Communication Protocol of Double Light Source System

Version <1.1.0>

Contents

I. History of Revisions

Date	Operation	Author	Version number	Content	Remarks
2021.01.27	Add	Yang Neng	V1.0.0	Added	
2021.02.03	Revise	Yang Neng	V1.0.1	1 Noise reduction and detail support are added to infrared image related interfaces. 2 The value range of the image setting interface is clearly increased.	
2021.02.05	Revise	Yang Neng	V1.0.2	1 The temperature measuring gear is removed from the rule, the temperature measuring gear selection is added in the global settings, and the temperature measuring gear range is added in the public information query. 2 The upload data interface is revised, k and curve upload are removed, and overall temperature measurement package upload is supported. 3 Report return supports null interval. 4 Device alarm supports stream 0.	
2021.02.25	Revise	Yang Neng	v1.0.3	1 rtsp related options are added for general query interfaces. 2 The binary return is removed from the infrared screenshot acquisition interface and changed into base64 format.	
2021.03.02	Revise	Yang Neng	v1.0.4	1 Some interface spelling errors are revised. 2 The platform access interface is added.	
2021.03.09	Revise	Yang Neng	v1.0.5	1 In the return interface, retcode retmsg is placed to the outermost layer of json. 2 The field content and redundancy are removed from the return interface.	
2021.03.20	Revise	Yang Neng	v1.0.6	1 The 80x factory setting interface is added.	
2021.05.21	Revise	Yang Neng	v1.0.7	1 The detector information and infrared output information are added in the general acquisition interfaces. 2 The NUC interface and the temperature matrix acquisition interface are added. 3 Description related with coordinates is corrected. 4 The field system is added in the storage information interface. 5 The integrated management platform interface (from lxp) is added. 6 The description on content is removed from partial interfaces.	
2021.06.01	Revise	Yang Neng	v1.0.7	1 The IP setting error code is added.	
2021.06.10	Revis	Song	V1.0.8	1. K and detector parameter configuration	

	e	Danda n		interface (907,910) are added. 2. The return value of the temperature calibration parameter interface (901) is updated.	
2021.06.15	Revis e	Song Danda n	V1.0.8	1. Detector parameter configuration interface (907) 2. The return value of the temperature calibration parameter interface (901) is updated.	
2021.06.22	Revis e	Yang Neng	V1.0.9	1. The http upload file interface is added. 2. The 519 data upload function interface is deleted.	
2021.07.27	Revis e	Song Danda n	V1.0.10	1. The automatic temperature calibration interface 911 is added. 2. The automatic temperature calibration result parameter interface 912 is added. 3. The screenshot interface 523 is revised, and the original scale image type is increased. 4. The temperature matrix acquisition interface 530 is revised, and the image resolution field type is added. 5. The interface 913 for acquiring data such as Y16 and X16 is added, and the data such as Y16 is directly returned.	
2021.08.02	Revis e	Yang Neng	V1.0.11	1. Redundant interfaces on the platform are deleted. 2. The platform interface is revised, and multiple platforms are supported.	
2021.08.27	Add	Song Danda n	V1.0.12	1. The interface for saving Y16 is added to return data Y16 in parameters. 2. The function interface of the lid is added.	
2021.09.09	Add	Song Danda n	V1.0.13	801802 The lid field uniformswitch is added (the lid algorithm switch is removed) .	
2021.09.23	Revis e	Yang Neng	V1.0.14	1 The external output control and alarm related API are added. 2 The external input support is added.	
2021.09.26	Revis e	Yang Neng	V1.0.15	1 The external input notification message 1009 is added.	
2021.11.10	Revis e	Yang Neng	V1.0.16	1 The FTP configuration is added.	
2021.11.17	Revis e	Yang Neng	V1.0.17	1 Temperature difference and temperature slope alarm is supported. 2 The fire configuration is added. 3 The automatic reset interface is added.	
2022.04.11	Revis e	Yang Neng	V1.1.0	The document is re-arranged.	

II. Description

This Protocol is mainly to develop a set of general protocol for the communication mode used by the client and the server during the development of the card machine system and the content of the communication protocol, so as to facilitate the communication between the client and the server.

1.Keyword definition

Server: Server (mainly used for data collection, processing, alarm, message forwarding and other functions)

Client: Client (mainly used for application display, dynamic setting of alarm parameters, etc.)

Characters: all refers to English characters, and Chinese characters are in UTF-8 format.

2.Communication rules and codes

(1) This Protocol supports websocket/http data interactive communication;

(2) The application layer data is encoded by UTF-8 and in json format;

3.Data security

Currently, only the password is encrypted, and the upgrade data is verified for completeness.

4.Coordinate range

The coordinate range is related to the infrared output image. In this Protocol, the resolution of the infrared image 512x 384 is taken as an example, that is, unless otherwise specified, the coordinate is within the range of 512x 384, and the range of 512 or 384 is the width or height of the infrared output.

5.Use process

See (Appendix IV) for Websocket communication process:

(1) Obtain general information through the general interface (id = 500). It is optional.

(2) Obtain TOKEN through the login interface (id = 501). It is required.

(3) Send heartbeats (id = 9999) at intervals, ensure TOKEN is available, and keep alive for 2 minutes. It is required.

(4) Use business API according to business requirements.

(5) Log out through the logout interface (id = 502).
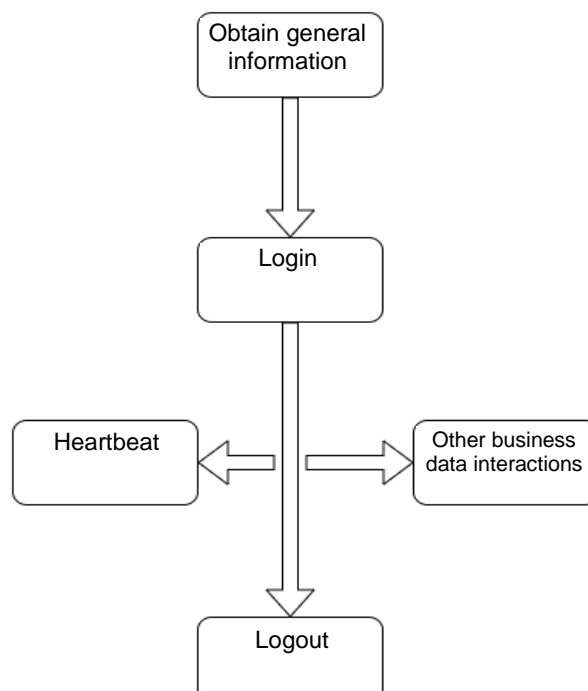
HTTP communication (Appendix V):

The http interface supports login-free communication, and HEAD adds header verification.

Keep the http login mode at the same time (if there is no http verification field, use the http login mode)

The format of http login-free request is as follows (see Appendix 5 for an example):

URI：/api/v1/msg

Method: POST

Signed: Sha256(Method + "\n" + URI+ "\n" + ExpiredTime + "\n" + AppKey + "\n" + Algorithm + "\n" + Sha256(CONTENT))

ExpiredTime:

Algorithm:

AppKey:

CONTENT: JSON

The format of the http login request is as follows (see Appendix 5 for an example):

URI:/api/v1/msg

Method: POST

CONTENT: JSON

6. Description of the Protocol

This Protocol includes the docking of the IMIS platform, in which the from and to fields need to be added to the header (the same level of action) during the docking process.

Field Name	Type	Options	Description
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.

III. Configuration of API interfaces

The configuration of API interface commands starts from 1.

1. Setting of global temperature measurement parameters

1.1 Field description

Field Name	Type	Options	Description
action	string	Required	request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message. response: It refers to the response message.
cmdtype	int	Required	1 Set temperature measurement parameters.
sequence	uint64	Required	Message serial number
token	string	Required	User access token, user identification
message	json	Optional	Message body node, defining the request message
retcode	int	Required, return value	0 succeeded, otherwise failed. See attachment for error code.
retmsg	string	Required, return value	Return message
flag	int	Optional	Switch 0 off 1 on
show_mouse	int	Optional	Click on the Web preview page to show the temperature, 0: Do not show 1: Show
show_mode	int	Optional	Temperature show mode, 1 Max Temp 2 Min Temp 3 Ave Temp 4 Max Temp + Min Temp 5 Max Temp + Ave Temp 6 Ave Temp + Min Temp 7 Max Temp + Min Temp + Ave Temp 8 Do not show
emiss_mode	int	Optional	Emissivity type 1 Standard Water (0.96) Stainless steel (0.14) Aluminum plate (0.09) Black aluminum (0.95) Asphalt (0.96) Black paper (0.86) Concrete (0.97) Cast iron (0.81) Gypsum (0.75) Rubber (0.95) Wood (0.85)

			Brick (0.75)
			Tape (0.96)
			Copper plate (0.06)
			Human skin (0.98)
			PVC plastic (0.93)
			Polycarbonate (0.8)
			Copper oxide (0.78)
			Rust (0.8)
			Paint (0.9)
			Soil (0.93)
			2 Custom
emiss	float	Optional	Emissivity 0.1~1.0
humi	int	Optional	Humidity, range 1~100
dist	float	Optional	Distance, in meters, 0.1~200.0
ref_temp	float	Optional	Reflection temperature -40~2,000 degrees Celsius
atmo_trans	float	Optional	Atmospheric transmittance 0.01~1
opti_trans	float	Optional	Optical transmittance 0.01~1
color_bar	int	Optional	Color band 1 WhiteHot 2 BlackHot 3 IronRed 4 Fulgurite 5 HotIron 6 Medical 7 Medical0 8 Medical1 9 Medical2 10 Arctic 11 BodyDRT 12 RainBow1 13 RainBow2 14 RainBow3 15 Tint1 16 Tint2 17 Tint3 18 IceFire 19 ReverseIceFire 20 Steps 21 Sepia 22 GreenHot 23 BlueHot 24 Winter 25 HighLight 26 Amber

color_show	int	Optional	Show color band in the heat map or not 0 No 1 Yes
mod_temp	float	Optional	Temperature correction, Range -1~1
show_string	int	Optional	Whether to use character overlay 1 Off 2 Upper left 3 Upper right 4 Lower left 5 Lower right
show_desc	string	Optional	Character overlay content, and the number of characters ranges from 0~42
isot_flag	int	Optional	Isotherm switch, 0 Off 1 On
isot_type	int	Optional	Type: 1 Turn off isotherm effect 2 Turn on the high isotherm 3 Turn on the low isotherm 4 Turn on the isotherm in the interval 5 Turn on the isotherm outside the interval
isot_high	float	Optional	High temperature threshold -40~2,000,
isot_low	float	Optional	Low temperature threshold -40~2,000,
isot_high_color	string	Optional	High temperature color string# RGB E.g., Red:#ff0000 Default: #ff0000
isot_low_color	string	Optional	Low temperature color string# RGB E.g., Red:#ff0000 Default: #0000ff
gear	int	Optional	For temperature measurement gear index, see the 500 interface for the number of gears currently supported. The value is synchronized with the supported gear, and the value is the gear index.

1.2 Requests and replies

Request:

```
{
    "action": "request",
    "cmdtype": 1,
    "token": "testtoken",
    "message": {
        "atmo_trans": 1,
        "color_bar": 3,
        "color_show": 1,
```

```
        "dist": 0.5,
        "emiss": 0.95,
        "emiss_mode": 1,
        "flag": 1,
        "gear": 0,
        "humi": 50,
        "isot_flag": 0,
        "isot_high": 40,
        "isot_high_color": "#ff0000",
        "isot_low": 10,
        "isot_low_color": "#00ff00",
        "isot_type": 1,
        "mod_temp": 0,
        "opti_trans": 1,
        "ref_temp": 23,
        "show_desc": "",
        "show_mode": 1,
        "show_mouse": 1,
        "show_string": 1
    },
    "sequence": 1
}
```

Reply:

```
{
    "action": "response",
    "cmdtype": 1,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

2. Acquisition of global temperature measurement parameters

2.1 Field description

Field Name	Type	Options	Description
action	string	Required	request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message. response: It refers to the response message.
cmdtype	int	Required	2 Acquire temperature measurement parameters.

sequence	uint64	Required	Message serial number
token	string	Required	User access token, user identification
message	json	Optional	Message body node, defining the request message
retcode	int	Required, return value	0 succeeded, otherwise failed. See attachment for error code.
retmsg	string	Required, return value	Return message
flag	int	Optional	Switch 0 off 1 on
show_mouse	int	Optional	Click on the Web preview page to show the temperature, 0: Do not show 1: Show
show_mode	int	Optional	Temperature show mode, 1 Max Temp 2 Min Temp 3 Ave Temp 4 Max Temp + Min Temp 5 Max Temp + Ave Temp 6 Ave Temp + Min Temp 7 Max Temp + Min Temp + Ave Temp 8 Do not show
emiss_mode	int	Optional	Emissivity type 1 Standard Water (0.96) Stainless steel (0.14) Aluminum plate (0.09) Black aluminum (0.95) Asphalt (0.96) Black paper (0.86) Concrete (0.97) Cast iron (0.81) Gypsum (0.75) Rubber (0.95) Wood (0.85) Brick (0.75) Tape (0.96) Copper plate (0.06) Human skin (0.98) PVC plastic (0.93) Polycarbonate (0.8) Copper oxide (0.78) Rust (0.8) Paint (0.9)

			Soil (0.93)
			2 Custom
emiss	float	Optional	Emissivity 0.1~1.0
humi	int	Optional	Humidity, range 1~100
dist	float	Optional	Distance, in meters, 0.1~200.0
ref_temp	float	Optional	Reflection temperature -40~2,000 degrees Celsius
atmo_trans	float	Optional	Atmospheric transmittance 0.01~1
opti_trans	float	Optional	Optical transmittance 0.01~1
color_bar	int	Optional	Color band 1　　　WhiteHot 2　　　BlackHot 3　　　IronRed 4　　　Fulgurite 5　　　HotIron 6　　　Medical 7　　　Medical0 8　　　Medical1 9　　　Medical2 10　　　Arctic 11　　　BodyDRT 12　　RainBow1 13　　　RainBow2 14　　RainBow3 15　　　Tint1 16　　Tint2 17　　　Tint3 18　　IceFire 19　　ReverseIceFire 20　　　Steps 21　　　Sepia 22　　　GreenHot 23　　　BlueHot 24　　　Winter 25　　　HighLight 26　　　Amber
color_show	int	Optional	Show color band in the heat map or not 0 No 1 Yes
mod_temp	float	Optional	Temperature correction, Range -1~1
show_string	int	Optional	Whether to use character overlay 1 Off 2 Upper left 3 Upper right 4 Lower left

			5 Lower right
show_desc	string	Optional	Character overlay content, and the number of characters ranges from 0~42
isot_flag	int	Optional	Isotherm switch, 0 Off 1 On
isot_type	int	Optional	Type: 1 Turn of isotherm effect 2 Turn on the high isotherm 3 Turn on the low isotherm 4 Turn on the isotherm in the interval 5 Turn on the isotherm outside the interval
isot_high	float	Optional	High temperature threshold -40~2,000,
isot_low	float	Optional	Low temperature threshold -40~2,000,
isot_high_color	string	Optional	High temperature color string# RGB E.g., Red:#ff0000 Default: #ff0000
isot_low_color	string	Optional	Low temperature color string# RGB E.g., Red:#ff0000 Default: #0000ff
gear	int	Optional	For temperature measurement gear index, see the 500 interface for the number of gears currently supported. The value is synchronized with the supported gear, and the value is the gear index.
default	int	Optional	Default parameter or not 0 No 1 Yes

2.2 Requests and replies

Request:

```
{
    "action": "request",
    "cmdtype": 2,
    "sequence": 1,
    "token": "testtoken",
    "message": {"default":0 }
}
```

Reply:

```
{
    "action": "response",
    "cmdtype": 2,
    "message": {
```

```json
        "atmo_trans": 1,
        "color_bar": 3,
        "color_show": 1,
        "dist": 0.5,
        "emiss": 0.95,
        "emiss_mode": 1,
        "flag": 1,
        "gear": 0,
        "humi": 50,
        "isot_flag": 0,
        "isot_high": 40,
        "isot_high_color": "#ff0000",
        "isot_low": 10,
        "isot_low_color": "#00ff00",
        "isot_type": 1,
        "mod_temp": 0,
        "opti_trans": 1,
        "ref_temp": 23,
        "show_desc": "",
        "show_mode": 1,
        "show_mouse": 1,
        "show_string": 1
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

3. Setting of temperature measurement rules

3.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message. response: It refers to the response message.
cmdtype	int	Required	3 Set rule objects
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json	Required	Message body node, defining the request message
show_type	int	Optional	Object temperature show, 1 Max Temp

			2 Min Temp 3 Ave Temp 4 Name only 5 Do not show
rule_list	json	Optional	Rule list
rule_name	string	Required	Rule name, and the number of characters ranges from 0~20
show_location	int	Required	Name show location: 1 Above 2 Below 3 Left 4 Right 5 Middle
type	int	Required	Object type: 1 Point 2 Wire 3 Rectangle 4 Polygon 5 Circular
points	json	Required	Point collection The coordinates of the object point are based on the video output size, with a maximum of 7 points, i.e. upper left (1,1) and lower right (w,h): Point object: the first point is required Wire object: two points are required Rectangular object: 4 points are required Polygon: points are saved as needed, up to 7 points Circular: 4 points are required, top, bottom, left and right. Format {"x": 12, "y": 12}
x	int	Required	x coordinates range from 1-512 and refer to infrared image
y	int	Required	y coordinates range from 1-384 and refer to infrared image
alarm_flag	int	Required	Enable alarm or not, 0 No 1 Yes
high_temp	float	Required	Alarm high temperature threshold,-40~2,000,
low_temp	float	Required	Alarm low temperature threshold,-40~2,000,
diff_temp	float	Required	Temperature difference, 0~2,000,
rate_temp	float	Required	Temperature slope,-600~600
alarm_type	int	Required	Alarm type 1 High temperature alarm 2 Low temperature alarm 3 Average temperature alarm 4 High and low temperature alarm 5 Temperature difference

			6 Temperature slope
alarm_condition	int	Required	Alarm conditions, 1 Above 2 Below 3 Matching 4 Above and Below
temp_mod	float	Required	Temperature error, 0~1
alarm_time	int	Required	Duration, 0~1,000
avg_temp	float	Required	Ave Temp, -40~2,000
emiss_mode	int	Required	Emissivity type 1 Standard Water (0.96) Stainless steel (0.14) Aluminum plate (0.09) Black aluminum (0.95) Asphalt (0.96) Black paper (0.86) Concrete (0.97) Cast iron (0.81) Gypsum (0.75) Rubber (0.95) Wood (0.85) Brick (0.75) Tape (0.96) Copper plate (0.06) Human skin (0.98) PVC plastic (0.93) Polycarbonate (0.8) Copper oxide (0.78) Rust (0.8) Paint (0.9) Soil (0.93) 2 Custom
emiss	float	Required	Emissivity 0.1~1.0
humi	int	Required	Humidity, range 1~100
dist	float	Required	Distance, in meters, 0.1~200.0
ref_temp	float	Required	Reflection temperature -40~2,000 degrees Celsius
atmo_trans	float	Required	Atmospheric transmittance 0.01~1
opti_trans	float	Required	Optical transmittance 0.01~1
preset	int	Optional	Preset bit, subject to actual equipment support, is the object attribute

			Range [1,256]

3.2 Requests and replies

Request:

```
{
    "action": "request",
    "cmdtype": 3,
    "token": "testtoken",
    "message": {
        "rule_list": [
            {
                "alarm_condition": 1,
                "alarm_flag": 0,
                "alarm_time": 30,
                "alarm_type": 1,
                "atmo_trans": 1,
                "avg_temp": 30,
                "diff_temp": 0,
                "dist": 0.5,
                "emiss": 0.961,
                "emiss_mode": 1,
                "flag": 0,
                "high_temp": 60,
                "humi": 50,
                "id": 0,
                "low_temp": 10,
                "opti_trans": 1,
                "preset": 1,
                "points": [
                    {
                        "x": 133,
                        "y": 73
                    },
                    {
                        "x": 261,
                        "y": 73
                    },
                    {
                        "x": 261,
                        "y": 133
```

```json
                },
                {
                    "x": 133,
                    "y": 133
                }
            ],
            "rate_temp": 0,
            "ref_temp": 23,
            " rule_name ": "Rectangle 1",
            "show_location": 1,
            "temp_mod": 0.2,
            "type": 3
        },
        {
            "alarm_condition": 1,
            "alarm_flag": 0,
            "alarm_time": 30,
            "alarm_type": 1,
            "atmo_trans": 1,
            "avg_temp": 30,
            "diff_temp": 0,
            "dist": 0.5,
            "emiss": 0.961,
            "emiss_mode": 1,
            "flag": 0,
            "high_temp": 60,
            "humi": 50,
            "id": 0,
            "low_temp": 10,
            "opti_trans": 1,
            "points": [
                {
                    "x": 305,
                    "y": 21
                },
                {
                    "x": 389,
                    "y": 21
                },
                {
```

```json
                    "x": 389,
                    "y": 41
                },
                {
                    "x": 305,
                    "y": 41
                }
            ],
            "rate_temp": 0,
            "ref_temp": 23,
            " rule_name ": "Rectangular 234",
            "show_location": 1,
            "temp_mod": 0.2,
            "type": 3
        },
        {
            "alarm_condition": 1,
            "alarm_flag": 0,
            "alarm_time": 30,
            "alarm_type": 1,
            "atmo_trans": 1,
            "avg_temp": 30,
            "diff_temp": 0,
            "dist": 0.5,
            "emiss": 0.961,
            "emiss_mode": 1,
            "flag": 0,
            "high_temp": 60,
            "humi": 50,
            "id": 0,
            "low_temp": 10,
            "opti_trans": 1,
            "points": [
                {
                    "x": 289,
                    "y": 335
                },
                {
                    "x": 389,
                    "y": 335
```

```
                        },
                        {
                            "x": 389,
                            "y": 366
                        },
                        {
                            "x": 289,
                            "y": 366
                        }
                    ],
                    "rate_temp": 0,
                    "ref_temp": 23,
                    " rule_name ": "Rectangle 3",
                    "show_location": 1,
                    "temp_mod": 0.2,
                    "type": 3
                }
            ],
            "show_type": 1
        },
    "sequence": 1
}
```

