System

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

system.cgi configures general system settings for Hanwha Vision video surveillance products. These settings include device information, system date and time, and serial port settings. **system.cgi** can also perform important system functions like factory resets, firmware updates and accessing logs.

The following submenus are used to configure system settings:

- **deviceinfo**: Sets and requests device information.
- **date**: Sets and requests the system date, time and time zone.
- **serial**: Configures the RS-485 settings.
- **systemlog**: Reads the system log.
- **accesslog**: Reads the HTTP client access log.
- **eventlog**: Reads the event log.
- **profileaccessinfo**: Requests information on the profile currently being used and on the users accessing the current profile.
- **factoryreset**: Resets the system back to factory default settings. It offers the option to exclude the network configuration, camera mapping information or user access level information.
- **power**: Resets the system power.
- **firmwareupdate**: Updates the device firmware.
- **configbackup**: Makes a copy of all the system settings for backup.
- **configrestore**: Restores the system configuration by using the backup.
- **storageinfo**: Requests storage device information.
- **gps**: Requests the GPS (global positioning system) information of the NVR.
- **autobackup**: Makes a backup of the videos recorded on NVR automatically on the server when the NVR is connected to the server.
- **digitalsignage**: Displays advertisements from the FTP server if a certain input or event does not occur for the specified time.
- **vehicleinformation**: Stores the vehicle-related information if mobile NVR is used.
- **onviffeature**: Enables some features in the ONVIF protocol.
- **databasereset**: Resets the database content in the camera.
- **logserver**: Configure clients to receive log information from device.
- **sessioninfo**: Gets the current session information of NVR.
- **sdcardinfo**: Gets the details of the current SD card.
- **iscsidiscovery**: Used to get the ISCSI targets.
- **holiday**: Configures the holiday settings for the device.

- **hddalarm**: Configures the hddalram settings for the device.
- **monitorin**: Configures the monitor input settings for the device.
- **monitorin**: Configures the monitor output settings for the device.
- **usbconfig**: Configures the USB port on the device.
- **stratocast**: Configures the stratocast service URL and communication interval.
- **stratocastregister**: Reads the current status regarding the stratocast service registration.
- **peerconnectioninfo**: Reads the current status of the peer connection information.
- **clientregister**: Client can connect to IO Box and checks connection status.
- **geolocation**: Gets and configures the geolocation information of the device.
- **systemimage**: Gets the image files used by the NVR device to the client.
- **powermode**: Used to configure the power mode of the device.
- **registeredsubdevices**: Used to manage the network speaker and mic in Audio Management System (AMS).
- **speakergroups**: Used to mananger speaker groupds in Audio Management System (AMS).
- **ssdstorage**: Used to manage the ssd storage in a SSD storage supported cameras.
- **localvms**: Used to install and manage the VMS application running inside camera.

Chapter 2. Device Information

2.1. Description

The **deviceinfo** submenu requests and sets device information.

Access level

Action	Camera	Encoder	NVR	Decoder
view	Guest	Guest	User	User
set	Admin	Admin	User	User

2.2. Syntax

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

2.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the device information settings.
	DeviceReady	RES	<bool>	Gives device ready status DECODER ONLY
	Model	RES	<string>	Model name
	SerialNumber	RES	<string>	Device serial number CAMERA ONLY ENCODER ONLY AMS ONLY
	FirmwareVersion	RES	<string>	Current firmware version
	BuildDate	RES	<string>	Date the firmware was built
	WebURL	RES	<string>	Web address

Action	Parameters	Request/ Response	Type/ Value	Description
	DeviceType	RES	<enum> NWC, NVR, DVR, Encoder, Decoder, Hybrid, IOBox, NetworkSpeaker, NetworkMic	Device type
	SpeakerType	RES	<enum> MASTER,SLAVE, SERVER, MODULE	When DeviceType is NetworkSpeaker it can be a master speaker, slave speaker, server or module AMS ONLY
	ConnectedMACAddress	RES	<string>	MAC address
	ISPVersion	RES	<string>	ISP version CAMERA ONLY
	PTZBoardVersion	RES	<string>	PTZ board version CAMERA ONLY
	InterfaceBoardVersion	RES	<string>	Interface board version CAMERA ONLY
	TrackingVersion	RES	<string>	Tracking version CAMERA ONLY
	BootloaderVersion	RES	<string>	Boot loader version CAMERA ONLY ENCODER ONLY
	CGIVersion	RES	<string>	CGI version
	GUIVersion	RES	<string>	Recorder's GUI version NVR ONLY
	MicomVersion	RES	<string>	Micom version NVR ONLY

Action	Parameters	Request/ Response	Type/ Value	Description
	PasswordStrength	RES	<enum> Weak, Strong	Strength of the password as per the password policy CAMERA ONLY ENCODER ONLY
	OpenSDKVersion	RES	<string>	Open SDK application version CAMERA ONLY
	ONVIFVersion	RES	<string>	ONVIF transmitter version supported by device CAMERA ONLY
	RequestedClientIPAddress	RES	<string>	Client's address as seen by device. NVR ONLY
	ActualDeviceType	RES	<enum> Encoder	When the device is an encoder and **DeviceType** is changed to the network camera, this parameter shows up.
set	DeviceName	REQ, RES	<string>	Device name
	DeviceLocation	REQ, RES	<string>	Location information CAMERA ONLY ENCODER ONLY
	Memo	REQ, RES	<string>	Additional information about the device CAMERA ONLY ENCODER ONLY
	DeviceDescription	REQ, RES	<string>	Detailed information about the device CAMERA ONLY ENCODER ONLY
	Language	REQ, RES	<enum>	Language of the interface
	CameraRegistrationMode	REQ,RES	<enum> Normal, PnP	Camera registration mode in PoE-supported NVR. NVR ONLY
	DeviceMode	REQ, RES	<enum> StandAlone, VMS	Mode of the device DECODER ONLY

2.4. Examples

2.4.1. Getting device information

REQUEST

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

TEXT RESPONSE

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

```
Model=XND-8080R
SerialNumber=ZECA6V2HB00008M
FirmwareVersion=1.29.99_20190125
BuildDate=2019.01.25
WebURL=http://www.hanwhavision.com/
DeviceType=NWC
ConnectedMACAddress=00:16:6C:F9:1F:EE
ISPVersion=1.46_180907
BootloaderVersion=ver=U-Boot 2016.01-svn2152 (Nov 29 2017 - 20:20:02
CGIVersion=2.5.5
ONVIFVersion=18.6
DeviceName=Camera
DeviceLocation=Location
DeviceDescription=Description
Memo=Memo
Language=English
PasswordStrength=Strong
OpenSDKVersion=3.50_181011
FirmwareGroup=XND-8080R
```

JSON RESPONSE

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

```
{
```

```
    "Model": "XND-8080R",
    "SerialNumber": "ZECA6V2HB00008M",
    "FirmwareVersion": "1.29.99_20190125",
    "BuildDate": "2019.01.25",
    "WebURL": "http://www.hanwhavision.com/",
    "DeviceType": "NWC",
    "ConnectedMACAddress": "00:16:6C:F9:1F:EE",
    "ISPVersion": "1.46_180907",
    "BootloaderVersion": "ver=U-Boot 2016.01-svn2152 (Nov 29 2017 - 20:20:02
",
    "CGIVersion": "2.5.5",
    "ONVIFVersion": "18.6",
    "DeviceName": "Camera",
    "DeviceLocation": "Location",
    "DeviceDescription": "Description",
    "Memo": "Memo",
    "Language": "English",
    "PasswordStrength": "Strong",
    "OpenSDKVersion": "3.50_181011",
    "FirmwareGroup": "XND-8080R"
}
```


2.4.2. Setting the language as English

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=deviceinfo&action=set&Language=English
```

Chapter 3. Date and Time

3.1. Description

The **date** submenu requests and sets the system date, time and time zone.

Access level

Action	Camera	Encoder	NVR	Decoder
view	Guest	Guest	User	User
set	Admin	Admin	User	User

3.2. Syntax

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

3.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Read system date and time settings
	TimeZoneList	REQ		Time zone list The response contains a list of time zones (with indexes) supported by the device. The time zone index should be used to set TimeZoneIndex. CAMERA ONLY ENCODER ONLY
	LocalTime	RES	<string>	Local time The time is defined in the <YYYY-MM-DD hh:mm:ss> format.
	UTCTime	RES	<string>	UTC time The time is defined in the <YYYY-MM-DD hh:mm:ss> format.

Action	Parameters	Request/ Response	Type/ Value	Description
	NTPStatus	RES	<enum> Success, Fail	NTP status This parameter is valid only when **SyncType** is set to NTP. NVR ONLY
	NTPLastUpdatedTime	RES	<string>	NTP last updated time This parameter is valid only when **SyncType** is set to NTP. NVR ONLY
	Week	RES	<enum> First, Second, Third, Fourth, Last	Week of the current month NVR ONLY DECODER ONLY
	DateFormat	RES	<enum> YYYY-MM-DD,DD-MM-YYYY,MM-DD-YYYY	The current system's date display format NVR ONLY
	TimeFormat	RES	HMS24, HMS12	The current system's hour display format NVR ONLY
set	SyncType	REQ, RES	<enum> NTP, Manual, GPS	Time synchronization types • NTP: Time sync with NTP • Manual: Manual time synchronization (**Year**, **Month**, **Day**, **Hour**, **Minute**, and **Second** parameters should be set together.) • GPS: Use GPS as the source (NVR Only) **Note** Either **SyncType** or **TimeZoneIndex** must be sent for the **set** action.

Action	Parameters	Request/ Response	Type/ Value	Description
	NTPURLList	REQ, RES	<csv>	NTP URL list This parameter is valid only when **SyncType** is set to NTP.
	DSTEnable	REQ, RES	<bool> True, False	Enables or disables daylight saving time adjustments. This parameter must be set together with **TimeZoneIndex**.
	TimeZoneIndex	REQ, RES	<int>	Index number of time zone **Note** Either **SyncType** or **TimeZoneIndex** must be sent for the **set** action. CAMERA ONLY ENCODER ONLY
	POSIXTimeZone	REQ, RES	<string>	Time zone with or without DST (daylight saving time)
	Year	REQ	<int>	Year of the system time This parameter is valid only when **SyncType** is set to Manual.
	Month	REQ	<int>	Month of the system time The values must be within the range of 1 to 12. This parameter is valid only when **SyncType** is set to Manual.
	Day	REQ	<int>	Day of the system time The values must be within the range of 1 to 31. This parameter is valid only when **SyncType** is set to Manual.

Action	Parameters	Request/ Response	Type/ Value	Description
	Hour	REQ	<int>	Hour of the system time The values must be within the range of 0 to 23. This parameter is valid only when **SyncType** is set to Manual.
	Minute	REQ	<int>	Minute of the system time The values must be within the range of 0 to 59. This parameter is valid only when **SyncType** is set to Manual.
	Second	REQ	<int>	Second of the system time The values must be within the range of 0 to 59. This parameter is valid only when **SyncType** is set to Manual.
	TimeZone	REQ, RES	<enum>	TimeZone information based on GMT NVR ONLY DECODER ONLY
	DateFormat	RES	<enum> YYYY-MM-DD,DD-MM-YYYY,MM-DD-YYYY	Sets the current system's date display format NVR ONLY
	TimeFormat	RES	HMS24, HMS12	Sets the current system's hour display format NVR ONLY

3.4. Examples

3.4.1. Getting system date/time information

The date and time information are provided in both local time and UTC time.

REQUEST

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

TEXT RESPONSE

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

```
NTPURLList=pool.ntp.org,asia.pool.ntp.org,europe.pool.ntp.org,north-
america.pool.ntp.org,time.nist.gov
LocalTime=2015-06-29 07:26:27
UTCTime=2015-06-29 07:26:27
SyncType=Manual
DSTEnable=False
TimeZoneIndex=33
POSIXTimeZone=STWT0STWST,M3.5.0/1,M10.5.0/2:0:0
```

JSON RESPONSE

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

```
{
    "NTPURLList": [
        "pool.ntp.org",
        "asia.pool.ntp.org",
        "europe.pool.ntp.org",
        "north-america.pool.ntp.org",
        "time.nist.gov"
    ],
    "LocalTime": "2015-06-29 07:26:27",
    "UTCTime": "2015-06-29 07:26:27",
    "SyncType": "Manual",
    "DSTEnable": false,
    "TimeZoneIndex": 33,
    "POSIXTimeZone": "STWT0STWST,M3.5.0/1,M10.5.0/2:0:0"
}
```

The following response example is for NVR only.

TEXT RESPONSE

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

```
NTPURLList=pool.ntp.org,asia.pool.ntp.org,europe.pool.ntp.org,north-
america.pool.ntp.org,time.nist.gov
LocalTime=2015-06-29 07:26:27
UTCTime=2015-06-29 07:26:27
SyncType=Manual
DSTEnable=False
TimeZoneIndex=33
POSIXTimeZone=STWT0STWST,M3.5.0/1,M10.5.0/2:0:0
DateFormat=YYYY-MM-DD
TimeFormat=HMS24
```

JSON RESPONSE

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

```
{
    "NTPLastUpdatedTime": "2015-06-29 16:32:24",
    "LocalTime": "2015-06-29 16:32:24",
    "UTCTime": "2015-06-29 07:32:24",
    "SyncType": "Manual",
    "DSTEnable": false,
    "POSIXTimeZone": "STWT-9STWST,M3.5.0/1:00:00,M10.5.0/1:00:00",
    "DateFormat": "YYYY-MM-DD",
    "TimeFormat": "HMS24"
}
```

3.4.2. Time zones

Getting the time zone list

The **TimeZoneList** returns a list of available time zones and their index numbers, which **TimeZoneIndex** refers to for setting time zones.

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=date&action=view&TimeZoneList
```

TEXT RESPONSE

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

```
00:(GMT-12:00) International Date Line West
01:(GMT-11:00) Coordinated Universal Time-11
02:(GMT-10:00) Hawaii
03:(GMT-09:00) Alaska[March.2nd.Sun/02:00:00,November.1st.Sun/02:00:00]
04:(GMT-08:00) Pacific Time (US &
Canada)[March.2nd.Sun/02:00:00,November.1st.Sun/02:00:00]
05:(GMT-08:00) Baja
California[April.1st.Sun/02:00:00,October.last.Sun/02:00:00]
06:(GMT-07:00) Chihuahua, La Paz,
Mazatlan[April.1st.Sun/02:00:00,October.last.Sun/02:00:00]
07:(GMT-07:00) Mountain Time (US &
Canada)[March.2nd.Sun/02:00:00,November.1st.Sun/02:00:00]
08:(GMT-07:00) Arizona
09:(GMT-06:00) Saskatchewan
10:(GMT-06:00) Central America
```

JSON RESPONSE

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

```
{
    "TimeZones": [
        {
            "TimeZoneIndex": 0,
            "TimeZone": "(GMT-12:00) International Date Line West"
        },
        {
            "TimeZoneIndex": 1,
```

```
            "TimeZone": "(GMT-11:00) Coordinated Universal Time-11"
        },
        {
            "TimeZoneIndex": 2,
            "TimeZone": "(GMT-10:00) Hawaii"
        },
        {
            "TimeZoneIndex": 3,
            "TimeZone": "(GMT-09:00) Alaska",
            "StartTime": "March.2nd.Sun/02:00:00",
            "EndTime": "November.1st.Sun/02:00:00"
        },
        {
            "TimeZoneIndex": 4,
            "TimeZone": "(GMT-08:00) Pacific Time (US & Canada)",
            "StartTime": "March.2nd.Sun/02:00:00",
            "EndTime": "November.1st.Sun/02:00:00"
        },
        {
            "TimeZoneIndex": 5,
            "TimeZone": "(GMT-08:00) Baja California",
            "StartTime": "April.1st.Sun/02:00:00",
            "EndTime": "October.last.Sun/02:00:00"
        },
        {
            "TimeZoneIndex": 6,
            "TimeZone": "(GMT-07:00) Chihuahua, La Paz, Mazatlan",
            "StartTime": "April.1st.Sun/02:00:00",
            "EndTime": "October.last.Sun/02:00:00"
        },
        {
            "TimeZoneIndex": 7,
            "TimeZone": "(GMT-07:00) Mountain Time (US & Canada)",
            "StartTime": "March.2nd.Sun/02:00:00",
            "EndTime": "November.1st.Sun/02:00:00"
        },
        {
            "TimeZoneIndex": 8,
            "TimeZone": "(GMT-07:00) Arizona"
        },
        {
```

```
            "TimeZoneIndex": 9,
            "TimeZone": "(GMT-06:00) Saskatchewan"
        },
        {
            "TimeZoneIndex": 10,
            "TimeZone": "(GMT-06:00) Central America"
        }
    ]
}
```

NOTE The time zone can be changed year by year.

