Open SDK

Hanwha

SUNAPI

v2.6.2
2023-04-07

Contact Information

Hanwha Vision Co., Ltd.
Hanwha Vision 6, Pangyo-ro 319beon-gil, Bundang-gu, Seongnam-si, Gyeonggi-do, 13488, KOREA
www.hanwhavision.com

Hanwha Vision America
500 Frank W. Burr Blvd. Suite 43 Teaneck, NJ 07666
hanwhavisionamerica.com

Hanwha Vision Europe
Heriot House, Heriot Road, Chertsey, Surrey, KT16 9DT, United Kingdom
hanwhavision.eu

Hanwha Vision Middle East FZE
Jafza View 18, Office 2001-2003, Po Box 263572, Jebel Ali Free Zone, Dubai, United Arab Emirates
www.hanwhavision.com/ar

Table of Contents

Chapter 1. Overview

1.1. Description

opensdk.cgi is used to install and manage the application.

The following submenus are used for open SDK functionalities:

- **apps**: Requests and configures the application general settings. The **apps** submenu is also used to install and remove the application.
- **appstatus**: Requests the application status such as memories and CPU used for one time or periodically.
- **manifest**: Requests the application manifest file.
- **debug**: Requests to debug an opensdk application using 'RemoteDebugViewer'.
- **opensdkeventinfo**: Reads the event schema from the camera's third party application.
- **metaframeschema**: Used to notify the metaframe schema supported by an app.
- **metaframecapability**: Submenu to notify all supported values/ranges of metadata parameters.

NOTE

This document applies to the network cameras only.
Attribute to check for feature support: "attributes/System/Support/OpenSDK"
For multi-directional cameras, please refer to the value "attributes/System/Support/OneOpenAppPerChannel". When this value is set to true, the application can be installed on any one channel.

Chapter 2. Applications

2.1. Description

The **apps** submenu requests, configures and controls the application settings. It is also used to install and remove the application.

> **NOTE** Attribute to check maximum applications: "**attributes/system/Limit/OpenSDK.MaxApps**"

Access level

Action	Camera
view	Admin
set	Admin
control	Admin
install	Admin
remove	Admin

2.2. Syntax

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=apps&action=<value>[&<parameter>=<value>...]
```

2.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the application setting information
	InstalledApps	RES	<int>	The number of currently installed applications
	<AppID>.Status	RES	<enum> UnInstalling,Installed,Installing,StartedNotRunning,Running,Stopped	Status of the application

Action	Parameters	Request/ Response	Type/ Value	Description
	<AppID>.InstalledDate	RES	<string>	Installation date of the application The date is specified in the format of <YYYY-MM-DDThh:mm:ssZ> (UTC time).
	<AppID>.Version	RES	<string>	Application version
	<AppID>.Permission	RES	<csv> Device, PTZ, Network, SDCard, None	Application permission is a Read Only parameter. This information is retrieved from the manifest xml of the application.
	<AppID>.AutoStart	RES	<bool> True, False	Enables or disables the application to automatically start
	<AppID>.Priority	RES	<enum> Low, Medium, High	Priority
	<AppID>.Channel	RES	<csv>	The channel index where the application is installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel
	<AppID>.IsDefault	RES	<bool> True, False	Shows whether this app is the Default app **Note** Default apps will not be uninstalled when a factory reset is performed, and the app's Priority setting will automatically be set to High and frozen

Action	Parameters	Request/ Response	Type/ Value	Description
	ControlForbidden	RES	<csv> StartStop,Priority,AutoStart	• StartStop: This means app will not be controled by "control" action "Mode" parameter Start/Stop. Always auto start. • Priority: This means Priority setting is forbidden. • AutoStart: This means AutoStart setting is forbidden.
set	AppID	REQ	<string>	Application ID **Note** **AppID** must be sent together with the **set** action.
	AutoStart	REQ	<bool> True, False	Enables or disables the application to automatically start
	Priority	REQ	<enum> Low, Medium, High	Priority
	Channel	REQ	<int>	Channel ID where the application is installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel
	IsDefault	RES	<bool> True, False	Whether this app is the default app **Note** Default apps will not be uninstalled when a factory reset is performed, and the app's Priority setting will automatically be set to High and frozen
control	AppID	REQ	<string>	Application ID **Note** **AppID** must be sent together with the **control** action.

Action	Parameters	Request/ Response	Type/ Value	Description
	Mode	REQ	<enum> Start, Stop	Mode
	Channel	REQ	<int>	Channel ID where the application is installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel
install	AppID	REQ	<string>	Application ID **Note** **AppID** must be sent together with the **install** action.
	Permission	RES	<csv>	Permission
	InstallType	RES	<enum> New, Upgrade	Installation type
	IgnoreCookie	REQ	<bool> True, False	Ignore Application Session ID as a cookie
	ApplicationSessionId	REQ, RES	<string>	Application Session ID if cookie is not used
	KeepOldSettings	REQ	<bool> True, False	Keeps the old settings **Note** **AppID** and **KeepOldSettings** must be sent together for the **install** action if **InstallType**parameter is NOT set to New.
	Channel	REQ	<int>	Channel ID where the application will be installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel

Action	Parameters	Request/ Response	Type/ Value	Description
remove	AppID	REQ	<string>	Application ID **Note** **AppID** must be sent together with the **remove** action.
	Channel	REQ	<int>	Channel ID where the application is installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel
update	AppID	REQ	<string>	It can upload any files to openapp. Ex.) AI binary data, application config json file, jpg, ...

2.4. Examples

2.4.1. Getting the currently installed apps

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=apps&action=view
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
InstalledApps=1
ANPR.InstalledDate=2014-01-29T15:00:00Z
ANPR.Verion=1.0
ANPR.Permission=SD,Network
ANPR.Status=Stopped
ANPR.AutoStart=False
ANPR.Priority=High
ANPR.Channel=0
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
    "InstalledApps": 1,
    "Apps": [
        {
            "AppID": "ANPR",
            "Status": "Stopped",
            "InstalledDate": "2014-01-29T15:00:00Z",
            "Version": "1.0",
            "Permission": [
                "Sd",
                "Network"
            ],
            "AutoStart": false,
            "Priority": "High",
            "Channel": "0"
        }
    ]
}
```

