ANUUTECH™

Private Chains

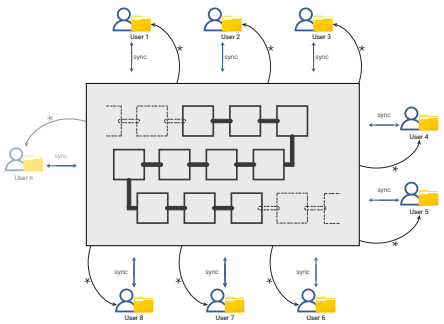
Private Chains

Table of Contents

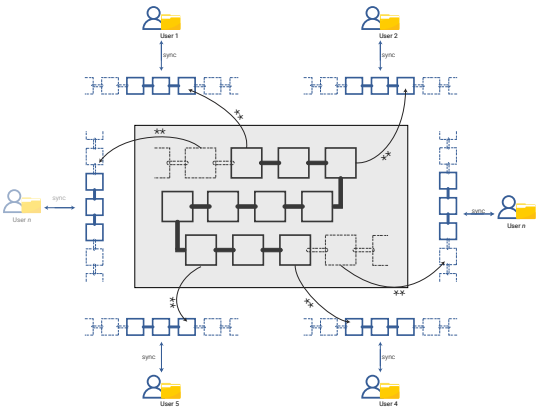
Premise

In classic blockchain networks there is one single "master" chain where everything is recorded and shared with all the users of the network. Not only does this require every user to constantly sync the entire chain and store all the transactions throughout the network which requires a vast amount of storage space, it also impacts the scalability and the network throughput. The use of private chains where every user creates his own chain that only contains the transactions directly related to the user's address(es) will eliminate a lot of the throughput and scalability issues currently found in "traditional" blockchains. Furthermore, it will provide additional privacy to users since transaction data is only shared between users and not the entire network.

BLOCKCHAIN NETWORKS



* All the transaction data that is recorded in the chain is shared with every user of the network



** Private Chains (in which every user creates his/her own chain) only contain transactions directly related to the user's address(es)

Classic Blockchain Networks (single chain)

- record all transactions
- share the data with users of the network
- users must constantly sync with the entire chain
- users must store all transactions that occur on the network

Limited network throughput and scalability

Vast amounts of storage space requirements

Private-Chains Networks (personalized chain)

- only contain transactions directly related to the user's address(es)
- transaction data is only shared between users (not the entire network)

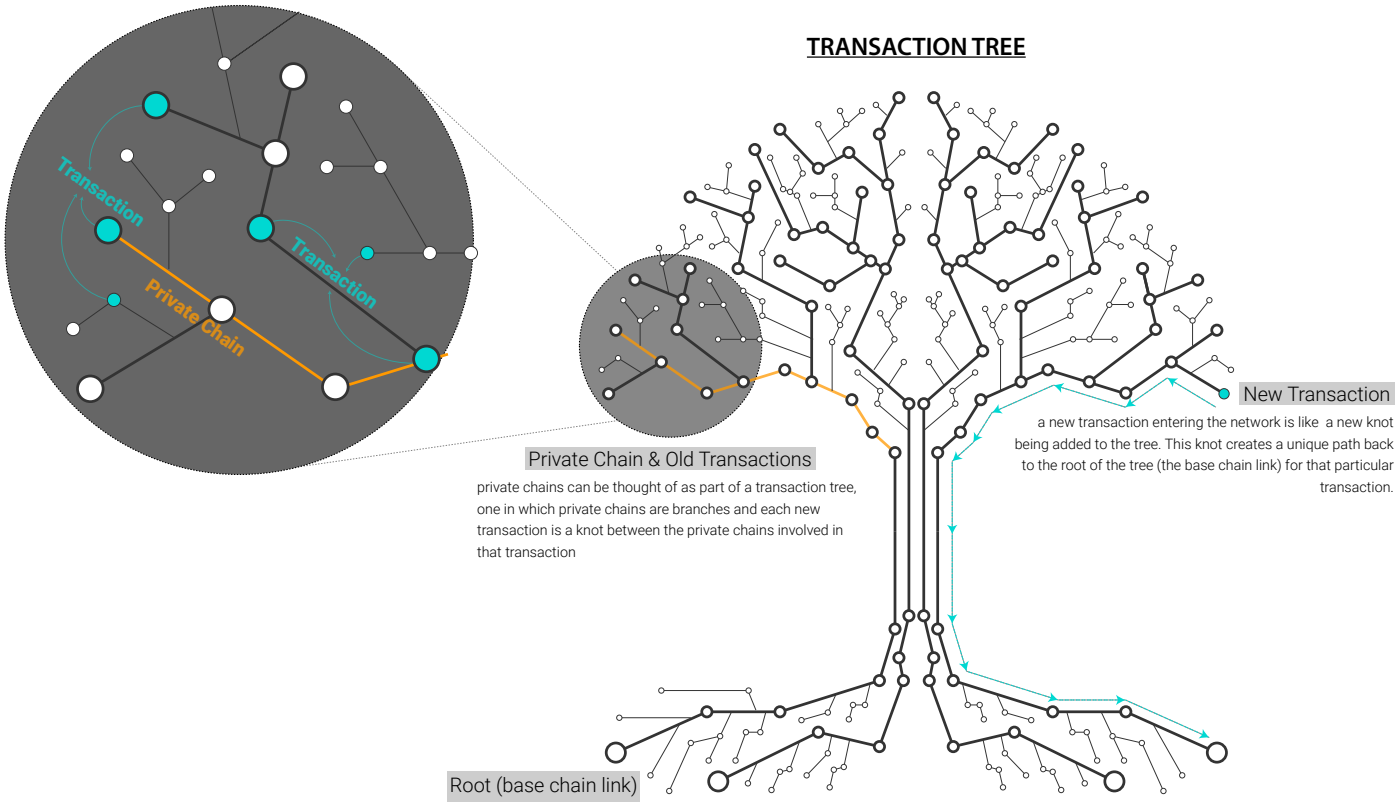
Elimination of the throughput and scalability issues currently found in "traditional" blockchains

