SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF INFORMATION TECHNOLOGY

BLOCK CHAIN AND CRYPTOCURRENCY

IV YEAR - VII SEM

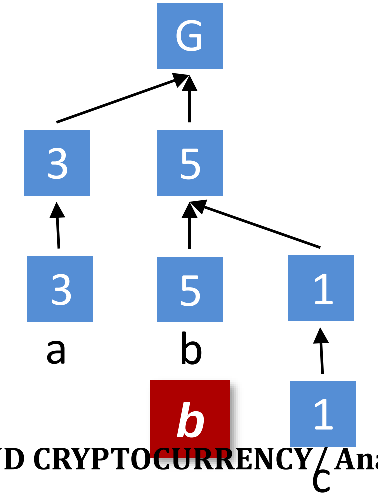
UNIT 3 - DISTRIBUTED CONSENSUS & BLOCK CHAIN APPLICATIONS

Nakamoto Consensus

Bitcoin uses **Nakamoto consensus**:

- **Fork-choice / proposal rule:** At any given time, each honest miner attempts to extend (i.e., mines on the tip of) the <u>heaviest</u> chain *held* in its view (Ties broken adversarially).

Difficulty values on the blocks



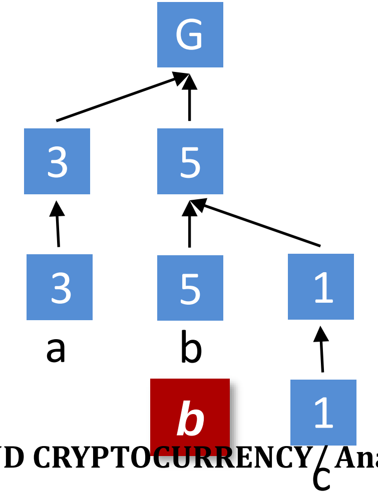
Nakamoto Consensus

Chain with the highest difficulty, i.e, largest sum of the difficulty D within blocks!

Bitcoin uses **Nakamoto consensus**:

- **Fork-choice / proposal rule:** At any given time, each honest miner attempts to extend (i.e., mines on the tip of) the heaviest chain *held* in its view (Ties broken adversarially).

Difficulty values on the blocks

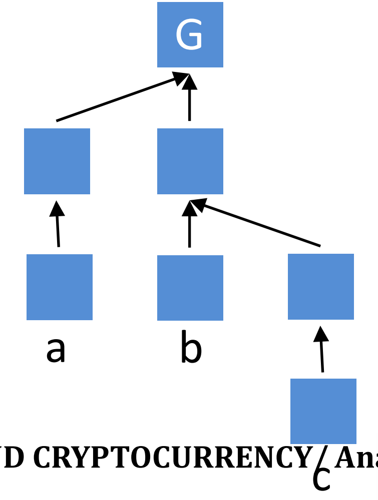


Nakamoto Consensus

Chain with the highest difficulty, i.e, largest sum of the difficulty D within blocks!

Bitcoin uses **Nakamoto consensus**:

- **Fork-choice / proposal rule:** At any given time, each honest miner attempts to extend (i.e., mines on the tip of) the heaviest (longest for us) chain *held* in its view (Ties broken adversarially).



Nakamoto Consensus

Bitcoin uses **Nakamoto consensus**:

- **Fork-choice / proposal rule:** At any given time, each honest miner attempts to extend (i.e., mines on the tip of) the heaviest (longest for us) chain *held* in its view (Ties broken adversarially).
- **Confirmation rule:** Each miner confirms the block (along with its prefix) that is *k*-deep within the longest chain in its view.
  - In practice, $k = 6$.
  - Miners and clients accept the transactions in the latest confirmed block and its prefix as their log.
  - Note that *confirmation* is different from *finalization*.
- **Leader selection rule:** Proof-of-Work.

Nakamoto Consensus

Chain with the highest difficulty, i.e, largest sum of the difficulty D within blocks!

Bitcoin uses **Nakamoto consensus**:

- **Fork-choice / proposal rule:** At any given time, each honest miner attempts to extend (i.e., mines on the tip of) the heaviest (longest for us) chain *held* in its view (Ties broken adversarially).
- **Confirmation rule:** Each miner confirms the block (along with its prefix) that is *k*-deep within the longest chain in its view.
  - In practice, $k = 6$.
  - Miners and clients accept the transactions in the latest confirmed block and its prefix as their log.
  - Note that *confirmation* is different from *finalization*.
- **Leader selection rule:** Proof-of-Work.

