IoT Hub

Product Introduction

Product Documentation

Contents

Product Introduction

Overview

Last updated：2021-09-10 10:30:00

Tencent Cloud Internet of Things Hub (IoT Hub) provides a secure, stable, and efficient connection platform that helps developers quickly achieve reliable and high-concurrency data communications among devices, user applications and cloud services at low costs. In other words, IoT Hub can realize cross-device interaction, device data reporting and configuration distribution. In addition, by opening up the link between device data and Tencent Cloud products using the rule engine, it allows for the quick and easy storage, real-time computation and intelligent processing and analytics of massive amounts of data.

Service Architecture
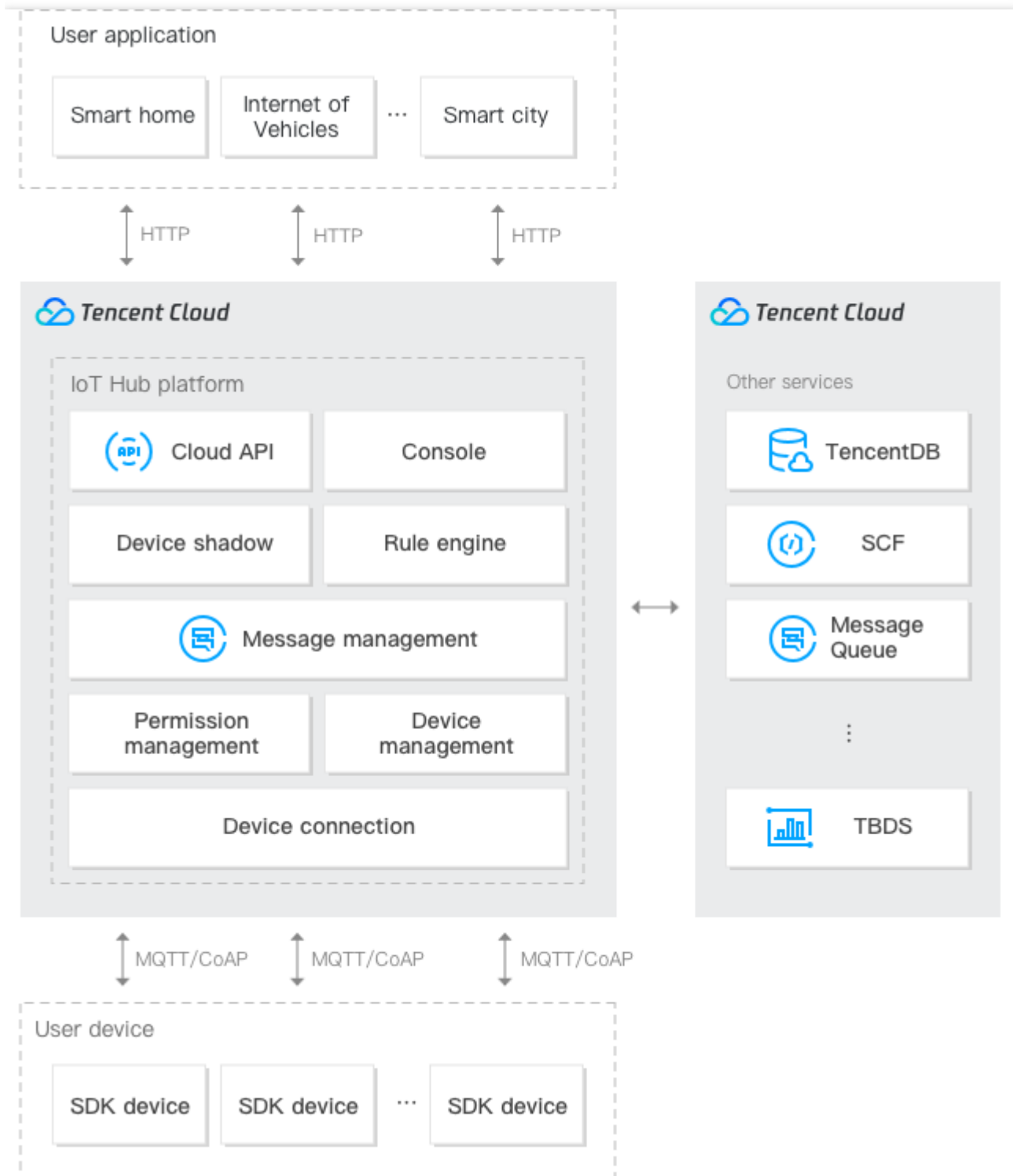
Devices can be connected to IoT Hub

