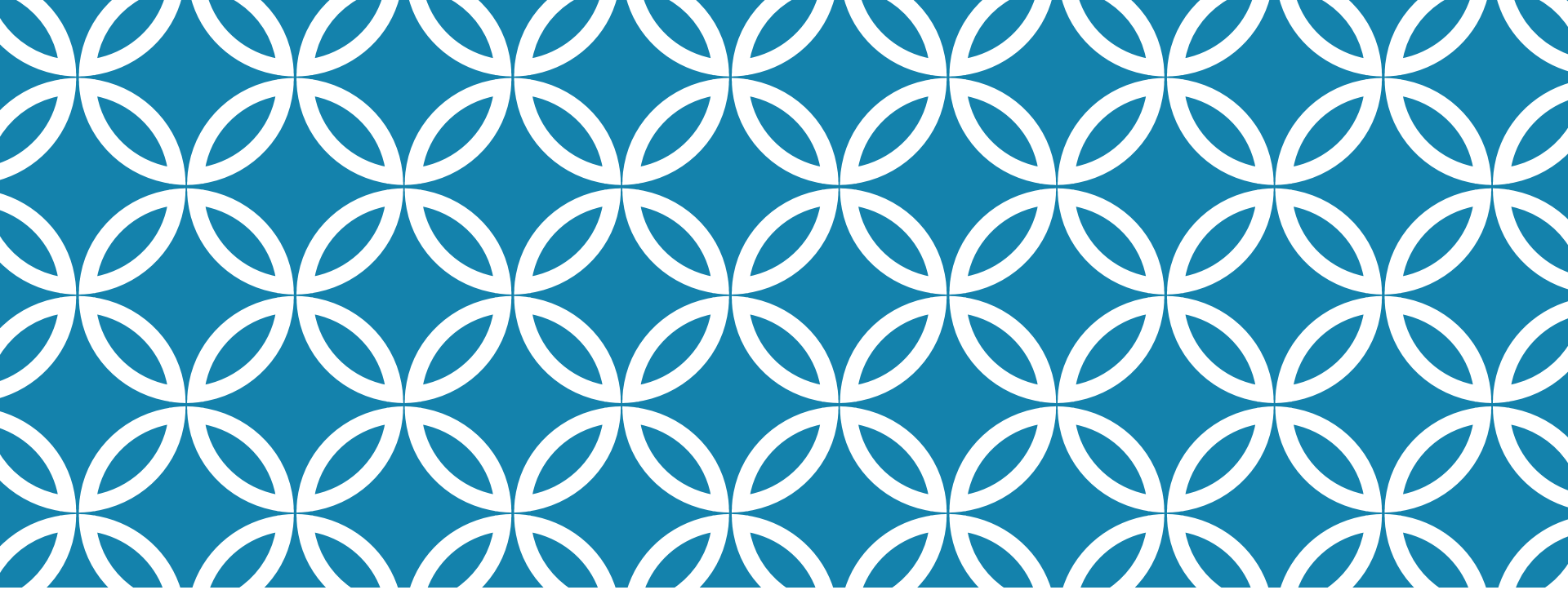
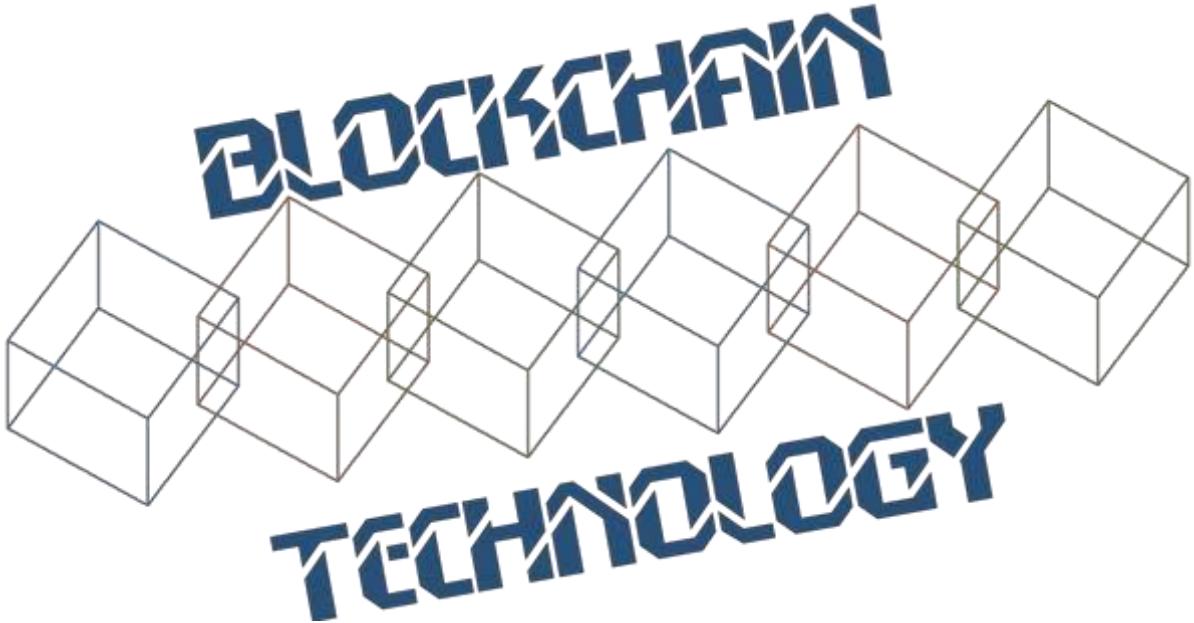


