



Device Network SDK (Queue Management)

Developer Guide

Legal Information

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE DOCUMENT IS PROVIDED "AS IS" AND "WITH ALL FAULTS AND ERRORS". OUR COMPANY MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IN NO EVENT WILL OUR COMPANY BE LIABLE FOR ANY SPECIAL, CONSEQUENTIAL, INCIDENTAL, OR INDIRECT DAMAGES, INCLUDING, AMONG OTHERS, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION OR LOSS OF DATA, CORRUPTION OF SYSTEMS, OR LOSS OF DOCUMENTATION, WHETHER BASED ON BREACH OF CONTRACT, TORT (INCLUDING NEGLIGENCE), OR OTHERWISE, IN CONNECTION WITH THE USE OF THE DOCUMENT, EVEN IF OUR COMPANY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR LOSS.

Contents

| | |
|---|-----------|
| Chapter 1 Overview | 1 |
| 1.1 Introduction | 1 |
| 1.2 Update History | 2 |
| Chapter 2 Typical Applications | 4 |
| 2.1 Alarm and Event Receiving | 4 |
| 2.1.1 Configure Queue Management Alarm | 4 |
| 2.1.2 Receive Alarm/Event in Arming Mode | 8 |
| 2.1.3 Receive Alarm/Event in Listening Mode | 11 |
| 2.2 Search Queue Management Report or Data | 14 |
| Chapter 3 API Reference | 20 |
| 3.1 NET_DVR_Cleanup | 20 |
| 3.2 NET_DVR_GetErrorMsg | 20 |
| 3.3 NET_DVR_GetLastError | 21 |
| 3.4 NET_DVR_Init | 21 |
| 3.5 NET_DVR_Login_V40 | 21 |
| 3.5.1 fLoginResultCallBack | 22 |
| 3.6 NET_DVR_Logout | 23 |
| 3.7 NET_DVR_SetSDKInitCfg | 23 |
| 3.8 NET_DVR_GetSTDAbility | 25 |
| 3.9 NET_DVR_GetSTDConfig | 25 |
| 3.10 NET_DVR_MIME_UNIT | 26 |
| 3.11 NET_DVR_SetSTDConfig | 27 |
| 3.12 NET_DVR_STDXMLConfig | 28 |
| 3.13 NET_DVR_CloseAlarmChan_V30 | 29 |
| 3.14 NET_DVR_GetDVRConfig | 29 |
| 3.15 NET_DVR_SetDVRConfig | 31 |

| | |
|--|-----------|
| 3.16 NET_DVR_SetDVRMessageCallBack_V50 | 32 |
| 3.16.1 MSGCallBack | 39 |
| 3.17 NET_DVR_SetupAlarmChan_V50 | 40 |
| 3.18 NET_DVR_StartListen_V30 | 41 |
| 3.19 NET_DVR_StopListen_V30 | 42 |
| Appendix A. Data Structure | 87 |
| A.1 DATE_TIME | 87 |
| A.2 NET_DVR_DEVICEINFO_V30 | 87 |
| A.3 NET_DVR_DEVICEINFO_V40 | 91 |
| A.4 NET_DVR_INIT_CFG_ABILITY | 94 |
| A.5 NET_DVR_LOCAL_SDK_PATH | 95 |
| A.6 NET_DVR_USER_LOGIN_INFO | 95 |
| A.7 NET_SDK_CALLBACK_STATUS_NORMAL | 96 |
| A.8 NET_VCA_RECT | 97 |
| A.9 NET_DVR_CRUISECHAN_INFO | 97 |
| A.10 NET_DVR_EVENT_SCHEDULE | 98 |
| A.11 NET_DVR_EVENT_TRIGGER | 98 |
| A.12 NET_DVR_HANDLEEXCEPTION_V41 | 99 |
| A.13 NET_DVR_MIME_UNIT | 100 |
| A.14 NET_DVR_PRESETCHAN_INFO | 100 |
| A.15 NET_DVR_PTZTRACKCHAN_INFO | 101 |
| A.16 NET_DVR_SCHEDTIME | 101 |
| A.17 NET_DVR_STD_ABILITY | 101 |
| A.18 NET_DVR_STD_CONFIG | 102 |
| A.19 NET_DVR_XML_CONFIG_INPUT | 103 |
| A.20 NET_DVR_XML_CONFIG_OUTPUT | 104 |
| A.21 NET_ALARM_CVR_SUBINFO_UNION | 105 |
| A.22 NET_ALARM_RECORD_EXCEPTION | 105 |

| | |
|---|-----------|
| A.23 NET_ALARM_RECORDFILE_LOSS | 106 |
| A.24 NET_ALARM_RESOURCE_USAGE | 107 |
| A.25 NET_ALARM_STREAM_EXCEPTION | 107 |
| A.26 NET_DVR_ALARMER | 107 |
| A.27 NET_DVR_ALARMINFO_DEV | 108 |
| A.28 NET_DVR_ALARMINFO_DEV_V40 | 109 |
| A.29 NET_DVR_ALARMINFO_V30 | 110 |
| A.30 NET_DVR_ALARMINFO_V40 | 111 |
| A.31 NET_DVR_ALARM_FIXED_HEADER | 112 |
| A.32 NET_DVR_ALARM_ISAPI_INFO | 116 |
| A.33 NET_DVR_ALARM_ISAPI_PICDATA | 117 |
| A.34 NET_DVR_ETHERNET_V30 | 117 |
| A.35 NET_DVR_IPADDR_UNION | 118 |
| A.36 NET_DVR_NETCFG_V50 | 118 |
| A.37 NET_DVR_PPPOECFG | 119 |
| A.38 NET_DVR_SETUPALARM_PARAM_V50 | 120 |
| A.39 NET_DVR_TIME | 123 |
| A.40 NET_DVR_TIME_EX | 123 |
| Chapter 4 Request URL | 43 |
| 4.1 /ISAPI/Intelligent/channels/<ID>/capabilities | 43 |
| 4.2 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/capabilities?format=json | 43 |
| 4.3 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingFile?format=json | 44 |
| 4.4 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingResult?format=json | 44 |
| 4.5 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueTimeFile?format=json ... | 45 |
| 4.6 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueTimeResult?format=json | 46 |
| 4.7 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/rawDataFile?format=json | 46 |

| | |
|---|------------|
| 4.8 /ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json | 47 |
| Chapter 5 Request and Response Message | 48 |
| 5.1 JSON_EventNotificationAlert_Alarm/EventInfo | 48 |
| 5.2 JSON_EventNotificationAlert_PeopleQueuingUpAlarmMsg | 49 |
| 5.3 JSON_EventNotificationAlert_RealTimeQueueManagementDataMsg | 51 |
| 5.4 JSON_EventNotificationAlert_WaitingTimeDetectionAlarmMsg | 53 |
| 5.5 JSON_PersonQueueCountingDescription | 56 |
| 5.6 JSON_PersonQueueCountingFileResult | 57 |
| 5.7 JSON_PersonQueueCountingResult | 58 |
| 5.8 JSON_PersonQueueDetectionInfo | 59 |
| 5.9 JSON_PersonQueueDetectionInfoCap | 62 |
| 5.10 JSON_PersonQueueRawDataFileDescription | 67 |
| 5.11 JSON_PersonQueueRawDataFileResult | 68 |
| 5.12 JSON_PersonQueueTimeDescription | 68 |
| 5.13 JSON_PersonQueueTimeFileResult | 69 |
| 5.14 JSON_PersonQueueTimeResult | 70 |
| 5.15 JSON_ResponseStatus | 73 |
| 5.16 XML_Cap_Schedule | 74 |
| 5.17 XML_EventNotificationAlert_AlarmEventInfo | 75 |
| 5.18 XML_EventTriggersCap | 76 |
| 5.19 XML_EventTriggerCapType | 79 |
| 5.20 XML_IntelliCap | 81 |
| 5.21 XML_ResponseStatus | 86 |
| Appendix B. Appendixes | 125 |
| B.1 Device Network SDK Errors | 125 |
| B.2 Response Codes of Text Protocol | 168 |

Chapter 1 Overview

This manual provides the integration methods and flows based on HCNetSDK for queue management applications.

1.1 Introduction

Queue management is applicable to analyze the queuing status of public places (e.g., market, ticket office), it provides the functions of people queuing-up detection and waiting time detection. You can configure these two kinds of alarms for receiving the alarm information in the third-party platform or system, and search for the detection reports or data.

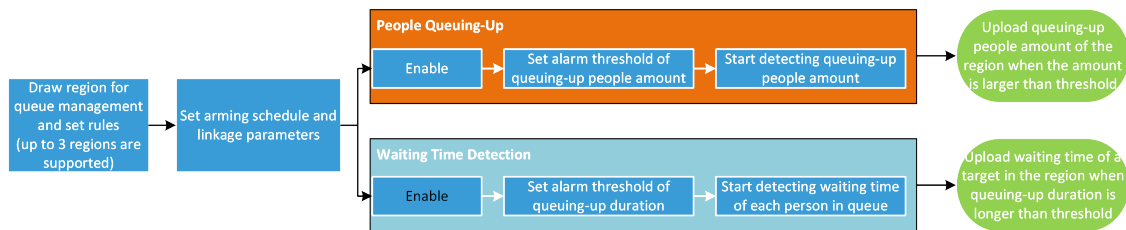


Figure 1-1 Queue Management Overall Flow

People Queuing-Up

The people queuing-up is to detect the queuing-up people amount in multiple rule regions. When the queuing-up people amount is larger than the configured amount threshold, the alarm is triggered and uploaded by the device. For example, if the configured queuing-up people amount threshold is 6 in the Region 1 of the following figure, when the number of persons is larger than or equal to 6, the alarm will be triggered and uploaded according to the configured uploading time interval.

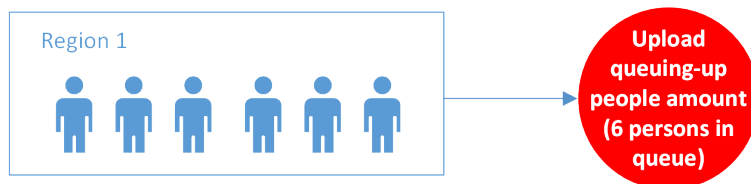


Figure 1-2 People Queuing-Up

Waiting Time Detection

The waiting time detection is to detect the waiting time duration of a target queuing-up person in multiple rule regions. When the waiting time of a target person is longer than the configured time duration threshold, the alarm will be triggered and uploaded by device. For example, if the configured queuing-up duration threshold is 30 minutes in the Region 2 of the following figure,

when the queuing-up duration of target person A is longer than or equal to 30 minutes, the alarm will be triggered and uploaded according to the configured uploading time interval.

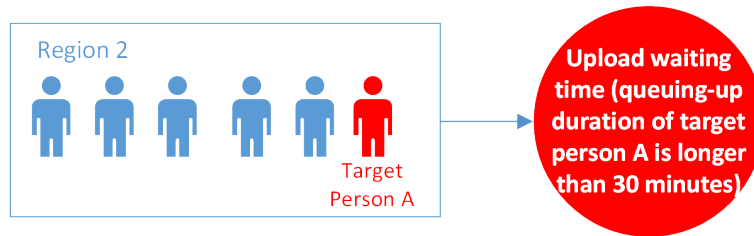


Figure 1-3 Waiting Time Detection

1.2 Update History

Summary of Changes in Version 6.0.2.36_May, 2019

1. Extended queue management configuration capability message ***JSON_PersonQueueDetectionInfoCap*** (related API: ***NET_DVR_STDXMLConfig*** ; related URL: ***/ISAPI/Intelligent/channels/<ID>/personQueueDetection/capabilities?format=json***):
added a node "isSupportAlgParam" (whether supports configuring advanced parameters for algorithm library);
added three sub nodes, i.e., "countTriggerType" (triggering type of people queuing-up alarm), "maxAlarmCount" (maximum number of queuing-up persons to trigger alarm), and "durationTime" (queuing-up status (number of queuing-up persons) keeping duration), to the node "PersonQueueCounting" (people queuing-up parameters);
added a sub node "timeTriggerType" (triggering type of waiting time detection alarm) to the node "PersonQueueTime" (waiting time detection parameters);
added a sub node "isSupportAverageTime" (whether supports calculating average waiting time duration in people queuing-up statistics) to the node "WaitTimeListCap" (waiting time information).
2. Extended configuration parameter message ***JSON_PersonQueueDetectionInfo*** of queue management (related API: ***NET_DVR_STDXMLConfig*** ; related URL: ***/ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json***):
added three sub nodes, i.e., "countTriggerType" (triggering type of people queuing-up alarm), "maxAlarmCount" (maximum number of queuing-up persons to trigger alarm), and "durationTime" (queuing-up status (number of queuing-up persons) keeping duration), to the node "PersonQueueCounting" (people queuing-up parameters);
added a sub node "timeTriggerType" (triggering type of waiting time detection alarm) to the node "PersonQueueTime" (waiting time detection parameters).
3. Extended waiting time detection alarm message with binary picture data or picture URL ():
added a sub node "timeTriggerType" (triggering type of waiting time detection alarm) to the node "rule" (rule properties).
4. Extended people queuing-up alarm message with binary picture data or picture URL ():

added two sub nodes "countTriggerType" (triggering type of people queuing-up alarm) and "durationTime" (queuing-up status (number of queuing-up persons) keeping duration) to the node "rule" (rule properties).

Summary of Changes in Version 6.0.2.X_Mar., 2019

New document.

Chapter 2 Typical Applications

2.1 Alarm and Event Receiving

The alarm/event information from the device can be received in third-party platform or system when the alarms are triggered or event occurred. Two modes are available for receiving alarms, including arming mode and listening mode.

Arming Mode

The third-party platform connects to device automatically, when the alarm is triggered, the platform sends alarm uploading command to the device, and then the device will upload the alarm to the platform.

Listening Mode

When alarm is triggered, the device automatically uploads the alarm, and then the third-party platform receives the uploaded alarm via the configured listening host (listening address and port should be configured). This mode is applicable for multiple devices uploading alarm/event information to one third-party platform without logging in to devices, and the restart of devices will not affect the alarm/event uploading. But a device can only support the configuration of one or two listening addresses and ports.

2.1.1 Configure Queue Management Alarm

Queue management alarm includes people queuing-up alarm, waiting time detection alarm, and real-time queue management alarm. You can set the alarm parameters to trigger the people queuing-up alarm when the queuing-up people amount in a region is larger than the configured threshold, trigger the waiting time detection alarm when the waiting time of a queue stays a certain length is longer than the configured threshold, and trigger real-time queue management alarm when queuing-up people amount changed or waiting time duration is longer than threshold.

Before You Start

- Make sure you have called **NET DVR Init** to initialize the development environment.
- Make sure you have called **NET DVR Login_V40** to log in to device.

Steps



Figure 2-1 Programming Flow of Configuring Queue Management Alarm

Note

The queue management parameters, alarm linkage actions, and arming schedules can be directly configured by logging in to device via web browser.

1. Call ***NET_DVR_STDXMLConfig*** to pass through the request URL: GET ***/ISAPI/Intelligent/channels/<ID>/capabilities*** for getting intelligent channel capability to check if the device supports queue management.

The intelligent channel capability is returned in the message ***XML_IntelliCap*** by output pointer (***lpOutputParam***).

If the node ***<isSupportPersonQueueDetection>*** is returned in the capability message, it indicates that queue management function is supported and you can perform the following steps.

Otherwise, this function is not supported, please end this task.

2. Optional: Call ***NET_DVR_STDXMLConfig*** to pass through the request URL: GET ***/ISAPI/Intelligent/channels/<ID>/personQueueDetection/capabilities?format=json*** for getting the queue management capability to check the supported configuration parameters.

The queue management capability is returned in the message ***JSON_PersonQueueDetectionInfoCap*** by output pointer (***lpOutputParam***).

3. Optional: Call ***NET_DVR_STDXMLConfig*** to pass through the request URL: GET ***/ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json*** for getting default or configured queue management parameters for reference.

The queue management parameters are returned in the message ***JSON_PersonQueueDetectionInfo*** by the output pointer (***lpOutputParam***).

4. Call ***NET_DVR_STDXMLConfig*** to pass through the request URL: PUT ***/ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json*** and set the input pointer (***lpInputParam***) to the message ***JSON_PersonQueueDetectionInfo*** for setting queue management parameters.
5. Configure linkage action for queue management alarm.

- 1) Call ***NET_DVR_GetSTDAbility*** with ***NET_DVR_GET_EVENT_TRIGGERS_CAPABILITIES*** (command No.: 3501) to get alarm linkage capability of device.

The alarm linkage capability is returned in the message ***XML_EventTriggersCap*** by the output buffer (***lpOutBuffer***) in structure ***NET_DVR_STD_ABILITY***.

- 2) Optional: Call ***NET_DVR_GetSTDConfig*** with ***NET_DVR_GET_PERSONQUEUE_TRIGGER*** (command No.: 4406) to get the default or configured linkage actions of queue management alarm for reference.

The linkage actions are returned in the structure ***NET_DVR_EVENT_TRIGGER*** by the output buffer (***lpOutBuffer***) in structure ***NET_DVR_STD_CONFIG***.

- 3) Call ***NET_DVR_SetSTDConfig*** with ***NET_DVR_SET_PERSONQUEUE_TRIGGER***

(command No.: 4407), and then set condition buffer (**IpCondBuffer**) and input buffer (**IpInBuffer**) in structure **NET_DVR_STD_CONFIG** to 4-byte channel No. and **NET_DVR_EVENT_TRIGGER**, respectively, for setting the linkage actions of queue management alarm.



Note

To receive alarms in arming mode via platform, the linkage action must be set to "center" (upload to alarm center).

6. Configure arming schedule of queue management alarm.

- 1) Call **NET_DVR_GetSTDAbility** with

NET_DVR_GET_PERSONQUEUE_SCHEDULE_CAPABILITIES

(command No.: 4408) to get the arming schedule capability of queue management device.

The arming schedule capability is returned in the message **XML_Cap_Schedule** by the output buffer (**IpOutBuffer**) of structure **NET_DVR_STD_ABILITY**.

- 2) Optional: Call **NET_DVR_GetSTDConfig** with

NET_DVR_GET_PERSONQUEUE_SCHEDULE

(command No.: 4409) and set condition buffer (**IpCondBuffer**) in structure

NET_DVR_STD_CONFIG to 4-byte channel No. for getting the default or configured arming schedule of queue management for reference.

The arming schedule parameters are returned in the structure **NET_DVR_EVENT_SCHEDULE** by the output buffer (**IpOutBuffer**) in structure **NET_DVR_STD_CONFIG**.

- 3) Call **NET_DVR_SetSTDConfig** with

NET_DVR_SET_PERSONQUEUE_SCHEDULE

(command No.: 4410), and then set condition buffer (**IpCondBuffer**) and input buffer

(**IpInBuffer**) in structure **NET_DVR_STD_CONFIG** to 4-byte channel No. and

NET_DVR_EVENT_SCHEDULE, respectively, for setting arming schedule of queue management.

7. Optional: Set the uploading message type (**ICommand**) in alarm callback function to "COMM_VCA_ALARM" (macro definition value: 0x4993) for receiving queue management alarms, i.e., people queuing-up alarm, waiting time detection alarm, and real-time queue management alarm, in arming mode (refer to **Receive Alarm/Event in Arming Mode**) or listening mode (refer to **Receive Alarm/Event in Listening Mode**).

The people queuing-up alarm is uploaded in the message

JSON_EventNotificationAlert_PeopleQueuingUpAlarmMsg; the waiting time detection alarm is

uploaded in the message **JSON_EventNotificationAlert_WaitingTimeDetectionAlarmMsg**; and

the real-time queue management alarm is uploaded in the message

JSON_EventNotificationAlert_RealTimeQueueManagementDataMsg.

What to do next

Call **NET_DVR_Logout** and **NET_DVR_Cleanup** to log out from device and release the resources.

2.1.2 Receive Alarm/Event in Arming Mode

When the alarm is triggered or the event occurred, the secondarily developed third-party platform can automatically connect and send alarm/event uploading command to the device, and then the device uploads the alarm/event information to the platform for receiving.

Before You Start

- Make sure you have called **NET_DVR_Init** to initialize the development environment.
- Make sure you have called **NET_DVR_Login_V40** to log in to the device.
- Make sure you have configured the alarm/event parameters, refer to the typical alarm/event configurations for details.

Steps

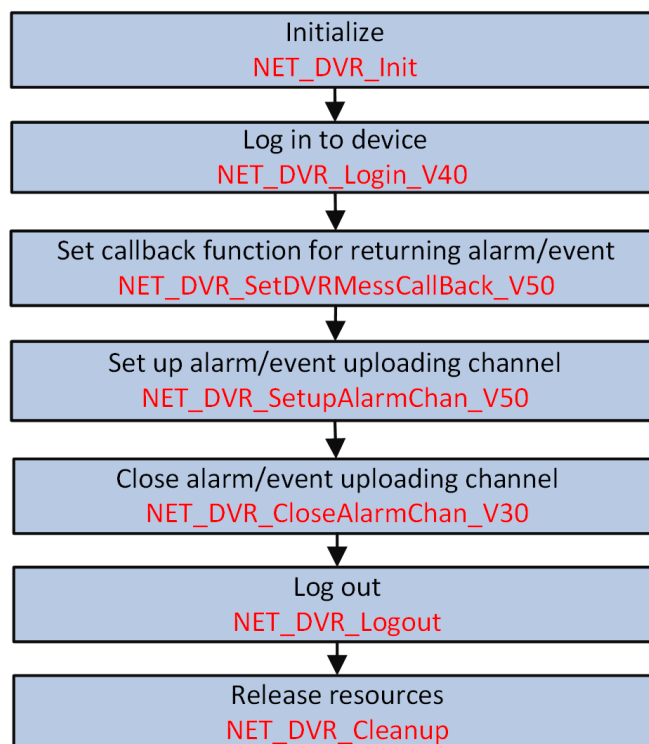


Figure 2-2 Programming Flow of Receiving Alarm/Event in Arming Mode

1. Call **NET_DVR_SetDVRMessageCallBack_V50** to set callback function for returning alarm/event information.



Note

- If the configured alarm is triggered or event occurred, the alarm/event information will be uploaded by device and returned in the callback function. You can view the alarm/event and do some processing operations.
- For the integration via device network SDK (HCNetSDK), to receive different types of alarm/event information, the parameter **lCommand** (data type to be uploaded) in the configured callback function should be different (refer to the typical alarm/event configurations). For the integration via text protocol, the **lCommand** should be set to "COMM_ISAPI_ALARM" (command No.: 0x6009) and the input parameter **pAlarmInfo** in the callback function **MSGCallBack** should be set to **NET_DVR_ALARM_ISAPI_INFO**.

2. Call **NET_DVR_SetupAlarmChan V50** to set up uploading channel.

3. Call **NET_DVR_CloseAlarmChan V30** to close uploading channel and stop receiving alarm or event information.

Example

Sample Code of Receiving Alarm or Event in Arming Mode

```
#include <stdio.h>
#include <iostream>
#include "Windows.h"
#include "HCNetSDK.h"
using namespace std;

void main() {
    //-----
    // Initialize
    NET_DVR_Init();
    //Set connection time and reconnection time
    NET_DVR_SetConnectTime(2000, 1);
    NET_DVR_SetReconnect(10000, true);
    //-----
    // Log in to device
    LONG lUserID;
    //Login parameters, including device IP address, user name, password, and so
on.
    NET_DVR_USER_LOGIN_INFO struLoginInfo = {0};
    struLoginInfo.bUseAsynLogin = 0; //Synchronous login mode
    strcpy(struLoginInfo.sDeviceAddress, "192.0.0.64"); //Device IP address
    struLoginInfo.wPort = 8000; //Service port No.
    strcpy(struLoginInfo.sUserName, "admin"); //User name
    strcpy(struLoginInfo.sPassword, "abcd1234"); //Password
    //Device information, output parameter
    NET_DVR_DEVICEINFO_V40 struDeviceInfoV40 = {0};
    lUserID = NET_DVR_Login_V40(&struLoginInfo, &struDeviceInfoV40);
    if (lUserID < 0)
    {
        printf("Login failed, error code: %d\n", NET_DVR_GetLastError());
        NET_DVR_Cleanup();
        return;
    }
}
```

```
}

//Set alarm callback function
NET_DVR_SetDVRMessageCallBack_V50(0, MessageCallbackNo1, NULL);
NET_DVR_SetDVRMessageCallBack_V50(1, MessageCallbackNo2, NULL);

//Enable arming
NET_DVR_SETUPALARM_PARAM_V50 struSetupParamV50={0};
struSetupParamV50.dwSize=sizeof(NET_DVR_SETUPALARM_PARAM_V50);
//Alarm category to be uploaded
struSetupParamV50.byAlarmInfoType=1;
//Arming level
struSetupParamV50.byLevel=1;

char szSubscribe[1024] = {0};
//The following code is for alarm subscription (subscribe all)
memcpy(szSubscribe, "<SubscribeEvent version=\"2.0\" xmlns=\"http://
www.isapi.org/ver20/XMLSchema\">\r\n<eventMode>all</eventMode>\r\n", 1024);
LONG lHandle = -1;
if (0 == strlen(szSubscribe))
{
    //Arm
    lHandle = NET_DVR_SetupAlarmChan_V50(lUserID, &struSetupParamV50, NULL,
strlen(szSubscribe));
}
else
{
    //Subscribe
    lHandle = NET_DVR_SetupAlarmChan_V50(lUserID, &struSetupParamV50,
szSubscribe, strlen(szSubscribe));
}

if (lHandle < 0)
{
    printf("NET_DVR_SetupAlarmChan_V50 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(lUserID);
    NET_DVR_Cleanup();
    return;
}

Sleep(20000);
//Disarm the uploading channel
if (!NET_DVR_CloseAlarmChan_V30(lHandle))
{
    printf("NET_DVR_CloseAlarmChan_V30 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(lUserID);
    NET_DVR_Cleanup();
    return;
}

//Log out
NET_DVR_Logout(lUserID);
```



```
//Release resources
NET_DVR_Cleanup();
return;
}
```

What to do next

Call **NET_DVR_Logout** and **NET_DVR_Cleanup** to log out and release resources.

2.1.3 Receive Alarm/Event in Listening Mode

When alarm is triggered or event occurred, the device uploads the alarm/event information automatically, so you can configure the listening address and port for listening and receiving the alarm/event in the secondarily developed third-part platform.

Before You Start

- Make sure you have called **NET_DVR_Init** to initialize the development environment.
- Make sure you have configured the alarm/event parameters, refer to the typical alarm/event configurations for details.

Steps

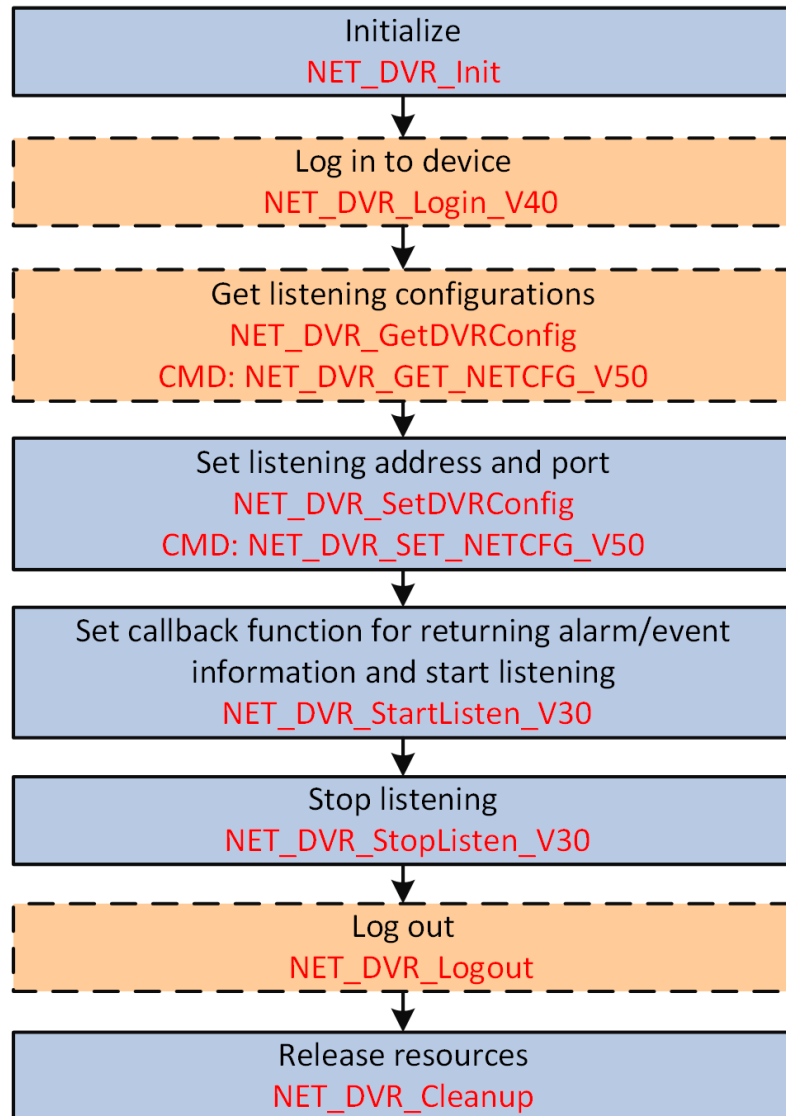


Figure 2-3 Programming Flow of Receiving Alarm/Event in Listening Mode

1. **Optional:** Call NET_DVR_Login_V40 to log in to device.
2. **Optional:** Call NET_DVR_GetDVRConfig with "NET_DVR_GET_NETCFG_V50" (command No.: 1015) to get the existing listening configurations (i.e., listening address and port) for reference. The listening parameters are retruned in the structure NET_DVR_NETCFG_V50 by the output parameter pointer lpOutBuffer.
3. Call NET_DVR_SetDVRConfig with "NET_DVR_SET_NETCFG_V50" (command No.: 1016) and specify the input parameter pointer lpInBuffer to the structure NET_DVR_NETCFG_V50 for setting the listening address and port.
4. Call NET_DVR_StartListen_V30 to set callback function for returning alarm/event information and start the listening.

Note

For the integration via device network SDK (HCNetSDK), to receive different types of alarm/event information, the parameter **lCommand** (data type to be uploaded) in the configured callback function should be different (refer to the typical alarm/event configurations). For the integration via text protocol, the **lCommand** should be set to "COMM_ISAPI_ALARM" and the input parameter **pAlarmInfo** in the callback function **MSGCallBack** should be set to **NET_DVR_ALARM_ISAPI_INFO**.

The alarm/event information is automatically uploaded by the device when the configured alarm is triggered or event occurred, and the third-party platform or system gets the alarm/event information from the configured callback function.

5. Call **NET_DVR_StopListen_V30** to stop listening and receiving alarm or event information.

Example

Sample Code of Receiving Alarm/Event in Listening Mode

```
#include <stdio.h>
#include <iostream>
#include "Windows.h"
#include "HCNetSDK.h"
using namespace std;
void main() {
    //-----
    // Initialize
    NET_DVR_Init();
    //Set connection time and reconnection time
    NET_DVR_SetConnectTime(2000, 1);
    NET_DVR_SetReconnect(10000, true);
    //-----
    // Log in to device
    LONG lUserID;
    NET_DVR_DEVICEINFO_V30 struDeviceInfo;
    lUserID = NET_DVR_Login_V30("172.0.0.100", 8000, "admin", "12345",
&struDeviceInfo);
    if (lUserID < 0)
    {
        printf("Login error, %d\n", NET_DVR_GetLastError());
        NET_DVR_Cleanup();
        return;
    }
    //Enable listening
    LONG lHandle;
    lHandle = NET_DVR_StartListen_V30(NULL, 7200, MessageCallback, NULL);
    if (lHandle < 0)
    {
        printf("NET_DVR_StartListen_V30 error, %d\n", NET_DVR_GetLastError());
        NET_DVR_Logout(lUserID);
        NET_DVR_Cleanup();
        return;
    }
}
```

```
Sleep(5000);
//Disable listening
if (!NET_DVR_StopListen_V30(lHandle))
{
    printf("NET_DVR_StopListen_V30 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(lUserID);
    NET_DVR_Cleanup();
    return;
}
//Log out
NET_DVR_Logout(lUserID);
//Release SDK resource
NET_DVR_Cleanup();
return;
}
```

What to do next

Call **NET_DVR_Logout** (if logged in) and **NET_DVR_Cleanup** to log out and release resources.

2.2 Search Queue Management Report or Data

When the queue management alarm is triggered and uploaded to third-party platform/system, you can search for the reports of people queuing-up and waiting time detection, and get the reports in different formats (i.e., report details and report URL). You can also directly get the file URL of unprocessed data (raw data).

Search People Queuing-Up Report

Get Report Details

Call **NET_DVR_STDXMLConfig** to pass through the request URL: POST **/ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingResult?format=json** .

The report details is returned in the message **JSON_PersonQueueCountingResult** by output pointer (**lpOutputParam**).

Get Report URL

Call **NET_DVR_STDXMLConfig** to pass through the request URL: POST **/ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingFile?format=json** and set the input pointer (**lpInputParam**) to the message **JSON_PersonQueueCountingDescription** .

The report URL is returned in the message **JSON_PersonQueueCountingFileResult** by output pointer (**lpOutputParam**). You can download the report from the URL.

Search Waiting Time Detection Report

Get Report Details

Call **NET_DVR_STDXMLConfig** to pass through the request URL: POST .

The report details is returned in the message **JSON_PersonQueueTimeResult** by output pointer (**IpOutputParam**).

Get Report URL

Call **NET_DVR_STDXMLConfig** to pass through the request URL: POST and set the input pointer (**IpInputParam**) to the message **JSON_PersonQueueTimeDescription** .

The report URL is returned in the message **JSON_PersonQueueTimeFileResult** by output pointer (**IpOutputParam**). You can download the report from the URL.

Search Queue Management Raw Data

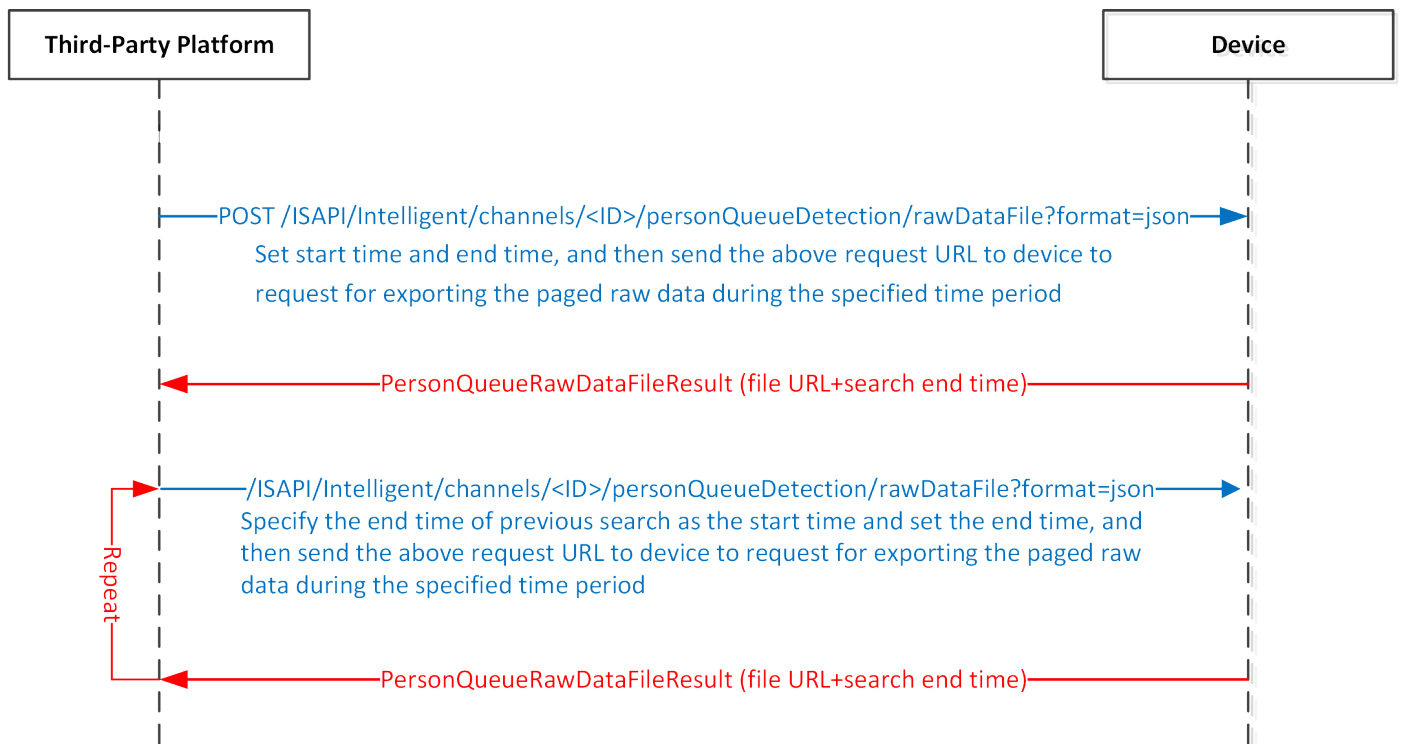


Figure 2-4 Interaction Process of Raw Data Search

As the raw data is paged, if you want to get all raw data, you should set search time period and repeatedly call **NET_DVR_STDXMLConfig** to pass through the request URL: POST and set the input pointer (**IpInputParam**) to the message **JSON_PersonQueueRawDataFileDescription** .

The searched raw data is returned in the message **JSON_PersonQueueRawDataFileResult** by output pointer (**IpOutputParam**).

Note

- When setting search time period, the previous search end time returned by device should be specified as the start time of next search.
- The raw data will be paged and returned by file URL, and the records are sorted from earliest to latest in the file.
- Up to 4000 records can be returned after searching for once.

Example

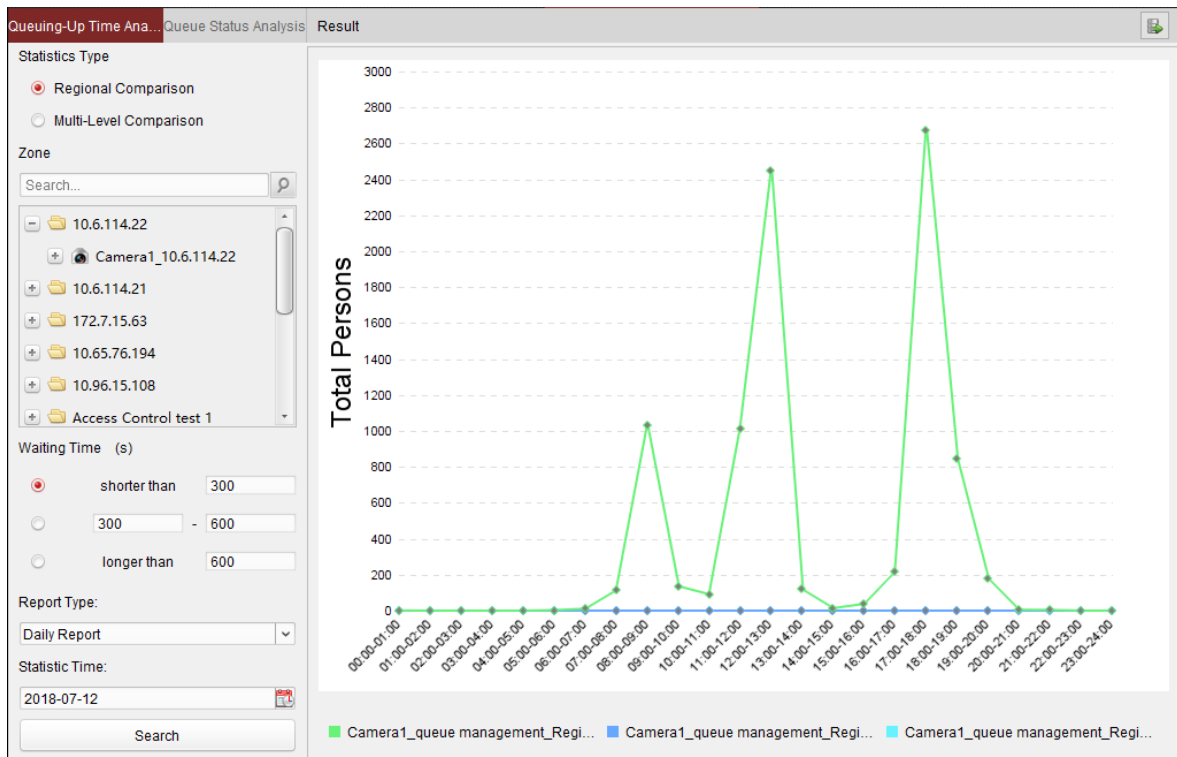


Figure 2-5 Compare Queuing-Up People Amount for Different Regions

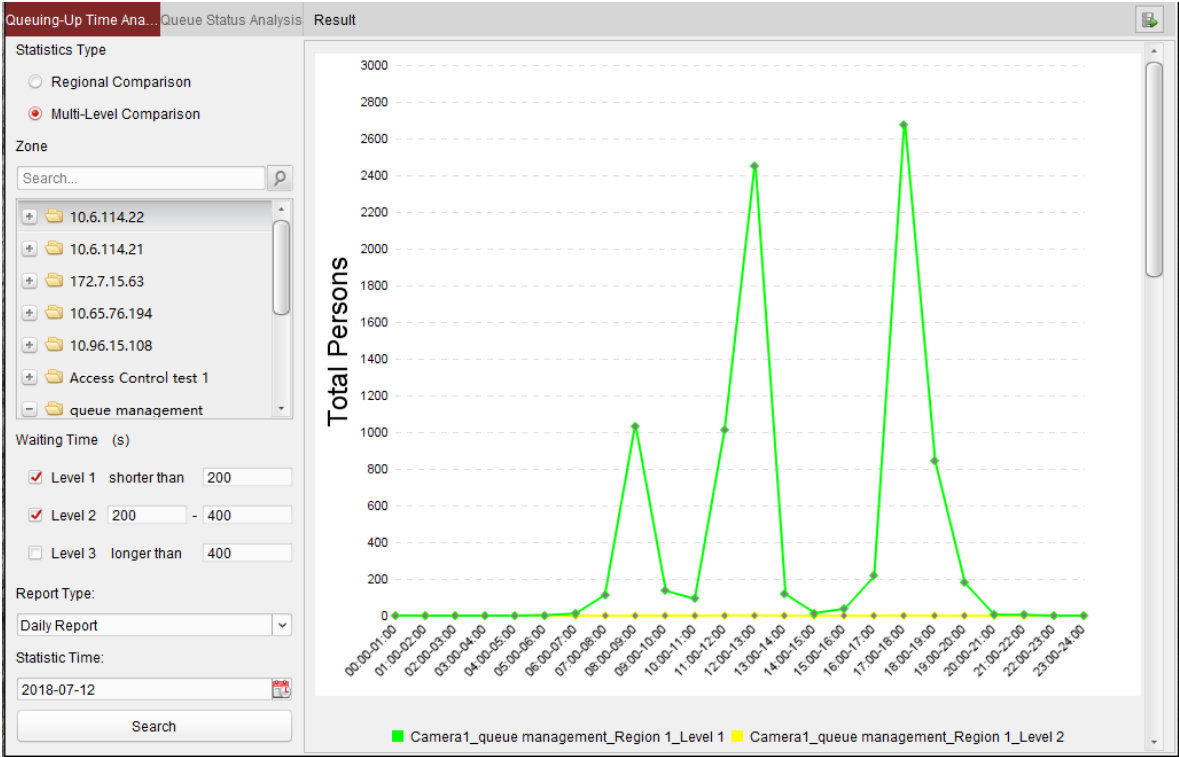


Figure 2-6 Compare Queuing-Up People Amount for Different Waiting Times

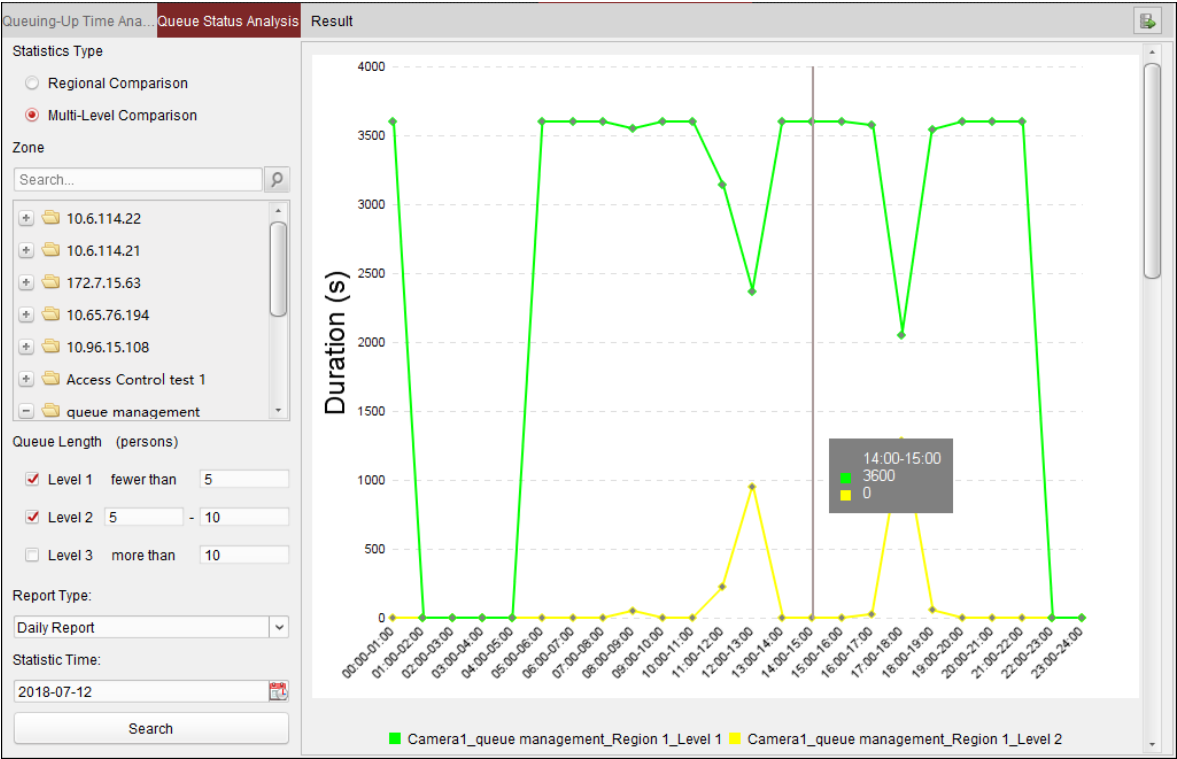


Figure 2-7 Compare Queuing-Up Duration for Different Regions

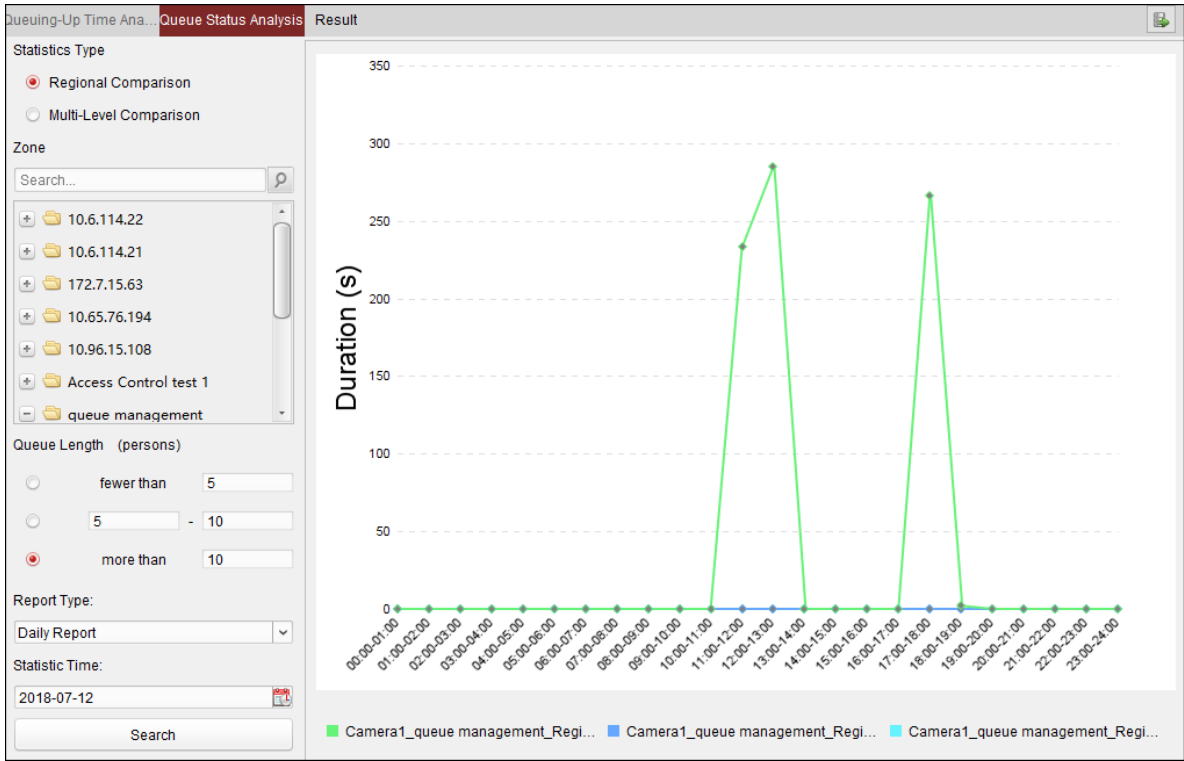


Figure 2-8 Compare Queuing-Up Duration for Different Queue Lengths

Chapter 3 API Reference

3.1 NET_DVR_Cleanup

Release the resources after the program is ended.