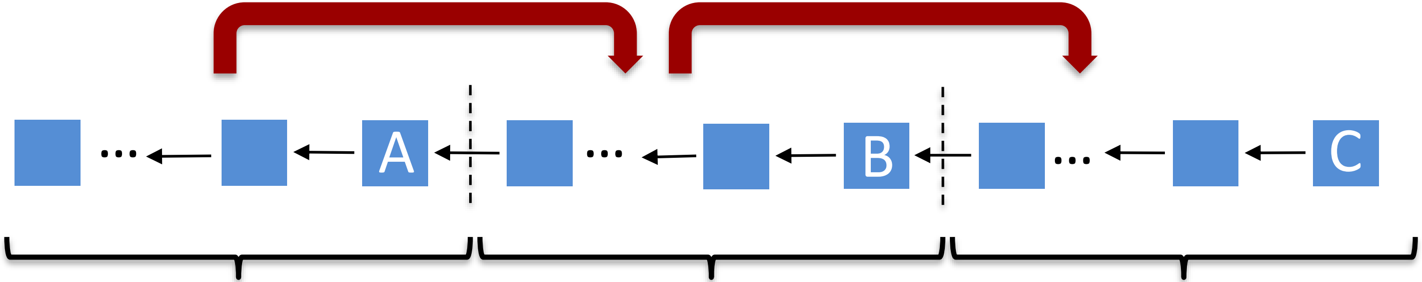
Nakamoto Consensus

Available **under dynamic participation**

Alice's log: $txs_1\ txs_3$

Bitcoin: Difficulty Adjustment

New target: $T_2 = T_1 \frac{t_1}{2016 \times 10\ mins}$ New target: $T_3 = T_2 \frac{t_2}{2016 \times 10\ mins}$



2016 blocks
Time it took to mine: t_1(min)
Target: T_1

2016 blocks
Time it took to mine: t_2(min)
Target: T_2

2016 blocks
Time it took to mine: t_3(min)
Target: T_3

New target is not allowed to be more than 4x old target.
New target is not allowed to be less than ¼ x old target.

t_2: difference between the timestamps in B and A

t_3: difference between the timestamps in C and B

The Bitcoin Backbone Protocol with Chains of Variable Difficulty (2016)

Consensus in the Internet Setting

Characterized by *open participation*.

Challenges:

- Adversary can create many Sybil nodes to take over the protocol.
- Honest nodes can come and go at will.

Requirements:

Achieved by Bitcoin!

- Limit adversary's participation.
  - **Sybil resistance (e.g., Proof-of-Work)!**
- Maintain availability (liveness) of the protocol when the honest nodes come and go at will, resulting in changes in the number of nodes.
  - **Dynamic availability!**

Security?

Can we show that Bitcoin is a secure state machine replication (SMR) protocol (satisfies safety and liveness) under synchrony against a Byzantine adversary?

$\beta(t)$ $\in [0,1]$ for all t

Fraction of the mining power controlled by the adversary at time t.

What is the highest $\beta(t)$ for which Bitcoin is secure??

Model for Bitcoin

- Many different miners, each with *infinitesimal* power.

  Total mining rate (growth rate of the chain): λ (1/minutes). In Bitcoin, $\lambda = 1/10$.

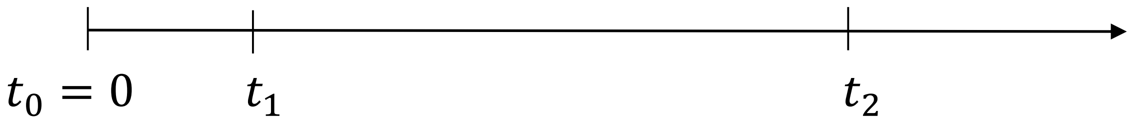
- Suppose Adversary is Byzantine and controls $\beta < \frac{1}{2}$ fraction of the mining power.
  - **Adversarial mining rate**: $\lambda_a = \beta\lambda$
  - **Honest mining rate**: $\lambda_h = (1 - \beta)\lambda$

- Network is **synchronous** with a known upper bound Δ on delay.

Reminder: Why is safety important?

