

1	Scope	4
2	References	4
3	Version update records	5
4	Definitions and abbreviations	7
4.1	Definitions	7
4.2	Abbreviations	7
5	Architecture and Transmission Mechanism	7
5.1	REST and HTTP Methods	7
5.2	XML	8
5.3	Resources overview	8
5.4	Protocol URL	8
5.5	Messages	9
5.5.1	Connection Header Field	9
5.5.2	Authorization and WWW-Authenticate Header Fields	9
5.5.3	Entity Body	10
5.5.4	Operations	10
6	Device discovery	12
7	Resource Description	12
7.1.1	Resource Description Outline	12
7.1.2	Annotation	13
8	Error Handling	13
9	Services and General Resources	17
9.1	Network	17
9.1.1	Network/PPPOE	17
9.1.1.1	Network/PPPOE/Config	18
9.1.1.2	Network/PPPOE/Status	19
9.1.2	Network/DDNS	19
9.1.2.1	Network/DDNS/ServerList	20
9.1.2.2	Network/DDNS/Config	21
9.1.2.3	Network/DDNS/Status	22
9.1.3	Network/UPNP	22
9.1.3.1	Network/UPNP/Config	23
9.1.3.2	Network/UPNP/Status	24
9.1.4	Network/Email	24
9.1.5	Network/Port	26
9.1.6	Network/P2P	27
9.1.7	Network/P2PV2	27
9.1.8	Network/FTP	28
9.1.9	Network/WifiParamTest	29
9.1.10	Network/WifiParamTestStatus	30
9.1.11	Network/WhichNetworkCard	30
9.1.12	Network/Interfaces	31
9.1.12.1	Network/Interfaces/NetworkCardID	32
9.2	System	38
9.2.1	System/Time	38
9.2.2	System/NTP	39
9.2.3	System/DST	39
9.2.4	System/DeviceInfo	40
9.2.5	System/DeviceStatus	41
9.2.5.1	System/DeviceStatus/ChannelID	42
9.2.6	System/FirmwareUpgrade	43
9.2.7	System/Reboot	43
9.2.8	System/FactoryDefault	43
9.2.9	System/FactoryDefaultV2	44
9.2.10	System/SendTestEmail	44

9.2.11	System/RecoveryAdminPassword	44
9.2.12	System/TryListenCurrentAlarmAudio	45
9.2.13	System/Logging	45
9.2.14	System/Users	47
9.2.15	System/Users/ID	48
9.2.16	System/ChannelID/RemoteForceIframe	49
9.2.17	System/ChannelID/RemoteAlarmOutClear/ID	49
9.2.18	System/ChannelID/RemoteAlarmoutControl/on	50
9.2.19	System/ChannelID/RemoteAlarmoutControl/off	50
9.2.20	System/AlarmoutState	50
9.2.21	System/AlarmInState	51
9.2.22	System/DeviceConfigFile	51
9.2.23	System/TransparentChannel/WriteOnce	52
9.2.24	System/TransparentChannel/Connect	52
9.2.25	System/TransparentChannel/Write	53
9.2.26	System/TransparentChannel/Read	54
9.2.27	System/TransparentChannel/Disconnect	54
9.2.28	System/DeviceAutoMaintenance	55
9.2.29	System/SendTestFTP	55
9.2.30	System/DeviceWifiSTA2AP	56
9.2.31	System/DeviceCustomFunction	56
9.2.32	System/CloudUpgradeServerInfo	56
9.2.33	System/CloudUpgradeFirmwareList	57
9.2.34	System/CloudUpgradeDownloadStart	57
9.2.35	System/CloudUpgradeDownloadStatus	58
9.2.36	System/CloudUpgradeDownloadTerminate	58
9.2.37	System/CloudUpgradeStartUpgrade	59
9.2.38	System/CloudUpgradeResult	59
9.2.39	System/ErrorInfoMap	60
9.2.40	System/DeviceCap	60
9.2.41	System/AudioAlarmConfig	62
9.2.42	System/TryListenSelectedAlarmAudio	62
9.2.43	System/SaveHardwareConfig	63
9.2.44	System/CancelHardwareConfig	63
9.3	Security	63
9.3.1	Security/UserAuth	63
9.4	Streams	64
9.4.1	Streams/ChannelID	64
9.4.1.1	Streams/ChannelID/ReceiveData	65
9.4.1.2	Streams/ChannelID/CapabilityV2	65
9.4.1.3	Streams/ChannelID/StreamID	67
9.4.2	Streams/AudioStream	69
9.5	Snapshot	70
9.5.1	Snapshot/ChannelID/RemoteImageCapture	70
9.5.2	Snapshot/ChannelID/StreamID/RemoteImageCaptureV2	70
9.5.3	Snapshot/Capability	71
9.5.4	Snapshot/Config	71
9.6	Images	72
9.6.1	Images/ChannelID	72
9.6.1.1	Images/ChannelID/Scene	73
9.6.1.2	Images/ChannelID/Basic	74
9.6.1.3	Images/ChannelID/Color	75
9.6.1.4	Images/ChannelID/Exposure	75
9.6.1.5	Images/ChannelID/WhiteBalance	76
9.6.1.6	Images/ChannelID/IrCutFilter	77

9.7	Pictures	78
9.7.1	Pictures/ChannelID.....	78
9.7.1.1	Pictures/ChannelID/OSD	79
9.7.1.2	Pictures/ChannelID/VideoLost	80
9.7.1.3	Pictures/ChannelID/Motion.....	83
9.7.1.4	Pictures/ChannelID/Mosaic	89
9.8	PTZ.....	90
9.8.1	PTZ/ChannelID/Presets/Set.....	90
9.8.2	PTZ/ChannelID/Presets/Remove	90
9.8.3	PTZ/ChannelID/Presets/Goto	91
9.8.4	PTZ/ChannelID/Cruise/StartCruiseMem.....	91
9.8.5	PTZ/ChannelID/Cruise/StopCruiseMem	92
9.8.6	PTZ/ChannelID/Cruise/ClearCruise	93
9.8.7	PTZ/ChannelID/Cruise/AddPresetToCruise.....	93
9.8.8	PTZ/ChannelID/Cruise/StartCruise.....	94
9.8.9	PTZ/ChannelID/Cruise/StopCruise	94
9.8.10	PTZ/ChannelID/Track/StartTrackMem.....	95
9.8.11	PTZ/ChannelID/Track/StopTrackMem	95
9.8.12	PTZ/ChannelID/Track/StartTrack.....	96
9.8.13	PTZ/ChannelID/Track/StopTrack	96
9.8.14	PTZ/ChannelID/FocusFar.....	97
9.8.15	PTZ/ChannelID/FocusNear	97
9.8.16	PTZ/ChannelID/ApertureBigger	98
9.8.17	PTZ/ChannelID/ApertureSmaller	99
9.8.18	PTZ/ChannelID/ZoomIn	99
9.8.19	PTZ/ChannelID/ZoomOut.....	100
9.8.20	PTZ/ChannelID/TurnLeft	100
9.8.21	PTZ/ChannelID/TurnRight	101
9.8.22	PTZ/ChannelID/TurnUp.....	101
9.8.23	PTZ/ChannelID/TurnDown	102
9.8.24	PTZ/ChannelID/TurnUpLeft.....	103
9.8.25	PTZ/ChannelID/TurnUpRight	103
9.8.26	PTZ/ChannelID/TurnDownLeft.....	104
9.8.27	PTZ/ChannelID/TurnDownRight.....	104
9.8.28	PTZ/ChannelID/AutoLeftRight	105
9.8.29	PTZ/ChannelID/Auxiliary	105
9.8.30	PTZ/ChannelID/Light.....	106
9.8.31	PTZ/ChannelID/RainBrush.....	106
9.8.32	PTZ/ChannelID/Fan.....	107
9.8.33	PTZ/ChannelID/Hearter	108
9.8.34	PTZ/ChannelID/Config	108
9.8.1	PTZ/ChannelID/AdvanceConfig.....	109
9.8.2	PTZ/ChannelID/Watch/StartWatch	110
9.8.3	PTZ/ChannelID/Watch/StopWatch	110
9.9	TwowayAudio	111
9.9.1	TwowayAudio/Open	111
9.9.2	TwowayAudio/Close	111
9.9.3	TwowayAudio/SendData.....	112
9.9.4	TwowayAudio/ReceiveData.....	112
9.10	EventsNotification.....	113
9.10.1	EventsNotification/SubscribedEventTypeList.....	113
9.10.2	EventsNotification/Notify.....	114
9.10.3	EventsNotification/EventList	115
9.11	Alarm	115
9.11.1	Alarm/AlarmIn	115

9.11.1.1	Alarm/AlarmIn/ChannelID.....	116
9.11.2	Alarm/AlarmOut.....	118
9.11.2.1	Alarm/AlarmOut/ChannelID.....	118
9.11.3	Alarm/AMS.....	119
9.11.4	Alarm/EventCfgList.....	119
9.12	Record.....	120
9.12.1	Record/HttpEvent/Config.....	121
9.12.2	Record/HttpEvent/Call.....	121
9.12.3	Record/Format/Call.....	121
9.12.4	Record/Notify/URL.....	122
9.12.5	Record/ChannelID/RecordSchedule.....	122
9.13	Disk.....	124
9.14	Playback.....	124
9.14.1	Playback/GetRecordInfoByMonth.....	124
9.14.2	Playback/GetRecordInfoByDay.....	125
9.15	RTMP.....	126
9.15.1	RTMP/BasicConfig.....	126
9.15.2	RTMP/ChannelID/StreamID/Config.....	126
9.16	RecordFileList.....	127
9.16.1	RecordFileList/Attribute.....	127
9.16.2	RecordFileList/Pages/PageNum.....	128
9.17	SnapshotFileList.....	129
9.17.1	SnapshotFileList/Attribute.....	129
9.17.2	SnapshotFileList/Pages/PageNum.....	129
10	Device discovery V2.....	130

1 Scope

This specification defines a HTTP-based application programming interface that enables physical security and video management systems to communicate with IP media devices in a particular way.

With regard to Media Streaming, please refer to —develop API of RTSP protocoll.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] RFC2616 Hypertext Transfer Protocol-HTTP/1.1

[2] W3C XML 1.0 specification

[3] W3C Character encodings

3 Version update records

Modify Date	Version	Modification
2016/04/22	1.1.9	New,first published
2016/05/31	1.1.10	1.Add the url relation with alarm. 2.Add the url relation with record. 3.Add the section " StorageDirectory " of the ftp. 4.Add the url "/Streams/ChannelID/ReceiveData". 5.Add the url "/Streams/ChannelID/StreamID/ReceiveData"
2016/10/08	1.1.11	Add the URL "/Network/WifiParamTest"
2016/10/14	1.1.12	1.Add the url "/Network/WifiParamTestStatus". 2.Add the url "/Network/WhichNetworkCard".
2016/11/01	1.1.13	1.Add the url "/Record/Notify/URL". 2.Add the url "/Record/ChannelID/RecordSchedule"
2016/11/08	1.1.14	Add the url "/Pictures/ChannelID/Motion/RegionV2"
2016/11/14	1.1.15	1.Add the url "/Playback/GetRecordInfoByMonth" 2.Add the url "/Playback/GetRecordInfoByDay"
2016/12/21	1.1.15	1.Update the url "/Network/Email". 2.Update the url "/Network/FTP". 3.Update the url "/Images/ChannelID/Scene". 4.Update the url "/Images/ChannelID/Exposure" 5.Update the url "System/NTP"
2017/01/09	1.1.17	Add the url "/System/SendTestFTP"
2017/01/17	1.1.18	1.Add the url "System/DeviceWifiSTA2AP". 2.Update the url "/System/DeviceWifiSTA2AP". 3.Update the url "/Network". 4.Update the url "Streams/AudioStream". 5.Update the url "Record".
2017/02/14	1.1.19	Add the url "/System/FactoryDefaultV2"

2017/04/14	1.1.20	1.Modify the error position. 2.Add the url “/System/DeviceCustomFunction”. 3.Add the url “/Network/P2PV2”.
2017/05/04	1.1.21	1.Updata the url /Record 2.Update the url /Record/ChannelID/RecordSchedule
2017/06/28	1.1.22	1,Add a new field for WPA password encoded by Base64.
2017/07/12	1.1.23	1.Add URL “RTMP/BasicConfig” for RTMP basic configuration. 2.Add URL “/RTMP/ChannelID/StreamID/Config” for RTMP stream configuration. 3.Add RTMP capability filed in URL “/System/DeviceInfo”.
2017/07/27	1.2.0	1,Add h265+ capability and codec setting.
2017/08/04	1.2.1	1.Add the url “/Pictures/ChannelID/Motion/Notify/URL”
2017/08/11	1.3.0	Add h264+ capability and codec setting.
2017/11/10	1.3.1	1.Add the url “/Alarm/AMS” 2.Add the url “Alarm/EventCfgList”
2018/05/04	1.3.2	1. Add the the volume of audio out(/Streams/AudioStream). 2. Add the url “/System/CloudUpgradeServerInfo” 3. Add the url “/System/CloudUpgradeFirmwareList” 4. Add the url “/System/CloudUpgradeDownloadStart” 5. Add the url “/System/CloudUpgradeDownloadTerminate” 6. Add the url “/System/CloudUpgradeDownloadStatus” 7. Add the url “/System/CloudUpgradeStartUpgrade” 8. Add the url “/System/CloudUpgradeResult” 9. Add the smart color function(/Images/ChannelID/IrCutFilter). 10. Add the url “/RecordFileList/Attribute” 11. Add the url “/RecordFileList/Pages/(PageNum)” 12. Add the url “/System/ErrorInfoMap” 13. Add the url “/System/DeviceCap” 14. Old version only support for sensitivity of highest,higher,lower and lowest,some version support for sensitivity of highest, higher,high, middle,low, lower and lowest(/Pictures/1/Motion). 15. Modified the problem of getting the failure of the MJPEG stream(/Streams/ChannelID/StreamTypeID/ReceiveData). 16. support for half time zone or 1/4(/System/NTP). 17. Add the url “/SnapshotFileList/Attribute” 18. Add the url “/SnapshotFileList/Pages/(PageNum)”
2018/05/30	1.3.3	1. add the status “ nodisk,exceptionneedformat,exceptionneedChange Disk,formatfailedneedreboot ” int the <DiskStorageAttribute> of the url /Disk 2. add the status “ umount failure,device is busy ” int the <DiskFormatStatus> of the url /Disk 3. add the status “ passwd error ” int the <Status> of the url /Network/WifiParamTestStatus 4. Add the url /Snapshot/ChannelID/StreamID/RemotelImageCaptureV2 5. Modify the url /RecordFileList/Pages/PageNum. 6. Modify the url /SnapshotFileList/Pages/PageNum.
2018/07/10	2.0.0	1. Add the url “/PTZ/ChannelID/TurnUpLeft” 2. Add the url “/PTZ/ChannelID/TurnUpRight” 3. Add the url “/PTZ/ChannelID/TurnDownLeft” 4. Add the url “/PTZ/ChannelID/TurnDownRight” 5. Add the url “/PTZ/ChannelID/AutoLeftRight”
2019/08/14	2.0.6	1. Add the url “/Network/Interfaces/2/WIFIAccessPointListV2”

4 Definitions and abbreviations

4.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

Services: a set of resources consisting of relevant General Resources.

General Resources: physical resources that supported by the devices.

Node: Services and General Resources.

4.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

REST REpresentational State Transfer

UPnP Universal Plug and Play

5 Architecture and Transmission Mechanism

The IP Media Device Management Protocol is based on REST architecture. The management and control interfaces defined in this specification are treated as resources utilizing the REpresentational State Transfer (REST) architecture. This architecture facilitates users by grouping related resources within hierarchical namespaces, and is more flexible for service discovery and future expansion.

REST architecture consists of clients and servers, among which clients initiate request to servers, while servers handle requests and response accordingly. Requests and responses are established via the transmission of —representations|| of —resources||. REST architecture need to be based on an Application Layer protocol which provides various of standard communication formats for applications based on the transfer of meaningful representational state. HTTP[1] has a very rich vocabulary in terms of verbs(or —methods||),URIs, request and response headers, Internet media types, HTTP request and response codes etc. In addition, HTTP also has some features particularly suitable for REST architecture. So HTTP is used as external Application Layer protocol in this specification. In the architecture, clients are physical security and video management systems; servers are IP media devices.

This specification also contains full XML schema for the introduced resources.

5.1 REST and HTTP Methods

The following table shows how HTTP verbs are typically used to implement a web service based on REST architecture.

Resourc	GET	PUT	POST	DELETE
---------	-----	-----	------	--------

Collection URI, such as http://webServer/resources	List the members of collection, complete with their member URIs for further navigation.	Meaning defined as — replace the entire collection with another collection.	Create a new entry in the collection where the ID is assigned automatically by the collection. The ID created is usually included as part of the data returned by this	Meaning defined as — delete the entire collection.
Member URI, such as http://webServer/resources/7416	Retrieve a representation of the addressed member of the collection expressed in an appropriate MIME type.	Update the addressed member of the collection or create it with the specified ID.	Treat the addressed member as a collection in its own right and create a new subordinate of it.	Delete the addressed member of the collection.

5.2 XML

A device must support the syntax defined by W3C XML 1.0 specification [2] and UTF-8 character set [3]. All XML files must adopt UTF-8 encoding according to RFC3629. Additionally, all resources share a common XML schema as defined in Annex.

Any resources can specify separate input and output XML Documents. If a specific data structure is defined inside these documents, then they must be specified as XML Schema Documents (xsd) in Annex.

Lists contained in XML blocks will be represented in the format of <XXXList>, and each <XXXList> tag may contain one or more nodes.

5.3 Resources overview

Services	Description	Reference
Network	Configure network interfaces.	8.1
System	Configure and operate the general system functions.	8.2
Streams	Handle video-related configuration.	8.3
Images	Configure the general Images configuration	8.4
Pictures	Configure the general pictures configuration	8.5

5.4 Protocol URL

The URL scheme is used to locate device resources via a specific protocol in the network.

This section defines the syntax and semantics for http(s) URLs.

protocol: URL scheme for the particular request. The http and https protocols are allowed in this specification.

host: The host field refer to the hostname, IP address, or the FQDN of an IP device.

port: The port field refer to the port number of that host on which the identified resource is located at the IP device listening for TCP connections. If the port is empty or not given, the default port is assumed. For HTTP, the default port 80. For HTTPS, the default port 443.

abs_path: The Request-URI [1] for the resources is abs_path [4]. The abs_path in this specification is most often of the form —[/Services]/[General Resources]/[Special Resources]||, which is suitable for resources to update or restore device configurations. —/DI|| which appears in the abs_path identifies one resource of a list resource in this specification.

query: The query field is a string of information to be interpreted by the resource. It can include some resource-related parameters. It must be listed in name-value pair syntax (p1=v1&p2=v2&...&pn=vn). Each resource can define a set of parameters. Defining input data which is specific to the resource will be prior than query usage.

5.5 Messages

HTTP messages are used for communication between physical security and video management systems and IP media devices in this specification. In order to configure and control the device, some provisions are specified for these HTTP message.

5.5.1 Connection Header Field

Devices that implement HTTP/1.1 should support persistent connections in order to meet video management systems or client applications' requirements that issue multiple HTTP(s) transactions. HTTP/1.1 is implemented and utilized according to RFC 2616 in the IP devices. For a video management system or client application that uses persistent connection for multiple transactions, it is required to implement —Connection: Keep-Alive|| HTTP header field, while also adopt the —Connection: close|| HTTP header field for the last transaction of the persistent connection. This process will assume that the application can identify the last request in a sequence of multiple requests.

5.5.2 Authorization and WWW-Authenticate Header

Fields

When a video management system or client application sends any request to the device, it must be authenticated by means of Basic Access [5] according to RFC 2617, and thus all the devices are required to support Basic Access. Authorization header field is sent along with each request, and if a user is authenticated, the request will follow the normal execution flow. If client HTTP request is with no authentication credentials,

unauthorized HTTP response (401) will be returned with WWW-Authenticate header field.

5.5.3 Entity Body

The Content-Type entity-header field indicates the media type of the entity body. The Content-Type may be designated as `application/xml; charset=UTF-8`, `application/octet-stream`, etc. For configuration information, the Content-Type is usually `application/xml; charset=UTF-8`. For example,

HTTP Request Message

```
GET /System/status HTTP/1.1
...
```

HTTP Response Message

```
HTTP/1.1 200 OK
...

Content-Type: application/xml; charset=UTF-8
...
```

HTTP Request Message

```
PUT /System/configurationData HTTP/1.1
...
Content-Type: application/octet-stream
...
[proprietary configuration file data content ]
```

HTTP Response Message

```
HTTP/1.1 200 OK
...

Content-Type: application/xml; charset=UTF-8
...
<?xml version="1.0" encoding=UTF-8?>
<ResponseStatus>
...
</ResponseStatus>
```

5.5.4 Operations

Different resources will specify different operation.

- The set device configuration resources use PUT operation. If there is an XML block parameter for the request, the inbound XML format is defined according to a resource-special XML schema. Request status will be returned by the XML response information of the device, and can be used for indicating the PUT operation status. The responded XML format is defined by —XML Response Schemall (please refer to section 4.5.5 for detail description). After the device configuration is updated successfully, it will return an XML response with status code —OKII; while another status code will be used for indicating unsuccessful operations. In either case, the device only responses after it is ready to continue normal operation, i.e. accepting streaming request, receiving configuration commands, etc.
- The get device configuration resources use GET operation. After a successful GET operation, the result will be returned in XML format according to the resource description. For an unsuccessful request (i.e. users is not authenticated), the result will be returned in XML format according to —XML Response Schemall.
- Resources to create device configurations information will use the POST operation. If there is an XML block parameter for the request, the inbound XML format is defined according to a resource-special XML schema. The request status will be indicated by the XML response information returned from the device, and can be used to indicate the status of the POST operation. This XML format is defined according to —XML Response Schemall (see section 4.5.5 for details). After successfully creating the data, the device returns an XML response with status code —OKII. A separate status code is used for unsuccessful operations.
- Resources to delete device configurations information will use the Delete operation. If successful, the result will be returned an XML response with status code —OKII. A separate status code is used for unsuccessful operations. This XML format is defined according to —XML Response Schemall (see section 4.5.5 for details).
- Data uploading resources (i.e. firmware upgrade, import configuration, etc.) will use PUT operation. The content of the data will be stored in the body of the HTTP request. If successful, the result will be returned an XML response with status code —OKII. A separate status code is used for unsuccessful operations. This XML format is defined according to —XML Response Schemall (see section 4.5.5 for details).
- Data receiving resources (i.e. export configuration file) use GET operation. If successful, the result will be returned the data according to the resource description. An XML block is used for unsuccessful operations. This XML format is defined according to —XML Response Schemall (see section 4.5.5 for details).
- For Special Resources, GET operation will be used. For more detailed description see Section 6.

