UNIVERSITÄT
ZU KÖLN

Computersicherheit
(Kryptographie, Authentifizierung, Vertrauen)

Nils Reiter,
nils.reiter@uni-koeln.de

October 31, 2023

INSTITUT FÜR
DIGITAL HUMANITIES
UNIVERSITÄT ZU KÖLN

- Was tust Du für Deine Computersicherheit?
- Was solltest Du tun? Warum tust Du es nicht?
- Welche Aspekte findest Du außerdem relevant?

bequem ↔ sicher

Computersicherheit

- Passwörter
- Verschlüsselung (Dateien, E-Mails, ...)
- Sicherheitslücken
- Datensicherheit
- Viren/Würmer ...
- Verhalten
- ...

Introduction

- Not everyone can do/has access to everything
  - E.g., only people employed by a university can grade students, and only specific people can see every grade by a specific student
  - E.g., only police has access to criminal records (and only in some situations)
  - E.g., only fire department can enter and check for emergency exits
- Default case in pre-digital society: Limited access
- Restricted access needs to be modeled digitally as well

Introduction

- Not everyone can do/has access to everything
  - E.g., only people employed by a university can grade students, and only specific people can see every grade by a specific student
  - E.g., only police has access to criminal records (and only in some situations)
  - E.g., only fire department can enter and check for emergency exits
- Default case in pre-digital society: Limited access
- Restricted access needs to be modeled digitally as well
- Multiple aspects
  - Identity and access rights
  - Security of data
  - Trust into systems – verifiable behaviour of systems

Security and Identity

- Closely related technically and conceptually
- If you supply your password, you need to make sure it's to a legitimate party
  - Otherwise, someone could steal it
- No built-in identity verification in many systems (e.g., e-mail)

"On the Internet, nobody knows you're a dog."

Peter Steiner, *The New Yorker*, 1993

Basics

- All content can be expressed numerically
  - E.g., the word “dog” as a $\langle 100, 111, 103 \rangle$ in ASCII
- To make things simpler, we only use numbers from now on
- Common example: Alice and Bob want to encrypt stuff, Mallory/Eve are malicious attackers

Wikipedia on Alice and Bob

Encryption

- Goal: Transform a message m, such that only Bob with key k can read it

Encryption

- Goal: Transform a message m, such that only Bob with key k can read it
- Two main variants
  - Symmetric encryption
  - Asymmetric encryption

Encryption

Symmetric Encryption

- What we do in PDFs, zip-files, ...
- Easiest idea: Key is some number n that we add to each letter to derive at cypher text c
  - $\mathrm{enc}(m, n) = m + n$
    - E.g.: $\mathrm{enc}(\langle 100, 111, 103\rangle, 20) = \langle 120, 131, 123\rangle$
  - $\mathrm{dec}(c, n) = c - n$
    - In this case it's obvious that $\mathrm{dec}(\mathrm{enc}(m, n), n) = m$

Encryption

Symmetric Encryption

- What we do in PDFs, zip-files, ...
- Easiest idea: Key is some number n that we add to each letter to derive at cypher text c
  - $\mathrm{enc}(m, n) = m + n$
    - E.g.: $\mathrm{enc}(\langle \overset{d}{\underline{100}}, 111, 103\rangle, 20) = \langle \overset{x}{\underline{120}}, 131, 123\rangle$
  - $\mathrm{dec}(c, n) = c - \textcircled{n}$
    - In this case it's obvious that $\mathrm{dec}(\mathrm{enc}(m, n), n) = m$
- k is a shared secret between Alice and Bob
- They need to agree on k beforehand
  - In a secure way!
- They need to store it and keep it safe
- Anyone with access to k can decrypt the cypher text

Encryption

Symmetric Encryption

- Variants
  - More complex calculations (e.g., multiplication instead of addition or complex equations)
    - But they need to be reversible!
  - Encryption of blocks of symbols
  - Generation of more complex keys (e.g., random numbers)
  - Algorithmically adapt n (Enigma)

Enigma simulator

Encryption

Symmetric Encryption

- Variants
  - More complex calculations (e.g., multiplication instead of addition or complex equations)
    - But they need to be reversible!
  - Encryption of blocks of symbols
  - Generation of more complex keys (e.g., random numbers)
  - Algorithmically adapt n (Enigma)