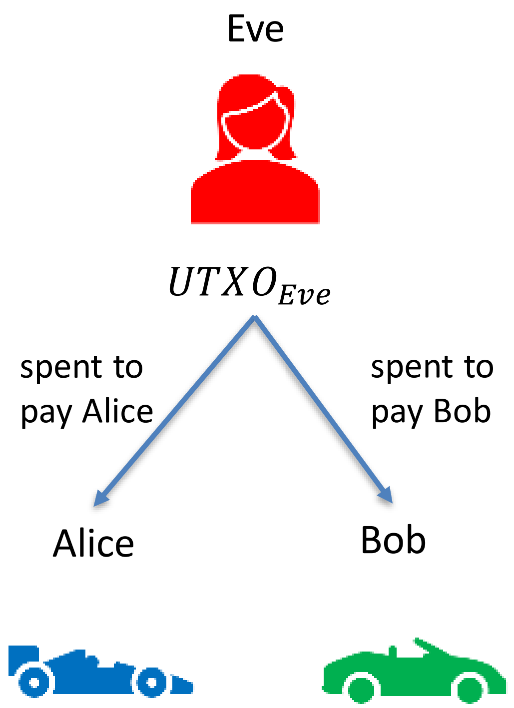
Suppose Eve has a UTXO.

- tx_1: transaction spending Eve's UTXO to pay to car vendor Alice.
- tx_2: transaction spending Eve's UTXO to pay to car vendor Bob.



- Alice's ledger at time t_1 contains tx_1: $LOG_{t_1}^{Alice} =< tx_1 >$
- Alice thinks it received Eve's payment and sends over the car.

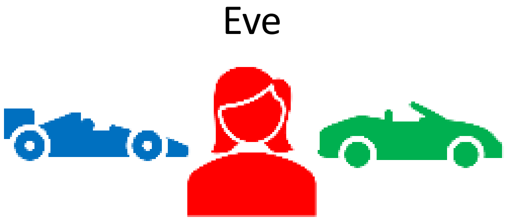
- Bob's ledger at time t_2 contains tx_2: $LOG_{t_2}^{Bob} =< tx_2 >$
- Bob thinks it received Eve's payment and sends over the car.

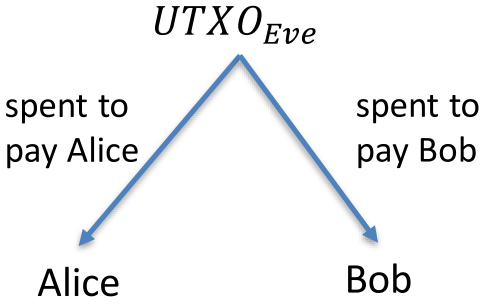


Reminder: Why is safety important?

Suppose Eve has a UTXO.

- tx_1: transaction spending Eve's UTXO to pay to car vendor Alice.
- tx_2: transaction spending Eve's UTXO to pay to car vendor Bob.





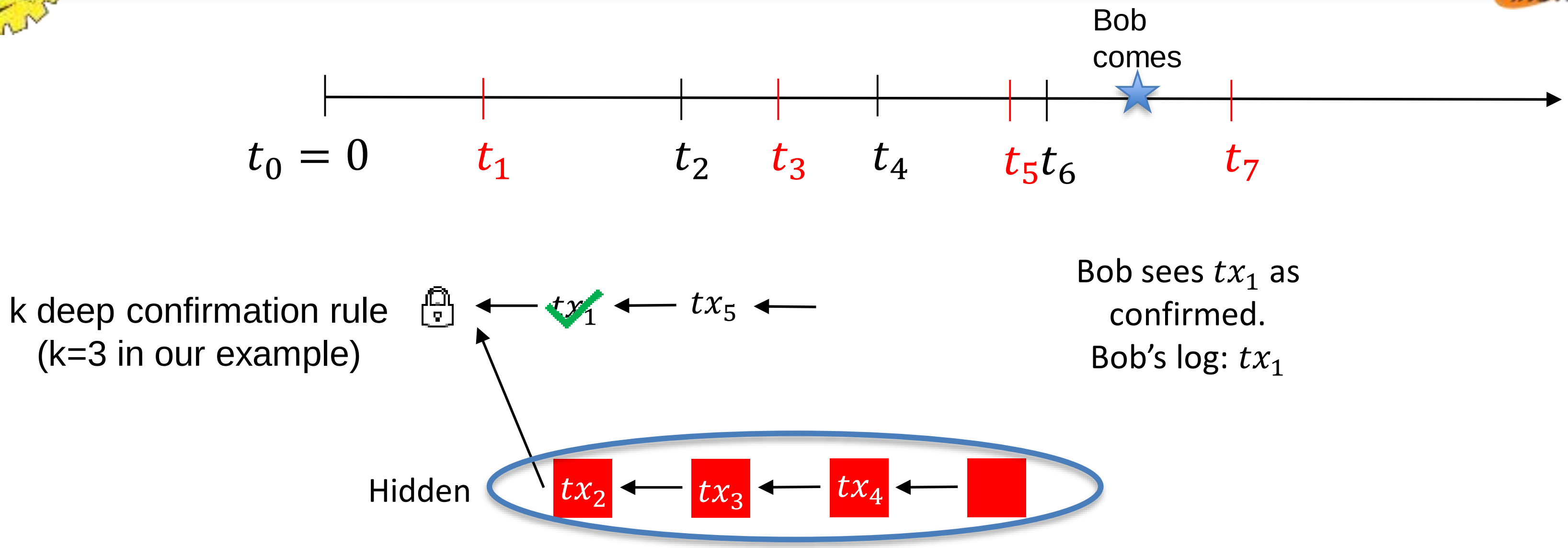
$t_0 = 0$ — t_1 — safety violation — t_2

