管理：进入天翼云 IAM 权限配置界面（https://iam.ctyun.cn/），在“企业项目”界面可以创建并管理企业项目，创建企业项目后可进行资源的迁入迁出，绑定用户组，并给用户组授予 KMS 产品的权限策略（包括系统策略贺自定义策略）。

KMS 权限及授权项说明

策略包含系统策略和自定义策略，如果系统策略不满足授权要求，管理员可以创建自定义策略，并通过给用户组授予自定义策略来进行精细的访问控制。策略支持的操作与 API 相对应，授权项列表说明如下：

- 权限：允许或拒绝 IAM 用户某项操作；
- 对应 API 接口：权限策略所作用的实际调用接口；
- 授权项：授权操作对应的权限三元组，创建自定义策略时，支持可视化 JSON 视图写入权限三元组实现策略配置；
- 权限类型：授权操作对应的读写类型。

KMS 支持的授权项

- 密钥管理

权限	对应 API 接口	授权项	读写类型
服务开通	——	kms:kmsService:open	写
创建密钥	/v1/cmkManage/createKey	kms:cmk:create	写
启用密钥	/v1/cmkManage/enableKey	kms:cmk:enable	写
禁用密钥	/v1/cmkManage/disableKey	kms:cmk:disable	写

计划删除密钥	/v1/cmkManage/scheduleKeyDeletion	kms:cmk:delete	写
取消计划删除密钥	/v1/cmkManage/cancelKeyDeletion	kms:cmk:undelete	写
更新密钥描述	/v1/keyManage/updateKeyDescription	kms:cmk:update	写
查看密钥列表	/v1/keyManage/listAliasKeys	kms:cmk:list	读
查看密钥详情	/v1/keyManage/describeKey	kms:cmk:describe	读
开启删除保护	/v1/cmkManage/deleteProtect	kms:cmk:deleteProtect	写
取消删除保护	/v1/cmkManage/cancelDeleteProtect	kms:cmk:cancelDeleteProtect	写
获取导入密钥材料参数	/v1/importKey/getParametersForImport	kms:cmk:getParameters	写
导入密钥材料	/v1/importKey/importKeyMaterial	kms:cmk:importMaterial	写
删除密钥材料	/v1/importKey/deleteKeyMaterial	kms:cmk:deleteMaterial	写
设置/更新轮转策略	/v1/versionControl/updateRotationPolicy	kms:cmk:updateRotation	写
创建密钥版本	/v1/versionControl/createKeyVersion	kms:cmk:createVersion	写
列出主密钥所有密钥版本	/v1/versionControl/listKeyVersions	kms:cmk:listVersions	读
查看指定密钥版本信息	/v1/versionControl/describeKeyVersion	kms:cmk:describeVersion	读
创建别名	/v1/keyName/createAlias	kms:cmk:createAlias	写
删除别名	/v1/keyName/deleteAlias	kms:cmk:deleteAlias	写

更新别名（非控制台功能）	/v1/keyName/updateAlias	kms:cmk:updateAlias	写
列出与指定密钥绑定的别名	/v1/keyName/listAliasByUuid	kms:cmk:listAliasByUuid	读
列出所有别名（非控制台功能）	/v1/keyName/listAlias	kms:cmk:listAlias	读
在线加密	/v1/keyCompute/encrypt	kms:cmk:encrypt	写
产品数据密钥（信封加密）	/v1/keyCompute/generateDataKey	kms:cmk:generateDataKey	写
产生无明文返回值的数据密钥（信封加密）	/v1/keyCompute/generateDataKeyWithoutPlaintext	kms:cmk:generateDataKeyWithoutPlaintext	写
导出数据密钥	/v1/keyCompute/exportDataKey	kms:cmk:exportDataKey	写
产生并导出数据密钥	/v1/keyCompute/generateAndExportDataKey	kms:cmk:generateAndExportDataKey	写
解密	/v1/keyCompute/decrypt	kms:cmk:decrypt	写
转加密	/v1/cmkManage/reEncrypt	kms:cmk:reEncrypt	写
产生数字签名	/v1/asymmetric/asymmetricSign	kms:cmk:asymmetricSign	写
验证签名	/v1/asymmetric/asymmetricVerify	kms:cmk:asymmetricVerify	写
非对称密钥加密	/v1/asymmetric/asymmetricEncrypt	kms:cmk:asymmetricEncrypt	写
非对称密钥解密	/v1/asymmetric/asymmetricDecrypt	kms:cmk:asymmetricDecrypt	写
获取非对称密钥公钥	/v1/asymmetric/getPublicKey	kms:cmk:getPublicKey	写

- 证书管理

权限	对应 API 接口	授权项	读写类型
创建证书 csr	/v1/manageCertificate/createCertificate	kms:cert:create	写
导入证书	/v1/manageCertificate/importCertificate	kms:cert:import	写
查看证书列表	/manageCertificate/listCertificate	kms:cert:list	读
查询证书信息	/v1/manageCertificate/describeCertificate	kms:cert:describe	读
更新证书状态	/v1/manageCertificate/updateCertificateStatus	kms:cert:update	写
获取证书	/v1/manageCertificate/getCertificate	kms:cert:get	写
导出证书私钥	/v1/manageCertificate/exportCertifiicatePrivatkey	kms:cert:exportPrivatkey	写
删除证书	/v1/manageCertificate/deleteCertificate	kms:cert:delete	写
证书私钥签名	/v1/certificateCompute/certificatePrivateKeySign	kms:cert:privateKeySign	写
证书公钥验签	/v1/certificateCompute/certificatePublicKeyVerify	kms:cert:publicKeyVerity	写
证书公钥加密	/v1/certificateCompute/certificatePublicKeyEncrypt	kms:cert:publicKeyEncrypt	写
证书私钥解密	/v1/certificateCompute/certificatePrivateKeyDecrypt	kms:cert:privateKeyDecrypt	写

权限	对应 API 接口	授权项	读写类型
生成随机数	/v1/certificatecompute/getRandom	kms:cert:getRandom	写

5. API 参考

5.1. API 概览

5.1.1. 概述

本说明提供了密钥管理 API 的描述、语法、参数说明及示例等内容。

5.1.2. API 概览

本文列出了密钥管理服务 KMS（Key Management Service）提供的 API 接口及相关描述。

密钥管理接口

密钥管理接口用于密钥的创建、属性修改以及生命周期管理等。

功能	API	描述
密钥托管	createKey	创建用户主密钥
	enableKey	启用密钥
	disableKey	禁用密钥
	scheduleKeyDeletion	计划删除
	cancelKeyDeletion	取消计划删除
	updateKeyDescription	更新密钥描述
	describeKey	查看密钥详情
	listKeys	查询密钥列表
密钥别名管理	createAlias	创建密钥别名
	updateAlias	更新密钥别名
	deleteAlias	删除密钥别名

	listAlias	列出云账号在本地域的所有别名
	listAliasByUuid	列出与指定用户主密钥绑定的别名
导入外部密钥材料	getParametersForImport	创建外部密钥（BYOK）时，获取导入主密钥的材料。
	importKeyMaterial	创建外部密钥（BYOK）时，已获取导入主密钥的材料后，将密钥材料导入到用户主密钥中。
	deleteKeyMaterial	删除主密钥材料。针对导入的外部密钥（BYOK），可以直接删除导入的密钥材料，删除密钥材料后的用户主密钥状态为等待导入。
密钥版本管理	describeKeyVersion	查看一个密钥版本信息。
	listKeyVersions	列出用户主密钥的所有密钥版本
	updateRotationPolicy	更新对称密钥的轮转策略。如果配置自动轮转，KMS将周期性自动生成新的密钥版本
	createKeyVersion	创建新密钥版本，适用于非对称密钥。

密码运算接口

密码运算接口用于对数据进行密码运算，例如加密和解密、签名验签等。

API	说明
encrypt	在线加密，使用指定用户主密钥加密数据，用于少量数据（不多于 6KB）的在线加密。
generateDataKey	生成信封加密的数据密钥，返回数据密钥的明文和经过指定用户主密钥加密的密文
generateDataKeyWithoutPlaintext	生成信封加密的数据密钥，返回经指定用户主密钥加密的密文
exportDataKey	导出数据密钥，返回经指定公钥加密的数据密钥的密文
generateAndExportDataKey	产生并导出数据密钥，生成信封加密的数据密钥，返回经指定用户主密钥加密的密文和经指定公钥加密的密文。

API	说明
decrypt	解密 Encrypt 或 GenerateDataKey 接口产生的密文，不需要指定用于解密的用户主密钥
reEncrypt	对密文进行转加密，即先解密密文，然后将解密得到的数据或者数据密钥使用新的主密钥再次进行加密，返回加密结果。待转加密的密文可以为对称加密或非对称加密返回的密文数据。
asymmetricSign	非对称密钥的私钥运算：产生数字签名
asymmetricVerify	非对称密钥的公钥运算：验证私钥产生的数字签名
asymmetricEncrypt	非对称密钥的公钥运算：加密数据。
asymmetricDecrypt	非对称密钥的私钥运算：解密公钥加密的数据。
getPublicKey	获取非对称密钥的公钥，可用于离线验证数字签名，或者加密数据

5.1.3. 状态码

本产品接口请求的返回状态码列表，包含正常和异常状态码、状态码对应状态信息、描述等。

API 名称	错误码	状态信息	描述
400	KMS.0102	CmkUuid cannot be empty	主密钥 id 不能为空
400	KMS.0103	AliasName cannot be empty	别名不能为空
400	KMS.0104	Digest cannot be empty	待签名值不能为空
400	KMS.0105	KeyVersionId cannot be empty	请选择主密钥版本
400	KMS.0106	Sign algorithm cannot be empty	请选择签名算法
400	KMS.0107	Plaintext can notbe empty	请输入需要加密的明文
400	KMS.0108	Ciphertextblob cannot be empty	请输入待解密密文

API 名称	错误码	状态信息	描述
400	KMS.0109	Reencrypt cmkuuid cannot be empty	请选择转加密主密钥
400	KMS.0110	Asymmetric encrypt need sourceuuid	非对称加密需选择密文对应的主密钥 id
400	KMS.0111	Asymmetric encrypt need sourceKeyVersion	非对称加密需选择密文对应的主密钥版本
400	KMS.0112	DestinationUuid cannot be empty	转加密主密钥不能为空
400	KMS.0113	Protectionlevel cannot be empty	保护等级不可为空
400	KMS.0114	Algorithm cannot be empty	加密算法不能为空
400	KMS.0115	Regionid cannot be empty	资源池 id 不能为空
400	KMS.0116	Enableautoatocrotatiom cannot be empty	请选择是否进行密钥轮转
400	KMS.0117	Keyusage can not be empty	请选择密钥用途
400	KMS.0118	Rotationinterval cannot be empty	请选择密钥轮转周期
400	KMS.0119	Nextrotationdate is empty	下次轮转时间为空
400	KMS.0120	Algorithm for import cannot be empty	选择导入密钥加密算法
400	KMS.0121	KeySpec for publickey cannot be empty	选择公钥类型
400	KMS.0122	Pendingwindowindays cannot be empty	选择延迟删除时间
400	KMS.0123	Token cannot be empty	token 为空
400	KMS.0124	Encryptenkeymeterial cannot be empty	密钥材料不能为空

API 名称	错误码	状态信息	描述
400	KMS.0125	Sign value cannot be empty	请输入签名值
400	KMS.0126	Keyspec cannot be empty	加密算法不能为空
400	KMS.0127	Userid cannot be empty	用戶 id 不能为空
400	KMS.0128	Request JSON format is illegal	请求 JSON 格式非法
400	KMS.0129	The request message is too long	请求消息太长
400	KMS.0201	Alias already exists	别名已存在
400	KMS.0202	The default CMK cannot create an alias	默认主密钥不能创建别名
400	KMS.0203	The selection CMK is not available	选择主密钥不可用
400	KMS.0204	The selected CMK is not available for signature and verification	选择的主密钥不可用于签名验签名
400	KMS.0205	The selected CMK is not available for encryption and decryption	选择的主密钥不可用于加密解密
400	KMS.0206	The CMK is disabled	密钥不是可用状态
400	KMS.0207	The CMK cannot be enabled	密钥不可启用
400	KMS.0208	The Cmk has deletion task	密钥已有删除计划
400	KMS.0209	The CMK has no deletion task	密钥无删除计划
400	KMS.0210	The CMK cannot be imported	密钥不可导入
400	KMS.0211	The CMK does not support automatic rotation	该密钥不支持自动轮转
400	KMS.0212	The selection algorithm is not supported	选择算法不支持

API 名称	错误码	状态信息	描述
400	KMS.0213	The CMK is not a symmetric key	密钥不是对称密钥
400	KMS.0214	The CMK is not a asymmetric key	密钥不是非对称密钥
400	KMS.0215	Wrong algorithm format	算法格式错误
400	KMS.0216	Wrong keyusage type	密钥用途类型错误
400	KMS.0217	Wrong input of parameter format! Please enter true / false	参数格式输入有误！请输入true/false
400	KMS.0218	Wrong protectionlevel	保护等级错误
400	KMS.0219	Wrong ciphertext	密文有误
400	KMS.0220	Key length input error,this algorithm only supports AES_256	密钥长度输入错误，该算法仅支持 AES_256
400	KMS.0221	Key length input error,this algorithm only supports SM4_128	密钥长度输入错误，该算法仅支持 SM4_128
400	KMS.0222	Key length input error,this algorithm only supports RSA_2048	密钥长度输入错误，该算法仅支持 RSA_2048
400	KMS.0223	This algorithm only supports AES/SM4	目前仅支持 AES/SM4 算法
400	KMS.0224	Wrong rotationlevel format	轮转周期格式不合法
400	KMS.0225	Asymmetric CMK cannot be imported	非对称密钥不可导入
400	KMS.0226	Wrong pendingwindowindays format	删除时间不合法

API 名称	错误码	状态信息	描述
400	KMS.0227	Unsupported keyspec	不支持的密钥类型
400	KMS.0228	The number of the CMK rotation of the user has reached the upper limit	该用户密钥轮转次数已达上限!
400	KMS.0229	Failed to execute key rotation. The key has been rotated within 7 days!	执行密钥轮转失败,该密钥 7 天内已执行过轮转!
400	KMS.0230	The current CMK is in the state of pause rotation. Please enable the master key or cancel the plan deletion before operation!	当前密钥处于暂停轮转状态,请启用主密钥或取消计划删除后再操作!
404	KMS.0301	The CMK does not exist or has been deleted	密钥不存在或已删除
404	KMS.0302	The target CMK does not exist	目标主密钥不存在
404	KMS.0303	The target alias does not exist	目标别名不存在
401	KMS.0401	Token validation failed	token 验证失败
401	KMS.0402	Sign wrong	AKSK 签名错误
401	KMS.0403	Time format wrong	时间格式错误
401	KMS.0404	Out of time	超时
500	KMS.0501	There are no rotation tasks to be performed	没有待执行的轮转任务
500	KMS.0502	Failed to delete alias	删除别名失败
500	KMS.0503	Key initialization failed	密钥初始化失败
500	KMS.0504	Key decryption failed	密钥解密失败
500	KMS.0505	HSM failed to create user key	密码机创建用户密钥失败

