



Mathematics, Cryptography, Blockchains, and Cryptocurrencies:

Myths and Realities.

Alberto Pace



<https://indico.cern.ch/event/848910/>

Outlook

- The concept of data integrity
- Cryptography and encryption algorithms
- Digital signatures
- Blockchains
- Cryptocurrencies
- General considerations and conclusions

Data Integrity

- A problem we have been facing at CERN since the use of computers: avoid data corruption or data changes.
- Ensure that reference, immutable data is unchanged over its entire lifetime and that eventual multiple copies are kept consistent
 - Multiple causes can trigger loss of data integrity: media failure, data movements, software errors, malicious intent, human errors, ...
- The traditional solution to verify data integrity is the use of 'checksums'

Checksums

- A checksum is a fixed-sized value derived from a block of data for the purpose of detecting errors
- Example:

Data	1	5	8	4	9	2	4	4	7	8	6	7	4	2	5	7	8	3	1	0	(check)sum	95
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------------	----

- Summing the values of the bytes and storing the modulo of the results in a single byte (8-bit) is an example of a particular weak checksum
 - Not immune to many common errors, example: byte swapping, byte shifting, ...

Data	1	5	8	4	9	2	4	4	8	7	6	7	4	2	5	7	8	3	1	0	(check)sum	95
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------------	----

Data	0	1	5	8	4	9	2	4	4	7	8	6	7	4	2	5	7	8	3	1	(check)sum	95
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------------	----

- In general, with a 8-bit checksum (0-255) there is a $1/256$ probability that an error would pass undetected

Better definition for Checksums

- Any (small) modification in the data generates a (large) modification in the resulting checksum. The computed result should be evenly distributed across the possible values, especially for inputs that are similar
 - Adler-32, CRC-32, CRC-64, ...
- Within this scenario, the probability of data corruption being undetected can be reduced by increasing the checksum size:
 - With 32 bit checksums: $1/2^{32} = 1/4294967296 = 2.32 \times 10^{-10}$
 - With 64 bit checksums: $1/2^{64} = 1/1.84467E+19 = 5.42 \times 10^{-20}$
- However ...
 - checksums can be forged easily and are unsafe for protecting against intentional modification

Original data

Data	1	5	8	4	9	2	4	4	7	8	6	7	4	2	5	7	8	3	1	0	(check)sum	95
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------------	----

Tampered data

Data	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	5	(check)sum	95
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------------	----

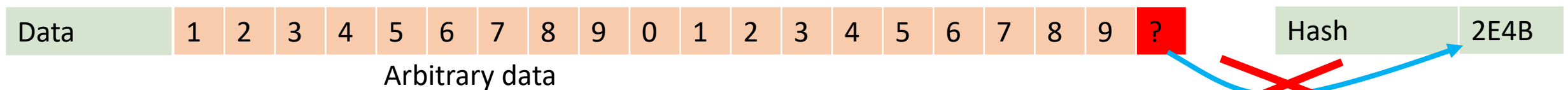
Arbitrary data

Forged last byte to match checksum



Are checksums enough to ensure integrity ?

- Yes, if the storage architecture prevents intentional data tampering ...
 - With the advantage that checksums are very fast to compute
- However, it may be better to calculate a cryptographic hash, which has more stringent requirements:
 - It is computationally unfeasible to construct an input that produces a given hash. Hashes cannot be reversed.



Computationally impossible to find data to match the given the hash
Even more difficult forge a subset of the data to match the given hash

Definition: Cryptographic Hash Functions

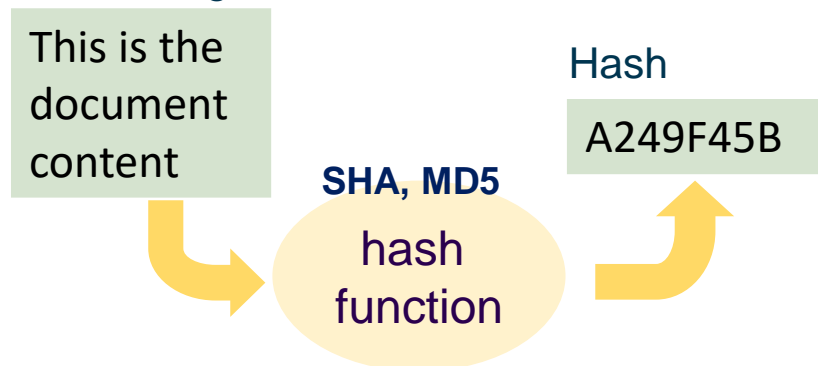
- Same properties as checksums: An efficiently computable transformation that returns a fixed-size string, which is a short representation of the data from which it was computed
 - Any (small) modification in the data generates a modification in the result
 - The computed result should be evenly distributed across the possible values, especially for inputs that are similar
- With one additional requirement: it must be:
 - Impossible to find a data that matches a given hash
 - Impossible to find “collisions”, where two different data have the same hash

SHA1 has 160 bits, more than 10^{48} values

SHA-256 has $> 10^{77}$ values

SHA-512 has $> 10^{154}$ values

Data or message or file



However the problem is not solved

- Hashes are still unsafe for protecting against *intentional* modifications
 - If a person can modify the data, he can also recalculate and modify the hash
- So the hash must contain a proof of the identity that calculated it. This is achieved by *encrypting* it. The entity that decrypts it will be able to verify this identity.

Original data

Data	1	5	8	4	9	2	4	4	7	8	6	7	4	2	5	7	8	3	1	0	hash	23ABD38C
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------	----------

Tampered data

Data	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	5	hash	B3452D67
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------	----------