User devices can be connected to IoT Hub through SDKs. The underlying data transfer is based on MQTT or CoAP protocols, which effectively lowers the consumption of network bandwidth. In terms of security, IoT Hub introduces secure network transfer protocols (TLS and DTLS) to prevent risks such as unauthorized access, data theft, and tampering. Taking into account the diversity of devices and use cases, it supports both asymmetric encryption

(authentication based on device certificates for scenarios with high security requirements) and symmetric encryption (authentication based on keys for resource-constrained devices).

Messages can be published and subscribed to by devices through SDKs

In order to isolate device data for security purposes, IoT Hub currently limits that devices can only publish and subscribe to messages in their own topics, but they can access messages of other entities by configuring the rule engine.

The rule engine can be configured in the console to enable devices to access messages of other entities

At present, the rule engine supports operations in SQL-like syntax, which implement message communication between devices through "repub" (republishing messages) and device message forwards to third-party services through "forward" (forwarding messages to servers). Meanwhile, device messages can be forwarded to Tencent Cloud services such as TencentDB, CTSDB, and message queue. Forwarding messages to SCF, TBDS, RayData, and BI will be supported soon.

Device messages can be interconnected with third-party services

As devices are connected only to IoT Hub, IoT Hub can quickly write specified device messages to Tencent Cloud CMQ or CKafka queues, with the message queue feature enabled. From there, third-party services can get and consume the data through the SDK of CMQ or CKafka queues, achieving async message communication between devices and third-party services.

Device shadows can effectively achieve two-way sync of configuration and status data between devices and applications

On the one hand, configuration parameters can be set for device shadows through TencentCloud API, so that when devices are connected or online, they can get the configuration parameters from the shadows. When the status of a device is queried, it is sufficient to query its shadow without having to perform direct network communication with the device.

Devices can be managed through TencentCloud API

IoT Hub provides convenient SDK tools to enhance IoT device management capabilities. These tools enable quick and batch creating, querying, and operation on the backend, greatly improving the efficiency. Currently, Python, PHP, and Java toolkits are supported.

Features

Last updated：2021-08-19 17:51:00

Device Connection

Connection through SDK

IoT Hub currently supports SDKs for various platforms such as Linux and Android. For SDK download addresses, please see SDK for C Download.

Transfer protocol

- TCP- and TLS-based MQTT (encrypted connection) is a mainstream IoT communication protocol and is appropriate for communication between devices or receiving reverse control signals and configurations.
- UDP- and DTLS-based CoAP (encrypted connection) has less consumption and requirements for resources, which is applicable for pure data reporting.

Security protocol

IoT Hub performs two-way authentication and encrypted data transfer between client and server based on security protocols such as TLS and DTLS, so as to prevent risks including unauthorized access, data theft, and tampering. Taking into account the diversity of devices and use cases, it supports both asymmetric encryption (authentication based on device certificates for scenarios with high security requirements) and symmetric encryption (authentication based on keys for resource-constrained devices). Authentication at the device granularity ensures the confidentiality of cloud-to-device and device-to-cloud messages.

RTOS portability

IoT Hub's SDK supports cross-platform porting and detachment of framework from hardware abstraction layer, enabling quick and easy connection to IoT Hub from different platforms.

Device firmware update

When firmware has security risks or functional problems, IoT Hub servers can perform OTA updates to eliminate dangers and reduce security risks.

Gateway product connection

IoT Hub supports the creation of gateway and subdevice products. You can bind a gateway device and the corresponding subdevice, and then the gateway device can connect, disconnect, and send/receive messages on behalf of the subdevice over the MQTT protocol.

Device Management

Lifecycle management

Devices can be registered, added, deleted, or terminated in the console. This can also be done through the SDK toolkits for Python, PHP, and Java, which is faster and more efficient.

Device status

Device status can be monitored throughout the entire process, with instant notifications for any status changes.

Group management

IoT Hub supports group management for devices under different products to meet the multi-level management needs of different types of devices in different business scenarios.

