Bitcoin Consensus

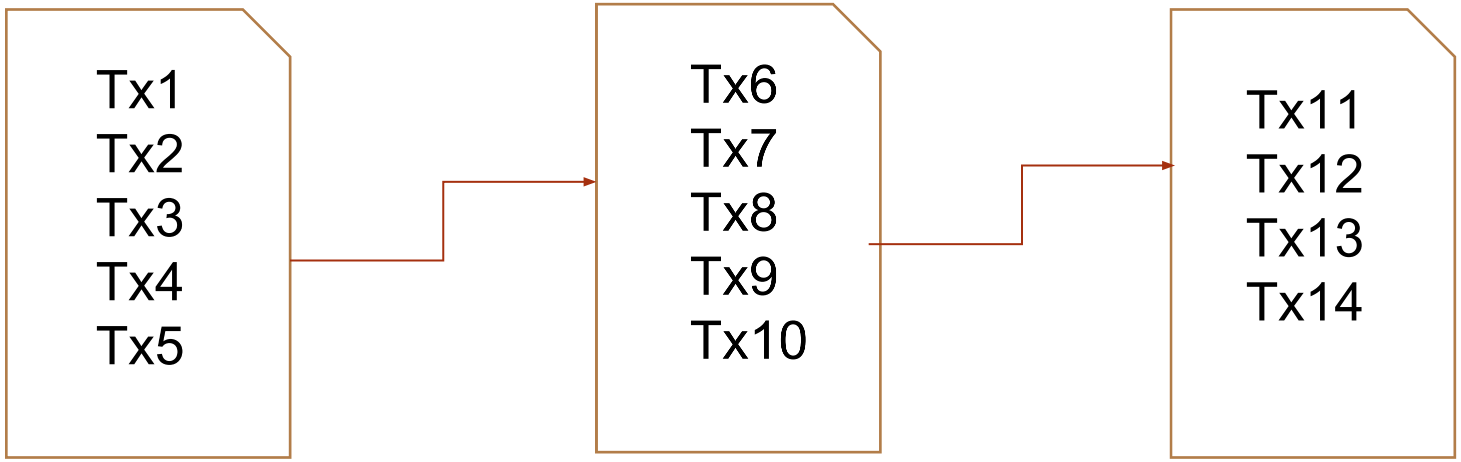
- The objective of the consensus algorithm is in bitcoin is to add a new block to the existing blockchain

- There can be multiple minors in the bitcoin network and individual minors can propose their new blocks based on the transaction that they have part off

- It is not necessary that every minor will propose a new block or a same block

- The time that the last block has been added in the blockchain they can include all those transactions in the new block provided that the total size of the block does not exists certain threshold

Bitcoin Consensus



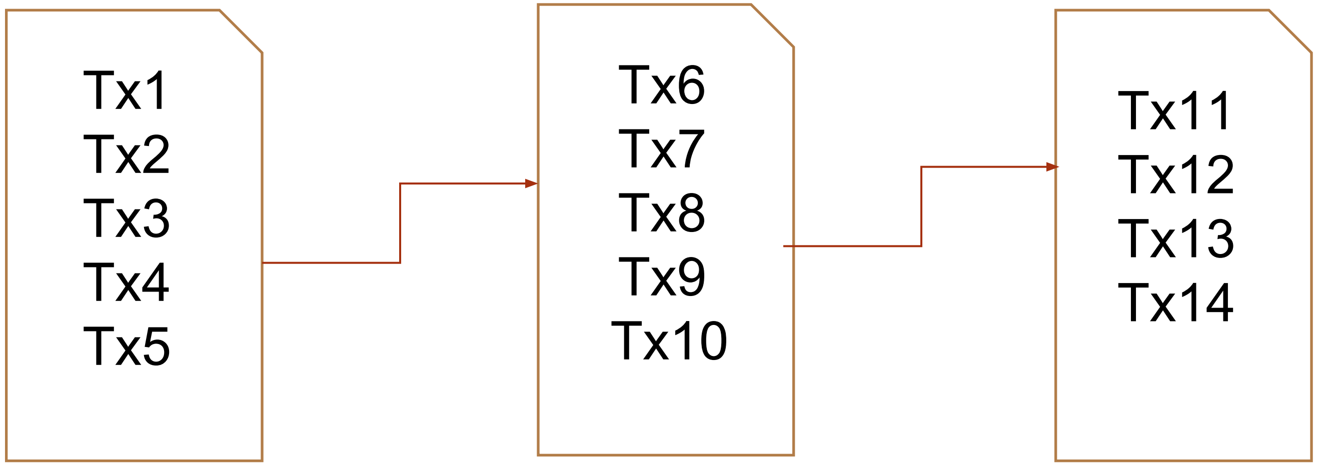
Bitcoin Consensus Objective:

Which block do we add next ?

Challenge: The miners do not know each other

Bitcoin Consensus

All minor broadcast the information and then apply choice function- distributed consensus algorithms