Arbitrary data

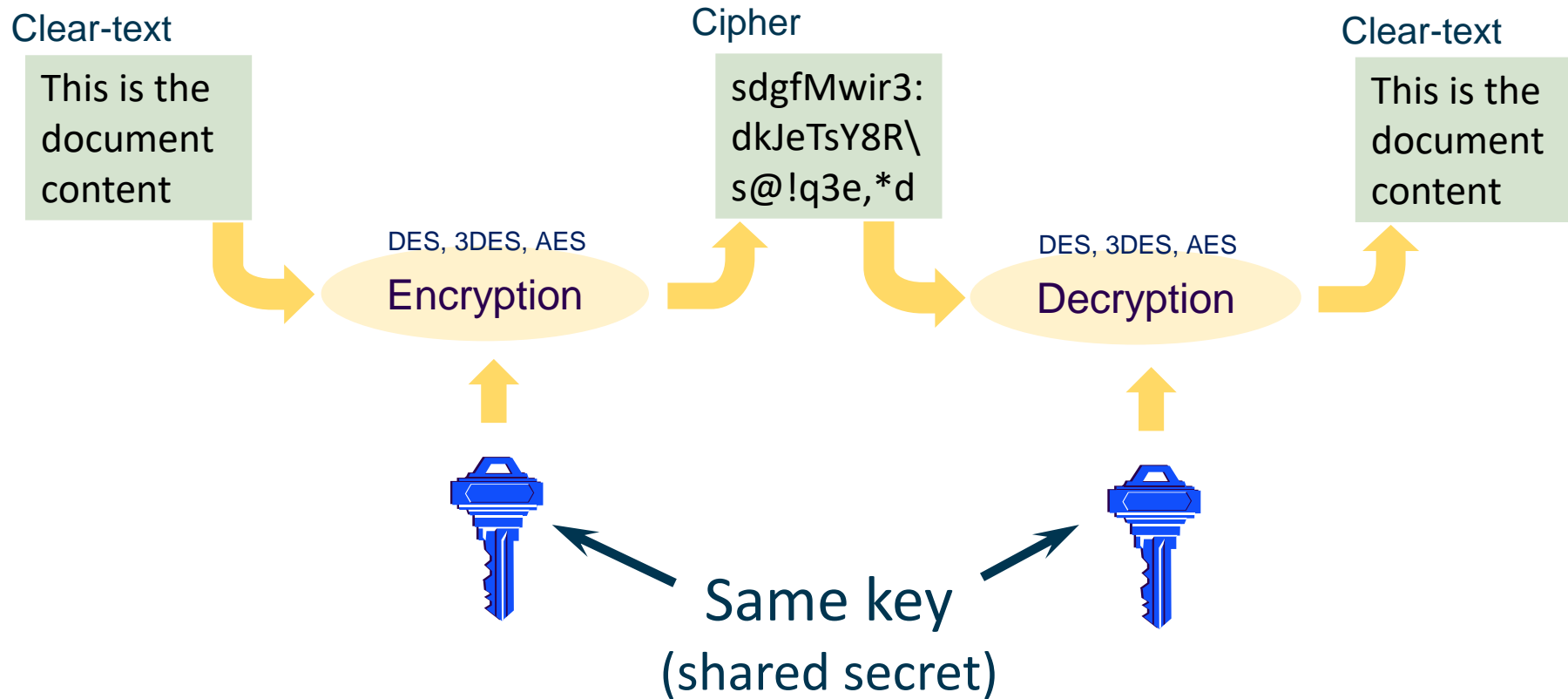
Forged data and hash recalculated to match data

Why Encryption ?

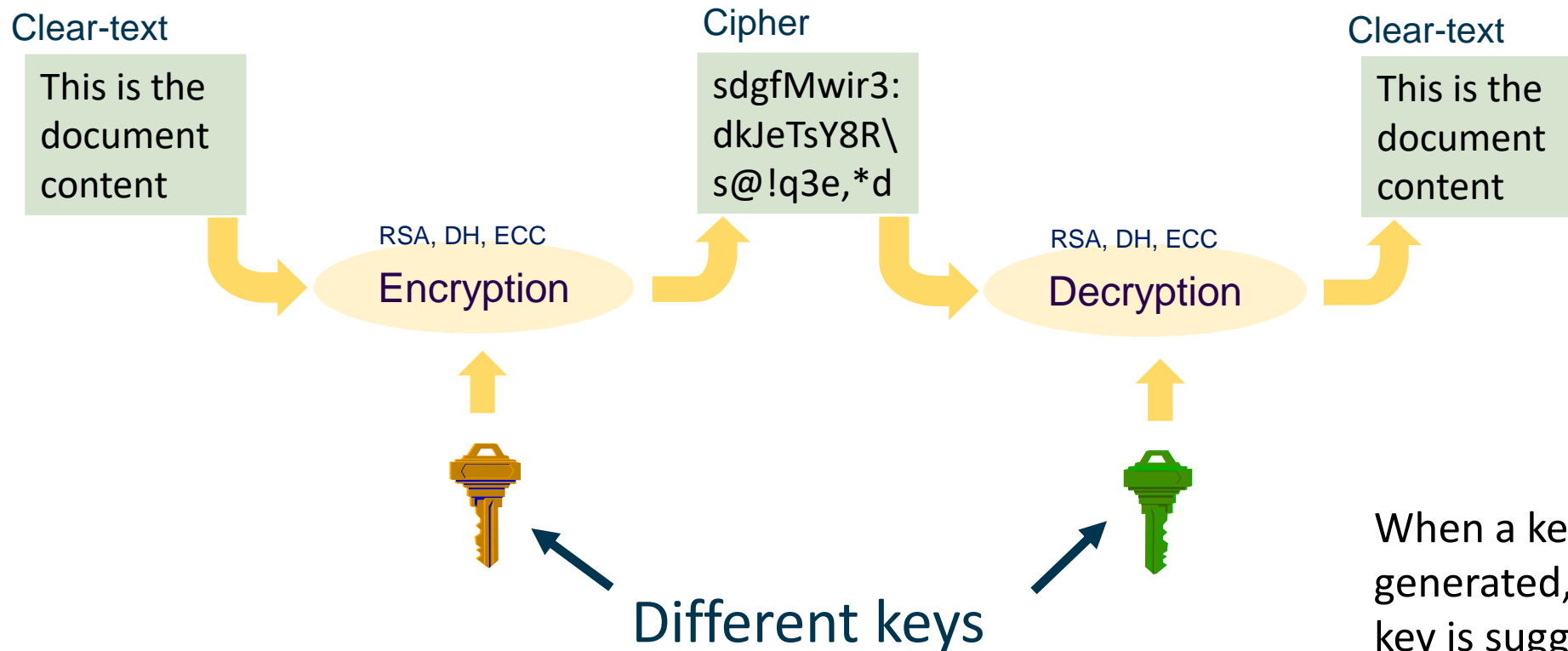
What can you achieve with Cryptography ?

- Confidentiality
 - Ensure that nobody can get knowledge of what you transfer even if listening to the whole conversation
- Integrity
 - Ensure that data has not been modified during the transmission
- Authenticity, Identity, Non-repudiation
 - You can verify that you are talking to the entity you think you are talking to
 - You can verify who is the specific individual behind that entity
 - The individual behind that asset cannot deny being associated with it