BLOCKCHAINS

ARCHITECTURE, DESIGN AND USE CASES

Dr. Sudeep Tanwar

Image courtesy: http://beetfusion.com/



DISTRIBUTED CONSENSUS

WHAT IS CONSENSUS ALGORITHM-WHICH HELP IN ACHIEVING CONSENSUS AMONG DIFFERENT DISTRIBUTED APPLICATIONS

DIFFERENT METHODS OF CONSENSUS

HOW THEY ARE APPLICABLE FOR GENERAL BLOCKCHAIN ENVIRONMENT

CONSENSUS

- A procedure to reach in a **common agreement** in a distributed or decentralized multi-agent platform
- Important for a message passing system
- Each General have his individual opinion/advise
- Each General can apply Choice Function, which can be majority decision in this particular case.
- After that system decides what to do next
- With the majority principle system, come to a consensus that Attack is ok.

Attack

Retreat

Attack

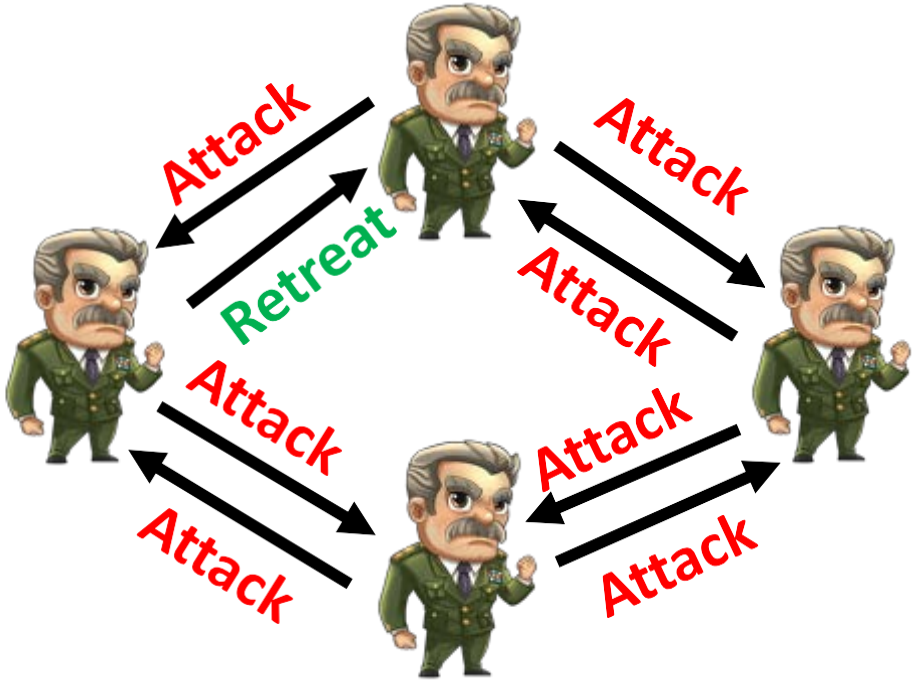
Attack

WHY WE REQUIRE CONSENSUS

- Ensure Reliability and fault tolerance in a convectional distributed system
  - Ensure correct operations in the presence of faulty individuals
  - When you have multiple parties then any one can work in a faulty or malicious way so for such scenario common view point is important
  - **Means perform correct operations in the presence of faulty users.**
- **Example:**
  - **Clock synchronization:** You have multiple clock in the network and every node wants to find which watch time is updated so they make a consensus among them and come to a single clock and by applying this clock synchronization architecture they can do further operation
