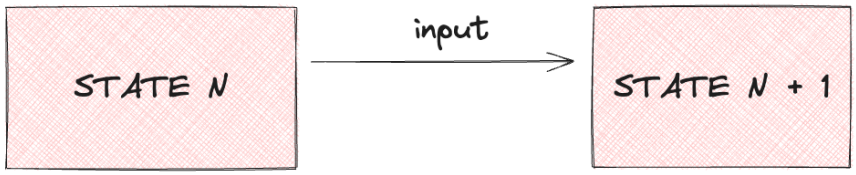
Blockchain Basic

Dang Quang Vu

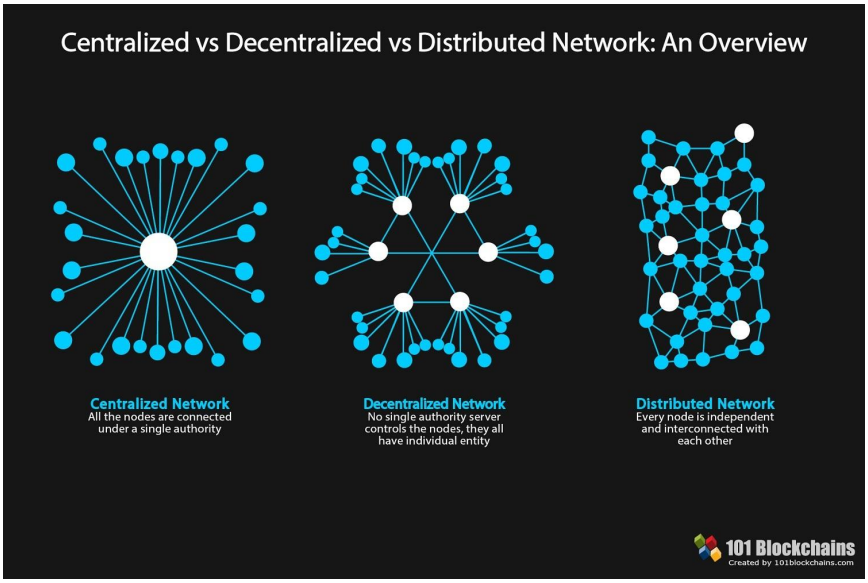
June 10, 2023

Contents

1 Program State

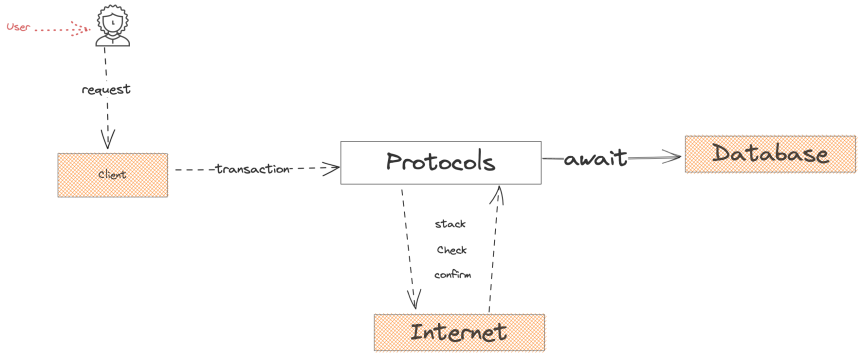


2 Overview Networks



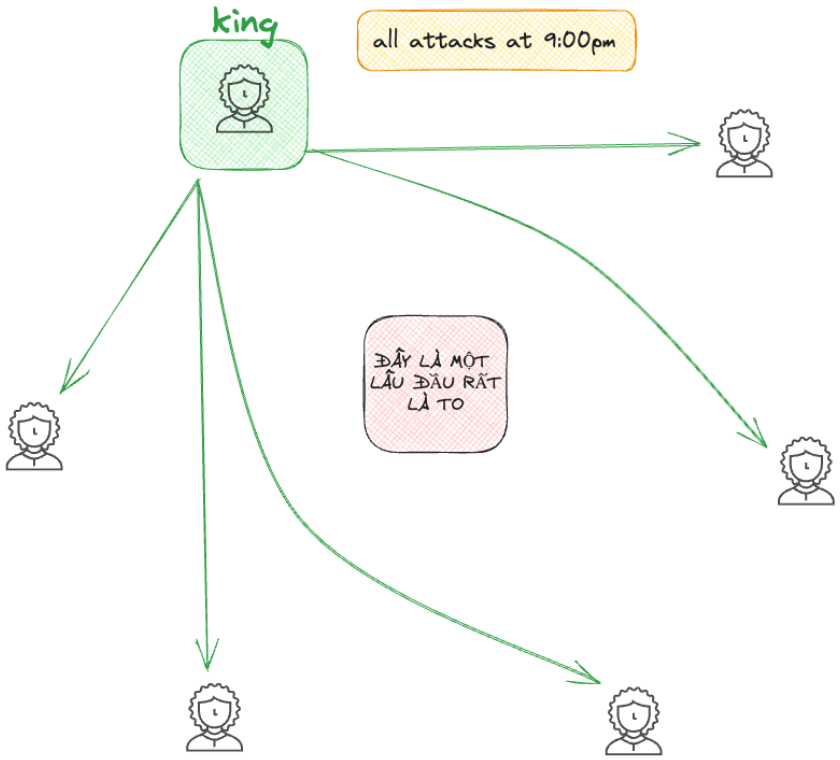
- Comparison Centralized & Decentralized & Distributed Networks