Log collection

IoT Hub can collect and report the upstream and downstream communication logs, message content logs, and SDK debugging logs of devices to meet the query requirements in multiple business scenarios.

Device Communication

The topics that devices can publish and subscribe to are managed by permission control, and all devices under the same product have the same topic class permissions. For data transfer over MQTT, QoS=0 and QoS=1 message features are supported. Plus, messages can be stored offline, and the rule engine can swiftly implement message communication between devices.

Device Shadow

Device shadow is essentially a copy of device data in JSON format cached on the server and is mainly used to save:

- Current device configurations
- Current device status

As a medium, device shadow can effectively implement two-way data sync between device and user application:

- For device configuration, the user application does not need to directly modify the device; instead, it can modify the device shadow on the server, which will sync modifications to the device. In this way, if the device is offline at the time of modification, it will receive the latest configuration from the shadow once going back online.

- For device status, the device reports the status to the device shadow, and when users initiate queries, they can simply query the shadow. This can effectively reduce the network interactions between the device and the server, especially for low-power devices.

Rule Engine

Syntax rules

IoT Hub supports SQL-like syntax and basic semantic operations. The contents of device messages can be parsed, filtered, extracted, and reintegrated through simple syntax, with the results forwarded to Tencent Cloud's backend services such as storage, function, and TBDS for seamless data connection.

Connection between devices

In order to isolate device data, devices can only publish and subscribe to messages in their own topics. Message connection between devices can be achieved through the repub feature of the rule engine.

Device message forwarding to third-party

The rule engine can configure directly forwarding device messages to third-party services, thereby quickly enabling communication between the device and the connecting party's backend services.

Device-Cloud connection

Tencent Cloud offers corresponding services (such as TencentDB, SCF, message queue, and TBDS) for scenarios where users require further processing of device data (such as persistent storage, function computing, and big data analysis). In addition, the direct connection between IoT Hub and these Tencent Cloud services will be available soon.

Message Queue

As devices are connected only to IoT Hub, IoT Hub can write specified device messages to Tencent Cloud CMQ or CKafka queues. From there, third-party services can get the device messages through the SDK APIs of CMQ or CKafka, enabling async message communication between devices and third-party services. Based on this, data storage, computational analysis, and device control logic can be implemented on the backend.

Collaboration Management

IoT Hub supports secure access, use, and management of cloud account resources through CAM. Isolation and collaboration of IoT Hub resources are implemented through identity and policy management of sub-accounts and

collaborators.

Data Processing

Real-Time computing

In the field of IoT, massive amounts of data is reported in real time, and core businesses have high requirements for the timeliness of data monitoring, making stream computing and real-time computing significant for such use cases. The rule engine forwards device data to CKafka in real time, which is connected to Storm/SparkStreaming for stream computing. This helps you implement real-time computation of device data.

Intelligent processing

IoT Hub enables data connection with TBDS. TBDS' extraordinary capabilities in data discovery, analysis, and mining enable users to quickly process data from billions of IoT devices, tap into the value of data, increase efficiency, and seize market opportunities.

Visualization

IoT Hub can access Tencent Cloud RayData, a service for big data visualization. Powered by the real-time data rendering technology, you can visualize, contextualize, and interact with massive amounts of data reported from devices, achieving personalized data management and usage.

Strengths

Last updated：2021-08-19 17:51:00

Security

IoT Hub integrates secure network transfer protocols (TLS or DTLS) for device connection and data transfer. In this way, every device has its own certificate for authentication, which effectively prevents unauthorized access, data theft, and tampering.

Speed

With the aid of the SDK, console, or TencentCloud API, IoT Hub allows you to quickly enable device data communication without having to worry about the details of the underlying communication protocols (such as MQTT protocol fields).

Stability

With Tencent's years of experience in numerous services to leverage upon, IoT Hub has various features on the backend, such as automatic disaster recovery and load balancing, providing you with 24/7 OPS monitoring services.

Scalability

By connecting device data to Tencent Cloud services based on the rule engine, IoT Hub can implement the storage, real-time computation, and intelligent processing and analysis of massive amounts of data with speed and ease.

Low Cost

- IoT Hub is billed by the number of sent messages, ensuring low initial costs.
- IoT Hub's one-stop service architecture reduces labor and time costs for development.

