Αρχές κρυπτογραφίας

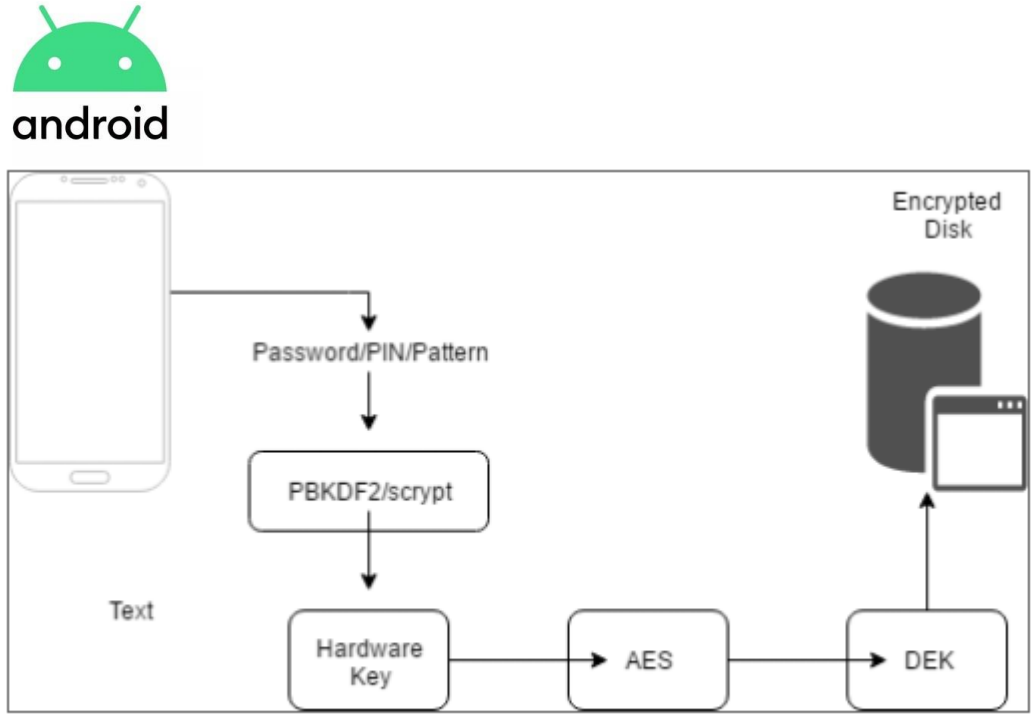
Επικ. Καθηγητής, Χριστόφορος Νταντογιάν
dadoyan@ionio.gr

Cryptography and Network Security

- Cryptography is everywhere nowadays
- Cryptography is a key factor of success and advance → Internet, mobile networks, Information systems in general

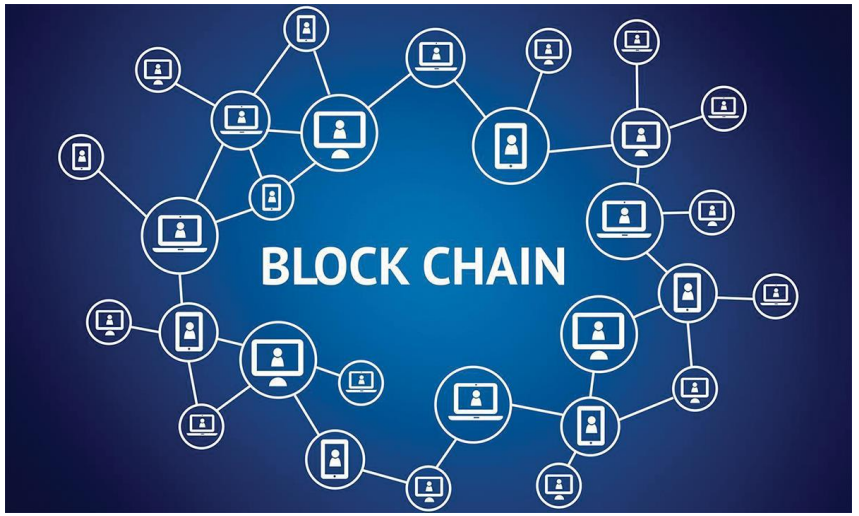
Disk encryption





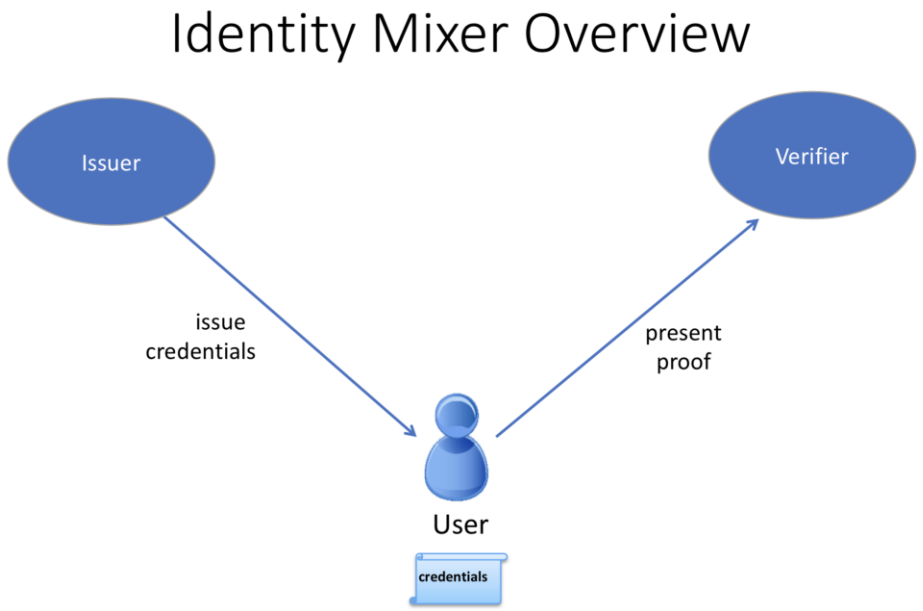
Cryptocurrencies





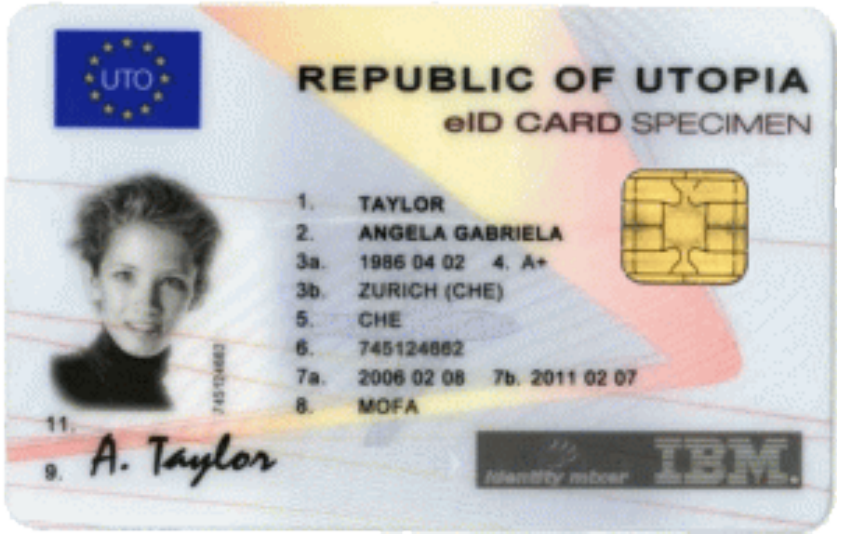
Anonymous credentials

- I can prove that I am over 18 without revealing my age
- I can prove that I am a student with revealing my Card ID and name