2.4.2. Installing a new application with CURL

The OpenSDK application installation is a two-step process. First, the application file needs to be sent to the camera via HTTP POST. The camera sends a session id cookie, installation type and the permissions required by the application. Then, based on the installation type and required permissions, a user can decide whether to install the application or not by sending the install command via HTTP GET.

AppID should be same as the application cap file name (without extension).

NOTE	To get JSON response add the -H "Accept: application/json" header to the request.

Without using cookies:

Step 1: CURL command for sending the application to the camera

REQUEST

```
curl -v --digest -u <userid>:<password> -F UploadedFile=@ServerPushMJPEG.cap
```

```
"http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
&IgnoreCookie=true" -H "Expect:"
```

The above command will produce a request to the device as below:

```
POST /stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
HTTP/1.1
User-Agent: curl/7.26.0
Host: 111.111.11.11
Accept: */*
Content-Length: 119523
Content-Type: multipart/form-data; boundary=----------------------------
fb674236d482
```

TEXT RESPONSE

```
HTTP/1.1 200 OK
Content-Type: text/plain
Content-Length: 32
Date: Thu, 20 Mar 2014 01:32:12 GMT
Server: lighttpd/1.4.31
<Body>
```

```
ApplicationSessionId=ServerPushMJPEG-111.111.11.111
Permission=SD,Network
InstallType=New
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 32
Date: Thu, 20 Mar 2014 01:32:12 GMT
Server: lighttpd/1.4.31
<Body>
```

```
{
```

```
"ApplicationSessionId": " ServerPushMJPEG-111.111.11.111",
"Permission": ["SD","Network"],
"InstallType": "New",
}
```

Step 2:CURL command for installing the application

REQUEST

```
curl -v --digest -u <userid>:<password> "http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
& ApplicationSessionId=ServerPushMJPEG-111.111.11.111
```

The above command will produce a request to the device as below:

```
GET /stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
HTTP/1.1
User-Agent: curl/7.26.0
Host: 111.111.11.11
Accept: */*
TEXT RESPONSE
HTTP/1.1 200 OK
Content-Type: text/plain
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd/1.4.31
<Body>
```

```
OK
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=deleted; expires=Wed, 20-Mar-2013 01:46:03
GMT
Content-Type: application/json
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd/1.4.31
```

```
<Body>
```

```
{
    "Response": "Success"
}
```

Using Cookies:

Step 1: CURL command for sending the application to the camera

REQUEST

```
curl -v --digest -u <userid>:<password> -F UploadedFile=@ServerPushMJPEG.cap
"http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
" -H "Expect:"
```

The above command will produce a request to the device as follows:

```
POST /stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
HTTP/1.1
User-Agent: curl/7.26.0
Host: 111.111.11.11
Accept: */*
Content-Length: 119523
Content-Type: multipart/form-data; boundary=----------------------------
fb674236d482
```

TEXT RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=ServerPushMJPEG-111.111.11.111
Content-Type: text/plain
Content-Length: 32
Date: Thu, 20 Mar 2014 01:32:12 GMT
Server: lighttpd/1.4.31
<Body>
```

```
Permission=SD,Network
```

```
InstallType=New
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=ServerPushMJPEG-111.111.11.111
Content-Type: application/json
Content-Length: 32
Date: Thu, 20 Mar 2014 01:32:12 GMT
Server: lighttpd/1.4.31
<Body>
```

```
{
    "Response": "Success"
}
```

Step 2:CURL command for installing the application

REQUEST

