

HTTP protocol document

Table of Contents

- 1, the HTTP protocol flow process description
- 2, HTTP upload parameters Number of Configurations
- 3, the information category do
 - 3.1 Upload Information
 - 3.2 Receive Message
- 4, the upload channel message format
 - * Device type
 - 4.1. Registration Information
 - Request parameter
 - Return parameter
 - JSON format example
 - 4.2. Heartbeat information
 - Request parameter 16
 - Return parameter 17
 - JSON format example 17
 - 4.3. Snapshot information 18
 - Request parameter 18
 - Return parameter 25
 - JSON format example 26
 - * Code list (provided apparatus upload channel information returned b a c k Code)
 - V 1.0.0 version
 - 1, face recognition channel information JSON format
 - 2, the face alignment channel information JSON format
 - 3. Heartbeat information JSON format
- 5, receiving channel information format
 - * Sign verification
 - * UUID 34
 - 5.1. Manage the list database
 - (1) get the name of a single list of
 - (2) acquiring person staff information
 - (3) Added by staff information
 - (4) editors staff information
 - (5) delete the name of a single list of
 - (6) Delete person staff information
 - (7) Bulk added plus art channel information(characteristic value)
 - (8) batch delete In addition to personnel information
 - Picture data base64 example
 - V 1.0.0 version
 - 2, the device acquires channel information
 - Request parameter
 - Return parameter
 - JSON format example
 - V 1.0.0 version
 - 5.3, the IO control
 - Request parameter

Return parameter
JSON format example
V 1.0.0 version
5.4 , upload parameters eligible to take
Request parameter
Return parameter
JSON format example
5.5 , upload parameter setting set
Request parameter
Return parameter
JSON format example
5.6. Time synchronization
Request parameter
Return parameter
JSON format example
5.7 , get face special eigenvalues
Request parameter
Return parameter
JSON format example
8 , face search name a single library
Request parameter
Return parameter
JSON format example
5.9, face recognition reference number to obtain
Request parameter
Return parameter
JSON format example
5.10, face recognition parameter number setting
Request parameter
Return parameter
JSON format example
5.11, face recognition Daily alarm parameters to obtain
Request parameter
Return parameter
JSON format example
5.12, face recognition reported alarm parameters
Request parameter
Return parameter
JSON format example
5.13.Network parameter acquisition
Request parameter
Return parameter
JSON format example
5.14.Network parameter setting
Request parameter
Return parameter
JSON format example
5.15 , reboot

Request parameter

Return parameter

JSON format example

5.16. Other errors return

JSON format example

V 1.0.0 version

* Code list(provided apparatus receiving channel information returned back Code)

* The Result wrong error

6. common asked questions

1. Failed to add picture

2. Parsing picture failed

3. Interface access failed

1. HTTP protocol flow description

V1.1.0 HTTP protocol adds device registration management. Equipment to be sent to the server to send the registration information, and records the server returns the current device registered Session value (Generated by the server) used to manage the device, the device will push information to the server only after successful registration The server needs to bring Session parameters when accessing the device interface, otherwise the device will reject the request.(No registration management functions in the reference Close Sign feature set number)

2. HTTP upload parameter configuration

The upload parameter configuration of the camera is as follows:

The I E Login camera: "Settings" -> "Network Settings" -> " H T T P Upload Settings" -> Set H T T P upload information.

Address format : [http://\[ip\]:\[port\]/\[path\]](http://[ip]:[port]/[path]) ([ip] : domain name or IP address)

Example : (fill address below **is only an example** , the actual address of the server application should fill actually received) <http://www.httpserver.com/captureinfo> to receive a capture address information; <http://www.httpserver.com/keepalive> of Address for receiving heartbeat information; <http://www.httpserver.com/register> is the address for receiving registration information.

Options	content	Explanation
Snapshot upload	Checked / unchecked	Whether to upload snapshot information
Upload address for snapshot information	http://www.httpserver.com/captureinfo	Address of server receiving snapshot information
Compare upload types	Everyone upload / Compare successfully upload / Blacklist upload / Whitelist upload / VIP list upload / Stranger Upload / non-whitelist upload (single choice)	The comparison machine / NVR pushes the corresponding type of snapshot information according to this option, supporting comparison Options specific to functional devices.
Snap content	FaceInfo / CompareInfo (multiple choices)	Alternatively "capture information" upload the content (the JS O N objects) the option to upload "the CompareInfo" required information device support Comparison function

upload picture	Face image / background image / list library photo / human figure (Multiple choice)	Whether to bring the corresponding information when pushing the snapshot information Picture data
Number of retransmissions	0 ~ n	Snapshot when the server fails to receive and return Information re-upload
registered	Checked / unchecked	Whether to open the registration function

		(After opening, the server needs to be connected to the registration phase Close the interface, please close the old version docking)
Registration information upload address	http://www.httpserver.com/register	Address of server receiving registration information
Heartbeat upload	Checked / unchecked	Whether to upload heartbeat information
Heartbeat information upload address	http://www.httpserver.com/keepalive	Address of the server receiving the heartbeat information
Heartbeat interval	Unit : seconds ; value greater than 0, default 60 seconds	Heartbeat interval
Sign verification	On / off	Whether to grant permission when accessing the http interface Verification (Sign value verification)
HTTP version	Current HTTP protocol version : V 1.0.0 /V1.1.0/V1.1.1/V1.1.2 / ... (current Latest HTTP version)	Push according to the format of the current HTTP protocol

		version, please select the old version to connect 1.0.0 version
--	--	---

After filling in the upload parameters, click Save to take effect.

3. Information category

Upload information

Registration information: HTTP request header (POST) + registration information (JSON)
 Heartbeat information: HTTP request header (POST) + heartbeat information (JSON)
 Snapshot information: HTTP request header (POST) + snapshot information (JSON)

Receive information

Equipment listening **8011** port (individual device version of this port can be modified) , provides http interface to manage the device. **LAN access**

address :

V1.0.0 version: http: // [device IP]: 8011 /

Example: http://192.168.1.88:8011/ (where 192.168.1.88 is the IP address of the device)

V1.1.x version: http: // [Device IP]: 8011 / **Request**

Example: http://192.168.1.88:8011/Request (where 192.168.1.88 is the IP address of the device)

WAN access address : (need to use the P2P service to access, the specific use is subject to the P2P service documentation.) Version 1.0.0 http: // [P2P program running server IP]: 8011

Example: http://127.0.0.1:8011/ (where 127.0.0.1 is the server address of the machine running P2P service)

V1.1.x version: http: // [P2P program running server IP]: 8011 / **Request**

Example: http://127.0.0.1:8011/Request

http request access format :

HTTP request header (POST) + (JSON) **Does not support gzip and other compression formats**

4. Upload information format

*** Device type**

1 : capture machine 2 : alignment of the machine . 3 : the NVR . 4: matching the service unit

1. Registration information Request parameters

ChannelNum	Number of channels	Y		Integer	
WebVersion	Page version	Y	<128	String	
CoreVersion	Main program version	Y	<128	String	
VersionDate	Version date. format: Nov 14 2018 18:02:07	Y	<64	String	
HTTPVersion	HTTP protocol version	Y		String	
HTTPDate	HTTP protocol version date	Y		String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "registerResponse"	Y	Fixed 16	String	
TimeStamp	Unix timestamp (seconds)	Y	Fixed 10	Integer	
Data	Back to content	Y		Json to Elephant	
Session	Generation rule: DeviceUUID _TimeStamp	Y		String	

ServerVersion	Server (HTTP protocol) version	Y		String	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "registerRequest",
  "Timestamp":
  1555558678, "
  Data":
  {
    "DeviceInfo":
    {
      "DeviceId ":
      "123456", "DeviceUUID":
      "abcdefghijklmef",
      "DeviceMac":
      "aa: bb: cc: dd: ee: ff",
      "DeviceIP": "192.168.1.88",
      "DeviceType": 1,
      "ChannelNum": 1,
      "WebVersion":
      "Face_9.20_FH4",
      "CoreVersion":
      "15.13.1.13",
      "VersionDate": "Nov
      14 2018 18:02:07"
    }
    "HTTPVersion": " 1.1.0",
    "HTTPDate": "2019-04-22"
  }
}
```

return:

```

{
  "Name":
  "register
  Respons
  e",
  "TimeSta
  mp":
  1555559
  039, "Dat
  a":
  {
    "Session": " abcdefabcdef _ 1555559039 ",
    "ServerVersion": " 1.1.0"
  },
  "Code": 1,
  "Message": ""
}
    
```

2. Heartbeat information Request parameters

	Nov 14 2018 18:02:07				
HTTPVersion	HTTP protocol version	Y		String	
HTTPDate	HTTP protocol version date	Y		String	
HeartbeatCount	Heartbeat message sending count (device Start counting at startup)	Y		Integer	
CaptureCount	Snapshot information sending count (device Start counting at startup)			Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	fixed "heartbeatResponse"	Y	Fixed 17	String	
TimeStamp	Unix timestamp (seconds)	Y	Fixed 10	Integer	
Session	Register to return to Session	Y		String	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name": "heartbeatRequest",

  "TimeStamp": 1555599123,
  "Session": "abcdefabcdef _ 155559039",
  "Data":
  {
    "DeviceInfo":
    {
      " DeviceId ":
      "123456", "DeviceUUID":
      "abcdefabcdef",
      "DeviceMac":
      "aa: bb: cc: dd: ee: ff",
      "DeviceIP": "192.168.1.88",
      "ChannelNo": 0,
      "WebVersion":
      "Face_9.20_FH4",
      "CoreVersion":
      "15.13.1.13",
      "VersionDate": "Nov
      14 2018 18:02:07"
    }
    "HTTPVersion": " 1.1.0",
    "HTTPDate": "2019-04-22",
    "HeartbeatCount": 10000,
    "CaptureCount": 1000
  }
}
```

```

    }
}

return:
{
    "Name":
    "heartbeat
    Response",
    "TimeStam
    p":
    155559912
    5,
    "Session": "abcdefabcdef _ 155559039",
    "Code": 1,
    "Message": ""
}

```

3. Snap information Request parameters

Name	Solid fixed "captureInfoRequest"	Y	Fixed 18	String	
TimeStamp	Unix timestamp (seconds)	Y	Fixed 10	Integer	
Session	Register to return to Session	Y		String	
Data	Request content	Y		Json object	
DeviceInfo	Device Information	Y		Json object	
DeviceId	device serial number	Y	<64	String	
DeviceUUID	Device UUID	Y	Fixed 12	String	
DeviceMac	Device Mac address Format aa: bb: cc: dd: ee: ff	Y		String	

DeviceIP	Device IP address	Y	<64	String	
ChannelNo	Channel number (of each channel of NVR IPC , value 255 is NVR)	Y		Integer	
SubDeviceInfo	Information about the sub-device connected to the device (Example: IPC of a channel of NVR)	N		JSON pair Elephant	
SubDeviceIP	Sub-device IP address	N	<64	String	
SubDeviceUUID	Sub-device UUID	N	Fixed 12	String	
CaptureInfo	Snap information	Y		JSON pair Elephant	
SendTime	Sending time Format :	Y		String	

	yyyy-mm-dd hh: mm: ss				
CaptureTime	Capture time format : yyyy-mm-dd hh: mm: ss	Y		String	
CaptureCount	Snapshot picture count (device startup Start counting)	Y		Integer	
FacePicture	Face image data (base64 edit Code format)	N		String	

BodyPicture	Humanoid picture data (base64 edit Code format)	N		String	
BackgroundPicture	Background image data (base64 edit Code format)	N		String	
FaceInfo	Face information	N		JSON object	
FaceId	Face ID	N		Integer	
FaceQuality	Face quality score	N	0 ~ 9999	Integer	
FacePosition	Face coordinates Format : "x1, y1, x2, y2" . Description: The background image size as the reference pictures. x1 is the distance from the left border of the face frame to the left border of the background image, x2 is the distance from the right border of the face frame to the back	N		String	As shown:

	<p>Scenes from the left border, y1 is the distance on the face frame on the background Border distance, y2 human face border from the background at the top box The distance of the box, in pixels.</p>				
CompareInfo	Compare information	N		JSON object	<p>Information only available for devices that support the comparison function Content</p>
AlarmEvent	<p>Alarm event</p> <p>0 : no alarm</p> <p>1 : Blacklist alarm</p> <p>2 : Whitelist alarm</p> <p>3 : VIP list alarm</p> <p>4 : Non-whitelist alarm</p>	N		Integer	<p>List alarm events set by the current device</p>

Liveness	Face live detection 0 : Non-living 1 : Living body	N		Integer	Non-living refers to photos, living refers to true people
Attribute	Face attributes	N		JSON object	
Age	Age [0 ~ 100]	N		Integer	

Gender	gender 1 : Male 2 : Female	N		Integer	
Glasses	Wear glasses 0 : No glasses 1 : Wear glasses 2 : Wear sunglasses	N		Integer	
Mask	Mask / Mask 0 : do not wear a mask / mask 1 : Wear a mask / mask	N		Integer	
Beard	Moustache 0 : no beard 1 : bearded	N		Integer	

Race	Skin race 1 : Yellow race 2 : Black people 3 : Caucasian	N		Integer	
CompareTime	Compare time format: yyyy-mm-dd hh: mm: ss	N		String	
PersonType	List type: 0 : Stranger 1 : Blacklist	N		Integer	Strangers said the comparison was unsuccessful

	2 : Whitelist 3 : VIP list				
VisitsCount	Visits	N		Integer	Compare the information that is unique to success Content

Similarity	Similarity Format: 0.1000 (floating point)	N		String	Compare the information that is unique to success Content
PersonInfo	Personnel information	N		JSON object	Compare the information that is unique to success Content
PersonId	Person ID	N	<36	String	
PersonName	Personnel name	N	<64	String	
Sex	gender 1 : Male 2 : Female	N	1-2	Integer	
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
SaveTime	Add time	N	<20	String	

LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	
PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON object	
PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	

PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	

PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	
PersonData5	Custom information 5	N	<88	String	
PersonPhoto	Staff photos (base64 encoding)	N	<2M <960 * 960	String	
TemperalInfo	Temperature information	N		JSON object	
Temperature	body temperature	N	<24	String	
EnvirTemperature	Ambient temperature	N	<24	String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "captureInfoResponse"	Y	Fixed 19	String	

TimeStamp	Unix timestamp (seconds)	Y	Fixed 10	Integer	
Session	Register to return to Session	Y		String	
Code	Return opcode	Y		Integer	

Message	Return operation information	Y	Max 200	String	
---------	------------------------------	---	---------	--------	--

JSON format example

send:

```
{
  "Name":
  "captureInfoRequest",
  "TimeStamp":
  1555598153,
  "Session":
  "abcdefabcdef _ 15
  55559039", "Data":
  {
    "DeviceInfo":
    {
      "DeviceId":
      "123456",
      "DeviceUUID":
      "abcdefabcdef
      ", "DeviceMac":
      "00-11-04-00-
      19-e3",
      "DeviceIP": "192.168.1.88",
      "
      C
      h
      a
      n
      n
      e
      l
      N
      o
      "
      :
      0
      ,
```

```
"
  S
  u
  b
  D
  e
  v
  i
  c
  e
  I
  n
  f
  o
  "
  :
  {
    "SubDeviceIP": "192.168.1.188",
    "SubDeviceUUID": "abcdefabcdef"
  }
},

"CaptureInfo":
{
  "SendTime": "2019-04-22 09:09:28",
  "CaptureTime ": "2019-04-22 09:09:26",
  "Captur
  eCount"
  : 1000,
  "FacePi
  ctуре": "
  ...
  base64"
  ,

  "BodyPicture
  ": "...
  base64",
  "Background
  Picture": "...
  base64"
},

"FaceInfo":
{
  "FaceId": 123456,
  "FaceQuality
  ": 5,
  "FacePositio
  n":
  "10,10,100,1
  00"
```

```
},  
  
  "TemperalInfo":  
  {  
    "Temperatu  
    re": "36.5",  
    "EnvirTemp  
    erature":  
    "24.0"  
  },  
  "CompareInfo":  
  {  
    "AlarmEvent": 0,  
    "  
    L  
    i  
    v  
    e  
    n  
    e  
    s  
    s  
    "  
    :  
    1  
    ;  
    "  
    A  
    t  
    t  
    r  
    i  
    b  
    u  
    t  
    e  
    "  
    :  
    {  
      "Age": 32,  
      "Gender": 1,  
      "Glasses": 1,  
      "Mask": 1,  
      "Race": 1,  
      "Beard": 1  
    },  
    "CompareTime ": "2019-04-22 09:09:26",  
    "PersonType": 2,  
    "VisitsCount": 10,
```

"
S
i
m
i
l
a
r
i
t
y
":
"
0
.
9
1
2
",
"
P
e
r
s
o
n
I
n
f
o
":
{

"PersonId":
"Person1",
"PersonName":
"PersonName1", "Sex":
1,
"IDCard": " ",
"Nation": " ",
"Birthday": "",
"Phone": " ",
"Address": " ",
"SaveTime": "2019-01-01 00:00:00",
"LimitTime": 1,
"StartTime": "2019-01-01 00:00:00",
"EndTime": "2020-01-01 00:00:00",
"
L
a
b
e
l


```
"
:
"
"
;
"
P
e
r
s
o
n
E
x
t
e
n
s
i
o
n
"
:
{
    "PersonCode1": "",
    "PersonCode2": "",
    "PersonCode3":
    "",
    "PersonReserveName": "",
    "PersonParam1":
    0,
    "PersonParam
```

2
"
:
0
,
"
P
e
r
s
o
n
P
a
r
a
m
3
"
:
0
,
"
P
e
r
s
o
n
P
a
r
a
m
4
"
:
0
,
"
P
e
r
s
o
n
P
a
r
a

```
    r
    a
    m
    5
    "
    :
    0
    ,
    "
    P
    e
    r
    s
    o
    n
    D
    a
    t
    a
    1
    "
    :
    "
    "
    ,
    "PersonData2": "",
    "PersonData3": "",
    "PersonData4": "",
    "PersonData5": ""
    },
    "PersonPhoto": "... (base64) ..."
  }
}
}
}
}
return:
{
  "Name":
  "captureInfoRespon
  se", "TimeStamp":
  1555598153,
  "Session": "abcdefabcdef _ 155559039",
  "Code": 1,
  "Message": ""
}
```

*** Code list (device upload information returns Code)**

Code	Explanation	Remarks
1	success	
0	failure	
1001	Device is not registered / registration failed	

V 1.0.0 version

1. Face recognition information JSON format

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "FaceInfo"	20	String	
DeviceId	device serial number	64	String	
DeviceUUID	Device UUID	32	String	
DeviceMac	Device Mac address	64	String	
FaceID	Face ID	0 ~ 9999999	Integer	
FaceQuality	Face score	0 ~ 65536	Integer	
Position	Face coordinates top bottom left right	1920 * 1080	Integer	
SnapTime	Capture time format : % Y% m% dT% H% M% S Example : 20180417T144000		String	(Old)
CaptureTime	Capture time, for example : 2018-04-17 14:53:23		String	

img	Face image data (base64 encoding)		String	
imgex	Face background image data (base64 encoding) Note : " Face Recognition . Parameter settings " in the " Picture upload format " select " Upload picture " or " upload Face and artwork " Will have the original image data	Current main stream resolution	String	

Note : There are (old) remarks in the field
"Keep compatible with the old version", please ignore
when the new version is docked. Example :

```
{
  "Name": "FaceInfo",
  "DeviceId": "",
  "DeviceUUID": "u
mksbhtsv6as",
  "DeviceMac": "00-
11-04-00-19-e3",
  "FaceID":
123456,

  "
F
a
c
e
Q
u
a
l
i
t
y
"
:
5
;
"
P
o
s
i
t
i
o
n
"
:

```

```
{
  "top": 0,
  "bottom": 0,
  "left": 0,
  "right": 0
},
"SnapTime": "20180417T144000",
"img": "",
"imgex": ""
}
```