  - **Commit a transaction in a database-Bank**
  - **State machine replication**

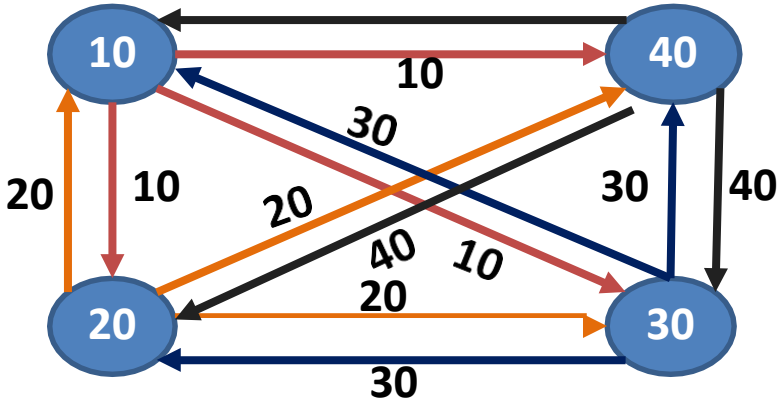
WHY CONSENSUS CAN BE DIFFICULT IN CERTAIN SCENARIOS

- Consider a message passing system, and a general behaves maliciously



DISTRIBUTED CONSENSUS

- If there is no failure, it is easy and trivial to reach in a consensus
  - Broadcast the personal choice to all
  - Apply a choice function, say the maximum of all the values **40**
  - Conditions for the same: systems should behave in a synchronous way, means a system where all nodes get the information within the given time frame and its failure free



DISTRIBUTED CONSENSUS

- There can be various types of faults in a distributed system.
- **Crash Fault:** A node suddenly crashes or becomes unavailable in the middle of a communication, means not received any expected messages
- **Network or Partitioned Faults**: A network fault occurs (say the link failure) and the network gets partitioned
- Example, assume multiple nodes in the networks in the network and these nodes are interconnected with each other. Then, consensus becomes problem here
- **Byzantine Faults**: A node starts behaving maliciously. Very difficult fault to deal with. Some times node can sent positive message or some times negative message.

