物联使能

物联使能 API

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2024 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【商标声明】

腾讯云

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

文档目录

物联使能 API
- 物联使能 API 简介
- 物联使能 API 概览
- 设备管理
  - 查看设备详情
  - 获取设备列表
  - 同步调用设备行为
  - 异步调用设备行为
  - 设备远程控制
- 数据查询
  - 获取设备物模型数据
  - 获取设备物模型历史数据
  - 获取设备的历史事件
- 数据模板
- 物联网安全保障服务
  - 服务领取
    - 创建用户信息
    - 获取用户信息列表
    - 创建订单
    - 查询订单列表
    - 获取保险领取页面
  - 设备与用户绑定关系推送
  - 设备数据推送
    - 设备数据上报
    - 设备状态上报
- 数据结构

版权所有：腾讯云计算（北京）有限责任公司

物联使能 API
物联使能 API 简介

最近更新时间：2024-09-30 10:38:01

简介

物联使能 API 是物联使能平台为帮助用户快速开发并高效托管物联网 SaaS 而提供的接口操作与云端服务，包括设备管理与控制、设备数据查询、数据模板查询等管理能力。用户可通过调用物联使能 API 来使用物联使能服务，支持的全部 API 请参见 物联使能 API 概览 。

调用方式

API URL 为 `https://iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
其中公共参数有： Action，RequestId，AppKey，Signature，Timestamp，Nonce。具体说明如下：

- **Action**：用于标识请求的方法名称。
- **RequestId**：用于标识一个唯一请求，推荐使用 uuid 作为参数值，定位问题时建议提供该参数值。
- **AppKey**：为 SaaS 的密钥。
- **Signature**：为本次请求的签名，具体计算方法见本文下方示例。
- **Timestamp**：为本次请求的 Unix 秒级时间戳。
- **Nonce**：为随机正整数，用于和时间戳一起，防范 API 重放攻击。

签名算法

获取 SaaS 的 AppKey 和 AppSecret

在创建 SaaS 时，平台将会为用户生成对应的安全凭证。安全凭证包括 AppKey 和 AppSecret。AppKey 是用于标识 API 调用者身份，AppSecret 是用于加密签名字符串和服务器端验证签名字符串的密钥。**用户应严格保管其AppSecret，避免泄露**。
具体获取步骤如下：

1. 登录 物联网开发平台控制台 ，选择已有项目进入项目详情页。
2. 选择左侧菜单**物联使能** > **SaaS 服务**，单击**新建**按钮 创建 SaaS 。
3. 创建 SaaS 成功后，即可获取系统自动生成的 AppKey 与 AppSecret。

生成签名串

有了安全凭证 AppKey 和 AppSecret 后，即可生成签名串。下面给出一个生成签名串的详细过程。
假设用户的 AppKey 和 AppSecret 分别是：

- AppKey： `ServiceAppKey`
- AppSecret： `ServiceAppSecret`

> **说明**
> 本文仅为示例，请您根据自己实际的 `AppKey` 和 `AppSecret` 进行后续操作。

以获取设备物模型数据 `ServiceDescribeDeviceData` 请求为例，当用户调用这一接口时，其请求参数可能如下：

参数名称	类型	描述	参考数值
RequestId	String	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需要提供该次请求的 RequestId	476c990a-f5b7-1575-987c-4ef70e474932
Action	String	公共参数，调用的接口方法名称	ServiceDescribeDeviceData
AppKey	String	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS	ServiceAppKey
Signature	String	公共参数，请求的签名	根据实际算法生成
Timestamp	Int64	公共参数，当前的 UNIX 时间戳（秒级）	1546315200
Nonce	Int64	公共参数，随机正整数，与时间戳一起，用于 API 防重放	71087795
ProductId	String	产品 ID	ProductA
DeviceName	String	设备名称	Device001

> **说明**
> - 请求参数中的公共请求参数有：RequestId、Action、AppKey、Timestamp、Nonce、Signature。ServiceDescribeDeviceData 接口特有参数：ProductId、DeviceName。

而参数 Signature（签名串）正是由上述参数共同生成的，具体步骤如下：

1. 对参数排序
   对所有请求参数按参数名做字典序升序排列，所谓字典序升序排列，直观上就如同在字典中排列单词一样排序，按照字母表或数字表里递增顺序的排列次序，即先考虑第一个 “字母”，在相同的情况下考虑第二个 “字母”，

依此类推。您可以借助编程语言中的相关排序函数来实现这一功能，例如 PHP 中的 ksort 函数。上述示例参数的排序结果如下：

```
{
        Action=ServiceDescribeDeviceData,
        AppKey=ServiceAppKey,
        DeviceName=Device001,
        Nonce=71087795,
        ProductId=ProductA,
        RequestId=476c990a-f5b7-1575-987c-4ef70e474932,
        Timestamp=1546315200
}
```

