Shim6 protocol

Erik Nordmark

Overview

- Nothing changed since l3shim
  - Placement of shim, principles of mapping ULID and locators
- But picked some design decisions (somewhat arbitrarily)
  - In order to work out the details and find the holes
- We're far from done with the draft
- There are some open issues in the draft
- This morning we will *enumerate* any other ones that come up
  - Discuss them later

Design decisions

- Put the context tag in the flow label field
  - shim allocates a flow label used for data packets sent after a failure
  - also used as “context tag” in shim control messages
  - appears to imply protocol number overload
- Do uncoordinated shim state removal
- An error message to discover when peer removed/lost state
- Use a new IP protocol number for the control protocol (akin to being parallel to ICMP)

Remaining design decisions

- Are we doing CUD or FBD for reachability detection?
- How does the pair exploration protocol work? (to find a working pair after failure)
- Failure during initial contact: do we need support in the shim?
  - Need the complete story starting with multiple AAAA in the DNS, some failure, until we have working communication
- Note: draft has placeholder messages for the above

Protocol Messages

- Context establishment
  - 4 messages (I1, R1, I2, R2)
- No context error message
  - In response to data packets as well as control packets
- Locator list update (and ack)
- Rehome request (locator preference update) and ack
- Payload
  - For things that don't use an overloaded protocol number

Placeholder Messages

- CUD: Reachability Probe and Reply
- FBD: Keepalive message
- Locator pair test and reply
  - Speculation: do we need something to handle failure during initial contact
- Context explore message
  - To find what locator pairs are working after a failure

Flow label usage

- ULP picks flow label per flow label RFC
- When shim context established, each end picks a 20 bit context tag. Used to find context in received packets.
  - <source locator, dest locator, context tag> identifies the context
- Context tag in a designated field in each shim control message
  - NOT in flow label field
- Data packets: when locator pair != ULID pair, sender puts receiver's context tag in flow label

Protocol field overloading

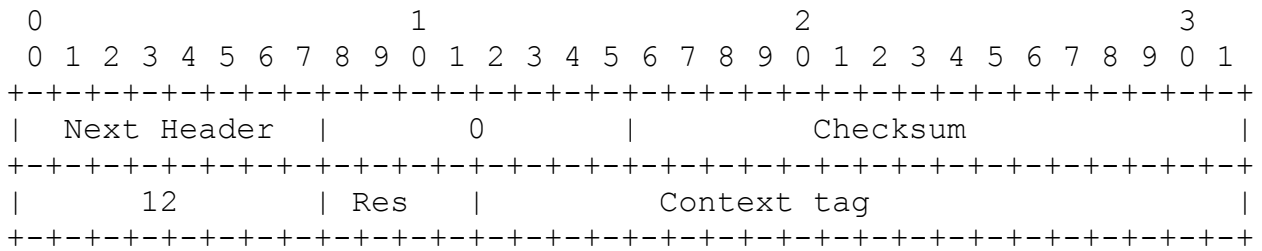
- Allocate IP protocol values for
  - TCP over shim
  - UDP over shim
  - etc
- Effectively a zero-length extension header
  - The context tag is carried in the flow label field

Need for this overloading?

- Identify to receiver that shim processing needs to be done
  - swap locators->ULIDs in IPv6 header before passing to ULP
- Say when to do this swapping
  - E.g., before or after processing something else in the header
- Efficiency: non-shim packets don't make the receiver lookup <src, dst, flow label> for a context
- Help detect lost context state (alternatives

Uncommon ULP payloads

- Have an 8-octet extension header (the shim6 payload message) to carry the next header value and the context tag



Fields:

Next Header: The payload which follows this header.

Hdr Ext Len: 0 (since the header is 8 octets).

Checksum: The checksum of the 8 octets.

Type: 12

Res: Reserved for future use. Zero on transmit. MUST be ignored on receipt.

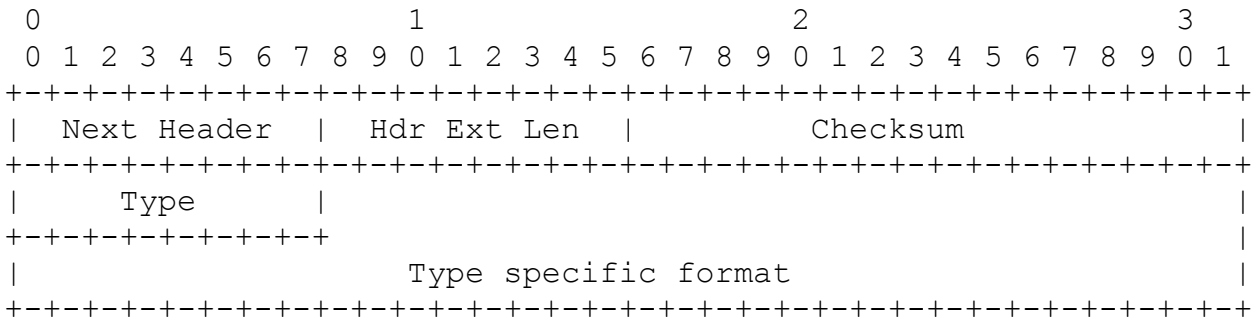
Context Tag: The context tag allocated by the receiver.

Context Establishment

- Normally
  - Send I1 Get I1: create no state
  - Get R1
  - Send I2 – include locator list
  - Get R2 – includes locator list
- Concurrent establishment
  - Crossing I1s; respond with (crossing) R2s
- Lost state: I1 can get R2 as response when responder has retained state

Shim message

- Base header: need IANA allocated protocol number



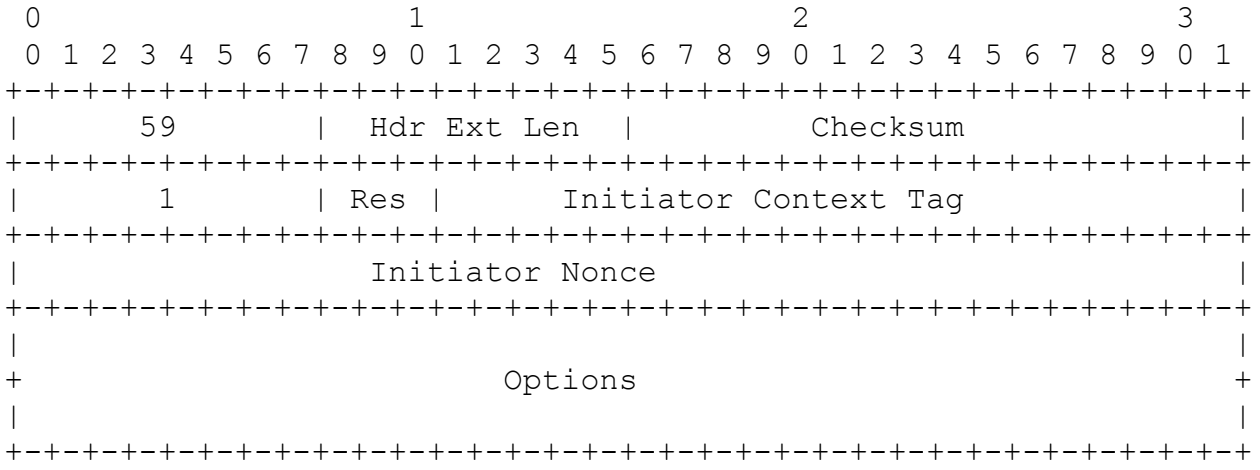
Fields:

Next Header: 8-bit selector. Normally set to NO_NXT_HDR (59). Indicates the next header value for the shim6 payload messages.

Hdr Ext Len: 8-bit unsigned integer. Length of the shim6 header in 8-octet units, not including the first 8 octets.

Checksum: 16-bit unsigned integer. The checksum is the 16-bit one's complement of the one's complement sum of the entire shim6 header message starting with the shim6 next header field, and ending as indicated by the Hdr Ext Len. Thus when there is a payload following the shim6 header, the payload is NOT included in the shim6 checksum.