```
curl -v --digest -u <userid>:<password> --cookie
AppInstallSessionID=ServerPushMJPEG-<Device IP> "http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
"
```

The above command will produce a request to the device as below:

```
GET /stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
HTTP/1.1
User-Agent: curl/7.26.0
Host: 111.111.11.11
Accept: */*
Cookie: AppInstallSessionID=ServerPushMJPEG-111.111.11.111
```

TEXT RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=deleted; expires=Wed, 20-Mar-2013 01:46:03
GMT
```

```
Content-Type: text/plain
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd/1.4.31
<Body>
```

```
OK
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=deleted; expires=Wed, 20-Mar-2013 01:46:03
GMT
Content-Type: application/json
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd/1.4.31
<Body>
```

```
{
    "Response": "Success"
}
```

2.4.3. Updating the existing application with CURL

Step 1: CURL command for sending the application to the camera

REQUEST

```
curl -v --digest -u <userid>:<password> -F UploadedFile=@ServerPushMJPEG.cap
"http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
" -H "Expect:"
```

TEXT RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=ServerPushMJPEG-111.111.11.111
Content-Type: text/plain
Content-Length: 32
Date: Thu, 20 Mar 2014 01:32:12 GMT
```

```
Server: lighttpd/1.4.31
<Body>
```

```
Permission=SD,Network
InstallType=Upgrade
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=deleted; expires=Wed, 20-Mar-2013 01:46:03
GMT
Content-Type: application/json
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd
<Body>
```

```
{
    "Response": "Success"
}
```

Step 2: CURL command for updating the application

REQUEST

```
curl -v --digest -u <userid>:<password> --cookie AppInstallSessionID=<value>
"http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerPushMJPEG&Channel=0
&KeepOldSettings=True"
```

TEXT RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=deleted; expires=Wed, 20-Mar-2013 01:46:03
GMT
Content-Type: text/plain
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd
```

```
<Body>
```

```
OK
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Set-Cookie: AppInstallSessionID=deleted; expires=Wed, 20-Mar-2013 01:46:03
GMT
Content-Type: application/json
Content-Length: 2
Date: Thu, 20 Mar 2014 01:46:04 GMT
Server: lighttpd

<Body>
```

```
{
    "Response": "Success"
}
```

2.4.4. Installing license

REQUEST

```
curl -v --digest -u admin:<password> -F LicenseFile=@filename
"http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=install&AppID=ServerTest&Channel=0" -H
"Expect:"
```

2.4.5. Removing the installed application

REQUEST

```
http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=apps&action=remove&AppID=ServerPushMJPEG&Channel=0
```

2.4.6. Starting the application

REQUEST

```
http://<Device IP>/stw-
```

```
cgi/opensdk.cgi?msubmenu=apps&action=control&AppID=ServerPushMJPEG&Mode=Start&Channel=0
```

2.4.7. Setting the application priority and enabling AutoStart

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=apps&action=set&AppID=ServerPushMJPEG&Priority=Medium&AutoStart=True&Channel=0
```

2.4.8. Updating (Uploading) a datafile to openapp

REQUEST

Binary data

```
curl -v --digest -u admin:<password> -F DataFile=@{datafilename} "http://<IP>/stw-cgi/opensdk.cgi?msubmenu=apps&action=update&AppID=SNTest" -H "Expect:"
```

In case of not binary data, we need to put "octet-stream type" in data.

REQUEST

```
curl -v --digest -u admin:<password> -F DataFile=@test.txt;type=application/octet-stream "http://<IP>/stw-cgi/opensdk.cgi?msubmenu=apps&action=update&AppID=test_Upload_File" -H "Expect:"
```

Chapter 3. Application Status

3.1. Description

The **appstatus** submenu requests the status of the application.

Access level

Action	Camera
view	Admin

3.2. Syntax

