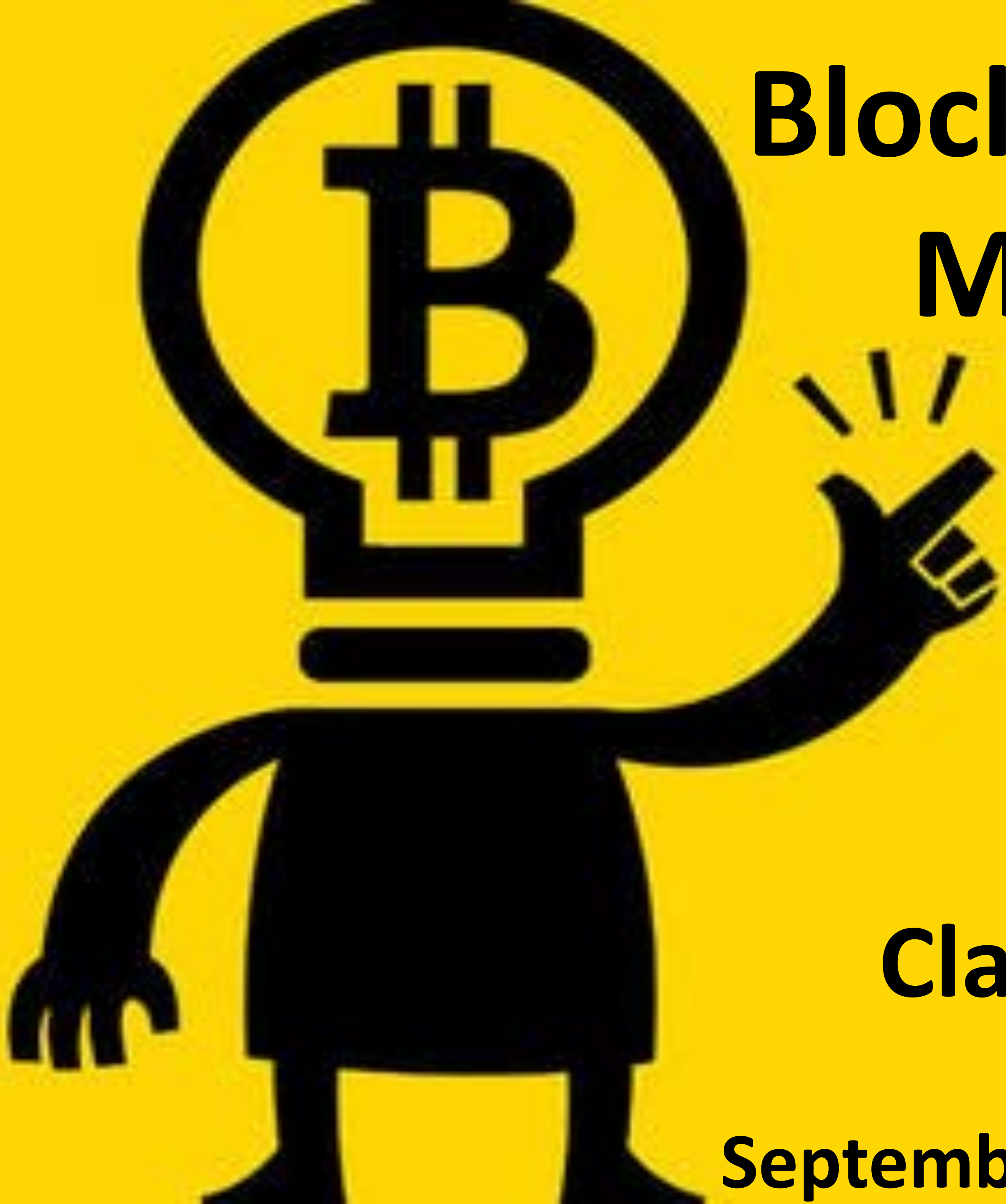


Blockchain & Money

Class 3

September 13, 2018

Class 3 (9/13): Study Questions

- What are the design features – cryptography, append-only timestamped blocks, distributed consensus algorithms, and networking - of Bitcoin, the first use case for blockchain technology?

- What are cryptographic hash functions, asymmetric cryptography and digital signatures? How are they utilized to help make blockchain technology verifiable and immutable?