Setting the time zone for Hawaii

To set the time zone for Hawaii, **TimeZoneIndex** should be set to 02. (Refer to the data returned by **TimeZoneList**.)

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=date&action=set&TimeZoneIndex=02
```

3.4.3. Daylight saving time

To enable daylight saving time, **DSTEnable** should be set together with **TimeZoneIndex**.

Setting the system time to October 10, 2012, 00:10:00 in the Alaska time zone with DST

Setting the Alaska time zone with DST will change the system time 1 hour ahead of standard Alaska local time.

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=date&action=set&SyncType=Manual&Year=2012&Month=10&Day=10&Hour=0&Minute=10&Second=0&TimeZoneIndex=03&DSTEnable=True
```

The following request example is for NVR only.

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=date&action=set&SyncType=Manual&Year=2012&Month=10&Day=10&Hour=0&Minute=10&Second=0&DSTEnable=True&POSIXTimeZone=STWT+9
```

3.4.4. POSIX TimeZone

POSIXTimeZone requests the time zone with or without DST. The following example is for Istanbul.

```
POSIXTimeZone=STWT-2STWST,M3.5.0/3,M10.5.0/4:0:0
```

It follows the syntax of POSIX Time zone;

```
std offset dst[offset],start[/time],end[/time]
STWT -2 STWST ,M3.5.0/3 ,M10.5.0/4:0:0
```

STWT-2STWST indicates that it is 2 hours ahead of GMT. M3.5.0/3,M10.5.0/4:0:0 indicates that daylight saving time starts on Sunday (0), the last week (5) of March (M3), at 3:00 AM, and it ends on Sunday (0), the last week (5) of October (M10), at 4:00 AM.

3.4.5. System time sync

Using Manual Time Sync and changing the date and time to January 1, 2000, 00:10:31

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=date&action=set&SyncType=Manual&Year=2000&Month=1&Da
y=1&Hour=0&Minute=10&Second=31
```

TEXT RESPONSE

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

```
OK
```

JSON RESPONSE

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

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

```
}
```

Chapter 4. RS-485

4.1. Description

The **serial** submenu configures the RS-485 settings.

Access level

Action	Camera	Encoder	NVR
view	Admin	Admin	User
set	Admin	Admin	(Not supported)

4.2. Syntax

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

4.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the RS-485/RS-422 settings.
set	SerialInterface	REQ, RES	RS422, RS485	Chooses the serial mode. CAMERA ONLY
	BaudRate	REQ, RES	<enum> 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200	Baud rate (read only for NVR)
	ParityBit	REQ, RES	<enum> None, Even, Odd	Parity bit mode (read only for NVR) • None: Off • Even: Even parity • Odd: Odd parity
	StopBits	REQ, RES	<enum> 1, 2	Stop bit (read only for NVR)

Action	Parameters	Request/ Response	Type/ Value	Description
	DataBits	REQ, RES	<enum> 7, 8	Data bits (read only for NVR)
	SignalTermination	REQ, RES	<bool> True, False	Sets the signal ending status CAMERA ONLY ENCODER ONLY
	DeviceId	REQ, RES	<int>	Hybrid NVR ID NVR ONLY

4.4. Examples

4.4.1. Getting serial port settings

REQUEST

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

TEXT RESPONSE

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

```
BaudRate=115200
ParityBit=None
StopBits=1
DataBits=8
```

JSON RESPONSE

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

```
{
    "BaudRate": "115200",
    "ParityBit": "None",
    "StopBits": "1",
```

```
    "DataBits": "8"
}
```

4.4.2. Configuring serial port settings

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=serial&action=set&BaudRate=9600&ParityBit=None&StopB
its=2&DataBits=8
```

Chapter 5. System Log

5.1. Description

The **systemlog** submenu reads the system log. Each line of a system log response consists of the following:

- Date
- Time
- Type
- Description

Access level

Action	Camera	Encoder	NVR
view	Admin	Admin	User

5.2. Syntax

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

5.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Level	REQ, RES	<enum> All	Log level CAMERA ONLY ENCODER ONLY
	Type	REQ, RES	<csv> Refer to **SystemLog Types** block below.	Detailed log type
	FromDate	REQ	<string>	Search start date The date is specified in the format of <YYYY-MM-DD>.
	ToDate	REQ	<string>	Search end date The date is specified in the format of<YYYY-MM-DD>.

SystemLog Types

```
 PowerOn, PowerOff, PowerRebootConfigChange, Backup, FWUpdate, FactoryReset,
HDDFull, HDDFail, HDDNone, FanError, SDFormat, SDFail, SDFull, SDInsert,
SDRemove, Network, TimeChange, Record, ConfigurationBackup,
ConfigurationRestore, NASFormat, NASFail, NASFull, NASConnect,
NASDisconnect, LocalSetupStart, LocalSetupEnd, RemoteSetup, PlaybackStart,
PlaybackEnd, CodecError, SystemUpgrade, DiskFull, DiskFail,
RecoverFromPowerFail, BackupStart, BackupEnd, BackupStop, BackupFail,
BackupOverwrite, InternalHDDErase, ExternalHDDErase, USBHDDErase,
USBMemoryErase, StreamCorrupt, ManualRecordStart, ManualRecordEnd,
OverwriteDecoding, DecoderRestart, HDDError, PTZModeIn, PTZModeOut,
RecordingError, NetworkBackupStart, NetworkBackupEnd, NetworkBackupStop,
AutoDeleteStart, AutoDeleteEnd, PasswordChange, RebootExternalHDD,
FrameFanFail, FrameFanRepair, CPUFanFail, CPUFanRepair, LeftFanFail,
LeftFanRepair, RightFanFail, RightFanRepair, EmergencyReset,
NetCamTrafficOverflow, NetCamTrafficRelease, DualSMPSFail, DualSMPSRepair,
iSCSIAttach, iSCSIDetach, iSCSIConnect, iSCSIDisconnect, RecordFrameDrop,
AlarmOut1, AlarmOut2, AlarmOut3, AlarmOut4, AlarmOutBeep, Net1Connect,
Net1Disconnect, Net2Connect, Net2Disconnect, Net3Connect, Net3Disconnect,
Net4Connect, Net4Disconnect, USBHDDConnect, USBHDDDisconnect,
DSPVASystemStart, DSPVASystemFault, DSPVAAMDLoadFail, DSPVAAMDStart,
DSPVAAMDReset, DSPDisplayStart, DSPDisplayFail, BatteryFailRecover,
NetCamConnect, NetCamDisconnect, NetProfileReplace, NetProfileRestore,
VideoLossRecordProfileReplace, VideoLossRecordProfileRestore, RecordStart,
RecordEnd, RAIDEnable, RAIDDisable, RAIDSetup, RAIDBuildStart, RAIDBuildEnd,
RAIDBuildCancel, RAIDBuildFail, RAIDDegrade, RAIDRebuildStart,
RAIDRebuildEnd, RAIDRebuildFail, RAIDFail, InternalHDDConnect,
InternalHDDDisconnect, InternalHDDWarmup, DatabaseFull, DatabaseRemove,
USBWIFIConnect, USBWIFIDisconnect, GSensorEvent, GPSDisconnect, HWSelfTest,
SWSelfTest, FanStopped, GPSInfo, EmergencyTrigger, MQTTConnection
```

5.4. Examples

5.4.1. Getting all system logs

Using the **view** action without specifying the **type** parameter returns logs of all types.

REQUEST

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

TEXT RESPONSE

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

```
Total=1000
[2015-06-30 13:18:29] [Network] Physical network is connected
[2015-06-30 13:18:21] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-30 13:18:14] [PowerOn] Camera System Power On
[2015-06-26 17:01:33] [Network] Physical network is connected
[2015-06-26 17:01:25] [Network] System gets an IPv4 address: 192.168.10.54
[2015-06-26 17:01:18] [PowerOn] Camera System Power On
[2015-06-26 17:00:21] [FWUpdate] System Firmware Update Complete
[2015-06-26 16:36:34] [Network] Physical network is connected
[2015-06-26 16:36:26] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-26 16:36:19] [PowerOn] Camera System Power On
[2015-06-26 16:35:26] [FWUpdate] System Firmware Update Complete
[2015-06-26 12:42:59] [Network] Physical network is connected
[2015-06-26 12:42:51] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-26 12:42:44] [PowerOn] Camera System Power On
[2015-06-25 14:41:41] [ConfigChange] Profile 5 RTP Multicast Port: 0 =>
47806
[2015-06-25 14:41:41] [ConfigChange] Profile 5 RTP Multicast IPv4 Address:
=> 224.16.17.52
[2015-06-25 14:41:41] [ConfigChange] Profile 5 RTP Multicast: off => on
[2015-06-25 14:41:41] [FanStopped] [Right] Fan failure
[2015-06-25 14:41:42] [FanStopped] [Left] Fan failure
...
```

JSON RESPONSE

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

```
{
    "SystemLog": [
        {
            "Date": "2015-06-30 13:18:29",
```

```
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-30 13:18:21",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-30 13:18:14",
            "Type": "PowerOn",
            "Description": "Camera System Power On"
        },
        {
            "Date": "2015-06-26 17:01:33",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-26 17:01:25",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.10.54"
        },
        {
            "Date": "2015-06-26 17:01:18",
            "Type": "PowerOn",
            "Description": "Camera System Power On"
        },
        {
            "Date": "2015-06-26 17:00:21",
            "Type": "FWUpdate",
            "Description": "System Firmware Update Complete"
        },
        {
            "Date": "2015-06-26 16:36:34",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-26 16:36:26",
            "Type": "Network",
```

```
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-26 16:36:19",
            "Type": "PowerOn",
            "Description": "Camera System Power On"
        },
        {
            "Date": "2015-06-26 16:35:26",
            "Type": "FWUpdate",
            "Description": "System Firmware Update Complete"
        },
        {
            "Date": "2015-06-26 12:42:59",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-26 12:42:51",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-26 12:42:44",
            "Type": "PowerOn",
            "Description": "Camera System Power On"
        },
        {
            "Date": "2015-06-25 14:41:41",
            "Type": "ConfigChange",
            "Description": "Profile 5 RTP Multicast Port: 0 => 47806"
        },
        {
            "Date": "2015-06-25 14:41:41",
            "Type": "ConfigChange",
            "Description": "Profile 5 RTP Multicast IPv4 Address: =>
224.16.17.52"
        },
        {
            "Date": "2015-06-25 14:41:41",
            "Type": "ConfigChange",
```

```
            "Description": "Profile 5 RTP Multicast: off => on"
        },
        {
            "Date": "2015-06-25 14:41:41",
            "Type": "FanStopped",
            "Description": "[Right] Fan failure"
        },
        {
            "Date": "2015-06-25 14:41:42",
            "Type": "FanStopped",
            "Description": "[Left] Fan failure"
        }
        ...
    ]
}
```

5.4.2. Getting the time change log and the network log

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=systemlog&action=view&Type=TimeChange,Network
```

TEXT RESPONSE

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

```
Total=85
[2015-06-30 13:18:29] [Network] Physical network is connected
[2015-06-30 13:18:21] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-26 17:01:33] [Network] Physical network is connected
[2015-06-26 17:01:25] [Network] System gets an IPv4 address: 192.168.10.54
[2015-06-26 16:36:34] [Network] Physical network is connected
[2015-06-26 16:36:26] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-26 12:42:59] [Network] Physical network is connected
[2015-06-26 12:42:51] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-25 03:32:40] [Network] Physical network is connected
[2015-06-25 03:32:31] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-23 16:23:27] [TimeChange] Time Change: 2015-03-23 16:23:18 => 2015-
```

```
06-23 16:23:27
[2015-03-23 16:19:04] [TimeChange] Time Change: 2015-06-23 16:19:11 => 2015-
03-23 16:19:04
[2015-06-22 16:21:59] [Network] Physical network is connected
[2015-06-22 16:21:45] [Network] System gets an IPv4 address: 192.168.75.51
[2015-06-22 16:09:04] [Network] Physical network is connected
[2015-06-22 16:08:50] [Network] System gets an IPv4 address: 192.168.75.51
...
```

JSON RESPONSE

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

```
{
    "SystemLog": [
        {
            "Date": "2015-06-30 13:18:29",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-30 13:18:21",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-26 17:01:33",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-26 17:01:25",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.10.54"
        },
        {
            "Date": "2015-06-26 16:36:34",
            "Type": "Network",
            "Description": "Physical network is connected"
```

```
        },
        {
            "Date": "2015-06-26 16:36:26",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-26 12:42:59",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-26 12:42:51",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-25 03:32:40",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-25 03:32:31",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-23 16:23:27",
            "Type": "TimeChange",
            "Description": "Time Change: 2015-03-23 16:23:18 => 2015-06-23
16:23:27"
        },
        {
            "Date": "2015-03-23 16:19:04",
            "Type": "TimeChange",
            "Description": "Time Change: 2015-06-23 16:19:11 => 2015-03-23
16:19:04"
        },
        {
            "Date": "2015-06-22 16:21:59",
            "Type": "Network",
```

```
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-22 16:21:45",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        },
        {
            "Date": "2015-06-22 16:09:04",
            "Type": "Network",
            "Description": "Physical network is connected"
        },
        {
            "Date": "2015-06-22 16:08:50",
            "Type": "Network",
            "Description": "System gets an IPv4 address: 192.168.75.51"
        }
    ]
}
```


5.4.3. Getting the time change log

This example reads time changes between January 1, 2015 and June 30, 2015

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=systemlog&action=view&Type=TimeChange&FromDate=2015-
01-01&ToDate=2015-06-30
```

TEXT RESPONSE

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

```
Total=2
[2015-06-23 16:23:27] [TimeChange] Time Change: 2015-03-23 16:23:18 => 2015-
06-23 16:23:27
[2015-03-23 16:19:04] [TimeChange] Time Change: 2015-06-23 16:19:11 => 2015-
03-23 16:19:04
```

JSON RESPONSE

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

```
{
    "SystemLog": [
        {
            "Date": "2015-06-23 16:23:27",
            "Type": "TimeChange",
            "Description": "Time Change: 2015-03-23 16:23:18 => 2015-06-23
16:23:27"
        },
        {
            "Date": "2015-03-23 16:19:04",
            "Type": "TimeChange",
            "Description": "Time Change: 2015-06-23 16:19:11 => 2015-03-23
16:19:04"
        }
    ]
}
```

Chapter 6. Access Log

6.1. Description

The **accesslog** submenu reads the HTTP client access log.

Access level

Action	Camera	Encoder	NVR
view	Admin	Admin	User

6.2. Syntax

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

6.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Type	REQ, RES	<csv> AdminLogin, AdminLogout, UserLogin, UserLogout, GuestLogin, GuestLogin, GuestLogout, Login, SesstionTimeout	Log type If **Type** is not sent, the response will contain all user-level data.
	FromDate	REQ	<string>	Search start date The date is specified in the format of <YYYY-MM-DD>.
	ToDate	REQ	<string>	Search end date The date is specified in the format of <YYYY-MM-DD>.

6.4. Examples

6.4.1. Getting all access logs

Using the **view** action without the **Type** parameter returns all response parameters and values.

REQUEST

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

TEXT RESPONSE

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

```
Total=402
[2015-06-30 13:30:49] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-30 13:22:19] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 17:37:06] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 17:19:13] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 17:19:04] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 17:18:23] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 17:18:19] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 17:02:53] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 16:53:44] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 16:43:59] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 16:43:57] [AdminLogin] Admin Log In Success: 192.168.75.137
...
```

JSON RESPONSE

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

```
{
    "AccessLog": [
        {
            "Date": "2015-06-30 13:30:49",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-30 13:22:19",
            "Type": "AdminLogin",
```

```
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:37:06",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:19:13",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:19:04",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:18:23",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:18:19",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:02:53",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:53:44",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:43:59",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
```

```
        },
        {
            "Date": "2015-06-26 16:43:57",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        }
        ...
    ]
}
```

6.4.2. Getting the log data on admin logins and logouts

This example reads the log information on admin logins and logouts between June 1, 2015 and June 30, 2015.

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=accesslog&action=view&Type=AdminLogin,AdminLogout&Fr
omDate=2015-06-01&ToDate=2015-06-30
```

TEXT RESPONSE

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

```
Total=402
[2015-06-30 13:30:49] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-30 13:22:19] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 17:37:06] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 17:19:13] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 17:19:04] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 17:18:23] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 17:18:19] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 17:02:53] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 16:53:44] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 16:43:59] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 16:43:57] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 16:28:52] [AdminLogin] Admin Log In Success: 192.168.75.137
[2015-06-26 16:28:35] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 16:28:19] [AdminLogin] Admin Log In Success: 192.168.75.137
```

```
[2015-06-26 16:26:16] [AdminLogout] Admin Log Out: 192.168.75.137
[2015-06-26 16:26:06] [AdminLogin] Admin Log In Success: 192.168.75.137
...
```