Enigma simulator

Practical Considerations

- All methods with a constant mapping of symbols are easy to break
- To regularly change or update n requires coordination
- Exchanging a secret is difficult in practice over the internet
- Does not scale well for an entire society

Encryption

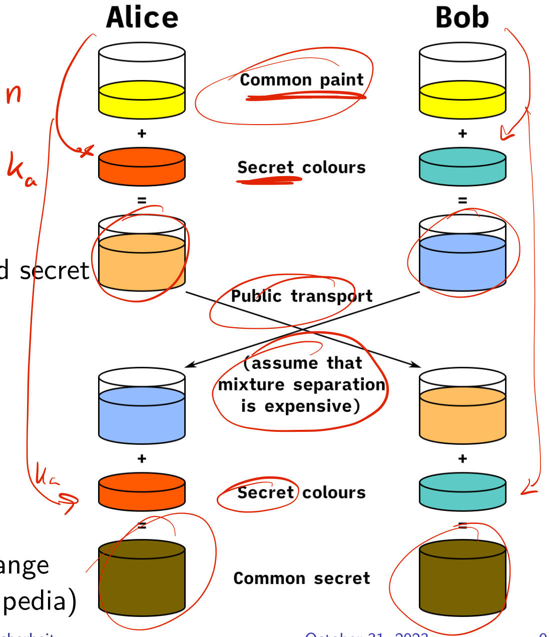
Asymmetric Encryption (to produce a shared secret)

- First described to securely generate a shared secret (Diffie/Hellman, 1976; Merkle, 1978)
- “Diffie-Hellman-Key-Exchange”

Encryption

Asymmetric Encryption (to produce a shared secret)

- First described to securely generate a shared secret (Diffie/Hellman, 1976; Merkle, 1978)
- “Diffie-Hellman-Key-Exchange”



Diffie-Hellman Key Exchange
(A.J. Han Vinck via Wikipedia)

Encryption

Asymmetric Encryption

- Core ingredient: Different keys for encryption k_e and decryption k_d
  - But generated as a pair, because they need to match
- Requirement: $\mathrm{dec}(\mathrm{enc}(m, k_e), k_d) = m$

Encryption

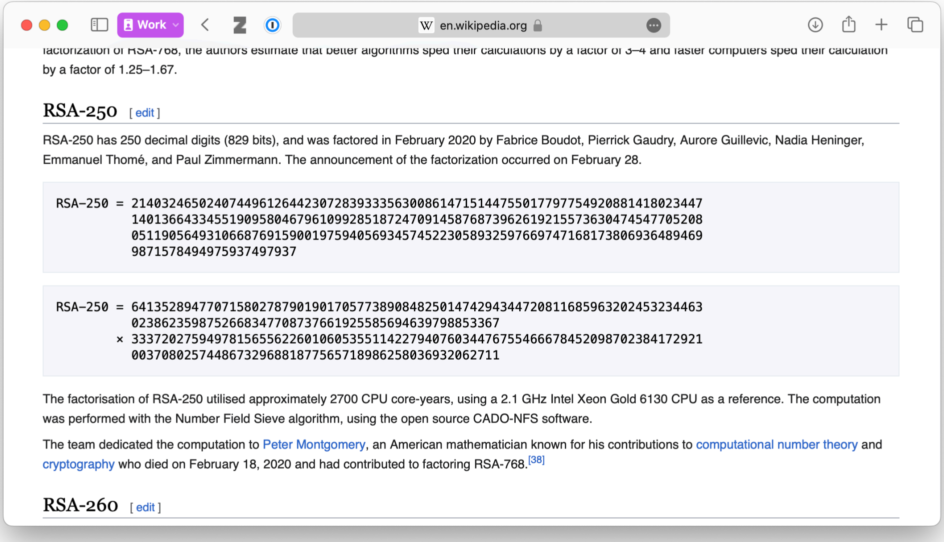
Asymmetric Encryption

- Core ingredient: Different keys for encryption k_e and decryption k_e
  - But generated as a pair, because they need to match
- Requirement: $\mathrm{dec}(\mathrm{enc}(m, k_e), k_d) = m$

One-Way-Functions

Functions that are easy to compute in one direction, but impossible to reverse

- Most well known: Prime factorisation
  - Let p and q be two prime numbers
  - Calculating $p * q = i$ is straightforward and fast
  - But there is no efficient way of getting p and q from i, except iterating over all primes
  - If p and q are large enough, this takes until the end of the universe



factorization of RSA-768, the authors estimate that better algorithms sped their calculations by a factor of 3–4 and faster computers sped their calculation by a factor of 1.25–1.67.