```
http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=appstatus&action=<value>[&<parameter>=<value>...]
```

3.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the application status
	AppID	REQ	<csv> <AppID>	Application ID If the specific App ID is not sent, the status of all applications is returned.
	Channel	REQ	<int>	Channel ID Used when the application status for a specific channel is needed. If **Channel** is not sent, all channels' application status is returned. **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel
	Check	REQ	<enum> Once, Periodically	Whether to checks the application status only once or periodically. If **Check** is not sent, the status of the application is returned to Once.

Action	Parameters	Request/ Response	Type/ Value	Description
	Periodicity	REQ	<int>	Interval for checking the application status If **Periodicity** is not sent, the default periodicity is applicable. The values must be within the range of 1 to 9 and the unit is a second.
	TotalCPUUsage	RES	<int>	Total CPU used The values must be within the range of 1 to 100.
	TotalMemoryUsage	RES	<int>	Memory totally used The values must be in the range of 1 to 100.
	<AppID>.CPUUsage	RES	<int>	CPU used in the corresponding application The values must be within the range of 1 to 100.
	<AppID>.MemoryUsage	RES	<int>	Memory used in the corresponding application The values must be within the range of 1 to 100.
	<AppID>.ThreadsCount	RES	<int>	Thread count of the corresponding application
	<AppID>.Duration	RES	<string>	Duration of the corresponding application Durations are represented by the format <P[n]Y[n]M[n]DT[n]H[n]M[n]S>, following the ISO 8601 duration format.

Action	Parameters	Request/ Response	Type/ Value	Description
	<AppID>.Channel	RES	<csv>	Channel ID where the application is installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel

3.4. Examples

3.4.1. Checking the application status once

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=appstatus&action=view&AppID=ANPR
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain
<Body>
```

```
TotalCPUUsage=40
TotalMemoryUsage=30
ANPR.CPUUsage=30
ANPR.MemoryUsage=10
ANPR.ThreadsCount=11
ANPR.Duration=P0Y0M0DT1H0M0S
ANPR.Channel=0
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
```

```
    "TotalCPUUsage": 40,
    "TotalMemoryUsage": 30,
    "Apps": [
        {
            "AppID": "ANPR",
            "CPUUsage": 30,
            "MemoryUsage": 10,
            "ThreadsCount": 11,
            "Duration": "P0Y0M0DT1H0M0S",
            "Channel": "0"
        }
    ]
}
```

3.4.2. Monitoring the application status of Channel 0 every 5 seconds

REQUEST

```
http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=appstatus&action=view&Check=Periodically&Periodicit
y=5&Channel=0
```

TEXT RESPONSE

```
<Body>
```

```
--SamsungTechwin
Content-Type: text/plain
TotalCPUUsage=40
TotalMemoryUsage=30
ANPR.CPUUsage=30
ANPR.MemoryUsage=10
ANPR.ThreadsCount=11
ANPR.Duration=P0Y0M0DT1H0M0S
ServerPushMJPEG.CPUUsage=10
ServerPushMJPEG.MemoryUsage=5
ServerPushMJPEG.ThreadsCount=9
ServerPushMJPEG.Duration=P0Y0M0DT2H10M0S

--SamsungTechwin
Content-Type: text/plain
```

```
TotalCPUUsage=40
TotalMemoryUsage=30
ANPR.CPUUsage=20
ANPR.MemoryUsage=10
ANPR.ThreadsCount=11
ANPR.Duration=P0Y0M0DT1H10M0S
ServerPushMJPEG.CPUUsage=20
ServerPushMJPEG.MemoryUsage=5
ServerPushMJPEG.ThreadsCount=9
ServerPushMJPEG.Duration=P0Y0M0DT2H15M0S
```

JSON RESPONSE

```
<Body>
```

```
--SamsungTechwin
Content-Type: application/json

{
    "TotalCPUUsage": 40,
    "TotalMemoryUsage": 30,
    "Apps": [
        {
            "AppID": "ANPR",
            "CPUUsage": 30,
            "MemoryUsage": 10,
            "ThreadsCount": 11,
            "Duration": "P0Y0M0DT1H0M0S"
        },
        {
            "AppID": "ServerPushMJPEG",
            "CPUUsage": 10,
            "MemoryUsage": 5,
            "ThreadsCount": 9,
            "Duration": "P0Y0M0DT2H10M0S"
        }
    ]
}

--SamsungTechwin
```

```
Content-Type: application/json

{
    "TotalCPUUsage": 40,
    "TotalMemoryUsage": 30,
    "Apps": [
        {
            "AppID": "ANPR",
            "CPUUsage": 20,
            "MemoryUsage": 10,
            "ThreadsCount": 11,
            "Duration": "P0Y0M0DT1H10M0S"
        },
        {
            "AppID": " ServerPushMJPEG",
            "CPUUsage": 20,
            "MemoryUsage": 5,
            "ThreadsCount": 9,
            "Duration": "P0Y0M0DT2H15M0S"
        }
    ]
}
```

Chapter 4. Application Manifest

4.1. Description

The **manifest** submenu requests the application manifest xml file which contains detailed information of the application such as the application name, location, version, etc.

Access level

Action	Camera
view	Admin

4.2. Syntax

```
http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=manifest&action=<value>[&<parameter>=<value>...]
```

4.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the application manifest in the xml file format.
	AppID	REQ	<string>	Application ID **Note** **AppID** must be sent together with the **view** action.
	Channel	REQ	<int>	Channel ID where the application is installed **Note** To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel

4.4. Examples

4.4.1. Getting the application manifest file

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=manifest&action=view&AppID=ServerPushMJPEG&Channel=0
```

RESPONSE