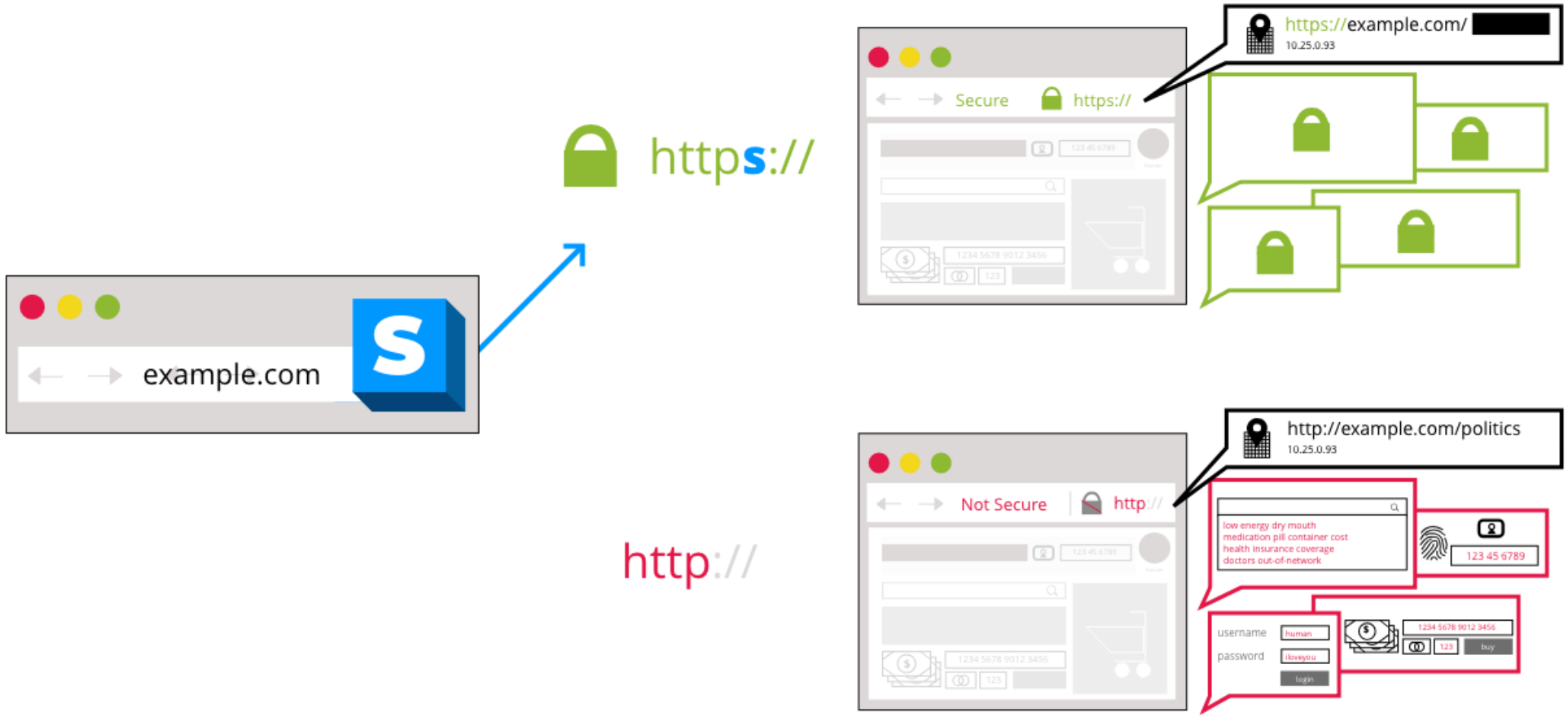


Smart cards

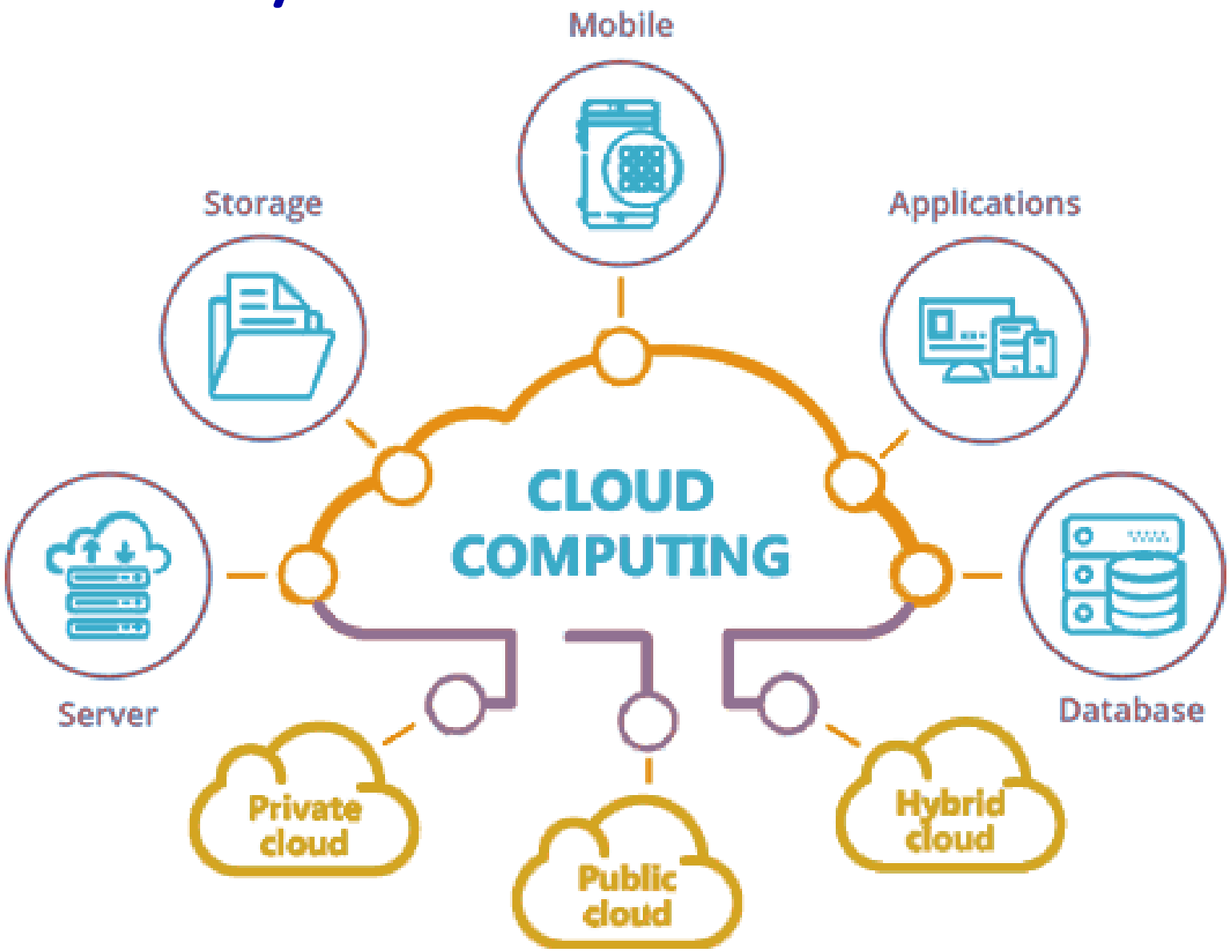

Standart SIM
Micro SIM
Nano SIM
eSIM


UTO
REPUBLIC OF UTOPIA
eID CARD SPECIMEN
1. TAYLOR
2. ANGELA GABRIELA
3a. 1986 04 02 4. A+
3b. ZURICH (CHE)
5. CHE
6. 745124662
7a. 2006 02 08 7b. 2011 02 07
8. MOFA
11.
9. A. Taylor
IBM

Internet Security



Cloud Security



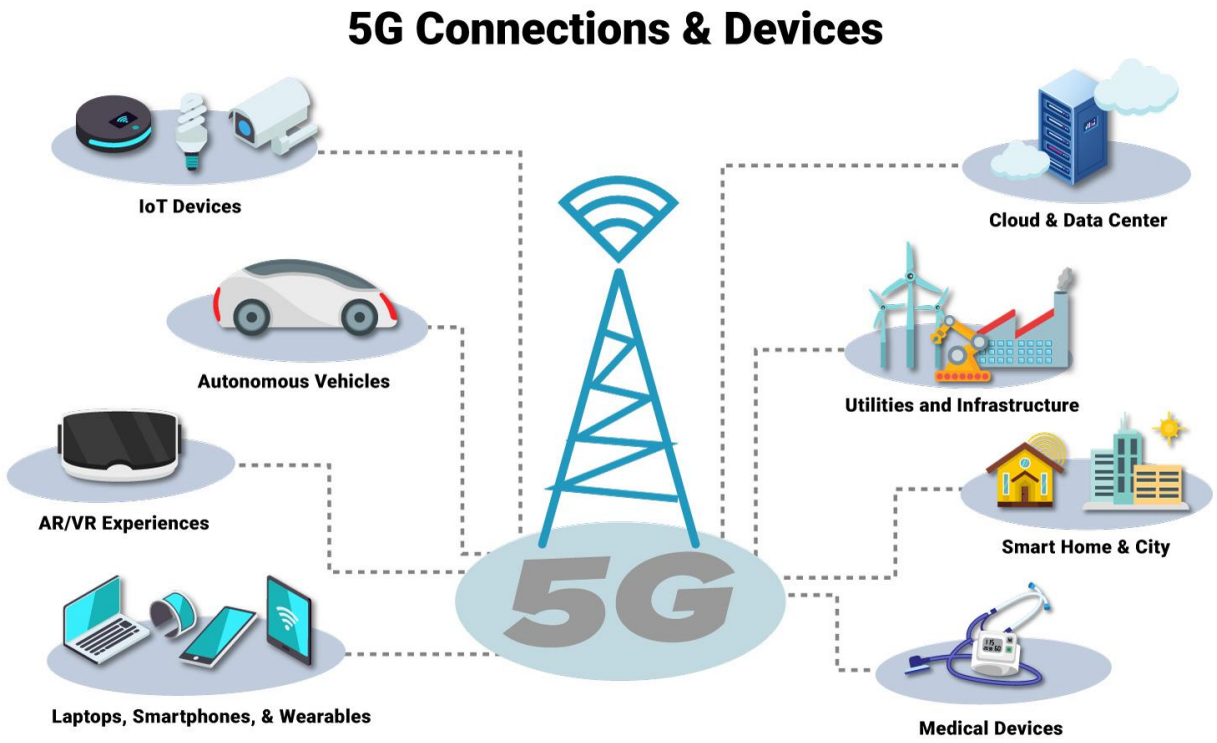
Wifi



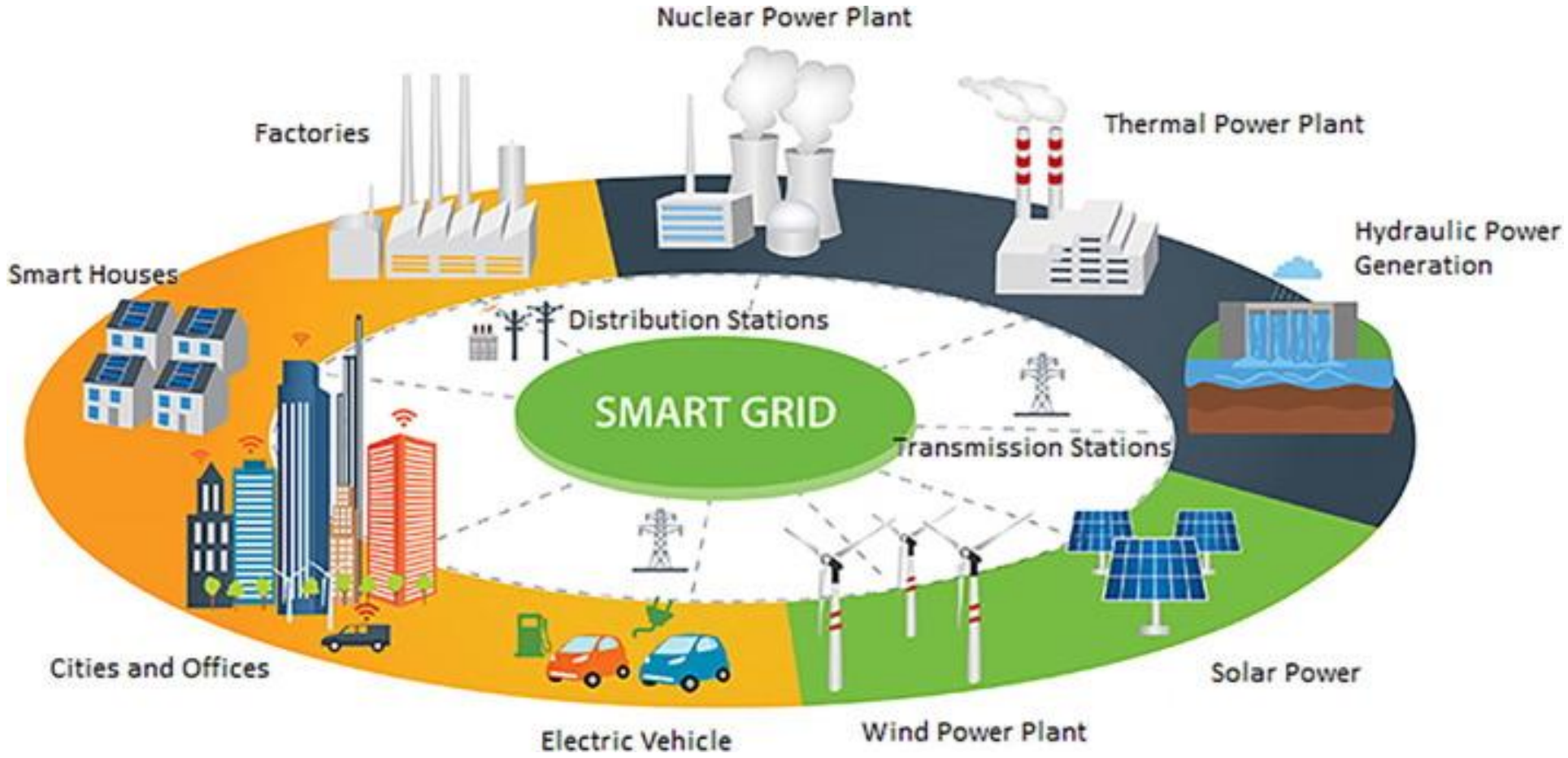


Mobile networks





Smart grid


Nuclear Power Plant
Factories
Thermal Power Plant
Hydraulic Power Generation
Smart Houses
Distribution Stations
SMART GRID
Transmission Stations
Cities and Offices
Solar Power
Electric Vehicle
Wind Power Plant

Intelligent Transportation Systems



What is network security?

confidentiality: only sender, intended receiver should "understand" message contents

- sender encrypts message
- receiver decrypts message

authentication: sender, receiver want to confirm identity of each other

message integrity: sender, receiver want to ensure message not altered (in transit, or afterwards) without detection

access and availability: services must be accessible and available to users

There are bad guys (and girls) out there!

