

# HTTP API of Event

## Version 3.0.5

Release notes			
Version	Description	Date	Authors
V3.0.1		2023/07/25	Kuang
V3.0.2	Add Alarm Subscription	2024/02/04	Shawn
V3.0.3	Add light and sound active triggers and config	2024/08/04	Shawn
V3.0.4	Adjusting alarm subscriptions and message content	2024/11/11	Shawn
V3.0.5	Default use of Digest authentication	2024/12/21	Shawn

# Catalog

1 Summary .....	6
2 Linkage .....	7
2.1 Description .....	7
2.2 Grammar .....	7
2.3 Parameter .....	7
2.4 Example .....	8
2.4.1 Get Action .....	8
2.4.2 Set Action .....	10
3 IO Alarm Configuration .....	11
3.1 Description .....	11
3.2 Grammar .....	11
3.3 Parameter .....	11
3.4 Example .....	12
3.4.1 Get IO Alarm configuration .....	12
3.4.2 Set IO Alarm configuration .....	14
4 Schedule Configuration .....	15
4.1 Description .....	15
4.2 Grammar .....	15
4.3 Parameter .....	15
4.4 Example .....	16
4.4.1 Get schedule configuration .....	16

4.4.2 Set schedule configuration .....	20
5 Motion Detection .....	21
5.1 Description .....	21
5.2 Grammar .....	21
5.3 Parameter .....	21
5.4 Example .....	22
5.4.1 Get motion detection configuration .....	22
5.4.2 Set motion detection configuration .....	24
6 Alarm Subscription .....	26
6.1 TCP Push Alarm Messages .....	26
6.1.1 Alarm Subscription .....	26
6.1.2 Update Subscription .....	29
6.1.3 Delete Subscription .....	30
6.1.4 Alarm messages .....	31
6.2 HTTP Get Alarm Message .....	35
7 Relay output.....	39
7.1 Description .....	39
7.2 Grammar .....	39
7.3 Parameter .....	39
7.4 Example .....	39
7.4.1 Get relay output config .....	39
7.4.2 Set relay output config .....	40

8 Trigger light flashing .....	41
8.1 Description .....	41
8.2 Grammar .....	41
8.3Parameter .....	41
8.4 Example .....	41
9 Configure light flashing .....	42
9.1 Description .....	42
9.2 Grammar .....	42
9.3Parameter .....	42
9.4 Example .....	43
9.4.1 Get light flashing parameters .....	43
9.4.2 Setting the light flashing parameters .....	43
10Getting sound files ( Sound File ) .....	44
10.1 Description .....	44
10.2 Getting sound files .....	44
10.2.1 Grammar .....	44
10.2.2 Parameter .....	44
10.2.3 Example .....	45
11.Playing sound files .....	45
11.1 Description .....	45
11.2 Grammar .....	45
11.3Parameter .....	46

11.4 Example .....	46
Appendix .....	47
1 Save Type .....	47
2 Action Type .....	47
3 PTZ Operation Type (Ptz command type) .....	47
4 PTZ Appendix .....	48
5 Alarm type .....	50

# 1 Summary

This command is used for get/set linkage, io alarm, schedule, motion detection, etc.

## 2 Linkage

### 2.1 Description

This command is used for get/set linkage name, linkage action.

### 2.2 Grammar

Get:

```
http://<Device IP>/cgi/event.cgi?get=action
```

Set:

```
http://<Device IP>/cgi/event.cgi?set=action&data=<?xml version="1.0"
encoding="utf-8"?><root><action xxx /></root>
```

### 2.3 Parameter

Parameter	Value	Description
<action>	none	
ch	Index	Action channel
valid	Practical	0:disable this linkage; 1:enable this linkage
action_name	Practical	Action name
prepare_second	Practical	Wait time before action is triggered. Obsoleting
dev_chn	Practical	Internal use, can be ignored
action_millisec	Practical	Action continuous time, Unit: micro second
save_type	Practical	Save type of record and snap, refer to Save Type
action_type	Practical	Action type, refer to Action Type
ftp_id	Practical	Ftp id

smtp_id	Practical	Smtip id
<snap>	none	
interval	Practical	Snap shot interval
num	Practical	Snap shot numbers, this parameter will be active when ActionMilisec is 0

Parameter value is determined by ActionType :

ActionType	Description	Include Parameter
0	None	NULL
1	Snap	<SaveType>、 <SnapInterval>、 <SnapNum>
2	Record	<SaveType>、 <RecStreamId>
3	PTZ	<PtzCmd>、 <PtzParam1>、 <PtzParam2>
4	Alarm out	<DOStatus>
5	Reboot	NULL