JSON RESPONSE

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


```
{
    "AccessLog": [
        {
            "Date": "2015-06-30 13:30:49",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-30 13:22:19",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:37:06",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:19:13",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:19:04",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:18:23",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
```

```
        },
        {
            "Date": "2015-06-26 17:18:19",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 17:02:53",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:53:44",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:43:59",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:43:57",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:28:52",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:28:35",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:28:19",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        },
```

```
        {
            "Date": "2015-06-26 16:26:16",
            "Type": "AdminLogout",
            "Description": "Admin Log Out: 192.168.75.137"
        },
        {
            "Date": "2015-06-26 16:26:06",
            "Type": "AdminLogin",
            "Description": "Admin Log In Success: 192.168.75.137"
        }
        ...
    ]
}
```

Chapter 7. Event Log

7.1. Description

The **eventlog** submenu reads the event log. Each line of an event log response consists of the following:

- Date
- Time
- Type
- Description

Access level

Action	Camera	Encoder	NVR
view	Admin	Admin	User

7.2. Syntax

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=eventlog&action=view&Type=<value>
```

7.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Type	REQ, RES	<csv> Refer to **EventLog Types** block below.	Event Type **Note** > The actual event logs may differ from model to model, depending on the functions supported by the product. For example, if a model does not support Face Detection, no "Face Detection" log will be available. Please check the events the device supports using attributes.cgi.
	FromDate	REQ	<string>	Search start date The date is specified in the format of <YYYY-MM-DD>.

Action	Parameters	Request/ Response	Type/ Value	Description
	ToDate	REQ	<string>	Search end date The date is specified in the format of <YYYY-MM-DD>.
check	Type	RES	<csv> Refer to **EventLog Types** block below.	Event type

EventLog Types

```
VideoAnalysis, Passing, Entering, Exiting, Appearing, Disappearing,
ScheduledEvent, MotionDetection, NetworkDisconnect, FaceDetection,
TamperingDetection, AlarmOutput, AlarmInput, AudioDetection, GotoPreset,
Aux, Videoloss, Tracking, OpenSDK, PTZMotion, UserInput, AlarmDetect,
SceneChange, CameraAlarm, CameraAppearDisappear, VideoLossRelease, AMDStart,
AMDStop, TamperMask, TamperRotate, TamperImage, LowFpsStart, LowFpsEnd,
DefocusDetection, TrackingStart, TrackingEnd, FogDetection, AudioAnalysis,
QueueEvent, ShockDetection, TamperatureChangeDetection,
BoxTemperatureDetection,HousingTampering, WaterLevelWarning,
ObjectDetection, BodyTemperatureDetection, MaskDetection,
SocialDistancingViolation, CallRequest, TamperingSwitch, DTMFReceived,
ProxmimitySensor ParkingDetection, DynamicRule, MQTTSubscription, LEDPreset
```

7.4. Examples

7.4.1. Getting all event logs

Using the **view** action without specifying the **Type** parameter returns all data.

REQUEST

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

TEXT RESPONSE

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

```
Total=598
```

```
[2015-06-25 14:34:29] [PTZMotion] PTZ STOP : 192.168.75.130
[2015-06-25 14:34:27] [PTZMotion] PTZ UI ZOOM OUT Click : 192.168.75.130
[2015-06-25 14:34:25] [PTZMotion] PTZ STOP : 192.168.75.130
...
[2015-06-17 02:08:31] [TamperingDetection] Tampering Event Detected
[2015-06-17 02:08:13] [TamperingDetection] Tampering Event Detected
[2015-06-17 02:07:18] [TamperingDetection] Tampering Event Detected
[2015-06-17 02:06:54] [TamperingDetection] Tampering Event Detected
...
[2015-06-12 10:08:01] [FaceDetection] Face Detection Start
[2015-06-12 10:07:59] [FaceDetection] Face Detection End
[2015-06-12 10:07:55] [FaceDetection] Face Detection Start
[2015-06-12 10:06:57] [FaceDetection] Face Detection End
```

JSON RESPONSE

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

```
{
    "ChannelEventLog": [
        {
            "Channel": 0,
            "EventLog": [
                {
                    "Date": "2015-06-25 14:34:29",
                    "Type": "PTZMotion",
                    "Description": "PTZ STOP : 192.168.75.130"
                },
                {
                    "Date": "2015-06-25 14:34:27",
                    "Type": "PTZMotion",
                    "Description": "PTZ UI ZOOM OUT Click : 192.168.75.130"
                },
                {
                    "Date": "2015-06-25 14:34:25",
                    "Type": "PTZMotion",
                    "Description": "PTZ STOP : 192.168.75.130"
                },
                {
```

```
                "Date": "2015-06-17 02:08:31",
                "Type": "TamperingDetection",
                "Description": "Tampering Event Detected"
            },
            {
                "Date": "2015-06-17 02:08:13",
                "Type": "TamperingDetection",
                "Description": "Tampering Event Detected"
            },
            {
                "Date": "2015-06-17 02:07:18",
                "Type": "TamperingDetection",
                "Description": "Tampering Event Detected"
            },
            {
                "Date": "2015-06-17 02:06:54",
                "Type": "TamperingDetection",
                "Description": "Tampering Event Detected"
            },
            {
                "Date": "2015-06-12 10:08:01",
                "Type": "FaceDetection",
                "Description": "Face Detection Start"
            },
            {
                "Date": "2015-06-12 10:07:59",
                "Type": "FaceDetection",
                "Description": "Face Detection End"
            },
            {
                "Date": "2015-06-12 10:07:55",
                "Type": "FaceDetection",
                "Description": "Face Detection Start"
            },
            {
                "Date": "2015-06-12 10:06:57",
                "Type": "FaceDetection",
                "Description": "Face Detection End"
            }
        ]
    }
```

```
    ]
}
```

7.4.2. Getting motion detection and face detection events

This example reads the motion detection and face detection event logs from June 1, 2015 to July 30, 2015.

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=eventlog&action=view&Type=MotionDetection,FaceDetect
ion&FromDate=2015-06-01&ToDate=2015-07-30
```

TEXT RESPONSE

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

```
Total=6
[2015-06-12 10:08:01] [FaceDetection] Face Detection Start
[2015-06-12 10:07:59] [FaceDetection] Face Detection End
[2015-06-12 10:07:55] [FaceDetection] Face Detection Start
[2015-06-12 10:06:57] [FaceDetection] Face Detection End
[2015-07-06 19:14:10] [MotionDetection] Motion Detection Start
[2015-07-06 19:14:13] [MotionDetection] Motion Detection End
```

JSON RESPONSE

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

```
{
    "ChannelEventLog": [
        {
            "Channel": 0,
            "EventLog": [
                {
                    "Date": "2015-06-12 10:08:01",
                    "Type": "FaceDetection",
```

```
                    "Description": "Face Detection Start"
                },
                {
                    "Date": "2015-06-12 10:07:59",
                    "Type": "FaceDetection",
                    "Description": "Face Detection End"
                },
                {
                    "Date": "2015-06-12 10:07:55",
                    "Type": "FaceDetection",
                    "Description": "Face Detection Start"
                },
                {
                    "Date": "2015-06-12 10:06:57",
                    "Type": "FaceDetection",
                    "Description": "Face Detection End"
                },
                {
                    "Date": "2015-07-06 19:14:10",
                    "Type": "MotionDetection",
                    "Description": "Motion Detection Start"
                },
                {
                    "Date": "2015-07-06 19:14:13",
                    "Type": "MotionDetection",
                    "Description": "Motion Detection End"
                }
            ]
        }
    ]
}
```

Chapter 8. Profile Access Information

8.1. Description

The **profileaccessinfo** submenu requests information on the profile currently being used and on the users accessing the current profile.

> **NOTE** This chapter is for network cameras only. Attribute to check for max channels: "attributes/System/Limit/**MaxChannel**"

Access level

Action	Camera	Encoder
view	Admin	Admin

8.2. Syntax

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

8.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	ViewGroup	REQ	<enum> Profile, User, All	Type of target information. • Profile: Profile info • User: User info • All: Both profile and user info
	Channel.#.Profile.#.CurrentBitrate	RES	<int>	The profile's current bit rate(in kbps) This parameter is valid only when **ViewGroup** is set to Profile or All.
	Channel.#.Profile.#.TotalBitrate	RES	<int>	The profile's total bit rate(in kbps) This parameter is valid only when **ViewGroup** is set to Profile or All.
	Channel.#.Profile.#.CurrentFPS	RES	<int>	The profile's current frame rate This parameter is valid only when **ViewGroup** is set to Profile or All.

Action	Parameters	Request/ Response	Type/ Value	Description
	Channel.#.Profile.#.TotalFPS	RES	<int>	The profile's total frame rate This parameter is valid only when **ViewGroup** is set to Profile or All.
	Channel.#.Profile.#.ATC	RES	<int>	The profile's ATC This parameter is valid only when **ViewGroup** is set to Profile or All.
	Channel.#.Profile.#.ConcurrentUserCount	RES	<int>	The number of users using the profile This parameter is valid only when **ViewGroup** is set to Profile or All.
	User.#.ProfileNameList	RES	<csv>	The profile name list This parameter is valid only when **ViewGroup** is set to User or All.
	User.#.ClientIPAddress	RES	<string>	The user's IP address. This parameter is valid only when **ViewGroup** is set to User or All.
	User.#.CurrentBitrate	RES	<int>	Bit rate transferred to the user This parameter is valid only when **ViewGroup** is set to User or All.
	User.#.ClientNetworkConnectionStatus	RES	<enum> Good, Bad, Optimized	User and camera's connection status This parameter is valid only when **ViewGroup** is set to User or All.
	User.#.Channel.#.ProfileNameList	RES	<csv>	Channel-based profile name list (Applicable for Multi directional camera)
	User.#.Channel.#.CurrentBitrate	RES	<int>	Channel-based current bitrate (Applicable for Multi directional camera)

NOTE # represents the index number of the channel, profile or user.

8.4. Examples

8.4.1. Getting the profile access information

Specifying the **ViewGroup** as Profile returns profile access information only and does not return user access information.

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=profileaccessinfo&action=view&ViewGroup=Profile
```

TEXT RESPONSE

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

```
Channel.0.Profile.0.CurrentBitrate=0
Channel.0.Profile.0.TotalBitrate=6144
Channel.0.Profile.0.CurrentFPS=0
Channel.0.Profile.0.TotalFPS=5
Channel.0.Profile.0.ATC=0
Channel.0.Profile.0.ConcurrentUserCount=0
Channel.0.Profile.1.CurrentBitrate=0
Channel.0.Profile.1.TotalBitrate=512
Channel.0.Profile.1.CurrentFPS=0
Channel.0.Profile.1.TotalFPS=30
Channel.0.Profile.1.ATC=0
Channel.0.Profile.1.ConcurrentUserCount=0
Channel.0.Profile.9.CurrentBitrate=0
Channel.0.Profile.9.TotalBitrate=300
Channel.0.Profile.9.CurrentFPS=0
Channel.0.Profile.9.TotalFPS=3
Channel.0.Profile.9.ATC=0
Channel.0.Profile.9.ConcurrentUserCount=0
```

JSON RESPONSE

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

```
{
    "ProfileAccessInfo": {
        "ProfileInfo": [
            {
                "Channel": 0,
                "Profiles": [
                    {
                        "Profile": 0,
                        "CurrentBitrate": 0,
                        "TotalBitrate": 6144,
                        "CurrentFPS": 0,
                        "TotalFPS": 5,
                        "ATC": 0,
                        "ConcurrentUserCount": 0
                    },
                    {
                        "Profile": 1,
                        "CurrentBitrate": 0,
                        "TotalBitrate": 512,
                        "CurrentFPS": 0,
                        "TotalFPS": 30,
                        "ATC": 0,
                        "ConcurrentUserCount": 0
                    },
                    {
                        "Profile": 9,
                        "CurrentBitrate": 0,
                        "TotalBitrate": 300,
                        "CurrentFPS": 0,
                        "TotalFPS": 3,
                        "ATC": 0,
                        "ConcurrentUserCount": 0
                    }
                ]
            }
        ]
    }
}
```

8.4.2. Getting the user information

This example reads profile information which includes the number of users accessing simultaneously.

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=profileaccessinfo&action=view&ViewGroup=User
```

TEXT RESPONSE

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

```
User.1.ProfileNameList=H.264
User.1.ClientIPAddress=192.168.75.137
User.1.ClientBitrate=195
User.1.ClientNetworkConnectionStatus=Good
```

JSON RESPONSE

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

```
{
    "ProfileAccessInfo": {
        "Users": [
            {
                "User": 1,
                "ProfileNameList": [
                    "H.264"
                ],
                "ClientIPAddress": "192.168.75.137",
                "ClientBitrate": 195,
                "ClientNetworkConnectionStatus": "Good"
            }
        ]
    }
```

```
}
```

Chapter 9. Factory Reset

9.1. Description

The **factoryreset** submenu resets the system to its default settings

The device can restart after a factory reset.

Access level

Action	Camera	Encoder	NVR
control	Admin	Admin	Admin

9.2. Syntax

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=factoryreset&action=control&ExcludeSettings=<value>
```

9.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
control	ExcludeSettings	REQ	<csv> Authority, Network, Camera, None	Selects a group of settings to be excluded from being set to defaults during a factory reset • Authority: Resets the system to default settings except for the user permission configuration. • Network: Resets the system to default settings except for the network configuration. • Camera: Resets the system to default settings except for the camera mapping information.

9.4. Examples

9.4.1. Resetting the system except for the network configuration

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=
```

```
factoryreset&action=control&ExcludeSettings=Network
```

Chapter 10. Reset System Power

10.1. Description

The **power** submenu resets the system power.

Access level

Action	Camera	Encoder	NVR
view	N/A	N/A	User
set	N/A	N/A	User
control	Admin	Admin	User

10.2. Syntax

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=power&action=control&Type=<value>
```

10.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the power reset settings
set	DelayBooting	REQ, RES	<enum> Off, 5s, 10s, 15s, 30s, 1m, 2m, 3m, 5m, 10m	Based on this setting device, booting time after restart can be controlled(for NVR only)
	EnableAlarmOutput	REQ, RES	<bool> True, False	Option to enable/disable alarmout (for NVR only)
	ShutdownTime	REQ, RES	<int>	Based on this setting, shutdown time can be controlled in seconds (for NVR only)
control	Type	REQ	<enum> Restart, Shutdown	Reset type • Restart: Restarts the system.Shutdown: Shutdown the system (for NVR only). **Note** > **Type** must be sent together with the **control** action.

10.4. Examples

10.4.1. Restarting the system

To restart the system with the **control** action, the **Type** parameter must be set.

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=power&action=control&Type=Restart
```

Chapter 11. Firmware Update

11.1. Description

The **firmwareupdate** submenu updates the device firmware.

The device can restart after a firmware update.

