Blockchain for Industrial Engineers: Decentralized Application Development

บล็อกเชนสำหรับวิศวกรอุตสาหการ: การพัฒนาแอปพลิเคชันแบบกระจายศูนย์

ERC20 Token

Remaining implementation

- `transfer()` function does not work when tokens are being used to in more complex activities.
  - Trading
- ERC20 standard defines
  - `allowed` (mapping data structure)
  - `approve()` function
  - `transferFrom()` function
- These permit owner to give another address to transfer up to a number of tokens
  - `allowance`

```
    function allowance(address _owner, address _spender) external view returns (uint256 remaining);

    function approve(address _spender, uint256 _value) external returns (bool success);

    function transferFrom(address _from, address _to, uint256 _value) external returns (bool success);

    event Approval(address indexed _owner, address indexed _spender, uint256 _value);
```

Case study

- `A` has 100 tokens.
- `A` approves `B` to spend 50 tokens.
- `B` can now transfer up to 50 tokens to `C`.

Decentralized data storage

- DApps need to store/retrieve data in a decentralized manner.
- Saving data in the blockchain is very expensive.
- We need to save data **off-chain**.

IPFS

- *Interplanetary File System*
  - Decentralized off-chain solution for data storage.
- When we put data on IPFS, we obtain **content identifier (CID)** that uniquely indentifies the data.
  - For example, `QmTHUf5DiynRc5WRJBMcZYtigmWqkdtJVxuQVKoAjMPnoB`
  - We store CID on the ethereum block chain.
- Note: `Swarm` is an alternative to IPFS.

IPFS

- Aims to replace HTTP.
- Peer-to-peer (based on bit-torrent)
  - Network consists of multiple *nodes*.
- Data is permanent and cannot be deleted or modified
  - Data are already distributed to other nodes.

Content/Location addressing

- **Location addressing**
  - `https://cmu.ac.th/file.pdf`
  - Who ever controls that location control the content.
- **Content addressing**
  - Files are based on **where they are**, but on **what they are**.
  - There is no location of the files.
  - No one controls the files.

Pinning

- Mechanism that allows IPFS to always keep a given content and never remove it.

Gateway

- Allows browsers to access IPFS.
- **Local gateway** (need to run IPFS client)
  - `http://localhost:8080/ipfs/[CID]` (ex)
- **Private gateway**
- **Public gateway**
  - `https://ipfs.io/ipfs/[CID]` (ex)
  - `https://gateway.pinata.cloud/ipfs/[CID]` (ex)

Pinata

- IPFS pinning service.
- https://www.pinata.cloud/