Tx1
Tx2
Tx3
Tx4
Tx5

Tx6
Tx7
Tx8
Tx9
Tx10

Tx11
Tx12
Tx13
Tx14

Minor will wait for certain duration that is technically infeasible

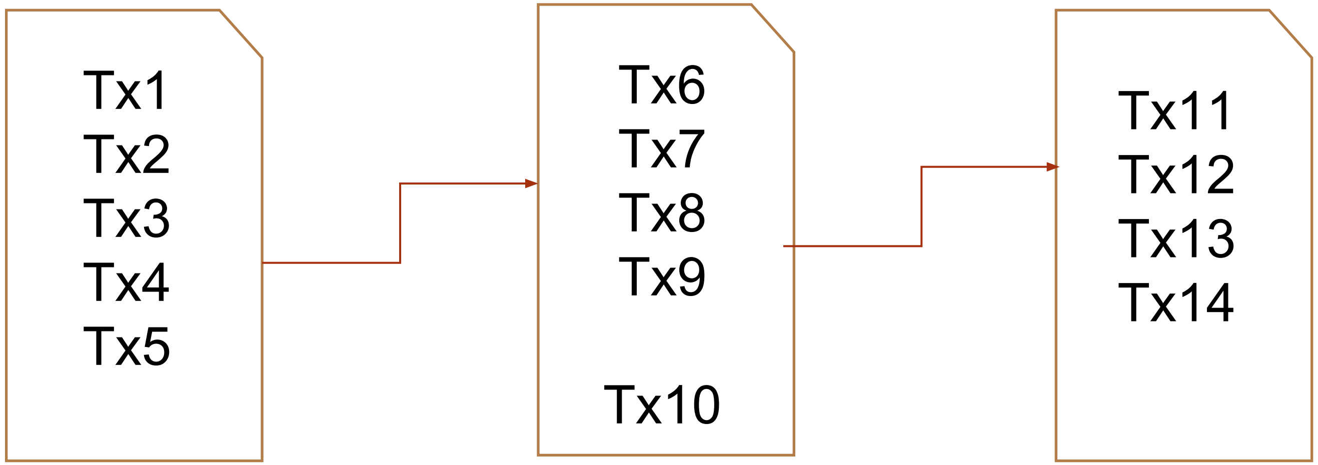
Tx15
Tx16
Tx18
Tx19

Tx15
Tx16
Tx17

Tx15
Tx16
Tx17
Tx18

Bitcoin Consensus

We do not have a global clock! How much time will we wait to hear the transactions



Tx1
Tx2
Tx3
Tx4
Tx5

Tx6
Tx7
Tx8
Tx9

Tx10

Tx11
Tx12
Tx13
Tx14

Tx15
Tx16
Tx18
Tx19

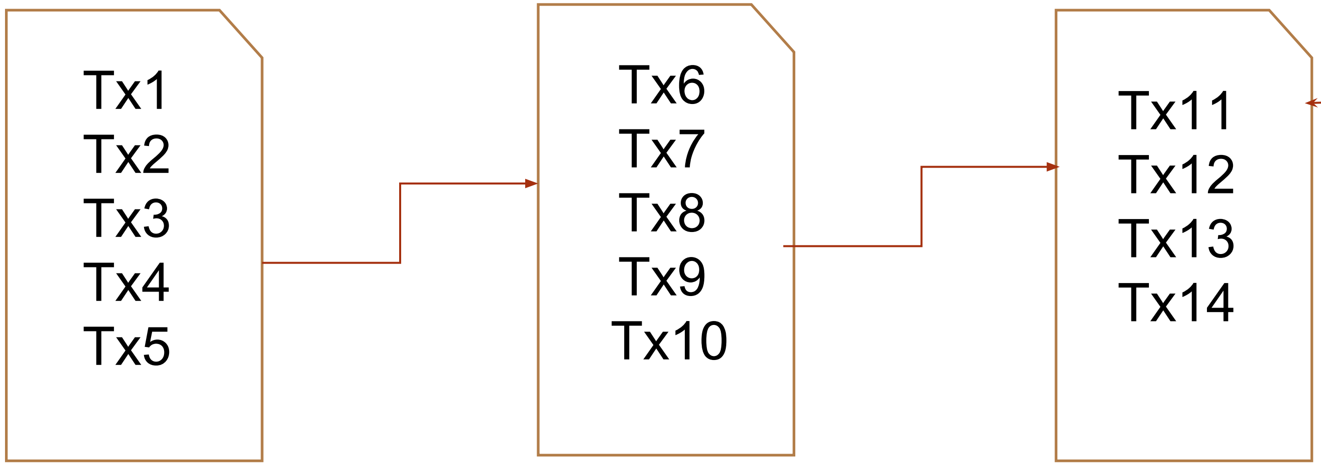
Tx15
Tx16
Tx17

Tx15
Tx16
Tx17
Tx18

Bitcoin Consensus

- The impossibility result in a purely asynchronous distributed network in the presence of even a single failure
- In this architecture if there is a minor which is an malicious, other minors may not be able to collect all the blocks which are coming from legitimate minors
- A consensus by applying this kind of traditional distributed algorithm based on broadcasting or message passing is infeasible in this scenario
- To solve this, we have two different observations
  - Any valid block(a block with all valid transactions) can be accepted even if it is proposed by only one miner and it is a valid block we can accept that block and connect that block to the existing blockchain
  - The protocol can work in rounds: broadcast the accepted block to the peers and collect the next set of transactions means a new block by excluding the transactions which has already committed in the blockchain

