

HTTP API of Network

Version 3.0.2

版本说明			
版本	描述	日期	作者
V3.0.1		2023/07/25	匡颖君
V3.0.2	默认使用 Digest 认证	2024/12/21	Shawn

目录

1 概述	5
2 设备本地网络配置 (Eth Config)	6
2.1 描述	6
2.2 语法	6
2.3 参数	6
2.4 示例	7
2.4.1 获取设备本地网络参数	7
2.4.2 设置设备本地网络参数	7
3 设备时间配置 (time)	9
3.1 描述	9
3.2 语法	9
3.3 参数	9
3.4 示例	10
3.4.1 获取设备时间	10
3.4.2 设置设备时间	10
4 NTP 配置	12
4.1 描述	12
4.2 语法	12
4.3 参数	12
4.4 示例	13
4.4.1 获取 NTP 参数	13
4.4.2 设置 NTP 参数	14
5 DDNS 配置	15
5.1 描述	15
5.2 语法	15
5.3 参数	15
5.4 示例	16
5.4.1 获取 DDNS 参数	16

5.4.2 设置 DDNS 参数	16
6 SMTP 配置	18
6.1 描述	18
6.2 语法	18
6.3 参数	18
6.4 示例	19
6.4.1 获取 SMTP 参数	19
6.4.2 设置 SMTP 参数	19
7 FTP 配置	21
7.1 描述	21
7.2 语法	21
7.3 参数	21
7.4 示例	22
7.4.1 获取 FTP 参数	22
7.4.2 设置 FTP 参数	22
8 Onvif 配置	24
8.1 描述	24
8.2 语法	24
8.3 参数	24
8.4 示例	25
8.4.1 获取 onvif 参数	25
8.4.2 设置 onvif 参数	25
9 串口配置 (Uart config)	27
9.1 描述	27
9.2 语法	27
9.3 参数	27
9.4 示例	28
9.4.1 获取 uart 参数	28
9.4.2 设置 uart 参数	28
10 串口透明传输	30

10.1 描述	30
10.2 语法	30
10.3 示例	30
10.3.1 向串口发送数据	30
11 RTP 配置	31
11.1 描述	31
11.2 语法	31
11.3 参数	31
11.4 示例	32
11.4.1 获取 RTP 参数	32
11.4.2 设置 RTP 参数	32

1 概述

Network CGI，可以请求获取或设置本地网络、time、ntp、ddns、smtp、ftp、onvif、Uart、串口透传、rtp 等

关于配置参数：当配置成功时，将收到返回值 successful 字符，错误时将收到由错误信息组成的 Xml 格式数据。

2 设备本地网络配置（Eth Config）

2.1 描述

该指令将获取由网络配置参数组成的 Xml 格式数据，参数包含 IPv4 地址、子网掩码、网关、mac 地址等。

2.2 语法

获取本地网络配置：

```
http://<Device IP>/cgi/network.cgi?get=eth
```

设置本地网络配置：

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

注：当设置本地网络参数中带有 IP 参数时，设备将改变 IP 地址

2.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<eth>	无	表示本地网络参数的父节点
ip	practical	表示 IPv4 地址
submask	practical	表示子网掩码
gateway	practical	表示网关
ondhcp	index	表示自动获取 DNS 开关。0 表示关闭，1 表示打开
ip6	practical	表示 IPv6 地址

submask6	practical	表示 IPv6 子网掩码
gateway6	practical	表示 IPv6 网关
dhcp_mode	index	内部使用，可忽略
dhcp_fixed_time	practical	内部使用，可忽略
<dns>	practical	表示首选 DNS 服务器
ondhcp	index	表示自动获取 DNS 开关。0 表示关闭，1 表示打开
dns1	practical	表示首选 DNS 服务器
dns2	practical	表示备用 DNS 服务器

