HIKVISION

Web3.2 development kit

Interfaces description

Catalog

1. Overview 5
1.1 Introduction 5
1.2 Supported devices 5
1.3 Running environment 5
2. Revision history 5
3. Error code definition 7
3.1. Abnormal event code 7
3.2. Error code 7
4. Function calling sequence 9
5. Function description 10
5.1. Plugin initialization 10
5.1.1. Check if the browser supports plugins or not 10
5.1.2. Check plugin is installed or not 10
5.1.3. Web plugin initialization (including plugin event registration) 10
5.1.4. Embed play plugin 12
5.1.5. Write plugin in web 12
5.2. Get device information 12
5.2.1. Get IP basing on DNS 12
5.2.2. Login device 13
5.2.3. Logout device 13
5.2.4. Get basic information of devices 13
5.2.5. Get analog channels information 14
5.2.6. Get digital channel infomation 15
5.2.7. Get zero channel information 16
5.2.8. Record search 16
5.2.9. Get voice intercom channel 18
5.2.10. Get port 18
5.3. Play and play control 19
5.3.1. Start realplay 19
5.3.2. Start playback 19
5.3.3. Start reverse playback 21
5.3.4. Stop play 21
5.3.5. Single frame 22
5.3.6. Pause 22
5.3.7. Resume Play 22
5.3.8. Slow forward 23
5.3.9. Fast forward 23
5.3.10. Get OSD time 23
5.3.11. Enable sound 24
5.3.12. Disable sound 24
5.3.13. Set volume 24

5.3.14. Capture pictures 24
5.3.15. Get image resource binary data without plugin 25
5.3.16. Split screen 25
5.4. Recording 25
5.4.1. Start recording 25
5.4.2. Stop recording 25
5.5. Download record 26
5.5.1. Start downloading 26
5.5.2. Start downloading by time 26
5.5.3. Get the records downloading status 27
5.5.4. Get the records downloading progress 27
5.5.5. Stop downloading records 27
5.6. Voice intercom 27
5.6.1. Start voice intercom 27
5.6.2. Stop voice intercom 27
5.7. PTZ control 28
5.7.1. PTZ direction control 28
5.7.2. Setting preset 28
5.7.3. Calling preset 28
5.8. Enlarging image 29
5.8.1. Enable electronic zoom 29
5.8.2. Disable electronic zoom 29
5.8.3. Enable 3D zoom 29
5.8.4. Disable 3D zoom 29
5.8.5. Full-screen play 29
5.9. Device maintaince 30
5.9.1. Export device configuring parameters 30
5.9.2. Import device configuring parameters 30
5.9.3. Restore the default parameters 30
5.9.4. Restart 30
5.9.5. Start upgrading 31
5.9.6. Start asynchronous upgrade 31
5.9.7. Get upgrading status 31
5.9.8. Get upgrading progress 31
5.9.9. Stop upgrading 32
5.9.10. Reconnect 32
5.9.11. Open remote configuration 32
5.10. Plugin information maintaince 32
5.10.1. Plugin version comparison 32
5.10.2. Get the local configuring parameters 33
5.10.3. Set the local configuration of plugins 33
5.10.4. Get playing window status 33
5.11. Draw polygon in window 34
5.11.1. Set play mode 34

5.11.2. Set draw mode 34
5.11.3. Set polygon information 34
5.11.4. Set polygon 34
5.11.5. Get polygon information 35
5.11.6. Clear polygon information 35
5.12. Others 35
5.12.1. Folder selection and files' path 35
5.12.2. Asynchronously folder selection and files' path 35
5.12.3. Get the last error code 35
5.12.4. Send HTTP Request 36
5.12.5. Set encapsulation format 36
5.12.6. Capture picture 36

1. Overview

1.1 Introduction

The web development kit V3.2 is developed basing on ActiveX and NPAPI, interfaces are encapsulated in javascript, so interfaces of javascript will be provided for integration. Functions of preview, playback, PTZ control and so on are supported via web. This development kit only can be used for B/S, but not C/S.

1.2 Supported devices

Most of devices of Hikvision are supported by this development kit V3.2, including DVR, NVR, DVS, network camera, network speed dome, etc., but devices should support PSIA or ISAPI protocol.

1.3 Running environment

OS: Windows XP, Windows7, Windows8, Windows8.1
Brower:
IE8~IE11, Chrome31+, Firefox35+, 32 bit browser
IE8~IE11, Chrome31+, Firefox35+, 64bit browser

2. Revision history

Version	Description
V1.0.0	● The development kit of this version supports synchronous / asynchronous, cross-domain CGI command (PSIA / ISAPI), but only some basic commands can be supported, such as basic device information, channel acquisition, PTZ control, etc. ● Playing mode only support rtsp over tcp and rtsp over udp at present.
V1.0.1	● Modify the HTTP interaction course and solve the problem that old version devices can’t login. ● Solve the Bug that NVR can’t call preset
V1.0.2	● Solve the problem that RTSP port can’t be obtained for some devices
V1.0.3	● Getting stream of private protocol is added, including preview, playback and

	reverse playback. The private protocol will be used in tcp mode by default.
V1.0.4	● Redesign PTZ operation interface (I_PTZControl), and focus, zoom, aperture functions are added. The flag of start and stop is also added in this interface.
V1.0.5	● Remote configuration library is modified as modeless way (I_RemoteConfig) to avoid jamming when using firefox. ● Extend remote configuration interface (I_RemoteConfig), add language choice(Chinese and English). ● Add Digest Authentication ● Solve the problem of 3D zoom
V1.0.9	● Add interface: check plugin is installed or not (I_CheckPluginInstall). ● Modify web plugin initialization interface (I_IninPlugin): add bWndFull. ● Add HTTP request interface (I_SendHTTPRequest).
V1.1.0	● Add interface: draw polygon in window(I_SetSnapDrawMode, I_SetSnapPolygonInfo, I_GetSnapPolygonInfo, I_ClearSnapInfo); ● Add interface : capture picture without preview(I_DeviceCapturePic); ● Initialize plugin interface(I_InitPlugin), add optional parameters(iPackageType, cbRemoteConfig, cbDoubleClickWnd, cbInitPluginComplete), and cbInitPluginComplete is required to define. ● For capture, record and download interface, add optional parameter: bDateDir. ● Add interface: I_SetPackageType, set encapsulation format for record and download files. ● Support searching, playing and downloading sub stream recording files. ● Support requirejs、seajs. ● IP Server/HiDDNS interface: I_GetIPInfoByMode, only supported by 32bit web SDK with remote configuration lib. ● Fix some bugs: Fix crashed problem when close multiple tab in IE11 Fix failed problem when requesting 256 channels at the same time ● Fix failed playback problem in some DS-8632N-I8
V1.1.1	● Add support non-plugin browsers. ● Add the browser plugin support status judgment interface (I_SupportNoPlugin). ● Add asynchronous interface: open the selection box (I2_OpenFileDlg), add no plug-in to get the image resource binary data interface (I2_CapturePicData), increase download by time interface (I_StartDownloadRecordByTime), add the setting drawing information interface (I2_SetSnapPolygonInfo), increase the asynchronous upgrade interface (I2_StartUpgrade). ● Add the encryption of sensitive information including parameter import and export

