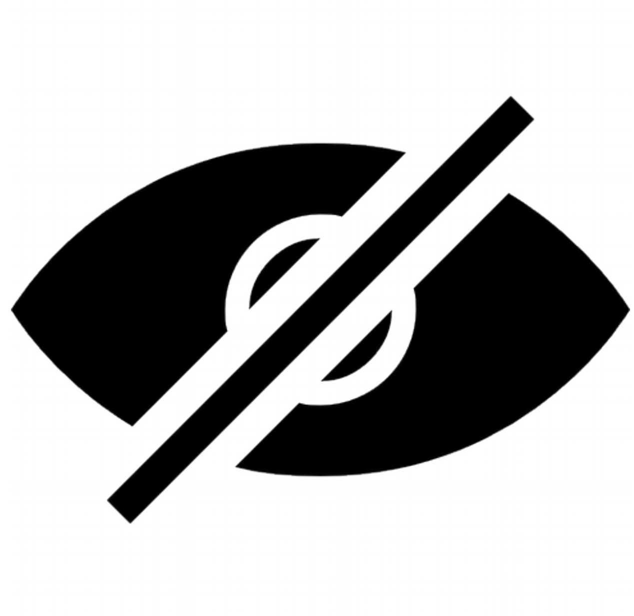
Applications of cryptography



BUITA lecture 9 (ITS 3)
October 11th, 2018

Niels Jørgensen
(nielsj@ruc.dk)

Agenda & *learning goals*

1. Technology: Access control with passwords and hashing

2. Technology: Access control with digital signatures & digital certificates
   - Digital signature-articel + wikipedia-entry

Understand basics of technology

3. Law: EU's General Data Protection Regulation (including psedonymization)
   - GDPR articles

Ability to choose appropriate technology

4. Usable Security (Why Johnny Can't Encrypt)
   - Whitten & Tygar paper

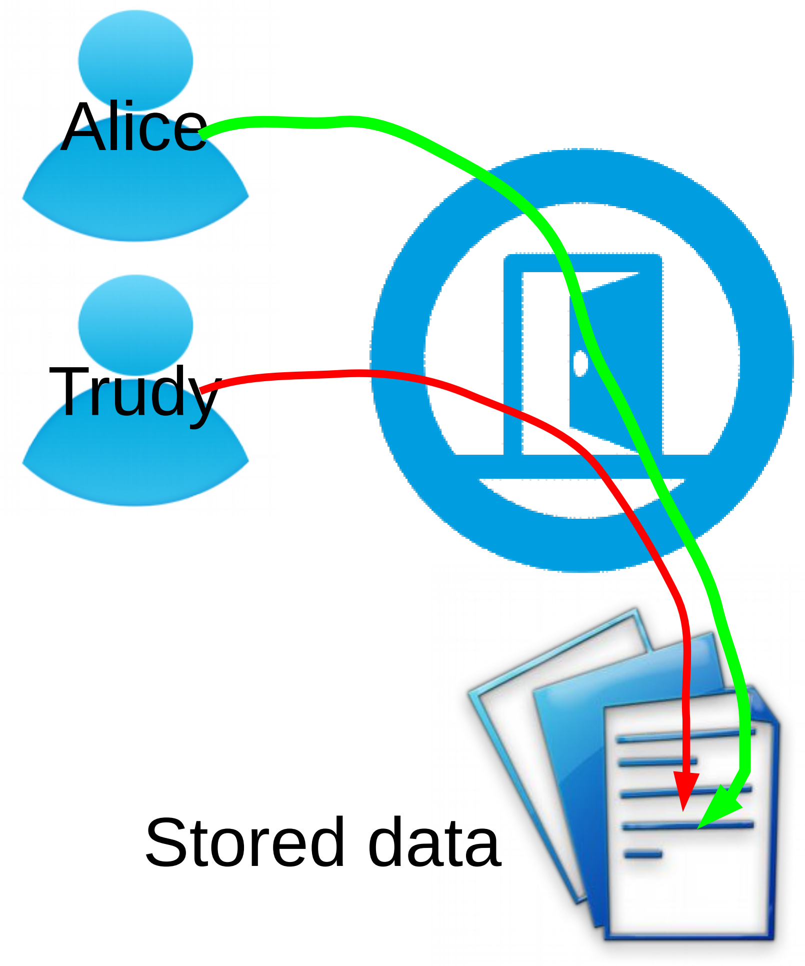
Understand basic challengas

Protection of stored data

Encryption
- sometimes merely anonymization or pseudonymization

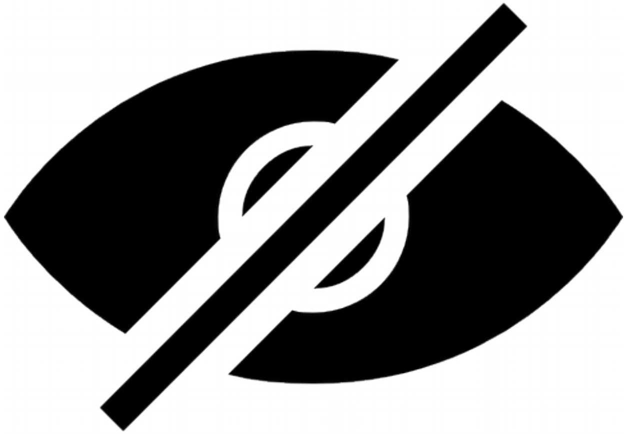
Sound organizational procedures
- “don’t bring unencrypted data to your private home”

Access control is
- authentication
- so part of “CIA++” goals

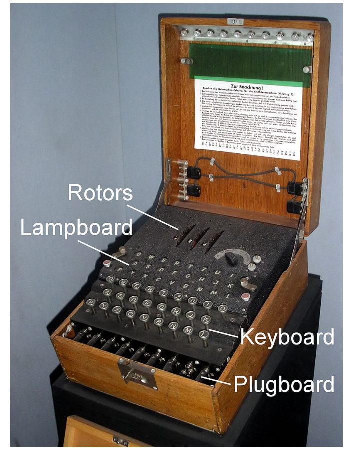


Recap

Encryption is secure



History of cryptography



Caeser cipher

Vigenère cipher

Enigma

DES, RSA

AES

Quantum cryptogr.

0 — 1500 — 1940 — 1975 — 2000 — 2030

human cryptography

machine c.

computer c.

?!??

AES (2000) is secure

AES is “unbreakable in practice”

Brute force: try all keys
- similar to a attack on cipter text CAMZAAPWCTLJMUILMIEIZM
- intelligent methods are not better (for AES)

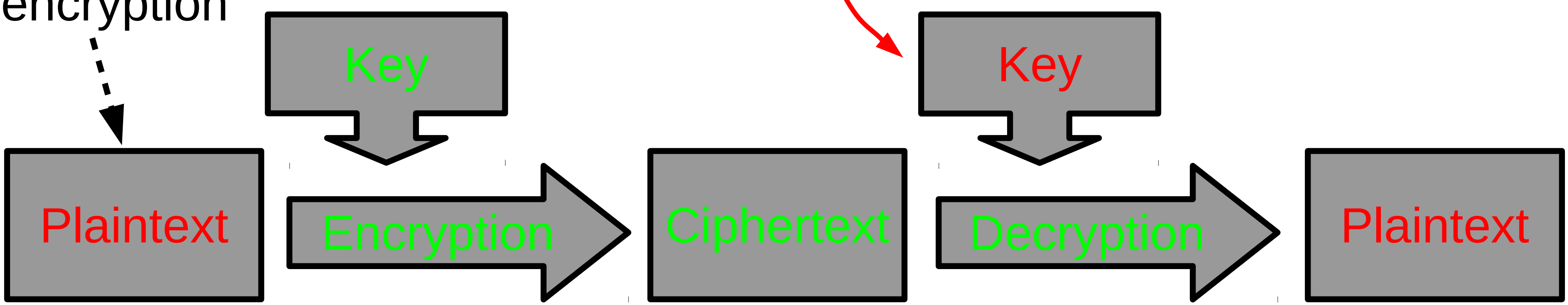
Safe key size
- 128 bits key size is generally considered to be enough
- 256 recommended by NSA
  - to protect against quantum computing

Unsafe key size
- DES was broken brute force
- 56 bits unsafe in 1998

Asymmetric encryption = solution to key exchange problem

Two different keys
- 1 public
- 1 private (secret)
- private key only works with its corresponding public key