- What is the double-spending problem and how it is addressed by blockchain technology?

Class 3 (9/13): Readings

- ‘*Bitcoin: A Peer-to-Peer Electronic Cash System*’ Nakamoto
- *‘Blockchain Technology Overview’* NIST (pages 9 – 23, sections 1 & 2)
- *‘Blockchain 101 – A Visual Demo’* Brownworth

Class 3 Overview

- Review of Class 2

- Bitcoin Design Features
- Cryptographic Hash Functions
- Timestamped Append-only logs
- Block Headers & Merkle Trees
- Asymmetric Cryptography & Digital Signatures
- Bitcoin Addresses

- Conclusions

Class 2 Review

- Money is a Social & Economic Consensus
- Fiat Money is but the Current Lead in a long Evolution of Money

- Fiat Currency has had Challenges & Instabilities as well

- Ledgers are a method for Recording Economic Activity & Financial Relationships
- Central Banking and Financial Sector are built upon a series of Ledgers

- We now Live in an Electronic Currency Age
- Many Efforts have been made at Cryptographic Digital Currencies

- Nakamoto's 'Bitcoin: A Peer to Peer Electronic Cash System' paper & related Blockchain Technology builds upon the long history of Money & Ledgers

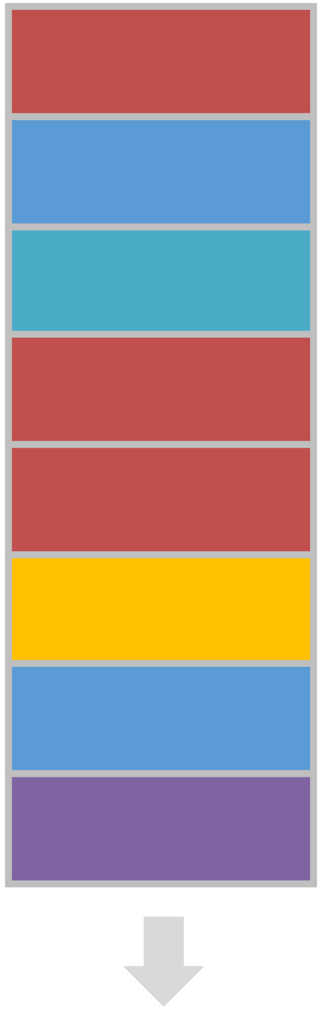
Bitcoin: A Peer-to-Peer Electronic Cash System

- From: Satoshi Nakamoto <satoshi <at> vistomail.com>
  Subject: Bitcoin P2P e-cash paper
  Newsgroups: gmane.comp.encryption.general
  Date: Friday 31st October 2008 18:10:00 UTC

- “I've been working on a new electronic cash system that's fully peer-to-peer, with no trusted third party.”

Blockchain Technology

timestamped
append-only log



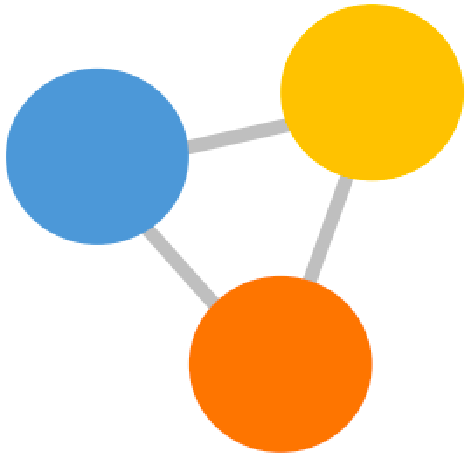
auditable database

Secured via cryptography

- Hash functions for **tamper resistance** and **integrity**
- Digital signatures for **consent**

Consensus for **agreement**

network consensus protocol



Addresses ‘**cost of trust**’ (Byzantine Generals problem)

- Permissioned
- Permissionless

Bitcoin – Technical Features

- Cryptographic Hash Functions
- Timestamped Append-only Logs (Blocks)
- Block Headers & Merkle Trees
- Asymmetric Cryptography & Digital Signatures
- Addresses

