MeshCore Payload Format Reference

Payload Format

Inside each MeshCore Packet is a payload, identified by the payload type in the packet header. The types of payloads are:

- Node advertisement.
- Acknowledgment.
- Returned path.
- Request (destination/source hashes + MAC).
- Response to REQ or ANON_REQ.
- Plain text message.
- Anonymous request.
- Group text message (unverified).
- Group datagram (unverified).
- Multi-part packet
- Control data packet
- Custom packet (raw bytes, custom encryption).

This document defines the structure of each of these payload types.

NOTE: all 16 and 32-bit integer fields are Little Endian.

Important concepts:

- Node hash: the first byte of the node's public key

Node advertisement

This kind of payload notifies receivers that a node exists, and gives information about the node

Field	Size (bytes)	Description
public key	32	Ed25519 public key of the node
timestamp	4	unix timestamp of advertisement
signature	64	Ed25519 signature of public key, timestamp, and app data
appdata	rest of payload	optional, see below

Appdata

Field	Size (bytes)	Description
flags	1	specifies which of the fields are present, see below
latitude	4 (optional)	decimal latitude multiplied by 1000000, integer
longitude	4 (optional)	decimal longitude multiplied by 1000000, integer
feature 1	2 (optional)	reserved for future use
feature 2	2 (optional)	reserved for future use
name	rest of appdata	name of the node

Appdata Flags

Value	Name	Description
`0x01`	is chat node	advert is for a chat node
`0x02`	is repeater	advert is for a repeater
`0x03`	is room server	advert is for a room server
`0x04`	is sensor	advert is for a sensor server
`0x10`	has location	appdata contains lat/long information
`0x20`	has feature 1	Reserved for future use.
`0x40`	has feature 2	Reserved for future use.
`0x80`	has name	appdata contains a node name

Acknowledgement

An acknowledgement that a message was received. Note that for returned path messages, an acknowledgement can be sent in the "extra" payload (see Returned Path) instead of as a separate

ackowledgement packet. CLI commands do not cause acknowledgement responses, neither discrete nor extra.

Field	Size (bytes)	Description
checksum	4	CRC checksum of message timestamp, text, and sender pubkey

Returned path, request, response, and plain text message

Returned path, request, response, and plain text messages are all formatted in the same way. See the subsection for more details about the ciphertext's associated plaintext representation.

Field	Size (bytes)	Description
destination hash	1	first byte of destination node public key
source hash	1	first byte of source node public key
cipher MAC	2	MAC for encrypted data in next field
ciphertext	rest of payload	encrypted message, see subsections below for details

Returned path

Returned path messages provide a description of the route a packet took from the original author. Receivers will send returned path messages to the author of the original message.

Field	Size (bytes)	Description
path length	1	length of next field
path	see above	a list of node hashes (one byte each)
extra type	1	extra, bundled payload type, eg., acknowledgement or response. Same values as in Packet Format
extra	rest of data	extra, bundled payload content, follows same format as main content defined by this document

Request

Field	Size (bytes)	Description
timestamp	4	sender time (unix timestamp)
request data	rest of payload	application-defined request payload body

For the common chat/server helpers in `BaseChatMesh`, the current request type values are:

Value	Name	Description
`0x01`	get stats	get stats of repeater or room server
`0x02`	keepalive	keep-alive request used for maintained connections

Get stats

Gets information about the node, possibly including the following:

- Battery level (millivolts)
- Current transmit queue length
- Current free queue length
- Last RSSI value
- Number of received packets
- Number of sent packets
- Total airtime (seconds)
- Total uptime (seconds)
- Number of packets sent as flood
- Number of packets sent directly
- Number of packets received as flood
- Number of packets received directly
- Error flags
- Last SNR value
- Number of direct route duplicates
- Number of flood route duplicates
- Number posted (?)
- Number of post pushes (?)

Get telemetry data

Not defined in `BaseChatMesh`. Sensor- and application-specific request payloads may be implemented by higher-level firmware.

Get Telemetry

Not defined in `BaseChatMesh`.