DISTRIBUTED CONSENSUS - PROPERTIES

- **Termination**: Every correct individual decides some value at the end of the consensus protocol
- **Validity:** If all the individuals proposes the same value, then all correct individuals decide on that value
- **Integrity:** Every correct individual decides at most one value, and the decided value must be proposed by some individuals
- **Agreement:** Every correct individual must agree on the same value

SYNCHRONOUS VS ASYNCHRONOUS SYSTEMS

- **Synchronous Message Passing System:** The message must be received within a predefined time interval
  - Strong guarantee on message transmission delay
  - Give the simplification in designing the protocol
- **Asynchronous Message Passing System**: There is no upper bound on the message transmission delay or the message reception time
  - No timing constraint, message can be delayed for arbitrary period of times
  - You can not expect for finite duration
  - Difficult to design it because delay is not known.

ASYNCHRONOUS CONSENSUS

- **FLP85 (Impossibility Result):** In a purely asynchronous distributed system, the consensus problem is impossible (with a deterministic solution) to solve if in the presence of a single crash failure.
  - Results by Fischer, Lynch and Patterson (most influential paper awarded in ACM PODC 2001)-for formal proof of the same refer paper.
  - Randomized algorithms may exist

M. Fischer, N. Lynch and M. Paterson.**Impossibility of distributed commit with one faulty process.***Journal of the ACM*, 32(5), pages 374--382. 1985.

SYNCHRONOUS CONSENSUS

- Various consensus algorithms have been explored by the distributed system community
  - Paxos
  - Raft
  - Byzantine fault tolerance (BFT)

We'll look into these consensus algorithms, but later !!

CORRECTNESS OF A DISTRIBUTED CONSENSUS PROTOCOL -PROPERTIES

- **Safety:** Correct individuals must not agree on an incorrect value
  - Nothing bad happend

- **Liveliness (or Liveness):** Every correct value must be accepted eventually
  - Something good eventually happens
  - If you are proposing good values then after the termination of consensus algorithm it will accepted eventually

CONSENSUS IN AN OPEN SYSTEM

- Look at the consensus mechanism in BC environment
- The tradition distributed consensus protocols are based on
  - **Message passing** (when individuals are connected over the Internet)
  - **Shared memory** (when a common memory place is available to read and write the shared variables that everyone can access)
- Message passing requires a closed environment – everyone need to know the identity of others- means identity of each node.
- But for BC especially BITCOIN environment, we have the open network or permission less environment where every node can join

CONSENSUS IN AN OPEN SYSTEM

- **In open system, we have two broad types of algorithms**
- **Shared memory** is not suitable for Internet grade computing, because you need to put memory, which should be readable and writable by every individual nodes in the network
  - Where do you put the shared memory?

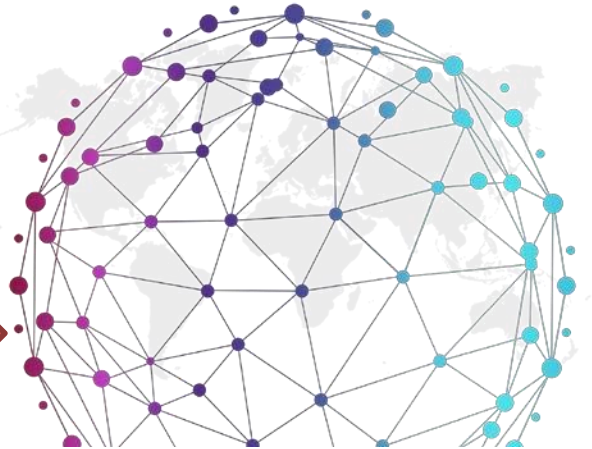
  Message passing is not possible in open environment
- Bitcoin is an open environment
  - Anyone can join in the Bitcoin network anytime
  - **How do you ensure consensus in such an open system? – A key challenge**

WHY DO WE REQUIRE CONSENSUS IN BITCOIN NETWORK

- Bitcoin is a peer-to-peer network
- Alice broadcast a transaction to Bob in this peer-to-peer network
- This broadcast is different from the traditional message passing system
- **All the nodes in this network need to agree on the correctness of this transaction**-Require Consensus because of this