Additional privacy to users

Principle

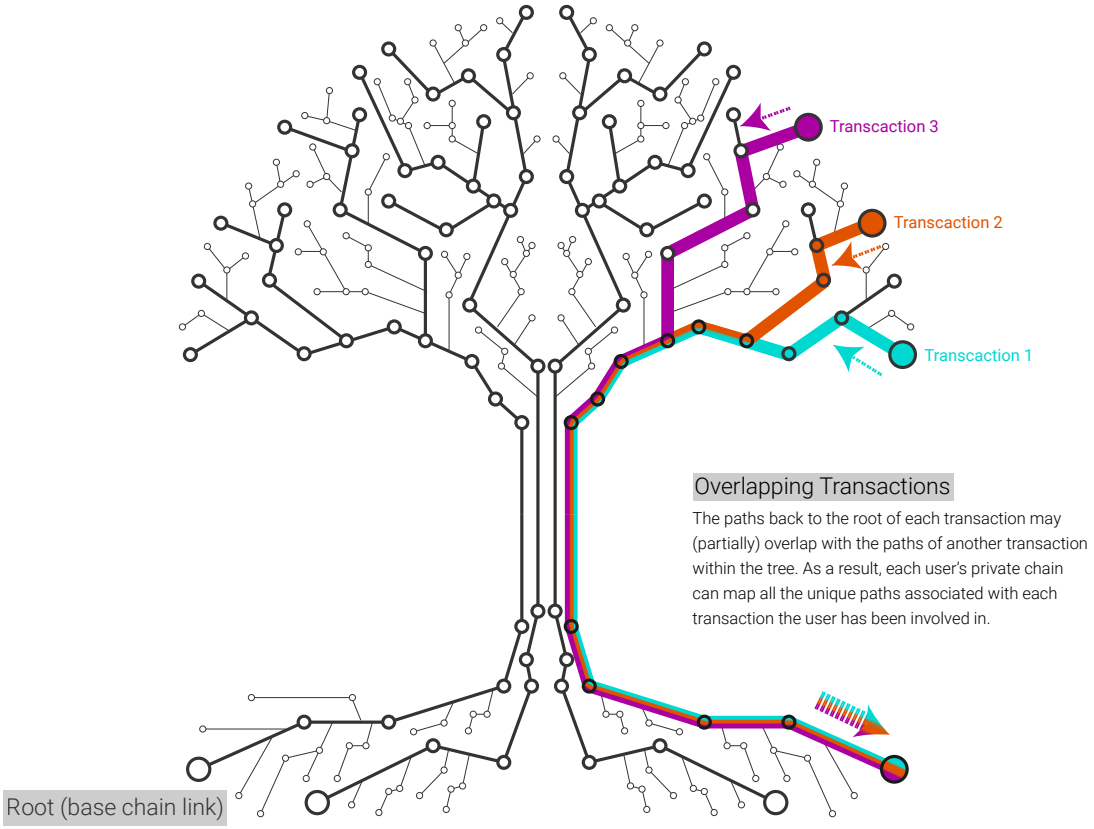
In a private chain network, every user is responsible for mining their own chain and providing the evidence for the origin of the data that is being transacted. The entire network consists of an unlimited number of private chains that interlink during each individual transaction.

The private chains are part of the transaction tree where the private chains are the branches and with each new transaction a knot between the private chains involved in that transaction gets added to the tree, creating a unique path back to the root of the tree (base chain link) for that particular transaction.

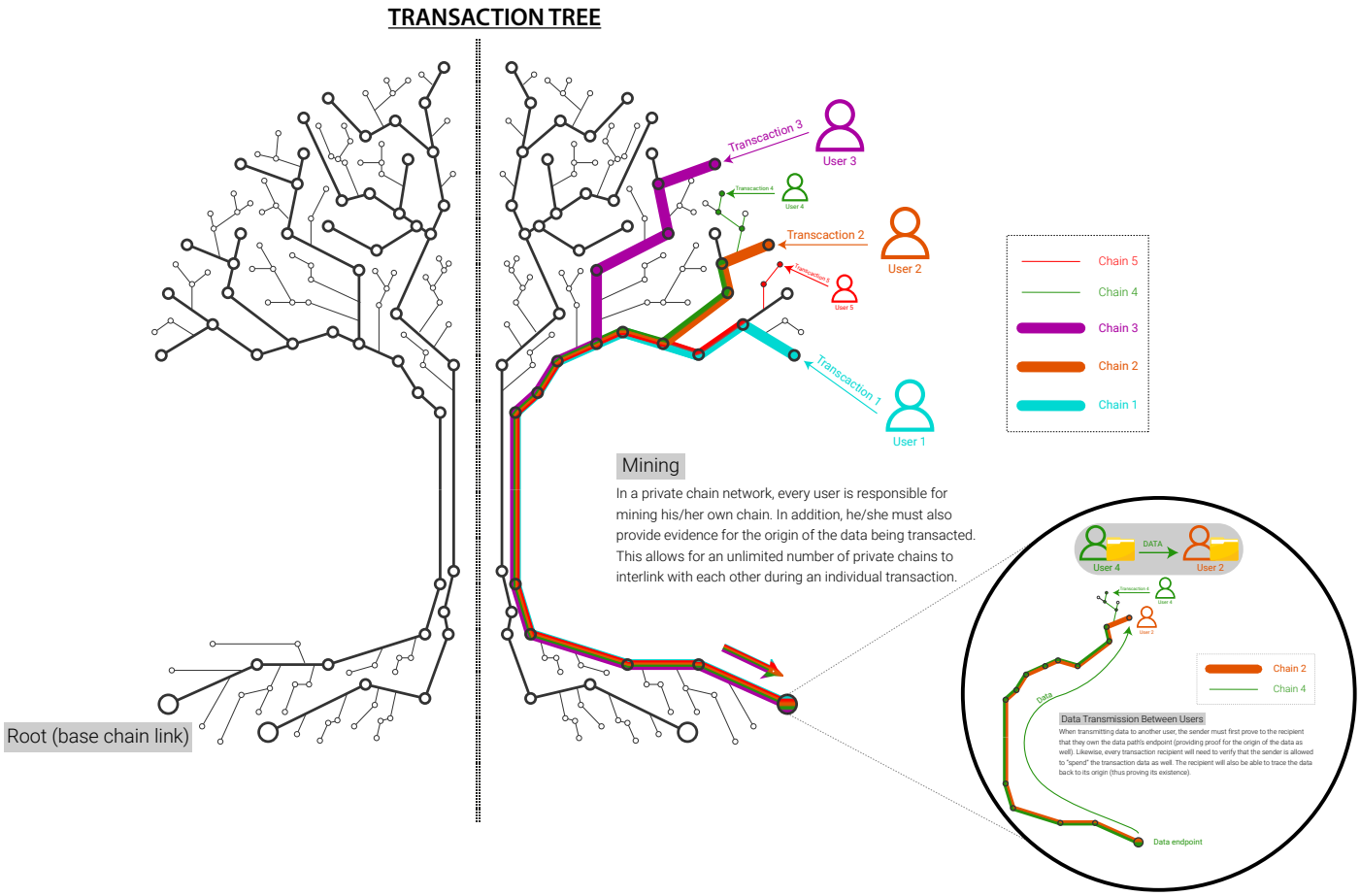


The paths back to the root for a transaction can (partially) overlap with the path of another transaction in the tree. A user's private chain is in fact a map of all the unique paths for each transaction the user was involved in.