Reply:

```
{
    "action": "response",
    "cmdtype": 3,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

4. Obtaining of temperature measurement rules

4.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message. response: It refers to the response message.
cmdtype	int	Required	4 Obtain rule objects

sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json	Required	Message body node, defining the request message
show_type	int	Optional	Object temperature show, 1 Max Temp 2 Min Temp 3 Ave Temp 4 Name only 5 Do not show
rule_list	json	Optional	Rule list
rule_name	string	Required	Rule name, and the number of characters ranges from 0~20
show_location	int	Required	Name show location: 1 Above 2 Below 3 Left 4 Right 5 Middle
type	int	Required	Object type: 1 Point 2 Wire 3 Rectangle 4 Polygon 5 Circular
points	json	Required	Point collection The coordinates of the object point are based on the video output size, with a maximum of 7 points, i.e. upper left (1,1) and lower right (w,h): Point object: the first point is required Wire object: two points are required Rectangular object: 4 points are required Polygon: points are saved as needed, up to 7 points Circular: 4 points are required, top, bottom, left and right. Format {"x": 12, "y": 12}
x	int	Required	x coordinates range from 1-512 and refer to infrared image
y	int	Required	y coordinates range from 1-384 and refer to infrared image
alarm_flag	int	Required	Enable alarm or not, 0 No 1 Yes
high_temp	float	Required	Alarm high temperature threshold,-40~2,000,
low_temp	float	Required	Alarm low temperature threshold,-40~2,000,
diff_temp	float	Required	Temperature difference, 0~2,000,
rate_temp	float	Required	Temperature slope,-600~600

alarm_type	int	Required	Alarm type 1 High temperature alarm 2 Low temperature alarm 3 Average temperature alarm 4 High and low temperature alarm 5 Temperature difference 6 Temperature slope
alarm_condition	int	Required	Alarm conditions, 1 Above 2 Below 3 Matching 4 Above and Below
temp_mod	float	Required	Temperature error, 0~1
alarm_time	int	Required	Duration, 0~1,000
avg_temp	float	Required	Ave Temp, -40~2,000
emiss_mode	int	Required	Emissivity type 1 Standard Water (0.96) Stainless steel (0.14) Aluminum plate (0.09) Black aluminum (0.95) Asphalt (0.96) Black paper (0.86) Concrete (0.97) Cast iron (0.81) Gypsum (0.75) Rubber (0.95) Wood (0.85) Brick (0.75) Tape (0.96) Copper plate (0.06) Human skin (0.98) PVC plastic (0.93) Polycarbonate (0.8) Copper oxide (0.78) Rust (0.8) Paint (0.9) Soil (0.93) 2 Custom
emiss	float	Required	Emissivity 0.1~1.0
humi	int	Required	Humidity, range 1~100
dist	float	Required	Distance, in meters, 0.1~200.0

ref_temp	float	Required	Reflection temperature -40~2,000 degrees Celsius
atmo_trans	float	Required	Atmospheric transmittance 0.01~1
opti_trans	float	Required	Optical transmittance 0.01~1
preset	int	Optional	Preset bit, subject to actual equipment support, is the object attribute Range [1,256]
default	int	Optional	Default parameter or not 0 No 1 Yes

4.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 4,
    "token": "testtoken",
    "message": { },
    "sequence": 1
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 4,
    "message": {
        "rule_list": [
            {
                "alarm_condition": 1,
                "alarm_flag": 0,
                "alarm_time": 30,
                "alarm_type": 1,
                "atmo_trans": 1,
                "avg_temp": 30,
                "diff_temp": 0,
                "dist": 0.5,
                "emiss": 0.961,
                "emiss_mode": 1,
                "flag": 0,
                "high_temp": 60,
                "humi": 50,
                "id": 0,
```

```json
        "low_temp": 10,
        "opti_trans": 1,
        "points": [
            {
                "x": 133,
                "y": 73
            },
            {
                "x": 261,
                "y": 73
            },
            {
                "x": 261,
                "y": 133
            },
            {
                "x": 133,
                "y": 133
            }
        ],
        "rate_temp": 0,
        "ref_temp": 23,
        "rule_name": "Rectangle 1",
        "show_location": 1,
        "temp_mod": 0.2,
        "type": 3
    },
    {
        "alarm_condition": 1,
        "alarm_flag": 0,
        "alarm_time": 30,
        "alarm_type": 1,
        "atmo_trans": 1,
        "avg_temp": 30,
        "diff_temp": 0,
        "dist": 0.5,
        "emiss": 0.961,
        "emiss_mode": 1,
        "flag": 0,
        "high_temp": 60,
```

        "humi": 50,
        "id": 0,
        "low_temp": 10,
        "opti_trans": 1,
        "points": [
            {
                "x": 305,
                "y": 21
            },
            {
                "x": 389,
                "y": 21
            },
            {
                "x": 389,
                "y": 41
            },
            {
                "x": 305,
                "y": 41
            }
        ],
        "rate_temp": 0,
        "ref_temp": 23,
        "rule_name": "Rectangle 2",
        "show_location": 1,
        "temp_mod": 0.2,
        "type": 3
    },
    {
        "alarm_condition": 1,
        "alarm_flag": 0,
        "alarm_time": 30,
        "alarm_type": 1,
        "atmo_trans": 1,
        "avg_temp": 30,
        "diff_temp": 0,
        "dist": 0.5,
        "emiss": 0.961,
        "emiss_mode": 1,

```json
            "flag": 0,
            "high_temp": 60,
            "humi": 50,
            "id": 0,
            "low_temp": 10,
            "opti_trans": 1,
            "points": [
                {
                    "x": 289,
                    "y": 335
                },
                {
                    "x": 389,
                    "y": 335
                },
                {
                    "x": 389,
                    "y": 366
                },
                {
                    "x": 289,
                    "y": 366
                }
            ],
            "rate_temp": 0,
            "ref_temp": 23,
            "rule_name": "Rectangle 3",
            "show_location": 1,
            "temp_mod": 0.2,
            "type": 3
        }
    ],
    "show_type": 1
},
"retcode": 0,
"retmsg": "success.",
"sequence": 1
}
```

5. Setting of image effect parameters

5.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	5 Set image effect parameters
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
vl_config	json object	Optional	Visible light image configuration
ir_config	json object	Optional	Infrared image configuration
brightness	int	Optional	Brightness, value range 0~100,
contrast	int	Optional	Contrast, value range ~ 100
saturation	int	Optional	Saturation, value range 0~100
sharpness	int	Optional	Sharpness, value range 0~100
reverse	int	Optional	Reverse or not, 0: No 1 Yes
auto_shutter	int	Optional	Automatic shutter compensation time 3~1,200,
blc	int	Optional	Backlight compensation, 0: Off 1: Up 2 Down 3 Left 4 Right 5.Middle 6 Automatic
exp	int	Optional	Exposure compensation: 0~100,
hlc	int	Optional	Strong light suppression, 0 Off 1 On
wdr	int	Optional	Wide dynamic, 0: Off 1:20% 2:40% 3:60% 4:80% 5:100%
time_flag	int	Optional	Noise reduction time domain filter switch 0 Off 1 On
time_value	int	Optional	Noise reduction time domain filter value 0~100

space_flag	int	Optional	Noise reduction airspace filter switch 0 Off 1 On
space_value	int	Optional	Noise reduction spatial filtering value 0~100
iee_flag	int	Optional	Detail enhancement switch 0 Off 1 On
iee_value	int	Optional	Detail enhancement value 0~100,
retcode	int	Optional	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Optional	Return message

5.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 5,
    "token": "testtoken",
    "message": {
        "ir_config": {
            "auto_shutter": 30,
            "brightness": 40,
            "contrast": 55,
            "iee_flag": 0,
            "iee_value": 66,
            "reverse": 0,
            "saturation": 50,
            "sharpness": 34,
            "space_flag": 1,
            "space_value": 50,
            "time_flag": 1,
            "time_value": 50
        },
        "vl_config": {
            "blc": 5,
            "brightness": 50,
            "contrast": 50,
            "exp": 0,
            "hlc": 1,
            "reverse": 0,
            "saturation": 50,
```

```
        "sharpness": 50,
        "wdr": 5
    }
},
"sequence": 1
}
```
Reply:
```
{
    "action": "response",
    "cmdtype": 5,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

6. Acquisition of image effect parameters

6.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	6 Acquire image effect parameters
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
vl_config	json object	Optional	Visible light image configuration
ir_config	json object	Optional	Infrared image configuration
brightness	int	Optional	Brightness, value range 0~100,
contrast	int	Optional	Contrast, value range ~ 100
saturation	int	Optional	Saturation, value range 0~100
sharpness	int	Optional	Sharpness, value range 0~100
reverse	int	Optional	Reverse or not, 0: No 1 Yes
auto_shutter	int	Optional	Automatic shutter compensation time 3~1,200,
blc	int	Optional	Backlight compensation, 0: Off 1: Up 2 Down 3 Left

			4 Right 5.Middle 6 Automatic
exp	int	Optional	Exposure compensation: 0~100,
hlc	int	Optional	Strong light suppression, 0 Off 1 On
wdr	int	Optional	Wide dynamic, 0: Off 1:20% 2:40% 3:60% 4:80% 5:100%
time_flag	int	Optional	Noise reduction time domain filter switch 0 Off 1 On
time_value	int	Optional	Noise reduction time domain filter value 0~100
space_flag	int	Optional	Noise reduction airspace filter switch 0 Off 1 On
space_value	int	Optional	Noise reduction spatial filtering value 0~100
iee_flag	int	Optional	Detail enhancement switch 0 Off 1 On
iee_value	int	Optional	Detail enhancement value 0~100,
retcode	int	Optional	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Optional	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

6.2 Requests and replies

Request:

```
{
    "action": "request",
    "cmdtype": 6,
    "token": "testtoken",
    "message": {},
    "sequence": 1
}
```

Reply:

```
{
```

```json
    "action": "response",
    "cmdtype": 6,
    "message": {
        "ir_config": {
            "auto_shutter": 30,
            "brightness": 40,
            "contrast": 55,
            "iee_flag": 0,
            "iee_value": 66,
            "reverse": 0,
            "saturation": 50,
            "sharpness": 34,
            "space_flag": 1,
            "space_value": 50,
            "time_flag": 1,
            "time_value": 50
        },
        "vl_config": {
            "blc": 5,
            "brightness": 50,
            "contrast": 50,
            "exp": 0,
            "hlc": 1,
            "reverse": 0,
            "saturation": 50,
            "sharpness": 50,
            "wdr": 5
        }
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

7. Setting of image fusion

7.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message

			response: It refers to the response message
cmdtype	int	Required	7 Set image fusion parameters
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
percent	int	Optional	Fusion ratio value 0~100
vl_match_points	json object	Optional	Visible light calibration point
ir_match_points	json object	Optional	Infrared calibration point
point1	json object	Required	1st calibration point
point2	json object	Required	2nd calibration point
point3	json object	Required	3rd calibration point
point4	json object	Required	4th calibration point
point5	json object	Required	5th calibration point
x	int	Required	X coordinates 1~512
y	int	Required	y coordinates 1~384
retcode	int	Return to required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Return to required	Return message

7.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 7,
    "token": "testtoken",
    "message": {
        "ir_match_points": {
            "point1": {
                "x": 86,
                "y": 181
            },
            "point2": {
                "x": 265,
                "y": 53
            },
            "point3": {
                "x": 416,
                "y": 179
            },
            "point4": {
```

```json
                "x": 268,
                "y": 346
            },
            "point5": {
                "x": 425,
                "y": 341
            }
        },
        "percent": 50,
        "vl_match_points": {
            "point1": {
                "x": 120,
                "y": 156
            },
            "point2": {
                "x": 261,
                "y": 19
            },
            "point3": {
                "x": 380,
                "y": 164
            },
            "point4": {
                "x": 261,
                "y": 340
            },
            "point5": {
                "x": 397,
                "y": 343
            }
        }
    },
    "sequence": 1
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 7,
    "message": null,
    "retcode": 0,
```

```
    "retmsg": "success.",
    "sequence": 1
}
```

8. Acquisition of image fusion

8.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	8 Acquire image fusion parameters
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
percent	int	Optional	Fusion ratio value 0~100
vl_match_points	json object	Optional	Visible light calibration point
ir_match_points	json object	Optional	Infrared calibration point
point1	json object	Required	1st calibration point
point2	json object	Required	2nd calibration point
point3	json object	Required	3rd calibration point
point4	json object	Required	4th calibration point
point5	json object	Required	5th calibration point
x	int	Required	X coordinates 1~512
y	int	Required	y coordinates 1~384
retcode	int	Return to required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Return to required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

8.2 Requests and replies

Request:

```
{
    "action": "request",
    "cmdtype": 8,
    "token": "testtoken",
    "message": {},
    "sequence": 1
```

```
        }
Reply:
{
    "action": "response",
    "cmdtype": 8,
    "message": {
        "ir_match_points": {
            "point1": {
                "x": 86,
                "y": 181
            },
            "point2": {
                "x": 265,
                "y": 53
            },
            "point3": {
                "x": 416,
                "y": 179
            },
            "point4": {
                "x": 268,
                "y": 346
            },
            "point5": {
                "x": 425,
                "y": 341
            }
        },
        "percent": 50,
        "vl_match_points": {
            "point1": {
                "x": 120,
                "y": 156
            },
            "point2": {
                "x": 261,
                "y": 19
            },
            "point3": {
                "x": 380,
```

```
                "y": 164
            },
            "point4": {
                "x": 261,
                "y": 340
            },
            "point5": {
                "x": 397,
                "y": 343
            }
        }
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

9.Network information setting

9.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	9 Set network information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
host_name	string	Optional	Host name, character range: 0~15
ipaddr	string	Optional	Network ip address xxx.xxx.xxx.xxx, Conform to ip format Default 192.168.1.168
netmask	string	Optional	Subnet mask xxx.xxx.xxx.xxx is consistent with IP authentication
gateway	string	Optional	Gateway xxx.xxx.xxx.xxx has the same network segment with ip and mask
dns1	string	Optional	dns server xxx.xxx.xxx.xxx
dns2	string	Optional	dns server xxx.xxx.xxx.xxx
card	int	Optional	NIC type: 0 limited NIC
mode	int	Optional	Mode 0 Static 1 Dynamic
ip_version	int	Optional	Version 0 ipv4 1 ipv6
retcode	int	Return to	0 succeeded, otherwise failed. See attachment

		required	for error code
retmsg	string	Return to required	Return message

9.2 Requests and replies

Request:

```json
{
    "from": "box",
    "to": "admin",
    "action": "request",
    "cmdtype": 9,
    "sequence": 1,
    "token": "testtoken",
    "message": {
        "card": 0,
        "dns1": "0.0.0.0",
        "dns2": "0.0.0.0",
        "gateway": "192.168.1.1",
        "host_name": "MC2000",
        "ip_version": 0,
        "ipaddr": "192.168.1.101",
        "mode": 0,
        "netmask": "255.255.255.0"
    }
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 9,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

10.Network information acquisition

10.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type

			message when receiving such message response: It refers to the response message
cmdtype	int	Required	10 Acquire network information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
host_name	string	Optional	Host name, character range: 0~15
ipaddr	string	Optional	Network ip address xxx.xxx.xxx.xxx, Conform to ip format Default 192.168.1.168
netmask	string	Optional	Subnet mask xxx.xxx.xxx.xxx is consistent with IP authentication
gateway	string	Optional	Gateway xxx.xxx.xxx.xxx has the same network segment with ip and mask
dns1	string	Optional	dns server xxx.xxx.xxx.xxx
dns2	string	Optional	dns server xxx.xxx.xxx.xxx
card	int	Optional	NIC type: 0 limited NIC
mode	int	Optional	Mode 0 Static 1 Dynamic
ip_version	int	Optional	Version 0 ipv4 1 ipv6
retcode	int	Return to required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Return to required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

10.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":10,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action": "response",
    "cmdtype": 10,
    "message": {
        "card": 0,
```

```
        "dns1": "0.0.0.0",
        "dns2": "0.0.0.0",
        "gateway": "192.168.1.1",
        "host_name": "MC200",
        "ip_version": 0,
        "ipaddr": "192.168.1.101",
        "mac": "00:15:18:10:CA:9D",
        "mode": 0,
        "netmask": "255.255.255.0"
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

11.Port setting

11.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	11 Set network port
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object		Message body node, defining the request message
max_connectios	int	Optional	Maximum number of web connections, 1~20
tcp_port	int	Optional	web message interaction port, 1024~65535, used for websocket message interaction
rtsp_port	int	Optional	Infrared rtsp port 1024~65535, used for rtsp streaming service. The default port 554 retains setting and does not need to be restarted.
http_port	int	Optional	http server port, 1024~65535, and the default port 80 retains settings
onvif_check	int	Optional	Onvif login verification, admin and password if verified 0 Not verify 1 Verify
retcode	int	Optional	0 succeeded, otherwise failed. See attachment for error code

retmsg	string	Optional	Return message

11.2 Requests and replies

Request:

```json
{
    "from": "box",
    "to": "admin",
    "action": "request",
    "cmdtype": 11,
    "sequence": 1,
    "token": "testtoken",
    "message": {
        "http_port": 80,
        "max_connectios": 10,
        "onvif_check": 0,
        "rtsp_port": 554,
        "tcp_port": 9980
    }
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 11,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

12.Port information acquisition

12.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	12 Obtain network port
sequence	uint64	Required	Message serial number

token	string	Required	User access token
message	json object		Message body node, defining the request message
max_connectios	int	Optional	Maximum number of web connections, 1~20
tcp_port	int	Optional	web message interaction port, 1024~65535, used for websocket message interaction
rtsp_port	int	Optional	Infrared rtsp port 1024~65535, used for rtsp streaming service. The default port 554 retains setting and does not need to be restarted.
http_port	int	Optional	http server port, 1024~65535, and the default port 80 retains settings
onvif_check	int	Optional	Onvif login verification, admin and password if verified 0 Not verify 1 Verify
retcode	int	Optional	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Optional	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

12.2 Requests and replies

Request:

```json
{
    "from": "box",
    "to": "admin",
    "action": "request",
    "cmdtype": 12,
    "sequence": 1,
    "token": "testtoken",
    "message": { }
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 12,
    "message": {
        "http_port": 80
        "max_connectios": 10,
        "onvif_check": 0,
```

```
        "rtsp_port": 554,
        "tcp_port": 9980
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

13.Setting of system storage information

13.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	13 Set system storage information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
push_stream	int	Optional	Enable streaming service or not 0: No 1: Yes
save_days	int	Optional	Automatic deletion interval of alarm pictures and videos, unit: day 1~365*2,
is_record	int	Optional	Allow video recording during holidays or not 0: Yes， 1: No
is_capture	int	Optional	Allow screenshots during holidays or not 0:Yes 1: No
month_data	json array	Optional	Holiday month array
mon	int	Required	1~12, month, ignored if wrong
data	array	Required	Day collection. If there are illegal dates, ignore them. Example: "data":[1,3,4,5,6,7], indicating that the 1st/3rd/4th/5th/6th/7th day of a month is holiday
retcode	int	Return to required	0 succeeded, otherwise failed. See attachment for error code

retmsg	string	Return to required	Return message

13.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 13,
    "message": {
        "format_status": 0,
        "free": 0,
        "is_capture": 0,
        "is_record": 1,
        "month_data": [
            {
                "mon": 1,
                "data": [
                    "4"
                ]
            }
        ],
        "push_stream": 1,
        "save_days": 90,
        "system": 0,
        "total": 0,
        "type": "tfcard"
    },
    "from": "root",
    "to": "box",
    "sequence": 15,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 13,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 15
}
```

14.Acquisition of system storage information

14.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	14 Acquire system storage information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
push_stream	int	Optional	Enable streaming service or not 0: No 1: Yes
save_days	int	Optional	Automatic deletion interval of alarm pictures and videos, unit: day 1~365*2,
is_record	int	Optional	Allow video recording during holidays or not 0: Yes， 1: No
is_capture	int	Optional	Allow screenshots during holidays or not 0:Yes 1: No
month_data	json array	Optional	Holiday month array
mon	int	Required	1~12, month, ignored if wrong
data	array	Required	Day collection. If there are illegal dates, ignore them. Example: "data":[1,3,4,5,6,7], indicating that the 1st/3rd/4th/5th/6th/7th day of a month is holiday
retcode	int	Return to required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Return to required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

14.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 14,
    "message": {
        "default": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 14,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 14,
    "message": {
        "format_status": 0,
        "free": 0,
        "is_capture": 0,
        "is_record": 0,
        "month_data": null,
        "push_stream": 1,
        "save_days": 90,
        "system": 0,
        "total": 0,
        "type": "tfcard"
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 14
}
```

15.Setting of system information

15.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the

			response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	15 Set system information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
time_format	int	Optional	Time format 0 12-hour system 1 24-hour system
time_zone	int	Optional	Time Zone 0: Beijing, China 1 UTC Time UTC 2 International Date Line West UTC-12 3 Samoa UTC-11 4 Coordinated Universal Time -11 UTC-11 5 Hawaii UTC-10 6 Alaska UTC-9 7 Pacific Time UTC-8 8 Baja California UTC-8 9 Chiwawa, La Paz, Mazatlan UTC-7 10 Mountain Time UTC-7 11 Arizona UTC-7 12 Guadalajara UTC-6 13 Saskatchewan UTC-6 14 Central Time UTC-6 15 Central American UTC-6 16 Bogota UTC-5 17 Central Time UTC-5 18 Indiana UTC-5 19 Caracas UTC-4:30 20 Atlantic Time UTC-4 21 Kuyaba UTC-4 22 Georgetown UTC-4 23 San Diego UTC-4 24 Asuncion UTC-4 25 Newfoundland UTC-3:30 26 Brasilia UTC-3 27 Buenos Aires UTC-3 28 Greenland UTC-3 29 Cayenne UTC-3 30 Montevideo UTC-3 31 Coordinated Universal Time UTC-2 32 Mid-Atlantic UTC-2

			33 Cape Verde Islands UTC-1
			34 Azores UTC-1
			35 Dublin UTC-0
			36 Casablanca UTC-0
			37 Monrovia UTC-0
			38 Coordinated Universal Time UTC-0
			39 Amsterdam UTC +1
			40 Belgrade UTC +1
			41 Bluesel UTC +1
			42 Sarajevo UTC +1
			43 Windhoek UTC +1
			44 West Central Africa UTC +1
			45 Amman UTC +2
			46 Beirut UTC +2
			47 Damascus UTC +2
			48 Harare
			49 Helsinki UTC +2
			50 Cairo UTC +2
			51 Minsk UTC +2
			52 Athens UTC +2
			53 Jerusalem UTC +2
			54 Baghdad UTC +3
			55 Kuwait UTC +3
			56 Moscow UTC +3
			57 Nairobi UTC +3
			58 Tehran UTC +3:30
			59 Abu Dhabi UTC +4
			60 Yerevan UTC +4
			61 Baku UTC +4
			62 Tbilisi UTC +4
			63 Port Louis UTC +4
			64 Kabul UTC +4:30
			65 Tashkent UTC +5
			66 Yekaterinburg UTC +5
			67 Chennai UTC +5:30
			68 Sri Lanka Standard Time UTC +5:30
			69 Kathmandu UTC +5:45
			70 Astana UTC +6
			71 Dhaka UTC +6
			72 Novosibirsk UTC +6
			73 Rangoon UTC +6:30
			74 Krasnoyarsk UTC +7
			75 Bangkok UTC +7
			76 Kuala Lumpur UTC +8
			77 Perth UTC +8
			78 Taipei UTC +8
			79 Ulaanbaatar UTC +8

			80 Irkutsk UTC +8
			81 Osaka UTC +9
			82 Seoul UTC +9
			83 Yakutsk UTC +9
			84 Adelaide UTC +9:30
			85 Darwin UTC +9:30
			86 Brisbane UTC +10
			87 Vladivostok UTC +10
			88 Guam UTC +10
			89 Hobart UTC +10
			90 Canberra UTC +10
			91 Magadan UTC +11
			92 Solomon Islands UTC +11:30
			93 Auckland UTC +12
			94 Fiji UTC +12
			95 Coordinated Universal Time UTC +12
			96 Nukualofa UTC +13
language	int	Optional	System language, 0 Simplified Chinese, 1 ENGLISH
logo	string	Optional	Login logo,base64 encoded character, support JPEG and PNG formats,with a size no more than 500k
home_logo	string	Optional	Home logo,base64 encoded characters Support JPEG and PNG formats, with a size no more than 500k
name	string	Optional	System name, Character range 0~32 Default: Guide front-end configuration software
dist_unit	int	Optional	Distance unit: 0 m 1 foot
temperature_unit	int	Optional	Temperature unit 0 ℃ 1 ℉
capture_interval	int	Optional	Screenshot interval, in s, 1~3,600
copyright	string	Optional	Copyright, 0~128
capture_count	int	Optional	Number of screenshots, 1~100
record_interval	int	Optional	Delay recording time 1~3,600
record_time	int	Optional	Recording duration, in s, 1-3,600

record_max_size	int	Optional	Maximum size of recorded file, in M,1~1,000
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