使用其它程序设计语言开发时，可对上面示例中的参数进行排序，得到的结果一致即可。

2. 拼接请求字符串
   将把上一步排序好的请求参数格式化成“参数名称” = “参数值”的形式，如对 Action 参数，其参数名称为 "Action"，参数值为 "AppCreateCellphoneUser"，因此格式化后就为 Action = AppCreateCellphoneUser。
   - “参数值” 为原始值而非 URL 编码后的值。
   - 若输入参数中包含下划线，则需要将其转换为 `“.”` 。

   将格式化后的各个参数用 `"&"` 拼接在一起，最终生成的请求字符串为：

```
Action=ServiceDescribeDeviceData&AppKey=ServiceAppKey&DeviceName=Devic
e001&Nonce=71087795&ProductId=ProductA&RequestId=476c990a-f5b7-1575-
987c-4ef70e474932&Timestamp=1546315200
```

3. 生成签名串
   使用 HMAC-SHA1 算法对上一步中获得的签名原文字符串进行签名，然后将生成的签名串使用 Base64 进行编码，即可获得最终的签名串。
   具体代码如下，以 PHP 语言为例：

```
$secretKey = 'ServiceAppSecret';
$srcStr =
'Action=ServiceDescribeDeviceData&AppKey=ServiceAppKey&DeviceName=Devi
ce001&Nonce=71087795&ProductId=ProductA&RequestId=476c990a-f5b7-1575-
987c-4ef70e474932&Timestamp=1546315200';
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey,
true));
echo $signStr
```

最终得到的签名串为：

```
P206d+JzP37FLKBDkD689wqnl4k=
```

使用其它程序设计语言开发时，可用上面示例中的原文进行签名验证，得到的签名串与例子中的一致即可。

物联使能 API 概览

最近更新时间：2021-08-31 10:57:11

设备管理

接口名称	接口功能
ServiceDescribeDevice	查看设备详情
ServiceGetDeviceList	获取设备列表
ServiceCallDeviceActionSync	同步调用设备行为
ServiceCallDeviceActionAsync	异步调用设备行为
ServiceControlDeviceData	设置控制设备的属性数据

数据查询

接口名称	接口功能
ServiceDescribeDeviceData	获取设备物模型数据
ServiceDescribeDeviceDataHistory	获取设备物模型历史数据
ServiceListEventHistory	获取设备历史事件

数据模板

接口名称	接口功能
ServiceDescribeModelDefinition	查询产品数据模板

数据订阅

接口名称	接口功能
CreateTopicRule	创建数据订阅规则
DeleteTopicRule	删除规则
DescribeTopicRule	获取规则信息
DisableTopicRule	禁用规则

版权所有：腾讯云计算（北京）有限责任公司

EnableTopicRule	启用规则
GetTopicRuleList	获取规则列表
ModifyTopicRule	修改规则
SearchTopicRule	搜索规则

物联网安全保障服务

服务领取

接口名称	接口功能
ServiceCreateCustomer	创建用户信息
ServiceGetCustomerList	获取用户信息列表
ServiceCreateOrder	创建订单
ServiceGetOrderList	查询订单列表
ServiceGetInsurancePageURL	获取保险领取页面地址

设备与用户绑定关系推送

接口名称	接口功能
ServiceBindDevice	设备与用户绑定关系推送

设备数据推送

接口名称	接口功能
ServiceReportDeviceData	用于设备数据上报
ServiceReportDeviceState	用于设备状态上报

版权所有：腾讯云计算（北京）有限责任公司

设备管理
查看设备详情

最近更新时间：2023-07-19 17:34:51

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceDescribeDevice）用于查看设备详情。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceDescribeDevice。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同。

版权所有：腾讯云计算（北京）有限责任公司

Device	Object of DeviceInfo	返回数据。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceDescribeDevice",
    "ProductId": "22F****I7O",
    "DeviceName": "light1"
}
```

输出示例：成功

```
{
    "Device": {
        "AppKey": "",
        "AppSKey": "",
        "CreateTime": 0,
        "DevAddr": "",
        "DevEUI": "",
        "DeviceCert": "",
        "DeviceName": "light1",
        "DevicePsk": "DevicePsk",
        "FirstOnlineTime": 0,
        "LogLevel": 0,
        "LoginTime": 0,
        "NwkSKey": "",
        "Status": 1,
        "Version": "1.0"
    },
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
```

```
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

获取设备列表

最近更新时间：2023-07-19 17:34:51

1. 接口描述

接口请求域名： `iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceGetDeviceList）用于获取设备列表。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceGetDeviceList。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
Offset	Int	是	分页偏移，0起始。
Limit	Int	是	单次拉取数量。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同。

Devices	Array of DeviceInfo	返回数据。
Total	Int	产品下的设备总数。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceGetDeviceList",
    "ProductId":"22****II7O",
    "Offset": 0,
    "Limit": 10
}
```

输出示例：成功

