

Grandstream Networks, Inc.

GSC36XX Grandstream IP Camera

HTTP API



Table of Contents

CHANGE LOG	5
Version 1.0.0.19.....	5
Version 1.0.0.18.....	5
Version 1.0.0.14.....	5
Version 1.0.0.13.....	5
Version 1.0.0.7.....	5
SUPPORTED DEVICES	6
OVERVIEW	7
Authentication	8
URL Parameter Definitions.....	10
XML Returned by Device.....	11
PARAMETERS	12
1. ISP parameter	12
2. Date and Time	12
3. Network Settings	17
4. Access Settings	20
5. User Management.....	22
6. SIP Settings	23
<i>Account Settings</i>	24
<i>SIP Advanced Settings</i>	26
7. White List.....	27
8. Video Settings	30
9. On Screen Display (OSD) Settings	33
10. CMOS Settings.....	34
11. Audio Settings.....	35
12. Privacy Masks	37
13. Alarm Config.....	38



14. Alarm Schedule Configuration	41
15. Alarm Action Configuration	42
16. Alarm Phone List	43
17. Email Settings.....	44
18. FTP Settings.....	45
19. Upgrade Configuration	45
20. Reboot & Reset	48
21. Syslog or Debug	48
22. Data Maintenance	49
23. Event Notification.....	49
24. Trusted CA Certificates.....	51
25. System Status	52
26. Network Status	53
27. Firmware Available Version Check.....	55
28. PTZ (GSC3620 only).....	55
28. TR069	56
28. Cloud Server Settings	57
HTTP API APPLICATION EXAMPLES.....	59
General Parameter Revision:	59
1. Authentication verification required First.....	59
2. Parameters can ONLY be adjusted after successful authentication verification.	61
Snapshots.....	62
MJPEG Stream (Encryption Mode or Basic Mode).....	66
MJPEG Challenge/Response Authentication at Default Mode.....	66
Obtain MJPEG stream at specified stream channel	68
MJPEG Basic Authentication For Default Method:	71
MJPEG Basic Authentication For Recommended Method:	72
MJPEG video or Snapshot image via HTTP API to 3 rd party System Integration	73
Anonymous Liveview/Snapshot/Stream:	73
Other Data API.....	74
HTTP SIP DIAL.....	74
REFERENCES	74





CHANGE LOG

This section documents significant changes from previous versions of the GSC36xx HTTP API. Only major new features or major document updates are listed here. Minor updates for corrections or editing are not documented here.

Version 1.0.0.19

- Added SIP URI Scheme When Using TLS P values: P2329/P2429/P2529/P2629. [Account Settings]
- Added SIP Instance ID P values: P288/P489/P589/P69. [Account Settings]

Version 1.0.0.18

- Added event notification HTTP Method P value: P15553 [23. Event Notification]

Version 1.0.0.14

- Update GMT-03:00 (Brazil, Sao Paulo) string. [Date and Time]
- This is the initial version for GSC3620.

Version 1.0.0.13

- Added tr069 P values: P1409/P4503/P4504/P4505/P4506/P4507/P4511/P4512/P4518 /P8220/P8221. [TR069]
- Added P8402 for Debug Log Protocol. [Syslog or Debug]
- Added P15518 for Multi Alarm Zones Motion Detection Config. [Alarm Config]
- Added P15517 for Motion Detection Mode. [Alarm Config]
- Added P15524 for Maximum Time to Identify a Valid Multi-Zone Alarm Detection (s). [Alarm Config]
- Added P15525 for Maximum Number of Alarming Zones for a Valid Multi-Zone Alarm Detection. [Alarm Config]
- Added P15519 for Minimum Number of Blocks Per Region to Trigger Detection. [Alarm Config]
- Added Cloud Server Settings. [Cloud Server Settings]
- Added Motion detection Status API. [Other Data API]

Version 1.0.0.7

- Initial Version



SUPPORTED DEVICES

The following table shows GSC36XX Grandstream products supporting HTTP API covered in this guide:

Model	Firmware
GSC3610	1.0.3.8 or higher
GSC3615	1.0.3.8 or higher
GSC3620	1.0.7.5 or higher



OVERVIEW

Grandstream GSC36xx Interface HTTP API (Application Programming Interface) supports RFC3550 (RTP).

This document explains in detail the external HTTP-based application programming interface and parameters of functions via the supported method. The HTTP API is firmware dependent. Please refer to the related firmware Release Note for the supported functions.

Administrator Privilege is required, and administrator authentication verification has to be executed before any operation to the related parameter configuration.

Syntax : **Client** ➔ **Server**

```
https://<servername>/goform/config?cmd=get&type=<module>
```

```
https://<servername>/goform/config?cmd=set&<param>=<value>&<param1>=<value>...
```

Notes:

- **<param>** is used to identify the different parameter of functions.
- **<module>** is used to identify the different module of functions.



Authentication

Authentication is using encrypted Challenge / Response mode. After successful authentication, the user with administration privilege needs to include the authentication information in each request header of the operation before the operation can be executed. The authentication process is shown as following:

Step 1:

Client → Server

```
https://<servername>/goform/login?cmd=login&user=admin&type=0
```

Server → Client

```
<?xml version="1.0"encoding="UTF-8" ?>
<Configuration>
<ResCode>0</ResCode>
<ChallengeCode>ChallengeString</ChallengeCode>
<LoginType>0</LoginType>
<RetMsg>OK</RetMsg>
</Configuration>
```

Step 2:

Client → Server

```
https://<servername>/goform/login?cmd=login&user=admin&authcode=<authcodestring>&type=0
```

Server → Client

```
HTTP/1.0 200 OK Content-Type: text/xml; charset=UTF-8
Set-Cookie: session=ac81cade4c9d9264f50c45018fb02c12;path=/;
Set-Cookie: uname=admin;path=/;
Set-Cookie: level=1;path=/;

<?xml version="1.0"encoding="UTF-8" ?>
<Configuration>
  <ResCode>0</ResCode>
  <LoginType>0</LoginType>
  <RetMsg>OK</RetMsg>
</Configuration>
```

Notes: The rule of authentication code string is generated via following rules:



- authcodestring = md5(ChallengeString:GSC36XXIZpRsFzCbM:password);
 - ChallengeString is the reply strings from Server by **Step 1**
 - Password is the login password of the GSC36xx device.
- The RED part of the string is the replied authentication information from Sever after successful authentication. Such information has to be existed in the operation message header of all the following operations, otherwise the operation is abnormal and will fail.

Example: Update SIP account SIP Server address to 192.168.80.22

Client → Server

```
POST /goform/config HTTP/1.1
Host: 192.168.80.24
Connection: keep-alive
Content-Length: 203
Cache-Control: max-age=0
Accept: application/xml, text/xml, */*; q=0.01
X-Requested-With: XMLHttpRequest
If-Modified-Since: 0
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/83.0.4103.116 Safari/537.36
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Origin: http://192.168.80.24
Referer: http://192.168.80.24/Pages/sip_basic1.html?t=1596680877093
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: session=017df60eae2612f12111c329c5fdf0d6; uname=admin; level=1;
gdsauthkey80=017df60eae2612f12111c329c5fdf0d6; curpage=sip_basic1.html
cmd=set&P47=192.168.80.22
```

Note: The **RED** part of the string is the Authentication String MUST be carried in the header message by the cookie during operation.

Tips of HTTP API usage:

- In most situation, parameters fetched by using “GET” can be revised by using “SET”, except Status or Display parameter information.



- When using “SET”, sometimes there are inter-action or restrictions among those related parameters, all parameters must “SET” or configure correctly together before the device taking effect.
- Please refer to related Web GUI page for reference if unsure about the HTTP API.

URL Parameter Definitions

<parameter>=<value>	Values	Description
cmd=<string>	add/del/set/get /reboot/reset/fw_upgrade /export/upload	<u>Operation command type (Required):</u> add: add parameter or data del: delete parameter or data set: set device parameter get: get command or parameter reboot: reboot device reset: factory reset device fw_upgrade: check FW version or upgrade FW



XML Returned by Device

Response Code	Response Text	Description
0	OK	Success
1		User does not exist
2		Password error
3		Maximum login or maximum white list number
4		Phone number exist
5		Parameter should not be empty
6		Parameter contain illegal characters
7		Message sending error
8		New password not match
9		FTP test error
10		Hostage password
11		Remote /Local PIN not same with hostage password
12		SMTP timeout
13		Retrieve Password (Account is no exist)
14		SMTP authentication failed
15		SMTP test failed
16		Invalid SMTP server
17		Email address is empty
18		Email test timeout
19		SIP is calling
401		Authentication failed
408		Session time out

Example:

Returned XML by device:

```
<?xml version="1.0"encoding="UTF-8" ?>
  <Configuration>
    <ResCode>0</ResCode>
    <RetMsg>OK</RetMsg>
  </Configuration>
```



PARAMETERS

1. ISP parameter

<parameter>=<value>	Values	Description
P15520=<int>	0 - 100	BRIGHTNESS
P15521=<int>	0 - 100	CONTRAST
P15522=<int>	0 - 100	SATURATION
P15523=<int>	0 - 100	SHARP

