



普通商密

中国电信集团公司企业标准

2017 SX-029

智慧家庭智能家居业务开放接口 技术要求

Open Interface Specification of Smart Home Service

(V2017)

2018-××月发布

2018-××月实施

中国电信集团公司 发布

目录

前言.....	4
1 适用范围.....	5
2 引用标准.....	5
3 术语和缩略语.....	6
3.1 术语.....	6
3.2 缩略语.....	7
4 体系架构.....	8
4.1 体系架构.....	8
4.2 对接模式.....	9
5 设备与智能家居平台接口 (Mc)	9
5.1 接口介绍.....	9
5.2 通信方式.....	10
5.2.1 包类型表.....	10
5.2.2 安全策略.....	12
5.3 关键流程.....	13
5.3.1 设备连接建立流程.....	13
5.3.2 APP 与设备交互流程.....	14
5.4 接口定义.....	14
5.4.1 设备 ID 的分配说明.....	14
5.4.2 设备通道层接口.....	15
5.4.3 设备业务层接口.....	19
5.4.4 错误码列表.....	47
6 设备与 APP 接口 (Md)	47
6.1 接口介绍.....	47
6.2 接口定义.....	48
6.2.1 设备发现协议.....	48
6.2.2 设备认证协议.....	49
6.2.3 设备状态查询.....	49
6.2.4 设备状态控制.....	51
6.2.5 设备状态上报.....	53
6.2.6 错误码列表.....	54
7 与第三方业务平台对接接口 (C)	54
7.1 接口介绍.....	54
7.1.1 接口示意图.....	55
7.1.2 通信方式.....	55
7.1.3 参数类型说明.....	55
7.1.4 安全策略.....	56

7.2 关键流程	56
7.3 接口定义	57
7.3.1 智能家居平台提供的接口	57
7.3.2 第三方业务平台提供的接口	63

前言

本标准是中国电信智慧家庭系列标准之一，定义了中国电信智能家居业务的相关接口协议，包括设备接入智能家居平台接口（Mc）、APP本地接口（Md）、与第三方业务平台互通（C）接口三部分，用于指导第三方设备厂商将设备直接接入或通过第三方业务平台接入到电信智能家居平台，并通过电信APP完成设备的联网配置。

本标准依据中国电信智能家居业务需求，结合中国电信企业技术标准的特点进行编写，随着中国电信企业技术标准管理系统的发展完善，还将制定后续的相关规范。

本标准由中国电信集团公司提出并归口。

本标准起草单位：中国电信股份有限公司上海研究院、中国电信智能家居运营中心

本标准主要起草人：杨晓玲、王斌、郭涛、韩建亭、赵新宁、马安华

1 适用范围

本标准适用于中国电信智能家居平台与第三方业务平台、设备与智能家居平台、手机 APP 与设备之间对接测试及认证测试，是中国电信进行智能家居平台、终端、手机 APP 开发运营、对接测试、管理维护的技术依据。

本标准包含设备与智能家居平台之间、设备与 APP 之间、第三方智能家居平台与电信智能家居平台之间三方面的接口。

设备接入智能家居平台接口(Mc)定义了设备与智能家居平台间的数据通信方法和格式，规范了它们之间建立和维持通信通道所需要的通道层接口，包括设备与智能家居平台的连接、鉴权、心跳等接口；规范了它们之间进行业务数据交互的业务层接口，包括设备控制、设备状态上报、设备查询、设备告警、设备故障上报、子设备绑定解绑、设备升级等功能接口。

APP 本地接口 (Md) 定义了智能家居业务中 APP 和设备在同一个局域网环境下时，APP 不经过智能家居平台而直接和设备进行通信的方法及接口。该接口目前支持设备的网络零配置功能。

第三方平台互通接口 (C) 定义了第三方设备通过第三方业务平台与电信智能家居平台进行接入时，双方平台之间的接口定义及标准。

2 引用标准

下列文件中部分内容通过本文的引用而成为本标准的条款。凡是注日期的参考文件，其随后所有的修改单（不包括勘误的内容）或修订版均不适用于本文，凡是不注日期的参考文件，其最新版本适用于本文。

3 术语和缩略语

3.1 术语

智能家居平台

实现中国电信智能家居业务汇聚管理、对外合作对接、设备管理的开放平台。本标准中简称平台。

IP 应用终端

家庭终端的一种，上行通过 WiFi 连接到智能网关的智能设备，本标准中与应用网关一并简称为设备或 device。

应用网关

家庭终端的一种，实现 WiFi 协议与其它近场通信协议转换的网关设备，上行通过 WiFi（或有线）接入智能网关，下行通过 ZigBee，Z-wave、RF 等近场通信协议与子设备通信。

应用网关子设备

家庭终端的一种，与应用网关之间采用 ZigBee，Z-wave、RF 等协议，通过应用网关接入网络。本标准中简称子设备。

IP 组网终端

家庭终端的一种类型，实现家庭内有线无线拓展组网、增大家庭 WiFi 覆盖范围的终端，包括无线 AP、面板 AP、PLC 设备以及 EoC 等，本标准中简称组网终端。

智能家居客户端

可安装在手机、PAD 和其他移动终端的用户操控软件，是用户使用智能家居业务的统一入口。本标准中简称 APP 或客户端。

设备鉴权

对访问智能家居平台的设备进行合法性验证。

设备发现

在家庭局域网内通过客户端或智能网关进行设备搜索、发现的过程。

指令

家庭终端可以识别并执行的消息。

设备控制

向家庭终端发送指令，家庭终端接收并执行的过程。

设备状态

家庭终端向智能家居平台发送或反馈的自身状态数据或运行参数。

设备型号

设备厂商在智能家居平台上录入的智能家居应用终端的型号信息。

设备型号 ID

智能家居平台上生成的设备型号的唯一标识。

设备PIN码

为了安全起见，每一个设备 ID 有一个对应的安全密码，称为设备 PIN 码。设备 PIN 码由智能家居平台分配，并在设备生产时写入到设备中。设备 PIN 码为 32 个字节。

设备绑定密码

为了防止平台或客户端被破解，或用户恶意绑定设备，每一个设备有一个对应的绑定密码。设备绑定密码由智能家居平台分配，并在设备生产时以二维码形式粘贴在设备上。设备

绑定密码为8个字节。

3.2 缩略语

缩略语	英文全名	中文全名
HTTP	HyperText Transfer Protocol	超文本传输协议
HTTPS	Hyper Text Transfer Protocol over Secure Socket Layer	超文本安全传输协议
JSON	JavaScript Object Notation	JavaScript 对象表示法
XML	Extensible Markup Language	可扩展标记语言
TCP	Transmission Control Protocol	传输控制协议
UDP	User Datagram Protocol	用户数据报协议
WPS	Wi-Fi Protected Setup	WiFi 保护设置
WiFi	Wireless-Fidelity	无线保真
ZigWave	ZigWave	一种无线传输协议
ZigBee	ZigBee	一种无线传输协议
APP	Application	应用程序
TLS	Transport Layer Security	安全传输层协议
SSL	Secure Sockets Layer	安全套接字
PIN	Personal Identification Numbe	个人识别码

4 体系架构

4.1 体系架构

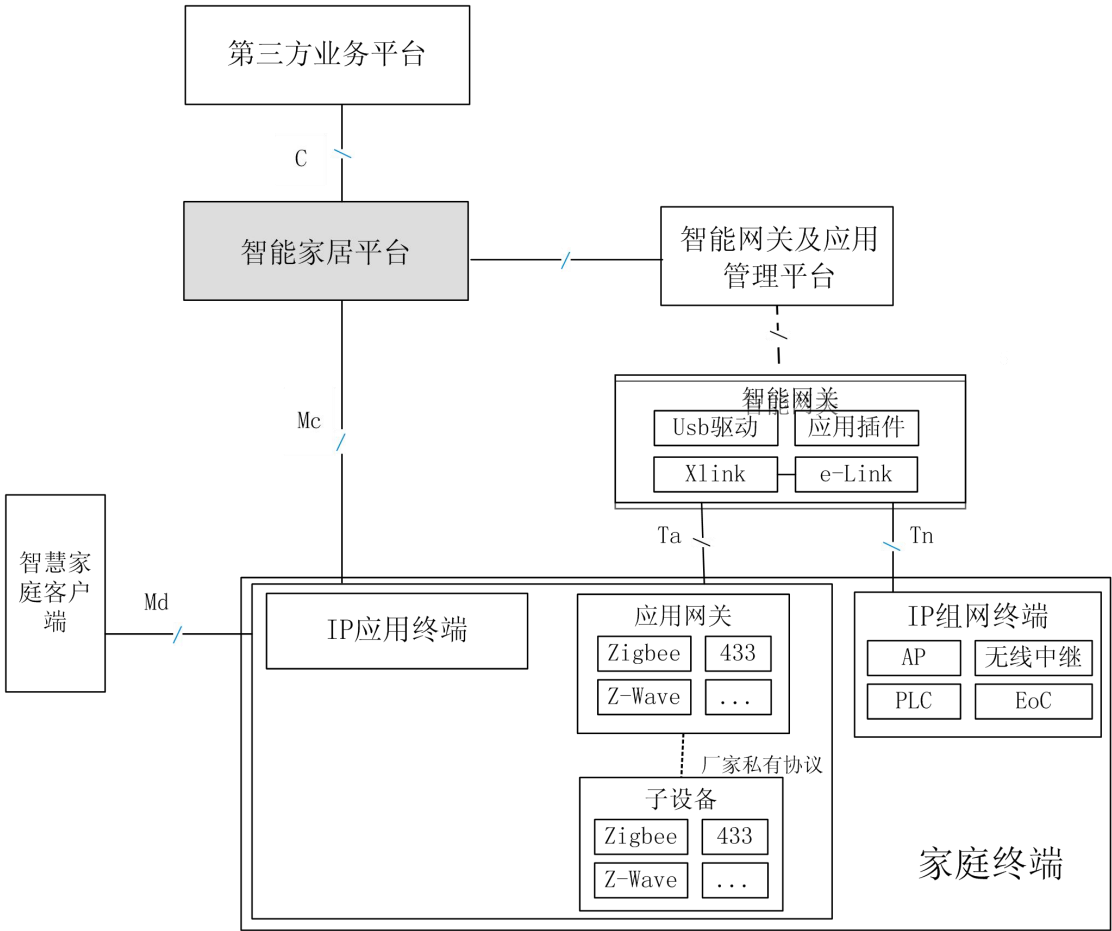


图 1 中国电信智能家居体系架构图

说明：

- **Ta 接口：**设备与智能网关之间的接口
Ta 接口是设备与智能网关之间的接口，Ta 接口定义了设备在家庭网络环境内的设备发现接口，用于在家庭环境内手机 APP 或智能网关快速发现设备，详见《智慧家庭家庭终端与智能家庭网关自动连接的接口技术要求》（QCT2621-2016）。
- **Tn 接口：**IP 组网终端与智能网关之间的接口
Tn 接口是组网终端与智能网关之间实现家庭智能组网的接口，通过 Tn 实现组网终端与智能网关间的密钥协商、设备注册、无线信息同步、配置下发、状态查询等功能，详见《智慧家庭家庭终端与智能家庭网关自动连接的接口技术要求》（QCT2621-2016）。
- **Mc 接口：**设备与智能家居平台之间的接口
Mc 接口是设备直连模式下，设备实现与智能家居平台之间的通信线路的建立、设备鉴权、设备控制、设备状态上报、设备告警、以及故障信息上报等功能。
- **Md 接口：**智能家居客户端与设备之间的接口

智能家居客户端通过此接口完成与同在一个局域网环境的设备的设备发现、认证、状态查询、控制和状态上报等功能。

- **C 接口：**第三方业务平台与智能家居平台之间的接口

该接口用于实现在平台互通对接模式下，第三方业务平台与智能家居平台之间的通信。

4.2 对接模式

智能家居平台的合作对接模式包括设备直连模式和平台互通模式。

设备直连模式：指合作伙伴的设备支持中国电信智能家居开放接口协议，实现设备与智能网关、智能家居平台、智能家居客户端之间的互通，满足家庭终端快速连接、平台接入、局域网控制等功能。

平台互通模式：指第三方业务平台与中国电信智能家居平台之间协议互通，实现通过智能家居客户端操控合作伙伴设备。平台之间采用智能家居开放接口协议互通。

不同业务和第三方业务平台，可根据实际情况采用不同对接模式实现，或两种模式相结合实现。

5 设备与智能家居平台接口（Mc）

5.1 接口介绍

设备接入智能家居平台接口(Mc)定义了设备与智能家居平台间的数据通信方法和格式，规范了它们之间建立和维持通信通道所需要的通道层接口，包括智能设备与智能家居平台的登录、连接、心跳等接口；还规范了它们之间进行业务数据交互的业务层接口，包括设备控制、状态上报、状态查询、设备告警、设备故障上报、子设备绑定解绑、设备升级等功能接口。