Use Cases

Last updated：2021-08-19 17:51:00

Internet of Vehicles

Human-Vehicle interaction: IoT Hub can collect and analyze data about the driving habits of the driver, monitor the real-time conditions of vehicle parts such as tires, brake pads, and air conditioners as well as traffic conditions, and thereby provide traffic advice in a timely manner. Further, such data can be combined with the data in insurance company databases to select the most appropriate insurance policy.


Smart Home

Human-Home appliance interaction: IoT Hub monitors the real-time temperatures, timing, and whether users are home through door sensors and turns on/off appliances such as air conditioners, lights, TV, and stereos.

Industrial Equipment

IoT Hub enables enterprises to build IoT networks with speed and ease to achieve automatic control and real-time monitoring of equipment, helping improve corporate efficiency.
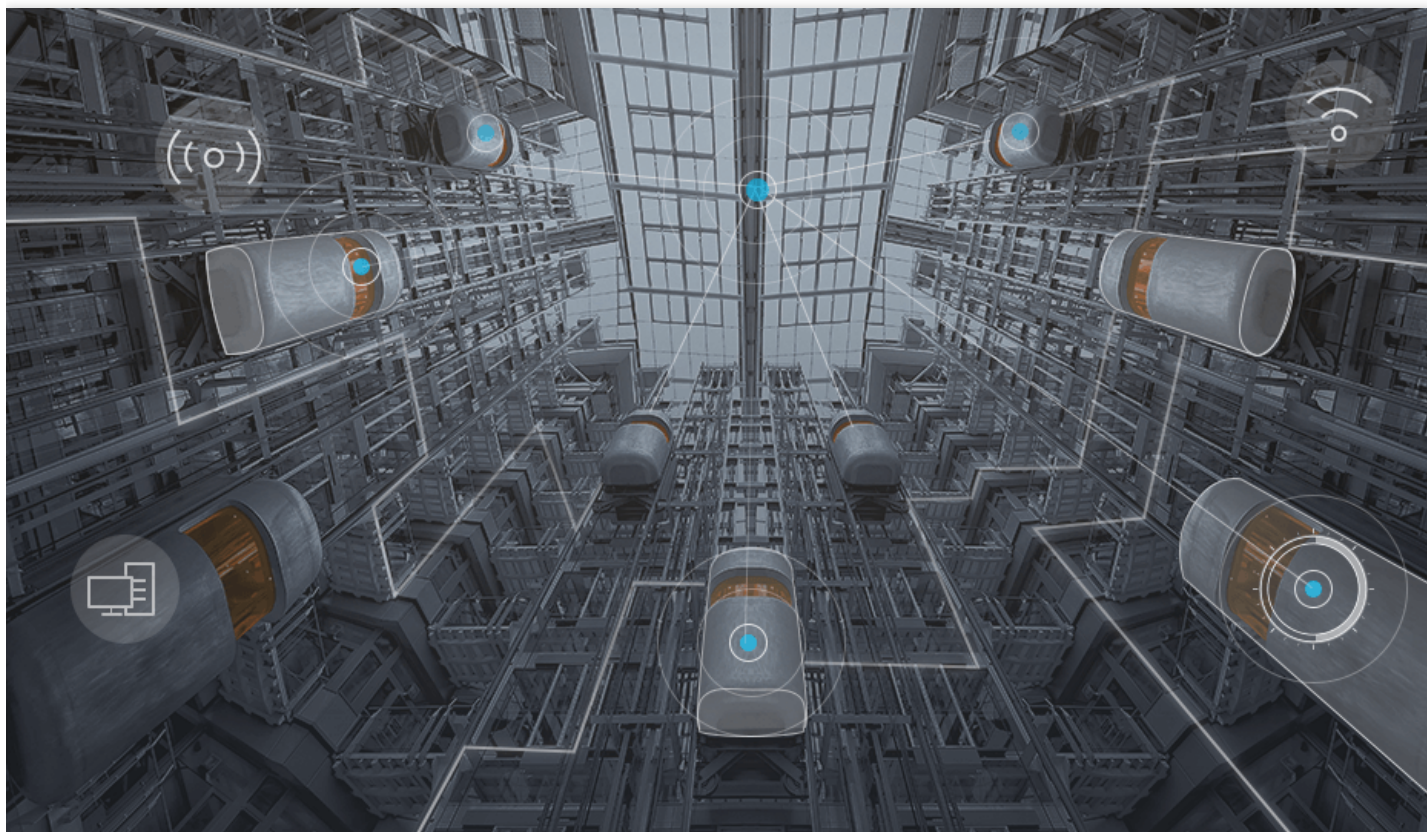
Smart Transportation/City

IoT Hub can connect a large number of sensing and controlling devices in cities and then use Tencent Cloud's big data services and intelligent algorithms to monitor and predict group behaviors, so as to improve the urban

management efficiency and reduce labor costs.