API 名称	错误码	状态信息	描述
500	KMS.0506	Failed to create delay delete task	创建延时删除任务失败
500	KMS.0507	Failed to send MQ for rotation task	轮转任务发送 mq 失败
500	KMS.0508	Delay delete task sending MQ failed	延时删除任务发送 mq 失败
500	KMS.0509	Failed to send MQ for delete key material task	删除密钥材料任务发送 mq 失败
500	KMS.0510	Failed to create default CMK	创建默认主密钥失败
500	KMS.0511	Failed to get random key	获取随机密钥失败
500	KMS.0512	Failed to create data key	创建数据密钥失败
500	KMS.0513	Data key decryption failed	数据密钥解密失败
500	KMS.0514	CMK encryption failed	用戶主密钥加密失败
500	KMS.0515	Failed to create CMK	创建用户密钥失败
500	KMS.0516	Key rotation failed!	执行密钥轮转失败！
500	KMS.0517	Failed to execute key rotation policy	执行密钥轮转策略失败

5.2. 如何调用 API

5.2.1. 终端节点

终端节点（Endpoint）即调用 API 的请求域名，不同服务的终端节点不同。

本产品终端节点为： kms-global.ctapi-test.ctyun.cn

5.2.2. 构造请求

本节介绍 REST API 请求的组成。

6.2.2.1. 请求的 URI

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

参数	描述	是否必选
URI-scheme	用于传输请求的协议，当前所有 API 均采用 **HTTPS** 协议。	是
Endpoint	当前资源池或者通用的域名	是
resource-path	资源路径，也即 API 访问路径。从具体 API 的 URI 模块获取，例如“获取用户 1 信息”API 的 resource-path 为“/users/1”。	是
query-string	查询参数，是可选部分，并不是每个 API 都有查询参数。查询参数前面需要带一个“?”，形式为“参数名=参数取值”，例如“?userID=1”，表示查询用户 ID 为 1 的数据。	是

6.2.2.2. 请求方法

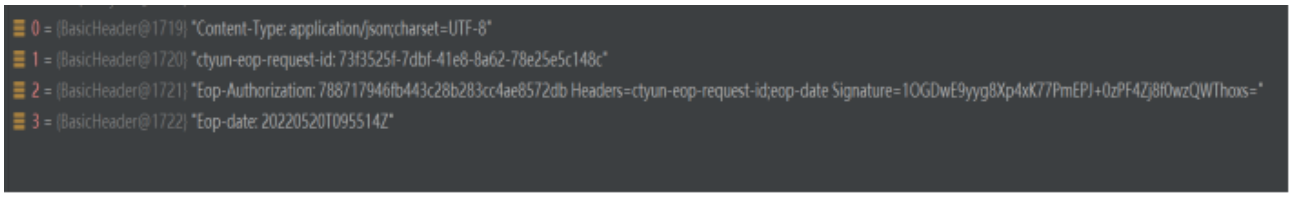
参数	是否必填
GET	请求服务器返回指定资源。
PUT	请求服务器更新指定资源。
POST	请求服务器新增资源或执行特殊操作。
DELETE	请求服务器删除指定资源，如删除对象等。
HEAD	请求服务器资源头部。

6.2.2.3. 请求消息头

调用接口时，您需要在请求头附加公共字段，主要用于签名鉴权。字段如下;

名称	描述	是否必填	示例
Content-Type	消息体的类型（格式）。推荐用户使用默认值application/json，有其他取值时会在具体接口中专门说明。	是	application/json
Eop-date	该字段的格式是“yyyymmddTHHMMSSZ”，言简意赅就是“年月日 T 时分秒 Z”。	是	20211221T163614Z
ctyun-eop-request-id	该字段是 uuid，32 位随机数。	是	33dfa732-b27b-464f-b15a-21ed6845afd5
Eop-Authorization	由天翼云官网 AccessKey 和 SecurityKey 经签名后生成，签名逻辑详见后续说明。	是	8fce08794e03478092be7f5ce31xxxxx Headers=ctyun-eop-request-id;eop-date Signature=NlMHOhk5bVfZ9MwDSSJydcZjjENmDtpNYigJGVb
regionId	资源池标识。	是	60a39fca876e11ea91cf0242ac110002

请求消息头示例



5.2.3. 签名鉴权

6.2.3.1. 信息获取

1. 天翼云注册账号。能力使用者于天翼云门户(https://www.ctyun.cn/)注册账号，并通过认证。

2. 获取 CTAPI 信息。能力使用者阅读和选择所需产品，了解产品业务场景及需要的 API 和需要使用的 endpoint。

3. 获取 AK/SK 信息。能力使用者登录天翼云门户，进入管理中心-->个人中心-->第三方账号管理，找到【用户 AccessKey】，获取 AK/SK，作为调用 CTAPI 的 key 使用。

6.2.3.2. 基本签名流程

ctyun-eop-ak/ctyun-eop-sk 基本签名流程如下：

1、构造待签字符串：使用规范请求和其他信息创建待签字符串；

2、计算密钥：使用 HEADER、ctyun-eop-sk、ctyun-eop-ak 来创建 Hmac 算法的密钥；

3、计算签名：使用第三步的密钥和待签字符串在通过 hmacsha256 来计算签名。

4、签名应用：将生成的签名信息作为请求消息头添加到 HTTP 请求中。

6.2.3.3. 构造待签字符串

构造待签字符串 sigture 流程如下：

sigture=需要进行签名的 Header 排序后的组合列表+ "\n" + 排序的 query + "\n" + toHex(sha256(原封的 body))

需要进行签名的 Header 排序后的组合列表（排序的 header）	将 ctyun-eop-request-id、eop-date 以“header_name:header_value”的形式、以“\n”作为每个 header 的结尾符、以英文字母表作为 header_name 的排序依据将它们拼接起来。 注意：EOP 强制要求 ctyun-eop-request-id、eop-date 必须进行签名。其他字段是否需要签名看自身需求。 例子(假设你需要将 ctyun-eop-request-id、eop-date、host 都要签名)： ctyun-eop-request-id:123456789\neop-date:20210531T100101Z\n
排序的 query	query 以&作为拼接，key 和值以=连接，排序规则使用 26 个英文字母的顺序来排序，Query 参数全部都需要进行签名。
toHex(sha256(原封的 body))	传进来的 body 参数进行 sha256 摘要，对摘要出来的结果转十六进制。

sigture 示例 1（假设 query 为空、需要进行签名的 Header 排序后的组合列表为“ctyun-eop-request-id:27cfe4dc-e640-45f6-92ca-492ca73e8680\neop-date:20220525T160752Z\n”、body 参数做 sha256 摘要后转十六进制为 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855）：

ctyun-eop-request-id:27cfe4dc-e640-45f6-92ca-492ca73e8680\neop-date:20220525T160752Z\n\n\ne3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

sigture 示例 2（假设 query 不为空、需要进行签名的 Header 排序后的组合列表为“ctyun-eop-request-id:27cfe4dc-e640-45f6-92ca-492ca73e8680\neop-date:20220525T160752Z\n”、body 参数做 sha256 摘要后转十六进制 e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855）：

ctyun-eop-request-id:27cfe4dc-e640-45f6-92ca-492ca73e8680\neop-date:20220525T160930Z\n\naa=1&bb=2\ne3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

6.2.3.4. 构造动态密钥

构造动态密钥 kdate 流程如下：

- 使用 eop-date 作为数据，sk 作为密钥，算出 ktime；
- 使用 ak 作为数据，ktime 作为密钥，算出 kAk；

- 使用 eop-date 的年月日值作为数据，kAk 作为密钥，算出 kdate。

eop-date	yyyymmddTHHMMSSZ（20211221T163614Z）（年月日 T 时分秒 Z）
Ktime	使用 eop-date 作为数据，sk 作为密钥，算出 ktime。 Ktime = hmacSha256(eop-date, sk)
kAk	使用 ak 作为数据，ktime 作为密钥，算出 kAk。 kAk = hmacsha256(ak, ktime)
kdate	使用 eop-date 的**年月日值**作为数据，kAk 作为密钥，算出 kdate。 kdate = hmacsha256(eop-date, kAk)

6.2.3.5. 计算签名

计算签名 Signature

使用 kdate 作为密钥、sigture 作为数据，将其得到的结果进行 base64 编码得出 Signature。

Signature	1. 将上述步骤的得出的待签名字符串 string_sigture、kdate，根据 hmacsha256(kdate, string_sigture)计算结果； 2. 再将结果进行 base64 编码得出 Signature。

6.2.3.6. 签名应用

将生成的签名信息作为请求消息头添加到 HTTP 请求中，在 http_client 的请求头增加 3 个字段，分别是 Eop-date、ctyun-eop-request-id、Eop-Authorization。

Eop-date：该字段的格式是“yyyymmddTHHMMSSZ”，言简意赅就是“年月日 T 时分秒 Z”，示例“eop-date:20211221T163614Z”。

ctyun-eop-request-id：该字段是 uuid，32 位随机数。

Eop-Authorization：ak Headers=xxx Signature==xxx。即将以上获取的参数拼接，并以空格隔开。

Headers	将需要进行签名的请求头字段以 “header_name”的形式、以“;”作为间隔符、以英文字母表作为 header_name 的排序依据将它们拼接起来。 例子（假设你需要将 ctyun-eop-request-id、eop-date 都要签名）： Headers=ctyun-eop-request-id;eop-date
Eop-Authorization	Eop-Authorization:ak Headers=xxx Signature=xxx。注意，ak、Headers、Signature 之间**以空格隔开**。 例如：Eop-Authorization:ak Headers=ctyun-eop-request-id;eop-date Signature=NlMHOhk5bVfZ9MwDSSJydcZjjENmDtpNYigJGVb

	注意：如果你需要进行签名的 Header 不止默认的 ctyun-eop-request-id 和 eop-date，那么你需要在 http_client 的请求头部中加上，并且 Eop-Authorization 中也需要增加

5.3. 密钥管理接口

5.3.1. 创建用户主密钥

URL

POST /v1/cmkManage/createKey

请求参数

参数	是否必填	参数位置	参数类型	说明
protectionLevel	是	body	String	加密等级： 0:硬件 1:软件
keySpec	是	body	String	密钥算法： 软件加密支持： 1. AES_256 2. RSA_2048 硬件加密支持： 1. AES_256 2. RSA_2048 3. Ctyun_SM2 4. Ctyun_SM4
regionId	是	body	String	资源池 id。
keyUsage	是	body	String	密钥用途： 1. Encrypt/Decrypt 。 2. Sign/Verify(对称密钥不支持该用途，如 AES_256,SM4 等)
enableAutomaticRotation	是	body	String	是否开启密钥轮转（非对称密钥不能进行密钥轮转）： true:开启 false:不开启
description	否	body	String	密钥描述，长度不能大于 8192。
rotationInterval	否	body	String	密钥轮转周期：时间范围为 7 天-730 天，只需要传数字+d 即可（例如 7d）。

参数	是否必填	参数位置	参数类型	说明
origin	是	body	String	密钥来源： 1. KMS 2. EXTERNAL（外部密钥）

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "automaticRotation": "1",
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "creationDate": "Fri Jul 30 14:46:11 CST 2021",
            "creator": "9fd671db0cc04c6d9f73e0169ba8d264",
            "description": "测试创建密钥",
            "keySpec": "AES_256",
            "keyState": "1",
            "keyUsage": "1",
            "lastRotationDate": "Fri Jul 30 14:46:11 CST 2021",
            "materialExpireTime": "null",
            "origin": "1",
            "protectionLevel": "1",
            "regionId": "100054c0416811e9a6690242ac110002"
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
cmkUuid	用户主密钥 uuid。
automaticRotation	是否开启自动密钥轮转，取值： 0：Enabled：自动轮转处于开启状态。 1：Disabled：自动轮转处于未开启状态。 2：Suspended：自动轮转被暂停执行 说明 仅适用于对称类型的 CMK，非对称类型的 CMK 不支持自动轮转。

参数	说明
creationDate	创建主密钥的日期和时间（UTC）
creator	主密钥创建者。
deleteDate	主密钥的预计删除时间。 说明 只有当 KeyState 值为 2:PendingDeletion 时，返回该值。
description	主密钥的描述。
keySpec	主密钥的类型。
keyState	密钥状态： 1：enabled（默认） 0：disabled 2：pendingdeletion 3：pendingimport(only EXTERNAL)
keyUsage	主密钥的用途。
lastRotationDate	上一次轮转時間。
materialExpireTime	密钥材料的过期时间（UTC）。当该值为空时，表示密钥材料不会过期。
nextRotationDate	下一次轮转的时间。 说明：只有当 AutomaticRotation 参数值为 Enabled 或 Suspended 时，返回该值。
origin	主密钥的密钥材料来源。
primaryKeyVersion	对称类型主密钥的当前主版本标志符。
protectionLevel	密钥的保护级别。
rotationInterval	密钥自动轮转的周期
regionId	资源池 id

5.3.2. 启用密钥

URL

POST https://api.ctyun.cn/apiproxy/v3/product/enableKey

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	用户主密钥 uuid，如果为空，则自动创建新的主密钥。

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.3.3. 禁用密钥

URL

POST https://api.ctyun.cn/apiproxy/v3/product/disableKey

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	用户主密钥 uuid，如果为空，则自动创建新的主密钥。

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.3.4. 计划删除

URL

POST https://api.ctyun.cn/apiproxy/v3/product/scheduleKeyDeletion

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	用户主密钥 uuid，如果为空，则自动创建新的主密钥。
pendingWindowInDays	是	body	String	密钥预删除周期。在这段时间内，您可以撤销删除处于待删除状态的密钥；预删除时间过后无法撤销删除。 取值范围：7~30。单位：天。

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.3.5. 取消计划删除

URL

POST https://api.ctyun.cn/apiproxy/v3/product/cancelKeyDeletion

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	用户主密钥 uuid，如果为空，则自动创建新的主密钥

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```


5.3.6. 更新密钥描述

URL

POST https://api.ctyun.cn/apiproxy/v3/product/updateKeyDescription

请求参数

参数	是否必填	参数位置	参数类型	说明
description	是	body	String	主密钥的描述性信息。通常用于描述主密钥的用途，例如主密钥保护的数据类型、可使用主密钥的应用等。
cmkUuid	是	body	String	密钥 ID。主密钥的全局唯一标识符。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```


5.3.7. 查看密钥详情

URL

GET https://api.ctyun.cn/apiproxy/v3/product/describeKey

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	header	String	主密钥（CMK）的全局唯一标识符。该参数也可以被指定为 CMK 绑定的别名。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "automaticRotation": 1,
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "creationDate": "2021-07-30 14:46:11",
            "creator": "9fd671db0cc04c6d9f73e0169ba8d264",
            "description": "测试 kms 密钥",
            "isDefault": 0,
            "keySpec": "AES_256",
            "keyUsage": 1,
            "lastRotationDate": "2021-07-30 14:46:11",
            "origin": 1,
            "primaryKeyVersion": "73670b28-4eea-4260-b497-ae0334cc0c85",
            "protectionLevel": 1,
            "status": 1
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
creationDate	创建 CMK 的日期和时间（UTC）。
description	CMK 的描述。
cmkUuid	CMK 的全局唯一标识符。
keyState	CMK 的状态主密钥状态： 1：enabled（默认） 0：disabled

