

## Session 1

# Block, Miner, Blockchain and Applications



This work is licensed under a Creative Commons Attribution – NonCommercial – NoDerivatives 4.0 International License.

# Objectives

- Understand what is **blockchain** and its components

*Distributed database with transactions stored in blocks*

- Global overview of how blockchain are **secured**

*Fingerprinting with hash algorithm and cryptographic challenges*

- Lookup at several **common applications** of blockchain

*Cryptocurrency, smart contract, identity management, etc.*

# Blockchain



# Distributed Database

- Blockchain is basically a technology to **store data**
  - Making it possible for reliable data exchange between users
  - Providing guarantees on data immutability
  - Without any central supervision entity
- Many **applications** have been developed on a blockchain
  - Cryptocurrency, smart contracts, identity management, etc.*

# Blockchain

- A **blockchain** is a data structure holding transactions  
*Completely open to any and everyone on the network*
- Blockchain technologies characterised by **three main properties**
  - **Security**: since theoretically not alterable
  - **Transparency**: since content visible by everyone
  - **Decentralisation**: since stored in a P2P fashion
- A chain of records controlled by **no single authority**  
*Extremely difficult to change a stored information*

# Blockchain Type

- Two broad categories of blockchain depending on privacy level
  - A blockchain can be a public or a private one*
- Public blockchain is a permissionless ledger
  - Anyone can download it, browse the history and modify it
  - Can be compromised if the rules are not executed strictly
- Only trusted participants can access a private blockchain
  - Overall control of the network in the hands of the owners
  - Possibility to define rules with levels of permissions

# Block

- Information is stored in a chain of secure **blocks**

*Each block can be seen as an instance of a data structure*

- A blockchain is an **array of blocks** referring each other linearly

*The size of the array can dynamically change with time*





# Mining

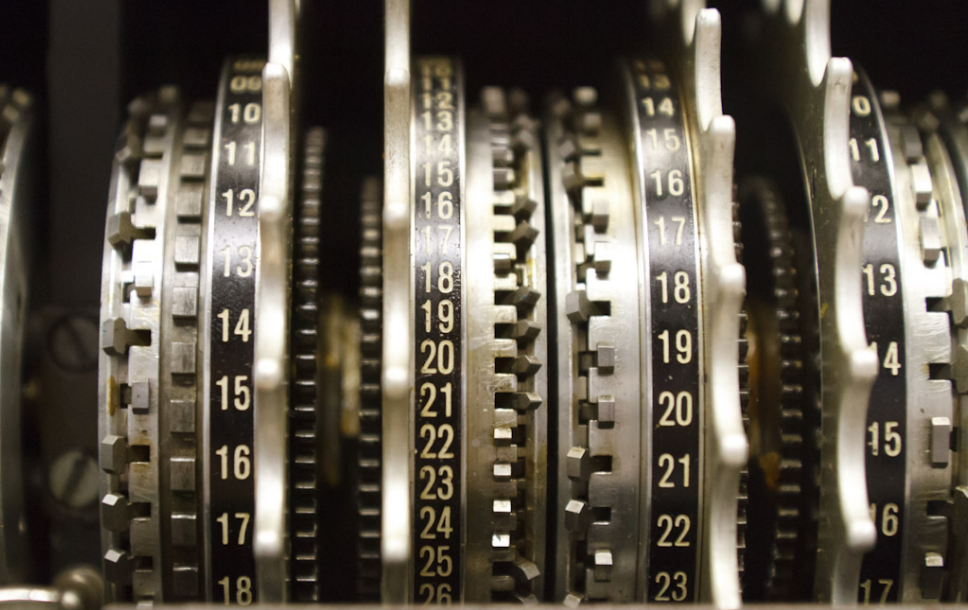
- New blocks can be added into a given blockchain

*This operation is done by the mining process*

- New blockchain is shared amongst all the users with P2P

*New blocks checked and propagated if correct, rejected otherwise*

- Checking that a block is valid is done with a specific algorithm
  - Typically the “proof of work” algorithm
  - Solving a “mathematical puzzle” with a given level of difficulty



**Cryptographic Tool**

# Alice and Bob (1)

- Alice and Bob exchange messages on **communication channel**

*Insecure channel, with Eve trying to intercept the exchanges*

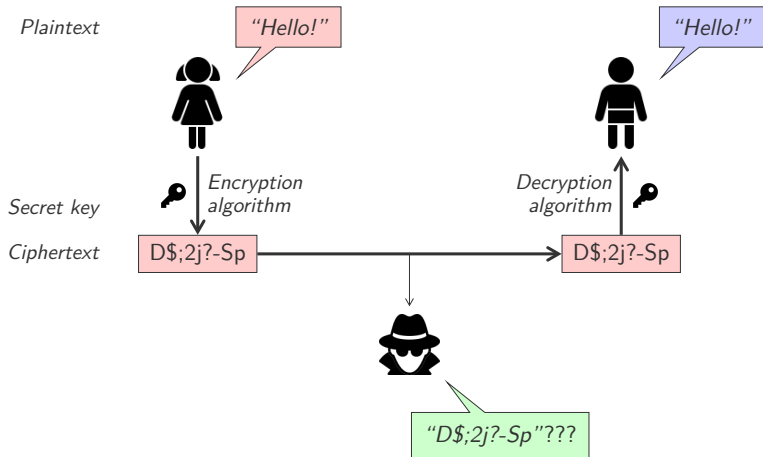
- **Cryptography** turns a clear text into a ciphered text