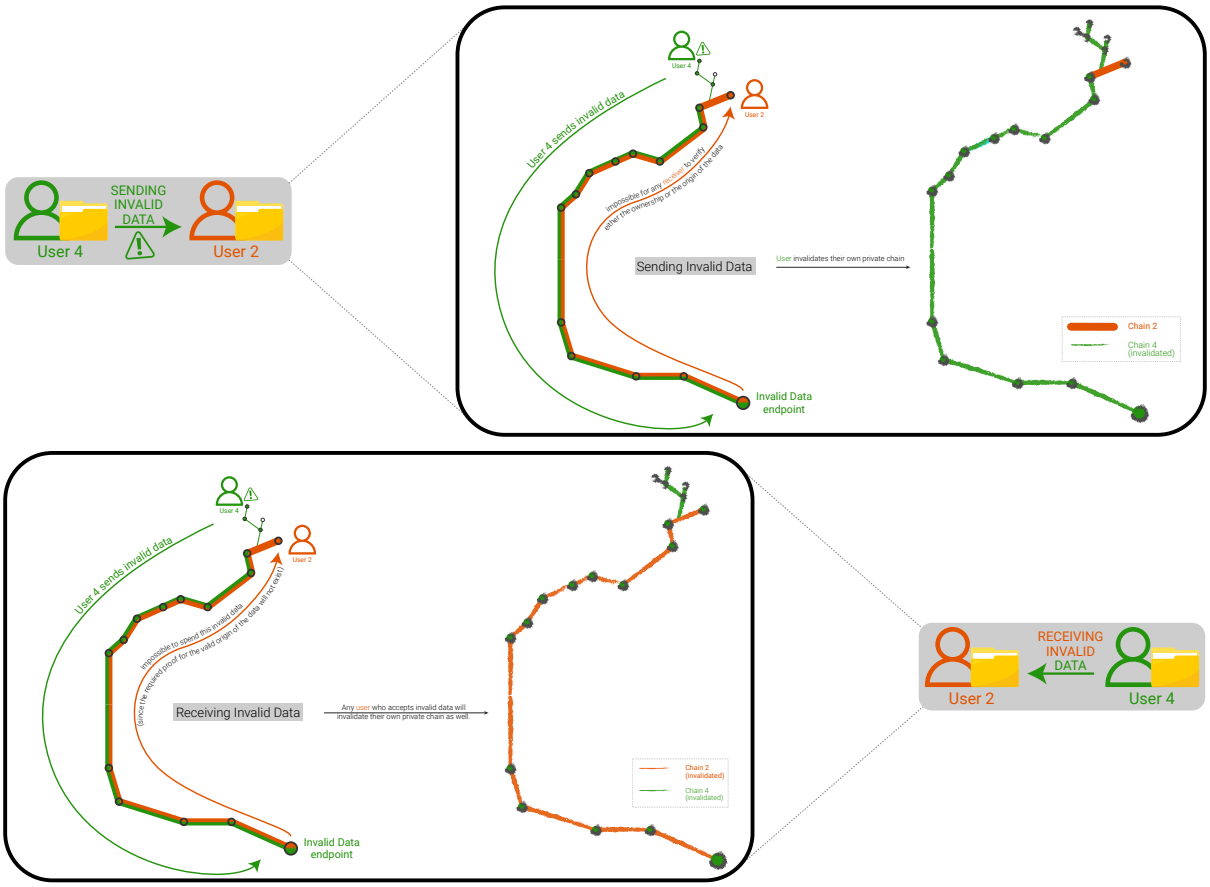
TRANSACTION TREE



When a user wants to transact data to another user, they will show the other user that they are in fact the owner of the data's path endpoint at that time of the transaction as well as provide the proof for the origin of the data. Meaning that every user receiving data in a transaction will be able to verify the sender is allowed to "spend" this data as well as trace the data back to its origin proving its existence.

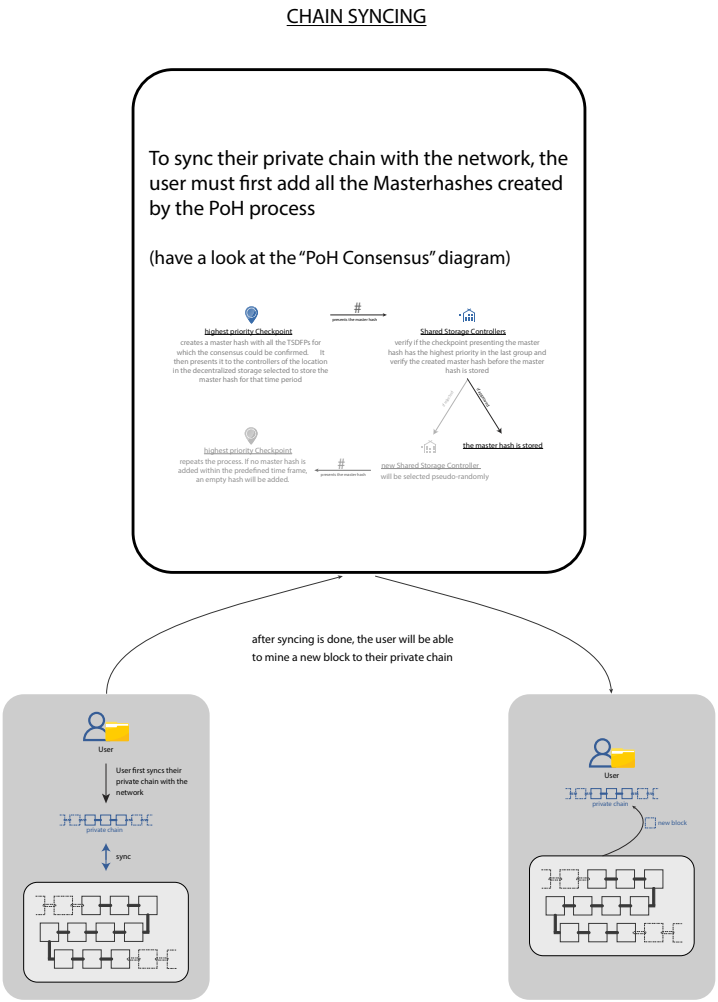


In a private chain network, the burden of proof lies solely with the person sending the data. The reliability and trust of the proof is provided by the network's consensus protocol. If a user tries to send invalid data, it will lead to invalidating that user's private chain since it will be impossible for any receiver to verify either the ownership or the origin of the data. Accepting invalid data in turn will also lead to invalidating one's private chain as it will be impossible to spend the invalid data since the required proof for the valid origin of the data will not exist.



Creating transactions

In order to create a transaction, the user needs to mine a new block to his private chain which includes the transaction.

