

HTTP Post Protocol User Guide
For IP Media Device

Version 2.1.0

2025-11

Contents

HTTP POST PROTOCOL USER GUIDE	I
FOR IP MEDIA DEVICE	I
VERSION 2.1.0	I
2025-10	I
DOCUMENT HISTORY	II
1 OVERVIEW	1
1.1 PREFACE	1
1.2 DATA PUSH DESCRIPTION	1
1.2.1 Push by subscription	1
1.2.2 Push by server address	4
1.3 COMMOD SUMMARY	4
2 COMMANDS	6
2.1 SETSUBSCRIBE	6
2.2 SETRENEW	9
2.3 SETUNSUBSCRIBE	11
2.4 SUBSCRIBETIMEOUT	13
2.5 GETPULLMESSAGES	14
2.6 SENDALARMDATA	19
2.6.1 General Intelligent Event	19
2.6.2 Statistical Event	31
2.6.3 Temperature Measurement	38
2.6.4 Fire	42
2.6.5 Face Detection	46
2.6.6 Face Match	48
2.6.7 License Plate Detection	51
2.7 SENDALARMSTATUS	54
2.7.1 Alarm Status	54
2.8 SENDTRAJECTORYDATA	62
2.8.1 Target Trajectory	62
2.9 SENDKEEPALIVE	64
2.9.1 Keepalive	64
ANNEX A	1
A.1 TYPE DEFINITION	1
A.1.1 smartTypeE	1
A.1.2 messageTypeE	2
A.1.3 targetTypeE	2
A.1.4 ruleShapeTypeE	2
A.1.5 tripwireDirectionE	2
A.1.6 boundaryTypeE	3

A.2 DEFINITION OF STRUCTURE.....	3
A.2.1 <i>deviceInfo</i>	3
A.2.2 <i>sourceDataInfo</i>	3
A.2.3 <i>associationSourceDataInfo</i>	4
A.2.4 <i>targetImageData</i>	4
A.2.5 <i>rect</i>	4
A.2.6 <i>directionLine</i>	5
A.2.7 <i>pointGroup</i>	5
A.2.8 <i>point</i>	5
A.2.9 <i>personAttribute</i>	6
A.2.10 <i>carAttribute</i>	6
A.2.11 <i>motorAttribute</i>	6
A.2.12 <i>faceAttribute</i>	6
A.2.13 <i>licensePlateAttribute</i>	7
A.2.14 <i>statisticalData</i>	7
A.2.15 <i>targetListInfo</i>	8
A.2.16 <i>faceListInfo</i>	9
A.2.17 <i>licensePlateListInfo</i>	9
A.2.18 <i>eventInfo</i>	10
A.2.19 <i>associationEventInfo</i>	10
A.2.20 <i>temperatureEventInfo</i>	11
A.2.21 <i>associationTemperatureEventInfo</i>	12
A.2.22 <i>licensePlateMatchInfo</i>	12
A.2.23 <i>targetAdditionalInfo</i>	13
A.2.23 <i>personInfo</i>	13
A.2.24 <i>personPicture</i>	13
A.2.22 <i>faceInfo</i>	14
A.2.23 <i>faceMatchInfo</i>	14
A.3 DEFINITION OF TARGET ATTRIBUTE.....	14
A.3.1 <i>person</i>	14
A.3.2 <i>car</i>	16
A.3.3 <i>motor</i>	17
A.3.4 <i>face</i>	17
A.3.5 <i>license plate</i>	17
ANNEX B	18
B.1 CHANGE LOG	18

1 Overview

1.1 Preface

This document describes the protocol for IP media devices to push event and alarm data to third-party servers.

1.2 Data Push Description

We support two ways of pushing data: push by subscription and push by server address.

- Push by subscription:

The client(or third-party server) establishes a persistent connection with the server and then sends a subscription request to the device. The device pushes event and alarm data to the client based on the HTTP POST protocol according to the subscription request.

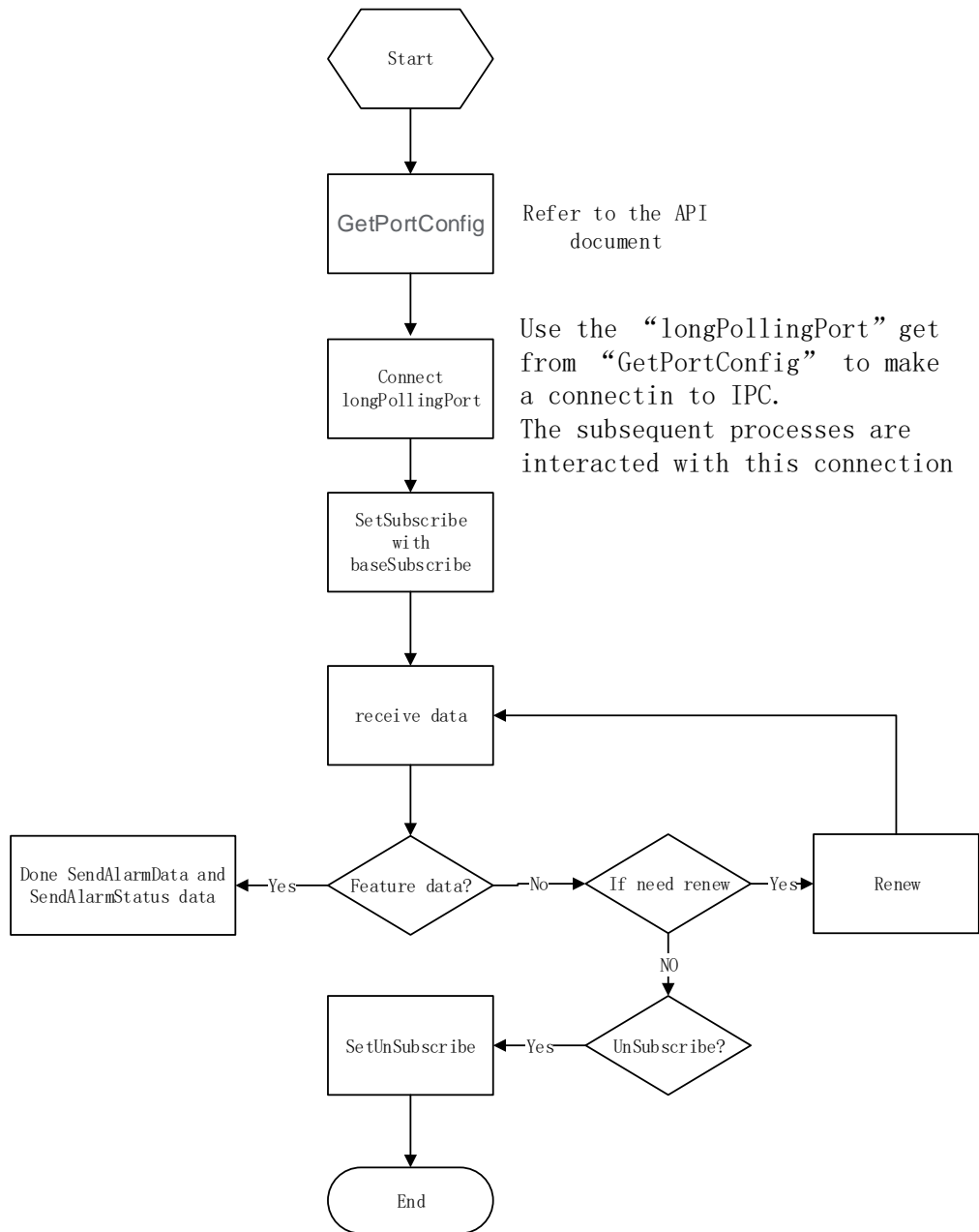
- Push by server address:

The device directly pushes event and alarm data to the specified server address (based on HTTP POST protocol).

1.2.1 Push by subscription

The system supports two subscription methods: Base Subscribe, RealTime Subscribe. We recommend using the "baseSubscribe" method.

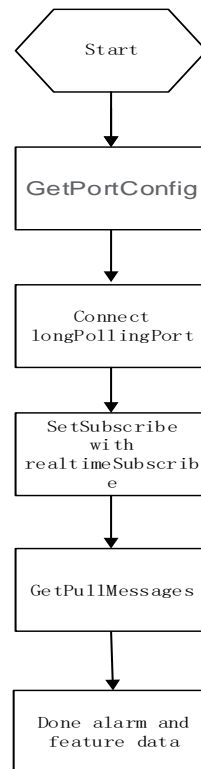
I. Base Subscribe(Recommend)



Notes:

1. The third-party server, as a TCP client, initiates a subscription to the IP Media Device, with a subscription type of "baseSubscribe". Refer to the description of the interface "SetSubscribe" for details.
2. After subscription, The IP Media Device will push alarm status and event data to the client.
3. Subscription has a validity period limit. If you need to continue subscribing, you need to periodically send "SetRenew" to extend the validity period of the subscription.
4. The client can end the subscription through 'SetUnSubscribe'.

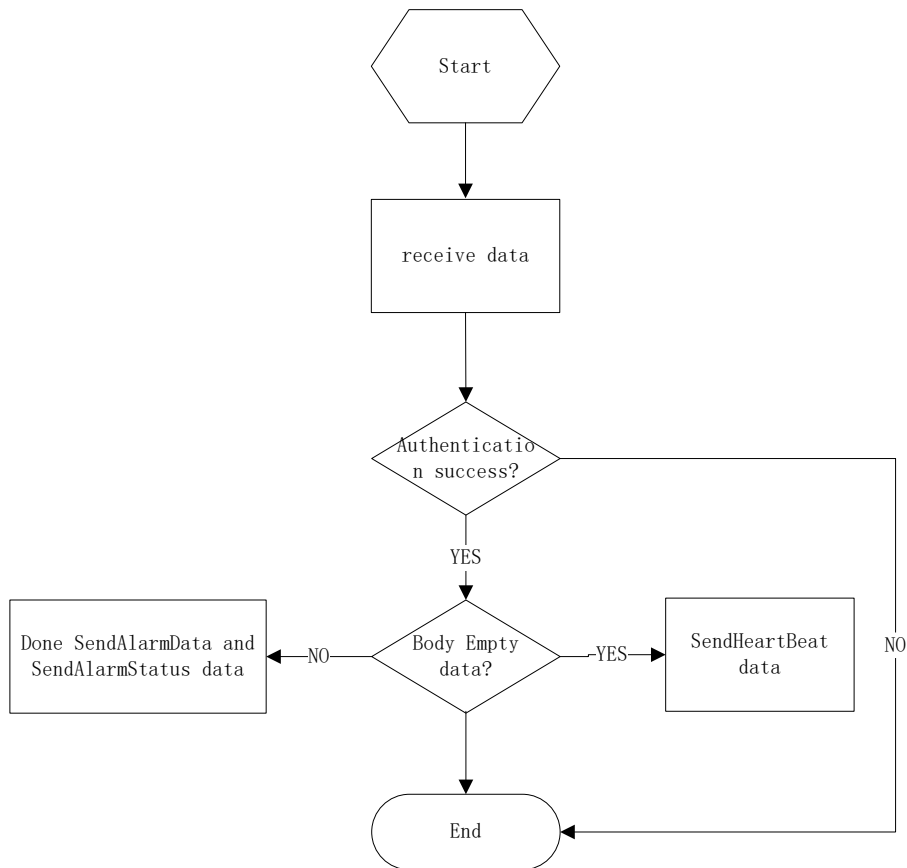
II. Realtime Subscribe



Notes:

1. The third-party server, as a TCP client, initiates a subscription to the IP Media Device, with a subscription type of "realtimeSubscribe". Refer to the description of the interface "**SetSubscribe**" for details.
2. After subscription, IP Media Devices do not push data to clients, but require clients to periodically send "**GetPullMessages**" to obtain alarm status and event data. Specifically, if there is no alarm status or event data, GetPullMessages will block and wait for a specific duration specified in "GetPullMessages".