参数	说明
	2：pendingdeletion 3：pendingimport(only EXTERNAL)
keyUsage	CMK 的用途。
deleteDate	CMK 的预计删除时间，详情请参见 ScheduleKeyDeletion。只有当 keyState 值为 PendingDeletion 时，返回该值。
creator	CMK 创建者。
origin	CMK 的密钥材料来源。密钥来源： 1:KMS 0:EXTERNAL
materialExpireTime	密钥材料的过期时间（UTC）。当该值为空时，表示密钥材料不会过期。
protectionLevel	密钥的保护级别。保护级别： 1:software 0:HSM
primaryKeyVersion	对称类型 CMK 的当前的主版本标志符。主版本是对称类型 CMK 的活跃加密密钥，KMS 使用主版本处理加密请求。不适用于非对称类型的 CMK。
lastRotationDate	最近一次轮转的时间（UTC）。如果是新创建密钥，则为初始密钥版本生成时间。
automaticRotation	是否开启自动密钥轮转，取值如下： 0-Enabled：自动轮转处于开启状态。 1-Disabled：自动轮转处于未开启状态。 2-Suspended：自动轮转被暂停执行，详情请参见自动轮转密钥。 仅适用于对称类型的 CMK。非对称类型的 CMK 不支持自动轮转。
rotationInterval	密钥自动轮转的周期（秒数）。格式为整数值后加上字符 s。例如，7 天的轮转周期为 604800s。只有当 Automatic_rotation 参数值为 Enabled 或 Suspended 时，返回该值。
nextRotationDate	下一次轮转的时间。只有当 Automatic_rotation 参数值为 Enabled 或 Suspended 时，返回该值。
keySpec	CMK 的类型。

5.3.8. 查询密钥列表

URL

GET https://api.ctyun.cn/apiproxy/v3/product/listAliasKeys

请求参数

参数	是否必填	参数位置	参数类型	说明
pageNumber	否	header	String	当前页数。取值范围：从0开始的整数。默认值：0。
pageSize	否	header	String	每页返回值的个数。取值范围：1~100。默认值：10。
keyUsage	否	header	String	密钥用途： 1：Encrypt/Decrypt 2：Sign/Verify(对称密钥不支持该用途，如AES_256,SM4等)
protectionLevel	否	header	String	密钥的保护级别。保护级别： 1:software 0:HSM
keySpec	否	header	String	密钥算法。 软件加密支持： 1. AES_256 2. RSA_2048 硬件加密支持： 1. AES_256 2. RSA_2048 3. Ctyun_SM2 4. Ctyun_SM4
status	否	header	String	CMK的状态。主密钥状态： 1：enabled（默认） 0：disabled 2：pendingdeletion 3：pendingimport(only EXTERNAL)
regionId	否	header	String	资源池id。

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
```

```
        "result": {
            "content": [
                {
                    "automaticRotation": 1,
                    "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
                    "createdTime": "2021-07-30 14:46:11",
                    "description": "测试 kms 密钥",
                    "isDefault": 0,
                    "keySpec": "AES_256",
                    "keyUsage": 1,
                    "lastRotationDate": "2021-07-30 14:46:11",
                    "origin": 1,
                    "protectionLevel": 1,
                    "regionId": "100054c0416811e9a6690242ac110002",
                    "status": 1,
                    "updatedTime": "2021-07-30 15:18:55",
                    "userId": "9fd671db0cc04c6d9f73e0169ba8d264"
                }
            ],
            "pageNumber": 1,
            "pageSize": 10,
            "totalCount": 1
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
number	当前页数。
size	每页返回值的个数。
totalElements	主密钥的总数。
content	主密钥版本列表。

5.4. 别名管理接口

5.4.1. 密钥别名创建

URL

POST https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/createAlias

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	CMK 的全局唯一标识符。
aliasName	是	body	String	要操作的别名。

请求示例

```
{
"cmkUuid": "b5161ed1-138e-403f-b943-818cae409ded",
"aliasName": "alias/uuid_1"
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.4.2. 密钥别名更新

URL

POST https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/updateAlias

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	CMK 的全局唯一标识符。
aliasName	是	body	String	要操作的别名。

请求示例

```
{
"cmkUuid": "b5161ed1-138e-403f-b943-818cae409ded",
"aliasName": "alias/uuid_1"
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.4.3. 密钥别名删除

URL

POST https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/deleteAlias

请求参数

参数	是否必填	参数位置	参数类型	说明
userId	是	body	String	用户 ID。
regionId	是	body	String	区域 ID。
aliasName	是	body	String	密钥别名。

请求示例

```
{
  "userId": "9fd671db0cc04c6d9f73e0169ba8d264",
  "regionId": "d8bbd132b53a11e9b0e40242ac110002",
  "aliasName": "alias/uuid_3"
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.4.4. 列出所有别名

URL

GET https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/listAlias

请求参数

参数	是否必填	参数位置	参数类型	说明
userId	是	heard	String	用户 id。
regionId	是	heard	String	资源池 id。
pageNumber	是	heard	String	当前页数。
pageSize	是	heard	String	每页返回的结果个数。

请求示例

```
{
    "userId": "5a20ab8588904a25bf35587f82aaae52",
    "regionId": "100054c0416811e9a6690242ac110002",
    "pageNumber": 1,
    "pageSize": 10
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "aliases": {
                "alias": []
            },
            "pageNumber": "1",
            "pageSize": "10",
            "totalCount": 0
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
aliases	用户别名信息。
pageNumber	当前页数。
pageSize	每页的返回结果个数。
totalCount	返回的别名总数。

5.4.5. 列出与指定密钥绑定的别名

URL

GET https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/listAliasByUuid

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	heard	String	CMK 的全局唯一标识符。
pageNumber	是	heard	String	当前页数。
pageSize	是	heard	String	每页返回的结果个数。

参数	是否必填	参数位置	参数类型	说明
pageSize	是	heard	String	每页返回的结果个数。

请求示例

```
{
"cmkUuid": "b5161ed1-138e-403f-b943-818cae409ded",
 "pageNumber": 1,
    "pageSize": 10
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "aliases": {
                "alias": [
                    {
                        "aliasName": "alias/uuid_1",
                        "cmkUuid": "b5161ed1-138e-403f-b943-818cae409ded"
                    },
                    {
                        "aliasName": "alias/uuid_2",
                        "cmkUuid": "b5161ed1-138e-403f-b943-818cae409ded"
                    }
                ]
            },
            "pageNumber": "1",
            "pageSize": "10",
            "totalCount": 2
        },
        "statusCode": 200,
        "success": 1
    }
}
```

返回参数说明

参数	说明
aliases	用户别名信息。

参数	说明
pageNumber	当前页数。
pageSize	每页的返回结果个数。
totalCount	返回的别名总数。

5.5. 外部密钥导入接口

5.5.1. 获取导入主密钥材料的参数

URL

GET https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/getParametersForImport

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	heard	String	主密钥（CMK）的全局唯一标识符，待导入类型。
wrappingAlgorithm	是	heard	String	用于加密密钥材料的算法。
wrappingKeySpec	是	heard	String	用于加密密钥材料的公钥类型。

请求示例

```
{
  "cmkUuid": "d72921c4-27ed-4252-9961-389aa1c59bb0",
  "wrappingAlgorithm": "RSAES_OAEP_SHA_1",
  "wrappingKeySpec": "RSA_2048"
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
```

```
        "code": 200,
        "result": {
            "cmkUuid": "d72921c4-27ed-4252-9961-389aa1c59bb0",
            "importToken":
"MmNkZDk4YjEtNjBhMi00MWFhLWI3ZDYtMmM4MGQyOWNkNDYyNzRmOTc2ZGEtYTJiMS000TEyLThlOTctMDc0NW
RlN2Y5MjZlJlJTQUVTX09BRVBfU0hBXzEmYTlHcmkxLzlPVkxZUWQxQWZobCsxZVBZdEtORm5MSUhHS0ZscmVuQ
nQwbDBsSHlsb0phcFJMbkxmWlVKbVZhc1BRbHc3dDZDTVJJWDlrSnowQXAwOTU5Q2kyalhnUDlrdGRsSnNVampT
bjZRVkNnVlpNVE1mSktXUm1ZL0xkNTEwSmRWQXJ5U2lLTVhLenZoanYxUUx2L2FlbUw0NE9wRmMwb0svS1JCS3h
DVzF1YmZSZnYyaEtDTmYyRlZFdGgyVy90VHh0enJkM2c1MlhZOVNyLzV6SXppNXVxUHZ3VWFpREozRGQwS25XOW
hhTnlhYll4MHBhZkFXdkNLL1pJODRNNm5ScDlKSkd4MG5BNU9iaFU5Nm1WUHJFb0tYdlptZy9xd3BDNW94QnhOO
DJ3eGNtSUdaVVFnTnU3UTJvdkswbGxvNVRkbDEwY3BqSUFiSlozSjV3PT0=",
            "publicKey":
"MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAkPIGFmMKBIWrkWDZV50L7Dd71+/ebh7QLIkOpCjGJQ
NdPe9QD+Df00HwhecSbGTKMiHfjhEtmXW1SDqnrz90SmEpEAfXdbqWPX68yhex9JIidmZP7iS8P1zwU9yW+cALV
ADbndDOhdLaJHy1zidpiBHh2u3GF1KDFSCPNEgn+TjCgI1JSFYbKIzeXSemC0cny2CG2LnuUcI54001F5hvHR1P
1tyrNkV006x0iyUbrB7FC79yrnhbvGY2QdwWGAZf1i2X76kxbJ2IZb+VBCIKK0KT6eiCVkyrDP2u6no0blS1sWL
qNB0IlGe1e8z/kbC+A2I3xB8Va2/0yH0k8t52xQIDAQAB",
            "tokenExpireTime": "2021-08-13 15:44:06"
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
importToken	导入令牌。
cmkUuid	主密钥全局唯一标识符。
publicKey	用于加密密钥材料的公钥。
tokenExpireTime	导入令牌的过期时间。

5.5.2. 导入主密钥材料

URL

POST https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/importKeyMaterial

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符，待导入类型。
keyMaterialExpireUnix	是	body	String	用于加密密钥材料的算法。
importToken	是	body	String	用于加密密钥材料的公钥类型。
encryptedKeyMaterial	是	body	String	用于加密密钥材料的公钥类型。

请求示例


```
{
  "cmkUuid": "d72921c4-27ed-4252-9961-389aa1c59bb0",
  "keyMaterialExpireUnix": "0",
  "importToken":
"MmNkZDk4YjEtNjBhMi00MWFhLWI3ZDYtMmM4MGQyOWNkNDYyNzRmOTc2ZGEtYTJiMS00OTEyLThlOTctMDc0NWRlN2Y5MjZlIJlJTQUVTX09BRVBfU0hBXzEmYTlHcmkxLzlPVkxZUWQxQWZobCsxZVBZdEtORm5MSUhHS0ZscmVuQnQwbDBsSHlsb0phcFJMbkxmWlVKbVZhc1BRbHc3dDZDTVJJWDlrSnowQXAwOTU5Q2kyalhnUDlrdGRsSnNVampTbjZRVkNnVlpNVE1mSktXUm1ZL0xkNTEwSmRWQXJ5U2lLTVhLenZoanYxUUx2L2FlbUw0NE9wRmMwb0svS1JCS3hDVzF1YmZSZnYyaEtDTmYyRlZFdGgyVy9OVHh0enJkM2c1MlhZOVNyLzV6SXppNXVxUHZ3VWFpREozRGQwS25XOWhhTnlhYlI4MHBhZkFXdkNLL1pJODRNNm5ScDlKSkd4MG5BNU9iaFU5Nm1WUHJFb0tYdlptZy9xd3BDNW94QnhOODJ3eGNtSUdaVVFnTnU3UTJvdkswbGxvNVRkbDEwY3BqSUFiSlozSjV3PT0=",
  "encryptedKeyMaterial":
"i1XWzs02bWYyImctYBEidIyTmvw8796S8p6iUwKGfnpyx8cJikBT1jE9Cd6ptwqKA3Y657NEilmKlRpSBPlYkWq9NkegBQ8kJGqfN1vcpEA6dE8oDxb3TAdCfX9n/InuTU217wQJC7wuoQaizMcEzsLaE/aYA8sRcGQPLDQJ0uXoJkrKKAiPLfutU+R3oL5ncyZWHz/OFmp8n5nGcbF/txxAUnOF3Dm3hHt/PMqcY81el/3F+R/DXuArXfx+efT1x2U3GViMOtkpWFQLlHbfECLrSfwXZ7KlpPgPbkcWmxa7gtjWmhmfoGthI3ycNLI+TzELeStm4K/0jOw5v8+mqg=="
}
```


成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.5.3. 删除主密钥材料

URL

POST https://api.ctyun.cn/apiproxy/v3/account/queryTrialBalance/deleteKeyMaterial

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符，待导入类型。

请求示例

```
{
  "cmkUuid": "d72921c4-27ed-4252-9961-389aa1c59bb0"
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.6. 密钥版本管理接口

5.6.1. 查看密钥版本

URL

GET https://api.ctyun.cn/apiproxy/v3/account/versionControl/describeKeyVersion

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	header	String	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
keyVersionId	是	header	String	用于加密明文的密钥版本标志符。是指定 CMK 的主版本。

请求示例

```
{
    "keyVersionId": "a7d75c81-cc4e-462b-9d5d-53d8becb9b06",
    "cmkUuid": "70b4d747-abc2-471b-abea-622daf112d41",
    "customInfo":
{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","userId":"5a20ab85
88904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "cmkUuid": "70b4d747-abc2-471b-abea-622daf112d41",
            "creationDate": "2021-08-12 08:48:28",
            "keyVersionId": "a7d75c81-cc4e-462b-9d5d-53d8becb9b06"
        },
        "statusCode": 200,
        "success": 1
    }
}
```

返回参数说明

参数	说明
creationDate	创建密钥版本时的日期和时间（UTC 时间）。
cmkUuid	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
keyVersionId	用于加密明文的密钥版本标志符。是指定 CMK 的主版本。

5.6.2. 列出主密钥的所有密钥版本

URL

GET https://api.ctyun.cn/apiproxy/v3/account/versionControl/listKeyVersions

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	header	String	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
pageNumber	否	header	String	当前页数。取值范围：从 0 开始的整数。默认值：0。
pageSize	否	header	String	每页返回值的个数。取值范围：1~100。默认值：10。

请求示例

```
{
    "cmkUuid": "70b4d747-abc2-471b-abea-622daf112d41",
    "pageNumber": "1",
    "pageSize": "10",
    "customInfo":
{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","userId":"5a20a
b8588904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "content": [
                {
                    "cmkUuid": "70b4d747-abc2-471b-abea-622daf112d41",
                    "primaryKeyVersion": "a7d75c81-cc4e-462b-9d5d-53d8becb9b06",
                    "updatedAt": "2021-08-12 08:48:28"
                }
            ],
            "pageNumber": 1,
```

```
            "pageSize": 10,
            "totalCount": 1
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
number	当前页数。
size	每页返回值的个数。
totalElements	主密钥的总数。
content	主密钥版本列表。

5.6.3. 更新轮转策略

URL

POST https://api.ctyun.cn/apiproxy/v3/account/versionControl/updateRotationPolicy

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	是否开启自动密钥轮转。取值：true | false，true 时 rotationInterval 为必填。
enableAutomaticRotation	是	body	String	待加密明文（必须经过 Base64 编码）。
rotationInterval	是	body	String	自动轮转的时间周期。格式为 integer[unit]，其中 integer 表示时间长度，unit 表示时间单位。合法的 unit 单位为：d（天）。取值：7~730 天。当 enableAutomaticRotation 为 true 时必传。

请求示例

```
{
```

```
    "cmkUuid": "70b4d747-abc2-471b-abea-622daf112d41",
    "rotationInterval": "7d",
    "enableAutomaticRotation": true,