Signature of Alice

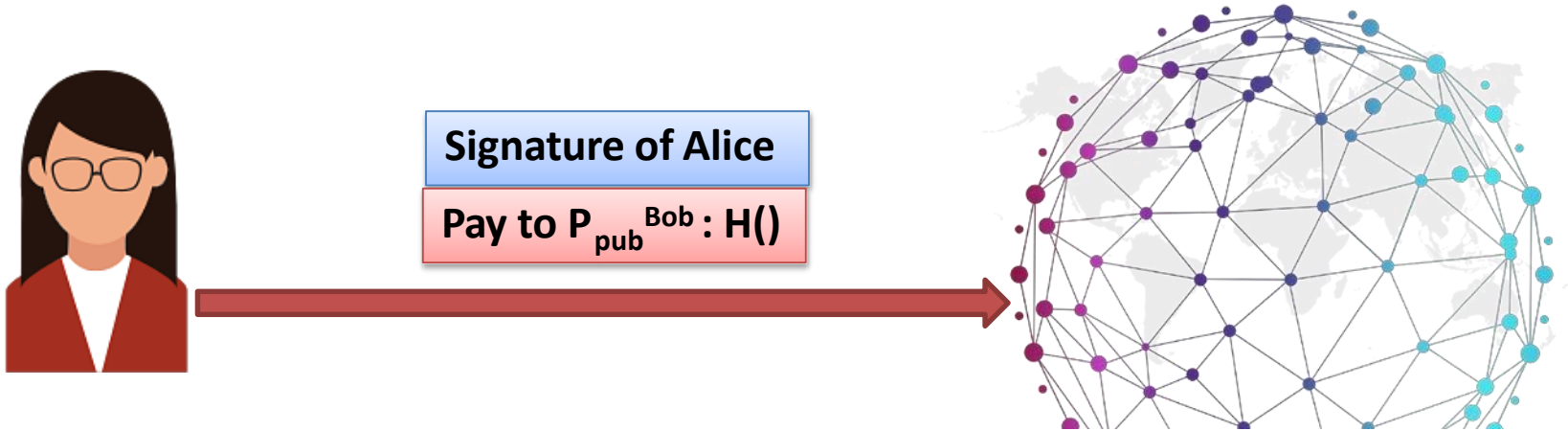
Pay to P_{pub}^{Bob} : H()



How will you verify that the transaction coming from Alice not from an Attacker(Means correctness of the transaction)?

WHY DO WE REQUIRE CONSENSUS IN BITCOIN NETWORK

- A node does not know all the peers in the network – this is an open network
- Some nodes can also initiate **malicious transactions**
- **We need to prevent the network from the malicious transactions**