1.2.2 Push by server address



Notes:

1. No subscription required, IP Media Device directly pushes alarm status and event data to specific third-party servers.
2. The device acts as a client and establishes a connection to the specified server. The system supports both persistent connection and short connection. For persistent connection, heartbeat messages are required to maintain the connection. For short connection, after sending the data, the TCP connection will be disconnected.
3. This method supports custom push commands. If there is no customization, the system default command will be used.

1.3 Commod Summary

Command	Direction	Note	Push by Subscription	Push by server address
SetSubscribe	User ---> Device	Subscribe to the device.	√	×

SetRenew	User ---> Device	Renew the subscription.	√	×
SetUnSubscribe	User ---> Device	Unsubscribe to the device.	√	×
SendSubscribeTime Out	Device ---> User	Indicate subscription timeout	√	×
GetPullMessage	User ---> Device	To get the alarm status or data from the device.	√	×
SendAlarmData	Device ---> User	Including the Alarm Data.	√	√
SendAlarmStatus	Device ---> User	Including the Alarm Status.	√	√
SendTrajectoryData	Device ---> User	Including the trajectory data.	√	√
SendKeepalive	Device ---> User	Including device information.	×	√

Note: "√": Indicates that this message is applicable to this scenario. "×": indicates that this scenario is not applicable.

2 commands

2.1 SetSubscribe

SetSubscribe	
Description	Subscribe to smart alerts or feature data functions, You can subscribe to different smart events multiple times. According to the interface "GetPortConfig", For field "longPollingPort" to Connect this port, Data is sent and received through this port.
Typical URL	POST or GET http://<host>[:port]/SetSubscribe
Entity Data	<config> of Entity Data
Successful Response	<config> of Successful Response
Direction	User ---> Device
Applicable scenarios	Push by subscription
Applicable products	IPC
<config> of Entity Data XML Block: <pre><config version="" xmlns="http://www.ipc.com/ver10"> <initTermTime> <!--Optional; uint32,Range[0,86400], Unit second. Initial termination time. 0: permanence, No renew is required, non-zero: Subscribe to the time. Default:0--> </initTermTime> <subscribeFlag> <!--Required; subscribeTypes,value in {baseSubscribe, realtimeSubscribe}, Unit second. Initial termination time. 0: permanence, No renew is required, non-zero: Subscribe to the time. Default:0.--> </subscribeFlag> </config></pre>	

```

</subscribeFlag>
<subscribeInfo><!--Required; List of subscribed content.-->
  <smartSubscribeList><!--Optional; List of smart subscribe.-->
    <item>
      <smartType>
        <!--Required;smartTypeE, Types of smart events that require subscription. Refer
        to Appendix A.1.-->
      </smartType>
      <subscribeRelation>
        <!--Required; subscribeRelation, definition of subscription content.value in
        {alarm, featureResult, alarmAndFeatureResult}.Refer to 'Tips' for more
        details.-->
      </subscribeRelation>
    </item>
  </smartSubscribeList>
  <generalSubscribe><!--Optional;general subscribe.-->
    <subscribeRelation><!--Required; value in {trajectory}.--></subscribeRelation>
  </generalSubscribe>
</subscribeInfo>
</config>

```

<config> of Successful Response XML Block:

```

<config version="" xmlns="http://www.ipc.com/ver10">
  <serverAddress>
    <!--Required;string,The subscription identifier assigned by the system, used for
    SetRenew/SetUnSubscribe/GetPullMessages.-->
  </serverAddress>
  <currentTime>
    <!--Required; uint64,Timestamp of successful subscription.-->
  </currentTime>
  <terminationTime>
    <!--Required; uint64,Timestamp of subscription expiration.-->
  </terminationTime>
  <timeout>

```

<!--Dependent;uint32,Valid when the <subscribeFlag> in Entity Data is set to "realtimeSubscribe". Indicate the time interval for the client to call 'GetPullMessages'. -->

</timeout>

</config>

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
```

```
  <initTermTime>3600</initTermTime>
```

```
  <subscribeFlag>baseSubscribe</subscribeFlag>
```

```
  <subscribeInfo>
```

```
    <smartSubscribeList>
```

```
      <item>
```

```
        <smartType>motion</smartType>
```

```
        <subscribeRelation>alarm</subscribeRelation>
```

```
      </item>
```

```
      <item>
```

```
        <smartType>objectStatusChange</smartType>
```

```
        <subscribeRelation>alarm</subscribeRelation>
```

```
      </item>
```

```
      <item>
```

```
        <smartType>regionEntrance</smartType>
```

```
        <subscribeRelation>alarmAndFeatureResult</subscribeRelation>
```

```
      </item>
```

```
    </smartSubscribeList>
```

```
    <generalSubscribe>
```

```
      <subscribeRelation>trajectory</subscribeRelation>
```

```
    </generalSubscribe>
```

```
  </subscribeInfo>
```

```
</config>
```

Example of Successful Response:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```

<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <serverAddress>
    <![CDATA[http://192.168.1.123:8080/IPC/event/subscription_0]]>
  </serverAddress>
  <currentTime>1506310717</currentTime>
  <terminationTime>1537846717</terminationTime>
</config>

```

[Tips]:

- The value definition of "subscribeRelation":
 - alarm**: alarm status.
 - featureResult**: smart feature result data.
 - alarmAndFeatureResult**: alarm status and smart feature result data.
- The value definition of "subscribeTypes":
 - baseSubscribe**: The device pushes messages to the client.
 - realtimeSubscribe**: Client pulls message from device (call GetPullMessages).
- The client can call **SetSubscribe** multiple times to add subscriptions, and the system will merge them. "SubscribeFlag" uses the value of the first subscription.

2.2 SetRenew

SetRenew	
Description	Renew subscription time.
Typical URL	/SetRenew
Entity Data	<config> of Entity Data
Successful Response	<config> of Successful Response
Direction	User ---> Device
Applicable scenarios	Push by subscription
Applicable products	IPC
<config> of Entity Data XML Block: <config version="" xmlns="http://www.ipc.com/ver10">	

```
<serverAddress>
```

```
    <!--Required;string,The subscription identifier. Retrieve from the successful response of calling  
    'Setsubscribe'.-->
```

```
</serverAddress>
```

```
<renewTime>
```

```
    <!--Required; uint32,Range[0,86400], Unit second. Refresh duration of subscription.-->
```

```
</renewTime>
```

```
</config>
```

<config> of Successful Response XML Block:

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
```

```
    <currentTime>
```

```
        <!--Required; uint64,Timestamp of successful subscription.-->
```

```
    </currentTime>
```

```
    <terminationTime>
```

```
        <!--Required; uint64,Timestamp of subscription expiration.-->
```

```
    </terminationTime>
```

```
</config>
```

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
```

```
    <serverAddress>
```

```
        <![CDATA[http://192.168.1.123:8080/IPC/event/subscription_0]]>
```

```
    </serverAddress>
```

```
    <renewTime>60</renewTime>
```

```
</config>
```

<config> of Successful Response XML Block:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
```

```
    <currentTime>1506311038</currentTime>
```

```
    <terminationTime>1506311098</terminationTime>
```

</config>

[Tips]:

2.3 SetUnSubscribe

SetUnSubscribe	
Description	To unsubscribe.
Typical URL	/SetUnSubscribe
Entity Data	<config> of Entity Data
Successful Response	<config> of Successful Response
Direction	User ---> Device
Applicable scenarios	Push by subscription
Applicable products	IPC
<config> of Entity Data XML Block: <config version="" xmlns="http://www.ipc.com/ver10"> <serverAddress> <!--Required;string,The subscription identifier. Retrieve from the successful response of calling 'Setsubscribe'.--> </serverAddress> <unsubscribeInfo><!--Required; List of unsubscrib content.--> <smartSubscribeList><!--Optional; List of smart subscribe.--> <item> <smartType> <!--Required;smartTypeE, Types of smart events that require subscription. Refer to Appendix A.1.--> </smartType> <subscribeRelation>	

<!--Required; subscribeRelation, definition of subscription content.value in {alarm, featureResult, alarmAndFeatureResult}.Refer to 'Tips' for more details.-->

</subscribeRelation>

</item>

</smartSubscribeList>

<generalSubscribe><!--Optional; general subscribe.-->

<subscribeRelation><!--Required; value in {trajectory}.--><subscribeRelation>

</generalSubscribe>

</unsubscribeInfo>

</config>

<config> of Successful Response

<config version="2.0.0" xmlns="http://www.ipc.com/ver10" status="" errorCode=""/><!--Required; Carrying operation results through status and errorCode fields.-->

Example of Entity Data:

<?xml version="1.0" encoding="UTF-8"?>

<config version="2.0.0" xmlns="http://www.ipc.com/ver10">

<serverAddress>

<![CDATA[http://192.168.1.123:8080/IPC/event/subscription_0]]>

</serverAddress>

<unsubscribeInfo>

<smartSubscribeList>

<item>

<smartType>motion</smartType>

<subscribeRelation>alarm</subscribeRelation>

</item>

<item>

<smartType>regionEntrance</smartType>

<subscribeRelation>alarmAndFeatureResult</subscribeRelation>

</item>

</smartSubscribeList>

<generalSubscribe>

<subscribeRelation>trajectory<subscribeRelation>

```
</generalSubscribe>
```

```
</unsubscribeInfo>
```

```
</config>
```

Example of Successful Response:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10" status="success" errorCode="200"/>
```

[Tips]:

2.4 SubscribeTimeOut

SubscribeTimeOut	
Description	Notify subscribers: The subscription has expired.
Typical URL	/SubscribeTimeOut
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription
Applicable products	IPC
<config> XML Block: <config version="" xmlns="http://www.ipc.com/ver10"> <serverAddress> <!--Required;string,The subscription identifier. Retrieve from the successful response of calling 'Setsubscribe'.--> </serverAddress> <currentTime> <!--Required; uint64,Timestamp of successful subscription.--> </config>	

<pre> </currentTime> <terminationTime> <!--Required; uint64, Timestamp of subscription expiration.--> </terminationTime> </config> </pre>
<p>Example of Entity Data:</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <config version="2.0.0" xmlns="http://www.ipc.com/ver10"> <serverAddress> <![CDATA[http://192.168.1.123:8080/IPC/event/subscription_0]]> </serverAddress> <currentTime>1506310717</currentTime> <terminationTime>1537846717</terminationTime> </config> </pre>
<p>[Tips]:</p>

2.5 GetPullMessages

GetPullMessages	
Description	To get pull message. Only used when the "subscribeFlag" of "SetSubscribe" set to "realtimeSubscribe".
Typical URL	/GetPullMessages
Entity Data	<config> of Entity Data
Successful Response	<config> of Successful Response
Direction	User ---> Device
Applicable scenarios	Push by subscription(Realtime Subscribe).
Applicable products	IPC

