HIRSCHMANN
A BELDEN BRAND

User Manual

Basic Configuration
HiProvision Add-on: CAR IP

The naming of copyrighted trademarks in this manual, even when not specially indicated, should not be taken to mean that these names may be considered as free in the sense of the trademark and tradename protection law and hence that they may be freely used by anyone.

The performance features described here are binding only if they have been expressly agreed when the contract was made. This document was produced by Hirschmann Automation and Control GmbH according to the best of the company's knowledge. Hirschmann reserves the right to change the contents of this document without prior notice. Hirschmann can give no guarantee in respect of the correctness or accuracy of the information in this document.

Hirschmann can accept no responsibility for damages, resulting from the use of the network components or the associated operating software. In addition, we refer to the conditions of use specified in the license contract.

You can get the latest version of this manual on the Internet at the Hirschmann product site (www.hirschmann.com).

Hirschmann Automation and Control GmbH
Stuttgarter Str. 45-51
72654 Neckartenzlingen
Germany

Contents

List of figures

List of Tables

1. INTRODUCTION

1.1 Description

This document is valid as of Dragon PTN Release 4.3DR. The CAR IP (=Central Alarm Reporter IP) add-on is an alarm interface between HiProvision and a CAS (=Central Alarm System) or umbrella management system both connected through an Ethernet link (UDP). The behavior and the Ethernet communication (UDP) can be customized through various settings in the '<HiProvision path>\bin\Carip.cfg' configuration file.

Apart from the failure alarm and failure repair messages (i.e. the alarm has been cleared and acknowledged) also handshake messages are exchanged between the CAR IP and the CAS (and vice versa).

The following chapters will explain more about:

- The installation/activation of the add-on;
- The format of the different messages;
- The communication processes between the CAS and the add-on;
- The configuration of the CAR IP;
- Some example messages;