Symmetric Encryption

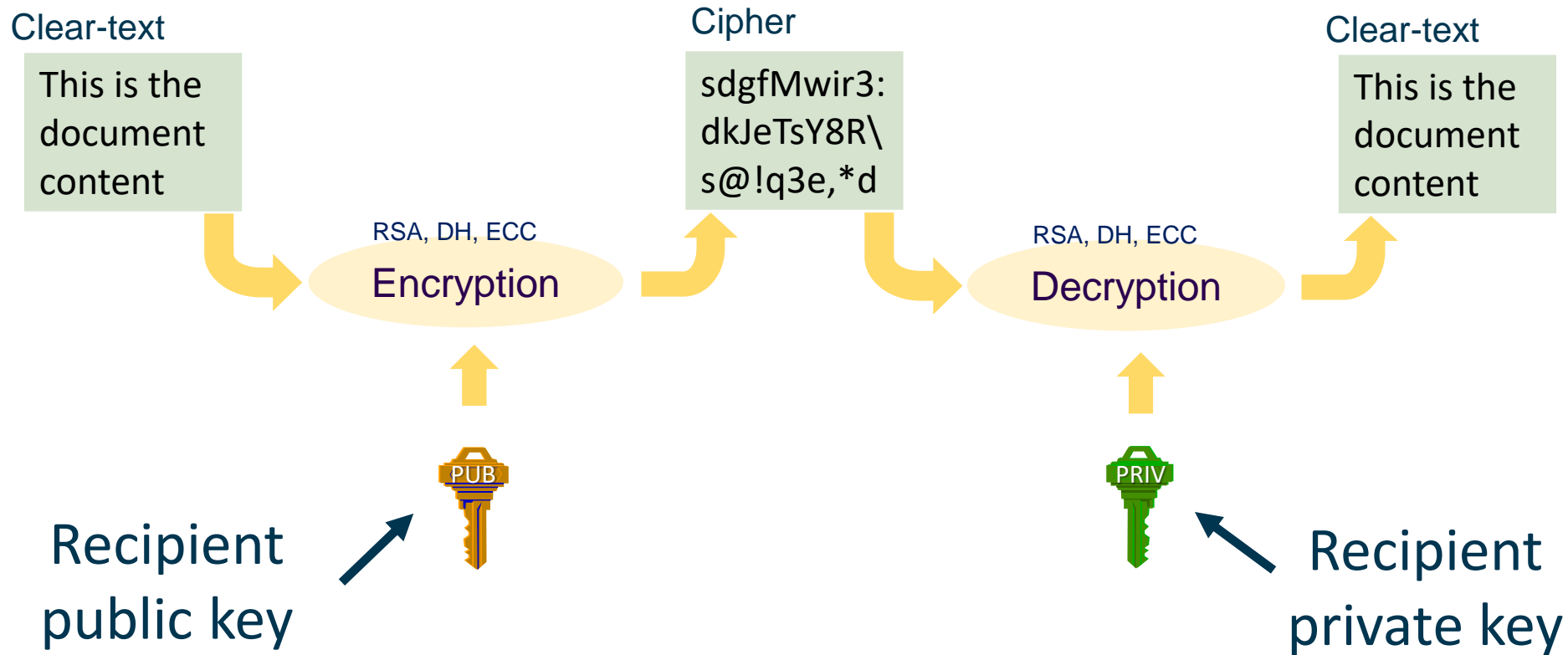


Asymmetric Encryption

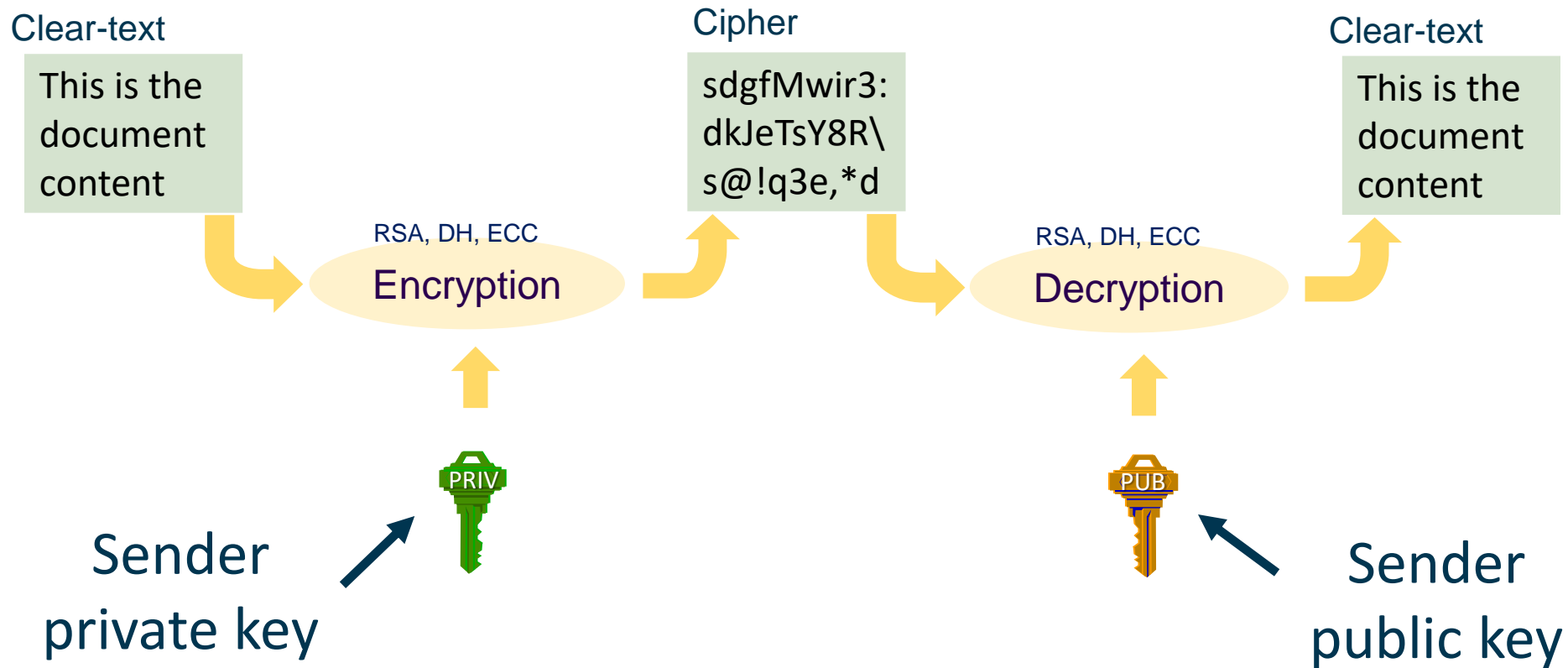


When a key pair is generated, one of the two key is suggested to become public and the other to be kept private

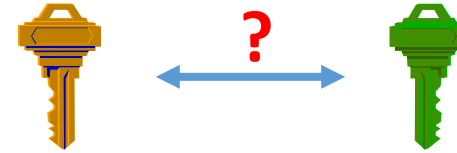
Example: Confidentiality



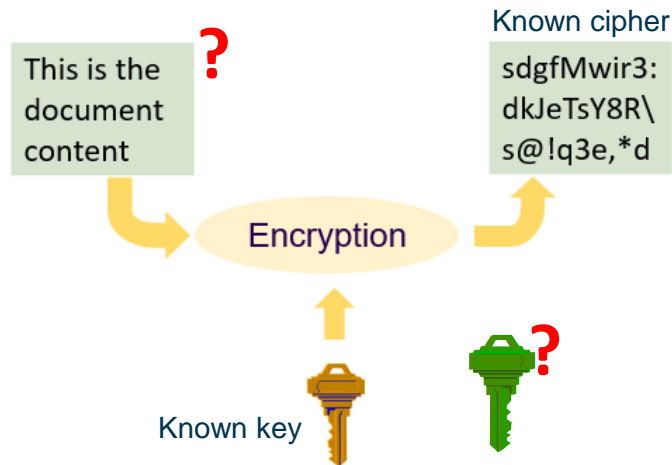
Example: Authenticity





Asymmetric Encryption



- Things to remember
 - Given one key used for encryption, there is only one key that allows the decryption
 - The relation between the two keys is unknown and from one key you cannot gain knowledge of the other, even if you have access to clear-text and cipher-text



I like apples	4DfghTy7%8 9HfrcF%7g	
The dog is white	Ms3dr%gSD TF6Huy&”	 
We came today	3fR6tg^bn,> o7y3EdsQ	
Don't Smoke	duJn64Dvn< .:kh%dw@	

As you have one key, you can generate all cipher text you want to guess the other key

Cracking asymmetric encryption ...