I1 message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 1

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

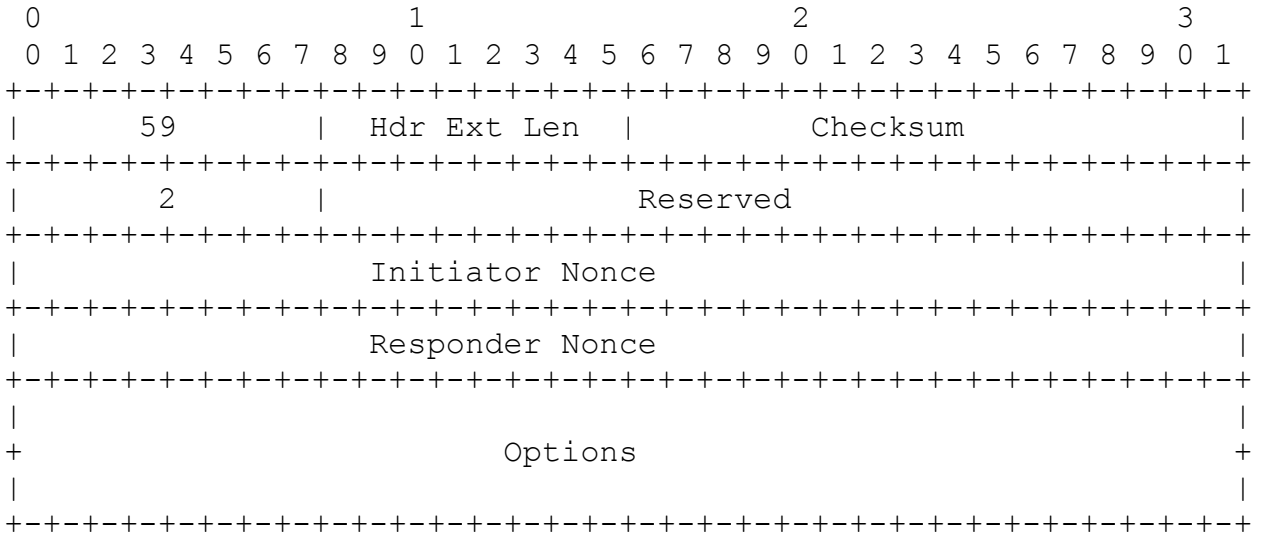
Initiator Context Tag: 20-bit field. The Context Tag the initiator has allocated for the context.

Initiator Nonce: 32-bit unsigned integer. A random number picked by the initiator which the responder will return in the R1 message.

The following options are allowed in the message:

ULID pair: TBD Do we need to carry the ULIDs, or assume they are the same as the address fields in the IPv6 header? Depends on how we handle failures during initial contact.

R1 message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 2

Reserved: 24-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

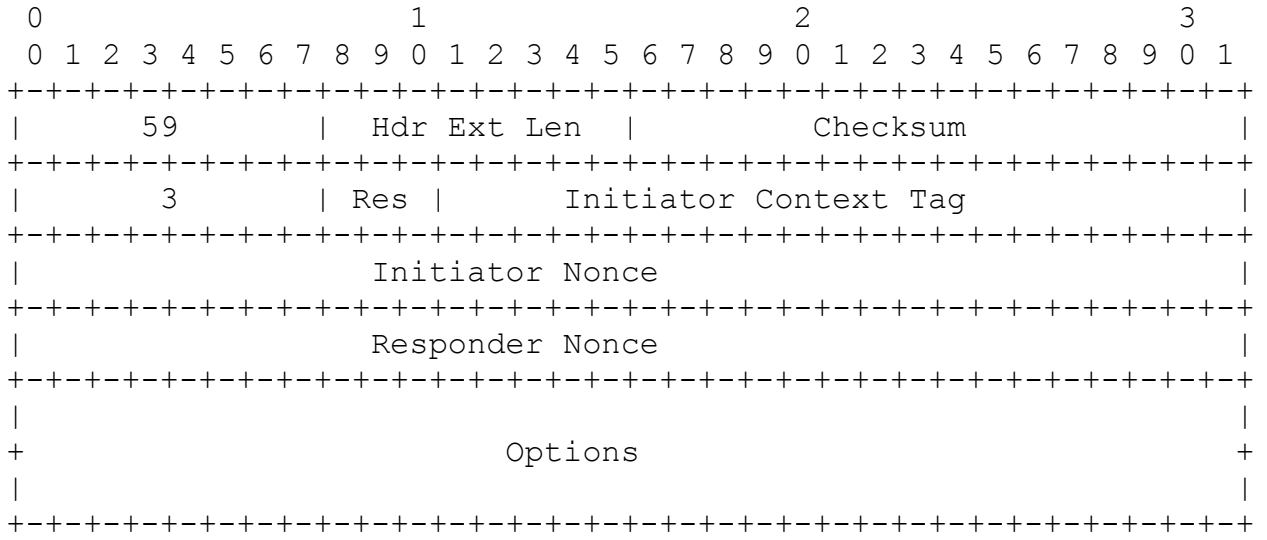
Initiator Nonce: 32-bit unsigned integer. Copied from the I1 message.

Responder Nonce: 32-bit unsigned integer. A number picked by the initiator which the initiator will return in the I2 message.

The following options are allowed in the message:

Responder Validator: Variable length mandatory option. Typically a hash generated by the responder, which the responder uses together with the Responder Nonce value to verify that an I2 message is indeed sent in response to a R1 message, and that the parameters in the I2 message are the same as those in the I1 message.

I2 message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 3

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Initiator Context Tag: 20-bit field. The Context Tag the initiator has allocated for the context

Initiator Nonce: 32-bit unsigned integer. A random number picked by the initiator which the responder will return in the R2 message.

Responder Nonce: 32-bit unsigned integer. Copied from the R1 message.

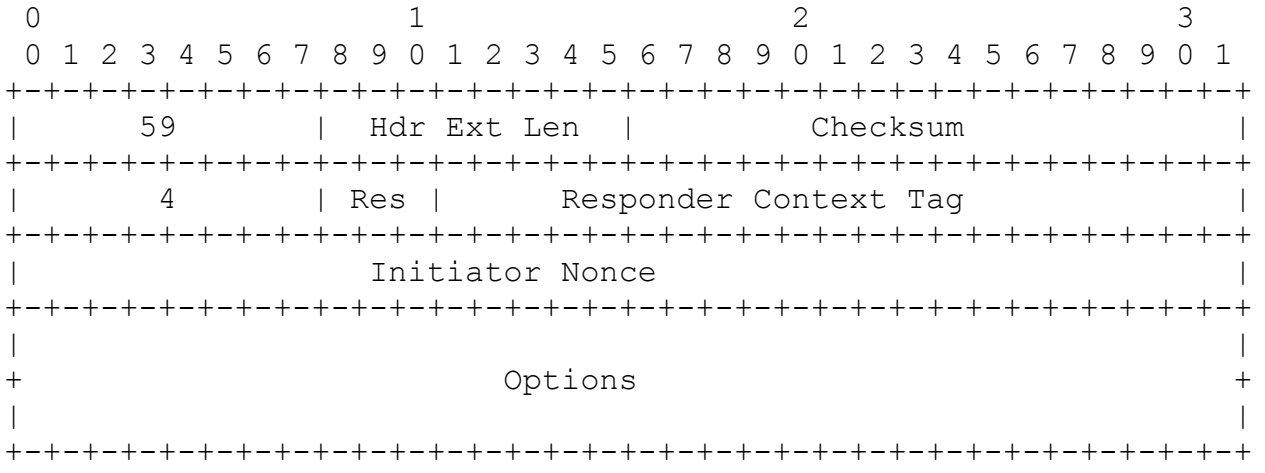
The following options are allowed in the message:

Responder Validator: Variable length mandatory option. Copied from the Validator in the R1 message.

ULID pair: TBD Do we need to carry the ULIDs, or assume they are the same as the address fields in the IPv6 header?

Locator list: Optionally sent when the initiator immediately wants to tell the responder its list of locators. When it

R2 message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 4

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Responder Context Tag: 20-bit field. The Context Tag the responder has allocated for the context

Initiator Nonce: 32-bit unsigned integer. Copied from the I2 message.