2. Face comparison information JSON format

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "CompareInfo"	20	String	
DeviceId	device serial number	64	String	
DeviceUUID	Device UUID	32	String	
DeviceMac	Device Mac address	64	String	
WBMode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
LibIndex	Black and white list library index	0 ~ 6	Integer	
LibNumber	Black and white list library number	0 ~ 3199	Integer	
PicName	Face library picture name (.jpg)	64	String	
CurPicName	Compare picture names in real time (.jpg)	64	String	
CompareResult	Comparison results	0 ~ 1	Floating point	
SnapTime	Capture time, for example : 2018-04-17 14:53:23		String	(old)
CaptureTime	Capture time, for example : 2018-04-17 14:53:23		String	
SendTime	Sending time, for example : 2018-04-17 14:53:23		String	
FaceQuality	Face quality score	0 ~ 99999	Integer	
VisitCount	Cumulative visits	1 ~ 2147483647	Integer	

Attribute	Face recognition attributes: Age Age Gender Gender Glasses glasses Mask wearing masks Beard beard Race skin race Emotion expression	Age [0 ~ 100]; G E n- D E R & It [1 : M 2 : M]; Glasses [0 : do not wear glasses 1: glasses 2 : be wearing Mirror] Mask / Beard [0: No 1: yes]; Race [1: Yellow Race 2: Black 3: Caucasian]; Emotion [1: Angry 2: Calm 3: Happy 4: S ad 5: Surprised];	Intege r	
img	Face image data (base64 encoding)		String	
imgex	Face background image data (base64 encoding)	Current main stream resolution	String	

	Note : " Face Recognition . Parameter settings " in the " Picture upload format " select " Upload picture " or " upload Face and artwork " will have original data			
FaceFeature	Face 128 - dimensional feature value (data format: feature1, feature 2, ..., feature127 , feature128) featureX is float Value to string)		String	

Note : There are (old) remarks in the field
"Keep compatible with the old version", please ignore
when the new version is docked. Example :

```
{
  "Name":
  "CompareInfo",
  "DeviceId": "1000
6612",

  "DeviceUUID": "u
mksbhtsv6as",
```

"DeviceMac": "00-
11-04-00-19-
e3", "WBMode": 2,

"LibIndex": 0,
"LibNumber": 3,
"PicName": " test _01.jpg",
"CurPicName": "FACE_10006612_20181114T220926120_2_Age32_GG.
jpg",
"CompareResult": "0.975545",
"CaptureTime ": "2018-11-14 22:09:26",
"SendTime": "2018-11-14 22:10:49 ",
"FaceQuality": 8863,

"Vi
sit
Co
unt
": 4
812
6,"
Attr
ibut
e":

```
{  
  "Age": 32,  
  "Gender": 1,  
  "Glasses": 1,  
  "Mask": 1,  
  "Race": 1,  
  "  
  B  
  e  
  a  
  r  
  d  
  "  
  :  
  1  
  ;  
  "  
  E  
  m  
  o  
  t  
  i  
  o  
  n  
  "
```



```

        :
        1
    },
    "img": "",
    "imgex": "",
    "FaceFeature": ""
}
    
```

3. Heartbeat information JSON format

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "KeepAlive"	20	String	
DeviceId	device serial number	64	String	
DeviceUUID	Device UUID	32	String	
TimeStamp	Unix timestamp		Integer	
KeepAliveCount	Heartbeat count		Integer	
FaceMsgCount	Push count: Number of pushes for face recognition / comparison results		Integer	
WebVersion	Page version	128	String	
CoreVersion	Main program version	128	String	

Example :

```

{
    "Name": "KeepAlive",
    "DeviceId": "",
    "DeviceUUID"
    : "umksbhtsv6
    as", "TimeSta
    mp": 1542180
    379,

    "KeepAliveCount": 1,
    "FaceMsgCount": 1,
    "WebVersion":
    "Face_9.20_FH4",
}
    
```

```
"CoreVersion":  
  "15.13.1.13",  
}
```

V. Format of receiving information

* Sign verification

User verification information **Sign** generation rules :

UUID: UserName: PassWord: TimeStamp and md5 encryption after splicing (lowercase letters)

UUID device UUID

UserName device user name

PassWord device user password

T i m e S t a m p U n i x timestamp (unit: seconds)(same as T i m e S t a m p value in j s o n)

: Splice connection symbol

Example :

UUID:

abcd

efabc

def

User

Nam

e:

admi

n

Pass

Word:

admi

n.123

Time

Stam

p:

1555

5586

78

After splicing: abcdefabcdef:

admin: admin.123: 1555558678

md5 encryption: c97a112d9cdc604

2fc89051aaf0f8748

Sign : c97a112d9cdc6042fc89051aaf0f8748 (**lowercase letters**)

* UUID

The UUID parameters required for WAN delivery .
 View the camera UU ID : IE Log in to the camera: "Settings" -> "Network Settings" -> "Mobile Monitoring" -> " UU I D ."

1. Manage the list database

(1) Get roster list

Request parameters

name	definition	essen tial	Length / R ange	Types of	Remarks
Name	Fixed "personListRequ est"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
Channe lNo	Channel number (manage NVR channels Of the IPC , the value 25 . 5 for managing N V R & It)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	
TimeSt amp	Unix timestamp (secon ds)	N	Fixed 10	Integer	Open Sign verif ication Required for certification

Sign	The authority verification information is verified when the server parameter setting " Management Authority Verification " is enabled.	N	Fixed 32	String	
Data	Request content	Y		Json to Elephant	
Action	Fixed "getPersonList"	Y		String	

PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PageNo	List page number (default 1)	N		Integer	
PageSize	List single page maximum number	N		Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed " personListResponse "	Y	Fixed 18	String	

Session	Register to return to Session	N		String	
Data	Back to content	Y		Json to Elephant	
Action	Fixed "getPersonList"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PageNo	List page number (default 1)	Y		Integer	

PageSize	List single page maximum number (Not the current number of returnees)	Y		Integer	
PageCount	Total number of pages	Y		Integer	
PersonCount	Total number of people on this list (Not the current number of returned lists)	Y		Integer	
PersonListNum	People currently returning to the roster Quantity	Y		Integer	

PersonList	List obtained	Y		JSON number group	
PersonId	Person ID	Y		String	
PersonName	Personnel name	N		String	
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personListRequest", "UUID":
  "abcdefabcdef",
  "Session": " abcdefabcdef _ 155559039 ",

  "TimeStamp": 155558678,
  "Sign":
  " c97a112d9cdc6042fc89051aaf
  0f8748", "Data":
  {
    "Action":
    " getPersonList ",
    " Person
    Type ": 1,
    "PageNo":
    1,
    "PageSize
    ": 1000
  }
}
```

```
}  
  
return:  
{  
  "Name":  
  "personListResponse"  
  , "Session":  
  " abcdefabcdef _ 1555  
559039 ", "Data":  
  {  
    "Action":  
    " getPers  
onList ",  
    " Person  
Type ": 1,  
    "PageNo":  
    1,  
    "Pag  
eSize  
":  
    1000,  
    "Pag  
eCou  
nt":  
    10,  
    " Per  
son  
Cou  
nt":  
    100  
00,  
    "PersonListNum": 1000,  
    "  
  
P  
e  
r  
s  
o  
n  
L  
i  
s  
t  
  
"  
:  
  [  

```

```

        {
            },
        {
            },
            ...
        ]
    },

```

```

" PersonId": "PersonId1",
" PersonName": " PersonName1"

```

```

" PersonId": "PersonId2",
" PersonName": " PersonName2"

```

```

    "Code": 1,
    "Message": ""
}

```

(2) Get personnel information

Request parameters

name	definition	essen tial	Length / R ange	Types of	Remarks
Name	Fixed "personListRequ est"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number

Channel No	Channel number (manage NVR channels Of the IPC , the value 25 . 5 for managing N V R & It)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	Y	Fixed 10	Integer	Open Sign verification Required for certification
Sign	Permission verification information, when the server Verification when the parameter setting " Management Authority Verification " is enabled	N	Fixed 32	String	
Data	Request content	Y		Json to Elephant	
Action	Fixed "getPerson"	Y		String	
PersonType	List type: 1 : Blacklist	Y		Integer	

	2 : Whitelist 3 : VIP list				
PersonId	Person ID	Y	<36	String	

GetPhoto	Whether to get image data 0 : do not return pictures 1 : Return to the picture	N	0-1	Integer	
----------	--	---	-----	---------	--

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListResponse"	Y	Fixed 18	String	
Session	Register to return to Session	N		String	
Data	Back to content	Y		JSON pair Elephant	
Action	Fixed "getPerson"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PersonInfo	Personnel information obtained	Y		Json to Elephant	

PersonId	Person ID	Y	<36	String	
----------	-----------	---	-----	--------	--

PersonName	Personnel name	N	<64	String	
Sex	gender 1 : Male 2 : Female 0 : Unknown	N	0-2	Integer	
Sex	gender 1 : Male 2 : Female	N	1-2	Integer	
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
SaveTime	Add time	N	<20	String	
LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	

Label	Personnel tags	N	<180	String	
PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON pair Elephant	
PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	
PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	
PersonData5	Custom information 5	N	<88	String	
PersonPhoto	Staff photos (base64 encoding)	N	<2M <960 * 960	String	
Code	Return opcode	Y		Integer	

	1 success				
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personList
  Request",
  "UUID":
  "abcdefabc
  def",
  "Session":
  " abcdefabcdef _ 1555
  559039 ", "TimeStam
  p": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    "Action
    ":
    " getP
    erson
    ",
    " Pers
    onTyp
    e ": 1,
    "Perso
    nId":
    "Perso
    nId1",
    " GetP
    hoto":
    1
  }
}
```

return:

```
{
  "Name":
  "personListResponse"
```

```
, "Session":  
" abcdefabcdef _ 1555  
559039 ", "Data":  
{  
  "Action":  
  " get  
  Person  
  on ",  
  " Person  
  son  
  Type  
  ": 1,  
  " PersonInfo ":  
  {  
    " PersonId":  
    "PersonId1",  
    "PersonName":  
    "PersonName1",  
    " Sex ": 1,  
    " IDCard ": " ",  
    " Nation ": " ",  
    " Birthday": "",  
    "Phone": " ",  
    "Address": " ",  
    "SaveTime": "2019-01-01 00:00:00",  
  
    "LimitTime": 1,  
    "StartTime": "2019-01-01 00:00:00",  
    "EndTime": "2020-01-01 00:00:00",  
    "  
    L  
    a  
    b  
    e  
    l  
    "  
    :  
    "  
    "  
    ,  
    "  
    P  
    e  
    r  
    s  
    o  
    n  
    E  
    x  
    t
```

```

    e
    n
    s
    i
    o
    n
    "
    :
    {
        "PersonCode1": "",
        "PersonCode2": "",
        "Person
        Code3":
        "", "Per
        sonRes
        erveNa
        me":
        "", "Per
        sonPar
        am1": 0,
        "Person
        Param2
        ": 0,
        "Person
        Param3
        ": 0,
        "Person
        Param4
        ": 0,
        "Person
        Param5
        ": 0,
        "Person
        Data1":
        "
        ",
        "PersonData2": "",
        "PersonData3": "",
        "PersonData4": "",
        "PersonData5": ""
    },
    "PersonPhoto": "... (base64) ..."
}
},
"Code": 1,
"Message": ""
}
    
```

(3) Add person information

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListRequest"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	WAN access Ask the required parameters

ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign verification Required for certification
Sign	Permission verification information, when the server Verification when the parameter setting " Management Authority Verification "	N	Fixed 32	String	

	is enabled			
Data	Request content	Y	Json to Elephant	
Action	Fixed "addPerson"	Y	String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y	Integer	
PersonInfo	Added person information	Y	Json to Elephant	
PersonCover	Overwrite (staff already existing) 0 : without covering (exists that the person memberID of the person is not added, return Return error message)	N	Integer	

	1 : Overwrite (overwrite the existing person Employee ID information)				
--	---	--	--	--	--

PersonId	Person ID	Y	<36	String	
PersonName	Personnel name	Y	<64	String	
Sex	gender 1 : Male 2 : Female 0 : Unknown	N	0-2	Integer	
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	

PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON pair Elephant	
PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	
PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	
PersonData5	Custom information 5	N	<88	String	
PersonPhoto	Staff photo (base64 encoding) Pi cture format : JPG	N	<2 M <96 0 * 960	String	If the eigenvalue is passed Way to add people Member, can not bring

					"Personal Photo"
--	--	--	--	--	------------------

					Paragraph .
FeatureValue	Personnel characteristic value data (base64 encoding)	N		String	If the eigenvalue is passed Way to add people Member, must bring Characteristic data Field.
FeatureInfo	Characteristic related information (If you bring this JSON object, it means Add personnel, the equipment will directly check the "special "Value information", no longer mention "personal photos" Take characteristic values. If you add via "Personal Photo" Add people, please do not bring this JSON object.)	N		JSON pair Elephant	Through eigenvalue Ways to add people The brought object.
FeatureType	Person characteristic value type 0 : float 1 : char 2 : int 3 : General type	Y	0-3	Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "	Y	Fixed 18	String	

	personListResponse "				
Session	Register to return to Session	N		String	
Data	Back to content	Y		JSON object	
Action	Fixed "addPerson"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PersonInfo	Added person information	Y		Json object	
PersonId	Person ID	Y		String	
PersonName	Personnel name	Y	<64	String	
Sex	gender 1 : Male 2 : Female 0 : Unknown	N	0-2	Integer	
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	

Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
SaveTime	Add time Format :	N	<20	String	

	yyyy-mm-dd hh: mm: ss				
LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	
PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON object	
PersonCode1	Custom code 1	N	<32	String	

PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	

PersonParam5	Custom parameter 5	N		Integer	
PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	
PersonData5	Custom information 5	N	<88	String	
Result	Internal error code	N		Integer	Code = 111 0, the addition of a failure to return the inside of the operation as

					error code.
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personListRequest", "UUID":
  "abcdefabcde",
  "Session": " abcdefabcdef _ 1555559039 ",
  "TimeStamp": 1542594185,
  "Sign": " ec304c35342cbe0609e7c7fec4d97ca5",

  "Data":
  {
    "Action":
    " addPerson",
    "PersonType":
```



```
e ":  
1, "F  
eatu  
reInf  
o":  
{  
    "FeatureType": 1  
}  
" PersonInfo ":  
{  
    " Pers  
onCo  
ver":  
1, " Pe  
rsonId  
":  
"Pers  
on1",  
"PersonName": "PersonName1",  
" Sex ": 1,  
" IDCard ": "IDCard",  
"  
N  
ati  
on  
":  
"N  
ati  
on  
",  
"  
Bi  
rt  
hd  
ay  
":  
"B  
irt  
hd  
ay  
",  
"P  
ho  
ne  
":  
"P  
ho  
ne  
",  
"A  
dd  
re  
ss
```

```
":  
"A  
dd  
re  
ss  
,  
"Li  
mi  
tTi  
m  
e":  
1,  
"StartTime": "2019-01-01 00:00:00",  
"EndTime": "2020-01-01 00:00:00",  
"  
L  
a  
b  
e  
l  
":  
"  
L  
a  
b  
l  
e  
"  
;  
"  
P  
e  
r  
s  
o  
n  
E  
x  
t  
e  
n  
s  
i  
o  
n  
":  
{  
    "PersonCode1": "",  
    "PersonCode2": "",  
    "Person  
Code3":
```

```
        "", "PersonReserveName":
        "", "PersonParam1": 0,
        "PersonParam2": 0,
        "PersonParam3": 0,
        "PersonParam4": 0,
        "PersonParam5": 0,
        "PersonData1":
        "",
        "PersonData2": "",
        "PersonData3": "",
        "PersonData4": "",
        "PersonData5": ""
    },
    "PersonPhoto": "... (base64) ...",
    "FeatureValue": "... (base64) ..."
}
}
}
return:
{
    "Name":
    "personListResponse"
    , "Session":
    " abcdefabcdef _ 1555
559039 ", "Data":
    {
        "Action":
        " addPerson ",
        " PersonType": 1,
        " PersonInfo":
        {
```

```
" PersonId":  
"Person1",  
"PersonName":  
"PersonName1",  
" Sex ": 1,  
" IDCard ": "IDCard",  
"  
N  
ati  
on  
":  
"N  
ati  
on  
",  
"  
Bi  
rt  
hd  
ay  
":  
"B  
irt  
hd  
ay  
",  
"P  
ho  
ne  
":  
"P  
ho  
ne  
",  
"A  
dd  
re  
ss  
":  
"A  
dd  
re  
ss  
",  
"Li  
mi  
tTi  
m  
e":  
1,  
"StartTime": "2019-01-01 00:00:00",  
"EndTime": "2020-01-01 00:00:00",
```

```
        "
        L
        a
        b
        e
        l
        "
        :
        "
        L
        a
        b
        l
        e
        "
        ;
        "
        P
        e
        r
        s
        o
        n
        E
        x
        t
        e
        n
        s
        i
        o
        n
        "
        :
        {
}
    },
    " Result ": -1
},

"PersonCode1": "",
"PersonCode2": "",
"PersonCode3": "", "PersonReserveName":
"", "PersonParam1": 0, "PersonParam2": 0,
"PersonParam3": 0, "PersonParam4": 0,
"PersonParam5": 0, "PersonData1": " ,
"PersonData2": "",
"PersonData3": "",
"PersonData4": "",
"PersonData5": ""
```

"Code": 0,

"Message": ""
}

(4) Edit staff information

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListRequest"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
Channel No	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	

TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification n
Sign	Permission verification information, when the server Verification when the parameter setting " Management Authority Verification " is enabled	N	Fixed 32	String	
Data	Request content	Y		Json to Elephant	

Action	Fixed "editPerson"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PersonInfo	Edited personnel information	Y		Json to Elephant	

PersonId	Person ID	Y	<36	String	
PersonName	Personnel name	N	<64	String	
Sex	gender 1 : Male 2 : Female 0 : Unknown	N	0-2	Integer	
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Personnel effective start time Format :	N	<20	String	

	yyyy-mm-dd hh: mm: ss				
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	

PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON pair Elephant	
PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	
PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	

PersonData5	Custom information 5	N	<88	String	
-------------	----------------------	---	-----	--------	--

PersonPhoto	Staff photos (base64 encoding) Picture requirements : JPG	N	<2 M <96 0 * 960	String	If the eigenvalue is passed Way to add people Member, can not bring "Personal Photo" Paragraph .
FeatureValue	Personnel characteristic value data (base64 encoding)	N		String	If the eigenvalue is passed Way to add people Member, must bring Characteristic data Field.
FeatureInfo	Characteristic related information (If you bring this JSON object, it means Add personnel, the equipment will directly check the "special "Value information", no longer mention "personal photos" Take characteristic values. If you add via "Personal Photo" Add people, please do not bring this JSON object.)	N		JSON pair Elephant	Through eigenvalue Ways to add people The brought object.
FeatureType	Person characteristic value type 0 : float 1 : char 2 : int 3 : General type	Y	0-3	Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListResponse"	Y	Fixed 18	String	
Session	Register to return to Session	N		String	
Data	Back to content	Y		JSON pair Elephant	
Action	Fixed "editPerson"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
ChangePhoto	Whether to edit the photo 0 : Unedited photo 1 : edited photo	N		Integer	
PersonInfo	Added person information	Y		Json to Elephant	
PersonId	Person ID	Y		String	
PersonName	Personnel name	N	<64	String	
Sex	gender	N	0-2	Integer	

	1 : Male 2 : Female 0 : Unknown				
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
SaveTime	Person add time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	

PersonExtension	Extended personnel information (customized storage Content)	N		JSON pair Elephant	
-----------------	---	---	--	---------------------------	--

PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	
PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	
PersonData5	Custom information 5	N	<88	String	

Result	Internal error code	N		Integer	Code = 1110 , the failure to add the internal operation returns failure error code.
Code	Return opcode	Y		Integer	

	1 success				
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personList
  Request",
  "UUID":
  "abcdefabc
  def",
  "Session": " abcdefabcdef _ 155559039 ",
  "TimeStamp": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    "Acti
    on":
    " edi
    tPer
    son
    ",
```

```
" Person
Typ
e ":
1, "F
eatu
reInf
o":
{
  "FeatureType": 1
}
" PersonInfo ":
{
  " PersonId":
  "Person1",
  "PersonName":
  "PersonName1",
  " Sex ": 1,
  " IDCard ": "IDCard",
  "
  N
  ati
  on
  ":
  "N
  ati
  on
  ",
  "
  Bi
  rt
  hd
  ay
  ":
  "B
  irt
  hd
  ay
  ",
  "P
  ho
  ne
  ":
  "P
  ho
  ne
  ",
  "A
  dd
  re
  ss
  ":
  "A
```

```
dd
re
ss
",
"Li
mi
tTi
m
e":
1,
"StartTime": "2019-01-01 00:00:00",
"EndTime": "2020-01-01 00:00:00",
"
L
a
b
e
l
"
:
"
L
a
b
l
e
"
,
"
P
e
r
s
o
n
E
x
t
e
n
s
i
o
n
"
:
{
    "PersonCode1": "",
    "PersonCode2": "",
    "Pers
onCo
de3":
    "",
```