```
{
    "Devices": {
        "AppKey": "",
        "AppSKey": "",
        "CreateTime": 0,
        "DevAddr": "",
        "DevEUI": "",
        "DeviceCert": "",
        "DeviceName": "light1",
        "DevicePsk": "DevicePsk",
        "FirstOnlineTime": 0,
        "LogLevel": 0,
        "LoginTime": 0,
        "NwkSKey": "",
        "Status": 1,
        "Version": "1.0"
```

```
    },
    "Total": 1,
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

同步调用设备行为

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceCallDeviceActionSync）用于同步调用设备行为。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceCallDeviceActionSync。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。
ActionId	String	是	产品数据模板中行为功能的标识符，由开发者自行根据设备的应用场景定义。
InputParams	String	否	输入参数。

3. 输出参数

名称	类型	描述

RequestId	String	公共参数，唯一请求 ID，与入参相同。
ClientToken	String	调用 ID。
OutputParams	String	输出参数。
Status	String	返回状态。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceCallDeviceActionSync",
    "ProductId":"22****II7O",
    "DeviceName":"light1",
    "ActionId": "echo",
    "InputParams": "{\"message\":\"hello.\"}"
 }
```

输出示例：成功

```
{
    "Response": {
        "ClientToken": "45016a4****74abcb05e25790b916dc3",
        "OutputParams": "{\"message\":\"hello.\"}",
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Status": "succ"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

版权所有：腾讯云计算（北京）有限责任公司

异步调用设备行为

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceCallDeviceActionAsync）用于异步调用设备行为。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceCallDeviceActionAsync。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。
ActionId	String	是	产品数据模板中行为功能的标识符，由开发者自行根据设备的应用场景定义。
InputParams	String	否	输入参数。

3. 输出参数

名称	类型	描述

RequestId	String	公共参数，唯一请求 ID，与入参相同。
ClientToken	String	调用 ID。
Status	String	返回状态。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceCallDeviceActionAsync",
    "ProductId": "22****II7O",
    "DeviceName": "light1",
    "ActionId": "echo",
    "InputParams": "{\"message\":\"hello.\"}"
}
```

输出示例：成功

```
{
    "Response": {
        "ClientToken": "45016a453****abcb05e25790b916dc3",
        "Status": "succ",
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述

InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

设备远程控制

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceControlDeviceData）用于设置控制设备的属性数据。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceControlDeviceData。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。
Data	String	是	控制设备报文。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同。

Data	String	返回数据。
Result	String	JSON 字符串， 返回下发控制的结果信息，Sent = 1 表示设备已经在线并且订阅了控制下发的 mqtt topic。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceControlDeviceData",
    "ProductId": "22****II7O",
    "DeviceName": "light1",
    "Data": "{\"light_switch\":0}"
}
```

输出示例：成功

```
{
    "Response": {
        "Data": "{\"light_switch\":0}",
        "Result": "{\"Sent\":1,\"pushResult\":0}",
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。

ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

数据查询
获取设备物模型数据

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceDescribeDeviceData）用于获取设备物模型数据。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceDescribeDeviceData。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。

Data	String	返回数据。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceDescribeDeviceData",
    "ProductId": "22****II7O",
    "DeviceName": "light1"
}
```

输出示例：成功

```
{
    "Response": {
        "Data": "{\"brightness\":
{\"Value\":8,\"LastUpdate\":1552904784552},\"color\":
{\"Value\":1,\"LastUpdate\":1552904784552},\"light_switch\":
{\"Value\":0,\"LastUpdate\":1552904784552},\"name\":
{\"Value\":\"light1\",\"LastUpdate\":1552904784552}}",
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。

InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

获取设备物模型历史数据

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名： `iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceDescribeDeviceDataHistory）用于获取设备物模型历史数据。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceDescribeDeviceDataHistory。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。
FieldName	String	是	查询的属性名称。
Context	String	否	翻页游标，首次查询时，非必填。
MinTime	Int	是	开始时间，毫秒时间戳。
MaxTi	Int	是	结束时间，毫秒时间戳。

版权所有：腾讯云计算（北京）有限责任公司

me			
Limit	Int	是	单页数据量。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Context	String	翻页游标。
FieldName	String	查询的属性名称。
Listover	Bool	数据是否查询完成。 • true：表示未完成可继续读取。 • false：表示已全部读取完成。
Results	Array of DataHistoryItem	返回数据。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceDescribeDeviceDataHistory",
    "ProductId": "22****II7O",
    "DeviceName": "light1",
    "MinTime": 1560949017000,
    "MaxTime": 1561035417000,
    "Limit": 20,
    "FieldName": "brightness"
}
```

输出示例：成功

```
{
    "Response": {
        "Context": "abc",
        "FieldName": "brightness",
        "Listover": false,
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Results": [{
            "Time": "1560949025494",
            "Value": "77"
        }]
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