API Definition

```
BOOL NET_DVR_Cleanup(  
);
```

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call [***NET_DVR_GetLastError***](#) to get the error code.

The available error codes may be returned by this API are 0 and 3. See details in [***Device Network SDK Errors***](#).

Remarks

- When calling this API, you cannot call other APIs at the same time.
- [***NET_DVR_Init***](#) and this API should be called by pair. That is, once the `NET_DVR_Init` is called, you should call `NET_DVR_Cleanup` to release the resources when exiting the program.

3.2 NET_DVR_GetErrorMsg

Return the error information of the last operation.

API Definition

```
char *NET_DVR_GetErrorMsg(  
    LONG    *pErrorNo  
);
```

Parameters

pErrorNo

[OUT] Error code pointer.

Return Values

The return values are the pointers of error information, see [***Device Network SDK Errors***](#) for details.

Remarks

You can call [***NET_DVR_GetLastError***](#) to get the error codes.

3.3 NET_DVR_GetLastError

Return the error code of the last operation.

API Definition

```
DWORD NET_DVR_GetLastError(  
);
```

Return Values

The return values are error codes, see [Device Network SDK Errors](#) for details.

Remarks

You can also call [NET_DVR_GetErrorMsg](#) to directly get the error information.

3.4 NET_DVR_Init

Initialize the programming environment before calling other APIs.

API Definition

```
BOOL NET_DVR_Init(  
);
```

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call [NET_DVR_GetLastError](#) to get the error code.

The available error codes of this API are 0, 41, and 53. See details in [Device Network SDK Errors](#).

Remarks

Before initializing, you can call [NET_DVR_SetSDKInitCfg](#) to set the initialization parameters, such as supported capabilities, loading path of component libraries (only supported by Linux system), and so on.

See Also

[NET_DVR_Cleanup](#)

3.5 NET_DVR_Login_V40

Log in to the device (supports asynchronous login).

API Definition

```
LONG NET_DVR_Login_V40 (
    NET_DVR_USER_LOGIN_INFO  pLoginInfo,
    NET_DVR_DEVICEINFO_V40   lpDeviceInfo
);
```

Parameters

pLoginInfo

[IN] Login parameters, including device address, user name, password, and so on. See details in the structure **NET_DVR_USER_LOGIN_INFO**.

lpDeviceInfo

[OUT] Device information. See details in the structure **NET_DVR_DEVICEINFO_V40**.

Return Values

- For asynchronous login, the callback function (**fLoginResultCallBack**) configured in the structure (**NET_DVR_USER_LOGIN_INFO**) returns the asynchronous login status, user ID and device information.
- For synchronous login, this API returns -1 for logging failed, and returns other values for the returned user IDs. The user ID is unique, and it helps to realize the further device operations.
- If -1 is returned, you can call **NET_DVR_GetLastError** to get the error code.

Remarks

- When **bUseAsynLogin** in **pLoginInfo** is 0, it indicates that login is in synchronous mode; when **bUseAsynLogin** in **pLoginInfo** is 1, it indicates that login is in asynchronous mode.
- Up to 2048 users are allowed to log in to HCNetSDK at same time, and the values of returned **UserID** are ranging from 0 to 2047.

See Also

NET_DVR_Logout

3.5.1 fLoginResultCallBack

Login Status Callback Function

| Member | Data Type | Description |
|--------------|--------------------------------------|---|
| lUserID | LONG | User ID, which is returned by <u>NET_DVR_Login_V40</u> . |
| dwResult | DWORD | Login status: 0-asynchronously logging in failed, 1-asynchronously logged in. |
| lpDeviceInfo | <u>NET_DVR_DEVICEINFO_V40</u> | Device information, such as serial No., channel, capability, and so on. |
| pUser | void* | User data. |

3.6 NET_DVR_Logout

Log out from devices.

API Definitions

```
BOOL NET_DVR_Logout (
    LONG    lUserID
);
```

Parameters

lUserID

[IN] User ID, which is returned by **NET_DVR_Login_V40**.

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call **NET_DVR_GetLastError** to get the error code.

The available error codes may be returned by this API are 0, 3, 7, 8, 9, 10, 14, 17, 41, 44, 47, 72, and 73. See details in **Device Network SDK Errors**.

3.7 NET_DVR_SetSDKInitCfg

Set initialization parameters.

API Parameters

```
BOOL NET_DVR_SetSDKInitCfg (
    NET_SDK_INIT_CFG_TYPE    enumType,
```

```
void* const lpInBuff
);
```

Parameters

enumType

[IN] Initialization parameter type. Different type values correspond to different parameters, see details in the table below.

Table 3-1 NET_SDK_INIT_CFG_TYPE

| enumType | Value | Description | lpInBuff |
|------------------------------|-------|--|--|
| NET_SDK_INIT_CFG_ABILITY | 1 | Capability supported by SDK. | <u>NET_DVR_INIT_CFG_ABILITY</u> |
| NET_SDK_INIT_CFG_SDK_PATH | 2 | Set loading path for component libraries (supported by both Linux and Windows system). | <u>NET_DVR_LOCAL_SDK_PATH</u> |
| NET_SDK_INIT_CFG_LIBEAY_PATH | 3 | Set path (including library name) for libeay32.dll (Windows), libcrypto.so (Linux), and libcrypto.dylib (Mac) of OpenSSL in version 1.1.1 and 1.0.2. | Path in string format, e.g., <u>C:\libeay32.dll</u> . |
| NET_SDK_INIT_CFG_SSLEAY_PATH | 4 | Set path (including library name) for ssleay32.dll (Windows), libssl.so (Linux), libssl.dylib (Mac) of OpenSSL in version 1.1.1 and 1.0.2. | Path in string format, e.g., <u>C:\ssleay32.dll</u> . |

lpInBuff

[IN] Input parameter. Different parameter types correspond to different structures, see details in the table above.

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call **NET_DVR_GetLastError** to get the error code.

Remarks

This API should be called before calling **NET_DVR_Init** to initialize and check the dependent libraries or capabilities. This API only takes effect for POSIX. For Windows, it takes no effect but success will be returned.

3.8 NET_DVR_GetSTDAbility

Get the device capabilities.

API Definition

```
BOOL NET_DVR_GetSTDAbility(  
    LONG                lUserID,  
    DWORD               dwAbilityType,  
    NET_DVR_STD_ABILITY lpAbilityParam  
);
```

Parameters

lUserID

[IN] Value returned by **NET_DVR_Login_V40**.

dwAbilityType

[IN] Capability types, which are different according to different functions.

lpAbilityParam

[IN/OUT] Capability details, including condition parameter, input parameter, output parameter, and so on (see details in the structure **NET_DVR_STD_ABILITY**), which are different according to different capability types.

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call **NET_DVR_GetLastError** to get the error code.

3.9 NET_DVR_GetSTDConfig

Get the device configuration information.

API Definition

```
BOOL NET_DVR_GetSTDConfig(  
    LONG                lUserID,  
    DWORD               dwCommand,
```

```
NET_DVR_STD_CONFIG    lpConfigParam
);
```

Parameters

lUserID

[IN] Value returned by [NET_DVR_Login_V40](#).

dwCommand

[IN] Device configuration commands, which are different according to different configuration functions.

lpConfigParam

[IN][OUT] Set input and output parameters, which are different according to different configuration functions. For different configuration functions, the **lpCondBuffer** and **lpOutBuffer** in the **lpConfigParam** are also different. See the structure [NET_DVR_STD_CONFIG](#) for details.



Note

When getting configuration parameters, the **lpInBuffer** in the **lpConfigParam** is invalid, you can set it to NULL.

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call [NET_DVR_GetLastError](#) to get the error code.

See Also

[NET_DVR_SetSTDConfig](#)

3.10 NET_DVR_MIME_UNIT

Input Content Details Structure of Message Transmission API (NET_DVR_STDXMLConfig)

| Member | Data Type | Description |
|---------------|---------------|---|
| szContentType | Array of char | Content type (corresponds to Content-Type field in the message), e.g., text/json. text/xml, and so on. The content format must be supported by HTTP. |
| szName | Array of char | Content name (corresponds to name field in the message), e.g., name="upload". |

| Member | Data Type | Description |
|--------------|---------------|---|
| szFilename | Array of char | Content file name (corresponds to filename field in the message), e.g., filename="C:\Users\test\Desktop\11.txt". |
| dwContentLen | DWORD | Content size |
| pContent | char* | Data point |
| bySelfRead | BYTE | 0-External file, 1-Internal data, whose address is specified by szFilename . |
| byRes | Array of BYTE | Reserved. Set to 0. Maximum: 15 bytes. |

See Also

NET_DVR_XML_CONFIG_INPUT

3.11 NET_DVR_SetSTDConfig

Set the device parameters.

API Definition

```

BOOL NET_DVR_SetSTDConfig(
    LONG                lUserID,
    DWORD               dwCommand,
    NET_DVR_STD_CONFIG  lpConfigParam
);

```

Parameters

lUserID

[IN] Value returned by **NET_DVR_Login_V40**.

dwCommand

[IN] Device configuration commands, which are different according to different configuration functions.

lpConfigParam

[IN][OUT] Set input and output parameters, which are different according to different configuration functions. For different configuration functions, the **lpCondBuffer** and **lpInBuffer** in the **lpConfigParam** are also different. See the structure **NET_DVR_STD_CONFIG** for details.



Note

When getting configuration parameters, the **lpOutBuffer** in the **lpConfigParam** is invalid, you can set it to "NULL".

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call **NET_DVR_GetLastError** to get the error code.

See Also

NET_DVR_GetSTDConfig

3.12 NET_DVR_STDXMLConfig

Transmit request URL with XML or JSON format to implement some typical functions.

API Definition

```
BOOL NET_DVR_STDXMLConfig(  
    LONG                                lUserID,  
    const NET_DVR_XML_CONFIG_INPUT      *lpInputParam,  
    NET_DVR_XML_CONFIG_OUTPUT           *lpOutputParam  
);
```

Parameters

lUserID

[IN] Value returned by **NET_DVR_Login_V40**.

lpInputParam

[IN] Input parameters, refer to the structure **NET_DVR_XML_CONFIG_INPUT** for details.

lpOutputParam

[IN][OUT] Output parameters, refer to the structure **NET_DVR_XML_CONFIG_OUTPUT** for details.

Return Values

Return *TRUE* for success, and return *FALSE* for failure.

If *FALSE* is returned, you can call **NET_DVR_GetLastError** to get the error code.

Remarks

The input parameter **lpInputParam** and output parameter **lpOutputParam** are different when transmitting text protocol for implementing different functions, and each parameter corresponds to a component of text protocol, see the relations below:

| Parameter of NET_DVR_STDXMLConfig | | Component of Text Protocol |
|-----------------------------------|---|--|
| IpInputParam | IpRequestUrl (see in structure <u>NET_DVR_XML_CONFIG_INPUT</u>) | Method+URL E.g., GET /ISAPI/System/capabilities |
| | IpInBuffer (see in structure <u>NET_DVR_XML_CONFIG_INPUT</u>) | Request Message |
| IpOutputParam | IpOutBuffer (see in structure <u>NET_DVR_XML_CONFIG_OUTPUT</u>) | Response Message |
| | IpStatusBuffer (see in structure <u>NET_DVR_XML_CONFIG_OUTPUT</u>) | Response Message |

3.13 NET_DVR_CloseAlarmChan_V30

Close alarm uploading channel.

API Definition

```

BOOL NET_DVR_CloseAlarmChan_V30(
    LONG    lAlarmHandle
);

```

Parameters

lAlarmHandle

Value returned by NET_DVR_SetupAlarmChan_V50 .

Return Values

Return *TURE* for success, and return *FALSE* for failure.

If *FALSE* is returned, you can call NET_DVR_GetLastError to get the error code.

The available error codes of this API are 0, 3, 6, 12, 17, 41, and 47. See details in the **Device Network SDK Errors** .

3.14 NET_DVR_GetDVRConfig

Get the device configuration information.

API Definition

```
BOOL NET_DVR_GetDVRConfig(  
    LONG        lUserID,  
    DWORD       dwCommand,  
    LONG        lRuleID,  
    LONG        lChannel,  
    LPVOID      lpOutBuffer,  
    DWORD       dwOutBufferSize,  
    LPDWORD     lpBytesReturned  
);
```

Parameters

lUserID

[IN] Value returned by [NET_DVR_Login_V40](#) .

dwCommand

[IN] Device getting commands, which are different according to different getting functions.

lRuleID

[IN] Rule ID.

lChannel

[IN] Channel No. (NIC No.), which varies with different commands. 0xffffffff-invalid or all channels, 1-main NIC, 2-extended NIC.

lpOutBuffer

[OUT] Pointer of buffer to receive data. For different getting functions, the structures of this parameter are different.

dwOutBufferSize

[IN] Size of buffer to receive data (unit: byte). It cannot be 0.

lpBytesReturned

[OUT] Pointer of actually received data size. It cannot be NULL.

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call [NET_DVR_GetLastError](#) to get the error code.

The following error codes may be returned by this API: 0, 3, 6, 7, 8, 9, 10, 12, 17, 41, 43, 44, 47, 72, 73, and 76. See the corresponding error types and descriptions in the [Device Network SDK Errors](#) .

See Also

[NET_DVR_SetDVRConfig](#)

3.15 NET_DVR_SetDVRConfig

Set the device parameters.

API Definition

```
BOOL NET_DVR_SetDVRConfig(  
    LONG        lUserID,  
    DWORD       dwCommand,  
    LONG        lChannel,  
    LPVOID      lpInBuffer,  
    DWORD       dwInBufferSize  
);
```

Parameters

lUserID

[IN] Value returned by **NET_DVR_Login_V40** .

dwCommand

[IN] Device configuration commands, which are different according to different configuration functions.

lChannel

[IN] Channel No. (NIC No.), which varies with different commands. 0xFFFFFFFF-invalid, 1-main NIC, 2-extended NIC.

lpInBuffer

[IN] Pointer of input data buffer. For different configuration functions, the structures of this parameter are different.

dwInBufferSize

[IN] Size of input data buffer (unit: byte).

Return Values

Returns *TRUE* for success, and returns *FALSE* for failure.

If *FALSE* is returned, you can call **NET_DVR_GetLastError** to get the error code.

The following error codes may be returned by this API: 0, 3, 6, 7, 8, 9, 10, 12, 17, 41, 43, 44, 47, 72, 73, and 76. See the corresponding error types and descriptions in the **Device Network SDK Errors** .

See Also

NET_DVR_GetDVRConfig

3.16 NET_DVR_SetDVRMessageCallBack_V50

Set callback functions for getting the video data.

API Definition

```
BOOL NET_DVR_SetDVRMessageCallBack_V50(  
    int            iIndex,  
    MSGCallBack    fMessageCallBack,  
    void           *pUser  
);
```

Parameters

iIndex

[IN] Callback function index No., which ranges from 0 to 15.

fMessageCallBack

[IN] Callback function, see details in [***MSGCallBack***](#).

pUser

[IN] User data.

Return Values

Return *TRUE* for success, and return *FALSE* for failure.

If *FALSE* returned, call [***NET_DVR_GetLastError***](#) to get the error code.

Remarks

- This API supports setting multiple callback functions for different channels (up to 16 channels are supported) at same time, and the configured callback functions are distinguished by the index No.
- All alarm/event information will be returned in each configured callback function, and you can distinguish the devices via the **pAlarmInfo** in the callback function ([***MSGCallBack***](#)).

Example

Sample Code of Setting Multiple Callback Functions to Receive Different Alarms/Events in Arming Mode

```
#include <stdio.h>  
#include <iostream>  
#include "Windows.h"  
#include "HCNetSDK.h"  
using namespace std;  
  
int iNum=0;  
void CALLBACK MessageCallbackNo1(LONG lCommand, NET_DVR_ALARMER *pAlarmer, char  
*pAlarmInfo, DWORD dwBufLen, void* pUser)  
{
```

```
int i=0;
char filename[100];
FILE *fSnapPic=NULL;
FILE *fSnapPicPlate=NULL;

//This sample code is for reference only. Actually, it is not recommended
to process the data and save file in the callback function directly.
//You'd better process the data in the message response function via message
mode (PostMessage).

switch(lCommand)
{
    case COMM_ALARM:
    {
        NET_DVR_ALARMINFO struAlarmInfo;
        memcpy(&struAlarmInfo, pAlarmInfo, sizeof(NET_DVR_ALARMINFO));
        switch (struAlarmInfo.dwAlarmType)
        {
            case 3: //Motion detection alarm
                for (i=0; i<16; i++)    //define MAX_CHANNUM    16    //The
maximum number of channels
                {
                    if (struAlarmInfo.dwChannel[i] == 1)
                    {
                        printf("Channel Number with Motion Detection Alarm
%d\n", i+1);
                    }
                }
                break;
            default:
                break;
        }
        break;
    }
    case COMM_UPLOAD_PLATE_RESULT:
    {
        NET_DVR_PLATE_RESULT struPlateResult={0};
        memcpy(&struPlateResult, pAlarmInfo, sizeof(struPlateResult));
        printf("License Plate Number: %s\n",
struPlateResult.struPlateInfo.sLicense); //License plate number

        switch(struPlateResult.struPlateInfo.byColor) //License plate color
        {
            case VCA_BLUE_PLATE:
                printf("Vehicle Color: Blue\n");
                break;
            case VCA_YELLOW_PLATE:
                printf("Vehicle Color: Yellow\n");
                break;
            case VCA_WHITE_PLATE:
                printf("Vehicle Color: White\n");
                break;
        }
    }
}
```

```
        case VCA_BLACK_PLATE:
            printf("Vehicle Color: Black\n");
            break;
        default:
            break;
    }
    //Scene picture
    if (struPlateResult.dwPicLen != 0 && struPlateResult.byResultType
== 1 )
    {
        sprintf(filename,"testpic_%d.jpg",iNum);
        fSnapPic=fopen(filename,"wb");
        fwrite(struPlateResult.pBuffer1,struPlateResult.dwPicLen,
1,fSnapPic);
        iNum++;
        fclose(fSnapPic);
    }
    //License plate picture
    if (struPlateResult.dwPicPlateLen != 0 &&
struPlateResult.byResultType == 1)
    {
        sprintf(filename,"testPicPlate_%d.jpg",iNum);
        fSnapPicPlate=fopen(filename,"wb");
        fwrite(struPlateResult.pBuffer1,struPlateResult.dwPicLen,
1,fSnapPicPlate);
        iNum++;
        fclose(fSnapPicPlate);
    }
    //Processing other data...
    break;
}
case COMM_ITS_PLATE_RESULT:
{
    NET_ITS_PLATE_RESULT struITSPlateResult={0};
    memcpy(&struITSPlateResult, pAlarmInfo, sizeof(struITSPlateResult));

    for (i=0;i<struITSPlateResult.dwPicNum;i++)
    {
        printf("License Plate Number: %s\n",
struITSPlateResult.struPlateInfo.sLicense);//License plate number
        switch(struITSPlateResult.struPlateInfo.byColor)//License plate
color
        {
            case VCA_BLUE_PLATE:
                printf("Vehicle Color: Blue\n");
                break;
            case VCA_YELLOW_PLATE:
                printf("Vehicle Color: Yellow\n");
                break;
            case VCA_WHITE_PLATE:
                printf("Vehicle Color: White\n");
                break;
        }
    }
}
```



```
        case VCA_BLACK_PLATE:
            printf("Vehicle Color: Black\n");
            break;
        default:
            break;
    }
    //Save scene picture
    if ((struITSPlateResult.struPicInfo[i].dwDataLen !=
0)&&(struITSPlateResult.struPicInfo[i].byType== 1)||
(struITSPlateResult.struPicInfo[i].byType == 2))
    {
        sprintf(filename,"testITSpic%d_%d.jpg",iNum,i);
        fSnapPic=fopen(filename,"wb");
        fwrite(struITSPlateResult.struPicInfo[i].pBuffer,
struITSPlateResult.struPicInfo[i].dwDataLen,1,fSnapPic);
        iNum++;
        fclose(fSnapPic);
    }
    //License plate thumbnails
    if ((struITSPlateResult.struPicInfo[i].dwDataLen !=
0)&&(struITSPlateResult.struPicInfo[i].byType == 0))
    {
        sprintf(filename,"testPicPlate%d_%d.jpg",iNum,i);
        fSnapPicPlate=fopen(filename,"wb");
        fwrite(struITSPlateResult.struPicInfo[i].pBuffer,
struITSPlateResult.struPicInfo[i].dwDataLen, 1, \ fSnapPicPlate);
        iNum++;
        fclose(fSnapPicPlate);
    }
    //Processing other data...
}
break;
}
default:
    break;
}
}

void CALLBACK MessageCallbackNo2(LONG lCommand, NET_DVR_ALARMER *pAlarmer, char
*pAlarmInfo, DWORD dwBufLen, void* pUser)
{
    int i=0;
    char filename[100];
    FILE *fSnapPic=NULL;
    FILE *fSnapPicPlate=NULL;

    //This sample code is for reference only. Actually, it is not recommended
to process the data and save file in the callback function directly.
    //You'd better process the data in the message response function via message
mode (PostMessage).

    switch(lCommand)
```

```
{
    case COMM_ALARM:
    {
        NET_DVR_ALARMINFO struAlarmInfo;
        memcpy(&struAlarmInfo, pAlarmInfo, sizeof(NET_DVR_ALARMINFO));
        switch (struAlarmInfo.dwAlarmType)
        {
            case 3: //Motion detection alarm
                for (i=0; i<16; i++)    //define MAX_CHANNUM 16 //The
maximum number of channel
                {
                    if (struAlarmInfo.dwChannel[i] == 1)
                    {
                        printf("Channel No. with Motion Detection Alarm %d
\n", i+1);
                    }
                }
                break;
            default:
                break;
        }
        break;
    }
    case COMM_UPLOAD_PLATE_RESULT:
    {
        NET_DVR_PLATE_RESULT struPlateResult={0};
        memcpy(&struPlateResult, pAlarmInfo, sizeof(struPlateResult));
        printf("License Plate Number: %s\n",
struPlateResult.struPlateInfo.sLicense); //License plate number

        switch(struPlateResult.struPlateInfo.byColor) //License plate color
        {
            case VCA_BLUE_PLATE:
                printf("Vehicle Color: Blue\n");
                break;
            case VCA_YELLOW_PLATE:
                printf("Vehicle Color: Yellow\n");
                break;
            case VCA_WHITE_PLATE:
                printf("Vehicle color: White\n");
                break;
            case VCA_BLACK_PLATE:
                printf("Vehicle Color: Black\n");
                break;
            default:
                break;
        }
        //Scene picture
        if (struPlateResult.dwPicLen != 0 && struPlateResult.byResultType
== 1 )
        {
            sprintf(filename, "testpic_%d.jpg", iNum);
```

```
fSnapPic=fopen(filename,"wb");
fwrite(struPlateResult.pBuffer1,struPlateResult.dwPicLen,
1,fSnapPic);
    iNum++;
    fclose(fSnapPic);
}
//License plate picture
if (struPlateResult.dwPicPlateLen != 0 &&
struPlateResult.byResultType == 1)
{
    sprintf(filename,"testPicPlate_%d.jpg",iNum);
    fSnapPicPlate=fopen(filename,"wb");
    fwrite(struPlateResult.pBuffer1,struPlateResult.dwPicLen,
1,fSnapPicPlate);
    iNum++;
    fclose(fSnapPicPlate);
}
//Processing other data...
break;
}
case COMM_ITS_PLATE_RESULT:
{
    NET_ITS_PLATE_RESULT struITSPlateResult={0};
    memcpy(&struITSPlateResult, pAlarmInfo, sizeof(struITSPlateResult));

    for (i=0;i<struITSPlateResult.dwPicNum;i++)
    {
        printf("License Plate Number: %s\n",
struITSPlateResult.struPlateInfo.sLicense);//License plate number
        switch(struITSPlateResult.struPlateInfo.byColor)//License plate
color
        {
            case VCA_BLUE_PLATE:
                printf("Vehicle Color: Blue\n");
                break;
            case VCA_YELLOW_PLATE:
                printf("Vehicle Color: Yellow\n");
                break;
            case VCA_WHITE_PLATE:
                printf("Vehicle Color: White\n");
                break;
            case VCA_BLACK_PLATE:
                printf("Vehicle Color: Black\n");
                break;
            default:
                break;
        }
        //Save scene picture
        if ((struITSPlateResult.struPicInfo[i].dwDataLen !=
0)&&(struITSPlateResult.struPicInfo[i].byType== 1)||
(struITSPlateResult.struPicInfo[i].byType == 2))
        {
```

```
        sprintf(filename, "testITSpic%d_%d.jpg", iNum, i);
        fSnapPic=fopen(filename, "wb");
        fwrite(struITSPlateResult.struPicInfo[i].pBuffer,
struITSPlateResult.struPicInfo[i].dwDataLen, 1, fSnapPic);
        iNum++;
        fclose(fSnapPic);
    }
    //License plate thumbnails
    if ((struITSPlateResult.struPicInfo[i].dwDataLen !=
0)&&(struITSPlateResult.struPicInfo[i].byType == 0))
    {
        sprintf(filename, "testPicPlate%d_%d.jpg", iNum, i);
        fSnapPicPlate=fopen(filename, "wb");
        fwrite(struITSPlateResult.struPicInfo[i].pBuffer,
struITSPlateResult.struPicInfo[i].dwDataLen, 1, \ fSnapPicPlate);
        iNum++;
        fclose(fSnapPicPlate);
    }
    //Processing other data...
}
break;
}
default:
    break;
}
}

void main() {

    //-----
    //Initialize
    NET_DVR_Init();
    //Set the connection time and reconnection time
    NET_DVR_SetConnectTime(2000, 1);
    NET_DVR_SetReconnect(10000, true);

    //-----
    //Log in to device
    LONG lUserID;
    NET_DVR_DEVICEINFO_V30 struDeviceInfo;
    lUserID = NET_DVR_Login_V30("172.0.0.100", 8000, "admin", "12345",
&struDeviceInfo);
    if (lUserID < 0)
    {
        printf("Login error, %d\n", NET_DVR_GetLastError());
        NET_DVR_Cleanup();
        return;
    }

    //Set alarm callback function
    NET_DVR_SetDVRMessageCallBack_V50(0, MessageCallbackNo1, NULL);
    NET_DVR_SetDVRMessageCallBack_V50(1, MessageCallbackNo2, NULL);
```

```
//Enable arming
NET_DVR_SETUPALARM_PARAM struSetupParam={0};
struSetupParam.dwSize=sizeof(NET_DVR_SETUPALARM_PARAM);

//Alarm information type to upload: 0-History Alarm (NET_DVR_PLATE_RESULT), 1-
Real-Time Alarm (NET_ITS_PLATE_RESULT)
struSetupParam.byAlarmInfoType=1;
//Arming Level: Level-2 arming (for traffic device)
struSetupParam.byLevel=1;

LONG lHandle = NET_DVR_SetupAlarmChan_V41(lUserID,&struSetupParam);
if (lHandle < 0)
{
    printf("NET_DVR_SetupAlarmChan_V41 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(lUserID);
    NET_DVR_Cleanup();
    return;
}

Sleep(20000);
//Disarm uploading channel
if (!NET_DVR_CloseAlarmChan_V30(lHandle))
{
    printf("NET_DVR_CloseAlarmChan_V30 error, %d\n", NET_DVR_GetLastError());
    NET_DVR_Logout(lUserID);
    NET_DVR_Cleanup();
    return;
}

//User logout
NET_DVR_Logout(lUserID);
//Release SDK resource
NET_DVR_Cleanup();
return;
}
```

See Also

NET_DVR_SetupAlarmChan_V50

3.16.1 MSGCallback

Alarm/event information callback function.

Callback Function Definition

```
typedef void(CALLBACK *MSGCallback) (
    LONG                lCommand,
    NET_DVR_ALARMER     *pAlarmer,
    char                *pAlarmInfo,
```

```
DWORD          dwBufLen,  
void           *pUser  
) ;
```

Parameters

ICommand

[OUT] Uploaded message type. You can distinguish the alarm/event information via the type.

pAlarmer

[OUT] Alarm device information, including serial No., IP address, login handle, and so on, see details in **NET_DVR_ALARMER**.

pAlarmInfo

[OUT] Alarm/event information, the details are returned in different structures according to **ICommand**.

dwBufLen

[OUT] Size of alarm/event information buffer.

pUser

[OUT] User data.

3.17 NET_DVR_SetupAlarmChan_V50

Set up persistent connection to receive alarm/event information (supports alarm/event subscription).

API Definition

```
LONG NET_DVR_SetupAlarmChan_V50(  
    LONG          lUserID,  
    NET_DVR_SETUPALARM_PARAM_V50 lpSetupParam,  
    char          *pData,  
    DWORD         dwDataLen,  
) ;
```

Parameters

lUserID

[IN] Value returned by **NET_DVR_Login_V40**.

lpSetupParam

[IN] Arming parameters, refer to the structure **NET_DVR_SETUPALARM_PARAM_V50** for details.

pData

[IN] Alarm/event subscription conditions.

dwDataLen

[IN] Length of alarm/event subscription conditions.

Return Values

Return -1 for failure, and return other values as the handles of **NET_DVR_CloseAlarmChan_V30** .
If -1 is returned, you can call **NET_DVR_GetLastError** to get the error code.

Remarks

This API supports alarm/event subscription, you can specify the types of alarm or event to be uploaded by device by setting **pData** and **dwDataLen**.

3.18 NET_DVR_StartListen_V30

Register callback function for receiving alarm/event information and start listening (supports multiple threads).

API Definition

```
LONG NET_DVR_StartListen_V30(  
    char            *sLocalIP,  
    WORD            wLocalPort,  
    MSGCallBack     DataCallback,  
    void            *pUserData  
);
```

Parameters

sLocalIP

[IN] IP address of local PC. It can be set to null.

wLocalPort

[IN] Listening port No. of local PC. It is configured by user, and it should be the same with that of device.

DataCallback

[IN] Alarm/event information callback function, see details in **MSGCallBack** .

pUserData

[IN] User data.

Return Values

Return -1 for failure, and return other values for the handle parameters of **NET_DVR_StopListen_V30** .
If -1 is returned, you can call **NET_DVR_GetLastError** to get the error code.

The available error codes of this API are 0, 3, 6, 12, 17, 41, 44, 47, 72, and 75. See details in the [**Device Network SDK Errors**](#) .

Remarks

- To receive the alarm/event information sent by device, you should set the management host server address or listening host server address of device to the IP address of PC (which is same with the **sLocalIP**), or set the management host server port or listening host server port to the listening port No. of PC (which is same with the **wLocalPort**).
- The callback function in this API is prior to other callback functions, that is, if the callback function is configured in this API, other callback functions will not receive the alarm information. All the device alarm information is returned in same callback function, and you can distinguish the devices via the alarm device information (**pAlarmInfo**).

3.19 NET_DVR_StopListen_V30

Stop listening (supports multiple threads).

API Definition

```
BOOL NET_DVR_StopListen_V30(  
    LONG    lListenHandle  
);
```

Parameters

lListenHandle

Listening handle, which is returned by [**NET_DVR_StartListen_V30**](#) .

Return Values

Return *TRUE* for success, and return *FALSE* for failure.

If *FALSE* is returned, you can call [**NET_DVR_GetLastError**](#) to get the error code.

The available error codes of this API are 0, 3, 12, and 17. See details in the [**Device Network SDK Errors**](#) .

Chapter 4 Request URL

The request URLs for implementing the functions in this manual are listed here for reference. You can search for the URLs and view their definitions.

4.1 /ISAPI/Intelligent/channels/<ID>/capabilities

Get the intelligent capability of device.

Request URI Definition

Table 4-1 GET /ISAPI/Intelligent/channels/<ID>/capabilities

| | |
|-------------|---|
| Method | GET |
| Description | Get the intelligent capability of device. |
| Query | None |
| Request | None. |
| Response | Succeeded: <u><i>XML_IntelliCap</i></u> Failed: <u><i>XML_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

4.2 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/capabilities?format=json

Get the configuration capability of queue management by channel.

Request URI Definition

Table 4-2 GET /ISAPI/Intelligent/channels/<ID>/personQueueDetection/capabilities?format=json

| | |
|-------------|---|
| Method | GET |
| Description | Get the configuration capability of queue management by channel. |
| Query | format: determine the format of request or response message. |

| | |
|-----------------|--|
| Request | None |
| Response | Succeeded: <u><i>JSON_PersonQueueDetectionInfoCap</i></u> Failed: <u><i>JSON_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the channel ID.

4.3 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingFile?format=json

Search for and get the people queuing-up detection report file.

Request URI Definition

Table 4-3 POST /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingFile?format=json

| | |
|--------------------|--|
| Method | POST |
| Description | Search for and get the people queuing-up detection report file. |
| Query | format: determine the format of request or response message. |
| Request | <u><i>JSON_PersonQueueCountingDescription</i></u> |
| Response | Succeeded: <u><i>JSON_PersonQueueCountingFileResult</i></u> Failed: <u><i>JSON_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

4.4 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingResult?format=json

Search for and get the people queuing-up detection report details.

Request URI Definition

Table 4-4 POST /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueCountingResult?format=json

| | |
|-------------|--|
| Method | POST |
| Description | Search for and get the people queuing-up detection report details. |
| Query | format : determine the format of request or response message. |
| Request | <u><i>JSON_PersonQueueCountingDescription</i></u> |
| Response | Succeeded: <u><i>JSON_PersonQueueCountingResult</i></u> Failed: <u><i>JSON_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

4.5 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueTimeFile?format=json

Search for and get the waiting time detection report file.

Request URI Definition

Table 4-5 POST /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueTimeFile?format=json

| | |
|-------------|--|
| Method | POST |
| Description | Search for and get the waiting time detection report file. |
| Query | format : determine the format of request or response message. |
| Request | <u><i>JSON_PersonQueueTimeDescription</i></u> |
| Response | Succeeded: <u><i>JSON_PersonQueueTimeFileResult</i></u> Failed: <u><i>JSON_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

4.6 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueTimeResult?format=json

Search for and get the waiting time detection report details.

Request URI Definition

Table 4-6 POST /ISAPI/Intelligent/channels/<ID>/personQueueDetection/queueTimeResult?format=json

| | |
|-------------|--|
| Method | POST |
| Description | Search for and get the waiting time detection report details. |
| Query | format : determine the format of request or response message. |
| Request | None |
| Response | Succeeded: <u><i>JSON_PersonQueueTimeResult</i></u> Failed: <u><i>JSON_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

4.7 /ISAPI/Intelligent/channels/<ID>/personQueueDetection/rawDataFile?format=json

Search for and get the queue management raw data in file format.

Request URI Definition

Table 4-7 POST /ISAPI/Intelligent/channels/<ID>/personQueueDetection/rawDataFile?format=json

| | |
|-------------|---|
| Method | POST |
| Description | Search for and get the queue management raw data in file format. |
| Query | format : determine the format of request or response message. |
| Request | <u><i>JSON_PersonQueueRawDataFileDescription</i></u> |
| Response | Succeeded: <u><i>JSON_PersonQueueRawDataFileResult</i></u> Failed: <u><i>JSON_ResponseStatus</i></u> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

4.8 /ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json

Get or set parameters of queue management alarm by channel.

Request URI Definition

Table 4-8 GET /ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json

| | |
|-------------|---|
| Method | GET |
| Description | Get parameters of queue management alarm by channel. |
| Query | format : determine the format of request or response message. |
| Request | None |
| Response | Succeeded: <i><u>JSON_PersonQueueDetectionInfo</u></i> Failed: <i><u>JSON_ResponseStatus</u></i> |

Table 4-9 PUT /ISAPI/Intelligent/channels/<ID>/personQueueDetection?format=json

| | |
|-------------|--|
| Method | PUT |
| Description | Set parameters of queue management alarm by channel. |
| Query | format : determine the format of request or response message. |
| Request | <i><u>JSON_PersonQueueDetectionInfo</u></i> |
| Response | <i><u>JSON_ResponseStatus</u></i> |

Remarks

The <ID> in the URI refers to the intelligent channel ID.

Chapter 5 Request and Response Message

The request and response messages in XML or JSON format of each request URL are listed here for reference. You can search for the parameters by the message name.