"Person
Name
": "",

"
P
e
r
s
o
n
P
a
r
a
m
1
":

0
,

"
P
e
r
s
o
n
P
a
r
a
m
2
":

0
,

"
P
e
r
s
o
n
P
a
r
a
m
3
":

r
a
m
3
"
:
0
,
"
P
e
r
s
o
n
P
a
r
a
m
4
"
:
0
,
"
P
e
r
s
o
n
P
a
r
a
m
5
"
:
0
,
"
P
e
r
s
o

```
        n
        D
        a
        t
        a
        1
        "
        :
        "
        "
        ,
        "PersonData2": "",
        "PersonData3": "",
        "PersonData4": "",
        "PersonData5": ""
    },
    "PersonPhoto": "... (base64) ...",
    "FeatureValue": "... (base64) ..."
},
"Result ": -1
}
}

return:
{
    "Name":
    "personListResponse"
    , "Session":
    " abcdefabcdef _ 1555
    559039 ", "Data":
    {
        "Acti
        on":
        " edi
        tPer
        son
        ",
        " Per
        son
        Typ
        e ": 1,
        "ChangePhoto": 1,
        " PersonInfo ":
        {
            " PersonId":
            "Person1",
            "PersonName":
            "PersonName1",
            " Sex ": 1,
            " IDCard ": "IDCard",
```

```
"
N
ati
on
":
"N
ati
on
",
"
Bi
rt
hd
ay
":
"B
irt
hd
ay
",
"P
ho
ne
":
"P
ho
ne
",
"A
dd
re
ss
":
"A
dd
re
ss
",
"Li
mi
tTi
m
e":
1,
"StartTime": "2019-01-01 00:00:00",
"EndTime": "2020-01-01 00:00:00",
"
L
a
b
e
l
"
```

```
:  
"  
L  
a  
b  
l  
e  
"  
,  
"  
P  
e  
r  
s  
o  
n  
E  
x  
t  
e  
n  
s  
i  
o  
n  
"  
:  
{  
  
    "PersonCode1": "",  
  
    "PersonCode2": "",  
    "Person  
Code3":  
    "", "Per  
sonRes  
erveNa  
me":  
    "", "Per  
sonPar  
am1": 0,  
    "Person  
Param2  
": 0,  
    "Person  
Param3  
": 0,  
    "Person  
Param4  
": 0,  
    "Person  
Param5  
": 0,
```

```

        "Person
        Data1":
        "
        ,
        "PersonData2": "",
        "PersonData3": "",
        "PersonData4": "",
        "PersonData5": ""
    }
}
},
"Code": 1,
"Message": ""
}

```

(5) Delete list

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListRequest"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
Channel No	Channel number (manage NVR channels Of the IPC , the value 25.5 for managing NVR & It)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	

TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign verification Required for certification
Sign	Authority verification information, when the server parameter is set to " management authority verification " Verify when turned on.	N	Fixed 32	String	
Data	Request content	Y		Json to Elephant	
Action	Fixed "deletePersonList"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed " personListResponse "	Y	Fixed 18	String	
Session	Register to return to Session	N		String	

Data	Back to content	Y		Json to Elephant	
------	-----------------	---	--	---------------------	--

Action	Fixed "deletePersonList"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

```
send:
{
  "Name":
  "personList
  Request",
  "UUID":
  "abcdefabc
  def",
  "Session": " abcdefabcdef _ 1555559039 ",
  "TimeStamp": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    "Action":
    " deletePersonLis
    t ",
    " PersonType ": 1
  }
}
```



```

    }

    return:
    {
        "Name":
        "personListResponse"
        , "Session":
        " abcdefabcdef _ 1555
        559039 ", "Data":
        {
            "Action": " deletePersonList ",

            " PersonType ": 1,
        },
        "Code": 1,
        "Message": ""
    }
    
```

(6) Delete personnel information

Request parameters

name	definition	essen tial	Length / R ange	Types of	Remarks
Name	Fixed "personListRequ est"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
Channe lNo	Channel number (manage NVR channels Of the I P C , r e v e 2 5 . 5 for managing N V R & It)	N		Integer	NVR docking Required parameters

Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign verification Required for certification
Sign	Authority verification information, when the server parameter is set to " management authority verification " Verify when turned on.	N	Fixed 32	String	
Data	Request content	Y		Json to	

				Elephant	
Action	Fixed "deletePerson"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PersonId	Person ID	Y	<36	String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
------	------------	-----------	----------------	----------	---------

Name	Fixed "personListResponse"	Y	Fixed 18	String	
Session	Register to return to Session	N		String	
Data	Back to content	Y		JSON pair Elephant	
Action	Fixed "deletePerson"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	

PersonId	Person ID	N	<36	String	
Result	Internal error code	N		Integer	When Code = 1110 , the internal operation error returned when the deletion fails code.

Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personList
  Request",
  "UUID":
  "abcdefabc
  def",
  "Session": " abcdefabcdef _ 155559039 ",
  "TimeStamp": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    "Action":
    " deleteP
    erson ",
    " Person
    Type ": 1,
    "PersonId":
    "PersonId1
    "
  }
}
```

return:

```
{
  "Name":
  "personListResponse"
  , "Session":
  " abcdefabcdef _ 1555
  59039 ", "Data":
  {
    "Action":
    " delete
    Person
```

```

    ,
    "PersonType":
    1,
    "PersonId":
    "Person1", "Result":
    -1
  },
  "Code": 1,
  "Message": ""
}

```

(7) Add personnel information (characteristic values) in batches

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListRequest"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	

TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification
Sign	Permission verification information, when the server	N	Fixed 32	String	

	Parameter setting " Management Authority Verification "				
	Verify when turned on.				
Data	Request content	Y		Json object	
Action	Fixed "addPersonList"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
FeatureInfo	Characteristic related information (Only support "characteristic value" to add people in batch)	Y		JSON object	

FeatureType	Person characteristic value type 0 : float 1 : char 2 : int 3 : General type	Y	0-3	Integer	
PersonInfoList	Personnel information list	Y	<100	JSON array	
PersonId	Person ID	Y	<36	String	
PersonName	Personnel name	Y	<64	String	
Sex	gender 1 : Male 2 : Female 0 : Unknown	N	0-2	Integer	
IDCard	ID number	N	<32	String	

Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	

StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	
PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON object	
PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	

PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	

PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	
PersonData5	Custom information 5	N	<88	String	
FeatureValue	Personnel characteristic value data (base64 encoding)	Y		String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListResponse"	Y	Fixed 18	String	
Session	Register to return to Session	N		String	
Data	Back to content	Y		JSON object	
Action	Fixed "addPersonList"	Y		String	
PersonType	List type:	Y		Integer	

	1 : Blacklist 2 : Whitelist 3 : VIP list				
--	--	--	--	--	--

PersonInfoList	Personnel information list	Y		JSON array	
PersonId	Person ID	Y		String	
PersonName	Personnel name	Y	<64	String	
SubCode	Batch operation list return code (See Code list)	Y		Integer	
SubMessage	Batch operation list return operation information	Y		String	
Result	Internal error code (See Result list)	N		Integer	
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personList
  Request",
  "UUID":
  "abcdefabc
  def",
  "Session": " abcdefabcdef _ 1555559039 ",
  "TimeStamp": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":

  {
```

```
"Action":
" addPer
sonList ",
" Person
Type ":
1, "Featur
elInfo":
{
    "FeatureType": 1
}
"
P
e
r
s
o
n
l
i
n
e
L
i
s
t
"
:
[
    {
        " PersonId":
        "Person1",
        "PersonName":
        "PersonName1",
        " Sex ": 1,
        " IDCard ": "IDCard",
        "
        N
        ati
        on
        ":
        "N
        ati
        on
        ",
        "
        B
        ir
        t
        h
        d
        a
        y
```

":
"B
irt
hd
ay
,
"P
ho
ne
":
"P
ho
ne
,
"A
dd
re
ss
":
"A
dd
re
ss
,
"Li
mi
tTi
m
e":
1,
"StartTime": "2019-01-01 00:00:00",
"EndTime": "2020-01-01 00:00:00",
"
L
a
b
e
l
":
"
L
a
b
e
"
,
"P
e
r
s
o

```
n
E
x
t
e
n
s
i
o
n
"
:
{
    "PersonCode1": "",
    "PersonCode2": "",
    "Person
Code3":
"", "Per
sonRes
erveNa
me":
"", "Per
sonPar
am1": 0,
"Person
Param2
": 0,
"Person
Param3
": 0,
"Person
Param4
": 0,
"Person
Param5
": 0,
"Person
Data1":
"
",
"PersonData2": "",
"PersonData3": "",
"PersonData4": "",
"PersonData5": ""
},
"FeatureValue": "... (base64) ..."
},
{
    " PersonId":
"Person2",
"PersonName":
"PersonName2",
```

```
" Sex ": 1,  
" IDCard ": "IDCard",  
"  
N  
ati  
on  
":  
"N  
ati  
on  
",  
"  
Bi  
rt  
hd  
ay  
":  
"B  
irt  
hd  
ay  
",  
"P  
ho  
ne  
":  
"P  
ho  
ne  
",  
"A  
dd  
re  
ss  
":  
"A  
dd  
re  
ss  
",  
"Li  
mi  
tTi  
m  
e":  
1,  
"StartTime": "2019-01-01 00:00:00",  
"EndTime": "2020-01-01 00:00:00",  
"  
L
```

```
a  
b  
e  
l  
:  
"  
L  
a  
b  
l  
e  
"  
;  
"  
P  
e  
r  
s  
o  
n  
E  
x  
t  
e  
n  
s  
i  
o  
n  
"  
:  
{
```

```
"PersonCode1": "",  
"PersonCode2": "",  
"Person  
Code3":  
"", "Per  
sonRes  
erveNa  
me":  
"", "Per  
sonPar  
am1": 0,  
"Person  
Param2  
": 0,  
"Person  
Param3  
": 0,  
"Person  
Param4  
": 0,
```

```
        "Person
        Param5
        ": 0,
        "Person
        Data1":
        ",
        "PersonData2": "",
        "PersonData3": "",
        "PersonData4": "",
        "PersonData5": ""
    },
    "FeatureValue": "... (base64) ..."
},
...
]
}
return:
{
    "Name":
    "personListResponse"
    , "Session":
    " abcdefabcdef _ 1555
    559039 ", "Data":
    {
        "Action":
        " addPerson
        List ",
        " PersonTyp
        e ": 1,
        " PersonInfo
        List ":
        [
            {
                },
            {
                },
            ...
        ]
    },
    ...
},
    ],
    },
```

```
" PersonId":
"Person1", "PersonName":
```



```
"PersonName1", "SubCode": 1110,
"SubMessage" " ",
" Result ": -1
```

```
" PersonId":
"Person2", "PersonName":
"PersonName2", "SubCode": 1110,
"SubMessage" " ",
" Result ": -1
```

```
    "Code": 0,
    "Message": ""
}
```

(8) Delete personnel information in bulk

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "personListRequest"	Y	Fixed 17	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection

Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification
Sign	Authority verification information, when the server parameter is set to " management authority verification " Verify when turned on.	N	Fixed 32	String	
Data	Request content	Y		Json to Elephant	
Action	Solid fixed "deletePersonListPart"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PersonInfoList	Personnel information list	Y		JSON number group	
PersonId	Person ID	Y	<36	String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
------	------------	-----------	----------------	----------	---------

Name	Fixed "personListResponse "	Y	Fixed 18	String	
Session	Register to return to Session	N		String	
Data	Back to content	Y		JSON pair Elephant	
Action	Solid fixed "deletePersonListPart"	Y		String	
PersonType	List type: 1 : Blacklist 2 : Whitelist 3 : VIP list	Y		Integer	
PersonInfoList	Personnel information list	Y		JSON number group	
PersonId	Person ID	N	<36	String	
SubCode	Batch operation list return code (See Code list)	Y		Integer	
SubMessage	Batch operation list return operation information	Y		String	

Result	Internal error code (See Result list)	N		Integer	
Code	Return opcode	Y		Integer	

	1 success				
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "personListRequest", "UUID":
  "abcdefabcde",
  "Session": " abcdefabcdef _ 155559039 ",
  "TimeStamp": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    "Action": " deletePersonList
    Part ", " PersonType ": 1,
    " PersonInfoList ":
    [
      {
      },
      {
      },
      ...
    ]
  }
}
```

```
    ]  
  }  
}
```

" PersonId": "Person1"

" PersonId": "Person2"

```
return:  
{  
  "Name":  
  "personListResponse"  
  , "Session":  
  " abcdefabcdef _ 1555  
  559039 ", "Data":  
  {  
    "Action": " dele  
    tePersonList  
    Part ", " Perso  
    nType " : 1,  
    " PersonInfoL  
    ist " :  
    [  
      {  
  
  
  
  
  
  
  
  
  
      },  
      {  
  
  
  
  
  
  
  
  
  
      },  
      ...  
    ]  
  },  
}
```

" PersonId": "Person1",
"SubCode": 1110,
"SubMessage" " "

" Result ": -1

" PersonId": "Person2",
"SubCode": 1110,
"SubMessage" " " ,
" Result ": -1