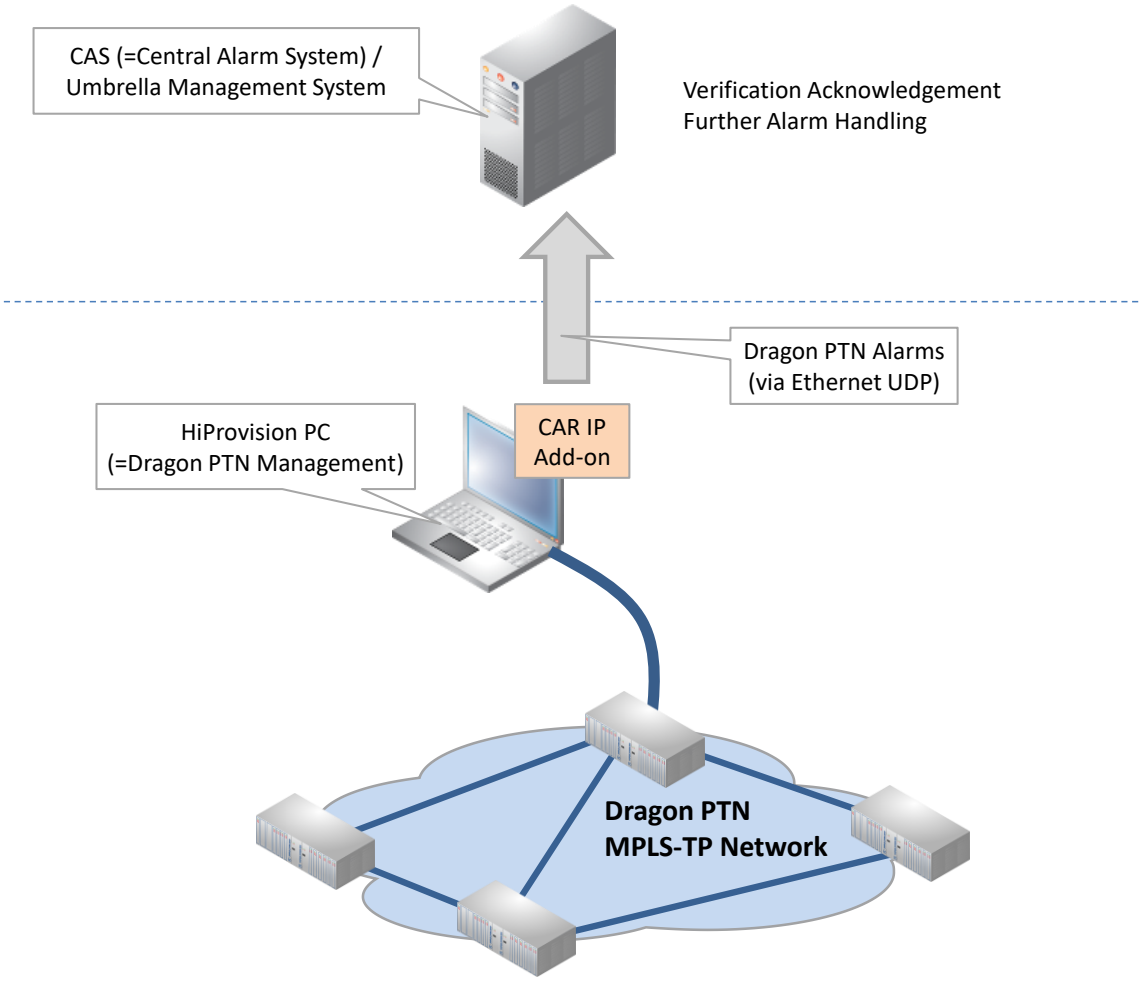

Figure 1 CAR IP Example

1.2 Prerequisite

A 'CAR IP Add-on' voucher or license must be purchased first. Once it has been purchased, install a license pack with this voucher on the HiProvision PC. See chapter 'SERIAL KEY / VOUCHERS / LICENSE PACK' in manual Ref.[2Mgt] in Table 1 to generate and install the license pack. It can be verified in HiProvision whether a CAR IP license has been installed via Dashboard → Licenses: Voucher type = CAR IP. 'Vouchers available' must be at least one.

1.3 Installation/Activation

1. The CAR IP add-on is by default available in HiProvision and can be found via Dashboard → (Tools) Add-ons;
2. Click the CAR IP tab;

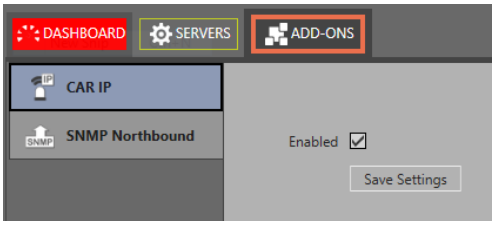

Figure 2 CAR IP Tab

3. Check the Enabled checkbox and click the Save Settings button;
4. When the HiProvision servers were already started, a pop-up requests to restart the servers. If the servers were not started yet, just start the servers. (Re)start the HiProvision servers via Dashboard → Servers → Play button;
5. The (re)start of the servers will check the availability of a valid CAR IP license and start up the CAR IP ('CI') process if the license is valid. The servers will not start without a valid license.
6. If the 'CI' process is in the Started state, the CAR IP add is up and running. It acts according the configured settings in the '<HiProvision path>\bin\Carip.cfg' configuration file, more info in §4.

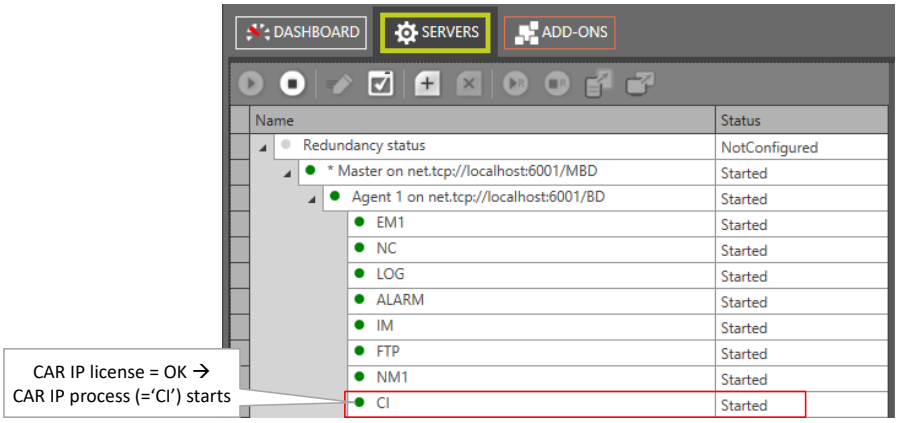

Figure 3 CAR IP (=CI) Process in Servers Tab

1.4 Supported Hardware, Firmware, Software

The supported hardware, firmware and software within this Dragon PTN release can be found on the Portal https://hiprovision.hirschmann.com via Shortcuts → Downloads.