"customInfo":{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","user
Id":"5a20ab8588904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "statusCode": 200,
        "success": 1
    }
}
```

5.6.4. 创建新密钥版本

URL

POST https://api.ctyun.cn/apiproxy/v3/account/versionControl/createKeyVersion

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	CMK 的全局唯一标识符。如果请求中的 cmkUuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符

请求示例

```
{
    "cmkUuid": "ca66428c-ff67-4b4d-941e-6382bb99b4ab",
    "customInfo":
{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","userId":"5a20ab85
88904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
```

```
        "result": {
            "cmkUuid": "625945e0-6f5e-42eb-bfda-a27e2d38c89a",
            "creationDate": "2021-08-09 18:12:38",
            "keyVersionId": "542c5f9b-59a2-446f-8b1c-f0d0d3f2b891"
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
creationDate	创建密钥版本时的日期和时间（UTC 时间）
cmkUuid	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符
keyVersionId	用于加密明文的密钥版本标志符，是指定 CMK 的主版本。

5.7. 密码运算接口

5.7.1. 在线加密

URL

POST https://api.ctyun.cn/apiproxy/v3/account/keyCompute/encrypt

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符。
plaintext	是	body	String	待加密明文（必须经过 Base64 编码）。

请求示例


```
{
    "plaintext": "SGVsbG8gd29ybGQ=",
    "cmkUuid": "241ede22-6261-4617-9caf-10d89990516c"
}
```

成功返回


```
{
    "code": 200,
    "result": {
        "ciphertextBlob":
"MDA2NE1qUXhaV1JsTWpJdE5qSTJNUzAwTmpFM0xUbGpZV1l0TVRCa09EazVPVEExTVRaakpqUTVaV00zTm1RM0
xXTmpOR010TkRBd1pTMDVaakU1TFdZNU1EQXhOVGczWVdVd1pnPT3oCYiGAy7mNTLitlIJaQ92",
        "cmkUuid": "241ede22-6261-4617-9caf-10d89990516c",
        "keyVersionId": "49ec76d7-cc4c-400e-9f19-f9001587ae0f"
    },
    "statusCode": 200,
    "success": 1
}
```


返回参数说明

参数	说明
ciphertextBlob	数据被指定 CMK 的主版本加密后的密文。
cmkUuid	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
keyVersionId	用于加密明文的密钥版本标志符，是指定 CMK 的主版本。

5.7.2. 产生数据密钥（信封加密）

URL

POST https://api.ctyun.cn/apiproxy/v3/product/generateDataKey

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	密钥 ID。主密钥的全局唯一标识符。
keySpec	否	body	String	指定生成的数据密钥的长度，取值： 1：AES_256：256 比特的对称密钥。 2：AES_128：128 比特的对称密钥。（若均为空，默认 AES_256）

参数	是否必填	参数位置	参数类型	说明
numberOfBytes	否	body	String	指定生成的数据密钥的长度。取值：1~1024。单位：字节。
regionId	否	body	String	资源池 id。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "ciphertextBlob":
"MDA2NE9HSmpZVGhtTXpNdFpEUXlZUzAwTkRoaExUZzJObUl0WVRBMk5HWTBOR0l5T1dJM0pqY3pOamN3WWpJNE
xUUmxaV0V0TkRJMk1DMWI0RGszTFdGbE1ETXpOR05qTUdNNE5RPT2PMfsXbpQgAn22PS9mrEh8yB/Ctjh1ILEyV
710Qk1wZFQnS0tfe2qJe9ZnsAYTQL8=",
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "keyVersionId": "73670b28-4eea-4260-b497-ae0334cc0c85",
            "plaintext": "sc7280+klUSIn3Y9FHdfKGUT+6kPrcIMW41uZQeXxGU="
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
ciphertextBlob	数据密钥被指定 CMK 的主版本加密后的密文。
keyVersionId	密钥版本 ID。主密钥版本的全局唯一标识符。
plaintext	数据密钥的明文经过 Base64 编码的后的值。
cmkUuid	密钥 ID。主密钥的全局唯一标识符。

5.7.3. 产生无明文返回值的数据密钥（信封加密）

URL

POST https://api.ctyun.cn/apiproxy/v3/product/generateDataKeyWithoutPlaintext

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	必填	body	String	密钥 ID。主密钥的全局唯一标识符。
keySpec	非必填	body	String	指定生成的数据密钥的长度，取值： 1：AES_256：256 比特的对称密钥。 2：AES_128：128 比特的对称密钥。 （若均为空，默认 AES_256）
numberOfBytes	非必填	body	String	指定生成的数据密钥的长度。取值：1~1024。 单位：字节。
regionId	非必填	body	String	资源池 id。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "ciphertextBlob":
"MDA2NE9HSmpZVGhtTXpNdFpEUXlZUzAwTkRoaExUZzJObUlOWVRBMk5HWTBOR0l5T1dJM0pqY3pOamN3WWpJNE
xUUmxaV0V0TkRJMk1DMWlORGszTFdGbE1ETXpOR05qTUdNNE5RPT0DwDXQXWR1LLH6Xzc9v+dwEZAT7IwZIlywI
nijifbJO1QnS0tfe2qJe9ZnsAYTQL8=",
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "keyVersionId": "73670b28-4eea-4260-b497-ae0334cc0c85"
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
ciphertextBlob	数据密钥被指定 CMK 的主版本加密后的密文。
keyVersionId	密钥版本 ID。主密钥版本的全局唯一标识符。

参数	说明
cmkUuid	密钥 ID。主密钥的全局唯一标识符。

5.7.4. 导出数据密钥

URL

POST https://api.ctyun.cn/apiproxy/v3/product/exportDataKey

请求参数

参数	是否必填	参数位置	参数类型	说明
ciphertextBlob	是	body	String	主密钥（CMK）加密的数据密钥的密文。
publicKeyBlob	是	body	String	Base64 格式的公钥。
wrappingAlgorithm	是	body	String	使用 publicKeyBlob 所指定的公钥，加密（Wrap）数据密钥时的加密算法取值：RSAES_OAEP_SHA_256 ，RSAES_OAEP_SHA_1
wrappingKeySpec	是	body	String	Public_key_blob 的密钥。类型取值：RSA_2048。

成功返回


```
{
  "statusCode": 800,
  "returnObj": {
    "code": 200,
    "result": {
      "keyVersionId": "c4e1beaf-b4e5-4bbd-b49f-252a2b2c1e86",
      "cmkUuid": "86a4c333-5c70-4b1c-81ba-b14242a34ebb",
      "exportedDataKey":
"MH5i0kasurouMUmlRBjwMZsSGjGV1bfoxJzDGkzUF/2Hi9i1RyRe2WhZwNdIdLxhAvzA5YX6jJTXPG5NQY/Z/I
f9zA4EJ/mjHBJizJ22mHpCD7iV8sfos0CUSxFQWRX9aNrScvDGRKg/Y3EM8AcP/V200o4hXDKycvK7L9KMCg5HI
XwHs5modftBlPg3hz7J0yp2qpkxc5ZcEd9kFncYTZvqV1phNg9ukkiqzfhHoTw+NKeDD29nh1lnPWuvQ2YkbJ8u
4xB0AQ785y1TT0Z3TS5uNEo5lebnHBfy2Q5PQsDzb3/STcPRLPNnE/B0Vkk0JU20X27DpVS6Y8Le+6dTMQ=="
    },
    "success": 1,
    "statusCode": 200
  }
```

}

返回参数说明

参数	说明
exportedDataKey	公钥加密保护导出的数据密钥。
cmkUuid	解密传入的数据密钥密文使用的主密钥 ID。主密钥的全局唯一标识符。
keyVersionId	用于解密传入的数据密钥密文的密钥版本标识符。

5.7.5. 产生并导出数据密钥

URL

POST https://api.ctyun.cn/apiproxy/v3/product/generateAndExportDataKey

请求参数

参数	是否必填	参数位置	参数类型	说明
publicKeyBlob	是	body	String	Base64 格式的公钥。
wrappingAlgorithm	是	body	String	使用 Public_key_blob 所指定的公钥，加密（Wrap）数据密钥时的加密算法。取值：RSAES_OAEP_SHA_256 ， RSAES_OAEP_SHA_1
wrappingKeySpec	是	body	String	Public_key_blob 的密钥。类型取值：RSA_2048。
cmkUuid	是	body	String	密钥 ID。主密钥（CMK）的全局唯一标识符。该参数也可以被指定为主密钥绑定的别名。
keySpec	否	body	String	指定生成的数据密钥的长度，取值： 1：AES_256：256 比特的对称密钥。 2：AES_128：128 比特的对称密钥。（若均为空，默认 AES_256）
numberOfBytes	否	body	String	指定生成的数据密钥的长度。取值：1~1024。单位：字节。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "ciphertextBlob":
"MDA2NE9HSmpZVGhtTXpNdFpEUXlZUzAwTkRoaExUZzJ0bUl0WVRBMk5HWTBOR0l5T1dJM0pqY3p0amN3WWpJNE
xUUmxaV0V0TkRJMk1DMWI0RGszTFdGbE1ETXpOR05qTUdNNE5RPT11qhnyhbYRDHQXUG9xD8VSV3bQ8KTCjlu0W
UX8hRlx5FQnS0tfe2qJe9ZnsAYTQL8=",
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "exportedDataKey":
"vOwoK9PrH6m24wmHyYnQRxzIseaUGbIec4tvTeYHHJoFIDr5+KVZeAgdOehwjcsbBIyxbDGVOnbiXwpU/OK/Fw
oeRtzCeOPHvZp/99eOhrhWtiX+q2jgZdmSBwzxH9+1ZrIqy5W/Il8ybwy/zOmoYY1QWWrthmjAuUIJUGHYVs05r
90eewfBnpIWb42IE9doIBugNKx5zMJdEHIm3JdHJXAiubRC13UFZVoLCV3vCMPMgN6A9fOcj9RMPiF/gbrKqeei
IRpP282z/wc5uXdPGP/BrfpY1nyoKShj30IL4pe1pnHAaW03WJBRZOxtkDbzmvWhdIXIY9v2IKnGaMVCWw==",
            "keyVersionId": "73670b28-4eea-4260-b497-ae0334cc0c85"
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
exportedDataKey	公钥加密保护导出的数据密钥。
cmkUuid	解密传入的数据密钥密文使用的主密钥 ID。主密钥的全局唯一标识符。
keyVersionId	用于解密传入的数据密钥密文的密钥版本标识符。
ciphertextBlob	数据密钥被指定 CMK 的主版本加密后的密文。

5.7.6. 解密

URL

POST https://api.ctyun.cn/apiproxy/v3/product/decrypt

请求参数

参数	是否必填	参数位置	参数类型	说明
ciphertextBlob	是	body	String	主密钥（CMK）加密的数据密钥的密文。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "keyVersionId": "73670b28-4eea-4260-b497-ae0334cc0c85",
            "plaintext": "sc7280+klUSln3Y9FHdfKGUT+6kPrcIMW41uZQeXxGU="
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
cmkUuid	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
keyVersionId	密钥版本 ID。主密钥版本的全局唯一标识符。
plaintext	解密后的明文经过 Base64 编码的后的值。

5.7.7. 转加密

URL

POST https://api.ctyun.cn/apiproxy/v3/product/reEncrypt

请求参数

参数	是否必填	参数位置	参数类型	说明
ciphertextBlob	是	body	String	待转加密的密文。该参数可以为对称加密或非对称加密返回的密文数据。

参数	是否必填	参数位置	参数类型	说明
				• 对称加密：调用 Encrypt、GenerateDataKey、GenerateDataKeyWithoutPlaintext 或 GenerateAndExportDataKey 接口返回的密文数据。 • 非对称加密：可以是调用 GenerateAndExportDataKey 接口返回的公钥加密数据，也可以是外部系统使用非对称公钥加密的数据。
destinationKeyId	是	body	String	对密文解密后再次加密时使用的对称主密钥 ID。
sourceKeyId	否	body	String	解密密文时使用的主密钥 ID。主密钥的全局唯一标识符。 说明：当 CiphertextBlob 是非对称加密返回的公钥加密数据时需要指定该参数。
sourceKeyVersionId	否	body	String	用于解密密文的密钥版本标识符。 说明：当 CiphertextBlob 是非对称加密返回的公钥加密数据时需要指定该参数。。
sourceEncryptionAlgorithm	否	body	String	CiphertextBlob 是公钥加密结果时，指定公钥加密的算法。算法详情，请参见 AsymmetricDecrypt。 取值： RSAES_OAEP_SHA_256，RSAES_OAEP_SHA_1。

成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "ciphertextBlob":
"MDA2NFIXWTVPREptWmpVdE5tRTFZaTAwT0RJekxXRTRNbU10TmpVNU9UUTNNVE5qTnprNEpqbGxaVEUxTldGaU
xXUmtaREV0TkRBMU15MDVPREF4TFdJMVIXSXpaamcyWkRGaFlnPT18s80dqMRNWjiTM9Aa7j8p",
            "cmkUuid": "af982ff5-6a5b-4823-a82c-65994713c798",
            "keyVersionId": "9ee155ab-ddd1-4053-9801-b5ab3f86d1ab"
        },
        "statusCode": 200,
        "success": 1
    }
```

```
}
```

返回参数说明

参数	说明
ciphertextBlob	使用指定的主密钥进行再次加密得到的密文。
cmkUuid	解密密文使用的主密钥 ID。主密钥的全局唯一标识符。
keyVersionId	主密钥下用于解密密文的密钥版本标识符。

5.7.8. 产生数字签名

URL

POST https://api.ctyun.cn/apiproxy/v3/account/asymmetric/asymmetricSign

请求参数

参数	是否必填	参数位置	参数类型	说明
algorithm	是	body	String	签名算法： RSA_PSS_SHA_256 RSA_PKCS1_SHA_256
digest	是	body	String	使用 Algorithm 中对应的哈希算法，对原始消息生成的摘要。使用 Base64 编码。
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符，该参数也可以被指定为主密钥绑定的别名。
keyVersionId	是	body	String	密钥版本的全局唯一标识符。
sourceEncryptionAlgorithm	否	body	String	CiphertextBlob 是公钥加密结果时，指定公钥加密的算法。算法详情，请参见 AsymmetricDecrypt。 取值： RSAES_OAEP_SHA_256，RSAES_OAEP_SHA_1。

请求示例

```
{
    "keyVersionId": "d818bece-abf2-4b29-b36c-93aae6995c2f",
    "cmkUuid": "798f17ab-f712-4029-900b-ace82dd1928f",
```

```
    "digest": "ZOylygCyaOW6GjVnihtTFtIS9PNmskdyMlNKiuyjfzw=",
    "algorithm": "RSA_PSS_SHA_256",