Access level

Action	Camera	Encoder	NVR
control	Admin	Admin	Admin
check			Admin
set			Admin
view			Admin

NOTE Attribute to check for Firmware Update: "attributes/System/Support/**FWUpdate**"

11.2. Syntax

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

11.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
check	FWVersionAvailableIn Server	RES	<string>	Firmware version available in server. NVR ONLY
control	DownloadFromServer	REQ	<bool> True, False	If true, the firmware is downloaded from the server. NVR ONLY

Action	Parameters	Request/ Response	Type/ Value	Description
	Type	REQ	<enum> Normal, FactoryDefault	Firmware update type • Normal: Only updates the firmware. • FactoryDefault: Updates the firmware and resets the system to factory defaults. **Note** **Type** must be sent together with the **control** action.
	IgnoreMultipartResponse	REQ	<bool>	If true, intermediate events will not be sent from the device. This is used to ignore multipart messages
	Status	RES	<enum> DownloadAck, DownloadOK, DownloadFail, Start, End, OK, Skip, Fail, UpdatingISP, Alive	Status of firmware update • DownloadAck: Successfully received the firmware data • DownloadOK: Finished receiving firmware data • DownloadFail: Failed to download the firmware data • Start: Starting firmware update • End: Completed firmware update • OK: Completed each firmware module update • Skip: Skips update for the firmware module using the latest version • Fail: Failed to update firmware • Alive: Reports periodically that the device is alive.
	FirmwareModule	RES	<enum> None, Kernel, App, Web, ISP	Firmware module to update CAMERA ONLY ENCODER ONLY

Action	Parameters	Request/ Response	Type/ Value	Description
	Progress	RES	<int>	Progress of firmware update The value must be in the range of 0 to 100; it indicates the progress in percent. 0 means that 0% of update has been completed and 100 means 100% has been completed.
set	OnlineUpgrade	REQ	<bool> True, False	To enable online upgrade of NVR NVR ONLY
view	OnlineUpgrade	RES	<bool> True, False	To check the current online upgrade settings NVR ONLY

11.4. Examples

11.4.1. Normal type firmware updates

REQUEST

The file content is provided in the HTTP body as below according to the format given in RFC 1867.

```
POST /stw-cgi/system.cgi?msubmenu=firmwareupdate&action=control&Type=Normal
HTTP/1.1

Content-Length: <content length>
Content-Type: multipart/form-data; boundary=<boundary>

--<boundary>
Content-Disposition: form-data; name="UploadedFile";
filename="<file name>"
Content-Type: application/octet-stream

<firmware file content>

--<boundary>--
```

TEXT RESPONSE

The status of firmware update and the progress in percent.

```
HTTP/1.1 200 OK
```

```
Content-Type: multipart/x-mixed-replace; boundary=SamsungTechwin
--SamsungTechwin
Content-Type: text/plain
--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=0%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=4%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=8%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=13%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=17%
```

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

Status=DownloadAck
FirmwareModule=None
Progress=21%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=26%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=30%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=34%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=39%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=43%
```

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

Status=DownloadAck
FirmwareModule=None
Progress=47%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=52%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=56%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=60%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=65%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
```

```
Progress=69%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=73%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=78%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=82%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=86%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=91%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
```

```
FirmwareModule=None
Progress=95%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=99%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=100%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadOK
FirmwareModule=None
Progress=100%

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=Alive

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

```
Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=OK
FirmwareModule=Kernel
Progress=100%

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=OK
FirmwareModule=App
Progress=100%

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=OK
FirmwareModule=Web
Progress=100%

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

```
Status=End
FirmwareModule=None
Progress=100%

--SamsungTechwin
Content-type:text/plain

OK
```

JSON RESPONSE

Status of the firmware update, and the progress as a percentage.

```
HTTP/1.1 200 OK
```

```
Content-Type: multipart/x-mixed-replace; boundary=SamsungTechwin
--SamsungTechwin
Content-Type: application/json

--SamsungTechwin
Content-Type: application/json

{
    "Status": "DownloadAck"
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 4
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 8
}
--SamsungTechwin
Content-Type: application/json
```

```
{
    "Progress": 13
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 17
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 21
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 26
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 30
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 34
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 39
}
--SamsungTechwin
Content-Type: application/json
```

```
{
    "Progress": 43
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 47
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 52
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 56
}

--SamsungTechwin
Content-Type: application/json

{
    "Progress": 60
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 65
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 69
}
```

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

{
    "Progress": 73
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 78
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 82
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 86
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 91
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 95
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 99
```

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

{
    "Progress": 100
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "DownloadOK"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Start"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
```

```
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "OK",
    "FirmwareModule": "Kernel",
    "Progress": 100
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "OK",
    "FirmwareModule": "App",
    "Progress": 100
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "OK",
```

```
    "FirmwareModule": "Web",
    "Progress": 100
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "End"
}
--SamsungTechwin
Content-type:application/json

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

CURL command

A firmware update can be tested with CURL as below. To learn about CURL, please refer to http://curl.haxx.se.

NOTE For getting JSON response, add the -H ted with CURL as below. To header to the request.

```
curl --digest -u <userid>:<password> -F
UploadedFile=@snb5004_Series_1.13_131218.img "http://<Device IP>/stw-
cgi/system.cgi?msubmenu=firmwareupdate&action=control&Type=Normal" -H
"Expect:"
```

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

```
POST /stw-cgi/system.cgi?msubmenu=firmwareupdate&action=control&Type=Normal
HTTP/1.1
Content-Length: 27920272
Content-Type: multipart/form-data; boundary=<boundary>
```

```
--<boundary>
Content-Disposition: form-data; name="UploadedFile";
filename="snb5004_Series_1.13_131218.img"
Content-Type: application/octet-stream
```

```
<firmware file content>

--<boundary>--
```

TEXT RESPONSE

```
HTTP/1.1 200 OK
Content-Type: multipart/x-mixed-replace; boundary=SamsungTechwin

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=0%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=4%

--SamsungTechwin
Content-Type: text/plain

Status=DownloadAck
FirmwareModule=None
Progress=8%

--SamsungTechwin
Content-Type: text/plain

...

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
```

```
Content-Type: text/plain

Status=OK
FirmwareModule=App
Progress=100%

--SamsungTechwin
Content-Type: text/plain

Status=Alive

--SamsungTechwin
Content-Type: text/plain

Status=OK
FirmwareModule=Web
Progress=100%

--SamsungTechwin
Content-Type: text/plain

Status=End
FirmwareModule=None
Progress=100%

--SamsungTechwin
Content-type:text/plain

OK
```

JSON RESPONSE

```
HTTP/1.1 200 OK
Content-Type: multipart/x-mixed-replace; boundary=SamsungTechwin
```

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

{
    "Status": "DownloadAck"
}
```

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

{
    "Progress": 4
}
--SamsungTechwin
Content-Type: application/json

{
    "Progress": 8
}
...

--SamsungTechwin
Content-Type: application/json

{
    "Status": "Alive"
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "OK",
    "FirmwareModule": "Web",
    "Progress": 100
}
--SamsungTechwin
Content-Type: application/json

{
    "Status": "End"
}
--SamsungTechwin
Content-type:application/json

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

Chapter 12. Configuration Backup

12.1. Description

The **configbackup** submenu makes a copy of all system settings for backup.

The format of the configuration is device-dependent.

The device can restart after making a copy of the system settings for backup.

Access level

Action	Camera	Encoder	NVR
control	Admin	Admin	User

NOTE Attribute to check for Configuration Backup: "attributes/System/Support/**ConfigBackup**"

12.2. Syntax

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=configbackup&action=control
```

CURL command The configuration backup can be tested with CURL as below. To learn about CURL, please refer to http://curl.haxx.se.

```
curl -v --digest -u admin:isv13579! "http://<Device IP>/stw-
cgi/system.cgi?msubmenu=configbackup&action=control" > config.bin
```

Chapter 13. Configuration Restore

13.1. Description

The **configrestore** submenu restores the system configuration by using the backup.

You can reset multiple cameras to the same configuration by using **configbackup** and **configrestore**.

The format of the configuration is device-dependent.

The device can restart after a configuration restore.

Access level

Action	Camera	Encoder	NVR
control	Admin	Admin	Admin

NOTE Attribute to check for Configuration Restore: "attributes/System/Support/**ConfigRestore**"

13.2. Syntax

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

13.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
control	ExcludeSettings	REQ	<csv> Authority, Network, Camera, None	Selects a group of settings to be excluded from system configuration restore • Authority: Restores the system configuration except for the user permission settings. • Network: Restores the system configuration except for the network settings. • Camera: Restores the system configuration except for the camera mapping information assigned to each channel. • None: Restores all settings.

13.4. Examples

13.4.1. Restoring the system configuration except for current network settings

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=
configrestore&action=control&ExcludeSettings=Network
```

File content

The file content is provided in the HTTP body as below in URL-encoded format after being encoded in base64 format.

```
<Body>
```

```
POST /stw-
cgi/system.cgi?msubmenu=configrestore&action=control&ExcludeSettings=Network
HTTP/1.1
Content-type: application/x-www-form-urlencoded; charset=utf-8
Content-Length: <content length>

<config file content>
```

CURL command

The configuration backup and configuration restore can be tested with CURL as below. To learn about CURL, please refer to http://curl.haxx.se.

CURL command for configuration backup looks like below;

```
curl -v --digest -u admin:isv13579! "http://<Device IP>/stw-
cgi/system.cgi?msubmenu=configbackup&action=control" > config.bin
```

The configuration file needs to be base64 encoded before sending the POST request.

```
openssl base64 -in config.bin -out encoded.bin
```

The CURL command for configuration restore is as below;

```
curl -v --digest -u <userid>:<password> --data-urlencode @encoded.bin
"http://<Device IP>/stw-
cgi/system.cgi?msubmenu=configrestore&action=control&ExcludeSettings=Network
" -H "Expect:"
```

Chapter 14. Storage Information

14.1. Description

The **storageinfo** submenu requests storage device information.

> **NOTE** Attributes to check if the device supports DAS encryption: **"attributes/System/Support/SDCardEncryption"**

Access level

Action	Camera	NVR
view	Admin	User
set	Admin	User
control	Admin	User

14.2. Syntax

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

14.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the storage information settings.
	SlotNumber	REQ, RES	<int>	Slot number NVR ONLY
	Storage.#.Usage	RES	<enum> Internal, External	Usage of corresponding storage NVR ONLY
	Storage.#.Model	RES	<string>	Model of corresponding storage NVR ONLY
	UsedSpace	RES	<string>	Amount of storage device space currently in use (in megabytes) UsedSpace means the sum of the used space for all storage devices; Sum of Storage.1.UsedSpace, Storage.2.UsedSPace, and so forth.

Action	Parameters	Request/ Response	Type/ Value	Description
	TotalSpace	RES	<string>	Total storage device space (in megabytes) TotalSpace means the sum of total space of all storage devices; Sum of Storage.1.TotalSpace, Storage.2.TotalSPace, and so forth.
	Storage.#.Temperature	RES	<string>	Temperature of storage NVR ONLY
	Storage.#.UsedSpace	RES	<string>	Amount of storage device space currently in use (in megabytes) in corresponding storage
	Storage.#.TotalSpace	RES	<string>	Total storage device space (in megabytes) in corresponding storage
	Storage.#.Type	RES	<enum> DAS, NAS	Storage type of the corresponding storage **Note** Attribute to check for NAS Support: "attributes/Recording/Support/NAS"
	Storage.#.FileSystem	RES	<enum>	File system of the corresponding storage This parameter is valid only when **Storage.#.Type** is set to DAS. CAMERA ONLY
	Storage.#.Status	RES	<enum> Normal, Error, Active, Formatting, Lock, Check, Error. InvalidSlot, Error. UnknownRAID, Full, PWError	Status of the corresponding storage CAMERA ONLY
	Storage	REQ	<int>	Storage number
	IsSDCardEncrypted	RES	<bool> True, False	Status of storage encryption

Action	Parameters	Request/ Response	Type/ Value	Description
set	Storage	REQ	<int>	Storage number (read-only for NVR) **Note** **Storage** and **Enable** must be sent together for the **set** action.
	Enable	REQ, RES	<bool> True, False	Enables or disables the storage **Note** **Storage** and **Enable** must be sent together for the **set** action. CAMERA ONLY
	DefaultFolder	REQ, RES	<string>	Path on the NAS for recording This parameter is valid only when Type is set to NAS. CAMERA ONLY
	NASIP	REQ, RES	<string>	NAS IP address This parameter is valid only when Type is set to NAS. CAMERA ONLY
	NASUserID	REQ, RES	<string>	NAS user ID This parameter is valid only when Type is set to NAS. CAMERA ONLY
	NASPassword	REQ, RES	<string>	NAS password This parameter is valid only when **Type** is set to NAS. CAMERA ONLY

Action	Parameters	Request/ Response	Type/ Value	Description
	IsNASPasswordEncrypted	REQ	<bool> True, False	When this is set as true, the password is encrypted using the public key obtained using the **rsa** submenu of security.cgi, and sent as payload content for the POST command. CAMERA ONLY
	FileSystem	REQ, RES	<enum> VFAT, ext4	File system This parameter is valid only when **Storage.#.Type** is set to DAS. CAMERA ONLY
	TargetIP	REQ, RES	<string>	Target IP address (read-only for NVR)
	IsDASEncryptEnable	REQ, RES	<bool> True, False	Enables or disables disk array storage encryption CAMERA ONLY
	NewDASPassword	REQ, RES	<string>	New DAS password for password change
	IsNewDASPasswordEncrypted	REQ, RES	<bool> True, False	When this is set to true, the new DAS password is encrypted using the public key obtained from the **rsa** submenu of security.cgi, and sent as payload content for the POST command. CAMERA ONLY
	DASPassword	REQ, RES	<string>	DAS password used for encryption CAMERA ONLY
	IsDASPasswordEncrypted	REQ	<bool> True, False	When this is set as true, the DAS password is encrypted using the public key obtained using the **rsa** submenu of security.cgi and sent as payload content for the POST command. CAMERA ONLY
	Port	REQ, RES	<string>	Port number (read-only for NVR)

Action	Parameters	Request/ Response	Type/ Value	Description
	TargetIQN	REQ, RES	<string>	Target IQN (read-only for NVR)
	CHAPUserID	REQ, RES	<string>	CHAP (Challenge Handshake Authentication Protocol) user ID (read-only for NVR)
	CHAPPassword	REQ	<string>	CHAP (Challenge Handshake Authentication Protocol) password (Read-only for NVR)
	IsCHAPPasswordEncrypted	REQ	<bool> True, False	Returns true if password sent is encrypted Encrypted password should be sent as a post message
control	Status	RES	<enum> Fail, Success	Connection status This parameter is valid only when **Mode** is set to NASTest. CAMERA ONLY
	Storage	REQ	<int>	Storage number **Note** To use the **control** action, **Storage** and **Mode** must be sent together.
	Mode	REQ	<enum> Format, NASTest, Connect, Disconnect	Storage mode NASTest is available when **Storage.#.Type** is set to NAS and Storage.# is not in an active state (i.e., already connected and enabled for recording). **Note** To use the **control** action, **Storage** and **Mode** must be sent together.

14.4. Examples

14.4.1. Getting the current storage info when the device supports SD card encryption

REQUEST

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

TEXT RESPONSE

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

```
UsedSpace=105
TotalSpace=30185

Storage.1.Type=DAS
Storage.1.UsedSpace=105
Storage.1.TotalSpace=30185
Storage.1.FileSystem=VFAT
Storage.1.Enable=True
Storage.1.Status=Active
Storage.1.IsDASEncrypteEnable=false
Storage.1.DASPassword=
Storage.1.IsSDCardEncrypted=false

Storage.2.Type=NAS
Storage.2.UsedSpace=0
Storage.2.TotalSpace=0
Storage.2.Enable=False
Storage.2.Status=
Storage.2.DefaultFolder=SNB6004_test
Storage.2.NASIP=192.168.75.180
Storage.2.NASUserID=admin
Storage.2.NASPassword=admin4321
```

JSON RESPONSE

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

```
{
```

```
    "UsedSpace": "105",
    "TotalSpace": "30185",
    "Storages": [
        {
            "Storage": 1,
            "Type": "DAS",
            "UsedSpace": "105",
            "TotalSpace": "30185",
            "FileSystem": "VFAT",
            "Enable": true,
            "Status": "Active",
            "DASConfig": {
                "IsDASEncryptEnable": false,
                "DASPassword": "",
                "IsSDCardEncrypted": false
            }
        },
        {
            "Storage": 2,
            "Type": "NAS",
            "UsedSpace": "0",
            "TotalSpace": "0",
            "Enable": false,
            "Status": "Active",
            "NASConfig": {
                "DefaultFolder": "SNB6004_test",
                "NASIP": "192.168.75.180",
                "NASUserID": "admin",
                "NASPassword": "admin4321"
            }
        }
    ]
}
```


14.4.2. Enabling storage 1

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Storage=1&Enable=True
```

14.4.3. Setting storage mode to NASTest

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=control&Storage=2&Mode=NASTest
```

14.4.4. Initially set new SD card password

REQUEST Get

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Enable=True&Storage=1&IsDASEn
cryptEnable=True&NewDASPassword={Password}
```

REQUEST Post

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Enable=True&Storage=1&IsDASEn
cryptEnable=True&IsNewDASPasswordEncrypted=True
```

```
<SDEncryption>
    <NewDASPassword>{RSA encrypted pw}</NewDASPassword>
</SDEncryption>
```

- {RSA encrypted pw} should be rsa encrypted & base64 encoded.
- whole message should be URL encoded.

14.4.5. Set SD card's password to decrypt SD card(In case of SD Card was encrypted by other camera device, user want to use this sd card in this camera device.)

REQUEST Get

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Enable=True&Storage=1&IsDASEn
cryptEnable=True&DASPassword={Password}
```

REQUEST Post

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Enable=True&Storage=1&IsDASEn
```

```
cryptEnable=True&IsDASPasswordEncrypted=True
```

```
<SDEncryption>
    <DASPassword>{RSA encrypted pw}</DASPassword>
</SDEncryption>
```

- {RSA encrypted pw} should be rsa encrypted & base64 encoded.
- whole message should be URL encoded.

14.4.6. Change SD card password

REQUEST Get

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Enable=True&Storage=1&IsDASEn
cryptEnable=True&DASPassword={Password}&NewDASPassword={Password}
```

REQUEST Post

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=storageinfo&action=set&Enable=True&Storage=1&IsDASEn
cryptEnable=True&IsDASPasswordEncrypted=True&IsNewDASPasswordEncrypted=True
```

```
<SDEncryption>
    <DASPassword>{RSA encrypted pw}</DASPassword>
    <NewDASPassword>{RSA encrypted pw}</NewDASPassword>
</SDEncryption>
```

- {RSA encrypted pw} should be rsa encrypted & base64 encoded.
- whole message should be URL encoded.