5.1 JSON_EventNotificationAlert_Alarm/EventInfo

EventNotificationAlert message with alarm or event information in JSON format.

```
{
  "ipAddress": "",
  /*required, device IPv4 address , string, the maximum size is 32 bytes*/
  "ipv6Address": "",
  /*optional, device IPv6 address, string, the maximum size is 128 bytes*/
  "portNo": ,
  /*optional, device port No., integer32*/
  "protocol": "",
  /*optional, protocol type, "HTTP, HTTPS", string, the maximum size is 32 bytes*/
  "macAddress": "",
  /*optional, MAC address, string, the maximum size is 32 bytes, e.g.,
  01:17:24:45:D9:F4*/
  "channelID": "",
  /*optional, device channel No., integer32*/
  "dateTime": "",
  /*optional, string, alarm/event triggered or occurred time based on ISO8601,
  the maximum size is 32 bytes, e.g., 2009-11-14T15:27Z*/
  "activePostCount": "",
  /*required, alarm/event frequency, integer32*/
  "eventType": "",
  /*required, alarm/event type, "captureResult, faceCapture,...", string, the
  maximum size is 128 bytes*/
  "eventState": "",
  /*required, string, the maximum size is 32 bytes, durative alarm/event status:
  "active"-valid, "inactive"-invalid*/
  "eventDescription": "",
  /*required, event description, string, the maximum size is 128 bytes*/
  "deviceId": "",
  /*string type, device ID*/
  "uuid": "",
  /*string type, event UUID, which is used to uniquely identify an event, the
  standard UUID format is xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx*/
  ...
  /*optional, for different alarm/event types, the nodes are different, see the
  message examples in different applications*/
}
```

5.2 JSON_EventNotificationAlert_PeopleQueuingUpAlarmMsg

JSON message about regional people queuing-up alarm details

Regional People Queuing-Up Alarm with Binary Picture Data

```
Content-Type: multipart/form-data; boundary=MIME_boundary
--MIME_boundary
Content-Type: application/json
Content-Length: 480
{
  "ipAddress": "",
  "ipv6Address": "",
  "portNo": ,
  "protocol": "",
  "macAddress": "",
  "channelID": "",
  "relatedChannelList": [1,2,3],
  /*optional, array of integers, list of alarm related channels, which are of the
  same camera with channelID; this parameter is used for live view or playback on
  the platform*/
  "dateTime": "",
  "activePostCount": ,
  "eventType": "",
  /*required, string, occurred event type, here it should be set to
  "personQueueCounting", and the maximum value size is 128 bytes*/
  "eventState": "",
  "eventDescription": "",
  "deviceID": "",
  "RegionCapture"{
  /*captured region information*/
    "humanCounting":{
  /*people amount*/
      "count":
  /*required, integer, actual people amount, it is ranging from 0 to 60*/
    }
    "rule":{
  /*rule properties*/
      "ruleID": ,
  /*required, integer, rule ID, which corresponds to the target person region ID*/
      "alarmCount": ,
  /*required, integer, configured threshold of queuing-up people amount, it is
  ranging from 0 to 60*/
      "regionColor": "",
  /*required, string, rule region color, "blue", "red", and "yellow" are
  available*/
      "Region": [{
  /*required, rule region coordinates, it is polygonal (3 to 10 edges)*/
        "x": ,
  /*required, float, x-coordinate, it is ranging from 0.000 to 1*/
```

```
"y":
/*required, float, y-coordinate, it is ranging from 0.000 to 1*/
  ]]
  "countTriggerType": "",
/*optional, string type, triggering type of people queuing-up alarm: "greater",
"less", "equal", "unequal"-the alarm is triggered when the number of queuing-up
persons is larger than, smaller than, equal to, or not equal to the value of
alarmCount; "range"-the alarm is triggered when the number of queuing-up person
is in the range between maxAlarmCount and alarmCount*/
  "durationTime":
/*optional, integer type, queuing-up status (number of queuing-up persons)
keeping duration, which is between 0 and 3600, the default duration is 300,
unit: second; the people queuing-up alarm is triggered if the number of queuing-
up persons keeps for the configured duration*/
  }
}
"contentID": ""
/*background picture*/

}
--MIME_boundary
Content-Type: image/jpeg
Content-Length: 42569
Content-ID: human_image

fefefwageegfqaeg...
--MIME_boundary--
```

Regional People Queuing-Up Alarm with Picture URL

```
{
  "ipAddress": "",
  "ipv6Address": "",
  "portNo": ,
  "protocol": "",
  "macAddress": "",
  "channelID": "",
  "relatedChannelList": [1,2,3],
/*optional, array of integers, list of alarm related channels, which are of the
same camera with channelID; this parameter is used for live view or playback on
the platform*/
  "dateTime": "",
  "activePostCount": ,
  "eventType": "",
/*required, string, occurred event type, here it should be set to
"personQueueCounting", and the maximum value size is 128 bytes*/
  "eventState": "",
  "eventDescription": "",
  "deviceID": "",
  "RegionCapture"{
/*captured region information*/
    "humanCounting":{
```



```
/*people amount*/
  "count":
/*required, integer, actual people amount, it is ranging from 0 to 60*/
  }
  "rule":{
/*rule properties*/
  "ruleID": ,
/*required, integer, rule ID, which corresponds to the target person region ID*/
  "alarmCount": ,
/*required, integer, configured threshold of queuing-up people amount, it is
ranging from 0 to 60*/
  "regionColor": "",
/*required, string, rule region color, "blue", "red", and "yellow" are
available*/
  "Region": [{
/*required, rule region coordinates, it is polygonal (3 to 10 edges)*/
  "x": ,
/*required, float, x-coordinate, it is ranging from 0.000 to 1*/
  "y":
/*required, float, y-coordinate, it is ranging from 0.000 to 1*/
  }]
  "countTriggerType": "",
/*optional, string type, triggering type of people queuing-up alarm: "greater",
"less", "equal", "unequal"-the alarm is triggered when the number of queuing-up
persons is larger than, smaller than, equal to, or not equal to the value of
alarmCount; "range"-the alarm is triggered when the number of queuing-up person
is in the range between maxAlarmCount and alarmCount*/
  "durationTime":
/*optional, integer type, queuing-up status (number of queuing-up persons)
keeping duration, which is between 0 and 3600, the default duration is 300,
unit: second; the people queuing-up alarm is triggered if the number of queuing-
up persons keeps for the configured duration*/
  }
}
  "backgroundImageURL": ""
/*background picture URL*/
}
```

5.3 JSON_EventNotificationAlert_RealTimeQueueManagementDataMsg

JSON message about details of real-time regional people queuing-up alarm

```
{
  "ipAddress": "",
/*required, string, IPv4 address of alarm device, the maximum value size is 32
bytes, e.g., 172.6.64.7*/
  "ipv6Address": "",
/*required, string, IPv6 address of alarm device, the maximum value size is 128
bytes*/
  "portNo": ,
```

```
/*optional, integer32, port No. of alarm device*/
  "protocol": "",
/*optional, string, protocol type: "HTTP"-for device network SDK, "HTTPS",
"EHome"-for ISUP SDK; the maximum value size is 32 bytes*/
  "macAddress": "",
/*optional, string, MAC address, the maximum value size is 32 bytes, e.g.,
01:17:24:45:D9:F4*/
  "channelID": ,
/*optional, integer32, No. of alarm triggered channel*/
  "relatedChannelList": [1,2,3],
/*optional, array of integers, list of alarm related channels, which are of the
same camera with channelID; this parameter is used for live view or playback on
the platform*/
  "dateTime": "",
/*required, string, alarm triggered time (ISO 8601 format), the maximum value
size is 32 bytes, e.g., 2004-05-03T17:30:08+08:00*/
  "activePostCount": ,
/*required, integer32, uploaded times of one alarm*/
  "eventType": "",
/*required, string, occurred event type, here it should be set to
"personQueueRealTime", and the maximum value size is 128 bits*/
  "eventState": "",
  "eventDescription": "",
/*see the description of above nodes in JSON_EventNotificationAlert_Alarm/
EventInfo*/
  "deviceID": "",
/*optional, device ID (PUID), which is returned in the alarm based on ISUP
(Intelligent Security Uplink Protocol)*/
  "PersonQueueRealTimeData": {
/*real-time information of queue management*/
    "targetID": ,
/*required, integer type, target ID*/
    "ruleID": ,
/*required, integer type, rule ID*/
    "personState": "",
/*required, string type, people status: "enter"-region entrance, "leave"-region
exiting*/
    "waitTime": ,
/*dependent, integer32 type, actual waiting time, this node is valid only when
personState is "leave", whose value is between 0 and 3600s*/
    "peopleNum": ,
/*required, integer type, total number of people in this region*/
    "enterTime": ""
/*dependent, target entered time, which is in ISO8601 time format, this node is
valid only when personState is "leave"*/
  }
}
```

5.4 JSON_EventNotificationAlert_WaitingTimeDetectionAlarmMsg

JSON message about details of waiting time detection alarm

Waiting Time Detection Alarm with Binary Picture Data

```
Content-Type: multipart/form-data; boundary=MIME_boundary
--MIME_boundary
Content-Type: application/json
Content-Length: 480
{
  "ipAddress": "",
  /*required, string, IPv4 address of alarm device, the maximum value size is 32
  bytes, e.g., 172.6.64.7*/
  "ipv6Address": "",
  /*required, string, IPv6 address of alarm device, the maximum value size is 128
  bytes*/
  "portNo": ,
  /*optional, integer32, port No. of alarm device*/
  "protocol": "",
  /*optional, string, protocol type: "HTTP"-for device network SDK, "HTTPS",
  "EHome"-for ISUP SDK; the maximum value size is 32 bytes*/
  "macAddress": "",
  /*optional, string, MAC address, the maximum value size is 32 bytes, e.g.,
  01:17:24:45:D9:F4*/
  "channelID": ,
  /*optional, integer32, No. of alarm triggered channel*/
  "relatedChannelList": [1,2,3],
  /*optional, array of integers, list of alarm related channels, which are of the
  same camera with channelID; this parameter is used for live view or playback on
  the platform*/
  "dateTime": "",
  /*required, string, alarm triggered time (ISO 8601 format), the maximum value
  size is 32 bytes, e.g., 2004-05-03T17:30:08+08:00*/
  "activePostCount": ,
  /*required, integer32, uploaded times of one alarm*/
  "eventType": "",
  /*required, string, occurred event type, here it should be set to
  "personQueueTime", and the maximum value size is 128 bytes*/
  "eventState": "",
  "eventDescription": "",
  "deviceID": "",
  "humanCapture": {
  /*captured human information*/
    "human":{
  /*human attributes*/
    "targetID": ,
  /*required, integer, person target ID, whose waiting time is longer than the
  configured alarm threshold. The ID is generated by DSP, and corresponds to the
  person position*/
```

```
        "waitTime": ,
/*required, integer, actual waiting time, it is ranging from 0 to 3600s*/
        "Region": [{
/*required, target person region coordinates, it is quadrilateral*/
            "x": ,
/*required, float, x-coordinate, it is ranging from 0.000 to 1*/
            "y":
/*required, float, y-coordinate, it is ranging from 0.000 to 1*/
        }]
        contentID": ""
/*captured human thumbnail*/
    },
    "rule":{
/*rule properties*/
        "ruleID": ,
/*required, integer, rule ID, which corresponds to the target person region ID*/
        "alarmTime": ,
/*required, integer, configured waiting time alarm threshold, it is ranging
from 0 to 3600s*/
        "regionColor": "",
/*required, string, rule region color, "blue", "red", and "yellow" are
available*/
        "Region": [{
/*required, rule region coordinates, it is polygonal (3 to 10 edges)*/
            "x": ,
/*required, float, x-coordinate, it is ranging from 0.000 to 1*/
            "y":
/*required, float, y-coordinate, it is ranging from 0.000 to 1*/
        }]
        "timeTriggerType": ""
/*optional, string type, triggering type of waiting time detection alarm:
"greater", "less", "equal", "unequal"-the alarm is triggered when waiting time
is larger than, smaller than, equal to, or not equal to the value of alarmTime*/
    }
},
    "contentID": ""
/*background picture*/
}
--MIME_boundary
Content-Type: image/jpeg
Content-Length: 42569
Content-ID: human_image

fefefwageegfqaeg...
--MIME_boundary
Content-Type: image/jpeg
Content-Length: 516876
Content-ID: background_image

fefefwageegfqaeg...
--MIME_boundary--
```

Waiting Time Detection Alarm with Picture URL

```
{
  "ipAddress": "",
  "ipv6Address": "",
  "portNo": ,
  "protocol": "",
  "macAddress": "",
  "channelID": "",
  "relatedChannelList": [1,2,3],
  /*optional, array of integers, list of alarm related channels, which are of the
  same camera with channelID; this parameter is used for live view or playback on
  the platform*/
  "dateTime": "",
  "activePostCount": ,
  "eventType": "",
  /*required, string, occurred event type, here it should be set to
  "personQueueTime", and the maximum value size is 128 bytes*/
  "eventState": "",
  "eventDescription": "",
  "deviceID": "",
  "humanCapture": {
  /*captured human information*/
    "human":{
  /*human attributes*/
      "targetID": ,
  /*required, integer, person target ID, whose waiting time is longer than the
  configured alarm threshold. The ID is generated by DSP, and corresponds to the
  person position*/
      "waitTime": ,
  /*required, integer, actual waiting time, it is ranging from 0 to 3600s*/
      "Region": [{
  /*required, target person region coordinates, it is quadrilateral*/
        "x": ,
  /*required, float, x-coordinate, it is ranging from 0.000 to 1*/
        "y":
  /*required, float, y-coordinate, it is ranging from 0.000 to 1*/
      }],
      "humanImageURL": ""
  /*captured human thumbnail URL*/
    },
    "rule":{
  /*rule properties*/
      "ruleID": ,
  /*required, integer, rule ID, which corresponds to the target person region ID*/
      "alarmTime": ,
  /*required, integer, configured waiting time alarm threshold, it is ranging
  from 0 to 3600s*/
      "regionColor": "",
  /*required, string, rule region color, "blue", "red", and "yellow" are
  available*/
```

```
    "Region": [{
/*required, rule region coordinates, it is polygonal (3 to 10 edges)*/
        "x": ,
/*required, float, x-coordinate, it is ranging from 0.000 to 1*/
        "y":
/*required, float, y-coordinate, it is ranging from 0.000 to 1*/
    }]
    "timeTriggerType": ""
/*optional, string type, triggering type of waiting time detection alarm:
"greater", "less", "equal", "unequal"-the alarm is triggered when waiting time
is larger than, smaller than, equal to, or not equal to the value of alarmTime*/
    },
    "backgroundImageURL": ""
/*background picture URL*/
}
```

5.5 JSON_PersonQueueCountingDescription

JSON message about conditions of searching for people queuing-up reports

```
{
  "PersonQueueCountingDescription": {
/*required, search condition of people queuing-up report*/
    "fileType": "",
/*optional, string, file type, opt="excel, txt"*/
    "reportType": "",
/*required, string, report type, opt="daily, weekly, monthly, custom"*/
    "startTime": "",
/*required, IOS8601 time, search start time, e.g., "2004-05-03T00:00:00+08:00"
or "2004-05-03T00:00:00.000Z"*/
    "RuleList": [{
/*required, rule information, up to 3 rules can be supported*/
        "ruleID": ,
/*required, integer, rule ID, which corresponds to the region ID when
configuring rule*/
    }],
    "CountRangeList" : [{
/*required, array, people queuing-up data of all regions under a rule, up to 3
regions are supported*/
        "minCount": ,
/*optional, integer, minimum number of queuing-up people, range: from 0 to
60*/
        "maxCount": ,
/*optional, integer, maximum number of queuing-up people, range: from 0 to 60*/
    }]
    "endTime": "",
/*dep, IOS8601t type, end time of report search, e.g.,
2004-05-03T00:00:00+08:00, it is valid only when the reportType is "custom"*/
    "customReportAccuracy": "day"
  }
}
```

```
/*dep, cuatom report accuracy, currently, it only corrects to day, and it is
valid only when the reportType is "custom"*/
}
}
```

Note

At least one of the minCount and maxCount fields is required, if only minCount is configured, it indicates no upper-limit, e.g., minCount = 10, maxCount is reserved, the statistics data more than 10 people will be searched; if only maxCount is configured, it indicates no lower-limit, e.g., maxCount = 10, minCount is reserved, the statistics data less than 10 persons will be searched.

Example

Message Example

```
{
  "PersonQueueTimeDescription":{
    "fileType":"excel",
    "reportType":"daily",
    "startTime":"2017-1-14T00:00:00Z",
    "searchType":"rule",
    "RuleList":[{
      "ruleID":1
    }],
    "CountList":[{
      "minCount":0,
      "maxCount":20
    },
    {
      "minCount":20,
      "maxCount":40
    }
  ]
}
```

5.6 JSON_PersonQueueCountingFileResult

JSON message about results of searching for people queuing-up report files

```
{
  "PersonQueueCountingFileResult":
/*required, search result of people queuing-up report*/
{
  "fileURL": ""
/*file URL of people queuing-up report*/
}
}
```

Example

Message Example

```
{
  "PersonQueueCountingFileResult": {
    "fileURL": "http://10.10.36.163/ISAPI/Intelligent/DownloadFile?filePath=/home/Day_CountFile_2017-01-14T00:00:00.xls&"
  }
}
```

5.7 JSON_PersonQueueCountingResult

JSON message about results of searching for people queuing-up report details

```
{
  "PersonQueueCountingResult": {
    /*required, search result of people queuing-up detection report*/
    "RuleList" : [{
      /*required, array, people queuing-up detection reports of all regions under multiple rules, up to 3 rules are supported*/
      "ruleID": ,
      /*required, integer, rule ID, which corresponds to the region ID when configuring rules*/
      "CountRangeList" : [{
        /*required, array, people queuing-up data of all regions under a single rule, up to 3 regions are supported*/
        "minCount": ,
        /*optional, integer, minimum number of queuing-up people, range: from 0 to 60*/
        "maxCount": ,
        /*optional, integer, maximum number of queuing-up people, range: from 0 to 60*/
        "CountingInfoList": [{
          /*required, people queuing-up report details*/
          "startTime": "",
          /*required, IOS8601 time, start time of the time period, e.g., 2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
          "endTime": "",
          /*required, IOS8601 time, end time of the time period, e.g., 2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
          "count":
          /*required, integer, number of queuing-up persons: between 0 to 60*/
        }]
      }]
    }]
  }
}
```

Note

- At least one of the minCount and maxCount fields is required, if only minCount is configured, it indicates no upper-limit, e.g., minCount = 10, maxCount is reserved, the statistics data more

than 10 persons will be searched; if only maxCount is configured, it indicates no lower-limit, e.g., maxCount = 10, minCount is reserved, the statistics data less than 10 persons will be searched.

- For daily report, one result will be returned at each hour, total 24 results; for weekly report, one result will be returned at each day, total 7 results; for monthly report, one result will be returned at each day, total 30 results.

Example

Message Example

```
{
  "PersonQueueCountingDescription":{
    "reportType":"daily",
    "startTime":"2017-12-21T00:00:00Z",
    "searchType":"rule",
    "RuleList":[{"ruleID":1
    }],
    "WaitTimeList":[{"minWaitTime":600,
    "maxWaitTime":1000
    }]
  }
}
```

5.8 JSON_PersonQueueDetectionInfo

JSON message about people queuing-up parameters

```
{
  "PersonQueueDetectionInfo": {
    /*required, queue management configuration information*/
    "countingEnabled": ,
    /*required, boolean, enable people queuing-up detection*/
    "queueTimeEnabled": ,
    /*required, boolean, enable waiting time detection*/
    "RuleInfo": [{
    /*required, queue management rule information, the number of rules is ranging
    from 1 to 3*/
      "ruleID": ,
    /*required, integer, rule ID*/
      "ruleName": "",
    /*required, string, rule name, string size: between 0 and 32*/
      "OSDOverlayEnabled": ,
    /*required, boolean, enable OSD*/
      "alarmIntervalTime": ,
    /*required, integer, alarm time interval, range: between 0 to 3600s, default:
    300s*/
    "PersonQueueCounting":{
    /*required, people queuing-up parameters*/
```

```
    "alarmCount": ,
    /*required, integer, number threshold of persons for triggering alarm, range:
    between 0 and 60 persons, default: 10*/
    "countTriggerType": "",
    /*optional, string type, triggering type of people queuing-up alarm: "greater",
    "less", "equal", "unequal"-the alarm is triggered when the number of queuing-up
    persons is larger than, smaller than, equal to, or not equal to the value of
    alarmCount; "range"-the alarm is triggered when the number of queuing-up person
    is in the range between maxAlarmCount and alarmCount*/
    "maxAlarmCount": ,
    /*dep, integer type, maximum number of queuing-up persons to trigger alarm, it
    is valid when countTriggerType is "range"*/
    "durationTime":
    /*optional, integer type, queuing-up status (number of queuing-up persons)
    keeping duration, which is between 0 and 3600, the default duration is 300,
    unit: second; the people queuing-up alarm is triggered if the number of queuing-
    up persons keeps for the configured duration*/
    },
    "PersonQueueTime":{
    /*required, waiting time detection parameters*/
    "alarmTime": ,
    /*required, integer, waiting time threshold for triggering alarm, range:
    between 0 and 3600s, default: 600s*/
    "timeTriggerType": ""
    /*optional, string type, triggering type of waiting time detection alarm:
    "greater", "less", "equal", "unequal"-the alarm is triggered when waiting time
    is larger than, smaller than, equal to, or not equal to the value of alarmTime*/
    },
    "regionColor": "",
    /*required, string, color of region that configured with rule, "blue, red,
    yellow"*/
    "Region": [{
    /*required, coordinates of region (polygon, 3 to 10 sides) that configured with
    rule*/
    "x": ,
    /*required, float, x-coordinate, range: between 0.000 and 1*/
    "y":
    /*required, float, y-coordinate, range: between 0.000 and 1*/
    }]
    }
    "alarmPicEnabled": ,
    /*optional, boolean, whether to upload pictures (i.e., background picture and
    human body thumbnail) when alarm is triggered, if this node is not supported by
    device, the alarm pictures will be uploaded by default*/
    "realTimeEnabled":
    /*optional, boolean, whether to upload real-time data of queue management, the
    corresponding event type is "personQueueRealTime"*/
    }
  }
```

Example

Message Example

```
{
  "PersonQueueDetectionInfo": {
    "countingEnabled": false,
    "queueTimeEnabled": true,
    "RuleInfo": [{
      "ruleID": 1,
      "ruleName": "East Window",
      "OSDOverlayEnabled": true,
      "alarmIntervalTime": 300,
      "PersonQueueCounting": {
        "alarmCount": 5,
        "countTriggerType": "range",
        "maxAlarmCount": 5,
        "durationTime": 300
      },
      "PersonQueueTime": {
        "alarmTime": 300,
        "timeTriggerType": "greater"
      },
      "regionColor": "yellow",
      "Region": [{
        "x": 0.411000,
        "y": 0.325000
      },
      {
        "x": 0.250000,
        "y": 0.643000
      },
      {
        "x": 0.435000,
        "y": 0.751000
      },
      {
        "x": 0.576000,
        "y": 0.675000
      },
      {
        "x": 0.647000,
        "y": 0.501000
      }
    ]
  },
  {
    "ruleID": 2,
    "ruleName": "Area2",
    "OSDOverlayEnabled": true,
    "alarmIntervalTime": 300,
    "PersonQueueCounting": {
      "alarmCount": 5
    },
    "PersonQueueTime": {
      "alarmTime": 300
    }
  },
}
```

```
    "regionColor": "yellow",
    "Region": [{
      "x": 0.106000,
      "y": 0.218000
    },
    {
      "x": 0.078000,
      "y": 0.383000
    },
    {
      "x": 0.282000,
      "y": 0.429000
    },
    {
      "x": 0.369000,
      "y": 0.204000
    },
    {
      "x": 0.188000,
      "y": 0.019000
    }
  ]
}
"alarmPicEnabled": true,
"realTimeEnabled": true
}
```

5.9 JSON_PersonQueueDetectionInfoCap

JSON message about configuration capability of people queuing-up

```
{
  "PersonQueueDetectionInfoCap": {
    /*required, queue management capability*/
    "countingEnabled": "true,false",
    /*required, boolean, whether to enable people queuing-up*/
    "queueTimeEnabled": "true,false",
    /*required, boolean, whether to enable waiting time detection*/
    "RuleInfoCap": {
      /*required, queue management rule information, the number of rules is ranging
      from 1 to 3*/
      "maxSize": 3,
      /*required, integer, capability of maximum number of supported queue management
      rules*/
      "ruleID": {
        /*required, integer, rule ID*/
        "@min": 1,
        "@max": 3
      },
      "ruleName": {
```

```
/*required, string, rule name, the string size is ranging from 0 to 32*/
    "@min": 0,
    "@max": 32
  },
  "OSDOverlayEnabled": "true,false", /*required, boolean, enable
OSD*/
  "alarmIntervalTime": {
/*required, integer, alarm time interval, range: between 0 and 3600s, default:
300s*/
    "@min": 0,
    "@max": 3600,
    "#text": 300
  },
  "PersonQueueCounting":{
/*required, people queuing-up parameters*/
    "alarmCount": {
/*required, integer, number threshold of persons for triggering alarm, range:
between 0 and 60 persons, default: 10*/
      "@min": 0,
      "@max": 60,
      "#text": 10
    }
    "countTriggerType": {
/*optional, string type, triggering type of people queuing-up alarm: "greater",
"less", "equal", "unequal"-the alarm is triggered when the number of queuing-up
persons is larger than, smaller than, equal to, or not equal to the value of
alarmCount; "range"-the alarm is triggered when the number of queuing-up person
is in the range between maxAlarmCount and alarmCount*/
      "@opt": "greater,less,equal,unequal,range",
      "@def": "greater"
    },
    "maxAlarmCount": {
/*dep, integer type, maximum number of queuing-up persons to trigger alarm, it
is valid when countTriggerType is "range"*/
      "@min": 0,
      "@max": 60,
      "#text": 5
    },
    "durationTime": {
/*optional, integer type, queuing-up status (number of queuing-up persons)
keeping duration, which is between 0 and 3600, the default duration is 300,
unit: second; the people queuing-up alarm is triggered if the number of queuing-
up persons keeps for the configured duration*/
      "@min": 0,
      "@max": 3600,
      "#text": 300
    }
  },
  "PersonQueueTime":{
/*required, waiting time detection parameters*/
    "alarmTime": {
/*required, integer, waiting time threshold for triggering alarm, range:
```

```
between 0 and 3600s, default: 600s*/
    "@min": 0,
    "@max": 3600,
    "#text": 600
  }
  "timeTriggerType": {
/*optional, string type, triggering type of waiting time detection alarm:
"greater", "less", "equal", "unequal"-the alarm is triggered when waiting time
is larger than, smaller than, equal to, or not equal to the value of alarmTime*/
    "@opt": "greater,less,equal,unequal",
    "@def": "greater"
  },
  },
  "regionColor": {
/*required, string, color of region that configured with rule, "blue, red,
yellow"*/
    "@opt": "blue,red,yellow",
    "#text": "yellow"
  },
  "RegionCap": {
/*required, coordinates of region (polygon, 3 to 10 sides) that configured with
rule*/
    "minSize": 3,
/*required, integer, capability of minimum region sides*/
    "maxSize": 10,
/*required, integer, capability of maximum region sides*/
    "x": {
/*required, float, x-coordinate, range: between 0.000 and 1*/
      "@min": 0.000,
      "@max": 1.000,
      "#text": 0.120
    },
    "y": {
/*required, float, y-coordinate, range: between 0.000 and 1*/
      "@min": 0.000,
      "@max": 1.000,
      "#text": 0.120
    }
  }
  "alarmPicEnabled":{
/*optional, boolean, whether to upload pictures (i.e., background picture and
human body thumbnail) when alarm is triggered, if this node is not supported by
device, the alarm pictures will be uploaded by default*/
    "@opt": "true,false",
/*"true"-upload alarm picture (default), "false"-not upload alarm pictures*/
    "@def": "true"
  },
  "realTimeEnabled":{
/*optional, boolean, whether to upload real-time data of queue management, the
corresponding event type is "personQueueRealTime"*/
    "@opt": "true,false", /*"true"-upload real-time data (default),
"false"-not upload*/
```

```
        "@def": "true"
      }
    },
    "PersonQueueCountingCap": {
/*optional, capability of getting people queuing-up result and report*/
      "fileType": {
/*optional, string, file type, "excel, txt"*/
        "@opt": "excel,txt",
        "#text": "excel"
      },
      "reportType": {
/*required, string, report type, "daily, weekly, monthly, custom"*/
        "@opt": "daily,weekly,monthly,custom",
        "#text": "daily"
      },
      "isSupportStartTime": "true,false",
/*required, bool, whether supports setting start time of searching
reports*/
      "RuleListCap": {
/*required, rule information, up to 3 rules can be supported*/
        "maxSize": 3,
/*required, integer, capability of capability of maximum number of supported
queue management rules*/
        "ruleID": {
/*required, integer, rule ID*/
          "@min": 1,
          "@max": 3
        }
      },
      "WaitTimeListCap": {
/*required, waiting time information, up to 3 time durations are supported*/
        "maxSize": 3,
/*required, integer, supported maximum number of time durations*/
        "minWaitTime": {
/*required, integer, minimum waiting time, it ranges from 0s to 3600s*/
          "@min": 0,
          "@max": 3600
        },
        "maxWaitTime": {
/*required, integer, maximum waiting time, it ranges from 0s to 3600s*/
          "@min": 0,
          "@max": 3600
        }
      },
      "isSupportEndTime": "true,false",
/*dep, boolean, whether support the end time of report search, and it is valid
only when reportType is custom*/
      "customReportAccuracy": {
/*dep, custom report accuracy, currently, it corrects to day. And it is valid
only when reportType is custom*/
        "@opt": "day",
```

```
        "@def": "day"
    }
    "isSupportAverageTime": "true,false"
    /*optional, boolean type, whether supports calculating average waiting time
    duration in people queuing-up statistics*/
    },
    "PersonQueueTimeCap": {
    /*optional, capability of getting waiting time detection result and
    report*/
        "fileType": {
        /*optional, string, file type, "excel, txt"*/
            "@opt": "excel,txt",
            "#text": "excel"
        },
        "reportType": {
        /*required, string, report types, "daily, weekly, monthly, custom"*/
            "@opt": "daily,weekly,monthly, custom",
            "#text": "daily"
        },
        "isSupportStartTime": "true,false",
        /*required, bool, whether supports setting start time of searching reports*/
        "RuleListCap": {
        /*required, rule information, up to 3 rules can be supported*/
            "maxSize": 3,
        /*required, integer, capability of capability of maximum number of supported
        queue management rules*/
            "ruleID": {
        /*required, integer, rule ID*/
                "@min": 1,
                "@max": 3
            }
        },
        "CountListCap": {
        /*required, queuing-up people amount information, up to 3 queues are supported*/
            "maxSize": 3,
        /*required, integer, supported maximum number of queues*/
            "minCount": {
        /*required, integer, minimum number of queuing-up people, it ranges from 0 to
        60*/
                "@min": 0,
                "@max": 60
            },
            "maxCount": {
        /*required, integer, maximum number of queuing-up people, it ranges from 0 to
        60*/
                "@min": 0,
                "@max": 60
            }
        },
        "isSupportEndTime": "true,false",
        /*dep, boolean, whether support the end time of report search, and it is valid
        only when reportType is custom*/
```



```
"customReportAccuracy": {
/*dep, custom report accuracy, currently, it corrects to day. And it is valid
only when reportType is custom*/
  "@opt": "day",
  "@def": "day"
}
}
"isSupportRawDataFile": "true,false"
/*required, boolean, whether supports getting queue management raw data*/
"isSupportAlgParam": "true,false"
/*optional, boolean type, whether supports configuring advanced parameters for
algorithm library*/
}
```

5.10 JSON_PersonQueueRawDataFileDescription

JSON message about conditions of searching for queue management raw data

```
{
  "PersonQueueRawDataFileDescription": { /*required, search condition of queue
management raw data files*/
    "searchID": , /*required, integer, search ID, which is used to save the
search history for fast search at next time*/
    "searchResultPosition": , /*required, integer, start position of in the
search result list*/
    "maxResults": , /*required, integer, number of matched records, up to 4000
records can be searched for once*/
    "startTime": "", /*required, ios_8601 time, start time of search, e.g.,
2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
    "endTime": "" /*required, ios_8601 time, end time of search, e.g.,
2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
  }
}
```

Example

Message Example

```
{
  "PersonQueueRawDataFileDescription":{
    "searchID":1,
    "searchResultPosition":1,
    "maxResults":4000,
    "startTime":"2017-01-14T00:00:00.000Z",
    "endTime":"2017-01-15T00:00:00.000Z"
  }
}
```

5.11 JSON_PersonQueueRawDataFileResult

JSON message about results of searching for queue management raw data

```
{
  "PersonQueueRawDataFileResult": /*required, search result of queue management
raw file*/
  {
    "searchID": , /*required, integer, search ID, which is used to save the
search history for fast search at next time*/
    "responseStatus": "", /*required, string, search status, "OK"-receiving
completed, "NO MATCHES"-no data found, "MORE"-more data to be received at next
time*/
    "numOfMatches": , /*required, integer, number of returned records*/
    "totalMatches": , /*required, integer, total number of matched
records*/
    "endTime": "", /*required, ios_8601 time, time of returning the last data
record, e.g., 2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
    "fileURL": "" /*required, string, file URL*/
  }
}
```

Example

Message Example

```
{
  "PersonQueueRawDataFileResult": {
    "searchID": 1,
    "responseStatus": "NO MATCHES",
    "numOfMatches": 0,
    "totalMatches": 0,
    "endTime": "1970-01-01T00:00:00",
    "fileURL": "http://10.10.36.163/ISAPI/Intelligent/DownloadFile?filePath=/
home/RawFile_2017-01-14T00:00:00_1970-01-01T00:00:00.xls&"
  }
}
```

5.12 JSON_PersonQueueTimeDescription

JSON message about conditions of searching for waiting time detection reports

```
{
  "PersonQueueTimeDescription": { /*required, search condition of waiting time
detection report*/
    "fileType": "", /*optional, string, file type, "excel, txt"*/
    "reportType": "", /*required, string, report type, "daily, weekly, monthly,
custom"*/
    "startTime": "", /*required, IOS8601 time, start time of searching report,
e.g., 2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
  }
```

```
"RuleList": [{ /*required, rule information, up to 3 rules can be
supported*/
  "ruleID": , /*required, integer, rule ID, which corresponds to the
region ID when configuring rule*/
},
  "WaitTimeRangeList" : [{ /*required, array, waiting time of all time
periods under a single rule, up to 3 time periods are
supported*/
    "minWaitTime": , /*Optional, integer, minimum waiting time, range: from
0 to 3600s*/
    "maxWaitTime": , /*Optional, integer, maximum waiting time, range: from 0
to 3600s*/
  }
  "endTime": "", /*dep, IOS8601t type, end time of report search, e.g.,
2004-05-03T00:00:00+08:00, it is valid only when the reportType is "custom"*/
  "customReportAccuracy": "day" /*dep, cuatom report accuracy, currently, it
only corrects to day, and it is valid only when the reportType is "custom"*/
}
}
```



Note

At least one of minWaitTime and maxWaitTime fields is required, if only minWaitTime is configured, it indicates no upper-limit, e.g., minWaitTime=600, maxWaitTime is reserved, the waiting time which is longer than 600s will be searched; if only maxWaitTime is configured, it indicates no lower-limit, e.g., maxWaitTime=600, minWaitTime is reserved, the waiting time which is shorter than 600s will be searched.

Example

Message Example

```
{
  "PersonQueueTimeDescription":{
    "reportType":"daily","startTime":"2017-1-14T00:00:00Z",
    "searchType":"rule",
    "RuleList":[{"
      "ruleID":1
    }],
    "CountList":[{"
      "minCount":0,
      "maxCount":20
    }]
  }
}
```

5.13 JSON_PersonQueueTimeFileResult

JSON message about results of searching for waiting time detection report files

```
{
  "PersonQueueTimeFileResult": /*required, search result of waiting time
detection report*/
  {
    "fileURL": "" //file URL of waiting time detection report
  }
}
```

Example

Message Example

```
{
  "PersonQueueTimeFileResult": {
    "fileURL": "http://10.10.36.163/ISAPI/Intelligent/DownloadFile?filePath=/
home/Day_TimeFile_2017-01-14T00:00:00.xls&"
  }
}
```

5.14 JSON_PersonQueueTimeResult

JSON message about results of searching for waiting time detection report details

```
{
  "PersonQueueTimeResult": { /*required, search results of waiting time
detection report*/
    "RuleList" : [{ /*required, array, waiting time of all time periods under
multiple rules, up to 3 rules are supported*/
      "ruleID": , /*required, integer, rule ID, which corresponds to the time
period ID when configuring rules*/
      "WaitTimeRangeList" : [{ /*required, array, waiting time of all time
periods under a single rule, up to 3 time periods are
supported*/
        "minWaitTime": , /*Optional, integer, minimum waiting time, range:
from 0 to 3600s*/
        "maxWaitTime": 600, /*Optional, integer, maximum waiting time, range:
from 0 to 3600s*/
        "TimeInfoList": [{ /*required, waiting time detection report
details*/
          "startTime": "", /*required, IOS8601 time, start time of the time
period, e.g., 2004-05-03T00:00:00+08:00 or
2004-05-03T00:00:00.000Z*/
          "endTime": "", /*required, IOS8601 time, end time of the time period,
e.g., 2004-05-03T00:00:00+08:00 or 2004-05-03T00:00:00.000Z*/
          "time": 60 /*required, integer, waiting time: from 0 to 3600s*/
        }]
      }]
    }]
  }
}
```

Note

- At least one of minWaitTime and maxWaitTime fields is required, if only minWaitTime is configured, it indicates no upper-limit, e.g., minWaitTime=600, maxWaitTime is reserved, the waiting time which is longer than 600s will be searched; if only maxWaitTime is configured, it indicates no lower-limit, e.g., maxWaitTime=600, minWaitTime is reserved, the waiting time which is shorter than 600s will be searched.
 - For daily report, one result will be returned at each hour, total 24 results; for weekly report, one result will be returned at each day, total 7 results; for monthly report, one result will be returned at each day, total 30 results.
-

Example

Message Example

```
{
  "PersonQueueTimeResult": {
    "RuleList": [{
      "ruleID": 1,
      "CountRangeList": [{
        "minCount": 0,
        "maxCount": 20,
        "TimeInfoList": [{
          "time": 0,
          "startTime": "2017-01-14T00:00:00",
          "endTime": "2017-01-14T00:59:59"
        },
        {
          "time": 0,
          "startTime": "2017-01-14T01:00:00",
          "endTime": "2017-01-14T01:59:59"
        },
        {
          "time": 0,
          "startTime": "2017-01-14T02:00:00",
          "endTime": "2017-01-14T02:59:59"
        },
        {
          "time": 0,
          "startTime": "2017-01-14T03:00:00",
          "endTime": "2017-01-14T03:59:59"
        },
        {
          "time": 0,
          "startTime": "2017-01-14T04:00:00",
          "endTime": "2017-01-14T04:59:59"
        },
        {
          "time": 0,
          "startTime": "2017-01-14T05:00:00",
          "endTime": "2017-01-14T05:59:59"
        }
      ]
    }
  ]
}
```

```
},
{
  "time": 0,
  "startTime": "2017-01-14T06:00:00",
  "endTime": "2017-01-14T06:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T07:00:00",
  "endTime": "2017-01-14T07:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T08:00:00",
  "endTime": "2017-01-14T08:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T09:00:00",
  "endTime": "2017-01-14T09:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T10:00:00",
  "endTime": "2017-01-14T10:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T11:00:00",
  "endTime": "2017-01-14T11:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T12:00:00",
  "endTime": "2017-01-14T12:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T13:00:00",
  "endTime": "2017-01-14T13:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T14:00:00",
  "endTime": "2017-01-14T14:59:59"
},
{
  "time": 0,
  "startTime": "2017-01-14T15:00:00",
  "endTime": "2017-01-14T15:59:59"
},
{
```

```
        "time": 0,
        "startTime": "2017-01-14T16:00:00",
        "endTime": "2017-01-14T16:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T17:00:00",
        "endTime": "2017-01-14T17:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T18:00:00",
        "endTime": "2017-01-14T18:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T19:00:00",
        "endTime": "2017-01-14T19:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T20:00:00",
        "endTime": "2017-01-14T20:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T21:00:00",
        "endTime": "2017-01-14T21:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T22:00:00",
        "endTime": "2017-01-14T22:59:59"
    },
    {
        "time": 0,
        "startTime": "2017-01-14T23:00:00",
        "endTime": "2017-01-14T23:59:59"
    }
  ]
}
```

5.15 JSON_ResponseStatus

JSON message about response status

```
{
  "requestURL": "",
```

```
/*optional, string, request URL*/
    "statusCode": ,
/*optional, int, status code*/
    "statusString": "",
/*optional, string, status description*/
    "subStatusCode": "",
/*optional, string, sub status code*/
    "errorCode": ,
/*required, int, error code, which corresponds to subStatusCode, this field is
required when statusCode is not 1. The returned value is the transformed
decimal number*/
    "errorMsg": "",
/*required, string, error details, this field is required when statusCode is
not 1*/
    "MErrCode": "0xFFFFFFFF",
/*optional, string, error code categorized by functional modules*/
    "MErrDevSelfEx": "0xFFFFFFFF"
/*optional, string, extension of MErrCode. It is used to define the custom
error code, which is categorized by functional modules*/
}
```

5.16 XML_Cap_Schedule

Schedule capability message in XML format

```
<Schedule version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">
  <id><!--req, xs:string, ID--></id>
  <eventType
opt="IO,VMD,videoloss,PIR,linedetection,fieldddetection,audioexception,facedetect
ion,regionEntrance,regionExiting,loitering,group,rapidMove,parking,unattendedBag
gage,attendedBaggage,storageDetection,shipsDetection,HUMANATTRIBUTE,humanAttribu
te,faceContrast,faceSnap, mixedTargetDetection">
    <!--opt, xs:string, event type-->
  </eventType>
  <inputIOPortID><!--ro, dep, xs:string, alarm input ID--></inputIOPortID>
  <outputIOPortID><!--ro, dep, xs:string, alarm output ID--></inputIOPortID>
  <videoInputChannelID><!--ro, dep, xs:string, video input channel ID--></
videoInputChannelID>
  <TimeBlockList size="8"><!--req-->
    <TimeBlock>
      <dayOfWeek opt="1,2,3,4,5,6,7">
        <!--opt, xs:integer, days of the week in ISO8601 time format, 1-Monday,
2-Tuesday, 3-Wednesday, 4-Thursday, 5-Friday, 6-Saturday, 7-Sunday-->
      </dayOfWeek>
      <TimeRange><!--req, time periods in arming schedule-->
        <beginTime><!--req, xs:time, start time of time period, in ISO8601 time
format--></beginTime>
        <endTime><!--req, xs:time, end time of time period, in ISO8601 time
format--></endTime>
      </TimeRange>
    </TimeBlock>
  </TimeBlockList>
</Schedule>
```



```
</TimeBlock>
</TimeBlockList>
<HolidayBlockList><!--opt, holiday arming period-->
  <TimeBlock>
    <TimeRange><!--req, time period-->
      <beginTime><!--req, xs:time, start time, in ISO8601 time format--></
beginTime>
      <endTime><!--req, xs:time, end time, in ISO8601 time format--></
endTime>
    </TimeRange>
  </TimeBlock>
</HolidayBlockList>
</Schedule>
```

5.17 XML_EventNotificationAlert_AlarmEventInfo

EventNotificationAlert message with alarm/event information in XML format.

```
<EventNotificationAlert version="2.0" xmlns="http://www.isapi.org/ver20/
XMLSchema">
  <ipAddress><!--dep, xs:string, device IPv4 address--></ipAddress>
  <ipv6Address><!--dep, xs:string, device IPv6 address--></ipv6Address>
  <portNo><!--opt, xs:integer, device port number--></portNo>
  <protocol><!--opt, xs:string, protocol type for uploading alarm/event
information, "HTTP,HTTPS"--></protocol>
  <macAddress><!--opt, xs:string, MAC address--></macAddress>
  <channelID><!--dep, xs:string, device channel No., starts from 1--></
channelID>
  <dateTime><!--req, alarm/event triggered or occurred time, format:
2017-07-19T10:06:41+08:00--></dateTime>
  <activePostCount><!--req, xs:integer, alarm/event frequency, starts from 1--
></activePostCount>
  <eventType><!--req, xs:string, alarm/event type, "peopleCounting, ANPR,..."--
></eventType>
  <eventState>
    <!--req, xs:string, durative alarm/event status: "active"-valid, "inactive"-
invalid, e.g., when a moving target is detected,
    the alarm/event information will be uploaded continuously unit the status
is set to "inactive"-->
  </eventState>
  <eventDescription><!--req, xs:string, alarm/event description--></
eventDescription>
  <...><!--opt, for different alarm/event types, the nodes are different, see
the message examples in different applications--></...>
</EventNotificationAlert>
```

5.18 XML_EventTriggersCap

XML message about linkage capabilities of different alarm categories

```
<EventTriggersCap version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">
  <DiskfullTriggerCap><!--optional, xs: EventTriggerCapType--></
DiskfullTriggerCap>
  <DiskerrorTriggerCap><!--optional, xs: EventTriggerCapType--></
DiskerrorTriggerCap>
  <NicbrokenTriggerCap><!--optional, xs: EventTriggerCapType--></
NicbrokenTriggerCap>
  <IpconflictTriggerCap><!--optional, xs: EventTriggerCapType--></
IpconflictTriggerCap>
  <IllacesTriggerCap><!--optional, xs: EventTriggerCapType--></
IllacesTriggerCap>
  <BadvideoTriggerCap><!--optional, xs: EventTriggerCapType--></
BadvideoTriggerCap>
  <VideomismatchTriggerCap><!--optional, xs: EventTriggerCapType--></
VideomismatchTriggerCap>
  <IOTriggerCap><!--optional, xs: EventTriggerCapType--></IOTriggerCap>
  <LineDetectTriggerCap><!--optional, xs: EventTriggerCapType--></
LineDetectTriggerCap>
  <RegionEntranceTriggerCap><!--optional, xs: EventTriggerCapType--></
RegionEntranceTriggerCap>
  <RegionExitingTriggerCap><!--optional, xs: EventTriggerCapType--></
RegionExitingTriggerCap>
  <LoiteringTriggerCap><!--optional, xs: EventTriggerCapType--></
LoiteringTriggerCap>
  <GroupDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
GroupDetectionTriggerCap>
  <RapidMoveTriggerCap><!--optional, xs: EventTriggerCapType--></
RapidMoveTriggerCap>
  <ParkingTriggerCap><!--optional, xs: EventTriggerCapType--></
ParkingTriggerCap>
  <UnattendedBaggageTriggerCap><!--optional, xs: EventTriggerCapType--></
UnattendedBaggageTriggerCap>
  <AttendedBaggageTriggerCap><!--optional, xs: EventTriggerCapType--></
AttendedBaggageTriggerCap>
  <FireDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
FireDetectionTriggerCap>
  <FireDetectionCap><!--optional, xs: EventTriggerCapType--></FireDetectionCap>
  <StorageDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
StorageDetectionTriggerCap>
  <ShipsDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
ShipsDetectionTriggerCap>
  <ThermometryCap><!--optional, xs: EventTriggerCapType--></ThermometryCap>
  <VandalProofTriggerCap><!--optional, xs: EventTriggerCapType--></
VandalProofTriggerCap>
  <BlackListTriggerCap><!--opt, xs: EventTriggerCapType, configuration
capability of blocklist arming linkage--></BlackListTriggerCap>
  <WhiteListTriggerCap><!--opt, xs: EventTriggerCapType, configuration
```

```
capability of allowlist arming linkage--></WhiteListTriggerCap>
  <AllVehicleListTriggerCap><!--optional, xs:EventTriggerCapType, configuration
capability of other list arming linkage--></AllVehicleListTriggerCap>
  <OtherVehicleListTriggerCap><!--optional, xs:EventTriggerCapType--></
OtherVehicleListTriggerCap>
  <PeopleDetectionTriggerCap><!--optional, xs:EventTriggerCapType--></
PeopleDetectionTriggerCap>
  <PIRAAlarmCap><!--optional, xs: EventTriggerCapType--></PIRAAlarmCap>
  <TamperDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
TamperDetectionTriggerCap>
  <DefocusDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
DefocusDetectionTriggerCap>
  <FaceDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
FaceDetectionTriggerCap>
  <SceneChangeDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
SceneChangeDetectionTriggerCap>
  <VandalProofAlarmCap><!--optional, xs: EventTriggerCapType--></
VandalProofAlarmCap>
  <JudgmentTriggerCap><!--optional, xs: EventTriggerCapType--></
JudgmentTriggerCap>
  <FightingTriggerCap><!--optional, xs: EventTriggerCapType--></
FightingTriggerCap>
  <RisingTriggerCap><!--optional, xs: EventTriggerCapType--></RisingTriggerCap>
  <DozingTriggerCap><!--optional, xs: EventTriggerCapType--></DozingTriggerCap>
  <CountingTriggerCap><!--optional, xs: EventTriggerCapType--></
CountingTriggerCap>
  <VideoLossTriggerCap><!--optional, xs: EventTriggerCapType--></
VideoLossTriggerCap>
  <HideTriggerCap><!--optional, xs:EventTriggerCapType--></HideTriggerCap>
  <AlarmInTriggerCap><!--optional, xs: EventTriggerCapType--></
AlarmInTriggerCap>
  <VehicleDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
VehicleDetectionTriggerCap>
  <AudioExceptionCap><!--optional, xs: EventTriggerCapType--></
AudioExceptionCap>
  <FiledDetectTriggerCap><!--optional, xs: EventTriggerCapType--></
FiledDetectTriggerCap>
  <MotionDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
MotionDetectionTriggerCap>
  <TemperatureCap><!--optional, xs: EventTriggerCapType--></TemperatureCap>
  <IntelligentTriggerCap><!--optional, xs: EventTriggerCapType--></
IntelligentTriggerCap>
  <FaceContrastTriggerCap><!--optional, xs: EventTriggerCapType, face picture
comparison alarm linkage--></FaceContrastTriggerCap>
  <PersonDensityDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
PersonDensityDetectionTriggerCap>
  <PersonQueueDetectionTriggerCap><!--optional, xs: EventTriggerCapType, queue
management alarm linkage--></PersonQueueDetectionTriggerCap>
  <HumanRecognitionTriggerCap><!--optional, xs: EventTriggerCapType--></
HumanRecognitionTriggerCap>
  <FaceSnapTriggerCap><!--optional, xs: EventTriggerCapType--></
FaceSnapTriggerCap>
```

```
<isSupportWhiteLightAction>
  <!--dependent, xs: boolean, see details in EventTriggerCapType, it is valid
when isSupportWhiteLight is "true"-->
</isSupportWhiteLightAction>
<isSupportAudioAction>
  <!--dependent, xs: boolean, see details in EventTriggerCapType, it is valid
when isSupportBeep is "true"-->
</isSupportAudioAction>
<HFPDTriggerCap><!--optional, xs: EventTriggerCapType--></HFPDTriggerCap>
<MixedTargetDetectionCap><!--optional, xs: EventTriggerCapType--></
MixedTargetDetectionCap>
<HVTVehicleDetectionTriggerCap><!--optional, xs: EventTriggerCapType--></
HVTVehicleDetectionTriggerCap>
<VCATriggerCap><!--optional, xs: EventTriggerCapType--></VCATriggerCap>
<PIRCap><!--optional, xs: EventTriggerCapType--></PIRCap>
<IllegalParkingTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports illegal parking detection--></IllegalParkingTriggerCap>
<PedestrianTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports pedestrian detection--></PedestrianTriggerCap>
<TrafficAccidentTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports traffic accident detection--></TrafficAccidentTriggerCap>
<ConstructionTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports construction detection--></ConstructionTriggerCap>
<RoadBlockTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports roadblock detection--></RoadBlockTriggerCap>
<AbandonedObjectTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports objects dropped down detection--></AbandonedObjectTriggerCap>
<ParallelParkingTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports parallel parking detection--></ParallelParkingTriggerCap>
<ParkingStateTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports parking space status detection, currently this node is not supported--
></ParkingStateTriggerCap>
<CongestionTriggerCap><!--optional, xs: EventTriggerCapType, whether it
supports congestion detection--></CongestionTriggerCap>
<IntersectionAnalysisCap><!--optional, xs: EventTriggerCapType, whether it
supports intersection analysis--></IntersectionAnalysisCap>
<ShipsFlowDetectionTriggerCap><!--optional,xs:EventTriggerCapType, ship flow
detection--></ShipsFlowDetectionTriggerCap>
<dredgerDetectionTriggerCap><!--optional,xs:EventTriggerCapType, dredger
detection--></dredgerDetectionTriggerCap>
<voltageInstableTriggerCap><!--optional,xs:EventTriggerCapType, supply
voltage exception--></voltageInstableTriggerCap>
<HighHDDTemperatureTriggerCap><!--optional, xs:EventTriggerCapType, HDD high
temperature detection--></HighHDDTemperatureTriggerCap>
<LowHDDTemperatureTriggerCap><!--optional, xs:EventTriggerCapType, HDD low
temperature detection--></LowHDDTemperatureTriggerCap>
<HDImpactTriggerCap><!--optional, xs:EventTriggerCapType, HDD impact
detection--></HDImpactTriggerCap>
<HDBadBlockTriggerCap><!--optional, xs:EventTriggerCapType, HDD bad sector
detection--></HDBadBlockTriggerCap>
<SevereHDDFailureTriggerCap><!--optional, xs:EventTriggerCapType, HDD severe
fault detection--></SevereHDDFailureTriggerCap>
```

```
<HUMANATTRIBUTECap><!--optional, xs:EventTriggerCapType--></HUMANATTRIBUTECap>
<HumanAttributeTriggerCap><!--optional, xs:EventTriggerCapType, human body
attribute--></HumanAttributeTriggerCap>
<BlackListFaceContrastTriggerCap><!--opt, xs:EventTriggerCapType, alarm
linkage capability of blacklist face comparison--></
BlackListFaceContrastTriggerCap>
<FaceLibTriggerCap><!--optional, xs:EventTriggerCapType--></FaceLibTriggerCap>
<SafetyHelmetDetectionTriggerCap><!--optional, xs:EventTriggerCapType, alarm
linkage capability of hard hat detection--></SafetyHelmetDetectionTriggerCap>
<VibrationDetectionTriggerCap><!--optional, xs:EventTriggerCapType, alarm
linkage capability of vibration detection--></VibrationDetectionTriggerCap>
<RadarLineDetectionTriggerCap><!--optional, xs:EventTriggerCapType, alarm
linkage capability of radar line crossing detection--></
RadarLineDetectionTriggerCap>
<RadarFieldDetectionTriggerCap><!--optional, xs:EventTriggerCapType, alarm
linkage capability of radar intrusion detection--></
RadarFieldDetectionTriggerCap>
<HBDLibTriggerCap><!--optional, xs:EventTriggerCapType, alarm linkage
capability of human body picture library--></HBDLibTriggerCap>
<FaceThermometryCap><!--optional, xs:EventTriggerCapType--></
FaceThermometryCap>
<NoMaskDetectionTriggerCap><!--optional, xs:EventTriggerCapType, alarm
linkage capability of no wearing mask detection--></NoMaskDetectionTriggerCap>
<TMPATriggerCap><!--optional, xs:EventTriggerCapType, alarm linkage
capability of temperature measurement pre-alarm--></TMPATriggerCap>
<FireEscapeDetectionTriggerCap><!--optional, xs:EventTriggerCapType, alarm
linkage capability of fire engine access detection--></
FireEscapeDetectionTriggerCap>
<TakingElevatorDetectionTriggerCap><!--optional, xs:EventTriggerCapType,
alarm linkage capability of elevator detection--></
TakingElevatorDetectionTriggerCap>
<RuleTriggerCap><!--optional, linkage capability of rule triggered alarm -->
  <isSupportCityManagement>
    <!--optional, xs:boolean, whether the city management supports setting
linkage actions by area; if supports, the value is true, otherwise, this node
will not be returned-->
  </isSupportCityManagement>
</RuleTriggerCap>
<ThermalCalibrationFileExceptionCap><!--optional, xs:EventTriggerCapType,
alarm linkage capability of thermography calibration file exception--></
ThermalCalibrationFileExceptionCap>
</EventTriggersCap>
```

