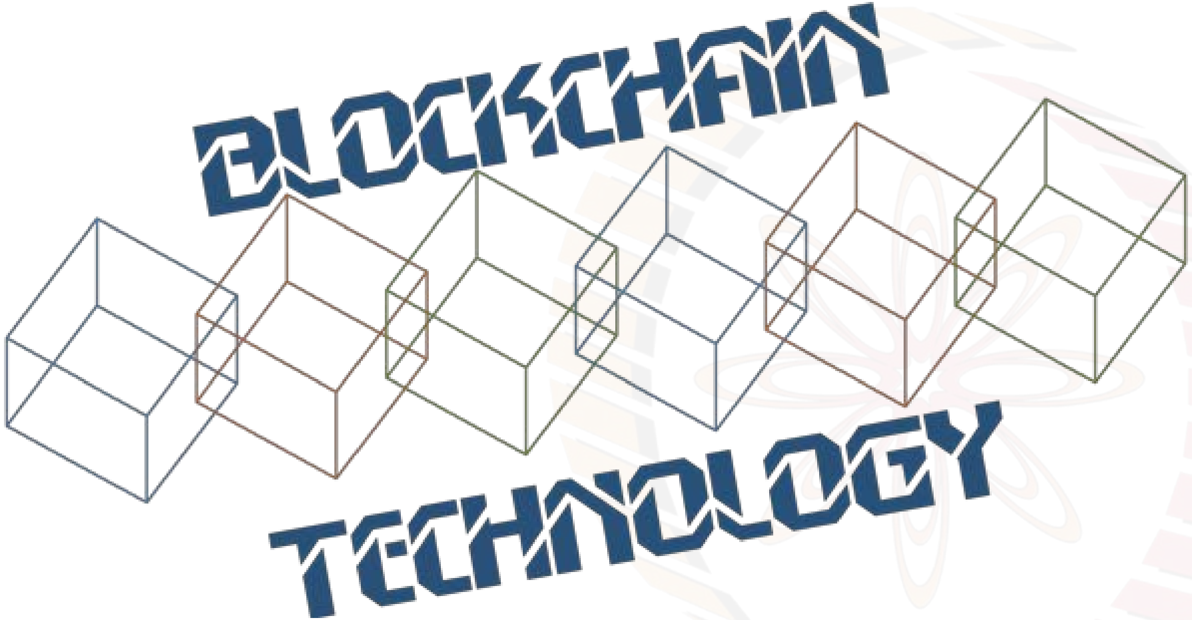
BLOCKCHAINS

ARCHITECTURE, DESIGN AND USE CASES

SANDIP CHAKRABORTY
COMPUTER SCIENCE AND ENGINEERING,
IIT KHARAGPUR

PRAVEEN JAYACHANDRAN
IBM RESEARCH,
INDIA

Image courtesy: http://beetfusion.com/



BASIC CRYPTO PRIMITIVES - II

What We have Looked Into

- Cryptographically Secured Hash Function
  - Collision Free
  - Information Hiding
  - Puzzle Friendly

- Hash Pointers and Data Structures
  - Hashchain
  - Hash Tree – Merkle Tree

Digital Signature

- A **digital code**, which can be included with an electronically transmitted document to verify
  - The content of the document is authenticated
  - The identity of the sender
  - Prevent *non-repudiation* – sender will not be able to deny about the origin of the document

Purpose of Digital Signature

- Only the **signing authority** can sign a document, but everyone can verify the signature

- Signature is **associated with** the particular document
  - Signature of one document cannot be transferred to another document

Public Key Cryptography

- Also known as **asymmetrical cryptography** or **asymmetric key cryptography**

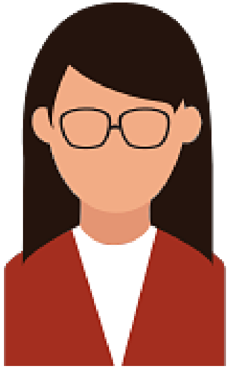
- **Key:** A parameter that determines the functional output of a cryptography algorithm
  - **Encryption:** The key is used to convert a plain-text to a cypher-text; $M' = E(M, k)$
  - **Decryption:** The key is used to convert the cypher-text to the original plain text; $M = E(M', k)$

Public Key Cryptography

- Properties of a cryptographic key (you need to prevent it from being guessed)
  - Generate the key truly randomly so that the attacker can not guess it
  - The key should be of sufficient length – increasing the length makes the key difficult to guess
  - The key should contain sufficient entropy, all the bits in the key should be equally random