2.4 示例

2.4.1 获取设备本地网络参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=eth
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<eth ondhcp='0' ip='192.168.2.172' submask='255.255.255.0' gateway='192.168.2.1'
ip6='' submask6='' gateway6='' dhcp_mode='0' dhcp_fixed_time='600' >
<dns ondhcp='0' dns1='8.8.8.8' dns2='8.8.8.8' />
</eth>
</root>
```

2.4.2 设置设备本地网络参数

示例：修改设备的 ip 地址为 192.168.2.171，子网掩码为 255.255.255.0，默认网关为

192.168.2.1

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=eth&data=<?xml version="1.0"
encoding="utf-8"?><root><eth ondhcp='0' ip='192.168.2.171'
submask='255.255.255.0' gateway='192.168.2.1' ></eth></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 设备时间配置（time）

3.1 描述

该指令将获取由时间参数组成的 Xml 格式数据，参数包含年、月、日、星期、时、分、秒等。

3.2 语法

获取设备时间配置：

```
http://<Device IP>/cgi/network.cgi?get=time
```

设置设备时间配置：

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

3.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<time>	无	所有时间参数的父节点
year	practical	表示年份
month	practical	表示月份
day	practical	表示日份
week	practical	表示星期数
hour	practical	表示小时
minute	practical	表示分钟

second	practical	表示秒数
--------	-----------	------

3.4 示例

3.4.1 获取设备时间

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=time
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<time year='2022' month='9' day='16' week='5' hour='15' minute='34' second='48' />
</root>
```

3.4.2 设置设备时间

示例：修改设备的时间为 2013 年 8 月 11 日

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=time&data=<?xml version="1.0"
encoding="utf-8"?><root><time year='2013' month='8' day='11' /></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 NTP 配置

4.1 描述

该指令将获取由 NTP 配置参数组成的 Xml 格式数据，参数包含 ntp 开关、时区、ntp 服务器地址等。

4.2 语法

获取 NTP 配置：

```
http://<Device IP>/cgi/network.cgi?get=ntp
```

设置 NTP 配置：

```
http://<Device IP>/cgi/network.cgi?set=ntp&data=<?xml version="1.0" encoding="utf-8"?><root><ntp xxx><summer xxx><begin xxx /><end xxx /></ntp></root>
```

4.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<ntp>	无	表示所有 ntp 参数的父节点
on	practical	表示 ntp 开关。0 表示关闭，1 表示打开
timezone	practical	表示设备时区
server	practical	表示 NTP IP 地址
clientindex	practical	表示索引文件，内部使用，可忽略
ntpport	practical	表示 NTP 端口
reftime	practical	表示 ntp 定时校正，单位：分钟

<summer>		表示所有 summer 参数的父节点
on	practical	表示夏令时启用开关。0 表示关闭，1 表示打开
advancetime	practical	表示夏令时偏移时间，单位：分钟
begin	无	表示夏令时开始日期的父节点
end	无	表示夏令时结束日期的父节点
week_index	practical	表示触发、结束夏令时在哪一周，范围（1-5）
week	practical	表示触发、结束夏令时在星期几，范围（1-7）
hour	practical	表示触发、结束夏令时在哪个小时，范围（0-23）
minute	practical	表示触发、结束夏令时在哪一分钟，范围（0-59）
month	practical	表示触发、结束夏令时在哪个月，范围（1-12）

4.4 示例

4.4.1 获取 NTP 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=ntp
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='0' />
<ntp on='0' timezone='8.00' reftime='10' server='pool.ntp.org' clientindex='226'
ntpport='123'>
<summer on='0' advancetime='60'>
<begin month='3' week_index='2' week='7' hour='2' minute='0' />
<end month='11' week_index='1' week='7' hour='2' minute='0' />
</summer>
```

```
</ntp>
</root>
```

4.4.2 设置 NTP 参数

示例：修改 NTP 的端口号为 1234

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=ntp&data=<?xml version="1.0"
encoding="utf-8"?><root><ntp ntpport='1234'></ntp></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 DDNS 配置

5.1 描述

该指令将获取由 DDNS 配置参数组成的 Xml 格式数据，参数包含 ddns 开关、注册域名、用户名密码、ddns 服务端口等。

5.2 语法

获取 ddns 配置：