The following options are allowed in the message:

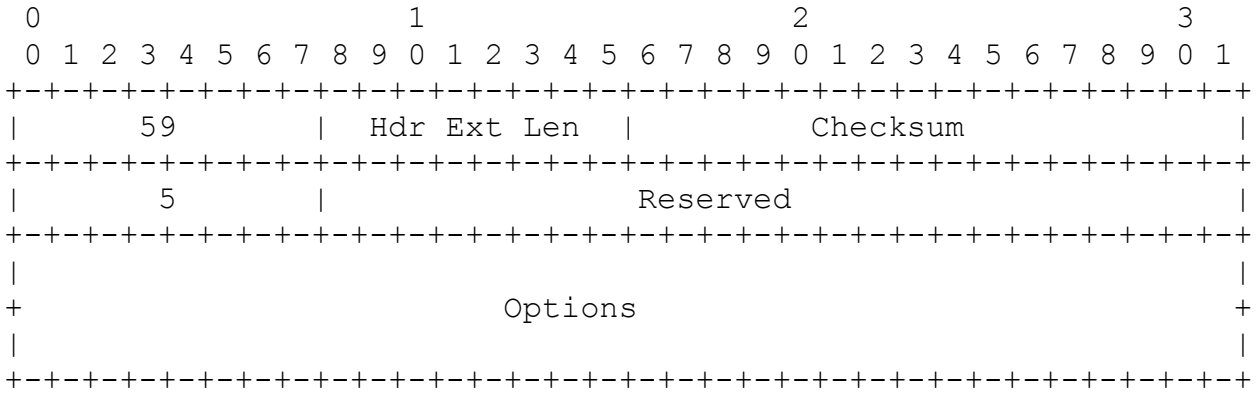
Locator List: Optionally sent when the responder immediately wants to tell the initiator its list of locators. When it is sent, the necessary HBA/CGA information for validating the locator list MUST also be included.

Locator Preferences: Optionally sent when the locators don't all have equal preference.

CGA Parameter Data Structure: Included when the locator list is included so the receiver can verify the locator list.

CGA Signature: Included when the some of the locators in the list use CGA (and not HBA) for validation.

No Context Error message



Fields:

Next Header: NO_NXT_HDR (59).

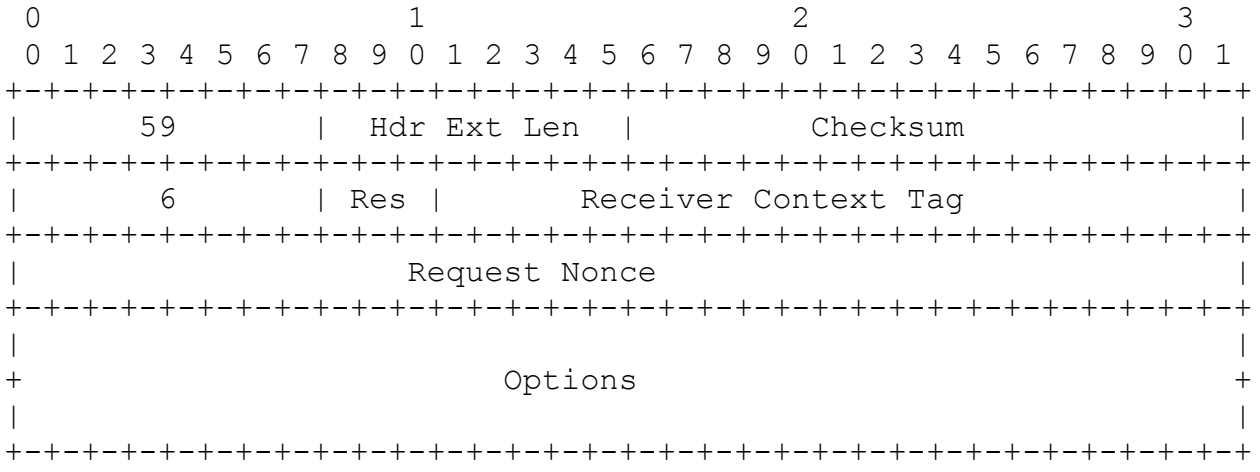
Type: 5

Reserved: 24-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

The following options are allowed in the message:

Packet in Error: Variable length mandatory option containing the IPv6 packet that was in error, starting with the IPv6 header, and normally containing the full packet. If the resulting No Context Error message would exceed 1280 octets, the Packet In Error option will not include the full packet in error in order to limit the error to 1280 octets.

Locator List Update message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 6

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Receiver Context Tag: 20-bit field. The Context Tag the receiver has allocated for the context.

Request Nonce: 32-bit unsigned integer. A random number picked by the sender which the receiver will return in the acknowledgement message.

The following options are allowed in the message:

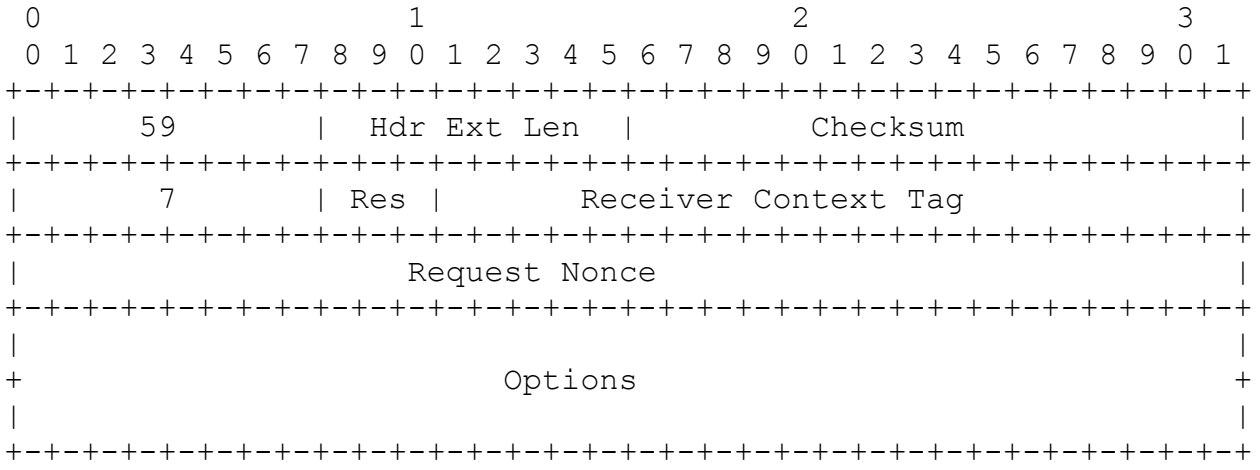
Locator List: The list of the senders (new) locators. The locators might be unchanged and only the preferences have changed.

Locator Preferences: Optionally sent when the locators don't all have equal preference.

CGA Parameter Data Structure: Included so the receiver can verify the locator list. **NOT NEEDED**

CGA Signature: Included when the some of the locators in the list use CGA (and not HBA) for validation.

Locator List Update Ack message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 7

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

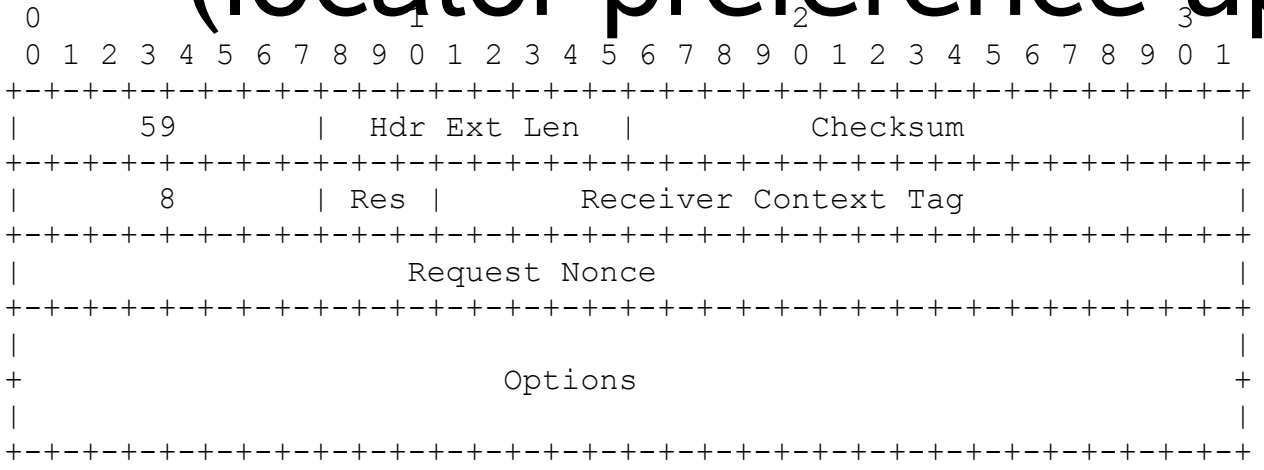
Receiver Context Tag: 20-bit field. The Context Tag the receiver has allocated for the context.

