



## **Configuring Decoder's Remote Address with HTTP API**

# 1. Overview

This document describes how to configure Decoder's remote address which corresponds to the address of IP camera or Encoder using HTTP API. Configuring remote address is necessary when Decoder is switched to IP camera or Encoder dynamically.

For basic usage of HTTP API, please refer to another document - "Truen HTTP API.doc".

## 2. Configuration parameters for remote address

### Decoder

Parameters	Description
NET_REMOTETYPE RMT_REMOTETYPE_0 RMT_REMOTETYPE_1 RMT_REMOTETYPE_2 RMT_REMOTETYPE_3	Remote(protocol) type 0: Normal (Truen proprietary) 1: RTSP 2: RTP 3: MPEG-TS 4: Onvif 5: RTMP
NET_PROTOCOL RMT_PROTOCOL_0 RMT_PROTOCOL_1 RMT_PROTOCOL_2 RMT_PROTOCOL_3	Media protocol: 0: TCP 1: UDP 2: Multicast
NET_DOMAINNAME RMT_DOMAINNAME_0 RMT_DOMAINNAME_1 RMT_DOMAINNAME_2 RMT_DOMAINNAME_3	IP address or URL of IP camera or Encoder to be connected. In case of RTSP URL, URL without RTSP protocol prefix is also allowed. rtsp://192.168.10.100/video1 192.168.10.100/video1 rtsp://192.168.10.100:8554.video1 192.168.10.100:8554.video1
NET_REMOTEPORT RMT_REMOTEPORT_0 RMT_REMOTEPORT_1 RMT_REMOTEPORT_2 RMT_REMOTEPORT_3	Port to be used. When the URL configured in NET_DOMAINNAME(or RMT_DOMAINNAME_X) contains a port number, this port setting is ignored.
RMT_SWDISABLE_0 RMT_SWDISABLE_1 RMT_SWDISABLE_2 RMT_SWDISABLE_3	Connection is overlapped for seamless switching when remote address is changed: 0: Overlapped connection enabled 1: Overlapped connection disabled

\*NET\_REMOTETYPE is for single channel decoder (TCS-300, TCS-2000 and old F/W of TCS-1500/TCS-3500 which doesn't support quad-viewing), while RMT\_REMOTETYPE\_0, RMT\_REMOTETYPE\_1, RMT\_REMOTETYPE\_2, RMT\_REMOTETYPE\_3 are for the decoders supporting quad-view (Recent F/W of TCS-1500/TCS-3500). This relationship is common for other parameters: NET\_PROTOCOL, NET\_DOMAINNAME etc.

Remote #1	Remote #2	Remote #3	Remote #4
Remote Type <input type="text" value="RTSP"/>			
Media Protocol <input type="text" value="TCP"/>			
Address <input type="text" value="192.168.32.172"/>			
Port <input type="text" value="2222"/>			
Overlapped Connection <input type="text" value="Enable"/> (Keep previous connection until new connection is established.)			

### 3. HTTP API Examples

Examples are shown for RTSP protocols. Similar scheme can be applied for other protocols.

#### **Setting only Address when default RTSP port is used**

[http://192.168.10.100/httpapi/WriteParam?action=writeparam&RMT\\_DOMAINNAME\\_0=rtsp://192.168.10.200/video1](http://192.168.10.100/httpapi/WriteParam?action=writeparam&RMT_DOMAINNAME_0=rtsp://192.168.10.200/video1)

#### **Setting only Address when a different RTSP port is used**

[http://192.168.10.100/httpapi/WriteParam?action=writeparam&RMT\\_DOMAINNAME\\_0=rtsp://192.168.10.200:8554/video1](http://192.168.10.100/httpapi/WriteParam?action=writeparam&RMT_DOMAINNAME_0=rtsp://192.168.10.200:8554/video1)

#### **Setting multiple parameters in a command**

[http://192.168.10.100/httpapi/WriteParam?action=writeparam&RMT\\_REMOTETYPE\\_0=1&RMT\\_PROTOCOL\\_0=1&RMT\\_DOMAINNAME=rtsp://192.168.10.200/video1](http://192.168.10.100/httpapi/WriteParam?action=writeparam&RMT_REMOTETYPE_0=1&RMT_PROTOCOL_0=1&RMT_DOMAINNAME=rtsp://192.168.10.200/video1)

#### **Reading the address of remote peer**

[http://192.168.32.212/httpapi/ReadParam?action=readparam&RMT\\_DOMAINNAME\\_0=0](http://192.168.32.212/httpapi/ReadParam?action=readparam&RMT_DOMAINNAME_0=0)