2. Date and Time

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=date

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P64>16</P64>
<P246></P246/>
<P5006>1</P5006>
<P30>pool.ntp.org</P30>
<P10006>1440</P10006>
<P10004>0</P10004>
<P10005></P10005/>
<P10008>0</P10008>
<P14040>2017</P14040>
<P14041>6</P14041>
<P14042>3</P14042>
<P14043>9</P14043>
<P14044>37</P14044>
<P14045>33</P14045>
</Configuration>
```

Set:



http://<servername>/goform/config?cmd=set<parameter>=<value>...

<parameter>=<value>	Values	Description
P144	<0 1>	Allow DHCP Option 42 to override NTP server
P64=<string>		Time Zone: auto: "Auto" TAZ+12: "GMT-12:00 (International Date Line West)" TAB+11: "GMT-11:00 (Midway Island, Samoa)" HAW10: "GMT-10:00 (US Hawaiian Time)" AKST9AKDT: "GMT-09:00 (US Alaska Time)" PST8PDT: "GMT-08:00 (US Pacific Time, Los Angeles)" PST8PDT,M3.2.0,M11.1.0: "GMT-08:00 (Baja California)" MST7MDT: "GMT-07:00 (US Mountain Time, Denver)" MST7: "GMT-07:00 (Mountain Time (Arizona, no DST))" MST7MDT,M4.1.0,M10.5.0: "GMT-07:00 (Chihuahua, La Paz, Mazatlan)" CST6CDT: "GMT-06:00 (Central Time)" CST+6: "GMT-06:00 (Central America)" CST6CDT,M4.1.0,M10.5.0: "GMT-06:00 (Guadalajara, Mexico City, Monterrey)" EST5EDT: "GMT-05:00 (Eastern Time)" EST5: "GMT-05:00 (Eastern Time without daylight saving)" TZf+4:30: "GMT-04:30 (Caracas)" AST4ADT: "GMT-04:00 (Atlantic Time)" AST4ADT,M3.2.0,M11.1.0: "GMT-04:00 (Atlantic Time (New Brunswick))"



NST+3:30NDT+2:30,M4.1.0/00:01:00,M10.5.0/00:01:00: "GMT-03:30 (Newfoundland Time)"
 TZK+3: "GMT-03:00 (Greenland)"
 BRT+3: "GMT-03:00 (Brazil, Sao Paulo)"
 UTC+3: "GMT-02:00 (Argentina)"
 TZL+2: "GMT-02:00 (Mid-Atlantic)"
 TZM+1: "GMT-01:00 (Azores, Cape Verde Is.)"
 TZN+0: "GMT (Edinburgh, Casablanca, Monrovia)"
 GMT+0BST-1,M3.5.0/01:00:00,M10.5.0/02:00:00: "GMT (London, Great Britain)"
 WET-0WEST-1,M3.5.0/01:00:00,M10.5.0/02:00:00: "GMT (Lisbon, Portugal)"
 GMT+0IST-1,M3.5.0/01:00:00,M10.5.0/02:00:00: "GMT (Dublin, Ireland)"
 CET-1CEST-2,M3.5.0/02:00:00,M10.5.0/03:00:00: "GMT+01:00 (Paris, Vienna, Warsaw, Roma, Madrid, Prague, Berlin, Budapest, Amsterdam, Belgium)"
 TZP-2: "GMT+02:00 (Israel, Cairo, Athens, Istanbul, Bucharest)"
 EET-2EEST-3,M3.5.0/03:00:00,M10.5.0/04:00:00: "GMT+02:00 (Helsinki, Athens, Tallinn)"
 EET-2EEST,M3.5.0/3,M10.5.0/4: "GMT+02:00 (Kyiv, Ukraine)"
 TZQ-3: "GMT+03:00 (Kuwait, Baghdad, Tehran, Nairobi)"
 MSK-3: "GMT+03:00 (Moscow, Russia)"
 TZR-4: "GMT+04:00 (Abu Dhabi, Baku)"
 TZS-5: "GMT+05:00 (Islamabad, Ekaterinburg, Karachi, Tashkent)"
 TZT-5:30: "GMT+05:30 (Chennai, New Delhi, Mumbai)"
 TZU-5:45", "GMT+05:45 (Kathmandu)"
 TZV-6: "GMT+06:00 (Almaty, Astana, Dhaka, Novosibirsk)"



		TZW-6:30: "GMT+06:30 (Rangoon)" TZX-7", "GMT+07:00 (Bankok, Hanoi, Krasnoyarsk)" WIB-7", "GMT+07:00 (Jakarta)" TZY-8", "GMT+08:00 (Beijing, Taipei, Kuala Lumpur, Irkutsk)" SGT-8", "GMT+08:00 (Singapore)" ULAT-8", "GMT+08:00 (Ulaanbaatar, Mongolia)" WST-8: "GMT+08:00 (Perth)" TZZ-9: "GMT+09:00 (Japan, Korea, Yakutsk)" CST-9:30CDT-10:30,M10.1.0/02:00:00,M4.1.0/03:00:00: "GMT+09:30 (Adelaide)" CST-9:30: "GMT+09:30 (Darwin)" TZb-10: "GMT+10:00 (Guam)" EST-10EDT-11,M10.1.0/02:00:00,M4.1.0/03:00:00": "GMT+10:00 (Hobart)" EST-10EDT-11,M10.5.0/02:00:00,M4.1.0/03:00:00: "GMT+10:00 (Sydney, Melbourne, Canberra)" EST-10: "GMT+10:00 (Brisbane)" TZc-11: "GMT+11:00 (Magadan, Solomon Is., New Caledonia)" NZST-12NZDT-13,M9.5.0/02:00:00,M4.1.0/03:00:00: "GMT+12:00 (Auckland, Wellington)" TZd-12: "GMT+12:00 (Fiji)" TZe-13: "GMT+13:00 (Nuku'alofa)" customize: "Self-Defined Time Zone"
P246=<string>		Self-Defined Time Zone



P5006=<int>	<0 1>	Enable NTP 0: Disable 1: Enable
P30=<string>		NTP Server
P10006=<int>	60 - 1440	Interval of Time Update (minute)



3. Network Settings

Support **cmd=get/set**

Get:

http://<servername>/goform/config?cmd=get&type=net

Example:

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

<Configuration>

<ResCode>0</ResCode>

<RetMsg>OK</RetMsg>

<P67>00:0B:82:AB:CC:BA</P67>

<P8>0</P8>

<P9>192</P9>

<P10>168</P10>

<P11>86</P11>

<P12>3</P12>

<P13>255</P13>

<P14>255</P14>

<P15>255</P15>

<P16>0</P16>

<P17>192</P17>

<P18>168</P18>

<P19>86</P19>

<P20>1</P20>

<P10107>0</P10107>

<P21>192</P21>

<P22>168</P22>

<P23>84</P23>
```



```

<P24>217</P24>

<P25>192</P25>

<P26>168</P26>

<P27>10</P27>

<P28>253</P28>

</Configuration>

```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P8=<int>	<0 1>	IP Address Mode 0: DHCP 1: Static IP
P9=<int>	0 - 255	IP Address: P9.P10.P11.P12
P10=<int>	0 - 255	
P11=<int>	0 - 255	
P12=<int>	0 - 255	
P13=<int>	0 - 255	Subnet Mask: P13.P14.P15.P16
P14=<int>	0 - 255	
P15=<int>	0 - 255	
P16=<int>	0 - 255	



P17=<int>	0 - 255	Gateway: P17.P18.P19.P20
P18=<int>	0 - 255	
P19=<int>	0 - 255	
P20=<int>	0 - 255	
P10107=<int>	<0 1>	DNS
P21=<int>	0 - 255	DNS Server 1 P21.P22.P23.P24
P22=<int>	0 - 255	
P23=<int>	0 - 255	
P24=<int>	0 - 255	
P25=<int>	0 - 255	DNS Server 2 P25.P26.P27.P28
P26=<int>	0 - 255	
P27=<int>	0 - 255	
P28=<int>	0 - 255	
P1684=<int>	0,1	Enable LLDP 0: Disable 1: Enable
P27004=<int>	0,1	Enable VLAN 0: Disable 1: Enable



P51=<int>	0-4094	Layer 2 QoS 802.1Q/VLAN Tag
P87=<int>		Layer 2 QoS 802.1p Priority Value
P5038=<int>	0-7	Layer 2 QoS 802.1p Priority Value
P5042=<int>	0-7	Layer 2 QoS 802.1p Priority Value for RTP Media

4. Access Settings

Support **cmd=get/set**

Get:

http://<servername>/goform/config?cmd=get&type=access

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P12054>0</P12054>
<P901>80</P901>
<P903>554</P903>
<P12056>60</P12056>
<P14832>5</P14832>
<P14834>5</P14834>
<P12052>1</P12052>
<P12053>1</P12053>
```



```
<P276>1</P276>

<P27006>22</P27006>

<P15512></P15512>

</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set<parameter>=<value>...

<parameter>=<value>	Values	Description
P12054=<int>	<0 1>	Web Access Mode 0: HTTP 1: HTTPS
P901=<int>	HTTP: 80 - 65535 HTTPS: 443 – 65535	Web Port
P903=<int>	554 – 65535	RTSP Port
P12056=<int>	3 – 60	Time Out for Login (minute)
P14832=<int>	3 – 10	Max. continuous errors for Login Lock
P14834=<int>	5 – 60	Login error lockout time (minute)
P15473=<int>	<0 1>	Disable Web Access 0: Disable 1: Enable
P12053=<int>	<0 1>	Enable UPnP Search 0: Disable 1: Enable



P15469=<int>	<0 1>	Enable Anonymous LiveView 0: Disable 1: Enable
P276=<int>	<0 1>	Enable SSH 0: Disable 1: Enable
P27006=<int>	22 – 65535	SSH Port
P15512=<string>		RTSP Password MIN length 1, MAX length is 32

5. User Management

Support **cmd=set**

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
oldpwd=<string>		Oldpwd: Old Password, Max. Length = 32
Newpwd=<string>		Newpwd: New Password Max. Length = 32
cfmnewpwd=<string>		cfmnewpwd: Confirmed New Password; Max. Length = 32
		e.g.: cmd=set&oldpwd=123&newpwd=admin&cfmnewpwd=admin



		Oldpwd、Newpwd、cfmnewpwd , Three must be set at the same time, and Newpwd must be the same as cfmnewpwd. Otherwise invalid.
P10124=<string>		Password Recover Email Address