此接口适用于设备直连智能家居平台的对接模式。

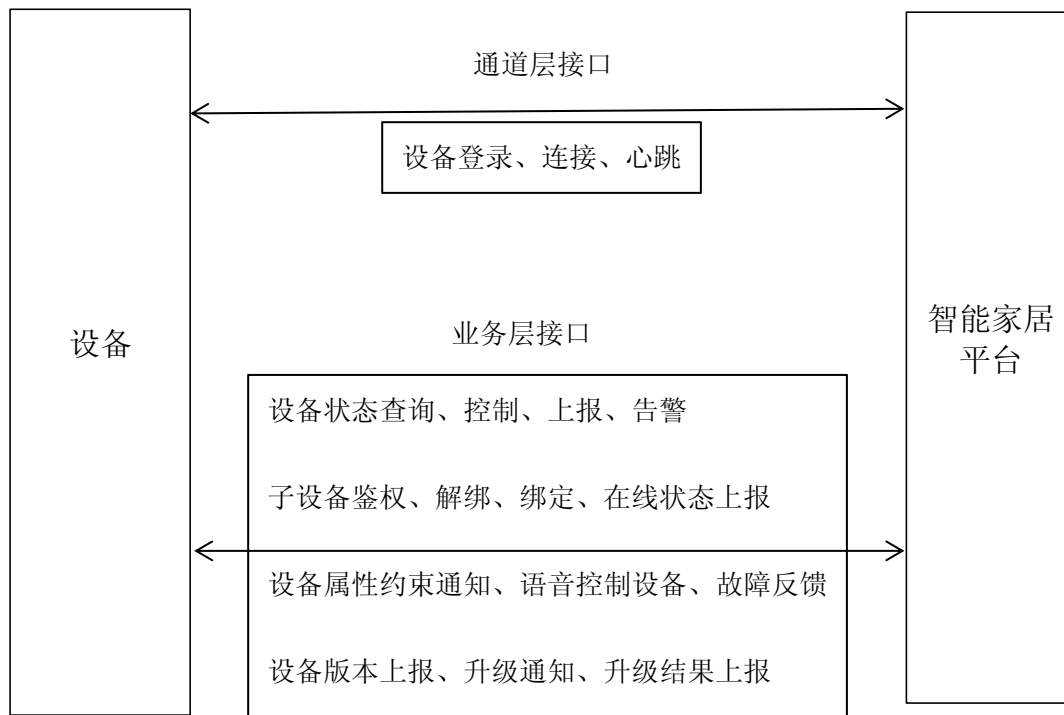


图 2 设备接入智能家居平台接口（Mc）接口列表

设备接入智能家居平台接口包括两种类型的接口：

- 1) 设备通道层接口：包括设备登录、连接、心跳等接口；
- 2) 设备业务层接口：包括设备状态查询、状态控制、状态上报、状态告警，子设备鉴权、解绑、绑定、在线状态上报，设备属性约束通知，语音控制设备，设备故障反馈，设备版本上报、升级通知、升级结果上报等接口。

5.2 通信方式

智能设备与智能家居平台的通信接口又可以分为两个层次：通信通道的建立与维护、业务层的数据交换。

通信通道层的建立与维护协议主要包括设备接入、设备心跳、设备连接等。其中设备接入又包含登录和连接两个过程。

业务层的数据交换协议包括设备状态查询、状态控制、状态上报、状态告警，子设备鉴权、解绑、绑定、在线状态上报，设备属性约束通知，语音控制设备，设备故障反馈，设备版本上报、升级通知、升级结果上报等。

设备与智能家居平台的通信采用 TCP 或 UDP 方式，数据包统一采用 JSON 形式。

数据包以 CTS 开头，以\r\n结尾，内容为加密并经 Base64 编码的消息体。

设备或平台发送消息后，消息接收方应在 10s 内作出响应，如果消息发送方在 10s 内未收到返回消息，则判定消息发送失败，需要重新发送消息，连续 3 次发送信息失败，则设备需要重新进行鉴权流程。

5.2.1 包类型表

本协议支持的数据包类型列表如下：

包类型值	包类型说明
0	保留
1000	心跳包
1001	心跳响应包
1002	设备登录包
1003	设备登录响应包
1004	设备连接包
1005	设备连接响应包
2000	设备属性约束通知响应包
2001	设备属性约束通知请求包
2002	设备状态查询响应包
2003	设备状态查询请求包
2004	设备状态控制响应包
2005	设备状态控制请求包
2006	设备状态上报请求包
2007	设备状态上报响应包
2008	设备状态告警请求包
2009	设备状态告警响应包
2010	设备故障反馈请求包
2011	设备故障反馈响应包
2012	子设备鉴权请求包
2013	子设备鉴权响应包
2014	子设备解绑响应包
2015	子设备解绑请求包
2016	子设备绑定上报请求包

2017	子设备绑定上报响应包
2018	子设备解绑上报请求包
2019	子设备解绑上报响应包
2020	子设备在线状态上报请求包
2021	子设备在线状态上报响应包
2022	设备版本上报请求包
2023	子设备版本上报响应包
2024	设备版本升级通知响应包
2025	设备版本升级通知请求包
2026	设备版本升级结果上报请求包
2027	设备版本升级结果上报响应包
2028	设备资源变化上报请求包
2029	设备资源变化上报响应包
2030	设备资源管理响应包
2031	设备资源管理请求包
2032	设备事件信息上报请求包
2033	设备事件信息上报响应包
3000	语音合成请求包
3001	语音合成响应包
3002	语音控制请求包
3003	语音控制响应包

5.2.2 安全策略

安全策略遵循以下几个原则：

- 1) 数据包中关键数据部分加密，非关键数据部分不加密。
- 2) 关键数据全部采用 AES128CBC/PKCS5 对数据进行加密，设备登录使用设备 PIN 码（32 位）高 16 位作为加密密钥，使用设备 PIN 码低 16 位作为初始化向量，设备连接当中使用平台返回的会话密钥（16 位）作为加密密钥及初始化向量。
- 3) 非关键数据，采用明文传输，提高传输效率。
- 4) 局域网交互过程也全部进行加密处理。
- 5) 接口调用时包含序列号，被调用方对序列号进行校验，序列号校验失败时返回错误信息。

平台与设备使用独立的序列号发送消息，平台与设备都需维护两套序列号，设备登录及连接时该序列号随机产生，接口调用时序列号每次都在之前调用的基础上加 1，达到最大值 65535 后从 0 开始。平台检测设备序列号错乱后返回错误信息，设备重新进行登录流程，设备检测平台序列号错乱后，重新进行登录流程。

- 6) 设备与平台通信过程中，双方都对数据包加密进行校验，数据包加密校验失败的，不作任何处理，不返回信息。

5.3 关键流程

5.3.1 设备连接建立流程

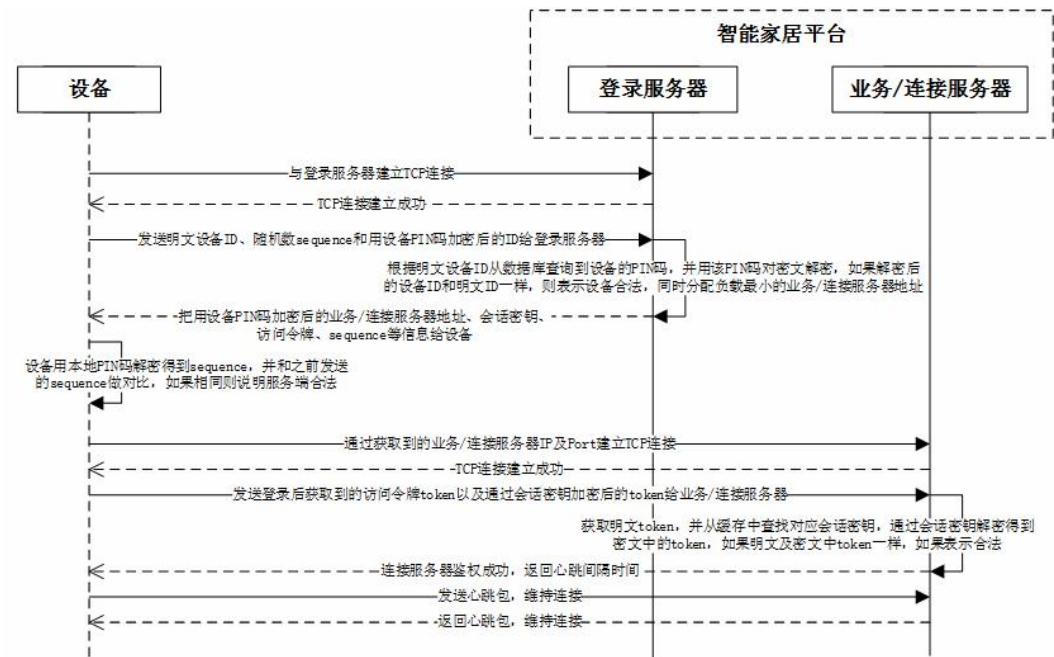


图 3 设备连接建立流程

【流程说明】

- 1) 设备与平台的登录服务器建立 TCP 连接；
- 2) 设备发送明文的设备 ID、随机数 Sequence 和用 PIN 码加密的 ID 给平台；
- 3) 平台根据设备 ID 从数据库中找到对应的 PIN 码，并用该 PIN 码对密文解密，如果解密后的设备 ID 和明文 ID 一样，则表示设备合法，同时分配一个负载最小的连接服务器地址。平台把用设备 PIN 码加密后的业务/连接服务器 IP 及端口、会话密钥 SessionKey、设备访问令牌 Token、Sequence 等信息发送给设备；若校验设备不合法，则断开当前连接，不返回任何信息；
- 4) 设备用本地 PIN 码解密出密文中的 Sequence，并和之前发送的 Sequence 进行比较，如果相同则表示平台是合法的；若不合法则断开当前连接，按登录失败处理；
- 5) 设备通过获取的 IP 和端口号，同平台的业务/连接服务器建立 TCP 连接，发送登录成功返回的 Token 及通过会话密钥 SessionKey 加密后的 Token 给平台的连接服务器，平台的连接服务器获取 Token，判别是否合法，若合法，则向设备发送连接服务器鉴权成功消息；若不合法则断开当前连接，不返回任何信息。

- 6) 设备向平台的连接服务器发送心跳包，维持连接。
- 7) 平台的连接服务器向设备发送心跳包响应，维持连接。

5.3.2 APP与设备交互流程

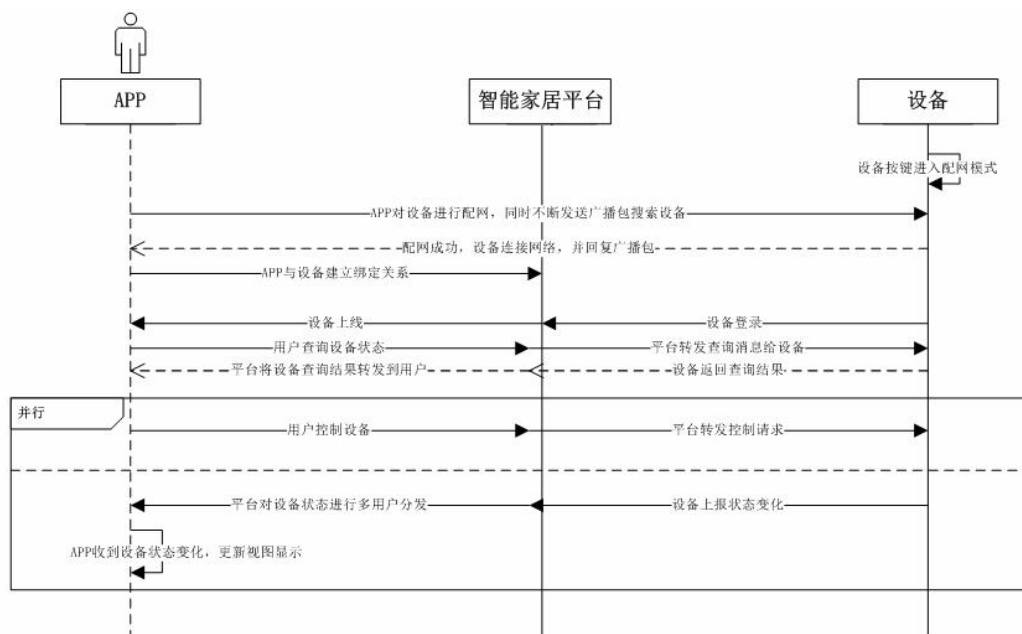


图 4 APP 与设备交互流程

【流程说明】

- 1) 设备完成配网，APP 与设备建立绑定关系，设备登录平台，平台通知 APP 设备上线；
- 2) 用户通过 APP 发送查询设备状态消息到平台；
- 3) 平台转发该查询消息给设备；
- 4) 设备返回查询结果给平台；
- 5) 平台将设备查询结果转发给 APP，APP 收到查询结果，通过界面展示；
- 6) 用户通过 APP 发送控制设备的指令信息到平台；
- 7) 平台将此控制请求转发给设备；
- 8) 设备执行控制指令，并上报状态变化到平台；
- 9) 平台对设备状态信息进行多用户 APP 分发；
- 10) APP 收到设备状态变化，更新视图显示。