If there is an XML block for the HTTP request or response, the Content-Type and Content-Length will be set in the headers of the HTTP message.

6 Device discovery

The IP devices support Universal Plug and Play (UPnP) technology to discovery/locate themselves. A UPnP compatible device will automatically announce its network address, supported devices and services types when connected to a network, and therefore becoming —plug-and-playll by allowing clients recognize those information and begin using this device immediately.if you want to get more information about upnp,please refer to 《Discovery version 1.0.0.docx》

7 Resource Description

7.1.1 Resource Description Outline

Each resource in this specification is defined using the following format.

<i>Resource_URI</i>	<i>Type</i>	<i>Version</i>
<i>Operation_Name</i>		<i>User Lever</i>
Description	<i>Description of the operation.</i>	
Query	<i>Indicates the name/value pairs (p1, p2, p3, ...,pn) for the resource.</i>	
Inbound Data	<i>Indicates inbound data for the resources.</i>	
Success Return	<i>the Type (if present) and the name of XML Data Block</i>	
Notes: <i>describes any special processing rules for the resource.</i>		

Type refers to —Special Resourcell, —Servicell and —General Resourcell.

Version is used to determine the version of the protocol. The version number shall be set to —1.0ll in this specification.

Operation_Name refers to —GETll, —PUTll, —POSTll and —DELETEll.

Inbound Data includes three types as follows:

- NONE –no input data
- DataBlock – the name of an XML Data Block. Datablocks used here must be defined according to the specification.
- Mime type – mime type for the input data in the HTTP payload.
Remark:—application/xmlll is not a valid mime type.

If a device does not support particular XML tags or blocks, then it may not be supported by the resource operations.

Generally, if a field is not provided in the inbound XML, then its current values shall not be

modified in the device's repository.

If a required field did not exist in the device's repository, then it must be provided in the applicable resource operations.

Success Return and Error Return detailed description see Section 8.

7.1.2 Annotation

The XML Data Blocks described in this document contains annotations for the field's properties. Please refer to the XML schema definitions for detail description.

The Following annotation content is inserted into the comments to describe the data carried in the field.

Table 6

Annotation	Description
req	Required field.
opt	Optional field. For data uploaded to the device, if the field is present but the device does not support it, it should be ignored.
dep	This field is required depending on the value of another field.
ro	Read-only. For XML data that is both read and written to the device, this field is only present in XML returned from the device. If this field is present in XML uploaded to the device, it should be ignored.
wo	Write-only. This field is only present in XML that can be uploaded to the device. This field should never be present in data returned from the device. [This is used for uploading passwords].
xs:<type>	A type defined in XML Schema Part 2: Datatypes Second Edition, see http://www.w3.org/TR/xmlschema-2

Remark: optional XML structures may contain required fields for the operation, which mean that even if the entire XML block is optional, some of its contained fields may still be necessary if required.

8 Error Handling

As with any other protocol, errors may occur during communications, protocol or message processing, and the specification classifies error handling into categories below:

- Protocol Errors, which are result of an incorrectly formed protocol message. Protocol Errors may contain header value or be received in an not expected or experience a socket timeout. To indicate and interpret protocol error, HTTP

protocol has defined a set of standard status codes [e.g., 1xx, 2xx, 3xx, 4xx, 5xx]. According to this specification, the IP devices will use appropriate HTTP protocol defined status codes for error reporting and when received handle accordingly.

- Application Errors, which are generated as a result of REST operations errors. All such application errors must be reported and handled through HTTP messages. The following table indicates the mapping relationship between HTTP status codes and REST operations, and also the information contained in response header and bodies.

HTTP Status Codes	REST Meaning	GET	PUT	POST	DELETE
200	—OK—The request has succeeded. Header Notes: None Body Notes: The requested resource will be returned in the body.	√	√		√
201	—Created— The request has created a new resource. Header Notes: The Location header contains the URI of the newly created resource. Body Notes: The response returns an entity describing the newly created resource.		√	√	
204	—No Content— The request succeeded, but there is no data to return. Header Notes: None Body Notes: No body is allowed.		√		√
301	—Moved Permanently— The requested resource has moved permanently. Header Notes: The Location Header contains the URI of the new location. Body Notes: The body may contain the new resource	√			

302	—Foundll – The requested resource should be accessed through this location, but the resource actually lives at another location. This is typically used to set up an alias.	√			
HTTP Status Codes	REST Meaning	GET	PUT	POST	DELETE
	Header Notes: The Location header contains the URI of the resource. Body Notes: The body may contain the new resource				
	—Bad Requestll – The request was badly formed. This is commonly used for creating or updating a resource, but the data was incomplete or incorrect. Header Notes: The				
	—Unauthorizedll – The request requires user authentication to access this resource. If the request contains invalid authentication data, this code is sent.				
403	—Forbiddenll – The request is not allowed because the server is refusing to fill the request. A header suggests to the client when to try resubmitting the	√	√	√	√

When setting the configuration, it will return a XML with an error code. Such as:

ResponseStatus XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<ResponseStatus Version="1.0" xmlns="http://www.zwcloud.wang/ver10/XMLSchema">
  <requestURL>/Network/Interfaces/2</requestURL>
  <statusCode>0</statusCode>
</ResponseStatus>
```

<requestURL> indicates URI operation.

<statusCode> represents the return value by the device.

StatusCode	Description
0	Success
-1	Decode error, maybe the XML and URL do not correspond.
-2	Decode error, maybe the XML missing fields with <req> tag.
-3	Unsupported operation
-4	Decode error, maybe It is associated with the channel
-5	Parameter error
-6	Query log busy
-7	Query log error
-8	call SDK fails
-9	Memory overflow
-10	URL Error
-11	User Overflow
-12	Invalid Operate
-13	User Is Exist

NOTE: For the upgrade, when the return value is greater than zero, indicating that the upgrade fails.

9 Services and General Resources

9.1 /Network

/Network		General Resource v1.0
GET		Viewer
Description	It is used to get the device network configuration	
Query	None	
Inbound Data	None	
Success Return	Network	
PUT		Viewer
Description	It is used to set the device network configuration	
Query	None	
Inbound Data	Network	
Success Return	ResponseStatus	
Notes:		
Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.		

Network XML Block

```
<Network version = "1.0" xmlns = "">
  <PPPOE> <!-- opt -->
  <DDNS> <!-- opt-->
  <UPNP> <!-- opt -->
  <EMail> <!-- opt-->
  <Port> <!-- opt-->
  <P2P> <!-- opt-->
  <Interfaces><!--opt-->
</Network>
```

9.1.1 /Network/PPPOE

/Network/PPPOE		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device PPPOE configuration	
Query	None	
Inbound Data	None	
Success Return	PPPOE	
PUT		Viewer
Description	It is used to set all of the device PPPOE configuration	
Query	None	

Inbound Data	PPPOE
Success Return	ResponseStatus
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.	

PPPOE XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<PPPOE>
  <Config/> <!-- req -->
  <Status /><!--opt-->
</PPPOE>
```

9.1.1.1/Network/PPPOE/Config

/Network/PPPOE/Config		General Resource v1.0
GET		Viewer
Description	It is used to get the readable and writable configuration of the device PPPOE	
Query	None	
Inbound Data	None	
Success Return	Config	
PUT		Viewer
Description	It is used to set the readable and writable configuration of the device PPPOE	
Query	None	
Inbound Data	Config	
Success Return	ResponseStatus	
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s. When it is retrieved the <Config> XML, <Password> field is not displayed ,but if there is not <Password> in the configuration settings, the password is not change, Otherwise, the password will be changed.		

PPPOE configuration XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Config>
  <Enable><!--req,xs:boolean--></Enable>
  <UserName><!--req,xs:string--></UserName>
  <Password><!--wo,opt,xs:string--></Password>
</Config>
```

9.1.1.2 /Network/PPPOE/Status

/Network/PPPOE/Status		General Resource v1.0
GET		Viewer
Description	It is used to get the read-only configuration of the device PPPOE	
Query	None	
Inbound Data	None	
Success Return	Status	
Notes: <p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>When <State> is the value of "connected" , <IPAddress>, <SubnetMask> and <Gateway> will be valid..</p>		

PPPOE Status XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Status>
  <State><!--ro,req,xs:string,connecting,connected,disconnected,disconnecting--></State>
  <IPAddress><!--ro,req,xs:string--></IPAddress>
  <SubnetMask>
    <!-- ro,req, xs:string, subnet mask for IPv4 address -->
  </SubnetMask>
  <Gateway><!--ro,req,xs:string--></Gateway>
</Status>
```

9.1.2 /Network/DDNS

/Network/DDNS		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device DDNS configuration	

Query	None
Inbound Data	None
Success Return	DDNS
PUT Viewer	
Description	It is used to set all of the device DDNS configuration
Query	None
Inbound Data	DDNS
Success Return	ResponseStatus
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.	

DDNS XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<DDNS>
  <ServerList/><!-- opt -->
  <Config/><!-- req -->
  <Status/><!-- opt -->
</DDNS>
```

9.1.2.1 /Network/DDNS/ServerList

/Network/DDNS/ServerList General Resource v1.0	
GET Viewer	
Description	It is used to get the read-only Server List of the device DDNS
Query	None
Inbound Data	None
Success Return	ServerList
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.	

ServerList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<ServerList>
  <Server><!-- opt -->
    <ServerName><!-- req, xs:string --></ServerName>
  </Server>
```

</ServerList>

9.1.2.2/Network/DDNS/Config

/Network/DDNS/Config		General Resource v1.0
GET		Viewer
Description	It is used to get the readable and writable configuration of the device DDNS	
Query	None	
Inbound Data	None	
Success Return	Config	
PUT		Viewer
Description	It is used to Set the readable and writable configuration of the device DDNS	
Query	None	
Inbound Data	Config	
Success Return	ResponseStatus	
Notes:		
<p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p>		
<p>When it is retrieved the <Config> XML, <Password> field is not displayed ,but if there is not <Password> in the configuration settings, the password is not change, Otherwise, the password will be changed.</p>		
<p>When setting <ProtocolName> field, we must ensure that the value of the field is obtained in the protocol list, otherwise you will set up unsuccessful,and no error message.</p>		
<p>Only in <Enable> is true, the other is set to be effective.</p>		
<p><ServerPort> field temporarily does not support setting, the default value is 0.</p>		

DDNS Config XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Config>
  <Enable><!--req,xs:boolean--></Enable>
  <ProtocolName>
    <!--req,xs:string,"the string is the one of <ServerList>">
  </ProtocolName>
  <ServerPort><!--req,xs:integer--></ServerPort>
  <DomainName><!--req,xs:string--></DomainName>
  <UserName><!--req,xs:string--></UserName>
  <Password><!--wo,req,xs:string--></Password>
  <RefreshIntervals><!--req,xs:integer, in sec--></RefreshIntervals>
</Config>
```

9.1.2.3 /Network/DDNS/Status

/Network/DDNS/ Status		General Resource v1.0
GET		Viewer
Description	It is used to get the read-only configuration of the device DDNS	
Query	None	
Inbound Data	None	
Success Return	Status	
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s。		

DDNS Status XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Status>
  <State>
    <!--ro,req,xs:string, "stopped,Registering ,Registered ,stopping"-->
  </State>
</Status>
```

9.1.3 /Network/UPNP

/Network/UPNP		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device UPNP configuration	
Query	None	
Inbound Data	None	
Success Return	UPNP	
PUT		Viewer
Description	It is used to set all of the device UPNP configuration	
Query	None	
Inbound Data	UPNP	
Success Return	ResponseStatus	
Notes: Please do not frequently set the network configuration, because the equipment end of the set the		

network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.

UPNP XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<UPNP>
  <Config/><!--req-->
  <Status/><!--opt-->
</UPNP>
```

9.1.3.1 /Network/UPNP/Config

/Network/UPNP/Config		General Resource v1.0
GET		Viewer
Description	It is used to get the readable and writable configuration of the device UPNP	
Query	None	
Inbound Data	None	
Success Return	Config	
PUT		Viewer
Description	It is used to set the readable and writable configuration of the device UPNP	
Query	None	
Inbound Data	Config	
Success Return	ResponseStatus	
Notes:		
Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.		

UPNP Config XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Config>
  <Enable><!--req,xs:boolean--></Enable>
  <WanIPAddress><!--req,xs:string--></WanIPAddress>
  <LocalHTTPPort><!--req,xs:integer--></LocalHTTPPort>
  <ExportHTTPPort><!--req,xs:integer--></ExportHTTPPort>
  <LocalMoblePort><!--req,xs:integer--></LocalMoblePort>
  <ExportMoblePort><!--req,xs:integer--></ExportMoblePort>
  <LocalCommandPort><!--req,xs:integer--></LocalCommandPort>
  <ExportCommandPort><!--req,xs:integer--></ExportCommandPort>
  <LocalMediaPort><!--req,xs:integer--></LocalMediaPort>
```

```
<ExportMediaPort><!--req,xs:integer--></ExportMediaPort>
</Config>
```

9.1.3.2 /Network/UPNP/Status

/Network/UPNP/Status		General Resource v1.0
GET		Viewer
Description	It is used to get the read-only configuration of the device UPNP	
Query	None	
Inbound Data	None	
Success Return	Status	
Notes:		
Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.		

UPNP Status XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Status>
  <HTTPPortState><!--ro,req,xs:string,"success,failed,mapping" --></HTTPPortState>
  <MobliePortState><!--ro,req,xs:string,"success,failed,mapping" --></HTTPMoblieState>
  <CommandPortState><!--ro,req,xs:string,"success,failed,mapping"--></CommandPortState>
  <MediaPortState><!--ro,req,xs:string,"success,failed,mapping"--></MediaPortState>
</Status>
```

9.1.4 /Network/Email

/Network/Email		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device Email configuration	
Query	None	
Inbound Data	None	
Success Return	Email	
PUT		Viewer
Description	It is used to set all of the device Email configuration	
Query	None	
Inbound Data	Email	


```

    <TimeBlock_1><!req,xs:string></TimeBlock_1>
    <TimeBlock_2><!req,xs:string></TimeBlock_2>
    <TimeBlock_3><!req,xs:string></TimeBlock_3>
    <TimeBlock_4><!req,xs:string></TimeBlock_4>
    <TimeBlock_5><!req,xs:string></TimeBlock_5>
    <TimeBlock_6><!req,xs:string></TimeBlock_6>
    <TimeBlockList>
  </Schedule>
</EMail>

```

9.1.5 /Network/Port

/Network/Port		General Resource	v1.0
GET		Viewer	
Description	It is used to get all of the device Port configuration		
Query	None		
Inbound Data	None		
Success Return	Port		
PUT		Viewer	
Description	It is used to set all of the device Port configuration		
Query	None		
Inbound Data	Port		
Success Return	ResponseStatus		
Notes:			
<p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>If you set the this configuration , the device will be restart.</p>			

Port XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<Port>
  <HTTP><!--req,xs:integer--></HTTPPort>
  <Command><!--req,xs:integer--></CommandPort>
  <Media><!--req,xs:integer--></MediaPort>
  <Mobile><!--req,xs:integer--></MobilePort>
  <RTP><!--req,xs:integer--></RtpPort>
  <Discovery><!--req,xs:integer--></Discovery>
</Port>

```

9.1.6 /Network/P2P

/Network/P2P		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device P2P configuration	
Query	None	
Inbound Data	None	
Success Return	P2P	
PUT		Viewer
Description	It is used to set all of the device P2P configuration	
Query	None	
Inbound Data	P2P	
Success Return	ResponseStatus	
Notes:		
<p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>After setting the UUID ,IPC will automatically restart, UID will take effect after the restart.</p>		

P2P XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<P2P>
  <Enable><!--req,xs:boolean--></Enable>
  <UUID><!--req,xs:string--></UUID>
  <Status><!--ro,opt,xs:string,"offline,internet-ready,intranet-ready" --></Status>
</P2P>
```

9.1.7 /Network/P2PV2

/Network/P2PV2		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device P2P List configuration	
Query	None	
Inbound Data	None	
Success Return	P2PList	
PUT		Viewer
Description	It is used to set all of the device P2P configuration	
Query	None	

Inbound Data	P2P
Success Return	ResponseStatus
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s. After setting the UUID ,IPC will automatically restart, UID will take effect after the restart.	

P2P XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<P2PList>
  <ID><!--req,xs:integer--></ID>
  <Enable><!--req,xs:boolean--></Enable>
  <P2PProtocolName><!--req,xs:string--></P2PProtocolName>
  <UUID><!--req,xs:string--></UUID>
  <AndroidAppDownload><!--req,xs:string--></AndroidAppDownload>
  <lphoneAppDownload><!--req,xs:string--></lphoneAppDownload>
  <Status><!--ro,opt,xs:string,"offline,internet-ready,intranet-ready"--></Status>
</P2PList>
```

9.1.8 /Network/FTP

/Network/FTP		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device FTP configuration	
Query	None	
Inbound Data	None	
Success Return	FTP	
PUT		Viewer
Description	It is used to set all of the device FTP configuration	
Query	None	
Inbound Data	FTP	
Success Return	ResponseStatus	
Notes: Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s. <TransferContentTypeMask> is a mask,like "00",every bit represent a option,if the value of the bit is "0",It is represent that it is selected,otherwise,not selected,default,From right to left, the first bit is "Image",the second bit is "video".		

When it is retrieved the <FTP> XML, <Password> field is not displayed ,but if there is not <Password> in the configuration settings, the password is not change, Otherwise, the password will be changed.

FTP XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<FTP Version="1.0" xmlns="http://www.zwcloud.wang/ver10/XMLSchema">
  <Enable><!--req,xs:boolean--></Enable>
  <UserName><!--req,xs:string--></UserName>
  <Password><!--wo,opt,xs:string--></Password>
  <IPAddress><!--req,xs:string--></IPAddress>
  <Port><!--req,xs:integer--></Port>
  <TransferContentTypeMask><!--req,xs:string--></TransferContentTypeMask>
  <FTPMode><!--req,xs:string,"Active, Passive"--></FTPMode>
  <StorageDirectory><!--req,xs:string--></StorageDirectory>
  <Status><!--opt,xs:string,"not connect","connected"--></Status>
</FTP>
```

9.1.9 /Network/WifiParamTest

/Network/WifiParamTest		General Resource v1.0
PUT		Viewer
Description	It is used to set all of the device ConnectionConfig configuration	
Query	None	
Inbound Data	ConnectionConfig	
Success Return	ResponseStatus	
Notes:		

ConnectionConfig XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<ConnectionConfig <!--opt-->
  <SSID><!--req,xs:string--></SSID>
  <MACAddress><!--ro,req,xs:string--></MACAddress>
  <NetworkType><!--req,xs:string,managed,Ad-hoc--></NetworkType>
  <AuthMode><!--req,xs:string,OPEN,SHRED,WPAUTO,WPA2PSK,WPA2PSK,
  WPANONE,WPA,WPA2,NOPASSWORD--></AuthMode>
  <WEP><!--dep,depends on <AuthMode-->
    <KeyID><!--req,xs:integer,"1-4"--></KeyID>
    <keyList>
      <Key>
```

```

        <ID><!-- req,xs:integer--></ID>
        <KeyType><!-- req,xs:string,"HEX,ASCII"--></KeyType>
        <KeyValue><!-- opt,xs:string--></KeyValue>
    </Key>
</keyList>
</WEP>
<WPA><!-- dep,depends on <AuthMode>-->
    <EncryptType><!-- req,xs:string,NONE,WEP,TKIP,AES--></EncryptType>
    <Key>
        <KeyType><!-- req,xs:string,"ASCII"--></KeyType>
        <KeyValueBase64><!-- req,xs:string,"the length of the source string is 8-31, and this
value will be codec by base64"--></KeyValueBase64>
    </key>
</WPA>
</ConnectionConfig>

```

9.1.10 /Network/WifiParamTestStatus

/Network/WifiParamTestStatus		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device WIFIStatusForTest configuration	
Query	None	
Inbound Data	None	
Success Return	WIFIStatusForTest	
Notes:		

WIFIStatusForTest XML Block

```

<?xml version="1.0" encoding="UTF-8" ?>
<WIFIStatusForTest Version="1.0">
    <Status><!-- req,xs:string,"disconnect,connected,passwd error"--></Status>
</WIFIStatusForTest>

```

9.1.11 /Network/WhichNetworkCard

/Network/WhichNetworkCard		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device WhichNetcard	

	configuration
Query	None
Inbound Data	None
Success Return	WhichNetcard
Notes:	

WhichNetcard XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<WhichNetcard Version="1.0">
  <NetcardType><!--xs:string,"Unknown,wired1,wired2,WIFI,3G,4G,5G"--></NetcardType>
</WhichNetcard>
```

9.1.12 /Network/Interfaces

/Network/Interfaces		General Resource v1.0
GET		Viewer
Description	It is used to get all of the device Interfaces configuration	
Query	None	
Inbound Data	None	
Success Return	InterfaceList	
PUT		Viewer
Description	It is used to set all of the device Interfaces configuration	
Query	None	
Inbound Data	InterfaceList	
Success Return	ResponseStatus	
Notes:		
Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.		

Network interface XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<InterfaceList>
  <Interface><!--opt-->
</InterfaceList>
```