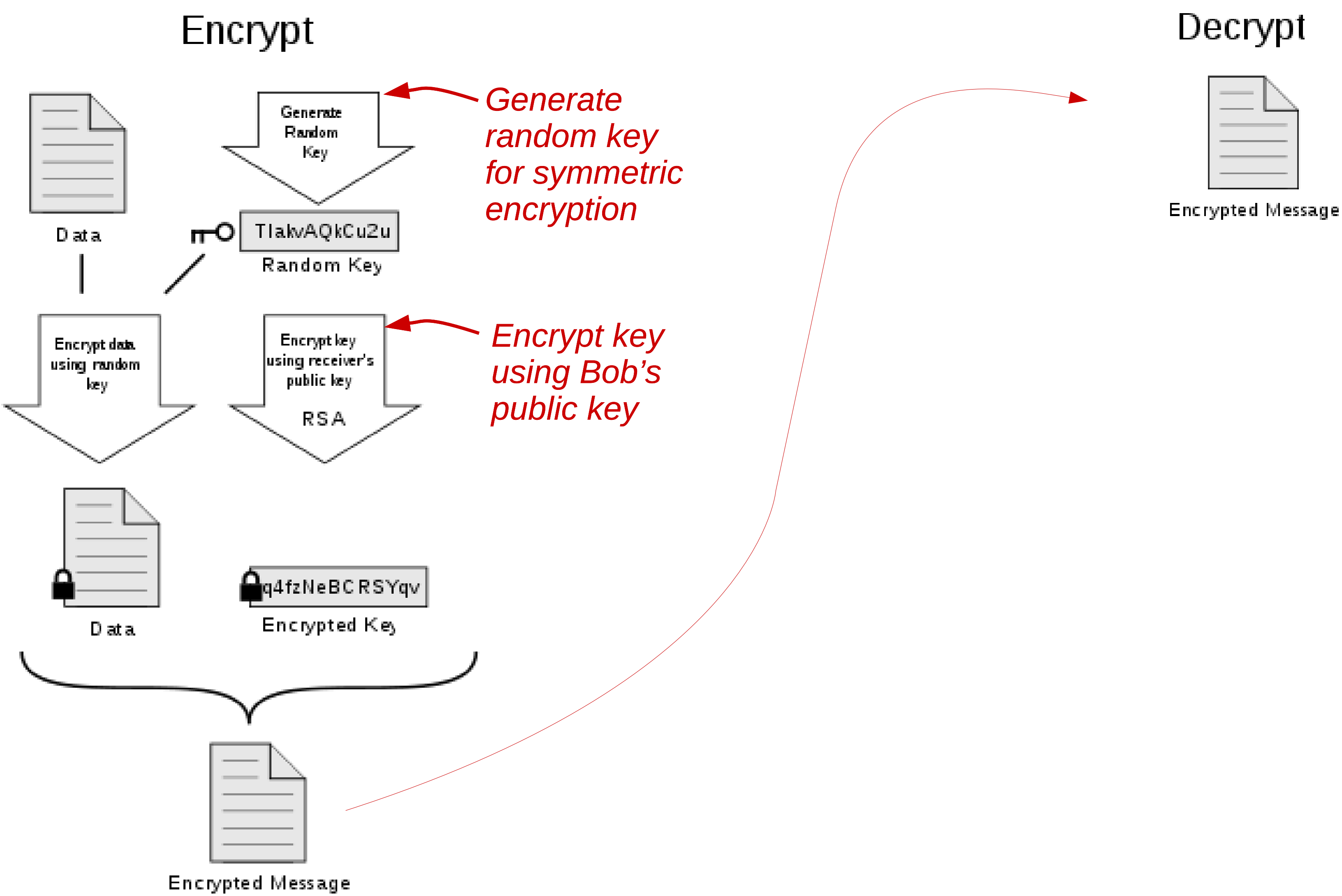
A key for symmetric encryption



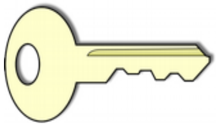
Encryption of transmitted data (PGP)

Alice encrypts

Exercise: How does Bob decrypt?



Agenda



1. Technology: Access control with passwords and hashing

2. Law: EU’s General Data Protection Regulation (including psedonymization)

3. Technology: Access control with digital signatures & digital certificates

4. Usable Security (Why Johnny Can’t Encrypt)

Identification, authentication

Identification

- the user's *assertion* of who he or she is (eg., a username)

Authentication

- *proving* the identity of the user

Authentication by passwords

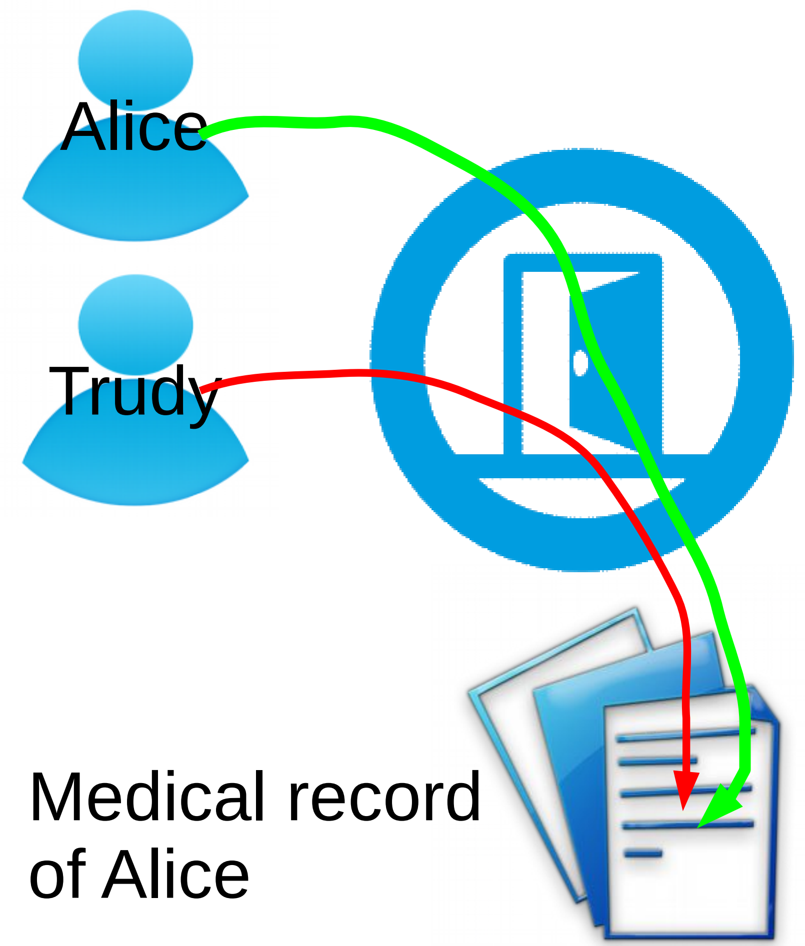
- “one factor-authentication”
- something you know (remember)
- password is <u>provided by user</u> and <u>compared to a stored version</u>

Design of access control

Attack #1: Password guessing
- password policy: strong passwords
- *dilemma: user may forget password*

Attack #2: Password phishing
- password policy: don't reveal password
- *not to anybody*

Attack #3: Password file stealing
- all passwords stored in table
  - compare at login
- *dilemma: passwords not accessible*
- solutions
  - hashing of passwords
  - salting of passwords



Attack #1: Password quessing

Dictionary of common passwords
1234
password
logmein
pizza
..

Simple dictionary attack:
- try alice with all passwords in dictionary
- try bob with all passwords in dictionary
- ..

To protect against dictionary attacks,
passwords must have multiple types of characters
- letters abc..
- digits 012..
- UPPER CASE and lower case
- other characters !”#€%&

Celebgate (2014)

- Almost 500 fotos of celebritites
- Kate Uptown, Kirsten Dunst, ..
- Published and sold on Reddit and other websites
- Obtained from Apple’s iCloud service

Convictions

- Emilio Herrera, Ryan Collins, Edward Majerczyk, George Garofano: 9-18 months

Attack #2: password phishing

- emails, pretending to be from iCloud, asking for account information

LinkedIn password leak (2012)

- 6,5 million users had their account data stolen, including passwords
- possibly more users and also their mails

Reactions

- LinkedIn deactivated passwords
- passwords in file were hashed, but not salted
- LinkedIn recieved criticism and lawsuits

Arrest

- Yevgeny Nikulin arrested in Prague in 2016 and extradited to the USA

Alleged method: attack #3 - stealing password file

- obtained password of LinkedIn employee

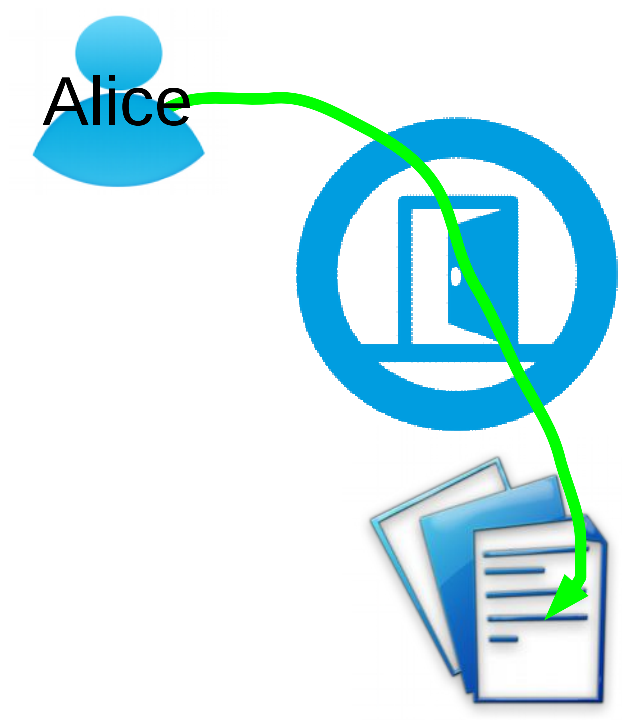
Password file

Password file (naive)