```
    "Code": 1,  
    "Message": ""  
}
```

Picture data base64 example

```
"/ 9j / 4AAQSkZJRgABAQAAAQABAAD /  
2wBDAAIABAQEBAQIBAQECAGlCAgQDAgICAgUEBAMEBg  
UGBgYFBgYGBwkIBgcJBWYGCAslCQoKCgoKBggLDAsKDAkKCgr /  
2wBDAQICAgICAgUDAwUKBwYHC goKCgoKCgoKCgoKCgoKCgoKC  
goKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgoKCgr /  
wAARCADKAlI DASIAAhEBAxEB /  
8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL /  
8QAtRAAAgEDAwI EAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhBy  
JxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoK  
+ + Pn6 TI5ufo6erx8vP09fb3  
mqKjpkWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4  
So0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd  
4eXqDhIWGh4iJipKTlJWWl5iZ /  
8QA HwEAAwEBAQEBAQEBAQAAAAAAAAAAECawQFBgcICQoL /  
8QAtREAAgECBAQDBAcFBAQAAQJ3AAEC  
AxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEI8RcYGR  
omJygpKjU2Nzg5OkNE  
RUZHSEIKU1RVVldYWVvpjZGVmZ2hpanN0dXZ3eHl6goOEhYaHilmKkpOUlZaX  
mJmaoqOkpaanqKm  
qsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4uPk5ebn6Onq8vP09fb3 + Pn6 /  
9oADAMBAAIRAxE APwD523eY3 +  
flpGiTd67q0pPBXjG2mnt73wfrMElt5X2mOfRLmJovNVZlmdGu1XV0ZWb5WVIZ  
WZ WrUv / AIPfFvSdSI0XWPhX4ksbu21JNPuYL / SZLYwXTqrrE /  
m7VRtrq24tt2tu3Yy1csTk9pCJyn2fb1qB7  
CHZtPO5cNXfp8APjQNbj8K33wy1ewvdpdXOliHVFS323YkWMxu8rhEO8qNxbb84  
+ bBBq1pv7MXxjv /  
Elj4QuvBzabqOoXTW1tBql7FbqZVkePY7OxVD5iMmS33uO9HKhxrR6s8T0vTP +  
Ed1yXQWybW6LS2 hHATn5k / AnP0NWtQ054V84AIBXuFeg /  
Eb9nb4iaBpD3uv6RFbvbW0d9DLb3lIZYyu5WITdkEHn0yc9 K88F /  
qWsaR9stLeNw3Eit1Dd6k2j70bISX5yEH /  
fVlslxs9KqXMGsKpuUkhTZ2HU1zXiTXfF0k0A004kG  
SwkMVtv2n0rCpP2cbm1CI7WXKdnblD + QG4qyikgNLHgA9a4y18JfHPX9PN /  
o0124UYIT7Ptwc / 4Vra R8EPjff38Fje3GpKJAWI8qXYsY9T8vSuN5haVj015dDI +
```

I6TAQoY3rnXu4rL4ho0uALiWYD5u6n / 69W / H ng688BRwWlrvV /
eFcDUbyW6zGrnogrkNRaGTXbAQu7KzMskrP1OM13Yer7aNzz69H2NSx3d5qcC
f u0myTxxVKZ4IG6WVc /
71ZsenWMP3yKd7c8mprTRRdN5VIZPLlZchFZq3 iYR3lbyW2LHNyVzH + / Wbc

aIYiUAXP1wa6B / BWuhSR4anxnH / AB7NVSTwlrG + U + HLkFe /
kNxUjMr + 19PHAnX / AL5oqy + h60HIOg3 PX / ng1FVZE3R + nX7K3x
++ B3jrTx8F / i / ca /
dapb6FHouialdWsSQIbW5jurO2aVn3AEReRG78J5qxlgNp
HsmrXHwH1Dw7qOm +
LPhr4r1CJfDml69paajcxL9sjsGQWfmLu2sLS5YAP8AlzW + zl +
TP52fA39qHxl 4T06H4TTX / he3ud + NG8YaxoFtPPHHu /
1DzOhKHqqv7KM19l / C79sWy1n4epoXxO / aL8LaVqNjpU9j bXkSrl06
/
a4JldyxQMqtHF58lcY3lyhssmT1RIA8SpCrE9rQfBrW7nUPhZdfswarqN
5o / iNrW1 / tfVndb2 NVFrlVm8piCptodqPu3K +
QSQRWpoOseG7rRG1EfsAXVzrEd1ONPg1 /
XZ7iFo23XA8uYWrFnMrS5Uq CDghmB48h1j9tH4XXtjF5 /
wC3FJYaklzau19pGn3zrP5dq0bOQvtjczpFlwxsc7zj5iDpSft3fs6anfWHi
G 8 /
bn8cadc2M0k09hpOg6o9pcSNLI2FhKhPLKuVCfwjaBjAlmXlaYdVW /
hPSbRPilpXhe28P + G / + Ceeg
QanLZSxw2urJc30EkP2jY11kYowVYpAdr42gMBkAV + VP7av7Nnxa /
Y3 / aCvNF + lfwvHhvQ / GMkupeH be2kaW0hVnO6CNI7N +
7JwVZiwXacnlNfoffftMfsx2fgEftC3X7THxdvLF7z +
yxHbR3EvkzR7C3mWc9z v8uTyid4AALEbga + Y / wBuX9oX /
gmf8cvghqXh7wfr3xIHjZ5FI8Ma14kiiS0t5 / MUOJS9w5jiKknkgZG0Ht
XFKUJaxPXo06sNjXPhzW9WvIrpvK2kj1Q7Qa2 /
gb4z8JeGfEN9L4wu7aKCaDiOZGbzDnlQV5yPevAfi R8XU8J +
lr3w43iCvypLKYwtdadeCWGYr / Ej4 + YH6VyF38WbnUCI5a +
KbeBgcpC6khs9cn + Go9nzQN4 StLQ / QHSf2svgf4Cha71GG +
h0t52RphYhlyP9nOfz61vf8Nd / CrxPNPZeGIJlrNoVE +
oXTRoUVhn7gYt6 Y9c18EfD9dW8eanHoGqfEPS9JjIXevmETxyL6lem7
/ ZJB9M1t / GPwfb + APF9po2j + MZ9QtXhhnhlCKrv
JkBkKrwMeh7Vx1aa5fhO +
hTVSfxHu3xB8deE774eax4Y8LTCW5j1xGmmISTzFyQ2Sx +
9kcCvLbkNbQ WlwXOY7tScnnqB / Wut8daRf6Zq +
tx6gYH837K3n2x2jPILjKjj0H4VxuqW0lxYPLLcMqgArtbHHFa4H +
GzDGx9nWopa5to497PyBxUug +
MNQ0m3mNlqRtwzth84J9hWRZ21u8IZb2Qkrn5q5b4rKsHw8nz
MARexlcnr + 8Su2BxHqf / AAsXxScouvzkdm8zvWTr + t /
FPXLSSDw34 + jsbgn5J7puPcYwc1574O1W + uZt Ta6kYp +
5KRmTAX5j0qn4l1K9W9c7kVR0w3SiO4HVNafthlj / wAX40vr2tf /
ALCivOZNT1LzG / 0 + Xqe7
UVVmZ2kfWF3Yadd2r2lxaKyyJtYBaradqU + h3Q03WHDof + Pe6 /
vezf7X86nWS6UqVhH / AH1VLUVm urNxxqEMQi2 /
MD3qSuU2G1eKFchz / ALtaXww +
lfgzSviz4Yk8b6dDqGkDXbRdS06d8JNbtKsbqcfMfvZ
x3xjvXiPiP4I65pxbRtLnQxK2xbl / wDHSa8N +
LHxf8UWupCz0LxdiRHZLLbnayuvPBH + NKVNyhYqIKFO Z + IP /
BWP9of4Tfs922keM / 2ctJexvNQ8SS /
2rpek3gGIXIS3TCtblxSPByNgGGdfPulFfIT8Wf2i / HPxP1K
R9UmtbWNizC2sofJTlicYU1zniDxx4r18s2t63e3TeYz7rm6dxxubv8xNc7

O89wQXXee2Fr nwmGjhqdn
K7OzG42eMmnGPLHsNudVv5pESSQbSetIlgkdWDluBxTI0u7dSn2dge
+ 5av2OkXEgMm3tiu04oxnEk8 NeJNT0G5V7e6Zk8xWKDoCOc8969Hj
+ Pj63q8Or6 + rTyW7IUaTnO0g8 / XpXnsfh2VQeMHFUm0ySGX O /
jvWcqcZmlOpOmz6a1b9pLQ / ib8Qltbsl / sT3lqFvoT8sKFRgcZ9K6f +
2dGvNMIaG9hO9W2kE8kjp + lf
llvJ9mDIxBz8pJ613vbg4za5oaJp8twrIDt2SKGBGPepp0 /
ZRshSqzqSuz6R0i6trjTYSXAwg + 79Kz / ABP4 UsvF + I / 2NeXMiR /
aEl3xjjKsGx + IYXw1 +
IGj69YRRwEQS7yGCtlD6cfw13doFMrW84Gc53DofeqliYFn4S
sdAtp7eK7kmNy2 + Qy9gFwAK5nUvDd1dXjmNcAE4BB4r1P +
zoHOPTJqu2k2lbfJEOeTSiM8kbwlrG44
AxnrRXrv9IWPYwgdgaKu4uU9A3gtujDD / eWud8ealJDbnTYyyyXHQ /
3R3rQOuz6TmLUkaWNV + S5 WPP / vDt9RXkP7T /
xgj8FacZ9NnX7XOnlW7k5Kk9SKmBqeUftBfGK6W8l +
H3hmMRpA226uV / i9vr61
5LbtczMZ5LxmXNRPfta5qE15dzSSPK7NNM5yXb1qRSMCKPonXFae
RjzEbxvcyCIP + PrXX + DPCUAgSZ oDM2fmBXp9Kw / C +
ky6jrEUSRM2XxwK +
qPgv8A5tTto5Dbb0POWTpXm18V7M9jA4B1Dx7T / hnqut
b7u00SUhr8u2LAqs3ws8TQMvFRZUAOMshr9APAHwRtLCFba506PZ
gD7vGa6uD4CeGL9v8ATdGR8 H5R5Y4 / OuBZhK56n9IRPzdsfgv4qu /
uadJtKkA7etYvil4J + KdOZnbS5kl65GeK / Uiw + BGh2AMcWiQFT
zzEP8K5b4ifBTRb +
zdrXTlxKgwA0fBFaU8w94yq ZTHIPyt1LR72xJgu4yhU9GSssho36mvsb
43fs + aNcm
WNdOe3mySdsXBP1xYyx458Eal4S1BoJELx7yOARXqYfExqHh4jCyoy
L / wt8UT6VrKRTXUiox + UA4DH 0r6d + H2v2uvaZHHi +
ZFPyOW546ivjSz1BrSRJoV + 6wlPoa9j +
FHjiXUGnPOufYjMuROSRsZTkj16VpUla NzlhG8 + U +
n7O3t5YQzAcjOSetH2C1ZjiMZz8v9a8h1vULPSrM6rcfEDU5YVTbKLZ
W2YPTk981q / B68u
NZ1a5upNTvGhtj5bC5kOTuTdz71y0sRzzOt4XljzHo502yzxBGR6k0VyF
34quXupXt2Tyzlxj + meKK6OZ
mPKux6Zc3bmNwu07h82418bfta3k1748t9OIkRIYUZ1hU8hizc /
jXs0n7SHhF4nSXVTu2cIVJD18zFP
xQPGXxDvNXaQtHn91n0HC1NGp7Q6MThKmGjzSiZen2RhtCehardlpz
sojxkvUVqxkUKf7tdV8L / Dtx4

o8RQ2EETfNIB8ozTrVOWIsJR9pVij0f9mf4KjxLrMV5eRts34VsV +
if7P3wUt9N0iNfsoAKAE7a8z / AGRf gHbaZYwz3Vljbydy19o /
DPwhp2mWaMU3DH3v2x5
TesXmifJs KLuHcdTXN67oOFkuHyAFICNTpnPONonyv8R
TfXqN5p9o0YCRYA6ACorbQncFoo1AxXMdnleexeGGgQK8LDJ6YrD8UeELaSHZ
9n3bsnJ + + / + OnwatsrzS BFpPLMJ1ySRgr2r5V / ZVBGcPt6ivu /
WAF aAHmlzHkAHAHY14j8R / BkGp6XOjp 8XdjivUw9TIPCxVHmPzE8feFp + / C +
qPEEBQnAYdqX4e621nqi20 8g WXerZ5yOv5g4r1T9o3wK9obpl4 + / +
Fe3SftKVj5upH2dU nQsxGO2RXhOmXJt71J0O1o5Pmz. 4 + / + Df of
wZuPGfhCCWHxJcC1uFDbYY15x2zjNT Cng9lni8hbiV4o7qUljLLsGNzeuK9i MYNJ
+ / + Beha ZK8OaL4p
/ DMSIzA0chU / ckUsD + grjv + CgPgH7H8FFvbORALPxDbu6yfebzF8sY +
hKn8K5KsVQzI9PLovGYiNJ / aP Flft8O7eFIGlclFckiPrjiivJobK1WFFa3tchXQFQ3

/ 3RXkGrFltWIAU17Qnw7 + N96y7PC2uNnpts3AP6V5V8RPD + peG /
Er6RrGnS2lyMF4LhNrBe1dWW wnDm5jk4jdKdCPs2VLeZ4YIEXO7p9K
+ u / 2APgPL4jni8S39qSrHEeV + 90r5h + F / g268aeJ7XSbeMsDlqt
X6n / sp + E / DXwx8F2iX0sUC2sOSzuBlvYd6jGVOBQ8vLaEYe / I9y +
F / gSz0HTobeVNuFHQYr1XwfZ2rMsE bfKgznPWvGbD4z +
HLsRr29kKWaNgzuOD9BXXeG / 2pvgXogvZ5LmXzPvMkW0k / SvJ +
rnvxxMI6Hsb6 TFcW /
mRwH5Vy22q2IW8ku4qmA0gIrA079p74Y3mnY0WKUyzYCbxniux8M6
1o + o2X9pQgYcZ2gdK
z9nynVTI7SOhtI0hp5cnbuUHtWF4k0GdHy4yGJwMda7GbUrGzh +
1Oq / NkgDtXkPx9 / aO0X4aWyzfY PtE +
0tAq881dKnzSOevU9nE5T4gaEschiwFkIPy4xXkPi3R433QOqDdkEd64
b4rtufF / xjcyXWheEl01V iAY4yWxXlep / H / wCL95eK +
q6XdWxJ5RoDg / nXftw77ng18S59Dhv2pPh6DNctBCeVJJA +
lfe11aGz1q a0cY2Tn9a + 3 / iZ498RazlI9b0kGORdvmxoc8 +
tffxb0INF8e3UEcZCykEAjpnBr08JLldjxsXHm1P0T / 4Jq al9u +
AEvhueRs2OoM0W4 / 3tpq5 / wAFB7ZX +
A1zMF4bXLIfUhhXLf8ABOXxLp + i + GbrQblwokEUo9SS pVv5Cu5 /
bwltDQ / Zm1C / t3yF8SWSruX3FY4 / + Az0uHY3zWjHz / Q +
G7aTWvs8f6ljrsG1vsudwx1zRW ta + MJLa2jtl0i2YRxxLZOBjPWivl +
afY / XnQd9jO9d8P6lFPKmlxaaYkja3Zp + W + vp3r8d8 4g2 +
nW94lvaxxlyVyEQxtuY / r + dflH8Trka / 451bVVbe s +
pzSq7dSu8hf6fX2cYxifcpTly8x7d + wv8LrvX1m8UW9kZpUuBHbkjOSK
+ 6Ph98BvFd5Kt74in3b0BR JT8kY9AvrXmv8AwSp + FLT / AAe0 /
XZrM5nuJp846gvhf0FfUPxU + IOlfCzRjB66KLsi3KrcA / 5xXhVK372U
T6XC0f3EZFTQfgBp3jO0RNfndYI9yNAT5YBUkE / j1rN8R /
ATwX4TiJ0VslDjdLMJMfjXzf4o / bv + IWqprG
qeAPB11rkOjwPcXbSI4raJFPPA5cD + VeOWf7efxU8d +
MbHS7Kw0YzahcQJDb6dDNGRJJwUyTj5T1Jpx
hKUBkSrqnO3KfePhXQ7HTr2lyMu1XyhU8GvpH4ZEzaNG8e3YV4Ibmvi
P4L + JPIx4zlm8O6nodxbanZE ebAwzvTP31Pcdq +
zv2cbK81DSo7bUoTHNEdsobsa4qkvePXw3w3NLxZqP9l2bdilycivilL4o
+ MdO1nxF Jb3qLKyFgoYdK + tfjlocjpkxcDaqHBAr8 + vie +
uW1v4p8W2cU00mmxP5FvApZnc8Dgc4 / wrWjuceKqe
8dx4f134VeG5hc +
Jrm3VwdzRhFJH9KofE341fAa9tltmSFd52rKYIbH1wTivjT42 /
Dj4zpbIPDFjdeKbTxB pkBvne1cy6bcrJuaOPDDaDwCTnlyK3vh1 +
xj411n4UteatZXIl4invpJYo4nKrbQ8YByTnnJxXdyJRvzHjK c5z +
E9g1m28HeK4vJ0k288LEqrxKK + J / 2y / hxP4T + JNul4mCXFuSjEddp
/ + vX2H + zt + zp8R / C + ofYvEOtp cwrzISvJ /
H1rgv8AgpN8Ko7Tw7pviuKA7tOvPLmYf885MD9CBXRhqnv2OTFUW4
3M / wDZl1 / QdNIgf UYnXZFAolYjkkZ / Svff2zrOC5 /
Zl1i7MioB4msWX3yRXyp + zXqdrM QhhllqkcUgB53KTg / IX1V + 2Pb
/ Z / 2O7xZULJ / wk2mAtjtk1WP0oNnbw / T5c2w7X8x8WW /
g6S5t0uP7d09PMQNse6wVyM4I7Gip4PCW
iSwJKfGVIGWQExuDIcjoFCUV8vzH7JzM / Qr / gp /
8OfHfw28YN4SfS3j1jUtJtlOgVh + / kmna1jUHpzL8o
9e9fRLo15f6vJClt3lzGMkHIADEM2R75 / Ov2l / 4LI / D +
61j4L3n7QWm + Kbkal4dt9Mt4441DrHCL1iZiS
crtazD9E3V8JfBX9mXS7 /
8AYV8RfGtLKJprLxQNMMpO6WaJZopSVH8Cnejbj9 / fjoK + 2qe7GR /
P1Ct7a UD7b / 4J6eBdP8N / s / eFbW2jHzaLCxPuVDV634n + AGI +
P2mvtb04SqW2qG5AX0rz39gnWk1D4OeH4 ZYwvIWiw4lxtC / L /
SvrjwBpyTWK290keSeMH0Oxr5GrGXtT9Bw1O + HjY +

ZLP9kb4caHp00CeFfKjul2j kFjGAXU8FWH8WfevE9U / wCCf /
wR8Kel18ReA / hdcw38cxaKdkKiNvUDJ2n3r9JdU8J6f5QVowT1yEr
F0vwDoV3eeVd26kqxJyOv41pCc4e6S8NCTvY + X /
2evgHfeEpz4t1HSyl15RjVi7MzKeSDk8c17v8ACH QbjSr +
S6vl8PK7MRXfN4U0uytmXbFGgTKLWFZ6hY2c7rAw3DjlrCUveOqnR9
3QzPjCF1Wwmgjl + Uw
kA44r5G0v4fPp3xCvbd4N0V8xDE9MV9NfFDV72O0n37trRkhs8CvGbD
VLCfxFbrNKvmMepNbUfh5j

gr0 + WVmZUP7NsxvzeafCmxxwSuQK14P2e9YeEw3l5tjY /
wACY4r2zwhbRPP6KwGCOpFa0enKkZkkU
FegXHFxzZkJUFJG4HdfCTRvDGNgkQJYZMhGMnJr5X /
AG4PAR8UfCbxDpMduHljtg8GBySGBGK + 5 PjHf2ljYvEkK7tnCryAa +
N / 2kPFsGn + G703R3SStHEqg9cyKMfIXXguaMzx8wpxjA +
Dv2TYZrH4pzaDGv 7ueZUiB4w2RnNfdX7aPhiS6 /
Yl8UX5fYdK8QaEZFA4bzbXlR / 6HXxR +
zvpM2pfFbUdXtEZFTdXLWYdsN 7Ej8Bivvn9r92l / wCCf /
xBupUBkfXPC4UjJL / 2tAx4 + hP5V6GM9 +
gYZLKVPNMP6n5qxae8kavgfMoNFQ
Q3l8sSqsBICgAg0V83Zn7Y2rn7h / 8ABQkap8Z / gF4k + FGi /
DrWlrxPpc9qu60AZmWC6njROcBpHhjQZ 5AJ74r4p / Zy8WelR9njS /
h9NoMcekeMvC1jNdh4h ++ u7Z44mYf3HH2cKeOx9K9W8R /
tReGfGNpJa OnjCaRFSW3kbQ7zEc0ZLI / K /
gfYmvGfh78UrDQPEGufDzWvC2ug2ev3ep +
H7eDRJt0FnezSXcUZH03 a0syjOAQgr66rLmhJH88YbDVaM47fej6k /
Zx8L6j8KNvhbVY9kciC5tQT0Ruo / Q / nX034B8Swb4THPh
XGMmvkPwH8eNN8fa1pq3Wjavp97DarblDQ09okYgygbSQMkoitg89e1
e8 + DfFUluixTkKExtxxXzFS M1PU /
QsvxMJw5UfSkMq32nxyR3LMuP3hY5l6 +
grF1HWrGw8zypAGAyWl6jtXJ + GfiLAbcW0kwljC / NGDgVm + L /
ABb9ohAtH2jaBgCs + ZHpRjEf4 /
8Aio9nB5VvOrORtWNRk1jeCtQ1nXJDJd2jxgtgEjrXOa7
pGr2kEfi27095IkYswVSSF9cVwGp / tAftH694vTS / hd8L tLbw3Z4 +
3a1fa2lJtjqI4dpZz9cVly8xqmoX0Pc fil4S1FNGliMJJ +
z5lNfLpIjwv4h024bxXpt8C1jPuMHYqDyK7T4p /
tHa6vvh7a2juJ9RkTYsIDM2ewJ + te H + D7r9pqW81B /
Gt5ZXdpfgiy0uysWR4M93kY8 / IXTR92J5WJlzzPrP4K /
ECw8deG7a7guNu5cOiniGH
UGu81LUobK2a4W5O1FOVavCv2ZvC1z4Q0BoNekWOWSRnZQ3Qk5
rsfGnjKGws5FnusoAcAtyal / GS pXp3Zw3x1 + IEdnBMyPuXJC4P0r4w
+ OOo3niCww9VID22nRNcSEtwCq5X8c17H8bfG8 + p3rWdox2 + Y
ec9q + efjhdePR8O7rw74W0e1ultduFtA4ucTGRmVQu30JbOfY /
j6uEp63PmMxrQlI4L9mHwo + keDx4 luHMBajNcTgOOeCF /
UZNFUH7bOrpYfsKelbQy83uu + Hgu4EksL2N8 / kp / WvOfCHwj + O +
i + FLPR4vhZ pkkdrElo5Jb / acjuR71o / tk6x49tP2H7yy + IOhW1lfz +
M9JihtJi6 + XHIZAFr8ta4mUvZfZQ4 / wBpUEns / w
BT40tvhv48u7eO7t9Fu2jIQPGyxZBBGQaK9e8M +
LPhTbeG9Pt5viiYnjsYlaJicoQgBX8OIFfLPFTvsfrLx
U7n6rWtrZKqsqkjHPJ4rzD4u + ALHS /
GNn8XbKlPalljqUmSAbVnDROfaOU4PoszH + EV6Jb6kAMLzTb
5bXUlpbG5iDRTQ7XjPRsjBH5cV91yo /
nvY43xbeafHBojWMU1zc3JeOKMW0kaW0 + 6FnVpHQJIGg +
ddrZXZznJrt9OtTe6RHfWkjbgmSDXOeDtQvPCc0nhy / kLRoxWzmf +

KHPyqT321q / D / xVY2usXHha6u0 aa3fOzP3IP / 668bMqNoqcT6bh /
F2qunM2tF8XTWH7iY8jg4GK6PStZtdQkWW7f92pyVJHNYmo6TYS 3g
kjiwGb5T2rB + Kya /
4E8NzazoVjJdySf6q3gGSc14nMfaxkj2aDxho1zAsLsoVV2 /
Me1Yt7calZn + w2ca hsgGNA2etfKPiH4 + / H / wFp4n1P4c2MkcygxS
/ 2mTj2cbAA3rzj61y + t / tE / ti + KdLkudF8Ex21sF / 12mb
XyPqx4q6dJ1Njoi5ch9a3GI +
E4LO9vVWJblBu2uolrgfEHjOCC2Itlo0cNy + Ov0r5X8U / Gr9rbxDon /
COp ot7HOSRPLBEqOy + 79BXkHiXx5 + 0bpuoNoE / iue1ckfuPtKzOT
+ HSuinRcTz8VzRPuqz + LdumpDS / tADZ
JOMDmsjxRrniPXr06hPfYtYrZgLdovvNkfNn6AivHv2SPHT4213V4fGXx
K8Y3ly4 + 9auqqmPoOa9r + LE2 haTEun6e4VNmGAPQVcY ++ eVUm
/ Znj3iRbe3tb3WtTIAEcZbDcYUc5 + teafsD + O9W / ab8faofFuhwP
p3gjVjNpVyIARszM + 2NyAC21o2YEk4D1T / bH +
MsXgwwNLoPh6ZJb27 / AHaqz4Kkgkm / KoP + CXnirVf DOj +
J5dWtbe3sby5hWPahLmRATw3QjDHI9cV7eCjofLZhK590ulo4YwVB2s
eQOfNP / BUyxuNS / Zke axdA9p4m0 + YmT + 786n / ANDH5V7Zc /
EHTZMLpQW6At0rbw / AIKJ68l / + zPqAjID / wDE0suh / wCmtP
FpLDs14cd83o + v6H5 / Q6Bozwo8niVUYqCyfZGO0 +
mc80VqWvhaK4tY5ybnLxhjgLjkUV8j7Q / ZrRP2B
s9ZneIFXx9DUv8Aa9wsXmDJKcqM9awtPmf5BWIA5MO4kZr7o /
nzmR5X + 0f + 0hrfwYGjpc / DafWT4g 1f +
z9lj0y7DTSS7GkQbSOMhSOM1876z / wUst /
AvxJuk8TfDLWdL1LTHNtqlrM372OTfnDLt67e3tXpf / BS7xDb +
Cvh94G8cSrOTpvxDtJUe1dVYE2 10AdzK2Pm2knaeAcDNfAXxQJ8Ve
PviB4p8X + Lgk8E0LCX
VbQx3WozSg7RHFtUrIvZjuxsUruySKznQhVhZm + GqzpT5kfsH8C / 2l
/ BHx58BWXirwhraXKTQjAGQy N6Nx8rD3r1fwN4hg19fJ1NvNFu +
HjmA49xX4xf8E8vi34o0j4z6homj6wbaTVbNri3swV8uSSNhuXav
AJU9R6V + iXgz4 + ahoesQS6zbvaSHCzQyrwRn74PTFFk4rD /
AFeryn6LlmlWKwybPqLxT8PvD / iWzkhF
ukqSjDxyDXkHij9nNdHmlm8H3l5YIs5hjkzG3 / Aa9Z +
HXjC08Q2cV3HcKYplGTuGPwruDpmkyWZZ13 D1BrnpucT2o1LRPjS8
+ Bnj25RreW + nEcgO9hGAf51T8PfsvaJpN + 11dacZLmRgZpZTk /
melfWWvW2k
QQSSxwbiCQoLV55461Wy0TT5Lh5FjyCRzXVTnKUjzcVLmjc8 /
v103wJo4s7JURwTuCnoMV87fH79oH

RvCVlqHiTVtRCWtjES +
DncewHqTW38dvjNJE8mnaZcPLPJKBwelfGH7aerahYfDZVvp383ULxI2XswLb j +
gNd + HpxINI + YxVaUaUmcP8AFf4paf8AFvxHp3iz + Gyj2
+ /
DfU9Bi8GeF9QNtqcCkXFicMDJI38TKRw69eRzXyP4P8b2Q8PJQNTIdJPtyPbXDH
PIRsrLKvP8JG3g +
+ OwOh8MtJk8E6uPFkV0LfUbO5caZJlh2LtyA3qc + hr3VGFLQ + a /
eVep + ldx4ucToRMPbmvMP2wvGF xe / APULXzC6NqdpvHp + 8GK8r
/ Zp + LxXJ8eeLp9K8beKEvY4rUn5LVYwTntiut /
awuHtPhDe2sm7Ei7ak bTwP3ormxK5qMkenkXuZtT9T55ttSjFtGG0u +
JCDJWMEHjtxRWTH4j1qKNYo9WuQqqAoEh4FFfNfU
ZPofrDxSTP13tPHfhZwsa + I7UH1E4res / Efh9ogRrMB3cgiUV +
fviP9tj9n7wnayTaR4vvNeul1xHFplk21 3 / 66Oqoo + p / OvM / E /
wDwVQ + NEI2LPwJ4N0TTIRwq3zyXb4HUsQYwPwWvriKbPw20T7R /

4KTfEDw z4M + FHgzx7qH +
mQ6D8UdJv7i1t2BeeONLkuFzXk8AfgT0r80 / jj8YdX +
OHxe134razpsFlca5fGX7LBUk xKAqqgJ6kKq5Pc0744 / tLFB9oXWldU
+ luuJLfaLstNPtY / Ktbc4w5VT1Y8ksx9F6VxNvIliZBXTGPulR90
3vAXjjU / h3460jx7o8xE2lahHOMcb1z86 /
QqWr9mPgvP3g39oH4X6X4nigjlt9StEmT / ZJUHAPXIPevx
OjRplwCwwDX7zf8EGovgd + 0J +
y5oPh3WPDrlQ2gAWtw1vfTJJIsWcHaGIZMEA4HOM5zmvJzHAPeR
mie1IWarBT5XsL4P8FfEH4UJ5WgzvqGnB / wB3HMCHj9geh /
Gu10n43apaqbPVbG4gc9VdOPzFfXUv
7HHwsl1SK4W98RskitE0P9plkf3sggeSSTj5fvYGAeuazdb /
AGBvh1qGrKbHUuUQlrGW3lu1dX46HC5 Uj2615EcvrRPe /
t7Dv4UfKWrfFZbiJkEjE4JxtNeJfGn4ja3fWj2cEUjF8hNvavvgfsA +
ELRT9r8R67PHOTG UjMO + JsdVJj / LIPvmlv8A / gnj8B5NN +
x69od 5ex3CbJJLu5cSKCcgq8Wwow / KumjgJnBiM5ptWTPyh0
X4dXXifWjNqcgYlsthc459P0xXHfGn / gnf8Y /
2sdf07w94a0290fSNPSS7bVL + yKQSAgqqgsRubJ6KcgZO O9fsT4T /
AGE / 2c / AEv8AbHhbwZFNciNUInupZZThCSrfO7APnHKgE4q /
408P2ukQvFaxIsUpIACnBHG
QQa9HD4OVKfMzw8Tj3WjyxP5Xfir8NfFPwU +
JOtfC7xlbmLVNBv3tLyPllwd3HG1gQQR2Oaq3HiH
W5YEiRzLtG3L8NtHf8K + s / 8AguT4Ut9K / bTu9a0 /
SoYEudDtGungXBmkUyLvOOpXgZ9AB2r5Bsg7x4U
gEdTnNelGKq14U500e6fsn +
JbDSfGt1f6nqoULaqtuWXGVLfMfyr2X9pDxPpfif4P30 + m3lkWG7t88d9
4r4w0rxHq + h3v23S71kIGQ3GQwz0NdZJ +
0L4zvPCI34O1GCzmtLyWN5pShEilXB42sAK56tJuNjvyzEx
oYyNWX2Tbg + J8lhClj / Y2nt5KiPc8PJwMZPPXiiuB / wCEsd /
naNsnk4cY / wDQaK5 / YrsfvvP8O3e5iGQM vmNk46Z7VGHEyz /
Fj5fQUSMwRgGONrd / 96lkVdnQfd / + Kr1 /
ZpHwBPaglfM56YqVCWATPWoySREw B / gFS2vJQn / PNFfi / sFy2A7
/ zmvWP2IXdP2h9MVHIBtpwQD1 + 7WkUjnkf1b /
Cz4i6Z478Mwa1pl4SwiR44liyMn26g / SutismvT9p
v4vKLcrGAwPHqM18y / sB3Fw / wi0rfO5zbDqx / uivpjw9JJOvVp3Lln +
Yuc561jKIHmLhUlykF / ZTRS7 / SO 5deGV + SFQ /
Sql5pNy10kk5Q7CW2yHp2rqy7vbXhdidoBXJ6c1h3xK6nbBTgG2kJA74
XilGnG43J8xzur aOunROZt21 / nUxjd + Tey / EjS5r / AE5HtVISIPkk /
ebj9a9n113bT4omYIWjfKk8GvLvH6rH4Nd41CkXnBA wR8orVJchnHc /
Dz / gtv8ACx5PjnZ63KGMNxoqQtKZwwduf1r889f +
HHiPwvaf2zLan7DK4AnHRSSRtl / D9a / WH /
gsqqzahZzzKHf7l3zsMn7w718E / EqGFvgjqLNEpl2EEr0O4c15 +
EqynOcX0Z7 + NowhhKM11
R88SIFy5HBPFRPgnKjp0zV2RExjaOh7VST5gd3PHeu /
2akzyb2GELnkj / vmioyzAnDH86Kz5UZn / 9k = "

V 1.0.0 version

HTTP protocol

management black

and white

list JSON format to

obtain black and

white list library list

Request parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoREQ"	20	String	
UUID	Device UUID	32	String	Wide area Required for internet
TimeStamp	Unix timestamp		Integer	
Sign	Permission verification information	Fixed 32 characters	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
Action	Fixed "GetList", Get a list of black and white lists		String	

Response parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoACK"	20	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	

Action	Fixed "GetList", Get a list of black and white lists		String	
GetList	List parameters obtained Contains List		Json object	
List	To get the picture list, the package contains LibIndex , FliIndex , Json object composed of Pic		Json array	
LibIndex	Library index	0 ~ 6	Integer	
FliIndex	Picture index	0 ~ 3199	Integer	
Pic	Picture name (fixed .jpg suffix)	64	String	

S

e

n

d

i

n

J

s

o

n

f

o

r

m

a

t

:

{