1.5 Manual References

Table 1 is an overview of the manuals referred to in this manual. '&' refers to the language code, '*' refers to the manual issue. All these manuals can be found in the HiProvision Help function as well.

Table 1 Manual References

Ref.	Number	Title
[1]	DRA-DRM801-&-*	Dragon PTN Installation and Operation
[2Mgt]	DRA-DRM830-&-*	HiProvision Management Operation
[3]	DRA-DRM822-&-*	HiProvision Alarms List

2. MESSAGE TYPES AND LAYOUT

2.1 Hexadecimal Notation

Some values in the tables below are hexadecimal (0xnn), e.g. 0x0C = decimal 12.

2.2 Failure Alarm Message

Each alarm has e.g. a code, which uniquely identifies the alarm, a certain time on which the alarm was generated and an URL. This URL is the source of the network element that generated the alarm.

When certain errors occur in the Dragon PTN network, corresponding alarms will be raised in HiProvision. If the unique alarm code of the raised alarm is configured in the Carip.cfg file, the CAR IP add-on will send a failure alarm message via UDP to the CAS. The failure alarm message can have 4 different formats as listed below. The used format (or ProtocolVersion) is configured in the Carip.cfg file:

- ProtocolVersion=1 (13 bytes: byte 0-12). Message Format → Table 2;
- ProtocolVersion=2 (14 bytes: byte 0-13). Message Format → Table 2;
- ProtocolVersion=3 (20 bytes: byte 0-19). Message Format → Table 3;
- ProtocolVersion=4 (24 bytes: byte 0-23). Message Format → Table 4;

Table 2 Failure Alarm Message Format: ProtocolVersion 1, 2

ProtocolVersion = 1		ProtocolVersion = 2		Description
Byte	**Field**	**Byte**	**Field**	
0	Subsystem code	0	Subsystem code	For Dragon PTN, default value = 1, value can be configured in the 'Carip.cfg' file.
1	Station number	1	Station number	Each Dragon PTN station can be given a number in the configuration file (e.g. it is useful to group all the nodes of 1 location to the same station number)
2	Node number	2	Node number	The number of the node that reports the error
3	Slot number	3	Slot number	The number of the interface slot that reports the alarm (translation can be found in the configuration file)
4	Failure code (higher digits)	4	Failure code (higher digits)	Highest byte of the configured failure code, see Note (*) further on
5	Failure code (lower digits)	5	Failure code (lower digits)	Lowest byte of the configured failure code, see Note (*) further on
(not available)		6	Error recovery type	Always 0xFF = Fault Situation
6	Year (higher digits)	7	Year (higher digits)	e.g. 20
7	Year (lower digits)	8	Year (lower digits)	e.g. 06
8	Month	9	Month	0x01 – 0x0C
9	Day	10	Day	0x01 – 0x1F
10	Hour	11	Hour	0x00 – 0x17
11	Minutes	12	Minutes	0x00 – 0x3B
12	Seconds	13	Seconds	0x00 – 0x3B

Table 3 Failure Alarm Message Format: ProtocolVersion 3

Byte	Field	Description
0	Subsystem code	For Dragon PTN, default value = 1, value can be configured in the 'Carip.cfg' file.
1	Station number	Each Dragon PTN station can be given a number in the configuration file (e.g. it might come in handy to group all the nodes of 1 location to the same station number)
2	Node number	The number of the node that reports the error
3	Slot number	The number of the interface slot that reports the alarm (translation can be found in the configuration file)
4	Failure code (higher digits)	Highest byte of the configured failure code, see Note (*) further on
5	Failure code (lower digits)	Lowest byte of the configured failure code, see Note (*) further on
6	Year (higher digits)	e.g. 20
7	Year (lower digits)	e.g. 11
8	Month	0x01 – 0x0C
9	Day	0x01 – 0x1F
10	Hour	0x00 – 0x17
11	Minutes	0x00 – 0x3B

Byte	Field	Description
12	Seconds	0x00 – 0x3B
13	Year (higher digits)	e.g. 20
14	Year (lower digits)	e.g. 11
15	Month	0x01 – 0x0C
16	Day	0x01 – 0x1F
17	Hour	0x00 – 0x17
18	Minutes	0x00 – 0x3B
19	Seconds	0x00 – 0x3B