Public Key Cryptography

- Two keys are used
  - Private key: Only Alice has her private key
  - Public key: “Public” to everyone – everyone knows Alice’s public key



Encrypt the message with Bob’s public key

$\boldsymbol{M' = E(M, K^{B}_{pub})}$

M’

Encrypt the message with his private key

$\boldsymbol{M = E(M', K^{B}_{pri})}$

Public Key Encryption - RSA

- Named over (Ron) Rivest – (Adi) Shamir – (Leonard) Adleman – inventors of the public key cryptosystem

- The encryption key is public and decryption key is kept secret (private key)
  - Anyone can encrypt the data
  - Only the intended receiver can decrypt the data

RSA Algorithm

- Four phases
  - Key generation
  - Key distribution
  - Encryption
  - Decryption

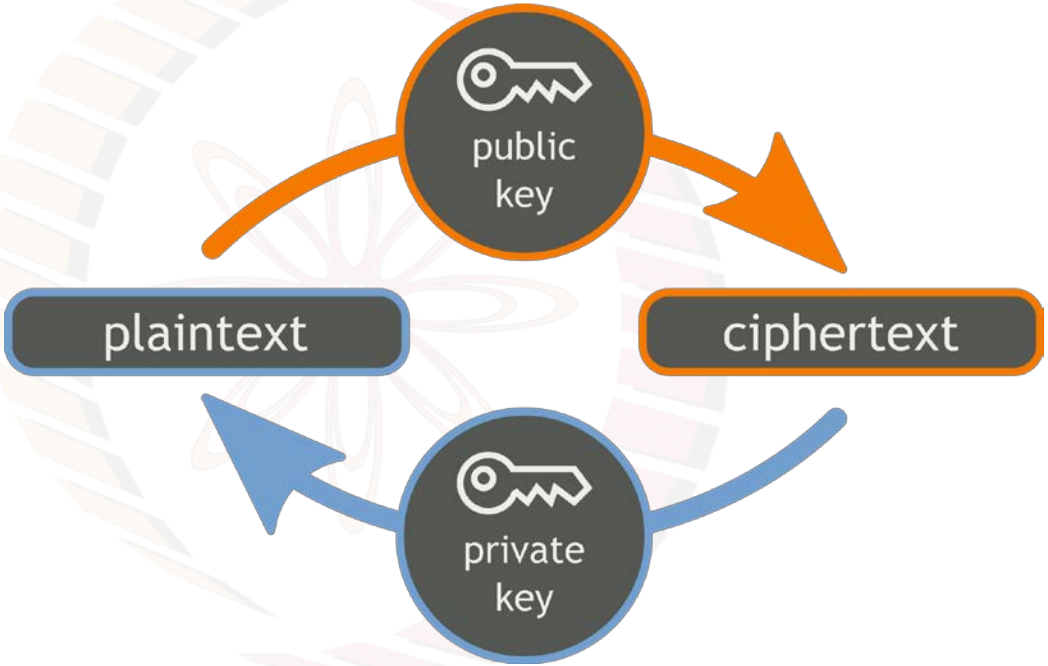


Image source: https://commons.wikimedia.org/

Public and Private Keys in RSA

- It is feasible to find **three very large positive integers** e, d and n; such that *modular exponentiation* for integers m $(0 \leq m < n)$:

$$(m^e)^d \equiv m \ (mod\ n)$$

- Even if you know e, n and m; it is extremely difficult to find d
- Note that

$$(m^e)^d \equiv m \ (mod\ n) = \left(m^d\right)^e \equiv m \ (mod\ n)$$

- (e, n) is used as the public key and (d, n) is used as the private key. m is the message that needs to be encrypted.

RSA Key Generation and Distribution

- Chose two distinct prime integer numbers p and q
  - p and q should be chosen at random to ensure tight security
- Compute $n = pq$; n is used as the modulus, the length of n is called the key length
- Compute $\phi(n) = (p-1)(q-1)$ – *Euler totient function*
- Choose an integer e such that $1 < e < \phi(n)$ and $\gcd\left(e, \phi(n)\right) = 1$; e and $\phi(n)$ are co-prime
- Determine $d = e^{-1}(mod\ \phi(n))$: d is the *modular multiplicative inverse* of $e(mod\ \phi(n))$ [Note $d.e = 1(mod\ \phi(n))$]

RSA Encryption and Decryption

- Let m be the integer representation of a message M.