## 2.4 Example

### 2.4.1 Get Action

#### REQUEST

```
http://192.168.2.172/cgi/event.cgi?get=action
```

#### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

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

```
<root>
```

```
<version version='0' />
```

```
<action ch='0' valid='1' action_name='snap' action_type='1' dev_chn='0'
```



```
prepare_second='0' action_millisec='0' save_type='3' ftp_id='255' smtp_id='255'>
<snap num='1' interval='2' snap_streamid='2'/>
</action>
<action ch='1' valid='1' action_name='record' action_type='2' dev_chn='0'
prepare_second='0' action_millisec='0' save_type='3' ftp_id='255' smtp_id='255'>
<record streamid='1'/>
</action>
<action ch='2' valid='1' action_name='relay' action_type='4' dev_chn='0'
prepare_second='0' action_millisec='10000' save_type='0' ftp_id='255' smtp_id='255'>
<do status='1'/>
</action>
<action ch='3' valid='1' action_name='light' action_type='6' dev_chn='0'
prepare_second='0' action_millisec='0' save_type='0' ftp_id='255' smtp_id='255'>
<light status='1'/>
</action>
<action ch='4' valid='1' action_name='voice' action_type='7' dev_chn='0'
prepare_second='0' action_millisec='0' save_type='0' ftp_id='255' smtp_id='255'>
<voice num='5' interval='0' fileid='5'/>
</action>
<action ch='5' valid='1' action_name='reboot' action_type='5' dev_chn='0'
prepare_second='0' action_millisec='0' save_type='0' ftp_id='255' smtp_id='255'>
</action>
<action ch='6' valid='0' action_name="" action_type='0' dev_chn='0' prepare_second='0'
action_millisec='0' save_type='0' ftp_id='255' smtp_id='255'>
</action>
<action ch='7' valid='0' action_name="" action_type='0' dev_chn='0' prepare_second='0'
action_millisec='0' save_type='0' ftp_id='255' smtp_id='255'>
</action>
</root>
```

## 2.4.2 Set Action

Example: Modify the action type in channel 1 to capture, capture the action name snap1, capture the number of sheets for 10, capture interval for 1000ms.

Tip: must specify the channel information.

### REQUEST

```
http://192.168.2.172/cgi/event.cgi?set=action&data=<?xml version="1.0"
encoding="utf-8"?><root><action ch='0' valid='1' action_name='snap1' action_type='1'
save_type='1'> <snap num='10' interval='1000'/></action></root>
```

### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<code>0</code>
<error>successful</error>
</root>
```

# 3 IO Alarm Configuration

## 3.1 Description

This command is used for get/set IO alarm information, input type, action id, etc.

## 3.2 Grammar

Get:

```
http://<Device IP>/cgi/event.cgi?get=alarm_io
```

Set:

```
http://<Device IP>/cgi/event.cgi?set=alarm_io&data=<?xml version="1.0"
encoding="utf-8"?><root><di ch= '0' xxx><action xxx /><segment
xxx></segment></di></root>
```

Note: ch is IO channel

## 3.3 Parameter

Parameter	Value	Description
<di>	none	
ch	index	IO channel
on	practical	0:Close IO alarm ; 1: Open IO alarm
type	practical	0:Close linkage action of IO alarms; 1: Open linkage action of IO alarms
persistent	practical	Duration of IO alarms (1-600)
<action>	practical	Linkage action, there are eight values, each value represents the ID of the linkage action, the value of -1 is not used

<segment>	none	
id	index	Time Setting ID
on	practical	0: Time segment is off; 1: Time segment is on
begin_day	practical	Monday to Sunday (1~7)
begin_hour	practical	Begin hour (0~23)
begin_minute	practical	Begin minute (0~59)
begin_second	practical	Begin second (0~59)
begin_millisecond	practical	Begin millisecond (0~1000)
end_hour	practical	End hour (0~23)
end_minute	practical	End minute (0~59)
end_second	practical	End second (0~59)

## 3.4 Example

### 3.4.1 Get IO Alarm configuration

#### REQUEST

```
http://192.168.2.172/cgi/event.cgi?get=alarm_io
```

#### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='0' />
<di ch='0' type='0' on='1' persistent='5000'>
<action>0</action>
```

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<segment id='0' on='1' begin\_day='7' end\_day='7' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='1' on='1' begin\_day='1' end\_day='1' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='2' on='1' begin\_day='2' end\_day='2' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='3' on='1' begin\_day='3' end\_day='3' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='4' on='1' begin\_day='4' end\_day='4' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='5' on='1' begin\_day='5' end\_day='5' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='6' on='1' begin\_day='6' end\_day='6' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

</di>

```
<di ch='1' type='0' on='0' persistent='5000'>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
</di>
</root>
```

### 3.4.2 Set IO Alarm configuration

Example: Modify the switch of IO alarm in channel 1 to be on and the effective level to be normally closed.

Tip: Channel information must be specified.