```
<Body>
```

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<manifest>
    <appName>ServerPushMJPEG</appName>
    <appLocation>/home/user/workspace/ServerPushMJPEG</appLocation>
    <appVersion>1.0</appVersion>
    <minSDK>2.0</minSDK>
    <targetSDK>2.0</targetSDK>
    <maxSDK>2.0</maxSDK>
    <debug>false</debug>
    <vendor>Hanwha</vendor>
    <description/>
    <platform>
        <model>SNP6320</model>
        <videoEncoding>
            <codec>MJPEG</codec>
            <resolution>640 X 480</resolution>
            <frameRate>10</frameRate>
            <compression>2</compression>
            <bitRate>10240</bitRate>
            <audio>false</audio>
        </videoEncoding>
        <rawVideo>
            <format>YUV 400</format>
            <resolution>1920 X 1080</resolution>
            <frameRate>3</frameRate>
        </rawVideo>
    </platform>
    <permissions/>
    <appConfigData>
        <portNo>8080</portNo>
    </appConfigData>
```

```
</manifest>
```

Chapter 5. Application Debug

5.1. Description

The **debug** submenu requests to debug the application using the 'RemoteDebugViewer' program. Only one application can be debugged at a time. All applications can be debugged.

Access level

Action	Camera
set	Admin

5.2. Syntax

```
http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=debug&action=<value>[&<parameter>=<value>...]
```

5.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
set	AppID	REQ, RES	<string>	Application ID **Note** **AppID**, **Port**, and **Enable** must be sent together.
	Port	REQ, RES	<int>	Camera's port number **Note** **AppID**, **Port**, and **Enable** must be sent together.
	Enable	REQ, RES	<bool>	Enabling debugging **Note** **AppID**, **Port**, and **Enable** must be sent together.

Action	Parameters	Request/ Response	Type/ Value	Description
	Channel	REQ	<int>	Channel ID where the application is installed **Note** An optional parameter. If **Channel** is not sent, the first channel's application will be debugged. To use the Channel parameter, check attributes/system/support/OneOpenAppPerChannel

5.4. Examples

5.4.1. Setting the application to debug

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=debug&action=set&AppID=abc&Port=8080&Enable=True&Channel=0
```

Chapter 6. Application Event Information

6.1. Description

The **opensdkeventinfo** submenu provides the open SDK application's event schema. Users can get event results from the open SDK application using eventstatus.cgi. Please refer to the document regarding this event.

Access level

Action	Camera
view	Admin

6.2. Syntax

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=opensdkeventinfo&action=<value>[&<parameter>=<value>...]
```

6.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	AppName	RES	<string>	Application name
	AppEvent	RES	<string>	Application event name
	EventTopic	RES	<string>	Event topic name
	Type	RES	<enum> Event, Meta	It can be either an Event or Metadata When type is event, schema follows the onvif event schema format. When type is Metadata, metadata xml schema is provided. e.g. Like licenplate information etc.,
	EventSchema	REQ	<string>	Event schema **Note** This schema information is set with an open application.

6.4. Examples

6.4.1. Getting the event result format from installed opensdk applications

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=opensdkeventinfo&action=view
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
    "OpenSDKEventInfo": [
        {
            "AppName": "LicensePlateDetection",
            "AppEvent": "LicensePlateNumber",
            "Type": "Event",
            "EventTopic":
"tns1:OpenApp/LicensePlateDetection/LicensePlateNumber",
            "EventSchema":
"<tns1:OpenApp><LicensePlateDetection><LicensePlateNumber
wstop:topic=\"true\"><tt:MessageDescription><tt:Source><tt:SimpleItemDescrip
tion Name=\"VideoSourceToken\"
Type=\"tt:ReferenceToken\"/></tt:Source><tt:Data><tt:SimpleItemDescription
Name=\"LicensePlateNumber\"
Type=\"xsd:string\"/></tt:Data></tt:MessageDescription></LicensePlateNumber>
</LicensePlateDetection></tns1:OpenApp>"
        },
        {
            "AppName": "VehicleDetection",
            "AppEvent": "VehicleDetected",
            "Type": "Event",
            "EventTopic": "tns1:OpenApp/VehicleDetection/VehicleDetected",
            "EventSchema": "<tns1:OpenApp><VehicleDetection><VehicleDetected
wstop:topic=\"true\"><tt:MessageDescription><tt:Source><tt:SimpleItemDescrip
tion Name=\"VideoSourceToken\"
Type=\"tt:ReferenceToken\"/></tt:Source><tt:Data><tt:SimpleItemDescription
Name=\"VehicleDetected\"
Type=\"xsd:boolean\"/></tt:Data></tt:MessageDescription></VehicleDetected></
```

```
VehicleDetection></tns1:OpenApp>"
        }
    ]
}
```

Chapter 7. Metaframe Schema

7.1. Description

The **metaframeschema** submenu, used to provide the schema of the frame metadata supported by an installed app.

Access level

Action	Camera
view	User

7.2. Syntax

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=metaframeschema&action=<value>[&<parameter>=<value>...]
```

7.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Channel	REQ, RES	<int>	ChannelID, optional parameter in the request. If passed, the result will be filtered only for that channel.
	AppID	REQ, RES	<string>	Application ID
	Schema	RES	<string>	Frame metadata schema as base64 encoded string.
	Encoding	RES	<enum> base64	Used to notify the encoding format of schema.