3. Error code definition

3.1. Abnormal event code

Abnormal event callback is disposed in the call back function that users input, the first parameter is event code (abnormal playback, playback stop and not enough hard disk space), and the second parameter is the window number of events.

Event Name	Code	
PLUGIN_EVENTTYPE_PLAYABNORMAL	0	abnormal playback
PLUGIN_EVENTTYPE_PLAYBACKSTOP	2	playback stop
PLUGIN_EVENTTYPE_AUDIOTALKFAIL	3	voice intercom failure
PLUGIN_EVENTTYPE_NOFREESPACE	21	not enough hard disk space(record)

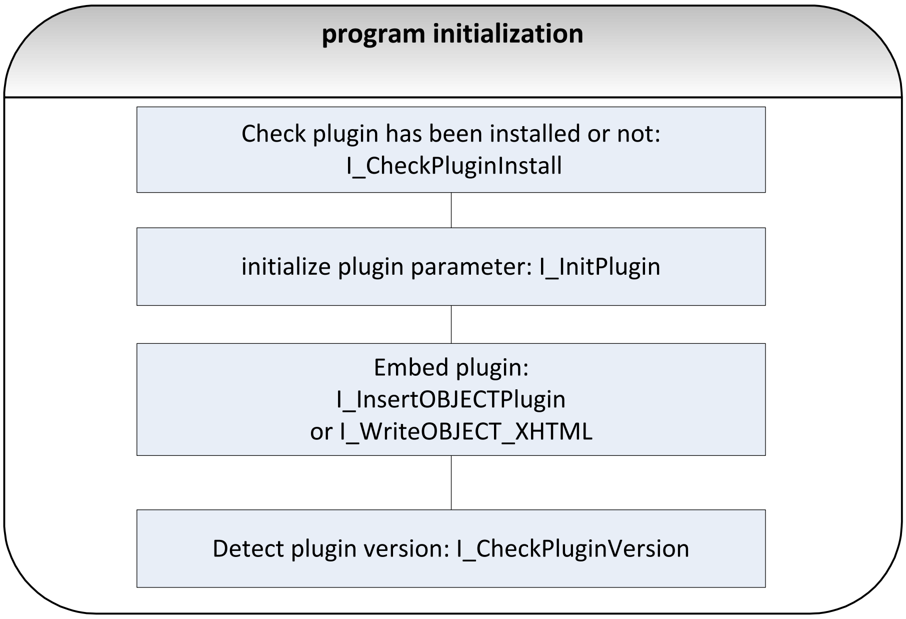
3.2. Error code

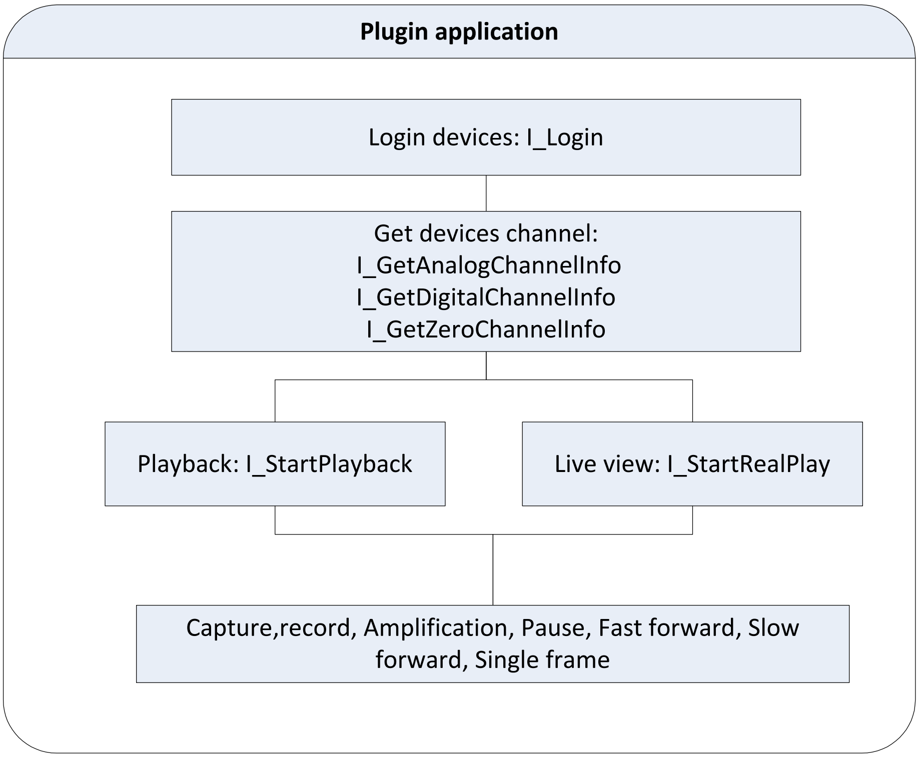
Error code is obtained by calling I_GetLastError, it belongs to the bottom error code. The upper logical errors have no error code.