- Consensus through Proof of Work
- Network of Nodes
- Native Currency

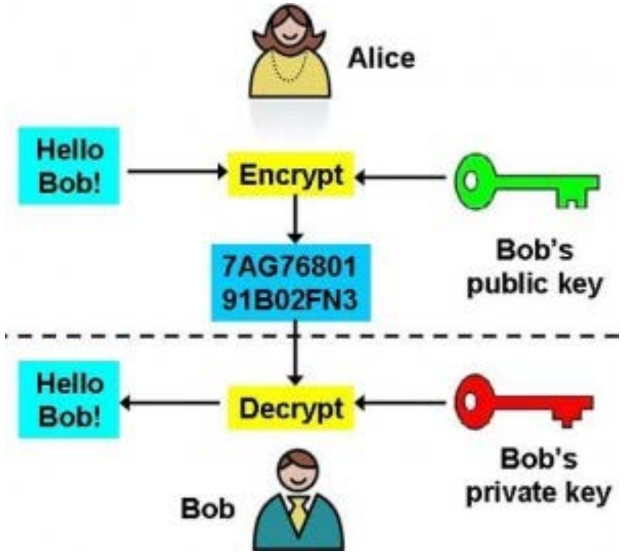
- Transaction Inputs & Outputs
- Unspent Transaction Output (UTXO)
- Scripting language

Cryptography:

Communications in the presence of adversaries

Scytale Cipher
Ancient Times

Enigma Machine
1920s - WWII



Asymmetric Cryptography
1976 to today

Cryptographic Hash Functions

Digital Fingerprints for Data

- General Properties
  - Maps Input **x** of any size to an Output of fixed size – called a 'Hash'
  - Deterministic: Always the same Hash for the same **x**
  - Efficiently computed

- Cryptographic Properties
  - Preimage resistant (One way): infeasible to determine **x** from Hash(x)
  - Collision resistant: infeasible to find and **x** and **y** where Hash(**x**) = Hash(**y**)
  - Avalanche effect: Change **x** slightly and Hash(**x**) changes significantly
  - Puzzle friendliness: knowing Hash(**x**) and part of **x** it is still very hard to find rest of **x**

Cryptographic Hash Functions

Digital Fingerprints for Data

- Uses as
  - Names
  - References
  - Pointers
  - Commitments

- Bitcoin Hash Functions
  - Headers & Merkle Trees – SHA 256
  - Bitcoin Addresses – SHA 256 and RIPEMD160

‘How to Time-Stamp a Digital Document’

Habor & Stornetta (1991)

Surety 1995 - present

NOTICES & LOST AND FOUND
(5100-5102)