See Also

[XML_EventTriggerCapType](#)

5.19 XML_EventTriggerCapType

XML message about capability of alarm linkage action types

```
<EventTriggerCapType version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">
  <isSupportCenter><!--optional, xs:boolean--></isSupportCenter>
  <isSupportRecord><!--optional, xs:boolean--></isSupportRecord>
  <isSupportMonitorAlarm><!--optional, xs:boolean--></isSupportMonitorAlarm>
  <isSupportBeep><!--optional, xs: boolean, whether it supports audible
warning--></isSupportBeep>
  <isSupportIO><!--optional, xs:boolean--></isSupportIO>
  <isSupportFTP><!--optional, xs:boolean--></isSupportFTP>
  <isSupportEmail><!--optional, xs:boolean--></isSupEmail>
  <isSupportLightAudioAlarm><!--optional, xs:boolean--></
isSupportLightAudioAlarm>
  <isSupportFocus><!--optional, xs:boolean--></isSupportFocus>
  <isSupportPTZ><!--optional, xs:boolean--></isSupportPTZ>
  <maxPresetActionNum>
    <!--dependent, xs:integer, it is valid only when <isSupportPTZ> is "true"-->
  </maxPresetActionNum>
  <maxPatrolActionNum>
    <!--dependent, xs:integer, it is valid only when <isSupportPTZ> is "true"-->
  </maxPatrolActionNum>
  <maxPatternActionNum>
    <!--dependent, xs:integer, it is valid only when <isSupportPTZ> is "true"-->
  </maxPatternActionNum>
  <isSupportTrack><!--optional, xs:boolean, whether it supports PTZ linked
tracking--></isSupportTrack>
  <isSupportWhiteLight>
    <!--optional, xs: boolean, whether it supports supplement light alarm
linkage-->
  </isSupportWhiteLight>
  <isSupportCloud><!--optional, xs:boolean, whether it supports upload to the
cloud--></isSupportCloud>
  <targetNotificationInterval max="1000" min="0" default="30"><!--xs:integer,
range: [0, 1000], the default value is 30, unit: seconds, this node is valid
for <MotionDetectionTriggerCap> and <TamperDetectionTriggerCap> and this node
is valid when <isSupportPTZ> is "true"--></targetNotificationInterval>
  <direction opt="both,forward,reverse"><!--xs:string, triggering direction,
this node is valid for the node <BlackListTriggerCap>, <WhiteListTriggerCap>,
and <VehicleDetectionTriggerCap>--></direction>
  <presetDurationTime min="" max=""><!--dependent, xs:integer--></
presetDurationTime>
  <isSupportSMS><!--optional, xs:boolean, whether to support SMS (Short Message
Service)--></isSupportSMS>
  <maxCellphoneNum><!--dependent, xs:integer, the maximum number of cellphones,
which is node is valid only when <isSupportSMS> is "true"--></maxCellphoneNum>
  <isSupportOSD><!--optional, xs:boolean--></isSupportOSD>
  <isSupportAudio><!--optional, xs:boolean, whether it supports setting audio
alarm independently. If this node is set to "true", audio alarm and buzzer
alarm can be linked separately, and the linage method is audio--></
isSupportAudio>
  <AudioAction><!--dependent, this node is valid when <isSupportBeep> is "true"
or <isSupportAudio> is "true"-->
  <audioTypeList>
    <audioType><!--list-->
```

```
<audioID><!--required, xs:integer, alarm sound type--></audioID>
<audioDescription><!--required, xs:string, alarm sound description, it
should correspond to the alarm sound type--></audioDescription>
</audioType>
</audioTypeList>
<alarmTimes opt="0,1,2,3,4,5,6,7,8,9,255"><!--required, xs:integer, alarm
times, it is between 0 and 9, 255-continuous alarm, unit: time--></alarmTimes>
</AudioAction>
<isSupportSMS><!--optional, xs:boolean --></isSupportSMS>
<maxCellphoneNum><!--dependent, if <isSupportSMS> is true, xs:integer--></
maxCellphoneNum>
<isNotSupportCenterModify><!--optional, xs:boolean, whether editing
configuration parameters of the monitoring center is not supported: "true"-yes
(configuration parameters of the monitoring center cannot be edited), "false"
or this node is not returned-no (configuration parameters of the monitoring
center can be edited)--></isNotSupportCenterModify>
<isSupportMessageConfig>
  <!--optional, xs:boolean, whether it supports SMS configuration, if
supports, set cellphoneNumber to null-->
</isSupportMessageConfig>
<isSupportAnalogOutput><!--optional, xs:boolean, whether it supports IO
output of linkage analog channel--></isSupportAnalogOutput>
<isSupportIOOutputUnify><!--optional, xs:boolean, whether it supports
configuration of IO output--></isSupportIOOutputUnify>
<isSupportFaceContrast><!--optional, xs:boolean, whether it supports face
picture comparison linkage--></isSupportFaceContrast>
<isSupportSiren><!--optional, xs:boolean, whether it supports siren linkage--
></isSupportSiren>
<isSupportOutput><!--optional, xs:boolean, whether it supports relay linkage--
></isSupportOutput>
</EventTriggerCapType>
```

5.20 XML_IntelliCap

XML message about intelligent capability

```
<IntelliCap version="2.0" xmlns="http://www.isapi.org/ver20/XMLSchema">
  <isFaceSupport><!--optional, xs:boolean, whether it supports face detection--
></isFaceSupport>
  <isBehaviorSupport><!--optional, xs:boolean, whether it supports behavior
analysis--></isBehaviorSupport>
  <isLineDetectionSupport><!--optional, xs:boolean, whether it supports line
crossing detection--></isLineDetectionSupport>
  <isFieldDetectionSupport><!--optional, xs:boolean, whether it supports
intrusion detection--></isFieldDetectionSupport>
  <isRegionEntranceSupport><!--optional, xs:boolean, whether it supports
region entrance detection--></isRegionEntranceSupport>
  <isRegionExitingSupport><!--optional, xs:boolean, whether it supports region
exiting detection--></isRegionExitingSupport>
  <isLoiteringSupport><!--optional, xs:boolean, whether it supports loitering
```

```
detection--></isLoiteringSupport>
  <isGroupSupport><!--optional,  xs:boolean, whether it supports people
gathering detection--></isGroupSupport>
  <isRapidMoveSupport><!--optional,  xs:boolean, whether it supports fast
moving detection--></isRapidMoveSupport>
  <isParkingSupport><!--optional,  xs:boolean, whether it supports parking
detection--></isParkingSupport>
  <isUnattendedBaggageSupport><!--optional,  xs:boolean, whether it supports
unattended baggage detection--></isUnattendedBaggageSupport>
  <isAttendedBaggageSupport><!--optional,  xs:boolean, whether it supports
object removal detection--></isAttendedBaggageSupport>
  <isTeacherSupport><!--optional,  xs:boolean, whether it supports teacher's
behavior detection--></isTeacherSupport>
  <isStudentSupport><!--optional,  xs:boolean, whether it supports student's
behavior detection--></isStudentSupport>
  <isCombinedSupport><!--optional,  xs:boolean, whether it supports combined
detection--></isCombinedSupport>
  <isTrafficSupport><!--optional,  xs:boolean, whether it supports traffic
detection--></isTrafficSupport>
  <RestoreLib>
    <libName opt="studentsStoodUp,peopleCounting,shipDetection"><!--optional,
xs:string,--></libName>
  </RestoreLib>
  <RestoreDefParamForbid>
    <!--optional,  xs:boolean "true, false", forbid restoring algorithm library
to default. When the node does not exist, it indicates restoring to default is
not supported. When the node exists and its value is "ture", it indicates
support, when the value is "false", it indicates not support-->
  </RestoreDefParamForbid>
  <RestoreAlgLibParam>
    <!--optional,  the capabilities of restoring algorithm library parameters
according to algorithm library name-->
    <libName opt="faceSnap,HMS,behaviorAnalysis"><!--optional,  xs:string--></
libName>
  </RestoreAlgLibParam>
  <isFaceCaptureStatisticsSupport><!--whether it supports face picture
statistics--></isFaceCaptureStatisticsSupport>
  <isSupportPersonQueueDetection><!--whether it supports queue management--></
isSupportPersonQueueDetection>
  <isSupportIntersectionAnalysis>
    <!--optional,  xs: boolean, whether it supports intersection analysis-->
  </isSupportIntersectionAnalysis>
  <isSupportCityManagement><!--optional, xs:boolean, whether it supports
intelligent city management. Refer to URI: /ISAPI/Intelligent/channels/<ID>/
cityManagement/capabilities?format=json--></isSupportCityManagement>
  <mixedTargetDetectionWithoutAttribute><!--optional,  xs: boolean--></
mixedTargetDetectionWithoutAttribute>
  <isSupportUploadFacePictureByForm><!--optional,  xs:boolean, whether it
supports uploading face pictures by form--></isSupportUploadFacePictureByForm>
  <isSupportUploadFacePictureByUrl><!--optional,  xs: boolean, whether it
supports uploading face pictures by URL--></isSupportUploadFacePictureByUrl>
  <isSupportUploadHumanPictureByForm><!--optional,  xs:boolean, whether it
```



```
supports uploading human pictures in form--></isSupportUploadHumanPictureByForm>
  <isSupportMisinfoFilterStatisticalMode><!--optioal, xs:boolean, whether it
supports statistics of filtering face comparison--></
isSupportMisinfoFilterStatisticalMode>
  <isSupportFaceScore><!--optional, xs:boolean, whether it supports face
grading configuration (camera)--></isSupportFaceScore>
  <HumanRecognitionModeSearchCap>
    <searchTargetsNumMax><!--optional, xs:integer, maximum number of sample
pictures that can be imported for searching by picture--></searchTargetsNumMax>
    <HumanMode>
      <searchCond opt="age_group,gender,jacet_color,glass,bag,ride,unlimit"/
><!--setting multiple search conditions is supported-->
      <similarity min="0.0" max="100.0"/><!--optional, xs:float, similarity,
range: [0.0,100.0]-->
    </HumanMode>
    <HumanInfo><!--optional, person attribute information, it is required when
searching for picture by attributes-->
      <age_group
opt="child,infant,kid,middle,middleAged,old,prime,teenager,unknown,young">
        <!--optional, xs:string, age group:
"child,infant,kid,middle,middleAged,old,prime,teenager,unknown,young"-->
      </age_group>
      <gender opt="female,male,unknown"><!--optional, xs:string, gender:
"female,male,unknown"--></gender>
      <jacet_color
opt="black,black,blue,brown,cyan,deepBlue,deepGray,gray,green,mixture,orange,pin
k,purple,red,silver,unknown,white,yellow">
        <!--optional, xs:string, tops color-->
      </jacet_color>
      <glass opt="yes,no,unknown,sunglasses"><!--optional, xs:string, whether
the person is wearing glasses--></glass>
      <bag opt="yes,no,unknown"><!--optional, xs:string, whether the person is
carrying a bag--></bag>
      <ride opt="yes,no,unknown"><!--optional, xs:string, whether the person is
riding--></ride>
    </HumanInfo>
    <positive opt="true,false"/><!--optional, xs:boolean, whether it is false
human body recognition alarm: "true"-yes, "false"-no-->
    <eventType opt="unlimit,humanRecognition">
      <!--optional, xs:string, event type: "unlimit"-no limit,
"humanRecognition"-human body detection alarm-->
    </eventType>
    <isSupportMultiChannelSearch>
      <!--optional, xs:boolean, whether it supports multi-channel search-->
    </isSupportMultiChannelSearch>
    <isSupportTotalSearchResult>
      <!--optional, xs:boolean, whether it supports limiting number of results
that can be obtained after a single search-->
    </isSupportTotalSearchResult>
  </HumanRecognitionModeSearchCap>
  <VehicleRecognitionModeSearchCap>
    <searchTargetsNumMax><!--optional, xs:integer, maximum number of sample
```

```
pictures that can be imported for searching by picture--></searchTargetsNumMax>
  <eventType opt="unlimit,vehicleBlackList,vehicleWhiteList"/><!--xs:string,
event type: "unlimit"-no limit, "vehicleBlackList"-vehicle blacklist,
"vehicleWhiteList"-vehicle allowlist-->
  <VehicleMode>
    <searchCond
opt="licensePlate,vehicleLogo,vehicleSubLogoRecog,vehicleType,vehicleColor,unlim
it"/>
    <similarity min="0.0" max="100.0"/><!--optional, xs:float, similarity,
range: [0.0,100.0]-->
  </VehicleMode>
  <isSupportMultiChannelSearch>
    <!--optional, xs:boolean, whether it supports multi-channel search-->
  </isSupportMultiChannelSearch>
  <isSupportTotalSearchResult>
    <!--optional, xs:boolean, whether it supports limiting number of results
that can be obtained after a single search-->
  </isSupportTotalSearchResult>
  </VehicleRecognitionModeSearchCap>
  <FaceContrastAnalyzeCap><!--capability of manual comparison and analysis of
face pictures-->
  <isSupportTrigger opt="false,true"><!--optional, xs:boolean--></
isSupportTrigger>
  <SearchCondition><!--optional-->
    <startTime><!--required, xs:time, ISO8601 time--></startTime>
    <endTime><!--required, xs:time, ISO8601 time--></endTime>
  </SearchCondition>
  </FaceContrastAnalyzeCap>
  <FramesPeopleCountingCap><!--capability of people counting statistics in a
single frame-->
  <Statistics>
    <SearchCondition>
      <startTime><!--required, xs:time,ISO8601 time--></startTime>
      <endTime><!--required, xs:time,ISO8601 time--></endTime>
    </SearchCondition>
  </Statistics>
  <MaskRegion><!--optional, shielded region-->
    <maxRegionNum><!--optional, xs:integer, number of regions--></
maxRegionNum>
    <Region>
      <vertexNum min="3" max="10"><!--optional, xs:integer, number of region
vertexes--></vertexNum>
    </Region>
  </MaskRegion>
</FramesPeopleCountingCap>
  <FaceContrastPersonInfoExtend><!--configuration capability of face comparison
tag-->
  <personInfoCap>
    <maxPersonInfo min="0" max="4">
      <!--optional, xs:integer, maximum number of person tags-->
    </maxPersonInfo>
  </personInfoCap>
```

```
<personInfoFDlibCap>
  <maxPersonInfo min="0" max="4">
    <!--optional, xs:integer, maximum number of person tags-->
  </maxPersonInfo>
</personInfoFDlibCap>
</FaceContrastPersonInfoExtend>
<isSupportSafetyHelmetDetection>
  <!-- opt, xs:boolean, whether it supports hard hat detection-->
</isSupportSafetyHelmetDetection>
<isSupportDistanceRecognition><!--optional, xs:boolean, whether supports
distance measurement--></isSupportDistanceRecognition>
<isSupportATM><!--optional, xs:boolean, ATM intelligent configuraiton,
corresponds to URI: /ISAPI/Intelligent/ATM/capabilities?format=json--></
isSupportATM>
  <isSupportFaceTemperature><!--optional, xs:boolean, whether the device
supports face thermography, corresponds to URI: /ISAPI/Intelligent/
faceTemperature/capabilities?format=json--></isSupportFaceTemperature>
  <isSupportFireEscapeDetectionSearch>
    <!--optional, xs:boolean, whether the device supports searching for fire
engine access detection events of multiple channels (related URI: URI/ISAPI/
Intelligent/fireEscapeDetection/search/capabilities?format=json)-->
  </isSupportFireEscapeDetectionSearch>
  <isSupportTakingElevatorDetectionSearch>
    <!--optional, xs:boolean, whether the device supports searching for
elevator detection events of multiple channels (related URI: /ISAPI/Intelligent/
takingElevatorDetection/search/capabilities?format=json)-->
  </isSupportTakingElevatorDetectionSearch>
  <isSupportCityManagementSearch>
    <!--optional, xs:boolean, whether the device supports searching for city
management events of multiple channels (related URI: /ISAPI/Intelligent/
cityManagement/search/capabilities?format=json)-->
  </isSupportCityManagementSearch>
  <isSupportSafetyHelmetDetectionSearch>
    <!--optional, xs:boolean, whether the device supports searching for hard
hat detection events of multiple channels (related URI: /ISAPI/Intelligent/
safetyHelmetDetection/search/capabilities?format=json)-->
  </isSupportSafetyHelmetDetectionSearch>
  <isSupportFramesPeopleCountingSearch>
    <!--optional, boolean, ro, whether it supports searching for the regional
people counting event (related URI: /ISAPI/Intelligent/framesPeopleCounting/
search/capabilities?format=json)-->true
  </isSupportFramesPeopleCountingSearch>
  <isSupportChannelPollingSchedules>
    <!--optional, xs:boolean, whether the device supports channel polling
configuration: "true,false"-->
  </isSupportChannelPollingSchedules>
  <AnalysisImageFaceCap>
    <!--optional, face picture analysis type (related URI:/ISAPI/Intelligent/
analysisImage/face)-->
    <modeling opt="false"><!--optional,xs:boolean, whether the device
supports modeling: "true,false"--></modeling>
```

```
</AnalysisImageFaceCap>  
</IntelliCap>
```

5.21 XML_ResponseStatus

XML message about response status

```
<?xml version="1.0" encoding="utf-8"?>  
<ResponseStatus version="2.0" xmlns="http://www.std-cgi.org/ver20/XMLSchema">  
  <requestURL>  
    <!--required, read-only, xs:string, request URL-->  
  </requestURL>  
  <statusCode>  
    <!--required, read-only, xs:integer, status code: 0,1-OK, 2-Device Busy, 3-  
Device Error, 4-Invalid Operation, 5-Invalid XML Format, 6-Invalid XML Content,  
7-Reboot Required, 9-Additional Error-->  
  </statusCode>  
  <statusString>  
    <!--required, read-only, xs:string, status description: OK, Device Busy,  
Device Error, Invalid Operation, Invalid XML Format, Invalid XML Content,  
Reboot, Additional Error-->  
  </statusString>  
  <subStatusCode>  
    <!--required, read-only, xs:string, describe the error reason in detail-->  
  </subStatusCode>  
  <MErrCode>  
    <!--optional, xs:string, error code categorized by functional modules,  
e.g., 0x12345678-->  
  </MErrCode>  
  <MErrDevSelfEx>  
    <!--optional, xs:string, extension field of MErrCode. It is used to define  
the custom error code, which is categorized by functional modules-->  
  </MErrDevSelfEx>  
</ResponseStatus>
```

Appendix A. Data Structure

A.1 DATE_TIME

Date and Time Structure

| Member | Data Type | Description |
|--------------------|-----------|---|
| year | short | Year. |
| month | short | Month. |
| dayOfWeek | short | Days of the week: 0-Sunday, 1-Monday, 2-Tuesday, 3-Wednesday, 4-Thursday, 5-Friday, 6-Saturday. |
| day | short | Day. |
| hour | short | Hour. |
| minute | short | Minute. |
| second | short | Second. |
| milliSecond | short | Millisecond. |

A.2 NET_DVR_DEVICEINFO_V30

Device parameter structure (V30).

Device Parameter Structure (V30)

| Member | Data Type | Description |
|-------------------|-----------|--------------------------------|
| sSerialNumber | BYTE | Device serial No. |
| byAlarmInPortNum | BYTE | Number of analog alarm inputs |
| byAlarmOutPortNum | BYTE | Number of analog alarm outputs |
| byDiskNum | BYTE | Number of HDDs |
| byDVRType | BYTE | Device type |
| byChanNum | BYTE | Number of analog channels |

| Member | Data Type | Description |
|----------------|-----------|---|
| byStartChan | BYTE | Start No. of analog channel, which starts from 1. |
| byAudioChanNum | BYTE | Number of two-way audio channels |
| byIPChanNum | BYTE | Number of digital channels, low 8-bit. |
| byZeroChanNum | BYTE | Number of channel-zero |
| byMainProto | BYTE | Transmission protocol type of main stream: 0-private protocol (default), 1-RTSP, 2-private protocol+RTSP |
| bySubProto | BYTE | Transmission protocol type of sub-stream: 0-private protocol (default), 1-RTSP, 2-private protocol+RTSP |
| bySupport | BYTE | <p>Capabilities, if the result of bitwise operation is 0, it refers that the capability is not supported, if the result is 1, it indicates that the capability is supported.</p> <ul style="list-style-type: none"> • bySupport&0x1: whether supports VCA search. • bySupport&0x2: whether supports backup. • bySupport&0x4: whether supports getting encoding parameters. • bySupport&0x8: whether supports dual-NIC. • bySupport&0x10: whether supports remote SADP. • bySupport&0x20: whether supports RAID card. • bySupport&0x40: whether supports searching in IPSAN directory. • bySupport&0x80: whether supports RTP over RTSP. |
| bySupport1 | BYTE | <p>Extended capabilities, if the result of bitwise operation is 0, it refers that the capability is not supported, if the result is 1, it indicates that the capability is supported.</p> <ul style="list-style-type: none"> • bySupport1&0x1: whether supports SNMP with version 30. • bySupport1&0x2: whether supports playback and downloading video files. |

| Member | Data Type | Description |
|------------|-----------|---|
| | | <ul style="list-style-type: none"> bySupport1&0x4: whether supports setting the arming priority. bySupport1&0x8: whether supports extending the arming time period. bySupport1&0x10: whether supports multiple HDDs (more than 33). bySupport1&0x20: whether supports RTP over RTSP. bySupport1&0x80: whether supports license plate recognition alarm. |
| bySupport2 | BYTE | <p>Extended capabilities, if the result of bitwise operation is 0, it refers that the capability is not supported, if the result is 1, it indicates that the capability is supported.</p> <ul style="list-style-type: none"> bySupport2&0x1: whether supports getting stream via URL. bySupport2&0x2: whether supports FTP with version 40. bySupport2&0x4: whether supports ANR. bySupport2&0x20: whether supports getting device status. bySupport2&0x40: whether supports encrypting stream. |
| wDevType | WORD | Device model |
| bySupport3 | BYTE | <p>Extended capabilities, if the result of bitwise operation is 0, it refers that the capability is not supported, while, if the result is 1, it indicates that the capability is supported.</p> <ul style="list-style-type: none"> bySupport3&0x1: whether supports multi-stream. bySupport3&0x4: whether supports configuring by group (e.g., image, alarm input, alarm output, user, device status, JPEG picture capture, continuous and scheduled capture, .HDD group management, and so on). bySupport3&0x20: whether supports getting stream via DDNS. |

| Member | Data Type | Description |
|--------------------|-----------|---|
| byMultiStreamProto | BYTE | Whether supports multi-stream, if the result of bitwise operation is 0, it refers to not support, if the result is 1, it refers to support. <ul style="list-style-type: none"> byMultiStreamProto&0x1: whether supports third-stream. byMultiStreamProto&0x2: whether supports fourth-stream. byMultiStreamProto&0x40: whether supports main stream. byMultiStreamProto&0x80: whether supports sub-stream. |
| byStartDChan | BYTE | Start No. of digital channel, 0-no digital channel (e.g., DVR, network camera). |
| byStartDTalkChan | BYTE | Start No. of two-way audio channel, 0-no two-way audio channel. |
| byHighDChanNum | BYTE | Number of digital channels, high 8-bit. |
| bySupport4 | BYTE | Extended capabilities, if the result of bitwise operation is 0, it refers that the capability is not supported, if the result is 1, it indicates that the capability is supported. <ul style="list-style-type: none"> bySupport4&0x01: whether all stream types support RTSP and private protocol. bySupport4&0x02: whether the device supports transmitting form format data via API (NET_DVR_STDXMLConfig). bySupport4&0x10: whether supports loading network disk by domain name. |
| byLanguageType | BYTE | Supported language types, if the result of bitwise operation is 0, it refers to not support, if the result is 1, it refers to support. <ul style="list-style-type: none"> byLanguageType==0: this field is not supported by device. byLanguageType&0x1: whether supports Chinese. byLanguageType&0x2: whether supports English. |
| byVoiceInChanNum | BYTE | Number of audio input channels |

| Member | Data Type | Description |
|----------------------|---------------|--|
| byStartVoiceInChanNo | BYTE | Start No. of audio input channel, 0-invalid. |
| byRes3 | Array of BYTE | Reserved, set to 0. |
| byMirrorChanNum | BYTE | Number of mirror channels |
| wStartMirrorChanNo | WORD | Start No. of mirror channel |
| byRes2 | Array of BYTE | Reserved, set to 0. |

Remarks

- The maximum number of digital channels equal to $\text{byIPChanNum} + \text{byHighDChanNum} * 256$.
- For login via text protocol, the following parameters are not supported: **byMainProto**, **bySubProto**, **bySupport**, **bySupport1**, **bySupport2**, **bySupport3**, **bySupport4**, **bySupport5**, **bySupport6**, **bySupport7**, **byMultiStreamProto**, **byStartDTalkChan**, **byVoiceInChanNum**, **byStartVoiceInChanNo**, **byMirrorChanNum**, and **wStartMirrorChanNo**.

See Also

NET_DVR_DEVICEINFO_V40

A.3 NET_DVR_DEVICEINFO_V40

Device Parameter Structure (V40)

| Member | Data Type | Description |
|------------------|--------------------------------------|--|
| struDeviceV30 | <u>NET_DVR_DEVICEINFO_V30</u> | Device parameters |
| bySupportLock | BYTE | Whether supports locking function: 1-support. |
| byRetryLoginTime | BYTE | Remaining login attempts, it is valid when the user name or password is incorrect and the bySupportLock is 1. |
| byPasswordLevel | BYTE | Password strength: 0-invalid, 1-default password, 2-valid password, 3-risky password. For default password or risky password, the users are reminded to change password. |
| byProxyType | BYTE | Proxy type: 0-no proxy, 1-standard proxy, 2-EHome proxy. |

| Member | Data Type | Description |
|--------------------|-----------|--|
| dwSurplusLockTime | DWORD | Remaining locking time, unit: second. It is valid only when bySupportLock is 1. During the locking time, if the user try to log in to again, the remaining locking time will resume to 30 minutes. |
| byCharEncodeType | BYTE | Character encodings. 0-no decoding information, 1-GB2312 (Simplified Chinese), 2-GBK, 3-BIG5 (Traditional Chinese), 4-Shift_JIS (Japanese), 5-EUC-KR (Korean), 6-UTF-8, 7-ISO8859-1, 8-ISO8859-2, 9-ISO8859-3, ..., 21-ISO8859-15 (Western European) |
| bySupportDev5 | BYTE | Whether to support getting the parameters of devices that support HCNetsdk version 5.0 or above, the size of device name and type name are extended to 64 bytes. |
| bySupport | BYTE | Whether it supports uploading changes, it depends on the result of bitwise AND (&) operation: 0-not support, 1-support. The result of bySupport&0x1 indicates that this member is reserved; the result of bySupport&0x2 indicates that whether it supports uploading changes: 0-not support, 1-support. This member is the capability set extension. |
| byLoginMode | BYTE | Login mode: 0-login via private protocol, 1-login via text protocol. For private protocol, the default login port number is 8000, and for text protocol, the default login port number is 80 or 443. |
| dwOEMCode | DWORD | OEM code. |
| iResidualValidity | int | Remaining valid days of the user's password, unit: day. If the negative number is returned, it indicates that the password being used has expired. For example, if -3 is returned, it indicates that the password being used has expired for three days. |
| byResidualValidity | BYTE | Whether the member iResidualValidity is valid: 0-invalid, 1-valid. |

| Member | Data Type | Description |
|------------------------|---------------|---|
| bySingleStartDTalkChan | BYTE | Start channel No. for connecting independent audio tracks to the device. The value 0 is reserved and invalid. The channel No. of audio tracks cannot start from 0. |
| bySingleDTalkChanNums | BYTE | Total number of channels of the device connected with independent tracks, 0-not support. |
| byPassWordResetLevel | BYTE | Whether to prompt the non-admin user to change the password: 0 (invalid), 1 (If the administrator creates a non-admin user account with an initial password, the non-admin user will be prompted "Please change the initial password" each time he/she logs in to the device until he/she changes the initial password), 2(If the non-admin user's password has been changed by the administrator, the non-admin user will be prompted "Please set a new password" each time he/she logs in to the device until he/she changes the password). |
| bySupportStreamEncrypt | BYTE | Whether it supports stream encryption, it depends on the result of bitwise AND (&) operation: 0-no, 1-yes. The result of bySupportStreamEncrypt&0x1 indicates whether to support RTP/TLS streaming, the result of bySupportStreamEncrypt&0x2 indicates whether to support SRTP/UDP streaming, and the result of bySupportStreamEncrypt&0x4 indicates whether to support SRTP/MULTICAST streaming. |
| byRes2 | Array of BYTE | Reserved, set to 0. |

Remarks

- Four character types are allowed in the password, including digits, lowercase letters, uppercase letters and symbols. The maximum password length is 16 bits, and there are four password strength levels, see details below:

- Level 0 (Risky Password): The password length is less than 8 bits, or only contains one kind of the character types. Or the password is the same with the user name, or is the mirror writing of the user name.
- Level 1 (Weak Password): The password length is more than or equal to 8 bits, and contains two kinds of the character types. Meanwhile, the combination should be (digits + lowercase letters) or (digits + uppercase letters).
- Level 2 (Medium Password): The password length is more than or equal to 8 bits, and contains two kinds of the character types. Meanwhile, the combination cannot be (digits + lowercase letters) and (digits + uppercase letters).
- Level 3 (Strong Password): The password length is more than or equal to 8 bits, and at least contains three kinds of the character types.
- For login via text protocol, the following parameters are not supported: **bySupportLock**, **byRetryLoginTime**, **byPasswordLevel**, **byProxyType**, **dwSurplusLockTime**, **byCharEncodeType**, and **bySupportDev5**.

A.4 NET_DVR_INIT_CFG_ABILITY

Initialization Capability Structure

| Member | Data Type | Description |
|----------------------|------------------|--|
| enumMaxLoginUsersNum | INIT_CFG_MAX_NUM | <p>Maximum number of users can log in, see details below:</p> <pre>enum _INIT_CFG_MAX_NUM_{ INIT_CFG_NUM_2048 = 2048, INIT_CFG_NUM_5120 = 5120, INIT_CFG_NUM_10240 = 10240, INIT_CFG_NUM_15360 = 15360, INIT_CFG_NUM_20480 = 20480 }INIT_CFG_MAX_NUM</pre> |
| enumMaxAlarmNum | INIT_CFG_MAX_NUM | <p>Maximum number of alarm channels, see details below:</p> <pre>enum _INIT_CFG_MAX_NUM_{ INIT_CFG_NUM_2048 = 2048, INIT_CFG_NUM_5120 = 5120, INIT_CFG_NUM_10240 = 10240, INIT_CFG_NUM_15360 = 15360, INIT_CFG_NUM_20480 = 20480 }INIT_CFG_MAX_NUM</pre> |
| byRes | Array of BYTE | Reserved, set to 0. |

Remarks

By default, up to 2048 channels are supported. More channels require higher computer performance and network bandwidth.

See Also

NET_DVR_SetSDKInitCfg

A.5 NET_DVR_LOCAL_SDK_PATH

Path Information Structure for Loading Component Libraries

| Member | Data Type | Description |
|--------|---------------|--------------------------------|
| sPath | Array of char | Component libraries' addresses |
| byRes | Array of BYTE | Reserved. |

Remarks

If the path of HCNetSDKCom folder and HCNetSDK libraries are same, but the path of executable programs are different, you can call **NET_DVR_SetSDKInitCfg** to specify the path of HCNetSDKCom folder to make sure the component libraries are loaded normally.

A.6 NET_DVR_USER_LOGIN_INFO

Structure About Login Parameters

| Member | Data Type | Description |
|----------------|------------------------------------|--|
| sDeviceAddress | char | Device IP address, or domain name. |
| byUseTransport | BYTE | Enable capability transmission or not: 0-no (default), 1-yes. |
| wPort | WORD | Device port number, e.g., 8000 (when login by private protocol), 80 (when login by text protocol). |
| sUserName | char | User name for logging in to device. |
| sPassword | char | Login password. |
| cbLoginResult | <u>fLoginResultCallback</u> | Callback function used to return login status, it is valid only when bUseAsynLogin is "1". |

| Member | Data Type | Description |
|---------------|-----------|---|
| pUser | void* | User data. |
| bUseAsynLogin | BOOL | Whether to enable asynchronous login: 0-no, 1-yes. |
| byProxyType | BYTE | Proxy server type: 0-no proxy, 1-standard proxy, 2-EHome proxy. |
| byUseUTCTime | BYTE | 0-not convert (default), 1-input or output UTC time, 2-input or output local time. |
| byLoginMode | BYTE | Login mode: 0-login by private protocol, 1-login by text protocol, 2-self-adaptive (it is available when the protocol type supported by device is unknown, and this mode does not support asynchronous login). |
| byHttps | BYTE | Whether to enable TLS for login (by private protocol or by text protocol): 0-no, 1-yes, 2-self-adaptive (which is usually used when the protocol type supported by device is unknown. Both HTTP and HTTPS requests will be sent). |
| iProxyID | LONG | Proxy server No. |
| byVerifyMode | BYTE | Whether to enable verification mode: 0-no, 1-bidirectional verification (currently not available), 2-unidirectional verification (it is valid when byLoginMode is 0 and byHttps is 1); when byVerifyMode is 0, CA certificate is not required, when byVerifyMode is 2, you should call NET_DVR_SetSDKLocalCfg to load CA certificate, and the enumeration value is "NET_SDK_LOCAL_CFG_CERTIFICATION". |
| byRes3 | BYTE[] | Reserved, the maximum length is 119 bytes. |

A.7 NET_SDK_CALLBACK_STATUS_NORMAL

Enumeration About Persistent Connection Status

| Enumeration Type | Marco Definition Value | Description |
|------------------------------------|------------------------|--|
| NET_SDK_CALLBACK_STATUS_SUCCESS | 1000 | Succeeded. |
| NET_SDK_CALLBACK_STATUS_PROCESSING | 1001 | Connecting. The lpBuffer is 4-byte status. |
| NET_SDK_CALLBACK_STATUS_FAILED | 1002 | Failed. The lpBuffer is the value of 4-byte status and 4-byte error code. |

A.8 NET_VCA_RECT

Structure About Rectangle Region Coordinate Parameters

| Member | Data Type | Description |
|---------|-----------|---|
| fX | float | X-coordinate of frame's upper-left corner, it ranges from 0.000 to 1. |
| fY | float | Y-coordinate of frame' upper-left corner, it ranges from 0.000 to 1. |
| fWidth | float | Frame width, it ranges from 0.000 to 1. |
| fHeight | float | Frame height, it ranges from 0.000 to 1. |

A.9 NET_DVR_CRUISECHAN_INFO

Structure about The Information of Channel That Called Patrol

| Member | Data Type | Description |
|--------------------|-----------|---------------------------------|
| dwEnableCruiseChan | DWORD | Channel that called patrol. |
| dwCruiseNo | DWORD | Patrol No., 0xffffffff-invalid. |

A.10 NET_DVR_EVENT_SCHEDULE

Arming Schedule Parameter Structure

| Member | Data Type | Description |
|----------------------|---|--|
| dwSize | DWORD | Structure size |
| struAlarmTime | Array of <u>NET_DVR_SCHEDULETIME</u> | Arming schedule, 7 days per week, 8 time periods per day |
| struHolidayAlarmTime | Array of <u>NET_DVR_SCHEDULETIME</u> | Holiday arming schedule, see details in the structure . |
| byRes | BYTE | Reserved. |

A.11 NET_DVR_EVENT_TRIGGER

Structure About Event Linkage Configuration

| Member | Data Type | Description |
|---------------------|---|---|
| dwSize | DWORD | Structure size. |
| struHandleException | Array of <u>NET_DVR_HANDLEEXCEPTION_V41</u> | Exception handling mode |
| dwRelRecordChan | Array of DWORD | Actually triggered video channel, represented by value, read starts from 0, and it is invalid after the value of 0xffffffff being read. |
| struPresetChanInfo | Array of <u>NET_DVR_PRESETCHANNEL_INFO</u> | Information of channel that called preset |
| struCruiseChanInfo | Array of <u>NET_DVR_CRUISECHANNEL_INFO</u> | Information of channel that called patrol |
| struPtzTrackInfo | Array of <u>NET_DVR_PTZTRACKCHANNEL_INFO</u> | Information of channel that called pattern |

| Member | Data Type | Description |
|-------------|---------------|--|
| byDirection | Array of BYTE | Triggering direction: 0-reserved, 1-all, 2-forward, 3-backward |
| szFDID | Char | Face picture library ID |
| byRes2 | Array of BYTE | Reserved |

A.12 NET_DVR_HANDLEEXCEPTION_V41

Exception Information Structure

| Member | Data Type | Description |
|-------------------------|-----------|---|
| dwHandleType | DWORD | <p>Handling types, see details below:</p> <ul style="list-style-type: none"> • 0x00: no response • 0x01: display alarm on monitor screen • 0x02: audio warning • 0x04: upload to center • 0x08: trigger alarm output • 0x10: send picture with JPEG format by e-mail • 0x20: trigger wireless sound and light alarm • 0x40: trigger e-map (supported by PCNVR only) • 0x200: capture picture and upload to FTP • 0x400: focus mode linkage (for defocus detection) • 0x800: PTZ linkage (speed dome tracks the target) • 0x1000: capture picture and upload to cloud storage. • 0x10000: message alarm <p>E.g., if dwHandleType is 0x01 0x04, it indicates that the alarm information will be displayed on monitor screen and uploaded to alarm center when the alarm is triggered.</p> |
| dwMaxAlarmOutChannelNum | DWORD | Manixmum number of alarm outputs (read only) supported by the device. |

| Member | Data Type | Description |
|---------------|----------------|--|
| dwRelAlarmOut | Array of DWORD | Alarm output No. triggered by alarm, which starts from 0, 0xffffffff-invalid. E.g. byRelAlarmOut[i]==3 indicates that the alarm output No.4 is triggered. |
| byRes | Array of BYTE | Reserved, set to 0. |

A.13 NET_DVR_MIME_UNIT

Input Content Details Structure of Message Transmission API (NET_DVR_STDXMLConfig)

| Member | Data Type | Description |
|---------------|---------------|---|
| szContentType | Array of char | Content type (corresponds to Content-Type field in the message), e.g., text/json. text/xml, and so on. The content format must be supported by HTTP. |
| szName | Array of char | Content name (corresponds to name field in the message), e.g., name="upload". |
| szFilename | Array of char | Content file name (corresponds to filename field in the message), e.g., filename="C:\Users\test\Desktop\11.txt". |
| dwContentLen | DWORD | Content size |
| pContent | char* | Data point |
| bySelfRead | BYTE | 0-External file, 1-Internal data, whose address is specified by szFilename . |
| byRes | Array of BYTE | Reserved. Set to 0. Maximum: 15 bytes. |

See Also

NET_DVR_XML_CONFIG_INPUT

A.14 NET_DVR_PRESETCHAN_INFO

Preset Information Structure

| Member | Data Type | Description |
|--------------------|-----------|--|
| dwEnablePresetChan | DWORD | Channel that called preset. |
| dwPresetPointNo | DWORD | Called preset No., 0xffffffff-not call preset. |

A.15 NET_DVR_PTZTRACKCHAN_INFO

Pattern Information Structure

| Member | Data Type | Description |
|----------------------|-----------|---|
| dwEnablePtzTrackChan | DWORD | Channel that called the pattern. |
| dwPtzTrackNo | DWORD | Called pattern No., 0xffffffff-invalid. |

A.16 NET_DVR_SCHEDTIME

Structure About Start and End Time Parameters

| Member | Data Type | Description |
|-------------|-----------|---------------------|
| byStartHour | BYTE | Start time: hour. |
| byStartMin | BYTE | Start time: minute. |
| byStopHour | BYTE | End time: hour. |
| byStopMin | BYTE | End time: minute. |

A.17 NET_DVR_STD_ABILITY

Input and Output Parameter Structure for Getting Capabilities

| Member | Data Type | Description |
|-----------------------|--------------|---|
| IpCondBuffer | LPVOID | Condition parameters (ASCII character format), e.g., the channel No., it can be set to "null". |
| dwCondSize | DWORD | Buffer size of condition parameters. |
| IpOutBuffer | LPVOID | Output parameters buffer (the parameter is returned in the message with XML format), it cannot be set to "null". |
| dwOutSize | DWORD | Output buffer size. |
| IpStatusBuffer | LPVOID | Get the returned status parameters (<i><u>XML ResponseStatus</u></i>) when getting capabilities failed. It can be set to null. |
| dwStatusSize | DWORD | Status buffer size. |
| dwRetSize | DWORD | Obtained data size (if the capability is obtained, the value refers to the size of IpOutBuffer ; if getting failed, the value refers to the size of IpStatusBuffer). |
| byRes | Array [BYTE] | Reserved. The maximum size is 32 bytes. |

Remarks

For different capability types (which depend on the parameter **dwAbilityType** in the API ***NET_DVR_GetSTDAbility***), the condition parameter **IpCondBuffer** and output parameter **IpOutBuffer** are different. For details, refer to the typical applications.