PLUGIN_ERROR_NOERROR	0	no error
PLUGIN_ERROR_LOAD_RTSP_FAILED	1	failed to load rtsp library
PLUGIN_ERROR_LOAD_PLAYCTRL_FAILED	2	failed to load play library
PLUGIN_ERROR_LOAD_SYSTRANSFORM_FAILED	3	failed to load code stream transform wrapping library
PLUGIN_ERROR_LOAD_HTTPCLIENT_FAILED	4	failed to load http library
PLUGIN_ERROR_PARAMETER_ERROR	5	parameter error
PLUGIN_ERROR_ORDER_ERROR	6	Calling sequence error
PLUGIN_ERROR_ALLOC_RESOURCE_FAILED	7	failed to allocate resources
PLUGIN_ERROR_NOT_INITLIB	8	not initialized
PLUGIN_ERROR_OPERTION_NOSUPPORT	9	operation is not supported
PLUGIN_ERROR_OPENFILE_ERROR	10	failed to open files
PLUGIN_ERROR_WRITEFILE_ERROR	11	failed to write files
PLUGIN_ERROR_READFILE_ERROR	12	failed to read files
PLUGIN_ERROR_INIT_HPR_FAILED	13	Failed to initialize hpr library
PLUGIN_ERROR_AUDIO_MONOPOLIZED	14	sound card is monopolized
PLUGIN_ERROR_CREATE_SOCKET_ERROR	15	failed to create socket
PLUGIN_ERROR_NETWORK_CONNECT_FAILED	16	connection failure
PLUGIN_ERROR_NETWORK_SEND_ERROR	17	failed to send
PLUGIN_ERROR_NETWORK_RECV_ERROR	18	failed to receive

PLUGIN_ERROR_NETWORK_SEND_TIMEOUT	19	send timeout
PLUGIN_ERROR_NETWORK_RECV_TIMEOUT	20	receive timeout
PLUGIN_ERROR_NETWORK_RESOLVE_FAILED	21	domain name resolution error
PLUGIN_ERROR_XML_PARSE_ERROR	22	xml parsing error
PLUGIN_ERROR_XML_NODE_ERROR	23	xml node error
PLUGIN_ERROR_NO_EXCEL_DRIVER_ERROR	24	Excel driver is not installed
PLUGIN_ERROR_PARSE_URL_FAILED	25	URL parsing failure
PLUGIN_ERROR_LOADRTSPSDKPROC_ERROR	26	can't find rtsp interface address
PLUGIN_ERROR_LOADPLAYERSDKPROC_ERROR	27	can't find play library interface address
PLUGIN_ERROR_LOADSYSTRANSFORMPROC_ERROR	28	can't find code stream transform wrapping library interface address
PLUGIN_ERROR_LOADHTTPSDKPROC_ERROR	29	can't find http library interface address
PLUGIN_ERROR_START_WAVEIN_FAILED	30	failed to start audio capture
PLUGIN_ERROR_START_WAVEOUT_FAILED	31	failed to start playing audio
PLUGIN_ERROR_INIT_G722_CODEC_FAILED	32	failed to initialize G722 codec
PLUGIN_ERROR_NOT_ENOUGH_DISK_FREESPACE	33	not enough hard disk
PLUGIN_ERROR_FILE_ALREADY_EXIST	34	file has already existed

4. Function calling sequence


program initialization
Check plugin has been installed or not: I_CheckPluginInstall
initialize plugin parameter: I_InitPlugin
Embed plugin: I_InsertOBJECTPlugin or I_WriteOBJECT_XHTML
Detect plugin version: I_CheckPluginVersion


Plugin application
Login devices: I_Login
Get devices channel: I_GetAnalogChannelInfo I_GetDigitalChannelInfo I_GetZeroChannelInfo
Playback: I_StartPlayback
Live view: I_StartRealPlay
Capture,record, Amplification, Pause, Fast forward, Slow forward, Single frame

5. Function description

5.1. Plugin initialization

5.1.1. Check if the browser supports plugins or not

Function: I_SupportNoPlugin()

Instruction: Check if the browser supports plugins or not

Parameters: none

Return value: true: The browser supports non-plugins;
flase: The browser does not support non-plugins

5.1.2. Check plugin is installed or not

Function: I_CheckPluginInstall()

Instruction: check plugin is installed or not (including Chrome version)

Parameters: none

Return value: -2: The Chrome version doesn’t support NPAPI; -1: not installed; 0: installed.

5.1.3. Web plugin initialization (including plugin event registration)

Function: I_InitPlugin (szWidth, szHight, options)

Instruction: initialize various properties of plugin

Parameter: szWidth plugin width (unit: px, 100% represents full occupy plugin container)

szHight plugin height (unit: px, 100% represents full occupy plugin container)

options Optional parameter objects:

szContainerID the container ID of plugin (the DOM node of HTML), which can be input both when initialization and and when plugin is being embedded.

szColorProperty the color properties, which represents background color of plugin, background color of child window, child window border color, selected border color of child window frame. Plugin has its own default color.

szOcxClassId ocx plugin ID, corresponding ID can be modified when OEM to achieve development kit to bind different plugins, and default plug-in is Hikvision WEB3.0.

szMimeTypes non-IE plugin, corresponding ID can be modified when OEM to achieve development kit to bind different plugins, and default plug-in is Hikvision WEB3.0.

iWndowType split screen types, 1(1*1), 2(2*2), 3(3*3), 4(4*4). The default type is a single screen.

bWndFull double-click to switch full screen in single window, true(support), false(not support).

iPlayMode play mode, the default value is 2: normal play mode. Other modes are not supported at present.

bDebugMode JS debug mode, console prints debug information: true(enable), false(disable)

cbSelWnd window selects event callback function, contains only one string parameter, the value of which is XML.

cbEvent plugin event callback function, has three parameters: the first is event type, and the second is window number.

iPackageType encapsulation format, 2-PS format 11-MP4 format.

cbDoubleClickWnd window double click callback function, there're two parameters, the 1st parameter is window number, the 2nd parameter means fullscreen or not.

cbRemoteConfig callback of closing remote configuration lib

cbInitPluginComplete callback of finishing plugin initialize(required).

Return value: none

Note: the format of szColorProperty: "plugin-background:ffffff; sub-background:ffffff; sub-border:ffffff; sub-border-select:ffffff", which represents background color of plugin, background color of child window, child window border color, selected border color of child window frame.

cbSelWnd is window selects event callback function, users can input function, and development kit will automatically call this function after the window is selected. The parameter is XML, format is as follows:

```
<?xml version="1.0"?>
<RealPlayInfo>
```

<SelectWnd>0</SelectWnd>//the number of window that triggers event, start from 0
</RealPlayInfo>

cbEvent is a callback function of plugin's abnormal event, which has three parameters: the first parameter is event type (each value of abnormal events is introduced in **Abnormal event code**), the second represents the number of window that triggers events.