#### REQUEST

```
http://192.168.2.172/cgi/event.cgi?set=alarm_io&data=<?xml version="1.0"
encoding="utf-8"?><root><di ch='0' type='1' on='1'></di></root>
```

#### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<code>0</code>
<error>successful</error>
</root>
```

# 4 Schedule Configuration

## 4.1 Description

This command is used for schedule on-off, time segment, linkage actions.

## 4.2 Grammar

Get:

```
http://<Device IP>/cgi/event.cgi?get=scheme
```

Set:

```
http://<Device IP>/cgi/event.cgi?set=scheme&data=<?xml version="1.0"
encoding="utf-8"?><root><scheme ch='0' xxx><action xxx /><segment
xxx></segment></scheme></root>
```

Note: ch is schedule channel, range is [0-7]

## 4.3 Parameter

Parameter	Value	Description
<scheme>	none	
ch	index	Scheme channel
on	practical	0:Close schedule;1:Open schedule
name	practical	Schedule name
<action>	practical	Linkage action, there are eight values, each value represents the ID of the linkage action, the value of -1 is not used
<segment>	none	
id	index	Time Setting ID

on	practical	1: Time segment is on. 0: Time segment is off
begin_day	practical	Monday to Sunday (1~7)
begin_hour	practical	Begin hour (0~23)
begin_minute	practical	Begin minute (0~59)
begin_second	practical	Begin second (0~59)
begin_millisec	practical	Begin millisecond (0~1000)
end_hour	practical	End hour (0~23)
end_minute	practical	End minute (0~59)
end_second	practical	End second (0~59)

## 4.4 Example

### 4.4.1 Get schedule configuration

#### REQUEST

```
http://192.168.2.172/cgi/event.cgi?get=scheme
```

#### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='0' />
<scheme ch='0' on='1' name='record'>
<action>1</action>
<action>-1</action>
<action>-1</action>
```



<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<segment id='0' on='1' begin\_day='7' end\_day='7' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='1' on='1' begin\_day='1' end\_day='1' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='2' on='1' begin\_day='2' end\_day='2' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='3' on='1' begin\_day='3' end\_day='3' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='4' on='1' begin\_day='4' end\_day='4' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='5' on='1' begin\_day='5' end\_day='5' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

<segment id='6' on='1' begin\_day='6' end\_day='6' begin\_hour='0' begin\_minute='0'  
begin\_second='0' begin\_millisecond='0' end\_hour='24' end\_minute='0' end\_second='0'  
end\_millisecond='0' type='0' ></segment>

</scheme>

<scheme ch='1' on='0' name=''>

<action>-1</action>

```
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
</scheme>
<scheme ch='2' on='0' name=''>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
</scheme>
<scheme ch='3' on='0' name=''>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
</scheme>
<scheme ch='4' on='0' name=''>
```

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

&lt;/scheme&gt;

```
<scheme ch='5' on='0' name='>
```

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

&lt;/scheme&gt;

```
<scheme ch='6' on='0' name="">
```

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

<action>-1</action>

&lt;action&gt;-1&lt;/action&gt;

<action>-1</action>

<action>-1</action>

&lt;/scheme&gt;

```
<scheme ch='7' on='0' name=''>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
<action>-1</action>
</scheme>
</root>
```

## 4.4.2 Set schedule configuration

Example: Modify the plan name of the plan id of the plan task in channel 1 to 0 to abc and enable the plan.

Tip: You must specify the channel information.

### REQUEST

```
http://192.168.2.172/cgi/event.cgi?set=scheme&data=<?xml version="1.0"
encoding="utf-8"?><root><scheme ch='0' on='1' name='abc' type='1'></scheme></root>
```

### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<code>0</code>
<error>successful</error>
</root>
```

# 5 Motion Detection

## 5.1 Description

This command is used for motion detection on-off, sensitivity, threshold , continuous time, linkage action, etc.

## 5.2 Grammar

Get:

```
http://<Device IP>/cgi/event.cgi?get=motion
```

Set:

```
http://<Device IP>/cgi/event.cgi?set=motion&data=<?xml version="1.0"
encoding="utf-8"?><root><motion ch='0' xxx><action>xxx</action><segment xxx
/></motion></root>
```

Note: ch is schedule channel, range is [0-7]

## 5.3 Parameter

Parameter	Value	Description
<motion>	none	
ch	index	Motion channel
on	practical	0:Close detection; 1:Open detection
sensitivity	practical	Detect sensitivity, range (0~100)
threshold	practical	Detect threshold, range (0~100)
persistent	practical	Motion detection continuous time, Unit: ms
<action>	practical	After the motion detection trigger linkage action,

		there are eight values, each value represents the ID of the linkage action, the value of -1 is not used
<block>	practical	Motion detection block, device separate the image to 18 x 22 blocks
<segment>	none	
id	index	Time Setting ID
on	practical	1: Time segment is on. 0: Time segment is off
begin_day	practical	Monday to Sunday (1~7)
begin_hour	practical	Begin hour (0~23)
begin_minute	practical	Begin minute (0~59)
begin_second	practical	Begin second (0~59)
begin_millisecond	practical	Begin millisecond (0~1000)
end_hour	practical	End hour (0~23)
end_minute	practical	End minute (0~59)
end_second	practical	End second (0~59)

## 5.4 Example

### 5.4.1 Get motion detection configuration

#### REQUEST

