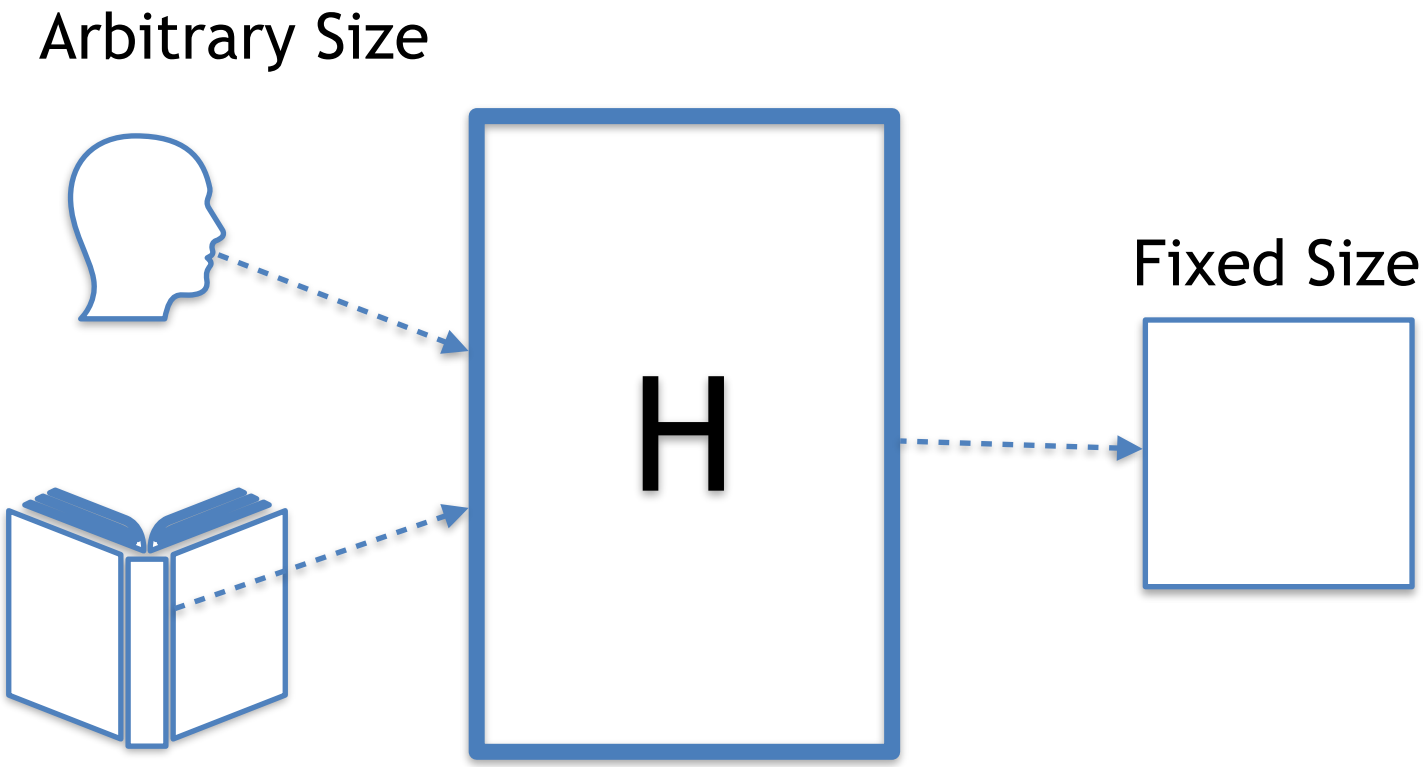
Introduction to Cryptography for Cryptocurrency

Hash Functions and Public/Private Keys



A hash function, H , maps input strings of arbitrary size to outputs of fixed size.

Collisions in Hash Functions

For a hash function H, inputs x, y are said to *collide* if

$$H(x) = H(y)$$

Design an algorithm to find collisions given H.