5.1.4. Embed play plugin

Function: I_InsertOBJECTPlugin (szContainerID)
Instruction: embed play plugin in HTML DOM
Parameter: szContainerID the container ID of plugin, which is the DOM of HTML
Return value: 0-success, -1-failure

5.1.5. Write plugin in web

Function: I_WriteOBJECT_XHTML ()
Instruction: Playing plug is inserted in web directly
Parameter: none
Return value: 0-success, -1-failure

5.2. Get device information

5.2.1. Get IP basing on DNS

Function: I_GetIPInfoByMode (iMode, szAddress, iPort, szDeviceInfo)
Instruction: Get IP basing on DNS
Parameter: iMode DNS server mode, 0-IP_Domain 1-IPServer 2-HIDDNS
szAddress DNS server IP
iPort DNS server port
szDeviceInfo device serial number or device name(or HiDDNS)
Return value: success: return "device IP address-device SDK port"("-" is used to serve as separator between IP and port); failure: return ""(null string). I_Login is called after getting the IP address of device.

5.2.2. Login device

Function: I_Login (szIP, iPrototocol, iPort, szUserName, szPassword, options)

Instruction: login device

Parameter: szIP device IP address

iPrototocol 1: http protocol, 2: https protocol

iPort login the http/https port of devices, choose different ports according to iPrototocol

szUserName username

szPassword passward

options optional parameter objects:

async http interactive way, true: asynchronous, false: synchronous

cgi CGI protocol, 1:ISAPI, 2:PSIA. If this parameter is not input, a kind of protocol that devices support will be chosen.

success success callback function, there is one parameter that represents the content of XML.

error failure callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

Return value: none

Note: login devices by calling this function; if login successfully, http/https protocol and PSIA/ISAPI protocol will be selected, and the following interaction with devices will adopt the selected protocol. The successful callback function will be called when interact successfully; otherwise, failed callback function will be called.

5.2.3. Logout device

Function: I_Logout (szDeviceIdentify)

Instruction: Logout device

Parameters: szDeviceIdentify device identity（IP_Port）

Return value: 0-success, -1-failure

5.2.4. Get basic information of devices

Function: I_GetDeviceInfo (szDeviceIdentify, options)

Instruction: get basic information of devices

Parameters: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

async http interactive way, true: asynchronous，false: synchronous

success successful callback function, there is one parameter that represents the content of XML.

error failed callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

Return value: none

Note: The successful callback function will be called when interact successfully; otherwise, failed callback function will be called. The first parameter of callback function is the XML of devices information.

XML format is as follows:

```
<DeviceInfo>
    <deviceName></deviceName>      //device name
    <deviceID></deviceID>          //device ID
    <deviceType></deviceType>      //device type(may be null)
    <model></model>                //device model
    <serialNumber></serialNumber>  //device serial number
    <macAddress></macAddress>      //device mac address
    <firmwareVersion></firmwareVersion> //device firmware vertion
    <firmwareReleasedDate></firmwareReleasedDate>   //release date of firmware
    <encoderVersion></encoderVersion>   //encoder version
    <encoderReleasedDate></encoderReleasedDate>     //release date of encoder
</DeviceInfo>
```

5.2.5. Get analog channels information

Function: I_GetAnalogChannelInfo (szDeviceIdentify, options)

Instruction: get analog channels information

Parameter: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

async http interactive way, true:asynchronous，false:synchronous

success successful callback function, there is one parameter that represents the content of XML.

error failed callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

Return value: none

Note: The successful callback function will be called when interact successfully; otherwise, failed callback function will be called. The first parameter of callback function is the XML of devices information.

XML format is as follows:

```
<VideoInputChannelList>
    <VideoInputChannel>
        <id></id>        //channel ID
        <inputPort></inputPort>    //channel number
        <videoInputEnabled></videoInputEnabled>    //whether to enable
        <name></name>     //channel name
        <videoFormat></videoFormat>    //channel format
    </VideoInputChannel>
</VideoInputChannelList>
```

5.2.6. Get digital channel infomation

Function: I_GetDigitalChannelInfo (szDeviceIdentify, options)

Instruction: get digital channel information

Parameter: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

async http interactive way, true: asynchronous，false: synchronous

success successful callback function, there is one parameter that represents the content of XML.

error failed callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

Return value: none

Note: The successful callback function will be called when interact successfully; otherwise, failed callback function will be called. The first parameter of callback function is the XML of devices information.

XML format is as follows:

```
<InputProxyChannelStatusList>
    <InputProxyChannelStatus>
        <id></id>    //channel ID
        <sourceInputPortDescriptor>
            <proxyProtocol></proxyProtocol>    //connection protocol
            <addressingFormatType></addressingFormatType>    //IP address format type
            <ipAddress></ipAddress>    //IP address
            <managePortNo></managePortNo>    //managing port number
            <srcInputPort></srcInputPort>    //IP channel number
            <userName></userName>    //connected user name
            <streamType></streamType> //code stream type
            <online></online> //online or not（true/false）
        </sourceInputPortDescriptor>
    </InputProxyChannelStatus>
</InputProxyChannelStatusList>
```

5.2.7. Get zero channel information

Function: I_GetZeroChannelInfo (szDeviceIdentify, options)

Instruction: get zero channel information

Parameters: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

async http interactive way, true:asynchronous，false:synchronous

success successful callback function, there is one parameter that represents the content of XML.

error failed callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

Return value: none

Note: The successful callback function will be called when interact successfully; otherwise, failed callback function will be called. The first parameter of callback function is the XML of devices information.