Universal Registry Entries:
Zone 2 -

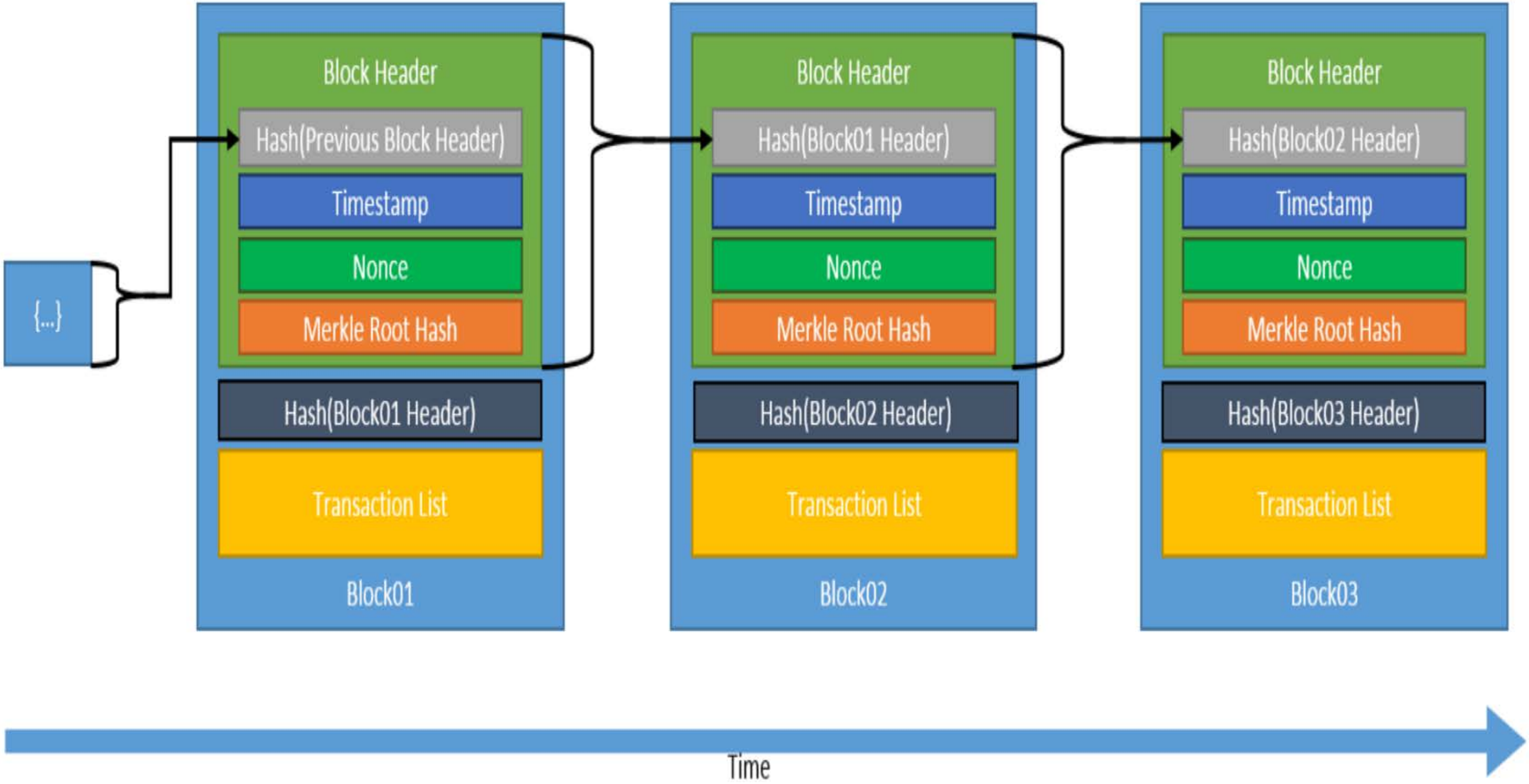
dS8492cgVOFAoP9kyE1XzMOrQ
HgEwzkVbVafNylkUz99qvq8/ME
p5y9EFSG8XxzMBaIGQQ==

Zone 3 -

JnFCg+HCmvhj8GmmUP7VZna71
NgZup/RfuKUQNzCHWXMuqLK
durxHQV5pSHLqBGPRly+mg==

These base64-encoded values represent the combined fingerprints of all digital records notarized by Surety between 2009-06-03Z 2009-06-09Z.
www.surety.com 571-748-5800