9.1.12.1 /Network/Interfaces/NetworkCardID

/Network/Interfaces/NetworkCardID		General Resource v1.0
GET		Viewer
Description	It is used to get one of the device Interfaces configuration by the NetworkCardID.	
Query	None	
Inbound Data	None	
Success Return		
PUT		Viewer
Description	It is used to set one of the device Interfaces configuration by the NetworkCardID.	
Query	None	
Inbound Data		
Success Return	ResponseStatus	
<p>Notes:</p> <p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>By default, the value of "NetworkCardID" is 1 which represents a wired network, the value of "NetworkCardID" is 2 which represents a wireless network card. For the future, to support more network card, then turn increases here does not support more than 2.</p>		

Wire XML BLock

```
<?xml version="1.0" encoding="utf-8"?>
<Interface>
  <ID><!--req,xs:integer--></ID>
  <NetworkCardMode><!--req,xs:string--></NetworkCardMode>
  <IPAddress/><!--req-->
</Interface>
```

9.1.12.1.1 /Network/Interfaces/NetworkCardID/IPAddress

/Network/Interfaces/NetworkCardID/IPAddress		General Resource v1.0
GET		Viewer
Description	It is used to get the ip address of a particular network interface	
Query	None	
Inbound Data	None	
Success Return	IPAddress	

PUT		Viewer
Description	It is used to set the ip address of a particular network interface	
Query	None	
Inbound Data	IPAddress	
Success Return	ResponseStatus	
Notes:		
<p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>By default, the value of "NetworkCardID" is 1 which represents a wired network, the value of "NetworkCardID" is 2 which represents a wireless network card. For the future, to support more network card, then turn increases here does not support more than 2.</p>		

IPAddress XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<IPAddress>
  <ipVersion><!--ro,opt,xs:string,V4--></ipVersion>
  <ipAddress><!--req,xs:string--></ipAddress>
  <MacAddress><!--ro,opt,xs:string,readonly--></MacAddress>
  <SubnetMask>
    <!-- req, xs:string, subnet mask for IPv4 address -->
  </SubnetMask>
  <Gateway><!--req,xs:string--></ Gateway>
  <DHCP/><!--req-->
  <DNS/><!--req-->
</IPAddress>
```

9.1.12.1.1.1 /Network/Interfaces/NetworkCardID/IPAddress/DHCP

/Network/Interfaces/NetworkCardID/IPAddress/DHCP		General Resource v1.0
GET		Viewer
Description	It is used to get the DHCP of a particular network interface.	
Query	None	
Inbound Data	None	
Success Return	DHCP	
PUT		Viewer
Description	It is used to set the DHCP of a particular network interface.	
Query	None	
Inbound Data	DHCP	
Success Return	ResponseStatus	

Notes:

Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.

Network Card 1 is support the URI /Network/Interfaces/NetworkCardID/IPAddress/DHCP,but Network Card 2 is not support..

The value of the <Status> is invalid in the Network Card 2.

By default, the value of "NetworkCardID" is 1 which represents a wired network, the value of "NetworkCardID" is 2 which represents a wireless network card.For the future, to support more network card, then turn increases here does not support more than 2.

DHCP XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<DHCP>
  <Enable><!--req,xs:boolean--></Enable>
  <Status><!--ro,opt,xs:string,"initializing,obtaining,obtained" --></Status>
</DHCP>
```

9.1.12.1.1.2 /Network/Interfaces/NetworkCardID/IPAddress/DNS

/Network/Interfaces/NetworkCardID/IPAddress/DNS		General Resource	v1.0
GET		Viewer	
Description	It is used to get the DNS of a particular network interface.		
Query	None		
Inbound Data	None		
Success Return	DNS		
PUT		Viewer	
Description	It is used to set the DNS of a particular network interface.		
Query	None		
Inbound Data	DNS		
Success Return	ResponseStaus ResponseStatus		

Notes:

Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.

Network Card 1 is support the URI /Network/Interfaces/NetworkCardID/IPAddress/DNS,but Network Card 2 is not support.

The value of The <Enable> in the Network Card 2 is depended on the value of the <Enable> in the <DHCP> ,so It is invalid that you set the value of the <Enable> in the <DNS>.

By default, the value of "NetworkCardID" is 1 which represents a wired network, the value of "NetworkCardID" is 2 which represents a wireless network card.For the future, to support

more network card, then turn increases here does not support more than 2.

DNS XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<DNS>
  <Enable><!--req,xs:boolean--></Enable>
  <PrimaryDNS><!--req,xs:string--></PrimaryDNS>
  <SecondaryDNS><!--req,xs:string--></SecondaryDNS>
</DNS>
```

9.1.12.1.2 /Network/Interfaces/ NetworkCardID/WIFIAccessPointList

/Network/Interfaces/NetworkCardID/WIFIAccessPointList		General Resource	v1.0
GET		Viewer	
Description	It is used to get all wireless Access Point List.		
Query	None		
Inbound Data	None		
Success Return	WIFIAccessPointList		
Notes:			
<p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p>			
<p>Only wireless card supports the URI /Network/Interfaces/NetworkCardID/WIFIAccessPointList.</p>			
WirelessMode:			
<ol style="list-style-type: none">1. legacy 11B only2. legacy 11A only3. legacy 11a/b/g mixed4. legacy 11G only5. 11ABGN mixed6. 11N only7. 11GN mixed8. 11AN mixed9. 11BGN mixed			
<p>By default, the value of "NetworkCardID" is 1 which represents a wired network, the value of "NetworkCardID" is 2 which represents a wireless network card. For the future, to support more network card, then turn increases here does not support more than 2.</p>			

WIFIAccessPointList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<WIFIAccessPointList><!--req-->
  <AccessPoint><!--opt-->
    <ID><!--ro,req,xs:integer--></ID>
```

```

<SSID><!ro,req,xs:string--></SSID>
<MACAddress><!--ro,req,xs:string--></MACAddress>
<SignalValue><!--ro,req,xs:integer,0-100--></SignalValue>
<Authentication><!--ro,req,xs:string,None,shared,WPA,WPAPSK,WPA2PSK,
WPANONE,WPA,WPA2--></Authentication>
<EncryptType><!--ro,req,xs:string,NONE,WEP,TKIP,AES--></EncryptType>
<WirelessMode><!--ro,req,xs:integer--></WirelessMode>
</AccessPoint>
</WiFiAccessPointList>

```

9.1.12.1.3 /Network/Interfaces/NetworkCardID/Wireless

/Network/Interfaces/NetworkCardID/Wireless		General Resource v1.0
GET		Viewer
Description	It is used to get the WIFI information of a wireless network interface.	
Query	None	
Inbound Data	None	
Success Return	ConnectionConfig	
PUT		Viewer
Description	It is used to set the WIFI information of a wireless network interface.	
Query	None	
Inbound Data	ConnectionConfig	
Success Return	ResponseStatus	
Notes:		
<p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>Only wireless card supports the URI /Network/Interfaces/NetworkCardID/WiFiAccessPointList.</p> <p>If <AuthMode> mode is OPEN, SHARED mode, then the <WEP> is Effective. If <AuthMode> model is WPAPSK, WPA2PSK more, then the <WPA> is Effective. Otherwise, they are all not valid.but they are all displayed.</p> <p>if the <AuthMode> is WPAPSK or WPA2PSK,the value of the <keyvalue> in the <WPA> should not be empty.</p> <p>The value of the <KeyType> in the <WPA> must be "ASCII".</p> <p>By default, the value of "NetworkCardID" is 1 which represents a wired network, the value of "NetworkCardID" is 2 which represents a wireless network card.For the future, to support more network card, then turn increases here does not support more than 2.</p>		

WIFI ConnectionConfig XML Block

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

```

<ConnectionConfig> <!--opt-->
  <Enable><!--req,xs:integer,"disconnected,connected" --></Enable>
  <SSID><!--req,xs:string--></SSID>
  <MACAddress><!--ro,req,xs:string--></MACAddress>
  <NetworkType><!--req,xs:string,managed,Ad-hoc--></NetworkType>
  <AuthMode><!--req,xs:string,OPEN,SHRED,WEPAUTO,WPAPSK,WPA2PSK,
  WPANONE,WPA,WPA2,NOPASSWORD--></AuthMode>
  <WEP><!--dep,depends on <AuthMode>-->
    <KeyID><!--req,xs:integer,"1-4" --></KeyID>
    <keyList>
      <Key>
        <ID><!--req,xs:integer--></ID>
        <KeyType><!--req,xs:string,"HEX,ASCII"--></KeyType>
        <KeyValue><!--opt,xs:string--></KeyValue>
      </Key>
    </keyList>
  </WEP>
  <WPA><!--dep,depends on <AuthMode>-->
    <EncryptType><!--req,xs:string,NONE,WEP,TKIP,AES--></EncryptType>
    <Key>
      <KeyType><!--req,xs:string,"ASCII"--></KeyType>
      <KeyValueBase64><!--req,xs:string,"the length of the source string is 8-31, and this
value will be codec by base64"--></KeyValueBase64>
    </key>
  </WPA>
</ConnectionConfig>

```

9.1.12.1.4 /Network/Interfaces/NetworkCardID/WIFIStatus

/Network/Interfaces/NetworkCardID/WIFIStatus		General Resource v1.0
GET		Viewer
Description	It is used to get the status of a wireless network interface.	
Query	None	
Inbound Data	None	
Success Return	WIFIStatus	
Notes: <p>Please do not frequently set the network configuration, because the equipment end of the set the network configuration is asynchronous responses, so if you frequently set, only the response to the first operation, the recommended time interval set 3-5s.</p> <p>Only wireless card supports the URI /Network/Interfaces/NetworkCardID/WIFIAccessPointList.</p> <p>By default, the value of "NetworkCardID" is 1 which represents a wired network, the value</p>		

of "NetworkCardID" is 2 which represents a wireless network card. For the future, to support more network card, then turn increases here does not support more than 2.

WiFiStatus XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<WiFiStatus><!--req-->
  <State><!--ro,req,xs:string,"open,close,notsupport" --></State>
</WiFiStatus>
```

9.2 /System

/System	General Resource	v1.0
---------	------------------	------

9.2.1 /System/Time

/System/Time		General Resource	v1.0
GET		Viewer	
Description	It is used to get the device time information.		
Query	None		
Inbound Data	None		
Success Return	Time		
PUT		Viewer	
Description	It is used to update the device time information.		
Query	None		
Inbound Data	Time		
Success Return	ResponseStaus ResponseStatus		
Notes:			
SystemTime:An ISO 8601 Date/Time string is accepted and returned.			

Time XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Time>
  <DateTimeFormat>
    <!--req,string,YYYYMMDDWhhmmss,YYYYMMDDhhmmss,MMDDYYYYWhhmmss,MMDD
    YYYYhhmmss,DDMMYYYYWhhmmss,DDMMYYYYhhmmss-->
  </DateTimeFormat>
  <TimeFormat><!--req,xs:string,12hour,24hour--></TimeFormat>
  <SystemTime><!--req,xs:datetime," 20040503T173008+08"--></SystemTime>
</Time>
```

9.2.2 /System/NTP

/System/NTP		General Resource v1.0
GET		Viewer
Description	It is used to get the configuration of NTP server for the device.	
Query	None	
Inbound Data	None	
Success Return	NTP	
PUT		Viewer
Description	It is used to set the configuration of NTP server for the device.	
Query	None	
Inbound Data	NTP	
Success Return	ResponseStatus	
Notes: TimeZone: It must be start with "CST".		

NTP XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<NTP>
  <Enable><!--req,xs:boolean--></Enable>
  <ServerName><!--req,xs:string,"example: time.windows.com"--></ServerName>
  <TimeZone><!--req, xs:string, POSIX time zone string,"CST-09:00:00"--></TimeZone>
  <TimeSyncInterval><!--req,xs:integer--></TimeSyncInterval>
</NTP>
```

9.2.3 /System/DST

/System/DST		General Resource v1.0
GET		Viewer
Description	It is used to get the configuration of DST for the device.	
Query	None	
Inbound Data	None	
Success Return	DST	
PUT		Viewer
Description	It is used to set the configuration of DST for the device.	
Query	None	
Inbound Data	DST	

Success Return	ResponseStatus
<p>Notes:</p> <p>When the <Mode> is "date" ,the <DateModeStart> and<DateModeStop> is effective , When the <Mode> is "week" ,the < WeekModeStart > and< WeekModeStop > is effective</p> <p><TimeDifference> the unit is seconds,example:3600 is 3600 seconds</p> <p><DateModeStart> and <DateModeStop>:An ISO 8601 Date/Time string is accepted and returned.</p> <p><WeekModeStart> and <WeekModeStop>:the first number of the "3-1-1-2-3-6" is March,the second is the first week of the month,the third is Monday of the first week,the forth is 2 hour,the fifth is 3 minute, the sixth is 6 second</p>	

NTP XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<DST Version="1.0" xmlns="http://www.zwcloud.wang/ver10/XMLSchema">
  <Enable><!--req,xs:boolean--></Enable>
  <Mode><!--req,xs:string,"date,week"--></Mode>
  <TimeDifference>req,xs:string, </TimeDifference>
  <DateModeStart><!--req,xs:datetime," 20040503T173008+08"--></DateModeStart>
  <DateModeStop><!--req,xs:datetime," 20040503T173008+08"--></DateModeStop>
  <WeekModeStart><!--req,xs:string," 3-1-1-2-3-6"--></WeekModeStart>
  <WeekModeStop><!--req,xs:string," 2-4-5-7-8-8"--></WeekModeStop>
</DST>
```

9.2.4 /System/DeviceInfo

/System/DeviceInfo		General Resource v1.0
GET		Viewer
Description	It is used to get device information.	
Query	None	
Inbound Data	None	
Success Return	DeviceInfo	
PUT		Viewer
Description	It is used to set the configuration of DeviceName for the device.	
Query	None	
Inbound Data	DeviceInfo	
Success Return	ResponseStatus	
<p>Notes:</p> <p><DeviceName> and <ActiveCode> can be set.others are not be set.</p> <p><ActiveStatus> "success" means active success, or be "invalid active code"</p> <p><IsSupportActiveCode>, if this field was false, the client should not show Active Code and Active Status field.</p>		

DeviceInfo XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<DeviceInfo>
  <DeviceName><!--rw,req,xs:string--></DeviceName>
  <BoardType><!--ro,req,xs:integer--></BoardType>
  <DeviceTypeName><!--ro,req,xs:string--></DeviceTypeName>
  <SerialNum><!--ro,req,xs:string--></SerialNum>
  <HardWareVersion><!--ro,req,xs:string--></HardWareVersion>
  <SoftWareVersion><!--ro,req,xs:string--></SoftWareVersion>
  <VideoInputChannels><!--ro,-req,xs:integer--></VideoChannels>
  <AudioChannels><!--ro,req,xs:integer--></AudioChannels>
  <AlarmInputChannels><!--ro,req,xs:integer--></AlarmChannel>
  <AlarmOutputChannels><!--ro,req,xs:integer--></AlarmOutputChannel>
  <MonitorCount><!--ro,req,xs:integer--></MonitorCount>
  <SpotOutChannels><!--ro,req,xs:integer--></SpotOutChannels>
  <TwoWayAudio><!--ro,req,xs:boolean--></TwoWayAudio>
  <SmartAVCList Version="1.0" xmlns="http://www.zwcloud.wang/ver10/XMLSchema">
    <SmartAVC Version="1.0" xmlns="http://www.zwcloud.wang/ver10/XMLSchema">
      <Support><!--ro,opt,xs:Boolean--></Support>
    </SmartAVC>
  </SmartAVCList>
  <IsSupportActiveCode><!--ro,req,xs:boolean--></IsSupportActiveCode>
  <ActiveCode><!--ro,req,xs:string--></ActiveCode>
  <ActiveStatus><!--ro,req,xs:string--></ActiveStatus>
  // Algorithm Version, The version that describes the ai algorithm
  <AlgorithmVersion><!--ro,req,xs:string--></AlgorithmVersion>
</DeviceInfo>
```

9.2.5 /System/DeviceStatus

/System/DeviceStatus		General Resource v1.0
GET		Viewer
Description	It is used to get all Channels Status of the device	
Query	None	
Inbound Data	None	
Success Return	DevicestatusList	
Notes:		

DevicestatusList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<DeviceStatusList>
  <DeviceStatus/><!--opt-->
</DeviceStateList>
```

9.2.5.1 /System/DeviceStatus/ChannelID

/System/DeviceStatus/ID		General Resource v1.0
GET		Viewer
Description	It is used to get a Channel Status of the device	
Query	None	
Inbound Data	None	
Success Return	DeviceStatus	
Notes:		
<p>The value of the <FirstSubStreamEnable> field represents whether It is open the first sub stream or not.</p> <p>The value of the <SecondSubStreamEnable> field represents whether It is open the second sub stream or not.</p> <p>The value of the <RecordState> field is "none",represents"not support" for the Unsupported device,but "no records" for the supported device.</p> <p>The value of the <VideoLostState> field is "false",represents"not support" for the unsupported device,but "no video lost event" for the supported device.</p> <p>The value of the < MotionState > field is "false",represents"not support" for the unsupported device,but "no motion event" for the supported device.</p> <p>The value of the < AlarmInState > field is "false",represents"not support" for the unsupported device,but "no alarm in event" for the supported device.</p> <p>The value of the < AlarmInState > field is "false",represents"not support" for the unsupported device,but "no alarm out event" for the supported device.</p>		

DeviceStatus XML Block

```
<DeviceStatus>
  <ID><!--req,xs:integer--></ID>
  <MainStreamVideoBitrate><!--ro,req,xs:integer --></ MainStreamVideoBitrate>
  <FirstSubStreamEnable><!--ro,req,xs:boolean--></ FirstSubStreamEnable>
  <FirstSubStreamVideoBitrate><!--ro,req,xs:integer--></ FirstSubStreamVideoBitrate>
  <SecondSubStreamEnable><!--ro,req,xs:boolean--></ SecondSubStreamEnable>
  <SecondSubStreamVideoBitrate><!--ro,req,xs:integer--></SecondStreamVideoBitrate>
  <RecordState><!--ro,req,xs:string," none,alarm,normal,manual,motion" --></RecordState>
  <VideoLostState><!--ro,req,xs:boolean--></VideoLostState>
  <MotionState><!--ro,req,xs:boolean--></MotionState>
  <AlarmInState><!--ro,req,xs:boolean--></AlarmInState>
```

```
<AlarmOutState><!--ro,req,xs:boolean--></AlarmOutState>
</DeviceStatus>
```

9.2.6 /System/FirmwareUpgrade

/System/FirmwareUpgrade		General Resource v1.0
PUT		Viewer
Description	It is used to upgrade the firmware of the device.	
Query	None	
Inbound Data	Opaque Data	
Success Return	ResponseStatus	
Notes:		
The URI /System/FirmwareUpgrade shows the equipment upgrade, using HTTP PUT method, with authentication and update files to be uploaded.		

9.2.7 /System/Reboot

/System/Reboot		General Resource v1.0
PUT		Viewer
Description	It is used to reboot the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		

9.2.8 /System/FactoryDefault

/System/FactoryDefault		General Resource v1.0
PUT		Viewer
Description	It is used to reset the configuration for the device to the factory default.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		
It is not recovery the network configuration.		

9.2.9 /System/FactoryDefaultV2

/System/FactoryDefaultV2		General Resource v1.0
PUT		Viewer
Description	It is used to reset the configuration for the device to the factory default.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes: It is recovery all configuration.		

9.2.10 /System/SendTestEmail

/System/SendTestEmail		General Resource v1.0
PUT		Viewer
Description	It is used to Send Test Email.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes: Please first properly configured Email Configuration with using the URL /Network/Email.		

9.2.11 /System/RecoveryAdminPassword

/System/RecoveryAdminPassword		General Resource v1.0
PUT		Viewer
Description	It is used to Recovery the password of the user "admin" Password And IP.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes: Defaultly the password of the user "admin" is empty,The IP is "192.168.1.11" .		

9.2.12 /System/TryListenCurrentAlarmAudio

/System/TryListenCurrentAlarmAudio		General Resource v1.0
PUT		Viewer
Description	Listen to the current alarm audio	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		

9.2.13 /System/Logging

/System/Logging		General Resource v1.0
GET		Viewer
Description	It is used to get the log information of the device.	
Query	majorType minorType startTime stopTime	
Inbound Data	None	
Success Return	LogList	
Notes:		
<p>The value of "majorType" is:</p> <ul style="list-style-type: none"> 1: System Operation 2: Configuration Operation 3: Record Operation 4: User Management 5: Alarm Event 6: Exception Event 7: Storage Management 8: All Operation <p>When the value of "majorType" is 1, the value of "minorType" is:</p> <ul style="list-style-type: none"> 0: All System Operation 1: System upgrade 2: PTZ control 3: Illegal shutdown 4: Normal shutdown 5: Startup 6: Remote reboot 		

7:Clear alarm

When the value of "majorType" is 2, the value of "minorType" is:

- 100:All Configuration Operation
- 101: Record time table settings
- 102: Normal settings
- 103: Encoding settings
- 104: Network settings
- 105: Server settings
- 106: Motion detection settings
- 107: PTZ decoder settings
- 108: Restore default settings
- 109: Set display device parameters
- 110: Display mode settings
- 111: Image color settings
- 112: Auto-maintenance settings
- 113: Alarm output settings
- 114: Alarm input settings
- 115: Function customize settings
- 116: Import PTZ protocol
- 117: Export PTZ protocol
- 118: Import parameters settings
- 119: Export parameters settings
- 120: Abnormal parameters settings

When the value of "majorType" is 3, the value of "minorType" is:

- 200: All Record Operation
- 201: Start manual record
- 202: Start auto record
- 203: Stop

When the value of "majorType" is 4, the value of "minorType" is:

- 300:All User Management
- 301: Add user
- 302: Delete user
- 303: Modify user
- 304: User login
- 305: User logout

When the value of "majorType" is 5, the value of "minorType" is:

- 400: All Alarm Event
- 401: Motion detection begin
- 402: Motion detection end
- 403: Input alarm begin

When the value of "majorType" is 6, the value of "minorType" is:

- 500: All Exception Event
- 501: IP confliction
- 502: HDD full

- 503: HDD error
- 504: Illegal visit
- 505: Signal loss
- 506: Signal restore
- 507: Network disconnect
- When the value of “majorType” is 7, the value of “minorType” is:
- 600: All Storage Management
- 601: Format partition
- 602: Create partition
- 603: Delete partition
- 604: Uninstall HDD
- 605: HDD group management

The format of “startTime” and “stopTime” is “YYYY-MM-DDThh:mm:ss”.
 Devices support up to 1000 log.