User	Password
Alice	a123456A
Bob	b456890B

Used to verify a user's password at login

Alice types password "a123456A" at login



Server compares text with password in file

Password file

Password file (naive)

User	Password
Alice	a123456A
Bob	b456890B

Used to verify a user's password at login

Passwords in password files must be protected
- a system administrator may not be trustworthy
- or the password file may be stolen

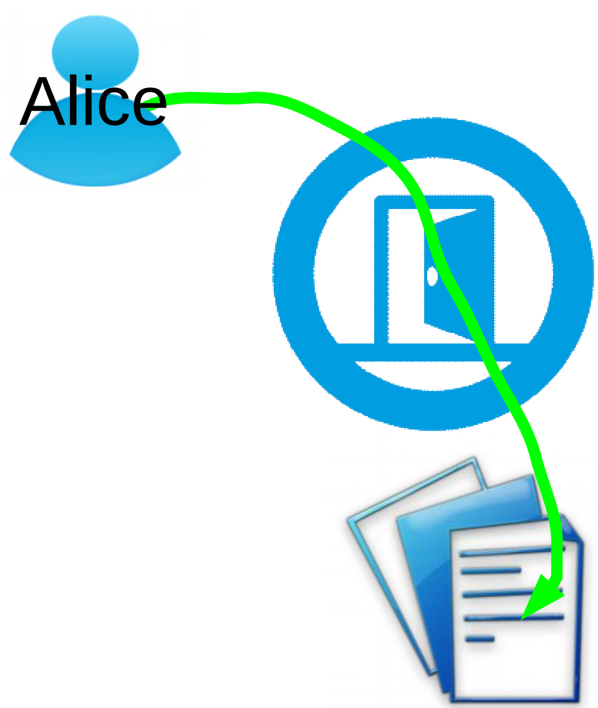
LinkedIn password file was protected by hashing

Password file (naive)

User	Password
Alice	a123456A
Bob	b456890B

Password file with hashed passwords

User	Hashed passwd.
Alice	3X€!BXY7
Bob	Y4KUI??X



Alice types password “a123456A” at login

Server computes hash(a123456A) and compares to value in password file

so password is never stored

LinkedIn password file was protected by hashing

Password file (naive)

User	Password
Alice	a123456A
Bob	b456890B

Password file with hashed passwords

User	Hashed passwd.
Alice	3X€!BXY7
Bob	Y4KUI??X

Hash function:

- a123456A -> 3X€!BXY7

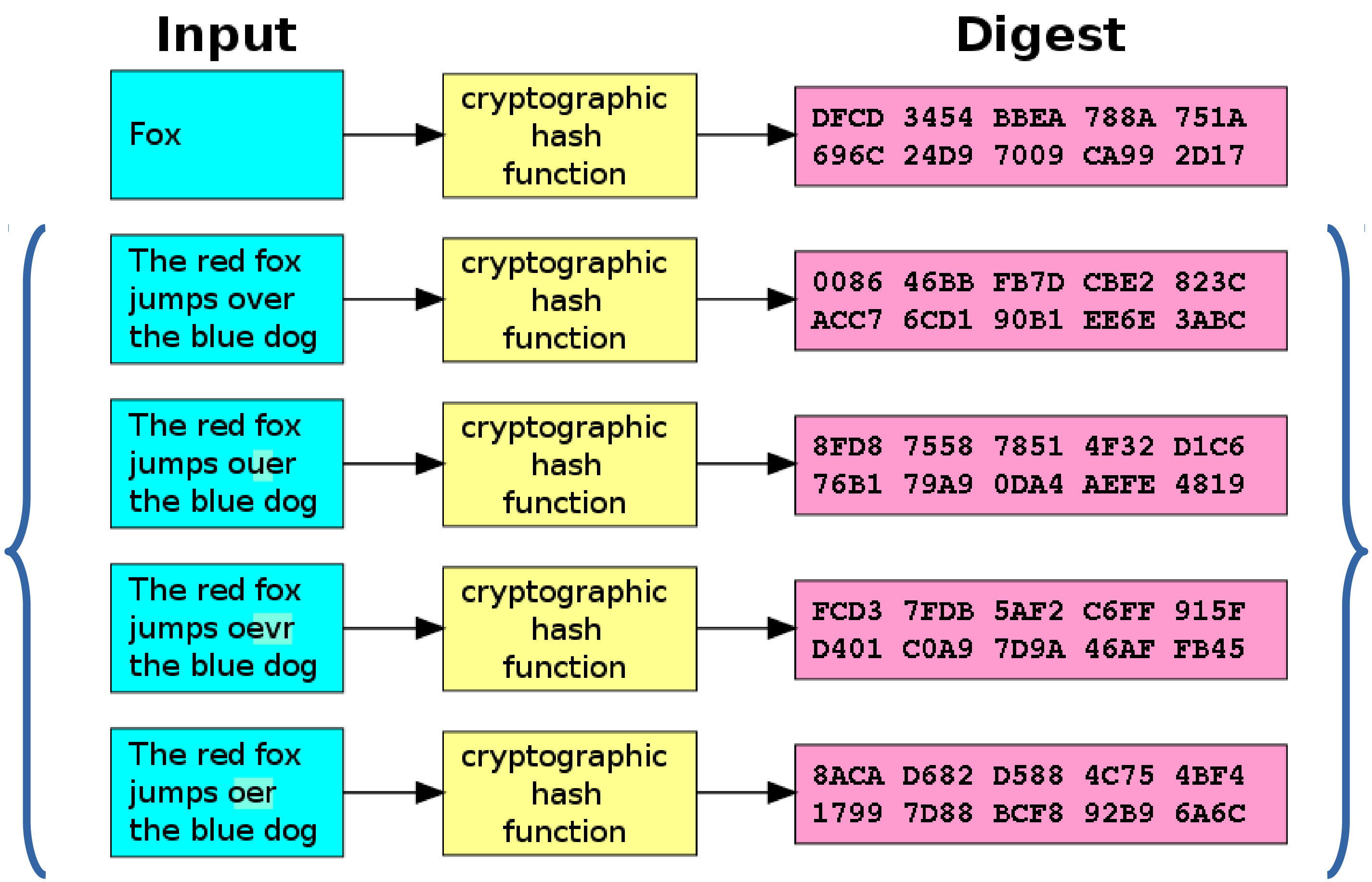
Requirements for cryptographic hash function

- "Practically impossible" to infer passwords from hashes
  - no method significantly better than brute force
  - so resembles requirements for encryption
  - except we brute force "all passwords", instead of "all keys"

“Avalance” effect of cryptographic hash funtions

.. implies that an attack does not know whether a guessed password is closed to the true password

