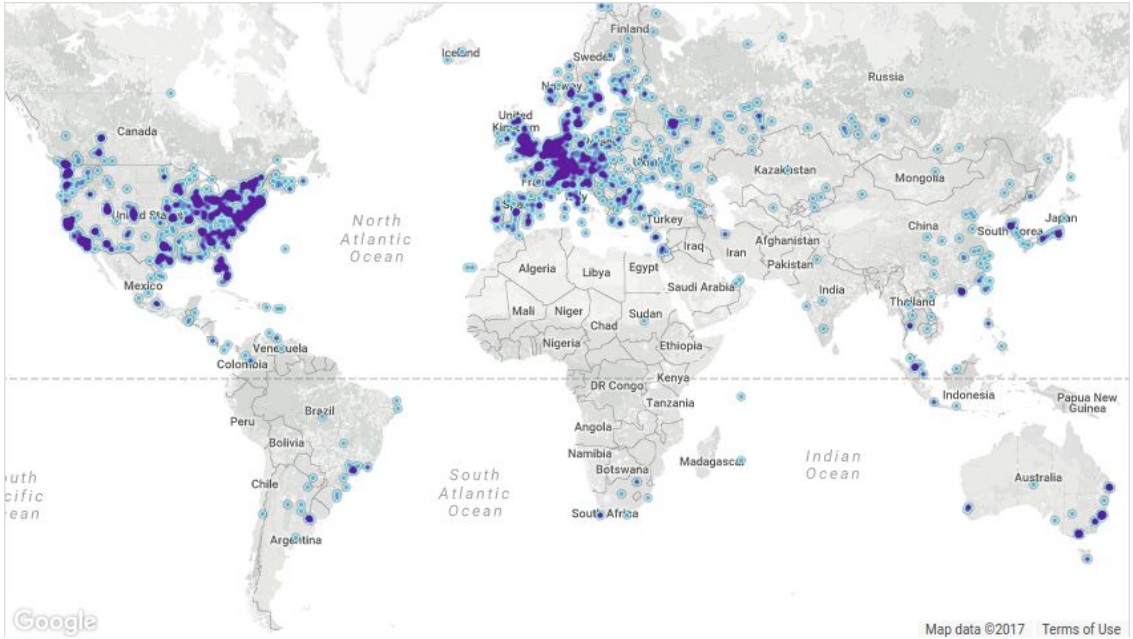
Introduction to Blockchain & Ethereum

How does a normal payment system work?

Account	A	B	C	D
Initial	75	60	10	100
Transfer	-50	+50		
Transfer		+15	+15	
Transfer	-25			+25
Transfer		-40	+40	
~~Transfer~~			~~+200~~	~~-200~~
Transfer	+50	-50		