RSA-250 [edit]

RSA-250 has 250 decimal digits (829 bits), and was factored in February 2020 by Fabrice Boudot, Pierrick Gaudry, Aurore Guillevic, Nadia Heninger, Emmanuel Thomé, and Paul Zimmermann. The announcement of the factorization occurred on February 28.

```
RSA-250 = 2140324650240744961264423072839333563008614715144755017797754920881418023447
          1401366433455190958046796109928518724709145876873962619215573630474547705208
          0511905649310668769159001975940569345745223058932597669747168173806936489469
          9871578494975937497937
```

```
RSA-250 = 6413528947707158027879019017057738908482501474294344720811685963202453234463
          0238623598752668347708737661925585694639798853367
        × 3337202759497815655622601060535511422794076034476755466678452098702384172921
          0037080257448673296881877565718986258036932062711
```

The factorisation of RSA-250 utilised approximately 2700 CPU core-years, using a 2.1 GHz Intel Xeon Gold 6130 CPU as a reference. The computation was performed with the Number Field Sieve algorithm, using the open source CADO-NFS software.

The team dedicated the computation to Peter Montgomery, an American mathematician known for his contributions to computational number theory and cryptography who died on February 18, 2020 and had contributed to factoring RSA-768.[38]

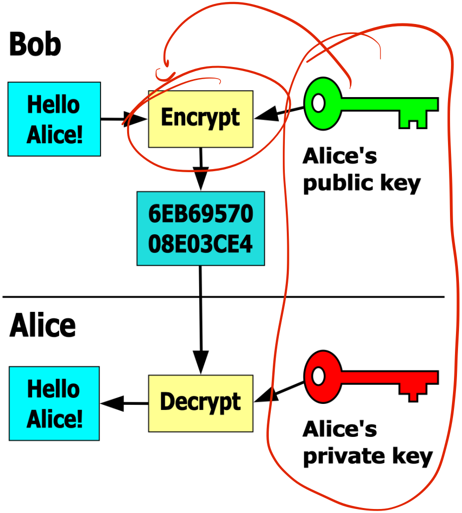
RSA-260 [edit]

Wikipedia

Encryption

Public Key Cryptography

- Core of most modern cryptography
- Public key to encrypt
- Private key to decrypt
- Keys generated as a matching pair
- Used for
  - E-Mails (S/MIME, PGP/GPG)
  - the Web (HTTPS/SSL)
  - SSH
  - ...



Encryption

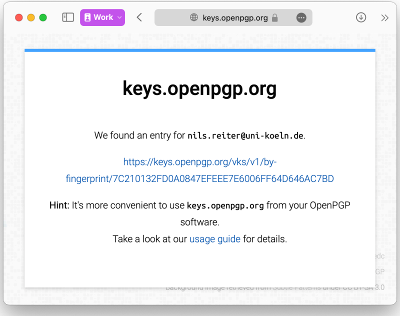
Public Key Cryptography

- Public Key Infrastructure (PKI)
- Maps public keys to ‘identities’
- For OpenPGP: `https://keys.openpgp.org`

Encryption

Public Key Cryptography

- Public Key Infrastructure (PKI)
- Maps public keys to ‘identities’
- For OpenPGP: `https://keys.openpgp.org`



Work
keys.openpgp.org
Background image retrieved from Subtle Patterns under CC BY-SA 3.0

keys.openpgp.org

We found an entry for `nils.reiter@uni-koeln.de`.

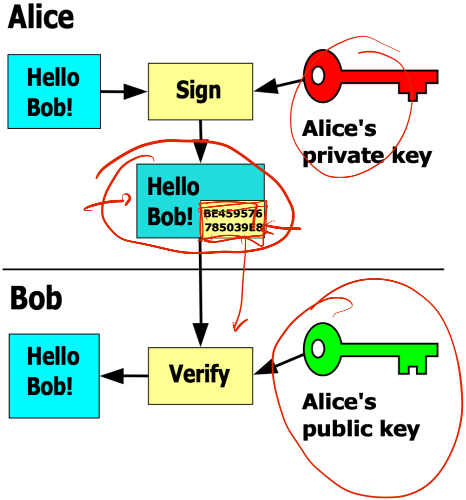
https://keys.openpgp.org/vks/v1/by-fingerprint/7C210132FD0A0847EFEEE7E6006FF64D646AC7BD

Hint: It's more convenient to use `keys.openpgp.org` from your OpenPGP software.