3 DApp Architecture

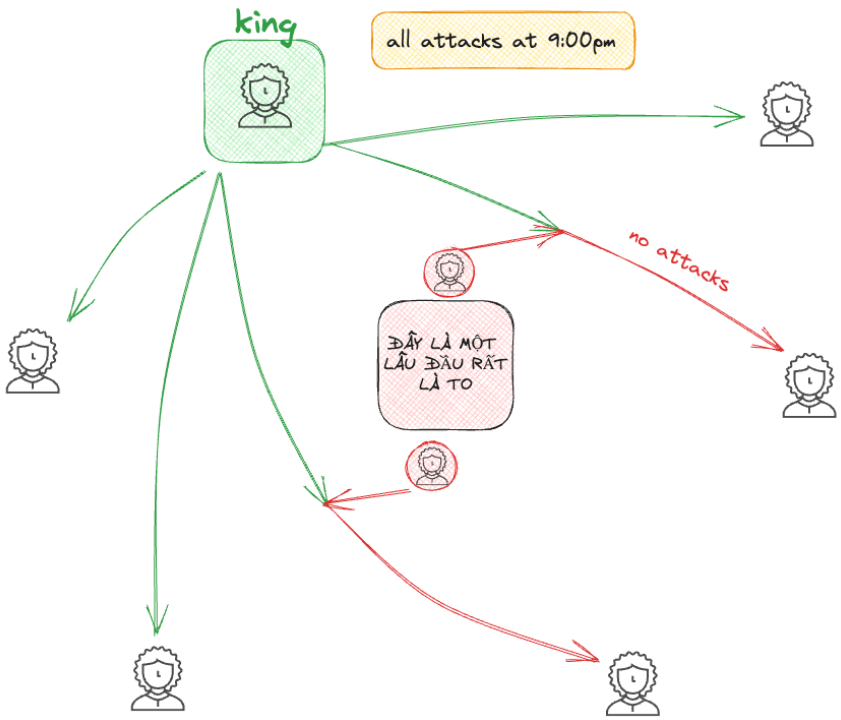


4 Byzantine General's Problem

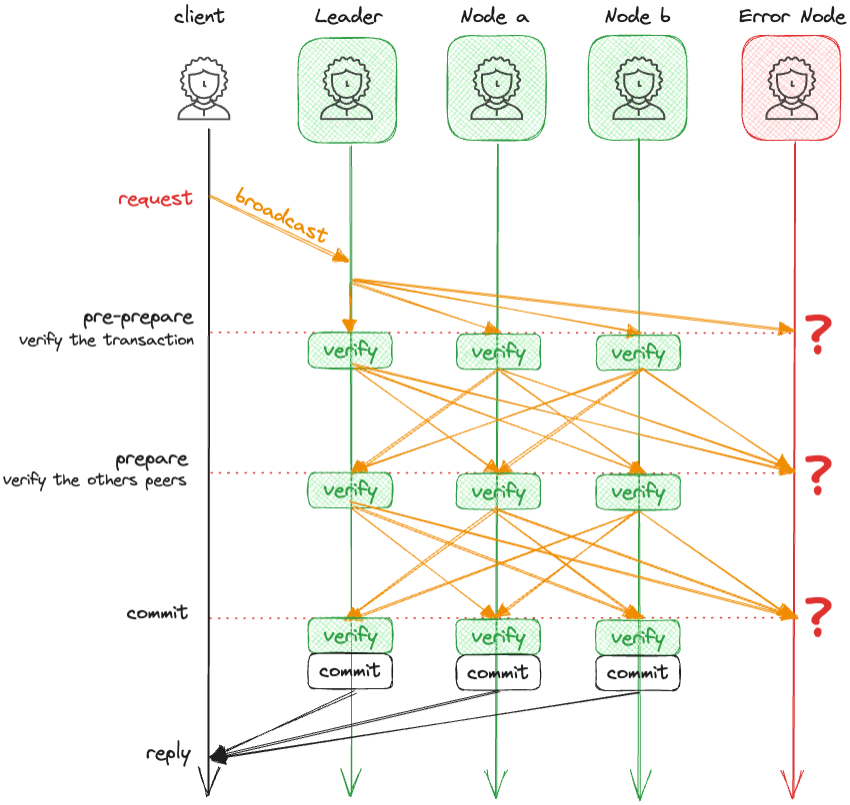
4.1 Image 01



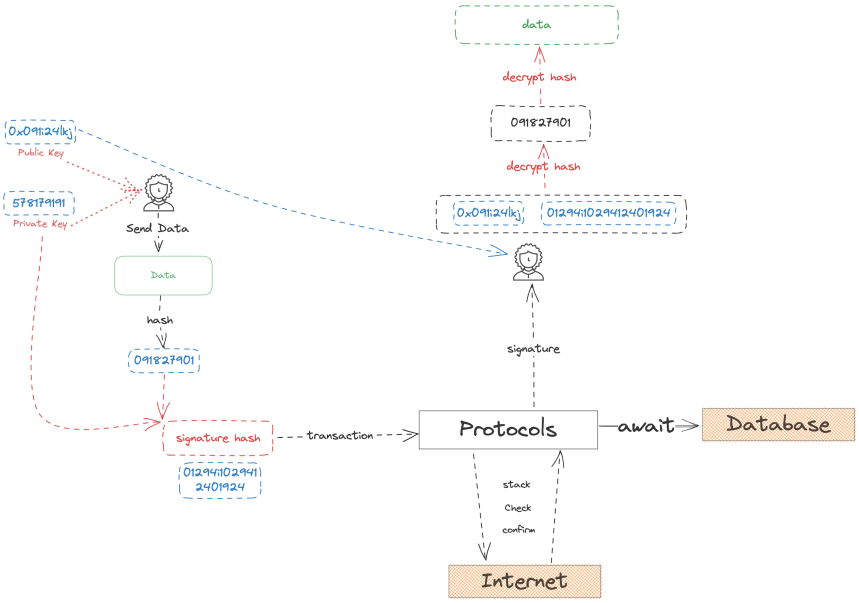
4.2 Image 02


king
all attacks at 9:00pm
no attacks
ĐÂY LÀ MỘT LẦU ĐẦU RẤT LÀ TO

4.3 Practical Byzantine Fault Tolerance

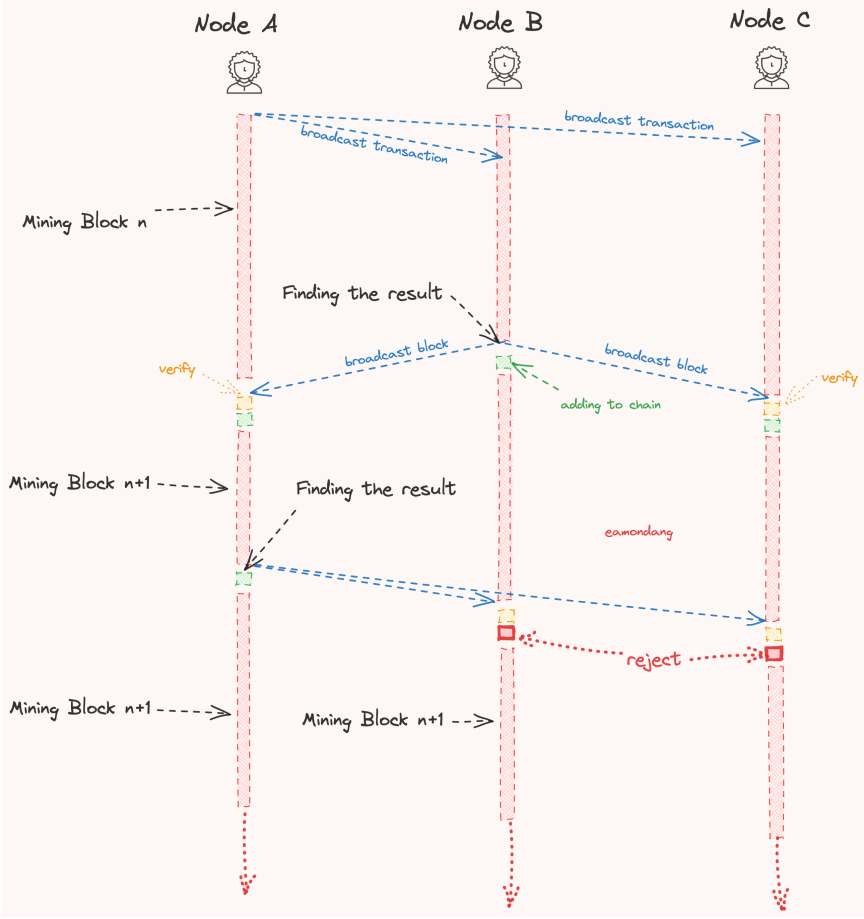


5 Transaction flow

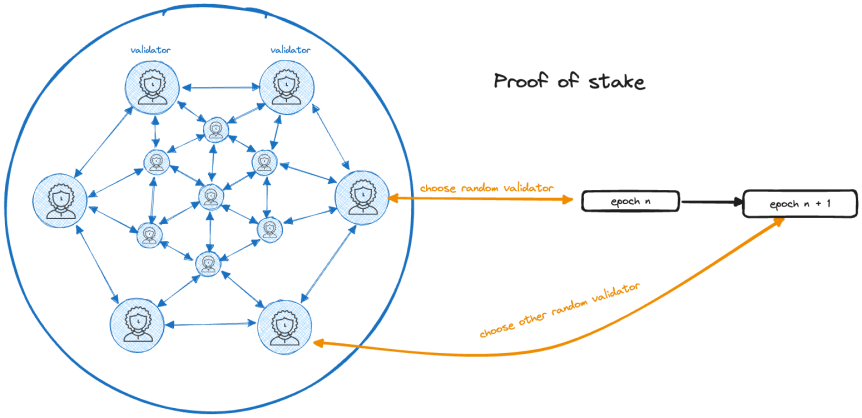


6 Consensus

6.1 Proof of Work



6.2 Proof of Stake



7 Requirements of Blockchain?

- High Availability
- Sustainability
- Irreversibility
- Transparency
- Append-only
- `Consensus`
- Security
- Global

8 Types of Blockchain

- Permissionless Blockchain - Public Blockchain
  - Anyone can access to write the data
  - Anyone can read
  - Ex: 99%
- Permissioned Blockchain - Private Blockchain