Request Nonce: 32-bit unsigned integer. Copied from the LLU message.

The following options are allowed in the message:

NONE

Rehome Request message (locator preference update)



Fields:

Next Header: NO_NXT_HDR (59).

Type: 8

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Receiver Context Tag: 20-bit field. The Context Tag the receiver has allocated for the context.

Request Nonce: 32-bit unsigned integer. Generated by the sender

The following options are allowed in the message:

Locator Preferences: Indicate which ones have failed.

Rehome Request Ack message (locator preference update)

Fields:

Next Header: NO_NXT_HDR (59).

Type: 9

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

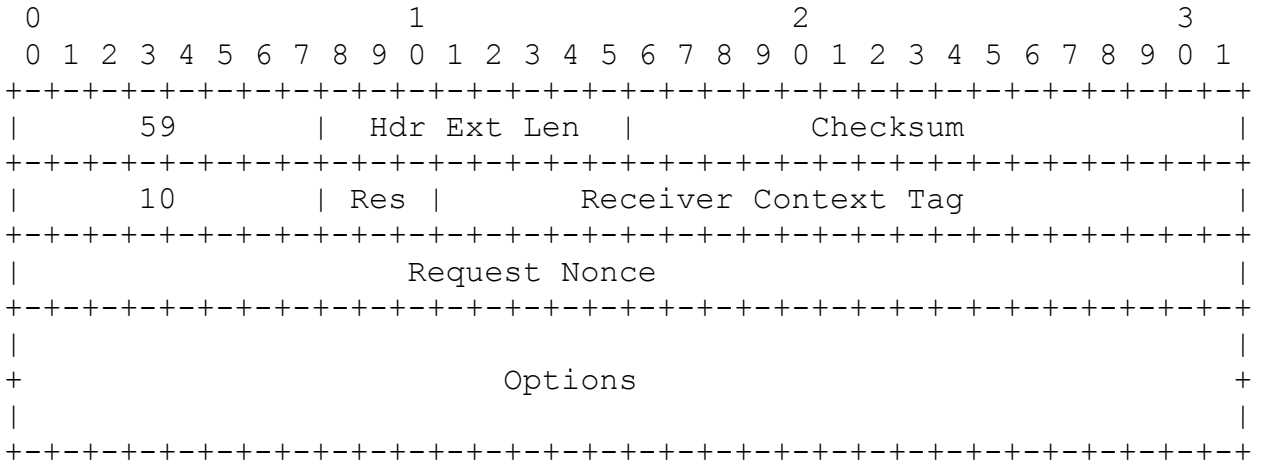
Receiver Context Tag: 20-bit field. The Context Tag the receiver has allocated for the context.

Request Nonce: 32-bit unsigned integer. Copied from the request.

The following options are allowed in the message:

NONE

Reachability Probe message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 10

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

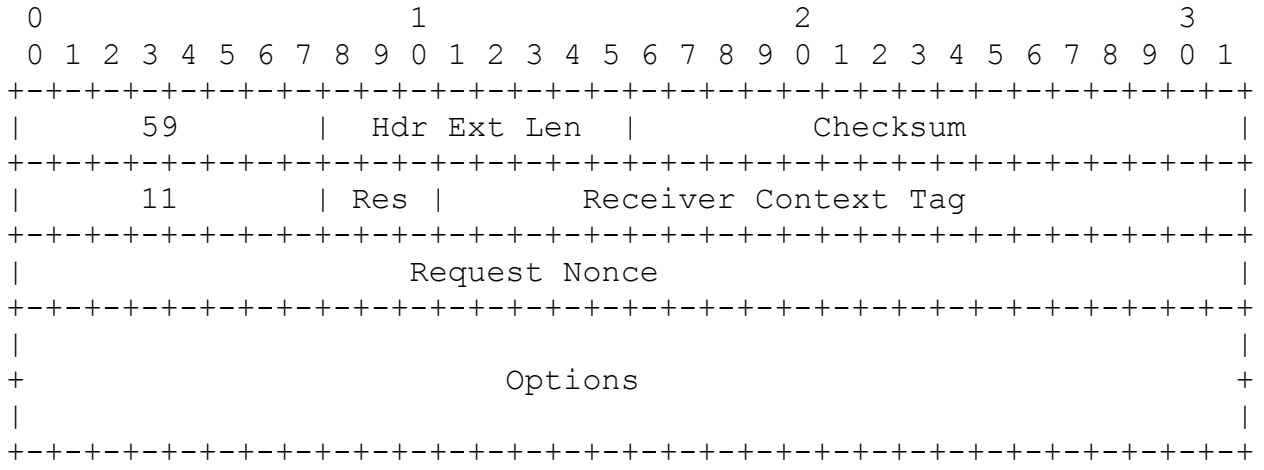
Receiver Context Tag: 20-bit field. The Context Tag the peer has allocated for the context.

Request Nonce: 32-bit unsigned integer. A random number picked by the initiator which the responder will return in the acknowledgement message.

The following options are allowed in the message:

ULID pair: The ULID pair that is being probed.

Reachability Probe Ack message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 11

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

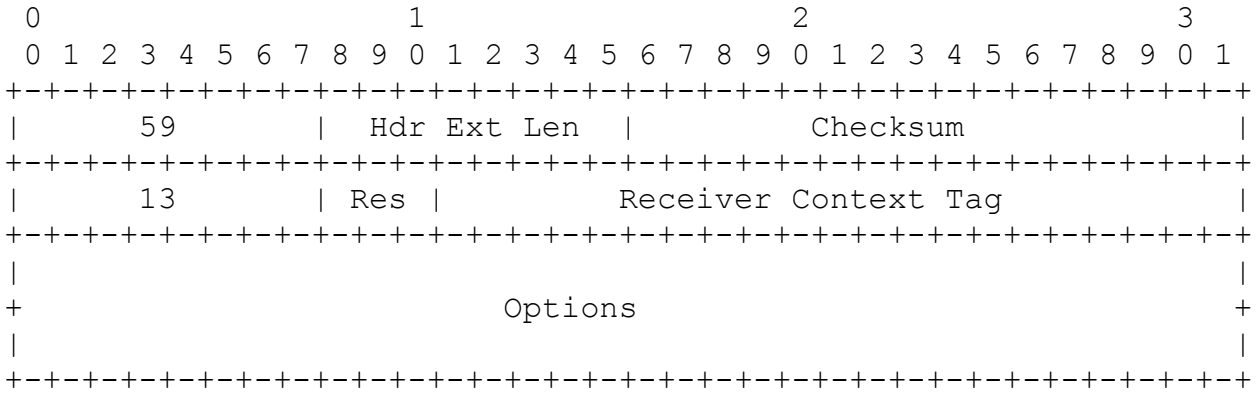
Receiver Context Tag: 20-bit field. The Context Tag the peer has allocated for the context.

Request Nonce: 32-bit unsigned integer. Copied from the request message.

The following options are allowed in the message:

ULID pair: The ULID pair that is being probed. Copied from the Probe message.