A.18 NET_DVR_STD_CONFIG

Structure About Configuring Input and Output Parameters

| Member | Data Type | Description |
|---------------------|-----------|---|
| IpCondBuffer | LPVOID | Condition parameters, e.g., channel No., it can be set to "NULL". |
| dwCondSize | DWORD | Size of buffer for storing condition parameters |
| IpInBuffer | LPVOID | Input parameters (a structure) |

| Member | Data Type | Description |
|-----------------------|--------------|---|
| dwInSize | DWORD | Size of buffer for storing input parameters |
| lpOutBuffer | LPVOID | Output parameters (a structure) |
| dwOutSize | DWORD | Size of buffer for storing output parameters |
| lpStatusBuffer | LPVOID | Returned status parameters in XML format, it can be set to NULL. |
| dwStatusSize | DWORD | Size of buffer for storing status parameters |
| lpXmlBuffer | LPVOID | Request or response message in XML format, it is valid when byDataType is 1. |
| dwXmlSize | DWORD | Size of memory pointed by lpXmlBuffer . |
| byDataType | BYTE | Input or output parameter type: 0-valid when the input or output parameters is a structure; 1-valid when the input or output parameters is a XML message. |
| byRes | Array [BYTE] | Reserved, set to 0. The maximum size is 32 bytes. |

A.19 NET_DVR_XML_CONFIG_INPUT

Input Parameter Structure of Message Transmission API (NET_DVR_STDXMLConfig)

| Member | Data Type | Description |
|------------------------|-----------|--|
| dwSize | DWORD | Structure size. |
| lpRequestUrl | void* | Request URL (command) for implement different functions, and it is in string format. |
| dwRequestUrlLen | DWORD | Request URL size. |
| lpInBuffer | void* | Buffer for storing input parameters (request messages), see the input content details structure in <i>NET_DVR_MIME_UNIT</i> . |
| dwInBufferSize | DWORD | Input buffer size. |
| dwRecvTimeOut | DWORD | Receiving timeout, unit: ms, 0-5000ms (default). |

| Member | Data Type | Description |
|-------------------------|---------------|---|
| byForceEncript | BYTE | Whether to enable force encryption (the messages will be encrypted by AES algorithm for transmission): 0-no, 1-yes. |
| byNumOfMultiPart | BYTE | Number of message segments: 0-invalid; other values-number of message segments, which is transmitted by the parameter lpInBuffer in the structure NET_DVR_MIME_UNIT . |
| byRes | Array of BYTE | Reserved, set to 0. |

Related API

NET_DVR_STDXMLConfig

A.20 NET_DVR_XML_CONFIG_OUTPUT

Output Parameter Structure of Message Transmission API (NET_DVR_STDXMLConfig)

| Member | Data Type | Description |
|-------------------|-------------|---|
| dwSize | DWORD | Structure size. |
| lpOutBuffer | void* | Buffer for storing output parameters (response messages), which is allocated when passing through URL by GET method. |
| dwOutBufferSize | DWORD | Output buffer size. |
| dwReturnedXMLSize | DWORD | Actual size of response message. |
| lpStatusBuffer | void* | Response status (ResponseStatus message). This parameter will not be assigned if performing GET operation succeeded, and you can also set it to "NULL" if not required. |
| dwStatusSize | DWORD | Size of response status buffer. |
| lpDataBuffer | HPR_VOIDPTR | Buffer for transmitted data. This parameter is valid when the value of byNumOfMultiPart is larger than 0. |

| Member | Data Type | Description |
|------------------|-----------|---|
| byNumOfMultiPart | HPR_UINT8 | Number of parts that the message is divided into. |
| byRes [23] | BYTE | Reserved, set to 0. |

Related API

[NET_DVR_STDXMLConfig](#)

A.21 NET_ALARM_CVR_SUBINFO_UNION

Union about CVR Alarm Information

| Member | Data Type | Description |
|---------------------|---|---|
| byLen | BYTE[] | Union size, the maximum array length is 492 bytes. |
| struRecordLost | <u>NET_ALARM_RECORD_FILE_LOSS</u> | Video loss alarm information, the value of dwAlarmType in <u>NET_DVR_ALARMINFO_DEV_V40</u> is 8. |
| struStreamException | <u>NET_ALARM_STREAM_EXCEPTION</u> | Streaming exception alarm information, the value of dwAlarmType in <u>NET_DVR_ALARMINFO_DEV_V40</u> is 9. |
| struResourceUsage | <u>NET_ALARM_RESOURCE_USAGE</u> | Resource usage alarm information, the value of dwAlarmType in <u>NET_DVR_ALARMINFO_DEV_V40</u> is 10. |
| struRecordException | <u>NET_ALARM_RECORD_EXCEPTION</u> | Recording exception alarm information, the value of dwAlarmType in <u>NET_DVR_ALARMINFO_DEV_V40</u> is 12. |

A.22 NET_ALARM_RECORD_EXCEPTION

Structure about Recording Exception Alarm Information

| Member | Data Type | Description |
|--------------------|-----------|---|
| byReason | BYTE | Exception reason: 0-video volume full, 1-video volume exception, 2-no available video volume. |
| byRes1 | BYTE[] | Reserved, set to 0. The maximum array length is 3 bytes. |
| sVolumeName | BYTE[] | Video volume name, the maximum array length is "MAX_VOLUMENAME_LEN" (32 bytes). |
| dwVolumeID | DWORD | Video volume ID, or HDD No. |
| byRes | BYTE[] | Reserved, set to 0. The maximum array length is 452 bytes. |

A.23 NET_ALARM_RECORDFILE_LOSS**Structure about Video Loss Alarm Information**

| Member | Data Type | Description |
|--------------------------|------------------------------------|---|
| struInspectStart | <u>NET_DVR_TIME_EX</u> | Start time of video loss check. |
| struInspectEnd | <u>NET_DVR_TIME_EX</u> | End time of video loss check. |
| struIP | <u>NET_DVR_IPADDR_UNION</u> | IP address of video loss channel. |
| dwChanNo | DWORD | Channel No. |
| dwIDIndex | DWORD | Encoder ID. |
| sName | BYTE[] | Encoder name, the maximum array length is "STREAM_ID_LEN" (32 bytes). |
| struLossStartTime | <u>NET_DVR_TIME_EX</u> | Start time of video loss. |
| struLossEndTime | <u>NET_DVR_TIME_EX</u> | End time of video loss. |
| dwLostNum | DWORD | Number of lost video files, 0xffffffff-all video files are lost. |
| byRes | BYTE[] | Reserved, set to 0. The maximum array length is 240 bytes. |

A.24 NET_ALARM_RESOURCE_USAGE

Structure about Resource Usage Alarm Information

| Member | Data Type | Description |
|---------|-----------|---|
| byLevel | BYTE | Usage alarm level: 0-normal, 1-alarm level 1, 2-alarm level 2, 3-alarm level 3. |
| byRes | BYTE[] | Reserved, set to 0. The maximum array length is 491 bytes. |

A.25 NET_ALARM_STREAM_EXCEPTION

Structure about Video Exception Alarm Information

| Member | Data Type | Description |
|-----------------|-----------------------------|---|
| strulP | <u>NET_DVR_IPADDR_UNION</u> | IP address of video exception channel. |
| dwChanNo | DWORD | Channel No. |
| dwIDIndex | DWORD | Encoder ID. |
| sName | BYTE[] | Encoder name, the maximum array length is "STREAM_ID_LEN" (32 bytes). |
| byExceptionCase | BYTE | Exception reason: 0-data writing exception, 1-network exception. |
| byRes | BYTE[] | Reserved, set to 0. The maximum array length is 307 bytes. |

A.26 NET_DVR_ALARMER

Alarm Device Information Structure

| Member | Data Type | Description |
|-------------------|---------------|--|
| byUserIDValid | BYTE | Whether the user ID is valid: 0-no, 1-yes |
| bySerialValid | BYTE | Whether the serial No. is valid: 0-no, 1-yes |
| byVersionValid | BYTE | Whether the version No. is valid: 0-no, 1-yes |
| byDeviceNameValid | BYTE | Whether the device name is valid: 0-no, 1-yes |
| byMacAddrValid | BYTE | Whether the MAC address is valid: 0-no, 1-yes |
| byLinkPortValid | BYTE | Whether the login port No. is valid: 0-no, 1-yes |
| byDeviceIPValid | BYTE | Whether the device IP address is valid: 0-no, 1-yes |
| bySocketIPValid | BYTE | Whether the Socket IP address is valid: 0-no, 1-yes |
| lUserID | LONG | Value returned by <i>NET_DVR_Login_V40</i> , it is valid when arming. |
| sSerialNumber | Array of BYTE | Serial No. |
| dwDeviceVersion | DWORD | Version information |
| sDeviceName | Array of char | Device name |
| byMacAddr | Array of BYTE | MAC address |
| wLinkPort | WORD | Device communication port No. |
| sDeviceIP | Array of char | Device IP address |
| sSocketIP | Array of char | Socket IP address when actively uploading alarm. |
| byIpProtocol | BYTE | Network protocol: 0-IPv4, 1-IPv6 |
| byRes2 | Array of BYTE | Reserved, set to 0. |

A.27 NET_DVR_ALARMINFO_DEV

Device Alarm Information Structure

| Memeber | Data Type | Description |
|--------------------|---------------|--|
| dwAlarmType | DWORD | Alarm types: 0-alarm input alarm of encoder, 1-second private volume damaged, 2-NVR disconnected, 3-encoder exception, 4-system clock exception, 5-the remaining capacity of the recording volume is too low, 6-motion detection alarm of encoder or encoding channel, 7-video tampering alarm of encoder or encoding channel. |
| struTime | | Alarm time |
| byRes | Array of BYTE | Reserved, set to 0. |
| dwNumber | DWORD | Number of alarm triggered channels. |
| pNO | WORD* | Channel No. or disk No., which ranges from 0 to 65535. |

Remarks

For **pNO**: if **dwAlarmType** is 0, 3, 6, or 7, it may be channel No.; if **dwAlarmType** is 5, it may be disk No.

A.28 NET_DVR_ALARMINFO_DEV_V40

Structure about CVR Alarm Information

| Member | Data Type | Description |
|--------------------|-----------|--|
| dwAlarmType | DWORD | Alarm categories: 0-alarm input alarm of encoder, 1-second private volume damaged, 2-NVR disconnected, 3-encoder exception, 4-system clock exception, 5-the remaining capacity of the recording volume is too low, 6-motion detection alarm of encoder or encoding channel, 7-video tampering alarm of encoder or encoding channel, 8-video loss alarm, 9-real-time health monitoring alarm, 10-usage alarm, |

| Member | Data Type | Description |
|----------------------|---|---|
| | | 11-CVR exception recovered, 12-recording exception. |
| struTime | <u>NET_DVR_TIME</u> | Alarm time |
| uSubAlarmInfo | <u>NET_ALARM_CVR_SUBINFO_UNION</u> | CVR alarm information structure, and it is valid when the alarm type is 8, 9, 10, and 12. |
| byRes | Array of BYTE | Reserved, set to 0. The maximum size is 256 bytes. |
| dwNumber | DWORD | Number of alarm triggered channels. |
| pNO | WORD* | Channel No. or disk No., which ranges from 0 to 65535. |

Remarks

For **pNO**: if **dwAlarmType** is 0, 3, 6, or 7, it may be channel No.; if **dwAlarmType** is 5, it may be disk No.

A.29 NET_DVR_ALARMINFO_V30

Structure About Uploaded Alarm Information

| Member | Data Type | Description |
|----------------------------|---------------|---|
| dwAlarmType | DWORD | Alarm types: 0-alarm input alarm of encoder, 1-second private volume damaged, 2-NVR disconnected, 3-encoder exception, 4-system clock exception, 5-the remaining capacity of the recording volume is too low, 6-motion detection alarm of encoder or encoding channel, 7-video tampering alarm of encoder or encoding channel, 8-video loss alarm, 9-real-time health monitoring alarm, 10-usage alarm, 11-CVR exception recovered, 12-recording exception. |
| dwAlarmInputNumber | DWORD | Alarm input No., it is valid when alarm type is 0 or 23 |
| byAlarmOutputNumber | Array of BYTE | The triggered alarm output No. E.g. dwAlarmOutputNumber[0]==1 indicates that |

| Member | Data Type | Description |
|----------------------|---------------|--|
| | | alarm output No.1 is triggered; dwAlarmOutputNumber[1]==1 indicates that alarm output No.2 is triggered. |
| byAlarmRelateChannel | Array of BYTE | The triggered recording channel No.: 0-not triggered, 1-triggered. E.g. dwAlarmRelateChannel[0]==1 indicates that the channel No.1 is triggered to record. |
| byChannel | Array of BYTE | Alarm channel, it is valid when alarm type is 2, 3, 6, 9, 10 or 11. E.g. dwChannel[0]==1 indicates that the channel No. is in alarm. |
| byDiskNumber | Array of BYTE | Alarm HDD, it is valid when alarm type is 1, 4, or 5. E.g. dwDiskNumber [0]==1 indicates that the HDD No.1 is abnormal. |

Remarks

The time interval to upload the alarm of face picture library changed is 1 hour; for other alarm type, the alarm information is uploaded in real-time, and the time interval is 1s. Currently, editing the time interval is not supported.

A.30 NET_DVR_ALARMINFO_V40

Structure About Uploaded Alarm Information

| Member | Data Type | Description |
|----------------------|--|---|
| struAlarmFixedHeader | <u>NET_DVR_ALRAM_FIXED_HEADER</u> | Constant content in alarm information, see details in the structure . |
| pAlarmData | DWORD* | Variable content in alarm information |

Remarks

- The time interval to upload the alarm of face picture library changed is 1 hour; for other alarm type, the alarm information is uploaded in real-time, and the time interval is 1s. Currently, editing the time interval is not supported.
- The content of **pAlarmData** varies with the value of **dwAlarmType** in the structure **NET_DVR_ALRAM_FIXED_HEADER**, see details in the table below:

Table A-1 Relations Between pAlarmData and dwAlarmType

| dwAlarmType | Description | pAlarmData |
|------------------------------------|---|---|
| 0, 23 | Alarm input alarm, pulse alarm | dwTrigerAlarmOutNum*(DWORD) Alarm output No., +dwTrigerRecordChanNum*(DWORD) Channel No. |
| 2, 3, 6, 9, 10, 11, 13, 15, 16, 19 | Video loss, motion detection, video tampering alarm, video exception, recording exception, scene change, resolution mismatched, VCA detection, PoE power supply exception, audio loss | dwAlarmChanNum*(DWORD) channel No. |
| 1, 4, 5 | HDD full, HDD uninitialized, writing to HDD failed | dwAlarmHardDiskNum*(DWORD) HDD No. |
| 7, 8, 12, 17, 18, 24, 25, 26 | Standard mismatches, invalid login, array exception, education sharing system alarm, two-way audio request alarm, face library HDD exception, face library changed, picture changed in face picture library | None |

A.31 NET_DVR_ALARM_FIXED_HEADER

Structure About Constant Alarm Information

| Member | Data Type | Description |
|---------------|------------------|---|
| dwAlarmType | DWORD | Alarm information type: 0-alarm input alarm, 1-HDD full, 2-video loss, 3-motion detection, 4-HDD unformatted, 5-writing to HDD failed, 6-video tampering alarm, 7-standard mismatched, 8-invalid login, 9-video exception, 10-recording exception, 11-scene change, 12-RAID exception, 13-resolution mismatched, 15-VCA detection, 16- PoE power supply exception, 17-education sharing system alarm, 18-two-way audio request alarm, 23-pulse alarm, 24-face picture |

| Member | Data Type | Description |
|------------------|-----------------------------------|--|
| | | library HDD exception, 25-face picture library changed, 26-picture of face picture library changed, 27-POC exception, 28-camera FOV exception, 30-no SD card, 31-supply voltage exception, 32-PTZ locked |
| struAlarmTime | <u>NET DVR TIME EX</u> | Alarm time |
| uStruAlarm | Union (<u>Table 4-2</u>) | Alarm information union |
| pRes | DWORD* | Reserved. |
| byTimeDiffFlag | BYTE | Whether the time difference parameter is valid: 0-invalid, 1-valid. |
| cTimeDifferenceH | char | Time difference between time and UTC time, unit: hour, the value is between -12 and +14 ("+" indicates the east time zone), it is valid when byISO8601 is "1". |
| cTimeDifferenceM | char | Time difference between time and UTC time, unit: minute, the value is -30, +30, or +45 ("+" indicates the east time zone), it is valid when byISO8601 is "1". |
| byRes | Array of BYTE | Reserved, set to 0. The maximum size is 5 bytes. |

Table A-2 Union about Alarm Information Structures (uStruAlarm)

| Member | Data Type | Description |
|---------------------|------------------------------------|--|
| byUnionLen | Array of BYTE | Union size, which is 116 bytes. |
| struIOAlarm | Struct (<u>Table 4-3</u>) | Structure about alarm input parameters |
| struAlarmChannel | Struct (<u>Table 4-4</u>) | Structure about alarm channel parameters |
| struAlarmHardDisk | Struct (<u>Table 4-5</u>) | Structure about HDD alarm parameters |
| struRecordingHost | Struct (<u>Table 4-6</u>) | Structure about alarm parameters of education sharing system |
| struVoltageInstable | Struct (<u>Table 4-7</u>) | Structure about alarm parameters of supply voltage exception |
| struPTLocking | Struct (<u>Table 4-8</u>) | Structure about parameters of PTZ locked alarm |

Table A-3 Structure about Alarm Input Parameters (struIOAlarm)

| Member | Data Type | Description |
|-----------------------|-----------|--|
| dwAlarmInputNo | DWORD | Alarm input No. |
| dwTrigerAlarmOutNum | DWORD | The number of triggered alarm outputs. It is used for calculating the number of all triggered alarm outputs by pAlarmData in NET_DVR_ALARMINFO_V40 , each alarm output is represented by 4 bytes. |
| dwTrigerRecordChanNum | DWORD | The number of triggered recording channels. It is used for calculating the number of all triggered recording channels by pAlarmData of NET_DVR_ALARMINFO_V40 , each channel is represented by 4 bytes. |

Table A-4 Structure about Alarm Channel Parameters (struAlarmChannel)

| Member | Data Type | Description |
|----------------|---------------|--|
| dwAlarmChanNum | DWORD | The number of alarm channels. It is used for calculating the number of all alarm channels by pAlarmData of NET_DVR_ALARMINFO_V40 , each alarm channel is represented by 4 bytes. |
| dwPicLen | DWORD | Size of JPEG picture. |
| byPicURL | BYTE | Picture data format: 0-binary data, 1-URL. |
| byTarget | BYTE | Detection target type: 0-not supported, 1-person, 2-vehicle. |
| byRes1 | Array of BYTE | Reserved, the maximum size is 2 bytes. |
| pDataBuff | char* | Alarm picture data or URL. The pointer size is 8 bytes. |
| byRes3 | Array of BYTE | Reserved, the maximum size is 4 bytes. This member is only available for 64-bit Window operating system and 64-bit Linux operating system. |

Table A-5 Structure about HDD Alarm Parameters (struAlarmHardDisk)

| Member | Data Type | Description |
|--------------------|-----------|--|
| dwAlarmHardDiskNum | DWORD | The number of alarm HDD. It is used for calculating the number of all alarm HDDs by pAlarmData of <u>NET DVR ALARMINFO V40</u> , each alarm HDD is represented by 4 bytes. |

Table A-6 Structure about Alarm Parameters of Education Sharing System (struRecordingHost)

| Member | Data Type | Description |
|-------------------|-------------------------------|--|
| bySubAlarmType | BYTE | Alarm minor type: 1-one-touch post-record |
| byRes1 | Array of BYTE | Reserved, set to 0. The maximum size is 3 bytes. |
| struRecordEndTime | <u>NET DVR TIME EX</u> | Recording end time. |

Table A-7 Structure about Alarm Parameters of Supply Voltage Exception (struVoltageInstable)

| Member | Data Type | Description |
|--------------------|---------------|--|
| fVoltageValue | float | Supply voltage, unit: V, corrects to one decimal place. |
| byVoltageAlarmType | BYTE | Supply voltage exception type: 0-high supply voltage, 1-low supply voltage |
| byRes1 | Array of BYTE | Reserved, set to 0. The maximum size is 3 bytes. |

Table A-8 Structure about Parameters of PTZ Locked Alarm (struPTLocking)

| Member | Data Type | Description |
|--------------------|-----------|---|
| fTemperature | float | Sensor temperature, which is accurate to one decimal place. |
| dwCustomInfoLength | DWORD | Custom information length. |
| pCustomInfo | BYTE* | Custom information. |
| byType | BYTE | PTZ locked direction: 1-panning is locked, 2-tilting is locked. |
| byDeicingEnabled | BYTE | Whether to enable heat for PTZ: 0-no, 1-yes. |

Remarks

dwAlarmType==0, 23 corresponds to the structure **struIOAlarm**; **dwAlarmType**==2/3/6/9/10/11/13/15/16/28 corresponds to the structure **struAlarmChannel**; **dwAlarmType**==

1/4/5 corresponds to the structure `struAlarmHardDisk`; `dwAlarmType== 17` corresponds to the structure `struRecordingHost`; `dwAlarmType== 31` corresponds to the structure `struVoltageInstable`; for other value, the union is not available.

A.32 NET_DVR_ALARM_ISAPI_INFO

Structure about Alarm Information Transmitted Based on Text Protocol

| Member | Data Type | Description |
|-------------------------|---------------|---|
| pAlarmData | char* | Alarm information based on text protocol (XML or JSON message without binary data). |
| dwAlarmDataLen | DWORD | Alarm data length. |
| byDataType | BYTE | Alarm data type: 0-invalid, 1-XML, 2-JSON. |
| byPicturesNumber | BYTE | The number of pictures (number of pPicPackData returned). When this member is 1, only one structure of <u>NET_DVR_ALARM_ISAPI_PICDATA</u> will be returned by pPicPackData . When this member is larger than 1, multiple structures of <u>NET_DVR_ALARM_ISAPI_PICDATA</u> will be returned by pPicPackData . |
| byRes | Array of BYTE | Reserved, set to 0. The maximum size is 2 bytes. |
| pPicPackData | void* | Alarm picture structure, see <u>NET_DVR_ALARM_ISAPI_PICDATA</u> for details. |
| byRes | Array of BYTE | Reserved. The maximum size is 32 bytes. |

Remarks

When enabling the listening mode, you should call the network configuration API based on text protocol to set the IP address for the listening service.

A.33 NET_DVR_ALARM_ISAPI_PICDATA

Structure about Alarm Picture Data Transmitted Based on Text Protocol

| Member | Data Type | Description |
|------------|---------------|---|
| dwPicLen | DWORD | Alarm picture data length. |
| byRes | Array of BYTE | Reserved, set to 0. The maximum size is 4 bytes. |
| szFilename | Array of char | Picture file saving path, including file name. The maximum size is 256 bytes. |
| pPicData | BYTE* | Pointer that pointing to the uploaded image data. |

A.34 NET_DVR_ETHERNET_V30

Ethernet Configuration Structure

| Member | Data Type | Description |
|------------------|------------------------------------|--|
| struDVRIP | <u>NET_DVR_IPADDR_UNION</u> | Device IP address |
| struDVRIPMask | <u>NET_DVR_IPADDR_UNION</u> | Mask of device IP address |
| dwNetInterface | DWORD | Network interface type: 1-10MBase-T; 2-10MBase-T (full duplex); 3-100MBase-TX; 4-100M (full duplex); 5-10M/100M/1000M (self-adaptive); 6-1000M (full duplex) |
| wDVRPort | WORD | Device port No. |
| wMTU | WORD | MTU settings, the default is 1500. |
| byMACAddr | Array of BYTE | Device physical address. |
| byEthernetPortNo | BYTE | Network interface No.: 0-invalid, 1-interface 0, 2-interface 1, and so on. This parameter is read-only. |
| byRes | Array of BYTE | Reserved. |

A.35 NET_DVR_IPADDR_UNION

IP Address Union

| Member | Data Type | Description |
|--------|-----------|--|
| szIPv4 | char[] | IPv4 address. The maximum length is 16 bytes. |
| szIPv6 | char[] | IPv6 address. The maximum length is 256 bytes. |

A.36 NET_DVR_NETCFG_V50

Network Configuration Structure

| Member | Data Type | Description |
|----------------------|---|---|
| dwSize | DWORD | Structure size. |
| struEtherNet | Array of <u>NET_DVR_ETHERNET_V30</u> | Ethernet interface |
| struRes1 | Array of | Reserved, set to 0. |
| struAlarmHostIpAddr | <u>NET_DVR_IPADDR_UNION</u> | Listening service IP address |
| byRes2 | Array of BYTE | Reserved, set as 0 |
| wAlarmHostIpPort | WORD | Listening service port No. |
| byUseDhcp | BYTE | Whether to enable DHCP: 0xff- invalid; 0-disable, 1-enable |
| byIPv6Mode | BYTE | Allocation mode of IPv6 address: 0-by router advertisement, 1-by manual setting, 2-by enabling DHCP allocation. |
| struDnsServer1IpAddr | <u>NET_DVR_IPADDR_UNION</u> | IP address of domain name server 1 |
| struDnsServer2IpAddr | <u>NET_DVR_IPADDR_UNION</u> | IP address of domain name server 2 |

| Member | Data Type | Description |
|-----------------------------------|------------------------------------|--|
| byIpResolver | Array of BYTE | IP resolver domain name or IP address (if the port No. of device is 8000, the domain name is not supported). |
| wIpResolverPort | WORD | IP resolver port No. |
| wHttpPortNo | WORD | HTTP port No. |
| struMulticastIpAddr | <u>NET_DVR_IPADDR_UNION</u> | Multicast group address |
| struGatewayIpAddr | <u>NET_DVR_IPADDR_UNION</u> | Gateway address |
| struPPPoE | <u>NET_DVR_PPPOECFG</u> | PPPoE parameters |
| byEnablePrivateMulticastDiscovery | BYTE | Private multicast search (SADP): 0-default, 1-enable, 2-disable |
| byEnableOnvifMulticastDiscovery | BYTE | Onvif multicast search (SADP): 0-default, 1-enable, 2-disable |
| wAlarmHost2IpPort | WORD | Port No. of listening host 2. |
| struAlarmHost2IpAddr | <u>NET_DVR_IPADDR_UNION</u> | IP address of listening host 2 |
| byEnableDNS | BYTE | DNS address setting mode: 0-automatically get, 1-manually set. |
| byRes | Array of BYTE | Reserved, set to 0 |

Remarks

- For device only supports the private protocol with version 3.0 or lower, when the parameter **byUseDhcp**="0xff", you should set the device IP address to null, and then the device will automatically get the DHCP information.
- When the parameter **byIPv6Mode** is set to 0 or 2, setting IPv6 address in the parameter **struEtherNet** is not required, it will be obtained automatically by the device; when **byIPv6Mode** is set to 1, you should set IPv6 address. As there are multiple IPv6 addresses, the IPv6 address of current logged-in device may be different with that in **struEtherNet**.

A.37 NET_DVR_PPPOECFG

PPPoE Configuration Structure

| Member | Data Type | Description |
|----------------|------------------------------------|---------------------------------------|
| dwPPPOE | DWORD | Whether to enable PPPoE: 0-no, 1-yes. |
| sPPPoEUser | Array of BYTE | PPPoE user name. |
| sPPPoEPassword | Array of char | PPPoE password. |
| struPPPoEIP | <u>NET_DVR_IPADDR_UNION</u> | PPPoE IP address |

A.38 NET_DVR_SETUPALARM_PARAM_V50

Arming Parameter Structure

| Member | Data Type | Description |
|---------------------|-----------|--|
| dwSize | DWORD | Structure size. |
| byLevel | BYTE | Arming priority: 0-high, 1-medium, 2-low. |
| byAlarmInfoType | BYTE | Intelligent traffic alarm information type: 0-old (NET_DVR_PLATE_RESULT),1-new (NET_ITS_PLATE_RESULT). |
| byRetAlarmTypeV40 | BYTE | 0-the motion detection, video loss, video tampering, and alarm input alarm information is uploaded in normal mode (alarm type: COMM_ALARM_V30, alarm information structure: <u>NET_DVR_ALARMINFO_V30</u>); 1-alarm information is uploaded in variable size (alarm type: COMM_ALARM_V40, alarm information structure: <u>NET_DVR_ALARMINFO_V40</u>). |
| byRetDevInfoVersion | BYTE | Alarm types of CVR: 0-COMM_ALARM_DEVICE (alarm information structure: <u>NET_DVR_ALARMINFO_DEV</u>), 1-COMM_ALARM_DEVICE_V40 (alarm information structure: <u>NET_DVR_ALARMINFO_DEV_V40</u>). |
| byRetVQDAlarmType | BYTE | VQD alarm types: 0-COMM_ALARM_VQD (alarm information structure: NET_DVR_VQD_ |

| Member | Data Type | Description |
|-----------------------------|-----------|--|
| | | DIAGNOSE_INFO), 1-COMM_ALARM_VQD_EX (alarm information structure: NET_DVR_VQD_ALARM, including camera information and captured pictures) |
| byFaceAlarmDetection | BYTE | Face detection alarm types: 1-face detection alarm (alarm type: COMM_ALARM_FACE_DETECTION, alarm information structure: NET_DVR_FACE_DETECTION), 0-face capture alarm (alarm type: COMM_UPLOAD_FACESNAP_RESULT, alarm information structure: NET_VCA_FACESNAP_RESULT). |
| bySupport | BYTE | Capabilities, which is represented by bit: <ul style="list-style-type: none"> • bit0-whether to upload picture: 0=yes, 1=no • bit1-whether to enable ANR: 0=no, 1=yes • bit4-whether to upload behavior analysis events of all detection targets: 0=no, 1=yes. It is used to enable the NVR to get events of all targets detected by network cameras. • bit5-whether to enable all-day event or alarm uploading: 0=no, 1=yes. It is used to enable the NVR to receive all alarms from network cameras. |
| byBrokenNetHttp | BYTE | ANR type, which is represented by bit and should be supported by device: <ul style="list-style-type: none"> • bit0-whether to enable ANR for ANPR: 0=no, 1=yes. • bit1-whether to enable ANR for people counting: 0=no, 1=yes. • bit2-whether to enable ANR for heat map: 0=no, 1=yes. • bit3-whether to enable ANR for face capture: 0=no, 1=yes. • bit4-whether to enable ANR for face picture comparison: 0=no, 1=yes. • bit5-whether to enable ANR for JSON message transmission: 0=no, 1=yes. |

| Member | Data Type | Description |
|-----------------------|--------------|--|
| | | <ul style="list-style-type: none"> bit6: whether to enable ANR for uploading heat map data by dwell time duration and by people quantity: 0-no, 1-yes. bit7: whether to enable ANR for uploading intersection analysis result: 0-no, 1-yes. |
| wTaskNo | BYTE | Task No. |
| byDeployType | BYTE | Arming type: 0-arm via client software, 1-real-time arming. |
| bySubScripton | BYTE | Subscription parameters, which is represent by bit. Bit7-whether to upload picture after subscribing motion detection alarm by person or vehicle: 0-no, 1-yes. |
| byRes1 | Array [BYTE] | Reserved, set to 0. The maximum size is 2 bytes. |
| byAlarmTypeURL | BYTE | Alarm picture data type, which is represented by bit, if the device supports uploading alarm pictures in binary format and URL format, you can specify the data type to be uploading via this parameter, if the device only supports URL format, this parameter is invalid. If the URL format is selected, you should set the device and enable the cloud storage, otherwise, the picture will still be transmitted in binary format. <ul style="list-style-type: none"> bit0-type of captured face pictures: 0-binary data, 1-URL bit1-type of picture uploaded in message: 0-binary, 1-URL bit2-type of picture uploaded for face picture comparison: 0-binary, 1-URL |
| byCustomCtrl | BYTE | Custom control type, which is represented by bit, bit0-whether to upload the face thumbnail of the front passenger: 0-no, 1-yes |
| byRes4 | Array [BYTE] | Reserved, set to 0. The maximum size is 128 bytes. |

Remarks

- The parameters **byLevel** and **byAlarmInfoType** are available for traffic cameras. Up to 1 cameras can be armed in the priority of level 0, up to 3 cameras can be armed in the priority of level 1, and up to 5 cameras can be armed in the priority of level 3, the alarm/event information from the camera in highest priority will be uploaded first.
- For arming via client software, only supports arming one channel, and supports uploading the alarm/event when device is offline; for real-time arming, up to four channels can be armed at same time, but uploading alarm/event when device is offline is not supported.
- The parameter **wTaskNo** is used to distinguish different arming connections. If the value of this parameter in different arming connections is same, error will be returned.

A.39 NET_DVR_TIME

Time Parameter Structure

| Member | Data Type | Description |
|----------|-----------|-------------|
| dwYear | DWORD | Year |
| dwMonth | DWORD | Month |
| dwDay | DWORD | Day |
| dwHour | DWORD | Hour |
| dwMinute | DWORD | Minute |
| dwSecond | DWORD | Second |

A.40 NET_DVR_TIME_EX

Extended Time Parameter Structure

| Member | Data Type | Description |
|----------|-----------|-------------|
| wYear | WORD | Year |
| byMonth | BYTE | Month |
| byDay | BYTE | Day |
| byHour | BYTE | Hour |
| byMinute | BYTE | Minute |

| Member | Data Type | Description |
|----------|-----------|-------------|
| bySecond | BYTE | Second |
| byRes | BYTE | Reserved. |

Appendix B. Appendixes

B.1 Device Network SDK Errors

The errors that may occur during the device network SDK integration are listed here for reference. You can search for the error descriptions according to the error codes or names returned by a specific API (NET_DVR_GetLastError or NET_DVR_GetErrorMsg).

General Errors

| Error Name | Error Code | Error Description |
|------------------------------|------------|--|
| NET_DVR_NOERROR | 0 | No error. |
| NET_DVR_PASSWORD_ERROR | 1 | Incorrect user name or password. |
| NET_DVR_NOENOUGHPRI | 2 | No permission. |
| NET_DVR_NOINIT | 3 | Uninitialized. |
| NET_DVR_CHANNEL_ERROR | 4 | Incorrect channel No. |
| NET_DVR_OVER_MAXLINK | 5 | No more device can be connected. |
| NET_DVR_VERSIONNOMATCH | 6 | Version mismatches. |
| NET_DVR_NETWORK_FAIL_CONNECT | 7 | Connecting to device failed. The device is offline or network connection timed out. |
| NET_DVR_NETWORK_SEND_ERROR | 8 | Sending data to device failed. |
| NET_DVR_NETWORK_RECV_ERROR | 9 | Receiving data from device failed. |
| NET_DVR_NETWORK_RECV_TIMEOUT | 10 | Receiving data from device timed out. |
| NET_DVR_NETWORK_ERRORDATA | 11 | The data sent to the device is illegal, or the data received from the device error. E.g. The input data is not supported by the device for remote configuration. |
| NET_DVR_ORDER_ERROR | 12 | API calling order error. |
| NET_DVR_OPERNOPERMIT | 13 | No permission for this operation. |
| NET_DVR_COMMANDTIMEOUT | 14 | Executing device command timed out. |
| NET_DVR_ERRORSERIALPORT | 15 | Incorrect serial port No. The specified serial port does not exist. |

| Error Name | Error Code | Error Description |
|-------------------------------|------------|--|
| NET_DVR_ERRORALARMPORT | 16 | Alarm port No. error. The alarm input or output port of the specified device does not exist. |
| NET_DVR_PARAMETER_ERROR | 17 | Incorrect parameter. The input or output parameters of the SDK API is empty, or the parameter value or format is invalid. |
| NET_DVR_CHAN_EXCEPTION | 18 | Device channel is in exception status. |
| NET_DVR_NODISK | 19 | No HDD in the device. |
| NET_DVR_ERRORDISKNUM | 20 | Incorrect HDD No. |
| NET_DVR_DISK_FULL | 21 | HDD full. |
| NET_DVR_DISK_ERROR | 22 | HDD error. |
| NET_DVR_NOSUPPORT | 23 | Device does not support this function. |
| NET_DVR_BUSY | 24 | Device is busy. |
| NET_DVR_MODIFY_FAIL | 25 | Failed to edit device parameters. |
| NET_DVR_PASSWORD_FORMAT_ERROR | 26 | Invalid password format. |
| NET_DVR_DISK_FORMATING | 27 | HDD is formatting. Failed to startup. |
| NET_DVR_DVRNORESOURCE | 28 | Insufficient device resources. |
| NET_DVR_DVROPRATEFAILED | 29 | Device operation failed. |
| NET_DVR_OPENHOSTSOUND_FAIL | 30 | Failed to collect local audio data or open audio output during two-way audio and broadcast. |
| NET_DVR_DVRVOICEOPENED | 31 | Two-way audio channel is occupied. |
| NET_DVR_TIMEINPUTERROR | 32 | Incorrect time input. |
| NET_DVR_NOSPECFILE | 33 | No video file for playback. |
| NET_DVR_CREATEFILE_ERROR | 34 | Failed to create a file during local recording, saving picture, getting configuration file or downloading video file remotely. |
| NET_DVR_FILEOPENFAIL | 35 | Failed to open a file. The file does not exist or directory error. |

| Error Name | Error Code | Error Description |
|------------------------------|------------|---|
| NET_DVR_OPERNOTFINISH | 36 | Operation conflicted. |
| NET_DVR_GETPLAYTIMEFAIL | 37 | Failed to get the current played time. |
| NET_DVR_PLAYFAIL | 38 | Failed to play. |
| NET_DVR_FILEFORMAT_ERROR | 39 | Invalid file format. |
| NET_DVR_DIR_ERROR | 40 | File directory error. |
| NET_DVR_ALLOC_RESOURCE_ERROR | 41 | Allocating resources failed. |
| NET_DVR_AUDIO_MODE_ERROR | 42 | Invalid sound card mode error. The opened sound play mode and configured mode mismatched. |
| NET_DVR_NOENOUGH_BUF | 43 | Insufficient buffer for receiving data or saving picture. |
| NET_DVR_CREATESOCKET_ERROR | 44 | Failed to create SOCKET. |
| NET_DVR_SETSOCKET_ERROR | 45 | Failed to set SOCKET. |
| NET_DVR_MAX_NUM | 46 | No more registrations and live views can be connected. |
| NET_DVR_USERNOTEXIST | 47 | The user does not exist. The user ID is logged out or unavailable. |
| NET_DVR_WRITEFLASHERROR | 48 | Writing FLASH error during device upgrade. |
| NET_DVR_UPGRADEFAIL | 49 | Failed to upgrade device. Network problem or language mismatches. |
| NET_DVR_CARDHAVEINIT | 50 | The decoding card is already initialized. |
| NET_DVR_PLAYERFAILED | 51 | Failed to call the function of player SDK. |
| NET_DVR_MAX_USERNUM | 52 | No more users can log in to. |
| NET_DVR_GETLOCALIPANDMACFAIL | 53 | Failed to get the IP address or physical address of local PC. |
| NET_DVR_NOENCODEING | 54 | The decoding function of this channel is not enabled. |
| NET_DVR_IPMISMATCH | 55 | IP address mismatches. |

| Error Name | Error Code | Error Description |
|---------------------------------|------------|---|
| NET_DVR_MACMISMATCH | 56 | MAC address mismatches. |
| NET_DVR_UPGRADELANGMISMATCH | 57 | The language of upgrade file mismatches. |
| NET_DVR_MAX_PLAYERPORT | 58 | No more channels can be started to play. |
| NET_DVR_NOSPACEBACKUP | 59 | Insufficient space to back up file. |
| NET_DVR_NODEVICEBACKUP | 60 | No backup device found. |
| NET_DVR_PICTURE_BITS_ERROR | 61 | Picture pixel bit mismatches. Only 24 bits are allowed. |
| NET_DVR_PICTURE_DIMENSION_ERROR | 62 | Too large picture. The height*width should be less than 128x256. |
| NET_DVR_PICTURE_SIZ_ERROR | 63 | Too large picture. The picture size should be smaller than 100K. |
| NET_DVR_LOADPLAYERSDKFAILED | 64 | Failed to load the player(PlayCtrl.dll, SuperRender.dll, AudioRender.dll) to the current directory. |
| NET_DVR_LOADPLAYERSDKPROC_ERROR | 65 | Failed to find the function in player SDK. |
| NET_DVR_LOADDSSDKFAILED | 66 | Failed to load the DS SDK to the current directory. |
| NET_DVR_LOADDSSDKPROC_ERROR | 67 | Failed to find the function in the DS SDK. |
| NET_DVR_DSSDK_ERROR | 68 | Failed to call the API in the hardware decoding library. |
| NET_DVR_VOICEMONOPOLIZE | 69 | The sound card is exclusive. |
| NET_DVR_JOINMULTICASTFAILED | 70 | Failed to join to multicast group. |
| NET_DVR_CREATEDIR_ERROR | 71 | Failed to create log file directory. |
| NET_DVR_BINDSOCKET_ERROR | 72 | Failed to bind socket. |
| NET_DVR_SOCKETCLOSE_ERROR | 73 | Socket disconnected. Network disconnected or the destination is unreachable. |

| Error Name | Error Code | Error Description |
|-----------------------------|------------|---|
| NET_DVR_USERID_ISUSING | 74 | Operation is executing. Failed to log out. |
| NET_DVR_SOCKETLISTEN_ERROR | 75 | Failed to listen. |
| NET_DVR_PROGRAM_EXCEPTION | 76 | Program exception. |
| NET_DVR_WRITEFILE_FAILED | 77 | Failed to write file during local recording, downloading file remotely or saving picture. |
| NET_DVR_FORMAT_READONLY | 78 | The HDD is read-only. Formatting is forbidden. |
| NET_DVR_WITHSAMEUSERNAME | 79 | The user name already exists. |
| NET_DVR_DEVICETYPE_ERROR | 80 | Device model mismatches when importing parameters. |
| NET_DVR_LANGUAGE_ERROR | 81 | Language mismatches when importing parameters. |
| NET_DVR_PARAVERSION_ERROR | 82 | Software version mismatches when importing parameters. |
| NET_DVR_IPCHAN_NOTALIVE | 83 | The external IP channel is offline live view. |
| NET_DVR_RTSP_SDK_ERROR | 84 | Failed to load StreamTransClient.dll. |
| NET_DVR_CONVERT_SDK_ERROR | 85 | Failed to load SystemTransform.dll. |
| NET_DVR_IPC_COUNT_OVERFLOW | 86 | No more IP channels can access to. |
| NET_DVR_MAX_ADD_NUM | 87 | No more video tags can be added. |
| NET_DVR_PARAMMODE_ERROR | 88 | Invalid parameter mode of image enhancement. |
| NET_DVR_CODESPITTER_OFFLINE | 89 | Code distributer is offline. |
| NET_DVR_BACKUP_COPYING | 90 | Device is backing up. |
| NET_DVR_CHAN_NOTSUPPORT | 91 | This operation is not supported by the channel. |
| NET_DVR_CALLINEINVALID | 92 | The height line is too concentrated, or the length line is not inclined enough. |

| Error Name | Error Code | Error Description |
|------------------------------------|------------|--|
| NET_DVR_CALCANCELCONFLICT | 93 | Cancel calibration conflict, if the rule and global actual size filter are configured. |
| NET_DVR_CALPOINTOUTRANGE | 94 | The calibration point is out of limitation. |
| NET_DVR_FILTERRECTINVALID | 95 | The size filter does not meet the requirement. |
| NET_DVR_DDNS_DEVOFFLINE | 96 | Device has not registered to DDNS. |
| NET_DVR_DDNS_INTER_ERROR | 97 | DDNS internal error. |
| NET_DVR_FUNCTION_NOT_SUPPORT_OS | 98 | This function is not supported by this Operating system. |
| NET_DVR_DEC_CHAN_REBIND | 99 | Decoding channel binding display output is limited. |
| NET_DVR_INTERCOM_SDK_ERROR | 100 | Failed to load the two-way audio SDK of the current directory. |
| NET_DVR_NO_CURRENT_UPDATEFILE | 101 | No correct upgrade packet. |
| NET_DVR_USER_NOT_SUCC_LOGIN | 102 | Login failed. |
| NET_DVR_USE_LOG_SWITCH_FILE | 103 | The log switch file is under using. |
| NET_DVR_POOL_PORT_EXHAUST | 104 | No port can be bound in the port pool. |
| NET_DVR_PACKET_TYPE_NOT_SUPPORT | 105 | Incorrect stream packaging format. |
| NET_DVR_IPPARA_IPID_ERROR | 106 | Incorrect IPID for IP access configuration. |
| NET_DVR_LOAD_HCPREVIEW_SDK_ERROR | 107 | Failed to load the live view component. |
| NET_DVR_LOAD_HCVOICETALK_SDK_ERROR | 108 | Failed to load the audio component. |
| NET_DVR_LOAD_HCALARM_SDK_ERROR | 109 | Failed to load the alarm component. |
| NET_DVR_LOAD_HCPLAYBACK_SDK_ERROR | 110 | Failed to load the playback component. |

| Error Name | Error Code | Error Description |
|---|------------|---|
| NET_DVR_LOAD_HCDISPLAY_SDK_ERROR | 111 | Failed to load the display component. |
| NET_DVR_LOAD_HCINDUSTRY_SDK_ERROR | 112 | Failed to load application component. |
| NET_DVR_LOAD_HCGENERALCFGMGR_SDK_ERROR | 113 | Failed to load the general configuration management component. |
| NET_DVR_CORE_VER_MISMATCH | 121 | Component version and core version mismatched when loading the component singly. |
| NET_DVR_CORE_VER_MISMATCH_HCPREVIEW | 122 | Live view component version and core version mismatched. |
| NET_DVR_CORE_VER_MISMATCH_HCVOICETALK | 123 | Audio component version and the core version mismatched. |
| NET_DVR_CORE_VER_MISMATCH_HCALARM | 124 | Alarm component version and the core version mismatched. |
| NET_DVR_CORE_VER_MISMATCH_HCPLAYBACK | 125 | Playback component version and the core version mismatched. |
| NET_DVR_CORE_VER_MISMATCH_HCDISPLAY | 126 | Display component version and the core version mismatched. |
| NET_DVR_CORE_VER_MISMATCH_HCINDUSTRY | 127 | Application component version and the core version mismatched. |
| NET_DVR_CORE_VER_MISMATCH_HCGENERALCFGMGR | 128 | General configuration management component version and the core version mismatched. |
| NET_DVR_COM_VER_MISMATCH_HCPREVIEW | 136 | Live view component version and SDK version mismatched. |
| NET_DVR_COM_VER_MISMATCH_HCVOICETALKy | 137 | Audio component version and SDK version mismatched. |
| NET_DVR_COM_VER_MISMATCH_HCALARM | 138 | Alarm component version and SDK version mismatched. |
| NET_DVR_COM_VER_MISMATCH_HCPLAYBACK | 139 | Playback component version and SDK version mismatched. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_COM_VER_MISMATCH_HCDISPLAY | 140 | Display component version and SDK version mismatched. |
| NET_DVR_COM_VER_MISMATCH_HCINDUSTRY | 141 | Application component version and SDK version mismatched. |
| NET_DVR_COM_VER_MISMATCH_HCGENERALCFGMGR | 142 | General configuration management component version and SDK version mismatched. |
| NET_DVR_ALIAS_DUPLICATE | 150 | Duplicated alias(for HiDDNS configuration). |
| NET_DVR_USERNAME_NOT_EXIST | 152 | User name does not exist (error code of network camera and network speed dome with version from 5.1.7 to 5.3.1). |
| NET_ERR_USERNAME_LOCKED | 153 | The user name is locked. |
| NET_DVR_INVALID_USERID | 154 | Invalid user ID. |
| NET_DVR_LOW_LOGIN_VERSION | 155 | The version is too low. |
| NET_DVR_LOAD_LIBEAY32_DLL_ERROR | 156 | Failed to load libeay32.dll. |
| NET_DVR_LOAD_SSLEAY32_DLL_ERROR | 157 | Failed to load ssleay32.dll. |
| NET_ERR_LOAD_LIBICONV | 158 | Failed to load libiconv.dll. |
| NET_ERR_SSL_CONNECT_FAILED | 159 | Connecting to SSL failed. |
| NET_DVR_TEST_SERVER_FAIL_CONNECT | 165 | Failed to connect to test server. |
| NET_DVR_NAS_SERVER_INVALID_DIR | 166 | Failed to load NAS server to the directory, Invalid directory, or incorrect user name and password. |
| NET_DVR_NAS_SERVER_NOENOUGH_PRI | 167 | Failed to load NAS server th the directory. No permission. |
| NET_DVR_EMAIL_SERVER_NOT_CONFIG_DNS | 168 | The server uses domain name without configuring DNS, the domain name may be invalid. |