```
http://192.168.2.172/cgi/event.cgi?get=motion
```

#### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

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

```
<root>  
  
<version version='0' />  
  
<motion ch='0' on='0' sensitivity='85' threshold='20' persistent='5000'>  
  
<block>0101010101010101010101010101010101010101010101010101010101010101  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
0101010101010101010101010101010101010101010101010101010101010101  
1010101010101010101010101010101010101010101010101010101010101010  
010101010101010101</block>  
  
<action>-1</action>  
  
<action>1</action>  
  
<action>-1</action>  
  
<action>-1</action>  
  
<action>-1</action>  
  
<action>-1</action>  
  
<action>-1</action>  
  
<action>-1</action>  
  
<segment id='0' on='1' begin_day='7' end_day='7' begin_hour='0' begin_minute='0'  
begin_second='0' begin_millisecond='0' end_hour='24' end_minute='0' end_second='0'  
end_millisecond='0' type='0' ></segment>  
  
<segment id='1' on='1' begin_day='1' end_day='1' begin_hour='0' begin_minute='0'  
begin_second='0' begin_millisecond='0' end_hour='24' end_minute='0' end_second='0'  
end_millisecond='0' type='0' ></segment>
```

```
<segment id='2' on='1' begin_day='2' end_day='2' begin_hour='0' begin_minute='0'
begin_second='0' begin_millisec='0' end_hour='24' end_minute='0' end_second='0'
end_millisec='0' type='0' ></segment>
<segment id='3' on='1' begin_day='3' end_day='3' begin_hour='0' begin_minute='0'
begin_second='0' begin_millisec='0' end_hour='24' end_minute='0' end_second='0'
end_millisec='0' type='0' ></segment>
<segment id='4' on='1' begin_day='4' end_day='4' begin_hour='0' begin_minute='0'
begin_second='0' begin_millisec='0' end_hour='24' end_minute='0' end_second='0'
end_millisec='0' type='0' ></segment>
<segment id='5' on='1' begin_day='5' end_day='5' begin_hour='0' begin_minute='0'
begin_second='0' begin_millisec='0' end_hour='24' end_minute='0' end_second='0'
end_millisec='0' type='0' ></segment>
<segment id='6' on='1' begin_day='6' end_day='6' begin_hour='0' begin_minute='0'
begin_second='0' begin_millisec='0' end_hour='24' end_minute='0' end_second='0'
end_millisec='0' type='0' ></segment>
</motion>
</root>
```

## 5.4.2 Set motion detection configuration

Example: Modify the sensitivity of motion detection in channel 1 to 20.

Tip: must specify the channel information.

### REQUEST

```
http://192.168.2.172/cgi/event.cgi?set=motion&data=<?xml version="1.0"
encoding="utf-8"?><root><motion ch='0' sensitivity='20'></motion></root>
```

### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain; charset=utf-8*



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

```
<root>
```

```
<code>0</code>
```

```
<error>successful</error>
```

```
</root>
```

# 6 Alarm Subscription

The module supports two forms of alarm message subscription:

I: Push alarm messages via TCP.

II: Get alarm messages via HTTP.

## 6.1 TCP Push Alarm Messages

This command is used to allow an external TCP server to receive alarms from the device.

The TCP server first needs to subscribe to the alarm via the cgi command, at which point the device returns the subscription ID, which is used to update and delete the subscription, and the device will push the resulting alarm message to the subscribed TCP server.

### 6.1.1 Alarm Subscription

#### 6.1.1.1 Description

This command is used for alarm subscription; the subscription is guaranteed for 90sec and needs to be renewed within the guarantee period (see section 6.1.2 Renewal of subscription for details).

#### 6.1.1.2 Grammar

```
http://<Device IP>/cgi/event.cgi?act=alarm_subscribe&data=<?xml version="1.0"
encoding="utf-8"?><root><address>[tcp-server-ip]:[tcp-server-port]</address></r
oot>
```

Note: You need to url-encode the data in the data when subscribing, otherwise the colon (:) in [tcp-server-ip]:[tcp-server-port] will cause parsing errors on the device side.

### 6.1.1.3 Parameter

Parameter	Value	Description
<address>	practical	Indicates TCP server address information by consisting of [IP:PORT].
<connecttimeout>	practical	Indicates the length of connection timeout when the device sends a message via TCP, default is 500ms, the supported timeout interval is [500,5000]ms, optional.
<sendtimeout>	practical	Indicates the timeout duration of the message sent by the device via TCP, default is 500ms, the supported timeout interval is [500,5000]ms, optional.
<details>	practical	Indicates whether the coordinates and picture data node of the detection target are included when getting the alarm; 0: basic information (including alarm timestamp, alarm type and status), 1: completion information (basic information + detection coordinates and picture data). Default [basic information].

### 6.1.1.4 ExampleA

Subscription alarms only, not carrying capture information, send connection timeout and send timeout times。

### REQUEST