For example:

<http://192.168.1.171/System/Logging?majorType=8&minorType=0&startTime=2015-07-28T00:00:00&stopTime=2015-07-29T23:59:00>.

The value of the minorType is invalid for some device.

LogList XML Block

```
<LogList>
  <Log> <!-- opt -->
    <ID> <!-- req, xs:integer --> </ID>
    <time> <!-- req, xs:datetime --> </time>
    <majorType> <!-- req, xs:integer --> </majorType>
    <minorType> <!-- req, xs:integer --> </minorType>
    <netUser> <!-- req, xs:string --> </netUser>
    <hostIPAddress> <!-- req, xs:string --> </hostIPAddress>
    <channel> <!-- req, xs:integer --> </channel>
  </Log>
</LogList>
```

9.2.14 /System/Users

/System/Users		General Resource v1.0
GET		Viewer
Description	It is used to get the configuration of all users for the device.	
Query	None	
Inbound Data	None	
Success Return	UserList	

Notes:

NTP XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<UserList>
  <User> <!-- opt -->
</UserList>
```

9.2.15 /System/Users/ID

/System/Users/ID		General Resource v1.0
GET		Viewer
Description	It is used to get the configuration of the specifies ID user for the device.	
Query	None	
Inbound Data	None	
Success Return	User	
PUT		Viewer
Description	It is used to update the specifies user for the device.	
Query	None	
Inbound Data	User	
Success Return	ResponseStatus	
POST		Viewer
Description	It is used to add a user for the device.	
Query	None	
Inbound Data	User	
Success Return	ResponseStatus	
Delete		Viewer
Description	It is used to delete the specifies user for the device.	
Query	None	
Inbound Data	User	
Success Return	ResponseStatus	
Notes:		
When call the "GET" method ,The <ID> unit is effective,when call other method , The <ID> unit is no effective.you can not set the user information by <ID>,you just set user password by <UserName>.		
The <ID> is start at 1.		
Please do not delete the "Default" or "admin" user.		

Please do not create a "Default" or "Administrator" level user,you just can create a "Normal" level user.

We just support the user level ,the right for the different level is your decision.

NTP XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<User>
  <ID><!--req,xs:integer--></ID>
  <UserName><!--req,xs:string--></UserName>
  <Password><!--wo,req,xs:string--></Password>
  <UserLevel><!--req,xs:string,"Default,Administrator,Normal"> </UserLevel>
  <IPAddress><!--ro,opt,xs:string></IPAddress>
  <MACAddress><!--ro,opt,xs:string></MACAddress>
</User>
```

9.2.16 /System/ChannelID/RemoteForceIframe

/System/ChannelID/RemoteForceIframe		General Resource	v1.0
PUT		Viewer	
Description	It is used to Remote Force I Frame.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
The <ChannelID> in URI shows the channel number.The <ChannelID> is start at 1.			

9.2.17 /System/ChannelID/RemoteAlarmOutClear/ID

/System/ChannelID/RemoteAlarmOutClear/ID		General Resource	v1.0
PUT		Viewer	
Description	It is used to Remote Clear Alarm out.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
The <ChannelID> in URI shows the alarm channel number. The <ChannelID> is start at 1.The <ID> in URI shows the Alarm Type, please set the ID to 0.			

9.2.18 /System/ChannelID/RemoteAlarmoutControl/on

/System/ChannelID/RemoteAlarmoutControl/on		General Resource v1.0
PUT		Viewer
Description	It is used to open alarmout state	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		
<p>The ChannelID in URI shows the alarm channel number. The <ChannelID> is start at 1. It is used to set the alarmout state to open in the /System/AlarmoutState.</p>		

9.2.19 /System/ChannelID/RemoteAlarmoutControl/off

/System/ChannelID/RemoteAlarmoutControl/off		General Resource v1.0
PUT		Viewer
Description	It is used to close alarmout state	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		
<p>The ChannelID in URI shows the alarm channel number. The <ChannelID> is start at 1. It is used to set the alarmout state to close in the /System/AlarmoutState.</p>		

9.2.20 /System/AlarmoutState

/System/Alarmoutstate		General Resource v1.0
GET		Viewer
Description	It is used to close alarmout	
Query	None	
Inbound Data	None	
Success Return	AlarmoutState	
Notes:		
<p>The format of <State>is similar to "1111111" or "00000000", Each bit represents whether the alarm of a channel is open or not, NO.1 means trigger the alarm-out, NO.0 means not yet trigger the alarm-out, for example, the string "1111111" Represents that the device is 8 channels, from left to right, represent channel 1,channel 2,channel 3..... and so on.</p>		

AlarmoutState XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<AlarmoutState><!--req-->
  <State><!--ro,req,xs:string,"0000000000000000" --></State>
</ AlarmoutState>
```

9.2.21 /System/AlarmInState

/System/AlarmInstate		General Resource v1.0
GET		Viewer
Description	It is used to close alarmout	
Query	None	
Inbound Data	None	
Success Return	AlarmInState	
Notes:		
<p>The format of <State>is similar to "1111111" or "0000000", Each bit represents whether the alarm of a channel is open or not, NO.1 means close, NO.0 means open, for example, the string "1111111" Represents that the device is 8 channels, from left to right, represent channel 1,channel 2,channel 3..... and so on.</p>		

AlarmInState XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<AlarmInState><!--req-->
  <State><!--ro,req,xs:string,"0000000000000000" --></State>
</ AlarmInState>
```

9.2.22 /System/DeviceConfigFile

/System/DeviceConfigFile		General Resource v1.0
GET		Viewer
Description	It is used to download the device configuration file	
Query	None	
Inbound Data	None	
Success Return	Config file	
PUT		Viewer
Description	It is used to upload the device configuration file	
Query	None	
Inbound Data	Config file	
Success Return	ResponseStatus	

Notes:

9.2.23 /System/TransparentChannel/WriteOnce

/System/TransparentChannel/WriteOnce		General Resource v1.0
PUT		Operator
Description	It is used to send data on the transparent channel.	
Query	None	
Inbound Data	TransparentChannelWriteonce	
Success Return	ResponseStatus	
Notes: <p><Data> please put the data you want to send to the string type.the format of data is "12 23 ff fe fc",the max count of number is 32.the range of value is 0--ff</p> <p>The value of "majorType" is the number such as,0,1,2....., when the value is 0,the type of Serial port is RS232,when the value is 1,the Type of Serial port is RS485,when the value is 2, the type of Serial port is RS422.</p> <p>This value of "minorType" is also number such as 0,1,2....., default the value is 0.</p> <p>Default the value of the majorType is 2 and the value of the minorType is 0.others is not support .</p> <p>The baudrate is depend on the URL /PTZ/ChannelID/Config.</p> <p>For example: http://192.168.1.171/System/TransparentChannel/WriteOnce?majorType=2&minorType=0</p>		

TransparentChannelWriteOnce XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<TransparentChannelWriteonce><!--req-->
  <Data><!--ro,req,xs:string, --></Data>
</TransparentChannelWriteonce>
```

9.2.24 /System/TransparentChannel/Connect

/System/TransparentChannel/Connect		General Resource v1.0
PUT		Operator
Description	It is used to create a connection on the transparent channel.	
Query	None	
Inbound Data	TransparentChannelConnectReponse	

Success Return	ResponseStatus
<p>Notes:</p> <p>The value of "majorType" is the number such as,0,1,2.....,default the value is 0.</p> <p>This value of "minorType" is also number such as 0,1,2.....,when the value is 0,the type of Serial port is RS232,when the value is 1,the Type of Serial port is RS485,when the value is 2, the type of Serial port is RS422.</p> <p>This URL is not support.</p> <p>For example:</p> <p>http://192.168.1.171/System/TransparentChannel/Connect?majorType=2&minorType=0</p>	

TransparentChannelConnectReponse **XML Block**

```
<?xml version="1.0" encoding="utf-8"?>
<TransparentChannelConnectReponse><!--req-->
  <Index><!--ro,req,xs:string, --></Index>
</TransparentChannelConnectReponse>
```

9.2.25 /System/TransparentChannel/Write

/System/TransparentChannel/Write		General Resource v1.0
PUT		Operator
Description	It is used to write datas on the transparent channel.	
Query	None	
Inbound Data	TransparentChannelWrite	
Success Return	ResponseStatus	
<p>Notes:</p> <p><Index> the Index need be set, It is from the response of "/System/TransparentChannel/Connect".</p> <p><Data> please put the data you want to send to the string type.</p> <p>This URL is not support.</p>		

AlarmInState **XML Block**

```
<?xml version="1.0" encoding="utf-8"?>
<TransparentChannelWrite><!--req-->
  <Index><!--ro,req,xs:string, --></Index>
  <Data><!--ro,req,xs:string, --></Data>
</TransparentChannelWrite>
```

9.2.26 /System/TransparentChannel/Read

/System/TransparentChannel/Read		General Resource v1.0
PUT		Operator
Description	It is used to read datas on the transparent channel.	
Query	None	
Inbound Data	TransparentChannelRead	
Success Return	ResponseStatus	
Notes:		
<p><Index> the Index need be set, It is from the response of “/System/TransparentChannel/Connect”.</p> <p><Data> the data is from device.</p> <p>This URL is not support.</p>		

AlarmInState XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<TransparentChannelRead><!--req-->
  <Index><!--ro,req,xs:string, --></Index>
  <Data><!--ro,req,xs:string, --></Data>
</TransparentChannelRead>
```

9.2.27 /System/TransparentChannel/Disconnect

/System/TransparentChannel/Disconnect		General Resource v1.0
PUT		Operator
Description	It is used to destroy the connection on the transparent channel.	
Query	None	
Inbound Data	TransparentChannelDisconnect	
Success Return	ResponseStatus	
Notes:		
<p><Index> the Index need be set, It is from the response of “/System/TransparentChannel/Connect”.</p> <p>This URL is not support.</p>		

AlarmInState XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<TransparentChannelDisconnect><!--req-->
  <Index><!--ro,req,xs:string, --></Index>
</TransparentChannelDisconnect>
```

9.2.28 /System/DeviceAutoMaintenance

/System/DeviceAutoMaintenance		General Resource v1.0
GET		Viewer
Description	It is used to send data on the transparent channel.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or AutoMaintenance	
PUT		Operator
Description	It is used to send data on the transparent channel.	
Query	None	
Inbound Data	AutoMaintenance	
Success Return	ResponseStatus	
Notes: <Enable>: when <Enable> is false ,other value is invalid, otherwise,other value is valid. <Mode>: when <Mode> is "everyweek",<WeekDayMask> and <MaintenanceTime> is valid,when <Mode> is "everyday" or "once",only <MaintenanceTime>is valid. <WeekDayMask>:the format of <WeekDayMask> is "0000000",from left to right,it is show Sunday,Monday,Tuesday,Wednesday,Thursday,Friday,Saturday.if the bit for one day is "1",then this day is open automatic-maintenance,otherwise,this is close automatic-maintenance.		

AlarmInState XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<AutoMaintenance><!--req-->
  <Enable><!--req,xs:boolean--></Enable>
  <Mode><!--req,xs:string,"everyweek,everyday,once"--></Mode>
  <WeekDayMask><!--req,xs:string,"0000000"--></WeekDayMask>
  <MaintenanceTime><!--req,xs:datetime," 20040503T173008"-->
  </ MaintenanceTime>
</AutoMaintenance>
```

9.2.29 /System/SendTestFTP

/System/SendTestFTP		General Resource v1.0
PUT		Viewer
Description	It is used to Send Test FTP.	
Query	None	
Inbound Data	None	

Success Return	ResponseStatus
Notes: Please first properly configured Email Configuration with using the URL /Network/FTP.	

9.2.30 /System/DeviceWifiSTA2AP

/System/DeviceWifiSTA2AP		General Resource v1.0
PUT		Viewer
Description	It is used to Enter AP Mode	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		

9.2.31 /System/DeviceCustomFunction

/System/DeviceCustomFunction		General Resource v1.0
PUT		Viewer
Description	It is used to custom function.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes: You can transfer data that you want to transfer.		

9.2.32 /System/CloudUpgradeServerInfo

/System/CloudUpgradeServerInfo		General Resource v1.0
PUT		Viewer
Description	It is used to Set cloud upgrade server information.	
Query	None	
Inbound Data	CloudUpgradeServer	
Success Return	ResponseStatus	
Notes: The following configuration is the configuration of the upgrade server, including the IP,<HostIP>, the port number of the service,<Port>, the username,<Username> and password,<Password>, the protocol,<Protocol> used, the corresponding path,<Path>		

CloudUpgradeServer XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<CloudUpgradeServer><!--req-->
  <Port><!--req,xs:interger,0-65535--></Port>
  <HostIP><!--req,xs:string,it must be a ip address--></HostIP>
  <Username><!--req,xs:string,the max length is 256--></Username>
  <Password><!--req,xs:string,the max length is 256--></Password>
  <Protocol><!--req,xs:string,http or ftp--></Protocol>
  <Path><!--req,xs:string,the max length is 128--></Path>
</CloudUpgradeServer>
```

9.2.33 /System/CloudUpgradeFirmwareList

/System/CloudUpgradeFirmwareList		General Resource v1.0
GET		Viewer
Description	It is used to get cloud upgrade firmware list.	
Query	None	
Inbound Data	FirmwareList	
Success Return	ResponseStatus	
Notes:		
Example:		
http://192.168.1.11/System/CloudUpgradeFirmwareList?VersionType=all default (all, It is get all firmware version list, default,It is get a the newest firmware version)		

FirmwareList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<FirmwareList><!--req-->
  <Firmware>
    <Name><!--req,xs:string--></Name>
    <Desc><!--req,xs:string--></Desc>
    <Version><!--req,xs:string--></Version>
    <TotalSize><!--req,xs:interger--></TotalSize>
  </Firmware>
</FirmwareList>
```

9.2.34 /System/CloudUpgradeDownloadStart

/System/CloudUpgradeDownloadStart		General Resource v1.0
PUT		Viewer
Description	It is used to Start the Download.	

Query	None
Inbound Data	Action
Success Return	ResponseStatus
Notes: <p style="text-align: center;"><Passport> It must be a unique string and generated by the client, and all subsequent operations depend on this string.</p> <p style="text-align: center;"><FirmwareName> It is a string getting by the url “/System/CloudUpgradeFirmwareList”</p>	

Action XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Action><!--req-->
  <Passport><!--req,xs:string--></Passport>
  <FirmwareName><!--req,xs:string,--></FirmwareName>
</Action>
```

9.2.35 /System/CloudUpgradeDownloadStatus

/System/CloudUpgradeDownloadStatus		General Resource	v1.0
GET		Viewer	
Description	It is used to get the result of the download.		
Query	None		
Inbound Data	None		
Success Return	Status		
Notes: Example: http://192.168.1.11/System/CloudUpgradeDownloadStatus?Passport=abc (the “abc” is a a string generated by the client)			

Status XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Status><!--req-->
  <Status><!--req,xs:string--></Status>
  <TotalSize><!--req,xs:string--></TotalSize>
  <SentSize><!--req,xs:string--></SentSize>
  <Passport><!--req,xs:string--></Passport>
</Status>
```

9.2.36 /System/CloudUpgradeDownloadTerminate

/System/CloudUpgradeDownloadTerminate	General Resource	v1.0
--	------------------	------

PUT		Viewer
Description	It is used to terminate the Download.	
Query	None	
Inbound Data	Action	
Success Return	ResponseStatus	
Notes:		

Action XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Action><!--req-->
  <Passport><!--req,xs:string--></Passport>
</Action>
```

9.2.37 /System/CloudUpgradeStartUpgrade

/System/CloudUpgradeStartUpgrade		General Resource v1.0
PUT		Viewer
Description	It is used to start to upgrade.	
Query	None	
Inbound Data	Action	
Success Return	ResponseStatus	
Notes:		

Action XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Action><!--req-->
  <Passport><!--req,xs:string--></Passport>
</Action>
```

9.2.38 /System/CloudUpgradeResult

/System/CloudUpgradeResult		General Resource v1.0
GET		Viewer
Description	It is used to get the result of the cloud upgrade	
Query	None	
Inbound Data	None	
Success Return	UpgradeStatus	
Notes:		
Example: http://192.168.1.11/System/CloudUpgradeResult?Passport=abc (the "abc" is a a string		

generated by the client)

UpgradeStatus XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<UpgradeStatus><!--req-->
  <Status><!--req,xs:string--></Status>
  <Passport><!--req,xs:string--></Passport>
  <FirmwareName><!--req,xs:string--></FirmwareName>
</UpgradeStatus>
```

9.2.39 /System/ErrorInfoMap

/System/ErrorInfoMap		General Resource v1.0
GET		Viewer
Description	It is used to get the special error description.	
Query	None	
Inbound Data	None	
Success Return	ErrorInfo	
Notes: <ErrorCode> it is a returned error code by the http server. <ErrorDescription>it is a returned error description by the http server. Example: http://192.168.1.11/System/ErrorInfoMap?ErrorCode=(0 1 2)		

ErrorInfo XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<ErrorInfo><!--req-->
  <ErrorCode><!--req,xs:interger--></ErrorCode>
  <ErrorDescription><!--req,xs:string--></ErrorDescription>
</ErrorInfo>
```

9.2.40 /System/DeviceCap

/System/DeviceCap		General Resource v1.0
GET		Viewer
Description	It is used to get the device capability.	
Query	None	
Inbound Data	None	
Success Return	Device	
Notes:		

With the upgrading of devices and the increase of functions, the following XML is constantly updated.

Device XML Block

```
<?xml version="1.0" encoding="UTF-8" ?>
<Device>
  <AlarmOutCount>0</AlarmOutCount>
  <AlarmInCount>0</AlarmInCount>
  <FunctionList>
    <SD><!--xs:boolean--></SD>
    <WIFI><!--xs:boolean--></WIFI>
    <TwoWayAudio><!--xs:boolean--></ TwoWayAudio >
    <AP><!--xs:boolean--></AP>
    <Wavelink><!--xs:boolean--></Wavelink>
    <Smartlink><!--xs:boolean--></Smartlink>
    <Email><!--xs:boolean--></Email>
    <FTP><!--xs:boolean--></FTP>
    <RemoteRebooe><!--xs:boolean--></RemoteReboot>
    <Restore><!--xs:boolean--></Restore>
    <PasswordModify><!--xs:boolean--></PasswordModify>
  </FunctionList>
  <FunctionListAboutChannel>
    <ChannelList>
      <Channel ChannelID="0">
        <Audio><!--xs:boolean--></Audio>
        <Motion><!--xs:boolean--></Motion>
        <ColorNight><!--xs:boolean--></ColorNight>
        <IntelligentNight><!--xs:boolean--></IntelligentNight>
        <MotionSensitive>
          <Highest><!--xs:boolean--></Highest>
          <Higher><!--xs:boolean--></Higher>
          <High><!--xs:boolean--></High>
          <Middle><!--xs:boolean--></Middle>
          <Low><!--xs:boolean--></Low>
          <Lower><!--xs:boolean--></Lower>
          <Lowest><!--xs:boolean--></Lowest>
        </MotionSensitive>
        <multi_osd>true</multi_osd>
        <mosd_cnt>16</mosd_cnt>
        <mosd_cord>16</mosd_cord>
      </Channel>
    </ChannelList>
  </FunctionListAboutChannel>
</Device>
```

9.2.41 /System/AudioAlarmConfig

/System/AudioAlarmConfig		General Resource	v1.0
GET		Viewer	
Description	Get the sound alarm configuration		
Query	None		
Inbound Data	None		
Success Return	AudioAlarm		
PUT		Viewer	
Description	Set the sound alarm configuration		
Query	None		
Inbound Data	AudioAlarm		
Success Return	ResponseStatus		
Notes:			

AudioAlarm XML

```
<?xml version="1.0" encoding="utf-8"?>
<AudioAlarm><!--req-->
  <Enable><!--req,xs:boolean--></Enable>
  //custom, alarm, dog, warning area, monitoring area, welcome, Deep Water Warning,
  Private territory, danger zone, Valuable objects, High Warning,Private Parking
  <Type><!--req,xs:string,"alarm,dog,custom"--></Type>
  <Delay><!--req,xs:interger,the unit is second--></Delay>
  <Volume><!--req,xs:interger,0--100--></Volume>
</AudioAlarm>
```

9.2.42 /System/TryListenSelectedAlarmAudio

/System/TryListenSelectedAlarmAudio		General Resource	v1.0
PUT		Viewer	
Description	Listen to the currently selected alarm audio		
Query	None		
Inbound Data	AudioAlarm		
Success Return	ResponseStatus		
Notes:			

AudioAlarm XML

```
<?xml version="1.0" encoding="utf-8"?>
<AudioAlarm><!--req-->
  <Enable><!--req,xs:boolean--></Enable>
```

```

//custom, alarm, dog, warning area, monitoring area, welcome, Deep Water Warning,
Private territory, danger zone, Valuable objects, High Warning,Private Parking
<Type><!--req,xs:string,"alarm,dog,custom"--></Type>
<Delay><!--req,xs:interger,the unit is second--></Delay>
<Volume><!--req,xs:interger,0--100--></Volume>
</AudioAlarm>

```

9.2.43 /System/SaveHardwareConfig

/System/SaveHardwareConfig		General Resource	v1.0
PUT		Viewer	
Description	Save hardware configuration		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			

9.2.44 /System/CancelHardwareConfig

/System/CancelHardwareConfig		General Resource	v1.0
PUT		Viewer	
Description	Unconfigure hardware		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			

9.3 /Security

9.3.1 /Security/UserAuth

/Security/UserAuth		General Resource	v1.0
PUT		Viewer	
Description	It is used to verify user information		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		

Notes:

You not need to set the http User authentication information.