15.2 Settings of requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 15,
    "message": {
        "capture_count": 3,
        "capture_interval": 5,
        "copyright": "COPYRIGHT © 2017-2020 WWW.GUIDEIR.COM ALL RIGHTS RESERVED Guide Sensmart Tech Co., Ltd.",
        "dist_unit": 0,
        "home_logo": "",
        "language": 0,
        "logo": "",
        "name": " Guide Front-end Configuration Software",
        "porting_name": "NC200",
        "record_interval": 5,
        "record_max_size": 999,
        "record_time": 300,
        "temperature_unit": 0,
        "time_format": 0,
        "time_zone": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 17,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 15,
    "message": null,
    "retcode": 0,
```

```
    "retmsg": "success.",
    "sequence": 17
}
```

16. System information acquisition

16.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	16 Obtain system information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, defining the request message
time_format	int	Optional	Time format 0 12-hour system 1 24-hour system
time_zone	int	Optional	Time Zone 0: Beijing, China 1 UTC Time UTC 2 International Date Line West UTC-12 3 Samoa UTC-11 4 Coordinated Universal Time -11 UTC-11 5 Hawaii UTC-10 6 Alaska UTC-9 7 Pacific Time UTC-8 8 Baja California UTC-8 9 Chiwawa, La Paz, Mazatlan UTC-7 10 Mountain Time UTC-7 11 Arizona UTC-7 12 Guadalajara UTC-6 13 Saskatchewan UTC-6 14 Central Time UTC-6 15 Central American UTC-6 16 Bogota UTC-5 17 Central Time UTC-5 18 Indiana UTC-5 19 Caracas UTC-4:30 20 Atlantic Time UTC-4

			21 Kuyaba UTC-4
			22 Georgetown UTC-4
			23 San Diego UTC-4
			24 Asuncion UTC-4
			25 Newfoundland UTC-3:30
			26 Brasilia UTC-3
			27 Buenos Aires UTC-3
			28 Greenland UTC-3
			29 Cayenne UTC-3
			30 Montevideo UTC-3
			31 Coordinated Universal Time UTC-2
			32 Mid-Atlantic UTC-2
			33 Cape Verde Islands UTC-1
			34 Azores UTC-1
			35 Dublin UTC-0
			36 Casablanca UTC-0
			37 Monrovia UTC-0
			38 Coordinated Universal Time UTC-0
			39 Amsterdam UTC +1
			40 Belgrade UTC +1
			41 Bluesel UTC +1
			42 Sarajevo UTC +1
			43 Windhoek UTC +1
			44 West Central Africa UTC +1
			45 Amman UTC +2
			46 Beirut UTC +2
			47 Damascus UTC +2
			48 Harare
			49 Helsinki UTC +2
			50 Cairo UTC +2
			51 Minsk UTC +2
			52 Athens UTC +2
			53 Jerusalem UTC +2
			54 Baghdad UTC +3
			55 Kuwait UTC +3
			56 Moscow UTC +3
			57 Nairobi UTC +3
			58 Tehran UTC +3:30
			59 Abu Dhabi UTC +4
			60 Yerevan UTC +4
			61 Baku UTC +4
			62 Tbilisi UTC +4
			63 Port Louis UTC +4
			64 Kabul UTC +4:30
			65 Tashkent UTC +5
			66 Yekaterinburg UTC +5
			67 Chennai UTC +5:30

			68 Sri Lanka Standard Time UTC +5:30
			69 Kathmandu UTC +5:45
			70 Astana UTC +6
			71 Dhaka UTC +6
			72 Novosibirsk UTC +6
			73 Rangoon UTC +6:30
			74 Krasnoyarsk UTC +7
			75 Bangkok UTC +7
			76 Kuala Lumpur UTC +8
			77 Perth UTC +8
			78 Taipei UTC +8
			79 Ulaanbaatar UTC +8
			80 Irkutsk UTC +8
			81 Osaka UTC +9
			82 Seoul UTC +9
			83 Yakutsk UTC +9
			84 Adelaide UTC +9:30
			85 Darwin UTC +9:30
			86 Brisbane UTC +10
			87 Vladivostok UTC +10
			88 Guam UTC +10
			89 Hobart UTC +10
			90 Canberra UTC +10
			91 Magadan UTC +11
			92 Solomon Islands UTC +11:30
			93 Auckland UTC +12
			94 Fiji UTC +12
			95 Coordinated Universal Time UTC +12
			96 Nukualofa UTC +13
language	int	Optional	System language, 0 Simplified Chinese, 1 ENGLISH
logo	string	Optional	Login logo,base64 encoded character, support JPEG and PNG formats,with a size no more than 500k
home_logo	string	Optional	Home logo,base64 encoded characters Support JPEG and PNG formats, with a size no more than 500k
name	string	Optional	System name, Character range 0~32 Default: Guide front-end configuration software
dist_unit	int	Optional	Distance unit: 0 m

				1 foot
temperature_unit	int	Optional		Temperature unit 0 ℃ 1 ℉
capture_interval	int	Optional		Screenshot interval, in s, 1~3,600
copyright	string	Optional		Copyright, 0~128
capture_count	int	Optional		Number of screenshots, 1~100
record_interval	int	Optional		Delay recording time 1~3,600
record_time	int	Optional		Recording duration, in s, 1-3,600
record_max_size	int	Optional		Maximum size of recorded file, in M,1~1,000
retcode	int	Required		0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required		Return message
default	int	Optional		Default parameter or not 0 No 1 Yes

16.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 16,
    "message": {
        "default": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 16,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 16,
    "message": {
        "capture_count": 3,
        "capture_interval": 5,
        "copyright": "COPYRIGHT © 2017-2020 WWW.GUIDEIR.COM ALL RIGHTS RESERVED Guide Sensmart Tech Co., Ltd.",
```

```
        "dist_unit": 0,
        "home_logo": "",
        "language": 0,
        "logo": "",
        "name": " Guide Front-end Configuration Software",
        "porting_name": "NC200",
        "record_interval": 5,
        "record_max_size": 999,
        "record_time": 300,
        "temperature_unit": 0,
        "time_format": 0,
        "time_zone": 0
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 16
}
```

17. Settings of shielding area

Set the shielding area. No temperature sensing is performed in the shielding area. If the coordinate range exceeds the infrared image (x + w> image width), take the boundary for processing.

17.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	17 Set shielding area
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
rects	json array	Required	Area array, upper left coordinate is 0,0 by standard, supports up to two
x	int	Required	x coordinate, 1~512, supports 0 and points to 1 by default,
y	int	Required	y coordinate, 1~384 supports 0 and points to 1 by default,
w	int	Required	Wide area, interaction with coordinates, and value range of

			1~512
h	int	Required	High area, interaction with coordinates, and value range of 1~384
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

17.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 17,
    "message": {
        "rects": [
            {
                "h": 202,
                "w": 249,
                "x": 168,
                "y": 69
            }
        ]
    },
    "from": "root",
    "to": "box",
    "sequence": 32,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 17,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 32
}
```

18. Shielded area acquisition

18.1 Field description

Field Name	Type	Options	Description

action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	18 Obtain shielding area
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
rects	json array	Required	Area array, upper left coordinate is 0,0 by standard, supports up to two
x	int	Required	x coordinate, 1~512, supports 0 and points to 1 by default,
y	int	Required	y coordinate, 1~384 supports 0 and points to 1 by default,
w	int	Required	Wide area, interaction with coordinates, value range 1~512
h	int	Required	High area, interaction with the coordinates, and the value range 1~384
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

18.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 18,
    "message": {
        "default": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 31,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 18,
    "message": {
        "rects": [
            {
                "h": 202,
                "w": 249,
                "x": 168,
                "y": 69
            }
        ]
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 31
}
```

19. Settings of cold hotspot tracking

19.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to response message
cmdtype	int	Required	19 Set cold hot spot tracking
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
trace_flag	int	Optional	Turn on or off 0 Off 1 On
color_mod	int	Optional	Color selection mode: 0 Auto 1 Manual
high_color	int	Optional	High temperature color: #RGB Red# ff0000
low_color	int	Optional	Low temperature color: #RGB

			Blue# 0000ff
alarm_shake	int	Optional	Shake alarm, in seconds, 0~100
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time, format HH:mm:ss
end	string	Optional	End time, format HH:mm:ss
record_flag	int	Optional	Record video or not 1 Yes 0 No
record_stream	int	Optional	Video type 1 Visible light 2 Infrared 3 Infrared and visible light
record_time	int	Optional	Video duration 10-300s
capture_flag	int	Optional	Screenshot or not 0: No 1: Yes
capture_stream	int	Optional	Screenshot type 1 Visible light 2 Infrared 3 Infrared and visible
high_flag	int	Optional	Detect high temperature or not 0 No 1 Yes
low_flag	int	Optional	Detect low temperature or not 0 No 1 Yes
high_temp	float	Optional	High temp,-40~2000 Default 60
low_temp	float	Optional	Low temp,-40~2000
sendmail	int	Optional	Send mail or not 0 No 1 Yes
light_flag	int	Optional	Turn flash on or off 0 Off 1 On
light_hold	int	Optional	Flash lasts for a long time, flashing at intervals of 10~300
output_flag	int	Optional	External output or not 0 No 1 Yes

output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

19.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 19,
    "message": {
        "light_hold": 10,
        "light_flag": 0,
        "alarm_shake": 10,
        "audio_flag": 1,
        "audio_index": 0,
        "audio_mode": 1,
        "audio_value": 10,
        "capture_flag": 1,
        "capture_stream": 3,
        "color_mod": 0,
        "effect_day": [
            {
                "day": 1,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            {
                "day": 2,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
```

```json
            ]
        },
        {
            "day": 3,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 4,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 5,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 6,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 7,
```

```json
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            "high_color": "#ff0000",
            "high_flag": 1,
            "high_temp": 10,
            "low_color": "#0000ff",
            "low_flag": 0,
            "low_temp": 10,
            "output_flag": 1,
            "output_hold": 10,
            "record_delay": 10,
            "record_flag": 1,
            "record_stream": 3,
            "sendmail": 0,
            "trace_flag": 1
        },
        "from": "root",
        "to": "box",
        "sequence": 36,
        "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 19,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 36
}
```

20. Cold hotspot tracking and acquisition

20.1 Field description

Field Name	Type	Options	Description

action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	20 Obtain cold hot spot tracking
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
trace_flag	int	Optional	Turn on or off 0 Off 1 On
color_mod	int	Optional	Color selection mode: 0 Auto 1 Manual
high_color	int	Optional	High temp color:#RGB Red# ff0000
low_color	int	Optional	Low temp color: #RGB Blue# 0000ff
alarm_shake	int	Optional	Shake alarm, in seconds, 0~100
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time, format HH:mm:ss
end	string	Optional	End time, format HH:mm:ss
record_flag	int	Optional	Record video or not 1 Yes 0 No
record_stream	int	Optional	Video type, 1 Visible light 2 Infrared 3 Infrared and visible light
record_time	int	Optional	Video duration 10-300s
capture_flag	int	Optional	Screenshot or not 0: No 1: Yes
capture_stream	int	Optional	Screenshot Type 1 Visible light 2 Infrared 3 Infrared and visible light
high_flag	int	Optional	Detect high temperature or

			not 0 No 1 Yes
low_flag	int	Optional	Detect low temperature or not 0 不检测 0 No 1 检测 1 Yes
high_temp	float	Optional	High temp,-40~2000 Default 60
low_temp	float	Optional	Low temp,-40~2000
sendmail	int	Optional	Send mail or not 0 No 1 Yes
light_flag	int	Optional	Turn flash on or off 0 Off 1 On
light_hold	int	Optional	Flash lasts for a long time, flashing at intervals of 10~300
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

20.2 Acquisition of requests and replies

Request:

```
{
    "action": "request",
    "cmdtype": 20,
    "message": {
        "default": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 34,
```

```json
        "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 20,
    "message": {
        "light_hold": 10,
        "light_flag": 0,
        "alarm_shake": 10,
        "audio_flag": 1,
        "audio_index": 0,
        "audio_mode": 1,
        "audio_value": 10,
        "capture_flag": 1,
        "capture_stream": 3,
        "color_mod": 0,
        "effect_day": [
            {
                "day": 1,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            {
                "day": 2,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            {
                "day": 3,
                "period": [
                    {
```

```json
                "end": "23:59:59",
                "start": "00:00:00"
            }
        ]
    },
    {
        "day": 4,
        "period": [
            {
                "end": "23:59:59",
                "start": "00:00:00"
            }
        ]
    },
    {
        "day": 5,
        "period": [
            {
                "end": "23:59:59",
                "start": "00:00:00"
            }
        ]
    },
    {
        "day": 6,
        "period": [
            {
                "end": "23:59:59",
                "start": "00:00:00"
            }
        ]
    },
    {
        "day": 7,
        "period": [
            {
                "end": "23:59:59",
                "start": "00:00:00"
            }
        ]
```

```
            }
        ],
        "high_color": "#ff0000",
        "high_flag": 1,
        "high_temp": 10,
        "low_color": "#0000ff",
        "low_flag": 0,
        "low_temp": 10,
        "output_flag": 1,
        "output_hold": 10,
        "record_delay": 10,
        "record_flag": 1,
        "record_stream": 3,
        "sendmail": 0,
        "trace_flag": 1
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 34
}
```

21. Temperature alarm settings

21.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	21 Set temperature alarm information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
alarm_flag	int	Optional	Enable alarm or not 1 Yes 0 No
effect_day	json array	Optional	Time array
day	int	Required	1~7, day of week
period	json	Required	Time period
start	string	Required	Start time, format HH:mm:ss

end	string	Required	End time, format HH:mm:ss
record_flag	int	Optional	Record video or not 1 Yes 0 No
record_stream	int	Optional	Video type 0 No video 1 Visible light 2 Infrared 3 Infrared and visible light
record_delay	int	Optional	Video duration 10~300
capture_flag	int	Optional	Take a screenshot or not 0: No 1: Yes
capture_stream	int	Optional	Screenshot type 0 No screenshot 1 Visible light 2 Infrared 3 Infrared and visible
audio_flag	int	Optional	Music linkage or not 1 Yes 0 No
audio_playtype	int	Optional	Music mode 1 Plays 2 Duration
audio_playvalue	int	Optional	Music value Subject to mode definition Duration: seconds Plays: Plays 0~120
audio_play_index	int	Optional	Music file index, available from common interface
light_hold	int	Optional	Flash output time, 10~300
light_flag	int	Optional	Turn on flash or not 0 Off 1 On
sendmail	int	Optional	Send mail or not 0 No 1 Yes
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

21.2 Requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 21,
    "message": {
        "alarm_flag": 1,
        "light_hold": 10,
        "light_flag": 0,
        "alarm_shake": 10,
        "audio_flag": 1,
        "audio_play_index": 1,
        "audio_playtype": 1,
        "audio_playvalue": 10,
        "capture_flag": 1,
        "capture_stream": 3,
        "effect_day": [
            {
                "day": 1,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            {
                "day": 2,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            {
                "day": 3,
                "period": [
                    {
```

                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 4,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 5,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 6,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 7,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]

```
            }
        ],
        "output_flag": 0,
        "output_hold": 10,
        "record_delay": 10,
        "record_flag": 1,
        "record_stream": 3,
        "sendmail": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 38,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```
{
    "action": "response",
    "cmdtype": 21,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 38
}
```

22. Temperature alarm acquisition

22.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	22 Obtain temperature alarm information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
alarm_flag	int	Optional	Enable alarm or not 1 Yes 0 No
effect_day	json array	Optional	Time array

day	int	Required	1~7, day of week
period	json	Required	Time period
start	string	Required	Start time, format HH:mm:ss
end	string	Required	End time, format HH:mm:ss
record_flag	int	Optional	Record video or not 1 Yes 0 No
record_stream	int	Optional	Video type 0 No video 1 Visible light 2 Infrared 3 Infrared and visible light
record_delay	int	Optional	Video duration 10~300
capture_flag	int	Optional	Take a screenshot or not 0: No 1: Yes
capture_stream	int	Optional	Screenshot type 0 No screenshot 1 Visible light 2 Infrared 3 Infrared and visible
audio_flag	int	Optional	Music linkage or not 1 Yes 0 No
audio_playtype	int	Optional	Music mode 1 Plays 2 Duration
audio_playvalue	int	Optional	Music value Subject to mode definition Duration: seconds Plays: Plays 0~120
audio_play_index	int	Optional	Music file index, available from common interface
light_hold	int	Optional	Flash output time, 10~300
light_flag	int	Optional	Turn on flash or not 0 Off 1 On
sendmail	int	Optional	Send mail or not 0 No 1 Yes
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration

			10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

22.2 Acquisition of requests and replies

Request:

```json
{
    "action": "request",
    "cmdtype": 22,
    "message": {
        "default": 0
    },
    "from": "root",
    "to": "box",
    "sequence": 37,
    "token": "a7fece3ca9b111ecb8e200151810ca9d"
}
```

Reply:

```json
{
    "action": "response",
    "cmdtype": 22,
    "message": {
        "alarm_flag": 1,
        "light_hold": 10,
        "light_flag": 0,
        "alarm_shake": 10,
        "audio_flag": 1,
        "audio_play_index": 1,
        "audio_playtype": 1,
        "audio_playvalue": 10,
        "capture_flag": 1,
        "capture_stream": 3,
        "effect_day": [
            {
                "day": 1,
                "period": [
                    {
```

```json
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 2,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 3,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 4,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
        },
        {
            "day": 5,
            "period": [
                {
                    "end": "23:59:59",
                    "start": "00:00:00"
                }
            ]
```

```
            },
            {
                "day": 6,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            },
            {
                "day": 7,
                "period": [
                    {
                        "end": "23:59:59",
                        "start": "00:00:00"
                    }
                ]
            }
        ],
        "output_flag": 0,
        "output_hold": 10,
        "record_delay": 10,
        "record_flag": 1,
        "record_stream": 3,
        "sendmail": 0
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 37
}
```

23. Video parameter settings

23.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response

			message
cmdtype	int	Required	23 Set video parameter
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
params	json array	Optional	Video Parameter List
ratio	string	Optional	The resolution of the main stream of the visible light is supported by the project. 2592x1944 1920x1080 1280x960 1280x720 Resolution of substream of visible light 704x576 640x480 Resolution of infrared main stream 512x384 Resolution of infrared substream 256x192
encodec	int	Optional	Coding of main stream of visual light currently only supports H264 0:h264 1:h265 2:mjpeg
fps	int	Optional	Frame rate of main stream of visible light 1~25
rate_control	int	Optional	Main stream control of visible light 0: Variable stream 1: Fixed stream
bit_size	int	Optional	Code stream has a fixed value, and also needs to be set if there is variable code stream 50~409600 Kb/s
gop_size	int	Optional	Main frame interval of visible light, 2~150, the maximum cannot be more than 2 times fps
type	int	Optional	Stream type: (if present, required; If not, other parameter are invalid) 1 Main stream of visible light 2 Substream of visible light 3 Infrared main stream 4 Infrared substream
level	int	Optional	Code quality level. The level effect changes with actual situations, 1 Best 2 Better

			3 OK 4 Poor 5 Worse 6 Worst
svc	int	Optional	Frame rate can be coded in layers, H264 is valid, other formats also need to be passed in 1 Not layered 0 Layered
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

23.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":23,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "params" :
            [
                {
                    "bit_size" : 4096,
                    "encodec" : 0,
                    "fps" : 25,
                    "gop_size" : 50,
                    "level" : 3,
                    "rate_control" : 0,
                    "ratio" : "1920x1080",
                    "svc" : 1,
                    "type" : 1
                },
                {
                    "bit_size" : 4096,
                    "encodec" : 0,
                    "fps" : 25,
                    "gop_size" : 50,
                    "level" : 3,
                    "rate_control" : 0,
                    "ratio" : "702x576",
```

                       "svc" : 1,

                       "type" : 2

               },

               {

                       "bit_size" : 4096,

                       "encodec" : 0,

                       "fps" : 25,

                       "gop_size" : 50,

                       "level" : 1,

                       "rate_control" : 0,

                       "ratio" : "512x384",

                       "svc" : 1,

                       "type" : 3

               },

               {

                       "bit_size" : 4096,

                       "encodec" : 0,

                       "fps" : 25,

                       "gop_size" : 50,

                       "level" : 1,

                       "rate_control" : 0,

                       "ratio" : "256x192",

                       "svc" : 1,

                       "type" : 4

               }

           ]

    }

}

Reply:

{

    "action" : "response",

    "cmdtype" : 23,

    "sequence" : 1,

    "retcode" : 0,

    "retmsg" : "success."

}

24. Video parameter acquisition

24.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	24 Obtain video parameter
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
params	json array	Optional	Video Parameter List
ratio	string	Optional	The resolution of the main stream of the visible light is supported by the project. 2592x1944 1920x1080 1280x960 1280x720 Resolution of substream of visible light 704x576 640x480 Resolution of infrared main stream 512x384 Resolution of infrared substream 256x192
encodec	int	Optional	Coding of main stream of visual light currently only supports H264 0:h264 1:h265 2:mjpeg
fps	int	Optional	Frame rate of main stream of visible light 1~25
rate_control	int	Optional	Main stream control of visible light 0: Variable stream 1: Fixed stream
bit_size	int	Optional	Code stream has a fixed value, and also needs to be set if there is variable code stream 50~409600 Kb/s
gop_size	int	Optional	Main frame interval of visible light, 2~150, the maximum cannot be more than 2 times fps
type	int	Optional	Stream type: (if present, required; If

				not, other parameter are invalid)
				1 Main stream of visible light
				2 Substream of visible light
				3 Infrared main stream
				4 Infrared substream
level	int	Optional	Code quality level. The level effect changes with actual situations,
			1 Best
			2 Better
			3 OK
			4 Poor
			5 Worse
			6 Worst
svc	int	Optional	Frame rate can be coded in layers, H264 is valid, other formats also need to be passed in
			1 Not layered
			0 Layered
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not
			0 No
			1 Yes

24.2 Acquisition of requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":24,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 24,
    "sequence" : 1,
    "message" :
    {

        "params" :
        [
            {
```

            "bit_size" : 4096,

            "encodec" : 0,

            "fps" : 25,

            "gop_size" : 50,

            "level" : 3,

            "rate_control" : 0,

            "ratio" : "1920x1080",

            "svc" : 1,

            "type" : 1

    },

    {

            "bit_size" : 4096,

            "encodec" : 0,

            "fps" : 25,

            "gop_size" : 50,

            "level" : 3,

            "rate_control" : 0,

            "ratio" : "702x576",

            "svc" : 1,

            "type" : 2

    },

    {

            "bit_size" : 4096,

            "encodec" : 0,

            "fps" : 25,

            "gop_size" : 50,

            "level" : 1,

            "rate_control" : 0,

            "ratio" : "512x384",

            "svc" : 1,

            "type" : 3

    },

    {

            "bit_size" : 4096,

            "encodec" : 0,

            "fps" : 25,

            "gop_size" : 50,

            "level" : 1,

            "rate_control" : 0,

            "ratio" : "256x192",

```
                    "svc" : 1,
                    "type" : 4
                }
            ]


    }
    "retcode" : 0,
    "retmsg" : "success."


}
```

25. Storage exception settings

From this interface, there is the processing settings when storage exceptions occur (full memory, storage failure, streaming exceptions, etc.).