Chapter 15. GPS

15.1. Description

The **gps** submenu requests the GPS (global positioning system) information of the NVR.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
view	User

15.2. Syntax

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

15.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Check	REQ	<enum> Once, Periodically	Requests the GPS data only once or periodically
	Periodicity	REQ	<int>	Interval to request the GPS data The range is from 1 to 300.
	GPSData	RES	<string>	GPS data $GPRMC sentence is only available.

15.4. Examples

15.4.1. Getting the GPS data only one time

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=gps&action=view&Check=Once
```

TEXT RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain
<Body>
```

```
GPSData=
$GPRMC,084142.00,A,3729.03548,N,12653.80696,E,11.581,119.28,290610,,,A*51
```

JSON RESPONSE

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

```
{
    "GPSData":
"$GPRMC,084142.00,A,3729.03548,N,12653.80696,E,11.581,119.28,290610,,,A*51"
}
```

15.4.2. Requesting the GPS data every 5 seconds

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=gps&action=view&Check=Periodically&Periodicity=5
```

TEXT RESPONSE

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

```
GPSData=
$GPRMC,102122.00,A,3521.03548,N,12843.80696,E,12.381,118.17,290610,,,A*51
```

JSON RESPONSE

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

```
{
    "GPSData":
"$GPRMC,102122.00,A,3521.03548,N,12843.80696,E,12.381,118.17,290610,,,A*51"
}
```

Chapter 16. Automatic Backup

16.1. Description

The **autobackup** submenu makes a backup of the videos recorded on NVR automatically on the server when the NVR is connected to the server.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
view	User
set	User

16.2. Syntax

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

16.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the automatic backup settings.
	Status	RES	<enum>	Backup status
set	ConnectionType	REQ, RES	<enum> WiFi, Ethernet	Connection type
	ServerSSID	REQ, RES	<string>	Server SSID
	PollingFrequency	REQ, RES	<enum> Off, 5s, 10s, 20s, 30s, 1m, 5m, 10m	Polling frequency interval

16.4. Examples

16.4.1. Getting the current auto backup settings

REQUEST

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

TEXT RESPONSE

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

```
Status=processing
ConnectionType=Wifi
ServerSSID=stw_post
PollingFrequency=30s
```

JSON RESPONSE

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

```
{
    "Status": "processing",
    "ConnectionType": "Wifi",
    "ServerSSID": "stw_post",
    "PollingFrequency": "30s"
}
```

16.4.2. Setting to make backups through the WiFi connection

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=autobackup&action=set&ConnectionType=WiFi&ServerSSID
=stw_post&PollingFrequency=5s
```

16.4.3. Setting to make backups through the Ethernet connection

REQUEST

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

```
cgi/system.cgi?msubmenu=autobackup&action=set&ConnectionType=Ethernet
```

Chapter 17. Digital Signage

17.1. Description

The **digitalsignage** submenu is to display the advertisements from the FTP server if a certain input or event does not occur for the specified time.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
view	User
set	User

17.2. Syntax

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

17.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the settings for the digital signage function.
set	AutoStart	REQ, RES	<bool> True, False	Enables or disables the digital signage function.
	AutoStartDuration	REQ, RES	<int> 10s, 20s, 30s, 1m, 2m, 3m, 4m, 5m, 10m, 20m, 30m, 1h	Auto start duration **AutoStartDuration** is valid only when **AutoStart** is set to True.
	FTPSync	REQ, RES	<bool> True, False	Enables or disables the sync in the FTP server to get advertisements **FTPSync** is valid only when **AutoStart** is set to True.

Action	Parameters	Request/ Response	Type/ Value	Description
	FTPServer	REQ, RES	<string>	FTP server **FTPServer** is valid only when **AutoStart** and **FTPSync** are both set to True.
	FTPUserName	REQ, RES	<string>	FTP user name **FTPUserName** is valid only when **AutoStart** and **FTPSync** are both set to True.
	FTPPassword	REQ, RES	<string>	FTP password **FTPPassword** isvalid only when **AutoStart** and **FTPSync** are both set to True.
	IsFTPPasswordEncrypted	REQ	<bool> True, False	When this is set as true, password is encrypted using the public key obtained using the **rsa** submenu of security.cgi and sent as payload content for the POST command.
	FTPFilename	REQ, RES	<string>	FTP file name **FTPFilename** is valid only when **AutoStart** and **FTPSync** are both set to True.

17.4. Examples

17.4.1. Getting the current digital signage settings

REQUEST

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

TEXT RESPONSE

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

```
AutoStart=True
```

```
AutoStartDuration=1m
FTPSync=True
FTPServer=ftpserver
FTPUserName=anonymous
FTPPassword=signage1234
FTPPath=DigitalSignage.dsf
```

JSON RESPONSE

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

```
{
    "AutoStart": true,
    "AutoStartDuration": "1m",
    "FTPSync": true,
    "FTPServer": "ftpserver",
    "FTPUserName": "anonymous",
    "FTPPassword": "signage1234",
    "FTPPath": "DigitalSignage.dsf"
}
```

17.4.2. Setting to use the advertisements from the FTP server

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=digitalsignage&action=set&AutoStart=Ture&AutoStartDu
ration=1m&FTPSync=True&FTPServer=stw_signage_server&FTPUserName=admin&FTPPas
sword=123456&FTPPath=DigitalSignage.dsf
```

17.4.3. Setting to use the advertisements from USB

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=digitalsignage&action=set&AutoStart=True&AutoStartDu
ration=30s&FTPSync=False
```

Chapter 18. Vehicle Information

The **vehicleinformation** submenu specifies the vehicle-related information on the device.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
view	User
set	User

18.1. Syntax

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

18.2. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the vehicle information configured on the device
set	Company	REQ, RES	<string>	Company information of the vehicle
	Maker	REQ, RES	<string>	Vehicle brand name
	Model	REQ, RES	<string>	Vehicle model name
	RegistrationNumber	REQ, RES	<string>	Registration number of the vehicle (Number plate information)
	License	REQ, RES	<string>	Driver License information
	DriverName	REQ, RES	<string>	Driver name
	DelayUnits	REQ, RES	<enum> Seconds, Minutes, Hours	Shutdown delay unit
	ShutdownDelay	REQ, RES	<int>	Shutdown delay duration

18.3. Examples

Gets the vehicle information stored in the device.

REQUEST

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

TEXT RESPONSE

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

```
Company=ABC Corporation
Maker=Hyundai
Model=Sonata
RegistrationNumber=AZ1000
License=AZ10001001
DriverName=Thompson
DelayUnits=Seconds
ShutdownDelay=30
```

JSON RESPONSE

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

```
{
    "Company": "ABC Corporation",
    "Maker": "Hyundai",
    "Model": "Sonata",
    "RegistrationNumber": "AZ1000",
    "License": "AZ10001001",
    "DriverName": "Thompson",
    "DelayUnits": "Seconds",
    "ShutdownDelay": 30
}
```

Chapter 19. ONVIF Feature

The **onviffeature** submenu is used to enable or disable some features exposed in the ONVIF protocol.

> **NOTE** This chapter applies to cameras only.

Access level

Action	Camera
view	Admin
set	Admin

19.1. Syntax

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

19.2. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the ONVIF feature that can be enabled or disabled
set	FocusControl	REQ, RES	<bool>	Enables or disables focus control feature using ONVIF image service.

19.3. Examples

Get the current FocusControl status for onvif

REQUEST

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

TEXT RESPONSE

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

```
FocusControl=False
```

JSON RESPONSE

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

```
{
    "FocusControl": false
}
```

Chapter 20. Database Reset

This **databasereset** submenu is used to reset the database entries.

NOTE This chapter applies to cameras only.

Access level

Action	Camera
control	Admin

20.1. Syntax

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

20.2. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
control	IncludeDataType	REQ	<csv> PeopleCount, HeatMap, QueueEvents, All	Based on the parameter, the passed database will be reset.

20.3. Examples

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=databasereset&action=control&IncludeDataType=All
```

Chapter 21. Log Server

This **logserver** submenu is used to configure clients to receive log messages.

Access level

Action	Camera	NVR
view	Admin	User
add/update	Admin	User
remove	Admin	User

21.1. Syntax

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

21.2. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Index	REQ	<int>	
add/update	Index.#.Enable	REQ, RES	<bool> True, False	Enable or disable a client
	Index.#.IPType	REQ, RES	<enum> IPv4, IPv6	IP type selection
	Index.#.IPAddress	REQ, RES	<string>	IP address
	Index.#.Port	REQ,REQ	<int>	Port number
remove	Index	REQ	<csv>	List of index for removal

21.3. Examples

21.3.1. Getting the current logserver settings

REQUEST

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

TEXT RESPONSE

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

```
<Body>
```

```
Index.1.Enable=True
Index.1.IPType=IPv4
Index.1.IPAddress=192.168.1.100
Index.1.Port=501
Index.2.Enable=False
Index.2.IPType=IPv4
Index.2.IPAddress=192.168.1.101
Index.2.Port=502
```

JSON RESPONSE

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

```
{
    "LogServer": [
        {
            "Index": "1",
            "Enable": true,
            "IPType": "IPv4",
            "IPAddress": "192.168.1.100",
            "Port": 501
        },
        {
            "Index": "2",
            "Enable": false,
            "IPType": "IPv4",
            "IPAddress": "192.168.1.101",
            "Port": 502
        }
    ]
}
```

21.3.2. Add a new client

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=logserver&action=add&Enable=True&IPType=IPv4&IPAddre
ss=192.168.1.103&Port=503
```

TEXT RESPONSE

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

```
OK
Index=3
```

JSON RESPONSE

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

```
{
    "Response": "Success",
    "Index": 3
}
```

21.3.3. Remove client using index

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=logserver&action=remove&Index=1,2,3
```

TEXT RESPONSE

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

```
OK
```

JSON RESPONSE

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

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

Chapter 22. Session Info

This **sessioninfo** submenu is used to get the current sessions in use.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
view	User

22.1. Syntax

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

22.2. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Live	RES	<int>	Current live session count
	Search	RES	<int>	Current search session count
	Backup	RES	<int>	Current backup session count

22.3. Examples

REQUEST

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

TEXT RESPONSE

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

```
Live=2
Search=0
Backup=0
```

JSON RESPONSE

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

```
{
    "Live": 2,
    "Search": 0,
    "Backup": 0
}
```

Chapter 23. SD card information

23.1. Description

The **sdcardinfo** submenu is used to get the details of the current SD card.

> **NOTE** This chapter applies to multi directional cameras only.

Access level

Action	Camera
view	Admin

23.2. Syntax

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

23.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Channel	REQ	<csv>	
	Channel.#.UsedSpace	RES	<string>	SD card space used
	Channel.#.TotalSpace	RES	<string>	Total SD card capacity
	Channel.#.FileSystem	RES	<enum> EXT4, VFAT	Filesystem used in SD card
	Channel.#.Status	RES	<enum> Normal, Error, Active, Formatting, Lock, Full	Current state of SD card

23.4. Examples

REQUEST

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

TEXT RESPONSE

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

```
Channel.0.UsedSpace=15067
Channel.0.TotalSpace=30420
Channel.0.FileSystem=VFAT
Channel.0.Status=Active
Channel.1.UsedSpace=2269
Channel.1.TotalSpace=30420
Channel.1.FileSystem=VFAT
Channel.1.Status=Active
Channel.2.UsedSpace=0
Channel.2.TotalSpace=0
Channel.2.FileSystem=VFAT
Channel.2.Status=
Channel.3.UsedSpace=0
Channel.3.TotalSpace=0
Channel.3.FileSystem=VFAT
Channel.3.Status=
Channel.3.Status=
```

JSON RESPONSE

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

```
[
    {
        "Channel": 0,
        "UsedSpace": "15067",
        "TotalSpace": "30420",
        "FileSystem": "VFAT",
        "Status": "Active"
    },
    {
        "Channel": 1,
```

```
        "UsedSpace": "2269",
        "TotalSpace": "30420",
        "FileSystem": "VFAT",
        "Status": "Active"
    },
    {
        "Channel": 2,
        "UsedSpace": "0",
        "TotalSpace": "0",
        "FileSystem": "VFAT",
        "Status": ""
    },
    {
        "Channel": 3,
        "UsedSpace": "0",
        "TotalSpace": "0",
        "FileSystem": "VFAT",
        "Status": ""
    }
]
```

Chapter 24. ISCSI Discovery

24.1. Description

This **iscsidiscovery** submenu is used to get the ISCSI targets.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
control	ADMIN

24.2. Syntax

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

24.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
control	Mode	REQ	<enum> Discover	
	PortalIP	REQ	<string>	Storage group IP address
	Port	REQ	<int>	Port number of storage group
	CHAPUserID	REQ	<string>	User name
	CHAPPassword	REQ	<string>	Password
	IsCHAPPasswordEncrypted	REQ	<bool> True, False	When set to true, the password is encrypted using the public key provided by the **rsa** submenu of security.cgi and sent as a post payload.
	AvailableTargets	RES	<csv>	List of available targets

24.4. Examples

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=iscsidiscovery&action=control&Mode=Discover&PortalIp
```

```
=192.168.17.1&Port=3260&CHAPUserId=Martin&CHAPPassword=testtt
```

TEXT RESPONSE

```
AvailableTargets=Target1,Target2,Target3
```

JSON RESPONSE

```
{
    "AvailableTargets": [
        "Target1",
        "Target2",
        "Target3"
    ]
}
```

Chapter 25. Holiday

25.1. Description

The **holiday** submenu configures the holiday settings for the device.

Access level

Action	NVR	Decoder
view	User	User
set	User	User
remove	User	User

25.2. Syntax

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

25.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads the holiday settings.
	Year	REQ	<int>	Year for the holidays to be searched The values must be within the range of 2000 to 2037.
	Month	REQ	<csv> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	Month(s) for the holiday to be searched
	Holiday	RES	<csv> 1, 2, 3, 4, , , , 31	Holiday number in a year and in a month(s)
set	Year	REQ, RES	<int>	Year for the holiday The values must be within the range of 2000 to 2037.
	Month	REQ, RES	<int>	Month for the holiday

Action	Parameters	Request/ Response	Type/ Value	Description
	Day	REQ, RES	<csv> 1, 2, 3, 4, , , , 31	Day for the holiday
	Week	REQ, RES	<enum> First, Second, Third, Fourth, Last	Week for the holiday
	<ddd>	REQ, RES	<bool> True, False	Day of week for the holiday <ddd> represents day of the week, and should be specified in the short form such as SUN, MON, TUE, WED, THU, FRI, and SAT in uppercase. e.g.) 'SUN=True' indicates every Sunday of the month is set as the holiday.
remove	Year	REQ	<int>	Holiday year
	Month	REQ	<int>	Holiday month
	Holiday	REQ	<csv> 1, 2, 3, 4, , , , 31	Holiday number