Bitcoin Consensus



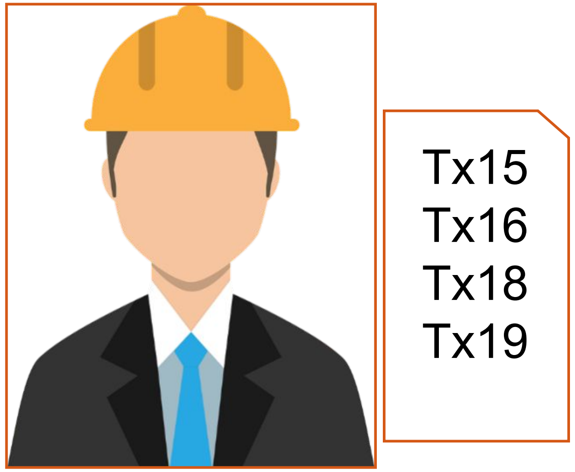
Tx1
Tx2
Tx3
Tx4
Tx5

Tx6
Tx7
Tx8
Tx9
Tx10

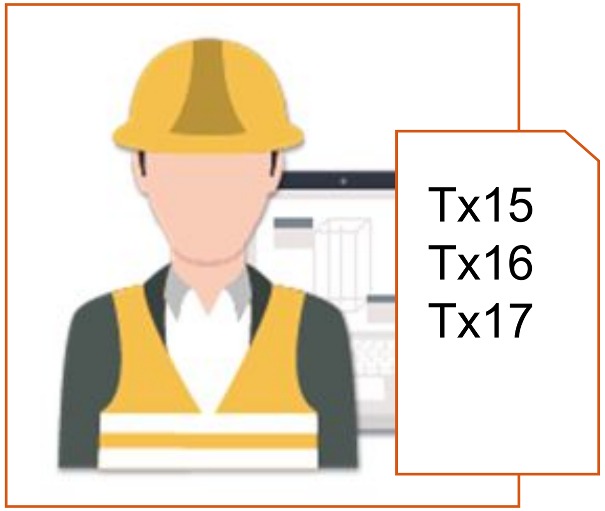
Tx11
Tx12
Tx13
Tx14

Last block in the blockchain,committed transaction

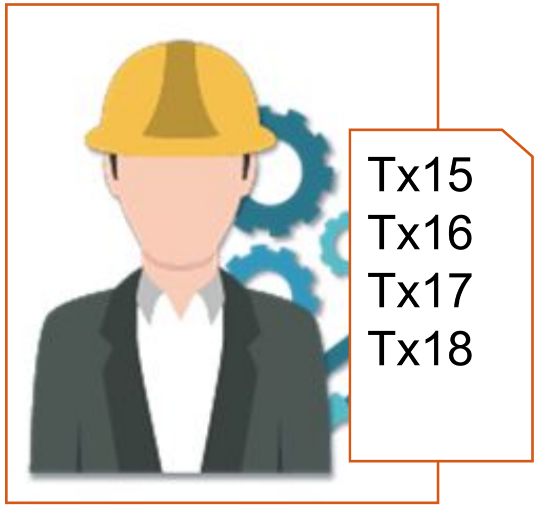
Propose a new block by excluding the transactions which has already committed in the blockchain



Tx15
Tx16
Tx18
Tx19

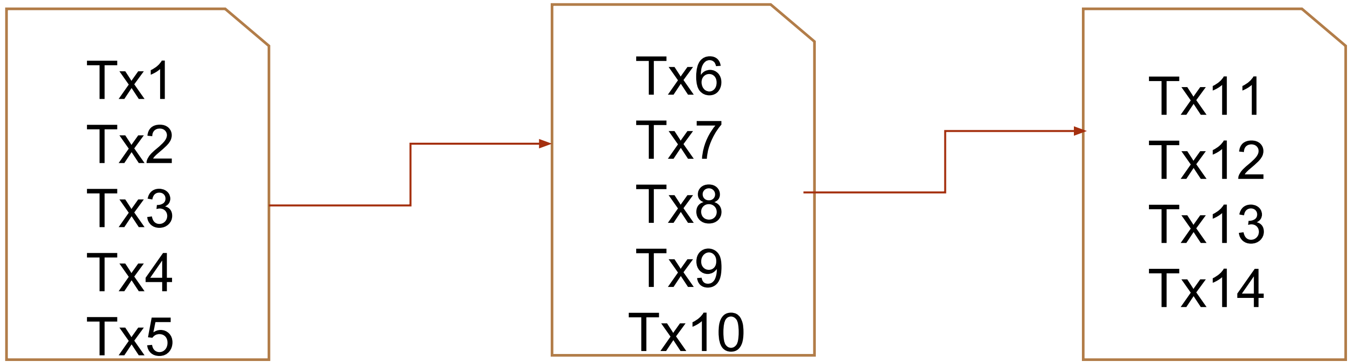


Tx15
Tx16
Tx17



Tx15
Tx16
Tx17
Tx18

Bitcoin Consensus



Tx1
Tx2
Tx3
Tx4
Tx5

Tx6
Tx7
Tx8
Tx9
Tx10

Tx11
Tx12
Tx13
Tx14

Solution:

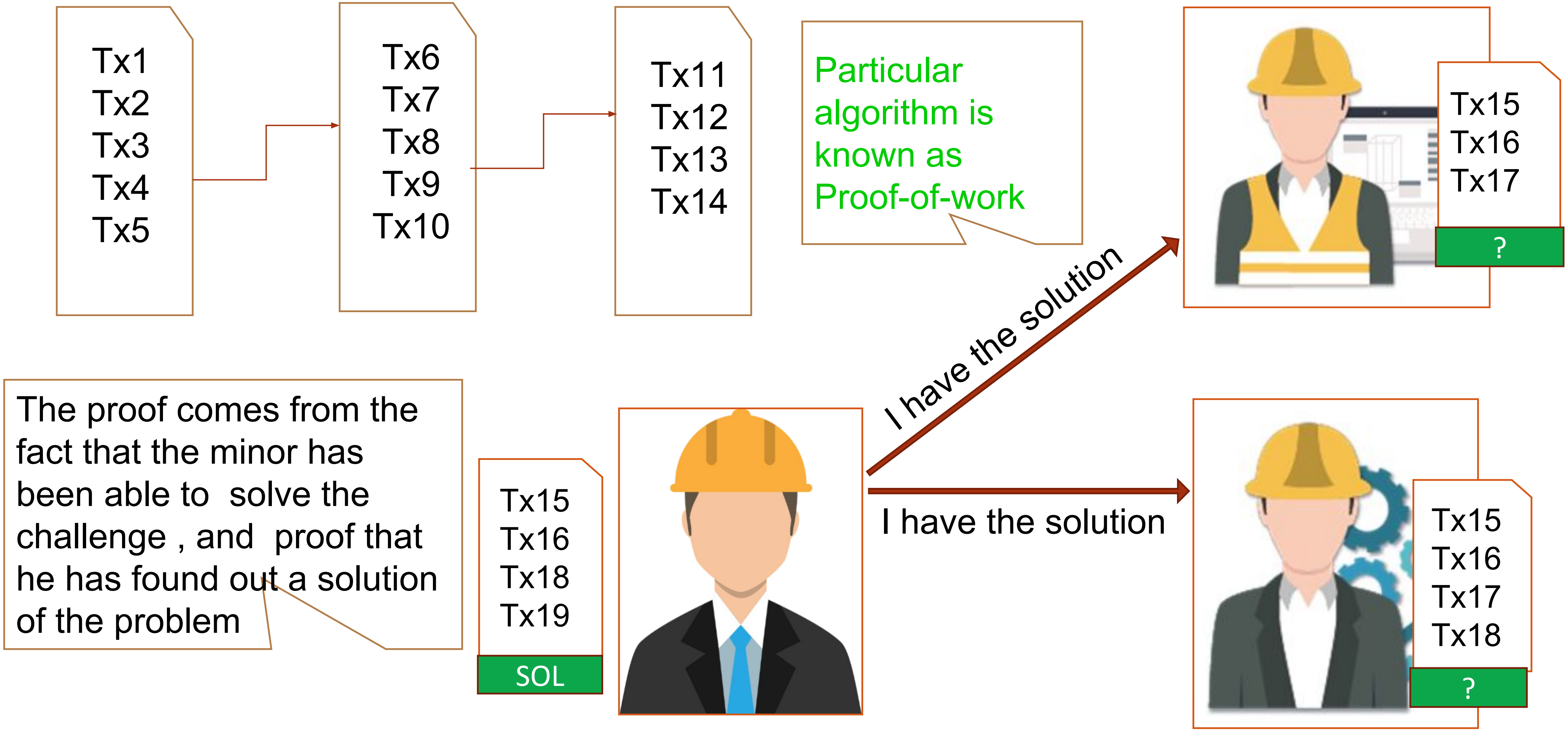
- Every miner independently tries to solve a challenge, provided by the network
- The block is accepted for the miner who can prove first that the challenge has been solved