Example:

URL: http://192.168.1.46/Security/UserAuth

Data format:

PUT /Security/UserAuth HTTP/1.1\r\n

Connection:close\r\n

Host: 192.168.0.46\r\n

Content-Length:\r\n

\r\n

Username=admin&Password=admin

9.4 /Streams

/Streams		General Resource v1.0
GET		Viewer
Description	It is used to get the properties of streaming channels for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or StreamList	
PUT		Viewer
Description	It is used to update the properties of streaming channels for the device.	
Query	None	
Inbound Data	StreamList	
Success Return	ResponseStatus	
Notes:		

Streams XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<StreamList>
  <Stream><!-- opt -->
</StreamList>
```

9.4.1 /Streams/ChannelID

/Streams/ChannelID		General Resource v1.0
GET		Viewer
Description	It is used to get the properties of a particular	

	streaming channel for the device.
Query	None
Inbound Data	None
Success Return	ResponseStatus or Stream
PUT Viewer	
Description	It is used to set the properties of a particular streaming channel for the device.
Query	None
Inbound Data	Stream
Success Return	ResponseStatus
Notes: The ChannelID in URI shows the channel number. The <ChannelID> is start at 1.	

Stream XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Stream>
  <ID><!--opt,xs:integer--></ID>
  <StreamCapability><!--opt-->
  <StreamConfigList>
    <StreamConfig><!--opt-->
  </StreamConfigList>
</Stream>
```

9.4.1.1 /Streams/ChannelID/ReceiveData

/Streams/ChannelID/StreamID/ReceiveData General Resource v1.0	
GET Viewer	
Description	It is used to get the MJPEG stream for the device.
Query	None
Inbound Data	None
Success Return	ResponseStatus or Stream data
Notes: The channelID in the URI shows the channel number, The <ChannelID> is start at 1.defaultly the first substream is opened.	

9.4.1.2 /Streams/ChannelID/CapabilityV2

/Streams/ChannelID/CapabilityV2 General Resource v1.0	
GET Viewer	

Description	It is used to get the Stream Capability of a particular streaming channel for the device.
Query	None
Inbound Data	None
Success Return	ResponseStatus or StreamCapability
<p>Notes:</p> <p>The <ChannelID> in the URI shows the channel number, The <ChannelID> is start at 1.</p> <p>The <ID> under the <StreamCapability> represents the channel ID.</p> <p>The <ID> under the <MainStreamCapability> indicates the main stream of ID the device supported</p> <p>The <ID> under the <MainStreamCapability>,<SecondStreamCapabilityList>, indicates the sub stream of ID the device supported Resolution and maximum frame rate supported by the main stream is not the same, but different main stream supports different sub-stream: Understanding the structure of the flow capacity,It is not the same, that is to say the resolution and maximum frame rate of the sub-stream by the resolution of the main stream to the decision. Therefore, the design follows the structure to represent the encoding capabilities of the device. For example, a device supports main stream 720P / 25fps and 1080P / 25fps, while the 720P erupted stream supported resolution and maximum frame rate is 360p / 20fps, 1080p stream erupted in support of the resolution and maximum frame rate is D1 / 25fps.</p> <p><u><CodecCapabilityList> describe which codec type support.such as H.264,H265,MJPEG,H.265+,H264+.</u></p>	

StreamCapability XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<StreamCapability>
  <ID><!--req,xs:integer, --></ID>
  <MainStreamCapabilityList><!--ro,req-->
    <MainStreamCapability><!--ro,opt->
      <ID><!--req,xs:integer, --></ID>
      <ResolutionWidth><!--ro,opt,xs:integer--></ResolutionWidth>
      <ResolutionHeigth><!--ro,opt,xs:integer--></ResolutionHeigth>
      <MaxFrameRate><!--ro,req,xs:integer,--></MaxFrameRate>
      <SubStreamCapabilityList><!--ro,req-->
        <StreamCapability><!--ro,opt-->
          <ID><!--ro,req,xs:integer--></ID>
          <ResolutionWidth><!--ro,opt,xs:integer--></ResolutionWidth>
          <ResolutionHeigth><!--ro,opt,xs:integer--></ResolutionHeigth>
          <MaxFrameRate><!--ro,req,xs:integer,--></MaxFrameRate>
        </StreamCapability>
      </SubStreamCapabilityList>
    <SecondStreamCapabilityList><!--ro,req-->
      <StreamCapability><!--ro,opt-->
        <ID><!--ro,req,xs:integer--></ID>
        <ResolutionWidth><!--ro,opt,xs:integer--></ResolutionWidth>
```

```

        <ResolutionHeigth><!--ro,opt,xs:integer--></ResolutionHeigth>
        <MaxFrameRate><!--ro,req,xs:integer,--></MaxFrameRate>
    </StreamCapability>
</SecondStreamCapabilityList>
</MainStreamCapability>
<CodecCapabilityList Version="1.0">
    <streamCodecCapability Version="1.0">
        <ID><!--ro,req,xs:integer--></ID>
        <Codec><!--req,xs:string,"H.264,H.265,MJPEG,H.265+,H.264+"--></Codec>
    </streamCodecCapability>
</CodecCapabilityList>
</MainStreamCapabilityList>
</StreamCapability>

```

9.4.1.3 /Streams/ChannelID/StreamID

/Streams/ChannelID/StreamID		General Resource v1.0
GET		Viewer
Description	It is used to get the StreamID Streaming properties of a particular streaming channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or StreamConfig	
PUT		Viewer
Description	It is used to set the StreamID Streaming properties of a particular streaming channel for the device.	
Query	None	
Inbound Data	StreamConfig	
Success Return	ResponseStatus	
Notes:		
<p>The <ChannelID> in the URI shows the channel number, The <ChannelID> is start at 1.</p> <p>The <StreamID> represents stream type, generally "1" represents the main stream, "2" means "the first sub-Stream", "3" means "the second sub-stream", and so on.</p> <p>The <FrameRate> field indicates the frame rate, if the value is 25500, which indicates that the frame rate is a full frame rate.</p> <p>The <FixedQuality> field represents the quality, the smaller the value, the higher the quality.</p> <p>If <StreamID> corresponds to a sub-stream, then <StreamType>, <VideoCodecType>, <FixedQuality> is invalid, <BitrateType> is always a CBR (constant bit rate).</p> <p><keyFrameInterval> is not supported.</p> <p>Resolution and frame rate ranges refer to the <StreamCapability></p>		

If <StreamID> corresponds to a snapshot stream, then <StreamType>, <VideoCodecType>, <FixedQuality> is invalid, <BitrateType> is always a CBR (constant bit rate).

StreamConfig XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<StreamConfig><!--opt-->
  <ID><!--req,xs:integer--><ID>
  <StreamType><!--req,xs:string,"Video,VideoAndAudio"--></StreamType>

<VideoCodecType><!--ro,opt,xs:string,"H.264,H.265,MJPEG,H.265+,H.264+"--></VideoCodecType>
  <ResolutionWidth><!--req,xs:integer--></ResolutionWidth>
  <ResolutionHeight><!--req,xs:integer--></ResolutionHeight>
  <FrameRate><!--req,xs:integer, maximum frame rate x100--></FrameRate>
  <BitrateType><!--req,xs:string,"CBR,VBR,SmartAVC"--></BitrateType>
  <Bitrate><!--opt, xs:integer, in kbps--></Bitrate>
  <FixedQuality><!--req,xs:integer,"0-5"--></FixedQuality>
  <keyFrameInterval> <!-- opt, xs:integer--> </keyFrameInterval>
  <Transport><!--opt-->
    <RTSPURI><!--ro,req,xs:string--></RTSPURI>
  </Transport>
</StreamConfig>
```

9.4.1.3.1 /Streams/ChannelID/StreamID/Transport

/Streams/ChannelID/StreamID/Transport		General Resource v1.0
GET		Viewer
Description	It is used to get the RTSP Address for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Transport	
Notes:		
<p>The <ChannelID> in the URI shows the channel number, The <ChannelID> is start at 1.</p> <p>The <StreamID> represents stream type, generally "1" represents the main stream, "2" means "the first sub-Stream", and so on.</p> <p>The <RTSPURI> field indicates the address which is access to the stream by the RTSP Protocol.</p>		

Stream Transport XML Block

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

```

<Transport>
  <RTSPURI><!--ro,req,xs:string--></RTSPURI>
</Transport>

```

9.4.1.3.2 /Streams/ChannelID/StreamID/ReceiveData

/Streams/ChannelID/StreamID/ReceiveData		General Resource	v1.0
GET		Viewer	
Description	It is used to get the MJPEG stream for the device.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus or Stream Data		
Notes:			
<p>The <ChannelID> in the URI shows the channel number, The <ChannelID> is start at 1.</p> <p>The <StreamID> represents stream type, generally "1" represents the main stream, "2" means "the first sub-Stream", and so on.</p>			

9.4.2 /Streams/AudioStream

/Streams/AudioStream		General Resource	v1.0
GET		Viewer	
Description	It is used to get audio stream configuration.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus or AudioStream		
PUT		Viewer	
Description	It is used to set audio stream configuration.		
Query	None		
Inbound Data	AudioStream		
Success Return	ResponseStatus		
Notes:			

Audio XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<AudioStream>
  <AudioInMethod><!--rw,xs:string,"microphone,line-in"--></AudioInMethod>
  <AudioInCodec><!--rw,xs:string,"G.711U"--></AudioInCodec>
  <AudioInVolume><!--rw,xs:integer,"0-100"--></AudioInVolume>
  <AudioOutVolume><!--rw,xs:integer,"0-100"--></AudioOutVolume>

```

</AudioStream>

9.5 /Snapshot

9.5.1 /Snapshot/ChannelID/RemoteImageCapture

/Snapshot/ChannelID/RemoteImageCapture		General Resource	v1.0
GET		Viewer	
Description	It is used to get a jpeg picture.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus or the JPEG picture stream		
Notes: The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. "ImageFormat" is image Type such as,bmp,jpg and so on,if "ImageFormat" is 1,the picture you capture is the format of bmp, if "ImageFormat" is 2, It is the format of jpg, default is the format of jpg, so ,you should set "ImageFormat" to 2, if you set "ImageFormat" to other value,maybe you will get a result you do not want to see. Example: http://192.168.1.46/Snapshot/1/RemoteImageCapture?ImageFormat=2			

9.5.2 /Snapshot/ChannelID/StreamID/RemoteImageCaptur

eV2

/Snapshot/ChannelID/StreamID/RemoteImageCaptureV2		General Resource	v1.0
GET		Viewer	
Description	It is used to get a jpeg picture.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus or the JPEG picture stream		
Notes: The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. The <StreamID> represents stream type, generally "1" represents the main stream, "2" means "the first sub-Stream", and so on. "ImageFormat" is image Type such as,bmp,jpg and so on, if you set "ImageFormat" to other value,maybe you will get a result you do not want to see. Example: http://192.168.1.46/Snapshot/1/2/RemoteImageCaptureV2?ImageFormat=bmp jpg			

9.5.3 /Snapshot/Capability

/Snapshot/Capability		General Resource	v1.0
GET		Viewer	
Description	It is used to get the picture capability.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus or SnapshotCapabilityList		
Notes:			

SnapshotConfig XML Block

```
<SnapshotCapabilityList><!--ro,req-->
  <SnapshotCapability><!--ro,opt-->
    <ID><!--ro,req,xs:integer--></ID>
    <ResolutionWidth><!--ro,req,xs:integer--></ResolutionWidth>
    <ResolutionHeight><!--ro,req,xs:integer--></ResolutionHeight>
  </SnapshotCapability>
</SnapshotCapabilityList>
```

9.5.4 /Snapshot/Config

/Snapshot/Config		General Resource	v1.0
GET		Viewer	
Description	It is used to get a Snapshot Config.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus or SnapshotConfig		
PUT		Viewer	
Description	It is used to set a Snapshot Config.		
Query	None		
Inbound Data	SnapshotConfig		
Success Return	ResponseStatus		
Notes:			

SnapshotConfig XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<SnapshotConfig>
  <ResolutionWidth><!--rw,req,xs:interger--></ResolutionWidth>
  <ResolutionHeight><!--rw,req,xs:interger--></ResolutionHeight>
```

```

<Quality><!--rw,req,xs:string,"high,middle,low" --></Quality>
<SnapshotCount><!--rw,req,xs:integer,"1,2,3" --></SnapshotCount>
<SnapshotIntervalTime><!--rw,req,xs:interger,in sec,"1,2,3,4,5" -->
</ SnapshotIntervalTime>
<StreamType><!--rw,req,xs:string,"main,sub,hdsb"--></StreamType>
</SnapshotConfig>

```

9.6 /Images

/Images		General Resource v1.0
GET		Viewer
Description	It is used to get All Image Configuration for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or ImageList	
PUT		Viewer
Description	It is used to set All Image Configuration for the device.	
Query	None	
Inbound Data	ImageList	
Success Return	ResponseStatus	
Notes:		

ImageList XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<ImageList><!--req-->
  <Image><!--opt-->
</ImageList>

```

9.6.1 /Images/ChannelID

/Images/ ChannelID		General Resource v1.0
GET		Viewer
Description	It is used to get All Image Configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Image	
PUT		Viewer

Description	It is used to set All Image Configuration of a particular channel for the device.
Query	None
Inbound Data	Image
Success Return	ResponseStatus
Notes: The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.	

Image XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Image>
  <ID><!--req,xs:integer--></ID>
  <Scene><!--opt-->
  <Basic/><!--opt-->
  <Color/><!--opt-->
  <Exposure/><!--opt-->
  <WhiteBlance/><!--opt-->
  <IrCutFillter><!--opt-->
    //LDC Advance, range: 0-100
    <LDCAdvance><!--req,xs:integer--></LDCAdvance>
  </IrCutFillter>
</Image>
```

9.6.1.1 /Images/ChannelID/Scene

/Images/ChannelID/Scene		General Resource v1.0
GET		Viewer
Description	It is used to get the scene configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Scene	
PUT		Viewer
Description	It is used to set the scene configuration of a particular channel for the device.	
Query	None	
Inbound Data	Scene	
Success Return	ResponseStatus	
Notes: The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.		

Scene XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<Scene>
  <Support><!--ro,req,xs:boolean></Support>
  <Mode><!--req,xs:string,"standard,soft,vivid,custom"></Mode>
  <LDC><!--req,xs:boolean ></LDC>
  <Fog><!--req,xs:boolean ></Fog>
  <FogValue><!--rw,req,xs:interger,0-100--></FogValue>
  <Corridor><!--req,xs:boolean ></Corridor>
  <Gamma><!--rw,req,xs:interger,0-100--></Gamma>
</Scene>

```

9.6.1.2 /Images/ChannelID/Basic

/Images/ChannelID/Basic		General Resource v1.0
GET		Viewer
Description	It is used to get the basic configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Basic	
PUT		Viewer
Description	It is used to set the basic configuration of a particular channel for the device.	
Query	None	
Inbound Data	Basic	
Success Return	ResponseStatus	
Notes:		

Basic XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<Basic>
  <powerLineFrequencyMode><!--req,xs:string,50HZ,60HZ --></ powerLineFrequencyMode>
  <NoiseReduce><!--req,xs:string,"close,weak,normal,strong" --></NoiseReduce>
  <BLC><!--req,xs:string,"close,open,notsupport" --></BLC>
  <WDR><!--req,xs:string, "close,weak,normal,strong" --></WDR>
  <FlipMode>
    <!--req,xs:string,"close,leftright,updown,center,notsupport" -->
  </FlipMode>
</Basic>

```

9.6.1.3 /Images/ChannelID/Color

/Images/ChannelID/Color		General Resource v1.0
GET		Viewer
Description	It is used to get the color configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Color	
PUT		Viewer
Description	It is used to set the color configuration of a particular channel for the device.	
Query	None	
Inbound Data	Color	
Success Return	ResponseStatus	
Notes:		
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p><Mode> shows automatically adjust the brightness. When the mode is "auto", setting the value of the <Brightness> field are unsuccessed, when the mode is "manual", setting the value of the <Brightness> field is valid. For unsupported devices, setting the value of the <Brightness> field is always valid.</p>		

Color XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Color>
  <Mode><!--req,xs:string,"auto,manual,notsupport"--></Mode>
  <Brightness><!--req,xs:integer,0-100--></Bright>
  <Contrast><!--req,xs:integer,0-100--></Contrast>
  <Saturation><!--req,xs:integer,0-100--></Saturation>
  <Hue><!--req,xs:integer,0-100--></Hue>
  <Sharpness><!--req,xs:integer,0-100--></Sharpness>
</Color>
```

9.6.1.4 /Images/ChannelID/Exposure

/Images/ChannelID/Exposure		General Resource v1.0
GET		Viewer
Description	It is used to get the exposure configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	

Success Return	ResponseStatus or Exposure
PUT Viewer	
Description	It is used to set the exposure configuration of a particular channel for the device.
Query	None
Inbound Data	Exposure
Success Return	ResponseStatus
Notes: The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.	

Exposure XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Exposure>
  <IrisType><!--req,xs:string,"manual, DC-Iris"--></IrisType>
  <shutter>
    <!--req,xs:string,"auto, 1/1, 1/3, 1/8, 1/12, 1/15,1/25,1/30,1/50,1/60,1/100, 1/120,
    1/240, 1/480, 1/960, 1/1000, 1/2000, 1/4000, 1/8000, 1/10000,"-->
  </shutter>
  <Gain><!--req,xs:string,"auto,6dB,12dB,18dB,24dB,30dB,36dB,42dB" --></Gain>
  <Shuttermode><!--req,xs:string,"auto,manual"--></Shuttermode>
</Exposure>
```

9.6.1.5 /Images/ChannelID/WhiteBalance

/Images/ChannelID/WhiteBalance		General Resource v1.0
GET		Viewer
Description	It is used to get the whitebalance configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or WhiteBlance	
PUT		Viewer
Description	It is used to set the whitebalance configuration of a particular channel for the device.	
Query	None	
Inbound Data	WhiteBlance	
Success Return	ResponseStatus	
Notes: The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. When the <Mode> is "manual",<GainRed> and <GainBlue> are effective ,otherwise are not effective.		

WhiteBlance XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<WhiteBlance>
  <Mode>
    <!--req,xs:string,"auto,incandescent,warm,fluorescent,natural>manual,
    locked"-->
  </ Mode>
  <GainRed><!--dep,xs:integer,0-100--></GainRed>
  <GainBlue><!--dep,xs:integer,0-100--></GainBlue>
</WhiteBlance>
```

9.6.1.6 /Images/ChannelID/IrCutFilter

/Images/ChannelID/IrCutFilter		General Resource v1.0
GET		Viewer
Description	It is used to get the IrCutFilter configuration of a particular channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or IrCutFilter	
PUT		Viewer
Description	It is used to set the IrCutFilter configuration of a particular channel for the device.	
Query	None	
Inbound Data	IrCutFilter	
Success Return	ResponseStatus	
Notes:		
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>When the <Mode> is "time",<StartTime> and <EndTime> are effective,other is not effective.When the <Mode> is "initiative" or "passivity", <Sensitivity> and <SwitchTime> are not effective ,other is not effective.</p> <p>The greater the value of the <Sensitivity> ,the more sensitive.</p>		

IrCutFilter XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<IrCutFilter>
  <Mode><!--req,xs:string,"initiative,passivity(auto),day,night,time,color,intelligent"--></Mode>
  <DayStartTime>
    <!--dep,xs:time, "17:30:08"-->
  </ DayStartTime >
```

```

<DayEndTime>
  <!--dep,xs:time," 17:30:08"-->
</ DayStartTime >
<Sensitivity><!--dep,xs:integer,0-100--></Sensitivity>
<SwitchTime><!--dep,xs:integer,3s-15s--></SwitchTime>
<ColorLastTime><!--dep,xs:integer,3s-15s--></ColorLastTime>
</lrCutFilter>

```

9.7 /Pictures

/Pictures		General Resource v1.0
GET		Viewer
Description	It is used to get all picture configuration for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or PictureList	
PUT		Viewer
Description	It is used to set all picture configuration for the device.	
Query	None	
Inbound Data	PictureList	
Success Return	ResponseStatus	
Notes:		

PictureList XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<PictureList>
  <Picture><!--opt-->
</PictureList>

```

9.7.1 /Pictures/ChannelID

/Pictures/ChannelID		General Resource v1.0
GET		Viewer
Description	It is used to get all picture configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Picture	

PUT		Viewer
Description	It is used to set all picture configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	Picture	
Success Return	ResponseStatus	
Notes:		
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.		

Picture XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Picture>
  <ID><!--req,xs:integer--></ID>
  <OSD/><!--req-->
  <VideoLost/><!--req-->
  <Motion/><!--req-->
  <Mosaic/><!--req-->
</Picture>
```

9.7.1.1 /Pictures/ChannelID/OSD

/Pictures/ChannelID/OSD		General Resource v1.0
GET		Viewer
Description	It is used to get the OSD configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or OSD	
PUT		Viewer
Description	It is used to set the OSD configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	OSD	
Success Return	ResponseStatus	
Notes:		
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.		
In setting the OSD configuration, do not let the coordinates of time and the coordinates of name are the same. In addition, the accuracy of the coordinate values of the OSD is 8, that is to say if these numbers are set 8,9,10,11,12,13,14,15 past, when the re-acquired still 8.		