6. SIP Settings

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=sip

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P499>0</P499>
<P407></P407>
<P402></P402>
<P2412></P2412>
<P403></P403>
<P2433></P2433>
<P404></P404>
<P405></P405>
<P409>0</P409>
<P76></P76>
<P412>60</P412>
<P413>5060</P413>
<P39>5004</P39>
<P10451>300</P10451>
<P448>0</P448>
<P8000>1</P8000>
<P8004>0</P8004>
<P2402>1</P2402>
<P2403>0</P2403>
<P1309>0</P1309>
```



```

<P10453>1</P10453>
<PipCallSave>1</PipCallSave>
<P10454>0</P10454>
<P8001>0</P8001>
<P8003>0</P8003>
<P411>0</P411>
<P443>0</P443>
<P15427>0</P15427>
<P15476>0</P15476>
<P424>100</P424>
<P406>N</P406>
<P281>Y</P281>
</Configuration>

```

Set:

http://<servername>/goform/config?cmd=set¶meter=<value>...

Account Settings

Account 1	Account 2	Account 3	Account 4		
<parameter> =<value>	<parameter> =<value>	<parameter> =<value>	<parameter> =<value>	Values	Description
P271=<int>	P401=<int>	P501=<int>	P601=<int>	<0 1>	Account Active (In Use). 0:No, 1:Yes
P210=<int>	P499=<int>	P599=<int>	P699=<int>	<0 1>	SIP Registration Status(Read Only)
P3=<string>	P407=<string>	P507=<string>	P607=<string>		Account Name Max. Length = 64
P47=<string>	P402=<string>	P502=<string>	P602=<string>		SIP Server Max. Length = 255
P48=<string>	P403=<string>	P503=<string>	P603=<string>		Outbound Proxy Max. Length = 255
P2333=<string>	P2433=<string>	P2533=<string> >	P2633=<string>		Backup Outbound Proxy Max. Length = 255
P103=<int>	P408=<int>	P508=<int>	P608=<int>	<0 1 2>	DNS Mode 0:A Record 1:SRV 2:NAPTR/SRV



P35=<string>	P404=<string>	P504=<string>	P604=<string>		SIP User ID Max. Length = 255
P36=<string>	P405=<string>	P505=<string>	P605=<string>		Authenticate ID Max. Length = 255
P34=<string>	P406=<string>	P506=<string>	P606=<string>		Authenticate password Max. Length = 255
P63=<int>	P409=<int>	P509=<int>	P609=<int>	<0 1 2>	TEL URI 0: Disabled 1: User = Phone 2: Enable
P32=<int>	P412=<int>	P512=<int>	P612=<int>	1 – 64800	Registration Expiration(m)
P2330=<int>	P2430=<int>	P2530=<int>	P2630=<int>	0 – 64800	Re-register before Expiration(s)
P40=<int>	P413=<int>	P513=<int>	P613=<int>	1 – 65535	Local SIP Port
P130=<int>	P448=<int>	P548=<int>	P648=<int>	<0 1 2>	SIP Transport 0: UDP 1: TCP 2: TLS/TCP
P15480=<int>	P8000=<int>	P15481=<int>	P15482=<int>	<0 1>	Stream 1: Stream 1 0: Stream 2
P81=<int>	P411=<int>	P511=<int>	P611=<int>	<0 1>	Unregister On Reboot 0: Disable 1: Enable
P52=<int>	P414=<int>	P514=<int>	P614=<int>	<0 1 2 3 4>	NAT Traversal 0:No, 1:Stun 2:Keep Alive 3:UPnP 4:Auto
P183=<int>	P443=<int>	P543=<int>	P643=<int>	<0 1 2>	Enable SRTP 0: Disable 1: Enable but not Forced 2: Enable and Forced
P198=<int>	P424=<int>	P524=<int>	P624=<int>	<100 102 129>	Special Feature 100: Standard 102: Broadsoft 129: Telefonica Spain
P2305=<int>	P2405=<int>	P2505=<int>	P2605=<int>	<0 1 2>	Outbound Proxy Mode 0:in route 1:not in route 2:always send to



P2492=<int>	P2392=<int>	P2592=<int>	P2692=<int>	<0 1 2>	Enable RTCP 0: Disable 1: RTCP 2: RTCP-XR
P293=<int>	P462=<int>	P562=<int>	P662=<int>	<96-127 & !=101>	H.264 Payload Type
P2347=<int>	P2447=<int>	P2547=<int>	P2647=<int>	<0 1>	Accept Incoming SIP from Proxy Only 0: Disable 1: Enable
P43064=<int>	P43065=<int>	P43066=<int>	P43067=<int>	<0 1>	UPnP NAT Traversal 0: Disable 1: Enable
P2329=<int>	P2429=<int>	P2529=<int>	P2629=<int>	<0 1>	SIP URI Scheme When Using TLS 0: sip 1: sips
P288=<int>	P489=<int>	P589=<int>	P689=<int>	<0 1>	Support SIP Instance ID 0: Disable 1: Enable
P37=<int>	P486=<int>	P586=<int>	P686=<int>	1 - 64	Voice Frames Per TX
P57=<int>	P451=<int>	P551=<int>	P651=<int>	<0 8 9 18 >	Preferred Vocoder 1 0: PCMU 8: PCMA 9: G.722
P58=<int>	P452=<int>	P552=<int>	P652=<int>	<0 8 9 18 >	Preferred Vocoder 2 0: PCMU 8: PCMA 9: G.722
P59=<int>	P453=<int>	P553=<int>	P653=<int>	<0 8 9 18 >	(Reserved) Preferred Vocoder 3 0: PCMU 8: PCMA 9: G.722
P2395=<int>	P2495=<int>	P2595=<int>	P2695=<int>	<0 1>	Enable Session Timer 0: Disable 1: Enable

SIP Advanced Settings

<parameter>=<value>	Values	Description
P76=<string>		STUN Server
P39=<int>	1 – 65535	Local SIP Port



P10451=<int>	0 – 65535	Auto On-Hook Timer (Seconds)
P29610=<int>	48-10000	Use Random Port
P280=<string>		SIP TLS Certificate
P279=<string>		SIP TLS Private Key
P281=<string>		SIP TLS Private Key Password
P10453=<int>	<0 1>	Enable Direct IP Call 0: Disable 1: Enable
P8001=<int>	<0 1>	Enable two-way SIP Calling 0: Disable 1: Enable
P8003=<int>	<0 1>	SIP Proxy Compatibility Mode 1: Disable 0: Enable
P957=<int>	<0 1>	SIP Packetization Compatibility Mode 1: Disable 0: Enable
P15476=<int>	<0 1>	Allow Reset Via SIP NOTIFY 0: Disable (default) 1: Enable

7. White List

Support **cmd= get/add/set/del**

Get:

http://<servername>/goform/config?cmd=get&type=sip

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10460>1111</P10460>
<P10460>2222</P10460>
<P10460>3333</P10460>
</Configuration>
```



Set:

http://<servername>/goform/config?cmd=set&P10460=num

Add:

http://<servername>/goform/config?cmd=add&P10460=num

Del:

http://<servername>/goform/config?cmd=del&P10460=num1,num2,num3, ...

Account 1 White List

<parameter>=<value>	Values	Description
P10410=<int>	<0 1>	Enable White List 0: Disable 1: Enable
P10411=<string>		Use (,) as separator when deleting multiple numbers.

Account 2 White List

<parameter>=<value>	Values	Description
P10454=<int>	<0 1>	Enable White List 0: Disable 1: Enable
P10460=<string>		Use (,) as separator when deleting multiple numbers.

Account 3 White List

<parameter>=<value>	Values	Description
P10420=<int>	<0 1>	Enable White List 0: Disable 1: Enable



P10421=<string>

Use (,) as separator when deleting multiple numbers.

Account 4 White List

<parameter>=<value>	Values	Description
P10430=<int>	<0 1>	Enable White List 0: Disable 1: Enable
P10431=<string>		Use (,) as separator when deleting multiple numbers.