  - Participants need permission to accessing the data
  - Readers need permission to read
  - Ex: Hyperledger Fabric, Corda
- Consortium Blockchain
  - Participants need permission to accessing the data
  - Anyone can read
  - Ex: Ripple

9 What is Blockchain

- It can be thought of as `ONE BIG COMPUTER` made up of small computers around the world.
- All these computers (nodes) are connected to one another and have a full copy of the code and data.
- One of the best ways to understand blockchain is by comparing it with a `traditional client/server architecture`

10 Components of Blockchain

10.1 Data Models

- State Models
- Account Models
- Object Models

10.2 Decentralized Models

- Permissionless Blockchain
- Permissioned Blockchain
- Consortium Blockchain

10.3 Network Models

- Asynchronous
- Synchronous
- Partial Synchronous

10.4 `Consensus`

- PoW/PoS/DPoS
- Tendermint BFT
- Doomslug
- TowerBFT
- HotStuff
- Narwhall & bullshark
- AptosBFT
- Nominated BFT (GRANDPA & BABE)
- …etc

10.5 Cryptographic schemes

- Hashing
- Signature
- Merkle Tree
- Pub/Priv Key
- Zero-Knowledge Proofs

11 Decentralized Features

11.1 Level of Decentralization

- Trilemma
  - Security
  - Decentralized
  - Speed

11.2 Security

- Single Failure Tolerance **(Consensus)**
- Availability
- Sybil Attacks
- 51% Attacks

11.3 Performance

- Communicate - Broadcast Data
- Argreement among Participants

12 Example

12.1 Install

- Install WSL if use Window
- Install Node.js or Node Version Manager
- Install Rust Lang

```
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

- Install near-cli

```
npm install -g near-cli
```

13 Cons

- Very Slow
- Expensive
- High Latency