25.4. Examples

25.4.1. Getting holiday settings

REQUEST

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

TEXT RESPONSE

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

```
Year.2018.Month.1.Holiday=
Year.2018.Month.1.Day=
```

```
Year.2018.Month.1.Week.First.WeekDay=
Year.2018.Month.1.Week.Second.WeekDay=
Year.2018.Month.1.Week.Third.WeekDay=
Year.2018.Month.1.Week.Fourth.WeekDay=
Year.2018.Month.1.Week.Last.WeekDay=
Year.2018.Month.2.Holiday=
Year.2018.Month.2.Day=
Year.2018.Month.2.Week.First.WeekDay=
Year.2018.Month.2.Week.Second.WeekDay=
Year.2018.Month.2.Week.Third.WeekDay=
Year.2018.Month.2.Week.Fourth.WeekDay=
Year.2018.Month.2.Week.Last.WeekDay=
Year.2018.Month.3.Holiday=
Year.2018.Month.3.Day=
Year.2018.Month.3.Week.First.WeekDay=
Year.2018.Month.3.Week.Second.WeekDay=
Year.2018.Month.3.Week.Third.WeekDay=
Year.2018.Month.3.Week.Fourth.WeekDay=
Year.2018.Month.3.Week.Last.WeekDay=
Year.2018.Month.4.Holiday=7,8,22,27
Year.2018.Month.4.Day=7,8,22,27
Year.2018.Month.4.Week.First.WeekDay=SAT
Year.2018.Month.4.Week.Second.WeekDay=SUN
Year.2018.Month.4.Week.Third.WeekDay=
Year.2018.Month.4.Week.Fourth.WeekDay=
Year.2018.Month.4.Week.Last.WeekDay=
Year.2018.Month.5.Holiday=1,7
Year.2018.Month.5.Day=1,7
Year.2018.Month.5.Week.First.WeekDay=
Year.2018.Month.5.Week.Second.WeekDay=
Year.2018.Month.5.Week.Third.WeekDay=
Year.2018.Month.5.Week.Fourth.WeekDay=
Year.2018.Month.5.Week.Last.WeekDay=
Year.2018.Month.6.Holiday=
Year.2018.Month.6.Day=
Year.2018.Month.6.Week.First.WeekDay=
Year.2018.Month.6.Week.Second.WeekDay=
Year.2018.Month.6.Week.Third.WeekDay=
Year.2018.Month.6.Week.Fourth.WeekDay=
Year.2018.Month.6.Week.Last.WeekDay=
Year.2018.Month.7.Holiday=
```

```
Year.2018.Month.7.Day=
Year.2018.Month.7.Week.First.WeekDay=
Year.2018.Month.7.Week.Second.WeekDay=
Year.2018.Month.7.Week.Third.WeekDay=
Year.2018.Month.7.Week.Fourth.WeekDay=
Year.2018.Month.7.Week.Last.WeekDay=
Year.2018.Month.8.Holiday=
Year.2018.Month.8.Day=
Year.2018.Month.8.Week.First.WeekDay=
Year.2018.Month.8.Week.Second.WeekDay=
Year.2018.Month.8.Week.Third.WeekDay=
Year.2018.Month.8.Week.Fourth.WeekDay=
Year.2018.Month.8.Week.Last.WeekDay=
Year.2018.Month.9.Holiday=
Year.2018.Month.9.Day=
Year.2018.Month.9.Week.First.WeekDay=
Year.2018.Month.9.Week.Second.WeekDay=
Year.2018.Month.9.Week.Third.WeekDay=
Year.2018.Month.9.Week.Fourth.WeekDay=
Year.2018.Month.9.Week.Last.WeekDay=
Year.2018.Month.10.Holiday=
Year.2018.Month.10.Day=
Year.2018.Month.10.Week.First.WeekDay=
Year.2018.Month.10.Week.Second.WeekDay=
Year.2018.Month.10.Week.Third.WeekDay=
Year.2018.Month.10.Week.Fourth.WeekDay=
Year.2018.Month.10.Week.Last.WeekDay=
Year.2018.Month.11.Holiday=
Year.2018.Month.11.Day=
Year.2018.Month.11.Week.First.WeekDay=
Year.2018.Month.11.Week.Second.WeekDay=
Year.2018.Month.11.Week.Third.WeekDay=
Year.2018.Month.11.Week.Fourth.WeekDay=
Year.2018.Month.11.Week.Last.WeekDay=
Year.2018.Month.12.Holiday=
Year.2018.Month.12.Day=
Year.2018.Month.12.Week.First.WeekDay=
Year.2018.Month.12.Week.Second.WeekDay=
Year.2018.Month.12.Week.Third.WeekDay=
Year.2018.Month.12.Week.Fourth.WeekDay=
Year.2018.Month.12.Week.Last.WeekDay=
```

JSON RESPONSE

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


```
{
    "Year": 2018,
    "MonthWiseHolidays": [
        {
            "Month": 1,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
                    "WeekDay": []
                },
                {
                    "Week": "Fourth",
                    "WeekDay": []
                },
                {
                    "Week": "Last",
                    "WeekDay": []
                }
            ]
        },
        {
            "Month": 2,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
```

```
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
                    "WeekDay": []
                },
                {
                    "Week": "Fourth",
                    "WeekDay": []
                },
                {
                    "Week": "Last",
                    "WeekDay": []
                }
            ]
        },
        {
            "Month": 3,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
                    "WeekDay": []
                },
                {
                    "Week": "Fourth",
```

```
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
    {
        "Month": 4,
        "Holiday": [
            "7",
            "8",
            "22",
            "27"
        ],
        "Day": [
            "7",
            "8",
            "22",
            "27"
        ],
        "WeekwiseHolidays": [
            {
                "Week": "First",
                "WeekDay": [
                    "SAT"
                ]
            },
            {
                "Week": "Second",
                "WeekDay": [
                    "SUN"
                ]
            },
            {
                "Week": "Third",
                "WeekDay": []
            },
            {
                "Week": "Fourth",
```

```
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
    {
        "Month": 5,
        "Holiday": [
            "1",
            "7"
        ],
        "Day": [
            "1",
            "7"
        ],
        "WeekwiseHolidays": [
            {
                "Week": "First",
                "WeekDay": []
            },
            {
                "Week": "Second",
                "WeekDay": []
            },
            {
                "Week": "Third",
                "WeekDay": []
            },
            {
                "Week": "Fourth",
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
```

```
        {
            "Month": 6,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
                    "WeekDay": []
                },
                {
                    "Week": "Fourth",
                    "WeekDay": []
                },
                {
                    "Week": "Last",
                    "WeekDay": []
                }
            ]
        },
        {
            "Month": 7,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
```

```
                "Week": "Third",
                "WeekDay": []
            },
            {
                "Week": "Fourth",
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
    {
        "Month": 8,
        "Holiday": [],
        "Day": [],
        "WeekwiseHolidays": [
            {
                "Week": "First",
                "WeekDay": []
            },
            {
                "Week": "Second",
                "WeekDay": []
            },
            {
                "Week": "Third",
                "WeekDay": []
            },
            {
                "Week": "Fourth",
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
    {
```

```
            "Month": 9,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
                    "WeekDay": []
                },
                {
                    "Week": "Fourth",
                    "WeekDay": []
                },
                {
                    "Week": "Last",
                    "WeekDay": []
                }
            ]
        },
        {
            "Month": 10,
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
```

```
                "WeekDay": []
            },
            {
                "Week": "Fourth",
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
    {
        "Month": 11,
        "Holiday": [],
        "Day": [],
        "WeekwiseHolidays": [
            {
                "Week": "First",
                "WeekDay": []
            },
            {
                "Week": "Second",
                "WeekDay": []
            },
            {
                "Week": "Third",
                "WeekDay": []
            },
            {
                "Week": "Fourth",
                "WeekDay": []
            },
            {
                "Week": "Last",
                "WeekDay": []
            }
        ]
    },
    {
        "Month": 12,
```

```
            "Holiday": [],
            "Day": [],
            "WeekwiseHolidays": [
                {
                    "Week": "First",
                    "WeekDay": []
                },
                {
                    "Week": "Second",
                    "WeekDay": []
                },
                {
                    "Week": "Third",
                    "WeekDay": []
                },
                {
                    "Week": "Fourth",
                    "WeekDay": []
                },
                {
                    "Week": "Last",
                    "WeekDay": []
                }
            ]
        }
    ]
}
```

25.4.2. Setting June 2018 as the holiday

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=holiday&action=set&Year=2018&Month=6&Day=2,3,9,10,16
,17,23,24
```

25.4.3. Deselecting April 2018 from the holidays

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=holiday&action=remove&Year=2018&Month=4
```

Chapter 26. HDD Alarm

26.1. Description

The **hddalarm** submenu configures the HDD alarm settings for the device.

Access level

Action	NVR	Decoder
view	User	User
set	User	User

26.2. Syntax

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

26.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads HDD Alarm settings.
	Type	REQ	<csv> Check, Replace, iSCSI	Alarm output terminal yype Check status means that the HDD is operating but it has problems that require technical examination. Replace status means that the HDD has defect and requires immediate replacement.
set	Type	REQ, RES	<csv> Check, Replace, iSCSI	Alarm output terminal type
	AlarmOutput	REQ, RES	<csv> 1, 2, 3, 4, Beep, None	Alarm output number If Beep was selected, a beep will sound. Alarm signal will output through the alarm out port on the rear side when select <1>, <2>, <3> and <4>.

Action	Parameters	Request/ Response	Type/ Value	Description
	Duration	REQ, RES	<enum> None, 5s, 10s, 20s, 30s, Always	Alarm duration for the alarm signal and beep sound

26.4. Examples

26.4.1. Getting HDD alarm settings

REQUEST

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

TEXT RESPONSE

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

```
Type.Check.AlarmOutput=Beep
Type.Check.Duration=Always
Type.Replace.AlarmOutput=Beep
Type.Replace.Duration=Always
Type.Replace.AlarmOutput=Beep
Type.Replace.Duration=Always
```

JSON RESPONSE

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

```
{
    "HDDAlarm": [
        {
            "Type": "Check",
            "AlarmOutput": [
                "Beep"
            ],
```

```
            "Duration": "Always"
        },
        {
            "Type": "Replace",
            "AlarmOutput": [
                "Beep"
            ],
            "Duration": "Always"
        },
        {
            "Type": "Replace",
            "AlarmOutput": [
                "Beep"
            ],
            "Duration": "Always"
        }
    ]
}
```

26.4.2. Setting HDD alarm

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=hddalarm&action=set&Type=Check&AlarmOutput=1&Duratio
n=10s
```

Chapter 27. Monitor Input

27.1. Description

The **monitorin** submenu configures the monitor input settings for the device.

Access level

Action	Decoder
view	User
set	User

27.2. Syntax

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

27.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view		REQ	<csv>	Reads monitor in settings for the device.
	Index	REQ	<csv>	Index number
	Index.#.VideoOutput	RES	<enum> HDMI, VGA	Video output format of monitor in device
	Index.#.ConnectedVGAIndex	RES	<int>	Connected VGA index number
	Index.#.Use	RES	<bool> True, False	Enable or disable status of monitor in device
set	Index.#.Resolution	REQ, RES	<enum> 1280x720_HDMI, 1920x1080_HDMI, 1280x720_DVI, 1920x1080_DVI	Resolution of monitor in device

27.4. Examples

27.4.1. Getting monitor input settings

REQUEST

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

TEXT RESPONSE

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

```
Index.1.Use=True
Index.1.ConnectedVGAIndex=1
Index.1.VideoOutput=HDMI
Index.1.Resolution=1920x1080_HDMI
Index.2.Use=True
Index.2.ConnectedVGAIndex=2
Index.2.VideoOutput=HDMI
Index.2.Resolution=1920x1080_HDMI
Index.3.Use=False
Index.3.ConnectedVGAIndex=3
Index.3.VideoOutput=HDMI
Index.3.Resolution=1920x1080_HDMI
Index.4.Use=False
Index.4.ConnectedVGAIndex=4
Index.4.VideoOutput=HDMI
Index.4.Resolution=1920x1080_HDMI
Index.5.Use=False
Index.5.ConnectedVGAIndex=5
Index.5.VideoOutput=HDMI
Index.5.Resolution=1920x1080_HDMI
Index.6.Use=False
Index.6.ConnectedVGAIndex=6
Index.6.VideoOutput=HDMI
Index.6.Resolution=1920x1080_HDMI
Index.7.Use=False
Index.7.ConnectedVGAIndex=7
Index.7.VideoOutput=HDMI
Index.7.Resolution=1920x1080_HDMI
```

```
Index.8.Use=False
Index.8.ConnectedVGAIndex=8
Index.8.VideoOutput=HDMI
Index.8.Resolution=1920x1080_HDMI
```

JSON RESPONSE

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

```
{
    "MonitorIn": [
        {
            "Index": 1,
            "Use": true,
            "ConnectedVGAIndex": 1,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
        {
            "Index": 2,
            "Use": true,
            "ConnectedVGAIndex": 2,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
        {
            "Index": 3,
            "Use": false,
            "ConnectedVGAIndex": 3,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
        {
            "Index": 4,
            "Use": false,
            "ConnectedVGAIndex": 4,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
```

```
        {
            "Index": 5,
            "Use": false,
            "ConnectedVGAIndex": 5,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
        {
            "Index": 6,
            "Use": false,
            "ConnectedVGAIndex": 6,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
        {
            "Index": 7,
            "Use": false,
            "ConnectedVGAIndex": 7,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        },
        {
            "Index": 8,
            "Use": false,
            "ConnectedVGAIndex": 8,
            "VideoOutput": "HDMI",
            "Resolution": "1920x1080_HDMI"
        }
    ]
}
```

27.4.2. Setting monitor input resolution as 1280X720_HDMI at Index 6

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=monitorin&action=set&Index.6.Resolution=1280x720_HDM
I
```

Chapter 28. Monitor Out

28.1. Description

The **monitorout** submenu configures the monitor output settings for the device.

Access level

Action	Decoder	NVR
view	User	User
set	User	User

28.2. Syntax

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

28.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				Reads monitor out settings.
	Index	REQ	<csv>	Index number
	Index.#.VideoOutput	RES	<enum> HDMI, VGA	Video output format of monitor out device
	Index.#.ConnectedVGAIndex	RES	<int>	Connected VGA index number
	Index.#.Use	RES	<bool> True, False	Enable or disable status of monitor out device
	Index.#.OptimalResolution	RES	<string> widthxheight	Optimal video output resolution (e.g. 1920x1080)
set	Index.#.Resolution	REQ, RES	<enum> 858x480, 1280x1024, 1280x720, 1920x1080, 2560x1440, 3840x2160	Resolution of monitor out device

Action	Parameters	Request/ Response	Type/ Value	Description
	Display	REQ, RES	<csv> ChannelName, IPAddress, Date, Time, Icon, Resolution, FPS, MonitorIndex, None	Display output

28.4. Examples

28.4.1. Getting monitor out settings

REQUEST

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

TEXT RESPONSE

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

```
Display=None
Index.1.Use=True
Index.1.ConnectedVGAIndex=1
Index.1.VideoOutput=HDMI
Index.1.Resolution=1280x1024
Index.2.Use=True
Index.2.ConnectedVGAIndex=1
Index.2.VideoOutput=HDMI
Index.2.Resolution=1280x1024
Index.3.Use=True
Index.3.ConnectedVGAIndex=2
Index.3.VideoOutput=HDMI
Index.3.Resolution=1280x1024
Index.4.Use=True
Index.4.ConnectedVGAIndex=2
```

```
Index.4.VideoOutput=HDMI
Index.4.Resolution=1280x1024
Index.5.Use=False
Index.5.ConnectedVGAIndex=3
Index.5.VideoOutput=HDMI
Index.5.Resolution=1280x1024
Index.6.Use=False
Index.6.ConnectedVGAIndex=3
Index.6.VideoOutput=HDMI
Index.6.Resolution=1280x1024
Index.7.Use=False
Index.7.ConnectedVGAIndex=4
Index.7.VideoOutput=HDMI
Index.7.Resolution=1280x1024
Index.8.Use=False
Index.8.ConnectedVGAIndex=4
Index.8.VideoOutput=HDMI
Index.8.Resolution=1280x1024
Index.9.Use=False
Index.9.ConnectedVGAIndex=5
Index.9.VideoOutput=HDMI
Index.9.Resolution=1280x1024
Index.10.Use=False
Index.10.ConnectedVGAIndex=5
Index.10.VideoOutput=HDMI
Index.10.Resolution=1280x1024
Index.11.Use=False
Index.11.ConnectedVGAIndex=6
Index.11.VideoOutput=HDMI
Index.11.Resolution=1280x1024
Index.12.Use=False
Index.12.ConnectedVGAIndex=6
Index.12.VideoOutput=HDMI
Index.12.Resolution=1280x1024
Index.13.Use=False
Index.13.ConnectedVGAIndex=7
Index.13.VideoOutput=HDMI
Index.13.Resolution=1280x1024
Index.14.Use=False
Index.14.ConnectedVGAIndex=7
Index.14.VideoOutput=HDMI
```

```
Index.14.Resolution=1280x1024
Index.15.Use=False
Index.15.ConnectedVGAIndex=8
Index.15.VideoOutput=HDMI
Index.15.Resolution=1280x1024
Index.16.Use=False
Index.16.ConnectedVGAIndex=8
Index.16.VideoOutput=HDMI
Index.16.Resolution=1280x1024
```

JSON RESPONSE

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

```
{
    "Display": [
        "None"
    ],
    "MonitorOut": [
        {
            "Index": 1,
            "Use": true,
            "ConnectedVGAIndex": 1,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 2,
            "Use": true,
            "ConnectedVGAIndex": 1,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 3,
            "Use": true,
            "ConnectedVGAIndex": 2,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
```

```
        },
        {
            "Index": 4,
            "Use": true,
            "ConnectedVGAIndex": 2,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 5,
            "Use": false,
            "ConnectedVGAIndex": 3,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 6,
            "Use": false,
            "ConnectedVGAIndex": 3,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 7,
            "Use": false,
            "ConnectedVGAIndex": 4,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 8,
            "Use": false,
            "ConnectedVGAIndex": 4,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 9,
            "Use": false,
            "ConnectedVGAIndex": 5,
            "VideoOutput": "HDMI",
```

```
            "Resolution": "1280x1024"
        },
        {
            "Index": 10,
            "Use": false,
            "ConnectedVGAIndex": 5,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 11,
            "Use": false,
            "ConnectedVGAIndex": 6,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 12,
            "Use": false,
            "ConnectedVGAIndex": 6,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 13,
            "Use": false,
            "ConnectedVGAIndex": 7,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 14,
            "Use": false,
            "ConnectedVGAIndex": 7,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 15,
            "Use": false,
            "ConnectedVGAIndex": 8,
```

```
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        },
        {
            "Index": 16,
            "Use": false,
            "ConnectedVGAIndex": 8,
            "VideoOutput": "HDMI",
            "Resolution": "1280x1024"
        }
    ]
}
```

28.4.2. Setting monitor out resolution as 1280x720 at Index 6

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=monitorout&action=set&Index.6.Resolution=1280x720
```

Chapter 29. USB Configuration

29.1. Description

The **usbconfig** submenu configures the USB port on the camera.

> **NOTE** This chapter applies to camera only.

Access level

Action	Camera
view	Admin
set	Admin

29.2. Syntax

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

29.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				
set	Enable	REQ, RES	<bool> True, False	Enables or disables the USB port.

29.4. Examples

29.4.1. Getting USB configuration usbconfig

REQUEST

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

TEXT RESPONSE

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

```
Enable=True
```

JSON RESPONSE

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

```
{
    "Enable": true
}
```

29.4.2. Setting to enable the USB port

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=usbconfig&action=set&Enable=True
```

Chapter 30. Stratocast Service Configuration

30.1. Description

The **stratocast** submenu provides URLs to enroll devices to Stratocast web cloud service provided by Genetec.