```
http://192.168.2.224/cgi/event.cgi?auth=7b5f58d0bf4c11eea1761b3920b29d14&act=alarm_subscribe&data=<?xml version="1.0" encoding="utf-8"?><root><address>192.168.2.44:2023</address></root>
```

After encoding data

```
http://192.168.2.224/cgi/event.cgi?auth=7b5f58d0bf4c11eea1761b3920b29d14&act=alarm_subscribe&data=%3c%3fxml+version%3d%221.0%22+encoding%3d%22utf-8%22%3f%3e%3croot%3e%3caddress%3e192.168.2.44%3a2023%3c%2faddress%3e%3c%2froot%3e
```

Direct url encoding colon (:

```
http://192.168.2.224/cgi/event.cgi?auth=7b5f58d0bf4c11eea1761b3920b29d14&act=alarm_subscribe&data=<?xml version="1.0" encoding="utf-8"?><root><address>192.168.2.44%3a2023</address></root>
```

## RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain; charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<code>0</code>
<error>successfuel</error>
<subscriptionid>9482d100e7b611ef9d9843627f60a385</subscriptionid>
</root>
```

### 6.1.1.5 ExampleB

Carries customized connection timeout, customized send timeout and detection target coordinates as well as image data, which can be configured as required.

## REQUEST

```
http://192.168.2.224/cgi/event.cgi?act=alarm_subscribe&data=<?xml version="1.0" encoding="utf-8"?><root><address>192.168.2.44%3a2023</address><connecttimeout>600</connecttimeout><sendtimeout>600</sendtimeout><details>1</details></root>
```

## RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain; charset=utf-8*

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

<root>

<code>0</code>

<error>successfuel</error>

<subscriptionid>8e92e3a01dfc11b28487bf505185d5ed</subscriptionid>

</root>
```

## 6.1.2 Update Subscription

### 6.1.2.1 Description

This command is used to update the alarm subscription, the subscription keep-alive period is 90 seconds, you need to use this command to update the alarm subscription in time, otherwise the device side will automatically delete this subscription.

### 6.1.2.2 GrammarA

Update alarm subscriptions by subscriptionid.

```
http://<DeviceIP>/cgi/event.cgi?act=alarm_updatesubscribe&data=<subscriptionid>
xxxxxx</subscriptionid>
```

### REQUEST

```
http://192.168.2.234/cgi/event.cgi?act=alarm_updatesubscribe&data=<?xml
version="1.0"
encoding="utf-8"?><root><subscriptionid>8e92e3a01dfc11b28487bf505185d5ed</subsc
riptionid></root>
```

### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain; charset=utf-8*

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

```
<root>

  <code>0</code>

  <error>successful</error>

</root>
```

## 6.1.3 Delete Subscription

### 6.1.3.1 Description

This command is used to remove the alert subscription.

### 6.1.3.2 GrammarA

Remove alarm subscriptions by subscriptionid.

```
http://<Device IP>/cgi/event.cgi?act=alarm_unsubscribe&data=<?xml version="1.0"
encoding="utf-8"?><root><subscriptionid>xxxxxx</subscriptionid></root>
```

### REQUEST

```
http://192.168.2.234/cgi/event.cgi?act=alarm_unsubscribe&data=<?xml
version="1.0"
encoding="utf-8"?><root><subscriptionid>8e92e3a01dfc11b28487bf505185d5ed</subsc
riptionid></root>
```

### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain; charset=utf-8*

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

<root>

  <code>0</code>

  <error>successful</error>

</root>
```

## 6.1.4 Alarm messages

Alarm message telegrams pushed by IPC.

### 6.1.4.1 Parameter

Parameter	Value	Description
<alarm>	practical	Date of Alarm triggered
timestamp	practical	Indicates the device timestamp when the alarm was generated
type	practical	See Appendix 5 Alarm Types for details
channel	index	Indicates the alarm channel, when the generated alarm type is motion detection alarm, it indicates the video channel; when the generated alarm type is IO alarm, it indicates the channel number of DI
signal	practical	Indicates an alarm signal. 0 means the alarm is canceled, 1 means the alarm is generated, and they will appear in pairs.
		The following face, person, model, pet, and other alarms can be controlled for inclusion via the details field when subscribing
<face>	practical	Face Alarm Information Node
num	practical	Number of face detection targets
<faceinfo>	practical	Face Target Information Node
id	index	Face Target Index
type	practical	Detect target type, default 0
large_pic_index	index	Alarm large pic index
small_pic_index	index	Alarm small pic index
<position>	practical	Detect target coordinate information, parameterized