Example:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
  <ResCode>0</ResCode>
  <RetMsg>OK</RetMsg>
</Configuration>
```



8. Video Settings

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=video

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10572>1</P10572>
<P12306>96</P12306>
<P12313>1</P12313>
<P12307>1025</P12307>
<P12315>4096</P12315>
<P12904>30</P12904>
<P12924>0</P12924>
<P12311>2</P12311>
<P12312>80</P12312>
<P12706>96</P12706>
<P12713>1</P12713>
<P12707>1022</P12707>
<P12708>512</P12708>
<P12709>25</P12709>
<P12710>0</P12710>
<P12711>2</P12711>
<P12712>80</P12712>
<P13106>96</P13106>
<P13113>1</P13113>
<P13107>0</P13107>
<P13108>256</P13108>
<P13109>30</P13109>
<P13110>0</P13110>
<P13111>2</P13111>
<P13112>80</P13112>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...



Stream 1		
<parameter>=<value>	Values	Description
P12306=<int>	<26 96>	Video Codec: 26: H.264 96: MJPEG
P12313	<0 1 2>	Profile 0: Baseline 1: Main Profile 2: High Profile
P12307=<int>	<1022 1025>	Resolution: 1025: 1920*1080 (16:9) 1023: 1280*960 (4:3) 1022: 1280*720 (16:9)
P12315=<int>	<1024 2048 4096>	Video Bit Rate (kbps) 512: 512kb 1024: 1024kb 2048: 2048kb 3072: 3072kb 4096: 4096kb 6144: 6144kb
P12904=<int>	<5 10 15 20 25 30>	Max. Frame Rate (fps) 5: 5fps 10: 10fps 15: 15fps 20: 20fps 25: 25fps 30: 30fps
P12924=<int>	<0 1>	Video Bit Rate Control 0: Constant Bit Rate (CBR) 1: Variable Bit Rate (VBR)
P12311=<int>	<0 1 2 3 4>	Image Quality 0: Very High 1: High 2: Normal 3: Low 4: Very Low
P12312=<int>	5 – 100	I-Frame Interval



Stream 2		
<parameter>=<value>	Values	Description
P12706=<int>	<26 96>	Video Codec: 26: H.264 96: MJPEG
P12713	<0 1 2>	Profile 0: Baseline 1: Main Profile 2: High Profile
P12707=<int>	<1022 1016 1 1009 0>	Resolution: 1022: 120*720 1016: 702*576 1: 640*480 (4:3) 1009: 352*288 0: 320*240 (QVGA)
P12708=<int>	Second resolution: 1280*720 74*576 640*480 <256 512 1024 2048 3072> Second resolution: 352*288 320*240 <112 114 160 256 512>	Video Bit Rate (kbps) Second resolution: 640*480 256: 256kb 512: 512kb 1024: 1024kb 2048: 2048kb 3072: 3072kb Second resolution: 352*288 or 320*240 112: 112kb 114: 114kb 160: 160kb 256: 256kb 512: 512kb
P12709=<int>	<5 10 15 20 25 30>	Max. Frame Rate (fps) 5: 5fps 10: 10fps 15: 15fps 20: 20fps 25: 25fps 30: 30fps
P12710=<int>	<0 1>	Video Bit Rate Control 0: Constant Bit Rate (CBR) 1: Variable Bit Rate (VBR)
P12711=<int>	<0 1 2 3 4>	Image Quality 0: Very High 1: High 2: Normal 3: Low 4: Very Low
P12712=<int>	5 – 100	I-Frame Interval



9. On Screen Display (OSD) Settings

Supported Methods: **cmd = get/set**

Get:

http://<servername>/goform/config?cmd=get&type=osd

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10044>1</P10044>
<P10045>0</P10045>
<P10001>1</P10001>
<P10007>0</P10007>
<P10040></P10040>
<P10041>0</P10041>
<P10046>0</P10046>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P10044=<int>	<0 1>	Display Time - 0: Disable; 1: Enable
P10045=<int>	<0 1>	Display Text 0: Disable 1: Enable
P10001=<int>	<0 1 2>	OSD Date Format 0:YYYY-MM-DD 1:MM-DD-YYYY 2:DD-MM-YYYY
P10040=<string>		OSD Text Max. Length = 32
P10041=<int>	<0 1 2 3>	OSD Position for Date/Time 0: Top Left 1: Bottom Left 2: Top Right 3: Bottom Right
P10046=<int>	<0 1 2 3>	OSD Position for Text 0: Top Left 1: Bottom Left 2: Top Right 3: Bottom Right



10. CMOS Settings

Supported Methods: **cmd = get/set**

Get:

<http://<servername>/goform/config?cmd=get&type=cmos>

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10572>1</P10572>
<P10573>0</P10573>
<P12314>1</P12314>
<P10503>0</P10503>
</Configuration>
```

Set:

<http://<servername>/goform/config?cmd=set&<parameter>=<value>...>

<parameter>=<value>	Values	Description
P15513=<int>	<0 1>	WDR (Wide Dynamic Range) 0: Disable 1: Enable
P12314=<int>	GSC3610/GSC3615: <0 1 2> GSC3620: <0 1>	Power Frequency (For Avoiding Video Flicker) GSC3610/GSC3615: 0: Indoor (50Hz Power Frequency) 1: Indoor (60Hz Power Frequency) 2: Outdoor GSC3620: 0: 50Hz 1: 60Hz



P10503=<int>	<0 30 60 125 250 500 1000 2000 5000 10000>	(Reserved fields) Shutter Speed 0: Auto 30: (1/30)s 60: (1/60)s 125: (1/125)s 250: (1/250)s 500: (1/500)s 1000: (1/1000)s 2000: (1/2000)s 5000: (1/5000)s 10000: (1/10000)s
P10500=<int>	<0 1 2 3>	Flip 0: NONE 1: Horizontal&Vertical 2: Horizontal 3: Vertical
P10504	<0 1 3>	IR CUT Settings 0: Manual 1: Automatic 3: Scheduled-Switch (For GSC3620 only)
P10505	<0 1>	Daytime/Night Model 0: Daytime 1: Night
P15552=<string>		Daytime Mode (for GSC3620 only): Format:start_time-end_time 0<=start_time<=23 0<=end_time<=23 Example: 08-22 means define Daytime from 08:00:00 to 22:00:00 Note: Start time and end time cannot be the same .

11. Audio Settings

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=audio

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
```



```
<RetMsg>OK</RetMsg>
<P14000>1</P14000>
<P14002>13</P14002>
<P14003>4</P14003>
<P14836>4</P14836>
<P14835>4</P14835>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P14000=<int>	<1 2>	Audio Codec 1: PCMU 2: PCMA
P14003=<int>	0 -6	Audio Out Volume
P14002=<int>	0 -6	Audio In Volume



12. Privacy Masks

Supported Methods: **cmd = get/set**

Get:

http://<servername>/goform/config?cmd=get&type=privacy

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P14225>0@0,0,0,0;0@0,0,0,0;0@0,0,0,0;0@0,0,0,0
</P14225>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set<parameter>=<value>...

<parameter>=<value>	Values	Description
P14225=<string>		<p>Region for Privacy Masks</p> <p><u>Format:</u></p> <p>Enable @Top Left x1 coordinates, Top Left y1 coordinates; Bottom Right x2 coordinates, Bottom Right y2 coordinates.</p> <p>e.g.: 0@0,0,0,0; 0@0,0,0,0; 0@0,0,0,0; 0@0,0,0,0;</p> <p>Digit before @ represents function, 0: Disable/Hide 1: Enable/Display</p> <p>Coordinates are separated by (,) Region is separated by (;)</p> <p>Please Do remember the last; and configure the 4 regions at the same time all at once.</p> <p>Rules: 0 <= x1, x2 <= 704; 0 <= y1, y2 <= 576; x1 <= x2; y1 <= y2</p>



13. Alarm Config

Support cmd= get/set

Get:

<http://<servername>/goform/config?cmd=get&type=event>

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
  <ResCode>0</ResCode>
  <RetMsg>OK</RetMsg>
  <P10250>0</P10250>
  <P14221>0</P14221>
  <P14222>1</P14222>
  <P14223>50</P14223>
  <P14224>[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0],[0,0,0,0]<
  /P14224>
  <P14320>0</P14320>
  <P14321>0</P14321>
  <P14322>1</P14322>
  <P14325>0</P14325>
  <P14326>0</P14326>
  <P14327>1</P14327>
  <P903>554</P903>
  <P3>admin</P3>
  <P2>123</P2>
  <P14200>schedule1,0@0000@2359,</P14200>
  <P14201>schedule2,0@0000@2359,</P14201>
  <P14202>schedule3,0@0000@2359,</P14202>
  <P14203>schedule4,0@0000@2359,</P14203>
  <P14204>schedule5,0@0000@2359,</P14204>
  <P14205>schedule6,0@0000@2359,</P14205>
  <P14206>schedule7,0@0000@2359,</P14206>
  <P14207>schedule8,0@0000@2359,</P14207>
  <P14208>schedule9,0@0000@2359,</P14208>
  <P14209>schedule10,0@0000@2359,</P14209>
  <P14300>profile1@0,0,0,0,0,0,</P14300>
  <P14301>profile2@0,0,0,0,0,0,</P14301>
  <P14302>profile3@0,0,0,0,0,0,</P14302>
  <P14303>profile4@0,0,0,0,0,0,</P14303>
  <P14304>profile5@0,0,0,0,0,0,</P14304>
  <P14305>profile6@0,0,0,0,0,0,</P14305>
  <P14306>profile7@0,0,0,0,0,0,</P14306>
  <P14307>profile8@0,0,0,0,0,0,</P14307>
  <P14308>profile9@0,0,0,0,0,0,</P14308>
  <P14309>profile10@0,0,0,0,0,0,</P14309>
  <P10467>0</P10467>
  <P10466>N</P10466>
  <P14341>5</P14341>
  <P10468>1</P10468>
  <P10469></P10469>
  <P14350>0</P14350>
  <P14355>1</P14355>
  <P14354></P14354>
  <P14825>1</P14825>
  <P14826>1</P14826>
  <P15407>1</P15407>
  <P15408>1</P15408>
</Configuration>
```

Set:



http://<servername>/goform/config?cmd=set<parameter>=<value>...

<parameter> =<value>	Values	Description
P14224=<string>		<p>Region of Motion Detection</p> <p><u>Format:</u></p> <p>[area Enable, Top Left x1 coordinates, Top Left y1 coordinates, Bottom Right x2 coordinates, Bottom Right y2 coordinates, Sensitivity]</p> <p>e.g.:</p> <p>[0,1,5,7,15,50],[1,11,8,16,17,50],[1,19,9,26,19,50],[0,0,0,0,0,50], [0,0,0,0,0,50],[0,0,0,0,0,50],[0,0,0,0,0,50]</p> <p>Coordinates are separated by (,) Region is separated by (;)</p> <p>The 8 regions must be configured at the same time all at once.</p> <p>Rules: Enable: 0/1 0 <= x1, x2 <= 704; 0 <= y1, y2 <= 576; x1 <= x2; y1 <= y2 Sensitivity: 1 - 100</p>
Pget=<string>		<p>Multi Region of Motion Detection</p> <p><u>Format:</u></p> <p>[area Enable, Top Left x1 coordinates, Top Left y1 coordinates, Bottom Right x2 coordinates, Bottom Right y2 coordinates, Sensitivity]</p> <p>e.g.:</p> <p>[0,1,5,7,15,50],[1,11,8,16,17,50],[1,19,9,26,19,50],[0,0,0,0,0,50], [0,0,0,0,0,50],[0,0,0,0,0,50],[0,0,0,0,0,50]</p> <p>Coordinates are separated by (,) Region is separated by (;)</p>



		<p>The 8 regions don't need to be configured at the same time all at once.</p> <p>Rules: Enable: 0/1 $0 \leq x1, x2 \leq 32; 0 \leq y1, y2 \leq 24;$ $x1 \leq x2; y1 \leq y2$ Sensitivity: 1 – 100</p> <p>NOTE: This must be a valid alarm region and exist in P14224</p>
P15517=<int>	<0 1 2 3>	<p>Motion Detection Mode</p> <p>0: Default Alarm Mode 1: Sequence Alarm Mode 2: Multi-zone Combined Alarm Mode 3: Zone Trigger Ratio Alarm Mode</p>
P15524=<int>	5 - 60	Maximum Time to Identify a Valid Multi-Zone Alarm Detection (s)
P15525=<int>	1 - 4	<p>Maximum Time to Identify a Valid Multi-Zone Alarm Detection (s)</p> <p>Note: The number cannot be bigger than the number of Alarm Zone.</p>
P15519=<int>	1 - 100	Minimum Number of Blocks Per Region to Trigger Detection (valid only in night mode)
P10267	<200 300 400 500 1000 2000 3000>	Minimum duration required by motion detection trigger(s)
P10268	<3 5 10 15 20 25 30 35 40 45 50 55 60>	Minimum interval of different motion detection event(s)
P14221=<int>	<0 1 2 3 4 5 6 7 8 9 10>	<p>Select Alarm Schedule</p> <p>0: All Day 1: Schedule1 2: Schedule2 3: Schedule3 4: Schedule4 5: Schedule5 6: Schedule6 7: Schedule7 8: Schedule8 9: Schedule9 10: Schedule10</p>



P14222=<int>	GSC3610/ GSC3615/ GSC3620	<1 2 3 4 5 6 7 8 9 10>	Alarm Action Profile 1: profile1 2: profile2 3: profile3 4: profile4 5: profile5 6: profile6 7: profile7 8: profile8 9: profile9 10: profile10
--------------	---------------------------------	------------------------	--

14. Alarm Schedule Configuration

Support cmd= get/set

Get:

http://<servername>/goform/config?cmd=get&type=alarm_schedule

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
  <ResCode>0</ResCode>
  <RetMsg>OK</RetMsg>
  <P14200>schedule1,0@0000@2359,</P14200>
  <P14201>schedule2,0@0830@1459,0@1500@1820,</P14201>
  <P14202>schedule3,0@0000@2359,</P14202>
  <P14203>schedule4,0@0000@2359,</P14203>
  <P14204>schedule5,0@0000@2359,</P14204>
  <P14205>schedule6,0@0000@2359,</P14205>
  <P14206>schedule7,0@0000@2359,</P14206>
  <P14207>schedule8,0@0000@2359,</P14207>
  <P14208>schedule9,0@0000@2359,</P14208>
  <P14209>schedule10,0@0000@2359,</P14209>
</Configuration>
```