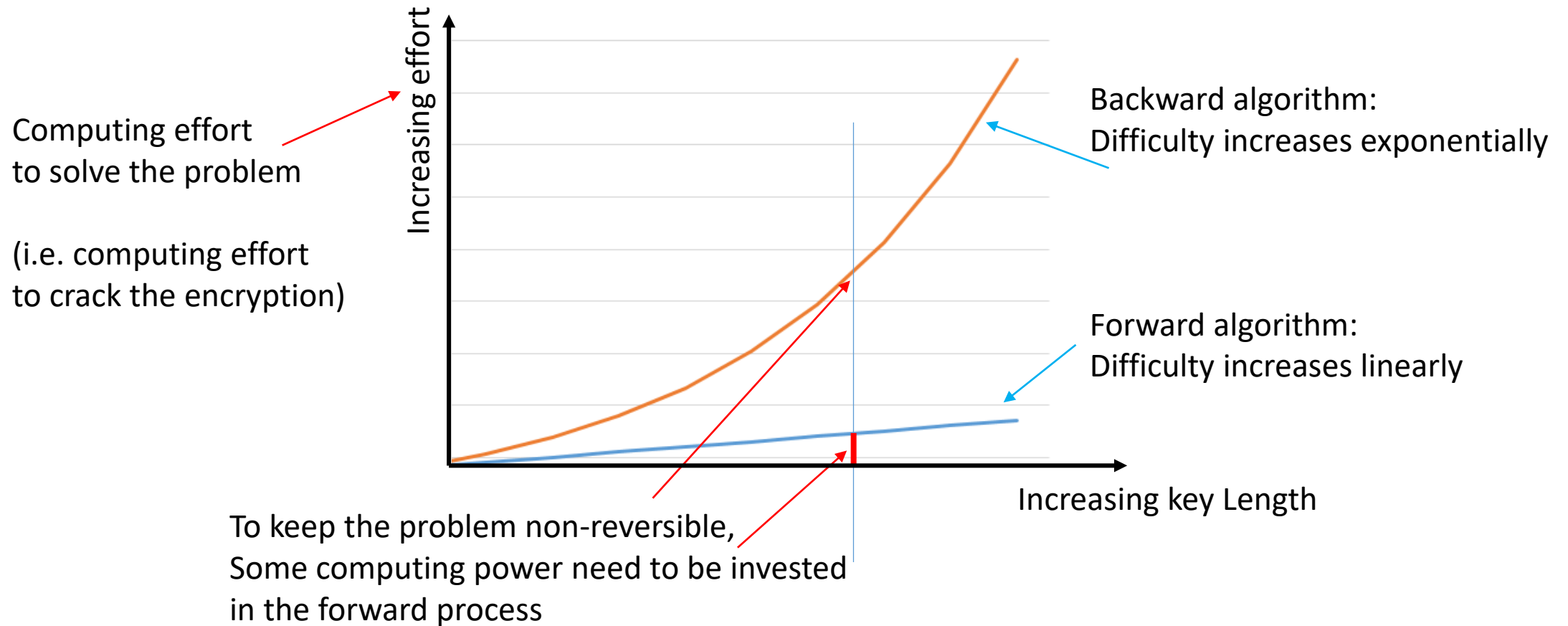
- Cracking asymmetric encryption is like solving a (difficult) mathematical problem that is entirely defined
 - Example: Find x, y so that
 - $x * y = 5549139029240772017554613865259030307060771696148489$
 - (Answer: 2833419889721787128217599, 195845982777569926302400511)
- Asymmetric encryption security relies on the fact that some easy calculations are computationally difficult to reverse
 - unless you have a *hint* (that you find in the private key)
 - RSA: Easy to multiply, difficult to factorize
 - Diffie-Hellman (DH), Elliptic Curve Cryptography (ECC): exponentiation is easy, logarithm calculation is difficult in some groups

Discrete logarithm problem

$$y = g^x \pmod{p}$$



- To ensure that reversing the algorithm is difficult, the more calculation you spend in the forward algorithm, the more you make it difficult to reverse
 - This explains why asymmetric encryption is always slow compared to the symmetric one.
- A race between guns and armors ...



Want to take the challenge ?

- Find any x and y so that

$x * y =$ 25195908475657893494027183240048398571429282126204032027777137836
0436620207075955562640185258807844069182906412495150821892985591491761845
0280848912007284499268739280728777673597141834727026189637501497182469116
5077613379859095700097330459748808428401797429100642458691817195118746121
5151726546322822168699875491824224336372590851418654620435767984233871847
7444792073993423658482382428119816381501067481045166037730605620161967625
6133844143603833904414952634432190114657544454178424020924616515723350778
7077498171257724679629263863563732899121548314381678998850404453640235273
81951378636564391212010397122822120720357

This is the RSA 2048 bit challenge, it has 617 digits $\log_{10}(2^{2048}) + 1$

- If you find the solution you will be rewarded 200'000 USD (RSA-2048 challenge)

Digital signatures

What is a Digital Signature ?

- We need a way to guarantee the integrity of our data if the attacker is able to modify both data and the hash.
 - A digital signature solves this problem

Original data

Data	1	5	8	4	9	2	4	4	7	8	6	7	4	2	5	7	8	3	1	0	hash	23ABD38C
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	------	----------

Creating a digital signature

- A digital signature is the hash of the data encrypted with the signatory private key

