1. Binary message.

message structure: [header, user_data, tail]

header:

header =

byte 1	byte 2	byte 3	byte 4
msg_size LSB	msg_size MSB	flags 8bit	cmdID 8bit

message size (**msg_size**) includes size of header, user_data and tail.
cmdID=0x00 is special echo command and should not be used.

flags:

bit 8	bit 7	bit 6	bit 5
reserved	reserved	reserved	reserved

bit 4	bit 3	bit 2	bit 1
reserved	INTEGRITY_STAMP	SUPPRESS_REPLY_HEADER	EXTANDED_HEADER

EXTANDED_HEADER : should be 0 for now.

SUPPRESS_REPLY_HEADER:
if 0:
reply will include header: [header, data]. this is recommended option.
if 1:
no header inserted in reply. reply structure: [data]

INTEGRITY_STAMP:
if 0:
tail will not include integrity stamp.
if 1:
a) tail will include 4 bytes to check message integrity: |msg_size LSB | msg_size MSB | 0x00 | 0x00 |
b) reply will also include integrity stamp of the same structure

2. Reply
reply syntax: [header, data, tail]

header:

header =

byte 1	byte 2	byte 3	byte 4
msg_size LSB	msg_size MSB	0x00	replyStatus

message size (**msg_size**) includes size of header and data.

replyStatus :
0x00 - NO ERROR
0x01 - command not found
0x02 - test of integrity stamp failed
0x03 - message length is too long

tail:
If command includes integrity stamp bit then reply will include tail automatically in following format:
tail =

byte 1	byte 2	byte 3	byte 4
msg_size LSB	msg_size MSB	0x00	0x00