```
http://<Device IP>/cgi/network.cgi?get=ddns
```

设置 ddns 配置：

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

5.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<ddns>	无	所有 ddns 参数的父节点
on	practical	表示 ddns 开关。0 表示关闭，1 表示打开
account	practical	表示 ddns 注册域名
pwd	practical	表示用户密码
name	practical	表示用户名
port	practical	表示 ddns 服务器端口
times	practical	表示 ddns 更新时间

type	practical	表示 ddns 类型，0 表示 dyndns，1 表示 3322
------	-----------	----------------------------------

5.4 示例

5.4.1 获取 DDNS 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=ddns
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<ddns on='0' type='0' account='test.dyndns.org' port='81' name='123' pwd='1235'
times='3600' />
</root>
```

5.4.2 设置 DDNS 参数

示例：修改 DDNS 服务器端口为 98

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=ddns&data=<?xml version="1.0"
encoding="utf-8"?><root><ddns port=98 /></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 SMTP 配置

6.1 描述

该指令将获取由 smtp 配置参数组成的 Xml 格式数据，参数包含 stmp 服务器地址、smtp 端口、发件人、用户名密码、收件人等。

6.2 语法

获取 smtp 配置：

```
http://<Device IP>/cgi/network.cgi?get=smtp
```

设置 smtp 配置：

```
http://<Device IP>/cgi/network.cgi?set=smtp&data=<?xml version="1.0" encoding="utf-8"?><root><smtp ch=0 xxx /></root>
```

6.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<smtp>	无	所有 smtp 参数的父节点
ch	index	表示当前通道
hub	practical	表示 stmp 服务器地址
addr	practical	表示发送人账号
recipient_addr	practical	表示收件人账号，如果有多个，则用 “,” 隔开
user	practical	表示用户名
psw	practical	表示用户密码

title	practical	表示邮件标题
port	practical	表示 smtp 服务器端口
enable_ssl	practical	表示 SSL 开关。0 表示关闭，1 表示启用

6.4 示例

6.4.1 获取 SMTP 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=smtp
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<smtp ch='0' hub='' addr='' user='' psw='' recipient_addr='' port='25' title=''
enable_ssl='0' >
</smtp>
</root>
```

6.4.2 设置 SMTP 参数

示例：修改通道 1 中 SMTP 服务器端口为 80

提示：必须指明通道信息

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=smtp&data=<?xml version="1.0"
encoding="utf-8"?><root><smtp ch=0 port=80 /></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>
```

7 FTP 配置

7.1 描述

该指令将获取由 FTP 配置参数组成的 Xml 格式数据，参数包含 ftp 服务器地址、路径、用户名密码、FTP 端口等。

7.2 语法

获取 FTP 配置：

```
http://<Device IP>/cgi/network.cgi?get=ftp
```

设置 FTP 配置：

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

7.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<ftp>	无	所有 ftp 参数的父节点
ch	index	表示当前通道
path	practical	表示 ftp 服务器下的文件路径
psw	practical	表示 ftp 登录密码
user	practical	表示 ftp 登录账号
url	practical	表示 ftp 服务器地址
port	practical	表示 ftp 服务器端口

7.4 示例

7.4.1 获取 FTP 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=ftp
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='0' />
<ftp ch='0' url='' user='' psw='' port='21' path=''>
</ftp>
</root>
```

7.4.2 设置 FTP 参数

示例：修改通道 1 中 FTP 服务器端口为 23

提示：必须指明通道信息

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=ftp&data=<?xml version="1.0"
encoding="utf-8"?><root><ftp ch=0 port=23 ></ftp></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 Onvif 配置

8.1 描述

该指令将获取由 onvif 配置参数组成的 Xml 格式数据，参数包含 onvif 开关、对讲开关、元数据告警开关等。

8.2 语法

获取 onvif 配置：