25.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message
cmdtype	int	Required	25 Obtain storage exception
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
store	json array	Required	Storage array
type	int	Required	Type 1 Memory full 2 Storage failure 3 streaming exception
flag	int	Optional	Turn on or off 0: Off 1: On
light_flag	int	Optional	Flash output or not 0: No 1 Yes
light_hold	int	Optional	Output time, in seconds, 10~300
audio_flag	int	Optional	Audio linkage is enabled by default or not 0: No 1 Yes
audio_mode	int	Optional	Audio mode, default times 0: Duration 1: Plays
audio_value	int	Optional	Music value is 10 by default, no mode is distinguished, and if distinguished, add fields

			Subject to mode definition Duration: seconds Plays: Plays 0~200
audio_index	int	Optional	Music file index, available from common interface
sendmail	int	Optional	Send mail or not 0 No 1 Yes
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

25.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":25,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "store" :
            [
                {
                    "audio_flag" : 0,
                    "audio_index" : 0,
                    "audio_mode" : 0,
                    "audio_value" : 10,
                    "flag" : 0,
                    "sendmail" : 0,
                    "light_flag" : 0,
                    "light_hold" : 10,
                    "output_flag" : 0,
                    "output_hold" : 0,
                    "type" : 1
                },
                {
                    "audio_flag" : 0,
```

```
                    "audio_index" : 0,
                    "audio_mode" : 0,
                    "audio_value" : 10,
                    "flag" : 0,
                    "sendmail" : 0,
                    "light_flag" : 0,
                    "light_hold" : 10,
                    "output_flag" : 0,
                    "output_hold" : 0,
                    "type" : 2
              },
              {
                    "audio_flag" : 0,
                    "audio_index" : 0,
                    "audio_mode" : 0,
                    "audio_value" : 10,
                    "flag" : 0,
                    "sendmail" : 0,
                    "light_flag" : 0,
                    "light_hold" : 10,
                    "output_flag" : 0,
                    "output_hold" : 0,
                    "type" : 3
              }
        ]

    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 25,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."

}
```

26. Storage exception acquisition

From this interface, there is the processing settings for storage exceptions (full memory, storage failure, streaming exception, etc.).

26.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	26 Obtain storage exception
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
store	json array	Required	Storage array
type	int	Required	Type 1 Memory full 2 Storage failure 3 streaming exception
flag	int	Optional	Turn on or off 0: Off 1: On
light_flag	int	Optional	Flash output or not 0: No 1 Yes
light_hold	int	Optional	Output time, in seconds, 10~300
audio_flag	int	Optional	Audio linkage is enabled by default or not 0: No 1 Yes
audio_mode	int	Optional	Audio mode, default times 0: Duration 1: Plays
audio_value	int	Optional	Music value is 10 by default, no mode is distinguished, and if distinguished, add fields Subject to mode definition Duration: seconds Plays: Plays 0~200
audio_index	int	Optional	Music file index, available from common interface
sendmail	int	Optional	Send mail or not 0 No 1 Yes
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration

				10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code	
retmsg	string	Required	Return message	
default	int	Optional	Default parameter or not 0 No 1 Yes	

26.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":26,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 26,
    "sequence" : 1,
    "message" :
    {

            "store" :
            [
                {
                    "audio_flag" : 0,
                    "audio_index" : 0,
                    "audio_mode" : 0,
                    "audio_value" : 10,
                    "flag" : 0,
                    "sendmail" : 0,
                    "light_flag" : 0,
                    "light_hold" : 10,
                    "output_flag" : 0,
                    "output_hold" : 0,
                    "type" : 1
                },
                {
```

                    "audio_flag" : 0,

                    "audio_index" : 0,

                    "audio_mode" : 0,

                    "audio_value" : 10,

                    "flag" : 0,

                    "sendmail" : 0,

                    "light_flag" : 0,

                    "light_hold" : 10,

                    "output_flag" : 0,

                    "output_hold" : 0,

                    "type" : 2

            },

            {

                    "audio_flag" : 0,

                    "audio_index" : 0,

                    "audio_mode" : 0,

                    "audio_value" : 10,

                    "flag" : 0,

                    "sendmail" : 0,

                    "light_flag" : 0,

                    "light_hold" : 10,

                    "output_flag" : 0,

                    "output_hold" : 0,

                    "type" : 3

            }

        ]

    }

    "retcode" : 0,

    "retmsg" : "success."

}

27. Network exception settings

From this interface, there is the processing settings for network exceptions.

27.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message

			response: It refers to the response message
cmdtype	int	Required	27 Set network exception
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
flag	int	Optional	Turn on or off 0: Off 1: On
light_flag	int	Optional	Output or not 0: No 1 Yes
light_hold	int	Optional	Output time, in seconds, 10~300
audio_flag	int	Optional	Audio linkage or not 0: No 1 Yes
audio_mode	int	Optional	Audio mode 0: Duration 1: Plays
audio_value	int	Optional	Audio mode value 0~200 (times/second)
audio_index	int	Optional	Audio file index
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

27.2 Requests and replies

Request:

{

    "action":"request",

    "cmdtype":27,

    "sequence":1,

    "token":"8f7d83404fc011ebb4c18eeaf505828c"

    "message":{

        "audio_flag" : 0,

        "audio_index" : 0,

        "audio_mode" : 0,

        "audio_value" : 10,

        "flag" : 0,

        "output_flag" : 0,

        "output_hold" : 0,

        "light_flag" : 0,

```
        "light_hold" : 10


    }

}

Reply:

{

    "action" : "response",

    "cmdtype" : 27,

    "sequence" : 1,


    "retcode" : 0,

    "retmsg" : "success."


}
```

28. Network exception acquisition

From this interface, there is the processing settings in case of network exception.

28.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	28 Obtain network exception
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
flag	int	Optional	Turn on or off 0: Off 1: On
light_flag	int	Optional	Output or not 0: No 1 Yes
light_hold	int	Optional	Output time, in seconds, 10~300
audio_flag	int	Optional	Audio linkage or not 0: No 1 Yes
audio_mode	int	Optional	Audio mode 0: Duration 1: Plays
audio_value	int	Optional	Audio mode value 0~200 (times/second)
audio_index	int	Optional	Audio file index
output_flag	int	Optional	External output or not 0 No

			1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

28.2 Requests and replies

Request:

```
{
   "action":"request",
   "cmdtype":28,
   "sequence":1,
   "token":"test",
   "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 28,
    "sequence" : 1,
    "message" :
    {

            "audio_flag" : 0,
            "audio_index" : 0,
            "audio_mode" : 0,
            "audio_value" : 10,
            "flag" : 0,
            "light_flag" : 0,
            "output_flag" : 0,
            "output_hold" : 0,
            "light_hold" : 10
    }
    "retcode" : 0,
    "retmsg" : "success."

}
```

29. Illegal access settings

From this interface, there is the processing settings when illegal access occurs.

29.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	29 Set illegal access Parameter
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
flag	int	Optional	Turn on or off 0: Off 1: On
light_flag	int	Optional	Output or not 0: No 1 Yes
light_hold	int	Optional	Output time, in seconds, 10~300
audio_flag	int	Optional	Audio linkage or not 0: No 1 Yes
audio_mode	int	Optional	Audio mode 0: Duration 1: Plays
audio_value	int	Optional	Audio mode value 0~200 (times/second)
audio_index	int	Optional	Audio file index
sendmail	int	Optional	Send mail or not 0 No 1 Yes
allow_count	int	Optional	Allowed login count 3~10 times
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

29.2 Requests and replies

Request:

{

    "action":"request",

```
    "cmdtype":29,

    "sequence":1,

    "token":"8f7d83404fc011ebb4c18eeaf505828c"

    "message":{

        "allow_count" : 5,

        "audio_flag" : 0,

        "audio_index" : 0,

        "audio_mode" : 0,

        "audio_value" : 10,

        "flag" : 0,

        "sendmail" : 0,

        "light_flag" : 0,

        "output_flag" : 0,

        "output_hold" : 0,

        "light_hold" : 10

    }

}
Reply:

{

    "action" : "response",

    "cmdtype" : 29,

    "sequence" : 1,


    "retcode" : 0,

    "retmsg" : "success."


}
```

30. Illegal access acquisition

From this interface, there is the processing settings when an illegal access occurs.

30.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	30 Obtain illegal access parameter
sequence	uint64	Required	Message serial number
token	string	Required	User access token

message	json object	Required	A message body node defines request message
flag	int	Optional	Turn on or off 0: Off 1: On
light_flag	int	Optional	Output or not 0: No 1 Yes
light_hold	int	Optional	Output time, in seconds, 10~300
audio_flag	int	Optional	Audio linkage or not 0: No 1 Yes
audio_mode	int	Optional	Audio mode 0: Duration 1: Plays
audio_value	int	Optional	Audio mode value 0~200 (times/second)
audio_index	int	Optional	Audio file index
sendmail	int	Optional	Send mail or not 0 No 1 Yes
allow_count	int	Optional	Allowed login count 3~10 times
output_flag	int	Optional	External output or not 0 No 1 Yes
output_hold	int	Optional	External output duration 10~300s
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

30.2 Requests and replies

Request:

```
{
   "action":"request",
   "cmdtype":30,
   "sequence":1,
   "token":"test",
   "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 30,
```

```
        "sequence" : 1,
        "message" :
        {

                "allow_count" : 5,
                "audio_flag" : 0,
                "audio_index" : 0,
                "audio_mode" : 0,
                "audio_value" : 10,
                "flag" : 0,
                "sendmail" : 0,
                "light_flag" : 0,
                "output_flag" : 0,
                "output_hold" : 0,
                "light_hold" : 10
        }
        "retcode" : 0,
        "retmsg" : "success."


}
```

31. Mail information settings

Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message, The receiver needs to reply when receiving such message response: It refers to the response message
cmdtype	int	Required	31 Set mail information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
smtp_server	string	Optional	smtp server, default empty xxx.xxx.xxx.xxx
smtp_port	int	Optional	smtp service port, default 25,
encry_type	int	Optional	Encryption scheme 0: none 1 tls 2 ssl
enclosure	int	Optional	With accessories 0: No 1 Yes
is_anon	int	Optional	Anonymous

			0：否 0: No 1 Yes
from	string	Optional	From, Character range: 0~128
subject	string	Optional	Theme Character range: 0~128
username	string	Optional	Login server username, Character range 0~128
password	string	Optional	Login password, ciphertext transmission See attachment for format Character range: 0~256
mailto	json array	Optional	Recipient list, up to five people allowed Character range: 0~128
health	int	Optional	Use health mail 0: No 1 Yes
health_value	int	Optional	Health mail interval 1-360
alarm	int	Optional	Use alarm mail or not 0: No 1 Yes
alarm_value	int	Optional	Alarm mail interval 1~3600
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

31.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":31,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "alarm" : 0,
        "alarm_value" : 10,
        "enclosure" : 0,
        "encry_type" : 0,
        "from" : "",
        "health" : 0,
        "health_value" : 10,
        "is_anon" : 0,
        "mailto":
                [
                    "1"
                ],
```

```
        "password" : "",

        "smtp_port" : 25,

        "smtp_server" : "",

        "subject" : "",

        "username" : ""

    }

}

Reply:

{

    "action" : "response",

    "cmdtype" : 31,

    "sequence" : 1,


    "retcode" : 0,

    "retmsg" : "success."


}
```

32. Message information acquisition

32.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message, The receiver needs to reply when receiving such message response: It refers to the response message
cmdtype	int	Required	32 Obtain mail information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
smtp_server	string	Optional	smtp server, default empty xxx.xxx.xxx.xxx
smtp_port	int	Optional	smtp service port, default 25,
encry_type	int	Optional	Encryption scheme 0: none 1 tls 2 ssl
enclosure	int	Optional	With accessories 0: No 1 Yes
is_anon	int	Optional	Anonymous 0：否 0: No 1 Yes
from	string	Optional	From, Character range: 0~128
subject	string	Optional	Theme

			Character range: 0~128
username	string	Optional	Login server username, Character range 0~128
password	string	Optional	Login password, ciphertext transmission See attachment for format Character range: 0~256
mailto	json array	Optional	Recipient list, up to five people allowed Character range: 0~128
health	int	Optional	Use health mail 0: No 1 Yes
health_value	int	Optional	Health mail interval 1-360
alarm	int	Optional	Use alarm mail or not 0: No 1 Yes
alarm_value	int	Optional	Alarm mail interval 1~3600
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

32.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":32,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 32,
    "sequence" : 1,
    "message" :
    {

            "alarm" : 0,
            "alarm_value" : 10,
            "enclosure" : 0,
            "encry_type" : 0,
```

```
                "from" : "",

                "health" : 0,

                "health_value" : 10,

                "is_anon" : 0,

                "mailto":

                        [
                        "1","1","","","","","","",""
                        ],

            mail1

            mail2

            mail3

                "password" : "",

                "smtp_port" : 25,

                "smtp_server" : "",

                "subject" : "",

                "username" : ""

        }

        "retcode" : 0,

        "retmsg" : "success."


}
```

33. MQTT settings

33.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	33 Set MQTT
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
username	string	Optional	Username Character range 0~128
password	string	Optional	For password, ciphertext transmission and format, see attachment Character range 0~256
serverip	string	Optional	Server ip,xxx.xxx.xxx.xxx Character range 0~64

flag	int	Optional	Turn on or off, 0 On 1 Off
serverport	int	Optional	Port, 1~6553,
deviceid	string	Optional	Device id Character range 0~128
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

33.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":33,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "deviceid" : "guide",
        "password" : "admin",
        "serverip" : "192.168.1.16",
        "serverport" : 1883,
        "flag" : 1,
        "username" : "admin"
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 33,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."

}
```

34. MQTT information acquisition

34.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type:

			request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	34 Obtain MQTT
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
username	string	Optional	Username Character range 0~128
password	string	Optional	For password, ciphertext transmission and format, see attachment Character range 0~256
serverip	string	Optional	Server ip,xxx.xxx.xxx.xxx Character range 0~64
flag	int	Optional	Turn on or off, 0 On 1 Off
serverport	int	Optional	Port, 1~6553,
deviceid	string	Optional	Device id Character range 0~128
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0 No 1 Yes

34.2 Requests and replies

Request:

{

  "action":"request",

  "cmdtype":34,

  "sequence":1,

  "token":"test",

  "message":{}

}

Reply:

{

    "action" : "response",

    "cmdtype" : 34,

    "sequence" : 1,

    "message" :

```
        {

                "deviceid" : "guide",

                "password" : "admin",

                "serverip" : "192.168.1.16",

                "serverport" : 1883,

                "flag" : 1,

                "username" : "admin"



        }
        "retcode" : 0,
        "retmsg" : "success."


}
```

35. Fill light settings

35.1 Field description

Field Name	Type	Options	Description
action	string	Required	Message type: request: It refers to the request message. The receiver needs to reply to the response type message when receiving such message response: It refers to the response message
cmdtype	int	Required	35 Set fill light information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	A message body node defines request message
mode	int	Optional	Lamp mode 0 Manual 1 Automatic
light	int	Optional	Lamp on 0 Off 1 On
brightness	int	Optional	Brightness 0-100
retcode	int	Required	0 succeeded, otherwise failed. See attachment for error code
retmsg	string	Required	Return message

35.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":35,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "brightness" : 0,
        "light" : 0,
        "mode" : 0
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 35,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."

}
```

36. Fill light information acquisition

36.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	36: obtain fill light information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
mode	int	Optional	Light mode 0: manual 1: automatic
light	int	Optional	Light status

				0: off 1: on
brightness	int	Optional		Brightness: 0-100
retcode	int	Required		0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required		Return message
default	int	Optional		Default parameter or not 0: no 1: yes

36.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":36,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 36,
    "sequence" : 1,
    "message" :
    {

            "brightness" : 0,
            "light" : 0,
            "mode" : 0
    }
    "retcode" : 0,
    "retmsg" : "success."


}
```

39. Infrared mode settings

This interface is used to set whether the current mode is the fusion mode or single-light-source infrared mode.

39.1 Field description

Field name	Type	Option	Description

action	string	Requir ed	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Requir ed	39: set the infrared mode
sequence	uint64	Requir ed	Message serial number
token	string	Requir ed	User access token
message	json object	Requir ed	Message body node, used to define the request message
mode	int	Requir ed	Infrared mode 0 - single-light-source infrared 1 - double-light-source infrared
retcode	int	Requir ed	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Requir ed	Return message

39.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":39,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        " mode" : 0,
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 41,
    "sequence" : 1,

        "retcode" : 0,
        "retmsg" : "success."

}
```

40. Infrared mode acquisition

40.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	40: obtain the infrared mode
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
mode	int	Required	Infrared mode. 1 - single-light-source infrared 2 - double-light-source infrared
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no 1: yes

40.2 Requests and replies

Request:

```
{
   "action":"request",
   "cmdtype":40,
   "sequence":1,
   "token":"test",
   "message":{}
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 40,
    "sequence" : 1,
    "message" :
    {

            "mode" : 0,
```

```
        }
    "retcode" : 0,
    "retmsg" : "success."


}
```

41. Mute state settings

41.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	41: set the mute state
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
silent	int	Required	0: not mute 1: mute
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message

41.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":41,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "silent" : 0,
    }
}
```
Reply:

```
{
    "action" : "response",
    "cmdtype" : 43,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."

}
```

42. Mute state acquisition

42.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	42: obtain the mute state
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
silent	int	Required	0: not mute 1: mute
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no 1: yes

42.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":42,
    "sequence":1,
    "token":"test",
    "message":{}
```

```
}
Reply:
{
    "action" : "response",
    "cmdtype" : 42,
    "sequence" : 1,
    "message" :
    {

            "silent" : 0,

    }
    "retcode" : 0,
    "retmsg" : "success."

}
```

43. National standard information settings

43.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	43: set national standard information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
sip_id	string	Optional	sip server number, Character range: 0~128
sip_domain	string	Optional	sip domain Character range: 0~128
sip_server	string	Optional	sip server ip Character range: 0~128
sip_port	int	Optional	sip server port: 1024~65535
device_id	string	Optional	Equipment No.

				Character range: 0~128
device_port	int	Optional		Local sip server port: 1024~65535
reg_password	string	Optional		Registration password, ciphertext Character range: 0~256
reg_expires	int	Optional		Validity period of registration (s): 0~7200
heart_interval	int	Optional		Heartbeat cycle (s): 0~120
heart_count	int	Optional		Maximum heartbeat timeout times: 1~10
ir_id	string	Optional		Infrared channel number Character range: 0~128
vl_id	string	Optional		Visible light channel number Character range: 0~128
protocol	int	Optional		Signaling communication mode: 0：udp 0: udp 1：tcp 1: tcp
retcode	int	Required		0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required		Return message

43.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":43,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "device_port":5060,
    "flag":0,
    "heart_count":3,
    "heart_interval":60,
    "protocol":0,
    "reg_expires":3600,
    "reg_password":"",
    "sip_port":5060,
    "sip_server":""
    }
}
```

Reply:

```
{

    "action" : "response",

    "cmdtype" : 43,

    "sequence" : 1,


    "retcode" : 0,

    "retmsg" : "success."


}
```

44. Acquisition of national standard information

44.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	44: obtain national standard information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
sip_id	string	Optional	sip server number, Character range: 0~128
sip_domain	string	Optional	sip domain Character range: 0~128
sip_server	string	Optional	sip server ip Character range: 0~128
sip_port	int	Optional	sip server port: 1024~65535
device_id	string	Optional	Equipment No. Character range: 0~128
device_port	int	Optional	Local sip server port: 1024~65535
reg_password	string	Optional	Registration password, ciphertext Character range: 0~256
reg_expires	int	Optional	Validity period of registration (s):

			0~7200
heart_interval	int	Optional	Heartbeat cycle (s): 0~120
heart_count	int	Optional	Maximum heartbeat timeout times: 1~10
ir_id	string	Optional	Infrared channel number Character range: 0~128
vl_id	string	Optional	Visible light channel number Character range: 0~128
protocol	int	Optional	Signaling communication mode: 0: udp 1: tcp
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no 1: yes

44.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":44,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action":"response",
    "cmdtype":44,
    "message":{
      "device_port":5060,
      "flag":0,
      "heart_count":3,
      "heart_interval":60,
      "protocol":0,
      "reg_expires":3600,
        "reg_password":"",
      "sip_port":5060,
       "sip_server":""
    },
        "retcode" : 0,
      "retmsg" : "success."
```

```
    "sequence":1
}
```

45. Platform access settings

45.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	45: set access platform information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
serverip	string	Optional	Server ip Character range: 0~64 Default: null
serverport	string	Optional	Server port: 1~65535
user	string	Optional	Server user name, Character range: 0~128
password	int	Optional	Server password, Character range: 0~256
manufactor	int	Optional	Manufacturer 0: other protocols: connecting with integrated management platform phase 1 1: Guide Sensmart: connecting with integrated management platform phase 2
devicename	string	Optional	Equipment name Character range: 0~128
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message

45.2 Requests and replies

Request:

{

```
    "action":"request",

    "cmdtype":45,

    "sequence":1,

    "token":"8f7d83404fc011ebb4c18eeaf505828c"

    "message":{

        "flag": 0,

         "password": "",

         "serverip": "",

          "serverport": 0,

        "manufactor"：1，

        "devicename":"",

         "user": ""

    }

}
Reply:

{

    "action" : "response",

    "cmdtype" : 45,

    "sequence" : 1,


    "retcode" : 0,

    "retmsg" : "success."

    }
```

46. Platform access acquisition

46.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	46: obtain access platform information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes

serverip	string	Optional	Server ip Character range: 0~64 Default null
serverport	string	Optional	Server port: 1~65535
user	string	Optional	Server user name, Character range: 0~128
password	int	Optional	Server password, Character range: 0~256
manufactor	int	Optional	Manufacturer 0: other protocols: connecting with integrated management platform phase 1 1: Guide Sensmart: connecting with integrated management platform phase 2
devicename	string	Optional	Equipment Name Character range: 0~128
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no 1: yes

46.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":46,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
{
    "action": "response",
    "cmdtype": 46,
    "message": {
            "manufactor"：1,
            "flag": 0,
            "password": "",
            "serverip": "",
            "serverport": 0,
```

