String

Must be a string parsable by the underlying uuid library's parse function. This typically means the standard 36-character hexadecimal string representation including hyphens (e.g., "123e4567-e89b-12d3-a456-426614174000").

Conforms to RFC 4122 which defines UUID structure and representation.

Bytes

Must be a Uint8Array containing exactly 16 bytes. These bytes represent the raw 128-bit binary value of the UUID. The constructor validates the length.

Represents the binary structure defined in RFC 4122.

UUID

Tag: 37
URType: cbor-uuid
Standards:
IANA CBOR Tag Registry: Tag 37
RFC 4122: Defines the UUID structure itself.

Request ID

Optional

UUID instance | string (standard format) | Uint8Array (16 bytes) | undefined (will be auto-generated)

A unique identifier for the request (RFC 4122 UUID). Encoded using the cbor-uuid UR type (Tag 37).

PublicKey

Bytes (e.g., Uint8Array, Buffer in JS/TS; []byte in Go).

Must be the compressed public key format (33 bytes long, typically starting with 0x02 or 0x03). Using the uncompressed format (65 bytes) will produce an incorrect fingerprint according to the standard.

The public key corresponding to the parent or master key whose fingerprint is needed. The derivation and format of public keys are defined by standards relevant to the elliptic curve being used (e.g., SEC standards for secp256k1) and BIP-32.

HASH160

The 33-byte compressed public key from step 1.

Applies two cryptographic hash functions sequentially: RIPEMD160(SHA256(compressed_public_key_bytes)).

A standard cryptographic hashing process combining SHA-256 and RIPEMD-160. Its use for creating key identifiers is specified in BIP-32.

First 4 Bytes

The 20-byte HASH160 result from step 2.
Select the first 4 bytes (bytes 0, 1, 2, 3) from the 20-byte hash
A 4-byte (32-bit) value. This is the fingerprint in its raw byte form
This selection process is the specific definition of the key fingerprint as defined in BIP-32.

Curve

Required

EllipticCurve (enum value, which resolves to a number like 1 for P256, 8 for secp256k1, 6 for Ed25519, etc.)

The numeric identifier for the cryptographic elliptic curve used by the blockchain or asset, as defined in the IANA COSE Elliptic Curves registry (RFC 9053).

Type

The integer identifier for the base blockchain or cryptocurrency, as defined in the SLIP-44 "Registered coin types for BIP-0044" registry (e.g., 0 for Bitcoin, 60 for Ethereum, 501 for Solana).

Sub Type

Optional

Array<number | string | Uint8Array> (An array containing one or more elements, where each element can be a number, a string, or a byte array).

Used for further specification beyond the base SLIP-44 type. Its meaning is convention-based but Often holds the EIP-155 Chain ID (as a number) for EVM-compatible chains (where type is typically 60). Example: [1] for Ethereum Mainnet.

Coin Identity

Tag: 41401
URType: coin-identity
Standard: Custom

Coin ID

CoinIdentity instance

Specifies the target blockchain and curve. Encoded using the coin-identity UR type (Tag 41401). Uses COSE curve identifiers and SLIP-44 coin types internally.

Path

string | PathComponent[]
The sequence of derivation steps.

The string format adheres to BIP-32 / SLIP-10 notation (e.g., m/44'/0'/0'/0/0) and also supports extensions like ranges (1-5'), wildcards (*), and pairs (<0;1>) defined by Bitcoin Core Output Descriptors and incorporated into BCR-2020-007.

Source Fingerprint

Optional

0xabcdef12
The 4-byte fingerprint identifying the ancestor key (parent or master) from which the path begins. It must be a positive integer.

- The concept and calculation method (first 4 bytes of HASH160 of compressed public key) are defined in BIP-32.
- Its specific use and field name within the keypath structure are defined in BCR-2020-007.

Depth

Optional

The number of derivation levels from the ultimate root master key (depth 0) to the key identified by sourceFingerprint.

- The concept is defined in BIP-32.
- Its specific use and field name within the keypath structure are defined in BCR-2020-007.

Keypath

Tag: 40304
URType: keypath
Standards:
BCR-2020-007
BIP-32 / SLIP-10:
BIPs 380-389

Derivation Path

Optional

Keypath instance | string (BIP-32 + Output Descriptor format) | undefined

The HD derivation path. Encoded using the keypath UR type (Tag 40304). Structure defined by BCR-2020-007, notation based on BIP-32/SLIP-10 + Output Descriptors.

SignData

Buffer | Uint8Array (must not be empty)

The raw binary data (transaction bytes, message hash, etc.) that needs to be signed. Represented as bytes in CBOR.

Origin

Optional

string | undefined

A simple text string indicating the source application/wallet. Represented as text in CBOR.

txType

Optional

number (integer) | undefined

An integer representing a blockchain-specific transaction type (e.g., EIP-2718 types for Ethereum). Represented as int in CBOR.

Address

Optional

string | Uint8Array | undefined

The sender address, provided as either a string (format depends on the chain) or raw bytes. Represented as string / bytes in CBOR.

SignRequest

Tag: 41411
URType: sign-request
Standard: Custom