- Alice's ledger at time t_1 contains tx_1: $LOG_{t_1}^{Alice} =< tx_1 >$
- Alice thinks it received Eve's payment and sends over the car.

- Bob's ledger at time t_2 contains tx_2: $LOG_{t_2}^{Bob} =< tx_2 >$
- Bob thinks it received Eve's payment and sends over the car.

When safety is violated, Eve can double-spend!

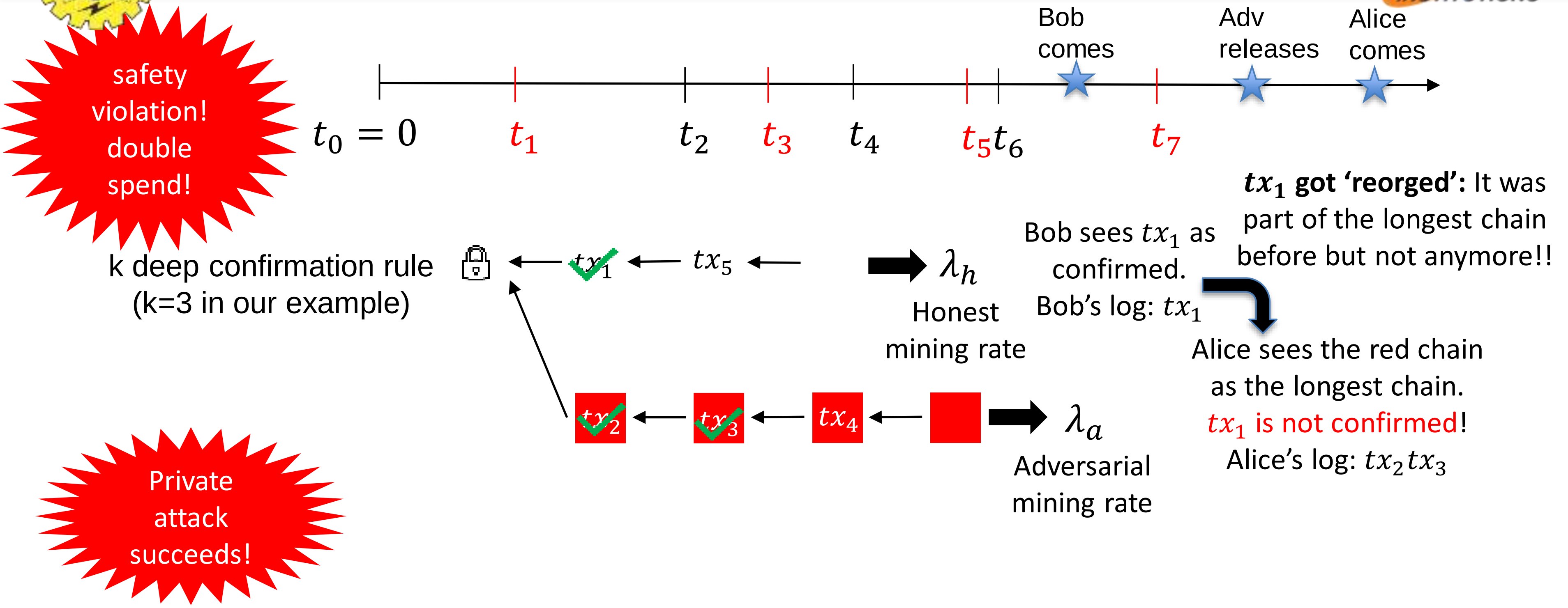
Nakamoto's Private Attack: $\beta \geq 1/3$