		as a percentage of the panorama and converted to (0-1000)
x	practical	x
y	practical	y
width	practical	width
height	practical	height
<person>	practical	Humanoid alarm message node
num	practical	Number of human detection targets
<personinfo>	practical	humanoid information node
id	index	Humanoid Target Index
type	practical	Detect target type, default 0
large_pic_index	index	Alarm large pic index
small_pic_index	index	Alarm small pic index
<position>	practical	Detect target coordinate information, parameterized as a percentage of the panorama and converted to (0-1000)
x	practical	x
y	practical	y
width	practical	width
height	practical	height
<vehicle>	practical	Vehicle Alarm Message Node
id	index	Vehicle target Index
type	practical	Detect target type, default 0
large_pic_index	index	Alarm large pic index
small_pic_index	index	Alarm small pic index
<position>	practical	Detect target coordinate information, parameterized as a percentage of the panorama and converted to (0-1000)



x	practical	x
y	practical	y
width	practical	width
height	practical	height
</vehicle>		
<pet>	practical	Pet Alarm Information Node
id	index	Pet Target Index
type	practical	Detect target type, default 0
large_pic_index	index	Alarm large pic index
small_pic_index	index	Alarm small pic index
<position>	practical	Detect target coordinate information, parameterized as a percentage of the panorama and converted to (0-1000)
x	practical	x
y	practical	y
width	practical	width
height	practical	height
<largepic>	practical	Alarm Panorama Node
num	practical	Number of pictures
<pic>	practical	Alarm Map Information Node
id	index	Panorama Index
format	practical	Image format, default 0: jpg image in base64 format
width	practical	width
height	practical	height
size	practical	Base64 image string length
<smallpic>	practical	Alarm target mini-graph node
num	practical	Number of pictures
<pic>	practical	Alarm Map Information Node

id	index	Targeted Small Figure Index
format	practical	Image format, default 0: jpg image in base64 format
width	practical	width
height	practical	height
size	practical	Base64 image string length

### 6.1.4.2 ExampleA

Subscription without <details>1<details> node

Tip: To receive alerts, you need to create a TCP server and subscribe to it within the alive cycle before you can receive alert messages.

#### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='1' />
<alarm timestamp='1730805729' type='128' channel='0' signal='1'>
</alarm>
</root>
```

### 6.1.4.3 ExampleB

Subscription with <details>1<details> node

Tip: To receive alerts, you need to create a TCP server and subscribe to it within the alive cycle before you can receive alert messages.

#### RESPONSE

HTTP/1.0 200 OK

Content-type: text/plain;charset=utf-8

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <version version='1' />
  <alarm timestamp='1730805726' type='128' channel='0' signal='1'>
    <vehicle num='1'>
      <vehicleinfo id='0' type='2' large_pic_index='0' small_pic_index='0'>
        <position x='1400' y='4100' width='2700' height='5300' />
      </vehicleinfo>
    </vehicle>
    <largepic num='1'>
      <pic id='0' format='0' width='800' height='480'
size='29112'>--base64-jpg-data--</pic>
    </largepic>
    <smallpic num='1'>
      <pic id='0' format='0' width='128' height='128'
size='8156'>--base64-jpg-data--</pic>
    </smallpic>
  </alarm>
</root>
```

## 6.2 HTTP Get Alarm Message

### 6.2.1 Description

HTTP to get alarm information without subscribing, updating, or deleting alarm subscriptions individually, request and response in a single request.

HTTP request to get the alarm message, if there is no alarm on the device side at this time, the request needs to wait for 60s, if waiting for the process of the device to generate

an alarm, then return immediately, more than 60s, then return an xml data without alarm data.

## 6.2.2 Alarm Message Parameters

Refer to xml node description in 6.1.4.1.

## 6.2.3 GrammarA

No need to carry alarm capture data.

```
http://<Device IP>/cgi/event.cgi?act=alarm_message
```

Example: The device generates an alarm within 1min.

### REQUEST

```
http://192.168.2.171/cgi/event.cgi?act=alarm_message
```

### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <version version='1' />
  <alarm id='0' timestamp='1730801315' type='512' channel='0' signal='1'>
  </alarm>
</root>
```

Example: No alarms are generated in 1min.

### REQUEST

```
http://192.168.2.171/cgi/event.cgi?act=alarm_message
```

### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

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

<root>

    <version version='1' />

</root>
```

## 6.2.4 GrammarB

Carrying detail node data.

```
http://<Device IP>/cgi/event.cgi?act=alarm_message&data=<?xml version="1.0"
encoding="utf-8"?><root><details>1</details></root>
```

Parameter	Value	Description
<details>	practical	Indicates whether the coordinates and picture data node of the detection target are included when getting the alarm; 0: basic information (including alarm timestamp, alarm type and status), 1: completion information (basic information + detection coordinates and picture data). Default [basic information].