版权所有：腾讯云计算（北京）有限责任公司

获取设备的历史事件

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceListEventHistory）用于获取设备的历史事件。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceListEventHistory。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。
DeviceName	String	是	设备名。
Type	String	是	搜索的事件类型。 • alert：表示告警。 • fault：表示故障。 • info：表示信息。 • 为空则表示查询上述所有类型事件。
Conte	Strin	否	翻页游标，首次查询时，非必填。

xt	g		
Start Time	Int	是	起始时间（Unix 时间戳，秒级），为0表示当前时间 - 24h。
EndTime	Int	是	结束时间（Unix 时间戳，秒级），为0表示当前时间。
Size	Int	是	单页数据量。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Context	String	翻页游标。
Total	Int	搜索结果数量。
Listover	Bool	数据是否查询完成。 • true：表示未完成可继续读取。 • false：表示已全部读取完成。
EventHistory	Array of EventHistoryItem	返回数据。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceListEventHistory",
    "Context": "",
    "Type":"",
```

```
    "ProductId": "QD****LBNB",
    "DeviceName": "dev01",
    "StartTime": 1560355200,
    "EndTime": 1560417460,
    "Size": 20
}
```

输出示例：成功


```
{
    "Listover": true,
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "EventHistory": [{
        "TimeStamp": 1561619346000,
        "ProductId": "QD****LBNB",
        "DeviceName": "dev01",
        "EventId": "low_voltage",
        "Type": "alert",
        "Data": "{\"voltage\": 3.487172}"
    }],
    "Total": 1,
    "Context":
"DnF1ZXJ5VGhlbkZldGNoCgAAAAAABO3FkI2SzRCVTJYUjZxUVRnbUJRaUJzc0EAAAAAAAA
TuBZCNks0QlUyWFI2cVFUZ21CUWlCc3NBAAAAAAAAE7kWQjZLNEJVMlhSNnFRVGdtQlFpQnN
zQQAAAAAAABO6FkI2SzRCVTJYUjZxUVRnbUJRaUJzc0EAAAAAAAATuxZCNks0QlUyWFI2cVF
UZ21CUWlCc3NBAAAAAAAAE7wWQjZLNEJVMlhSNnFRVGdtQlFpQnNzQQAAAAAAABO9FkI2SzR
CVTJYUjZxUVRnbUJRaUJzc0EAAAAAAAATvhZCNks0QlUyWFI2cVFUZ21CUWlCc3NBAAAAAAA
AE78WQjZLNEJVMlhSNnFRVGdtQlFpQnNzQQAAAAAAABPAFkI2SzRCVTJYUjZxUVRnbUJRaUJ
zc0E="
}
```


5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。

InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

数据模板

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceDescribeModelDefinition）用于查询产品数据模板。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceDescribeModelDefinition。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
ProductId	String	是	产品 ID。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同。
Model	Object of ProductModelDefinition	返回数据。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceDescribeModelDefinition",
    "ProductId": "22F****I7O"
 }
```