Set:

<http://<servername>/goform/config?cmd=set¶meter=<value>...>

<parameter> =<value>	Values	Description
P14200=<string> P14201=<string> P14202=<string> P14203=<string> P14204=<string>		e.g.: <P14201>schedule2,0@0830@1459,0@1500@1820,</P14201>



P14205=<string>
 P14206=<string>
 P14207=<string>
 P14208=<string>
 P14209=<string>

schedule2 is a revisable name for a time period, Max. Length = 20

1st @ represent date of the week, 0-6, Sunday to Saturday
 2nd @ the pre-four digits represent start time of that period, 0830 represent 08:30 in 24 hour format;
 2nd @ the post-four digits represent end time of that period, 1459 represent 14:59 in 24 hour format.

NOTE:

There should be **NO** overlap in any time period configured in the same day.

(,) should be used to separate the different time period configured.

15. Alarm Action Configuration

Support cmd= get/set

Get:

http://<servername>/goform/config?cmd=get&type=alarm_profile

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P14827>0</P14827>
<P14300>profile1@0,0,0,0,0,0,0</P14300>
<P14301>profile2@0,0,0,0,0,0,0</P14301>
<P14302>profile3@0,0,0,0,0,0,0</P14302>
<P14303>profile4@0,0,0,0,0,0,0</P14303>
<P14304>profile5@0,0,0,0,0,0,0</P14304>
<P14305>profile6@0,0,0,0,0,0,0</P14305>
<P14306>profile7@0,0,0,0,0,0,0</P14306>
<P14307>profile8@0,0,0,0,0,0,0</P14307>
<P14308>profile9@0,0,0,0,0,0,0</P14308>
<P14309>profile10@0,0,0,0,0,0,0</P14309>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P14300=<string> P14301=<string>		e.g.: <P14300>profile1@0,0,0,0,0,0,0</P14300>



P14302=<string> P14303=<string> P14304=<string> P14305=<string> P14306=<string> P14307=<string> P14308=<string> P14309=<string>		Profile1 is revisable name of alarm action, Max. Length = 20 1 st digit after @ represents whether "Upload to Alarm Center": 0: Disable 1: Enable 2 nd digit after @ represents whether Constant 0 3 rd digit after @ represents whether "Voice Alarm to SIP Phone" 0: Disable 1: Enable 4 th digit after @ represents whether "Send Email" 0: Disable 1: Enable 5 th digit after @ represents whether "Upload JPEG" Snapshot 0: Disable 1: Enable
--	--	--

16. Alarm Phone List

Support cmd= get/add/del

Get:

http://<servername>/goform/config?cmd=get&type=sip

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10459>111</P10459>
<P10459>aaa</P10459>
</Configuration>
```

Add:

http://<servername>/goform/config?cmd=add&P10459=num1,num2, ...

<parameter>=<value>	Values	Description
P10459=<string>		Alarm Phone

Del:

http://<servername>/goform/config?cmd=del&P10459=num1,num2,num3, ...

<parameter>=<value>	Values	Description
P10459=<string>		(,) used as a separator when multiple alarm numbers deleted.



17. Email Settings

Support cmd= get/set

Get:

http://<servername>/goform/config?cmd=get&type=smtp

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10120>0</P10120>
<P10121>smtp.google.com</P10121>
<P10122>25</P10122>
<P10123>kevin@google.com</P10123>
<P10124>kevin@google.com</P10124>
<P14348>kevin@google.com</P14348>
<P14349>kevin@google.com</P14349>
<P10127>kevin</P10127>
<P10129>0</P10129>
<P10128r>123</P10128r>
<P10128>Y</P10128>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P10121=<string>		SMTP Mail Server
P10122=<string>		Port of SMTP Server
P10123=<string>		Sender's email address. Max. Length = 127
P10127=<string>		Sender's User Name, Max. Length = 63
P10128=<string>		Sender's email password, Max. Length = 63
P14348=<string>		Alarm to email address 1, Max. Length = 127
P14349=<string>		Alarm to email address 2, Max. Length = 127
P10129=<int>	<0 1>	SSL Encryption 0: Disable 1: Enable



18. FTP Settings

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=ftp

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P10140>1</P10140>
<P14113>0</P14113>
<P14114>192.168.1.10</P14114>
<P14115>21</P14115>
<P14116>kevin</P14116>
<P14118>/</P14118>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P14113=<int>	<0 1>	Storage Server Type 0: FTP 1: Central Storage (GSCManager Configure)
P14114=<string>		FTP Server IP or FQDN, Max. Length = 255
P14115=<int>		FTP Server Port, Max. Length = 5
P14116=<string>		FTP User Name, Max. Length = 63
P14118=<string>		FTP Path, Max. Length = 255

19. Upgrade Configuration

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=upgrade

Example:

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