<config> of Entity Data XML Block:

```
<config version="" xmlns="http://www.ipc.com/ver10">
  <serverAddress>
    <!--Required;string,The subscription identifier. Retrieve from the successful response of calling
    'Setsubscribe'.-->
  </serverAddress>
  <timeout>
    <!--Required;uint32, If there is no data, the maximum blocking duration for calling this interface.
    Unit: seconds.-->
  </timeout>
  <messageLimit>
    <!--Required;uint32,Range[1,30], The maximum number of alarm or event data that need to be
    pulled.-->
  </messageLimit>
</config>
```

<config> of Successful Response XML Block:

```
<config version="" xmlns="http://www.ipc.com/ver10">
  <currentTime>
    <!--Required;uint64,Current timestamp.-->
  </currentTime>
  <terminationTime>
    <!--Required; uint64,Timestamp of subscription expiration.-->
  </terminationTime>
  <messageList><!--Required; list,message list. When there is no message, count is 0.-->
    <item>
      <!--Optional; Package one or more of the following messages:
      "SendEventData","SendAlarmData","SendAlarmStatus","SendTrajectoryData","SendRuleInfo".-
      ->
    </item>
  </messageList>
</config>
```

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8"?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <serverAddress><![CDATA[http://192.168.13.178:8080/TVT/event/subscription_0]]></serverAddress>
  <timeout>20</timeout>
  <messageLimit>10</messageLimit>
</config>
```

Example of Successful Response:

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <currentTime>1728444602</currentTime>
  <terminationTime>1728444632</terminationTime>
  <messageList>
    <item>
      <config version="2.0.0" xmlns="http://www.ipc.com/ver10">
        <messageType>alarmStatus</messageType>
        <deviceInfo>
          <deviceName><![CDATA[IPC]]></deviceName>
          <ip><![CDATA[10.10.10.10]]></ip>
          <mac><![CDATA[00:18:ae:34:44:56]]></mac>
        </deviceInfo>
        <currentTime>1699520826280829</currentTime>
        <alarmStatusInfo>
          <sensorAlarmIn>
            <item id="1">true</item>
          </sensorAlarmIn>
          <motionAlarm>
            <item id="1">true</item>
          </motionAlarm>
        </alarmStatusInfo>
      </config>
    </item>
    <item>
      <config version="2.0.0" xmlns="http://www.ipc.com/ver10">
```

```
<messageType>alarmData</messageType>
<deviceInfo>
  <deviceName><![CDATA[IPC]]></deviceName>
  <ip><![CDATA[10.10.10.10]]></ip>
  <mac><![CDATA[00:18:ae:34:44:56]]></mac>
  <channelId>1</channelId>
</deviceInfo>
<smartType>regionIntrusion</smartType>
<currentTime>1699520826280829</currentTime>
<eventInfo>
  <item>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <boundaryType>area</boundaryType>
    <pointGroup>
      <item>
        <x>2000</x>
        <y>1400</y>
      </item>
      <item>
        <x>8225</x>
        <y>2266</y>
      </item>
      <item>
        <x>7450</x>
        <y>7500</y>
      </item>
      <item>
        <x>2425</x>
        <y>8466</y>
      </item>
    </pointGroup>
  </item>
</eventInfo>
```

```
<rect>
  <x1>1789</x1>
  <y1>1944</y1>
  <x2>6363</x2>
  <y2>9652</y2>
</rect>
</item>
</eventInfo>
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
  </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetType>motor</targetType>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]]>
      </targetBase64Data>
    </targetImageData>
  </item>
</targetListInfo>
```

```

    </config>
  </item>
</messageList>
</config>

```

[Tips]:

2.6 SendAlarmData

2.6.1 General Intelligent Event

SendAlarmData	
Description	<p>Push the event data for general intelligent event to the subscribe server/client or specific server address when an alarm happens.</p> <p>Applicable to:</p> <ul style="list-style-type: none"> • Region Instusion • Region Entrance • Region Leave • Illegal Parking Detection • Loitering Detection • Line Crossing • Crowd Gathering Detection • Object Status Change • Video Metadata • Personnel Tailgating • Personnel Intrusion • Absence Detection
Typical URL	<ul style="list-style-type: none"> • Push by subscription& Push by server address(default): /SendAlarmData • Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.

Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.--> </deviceInfo> <smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1.--> </smartType> <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime> <eventInfo><!--Required; list,event information. Refer to Appendix A.2.--> </eventInfo> <sourceDataInfo><!--Optional; Source picture. Refer to Appendix A.2.--> </sourceDataInfo> <targetListInfo><!--Optional; list,List of target picture and attribute infomation. Refer to Appendix A.2.--> </targetListInfo> </config> </pre>	
<p>Example of Entity Data(Region Intrusion):</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <config version="2.0.0" xmlns="http://www.ipc.com/ver10"> <messageType>alarmData</messageType> <deviceInfo> <deviceName><![CDATA[IPC]]></deviceName> <ip><![CDATA[10.10.10.10]]></ip> <mac><![CDATA[00:18:ae:34:44:56]]></mac> <channelId>1</channelId> </deviceInfo> </pre>	

```
<smartType>regionIntrusion</smartType>
<currentTime>1699520826280829</currentTime>
<eventInfo>
  <item>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <boundaryType>area</boundaryType>
    <pointGroup>
      <item>
        <x>2000</x>
        <y>1400</y>
      </item>
      <item>
        <x>8225</x>
        <y>2266</y>
      </item>
      <item>
        <x>7450</x>
        <y>7500</y>
      </item>
      <item>
        <x>2425</x>
        <y>8466</y>
      </item>
    </pointGroup>
    <rect>
      <x1>1789</x1>
      <y1>1944</y1>
      <x2>6363</x2>
      <y2>9652</y2>
    </rect>
  </item>
```

```

</eventInfo>
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]>
  </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetType>motor</targetType>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]>
      </targetBase64Data>
    </targetImageData>
  </item>
</targetListInfo>
</config>

```

Example of Entity Data(Line Crossing):

```

<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>

```

```
<deviceName><![CDATA[IPC]]></deviceName>
<ip><![CDATA[10.10.10.10]]></ip>
<mac><![CDATA[00:18:ae:34:44:56]]></mac>
<channelId>1</channelId>
</deviceInfo>
<smartType>lineCrossing</smartType>
<currentTime>1699520826280829</currentTime>
<eventInfo >
  <item>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <boundaryType>tripwire</boundaryType>
    <directionLine>
      <startPoint>
        <x>975</x>
        <y>5233</y>
      </startPoint>
      <endPoint>
        <x>9950</x>
        <y>30</y>
      </endPoint>
      <direction>rightortop</direction>
    </directionLine>
    <rect>
      <x1>1789</x1>
      <y1>1944</y1>
      <x2>6363</x2>
      <y2>9652</y2>
    </rect>
  </item>
</eventInfo>
<sourceDataInfo>
```

```

<dataType>0</dataType>
<width>1280</width>
<height>720</height>
<sourceBase64Length>205168</sourceBase64Length>
<sourceBase64Data>
  <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
</sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetType>motor</targetType>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]]>
      </targetBase64Data>
    </targetImageData>
  </item>
</targetListInfo>
</config>

```

Example of Entity Data(Object Status Change):

```

<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]</deviceName>
    <ip><![CDATA[10.10.10.10]]</ip>

```

```
<mac><![CDATA[00:18:ae:34:44:56]]></mac>
<channelId>1</channelId>
</deviceInfo>
<smartType>objectStatusChange</smartType>
<currentTime>1699520826280829</currentTime>
<eventInfo>
  <item>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <boundaryType>area</boundaryType>
    <pointGroup>
      <item>
        <x>2000</x>
        <y>1400</y>
      </item>
      <item>
        <x>8225</x>
        <y>2266</y>
      </item>
      <item>
        <x>7450</x>
        <y>7500</y>
      </item>
      <item>
        <x>2425</x>
        <y>8466</y>
      </item>
    </pointGroup>
  </item>
</eventInfo>
</config>
```

Example of Entity Data(Video metadata):

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
    <channelId>1</channelId>
  </deviceInfo>
  <smartType>videoMetadata</smartType>
  <currentTime>1699520826280829</currentTime>
  <eventInfo>
    <item>
      <eventId>208</eventId>
      <targetId>8</targetId>
      <boundaryType>area</boundaryType>
      <pointGroup>
        <item>
          <x>2000</x>
          <y>1400</y>
        </item>
        <item>
          <x>8225</x>
          <y>2266</y>
        </item>
        <item>
          <x>7450</x>
          <y>7500</y>
        </item>
        <item>
          <x>2425</x>
```

```
<y>8466</y>
</item>
</pointGroup>
<rect>
  <x1>1789</x1>
  <y1>1944</y1>
  <x2>6363</x2>
  <y2>9652</y2>
</rect>
</item>
</eventInfo>
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]>
  </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetType>person</targetType>

    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]>
      </targetBase64Data>
    </targetImageData>
  </item>
</targetListInfo>
</sourceDataInfo>
</eventInfo>
</sourceDataInfo>
```

```
</targetBase64Data>
</targetImageData>
</item>
</targetListInfo>

<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetType>person</targetType>
    <personAttribute>
      <age><![CDATA[young]]></age>
      <backpack><![CDATA[backpack]]></backpack>
      <glasses><![CDATA[noGlasses]]></glasses>
      <hat><![CDATA[hat]]></hat>
      <lowerColor><![CDATA[yellow]]></lowerColor>
      <lowerLength><![CDATA[shorts]]></lowerLength>
      <mask><![CDATA[noMask]]></mask>
      <orient><![CDATA[face]]></orient>
      <sex><![CDATA[male]]></sex>
      <shoulderBag><![CDATA[noShoulderBag]]></shoulderBag>
      <skirt><![CDATA[noSkirt]]></skirt>
      <upperColor><![CDATA[white]]></upperColor>
      <upperLength><![CDATA[shortSleeve]]></upperLength>
    </personAttribute>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]]>
      </targetBase64Data>
```

```
        </targetImageData>
    </item>
</targetListInfo>
</config>
```

Example of Entity Data(absenceDetect):

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
    <messageType>alarmData</messageType>
    <deviceInfo>
        <deviceName><![CDATA[AI-BOX]]></deviceName>
        <ip><![CDATA[10.10.10.10]]></ip>
        <mac><![CDATA[00:18:ae:34:44:56]]></mac>
        <channelId>1</channelId>
    </deviceInfo>
    <smartType>absenceDetect</smartType>
    <currentTime>1699520826280829</currentTime>
    <eventInfo>
        <item>
            <eventId>301</eventId>
            <targetId>8</targetId>
            <boundaryType>area</boundaryType>
            <pointGroup>
                <item>
                    <x>2000</x>
                    <y>1400</y>
                </item>
                <item>
                    <x>8225</x>
                    <y>2266</y>
                </item>
            </pointGroup>
        </item>
    </eventInfo>
```

```
        <x>7450</x>
        <y>7500</y>
    </item>
    <item>
        <x>2425</x>
        <y>8466</y>
    </item>
</pointGroup>
<rect>
    <x1>1789</x1>
    <y1>1944</y1>
    <x2>6363</x2>
    <y2>9652</y2>
</rect>
<targetAdditionalInfo>
    <name><![CDATA[Jack]]></name>
    <telephone><![CDATA[0085-12345678]]></telephone>
</targetAdditionalInfo>
</item>
</eventInfo>
<sourceDataInfo>
    <dataType>0</dataType>
    <width>1280</width>
    <height>720</height>
    <sourceBase64Length>205168</sourceBase64Length>
    <sourceBase64Data>
        <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
    </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
    <item>
        <targetId>8</targetId>
```

```

<targetType>motor</targetType>
<targetImageData>
  <dataType>0</dataType>
  <width>48</width>
  <height>96</height>
  <targetBase64Length>2310</targetBase64Length>
  <targetBase64Data>
    <![CDATA[/9j/4AERRRgABA..... //2Q==]]>
  </targetBase64Data>
</targetImageData>
</item>
</targetListInfo>
</config>