Use Limits

Last updated：2022-03-11 18:29:12

Device Connection

Item	Description	Limit
Products	Maximum number of products that can be created under a single account	2,000
Devices	Maximum number of devices that can be added under a single product	1,000,000
Gateways and subdevices	Maximum number of subdevices that can be added under a single gateway	1,500
Device groups	Maximum number of parent groups and subgroups in total allowed for a single account	Unlimited
	Maximum number of devices that can be added to a single group	Unlimited
	Maximum number of groups that a single device can be added to	Unlimited
Remote configuration	Maximum size of the remote configuration file (supports JSON format only)	128KB
Cloud log	Storage period of cloud logs generated by device operations	7 days
File management	Maximum size of all files that can be stored in the IoT Hub server for a single account	1GB
	Maximum number of files that can be stored for a single device	1,000
OTA update	Maximum size of a single update package file allowed	2,048 MB
	Maximum number of devices that can be updated in a single batch update	100,000

Message Communication

Item	Description	Limit
Device connection	With the same device certificate information, only one connection can be established with the platform server concurrently.	Yes
Connection requests	Maximum number of MQTT connection requests per second allowed for a single device	1 request/5s

Item	Description	Limit
Device subscriptions	Maximum number of subscriptions allowed for a single device	Unlimited
Requests	Number of requests sent from the device side to the IoT Hub platform per second allowed for a single account	Unlimited
	Number of requests sent from the IoT Hub platform to the device side per second allowed for a single account	Unlimited
Message communication	Maximum number of messages that a single device can report per second	30
	Maximum number of messages that a single device can receive per second (subject to the network environment)	50
Bandwidth	Maximum throughput (bandwidth) per second allowed for a single connection	Unlimited
Message storage period	Maximum storage time of QoS1 messages	7 days
MQTT message size	Maximum size allowed for a single published MQTT message. If this limit is exceeded, the publish request will be rejected.	16KB
CoAP message size	Maximum size allowed for a single published CoAP message. If this limit is exceeded, the publish request will be rejected.	1KB
MQTT keepalive	MQTT connection heartbeat time. If the heartbeat time is not within this limit, the server will reject the connection request.	900s
RRPC timeout period	Timeout period of device response to RRPC request	10s
Offline message	Number of offline messages	Up to 150 messages per device
	Storage period of offline messages	Messages can be stored for up to 24 hours.
`KeepAlive` value	Value range of `KeepAlive`	0-900s

Topic

Item	Description	Limit
Custom topic classes	Maximum number of custom topic classes allowed for a single product	100
Topic length	Maximum length of a topic allowed	255 UTF-8 encoded characters
Topic levels	Maximum number of hierarchical levels that can be contained in a topic, that is, maximum number of slashes in a topic	10
Subscriptions	Maximum number of subscriptions allowed per subscribe request	1
Operation effective time	Effective time of a subscription or unsubscription operation	5s
Broadcast topic	Restrictions on the body of the message to broadcast	Max 8 KB. The original message must be converted into binary data and encoded with Base64 to generate the message body.
	Message broadcasts by the server side to all devices per minute	A single product can perform only one task at a time.

Rule Engine

Item	Description	Limit
Rules	Maximum number of rules that can be set for a single account	1,000

Item	Description	Limit
Data forwarding destinations	Maximum number of data forwarding actions allowed in a rule	10
Messages processed by rule engine	The data processing capability of an account	Unlimited
Written message count	The data forwarding capability of an account when the performance of the target cloud product instance is sufficient	Unlimited
Requirements of data forwarding destinations	Data forwarding depends on the target product. Make sure that the instance of the target cloud product runs properly. Message forwarding fails in multiple scenarios. These scenarios include instance failure, overdue payments, parameter error, and configuration error.	The instance must run properly.
Message deduplication	When you forward a message, the message may be repeatedly sent until the client returns ACK or the message expires.	Unlimited. Each message has a unique ID.