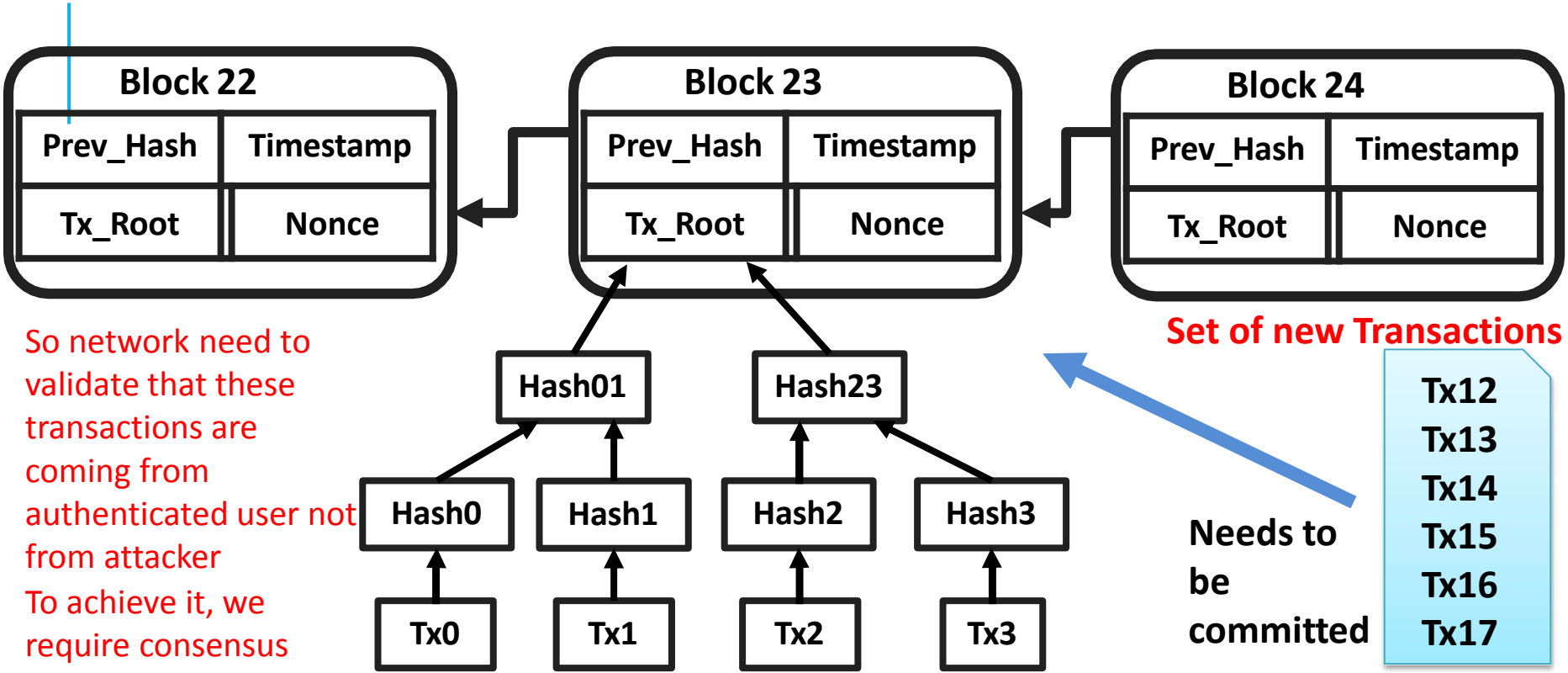


CONSENSUS IN A BITCOIN NETWORK

- Every node has **block of transactions** that has already reached into the consensus (block of committed transactions), which is stored in the form of Blockcahin

- The nodes also has a list of outstanding transactions that need to be validated against the block of committed transactions (Means existing Blockchain)