Brute-force way to find collisions.

Suppose H returns a 4-bit number.

Let D be an array of size 16 whose elements are all null.

Pick a random input x.
if D[H(x)] is null then set D[H(x)] = H(x)
else collision has been found between H(x) and D[H(x)]

Worse case time to find a collision is 17

Collisions in Hash Functions

For a hash function H, inputs x, y are said to *collide* if

$$H(x) = H(y)$$

Collision Resistant Hash Function

is a hash where finding collisions is intractable

If H returns an N-bit number then the worse-case time to find a collision is ?

Finding a collision is possible but is intractable when N = 256 if H is a collision resistant hash function.

Trusted Third Party Commitment

X predicts an outcome of a game, e.g. World Cup

The prediction is given to trusted third party.

You can't see her prediction. *(Information hiding)*

X can't change her prediction. *(Binding)*

After the game is over, the third party reveals X's prediction.

You can check if her prediction is accurate.

Trusted Third Party Commitment using Hashes

X predicts an outcome W and gives you a hash function H and a value y where y = H(W)

You can't find W given y, H. *(Or can you?)*

After the game is over, she reveals W.

You can check if her prediction is accurate. Is y = H(W)?

Trusted Third Party Commitment using Hashes

X predicts an outcome W of the World Cup and gives you a hash function H and a value y where y = H(W)

Can you find W?

Information Hiding

If there are N possible outcomes where N is not too large you can go through outcome V and determine whether

$$y = H(V)$$

So not really information hiding.

Information Hiding

X selects secret value r and gives you y = H(W + r)

Can you guess y given W (but not r)?

Information Hiding

X selects secret value r and gives you y = H(W + r)

If r is from a spread-out distribution, then given y guessing r so that y = H(W + r) is intractable.

Binding Commitment

X selects secret value r and gives you y = H(W + r)

X predicts W wins World Cup.

After game is over, if W' won then X changes W to W'

Can X change W to W'?

Binding Commitment

X selects secret value r and gives you y = H(W + r)

Can X change r and W later to r', W' so that
H(W+r) = H(W'+r')?

A hash function is a *binding hash function* if finding different pairs (x, y) and (x', y') such that
H(x + y) = H(x' + y') is intractable

How to use Hash Functions for Commitment

X selects secret value r and gives you y = H(W + r)

Use a *binding hash function*.

Puzzle Friendly Hash Function

H(r + x) = y, where r is a secret as in information hiding
Output of H is an n-bit string

Puzzle: given r and a set Y of n-bit strings find any x such that H(r + x) is an element of set Y.

What is the relation between the expected time to solve the puzzle and the size of set Y?

Puzzle Friendly Hash Function

Finding x such that H(r + x) is in set Y is proportional to the cardinality of Y.

You can control the difficulty of the puzzle by the size of Y.

Given a hash function H which maps strings of length N + M to strings of length M

define a hash function which maps strings of arbitrary length to strings of length M

Sending Messages Securely

How could you send a secret message to a friend ensuring that when your friend received the message the friend knows that you sent the message and that the message wasn't corrupted and wasn't read by anybody else?