Get Min/Max/Ave (Sensor nodes)

Not defined in `BaseChatMesh`.

Get Access List

Not defined in `BaseChatMesh`.

Get Neighbors

Not defined in `BaseChatMesh`.

Get Owner Info

Not defined in `BaseChatMesh`.

Response

Field	Size (bytes)	Description
content	rest of payload	application-defined response body

Response contents are opaque application data. There is no single generic response envelope beyond the encrypted payload wrapper shown above.

Plain text message

Field	Size (bytes)	Description
timestamp	4	send time (unix timestamp)
txt_type + attempt	1	upper six bits are txt_type (see below), lower two bits are attempt number (0..3)
message	rest of payload	the message content, see next table

txt_type

Value	Description	Message content
`0x00`	plain text message	the plain text of the message
`0x01`	CLI command	the command text of the message

Value	Description	Message content
`0x02`	signed plain text message	first four bytes is sender pubkey prefix, followed by plain text message

Anonymous request

Field	Size (bytes)	Description
destination hash	1	first byte of destination node public key
public key	32	sender's Ed25519 public key
cipher MAC	2	MAC for encrypted data in next field
ciphertext	rest of payload	encrypted message, see below for details

Room server login

Field	Size (bytes)	Description
timestamp	4	sender time (unix timestamp)
sync timestamp	4	sender's "sync messages SINCE x" timestamp
password	rest of message	password for room

Repeater/Sensor login

Field	Size (bytes)	Description
timestamp	4	sender time (unix timestamp)
password	rest of message	password for repeater/sensor

Repeater - Regions request

Field	Size (bytes)	Description
timestamp	4	sender time (unix timestamp)
req type	1	0x01 (request sub type)

Field	Size (bytes)	Description
reply path len	1	path len for reply
reply path	(variable)	reply path

Repeater - Owner info request

Field	Size (bytes)	Description
timestamp	4	sender time (unix timestamp)
req type	1	0x02 (request sub type)
reply path len	1	path len for reply
reply path	(variable)	reply path

Repeater - Clock and status request

Field	Size (bytes)	Description
timestamp	4	sender time (unix timestamp)
req type	1	0x03 (request sub type)
reply path len	1	path len for reply
reply path	(variable)	reply path

Group text message

Field	Size (bytes)	Description
channel hash	1	first byte of SHA256 of channel's shared key
cipher MAC	2	MAC for encrypted data in next field
ciphertext	rest of payload	encrypted message, see below for details

The plaintext contained in the ciphertext matches the format described in plain text message. Specifically, it consists of a four byte timestamp, a flags byte, and the message. The flags byte will generally be `0x00` because it is a "plain text message". The message will be of the form `: ` (eg., `user123: I'm on my way`).

Group datagram

Field	Size (bytes)	Description
channel hash	1	first byte of SHA256 of channel's shared key
cipher MAC	2	MAC for encrypted data in next field
ciphertext	rest of payload	encrypted data, see below for details

The data contained in the ciphertext uses the format below:

Field	Size (bytes)	Description
data type	2	Identifier for type of data. (See number_allocations.md)
data len	1	byte length of data
data	rest of payload	(depends on data type)

Control data

Field	Size (bytes)	Description
flags	1	upper 4 bits is sub_type
data	rest of payload	typically unencrypted data

DISCOVER_REQ (sub_type)

Field	Size (bytes)	Description
flags	1	0x8 (upper 4 bits), prefix_only (lowest bit)
type_filter	1	bit for each ADV_TYPE_*
tag	4	randomly generate by sender
since	4	(optional) epoch timestamp (0 by default)

DISCOVER_RESP (sub_type)

Field	Size (bytes)	Description
flags	1	0x9 (upper 4 bits), node_type (lower 4)
snr	1	signed, SNR*4
tag	4	reflected back from DISCOVER_REQ
pubkey	8 or 32	node's ID (or prefix)

Custom packet

Custom packets have no defined format.

Revision #3
Created 2026-05-03 05:05:14 UTC by Mesh America Admin
Updated 2026-06-09 14:41:09 UTC by Mesh America Admin