CONSENSUS IN A BITCOIN NETWORK EXAMPLE

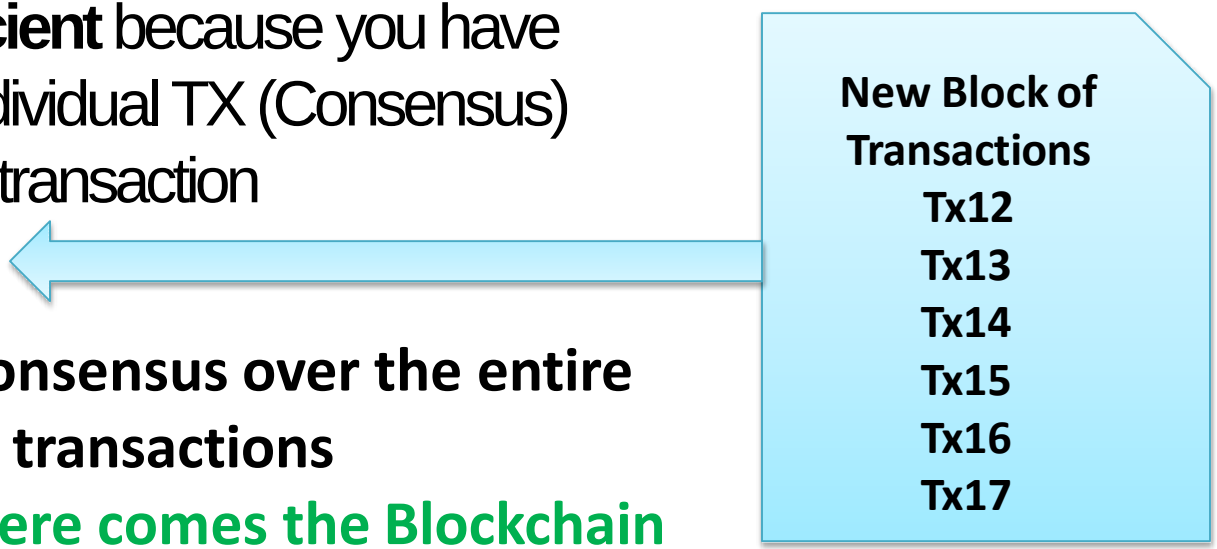


- So network need to validate that these transactions are coming from authenticated user not from attacker
- To achieve it, we require consensus protocols in Bitcoin Network

CONSENSUS IN BITCOIN

- **Per transaction consensus**
  - **Inefficient** because you have to run individual TX (Consensus) for each transaction

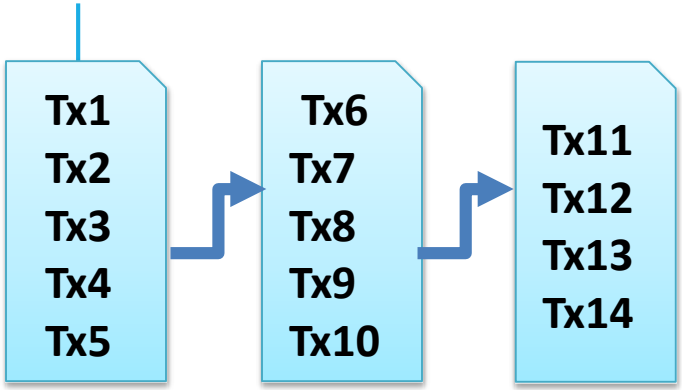
- **Block based consensus**

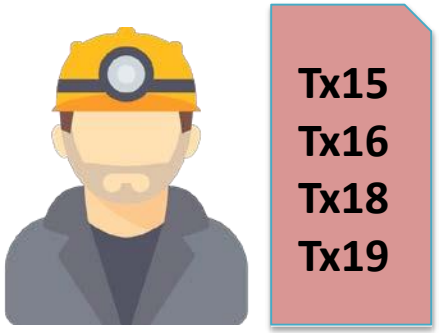


Apply consensus over the entire block of transactions

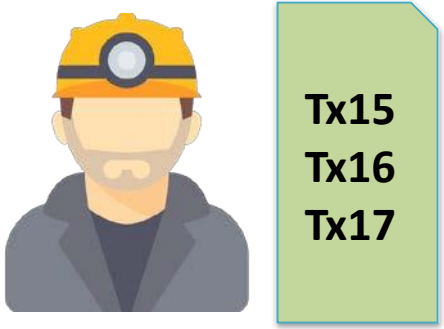
- **Here comes the Blockchain**

CONSENSUS IN BITCOIN





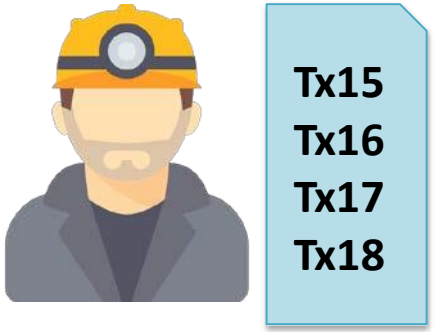
M1



M2

Bitcoin Consensus Objective:

- Which block do we add next to the existing BC?
- This is the problem of Bitcoin Consensus Algorithms ??



M3