输出示例：成功

```
{
    "Response": {
        "Model": {
            "CategoryModel": "{\"version\":\"1.0\",\"profile\":
{\"productID\":\"*\"},\"properties\":[],\"events\":[]}",
            "CreateTime": 1556588325,
            "ModelDefine": "{\"version\":\"1.0\",\"profile\":
{\"ProductId\":\"WTJTZX0MF2\",\"CategoryId\":\"31\"},\"properties\":
[{\"id\":\"power_switch\",\"name\":\"电灯开关\",\"desc\":\"控制电灯开灭
\",\"required\":true,\"mode\":\"rw\",\"define\":
{\"type\":\"bool\",\"mapping\":{\"0\":\"关\",\"1\":\"开\"}}},
{\"id\":\"color\",\"name\":\"颜色\",\"desc\":\"灯光颜色
\",\"mode\":\"rw\",\"define\":{\"type\":\"enum\",\"mapping\":
{\"0\":\"Red\",\"1\":\"Green\",\"2\":\"Blue\"}}},
{\"id\":\"brightness\",\"name\":\"亮度\",\"desc\":\"灯光亮度
\",\"mode\":\"rw\",\"define\":
{\"type\":\"int\",\"unit\":\"%\",\"step\":\"1\",\"min\":\"0\",\"max\":\"
100\",\"start\":\"1\"}},{\"id\":\"name\",\"name\":\"灯位置名称
\",\"desc\":\"灯位置名称：书房、客厅等
\",\"mode\":\"rw\",\"required\":false,\"define\":
{\"type\":\"string\",\"min\":\"0\",\"max\":\"64\"}}],\"events\":
[{\"id\":\"status_report\",\"name\":\"DeviceStatus\",\"desc\":\"Report
the device status\",\"type\":\"info\",\"required\":false,\"params\":
```

```
[{\"id\":\"status\",\"name\":\"running_state\",\"desc\":\"Report current
device running state\",\"define\":{\"type\":\"bool\",\"mapping\":
{\"0\":\"normal\",\"1\":\"fault\"}}},
{\"id\":\"message\",\"name\":\"Message\",\"desc\":\"Some extra
message\",\"define\":
{\"type\":\"string\",\"min\":\"0\",\"max\":\"64\"}}]},
{\"id\":\"low_voltage\",\"name\":\"LowVoltage\",\"desc\":\"Alert for
device voltage is
low\",\"type\":\"alert\",\"required\":false,\"params\":
[{\"id\":\"voltage\",\"name\":\"Voltage\",\"desc\":\"Current
voltage\",\"define\":
{\"type\":\"float\",\"unit\":\"V\",\"step\":\"1\",\"min\":\"0.0\",\"max\
":\"24.0\",\"start\":\"1\"}}]},
{\"id\":\"hardware_fault\",\"name\":\"Hardware_fault\",\"desc\":\"Report
hardware fault\",\"type\":\"fault\",\"required\":false,\"params\":
[{\"id\":\"name\",\"name\":\"Name\",\"desc\":\"Name like: memory,tf
card, censors ...\",\"define\":
{\"type\":\"string\",\"min\":\"0\",\"max\":\"64\"}},
{\"id\":\"error_code\",\"name\":\"Error_Code\",\"desc\":\"Error code for
fault\",\"define\":
{\"type\":\"int\",\"unit\":\"\",\"step\":\"1\",\"min\":\"0\",\"max\":\"2
000\",\"start\":\"1\"}}]}],\"actions\":
[{\"id\":\"echo\",\"name\":\"echo\",\"desc\":\"\",\"input\":
[{\"id\":\"message\",\"name\":\"message\",\"define\":
{\"type\":\"string\",\"min\":\"0\",\"max\":\"64\"}}],\"output\":
[{\"id\":\"message\",\"name\":\"message\",\"define\":
{\"type\":\"string\",\"min\":\"0\",\"max\":\"64\"}}],\"required\":false}
]}",
            "NetTypeModel": "",
            "ProductId": "22F****I70",
            "UpdateTime": 1571855825
        },
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```


5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。

InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

物联网安全保障服务
服务领取
创建用户信息

最近更新时间：2023-07-19 17:34:52

1. 接口描述

接口请求域名： `iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceCreateCustomer）用于创建用户信息。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceCreateCustomer。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。

CustomerId	String	是	厂商应用端的唯一用户 ID，英文（a－z，A－Z）、数字（0－9）、下划线的组合（_），最多不超过32个字符。
CountryCode	String	是	用户手机国家码。
PhoneNumber	String	是	用户手机号。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Status	String	结果，OK 表示成功，失败则不返回此字段。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceCreateCustomer",
    "CustomerId": "Customer100001",
    "CountryCode": "86",
    "PhoneNumber": "13900000000"
}
```

输出示例：成功

```
{
    "Response": {
        "RequestId": "f92406b3-****-bc43-45e3d794bb68",
        "Status": "OK"
    }
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
        "RequestId": "f92406b3-****-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。
invalid CustomerId	CustomerId 格式错误。
invalid Email	Email 格式错误。
invalid Phone	Phone 或者 CountryCode 格式错误。

版权所有：腾讯云计算（北京）有限责任公司

获取用户信息列表

最近更新时间：2023-07-19 17:34:53

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceGetCustomerList）用于获取用户信息列表。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceGetCustomerList。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
CustomerId	String	否	厂商应用端的唯一用户 ID，可选参数，未指定则查询所有，指定则根据输入内容查询指定用户。
CountryCode	String	否	国家码，可选参数，未指定则查询所有，指定则根据输入内容查询指定用户。
PhoneNumber	String	否	手机号，可选参数，未指定则查询所有，指定则根据输入内容查询指定用户。
Offset	Int	否	分页偏移，0起始。
Limit	Int	否	单页数据量。

3. 输出参数

名称	类型	描述

RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Customers	Array of CustomerInfo	用户信息。
Total	Int	总用户数。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceGetCustomerList",
    "CustomerId": "Customer100001",
    "Offset": 0,
    "Limit": 10
}
```