7.4. Examples

7.4.1. Getting the schema of frame metadata supported by an app.

REQUEST

```
http://<Device IP>/stw-cgi/opensdk.cgi?msubmenu=metaframeschema&action=view
```

JSON RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: application/json
<Body>
```

```
{
    "MetaFrameSchema": [
        {
            "Channel": 0,
            "AppFrameSchema": [
                {
                    "AppID": "SampleAIAPP",
                    "Schema":
"PHhzOnNjaGVtYSBhdHRyaWJ1dGVGb3JtRGVmYXVsdD0idW5xdWFsaWZpZWQiIGVsZW1lbnRGb3J
tRGVmYXVsdD0icXVhbGlmaWVkIiB0YXJnZXROYW1lc3BhY2U9Imh0dHA6Ly93d3cub252aWYub3J
nL3ZlcjEwL3NjaGVtYSIgeG1sbnM6eHM9Imh0dHA6Ly93d3cudzMub3JnLzIwMDEvWE1MU2NoZW1
hIj48eHM6ZWxlbWVudCBuYW1lPSJNZXRhZGF0YVN0cmVhbSI+PHhzOmNvbXBsZXhUeXBlPjx4czp
zZXF1ZW5jZT48eHM6ZWxlbWVudCBuYW1lPSJWaWRlb0FuYWx5dGljcyI+PHhzOmNvbXBsZXhUeXB
lPjx4czpzZXF1ZW5jZT48eHM6ZWxlbWVudCBuYW1lPSJGcmFtZSI+PHhzOmNvbXBsZXhUeXBlPjx
4czpzZXF1ZW5jZT48eHM6ZWxlbWVudCBuYW1lPSJUcmFuc2Zvcm1hdGlvbiI+PHhzOmNvbXBsZXh
UeXBlPjx4czpzZXF1ZW5jZT48eHM6ZWxlbWVudCBuYW1lPSJUcmFuc2xhdGUiPjx4czpjb21wbGV
4VHlwZT48eHM6c2ltcGxlQ29udGVudD48eHM6ZXh0ZW5zaW9uIGJhc2U9InhzOnN0cmluZyI+PHh
zOmF0dHJpYnV0ZSB0eXBlPSJ4czpmbG9hdCIgbmFtZT0ieCIvPjx4czphdHRyaWJ1dGUgdHlwZT0
ieHM6ZmxvYXQiIG5hbWU9InkiLz48L3hzOmV4dGVuc2lvbj48L3hzOnNpbXBsZUNvbnRlbnQ+PC9
4czpjb21wbGV4VHlwZT48L3hzOmVsZW1lbnQ+PHhzOmVsZW1lbnQgbmFtZT0iU2NhbGUiPjx4czp
jb21wbGV4VHlwZT48eHM6c2ltcGxlQ29udGVudD48eHM6ZXh0ZW5zaW9uIGJhc2U9InhzOnN0cml
uZyI+PHhzOmF0dHJpYnV0ZSB0eXBlPSJ4czpmbG9hdCIgbmFtZT0ieCIvPjx4czphdHRyaWJ1dGU
gdHlwZT0ieHM6ZmxvYXQiIG5hbWU9InkiLz48L3hzOmV4dGVuc2lvbj48L3hzOnNpbXBsZUNvbnR
lbnQ+PC94czpjb21wbGV4VHlwZT48L3hzOmVsZW1lbnQ+PC94czpzZXF1ZW5jZT48L3hzOmNvbXB
sZXhUeXBlPjwveHM6ZWxlbWVudD48eHM6ZWxlbWVudCBuYW1lPSJPYmplY3QiPjx4czpjb21wbGV
4VHlwZT48eHM6c2VxdWVuY2U+PHhzOmVsZW1lbnQgbmFtZT0iQXBwZWFyYW5jZSI+PHhzOmNvbXB
sZXhUeXBlPjx4czpzZXF1ZW5jZT48eHM6ZWxlbWVudCBuYW1lPSJTaGFwZSI+PHhzOmNvbXBsZXh
UeXBlPjx4czpzZXF1ZW5jZT48eHM6ZWxlbWVudCBuYW1lPSJCb3VuZGluZ0JveCI+PHhzOmNvbXB
sZXhUeXBlPjx4czpzaW1wbGVDb250Pjx4czpleHRlbnNpb24gYmFzZT0ieHM6c3RyaW5nIj48
8eHM6YXR0cmlidXRlIHR5cGU9InhzOmZsb2F0IiBuYW1lPSJsZWZ0IiBmaXhlZD0ieHM6c3RyaW5nIj4
0eXBlPSJ4czpmbG9hdCIgbmFtZT0idG9wIi8+PHhzOmF0dHJpYnV0ZSB0eXBlPSJ4czpmbG9hdCI
gbmFtZT0icmlnaHQiLz48eHM6YXR0cmlidXRlIHR5cGU9InhzOmZsb2F0IiBuYW1lPSJib3R0b20
iLz48L3hzOmV4dGVuc2lvbj48L3hzOnNpbXBsZUNvbnRlbnQ+PC94czpjb21wbGV4VHlwZT48L3h
zOmVsZW1lbnQ+PHhzOmVsZW1lbnQgbmFtZT0iQ2VudGVyT2ZHcmF2aXR5Ij48eHM6Y29tcGxleFR
5cGU+PHhzOnNpbXBsZUNvbnRlbnQ+PHhzOmV4dGVuc2lvbiBiYXNlPSJ4czpzdHJpbmciPjx4czp
hdHRyaWJ1dGUgdHlwZT0ieHM6ZmxvYXQiIG5hbWU9IngiLz48eHM6YXR0cmlidXRlIHR5cGU9Inh
zOmZsb2F0IiBuYW1lPSJ5Ii8+PC94czpleHRlbnNpb24+PC94czpzaW1wbGVDb250ZW50PjwveHM
```