```
"Name":  
"WBLis  
tInfoRE  
Q",  
"TimeS  
tamp":  
154259  
4185,  
"Sign":  
" 873c7f21d3dc7978aeec7c8c  
29711c0 ", "Mode": 1,  
"Action": "GetList"
```

}

Returns :

{

```
"Name":  
"WBLisl  
nfoACK",  
"Mode":  
1,  
"Actio  
n":  
"GetLi  
st",  
"GetLi  
st": {
```

"List":

[

{

```
"LibIndex": 0,  
"FlieIndex": 0,  
"Pic": "chenzhen_12354.jpg"
```

},

{

```
"LibIndex": 0,  
"FlieIndex": 1,  
"Pic": "chenzhen_12355.jpg"
```

```

    }
  ]
},
"Result": "ok"
}

```

Get a list picture

Request parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoREQ"	20	String	
UUID	Device UUID	32	String	Wide area Required for internet
TimeStamp	Unix timestamp		Integer	
Sign	Permission verification information	Fixed 32 characters	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
Action	Fixed "GetOnePic", Get a picture		String	
GetOnePic	Get the parameters needed for the picture Package comprising LibIndex , FliIndex , Pic		Json object	
LibIndex	Library index	0 ~ 6	Integer	
FliIndex	Picture index	0 ~ 3199	Integer	
Pic	Picture name (fixed .jpg suffix)	64	String	

Response parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoACK"	20	String	

Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
Action	Fixed "GetOnePic", Get a picture		String	
GetOne Pic	Image parameters obtained Package comprising LibIndex , FliIndex , Pic , Img		Json object	
LibIndex	Library index	0 ~ 6	Integer	
FliIndex	Picture index	0 ~ 3199	Integer	
Pic	Picture name (fixed .jpg suffix)	64	String	
Img	Picture data (base64 encoding)	2M	String	

json format

Send :

```
{
  "Name":
  "WBLis
  tInfoRE
  Q",
  "TimeStamp":
  154259
  4185,
  "Sign":
  " 873c7f21d3dc7978aeec7c8c
  29711c0 ", "Mode": 1,
  "Action":
  "GetOnePic",
  "GetOnePic":
  {
    "LibIndex": 0,
    "FliIndex": 0,
    "Pic": "chenzhen_12354.jpg"
  }
}
```

```
}

```

Returns :

```
{
  "Name":
  "WBListI
  nfoACK",
  "Mode":
  1,
  "Action":
  "GetOne
  Pic",
  "GetOne
  Pic": {
    "LibIndex": 0,
    "FlieIndex": 0,
    "Pic": "chenzhen_12354.jpg",
    "Img": "/ 9j / ... ( picture data base64 encoding , omitted ) "
  },
  "Result": "ok"
}
```

Delete a picture from the list library

Request parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoREQ"	20	String	
UUID	Device UUID	32	String	Wide area Required for internet
TimeStamp	Unix timestamp		Integer	
Sign	Permission verification information	Fixed 32 characters	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
Action	Fixed "DeleteOnePic", Delete a picture		String	

DeleteOnePic	Get the parameters needed for the picture Package comprising LibIndex , FliIndex , Pic		Json object	
LibIndex	Library index	0 ~ 6	Integer	
FliIndex	Picture index	0 ~ 3199	Integer	
Pic	Picture name (fixed .jpg suffix)	64	String	

Response parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoACK"	20	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
Action	Fixed "DeleteOnePic", Delete a picture		String	
DeleteOnePic	Deleted picture parameters Package comprising LibIndex , FliIndex , Pic		Json object	
LibIndex	Library index	0 ~ 6	Integer	
FliIndex	Picture index	0 ~ 3199	Integer	
Pic	Picture name (fixed .jpg suffix)	64	String	

S

e

n

d

i

n

J

s

o

n

f

o

r

m

a

t

:

{

```
"Name":  
"WBLis  
tInfoRE  
Q",  
"TimeS  
tamp":  
154259  
4185,  
"Sign":  
" 873c7f21d3dc7978aeec7c8c  
29711c0 ", "Mode": 1,  
"Actio  
n":  
"Dele  
teOn  
ePic",  
"Dele  
teOn  
ePic":  
{  
  "LibIndex": 0,  
  "FlieIndex": 0,  
  "Pic": "chenzhen_12354.jpg"  
}
```

}

Returns :

{

```

    "Name":
    "WBListl
    nfoACK",
    "Mode":
    1,
    "Actio
    n":
    "Delet
    eOne
    Pic",
    "Delet
    eOne
    Pic": {

        "LibIndex": 0,
        "FliIndex": 0,
        "Pic": "chenzhen_12354.jpg"
    },
    "Result": "ok"
}

```

Add a list library picture

Request parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoREQ"	20	String	
UUID	Device UUID	32	String	Wide area Required for internet
TimeStamp	Unix timestamp		Integer	
Sign	Permission verification information	Fixed 32 characters	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
Action	Fixed "AddOnePic", Add a picture		String	

AddOne Pic	Parameters needed to add pictures Package comprising imgName , ImgNum , Img		Json object	
ImgName	Picture naming (custom)	28 characters	String	
ImgNum	Picture number (custom)	28 characters	String	
Img	Picture data (base64 encoding)	2M and (1080 * 1080) jpg format	String	

Response parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "WBListInfoACK"	20	String	
Mode	List category 1 : Blacklist 2 : Whitelist 3 : VIP list	1 ~ 3	Integer	
AddOne Pic	Fixed "AddOnePic", Add a picture		String	
AddOne Pic	Added picture parameters Package comprising LibIndex , FliIndex , Pic		Json object	
LibIndex	Library index	0 ~ 6	Integer	
FliIndex	Picture index	0 ~ 3199	Integer	
Pic	Picture name composition rules : ImgName + ImgNum.jpg	64	String	

json format

Send :

```
{
  "Name":
  "WBLis
  tInfoRE
  Q",
  "TimeS
  tamp":
  154259
  4185,
```

```
"Sign":  
" 873c7f21d3dc7978aeec7c8c  
29711c0 ", "Mode": 1,  
"Ac  
tion  
":  
"Ad  
dO  
ne  
Pic  
",  
"Ad  
dO  
ne  
Pic  
":  
{  
    "Img  
Nam  
e":  
    "che  
nzhe  
n",  
    "Img  
Num  
":  
    "123  
54",  
    "Img": "/ 9j / ... ( picture data base64 encoding , omitted ) "  
}  
}
```

Returns :

```
{  
    "Name":  
    "WBListl  
nfoACK",  
    "Mode":  
    1,  
    "Actio  
n":  
    "Dele  
teOn  
ePic",  
    "Dele  
teOn  
ePic":  
    {  
        "LibIndex": 0,  
        "FlieIndex": 0,  
        "Pic": "chenzhen_12354.jpg"  
    }  
}
```

```

    },
    "Result": "ok"
}
    
```

error code

Add black and white list error code	meaning
0	other
-1	failure
-2	Failed to extract feature value
-3	File name number duplicate
-4	Kuman
-5	Add timeout
-6	Parameter error
-7	file size too big
-8	not enough storage space
-9	File failed to open

-10	No database
-11	Image reading failed
-12	Database file is damaged
-13	Poor picture quality
-14	Image size is wrong, width and height cannot be odd
-15	Face detection failed (no face detected or multiple faces detected)
-16	Picture format error
-17	Face area error
-18	Algorithm creation handle error
-19	Equipment busy

2. O

btain

devi

ce

infor

mati

onR

eque

st

para

mete

rs

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	fixed	Y	Fixed 18	String	

	"deviceInfoResponse"				
Session	Register to return to Session	N		String	
Data	Request content	Y		Json to Elephant	
DeviceId	device serial number	Y	<64	String	
DeviceUUID	Device UUID	Y	= 12	String	
DeviceMac	Device Mac address	Y	<64	String	
DeviceIP	Device IP address	Y	<64	String	
WebVersion	Page version	Y	<128	String	
CoreVersion	Main program version	Y	<128	String	
VersionDate	Version date. format: Nov 14 2018 18:02:07	Y	<64	String	
HTTPVersion	HTTP protocol version	Y		String	
HTTPDate	HTTP protocol version date	Y		String	
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

Send :

```
{
  "Name": "deviceInfoRequest",

  "UUID":
  "abcdefabcdef", "Session":
  " abcdefabcdef _ 15425941
85 ", "TimeStamp":
  1542594185, "Sign":
  " ec304c35342cbe0609e7c7fec
4d97ca5 "
}
```

Returns :

```
{
  "Name":
  "deviceInfoResponse"
  , "Session":
  " abcdefabcdef _ 1542
594185 ", "Data":
  {
    "DeviceId": "10006612",
    "DeviceUUID":
    "umksbhtsv6as",
    "DeviceMac":
    "00-11-04-00-19-
e3",
    "DeviceIP": "128.128.20.84",
    "WebVersion":
    "Face_9.20_FH4",
    "CoreVersion":
    "15.13.1.13",
    "VersionDate":
    "Nov 14 2018
18:02:07",
    "HTTPVersion": " 1.1.0",
    "HTTPDate": "2019-04-22"
  }
  "Code": 1,
  "Message": "Succeed."
}
```

V 1.0.0 version

HTTP protocol to obtain device information in JSON format (old)

Request parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "DeviceInfoREQ"	20	String	
UUID	Device UUID	32	String	WAN time need
ChannelNo	Channel number (management of each NVR The IPC of the channel , the value 255 is		Integer	NVR pair Connect the required parameters

	Manage NVR)			
TimeStamp	Unix timestamp		Integer	
Sign	Permission verification information	Fixed 32 characters	String	

http response parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Fixed "DeviceInfoACK"	20	String	
DeviceInfo	Device information parameters, including DeviceId DeviceUUID DeviceMac LocalIp WebVersion CoreVersion VerDate		Json object	
DeviceId	device serial number	64	String	
DeviceUUID	Device UUID	32	String	
DeviceMac	Device Mac address	64	String	
LocalIp	Device IP address		String	
WebVersion	Page version	128	String	
CoreVersion	Main program version	128	String	
VerDate	Version date format: Nov 14 2018 18:02:07	64	String	

S

e

n

d

i

n

j

s

o

n

f

o

r

m

a

t

:

{

```

    "Name":
    "DeviceInfoREQ",
    "TimeStamp":
    154259
    4185,
    "Sign": " 873c7f21d3dc7978aeecc7c8c29711c0 ",
}

```

Returns :

```

{
    "Name":
    "DeviceInfoACK",
    "DeviceInfo": {
        "DeviceId": "10006612",
        "DeviceUUID":
        "umksbhtsv6as",
        "DeviceMac":
        "00-11-04-00-19-
        e3",
        "LocalIp": "128.128.20.84",
        "WebVersion": "Face_9.20_FH4",