```

<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P6767>1</P6767>
<P192>fm.grandstream.com/gs</P192>
<P6768></P6768>
<P232></P232>
<P233></P233>
<P212>1</P212>
<P237>fm.grandstream.com/gs</P237>
<P1360></P1360>
<P234></P234>
<P235></P235>
<P6769></P6769>
<P1361></P1361>
<P1359></P1359>
<version_limit>1.0.3.1</version_limit>
<P193>10080</P193>
<P194>1</P194>
<P8463>1</P8463>
<P145>0</P145>
<P1414>0</P1414>
</Configuration>

```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P6767=<int>	<0 1 2>	Firmware Upgrade Method 0: TFTP 1: HTTP 2: HTTPS
P192=<string>		Firmware Server Path
P6768=<string>		Firmware HTTP/HTTPS User Name
P6769=<string>		Firmware HTTP/HTTPS Password
P232=<string>		Firmware Upgrade File Prefix
P233=<string>		Firmware Upgrade File Postfix
P193=<int>	60 - 525600	Automatic Upgrade Interval (Minutes)
P194=<int>	<0 1 2 3>	Automatic Upgrade 0: No 1: Yes, check for periodic 2: Yes, check for every day 3: Yes, check for every week
P212=<int>	<0 1 2>	Config Upgrade Method



		0: TFTP 1: HTTP 2: HTTPS
P237=<string>		Config Server Path
P1360=<string>		Config HTTP/HTTPS User Name
P1361=<string>		Config HTTP/HTTPS Password
P234=<string>		Config Upgrade File Prefix
P235=<string>		Config Upgrade File Postfix
P1359=<string>		XML Config File Password
P145=<int>	<0 1>	DHCP Option 66 Override Server 0: Disable 1: Enable
P1414=<int>	<0 1>	Zero Config (for UCM Provisioning) 0: Disable 1: Enable
P8458=<int>	<0 1>	Randomized Automatic Upgrade 0: Disable 1: Enable
P285=<int>	<0-23>	Hour of the Day – Start Time
P8459=<int>	<0-23>	Hour of the Day – End Time
P286=<int>	<0-6>	Day of the Week – Week
P1411=<int>	<0 1>	DHCP Option 120 Override SIP Server 0: Disable 1: Enable
P8463=<int>	<0 1>	Validate Server Certificates 0: Disable 1: Enable



20. Reboot & Reset

Support **cmd= reboot/reset**

Reboot:

http://<servername>/goform/config?cmd=reboot

Reset:

http://<servername>/goform/config?cmd=reset&P12055=<value>

<parameter>=<value>	Values	Description
P12055=<int>	<0 1 2 3>	Type of Reset Operation 0: Clear All Data 1: Retain Network Data Only 2: Retain Only Card Information 3: Retain Network Data and Card Information

21. Syslog or Debug

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=debug

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
  <ResCode>0</ResCode>
  <RetMsg>OK</RetMsg>
  <P207>192.168.1.10</P207>
  <P208>0</P208>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...



<parameter>=<value>	Values	Description
P8402=<int>	<0 1>	Debug Log Protocol 0: UDP 1: SSL/TLS
P207=<string>		Debug Log Server IP or FQDN
P208=<int>	<0 1 2 3 4>	Debug Log Level: 0: None 1: Debug 2: Info 3: Warning 4: Error

22. Data Maintenance

Support cmd=export/upload

Export:

http://<servername>/goform/config?cmd=export&type=0&data_type=0

<parameter>=<value>	Values	Description
data_type=<int>	<0 1>	The type of system config file. 0: System Config Data(No include password) 1: System Config Data(Include password)

Upload:

http://<servername>/goform/config?cmd=upload&type=1

23. Event Notification

Support cmd= get/set

Get:

http://<servername>/goform/config?cmd=get&type=log



Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P15410>1</P15410>
  <P15413>http://192.168.1.2:80/</P15413>
  <P15414>admin</P15414>
  <P15415>admin</P15415>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set<parameter>=<value>...

<parameter>=<value>	Values	Description
P15410=<int>	<0 1>	Enable Event Notification 0: Disable 1: Enable
P15417=<int>	<1 2>	Via Type 1:HTTP 2:HTTPS
P15553=<int>	<0 1>	HTTP Method 0: POST 1: GET
P15413=<string>		HTTP Server URL: Format ip:port/path . Example: 192.168.1.2:80/ Max. Length=256
P15414=<string>		HTTP Server Username: If don't need the Username & Password to access the HTTP server, please keep empty. Max. Length=128
P15415=<string>		HTTP Server Password: If don't need the Username & Password to access the HTTP server, please keep empty. Max. Length=128



P15416=<string>	URL Template Default: { "mac": "\${MAC}", "content": "\${WARNING_MSG}" }
-----------------	--

24. Trusted CA Certificates

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type=trustedca

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P8433></P8433>
<P8433></P8434>
<P8433></P8435>
<P8433></P8436>
<P8433></P8437>
<P8433></P8438>
<P8475></P8475>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set¶meter=<value>...

<parameter>=<value>	Values	Description
P8433=<string> P8434=<string> P8435=<string> P8436=<string> P8437=<string> P8438=<string>		Trusted CA Certificates
P8475=<string>		Custom Certificate



25. System Status

Support cmd= get

Get:

http://<servername>/goform/config?cmd=get&type=sysinfo

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P89>GSC3610</P89>
<P917>V1.5A</P917>
<P1397>9650001415A</P1397>
<P70>1.0.3.1</P70>
<P45>1.0.3.1</P45>
<P68>1.0.3.1</P68>
<P15009>1 hour 31 minutes</P15009>
<P499>0</P499>
</Configuration>
```

<parameter>=<value>	Values	Description
P89=<string>		Product Model
P208=<int>	<0 1 2 3 4>	Hardware Version
P1397=<string>		Part Number
P69=<string>		Boot Version
P70=<string>		Core Version
P45=<string>		Base Version
P68=<string>		Program Version
P15009=<string>		System Up Time
Pfw_available_version = <string>		Firmware available version



26. Network Status

Support cmd= get

Get:

http://<servername>/goform/config?cmd=get&type=net

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P67>00:0B:82:AB:CC:BA</P67>
<P8>0</P8>
<P9>192</P9>
<P10>168</P10>
<P11>86</P11>
<P12>3</P12>
<P13>255</P13>
<P14>255</P14>
<P15>255</P15>
<P16>0</P16>
<P17>192</P17>
<P18>168</P18>
<P19>86</P19>
<P20>1</P20>
<P10107>0</P10107>
<P21>192</P21>
<P22>168</P22>
<P23>84</P23>
<P24>217</P24>
<P25>192</P25>
<P26>168</P26>
<P27>10</P27>
<P28>253</P28>
</Configuration>
```

<parameter>=<value>	Values	Description
P8=<int>	<0 1>	IP Address Mode 0: DHCP 1: Static IP
P9=<int>	0 - 255	IP Address:
P10=<int>	0 - 255	



P11=<int>	0 - 255	P9.
P12=<int>	0 - 255	P10.
		P11.
		P12.
P13=<int>	0 - 255	Subnet Mask:
P14=<int>	0 - 255	P13.
P15=<int>	0 - 255	P14.
		P15.
P16=<int>	0 - 255	P16.
P17=<int>	0 - 255	Gateway:
P18=<int>	0 - 255	P17.
P19=<int>	0 - 255	P18.
		P19.
P20=<int>	0 - 255	P20.
P10107=<int>	<0 1>	DNS
P21=<int>	0 - 255	DNS Server 1
P22=<int>	0 - 255	P21.
P23=<int>	0 - 255	P22.
		P23.
P24=<int>	0 - 255	P24.
P25=<int>	0 - 255	DNS Server 2
P26=<int>	0 - 255	P25.
P27=<int>	0 - 255	P26.
		P27.
P28=<int>	0 - 255	P28.



27. Firmware Available Version Check

Support cmd= fw_upgrade

fw_upgrade:

http://<servername>/goform/config?cmd=fw_upgrade&type=num

<parameter>=<value>	Values	Description
type=<int>	<0 1>	Operate type 0: check firmware available version 1: firmware upgrade

28. PTZ (GSC3620 only)

Support cmd= get/set

Get:

http://<servername>/goform/config?cmd=get&type=ptz

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<Pzoom_m>2</Pzoom_m>
</Configuration>
```

Note:

Pzoom_m	PTZ zoom factor value: 0 – 30 zoom_factor = 1+value/10, mean that if the value is 2, zoom_factor = 1+2/10=1.2
---------	---

Set:

http://<servername>/goform/config?cmd=set&type=ptz&act=<action>&opt=<option>

<parameter>=<value>	Values	Option	Description
---------------------	--------	--------	-------------



Zoom+	<0 1>	9	PTZ zoom+ action: 0: stop zoom+ 1: start zoom+
Zoom-	<0 1>	10	PTZ zoom- action: 0: stop zoom- 1: start zoom-
Focus+	<0 1>	11	PTZ focus+ action: 0: stop focus+ 1: start focus+
Focus-	<0 1>	12	PTZ focus- action: 0: stop focus- 1: start focus-
One key focus		20	PTZ one key focus

28. TR069

Support cmd= get/set

Get:

<http://<servername>/goform/config?cmd=get&type=tr069>

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P1409>1</P1409>
<P4503>https://acs.gdms.cloud/</P4503>
<P4504/>
<P4505>N</P4505>
<P4506>1</P4506>
<P4507>86400</P4507>
<P4511>C074AD1DB1F9</P4511>
<P4512>Y</P4512>
<P4518>7547</P4518>
<P8220/>
<P8221/>
</Configuration>
```

Set:



http://<servername>/goform/config?cmd=set<parameter>=<value>...

<parameter>=<value>	Values	Description
P1409=<int>	0,1	Enable TR-069 0- disable enable
P4503=<string>		ACS URL
P4504=<string>		ACS User Name
P4505=<string>		ACS Password
P4506=<int>	0,1	Periodic Inform Enable 0-disable 1-enable
P4507=<int>	1-4294967295	Periodic Inform Interval (s)
P4511=<string>		Connection Request User Name
P4512=<string>		Connection Request Password
P4518=<int>	1-65535	Connection Request Port
P8220=<string>		CPE Cert File
P8221=<string>		CPE Cert Key

28. Cloud Server Settings

Support **cmd= get/set**

Get:

http://<servername>/goform/config?cmd=get&type= neccloud

Example:

```
<?xml version="1.0" encoding="utf-8"?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
<P63010>0</P63010>
<P63012>0</P63012>
```



```
<P63013>0</P63013>
</Configuration>
```

Set:

http://<servername>/goform/config?cmd=set&<parameter>=<value>...

<parameter>=<value>	Values	Description
P63010=<int>	<0 1>	Enable Cloud Server: 0: Disable 1: Enable
P63012=<string>	string	Token: MAX Length is 1024
P63013=<int>	<0 1>	Stream: 0: Stream 1 1: Stream 2



HTTP API Application Examples

Now we provide three examples for reference:

- 1) General parameter revision, e.g.: change the "Unlocking Latency"
- 2) Obtain Snapshot
- 3) Obtain MJPEG stream

General Parameter Revision:

1. Authentication verification required First

Login Authentication: The authentication is using Challenge / Response encryption mode. After the successful authentication by the administrator, each later on operations will be accepted and successful ONLY when the correct authentication information is included inside the HTTP/HTTPS request header.

The authentication process is as following:

Step 1:

Client → Server (**Note:** *type=0 means login authentication verification required*)

https://<servername>/goform/login?cmd=login&user=admin&type=0

The Wireshark Capture Listed Below:

POST /goform/login HTTP/1.1

Host: 192.168.86.3

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:51.0) Gecko/20100101 Firefox/51.0

Accept: application/xml, text/xml, */*; q=0.01

Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

If-Modified-Since: 0

X-Requested-With: XMLHttpRequest

Referer: http://192.168.1.3/login.html

Content-Length: 27

Connection: keep-alive

cmd=login&user=admin&type=0



Server → Client (Server sending ChallengeCode to Client after receiving the request from Client)

HTTP/1.0 200 OK Content-Type: text/xml;charset=UTF-8