| Error Name | Error Code | Error Description |
|--|------------|---|
| NET_DVR_EMAIL_SERVER_NOT_CONFIG_GATEWAY | 169 | No gateway configured. Sending email may be failed. |
| NET_DVR_TEST_SERVER_PASSWORD_ERROR | 170 | Incorrect user name or password of test server. |
| NET_DVR_EMAIL_SERVER_CONNECT_EXCEPTION_WITH_SMTP | 171 | Interaction exception between device and SMTP server. |
| NET_DVR_FTP_SERVER_FAIL_CREATE_DIR | 172 | FTP server creating directory failed. |
| NET_DVR_FTP_SERVER_NO_WRITE_PIR | 173 | FTP server has no writing permission. |
| NET_DVR_IP_CONFLICT | 174 | IP conflicted. |
| NET_DVR_INSUFFICIENT_STORAGEPOOL_SPACE | 175 | Storage pool space is full. |
| NET_DVR_STORAGEPOOL_INVALID | 176 | Invalid cloud storage pool. No storage pool configured or incorrect storage pool ID. |
| NET_DVR_EFFECTIVENESS_REBOOT | 177 | Restart to take effect. |
| NET_ERR_ANR_ARMING_EXIST | 178 | The ANR arming connection already exists(the error will be returned when arming with ANR function if the private SDK protocol arming connection is established). |
| NET_ERR_UPLOADLINK_EXIST | 179 | The ANR uploading connection already exists(the error will be returned when EHome protocol and private SDK protocol do not support ANR at the same time). |
| NET_ERR_INCORRECT_FILE_FORMAT | 180 | The imported file format is incorrect. |
| NET_ERR_INCORRECT_FILE_CONTENT | 181 | The imported file content is incorrect. |
| NET_ERR_MAX_HRUDP_LINK | 182 | No more HRUDP can be connected to device. |
| NET_ERR_MAX_PORT_MULTIPLEX | 183 | Maximum number of multiplexed ports reaches. |
| NET_ERR_CREATE_PORT_MULTIPLEX | 184 | Creating port multiplier failed. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_NONBLOCKING_CAPTURE_NOTSUPPORT | 185 | Non-blocking picture capture is not supported. |
| NET_SDK_ERR_FUNCTION_INVALID | 186 | Invalid function. The asynchronous mode is enabled. |
| NET_SDK_ERR_MAX_PORT_MULTIPLEX | 187 | Maximum number of multiplex ports reached. |
| NET_DVR_INVALID_LINK | 188 | Link has not been created or the link is invalid. |
| NET_DVR_NAME_NOT_ONLY | 200 | This name already exists. |
| NET_DVR_OVER_MAX_ARRAY | 201 | The number of RAID reaches the upper-limit. |
| NET_DVR_OVER_MAX_VD | 202 | The number of virtual disk reaches the upper-limit. |
| NET_DVR_VD_SLOT_EXCEED | 203 | The virtual disk slots are full. |
| NET_DVR_PD_STATUS_INVALID | 204 | The physical disk for rebuilding RAID is error. |
| NET_DVR_PD_BE_DEDICATE_SPARE | 205 | The physical disk for rebuilding RAID is specified as hot spare. |
| NET_DVR_PD_NOT_FREE | 206 | The physical disk for rebuilding RAID is busy. |
| NET_DVR_CANNOT_MIG2NEWMODE | 207 | Failed to migrate the current RAID type to the new type. |
| NET_DVR_MIG_PAUSE | 208 | Migration is paused. |
| NET_DVR_MIG_ABOUTED | 209 | Migration is cancelled. |
| NET_DVR_EXIST_VD | 210 | Failed to delete RAID. Virtual disk exists in the RAID. |
| NET_DVR_TARGET_IN_LD_FUNCTIONAL | 211 | Target physical disk is a part of the virtual disk and it is working normally. |
| NET_DVR_HD_IS_ASSIGNED_ALREADY | 212 | The specified physical disk is allocated as virtual disk. |
| NET_DVR_INVALID_HD_COUNT | 213 | The number of physical disks and specified RAID level mismatched. |

| Error Name | Error Code | Error Description |
|------------------------------|------------|--|
| NET_DVR_LD_IS_FUNCTIONAL | 214 | The RAID is normal. Failed to rebuild. |
| NET_DVR_BGA_RUNNING | 215 | Background task is executing. |
| NET_DVR_LD_NO_ATAPI | 216 | Failed to create virtual disk by ATAPI disk. |
| NET_DVR_MIGRATION_NOT_NEED | 217 | There is no need to migrate the RAID. |
| NET_DVR_HD_TYPE_MISMATCH | 218 | The physical disk type is not allowed. |
| NET_DVR_NO_LD_IN_DG | 219 | No virtual disk. Operation failed. |
| NET_DVR_NO_ROOM_FOR_SPARE | 220 | Insufficient disk space. Failed to allocate the disk as hot spare. |
| NET_DVR_SPARE_IS_IN_MULTI_DG | 221 | The disk is already allocated as the hot spare of one RAID. |
| NET_DVR_DG_HAS_MISSING_PD | 222 | No disk in the RAID. |
| NET_DVR_NAME_EMPTY | 223 | The name is empty. |
| NET_DVR_INPUT_PARAM | 224 | Incorrect input parameters. |
| NET_DVR_PD_NOT_AVAILABLE | 225 | The physical disk is not available. |
| NET_DVR_ARRAY_NOT_AVAILABLE | 226 | The RAID is not available. |
| NET_DVR_PD_COUNT | 227 | Incorrect number of physical disks. |
| NET_DVR_VD_SMALL | 228 | Insufficient virtual disk space. |
| NET_DVR_NO_EXIST | 229 | Not exist. |
| NET_DVR_NOT_SUPPORT | 230 | This operation is not supported. |
| NET_DVR_NOT_FUNCTIONAL | 231 | The RAID status is exception. |
| NET_DVR_DEV_NODE_NOT_FOUND | 232 | The device node of virtual disk does not exist. |
| NET_DVR_SLOT_EXCEED | 233 | No more slots are allowed. |
| NET_DVR_NO_VD_IN_ARRAY | 234 | No virtual disk exists in the RAID. |
| NET_DVR_VD_SLOT_INVALID | 235 | Invalid virtual disk slot. |
| NET_DVR_PD_NO_ENOUGH_SPACE | 236 | Insufficient physical disk space. |
| NET_DVR_ARRAY_NONFUNCTION | 237 | Only the RAID in normal status supports to be migrated. |

| Error Name | Error Code | Error Description |
|------------------------------------|------------|---|
| NET_DVR_ARRAY_NO_ENOUGH_SPACE | 238 | Insufficient RAID space. |
| NET_DVR_STOPPING_SCANNING_ARRAY | 239 | Pulling disk out safely or rescanning. |
| NET_DVR_NOT_SUPPORT_16T | 240 | Creating RAID with size larger than 16T is not supported. |
| NET_DVR_ERROR_DEVICE_NOT_ACTIVATED | 250 | The device is not activated (login failed.) |
| NET_DVR_ERROR_RISK_PASSWORD | 251 | Risky password. |
| NET_DVR_ERROR_DEVICE_HAS_ACTIVATED | 252 | The device is already activated. |
| NET_DVR_ID_ERROR | 300 | The configured ID is invalid. |
| NET_DVR_POLYGON_ERROR | 301 | Invalid polygon shape. |
| NET_DVR_RULE_PARAM_ERROR | 302 | Invalid rule parameters. |
| NET_DVR_RULE_CFG_CONFLICT | 303 | Configured information conflicted. |
| NET_DVR_CALIBRATE_NOT_READY | 304 | No calibration information. |
| NET_DVR_CAMERA_DATA_ERROR | 305 | Invalid camera parameters. |
| NET_DVR_CALIBRATE_DATA_UNFIT | 306 | Invalid inclination angle for calibration. |
| NET_DVR_CALIBRATE_DATA_CONFLICT | 307 | Calibration error. |
| NET_DVR_CALIBRATE_CALC_FAIL | 308 | Failed to calculate calibration parameter values of camera. |
| NET_DVR_CALIBRATE_LINE_OUT_RECT | 309 | The inputted calibration line exceeds the external sample rectangle. |
| NET_DVR_ENTER_RULE_NOT_READY | 310 | No region entrance is configured. |
| NET_DVR_AID_RULE_NO_INCLUDE_LANE | 311 | No lane configured in the traffic event rule (especially for traffic jam or driving against the traffic). |
| NET_DVR_LANE_NOT_READY | 312 | Lane not configured. |
| NET_DVR_RULE_INCLUDE_TWO_WAY | 313 | Two different directions are contained in event rule. |

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|--|
| NET_DVR_LANE_TPS_RULE_CONFLICT | 314 | Lane and data rule conflicted. |
| NET_DVR_NOT_SUPPORT_EVENT_TYPE | 315 | This event type is not supported. |
| NET_DVR_LANE_NO_WAY | 316 | The lane has no direction. |
| NET_DVR_SIZE_FILTER_ERROR | 317 | Invalid size of filter frame. |
| NET_DVR_LIB_FFL_NO_FACE | 318 | No face picture exists in the image inputted when positioning feature point. |
| NET_DVR_LIB_FFL_IMG_TOO_SMALL | 319 | The inputted image is too small when positioning feature point. |
| NET_DVR_LIB_FD_IMG_NO_FACE | 320 | No face picture exists in the image inputted when detecting single face picture. |
| NET_DVR_LIB_FACE_TOO_SMALL | 321 | Face picture is too small when building model. |
| NET_DVR_LIB_FACE_QUALITY_TOO_BAD | 322 | The face picture quality is too poor when building model. |
| NET_DVR_KEY_PARAM_ERR | 323 | The configured advanced parameter is incorrect. |
| NET_DVR_CALIBRATE_DATA_ERR | 324 | Calibration sample number error, or data value error, or the sample points are beyond the horizontal line. |
| NET_DVR_CALIBRATE_DISABLE_FAIL | 325 | Canceling calibration is not allowed for configured rules. |
| NET_DVR_VCA_LIB_FD_SCALE_OUTRANGE | 326 | The minimum width and height of maximum filter frame are twice or more larger than the maximum width and height of minimum filter frame. |
| NET_DVR_LIB_FD_REGION_TOO_LARGE | 327 | Too large detection region. The maximum region should be 2/3 of the image. |
| NET_DVR_TRIAL_OVERDUE | 328 | Trial period is ended. |
| NET_DVR_CONFIG_FILE_CONFLICT | 329 | Device type and configuration file conflicted. |

| Error Name | Error Code | Error Description |
|------------------------------------|------------|--|
| NET_DVR_FR_FPL_FAIL | 330 | Failed to positioning face feature points. |
| NET_DVR_FR_IQA_FAIL | 331 | Failed to test face picture quality. |
| NET_DVR_FR_FEM_FAIL | 332 | Failed to extract the face feature points. |
| NET_DVR_FPL_DT_CONF_TOO_LOW | 333 | The face detection validity is too low when positioning face feature points. |
| NET_DVR_FPL_CONF_TOO_LOW | 334 | The validity of feature points positionong is too low. |
| NET_DVR_E_DATA_SIZE | 335 | Data size mismatches. |
| NET_DVR_FR_MODEL_VERSION_ERR | 336 | Incorrect model version in face model library. |
| NET_DVR_FR_FD_FAIL | 337 | Failed to detect face in the face recognition library. |
| NET_DVR_FA_NORMALIZE_ERR | 338 | Failed to normalize face attribute. |
| NET_DVR_DOG_PUSTREAM_NOT_MATCH | 339 | Dongle type and camera type mismatched. |
| NET_DVR_DEV_PUSTREAM_NOT_MATCH | 340 | Camera version mismatches. |
| NET_DVR_PUSTREAM_ALREADY_EXISTS | 341 | This camera is already added to other channels of devices. |
| NET_DVR_SEARCH_CONNECT_FAILED | 342 | Failed to connect to face retrieval server. |
| NET_DVR_INSUFFICIENT_DISK_SPACE | 343 | Insufficient storage space. |
| NET_DVR_DATABASE_CONNECTION_FAILED | 344 | Failed to connect to database. |
| NET_DVR_DATABASE_ADM_PW_ERROR | 345 | Incorrect database user name and password. |
| NET_DVR_DECODE_YUV | 346 | Decoding failed. |
| NET_DVR_IMAGE_RESOLUTION_ERROR | 347 | Invalid picture resolution |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_CHAN_WORKMODE_ERROR | 348 | Invalid channel working mode. |
| NET_ERROR_TRUNK_LINE | 711 | Sub system is configured as the trunk line. |
| NET_ERROR_MIXED_JOINT | 712 | Mixed joint is not supported. |
| NET_ERROR_DISPLAY_SWITCH | 713 | Switch of display channel is not supported. |
| NET_ERROR_USED_BY_BIG_SCREEN | 714 | Decoded resource is occupied by the big screen. |
| NET_ERROR_USE_OTHER_DEC_RESOURCE | 715 | Using resources of other sub system is not allowed. |
| NET_ERROR_SCENE_USING | 717 | The scene is being used. |
| NET_ERR_NO_ENOUGH_DEC_RESOURCE | 718 | Insufficient resources for decoding. |
| NET_ERR_NO_ENOUGH_FREE_SHOW_RESOURCE | 719 | Insufficient resources for display. |
| NET_ERR_NO_ENOUGH_VIDEO_MEMORY | 720 | Insufficient video storage resources. |
| NET_ERR_MAX_VIDEO_NUM | 721 | Insufficient resources for multiple channels. |
| NET_ERR_WINDOW_COVER_FREE_SHOW_AND_NORMAL | 722 | Windows cover free display output channel and normal output channel. |
| NET_ERR_FREE_SHOW_WINDOW_SPLIT | 723 | Window division is not supported for free display windows. |
| NET_ERR_INAPPROPRIATE_WINDOW_FREE_SHOW | 724 | For the windows whose number is not integral multiple of the number of output channels, free display is not supported. |
| NET_DVR_TRANSPARENT_WINDOW_NOT_SUPPORT_SPLIT | 725 | For windows whose transparency configuration is enabled, window division is not supported. |
| NET_DVR_SPLIT_WINDOW_NOT_SUPPORT_TRANSPARENT | 726 | For windows whose window division is enabled, transparency configuration is not supported. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_ERR_TERMINAL_BUSY | 780 | The terminal busy. |
| NET_DVR_FUNCTION_RESOURCE_USAGE_ERROR | 791 | Failed to enable this function. The resources is occupied by other functions. |
| NET_DVR_DEV_NET_OVERFLOW | 800 | Network traffic is out of the limitation. |
| NET_DVR_STATUS_RECORDFILE_WRITING_NOT_LOCK | 801 | Failed to lock. The video file is recording. |
| NET_DVR_STATUS_CANT_FORMAT_LITTLE_DISK | 802 | Failed to format HDD. The HDD space is too small. |
| NET_SDK_ERR_REMOTE_DISCONNECT | 803 | Failed to connect to the remote terminal. |
| NET_SDK_ERR_RD_ADD_RD | 804 | Spare server cannot be added to spare server. |
| NET_SDK_ERR_BACKUP_DISK_EXCEPT | 805 | Backup disk exception. |
| NET_SDK_ERR_RD_LIMIT | 806 | No more spare server can be added. |
| NET_SDK_ERR_ADDED_RD_IS_WD | 807 | The added spare server is a working server. |
| NET_SDK_ERR_ADD_ORDER_WRONG | 808 | Adding flow error. |
| NET_SDK_ERR_WD_ADD_WD | 809 | Working server cannot be added to working server. |
| NET_SDK_ERR_WD_SERVICE_EXCETP | 810 | CVR service exception (For N+1 mode, it refers to CVR working server exception). |
| NET_SDK_ERR_RD_SERVICE_EXCETP | 811 | Spare CVR server exception. |
| NET_SDK_ERR_ADDED_WD_IS_RD | 812 | The added working server is spare server. |
| NET_SDK_ERR_PERFORMANCE_LIMIT | 813 | The performance reaches the upper-limit. |
| NET_SDK_ERR_ADDED_DEVICE_EXIST | 814 | This device already exists. |
| NET_SDK_ERR_INQUEST_RESUMING | 815 | Inquest resuming. |
| NET_SDK_ERR_RECORD_BACKUPING | 816 | Inquest video backing up. |

| Error Name | Error Code | Error Description |
|--------------------------------------|------------|---|
| NET_SDK_ERR_DISK_PLAYING | 817 | Playing. |
| NET_SDK_ERR_INQUEST_STARTED | 818 | Inquest started. |
| NET_SDK_ERR_LOCAL_OPERATING | 819 | Locally operating. |
| NET_SDK_ERR_INQUEST_NOT_START | 820 | Inquest is not started. |
| NET_SDK_ERR_CHAN_AUDIO_BIND | 821 | The channel is not bound or binding two-way audio failed. |
| NET_DVR_N_PLUS_ONE_MODE | 822 | Ddevice is in N+1 mode. Cloud storage is not supported. |
| NET_DVR_CLOUD_STORAGE_OPENED | 823 | Cloud storage mode is enbaled. |
| NET_DVR_ERR_OPER_NOT_ALLOWED | 824 | Operation failed. The device is in N+0 taken over status. |
| NET_DVR_ERR_NEED_RELOCATE | 825 | The device is in N+0 taken over status. Get re-positioning information and try again. |
| NET_SDK_ERR_IR_PORT_ERROR | 830 | IR output error. |
| NET_SDK_ERR_IR_CMD_ERROR | 831 | IR output port command number error |
| NET_SDK_ERR_NOT_INQUESTING | 832 | Device is not in inquest status. |
| NET_SDK_ERR_INQUEST_NOT_PAUSED | 833 | Device is not in paused status. |
| NET_DVR_CHECK_PASSWORD_MISTAKE_ERROR | 834 | Incorrect verification code. |
| NET_DVR_CHECK_PASSWORD_NULL_ERROR | 835 | Verification code is required. |
| NET_DVR_UNABLE_CALIB_ERROR | 836 | Failed to calibrate. |
| NET_DVR_PLEASE_CALIB_ERROR | 837 | Calibration first. |
| NET_DVR_ERR_PANORAMIC_CAL_EMPTY | 838 | Panoramic calibration is empty in Flash. |
| NET_DVR_ERR_CALIB_FAIL_PLEASEAGAIN | 839 | Calibration failed, please try again. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_ERR_DETECTION_LINE | 840 | Rule line configuration error. Please try again and make sure the line is within the red region. |
| NET_DVR_EXCEED_FACE_IMAGES_ERROR | 843 | No more face pictures can be added. |
| NET_DVR_ANALYSIS_FACE_IMAGES_ERROR | 844 | Picture recognition failed. |
| NET_ERR_ALARM_INPUT_OCCUPIED | 845 | A<-1 alarm number is used for triggering vehicle capture. |
| NET_DVR_FACELIB_DATABASE_ERROR | 846 | Database version in face picture library mismatched. |
| NET_DVR_FACELIB_DATA_ERROR | 847 | Face picture library data error. |
| NET_DVR_FACE_DATA_ID_ERROR | 848 | Invalid face data PID. |
| NET_DVR_FACELIB_ID_ERROR | 849 | Invalid face picture library ID. |
| NET_DVR_EXCEED_FACE_LIBRARY_ERROR | 850 | No more face picture libraries can be established.. |
| NET_DVR_PIC_ANALYSIS_NO_TARGET_ERROR | 851 | No target recognized in the picture. |
| NET_DVR_SUBPIC_ANALYSIS_MODELING_ERROR | 852 | Sub picture modeling failed. |
| NET_DVR_PIC_ANALYSIS_NO_RESOURCE_ERROR | 853 | No VCA engine supports picture secondary recognition. |
| NET_DVR_ANALYSIS_ENGINES_NO_RESOURCE_ERROR | 854 | No VCA engine. |
| NET_DVR_ANALYSIS_ENGINES_USAGE_EXCEED_ERROR | 855 | Overload. The engine CPU reached 100%. |
| NET_DVR_EXCEED_HUMANMISINFO_FILTER_ENABLED_ERROR | 856 | No more false alarm channel can be enabled. |
| NET_DVR_NAME_ERROR | 857 | Name error. |
| NET_DVR_NAME_EXIST_ERROR | 858 | The name already exists. |
| NET_DVR_FACELIB_PIC_IMPORTING_ERROR | 859 | The pictures is importing to face picture library. |

| Error Name | Error Code | Error Description |
|--|------------|---|
| NET_DVR_PIC_FORMAT_ERROR | 864 | Invalid picture format. |
| NET_DVR_PIC_RESOLUTION_INVALID_ERROR | 865 | Invalid picture resolution. |
| NET_DVR_PIC_SIZE_EXCEED_ERROR | 866 | The picture size is too large. |
| NET_DVR_PIC_ANALYSIS_TARGRT_NUM_EXCEED_ERROR | 867 | Too many targets in the picture. |
| NET_DVR_ANALYSIS_ENGINES_LOADING_ERROR | 868 | Initializing analysis engine. |
| NET_DVR_ANALYSIS_ENGINES_ABNORMA_ERROR | 869 | Analysis engine exception. |
| NET_DVR_ANALYSIS_ENGINES_FACELIB_IMPORTING | 870 | Analysis engine is importing pictures to face picture library. |
| NET_DVR_NO_DATA_FOR_MODELING_ERROR | 871 | No data for modeling. |
| NET_DVR_FACE_DATA_MODELING_ERROR | 872 | Device is modeling picture. Concurrent processing is not supported. |
| NET_ERR_FACELIBDATA_OVERLIMIT | 873 | No more face picture can be added to the device (the data of imported face picture library) |
| NET_DVR_ANALYSIS_ENGINES_ASSOCIATED_CHANNEL | 874 | Channel is linked to the analysis engine. |
| NET_DVR_ERR_CUSTOMID_LEN | 875 | The minimum length of upper layer custom ID is 32 bytes. |
| NET_DVR_ERR_CUSTOMFACELIBID_REPEAT | 876 | The applied custom face picture library ID is duplicated |
| NET_DVR_ERR_CUSTOMHUMANID_REPEAT | 877 | The applied custom person ID is duplicated. |
| NET_DVR_ERR_URL_DOWNLOAD_FAIL | 878 | URL download failed. |
| NET_DVR_ERR_URL_DOWNLOAD_NOTSTART | 879 | URL download has not started. |

| Error Name | Error Code | Error Description |
|---|------------|---|
| NET_DVR_CFG_FILE_SECRETKEY_ERROR | 880 | The security verification key of configuration file is error. |
| NET_DVR_THERMOMETRY_REGION_OVERSTEP_ERROR | 883 | Invalid thermometry region |
| NET_DVR_ERR_TOO_SHORT_CALIBRATING_TIME | 894 | Too short time for calibration. |
| NET_DVR_ERR_AUTO_CALIBRATE_FAILED | 895 | Auto calibration failed. |
| NET_DVR_ERR_VERIFICATION_FAILED | 896 | Verification failed. |
| NET_DVR_NO_TEMP_SENSOR_ERROR | 897 | No temperature sensor. |
| NET_DVR_PUPIL_DISTANCE_OVERSIZE_ERROR | 898 | The pupil distance is too large. |
| NET_ERR_WINCHAN_IDX | 901 | Window channel index error. |
| NET_ERR_WIN_LAYER | 902 | Window layer number error(the count of window layers on a single screen exceeds the max number). |
| NET_ERR_WIN_BLK_NUM | 903 | Window block number error(the count of screens that single window overlays exceeds the max number). |
| NET_ERR_OUTPUT_RESOLUTION | 904 | The output resolution error. |
| NET_ERR_LAYOUT | 905 | Layout index error. |
| NET_ERR_INPUT_RESOLUTION | 906 | The input resolution is not supported. |
| NET_ERR_SUBDEVICE_OFFLINE | 907 | The sub-device is off-line. |
| NET_ERR_NO_DECODE_CHAN | 908 | There is no free decoding channel. |
| NET_ERR_MAX_WINDOW_ABILITY | 909 | The upper limit of window number. |
| NET_ERR_ORDER_ERROR | 910 | Calling order error. |
| NET_ERR_PLAYING_PLAN | 911 | Be playing plan. |
| NET_ERR_DECODER_USED | 912 | Decoder board is being used. |
| NET_ERR_OUTPUT_BOARD_DATA_OVERFLOW | 913 | Output board data overflow |
| NET_ERR_SAME_USER_NAME | 914 | Duplicate user name |

| Error Name | Error Code | Error Description |
|-------------------------------------|------------|--|
| NET_ERR_INVALID_USER_NAME | 915 | Invalid user name |
| NET_ERR_MATRIX_USING | 916 | Input matrix is in use. |
| NET_ERR_DIFFERENT_CHAN_TYPE | 917 | Different channel type (the type of matrix output channel mismatches that of the controller input channel) |
| NET_ERR_INPUT_CHAN_BINDED | 918 | Input channel has been bound by other matrix |
| NET_ERR_BINDED_OUTPUT_CHAN_OVERFLOW | 919 | The matrix output channels in use exceeded the number bound by matrix and controller |
| NET_ERR_MAX_SIGNAL_NUM | 920 | Number of input signals reached upper limit |
| NET_ERR_INPUT_CHAN_USING | 921 | Input channel is in use |
| NET_ERR_MANAGER_LOGON | 922 | Administrator has logged in, operation failed |
| NET_ERR_USERALREADY_LOGON | 923 | The user has logged in, operation failed |
| NET_ERR_LAYOUT_INIT | 924 | Scene is initializing, operation failed |
| NET_ERR_BASEMAP_SIZE_NOT_MATCH | 925 | Base image size does not match |
| NET_ERR_WINDOW_OPERATING | 926 | Window is in other operation, operation failed |
| NET_ERR_SIGNAL_UPLIMIT | 927 | Number of signal source window reached upper limit |
| NET_ERR_WINDOW_SIZE_OVERLIMIT | 943 | The window size exceeds the limit. |
| NET_ERR_MAX_WIN_OVERLAP | 951 | The number of windows overlap has reached the maximum limit. |
| NET_ERR_STREAMID_CHAN_BOTH_VALID | 952 | stream ID and channel number are both valid. |
| NET_ERR_NO_ZERO_CHAN | 953 | The device has no zero channel. |
| NEED_RECONNECT | 955 | Need redirection (for transcoding system) |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_ERR_NO_STREAM_ID | 956 | The stream ID does not exist. |
| NET_DVR_TRANS_NOT_START | 957 | The transcoding has not been started. |
| NET_ERR_MAXNUM_STREAM_ID | 958 | The number of stream ID has reached the maximum limit. |
| NET_ERR_WORKMODE_MISMATCH | 959 | The work mode does not match with the requirement. |
| NET_ERR_MODE_IS_USING | 960 | It Has been working in current mode. |
| NET_ERR_DEV_PROGRESSIONG | 961 | The device is in processing |
| NET_ERR_PASSIVE_TRANSCODING | 962 | It is in transcoding. |
| NET_DVR_ERR_WINDOW_SIZE_PLACE | 975 | Wrong window position. |
| NET_DVR_ERR_RGIONAL_RESTRICTIONS | 976 | Screen distance exceeds the limit. |
| NET_DVR_ERR_CLOSE_WINDOWS | 984 | Operation failed. Close the window first. |
| NET_DVR_ERR_MATRIX_LOOP_ABILITY | 985 | Beyond the cycle decoding capacity. |
| NET_DVR_ERR_MATRIX_LOOP_TIME | 986 | Invalid cycle decoding time. |
| NET_DVR_ERR_LINKED_OUT_ABILITY | 987 | No more linked camera can be added. |
| NET_ERR_RESOLUTION_NOT_SUPPORT_ODD_VOUT | 990 | The resolution is not supported (odd No.). |
| NET_ERR_RESOLUTION_NOT_SUPPORT_EVEN_VOUT | 991 | The resolution is not supported (even No.). |
| NET_ERR_UnitConfig_Failed | 998 | Unit configuration failed. |
| XML_ABILITY_NOTSUPPORT | 1000 | Getting capability node is not supported |
| XML_ANALYZE_NOENOUGH_BUF | 1001 | Not enough output memory |
| XML_ANALYZE_FIND_LOCALXML_ERROR | 1002 | Failed to find related local xml |
| XML_ANALYZE_LOAD_LOCALXML_ERROR | 1003 | Loading local xml error |

| Error Name | Error Code | Error Description |
|---|------------|---|
| XML_NANLYZE_DVR_DATA_FORMAT_ERROR | 1004 | Device capability data format error |
| XML_ANALYZE_TYPE_ERROR | 1005 | Capability set type error |
| XML_ANALYZE_XML_NODE_ERROR | 1006 | XML capability node format error |
| XML_INPUT_PARAM_ERROR | 1007 | Input capability XML node value error |
| XML_VERSION_MISMATCH | 1008 | XML version does not match |
| NET_ERR_TRANS_CHAN_START | 1101 | Transparent channel has been open, operation failed |
| NET_ERR_DEV_UPGRADING | 1102 | Device is upgrading |
| NET_ERR_MISMATCH_UPGRADE_PACK_TYPE | 1103 | Upgrade pack type does not match |
| NET_ERR_DEV_FORMATTING | 1104 | Device is formatting |
| NET_ERR_MISMATCH_UPGRADE_PACK_VERSION | 1105 | Upgrade pack version does not match |
| NET_ERR_PT_LOCKED | 1106 | PT is locked. |
| NET_DVR_ERR_ILLEGAL_VERIFICATION_CODE | 1111 | Illegal verification code. Change the verification code. |
| NET_DVR_ERR_LACK_VERIFICATION_CODE | 1112 | No verification code. Enter the verification code. |
| NET_DVR_ERR_FORBIDDEN_IP | 1113 | The IP address cannot be configured. |
| NET_DVR_ERR_HTTP_BKN_EXCEED_ONE | 1125 | Up to one channel's ANR function can be enabled. |
| NET_DVR_ERR_FORMATTING_FAILED | 1131 | Formatting HDD failed. |
| NET_DVR_ERR_ENCRYPTED_FORMATTING_FAILED | 1132 | Formatting encrypted HDD failed. |
| NET_DVR_ERR_WRONG_PASSWORD | 1133 | Verifying password of SD card failed. Incorrect password. |
| NET_ERR_SEARCHING_MODULE | 1201 | Searching peripherals. |
| NET_ERR_REGISTERING_MODULE | 1202 | Registering external module |
| NET_ERR_GETTING_ZONES | 1203 | Getting arming region parameter |
| NET_ERR_GETTING_TRIGGERS | 1204 | Getting trigger |

| Error Name | Error Code | Error Description |
|---|------------|--|
| NET_ERR_ARMED_STATUS | 1205 | System is in arming status |
| NET_ERR_PROGRAM_MODE_STATUS | 1206 | System is in programming mode |
| NET_ERR_WALK_TEST_MODE_STATUS | 1207 | System is in pacing measuring mode |
| NET_ERR_BYPASS_STATUS | 1208 | Bypass status |
| NET_ERR_DISABLED_MODULE_STATUS | 1209 | Function not enabled |
| NET_ERR_NOT_SUPPORT_OPERATE_ZONE | 1210 | Operation is not supported by arming region |
| NET_ERR_NOT_SUPPORT_MOD_MODULE_ADDR | 1211 | Module address cannot be modified |
| NET_ERR_UNREGISTERED_MODULE | 1212 | Module is not registered |
| NET_ERR_PUBLIC_SUBSYSTEM_ASSOCIATE_SELF | 1213 | Public sub system associate with its self |
| NET_ERR_EXCEEDS_ASSOCIATE_SUBSYSTEM_NUM | 1214 | Number of associated public sub system reached upper limit |
| NET_ERR_BE_ASSOCIATED_BY_PUBLIC_SUBSYSTEM | 1215 | Sub system is associated by other public sub system |
| NET_ERR_ZONE_FAULT_STATUS | 1216 | Arming region is in failure status |
| NET_ERR_SAME_EVENT_TYPE | 1217 | Same event type exists in enable event trigger alarm output and disable event trigger alarm output |
| NET_ERR_ZONE_ALARM_STATUS | 1218 | Arming region is in alarm status |
| NET_ERR_EXPANSION_BUS_SHORT_CIRCUIT | 1219 | Extension bus short-circuit |
| NET_ERR_PWD_CONFLICT | 1220 | Password conflict, e.g., lock password is identical with duress password |
| NET_ERR_DETECTOR_GISTERED_BY_OTHER_ZONE | 1221 | Detector has been registered by other arming regions |
| NET_ERR_DETECTOR_GISTERED_BY_OTHER_PU | 1222 | Detector has been registered by other hosts |
| NET_ERR_DETECTOR_DISCONNECT | 1223 | Detector offline |
| NET_ERR_CALL_BUSY | 1224 | Device in call |

| Error Name | Error Code | Error Description |
|--------------------------------------|------------|--|
| NET_ERR_FILE_NAME | 1357 | File name error, empty or invalid |
| NET_ERR_BROADCAST_BUSY | 1358 | Device in broadcast |
| NET_DVR_ERR_LANENUM_EXCEED | 1400 | Over the number of lanes. |
| NET_DVR_ERR_PRAREA_EXCEED | 1401 | Recognition area is too large. |
| NET_DVR_ERR_LIGHT_PARAM | 1402 | Signal lamp access parameters error. |
| NET_DVR_ERR_LANE_LINE_INVALID | 1403 | Lane configuration error. |
| NET_DVR_ERR_STOP_LINE_INVALID | 1404 | Stop line configuration error. |
| NET_DVR_ERR_LEFTORRIGHT_LINE_INVALID | 1405 | Turn left / right boundary configuration error. |
| NET_DVR_ERR_LANE_NO_REPEAT | 1406 | Overlay lane number repetition. |
| NET_DVR_ERR_PRAREA_INVALID | 1407 | The polygon does not meet the requirements. |
| NET_DVR_ERR_LIGHT_NUM_EXCEED | 1408 | Video detection of traffic light signal exceeds the maximum number of. |
| NET_DVR_ERR_SUBLIGHT_NUM_INVALID | 1409 | Video detection of traffic signal lamp lights are not legitimate |
| NET_DVR_ERR_LIGHT_AREASIZE_INVALID | 1410 | The size of the video detection of traffic light input signal lamp is not valid. |
| NET_DVR_ERR_LIGHT_COLOR_INVALID | 1411 | The color of the video detection of traffic light input signal lamp color is not legitimate. |
| NET_DVR_ERR_LIGHT_DIRECTION_INVALID | 1412 | The direction property of the video detection of traffic light input light is not valid. |
| NET_DVR_ERR_LACK_IOABLITY | 1413 | Lack of IO ability. |
| NET_DVR_ERR_FTP_PORT | 1414 | FTP port error. |
| NET_DVR_ERR_FTP_CATALOGUE | 1415 | FTP catalogue error. |
| NET_DVR_ERR_FTP_UPLOAD_TYPE | 1416 | FTP upload type error. |
| NET_DVR_ERR_FLASH_PARAM_WRITE | 1417 | Setting param flash write error. |

| Error Name | Error Code | Error Description |
|---|------------|-------------------------------------|
| NET_DVR_ERR_FLASH_PARAM_READ | 1418 | Getting param flash read error. |
| NET_DVR_ERR_PICNAME_DELIMITER | 1419 | Pic name delimiter error. |
| NET_DVR_ERR_PICNAME_ITEM | 1420 | Pic name item error. |
| NET_DVR_ERR_PLATE_RECOGNIZE_TYPE | 1421 | Plate recognize type error. |
| NET_DVR_ERR_CAPTURE_TIMES | 1422 | Capture times error. |
| NET_DVR_ERR_LOOP_DISTANCE | 1423 | Loop distance error. |
| NET_DVR_ERR_LOOP_INPUT_STATUS | 1424 | Loop input status error. |
| NET_DVR_ERR_RELATE_IO_CONFLICT | 1425 | Related IO conflict. |
| NET_DVR_ERR_INTERVAL_TIME | 1426 | Interval time error. |
| NET_DVR_ERR_SIGN_SPEED | 1427 | Sign speed error. |
| NET_DVR_ERR_PIC_FLIP | 1428 | Flip is used. |
| NET_DVR_ERR_RELATE_LANE_NUMBER | 1429 | Related lane number error. |
| NET_DVR_ERR_TRIGGER_MODE | 1430 | Trigger mode error. |
| NET_DVR_ERR_DELAY_TIME | 1431 | Delay time error. |
| NET_DVR_ERR_EXCEED_RS485_COUNT | 1432 | Exceed RS485 count. |
| NET_DVR_ERR_RADAR_TYPE | 1433 | Radar type error. |
| NET_DVR_ERR_RADAR_ANGLE | 1434 | Radar angle error. |
| NET_DVR_ERR_RADAR_SPEED_VALID_TIME | 1435 | Radar speed valid time error. |
| NET_DVR_ERR_RADAR_LINE_CORRECT | 1436 | Radar line correct error. |
| NET_DVR_ERR_RADAR_CONST_CORRECT | 1437 | Radar const correct error. |
| NET_DVR_ERR_RECORD_PARAM | 1438 | Record param error. |
| NET_DVR_ERR_LIGHT_WITHOUT_COLOR_AND_DIRECTION | 1439 | Light number and other param error. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_ERR_LIGHT_WITHOUT_DETECTION_REGION | 1440 | Light number and detection region error. |
| NET_DVR_ERR_RECOGNIZE_PROVINCE_PARAM | 1441 | Plate recognize Province param error. |
| NET_DVR_ERR_SPEED_TIMEOUT | 1442 | IO Speed TimeOut Param error. |
| NET_DVR_ERR_NTP_TIMEZONE | 1443 | NTP TimeZone Param error. |
| NET_DVR_ERR_NTP_INTERVAL_TIME | 1444 | NTP Interval Time error. |
| NET_DVR_ERR_NETWORK_CARD_NUM | 1445 | Network Card Num error. |
| NET_DVR_ERR_DEFAULT_ROUTE | 1446 | Default Route error. |
| NET_DVR_ERR_BONDING_WORK_MODE | 1447 | Banding Work Mode error. |
| NET_DVR_ERR_SLAVE_CARD | 1448 | Sub-Card error. |
| NET_DVR_ERR_PRIMARY_CARD | 1449 | Primary Card error. |
| NET_DVR_ERR_DHCP_PPOE_WORK | 1450 | DHCP and PPOE not Meanwhile start. |
| NET_DVR_ERR_NET_INTERFACE | 1451 | Net Interface invalid. |
| NET_DVR_ERR_MTU | 1452 | Invalid MTU parameters. |
| NET_DVR_ERR_NETMASK | 1453 | Netmask address invalid. |
| NET_DVR_ERR_IP_INVALID | 1454 | IP address invalid. |
| NET_DVR_ERR_MULTICAST_IP_INVALID | 1455 | Multicast IP address invalid. |
| NET_DVR_ERR_GATEWAY_INVALID | 1456 | Gateway address invalid. |
| NET_DVR_ERR_DNS_INVALID | 1457 | DNS Param invalid. |
| NET_DVR_ERR_ALARMHOST_IP_INVALID | 1458 | AlarmHost IP invalid. |
| NET_DVR_ERR_IP_CONFLICT | 1459 | IP address Conflict. |
| NET_DVR_ERR_NETWORK_SEGMENT | 1460 | IP not support Multi Network segment. |
| NET_DVR_ERR_NETPORT | 1461 | NetPort error. |
| NET_DVR_ERR_PPPOE_NOSUPPORT | 1462 | PPPoE is not supported. |

| Error Name | Error Code | Error Description |
|---|------------|---|
| NET_DVR_ERR_DOMAINNAME_NOSUPPORT | 1463 | Not Support Domain Name. |
| NET_DVR_ERR_NO_SPEED | 1464 | Speed Not Enabled. |
| NET_DVR_ERR_IOSTATUS_INVALID | 1465 | IO Status invalid. |
| NET_DVR_ERR_BURST_INTERVAL_INVALID | 1466 | Burst Interval invalid. |
| NET_DVR_ERR_RESERVE_MODE | 1467 | Reserve Mode invalid. |
| NET_DVR_ERR_LANE_NO | 1468 | Lane No error. |
| NET_DVR_ERR_COIL_AREA_TYPE | 1469 | Coil Area Type error. |
| NET_DVR_ERR_TRIGGER_AREA_PARAM | 1470 | Trigger Area Param error. |
| NET_DVR_ERR_SPEED_LIMIT_PARAM | 1471 | Speed Limit Param error. |
| NET_DVR_ERR_LANE_PROTOCOL_TYPE | 1472 | Lane Protocol Type error. |
| NET_DVR_ERR_INTERVAL_TYPE | 1473 | Capture Interval Type error. |
| NET_DVR_ERR_INTERVAL_DISTANCE | 1474 | Capture Interval Distance error. |
| NET_DVR_ERR_RS485_ASSOCIATE_DEVTYPE | 1475 | Rs485 Associate DevType error. |
| NET_DVR_ERR_RS485_ASSOCIATE_LANENO | 1476 | Rs485 Associate LaneNo error. |
| NET_DVR_ERR_LANENO_ASSOCIATE_MULTIRS485 | 1477 | LaneNo Associate MulitRs485 error. |
| NET_DVR_ERR_LIGHT_DETECTION_REGION | 1478 | Light Detection Region error. |
| NET_DVR_ERR_DN2D_NOSUPPORT | 1479 | UnSupport Capture Frame 2D Noise Reduction. |
| NET_DVR_ERR_IRISMODE_NOSUPPORT | 1480 | UnSupport scene Mode. |
| NET_DVR_ERR_WB_NOSUPPORT | 1481 | UnSupport White Balance Mode. |
| NET_DVR_ERR_IO_EFFECTIVENESS | 1482 | IO Effectiveness invalid. |

| Error Name | Error Code | Error Description |
|---|------------|---|
| NET_DVR_ERR_LIGHTNO_MAX | 1483 | Access Detector Lights Red / Yellow Overrun. |
| NET_DVR_ERR_LIGHTNO_CONFLICT | 1484 | Access Detector Lights Red / Yellow Conflict. |
| NET_DVR_ERR_CANCEL_LINE | 1485 | Trigger straight line error. |
| NET_DVR_ERR_STOP_LINE | 1486 | Subject line area stop line error. |
| NET_DVR_ERR_RUSH_REDLIGHT_LINE | 1487 | Red light trigger lines error. |
| NET_DVR_ERR_IOOUTNO_MAX | 1488 | IO out port error. |
| NET_DVR_ERR_IOOUTNO_AHEADTIME_MAX | 1489 | IO out ahead time error. |
| NET_DVR_ERR_IOOUTNO_IOWORKTIME | 1490 | IO out inwork time error. |
| NET_DVR_ERR_IOOUTNO_FREQMULTI | 1491 | IO out frequency multiplication error. |
| NET_DVR_ERR_IOOUTNO_DUTYRATE | 1492 | IO out duty rate error. |
| NET_DVR_ERR_VIDEO_WITH_EXPOSURE | 1493 | IO out work mode error. |
| NET_DVR_ERR_PLATE_BRIGHTNESS_WITHOUT_FLASHDET | 1494 | Plate enable in plate compensate mode on. |
| NET_DVR_ERR_RECOGNIZE_TYPE_PARAM | 1495 | Recognize Type error. |
| NET_DVR_ERR_PALTE_RECOGNIZE_AREA_PARAM | 1496 | Plate Recognize Area Param error. |
| NET_DVR_ERR_PORT_CONFLICT | 1497 | Port Conflict. |
| NET_DVR_ERR_LOOP_IP | 1498 | IP cannot be the loopback address. |
| NET_DVR_ERR_DRIVELINE_SENSITIVE | 1499 | Driveline sensitivity error. |
| NET_ERR_VQD_TIME_CONFLICT | 1500 | The time period conflict. |
| NET_ERR_VQD_PLAN_NO_EXIST | 1501 | The diagnostic plan of VQD dese not exist. |
| NET_ERR_VQD_CHAN_NO_EXIST | 1502 | The channel dese not exist. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_ERR_VQD_CHAN_MAX | 1503 | The total number of VQD plans exceeds the max limit. |
| NET_ERR_VQD_TASK_MAX | 1504 | The total number of VQD tasks exceeds the max limit. |
| NET_DVR_ERR_EXCEED_MAX_CAPTURE_TIMES | 1600 | Capture times exceed 2 in flash mode. |
| NET_DVR_ERR_RADAR_TYPE_CONFLICT | 1601 | Radar type conflict. |
| NET_DVR_ERR_LICENSE_PLATE_NULL | 1602 | The license plate is null. |
| NET_DVR_ERR_WRITE_DATABASE | 1603 | Failed to write data into the database. |
| NET_DVR_ERR_LICENSE_EFFECTIVE_TIME | 1604 | The effective time of license plate error. |
| NET_DVR_ERR_PRERECORDED_STARTTIME_LONG | 1605 | The pre recorded start time is greater than the number of illegal capture. |
| NET_DVR_ERR_TRIGGER_RULE_LINE | 1606 | Trigger rule line error. |
| NET_DVR_ERR_LEFTRIGHT_TRIGGERLINE_NOTVERTICAL | 1607 | Left and right trigger line is not vertical. |
| NET_DVR_ERR_FLASH_LAMP_MODE | 1608 | Flash lamp mode error. |
| NET_DVR_ERR_ILLEGAL_SNAPSHOT_NUM | 1609 | Illegal capture number error. |
| NET_DVR_ERR_ILLEGAL_DETECTION_TYPE | 1610 | Illegal detection type error. |
| NET_DVR_ERR_POSITIVEBACK_TRIGGERLINE_HIGH | 1611 | Positive back to trigger line height error. |
| NET_DVR_ERR_MIXEDMODE_CAPTYPE_ALLTARGETS | 1612 | Mixed mode only supports capture type all targets. |
| NET_DVR_ERR_CARSIGNSPEED_GREATERTHAN_LIMITSPEED | 1613 | Car sign speed greater than speed limit value. |
| NET_DVR_ERR_BIGCARSIGNSPEED_GREATERTHAN_LIMITSPEED | 1614 | Big car sign speed limit greater than speed limit value. |
| NET_DVR_ERR_BIGCARSIGNSPEED_GREATERTHAN_CARSIGNSPEED | 1615 | Big car sign speed limit is greater than the car sign speed limit value. |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_ERR_BIGCARLIMITSPEED_GREATERTHAN_CARLIMITSPEED | 1616 | Big car speed limit value is greater than the car speed limit value. |
| NET_DVR_ERR_BIGCARLOWSPEEDLIMIT_GREATERTHAN_CARLOWSPEEDLIMIT | 1617 | Big car low speed limit value is greater than the car low speed limit value. |
| NET_DVR_ERR_CARLIMITSPEED_GREATERTHAN_EXCEPHIGHSPEED | 1618 | Car speed limit greater than exception high speed value. |
| NET_DVR_ERR_BIGCARLIMITSPEED_GREATERTHAN_EXCEPHIGHSPEED | 1619 | Big car speed limit greater than exception high speed value. |
| NET_DVR_ERR_STOPLINE_MORETHAN_TRIGGERLINE | 1620 | Stopping more than straight lines trigger lines. |
| NET_ERR_TIME_OVERLAP | 1900 | Time periods overlap |
| NET_ERR_HOLIDAY_PLAN_OVERLAP | 1901 | Holiday plan overlap |
| NET_ERR_CARDNO_NOT_SORT | 1902 | Card number is not sorted |
| NET_ERR_CARDNO_NOT_EXIST | 1903 | Card number does not exist |
| NET_ERR_ILLEGAL_CARDNO | 1904 | Card number error |
| NET_ERR_ZONE_ALARM | 1905 | Arming region is in arming status (parameter cannot be modified) |
| NET_ERR_ZONE_OPERATION_NOT_SUPPORT | 1906 | Arming region does not support the operation |
| NET_ERR_INTERLOCK_ANTI_CONFLICT | 1907 | Interlock and anti-passback configuration conflict |
| NET_ERR_DEVICE_CARD_FULL | 1908 | Card full (return after card reached 10,000) |
| NET_ERR_HOLIDAY_GROUP_DOWNLOAD | 1909 | Failed to download holiday group |
| NET_ERR_LOCAL_CONTROL_OFF | 1910 | Distributed access controller offline |
| NET_ERR_LOCAL_CONTROL_DISADD | 1911 | Distributed access controller is not added |
| NET_ERR_LOCAL_CONTROL_HASADD | 1912 | Distributed access controller is added |
| NET_ERR_LOCAL_CONTROL_DOORNO_CONFLICT | 1913 | Conflict with added distributed access controller |