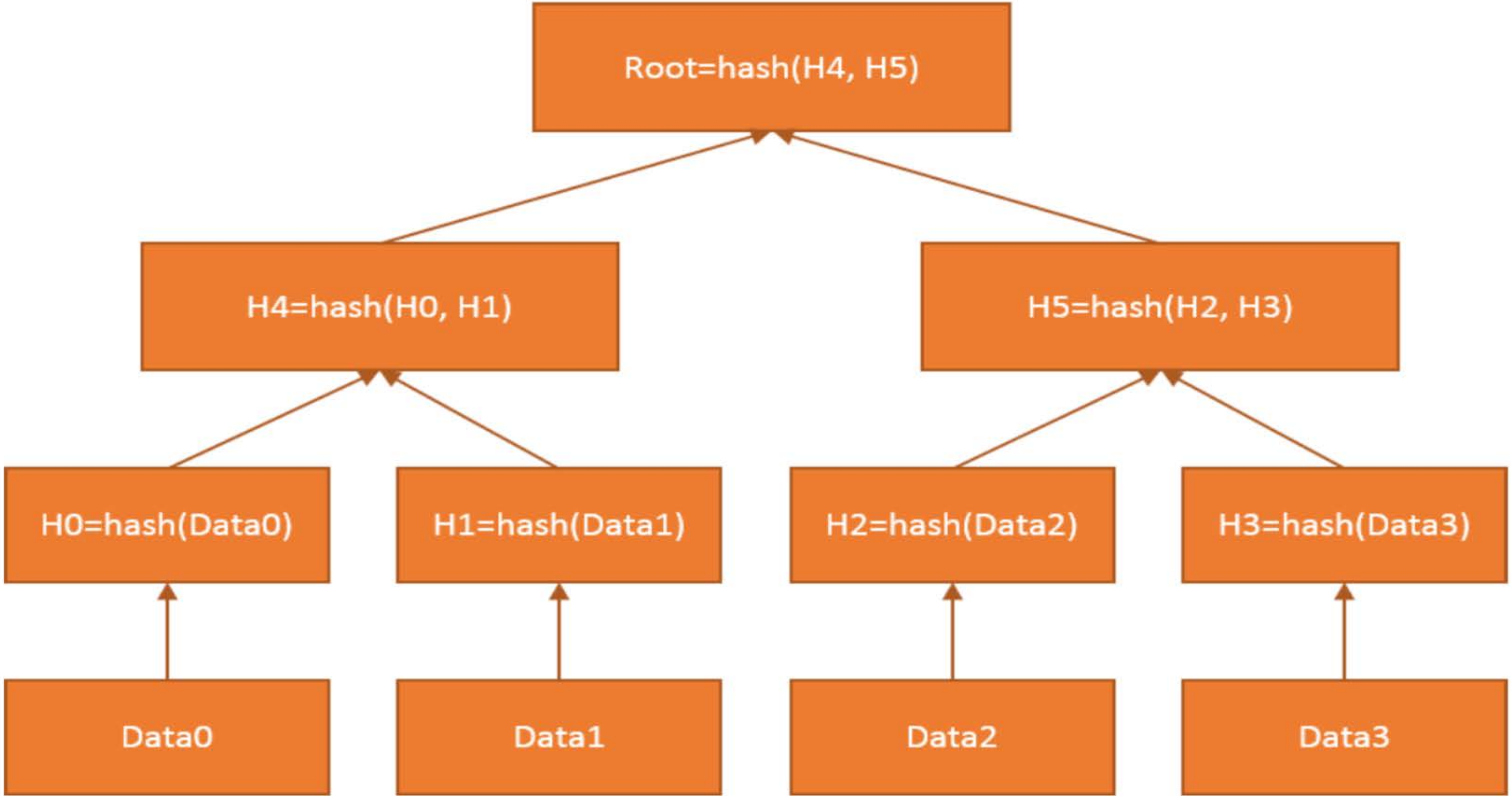
Timestamped Append-only Log - Blockchain