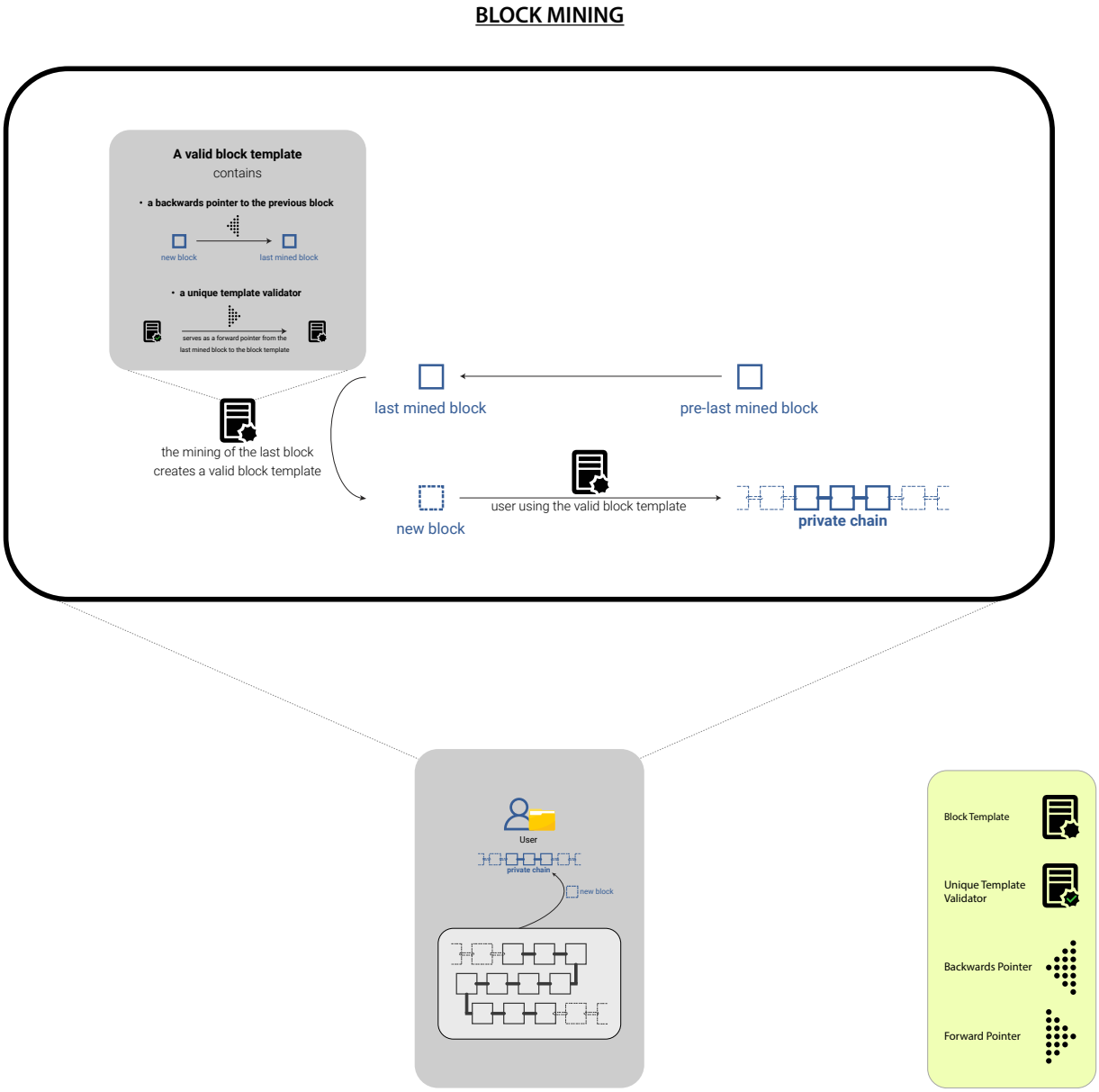


Chain syncing

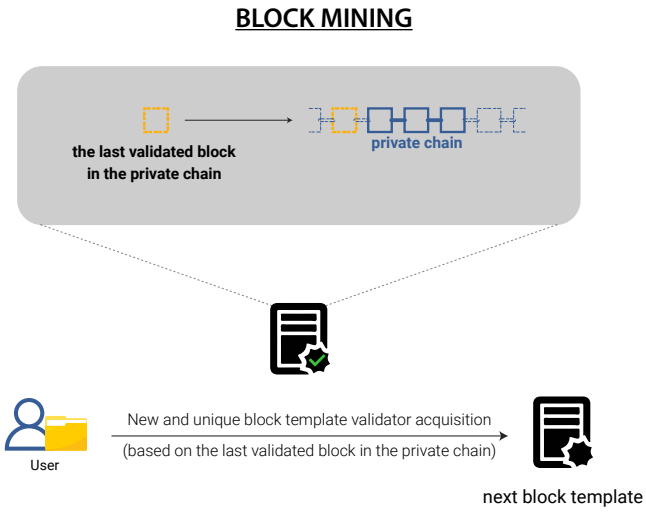
Before a user can mine a new block, the user needs to sync his private chain with the network. This requires the user to add all Masterhashes created by the PoH process since the last time the user was synchronized with the network. To get these Masterhashes the user will connect to the node network where the user can download the missing Masterhashes. When the user has synced his/her private chain, the user can now mine a new block to his/her chain. For comparison, a year's worth of Masterhashes will require +/- 16 MB compared to the dozens of GB in a traditional blockchain network. Not only will this make the syncing process exponentially faster than a classic blockchain network, it will also cut down significantly on the storage required to keep in sync with the rest of the network.

Block mining

To mine a new block to a private chain the user must use a valid block template that was created when the previous block was mined. A valid block template contains a backwards pointer to the previous block and a unique template validator that can be found in the previous block which serves as a forward pointer from the last block to the block template. The first phase of the block mining is to acquire a new unique block template validator for the next block template.

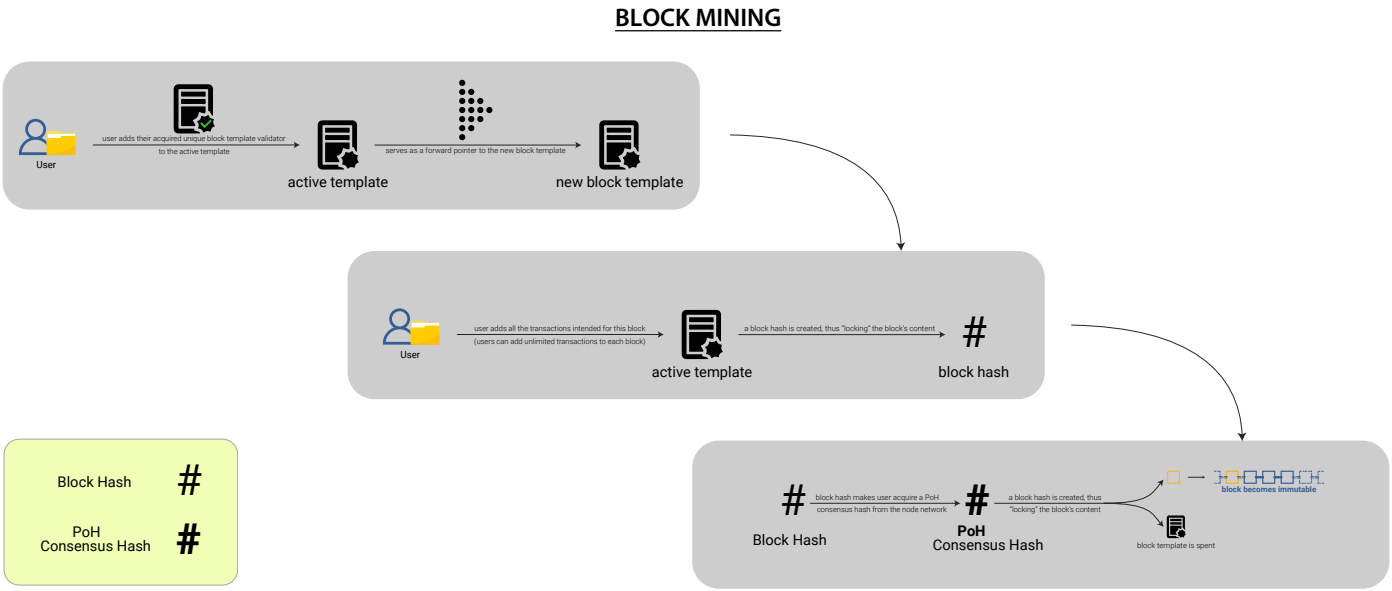


This new unique block template validator is generated by the node network based on the last validated block in the private chain. The last validated block in the private chain proves the user has paid the fee for a new block template validator and this block will be added to the private chain of the reward pool. By adding the block to the private chain of the reward pool it is impossible to get a second unique block template validator using this block.