Keepalive message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 13

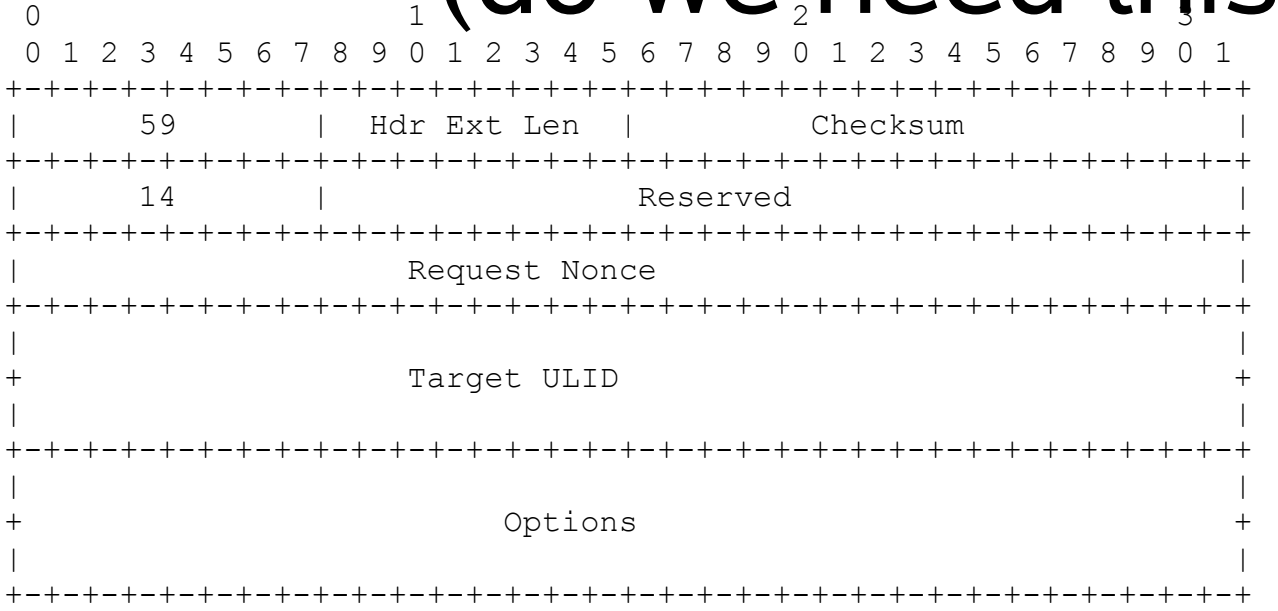
Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Receiver Context Tag: 20-bit field. The Context Tag the peer has allocated for the context.

The following options are allowed in the message:

TBD any options?:

Locator Pair Test message (do we need this)



Fields:

Next Header: NO_NXT_HDR (59).

Type: 14

Reserved: 24-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

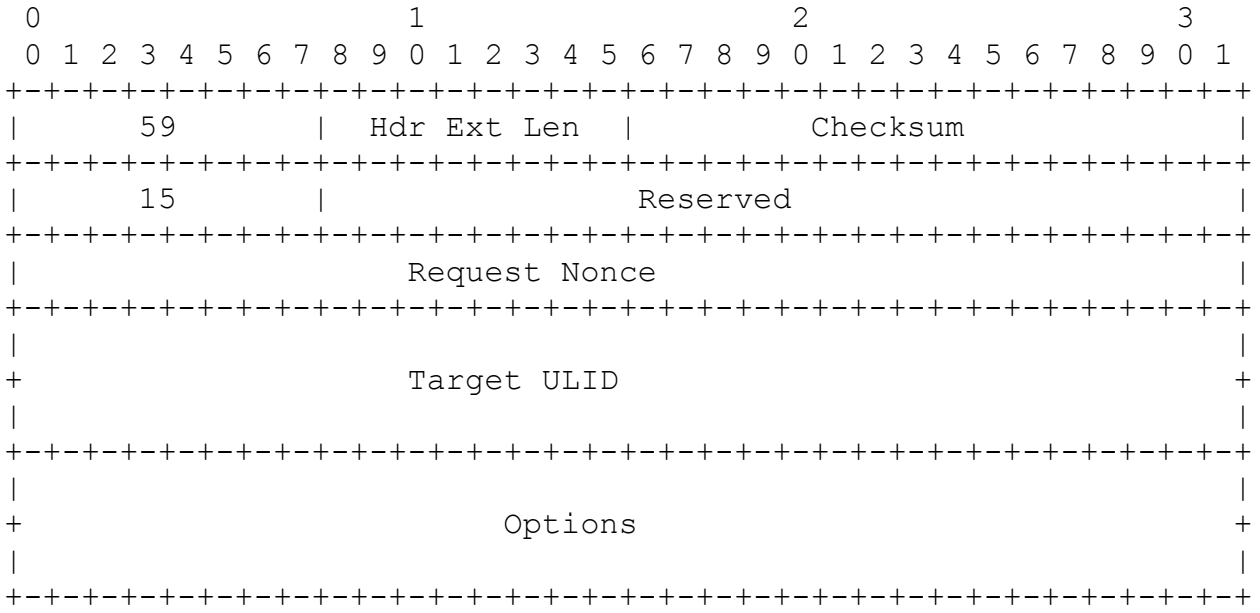
Request Nonce: 32-bit unsigned integer. A random number picked by the sender which the target will return in the reply message.

Target ULID: 128-bit IPv6 address.

The following options are allowed in the message:

TBD any options?:

Locator Pair Test Reply message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 15

Reserved: 24-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

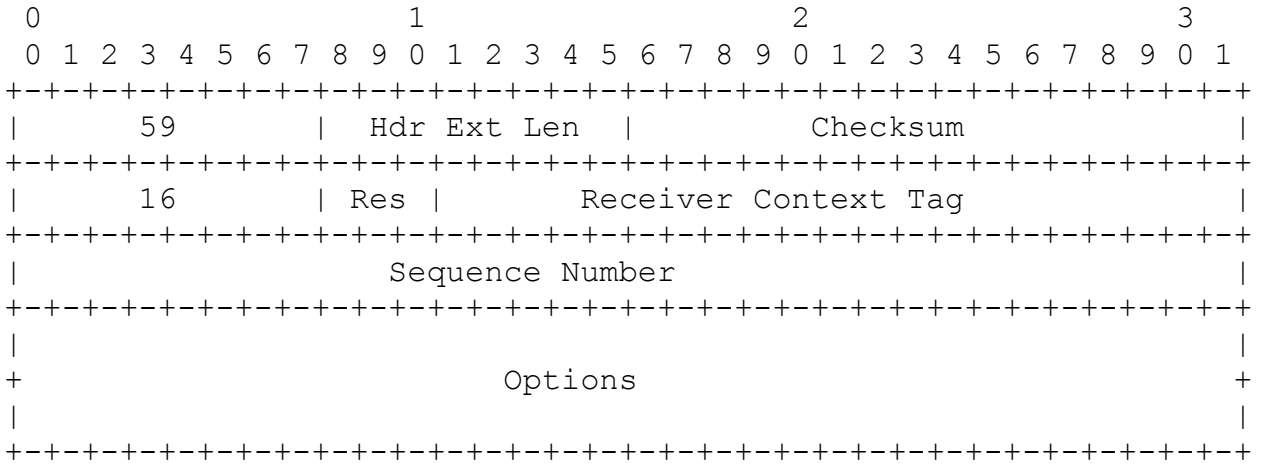
Request Nonce: 32-bit unsigned integer. Copied from the test message.

Target ULID: 128-bit IPv6 address. Copied from the test message. TBD: Or should the host be able to fill this in to make it easier for the peer to determine which locators refer to the same host?

The following options are allowed in the message:

TBD any options?:

Explore message



Fields:

Next Header: NO_NXT_HDR (59).

Type: 16

Res: 4-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

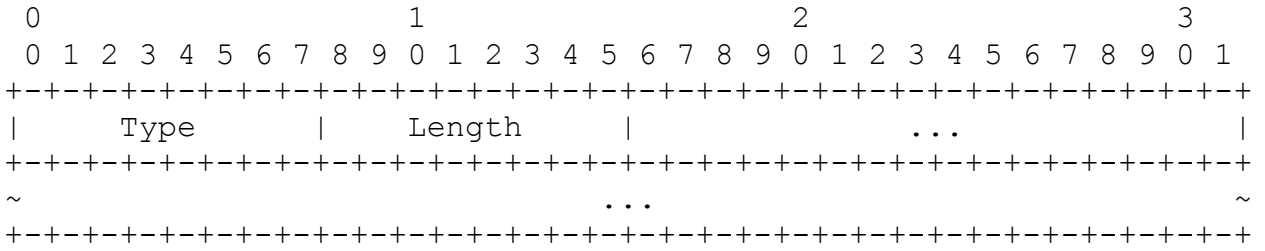
Receiver Context Tag: 20-bit field. The Context Tag the peer has allocated for the context.