"customInfo":{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","user
Id":"5a20ab8588904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "cmkUuid": "798f17ab-f712-4029-900b-ace82dd1928f",
            "keyVersionId": "d818bece-abf2-4b29-b36c-93aae6995c2f",
            "value":
"BAz6BfLrO5VZkQyxsEcZs7kJRNIBvH7KbBwsYb1QacHNxaDS1tC5g4vlcfCysjQhbPbvn2qby7TtgPlpfjQ7FM
DDBmFaf/i1vcBv037aUoH9ngACV8VkueUEVrP+KwY3T7ZQtaLBpkpV+Rr1ufCq6eKG2z28mlPH3Xn9w1rlXCYxt
7rwlmuC5AsieYwYJF4SGocFKmeA+eu7ussn+ErSahQYq7rwG6L2kdzx8SQOfekMc3c3MNiXM99hTl/yD+Rp14za
mb6PRsnEz8WqDcByR937K3i8TR6w7DFHRzh7Zxt3CnQZhuiE/L4Q3NH1QZGWwE9PavTdwoSPhiEoVCyJ7A=="
        },
        "statusCode": 200,
        "success": 1
    }
}
```

返回参数说明

参数	说明
cmkUuid	主密钥的全局唯一标识符。
keyVersionId	密钥版本的全局唯一标识符。
value	计算出来的签名。

5.7.9. 数字签名验签

URL

POST https://api.ctyun.cn/apiproxy/v3/account/asymmetric/asymmetricVerify

请求参数

参数	是否必填	参数位置	参数类型	说明
algorithm	是	body	String	签名算法。
digest	是	body	String	使用 Algorithm 中对应的哈希算法，对原始消息生成的摘要。使用 Base64 编码。
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符，该参数也可以被指定为主密钥绑定的别名。
keyVersionId	是	body	String	密钥版本的全局唯一标识符。
value	是	body	String	待验证的签名值。

请求示例


```
{
    "keyVersionId": "d818bece-abf2-4b29-b36c-93aae6995c2f",
    "cmkUuid": "798f17ab-f712-4029-900b-ace82dd1928f",
    "value":
"BAz6BfLrO5VZkQyxsEcZs7kJRNIBvH7KbBwsYb1QacHNxaDS1tC5g4vlcfCysjQhbPbvn2qby7TtgPlpfjQ7FM
DDBmFaf/i1vcBv037aUoH9ngACV8VkueUEVrP+KwY3T7ZQtaLBpkpV+Rr1ufCq6eKG2z28mIPH3Xn9w1rIXCYxt
7rwImuC5AsieYwYJF4SGocFKmeA+eu7ussn+ErSahQYq7rwG6L2kdzx8SQOfekMc3c3MNiXM99hTl/yD+Rp14za
mb6PRsnEz8WqDcByR937K3i8TR6w7DFHRzh7Zxt3CnQZhuiE/L4Q3NH1QZGWwE9PavTdwoSPhiEoVCyJ7A==",
    "digest": "ZOylygCyaOW6GjVnihtTFtIS9PNmskdyMlNKiuyjfzw=",
    "algorithm": "RSA_PSS_SHA_256",

"customInfo":{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","user
Id":"5a20ab8588904a25bf35587f82aaae52"}}
}
```


成功返回


```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "cmkUuid": "798f17ab-f712-4029-900b-ace82dd1928f",
            "keyVersionId": "d818bece-abf2-4b29-b36c-93aae6995c2f",
            "value": "false"
        },
        "statusCode": 200,
        "success": 1
```

```
    }
}
```

返回参数说明

参数	说明
cmkUuid	主密钥的全局唯一标识符。
keyVersionId	密钥版本的全局唯一标识符。
value	签名验证是否通过。

5.7.10. 非对称密钥加密

URL

POST https://api.ctyun.cn/apiproxy/v3/account/asymmetric/asymmetricEncrypt

请求参数

参数	是否必填	参数位置	参数类型	说明
algorithm	是	body	String	加密算法： RSAES_OAEP_SHA_256 RSAES_OAEP_SHA_1 RSAES_PKCS1_V1_5
plaintext	是	body	String	要加密的明文，使用 Base64 编码。
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符。
keyVersionId	是	body	String	密钥版本的全局唯一标识符。

请求示例

```
{
    "plaintext": "SGVsbG8gd29ybGQ=",
    "cmkUuid": "b82cab6d-f83b-48c5-bcd2-9fdfcc556b40",
    "keyVersionId": "f94237b9-d8d7-4e6f-a3b4-4fe6ac27e4a5",
    "algorithm": "RSAES_OAEP_SHA_256",
```

```
"customInfo":{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","user
Id":"5a20ab8588904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "ciphertextBlob":
"QQA19Zn95F91AAv3nqwmY6KejuvFseSJyuNTu9UzyEH2MyeFPbD5FzergjD2aTSaopSIcnmHhwc8szkrr9KT81
2Q7TAX2GmQIuP2yx6J/t8Yg1p6uxeqBI/X3X11TCZUMVvZQsn9K0GcgCbpD0YTaSgeijONYnI5Fy23Eb9pSw4NC
WUDueqgtMm9nydYCT/mnEzZD53wUuqTd61Ugn1bu9h+/dxJsFztwrxQf4tRptGHY77snr/jDGwqb0JZwSuodrqJ
ZKthuX8ZBvF//j36o/dCtwBJ+277d8sdwgHjyhw8mEu3SEqA4bAR54jWLIX0gCmVM1k0dP4g4a2ZVFPudw==",
            "cmkUuid": "b82cab6d-f83b-48c5-bcd2-9fdfcc556b40",
            "keyVersion": "f94237b9-d8d7-4e6f-a3b4-4fe6ac27e4a5"
        },
        "statusCode": 200,
        "success": 1
    }
}
```

返回参数说明

参数	说明
cmkUuid	主密钥的全局唯一标识符。
keyVersionId	密钥版本的全局唯一标识符。
ciphertextBlob	加密后的密文。

5.7.11. 非对称密钥解密

URL

POST https://api.ctyun.cn/apiproxy/v3/account/asymmetric/asymmetricDecrypt

请求参数

参数	是否必填	参数位置	参数类型	说明
algorithm	是	body	String	解密算法： RSAES_OAEP_SHA_256 RSAES_OAEP_SHA_1 RSAES_PKCS1_V1_5
ciphertextBlob	是	body	String	解密密文。
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符。
keyVersionId	是	body	String	密钥版本的全局唯一标识符。

请求示例


```
{
    "keyVersionId": "f94237b9-d8d7-4e6f-a3b4-4fe6ac27e4a5",
    "cmkUuid": "b82cab6d-f83b-48c5-bcd2-9fdfcc556b40",
    "algorithm": "RSAES_OAEP_SHA_256",
    "ciphertextBlob":
"QQA19Zn95F91AAv3nqwmY6KejuvFseSJyuNTu9UzyEH2MyeFPbD5FzergjD2aTSaopSIcnmHhwc8szkrr9KT81
2Q7TAX2GmQIuP2yx6J/t8Yg1p6uxeqBI/X3X11TCZUMVvZQsn9K0GcgCbpD0YTaSgeijONYnI5Fy23Eb9pSw4NC
WUDueqgtMm9nydYCT/mnEzZD53wUuqTd61Ugn1bu9h+/dxJsFztwrxQf4tRptGHY77snr/jDGwqb0JZwSuodrqJ
ZKthuX8ZBvF//j36o/dCtwBJ+277d8sdwgHjyhw8mEu3SEqA4bAR54jWLIX0gCmVM1k0dP4g4a2ZVFPudw==",

"customInfo":{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","user
Id":"5a20ab8588904a25bf35587f82aaae52"}}
}
```


成功返回


```
{
   "statusCode": 800,
  "returnObj": {
   "code": 200,
  "result": {
    "cmkUuid": "e3ed7aed-fee1-467b-ba0a-f6ce018a8894",
    "keyVersionId": "0446b7f5-d6f9-45f0-8743-c674f4f11ba5",
    "plaintext": "SGVsbG8gd29ybGQ="
     },
    "statusCode": 200,
    "success": 1
  }
}
```

返回参数说明

参数	说明
cmkUuid	主密钥的全局唯一标识符。
keyVersionId	密钥版本的全局唯一标识符。
plaintext	解密后的明文，使用 Base64 编码。

5.7.12. 获取非对称密钥公钥

URL

POST https://api.ctyun.cn/apiproxy/v3/account/asymmetric/getPublicKey

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符。
keyVersionId	是	body	String	密钥版本的全局唯一标识符。

请求示例

```
{
    "keyVersionId": "f94237b9-d8d7-4e6f-a3b4-4fe6ac27e4a5",
    "cmkUuid": "b82cab6d-f83b-48c5-bcd2-9fdfcc556b40",

"customInfo":{"type":2,"identity":{"accountId":"9fd671db0cc04c6d9f73e0169ba8d264","user
Id":"5a20ab8588904a25bf35587f82aaae52"}}
}
```

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "cmkUuid": "b82cab6d-f83b-48c5-bcd2-9fdfcc556b40",
```

```
            "keyVersion": "f94237b9-d8d7-4e6f-a3b4-4fe6ac27e4a5",
            "publicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxG8jDD6nmXP3VAjrZU/a\nTvmFeXOE3s7BIasaDPA
1DS45CoLOuO1BvKI4djEAm0xYp19sdKdkbBEXhU4HJAIE\nkPGsX+NATuhPKXIGuJLY/uZ9f8yz/Vq+SpwSmnIo
gC834Wykeyc5psMO9yGkiun6\nvPoPz/VnOJ/y9jCJMrUtIOzRPp76dkY6h0xQmh4pXV+Jb3YWIogRBZOoSiYAj
vQD\n3a6f9+YcNFaBNvd5pSe5kEakHebtIS8c1WvMLZRBI1VaDoLL/2vKrIwij4/Zyk8w\n9QxCcq9CySgz5/jV
0JnABqNjzr9xUTjWZxf6rIwLA/0ixrMTJUHaGH5vi3UJ9/ei\n5wIDAQAB\n-----END PUBLIC KEY-----\n"
        },
        "statusCode": 200,
        "success": 1
    }
}
```


返回参数说明

参数	说明
cmkUuid	主密钥的全局唯一标识符。
keyVersionId	密钥版本的全局唯一标识符。
publicKey	PEM 格式的公钥。

6. SDK 参考

6.1. KMS SDK forJava

KMS-SDK 可以帮助用户通过简单的编程访问 KMS 提供的 API 接口，实现加密解密、签名验签、密钥管理等业务诉求。本文将介绍如何初始化 SDK 以及如何调用接口实现以上功能。

前提条件

- 已购买 KMS 包周期服务。
- 已完成应用接入点创建，获取 KMS 应用接入点地址。
- 已完成访问凭证 AKSK 创建。
- 已完成密钥资源创建。

下载 SDK

请点击下载 SDK：SDK.zip

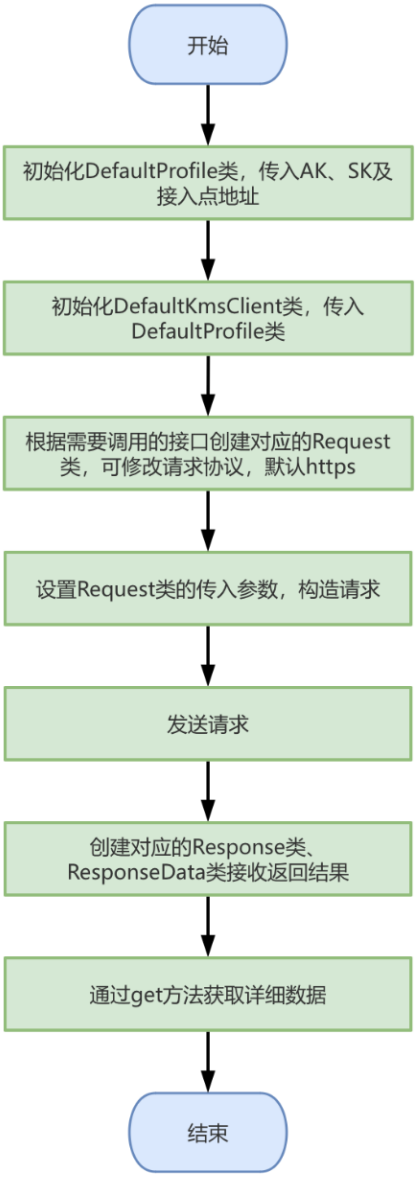
环境依赖

只需将提供的 SDK 导入到开发的 Java 项目中，并在配置中添加以下依赖即可使用 KMS-SDK 的功能。

```
<dependency>
    <groupId>org.kmssdk</groupId>
    <artifactId>ctyun_kms_sdk_java</artifactId>
    <version>1.0</version>
</dependency>
```

调用流程

- 使用 KMS 提供的 Java SDK 调用接口的完整流程如下图所示。
- 首先初始化 DefaultProfile 类，传入参数 AK、SK 和应用接入点地址，然后初始化 DefaultKmsClient 类，传入上一步的 DefaultProfile 类。
- 用户根据需要调用的接口创建相应的 Request 类，并为其构建传入参数，发送 HTTP/HTTPS 请求。
- Request 类可以修改请求协议，默认为 HTTPS 协议。
- 最后通过创建的 Response 类和 ResponseData 类接收响应结果，并使用 get 方法获取详细数据。



初始化 SDK

使用 Java SDK 调用接口时，首先初始化 DefaultProfile 类和 DefaultKmsClient 类。

初始化示例如下：

```
//首先初始化 DefaultProfile 类，new DefaultProfile（String ak, String sk, String ipport），传入用户的 ak、sk 以及接入点地址
DefaultProfile defaultProfile = new DefaultProfile("ae2cc5cc5e8211ea978a186590d96509", "bf9ebf2fb797d85818f46d136a8637388c988813", "127.0.0.1:9091");
//初始化 DefaultKmsClient 类，需要传入上一步的 DefaultProfile 类
DefaultKmsClient defaultKmsClient = new DefaultKmsClient(defaultProfile);
```

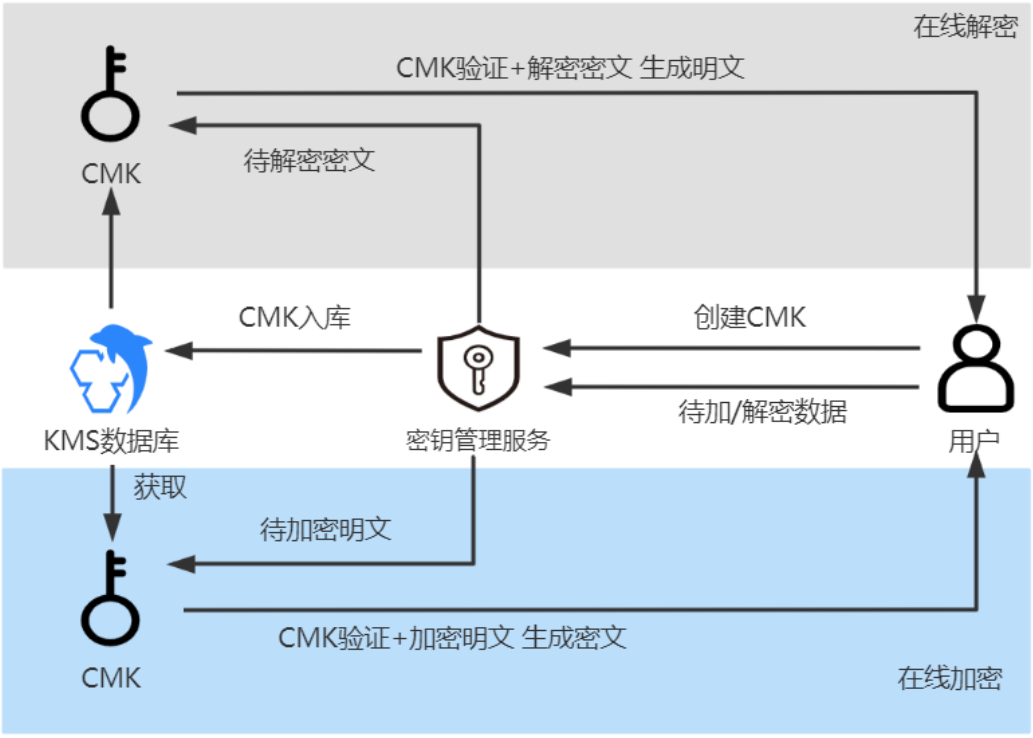
7. 最佳实践

7.1. 使用 KMS 用户主密钥在线加解密数据

KMS 提供针对敏感信息的加密能力，适用于保护小型敏感数据（小于 6KB），如口令、身份信息、证书、后台配置文件等。

通过密钥管理服务 KMS 的在线加密 API，使用用户主密钥（CMK）直接加密敏感数据信息，而非直接将明文存储，确保敏感数据安全。

场景示意图



操作流程（以证书加密为例）

1. 通过 KMS 控制台或者调用 CreateKey 接口，创建一个用户主密钥（CMK）；
2. 调用 KMS 服务的 Encrypt 接口，将明文证书加密为密文证书；
3. 将密文证书部署在服务器上；
4. 当服务器启动需要使用证书时，调用 KMS 服务的 Decrypt 接口将密文证书解密为明文证书。