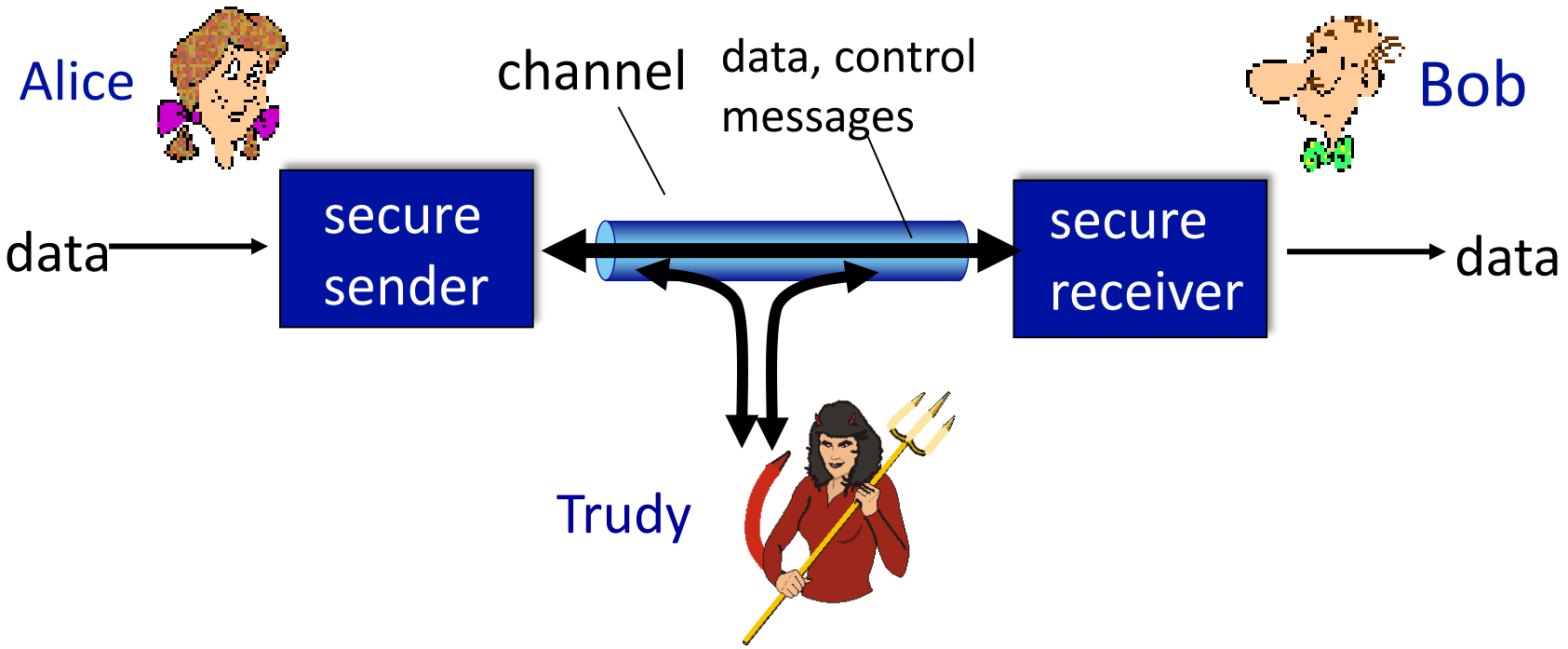
Q: What can a “bad guy” do?

A: A lot!

- eavesdrop: intercept messages
- actively insert messages into connection
- impersonation: can fake (spoof) source address in packet (or any field in packet)
- hijacking: “take over” ongoing connection by removing sender or receiver, inserting himself in place
- denial of service: prevent service from being used by others (e.g., by overloading resources)

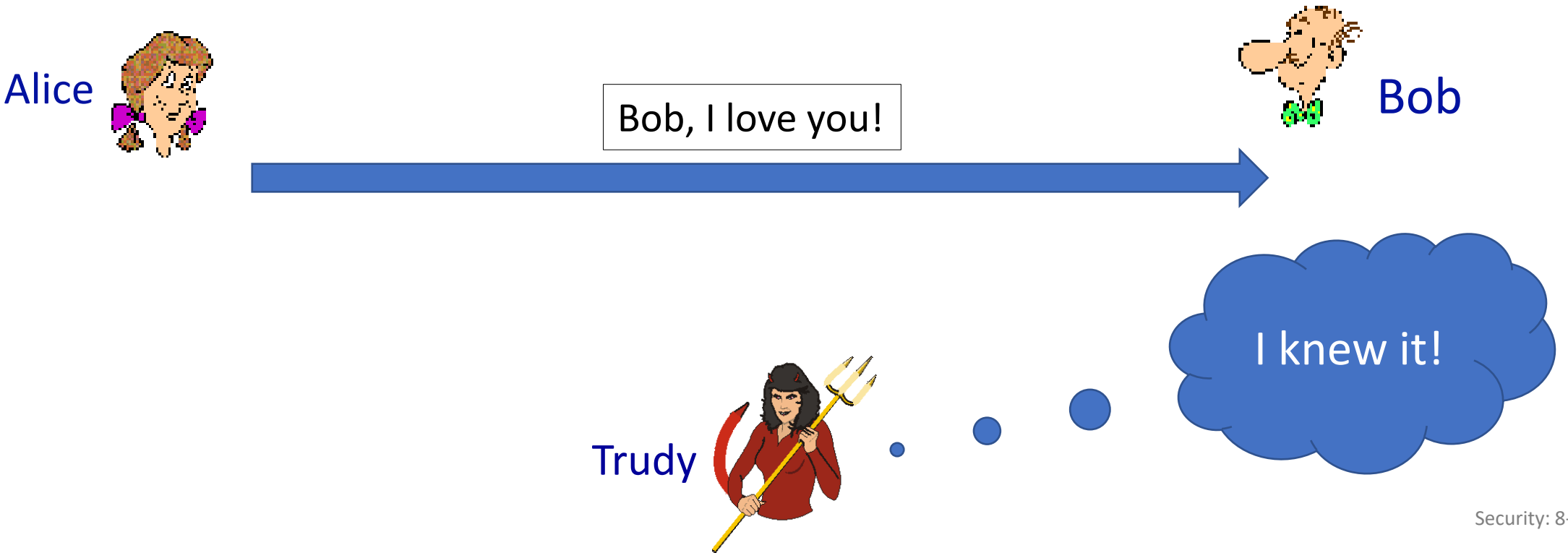
Friends and enemies: Alice, Bob, Trudy

- well-known in network security world
- Bob, Alice (lovers!) want to communicate “securely”
- Trudy (intruder) may intercept, delete, add messages