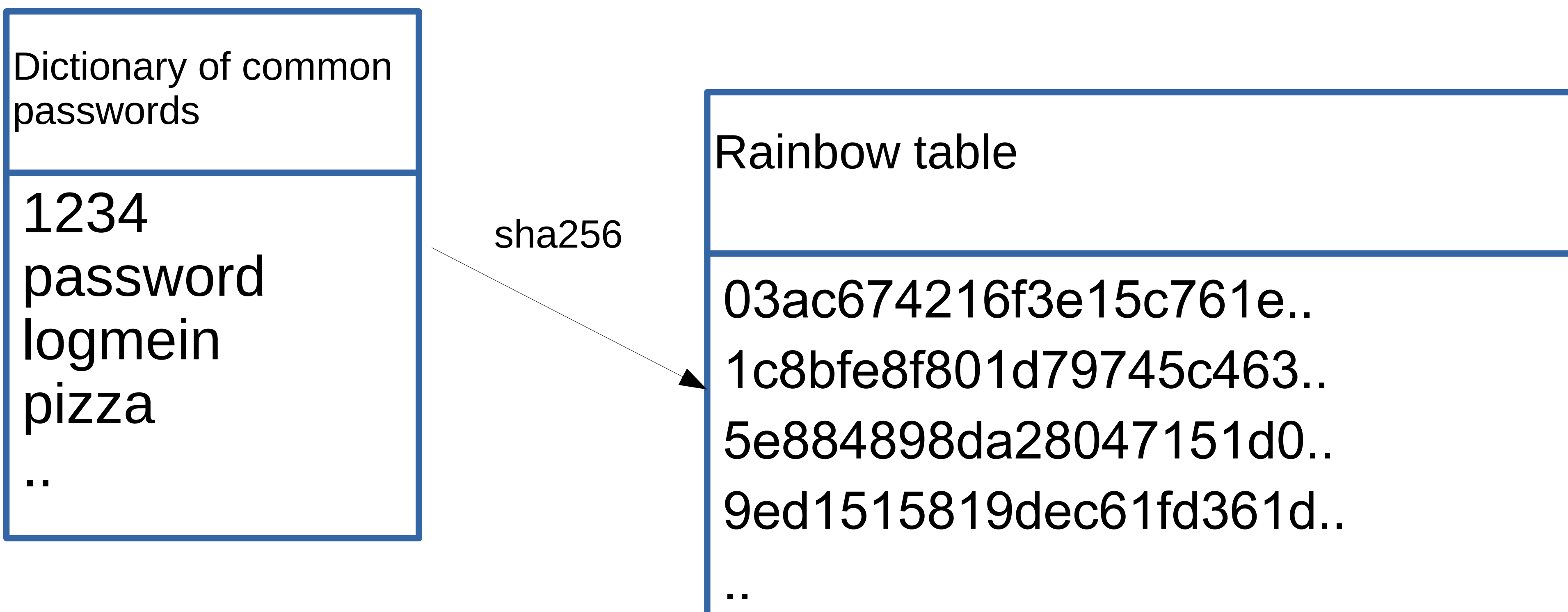
Small change password



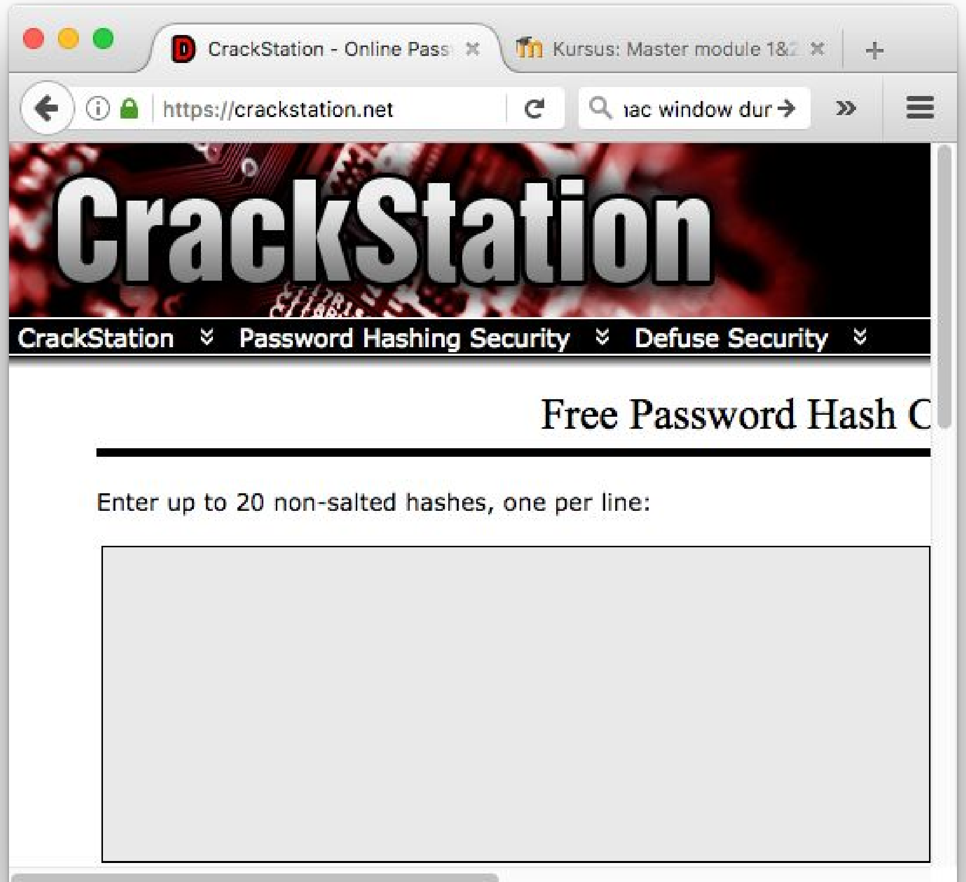
Big change in hash value

Attack #3: Attack applies to password files with encrypted passwords!

Rainbox table
- precomputed, may take months
- cracks naive passwords
- maybe 1 million out of 6,5 million LinkedIn passwords



Rainbow tables are online



crackstation.net

- pre-computed hash values
- hash algorithms: SHA-1, SHA-256,..
- of many, many passwords

Salt: protection against rainbow tables

Password file with salts
• a salt is added to the password
(data in parenthesis is not in file)

different hash value even though password is the same

user	(passw.)	salt	hashed passw+salt
alice	(a12345A)	ab	2d5d3e44
bob	(b67890B)	2d	2d46e346
peter	(a12345A)	3f	e30f4d27

Now a precomputed password table is useless.
Attacker must now
(1) guess password
(2) hash password + salt

Note: salt can be read from password file

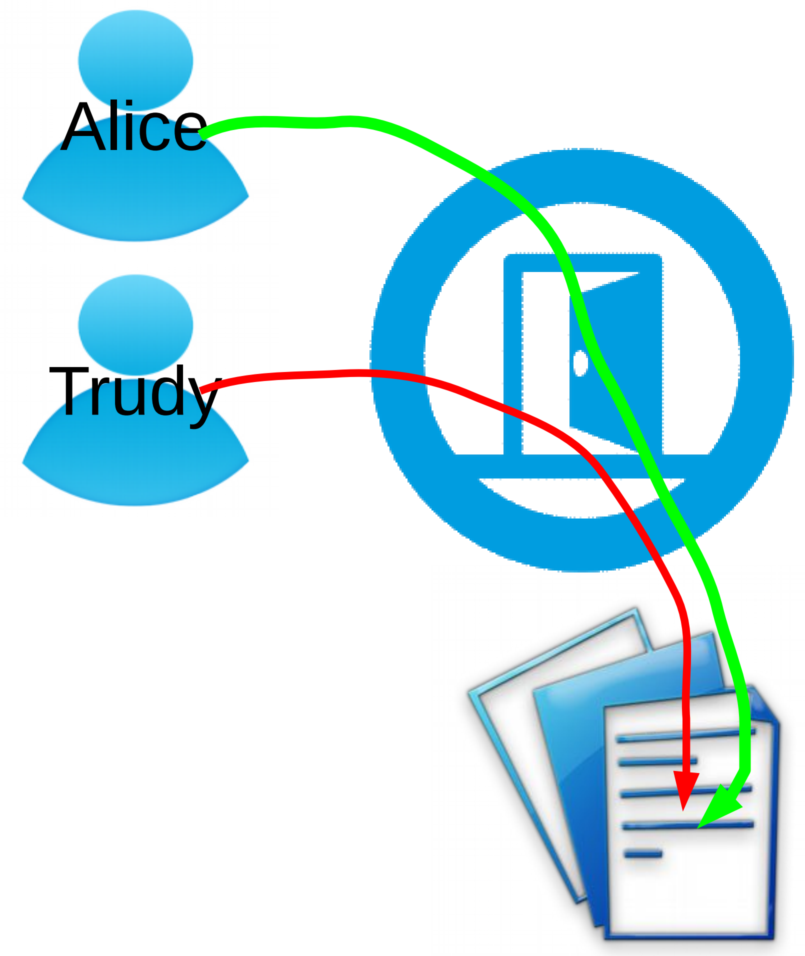
Access control - conclusion

Attack #1: Password guessing
- password policy: strong passwords
- *dilemma: user may forget password*

Attack #2: Password phishing
- password policy: don't reveal password
- *not to anybody*

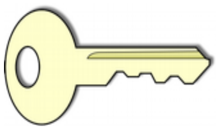
Attack #3: Password file stealing
- all passwords stored in table
  - compare at login
- *dilemma: passwords not accessible*
- solutions
  - hashing of passwords
  - salting of passwords

However, if password file is stolen
- any particular password can be brute-forced
- number of passwords < number of DES keys
- additional security is required
  - digital signatures
  - two-factor authentication



