IP Installer

Hanwha

SUNAPI

v2.6.2
2023-04-07

Contact Information

Hanwha Vision Co., Ltd.
Hanwha Vision 6, Pangyo-ro 319beon-gil, Bundang-gu, Seongnam-si, Gyeonggi-do, 13488, KOREA
www.hanwhavision.com

Hanwha Vision America
500 Frank W. Burr Blvd. Suite 43 Teaneck, NJ 07666
hanwhavisionamerica.com

Hanwha Vision Europe
Heriot House, Heriot Road, Chertsey, Surrey, KT16 9DT, United Kingdom
hanwhavision.eu

Hanwha Vision Middle East FZE
Jafza View 18, Office 2001-2003, Po Box 263572, Jebel Ali Free Zone, Dubai, United Arab Emirates
www.hanwhavision.com/ar

Table of Contents

1. Preface . . . 4
2. Introduction . . . 5
3. IPv4 . . . 6
   3.1. Communicate Port . . . 6
   3.2. SendData Format (DEPRECATED) . . . 6
   3.3. RecvData Format(DEPRECATED) . . . 7
   3.4. IP Scan(DEPRECATED) . . . 10
      3.4.1. Request . . . 10
      3.4.2. Response(DEPRECATED) . . . 10
   3.5. IP Setting(DEPRECATED) . . . 11
      3.5.1. Request . . . 11
      3.5.2. Response . . . 12
   3.6. SendData Format for SUNAPI . . . 13
   3.7. RecvData Format for SUNAPI . . . 15
   3.8. IP Scan for SUNAPI . . . 18
      3.8.1. Request . . . 18
      3.8.2. Response . . . 19
   3.9. SCAN Uninitialized Devices for RSA Key . . . 21
      3.9.1. Request . . . 21
      3.9.2. Response . . . 22
   3.10. IP Setting for SUNAPI . . . 22
      3.10.1. Request . . . 22
      3.10.2. Response . . . 23
   3.11. Set Password in factory default state(SUNAPI) . . . 25
      3.11.1. Request . . . 25
      3.11.2. Response . . . 25
4. IPv6 . . . 27
   4.1. Communicate Port . . . 27
   4.2. SendData Format(DEPRECATED) . . . 27
   4.3. RecvData Format(DEPRECATED) . . . 29
   4.4. IP Scan(DEPRECATED) . . . 31
      4.4.1. Request . . . 31
      4.4.2. Response . . . 32
   4.5. IP Setting(DEPRECATED) . . . 33
      4.5.1. Request . . . 33
      4.5.2. Response . . . 33

4.6. SendData Format for SUNAPI . . . 34
4.7. RecvData Format for SUNAPI . . . 36
4.8. IP Scan for SUNAPI . . . 40
4.8.1. Request . . . 40
4.8.2. Response . . . 40
4.9. SCAN Uninitialized Devices for RSA Key . . . 42
4.9.1. Request . . . 42
4.9.2. Response . . . 43
4.10. IP Setting for SUNAPI . . . 44
4.10.1. Request . . . 44
4.10.2. Response . . . 45
4.11. Set password in factory default state (SUNAPI). . . . 46
4.11.1. Request . . . 46
4.11.2. Response . . . 47
5. Annex A . . . 50
5.1. Updated Structure Used . . . 50
6. Revision History . . . 54

Chapter 1. Preface

Purpose
This protocol enables client to discover camera, configure IP address, port and intial password.

Audience
This document is intended for clients that needs to discover and setup Hanwha Devices.

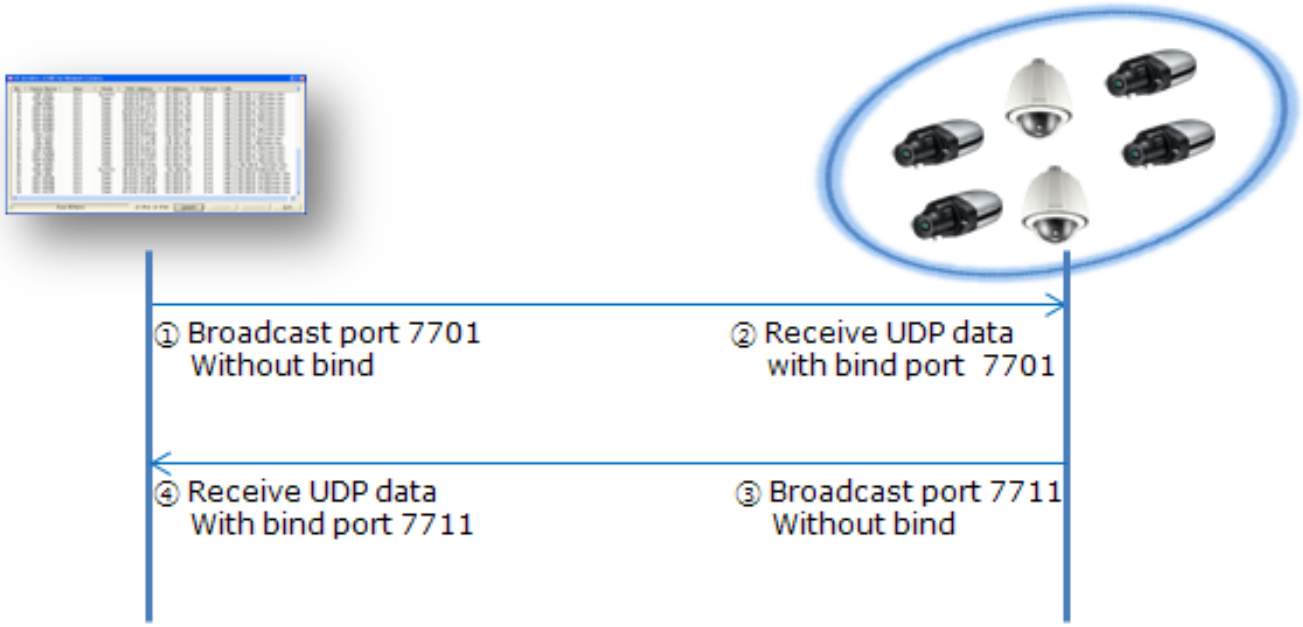
Scope
This document describes the IP installer protocol used to discover and configure camera (mainly initial setup)

Chapter 2. Introduction

This document describes the IP-Installer (IP-Scanner) protocol guide.
Some of Hanwha Vision’s Network Device support IPv4 and IPv6. Some of device support only IPv4.
This document describes both IPv4 and IPv6 protocol.