5.4 接口定义

5.4.1 设备ID的分配说明

接入到智能家居平台的每一个设备必须具有设备 ID、设备 PIN 码、设备绑定密码。设备 ID 是唯一的，用于区分不同的设备；设备 PIN 码是设备 ID 对应的安全密码；设备绑定密码用于设备绑定时，校验用户对设备的拥有情况。

设备 ID 的分配流程如下：

由设备生产厂商向智能家居平台提出申请，智能家居平台首先对厂家进行审核，审核通过后智能家居平台可以与厂家签订安全协议并分配一定数量的设备 ID、PIN 码、绑定密码提供给厂商使用。也可由设备厂商自行分配后，提供给智能家居平台进行入库备案。

5.4.2 设备通道层接口

通道层连接协议主要完成了设备登录、连接的建立、维持和断开等功能。

1) 连接的建立

设备与智能家居平台的连接采用 TCP 方式，设备上电或者复位后与平台建立 TCP 连接，连接建立成功后立即向智能家居平台发送鉴权消息进行登录鉴权。鉴权成功后连接智能家居平台即完成连接的建立。

2) 连接的维持

在设备连接到连接智能家居平台之后，设备需要周期性的向平台发送心跳消息，智能家居平台收到心跳后向设备发送心跳应答消息，具体心跳时间间隔有智能家居平台统一配置，在设备连接成功后会返回给设备，平台未返回心跳配置时，设备使用默认心跳周期。

3) 连接的断开

智能家居平台和设备均可根据 TCP 协议主动断开连接，双方都应主动判断 TCP 连接是否断开。

智能家居平台判断 TCP 连接断开的方法：

- (1) 根据 TCP 协议判断出设备主动断开；
- (2) 同一设备建立新连接，表明原连接已断开；
- (3) 在 3 个心跳周期范围内未收到设备发出的消息或心跳。

设备判断 TCP 连接断开的方法：

- (1) 根据 TCP 协议判断出平台主动断开；
- (2) 数据通信链路断开；
- (3) 数据通信链路正常，达到重传次数后仍未收到应答。

5.4.2.1 设备登录

接口描述：

登录主要目的是为了鉴权设备信息的合法性，需要提供设备 ID 与设备 PIN 码。鉴权成功后智能家居平台会根据负载均衡算法分配一个业务连接服务器的 IP 地址和端口号，以便设备继续连接业务服务器。

登录消息采用设备 PIN 码作为密钥对数据部进行加密，设备 ID 采用明文传输。同时会将设备 ID 也放在密文中，智能家居平台收到该消息后，会用智能家居平台保存的设备 PIN 码对消息解密，解密后比对明文中的设备 ID 和密文中的设备 ID，如果两个 ID 相同，则认为设备是合法的。

智能家居平台收到登录包对设备鉴权通过后，进行回复，通过回复包返回连接服务器 IP、端口以及访问令牌等信息。同时智能家居平台会将设备上传的随机数重新使用设备 PIN 码加密后返回给设备，设备收到消息后如果解密得到的随机数 sequence 和请求数据包中的随机数 sequence 相同，则认为智能家居平台是合法的，否则智能家居平台是非法的。

登录失败时，设备间隔 1 分钟再次发起登录请求，连续 3 次登录失败后，设备以 10 分钟/服务器返回的鉴权时间间隔发起登录请求。

设备登录鉴权是设备接入智能家居平台的关键，为了保证数据传输的可靠性，需使用 TCP 协议进行通信。

设备可利用鉴权服务器返回信息中的时间戳进行本地时间校准，当设备本地时间与服务器时间偏差较大时，会造成无法通信。

消息方向：device->Server->device

请求数据包	
需加密的请求内容	{

	<pre> "sequence": "12345", "deviceId": "0000111122223333aaaabbbb", "devMac": "AA:BB:CC:DD:EE:FF", "devCTEI": "CT123456789012345", "version": "XXX.YYY.ZZZ.MMM" "time": 1361542433 } </pre>
使用PIN码作为密钥进行加密	data = Base64 (AES (鉴权内容 , 设备 PIN 码))
完整的请求数据包	<pre> { "code": 1002, "deviceId": "0000111122223333aaaabbbb", "data": "XXXXXXXXXXXX", } </pre>
响应数据包	
需加密的响应内容	<pre> { "result": 0, "sequence": "12345", "sessionKey": "abcdef0123456789abcdef0", "tcpHost": "XXX.XXX.XXX.XXX:YYYY", "udpHost": "XXX.XXX.XXX.XXX:ZZZZ", "token": "qwertyuiop123456", "time": 1361542433 } </pre>
使用PIN码作为密钥进行加密	data = Base64 (AES (鉴权结果内容 , 设备 PIN 码))
完整的响应数据包	<pre> { "code": 1003, "data": "XXXXXXXXXXXX" } </pre>

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码, 根据不同消息代码不同
deviceId	是	设备 ID, 设备在电信智能家居平台的唯一标识, 包含设备厂商、品牌、产品类别、型号、版本、批次、序列号等信息
devMac	否	设备 MAC 地址, 多个 MAC 时使用第一个网卡 MAC 地址
devCTEI	否	设备 CTEI 码/IMEI (针对移动通讯设备)
version	是	协议版本, 版本号格式为 XXX.YYY.ZZZ.MMM, 代表含义如下: XXX: 主版本号 YYY: 次版本号 ZZZ: 小版本号 MMM: 修订版本号
sequence	是	接口调用序列号, 取值 0~ 65535, 回复方使用同样的序列号返回响应信息。

time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
sessionKey	是	设备鉴权成功后，平台下发的会话密钥，用于设备控制、状态上报、心跳等消息加密
tcpHost	是	TCP 协议通信的连接服务器主机信息，包括 IP（域名）或端口
udpHost	是	UDP 协议通信的连接服务器主机信息，包括 IP（域名）或端口
token	是	设备访问令牌，智能家居平台端会保存本次登录的 deviceId 和 token 的对应关系，设备拿到 token 后去连接到长连接服务器时只要提供该 token，长连接服务器即认为该连接是合法的
data	是	数据加密后的内容

5.4.2.2 设备连接

接口描述：

登录成功后设备获取连接服务器（即业务服务器）的主机信息（含 IP 地址与端口），设备会向此连接主机起心跳保持网络连接，这次连接是智能家居平台提供业务服务的连接，需要保持。

设备登录成功后，优先连接 UDP 服务器，每次上电使用过程中当设备累计进行 3 次鉴权操作后，则后续使用过程中切换到 TCP 连接。

设备连接连接服务器后，首先需要发送一个连接确认消息，以便智能家居平台确认本次连接是否有效。

智能家居平台收到连接包后，进行回复，通过回复包，通知设备心跳间隔、智能家居平台的最新时间等信息。

设备得到智能家居平台返回消息后，应立即上报一次当前设备状态及资源信息。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "token": "qwertyuiop123456", "devVersion": "XXX.YYY.ZZZ.MMM", "model": "XXXX", "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre>{ "code": 1004, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }</pre>

响应数据包	
需加密的响应内容	<pre>{ "result":0, "sequence":"12345", "heartBeat":45, "authInterval":600, "time":1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre>{ "code":1005, "data":"XXXXXXXXXX" }</pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
devVersion	是	设备固件版本，版本号格式为 XXX.YYY.ZZZ.MMM，代表含义如下： XXX：主版本号 YYY：次版本号 ZZZ：小版本号 MMM：修订版本号
model	是	设备型号，内含厂家标识、品牌标识、型号标识等
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
heartBeat	否	设备心跳时间间隔，单位秒，默认 30 秒
authInterval	否	鉴权重试时间间隔，默认前 3 次重试间隔为 60 秒钟，后续默认 600 秒钟。 平台返回 authInterval，就以 authInterval 为准
time	是	平台/设备当前时间，linux 标准格式时间戳
data	是	数据加密后的内容

5.4.2.3 设备心跳

接口描述：

心跳由设备发起，设备每隔 X（X 值具体由前面连接确认命令的响应中获取）秒会发向智能家居平台发送一条心跳命令，智能家居平台会对设备的心跳进行响应。当智能家居平台在一个心跳保持周期（3 个心跳间隔） + 10s 内没有收到设备的心跳数据则认为设备下线，设备需要重新进行鉴权。智能家居平台收到心跳包后，立即进行回复。

消息方向：device->server->device

请求数据包	
需加密的请求内容	{ "sequence": "12345", "token": "qwertyuiop123456", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	{ "code": 1000, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }
响应数据包	
需加密的响应内容	{ "result": 0, "sequence": "12345", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	{ "code": 1001, "data": "XXXXXXXXXXXX" }

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3 设备业务层接口

业务层协议主要指与设备本身的操控逻辑相关的协议，包括设备状态查询、设备控制、设备状态上报、子设备解绑与绑定、语音控制设备、版本升级等。

业务消息指令编码、错误代码、时间戳使用数字类型，设备控制及状态上报参数类型统一使用 string 类型。

5.4.3.1 设备状态查询

接口描述：

智能家居平台通过本接口查询设备及绑定到设备的子设备的当前状态,设备会把当前状态集合按照规定的格式返回。

设备在接收到智能家居平台的设备查询请求后,先通过参数中的 **deviceId** 判断是不是要查询自己;如果不是则查询设备绑定的子设备中是否存在设备 ID 与 **deviceId** 相同的,若存在则将此命令进行转换来查询对应子设备。

设备在返回接受指令成功后,应当在 **10s** 内上报设备的最新状态。

消息方向: server->device->server

请求数据包	
需加密的请求内容	{ "sequence": "12345", "deviceId": "XXX", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	{ "code": 2003, "data": "XXXXXXXXXXXX" }
响应数据包	
需加密的响应内容	{ "result": 0, "sequence": "12345", "deviceId": "XXX", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	{ "code": 2002, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码,根据不同消息代码不同
deviceId	是	设备 ID, 为应用网关或子设备的 ID
sequence	是	接口调用序列号,取值 0~65535, 回复方使用同样的序列号返回响应信息。
time	是	平台/设备当前时间, linux 标准格式时间戳。 请求包中的 time 表示请求发起时间, 响应包中的 time 表示请求响应时间。
result	是	处理结果码, 为 0 表示成功, 非 0 表示失败, 错误码信息表见附录 1
token	是	设备访问平台的令牌, 通过设备登录接口获取
data	是	数据加密后的内容

5.4.3.2 设备状态控制

接口描述:

智能家居平台通过此接口将 APP 发出的控制设备状态的命令发送给设备，一次命令可以控制设备的多个属性同时发生变化,当指令存在先后顺序时,平台按先后顺序下发给设备，设备按顺序执行。设备绑定的子设备控制也通过此接口进行。

设备在接收到智能家居平台的设备控制请求后，先通过参数中的 deviceId 判断是不是要控制自己；如果不是则查询设备绑定的子设备中是否存在设备 ID 与 deviceId 相同的，若存在则将此控制命令进行转换来控制对应子设备。

设备在返回接受指令成功后，应当在 10s 内上报设备执行控制指令后的最新状态。

消息方向：server->device->server

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "serialId": "0", "cmd": [{ "cmdName": "SET_POWER", "cmdParam": "1" }, { "cmdName": "SET_MODE", "cmdParam": "2" }, { "cmdName": "SET_TEMP", "cmdParam": "26" }], "time": 1361542433 }</pre>
使用会话密钥进行加密	<code>data = Base64 (AES (连接请求内容 , sessionKey))</code>
完整的请求数据包	<pre>{ "code": 2005, "data": "XXXXXXXXXXXX" }</pre>
响应数据包	
需加密的响应内容	<pre>{ "result": 0, "sequence": "12345", "dscp": "操作成功", "deviceId": "XXX", "time": 1361542433 }</pre>

	}
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	{ "code":2004, "token":"qwertyuiop123456", "data":"XXXXXXXXXXXX" }

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
serialId	是	控制多路设备的线路序号(1,2,3...)，单路设备为 0，多路设备为 0 时代表全部控制
cmd	是	设备控制命令键值对集合，键值对中的键表示命令名称，键值对中的值表示命令参数
cmdName	是	命令名称
cmdParam	是	命令参数
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
dscp	是	设备操作结果描述，可为空，建议填写
token	是	设备访问平台的令牌，通过设备登录接口获取
data	是	数据加密后的内容

5.4.3.3 设备状态/资源上报

接口描述：

当设备及子设备的状态发生变化时，设备通过本接口将变化的信息上报给智能家居平台。
当平台标记为离线的子设备上报状态变化时，设备在线状态自动设置为在线。

对存在资源的设备（如加湿器的水、空净的滤网、门锁的钥匙列表），可在状态上报时同时上报资源信息，也可分别上报状态和资源信息。

设备每次连网或重新鉴权后，必须上报一次完整的状态及资源信息。

设备状态及资源信息可一次上报，也可分多次上报。当分多次上报资源信息时，两条消息的间隔不得超过 4 个心跳周期（默认 120s），否则视为全部资源信息已经上报完毕。