Without confidentiality

- Trudy (intruder) eavesdrops on messages



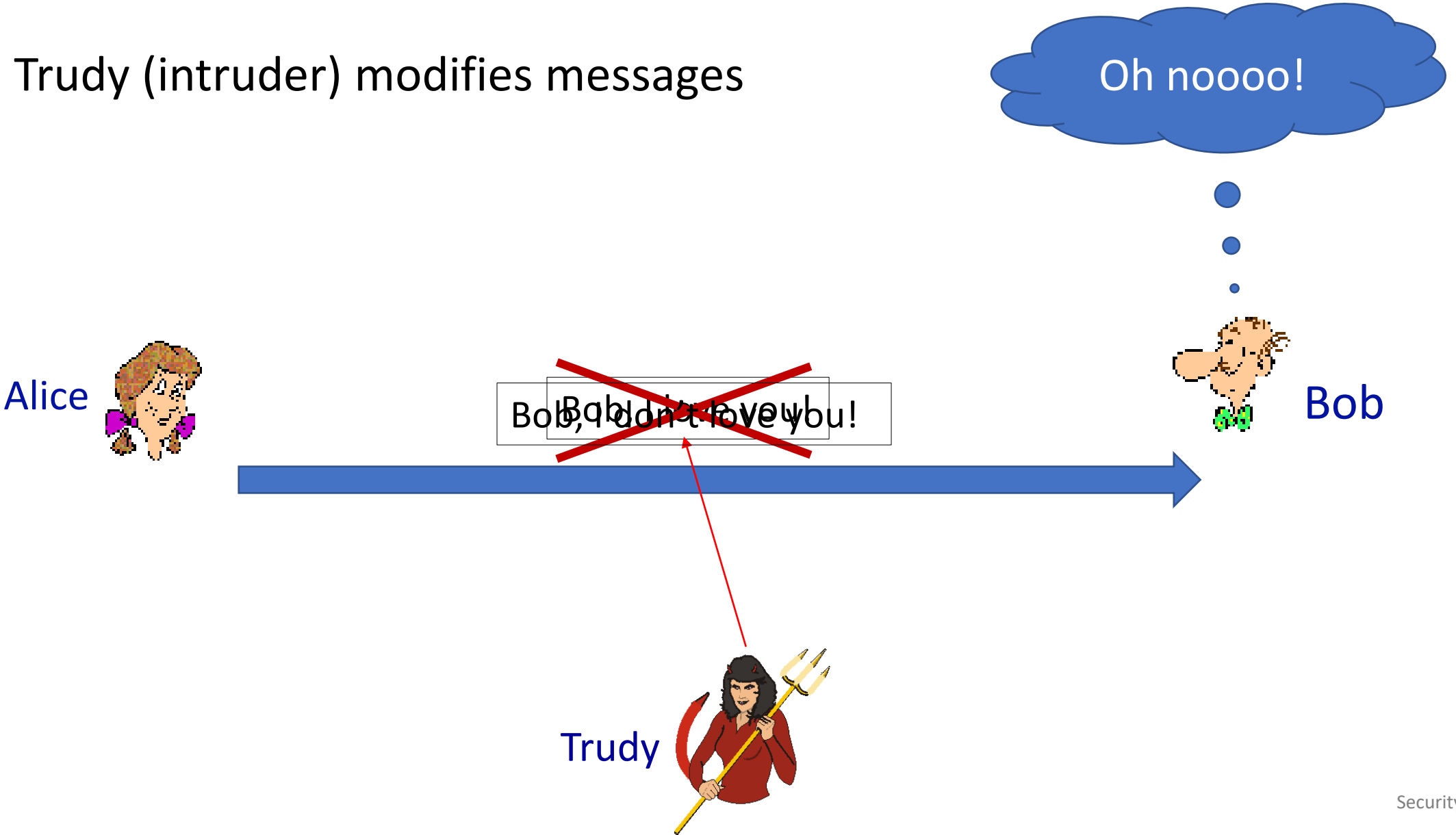
Confidentiality

- With encryption, Trudy (intruder) cannot eavesdrop on messages



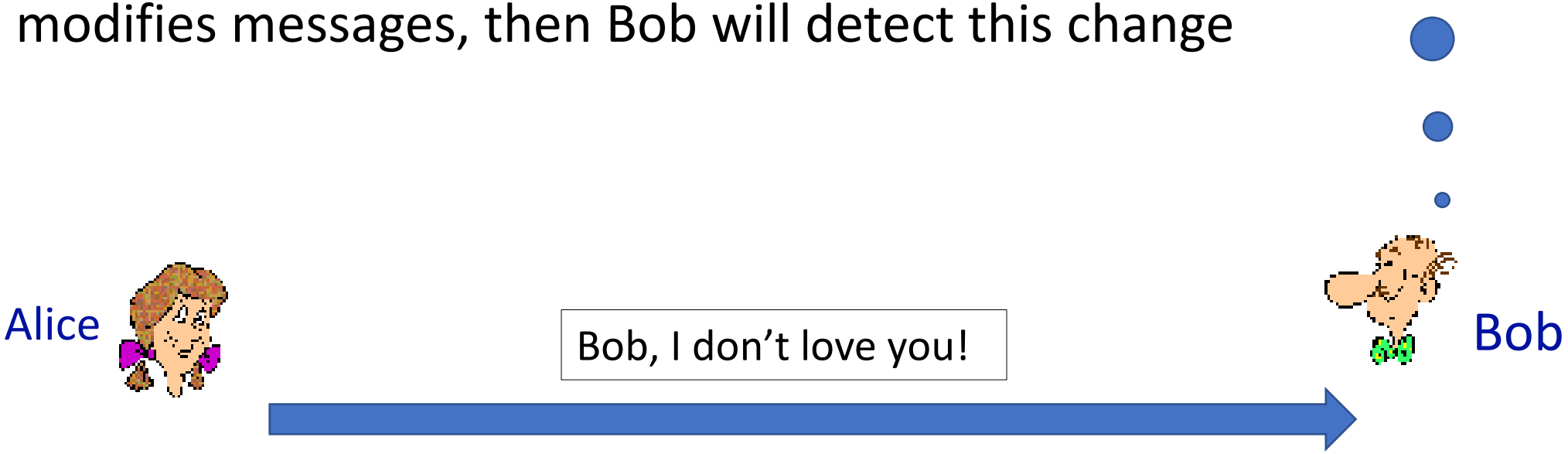
Without integrity

- Trudy (intruder) modifies messages



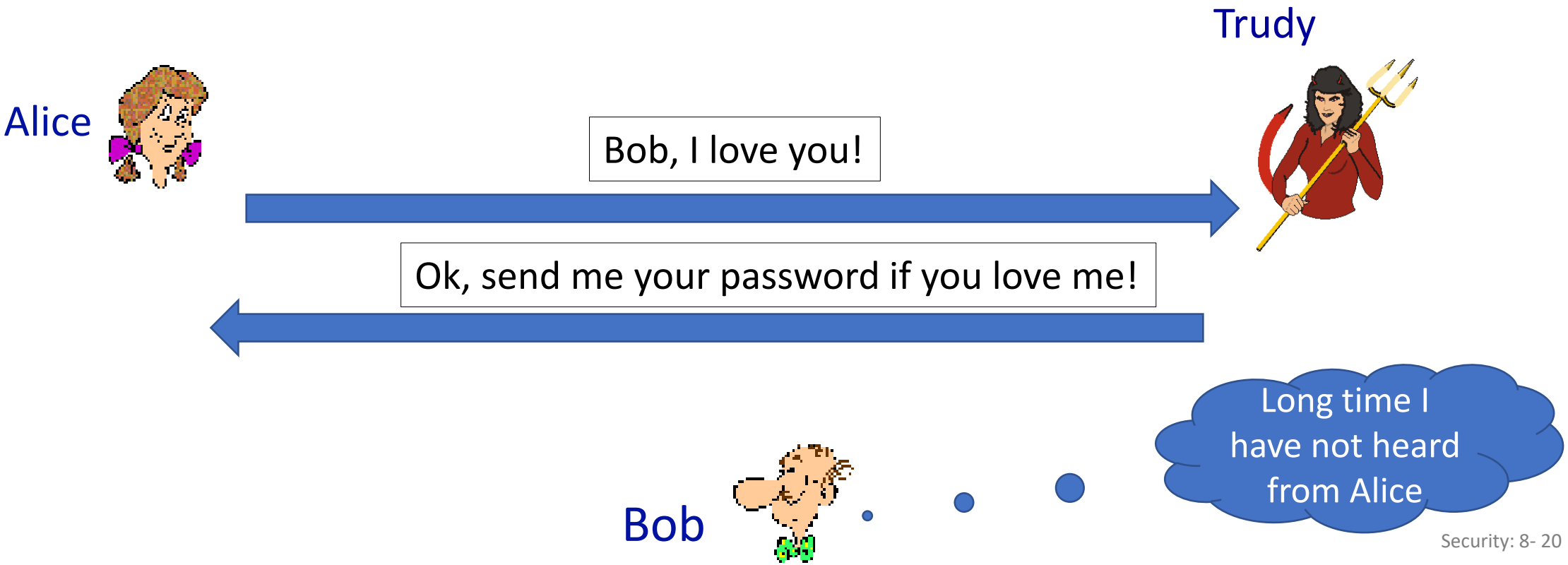
With integrity

- With integrity protection, if Trudy (intruder) modifies messages, then Bob will detect this change



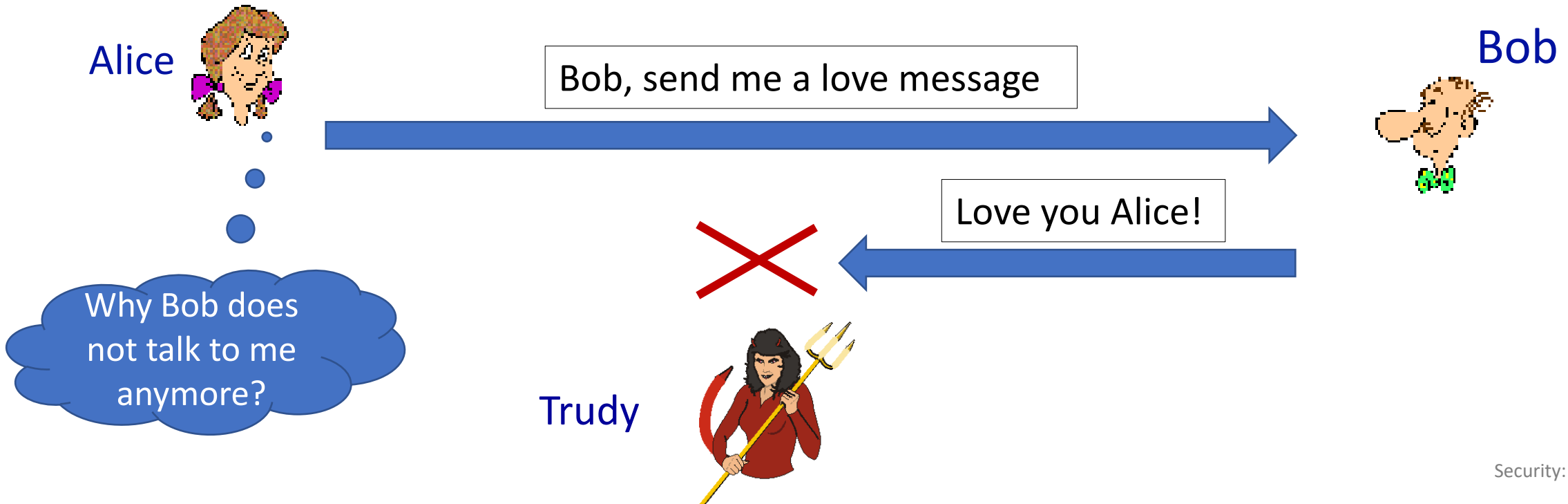
Without authentication

- Without authentication, Alice does not verify the identity of the recipient (is it Bob or someone else?)



No access and availability

- Loss of availability means that someone or something is stopping Alice to access services (in this scenario to receive Bob message)

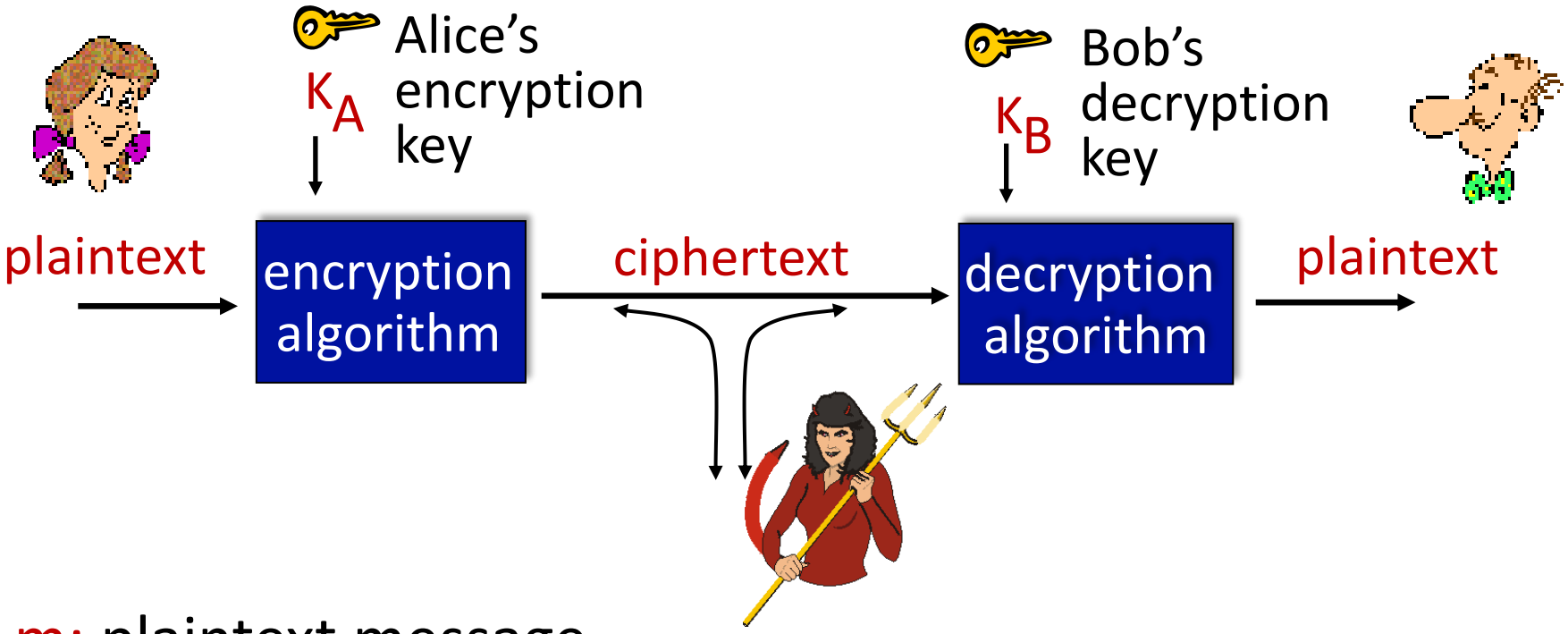


Friends and enemies: Alice, Bob, Trudy

Who might Bob and Alice be?

- … well, *real-life* Bobs and Alices!
- Web browser/server for electronic transactions (e.g., on-line purchases)
- on-line banking client/server
- DNS servers
- BGP routers exchanging routing table updates
- other examples?

The language of cryptography



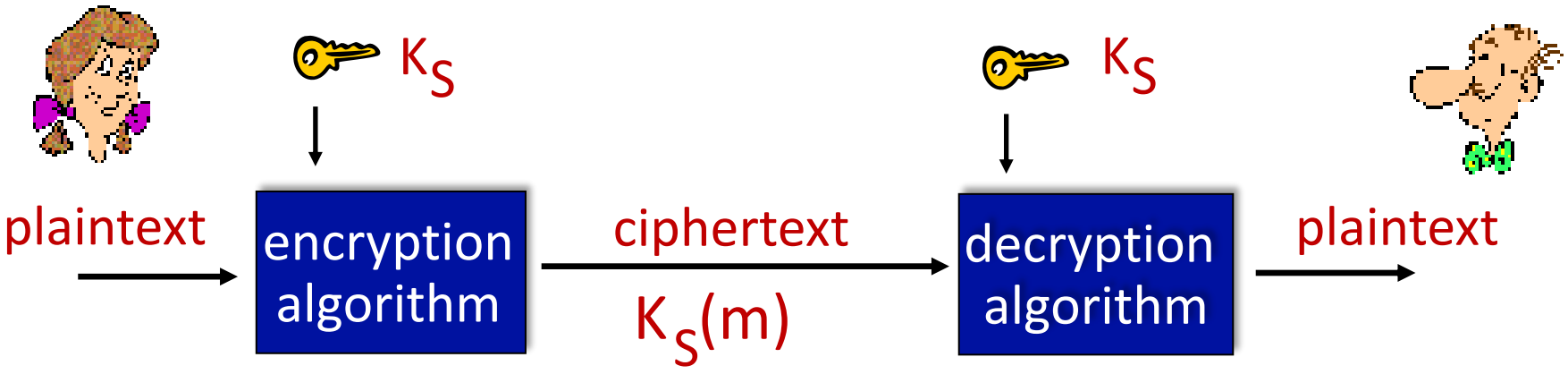
m: plaintext message

$K_A(m)$: ciphertext, encrypted with key K_A

$m = K_B(K_A(m))$


WE WILL ATTACK TOMORROW EVENING
PLAIN TEXT
CIPHER
CIPHERTEXT
ZHZLOOWWDFNWRPRZHYHQLQJ

Symmetric key cryptography



symmetric key crypto: Bob and Alice share same (symmetric) key: K

- *e.g.,* key is knowing substitution pattern in mono alphabetic substitution cipher

Q: how do Bob and Alice agree on key value?