输出示例：成功

```
{
    "Response": {
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Customers": [{
            "CustomerId": "Customer100001",
            "CountryCode": "86",
            "PhoneNumber": "13900000000"
        }],
        "Total": 1
    }
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。
invalid CustomerId	CustomerId 格式错误。
invalid Email	Email 格式错误。
invalid Phone	Phone 或者 CountryCode 格式错误。
invalid CustomerId or invalid ServiceAppKey	CustomerId 或 ServiceAppKey 不存在。

版权所有：腾讯云计算（北京）有限责任公司

创建订单

最近更新时间：2023-07-19 17:34:53

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceCreateOrder）用于创建订单。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceCreateOrder。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
CustomerId	String	是	厂商应用端的唯一用户 ID。

OrderNo	String	是	厂商应用端的唯一订单号。
ItemId	String	是	订单下的服务类型。 • FireInsurance2020：物联网火灾安全保障服务-正式。 • D-FireInsurance2020：物联网火灾安全保障服务-测试。 • CarInsurance2021：物联网两轮车安全保障服务-正式。 • D-CarInsurance2021：物联网两轮车安全保障服务-测试。 说明：正式单会消耗授权数，服务发布后才可选用；测试单不会消耗授权数，在保险领取页面领取的是无效保单，供调试使用，服务发布前后均可用。
ItemVNO	String	是	订单下的服务版本。 以下为物联网火灾安全保障服务版本： • FireBusiness：商业版。 • FireMER20W：商户企业园区版20w。 • FireMER50W：商户企业园区版50w。 • FireMER100W：商户企业园区版100w。 • FirePRO20W ：生产园区版20w。 • FirePRO50W：生产园区版50w。 • FirePRO100W：生产园区版100w。 以下为物联网两轮车安全保障服务版本： • AutoBase：基础版。 • AutoUpgrade：升级版。 • AutoVIP：无忧版。
ItemNum	Int	是	订单下的服务数量。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Status	String	请求结果，成功时返回此字段，返回值为 OK。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceCreateOrder",
    "CustomerId": "Customer100001",
    "OrderNo": "Order100001",
    "ItemId": "FireInsurance2020",
    "ItemNum": 2
    "ItemVNO": "FireBusiness"
}
```

输出示例：成功

```
{
    "Response": {
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Status": "OK"
    }
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

查询订单列表

最近更新时间：2023-07-19 17:34:53

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceGetOrderList）用于查询订单列表。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceGetOrderList。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
CustomerId	String	否	厂商应用端的唯一用户 ID。
OrderNo	String	否	厂商应用端的唯一订单号。
Offset	Int	否	分页偏移，0起始。
Limit	Int	否	单页数据量。

3. 输出参数

名称	类型	描述

RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Orders	Array of OrderItem	订单列表，成功时返回。
Total	Int	总订单数，成功时返回。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceGetOrderList",
    "CustomerId": "Customer100001",
    "Offset": 0,
    "Limit": 10

}
```

输出示例：成功

```
{
    "Response": {
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Orders": [{
            "CustomerId": "Customer100001",
            "OrderNo": "Order100001",
            "ItemId": "FireInsurance2020",
            "ItemNum": 2,
            "ItemVNO": "FireBusiness"
        }],
        "Total": 1
    }
```

```
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

获取保险领取页面

最近更新时间：2023-07-19 17:34:53

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceGetInsurancePageURL）用于获取保险领取页面地址。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceGetInsurancePageURL。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
CustomerId	String	是	厂商应用端的唯一用户 ID。
ItemId	String	是	订单下的服务类型。 • FireInsurance2020：物联网火灾安全保障服务-正式 。 • D-FireInsurance2020：物联网火灾安全保障服务-测试。 • CarInsurance2021：物联网两轮车安全保障服务-正式。 • D-CarInsurance2021：物联网两轮车安全保障服务-测试。 说明：正式单会消耗授权数，服务发布后才可选用；测试单不会消耗授权数，在保险领取页面领取的 是无效保单，供调试使用，服务发布前后均可用。

ItemVNO	String	是	订单下的服务版本。 以下为物联网火灾安全保障服务版本： • FireBusiness：商业版。 • FireMER20W：商户企业园区版20w。 • FireMER50W：商户企业园区版50w。 • FireMER100W：商户企业园区版100w。 • FirePRO20W ：生产园区版20w。 • FirePRO50W：生产园区版50w。 • FirePRO100W：生产园区版100w。 以下为物联网两轮车安全保障服务版本： • AutoBase：基础版。 • AutoUpgrade：升级版。 • AutoVIP：无忧版。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
URL	String	领取页面 H5 地址。 注意： • 当订单为测试单或者订单为物联网火灾安全保障服务正式单-商业版时，请使用 H5 地址作为保险领取页。 • 当订单为其他类型的正式单时，请使用 MpAppId 和 MpAppPath 作为保险领取页，才可以正常进入投保流程。
MpAppId	String	领取页面-腾讯连连小程序 APPID。
MpAppPath	String	领取页面-腾讯连连小程序跳转路径。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
```

```
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceGetInsurancePageURL",
    "CustomerId": "Customer100001",
    "ItemId": "FireInsurance2020",
    "ItemVNO": "FireMER20W"

}
```

