IBC

Interconnected Blockchains

The core vision of Cosmos is to scale horizontally by having an ecosystem of interconnected application specific blockchains. Inter-Blockchain Communication Protocol (IBC) is responsible for interconnecting heterogeneous blockchains or in another way, relaying data packets between arbitrary state machines (i.e application blockchains).

It could also be defined as a generic and standard protocol implementation for transferring value between chains, not only limited to Cosmos chains as other blockchains with different consensus could support IBC to communicate with the Cosmos Ecosystem (Polkadot, Ethereum through a peg zone for example).

IBC provides a common protocol for blockchains to communicate in a standardized and trustless way.

There is a lot to talk about IBC in terms of current and future capabilities, and the technology layers, but that’s not the goal of this paragraph. We’ll go through the high level principles here.

The IBC stack

IBC is composed of two layers:

- Transport layer (IBC/TAO) - provides the infrastructure for establishing secure connections and data packets authentication between chains
- Application layer (IBC/APP) - defines how data packets should be packaged and interpreted by sending and receiving chains
-

This is wrapped up in a “light client” and relayers are external components for passing messages through blockchains:

Source: What is IBC? | Cosmos Developer Portal

Relayers

Relayers are off-chain actors ensuring the “physical” connection between two chains, scanning the state of the two interconnected chains, looking for an intention to send a transaction on a chain and then relaying the data and its commitment proof to the other chain.

For example in the case of fungible token transfer between two chains, the relayers are responsible for proving that your tokens are locked in Chain A and giving a representation (Voucher) on Chain B.

Relayers are using the light client of each blockchain to verify incoming messages on chain A and submit them (and the proof of commitment) on chain B. Chain A can’t send data directly to chain B and instead “commit” the hash of the data packet in its own state machine. That’s this specific state that relayers are monitoring to send this packet and its proof to the chain B.

Native security - no additional trust

The major difference of IBC compared to existing bridge solutions is that there are no third parties to trust, no multisig involved, for transferring value/messages between blockchains. This concept of IBC native security means that if you trust chain A and chain B then you also trust IBC-TokenA on chain B and vice versa. As long as relayers are operated by any party and channels/ports are open and authentication successful between blockchains, messages can be passed along.

An analogy of IBC could be seen as an internet application on a computer. A channel is an IP connection, with the IBC portID being an IP port and the IBC channelID being an IP address.

IBC security does not depend on third parties to verify the validity of transactions between blockchain. IBC security is mostly done by the light clients who verify proofs of commitments and the state of the two interconnected blockchain. In short terms, IBC security is based on::

- Trust in the chains you connect with
- Fault isolation mechanisms, to limit damage done if a chain is acting maliciously
-

IBC is then still Byzantine resistant thanks to the proof validation by the light client. If a relayer were to act maliciously, the packet would be rejected by the counterparty chain because the proof would be invalid (because light client and relayers are independent).

Interoperable messaging & contracts

Fungible token transfer is one example but IBC allows for Non-Fungible token transfer, multi-chain smart contracts and interchain accounts (interacting on a blockchain account from another source blockchain). This is the function enabling Secret Network to be the privacy hub for the Interchain. Other Cosmos chains can store and manipulate private data (even private keys) on Secret Network from the comfort of their own chain, Privacy as a service.

Finally, below a figure explaining the travel of an IBC packet between blockchains:

Last updated1 year ago On this page *Interconnected Blockchains * The IBC stack * Relayers * Native security - no additional trust * Interoperable messaging & contracts

Was this helpful? Edit on GitHub Export as PDF