Agenda

1. Technology: Access control with passwords and hashing

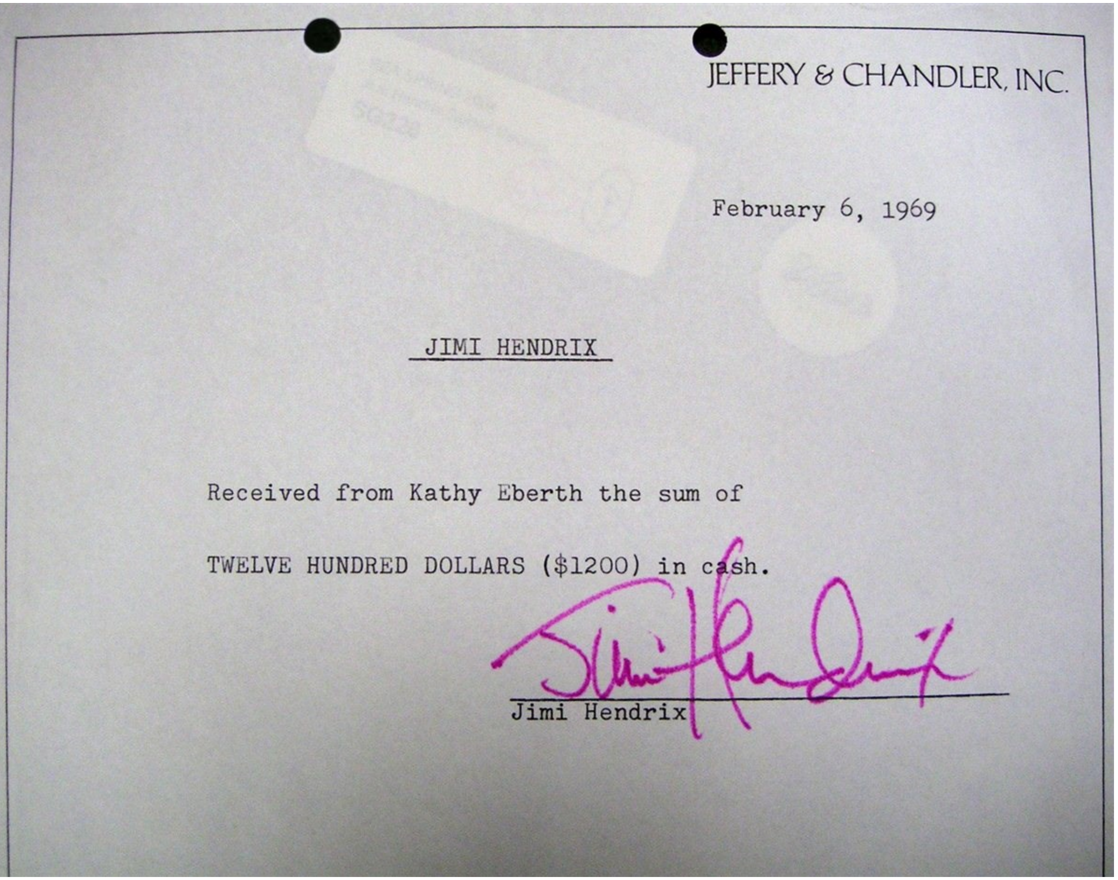


2. Technology: Access control with digital signatures & digital certificates

3. Law: EU's General Data Protection Regulation (including psedonymization)

4. Usable Security (Why Johnny Can't Encrypt)

Digital signatures resemble paper-based signatures



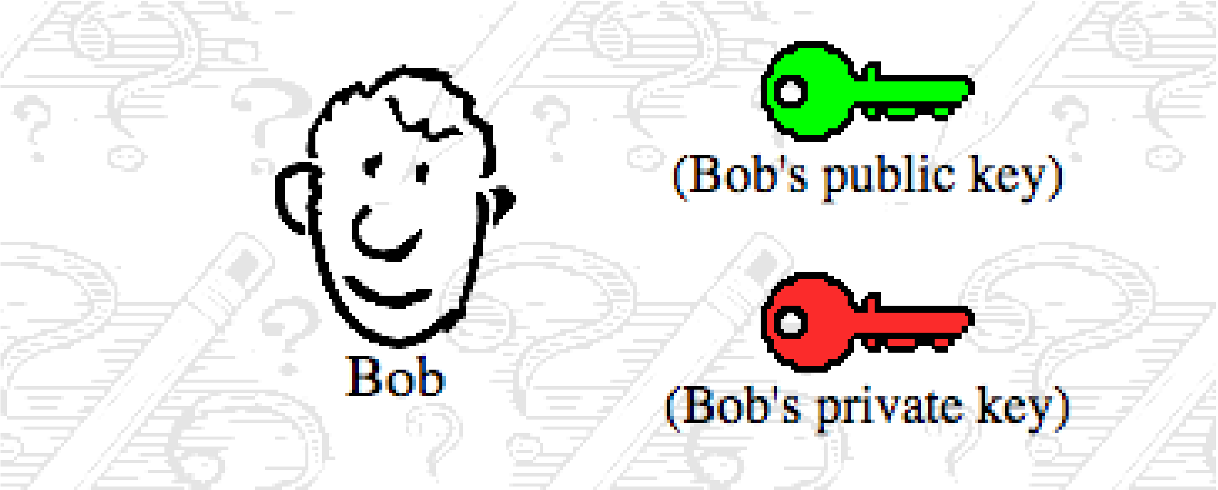
Properties of digital and physical signatures

- originality of document is protected
- autenticity of signer is protected
- links document and signer

Digital signatures

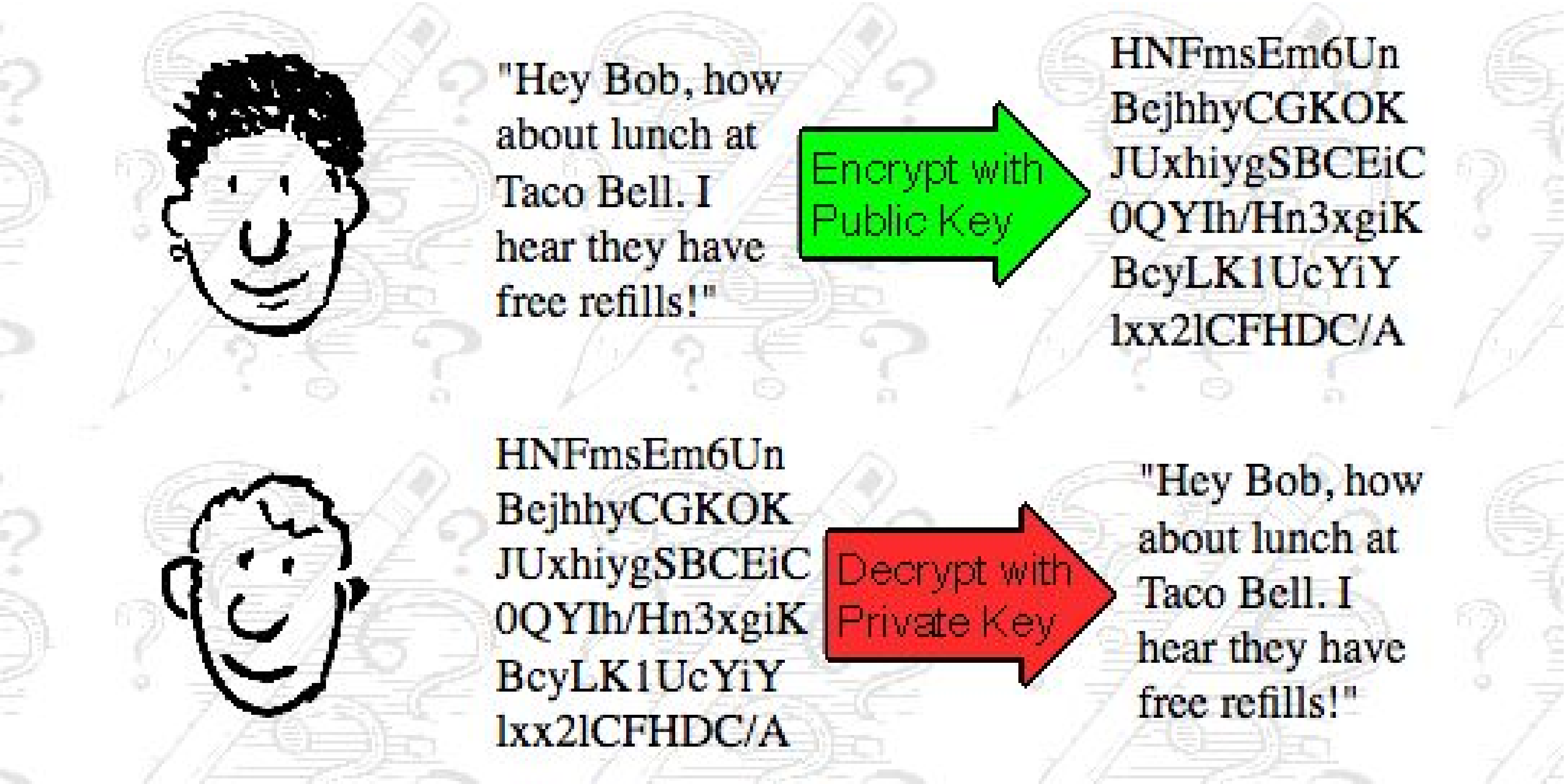
- created using signer’s private key
- confirmed using signer’s public key

Public key pair



(Thanks to David Youd for images and a very pedagogical explanation)

Encryption, transmission, decryption



(Please note: in practice, asymmetric encryption/decryption is not for large messages, because asymmetric encryption/decryption is time-consuming).