XML format is as follows:

```
<ZeroVideoChannelList>
    <ZeroVideoChannel>
        <id>1</id>    //channel ID
        <enabled>true</enabled>    //whether to enable
        <inputPort>1</inputPort> //input port
    </ZeroVideoChannel>
</ZeroVideoChannelList>
```

5.2.8. Record search

Function: I_RecordSearch (szDeviceIdentify, iChannelID, szStartTime, szEndTime, options)

Instruction: record search

Parameters: szDeviceIdentify device identity（IP_Port）

iChannelID channel ID

szStartTime start time, eg: 2013-12-23 00:00:00

szEndTime end time, eg:2013-12-23 23:59:59

options optional parameter objects:

async http interactive way, true:asynchronous，false:synchronous

iSearchPos search video location(the default value is 0),0 represents the 0-40 returned results, 40 represents 40-80 returned results, and so on.

success successful callback function, there is one parameter that represents the content of XML.

error failed callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

iStreamType code stream type: 1-main stream, 2-sub stream, main stream is adopted to preview by default.

Return value: none

Note: The successful callback function will be called when interact successfully; otherwise, failed callback function will be called. The first parameter of callback function is the XML of devices information.

The maximum number of returned results is 40, and if the number is over 40, you need to call this interface many times, and set up a search location.

XML format is as follows:

```
<CMSearchResult>
    <responseStatus>true</responseStatus>
    <responseStatusStrg>MORE</responseStatusStrg>// decide to search or not according to this status flag. OK stands for search is finished.
    <numOfMatches>40</numOfMatches> // The number of videos returned this search
    <matchList>
        <searchMatchItem>
            <trackID>101</trackID>//record ID
            <startTime>2013-12-23T03:06:58Z</startTime> //the start time of record
            <endTime>2013-12-23T03:16:57Z</endTime>//the  end time of record

            <playbackURI>rtsp://172.9.4.222/Streaming/tracks/101/?starttime=20131223T030658Z&amp;endtime=20131223T031657Z&amp;name=02000000076000101&amp;size=115665012</playbackURI>// This node contains record start time, end time, video name, video size and other information, you need to input this value when download video.
            <metadataDescriptor>motion</metadataDescriptor>//record type: timing-timing record, motion-motion detection record, motionOrAlarm- motion detection or alarm, motionAndAlarm- motion detection and alarm, manual-manual recording,
            smart- intelligient
        </searchMatchItem>
    </matchList>
</CMSearchResult>
```

5.2.9. Get voice intercom channel

Function: I_GetAudioInfo (szDeviceIdentify, options)

Instruction: get voice intercom channel information

Parameters: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

async http interactive way, true: asynchronous，false: synchronous

success successful callback function, there is one parameter that represents the content of XML.

error failed callback function, there are two parameters, the first is http status code, and the second is the XML that devices return(may be null)

Return value: none

Note: The successful callback function will be called when interact successfully; otherwise, failed callback function will be called. The first parameter of callback function is the XML of devices information.

XML format is as follows:

```
<TwoWayAudioChannelList>
    <TwoWayAudioChannel>
        <id></id> //Channel ID
        <enabled></enabled> //whether to enable
        <audioCompressionType></audioCompressionType>  //audio codec
    </TwoWayAudioChannel>
</TwoWayAudioChannelList>
```

5.2.10. Get port

Function: I_GetDevicePort (szDeviceIdentify)

Instruction: get port

Parameters: szDeviceIdentify device identity（IP_Port）

Return value: object-success, null-failure

5.3. Play and play control

5.3.1. Start realplay

Function: I_StartRealPlay (szDeviceIdentify, options)

Instruction: start realplay

Parameters: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

iWndIndex	Play window, if you do not input this parameter, current window will be chosen by default (the default window is 0)
iStreamType	code stream type: 1-main stream, 2-sub stream, main stream is adopted to preview by default.
iChannelID	play channel number: the default channel is 1.
bZeroChannel	whether to play zero channel, and the default is false.
iPort	RTSP port number: if you do not input this parameter, the RTSP port will be detected automatically by development kit.
success	successful callback function.
error	failed callback function.

Return value: none

Note: this function can’t be called until you login.

5.3.2. Start playback

Function: I_StartPlayback (szDeviceIdentify, options)

Instruction: start playback

Parameters: szDeviceIdentify device identity（IP_Port）

options optional parameter objects:

iWndIndex	Play window, if you do not input this parameter, current window will be chosen by default (the default window is 0)
szStartTime	start time; the default time is the very day 00:00:00, format is: 2013-12-23 00:00:00
szEndTime	end time, the default time is the very day 23:59:59, format is: 2013-12-23 23:59:59
iChannelID	play channel number, the default channel

is 1.

iPort RTSP port number: if you do not input this parameter, the RTSP port will be detected automatically by development kit.

oTransCodeParam Transcoding playback parameters object, if this parameter is input, transcoding playback will be executed according to this object(transcoding playback need support by device, if not support, this parameter doesn’t need to be input)

iStreamType code stream type: 1-main stream, 2-sub stream, main stream is adopted to preview by default.

success successful callback function.

error failed callback function.

Return value: none

Note: this interface is for playback by time. Playback by time is supported by development kit, but playback by files is not supported at present. You can search out the video, and then playback by start time and end time.

oTransCodeParam is a javascript object:

```
{
    TransFrameRate: "16",
    TransResolution: "2",
    TransBitrate: "23"
 }
```

TransFrameRate: Frame rate

Value range：0-all， 5-1，6-2，7-4，8-6，9-8，10-10，11-12，12-16，13-20，14-15，15-18，16—22， 255-Auto

TransResolution: Resolution

Value range: 1-CIF(352*288/352*240)，2-QCIF(176*144/176*120)，3-4CIF(704*576/704*480) or D1(720*576/720*486)，255-Auto(Using current resolution)

TransBitrate: Bit rate

Value range：2-32K，3-48k，4-64K，5-80K，6-96K，7-128K，8-160k，9-192K，10-224K，11-256K，12-320K，13-384K，14-448K，15-512K，16-640K，17-768K，18-896K，19-1024K，20-1280K，21-1536K，22-1792K，23-2048K，24-3072K，25-4096K，26-8192K，255-Auto

5.3.3. Start reverse playback

Function: I_ReversePlayback (szDeviceIdentify, options)

Instruction: start reverse playback

Parameters: szDeviceIdentify device identity（IP_Port）

options optional parameters objects:

iWndIndex	Play window number, if you do not input this parameter, current window will be chosen by default (the default window is 0)
szStartTime	start time; the default time is the very day 00:00:00, format is: 2013-12-23 00:00:00
szEndTime	end time, the default time is the very day 23:59:59, format is: 2013-12-23 23:59:59
iChannelID	play channel number, the default channel is 1.
iPort	RTSP port number: if you do not input this parameter, the RTSP port will be detected automatically by development kit.
iStreamType	code stream type: 1-main stream, 2-sub stream, main stream is adopted to preview by default.

