Ethereum

Outline

- Introduction
- Differentiation
- Ether
- Accounts
- Transactions and Messages
- Gas
- Blocks
- Wallets

Ethereum - Introduction

- A blockchain network with smart contract functionality (via scripting)
- Permissionless, public, PoW-based blockchain
- Proposed (2013) by Vitalik Buterin
- Live since 2015
- Ether (ETH) is the cryptocurrency of the network
- 1 ETH varies around $2000 (currently) with high variability
- Current blockchain value exceeds $200 billion dollars
- More than 1 billion transactions have been confirmed
- 12-15 seconds block rate with ~15 TPS
- Current block formation reward is 2 ETH

Ethereum - Differentiation

- Bitcoin viewed as primarily a store of value (cryptocurrency) with no major changes
- Ethereum founders saw blockchain as base for multiple applications in various domains
- Focus on decentralized application development with flexible scripting capability
- Scripting flexibility allows development of new innovative applications
- Continuous planned protocol upgrades as hard forks
  - Hard forks - Breaks compatibility with old versions to allow major features
  - 8 planned and 3 unplanned upgrades done
  - Ethereum 2.0 planned for major upgrades, for tps increase and consensus mechanism change
- Detailed definition of the underlying computation machine - EVM
  - Described in Ethereum Yellow Paper
  - Implemented in multiple languages
- All computational efforts taxed by Gas (fee charged based on effort) to limit unnecessary computational load

Ether - The Currency

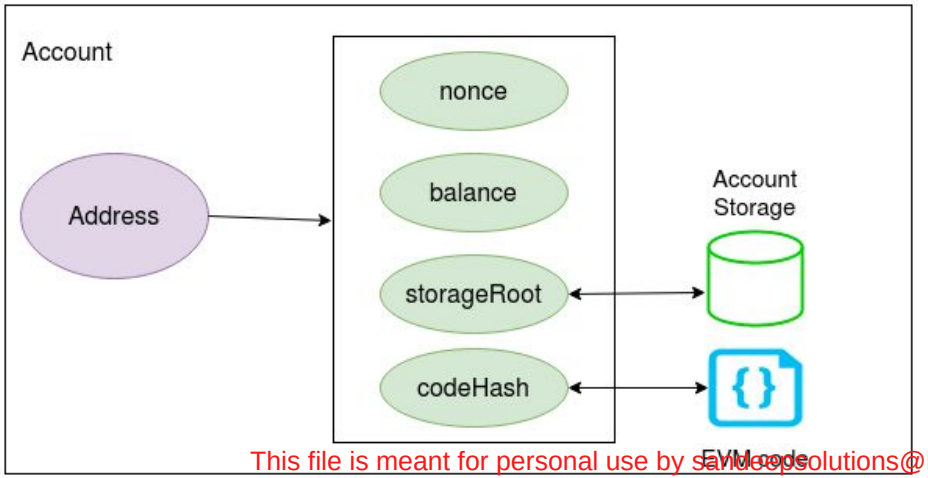
- Ether is the underlying currency for the whole Ethereum network
- Multiple subunits with wei being the lowest
  - 1 Wei = 10^{-18} Ether
  - 1 Gwei (Shannon) = 10^{-9} Ether
  - 1 Microether (Szabo) = 10^{-6} Ether
  - 1 Milliether (Finney) = 10^{-3} Ether
- Different subunits might be referred to in different contexts like rewards, gas, transactions
- Used for all payments - transaction fees, gas for computational resources, etc.

Ethereum - Accounts

- Two types - Externally Owned Accounts (EOAs) and Contracts
- Both can
  - send, receive, and store value (Ether and tokens)
  - interact with smart contracts
- EOAs
  - Free to create
  - Controlled by private keys
  - No associated code
  - Can initiate transactions to any account
  - Only ETH transfer between EOA<->EOA
- Contracts
  - Deployment has some associated cost
  - Can take action only in response to transactions/messages received from others
  - Contract to contract interaction - Messages or internal transactions
  - Requires gas to deploy and execute

Ethereum - Accounts

- Fields
  - nonce - Counter for number of transactions sent from EOA or number of contracts created by a contract
  - balance - Amount of wei (10^{-18} ETH) stored in the account
  - codeHash - hash of the EVM code for contract, hash of an empty string for EOA. Immutable once deployed
  - storageRoot - 256-bit root node hash of a Merkle Patricia Tree, hash of all storage in the account