资源上报过程中，发生资源变化时，应在资源上报完成后，再上报资源变化信息。

消息方向：device->server->device

请求数据包	
需加密的请求内容	{ "sequence":"12345", "deviceId":"XXX", "statusSerials":[] }

	<pre> { "serialId": "1", "statusSerial": [{ "statusName": "POWER", "curStatusValue": "1" }], }, { "serialId": "2", "statusSerial": [{ "statusName": "POWER", "curStatusValue": "0" }] }], "resourceSerials": [{ "serialId": "1", "resourceSerial": [{ "resourceName": "CONSUM", "resourceInfo": [{ "CONSUM_ID": "348749488", "CONSUM_NAME": "MAIN_FILTER", // 名字可为空 "REMAIN_LIFE": "87" }] }] }], "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre> { "code": 2006, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" } </pre>

	}
响应数据包	
需加密的响应内容	{ "result":0, "sequence":"12345", "deviceId":"XXX", "time":1361542433 }
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	{ "code":2007, "data":"XXXXXXXXXXXX" }

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
statusSerials	是	设备状态集合，不少智能设备多路，每一路相当于是一个设备，都有着相同的状态信息。比如多路插座、多路开关等 设备状态的具体定义说明见上文“设备状态查询”接口中的定义
serialId	是	多路设备的路数，0 表示为单路设备，1、2、3 表示多路设备的第 1、2、3 路
statusSerial	是	设备某一路的状态集合
statusName	是	状态名称
curStatusValue	是	当前状态值
resourceSerials	否	设备所有资源列表（设备通过这个参数上报资源信息，如各种耗材状态等）
resourceSerial	否	某一路资源信息列表，多种资源依次列出
resourceName	否	资源种类名称
resourceInfo	否	详细资源信息列表（不同资源种类资源信息格式不一样，详见设备资源定义表）
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.4 设备资源管理

接口描述：

当设备资源允许远程进行管理时，通过该接口进行管理。

设备成功接收资源管理请求后，应当在 **10s** 内上报资源变化信息。

消息方向：server->device->server

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "resourceSerials": [{ "serialId": "0", "resourceSerial": [{ "resourceName": "KEY_LIST", // 门锁钥匙 "resourceInfo": [{ "KEY_ID": "1", "KEY_TYPE": "1", "MODE": "1" // 1: 新增, 2: 修改, 3: 删除 }, { "KEY_ID": "2", "KEY_TYPE": "2", "MODE": "3" // 1: 新增, 2: 修改, 3: 删除 }] }] }], "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre>{ "code": 2031, "data": "XXXXXXXXXX" }</pre>
响应数据包	
需加密的响应内容	<pre>{ "result": 0, "sequence": "12345", }</pre>

	<pre>"deviceId":"XXX", "time":1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre>{ "code":2030, "token":"qwertyuiop123456", "data":"XXXXXXXXXXXX" }</pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
serialId	是	多路设备的路数，0 表示为单路设备，1、2、3 表示多路设备的第 1、2、3 路
resourceSerials	否	设备所有资源列表（设备通过这个参数上报资源信息，如各种耗材状态等）
resourceSerial	否	某一路资源信息列表，多种资源依次列出
resourceName	否	资源种类名称
resourceInfo	否	详细资源信息列表（不同资源种类资源信息格式不一样，详见设备资源定义表）
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.5 设备资源变化上报

接口描述：

当设备及子设备的资源发生变化时，设备通过本接口将变化的信息上报给智能家居平台。
当平台标记为离线的子设备上资源变化时，设备在线状态自动设置为在线。

设备资源变化信息可以一次上报，也可分多次上报。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence":"12345", "deviceId":"XXX", "resourceSerials":[{</pre>

信息示例	<pre> "serialId": "1", "resourceSerial": [{ "resourceName": "KEY_LIST", // 门锁钥匙 "resourceInfo": [{ "KEY_ID": "1", "KEY_TYPE": "1", "MODE": "1" // 1: 新增, 2: 修改, 3: 删除 }, { "KEY_ID": "2", "KEY_TYPE": "2", "MODE": "3" // 1: 新增, 2: 修改, 3: 删除 }] }]], "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre> { "code": 2028, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" } </pre>
响应数据包	
需加密的响应内容	<pre> { "result": 0, "sequence": "12345", "deviceId": "XXX", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code": 2029, "data": "XXXXXXXXXXXX" } </pre>

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
serialId	是	多路设备的路数，0 表示为单路设备，1、2、3 表示多路设备的第 1、2、3 路
resourceSerials	否	设备所有资源列表（设备通过这个参数上报资源信息，如各种耗材状态等）
resourceSerial	否	某一路资源信息列表，多种资源依次列出
resourceName	否	资源种类名称
resourceInfo	否	详细资源信息列表（不同资源种类资源信息格式不一样，详见设备资源定义表）
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.6 设备信息上报

接口描述：

当设备运行过程中，产生的信息/事件（如闹铃响起、开门信息），可通过该接口上报。
当平台标记为离线的子设备上报状态变化时，设备在线状态自动设置为在线。

设备信息可以一次上报一条，也可一次上报多条。

当设备信息记录为文件时，通过该格式将文件记录逐条或多条上报。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "event": [{ "eventId": "1", //可选，信息序号，单个消息上报可为空，一次上报多条信息建议填上，以便对多种事件进行区分 "serialId": "1", "eventName": "LOCK_OPEN", //开锁信息示例 "eventInfo": [{ "KEY_ID": "1", "KEY_TYPE": "1", </pre>

	<pre> "TIME": "1361542433" }] }], "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre> { "code": 2032, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" } </pre>
响应数据包	
需加密的响应内容	<pre> { "result": 0, "sequence": "12345", "deviceId": "XXX", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code": 2033, "data": "XXXXXXXXXXXX" } </pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535 ，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
serialId	是	多路设备的路数， 0 表示为单路设备， 1、2、3 表示多路设备的第 1、2、3 路
event	否	设备上报信息列表（设备通过这个参数上报信息，如开锁事件、闹钟响等）
eventId	否	信息 ID，可选，一次上报多个信息时，用于多个信息的区分
eventName	否	信息类型名称
eventInfo	否	详细信息内容（不同信息类型内容格式不一样，详见设备信息定义表）
time	是	平台/设备当前时间， linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。

result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.7 设备状态告警

接口描述：

智能家居平台对设备属性进行阈值配置，设备检测到设备属性超过设定的阈值时，会发送告警数据到智能家居平台。普通设备不需要上报告警信息，但一些复杂设备的一些告警数据需要及时上报。设备及其子设备的状态告警都通过此接口进行上报。

消息方向：device->server->device

请求数据包		
需加密的请求内容		<pre>{ "sequence": "12345", "deviceId": "XXX", "alarm": { "serialId": "1", "statusName": "TEMP", "curStatusValue": "40", "dscp": "空调温度过高", "alarmTime": 1361542433 }, "time": 1361542433 }</pre>
使用会话密钥进行加密		data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包		<pre>{ "code": 2008, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }</pre>
响应数据包		
需加密的响应内容		<pre>{ "result": 0, "sequence": "12345", "deviceId": "XXX", "time": 1361542433 }</pre>
使用会话密钥进行加密		data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包		<pre>{ "code": 2009, "data": "XXXXXXXXXXXX" }</pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同

token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
alarm	是	设备告警信息，statusName 为设备属性名称、curStatusValue 为设备属性值，dscp 为设备告警信息描述
alarmTime	是	设备告警上报时间
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.8 设备故障反馈

接口描述：

当设备或其子设备发生故障时，设备会通过本接口消息发送故障代码列表给智能家居平台。智能家居平台收到后会评估设备是否需要升级或通知 APP 设备故障的原因。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "error": { "serialId": "1", "errorCode": "100001", "errorInfo": "XXXXX", "errorTime": 1361542433 }, "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre>{ "code": 2010, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }</pre>
响应数据包	
需加密的响应内容	<pre>{ "result": 0, "sequence": "12345", "deviceId": "XXX", "time": 1361542433 }</pre>

使用会话密钥进行加密	<code>data = Base64 (AES (响应内容 , sessionKey))</code>
完整的响应数据包	<pre>{ "code":2011, "data":"XXXXXXXXXX" }</pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
errorCode	是	故障错误码
errorInfo	是	故障详细信息
errorTime	是	故障发生时间
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.9 子设备鉴权

接口描述：

子设备连接到上连设备后，需要进行鉴权，由应用网关向智能家居平台发起鉴权请求，对子设备进行鉴权。当应用网关网络连接断开后再次连网时，应用网关需要进行登录，其所有子设备需要通过该接口进行鉴权认证。

应用网关在收到平台返回的鉴权结果后，应根据鉴权结果，上报一次鉴权通过的子设备的在线状态，并继续上报子设备的当前状态信息。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "pin": "kkkk", "time": 1361542433 }</pre>
使用密钥进行加密	<code>data = Base64 (AES (连接请求内容 , sessionKey))</code>
完整的请求数据包	<pre>{ "code":2012, "token":"qwertyuiop123456", "data":"XXXXXXXXXX" }</pre>

响应数据包	
需加密的响应内容	<pre>{ "result": 0, "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "authResult": "0", "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre>{ "code": 2013, "data": "XXXXXXXXXXXX" }</pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID
subDevices	是	子设备列表
subDeviceId	是	子设备 ID
pin	是	鉴权数据，使用子设备的 pin 码
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
authResult	是	子设备认证结果，0：认证通过，1：认证不通过
data	是	数据加密后的内容

5.4.3.10 子设备解绑

接口描述：

在 APP 上用户删除子设备，即为解绑子设备，此时在智能家居平台会删除子设备和应用网关的关系。同时，智能家居平台也会通知应用网关对子设备进行解绑，将对应的子设备断开连接，进行下线处理。

消息方向：server-> device -> server

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "time": 1361542433 }</pre>

	}
使用密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	{ "code":2015, "data":"XXXXXXXXXX" }
响应数据包	
需加密的响应内容	{ "result": 0, "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	{ "code":2014, "token":"qwertyuiop123456", "data":"XXXXXXXXXX" }

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID
subDeviceId	是	子设备 ID
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.11 子设备绑定上报（暂不用）

接口描述：

子设备绑定到应用网关后，由应用网关向智能家居平台发起子设备绑定上报数据，智能家居平台保存应用网关和子设备的关系。

消息方向：device->server->device

请求数据包	
需加密的请求内容	{ "sequence": "12345", "deviceId": "XXX", }

	<pre> "subDeviceId": "YYY" "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre> { "code":2016, "token":"qwertyuiop123456", "data":"XXXXXXXXXXXX" } </pre>
响应数据包	
需加密的响应内容	<pre> { "result": 0, "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code":2017, "data":"XXXXXXXXXXXX" } </pre>

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码, 根据不同消息代码不同
token	是	设备访问平台的令牌, 通过设备登录接口获取
sequence	是	接口调用序列号, 取值 0~ 65535, 回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID
subDeviceId	是	子设备 ID
time	是	平台/设备当前时间, linux 标准格式时间戳。 请求包中的 time 表示请求发起时间, 响应包中的 time 表示请求响应时间。
result	是	处理结果码, 为 0 表示成功, 非 0 表示失败, 错误码信息表见附录 1
data	是	数据加密后的内容

5. 4. 3. 12 子设备解绑上报

接口描述:

子设备从应用网关解绑后, 由应用网关向智能家居平台发起子设备解绑上报数据, 智能家居平台删除应用网关和子设备的关系。

消息方向: device->server->device

请求数据包	
需加密的请求内容	{

	<pre> "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre> { "code":2018, "token":"qwertyuiop123456", "data":"XXXXXXXXXXXX" } </pre>
响应数据包	
需加密的响应内容	<pre> { "result": 0, "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code":2019, "data":"XXXXXXXXXXXX" } </pre>

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码, 根据不同消息代码不同
token	是	设备访问平台的令牌, 通过设备登录接口获取
sequence	是	接口调用序列号, 取值 0~ 65535, 回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID
subDeviceId	是	子设备 ID
time	是	平台/设备当前时间, linux 标准格式时间戳。 请求包中的 time 表示请求发起时间, 响应包中的 time 表示请求响应时间。
result	是	处理结果码, 为 0 表示成功, 非 0 表示失败, 错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.13 子设备在线状态上报

接口描述:

子设备会通过心跳保持和应用网关的连接, 一旦子设备和应用网关断开连接或重新连接, 应用网关就需要通过该接口将子设备的在线状态上报到智能家居平台。

消息方向: device->server->device

请求数据包	
需加密的请求内容	{ "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "onLine": "1", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	{ "code":2020, "token":"qwertyuiop123456", "data":"XXXXXXXXXXXX" }
响应数据包	
需加密的响应内容	{ "result": 0, "sequence": "12345", "deviceId": "XXX", "subDeviceId": "YYY", "time": 1361542433 }
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	{ "code":2021, "data":"XXXXXXXXXXXX" }