```

[Tips]:

2.6.2 Statistical Event

SendAlarmData	
Description	Push the event data for statistical event to the subscribe server/client or specific server address when an alarm happens. Applicable to: <ul style="list-style-type: none"> • Target Counting by Line • Target Counting by Area
Typical URL	<ul style="list-style-type: none"> • Push by subscription& Push by server address(default): /SendAlarmData • Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User

Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.--> </deviceInfo> <smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1.--> </smartType> <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime> <eventInfo><!--Required; list,event information. Refer to Appendix A.2.--> </eventInfo> <sourceDataInfo><!--Optional; Source picture. Refer to Appendix A.2.--> </sourceDataInfo> <targetListInfo><!--Optional; list,List of target picture and attribute infomation. Refer to Appendix A.2.--> </targetListInfo> <statisticalData><!--Required;Target quantity statistics. Refer to Appendix A.2.--> </statisticalData> </config> </pre>	
<p>Example of Entity Data(Target Counting by Area):</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <config version="2.0.0" xmlns="http://www.ipc.com/ver10"> <messageType>alarmData</messageType> <deviceInfo> <deviceName><![CDATA[IPC]]</deviceName> </pre>	

```
<ip><![CDATA[10.10.10]]></ip>
<mac><![CDATA[00:18:ae:34:44:56]]></mac>
<channelId>1</channelId>
</deviceInfo>
<smartType>targetCountingByArea</smartType>
<currentTime>1699520826280829</currentTime>
<eventInfo>
  <item>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <boundaryType>area</boundaryType>
    <pointGroup>
      <item>
        <x>2000</x>
        <y>1400</y>
      </item>
      <item>
        <x>8225</x>
        <y>2266</y>
      </item>
      <item>
        <x>7450</x>
        <y>7500</y>
      </item>
      <item>
        <x>2425</x>
        <y>8466</y>
      </item>
    </pointGroup>
    <rect>
      <x1>1789</x1>
      <y1>1944</y1>
```

```
<x2>6363</x2>
<y2>9652</y2>
</rect>
</item>
</eventInfo>
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]>
  </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetType>motor</targetType>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]>
      </targetBase64Data>
    </targetImageData>
  </item>
</targetListInfo>
<statisticalData>
  <enterCarCount>2</enterCarCount>
  <enterPersonCount>3</enterPersonCount>
```

```
<enterBikeCount>1</enterBikeCount>
<leaveCarCount>0</leaveCarCount>
<leavePersonCount>0</leavePersonCount>
<leaveBikeCount>0</leaveBikeCount>
<existCarCount>2</existCarCount>
<existPersonCount>3</existPersonCount>
<existBikeCount>1</existBikeCount>
<onceEnterCarCount>2</onceEnterCarCount>
<onceEnterPersonCount>3</onceEnterPersonCount>
<onceEnterBikeCount>1</onceEnterBikeCount>
<onceLeaveCarCount>0</onceLeaveCarCount>
<onceLeavePersonCount>0</onceLeavePersonCount>
<onceLeaveBikeCount>0</onceLeaveBikeCount>
</statisticalData>
</config>
```

Example of Entity Data(Target Counting by Line):

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
    <channelId>1</channelId>
  </deviceInfo>
  <smartType>targetCountingByLine</smartType>
  <currentTime>1699520826280829</currentTime>
  <eventInfo >
    <item>
      <eventId>208</eventId>
      <targetId>8</targetId>
```

```
<boundaryType>tripwire</boundaryType>
<directionLine>
  <startPoint>
    <x>975</x>
    <y>5233</y>
  </startPoint>
  <endPoint>
    <x>9950</x>
    <y>30</y>
  </endPoint>
  <direction>rightortop</direction>
</directionLine>
<rect>
  <x1>1789</x1>
  <y1>1944</y1>
  <x2>6363</x2>
  <y2>9652</y2>
</rect>
</item>
</eventInfo>
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
  </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
```

```

<targetType>motor</targetType>
  <targetImageData>
    <dataType>0</dataType>
    <width>48</width>
    <height>96</height>
    <targetBase64Length>2310</targetBase64Length>
    <targetBase64Data>
      <![CDATA[/9j/4AERRRgABA..... //2Q==]]>
    </targetBase64Data>
  </targetImageData>
</item>
</targetListInfo>
<statisticalData>
  <enterCarCount>2</enterCarCount>
  <enterPersonCount>3</enterPersonCount>
  <enterBikeCount>1</enterBikeCount>
  <leaveCarCount>0</leaveCarCount>
  <leavePersonCount>0</leavePersonCount>
  <leaveBikeCount>0</leaveBikeCount>
  <existCarCount>2</existCarCount>
  <existPersonCount>3</existPersonCount>
  <existBikeCount>1</existBikeCount>
  <onceEnterCarCount>2</onceEnterCarCount>
  <onceEnterPersonCount>3</onceEnterPersonCount>
  <onceEnterBikeCount>1</onceEnterBikeCount>
  <onceLeaveCarCount>0</onceLeaveCarCount>
  <onceLeavePersonCount>0</onceLeavePersonCount>
  <onceLeaveBikeCount>0</onceLeaveBikeCount>
</statisticalData>
</config>

```

[Tips]:

2.6.3 Temperature Measurement

SendAlarmData	
Description	Push the event data for temperature event to the subscribe server/client or specific server address when an alarm happens.
Typical URL	<ul style="list-style-type: none"> • Push by subscription& Push by server address(default): /SendAlarmData • Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.--> </deviceInfo> <smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1.--> </smartType> <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime> <temperatureEventInfo><!--Required;list,event information. Refer to Appendix A.2.--> </temperatureEventInfo> <associationTemperatureEventInfo><!--Optional; list,Event information of associated optical channels. Refer to Appendix A.2.--> </associationTemperatureEventInfo> </pre>	

```
<sourceDataInfo><!--Optional; Source picture. Refer to Appendix A.2.-->
</sourceDataInfo>
<associationSourceDataInfo><!--Optional; list, source image of associated optical channels.-->
</associationSourceDataInfo>
</config>
```

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
    <channelId>2</channelId>
  </deviceInfo>
  <smartType>temperature</smartType>
  <currentTime>1699520826280829</currentTime>

  <temperatureEventInfo>
    <item>
      <ruleId>1</ruleId>
      <maxTemperature>
        <value>100000</value>
        <point>
          <x>2000</x>
          <y>3000</y>
        </point>
      </maxTemperature>
      <minTemperature>
        <value>80000</value>
        <point>
          <x>1500</x>
```

```
<y>1000</y>
</point>
</minTemperature>
<averageTemperature>9000</averageTemperature>
<ruleInfo>
  <ruleShapeType>line</ruleShapeType>
  <pointGroup>
    <item>
      <x>2000</x>
      <y>1400</y>
    </item>
    <item>
      <x>8225</x>
      <y>2266</y>
    </item>
  </pointGroup>
</ruleInfo>
</item>
</temperatureEventInfo>
<associationTemperatureEventInfo>
  <item>
    <ruleId>1</ruleId>
    <channelId>1</channelId>
    <maxTemperaturePoint>
      <point>
        <x>2100</x>
        <y>1500</y>
      </point>
    </maxTemperaturePoint>
    <minTemperaturePoint>
      <point>
        <x>8100</x>
```

```
<y>2000</y>
</point>
</minTemperaturePoint>

<ruleInfo>
  <ruleShapeType>line</ruleShapeType>
  <pointGroup>
    <item>
      <x>2100</x>
      <y>1500</y>
    </item>
    <item>
      <x>8100</x>
      <y>2050</y>
    </item>
  </pointGroup>
</ruleInfo>
</item>
</associationTemperatureEventInfo>

<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
  </sourceBase64Data>
</sourceDataInfo>

<associationSourceDataInfo>
  <item>
```

```

<channelId>1</channelId>
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
  </sourceBase64Data>
</sourceDataInfo>
</item>
</associationSourceDataInfo>
</config>

```

[Tips]:

1. In order to measure accuracy, the values of elements<maxTemperature>,<minTemperature>,<averageTemperature> has been multiplied by 10000.

2.6.4 Fire

SendAlarmData	
Description	Push the event data for fire event to the subscribe server/client or specific server address when an alarm happens.
Typical URL	<ul style="list-style-type: none"> Push by subscription& Push by server address(default): /SendAlarmData Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.

Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.--> </deviceInfo> <smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1.--> </smartType> <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime> <eventInfo><!--Required; list,event information. Refer to Appendix A.2.--> </eventInfo> <associationEventInfo><!--Optional; list. Information on associated visible optical channels--> </associationEventInfo> <sourceDataInfo><!--Optional; Source picture. Refer to Appendix A.2.--> </sourceDataInfo> <targetListInfo><!--Optional; List of target picture and attribute infomation. Refer to Appendix A.2.--> </targetListInfo> <associationSourceDataInfo><!--Optional; When the thermal channel triggers an event, it will carry an additional source image of other channel. Refer to Appendix A.2.--> </associationSourceDataInfo> </config> </pre>	
<p>Example of Entity Data:</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <config version="2.0.0" xmlns="http://www.ipc.com/ver10"> <messageType>alarmData</messageType> <deviceInfo> </pre>	

```
<deviceName><![CDATA[IPC]]></deviceName>
<ip><![CDATA[10.10.10.10]]></ip>
<mac><![CDATA[00:18:ae:34:44:56]]></mac>
<channelId>2</channelId>
</deviceInfo>
<smartType>fire</smartType>
<currentTime>1699520826280829</currentTime>
<eventInfo>
  <item>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <rect>
      <x1>1789</x1>
      <y1>1944</y1>
      <x2>6363</x2>
      <y2>9652</y2>
    </rect>
  </item>
</eventInfo>
<associationEventInfo>
  <item>
    <channelId>1</channelId>
    <eventId>208</eventId>
    <targetId>8</targetId>
    <rect>
      <x1>1800</x1>
      <y1>1950</y1>
      <x2>6370</x2>
      <y2>9660</y2>
    </rect>
  </item>
</associationEventInfo>
```

```
<sourceDataInfo>
  <dataType>0</dataType>
  <width>1280</width>
  <height>720</height>
  <sourceBase64Length>205168</sourceBase64Length>
  <sourceBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]>
  </sourceBase64Data>
</sourceDataInfo>
<targetListInfo>
  <item>
    <targetId>8</targetId>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AERRRgABA..... //2Q==]>
      </targetBase64Data>
    </targetImageData>
  </item>
</targetListInfo>
<associationSourceDataInfo>
  <item>
    <channelId>1</channelId>
    <sourceDataInfo>
      <dataType>0</dataType>
      <width>1280</width>
      <height>720</height>
      <sourceBase64Length>205168</sourceBase64Length>
      <sourceBase64Data>
```