Tx15
Tx16
Tx18
Tx19
?

Tx15
Tx16
Tx17
?

Tx15
Tx16
Tx17
Tx18
?

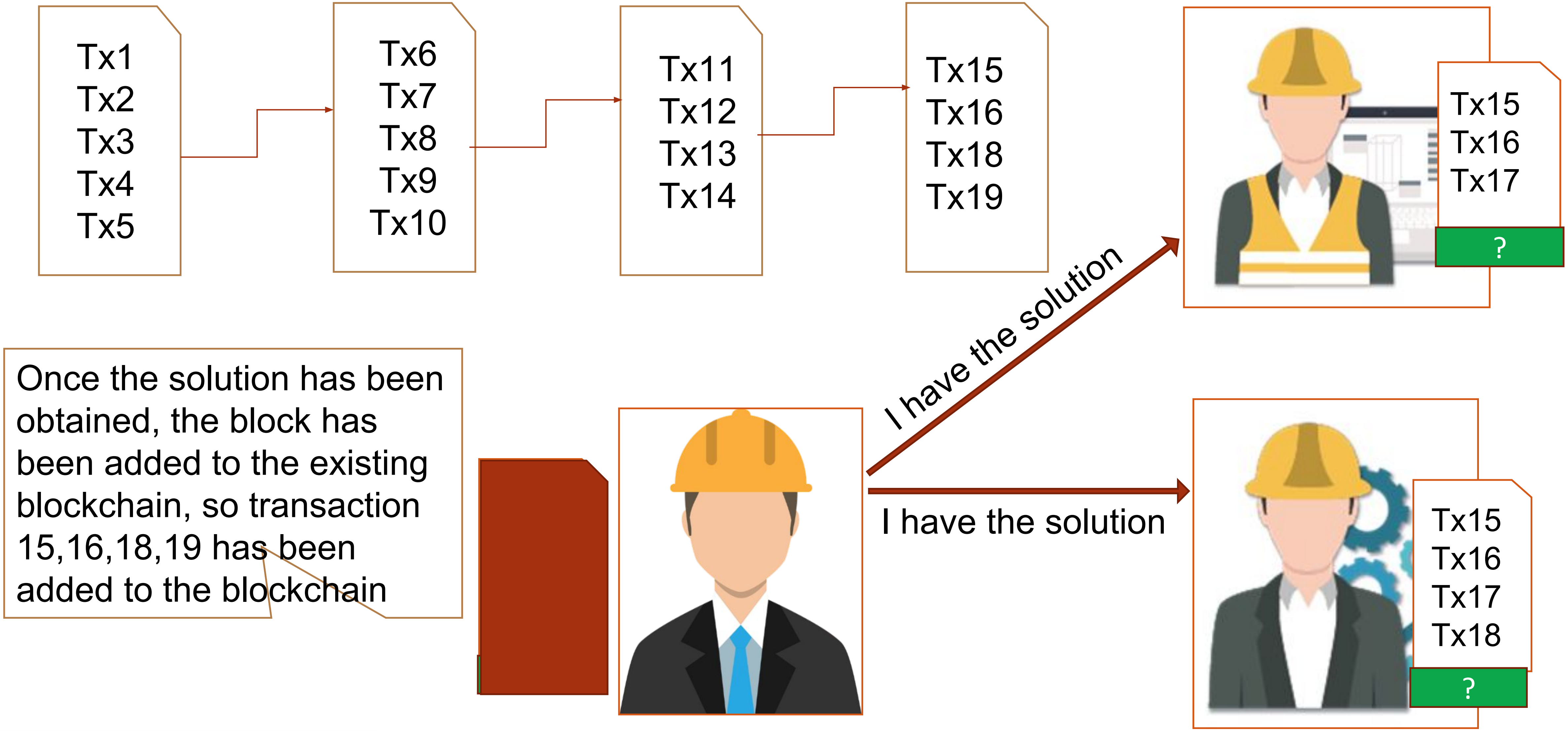
Bitcoin Consensus



Particular algorithm is known as Proof-of-work

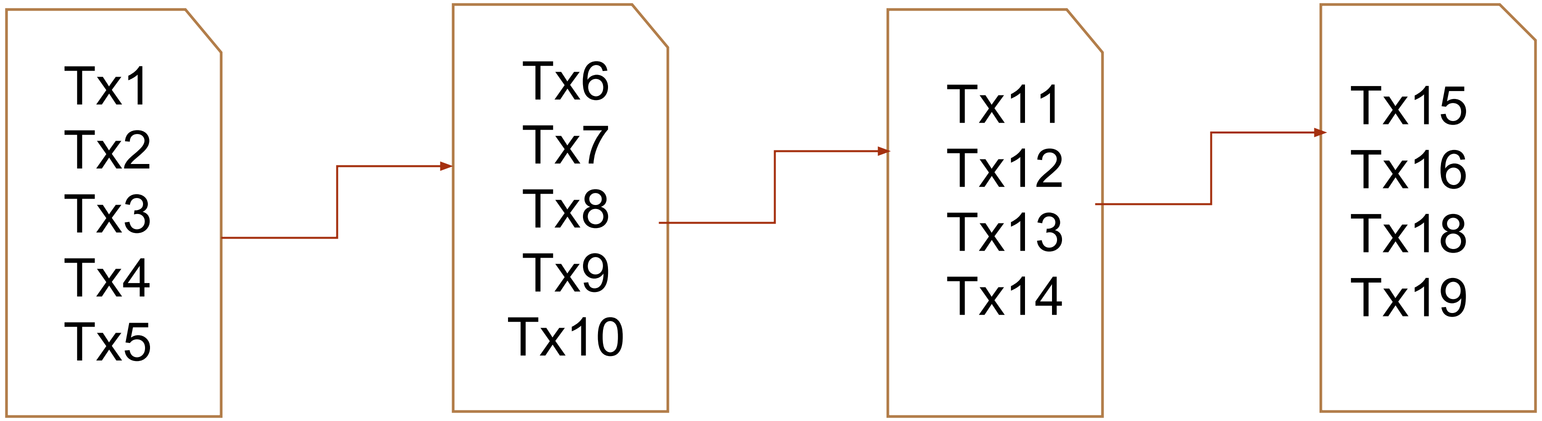
The proof comes from the fact that the minor has been able to solve the challenge , and proof that he has found out a solution of the problem

Bitcoin Consensus