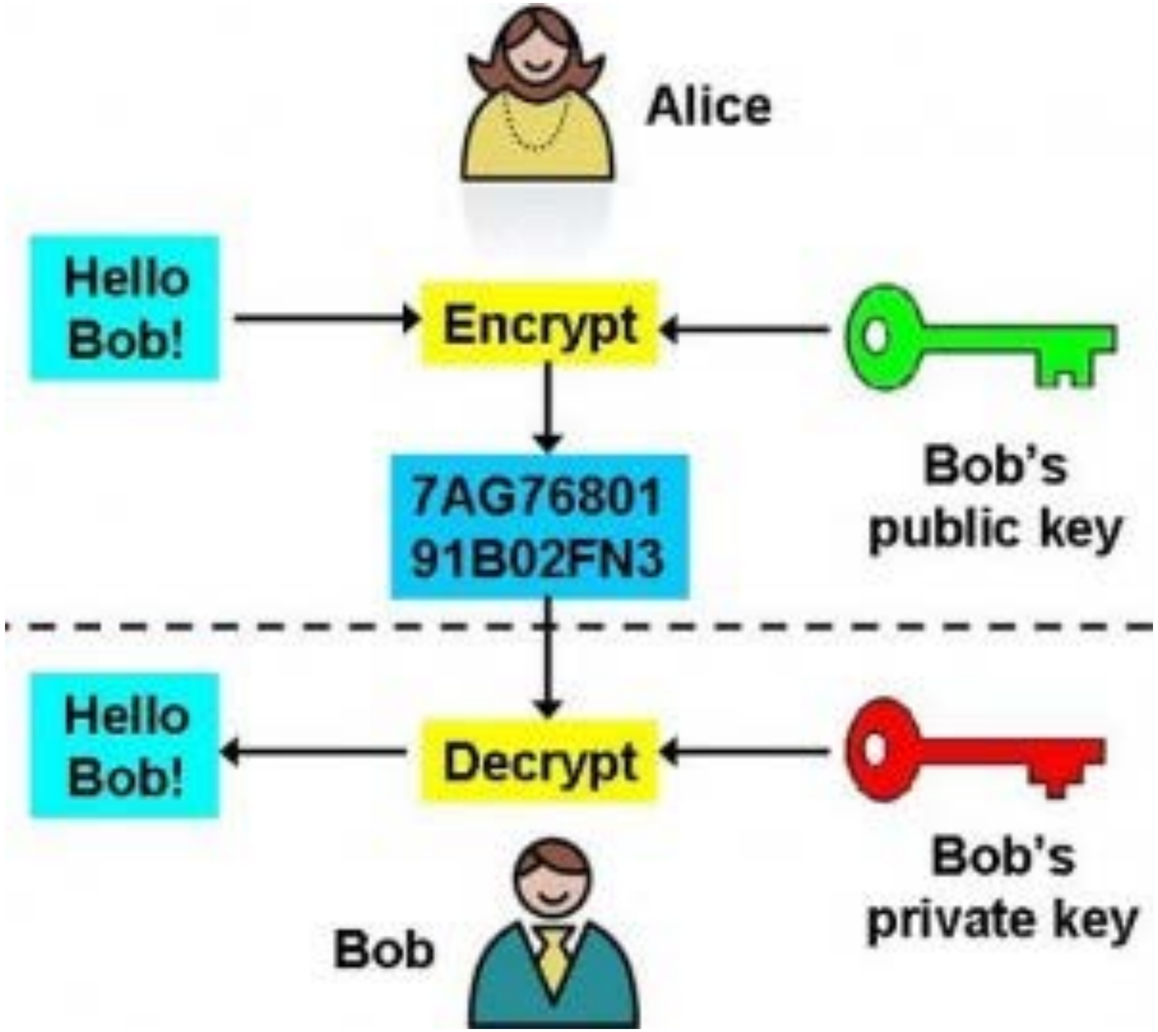
Block Header

- Version
- Previous Block hash
- Merkle Root hash
- Timestamp
- Difficulty target
- Nonce