OSD XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<OSD>
  <DisplayTime><!--req-->
    <Enable><!--req,xs:boolean--></Enable>
    <PosX><!--req,xs:integer,0-703--></PosX>
    <PosY><!--req,xs:integer,0-575--></PosY>
  </DisplayTime>
  <DisplayName><!--req-->
    <Enable><!--req,xs:boolean--></Enable>
    <PosX><!--req,xs:integer,0-703--></PosX>
    <PosY><!--req,xs:integer,0-575--></PosY>
    <Name><!--req,xs:string,"in Unicode"--></Name>
  </DisplayName>
</OSD>

```

9.7.1.2 /Pictures/ChannelID/VideoLost

/Pictures/ChannelID/VideoLost		General Resource v1.0
GET		Viewer
Description	It is used to get the Video Lost configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or VideoLost	
PUT		Viewer
Description	It is used to set the VideoLost configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	VideoLost	
Success Return	ResponseStatus	
Notes:		
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>For video loss, some devices are supported, some devices are not supported,For non-supported devices, though it can get some configurations, in fact, it does not make sense</p>		

VideoLost XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<VideoLost>
  <Enable><!--req,xs:boolean--></Enable>
  <Trigger><!--req-->
  <Schedule><!--req-->
</VideoLost>

```

9.7.1.2.1 /Pictures/ChannelID/VideoLost/Trigger

/Pictures/ChannelID/VideoLost/Trigger		General Resource v1.0
GET		Viewer
Description	It is used to get the Video Lost Trigger configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Trigger	
PUT		Viewer
Description	It is used to set the Video Lost Trigger configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	Trigger	
Success Return	ResponseStatus	
Notes:		
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>The format of <AlarmOutMask>,<SnapshotMask>,<RecordMask> is similar to "11111111" or "00000000", Each bit represents whether a channel is linkage or not, NO.1 means linkage, NO.0 means not linkage, for example, the string "11111111" Represents that the device is 8 channels, from left to right, represent channel 1,channel 2,channel 3..... and so on.</p> <p>For single-channel device, <AlarmOutMask>, <SnapshotMask>, <RecordMask>, <PTZActionList> is not effective .</p>		

VideoLost Trigger XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Trigger>
  <NotifyAMS>
    <Enable><!--req,xs:boolean--></Enable>
  </NotifyAMS>
  <BeepAlert>
    <Enable><!--req,xs:boolean--></Enable>
  </BeepAlert>
  <FullScreen>
    <Enable><!--req,xs:boolean--></Enable>
  </FullScreen>
  <Mail>
    <Enable><!--req,xs:boolean--></Enable>
  </Mail>
```

```

<AlarmOut>
  <Enable><!--req,xs:boolean--></Enable>
  <AlarmOutMask>
    <!--req,xs:string,"11111111111111"-->
  </AlarmOutMask>
</AlarmOut>
<Snapshot>
  <Enable><!--req,xs:boolean--></Enable>
  <SnapshotMask>
    <!--req,xs:string,"11111111111111"-->
  </SnapshotMask>
</Snapshot>
<Record>
  <Enable><!--req,xs:boolean--></Enable>
  <RecordMask>
    <!--req,xs:string,"11111111111111"-->
  </RecordMask>
</Record>
<PTZ>
  <Enable><!--req,xs:boolean--></Enable>
  <PTZActionList>
    <PTZAction>
      <ChannelID><req,xs:integer></ChannelID>
      <ActionName><!-- req, xs:string,"none, preset,
      pattern, patrol" --></ActionName>
      <ActionNum><!--req,xs:integer,depend on <ActionName--></ActionNum>
    </PTZAction>
  </PTZActionList>
</PTZ>
</Trigger>

```

9.7.1.2.2 /Pictures/ChannelID/VideoLost/Schedule

/Pictures/ChannelID/VideoLost/Schedule		General
Resource v1.0		
GET		Viewer
Description	It is used to get the Video Lost schedule configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Schedule	

	particular video input channel for the device.
Query	None
Inbound Data	Motion
Success Return	ResponseStatus
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p><Sensitive> Some versions support for sensitivity of Highest, Higher, high,middle,low,Lower and lowest,Some versions only support for sensitivity of high,middle,low.</p>	

Motion XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Motion>
  <Enable><!--req,xs:boolean--></Enable>
  <Sensitive><!--req,xs:string,"lowest,lower,low,middle,high,higher,highest",--></Sensitive>
  <Trigger><!--opt-->
  <Schedule><!--opt-->
  <MotionRegionList><!--opt-->
</Motion>
```

9.7.1.3.1 /Pictures/ChannelID/Motion/Trigger

/Pictures/ChannelID/Motion/Trigger		General Resource v1.0
GET		Viewer
Description	It is used to get the motion trigger configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Trigger	
PUT		Viewer
Description	It is used to set the motion trigger configuration of a particular video input channel for the device	
Query	None	
Inbound Data	Trigger	
Success Return	ResponseStatus	
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>The format of <AlarmOutMask>,<SnapshotMask>,<RecordMask> is similar to "11111111" or "00000000", each bit represents whether a channel is linkage or not, 1 means linkage,0 means not linkage, for example, the string "11111111" represents that the device is 8 channels, from left</p>		

to right, represent channel 1,channel 2,channel 3..... and so on.

For single-channel device, <AlarmOutMask>, <SnapshotMask>, <RecordMask> and<PTZActionList> are not effective .

Motion Trigger XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Trigger>
  <NotifyAMS>
    <Enable><!--req,xs:boolean--></Enable>
  </NotifyAMS>
  <BeepAlert>
    <Enable><!--req,xs:boolean--></Enable>
  </BeepAlert>
  <FullScreen>
    <Enable><!--req,xs:boolean--></Enable>
  </FullScreen>
  <Mail>
    <Enable><!--req,xs:boolean--></Enable>
  </Mail>
  <AlarmOut>
    <Enable><!--req,xs:boolean--></Enable>
    <AlarmOutMask><!--req,xs:string," 1111111111111111"--></AlarmOutMask>
  </AlarmOut>
  <Snapshot>
    <Enable><!--req,xs:boolean--></Enable>
    <SnapshotMask><!--req,xs:string," 1111111111111111"--></SnapshotMask>
  </Snapshot>
  <Record>
    <Enable><!--req,xs:boolean--></Enable>
    <RecordMask><!--req,xs:string," 1111111111111111"--></RecordMask>
  </Record>
  <PTZ>
    <Enable><!--req,xs:boolean--></Enable>
    <ptzChannelID> <!--req, xs:string --></ptzChannelID>
    <ActionName><!-- req, xs:string,"none, preset, pattern, patrol" --></ActionName>
    <ActionNum><!--req,xs:integer,depend on <ActionName>--></ActionNum>
  </PTZ>
</Trigger>
```

9.7.1.3.2 /Pictures/ChannelID/Motion/Regions

GET		Viewer
Description	It is used to get the motion regions configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Regions	
PUT		Viewer
Description	It is used to set the motion regions configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	Regions	
Success Return	ResponseStatus	
Notes: <p style="color: red;">The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. Use CGI or Use IE to set the motion detection area, these two approaches are independently of each other.</p>		

Motion Regions XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<MotionRegionList>
  <Region><!--opt-->
    <ID><!--req,xs:integer--></ID>
    <TopLeftX><!--req,xs:integer,0-703--></TopLeftX>
    <TopLeftY><!--req,xs:integer,0-575--></TopLeftY>
    <BottomRightX><!--req,xs:integer,0-703--></BottomRightX>
    <BottomRightY><!--req,xs:integer,0-575--></BottomRightY>
  </Region>
</MotionRegionList>
```

9.7.1.3.3 /Pictures/ChannelID/Motion/RegionsV2

/Pictures/ChannelID/Motion/RegionsV2		General Resource v1.0
GET		Viewer
Description	It is used to get the motion regions configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or MotionRegionListV2	

PUT		Viewer
Description	It is used to set the motion regions configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	MotionRegionListV2	
Success Return	ResponseStatus	
Notes:		
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.		

Motion RegionsV2 XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<MotionRegionListV2>
  <RegionX_0><!--xs:string,"00000000000000"--></RegionX_0>
  <RegionX_1><!--xs:string,"00000000000000"--></RegionX_1>
  <RegionX_2><!--xs:string,"00000000000000"--></RegionX_2>
  <RegionX_3><!--xs:string,"00000000000000"--></RegionX_3>
  <RegionX_4><!--xs:string,"00000000000000"--></RegionX_4>
  <RegionX_5><!--xs:string,"00000000000000"--></RegionX_5>
  <RegionX_6><!--xs:string,"00000000000000"--></RegionX_6>
  <RegionX_7><!--xs:string,"00000000000000"--></RegionX_7>
  <RegionX_8><!--xs:string,"00000000000000"--></RegionX_8>
  <RegionX_9><!--xs:string,"00000000000000"--></RegionX_9>
  <RegionX_10><!--xs:string,"00000000000000"--></RegionX_10>
  <RegionX_11><!--xs:string,"00000000000000"--></RegionX_11>
</MotionRegionListV2>
```

9.7.1.3.4 /Pictures/ChannelID/Motion/Schedule

/Pictures/ChannelID/Motion/Schedule		General Resource
v1.0		
GET		Viewer
Description	It is used to get the motion schedule configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Schedule	
PUT		Viewer
Description	It is used to set the motion schedule configuration of a particular video input channel for the device.	

Success Return	ResponseStatus
Notes:	

Motion URL XML Block

```
<URL>
  <Protocol><!--req,xs:string--></Protocol>
  <Host><!--req,xs:string></Host>
  <Port><!--req,xs:string></Port >
  <Path><!--req,xs:string></Path >
  <Description><!--req,xs:string></Description >
  <Username><!--req,xs:string></Username >
  <Password><!--req,xs:string></Password>
</URL>
```

9.7.1.4/Pictures/ChannelID/Mosaic

/Pictures/ChannelID/Mosaic		General Resource v1.0
GET		Viewer
Description	It is used to get the mosaic video input configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or Mosaic	
PUT		Viewer
Description	It is used to set the mosaic video input configuration of a particular video input channel for the device.	
Query	None	
Inbound Data	Mosaic	
Success Return	ResponseStatus	
Notes:		
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.		

Mosaic XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Mosaic>
  <Enable><!--req,xs:boolean--></Enable>
  <MosaicRegionList>
    <Region><!--opt-->
      <ID><!--req,xs:integer--></ID>
```

```

<TopLeftX><!--req,xs:integer,0-703--></TopLeftX>
<TopLeftY><!--req,xs:integer,0-575--></TopLeftY>
<BottomRightX><!--req,xs:integer,0-703--></BottomRightX>
<BottomRightY><!--req,xs:integer,0-575--></BottomRightY>

</Region>
</MosaicRegionList>
</Mosaic>

```

9.8 /PTZ

/PTZ	General Resource	v1.0
Notes:		

9.8.1 /PTZ/ChannelID/Presets/Set

/PTZ/ChannelID/Presets/Set		General Resource	v1.0
PUT		Viewer	
Description	It is used to set a preset.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is one param,the "Param1" is preset number.the range of value is 0—127.</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Presets/Set</p> <p>Data format:</p> <pre> PUT /PTZ/1/Presets/Set HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1 </pre>			

9.8.2 /PTZ/ChannelID/Presets/Remove

/PTZ/ChannelID/Presets/Remove		General Resource	v1.0
PUT		Viewer	
Description	It is used to remove a preset.		
Query	None		

Inbound Data	None
Success Return	ResponseStatus
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is preset number.the range of value is 0—127. some devices is not support for this URL.</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Presets/Remove</p> <p>Data format:</p> <pre>PUT /PTZ/1/Presets/Remove HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</pre>	

9.8.3 /PTZ/ChannelID/Presets/Goto

/PTZ/ChannelID/Presets/Goto		General Resource	v1.0
PUT		Viewer	
Description	It is used to goto a preset		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is preset number.the range of value is 0—127.</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Presets/Goto</p> <p>Data format:</p> <pre>PUT /PTZ/1/Presets/Goto HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</pre>			

9.8.4 /PTZ/ChannelID/Cruise/StartCruiseMem

/PTZ/ChannelID/Cruise/StartCruiseMem		General Resource	v1.0
PUT		Viewer	

Description	It is used to start to record cruise point.
Query	None
Inbound Data	None
Success Return	ResponseStatus
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is Cruise number.the range of value is 0—15. Example: URL: http://192.168.1.46/PTZ/1/Cruise/StartCuriseMen Data format: PUT /PTZ/1/Cruise/StartCuriseMen HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</p>	

9.8.5 /PTZ/ChannelID/Cruise/StopCruiseMem

/PTZ/ChannelID/Cruise/StopCruiseMem		General Resource
v1.0		
PUT	Viewer	
Description	It is used to stop to record cruise point.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is Cruise number.the range of value is 0—15. Example: URL: http://192.168.1.46/PTZ/1/Cruise/StopCuriseMen Data format: PUT /PTZ/1/Cruise/StopCuriseMen HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</p>		

9.8.6 /PTZ/ChannelID/Cruise/ClearCruise

/PTZ/ChannelID/Cruise/ClearCruise		General Resource	v1.0
PUT		Viewer	
Description	It is used to Clear presets in a Cruise.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is Cruise number.the range of value is 0—15. some devices is not support for this URL.</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Cruise/ClearCruise</p> <p>Data format:</p> <pre>PUT /PTZ/1/Cruise/ClearCruise HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</pre>			

9.8.7 /PTZ/ChannelID/Cruise/AddPresetToCruise

/PTZ/ChannelID/Cruise/AddPresetToCruise		General Resource	v1.0
PUT		Viewer	
Description	It is used to add a preset to cruise.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is four params:</p> <ul style="list-style-type: none"> the "Param1" is cruise number.the range of value is 0—15. the "Param2" is preset number, the range of value is 0--127 the "Param3" is time to stay.It is in second. the "Param4" is moved speed,.the range of value is 1-10. <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Cruise/AddPresetToCruise</p> <p>Data format:</p> <pre>PUT /PTZ/1/Cruise/AddPresetToCruise HTTP/1.1\r\n</pre>			

```

Connection:close\r\n
Host: 192.168.0.46\r\n
Content-Length:\r\n
\r\n
Param1=1&Param2=1&Param3=1&Param4=1

```

9.8.8 /PTZ/ChannelID/Cruise/StartCruise

/PTZ/ChannelID/Cruise/StartCruise		General Resource	v1.0
PUT		Viewer	
Description	It is used to start cruise.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is Cruise number.the range of value is 0—15. Example: URL: http://192.168.1.46/PTZ/1/Cruise/StartCruise Data format: PUT /PTZ/1/Cruise/StartCruise HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</p>			

9.8.9 /PTZ/ChannelID/Cruise/StopCruise

/PTZ/ChannelID/Cruise/StopCruise		General Resource	v1.0
PUT		Viewer	
Description	It is used to stop cruise.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is Cruise number.the range of value is 0—15. Example: URL: http://192.168.1.46/PTZ/1/Cruise/StopCruise</p>			

Data format:
 PUT /PTZ/1/Cruise/StopCruise HTTP/1.1\r\n
 Connection:close\r\n
 Host: 192.168.0.46\r\n
 Content-Length:\r\n
 \r\n
 Param1=1

9.8.10 /PTZ/ChannelID/Track/StartTrackMem

/PTZ/ChannelID/Track/StartTrackMem		General Resource	v1.0
PUT		Viewer	
Description	It is used to start to record the track.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. There is one param,the "Param1" is track number.the range of value is 0—15. Example: URL: http://192.168.1.46/PTZ/1/Track/StartTrackMem Data format: PUT /PTZ/1/Track/StartTrackMem HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</p>			

9.8.11 /PTZ/ChannelID/Track/StopTrackMem

/PTZ/ChannelID/Track/StopTrackMem		General Resource	v1.0
PUT		Viewer	
Description	It is used to stop to record the track.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1. Example:</p>			

```

URL: http://192.168.1.46/PTZ/1/Track/StopTrackMem
Data format:
  PUT /PTZ/1/Track/StopTrackMem HTTP/1.1\r\n
  Connection:close\r\n
  Host: 192.168.0.46\r\n
  Content-Length:\r\n
  \r\n
  Param1=1

```

9.8.12 /PTZ/ChannelID/Track/StartTrack

/PTZ/ ChannelID/Track/StartTrack		General Resource	v1.0
PUT		Viewer	
Description	It is used to start track.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Track/StartTrack</p> <p>Data format:</p> <pre> PUT /PTZ/1/Track/StartTrack HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1 </pre>			

9.8.13 /PTZ/ChannelID/Track/StopTrack

/PTZ/ChannelID/Track/StopTrack		General Resource	v1.0
PUT		Viewer	
Description	It is used to stop track.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is one param,the "Param1" is track number.the range of value is 0—15.</p>			

Example:

URL: http://192.168.1.46/PTZ/1/Track/StopTrack

Data format:

PUT /PTZ/1/Track/StopTrack HTTP/1.1\r\n

Connection:close\r\n

Host: 192.168.0.46\r\n

Content-Length:\r\n

\r\n

Param1=1

9.8.14 /PTZ/ChannelID/FocusFar

/PTZ/ChannelID/FocusFar		General Resource	v1.0
PUT			Viewer
Description	It is used to adjust focus far.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.			
There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:			
Example:			
URL: http://192.168.1.46/PTZ/1/FocusFar			
Data format:			
PUT /PTZ/1/FocusFar HTTP/1.1\r\n			
Connection:close\r\n			
Host: 192.168.0.46\r\n			
Content-Length:\r\n			
\r\n			
Param1=1&Param2=3			

9.8.15 /PTZ/ChannelID/FocusNear

/PTZ/ChannelID/FocusNear		General Resource	v1.0
PUT			Viewer
Description	It is used to adjust focus near.		
Query	None		
Inbound Data	None		

Success Return	ResponseStatus
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is “Param1”,the name of the second param is “Param2”,if the “Param1” is “1” , it is present to start,otherwise,it is present to stop, the “Param2” is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/FocusNear</p> <p>Data format:</p> <pre>PUT /PTZ/1/FocusNear HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3</pre>	

9.8.16 /PTZ/ChannelID/ApertureBigger

/PTZ/ChannelID/ApertureBigger		General Resource v1.0
PUT		Viewer
Description	It is used to adjust aperture bigger.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is “Param1”,the name of the second param is “Param2”,if the “Param1” is “1” , it is present to start,otherwise,it is present to stop, the “Param2” is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/ApertureBigger</p> <p>Data format:</p> <pre>PUT /PTZ/1/ApertureBigger HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3</pre>		

9.8.17 /PTZ/ChannelID/ApertureSmaller

/PTZ/ChannelID/ApertureSmaller		General Resource v1.0
PUT		Viewer
Description	It is used to adjust aperture smaller	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: <code>http://192.168.1.46/PTZ/1/ApertureSmaller</code></p> <p>Data format:</p> <pre>PUT /PTZ/1/ApertureSmaller HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3</pre>		

9.8.18 /PTZ/ChannelID/ZoomIn

/PTZ/ChannelID/ZoomIn		General Resource v1.0
PUT		Viewer
Description	It is used to adjust zoom in.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: <code>http://192.168.1.46/PTZ/1/ZoomIn</code></p> <p>Data format:</p> <pre>PUT /PTZ/1/ZoomIn HTTP/1.1\r\n Connection:close\r\n</pre>		

```
Host: 192.168.0.46\r\n
Content-Length:\r\n
\r\n
Param1=1&Param2=3
```

9.8.19 /PTZ/ChannelID/ZoomOut

/PTZ/ChannelID/ZoomOut		General Resource	v1.0
PUT		Viewer	
Description	It is used to adjust zoom out.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/ZoomOut</p> <p>Data format:</p> <pre>PUT /PTZ/1/ZoomOut HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3</pre>			

9.8.20 /PTZ/ChannelID/TurnLeft

/PTZ/ChannelID/TurnLeft		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn left the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the</p>			

“Param2” is the speed, the range of the value is 1-10, the Format as shown:

Example:

URL: `http://192.168.1.46/PTZ/1/TurnLeft`

Data format:

`PUT /PTZ/1/TurnLeft HTTP/1.1\r\n`

`Connection:close\r\n`

`Host: 192.168.0.46\r\n`

`Content-Length:\r\n`

`\r\n`

`Param1=1&Param2=3`

9.8.21 /PTZ/ChannelID/TurnRight

/PTZ/ChannelID/TurnRight		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn right the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.			
There is two params. The name of the first param is “Param1”,the name of the second param is “Param2”,if the “Param1” is “1” , it is present to start,otherwise,it is present to stop, the “Param2” is the speed, the range of the value is 1-10, the Format as shown:			
Example:			
URL: <code>http://192.168.1.46/PTZ/1/TurnRight</code>			
Data format:			
<code>PUT /PTZ/1/TurnRight HTTP/1.1\r\n</code>			
<code>Connection:close\r\n</code>			
<code>Host: 192.168.0.46\r\n</code>			
<code>Content-Length:\r\n</code>			
<code>\r\n</code>			
<code>Param1=1&Param2=3</code>			

9.8.22 /PTZ/ChannelID/TurnUp

/PTZ/ChannelID/TurnUp		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn up the video		
Query	None		

Inbound Data	None
Success Return	ResponseStatus
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/TurnUp</p> <p>Data format:</p> <p>PUT /PTZ/1/ TurnUp HTTP/1.1\r\n</p> <p>Connection:close\r\n</p> <p>Host: 192.168.0.46\r\n</p> <p>Content-Length:\r\n</p> <p>\r\n</p> <p>Param1=1&Param2=3</p>	

9.8.23 /PTZ/ChannelID/TurnDown

/PTZ/ChannelID/TurnDown		General Resource v1.0
PUT		Viewer
Description	It is used to turn down the video	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/TurnDown</p> <p>Data format:</p> <p>PUT /PTZ/1/TurnDown HTTP/1.1\r\n</p> <p>Connection:close\r\n</p> <p>Host: 192.168.0.46\r\n</p> <p>Content-Length:\r\n</p> <p>\r\n</p> <p>Param1=1&Param2=3</p>		