Chapter 3. IPv4

3.1. Communicate Port



3.2. SendData Format (DEPRECATED)

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chIP	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway
chPassword	char[20]	20	Password
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, it only is 10 character in the whole model name)

Name	Type	Byte size	Description
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V4_OLD
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chPassword[20];
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
} DATAPACKET_V4_OLD;
```

3.3. RecvData Format(DEPRECATED)

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chip	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway
chPassword	char[20]	20	Password
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address
chAlias	char[64]	64	Alias name for NVR or Encoder only • Network camera does not include this field in response packet.

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V4
{
```

```
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chPassword[20];
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
    char chAlias[64];
} DATAPACKET_V4;
```

<Structure Field description>

Table 1. nMode

Mode Definition	VALUE
DEF_REQ_SCAN	1
DEF_REQ_APPLY	2
DEF_REQ_PORTMAPPING	4
DEF_RES_SCAN	11
DEF_RES_APPLY	22
DEF_RES_PASSWORD_ERR	33
DEF_RES_PORT_MAPPING	44
DEF_RES_PORT_MAPPING_ERR	55
DEF_RES_ROUTER_CONN_ERR	66
DEF_RES_APPLY_ERR	77

chPacketID

This value is used to identify Client. Hanwha use this value as unique ID derived from MAC address of PC and random value.

3.4. IP Scan(DEPRECATED)

3.4.1. Request

Field Name	Value
nMode	1 (DEF_REQ_SCAN)
chPacketID	Unique value of client
chMAC	Unused
chIP	Unused
chSubnetMask	Unused
chGateway	Unused
chPassword	Unused
nPort	Unused
nStatus/nVersion	Unused, 0x08 : when client passes this version and if device don’t support SVNP ignore ipscan response.
chDeviceName	Unused
nHttpPort	Unused
nDevicePort	Unused
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Unused
chDDNS	Unused

3.4.2. Response(DEPRECATED)

Field Name	Value
nMode	11 (DEF_RES_SCAN)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chPassword	Unused
nPort	HTTP port for web-connection

Field Name	Value
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	DDNS URL*
chAlias	Alias[2] of NVR or Encoder (In case of NW camera, unused)

- **DDNS URL**[1] has the DDNS URL. If camera registered successfully the URL to DDNS server, this field is filled by the registered URL. If camera fails to register, this field is filled by the URL that is made based on the IP. (ex : http://192.168.1.200:8080)
- **Alias**[2] is used to distinguish each device of NVR/Encoder. NVR/Encoder But Network camera does not include ‘chAlias' field in response packet.
- **nDevicePort** : Port number to connect using SVNP, VNP or SSNP protocol.
- **nTcpPort** : Port number to get stream via tcp. This port is valid only if Client uses VNP.
- **nUDPPort** : Port number to get stream via udp. This port is valid only if Client uses VNP.
- **nUploadPort** : Port number to upload camera’s f/w (TCP). This port is valid only if Client uses VNP.
- **nMulticastPort** : Port number to get stream via multicast. This port is valid only if Client uses VNP.

3.5. IP Setting(DEPRECATED)

3.5.1. Request

Field Name	Value
nMode	2 (DEF_REQ_APPLY)
chPacketID	Unique value of client
chMAC	MAC address of Camera
chIP	IP address of Camera. Valid only if (nNetworkMode == 0)

Field Name	Value
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chPassword	Password for connecting to camera with admin privilege
nPort	The same value of nHttpPort
nStatus	Unused
chDeviceName	Unused
nHttpPort	Port number to change
nDevicePort	Port number to connect using SVNP
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused

- Using this command, client can change the IP type, address and port number.

3.5.2. Response

Field Name	Value
nMode	22 (DEF_RES_APPLY)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chPassword	Unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document

Field Name	Value
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused
chAlias	Alias of NVR or Encoder (In case of NW camera, unused)

- Changed values are filled.

3.6. SendData Format for SUNAPI

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chIP	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway
chPassword	char[20]	20	Password
is_only_support_sunapi	char	1	If device supports only SUNAPI and not SVNP
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port

Name	Type	Byte size	Description
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address
chAlias	Char[32]	32	Camera name
chNewModelName	Char[32]	32	New model Name(since April 2016)
nModelType	char	1	Model Type
nVersion	Unsigned short	2	version
nHttpMode	Char	1	Http Mode
nHttpsPort	Unsigned short	2	HTTPS Port
nSupportedProtocol	Char	1	Supported Protocol
nPasswordStatus	Char	1	Password Status

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V4_OLD
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chPassword[20];
    char is_only_support_sunapi;
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
```

```
    char chAlias[32];
    char chNewModelName[32];
    char nModelType;
    unsigned short nVersion;
    char nHttpMode;
    unsigned short nHttpsPort;
    char nSupportedProtocol;
    char nPasswordStatus;
} DATAPACKET_V4_EXT;
```

3.7. RecvData Format for SUNAPI

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chip	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway
chPassword	char[20]	20	Password
is_only_support_sunapi	char	1	If only sunapi is supported by device
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port

Name	Type	Byte size	Description
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address
chAlias	char[32]	32	Alias name for NVR or Encoder only • Network camera does not include this field in response packet.
chNewModelName	Char[32]	32	New model Name(since April 2016)
nModelType	char	1	Model Type
nVersion	Unsigned short	2	version
nHttpMode	Char	1	Http Mode
nHttpsPort	Unsigned short	2	HTTPS Port
nSupportedProtocol	Char	1	Supported Protocol
nPasswordStatus	Char	1	Password Status

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V4
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chPassword[20];
    char is_only_support_sunapi;
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
```

```
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
    char chAlias[32];
    char chNewModelName[32];
    char nModelType;
    unsigned short nVersion;
    char nHttpMode;
    unsigned short nHttpsPort;
    char nSupportedProtocol;
    char nPasswordStatus;
} DATAPACKET_V4_EXT;

typedef struct tagRsaScanResponse
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char MaxPasswordLen;
    char Reserved[10];
    char PayloadSize[2];
    char Payload[payloadSize];
} DATAPACKET_RSA_RESPONSE;

typedef struct tagApplyPasswordRequest
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char Reserved[10];
    char PayloadSize[2];
    char Payload[payloadSize];
} DATAPACKET_APPLY_PASSWORD_REQUEST;
```

<Structure Field description>

Table 2. nMode

Mode Definition	VALUE
DEF_REQ_SCAN_EXT	6
DEF_REQ_APPLY_EXT	7
DEF_REQ_SCAN_RSA	8
DEF_REQ_APPLY_PASSWORD	9
DEF_RES_SCAN_EXT	12
DEF_RES_SCAN_RSA	13
DEF_RES_APPLY_EXT	23
DEF_RES_APPLY_PASSWORD_ERR	24
DEF_RES_APPLY_PASSWORD	25
DEF_RES_PASSWORD_ERR	33
DEF_RES_ROUTER_CONN_ERR	66
DEF_RES_APPLY_ERR	77

chPacketID

This value is used to identify Client. Hanwha use this value as unique ID derived from MAC address of PC and random value.