Let's show that Bitcoin is insecure if $\beta(t) \geq 1/2$

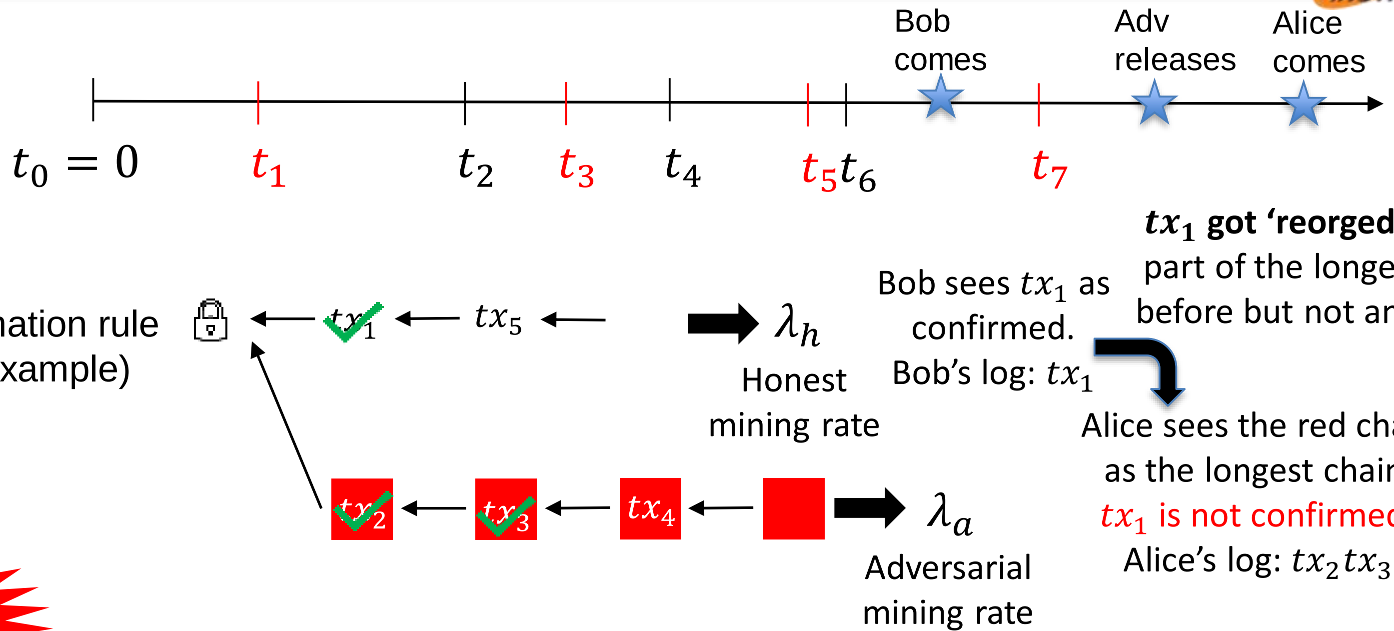
A Peer-to-Peer Electronic Cash System (2008)

Nakamoto's Private Attack: $\beta \geq 1/3$



Nakamoto's Private Attack: $\beta \geq 1/3$

safety violation! double spend!



k deep confirmation rule (k=3 in our example)

Bob sees tx_1 as confirmed. Bob's log: tx_1

tx_1 got 'reorged': It was part of the longest chain before but not anymore!!

Alice sees the red chain as the longest chain. tx_1 is not confirmed! Alice's log: $tx_2 tx_3$

Private attack succeeds!

Private attack (mostly) succeeds if $\lambda_a \geq \lambda_h$, i.e., if $\beta \geq 1 - \beta$, i.e., if $\beta \geq \frac{1}{2}$.

Private attack (mostly) fails if $\lambda_a < \lambda_h$, i.e., if $\beta < 1 - \beta$, i.e., if $\beta < \frac{1}{2}$.

Can another attack succeed?

A Peer-to-Peer Electronic Cash System (2008)