Data or message or file

This is the document content

SHA, MD5
Hash function

Hash

W4f%7G*g

Digital Signature

G5^gj&J8

RSA, DH, ECC
Encryption



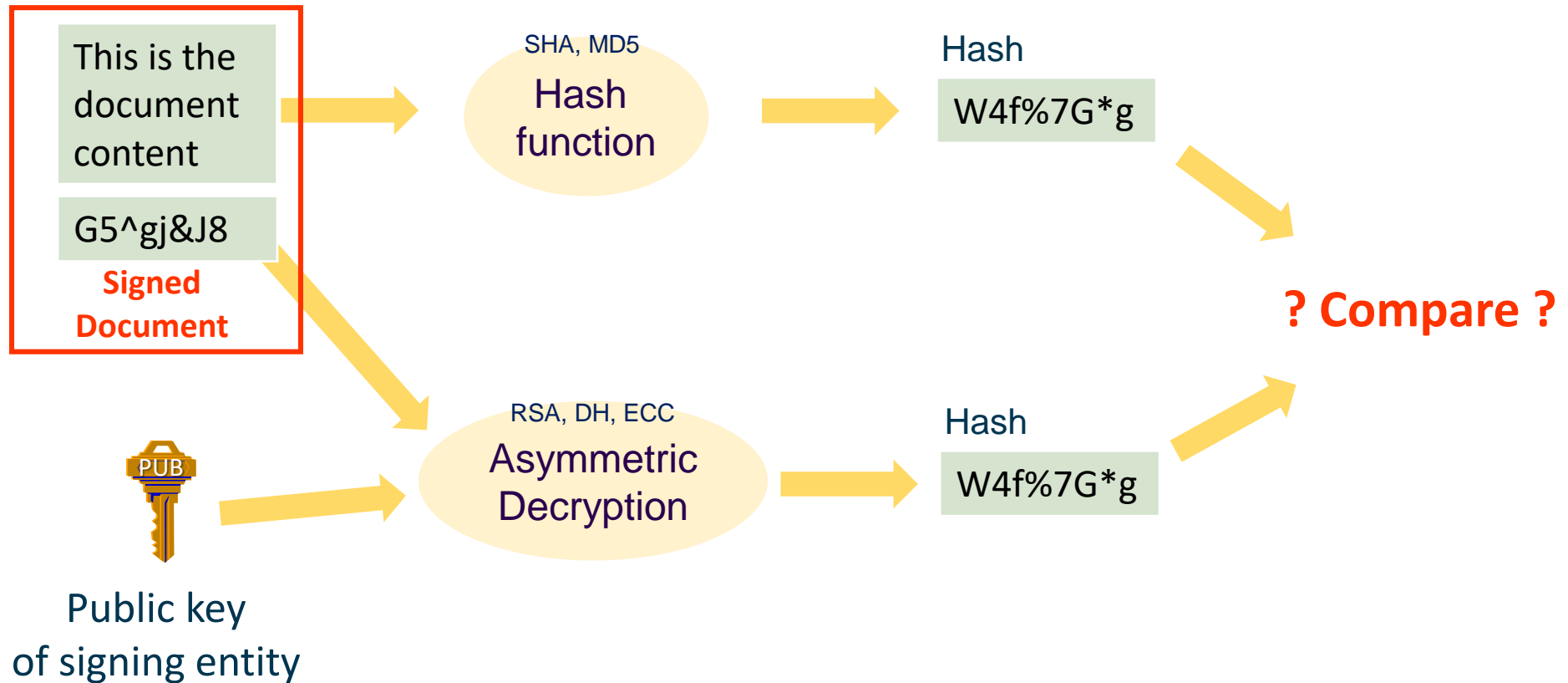
private key of signing entity

This is the document content

G5^gj&J8

Signed Document

Verifying a digital signature



Why digital signatures ?

- The digital signature solves the initial integrity problem and prevents all tampering attempts
 - Because the attacker needs knowledge of the signatory private key to regenerate a digital signature

Original data

Data	1	5	8	4	9	2	4	4	7	8	6	7	4	2	5	7	8	3	1	0	signature	23ABD38C
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----------	----------

Tampered data

Data	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	5	signature	23ABD38C
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----------	----------

Hash mismatch

Data	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	5	signature	B3452D67
------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----------	----------

Public key decryption fails

Things to remember ...

- Digital signed documents are NOT encrypted.
 - Anyone who has access to the document can read it
 - No knowledge of any key is necessary to read the document
- Anyone can verify the integrity and the authenticity of the document
 - The *knowledge of the public key* of the signatory is necessary for the verification
- If the document is modified, it needs to be signed again
 - *Knowledge of the private key* required
- Q: how do you get the public key of the signatory ?
 - Using its *certificate* which is also a signed document

This is the
document
content

G5^gj&J8

**Signed
Document**

The public
key of
CERN is
eE3\$%d4t

w3E5g^&4

**Signed
Document**

CERN Certificate
(Signed by an authority that you trust)

And ...

- Remember the battle between guns and armors ?
- Computers become faster and faster ...
 - When cryptography is used for authentication there is no problem: you increase the key length
 - But encrypted or digital signed data can become crackable after some years !
 - Widely used RSA-512 required 300 computers working for a period of few months in 1999. Today, you can rent cloud computing power and get the result in few hours.
- So ... periodically (every 10 years ?) you may need to re-encrypt or re-sign all your documents with longer keys

Blockchains

Why blockchains ?

- Digital signatures are safe, but require an entity (who owns the signatory private key) to certify the truth. This entity must be trusted by everybody.
- Anybody can read the data and verify that it has not been tampered.
 - Using the signatory public key
- Blockchain is a technology that voids the need for a unique trusted signatory

What is a blockchain (1)

- It is a list of data blocks that are linked together with a timestamp.
- Each block contains the hash of its data concatenated to the hash of the previous block(s)

block1 (Genesis)

Data	Hash	Timestamp
Alice owns res 23	Hash(Data1)	3434532

Note: only *hashes*,
no digital signatures

block2

Data	Hash	Timestamp
Bob owns res 45	Hash(Hash1 + Hash(Data2))	3434542

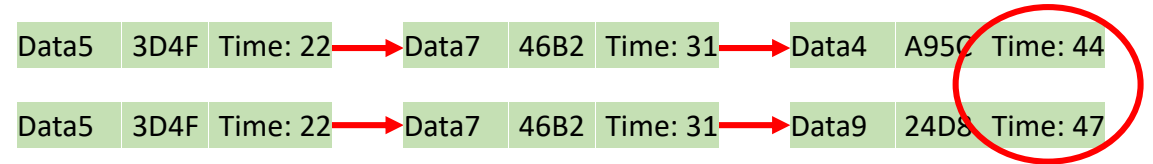
block3

Data	Hash	Timestamp
Michel can read res 45	Hash(Hash2 + Hash(Data3))	3434642

block4

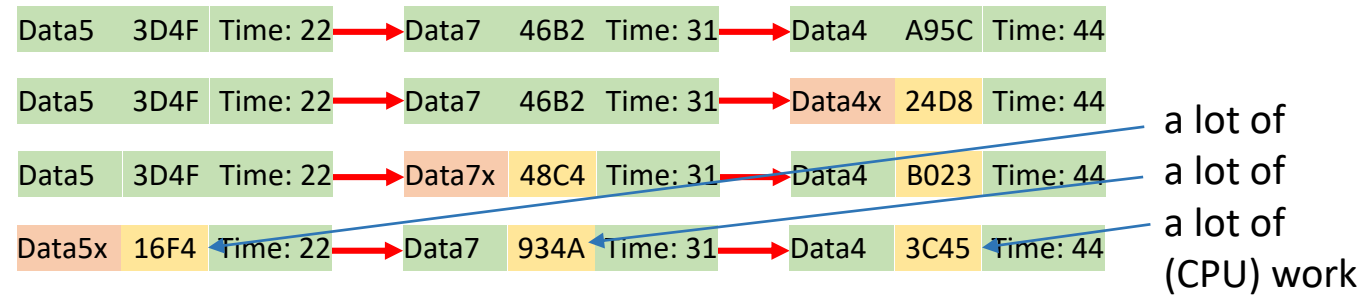
Data	Hash	Timestamp
Charlie can write res 23	Hash(Hash3 + Hash(Data4))	3435634

What is a blockchain (2)



- The blockchain represents a write-only database
 - its immutability is guaranteed by a 'large' number of 'peers' computers that have a copy of the blockchain
 - It can be used as a ledger, widely accessible because it is distributed.
- Any peer computer can add a new block to the chain and, after updating its hashes, can distribute a newer version to all peers.
 - As blocks are produced concurrently the blockchain is constantly forking
- Each blockchain has an algorithm for scoring different versions and retaining only the one with a highest score
 - Therefore, whenever two peers discover different blockchains, they will unambiguously retain only one version.
 - Peers that have inserted blocks into rejected blockchains have to insert it again on the new reference blockchain
- There is an incentive to extend another more recent chain rather than attempting to replace it. This is achieved by requiring (serious) computational effort to extend.
 - This process is called mining ...

The mining process



- Without the mining process, anyone could change the content of the blockchain and recalculate the hashes
 - Recalculating the hashes is deliberately made hard by adding additional constraints
- Mining is the algorithm to add a block. It must require some proof of (CPU) work
 - Example: The bitcoin blockchain requires to increment a nonce (derived from the data) until its hash has a certain number of leading zeros. Only peers that have done this work can add the new block to the blockchain.
 - Verification of the proof of work (leading zeros) is very fast (recalculate the hash with the nonce)
- A peer computer is never guaranteed that he is working on the ‘master’ copy of a blockchain or on a fork.
- However because with the mining work, the probability of an old block becoming superseded decreases exponentially as new blocks are added.
 - A valid block cannot be changed without redoing the work. As later blocks are chained after it, the work to change the chain would include redoing the work for each subsequent block. Computationally challenging !
 - So blocks in the chain that are ‘in depth’ can be considered statistically impossible not to be true

Why blockchains ?