## REQUEST

```
http://192.168.2.171/cgi/event.cgi?data=<?xml version="1.0"
encoding="utf-8"?><root><details>1</details></root>
```

## RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain; charset=utf-8*

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

<root>

    <version version='1' />

    <alarm timestamp='1730805726' type='128' channel='0' signal='1'>
```

```
<vehicle num='1'>
  <vehicleinfo id='0' type='2' large_pic_index='0' small_pic_index='0'>
    <position x='1400' y='4100' width='2700' height='5300' />
  </vehicleinfo>
</vehicle>
<largepic num='1'>
  <pic id='0' format='0' width='800' height='480'
size='29112'>--base64-jpg-data--</pic>
</largepic>
<smallpic num='1'>
  <pic id='0' format='0' width='128' height='128'
size='8156'>--base64-jpg-data--</pic>
</smallpic>
</alarm>
</root>
```

# 7 Relay output

## 7.1 Description

This group of commands will get and set the Xml format data consisting of DO alarm parameters, the parameters include relay output channel, relay status control.

## 7.2 Grammar

Get DO param:

```
http://<Device IP>/cgi/event.cgi?get=alarm_do
```

Set DO param:

```
http://<Device IP>/cgi/event.cgi?set=alarm_do&data=<?xml version="1.0"
encoding="utf-8"?><root><docfg ch='0' type='1'></di></root>
```

Note: ch indicates DO channel

## 7.3 Parameter

Parameter	Value	Description
<docfg>	无	
ch	index	DO channel
on	practical	0-off, 1-on
type	practical	0-NO, 1-NC

## 7.4 Example

### 7.4.1 Get relay output config

#### REQUEST

```
http://192.168.2.227/cgi/event.cgi?get=alarm_do
```

## RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <version version='0' />
  <docfg ch='0' type='0'>
  </docfg>
</root>
```

## 7.4.2 Set relay output config

Example: Modify channel 0 relay output to NC state.

## REQUEST

```
http://192.168.2.227/cgi/event.cgi?set=alarm_do&data=<?xml version="1.0"
encoding="utf-8"?><root><docfg ch='0' type='1'></docfg></root>
```

## RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<code>0</code>
<error>successful</error>
</root>
```



## 8 Trigger light flashing

### 8.1 Description

Active triggering white light flashing, can control the light flashing state, frequency, duration.

### 8.2 Grammar

Trigger white light flashing:

```
http://192.168.2.248/cgi/event.cgi?act=flashing_light&data=<?xml version="1.0"
encoding="utf-8"?><root><flashinglight ch="0" status="1" frequency="0" presist_time='1000' /></root>
```

### 8.3Parameter

Parameter	Value	Description
<flashinglight >	practical	
ch	practical	channel:0 default
status	practical	status: 1-on 0-off
frequency	practical	Lights flashing flashing frequency: 0-high 1-medium 2-low
presist_time	practical	Duration: ms, -1 aways.

### 8.4 Example

#### REQUEST

```
http://192.168.2.248/cgi/event.cgi?act=flashing_light&data=<?xml version="1.0"
encoding="utf-8"?><root><flashinglight ch="0" status="1" frequency="2"
presist_time='1000' /></root>
```

#### RESPONSE

```
HTTP/1.0 200 OK
```

Content-type: text/plain;charset=utf-8

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <code>0</code>
  <error>successful</error>
</root>
```

## 9 Configure light flashing

### 9.1 Description

Getting and setting the light flashing parameters, can control the light flashing mode, frequency and duration.

### 9.2 Grammar

Get light flashing parameters:

```
http://<DeviceIP>/cgi/event.cgi?get=flashing_light
```

Set light flashing parameters:

```
http://<DeviceIP>/cgi/event.cgi?set=flashing_light&data=<?xml version="1.0"
encoding="GBK"?><root><flashinglight act_mode='0' frequency='0' presist_time='1000' /></root>
```

### 9.3Parameter

Parameter	Value	Description
<flashinglight >	practical	
act_mode	practical	Mode: 0 continuous trigger, 1 single trigger 发 For scenes that use the interface to actively trigger light flashing without alarm linkage on the device side, it is recommended that 0

		<p>Effect of different modes :</p> <p>0 continuous triggering: according to the duration of the alarm module to decide, the alarm starts the light flashing, the alarm ends, the light stops.</p> <p>1 Single Trigger: Each time the alarm is triggered, it only flashes once, the flashing time is determined by the light duration, if the duration of this alarm is longer than the set light duration, it also only flashes the light for the set length of time.</p>
frequency	practical	Lights flashing flashing frequency: 0-high 1-medium 2-low
presist_time	practical	Light duration: unit/ms, range (10000-3600000) ms

## 9.4 Example

### 9.4.1 Get light flashing parameters

#### REQUEST

```
http://192.168.2.248/cgi/event.cgi?get=flashing_light
```

#### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

```
<?xml version="1.0" encoding="GBK"?>
<root>
  <flashinglight act_mode='0' frequency='0' presist_time='11000' />
</root>
```