NOTE This chapter applies to cameras only.

Access level

Action	Camera
view	Admin
set	Admin

30.2. Syntax

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

30.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				
set	ServiceURL	REQ, RES	<string>	URL of Stratocast web cloud service in EN
	DeviceEntryURL	REQ, RES	<string>	Device entry point When **SimplifiedEnrollmentEnable** is set to true, the device sends its own information to **DeviceEntryURL**. Stratocast collects and takes this information to the queue for registration. When user reads the QR code of the device with Stratocast mobile application, the device will finally be activated.

Action	Parameters	Request/ Response	Type/ Value	Description
	ProbeServiceURL	REQ, RES	<string>	URL of Stratocast probe server The device sends its own information such as CPU/memory usage and logs to the probe server. The probe server can send commands to the camera (e.g. reboot)
	CameraProbeEnable	REQ, RES	<bool> True, False	Enables or disables sending device information to the probe server
	ProbeInterval	REQ, RES	<int>	Interval between probes The range is from 30 to 300.
	SimplifiedEnrollmentEnable	REQ, RES	<bool> True, False	Enables or disables simplified enrollment When simplified enrollment is enabled, users can activate their cameras with a QR code using Stratocast mobile application without accessing any web viewers

30.4. Examples

30.4.1. Getting the current configurations

REQUEST

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

TEXT RESPONSE

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

```
ServiceURL=app.stratcast.com
DeviceEntryURL=
ProbeServiceURL=app.stratocast.com
CameraProbeEnable=True
ProbeInterval=30
```

```
SimplifiedEnrollmentEnable=False
```

JSON RESPONSE

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

```
{
    "ServiceURL": "app.startocast.com",
    "DeviceEntryURL": "",
    "ProbeServieURL": "app.stratocast.com",
    "CamerProbeEnable": true,
    "ProbeInterval": 30,
    "SimplifiedEnrollmentEnable": false
}
```

30.4.2. Enabling the transfer of the camera information to the probe server

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=stratocast&action=set&CamerProbeEnable=True&ProbeInt
erval=100
```

Chapter 31. Status of Stratocast Service

31.1. Description

The **stratocastregister** submenu provides the current status of registration process on Stratocast web cloud services.

NOTE | This chapter applies to cameras only.

Access level

Action	Camera
view	Admin
set	Admin
check	Admin

31.2. Syntax

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

31.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				
set	ActivationCode	REQ, RES	<string>	Unique identification code to register the device with Stratocast service **ActivationCode** must be issued from Stratocast server.
check	RegistrationPhase	RES	<enum> ActivationReady, GetEnrollmentInfo, GetNTPServerInfo, RegistrationDevice , GetLoadBalancerInfo, GetSSHServerInfo, ConnectSSHSession, ActivationDone	Current status when registration process is ongoing

Action	Parameters	Request/ Response	Type/ Value	Description
	Detail	RES	<string>	With **RegistrationPhase**, an additional description is provided

31.4. Examples

31.4.1. Getting the activation code

REQUEST

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

TEXT RESPONSE

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

```
ActivationCode=
```

JSON RESPONSE

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

```
{
    "ActivationCode": ""
}
```

31.4.2. Setting the activation code issued by the Stratocast service

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=stratocastregister&action=set&ActivationCode=0123456
7890
```

31.4.3. Checks the current status of registration process

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=stratocastregister&action=check
```

TEXT RESPONSE

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

```
RegistrationPhase=ActivationDone
Detail=
```

JSON RESPONSE

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

```
{
    "RegistrationPhase": "ActivationDone",
    "Detail": ""
}
```

Chapter 32. Peer Connection Information

32.1. Description

The **peerconnectioninfo** submenu provides the session status of the currently connected client.

> **NOTE** This chapter applies to cameras only.

Access level

Action	Camera
view	Admin

32.2. Syntax

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

32.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				
	Client.#.IPAddress	RES	<string>	IP address IP address of the connected user

Action	Parameters	Request/ Response	Type/ Value	Description
	Client.#.ClientHttpsStatus	RES	<enum> NO_HTTPS, HTTPS_WITHOUT_CLIENT_CERT, HTTPS_WITH_INVALID_CLIENT_CERT, HTTPS_WITH_VALID_CLIENT_CERT	Client HTTPS status HTTPS status of the connected user • NO_HTTPS: Client is connected using HTTP • HTTPS_WITHOUT_CLIENT_CERT: Client is connected using HTTPS without client certificate • HTTPS_WITH_INVALID_CLIENT_CERT: Client is connected using HTTPS with client certificate • HTTPS_WITH_VALID_CLIENT_CERT: Client connected with a valid certificate.

32.4. Examples

32.4.1. Getting peer connection information

REQUEST

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

TEXT RESPONSE

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

```
Client.1.IPAddress=192.168.71.93
Client.1.ClientHttpsStatus=NO_HTTPS
Client.2.IPAddress=192.168.71.93
Client.2.ClientHttpsStatus=HTTPS_WITHOUT_CLIENT_CERT
```

JSON RESPONSE

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

```
{
    "Clients": [
        {
            "Index": 1,
            "IPAddress": "192.168.71.93",
            "ClientHttpsStatus": "NO_HTTPS"
        },
        {
            "Index": 2,
            "IPAddress": "192.168.71.93",
            "ClientHttpsStatus": "HTTPS_WITHOUT_CLIENT_CERT"
        }
    ]
}
```

Chapter 33. IOBox connection

33.1. Description

The **clientregister** submenu is used for IO Box. This submenu allows clients (cameras) to connect to IO Box. Clients can check connection status. This submenu is related to **ioboxregister** submenu, which is a camera submenu. By using **ioboxregister**, clients can configure information needed to connect to IO Box.

NOTE

This chapter applies to IO Box only.
As of now, client is only camera, but it would be other VMS or device in the feature.

Access level

Action	IOBox
check	Admin
control	Admin

33.2. Syntax

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

33.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
check	ConnectionStatus	RES	<enum> "NotConnected", "ConnectedByMe", "ConnectedByOther"	• NotConnected: IO Box is not connected to client. Please check **ioboxregistrer** submenu's configuration and request **ConnectionRequest=True**. • ConnectedByMe: IO Box is connected to client which requests check action. • ConnectedByOther: IO Box is already connected to another client.

Action	Parameters	Request/ Response	Type/ Value	Description
control	ConnectionRequest	REQ	<bool>	Request to connect to IO Box based on **ioboxregister** submenu's configuration. Only 1 to 1 connection is possible.

33.4. Examples

33.4.1. Getting IOBox connection information

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=clientregister&action=check
```

JSON RESPONSE

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

```
{
    "ConnectionStatus": "NotConnected"
}
```

Chapter 34. Geolocation

34.1. Description

The **geolocation** submenu provides how to set and get the geolocation information of the device.

> **NOTE** This chapter applies to cameras only.

Access level

Action	Camera
view	User
set	Admin

34.2. Syntax

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

34.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				
set	Mode	REQ, RES	<enum> Static, Auto	When GPS sensor is supported, Mode is set to Auto, if not Static
	Latitude	REQ, RES	<float>	Latitude of device
	Longitude	REQ, RES	<float>	Longitude of device
	Elevation	REQ, RES	<float>	Elevation of device

34.4. Examples

34.4.1. Getting geolocation information

REQUEST

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

TEXT RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain
<Body>
```

```
Mode=Static
Longitude=30.3333
Latitude=12.0
Elevation=40.44
```

JSON RESPONSE

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

```
{
    "Mode": "Static",
    "Longitude": 30.3333,
    "Latitude": 12,
    "Elevation": 40.44
}
```

34.4.2. Setting longitude of device

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?
msubmenu=geolocation&action=set&Longitude=24.99
```

Chapter 35. SystemImage

35.1. Description

The **systemimage** submenu is used to transmit the image files used by the device to the client.

> **NOTE** This chapter applies to NVR only.

Access level

Action	NVR
view	User

35.2. Syntax

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

35.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	ImageType	RES	<enum> HDDMAP, P2PQRCODE, QRHELP,MOBILEIOS,MOBILEAND	

35.4. Examples

35.4.1. Retrieving the p2pqrcode image

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=systemimage&action=view&ImageType=P2PQRCODE
```

RESPONSE

```
HTTP/1.0 200 OK
Content-type: image/png or jpeg
```

```
<Body>
```

```
<PNG image data>
```

Chapter 36. PowerMode

36.1. Description

The **powermode** submenu used to configure the input power mode of the device.

> **NOTE** This chapter applies to cameras only.

Access level

Action	Camera
view	Guest
set	Admin

36.2. Syntax

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

36.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view				
set	Mode	REQ, RES	<enum> PoE+, PoE	Input power mode

36.4. Examples

36.4.1. Getting the current power mode

REQUEST

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

TEXT RESPONSE

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

```
Mode=PoE+
```

JSON RESPONSE

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

```
{
    "Mode": "PoE+"
}
```

36.4.2. Changing the power mode

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=powermode&action=set&Mode=PoE
```

TEXT RESPONSE

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

```
OK
```

JSON RESPONSE

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

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

Chapter 37. Registered Subdevices

37.1. Description

The **registeredsubdevices** submenu can be used to get the list of registered subdevices. A subdevice can be either a speaker or mic.

> **NOTE** This chapter applies to only AMS.
> If the device has an inbuilt speaker, the first subdevice will be local speaker, identified with IP address of the same device. In case of SPA-S1000 device, there is no inbuilt speaker.

Access level

Action	AMS
view	Admin

37.2. Syntax

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

37.3. Parameters

Action	Parameter	Request/ Response	Type/ Value	Description
view	Device.#.IP	RES	<string>	IP V4 address format or IPV6 address format
	Device.#.ID	RES	<int>	Unique id of the device
	Device.#.Type	RES	<enum> SPEAKER,MIC	Can be one of the enum values, either speaker or mic.
	Device.#.Name	RES	<string>	Optional name of the device

37.4. Examples

37.4.1. Getting sub devices information

REQUEST

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

JSON RESPONSE

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


```
{
  "Device":[
    {
       "IP":"192.168.71.22",
       "ID": 1221,
       "Name": "Speaker1stFloor",
       "Type": "SPEAKER"
    },
    {
       "IP":"192.168.71.23",
       "ID": 1223,
       "Name": "Speaker2ndFloor",
       "Type": "SPEAKER"
    },
    {
       "IP":"192.168.71.24",
       "ID": 1224,
       "Name": "Speaker3ndFloor",
       "Type": "SPEAKER"
    },
    {
       "IP":"192.168.71.20",
       "ID": 1225,
       "Name": "Mic1ControlRoom",
       "Type": "MIC"
    },
    {
       "IP":"192.168.71.21",
       "ID": 1226,
       "Name": "Mic2ControlRoom",
       "Type": "MIC"
    }
    ]
}
```

Chapter 38. Speaker groups

38.1. Description

The **speakergroups** submenu can be used to get speaker groups configured on the master speaker.

NOTE | This chapter applies to only AMS.

Access level

Action	AMS
view	Admin

38.2. Syntax

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

38.3. Parameters

Action	Parameter	Request/ Response	Type/ Value	Description
view	Group.#.ID	RES	<int>	Unique id of the device
	Group.#.DeviceIDs	RES	<csv>	Array of device ids in the group, only device type speaker can be part of group.
	Group.#.Name	RES	<string>	Optional name of the group

38.4. Examples

38.4.1. Getting group information

REQUEST

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

JSON RESPONSE

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

```
{
  "Group":[
    {
       "ID": 1,
       "Name": "Speakerslistin1stand2nd",
       "DeviceIDs": [ 1221,1223]
    },
    {
       "ID": 2,
       "Name": "Speakerslistin1stand3rd",
        "DeviceIDs": [ 1221,1224]
    }
    ]
}
```

Chapter 39. SSDStorage

39.1. Description

The **ssdstorage** submenu used to manage the ssd storage installed on the device.

> **NOTE**
> This chapter applies to cameras only
> Attribute to check: "attributes/System/Support/**SSDStorage**"

Access level

Action	Camera
view	Admin
set	Admin
update	Admin
control	Admin

39.2. Syntax

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

39.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	Storage	REQ	<int>	Storage Index, if passed only requested storage information is provided in response. **Note** For max supported SSD Storage count check below attributes "attributes/System/Limit/**MaxSSD Storage**"
	Storage.#.Enable	RES	<bool>	Storage enable state.

Action	Parameters	Request/ Response	Type/ Value	Description
	Storage.#.Status	RES	<enum> None,Error, Uninitialized, Initializing, Formatting, Wait,InUse	Storage status
	Storage.#.TotalSpace	RES	<string>	Total Space in MB provided as string.
	Storage.#.Partition.#.Device	RES	<string>	Device path eg: /dev/sda1
	Storage.#.Partition.#.MountPoint	RES	<string>	mount path eg: /mnt/sda1
	Storage.#.Partition.#.File System	RES	<enum> ext4	Total Space in MB provided as string.
	Storage.#.Partition.#.Size	RES	<string>	Parition size in MB provided as string.
	Storage.#.Partition.#.FreeSize	RES	<string>	Free Space in MB provided as string.
	Storage.#.Partition.#.Status	RES	<enum> None, Formatting, InUse	Total Space in MB provided as string.
set	Storage.#.Enable	REQ	<bool>	Enable or Disable a storage.
update	Storage	<int>	REQ	Storage Index
	TotalPartitionCount	<int>	REQ	Total partition count **Note** For max parition count supported check below attributes "attributes/System/Limit/**MaxPartitionPerSSD**"
	Partition.#.SizeInMB	<int>	REQ	Partition size in MB.
control	Mode	REQ	<enum> Format	control operation supported on the SSD .
	Storage.#.Partition	REQ	<int>	Parition Index on which the control operation is applicable.
check	Storage	REQ	<int>	SSD Storage Index.
	Storage.#.WriteSpeed	<RES>	<int>	Write speed in MB/s.

Action	Parameters	Request/ Response	Type/ Value	Description
	Storage.#.ReadSpeed	<RES>	<int>	Read speed in MB/s.
	Storage.#.PowerOnHours	<RES>	<int>	Total power on hours.
	Storage.#.TemperatureInCelsius	<RES>	<int>	Temperature of SSD in celcius.
	Storage.#.ValidSpareBlock	<RES>	<int>	Total number of valid spareblock.
	Storage.#.SpareBlockAlarm	<RES>	<enum> Good,Normal,Poor,Bad	Status of SSD.
	Storage.#.RemainingLifeInPercentage	<RES>	<int>	Remining lifetime in percentage.
	Storage.#.LifetimeAlarm	<RES>	<enum> Good,Normal,Poor,Bad	Health state of SSD.
	Storage.#.SMART.Model	<RES>	<string>	Model name.
	Storage.#.SMART.SerialNumber	<RES>	<string>	Serial number of SSD.
	Storage.#.SMART.FirmwareVersion	<RES>	<string>	Firmware version of SSD.
	Storage.#.SMART.Attributes.#.Id	<RES>	<string>	SMART attribute id.
	Storage.#.SMART.Attributes.#.Name	<RES>	<string>	SMART attribute name.
	Storage.#.SMART.Attributes.#.Value	<RES>	<int>	SMART attribute value.