Once the solution has been obtained, the block has been added to the existing blockchain, so transaction 15,16,18,19 has been added to the blockchain

Bitcoin Consensus



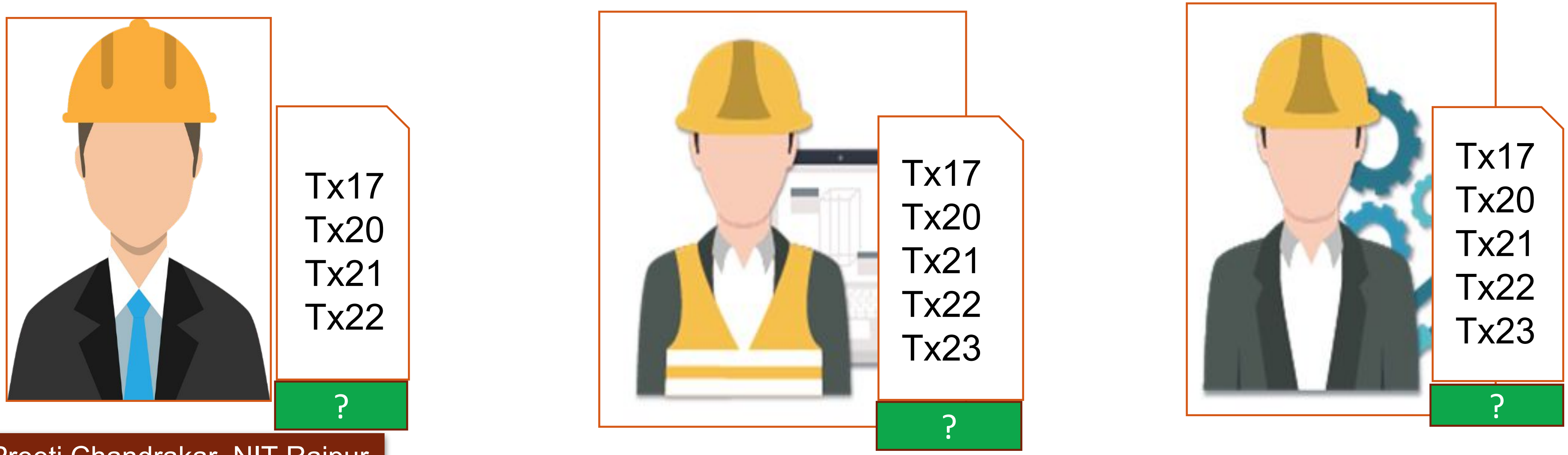
Tx1
Tx2
Tx3
Tx4
Tx5

Tx6
Tx7
Tx8
Tx9
Tx10

Tx11
Tx12
Tx13
Tx14

Tx15
Tx16
Tx18
Tx19

Note: Everyone can see that Tx18 and Tx19 have been committed, but Tx17 has not been committed. Include that in the next round



Tx17
Tx20
Tx21
Tx22

?

Tx17
Tx20
Tx21
Tx22
Tx23

?

Tx17
Tx20
Tx21
Tx22
Tx23

?