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID
subDevices	是	子设备列表
subDeviceId	是	子设备 ID
onLine	是	设备是否在线，0：离线，1：在线
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.14 设备版本上报

接口描述:

设备上线时会通过该接口把自己及子模块的版本号信息上报到智能家居平台,如果智能家居平台发现设备或模块需要升级时,会发送消息通知设备有新的版本需要升级。

智能家居平台收到设备版本上报后,立即进行回复,并异步判断设备固件或模块是否需要升级,如果需要升级,将通过设备版本升级通知接口通知设备升级。

子设备的版本信息也通过该接口进行上报。

消息方向: device→server→device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "model": "XXXX", "device": { "version": "XXX.YYY.ZZZ.MMM" }, "modules": [{ "name": "aaaa", "version": "XXX.YYY.ZZZ.MMM" }, { "name": "bbbb", "version": "XXX.YYY.ZZZ.MMM" }], "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre>{ "code": 2022, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }</pre>
响应数据包	
需加密的响应内容	<pre>{ "result": 0, "sequence": "12345", "deviceId": "XXX", "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre>{</pre>

	<pre>"code":2023, "data":"XXXXXXXXXX" }</pre>
--	---

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~65535，回复方使用同样的序列号返回响应信息。
device.version	是	设备固件版本，版本号格式为 XXX.YYY.ZZZ.MMM，代表含义如下： XXX：主版本号 YYY：次版本号 ZZZ：小版本号 MMM：修订版本号
modules.name	否	设备模块名称
modules.version	否	设备模块版本，版本号格式为 XXX.YYY.ZZZ.MMM，代表含义如下： XXX：主版本号 YYY：次版本号 ZZZ：小版本号 MMM：修订版本号
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.15 设备版本升级通知

接口描述:

设备上线后的版本升级分为两种情况:

- 1) 静默升级: 智能家居平台端对某型号设备的某个版本配置了自动升级策略，则一旦设备上报的版本信息符合升级策略时，会由智能家居平台直接推送给设备进行升级。
- 2) 客户端查询升级: 智能家居平台没有对某型号设备配置自动升级，则要想给设备升级，需要先通过 APP 查询设备版本升级信息并提示用户，当设备当前版本配置了升级版本时，提示用户可对设备进行版本升级，由用户确认升级后，再发送该接口消息进行升级。

子设备的版本升级通知也通过该接口下发，应用网关判断 **deviceId** 不是自身 ID 时，查询下属子设备是否有符合的，有则进行 ROM 下载升级操作，并上报升级结果。

消息方向: server->device ->server

请求数据包	
需加密的请求内容	<pre>{ "sequence":"12345", "deviceId":"XXX", }</pre>

	<pre> "device":{ "version":"XXX.YYY.ZZZ.MMM", "url":"http://XXXX", "md5":"asdfghjkl" }, "modules":[{ "name":"aaaa", "version":" XXX.YYY.ZZZ.MMM", "url":"http://XXXX", "md5":"asdfghjkl" }, { "name":"bbbb", "version":" XXX.YYY.ZZZ. MMM", "url":"http://YYYY", "md5":"asdfghjkl" }], "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的设备请求数据包	<pre> { "code":2025, "data":"XXXXXXXXXXXX" } </pre>
响应数据包	
需加密的响应内容	<pre> { "result":0, "sequence":"12345", "deviceId": "XXX", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code":2024, "token":"qwertyuiop123456", "data":"XXXXXXXXXXXX" } </pre>

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序

		列号返回响应信息。
device.version	是	设备固件的最新版本，版本号格式为 XXX.YYY.ZZZ.MMM，代表含义如下： XXX：主版本号 YYY：次版本号 ZZZ：小版本号 MMM：修订版本号
device.url	是	设备固件的最新版本下载地址
device.md5	是	设备固件的最新版本安装文件摘要，防止安装文件被篡改，设备要做校验
modules.name	否	设备模块名称
modules.version	否	设备模块的最新版本，版本号格式为 XXX.YYY.ZZZ.MMM，代表含义如下： XXX：主版本号 YYY：次版本号 ZZZ：小版本号 MMM：修订版本号
modules.url	否	设备模块最新版本下载地址
modules.md5	否	设备模块的最新版本安装文件摘要，防止安装文件被篡改，设备要做校验
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.16 设备版本升级结果上报

接口描述：

设备在获取到最新版本信息后会自行下载升级包，下载到本地后执行升级。设备通过本接口向智能家居平台上报升级结果。

子设备的升级版本由应用网关进行下载，并进行升级操作，升级完成后由应用网关通过该接口上报升级结果。

智能家居平台收到设备版本上报后，立即进行回复，并会广播该消息给该设备的所有在线拥有者的 APP。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "type": "1", "moduleName": "XXXX", //type 为 0 时，不需该字段 "status": "0", "version": "XXX.YYY.ZZZ.MMM",</pre>

	<pre> "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre> { "code":2026, "token":"qwertyuiop123456", "data":"XXXXXXXXXX" } </pre>
响应数据包	
需加密的响应内容	<pre> { "result":0, "sequence":"12345", "deviceId": "XXX", "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code":2027, "data":"XXXXXXXXXX" } </pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
type	是	用于指示要升级的是设备还是模块，默认为 0。不允许同时升级设备和模块。 0：给设备升级 1：给模块升级
moduleName	否	模块名称，当进行模块升级时此处不为空
status	是	表示升级进度状态，规定如下： 0：升级成功 1：升级失败
version	是	表示当前设备或模块的版本号
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	处理结果码，为 0 表示成功，非 0 表示失败，错误码信息表见附录 1
data	是	数据加密后的内容

5.4.3.17 语音控制

接口描述：

智能设备如果具备拾音功能，就可以通过此接口向智能家居平台发起语音控制请求。

智能家居平台根据设备上传的语音数据，进行语音识别并得到设备控制的指令，并在智能家居平台执行相关指令，最后智能家居平台将指令执行结果返回给智能设备。

为了减少语音识别过程的整体耗时，语音控制接口支持分段接收语音数据，从而使得智能设备在录音的同时，语音数据也被同步上传。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "ctrlParam": { "format": "amr", "rate": 8000, "channel": 1, "voiceId": "A08CF897B20170101120000", "sn": 1, "isLast": "0", "audioData": "", "extensions": [{ "key": "", "value": "" }] }, "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre>{ "code": 3002, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }</pre>
响应数据包	
需加密的响应内容	<p>语音控制被识别处理后的响应：</p> <pre>{ "result": 0, "sequence": "12345", "deviceId": "XXX", "ctrlResult": { "rawText": "把空调调到 28 度", "resultDesc": "已将空调温度设置为 28 度" }, "time": 1361542433 }</pre>

	<pre> } 分段数据发送成功后的响应: { "result": 400002, "sequence": "12345", "deviceId": "XXX", "ctrlResult": { }, "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code":3003, "data":"XXXXXXXXXXXX" } </pre>

参数说明:

字段名称	是否必选	说明
code	是	消息协议代码, 根据不同消息代码不同
token	是	设备访问平台的令牌, 通过设备登录接口获取
sequence	是	接口调用序列号, 取值 0~ 65535, 回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID, 为应用网关或子设备的 ID
ctrlParam	是	语音控制输入参数
format	否	语音数据压缩格式, 取值: pcm speex amr 默认值为 amr
rate	否	语音数据采样率, 取值: 8000 16000 默认值为 16000
channel	否	语音数据声道 1: 单声道 默认值为 1
voiceId	否	语音数据编号。 在录音数据分段录制并分段传输时进行赋值, 一次录音过程该编号相同, 格式: deviceId+时间戳
sn	否	每个分段录音数据的顺序号, 从 1 开始递增, 需要设备按照分段顺序上传, 语音分段上传时, sequence 使用同一个
isLast	否	是否是最后一段数据, 取值: 1: 是最后一段数据 0: 不是最后一段数据
audioData	是	音频数据经 Base64 编码后的字符串

extensions	否	扩展字段 key/value 数据组
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表示请求响应时间。
result	是	语音控制结果码，为 0 表示成功，非 0 表失败，错误码信息表见附录 1
data	是	数据加密后的内容
ctrlResult	否	语音控制结果
rawText	否	语音识别转写后的文本
resultDesc	否	设备控制结果描述

5.4.3.18 语音合成

接口描述：

当智能设备需要通过语音播报设备状态、告警、通知、设备控制处理结果等时，可以通过该接口向智能家居平台发起语音合成请求，输入文本获取到相应语音数据。

智能设备能够设置语音合成的发音人、语速、语调以及合成的音频数据的编码格式、采用率和声道等。智能设备在接收到智能家居平台返回合成后的音频数据后，将音频编码格式进行转换即可进行播放。

消息方向：device->server->device

请求数据包	
需加密的请求内容	<pre>{ "sequence": "12345", "deviceId": "XXX", "ttsParam": { "content": "已为您打开空调！", "anchor": "xiaoyan", "speed": 50, "volume": 50, "pitch": 50, "format": "amr", "rate": 16000, "channel": 1 }, "time": 1361542433 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , sessionKey))
完整的请求数据包	<pre>{ "code": 3000, "token": "qwertyuiop123456", "data": "XXXXXXXXXXXX" }</pre>
响应数据包	
需加密的响应内容	{

	<pre> "result": 0, "sequence": "12345", "deviceId": "XXX", "ttsResult": { "audioData": "" }, "time": 1361542433 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , sessionKey))
完整的响应数据包	<pre> { "code":3001, "data":"XXXXXXXXXXXX" } </pre>

参数说明：

字段名称	是否必选	说明
code	是	消息协议代码，根据不同消息代码不同
token	是	设备访问平台的令牌，通过设备登录接口获取
sequence	是	接口调用序列号，取值 0~ 65535，回复方使用同样的序列号返回响应信息。
deviceId	是	设备 ID，为应用网关或子设备的 ID
ttsParam	是	语音合成输入参数
content	是	语音合成文本内容
anchor	否	发音人标识 默认值为家开平台默认中文发音人
speed	否	语速，取值 0~100 默认 50
volume	否	音量，取值 0~100 默认 50
pitch	否	音调，取值 0~100 默认 50
format	否	合成的语音编码格式，取值： pcm speex amr 默认值为 amr
rate	否	采样率，取值： 8000 16000 默认值为 16000
channel	否	声道，取值 1: 单声道 默认值为 1
time	是	平台/设备当前时间，linux 标准格式时间戳。 请求包中的 time 表示请求发起时间，响应包中的 time 表

		示请求响应时间。
result	是	语音合成结果码，为 0 表示成功，非 0 表失败，错误码信息表见附录 1
data	是	数据加密后的内容
ctrlResult	是	语音合成输出结果
audioData	是	合成的音频数据经 Base64 编码后的字符串

5.4.4 错误码列表

error_code	错误码说明	备注
0	成功	
100001	请求参数非法	包括参数格式不正确，必选参数为空等
200001	设备 ID 不存在	
200002	设备 PIN 码错误	
200003	设备鉴权失败	包括设备未备案
300001	设备或子设备不在线	
300002	设备或子设备不支持该指令	
400001	发音人不存在	
400002	等待剩余语音数据上传	
400003	网关下没有语音所提设备	
400004	系统不支持语音所提业务	
400005	语音识别失败	
400001	发音人不存在	
899999	设备侧未知错误	
999999	平台侧未知错误	

6 设备与 APP 接口 (Md)

6.1 接口介绍

局域网目前支持设备的发现、状态查询、状态控制和状态上报。设备发现协议支持二进制方式，设备的认证、查询、控制、状态上报支持 JSON 协议格式。

设备发现就是局域网内通过手机 APP 搜索设备的过程，设备搜索出来可能有两种情况，一种是设备刚刚配网完成，尚未退出配网状态，则设备返回 bindKey 值，此时设备可以被用户绑定。另一种是设备处于正常联网状态，可以与设备正常交互但不能添加设备。

设备发现过程中，设备端监听 7680 端口。获取到设备 IP 地址后连接设备时设备监听端口为 7681。

二进制格式数据：该种方式下包数据部分为设备可以识别的二进制数据。APP 端的数据需要首先通过加载转换脚本，把数据转换成设备可以识别的二进制数据。此时 WiFi 模块处理比较简单，只需要透传即可。

JSON 格式数据：该种方式下包数据部分为纯 JSON 格式，APP 端处理比较简单，WiFi 模块需要对 JSON 数据进行解析。

6.2 接口定义

6.2.1 设备发现协议

接口描述：

客户端在局域网内发送 UDP 广播包，搜索指定型号设备或者搜索全部设备（不带设备型号时），设备收到广播包后进行回复，如果当前设备的型号不是搜索型号，则不返回。客户端根据设备的返回结果进行相应的操作。如果设备处于待添加状态（刚刚配网结束），则设备会返回基本信息和 bindKey 值作为用户绑定设备的输入条件，否则仅返回设备的基本信息。