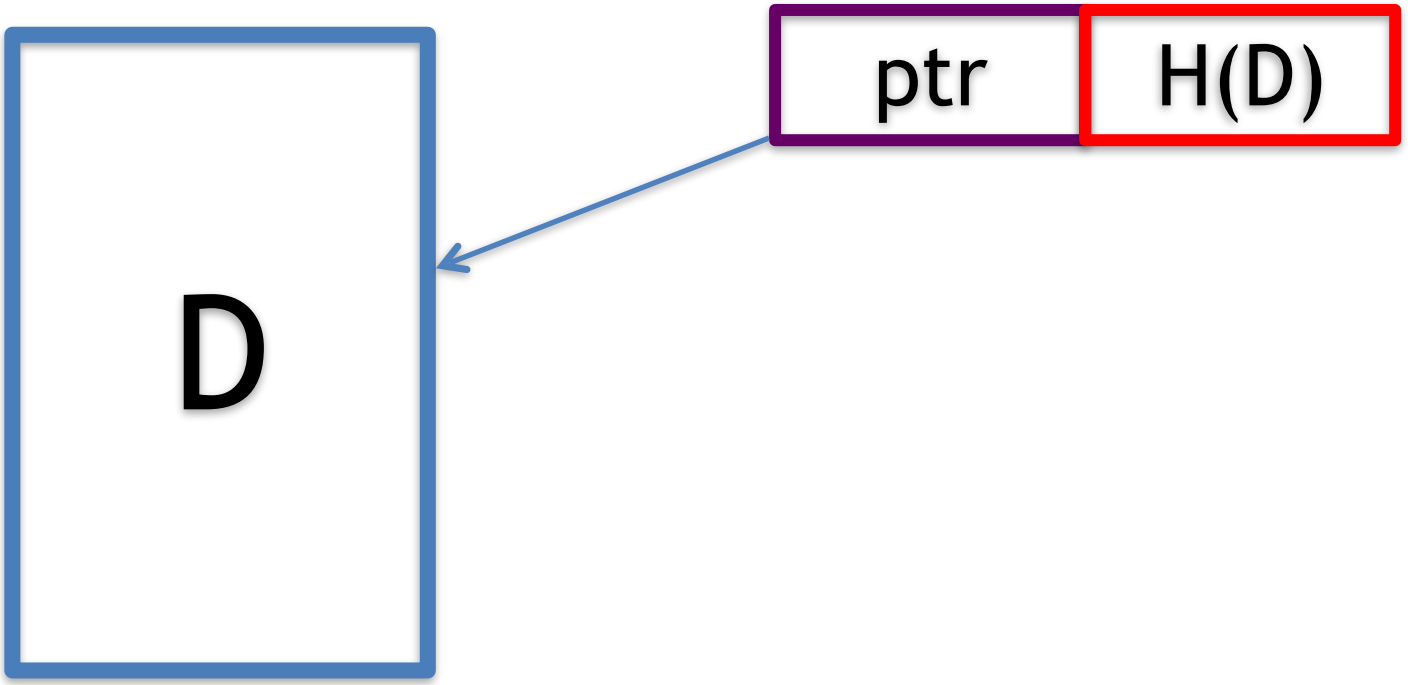
Kamala wants to send a secret message M to Joe so that when Joe receives it, Joe is certain that Kamala sent M, and nobody else read the message.

Sending messages securely

1. Kamala encrypts M with her private key to get M'.
2. She encrypts the pair (M, M') with Joe's public key.
3. Joe decrypts the message with Joe's private key to get (M, M')
4. Joe decrypts M' with Kamala's public key to get M''.
5. If M'' = M then Joe knows that Kamala sent the message.

A hash pointer to an item D is a pair (ptr, H(D)) where ptr points to the location of D and H is a cryptographic hash function.