3.8. IP Scan for SUNAPI

3.8.1. Request

Field Name	Value
nMode	6(DEF_REQ_SCAN_EX)
chPacketID	Unique value of client
chMAC	Unused
chIP	Unused
chSubnetMask	Unused
chGateway	Unused
chPassword	Unused
is_only_support_sunapi	Unused
nPort	Unused
nStatus/nVersion	Unused, 0x08 : when client passes this version and if device don’t support SNVP ignore ipscan response.
chDeviceName	Unused
nHttpPort	Unused

Field Name	Value
nDevicePort	Unused
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Unused
chDDNS	Unused
chAlias	Unused
nModelType	Unused
nVersion	Unused
nHttpMode	Unused
nHttpsPort	Unused
nSupportedProtocol	Unused
nPasswordStatus	Unused

3.8.2. Response

Field Name	Value
nMode	12(DEF_RES_SCAN_EX)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
Nonce	When nVersion 0x08 is supported will have the nonce value, that can be used for password digest.
is_only_support_sunapi	If device supports only SUNAPI
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document

Field Name	Value
nTcpPort	Unused
nUdpPort	Unused
nUploadPort/SpeakerType	Unused / When Modeltype is set to 0x05 network speaker, SpeakerType can be 0: Slave, 1: Master, 2:Server, 3: Module
MaxChannel	When nVersion 0x08 is supported will have max channel number **Note** For models that support virtual channel addition/deletion, it represents the current channel count.
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	DDNS URL[1]
chAlias	Alias[2] of NVR or Encoder (In case of NW camera, unused)
chNewModelName	New model Name(since April 2016)
nModelType	Device Type (0x00: Camera, 0x01: Encoder, 0x02: Decoder, 0x03: Recorder, 0x04: IOBox,0x05: NetworkSpeaker, 0x06: NetworkMic, 0x07: LEDBox)
nVersion	version : This parameter can be a combination of the following values. • 0x01: Can't change the HTTPS port in web page • 0x02: Can change the HTTPS port in web page • 0x04: Support New Model Name variable • 0x08: SupportPasswordVerification using digest
nHttpMode	HTTP Mode of camera (0x00: HTTP, 0x01: HTTPS)
nHttpPort	HTTPS port for web-connection
nSupportedProtocol	Supported Protocol of camera (0x01: SVNP, 0x02: SUNAPI1.0, 0x04: SUNAPI2.0, 0x08 SUNAPI2.3.1above,0x10:SVP)
nPasswordStatus	Password status of camera (0x00: The camera has a password, 0x01: The camera hasn't a password) When nVersion 0x08 is supported this field can be used to check if device is initialized.

- **DDNS URL**[1] has the DDNS URL. If camera registered successfully the URL to DDNS server, this field is filled by the registered URL. If camera fails to register, this field is filled by the URL that is made based on the IP. (ex : http://192.168.1.200:8080)
- **Alias**[2] is used to distinguish each device of NVR/Encoder. NVR/Encoder But Network camera does not include 'chAlias' field in response packet.
- **nDevicePort** : Port number to connect using SVNP, VNP or SSNP protocol.
- **nTcpPort** : Port number to get stream via tcp. This port is valid only if Client uses VNP.
- **nUDPPort** : Port number to get stream via udp. This port is valid only if Client uses VNP.
- **nUploadPort** : Port number to upload camera's f/w (TCP). This port is valid only if Client uses VNP.
- **nMulticastPort** : Port number to get stream via multicast. This port is valid only if Client uses VNP.

3.9. SCAN Uninitialized Devices for RSA Key

3.9.1. Request

Field Name	Value
nMode	8(DEF_REQ_SCAN_RSA)
chPacketID	Unique value of client
chMAC	Unused
chIP	Unused
chSubnetMask	Unused
chGateway	Unused
chPassword	Unused
nPort	Unused
nStatus/nVersion	Unused
chDeviceName	Unused
nHttpPort	Unused
nDevicePort	Unused
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Unused
chDDNS	Unused
chAlias	Unused
nModelType	Unused

Field Name	Value
nVersion	Unused
nHttpMode	Unused
nHttpsPort	Unused
nSupportedProtocol	Unused
nPasswordStatus	Unused

3.9.2. Response

Field Name	Value
nMode	13(DEF_RES_SCAN_RSA)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
maxPasswordLen	Maximum password length supported. If this field is 0, the maximum password length supported is 15; if it is larger than 0, the value supplied is the maximum password length.
Reserved	
PacketSize	Size of Payload
RSAKeyPayload	RSA Public Key

3.10. IP Setting for SUNAPI

3.10.1. Request

Field Name	Value
nMode	7 (DEF_REQ_APPLY_EX)
chPacketID	Unique value of client
chMAC	MAC address of Camera
chIP	IP address of Camera. Valid only if (nNetworkMode == 0)
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera

Field Name	Value
chPassword	Password for connecting to camera with admin privilege(When version 0x08 is supported, pass sha256(Username:password:nonce) truncated to first 20 bytes of total 32 bytes)
nPort	The same value of nHttpPort
nStatus	Unused
chDeviceName	Unused
nHttpPort	Port number to change
nDevicePort	Port number to connect using SVNP
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused
UserName	Pass the username in this field (When version 0x08)
chNewModelName	Unused
nModelType	Unused
nVersion	Unused,when digest password is used, this should be set to 0x08
nHttpMode	Unused
nHttpPort	Port number to change (if version value is 2)
nSupportedProtocol	Unused
nPasswordStatus	Unused

- Using this command, client can change the IP type, address and port number.

3.10.2. Response

Field Name	Value
nMode	23(DEF_RES_APPLY_EXT)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera

Field Name	Value
chPassword	Unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort/SpeakerType	Unused / When Modeltype is set to 0x05 network speaker, SpeakerType can be 0: Slave, 1: Master, 2:Server, 3: Module
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused
chAlias	Alias of NVR or Encoder (In case of NW camera, unused)
chNewModelName	New model Name(since April 2016)
nModelType	Device Type (0x00: Camera, 0x01: Encoder, 0x02: Decoder, 0x03: Recorder, 0x04: IOBox, 0x05: NetworkSpeaker, 0x06 NetworkMic, 0x07: LEDBox)
nVersion	version : This parameter can be a combination of the following values. • 0x01: Can’t change the HTTPS port in web page • 0x02: Can change the HTTPS port in web page • 0x04: Support New Model Name variable • 0x08: SupportPasswordVerification using digest
nHttpMode	HTTP Mode of camera (0x00: HTTP, 0x01: HTTPS)
nHttpsPort	HTTPS port for web-connection
nSupportedProtocol	Supported Protocol of camera (0x01: SVNP, 0x02: SUNAPI1.0, 0x04: SUNAPI2.0, 0x08 SUNAPI2.3.1above,0x10:SVP)

Field Name	Value
nPasswordStatus	Password status of camera (0x00: The camera has a password, 0x01: The camera hasn't a password)

- Changed values are filled.