Sequence Number: 32-bit unsigned integer. Used to determine which packets have been received by the peer.

The following options are allowed in the message:

Explorer Results: Indication of what Explorer messages the sender has recently received from the peer.

Option Formats



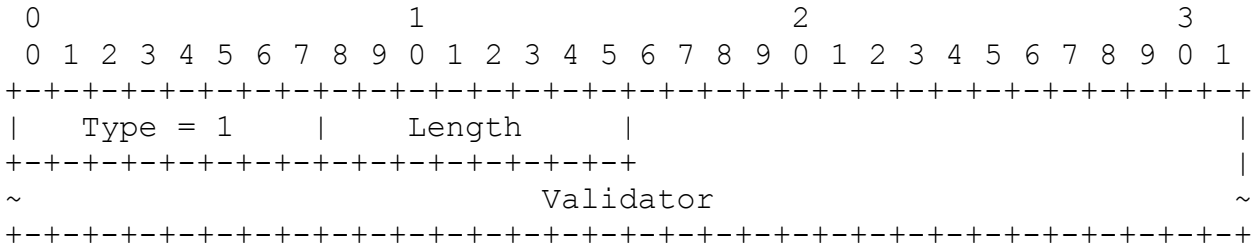
Fields:

Type: 8-bit identifier of the type of option. The options defined in this document are below.

Length: 8-bit unsigned integer. The length of the option (including the type and length fields) in units of 8 octets. The value 0 is invalid. Nodes MUST silently discard an ND packet that contains an option with length zero.

Option Name	Type
Validator	1
Locator List	2
Locator Preferences	3
CGA Parameter Data Structure	4
CGA Signature	5
ULID Pair	6
Packet In Error	7
Explorer Results	8

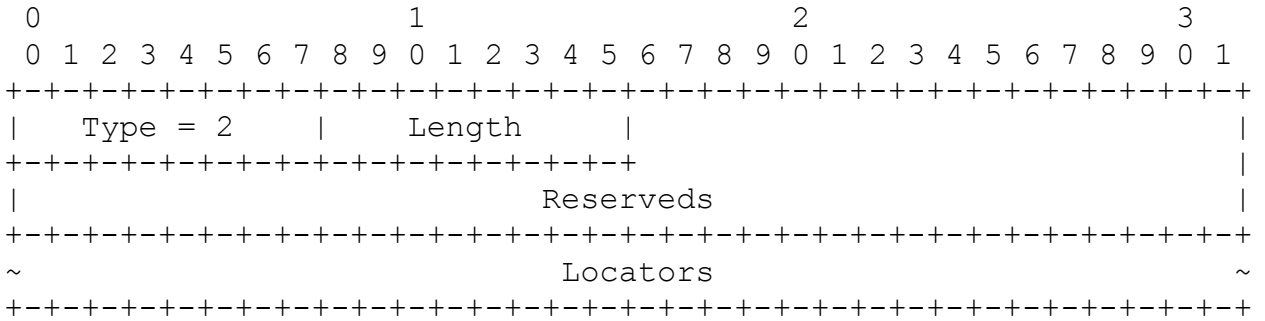
Validator Option Format



Fields:

Validator: Variable length content whose interpretation is local to the responder.

Locator List Option Format



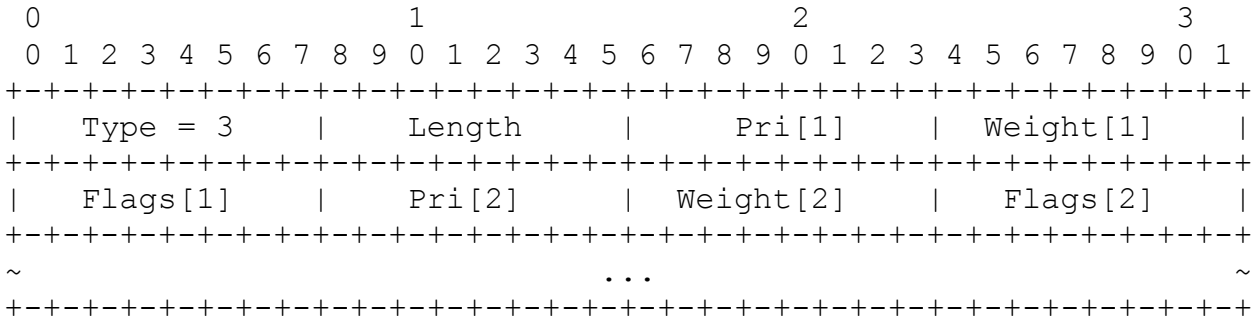
Fields:

Reserved: 48-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Locators: A variable number of 128-bit locators. The number of locators present can be determined by the option length field.

- Issues:
  - Need a generation/version count?
    - To make sure the preferences and explore results refer to the correct locators

Locator Preferences Option



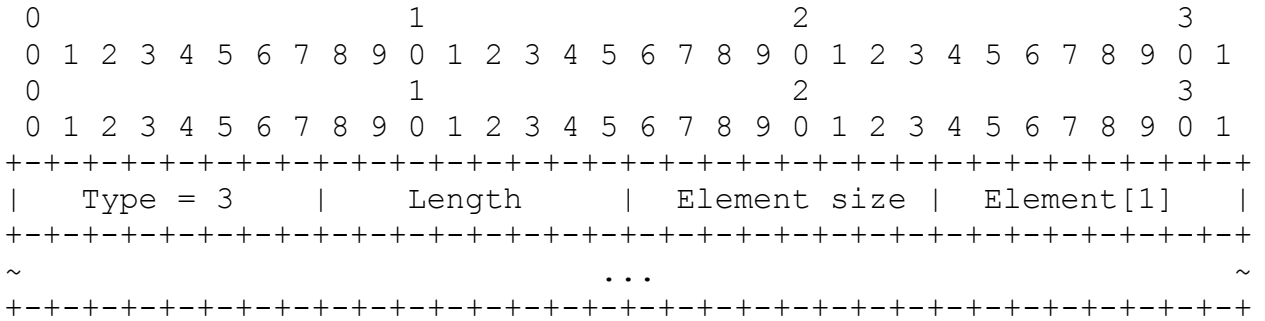
Fields:

Pri[i]: 8-bit unsigned integer. The Priority associated with the i'th locator in the Locator List option that is in use.

Weight[i]: 8-bit unsigned integer. The Weight associated with the i'th locator in the Locator List option that is in use.

Flags[i]: 8-bit unsigned integer. The flags associated with the i'th locator in the Locator List option that is in use.

New Locator Preferences Option



Fields:

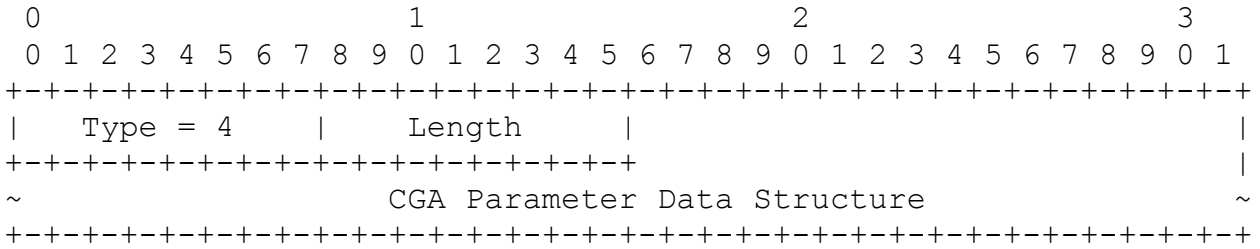
Element Size 8-bit unsigned integer. The size of each element in octets.
This version only defines the case of elelement size = 1

Element[i]: “element size” number of octets. The information associated with the i'th locator in the Locator List option that is in use.