消息方向：app->device->app

请求数据包	
请求内容	<pre>typedef struct{ u8 modelLength; //设备型号长度 u8 model[]; //不定长数据，表示设备型号 u32 timestamp; //客户端当前时间秒数 u16 platform; //客户端平台代号 }data_search;</pre>
响应数据包	
响应内容	<pre>typedef struct{ u16 code; //错误代码，具体参考附录 u8 deviceIdLength; //表示设备长度 u8 deviceId[]; //不定长数据，表示设备 ID u8 modelLength; //设备型号长度 u8 model[]; //设备型号，不定长字符串 u8 protocol; //设备支持的数据协议 u8 lanControl; //是否支持局域网内控制 u8 onConfig; //是否正在处于待添加状态 u64 bindKey; //设备 bindKey }data_response;</pre>

参数说明：

字段名称	是否必选	说明
modelLength	是	设备的型号长度
model	是	要搜索的设备型号，不定长数据，如果不携带该属性表示搜索所有型号设备
timestamp	是	客户端在搜索设备时的当前时间的秒数
platform	是	客户端平台代号，默认为 0，表示电信云，非 0 代表其他
code	是	操作结果码，为 0 表示操作成功，非 0 表示错误码信息
deviceIdLength	是	设备 ID 的长度
deviceId	是	设备 ID，不定长字符串
mac	是	定长 6 个字节和设备 MAC 地址
modelLength	是	设备的型号长度
model	是	设备的型号，不定长字符串，长度根据 modelLength 获

		取
lanControl	是	是否支持局域网下的控制，1 支持，2 不支持，如果设备尚未授权返回 0
protocol	是	通信时数据部中内容的格式。目前支持 json，后续可以支持其它协议。 0: json 格式
onConfig	是	如果设备正在处于被添加状态则返回 1，否则者返回 0，设备未配置时，通电后 120 秒内处于可配置状态
bindKey	是	设备的绑定密钥，仅当 onConfig 为 1 时有效，当设备刚配网完成，仍处于被添加状态时，才会返回该字段

6.2.2 设备认证协议

接口描述：

客户端在局域网内发现设备后，首先根据设备 IP 地址向设备的 7681 端口发起 TCP 连接，连接成功后，客户端会生成一个随机的 16 位整数 X，并发送到智能家居平台，平台使用设备 PIN 码进行加密，并把加密结果 lanKey 返回给客户端，客户端将随机数 X 和加密结果一同发给设备。设备对 lanKey 进行解密，解密数与 X 一致，则认定客户端合法，否则客户端非法。后续客户端及设备使用 lanKey 作为通信加密密钥。

消息方向：app->device->app

请求数据包	
需加密的请求内容	<pre>{ "cmd": "auth", "to": "deviceId", "rand": "x", "lanKey": "xxx", "liveTime": "300" }</pre>
响应数据包	
需加密的响应内容	<pre>{ "cmd": "resp", "code": 0 }</pre>

参数说明：

字段名称	是否必选	说明
rand	是	16 位定长随机数
lanKey	是	局域网控制密钥及认证数据
liveTime	是	局域网控制密钥失效时间，单位秒
code	是	操作结果码，为 0 表示操作成功，非 0 表示错误码信息

6.2.3 设备状态查询

接口描述：

当需要获取设备的当前状态时，可以通过此命令进行查询，设备会把当前状态集合按照规定的格式返回。如果恰巧此时设备出现故障，则也会返回故障码列表(fids)；当设备正在升级时，会返回升级状态和进度。

消息方向：app->device->app

请求数据包	
需加密的请求内容	<pre>{ "cmd": "query", "to": "deviceId", "mid": "0001233" }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , lanPin))
响应数据包	
需加密的响应内容	<pre>{ "cmd": "resp", "mid": "0001233", "code": 0, "as": { //设备的属性编号集合 "1": 2, "2": 3, }, "asc": { "1": 1, //属性 1 只读 "2": 0, //属性 2 禁止读写 }, "fids": [1, 23, 45], //如果设备有故障，可以通过 此处上报故障码 "ota": [0, 1, 85], //如果设备正升级，则标识设备 当前升级状态 }</pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , lanPin))

参数说明：

字段名称	是否必选	说明
cmd	是	操作类型
code	是	操作结果码，为 0 表示操作成功，非 0 表示错误码信息
mid	是	消息序列号，此处需要与 query 命令中的 mid 保持一致
as	是	<p>as (attributes) 表示设备属性集合，集合对象中每个键值对表示设备的一个属性。键值对的键为设备属性的编号，键值对中的值表示设备属性的值。</p> <p>比如空调有两个属性：开关和模式，用 1 表示开关，2 表示模式。而开关有两个状态（开和关），表示为 1 和 0，模式有四个状态（制冷、制热、换气、除湿），表示为 1, 2, 3, 4，如果此时空调的状态为打开，模式为除湿，则可以表示为： "as": {"1": 1, "2": 4}</p>
asc	否	asc (Attributes Constraints) 属性权限约束，集合对象中每个键值对表示设备的一个属性约束。键值对的键为设备属

		性的编号，键值对中的值表示设备属性的约束类型。目前支持两种约束类型，0：禁止读写 1：只读属性。
fids	否	当设备出现故障时，携带故障码列表
ota	否	<p>设备的升级状态信息三元组，以数组方式表示，数组有三个元素，数组的第一个元素表示要升级的是模块还是设备，默认为 0。具体取值如下：</p> <p>0：给模块升级</p> <p>1：给设备升级</p> <p>注意：不允许同时升级模块和设备。</p> <p>数组的第二个元素表示升级进度，规定如下：</p> <p>1：开始下载</p> <p>2：下载中</p> <p>3：下载成功</p> <p>4：下载失败</p> <p>5：开始升级</p> <p>6：升级中</p> <p>7：升级成功</p> <p>8：升级失败</p> <p>数组的第三个元素表示当前升级进度下的百分比值，默认为 0</p>

6.2.4 设备状态控制

接口描述：

当需要获取设备的当前状态时，可以通过此命令进行查询，设备会把当前状态集合按照规定的格式返回。如果恰巧此时设备出现故障，则也会返回故障码列表(fids)；当设备正在升级时，会返回升级状态和进度。

消息方向：app->device->app

请求数据包	
需加密的请求内容	<pre>{ "cmd": "opt", "mid": "111222", "to": "deviceId", "as": { "1": 2, "2": 3, } }</pre> <p>//设备的属性编号集合</p>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , lanPin))
响应数据包	
需加密的响应内容	<pre>{ "cmd": "post", "mid": "1234", "from": "deviceId", "as": { "1": 2, } }</pre> <p>//设备的属性编号集合</p>

	<pre> "2":3, }, "fids":[1, 23, 45], //如果设备有故障，可以通过 此处上报故障码 "ota":[0, 1, 85], //如果设备正升级，则标识设 备当前升级状态 } </pre>
使用会话密钥进行加密	data = Base64 (AES (响应内容 , lanPin))

参数说明：

字段名称	是否必选	说明
cmd	是	操作类型
mid	是	<p>当客户端操作设备时，需要对操作设备的命令进行响应，mid 的值与控制命令的 mid 相同，以便标识当前消息时对控制命令的响应，举例如下：</p> <p>设备收到操作设备的命令如下：</p> <pre>{ "cmd": "opt", "mid": "12345", "as": { "1": 3 } }</pre> <p>则如果设备响应成功，进行回复时，发送如下内容：</p> <pre>{ "cmd": "post", "mid": "12345", "as": { "1": 3 } }</pre> <p>另外：如果是设备主动上报状态变化，无需携带 mid</p>
from	是	此处为设备或网关 ID，用于标识消息的来源， from 的值与 opt 命令中的 to 保持一致
as	是	<p>（attributes 缩写）设备属性集合，集合对象中每个键值对表示设备的一个属性。键值对的键为设备属性的编号，键值对中的值表示设备属性的值。</p> <p>比如空调有两个属性：开关和模式，用 1 表示开关，2 表示模式。而开关有两个状态（开和关），表示为 1 和 0，模式有四个状态（制冷、制热、换气、除湿），表示为 1, 2, 3, 4，如果此时空调的状态为打开，模式为除湿，则可以表示为：</p> <pre>"as": { "1": 1, "2": 4 }</pre>
fids	否	当设备出现故障时，携带故障码列表
ota	否	<p>设备的升级状态信息三元组，以数组方式表示，数组有三个元素，数组的第一个元素表示要升级的是模块还是设备，默认为 0。具体取值如下：</p> <p>0：给模块升级 1：给设备升级</p> <p>注意：不允许同时升级模块和设备。</p> <p>数组的第二个元素表示升级进度，规定如下：</p> <p>1：开始下载 2：下载中 3：下载成功 4：下载失败 5：开始升级 6：升级中</p>

		7: 升级成功 8: 升级失败 数组的第三个元素表示当前升级进度下的百分比值, 默认为 0
--	--	---

6.2.5 设备状态上报

接口描述:

当设备的状态发生变化时, 需要上报给局域网内的客户端。

消息方向: device→app

请求数据包	
需加密的请求内容	<pre>{ "cmd": "post", "mid": "1234", "from": "deviceId", "as": { //设备的属性编号集合 "1": 2, "2": 3, }, "fids": [1, 23, 45], //如果设备有故障, 可以通过 此处上报故障码 "ota": [0, 1, 85], //如果设备正升级, 则标识设备 当前升级状态 }</pre>
使用会话密钥进行加密	data = Base64 (AES (连接请求内容 , lanPin))

参数说明:

字段名称	是否必选	说明
cmd	是	操作类型
mid	是	<p>当客户端操作设备时, 需要对操作设备的命令进行响应, mid 的值与控制命令的 mid 相同, 以便标识当前消息时对控制命令的响应, 举例如下:</p> <p>设备收到操作设备的命令如下:</p> <pre>{ "cmd": "opt", "mid": "12345", "as": { "1": 3 } }</pre> <p>则如果设备响应成功, 进行回复时, 发送如下内容:</p> <pre>{ "cmd": "post", "mid": "12345", "as": { "1": 3 } }</pre> <p>另外: 如果是设备主动上报状态变化, 无需携带 mid</p>
from	是	此处为设备或网关 ID, 用于标识消息的来源, from 的值与 opt 命令中的 to 保持一致
as	是	<p>(attributes 缩写) 设备属性集合, 集合对象中每个键值对表示设备的一个属性。键值对的键为设备属性的编号, 键值对中的值表示设备属性的值。</p> <p>比如空调有两个属性: 开关和模式, 用 1 表示开关, 2 表示模式。而开关有两个状态 (开和关), 表示为 1 和 0, 模式有四个状态 (制冷、制热、换气、除湿), 表示为 1, 2, 3, 4, 如果此时空调的状态为打开, 模式为除湿, 则可以表示为:</p> <pre>"as": { "1": 1, "2": 4 }</pre>

fids	否	当设备出现故障时，携带故障码列表
ota	否	<p>设备的升级状态信息三元组，以数组方式表示，数组有三个元素，数组的第一个元素表示要升级的是模块还是设备，默认为 0。具体取值如下：</p> <p>0：给模块升级 1：给设备升级</p> <p>注意：不允许同时升级模块和设备。</p> <p>数组的第二个元素表示升级进度，规定如下：</p> <p>1：开始下载 2：下载中 3：下载成功 4：下载失败 5：开始升级 6：升级中 7：升级成功 8：升级失败</p> <p>数组的第三个元素表示当前升级进度下的百分比值，默认为 0</p>

6.2.6 错误码列表

error_code	错误码说明	备注
0	成功	
100001	请求参数非法	包括参数格式不正确，必选参数为空等
200001	设备 ID 不存在	
200002	设备 PIN 码错误	
200003	设备鉴权失败	包括设备未备案
300001	设备或子设备不在线	
300002	设备或子设备不支持该指令	
400001	发音人不存在	
400002	等待剩余语音数据上传	
400003	网关下没有语音所提设备	
400004	系统不支持语音所提业务	
400005	语音识别失败	
400001	发音人不存在	
899999	设备侧未知错误	
999999	平台侧未知错误	