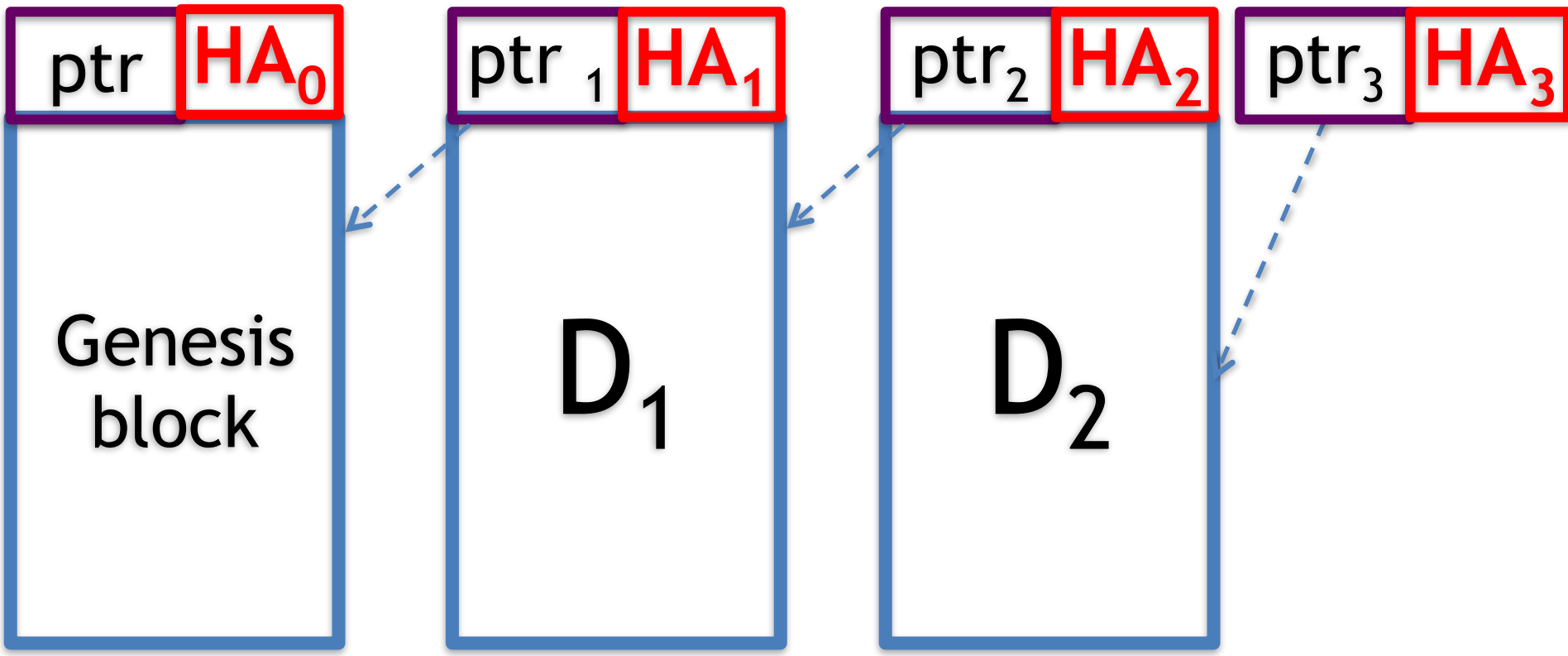
A **cryptographic hash function** is one which is collision resistant, hiding, and puzzle friendly.



Tamper-resistant data

If D is modified the hash pointer will not matc

D



Tamper-resistant data

If ptr $_1$ or HA_1 or D_1 are modified then the composite block [ptr $_1$; HA_1 ; D_1] is modified and so the composite block won't match HA_2

Cryptocurrency managed by a trusted bank

The bank maintains a tamper-evident ledger which is the foundation of the currency.

Cryptocurrency managed by a trusted bank

The bank maintains a tamper-evident ledger which is the foundation of the currency.

The ledger keeps track of transactions which are either
- **create** transactions or
- **pay** transactions

Create: Bank creates coins which the bank gives to agents.
Payer: Bank; *Payee*: agents