Return value: 0-success, -1-failure

Note: reverse playback starts from end time. Many devices can’t support reverse playback at present, if the interface is called, it will return failure.

5.3.4. Stop play

Function: I_Stop (options)

Instruction: stop play (stop realplay and stop playback will call this function):

Parameters: options optional parameters objects:

iWndIndex	play window number: it is optional to input or not input this parameter. It represents the current selected window.
success	successful callback function.
error	failed callback function.

Return value: none

5.3.5. Single frame

Function: I_Frame (options)

Instruction: single-frame play: one frame will be played when this function is called one time. It can be called when playback and reverse playback.

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function.

error failed callback function.

Return value: none

5.3.6. Pause

Function: I_Pause (options)

Instruction: Pause: it can be called when playback and reverse playback.

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function.

error failed callback function.

Return value: none

5.3.7. Resume Play

Function： I_Resume (options)

Instruction: Resume Play: resume playing status from the single-frame / Pause to normal playback

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function.

error failed callback function.

Return value: none

5.3.8. Slow forward

Function： I_PlaySlow (options)

Instruction： Slow forward: the playing speed will reduce one level when you call this interface one time, this plugin supports maximum 1/8 speed, and the device itself may also have restrictions.

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function.

error failed callback function.

Return value: none

5.3.9. Fast forward

Function： I_PlayFast (options)

Instruction：Fast forward, the playing speed will increase one level when you call this interface one time, this plugin supports maximum 8x speed, and the device itself may also have restrictions.

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function.

error failed callback function.

Return value: none

5.3.10. Get OSD time

Function： I_GetOSDTime (options)

Instruction：Get the OSD time of current code stream to realize playback progress

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function. there is one parameter that represents the osd time.

error failed callback function.

Return value: none

5.3.11. Enable sound

Function： I_OpenSound (iWndIndex)

Instruction：enable sound

Parameters: iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.

Return value: 0-success, -1-failure

5.3.12. Disable sound

Function： I_CloseSound (iWndIndex)

Instruction：disable sound

Parameters: iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.

Return value: 0-success, -1-failure

5.3.13. Set volume

Function： I_SetVolume (iVolume, iWndIndex)

Instruction：Set the volume, the volume range :0-100

Parameters: iVolume volume

iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.

Return value: 0-success, -1-failure

5.3.14. Capture pictures

Function： I_CapturePic (szPicName, options)

Instruction：Capture preview / playback picture, and save it to the local PC, the storage path will be displayed in local configuration

Parameters: szPicName file name of picture

options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

bDateDir Create date directory or not (true：create, false：not create), default is true.

Return value: 0-success, -1-failure

Note: The format of captured picture is related to the file name that input when call interface. If the suffix is with '.bmp', then capture bmp picture, and if there is no suffix, the format will be jpg. You can call I_GetLocalCfg() to get the saving path of picture.

5.3.15. Get image resource binary data without plugin

Function： I2_CapturePicData(options)

Instruction：Get image resource binary data without plugin

Parameters: options Optional parameter object

iWndIndex Play window number, allow do not pass, indicating the current selected window

Return value: Successfully returned image resource binary data, failed to return error value

Note: Only support bmp format

5.3.16. Split screen

Function： I_ChangeWndNum (iWndType)

Instruction：modify the type of split screen

Parameters: iWndType the type includes:1-1*1，2-2*2，3-3*3，4-4*4

Return value: 0-success, -1-failure

5.4. Recording

5.4.1. Start recording

Function： I_StartRecord (szFileName, options)

Instruction：save video to the local PC, and the storage path will be displayed in local configuration

Parameters: szFileName recording files name

options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

bDateDir Create date directory or not (true：create, false：not create), default is true.

success successful callback function.

error failed callback function.

Return value: none

5.4.2. Stop recording

Function： I_StopRecord (options)

Instruction：stop recording

Parameters: options optional parameters objects:

iWndIndex play window number: it is optional to input or not input this parameter. It represents the current selected window.

success successful callback function.

error failed callback function.

Return value: none

5.5. Download record

5.5.1. Start downloading

Function： I_StartDownloadRecord (szDeviceIdentify, szPlaybackURI, szFileName, options)

Instruction：The records stored in devices will be downloaded when this function is called.

Parameters: szDeviceIdentify device identity（IP_Port）

szPlaybackURI Record URL, this URL can be obtained by searching record.

szFileName the name of files that need downloading

options optional parameters objects:

bDateDir Create date directory or not (true：create, false：not create), default is true.

Return value: It will return a downloading ID that greater than or equal to 0 when success; otherwise, return -1.(No plugin direct download)

5.5.2. Start downloading by time

Function： I_StartDownloadRecordByTime(szDeviceIdentify, szPlaybackURI, szFileName, szStartTime,szEndTime, options)

Instruction：The records stored in devices will be downloaded when this function is called.

Parameters: szDeviceIdentify device identity（IP_Port）

szPlaybackURI Record URL, this URL can be obtained by searching record.

szFileName the name of files that need downloading

szStartTime Recording start time

szEndTime Recording end time

options optional parameters objects:

bDateDir Create date directory or not (true：create, false：not create), default is true.

Return value: It will return a downloading ID that greater than or equal to 0 when success; otherwise, return -1(No plugin direct download)

5.5.3. Get the records downloading status

Function： I_GetDownloadStatus (iDownloadID)
Instruction：Get the records downloading status to determine whether the download is in progress.
Parameters: iDownloadID Download ID: the return value of ‘Start downloading’.
Return value: it returns 0(stands for downloading is in progress) when success; otherwise, it returns -1 (stands for download failure)

5.5.4. Get the records downloading progress

Function： I_GetDownloadProgress (iDownloadID)
Instruction：Get the downloading progress
Parameters: iDownloadID download ID: the return value of ‘Start downloading’.
Return value: it will return a downloading progress value that greater than or equal to 0.