*Transmission of the ciphered text, Eve cannot understand it*

- Only Alice and Bob can read the message thanks to a **key**

*This key needs to be shared between both stakeholders*

# Alice and Bob (2)



# Symmetric Encryption

- Using the same secret key  $K$  with **symmetric encryption**

*The key defines the encryption  $e_K$  and decryption  $d_K$  functions*

- **Exposure** of either  $e_K$  or  $d_K$  renders the system insecure

*Also,  $e_K$  and  $d_K$  are typically very close*

- Require **secure channel** between Alice and Bob to exchange  $K$

*Very difficult if they live far away or do not know each other*

# Asymmetric Encryption

- $d_K$  impossible to find from  $e_K$  with **asymmetric encryption**
  - Public key  $e_K$  to encrypt a plaintext
  - Private key  $d_K$  to decrypt a ciphertext
- **No need** for a key exchange on a secure channel

*Only Bob can decrypt a plaintext encrypted with  $e_K$*
- Several **public-key cryptosystem** do exist

*Diffie-Hellman, RSA and ElGamal (and their variants)*

# Hash Function

- Technique used to check for **data integrity**

*Computing a digital fingerprint for a given data*

- Using a **hash function**  $h$  to get a fingerprint  $y = h(x)$

*For any  $x$ , a binary sequence of arbitrary length*

- A **fingerprint** is a binary sequence (typically 160 bits)
  - Storing the data  $x$  and its fingerprint  $h(x)$  separately
  - The fingerprint  $h(x)$  should be stored in a secure place

# Collision

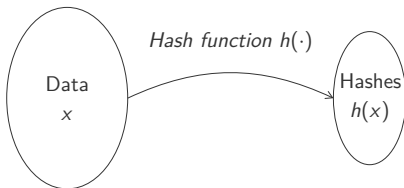
- Hash function does some **compression** of the data

*The domain of the function is larger than its image*

- Two different data  $x$  and  $x'$  can result in the **same fingerprint**

*This is known as collision and is expected from hash functions*

- Ideally **collisions** must be minimised





# Signature Scheme

- Digitally sign a document with a **signature scheme**

*Adding the signature to the message, not “writing on top of it”*

- Problem with the **verification of a signature**

*How is it possible to compare a signature with the “original” one*

- Signed document can be **used several times**

*For example, authorisation for an action (withdraw 100 euros)*

# Certificate

- Mechanism to **authenticate public keys** with certificate

*Require some kind of Public Key Infrastructure (PKI)*

- Relies on a trusted **certification authority** (CA)

- Signs the public keys of all people in the network
- Verification key  $ver_{CA}$  known “by magic” by everyone

- Signed **certificate** contains several information

*Name, email, address, list of public keys*



# Application

# Cryptocurrency

- Bitcoin and **cryptocurrencies** are the first application  
*Electronic decentralised medium of exchange without control*

- Opposed to currency managed by **central banking systems**  
*Confidence towards the bank institutions are necessary*

- Blockchain used to host and publish a **distributed ledger**  
*Public financial transaction database allowing control by peers*

# Smart Contract

- Adding **code to be executed** inside a blockchain

*Makes it possible to establish a contract between entities*

- **Smart contract** holds executable content

*Triggered and executed when some conditions are met*

# Identity Management

- Blockchain used to build **trusted database** with public access
  - Used to manage identities of people
  - Store degrees delivered by schools and checked by companies
- Can also be used to protect **intellectual property**
  - Storing copyrights information as smart contracts*

# References

- Jean-Luc Verhelst (2017). *Bitcoin, the Blockchain and Beyond: A 360-Degree onboarding guide to the first cryptocurrency and blockchain*, Self-Published, ISBN: 978-2-930-97100-1.
- IBM. *What is blockchain technology?*, retrieved on February 2, 2020. <https://www.ibm.com/blockchain/what-is-blockchain>
- Mayank Pratap (2018). *Blockchain Technology Explained: Introduction, Meaning, and Applications*, July 30, 2018. <https://hackernoon.com/blockchain-technology-explained-introduction-meaning-and-applications-edbd6759a2b2>
- Eden Au (2019). *Building a Minimal Blockchain in Python: Understanding blockchain by coding*, July 13, 2019. <https://towardsdatascience.com/building-a-minimal-blockchain-in-python-4f2e9934101d>

# Credits

- Icons from <https://icons8.com/icons>.
- Mario Antonio Pena Zapatería, November 21, 2006, <https://www.flickr.com/photos/oneras/32634430557>.
- Adam Foster, December 7, 2011,  
<https://www.flickr.com/photos/twosevenoneonenineeightthreesevenatenzerosix/6655759625>.
- Stock Catalog, February 14, 2018, <https://www.flickr.com/photos/stockcatalog/38461156880>.