Using digital signatures



- Truth can be guaranteed by a notary role trusted by the community
 - A person, a company, a government ...
 - In everyday life:
 - You trust the notary that you are buying a house or when you pay him
 - You trust your bank when you deposit money in your account
 - You trust a government when you accept a person's passport or an identity card
- Blockchains
 - allow to have distributed system that no authority can control / manipulate
 - implement and enforce the algorithm (the contract) that has been designed in the chain
 - We have seen that it is probabilistically impossible that blockchains could be faked
 - So they represent a *super partes* truth, above all stakeholders

Why blockchains ? (2)

- Many examples of potentials usages ...
 - Electronic votes, traceability of goods, trading contracts, copyright records, digital rights management, energy efficiency records, electronic diplomas, ...
- The blockchain becomes a trusted database with defined processes that allows peer-to-peer entities to trust each other
- Solves many weakness of the centralized notary approach that:
 - can be hacked
 - can be inefficient in handling transaction and slow down business
 - can discriminate
 - can take an unreasonable % of the transaction value to deliver little service
 - can steal information, privacy concerns

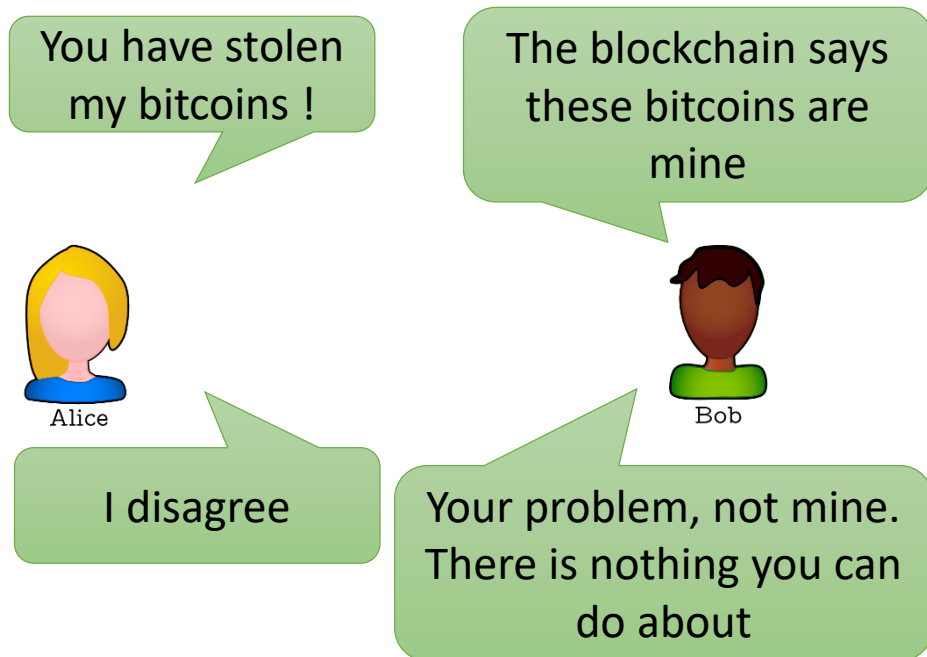
Blockchains everywhere ?

- It depends
- The community has to agree that the master reference information (the ledger) is transferred from the current human-guaranteed systems into the blockchain
 - Transfer of authority
- Two cases:
 - The resource is the ledger itself (example: a cryptocurrency) or resource is managed directly by the ledger
 - The ledger is the reference, cannot be disputed.
 - The resource is a real good, referenced by the ledger
 - Every stakeholder must accept that the ledger is the only and unique book representing the truth
 - What if a stakeholder does not accept what the ledger says ? Everyday we see people denying evidence in all fields. Also in science !

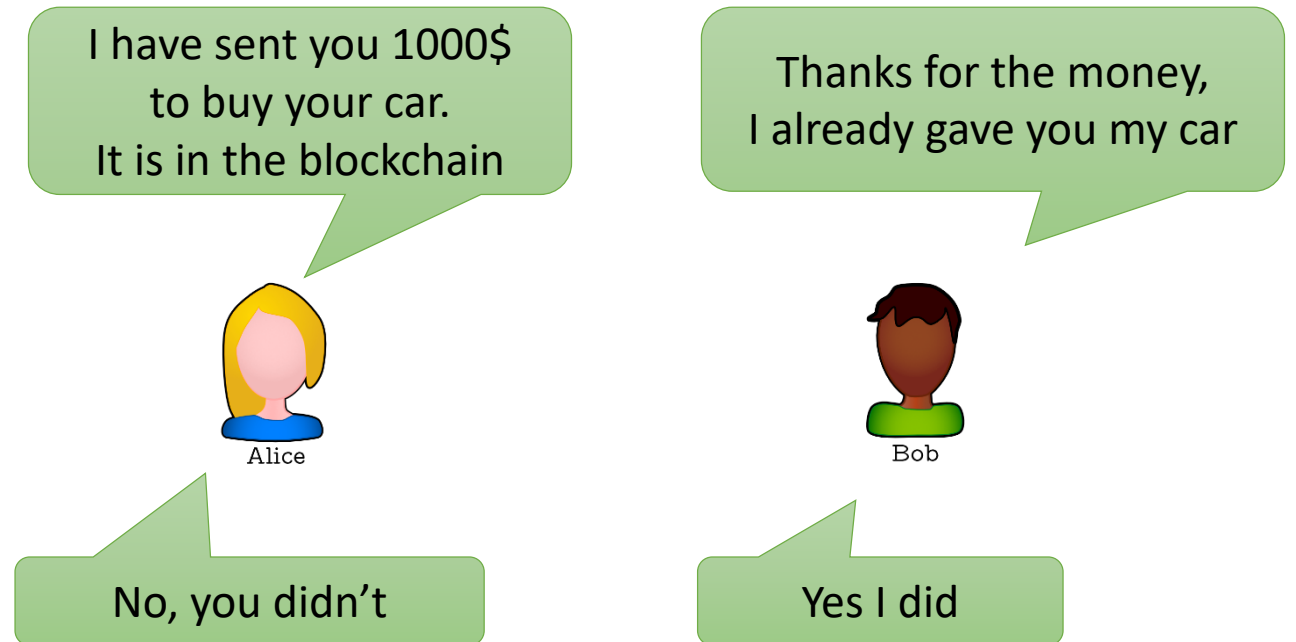
Personal opinion

Examples

A Cryptocurrency blockchain



A blockchain that records peer-to-peer sales but does not manage the real-world resources



How to resolve the dispute ?

Examples

A blockchain that records peer to peer sales paid in cryptocurrencies and only references real-world resources

I have sent you 10 ETH to rent your car. It is in the blockchain

Here is the proof in the blockchain that I gave you my car



Alice



Bob

But this is a different car and it is not working

No, it is the same car and it is working

How to resolve the dispute ?