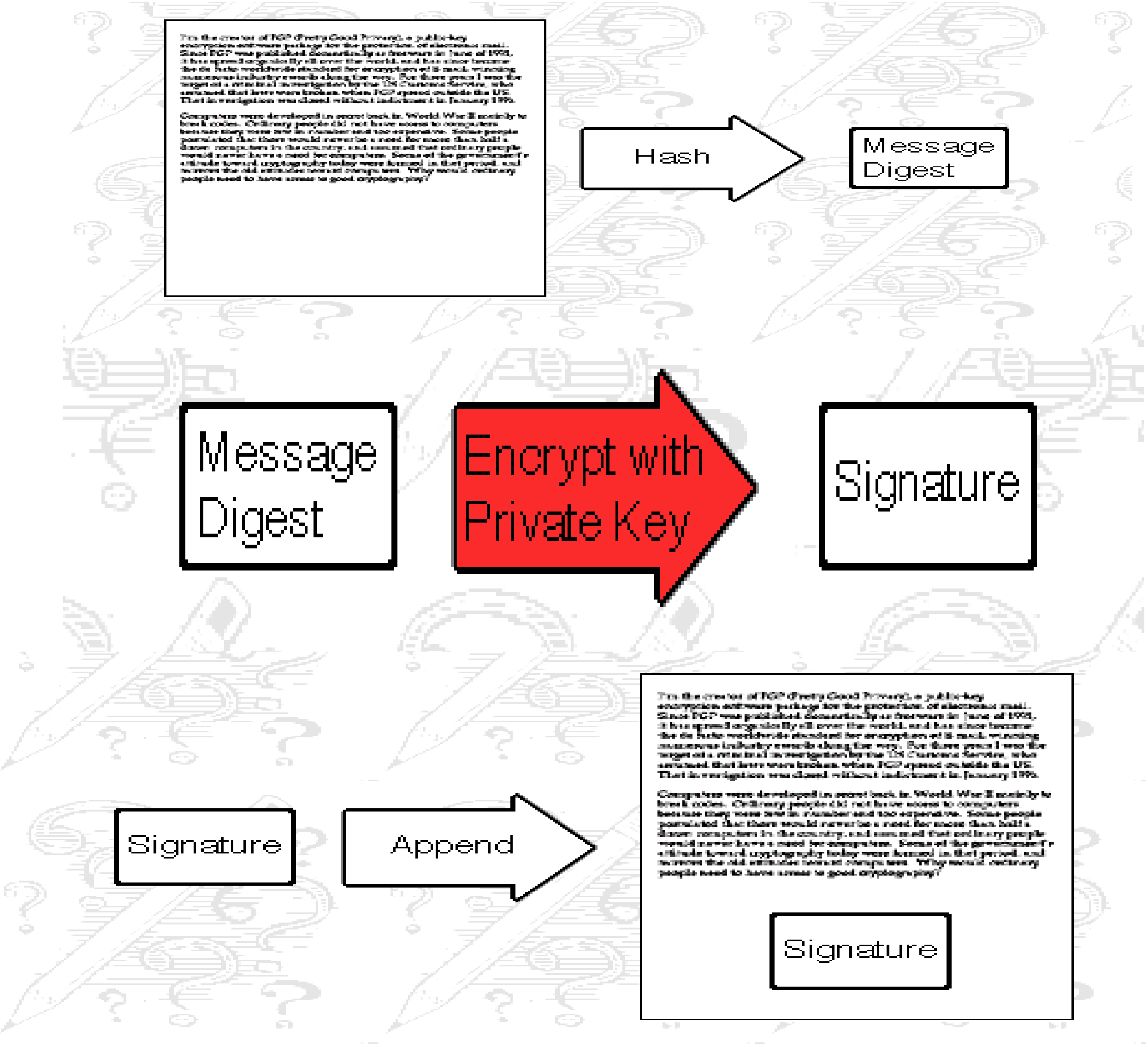
Digital signature produced by sender

Bob (signer, sender)

1. Compute message digest

2. Compute signature with private key

3. Append signature to documet, then send.



Cryptographic hash functions

A hash value is a “fingerprint” of a text

- small
- *unique for the text*

Full set of requirements for cryptographic hash functions

Easy

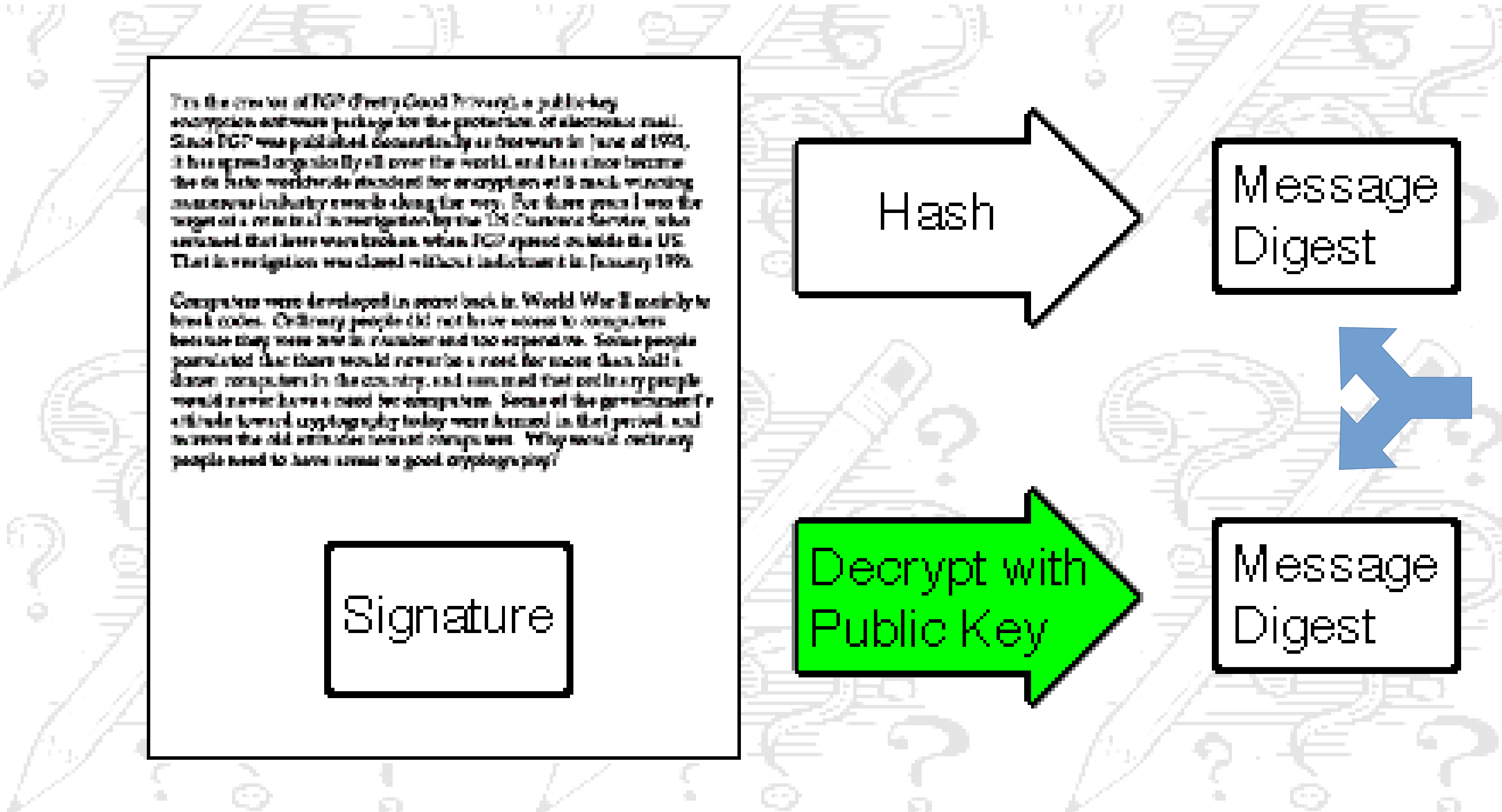
- variable input length
- fixed output length (ie., 128 or 256 bit)
- reasonable speed (about the same as symm. encryption)

Difficult

- pre-image resistance (not find passwd from passwd hash)
- second pre-image resistance
- collision resistance

Receiver’s verification of signature

Pat (receiver)