Proof-of- Work (PoW)

- An economic measure to deter service abuses by requiring some work from the service requester (usually processing time by a computer)

- The idea from Dwork and Naor (1992), from an interesting paper published in Crypto Conference to combat junk emails
  - We have to do some work to send a valid email
  - The attacker would be discouraged to send junk emails

Proof of Work(PoW) Features

- Asymmetry
  - The work must be moderately hard, but feasible for the service requester
  - The work must be easy check for the service provider

- Service requesters will get discouraged to forge the work, but service providers can easily check the validity of the work

Cryptographic Hash as the PoW

- Use the puzzle friendliness property of cryptographic hash function as the work
- Given X and Y, find out k, such that Y= Hash(X || k)
- It is difficult(but not infeasible) to find such k
- However, once you have a k, you can easily verify the challenge
- Used in Hashcash, a proof of work that can be added with an email as a “good-will” token

Hashcash PoW

- A non-interactive, publicly auditable, trapdoor-free and probabilistic unbounded cost-function
- Notation
  - Bit string $s=\{0,1\}^*$
  - Let $[s]_i$ be the ith bit, $[s]_1$ is the leftmost bit, and $[s]_s$ is the rightmost bit
  - $[s]_{i\ldots.j}$ is the substring from positions i to j
  - $||$ is the string concatenation operator
  - Operator $=_b$ compares first b bits(starting from left)
  - $x =_b y$ implies □i □ 1….b, $[x]_i = [y]_i$, and $[x]_{b+1} = [y]_{b+1}$
- Let 0^k be a very long string of zeros(k is a large number)

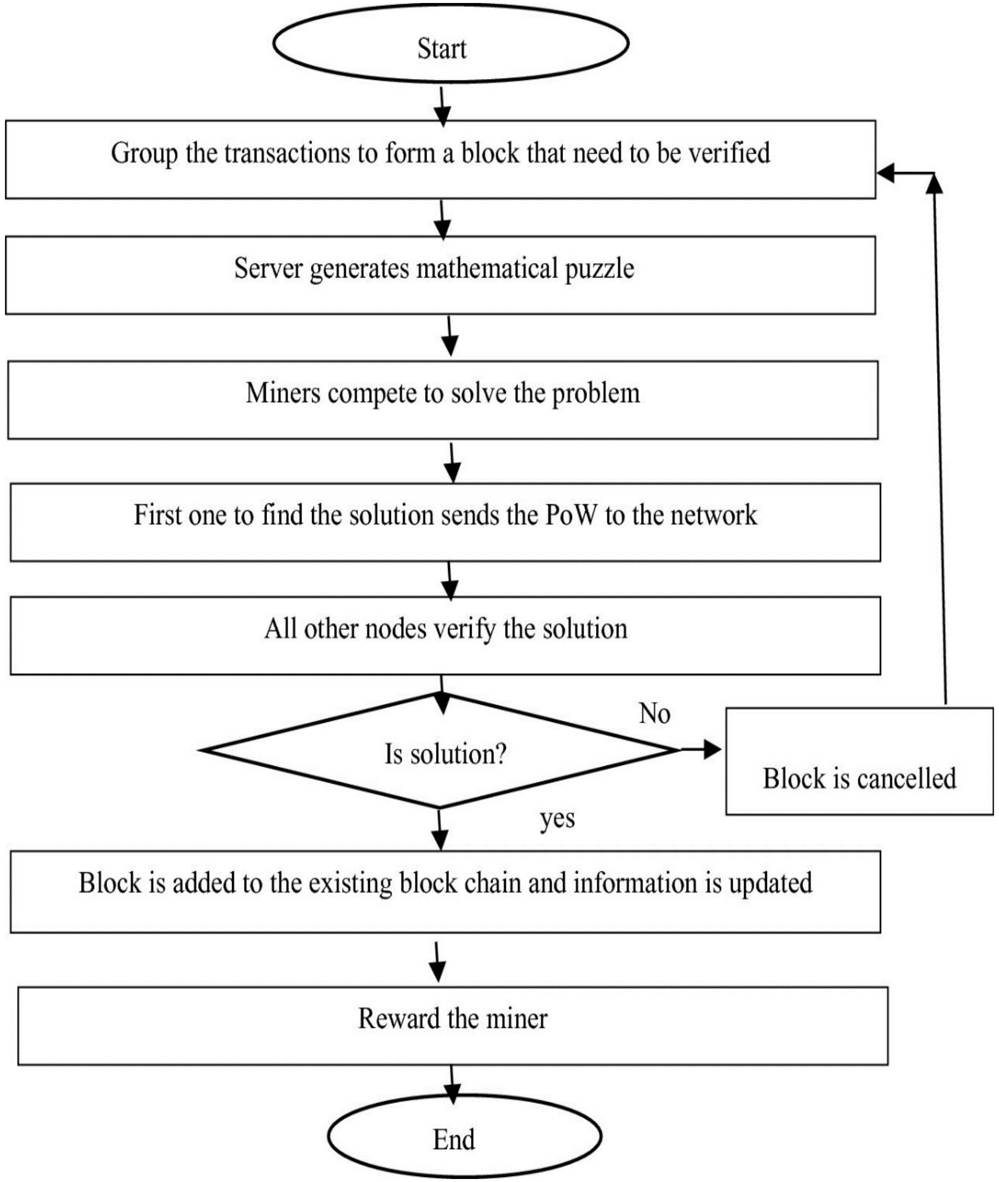
Hashcash PoW

- mint(….) and value(…) functions in the non-interactive variant
- $T \longleftarrow \text{mint}(s,w)$
- Find a bit-string x such that
  - $H(s \,||\, x) =_w 0^k$
  - T=(s,x)
  - $V \longleftarrow \text{value}(T)$
  - $H(s \,||\, x)=v\ 0^k$
- If v=w solution found
- H is hash function such as SHA-256
- Time to mint a token is exponentially dependent on w
- Different tokens have different values of s