A blockchain that sell music songs paid in cryptocurrencies. The blockchain manages the resources involved

I have sent you 10 ETH to buy your song. Here is my public key

In the blockchain you can find the token that allows to get your music using your private key



Alice

Disputes resolved automatically



Bob

It didn't work

I have lost my private key

Contract says: not possible

Read the instructions again

contract says: your problem

Changed mind, please return

Myths and Realities

- Myth

- Blockchains will be used everywhere

- Reality

- Blockchain can be used everywhere the community accepts to delegate both resources management and payments to an algorithm (contract) that cannot be changed. **Forever.**
- Whenever there are “real” resources involved, it is hard (impossible ?) to enforce contracts (or resolve disputes) with software
 - The success of companies connecting peer-to-peer actors for business (Alibaba, Ebay, Amazon Marketplace, Aliexpress, Uber, Airbnb, ...) is not related to the ability to establish a contract between two untrusted entities but on the reputation gained in resolving disputes.

Cryptocurrencies

Why blockchains are successful as alternative currencies ?

- The cryptocurrency blockchain contains the list of transactions that describe changes of ownership
 - It is publicly readable, produces a single version of the 'truth'
 - There is *some value* in having a single version of the 'truth'
- It is portable, coin creation is predictable, rate-limited, and not infinite (scarce).
 - It has the properties of a currency
- The contract is clear and cannot be changed.
 - Contrary to normal currencies, governments (the notaries of normal currencies) cannot interfere. No one can interfere.
- If the value is > 0 , and it is subjective - it can be traded !

Future scenarios

- Validating transactions in the blockchain is rewarded using the same cryptocurrency.
 - Mining cryptocurrencies gives less (nominal) rewards as time passes.
 - Sustainable only if there is deflation (negative inflation)
- Bitcoin example:
 - Mining rewards 12.5 XBT/block halves every 4 years. In mid 2020 it will drop to 6.25 XBT/block. There is an estimated number of 3-5 million miners
 - Computers become faster and will mine faster but mining remunerate less.
 - Miners may stop mining if the value of the bitcoin drops
 - Less miners means that the revenue/miner will increase
 - So it is an auto stabilizing process that is linked to the value of the bitcoin
 - No risk for the ledger until there is one copy: the cumulated mining effort remains and cannot be faked
- Ethereum example:
 - There is no guaranteed time nor a guaranteed cost for a transaction to be executed. But by paying more 'gas' you can have a transaction processed before others.



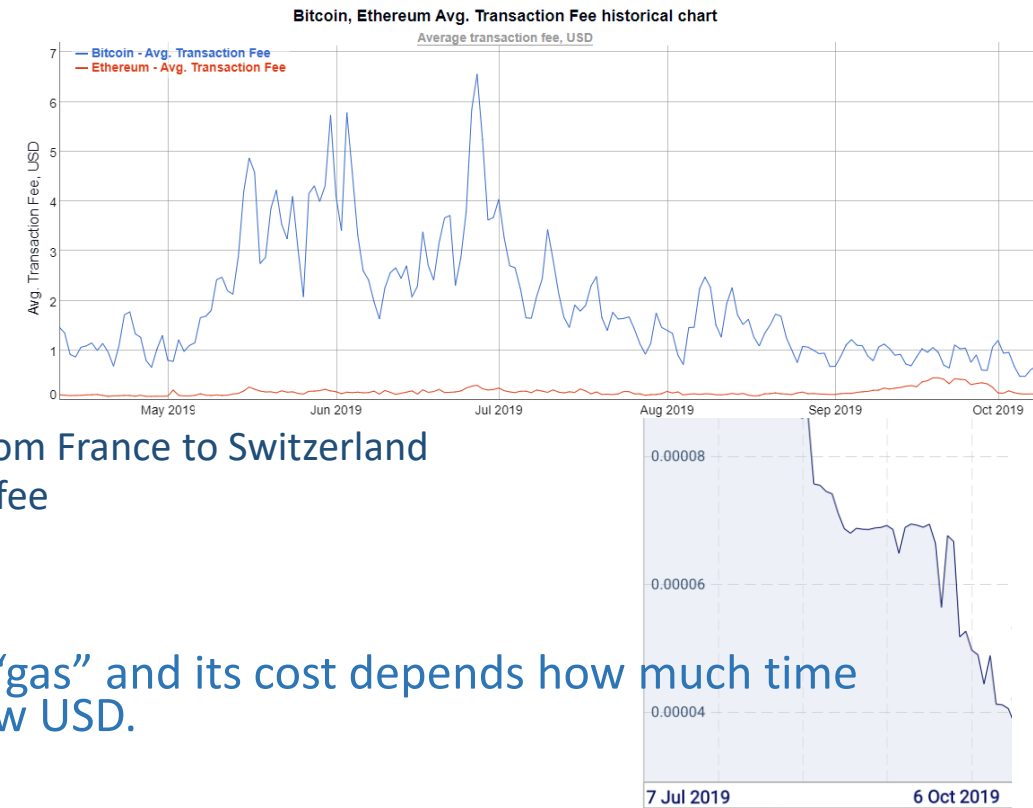
Myths and Realities

- Myths

- Cryptocurrencies are anonymous
 - Not really, you have the public key of the owner. Cash is more anonymous.
- Cryptocurrencies are not safe and can be easily lost
 - They are safe ... but can be lost, like real money
- Are an easy way to make money
 - Cryptocurrencies are traded. You do not need to own them to trade
- Are used by terrorists or for illegal reasons
 - Of course. As are USD, EUR, CHF or any other currency
- Are an effective technique to hide wealth
 - In this respect, identical to cash.
- Cryptocurrencies will be everywhere
 - Next slide

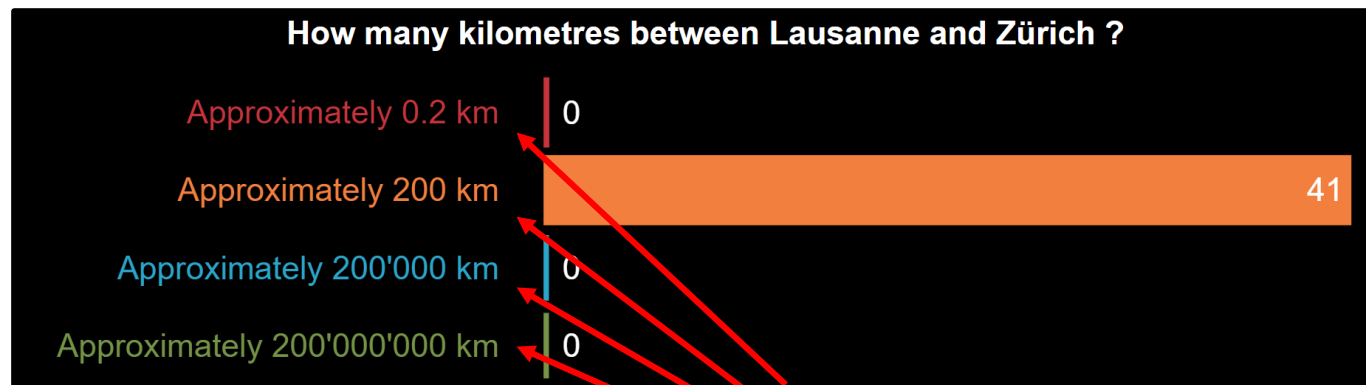
Realities

- Why Cryptocurrencies ? Will they be everywhere ?
 - The notary role has been (sometimes) abused
 - Examples:
 - A traditional bank can charge 28 EUR for one wire transfer from France to Switzerland
 - Migrants sending money to remote countries can pay a 10% fee
 - A government can junk savings of an entire population
- But trading cryptocurrencies has a fee.
 - Example: a single Ethereum transaction is traded using “gas” and its cost depends how much time you are willing to wait to have it executed. It can cost few USD.
 - What if you need to record a million transactions ?
- You will always have new ‘notaries’ that will run a ledger for free just to gain your trust
 - New banking models
 - New trading models
 - No exorbitant fees
- Cryptocurrencies have been a technology that has allowed to stop several abuses in dominant positions and will continue to do so.
 - They expose the real cost of a guaranteed distributed ledger.