7 与第三方业务平台对接接口（C）

7.1 接口介绍

第三方业务平台对接接口 C 定义了第三方业务平台与电信智能家居平台的对接规范，

第三方设备通过平台互通方式接入电信智能家居平台时，按此接口定义进行对接。

此接口适用于平台互通对接模式。

7.1.1 接口示意图

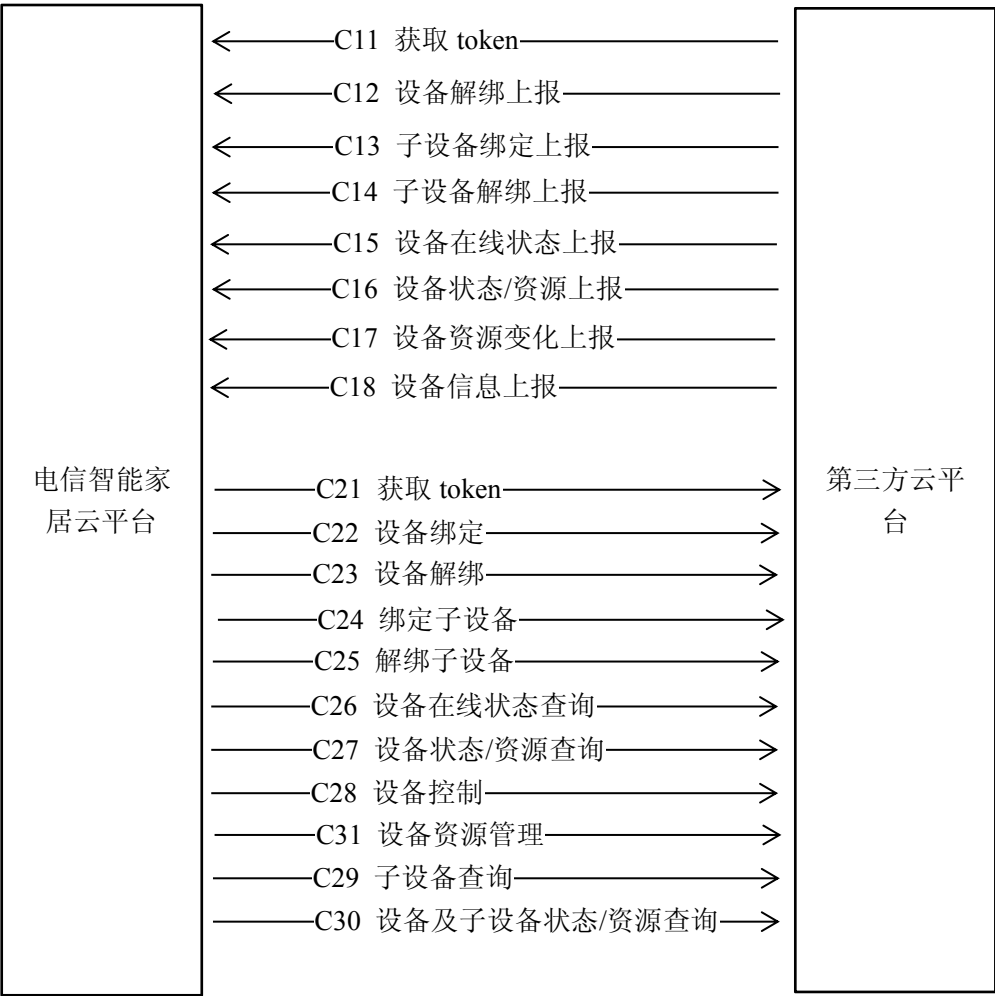


图 5 智能家居平台与第三方业务平台之间接口示意图

7.1.2 通信方式

智能家居平台与第三方业务平台之间的通信采用接 HTTPS 协议，进行双向证书认证；服务端为客户端提供访问的 URL，客户端使用 HTTPS 方法发送请求报文并得到应答报文，HTTPS 响应采用 JSON 格式。

7.1.3 参数类型说明

类型	描述
String	字符串，UTF-8 编码，注意有些字段如描述、URL 可能出现非 ascii 的，解包时需要按 utf8 解码
Integer	整型
float	浮点型字符串
Date	定长 String 格式:YYYYMMDD
DateTime	定长 String 格式:YYYYMMDDHHMMSS

7.1.4 安全策略

智能家居平台与第三方业务平台之间的通信采用接 HTTPS 协议，进行双向证书认证。

7.2 关键流程

本流程用于平台互通模式下的智能家居设备接入模式下，平台鉴权、设备配网、设备绑定、设备控制等环节的交互流程说明。

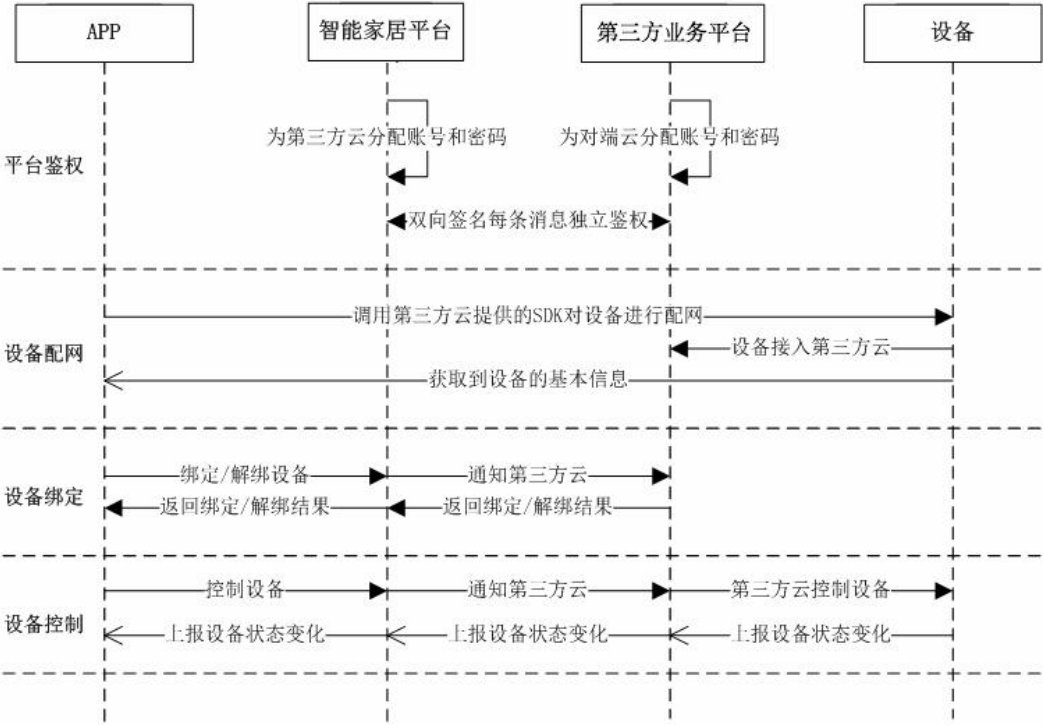


图 6 平台互通流程

注：本流程中的设备指 IP 应用终端，应用网关与下挂的 Zigbee、Z-wave、RF 终端之间的数据交互根据应用网关上的定义执行。

【流程说明】

- 1) **平台鉴权：**智能家居平台与第三方业务平台在线下交换用于身份认证的 ID 和 Key 密钥对。智能家居平台与第三方业务平台通过签名进行平台的相互鉴权，以确认对方是合法的对接平台。
- 2) **设备配网：**APP 使用第三方业务平台提供的设备配网 SDK 配置设备接入第三方业务平台，同时获得设备的基本信息。
- 3) **设备绑定：**APP 向智能家居平台请求绑定设备，智能家居平台同步通知第三方业务平台绑定设备。
- 4) **设备控制：**设备绑定成功后，APP 就可以控制设备时，智能家居平台将操控指令经过协议转换后发送到第三方业务平台（或将操控指令先发送到第三方业务平台，由第三方业务平台进行协议转换），并执行设备控制。当设备状态发生变化时，由第三方业务平台将状态信息发送到智能家居平台，智能家居平台进行协议转换（或由第三方业务平台协议转换后将状态数据上报智能家居平台）、数据分析、存储，并通知 APP 等。

7.3 接口定义

7.3.1 智能家居平台提供的接口

本章节定义智能家居平台提供面向第三方业务平台的开放接口。

7.3.1.1 C11 获取 Token 接口

当第三方业务平台调用智能家居平台接口时，需要校验 Token，比如：设备上线、设备下线、子设备上线、子设备下线、设备状态变化等。

接口名称	获取 Token 接口
接口描述	第三方业务平台调用该接口获取智能家居平台的 Token
请求路径	https://{domain}/notify/token
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
platformId	是	String	body	由第三方业务平台分配给智能家居平台的身份 ID，用于第三方业务平台的身份认证
platformKey	是	String	body	由第三方业务平台分配给智能家居平台的密钥，用于第三方业务平台的身份认证
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	是	String	body	返回码说明
access_token	是	String	body	Token 值
expiredAt	是	Long	body	Token 有效期，系统时间戳格式

7.3.1.2 C12 设备解绑上报接口

当第三方业务平台接收到设备解绑成功时，调用该接口通知到智能家居平台，智能家居平台对上报数据进行存储，通知手机 APP 等处理。

接口名称	设备解绑上报接口
接口描述	当设备解绑成功时，第三方业务平台调用该接口将设备绑定数据上报到智能家居平台。
请求路径	https://{domain}/notify/device/unbind
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
------	------	----	----	------

access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	否	String	body	返回码描述

7.3.1.3 C13 子设备绑定上报接口

当第三方业务平台接收到子设备绑定成功时，调用该接口通知到智能家居平台，智能家居平台对上报数据进行存储，通知手机 APP 等处理。

接口名称	子设备绑定上报接口
接口描述	当子设备绑定成功时，第三方业务平台调用该接口将设备绑定数据上报到智能家居平台。
请求路径	https://{domain}/notify/device/sub/bind
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
gwId	是	String	body	应用网关 ID
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	否	String	body	返回码描述

7.3.1.4 C14 子设备解绑上报接口

当第三方业务平台接收到子设备解绑成功时，调用该接口通知到智能家居平台，智能家

居平台对上报数据进行存储，通知手机 APP 等处理。

接口名称	子设备解绑上报接口
接口描述	当子设备解绑成功时，第三方业务平台调用该接口将设备绑定数据上报到智能家居平台。
请求路径	https://{domain}/notify/device/sub/unbind
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
gwId	是	String	body	应用网关 ID
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	否	String	body	返回码描述

7.3.1.5 C15 设备在线状态上报接口

当第三方业务平台接收到设备或子设备离线或在线时，调用该接口通知到智能家居平台，智能家居平台对上报数据进行存储，通知手机 APP 等处理。

接口名称	设备在线状态上报接口
接口描述	当设备或子设备离线或在线时，第三方业务平台调用该接口将数据上报到智能家居平台。
请求路径	https://{domain}/notify/device/online
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一

sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。
online	是	String	body	设备在线状态，1 在线，0 离线

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	否	String	body	返回码描述

7.3.1.6 C16 设备状态/资源上报接口

当第三方业务平台接收到设备或子设备状态变化时，调用该接口通知到智能家居平台，智能家居平台对上报数据进行存储，通知手机 APP 等处理。

接口名称	设备状态/资源信息上报接口
接口描述	当设备或子设备状态变化时，第三方业务平台调用该接口将数据上报到智能家居平台。设备连网时也通过该接口上报当前的资源信息，可分多次上报。 当分多次上报资源信息时，两条消息的间隔不得超过 120s，否则视为全部资源信息已经上报完毕。 当平台标记为离线的设备上报状态变化时，当前设备及上级设备在线状态自动设置为在线。
请求路径	https://{domain}/notify/device/status
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳 + 序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。
status	是	Array<StatusInfo>	body	设备状态列表
resource	否	Array<ResourceList>	body	设备资源列表，设备存在资源时，通过该自动上报

StatusInfo 定义：

参数名称	是否必选	类型	描述信息
------	------	----	------

serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
name	是	String	状态名称
value	是	String	状态值

ResourceList 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
resName	是	String	资源名称
resInfo	是	Array<ResDetail>	资源信息

ResDetail 定义:

参数名称	是否必选	类型	描述信息
key1	是	String	资源参数 (不同资源参数定义不一样)
key2	是	String	资源参数 (不同资源参数定义不一样)
...

响应参数:

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功, 其他-失败
desc	否	String	body	返回码描述

7.3.1.7 C17 设备资源变化上报接口

当第三方业务平台接收到设备或子设备资源变化时, 调用该接口通知到智能家居平台, 智能家居平台对上报数据进行存储, 通知手机 APP 等处理。

接口名称	设备资源变化上报接口
接口描述	当设备或子设备资源变化时, 第三方业务平台调用该接口将数据上报到智能家居平台。 当平台标记为离线的设备上报资源变化时, 当前设备及上级设备在线状态自动设置为在线。
请求路径	https://{domain}/notify/device/resource
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间

				戳 + 序列号 : YYYYMMDDHHMMSS xxxx, 其中 xxxx 为序列 号随机数。
resource	是	Array<ResourceList>	body	设备资源变化列表

ResourceList 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
resName	是	String	资源名称
resInfo	是	Array<ResDetail>	资源信息

ResDetail 定义:

参数名称	是否必选	类型	描述信息
mode	是	String	1: 新增, 2: 修改, 3: 删除
key1	是	String	资源参数 (不同资源参数定义不一样)
key2	是	String	资源参数 (不同资源参数定义不一样)
...