<pre> <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]> </sourceBase64Data> </sourceDataInfo> </item> </associationSourceDataInfo> </config> </pre>
[Tips]:

2.6.5 Face Detection

SendAlarmData	
Description	Push the event data for face detection event to the subscribe server/client or specific server address when an alarm happens.
Typical URL	<ul style="list-style-type: none"> Push by subscription& Push by server address(default): /SendAlarmData Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.--> </deviceInfo> <smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1--> </pre>	

```
</smartType>
<currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime>

<sourceDataInfo><!--Optional; Source picture. Refer to Appendix A.2.-->
</sourceDataInfo>

<faceListInfo><!--Required; list,List of face picture and attribute infomation. Refer to Appendix
A.2.-->
</faceListInfo>
</config>
```

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
    <channelId>2</channelId>
  </deviceInfo>
  <smartType>videoFaceDetect</smartType>
  <currentTime>1699520826280829</currentTime>

  <sourceDataInfo>
    <dataType>0</dataType>
    <width>1280</width>
    <height>720</height>
    <sourceBase64Length>205168</sourceBase64Length>
    <sourceBase64Data>
      <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
    </sourceBase64Data>
  </sourceDataInfo>
```

```
<faceListInfo>
  <item>
    <targetId>2</targetId>
    <rect>
      <x1>1789</x1>
      <y1>1944</y1>
      <x2>6363</x2>
      <y2>9652</y2>
    </rect>
    <faceAttribute>
      <age><![CDATA[young]]></age>
      <sex><![CDATA[male]]></sex>
      <glasses><![CDATA[noGlasses]]></glasses>
      <mask><![CDATA[noMask]]></mask>
    </faceAttribute>
    <targetImageData>
      <dataType>0</dataType>
      <width>48</width>
      <height>96</height>
      <targetBase64Length>2310</targetBase64Length>
      <targetBase64Data>
        <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
      </targetBase64Data>
    </targetImageData>
  </item>
</faceListInfo>
</config>
```

[Tips]:

2.6.6 Face Match

SendAlarmData

Description	Push the event data for face detection and match event to the subscribe server/client or specific server address when an alarm happens.
Typical URL	<ul style="list-style-type: none"> • Push by subscription& Push by server address(default): /SendAlarmData • Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.--> </deviceInfo> <smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1--> </smartType> <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime> <faceInfo><!--Required; face picture and attribute infomation. Refer to Appendix A.2.--> </faceInfo> <faceMatchInfo><!--Required; face match infomation. Refer to Appendix A.2.--> </faceMatchInfo> </config> </pre>	
<p>Example of Entity Data:</p> <pre> <?xml version="1.0" encoding="UTF-8" ?> <config version="2.0.0" xmlns="http://www.ipc.com/ver10"> </pre>	

```
<messageType>alarmData</messageType>
<deviceInfo>
  <deviceName><![CDATA[IPC]]></deviceName>
  <ip><![CDATA[10.10.10.10]]></ip>
  <mac><![CDATA[00:18:ae:34:44:56]]></mac>
  <channelId>2</channelId>
</deviceInfo>
<smartType>videoFaceMatch</smartType>
<currentTime>1699520826280829</currentTime>

<faceInfo>
  <targetId>2</targetId>
  <targetImageData>
    <dataType>0</dataType>
    <width>48</width>
    <height>96</height>
    <targetBase64Length>2310</targetBase64Length>
    <targetBase64Data>
      <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
    </targetBase64Data>
  </targetImageData>
</faceInfo>
<faceMatchInfo>
  <matchResult>true</matchResult>
  <similarity>80</similarity>
  <personInfo>
    <groupId><![CDATA[1]]></groupId>
    <name><![CDATA[Tom]]></name>
    <sex><![CDATA[male]]></sex>
    <identifyNumber><![CDATA[1234567890231456]]></identifyNumber>
    <telephone><![CDATA[86547891]]> </telephone>
  </personInfo>
```

```

    <personPicture>
      <dataType>0</dataType>
      <width>102</width>
      <height>99</height>
      <imageBase64Length>2812</imageBase64Length>
      <imageBase64Data><![CDATA[/9j/4AAQSkZJRgB.....RRRQB//2Q==]]>
      </imageBase64Data>
    </personPicture>
  </faceMatchInfo>

</config>

```

[Tips]:

2.6.7 License Plate Detection

SendAlarmData	
Description	Push the event data for license plate detection event to the subscribe server/client or specific server address when an alarm happens.
Typical URL	<ul style="list-style-type: none"> Push by subscription& Push by server address(default): /SendAlarmData Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre> <config version="" xmlns="http://www.ipc.com/ver10"> <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType> </pre>	

```
<deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.-->
</deviceInfo>

<smartType><!--Required; smartTypeE,Type of intelligent event. Refer to Appendix A.1-->
</smartType>

<currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime>

<sourceDataInfo><!--Required; Source picture. Refer to Appendix A.2.-->
</sourceDataInfo>

<licensePlateListInfo><!--Required; list,List of license plate picture and attribute infomation. Refer to
Appendix A.2.-->
</licensePlateListInfo>
</config>
```

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmData</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
    <channelId>2</channelId>
  </deviceInfo>
  <smartType>vehicle</smartType>
  <currentTime>1699520826280829</currentTime>

  <sourceDataInfo>
    <dataType>0</dataType>
    <width>1280</width>
    <height>720</height>
    <sourceBase64Length>205168</sourceBase64Length>
```

```
<sourceBase64Data>
  <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]>
```

```
</sourceBase64Data>
```

```
</sourceDataInfo>
```

```
<licensePlateListInfo>
```

```
  <item>
```

```
    <targetId>2</targetId>
```

```
    <rect>
```

```
      <x1>1789</x1>
```

```
      <y1>1944</y1>
```

```
      <x2>2363</x2>
```

```
      <y2>2652</y2>
```

```
    </rect>
```

```
    <licensePlateAttribute>
```

```
      <licensePlateNumber><![CDATA[A.8888]]></licensePlateNumber>
```

```
      <color><![CDATA[red]]></color>
```

```
    </licensePlateAttribute>
```

```
    <carRect>
```

```
      <x1>100</x1>
```

```
      <y1>3000</y1>
```

```
      <x2>6363</x2>
```

```
      <y2>9652</y2>
```

```
    </carRect>
```

```
    <carAttribute>
```

```
      <carType><![CDATA[suv]]></carType>
```

```
      <color><![CDATA[red]]></color>
```

```
      <brand><![CDATA[BMW]]></brand>
```

```
      <model><![CDATA[unknown]]></model>
```

```
    </carAttribute>
```

```
    <licensePlateMatchInfo>
```

```
      <carOwner><![CDATA[TOM]]></carOwner>
```

```

<licensePlateNumber><![CDATA[888888]]></licensePlateNumber>
<groupId>1</groupId>
<groupName><![CDATA[whiteList]]></groupName>
<telephone><![CDATA[13900001235]]></telephone>
</licensePlateMatchInfo>
<targetImageData>
  <dataType>0</dataType>
  <width>48</width>
  <height>96</height>
  <targetBase64Length>2310</targetBase64Length>
  <targetBase64Data>
    <![CDATA[/9j/4AAQSkZJRgABA..... //2Q==]]>
  </targetBase64Data>
</targetImageData>
</item>
</faceListInfo>
</config>

```

[Tips]:

2.7 SendAlarmStatus

2.7.1 Alarm Status

SendAlarmStatus	
Description	Push the alarm status to the subscribe server/client or specific server address when an alarm happens.
Typical URL	<ul style="list-style-type: none"> • Push by subscription & Push by server address(default): /SendAlarmStatus • Push by server address(custom path): custom path

Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR

<config>XML Block:

```

<config version="" xmlns="http://www.ipc.com/ver10">
  <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType>
  <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.-->
  </deviceInfo>
  <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime>

  <alarmStatusInfo><!--Required.the status of the alarm.-->
    <deviceAlarmStatus><!--Optional.-->
      <noDisk type="boolean">
        <!--Optional. No disk alarm status.Only valid for NVR.-->
      </noDisk>
      <diskFull type="list" maxCount="" count=""><!--Optional. Disk full alarm.-->
        <item>
          <diskId type="string"><!--Required. Disk ID.--></diskId>
        </item>
      </diskFull>
      <diskPullOut type="list" maxCount="" count=""><!--Optional. Disk pullout. Only valid for NVR.-->
        <item>
          <slotId type="string"><!--Required. Disk slot ID.--></slotId>
        </item>
      </diskPullOut>
      <diskRWEError type="list" maxCount="" count=""><!--Optional. Disk read/write error.-->
        <item>