Conclusion

- Has something changed ?
 - Progress in computing technology has been exponential from its inception
 - And it is not over yet ! We are only at the very beginning ...
- The general population is not aware of these changes



Answer can be found by common sense

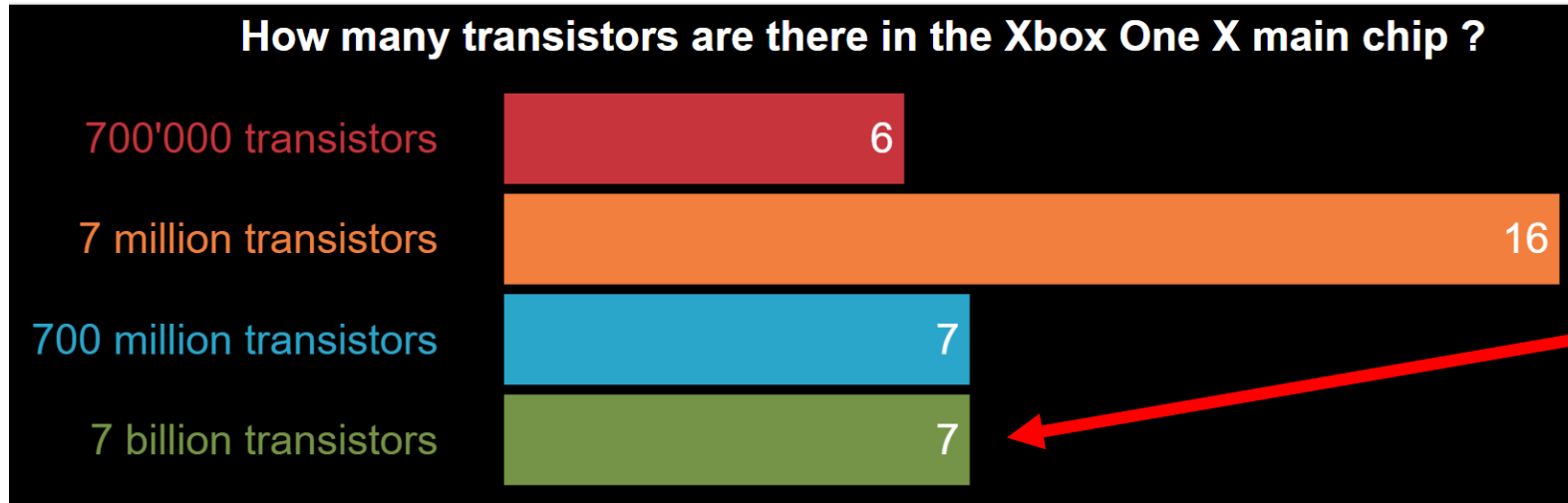
Survey: How many QR-codes are possible ...

- ... using the smallest 16x16 size ?
 - A. 256
 - B. One million
 - C. One for every human being on earth (~ 8 billions)
 - D. One for every atom on Earth ($\sim 10^{48}$)
 - E. One for every atom on the universe ($\sim 10^{77}$)



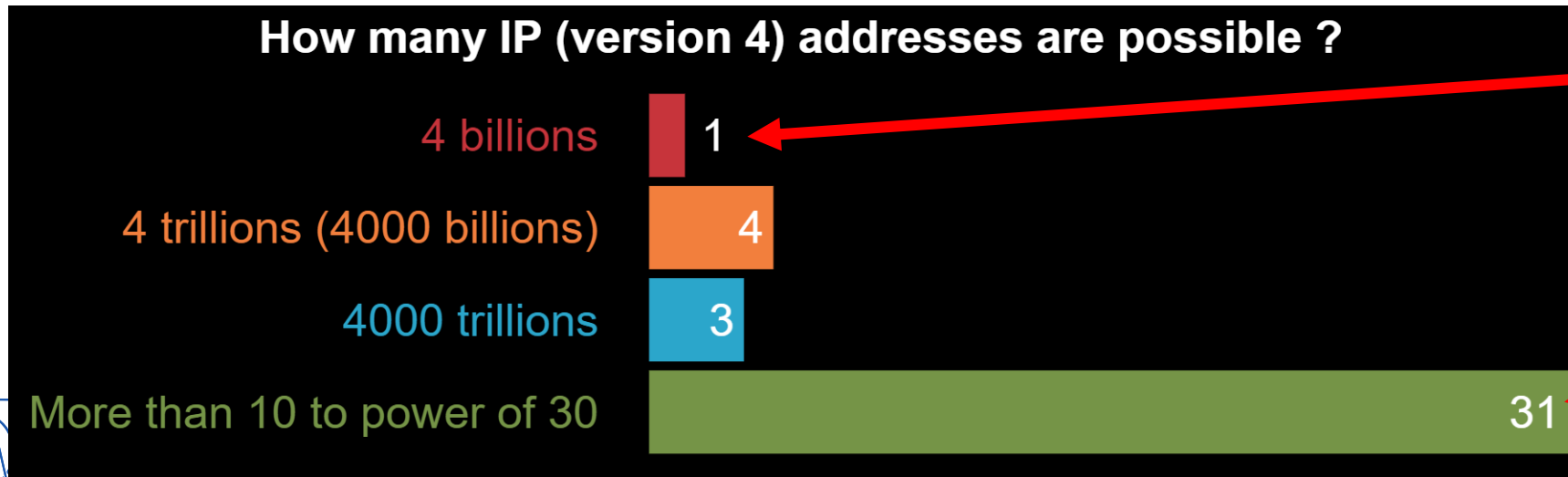
A 25x25 QR-code

but ... what about questions on computing ?



most popular answer has 3 orders of magnitude error

less than 1/5 Correct answers !



only 1 Correct answers !

20 orders of magnitude error

Conclusion (2)

- Some “notary” roles of transactions can be **guaranteed by algorithms**
 - An application distributed across thousands of computers can ensure, verify and guarantee by itself its own integrity
 - No need of third party intermediary, no commissions
 - No possibility to manipulate the book of writings
 - No possibility to change the “rule of the game” (the contracts) in an ongoing process
- Everything we learned in hard science during traditional studies is still valid:
 - Mathematics, Statistics, Physics laws still applies. Nothing changes.
- But ... computers and networks can do more today than a few years ago
 - Solutions that were computationally unfeasible in the past become possible today.
- Cannot fight progress
 - Many of these approaches can bring significant improvements to everyone's life
 - plenty of new business opportunities, ethical consequences must be understood and handled
 - **Education** is of the utmost importance