3.11. Set Password in factory default state(SUNAPI)

3.11.1. Request

Field Name	Value
nMode	9(DEF_REQ_APPLY_PASSWORD)
chPacketID	Unique value of client
chMAC	MAC address of Camera
Reserved	
Payload size	Size of payload
Payload	RSA encrypted password

- Using this command, the client can configure the initial password of the device. Please refer to Annex A for the structure used.

3.11.2. Response

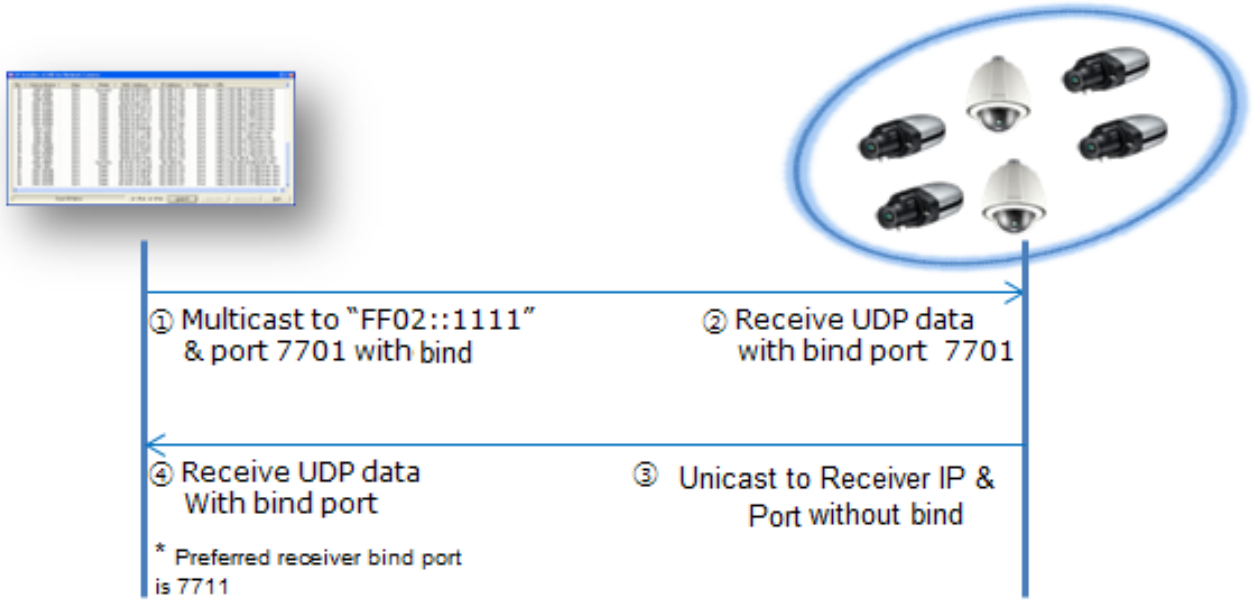
Field Name	Value
nMode	25(DEF_RES_APPLY_PASSWORD)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chPassword	Unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection

Field Name	Value
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort/SpeakerType	Unused / When Modeltype is set to 0x05 network speaker, SpeakerType can be 0: Slave, 1: Master, 2:Server, 3: Module
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused
chAlias	Alias of NVR or Encoder (In case of NW camera, unused)
chNewModelName	New model Name(since April 2016)
nModelType	Device Type (0x00: Camera, 0x01: Encoder, 0x02: Decoder, 0x03: Recorder, 0x04: IOBox, x05: NetworkSpeaker, 0x06 NetworkMic, 0x07: LEDBox)
nVersion	version : This parameter can be a combination of the following values. • 0x01: Can’t change the HTTPS port in web page • 0x02: Can change the HTTPS port in web page • 0x04: Support New Model Name variable • 0x08: SupportPasswordVerification using digest
nHttpMode	HTTP Mode of camera (0x00: HTTP, 0x01: HTTPS)
nHttpsPort	HTTPS port for web-connection
nSupportedProtocol	Supported Protocol of camera (0x01: SVNP, 0x02: SUNAPI1.0, 0x04: SUNAPI2.0, 0x08 SUNAPI2.3.1above,0x10:SVP)
nPasswordStatus	Password status of camera (0x00: The camera has a password, 0x01: The camera hasn’t a password)

- Changed values are filled.

Chapter 4. IPv6

4.1. Communicate Port



4.2. SendData Format(DEPRECATED)

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chIP	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway
chIpv6addr	char[40]	40	IPv6 address
chPassword	char[20]	20	Password
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value

Name	Type	Byte size	Description
chDeviceName	char[10]	10	Model name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V6_OLD
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chIPv6addr[40];
    char chPassword[20];
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
```

```
} DATAPACKET_V6_OLD;
```

4.3. RecvData Format(DEPRECATED)

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chIP	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway
chIpv6addr	char[40]	40	IPv6 address
chPassword	char[20]	20	Password
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address

Name	Type	Byte size	Description
chAlias	char[64]	64	Alias name for NVR or Encoder only • Network camera does not include this field in response packet.

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V6_OLD
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chIPv6addr[40];
    char chPassword[20];
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
    char chAlias[64];
} DATAPACKET_V6_OLD;
```

<Structure Field description>

Table 3. nMode

Mode Definition	VALUE
DEF_REQ_SCAN	1
DEF_REQ_APPLY	2

Mode Definition	VALUE
DEF_REQ_PORTMAPPING	4
DEF_RES_SCAN	11
DEF_RES_APPLY	22
DEF_RES_PASSWORD_ERR	33
DEF_RES_PORT_MAPPING	44
DEF_RES_PORT_MAPPING_ERR	55
DEF_RES_ROUTER_CONN_ERR	66
DEF_RES_APPLY_ERR	77

chPacketID

This value is used to identify Client. Hanwha use this value as unique ID derived from MAC address of PC and random value.

4.4. IP Scan(DEPRECATED)

4.4.1. Request

Field Name	Value
nMode	1 (DEF_REQ_SCAN)
chPacketID	Unique value of client
chMAC	Unused
chIP	Unused
chSubnetMask	Unused
chGateway	Unused
chIpv6addr	IPv6 address
chPassword	Unused
nPort	Unused
nStatus	Unused
chDeviceName	Unused
nHttpPort	Unused
nDevicePort	Unused
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused

Field Name	Value
nNetworkMode	Unused
chDDNS	Unused

4.4.2. Response

Field Name	Value
nMode	11 (DEF_RES_SCAN)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chIpv6addr	IPv6 address
chPassword	Unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	DDNS URL[1]
chAlias	Alias[2] of NVR or Encoder (In case of NW camera, unused)

- **DDNS URL**[1] has the DDNS URL. If camera registered successfully the URL to DDNS server, this field is filled by the registered URL. If camera fails to register, this field is filled by the URL that is made based on the IP. (ex : http://192.168.1.200:8080)
- **Alias**[2] is used to distinguish each device of NVR/Encoder. NVR/Encoder But Network camera does not include ‘chAlias' field in response packet.
- **nDevicePort** : Port number to connect using SVNP, VNP or SSNP protocol.