Signature verified if message digests are the same

Receiver decrypts with sender’s public key

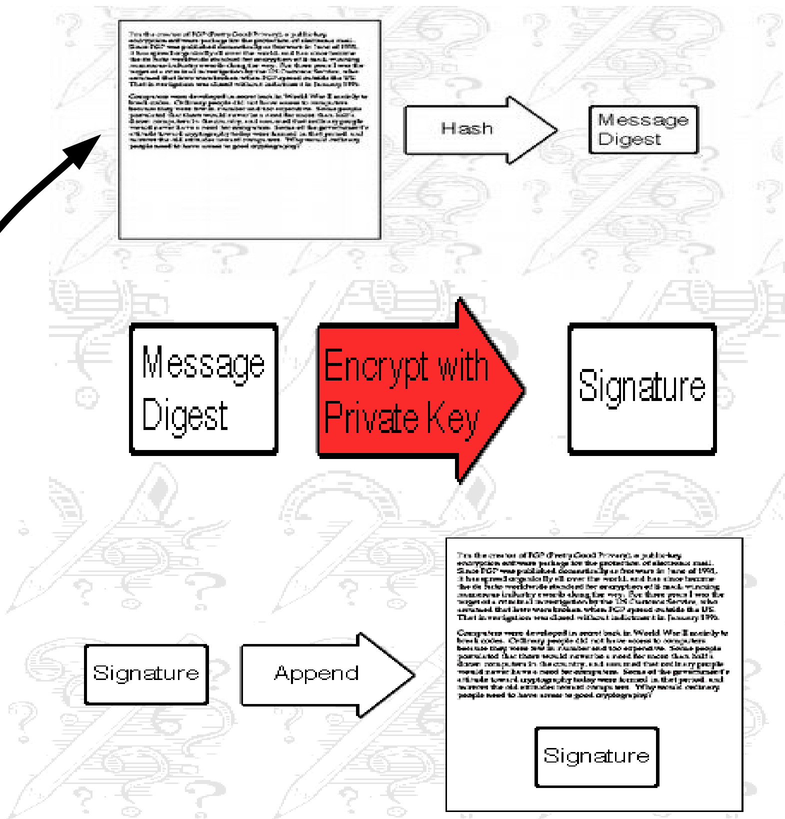
- Proved that message was signed with sender’s private key
- Because no other key permits decryption with this public key

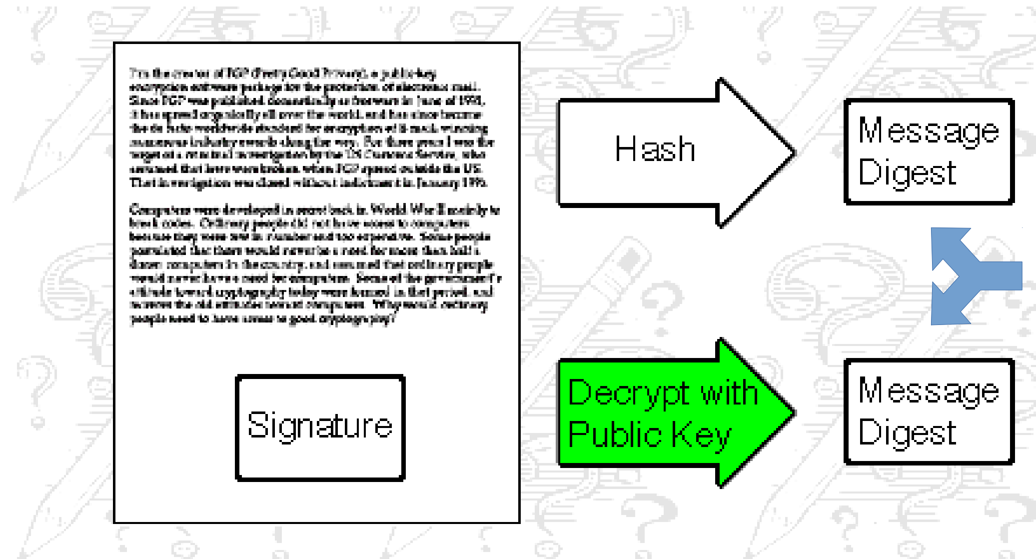
Exercise:

~~Bob (signer, sender)~~

Trudy is changing document..

Pat (receiver)



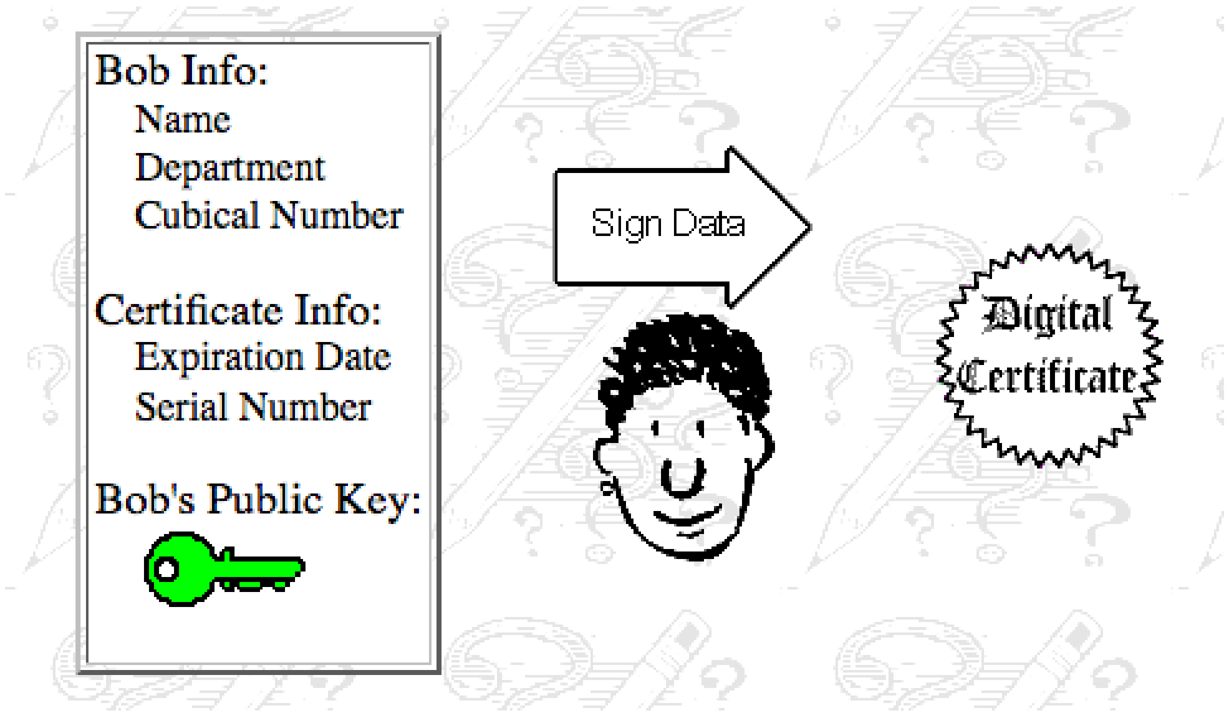


Suppose Trudy intercepts the communication

- *changes the document*
- computes a new message digest
- produce a signature, using Trudy's own private key

How would Trudy's changes be revealed by the receiver?

Digital certificate



Purpose:

- to guarantee “this public key belongs to Bob”

Mechanism

- certificate is itself a digitally signed document
- signed with a trusted person’s private key
- certificate mainly contains: ID (“Bob”) + public key + signature

NemID

NemID is a PKI-based system but with central storage of private keys.

Advantages

- Much easier for the user
  - no installation of a signature file
- More secure for un-protected users (the majority)
  - the signature file can't be stolen from the user's computer

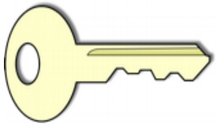
Disadvantages

- The user must trust Nets
  - disgruntled employees at Nets could steel the private key
    (if they could circumvent security at Nets)
- Danish government could demand access to the private keys
- A foreign government could demand access
  - Nets was sold in 2014 to a consortium of Danish and foreign companies
    (17 billion DKK)

Agenda

1. Technology: Access control with passwords and hashing

2. Technology: Access control with digital signatures & digital certificates



3. Law: EU’s General Data Protection Regulation (including psedonymization)

4. Usable Security (Why Johnny Can’t Encrypt)

EU’s General Data Protection Regulation (GDPR)

GDPR timeline

- passed April, 2016
- applies in all EU member states from May, 2018

GDPR contents

- focus on privacy, ie., of health-related data
- requirements, fines
- companies/organizations must have a data protection policy
- but requirements are general/vague

Denmark, before GDPR

- “Persondataloven”
- Compliance with ISO 27.000 (a security standard)
  - applies to all government institutions + suppliers

ISO 27.000 in Denmark

The Office of the Auditor General (Rigsrevisionen)

2015 analysis of MedCom
- heath care network (medcom.dk)
- transmits health care data between hospitals etc.