输出示例：成功

```
{
    "Response": {
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "URL": "https://www.example.com/path/to/h5homepage?
CustomerId=abc&ServiceAppKey=xyz&ItemId=FireInsurance2020&ItemVNO=FireBu
siness",
        "MpAppId": "WX123",
        "MpAppPath": "/IOT/JUMPPATH"

    }
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述

InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

设备与用户绑定关系推送

最近更新时间：2023-07-19 17:34:53

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceBindDevice）用于推送设备与用户的绑定关系。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceBindDevice。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
CustomerId	String	是	厂商应用端的唯一用户 ID。
DeviceInfos	Array of ServiceDeviceInfo	是	设备信息列表。 支持推送用户与单个或批量设备的绑定关系。当用户对应多个厂商设备编号时，必须采用批量推送方式。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。

Status	String	请求结果，成功时返回此字段，返回值为 OK。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceBindDevice",
    "CustomerId": "CustomerId",
    "DeviceInfos":
[{"CustomerDeviceNo":"123","DeviceId":"ProductId/DeviceName","ProductNam
e":"***"}]
}
```

输出示例：成功

```
{
  "Response": {
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
     "Status": "OK"
  }
}
```

输出示例：失败

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue",
      "Message": "Signature 无效"
    },
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
  }
```

```
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

设备数据推送
设备数据上报

最近更新时间：2024-09-30 10:38:01

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceReportDeviceData）用于设备数据上报。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceReportDeviceData。
RequestId	String	是	公共参数，唯一请求 ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
DeviceId	String	是	在物联网开发平台创建的设备 ID，唯一标识一个设备，由 `产品 ID + "/" + 设备名称` 组合而成，例如 `ProductABC/Device001` 。
CustomerId	String	是	厂商应用端的唯一用户 ID。
DataCategory	String	是	数据分类。 • 100：电压 • 200：温度 • 300：电流 • 310：漏电电流 • 400：功率

			• 500：烟感
DataInfo	Array of ReportDeviceData	是	上报数据信息。

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Status	String	请求结果，成功时返回此字段，返回值为 OK。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceReportDeviceData",
    "DeviceId": "ProductId/DeviceName",
    "CustomerId": "customerid",
    "DataCategory": 100,
    "DataInfo": {
        "ReportTime": 1618580291,
        "ReportInfos":[{"SubType": 1000,"Value": 1},{"SubType":
1001,"Value": 265}]
    }
}
```

输出示例：成功

```
{
    "Response": {
```

```
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Status": "OK"
    }
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

设备状态上报

最近更新时间：2023-07-19 17:34:53

1. 接口描述

接口请求域名：`iot.cloud.tencent.com/api/exploreropen/serviceapi` 。
本接口（ServiceReportDeviceState）用于设备状态上报。

2. 输入参数

名称	类型	必选	描述
Action	String	是	公共参数，本接口取值：ServiceReportDeviceState。
RequestId	String	是	公共参数，唯一请求ID，可自行生成，推荐使用 uuid。定位问题时，需提供该次请求的 RequestId。
AppKey	String	是	公共参数，物联网 SaaS 的 AppKey ，用于标识对应的物联网 SaaS。
Signature	String	是	公共参数，请求签名，需用户自行生成，用于校验请求的合法性。
Timestamp	Int64	是	公共参数，请求的 Unix 时间戳（秒级）。
Nonce	Int	是	公共参数，随机正整数，与 Timestamp 联合起来，防止重放攻击。
DeviceId	String	是	在物联网开发平台创建的设备 ID，唯一标识一个设备，由 `产品ID + "/" + 设备名称` 组合而成，例如 `ProductABC/Device001` 。

OnlineStatus	Int	是	设备状态。 • 0：离线 • 1：在线 • 2：未知

3. 输出参数

名称	类型	描述
RequestId	String	公共参数，唯一请求 ID，与入参相同，定位问题时，需提供该次请求的 RequestId。
Status	String	请求结果，成功时返回此字段，返回值为 OK。

4. 示例

输入示例

```
POST https://iot.cloud.tencent.com/api/exploreropen/serviceapi HTTP/1.1
content-type: application/json
{
    "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
    "Timestamp": 1599468019,
    "Nonce": 170533123,
    "AppKey": "AppKey",
    "Signature": "Signature",
    "Action": "ServiceReportDeviceState",
    "OnlineStatus": 1,
    "DeviceId": "ProductId/DeviceName"
}
```

输出示例：成功

```
{
    "Response": {
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68",
        "Status": "OK"
    }
}
```

输出示例：失败

```
{
    "Response": {
        "Error": {
            "Code": "InvalidParameterValue",
            "Message": "Signature 无效"
        },
        "RequestId": "f92406b3-****-4fe8-bc43-45e3d794bb68"
    }
}
```

5. 错误码

错误码	描述
InternalError	内部错误。
ErrorRequiredParamNotFound	必选参数缺失。
InvalidAction	Action 非法。
InvalidParameterValue	参数非法。
InvalidParameterValue.InvalidJSON	请求格式不是 JSON。