- **nTcpPort** : Port number to get stream via tcp. This port is valid only if Client uses VNP.
- **nUDPPort** : Port number to get stream via udp. This port is valid only if Client uses VNP.
- **nUploadPort** : Port number to upload camera's f/w (TCP). This port is valid only if Client uses VNP.
- **nMulticastPort** : Port number to get stream via multicast. This port is valid only if Client uses VNP.

4.5. IP Setting(DEPRECATED)

4.5.1. Request

Field Name	Value
nMode	2 (DEF_REQ_APPLY)
chPacketID	Unique value of client
chMAC	MAC address of Camera
chIP	IP address of Camera. Valid only if (nNetworkMode == 0)
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chIpv6addr	IPv6 address
chPassword	Password for connecting to camera with admin privilege
nPort	The same value of nHttpPort
nStatus	Unused
chDeviceName	Unused
nHttpPort	Port number to change
nDevicePort	Port number to connect using SVNP
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused

- Using this command, client can change the IP type, address and port number.

4.5.2. Response

Field Name	Value
nMode	22 (DEF_RES_APPLY)

Field Name	Value
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chIpv6addr	IPv6 address
chPassword	Unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	DDNS URL*
chAlias	Alias of NVR or Encoder (In case of NW camera, unused)

- Changed values are filled.

4.6. SendData Format for SUNAPI

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chIP	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask
chGateway	char[16]	16	Gateway

Name	Type	Byte size	Description
chIpv6addr	char[40]	40	IPv6 address
chPassword	char[20]	20	Password
is_only_support_sunapi	char	1	If only sunapi is supported
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, only up to 10 in the whole character)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address
chAlias	char[64]	64	Alias name for NVR or Encoder only • Network camera does not include this field in response packet.
nModelType	char	1	Model Type
nVersion	Unsigned short	2	Sturcture version
nHttpMode	Char	1	Http Mode
nHttpsPort	Unsigned short	2	HTTPS Port
nSupportedProtocol	Char	1	Supported Protocol
nPasswordStatus	Char	1	Password Status

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V6_OLD
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chIPv6addr[40];
    char chPassword[20];
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
    char chAlias[64];
    char nModelType;
    unsigned short nVersion;
    char nHttpMode;
    unsigned short nHttpsPort;
    char nSupportedProtocol;
    char nPasswordStatus;
} DATAPACKET_V6_EXT;
```

4.7. RecvData Format for SUNAPI

Name	Type	Byte size	Description
nMode	Unsigned char	1	Mode, refer to below table 1
chPacketID	unsigned char[18]	18	Packet value
chMAC	char[18]	18	Camera MAC address
chIP	char[16]	16	IPv4 address
chSubnetMask	char[16]	16	Subnet mask

Name	Type	Byte size	Description
chGateway	char[16]	16	Gateway
chIpv6addr	char[40]	40	IPv6 address
chPassword	char[20]	20	Password
Is_only_sunapi	char	1	If only SUNAPI is supported
nPort	unsigned short	2	Port
nStatus	unsigned char	1	Port mapping return value
chDeviceName	char[10]	10	Model name (In new model name case, only up to 10 in the whole character)
nHttpPort	unsigned short	2	HTTP port
nDevicePort	unsigned short	2	Device port
nTcpPort	unsigned short	2	TCP port
nUdpPort	unsigned short	2	UDP port
nUploadPort	unsigned short	2	Upload port
nMulticastPort	unsigned short	2	Multicast port
nNetworkMode	unsigned char	1	Mode status (Static or DHCP)
chDDNS	char[128]	128	DDNS address
chAlias	char[32]	32	Alias name for NVR or Encoder only • Network camera does not include this field in response packet.
chNewModelName	Char[32]	32	New Model Name(since April 2016)
nModelType	char	1	Model Type
nVersion	Unsigned short	2	Sturcture version
nHttpMode	Char	1	Http Mode
nHttpsPort	Unsigned short	2	HTTPS Port
nSupportedProtocol	Char	1	Supported Protocol
nPasswordStatus	Char	1	Password Status

(On windows app) Struct define in C++ (Not packed)

```
typedef struct tagDataPaket_V6_OLD
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
    char chIPv6addr[40];
    char chPassword[20];
    char is_only_support_sunapi;
    unsigned short nPort;
    unsigned char nStatus;
    char chDeviceName[10];
    unsigned short nHttpPort;
    unsigned short nDevicePort;
    unsigned short nTcpPort;
    unsigned short nUdpPort;
    unsigned short nUploadPort;
    unsigned short nMulticastPort;
    unsigned char nNetworkMode;
    char chDDNS[128];
    char chAlias[64];
    char nModelType;
    unsigned short nVersion;
    char nHttpMode;
    unsigned short nHttpsPort;
    char nSupportedProtocol;
    char nPasswordStatus;
} DATAPACKET_V6_EXT;

typedef struct tagRsaScanResponse
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char chIP[16];
    char chSubnetMask[16];
    char chGateway[16];
```

```
    char MaxPasswordLen;
    char Reserved[10];
    char PayloadSize[2];
    char Payload[payloadSize];
} DATAPACKET_RSA_RESPONSE;