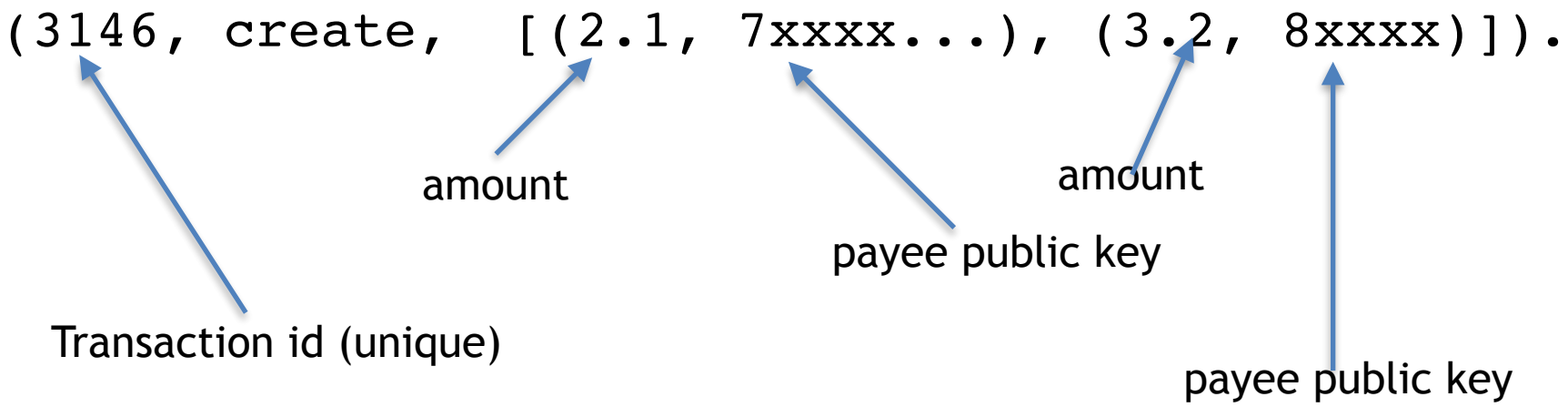
Pay: Agents (payers) give coins that they possess to other agents (payees).

Cryptocurrency managed by a trusted bank

Agents identified by their public keys.

Each transaction is signed securely by all payers of the transaction.