```

```
<diskId type="string"><!--Required. Disk ID.--></diskId>
  <status type="boolean"><!--Required. alarm status.--><status>
</item>
</diskRWError>
<diskFaultAlarm type="list" maxCount="" count=""><!--Optional.Disk fault.-->
  <item>
    <diskId type="string"><!--Required. Disk ID.--></diskId>
    <status type="boolean"><!--Required. alarm status.--><status>
  </item>
</diskFaultAlarm>
<raidHotBackupErrorAlarm type="list" maxCount="" count=""><!--Optional.RAID hot
backup error. Only valid for NVR.-->
  <item>
    <raidId type="string"><!--Required. RAID ID.--></raidId>
    <status type="boolean"><!--Required. alarm status.--><status>
  </item>
</raidHotBackupErrorAlarm>
<raidUnavailableAlarm type="list" maxCount="" count=""><!--Optional. Only valid for
NVR.-->
  <item>
    <raidId type="string"><!--Required. RAID ID.--></raidId>
    <status type="boolean"><!--Required. alarm status.--><status>
  </item>
</raidUnavailableAlarm>
<raidSubhealthAlarm type="list" maxCount="" count=""><!--Optional. Only valid for
NVR.-->
  <item>
    <raidId type="string"><!--Required. RAID ID.--></raidId>
    <status type="boolean"><!--Required. alarm status.--><status>
  </item>
</raidSubhealthAlarm>
<ipConflict type="list" maxCount="" count=""><!--Optional.-->
  <item>
```

```
<ipAddress type="string"><!--Required. Conflicting IP.--></ipAddress>
  <status type="boolean"><!--Required. alarm status.--><status>
    </item>
  </ipConflict >
  <illegalAccess type="boolean"><!--Optional. Illegal login alarm.--></illegalAccess>
  <wifiDisconnect type="boolean">
    <!--Optional. wifi disconnect alarm status.-->
  </wifiDisconnect>
</deviceAlarmStatus>

<sensorAlarmIn><!--Optional.list,the status of alarm in.-->
  <item id="1">
    <!--Required.boolean. the attribute 'id' represents Alarm in ID(uint32).-->
  </item>
</sensorAlarmIn>
<motionAlarm><!--Optional.list, the alarm status of motion.-->
  <item id="1">
    <!--Required. boolean, the attribute 'id' represents channel ID.-->
  </item>
</motionAlarm>
<perimeterAlarm><!--Optional.list,the alarm status of region intrusion.-->
  <item id="1">
    <!--Required. boolean, the attribute 'id' represents channel ID.-->
  </item>
</perimeterAlarm>
<tripwireAlarm><!--Optional. list,the alarm status of line crossing. -->
  <item id="1">
    <!--Required. boolean, the attribute 'id' represents channel ID.-->
  </item>
</tripwireAlarm>
<oscAlarm><!--Optional.list,the alarm status of object status change.-->
  <item id="1">
```

```
<!--Required. boolean, the attribute 'id' represents channel ID.-->
</item>
</oscAlarm>
<sceneChange><!--Optional.list, the status of the scene change alarm.-->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
    </item>
  </sceneChange>
<clarityAbnormal><!--Optional.list, the status of the clarity abnormal alarm.-->
  <item id="1">
    <!--Required. boolean,the attribute 'id' represents channel ID.-->
    </item>
  </clarityAbnormal>
<colorAbnormal><!--Optional.list, the status of the color abnormal alarm.-->
  <item id="1">
    <!--Required. boolean,the attribute 'id' represents channel ID.-->
    </item>
  </colorAbnormal>
<vfdAlarm><!--Optional.list, the status of the video face detection alarm -->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
    </item>
  </vfdAlarm>
<vfdMatchAlarm><!--Optional.list, the status of the video face match alarm -->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
    </item>
  </vfdMatchAlarm>
<aoiEntryAlarm><!--Optional. the status of region entrance alarm -->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
    </item>
```

```
</aoiEntryAlarm>
<aoiLeaveAlarm><!--Optional.list, the status of the area of interest leave alarm.-->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
  </item>
</aoiLeaveAlarm>
<passlineAlarm><!--Optional.list, the status of the target counting by line alarm.-->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
  </item>
</passlineAlarm>
<trafficAlarm><!--Optional.list, the status of the target counting by area alarm.-->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
  </item>
</trafficAlarm>
<pvdAlarm><!--Optional.list, the status of illegal parking alarm.-->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
  </item>
</pvdAlarm>
<loiteringAlarm><!--Optional.list,The status of loitering alarm.-->
  <item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
  </item>
</loiteringAlarm>
<crowdGatheringAlarm><!--Optional.list, The status of the crowd gathering detection alarm -->
  <item id="1">
    <!--Required.boolean, the attribute 'id' represents channel ID.-->
  </item>
</crowdGatheringAlarm>
<fireAlarm><!--Optional.list, The status of the fire detection alarm.-->
```

```
<item id="1">
    <!--Required.boolean,the attribute 'id' represents channel ID.-->
</item>
</fireAlarm>
<temperatureAlarm><!--Optional.list, The status of the temperature detection alarm.-->
    <item id="1">
        <!--Required.boolean,the attribute 'id' represents channel ID.-->
    </item>
</temperatureAlarm>
<soundAbruptUpAlarm><!--Optional,list, the status of the abnormal sound alarm(abrupt up),
The sound rises sharply.-->
    <item id="1">
        <!--Required.boolean,the attribute 'id' represents channel ID,Optional,If the attribute
appears, it indicates an alarm for the corresponding channel; if not, it indicates an
alarm for the device.-->
    </item>
</soundAbruptUpAlarm>
<soundAbruptDownAlarm><!--Optional,list, the status of the abnormal sound alarm(abrupt
down), The sound drops sharply.-->
    <item id="1">
        <!--Required.boolean,the attribute 'id' represents channel ID,Optional,If the attribute
appears, it indicates an alarm for the corresponding channel; if not, it indicates an
alarm for the device.-->
    </item>
</soundAbruptDownAlarm>
<chlOfflineAlarm type="list" maxCount="" count=""><!--Optional.Only valid for NVR.-->
    <item id="1">
        <!--Required. boolean,channel offline alarm status. the attribute 'id' represents channel
ID.-->
    </item>
</chlOfflineAlarm>
<personnelTailgatingAlarm type="list" maxCount="" count=""><!--Optional.Only valid for
AI-BOX.-->
    <item id="1">
        <!--Required. boolean,personnel tailgating alarm status. the attribute 'id' represents
channel ID.-->
```

```

    </item>
  </personnelTailgatingAlarm>
  <personnelIntrusionAlarm type="list" maxCount="" count=""><!--Optional.Only valid for
  AI-BOX.-->
    <item id="1">
      <!--Required. boolean, Personnel entering the detection area alarm status. the attribute
      'id' represents channel ID.-->
    </item>
  </personnelIntrusionAlarm>
  <absenceDetectAlarm type="list" maxCount="" count=""><!--Optional.Only valid for
  AI-BOX.-->
    <item id="1">
      <!--Required. boolean, Personnel leaving their designated location alarm status. the
      attribute 'id' represents channel ID.-->
    </item>
  </absenceDetectAlarm>
  <videoLossAlarm type="list" maxCount="" count=""><!--Optional.Only valid for DVR.-->
    <item id="1">
      <!--Required. boolean,channel offline alarm status. the attribute 'id' represents
      channel ID.-->
    </item>
  </videoLossAlarm>
</alarmStatusInfo>
</config>

```

Example of Entity Data:

```

<?xml version="1.0" encoding="UTF-8"?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>alarmStatus</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
  </deviceInfo>
  <currentTime>1699520826280829</currentTime>

```

<pre> <alarmStatusInfo> <sensorAlarmIn> <item id="1">true</item> </sensorAlarmIn> <motionAlarm> <item id="1">true</item> <item id="2">true</item> </motionAlarm> </alarmStatusInfo> </config> </pre>
<p>[Tips]:</p> <p>1.The "id" attribute for each item starts from "1".</p>

2.8 SendTrajectoryData

2.8.1 Target Trajectory

SendTrajectoryData	
Description	Push the target trajectory data to the subscribe server/client or specific server address.
Typical URL	<ul style="list-style-type: none"> • Push by subscription& Push by server address(default): /SendTrajectoryData • Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by subscription(baseSubscribe), Push by server address.
Applicable products	IPC,NVR,AI-BOX,DVR

<config> XML Block:

```
<config version="" xmlns="http://www.ipc.com/ver10">
  <messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType>
  <deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.-->
  </deviceInfo>
  <currentTime><!--Required; uint64,The time of occurrence of the event.--></currentTime>
  <trajectory><!--Required; list.-->
    <item>
      <targetId><!--Required; uint32,Target ID.--></targetId>
      <targetType><!--Required; targetTypeE,target type.--></targetType>
      <rect><!--Required; The coordinates of face detection area.Refer to Appendix A.2.-->
      </rect>
    </item>
  </trajectory>
</config>
```

Example of Entity Data:

```
<?xml version="1.0" encoding="UTF-8" ?>
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>trajectory</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
  </deviceInfo>
  <currentTime>1699520826280829</currentTime>
  <trajectory>
    <item>
      <targetId>10</targetId>
      <targetType>person</targetType>
      <rect>
        <x1>3778</x1>
```

```

        <y1>208</y1>
        <x2>4318</x2>
        <y2>2708</y2>
    </rect>
</item>
</trajectory>
</config>

```

[Tips]:

1. This interface is only applicable to the following target types:
 - person
 - car
 - motor.

2.9 SendKeepalive

2.9.1 Keepalive

SendKeepalive	
Description	Send heartbeat messages to specific servers to keep the connection alive.
Typical URL	<ul style="list-style-type: none"> • Push by server address(default): /SendKeepalive • Push by server address(custom path): custom path
Entity Data	<config>
Successful Response	None
Direction	Device ---> User
Applicable scenarios	Push by server address(persistent connection).
Applicable products	IPC,NVR,AI-BOX,DVR
<p><config> XML Block:</p> <pre><config version="" xmlns="http://www.ipc.com/ver10"></pre>	

```
<messageType><!--Required; messageTypeE,Refer to Appendix A.1.--></messageType>
<deviceInfo><!--Required; Device information of data source. Refer to Appendix A.2.-->
</deviceInfo>
</config>
```

Example of Push by server address:

```
<config version="2.0.0" xmlns="http://www.ipc.com/ver10">
  <messageType>keepalive</messageType>
  <deviceInfo>
    <deviceName><![CDATA[IPC]]></deviceName>
    <ip><![CDATA[10.10.10.10]]></ip>
    <mac><![CDATA[00:18:ae:34:44:56]]></mac>
  </deviceInfo>
</config>
```

[Tips]:

1. This interface is also used to test the connectivity of http post.

A.1 Type Definition

A.1.1 smartTypeE

```
<smartTypeE><!--Define event types.-->
  <enum>lineCrossing</enum><!--Optional; Line Crossing and Intrusion.-->
  <enum>regionIntrusion</enum><!--Optional; Line Crossing and Intrusion.-->
  <enum>objectStatusChange</enum><!--Optional; Object Status Change.-->
  <enum>videoFaceDetect</enum><!--Optional; Video Face Detection.-->
  <enum>videoFaceMatch</enum><!--Optional; Video Face Detection and Match.-->
  <enum>vehicle</enum><!--Optional; License Plate Detection.-->
  <enum>regionEntrance</enum><!--Optional; Region Entrance.-->
  <enum>regionExiting</enum><!--Optional; Region Exiting.-->
  <enum>targetCountingByLine</enum><!--Optional; Target Counting by Line.-->
  <enum>targetCountingByArea</enum><!--Optional; Target Counting by Area.-->
  <enum>videoMetadata</enum><!--Optional; Video Metadata.-->
  <enum>illegalParkDetect</enum><!--Optional; Illegal Parking Detection.-->
  <enum>loitering</enum><!--Optional; Loitering Detection.-->
  <enum>crowdGathering</enum><!--Optional; Crowd gathering.-->
  <enum>fire</enum><!--Optional; Fire Detection.-->
  <enum>temperature</enum><!--Optional; Temperature measurement.-->
  <enum>abnormalVideoDetect</enum><!--Optional; Abnormal Video Detection.-->
  <enum>abnormalSoundDetect</enum><!--Optional; Abnormal Sound Detection.-->
  <enum>motion</enum><!--Optional; motion.-->
  <enum>sensor</enum><!--Optional; sensor.-->
  <enum>personnelTailgating</enum><!--Optional; Detecting tailgating personnel.-->
  <enum>personnelIntrusion </enum><!--Optional; Personnel entering the detection area.-->
```

```
<enum>absenceDetect</enum><!--Optional; Personnel leaving their designated location.-->
</smartTypeE>
```

A.1.2 messageTypeE

```
<messageTypeE><!--Define the type of message.-->
  <enum>alarmData</enum><!--Required; Alarm data.-->
  <enum>alarmStatus</enum><!--Required; The alarm status(event or alarm).-->
  <enum>trajectory</enum><!--Required; Trajectory data of target.-->
  <enum>keepalive</enum><!--Required; Heartbeat message.-->
</messageTypeE>
```

A.1.3 targetTypeE

```
<targetTypeE><!--Define the object types for intelligent detection.-->
  <enum>person</enum>
  <enum>car</enum>
  <enum>motor</enum>
</targetTypeE>
```

A.1.4 ruleShapeTypeE

```
<ruleShapeTypeE><!--Type of detection area.-->
  <enum>point</enum>
  <enum>line</enum>
  <enum>area</enum>
</ruleShapeTypeE>
```