```
            "devicename": "",
            "user": ""
        },
    "retcode" : 0,
    "retmsg" : "success."
      "sequence": 1
  }
```

47. External input settings

47.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	47: set external input control information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s): 10~300
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 1: yes 0: no
record_stream	int	Optional	Recording type, 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10~300
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes

capture_stream	int	Optional	Screenshot type 1: visible light only 2: infrared light only 3: infrared and visible light
sendmail	int	Optional	Send mail? 0: no 1: yes
alarm_out_flag	int	Optional	Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional	Alarm output delay: 10~300
output_flag	int	Optional	Output? 0: no 1: yes
output_hold	int	Optional	External output duration 10-300s
audio_flag	int	Optional	Audio linkage? 0: no 1: yes
audio_mode	int	Optional	Audio mode, default playback times 0: duration 1: playback times
audio_value	int	Optional	Audio mode value: 0-200 (times/second)
audio_index	int	Optional	Audio file index. The file corresponding to the index is obtained from the general configuration interface
type	int	Optional	Input type: 0: normally open type 1: normally closed type
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message

47.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":47,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "alarm_out_delay":10,
        "alarm_out_flag":1,
        "alarm_shake":10,
        "audio_flag":0,
        "audio_index":0,
        "audio_mode":1,
```

"audio_value":10,
"capture_flag":1,
"capture_stream":3,
"effect_day":[
        {
                "day":1,
                "period":[
                        {
                                "end":"23:59:59",
                                "start":"00:00:00"
                        }
                ]
        },
        {
                "day":2,
                "period":[
                        {
                                "end":"23:59:59",
                                "start":"00:00:00"
                        }
                ]
        },
        {
                "day":3,
                "period":[
                        {
                                "end":"23:59:59",
                                "start":"00:00:00"
                        }
                ]
        },
        {
                "day":4,
                "period":[
                        {
                                "end":"23:59:59",
                                "start":"00:00:00"
                        }
                ]
        },

```
                    {
                        "day":5,
                        "period":[
                            {
                                "end":"23:59:59",
                                "start":"00:00:00"
                            }
                        ]
                    },
                    {
                        "day":6,
                        "period":[
                            {
                                "end":"23:59:59",
                                "start":"00:00:00"
                            }
                        ]
                    },
                    {
                        "day":7,
                        "period":[
                            {
                                "end":"23:59:59",
                                "start":"00:00:00"
                            }
                        ]
                    }
                ],
                "flag":1,
                "output_flag":0,
                "output_hold":10,
                "record_delay":10,
                "record_stream":3,
                "sendmail":0,
                "type":0
            }
        }
Reply:
{
    "action" : "response",
```

```
        "cmdtype" : 19,
        "sequence" : 1,

        "retcode" : 0,
        "retmsg" : "success."


}
```

48. External input acquisition

48.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	48: set external input control information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s): 10~300
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 1: yes 0: no
record_stream	int	Optional	Recording type, 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10~300
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes
capture_stream	int	Optional	Screenshot type 1: visible light only

			2: infrared light only 3: infrared and visible light
sendmail	int	Optional	Send mail? 0: no 1: yes
alarm_out_flag	int	Optional	Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional	Alarm output delay: 10~300
output_flag	int	Optional	Output? 0: no 1: yes
output_hold	int	Optional	External output duration 10-300s
audio_flag	int	Optional	Audio linkage? 0: no 1: yes
audio_mode	int	Optional	Audio mode, default playback times 0: duration 1: playback times
audio_value	int	Optional	Audio mode value: 0~200 (times/second)
audio_index	int	Optional	Audio file index. The file corresponding to the index is obtained from the general configuration interface
type	int	Optional	Input type: 0: normally open type 1: normally closed type
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no 1: yes

48.2 Acquisition of requests and replies

Request:

```
{
   "action":"request",
   "cmdtype":48,
   "sequence":1,
   "token":"test",
   "message":{}
}
```

Reply:

```
{
      "action":"response",
      "cmdtype":48,
```

```
"message":{
    "alarm_out_delay":10,
    "alarm_out_flag":1,
    "alarm_shake":10,
    "audio_flag":0,
    "audio_index":0,
    "audio_mode":1,
    "audio_value":10,
    "capture_flag":1,
    "capture_stream":3,
    "effect_day":[
        {
            "day":1,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":2,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":3,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":4,
```

```
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":5,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":6,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":7,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            }
        ],
        "flag":1,
        "output_flag":0,
        "output_hold":10,
        "record_delay":10,
        "record_stream":3,
```

```
            "sendmail":0,
            "type":0
        },
        "retcode":0,
        "retmsg":"success.",
        "sequence":1
    }
```

49. FTP Settings

49.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	49： set ftp information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
serverip	string	Optional	Server ip Character range: 0~64
serverport	string	Optional	Server port: 0~65535
user	string	Optional	Server user name Character range: 0~128
password	int	Optional	Server Password Character range: 0~256
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message

49.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":49,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "flag": 0,
```

```
            "password": "",
            "serverip": "",
            "serverport": 0,
            "user": ""
        }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 49,
    "sequence" : 1,


    "retcode" : 0,
    "retmsg" : "success."
    }
```

50. FTP configuration acquisition

50.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	50: obtain ftp information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
serverip	string	Optional	Server ip Character range: 0~64
serverport	string	Optional	Server port: 0~65535
user	string	Optional	Server user name Character range: 0~128
password	int	Optional	Server password Character range: 0~256
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no

			1: yes

50.2 Requests and replies

Request:

```
{
  "action":"request",
  "cmdtype":50,
  "sequence":1,
  "token":"test",
  "message":{}
}
```

Reply:

```
  {
      "action": "response",
      "cmdtype": 50,
      "message": {
              "flag": 0,
              "password": "",
              "serverip": "",
              "serverport": 0,
              "user": ""
          },
      "retcode" : 0,
      "retmsg" : "success."
       "sequence": 1
  }
```

51. Fire information settings

51.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	51: set fire information

sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s), 0~100
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 1: yes 0: no
record_stream	int	Optional	Recording type 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10~300s
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes
capture_stream	int	Optional	Screenshot type 1: visible light only 2: infrared light only 3: infrared and visible light
fire_temp	float	Optional	Fire temperature: -40~2000
sendmail	int	Optional	Send mail? 0: no 1: yes
alarm_out_flag	int	Optional	Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional	Flash time: 10-300
output_flag	int	Optional	Output? 0: no 1: yes
output_hold	int	Optional	External output duration 10~300s
area[x]	Json	Optional	Area: area 1~area 2
sensitivity	Int	Optional	Sensitivity: 0~100 Default: 50

top_left_x	Int	Optional	X coordinate in the upper left corner of the area: 1~w
top_left_y	Int	Optional	Y coordinate in the upper left corner of the area: 1~h
bo_right_x	Int	Optional	X coordinate in the lower right corner of the area: 1~w
bo_right_y	Int	Optional	Y coordinate in the lower right corner of the area: 1~h
area_name	String	Optional	Area name. It is invalid if a valid area cannot be formed within the area
detect_mode	int	Optional	Fire mode 0: both temperature alarm and image alarm 1: temperature alarm or image alarm The default value is 0
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message

51.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":51,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "alarm_out_delay":10,
        "alarm_out_flag":0,
        "alarm_shake":10,
        "detect_mode":0
        "area1":{
            "area_name":"area1",
            "bo_right_x":1,
            "bo_right_y":1,
            "sensitivity":50,
            "top_left_x":1,
            "top_left_y":1
        },
        "area2":{
            "area_name":"area",
```

```json
            "bo_right_x":1,
            "sensitivity":50,
            "top_left_x":1,
            "top_left_y":1
        },
        "area3":{
            "area_name":"area3",
            "bo_right_x":1,
            "bo_right_y":1,
            "sensitivity":50,
            "top_left_x":1,
            "top_left_y":1
        },
        "area4":{
            "area_name":"area4",
            "bo_right_x":1,
            "bo_right_y":1,
            "sensitivity":50,
            "top_left_x":1,
            "top_left_y":1
        },
        "audio_flag":1,
        "audio_index":0,
        "audio_mode":1,
        "audio_value":10,
        "capture_flag":1,
        "capture_stream":3,
        "effect_day":[
            {
                "day":1,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":2,
                "period":[
```

```
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":3,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":4,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":5,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":6,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
```

```
                    ]
                },
                {
                    "day":7,
                    "period":[
                        {
                            "end":"23:59:59",
                            "start":"00:00:00"
                        }
                    ]
                },
                {
                    "day":8,
                    "period":[
                        {
                            "end":"23:59:59",
                            "start":"00:00:00"
                        }
                    ]
                }
            ],
            "fire_temp":120,
            "flag":1,
            "output_flag":0,
            "output_hold":10,
            "record_delay":10,
            "record_flag":1,
            "record_stream":3,
            "sendmail":0
        }
    }
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 51,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."
```

}

52. Fire information acquisition

52.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	52: obtain fire information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s), 0~100
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 1: yes 0: no
record_stream	int	Optional	Recording type 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10~300s
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes
capture_stream	int	Optional	Screenshot type 1: visible light only 2: infrared light only 3: infrared and visible light
fire_temp	float	Optional	Fire temperature: -40~2000
sendmail	int	Optional	Send mail?

				0: no 1: yes
alarm_out_flag	int	Optional		Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional		Flash time: 10-300
output_flag	int	Optional		Output? 0: no 1: yes
output_hold	int	Optional		External output duration 10～300s 10~300s
area[x]	Json	Optional		Area: area 1~area 2
sensitivity	Int	Optional		Sensitivity: 0~100 Default: 50
top_left_x	Int	Optional		X coordinate in the upper left corner of the area: 1~w
top_left_y	Int	Optional		Y coordinate in the upper left corner of the area: 1~h
bo_right_x	Int	Optional		X coordinate in the lower right corner of the area: 1~w
bo_right_y	Int	Optional		Y coordinate in the lower right corner of the area: 1~h
area_name	String	Optional		Area name. It is invalid if a valid area cannot be formed within the area
detect_mode	int	Optional		Fire mode 0: both temperature alarm and image alarm 1: temperature alarm or image alarm The default value is 0
retcode	int	Required		0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required		Return message
default	int	Optional		Default parameter or not 0: no 1: yes

52.2 Acquisition of requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":52,
    "sequence":1,
```

```
        "token":"test",
        "message":{}
}
Reply:
{
        "action" : "response",
        "cmdtype" : 52,
        "sequence" : 1,
        "message" :
        {

                    "alarm_out_delay":10,
                "alarm_out_flag":0,
                "alarm_shake":10,
                "detect_mode":0
                "area1":{
                    "area_name":"area1",
                    "bo_right_x":1,
                    "bo_right_y":1,
                    "sensitivity":50,
                    "top_left_x":1,
                    "top_left_y":1
                },
                "area2":{
                    "area_name":"area",
                    "bo_right_x":1,
                    "sensitivity":50,
                    "top_left_x":1,
                    "top_left_y":1
                },
                "area3":{
                    "area_name":"area3",
                    "bo_right_x":1,
                    "bo_right_y":1,
                    "sensitivity":50,
                    "top_left_x":1,
                    "top_left_y":1
                },
                "area4":{
                    "area_name":"area4",
```

```json
            "bo_right_x":1,
            "bo_right_y":1,
            "sensitivity":50,
            "top_left_x":1,
            "top_left_y":1
    },
    "audio_flag":1,
    "audio_index":0,
    "audio_mode":1,
    "audio_value":10,
    "capture_flag":1,
    "capture_stream":3,
    "effect_day":[
        {
            "day":1,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":2,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":3,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
```

```json
    {
        "day":4,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":5,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":6,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":7,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":8,
        "period":[
            {
```

```
                    "end":"23:59:59",

                    "start":"00:00:00"

                }

            ]

        }

    ],

    "fire_temp":120,

    "flag":1,

    "output_flag":0,

    "output_hold":10,

    "record_delay":10,

    "record_flag":1,

    "record_stream":3,

    "sendmail":0


  }

  "retcode" : 0,

  "retmsg" : "success."


}
```

53. Ranging settings

53.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message
cmdtype	int	Required	53: set the ranging parameter
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
high	Float	Optional	Installation height: 0~100
alpha	Float	Optional	Installation angle: 0~360

retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message

53.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":53,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "flag": 0,
        "high": 0,
        "alpha": 0,

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 53,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."
    }
```

54. Ranging acquisition

54.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	54: obtain the ranging parameter
sequence	uint64	Required	Message serial number

token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
high	Float	Optional	Installation height: 0~100
alpha	Float	Optional	Installation angle: 0~360
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
default	int	Optional	Default parameter or not 0: no 1: yes

54.2 Requests and replies

Request:

```
{
   "action":"request",
   "cmdtype":54,
   "sequence":1,
   "token":"test",
   "message":{}
}
```

Reply:

```
   {
      "action": "response",
      "cmdtype": 54,
      "message": {
            "flag": 0,
            "high": 0,
            "alpha": 0,
      },
   "retcode" : 0,
   "retmsg" : "success."
    "sequence": 1
   }
```

55. Video occlusion settings

55.1 Field description

Field name	Type	Option	Description

action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	55: set video occlusion
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s), 0~100
effect_day	json	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 1: yes 0: no
record_stream	int	Optional	Recording type 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10~300
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes The default value is 1
capture_stream	int	Optional	Screenshot type 1: visible light only 2: infrared light only 3: infrared and visible light
sendmail	int	Optional	Send mail? 0: no 1: yes
alarm_out_flag	int	Optional	Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional	Alarm output time, 10~300s

output_flag	int	Optional	Output? 0: no 1: yes
output_hold	int	Optional	External output duration 10~300s
audio_flag	int	Optional	Audio linkage? 0: no 1: yes
audio_mode	int	Optional	Audio mode 0: duration 1: playback times The default value is 0
audio_value	int	Optional	Audio mode value: 0~200 (times/second)
audio_index	int	Optional	Audio file index
ratio	int	Optional	Trigger ratio: 0~100
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
x	int	Optional	x：1-w x: 1-w
y	int	Optional	y：1-h y: 1-h
w	int	Optional	w：1-w w: 1-w
h	int	Optional	h：1-h h: 1-h

55.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":55,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "alarm_out_delay":10,
        "alarm_out_flag":1,
        "alarm_shake":10,
        "audio_flag":0,
        "audio_index":0,
        "audio_mode":1,
        "audio_value":10,
        "capture_flag":1,
```

"capture_stream":3,
"effect_day":[
    {
        "day":1,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":2,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":3,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":4,
        "period":[
            {
                "end":"23:59:59",
                "start":"00:00:00"
            }
        ]
    },
    {
        "day":5,

```
                    "period":[
                        {
                            "end":"23:59:59",
                            "start":"00:00:00"
                        }
                    ]
                },
                {
                    "day":6,
                    "period":[
                        {
                            "end":"23:59:59",
                            "start":"00:00:00"
                        }
                    ]
                },
                {
                    "day":7,
                    "period":[
                        {
                            "end":"23:59:59",
                            "start":"00:00:00"
                        }
                    ]
                }
            ],
            "flag":1,
            "output_flag":0,
            "output_hold":10,
            "record_delay":10,
            "record_stream":3,
            "sendmail":0,
            "ratio":0
        }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 19,
    "sequence" : 1,
```

```
"retcode" : 0,

"retmsg" : "success."


}
```

56. Video occlusion acquisition

56.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	56: obtain video occlusion
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s), 0~100
effect_day	json	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 1: yes 0: no
record_stream	int	Optional	Recording type 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10~300
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes The default value is 1
capture_stream	int	Optional	Screenshot type 1: visible light only

| | | | | 2: infrared light only |
				3: infrared and visible light
sendmail	int		Optional	Send mail?
				0: no
				1: yes
alarm_out_flag	int		Optional	Output an alarm (flash)?
				0: no
				1: yes
alarm_out_delay	int		Optional	Alarm output time,
				10~300s
output_flag	int		Optional	Output?
				0: no
				1: yes
output_hold	int		Optional	External output duration
				10~300s
audio_flag	int		Optional	Audio linkage?
				0: no
				1: yes
audio_mode	int		Optional	Audio mode
				0: duration
				1: playback times
				The default value is 0
audio_value	int		Optional	Audio mode value: 0~200 (times/second)
audio_index	int		Optional	Audio file index
ratio	int		Optional	Trigger ratio: 0~100
retcode	int		Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string		Required	Return message
x	int		Optional	x：1-w
				x:1-w
y	int		Optional	y：1-h
				y: 1-h
w	int		Optional	w：1-w
				w: 1-w
h	int		Optional	h：1-h
				h: 1-h
default	int		Optional	Default parameter or not
				0: no
				1: yes

56.2 Acquisition of requests and replies

Request:

{

   "action":"request",

   "cmdtype":56,

```
        "sequence":1,
        "token":"test",
        "message":{}
}
Reply:
    {
            "action":"response",
            "cmdtype":56,
            "message":{
                "alarm_out_delay":10,
                "alarm_out_flag":1,
                "alarm_shake":10,
                "audio_flag":0,
                "audio_index":0,
                "audio_mode":1,
                "audio_value":10,
                "capture_flag":1,
                "capture_stream":3,
                "effect_day":[
                    {
                        "day":1,
                        "period":[
                            {
                                "end":"23:59:59",
                                "start":"00:00:00"
                            }
                        ]
                    },
                    {
                        "day":2,
                        "period":[
                            {
                                "end":"23:59:59",
                                "start":"00:00:00"
                            }
                        ]
                    },
                    {
                        "day":3,
                        "period":[
```

```
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":4,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":5,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":6,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":7,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
```

```
                    ]
              }
        ],
        "flag":1,
        "output_flag":0,
        "output_hold":10,
        "record_delay":10,
        "record_stream":3,
        "sendmail":0,
        "ratio":0
    },
    "retcode":0,
    "retmsg":"success.",
    "sequence":1
}
```

57. Window contamination settings

57.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	57: obtain window contamination information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no 1: yes
alarm_shake	int	Optional	Alarm shake (s) 0~100
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 0: no

			1: yes
record_stream	int	Optional	Recording type 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10-300
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes
capture_stream	int	Optional	Screenshot type 1: visible light only 2: infrared light only 3: infrared and visible light
sendmail	int	Optional	Send mail? 0: no 1: yes
alarm_out_flag	int	Optional	Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional	Flash output time, 10~300s
output_flag	int	Optional	Output? 0: no 1: yes
output_hold	int	Optional	External output duration 10-300s
audio_flag	int	Optional	Audio linkage? 0: no 1: yes
audio_mode	int	Optional	Audio mode 0: duration 1: playback times
audio_value	int	Optional	Audio mode value: 0~200 (times/second)
audio_index	int	Optional	Audio file index
ratio	int	Optional	Trigger ratio: 0~100
retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
x	int	Optional	x：1-w x: 1-w
y	int	Optional	y：1-h y: 1-h
w	int	Optional	w：1-w w: 1-w

h	int	Optional	h：1-h h: 1-h

57.2 Requests and replies

Request:

```json
{
    "action":"request",
    "cmdtype":57,
    "sequence":1,
    "token":"8f7d83404fc011ebb4c18eeaf505828c"
    "message":{
        "alarm_out_delay":10,
        "alarm_out_flag":1,
        "alarm_shake":10,
        "audio_flag":0,
        "audio_index":0,
        "audio_mode":1,
        "audio_value":10,
        "capture_flag":1,
        "capture_stream":3,
        "effect_day":[
            {
                "day":1,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
                "day":2,
                "period":[
                    {
                        "end":"23:59:59",
                        "start":"00:00:00"
                    }
                ]
            },
            {
```

```
            "day":3,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":4,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":5,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":6,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":7,
            "period":[
                {
                    "end":"23:59:59",
```

                "start":"00:00:00"
            }
        ]
    }
],
"flag":1,
"output_flag":0,
"output_hold":10,
"record_delay":10,
"record_stream":3,
"sendmail":0,
"ratio":0
    }
}

Reply:

{
    "action" : "response",
    "cmdtype" : 57,
    "sequence" : 1,

    "retcode" : 0,
    "retmsg" : "success."

}

58. Window contamination acquisition

58.1 Field description

Field name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	58: obtain window contamination information
sequence	uint64	Required	Message serial number
token	string	Required	User access token
message	json object	Required	Message body node, used to define the request message
flag	int	Optional	Enabled? 0: no

			1: yes
alarm_shake	int	Optional	Alarm shake (s) 0~100
effect_day	json array	Optional	Time array
day	int	Optional	1~7, day of week
period	json	Optional	Time period
start	string	Optional	Start time. Format: HH:mm:ss
end	string	Optional	End time. Format: HH:mm:ss
record_flag	int	Optional	Record? 0: no 1: yes
record_stream	int	Optional	Recording type 1: visible light only 2: infrared light only 3: infrared and visible light
record_delay	int	Optional	Recording delay: 10-300
capture_flag	int	Optional	Take a screenshot? 0: no 1: yes
capture_stream	int	Optional	Screenshot type 1: visible light only 2: infrared light only 3: infrared and visible light
sendmail	int	Optional	Send mail? 0: no 1: yes
alarm_out_flag	int	Optional	Output an alarm (flash)? 0: no 1: yes
alarm_out_delay	int	Optional	Flash output time, 10~300s
output_flag	int	Optional	Output? 0: no 1: yes
output_hold	int	Optional	External output duration 10-300s
audio_flag	int	Optional	Audio linkage? 0: no 1: yes
audio_mode	int	Optional	Audio mode 0: duration 1: playback times
audio_value	int	Optional	Audio mode value: 0~200 (times/second)
audio_index	int	Optional	Audio file index
ratio	int	Optional	Trigger ratio: 0~100

retcode	int	Required	0: succeeded, otherwise: failed. See the annex for the error code
retmsg	string	Required	Return message
x	int	Optional	x：1-w x: 1-w
y	int	Optional	y：1-h y: 1-h
w	int	Optional	w：1-w w: 1-w
h	int	Optional	h：1-h h: 1-h
default	int	Optional	Default parameter or not 0: no 1: yes

58.2 Acquisition of requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":58,
    "sequence":1,
    "token":"test",
    "message":{}
}
```

Reply:

```
    {
        "action":"response",
        "cmdtype":58,
        "message":{
            "alarm_out_delay":10,
            "alarm_out_flag":1,
            "alarm_shake":10,
            "audio_flag":0,
            "audio_index":0,
            "audio_mode":1,
            "audio_value":10,
            "capture_flag":1,
            "capture_stream":3,
            "effect_day":[
                {
                    "day":1,
```

            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":2,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":3,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":4,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":5,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"

```json
                }
            ]
        },
        {
            "day":6,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        },
        {
            "day":7,
            "period":[
                {
                    "end":"23:59:59",
                    "start":"00:00:00"
                }
            ]
        }
    ],
    "flag":1,
    "output_flag":0,
    "output_hold":10,
    "record_delay":10,
    "record_stream":3,
    "sendmail":0,
    "ratio":0
},
"retcode":0,
"retmsg":"success.",
"sequence":1
}
```

IV. Functional API interfaces

500. Obtain general information

500.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	500: obtain general information
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
datetime	string	Optional	System time. Format: 2020-05-21 12:55:12. All clients use this time to refresh in real time
language	string	Optional	System language, including Chinese and English options
control_port	int	Optional	Command interaction service port
control_uri	string	Optional	Command interaction service uri address, such as/api/v1/control
ir_video_port	int	Required	Infrared websocket service port. The default value is 9981
vl_video_port	int	Required	Visible light websocket service port. The default value is 9982
audiofile	json	Optional	Music file configuration
index	int	Optional	File No.
name	string	Optional	Music file identification name
path	string	Optional	Music file path
path1	string	Optional	Music file path
systemname	string	Optional	System name
ir_mode	int	Optional	Infrared mode 0 - single-light-source infrared 1 - fusion infrared
logo	string	Optional	Base64 format system logo
copyright	string	Optional	Copyright
rule_flush_intveral	int	Optional	Time interval for obtaining object

			temperature data when the rule is configured
gears	array	Optional	Temperature measuring gear array, returned as needed. For example, if there are only two temperature measuring gears, two will be returned; if no temperature measuring gear is needed, nothing will be returned
max	float	Optional	Highest value
min	float	Optional	Minimum value
vl_rtsp_url	string	Optional	Rtsp address of the main stream of visible light
vl_sub_rtsp_url	string	Optional	Rtsp address of the sub stream of visible light
ir_rtsp_url	string	Optional	Rtsp address of the main stream of infrared light
ir_sub_rtsp_url	string	Optional	Rtsp address of the sub stream of infrared light
ir_model_w	int	Optional	Infrared module width
ir_model_h	int	Optional	Infrared module height
ir_output_w	int	Optional	Infrared channel output width
ir_output_h	int	Optional	Infrared channel output height
device_type	int	Optional	Infrared channel output height
channel_type	int	Optional	Infrared channel output height

500.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":500,
    "sequence":1,
    "message":{
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 500,
    "sequence" : 1,
    "message" :
    {
        "audiofile" :
        [
            {
```

            "index" : 0,
            "name" : "DUDU",
            "path" : "/sound/0.wav",
            "path1" : "/sound/0.mp3"
        },
        {
            "index" : 1,
            "name" : "DIDI",
            "path" : "/sound/1.wav",
            "path1" : "/sound/1.mp3"
        },
        {
            "index" : 2,
            "name" : "DUDI",
            "path" : "/sound/2.wav",
            "path1" : "/sound/2.mp3"
        }
    ],
    "control_port" : 9980,
    "control_uri" : "/api/v1/control",
    "copyright" : "",
    "datetime" : "2021-01-06 22:22:40",
    "ir_mode" : 0,
    "ir_video_port" : 9982,
    "language" : "chinese",
    "logo" : "",
    "vl_video_url": " ",
    "vl_sub_video_url":" ",
    "ir_video_url":" ",
    "ir_sub_video_url":" ",

    "rule_flush_intveral" : 60,
    "systemname" : "",
    "vl_video_port" : 9981,
    "ir_model_w " : 256,
    "ir_model_h " : 192,
    "ir_output_w" : 512,
    "ir_output_h " : 384,
    "gears":[
        {"max":150, "min":-20},

```
            {"max":400, "min":150}

        ]

    }

    "retcode" : 0,

    "retmsg" : "success."