typedef struct tagApplyPasswordRequest
{
    unsigned char nMode;
    unsigned char chPacketID[18];
    char chMAC[18];
    char Reserved[10];
    char PayloadSize[2];
    char Payload[payloadSize];
}DATAPACKET_APPLY_PASSWORD_REQUEST;
```

<Structure Field description>

Table 4. nMode

Mode Definition	VALUE
DEF_REQ_SCAN_EXT	6
DEF_REQ_APPLY_EXT	7
DEF_REQ_SCAN_RSA	8
DEF_REQ_APPLY_PASSWORD	9
DEF_RES_SCAN_EXT	12
DEF_RES_SCAN_RSA	13
DEF_RES_APPLY_EXT	23
DEF_RES_APPLY_PASSWORD_ERR	24
DEF_RES_APPLY_PASSWORD	25
DEF_RES_PASSWORD_ERR	33
DEF_RES_ROUTER_CONN_ERR	66
DEF_RES_APPLY_ERR	77

chPacketID

This value is used to identify Client. Hanwha use this value as unique ID derived from MAC address of PC and random value.

4.8. IP Scan for SUNAPI

4.8.1. Request

Field Name	Value
nMode	6(DEF_REQ_SCAN_EX)
chPacketID	Unique value of client
chMAC	Unused
chIP	Unused
chSubnetMask	Unused
chGateway	Unused
chIpv6addr	IPv6 address
chPassword	Unused
Is_only_sunapi	Unused
nPort	Unused
nStatus	Unused
chDeviceName	Unused
nHttpPort	Unused
nDevicePort	Unused
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Unused
chDDNS	Unused
chAlias	Unused
nModelType	Unused
nVersion	Unused
nHttpMode	Unused
nHttpsPort	Unused
nSupportedProtocol	Unused
nPasswordStatus	Unused

4.8.2. Response

Field Name	Value
nMode	12 (DEF_RES_SCAN_EX)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chIpv6addr	IPv6 address
Nonce	When nVersion 0x08 is supported will have the nonce value, that can be used for password digest.
isOnlySunapi	This field is 1 in case if the device only supports SUNAPI and not SVNP
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort / SpeakerType	Unused / When Modeltype is set to 0x05 network speaker, SpeakerType can be 0: Slave, 1: Master, 2:Server, 3: Module
MaxChannel	When nVersion 0x08 is supported will have max channel number **Note** For models that support virtual channel addition/deletion, it represents the current channel count.
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	DDNS URL[1]
chAlias	Alias[2] of NVR or Encoder (In case of NW camera, unused)
chNewModelName	New model name (since April 2016)

Field Name	Value
nModelType	Device Type (0x00: Camera, 0x01: Encoder, 0x02: Decoder, 0x03: Recorder, 0x04: IOBox, 0x05: NetworkSpeaker, 0x06 NetworkMic, 0x07: LEDBox)
nVersion	version : This parameter can be a combination of the following values. • 0x01: Can't change the HTTPS port in web page • 0x02: Can change the HTTPS port in web page • 0x04: Support New Model Name variable • 0x08: SupportPasswordVerification using digest
nHttpMode	HTTP Mode of camera (0x00: HTTP, 0x01: HTTPS)
nHttpsPort	HTTPS port for web-connection
nSupportedProtocol	Supported Protocol of camera (0x01: SVNP, 0x02: SUNAPI1.0, 0x04: SUNAPI2.0, 0x08 SUNAPI2.3.1above,0x10:SVP)
nPasswordStatus	Password status of camera (if Version 0x08 need to notify the actual state of the device if password is initialized or not) (0x00: The camera has a password, 0x01: The camera hasn't a password)

- **DDNS URL**[1] has the DDNS URL. If camera registered successfully the URL to DDNS server, this field is filled by the registered URL. If camera fails to register, this field is filled by the URL that is made based on the IP. (ex : http://192.168.1.200:8080)
- **Alias**[2] is used to distinguish each device of NVR/Encoder. NVR/Encoder But Network camera does not include 'chAlias' field in response packet.
- **nDevicePort** : Port number to connect using SVNP, VNP or SSNP protocol.
- **nTcpPort** : Port number to get stream via tcp. This port is valid only if Client uses VNP.
- **nUDPPort** : Port number to get stream via udp. This port is valid only if Client uses VNP.
- **nUploadPort** : Port number to upload camera's f/w (TCP). This port is valid only if Client uses VNP.
- **nMulticastPort** : Port number to get stream via multicast. This port is valid only if Client uses VNP.

4.9. SCAN Uninitialized Devices for RSA Key

4.9.1. Request

Field Name	Value
nMode	8(DEF_REQ_SCAN_RSA)

Field Name	Value
chPacketID	Unique value of client
chMAC	Unused
chIP	Unused
chSubnetMask	Unused
chGateway	Unused
chPassword	Unused
nPort	Unused
nStatus/nVersion	Unused
chDeviceName	Unused
nHttpPort	Unused
nDevicePort	Unused
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Unused
chDDNS	Unused
chAlias	Unused
nModelType	Unused
nVersion	Unused
nHttpMode	Unused
nHttpsPort	Unused
nSupportedProtocol	Unused
nPasswordStatus	Unused

4.9.2. Response

Field Name	Value
nMode	13(DEF_RES_SCAN_RSA)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera

Field Name	Value
maxPasswordLen	Maximum password length supported. If this field is 0, the maximum password length supported is 15; if it is larger than 0, the value supplied is the maximum password length.
Reserved	
PacketSize	Size of Payload