        "CoreVersion":
        "15.13.1.13", "VersionDate": "Nov 14
        2018 18:02:07"
    }
}

```

3 , IO control Request parameters

				Elephant	
ContinueSeconds	IO output duration (Unit: seconds) Default 1 second	N	1-60	Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "IOControlResponse "	Y	Fixed 17	String	
ChannelNo	Channel number (manage IP C of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

Send :

```
{
  "Name":
  "IOControlRequest",
  "UUID":
  "abcdefabcdef",
  "Session": " abcdefabcdef _ 1542594185 ",
  "TimeStamp": 1542594185,
```

```
"Sign":
  " ec304c35342cbe0609e7c7fec
  4d97ca5 ", "Data":
  {
    "ContinueSeconds": 1
  }
}
```

Returns :

```
{
  "Name":
  "IOControlResponse", "Session":
  " abcdefabcdef _ 15
  42594185 ", "Code"
  : 1,
  "Message": "Succeed."
}
```

V 1.0.0 version

HTTP protocol remote open JSON format (old)

Request parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Solid fixed "remoteOpenDoorREQ"	20	String	
UUID	Device UUID	32	String	Want When needed
TimeStamp	Unix timestamp		Integer	
Sign	Permission verification information	Fixed 32 characters	String	

http response parameters

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Name	Solid fixed " remoteOpenDoorACK "	20	String	
ErrorCode	Response Code 1 : to work	4	Integer	

Result	Response message		String	
--------	------------------	--	--------	--

S

e

n

d

i

n

j

s

o

n

f

o

r

m

a

t

:

{

"Name": "remoteOpenDoorREQ",

```
"TimeStamp": 1542594185,
"Sign":
" 873c7f21d3dc7978aeec7c8c
29711c0 ",
}
```

Returns :

```
{
  "Name":
  "remoteOpen
  DoorACK",
  "ErrorCode" :
  1,
  "Resule": "ok"
}
```

4. Upload parameter acquisition Request parameters

Sign	Authority verification information, when the server parameter is set to " management authority verification " Verify when turned on.	N	Fixed 32	String	
------	--	---	----------	--------	--

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
------	------------	-----------	----------------	----------	---------

Name	Fixed "getHTTPParameters Response "	Y	Fixed 25	String	
ChannelNo	Channel number (manage NVR channels Of the I P C , level 25 . 5 for managing N V R & It)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	
Data	Back to content	Y		JsonObject Elephant	
CaptureEnabled	Upload snapshot information 0 : do not upload 1 : upload	Y	0-1	Integer	
CaptureAddress	Snapshot information push address Format : <u>http:// server / path</u>	Y	Max 128	String	Be careful not to have Space
CaptureType	Comparison upload type :	Y	0-6	Integer	

	<p>(Specific options for comparison machine) 0: upload by everyone 1: Compare and upload successfully 2 : Blacklist upload 3 : Whitelist upload 4: VIP list upload 5: Stranger upload 6: Non-whitelist upload</p>				
CaptureContent	Snap information upload content	Y		JSON pair Elephant	
FacelInfo	Upload "FacelInfo" content 0 : do not upload 1 : upload	Y		Integer	
CompareInfo	Upload "CompareInfo" content 0 : do not upload 1 : upload	Y		Integer	
PictureData	Whether to include snapshot information Corresponding picture data	Y	0-1	JSON pair Elephant	
FacePicture	Face image data 0 : not included 1 : Include				

BodyPicture	Humanoid picture data				
-------------	-----------------------	--	--	--	--

	0 : not included 1 : Include				
BackgroundPicture	Background image data 0 : not included 1 : Include				
PersonPhoto	Person photo data 0 : not included 1 : Include				
ResendTimes	Re-transmissions (retransmissions recorded identification data is returned upon receipt of incorrect Times)	N	0-5	Integer	
RegisterEnabled	Registration function 0 : do not open the registration 1 : Open registration	Y		Integer	
RegisterAddress	Registration information push address Format : <u>http: // server / path</u>	N	Max 128	String	Be careful not to have Space

HeartbeatEnabled	Upload heartbeat information 0 : do not upload 1 : upload	Y		Integer	
HeartbeatAddress	Heartbeat information push address Format : <u>http:// server / path</u>	Y	Max 128	String	Be careful not to have Space
HeartbeatInterval	Heartbeat push interval (unit: seconds)	Y	0-100000	Integer	

SignCheck	Whether to verify authority when accessing the http interface (sign value verification) 0 : do not verify sign 1 : Verify Sign	Y		Integer	
HTTPVersion	Current HTTP version 0 : 1.0.0 1 : 1.1.0 / 1.1.1 / ...	Y		Integer	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "getHTTPParameters
  Request", "UUID":
```

```
"abcdefabcdef", "Session":  
" abcdefabcdef _ 1542  
594185 ", "Timestamp":  
1542594185,  
"Sign": " ec304c35342cbe0609e7c7fec4d97ca5"  
}
```

return:

```
{  
  "Name":  
  "getHTTPParameters  
Response", "Session":  
  " abcdefabcdef _ 1542  
594185 ", "Data":  
  {  
    " CaptureEnabled ": 1,  
    " CaptureAddress ": " http:  
// server /  
path ", "CaptureType": 1,  
    "CaptureContent":  
    {
```

```
"  
F  
a  
c  
e  
l  
n  
f  
o  
"  
:  
1  
;  
"  
C  
o  
m  
p  
a  
r  
e  
l  
n  
f  
o  
"  
:  
1
```

```
},  
"PictureData":  
{  
  "FacePicture":  
    1,  
  "BodyPicture":  
    1,  
  "BackgroundPicture":  
    1,  
  "PersonPhoto": 1  
},  
"ResendTimes": 1,  
"RegisteredEnabled"
```



```

    ".
    1,
    "RegisterAddress": " http:
    // server / path ",
    " HeartbeatEnabled ": 1,
    " HeartbeatAddress ":
    " http: // server /
    path ", "HeartbeatInterval ":
    60,
    "Sign
    Chec
    k": 1,
    "HTT
    PVer
    sion":
    1
    },
    "Code": 1,
    "Message": ""
    }
    
```

5. Upload parameter settings Request parameters

	Of the IP C h e d 25 . 5 for managing N V R & It)				Required parameters
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification
Sign	Authority to verify information when server configuration " administrative rights verification " open Start verification.	N	Fixed 32	String	

Data	Request content	Y		Json to Elephant	
CaptureEnabled	Upload snapshot information 0 : do not upload 1 : upload	Y	0-1	Integer	
CaptureAddresses	Snapshot information push address Format : <u>http:// server / path</u>	Y	Max 128	String	Be careful not to With spaces
CaptureType	Comparison result upload type : (specific options for comparison machine) 0: everyone uploads 1: Compare and upload successfully 2 : Blacklist upload 3 : Whitelist upload 4: VIP list upload	Y	0-6	Integer	

	5: Stranger upload 6: Non-whitelist upload				
CaptureContent	Snap information upload content	Y		JSON pair Elephant	

FacelInfo	Upload "FacelInfo" content 0 : do not upload 1 : upload	Y		Integer	
CompareInfo	Upload "CompareInfo" content 0 : do not upload 1 : upload	Y		Integer	
PictureData	Whether to include snapshot information Corresponding picture data	Y	0-1	JSON pair Elephant	
BodyPicture	Humanoid picture data 0 : not included 1 : Include				
FacePicture	Face image data 0 : not included 1 : Include				
BackgroundPicture	Background image data 0 : not included 1 : Include				
PersonPhoto	Person photo data				

	0 : not included 1 : Include				
--	-------------------------------------	--	--	--	--

ResendTimes	Re-transmissions (retransmissions recorded identification data is returned upon receipt of incorrect Times)	N	0-5	Integer	
RegisterEnabled	Registration function 0 : do not open the registration 1 : Open registration	Y		Integer	
RegisterAddress	Registration information push address Format : <u>http:// server / path</u>	N	Max 128	String	Be careful not to With spaces
HeartbeatEnabled	Upload heartbeat information 0 : do not upload 1 : upload	Y		Integer	
HeartbeatAddresses	Heartbeat information push address Format : <u>http:// server / path</u>	Y	Max 128	String	Be careful not to With spaces
HeartbeatInterval	Heartbeat push interval (unit: seconds)	Y	0-100000	Integer	
SignCheck	Whether to verify authority when accessing the http interface (sign value verification) 0 : do not verify sign 1 : Verify Sign	Y			

HTTPVersion	HTTP version 0 : 1.0.0	Y			Change version The device will
-------------	-------------------------------	---	--	--	---------------------------------------

	1 : 1.1.0 / 1.1.1 / ...				Restart.
--	-------------------------	--	--	--	----------

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "setHTTPParameters Response "	Y	Fixed 25	String	
ChannelNo	Channel number (manage NVR channels Of the IPC , the value 25 . 5 for managing N V R & lt)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	
Data	Back to content	Y		JsonObject Elephant	

CaptureEnabled	Upload snapshot information 0 : do not upload 1 : upload	Y	0-1	Integer	
CaptureAddress	Snapshot information push address Format : <u>http:// server / path</u>	Y	Max 128	String	Be careful not to have Space
CaptureType	Comparison result upload type : (specific options for comparison machine) 0: upload by everyone	Y	0-6	Integer	

	1: Compare and upload successfully 2 : Blacklist upload 3 : Whitelist upload 4: VIP list upload 5: Stranger upload 6: Non-whitelist upload				
CaptureContent	Snap information upload content	Y		JSON pair Elephant	

FaceInfo	Upload "FaceInfo" content 0 : do not upload 1 : upload	Y		Integer	
CompareInfo	Upload "CompareInfo" content 0 : do not upload 1 : upload	Y		Integer	
PictureData	Whether to include snapshot information Corresponding picture data	Y	0-1	JSON pair Elephant	
FacePicture	Face image data 0 : not included 1 : Include				
BodyPicture	Humanoid picture data 0 : not included 1 : Include				

BackgroundPicture	Background image data 0 : not included 1 : Include				
PersonPhoto	Person photo data 0 : not included 1 : Include				

ResendTimes	Number of retransmissions (received incorrectly (Number of retransmission identification records when data is returned)	N	0-5	Integer	
RegisterEnabled	Registration function 0 : do not open the registration 1 : Open registration	Y		Integer	
RegisterAddress	Registration information push address Format : <u>http:// server / path</u>	N	Max 128	String	Be careful not to have Space
HeartbeatEnabled	Upload heartbeat information 0 : do not upload 1 : upload	Y		Integer	
HeartbeatAddress	Heartbeat information push address Format : <u>http:// server / path</u>	Y	Max 128	String	Be careful not to have Space
HeartbeatInterval	Heartbeat push interval (unit: seconds)	Y	0-100000	Integer	
SignCheck	Whether to access the http interface Authority verification (sign value verification)	Y			

	0 : Do not verify Sign 1 : Verify Sign				
HTTPVersion	Current HTTP version 0 : 1.0.0 1 : 1.1.0 / 1.1.1 / ...	Y			
Code	Return opcode 1 success 0 failed	Y	Max 8	Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "setHTTPParameters
  Request", "UUID":
  "abcdefabcdef", "Session":
  "abcdefabcdef _ 1542
  594185 ", "Timestamp":
  1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    " CaptureEnabled ": 1,
    " CaptureAddress ": " http:
    // server /
    path ", "CaptureType": 1,
    "CaptureContent":
    {
      "
      F
      a
```

```
c  
e  
l  
n  
f  
o  
"  
:  
1  
;  
"  
C  
o  
m  
p  
a  
r  
e  
l  
n  
f  
o  
"  
:  
1  
},  
"PictureData":  
{  
  
    "FacePicture": 1,  
    "BodyPicture": 1,  
    "BackgroundPicture": 1,  
    "PersonPhoto": 1  
},  
"  
R  
e  
s  
e  
n  
d  
T  
i  
m  
e  
s  
":  
0,  
"  
R  
e
```

```
g
i
s
t
e
r
E
n
a
b
l
e
d
":
0,
"RegisterAddress": " http:
// server / path ",
" HeartbeatEnabled ": 1,
" HeartbeatAddress ":
" http: // server /
path ", "HeartbeatInterval ":
60,
"Sign
Chec
k": 0,
"HTT
PVer
sion":
1
}
}

return:
{
  "Name":
  "setHTTPParameters
Response", "Session":
  " abcdefabcdef _ 1542
594185 ", "Data":
  {
    " CaptureEnabled ": 1,
    " CaptureAddress ": " http:
// server /
path ", "CaptureType": 1,
    "CaptureContent":
    {
      "
      F
      a
      c
      e
      l
```

```
n
f
o
"
:
1
;
"
C
o
m
p
a
r
e
l
n
f
o
"
:
1
},
"PictureData":
{
  "FacePicture": 1,
  "BodyPicture": 1,
  "BackgroundPicture": 1,
  "PersonPhoto": 1
},
"
R
e
s
e
n
d
T
i
m
e
s
":
0,
"
R
e
g
i
s
t
e
r
```

```

E
n
a
b
l
e
d
":
0,
"RegisterAddress": " http:
// server / path ",
" HeartbeatEnabled ": 1,
" HeartbeatAddress ":
" http: // server /
path ", "HeartbeatInterval ":
60,
"SignCheck": 0,

"HTTPVersion": 1
},
"Code": 1,
"Message": ""
}
    
```

6. Time synchronization Request parameters

TimeMode	Time synchronization mode 0 : NTP school time 1 : Manual time adjustment	N		Integer	(Default NTP when no parameter is passed School time)
LocalTime	Manual time correction time format : yyyy-mm-dd hh: mm: ss	N		String	Required parameters for manual calibration

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "timeSynchronizationResponse"	Y	Fixed 27	String	
Channel No	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
Data	Back to content	N		Json object	
LocalTime	Current time format : yyyy-mm-dd hh: mm: ss	N		String	

Code	Return opcode 1 success	Y		Integer	
------	--------------------------------	---	--	---------	--

Message	Return operation information	Y	Max 200	String	
---------	------------------------------	---	---------	--------	--

JSON format example

Send :

```
{
  "Name":
  "timeSynchronization
  Request", "UUID":
  "abcdefabcdef", "Session":
  " abcdefabcdef _ 1542
  594185 ", "Timestamp":
  " 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec
  4d97ca5 ", "Data":
  {
    "TimeMode": 1,
    "LocalTime": "2019-07-30 17:24:00"
  }
}
```

Returns :

```
{
  "Name":
  "timeSynchronization
  Response", "Session":
  " abcdefabcdef _ 1542
  594185 ", "Data":
  {
    "LocalTime": "2019-07-30 17:24:00"
  },
  "Code": 1,
  "Message": "Succeed."
}
```

7. Get face feature values

Import a face picture to extract face feature values

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Solid fixed "faceFeatureValueRequest"	Y	Fixed 23	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
Channel No	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification

Sign	Permission verification information, when the server Verification when the parameter setting " Management Authority Verification " is enabled	N	Fixed 32	String	
Data	Request content	Y		Json object	
FacePicture	Face image data (base64 edit Code Format) FIG sheet format JPG	Y	<2M <960 * 960	String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	fixed "faceFeatureValueResponse "	Y	Fixed 24	String	
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection

Session	Register to return to Session	N		String	
Data	Back to content	Y		JsonObject Elephant	
FeatureType	Characteristic value type 0 : float 1 : char 2 : int 3 : General type	Y	0-3	Integer	Parsing eigenvalue data according to type
FeatureValue	Eigenvalue data (base64 encoding)	Y	float (128) char (512) int (128)	String	
Code	Return opcode	Y		Integer	

Message	Return operation information	Y	Max 200	String	
---------	------------------------------	---	---------	--------	--

JSON format example

send:

```
{
  "Name":
  "faceFeatureValueRequest", "UUID":
  "abcdefabcdef", "Session":
  "abcdefabcdef _ 1542"
```

```

594185 ", "TimeStam
p": 1542594185,
"Sign":
" ec304c35342cbe0609e7c7fec4
d97ca5", "Data":
{
    "FacePicture": "... base64"
}
}

return:
{
    "Name":
"faceFeatureValueRe
sponse", "Session":
" abcdefabcdef _ 1542
594185 ", "Data":
{
    " Feature
Type": 0,
"Feature
Value":
"...
base64"
},
"Code": 1,
"Message": ""
}

```

8. Face search list library

Import a face image and search the list library personnel.

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "faceSearchRequest"	Y		String	

UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification
Sign	Permission verification information, when the server Verification when the parameter setting " Management Authority Verification " is enabled	N	Fixed 32	String	

Data	Request content	Y		Json object	
PersonType	List type: 0 : all lists 1 : Blacklist 2 : Whitelist	N		Integer	

	3 : VIP list				
FacePicture	Face image data (base64 edit Code Format) FIG sheet format JPG	Y	<2M <960 * 960	String	
GetPhoto	Whether to obtain the picture data of the list library personnel 0 : do not return pictures 1 : Return to the picture	N	0-1	Integer	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	fixed "faceSearchResponse"	Y		String	
ChannelNo	Channel number (manage I PC	N		Integer	Required parameters for NVR

	of NVR channels , value 255 is management NVR)				connection
Session	Register to return to Session	N		String	
Data	Back to content	Y		Json to Elephant	
PersonType	List type: 0 : Stranger	Y		Integer	Stranger said Unsuccessful comparison

	1 : Blacklist 2 : Whitelist 3 : VIP list				
Similarity	Similarity Format: 0.1000 (floating point)	N		String	Compare success Some informatio n content
PersonInfo	Personnel information	N		JSON pair Elephant	Compare success Some informatio n content

PersonId	Person ID	N	<36	String	
PersonName	Personnel name	N	<64	String	
Sex	gender 1 : Male 2 : Female 0 : Unknown	N	0-2	Integer	
Sex	gender 1 : Male 2 : Female	N	1-2	Integer	
IDCard	ID number	N	<32	String	
Nation	Nation	N	<12	String	
Birthday	birthday	N	<24	String	
Phone	telephone number	N	<16	String	
Address	address	N	<100	String	
SaveTime	Add time	N	<20	String	

LimitTime	Personnel effective time limit 0 : permanently valid 1 : Cycle valid	N	0-1	Integer	
StartTime	Person effective start time format : yyyy-mm-dd hh: mm: ss	N	<20	String	

EndTime	Person valid end time format : yyyy-mm-dd hh: mm: ss	N	<20	String	
Label	Personnel tags	N	<180	String	
PersonExtension	Extended personnel information (customized warranty Saved content)	N		JSON pair Elephant	
PersonCode1	Custom code 1	N	<32	String	
PersonCode2	Custom code 2	N	<32	String	
PersonCode3	Custom code 3	N	<32	String	
PersonReserveName	Custom alternate name	N	<32	String	
PersonParam1	Custom parameter 1	N		Integer	
PersonParam2	Custom parameter 2	N		Integer	
PersonParam3	Custom parameter 3	N		Integer	
PersonParam4	Custom parameter 4	N		Integer	
PersonParam5	Custom parameter 5	N		Integer	

PersonData1	Custom information 1	N	<32	String	
PersonData2	Custom information 2	N	<24	String	
PersonData3	Custom information 3	N	<32	String	
PersonData4	Custom information 4	N	<36	String	

PersonData5	Custom information 5	N	<88	String	
PersonPhoto	Staff photos (base64 encoding)	N	<2M <960 * 960	String	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "faceSearchRequest",
  "UUID":
  "abcdefabcdef",
  "Session": " abcdefabcdef _ 1542594185 ",
  "TimeStamp": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    " Person
    Type ":
    1, "Face
    Picture"
    : "...
    base64"
    , "GetP
    hoto": 1

  }
}
```

return:

```
{
  "Name":
  "faceSearchResponse", "Session":
```

" abcdefabcdef _ 1542
594185 ", "Data":

```
{
  " PersonType ": 1,
  "Similarity": "0.912",
  " PersonInfo ":
  {
    " PersonId":
    "PersonId1",
    "PersonName":
    "PersonName1",
    " Sex ": 1,
    " IDCard ": " ",
    " Nation ": " ",
    " Birthday": "",
    "Phone": " ",
    "Address": " ",
    "SaveTime": "2019-01-01 00:00:00",
    "LimitTime": 1,
    "StartTime": "2019-01-01 00:00:00",
    "EndTime": "2020-01-01 00:00:00",
    "
    L
    a
    b
    e
    l
    "
    :
    "
    "
    ,
    "
    P
    e
    r
    s
    o
    n
    E
    x
    t
    e
    n
    s
    i
    o
    n
  }
}
```

```

"
:
{
    "PersonCode1": "",
    "PersonCode2": "",
    "Person
Code3":
    "", "Per
sonRes
erveNa
me":
    "", "Per
sonPar
am1": 0,
    "Person
Param2
": 0,
    "Person
Param3
": 0,
    "Person
Param4
": 0,
    "Person
Param5
": 0,
    "Person
Data1":
    "",
    "PersonData2": "",
    "PersonData3": "",
    "PersonData4": "",
    "PersonData5": ""
},
    "PersonPhoto": "... (base64) ..."
}
},
"Code": 1,
"Message": ""
}

```

9. Obtaining face recognition parameters

For detailed parameter definitions, please refer to the product manual.