| Error Name | Error Code | Error Description |
|---|------------|---|
| NET_ERR_LOCAL_CONTROL_COMMUNICATION_FAIL | 1914 | Distributed access controller communication failed |
| NET_ERR_OPERAND_INEXISTENCE | 1915 | Operation object does not exist (operation to door, alarm output, alarm input, return when the object is not added) |
| NET_ERR_LOCAL_CONTROL_OVER_LIMIT | 1916 | Distributed access controller exceeded device capability upper limit |
| NET_ERR_DOOR_OVER_LIMIT | 1917 | Door exceeded device capability upper limit |
| NET_ERR_ALARM_OVER_LIMIT | 1918 | Alarm input and output exceeded device capability upper limit |
| NET_ERR_LOCAL_CONTROL_ADDRESS_INCONFORMITY_TYPE | 1919 | Distributed access controller address does not match with type |
| NET_ERR_NOT_SUPPORT_ONE_MORE_CARD | 1920 | not support one person multi-card |
| NET_ERR_DELETE_NO_EXISTENCE_FACE | 1921 | The face picture does not exist. |
| NET_ERR_DOOR_SPECIAL_PASSWORD_REPEAT | 1922 | Repeated door door duress code, the super password, or the dismiss code. |
| NET_ERR_AUTH_CODE_REPEAT | 1923 | Repeated device authentication code |
| NET_ERR_DEPLOY_EXCEED_MAX | 1924 | No more devices can be armed. |
| NET_ERR_NOT_SUPPORT_DEL_FP_BY_ID | 1925 | The fingerprint module does not support deleting fingerprint by finger ID. |
| NET_ERR_TIME_RANGE | 1926 | Invalid range of the effective period. |
| NET_ERR_CAPTURE_TIMEOUT | 1927 | Collection timed out. |
| NET_ERR_LOW_SCORE | 1928 | Low quality of collected data. |
| NET_ERR_OFFLINE_CAPTURING | 1929 | The device is collecting data offline and cannot respond. |
| NET_DVR_ERR_OUTDOOR_COMMUNICATION | 1950 | Communication exception with outdoor terminal |

| Error Name | Error Code | Error Description |
|---|------------|---|
| NET_DVR_ERR_ROOMNO_UNDEFINED | 1951 | Room number is not set |
| NET_DVR_ERR_NO_CALLING | 1952 | No call |
| NET_DVR_ERR_RINGING | 1953 | Ringing |
| NET_DVR_ERR_IS_CALLING_NOW | 1954 | Call in progress |
| NET_DVR_ERR_LOCK_PASSWORD_WRONG | 1955 | Incorrect smart lock password |
| NET_DVR_ERR_CONTROL_LOCK_FAILURE | 1956 | Lock control failure |
| NET_DVR_ERR_CONTROL_LOCK_OVERTIME | 1957 | Lock control timed out |
| NET_DVR_ERR_LOCK_DEVICE_BUSY | 1958 | Smart lock device busy |
| NET_DVR_ERR_UNOPEN_REMOTE_LOCK_FUNCTION | 1959 | Remote lock control not enabled |
| NET_DVR_ERR_FILE_NOT_COMPLETE | 2100 | Downloaded file is incomplete |
| NET_DVR_ERR_IPC_EXIST | 2101 | The camera already exists |
| NET_DVR_ERR_ADD_IPC | 2102 | Camera has been added to the channel |
| NET_DVR_ERR_OUT_OF_RES | 2103 | Not enough network bandwidth |
| NET_DVR_ERR_CONFLICT_TO_LOCALIP | 2104 | IP address of camera conflicts with that of DVR |
| NET_DVR_ERR_IP_SET | 2105 | Invalid IP address |
| NET_DVR_ERR_PORT_SET | 2106 | Invalid port number |
| NET_ERR_WAN_NOTSUPPORT | 2107 | Not in the same LAN, cannot set security question or export GUID file |
| NET_ERR_MUTEX_FUNCTION | 2108 | Mutually exclusive function |
| NET_ERR_QUESTION_CONFIGNUM | 2109 | Error in number of security question configurations |
| NET_ERR_FACECHAN_NORESOURCE | 2110 | All the face VCA channels are occupied. |
| NET_ERR_DATA_CALLBACK | 2111 | Data is calling back. |

| Error Name | Error Code | Error Description |
|---|------------|--|
| NET_ERR_ATM_VCA_CHAN_IS_RELATED | 2112 | The VCA channel is already linked. |
| NET_ERR_ATM_VCA_CHAN_IS_OVERLAPED | 2113 | The VCA channel is already overlaid. |
| NET_ERR_FACE_CHAN_UNOVERLAP_EACH_OTHER | 2114 | The face channels cannot be overlaid. |
| NET_DVR_SMD_ENCODING_NORESOURCE | 2116 | Insufficient SMD encoding resource |
| NET_DVR_SMD_DECODING_NORESOURCE | 2117 | Insufficient SMD decoding resource |
| NET_DVR_FACELIB_DATA_PROCESSING | 2118 | Face picture library data is in processing |
| NET_DVR_ERR_LARGE_TIME_DIFFERENCE | 2119 | There is a great time difference between device and server. |
| NET_DVR_NO_SUPPORT_WITH_PLAYBACK | 2120 | It is not supported. Playback is enabled. |
| NET_DVR_CHANNEL_NO_SUPPORT_WITH_SMD | 2121 | It is not supported. SMD of channel is enabled. |
| NET_DVR_CHANNEL_NO_SUPPORT_WITH_FD | 2122 | It is not supported. Face capture of channel is enabled. |
| NET_DVR_ILLEGAL_PHONE_NUMBER | 2123 | Invalid telephone number |
| NET_DVR_ILLEGAL_CERTIFICATE_NUMBER | 2124 | Invalid ID No. |
| NET_DVR_ERR_CHANNEL_RESOLUTION_NO_SUPPORT | 2125 | The channel resolution is not supported |
| NET_DVR_ERR_CHANNEL_COMPRESSION_NO_SUPPORT | 2126 | The channel encoding format is not supported |
| NET_DVR_ERR_CLUSTER_DEVICE_TOO_LESS | 2127 | Deleting is not allowed. The number of devices is not enough |
| NET_DVR_ERR_CLUSTER_DEL_DEVICE_CM_PLAYLOAD | 2128 | Deleting is not allowed. The device is cluster host. |
| NET_DVR_ERR_CLUSTER_DEVNUM_OVER_UPPER_LIMIT | 2129 | No more devices can be added. |

| Error Name | Error Code | Error Description |
|--|------------|---|
| NET_DVR_ERR_CLUSTER_DEVICE_TYPE_INCONFORMITY | 2130 | Device type mismatched. |
| NET_DVR_ERR_CLUSTER_DEVICE_VERSION_INCONFORMITY | 2131 | Device version mismatched. |
| NET_DVR_ERR_CLUSTER_IP_CONFLICT | 2132 | Cluster system IP address conflict: ipv4 address conflict, invalid ipv6. |
| NET_DVR_ERR_CLUSTER_IP_INVALID | 2133 | Invalid cluster system IP address: invalid ipv4, invalid ipv6. |
| NET_DVR_ERR_CLUSTER_PORT_CONFLICT | 2134 | Cluster system port conflict |
| NET_DVR_ERR_CLUSTER_PORT_INVALID | 2135 | Invalid cluster system port |
| NET_DVR_ERR_CLUSTER_USERNAEM_OR_PASSWORD_INVALID | 2136 | Invalid user name or password |
| NET_DVR_ERR_CLUSTER_DEVICE_ALREADY_EXIST | 2137 | The device already exists. |
| NET_DVR_ERR_CLUSTER_DEVICE_NOT_EXIST | 2138 | The device does not exist. |
| NET_DVR_ERR_CLUSTER_NON_CLUSTER_MODE | 2139 | The device working mode is not the cluster mode . |
| NET_DVR_ERR_CLUSTER_IP_NOT_SAME_LAN | 2140 | IP addresses are in different LAN. Building cluster or extending capacity for NVRs in different LAN is not allowed. |
| NET_DVR_ERR_IDENTITY_KEY | 2147 | Incorrect interaction password |
| NET_DVR_MISSING_IDENTITY_KEY | 2148 | Interaction password is missing |
| NET_DVR_ERR_CAPTURE_PACKAGE_FAILED | 2141 | Capturing packets failed. |
| NET_DVR_ERR_CAPTURE_PACKAGE_PROCESSING | 2142 | Capturing packet. |
| NET_DVR_ERR_SAFETY_HELMET_NO_RESOURCE | 2143 | No enough hard hat detection resource. |

| Error Name | Error Code | Error Description |
|---|------------|--|
| NET_DVR_NO_SUPPORT_WITH_ABSTRACT | 2144 | This function is not supported. Video synopsis is already enabled. |
| NET_DVR_INSUFFICIENT_DEEP_LEARNING_RESOURCES | 2146 | No more deep learning resources can be added. |
| NET_DVR_NO_SUPPORT_WITH_PERSON_DENSITY_DETECT | 2149 | People gathering density is enabled, it is not supported |
| NET_DVR_IPC_RESOLUTION_OVERFLOW | 2150 | The network camera resolution is too large |
| NET_DVR_IPC_BITRATE_OVERFLOW | 2151 | The network camera bitrate is too large |
| NET_DVR_ERR_INVALID_TASKID | 2152 | Invalid taskID |
| NET_DVR_PANEL_MODE_NOT_CONFIG | 2153 | The ATM panel mode is not configured. |
| NET_DVR_NO_HUMAN_ENGINES_RESOURCE | 2154 | No enough engine resource |
| NET_DVR_ERR_TASK_NUMBER_OVERFLOW | 2155 | No more task data is allowed |
| NET_DVR_ERR_COLLISION_TIME_OVERFLOW | 2156 | Collision time is over the limit |
| NET_DVR_ERR_EVENT_NOTSUPPORT | 2159 | Subscribing alarm/event is not supported. |
| NET_DVR_IPC_NUM_REACHES_LIMIT | 2184 | The max. number of network camera channels reached. |
| NET_DVR_IOT_NUM_REACHES_LIMIT | 2185 | The max. number of IoT channels reached |
| NET_DVR_IOT_CHANNEL_DEVICE_EXIST | 2186 | Device of the IoT channel already exists. |
| NET_DVR_IOT_CHANNEL_DEVICE_NOT_EXIST | 2187 | Device of the IoT channel does not exist. |
| NET_DVR_INVALID_IOT_PROTOCOL_TYPE | 2188 | Invalid IoT protocol type |
| NET_DVR_INVALID_EZVIZ_SECRET_KEY | 2189 | Invalid verification code |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_DUPLICATE_IOT_DEVICE | 2190 | Duplicated IoT device |
| NET_DVR_ERROR_NEED_DOUBLE_VERIFICATION | 2206 | Double verification is required |
| NET_DVR_NO_DOUBLE_VERIFICATION_USER | 2207 | No double verification user |
| NET_DVR_TIMESPAN_NUM_OVER_LIMIT | 2209 | Max. number of time buckets reached |
| NET_DVR_CHANNEL_NUM_OVER_LIMIT | 2210 | Max. number of channels reached |
| NET_DVR_NO_SEARCH_ID_RESOURCE | 2211 | Insufficient searchID resources |
| NET_DVR_SWITCH_TIMEDIFF_LESS_LIMIT | 2249 | Time difference between power on and off should be less than 10 minutes. |
| NET_DVR_NO_SUPPORT_DELETE_STRANGER_LIB | 2262 | Deleting stranger library is not supported |
| NET_DVR_NO_SUPPORT_CREATE_STRANGER_LIB | 2263 | Creating stranger library is not supported |
| NET_DVR_SSD_FILE_SYSTEM_ERROR | 2266 | SSD file system error |
| NET_DVR_INSUFFICIENT_SSD__FOR_FPD | 2267 | Insufficient SSD space for person frequency detection |
| NET_DVR_SMRDISK_NOT_SUPPORT_RAID | 2269 | SMR disk does not support RAID. |
| NET_DVR_ERR_NOTSUPPORT_DEICING | 3001 | Device does not support deicing function under current status.(Deicing function is only supported under the power status of POE+, AC24V, and DC12V). |
| NET_DVR_ERR_THERMENABLE_CLOSE | 3002 | Temperature measurement function is not enabled. (The enable function in NET_DVR_THERMOMETRY_BASICPARAM is not turned on) |
| NET_DVR_ERR_PANORAMIC_LIMIT_OPERATED | 3004 | Panoramic map and limit cannot be operated at same time |

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_DVR_ERR_SMARTH264_ROI_OPERATED | 3005 | SmartH264 and ROI cannot be enabled at the same time. |
| NET_DVR_ERR_RULENUM_LIMIT | 3006 | No more rules can be added. |
| NET_DVR_ERR_LASER_DEICING_OPERATED | 3007 | Laser and deicing function cannot be enabled at the same time. |
| NET_DVR_ERR_OFFDIGITALZOOM_OR_MINZOOMLIMIT | 3008 | Please disable the digital zoom function or set the zoom limit to the minimum value. Otherwise, when enabling smoke and fire detection, behavior analysis, ship detection, defective point correction, temperature measurement, smoke and fire shielding function, this error code will be prompted. |
| NET_DVR_SYNCHRONIZEFOV_ERROR | 3010 | Field of view synchronization failed. |
| NET_DVR_RULE_SHIELDMASK_CONFLICT_ERROR | 3013 | The rule region conflicts with the shielded area. |
| NET_DVR_ERR_NO_SAFETY_HELMET_REGION | 3501 | The hard hat detection area is not configured. |
| NET_DVR_ERR_UNCLOSED_SAFETY_HELMET | 3502 | The hard hat detection is enabled. |
| NET_DVR_UPLOAD_HBDLIBID_ERROR | 3504 | Incorrect ID of human body picture library (incorrect HBDID or customHBDID) |

RTSP Communication Library Related Errors

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|---|
| NET_DVR_RTSP_ERROR_NOENOUGHPRI | 401 | Authentication failed: if server returns 401, it will change to this error code |
| NET_DVR_RTSP_ERROR_ALLOC_RESOURCE | 402 | Failed to allocate the resource |
| NET_DVR_RTSP_ERROR_PARAMETER | 403 | Parameter error |

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|---|
| NET_DVR_RTSP_ERROR_NO_URL | 404 | The assigned URL does not exist: when the server returns 404, SDK turns to this error code. E.g. the channel is not available, or the channel does not support sub stream |
| NET_DVR_RTSP_ERROR_FORCE_STOP | 406 | The user forces to exit midway |
| NET_DVR_RTSP_GETPORTFAILED | 407 | RTSP port getting error. |
| NET_DVR_RTSP_DESCRIBERROR | 410 | RTSP DESCRIBE communicate error |
| NET_DVR_RTSP_DESCRIBESENDDTIMEOUT | 411 | Sending "RTSP DESCRIBE" is timeout. |
| NET_DVR_RTSP_DESCRIBESENDERROR | 412 | Failed to send "RTSP DESCRIBE". |
| NET_DVR_RTSP_DESCRIBERECDTIMEOUT | 413 | Receiving "RTSP DESCRIBE" is timeout. |
| NET_DVR_RTSP_DESCRIBERECDATALOST | 414 | Receiving data of "RTSP DESCRIBE" error. |
| NET_DVR_RTSP_DESCRIBERECDERROR | 415 | Failed to receive "RTSP DESCRIBE". |
| NET_DVR_RTSP_DESCRIBESERVERERR | 416 | "RTSP DESCRIBE, the device returns the error code: 501 (failed to allocate the resource in the device) |
| NET_DVR_RTSP_SETUPERROR | 420 | (or 419), RTSP SETUP interaction error. Generally, it is that the address(URL) returned by the device is not accessible, or it is rejected by the server |
| NET_DVR_RTSP_SETUPSENDDTIMEOUT | 421 | Sending "RTSP SETUP" is timeout. |
| NET_DVR_RTSP_SETUPSENDERROR | 422 | Sending "RTSP SETUP" error. |
| NET_DVR_RTSP_SETUPRECDTIMEOUT | 423 | Receiving "RTSP SETUP" is timeout. |
| NET_DVR_RTSP_SETUPRECDATALOST | 424 | Receiving data of "RTSP SETUP" error. |
| NET_DVR_RTSP_SETUPRECDERROR | 425 | Failed to receive "RTSP SETUP". |
| NET_DVR_RTSP_OVER_MAX_CHAN | 426 | "RTSP SETUP" device returns the error that values 401 or 501. It |

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|--|
| | | exceeds the max connection number. |
| NET_DVR_RTSP_PLAYERROR | 430 | RTSP PLAY interaction error. |
| NET_DVR_RTSP_PLAYSENDTIMEOUT | 431 | Sending "RTSP PLAY" is timeout. |
| NET_DVR_RTSP_PLAYSENDERROR | 432 | Sending "RTSP PLAY" error. |
| NET_DVR_RTSP_PLAYRECVTIMEOUT | 433 | Receiving "RTSP PLAY" is timeout. |
| NET_DVR_RTSP_PLAYRECVDATALOST | 434 | Receiving data of "RTSP PLAY" error. |
| NET_DVR_RTSP_PLAYRECVERROR | 435 | Failed to receive "RTSP PLAY". |
| NET_DVR_RTSP_PLAYSERVERERR | 436 | "RTSP PLAY" device returns the error that values 401 or 501. |
| NET_DVR_RTSP_TEARDOWNERROR | 440 | RTSP TEARDOWN interaction error. |
| NET_DVR_RTSP_TEARDOWNSENDTIMEOUT | 441 | Sending "RTSP TEARDOWN" is timeout. |
| NET_DVR_RTSP_TEARDOWNSENDERROR | 442 | Sending "RTSP TEARDOWN" error. |
| NET_DVR_RTSP_TEARDOWNRECVTIMEOUT | 443 | Receiving "RTSP TEARDOWN" is timeout. |
| NET_DVR_RTSP_TEARDOWNRECVDATALOST | 444 | Receiving data of "RTSP TEARDOWN" error. |
| NET_DVR_RTSP_TEARDOWNRECVERROR | 445 | Failed to receive "RTSP TEARDOWN". |
| NET_DVR_RTSP_TEARDOWNSERVERERR | 446 | "RTSP TEARDOWN" device returns the error that values 401 or 501. |

Software Decoding Library Related Errors

| Error Name | Error Code | Error Description |
|------------------------|------------|------------------------------------|
| NET_PLAYM4_NOERROR | 500 | No error. |
| NET_PLAYM4_PARA_OVER | 501 | Input parameter is invalid. |
| NET_PLAYM4_ORDER_ERROR | 502 | API calling order error. |
| NET_PLAYM4_TIMER_ERROR | 503 | Failed to create multimedia clock. |

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|---|
| NET_PLAYM4_DEC_VIDEO_ERROR | 504 | Failed to decode video data. |
| NET_PLAYM4_DEC_AUDIO_ERROR | 505 | Failed to decode audio data. |
| NET_PLAYM4_ALLOC_MEMORY_ERROR | 506 | Failed to allocate memory. |
| NET_PLAYM4_OPEN_FILE_ERROR | 507 | Failed to open the file. |
| NET_PLAYM4_CREATE_OBJ_ERROR | 508 | Failed to create thread event. |
| NET_PLAYM4_CREATE_DDRAW_ERROR | 509 | Failed to create DirectDraw object. |
| NET_PLAYM4_CREATE_OFFSCREEN_ERROR | 510 | Failed to create backstage cache for OFFSCREEN mode. |
| NET_PLAYM4_BUF_OVER | 511 | Buffer overflow, failed to input stream. |
| NET_PLAYM4_CREATE_SOUND_ERROR | 512 | Failed to create audio equipment. |
| NET_PLAYM4_SET_VOLUME_ERROR | 513 | Failed to set the volume. |
| NET_PLAYM4_SUPPORT_FILE_ONLY | 514 | This API can be called only for file playback mode. |
| NET_PLAYM4_SUPPORT_STREAM_ONLY | 515 | This API can be called only when playing stream. |
| NET_PLAYM4_SYS_NOT_SUPPORT | 516 | Not support by the system. Decoder can only work on the system above Pentium 3. |
| NET_PLAYM4_FILEHEADER_UNKNOWN | 517 | There is no file header. |
| NET_PLAYM4_VERSION_INCORRECT | 518 | The version mismatch between decoder and encoder. |
| NET_PLAYM4_INIT_DECODER_ERROR | 519 | Failed to initialize the decoder. |
| NET_PLAYM4_CHECK_FILE_ERROR | 520 | The file is too short, or the stream data is unknown. |
| NET_PLAYM4_INIT_TIMER_ERROR | 521 | Failed to initialize multimedia clock. |
| NET_PLAYM4_BLT_ERROR | 522 | BLT failure. |

| Error Name | Error Code | Error Description |
|----------------------------------|------------|---|
| NET_PLAYM4_UPDATE_ERROR | 523 | Failed to update overlay surface |
| NET_PLAYM4_OPEN_FILE_ERROR_MULTI | 524 | Failed to open video & audio stream file. |
| NET_PLAYM4_OPEN_FILE_ERROR_VIDEO | 525 | Failed to open video stream file. |
| NET_PLAYM4_JPEG_COMPRESS_ERROR | 526 | JPEG compression error. |
| NET_PLAYM4_EXTRACT_NOT_SUPPORT | 527 | Don't support the version of this file. |
| NET_PLAYM4_EXTRACT_DATA_ERROR | 528 | Extract video data failed. |

Container Format Conversion Library Related Errors

| Error Name | Error Code | Error Description |
|-------------------------------|------------|---|
| NET_CONVERT_ERROR_NOT_SUPPORT | 581 | This container format is not supported. |

Two Way Audio Library Related Errors

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|--------------------------------------|
| NET_AUDIOINTERCOM_OK | 600 | No error. |
| NET_AUDIOINTECOM_ERR_NOTSUPPORT | 601 | Not support. |
| NET_AUDIOINTECOM_ERR_ALLOC_MEMORY | 602 | Memory allocation error. |
| NET_AUDIOINTECOM_ERR_PARAMETER | 603 | Parameter error. |
| NET_AUDIOINTECOM_ERR_CALL_ORDER | 604 | API calling order error. |
| NET_AUDIOINTECOM_ERR_FIND_DEVICE | 605 | No audio device |
| NET_AUDIOINTECOM_ERR_OPEN_DEVICE | 606 | Failed to open the audio device |
| NET_AUDIOINTECOM_ERR_NO_CONTEXT | 607 | Context error. |
| NET_AUDIOINTECOM_ERR_NO_WAVFILE | 608 | WAV file error. |
| NET_AUDIOINTECOM_ERR_INVALID_TYPE | 609 | The type of WAV parameter is invalid |

| Error Name | Error Code | Error Description |
|-----------------------------------|------------|-----------------------|
| NET_AUDIOINTECOM_ERR_ENCODE_FAIL | 610 | Failed to encode data |
| NET_AUDIOINTECOM_ERR_DECODE_FAIL | 611 | Failed to decode data |
| NET_AUDIOINTECOM_ERR_NO_PLAYBACK | 612 | Failed to play audio |
| NET_AUDIOINTECOM_ERR_DENOISE_FAIL | 613 | Failed to denoise |
| NET_AUDIOINTECOM_ERR_UNKOWN | 619 | Unknown |

QoS Stream Control Library Related Errors

| Error Name | Error Code | Error Description |
|--|------------|--|
| NET_QOS_ERR_SCHEDPARAMS_BAD_MINIMUM_INTERVAL | 678 | Incorrect predefined minimum interval. |
| NET_QOS_ERR_SCHEDPARAMS_BAD_FRACTION | 679 | Incorrect predefined score. |
| NET_QOS_ERR_SCHEDPARAMS_INVALID_BANDWIDTH | 680 | Invalid predefined bandwidth. |
| NET_QOS_ERR_PACKET_TOO_BIG | 687 | The packet size is too large. |
| NET_QOS_ERR_PACKET_LENGTH | 688 | Invalid packet size. |
| NET_QOS_ERR_PACKET_VERSION | 689 | Incorrect packet versio information. |
| NET_QOS_ERR_PACKET_UNKNOW | 690 | Unknown packet. |
| NET_QOS_ERR_OUTOFMEM | 695 | Out of memory. |
| NET_QOS_ERR_LIB_NOT_INITIALIZED | 696 | The library is not initialized. |
| NET_QOS_ERR_SESSION_NOT_FOUND | 697 | No session found. |
| NET_QOS_ERR_INVALID_ARGUMENTS | 698 | Invalid parameters. |
| NET_QOS_ERROR | 699 | QoS Stream Control Library error. |
| NET_QOS_OK | 700 | No error. |

NPQ (Network Protocol Quality) Related Error

| Error Name | Error Code | Error Description |
|---------------------------------|------------|--|
| NET_ERR_NPQ_PARAM | 8001 | NPQ library: Incorrect parameter. |
| NET_ERR_NPQ_SYSTEM | 8002 | NPQ library: Operating system error. |
| NET_ERR_NPQ_GENRAL | 8003 | NPQ library: Internal error. |
| NET_ERR_NPQ_PRECONDITION | 8004 | NPQ library: Calling sequence error. |
| NET_ERR_NPQ_NOTSUPPORT | 8005 | NPQ library: This function is not supported. |
| NET_ERR_NPQ_NOTCALLBACK | 8100 | No data is called back. |
| NET_ERR_NPQ_LOADLIB | 8101 | Loading NPQ library failed. |
| NET_ERR_NPQ_STREAM_CLOSE | 8104 | The NPQ function of this stream is not enabled. |
| NET_ERR_NPQ_MAX_LINK | 8110 | No more streaming channel's NPQ function can be enabled. |
| NET_ERR_NPQ_STREAM_CFG_CONFLICT | 8111 | The configured encoding parameters conflicted. |

B.2 Response Codes of Text Protocol

The response codes returned during the text protocol integration is based on the status codes of HTTP. 7 kinds of status codes are predefined, including 1 (OK), 2 (Device Busy), 3 (Device Error), 4 (Invalid Operation), 5 (Invalid Message Format), 6 (Invalid Message Content), and 7 (Reboot Required). Each kind of status code contains multiple sub status codes, and the response codes are in a one-to-one correspondence with the sub status codes.

StatusCode=1

| SubStatusCode | Error Code | Description |
|---------------|------------|----------------------|
| ok | 0x1 | Operation completed. |
| riskPassword | 0x10000002 | Risky password. |
| armProcess | 0x10000005 | Arming process. |

StatusCode=2

| Sub Status Code | Error Code | Description |
|------------------------------|------------|--|
| noMemory | 0x20000001 | Insufficient memory. |
| serviceUnavailable | 0x20000002 | The service is not available. |
| upgrading | 0x20000003 | Upgrading. |
| deviceBusy | 0x20000004 | The device is busy or no response. |
| reConnectIpc | 0x20000005 | The video server is reconnected. |
| transferUpgradePackageFailed | 0x20000006 | Transmitting device upgrade data failed. |
| startUpgradeFailed | 0x20000007 | Starting upgrading device failed. |
| getUpgradeProcessfailed. | 0x20000008 | Getting upgrade status failed. |
| certificateExist | 0x2000000B | The Authentication certificate already exists. |

StatusCode=3

| Sub Status Code | Error Code | Description |
|------------------------|------------|---|
| deviceError | 0x30000001 | Hardware error. |
| badFlash | 0x30000002 | Flash operation error. |
| 28181Uninitialized | 0x30000003 | The 28181 configuration is not initialized. |
| socketConnectError | 0x30000005 | Connecting to socket failed. |
| receiveError | 0x30000007 | Receive response message failed. |
| deletePictureError | 0x3000000A | Deleting picture failed. |
| pictureSizeExceedLimit | 0x3000000C | Too large picture size. |
| clearCacheError | 0x3000000D | Clearing cache failed. |
| updateDatabasError | 0x3000000F | Updating database failed. |

| Sub Status Code | Error Code | Description |
|-------------------------------|------------|--|
| searchDatabaseError | 0x30000010 | Searching in the database failed. |
| writeDatabaseError | 0x30000011 | Writing to database failed. |
| deleteDatabaseError | 0x30000012 | Deleting database element failed. |
| searchDatabaseElementError | 0x30000013 | Getting number of database elements failed. |
| cloudAutoUpgradeException | 0x30000016 | Downloading upgrade packet from cloud and upgrading failed. |
| HBPEXception | 0x30001000 | HBP exception. |
| UDEPEXception | 0x30001001 | UDEP exception |
| elasticSearchException | 0x30001002 | Elastic exception. |
| kafkaException | 0x30001003 | Kafka exception. |
| HBaseException | 0x30001004 | Hbase exception. |
| sparkException | 0x30001005 | Spark exception. |
| yarnException | 0x30001006 | Yarn exception. |
| cacheException | 0x30001007 | Cache exception. |
| trafficException | 0x30001008 | Monitoring point big data server exception. |
| faceException | 0x30001009 | Human face big data server exception. |
| SSDFileSystemsIsError | 0x30001013 | SSD file system error (Error occurs when it is non-Ext4 file system) |
| insufficientSSDCapacityForFPD | 0x30001014 | Insufficient SSD space for person frequency detection. |
| wifiException | 0x3000100A | Wi-Fi big data server exception |
| structException | 0x3000100D | Video parameters structure server exception. |
| noLinkageResource | 0x30001015 | Insufficient linkage resources. |

| Sub Status Code | Error Code | Description |
|--------------------------------|------------|--|
| engineAbnormal | 0x30002015 | Engine exception. |
| engineInitialization | 0x30002016 | Initializing the engine. |
| algorithmLoadingFailed | 0x30002017 | Loading the model failed. |
| algorithmDownloadFailed | 0x30002018 | Downloading the model failed. |
| algorithmDecryptionFailed | 0x30002019 | Decrypting the model failed. |
| unboundChannel | 0x30002020 | Delete the linked channel to load the new model. |
| unsupportedResolution | 0x30002021 | Invalid resolution. |
| unsupportedStreamType | 0x30002022 | Invalid stream type. |
| insufficientDecRes | 0x30002023 | Insufficient decoding resources. |
| insufficientEnginePerformance | 0x30002024 | Insufficient engine performance (The number of channels to be analyzed exceeds the engine's capability). |
| improperResolution | 0x30002025 | Improper resolution (The maximum resolution allowed is 4096×4096). |
| improperPicSize | 0x30002026 | Improper picture size (The maximum size allowed is 5MB). |
| URLDownloadFailed | 0x30002027 | Downloading the picture via the URI failed. |
| unsupportedImageFormat | 0x30002028 | Invalid picture format (Only JPG is supported currently). |
| unsupportedPollingIntervalTime | 0x30002029 | Invalid polling interval (The interval should be more than 10s). |
| exceedImagesNumber | 0x30002030 | The number of pictures exceeds the limit (The platform can apply 1 to 100 picture URIs per time, the maximum number allowed is 100). |

| Sub Status Code | Error Code | Description |
|-----------------------------|------------|--|
| unsupportedMPID | 0x30002031 | The applied MPID does not exist in the device, so updating this MPID is not supported. |
| modelPackageNotMatchLabel | 0x30002032 | The model and the description file mismatch. |
| modelPackageNotMatchTask | 0x30002033 | The task and the model type mismatch. |
| insufficientSpace | 0x30002034 | Insufficient space (When the number of model packages does not reach the maximum number allowed but their size together exceeds the free space, the model packages cannot be added). |
| engineUnLoadingModelPackage | 0x30002035 | Applying the task failed. This engine is not linked to a model package (Canceling the linkage failed, this engine is not linked to a model package). |
| engineWithModelPackage | 0x30002036 | Linking the engine to this model package failed. The engine has been linked to another model package. Please cancel their linkage first. |
| modelPackageDelete | 0x30002037 | Linking the model package failed. The model package has been deleted. |
| deleteTaskFailed | 0x30002038 | Deleting the task failed (It is returned when the user fails to end a task). |
| modelPackageNumberslimited | 0x30002039 | Adding the model package failed. The number of model package has reached the maximum number allowed. |
| modelPackageDeleteFailed | 0x30002040 | Deleting the model package failed. |

| Sub Status Code | Error Code | Description |
|--------------------|------------|--------------------------------|
| noArmingResource | 0x30001016 | Insufficient arming resources. |
| calibrationTimeout | 0x30002051 | Calibration timed out. |
| captureTimeout | 0x30006000 | Data collection timed out. |
| lowScore | 0x30006001 | Low quality of collected data. |
| uploadingFailed | 0x30007004 | Uploading failed. |

StatusCode=4

| Sub Status Code | Error Code | Description |
|---------------------------|------------|--|
| notSupport | 0x40000001 | Not supported. |
| lowPrivilege | 0x40000002 | No permission. |
| badAuthorization | 0x40000003 | Authentication failed. |
| methodNotAllowed | 0x40000004 | Invalid HTTP method. |
| notSetHdiskRedund | 0x40000005 | Setting spare HDD failed. |
| invalidOperation | 0x40000006 | Invalid operation. |
| notActivated | 0x40000007 | Inactivated. |
| hasActivated | 0x40000008 | Activated. |
| certificateAlreadyExist | 0x40000009 | The certificate already exists. |
| operateFailed | 0x4000000F | Operation failed. |
| USBNotExist | 0x40000010 | USB device is not connected. |
| upgradePackageMorethan2GB | 0x40001000 | Up to 2GB upgrade package is allowed to be uploaded. |
| IDNotExist | 0x40001001 | The ID does not exist. |
| interfaceOperationError | 0x40001002 | API operation failed. |
| synchronizationError | 0x40001003 | Synchronization failed. |
| synchronizing | 0x40001004 | Synchronizing. |
| importError | 0x40001005 | Importing failed. |
| importing | 0x40001006 | Importing. |

| Sub Status Code | Error Code | Description |
|--------------------------------|------------|--|
| fileAlreadyExists | 0x40001007 | The file already exists. |
| invalidID | 0x40001008 | Invalid ID. |
| backupnodeNotAllowe Log | 0x40001009 | Accessing to backup node is not allowed. |
| exportingError | 0x4000100A | Exporting failed. |
| exporting | 0x4000100B | Exporting. |
| exportEnded | 0x4000100C | Exporting stopped. |
| exported | 0x4000100D | Exported. |
| IPOccupied | 0x4000100E | The IP address is already occupied. |
| IDAlreadyExists | 0x4000100F | The ID already exists. |
| exportItemsExceedLimi t | 0x40001010 | No more items can be exported. |
| noFiles | 0x40001011 | The file does not exist. |
| beingExportedByAnoth erUser | 0x40001012 | Being exported by others. |
| needReAuthentication | 0x40001013 | Authentication is needed after upgrade. |
| unitAddNotOnline | 0x40001015 | The added data analysis server is offline. |
| unitControl | 0x40001016 | The data analysis server is already added. |
| analysis unitFull | 0x40001017 | No more data analysis server can be added. |
| unitIDError | 0x40001018 | The data analysis server ID does not exist. |
| unitExit | 0x40001019 | The data analysis server already exists in the list. |
| unitSearch | 0x4000101A | Searching data analysis server in the list failed. |
| unitNotOnline | 0x4000101B | The data analysis server is offline. |
| unitInfoError | 0x4000101C | Getting data analysis server information failed. |
| unitGetNodeInfoError | 0x4000101D | Getting node information failed. |
| unitGetNetworkInfoErr or | 0x4000101E | Getting the network information of data analysis server failed |
| unitSetNetworkInfoErr or | 0x4000101F | Setting the network information of data analysis server failed |

| Sub Status Code | Error Code | Description |
|--------------------------|------------|--|
| setSmartNodeInfoError | 0x40001020 | Setting node information failed. |
| setUnitNetworkInfoError | 0x40001021 | Setting data analysis server network information failed. |
| unitRestartCloseError | 0x40001022 | Rebooting or shutting down data analysis server failed. |
| virtualIPnotAllowed | 0x40001023 | Adding virtual IP address is not allowed. |
| unitInstalled | 0x40001024 | The data analysis server is already installed. |
| badSubnetMask | 0x40001025 | Invalid subnet mask. |
| uintVersionMismatched | 0x40001026 | Data analysis server version mismatches. |
| deviceModelMismatched | 0x40001027 | Adding failed. Device model mismatches. |
| unitAddNotSelf | 0x40001028 | Adding peripherals is not allowed. |
| noValidUnit | 0x40001029 | No valid data analysis server. |
| unitNameDuplicate | 0x4000102A | Duplicated data analysis server name. |
| deleteUnitFirst | 0x4000102B | Delete the added data analysis server of the node first. |
| getLocalInfoFailed | 0x4000102C | Getting the server information failed. |
| getClientAddedNodeFailed | 0x4000102D | Getting the added node information of data analysis server failed. |
| taskExit | 0x4000102E | The task already exists. |
| taskInitError | 0x4000102F | Initializing task failed. |
| taskSubmitError | 0x40001030 | Submitting task failed. |
| taskDelError | 0x40001031 | Deleting task failed. |
| taskPauseError | 0x40001032 | Pausing task failed. |
| taskContinueError | 0x40001033 | Starting task failed. |
| taskSeverNoCfg | 0x40001035 | Full-text search server is not configured. |
| taskPicSeverNoCfg | 0x40001036 | The picture server is not configured. |
| taskStreamError | 0x40001037 | Streaming information exception. |
| taskRecSDK | 0x40001038 | History recording is not supported. |

| Sub Status Code | Error Code | Description |
|------------------------|------------|--|
| taskCasaError | 0x4000103A | Cascading is not supported. |
| taskVCARuleError | 0x4000103B | Invalid VCA rule. |
| taskNoRun | 0x4000103C | The task is not executed. |
| unitLinksNoStorageNode | 0x4000103D | No node is linked with the data analysis server. Configure the node first. |
| searchFailed | 0x4000103E | Searching video files failed. |
| searchNull | 0x4000103F | No video clip. |
| userScheOffline | 0x40001040 | The task scheduler service is offline. |
| updateTypeUnmatched | 0x40001041 | The upgrade package type mismatches. |
| userExist | 0x40001043 | The user already exists. |
| userCannotDelAdmin | 0x40001044 | The administrator cannot be deleted. |
| userInexistence | 0x40001045 | The user name does not exist. |
| userCannotCreatAdmin | 0x40001046 | The administrator cannot be created. |
| monitorCamExceed | 0x40001048 | Up to 3000 cameras can be added. |
| monitorCunitOverLimit | 0x40001049 | Adding failed. Up to 5 lower-levels are supported by the control center. |
| monitorReginOverLimit | 0x4000104A | Adding failed. Up to 5 lower-levels are supported by the area. |
| monitorArming | 0x4000104B | The camera is already armed. Disarm the camera and try again. |
| monitorSyncCfgNotSet | 0x4000104C | The system parameters are not configured. |
| monitorFdSyncing | 0x4000104E | Synchronizing. Try again after completing the synchronization. |
| monitorParseFailed | 0x4000104F | Parsing camera information failed. |
| monitorCreatRootFailed | 0x40001050 | Creating resource node failed. |
| deleteArmingInfo | 0x40001051 | The camera is already . Disarm the camera and try again. |
| cannotModify | 0x40001052 | Editing is not allowed. Select again. |

| Sub Status Code | Error Code | Description |
|------------------------|------------|---|
| cannotDel | 0x40001053 | Deletion is not allowed. Select again. |
| deviceExist | 0x40001054 | The device already exists. |
| IPErrorConnectFailed | 0x40001056 | Connection failed. Check the network port. |
| cannotAdd | 0x40001057 | Only the capture cameras can be added. |
| serverExist | 0x40001058 | The server already exists. |
| fullTextParamError | 0x40001059 | Incorrect full-text search parameters. |
| storParamError | 0x4000105A | Incorrect storage server parameters. |
| picServerFull | 0x4000105B | The storage space of picture storage server is full. |
| NTPUnconnect | 0x4000105C | Connecting to NTP server failed. Check the parameters. |
| storSerConnectFailed | 0x4000105D | Connecting to storage server failed. Check the network port. |
| storSerLoginFailed | 0x4000105E | Logging in to storage server failed. Check the user name and password. |
| searchSerConnectFailed | 0x4000105F | Connecting to full-text search server failed. Check the network port. |
| searchSerLoginFailed | 0x40001060 | Logging in to full-text search server failed. Check the user name and password. |
| kafkaConnectFailed | 0x40001061 | Connecting to Kafka failed. Check the network port. |
| mgmtConnectFailed | 0x40001062 | Connecting to system failed. Check the network port. |
| mgmtLoginFailed | 0x40001063 | Logging in to system failed. Check the user name and password. |
| TDACConnectFailed | 0x40001064 | Connecting to traffic data access server failed. Checking the server status. |
| 86sdkConnectFailed | 0x40001065 | Connecting to listening port of iVMS-8600 System failed. Check the parameters. |
| nameExist | 0x40001066 | Duplicated server name. |
| batchProcessFailed | 0x40001067 | Processing in batch failed. |

| Sub Status Code | Error Code | Description |
|-------------------------------|------------|---|
| IDNotExist | 0x40001068 | The server ID does not exist. |
| serviceNumberReachesLimit | 0x40001069 | No more service can be added. |
| invalidServiceType. | 0x4000106A | Invalid service type. |
| clusterGetInfo | 0x4000106B | Getting cluster group information failed. |
| clusterDelNode | 0x4000106C | Deletion node failed. |
| clusterAddNode | 0x4000106D | Adding node failed. |
| clusterInstalling | 0x4000106E | Creating cluster...Do not operate. |
| clusterUninstall | 0x4000106F | Reseting cluster...Do not operate. |
| clusterInstall | 0x40001070 | Creating cluster failed. |
| clusterIpError | 0x40001071 | Invalid IP address of task scheduler server. |
| clusterNotSameSeg | 0x40001072 | The main node and sub node must be in the same network segment. |
| clusterVirIpError | 0x40001073 | Automatically getting virtual IP address failed. Enter manually. |
| clusterNodeUnadd | 0x40001074 | The specified main (sub) node is not added. |
| clusterNodeOffline | 0x40001075 | The task scheduler server is offline. |
| nodeNotCurrentIP | 0x40001076 | The analysis node of the current IP address is required when adding main and sub nodes. |
| addNodeNetFailed | 0x40001077 | Adding node failed. The network disconnected. |
| needTwoMgmtNode | 0x40001078 | Two management nodes are required when adding main and sub nodes. |
| ipConflict | 0x40001079 | The virtual IP address and data analysis server's IP address conflicted. |
| ipUsed | 0x4000107A | The virtual IP address has been occupied. |
| cloudAlalyseOnline | 0x4000107B | The cloud analytic server is online. |
| virIP&mainIPnotSameNetSegment | 0x4000107C | The virtual IP address is not in the same network segment with the IP address of main/sub node. |
| getNodeDispatchInfoFailed | 0x4000107D | Getting node scheduler information failed. |