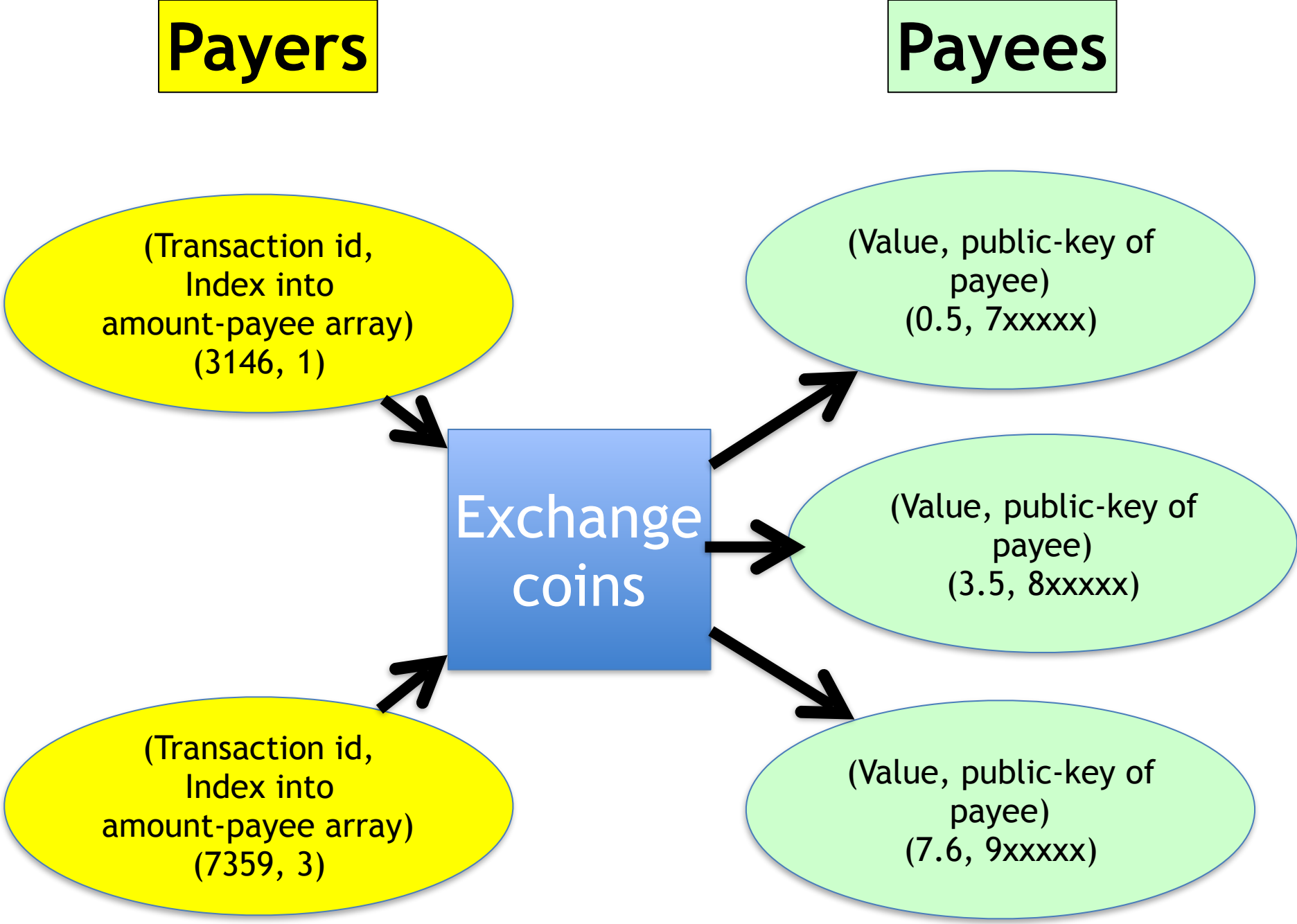
Example of a Create Transaction



(transaction id,
index into array of value-payee pairs)

uniquely identifies coins given to a payee in a transaction.

e.g (3146, 0) says that payee 7xxxx received 2.1 coins in transaction 3146


Payers
Payees
(Transaction id,
Index into
amount-payee array)
(3146, 1)
(Transaction id,
Index into
amount-payee array)
(7359, 3)
Exchange
coins
(Value, public-key of
payee)
(0.5, 7xxxxx)
(Value, public-key of
payee)
(3.5, 8xxxxx)
(Value, public-key of
payee)
(7.6, 9xxxxx)

You got 2 coins in a transaction 3145, and 3 coins in transaction 2193 but you want to pay 4 coins to somebody. What element goes into the ledger?

How does the system prevent double spending?

Why can't the agent with public key 7xxxx use the 2.1 coins that it received to buy items from Amazon and later use the same 2.1 coins to buy more items from Walmart?