```
http://<Device IP>/cgi/network.cgi?get=onvif
```

设置 onvif 配置：

```
http://<Device IP>/cgi/network.cgi?set=onvif<?xml version="1.0" encoding="GBK"
"?><root><onvif xxx /></root>
```

8.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<onvif>	无	表示所有 onvif 参数的父节点
on	practical	表示 onvif 开关。0 表示关闭，1 表示启用
enabletalk	practical	表示对讲开关。0 表示关闭，1 表示启用
synctime	practical	表示允许同步时间开关。0 表示关闭，1 表示启用
syncparam	practical	表示允许同步参数开关。0 表示关闭，1 表示启用
enablemeta	practical	表示元数据告警开关。0 表示关闭，1 表示启用
authorized	practical	表示进行身份验证开关。0 表示关闭，1 表示启用

companyname	practical	内部使用，可忽略
hostname	practical	表示主机名称，内部使用，可忽略
location	practical	表示引用包含完整的服务接口定义的 WSDL 文档的 URL，内部使用，可忽略
wlanip	practical	表示外网 IPV4 地址
compatiblemode	practical	内部使用，可忽略

8.4 示例

8.4.1 获取 onvif 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=onvif
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="GBK"?>
<root>
<version version='1' />
<onvif on='1' enabletalk='1' synctime='1' syncparam='1' enablemeta='1'
authorized='0' companyname='' hostname='' location='' wlanip='0.0.0.0'
compatiblemode='0' />
</root>
```

8.4.2 设置 onvif 参数

示例：修改 Onvif 的开关为开启

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=onvif&data=<?xml version="1.0"
encoding="GBK"?><root><onvif on='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>
```

9 串口配置 (Uart config)

9.1 描述

该指令将获取由 Uart 配置参数组成的 Xml 格式数据，参数包含波特率设置、停止位、数据位设置等。

9.2 语法

获取 uart 配置：

```
http://<Device IP>/cgi/network.cgi?get=uart
```

设置 uart 配置：

```
http://<Device IP>/cgi/network.cgi?set=uart&data=<?xml version="1.0" encoding="utf-8"?><root><uart ch='1' xxx /></root>
```

9.3 参数

参数值有三种形式，一种为实际值 (practical)，另一种为索引 (index)，还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<uart>	无	表示 uart 参数的父节点
ch	index	表示 uart 所属通道，值等于 0 时为 rs232 通道 大于 0 时为 rs485 通道
buadrate	practical	表示串口波特率
databit	practical	表示串口数据位长度。0 表示 5 位，1 表示 6 位，2 表示 7 位，其他表示 8 位
stopbit	practical	表示停止位。1 表示 1 位，2 表示 2 位，其他表示 1

		位
parity	practical	表示校验位。0 表示无校验，1 表示奇校验，2 表示偶校验，3 表示恒 1，4 表示恒 0；
flowctrl	practical	暂不使用
devname	practical	表示以太网设备的名字，内部使用，可忽略
gpio	practical	表示通用型输入输出，内部使用，可忽略
useptz	index	表示 uart 的开关。0 表示关闭，1 表示开启

9.4 示例

9.4.1 获取 uart 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=uart
```

RESPONSE

```
HTTP/1.0 200 OK
Content-type: text/plain;charset=utf-8

<?xml version="1.0" encoding="utf-8"?>
<root>
<uart ch='0' databit='3' stopbit='1' parity='0' flowctrl='0' buadrate='57600'
devname='/dev/ttyS2' gpio='0' useptz='1' />
</root>
```

9.4.2 设置 uart 参数

示例：修改 rs485 通道的串口数据位长度为 6 位，串口波特率为 300

提示：必须指明通道信息

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=uart&data=<?xml version="1.0"
```

```
encoding="utf-8"?><root><uart ch='1' databit='1' buadrate='300' /></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>
```

10 串口透明传输

10.1 描述

该指令将直接向串口发送数据。

10.2 语法

发送数据到串口：

```
http://<Device IP>/cgi/network.cgi?act=uart_data&data=123456
```

10.3 示例

10.3.1 向串口发送数据

REQUEST

```
http://192.168.2.172/cgi/network.cgi?act=uart_data&data=123456
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