Simple encryption scheme

substitution cipher: substituting one thing for another

- monoalphabetic cipher: substitute one letter for another



e.g.: Plaintext: bob. i love you. alice

ciphertext: nkn. s gktc wky. mgsbc

Encryption key: mapping from set of 26 letters to set of 26 letters

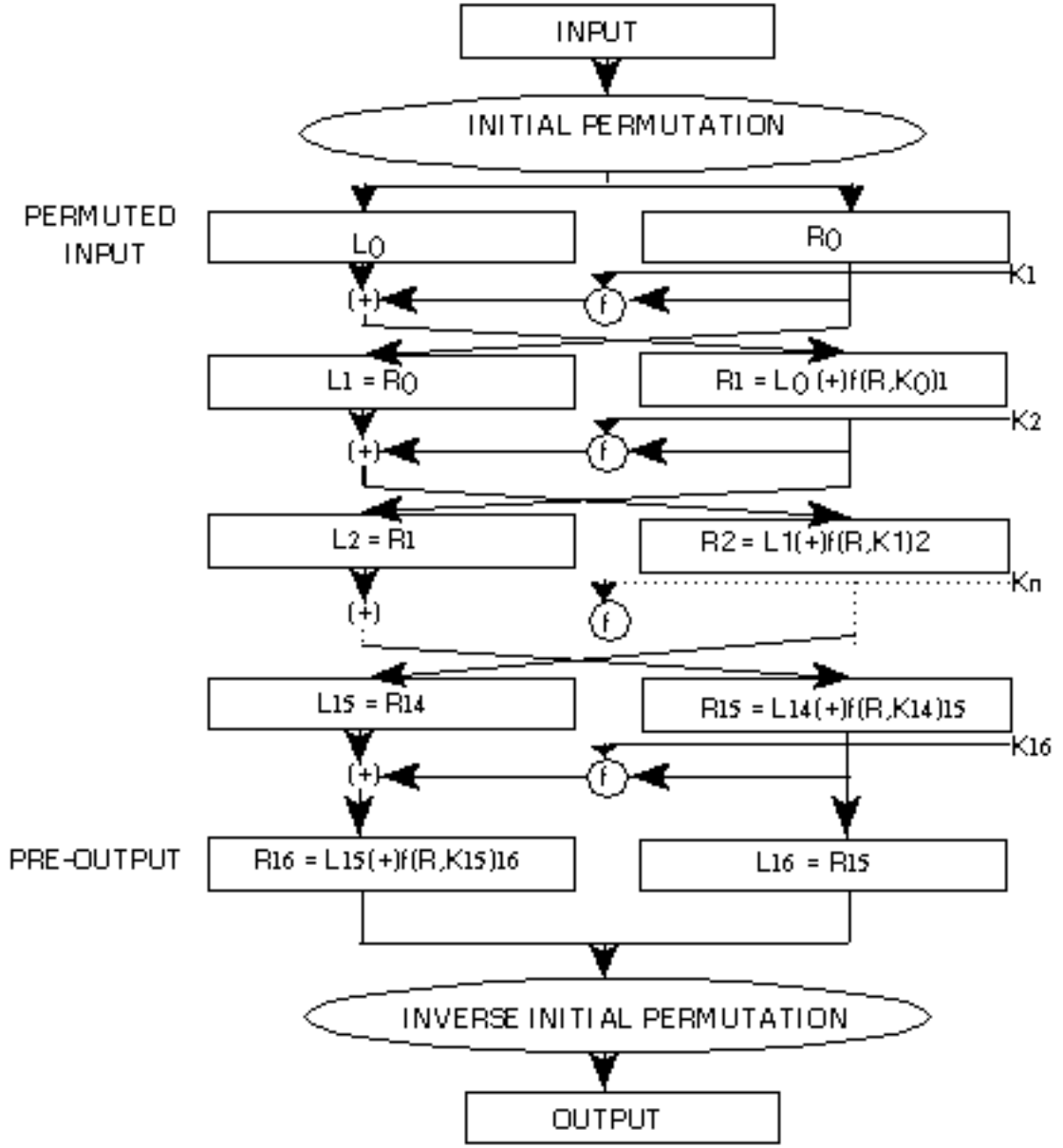
Modern Cryptography – Block Ciphers

Symmetric key crypto: DES

DES: Data Encryption Standard

- US encryption standard [NIST 1993]
- 56-bit symmetric key, 64-bit plaintext input
- block cipher with cipher block chaining
- how secure is DES?
  - DES Challenge: 56-bit-key-encrypted phrase decrypted (brute force) in less than a day
  - no known good analytic attack
- making DES more secure:
  - 3DES: encrypt 3 times with 3 different keys

DES


INPUT
INITIAL PERMUTATION
PERMUTED INPUT
L0
R0
K1
(+)
f
L1 = R0
R1 = L0 (+)f(R,K0)1
K2
L2 = R1
R2 = L1(+)f(R,K1)2
Kn
L15 = R14
R15 = L14(+)f(R,K14)15
K16
PRE-OUTPUT
R16 = L15(+)f(R,K15)16
L16 = R15
INVERSE INITIAL PERMUTATION
OUTPUT

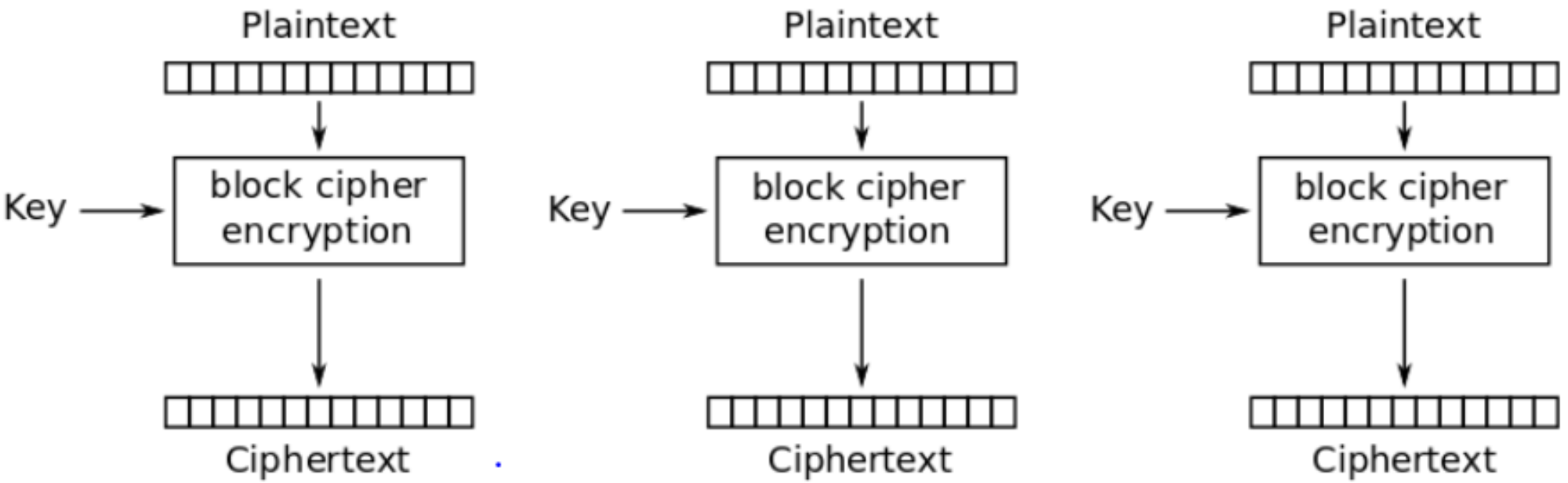
AES: Advanced Encryption Standard

- symmetric-key NIST standard, replaced DES (Nov 2001)
- processes data in 128 bit blocks
- 128, 192, or 256 bit keys
- brute force decryption (try each key) taking 1 sec on DES, takes 149 trillion years for AES

AES parameters

- Ο AES είναι ένα block cipher (κρυπτογραφεί ανα block δεδομένων).
- Υπάρχουν πολλαπλά modes για τον AES την κρυπτογράφηση των blocks
  - ECB, CBC, GCF, XTS, EAX, CFB….
- Μήκος block = 128 bits
- Μήκος κλειδιού = 128 ή 192 ή 256 bits
- Μήκος Initialization Vector = 128 bits (αν ορίζεται)