Criticism
- “a threat to the lives and health of patients”
- lack of a contingency plan (“beredskabsplan”)
  - what to do in case of non-availability of network services
- lack of guidelines about access control
  - data accessible to administrators at suppliers

Previous reports have criticized..
- .. the police, tax, defence, statistics and many other institutions

GDPR “considerations”

Considerations 1-173
- motivation, background (“bemærkninger”)
- focus: privacy; “assessing”; “appropriate”

Consideration 83:

[..] evaluate the risks inherent in the processing and implement measures to mitigate those risks, such as encryption. Those measures should ensure an appropriate level of security, including confidentiality, taking into account the state of the art and the costs of implementation in relation to the risks and the nature of the personal data to be protected. In assessing data security risk, consideration should be given to the risks that are presented by personal data processing, such as [..] unauthorised disclosure of, or access to, personal data transmitted, stored or otherwise processed [..]
(My emphasis)

GDPR articles

Articles 1-99

- the actual regulation
- applies in EU member states similarly as a national law

Article 5: Principles relating to processing of personal data (1,5 pages)

(a) lawfulness, fairness and transparency
(b) purpose limitation
(c) data minimisation
(d) accuracy
(e) storage limitation: kept in a form which permits identification of data subjects for no longer than is necessary
(f) integrity and confidentiality

Accountability: demonstrate that system complies with this article.

GDPR (continued)

Article 32: Security of processing

1 Taking into account the state of the art, the costs of implementation and [..] purposes of processing as well as the risk [..] appropriate [..] measures

(a) the pseudonymisation and encryption of personal data;
(b) [..] confidentiality, integrity, availability and resilience [..]
(c) [..] restore [..] in the event of a physical or technical incident;
(d) [..] regularly testing, assessing and evaluating the effectiveness of [..] measures for ensuring the security of the processing.

3. Adherence to an approved code of conduct [..] may be used as an element by which to demonstrate compliance with the requirements set out in paragraph 1 of this Article.

GDPR (continued)

Article 6: Principles relating to processing of personal data

Processing requires
(a) consensus, or
(b)-(e) necessity, for example “necessary for the performance of a task carried out in the public interest”

Otherwise (??), take into account
..
(c) the nature of the personal data
(d) the possible consequences of the intended further processing for data subjects;
(e) the existence of appropriate safeguards, which may include encryption or pseudonymisation.

Pseudonymi<u>s</u>ation (UK) = pseudonymi<u>z</u>ation (US)

Pseudonymization

Grading table

Student ID	Grade
Alice + cpr#	7
Bob + cpr#	10

Grading table with pseudonyms

Student study#	Grade
309235	7
763915	10

Pseudonym table

Student ID	Student study#
Alice + cpr#	309235
Bob + cpr#	763915

Use these pseudonyms when

- transmitting btw. adm/teacher
- publishing grades?
- in storage?

Pseudonym

- not phone#
- because one can look-up a phone#

Psydonymization ≠ anonymization

- if students were anonymous, the data is useless

Exercise (pseudonymization)

In your project, consider again the most sensitive data, yet not data that is so sensitive as to require encryption.

Suggest a way that psedonymization can increase the protection of this data.

If pseudonymization is not relevant, why?

Agenda

1. Technology: Access control with passwords and hashing

2. Technology: Access control with digital signatures & digital certificates

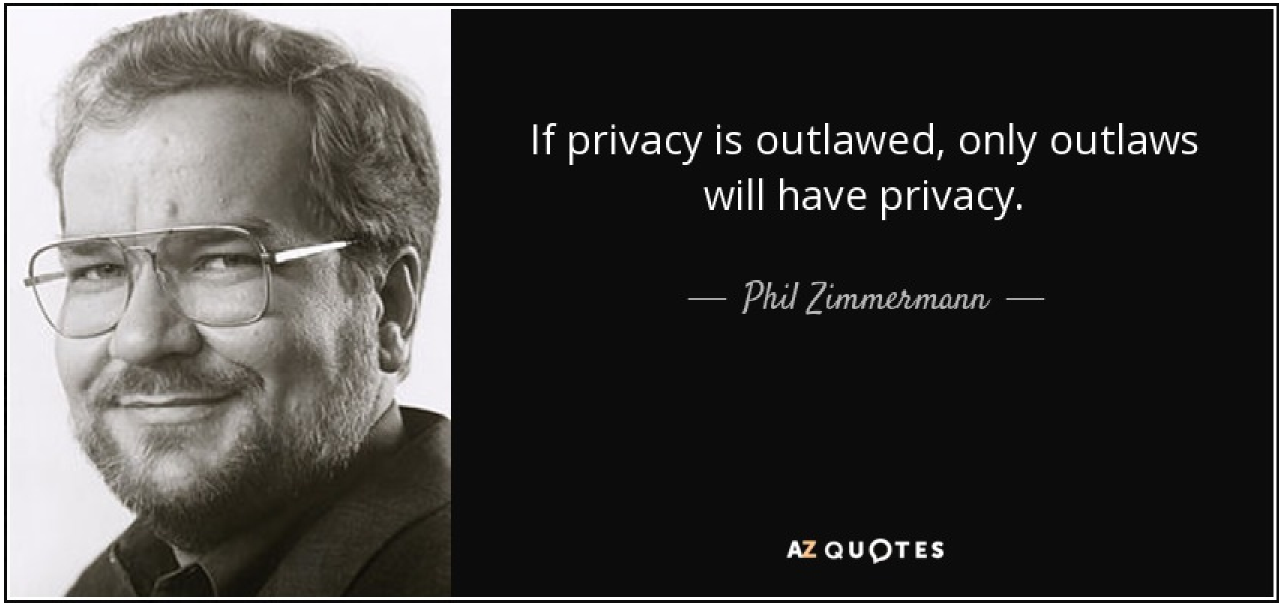
3. Law: EU’s General Data Protection Regulation (including psedonymization)

4. Usable Security (Why Johnny Can’t Encrypt)

Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0

PGP
= Pretty Good Privacy

A tool for encryption and signing mails, files, and other entities



Developed 1991+
- by Phil Zimmermann
- to promote privacy
- in opposition to US encryption regulation
- sold for 37 mill. USD in 1997

PGP software is used in Enigmail
- an extention of Mozilla Thunderbird

Why Johnny Can’t Encrypt: A Usability Evaluation of PGP 5.0

“PGP 5.0 is not usable enough to provide effective security for most computer uses” (p1)

PGP 5.0’s user interface does not come even reasonably close to achieving our usability standard - it does not make public key encryption of electronic mail manageable for average computer users” (p2)

Usable security

The general concept of usability, applied to security

> *“Definition: Security software is usable if the people who are expected to use it:*
>
> *1. are reliably made aware of the security tasks they need to perform;*
> *2. are able to figure out how to successfully perform those tasks;*
> *3. don’t make dangerous errors; and*
> *4. are sufficiently comfortable with the interface to continue using it.”*

(Whitten & Tygar p2)

Usability

Usability was a dominant paradigm in the 1990s-2000s

Compare to..

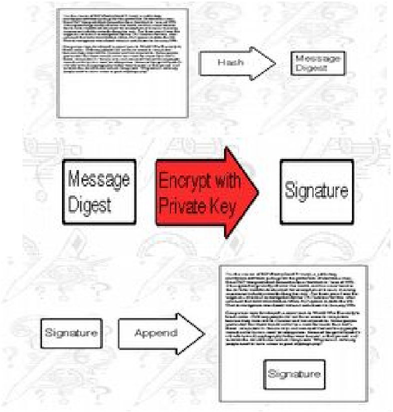
.. user-friendliness
- usability focuses more on the user’s task (getting the job done)

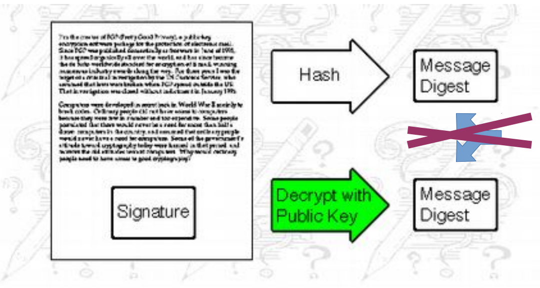
.. user experience
- usability focuses less on the user’s emotions

Challenges related to usable security

1. The unmotivated user property
2. The abstraction property
3. The lack of feedback property
4. The barn door property
5. The weakest link property

(Whitten & Tygar p3)





Whitten & Tygar report from two tests

Test #1: Cognitive walkthrough
- W. & T. reviewed the user interface themselves
- imagined they were novice users
- understand what to do?

Test #2: User test
- asked 12 test persons to use PGP
- test persons should image they were campaign coordinators
- sending an itinerary (schedule) using signing and encryption

Exercise

Summarize Whitten & Tygar’s critique of the visual metaphors
(pat of the cognitive walkthrough)

Exam questions (examples)

Your project

- what data is sensitive (in particular personal data)
  - why / not ?
- what data is encrypted? pseudonymized?
  - why / not ?
- how are you protecting passwords? (if any)
- other measures than encryption / pseudonymization?
  - such as delete data when no longer needed
- usable security
  - definition, goals: are users prevented from *making dangerous errors*?
  - challenges: is the *unmotivated user property* relevant?

Security technology in general

- explain basic properties of
  - symmetric encryption, asymmetric encryption
  - hashing, digital signatures, digital certificates
- algorithms such as AES are considered secure
  - in what sense are they secure?