11 RTP 配置

11.1 描述

该指令将获取由 rtp 配置参数组成的 Xml 格式数据，参数包含发送码流、服务器地址、端口、最大传输单元等。

11.2 语法

获取 rtp 配置：

```
http://<Device IP>/cgi/network.cgi?get=rtp
```

设置 rtp 配置：

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

11.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<rtp>	无	表示 rtp 参数的父节点
enable	practical	表示当前通道的 rtp 开关。0 表示关闭，1 表示启用
rtpport	practical	表示 rtp 接收端口
mtu	practical	表示最大传输单元
url	practical	表示 rtp 服务器 url
streamid	practical	表示使用哪一路码流进行推送

11.4 示例

11.4.1 获取 RTP 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=rtp
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='0' />
<rtp enable='0' streamid='1' rtpport='5000' url='' mtu='1400' />
</root>
```

11.4.2 设置 RTP 参数

示例：修改码流 2 中的 RTP 的端口为 1500

提示：必须指明码流信息

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=rtp&data=<?xml version="1.0"
encoding="utf-8"?><root><rtp streamid='1' rtpport='1500' /></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>
```

12 RTMP 配置

12.1 描述

该指令将获取由 rtmp 配置参数组成的 Xml 格式数据，参数包含码流、rtmp 服务器地址、rtmp 端口、应用名称、流 ID 等。

12.2 语法

获取 rtmp 配置：

```
http://<Device IP>/cgi/network.cgi?get=rtmp
```

设置 rtmp 配置：

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

12.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<rtmp>	无	所有 rtmp 参数的父节点
on	index	表示 rtmp 开关。0 表示关闭，1 表示启用
<url>	无	表示 URL 的节点
ch	index	表示当前通道
streamtype	practical	码流类型
serverip	practical	表示 rtmp 服务器地址
port	practical	表示端口

appname	practical	表示应用名称
streamid	practical	表示流 ID

12.4 示例

12.4.1 获取 RTMP 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=rtmp
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="GBK"?>
<root>
<rtmp on='1'>
<url ch='0' streamtype='0' serverip='' port='0' appname='' streamid='' /><url
ch='0' streamtype='1' serverip='' port='0' appname='' streamid='' /><url ch='0'
streamtype='2' serverip='192.168.2.28' port='1234' appname='live' streamid='123'
/></rtmp>
</root>
```

12.4.2 设置 RTMP 参数

示例：关闭 RTMP 服务

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=rtmp&data=<?xml version="1.0"
encoding="utf-8"?><root><rtmp on=0 /></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>

13 NAS 配置

13.1 描述

该指令将获取由 NAS 配置参数组成的 Xml 格式数据，参数包含 nas 服务器地址、路径、文件系统等。

13.2 语法

获取 NAS 配置：

```
http://<Device IP>/cgi/network.cgi?get=nas
```

设置 NAS 配置：

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

13.3 参数

参数值有三种形式，一种为实际值（practical），另一种为索引（index），还有一种无值，它一般为其他参数的父节点，用于表示对应的通道。

参数列表如下：

参数	值	描述
<nas>	无	所有 nas 参数的父节点
ch	index	表示当前通道
server	practical	表示 nas 服务器地址
path	practical	表示 nas 服务器下的文件路径，目录前强制添加根目录 “/”
file_system	practical	表示文件系统

13.4 示例

13.4.1 获取 NAS 参数

REQUEST

```
http://192.168.2.172/cgi/network.cgi?get=nas
```

RESPONSE

```
HTTP/1.0 200 OK
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<root>
<version version='0' />
<nas ch='0' server='' path='/' file_system='0'>
</nas>
</root>
```

13.4.2 设置 NAS 参数

示例：修改通道 1 中 NAS 服务器端口为 192.168.2.7

提示：必须指明通道信息

REQUEST

```
http://192.168.2.172/cgi/network.cgi?set=nas&data=<?xml version="1.0"
encoding="utf-8"?><root><nas ch=0 server=192.168.2.7 ></nas></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>
```