Bitcoin Proof-of- Work

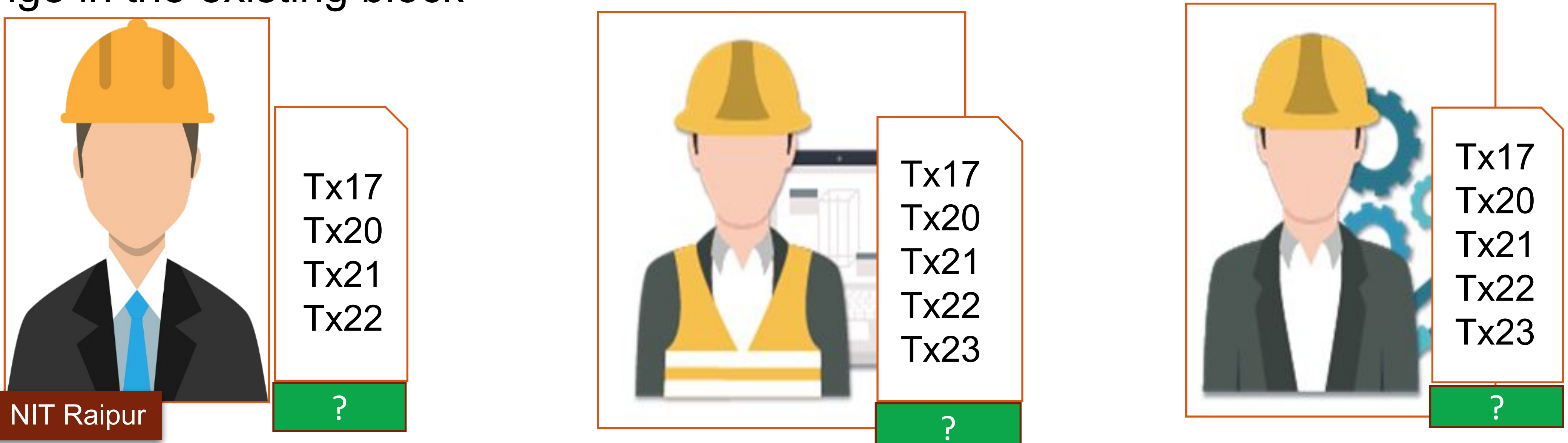
- Bitcoin proof-of-work (PoW) algorithm ensures consensus over a permission less setting based on challenge response
- Every node spends a large amount of computational power to solve the mathematical challenge in each iteration of consensus
- The computational effort expended by the nodes in achieving consensus would be paid for by cryptocurrency generated and managed by the network
- We choose a nonce such that:
  - $H(\text{ nonce } \| \text{prev}_{\text{hash}} \| \text{Tx1}\|\text{Tx2}\ldots\ldots\| \text{Txn}) =_{n} 0^{k}$
  - The first n bits of the cumulative hash are all 0s
  - It takes time to compute such a nonce
- Chain gets forked, nodes will always choose the longest chain

Bitcoin Proof-of- Work

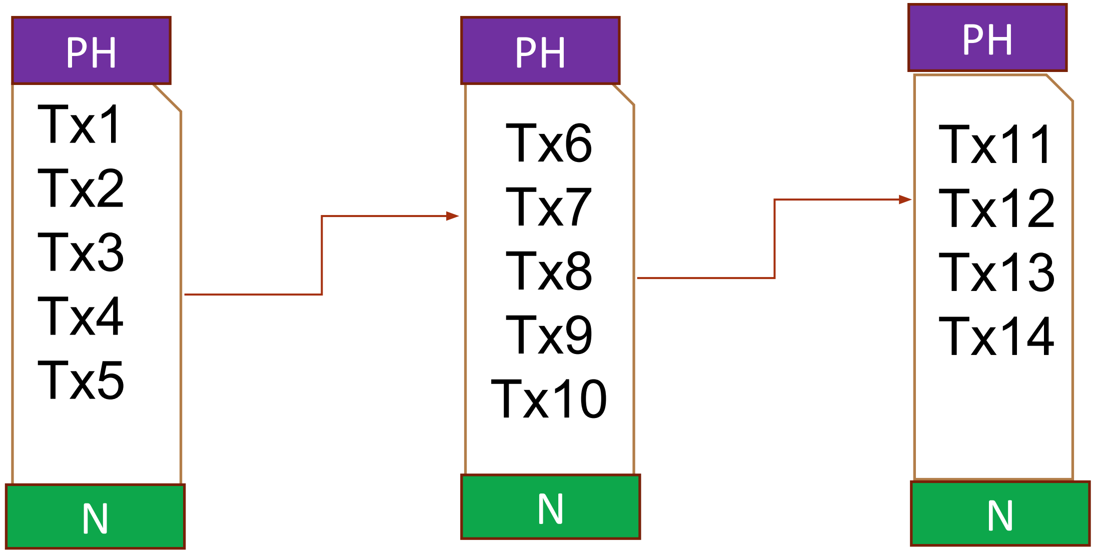

Start
Group the transactions to form a block that need to be verified
Server generates mathematical puzzle
Miners compete to solve the problem
First one to find the solution sends the PoW to the network
All other nodes verify the solution
Is solution?
No
Block is cancelled
yes
Block is added to the existing block chain and information is updated
Reward the miner
End

Bitcoin Proof of Work (PoW)

- Based on Hashcash PoW system
  - The miners need to give a proof that they have done some work, before proposing a new block
  - If they can successfully complete that work then they are able to submit that block as a part of the existence longest chain of the blockchain
  - The attackers will be discouraged to propose a new block, or make a change in the existing block