Request parameters

name	definition	essential	Length / Range	Types of	Remarks
Name	Solid fixed "getFaceParametersRequest"	Y	Fixed 24	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
ChannelNo	Channel number (manage IPC of NVR channels, value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Require verification
Sign	Authority verification information, when the server parameter is set to "management authority verification" Verify when turned on.	N	Fixed 32	String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "getFaceParametersResponse"	Y	Fixed 25	String	
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
Data	Back to content	Y		JsonObject Elephant	
FaceEnabled	Face recognition function switch 0 : close 1 : open	Y	0-1	Integer	
Sensitivity	Face sensitivity	Y	0-10	Integer	

CaptureMode	Snap mode 0: Snapshot after leaving (distance selection) 1 : Quick snapshot 2: Interval snapshot (in seconds) 3: Interval snapshot (in frames)	Y	0-12	Integer	
-------------	--	---	------	---------	--

	4 : Single player mode 5: Capture after leaving (quality selection) 6 : Quick + snapshot after leaving 12 : Full-grab mode				
CaptureModeParam	Required parameters in each snapshot mode (Corresponding to the parameters required for the snap mode, some modes do not require this parameter)	N		JSON pair Elephant	

MaxCaptureTimes	<p>Maximum number of snapshots</p> <p>Note: "Snapshot after leaving" mode</p> <p>(CaptureMode = 0 / . 5) specific parameter</p> <p>s, the parameters to another mode without effect.</p>	N	1-3	Integer
CaptureTimes	<p>Number of snapshots</p> <p>Note: "Interval snapshot", "Single mode"</p> <p>(C A P T U R E M O D E = 2 / . 3 / . 4) specific parameter number, invalid parameter other modes.</p>	N	<p>0-20</p> <p>(0: stands for "who continued to capture")</p>	Integer
Capture Interval	<p>Capture interval (unit: second / frame) Note: "spacer capture", "single mode" (C A P T U R E M O D E = 2 / . 3 / . 4) specific parameters,</p>	N	<p>1-30 seconds (in seconds between unit spacer capture) 10-1500 frames</p> <p>(As a single frame b i t</p>	Integer

	<p>other modes invalid</p> <p>parameter. When setting "grabbing in seconds</p>		interval grip		
--	--	--	---------------	--	--

	<p>When shooting, the unit of this parameter is second. The parameter unit is frame.</p>		<p>Beat)</p> <p>1- 1500 frames</p> <p>(Single player mode)</p>		
MaxFaceSize	Maximum pixels for face recognition	Y	300-500	Integer	
MinFaceSize	Face recognition minimum pixels	Y	30-300	Integer	
SceneMode	<p>Face recognition scene mode</p> <p>0 : normal scene</p> <p>1 : Lobby scene</p>	Y	0-1	Integer	
FaceTrackEnabled	<p>Face tracking (display tracking frame)</p> <p>0 : close</p> <p>1 : open</p>	Y	0-1	Integer	

FTPUUploadEnabled	FTP upload face capture 0 : close 1 : open	Y	0-1	Integer	
PictureMode	Upload picture mode 0 : face image 1 : Original scene 2 : Face and original picture	Y	0-2	Integer	
FacePictureQuality	Face image quality	Y	1-99	Integer	
ScenePictureQuality	Scene picture quality	Y	1-99	Integer	

FaceAttributeEnabled	Face attribute detection 0 : close 1 : open	Y	0-1	Integer	
TimeTable	Arming period (Only supports 2 time periods)	Y	<= 2	JSON number group	
TimeEnabled	Time period enabled 0 : not enabled 1 : enable	Y	0-1	Integer	
TimeBegin	Starting time Format: hh: mm: ss (Accurate to minutes)	Y		String	

TimeEnd	End Time Format: hh: mm: ss (Accurate to minutes)	Y		String	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "getFaceParameters
  Request", "UUID":
  "abcdefabcdef", "Session":
  " abcdefabcdef _ 1542
  594185 ",

  "TimeStamp": 1542594185,
  "Sign": " ec304c35342cbe0609e7c7fec4d97ca5"
}
```

return:

```
{
  "Name": "getFaceP
  arametersRespon
  se", "Session":
  " abcdefabcdef _ 15
  42594185 ", "Data" :
  {
    " Fa
    ceE
    nabl
    ed ":
    1, "S
    ensit
    ivity":
    8,
    "Ca
```

```
ptur
eMo
de":
2,
"Ca
ptur
eMo
deP
ara
m":
{
    "M
ax
Ca
ptu
re
Ti
me
s":
1,
"C
apt
ur
eTi
me
s":
0,
"C
apt
ur
e I
nte
rva
l ":
10
},
"MaxFaceSize": 300,
"MinFaceSiz
e": 30,
"SceneMod
e": 0,
" FaceTrack
Enabled ": 1,
" FTPUpload
Enabled ":
0, "PictureM
ode ": 2,
" FacePictur
eQuality ":
99,
" ScenePictu
reQuality ":
99,
```

```
" FaceAttributeEnabled": 1,
" TimeTable": [
  {
    " TimeEnabled": 1,
    " TimeBegin": "00:00:00",
    " TimeEnd": "23:59:00"
  },
  {
    " TimeEnabled": 1,
    " TimeBegin": "00:00:00",
    " TimeEnd": "23:59:00"
  }
]
},
"Code": 1,
"Message": ""
}
```

10. Face recognition parameter setting Request parameters

FaceEnabled	Face recognition function switch 0 : close 1 : open	N	0-1	Integer	
Sensitivity	Face sensitivity	N	0-10	Integer	
CaptureMode	Snap mode 0 : Snapshot after leaving (distance selection) 1 : Quick snapshot 2 : Interval snapshot (in seconds) 3 : Interval snapshot (in frames) 4 : Single player mode 5 : Capture after leaving (quality selection) 6 : Quick + snapshot after leaving 12 : Full-grab mode	N	0-12	Integer	
CaptureModeParam	Required parameters in each snapshot mode (Corresponding to the parameters required for the snap mode, some modes do not require this parameter)	N		JSON object	

MaxCaptureTimes	<p>Maximum number of snapshots</p> <p>Note: "Snapshot after leaving" mode</p> <p>(CaptureMode = 0 / . 5) unique reference number, invalid parameter other modes.</p>	N	1-3	Integer	
CaptureTimes	<p>Number of snapshots</p> <p>Note: "Interval snapshot", "Single player mode"</p>	N	0-20 (0 : representative	Integer	

	<p>(CaptureMode = 2/3/4) unique parameters,</p> <p>This parameter is invalid in other modes.</p>		"Continue to catch Stop)		
--	--	--	---------------------------------	--	--

Capture Interval	<p>Capture interval (unit: seconds / frame)</p> <p>Note: "Interval snapshot", "Single mode"</p> <p>(CaptureMode = 2 /. 3 /. 4) specific parameter number, invalid parameter other modes. When "Snapshot in seconds" is set, the parameter unit is seconds.</p> <p>When "Snapshot in frames" or "Single mode" is set, the parameter unit is frames.</p>	N	<p>1-30 seconds intervals</p> <p>(grip beat) 10-1500</p> <p>Face Snap shots at intervals of frames) 1-1500 frames</p> <p>(Single mode type)</p>	Integer
MaxFaceSize	Maximum pixels for face recognition	N	300-500	Integer
MinFaceSize	Face recognition minimum pixels	N	30-300	Integer
SceneMode	<p>Face recognition scene mode</p> <p>0 : normal scene</p> <p>1 : Lobby scene</p>	N	0-1	Integer

FaceTrackEnabled	Face tracking (display tracking frame) 0 : close 1 : open	N	0-1	Integer	
------------------	---	---	-----	---------	--

FTPUploadEnabled	FTP upload face capture 0 : close 1 : open	N	0-1	Integer	
PictureMode	Upload picture mode 0 : face image 1 : Original scene 2 : Face and original picture	N	0-2	Integer	
FacePictureQuality	Face image quality	N	1-99	Integer	
ScenePictureQuality	Scene picture quality	N	1-99	Integer	
FaceAttributeEnabled	Face attribute detection 0 : close 1 : open	N	0-1	Integer	
TimeTable	Arming period (Only supports 2 time periods)	N	<= 2	JSON number group	
TimeEnabled	Time period enabled 0 : not enabled 1 : enable	N	0-1	Integer	

TimeBegin	Starting time Format: hh: mm: ss (Accurate to minutes)	N		String	
TimeEnd	End Time Format: hh: mm: ss	N		String	

	(Accurate to minutes)				
--	-----------------------	--	--	--	--

Return parameter

name	definition	essen tial	Length / Ra nge	Types of	Remark s
Name	Fixed "setFaceParameters Response "	Y	Fixed 25	String	
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Require d paramet ers for NVR connecti on
Session	Register to return to Session	N		String	

Data	Back to content	Y		Json to Elephant	
FaceEnabled	Face recognition function switch 0 : close 1 : open	Y	0-1	Integer	
Sensitivity	Face sensitivity	Y	0-10	Integer	
CaptureMode	Snap mode 0 : Snapshot after leaving (distance selection) 1 : Quick snapshot	Y	0-12	Integer	

	2 : Interval snapshot (in seconds) 3 : Interval snapshot (in frames) 4 : Single player mode 5 : Capture after leaving (quality selection) 6 : Quick + snapshot after leaving 12 : Full-grab mode				
CaptureModeParam	Required parameters in each snapshot mode (Corresponding to the parameters required for the snap mode, some modes do not require	N		JSON pair Elephant	

	this parameter)			
MaxCaptureTimes	<p>Maximum number of snapshots</p> <p>Note: "Snapshot after leaving" mode</p> <p>(CaptureMode = 0/5) unique</p> <p>Parameter, this parameter is invalid in other modes.</p>	N	1-3	Integer
CaptureTimes	<p>Number of snapshots</p> <p>Note: "Interval snapshot", "Single mode"</p> <p>(CaptureMode = 2/3/4)</p> <p>unique parameter, this parameter is invalid in other modes.</p>	N	<p>0-20</p> <p>(0: stands for "who continued to capture")</p>	Integer
Capture Interval	<p>Capture interval (unit: second / frame) Note: "spacer capture", "single mode formula" (C A P T U R & It E Mode = 2 / . 3 / . 4)</p> <p>Unique parameters, which should be used in other modes</p>	N	<p>1-30 seconds (in seconds between unit spacer capture)</p> <p>10-1500 frames</p>	Integer

	<p>The number is invalid.</p> <p>When "Snapshot at intervals of seconds" is set, the unit of this parameter is seconds. The parameter unit is frame.</p>		<p>(In frame single bit interval grip beat) 1-1500 frames</p> <p>(Single mode type)</p>		
MaxFaceSize	Maximum pixels for face recognition	Y	300-500	Integer	
MinFaceSize	Face recognition minimum pixels	Y	30-300	Integer	
SceneMode	<p>Face recognition scene mode</p> <p>0 : normal scene</p> <p>1 : Lobby scene</p>	Y	0-1	Integer	
FaceTrackEnabled	<p>Face tracking (display tracking frame)</p> <p>0 : close</p> <p>1 : open</p>	Y	0-1	Integer	
FTPUploadEnabled	<p>FTP upload face capture</p> <p>0 : close</p> <p>1 : open</p>	Y	0-1	Integer	
PictureMode	<p>Upload picture mode</p> <p>0 : face image</p> <p>1 : Original scene</p> <p>2 : Face and original picture</p>	Y	0-2	Integer	

FacePictureQuality	Face image quality	Y	1-99	Integer	
ScenePictureQuality	Scene picture quality	Y	1-99	Integer	
FaceAttributeEnabled	Face attribute detection 0 : close 1 : open	Y	0-1	Integer	
TimeTable	Arming period (Only supports 2 time periods)	Y	<= 2	JSON number group	
TimeEnabled	Time period enabled 0 : not enabled 1 : enable	Y	0-1	Integer	
TimeBegin	Starting time Format: hh: mm: ss (Accurate to minutes)	Y		String	
TimeEnd	End Time Format: hh: mm: ss (Accurate to minutes)	Y		String	
Code	Return opcode 1 success 0 failed	Y	Max 8	Integer	

Message	Return operation information	Y	Max 200	String	
---------	------------------------------	---	---------	--------	--

JSON format example

send:

```
{
  "Name":
  "setFaceParametersR
  equest", "UUID":
  "abcdefabcdef", "Sessi
  on":
  "abcdefabcdef _ 1542
  594185 ", "TimeStam
  p": 1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    " Fa
    ceE
    nabl
    ed ":
    1, "S
    ensit
    ivity":
    8,
    "Ca
    ptur
    eMo
    de":
    2,
    "Ca
    ptur
    eMo
    deP
    ara
    m":
    {
      "M
      ax
      Ca
      ptu
      re
      Ti
      me
      s":
      1,
      "C
      apt
```

```
ur
eTi
me
s":
0,
"C
apt
ur
e I
nte
rva
l ":
10
},
"MaxFaceSi
ze": 300,
"MinFaceSiz
e": 30,
"SceneMod
e": 0,
" FaceTrack
Enabled ": 1,
" FTPUpload
Enabled ":
0, "PictureM
ode ": 2,
" FacePictur
eQuality ":
99,
" ScenePictu
reQuality ":
99,
" FaceAttrib
uteEnabled
":
1, " TimeTabl
e ":
[
    {
        " Ti
meE
nabl
ed":
1,
"Tim
eBe
gin":
"00:
00:0
0",
"TimeEnd": "23:59:00"
    },
    {
```

```
        "TimeBegin": "00:00:00",  
        "TimeEnd": "23:59:00"  
    }  
]  
}  
  
}  
  
return:  
{  
    "Name":  
    " setFaceParameter  
    sResponse", "Session":  
    " abcdefabcdef _ 15  
    42594185 ", "Data" :  
    {  
        "FaceEnabled":  
        1, "Sensitivity":  
        8,  
        "CaptureMode":  
        2,  
        "CaptureParameter":  
        {  
            "MaxCa
```



```
    ptu
    re
    Ti
    me
    s":
    1,
    "C
    apt
    ur
    eTi
    me
    s":
    0,
    "C
    apt
    ur
    e I
    nte
    rva
    l ":
    10
  },
  "MaxFaceSi
  ze": 300,
  "MinFaceSiz
  e": 30,
  "SceneMod
  e": 0,
  " FaceTrack
  Enabled ": 1,
  " FTPUpload
  Enabled ":
  0, "PictureM
  ode ": 2,
  " FacePictur
  eQuality ":
  99,
  " ScenePictu
  reQuality ":
  99,
  " FaceAttrib
  uteEnabled
  ":
  1, " TimeTabl
  e ":
  [
    {
      " Ti
      meE
      nabl
      ed":
      1,
      "Tim
```

```

        eBe
        gin":
        "00:
        00:0
        0",
        "TimeEnd": "23:59:00"
    },
    {
        "Ti
        meE
        nabl
        ed":
        1,
        "Tim
        eBe
        gin":
        "00:
        00:0
        0",
        "TimeEnd": "23:59:00"
    }
]
},
"Code": 1,
"Message": ""
}
    
```

11. Get face recognition alarm parameters

Request parameters

Return parameter

name	definition	essen tial	Length / R ange	Types of	Remark s
Name	Fixed "getFaceAlarmPar ameters Response "	Y	Fixed 30	String	

ChannelNo	Channel number (management NVR each through channel the IPC ,the value of 255 for the management of NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
Data	Back to content	Y		JsonObject Elephant	
FaceAlarmEnabled	Face recognition alarm switch 0 : close 1 : open	Y	0-1	Integer	
BlackListAlarmEnabled	Blacklist alarm switch 0 : close 1 : open	Y	0-1	Integer	
WhiteListAlarmEnabled	White list alarm switch 0 : close 1 : open	Y	0-1	Integer	

VIPListAlarmEnabled	VIP list alarm switch 0 : close 1 : open	Y	0-1	Integer	
---------------------	--	---	-----	---------	--

NonWhiteListAlarmEnabled	Non-white list alarm switch 0 : close 1 : open	Y	0-1	Integer	
IOAlarmEnabled	IO output switch 0 : close 1 : open	Y	0-1	Integer	
IOStateType	IO output status type 0 : normally open 1 : Normally closed	Y	0-1	Integer	
IOSignalType	IO output signal type 0 : continuous output 1 : pulse output	Y	0-1	Integer	
IOAlarmTime	Alarm output duration (Unit: seconds)	Y	1-60	Integer	
FaceAlarmMode	Face recognition mode 0 : frequency identification 1 : Always recognize 2 : Leaflet recognition	Y	0-2	Integer	
FaceAlarmTimes	Face recognition times	N	1-10	Integer	

	<p>(Unit: times)</p> <p>Note: The specific parameters of the "number recognition" mode</p> <p>Number</p> <p>(F. A CE A L A R & I t m M O D E = 0 which is invalid other modes.</p>				
Similarity	Face recognition acquaintance	Y	1-100	Integer	
TimeTable	<p>Arming period</p> <p>(Only supports 2 time periods)</p>	Y	<= 2	JSON number group	
TimeEnable	<p>Time period enabled</p> <p>0 : not enabled</p> <p>1 : enable</p>	Y	0-1	Integer	
TimeBegin	<p>Starting time</p> <p>Format: hh: mm: ss</p> <p>(Accurate to minutes)</p>	Y		String	
TimeEnd	<p>End Time</p> <p>Format: hh: mm: ss</p> <p>(Accurate to minutes)</p>	Y		String	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{  
  
  "Name":  
  "getFaceAlarmParametersR  
  equest", "UUID":  
  "abcdefabcdef", "Session":  
  " abcdefabcdef _ 154259418  
  5 ", "TimeStamp":  
  1542594185,  
  "Sign": " ec304c35342cbe0609e7c7fec4d97ca5"  
}
```

return:

```
{  
  "Name":  
  "getFaceAlarmParameter  
  sResponse", "Session":  
  " abcdefabcdef _ 1542594  
  185 ", "Data":  
  {  
    " FaceAlarm  
    Enabled ": 1,  
    " BlackListAl  
    armEnabled  
    ": 1,  
    " WhiteListA  
    larmEnable  
    d ": 1,  
    " VIPListAla  
    rmEnabled ":  
    1,  
    " NonWhite  
    ListAlarmEn  
    abled ": 1,  
    " IOAlarmEn  
    abled ": 1,  
    " IOStateTy  
    pe ": 1,  
    " IOSignalT  
    ype ": 1,  
    " IOAlarmTi  
    me ":  
    1," FaceAlar  
    mMode ":  
    1," FaceAlar  
    mTimes ":
```

```
1," Similarity
": 80,
"
T
i
m
e
T
a
b
l
e
"
:
[
{
"Ti
meE
nabl
e": 1,
"Tim
eBe
gin":
"00:
00:0
0",
"TimeEnd": "23:59:00"
},
{
"Ti
meE
nabl
e": 1,
"Tim
eBe
gin":
"00:
00:0
0",
"TimeEnd": "23:59:00"
}
]
},
"Code": 1,
"Message": ""
}
```