```
6Y29tcGxleFR5cGU+PC94czplbGVtZW50PjwveHM6c2VxdWVuY2U+PC94czpjb21wbGV4VHlwZT4
8L3hzOmVsZW1lbnQ+PHhzOmVsZW1lbnQgbmFtZT0iQ29sb3IiPjx4czpjb21wbGV4VHlwZT48eHM
6c2VxdWVuY2U+PHhzOmVsZW1lbnQgbmFtZT0iQ29sb3JDbHVzdGVyIj48eHM6Y29tcGxleFR5cGU
+PHhzOnNlcXVlbmNlPjx4czplbGVtZW50IG5hbWU9IkNvbG9yIj48eHM6Y29tcGxleFR5cGU+PHh
zOnNpbXBsZUNvbnRlbnQ+PHhzOmV4dGVuc2lvbiBiYXNlPSJ4czpzdHJpbmciPjx4czphdHRyaWJ
1dGUgdHlwZT0ieHM6Ynl0ZSIgbmFtZT0iWCIvPjx4czphdHRyaWJ1dGUgdHlwZT0ieHM6Ynl0ZSI
gbmFtZT0iWSIvPjx4czphdHRyaWJ1dGUgdHlwZT0ieHM6c2hvcnQiIG5hbWU9IloiLz48L3hzOmV
4dGVuc2lvbj48L3hzOnNpbXBsZUNvbnRlbnQ+PC94czpjb21wbGV4VHlwZT48L3hzOmVsZW1lbnQ
+PHhzOmVsZW1lbnQgbmFtZT0iQ292YXJpYW5jZSI+PHhzOmNvbXBsZXhUeXBlPjx4czpzaW1wbGV
Db250ZW50Pjx4czpleHRlbnNpb24gYmFzZT0ieHM6c3RyaW5nIj48eHM6YXR0cmlidXRlIHR5cGU
9InhzOmZsb2F0IiBuYW1lPSJYWCIvPjx4czphdHRyaWJ1dGUgdHlwZT0ieHM6Ynl0ZSIgbmFtZT0
iWVkiLz48eHM6YXR0cmlidXRlIHR5cGU9InhzOmJ5dGUiIG5hbWU9IlpaIi8+PC94czpleHRlbnN
pb24+PC94czpzaW1wbGVDb250ZW50PjwveHM6Y29tcGxleFR5cGU+PC94czplbGVtZW50Pjx4czp
lbGVtZW50IHR5cGU9InhzOmZsb2F0IiBuYW1lPSJXZWlnaHQiLz48L3hzOnNlcXVlbmNlPjwveHM
6Y29tcGxleFR5cGU+PC94czplbGVtZW50PjwveHM6c2VxdWVuY2U+PC94czpjb21wbGV4VHlwZT4
8L3hzOmVsZW1lbnQ+PHhzOmVsZW1lbnQgbmFtZT0iQ2xhc3MiPjx4czpjb21wbGV4VHlwZT48eHM
6c2VxdWVuY2U+PHhzOmVsZW1lbnQgbmFtZT0iVHlwZSI+PHhzOmNvbXBsZXhUeXBlPjx4czpzaW1
wbGVDb250ZW50Pjx4czpleHRlbnNpb24gYmFzZT0ieHM6c3RyaW5nIj48eHM6YXR0cmlidXRlIHR
5cGU9InhzOmZsb2F0IiBuYW1lPSJMaWtlbGlob29kIi8+PC94czpleHRlbnNpb24+PC94czpzaW1
wbGVDb250ZW50PjwveHM6Y29tcGxleFR5cGU+PC94czplbGVtZW50PjwveHM6c2VxdWVuY2U+PC9
4czpjb21wbGV4VHlwZT48L3hzOmVsZW1lbnQ+PC94czpzZXF1ZW5jZT48L3hzOmNvbXBsZXhUeXB
lPjwveHM6ZWxlbWVudD48L3hzOnNlcXVlbmNlPjx4czphdHRyaWJ1dGUgdHlwZT0ieHM6Ynl0ZSI
gbmFtZT0iT2JqZWN0SWQiLz48eHM6YXR0cmlidXRlIHR5cGU9InhzOmJ5dGUiIG5hbWU9IlBhcmV
udCIvPjwveHM6Y29tcGxleFR5cGU+PC94czplbGVtZW50PjwveHM6c2VxdWVuY2U+PHhzOmF0dHJ
pYnV0ZSB0eXBlPSJ4czpkYXRlVGltZSIgbmFtZT0iVXRjVGltZSIvPjwveHM6Y29tcGxleFR5cGU
+PC94czplbGVtZW50PjwveHM6c2VxdWVuY2U+PC94czpjb21wbGV4VHlwZT48L3hzOmVsZW1lbnQ
+PC94czpzZXF1ZW5jZT48L3hzOmNvbXBsZXhUeXBlPjwveHM6ZWxlbWVudD48L3hzOnNjaGVtYT4
=",
                    "Encoding": "base64"
                }
            ]
        }
    ]
}
```

Chapter 8. Metaframe Capability

8.1. Description

The **metaframecapability** submenu, used to provide all supported values of a metadata field. Therefore, the client can know what to expect based on this capability.

XPATH based capability notification mechanism is being used.

Providing the XPATH of the parameter and its data type and supported values or range.

If there are dependencies with another parameter, the dependency section is provided, for which XPATH affects the supported values of this parameter.

Please check the example section.

Access level

Action	Camera
view	User

8.2. Syntax