版权所有：腾讯云计算（北京）有限责任公司

数据结构

最近更新时间：2024-09-30 10:38:01

CustomerInfo

客户信息
被如下接口引用：ServiceGetCustomerList

名称	类型	必选	描述
CustomerId	String	是	厂商应用端的唯一用户 ID
CountryCode	String	是	国家码
PhoneNumber	String	是	手机号

DataHistoryItem

历史数据记录
被如下接口引用：ServiceDescribeDeviceDataHistory

名称	类型	描述
Time	String	毫秒时间戳
Value	String	数据取值

DeviceInfo

设备详细信息
被如下接口引用：ServiceDescribeDevice、ServiceGetDeviceList。

名称	类型	必选	描述
DeviceName	String	是	设备名
Status	Int	是	• 0：离线 • 1：在线 • 2：获取失败 • 3：未激活
DevicePsk	String	是	设备密钥，密钥加密的设备返回

版权所有：腾讯云计算（北京）有限责任公司

FirstOnlineTime	Int	否	首次上线时间
LoginTime	Int	否	最后一次上线时间
CreateTime	Int	否	设备创建时间
Version	String	否	设备固件版本
DeviceCert	String	否	设备证书
LogLevel	Int	否	日志级别
DevAddr	String	否	LoRaWAN 设备地址
AppKey	String	否	LoRaWAN 应用密钥
DevEUI	String	否	LoRaWAN 设备唯一标识
AppSKey	String	否	LoRaWAN 应用会话密钥
NwkSKey	String	否	LoRaWAN 网络会话密钥

ErrorMessage

错误信息

名称	类型	描述
Code	String	错误码
Message	String	错误信息

EventHistoryItem

设备事件的搜索结果项

被如下接口引用：ServiceListEventHistory

名称	类型	必选	描述
TimeStamp	String	否	事件的时间戳
ProductId	String	是	产品 ID
DeviceName	String	是	设备名
EventId	String	是	事件的标识符 ID

Type	String	是	事件类型。 • alert：表示告警 • fault：表示故障 • info：表示信息
Data	String	是	事件的数据

OrderItem

订单数据

被如下接口引用：ServiceGetOrderList

名称	类型	必选	描述
CustomerId	String	是	厂商应用端的唯一用户 ID
OrderNo	String	是	厂商应用端的唯一订单号
ItemId	String	是	订单下的服务类型。 • FireInsurance2020：物联网火灾安全保障服务-正式。 • D-FireInsurance2020：物联网火灾安全保障服务-测试。 • CarInsurance2021：物联网两轮车安全保障服务-正式。 • D-CarInsurance2021：物联网两轮车安全保障服务-测试。 说明：正式单会消耗授权数，服务发布后才可选用；测试单不会消耗授权数，在保险领取页面领取的是无效保单，供调试使用，服务发布前后均可用。
ItemVNO	String	是	订单下的服务版本。 以下为物联网火灾安全保障服务版本： • FireBusiness：商业版 • FireMER20W：商户企业园区版20w • FireMER50W：商户企业园区版50w • FireMER100W：商户企业园区版100w • FirePRO20W ：生产园区版20w • FirePRO50W：生产园区版50w • FirePRO100W：生产园区版100w

			以下为物联网两轮车安全保障服务版本： • AutoBase：基础版 • AutoUpgrade：升级版 • AutoVIP：无忧版
ItemNum	Int	是	订单下的服务数量

ProductModelDefinition

产品数据模板

被如下接口引用：ServiceDescribeModelDefinition

名称	类型	必选	描述
ProductId	String	是	产品 ID
ModelDefine	String	是	模型定义
CategoryModel	String	是	产品所属分类的模型快照（产品创建时刻）
UpdateTime	Int	否	更新时间，秒级时间戳
CreateTime	Int	否	创建时间，秒级时间戳

ReportDataItem

上报数据信息元数据

被如下接口引用：ServiceReportDeviceData

名称	类型	必选	描述
SubType	Int64	是	子类型。 • 1000：路数 • 1001：电压 • 1002：温度 • 1003：电流 • 1004：额定电流 • 1005：漏电电流 • 1006：功率 • 1007：烟感
Value	Int64 or Float64	是	值

ReportDeviceData

上报数据信息

被如下接口引用：ServiceReportDeviceData

名称	类型	必选	描述
ReportInfos	Array of ReportDataItem	是	上报信息列表
ReportTime	Int64	是	上报时间 秒级时间戳

ServiceDeviceInfo

名称	类型	必选	描述
CustomerDeviceNo	String	是	用户对应的厂商设备编号
DeviceId	String	是	在物联网开发平台创建的设备 ID，唯一标识一个设备，由 产品 ID + "/" + 设备名称 组合而成，例如 ProductABC/Device001。
ProductName	String	是	在物联网开发平台创建的产品名称

版权所有：腾讯云计算（北京）有限责任公司