9.8.24 /PTZ/ChannelID/TurnUpLeft

/PTZ/ChannelID/TurnDown		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn down the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: <code>http://192.168.1.46/PTZ/1/TurnUpLeft</code></p> <p>Data format:</p> <pre>PUT /PTZ/1/TurnDown HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3</pre>			

9.8.25 /PTZ/ChannelID/TurnUpRight

/PTZ/ChannelID/TurnDown		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn down the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: <code>http://192.168.1.46/PTZ/1/TurnUpRight</code></p> <p>Data format:</p> <pre>PUT /PTZ/1/TurnDown HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n</pre>			

```
Content-Length:\r\n
\r\n
Param1=1&Param2=3
```

9.8.26 /PTZ/ChannelID/TurnDownLeft

/PTZ/ChannelID/TurnDown		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn down the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/TurnDownLeft</p> <p>Data format:</p> <pre>PUT /PTZ/1/TurnDown HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3</pre>			

9.8.27 /PTZ/ChannelID/TurnDownRight

/PTZ/ChannelID/TurnDown		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn down the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/TurnDownRight</p>			

```

Data format:
PUT /PTZ/1/TurnDown HTTP/1.1\r\n
Connection:close\r\n
Host: 192.168.0.46\r\n
Content-Length:\r\n
\r\n
Param1=1&Param2=3

```

9.8.28 /PTZ/ChannelID/AutoLeftRight

/PTZ/ChannelID/TurnDown		General Resource	v1.0
PUT		Viewer	
Description	It is used to turn down the video		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is two params. The name of the first param is "Param1",the name of the second param is "Param2",if the "Param1" is "1" , it is present to start,otherwise,it is present to stop, the "Param2" is the speed, the range of the value is 1-10, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/AutoLeftRight</p> <p>Data format:</p> <pre> PUT /PTZ/1/TurnDown HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1&Param2=3 </pre>			

9.8.29 /PTZ/ChannelID/Auxiliary

/PTZ/ChannelID/Auxiliary		General Resource	v1.0
PUT		Viewer	
Description	It is used to open the auxiliary.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
<p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p>			

There is one param that is "Param1" ,if the "Param1" is "1" , it is present to open,otherwise,it is present to close, the Format as shown:

Example:

URL: http://192.168.1.46/PTZ/1/Auxiliary

Data format:

PUT /PTZ/1/Auxiliary HTTP/1.1\r\n

Connection:close\r\n

Host: 192.168.0.46\r\n

Content-Length:\r\n

\r\n

Param1=1

9.8.30 /PTZ/ChannelID/Light

/PTZ/ChannelID/Light		General Resource	v1.0
PUT		Viewer	
Description	It is used to open the light.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes:			
The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.			
There is one param that is "Param1" ,if the "Param1" is "1" , it is present to open,otherwise,it is present to close, the Format as shown:			
Example:			
URL: http://192.168.1.46/PTZ/1/Light			
Data format:			
PUT /PTZ/1/Light HTTP/1.1\r\n			
Connection:close\r\n			
Host: 192.168.0.46\r\n			
Content-Length:\r\n			
\r\n			
Param1=1			

9.8.31 /PTZ/ChannelID/RainBrush

/PTZ/ChannelID/RainBrush		General Resource	v1.0
PUT		Viewer	
Description	It is used to open the rain brush.		
Query	None		

Inbound Data	None
Success Return	ResponseStatus
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is one params that is "Param1" ,if the "Param1" is "1" , it is present to open,otherwise,it is present to close, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/RainBrush</p> <p>Data format:</p> <pre>PUT /PTZ/1/RainBrush HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</pre>	

9.8.32 /PTZ/ChannelID/Fan

/PTZ/ChannelID/Fan		General Resource	v1.0
PUT		Viewer	
Description	It is used to open the fan.		
Query	None		
Inbound Data	None		
Success Return	ResponseStatus		
Notes: <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is one params that is "Param1" ,if the "Param1" is "1" , it is present to open,otherwise,it is present to close, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Fan</p> <p>Data format:</p> <pre>PUT /PTZ/1/Fan HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</pre>			

9.8.33 /PTZ/ChannelID/Hearter

/PTZ/ChannelID/Hearter		General Resource v1.0
PUT		Viewer
Description	It is used to open the hearter.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p>There is one params that is "Param1" ,if the "Param1" is "1" , it is present to open,otherwise,it is present to close, the Format as shown:</p> <p>Example:</p> <p>URL: http://192.168.1.46/PTZ/1/Hearter</p> <p>Data format:</p> <pre>PUT /PTZ/1/Hearter HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1</pre>		

9.8.34 /PTZ/ChannelID/Config

/PTZ/ChannelID/Config		General Resource v1.0
GET		Viewer
Description	It is used to get PTZ configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or PTZConfigChannel	
PUT		Viewer
Description	It is used to close the hearter	
Query	None	
Inbound Data	PTZConfigChannel	
Success Return	ResponseStatus	
<p>Notes:</p> <p>The <ChannelID> in URI shows the channel number. The <ChannelID> is start at 1.</p> <p><WatchMode> indicate start watch after some second.the value of the <WatchMode> is in second.</p> <p>The Value of the <PTZActionID> is 0,1.....,the range of the value is depend on the value of <PTZActionType>, if the <PTZActionType> is "preset", the range is 0-15,if the <PTZActionType> is</p>		

“cruise” or “track”, the value of the <PTZActionID> is must be 0.

The value of the <Protocol> must be the one of the value of the <ProtocolName> int the <PTZProtocolList>.

PTZConfigChannel Config XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<PTZConfigChannel>
  <Enable><!--req,xs:bool--></Enable>
  <Address><!--req,xs:string--></Address>
  <Bitrate><!--req,xs:string--></Bitrate>
  <Protocol><!--req,xs:string--></Protocol>
  <PTZWatch>
    <Enable><!--req,xs:boolean--></Enable>
    <WatchMode><!--req,xs:integer,in second--></WatchMode>
    <PTZActionType><!--req,xs:string,"preset,cruise,track"--></PTZActionType>
    <PTZActionID><!--req,xs:integer--></PTZActionID>
  </PTZWatch>
  <PTZProtocolList>
    <ProtocolCount><!--req,xs:string--></ProtocolCount>
    <PTZProtocol>
      <ProtocolName><!--req,xs:string--></ProtocolName>
    </PTZProtocol>
  </PTZProtocolList>
</PTZConfigChannel>
```

9.8.1 /PTZ/ChannelID/AdvanceConfig

/PTZ/ChannelID/AdvanceConfig		General Resource v1.0
GET		Viewer
Description	Get advanced configuration information for the PTZ	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or PTZAdvance	
PUT		Viewer
Description	Set advanced configuration information for the PTZ	
Query	None	
Inbound Data	PTZAdvance	
Success Return	ResponseStatus	
注意:		
<ChannelID> Starting from 1.		

PTZAdvance Config XML

```

<PTZAdvance>
  <CruiseEnable><!--xs:boolean,"true,false"--></CruiseEnable>
  <BeginPresetNumForCruise><!--xs:interger--></BeginPresetNumForCruise>
  <BeginPresetActionForCruise><!--xs:string--></BeginPresetActionForCruise>
  <EndPresetNumForCruise><!--xs:interger--></EndPresetNumForCruise>
  <EndPresetActionForCruise><!--xs:string--></EndPresetActionForCruise>
  <WatchEnable><!--xs: boolean,"true,false"--></WatchEnable>
  <BeginPresetNumForWatch><!--xs:interger--></BeginPresetNumForWatch>
  <BeginPresetActionForWatch><!--xs:string--></BeginPresetActionForWatch>
  <EndPresetNumForWatch><!--xs:interger--></EndPresetNumForWatch>
  <EndPresetActionForWatch><!--xs:string--></EndPresetActionForWatch>
</PTZAdvance>

```

9.8.2 /PTZ/ChannelID/Watch/StartWatch

/PTZ/ChannelID/Watch/StartWatch		General Resource v1.0
PUT		Viewer
Description	Start Watch	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Note: <ChannelID> Starting from 1. Param1 is Cruise.Range is 0-15. Sample: URL: http://192.168.1.46/PTZ/1/Watch/StartWatch Data: PUT /PTZ/1/Watch/StartWatch HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1		

9.8.3 /PTZ/ChannelID/Watch/StopWatch

/PTZ/ChannelID/Watch/StopWatch		General Resource v1.0
PUT		Viewer
Description	Stop Watch	

Query	None
Inbound Data	None
Success Return	ResponseStatus
Note: <ChannelID> Starting from 1. Param1 is Cruise.Range is 0-15. Sample: URL: http://192.168.1.46/PTZ/1/Watch/StopWatch Data: PUT /PTZ/1/Watch/StopWatch HTTP/1.1\r\n Connection:close\r\n Host: 192.168.0.46\r\n Content-Length:\r\n \r\n Param1=1	

9.9 /TwoWayAudio

/TwoWayAudio	General Resource v1.0
Notes:	

9.9.1 /TwoWayAudio/Open

/TwoWayAudio/Open	General Resource v1.0
PUT	Viewer
Description	It is used to open the two way audio.
Query	None
Inbound Data	TwoWayAudio
Success Return	ResponseStatus
Notes:	

```
<TwoWayAudio>
  <VoiceMode><!--req,xs:string,"G726,G711U,G711A" --></VoiceMode>
</TwoWayAudio>
```

9.9.2 /TwoWayAudio/Close

/TwoWayAudio/Open	General Resource v1.0
PUT	Viewer

Description	It is used to close the two way audio.
Query	None
Inbound Data	None
Success Return	ResponseStatus
Notes:	

9.9.3 /TwowayAudio/SendData

/TwowayAudio/SendData		General Resource	v1.0
PUT		Viewer	
Description	It is used to send audio data.		
Query	None		
Inbound Data	Audio Data		
Success Return	ResponseStatus		
Notes:			

Example:

```

PUT /TwowayAudio/sendData HTTP/1.1
...
Content-Type: audio/basic
Content-Length: xxx
\r\n
TwowayAudio Data

```

9.9.4 /TwowayAudio/ReceiveData

/TwowayAudio/ReceiveData		General Resource	v1.0
GET		Viewer	
Description	It is used to receive audio data.		
Query	None		
Inbound Data	None		
Success Return	Audio Data or ResponseStatus		
Notes:			

Example:

```

GET /TwowayAudio/receiveData HTTP/1.1
...

```

```

HTTP/1.1 200 OK
...
Content-Type: audio/basic
Content-Length: xxx
\r\n
TwowayAudio Data...

```

9.10 /EventsNotification

/EventsNotification	General Resource v1.0
Notes:	

9.10.1 /EventsNotification/SubscribedEventTypeList

/EventsNotification/SubscribedEventTypeList		General Resource v1.0
GET		Viewer
Description	It is used to get a particular channel subscribed event type list.	
Query	None	
Inbound Data	None	
Success Return	SubscribedEventTypeList or ResponseStatus	
PUT		Operator
Description	It is used to subscribe a event type.	
Query	None	
Inbound Data	SubscribedEventTypeList	
Success Return	ResponseStatus	
DELETE		Operator
Description	It is used to delete subscribed event type.	
Query	None	
Inbound Data	SubscribedEventTypeList	
Success Return	ResponseStatus	
Notes:		
You only use "PUT" method to subscribe a event type and you only set the value of the <EventType>		

EventList XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<SubscribedEventTypeList>
  <SubscribedEventType>
    <ID><!--req,xs:integer--></ID>

```

```

    <EventType><!--req,xs:string, "VMD,videolost,AlarmIn" --></EventType>
    <EventTypeDescription><!--req,xs:string--></EventTypeDescription>
  </SubscribedEventType>
</SubscribedEventTypeList>

```

9.10.2 /EventsNotification/Notify

/EventsNotification/Notify(not support)		General Resource v1.0
GET		Viewer
Description	It is used to get the event notification data stream through HTTP server push.	
Query	None	
Inbound Data	None	
Success Return	EventNotify	
Notes:		
<p>This function is used to get an event notification alert stream from the media device via HTTP or HTTPS. This function does not require that a client/VMS system be added as an HTTP(S) destination on the media device. Instead, the client/VMS system can call this API to initialize a stream of event information from the device. In other words, a connection is established with the device when this function is called, and stays open to constantly receive event notifications.</p> <p>This API uses HTTP server-push with the MIME type multipart/mixed defined in RFC 2046.</p> <p><protocol> is the protocol name, i.e. "HTTP" or "HTTPS".</p> <p><channelID> is present for video and analytics events.</p> <p>This URL is not support.</p>		

EventList XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<EventNotify>
  <IPAddress><!--req,xs:string--></IPAddress>
  <Port><!--req,xs:integer--></Port>
  <Protocol><!--req,xs:string--></Protocol>
  <ChannelID><!--req,xs:string--></ChannelID>
  <DateTime><!--req,xs:datetime--></DateTime>
  <EventType><!--req,xs:string--></EventType>
  <EventDescription><!--req,xs:string--></EventDescription>
</EventNotify>

```

9.10.3 /EventsNotification/EventList

/EventsNotification/EventList		General Resource v1.0
GET		Viewer
Description	It is used to get the event notification data stream through HTTP server push.	
Query	None	
Inbound Data	None	
Success Return	EventList	
Notes:		
<p>Firstly, You must be subscribe a event(VMD, VideoLost, AlarmIn) you want by the URL of /EventsNotification/SubscribedEventTypeList, then, you can get the current event list by the url /EventsNotification/EventList.</p>		

EventList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<EventList>
  <Event>
    <ID><!--req,xs:string --></ID>
    <IPAddress><!--req,xs:string--></IPAddress>
    <Port><!--req,xs:integer--></Port>
    <Protocol><!--req,xs:string--></Protocol>
    <ChannelID><!--req,xs:string--></ChannelID>
    <DateTime><!--req,xs:datetime--></DateTime>
    <EventType><!--req,xs:string--></EventType>
    <EventDescription><!--req,xs:string--></EventDescription>
  </Event>
</EventList>
```

9.11 /Alarm

/Alarm	General Resource v1.0
---------------	-----------------------

9.11.1 /Alarm/AlarmIn

/Alarm/AlarmIn		General Resource v1.0
GET		Viewer
Description	It is used to get all alarm-in configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or AlarmInList	
PUT		Operator

Description	It is used to set all alarm-in configuration.
Query	None
Inbound Data	AlarmInList
Success Return	ResponseStatus
Notes:	

AlarmIn XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<AlarmInList>
  <AlarmIn><!-- opt -->
</AlarmInList>
```

9.11.1.1 /Alarm/AlarmIn/ChannelID

/Alarm/AlarmIn/ChannelID		General Resource v1.0
GET		Viewer
Description	It is used to get alarm-in configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or AlarmIn	
PUT		Operator
Description	It is used to set alarm-in configuration.	
Query	None	
Inbound Data	AlarmIn	
Success Return	ResponseStatus	
Notes:		
<p>The format of <AlarmOutMask>,<SnapshotMask>,<RecordMask> is similar to "11111111" or "00000000", each bit represents whether a channel is linkage or not, 1 means linkage,0 means not linkage, for example, the string "11111111" represents that the device is 8 channels, from left to right, represent channel 1,channel 2,channel 3..... and so on.</p> <p>For single-channel device, <AlarmOutMask>, <SnapshotMask>, <RecordMask> and<PTZActionList> are not effective .</p>		

AlarmIn XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<AlarmIn>
  <ID><!--req,xs:string --></ID>
  <Name><!--req,xs:string,max length little than 128--></Name>
  <AlarmInType><!--req,xs:string,"open,close"--></AlarmInType>
  <Schedule>
    <AllDay><!--req,xs:Boolean--></AllDay>
    <TimeBlock>
```

```
<Block_0><!req,xs:string></Block_0>
<Block_1><!req,xs:string></Block_1>
<Block_2><!req,xs:string></Block_2>
<Block_3><!req,xs:string></Block_3>
<Block_4><!req,xs:string></Block_4>
<Block_5><!req,xs:string></Block_5>
<Block_6><!req,xs:string></Block_6>
  <TimeBlock>
</Schedule>
<Trigger>
  <NotifyAMS>
    <Enable><!--req,xs:boolean--></Enable>
  </NotifyAMS>
  <BeepAlert>
    <Enable><!--req,xs:boolean--></Enable>
  </BeepAlert>
  <FullScreen>
    <Enable><!--req,xs:boolean--></Enable>
  </FullScreen>
  <Mail>
    <Enable><!--req,xs:boolean--></Enable>
  </Mail>
  <AlarmOut>
    <Enable><!--req,xs:boolean--></Enable>
    <AlarmOutMask><!--req,xs:string," 11111111111111"--></AlarmOutMask>
  </AlarmOut>
  <Snapshot>
    <Enable><!--req,xs:boolean--></Enable>
    <SnapshotMask><!--req,xs:string," 11111111111111"--></SnapshotMask>
  </Snapshot>
  <Record>
    <Enable><!--req,xs:boolean--></Enable>
    <RecordMask><!--req,xs:string," 11111111111111"--></RecordMask>
  </Record>
  <PTZ>
    <Enable><!--req,xs:boolean--></Enable>
    <ptzChannelID> <!--req, xs:string --></ptzChannelID>
    <ActionName><!-- req, xs:string,"none, preset, pattern, patrol" --></ActionName>
    <ActionNum><!--req,xs:integer,depend on <ActionName>--></ActionNum>
  </PTZ>
</Trigger>
</AlarmIn>
```

9.11.2 /Alarm/AlarmOut

/Alarm/AlarmOut		General Resource v1.0
GET		Viewer
Description	It is used to get all alarm-out configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or AlarmOutList	
PUT		Operator
Description	It is used to set all alarm-out configuration.	
Query	None	
Inbound Data	AlarmOutList	
Success Return	ResponseStatus	
Notes:		

AlarmOutList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<AlarmOutList>
  <AlarmOut><!-- opt -->
</AlarmOutList>
```

9.11.2.1 /Alarm/AlarmOut/ChannelID

/Alarm/AlarmOut/ChannelID		General Resource v1.0
GET		Viewer
Description	It is used to get a alarm-out configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or AlarmOut	
PUT		Operator
Description	It is used to set a alarm-out configuration.	
Query	None	
Inbound Data	AlarmOut	
Success Return	ResponseStatus	
Notes:		
If The value of the <Delay> is "keep", the alarm keep output.if The value of the <Delay> is number such as 5,the alarm will be keep 5 second output.		

AlarmOut XML Block

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

```

<ID><!--req,xs:string --></ID>
<Name><!--req,xs:string,max length little than 128--></Name>
<Delay><!--req,xs:string,"keep,custom(the unit is second)"--></Delay>
</AlarmOut>

```

9.11.3 /Alarm/AMS

/Alarm/AMS		General Resource v1.0
GET		Viewer
Description	It is used to get ams configuration	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or AMS	
PUT		Operator
Description	It is used to set ams configuration	
Query	None	
Inbound Data	AMS	
Success Return	ResponseStatus	
Notes:		

AMS XML Block

```

<?xml version="1.0" encoding="utf-8"?>
<AMS>
  <Enable><!--rw,req,xs:boolean --></Enable>
  <ServerName><!--rw,req,xs:string --></ServerName>
  <UserName><!--rw,req,xs:string --></UserName>
  <Password><!--rw,req,xs:string --></Password>
  <ServerURL><!--rw,req,xs:string --></ServerURL>
</AMS>

```

9.11.4 /Alarm/EventCfgList

/Alarm/EventCfgList		General Resource v1.0
GET		Viewer
Description	It is used to get event configuration list	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or EventCfgList	
PUT		Operator
Description	It is used to set event configuration list	
Query	None	

Inbound Data	EventCfgList
Success Return	ResponseStatus
Notes:	

EventCfgList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<EventCfgList>
  <motion>
    <PreRecordTime><!--rw,req,xs:interger--></PreRecordTime>
    <AlarmInterval><!--rw,req,xs:interger--></AlarmInterval>
    <Message><!--rw,req,xs:string--></Message>
  </motion>
  <http>
    <PreRecordTime><!--rw,req,xs:interger--></PreRecordTime>
    <AlarmInterval><!--rw,req,xs:interger--></AlarmInterval>
    <Message><!--rw,req,xs:string--></Message>
  </http>
</EventCfgList>
```

9.12 /Record

/Record	General Resource v1.0
GET	Viewer
Description	It is used to get record configuration.
Query	None
Inbound Data	None
Success Return	ResponseStatus or Record
PUT	Operator
Description	It is used to set record configuration.
Query	None
Inbound Data	Record
Success Return	ResponseStatus
Notes:	

Record XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Record>
  <RecordWriteCircleType><!--rw,req,xs:string,"fullnotwrite,writecircle"--></RecordWriteCircleType>
  <RecordDeleteType><!--rw,req,xs:string,"24,48,none"--></RecordDeleteType>
  <RecordStreamType><!--rw,req,xs:string,"main,sub,hdsb"--></RecordStreamType>
  <PreRecord><!--rw,opt,xs:integer,1-120,the unit is second--></PreRecord>
  <DelayRecord><!--rw,opt,xs:integer,1-120,the unit is second--></DelayRecord>
```

```
<RecordMode><!--rw,opt,xs:string,"stop,manual,auto"--></RecordMode>
</Record>
```

9.12.1 /Record/HttpEvent/Config

/Record/HttpEvent/Config		General Resource v1.0
GET		Viewer
Description	It is used to get http event configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or HttpEventConfig	
PUT		Operator
Description	It is used to set http event configuration.	
Query	None	
Inbound Data	HttpEventConfig	
Success Return	ResponseStatus	
Notes:		

AlarmOut XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<HttpEventConfig>
  <Enable><!--req,xs:boolean--></Enable>
</HttpEventConfig>
```

9.12.2 /Record/HttpEvent/Call

/Record/HttpEvent/Call		General Resource v1.0
PUT		Viewer
Description	It is used to Call a record event.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus	
Notes:		

9.12.3 /Record/Format/Call

/Record/Format/Call		General Resource v1.0
PUT		Viewer

Description	It is used to Format the storage device.
Query	None
Inbound Data	None
Success Return	ResponseStatus
Notes:	

9.12.4 /Record/Notify/URL

/Record/Notify/URL		General Resource	v1.0
GET		Viewer	
Description	It is used to get the URL of the upload server.		
Query	None		
Inbound Data	None		
Success Return	URL or ResponseStatus		
PUT		Viewer	
Description	It is used to Set the URL of the upload server.		
Query	None		
Inbound Data	URL		
Success Return	ResponseStatus		
Notes:			

URL XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<URL>
  <Description><!--req,xs: string --></Description>
  <Protocol><!--req,xs:string,such as http,ftp,https --></Protocol>
  <Host><!--req,xs: string --></Host>
  <Port><!--req,xs:interger--></Port>
  <Path><!--req,xs:string--></Path>
  <Username><!--req,xs:string--></Username>
  <Password><!--req,xs:string--></Password>
</URL>
```

9.12.5 /Record/ChannelID/RecordSchedule

/Record/Notify/URL		General Resource	v1.0
GET		Viewer	
Description	It is used to get the record schedule		
Query	None		
Inbound Data	None		
Success Return	RecordSchedule or ResponseStatus		

9.13 /Disk

/Disk		General Resource v1.0
GET		Viewer
Description	It is used to get the disk configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or DiskList	
PUT		Operator
Description	It is used to set the disk configuration.	
Query	None	
Inbound Data	DiskList	
Success Return	ResponseStatus	
Notes:		
The <PictureCapacityPercent> and <RecordCapacityPercent> is the actual storage size, not the percentage.		

DiskList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<DiskList>
  <Disk>
    <DiskID><!--ro,req,xs:integer,"0,1,2..."--></DiskID>
    <TotalCapacity><!--ro,req,xs:integer,the unit is MB--></TotalCapacity>
    <AvailableCapacity><!--ro,req,xs:integer,the unit is MB--></AvailableCapacity>
    <DiskStorageType><!--ro,req,xs:string,"SD,samba"--></DiskStorageType>
    <DiskStorageAttribute><!--ro,req,xs:string,"idle,writing,reading,diskexception,diskerror,readonly,readwrite,nodisk,exceptionneedformat,exceptionneedChangeDisk,formatfailedneedreboot"--></DiskStorageAttribute>
    <PictureCapacityPercent><!--rw,req,xs:integer--></PictureCapacityPercent>
    <RecordCapacityPercent><!--rw,req,xs:integer--></RecordCapacityPercent>
    <DiskFormatStatus><!--ro,req,xs:string,"unformatted,formatting,formatted,format failure,umount failure,device is busy"--></DiskFormatStatus>
  </Disk>
</DiskList>
```

9.14 /Playback

9.14.1 /Playback/GetRecordInfoByMonth

/Playback/GetRecordInfoByMonth		General Resource v1.0
GET		Viewer
Description	It is used to get the RecordInfo by Month.	