Merkle Tree – Binary Data Tree with Hashes



Asymmetric Cryptography & Digital Signatures



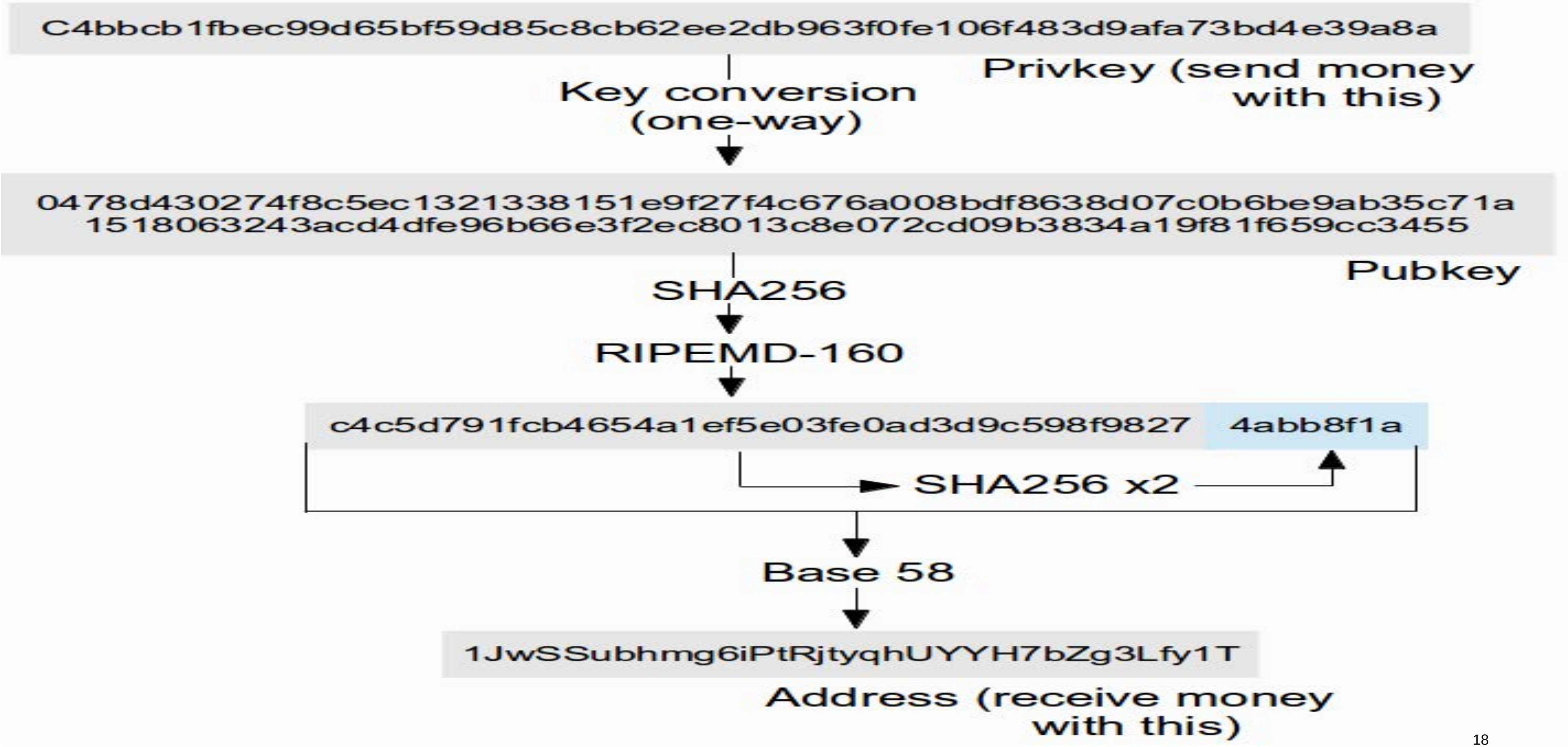
Asymmetric Cryptography & Digital Signatures

- Digital Signature Algorithms
  - Generate Key Pair - Public Key (**PK**) & Private Key (**sk**) - from random number
  - Signature – Creates Digital Signature (**Sig**) from message (**m**) and Private Key (**sk**)
  - Verification – Verifies if a signature (**Sig**) is valid for a message (**m**) and a Public Key (**PK**)

- Properties
  - Infeasible to find Private Key (**sk)** from Public Key (**PK**)
  - All valid signatures verify
  - Signatures infeasible to forge

- Bitcoin Digital Signature Function
  - Elliptic Curve Digital Signature Algorithm (EDCSA) … y2 = x3 + 7

Bitcoin Addresses



Deposits & Negotiable Orders

Images are in the public domain.

Transaction format

Input	Output
Previous transaction ID	Value
Index	Public Key (Bitcoin Address)
Signature	
lock_time	

Uniquely identifies an output

In satoshis
$10^8 = 1$ bitcoin

A "coin"

Class 4 (9/17): Study Questions

- What is the Byzantine Generals problem? How does proof-of-work and mining in Bitcoin address it? More generally how does blockchain technology address it?
- What other consensus protocols are there? What are some of the tradeoffs of alternative consensus algorithms – proof-of-work, proof-of-stake, etc.?
- How does Bitcoin record transactions? What is unspent transaction output (UTXO)? What is script code embedded in each Bitcoin transaction and how flexible a programming language is it?

Class 4 (9/17): Readings

- *‘Geneva Report*’ Chapter 1 (pages 1 – 7); Casey, Crane, Gensler, Johnson, and Narula
- *‘Blockchain Technology Review’* NIST (pages 23 - 32, sections 3 & 4)
- *‘The Byzantine Generals Problem’* Lamport, Shostak, & Pease (382-387)
- *‘A Short Guide to Consensus Protocols’* CoinDesk

Conclusions

Discussed Bitcoin Design Features

- Timestamped Append-only Logs (Blocks)
- Secured through Cryptographic Hash Functions & Digital Signatures

Consensus Protocol

- Consensus through Proof of Work
- Network of Nodes
- Native Currency

Transactions Ledgers

- Transaction Inputs & Outputs
- Unspent Transaction Output (UTXO)
- Scripting language