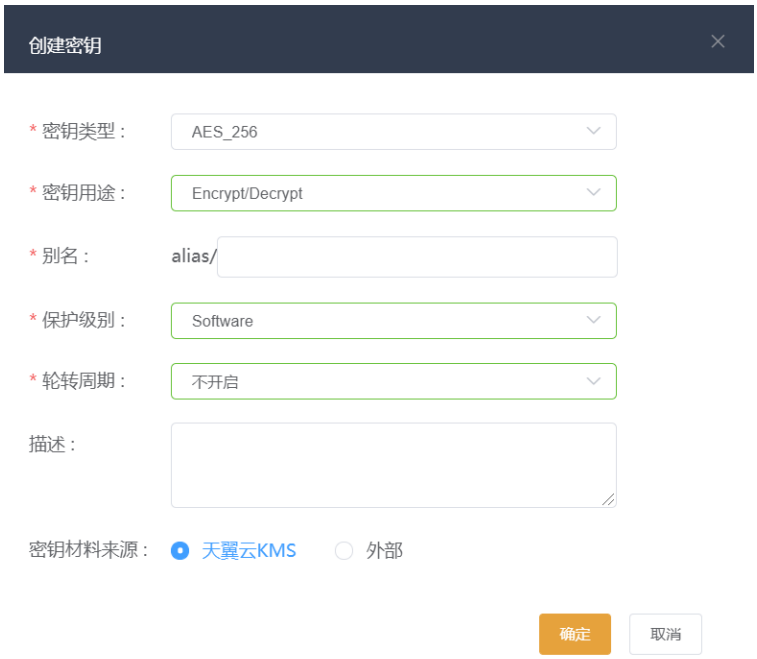
相关 API

可以调用以下 KMS API，轻松完成对数据的加密或解密操作。

API 名称	说明
createKey	创建用户主密钥（CMK）。
encrypt	指定 CMK，直接输入明文数据，由 KMS 在线加密数据。
decrypt	解密由 encrypt 接口加密的数据，不需要指定 CMK 即可完成在线解密。

操作步骤

1. 通过密钥管理服务控制台创建用户主密钥 CMK；



2. 通过 OpenAPI 在线加密接口，对敏感数据进行加密；

请求参数

参数	是否必填	参数位置	参数类型	说明
cmkUuid	是	body	String	主密钥（CMK）的全局唯一标识符。
plaintext	是	body	String	待加密明文（必须经过 Base64 编码）。

请求示例

```
{
```

"plaintext": "SGVsbG8gd29ybGQ=",
"cmkUuid": "241ede22-6261-4617-9caf-10d89990516c"
}

成功返回

{
"code": 200,
"result": {
"ciphertextBlob":
"MDA2NE1qUXhaV1JsTWpJdE5qSTJNUzAwTmpFM0xUbGpZV1l0TVRCa09EazVPVEExTVRaakpqUTVaV00zTm1RM0xXTmpOR010TkRBd1pTMDVaakU1TFdZNU1EQXhOVGczWVdVd1pnPT3oCYiGAy7mNTLitIlJaQ92",
"cmkUuid": "241ede22-6261-4617-9caf-10d89990516c",
"keyVersionId": "49ec76d7-cc4c-400e-9f19-f9001587ae0f"
},
"statusCode": 200,
"success": 1
}

返回参数说明

参数	说明
ciphertextBlob	数据被指定 CMK 的主版本加密后的密文。
cmkUuid	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
keyVersionId	用于加密明文的密钥版本标志符，是指定 CMK 的主版本。

3. 将加密后的数据存储；

根据业务的应用场景，将密文进行存储。

4. 通过 OpenAPI 解密接口，对密文数据进行解密。

请求参数

参数	是否必填	参数位置	参数类型	说明
ciphertextBlob	是	body	String	主密钥（CMK）加密的数据密钥的密文。

成功返回

```
{
    "statusCode": 800,
    "returnObj": {
        "code": 200,
        "result": {
            "cmkUuid": "8bca8f33-d42a-448a-866b-a064f44b29b7",
            "keyVersionId": "73670b28-4eea-4260-b497-ae0334cc0c85",
            "plaintext":
              "sc7280+klUSln3Y9FHdfKGUT+6kPrcIMW41uZQeXxGU="
        },
        "statusCode": 200,
        "success": 1
       }
}
```


返回参数说明

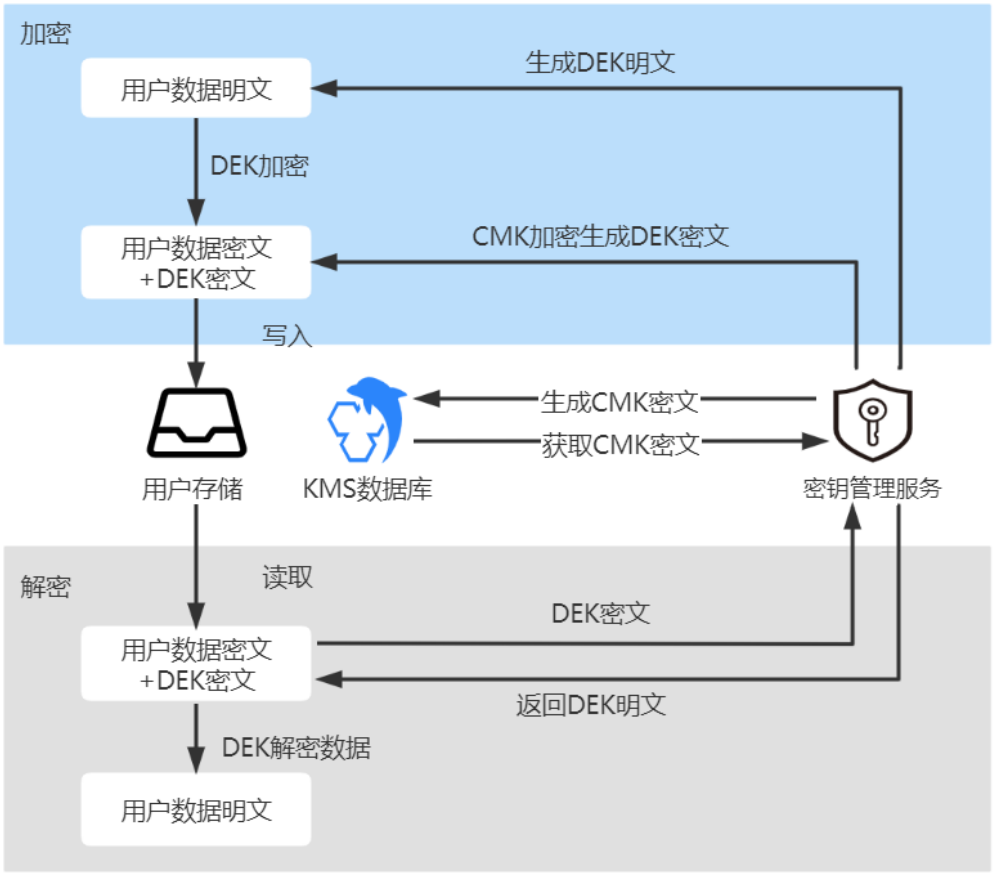
参数	说明
cmkUuid	CMK 的全局唯一标识符。如果请求中的 Cmk_uuid 参数使用的是 CMK 的别名，在响应中会返回别名对应的 CMK 标志符。
keyVersionId	密钥版本 ID。主密钥版本的全局唯一标识符。
plaintext	解密后的明文经过 Base64 编码的后的值。

7.2. 使用信封加密技术实现本地大规模数据加解密

信封加密（Envelope Encryption）是一种应对海量数据的高性能加解密方案。这种技术不再使用用户主密钥（CMK）直接加密和解密数据，而是通过生成加密数据的数据密钥（DEK），将其封入信封中（即通过 CMK 加密）存储、传递和使用，由 KMS 确保数据密钥的随机性和安全性。

实际使用时，用户无需将大量业务数据上传至 KMS 服务端，直接通过离线的数据密钥在本地实现加解密，有效避免安全隐患，保证了业务加密性能的要求。

场景示意图



加密操作流程

1. 通过 KMS 控制台或者调用 CreateKey 接口，创建一个用户主密钥（CMK）；
2. 调用 GenerateDataKey 接口创建一个数据密钥。KMS 会返回一个明文的数据密钥和一个经用户主密钥（CMK）加密的密文数据密钥；
3. 使用明文的数据密钥加密本地文件，产生密文文件，然后销毁内存中的明文数据密钥；
4. 用户将密文数据密钥和密文文件一同存储到持久化存储设备或服务中。

解密操作流程

1. 从本地文件中读取密文数据密钥。
2. 调用 KMS 服务的 Decrypt 接口，将密文数据密钥解密为明文数据密钥。
3. 用明文数据密钥为本地密文文件解密，再销毁内存中的明文密钥。

相关 API

API 名称	说明
createKey	创建用户主密钥（CMK）

API 名称	说明
generateDataKey	生成信封加密的数据密钥，返回数据密钥的明文和经过指定用户主密钥加密的密文
decrypt	解密由 generateDataKey 接口生成的数据密钥密文，不需要指定 CMK

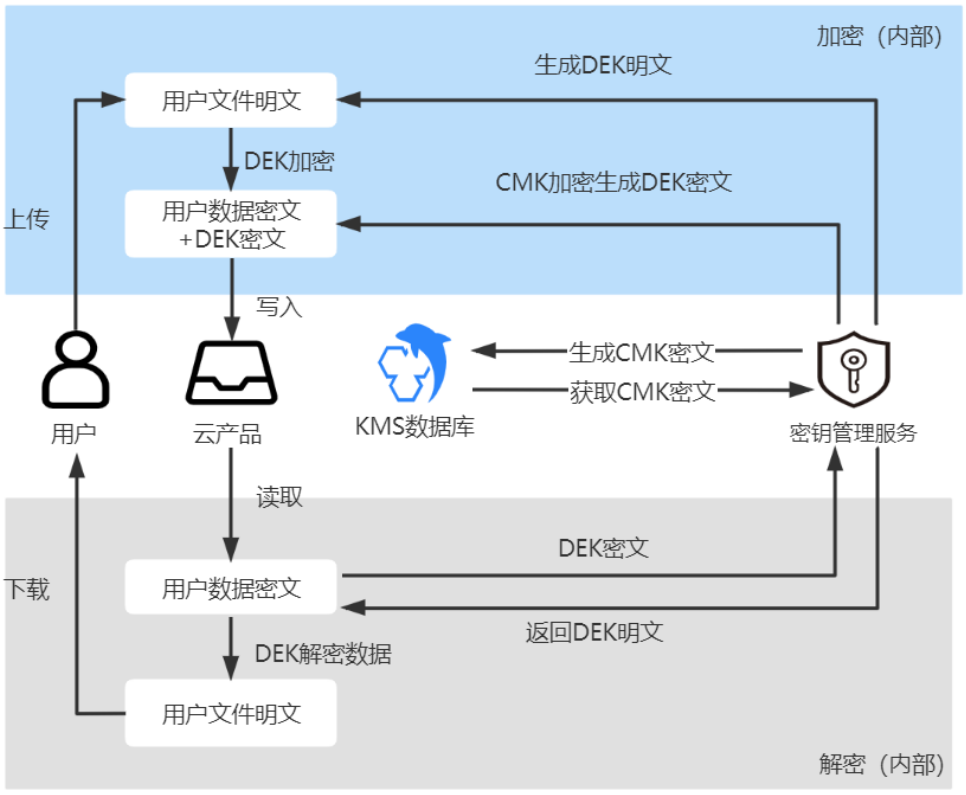
7.3. 云服务通过 KMS 实现服务端加密

密钥管理系统与天翼云产品无缝集成，在云产品中，仅需要选择在 KMS 中托管的主密钥，即可轻松实现对云产品数据的服务端加密。

云产品通过集成 KMS 实现对云上数据的加密存储，密钥由 KMS 托管，满足监管合规要求。整个服务端加密过程对用户透明无感知，只需要开启加密功能并指定密钥即可。同时用户无须自定构建和维护密钥管理基础设施，节省开发成本。