}
```

501 Login

501.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	501: login
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
username	string	Required	User name
password	string	Required	User password, ciphertext format
count	int	Optional	Remaining login times, valid in case of login failure
token	string	Required	User access token. Subsequent access requests should carry this token as the unique credential for authentication.
host_name	string	Optional	Host name, required for connecting with the integrated management platform
ipaddr	string	Optional	Network ip address (xxx.xxx.xxx.xxx), required for connecting with the integrated management platform
netmask	string	Optional	Subnet mask (xxx.xxx.xxx.xxx), required for connecting with the integrated management platform
gateway	string	Optional	Gateway (xxx.xxx.xxx.xxx), required for connecting with the integrated management platform
dns1	string	Optional	DNS server (xxx.xxx.xxx.xxx), required for connecting with the integrated management platform
dns2	string	Optional	DNS server (xxx.xxx.xxx.xxx), required for connecting with the integrated management platform
mac	string	Optional	Mac address, required for connecting with the integrated management platform
card	int	Optional	Network card type: 0: wired network card, required for connecting with the integrated management platform

mode	int	Optional	Mode: 0: static, 1: dynamic, required for connecting with the integrated management platform
ip_version	int	Optional	Version: 0: ipv4, 1: ipv6, required for connecting with the integrated management platform
device_name	string	Optional	Equipment name, required for connecting with the integrated management platform
device_type	int	Optional	1: nc200, 2: IPT device, 3: box camera, 4: ir236, 5: device conforming to the national standard GB 28181, 6: onvif access device, required for connecting with the integrated management platform
token	string	Optional	User access token. Subsequent access requests should carry this token as the unique credential for authentication. It is required to connect with the integrated management platform.
new_username	string	Optional	User name. The device side needs to check whether the returned field is the same as the name used for current device login. If not, update the user name locally, and log into the device again with the new user name. It is required for connecting with the integrated management platform.
vl_rtsp_url	string	Optional	Rtsp address of the main stream of visible light. It can be null if there is no visible light device
vl_sub_rtsp_url	string	Optional	Rtsp address of the sub stream of visible light. It can be null if there is no visible light device
ir_rtsp_url	string	Optional	Rtsp address of the main stream of infrared light
ir_sub_rtsp_url	string	Optional	Rtsp address of the sub stream of infrared light
channel_type	int	Optional	0: dual channel, 1: visible light channel, 2: infrared channel, required for connecting with the integrated management platform

501.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":501,
    "sequence":1,
    "message":{
        "username":"admin"
        "password":"admin"
    }
}
```

Reply:

```
{
    "action" : "response",
```

```
    "cmdtype" : 501,

    "sequence" : 1,

    "message" :

    {

        "count":3，

        "token":"uuid"

    },

    "retcode" : 0,

    "retmsg" : "success."


}
```

502 Logout

502.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	502: logout
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
username	string	Required	User name
token	string	Required	User token

502.2 Requests and replies

Request:

```
{

    "action":"request",

    "cmdtype":502,

    "sequence":1,

    "token":"",

    "message":{

    }
```

```
}
Reply:
{
    "action" : "response",
    "cmdtype" : 502,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"
}
```

503 Change password

503.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	503: change the password
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
username	string	Required	User name
token	string	Required	User token
oldpassword	string	Required	User's original password
newpassword	string	Required	User's new password

503.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":503,
    "sequence":1,
    "token":"",
    "message":{
        "username":"admin",
        "oldpassword":"admin",
        "newpassword":"admin"
```

```
        }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 503,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"
}
```

504 Reset password

504.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	504: reset the password
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
username	string	Required	User name
token	string	Required	User token

504.2 Requests and replies

Request:
```
{
    "action":"request",
    "cmdtype":504,
    "sequence":1,
    "token":"",
    "message":{
        "username":"admin"
    }
}
```
Reply:
```
{
```

```
    "action" : "response",
    "cmdtype" : 504,
    "sequence" : 1,


    "retcode":0,
    "retmsg":"success"


}
```

505 Acquisition of system version information

505.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	505: get the system version
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
model	string	Optional	Equipment model
version	string	Optional	System version
serial	string	Optional	Serial number
fpga_version	string	Optional	FPGA version
measure_version	string	Optional	Temperature measuring version

505.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":505,
    "sequence":1,
    "token":"",
    "message":{
```

```
        }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 505,
    "sequence" : 1,
    "message" :
    {
            "model" : "" ,
            "version" : "",
            "serial" : "",
            "fpga_version" : "",
            "measure_version" : ""


    },
    "retcode" : 0,
    "retmsg" : "success."

}
```

506 System time synchronization

506.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	506: time synchronization
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
datetime	string	Required	Synchronize system time: time format: 2020-05-21 12:22:33

506.2 Requests and replies

Request:

{

```
    "action":"request",
    "cmdtype":506,
    "sequence":1,
    "token":"",
    "message":{
        "datetime":"2020-05-21 12:22:33"
    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 506,
    "sequence" : 1,

    "retcode":0,
    "retmsg":"success"

}
```

507 Obtain upgraded version information

507.1 Field description

Parameter name	Type	Description
action	string	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	507: obtain the upgraded version
sequence	uint64	Message serial number
message	json object	Message body node, used to define the response message node
retcode	int	Return the call code (see the appendix)
retmsg	string	Return the call message (see the appendix)
token	string	User token
desc	string	Upgrade description
time	string	Upgrade time
version	string	Upgraded version

507.2 Requests and replies

Request:

{

```
    "action":"request",
    "cmdtype":507,
    "sequence":1,
    "token":"",
    "message":{
    }
}
```
Reply:
```
{
    "action" : "response",
    "cmdtype" : 507,
    "sequence" : 1,
    "message" :
    {
        "versions" :
        [
            {
                "desc" : "",
                "time" : "2020-12-07 18:43:00\r\n",
                "version" : "v1.1.4\r\n"
            },

        ]
    }
    "retcode" : 0,
    "retmsg" : "success."

}
```

508 System restart

This interface is used to restart the system directly.

508.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	508: restart
sequence	uint64	Required	Message serial number

message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token

508.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":508,
    "sequence":1,
    "token":"",
    "message":{
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 508,
    "sequence" : 1,

    "retcode":0,
    "retmsg":"success"
}
```

509 Recording

509.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	509: recording
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node

retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
subtype	int	Required	Recording options 1: start recording now 2: delayed recording 3: cancel recording 4: query 5: delayed screen capture 6: stop delayed recording 7: stop delayed screen capture
record_stream	int	Required	Recording stream 0: query current status 1: single-light-source visible light 2: single-light-source infrared 3:double-light-source simultaneous recording
status	int	Optional	Current recording status 0: no recording 1: visible recording is in progress 2: infrared recording is in progress
time	int	Optional	The number of seconds from the start of recording to the current time
allrecord	int	Optional	0: double-light-source recording is stopped 1: double-light-source recording is in progress
alltime	int	Optional	Double-light-source recording to current seconds
delay_status	int	Optional	OptionalDelayed recording status 0: delayed recording is stopped 1: Visible light delayed recording is in progress 2: infrared delayed recording is in progress 3: prepare for visible light recording 4 prepare for infrared recording
delay_time	int	Optional	It is valid when delay_video_status is 1 or 2, and is the number of seconds from the start of the delay to the current number When delay_video_status is 3, it is the number of countdown seconds
capture_status	int	Optional	Delayed screen capture status 0: stop delayed screen capture 1: visible light delayed screen capture is in progress

			2: infrared delayed screen capture is in progress
capture_count	int	Optional	Current number of delayed screenshots
capture_total	int	Optional	Total number of delayed screenshots

509.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":509,
    "sequence":1,
    "token":"",
    "subtype":1,
    "message":{
        "record_stream"：1
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 509,
    "sequence" : 1,
    "message" :
    {
            "status":1，
            "allrecord":0,
            "time":0,
            "alltime":0,
            "delay_status":0,
            "delay_time":0,
            "capture_status":0,
            "capture_count":0,
        "capture_total":0
    }
    "retcode" : 0,
    "retmsg" : "success."

}
```

510 Log download

510.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	510: log download
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
url	string	Required	Download path. You need to add the http header yourself

510.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":510,
    "sequence":1,
    "token":"",
    "message":{
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 510,
    "sequence" : 1,
    "message" :
    {
        "url":"/tmp/log.tar.gz"
    }
    "retcode" : 0,
    "retmsg" : "success."
```

}

511 Data cleanup

This interface is used to clean up disk cache data.

511.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	511: data cleanup
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token

511.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":511,
    "sequence":1,
    "token":"",
    "message":{
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 511,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"

}
```

512 System settings import and export

512.1 Field description

The import and export of IP and user information are not supported

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	512: system settings import and export
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
subtype	int	Required	1: restore factory settings 2: export settings
url	string	Required	Download path. You need to add the http header yourself

512.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":512,
    "sequence":1,
    "token":"",
    "subtype":1,
    "message":{
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 512,
    "sequence" : 1,
    "message" :
```

```
    {
        "url":"/tmp/axxx"


    }
    "retcode" : 0,
    "retmsg" : "success."


}
```

513 Obtain historical video/picture information

513.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	513: obtain historical video and picture information
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
starttime	string	Required	Start time. Format: 2020-05-21 12:22:33
endtime	string	Required	End time. Format: 2020-05-21 12:22:33
pageindex	int	Required	Current page. Value range: >= 1
pagecount	int	Required	Number of pages
videotype	int	Required	Video type: 1: visible light 2: infrared light
datattype	int	Required	Data type 1: video 2: picture
total	int	Required	Total
datas	json array	Required	Total data
type	int	Required	File type: 1: alarm 2: cold point 3: hot point 4: cold and hot points

			5: manual 6: event 7: alarm input
url	string	Optional	File address
start	string	Optional	Start time
end	string	Optional	End time
filesize	int	Optional	File size in bytes

513.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":513,
    "sequence":1,
    "token":"",
    "message":{
        "starttime ":"2020-05-21 12:22:33",
        "endtime":"2020-05-21 12:22:33",
        "pageindex":0,
        "pagecount":10,
        "videotype":1,
        "datatype":1
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 513,
    "sequence" : 1,
    "message" :
    {
            "total":111,
            "datas":[
            {
                "type":1,
                "url":"/picture/xxx.jpg",
                "start":"2020-05-21 12:22:33",
                "end":"2020-05-21 12:22:33",
                "filesize":23123324
            }]
```

```
        }

    "retcode" : 0,

    "retmsg" : "success."


}
```

514 Obtain historical video/picture distribution information

514.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	514: obtain historical video and other distribution information
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
videotype	int	Required	Video type: 1: visible light 2: infrared light
datatype	int	Required	Data type 1: video 2: picture
year	int	Required	Year
month	int	Required	Month
days	array	Required	Days with data: 1<= x<= 31

514.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":514,
    "sequence":1,
    "token":"",
    "message":{
        "videotype":0,
        "datatype",1,
        "year",2020
```

```
        "month",5
    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 514,
    "sequence" : 1,
    "message" :
    {
        "days":[1,2,3,4,5,6,7,4]
    },
    "retcode" : 0,
    "retmsg" : "success."

}
```

515 Obtain object temperature list

515.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message Notify: it refers to a notification message. The receiver may need no response when receiving such a message
cmdtype	int	Required	515: obtain the object temperature list
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
query_start_time	string	Required	Query start time. Format: 2020-05-21 12:22:33
query_end_time	string	Required	Query end time. Format: 2020-05-21 12:22:33
query_interval	int	Required	Time interval in minutes

query_type	int	Required	Query type, bitwise 1: point 2: line 4: oval 8: rectangle 16: polygon
query_temp_type	int	Required	Temperature type: 1: maximum temperature 2: minimum temperature 3: average temperature
report	int	Required	Export? 0: no 1: yes (only focus on the export address when exporting)
page_index	int	Required	Page index
page_count	int	Required	Number of pages
measure_list	json obj	Required	Temperature measuring object list
measure_type	int	Required	Temperature measuring object type 1: point 2: line 3: rectangle 4: polygon 5: circle
measure_id	int	Optional	Temperature measuring object id
measurer_name	string	Optional	Temperature measuring object name
temperature	array	Optional	Temperature list of the temperature measuring object. Null is supported. For example: [null,1,null,2,null,3,null,null]
report_path	string	Optional	Export address Http header splicing is required for access
total	int	Optional	Total number of records in time
start_time	string	Optional	Query record start time. Format: yyyy-MM-dd HH:mm:ss
end_time	string	Optional	Query record end time. Format: yyyy-MM-dd HH:mm:ss

515.2 Requests and replies

Request:

{

    "action":"request",

    "cmdtype":515,

    "sequence":1,

    "token":"",

    "message":{

```
            "query_start_time":"2020-05-21 12:22:33",

            "query_end_time":"2020-05-21 12:22:33",

            "query_type"： 2，

            "query_interval":2，

            "query_temp_type"： 1,

            "report":0,

            "page_index":1

            "page_count":2

        }

    }

}

Reply:

{

    "action" : "response",

    "cmdtype" : 515,

    "sequence" : 1,

    "message" :

    {

        "report_path"： "/report/2321434.cvs",

        "total":1,

        "start_time":"2020-05-21 12:22:33",

        "end_time":"2020-05-21 12:22:33",

        "measure_list":[

            {

                "measure_type":2,

                "measure_id":213,

                "measurer_name":"point0",

                "temperature":[12,23.21,2132.1,2,2.....,23]

                "report_path"： "/report/2321434.cvs"

            }

        ]

    }

    "retcode" : 0,

    "retmsg" : "success."


}
```

516 Subscribe to/unsubscribe from real-time report

516.1 Field description

Parameter name	Type	Option	Description

action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	516: subscribe to/unsubscribe from the real-time report
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
push_report	int	Required	Receive the real-time report? 0: no 1: yes
interval	int	Optional	Time period

516.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":516,
    "sequence":1,
    "token":"",
    "message":{
        "push_report":1
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 516,
    "sequence" : 1,
    "message" :
    {
        "interval":1,
    }
    "retcode" : 0,
    "retmsg" : "success."

}
```

517 Set real-time report period

517.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	517: set the report period
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
interval	int	Required	Push the real-time report period in minutes

517.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":517,
    "sequence":1,
    "token":"",
    "message":{
        "interval":1
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 517,
    "sequence" : 1,

    "retcode":0,
    "retmsg":"success"

}
```

518 Image effect settings

518.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	518: image effect settings
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
params	json object	Required	Message body node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
type	int	Required	1: visible light brightness setting: 0-100 2: visible light contrast setting: 0-100 3: visible light saturation setting: 0-100 4: visible light sharpness setting: 1-100 5: visible light image setting: 0-100 6: visible light flip setting: 0-1 7: infrared brightness setting: 0-100 8: infrared contrast setting: 0-100 9: infrared saturation setting: 0-100 10: infrared sharpness setting: 1-100 11: infrared shutter compensation. You can choose not to transmit the value 12: infrared flip setting: 0-1 13: visible light backlight compensation: 0-6 14: wide dynamic setting: 0-5 15: strong light suppression switch: 0-1 16: exposure compensation: 0-100 17: infrared time domain noise reduction: 0-100 18: infrared spatial domain noise reduction: 0-100 19: infrared detail enhancement: 0-100
value	int	Optional	See type

518.2 Requests and replies

Request:

{

    "action":"request",

    "cmdtype":518,

    "sequence":1,

    "token":"",

    "subtype":1,

```
    "message":{
        "params":[
        {
            "type":0,
            "value":0
        }
        ]
    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 518,
    "sequence" : 1,


    "retcode":0,
    "retmsg":"success"


}
```

520 Obtain object real-time temperature

520.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	520: obtain real-time temperature
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
rule_list	int	Required	Rule id
rule_id	int	Optional	Rule id
rule_name	string	Optional	Rule name
type	int	Optional	Object type: 1: point 2: line

			3: rectangle 4: polygon 5: circle
max_temp	float	Optional	Maximum temperature value
min_temp	float	Optional	Minimum temperature value
avg_temp	float	Optional	Average temperature value
preset	int	Optional	Preset position, subject to actual equipment support. It is an object attribute. Range: [1,256]

520.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":520,
    "sequence":1,
    "token":"",
    "message":{

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 520,
    "sequence" : 1,
    "message" :
    {
        rule_list:[
        {
            "rule_id":1,
            "rule_name":"point1",
            "type":1,
            "max_temp":23.1
            "min_temp":23.1
            "max_temp":23.1

        }
        ]
    }
    "retcode":0,
```

"retmsg":"success"

}

521 Point temperature acquisition

This interface is used to obtain point temperature. It can be used for a real-time cursor display of temperature (according to the return need), and can also be used for other functions.

521.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	521: obtain point temperature
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
cursor_x	int	Required	Real-time x coordinate of the cursor. If it is less than 0, it means that the cursor display of the temperature measurement is automatically turned off. The effective range is 1-512,
cursor_y	int	Required	Real-time y coordinate of the cursor. If it is less than 0, it means that the cursor display of the temperature measurement is automatically turned off. The effective range is 1-384,
cursor_type	int	Required	Coordinate system type of the cursor. 0 means setting the cursor position with the infrared image as the coordinate system, and 1 means setting the cursor position with the visible light image as the coordinate system. The default value is 0. Currently only the infrared image is supported.
value	float	Required	Input point temperature.
show	int	Required	Return displayable: 0: no 1: yes

521.2 Requests and replies

Request:

{

```
    "action":"request",
    "cmdtype":521,
    "sequence":1,
    "token":"",
    "message":{
        "cursor_type":0,
        "cursor_x":111,
        "cursor_y":22
    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 521,
    "sequence" : 1,
    "message" :
    {
        "value":23.3，
        "show":1，
    }
    "retcode":0,
    "retmsg":"success"
}
```

522 Obtain thermal image

522.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	522: obtain the thermal image
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token

image_url	string	Required	Thermal image url
image_content	string	Optional	Thermal image base64 data
image_width	string	Optional	Thermal image width
image_height	string	Optional	Thermal image height

522.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":522,
    "sequence":1,
    "token":"",
    "message":{

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 522,
    "sequence" : 1,
    "message" :
    {
        "image_url"："aaa.data"
    }
    "retcode":0,
    "retmsg":"success"
}
```

523 Obtain screenshot

523.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	523: obtain the screenshot

sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
type	int	Required	Type 1: visible light. Format: JPEG base64 2: infrared original image 256 module 256 x192. Format: RGB. RGB888 data in the base64 format are returned 3: infrared original image. Format: RGBA8888 base64
image_content	string	Required	IIMG base64 encoded string
image_width	int	Optional	IMG width
image_height	int	Optional	IMG height

523.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":523,
    "sequence":1,
    "token":"",
    "message":{
        "type":0,
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 523,
    "sequence" : 1,
    "message" :
    {
        "image_content"： ""
    }
    "retcode":0,
    "retmsg":"success"
}
```

524 Turn off alarm output/sound

This interface is used to turn off the current alarm (e.g., double-source light) or sound (e.g., music), and the status does not remain.