Ethereum - Transactions and Messages

- Transactions
  - Cryptographically signed actions Initiated only by EOAs
  - Value transfer or contract creation
  - Follow the normal cycle - created, broadcasted, mined, validated, confirmed
  - Change the state of the EVM
  - Generally require a fee to be mined
  - Unlike bitcoin, it's a single interaction between two accounts
- Messages
  - Similar to transactions but initiated by contracts
  - Code 'calling' other code
  - Triggers a specific function in the recipient contract
  - Initiated by CALL or DELEGATECALL opcodes in the sender contract
  - Allows multi-option decision flow for complex considerations
  - Allows a possible upgrade path by proxying the actual logic
  - Can help separate storage and computation structures

Ethereum - Transactions and Messages

- Fields
  - *nonce*
    - Property of account, not the transaction
    - Number of transactions or contract creations from the sender account
    - Instrumental in ordering multiple transactions still to be mined
    - Prevents replay attacks - signed hash is different every time so same transaction can't be replayed by someone else
  - *to* - Recipient address
  - *v,r,s* - Used to generate signature to identify sender
  - *value* - Wei transferred to the receiver or initial value store in the contract
  - *data/init*
    - EVM code in contract creation
    - function call and parameters in message calls
  - *gasLimit* - Maximum units of gas that can be spent including all sub-executions
  - *gasPrice* - Per unit gas price defined

Ethereum - Gas

- Every computation in a transaction and contract execution has a cost
- Prevents overload attacks and mistakes
- Gas is the unit to measure the computations
- Typically measured in Gwei (10^{-9} Ether)
- Gas price is the price of each unit of gas, set by the sender
- Gas limit is the maximum number of units that can be spent, set by the sender
- Gas limit should cover all computational cost including all sub-executions
- Normal transaction requires 21,000 gas
- Smart contract execution might require much more, based on complexity
- gasLimit * gasPrice has to be pre-loaded.
- If less is spent, unused gas value is returned back to the sender
- If gas limit is reached before execution finishes, the transaction is rolled back
- Higher gas price you set, better the chance of transaction pick-up

Ethereum - Blocks

- Similar to bitcoin, a block is mined based on Proof-of-work protocol
- Includes
  - Block header
  - Transaction list with details
  - Block headers from ommer blocks
    - Ommer/Uncle blocks are valid blocks created at the same time as the main accepted block
    - They are not part of the main chain but are stored since they are rewarded a small amount
- Bounded in size by an overall limit on the gas limit that a block can have
  - Currently around 15 million gas units
- Much smaller than bitcoin but generated much more frequently
- Each transaction generates a receipt as a response which is also stored
- logsBloom allows storage of logged events by contracts in a bloom filter

Ethereum - Blocks

- Block header
  - *parentHash* - Previous block's hash
  - *timestamp* - unix timestamp
  - *beneficiary* - miner account who received the reward
  - *difficulty* - difficulty level of the block
  - *number* - count of current block, genesis block being 0
  - *gasLimit* - Gas limit that a block can have including all transactions and resulting computations
  - *gasUsed* - Total gas actually used in block creation
  - *stateRoot* - Root hash of the state information
  - *transactionsRoot* - Root hash of the transactions
  - *receiptsRoot* - Root hash of the transaction receipts
  - *ommersHash* - Hash of list of ommer blocks
  - *logsBloom*
  - *extraData*
  - *mixHash*
  - *nonce*

Ethereum - Wallets

- Basis of any blockchain account is a private/public key pair
- Cryptocurrency Wallets allow
  - storing the keys for availability from different machines
  - redundant storage to keep key backups
  - secure access to various blockchain accounts without mandatory local storage
  - additional services like transaction creation and signage, currency exchange, etc.
- Multisignature wallets - Needs multiple parties to sign a transaction forcing a joint agreement for transaction initiations
- Types
  - Deterministic - All keys generated from a seed (a long phrase in practice), allowing for complete key recreation just from the seed
  - Non-deterministic - Each pair is independent and necessitates storing all of them
- Forces trust on the wallet provider, examples of embezzlement and loss

greatlearning

Power Ahead

Happy Learning !