- Encryption with public key $(\boldsymbol{e}, \boldsymbol{n})$

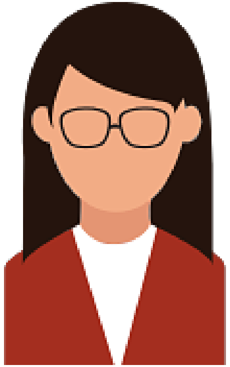
$$c \equiv m^e \ (mod\, n)$$

- Decryption with private key $(\boldsymbol{d}, \boldsymbol{n})$

$$m \equiv c^d \ (mod\, n) \equiv (m^e)^d (mod\, n)$$

Digital Signature using Public Key Cryptography

- Sign the message using the Private key
  - Only Alice can know her private key
- Verify the signature using the Public key
  - Everyone has Alice's public key and they can verify the signature



Sign the message with her private key

$M' = E(M, K^{A}_{pri})$

M, M'

Verify the signature using Alice's public key

$M = E(M', K^{A}_{pub})$

Reduce the Signature Size

- Use the message digest to sign, instead of the original message

Sign the message with her private key

$$S = E(H(M), K^A_{pri})$$

Verify the signature using Alice's public key

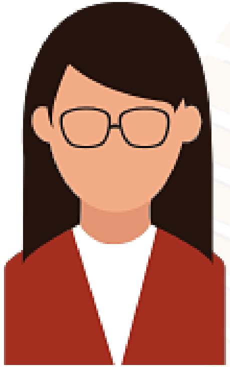
$$H(M) = E(S, K^A_{pub})$$

Digital Signature in Blockchain

- Used to validate the origin of a transaction
  - Prevent non-repudiation
    - Alice can not deny her own transactions
    - No one else can claim Alice's transaction as his/her own transaction

- Bitcoin uses *Elliptic Curve Digital Signature Algorithm (ECDSA)*
  - Based on elliptic curve cryptography
  - Supports good randomness in key generation

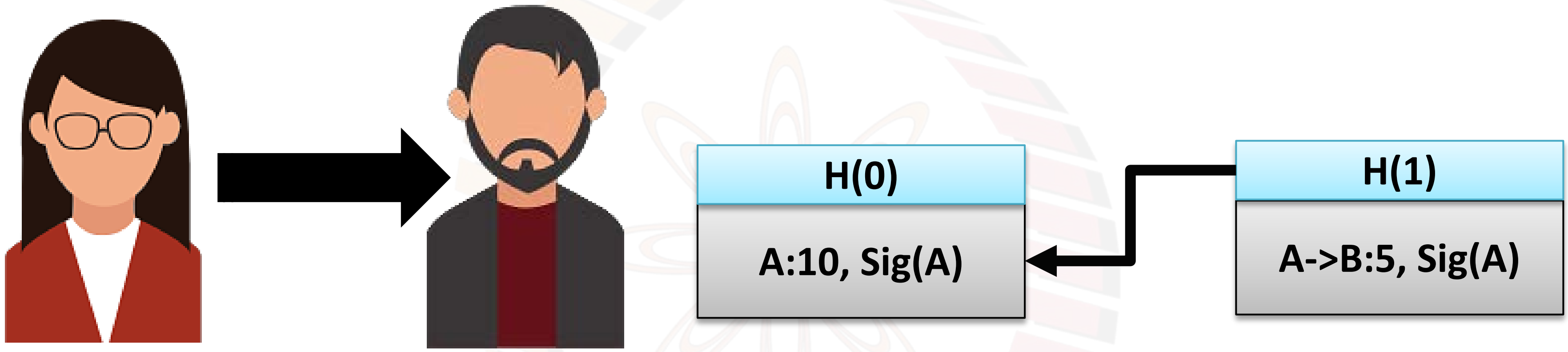
A Cryptocurrency using Hashchain and Digital Signatures



A:10, Sig(A)

- Alice generates 10 coins
- Sign the transaction A:10 using Alice’s private key and put that in the blockchain

A Cryptocurrency using Hashchain and Digital Signatures



- Alice transfers 5 coins to Bob
- Sign the transaction A-B:5 using Alice's private key and put that in the blockchain

A Cryptocurrency using Hashchain and Digital Signatures

- Maintain the economy
  - Generate new coins with time
  - Delete old coins with time

- A central authority like bank can create and destroy coins based on economic policies

- **Crucial Question:** How can we distribute coin management (creation and destroy) thank you!