Distributed Ledger

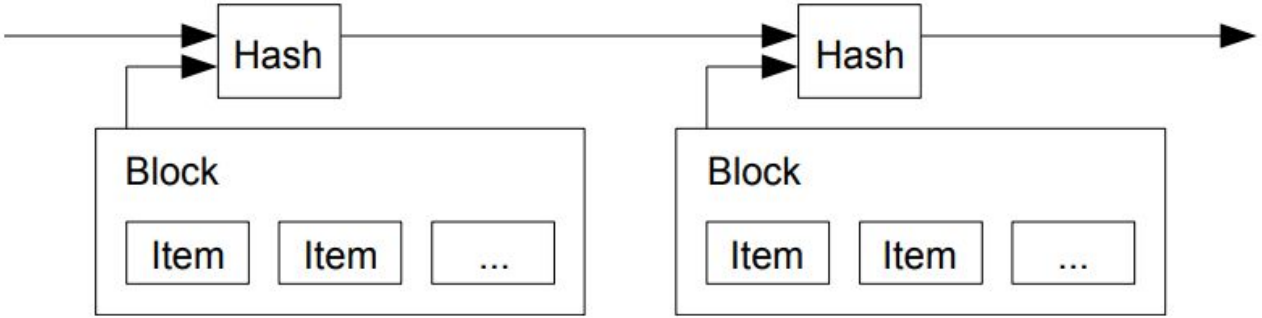


Hashing

```
$ python hash_example.py

I am Satoshi Nakamoto0 => a80a81401765c8eddee25df36728d732...
I am Satoshi Nakamoto1 => f7bc9a6304a4647bb41241a677b5345f...
I am Satoshi Nakamoto2 => ea758a8134b115298a1583ffb80ae629...
I am Satoshi Nakamoto3 => bfa9779618ff072c903d773de30c99bd...
I am Satoshi Nakamoto4 => bce8564de9a83c18c31944a66bde992f...
I am Satoshi Nakamoto5 => eb362c3cf3479be0a97a20163589038e...
I am Satoshi Nakamoto6 => 4a2fd48e3be420d0d28e202360cfbaba...
I am Satoshi Nakamoto7 => 790b5a1349a5f2b909bf74d0d166b17a...
I am Satoshi Nakamoto8 => 702c45e5b15aa54b625d68dd947f1597...
I am Satoshi Nakamoto9 => 7007cf7dd40f5e933cd89fff5b791ff0...
I am Satoshi Nakamoto10 => c2f38c81992f4614206a21537bd634a...
I am Satoshi Nakamoto11 => 7045da6ed8a914690f087690e1e8d66...
I am Satoshi Nakamoto12 => 60f01db30c1a0d4cbce2b4b22e88b9b...
I am Satoshi Nakamoto13 => 0ebc56d59a34f5082aaef3d66b37a66...
I am Satoshi Nakamoto14 => 27ead1ca85da66981fd9da01a8c6816...
I am Satoshi Nakamoto15 => 394809fb809c5f83ce97ab554a2812c...
I am Satoshi Nakamoto16 => 8fa4992219df33f50834465d3047429...
I am Satoshi Nakamoto17 => dca9b8b4f8d8e1521fa4eaa46f4f0cd...
I am Satoshi Nakamoto18 => 9989a401b2a3a318b01e9ca9a22b0f3...
I am Satoshi Nakamoto19 => cda56022ecb5b67b2bc93a2d764e75f...
```

Blockchain



Distributed Ledger

Account	A	B	C	D
Initial	75	60	10	100
Transfer	-50	+50		
Transfer		+15	+15	
Transfer	-25			+25
Transfer		-40	+40	
Transfer			+100	-100
Transfer	+50	-50		
Transfer	-30			+30
Transfer	-20	+20		
Transfer		+65	-65	
Transfer			-50	+50
Transfer	+20	-20		
Transfer		+30	-30	

Hash

Hash

Hash

Hash

Block Sample

version	02000000
previous block hash (reversed)	17975b97c18ed1f7e255adf297599b55 330edab87803c8170100000000000000
Merkle root (reversed)	8a97295a2747b4f1a0b3948df3990344 c0e19fa6b2b92b3a19c8e6badc141787
timestamp	358b0553
bits	535f0119
transaction count	63
transaction	
...	

Proof-of-Work

Example	428e5c057fd2030c8858d04bd9ede89a
Example0	a98ede9db40d8588c0302df750c5e824
Example1	ef7719f731431f2985916e5e1d9cecd1
Example2	cffab4b95a6c8eaee20e11ed79bddda8
Example3	00f7c3d939685b664e1d8a8d99a952a4

Example of an Actual Block

version	02000000
previous block hash (reversed)	17975b97c18ed1f7e255adf297599b55 330edab87803c8170100000000000000
Merkle root (reversed)	8a97295a2747b4f1a0b3948df3990344 c0e19fa6b2b92b3a19c8e6badc141787
timestamp	358b0553
bits	535f0119
nonce	48750833
transaction count	63
coinbase transaction	
transaction	
...	

Block hash

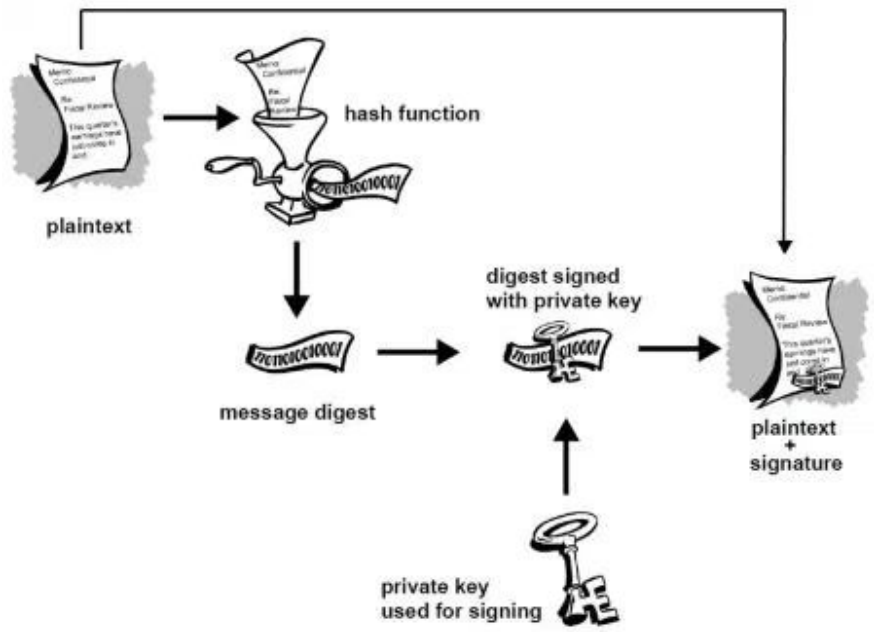
0000000000000000 e067a478024addfe cdc93628978aa52d 91fabd4292982a50