5.5.5. Stop downloading records

Function： I_StopDownloadRecord (iDownloadID)
Instruction：stop downloading record
Parameters: iDownloadID download ID: the return value of ‘Start downloading’.
Return value: 0-success, -1-failure

5.6. Voice intercom

5.6.1. Start voice intercom

Function： I_StartVoiceTalk (szDeviceIdentify, iAudioChannel)
Instruction：start voice intercom

Parameters: szDeviceIdentify	device IP address
iAudioChannel	voice intercom channel

Return value: 0-success, -1-failure

5.6.2. Stop voice intercom

Function： I_StopVoiceTalk ()
Instruction：Stop voice intercom
Parameters: none
Return value: 0-success, -1-failure

5.7. PTZ control

5.7.1. PTZ direction control

Function： I_PTZControl(iPTZIndex, bStop, options)
Instruction：PTZ direction control
Parameters: iPTZIndex operation type(1-up, 2-down, 3-left, 4-right, 5:up-left, 6:down-left, 7:up-right, 8:down-right, 9-auto, 10-zoom+,11-zoom-,12-focus+,13-focus-,14-aperture+,15-aperture-)
bStop Whether to stop the operation of iPTZIndex, true|false
options optional parameter objects
iWndIndex window number, the default window is the selected by default.
iPTZSpeed PTZ speed, the default is 4
Return value: none

5.7.2. Setting preset

Function： I_SetPreset (iPresetID, options)
Instruction：Set preset
Parameters: iPresetID Preset ID
options optional parameter objects
iWndIndex window number, the default window is the selected by default
Return value: none

5.7.3. Calling preset

Function： I_GoPreset (iPresetID, options)
Instruction：
Parameters: iPresetID preset ID
options optional parameter objects
iWndIndex window number，the default window is the selected by default
Return value: none

5.8. Enlarging image

5.8.1. Enable electronic zoom

Function： I_EnableEZoom (iWndIndex)
Instruction：enable electronic zoom
Parameters: iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.
Return value: 0-success, -1-failure

5.8.2. Disable electronic zoom

Function： I_DisableEZoom (iWndIndex)
Instruction：disable electronic zoom
Parameters: iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.
Return value: 0-success, -1-failure

5.8.3. Enable 3D zoom

Function： I_Enable3DZoom (iWndIndex)
Instruction：enable 3D zoom
Parameters: iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.
Return value: 0-success, -1-failure

5.8.4. Disable 3D zoom

Function： I_Disable3DZoom (iWndIndex)
Instruction：disable 3D zoom
Parameters: iWndIndex Play window number: if you do not input this parameter, current window will be chosen by default.
Return value: 0-success, -1-failure

5.8.5. Full-screen play

Function： I_FullScreen (bFull)
Instruction：full-screen play
Parameters: bFull full screen or not: true-full screen, false-quit full screen
Return value: none

5.9. Device maintaince

5.9.1. Export device configuring parameters

Function： I_ExportDeviceConfig (szDeviceIdentify)

Instruction： export the configuring parameters of devices: this interface will automatically pop up the path selection box

Parameters: szDeviceIdentify device identity（IP_Port）

Return value: 0-success, -1-failure

5.9.2. Import device configuring parameters

Function： I_ImportDeviceConfig (szDeviceIdentify, szFileName)

Instruction：import the configuring parameters: this interface will automatically pop up file selection box. After you import the configuring parameters, devices may restart.

Parameters: szDeviceIdentify device identity（IP_Port）
szFileName the path of configuring files

Return value: 0-success, -1-failure

5.9.3. Restore the default parameters

Function： I_RestoreDefault(szDeviceIdentify, szMode,options)

Instruction： Restore the default parameters

Parameters: szDeviceIdentify device identity（IP_Port）
szMode restoring type basic-simply restore, full-fully restore
options optional parameter objects
success: success function: it has one parameter that represents the contents of the returned XML.
Error: failure function: it has two parameters, and the first one is http status code, the second is the returned XML of devices (may be null).

Return value: none

Note: devices need rebooting after restore the default parameters. All users' information will be restored to default if you choose fully restore.

5.9.4. Restart

Function： I_Restart (szDeviceIdentify, options)

Instruction：restart

Parameters: szDeviceIdentify device identity（IP_Port） options optional parameter objects

success:success function: it has one parameter that represents the contents of the returned XML.

Error: failure function: it has two parameters, and the first one is http status code, the second is the returned XML of devices (may be null).

Return value: none

Note: success only stands for the device is restarted.

5.9.5. Start upgrading

Function： I_StartUpgrade (szDeviceIdentify, szFileName)

Instruction：Start upgrading; the device will restart after upgrade

Parameters: szDeviceIdentify device identity（IP_Port）

szFileName the path of upgrading files

Return value: 0-success, -1-failure

5.9.6. Start asynchronous upgrade

Function： I2_StartUpgrade(szDeviceIdentify, szFileName)

Instruction：Start asynchronous upgrading; the device will restart after upgrade

Parameters: szDeviceIdentify device identity（IP_Port）

szFileName the path of upgrading files

Return value: Successfully entered the callback, failed to enter the failure callback

5.9.7. Get upgrading status

Function： I_UpgradeStatus ()

Instruction：Get upgrading status to determine whether upgrading is in progress.

Parameters: none

Return value: It returns 0 when success, which represents upgrade is in progress. Otherwise, it returns -1, which stands for upgrade failure.

5.9.8. Get upgrading progress

Function： I_UpgradeProgress ()

Instruction：Get upgrading progress

Parameters: none

Return value: It will return upgrade ID that greater than or equal to 0 when success; otherwise, return -1.