Bitcoin Proof of Work System

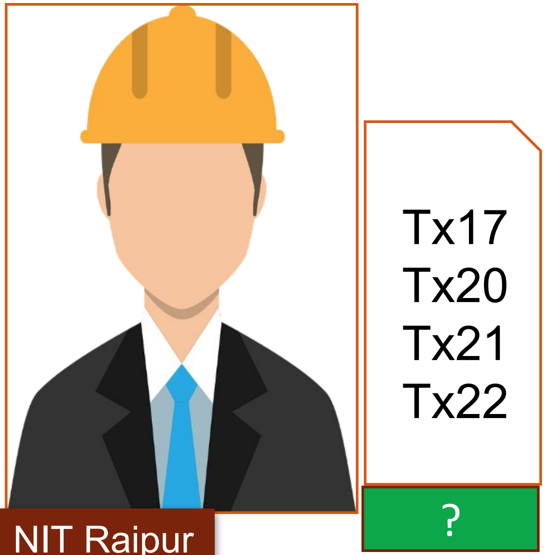


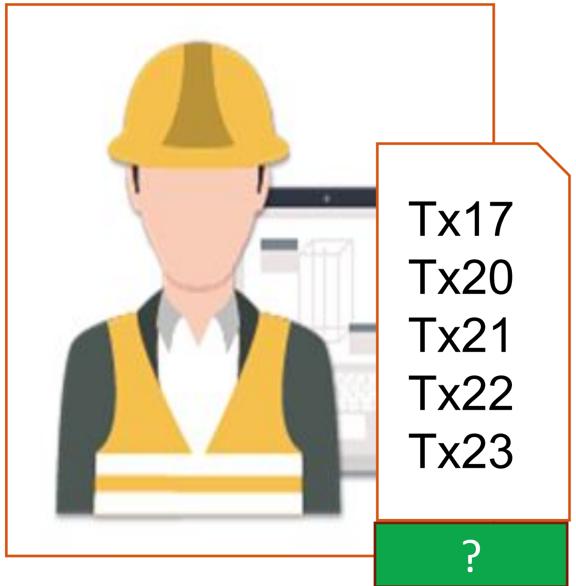
BH: Block Hash
PH: Previous Block Hash
MR: Merkle Root
N: Nonce

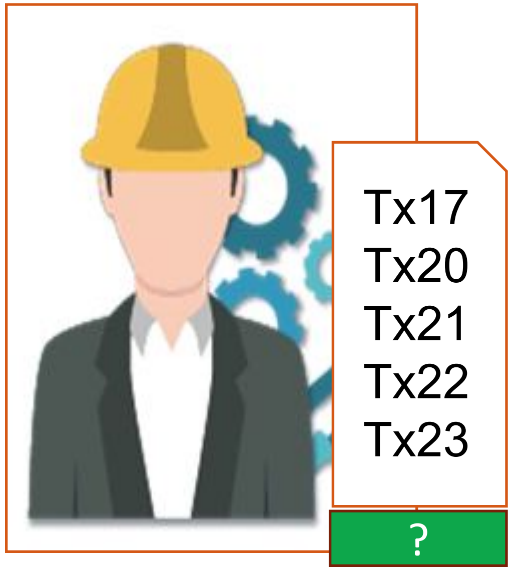
BH should have certain zero(difficulty) at the beginning

BH= Hash(PH:MR:N)

N=?







Bitcoin Proof of Work System

- Most implementations of Bitcoin PoW use double SHA256 hash function
- The miners collect the transactions for 10 minutes(default setup) and starts mining the PoW
- The Probability of getting a PoW is low- it is difficult to say which miner will be able to generate the block
- No miner will be able to control the bitcoin network single handedly

Breaking Bitcoin PoW

- Bitcoin PoW is computationally difficult to break, but not impossible to break the PoW based system
- Attackers can deploy high power servers to do more work than total work of the blockchain

The Monopoly Problem

- PoW depends on the computing resources available to a miner
- If a minor can pause a huge amount of computational resource, then a minor can control the entire network
- If huge number of blocks in the blockchain goes to a single minor, then this minor has the ability to control the entire flow of transactions in the blockchain
- This particular problem we call as the monopoly problem in bitcoin network, which is a short coming of the PoW based system
- Monopoly can increase over time
  - Miners will get less reward over time
  - Users will get discouraged to join as the miner
  - Few miners with large computing resource may get control over the network

Handling Monopoly and Power Consumption-Proof of Stake (PoS)

- Handling monopoly and power consumption problems
- Possibly proposed in 2011 by a member in Bitcoin Forum-
- PoW vs PoS
  - PoW: Probability of mining a block depends on the work done by the miner
  - PoS: Amount of bitcoin that the miner holds- Miner holding 1% of the Bitcoin can mine 1% of the PoS blocks

Proof of Stake (PoS)

- Provides increased protection
  - Existing an attack is expensive , we need more Bitcoin
  - **Reduced incentive for attack-** the attacker needs to own a majority of bitcoins- an attack will have more affect on the attacker
- Variants of “Stake”
  - Randomization in combination of the stake(used in Nxt and Blockchain)
  - **Coin-age**- Number of coins multiplied by the number of days the coins have been held (used in Peercoin)

Proof of Burn (PoB)

- Miners should show proof that they have burned some coins to mine a new block
  - Burning it is like that they have to send them to a verifiably un spendable address where no one will be able to spend that bitcoin
  - Expensive just like PoW, because in PoW, the physical money that we have to invest to purchase computational hardware, here we have to spend digital or logical resources which are bitcoins but no external resource are used other than the burned coins
- If the attacker wants to attack this system, the attacker actually have to make a loss of huge amount of bitcoins

Proof of Burn (PoB)

- PoW vs PoB-
  - In case of proof of work we are utilizing real resources whereas, in case of proof of burn we are utilizing virtualized or digital resources
- PoB works by burning PoW mined cryptocurrencies
- PoS and PoB Ultimately depends on PoW mined cryptocurrencies
  - once the proof of work get settle down and people have certain amount of digital currencies with them, then gradually we can move to proof of stake or proof of burned based mechanism
- We cannot use them to bootstrap a new blockchain

PoW vs PoS vs PoB

PoW	PoS	PoB
Do some work to mine a new block	Acquire sufficient stake to mine a new block	Burn some wealth to mine a new block
Consumes physical resources, like CPU power and time	Consumes no external resource, but participate in transactions	Consumes virtual or digital resources, like the coins
Power hungry	Power efficient	Power efficient

Proof of Elapsed Time (PoET)

- Proposed by Intel, as a part of Hyperledger Swatooth- a blockchain platform for building distributed ledger applications
- Basic idea:
  - Each participants in the blockchain network waits a random amount of time
  - The first participant to finish becomes the leader for the new block
  - We make a randomization among the miners. So, the minor who will be able to complete that random waiting first, that minor will be able to proposed the new block

Proof of Elapsed Time (PoET)

- The challenge is that, how will one verify that the proposer has really waited for a random amount of time?
  - Utilize special CPU instruction set- Intel Software Guard Extension(SGX)- a trusted execution platform
  - The trusted code is private to the rest of the application
  - The specialized hardware provides an attestation that the trusted code has been set up correctly