```
<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
  <ResCode>0</ResCode>
  <ChallengeCode>6243b032c7468a9e384e49933914e880</ChallengeCode>
  <LoginType>0</LoginType>
  <RetMsg>OK</RetMsg>
</Configuration>
```

Step 2:

Client → Server

(Client received ChallengeCode, via **authcodestring** =md5

(**ChallengeCode:GSC36XXIZpRsFzCbM:password**);

generating the **authcodestring**. Then sending this authcode string back to the Server)

https://<servername>/goform/login?cmd=login&user=admin&authcode=<authcodestring>&type=0

The Wireshark Capture Listed Below:

POST /goform/login HTTP/1.1

Host: 192.168.86.3

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:51.0) Gecko/20100101 Firefox/51.0

Accept: application/xml, text/xml, */*; q=0.01

Accept-Language: zh-CN, zh; q=0.8,en-US; q=0.5,en; q=0.3

Accept-Encoding: gzip, deflate

Content-Type: application/x-www-form-urlencoded; charset=UTF-8

If-Modified-Since: 0

X-Requested-With: XMLHttpRequest

Referer: http://192.168.1.3/login.html

Content-Length: 69

Connection: keep-alive

cmd=login&user=admin&authcode=bc8201d7a16262366aaf9d72b6b67f06&type=0



Note: The rule for authcodestring listed below:

authcodestring = md5(ChallengeCode:GSC36XXIZpRsFzCbM:password);

- The “ChallengeCode” is the returned string from Server in **Step 1**.
- password is the web login password of the device (GSC36xx)
- Authcode string must be length of 32 in lowercase hex as in MD5 hash.

Server → Client

```
HTTP/1.0 200 OK
Content-Type: text/xml;charset=UTF-8
Set-Cookie: session= bc8201d7a16262366aaf9d72b6b67f06;path=/;
Set-Cookie: uname=admin; path=/;
Set-Cookie: level=1; path=/;

<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
<ResCode>0</ResCode>
<LoginType>0</LoginType>
<RetMsg>OK</RetMsg>
</Configuration>
```

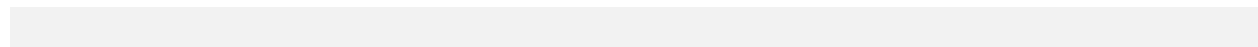
Notes:

- The **RED** part is the authentication information returned by the Server after successful authentication.
- This part of the information must exist in the message header before all later on operations can run normally.

2. Parameters can ONLY be adjusted after successful authentication verification.

Example: Adjust “Unlocking Latency” and “Unlock Hold Time” via HTTP API:

Client → Server



```
POST /goform/config HTTP/1.1
Host: 192.168.86.3
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:51.0) Gecko/20100101 Firefox/51.0
Accept: application/xml, text/xml, */*; q=0.01
Accept-Language: zh-CN, zh; q=0.8, en-US; q=0.5,en; q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
If-Modified-Since: 0
X-Requested-With: XMLHttpRequest
Referer: http://192.168.86.3/Pages/access.html?t=1496977082292
Content-Length: 299
Cookie: session= bc8201d7a16262366aaf9d72b6b67f06; uname=admin; level=1; GSCauthkey80=
bc8201d7a16262366aaf9d72b6b67f06; curpage=access.html
Connection: keep-alive

cmd=set&P14100=2&P14101=5
```

Notes:

- The “P14100=2” means adjusting “Unlocking Latency” to 2 seconds.
- The “P14101=5” means adjusting “Unlock Hold Time” to 5 seconds.
- The above **RED** part “Cookie” is the returned verification message of the successful authentication in Login Verification at **Step 2**.

Snapshots

Login Authentication: The authentication is using Challenge / Response encryption mode. After the successful authentication by the administrator, each later on operations will be accepted and successful ONLY when the correct authentication information is included inside the request header.

The authentication process is as following:

Step 1:

Note: type=1 means MJPEG/JPEG obtain login authentication verification)

Client ➔ Server



`https://<servername>/goform/login?cmd=login&user=admin&type=1`

Server → Client

```
<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
  <ResCode>0</ResCode>
  <ChallengeCode>ChallengeString</ChallengeCode>
  <LoginType>0</LoginType>
  <RetMsg>OK</RetMsg>
</Configuration>
```

Step2:

Client → Server

`https://<servername>/goform/login?cmd=login&user=admin&authcode=<authcodestring>&type=1`

Notes: The rule for authcodestring listed below:

- authcodestring = md5(ChallengeString:GSC36XXIDyTIHwNgZ:password);
- The “ChallengeString” is the returned string from Server in **Step 1**.
- password is the login password of the device (GSC36xx).
- Authcode string must be length of 32 in lowercase hex as in MD5 hash.

Server → Client

HTTP/1.0 200 OK Content-Type: text/xml;charset=UTF-8
Set-Cookie: mjpeg_sess=396a6328e70f3b1b984f7d6f08159a49;path=/
Set-Cookie: mjpeg_uname=admin;path=/
Set-Cookie: mjpeg_level=1;path=/;

```
<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
  <ResCode>0</ResCode>
  <RetMsg>OK</RetMsg>
</Configuration>
```

Notes:

- The **RED** part is the authentication message returned by the Server after successful authentication.
- This part of the message must exist in the message header before all later on operations can run successfully.



Step3:

Client → Server

```
http(s)://<servername>/snapshot/view.jpg
```

Notes:

- **Step 3** should be finished within **30 seconds** after authentication OK, otherwise, the authentication will be invalid.
- The timer that records authentication valid time will refresh if **Step 3** executes successfully.
- The timer when using <https://<servername>/snapshot/view.jpg> to obtain a snapshot will start counting from 0, within 30 seconds if the same request asking for snapshot again, the timer will reset; otherwise, the request without verified message will be timed out and the verification will be void. New verification is required when obtain snapshot again.
- The snapshot is using an internal 4th stream, with MJPEG video codec and fixed resolution 1280 x 720

Below is an Example:

Step 1:

Client → Server

(Client sending login request to Server, trying to get ChallengeCode)

```
POST /goform/login HTTP/1.1
Host: 192.168.86.3
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:51.0)
Gecko/20100101 Firefox/51.0
Accept: application/xml, text/xml, */*; q=0.01
Accept-Language: zh-CN, zh; q=0.8,en-US; q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
If-Modified-Since: 0
X-Requested-With: XMLHttpRequest
Referer: http://192.168.86.3/jpeg/mjpeg.html
Content-Length: 27
Connection: keep-alive

cmd=login&user=admin&type=1
```

Step 2:

Server → Client

(Server sending ChallengeCode to Client after receiving the request from Client)




```
HTTP/1.0 200 OK
Content-Type: text/xml; charset=UTF-8

<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
<ResCode>0</ResCode>
<ChallengeCode>clcd724b1fc2d552652bba09a56f6f3c</ChallengeCode>
<RetMsg>OK</RetMsg>
</Configuration>
```

Step 3:

Client → Server

Client receiving ChallengeCode, via authcodestring =

md5(ChallengeCode:GSC36XXIDyTIHwNgZ:password); generating authcodestring, then sending this authcodestring back to Server)

```
POST /goform/login HTTP/1.1
Host: 192.168.86.3
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:51.0) Gecko/20100101
Firefox/51.0
Accept: application/xml, text/xml, */*; q=0.01
Accept-Language: zh-CN, zh; q=0.8,en-US; q=0.5,en; q=0.3
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
If-Modified-Since: 0
X-Requested-With: XMLHttpRequest
Referer: http://192.168.86.3/jpeg/mjpeg.html
Content-Length: 69
Connection: keep-alive

cmd=login&user=admin&authcode=750b540fdf15366ae4bf329c1c6c6529&type=1
```

Step 4:

Server → Client

Server received the authcode from Client, successfully verified, Server will send below information to

Client.

```
HTTP/1.0 200 OK
Content-Type: text/xml; charset=UTF-8
Set-Cookie: mjpeg_sess=750b540fdf15366ae4bf329c1c6c6529; path=/;
Set-Cookie: mjpeg_uname=admin; path=/;
Set-Cookie: mjpeg_level=1; path=/;

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



```
<Configuration>
<ResCode>0</ResCode>
<RetMsg>OK</RetMsg>
</Configuration>
```

Step 5:

Later on, interactions between Client and Server, the HTTP header should contain strings like “mjpeg_sess, mjpeg_uname, mjpeg_level”. For example, as shown below:

```
GET /snapshot/view0.jpg?0.9801228921400826 HTTP/1.1
Host: 192.168.86.3
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:51.0) Gecko/20100101
Firefox/51.0
Accept: */*
Accept-Language: zh-CN, zh; q=0.8,en-US; q=0.5,en; q=0.3
Accept-Encoding: gzip, deflate
Referer: http://192.168.86.3/jpeg/mjpeg.html
Cookie: mjpeg_sess=750b540fdf15366ae4bf329c1c6c6529; mjpeg_uname=admin;
mjpeg_level=1
Connection: keep-alive
```

Note:

- “mjpeg_sess” is a string from Server to Client after successful authentication, used for all later requests.
- “mjpeg_uname” is User Name or ID
- “mjpeg_level” is fixed as “1”
- The above **RED** part “Cookie” is the returned verification message of the successful authentication in Login Verification at **Step 4**.

MJPEG Stream (Encryption Mode or Basic Mode)

MJPEG Authentication: The authentication is using either Challenge / Response encryption mode or Basic authentication. After the successful authentication by the administrator, each later on operations will be accepted and successful ONLY when the correct authentication information is included inside the request header.