ECB encryption

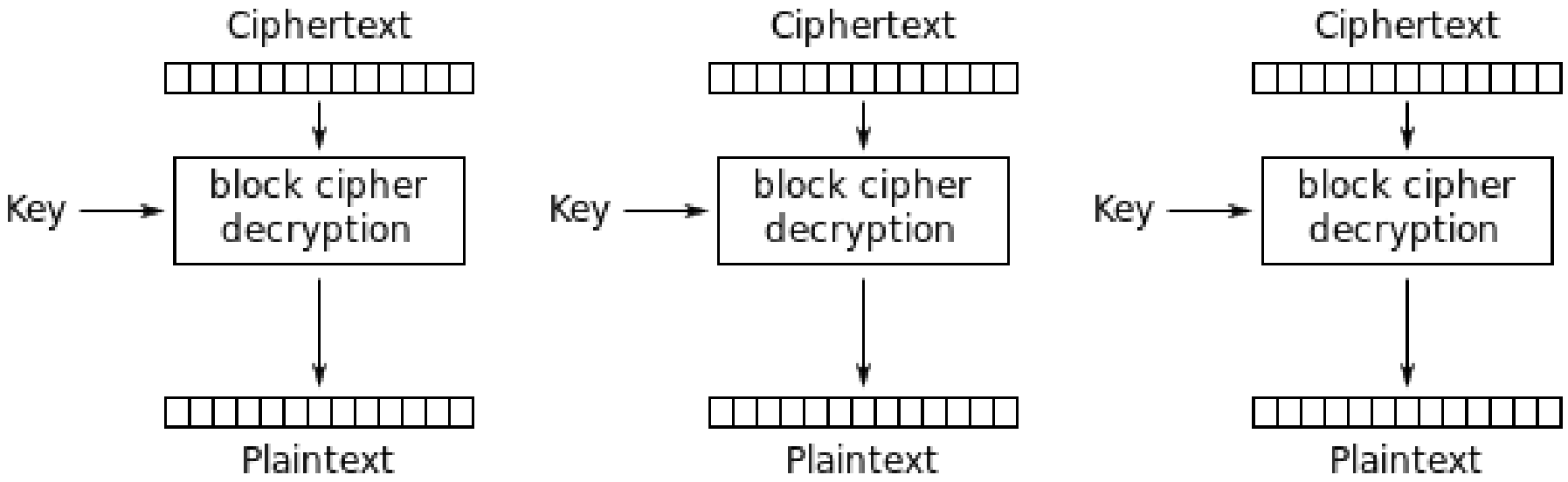


Electronic Codebook (ECB) mode encryption

Padding

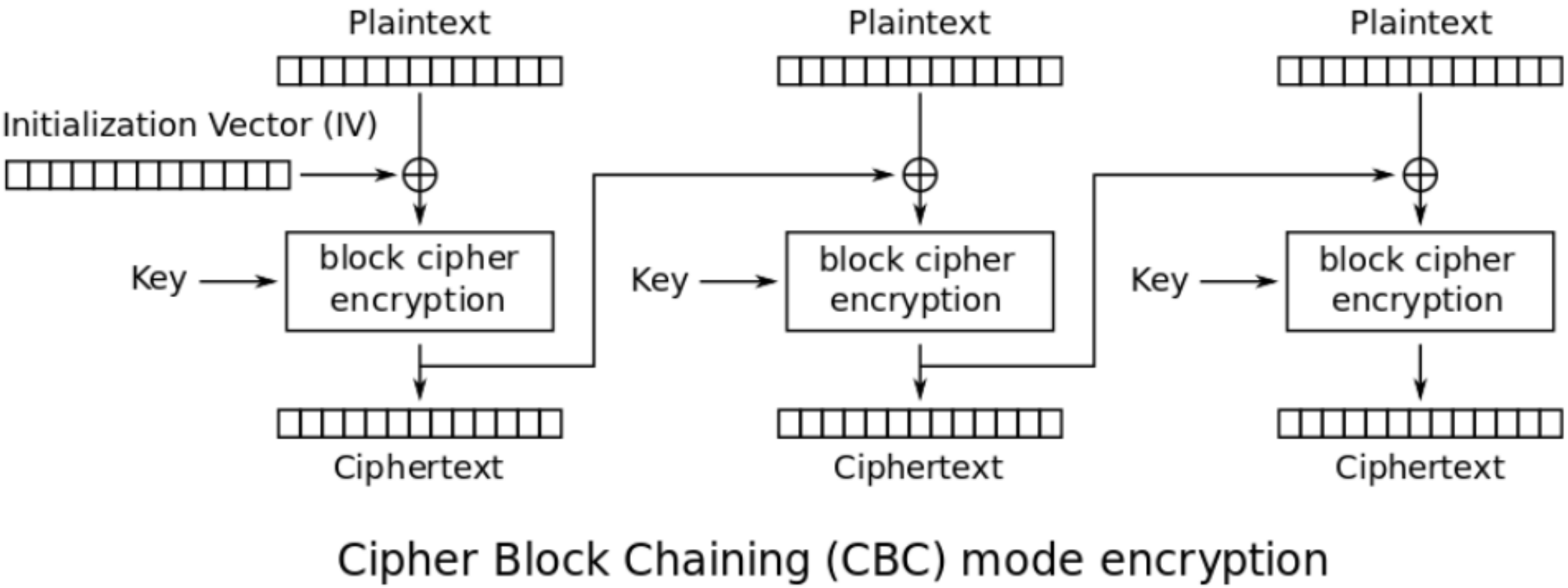
- Υπάρχει άλλη μια παράμετρο για τον AES...
- Padding: Όταν τα blocks δεν ειναι πολλαπλάσια των 128 bites, αναγκαστικά, πρέπει να προσθέσουμε επιπλέον bits για να είναι το τελευταίο block ίσο με 128 bits.
- Τι δεδομένα θα προσθέσουμε;;;; Τυχαία;;; Μηδενικά;;;;
  - Zero, pkcs7…

ECB decryption



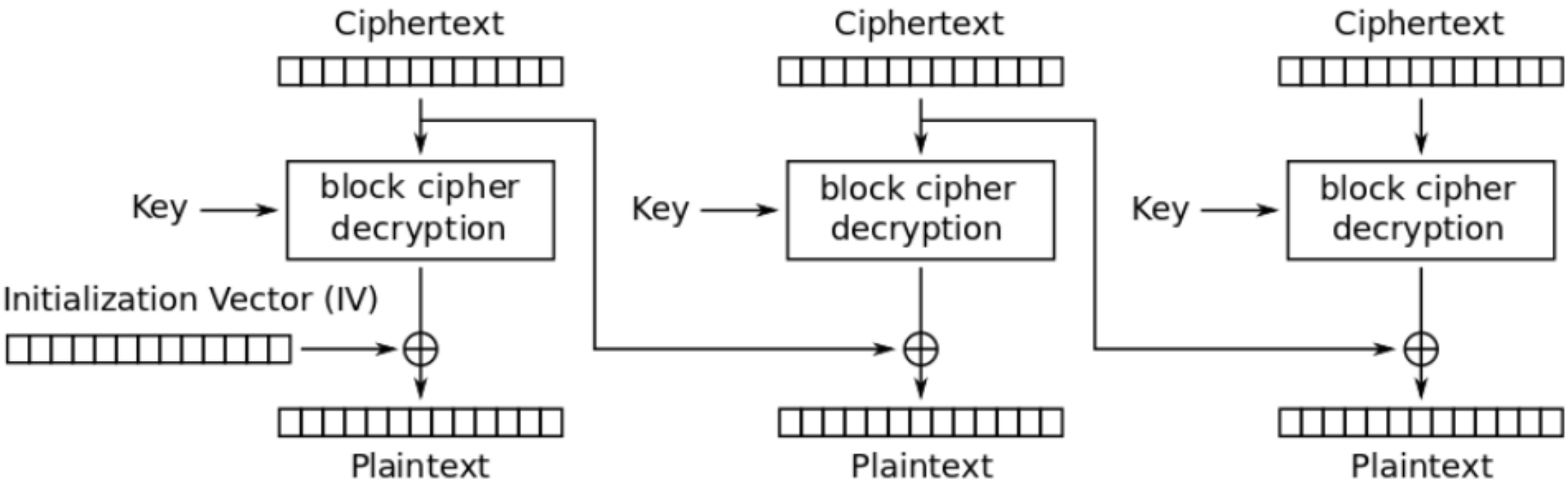
Electronic Codebook (ECB) mode decryption

CBC encryption



Cipher Block Chaining (CBC) mode encryption

CBC decryption



Cipher Block Chaining (CBC) mode decryption

Key Agreement

How Alice and Bob agree on a key?

1. Physical/manual exchange

2. Public key cryptography
   a) Encryption of symmetric key with a public key
   b) Diffie Hellman

Examples for manual/physical exchange

- Mobile networks (SIM card) (the key is inside the SIM and the operator stores it in its database)

- Wi-Fi router key (the password will be used for key generation)

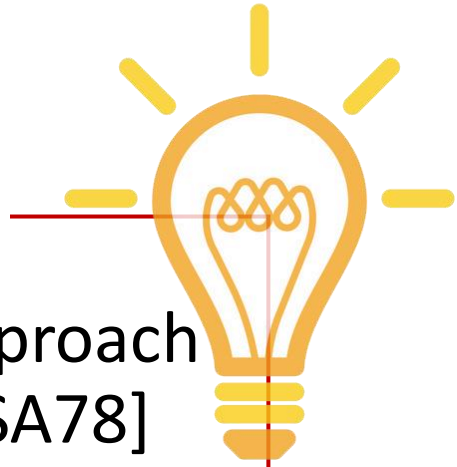
Public Key Cryptography

Public Key Cryptography

symmetric key crypto:

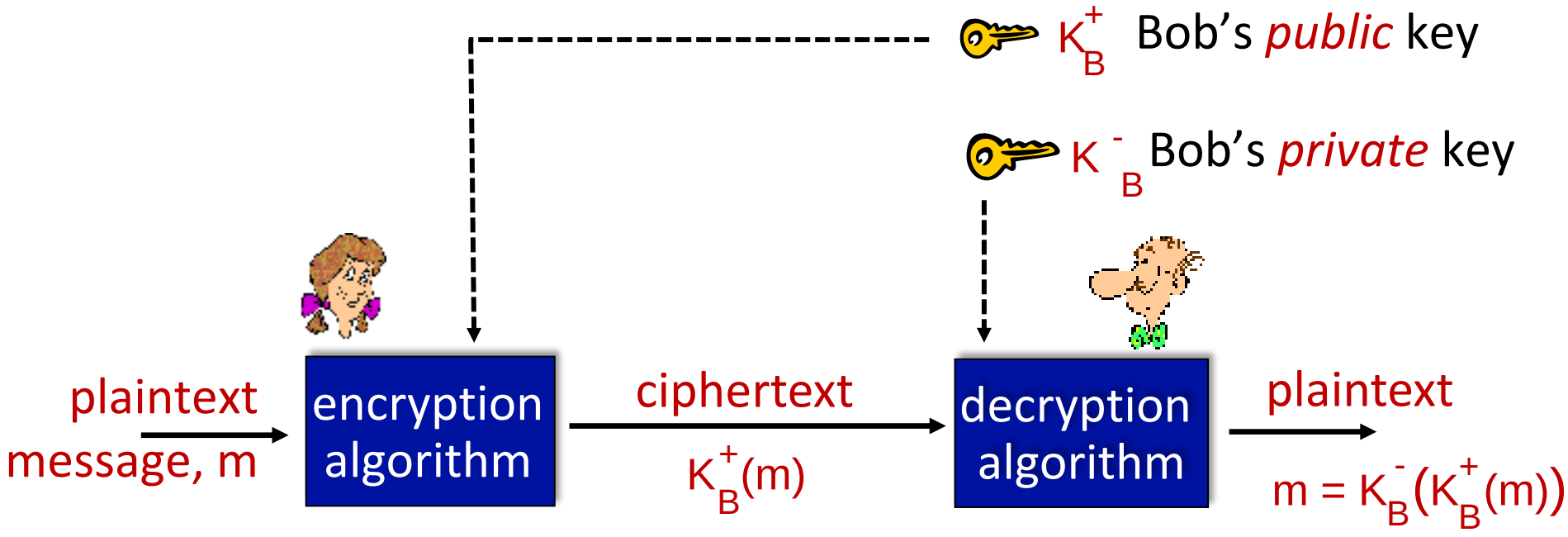
- requires sender, receiver know shared secret key
- Q: how to agree on key in first place (particularly if never “met”)?

public key crypto



- *radically* different approach [Diffie-Hellman76, RSA78]
- sender, receiver do *not* share secret key
- *public* encryption key known to *all*
- *private* decryption key known only to receiver

Public Key Cryptography



Wow - public key cryptography revolutionized 2000-year-old (previously only symmetric key) cryptography!