响应参数:

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功, 其他-失败
desc	否	String	body	返回码描述

7.3.1.8 C18 设备信息上报接口

当第三方业务平台接收到设备或子设备上报的信息时,调用该接口通知到智能家居平台,智能家居平台对上报数据进行存储,通知手机 APP 等处理。

接口名称	设备信息上报接口
接口描述	当设备或子设备产生信息时,第三方业务平台调用该接口将数据上报到智能家居平台。 当平台标记为离线的设备上报信息时,当前设备及上级设备在线状态自动设置为在线。
请求路径	https://{domain}/notify/device/event
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID, 设备所属产品型号的唯一标识,由第三方业务平台制定,不同设备型号必须保持唯

				一
sequenceNo	是	String	body	流水号。建议格式为时间戳 + 序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。
event	是	Array<EventList>	body	设备信息列表

ResourceInfo 定义：

参数名称	是否必选	类型	描述信息
eventId	否	String	消息编号，一次上报多条消息时必须填，用于同类消息区分
serialId	是	String	多路设备的线路序号(1,2,3...)，单路设备为 0，多路设备为 0 时代表全部
eventName	是	String	消息类型
eventInfo	是	Array<EventDetail>	消息内容

EventDetail 定义：

参数名称	是否必选	类型	描述信息
key1	是	String	消息参数（不同消息类型参数定义不一样）
key2	是	String	消息参数（不同消息类型参数定义不一样）
...

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	否	String	body	返回码描述

7.3.1.9 错误码

错误码	描述	说明
0	成功	
1001	Token 失效	Token 失效
1002	请求参数错误	参数异常
1003	设备不存在	设备不存在
9999	失败	系统异常

7.3.2 第三方业务平台提供的接口

本章节作为参考，定义了第三方业务平台提供面向智能家居平台的开放接口，第三方业务平台可按照本章节接口开发。

7.3.2.1 C21 获取 Token 接口

当 APP 通知智能家居平台进行第三方业务平台的设备操作时，比如：设备绑定、解绑、控制、查询状态等，第三方业务平台需要校验 Token，该 Token 有第三方业务平台提供。

接口名称	获取 Token 接口
接口描述	智能家居平台调用该接口获取第三方业务平台的 Token
请求路径	https://{domain}/token
请求类型	POST

请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
cloudId	是	String	body	智能家居平台分配给第三方业务平台的身份 ID，用于智能家居平台对第三方业务平台的身份验证
cloudKey	是	String	body	智能家居平台分配给第三方业务平台的密钥，用于智能家居平台对第三方业务平台的身份验证
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功，其他-失败
desc	是	String	body	返回码说明
access_token	是	String	body	Token 值
expiredAt	是	Long	body	Token 有效期，系统时间戳格式

7.3.2.2 C22 绑定设备接口

当 APP 通知智能家居平台设备（比如：网关）的绑定操作时，由智能家居平台调用本接口将绑定消息通知到第三方业务平台，进而实现设备的绑定。

智能家居平台维护用户与设备的绑定关系。

接口名称	绑定设备接口
接口描述	智能家居平台调用该接口完成设备在第三方业务平台绑定
请求路径	https://{domain}/device/bind
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
bindKey	否	String	body	设备绑定密钥
deviceType	否	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中 xxxx 为序列号随机数。

响应参数：

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功, 其他-失败
desc	是	String	body	返回码说明
deviceSN	是	String	body	设备原始 SN
deviceType	是	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号: YYYYMMDDHHMMSSxxxx, 其中 xxxx 为序列号随机数。

7.3.2.3 C23 解绑设备接口

当 APP 通知智能家居平台设备（比如：网关）的解绑操作时，由智能家居平台调用本接口将解绑消息通知到第三方业务平台，进而解除设备的绑定关系。

智能家居平台维护用户与设备的解绑关系。

接口名称	解绑设备接口
接口描述	智能家居平台调用该接口完成设备在第三方业务平台解绑
请求路径	https://{domain}/device/unbind
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号: YYYYMMDDHHMMSSxxxx, 其中 xxxx 为序列号随机数。

响应参数:

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功, 其他-失败
desc	否	string	body	返回码描述

7.3.2.4 C24 绑定子设备接口

当 APP 通知智能家居平台子设备入网时的绑定操作，由智能家居平台调用本接口将绑定消息通知到第三方业务平台，进而实现子设备的绑定。

智能家居平台维护用户与子设备的绑定关系。

接口名称	绑定子设备接口
接口描述	智能家居平台调用该接口完成设备在第三方业务平台绑定
请求路径	https://{domain}/device/sub/bind

请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
gwId	是	String	body	应用网关 ID
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号: YYYYMMDDHHMMSSxxxx, 其中 xxxx 为序列号随机数。

响应参数:

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功, 其他-失败
desc	是	String	body	返回码说明

7.3.2.5 C25 解绑子设备接口

当 APP 通知智能家居平台设备 (比如: 网关) 的解绑操作时, 由智能家居平台调用本接口将解绑消息通知到第三方业务平台, 进而解除子设备的绑定关系。

智能家居平台维护用户与子设备的解绑关系。

接口名称	解绑子设备接口
接口描述	智能家居平台调用该接口完成设备在第三方业务平台解绑
请求路径	https://{domain}/device/sub/unbind
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
gwId	是	String	body	应用网关 ID
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号: YYYYMMDDHHMMSSxxxx, 其中 xxxx 为序列号随机数。

响应参数:

参数名称	是否必选	类型	位置	描述信息
------	------	----	----	------

code	是	Int	body	返回码,0-成功, 其他-失败
desc	否	string	body	返回码描述

7.3.2.6 C26 设备在线状态查询接口

智能家居平台调用本接口查询单个设备的在线状态。

接口名称	设备在线状态查询
接口描述	智能家居平台调用该接口能够查询单个设备是否在线
请求路径	https://{domain}/device/online/get
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号: YYYYMMDDHHMMSSxxxx, 其中 xxxx 为序列号随机数。

响应参数:

参数名称	是否必选	类型	位置	描述信息
code	是	Int	body	返回码,0-成功, 其他-失败
desc	是	String	body	返回码描述
online	是	Int	body	0-离线; 1-在线

7.3.2.7 C27 设备状态/资源查询接口

智能家居平台调用本接口查询单个设备的完整状态, 包括设备的所有业务属性值。

接口名称	设备状态/资源查询
接口描述	智能家居平台调用该接口查询单个设备的完整状态及资源信息
请求路径	https://{domain}/device/status/get
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号:

				YYYYMMDDHHMMSSxxxx, 其中 xxxx 为序列号随机数。
--	--	--	--	--------------------------------------

响应参数:

参数名称	是否必选	类型	描述信息
code	是	Int	返回码,0-成功, 其他-失败
desc	是	String	返回码描述
status	是	Array<StatusInfo>	设备状态列表
resource	否	Array<ResourceList>	设备资源列表, 设备存在资源时, 通过该自动上报

StatusInfo 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
name	是	String	状态名称
value	是	String	状态值

ResourceList 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
resName	是	String	资源名称
resInfo	是	Array<ResDetail>	资源信息

ResDetail 定义:

参数名称	是否必选	类型	描述信息
key1	是	String	资源参数 (不同资源参数定义不一样)
key2	是	String	资源参数 (不同资源参数定义不一样)
...

7.3.2.8 C28 设备控制操作接口

当 APP 发起设备控制操作时, 智能家居平台调用该接口将操作指令通知到第三方业务平台, 由第三方业务平台向设备下发。

接口名称	设备控制操作请求
接口描述	智能家居平台调用该接口完成单个设备的操作
请求路径	https://{domain}/device/control
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID, 设备所属产品型号的唯一

				一标识，由第三方业务平台制定，不同设备型号必须保持唯一
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中xxxx为序列号随机数。
serialId	是	String	body	控制多路设备的线路序号(1,2,3...)，单路设备为0，多路设备为0时代表全部控制
cmds	是	Array<CmdInfo>	body	设备控制命令组。

CmdInfo 定义：

参数名称	是否必填	类型	位置	描述信息
name	是	String	body	控制命令名称
value	是	String	body	控制命令参数值

响应参数：

参数名称	是否必选	类型	描述信息
code	是	Int	返回码,0-成功，其他-失败
desc	是	String	返回码描述
result	否	Array<StatusInfo>	设备状态列表

StatusInfo 定义：

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...)，单路设备为0，多路设备为0时代表全部
name	是	String	状态名称
value	是	String	状态值

7.3.2.9 C31 设备资源管理接口

当 APP 发起设备资源管理操作时，智能家居平台调用该接口将操作指令通知到第三方业务平台，由第三方业务平台向设备下发。

接口名称	设备资源管理请求
接口描述	智能家居平台调用该接口对设备资源进行操作
请求路径	https://{domain}/device/resManage
请求类型	POST
请求格式	
响应格式	

请求参数：

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
deviceSN	是	String	body	设备原始 SN
deviceType	否	String	body	设备类型 ID，设备所属产品型号的唯一标识，由第三方业务平台制定，不同设备型号必须保持唯一

sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中xxxx为序列号随机数。
resource	否	Array<ResourceList>	body	设备资源管理列表

ResourceList 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
resName	是	String	资源名称
resInfo	是	Array<ResDetail>	资源信息

ResDetail 定义:

参数名称	是否必选	类型	描述信息
mode	是	String	1: 新增, 2: 修改, 3: 删除
key1	是	String	资源参数 (不同资源参数定义不一样)
key2	是	String	资源参数 (不同资源参数定义不一样)
...

响应参数:

参数名称	是否必选	类型	描述信息
code	是	Int	返回码,0-成功, 其他-失败
desc	是	String	返回码描述

7.3.2.10 C29 子设备查询接口

当设备通过应用网关接入智能网关时,智能家居平台调用该接口查询某应用网关下的子设备列表。

设备和网关的唯一标识符不会重复,即设备和网关都是可以被独立寻址的。因此,本文档仅支持通过应用网关 ID 搜索其子设备 ID 的接口,

接口名称	应用网关子设备查询接口
接口描述	适用于当设备通过应用网关接入智能网关的场景。智能家居平台调用本接口查询某应用网关下的子设备列表。
请求路径	https://{domain}/gateway/subList
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
gwId	是	String	body	应用网关 ID
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号：YYYYMMDDHHMMSSxxxx，其中xxxx为序列号随机数。

响应参数:

参数名称	是否必选	类型	描述信息
code	是	Int	返回码,0-成功, 其他-失败
desc	是	String	返回码描述
gwId	是	String	应用网关 ID
deviceType	是	String	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
online	是	Int	0-离线; 1-在线
devices	是	Array<DeviceInfo>	子设备列表

DeviceInfo 定义:

参数名称	是否必选	类型	描述信息
deviceSN	是	String	设备原始 SN
deviceType	是	String	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
online	是	Int	0-离线; 1-在线

7.3.2.11 C30 设备及子设备状态/资源查询接口

当设备通过应用网关接入智能网关时,智能家居平台调用该接口查询某应用网关下的设备及子设备状态及资源列表。

设备和网关的唯一标识符不会重复,即设备和网关都是可以被独立寻址的。因此,本文档仅支持通过应用网关 ID 搜索其子设备 ID 的接口,

接口名称	设备及子设备状态/资源查询接口
接口描述	适用于当设备通过应用网关接入智能网关的场景。智能家居平台调用本接口查询某应用网关下的设备及子设备状态/资源信息。
请求路径	https://{domain}/gateway/subDetailList
请求类型	POST
请求格式	
响应格式	

请求参数:

参数名称	是否必填	类型	位置	描述信息
access_token	是	String	body	智能家居平台从第三方业务平台获取的
gwId	是	String	body	应用网关 ID
sequenceNo	是	String	body	流水号。建议格式为时间戳+序列号:YYYYMMDDHHMMSSxxxx, 其中xxxx为序列号随机数。

响应参数:

参数名称	是否必选	类型	描述信息
code	是	Int	返回码,0-成功, 其他-失败
desc	是	String	返回码描述
gwId	是	String	应用网关 ID
deviceType	是	String	设备类型 ID, 设备所属产品型号的

			唯一标识,由第三方业务平台制定,不同设备型号必须保持唯一
status	是	StatusInfo	设备状态列表
resource	否	Array<ResourceList>	设备资源列表, 设备存在资源时, 通过该自动上报
devices	是	Array<DeviceInfo>	子设备列表

DeviceInfo 定义:

参数名称	是否必选	类型	描述信息
deviceSN	是	String	设备原始 SN
deviceType	是	String	设备类型 ID, 设备所属产品型号的唯一标识, 由第三方业务平台制定, 不同设备型号必须保持唯一
status	是	Array<StatusInfo>	设备状态列表
resource	否	Array<ResourceList>	设备资源列表, 设备存在资源时, 通过该自动上报

StatusInfo 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
name	是	String	状态名称
value	是	String	状态值

ResourceList 定义:

参数名称	是否必选	类型	描述信息
serialId	是	String	多路设备的线路序号(1,2,3...), 单路设备为 0, 多路设备为 0 时代表全部
resName	是	String	资源名称
resInfo	是	Array<ResDetail>	资源信息

ResDetail 定义:

参数名称	是否必选	类型	描述信息
key1	是	String	资源参数 (不同资源参数定义不一样)
key2	是	String	资源参数 (不同资源参数定义不一样)
...

7.3.2.12 错误码

错误码	描述	说明
0	成功	
1001	Token 失效	Token 失效
1002	请求参数错误	参数异常
1003	设备不存在	设备不存在
9999	失败	系统异常