A.1.5 tripwireDirectionE

```
<tripwireDirectionE><!--Define the direction of tripwire.-->
  <enum>any</enum>
  <enum>rightortop</enum><!--Looking from the starting point to the endpoint of the rule line, the
  target crosses from the left side to the right side of the rule line.-->
  <enum>leftorbottom</enum><!--Looking from the starting point to the endpoint of the rule line,
  the target crosses from the right side to the left side of the rule line.-->
```

```
</tripwireDirectionE>
```

A.1.6 boundaryTypeE

```
<boundaryTypeE><!--Define the type of detect boundary.-->
  <enum>tripwire</enum>
  <enum>area</enum>
</boundaryTypeE>
```

A.2 Definition of Structure

A.2.1 deviceInfo

```
<deviceInfo><!--Device information of data source.-->
  <deviceName><!--Required; string,Name of the device.--></deviceName>
  <ip><!--Required;string,IP address of the device.--></ip>
  <mac><!--Required; string,MAC address of the device.--></mac>
  <channelId><!--Required/Optional; uint32,channel Id of event. If it is a channel event, it is
    required. If it is a device event, it can be omitted.-->
  </channelId>
</deviceInfo>
```

A.2.2 sourceDataInfo

```
<sourceDataInfo><!--Source picture infomation.-->
  <dataType><!--Required;uint32,Picture type.0:JPG.--></dataType>
  <width><!--Required; uint32,The width of the picture.--></width>
  <height><!--Required; uint32,The height of the picture.--></height>
  <sourceBase64Length>
    <!--Required; uint32,The length of <sourceBase64Data>.-->
  </sourceBase64Length>
  <sourceBase64Data>
    <!--Required; string,picture data encoded with Base64.-->
  </sourceBase64Data>
</sourceDataInfo>
```

A.2.3 associationSourceDataInfo

```
<associationSourceDataInfo><!--list,source data of associated channel.-->
  <item>
    <channelId>
      <!--Required; uint32,channel Id of source picture.-->
    </channelId>
    <sourceDataInfo><!--Required; additional source picture infomation. Refer to Appendix
      A.2.-->
    </sourceDataInfo>
  </item>
</associationSourceDataInfo>
```

A.2.4 targetImageData

```
<targetImageData><!--Target picture data.-->
  <dataType><!--Required;uint32,Picture type.0:JPG.--></dataType>
  <width><!--Required; uint32,The width of the picture.--></width>
  <height><!--Required; uint32,The height of the picture.--></height>
  <targetBase64Length>
    <!--Required; uint32,The length of <targetBase64Data>.-->
  </targetBase64Length>
  <targetBase64Data>
    <!--Required; string,picture data encoded with Base64.-->
  </targetBase64Data>
</targetImageData>
```

A.2.5 rect

```
<rect><!--Define the point coordinates of Rectangular.-->
  <x1><!--Required; uint32,Upper left x-coordinate.--></x1>
  <y1><!-- Required; uint32,Upper left Y-coordinate.--></y1>
  <x2><!--Required; uint32,Bottom right X-coordinate.--></x2>
  <y2><!--Required; uint32,Bottom right Y-coordinate.--></y2>
```

</rect>

A.2.6 directionLine

```
<directionLine><!--Define directional line.-->
  <startPoint><!--Required; The starting point of the line.-->
    <x><!--Required; uint32,x-coordinate.--></x>
    <y><!--Required; uint32,y-coordinate.--></y>
  </startPoint>
  <endPoint><!--Required; The endpoint of the line.-->
    <x><!--Required; uint32,x-coordinate.--></x>
    <y><!--Required; uint32,y-coordinate.--></y>
  </endPoint>
  <direction>
    <!--Required; tripwireDirection,Cross line direction. Refer to the definition in Appendix
    1.-->
  </direction>
</directionLine>
```

A.2.7 pointGroup

```
<pointGroup><!--list,Define one or more point coordinates.-->
  <item>
    <x><!--Required; uint32,x-coordinate.--></x>
    <y><!--Required; uint32,y-coordinate.--></y>
  </item>
</pointGroup>
```

A.2.8 point

```
<point><!--Define the point coordinate.-->
  <x><!--Required; uint32,x-coordinate.--></x>
  <y><!--Required; uint32,y-coordinate.--></y>
</point>
```

A.2.9 personAttribute

```
<personAttribute><!--person attribute information. Refer to Appendix A.3 for the range of values.-->
  <age><!--Required; string.--></age>
  <backpack><!--Required; string.--></backpack>
  <glasses><!--Required; string.--></glasses>
  <hat><!--Required; string.--></hat>
  <lowerColor><!--Required; string.--></lowerColor>
  <lowerLength><!--Required; string.--></lowerLength>
  <mask><!--Required; string.--></mask>
  <orient><!--Required; string.--></orient>
  <sex><!--Required; string.--></sex>
  <shoulderBag><!--Required; string.--></shoulderBag>
  <skirt><!--Required; string.--></skirt>
  <upperColor><!--Required; string.--></upperColor>
  <upperLength><!--Required; string.--></upperLength>
</personAttribute>
```

A.2.10 carAttribute

```
<carAttribute><!--car attribute information. Refer to Appendix A.3 for the range of values.-->
  <carType><!--Required; string.--></carType>
  <color><!--Required; string.--></color>
  <brand><!--Required; string.--></brand>
  <model><!--Required; string.--></model>
</carAttribute>
```

A.2.11 motorAttribute

```
<motorAttribute><!--motor attribute information. Refer to Appendix A.3 for the range of values.-->
  <motorType><!--Required; string.--></motorType>
</motorAttribute>
```

A.2.12 faceAttribute

```
<faceAttribute><!--face attribute information. Refer to Appendix A.3 for the range of values.-->
  <age><!--Required; string.--></age>
  <sex><!--Required; string.--></sex>
```

```
<glasses><!--Required; string.--></glasses>
<mask><!--Required; string.--></mask>
</faceAttribute>
```

A.2.13 licensePlateAttribute

```
<licensePlateAttribute><!--license plate attribute information. Refer to Appendix A.3 for the range of
values.-->
<licensePlateNumber><!--Required; string.--></licensePlateNumber>
<color><!--Required; string.--></color>
</licensePlateAttribute>
```

A.2.14 statisticalData

```
<statisticalData><!-- Target quantity statistics.-->
  <enterCarCount><!--Optional; uint32,Count of cars (entered).--></enterCarCount>
  <enterPersonCount><!--Optional; uint32,Count of people(entered).--></enterPersonCount>
  <enterBikeCount><!--Optional; uint32,Count of bicycles (entered).--></enterBikeCount>
  <leaveCarCount><!--Optional; uint32,Count of cars (left).--></leaveCarCount>
  <leavePersonCount><!--Optional; uint32,Count of people (left).--></leavePersonCount>
  <leaveBikeCount><!--Optional; uint32, Count of bicycles (left).--></leaveBikeCount>
  <existCarCount><!--Optional; uint32,Count of cars (stay).--></existCarCount>
  <existPersonCount><!--Optional; uint32,Count of people (stay).--></existPersonCount>
  <existBikeCount><!--Optional; uint32,Count of bicycles (stay).--></existBikeCount>
  <onceEnterCarCount>
    <!--Optional; uint32,The last statistical increment.-->
  </onceEnterCarCount>
  <onceEnterPersonCount>
    <!--Optional; uint32,Count of people (The increment of the last <enterPersonCount> statistical
value).-->
  </onceEnterPersonCount>
  <onceEnterBikeCount>
    <!--Optional; uint32,Count of bicycles (The increment of the last <enterBikeCount> statistical
value).-->
  </onceEnterBikeCount>

  <onceLeaveCarCount>
```

```
        <!--Optional; uint32,Count of cars (The increment of the last <leaveCarCount> statistical
        value).-->
    </onceLeaveCarCount>
    <onceLeavePersonCount>
        <!--Optional; uint32,Count of people (The increment of the last <leavePersonCount> statistical
        value).-->
    </onceLeavePersonCount>
    <onceLeaveBikeCount>
        <!--Optional; uint32,Count of bicycles (The increment of the last <leaveBikeCount> statistical
        value).-->
    </onceLeaveBikeCount>
</statisticalData>
```

A.2.15 targetListInfo

```
<targetListInfo><!--list,List of target picture and attribute infomation.-->
    <item>
        <targetId><!--Required; uint32,Target ID.--></targetId>
        <targetType><!--Required/Optional; targetTypeE,target type.For fire,object status
        change,temperature,crowd gathering,video face detection are optional.-->
    </targetType>
    <personAttribute><!--Optional/Dependent.person attributes,valid when targetType is
    person,and valid for video metadata.-->
    </personAttribute>
    <carAttribute><!--Optional/Dependent.car attributes,valid when targetType is car,and valid
    for video metadata.-->
    </carAttribute>
    <motorAttribute><!--Optional/Dependent.motor attributes,valid when targetType is motor,and
    valid for video metadata.-->
    </motorAttribute>

    <targetImageData>
        <!--Optional. Target picture information. Refer to Appendix A.2.-->
    </targetImageData>
    </item>
</targetListInfo>
```

A.2.16 faceListInfo

```
<faceListInfo><!--list,List of face picture and attribute infomation.-->
  <item>
    <targetId><!--Required; uint32,Face target ID.--></targetId>
    <rect><!--Optional; The coordinates of the face. Refer to Appendix A.2.-->
    </rect>
    <faceAttribute><!--Optional;face attribute information. Refer to Appendix A.3 for the range of
      values.-->
    </faceAttribute>
    <targetImageData><!--Required;face target picture data.-->
    </targetImageData>
  </item>
</faceListInfo>
```

A.2.17 licensePlateListInfo

```
<licensePlateListInfo><!--list,List of license plate picture and attribute infomation.-->
  <item>
    <targetId><!--Required; uint32,license plate target ID.--></targetId>
    <licensePlateNumber><!--Required;string, car plate number.--></licensePlateNumber>
    <rect><!--Required; The coordinates of the license plate. Refer to Appendix A.2.-->
    </rect>
    <licensePlateAttribute><!--Required;license plate attribute information. Refer to Appendix A.3
      for the range of values.-->
    </licensePlateAttribute>
    <carRect><!--Optional; The coordinates of vehicle, same as <rect>.-->
    </carRect>
    <carAttribute><!--Optional,car attribute.-->
    </carAttribute>
    <licensePlateMatchInfo><!--Optional, License plate comparison information. Refer to
      Appendix A.2.-->
    </licensePlateMatchInfo>
    <targetImageData><!--Required;license plate target picture data.-->
    </targetImageData>
```

```
</item>
</licensePlateListInfo>
```

A.2.18 eventInfo

```
<eventInfo><!--Required; list.-->
  <item>
    <eventId><!--Required; uint32,event ID.--></eventId>
    <targetId><!--Required; uint32,Target ID that triggers the event. 0 indicates invalid. This event
      does not have a target ID.-->
    </targetId>
    <boundaryType><!--Optional; boundaryTypeE, If this node is not present, it indicates full screen
      detection,Refer to Appendix A.1.-->
    </boundaryType>
    <pointGroup><!--Dependent/Optional; The point coordinates of the detection polygon that
      triggers the event. Valid when boundaryType is 'area'. Refer to Appendix A.2.-->
    </pointGroup>
    <directionLine><!--Dependent/Optional; Valid when boundaryType is 'tripwire'.-->
    </directionLine>
    <rect><!--Optional; The coordinates of the target that triggers the event.For object status change,
      crowd gathering:No target rect. Refer to Appendix A.2.-->
    </rect>
    <targetAdditionalInfo><!--Optional; Description information of the target.Only valid for
      absence detect (AI-BOX). Refer to Appendix A.2.-->
    </targetAdditionalInfo>
  </item>
</eventInfo>
```