- similar ideas emerged at roughly same time, independently in US and UK (classified)

Public key encryption algorithms

requirements:

1. need $K_B^+(\cdot)$ and $K_B^-(\cdot)$ such that

$$K_B^-(K_B^+(m)) = m$$

2. given public key K_B^+, it should be impossible to compute private key K_B^-

RSA: Rivest, Shamir, Adelson algorithm

RSA in practice: session keys

- exponentiation in RSA is computationally intensive
- DES is at least 100 times faster than RSA
- use public key crypto to establish secure connection, then establish second key – symmetric session key – for encrypting data

session key, K_S

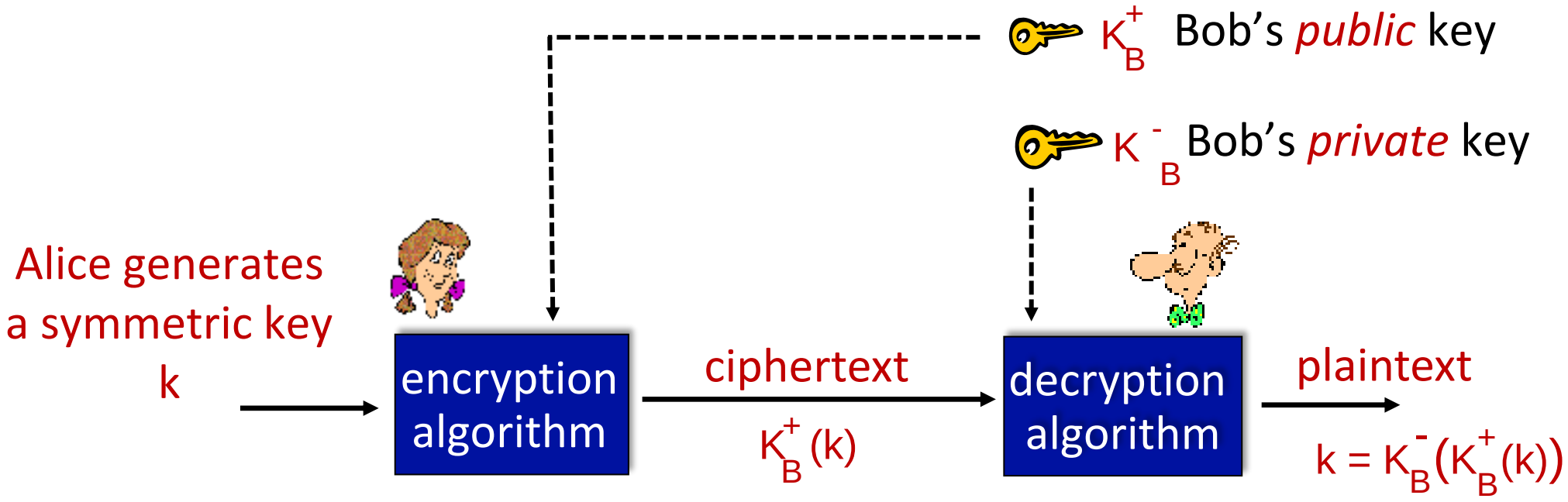
- Bob and Alice use RSA to exchange a symmetric session key K_S
- once both have K_S, they use symmetric key cryptography

How Alice and Bob agree on a key?

1. Physical/manual exchange

2. Public key cryptography
   a) **Encryption of symmetric key with a public key**
   b) Diffie Hellman

Agree on a symmetrical key



Public key encryption algorithms

- RSA
- ElGamal encryption system
- Elliptic curve ElGamal encryption

Digital Signatures

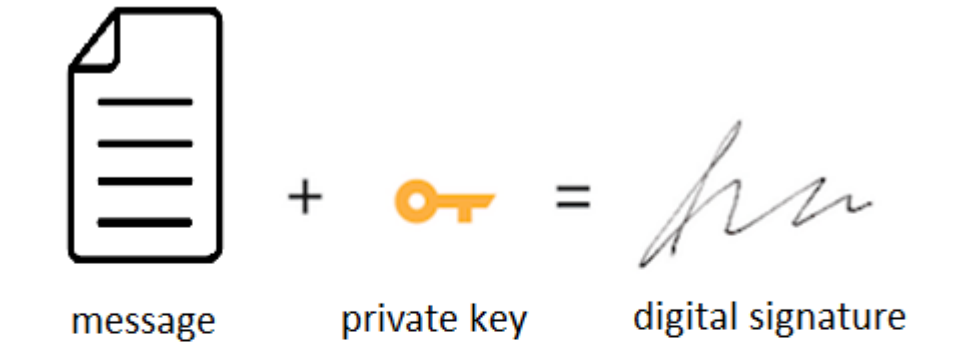
RSA: another important property

The following property will be *very* useful later:

$$\underbrace{K_B^-(K_B^+(m))}_{\text{use public key first, followed by private key}} = m = \underbrace{K_B^+(K_B^-(m))}_{\text{use private key first, followed by public key}}$$

result is the same!

Digital Signatures

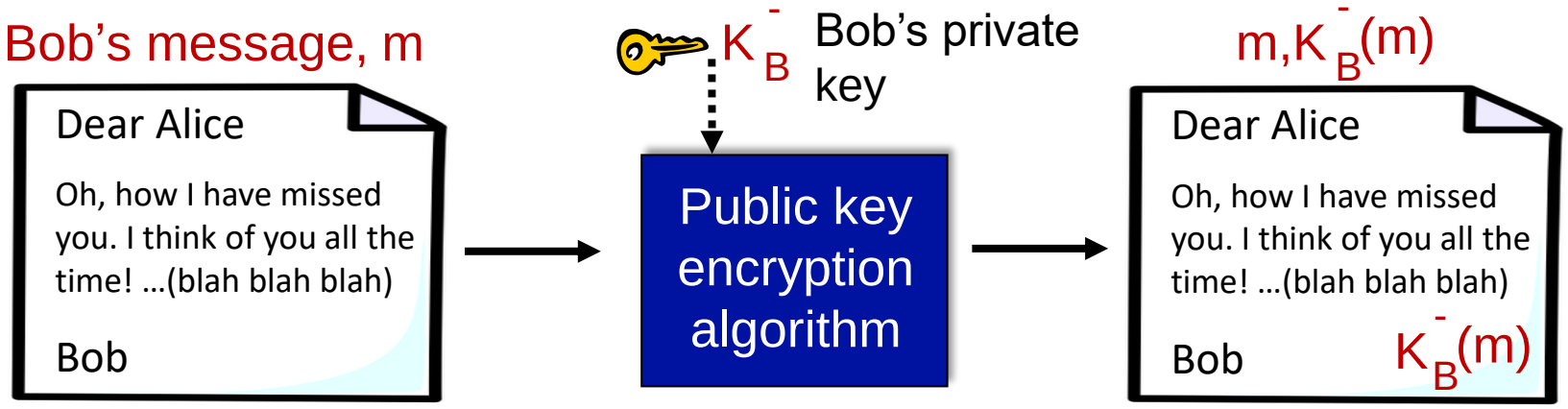




Digital signatures

cryptographic technique analogous to hand-written signatures:

- sender (Bob) digitally signs document: he is document owner/creator.
- *verifiable, nonforgeable:* recipient (Alice) can prove to someone that Bob, and no one else (including Alice), must have signed document
- simple digital signature for message m:
  - Bob signs m by encrypting with his private key K_B, creating "signed" message, $K_B^-(m)$



Digital signatures

- Suppose Alice receives msg m, with signature: m, $K_B^-(m)$
- Alice verifies m signed by Bob by applying Bob's public key K_B^+ to $K_B^-(m)$ then checks $K_B^+(K_B^-(m)) = m$.
- If $K_B^+(K_B^-(m)) = m$, whoever signed m must have used Bob's private key

Alice thus verifies that:

- Bob signed m
- no one else signed m
- Bob signed m and not m'

non-repudiation:

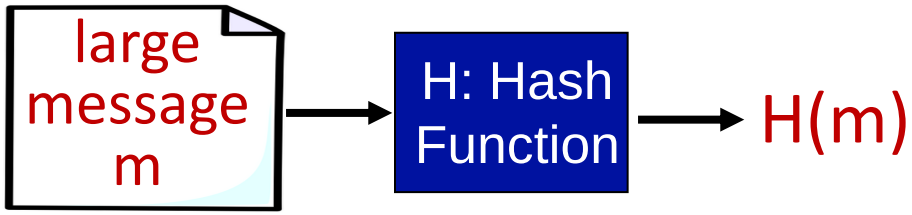
- ✓ Alice can take m, and signature $K_B^-(m)$ to court and prove that Bob signed m

Message digests with hash functions

computationally expensive to public-key-encrypt long messages

goal: fixed-length, easy- to-compute digital “fingerprint”

- apply hash function H to *m*, get fixed size message digest, *H(m)*



Hash function properties:

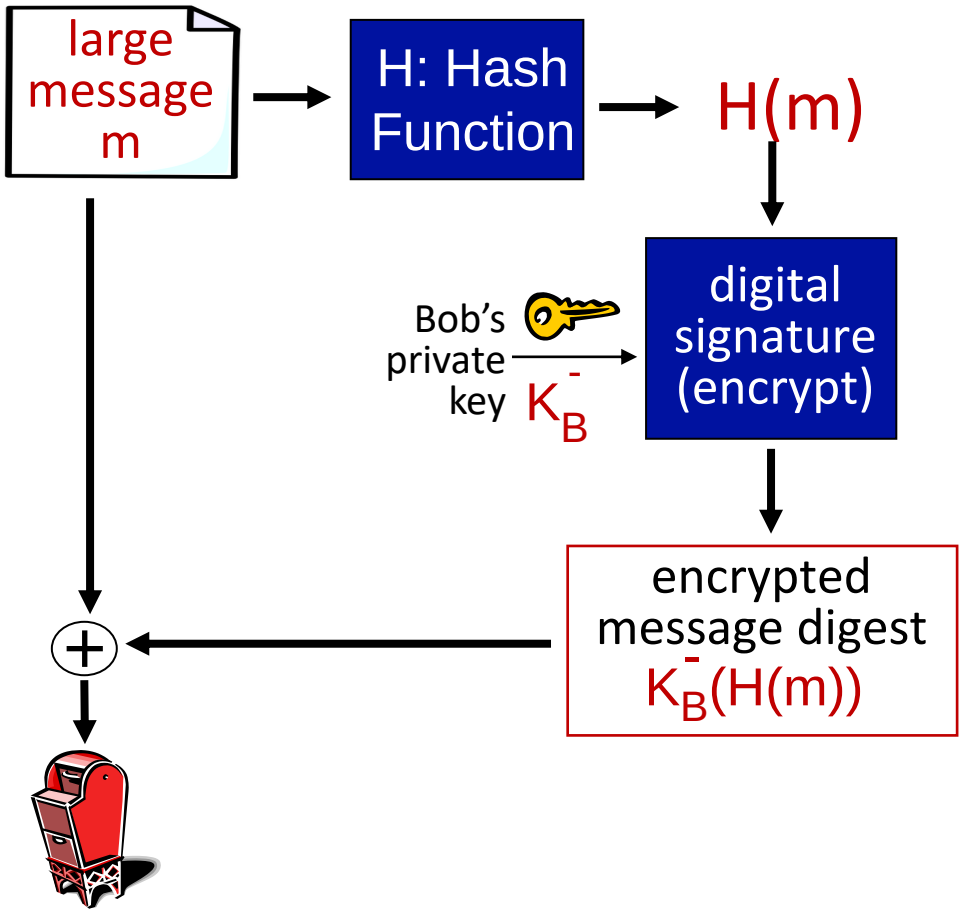
- many-to-1
- produces fixed-size msg digest (fingerprint)
- given message digest *x*, computationally infeasible to find *m* such that *x* = *H(m)*