| Sub Status Code | Error Code | Description |
|---------------------------------|------------|--|
| unableModifyManagementNetworkIP | 0x4000107E | Editing management network interface failed. The analysis board is in the cluster. |
| notSpecifyVirtualIP | 0x4000107F | Virtual IP address should be specified for main and sub cluster. |
| armingFull | 0x40001080 | No more device can be armed. |
| armingNoFind | 0x40001081 | The arming information does not exist. |
| disArming | 0x40001082 | Disarming failed. |
| getArmingError | 0x40001084 | Getting arming information failed. |
| refreshArmingError | 0x40001085 | Refreshing arming information failed. |
| ArmingPlateSame | 0x40001086 | The license plate number is repeatedly armed. |
| ArmingParseXLSError | 0x40001087 | Parsing arming information file failed. |
| ArmingTimeError | 0x40001088 | Invalid arming time period. |
| ArmingSearchTimeError | 0x40001089 | Invalid search time period. |
| armingRelationshipReachesLimit | 0x4000108A | No more relation can be created. |
| duplicateArmingName | 0x4000108B | The relation name already exists. |
| noMoreArmingListAdded | 0x4000108C | No more blacklist library can be armed. |
| noMoreCamerasAdded | 0x4000108D | No more camera can be armed. |
| noMoreArmingListAddedWithCamera | 0x4000108E | No more library can be linked to the camera. |
| noMoreArmingPeriodAdded | 0x4000108F | No more time period can be added to the arming schedule. |
| armingPeriodsOverlapped | 0x40001090 | The time periods in the arming schedule are overlapped. |
| noArmingAlarmInfo | 0x40001091 | The alarm information does not exist. |
| armingAlarmUnRead | 0x40001092 | Getting number of unread alarms failed. |
| getArmingAlarmError | 0x40001093 | Getting alarm information failed. |

| Sub Status Code | Error Code | Description |
|--------------------------------|------------|---|
| searchByPictureTimed Out | 0x40001094 | Searching picture by picture timeout. Search again. |
| comparisonTimeRange Error | 0x40001095 | Comparison time period error. |
| selectMonitorNumber UpperLimit | 0x40001096 | No more monitoring point ID can be filtered. |
| noMoreComparisonTasksAdded | 0x40001097 | No more comparison task can be executed at the same time. |
| GetComparisonResultFailed | 0x40001098 | Getting comparison result failed. |
| comparisonTypeError | 0x40001099 | Comparison type error. |
| comparisonUnfinished | 0x4000109A | The comparison is not completed. |
| facePictureModelInvalid | 0x4000109B | Invalid face model. |
| duplicateLibraryName. | 0x4000109C | The library name already exists. |
| noRecord | 0x4000109D | No record found. |
| countingRecordsFailed. | 0x4000109E | Calculate the number of records failed. |
| getHumanFaceFrameFailed | 0x4000109F | Getting face thumbnail from the picture failed. |
| modelingFailed. | 0x400010A0 | Modeling face according to picture URL failed. |
| 1V1FacePictureComparisonFailed | 0x400010A1 | Comparison 1 VS 1 face picture failed. |
| libraryArmed | 0x400010A2 | The blacklist library is armed. |
| licenseExceedLimit | 0x400010A3 | Dongle limited. |
| licenseExpired | 0x400010A4 | Dongle expired. |
| licenseDisabled | 0x400010A5 | Unavailable dongle. |
| licenseNotExist | 0x400010A6 | The dongle does not exist. |
| SessionExpired | 0x400010A7 | Session expired . |
| beyondConcurrentLimit | 0x400010A8 | Out of concurrent limit. |
| stopSync | 0x400010A9 | Synchronization stopped. |

| Sub Status Code | Error Code | Description |
|----------------------------------|------------|---|
| getProgressFailed | 0x400010AA | Getting progress failed. |
| uploadExtraCaps | 0x400010AB | No more files can be uploaded. |
| timeRangeError | 0x400010AC | Time period error. |
| dataPortNotConnected | 0x400010AD | The data port is not connected. |
| addClusterNodeFailed | 0x400010AE | Adding to the cluster failed. The device is already added to other cluster. |
| taskNotExist | 0x400010AF | The task does not exist. |
| taskQueryFailed | 0x400010B0 | Searching task failed. |
| modifyTimeRuleFailed | 0x400010B2 | The task already exists. Editing time rule is not allowed. |
| modifySmartRuleFailed | 0x400010B3 | The task already exists. Editing VAC rule is not allowed. |
| queryHistoryVideoFailed | 0x400010B4 | Searching history video failed. |
| addDeviceFailed | 0x400010B5 | Adding device failed. |
| addVideoFailed | 0x400010B6 | Adding video files failed. |
| deleteAllVideoFailed | 0x400010B7 | Deleting all video files failed. |
| createVideoIndexFailed | 0x400010B8 | Indexing video files failed. |
| videoCheckTypeFailed | 0x400010B9 | Verifying video files types failed. |
| configStructuredAddressFailed | 0x400010BA | Configuring IP address of structured server failed. |
| configPictureServerAddressFailed | 0x400010BB | Configuring IP address of picture stored server failed. |
| storageServiceIPNotExist | 0x400010BD | The storage server IP address does not exist. |
| syncBackupDatabaseFailed | 0x400010BE | Synchronizing sub database failed. Try again. |
| syncBackupNTPTimeFailed | 0x400010BF | Synchronizing NTP time of sub server failed. |
| clusterNotSelectLoopbackAddress | 0x400010C0 | Loopback address is not supported by the main or sub cluster. |

| Sub Status Code | Error Code | Description |
|----------------------------------|------------|---|
| addFaceRecordFailed | 0x400010C1 | Adding face record failed. |
| deleteFaceRecordFailed | 0x400010C2 | Deleting face record failed. |
| modifyFaceRecordFailed | 0x400010C3 | Editing face record failed. |
| queryFaceRecordFailed | 0x400010C4 | Searching face record failed. |
| faceDetectFailed | 0x400010C5 | Detecting face failed. |
| libraryNotExist | 0x400010C6 | The library does not exist. |
| blackListQueryExporting | 0x400010C7 | Exporting matched blocklists. |
| blackListQueryExported | 0x400010C8 | The matched blocklists are exported. |
| blackListQueryStopExporting | 0x400010C9 | Exporting matched blocklists is stopped. |
| blackListAlarmQueryExporting | 0x400010CA | Exporting matched blocklist alarms. |
| blackListAlarmQueryExported | 0x400010CB | The matched blocklists alarms are exported. |
| blackListAlarmQueryStopExporting | 0x400010CC | Exporting matched blocklist alarms is stopped. |
| getBigDataCloudAnalysisFailed | 0x400010CD | Getting big data cloud analytic information failed. |
| setBigDataCloudAnalysisFailed | 0x400010CE | Configuring big data cloud analytic failed. |
| submitMapSearchFailed | 0x400010CF | Submitting search by picture task failed. |
| controlRelationshipNotExist | 0x400010D0 | The relation does not exist. |
| getHistoryAlarmInfoFailed | 0x400010D1 | Getting history alarm information failed. |
| getFlowReportFailed | 0x400010D2 | Getting people counting report failed. |
| addGuardFailed | 0x400010D3 | Adding arming configuration failed. |

| Sub Status Code | Error Code | Description |
|--|------------|---|
| deleteGuardFailed | 0x400010D4 | Deleting arming configuration failed. |
| modifyGuardFailed | 0x400010D5 | Editing arming configuration failed. |
| queryGuardFailed | 0x400010D6 | Searching arming configurations failed. |
| uploadUserSuperCaps | 0x400010D7 | No more user information can be uploaded. |
| bigDataServerConnect Failed | 0x400010D8 | Connecting to big data server failed. |
| microVideoCloudRequ estInfoBuildFailed | 0x400010D9 | Adding response information of micro video cloud failed. |
| microVideoCloudRespo nseInfoBuildFailed | 0x400010DA | Parsing response information of micro video cloud failed. |
| transcodingServerRequ estInfoBuildFailed | 0x400010DB | Adding response information of transcoding server failed. |
| transcodingServerResp onseInfoParseFailed | 0x400010DC | Parsing response information of transcoding server failed. |
| transcodingServerOffli ne | 0x400010DD | Transcoding server is offline. |
| microVideoCloudOfflin e | 0x400010DE | Micro video cloud is offline. |
| UPSServerOffline | 0x400010DF | UPS monitor server is offline. |
| statisticReportRequestI nfoBuildFailed | 0x400010E0 | Adding response information of statistics report failed. |
| statisticReportRespons eInfoParseFailed | 0x400010E1 | Parsing response information of statistics report failed. |
| DisplayConfigInfoBuild Failed | 0x400010E2 | Adding display configuration information failed. |
| DisplayConfigInfoParse Failed | 0x400010E3 | Parsing display configuration information failed. |
| DisplayConfigInfoSaveF ailed | 0x400010E4 | Saving display configuration information failed. |
| notSupportDisplayConf igType | 0x400010E5 | The display configuration type is not supported. |
| passError | 0x400010E7 | Incorrect password. |

| Sub Status Code | Error Code | Description |
|--------------------------------------|------------|---|
| upgradePackageLarge | 0x400010EB | Too large upgrade package. |
| sessionUserReachesLimit | 0x400010EC | No more user can log in via session. |
| ISO8601TimeFormatError | 0x400010ED | Invalid ISO8601 time format. |
| clusterDissolutionFailed | 0x400010EE | Deleting cluster failed. |
| getServiceNodeInfoFailed | 0x400010EF | Getting service node information failed. |
| getUPSInfoFailed | 0x400010F0 | Getting UPS configuration information failed. |
| getDataStatisticsReportFailed | 0x400010F1 | Getting data statistic report failed. |
| getDisplayConfigInfoFailed | 0x400010F2 | Getting display configuration failed. |
| namingAnalysisBoardNotAllowed | 0x400010F3 | Renaming analysis board is not allowed. |
| onlyDrawRegionsOfConvexPolygon | 0x400010F4 | Only drawing convex polygon area is supported. |
| bigDataServerResponseInfoParseFailed | 0x400010F5 | Parsing response message of big data service failed. |
| bigDataServerReturnFailed | 0x400010F6 | No response is returned by big data service. |
| microVideoReturnFailed | 0x400010F7 | No response is returned by micro video cloud service. |
| transcodingServerReturnFailed | 0x400010F8 | No response is returned by transcoding service. |
| UPSServerReturnFailed | 0x400010F9 | No response is returned by UPS monitoring service. |
| forwardingServerReturnFailed | 0x400010FA | No response is returned by forwarding service. |
| storageServerReturnFailed | 0x400010FB | No response is returned by storage service. |

| Sub Status Code | Error Code | Description |
|---|------------|--|
| cloudAnalysisServerReturnFailed | 0x400010FC | No response is returned by cloud analytic service. |
| modelEmpty | 0x400010FD | No model is obtained. |
| mainAndBackupNodeCannotModifyManagementNetworkInterfaceIP | 0x400010FE | Editing the management interface IP address of main node and backup node is not allowed. |
| IDTooLong | 0x400010FF | The ID is too long. |
| pictureCheckFailed | 0x40001100 | Detecting picture failed. |
| pictureModelingFailed | 0x40001101 | Modeling picture failed. |
| setCloudAnalysisDefaultProvinceFailed | 0x40001102 | Setting default province of cloud analytic service failed. |
| inspectionAreasNumberExceedLimit | 0x40001103 | No more detection regions can be added. |
| picturePixelsTooLarge | 0x40001105 | The picture resolution is too high. |
| picturePixelsTooSmall | 0x40001106 | The picture resolution is too low. |
| storageServiceIPEmpty | 0x40001107 | The storage server IP address is required. |
| bigDataServerRequestInfoBuildFail | 0x40001108 | Creating request message of big data service failed. |
| analysisTimedOut | 0x40001109 | Analysis time out. |
| high-performanceModeDisabled. | 0x4000110A | Please enable high-performance mode. |
| configuringUPSMonitoringServerTimedOut | 0x4000110B | Configuring the UPS monitoring server time out. Check IP address. |
| cloudAnalysisRequestInformationBuildFailed | 0x4000110C | Creating request message of cloud analytic service failed. |
| cloudAnalysisResponseInformationParseFailed | 0x4000110D | Parsing response message of cloud analytic service failed. |
| allCloudAnalysisInterfaceFailed | 0x4000110E | Calling API for cloud analytic service failed. |
| cloudAnalysisModelCompareFailed | 0x4000110F | Model comparison of cloud analytic service failed. |

| Sub Status Code | Error Code | Description |
|---|------------|---|
| cloudAnalysisFacePictureQualityRatingFailed | 0x40001110 | Getting face quality grading of cloud analytic service failed. |
| cloudAnalysisExtractFeaturePointsFailed | 0x40001111 | Extracting feature of cloud analytic service failed. |
| cloudAnalysisExtractPropertyFailed | 0x40001112 | Extracting property of cloud analytic service failed. |
| getAddedNodeInformationFailed | 0x40001113 | Getting the added nodes information of data analysis server failed. |
| noMoreAnalysisUnitsAdded | 0x40001114 | No more data analysis servers can be added. |
| detectionAreaInvalid | 0x40001115 | Invalid detection region. |
| shieldAreaInvalid | 0x40001116 | Invalid shield region. |
| noMoreShieldAreasAdded | 0x40001117 | No more shield region can be drawn. |
| onlyAreaOfRectangleShapeAllowed | 0x40001118 | Only drawing rectangle is allowed in detection area. |
| numberReachedLimit | 0x40001119 | Number reached the limit. |
| wait1~3MinutesGetIPAfterSetupDHCP | 0x4000111A | Wait 1 to 3 minutes to get IP address after configuring DHCP. |
| plannedTimeMustbeHalfAnHour | 0x4000111B | Schedule must be half an hour. |
| oneDeviceCannotBuildCluster | 0x4000111C | Creating main and backup cluster requires at least two devices. |
| updatePackageFileNotUploaded | 0x4000111E | Upgrade package is not uploaded. |
| highPerformanceTasksNotSupportDrawingDetectionRegions | 0x4000111F | Drawing detection area is not allowed under high-performance mode. |
| controlCenterIDDoesNotExist | 0x40001120 | The control center ID does not exist. |
| regionIDDoesNotExist | 0x40001121 | The area ID does not exist. |
| licensePlateFormatError | 0x40001122 | Invalid license plate format. |

| Sub Status Code | Error Code | Description |
|--|------------|---|
| managementNodeDoesNotSupportThisOperation | 0x40001123 | The operation is not supported. |
| searchByPictureResourceNotConfiged | 0x40001124 | The conditions for searching picture by picture are not configured. |
| videoFileEncapsulationFormatNotSupported | 0x40001125 | The video container format is not supported. |
| videoPackageFailure | 0x40001126 | Converting video container format failed. |
| videoCodingFormatNotSupported | 0x40001127 | Video coding format is not supported. |
| monitorOfDeviceArmingdeleteArmingInfo | 0x40001129 | The camera is armed. Disarm it and try again. |
| getVideoSourceTypeFailed | 0x4000112A | Getting video source type failed. |
| smartRulesBuildFailed | 0x4000112B | Creating VAC rule failed. |
| smartRulesParseFailed | 0x4000112C | Parsing VAC rule failed. |
| timeRulesBuildFailed | 0x4000112D | Creating time rule failed. |
| timeRulesParseFailed | 0x4000112E | Parsing time rule failed. |
| monitoInfoInvalid | 0x4000112F | Invalid camera information. |
| addingFailedVersionMismatches | 0x40001130 | Adding failed. The device version mismatches. |
| theInformationReturnedAfterCloudAnalysisIsEmpty | 0x40001131 | No response is returned by the cloud analytic service. |
| selectingIpAddressOfHostAndSpareNodeFailedCheckTheStatus | 0x40001132 | Setting IP address for main node and backup node failed. Check the node status. |
| theSearchIdDoesNotExist | 0x40001133 | The search ID does not exist. |
| theSynchronizationIdDoesNotExist | 0x40001134 | The synchronization ID does not exist. |
| theUserIdDoesNotExist | 0x40001136 | The user ID does not exist. |

| Sub Status Code | Error Code | Description |
|---|------------|--|
| theIndexCodeDoesNotExist | 0x40001138 | The index code does not exist. |
| theControlCenterIdDoesNotExist | 0x40001139 | The control center ID does not exist. |
| theAreaIdDoesNotExist | 0x4000113A | The area ID does not exist. |
| theArmingLinkageIdDoesNotExist | 0x4000113C | The arming relationship ID does not exist. |
| theListLibraryIdDoesNotExist | 0x4000113D | The list library ID does not exist. |
| invalidCityCode | 0x4000113E | Invalid city code. |
| synchronizingThePasswordOfSpareServerFailed | 0x4000113F | Synchronizing backup system password failed. |
| editingStreamingTypeIsNotSupported | 0x40001140 | Editing streaming type is not supported. |
| switchingScheduledTaskToTemporaryTaskIsNotSupported | 0x40001141 | Switching scheduled task to temporary task is not supported. |
| switchingTemporaryTaskToScheduledTaskIsNotSupported | 0x40001142 | Switching temporary task to scheduled task is not supported. |
| theTaskIsNotDispatchedOrItIsUpdating | 0x40001143 | The task is not dispatched or is updating. |
| thisTaskDoesNotExist | 0x40001144 | This task does not exist in the cloud analytic service. |
| duplicatedSchedule | 0x40001145 | Schedule period cannot be overlapped. |
| continuousScheduleWithSameAlgorithmTypeShouldBeMerged | 0x40001146 | The continuous schedule periods with same algorithm type should be merged. |
| invalidStreamingTimeRange | 0x40001147 | Invalid streaming time period. |
| invalidListLibraryType | 0x40001148 | Invalid list library type. |

| Sub Status Code | Error Code | Description |
|--|------------|--|
| theNumberOfMatchedResultsShouldBeLargerThan0 | 0x40001149 | The number of search results should be larger than 0. |
| invalidValueRangeOfSimilarity | 0x4000114A | Invalid similarity range. |
| invalidSortingType | 0x4000114B | Invalid sorting type. |
| noMoreListLibraryCanBeLinkedToTheDevice | 0x4000114C | No more lists can be added to one device. |
| InvalidRecipientAddressFormat | 0x4000114D | Invalid address format of result receiver. |
| creatingClusterFailedTheDongleIsNotPluggedIn | 0x4000114E | Insert the dongle before creating cluster. |
| theURLIsTooLong | 0x4000114F | No schedule configured for the task. |
| noScheduleIsConfiguredForTheTask | 0x40001150 | No schedule configured for the task. |
| theDongleIsExpired | 0x40001151 | Dongle has expired. |
| dongleException | 0x40001152 | Dongle exception. |
| invalidKey | 0x40001153 | Invalid authorization service key. |
| decryptionFailed | 0x40001154 | Decrypting authorization service failed. |
| encryptionFailed | 0x40001155 | Encrypting authorization service failed. |
| AuthorizeServiceResponseError | 0x40001156 | Authorization service response exception. |
| incorrectParameter | 0x40001157 | Authorization service parameters error. |
| operationFailed | 0x40001158 | Operating authorization service error. |
| noAnalysisResourceOrNoDataInTheListLibrary | 0x40001159 | No cloud analytic resources or no data in the list library. |
| calculationException | 0x4000115A | Calculation exception. |
| allocatingList | 0x4000115B | Allocating list. |
| thisOperationIsNotSupportedByTheCloudAnalytics | 0x4000115C | This operation is not supported by the cloud analytic service. |

| Sub Status Code | Error Code | Description |
|---|------------|---|
| theCloudAnalyticsIsInterrupted | 0x4000115D | The operation of cloud analytic service is interrupted. |
| theServiceIsNotReady | 0x4000115E | The service is not ready. |
| searchingForExternalApiFailed | 0x4000115F | Searching external interfaces failed. |
| noOnlineNode | 0x40001160 | No node is online. |
| noNodeAllocated | 0x40001161 | No allocated node. |
| noMatchedList | 0x40001162 | No matched list. |
| allocatingFailedTooManyFacePictureLists | 0x40001163 | Allocation failed. Too many lists of big data service. |
| searchIsNotCompletedSearchAgain | 0x40001164 | Current searching is not completed. Search again. |
| allocatingListIsNotCompleted | 0x40001165 | Allocating list is not completed. |
| searchingForCloudAnalyticsResultsFailed | 0x40001166 | Searching cloud analytic service overtime. |
| noDataOfTheCurrentLibraryFound | 0x40001167 | No data in the current library. Make sure there is data in the Hbase. |
| noFacePictureLibraryIsArmed | 0x40001168 | No face picture library is armed for big data service. |
| noAvailableDataSlicingVersionInformationArmedFirstAndSliceTheData | 0x40001169 | Invalid standard version information. |
| duplicatedOperationDataSlicingIsExecuting | 0x4000116A | Slicing failed. Duplicated operation. |
| slicingDataFailedNoArmedFacePictureLibrary | 0x4000116B | Slicing failed. No arming information in the face big data. |
| GenerateBenchmarkFileFailedSlicingAgain | 0x4000116C | Generating sliced file failed. Slice again. |
| NonprimaryNodesProhibitedFromSlicingData | 0x4000116D | Slicing is not allowed by the backup node. |
| NoReadyNodeToClusterServers | 0x4000116E | Creating the cluster failed. No ready node. |

| Sub Status Code | Error Code | Description |
|--|------------|--|
| NodeManagementServicesOffline | 0x4000116F | The node management server is offline. |
| theCamera(s)OfTheControlCenterAreAlreadyArmed.DisarmThemFirst | 0x40001170 | Some cameras in control center are already armed. Disarm them and try again. |
| theCamera(s)OfTheAreaAreAlreadyArmed.DisarmThemFirst | 0x40001171 | Some cameras in this area are already armed. Disarm them and try again. |
| configuringHigh-frequencyPeopleDetectionFailed | 0x40001172 | Configuring high frequency people detection failed. |
| searchingForHigh-frequencyPeopleDetectionLogsFailed. | 0x40001173 | Searching detection event logs of high-frequency people detection failed. |
| gettingDetailsOfSearchHigh-frequencyPeopleDetectionLogsFailed. | 0x40001174 | Getting the search result details of frequently appeared person alarms failed. |
| theArmedCamerasAlreadyExistInTheControlCenter | 0x40001175 | Some cameras in control center are already armed. |
| disarmingFailedTheCamerasNotArmed | 0x40001177 | Disarming failed. The camera is not armed. |
| noDataReturned | 0x40001178 | No response is returned by the big data service. |
| preallocFailure | 0x40001179 | Pre-allocating algorithm resource failed. |
| overDogLimit | 0x4000117A | Configuration failed. No more resources can be pre-allocated. |
| analysisServicesDoNotSupport | 0x4000117B | Not supported. |
| commandAndDispatchServiceError | 0x4000117C | Scheduling service of cloud analytic service error. |
| engineModuleError | 0x4000117D | Engine module of cloud analytic service error. |

| Sub Status Code | Error Code | Description |
|--|------------|---|
| streamingServiceError | 0x4000117E | Streaming component of cloud analytic service error. |
| faceAnalysisModuleError | 0x4000117F | Face analysis module of cloud analytic service error. |
| vehicleAnalysisModuleError | 0x40001180 | Vehicle pictures analytic module of cloud analytic service error. |
| videoStructuralAnalysisModuleError | 0x40001181 | Video structuring module of cloud analytic service error. |
| postprocessingModuleError | 0x40001182 | Post-processing module of cloud analytic service error. |
| frequentlyAppearedPersonAlarmIsAlreadyConfiguredForListLibrary | 0x40001183 | Frequently appeared person alarm is already armed for blocklist library. |
| creatingListLibraryFailed | 0x40001184 | Creating list library failed. |
| invalidIdentityKeyOfListLibrary | 0x40001185 | Invalid identity key of list library. |
| noMoreDevicesCanBeArmed | 0x40001186 | No more camera can be added. |
| settingAlgorithmTypeForDeviceFailed | 0x40001187 | Allocating task resource failed. |
| gettingHighFrequencyPersonDetectionAlarmInformationFailed | 0x40001188 | Setting frequently appeared person alarm failed. |
| invalidSearchCondition | 0x40001189 | Invalid result. |
| theTaskIsNotCompleted | 0x4000118B | The task is not completed. |
| resourceOverRemainLimit | 0x4000118C | No more resource can be pre-allocated. |
| frequentlyAppearedPersonAlarmIsAlreadyConfiguredForTheCameraDisarmFirstAndTryAgain | 0x4000118D | The frequently appeared person alarm of this camera is configured. Delete the arming information and try again. |

| Sub Status Code | Error Code | Description |
|---------------------------------|------------|---|
| switchtimedifflesslimit | 0x4000123b | Time difference between power on and off should be less than 10 minutes. |
| associatedFaceLibNumOverLimit | 0x40001279 | Maximum number of linked face picture libraries reached. |
| noMorePeopleNumChangeRulesAdded | 0x4000128A | Maximum number of people number changing rules reached. |
| noMoreViolentMotionRulesAdded | 0x4000128D | Maximum number of violent motion rules reached. |
| noMoreLeavePositionRulesAdded | 0x4000128E | Maximum number of leaving position rules reached. |
| SMRDiskNotSupportRaid | 0x40001291 | SMR disk does not support RAID. |
| OnlySupportHikAndCustomProtocol | 0x400012A3 | IPv6 camera can only be added via Device Network SDK or custom protocols. |
| vehicleEnginesNoResource | 0x400012A6 | Insufficient vehicle engine resources. |
| noMoreRunningRulesAdded | 0x400012A9 | Maximum number of running rules reached. |
| noMoreGroupRulesAdded | 0x400012AA | Maximum number of people gathering rules reached. |
| noMoreFailDownRulesAdded | 0x400012AB | Maximum number of people falling down rules reached. |
| noMorePlayCellphoneRulesAdded | 0x400012AC | Maximum number of playing cellphone rules reached. |
| ruleEventTypeDuplicate | 0x400012C8 | Event type duplicated. |
| noMoreRetentionRulesAdded | 0x400015AD | Maximum number of people retention rules reached. |
| noMoreSleepOnDutyRulesAdded | 0x400015AE | Maximum number of sleeping on duty rules reached. |
| polygonNotAllowCrossing | 0x400015C2 | Polygons are not allowed to cross. |

| Sub Status Code | Error Code | Description |
|---------------------------------|------------|--|
| configureRuleBeforeAdvanceParam | 0x400015F8 | Advanced parameters fail to be configured as no rule is configured, please configure rule information first. |
| behaviorCanNotPackToPic | 0x40001603 | The behavior model cannot be packaged as a picture algorithm. |
| noCluster | 0x40001608 | No cluster created. |
| NotAssociatedWithOwnChannel | 0x400019C1 | Current channel is not linked. |
| AITargetBPCaptureFail | 0x400019C5 | Capturing reference picture for AI target comparison failed. |
| AITargetBPToDSPFail | 0x400019C6 | Sending reference picture to DSP for AI target comparison failed. |
| AITargetBPDuplicateName | 0x400019C7 | Duplicated name of reference picture for AI target comparison. |
| audioFileNameWrong | 0x400019D0 | Incorrect audio file name. |
| audioFileImportFail | 0x400019D1 | Importing audio file failed. |
| NonOperationalStandbyMachine | 0x400019F0 | Non-operational hot spare. |
| MaximumNumberOfDevices | 0x400019F1 | The maximum number of devices reached. |
| StandbyMachineCannotBeDeleted | 0x400019F2 | The hot spare cannot be deleted. |
| alreadyRunning | 0x40002026 | The application program is running. |
| notRunning | 0x40002027 | The application program is stopped. |
| packNotFound | 0x40002028 | The software packet does not exist. |
| alreadyExist | 0x40002029 | The application program already exists. |
| noMemory | 0x4000202A | Insufficient memory. |
| invalidLicense | 0x4000202B | Invalid License. |
| noClientCertificate | 0x40002036 | The client certificate is not installed. |
| noCACertificate | 0x40002037 | The CA certificate is not installed. |

| Sub Status Code | Error Code | Description |
|--|------------|--|
| authenticationFailed | 0x40002038 | Authenticating certificate failed. Check the certificate. |
| clientCertificateExpired | 0x40002039 | The client certificate is expired. |
| clientCertificateRevocation | 0x4000203A | The client certificate is revoked. |
| CACertificateExpired | 0x4000203B | The CA certificate is expired. |
| CACertificateRevocation | 0x4000203C | The CA certificate is revoked. |
| connectFail | 0x4000203D | Connection failed. |
| loginNumExceedLimit | 0x4000203F | No more user can log in. |
| HDMIResolutionIllegal | 0x40002040 | The HDMI video resolution cannot be larger than that of main and sub stream. |
| hdFormatFail | 0x40002049 | Formatting HDD failed. |
| formattingFailed | 0x40002056 | Formatting HDD failed. |
| encryptedFormattingFailed | 0x40002057 | Formatting encrypted HDD failed. |
| wrongPassword | 0x40002058 | Verifying password of SD card failed. Incorrect password. |
| audiolsPlayingPleaseWait | 0x40002067 | Audio is playing. Please wait. |
| twoWayAudioInProgressPleaseWait | 0x40002068 | Two-way audio in progress. Please wait. |
| calibrationPointNumFull | 0x40002069 | The maximum number of calibration points reached. |
| completeTheLevelCalibrationFirst | 0x4000206A | The level calibration is not set. |
| completeTheRadarCameraCalibrationFirst | 0x4000206B | The radar-camera calibration is not set. |
| pointsOnStraightLine | 0x4000209C | Calibrating failed. The calibration points cannot be one the same line. |
| TValueLessThanOrEqualZero | 0x4000209D | Calibration failed. The T value of the calibration points should be larger than 0. |

| Sub Status Code | Error Code | Description |
|-------------------------------------|------------|--|
| HBDLibNumOverLimit | 0x40002092 | The number of human body picture libraries reaches the upper limit |
| theShieldRegionError | 0x40002093 | Saving failed. The shielded area should be the ground area where the shielded object is located. |
| theDetectionAreaError | 0x40002094 | Saving failed. The detection area should only cover the ground area. |
| invalidLaneLine | 0x40002096 | Saving failed. Invalid lane line. |
| enableITSFunctionOfThisChannelFirst | 0x400020A2 | Enable ITS function of this channel first. |
| noCloudStorageServer | 0x400020C5 | No cloud storage server |
| NotSupportWithVideoTask | 0x400020F3 | This function is not supported. |
| noDetectionArea | 0x400050df | No detection area |
| armingFailed | 0x40008000 | Arming failed. |
| disarmingFailed | 0x40008001 | Disarming failed. |
| clearAlarmFailed | 0x40008002 | Clearing alarm failed. |
| bypassFailed | 0x40008003 | Bypass failed. |
| bypassRecoverFailed | 0x40008004 | Bypass recovery failed. |
| outputsOpenFailed | 0x40008005 | Opening relay failed. |
| outputsCloseFailed | 0x40008006 | Closing relay failed. |
| registerTimeOut | 0x40008007 | Registering timed out. |
| registerFailed | 0x40008008 | Registering failed. |
| addedByOtherHost | 0x40008009 | The peripheral is already added by other security control panel. |
| alreadyAdded | 0x4000800A | The peripheral is already added. |
| armedStatus | 0x4000800B | The partition is armed. |
| bypassStatus | 0x4000800C | Bypassed. |
| zoneNotSupport | 0x4000800D | This operation is not supported by the zone. |
| zoneFault | 0x4000800E | The zone is in fault status. |

| Sub Status Code | Error Code | Description |
|-----------------------------------|------------|--|
| pwdConflict | 0x4000800F | Password conflicted. |
| audioTestEntryFailed | 0x40008010 | Enabling audio test mode failed. |
| audioTestRecoveryFailed | 0x40008011 | Disabling audio test mode failed. |
| addCardMode | 0x40008012 | Adding card mode. |
| searchMode | 0x40008013 | Search mode. |
| addRemoterMode | 0x40008014 | Adding keyfob mode. |
| registerMode | 0x40008015 | Registration mode. |
| exDevNotExist | 0x40008016 | The peripheral does not exist. |
| theNumberOfExDevLimited | 0x40008017 | No peripheral can be added. |
| sirenConfigFailed | 0x40008018 | Setting siren failed. |
| chanCannotRepeatedBinded | 0x40008019 | This channel is already linked by the zone. |
| inProgramMode | 0x4000801B | The keypad is in programming mode. |
| inPaceTest | 0x4000801C | In pacing mode. |
| arming | 0x4000801D | Arming. |
| masterSlavelsEnable | 0x4000802c | The main-sub relationship has taken effect, the sub radar does not support this operation. |
| forceTrackNotEnabled | 0x4000802d | Mandatory tracking is disabled. |
| isNotSupportZoneConfigByLocalArea | 0x4000802e | This area does not support the zone type. |
| alarmLineCross | 0x4000802f | Trigger lines are overlapped. |
| zoneDrawingOutOfRange | 0x40008030 | The drawn zone is out of detection range. |
| alarmLineDrawingOutOfRange | 0x40008031 | The drawn alarm trigger line is out of detection range. |
| hasTargetInWarningArea | 0x40008032 | The warning zone already contains targets. Whether to enable mandatory arming? |
| radarMoudleConnectFail | 0x40008033 | Radar module communication failed. |

| Sub Status Code | Error Code | Description |
|----------------------------|------------|--|
| importCfgFilePasswordErr | 0x40008034 | Incorrect password for importing configuration files. |
| overAudioFileNumLimit | 0x40008038 | The number of audio files exceeds the limit. |
| audioFileNameIsLong | 0x40008039 | The audio file name is too long. |
| audioFormatIsWrong | 0x4000803a | The audio file format is invalid. |
| audioFileIsLarge | 0x4000803b | The size of the audio file exceeds the limit. |
| pircamCapTimeOut | 0x4000803c | Capturing of pircam timed out. |
| pircamCapFail | 0x4000803d | Capturing of pircam failed. |
| pircamIsCaping | 0x4000803e | The pircam is capturing. |
| audioFileHasExisted | 0x4000803f | The audio file already exists. |
| subscribeTypeErr | 0x4000a016 | This metadata type is not supported to be subscribed. |
| EISError | 0x4000A01C | Electronic image stabilization failed. The smart event function is enabled. |
| jpegPicWithAppendDataError | 0x4000A01D | Capturing the thermal graphic failed. Check if the temperature measurement parameters (emissivity, distance, reflective temperature) are configured correctly. |
| startAppFail | / | Starting running application program failed. |
| yuvconflict | / | The raw video stream conflicted. |
| overMaxAppNum | / | No more application program can be uploaded. |
| noFlash | / | Insufficient flash. |
| platMismatch | / | The platform mismatches. |
| emptyEventName | 0x400015E0 | Event name is empty. |
| sameEventName | 0x400015E1 | A same event name already exists. |
| emptyEventType | 0x400015E2 | Event type is required. |
| sameEventType | 0x400015E3 | A same event type already exists. |
| maxEventNameReached | 0x400015E4 | Maximum of events reached. |

| Sub Status Code | Error Code | Description |
|-----------------------------------|------------|--|
| hotSpareNotAllowedExternalStorage | 0x400015FC | External storage is not allowed when hot spare is enabled. |
| sameCustomProtocolName | 0x400015FD | A same protocol name already exists. |
| maxPTZTriggerChannelReached | 0x400015FE | Maximum of channels linked with PTZ reached. |
| POSCannotAddHolidayPlan | 0x400015FF | No POS events during holidays. |
| eventTypesTooLong | 0x40001600 | Event type is too long. |
| eventNamesTooLong | 0x40001601 | Event name is too long. |
| PerimeterEnginesNoResource | 0x40001602 | No more perimeter engines. |
| invalidProvinceCode | 0x40001607 | Invalid province code. |

StatusCode=5

| Sub Status Code | Error Code | Description |
|-----------------|------------|---------------------|
| badXmlFormat | 0x50000001 | Invalid XML format. |

StatusCode=6

| Sub Status Code | Error Code | Description |
|---------------------|------------|------------------------------|
| badParameters | 0x60000001 | Invalid parameter. |
| badHostAddress | 0x60000002 | Invalid host IP address. |
| badXmlContent | 0x60000003 | Invalid XML content. |
| badIPv4Address | 0x60000004 | Invalid IPv4 address. |
| badIPv6Address | 0x60000005 | Invalid IPv6 address. |
| conflictIPv4Address | 0x60000006 | IPv4 address conflicted. |
| conflictIPv6Address | 0x60000007 | IPv6 address conflicted. |
| badDomainName | 0x60000008 | Invalid domain name. |
| connectServerFail | 0x60000009 | Connecting to server failed. |

| Sub Status Code | Error Code | Description |
|-----------------------------------|------------|--|
| conflictDomainName | 0x6000000A | Domain name conflicted. |
| badPort | 0x6000000B | Port number conflicted. |
| portError | 0x6000000C | Port error. |
| exportErrorData | 0x6000000D | Importing data failed. |
| badNetMask | 0x6000000E | Invalid sub-net mask. |
| badVersion | 0x6000000F | Version mismatches. |
| badDevType | 0x60000010 | Device type mismatches. |
| badLanguage | 0x60000011 | Language mismatches. |
| incorrentUserNameOrPassword | 0x60000012 | Incorrect user name or password. |
| invalidStoragePoolOfCloudServer | 0x60000013 | Invalid storage pool. The storage pool is not configured or incorrect ID. |
| noFreeSpaceOfStoragePool | 0x60000014 | Storage pool is full. |
| riskPassword | 0x60000015 | Risky password. |
| UnSupportCapture | 0x60000016 | Capturing in 4096*2160 or 3072*2048 resolution is not supported when H.264+ is enabled. |
| userPwdLenUnder8 | 0x60000023 | At least two kinds of characters, including digits, letters, and symbols, should be contained in the password. |
| userPwdNameSame | 0x60000025 | Duplicated password. |
| userPwdNameMirror | 0x60000026 | The password cannot be the reverse order of user name. |
| beyondARGSRangeLimit | 0x60000027 | The parameter value is out of limit. |
| DetectionLineOutofDetectionRegion | 0x60000085 | The rule line is out of region. |

| Sub Status Code | Error Code | Description |
|------------------------------------|------------|---|
| DetectionRegionError | 0x60000086 | Rule region error. Make sure the rule region is convex polygon. |
| DetectionRegionOutOfCountingRegion | 0x60000087 | The rule region must be marked as red frame. |
| PedalAreaError | 0x60000088 | The pedal area must be in the rule region. |
| DetectionAreaABError | 0x60000089 | The detection region A and B must be in the a rule frame. |
| ABRegionCannotIntersect | 0x6000008a | Region A and B cannot be overlapped. |
| customHBPIDError | 0x6000008b | Incorrect ID of custom human body picture library |
| customHBPIDRepeat | 0x6000008c | Duplicated ID of custom human body picture library |
| dataVersionsInHBDLibMismatches | 0x6000008d | Database versions mismatches of human body picture library |
| invalidHBPID | 0x6000008e | Invalid human body picture PID |
| invalidHBDID | 0x6000008f | Invalid ID of human body picture library |
| humanLibraryError | 0x60000090 | Error of human body picture library |
| humanLibraryNumError | 0x60000091 | No more human body picture library can be added |
| humanImagesNumError | 0x60000092 | No more human body picture can be added |
| noHumanInThePicture | 0x60000093 | Modeling failed, no human body in the picture |
| analysisEnginesNoResourceError | 0x60001000 | No analysis engine. |
| analysisEnginesUsageExcced | 0x60001001 | The engine usage is overloaded. |

| Sub Status Code | Error Code | Description |
|----------------------------------|------------|--|
| PicAnalysisNoResourceError | 0x60001002 | No analysis engine provided for picture secondary recognition. |
| analysisEnginesLoadingError | 0x60001003 | Initializing analysis engine. |
| analysisEnginesAbnormaError | 0x60001004 | Analysis engine exception. |
| analysisEnginesFacelibImporting | 0x60001005 | Importing pictures to face picture library. Failed to edit analysis engine parameters. |
| analysisEnginesAssociatedChannel | 0x60001006 | The analysis engine is linked to channel. |
| smdEncodingNoResource | 0x60001007 | Insufficient motion detection encoding resources. |
| smdDecodingNoResource | 0x60001008 | Insufficient motion detection decoding resources. |
| diskError | 0x60001009 | HDD error. |
| diskFull | 0x6000100a | HDD full. |
| facelibDataProcessing | 0x6000100b | Handling face picture library data. |
| capturePackageFailed | 0x6000100c | Capturing packet failed. |
| capturePackageProcessing | 0x6000100d | Capturing packet. |
| noSupportWithPlaybackAbstract | 0x6000100e | This function is not supported. Playback by video synopsis is enabled. |
| insufficientNetworkBandwidth | 0x6000100f | Insufficient network bandwidth. |
| tapeLibNeedStopArchive | 0x60001010 | Stop the filing operation of tape library first. |
| identityKeyError | 0x60001011 | Incorrect interaction command. |
| identityKeyMissing | 0x60001012 | The interaction command is lost. |
| noSupportWithPersonDensityDetect | 0x60001013 | This function is not supported. The people density detection is enabled. |

| Sub Status Code | Error Code | Description |
|---------------------------------------|------------|--|
| ipcResolutionOverflow | 0x60001014 | The configured resolution of network camera is invalid. |
| ipcBitrateOverflow | 0x60001015 | The configured bit rate of network camera is invalid. |
| tooGreatTimeDifference | 0x60001016 | Too large time difference between device and server. |
| noSupportWithPlayback | 0x60001017 | This function is not supported. Playback is enabled. |
| channelNoSupportWithSMD | 0x60001018 | This function is not supported. Motion detection is enabled. |
| channelNoSupportWithFD | 0x60001019 | This function is not supported. Face capture is enabled. |
| illegalPhoneNumber | 0x6000101a | Invalid phone number. |
| illegalCertificateNumber | 0x6000101b | Invalid certificate No. |
| linkedCameraOutLimit | 0x6000101c | Connecting camera timed out. |
| achieveMaxChannelLimit | 0x6000101e | No more channels are allowed. |
| humanMisInfoFilterEnabledChanNumError | 0x6000101f | No more channels are allowed to enable preventing false alarm. |
| humanEnginesNoResource | 0x60001020 | Insufficient human body analysis engine resources. |
| taskNumberOverflow | 0x60001021 | No more tasks can be added. |
| collisionTimeOverflow | 0x60001022 | No more comparison duration can be configured. |
| invalidTaskID | 0x60001023 | Invalid task ID. |
| eventNotSupport | 0x60001024 | Event subscription is not supported. |
| invalidEZVIZSecretKey | 0x60001034 | Invalid verification code for Hik-Connect. |
| needDoubleVerification | 0x60001042 | Double verification required |
| noDoubleVerificationUser | 0x60001043 | No double verification user |

| Sub Status Code | Error Code | Description |
|--|------------|---|
| timeSpanNumOverLimit | 0x60001044 | Max. number of time buckets reached |
| channelNumOverLimit | 0x60001045 | Max. number of channels reached |
| noSearchIDResource | 0x60001046 | Insufficient searchID resources |
| noSupportDeleteStrangerLib | 0x60001051 | Deleting stranger library is not supported |
| noSupportCreateStrangerLib | 0x60001052 | Creating stranger library is not supported |
| behaviorAnalysisRuleInfoError | 0x60001053 | Behavior analysis rule parameters error. |
| safetyHelmetParamError | 0x60001054 | Hard hat parameters error. |
| OneChannelOnlyCanBindOneEngine | 0x60001077 | No more engines can be bound. |
| engineTypeMismatch | 0x60001079 | Engine type mismatched. |
| badUpgradePackage | 0x6000107A | Invalid upgrade package. |
| AudioFileNameDuplicate | 0x60001135 | Duplicated audio file name. |
| CurrentAudioFileAIRuleInUseAlreadyDelete | 0x60001136 | The AI rule linkage related to current audio file has been deleted. |
| TransitionUseEmmc | 0x60002000 | Starting device failed. The EMMC is overused. |
| AdaptiveStreamNotEnabled | 0x60002001 | The stream self-adaptive function is not enabled. |
| AdaptiveStreamAndVariableBitrateEnabled | 0x60002002 | Stream self-adaptive and variable bitrate function cannot be enabled at the same time. |
| noSafetyHelmetRegion | 0x60002023 | The hard hat detection area is not configured (if users save their settings without configuring the arming area, they should be prompted to configure one). |

| Sub Status Code | Error Code | Description |
|-------------------------------------|------------|---|
| unclosedSafetyHelmet | 0x60002024 | The hard hat detection is enabled (If users save their settings after deleting the arming area, they should be prompted to disable hard hat detection first and then delete the arming area). |
| width/ heightRatioOfPictureError | 0x6000202C | The width/height ratio of the uploaded picture should be in the range from 1:2 to 2:1. |
| PTZNotInitialized | 0x6000202E | PTZ is not initialized. |
| PTZSelfChecking | 0x6000202F | PTZ is self-checking. |
| PTZLocked | 0x60002030 | PTZ is locked. |
| advancedParametersError | 0x60002031 | Auto-switch interval in advanced parameters cannot be shorter than parking tolerance for illegal parking detection in speed dome rule settings. |
| resolutionError | 0x60005003 | Invalid resolution |
| deployExceedMax | 0x60006018 | The arming connections exceed the maximum number. |
| detectorTypeMismatch | 0x60008000 | The detector type mismatched. |
| nameExist | 0x60008001 | The name already exists. |
| uploadImageSizeError | 0x60008016 | The size of the uploaded picture is larger than 5 MB. |
| laneAndRegionOverlap | / | The lanes are overlapped. |
| unitConfigurationNotInEffect | / | Invalid unit parameter. |
| ruleAndShieldingMaskConflict | / | The line-rule region overlaps with the shielded area. |
| wholeRuleInShieldingMask | / | There are complete temperature measurement rules in the shielded area. |

| Sub Status Code | Error Code | Description |
|--|------------|--|
| LogDiskNotSetReadOnlyInGroupMode | 0x60001100 | The log HDD in the HDD group cannot be set to read-only. |
| LogDiskNotSetRedundancyInGroupMode | 0x60001101 | The log HDD in the HDD group cannot be set to redundancy. |
| holidayNameContainChineseOrSpecialChar | 0x60001080 | No Chinese and special characters allowed in holiday name. |
| genderValueError | 0x60001081 | Invalid gender. |
| certificateTypeValueError | 0x60001082 | Invalid identification type. |
| personInfoExtendValuesTooLong | 0x60001083 | The length of customized tags exceeds limit. |
| personInfoExtendValueContainsInvalidChar | 0x60001084 | Invalid characters are not allowed in customized tags of the face picture library. |
| excelHeaderError | 0x60001085 | Excel header error. |
| intelligentTrafficMutexWithHighFrames | 0x60008014 | Please disable all functions of traffic incident detection, violation enforcement, and traffic data collection, or adjust the video frame rate to that lower than 50 fps. |
| intelligentTrafficMutexWithHighFramesEx | 0x60008018 | Please disable all functions of traffic incident detection, violation enforcement, traffic data collection, and vehicle detection, or adjust the video frame rate to that lower than 50 fps. |

StatusCode=7

| SubStatusCode | Error Code | Description |
|----------------|------------|------------------------|
| rebootRequired | 0x70000001 | Reboot to take effect. |