The bank checks transaction validity

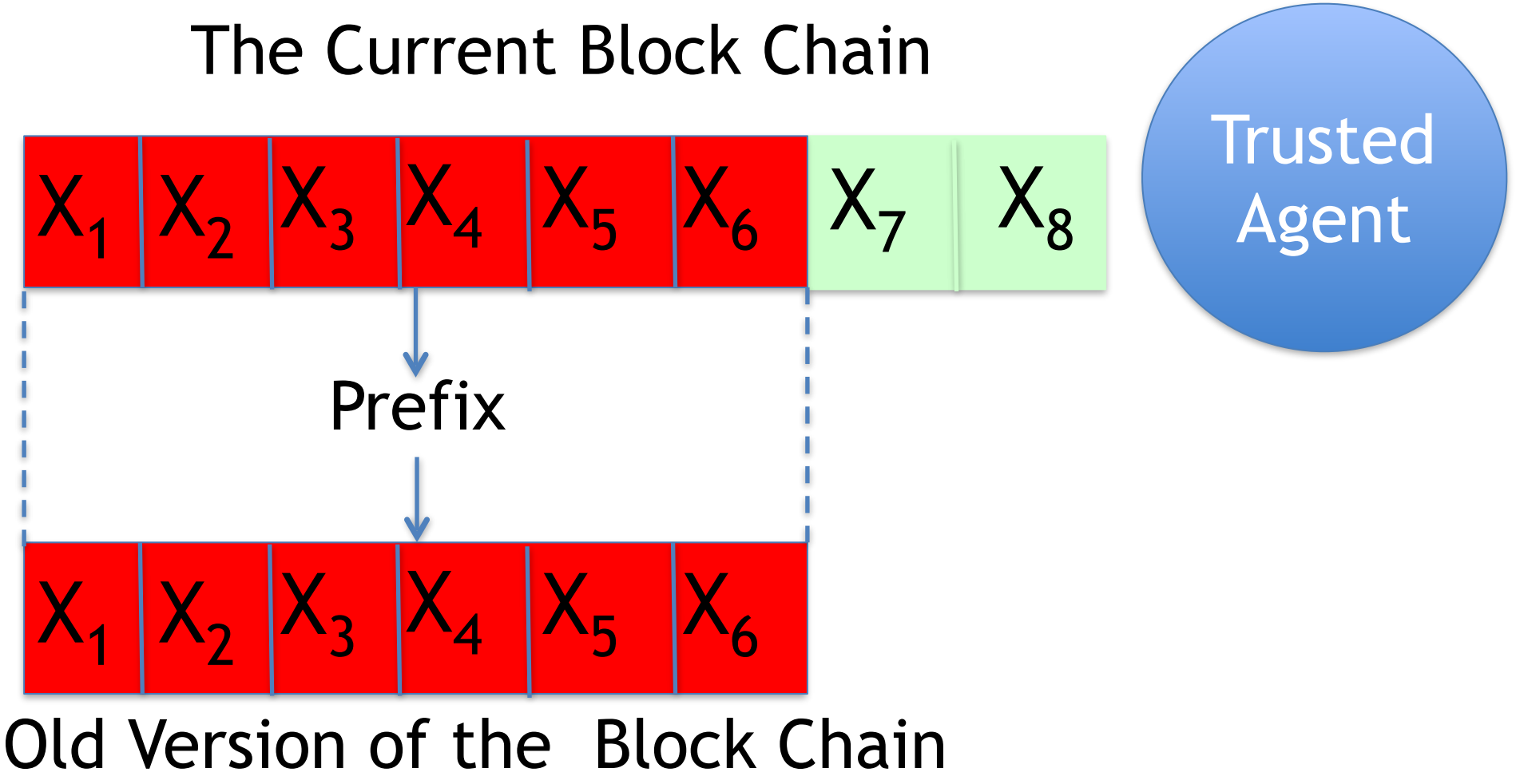
- checks signatures of payers
- total coins into and out of transaction
- checks that coins are genuine, i.e. were obtained in previous transactions and were not already spent.

Each transaction appended to the ledger is verified by the bank and signed by the bank.

The tamper-evident ledger grows as transactions are appended to its tail.

Agents keep copies of the ledger, and each agent can inspect the ledger to verify that double spending hasn't occurred.

But can the bank modify an existing ledger?



The old version must be a prefix of the current version

The bank maintains the master copy of the ledger

An agent's copy may differ from the master copy in only one way:
the agent's copy may be a prefix of the master copy.

Block chain

A tamper-evident ledger in which elements are blocks of many transactions (each of which is validated by the bank) is more efficient than a ledger in which each element is a single transaction.

Advantages and disadvantages of this cryptocurrency?

Advantages and disadvantages of this cryptocurrency

Advantages

- Each agent can verify each transaction in its ledger copy.
- An agent’s copy is a prefix of the bank’s master copy.
- Payers sign a transaction and nobody (not even the bank) can forge the signature.
- Agents are anonymous

Advantages and disadvantages of this cryptocurrency

Advantages

- Each agent can verify each transaction in its ledger copy.
- An agent's copy is a prefix of the bank's master copy.
- Payers sign a transaction and nobody (not even the bank) can forge the signature.
- Agents are anonymous

Disadvantages

- Single point of failure
- Agents may not trust the bank, e.g, denial of service.
- The bank can create arbitrary numbers of coins.