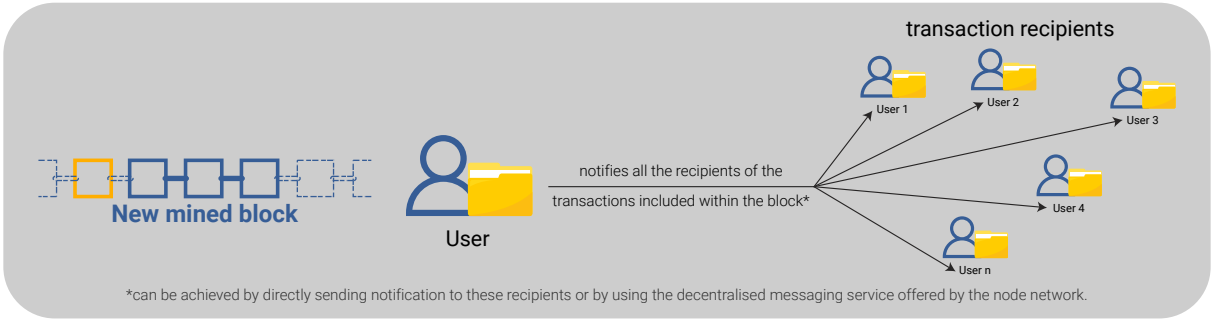
If the user did not pay the required fee for the previous block, the user will not receive a new unique block template validator and it will be impossible for the user to mine a new block to their private chain.

Next the user will add the acquired unique block template validator to the active template which will serve as a forward pointer to the new block template. The user will also add all the transactions intended for this block and create a block hash, "locking" the content of the block. It is important to note that a user can add an unlimited amount of transactions in each block and that the fee for 1 block is a fixed amount regardless of the number of transactions included in the block. With this newly created block hash, the user will acquire a PoH consensus hash from the node network. Once the PoH consensus hash has been acquired and the Masterhash containing this PoH consensus hash has been generated by the node network, the new block is unchangeable, and the block template is spent. By creating a new block, a new block template has been generated as well containing the blockhash of the newly created block which serves as a backwards pointer and the newly created unique block template validator which was included in the newly created block and serves as a forward pointer from the newly created block to the new block template.



Sending transaction

Once the user has successfully mined a new block, the user can now send this notification of this new block to all the recipients of the transactions included in this block. This can be done by either directly sending notification to these recipients or by using the decentralized messaging service offered by the node network.



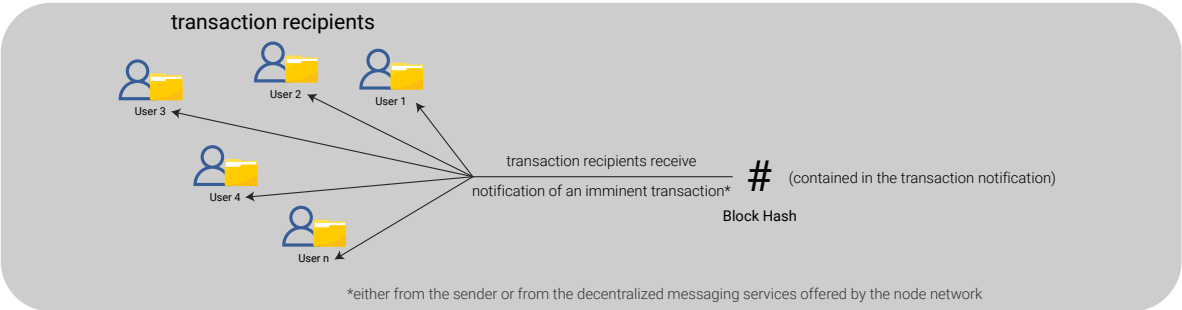
Block seeding

Since every user is required to provide the necessary proof for the origin of the data used in the transactions the user has the option to either provide this information directly to the recipients of the data in the transactions or by using the decentralized block seeding service offered by the node network.

Accepting transactions

Receiving transaction notification, the user will receive a notification of a transaction in which they are the receiver either directly from the sender or from the decentralized messaging services offered by the node network. This notification will contain the blockhash of the block which includes the transaction.

RECEIVING TRANSACTION NOTIFICATIONS



Chain syncing

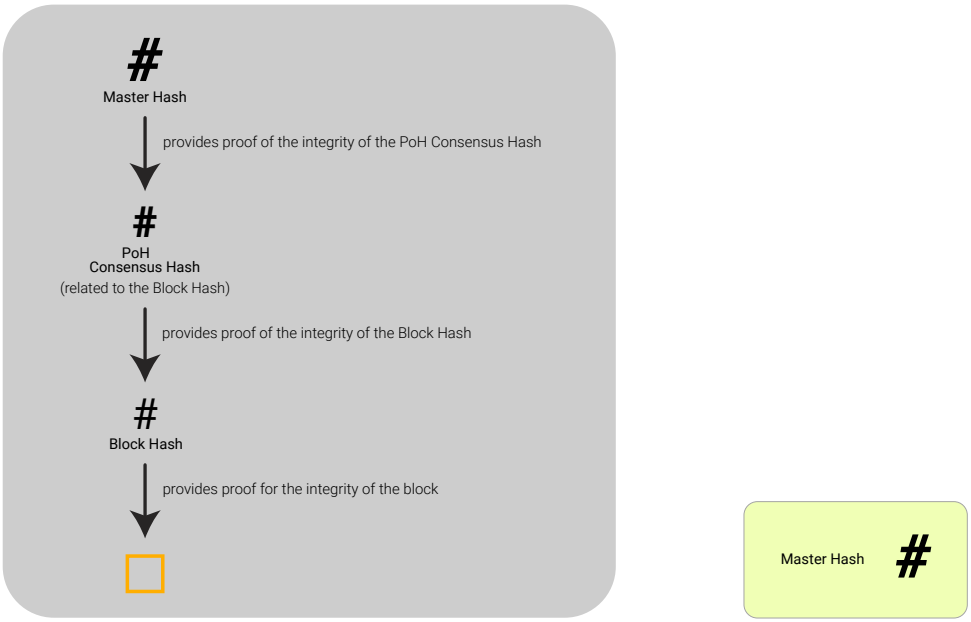
Before a user can accept a transaction, they first need to sync their private chain with the network by downloading all the new Masterhashes that were generated since the last time the user was synchronized with the network.

Block validation

PoH validation

Before validating the transaction, the user will first verify the integrity of the block containing the transaction. While blockhash will provide proof for the integrity of the block itself, the PoH consensus hash related to the blockhash will provide proof of the integrity of the blockhash. The Masterhash created for the time period in which the PoH consensus hash was created will provide proof of the integrity of the PoH consensus hash itself.