5.9.9. Stop upgrading

Function：　I_StopUpgrade ()
Instruction：stop upgrading
Parameters: none
Return value: 0-success, -1-failure

5.9.10. Reconnect

Function：　I_Reconnect (szDeviceIdentify, options)
Instruction：reconnect
Parameters: szDeviceIdentify　device indentity（IP_Port）
options　optional parameter objects
success:success function: it has one parameter that represents the contents of the returned XML.
Error:　failure function: it has two parameters, and the first one is http status code, the second is the returned XML of devices (may be null).
Return value: none

5.9.11. Open remote configuration

Function：　I_RemoteConfig (szDeviceIdentify, options)
Instruction：open remote configuration
Parameters: szDeviceIdentify　device identity（IP_Port）
options　optional parameter objects：
iDevicePort: the SDK port of devices, if you do not input this parameter, the plugin will obtain it automatically from devices.
iLan：　Configure lib language remotely, 0: English, 1: Chinese. The default is English.
Return value: 0-success, -1-failure

5.10. Plugin information maintaince

5.10.1. Plugin version comparison

Function：　I_CheckPluginVersion ()
Instruction：　Plugin version comparison: it can detect whether the plugin is installed before the plug-in is embedded

Parameters: none
Return value: -2: no plugins -1:need upgrading 0: no need upgrading

5.10.2. Get the local configuring parameters

Function： I_GetLocalCfg ()
Instruction： Get the local configuring parameters
Parameters: none
Return value: return the local configuring parameters of plugin (XML format)
Note: the format is as follows:

```
<LocalConfigInfo>
        < ProtocolType</ProtocolType>    //Protocol type: 0-TCP 2-UDP
        <PackgeSize></PackgeSize>     //record package size:0-256M 1-512M 2-1G
        <PlayWndType></PlayWndType> //play window type: 0-full 1-4:3 2-16:9
        <BuffNumberType></BuffNumberType>    // Play Library Buffer Size
        <RecordPath> </RecordPath>     //the saving path of record
        <CapturePath> </CapturePath>   //the saving path of captured pictures
during live view
       <PlaybackFilePath> </PlaybackFilePath> //the saving path of playback video
        <DeviceCapturePath> </DeviceCapturePath>//the saving path of captured
files
        <PlaybackPicPath> </PlaybackPicPath> //the saving path of captured
 pictures during playback
        <DownloadPath> </DownloadPath> //the saving path of downloaded files
during playback
        <IVSMode></IVSMode> //whether to enable rules
        <CaptureFileFormat></CaptureFileFormat> //the capture pictures' format
</LocalConfigInfo>
```

5.10.3. Set the local configuration of plugins

Function： I_SetLocalCfg ()
Instruction： set the local configuration parameters
Parameters: szLocalCofing the string of local configuration
Return value: 0:success -1:failure

5.10.4. Get playing window status

Function： I_GetWindowStatus (iWndIndex)
Instruction：get the current window status
Parameters: iWndIndex window index
Return value: return window status if success; otherwise it returns null.

Note：window object info：

iIndex window index

szIP the IP address of device that is playing in window

iChannelID the channel ID that is playing in window

iPlayStatus the playing status of window(0-no play,1-live view, 2-playback, 3-pause, 4-single frame, 5-reverse playback, 6-pause when reverse playback)

5.11. Draw polygon in window

5.11.1. Set play mode

Function： I_SetPlayModeType(iMode)

Instruction： set play mode

Parameters: iMode play mode, 0 preview mode(disable polygon), 6 polygon Mode

Return value: 0-success, -1-failure

Note: Preview mode doesn't support drawing polygon, and polygon mode means enable drawing polygon.

5.11.2. Set draw mode

Function： I_SetSnapDrawMode(iWndIndex, iMode)

Instruction： Set draw mode

Parameters: iWndIndex Play window number

iMode draw mode: -1-stop drawing, 2-add polygon, 3-edit polygon

Return value: 0-success, -1-failure

5.11.3. Set polygon information

Function： I_SetSnapPolygonInfo(iWndIndex, szInfo)

Instruction： set polygon information

Parameters: iWndIndex Play window number

szInfo XML, one or several polygons

Return value: 0-success, -1-failure, -2-error parameters, -3-polygon number reaches the upper limit, -4-polygon id exists

5.11.4. Set polygon

Function： I2_SetSnapPolygonInfo(iWndIndex, szInfo)

Instruction： set polygon

Parameters: iWndIndex Play window number szInfo XML, one or several polygons

Return value: 0-success, -1-failure

5.11.5. Get polygon information

Function： I_SetSnapPolygonInfo(iWndIndex)

Instruction： get polygon information

Parameters: iWndIndex Play window number

szInfo XML, one or several polygons

Return value: success：XML data, failure：null

5.11.6. Clear polygon information

Function： I_ClearSnapInfo(iWndIndex)

Instruction： clear polygon information

Parameters: iWndIndex Play window number

Return value: 0-success, -1-failure

5.12. Others

5.12.1. Folder selection and files' path

Function： I_OpenFileDlg (iType)

Instruction： Open folders or files' path

Parameters: iType Type: 1-folder, 2-file

Return value: It will return the selected foleder or files' path

5.12.2. Asynchronously folder selection and files' path

Function： I2_OpenFileDlg (iType)

Instruction： Asynchronously open folders or files' path

Parameters: iType Type: 1-folder, 2-file

Return value: It will asynchronously return the selected foleder or files' path

5.12.3. Get the last error code

Function： I_GetLastError ()

Instruction：	get the last error code
Parameters:	none
Return value:	It will return last error code.

5.12.4. Send HTTP Request

Function：	I_SendHTTPRequest(szDeviceIdentify, szURI, options)
Instruction：	send HTTP request
Parameters：	szDeviceIdentify	device indentity(IP_Port)
szURI	ISAPI/PSIA protocol
options	optional parameters
async	synchronous or not (true: asynchronous, false: synchronous), asynchronous by default.
type	GET,POST,PUT,DELETE, default is GET.
data	xml, null by default.
auth	authentication info, the current device authentication info by default.
success	have one parameter, represents the returned XML info.
error	have two parameters, the 1st parameter is http status code; the 2nd parameter is returned xml(may be null)

Return value: none
Note: Login first before using the interface.

5.12.5. Set encapsulation format

Function：	I_SetPackageType()
Instruction：	set encapsulation format
Parameters：	iPackageType	encapsulation format, 2-PS format	11-MP4 format
Return value: 0-success, -1-failure
Note: The record video and video clips will be encapsulated as such format.

5.12.6. Capture picture

Function：	I_ DeviceCapturePic(szDeviceIdentify, iChannelID, szPicName, options)
Instruction：	Capture picture without starting preview
Parameters：	szDeviceIdentify	Device identity(IP_Port)
iChannelID	channel number
szPicName	file name of picture
options	optional parameters
bDateDir	create date directory or not(true: create, false：not create), default value is true.

Return value: 0-success, -1-failure