12. Face recognition alarm parameter setting

Request parameters

FaceAlarmEnabled	Face recognition alarm switch 0 : close 1 : open	N	0-1	Integer	
BlackListAlarmEnabled	Blacklist alarm switch 0 : close 1 : open	N	0-1	Integer	
WhiteListAlarmEnabled	White list alarm switch 0 : close 1 : open	N	0-1	Integer	
VIPListAlarmEnabled	VIP list alarm switch 0 : close 1 : open	N	0-1	Integer	
NonWhiteListAlarmEnabled	Non-white list alarm switch 0 : close 1 : open	N	0-1	Integer	
IOAlarmEnabled	IO output switch 0 : close 1 : open	N	0-1	Integer	

IOStateType	IO output status type 0 : normally open 1 : Normally closed	N	0-1	Integer	
IOSignalType	IO output signal type	N	0-1	Integer	

	0 : continuous output 1 : pulse output				
IOAlarmTime	Alarm output duration (Unit: seconds)	N	1-60	Integer	
FaceAlarmMode	Face recognition mode 0 : frequency identification 1 : Always recognize 2 : Leaflet recognition	N	0-2	Integer	
FaceAlarmTimes	Face recognition times (Unit: times) Note: The "identification number" specific mode parameter number (F . A C E A L A R & I t m M O D E = 0 the other inactive mode.	N	1-10	Integer	
Similarity	Face recognition acquaintance	N	1-100	Integer	

TimeTable	Arming period (Only supports 2 time periods)	N	<= 2	JSON number group	
TimeEnable	Time period enabled 0 : not enabled 1 : enable	N	0-1	Integer	
TimeBegin	Starting time Format: hh: mm: ss (Accurate to minutes)	N		String	

TimeEnd	End Time Format: hh: mm: ss (Accurate to minutes)	N		String	
---------	---	---	--	--------	--

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "setFaceAlarmParametersResponse "	Y	Fixed 30	String	
ChannelNo	Channel number (manage IPC of NVR channels ,	N		Integer	Required parameter

	value 255 is management NVR)				ters for NVR connection
Session	Register to return to Session	N		String	
Data	Back to content	Y		Json to Elephant	
FaceAlarmEnabled	Face recognition alarm switch 0 : close 1 : open	Y	0-1	Integer	
BlackListAlarmEnabled	Blacklist alarm switch 0 : close	Y	0-1	Integer	

	1 : open				
WhiteListAlarmEnabled	White list alarm switch 0 : close 1 : open	Y	0-1	Integer	
VIPListAlarmEnabled	VIP list alarm switch 0 : close 1 : open	Y	0-1	Integer	

NonWhiteListAlarmEnabled	Non-white list alarm switch 0 : close 1 : open	Y	0-1	Integer	
IOAlarmEnabled	IO output switch 0 : close 1 : open	Y	0-1	Integer	
IOStateType	IO output status type 0 : normally open 1 : Normally closed	Y	0-1	Integer	
IOSignalType	IO output signal type 0 : continuous output 1 : pulse output	Y	0-1	Integer	
IOAlarmTime	Alarm output duration (Unit: seconds)	Y	1-60	Integer	
FaceAlarmMode	Face recognition mode	Y	0-2	Integer	

	0 : frequency identification 1 : Always recognize 2 : Leaflet recognition				
--	---	--	--	--	--

FaceAlarmTimes	Face recognition times (Unit: times) Note: The "identification number" specific mode parameter number(F. A C E A L A R & I t m M O D E = 0 the other Mode is invalid.	N	1-10	Integer	
Similarity	Face recognition acquaintance	Y	1-100	Integer	
TimeTable	Arming period (Only supports 2 time periods)	Y	<= 2	JSON number group	
TimeEnable	Time period enabled 0 : not enabled 1 : enable	Y	0-1	Integer	
TimeBegin	Starting time Format: hh: mm: ss (Accurate to minutes)	Y		String	
TimeEnd	End Time Format: hh: mm: ss (Accurate to minutes)	Y		String	
Code	Return opcode	Y	Max 8	Integer	

	1 success 0 failed				
Message	Return operation information	Y	Max 200	String	

JSON format example

send:

```
{
  "Name":
  "setFaceAlarmParametersR
  equest", "UUID":
  "abcdefabcdef", "Session":
  " abcdefabcdef _ 154259418
  5 ", "TimeStamp":
  1542594185,
  "Sign":
  " ec304c35342cbe0609e7c7fec4
  d97ca5", "Data":
  {
    " FaceAlarm
    Enabled ": 1,
    " BlackListAl
    armEnabled
    ": 1,
    " WhiteListA
    larmEnable
    d ": 1,
    " VIPListAla
    rmEnabled ":
    1,
    " NonWhite
    ListAlarmEn
    abled ": 1,
    " IOAlarmEn
    abled ": 1,
    " IOStateTy
    pe ": 1,
    " IOSignalT
    ype ": 1,
    " IOAlarmTi
    me ":
    1," FaceAlar
    mMode ":
    1," FaceAlar
    mTimes ":
```

```
1," Similarity
": 80,
"
T
i
m
e
T
a
b
l
e
"
:
[
{
"Ti
meE
nabl
ed":
1,
"Tim
eBe
gin":
"00:
00:0
0",
"TimeEnd": "23:59:00"
},
{
"Ti
meE
nabl
ed":
1,
"Tim
eBe
gin":
"00:
00:0
0",
"TimeEnd": "23:59:00"
}
]
}
}
```

```
return:
{
  "Name":
  "setFaceAlarmParameter
  sResponse", "Session":
  " abcdefabcdef _ 1542594
  185 ", "Data":
  {
    " FaceAlarm
    Enabled ": 1,
    " BlackListAl
    armEnabled
    ": 1,
    " WhiteListA
    larmEnable
    d ": 1,
    " VIPListAla
    rmEnabled ":
    1,
    " NonWhite
    ListAlarmEn
    abled ": 1,
    " IOAlarmEn
    abled ": 1,
    " IOStateTy
    pe ": 1,
    " IOSignalT
    ype ": 1,
    " IOAlarmTi
    me ":
    1," FaceAlar
    mMode ":
    1," FaceAlar
    mTimes ":
    1," Similarity
    ": 80,
    "
    T
    i
    m
    e
    T
    a
    b
    l
    e
    "
    :
    [
    {
```



```

        "TimeBegin": "00:00:00",
        "TimeEnd": "23:59:00"
    },
    {
        "TimeBegin": "00:00:00",
        "TimeEnd": "23:59:00"
    }
]
},
"Code": 1,
"Message": ""
}
    
```

13. Network parameter acquisition Request parameters

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
------	------------	-----------	----------------	----------	---------

Name	Fixed "getNetParametersResponse"	Y	Fixed 24	String	
Channel No	Channel number (manage NVR channels of the IPC, the value 25.5 for managing NVR & It)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	
Data	Back to content	Y		JsonObject Elephant	
DHCPEnabled	DHCP switch 0 : close 1 : open	Y	0-1	Integer	
IPAddresses	IP address Format: xxx.xxx.xxx.xxx Example: 192.168.1.88	Y		String	
SubNetMask	Subnet mask Format: xxx.xxx.xxx.xxx Example: 255.255.255.0	Y		String	
Gateway	Gateway	Y		String	

	Format: xxx.xxx.xxx.xxx Example: 192.168.1.1			
DNS1	Preferred DNS address Format: xxx.xxx.xxx.xxx Example: 114.114.114.114	Y		String
DNS2	Alternate DNS address Format: xxx.xxx.xxx.xxx Example: 8.8.8.8	Y		String
Code	Return opcode	Y		Integer
Message	Return operation information	Y	Max 200	String

JSON format example

send:

```
{
  "Name":
  "getNetParametersRequest", "UUID":
  "abcdefabcdef", "Session":
  "abcdefabcdef _ 1542594185", "Timestamp":
  "1542594185",
  "Sign": " ec304c35342cbe0609e7c7fec4d97ca5"
}
```

return:

```
{
  "Name":
  "getNetParametersResponse", "Session":
  "abcdefabcdef _ 1542594185", "Data":
  {
    "DHCPEnabled": 1,
```

```

"IPAddress": "192.168.1.88",
"SubNetMask": "255.255.255.0",

"Gateway": "192.168.1.1",
"DNS1": "114.114.114.114"
"DNS2": "8.8.8.8"
},
"Code": 1,
"Message": ""
}

```

14. Network parameter setting

Set the network parameters. If the parameters are incorrectly modified, it may cause the network to fail. Please set them carefully.

Request parameters

name	definition	must need	Length / Range	Types of	Remarks
Name	Solid fixed "setNetParametersRequest"	Y	Fixed 23	String	
UUID	Device UUID	N	Fixed 12	String	Required parameters for WAN access number
ChannelNo	Channel number (management of each channel of NVR IPC , value 255 is management NVR)	N		Integer	NVR docking Required parameters

Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification

Sign	Authority verification information, when the server parameter setting " Management Authority Verification " is enabled verification.	N	Fixed 32	String	
Data	Request content	Y		Json object	
DHCPEnabled	DHCP switch 0 : close 1 : open	N	0-1	Integer	
IPAddress	IP address Format: xxx.xxx.xxx.xxx Example: 192.168.1.88	N		String	
SubNetMask	Subnet mask Format: xxx.xxx.xxx.xxx Example: 255.255.0	N		String	
Gateway	Gateway Format: xxx.xxx.xxx.xxx Example: 192.168.1.1	N		String	

DNS1	Preferred DNS address Format: xxx.xxx.xxx.xxx Example: 114. 114.114.114	N	String
DNS2	Alternate DNS address Format: xxx.xxx.xxx.xxx Example: 8.8.8 .8	N	String

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "setNetParametersResponse"	Y	Fixed 24	String	
Channel No	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	

Data	Back to content	Y		Json to Elephant	
DHCPEnabled	DHCP switch 0 : close 1 : open	Y	0-1	Integer	
IPAddress	IP address Format: xxx.xxx.xxx.xxx Example: 192.168.1.88	Y		String	
SubNetMask	Subnet mask Format: xxx.xxx.xxx.xxx	Y		String	

	Example: 255.255.255.0				
Gateway	Gateway Format: xxx.xxx.xxx.xxx Example: 192.168.1.1	Y		String	
DNS1	Preferred DNS address Format: xxx.xxx.xxx.xxx Example: 114.114.114.114	Y		String	
DNS2	Alternate DNS address Format: xxx.xxx.xxx.xxx Example: 8.8.8.8	Y		String	

Code	Return opcode 1 success 0 failed	Y	Max 8	Integer
Message	Return operation information	Y	Max 200	String

JSON format example

send:

```
{
  "Name":
  "setNetParametersRequest", "UUID":
  "abcdefabcdef", "Session":
  "abcdefabcdef_1542594185", "Timestamp": 1542594185,
  "Sign":
  "ec304c35342cbe0609e7c7fec4d97ca5", "Data":
  {
    "DHCPEnabled": 0,
    "IPAddress": "192.168.1.88",
    "SubNetMask": "255.255.255.0",
    "Gateway": "192.168.1.1",
    "DNS1": "114.114.114.114",
    "DNS2": "8.8.8.8"
  }
}
```

return:

```
{
  "Name":
  "setNetParametersResponse", "Session":
  "abcdefabcdef_1542594185", "Data":
  {
```



```

        "DHCPEnabled": 0,
        "IPAddress": "192.168.1.88",
        "SubNetMask": "255.255.255.0",
        "Gateway": "192.168.1.1",
        "DNS1": "114.114.114.114"
        "DNS2": "8.8.8.8"
    },
    "Code": 1,
    "Message": ""
}
    
```

15. Restart the device Request parameters

ChannelNo	Channel number (manage NVR channels Of the IPC , the value 25 . 5 for managing N V R & It)	N		Integer	NVR docking Required parameters
Session	Register to return to Session	N		String	
TimeStamp	Unix timestamp (seconds)	N	Fixed 10	Integer	Open Sign Required for verification
Sign	Permission verification information, when the server Verification when the parameter setting " Management Authority Verification " is enabled	N	Fixed 32	String	

Return parameter

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "restartResponse"	Y	Fixed 15	String	
ChannelNo	Channel number (manage IPC of NVR channels , value 255 is management NVR)	N		Integer	Required parameters for NVR connection
Session	Register to return to Session	N		String	
Code	Return opcode 1 success	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

Send :

```
{
  "Name":
  "restartRequest",
  "UUID":
  "abcdefabcdef",
  "Session":
  " abcdefabcdef _ 1542594185 ",
  "TimeStamp":
  "1542594185",
  "Sign":
  " ec304c35342cbe0609e7c7fec4d97ca5 "
}
```

Returns :

```
{
  "Name":
  "restartResponse",
  "Session":
  " abcdefabcdef _ 15
  42594185 ", "Code"
  : 1,
  "Message": "Succeed."
}
```

16. Other errors returned

name	definition	essential	Length / Range	Types of	Remarks
Name	Fixed "errorResponse"	Y	Fixed 13	String	
Code	Return opcode	Y		Integer	
Message	Return operation information	Y	Max 200	String	

JSON format example

```
{
  "Name": "errorResponse",
  "Code": 0,
  "Message": ""
}
```

V 1.0.0 version

HTTP protocol error returned in JSON format (old)

parameter name	Parameter meaning	Maximum length (or range)	Types of	Remarks
Result	Explanation of		String	

	results			
ErrorCode	error code		Integer	

json format

```
{
  "
  R
  e
  s
  u
  l
  t
  "
  :
  "
  e
  r
  r
  o
  r
  "
  ;
  "
  E
  r
  r
  o
  r
  C
  o
  d
  e
  "
  :
  0
}
```

*** Code list (the device receives information and returns Code)**

Code	Explanation	Remarks
1	success	

0	failure	
1002	<p>JSON parsing error.</p> <p>1. "The 'JSON' format is illegal." : Please check the JSON format.</p>	
1003	<p>Session is invalid.</p> <p>1. " The 'Session' parameter is missing. " : Required</p>	<p>"Registration" function is on</p> <p>Problems that can arise when</p>

	<p>Take the " Session " field request.</p> <p>2. "The 'Session' parameter must be a string." : The " Session " field in JSON must be a string type.</p> <p>3. "The 'Session' parameter does not match." : " Session " does not match the current device registration session .</p>	<p>To turn off the "register" function.</p>
--	--	---

1004	<p>TimeStamp is invalid.</p> <p>1. "The 'TimeStamp' parameter is missing." : Need to bring "TimeStamp" field request. 2. "The 'TimeStamp' parameter must be a number." : The "TimeStamp" field in JSON must be a numeric integer type.</p>	<p>" Sign verify" function appears when you open ask questions, you can " Sign verify" function is turned off.</p>
1005	<p>Sign is invalid.</p> <p>1. of The 'Sign' Parameter IS Missing . :Need to bring "Sign" field of the request. 2. "The 'Sign' parameter must be a string." : The " Session " field in JSON must be a string type. 3. " T h e ' S i g n ' c h e c k f a i l e d . " 'S i g n " value verification failed,</p>	<p>" Sign verify" function appears when you open ask questions, you can " Sign verify" function is turned off.</p>

	<p>For " Sign " generation rules, please see * Sign verification #* Sign verification</p>	
--	---	--

1006	<p>Name is invalid.</p> <p>1. "The ' Name' parameter is missing." : Need to bring the " Name " field request.</p> <p>2. " T h e ' N a m e ' p a r r a m e t e r i s i n v a l i d . " N a m e "</p> <p>The segment is incorrect and there is no such interface.</p>	
1101	<p>Data is invalid.</p> <p>1. " The ' Data ' parameter is missing." : The " Data " request parameter is missing .</p>	
1102	<p>Action is invalid.</p> <p>1. "The ' Action ' parameter is missing." : The " Action " parameter is missing .</p> <p>2. " T h e ' A c t i o n ' p a r r a m e t e r i s i n v a l i d . " ! A c t i o n "</p> <p>The field is incorrect.</p>	
1103	<p>PersonType is invalid.</p> <p>1. "The 'PersonType' parameter is missing."</p> <p>2. "The 'PersonType' parameter must be a</p>	

	<p>number. "</p> <p>3. "The 'PersonType' parameter is invalid."</p>	
1104	<p>PersonId is invalid.</p> <p>1. "The 'PersonId' parameter is missing."</p> <p>2. "The 'PersonId' parameter must be a string."</p>	
1105	<p>GetPhoto error.</p> <p>1. "The 'GetPhoto' parameter is missing."</p> <p>2. "The 'GetPhoto' parameter must be a number."</p> <p>3. "The 'GetPhoto' parameter is invalid."</p>	
1106	<p>PersonId does not exist.</p> <p>1. "There IS NO SUCH ON the Person at The List." : List Gallery No</p> <p>If there is a person matching the " PersonId ",</p> <p>check whether the " PersonId " and</p> <p>" PersonType " are correct.</p>	

1107	<p>PersonInfo / PersonInfoList is invalid.</p> <ol style="list-style-type: none"> 1. "The 'PersonInfo' parameter is missing." 2. "The 'PersonInfoList' parameter is missing." 3. "The 'PersonInfoList' parameter must be an 	
------	--	--

	array. "	
1108	<p>PersonName is invalid.</p> <ol style="list-style-type: none"> 1. "The 'PersonName' parameter is missing." 2. "The 'PersonName' parameter must be a string." 	
1109	<p>PersonPhoto is invalid.</p> <ol style="list-style-type: none"> 1. "The 'PersonPhoto' parameter is missing." 2. "The 'PersonPhoto' parameter is invalid." 3. "The 'PersonPhoto' parameter must be a string." 	
1110	Failed to add a picture internally, see the Result field for the specific error code .	
1111	<p>Other errors.</p> <p>Refer to " Message " for details.</p>	

*** Result error code**

Result error code	meaning
-1	failure
-2	Failed to extract feature value
-3	File name number duplicate
-4	Kuman
-5	Add timeout
-6	Parameter error
-7	file size too big
-8	not enough storage space
-9	File failed to open
-10	No database

-11	Image reading failed
-12	Database file is damaged
-13	Poor picture quality
-14	Image size is wrong, width and height cannot be odd
-15	Face detection failed (no face detected or multiple faces detected)
-16	Picture format error
-17	Face area error
-18	Algorithm creation handle error
-19	Equipment busy
-20	File write failed
-twenty one	Failed to delete (the corresponding ID was not found to be deleted)

Six, common problems

1. Failed to add picture

- (1) View the list of "error codes".
- (2) The picture field is the base64 encoding of pure picture data , without the prefix of "data: image / jpeg; base64 ," .
- (3) Pictures only support .jpg format, check whether the added pictures are pictures in other formats.
- (4) Face picture requirements: size less than 2M , resolution less than 960 * 960

2. Failed to parse picture

(1) The image field is the base64 encoding of pure image data . If it is copied to the webpage for analysis, the prefix of "data: image / jpeg; base64," needs to be added .

3. Interface access failed

(1) If the access interface does not receive the returned content, it proves that the address or method of calling the interface is incorrect.

(2) Please return under Code codes, see the interface Code list.

(3) When Code returns to 1003 , it is because the "upload parameter setting" has enabled the "register" function. The device failed to verify the session value of the access interface , so access is denied. The correct session value should be the session value returned by the server when the device registers with the server . If you do not need the "registration" function, you can turn off the "registration" function in the "upload parameter settings".

(4) When the Code returns 1004 or 1005 , it is because " Sign Verification" is enabled in "Upload Parameter Setting ". Provided apparatus verifies the access interface Sign field and TimeStamp field, see the document generation rules " Sign check" section. If you do not need the " Sign verification" function, you can turn off the " Sign verification" function in the "Upload parameter settings" .