A.2.19 associationEventInfo

```
<associationEventInfo><!--Required; list.-->
  <item>
    <channelId><!--Required; channel Id of association event.--></channelId>
    <eventId><!--Required; uint32,event ID.--></eventId>
    <targetId><!--Required; uint32,Target ID that triggers the event.--></targetId>
    <rect><!--Optional; The coordinates of the target that triggers the event.For object status change,
      crowd gathering:No target rect. Refer to Appendix A.2.-->
```

```
    </rect>
  </item>
</associationEventInfo>
```

A.2.20 temperatureEventInfo

```
<temperatureEventInfo><!--list,List of temperature event information.-->
  <item>
    <ruleId><!--Required; uint32,Rule ID of temperature measurement.--></ruleId>
    <maxTemperature><!--Required; Maximum temperature information.-->
      <value><!--Required; int32,Temperature value.--></value>
      <point><!--Required; Coordinates of the maximum temperature point.refer to Appendix
        A.2.-->
      </point>
    </maxTemperature>
    <minTemperature><!--Required; Minimum temperature information.-->
      <value><!--Required; int32,Temperature value.--></value>
      <point><!--Required; Coordinates of the maximum temperature point.refer to Appendix
        A.2.-->
      </point>
    </minTemperature>
    </averageTemperature><!--Required; int32,average temperature value.-->
    </averageTemperature>

    <ruleInfo><!--Required; Rules for temperature measurement.-->
      <ruleShapeType>
        <!--Required; ruleShapeTypeE,the type of rule.-->
      </ruleShapeType>
      <pointGroup><!--Dependent; The point coordinates of the detection polygon that
        triggers the event. Refer to Appendix A.2.-->
      </pointGroup>
    </ruleInfo>
  </item>
</temperatureEventInfo>
```

A.2.21 associationTemperatureEventInfo

```
<associationTEventInfo><!--list,List of association temperature event information.-->
  <item>
    <ruleId><!--Required; uint32,Rule ID of temperature measurement.--></ruleId>
    <channelId><!--Required; channel Id of association event.--></channelId>
    <maxTemperaturePoint><!--Required; The coordinates of the max temperature.-->
      <point><!--Required; Coordinates of the maximum temperature point.refer to Appendix
        A.2.-->
      </point>
    </maxTemperaturePoint>
    <minTemperaturePoint><!--Required; The coordinates of the min temperature.-->
      <point><!--Required; Coordinates of the maximum temperature point.refer to Appendix
        A.2.-->
      </point>
    </minTemperaturePoint>

    <ruleInfo><!--Required; Rules for temperature measurement.-->
      <ruleShapeType>
        <!--Required; ruleShapeTypeE,the type of rule.-->
      </ruleShapeType>
      <pointGroup><!--Dependent; The point coordinates of the detection polygon that
        triggers the event. Refer to Appendix A.2.-->
      </pointGroup>
    </ruleInfo>
  </item>
</associationTemperatureEventInfo>
```

A.2.22 licensePlateMatchInfo

```
<licensePlateMatchInfo>
  <licensePlateNumber><!--Required;string, car plate number.--></licensePlateNumber>
  <carOwner><!--Optional; string, Name of the car owner.--></carOwner>
  <groupId><!--Optional; string,license plate group id.--></groupId>
  <groupName><!--Dependent; string,name of plate group id,Valid when groupId exists.-->
```

```
    </groupName>
    <telephone><!--Optional; string, Car owner's phone number.--></telephone>
    <cardNumber><!--Optional; string, Car number. Only valid for IPC.--></cardNumber>
</licensePlateMatchInfo>
```

A.2.23 targetAdditionalInfo

```
<targetAdditionalInfo>
    <name><!--Optional; string, Name.--></name>
    <telephone><!--Optional; string, Car owner's phone number.--></telephone>
</targetAdditionalInfo>
```

A.2.23 personInfo

```
<personInfo><!--Required; Personnel information.-->
    <groupId><!--Required; string, face group id.--></groupId>
    <groupName><!--Dependent; string,name of plate group id,Valid when groupId exists.-->
    </groupName>
    <name><!--Optional; string, person's name.--></name>
    <sex><!--Optional; sexType, gender, value in { male, female }.--></sex>
    <telephone><!--Optional; string, phone number.--></telephone>
</personInfo>
```

A.2.24 personPicture

```
<personPicture>
    <dataType><!--Required;uint32,Picture type.0:JPG.--></dataType>
    <width><!--Required; uint32, the width of the face picture.--></width>
    <height><!-- Required; uint32, the height of the face picture.--></height>
    <pictureDataLength><!--Required; uint32, the size of the pictureData.
    </pictureDataLength>
    <pictureData>
        <!--Required; string, face picture data, the value encrypted with base64.-->
    </pictureData>
</personPicture>
```

A.2.22 faceInfo

```
<faceInfo>
  <targetId><!--Required; uint32,Face target ID.--></targetId>
  <targetImageData><!--Required;face target picture data.-->
  </targetImageData>
</faceInfo>
```

A.2.23 faceMatchInfo

```
<faceMatchInfo>
  <matchResult><!--Required; boolean,face match success or not.--></matchResult>
  <similarity><!--Required. uint32,Similarity for face comparison.--></similarity>
  <personInfo>
    <!--Dependent; Personnel information.Valid when matchResult is true.-->
  </personInfo>
  <personPicture>
    <!--Dependent;information of face picture. Valid when matchResult is true.-->
  </personPicture>
</faceMatchInfo>
```

A.3 Definition of target attribute

A.3.1 person

Attribute Name	Description	Value Set	Notes
age	The age attribute of a person	<ul style="list-style-type: none">unknowninfantyoungyouthmiddleAgedelderly	<ul style="list-style-type: none">unknownunder 7 years old7-18 years old18-40 years old41-60 years oldover 60 years old
backpack	Carrying a backpack or not	<ul style="list-style-type: none">unknownbackpacknoBackpack	<ul style="list-style-type: none">unknowncarrying a backpackno backpack

glasses	Wearing glasses or not.	<ul style="list-style-type: none"> • unknown • glasses • noGlasses 	<ul style="list-style-type: none"> • unknown • wear glasses • not wearing glasses
hat	Wearing a hat or not.	<ul style="list-style-type: none"> • unknown • hat • noHat 	<ul style="list-style-type: none"> • unknown • wearing a hat • not wearing a hat
lowerColor	Lower garment color	<ul style="list-style-type: none"> • unknown • red • orange • yellow • green • blue • cyan • purple • black • white • silver • grey • golden • brown 	enumeration of colors
lowerLength	Lower garment length	<ul style="list-style-type: none"> • unknown • shorts • trousers 	
mask	Wearing a mask or not	<ul style="list-style-type: none"> • unknown • mask • noMask 	<ul style="list-style-type: none"> • unknown • wearing a mask • not wearing a mask
orient	Person body direction	<ul style="list-style-type: none"> • unknown • face • back • side 	
sex	Gender	<ul style="list-style-type: none"> • unknown • male • female 	
shoulderBag	Carrying a shoulder bag or not	<ul style="list-style-type: none"> • unknown • shoulderBag • noShoulderBag 	<ul style="list-style-type: none"> • unknown • carrying a shoulder bag • no shoulder bag on the back
skirt	Wearing a skirt or not	<ul style="list-style-type: none"> • unknown • skirt • noSkirt 	<ul style="list-style-type: none"> • unknown • wearing a skirt • not wearing a skirt
upperColor	Upper outer garment color	<ul style="list-style-type: none"> • unknown • red • orange • yellow • green • blue • cyan • purple • black • white • silver 	same value as 'lowerColor'.

		<ul style="list-style-type: none"> • grey • golden • brown 	
upperLength	Upper outer garment length	<ul style="list-style-type: none"> • unknown • shortSleeve • longSleeve 	<ul style="list-style-type: none"> • unknown • short sleeve • long sleeve

A.3.2 car

Attribute Name	Description	Value Set	Notes
carType	Type of car	<ul style="list-style-type: none"> • unknown • saloonCar • suv • mpv • sportsCar • van • coach • schoolBus • bus • minibus • pickUpTruck • truck • specialCar 	<ul style="list-style-type: none"> • unknown • saloon car • suv • mpv • sports car • van • coach • school bus • bus • minibus • pick up truck • truck • special car
color	The color of the car	<ul style="list-style-type: none"> • unknown • red • orange • yellow • green • blue • cyan • purple • black • white • silver • grey • golden • brown 	
brand	Car brand	<ul style="list-style-type: none"> • unknown • 	<p>The name of the car factory brand.</p> <p>sush as:</p> <p>BMW</p> <p>Benz</p> <p>Buick</p> <p>Audi.....</p>

model	Vehicle model	<ul style="list-style-type: none"> • unknown • 	Format: brand_madel. sush as: Audi_A3 Audi_A4L Audi_Q5L.....
-------	---------------	--	--

A.3.3 motor

Attribute Name	Description	Value Set	Notes
motorType	Type of motor	<ul style="list-style-type: none"> • unknown • bike • electromobile • motor • tricycle 	

A.3.4 face

Attribute Name	Description	Value Set	Notes
age	The age attribute of a person	<ul style="list-style-type: none"> • unknown • infant • young • youth • middleAged • elderly 	<ul style="list-style-type: none"> • unknown • under 7 years old • 7-18 years old • 18-40 years old • 41-60 years old • over 60 years old
mask	Wearing a mask or not	<ul style="list-style-type: none"> • unknown • mask • noMask 	<ul style="list-style-type: none"> • unknown • wearing a mask • not wearing a mask
glasses	Wearing glasses or not.	<ul style="list-style-type: none"> • unknown • glasses • noGlasses 	<ul style="list-style-type: none"> • unknown • wear glasses • not wearing glasses
sex	Gender	<ul style="list-style-type: none"> • unknown • male • female 	

A.3.5 license plate

Attribute Name	Description	Value Set	Notes
licensePlateNumber	Pate number	string	
color	The color of the car	<ul style="list-style-type: none"> • unknown • red 	

		<ul style="list-style-type: none"> • orange • yellow • green • blue • cyan • purple • black • white • silver • grey • golden • brown 	
--	--	--	--

Annex B

B.1 Change Log

Date	Version	Note
2024-12-30	2.0.0	1.Redefine the protocol.
2025-11-30	2.1.0	1.Add "License Plate Detection". 2. Add "deviceAlarmStatus". 3.New AI events supported: Personnel Tailgating, Personnel Intrusion, Absence Detection. Applicable to AI-BOX product. 4. Change the attribute of the element 'sourceDataInfo' in the FaceDetection interface to 'Optional'. 5. The name of the element 'lefttorbotton' in tripwireDirectionE has been changed to lefttorbottom. 6.Add new alarm "videoLossAlarm" and "wifiDisconnect" for DVR.