RSAKeyPayload	RSA Public Key

4.10. IP Setting for SUNAPI

4.10.1. Request

Field Name	Value
nMode	7 (DEF_REQ_APPLY_EX)
chPacketID	Unique value of client
chMAC	MAC address of Camera
chIP	IP address of Camera. Valid only if (nNetworkMode == 0)
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chIpv6addr	IPv6 address
chPassword	Password for connecting to camera with admin privilege (When version 0x08 is supported, pass sha256(username:password:nonce) truncated to first 20 bytes of total 32 bytes)
Is_only_sunapi	unused
nPort	The same value of nHttpPort
nStatus	Unused
chDeviceName	Unused
nHttpPort	Port number to change
nDevicePort	Port number to connect using SVNP
nTcpPort	Unused
nUdpPort	Unused
nUploadPort	Unused
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)

Field Name	Value
chDDNS	Unused
UserName	Pass the username in this field (When version 0x08)
nModelType	Unused
nVersion	Unused (When digest password is used should be set to 0x08)
nHttpMode	Unused
nHttpsPort	Port number to change (if version value is 2)
nSupportedProtocol	Unused
nPasswordStatus	Unused

- Using this command, client can change the IP type, address and port number.

4.10.2. Response

Field Name	Value
nMode	23 (DEF_RES_APPLY_EX)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chIpv6addr	IPv6 address
chPassword	Unused
Is_only_sunapi	unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused

Field Name	Value
nUploadPortt/SpeakerType	Unused / When Modeltype is set to 0x05 network speaker, this SpeakerType can be 0: Slave, 1: Master, 2:Server, 3: Module
nMulticastPor	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	DDNS URL*
chAlias	Alias of NVR or Encoder (In case of NW camera, unused)
chNewModelName	New Model Name(since April 2016)
nModelType	Device Type (0x00: Camera, 0x01: Encoder, 0x02: Decoder, 0x03: Recorder, 0x04: IOBox, 0x05: NetworkSpeaker, 0x06 NetworkMic, 0x07: LEDBox)
nVersion	version : This parameter can be a combination of the following values. • 0x01: Can't change the HTTPS port in web page • 0x02: Can change the HTTPS port in web page • 0x04: Support New Model Name variable • 0x08: SupportPasswordVerification using digest
nHttpMode	HTTP Mode of camera (0x00: HTTP, 0x01: HTTPS)
nHttpsPort	HTTPS port for web-connection
nSupportedProtocol	Supported Protocol of camera (0x01: SVNP, 0x02: SUNAPI1.0, 0x04: SUNAPI2.0, 0x08 SUNAPI2.3.1above,0x10:SVP)
nPasswordStatus	Password status of camera (0x00: The camera has a password, 0x01: The camera hasn't a password)

- Changed values are filled.

4.11. Set password in factory default state (SUNAPI).

4.11.1. Request

Field Name	Value
nMode	9(DEF_REQ_APPLY_PASSWORD)
chPacketID	Unique value of client

Field Name	Value
chMAC	MAC address of Camera
Reserved	
Payload size	Size of payload
Payload	RSA encrypted password

- Using this command, the client can configure the initial password of the device. Please refer to Annex A for the structure used.

4.11.2. Response

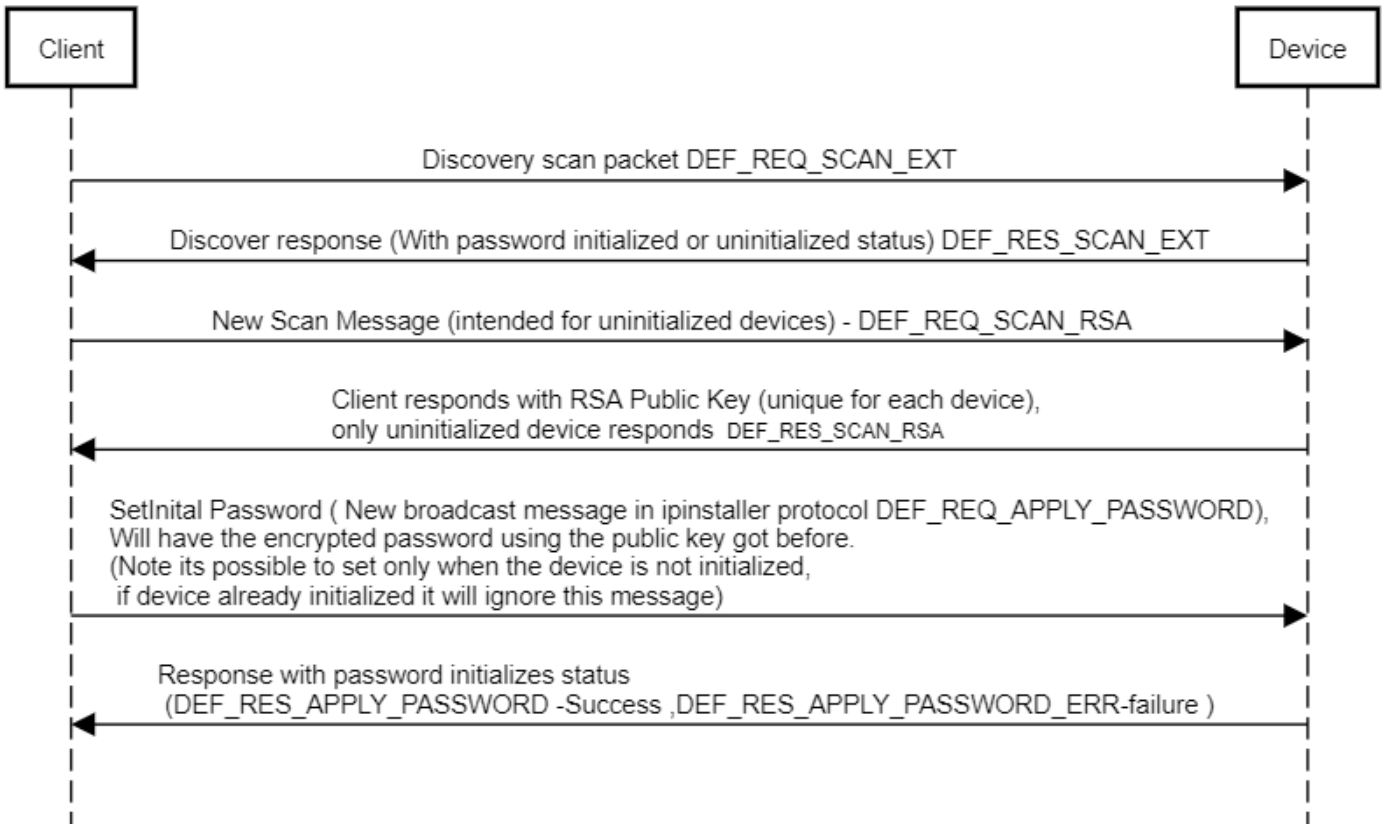
Field Name	Value
nMode	25(DEF_RES_APPLY_PASSWORD)
chPacketID	Unique value of camera
chMAC	MAC address of Camera
chIP	IP address of Camera
chSubnetMask	Subnet Mask of Camera
chGateway	Gateway of Camera
chPassword	Unused
nPort	HTTP port for web-connection
nStatus	Success or not (of UPNP port mapping)
chDeviceName	Model Name (In new model name case, it only is 10 character in the whole model name)
nHttpPort	HTTP port of web-connection
nDevicePort	Port number to connect using protocol document
nTcpPort	Unused
nUdpPort	Unused
nUploadPort/SpeakerType	Unused / When Modeltype is set to 0x05 network speaker, SpeakerType can be 0: Slave, 1: Master, 2:Server, 3: Module
nMulticastPort	Unused
nNetworkMode	Network IP type (0: Static , 1: DHCP, 2: PPPoE)
chDDNS	Unused
chAlias	Alias of NVR or Encoder (In case of NW camera, unused)
chNewModelName	New model Name(since April 2016)

Field Name	Value
nModelType	Device Type (0x00: Camera, 0x01: Encoder, 0x02: Decoder, 0x03: Recorder, 0x04: IOBox,0x05: NetworkSpeaker, 0x06 NetworkMic, 0x07: LEDBox)
nVersion	version : This parameter can be a combination of the following values. • 0x01: Can't change the HTTPS port in web page • 0x02: Can change the HTTPS port in web page • 0x04: Support New Model Name variable • 0x08: SupportPasswordVerification using digest
nHttpMode	HTTP Mode of camera (0x00: HTTP, 0x01: HTTPS)
nHttpsPort	HTTPS port for web-connection
nSupportedProtocol	Supported Protocol of camera (0x01: SVNP, 0x02: SUNAPI1.0, 0x04: SUNAPI2.0, 0x08 SUNAPI2.3.1above,0x10:SVP)
nPasswordStatus	Password status of camera (0x00: The camera has a password, 0x01: The camera hasn't a password)

- Changed values are filled.

PAssword setting Sequence Diagram

IP installer password setting (RSA)



Chapter 5. Annex A

5.1. Updated Structure Used