The elemenent with contain 8 bits of flags. The set of flags is TBD: Assume there will be two initially: BROKEN and TEMPORARY.

- Later we can define e.g. a 3 octet element which has flags, priority, weight.
- Or flags + TLV encoding of rest.
- Note: assumes that each locator has the same associated element size

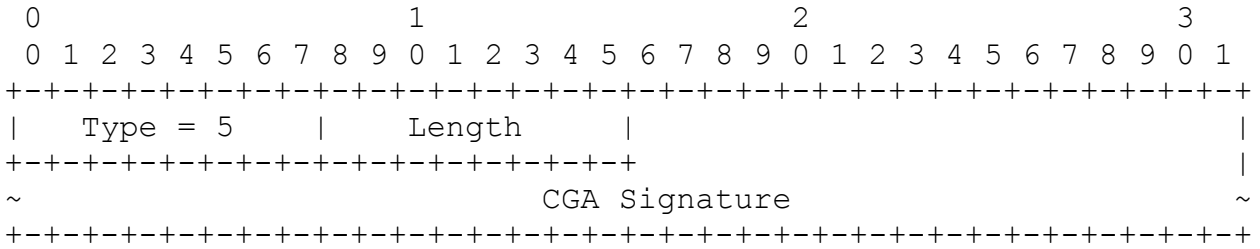
CGA Parameter Data Structure



Fields:

CGA Parameter Data Structure: Variable length content. Content defined in [4].

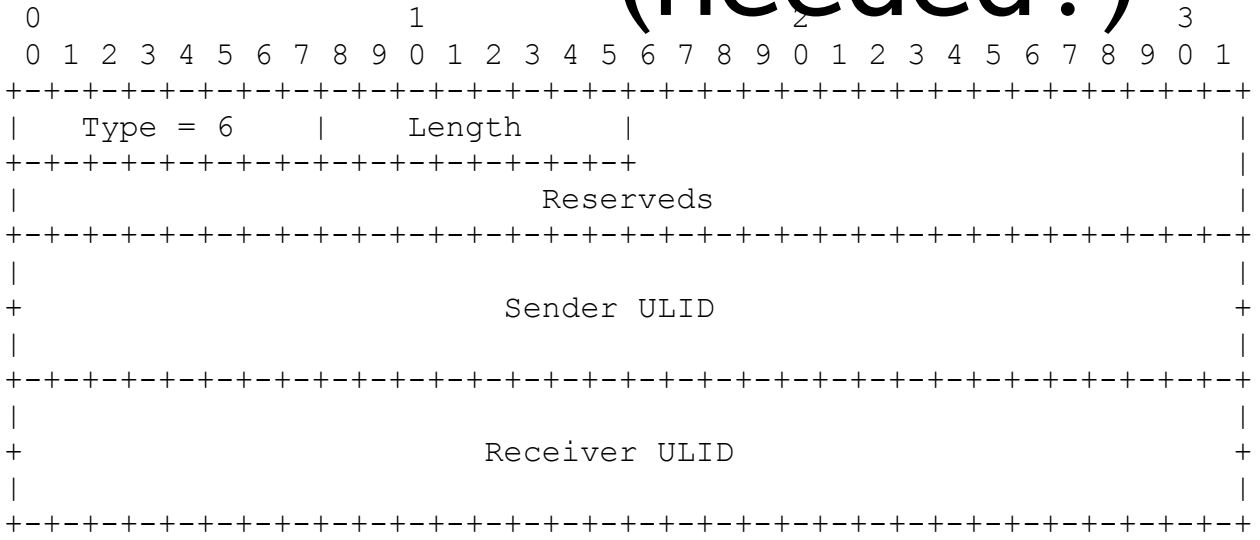
CGA Signature Option Format



Fields:

CGA Signature: Variable length content. Content defined in [4].

ULID Pair Option Format (needed?)



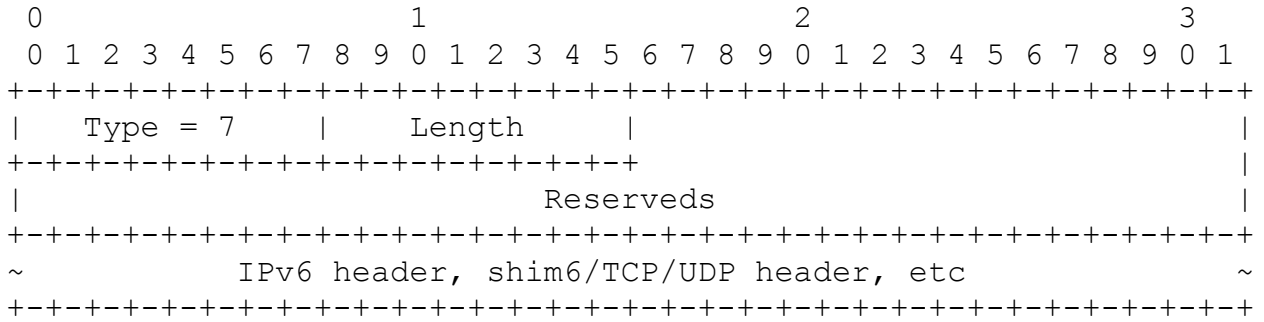
Fields:

Reserved: 48-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Sender ULID: A 128-bit IPv6 address.

Receiver ULID: A 128-bit IPv6 address.

Packet In Error Option Format

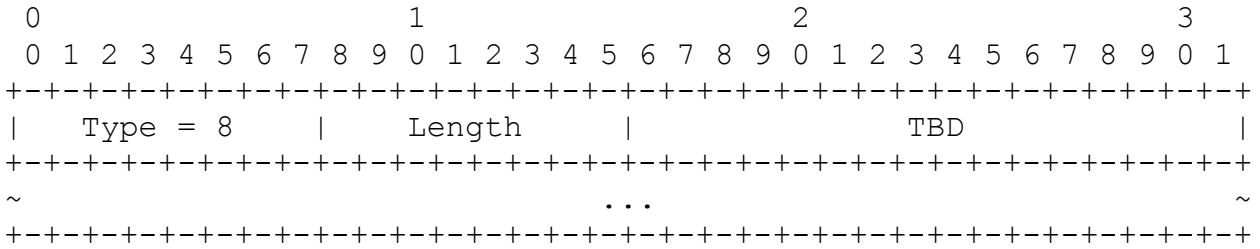


Fields:

Reserved: 48-bit field. Reserved for future use. Zero on transmit. MUST be ignored on receipt.

Packet: A variable length field which contains the packet in error starting with the IPv6 header.

Explorer Results Option Format



Fields:

TBD:

Open Issues (1)

- Need to keep list of locators private from on-path snoopers?
- Forking the context state? Versus the sender being able to use different locator pairs for different ULP communication?
  - Be able to do CUD/FBD per locator pair – not per ULID pair
- Extra bits: locator list option with all locators, and also in HBA parameter set
- Need generation/version for locator list

Open Issues (2)

- Detecting loss of context (using error message) doesn't detect loss while ULID pair works as the locator pair
  - So when failure, the peer might not have a context
  - Suggestions on how to solve?
- Which messages need sequence numbers? (if any)
- In the LLU do we need to indicate which locators should be verified with HBA, with CGA, and open for other future verification schemes?

Issues (3)

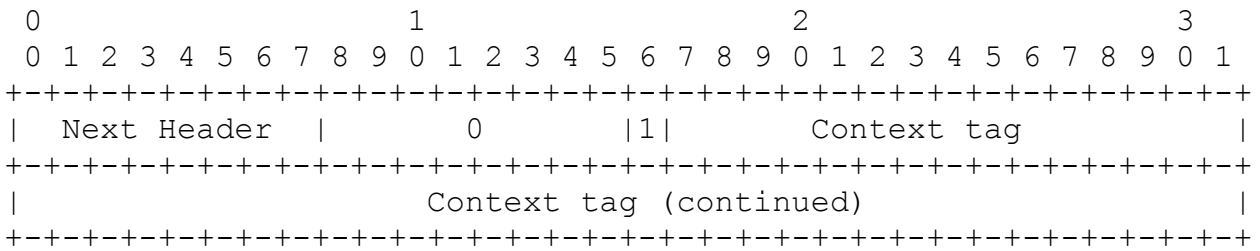
- What happens when we run out of 32 bit context tags?
  - When is it safe to reuse? (uncoordinated state removal)
  -

Design Alternatives

- State Cleanup
- Not overloading the flow label
- Detecting context loss
- ?? Not overloading the protocol number
  - Assumes or not, no flow label??