PoH VALIDATION



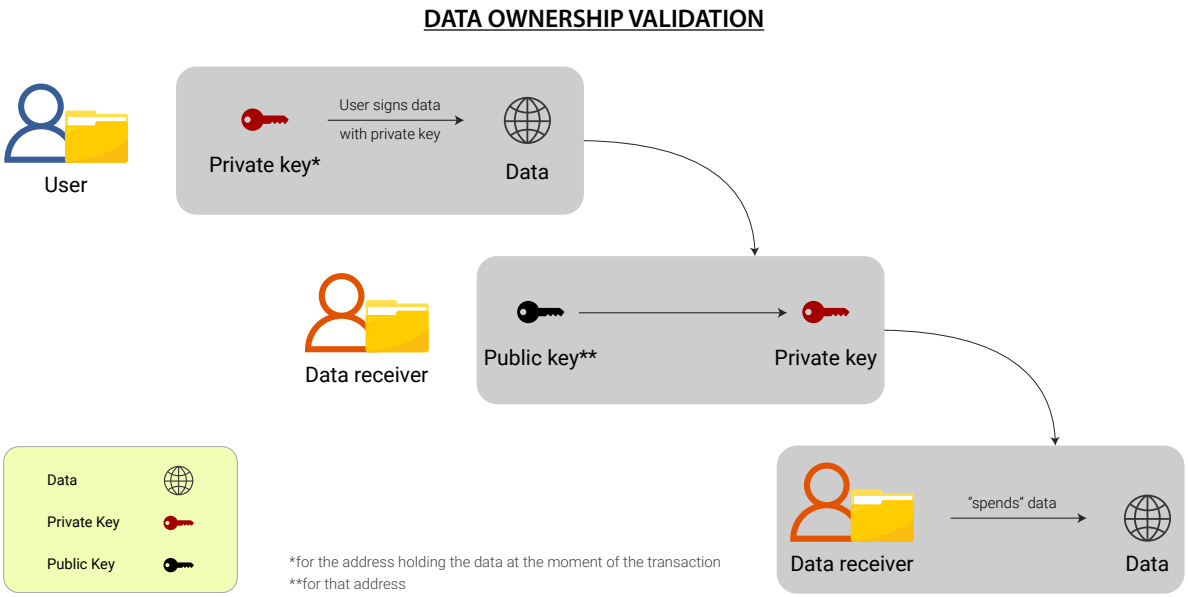
Chain integrity validation

Secondly, the user will verify the chain integrity of the sender's private chain ensuring that the transaction is included in a valid block of the sender's private chain. The user can use both forward and backward pointers included in the blocks to verify the chain integrity of the sender's private chain. The user only needs to check the chain integrity of the chain from the block containing the transaction to the block that included the previous transaction received from the same sender.

Transaction validation

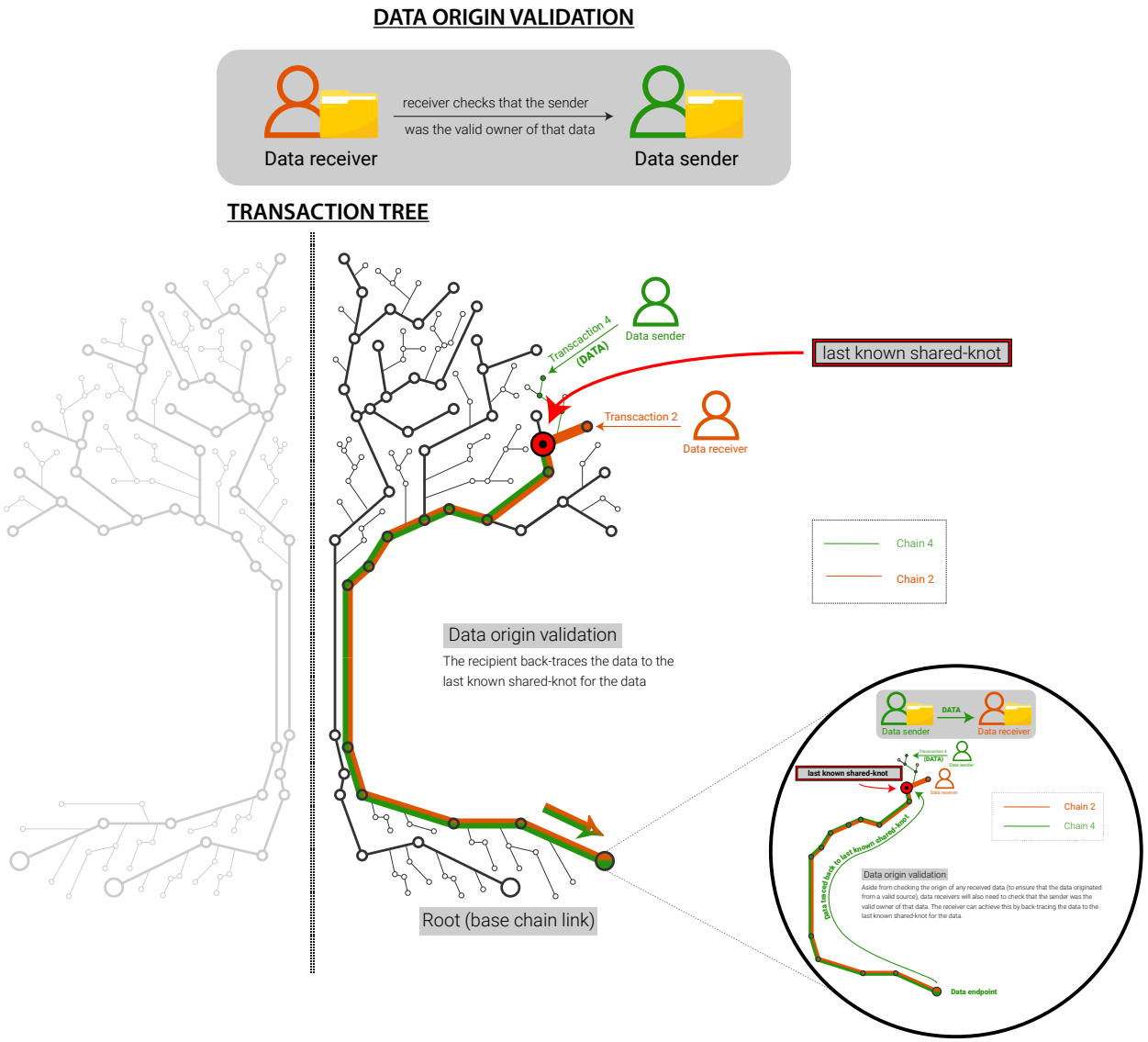
Data ownership validation

In order for someone to be able to "spend" data, they need to sign it with the private key for the address holding the data at the moment of the transaction. The receiver of the data can verify with the public key for that address if the data was signed with the correct private key.