524.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	524: turn off alarm output
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
type	int	Optional	It is set according to the actual demand. The output is turned off by default 0: turn off the sound only 1: turn off the current alarm output 2: turn off the sound and the current alarm

524.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":524,
    "sequence":1,
    "token":"",
    "message":{
        "type":0,
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 524,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"
}
```

525 Mail test

525.1 字段 525.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	525: mail test
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token

525.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":525,
    "sequence":1,
    "token":"",
    "message":{

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 525,
    "sequence" : 1,

    "retcode":0,
    "retmsg":"success"

}
```

527 Mark dead pixels

527.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	527: mark dead pixels
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
subtype	int	Required	Receive the real-time report? 0: mark dead pixels 1: restore to the original state in the factory 2: delete marks
points	json array	Required	Point array
x	int	Required	x: 0-255
y	int	Required	y: 0-191

527.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":527,
    "sequence":1,
    "token":"",
    "subtype":0，
    "message":{
        "points":[
            {
                "x":1,
                "y",2
            }
        ]

    }
}
```

Reply:

```
{
```

```
    "action" : "response",

    "cmdtype" : 527,

    "sequence" : 1,


    "retcode":0,

    "retmsg":"success"


}
```

528 Obtain dead pixel list

528.1 field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	528: obtain the dead pixel list
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
points	json array	Required	Point array
x	int	Required	x: 0-255
y	int	Required	y: 0-191

528.2 Requests and replies

Request:

```
{

    "action":"request",

    "cmdtype":528,

    "sequence":1,

    "token":"",

    "subtype":0，

    "message":{

     }

}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 528,
    "sequence" : 1,
    "message" :
    {
        "points":[
            {"x": 65, "y": 100}
        ]
    }
    "retcode":0,
    "retmsg":"success"
}
```

529 NUC operation

529.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	529 NUC operation
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token

529.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":529,
    "sequence":1,
    "token":"",
    "message":{
```

```
        }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 529,
    "sequence" : 1,
    "message" :
    {
    }
    "retcode":0,
    "retmsg":"success"
}
```

530 Obtain temperature matrix

530.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	530: obtain the temperature matrix
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
type	int	Optional	0 is 512x384 1 is 384x288
width	int	Optional	Return width
height	int	Optional	Return height
temps	string	Required	base64 string: Format: base64 (infrared channel output w * infrared channel output h * 2 bytes). The 2 bytes are an integer obtained from amplifying the point temperature by 100 times

530.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":530,
    "sequence":1,
    "token":"",
    "message":{
    "type":0
     }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 530,
    "sequence" : 1,
    "message" :
    {
        "temps" : "base64"
    }
    "retcode":0,
    "retmsg":"success"
}
```

531 Obtain point distance

531.1 Field description

Parameter name	Type	Option	Description
action	string	Required	Message type: Request: it refers to the request message. The receiver needs to respond with a response message when receiving such a message Response: it refers to the response message
cmdtype	int	Required	531: obtain point distance
sequence	uint64	Required	Message serial number
message	json object	Required	Message body node, used to define the response message node
retcode	int	Required	Return the call code (see the appendix)
retmsg	string	Required	Return the call message (see the appendix)
token	string	Required	User token
x	int	Optional	X coordinate: 1-w. 0 is supported by

			default, and 0 is equivalent to 1
y	int	Optional	y coordinate: 1-h. 0 is supported by default, and 0 is equivalent to 1
x	float	Optional	X-direction distance
y	float	Optional	Y-direction distance

531.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":531,
    "sequence":1,
    "token":"",
    "message":{
    "x":0,
    "y":0

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 531,
    "sequence" : 1,
    "message" :
    {
    "x":0,
    "y":0

    }
    "retcode":0,
    "retmsg":"success"
}
```

600. PTZ operation

600.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message

			response: it refers to the response message
cmdtype	int	Required	600 PTZ control operation
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
ptz_command	int	Required	1: light operation 2: wiper operation 3: fan operation 4: heating operation 5: operation of expansion auxiliary equipment 6: zoom in 7: zoom out 8: near-focus adjustment 9: far-focus adjustment 10: aperture enlargement 11: aperture reduction 12: PTZ up 13: PTZ down 14: PTZ left 15: PTZ right 16: PTZ up and left 17: PTZ up and right 18: PTZ down and left 19: PTZ down and right 20: PTZ up and zoom in 21: PTZ up and zoom out 22: PTZ down and zoom in 23: PTZ down and zoom out 24: PTZ left and zoom in 25: PTZ left and zoom out 26: PTZ right and zoom in 27: PTZ right of and zoom in 28: PTZ up and left and zoom in 29: PTZ up and left and zoom out 30: PTZ up and right and zoom in 31: PTZ up and right and zoom out 32: PTZ down and left and zoom in 33: PTZ down and left and zoom out 34: PTZ down and right and zoom in 35: PTZ down and right and zoon out The above instructions shall be subject to the functions realized by the equipment end, and the unsupported equipment will not be realized temporarily

ptz_action	int	Required	0: Start, 1: Stop, only for PTZ needs to start and stop control instructions. If PTZ does not need to start and stop operations, it shall be subject to the actual operation of the PTZ. For PTZ movement, zoom, focus, aperture and other operations, if the device does not support continuous movement, the device only takes effect in real time for parameter of 0. If continuous operation is supported, 0 starts execution and 1 ends execution.
ptz_param1	int	Required	Control parameter, this parameter is effective according to actual instructions, for example, if the step length of PTZ when rotating is 5, then this parameter device is 5, if
ptz_param2	int	Required	For extension parameter, the default Parameter is 0
ptz_param3	int	Required	For extension parameter, the default Parameter is 0
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

600.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":600,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "ptz_command" : 1,
        "ptz_action" : 0,
        "ptz_param1" : 5,
        "ptz_param2" : 0,
        "ptz_param3" : 0,
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 600,
    "sequence" : 1,
    "from":"client1",
```

```
        "to":"device1",

        "retcode":0,

        "retmsg":"success"


}
```

601. PTZ preset acquisition

601.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	601 PTZ preset acquisition
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
presets	json array	Required	An array of presets, possibly returning multiple presets
index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
name	string	Required	Preset name
param	int	Required	Preset extension parameter, default to 0, used for subsequent extensions
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

601.2 Requests and replies

Request:

```
{

    "action":"request",

    "cmdtype":601,

    "sequence":1,

    "from":"client1",

    "to":"device1",

    "token":"",
```

```
    “message”:{

    }
}
Reply:
{
    “action” : “response”,
    “cmdtype” : 601,
    “sequence” : 1,
    “from”:”client1”,
    “to”:”device1”,
    “retcode”:0,
    “retmsg”:”success”
    “message”:{
        “presets”:[
            {
             “index”:1,
             “name”: “预置位 1”,
             'Name': 'preset 1 ",
             “param”:0
            },
            {
             “index”:2,
             “name”: “预置位 12”,
             'Name': 'preset 12 ",
             “param”:0
            }
        ]
    }
}
```

602. Add PTZ preset

602.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	602 adding presets of PTZ or the instruction

			to add, is allowed to add multiple bits at a time, and one is added by default. If the device does not allow multiple bits to be added, it will be added individually when used.
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
presets	json array	Required	An array of presets, possibly returning multiple presets
index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
name	string	Required	Preset name
param	int	Required	Preset extension parameter, default to 0, used for subsequent extensions
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

602.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":602,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "presets":[
            {
                "index":1,
                'name': 'preset 1 ",
                "param":0
            },
            {
                "index":2,
                'name': 'preset 12 ",
                "param":0
            }
```

```
        ]

    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 602,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{
    }

}
```

603. PTZ preset deletion

603.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	603 the instruction to delete or deleting the presets of PTZ, is allowed to delete multiple bits at a time, and one is added by default. If the device does not allow multiple bits to be added, it will be added individually when used.
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
presets	json array	Required	An array of presets, possibly returning multiple presets
index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
retcode	int	Required	Return the calling code (see appendix)

retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

603.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":603,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "presets":[
            {
                "index":1,
            },
            {
                "index":2,
            }

        ]

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 603,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{
    }

}
```

604. PTZ preset change

604.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	604 PTZ preset information change, or the instruction to change preset, is allowed to change more than one at a time, and change one at a time by default. If the device does not allow multiple change, a single change can only change preset name and preset parameter, and the preset index cannot be changed.
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
presets	json array	Required	An array of presets, possibly returning multiple presets
index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
name	string	Required	Preset name
param	int	Required	Preset extension parameter, default to 0, used for subsequent extensions
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

604.2 Requests and replies

Request:

{

    "action":"request",

    "cmdtype":604,

    "sequence":1,

    "from":"client1",

    "to":"device1",

    "token":"",

    "message":{

       "presets":[

          {

             "index":1,

```
                    'name': 'preset 1 ",
                    "param":0
                },
                {
                    "index":2,
                    'name': 'preset 12 ",
                    "param":0
                }

        ]

    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 604,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }

}
```

605. PTZ preset movement

605.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	605 the PTZ preset moves, the PTZ receives this command and moves to the preset. If the preset is invalid, it will not be executed.
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid,

			deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
presets	json array	Required	An array of presets, possibly returning multiple presets
index	int	Required	The preset index, the PTZ moves to this preset, starting from 1, with a maximum of 255. Take the View Sheen PTZ as an example.
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

605.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":605,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        index":1
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 605,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }

}
```

606. PTZ cruise group query

606.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	606 query cruise group information and return all cruise group information
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
cruise_groups	json array	Required	Cruise group node, cruise group array
cruise_index	int	Required	Cruise group index, starting from 1, the maximum number of cruise groups is 8, taking View Sheen as an example
cruise_name	string	Required	Cruise group name, current cruise group name
cruise_param	int	Required	Cruise group parameter, extended field, not currently used
presets	json array	Required	The cruise array and preset array contained in the current cruise group may return multiple presets, which can be empty, indicating that the current cruise group has no cruise point
index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
name	string	Required	Preset name
param	int	Required	Preset extension parameter, default to 0, used for subsequent extensions
duration	int	Required	Duration, unit: seconds; default value: 10 seconds; value range: 1-300, which is determined by the actual situation of PTZ
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

606.2 Requests and replies

Request:

{

    "action":"request",

    "cmdtype":606,

  "sequence":1,

  "from":"client1",

  "to":"device1",

  "token":"",

  "message":{

  }

}

Reply:

{

  "action" : "response",

  "cmdtype" : 606,

  "sequence" : 1,

  "from":"client1",

  "to":"device1",

  "retcode":0,

  "retmsg":"success"

  "message":{

    "cruise_groups": [

      {

        "cruise_index":1,

        "cruise_name": "cruise group 1",

        "cruise_param":0

        "presets":[

        {

          "index":1,

          'name': 'preset 1 ',

          "param":0

        },

        {

          "index":2,

          'name': 'preset 12 ',

          "param":0

        }

        ]

      },

      {

        "cruise_index":2,

        "cruise_name": "cruise group 22",

        "cruise_param":0

        "presets":[

```
                    {
                        "index":1,
                        'name': 'preset 1 ",
                        "param":0
                    },
                    {
                        "index":2,
                        'name': 'preset 12 ",
                        "param":0
                    }
                    ]
                }
            ]
        }

}
```

607. Add PTZ cruise group

607.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	607 add cruise group information
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
cruise_groups	json array	Required	Cruise group node, cruise group array
cruise_index	int	Required	Cruise group index, starting from 1, the maximum number of cruise groups is 8, taking View Sheen as an example
cruise_name	string	Required	Cruise group name, current cruise group name
cruise_param	int	Required	Cruise group parameter, extended field, not currently used
presets	json array	Required	The cruise array and preset array contained in the current cruise group may return multiple presets, which can be empty, indicating that the current cruise group has no cruise point

index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
name	string	Required	Preset name
param	int	Required	Preset extension parameter, default to 0, used for subsequent extensions
duration	int	Required	Duration, unit: seconds; default value: 10 seconds; value range: 1-300, which is determined by the actual situation of PTZ
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

607.2 Requests and replies

```
{
    "action":"request",
    "cmdtype":607,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "cruise_groups": [
            {
                "cruise_index":1,
                "cruise_name": "cruise group 1",
                "cruise_param":0
                "presets":[
                {
                    "index":1,
                    'name': 'preset 1 ",
                    "param":0
                },
                {
                    "index":2,
                    'name': 'preset 12 ",
                    "param":0
                        }
                ]
            },
            {
                "cruise_index":2,
```

```
                    "cruise_index": 2,
                    "cruise_name": "cruise group 22",
                "cruise_param":0
                "presets":[
                {
                    "index":1,
                    'name': 'preset 1 ",
                    "param":0
                },
                {
                    "index":2,
                    'name': 'preset 12 ",
                    "param":0
                }
                ]
            }
        ]
    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 607,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }

}
```

608. Delete PTZ cruise group

608.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message

			response: it refers to the response message
cmdtype	int	Required	608 the cruise group information is deleted, the cruise group index needs to be established.
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
cruise_groups	json array	Required	Cruise group node, cruise group array
cruise_index	int	Required	Cruise group index, starting from 1, the maximum number of cruise groups is 8, taking View Sheen as an example
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

608.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":608,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "cruise_groups": [
            {
                "cruise_index":1,
            },
            {
                "cruise_index":2,
            }
        ]
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 608,
```

```
"sequence" : 1,

"from":"client1",

"to":"device1",

"retcode":0,

"retmsg":"success"

"message":{


}


}
```

609. PTZ cruise group change

609.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	609 change cruise group information
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
cruise_groups	json array	Required	Cruise group node, cruise group array
cruise_index	int	Required	Cruise group index, starting from 1, the maximum number of cruise groups is 8, taking View Sheen as an example
cruise_name	string	Required	Cruise group name, current cruise group name
cruise_param	int	Required	Cruise group parameter, extended field, not currently used
presets	json array	Required	Revise cruise array and preset array contained in the current cruise group. Multiple presets can be added or deleted. After the equipment is executed, the cruise point number of this cruise group on the equipment side can be updated. It can be an empty array, which means to clear the current cruise group.
index	int	Required	Preset index, starting from 1, up to 255, taking View Sheen PTZ as an example
name	string	Required	Preset name

param	int	Required	Preset extension parameter, default to 0, used for subsequent extensions
duration	int	Required	Duration, unit: seconds; default value: 10 seconds; value range: 1-300, which is determined by the actual situation of PTZ
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

609.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":609,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "cruise_groups": [
            {
                "cruise_index":1,
                "cruise_name": "cruise group 1",
                "cruise_param":0
                "presets":[
                {
                    "index":1,
                    'name': 'preset 1 ",
                    "param":0
                },
                {
                    "index":2,
                    'name': 'preset 12 ",
                    "param":0
                }
                ]
            },
            {
                "cruise_index":2,
                "cruise_name": "cruise group 22",
                "cruise_param":0
```

```
            "presets":[
            {
                "index":1,
                'name': 'preset 1 ",
                "param":0
            },
            {
                "index":2,
                'name': 'preset 12 ",
                "param":0
            }
            ]
        }
    ]
}
}
Reply:
{
    "action" : "response",
    "cmdtype" : 609,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }

}
```

610. PTZ Cruise Team Patrol

610.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	610 execute cruise group information
sequence	uint64	Required	Message serial number

from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
cruise_index	int	Required	Cruise group index, starting from 1, the maximum number of cruise groups is 8, taking View Sheen as an example
cruise_action	int	Required	Cruise action, 0 means start cruise, 1 means stop cruise
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

610.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":610,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "cruise_index":1,
        "cruise_action": 0,
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 610,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }
```

}

611. PTZ linear scan settings query

611.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	611 change cruise group information
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
linescan_groups	json array	Required	Return all linear scan groups for array type
linescan_index	int	Required	Linear scan index, starting from 1, the maximum number of linear scans is 5, taking View Sheen as an example
linescan_speed	int	Required	Linear scan speed
linescan_name	string	Required	Linear scanning Name, which can be empty, subject to actual equipment support
linescan_left_enable	int	Required	In the left boundary state, 0 is disable, 1 is enable, and the left and right boundaries are both enabled before scanning can be started.
linescan_left_duration	int	Required	Left boundary duration, unit: seconds; default is 0 seconds
linescan_right_enable	int	Required	In the right boundary state, 0 is disable, 1 is enable, and the left and right boundaries are both enabled before scanning can be started.
linescan_right_duration	int	Required	Right boundary duration, unit: seconds; default is 0 seconds
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

611.2 Requests and replies

Request:

{

```
        "action":"request",
        "cmdtype":611,
        "sequence":1,
        "from":"client1",
        "to":"device1",
        "token":"",
        "message":{
        }
}
Reply:
{
        "action" : "response",
        "cmdtype" : 611,
        "sequence" : 1,
        "from":"client1",
        "to":"device1",
        "retcode":0,
        "retmsg":"success"
        "message":{
            "linescan_groups": [
                {
                    "linescan_index":1,
                    "linescan_speed":3,
                    "linescan_name": "",
                    "linescan_left_enable":3,
                    "linescan_left_duration":30,
                    "linescan_right_enable":3,
                    "linescan_right_duration":30,
                },
                {
                    "linescan_index":2,
                    "linescan_speed":3,
                    "linescan_left_enable":3,
                    "linescan_left_duration":30,
                    "linescan_right_enable":3,
                    "linescan_right_duration":30,
                }

            ]
```

```
    }

}
```

612. PTZ linear scanning change

612.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	612 change the specified linear scan configuration
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
linescan_index	int	Required	Linear scan index, starting from 1, the maximum number of linear scans is 5, taking View Sheen as an example
linescan_speed	int	Required	Linear scan speed
linescan_left_enable	int	Required	In the left boundary state, 0 is disable, 1 is enable, and the left and right boundaries are both enabled before scanning can be started.
linescan_left_duration	int	Required	Left boundary duration, unit: seconds; default is 0 seconds
linescan_right_enable	int	Required	In the right boundary state, 0 is disable, 1 is enable, and the left and right boundaries are both enabled before scanning can be started.
linescan_right_duration	int	Required	Right boundary duration, unit: seconds; default is 0 seconds
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

612.2 Requests and replies

Request:

{

    "action":"request",

  "cmdtype":612,

  "sequence":1,

  "from":"client1",

  "to":"device1",

  "token":"",

  "message":{

    "linescan_index":1,

    "linescan_speed":3,

    "linescan_left_enable":3,

    "linescan_left_duration":30,

    "linescan_right_enable":3,

    "linescan_right_duration":30

  }

}

Reply:

{

  "action" : "response",

  "cmdtype" : 612,

  "sequence" : 1,

  "from":"client1",

  "to":"device1",

  "retcode":0,

  "retmsg":"success"

  "message":{

  }

}

613. PTZ linear scanning operation

613.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	613 perform the specified linear scan operation
sequence	uint64	Required	Message serial number

from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
linescan_index	int	Required	Linear scan index, starting from 1, the maximum number of linear scans is 5, taking View Sheen as an example
linescan_action	int	Required	Linear scanning action, 0 means to start scanning, and 1 means to stop scanning
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

613.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":613,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "linescan_index":1,
        "linescan_action":0
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 613,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }
```

}

614. PTZ trace setting query

614.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	614 trace configuration query
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
trace_groups	json array	Required	Return all trace groups of array type
trace_index	int	Required	The tracking index starts from 1, and the maximum linear scanning number is 5. Take the View Sheen as an example.
trace_name	string	Required	The trace name is ignored if the equipment does not support name. Take the View Sheen as an example.
trace_enable	int	Required	Trace switch, 0 means disable, and the 1 means enable. Only when this parameter is enabled can the trace be started.
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

614.2 Requests and replies

Request:

{

    "action":"request",

    "cmdtype":614,

    "sequence":1,

    "from":"client1",

    "to":"device1",

    "token":"",

```
    "message":{
    }
}
Reply:
{
    "action" : "response",
    "cmdtype" : 614,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{
        "trace_groups": [
            {
                "trace_index":1,
                "trace_enable":1,
                "trace_name": ""
            },
            {
                "trace_index":2,
                "trace_enable":1,
                "trace_name": ""
            },

        ]

    }

}
```

615. PTZ trace change

615.1 字段 615.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	615 change the specified linear scan

			configuration
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
trace_index	int	Required	Trace index, starting from 1, the maximum trace number is 5, taking View Sheen as an example
trace_record_action	int	Required	Trace recording operation, 0 means to start trace recording, and 1 means to stop trace recording. After the trace recording is started, stop trace operation must be performed to avoid that the equipment does not automatically stop trace recording operation, resulting in failure to change the configuration of other trace records. Take View Sheen as an example.
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

615.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":615,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "trace_index":1,
        "trace_record_action":0
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 615,
    "sequence" : 1,
```

“from”:”client1”,

“to”:”device1”,

“retcode”:0,

“retmsg”:”success”

“message”:{

}

}

616. PTZ trace operation

616.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	616 change the specified the linear scan configuration
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
trace_index	int	Required	Trace index, starting from 1, the maximum tour number is 5, taking View Sheen as an example
trace_action	int	Required	Trace operation, 0 means to start trace, and 1 means to stop trace. After trace starts, stop trace operation must be performed to avoid that the equipment does not automatically stop trace operation, resulting in other tracks that cannot be executed. Only when the field trace_enable of trace switch is 1, the equipment can start trace operation. Take View Sheen as an example.
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

616.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":616,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
        "trace_index":1,
        "trace_action":0
    }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 616,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{

    }

}
```

617. PTZ motion status query

617.1 Field description

Parameter name	Type	Options	Description
action	string	Required	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	Required	617 query the current PTZ cruise status
sequence	uint64	Required	Message serial number

from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
message	json object	Required	Message body node, responding to message node definition
ptz_movement	int	Required	PTZ cruise status, 0 means no cruise, 1 means in cruising
ptz_move_type	int	Required	1: PTZ cruise movement 2: linear sweep movement 3: trace movement
ptz_move_index	int	Required	An index representing the current motion type, such as the specific execution of cruise, linear scan, or trace.
retcode	int	Required	Return the calling code (see appendix)
retmsg	string	Required	Return call message (see appendix)
token	string	Required	User token

617.2 Requests and replies

Request:

```
{
    "action":"request",
    "cmdtype":617,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":{
     }
}
```

Reply:

```
{
    "action" : "response",
    "cmdtype" : 617,
    "sequence" : 1,
    "from":"client1",
    "to":"device1",
    "retcode":0,
    "retmsg":"success"
    "message":{
        "ptz_movement":0
        "ptz_move_type":0
```

"ptz_move_index":0

    }

}

9999 Send heartbeat

9999.1 Field description

Parameter name	Type	Description
action	string	Message type: request: it refers to the request message, the receiver needs to reply to the response type message when receiving such message response: it refers to the response message
cmdtype	int	9999 send heartbeat
sequence	uint64	Message serial number
message	json object	Message body node, responding to message node definition
retcode	int	Return the calling code (see appendix)
retmsg	string	Return call message (see appendix)
token	string	User token

9999.2 Requests and replies

Request:

```
{
    "action":"notify",
    "cmdtype":9999,
    "sequence":1,
    "token":"",
    "subtype":0，
    "message":{

    }
}
```

V. HTTP POST/PUT Interface Protocol

This section is mainly used to handle file upload operations.