MJPEG Challenge/Response Authentication at Default Mode

Select “Challenge+Response” from WebUI: **“System Settings → Access Settings → MJPEG Authentication Mode”** to obtain the default MJPEG stream (internal 3th stream) with 1280x720 resolution.

Step 1:



Client → Server

```
<http|https>://<servername>/jpeg/stream?type=0&user=admin
```

Server → Client

```
<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
<ResCode>0</ResCode>
<ChallengeCode>ChallengeString</ChallengeCode>
<IDCode>id_code</IDCode>
<LoginType>0</LoginType>
<RetMsg>OK</RetMsg>
</Configuration>
```

Step2:

Client → Server

```
<http|https>://<servername>/jpeg/stream?type=1&user=admin&authcode=<authcodestring>&idcode=i
d_code
```

Notes:

- authcodestring = md5(ChallengeString:GSC36XXIDyTIHwNgZ:password);
- The "ChallengeString" is the returned message string from Server in **Step 1**.
- password is the login password of the device (GSC36xx).
- Authcode string must be length of 32 in lowercase hex as in MD5 hash.
- The **RED** is the authentication message returned by the Server after successful authentication.
- This part of the message must exist in the message header before all later on operations can run successfully.

Example:

Step 1:

Client → Server (Client sending login request to Server, trying to get ChallengeCode)

```
GET /jpeg/stream?type=0&user=admin HTTP/1.1
Host: 192.168.100.123
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/66.0.3359.139 Safari/537.36
Accept:
```



```
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

Step 2:

Server → Client (Server sending ChallengeCode to Client after receiving the request from Client)

```
HTTP/1.0 200 OK
Content-Type: text/xml;charset=UTF-8

<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg></RetMsg>
<ChallengeCode>9c07f92ea2ab55c2d7ef433e5fa163a0</ChallengeCode>
<IDCode> QDOJBDQBT8MJ7WRVCZ7L</IDCode>
</Configuration>
```

Step 3:

Client → Server (Client receiving ChallengeCode,

via authcodestring = md5(ChallengeCode:GSC36XXIDyTIHwNgZ:password);

generating authcodestring, then sending this authcodestring back to Server)

```
GET
/jpeg/stream?type=1&user=admin&authcode=13e07970ceaadf702bf7200330c72b64&idcode=QDOJBDQBT8MJ7WRVCZ7L HTTP/1.1
Host: 192.168.100.123
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.139 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

Then the MJPEG stream will be successfully retrieved.

Obtain MJPEG stream at specified stream channel

Select "Challenge+Response" from WebUI: **"System Settings → Access Settings → MJPEG Authentication Mode"** to obtain the specified channel stream.

Note: MJPEG must be selected in the Preferred Video Codec for this to work.



Step 1:

Client → Server

```
<http|https>://<servername>/jpeg/stream?type=0&user=admin&stream=X
```

X: 0/1 (0 means 1st MJPEG stream, 1 means 2nd MJPEG stream.)

Server → Client

```
<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
  <ResCode>0</ResCode>
  <ChallengeCode>ChallengeString</ChallengeCode>
  <IDCode>id_code</IDCode>
  <LoginType>0</LoginType>
  <RetMsg>OK</RetMsg>
</Configuration>
```

Step2:

Client → Server

```
<http|https>://<servername>/jpeg/stream?type=1&user=admin&stream=X&authcode=<authcodestring>
&idcode=id_code
```

X: 0 / 1 (0 means 1st MJPEG stream, 1 means 2nd MJPEG stream.)

Notes:

- authcodestring = md5(ChallengeString:GSC36XXIDyTIHwNgZ:password);
- The "ChallengeString" is the returned message string from Server in **Step 1**.
- password is the login password of the device (GSC36xx).
- Authcode string must be length of 32 in lowercase hex as in MD5 hash.
- The **RED** is the authentication message returned by the Server after successful authentication.
- This part of the message must exist in the message header before all later operations can run successfully.

Example to get 2nd MJPEG Stream:

Step 1:

Client → Server (Client sending login request to Server, trying to get ChallengeCode)

```
GET /jpeg/stream?type=0&user=admin&stream=1 HTTP/1.1
```



```
Host: 192.168.100.123
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/66.0.3359.139 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=
0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

Step 2:

Server → Client

(Server sending ChallengeCode to Client after receiving the request from Client)

```
HTTP/1.0 200 OK
Content-Type: text/xml;charset=UTF-8

<?xml version="1.0" encoding="UTF-8" ?>
<Configuration>
<ResCode>0</ResCode>
<RetMsg></RetMsg>
<ChallengeCode>ceab1fac2056be0f889ed634716b466b</ChallengeCode>
<IDCode>LS79DELW7W7987NK87XO</IDCode>
</Configuration>
```

Step 3:

Client → Server

(Client receiving ChallengeCode, via authcodestring =

md5(ChallengeCode:GSC36XXIDyTIHwNgZ:password);

generating authcodestring, then sending this authcodestring back to Server)

```
GET /jpeg/stream?type=1&user=admin&stream=1&authcode=
b08d7dela1af2d2bf3b7e0f6e9fb87f2&idcode= LS79DELW7W7987NK87XO HTTP/1.1
Host: 192.168.100.123
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/66.0.3359.139 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=
0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

Then the MJPEG stream will be retrieved successfully.



MJPEG Basic Authentication For Default Method:

Select “Basic” from WebUI: “System Settings → Access Settings → MJPEG Authentication Mode” to obtain the default MJPEG stream (internal 3rd stream) with 1280x720 resolution.

Client → Server

```
<http|https>://username:password@<servername>/jpeg/stream
```

Note: Username: web login username; password: web login password

Below is an **Example**:

Client → Server

(Client sending GET request to Server, trying to get MJPEG Stream)

```
GET /jpeg/stream HTTP/1.1
Host: 192.168.100.123
Connection: keep-alive
Authorization: Basic YWRtaW46YWRtaW4=
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/67.0.3396.87 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

Then the MJPEG stream will be retrieved.



MJPEG Basic Authentication For Recommended Method:

Select "Basic" from WebUI: "System Settings → Access Settings → MJPEG Authentication Mode" to obtain the specified channel stream.

Note: MJPEG must be selected in the Preferred Video Codec for this to work.

Client → Server

```
<http|https>://username:password@<servername>/jpeg/stream=X
```

X: 0/1 (0 means 1st MJPEG stream, 1 means 2nd MJPEG stream.)

Notes:

- Username: web login username; password: web login password
- Below is an Example to get 2nd MJPEG Stream:
- Client → Server (Client sending GET request to Server, trying to get MJPEG Stream)

```
GET /jpeg/stream=1 HTTP/1.1
Host: 192.168.100.123
Connection: keep-alive
Authorization: Basic YWRtaW46YWRtaW4=
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/67.0.3396.87 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
```

Then the 2nd MJPEG Stream will be retrieved



MJPEG video or Snapshot image via HTTP API to 3rd party System Integration

For easy system integration (with the cost of less secure), once the feature enabled (default is disabled), user can send HTTP API with correct credentials to retrieve MJPEG video or JPEG snapshot from GSC3610, similar to the behavior of Grandstream IP Cameras.

The HTTP API or CLI command listed as below:

MJPEG Video:

[http\(s\)://admin:password@IP_GSC36XX:Port/jpeg/mjpeg.html](http(s)://admin:password@IP_GSC36XX:Port/jpeg/mjpeg.html)

JPEG Snapshot:

[http\(s\)://admin:password@IP_GSC36XX:Port/jpeg/view.html](http(s)://admin:password@IP_GSC36XX:Port/jpeg/view.html)

Anonymous Liveview/Snapshot/Stream :

Enable “Enable Anonymous LiveView” in the “**Access Settings**” WebUI page

MJPEG Video:

[http\(s\)://IP_GSC36xx:Port/videoview.html](http(s)://IP_GSC36xx:Port/videoview.html)

JPEG Snapshot:

[http\(s\)://IP_GSC36xx:Port/anonymous/snapshot/view.html](http(s)://IP_GSC36xx:Port/anonymous/snapshot/view.html)

[http\(s\)://IP_GSC36xx:Port/anonymous/snapshot/view.jpg](http(s)://IP_GSC36xx:Port/anonymous/snapshot/view.jpg)

MJPEG Stream:

[http\(s\)://IP_GSC36xx:Port/anonymous/jpeg/stream=X](http(s)://IP_GSC36xx:Port/anonymous/jpeg/stream=X)

X: 0/1 (0 means 1st MJPEG stream, 1 means 2nd MJPEG stream. MJPEG must be selected in the Preferred Video Codec for this to work.)

Notes:

- MJPEG stream may feel like animation due to the compromise of video quality and bandwidth.
- Similar command can be applied to open-source application like **VLC MediaPlayer** to retrieve H.264 video stream with better quality:

rtsp://admin:password@IP_GSC36XX:Port/X



(X= 0, 4, corresponding to 1st, 2nd video stream where 2nd stream recommended)

Other Data API

Support cmd= get

Get:

http://IP:Port/goform/config?cmd=api_get_data&type=0

Example

```
<Configuration>
  <MD_STATUS>1</MD_STATUS>
  <RetMsg>OK</RetMsg>
</Configuration>
```

<parameter>=<value>	Values	Description
P63010=<int>	<0 1>	Get Motion Detection Status 0: Non alarm status 1: alarm status

HTTP SIP DIAL

<parameter>=<value>	Values	Description
cmd=<string>	Call	http sip call cmd
call_type=<int>	<0 1>	0:end call 1:call
call_num=<string>		call num or IP

REFERENCES

HTTP Protocol

- [Hypertext Transfer Protocol for HTTP/1.0](#)

External application programming interfaces (Client side)