PGP

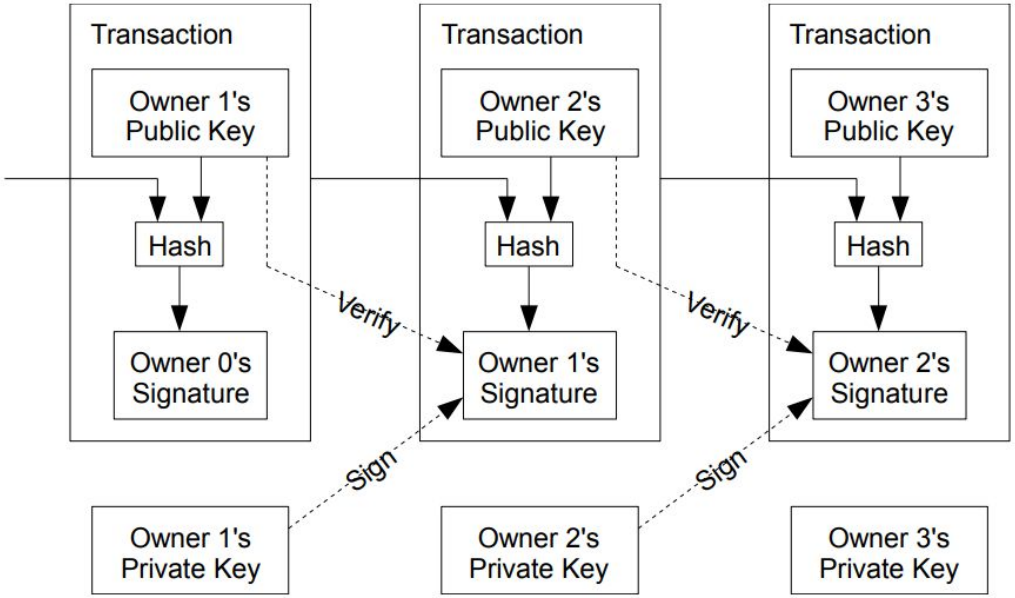
Private Key: A secret key that you don’t share

Public Key: Key that you share with others

Digital Signatures



Transactions



https://chrispacia.wordpress.com/2013/09/07/bitcoin-cryptography-digital-signatures-explained/

Steps for Running the Network

1.) Users broadcast their transactions to the network who give these to miners.
2.) Miners receive these transactions and include them in their next block and attempt to find a nonce.
3.) When a miner succeeds in finding a proof-of-work, they broadcast their block to the network. Nodes on the network verify that all transactions are valid and then broadcast it to other nodes till all nodes accept the block.
4.) Miners then repeat the process and use the previous block hash in their new block.

https://bitcoin.org/bitcoin.pdf

Ethereum

https://github.com/ethereum/wiki/wiki/White-Paper

Traditional Blockchains

-cryptocurrencies like bitcoin are only really meant for payment

-blockchains like namecoin can do other things but are still highly specialized

Ethereum

-Meant as a general usage platform upon which other applications can be built

-Ethereum Virtual Machine (EVM) is the core of the platform

-can deploy programs of arbitrary complexity to the EVM

http://ethdocs.org/en/latest/introduction/what-is-ethereum.html

Ethereum Virtual Machine (EVM)

-”World Computer”

-code deployed to the EVM is run by every node on the network

-allows for:

    -extreme fault-tolerance

    -zero down-time

    -censorship-resistant and immutable

Smart Contracts

-fully-autonomous code that cannot be retracted once deployed

-can do things such as reads, writes, computations, send messages to other contracts, store data

Gas

-as all smart contracts deployed to the network are also pushed to every node, there must be some limiting mechanism

-'Gas' is the fee for both smart contracts and transfers

-each operation done by a smart contract has a fee in gas

-each unit of gas must be paid in ether based on a gas/Ether price conversion

-usage as gas is the primary purpose of ethereum

https://blog.zeppelin.solutions/the-hitchhikers-guide-to-smart-contracts-in-ethereum-848f08001f05