### 9.4.2 Setting the light flashing parameters

#### REQUEST

```
http://192.168.2.248/cgi/event.cgi?set=flashing_light&data=<?xml version="1.0"
```

```
encoding="GBK"?><root><flashinglight act_mode='0' frequency='0' persist_time='1000'  
/></root>
```

## RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>  
<root>  
  <code>0</code>  
  <error>successful</error>  
</root>
```

## 10Getting sound files (Sound File)

### 10.1 Description

Get a list of sound files supported by the device。

### 10.2 Getting sound files

#### 10.2.1 Grammar

Get a list of sound files supported by the device:

```
http://<Device IP>/cgi/event.cgi?act=alarm_voice_file_list
```

#### 10.2.2 Parameter

Parameter	Value	Description
id	practical	sound file id
name	practical	sound file name

## 10.2.3 Example

Get a list of sound files supported by the device

### REQUEST

```
http://192.168.2.188/cgi/event.cgi?act=alarm_voice_file_list
```

### RESPONSE

```
HTTP/1.0 200 OK
```

```
Content-type: text/plain;charset=utf-8
```

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <version version='1' />
  <file id='0' name='tone_Dog_bark' />
  <file id='1' name='tone_phonecall' />
  <file id='2' name='tone_Lion_roars' />
  <file id='3' name='tone_Police_siren' />
  <file id='4' name='tone_Door_dingdong' />
  <file id='5' name='voice_CN_attention' />
  <file id='6' name='voice_You_are_being_monitored' />
  <file id='7' name='Voice_CN_garbage_classification' />
</root>
```

## 11. Playing sound files

### 11.1 Description

Playing sound files.

### 11.2 Grammar

Playing sound files:

```
http://<DeviceIP>/cgi/event.cgi?act=alarm_voice_file_test&data=<?xml version="1.0"
encoding="utf-8"?><root><file id='0' count='1' /></root>
```

## 11.3Parameter

Parameter	Value	Description
<file>	practical	
id	practical	Sound ID
count	practical	Indicates the number of times the sound is played, up to 20 times

## 11.4 Example

### REQUEST

```
http://192.168.2.104/cgi/event.cgi?act=alarm_voice_file_test&data=<?xml version="1.0"
encoding="utf-8"?><root><file id='0' count='1' /></root>
```

### RESPONSE

*HTTP/1.0 200 OK*

*Content-type: text/plain;charset=utf-8*

```
<?xml version="1.0" encoding="utf-8"?>
<root>
  <code>0</code>
  <error>successful</error>
</root>
```

# Appendix

## 1 Save Type

Definition	Value
None	0x0
Save to SD Card	0x1
Save to FTP	0x2
Send to Email	0x4
Send to Alarm Channel	0x8

## 2 Action Type

Definition	Value
None	0
Snap	1
Record	2
PTZ	3
Alarm out	4
Reboot	5

## 3 PTZ Operation Type (Ptz command type)

Definition	Value
Preset	24
Tour	29
Scan	108

## 4 PTZ Appendix

Definition	Value
0	Up (param2: speed )
1	Down (param2: speed )
2	left (param1: speed )
3	Right (param1: speed)
4	Left up (param1、 param2: speed )
5	Left down (param1、 param2: speed )
6	Right up (param1、 param2: speed)
7	Right down( param1、 param2: speed)
8	Focus in
9	Focus out
10	Iris in
11	Iris out
12	Zoom in
13	Zoom out
14	Stop
15	Scan open
16	Scan close
17	Light open
18	Light close
19	Wiper open
20	Wiper close
21	Water open
22	Water close



23	Set preset (param2: Preset ID)
24	Call preset (param2: Preset ID)
25	Clear preset (param2: Preset ID)
26	Rotate 180 degree
27	Return to Zero position
28	Reset
29	Start tour (param1: Tour ID)
30	Stop tour (param1: Tour ID)
31	
32	
33	
100	Top limit
101	Bottom limit
102	Left limit
103	Right limit
104	Insert preset into tou
105	Remove preset from tour
106	Start record tracking (param1: Track ID )
107	Stop record tracking (param1: Track ID )
108	Run tracking (param1: Track ID)
109	Stop tracking (param1: Track ID)
110	Set pan position
111	Tilt position

112	Up limit clear
113	Down limit clear
114	Left limit clear
115	Right clear
116	

## 5 Alarm type

Define	Value
Motion detection	0x00000002
Alarm in	0x00000004
Video loss	0x00000008
Network disconnection	0x00000010
Human detection	0x00000040
Vehicle detection	0x00000080
Face detection	0x00000200
CrossLine human detection	0x00000400
CrossLine vehicle detection	0x00000800
Perimeter protection human detection	0x00001000
Perimeter protection vehicle detection	0x00002000
Pet detection	0x00008000
Crossline pet detection	0x00010000
perimeter protection pet detection	0x00020000