39.4. Examples

39.4.1. View Storage and Partition Information

REQUEST

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

TEXT RESPONSE

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

```
<Body>
```

```
Storage.1.Enable=True
Storage.1.Status=InUse
Storage.1.TotalSpace=976762
Storage.1.Partition.1.Device=/dev/sda1
Storage.1.Partition.1.MountPoint=/mnt/sda1
Storage.1.Partition.1.FileSystem=ext4
Storage.1.Partition.1.Size=93356
Storage.1.Partition.1.FreeSize=87764
Storage.1.Partition.1.Status=InUse
Storage.1.Partition.2.Device=/dev/sda2
Storage.1.Partition.2.MountPoint=/mnt/sda2
Storage.1.Partition.2.FileSystem=ext4
Storage.1.Partition.2.Size=93356
Storage.1.Partition.2.FreeSize=88512
Storage.1.Partition.2.Status=InUse
Storage.1.Partition.3.Device=/dev/sda3
Storage.1.Partition.3.MountPoint=/mnt/sda3
Storage.1.Partition.3.FileSystem=ext4
Storage.1.Partition.3.Size=93356
Storage.1.Partition.3.FreeSize=88512
Storage.1.Partition.3.Status=InUse
```

JSON RESPONSE

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

```
{
    "Storage": [
        {
            "Index": 1,
            "Enable": true,
            "Status": "InUse",
            "TotalSpace": "976762",
            "Partition": [
                {
                    "Index": 1,
```

```
                    "Device": "/dev/sda1",
                    "MountPoint": "/mnt/sda1",
                    "FileSystem": "ext4",
                    "Size": "93356",
                    "FreeSize": "87764",
                    "Status": "InUse"
                },
                {
                    "Index": 2,
                    "Device": "/dev/sda2",
                    "MountPoint": "/mnt/sda2",
                    "FileSystem": "ext4",
                    "Size": "93356",
                    "FreeSize": "88512",
                    "Status": "InUse"
                },
                {
                    "Index": 3,
                    "Device": "/dev/sda3",
                    "MountPoint": "/mnt/sda3",
                    "FileSystem": "ext4",
                    "Size": "93356",
                    "FreeSize": "88512",
                    "Status": "InUse"
                }
            ]
        }
    ]
}
```


39.4.2. Enable SSD Storage

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=ssdstorage&action=set&Storage.1.Enable=True
```

TEXT RESPONSE

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

```
<Body>
```

```
OK
```

JSON RESPONSE

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

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

39.4.3. Create Partitions in SSD

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=ssdstorage&action=update&Storage=1&TotalPartitionCou
nt=3&Partition.1.SizeInMB=100000&Partition.2.SizeInMB=100000&Partition.3.Siz
eInMB=100000
```

TEXT RESPONSE

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

```
OK
```

JSON RESPONSE

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

```
{
```

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

39.4.4. Format parition

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=ssdstorage&action=control&Mode=Format&Storage.1.Part
ition=1
```

TEXT RESPONSE

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

```
OK
```

JSON RESPONSE

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

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

39.5. Check the Health status

REQUEST

```
http://<Device IP>/stw-cgi/system.cgi?msubmenu=ssdstorage&action=check
```

TEXT RESPONSE

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

```
Storage.1.Health.WriteSpeed=0
Storage.1.Health.ReadSpeed=0
Storage.1.Health.PowerOnHours=770
Storage.1.Health.TemperatueInCelsius=57
Storage.1.Health.ValidSpareBlock=100
Storage.1.Health.SpareBlockAlarm=Good
Storage.1.Health.RemainingLifeInPercentage=100
Storage.1.Health.LifetimeAlarm=Good
Storage.1.Health.SMART.Model=TS2TMTS952T2
Storage.1.Health.SMART.SerialNumber=H512150083
Storage.1.Health.SMART.FirmwareVersion=02J0T6OC

Storage.1.Health.SMART.Attributes.1.Id=01
Storage.1.Health.SMART.Attributes.1.Name=Read Error Rate
Storage.1.Health.SMART.Attributes.1.Value=0
Storage.1.Health.SMART.Attributes.2.Id=05
Storage.1.Health.SMART.Attributes.2.Name=Reallocated Sectors Count
Storage.1.Health.SMART.Attributes.2.Value=0
Storage.1.Health.SMART.Attributes.3.Id=09
Storage.1.Health.SMART.Attributes.3.Name=Power-On Hours
Storage.1.Health.SMART.Attributes.3.Value=770
Storage.1.Health.SMART.Attributes.4.Id=0C
Storage.1.Health.SMART.Attributes.4.Name=Power Cycle Count
Storage.1.Health.SMART.Attributes.4.Value=36
Storage.1.Health.SMART.Attributes.5.Id=94
Storage.1.Health.SMART.Attributes.5.Name=SLC Total Erase Count
Storage.1.Health.SMART.Attributes.5.Value=148
Storage.1.Health.SMART.Attributes.6.Id=95
Storage.1.Health.SMART.Attributes.6.Name=SLC Maximum Erase Count
Storage.1.Health.SMART.Attributes.6.Value=13
Storage.1.Health.SMART.Attributes.7.Id=96
Storage.1.Health.SMART.Attributes.7.Name=SLC Minimum Erase Count
Storage.1.Health.SMART.Attributes.7.Value=0
Storage.1.Health.SMART.Attributes.8.Id=97
Storage.1.Health.SMART.Attributes.8.Name=SLC Average Erase Count
Storage.1.Health.SMART.Attributes.8.Value=1
Storage.1.Health.SMART.Attributes.9.Id=9F
Storage.1.Health.SMART.Attributes.9.Name=DRAM one bit error count
Storage.1.Health.SMART.Attributes.9.Value=0
Storage.1.Health.SMART.Attributes.10.Id=A0
Storage.1.Health.SMART.Attributes.10.Name=Uncorrectable sectors count when
```

```
read/write
Storage.1.Health.SMART.Attributes.10.Value=0
Storage.1.Health.SMART.Attributes.11.Id=A1
Storage.1.Health.SMART.Attributes.11.Name=Number of Valid Spare Blocks
Storage.1.Health.SMART.Attributes.11.Value=114
Storage.1.Health.SMART.Attributes.12.Id=A3
Storage.1.Health.SMART.Attributes.12.Name=Number of Initial Invalid Blocks
Storage.1.Health.SMART.Attributes.12.Value=34
Storage.1.Health.SMART.Attributes.13.Id=A4
Storage.1.Health.SMART.Attributes.13.Name=TLC Total Erase Count
Storage.1.Health.SMART.Attributes.13.Value=100
Storage.1.Health.SMART.Attributes.14.Id=A5
Storage.1.Health.SMART.Attributes.14.Name=TLC Maximum Erase Count
Storage.1.Health.SMART.Attributes.14.Value=4
Storage.1.Health.SMART.Attributes.15.Id=A6
Storage.1.Health.SMART.Attributes.15.Name=TLC Minimum Erase Count
Storage.1.Health.SMART.Attributes.15.Value=0
Storage.1.Health.SMART.Attributes.16.Id=A7
Storage.1.Health.SMART.Attributes.16.Name=TLC Average Erase Count
Storage.1.Health.SMART.Attributes.16.Value=0
Storage.1.Health.SMART.Attributes.17.Id=A8
Storage.1.Health.SMART.Attributes.17.Name=Max Erase Count of Spec
Storage.1.Health.SMART.Attributes.17.Value=3000
Storage.1.Health.SMART.Attributes.18.Id=A9
Storage.1.Health.SMART.Attributes.18.Name=Remain Life (percentage)
Storage.1.Health.SMART.Attributes.18.Value=100
Storage.1.Health.SMART.Attributes.19.Id=B1
Storage.1.Health.SMART.Attributes.19.Name=Total Wear Level Count
Storage.1.Health.SMART.Attributes.19.Value=0
Storage.1.Health.SMART.Attributes.20.Id=B5
Storage.1.Health.SMART.Attributes.20.Name=Total Program Fail Count
Storage.1.Health.SMART.Attributes.20.Value=0
Storage.1.Health.SMART.Attributes.21.Id=B6
Storage.1.Health.SMART.Attributes.21.Name=Total Erase Fail Count
Storage.1.Health.SMART.Attributes.21.Value=0
Storage.1.Health.SMART.Attributes.22.Id=C0
Storage.1.Health.SMART.Attributes.22.Name=Power-Off Retract Count
Storage.1.Health.SMART.Attributes.22.Value=28
Storage.1.Health.SMART.Attributes.23.Id=C2
Storage.1.Health.SMART.Attributes.23.Name=Temperature
Storage.1.Health.SMART.Attributes.23.Value=57
```

```
Storage.1.Health.SMART.Attributes.24.Id=C3
Storage.1.Health.SMART.Attributes.24.Name=Total Correctable Count
Storage.1.Health.SMART.Attributes.24.Value=0
Storage.1.Health.SMART.Attributes.25.Id=C4
Storage.1.Health.SMART.Attributes.25.Name=Reallocation Event Count
Storage.1.Health.SMART.Attributes.25.Value=0
Storage.1.Health.SMART.Attributes.26.Id=C7
Storage.1.Health.SMART.Attributes.26.Name=Ultra DMA CRC Error Count
Storage.1.Health.SMART.Attributes.26.Value=0
Storage.1.Health.SMART.Attributes.27.Id=E8
Storage.1.Health.SMART.Attributes.27.Name=Available Reserved Space
Storage.1.Health.SMART.Attributes.27.Value=100
Storage.1.Health.SMART.Attributes.28.Id=F1
Storage.1.Health.SMART.Attributes.28.Name=Total LBA Written (each write
unit=32MB)
Storage.1.Health.SMART.Attributes.28.Value=1723
Storage.1.Health.SMART.Attributes.29.Id=F2
Storage.1.Health.SMART.Attributes.29.Name=Total LBA Read (each read
unit=32MB)
Storage.1.Health.SMART.Attributes.29.Value=1021
Storage.1.Health.SMART.Attributes.30.Id=F5
Storage.1.Health.SMART.Attributes.30.Name=Flash Write Sector Count
Storage.1.Health.SMART.Attributes.30.Value=5376
```

JSON RESPONSE

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

```
{
    "Storage": [
        {
            "Index": 1,
            "Health": {
                "WriteSpeed": 0,
                "ReadSpeed": 0,
                "PowerOnHours": 770,
                "TemperatureInCelsius": 59,
                "ValidSpareBlock": 100,
                "SpareBlockAlarm": "Good",
```

```
                "RemainingLifeInPercentage": 100,
                "LifetimeAlarm": "Good"
            },
            "SMART": {
                "Model": "TS2TMTS952T2",
                "SerialNumber": "H512150083",
                "FirmwareVersion": "02J0T6OC\n",
                "Attributes": [
                    {
                        "Index": 1,
                        "Id": "01",
                        "Name": "Read Error Rate",
                        "Value": 0
                    },
                    {
                        "Index": 2,
                        "Id": "05",
                        "Name": "Reallocated Sectors Count",
                        "Value": 0
                    },
                    {
                        "Index": 3,
                        "Id": "09",
                        "Name": "Power-On Hours",
                        "Value": 770
                    },
                    {
                        "Index": 4,
                        "Id": "0C",
                        "Name": "Power Cycle Count",
                        "Value": 36
                    },
                    {
                        "Index": 5,
                        "Id": "94",
                        "Name": "SLC Total Erase Count",
                        "Value": 148
                    },
                    {
                        "Index": 6,
                        "Id": "95",
```

```
                        "Name": "SLC Maximum Erase Count",
                        "Value": 13
                    },
                    {
                        "Index": 7,
                        "Id": "96",
                        "Name": "SLC Minimum Erase Count",
                        "Value": 0
                    },
                    {
                        "Index": 8,
                        "Id": "97",
                        "Name": "SLC Average Erase Count",
                        "Value": 1
                    },
                    {
                        "Index": 9,
                        "Id": "9F",
                        "Name": "DRAM one bit error count",
                        "Value": 0
                    },
                    {
                        "Index": 10,
                        "Id": "A0",
                        "Name": "Uncorrectable sectors count when
read/write",
                        "Value": 0
                    },
                    {
                        "Index": 11,
                        "Id": "A1",
                        "Name": "Number of Valid Spare Blocks",
                        "Value": 114
                    },
                    {
                        "Index": 12,
                        "Id": "A3",
                        "Name": "Number of Initial Invalid Blocks",
                        "Value": 34
                    },
                    {
```

```
                    "Index": 13,
                    "Id": "A4",
                    "Name": "TLC Total Erase Count",
                    "Value": 100
                },
                {
                    "Index": 14,
                    "Id": "A5",
                    "Name": "TLC Maximum Erase Count",
                    "Value": 4
                },
                {
                    "Index": 15,
                    "Id": "A6",
                    "Name": "TLC Minimum Erase Count",
                    "Value": 0
                },
                {
                    "Index": 16,
                    "Id": "A7",
                    "Name": "TLC Average Erase Count",
                    "Value": 0
                },
                {
                    "Index": 17,
                    "Id": "A8",
                    "Name": "Max Erase Count of Spec",
                    "Value": 3000
                },
                {
                    "Index": 18,
                    "Id": "A9",
                    "Name": "Remain Life (percentage)",
                    "Value": 100
                },
                {
                    "Index": 19,
                    "Id": "B1",
                    "Name": "Total Wear Level Count",
                    "Value": 0
                },
```

```
                    {
                        "Index": 20,
                        "Id": "B5",
                        "Name": "Total Program Fail Count",
                        "Value": 0
                    },
                    {
                        "Index": 21,
                        "Id": "B6",
                        "Name": "Total Erase Fail Count",
                        "Value": 0
                    },
                    {
                        "Index": 22,
                        "Id": "C0",
                        "Name": "Power-Off Retract Count",
                        "Value": 28
                    },
                    {
                        "Index": 23,
                        "Id": "C2",
                        "Name": "Temperature",
                        "Value": 59
                    },
                    {
                        "Index": 24,
                        "Id": "C3",
                        "Name": "Total Correctable Count",
                        "Value": 0
                    },
                    {
                        "Index": 25,
                        "Id": "C4",
                        "Name": "Reallocation Event Count",
                        "Value": 0
                    },
                    {
                        "Index": 26,
                        "Id": "C7",
                        "Name": "Ultra DMA CRC Error Count",
                        "Value": 0
```

```
                    },
                    {
                        "Index": 27,
                        "Id": "E8",
                        "Name": "Available Reserved Space",
                        "Value": 100
                    },
                    {
                        "Index": 28,
                        "Id": "F1",
                        "Name": "Total LBA Written (each write unit=32MB)",
                        "Value": 1723
                    },
                    {
                        "Index": 29,
                        "Id": "F2",
                        "Name": "Total LBA Read (each read unit=32MB)",
                        "Value": 1021
                    },
                    {
                        "Index": 30,
                        "Id": "F5",
                        "Name": "Flash Write Sector Count",
                        "Value": 5376
                    }
                ]
            }
        }
    ]
}
```

Chapter 40. LocalVMS

40.1. Description

The **localvms** submenu used to manage the vms application installed on the device.

> **NOTE**
> This chapter applies to cameras only
> Attribute to check: "attributes/System/Support/**LocalVMS**"

Access level

Action	Camera
view	Admin
control	Admin

40.2. Syntax

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

40.3. Parameters

Action	Parameters	Request/ Response	Type/ Value	Description
view	VmsName	RES	<enum> None, Wave	VMS application name, if not installed *None" is returned
	Version	RES	<string>	Version of the VMS application
	Status	RES	<enum> None, NotInstalled, Ready, Running, Installing, Stopped, Error	Status of the VMS application
	Location	RES	<string>	Parition where the VMS application is installed

Action	Parameters	Request/ Response	Type/ Value	Description
control	Mode	REQ	<enum> Install, Uninstall, Start, Stop	Operation to perform on the VMS **Note** For **Install** operation HTTP POST is used to send the installation file as binary octet stream.
	VmsName	REQ	<enum> Wave	Name of the vms application to be installed
	Location	REQ	<string>	Optional partition location where the application has to be installed. eg: /mnt/sda1

40.4. Examples

40.4.1. View the status of the installed application

REQUEST

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

TEXT RESPONSE

```
VmsName=Wave
Version=1.2XXX
Status=Ready
Location=/mnt/sda1
```

JSON RESPONSE

```
{
    "VmsName": "Wave",
    "Version": "1.2XXX",
    "Status": "Ready",
    "Location": "/mnt/sda1"
}
```

40.4.2. Install VMS Application

REQUEST [POST]

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

```
cgi/system.cgi?msubmenu=localvms&action=control&Mode=Install&VmsName=Wave&Location=/mnt/sda1
```

POST BODY

```
Content-Length: 180324956
Content-Type: application/octet-stream
```

```
Binary Data
```

TEXT RESPONSE

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

```
OK
```

JSON RESPONSE

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

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

40.4.2.1. Using Curl to install VMS

```
curl -v --digest -u user:password -H "Content-Type:application/octet-stream" --data-binary "@wave-server-5.0.0.34745-linux_arm64-beta.deb" -H "Expect:" “http://<Device IP>/stw-cgi/system.cgi?msubmenu=localvms&action=control&Mode=Install&VmsName=Wave&Location=/mnt/sda1”
```

40.4.3. Uninstall VSM Application

REQUEST

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=localvms&action=control&Mode=Uninstall
```

40.4.4. Start VMS Application

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=localvms&action=control&&Mode=Start
```

40.4.5. Stop VMS Application

```
http://<Device IP>/stw-
cgi/system.cgi?msubmenu=localvms&action=control&&Mode=Stop
```