Table 4 Failure Alarm Message Format: ProtocolVersion 4

Byte	Field	Description
0	Start Byte 1	0x10
1	Start Byte 2	0x02
2	Subsystem code	For Dragon PTN, default value = 1, value can be configured in the 'Carip.cfg' file.
3	Type of information	0x01 : Alarm occurrence 0x00 : Alarm recovery
4	Node number (high)	The node number high byte that reports the error
5	Node number (low)	The node number low byte that reports the error
6	Slot number	The number of the interface slot that reports the alarm (translation can be found in the configuration file)
7	Station number	Each Dragon PTN station can be given a number in the configuration file (e.g. it might come in handy to group all the nodes of 1 location to the same station number)
8	Alarm severity	0x00: No alarm, 0x01: Critical, 0x02: Major and 0x03H: Warning and minor
9	Failure code (higher digits)	Highest byte of the configured failure code
10	Failure code (lower digits)	Lowest byte of the configured failure code
11	Alarm sequence number	0x00 – 0xFF, circular numbering of the occurring alarms. The sequence number is resetted every 256 alarms. 0,1,...,255 → 0,1....,255 → 0,…
12	Reserved byte	0x00
13	Reserved byte	0x00
14	Year (higher digits)	e.g. 20
15	Year (lower digits)	e.g. 11
16	Month	0x01 – 0x0C
17	Day	0x01 – 0x1F
18	Hour	0x00 – 0x17
19	Minutes	0x00 – 0x3B
20	Seconds	0x00 – 0x3B
21	Checksum	Checksum is calculated on byte0-20
22	End byte 1	0x10
23	End byte 2	0x03

NOTE: (*) When an alarm is raised with Protocol version 1, 2, or 3, the first bit in the higher digit part of the failure code is always 1;

NOTE: (**) Which failure code (e.g. 100) corresponds to which alarm code (e.g. 5.5), can be configured by the user in the 'Carip.cfg' file, see §4. All HiProvision alarm codes can be found in Ref.[3] in Table 1.

2.3 Failure Repair Message

When an alarm in HiProvision is acknowledged and cleared, a failure repair message will be sent out. This message has the same layout as the failure alarm message. The only difference is the indication of the failure code: the first bit in the higher digit part will be 0.

See chapter 'Alarm Handling' in manual Ref.[2Mgt] in Table 1 for more information on clearing and acknowledging alarms.

2.4 Handshake Message from Add-on to CAS

The add-on can send out a handshake message to the CAS. The frequency of this message can be set by the user. The tables below show the layouts of these handshake messages for protocol 1, 2, 3 (=Table 5) and 4 (=Table 6).

Table 5 Add-on to CAS Handshake Message Format: ProtocolVersion 1, 2, 3

Byte no.	Item	Value
0	Start byte of the handshake protocol	FF
1	Subsystem code	For Dragon PTN, default value = 1, value can be configured in the 'Carip.cfg' file.
2	Higher digits of the year	e.g. 20
3	Lower digits of the year	e.g. 06
4	Month	0x01 – 0x0C
5	Day	0x01 – 0x1F
6	Hour	0x00 – 0x17
7	Minutes	0x00 – 0x3B
8	Seconds	0x00 – 0x3B

Table 6 Add-on to CAS Handshake Message Format: ProtocolVersion 4

Byte no.	Item	Value
0	Start Byte 1	0x10
1	Start Byte 2	0x02
2	Subsystem code	For Dragon PTN, value = 1
3	Type of information	0x04
4	Node number(high)	0x00
5	Node number(low)	0x00

Byte no.	Item	Value
6	Slot number	0x00
7	Station number	0x00
8	Alarm severity	0x00
9	Alarm (higher digits)	0x00
10	Alarm (lower digits)	0x00
11	Alarm sequence number	0x00
12	Reserved byte	0x00
13	Reserved byte	0x00
14	Year (higher digits)	e.g. 20
15	Year (lower digits)	e.g. 11
16	Month	0x01 – 0x0C
17	Day	0x01 – 0x1F
18	Hour	0x00 – 0x17
19	Minutes	0x00 – 0x3B
20	Seconds	0x00 – 0x3B
21	Checksum	Checksum is calculated on byte0-20
22	End byte 1	0x10
23	End byte 2	0x03

2.5 Handshake Message from CAS to Add-on

The CAS can send out handshake messages to the add-on. The tables below show the layouts of these handshake messages for protocol 1, 2, 3 (=Table 7) and 4 (=Table 8):