```
http://<Device IP>/stw-
cgi/opensdk.cgi?msubmenu=metaframecapability&action=<value>[&<parameter>=<va
lue>...]
```

8.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Channel	REQ, RES	<int>	ChannelID, optional parameter in request. If passed, result will be filtered only for that channel.
	AppID	REQ, RES	<string>	Application ID
	Capabilities	RES	<string>	Capability Response

8.4. Examples

8.4.1. Getting the metaframe capability of the installed apps.

REQUEST

```
http://<Device IP>/stw-
```

```
cgi/opensdk.cgi?msubmenu=metaframecapability&action=view
```

JSON RESPONSE

```
HTTP/1.0 200 OK
Content-type: application/json
<Body>
```

```
{
    "MetaFrameCapability": [
        {
            "Channel": 0,
            "AppCapabilities": [
                {
                    "AppID": "SampleAIApp",
                    "Capabilities": [
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:LicensePlateInfo/tt
:CountryCode",
                            "type": "xs:string",
                            "enum": [
                                "KR",
                                "US",
                                "CN",
                                "FN",
                                "IN"
                            ]
                        },
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:LicensePlateInfo/tt
:PlateType",
                            "type": "xs:string",
                            "enum": [
                                "Normal",
                                "Police",
                                "Diplomat",
                                "Temporary"
                            ]
                        },
```

```
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:Class/tt:Type",
                            "type": "xs:string",
                            "enum": [
                                "Vehicle",
                                "Bicycle"
                            ]
                        },
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Type
",
                            "type": "xs:string",
                            "enum": [
                                "Car",
                                "Bus"
                            ]
                        },
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Bran
d",
                            "type": "xs:string",
                            "enum": [
                                "Kia",
                                "Hyundai",
                                "Volvo"
                            ],
                            "Dependency": [
                                {
                                    "Condition":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Type
[text()='Car']",
                                    "enum": [
                                        "Kia",
                                        "Hyundai"
                                    ]
                                },
                                {
                                    "Condition":
```

```
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Type
[text()='Bus']",
                                    "enum": [
                                        "Volvo"
                                    ]
                                }
                            ]
                        },
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Mode
l",
                            "type": "xs:string",
                            "enum": [
                                "K7",
                                "Sonata",
                                "Accent",
                                "Soul"
                            ],
                            "Dependency": [
                                {
                                    "Condition":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Type
[text()='Car'] and
//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Brand
[text()='Kia']",
                                    "enum": [
                                        "K7",
                                        "Soul"
                                    ]
                                },
                                {
                                    "Condition":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Type
[text()='Car'] and
//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:VehicleInfo/tt:Brand
[text()='Hyundai']",
                                    "enum": [
                                        "Sonata",
                                        "Accent"
                                    ]
```

```
                                }
                            ]
                        },
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:Class/tt:Type/@Like
lihood",
                            "type": "xs:float",
                            "minimum": 0,
                            "maximum": 1
                        },
                        {
                            "xpath":
"//tt:VideoAnalytics/tt:Frame/tt:Object/tt:Appearance/tt:Color/tt:ColorClust
er/tt:ColorString",
                            "type": "xs:string",
                            "enum": [
                                "RED",
                                "BLUE",
                                "GREEN",
                                "BLACK",
                                "WHITE"
                            ]
                        }
                    ]
                }
            ]
        }
    ]
}
```