Define no HTTP interface

URI：/upload/apppkg/token

Method:POST/PUT

Conten: source data

The style is as follows:

▼ General

    Request URL: http://192.168.1.168/upload/apppkg/a412a60cae6811ec8cf800151810ca9d

    Request Method: POST

    Status Code: ● 200 OK

    Remote Address: 192.168.1.168:80

    Referrer Policy: strict-origin-when-cross-origin

▼ **Response Headers**   View source

    Connection: keep-alive

    Content-Encoding: gzip

    Content-Type: text/html

    Date: Mon, 28 Mar 2022 07:29:20 GMT

    Server: nginx/1.18.0

    Transfer-Encoding: chunked

▼ **Request Headers**   View source

    Accept: */*

    Accept-Encoding: gzip, deflate

    Accept-Language: zh-CN,zh;q=0.9

    Connection: keep-alive

    Content-Length: 535339

    Content-Type: multipart/form-data; boundary=-----WebKitFormBoundaryf2Bev6O7mGFGTXps

    Host: 192.168.1.168

    Origin: http://192.168.1.168

    Referer: http://192.168.1.168/

The returned result is described as follows, without token in the identifier

1、 /upload/lowpkg

Upload low temp file packet

Return:

    {

    "action" : "response",

    "cmdtype" : 751,

    "sequence" : 1,

    "retcode":0,

    "retmsg":"success"

    }

2、 /upload/highpkg

Upload high temp file packet

Return:

    {

```
    "action" : "response",
    "cmdtype" : 752,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"


    }
```

3、 /upload/config
Upload profile
Return:
```
    {
    "action" : "response",
    "cmdtype" : 753,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"


    }
```

4、 /upload/apppkg
Upload package
Return:
```
    {
    "action" : "response",
    "cmdtype" : 754,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"


    }
```

5、 /upload/ascpkg
Upload ASC packets
Return:
```
    {
    "action" : "response",
    "cmdtype" : 755,
    "sequence" : 1,
    "retcode":0,
```

```
"retmsg":"success"


    }
```

6、 /upload/firebin

Upload flame BIN packets

Return:

```
    {
    "action" : "response",
    "cmdtype" : 756,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"


    }
```

7、 /upload/fireparam

Upload flame PARAM packets

Return:

```
    {
    "action" : "response",
    "cmdtype" : 757,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"
    }
```

8、 /upload/distparam

Upload ranging packets

Return:

```
    {
    "action" : "response",
    "cmdtype" : 758,
    "sequence" : 1,
    "retcode":0,
    "retmsg":"success"
    }
```

VI. Push Interface Definition

1000 Temperature alarm message

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1000 temperature alarm information
sequence	uint64	Message serial number
content	json	Alarm content
high_temp	float	High temp temperature, valid when high temp alarm
low_temp	float	Low temp temperature, valid when low temp alarm
avg_temp	float	Ave temp, valid when ave temp alarm
temp_flag	int	Alarm type, 0 means ave temp, 1 high temp alarm, 2 low temp alarm, 3 high and low temp alarm
capture_time	string	Alarm capture time, in the format 2020-05-21 12:22:33
vl_img_url	string	Alarm record visible light capture, http jpeg path
ir_ img_url	string	Alarm record infrared capture, http jpeg path
vl_video_url	string	Visible video address
ir_video_url	string	Infrared video address
vl_img_content	string	JPEG image in BASE64 format
ir_img_content	string	JPEG image in BASE64 format
type	int	1 Temperature alarm 2 Hot spot warning 3 Cold point warning
time	string	Alarm time, in the format 2020-05-21 12:22:33
config	json	Configuration
type	int	Alarm type 1 High temp alarm 2 Low temp alarm 3 Ave temp alarm 4 High and low temp alarm
condition	int	Conditions 1 Above 2 Below 3 Matching 4 Above and below
high_temp	float	Configure high temp
low_temp	float	Configure low temp
avg_temp	float	Configure ave temp
objtype	int	Type: 0 cold hot spot

			1 Point 2 Wire 3 Rectangle 4 Polygon 5 Circular
points	json array		The required point of the corresponding object. The point rule is consistent with the rule input.
x	int		x，1-512
y	int		y,1-384

Complete example:

```
{
    "action":"notify",
    "cmdtype":1000,
    "sequence":1,
    "content":
    {
                "high_temp":37.5,
                "low_temp":37.5,
                "avg_temp":25.2
                "temp_flag":1,
                "handle_flag":1,
                "alarm_addr":"园区食堂大门",
                "alarm_addr":"Park Canteen Gate",
                "vl_image_url ":"fsdfljslsdjfkslfjsdklsdjlk",
                "ir_image_url ":"fsdfljslsdjfkslfjsdklsdjlk",
                "ir_image_content":"fsdfljslsdjfkslfjsdklsdjlk",
                "vl_image_content":"fsdfljslsdjfkslfjsdklsdjlk",
                "type":1，
                "name":"k"，
                "time":"2020-05-21 12:22:33"，
                "vl_video_url":" /jji/aaa.h264",
                "ir_video_url":" /jji/aaa.h264",
                "config":{
                    "type":1
                    "condition":1,
                    "high_temp":1,
                    "low_temp":1,
                    "avg_temp":1,
                    "objtype":1
                    "points":[
```

```
                    {"x":1,
                    "y":1
            }
            ]
        }
    }


}
```

1001 Forced logout

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1001 forced logout
sequence	uint64	Message serial number
content	json	
msg	string	Logout reason description

Complete example:

```
{

    "action":"notify",
    "cmdtype":1001,
    "sequence":1,
    "content ":{
        "msg":"other logining,force out"
        }
}
```

1002 memory full alarm

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1002 memory full
sequence	uint64	Message serial number
content	json	Alarm content
total	int	Total storage in M
free	int	Available sizes, in M
limit	int	Alarm threshold, available alarm when less than alarm threshold, unit M

Complete example:

```
{
    "action":"notify",
    "cmdtype":1002,
    "sequence":1,
    " content":{
        "total":1024,
        "free":1024,
        "limit":1024
    }
}
```

1003 Storage failure

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1003 storage failure
sequence	uint64	Message serial number
content	json	Alarm content

Complete example:

```
{
    "action":"notify",
    "cmdtype":1003,
    "sequence":1,
    "content":{
    }
}
```

1004 streaming exception

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1004 streaming exception
sequence	uint64	Message serial number
content	json	Alarm content
type	int	Type: 1 streaming opening failure or verification failure 2 Abnormal collection of streaming data source 3 Exception of excessive streaming cache frames, i.e., poor network or abnormal client

Complete example:

```
{
    "action":"notify",
    "cmdtype":1004,
    "sequence":1,
    "content":{
        "type":1
    }
}
```

1005 illegal access

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1005 illegal access
sequence	uint64	Message serial number
content	json	Alarm content
user	string	Abnormal login user
ip	string	Abnormal login IP
time	string	Abnormal login time

Complete example:

```
{
    "action":"notify",
    "cmdtype":1005,
    "sequence":1,
    "content":{
        "user":"root",
        "ip":"192.168.2.12",
        "time":"2020-11-30 14:45:00"
    }
}
```

1006 recording status information

Parameter name	Type	Description
action	string	Message type: notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1006 recording status
sequence	uint64	Message serial number

status	int	Current recording status 0 No recording 1 Visible recording 2 Infrared recording
time	int	Start recording to the current number of seconds
allrecord	int	0 double-light-source recording is stopped 1 double-light-source recording is in progress
alltime	int	Double-light-source recording to current seconds
delay_status	int	Delayed recording status 0 Delayed stop recording 1 Visible light delay recording 2 Infrared delay recording 3 Visible light preparation recording 4 Infrared ready to record
delay_time	int	Valid when the delay_video_status is 1 or 2, starting delay to the current number of seconds When is 3, it is the countdown seconds
capture_status	int	Delayed capture status 0 Stop delay capture 1 Visible light delay capture 2 Infrared delay capture
capture_count	int	Current number of delayed captures
capture_total	int	Total number of delayed captures

Complete example:

```
{
    "action":"notify",
    "cmdtype":1006,
    "sequence":1,
        "content":{
            "status":1,
            "allrecord":0,
            "time":0,
            "alltime":0,
            "delay_status":0,
            "delay_time":0,
            "capture_status":0,
            "capture_count":0,
            "capture_total":0
        }
```

}

1007 object information

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1007 image information
sequence	uint64	Message serial number
content	json object	Message body node, responding to message node definition
obj_list	json array	Object list
measure_type	uint64	Object type 1 Point 2 Wire 3 Rectangle 4 Polygon
measure_id	id	Object id
measurer_name	string	Object name
max_temp	float	Max temp
min_temp	float	Min temp
avg_temp	float	Ave temp
interval	int	Current interval, in s
preset	int	Preset, subject to actual equipment support, is the object attribute Range [1,256]

Complete example:

```
{
    "action":"notify",
    "cmdtype":1007,
    "sequence":1,
    "content":{
        "interval":60,
        "obj_list":[
            {
                "measure_type":2,
                "measure_id":213,
                "measurer_name":"point0",
                "max_temp":12,
                "min_temp":21,
                "avg_temp":21,
```

```
        }
    ]


}
}
```

1008 network exception

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1008 network exception information
sequence	uint64	Message serial number
content	json object	Message body node, responding to message node definition
type	int	Type: 1 ip conflict 2 ping is not connected to the gateway, ip is the gateway
ip	string	ip address

Complete example:

```
{"action":"notify",
    "cmdtype":1009,
    "sequence":1,
    "content":{
        "type":1,
        "ip":""
    }
}
```

1009 temperature alarm message

Parameter name	Type	Options	Description
action	string	Required	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	Required	1009 temperature alarm
sequence	uint64	Required	Message serial number
from	string	Required	Message sender, applicable to clientid, deviceid, serviceid, etc.
to	string	Required	Message receiver, applicable to clientid, deviceid, serviceid, etc.
token	string	Required	User token

message	json	Required	Alarm content
vl_img_url	string	Required	Alarm record visible light capture, http jpeg path, default to empty
ir_ img_url	string	Required	Alarm record infrared capture, http jpeg path, default to empty
vl_video_url	string	Required	Visible light video address, empty by default
ir_video_url	string	Required	Infrared video address, empty by default
vl_img_content	string	Required	JPEG picture in BASE64 format, default to empty
ir_img_content	string	Required	JPEG picture in BASE64 format, default to empty
alarm_type	int	Required	1 Object temperature 2 Global temperature 7 Abnormal body temperature (IR236 equipment uses abnormal body temperature Type)
alarm_time	string	Required	Alarm time, in the format 2020-05-21 12:22:33
config	json	Required	Configuration
alarm_points	json array	Required	The required Point of the corresponding object. The Point rule is consistent with the rule input.
temp_flag	int	Required	Alarm type, 0 means ave temp, 1 means high temp alarm, 2 means low temp alarm, 3 means high and low temp alarm
x	int	Required	Alarm point x coordinate, value range 0-8091
y	int	Required	Alarm point y coordinate, value range 0-8091
w	int	Required	The alarm point is wide. If the alarm point is area, it is valid; If it is Point, the value is 0, and the value range is 1-8092.
h	int	Required	The alarm point is high. If the alarm Point is area, it is valid; if it is point, it is 0, and the value range is 1-8092.
temperature	float	Required	Alarm temperature, alarm point temperature

Complete example:

```
{
    "action":"notify",
    "cmdtype":1009,
    "sequence":1,
    "from":"client1",
    "to":"device1",
    "token":"",
    "message":
    {
        "handle_flag":1,
        "alarm_addr": "Park Canteen Gate",
```

"vl_image_url":"fsdfljslsdjfkslfjsdklsdjlk",

"ir_image_url":"fsdfljslsdjfkslfjsdklsdjlk",

"ir_image_content":"fsdfljslsdjfkslfjsdklsdjlk",

"vl_image_content":"fsdfljslsdjfkslfjsdklsdjlk",

"alarm_type":7,

"alarm_time":"2020-05-21 12:22:33",

"vl_video_url":" /jji/aaa.h264",

"ir_video_url":" /jji/aaa.h264",

"alarm_points":[

{

"temp_flag":1,

"x":1,

"y":1,

"w":1,

"h":1,

"temperature":37.5

},

{

"temp_flag":2,

"x":1,

"y":1,

"w":1,

"h":1,

"temperature":37.5

}

]

}

}

1010 external alarm messages

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1010 external alarm
sequence	uint64	Message serial number
content	json	Alarm content
vl_img_url	string	Alarm record visible light capture, http jpeg path

ir_ img_url	string	Alarm record infrared capture, http jpeg path
vl_video_url	string	Visible video address
ir_video_url	string	Infrared video address
vl_img_content	string	JPEG image in BASE64 format
ir_img_content	string	JPEG image in BASE64 format
time	string	Alarm time, in the format 2020-05-21 12:22:33

Complete example:

```
{
    "action":"notify",
    "cmdtype":1009,
    "sequence":1,
    "content":
    {

                "vl_image_url ":"fsdfljslsdjfkslfjsdklsdjlk",
                "ir_image_url ":"fsdfljslsdjfkslfjsdklsdjlk",
                "ir_image_content":"fsdfljslsdjfkslfjsdklsdjlk",
                "vl_image_content":"fsdfljslsdjfkslfjsdklsdjlk",
                "time":"2020-05-21 12:22:33"，
                "vl_video_url":" /jji/aaa.h264",
                "ir_video_url":" /jji/aaa.h264",
    }

}
```

1011 fire alarm message

Parameter name	Type	Description
action	string	notify: the notification message, the receiver receives such a message without a reply
cmdtype	int	1011 fire alarm
sequence	uint64	Message serial number
content	json	Alarm content
high_temp	float	High temp temperature, valid when high temp alarm
low_temp	float	Low temp temperature, valid when low temp alarm
avg_temp	float	Ave temp, valid when ave temp alarm
capture_time	string	Alarm capture time, in the format 2020-05-21 12:22:33
vl_img_url	string	Alarm record visible light capture, http jpeg path

ir_ img_url	string	Alarm record infrared capture, http jpeg path
vl_video_url	string	Visible video address
ir_video_url	string	Infrared video address
vl_img_content	string	JPEG image in BASE64 format
ir_img_content	string	JPEG image in BASE64 format
time	string	Alarm time, in the format 2020-05-21 12:22:33

Complete example:

```
{
    "action":"notify",
    "cmdtype":1010,
    "sequence":1,
    "content":
    {
                "high_temp":37.5,
                "low_temp":37.5,
                "avg_temp":25.2
                "temp_flag":1,
                "vl_image_url ":"fsdfljslsdjfkslfjsdklsdjlk",
                "ir_image_url ":"fsdfljslsdjfkslfjsdklsdjlk",
                "ir_image_content":"fsdfljslsdjfkslfjsdklsdjlk",
                "vl_image_content":"fsdfljslsdjfkslfjsdklsdjlk",
                "name":"k"，
                "time":"2020-05-21 12:22:33"，
                "vl_video_url":" /jji/aaa.h264",
                "ir_video_url":" /jji/aaa.h264"
    }

}
```

Appendix I: Return Code List

Return Code	Description Information
0	Success
10001	Message content is empty
10002	Invalid message content
10003	Message field is empty
10004	Invalid username
10005	Not authenticated
10006	Password change failed
10007	User does not have permission to operate
10008	User action failed
10009	Wrong password
10010	User lock
10011	Wrong user or password
10012	Same version upgrade
10013	Lower version upgrade
10014	Illegal IP
10015	IP conflict
10016	Illegal subnet mask
10017	Illegal network management
10018	Maximum number of people online exceeded

Appendix II: Video Access Specification

The authentication information needs to be transmitted after the video is connected, and the stream will be pushed only after the correct authentication information is received. The authentication needs to be completed within 1s, and the video transmission is h264 bare code stream.

Field	Bytes	Description
header	4	Fixed head verification 0x71 0x85 0x73 0x68
type	1	1 Currently only verification
streamtype	1	1 Infrared data stream 2 Visible light data stream
subtype	1	1 Main code stream 2 Secondary code stream
token	32	32 byte token string 32-bit token string returned by login

Appendix III: Ciphertext Format

Keep clear text and use clear text as needed.

Password encryption process: plaintext-public key rsa encryption-base64 encryption

Public key:

-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCsn//YAXtvUmzfcdUSVc80NgMM
NFIc/EyzOnLKcUM6Xm+up8K7AymL6TpOpgdtxDB30GlQjK7RNwJLgNSzT7d7OXJq
382cX0V6aYXA9oeZ93bsdpiDNOMNu1ezlWZNJBS2sJoUnQI7mTGJN2b44wUcqh98

F3wPTUp/+rydh3oBkQIDAQAB

\n-----END PUBLIC KEY-----

Appendix IV: Sample Websocket Login

B->S connect：ws://192.168.1.168:9980/api/v1/control

B->S

```
{
    "action": "request",
    "cmdtype": 500,
    "message": { },
    "from": "admin",
    "to": "box",
    "sequence": 0,
    "token": ""
}
```

S->B

```
{
    "action": "response",
    "cmdtype": 500,
    "message": {
        "audiofile": [
            {
                "index": 0,
                "name": "DUDI",
                "path": "/sound/0.wav",
                "path1": "/sound/0.mp3"
            },
            {
                "index": 1,
                "name": "Classic",
                "path": "/sound/1.wav",
                "path1": "/sound/1.mp3"
            },
            {
                "index": 2,
```

                    "name": "Beeps",

                    "path": "/sound/2.wav",

                    "path1": "/sound/2.mp3"

                }

        ],

        "control_port": 9980,

        "control_uri": "/api/v1/control",

        "copyright": "COPYRIGHT © 2017-2020 WWW.GUIDEIR.COM ALL RIGHTS RESERVED
Guide Sensmart Tech Co., Ltd.",

        "datetime": "2022-03-08 14:33:40",

        "gears": [

            {

                "max": 150,

                "min": -20

            },

            {

                "max": 550,

                "min": 100

            }

        ],

        "ir_mode": 0,

        "ir_model_h": 192,

        "ir_model_w": 256,

        "ir_output_h": 384,

        "ir_output_w": 512,

        "ir_rtsp_url": "rtsp://192.168.1.168:554/video1/stream0",

        "ir_sub_rtsp_url": "rtsp://192.168.1.168:554/video1/stream1",

        "ir_video_port": 9982,

        "language": "chinese",

        "logo": "",

        "rule_flush_intveral": 1,

        "systemname": "Guide Front-end Configuration Software",

        "vl_rtsp_url": "rtsp://192.168.1.168:554/video2/stream0",

        "vl_sub_rtsp_url": "rtsp://192.168.1.168:554/video2/stream1",

        "vl_video_port": 9981

    },

    "retcode": 0,

    "retmsg": "success.",

    "sequence": 0

}

B->S

```
{
    "action": "request",
    "cmdtype": 501,
    "message": {
        "username": "admin",
        "password":
"S8futiRVYyinblVFFMemGLcgFAEir93HWSLhCerQ1BnacBcZttM1yy9DynNw/6uG1Zk49CMki+9+fFOiY
ONYZS+GvdrhADA9Q2ji1lt1HTuouRd2jp2b5nkp/4P+rmVxr+RqLJUiEfbkf6Fzmcd+Ao8YReV5NPV5J0
Ec7zdT0S4="
    },
    "from": "admin",
    "to": "box",
    "sequence": 1,
    "token": ""
}
```

S->B

```
{
    "action": "response",
    "cmdtype": 501,
    "message": {
        "token": "7d358ef09eaa11ecaf4d00151810ca9d"
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 1
}
```

B->S

```
{
    "token": "7d358ef09eaa11ecaf4d00151810ca9d",
    "action": "notify",
    "cmdtype": 9999,
    "sequence": 1,
    "message": { }
}
```

B-S

```
{
    "action": "request",
    "cmdtype": 16,
    "message": {
        "default": 0
```

```json
    },
    "from": "admin",
    "to": "box",
    "sequence": 2,
    "token": "7d358ef09eaa11ecaf4d00151810ca9d"
}
```

S->B

```json
{
    "action": "response",
    "cmdtype": 16,
    "message": {
        "capture_count": 3,
        "capture_interval": 5,
        "copyright": "COPYRIGHT © 2017-2020 WWW.GUIDEIR.COM ALL RIGHTS RESERVED
Guide Sensmart Tech Co., Ltd.",
        "dist_unit": 0,
        "home_logo": "",
        "language": 0,
        "logo": "",
        "name": "高德前端配置软件",
        "name": "Guide Front-end Configuration Software",
        "porting_name": "NC200",
        "record_interval": 5,
        "record_max_size": 999,
        "record_time": 300,
        "temperature_unit": 0,
        "time_format": 0,
        "time_zone": 0
    },
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 2
}
```

B->S

```json
{
    "action": "request",
    "cmdtype": 502,
    "message": {
        "username": "admin"
    },
```

```
    "sequence": 11,
    "token": "7d358ef09eaa11ecaf4d00151810ca9d"
}
```
S->B
```
{
    "action": "response",
    "cmdtype": 502,
    "message": null,
    "retcode": 0,
    "retmsg": "success.",
    "sequence": 11
}
```

Appendix V: HTTP Request Sample

B- >S (sample universal interface 500)

POST /getmsginfo HTTP/1.1

Referer: http://172.16.103.159/Default.aspx

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Language: en-US,en;q=0.8,zh-CN;q=0.6,zh;q=0.4,ru;q=0.2

Accept-Charset: GBK,utf-8;q=0.7,*;q=0.3

User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/38.0.2125.122 Safari/537.36

Content-Type: application/x-www-form-urlencoded

Host: 172.16.103.159

Content-Length: 95

Expect: 100-continue

```
{
"action":"request",
"cmdtype":500,
"sequence":1,
"message":{
}
}
```

S->B (sample universal interface 500)

HTTP/1.1 200 OK

Server: nginx/1.18.0

Date: Mon, 18 Apr 2022 06:36:15 GMT

{"action":"response","cmdtype":500,"message":{"audiofile":[{"index":0,"name":"DUDI","path":"/sound/0.wav","path1":"/sound/0.mp3"},{"index":1,"name":"Classic","path":"/sound/1.wav","path1":"/sound/1.mp3"},{"index":2,"name":"Beeps","path":"/sound/2.wav","path1":"/sound/2.mp3"}],"control_port":9980,"control_uri":"/api/v1/control","copyright":"COPYRIGHT \u00a9 2017-2020 WWW.GUIDEIR.COM ALL RIGHTS RESERVED Guide Sensmart Tech Co., Ltd.","datetime":"2022-04-18 14:36:15","gears":[{"max":150,"min":-20},{"max":550,"min":100}],"ir_mode":0,"ir_model_h":192,"ir_model_w":256,"ir_output_h":384,"ir_output_w":512,"ir_rtsp_url":"rtsp://172.16.103.159:1024/video1/stream0","ir_sub_rtsp_url":"rtsp://172.16.103.159:1024/video1/stream1","ir_video_port":9982,"language":"chinese","logo":"","rule_flush_intveral":1,"systemname":"........................","vl_rtsp_url":"rtsp://172.16.103.159:1024/video2/stream0","vl_sub_rtsp_url":"rtsp://172.16.103.159:1024/video2/stream1","vl_video_port":9981},"retcode":0,"retmsg":"success.","sequence":1}

B- >S (login sample)

POST /getmsginfo HTTP/1.1

Referer: http://172.16.103.159/Default.aspx

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Language: en-US,en;q=0.8,zh-CN;q=0.6,zh;q=0.4,ru;q=0.2

Accept-Charset: GBK,utf-8;q=0.7,*;q=0.3

User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/38.0.2125.122 Safari/537.36

Content-Type: application/x-www-form-urlencoded

Host: 172.16.103.159

Content-Length: 299

Expect: 100-continue

{"action":"request","cmdtype":501,"message":{"username":"root","password":"NrIHEqC5HzF6Xw8ShD6ig1Fx+Vl6sfk7h63dbqkK26uTjH5ucXuoEW0Z3vg/LgsGN5BVx5U5RR3wRHIjLhR8+I0LrksJD03ma1N+j996a3bXA18r1Bz162na9aon7af34i476HeMqcXYs5S9rAc1L6unGD4mrMGnFVAYLTc7zYM="},"from":"root","to":"box","sequence":1,"token":""}

S->B (login sample)

{"action":"response","cmdtype":501,"message":{"token":"7279a070bee211ec80bd00151810ca9d"},"retcode":0,"retmsg":"success.","sequence":1}

B- >S (get sample network information)

POST /getmsginfo HTTP/1.1

Referer: http://172.16.103.159/Default.aspx

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Accept-Language: en-US,en;q=0.8,zh-CN;q=0.6,zh;q=0.4,ru;q=0.2

Accept-Charset: GBK,utf-8;q=0.7,*;q=0.3

User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/38.0.2125.122 Safari/537.36

Content-Type: application/x-www-form-urlencoded

Host: 172.16.103.159

Content-Length: 115

Expect: 100-continue

{
"action":"request",
"cmdtype":10,
"sequence":1,
"token":"1f5bb274bee311ecbef900151810ca9d",
"message":{}
}

S->B (get sample network information)

HTTP/1.1 200 OK

Server: nginx/1.18.0

Date: Mon, 18 Apr 2022 06:45:53 GMT

Content-Type: text/html

Transfer-Encoding: chunked

Connection: keep-alive

{"action":"response","cmdtype":10,"message":{"card":0,"dns1":"0.0.0.0","dns2":"0.0.0.0","gateway":"172.16.100.1","host_name":"MC200","ip_version":0,"ipaddr":"172.16.103.159","mac":"00:15:18:10:CA:9D","mode":0,"netmask":"255.255.252.0"},"retcode":0,"retmsg":"success.","sequence":1}