Table 7 CAS to Add-on Handshake Message Format: ProtocolVersion 1, 2, 3

Byte no.	Item	Value
0	Start byte of the handshake protocol	AA
1	Subsystem code	For Dragon PTN, default value = 1, value can be configured in the 'Carip.cfg' file.
2	Higher digits of the year	e.g. 20
3	Lower digits of the year	e.g. 06
4	Month	0x01 – 0x0C
5	Day	0x01 – 0x1F
6	Hour	0x00 – 0x17
7	Minutes	0x00 – 0x3B
8	Seconds	0x00 – 0x3B

Table 8 CAS to Add-on Handshake Message Format: ProtocolVersion 4

Byte no.	Item	Value
0	Start Byte 1	0x10
1	Start Byte 2	0x02
2	Subsystem code	For Dragon PTN, value = 1
3	Type of information	0x02
4	Node number (high)	0x00
5	Node number (low)	0x00
6	Slot number	0x00
7	Station number	0x00
8	Alarm severity	0x00
9	Alarm (higher digits)	0x00
10	Alarm (lower digits)	0x00
11	Alarm sequence number	0x00
12	Reserved byte	0x00
13	Reserved byte	0x00
14	Year (higher digits)	e.g. 20
15	Year (lower digits)	e.g. 11
16	Month	0x01 – 0x0C
17	Day	0x01 – 0x1F
18	Hour	0x00 – 0x17
19	Minutes	0x00 – 0x3B
20	Seconds	0x00 – 0x3B
21	Checksum	Checksum is calculated on byte0-20
22	End byte 1	0x10
23	End byte 2	0x03

3. COMMUNICATION PROCESS

This chapter describes the different communication processes between the CAS and the CAR IP add-on.

3.1 Transmission Process of the Failure Alarm / Failure Repair Message

When an error in the Dragon PTN network raises an alarm in HiProvision or when the alarm is being cleared and acknowledged, a failure alarm or a failure repair message respectively will be sent out to the CAS. The CAS should return a feedback message 0xFE when the received message has the correct format and content. If not, the CAS sends out the message 0xFC. This feedback should be received by the add-on within 10 seconds after transmission. If not, the message will be sent out again as many times and with an interval as specified in the Carip.cfg configuration file.

Example of the parameters in the configuration file:

```
[Flow]
// ---------------------------------------------------------------------------------------------------
// if alarm transmission should be acknowledged by peer : delay > 0 msec
// 'delay' is the time between 2 transmission retries and is expressed in milliseconds.
// ---------------------------------------------------------------------------------------------------
max_retries = 5
delay = 10000
```

When the number of retries has expired and still no feedback message is received for this alarm, the next new alarm message in the list will be sent out after 10 seconds with again some retries if needed.

When there is a temporary communication loss between the CAS and the add-on, the alarm messages during that period of time will be stored in a circular buffer (oldest entries will be overwritten when the buffer is full). After restoration of the communication the alarms will be sent out all at once.

3.2 Transmission Process of the Handshake Message

3.2.1 Handshake from Add-on to CAS

In the Carip.cfg configuration file, it is possible to set a property indicating whether or not a handshake message has to be sent out and what the frequency of that heartbeat message should be.

When the CAS receives the handshake message it should respond with the same message of which the time stamp has changed and byte 4 changes (from 0x04) into 0x02.

If the CAS does not respond within the number of times specified in the configuration file the add-on will wait another 60 seconds before sending out a new handshake message.

Example of the parameters in the configuration file:

```
[Heartbeat]
// ---------------------------------------------------------------------------------------------------
// if a 'keepalive' message is expected by peer then set heartbeat = yes.
// otherwise set heartbeat = no
// 'rythm' is the frequency of the heartbeat expressed in milliseconds.
// "error_rythm' = is the frequency of the heartbeat when no reply is received expressed in milliseconds.
// 'error_max' = maximum number of retries for heartbeat messages
// ---------------------------------------------------------------------------------------------------
heartbeat=yes
rythm=120000
error_rythm=15000
error_max_retries=5
```

3.2.2 Handshake from CAS to Add-on

The CAS sends out a handshake message to which the add-on must respond with the same message format, but with the actual date and timestamp.

4. CONFIGURATION

The configuration file '<HiProvision path>\bin\Carip.cfg' contains all possible settings for the CAR IP add-on. Following table provides an overview of these parameters together with their meaning and the default values.

