

Message Format

Formatting arbitrary and application-specific CCTP messages [Suggest Edits](#)

Message Header

The top-level message header format is standard for all messages passing through CCTP:

Field	Bytes	Type	Index	Description
version	4	uint32	0	Version identifier
sourceDomain	4	uint32	4	Source domain id
destinationDomain	4	uint32	8	Destination domain id
nonce	8	uint64	12	Unique message nonce
sender	32	bytes32	20	Address of MessageTransmitter caller on source domain
recipient	32	bytes32	52	Address to handle message body on destination domain
destinationCaller	32	bytes32	84	Address permitted to call MessageTransmitter on destination domain, or bytes32(0) if message can be received by any address
messageBody	dynamic	bytes	116	Application-specific message to be handled by recipient

Why we use bytes32 type for addresses

CCTP is built to support EVM chains, which use 20 byte addresses, and non-EVM chains, many of which use 32 byte addresses. We provide a [Message.sol library](#) as a reference implementation for converting between address and bytes32 in Solidity.

Message Body

The message format includes a dynamically sized messageBody field, used for application-specific messages. For example, TokenMessenger defines a [BurnMessage](#) with data related to cross-chain transfers:

Field	Bytes	Type	Index	Description
version	4	uint32	0	Version identifier
burnToken	32	bytes32	4	Address of burned token on source domain
mintRecipient	32	bytes32	36	Address to receive minted tokens on destination domain
amount	32	uint256	68	Amount of burned tokens
messageSender	32	bytes32	100	Address of caller of depositForBurn (or depositForBurnWithCaller) on source domain

Message Nonce

A message nonce is a unique identifier for a message that can only be used once on the destination domain. The next available nonce on a source domain is an integer. On the destination domain, messages can be received in any order, and used nonces are stored as a hash of the source domain and nonce integer value. Updated 3 months ago * [Table of Contents](#)
* * [Message Header](#) * * [Message Body](#) * * [Message Nonce](#)