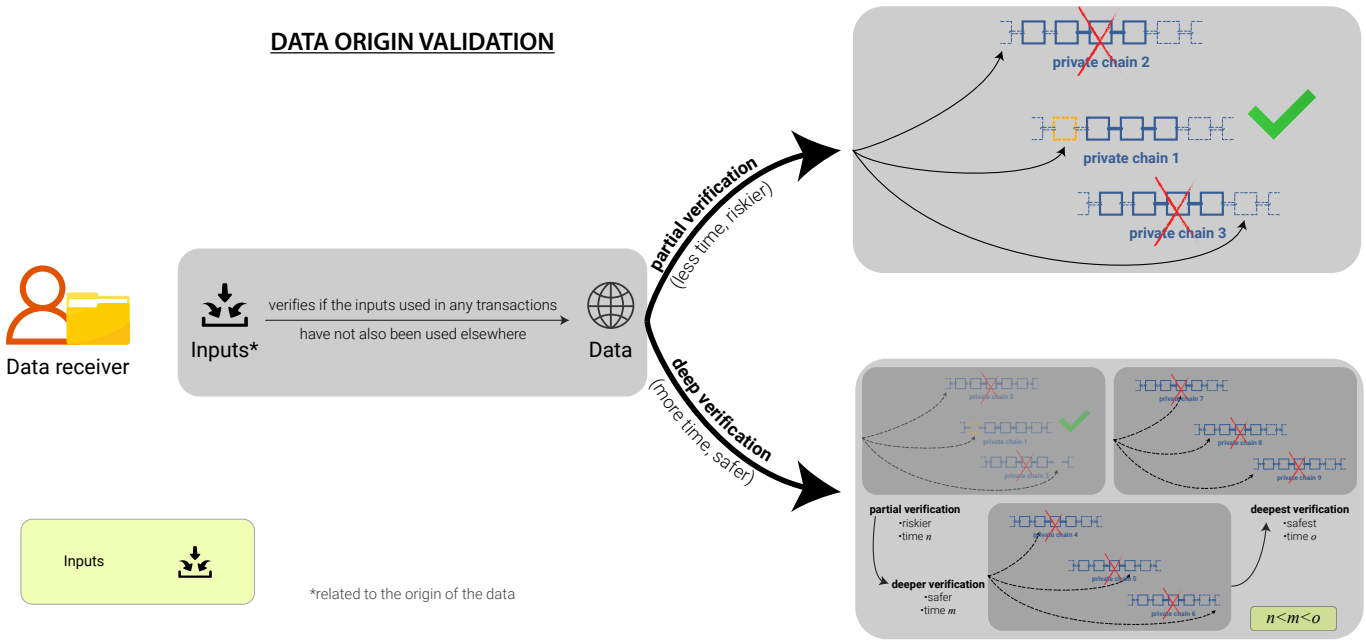


Data origin validation

A receiver of data will also check the origin of the data to make sure that the data originated from a valid source and that the sender was the valid owner of the data. Therefore, the receiver will need to back trace the data to the last known shared knot for the data. If there is no shared knot for the data, then the origin needs to be traced back all the way to the root of the transaction tree.

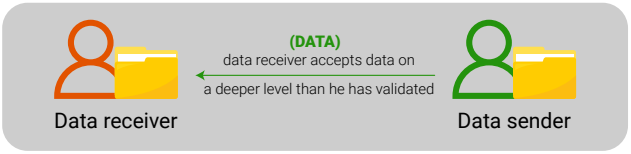


The receiver will trace back the data using the transactions related to the origin of the data and verify if the inputs and outputs of the connected transactions found in the traceback are valid. In order to check if the coins are not the result of a double spending, the receiver would verify if the inputs used in the transactions related to the origin of the data have not been used in more than one block of the chains involved in the various transactions related to the origin of the data.

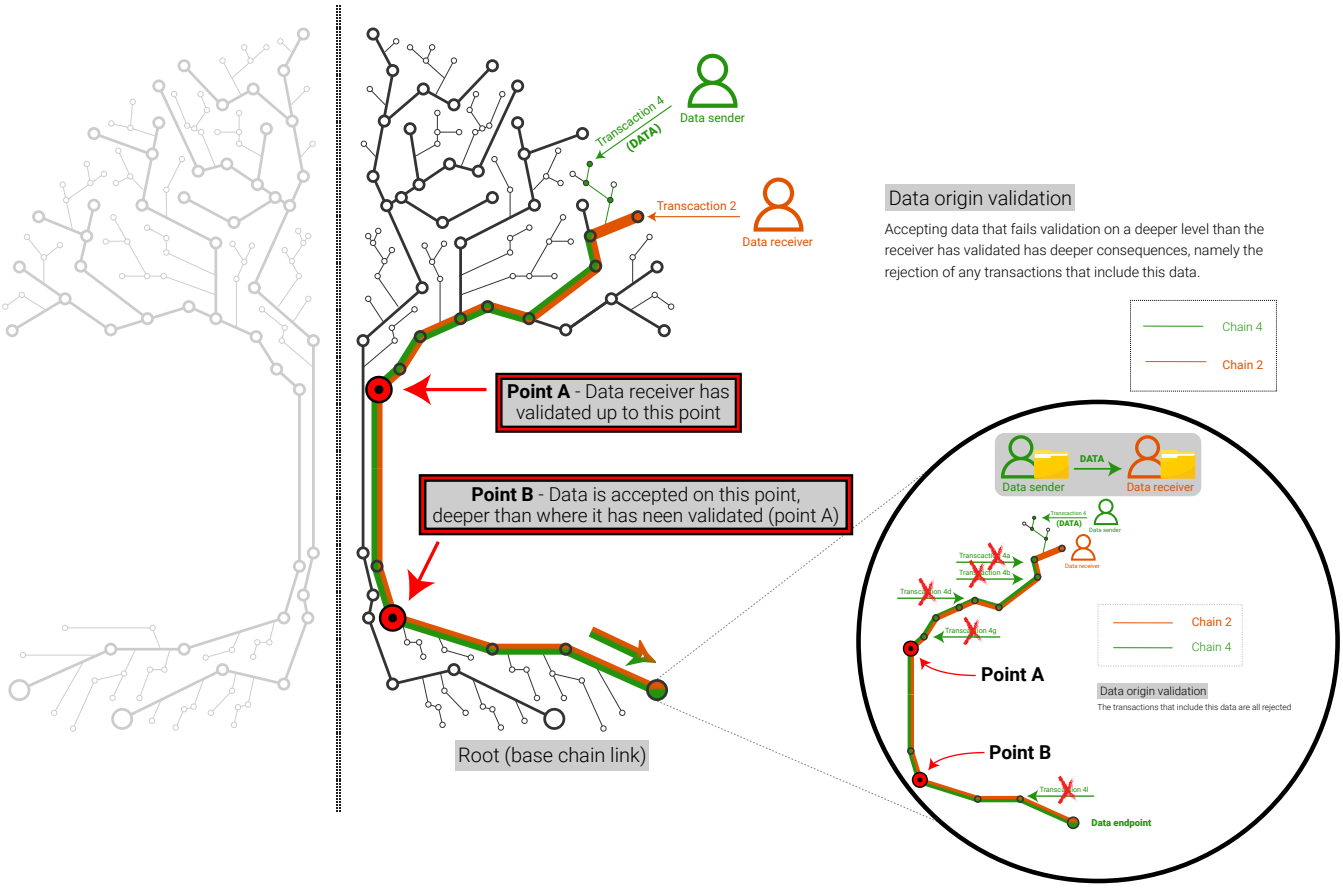


It is up to the validator to choose how far back he/she will do this verification. The receiver can base this choice on how trustable he believes the sender is. The deeper the verification is done; the more time is required to execute the verification. However, it is advised to always perform a complete validation of the entire data origin. Accepting data that fails validation on a deeper level than the receiver has validated will result in the rejection of transactions that include this data by receivers that will verify the entire origin of the data.