Device Shadow and Server-Side Subscription

Item	Description	Limit
JSON levels	Maximum number of levels that can be specified in a device shadow JSON file	5
File size	Maximum size of a device shadow JSON file	16KB
Attributes	Maximum number of attributes that can be specified in a device shadow JSON file	Unlimited
Requests per second	Maximum number of requests per second per device	Unlimited
Retry policy upon push failure	Due to the consumer client being disconnected and the slow consumption of messages, messages cannot be consumed in real time and instead enter the stacked queue.	Unlimited

Basic Concepts

Last updated：2021-08-30 16:39:35

Product

A product is a collection of devices of a certain type, which usually have the same features. IoT Hub assigns a globally unique `ProductID` to each product. Through products, you can manage devices, topic permissions, and product-level data processing operations.

ProductID

`ProductID` is a unique product identifier assigned by the platform for easier product search. It is also used for identity authentication when a device connects to the platform. It can be viewed in the basic information of a product.

DeviceName

`DeviceName` is the name of a device under a product. It is used for identity authentication when the device connects to the platform.

Node Type

According to the types of devices actually connected to the IoT Hub platform, nodes can be divided into device type and gateway type.

Gateway Node

Devices of a gateway node can be directly connected to the platform. Devices under a device product can be added as subdevices, and after their topic permissions are added, the gateway node can publish and subscribe to data on behalf of the subdevices.

Device Node

Devices of a device node can be connected to the platform directly or through gateway devices. If the added devices cannot be connected to the platform directly, this node type can be selected, so that the devices will be connected to the platform through gateway devices.

Product Type

According to different application scenarios, the IoT Hub platform defines the categories of various hardware products in different application fields, which can be selected based on the communication methods of the devices actually connected to the platform. For NB and LoRa products, the platform performs targeted processing on the data transfer link.

Authentication Method

Device connection authentication supports certificate authentication (based on TLS asymmetric encryption and suitable for scenarios with high security requirements) and key authentication (based on symmetric encryption and suitable for resource-constrained devices). Authentication is performed at the device granularity to ensure the cloud-to-device and device-to-cloud message confidentiality. At the same time, the platform has designed a dynamic registration feature for scenarios where it is impossible to burn different firmware for each device. This feature supports getting a device key (or certificate + private key) through product-level key registration and then performing connection authentication, which enhances the connection flexibility.

Certificate Authentication

In certificate authentication mode, a device needs to carry the `ProductID` , `DeviceName` , certificate file, key file, and CA certificate to prove its validity before connection to the platform. After the device is connected, a certificate

file and a key file will be generated, which can be viewed in the device information.

CA Certificate

CA certificate is one of the identity authentication conditions for devices authenticated with certificate, which can be viewed in the basic information of the product.

Key Authentication

In key authentication mode, a device needs to carry the `ProductID`, `DeviceName`, and device key to prove its validity before connection to the platform. The key can be viewed in the device information.

Basic Information

Device Name	dev0
Remarks	None
Online Status	Inactive Reset
Tag	No tag information. Add
Enabled/Disabled ⓘ	🔵 Enabled
Firmware Version	Not reported

Log Configuration

Device Log	Disabled
Log Level	None

Device-Gateway Connection Information

Gateway Product Name	prod3
Gateway Device Name	dev1

Device Key ⓘ

Device Key	****************
Client ID	****************
MQTT Username	****************
MQTT Password	****************

ProductSecret

`ProductSecret` is a key at the product level. It is used to calculate the device-side signature when a device is dynamically registered, in exchange for the device-level key or certificate + private key.

Dynamic Registration

A device carries a unified `ProductId`, `ProductSecret`, and custom `DeviceName` to complete the first authentication with the platform. After the authentication is successful, the platform will issue a device-level key or certificate + private key, which, together with `ProductId+DeviceName`, will be carried by the device to complete the eventual authentication with the platform.

Topic

A topic is a UTF-8 string as the medium for message publishing/subscribing. You can publish messages to a topic or subscribe to messages in a topic.

Publishing

This is a type of permission (Pub) that manipulates a topic, i.e., publishing messages to the topic.

Subscribing

This is a type of permission (Sub) that manipulates a topic, i.e., subscribing to messages in the topic.