New ULP payloads

- Have an 8-octet extension header (the shim6 payload message) to carry the next header value and the context tag



Fields:

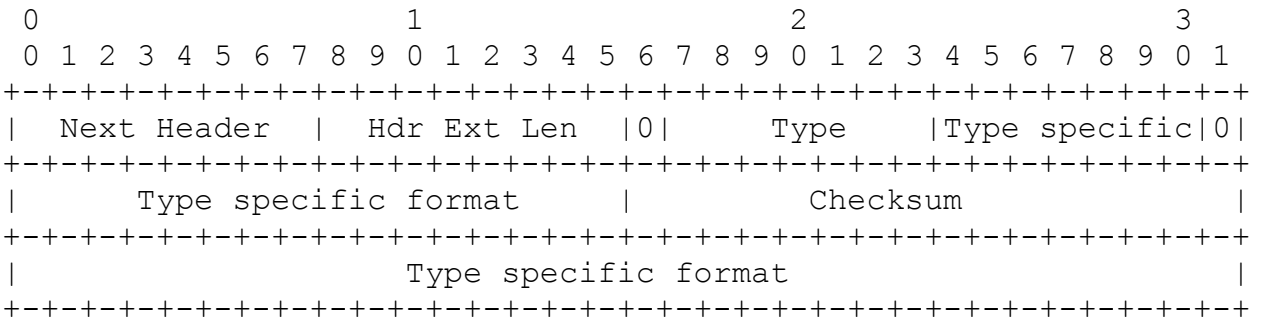
Next Header: The payload which follows this header.

Hdr Ext Len: 0 (since the header is 8 octets).

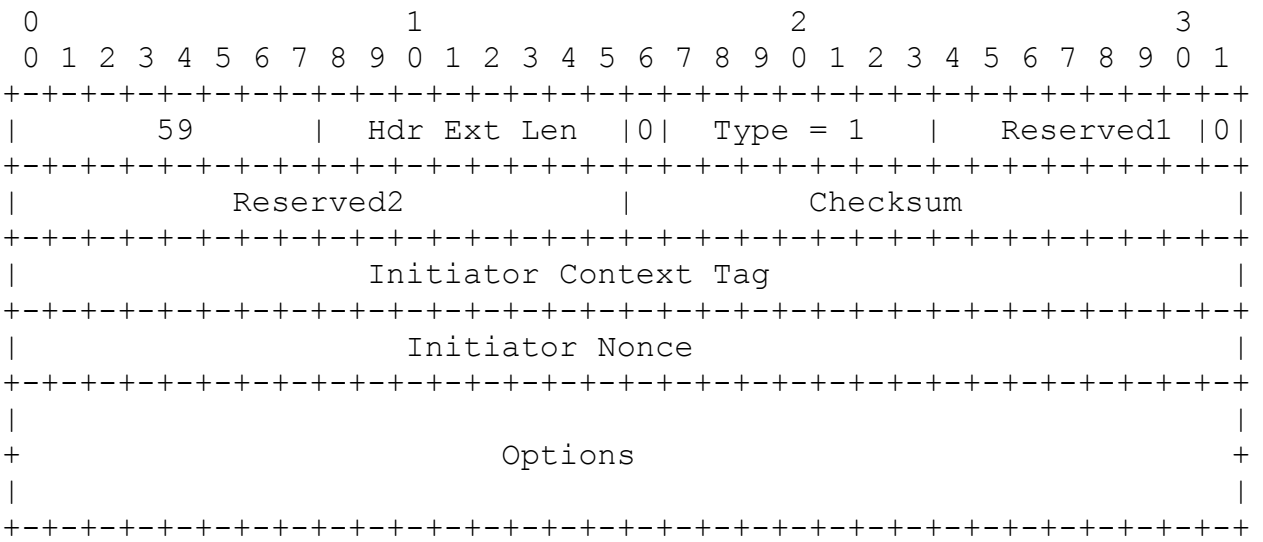
Context tag: 47 bits

New Shim message

- Base header

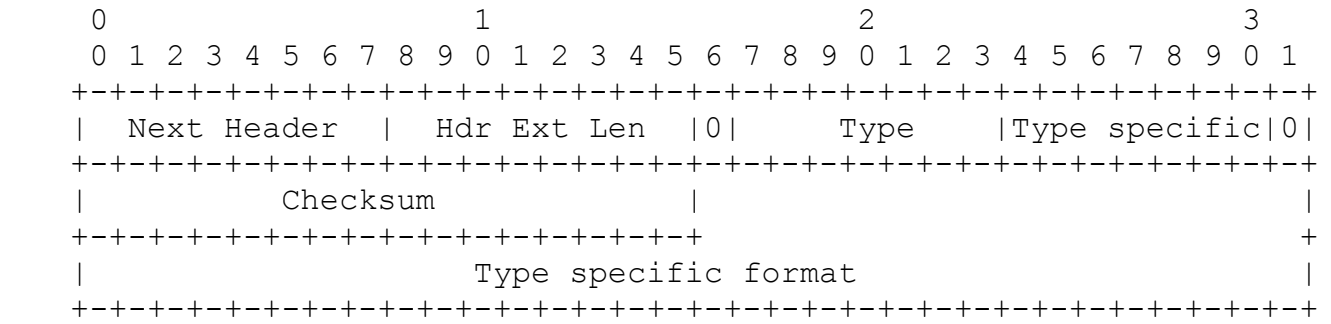


- Example: I1

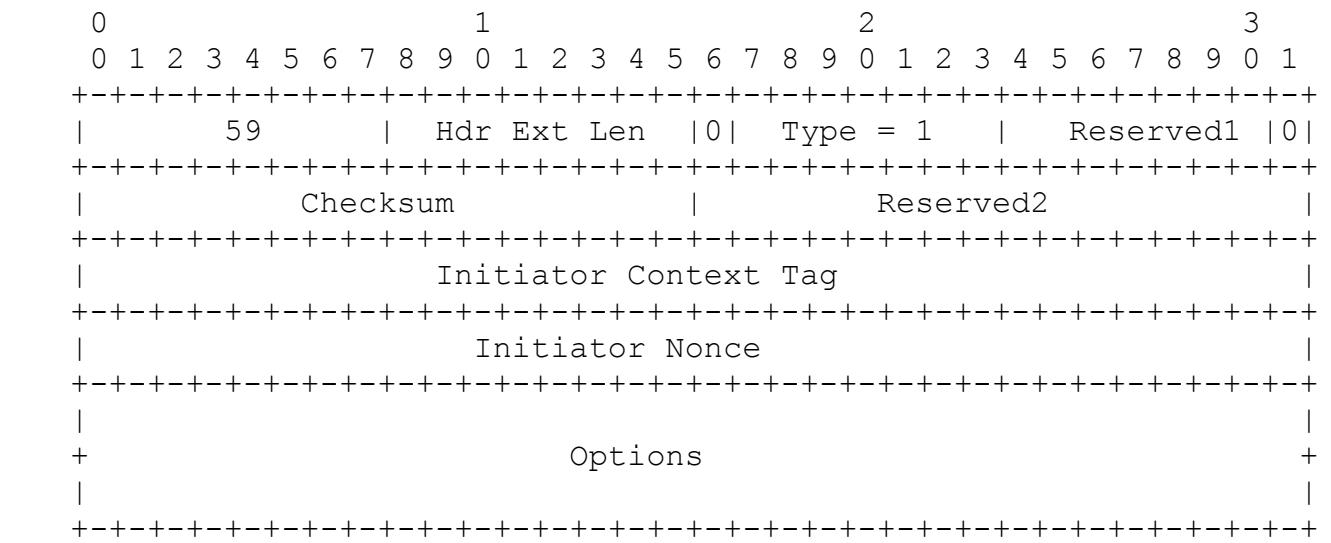


Newer Shim message

- Base header



- Example: I1



`