Query	None
Inbound Data	None
Success Return	ResponseStatus or RecordInfoByMonth
<p>Notes:</p> <p>From left to right,if the bit of the <RecordInfo> is "0",then the day has no record,otherwise ,the day has record.</p> <p>Example:</p> <p>http://192.168.0.180/Playback/GetRecordInfoByMonth?Year=2017&Month=2</p>	

RecordInfoByMonth XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<RecordInfoByMonth>
  <RecordInfo><!--ro,req,xs:string,"00000000000000001111111111111111"--></RecordInfo>
</RecordInfoByMonth>
```

9.14.2 /Playback/GetRecordInfoByDay

/Playback/GetRecordInfoByDay		General Resource v1.0
GET	Viewer	
Description	It is used to get the disk configuration.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or RecordInfoByDay	
<p>Notes:</p> <p><Channel> is start at 0,if the value is 0xff,It is present to search all channel.</p> <p><RecordType> is invalid,defaultly search all record information.</p> <p>Example:</p> <p>http://192.168.0.180/Playback/GetRecordInfoByDay?Year=2017&Month=2&Day=10&Channel=0&RecordType=0</p>		

RecordInfoByDay XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<RecordInfoByDay>
  <RecordInfoList>
    <RecordInfo>
      <Channel><!--ro,req,xs:interger--></Channel>
      <RecordType><!--ro,req,xs:string,"general,manual,alarm,motion"--></RecordType>
      <StartTime><!--req,xs:datetime," 20040503173008"--></StartTime>
      <EndTime><!--req,xs:datetime," 20040503173008"--></EndTime>
    </RecordInfo>
  </RecordInfoList>
</RecordInfoByDay>
```

9.15 RTMP

9.15.1 /RTMP/BasicConfig

/RTMP/BasicConfig		General Resource v1.0
GET		Viewer
Description	It is used to get basic RTMP settings.	
Query	None	
Inbound Data	None	
Success Return		
SET		Viewer
Description	It is used to get the .	
Query	None	
Inbound Data	None	
Success Return		
<p>the <IsSupportRetrievePushAddressFromServer> of value is "true" means device support retrieve pushing stream URL from a specific server. when "false", the pushing stream URL must be specific by manually.</p> <p><Mode> "dynamic" or "manual".</p> <p><PushURLGeneratorServer> is Server URL, this server must be ruled by our protocol. if you want to build a server, you should contact us.</p>		

RecordInfoByMonth XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<BasicConfig>
  <IsSupportRetrievePushAddressFromServer><!--req,xs:boolean--></IsSupportRetrievePushAddressFromServer>
  <Mode><!--ro,req,xs:string--></Mode>
  <PushURLGeneratorServer><!--ro,req,xs:string--></PushURLGeneratorServer>
</BasicConfig>
```

9.15.2 /RTMP/ChannelID/StreamID/Config

/RTMP/StreamID/Config		General Resource v1.0
GET		Viewer
Description	It is used to get the .	
Query	None	
Inbound Data	None	
Success Return		

SET		Viewer
Description	It is used to get the .	
Query	None	
Inbound Data	None	
Success Return		
Notes: <p>The <ChannelID> in the URI shows the channel number, The <ChannelID> is start at 1.</p> <p>The <StreamID> represents stream type, generally "1" represents the main stream, "2" means "the first sub-Stream", "3" means "the second sub-stream", and so on.</p> <p>The <Enable> field and <PushURL> are valid when <Mode> of BasicConfig equals "manual". "true" enable device to push stream to RTMP sever, "false" disable RTMP service.</p> <p>The <PushURL> field is RTMP URL for pushing stream by RTMP protocol. when device working at manual mode, this field must be fill by user.</p> <p>The <PlayURL> field is valid when device working at dynamic mode. User cloud open this URL by web browser to watch stream(ChannelID/StreamID).</p>		

RecordInfoByMonth XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<Config>
  <Enable><!--rw,req,xs:boolean--></Enable>
  <PushURL><!--rw,req,xs:string--></PushURL>
  <PushURL_Base64><!--rw,req,xs:string--></PushURL_Base64>
  <Status><!--ro,req,xs:string--></Status>
  <PlayURL><!--rw,req,xs:string--></PlayURL>
  <PlayURL_Base64><!--rw,req,xs:string--></PlayURL_Base64>
  <AudioEnable><!--rw,req,xs:boolean--></AudioEnable>
</Config>
```

9.16 /RecordFileList

9.16.1 /RecordFileList/Attribute

/RecordFileList/Attribute		General Resource v1.0
GET		Viewer
Description	It is used to get the Record file list attribute.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or RecordFileListAttribute	
PUT		Operator
Description	It is used to set the record file list attribute.	
Query	None	

Inbound Data	RecordFileListAttribute
Success Return	ResponseStatus
Notes: <begin_time>,<end_time>,<timezone_min_between_utc> is valid when the method of the http protocol is "PUT".	

RecordFileListAttribute XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<RecordFileListAttribute>
  <RecordFileSumCount><!--ro,req,xs:integer--></RecordFileSumCount>
  <RecordFileCountPerPage><!--ro,req,xs:integer--></RecordFileCountPerPage>
  <begin_time><!--ro,opt,xs:string,"20180507 020445"--></begin_time>
  <end_time><!--ro,opt,xs:string" 20180507 020445"--></end_time>
  <timezone_min_between_utc><!--ro,opt,xs:interger,the unit is minute -->
  </timezone_min_between_utc>
</RecordFileListAttribute>
```

9.16.2 /RecordFileList/Pages/PageNum

/RecordFileList/Pages/PageNum		General Resource v1.0
GET		Viewer
Description	It is used to get the Record list of the special page number.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or RecordFileList	
Notes: Example: <a href="http://192.168.0.180/RecordFileList/Pages/1?Pathtype=0 1(0,the <PathName> relative to the http root directory.1,the <PathName> is a absolute path.)">http://192.168.0.180/RecordFileList/Pages/1?Pathtype=0 1(0,the <PathName> relative to the http root directory.1,the <PathName> is a absolute path.)		

RecordFileList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<RecordFileList>
  <RecordFile>
    <PathName><!--ro,req,xs:string--></PathName>
    <FileName><!--ro,req,xs:string--></FileName>
    <StartTime><!--req,xs:datetime" 20040503 173008"--></StartTime>
    <StopTime><!--req,xs:datetime" 20040503 173008"--></StopTime>
    <FileSize><!--req,xs:interger,the unit is byte--></FileSize>
  </RecordFile>
</RecordFileList>
```

9.17/SnapshotFileList

9.17.1 /SnapshotFileList/Attribute

/SnapshotFileList/Attribute		General Resource v1.0
GET		Viewer
Description	It is used to get the snapshot file list attribute.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or SnapshotFileListAttribute	
PUT		Operator
Description	It is used to set the snapshot file list attribute.	
Query	None	
Inbound Data	SnapshotFileListAttribute	
Success Return	ResponseStatus	
Notes:		
<p><begin_time>,<end_time>,<timezone_min_between_utc> is valid when the method of the http protocol is "PUT".</p>		

SnapshotFileListAttribute XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<SnapshotFileListAttribute>
  <SnapshotFileSumCount><!--ro,req,xs:integer--></SnapshotFileSumCount>
  <SnapshotFileCountPerPage><!--ro,req,xs:integer--></SnapshotFileCountPerPage>
  <begin_time><!--ro,opt,xs:string,"20180507 020445"--></begin_time>
  <end_time><!--ro,opt,xs:string" 20180507 020445" --></end_time>
  <timezone_min_between_utc><!--ro,opt,xs:interger,the unit is minute -->
  </timezone_min_between_utc>
</SnapshotFileListAttribute>
```

9.17.2 /SnapshotFileList/Pages/PageNum

/SnapshotFileList/Pages/PageNum		General Resource v1.0
GET		Viewer
Description	It is used to get the snapshot list of the special page number.	
Query	None	
Inbound Data	None	
Success Return	ResponseStatus or SnapshotFileList	

Notes:**Example:**

[http://192.168.0.180/SnapshotFileList/Pages/1?PathType=0|1\(0,the <PathName> relative to the http root directory.1,the <PathName> is a absolute path.\)](http://192.168.0.180/SnapshotFileList/Pages/1?PathType=0|1(0,the <PathName> relative to the http root directory.1,the <PathName> is a absolute path.))

SnapshotFileList XML Block

```
<?xml version="1.0" encoding="utf-8"?>
<SnapshotFileList>
  <SnapshotFile>
    <PathName><!--ro,req,xs:string--></PathName>
    <FileName><!--ro,req,xs:string--></FileName>
    <StartTime><!--req,xs:datetime" 20040503 173008"--></StartTime>
    <StopTime><!--req,xs:datetime" 20040503 173008"--></StopTime>
    <FileSize><!--req,xs:interger,the unit is byte--></FileSize>
  </SnapshotFile>
</SnapshotFileList>
```

10 Device discovery V2

The IP Device in addition to support the upnp protocol to discovery device ,but this method can not set the network configuration overlay IP, so we support other method to discovery device and set the network configuration .this method use Multicast to transfer device information,the Multicast Address is 239.239.239.52,the port is 5566,the client only transfer a request package to the network on the 5566 port in the multicast method, we can get the device information but please note that this method is not over http,of course ,we also can use this method to set the network configuration,

The following is the description for use.

//Detect Request

Client:

```
<?xml version="1.0" encoding="UTF-8">
<Request>
  <Head Version="1.0.0" ConfigLength="xxx" CommandID="10" />
  <CfgInfo MainCommand="100" AssistCommand="-1" DeviceMac="xx" />
</Request>
```

Server:

```
<?xml version="1.0" encoding="UTF-8">
<Response>
  <Head Version="1.0.0" ConfigLength="xxx" CommandID="11" ReplyMsg="xx">
  <CfgInfo MainCommand="100" AssistCommand="-1" DeviceMac="xx" />
  <DeviceProbeConfig>
```

```

    <DeviceType><!--ro,req,xs:string--></DeviceType>
    <DeviceVersion><!--ro,req,xs:string--></DeviceVersion>
    <DeviceChannelCount><!--ro,req,xs:integer--></DeviceChannelCount>
    <DeviceEnableRegister><!--req,xs:boolean--></DeviceEnableRegister>
    <DeviceRegisterInterval>
        <!--req,xs:integer,"second"-->
    </DeviceRegisterInterval>
    <DeviceMac><!--ro,req,xs:string--></DeviceMac>
    <DeviceIP><!--req,xs:string--></DeviceIP>
    <DeviceMask><!--req,xs:string--><DeviceMask>
    <DeviceGateway><!--req,xs:string--></DeviceGateway>
    <DeviceManagerHost><!--req,xs:string--></DeviceManagerHost>
    <DeviceDetectPort><!--req,xs:integer--></DeviceDetectPort>
    <ManagerHostPort><!--req,xs:integer--></ManagerHostPort>
    <CommandPort><!--req,xs:integer--></CommandPort>
    <MediaPort><!--req,xs:integer--></MediaPort>
    <HttpPort><!--req,xs:integer--></HttpPort>
    <RtspPort><!--req,xs:integer--></RtspPort>
    <TutkUID><!--ro,req,xs:integer--><TutkUID>
    <UlkUID><!--ro,req,xs:integer--><UlkUID>
</DeviceProbeConfig>
</Response>

```

//Detect Set Device Configuration

Client:

```

<?xml version="1.0" encoding="UTF-8">
<Request>
    <Head Version="1.0.0" ConfigLength="xxx" CommandID="10">
    <CfgInfo MainCommand="101" AssistCommand="-1" DeviceMac="xx" />
    <DeviceProbeConfig>
        <DeviceType><!--ro,req,xs:string--></DeviceType>
        <DeviceVersion><!--ro,req,xs:string--></DeviceVersion>
        <DeviceChannelCount><!--ro,req,xs:integer--></DeviceChannelCount>
        <DeviceEnableRegister><!--req,xs:boolean--></DeviceEnableRegister>
        <DeviceRegisterInterval>
            <!--req,xs:integer,"second"-->
        </DeviceEnableRegInterval>
        <DeviceMac><!--ro,req,xs:string--></DeviceMac>
        <DeviceIP><!--req,xs:string--></DeviceIP>
        <DeviceNetmask><!--req,xs:string--><DeviceMask>
        <DeviceGateway><!--req,xs:string--></DeviceGateway>
        <DeviceManagerHost><!--req,xs:string--></DeviceManagerHost>
        <DetectPort><!--req,xs:integer--></DeviceDetectPort>
        <ManagerHostPort><!--req,xs:integer--></ManagerHostPort>
    </DeviceProbeConfig>
    </Request>

```

```
<CommandPort><!--req,xs:integer--></CommandPort>
<MediaPort><!--req,xs:integer--></MediaPort>
<HttpPort><!--req,xs:integer--></HttpPort>
<RtspPort><!--req,xs:integer--></RtspPort>
<TutkUID><!--ro,req,xs:integer--><TutkUID>
<UlkUID><!--ro,req,xs:integer--><UlkUID>
</DeviceProbeConfig>
</Request >
```

Server:

```
<?xml version="1.0" encoding="UTF-8">
<Response>
  <Head Version="x.x.x" ConfigLength="xxx" CommandID="11" ReplyMsg="xx" />
  <CfgInfo MainCommand="101" AssistCommand="-1" DeviceMac="xx" />
</Response>
```

//Detect Network Configuration , need **authentication**

Client:

```
<?xml version="1.0" encoding="UTF-8">
<Request>
  <Head Version="1.0.0" ConfigLength="xxx" CommandID="10" />
  <CfgInfo MainCommand="102" AssistCommand="-1" DeviceMac="xx" />
  <Auth UserName="xx" UserPassword="xx" />
</Request>
```

Server:

```
<?xml version="1.0" encoding="UTF-8">
<Response>
  <Head Version="1.0.0" ConfigLength="xxx" CommandID="11" ReplyMsg="xx">
  <CfgInfo MainCommand="102" AssistCommand="-1" DeviceMac="xx" />
  <DeviceProbeNetConfig>
    <DeviceType><!--ro,req,xs:string--></DeviceType>
    <DeviceVersion><!--ro,req,xs:string--></DeviceVersion>
    <DeviceChannelCount><!--ro,req,xs:integer--></DeviceChannelCount>
    <DeviceEnableRegister><!--req,xs:boolean--></DeviceEnableRegister>
    <DeviceEnableRegInterval>
      <!--req,xs:integer,second in unit-->
    </DeviceEnableRegInterval>
    <NetworkInterfaceList>
      <NetworkInterface>
        <ID><!--req,xs:string--></ID>
        <Enable><!--ro,req,xs:string,"open,close,notsupport"--></Enable>
```

```

        <DhcpEnable><!--req,xs:boolean--></DhcpEnable>
        <MAC><!--req,xs:string--></MAC>
        <IPAddress><!--req,xs:string--></IPAddress>
        <Netmask><!--req,xs:string--></Netmask>
        <Gateway><!--req,xs:string--></Gateway>
        <EnableAutoGenerateIP>
            <!--req,xs:boolean-->
        </EnableAutoGenerateIP>
        <NetworkState>
            <!--ro,opt,"get ip success,close,getting ip,get ip failed"-->
        </NetworkState>
    </NetworkInterface>
</NetworkInterfaceList>
<DeviceManagerHost><!--req,xs:string--></DeviceManagerHost>
<DeviceDetectPort><!--req,xs:integer--></DeviceDetectPort>
<ManagerHostPort><!--req,xs:integer--></ManagerHostPort>
<CommandPort><!--req,xs:integer--></CommandPort>
<MediaPort><!--req,xs:integer--></MediaPort>
<HttpPort><!--req,xs:integer--></HttpPort>
<RtspPort><!--req,xs:integer--></RtspPort>
<TutkUID><!--ro,req,xs:integer--><TutkUID>
<UlkUID><!--ro,req,xs:integer--><UlkUID>
</DeviceProbeNetConfig>
</Response>

```

//Detect Set Network Configuration ,**need authentication**

Client:

```

<?xml version="1.0" encoding="UTF-8">
<Request>
    <Head Version="1.0.0" ConfigLength="xxx" CommandID="10">
    <CfgInfo MainCommand="103" AssistCommand="-1" DeviceMac="xx" />
    <Auth UserName="xx" UserPassword="xx" />
    <DeviceProbeNetConfig>
        <DeviceType><!--ro,req,xs:string--></DeviceType>
        <DeviceVersion><!--ro,req,xs:string--></DeviceVersion>
        <DeviceChannelCount><!--ro,req,xs:integer--></DeviceChannelCount>
        <DeviceEnableRegister><!--req,xs:boolean--></DeviceEnableRegister>
        <DeviceEnableRegInterval>
            <!--req,xs:integer,second in unit-->
        </DeviceEnableRegInterval>
        <DeviceMac><!--ro,req,xs:string--></DeviceMac>
        <NetworkInterfaceList>
            <NetworkInterface>
                <ID><!--req,xs:string--></ID>

```

```

    <Enable><!--req,xs:string,"open,close,notsupport"--></Enable>
    <DhcpEnable><!--req,xs:boolean--></DhcpEnable>
    <MAC><!--req,xs:string--></MAC>
    <IPAddress><!--req,xs:string--></IPAddress>
    <Netmask><!--req,xs:string--></Netmask>
    <Gateway><!--req,xs:string--></Gateway>
    <EnableAutoGenerateIP>
        <!--req,xs:boolean-->
    </EnableAutoGenerateIP>
    <NetworkState>
        <!--opt,"get ip success,close,getting ip,get ip failed"-->
    </NetworkState>
</NetworkInterface>
</NetworkInterfaceList>
<DeviceManagerHost><!--req,xs:string--></DeviceManagerHost>
<DeviceDetectPort><!--req,xs:integer--></DeviceDetectPort>
<ManagerHostPort><!--req,xs:integer--></ManagerHostPort>
<CommandPort><!--req,xs:integer--></CommandPort>
<MediaPort><!--req,xs:integer--></MediaPort>
<HttpPort><!--req,xs:integer--></HttpPort>
<RtspPort><!--req,xs:integer--></RtspPort>
<TutkUID><!--ro,req,xs:integer--><TutkUID>
<UlkUID><!--ro,req,xs:integer--><UlkUID>
</DeviceProbeNetConfig>
</Request>

```

Server:

```

<?xml version="1.0" encoding="UTF-8" />
<Response>
    <Head Version="1.0.0" ConfigLength="xxx" CommandID="11" ReplyMsg="xx" />
    <CfgInfo MainCommand="103" AssistCommand="-1" DeviceMac="xx" />
</Response>

```

Note:

<Version> is show this interface version, now, we only support "1.0.0".
 <ConfigLength> is show the struct length that you send.
 <CommandID> is the Command ID,if it is the data that the client send ,the <CommandID> is 10,
 If it is the data that the server send,the <CommandID> is 11.
 <Replymsg> is the error code that the server send, if the <ReplyMsg> is 0,then reply
 success,otherwise, reply failed.
 <MainCommand> and <AssistCommand> is show the instruction that you execute. Different
 request has different instruction.
 <DeviceMac> is the Device mac that you want to set.