用户可以选择 KMS 为云产品自动创建的默认主密钥加密，也可以选择通过 KMS 创建的用户主密钥。其中默认密钥不收取密钥托管费用。

场景示意图



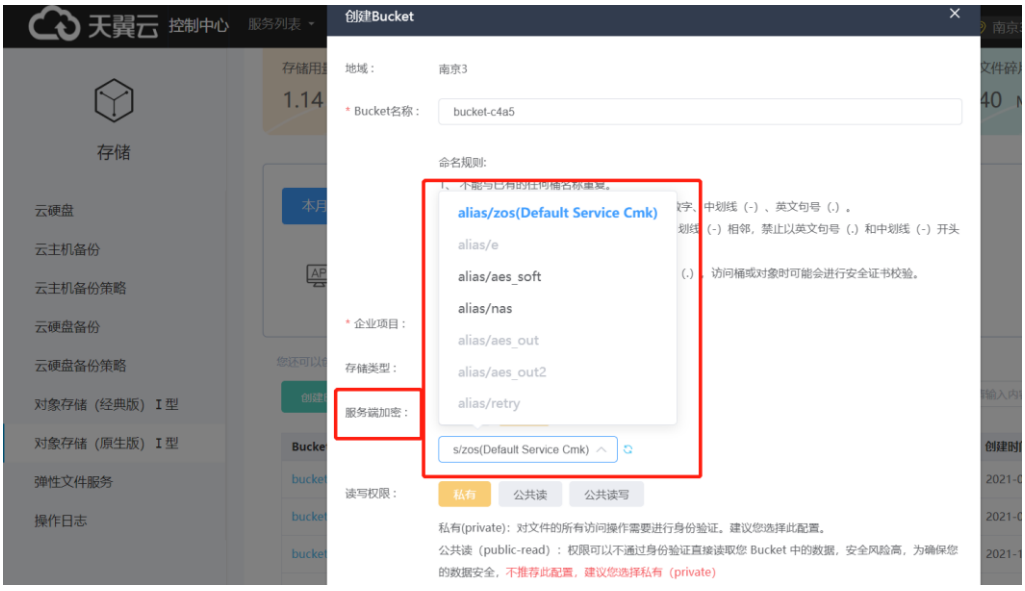
云产品开启服务端加密流程

1. 加密云硬盘；

在创建云硬盘页面，选择开启“磁盘加密”，并在密钥列表中选择加密密钥。

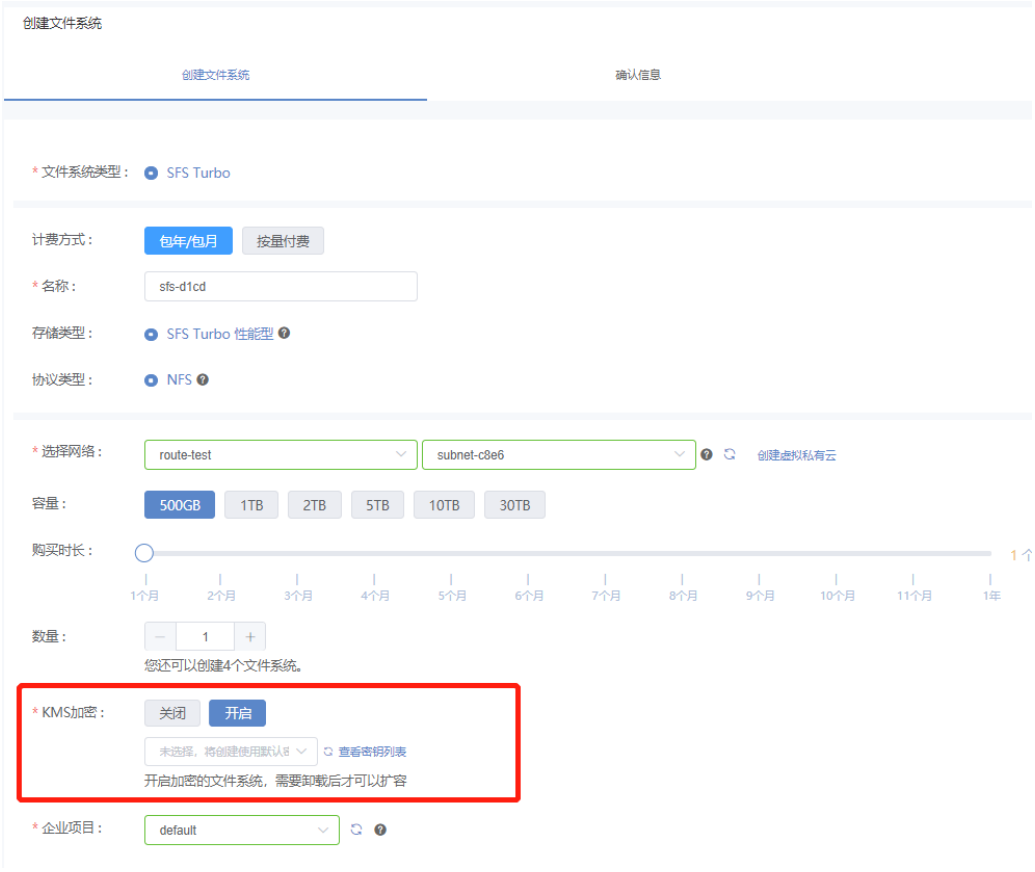
2. 加密对象存储；

在创建对象存储 Bucket 页面，选择开启“服务端加密”，并在密钥列表中选择加密密钥。



3. 加密弹性文件。

在创建文件系统页面，选择开启 KMS 加密，并在密钥列表中选择加密密钥。



7.4. 通过 KMS 实现签名验签

通过密钥管理服务（KMS）创建非对称密钥，签名者通过调用密码运算 API 使用私钥计算消息签名，同时获取公钥并分发至消息接收者，接收者使用公钥对消息进行签名验证。

- 场景特点

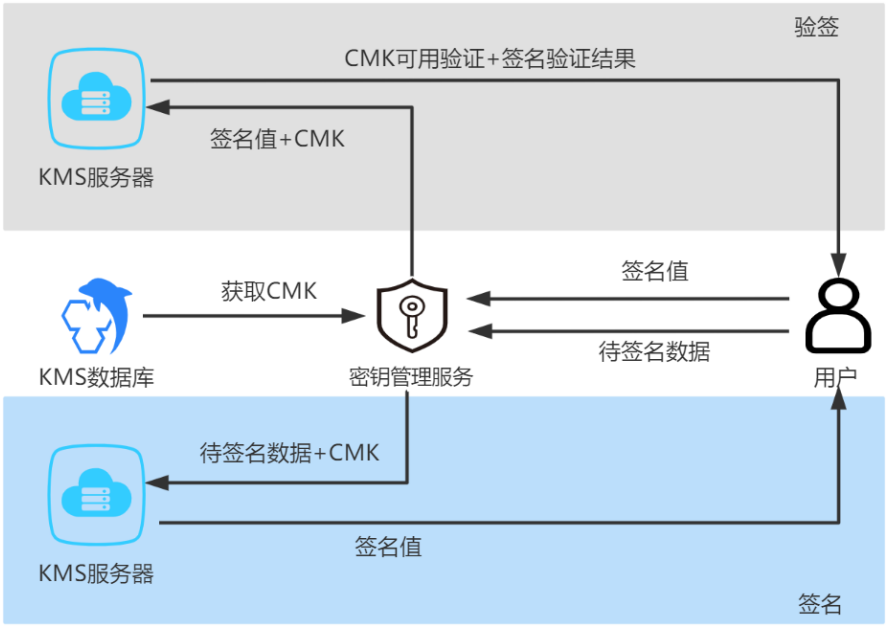
  用于信任程度不对等的系统之间，实现敏感信息的安全传递。

- 优势

  应用广泛：通过非对称密钥实现签名验签，广泛用于数据防篡改、身份认证等相关技术领域；

  安全保障：支持主流的非对称密钥算法并且提供足够的安全强度，保证数字签名的安全性。

场景示意图



操作流程

1. 信息发送者通过 KMS 控制台或者调用 CreateKey 接口，创建一个非对称的用户主密钥（CMK）；
2. 信息发送者通过调用 KMS 的 getPublicKey 接口获取到公钥，并将公钥分发给消息接收者；
3. 信息发送者通过调用 KMS 的 asymmetricSign 接口，使用创建的 CMK 私钥对需要传输的数据生成签名；
4. 信息发送者将签名和数据传递给信息接收者；
5. 信息接收者拿到签名和数据之后，在本地通过 gmssl、openssl、密码库、KMS 的国密 Encryption SDK 等验签方法，使用信息发送者分发的公钥进行验证。特殊需求场景下，也可调用 KMS 的 asymmetricVerify 接口，使用 CMK 进行签名校验。

相关 API

您可以调用以下 KMS API，完成对数据的签名验签处理。

API 名称	说明
createKey	创建用户主密钥（CMK）。
getPublicKey	获取非对称密钥的公钥，可用于离线验证数字签名，或者加密数据。
asymmetricSign	非对称密钥的私钥运算：产生数字签名。

API 名称	说明
asymmetricVerify	非对称密钥的公钥运算：验证私钥产生的数字签名。

7.5. 通过密钥轮转加强密钥使用的安全性

KMS 提供密钥轮转功能实现密钥版本化，从而加强密钥使用的安全性，有效提升业务数据加密的安全性。本文为您介绍如何配置对称密钥和非对称密钥的轮转。

密钥轮转的必要性

- 密码合规要求

  相关行业标准中明确规范，要求密钥进行周期性轮转。
- 减少每个密钥版本加密的数据量，降低密码分析攻击风险

  一个密钥的安全性与被它加密的数据量呈反相关。数据量通常是指同一个密钥加密的数据总字节数。通过定期轮转密钥，可使每个密钥具有更小的密码分析攻击面，使加密方案整体具有更高的安全性。
- 减少密钥破解的时间窗口

  如果在定期轮转密钥的基础上，将旧密钥加密的密文数据用新密钥重新加密，则轮转周期即为一个密钥的破解时间窗口。这意味着恶意者只有在两次轮转事件之间完成破解，才能拿到数据。

密钥版本概述

KMS 中的用户 CMK 支持多个密钥版本。每一个密钥版本是一个独立生成的密钥，同一个 CMK 下的多个密钥版本在密码学上互不相关。

对称密钥版本

密钥版本可通过自动轮转策略，由系统自动生成，对称密钥的版本分为主版本和非主版本。

- 一个对称密钥版本包含一个主版本和多个非主版本。密钥创建后 KMS 会生成初始密钥版本并将其设置为主版本，轮转后会生成一个新的密钥版本，并将新的密钥版本设置为主版本，原版本设置为非主版本；
- 在调用对称密钥进行加解密操作时，KMS 默认使用主版本实现；
- 密钥轮转产生新的主版本后，KMS 不会删除或禁用非主版本，它们需要被用作解密数据。

非对称密钥版本

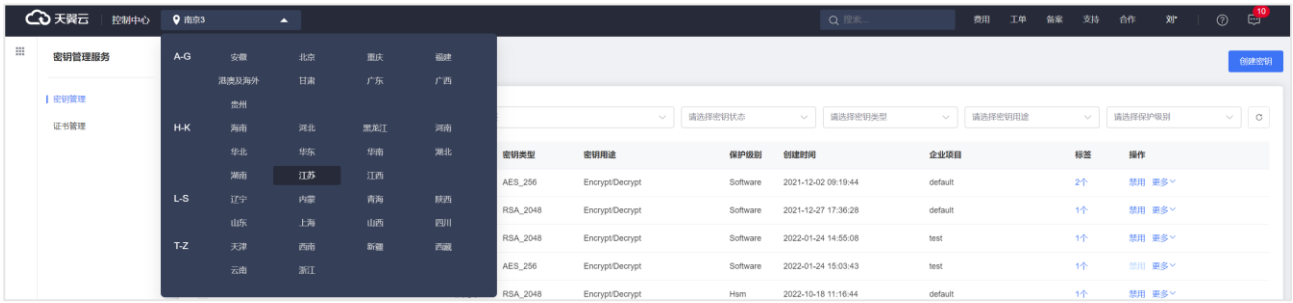
非对称密钥不支持自动轮转，需人工创建新的密钥版本，版本创建后立即生效。

- 非对称的用于主密钥没有主版本（PrimaryKeyVersion）的概念，因此使用非对称密码运算的接口除需指定用户主密钥标志符（或别名）之外，还需指定密钥版本。

操作步骤

设置自动轮转（对称密钥）

1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；



3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 定位待设置的对称密钥，点击**密钥 ID**，进入**密钥详情页**；
5. 在**密钥版本**区域，点击**设置轮转策略**；

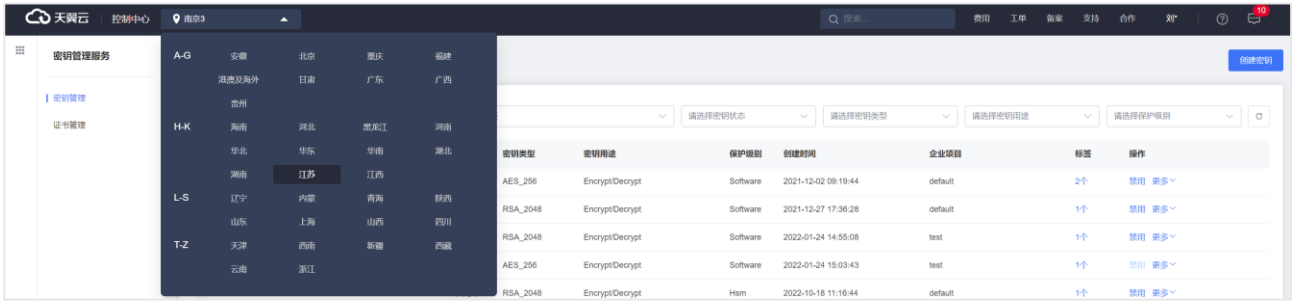
6. 在**设置轮转策略**对话框，选择轮转周期，**30 天**、**90 天**、**180 天**，或**自定义天数**；

7. 设置了自动轮转策略后，将显示密钥下次轮转时间。点击**确定**完成设置；

8. 可通过相同的步骤更改轮转周期，也可取消轮转策略。

创建密钥版本（非对称密钥）

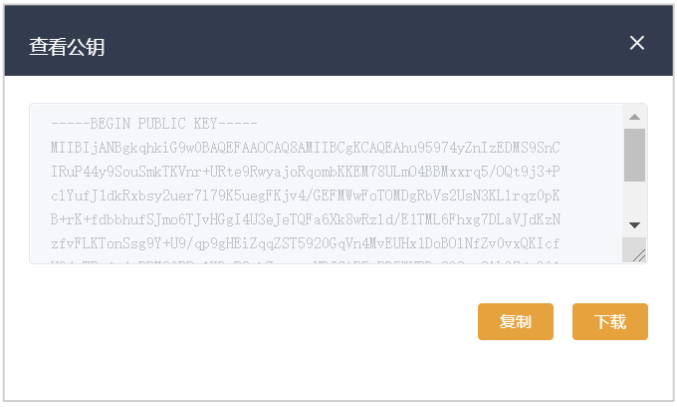
1. 登录密钥管理服务控制台；
2. 在页面最上方的导航栏选择密钥所在的区域；



3. 在左侧导航栏，单击**密钥管理服务**，进入**密钥列表**；
4. 定位待设置的非对称密钥，点击**密钥 ID**，进入**密钥详情页**；
5. 在**密钥版本**区域，点击**创建密钥版本**；

6. 在弹出的对话框内，点击**确定**；

7. 在密钥版本列表，可查看密钥版本 ID、创建日期。点击**查看公钥**，在弹出的对话框，可**复制**或**下载**公钥。



8. 常见问题

8.1. 计费类

密钥管理服务的计费方式是什么？

KMS 产品当前支持包周期版本及按需版本，对应两种计费模式：

- 包年/包月计费：一种预付费模式，即先付费再使用。您可根据业务需要，选择合适的包周期服务版本（基础版、企业版），一次性支付一个月/多个月/一年/多年的费用，支付成功后，KMS 服务资源将被系统分配给用户使用，直到超过保留期后被系统回收。
- 按使用量计费：一种后付费模式，即先使用再付费。在结算时会按照您在按需版本中，实际资源使用量收取费用，如密钥数量、API 调用量等。

密钥管理服务的计费项是什么？

KMS 包周期版服务的计费项包括基础版、企业版。

- 基础版提供软件保护等级服务，支持客户构建专属的密钥库，具备高度可扩展性；同时提供极简的密码运算接口能力，满足应用的安全快速集成。
- 企业版提供硬件保护等级服务，底层对接使用经国家密码管理局认证的密码机硬件，提供更高安全与合规等级保证的资源管理及密码运算服务，满足监管机构的检测认证要求。

KMS 按需版服务的计费项包括密钥托管费、API 调用费。

- 密钥托管费：密钥创建后托管在 KMS 服务产生的费用，按照密钥类型、密钥个数以天为周期进行计费。
- API 调用费：密钥创建后通过接口调用产生的请求费用，按照 API 请求次数计费，每个账户每月有 20000 次的免费请求次数，超过 20000 次后开始计费。