Examples of Basic Smart Contracts

-Proof-of-Existence

-Creation of a token

-Secure multi-party fund

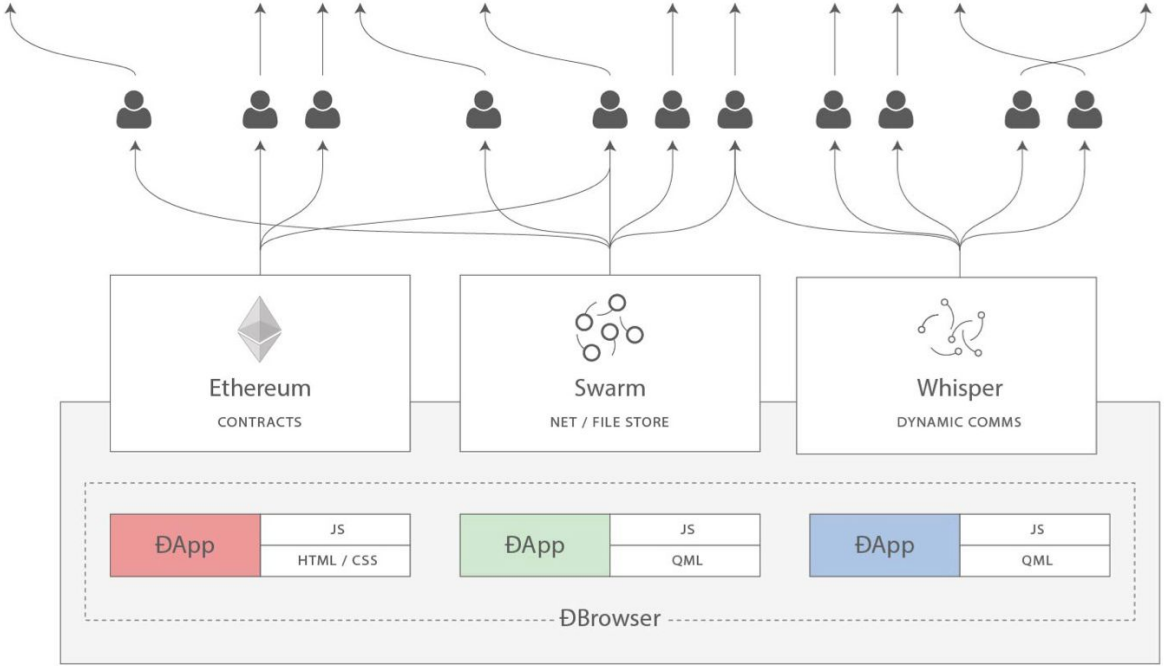
-Voting mechanisms

```
contract MyToken {
    /* This creates an array with all balances */
    mapping (address => uint256) public balanceOf;

    /* Initializes contract with initial supply tokens to the creator of the contract */
    function MyToken(
        uint256 initialSupply
        ) {
        balanceOf[msg.sender] = initialSupply;              // Give the creator all initial tokens
    }

    /* Send coins */
    function transfer(address _to, uint256 _value) {
        require(balanceOf[msg.sender] >= _value);           // Check if the sender has enough
        require(balanceOf[_to] + _value >= balanceOf[_to]); // Check for overflows
        balanceOf[msg.sender] -= _value;                    // Subtract from the sender
        balanceOf[_to] += _value;                           // Add the same to the recipient
    }
}
```

DApps



https://blog.ethereum.org/2014/08/18/building-decentralized-web/

Swarm

-meant for decentralized storage

-p2p, ddos-resistant, zero-downtime, fault-tolerant, and censorship-resistant

-smart-contracts have high gas costs for storing large amounts of information

-swarm is meant to work alongside smart-contracts for storing data as well as for hosting

-beta version available now, currently under active development

-built in incentive system for users to participate

-IPFS

http://swarm-guide.readthedocs.io/en/latest/introduction.html

Whisper

-meant to allow for communication between DApps/users

-still in early alpha

https://github.com/ethereum/wiki/wiki/Whisper

Examples of DApps

-provably fair gambling service/lottery

-DAO

-decentralized social network

-reddit alternative with currency as a motive

-decentralized cryptocurrency exchange

-DAPPS

Issues

-code can’t be changed easily after it’s deployed

-network congestion (ICOs)

-Disagreement on hard fork

Getting Access to Cryptocurrencies

Buying Cryptocurrencies

-in Korea, a couple options for online exchange are:

-Bithumb

-Korbit

-Coinone

Buying Cryptocurrencies

-localbitcoins.com

-http://bitcoincenterkorea.org/

-Buying from a friend

Converting Cryptocurrencies

-shapeshift.io

Crypto Wallets

-If you have the storage space and are planning to do dev work, mist is your best option for ethereum and ethereum tokens

-Otherwise, myetherwallet is a secure and trustable option

-For Bitcoin, electrum is the best open-source light-wallet

-If you're willing to take a bit of a risk, Exodus.io is a fantastic wallet with support for many currencies

-Most secure option (second to an offline computer/paper wallet) is a hardware wallet like the Ledger Nano S or Trezor

Investment Opportunities

-ICOs

-Trading

-Holding