Table 9 Settings in the Carip.cfg File

Section	Setting	Description	Default value (possible value(s))
[Ip]	Ip_address	IP address of the CAS	127.0.0.1
	Ip_port	IP port on which the CAR IP sends out the messages	5555
	ReceiveIp_port	IP port on which the CAR IP receives messages	5556
[Protocol]	ProtocolVersion	This setting configures which protocolversion to use in a Failure alarm message, either version 1, 2, 3 or 4. (see Table 2, Table 3, Table 4)	1 (2, 3, 4)

	NodeNumberUseStationNumber	This setting configures if higher node numbers, using a second byte (=extended mode), are allowed.
		value = 1 (normal mode)
		//byte 1: Station number
		//byte 2: Node number
		
		value = 2 (extended mode)
		//byte 1: Node number high byte
		//byte 2: Node number low byte
			1 (2)

	SubSystemCode	Indicates which subsystem code the application will use in the alarm messages	1 (1..255)
[Time]	UTC	Time format of the alarm message (UTC or local time)	Yes (No)
[Flow]	Max_retries	The number of times the failure alarm or failure repair message should be sent out again in case of no response	5
	Delay	Time between the different retries in milliseconds	10000
	Autoacknowledge_enabled	Not supported	
[Heartbeat]	Heartbeat	Whether or not the add-on should send a heartbeat to the CAS	Yes (no)
	Rythm	Frequency of the heartbeat in milliseconds	60000
	Error_rythm	Frequency of the heartbeat in case of no response to the heartbeat, in milliseconds	60000
	Error_max_retries	Maximum number of retries for heartbeat messages	5

Section	Setting	Description	Default value (possible value(s))
[Node]	e.g. <Node Number> 50=12 (50 = Node Number, 12 = Station Number) 51=12 60=13 61=13 40=14	Assign station numbers (e.g 12) to nodes that belong to the same location (e.g. Node 50 and 51).	<Station Number>
[Slot]		The number of the interface slot that reports the alarm	/HiProvision module mapping IFM-1 = 21 IFM-2 = 22 IFM-3 = 23 IFM-4 = 24 IFM-5 = 25 IFM-6 = 26 IFM-7 = 27 IFM-8 = 28 IFM-9 = 29 IFM-10 = 30
[Alarms]	e.g. 5.0 = 1	Each HiProvision alarm has a unique code which will be translated in this section to a failure code	The alarm codes are grouped per interface module.

5. EXAMPLE MESSAGES

Output example:

Start receiving messages

```
284 2005/12/23 12:55:50.301 [IPv4 ] (?) (~) Socket(udp).reuse(true) - (handle = 0x00000724)
284 2005/12/23 12:55:50.301 [IPv4 ] (?) (~) Socket(udp).reuse(true) - (handle = 0x0000071c)
284 2005/12/23 12:55:50.301 [IPv4 ] (?) (+) Socket(udp, 0b0).construct() - (handle = 0x0000071c)
284 2005/12/23 12:55:50.301 [IPv4 ] (?) (+) Socket(udp).bind(0.0.0.0:5000) - (handle = 0x00000724)
284 2005/12/23 12:55:50.301 [IPv4 ] (?) (+) Socket(udp).connect(192.168.100.30:5001) - (handle =
0x0000071c)
Received 13 Bytes : 0x01 0x01 0x0a 0x04 0x00 0xb5 0x14 0x05 0x0c 0x17 0x0c 0x36 0x04
```

Replying to alarm message

```
Received 13 Bytes : 0x01 0x01 0x0a 0x07 0x00 0xb4 0x14 0x05 0x0c 0x17 0x0c 0x36 0x04
```

Replying to alarm message

```
Received 13 Bytes : 0x01 0x01 0x0a 0x08 0x00 0xb4 0x14 0x05 0x0c 0x17 0x0c 0x36 0x04
```

Replying to alarm message

```
Received 13 Bytes : 0x01 0x01 0x0a 0x10 0x00 0x96 0x14 0x05 0x0c 0x17 0x0c 0x36 0x04
```

Replying to alarm message

```
Received 9 Bytes : 0xff 0x01 0x14 0x05 0x0c 0x17 0x0d 0x36 0x33
```

6. ABBREVIATIONS

CAS	Central Alarm System
CAR IP	Central Alarm Reporter Internet Protocol
GUI	Graphical User Interface
LNM	Large Network Monitor
MPLS-TP	Multiprotocol Label Switching – Transport Profile
PTN	Packet Transport Network
UDP	Universal Data Protocol
URL	Uniform Resource Locator
UTC	Coordinated Universal Time