```
static const int MAX_STRLEN_MAC = 18;
static const int MAX_STRLEN_IP = 16;
static const int MAX_STRLEN_IPv6 = 40;
static const int MAX_STRLEN_DEVICE_NAME = 10;
static const int MAX_STRLEN_DEVICE_NAME_NEW = 32;
static const int MAX_STRLEN_PASSWORD = 20;
static const int MAX_STRLEN_DDNS_URL = 128;
static const int MAX_STRLEN_ALIAS = 32;
static const int PACKETID_LEN = 18;
static const int MAX_STRLEN_RSA_PAYLOAD_SIZE = 512;

typedef struct _IPSET_T
{
    char ip_addr[MAX_STRLEN_IP];
    char subnetmask[MAX_STRLEN_IP];
    char gateway[MAX_STRLEN_IP];
} __attribute__ ((__packed__)) _ipset_t;

typedef struct _PORTSET_T
{
    unsigned short http_port;
    unsigned short device_port;
    unsigned short tcp_port;
    unsigned short udp_port;
    unsigned short upload_port;
    unsigned short multicast_port;
} __attribute__ ((__packed__)) _portset_t;

typedef struct DATAPACKET_IPv4_T
{
    unsigned char mode;
    unsigned char packet_id[PACKETID_LEN];
    char mac_addr[MAX_STRLEN_MAC];
    _ipset_t ipset;
    char password[MAX_STRLEN_PASSWORD];
    // char reserved1;
    char is_only_support_sunapi;
```

```
    unsigned short port;
    unsigned char status;
    char device_name[MAX_STRLEN_DEVICE_NAME];
    char reserved2;
    _portset_t portset;
    unsigned char network_mode;
    char ddns_url[MAX_STRLEN_DDNS_URL];
    char reserved3;
} __attribute__ ((__packed__)) datapacket_v4_t;

typedef struct DATAPACKET_IPv6_T
{
    unsigned char mode;
    unsigned char packet_id[PACKETID_LEN];
    char mac_addr[MAX_STRLEN_MAC];
    _ipset_t ipset;
    char ipv6_addr[MAX_STRLEN_IPv6];
    char password[MAX_STRLEN_PASSWORD];
    // char reserved1;
    char is_only_support_sunapi;
    unsigned short port;
    unsigned char status;
    char device_name[MAX_STRLEN_DEVICE_NAME];
    char reserved2;
    _portset_t portset;
    unsigned char network_mode;
    char ddns_url[MAX_STRLEN_DDNS_URL];
    char reserved3;
} __attribute__ ((__packed__)) datapacket_v6_t;

typedef struct DATAPACKET_EXT_IPv4_T
{
    unsigned char mode;
    unsigned char packet_id[PACKETID_LEN];
    char mac_addr[MAX_STRLEN_MAC];
    _ipset_t ipset;
    char password[MAX_STRLEN_PASSWORD];
    // char reserved1;
    char is_only_support_sunapi;
    unsigned short port;
    unsigned char status;
```

```
    char device_name[MAX_STRLEN_DEVICE_NAME];
    char reserved2;
    _portset_t portset;
    unsigned char network_mode;
    char ddns_url[MAX_STRLEN_DDNS_URL];
    char alias[MAX_STRLEN_ALIAS];
    char device_name_new[MAX_STRLEN_DEVICE_NAME_NEW]; // New Model Name (ex:
PNR-6320RH-001)
    unsigned char model_type;
    // version : This parameter can be a combination of the following values
    // 0x01: Can't change the HTTPS port in web page
    // 0x02: Can change the HTTPS port in web page
    // 0x04: Support New Model Name variable
    unsigned short version;
    unsigned char https_mode;
    unsigned char reserved3;
    unsigned short https_port;
    unsigned char supported_protocol;
    unsigned char no_password;
} __attribute__ ((__packed__)) datapacket_ext_v4_t;

typedef struct DATAPACKET_EXT_IPv6_T
{
    unsigned char mode;
    unsigned char packet_id[PACKETID_LEN];
    char mac_addr[MAX_STRLEN_MAC];
    _ipset_t ipset;
    char ipv6_addr[MAX_STRLEN_IPv6];
    char password[MAX_STRLEN_PASSWORD];
    // char reserved1;
    char is_only_support_sunapi;
    unsigned short port;
    unsigned char status;
    char device_name[MAX_STRLEN_DEVICE_NAME];
    char reserved2;
    _portset_t portset;
    unsigned char network_mode;
    char ddns_url[MAX_STRLEN_DDNS_URL];
    char alias[MAX_STRLEN_ALIAS];
    char device_name_new[MAX_STRLEN_DEVICE_NAME_NEW];
    unsigned char model_type;
```

```
    // version : This parameter can be a combination of the following values
    // 0x01: Can't change the HTTPS port in web page
    // 0x02: Can change the HTTPS port in web page
    // 0x04: Support New Model Name variable
    unsigned short version;
    unsigned char https_mode;
    unsigned char reserved3;
    unsigned short https_port;
    unsigned char supported_protocol;
    unsigned char no_password;
} __attribute__ ((__packed__)) datapacket_ext_v6_t;

typedef struct tagRsaScanResponse
{
    unsigned char mode;
    unsigned char packet_id[PACKETID_LEN];
    char mac_addr[MAX_STRLEN_MAC];
    _ipset_t ipset;
    char MaxPasswordLen;
    char reserved[10];
    unsigned short payloadSize;
    char payload[MAX_STRLEN_RSA_PAYLOAD_SIZE];
}__attribute__ ((__packed__)) datapacket_rsa_response_t;

typedef struct tagApplyPasswordRequest
{
    unsigned char mode;
    unsigned char packet_id[PACKETID_LEN];
    char mac_addr[MAX_STRLEN_MAC];
    char reserved[10];
    char reserved2;
    unsigned short payloadSize;
    char payload[MAX_STRLEN_RSA_PAYLOAD_SIZE];
}__attribute__ ((__packed__)) datapacket_apply_password_request_t;
```

Chapter 6. Revision History

Version	Description	Release Date
1.23.0	Edited according to the standard document format	30th, JULY, 2010
1.23.1	Edited the data structure according to source code	11th AUG, 2010
1.23.2	Added unpack structure for windows client & changed some sentences	20th FEB, 2012
1.23.3	Added new scan and apply command with extended structure for the SUNAPI	24th MAY, 2013
1.23.4	Support new model name (only SUNAPI)	29TH MAR,2016
1.24.0	Factory default state handling and password digest verification (nversion 0x08)	28th MAR, 2019
1.25.0	Password Initialization method added	27th JUNE 2019
1.26.0	Device Type 0x04: IOBox, 0x05: NetworkSpeaker/Mic added newly	19th AUG 2020
1.27.0	nSupportedProtocol list updated based on actual implementation	20th OCT 2020
1.28.0	Device Type modified ***0x05: NetworkSpeaker, 0x06 NetworkMic and SpeakerType added 0:Slave, 1:Master, 2:Server, 3:Module MaxChannel: definition updated***	***14th SEP 2021***
2.6.0	Added to SUNAPI Documentation Release	1st MAR 2022
2.6.1	RSA Key Response updated to include maximum password length supported Device Type 0x07: LEDBox newly added	1st AUG 2022