具体的计费项详情请参考计费项。

密钥管理服务有关密钥管理的接口调用，是否算在 API 调用费中进行计费？

当您正在使用密钥管理按需版服务，对于 API 接口调用所产生的费用，计算规则如下：

密钥管理相关接口调用产生的调用次数不计费。

密钥管理服务中 API 调用计费项，只统计密码运算类接口的调用次数并计费，密码运算类接口如下：

API	说明
encrypt	在线加密，使用指定用户主密钥加密数据，用于少量数据（不多于 6KB）的在线加密。
generateDataKey	生成信封加密的数据密钥，返回数据密钥的明文和经过指定用户主密钥加密的密文。
generateDataKeyWithoutPlaintext	生成信封加密的数据密钥，返回经指定用户主密钥加密的密文。
exportDataKey	导出数据密钥，返回经指定公钥加密的数据密钥的密文。
generateAndExportDataKey	产生并导出数据密钥，生成信封加密的数据密钥，返回经指定用户主密钥加密的密文和经指定公钥加密的密文。
decrypt	解密 Encrypt 或 GenerateDataKey 接口产生的密文，不需要指定用于解密的用户主密钥。
reEncrypt	对密文进行转加密，即先解密密文，然后将解密得到的数据或者数据密钥使用新的主密钥再次进行加密，返回加密结果。待转加密的密文可以为对称加密或非对称加密返回的密文数据。
asymmetricSign	非对称密钥的私钥运算：产生数字签名。
asymmetricVerify	非对称密钥的公钥运算：验证私钥产生的数字签名。
asymmetricEncrypt	非对称密钥的公钥运算：加密数据。
asymmetricDecrypt	非对称密钥的私钥运算：解密公钥加密的数据。
getPublicKey	获取非对称密钥的公钥，可用于离线验证数字签名，或者加密数据。

服务因欠费或到期后，密钥是否还能进行解密？

不可以。用户欠费或服务到期后，KMS 会冻结服务，所有对于 KMS 的访问均被限制，对密钥解密接口同样无法实现调用。

因此，为避免您的业务因无法解密造成影响，请及时进行充值及续费，确保 KMS 服务可用。

8.2. 操作类

如何使用密钥实现数据加解密？

KMS 提供了 REST（Representational State Transfer）风格 API，支持通过 HTTPS 请求调用。用户可使用提供的 API 实现加解密运算等操作。下面以使用用户主密钥进行数据加解密为例介绍实现过程：

加密流程（以加密证书为例）：

1. 通过 KMS 控制台或者调用 CreateKey 接口，创建一个用户主密钥（CMK）；
2. 调用 KMS 服务的 Encrypt 接口，将明文证书加密为密文证书；
3. 将密文证书部署在服务器上；
4. 当服务器启动需要使用证书时，调用 KMS 服务的 Decrypt 接口将密文证书解密为明文证书。

如何导入外部自带密钥？

创建密钥后，首先进入密钥详情页获取导入主密钥材料的参数，参数包括加密公钥及导入令牌，用户使用获取到的公钥加密自带密钥材料，然后在控制台密钥详情页，根据页面提示上传自带密钥材料即可。

从 KMS 获取到的导入令牌与加密密钥材料的公钥具有绑定关系，一个令牌只能为其生成时指定的主密钥导入密钥材料。导入令牌的有效期为 24 小时，在有效期内可以重复使用，失效以后需要获取新的导入令牌和加密公钥。

如何删除密钥？

KMS 不支持立即删除，仅支持计划删除，即用户需设置预删除周期（自定义 7~30 天），系统会在到期时自动删除密钥，预删除期间密钥仍托管至系统中，但无法被调用实现加解密等相关功能。

设置密钥计划删除后，密钥将不再产生费用。若您在预删除期间发现密钥仍需继续使用，则可以选择取消计划删除，使密钥重新变为可用。

为什么 KMS 不支持立即删除密钥？

由于密钥删除后不可恢复，一旦删除密钥，所有使用该密钥加密的数据均无法解密，因此删除密钥的操作需要非常谨慎，KMS 通过计划删除的机制，即执行计划删除操作后，密钥状态变为待删除，密钥不会立即删除，系统会根据用户设置的预删除周期推迟删除密钥。到达执行时间点，系统才会真正删除密钥，在此之前用户均可以取消删除计划。KMS 通过这种方式来减少用户误操作所带来的损失。

处于待删除状态的密钥不可用，无法用于加解密、产生数据密钥等。

如果您不再使用密钥，推荐您先禁用密钥，确保不影响您的业务后再通过计划删除密钥来进行删除。为避免误删，您可以开启删除保护功能。

KMS 是否支持国密算法？

支持。KMS 支持创建 SM2、SM4 类型的密钥，适用于数据加解密、签名验签等场景。

密码算法大类	密码算法子类	保护级别	是否支持加解密	是否支持签名验签
对称密钥	AES_256	Software HSM	支持	不支持
	SM4	HSM	支持	不支持
非对称密钥	RSA_2048	Software HSM	支持	支持
	SM2	HSM	支持	支持

软件保护级别和硬件保护级别的区别是什么？

软件保护级别的密钥通过软件模块进行保护，其根密钥通过公钥加密形成密文存储在软件文件系统中；而用于解密根密钥密文的私钥，通过对称加密形成密文存储在文件系统中的不同位置；同时对称密钥也存储在不同位置，进而增加系统根密钥的安全性；

硬件保护级别的密钥通过专用硬件保护密钥，硬件根密钥需要存储密码机的内部密钥索引，通过索引确认根密钥；所有涉及根密钥使用的过程均在密码机内部完成，包括加密解密等。

密钥自动轮转周期的可设置范围是什么？

对称密钥支持设置自动轮转周期，周期最短为 7 天，最长为 730 天（2 年）。

对称密钥支持设置自动轮转，系统根据自动轮转周期自动生成新的密钥版本，并将最新的版本设置为主版本，原密钥版本作为非主版本保存在 KMS 中，KMS 不会删除或禁用非主版本，它们需要被用作解密数据。

非对称密钥是否支持自动轮转？

非对称密钥不支持设置自动轮转，可以手动创建新的版本。

由于公私钥使用场景的特殊性，KMS 不支持对非对称的用户主密钥进行自动轮转。可在指定用户主密钥中人工创建新的密钥版本，生成全新的一对公钥和私钥。

除此之外，和对称类型的用户主密钥不同，非对称的用于主密钥没有主版本（PrimaryKeyVersion）的概念，因此使用非对称密码运算的接口除需指定用户主密钥标志符（或别名）之外，还需指定密钥版本。

什么情况下需要导入外部密钥？

当用户拥有自己的密钥材料，需要继续使用该密钥材料实现数据加解密，比如用户需要将本地加密数据迁移到云上时，云上云下共用同一个密钥材料，此时可以将密钥材料导入至 KMS 中进行托管，便于后续使用。

当您选择密钥材料来源为外部，使用您自己导入的密钥材料时，需要注意以下几点：

- 请确保您使用了符合安全要求的随机源生成密钥材料；
- 在使用导入密钥时，需要对自己密钥材料的可靠性负责；
- 请保存密钥材料的原始备份，以便在意外删除密钥材料时，能及时将备份的密钥材料重新导入 KMS。

什么类型的密钥支持导入外部密钥材料？

当前 AES_256 类型的对称密钥，支持导入外部密钥材料。

您可通过控制台创建密钥，选择 AES_256 类型的对称密钥后，密钥材料来源选项选择**外部**，并勾选“我了解使用外部密钥材料的方法和意义”，点击确定即可完成密钥创建；

密钥创建成功后，您需要进入密钥详情页，进行密钥材料导入操作。导入密钥材料前需要先获取导入材料参数，包括加密公钥、导入令牌，具体操作步骤可参考用户指南中的相关章节。

外部导入的密钥材料是否支持自动轮转？

不支持。外部导入的密钥材料不支持自动轮转，无密钥版本概念。

导入密钥材料时，可以设置可以设置密钥材料过期时间，密钥材料过期后，KMS 将自动删除密钥材料，但该主密钥及其元数据仍然保留。

外部导入的密钥材料支持手动删除，但该主密钥及其元数据仍然保留。

当密钥材料被误删或已经过期导致密钥不可用，如何处理使密钥恢复可用？

密钥材料被删除或过期时，可以再次导入相同的密钥材料，成功后密钥将恢复可用。您需要自行备份密钥材料，以便密钥材料失效或误删除时进行重新导入。

用户主密钥 CMK 与密钥材料具有关联性，当您将密钥材料导入某个 CMK 时，该 CMK 与该密钥材料永久关联，不能将其他密钥材料导入该 CMK 中，即便密钥材料已经过期或者被删除。

当用户主密钥的密钥材料删除或过期后，是否可以导入其他的密钥材料到该主密钥中？

不可以。用户主密钥包含密钥元数据（密钥 ID、密钥别名、描述、密钥状态与创建日期）和用于加解密数据的密钥材料。

将密钥材料成功导入主密钥后，该主密钥与密钥材料永久关联，不能再将其他密钥材料导入该主密钥中。

具备相同密钥材料的不同用户主密钥，是否可以相互加解密数据？

不可以。用户主密钥包含密钥元数据（密钥 ID、密钥别名、描述、密钥状态与创建日期）和用于加解密数据的密钥材料。

用户主密钥具有唯一性，使用主密钥加密的数据，只能用相同的主密钥解密。即使其他主密钥具有相同的密钥材料，也无法解密该主密钥加密的数据。

用户主密钥别名的作用是什么？

为密钥创建别名方便用户管理密钥，一个别名对应唯一的用户主密钥。在通过 openAPI 调用 KMS 服务接口时，参数中的密钥 ID 可以用别名代替。

别名必须依附于用户主密钥存在，其特点如下：

- 一个用户主密钥下可以绑定多个别名，删除别名不会删除其关联的用户主密钥。
- 别名不可修改。您可以通过为一个用户主密钥创建新的别名，并且删除旧的别名来达到修改主密钥别名的目的。
- 可以调用 UpdateAlias 接口更改别名绑定的用户主密钥，而不会影响用户主密钥。
- 默认主密钥的别名不能删除和添加。

用户主密钥与别名的对应关系是什么？

别名作为用户主密钥的可选标识，必须与密钥关联。同一个账户在一个地域中的别名具有唯一性。

一个用户主密钥可以绑定多个别名，同一个别名只能指向唯一的用户主密钥。

默认主密钥的别名不支持删除、更改。

别名支持修改吗？

别名不支持直接修改，您可以通过创建新的别名，并删除旧的别名来达到修改别名的目的。

为密钥创建新的别名时，不会影响已有的其他别名。删除旧的别名前，请确保该别名已不再使用，否则可能会导致数据加密失败。

默认主密钥的别名不支持删除和添加。

删除别名是否会影响用户主密钥的使用？

别名是用户主密钥的可选标识，支持删除别名，删除别名不会删除其关联的用户主密钥。

创建别名的作用是可以在调用 API 接口时，使用别名来代替密钥 ID。因此删除如果仍在使用别名作为 api 调用参数时，删除别名会导致服务调用失败，请确保预删除的别名已不再使用。

同一资源池内的用户主密钥，是否可以设置相同的别名？

不可以。同一个账户在同一个地域中的别名具有唯一性，每个别名只能指向同地域的一个用户主密钥，但是每个用户主密钥可以绑定多个别名。

相同的别名可以绑定不同资源池内的用户主密钥。

为用户主密钥设置自动轮转的目的是什么？

KMS 提供密钥轮转功能，支持通过密钥版本化和定期轮转来加强密钥使用的安全性，有效提升业务数据的安全性。

通过密钥轮转，可以减少每个密钥版本加密的数据量，降低没密码分析攻击风险；

密钥轮转可以减小破解密钥的时间窗口。应对密码分析攻击风险的有效实践是在定期轮转密钥的基础上，将旧密钥加密的密文数据使用新版本的密钥重新加密，这意味着如果想要破解密码拿到明文数据，需要在密钥轮转周期内完成密码破解。密钥轮转周期即为密钥破解时间窗口，该窗口越小，破解难度越大。

密钥经过轮转产生新的密钥版本后，是否会影响旧数据的解密？

不影响。密钥轮转产生新的版本后，加密数据时将使用新的版本；同时旧版本不会删除或禁用，在解密旧数据时，需要使用旧版本密钥完成。

对称密钥版本分为主版本和非主版本。主版本是 CMK 的活跃加密密钥（Active Encryption Key）。每个 CMK 在任何时间点上有且仅有一个主版本。调用 GenerateDataKey、Encrypt 等加密 API 接口时，KMS 使用指定 CMK 的主版本对明文进行加密。非主版本是 CMK 的非活跃加密密钥（Inactive Encryption Key）。每个 CMK 可以有零到多个非主版本。非主版本历史上曾经是主版本，在当时被用作活跃加密密钥。密钥轮转产生新的主版本后，KMS 不会删除或禁用非主版本，它们需要被用作解密数据。

非对称密钥是否支持自动轮转？

不支持。由于非对称密钥公钥使用场景的特殊性，KMS 不支持对非对称密钥进行自动轮转。用户可以人工手动创建新的版本，生成全新的一对公钥和私钥。

和对称类型的用户主密钥不同，非对称的用于主密钥没有主版本（PrimaryKeyVersion）的概念，因此使用非对称密码运算的接口除需指定用户主密钥标志符（或别名）之外，还需指定密钥版本。

使用密钥进行加密数据时，是否需要指定密钥版本？

使用密钥加密是否需要指定密钥版本与密钥类型有关。

调用对称密钥加密数据时，不需要指定密钥版本，系统会默认使用最新的主版本进行数据加密。

调用非对称密钥加密数据时，除了要指定主密钥外，还需要指定密钥版本。

8.3. 管理类

是否可以导出用户主密钥？

不可以。为确保用户主密钥的安全，用户只能在 KMS 中创建，并通过 API 接口调用实现加密等操作，无法导出用户主密钥。

创建密钥时，若您选择密钥材料来源于天翼云，则 KMS 系统会自动为用户主密钥生成密钥材料，且密钥材料无法单独删除、不可导出，仅支持随用户主密钥一并删除；

创建密钥时，若您选择密钥材料来源于外部，则支持手动删除密钥材料。

哪些云服务支持密钥管理系统加密数据？

KMS 服务无缝对接云硬盘、对象存储和弹性文件、数据库产品，提供服务端加密能力。您只需要在创建云硬盘时，勾选“加密”功能，则可一键开启硬盘加密功能，加密过程透明无感知。

产品底层通过信封加密的方式，实现云产品中数据的加密。

用户自建用主密钥与默认主密钥有何区别？

用户主密钥：是用户通过控制台或 API 来创建的用户主密钥。您可以对用户密钥进行创建/设置别名/上传自带密钥材料/启用/禁用/轮转/版本管理/删除等操作。用户主密钥按照标准资费进行计费。

默认主密钥：是用户首次通过云服务调用 KMS 实现加密时，由系统自动生成的主密钥，别名以云产品命名，如“alias/ecs”。不支持禁用/删除/轮转/上传自带密钥材料等操作。默认主密钥免费提供密钥管理服务，API 调用费与用户主密钥一同统计收费。

如果用户主密钥被禁用/删除，用户数据是否还可以解密？

不可以。被禁用的密钥无法用于加密和解密。若想继续使用密钥解密，则需将密钥变为启用中。

若用户主密钥被彻底删除，KMS 将不再保留任何该密钥的数据，使用该密钥加密的数据将无法解密；

因此密钥管理不支持立即删除操作，仅支持计划删除，在用户设置的计划删除的期限到达时删除密钥，用户可以通过 KMS 界面取消计划删除用户主密钥。

若用户主密钥是通过 KMS 导入的密钥，且仅删除了密钥材料，则可以将本地备份的密钥材料再次导入原来的空密钥，回收用户数据。若密钥材料没有在本地备份，则无法回收用户数据。

用户最多可以创建多少个主密钥？

对于对称密钥，暂不限制创建个数。对于非对称密钥，目前限制密钥的版本数量，即同一用户在同一资源池最多创建 50 个版本。