DATA ORIGIN VALIDATION

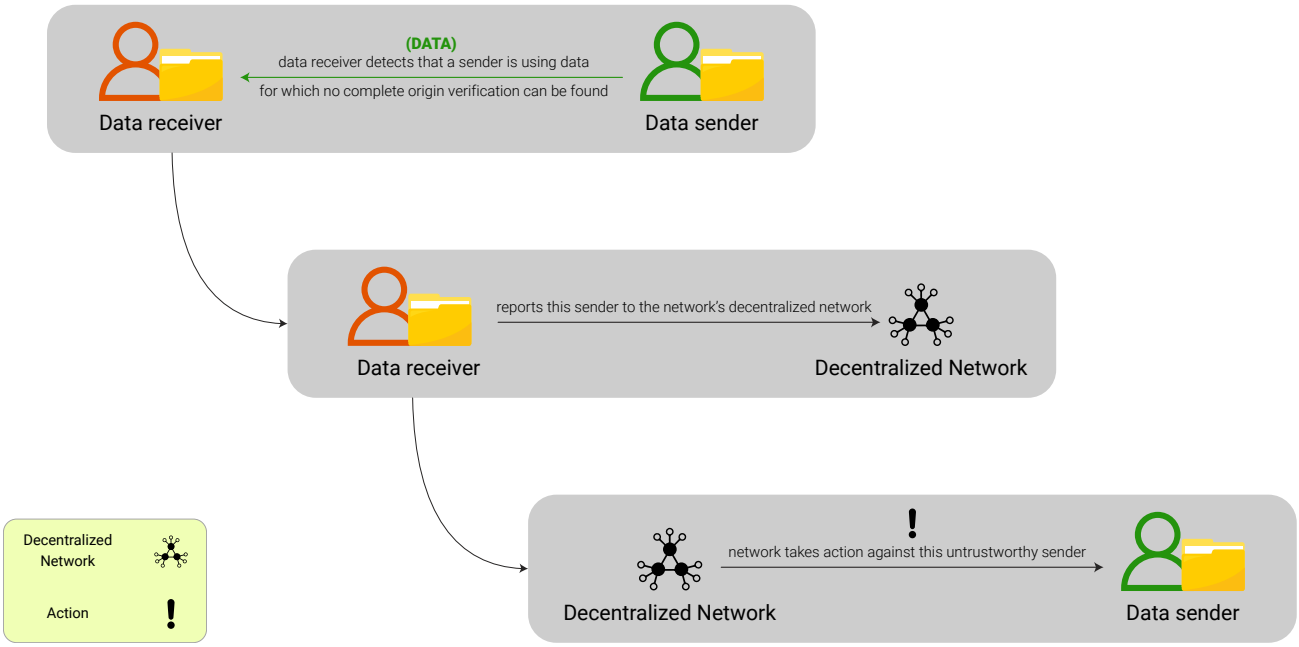


TRANSACTION TREE



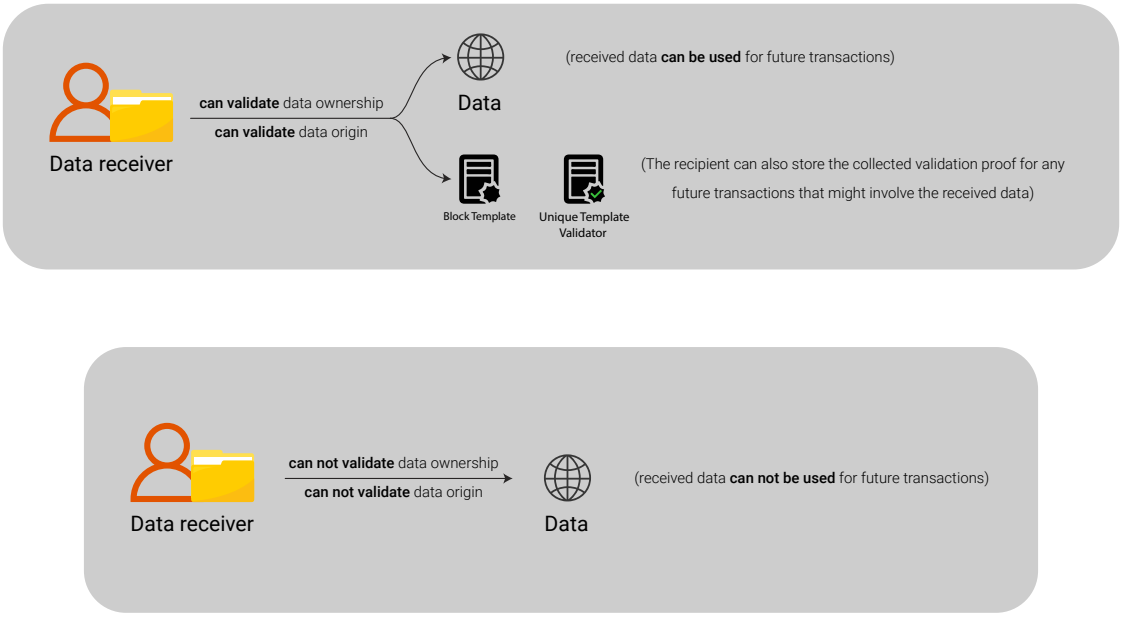
When a receiver detects that a sender is using data for which no complete origin verification can be done, they will report this sender to the network's decentralized network, which can then take action towards the untrustable sender. If the receiver can validate the data ownership and can validate the data origin the recipient can use the received data for future transactions. The receiver will also store the collected validation proof to provide proof for future transactions with the received data.

DATA ORIGIN VALIDATION



If the receiver can not validate the data ownership and/or the data origin it will be impossible for the receiver to use this data in a future transaction since they will be unable to provide the necessary proof of ownership and origin of the data. As long as the receiver does not use the received data in a new transaction, the received data will not be visible on the chain of the receiver.

DATA ORIGIN VALIDATION



Glossary

Block Mining:	Bitcoin miners receive Bitcoin as a reward for completing "blocks" of verified transactions which are added to the blockchain.
Block Template Validator:	Is generated by the node network based on the last validated block in the private chain.
Blockhash:	The primary identifier of a block is its cryptographic hash, a digital fingerprint, made by hashing the block header twice through the SHA256 algorithm. The resulting 32-byte hash is called the block hash but is more accurately the block header hash, because only the block header is used to compute it.
Classic Blockchain:	A blockchain where a single "master" chain records all the transactions and shares the date with users in the Network.
Decentralized Messaging:	No centralized server or single points of failure that can be compromised or shut down.
Masterhash:	The Master hash is the result of running true all phases of the PoH consensus before being stored.
PoH:	Proof of Hash, it is a network consensus protocol that provides trustable unalterable proof of the existence of a data subset at a given moment in time.
Private Chains:	Only contain transactions directly related to the user's address(es).
Reward Pool:	Miners receive coins out of the reward pool for working in the network.
Shared Knot:	Shared piece by two clients in a private network.
Transaction Tree:	Running different tasks that have no transactional connection. Each branch is its own transaction tree without any connection to other branches.
Trustability of Proof:	To the extent you can trust the proof that is given.