Hash function algorithms

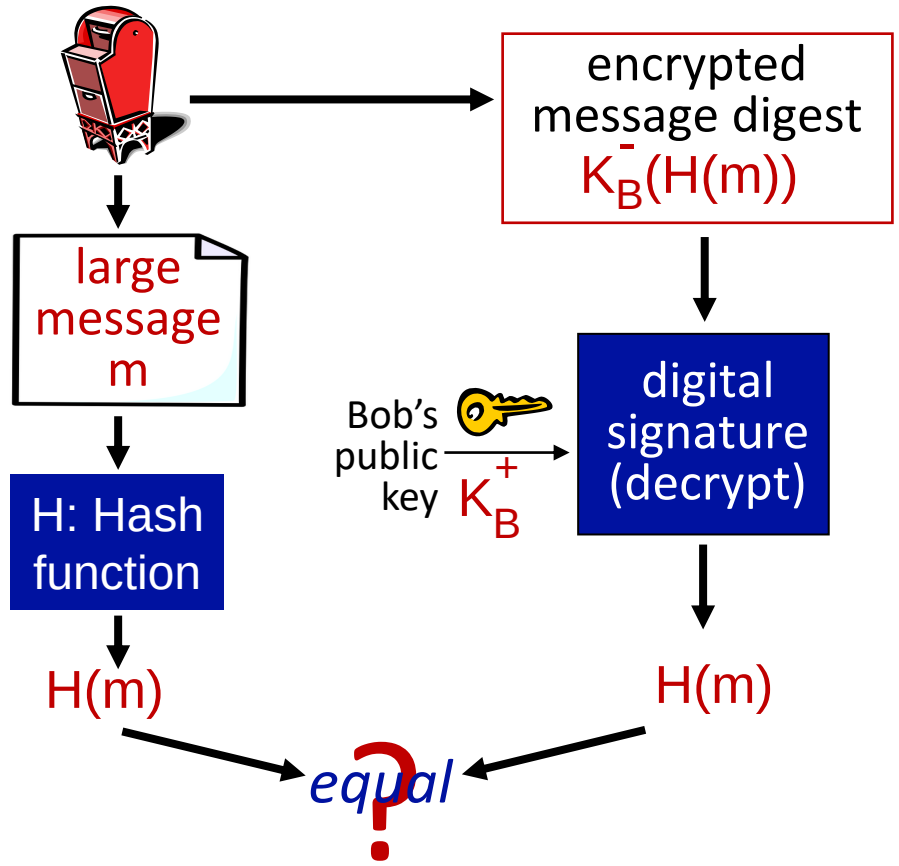
- MD5 hash function was widely used
  - computes 128-bit message digest.
- SHA-1 was also used
  - 160-bit message digest
- Modern hash functions are: SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512), SHA-3 (SHA3-224, SHA3-256, SHA3-384, SHA3-512)

Digital signature = signed message digest

Bob sends digitally signed message:



Alice verifies signature, integrity of digitally signed message:



Digital Signature Algorithms

- RSA (Rivest–Shamir–Adleman)
- DSA (Digital Signature Algorithm) (based on ElGamal signature scheme)
- ECDSA (Elliptic Curve DSA)
- EdDSA (Edwards-Curve DSA)

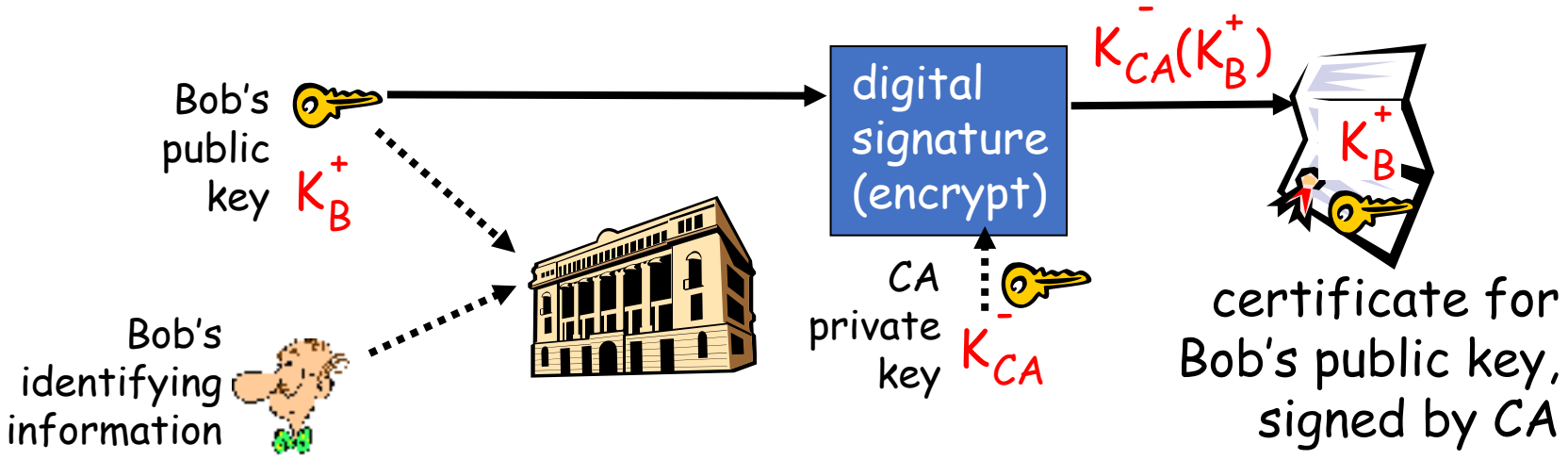
Certifcates

Trusting public keys

- How Bob will obtain the public key of Alice to verify a signature or to send a symmetric key encrypted with the public key of Alice?
  - Maybe Alice can send to him…but
- How Bob verifies the validity of a public key?
- Can he obtain the public key from a central trusted repository of public keys?
  - This may work for an organization

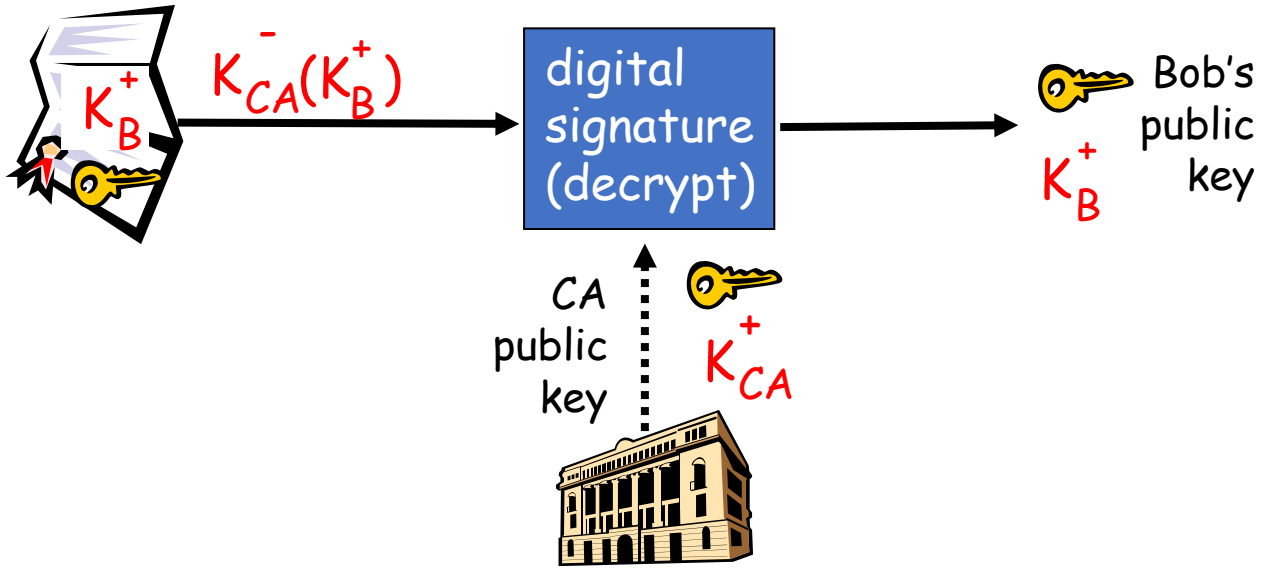
Certification Authorities

- Certification Authority (CA): binds public key to particular entity, E.
- E registers its public key with CA.
  - E provides "proof of identity" to CA.
  - CA creates certificate binding E to its public key.
  - certificate containing E's public key digitally signed by CA: CA says "This is E's public key."

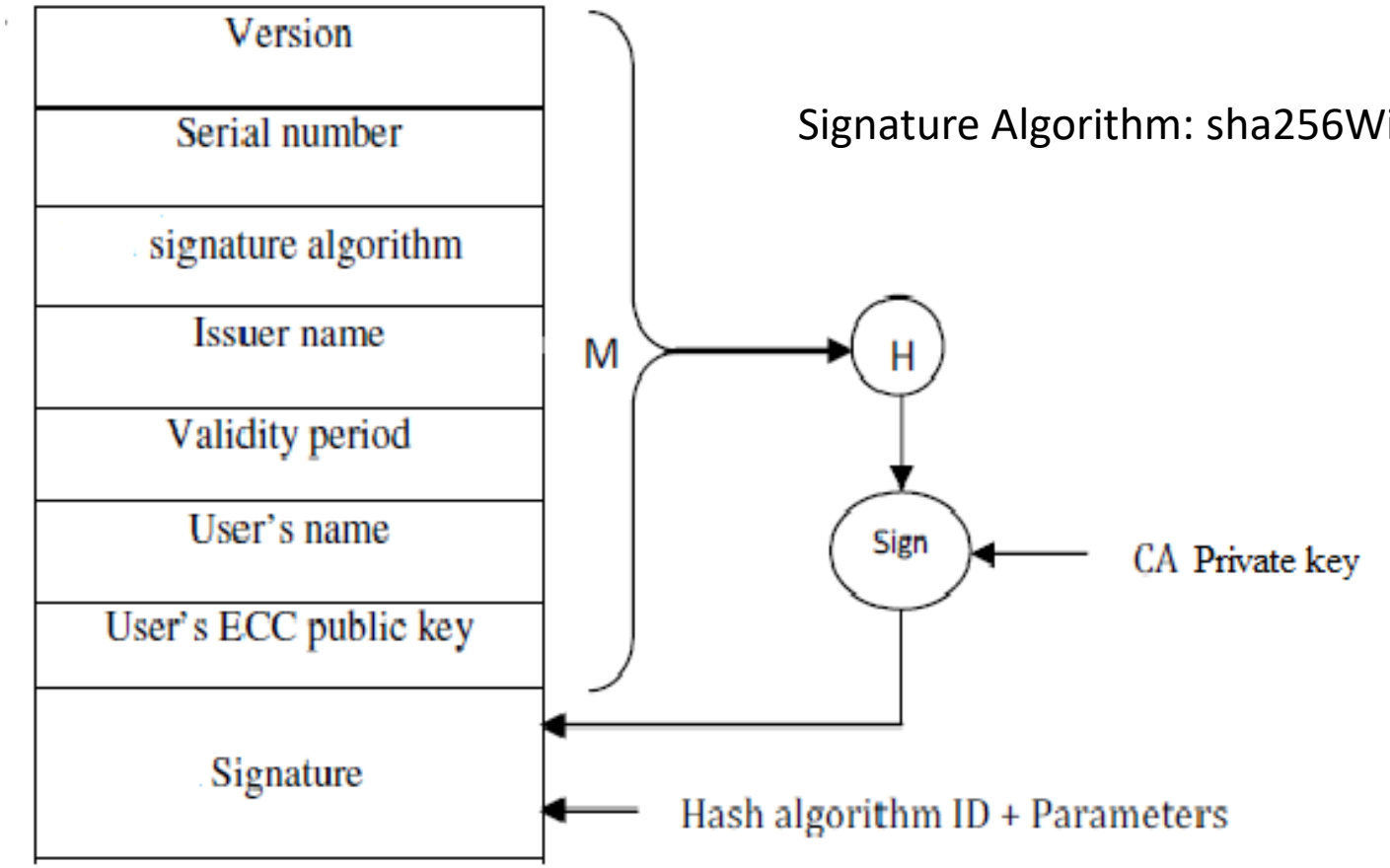


Certification Authorities

- when Alice wants Bob's public key:
  - gets Bob's certificate (Bob or elsewhere).
  - apply CA's public key to Bob's certificate, get Bob's public key



Signature in certificates



Signature Algorithm: sha256WithRSAEncryption