Take a look at our usage guide for details.

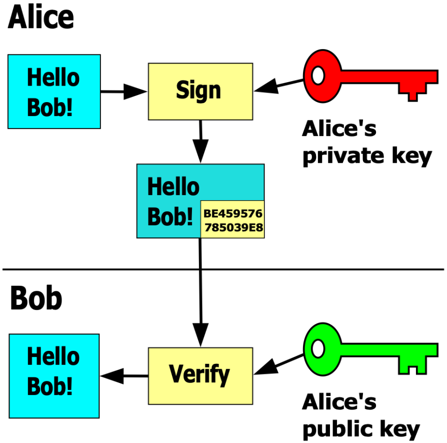
Signing with Public Key Cryptography

- Public/private keys can also be used for signing messages
- Only Alice is able to produce a message that can be reverted with her public key



Signing with Public Key Cryptography

- Public/private keys can also be used for signing messages
- Only Alice is able to produce a message that can be reverted with her public key
- Map identities to public keys: "public key infrastructure" (PKI)



Digital Covid Certificate (EU)

- EU provides central public key infrastructure to sign DCCs
- Member states have their own national systems
  - Public keys of member states are signed on EU level
- Member states can sign public keys of subsidiary elements
  - Theoretically all the way down to individual pharmacies

EU

↓

Deutschland

↓

NRW

↓

Bezirksregion Köln

↓

~~Apotheke am Markt~~

Digital Covid Certificate (EU)

- EU provides central public key infrastructure to sign DCCs
- Member states have their own national systems
  - Public keys of member states are signed on EU level
- Member states can sign public keys of subsidiary elements
  - Theoretically all the way down to individual pharmacies

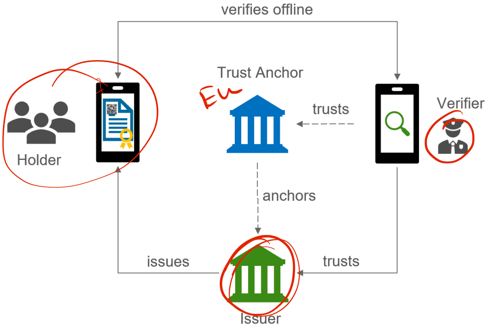
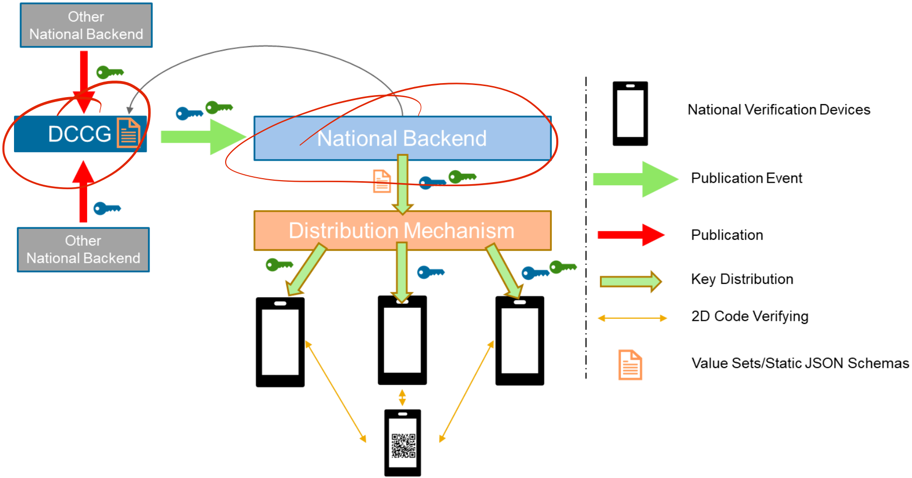


Figure: Trust Anchoring

Digital Covid Certificate (EU)

Distribution of Signing and Validation Information



Other National Backend

DCCG

National Backend

Distribution Mechanism

Other National Backend

National Verification Devices

Publication Event

Publication

Key Distribution

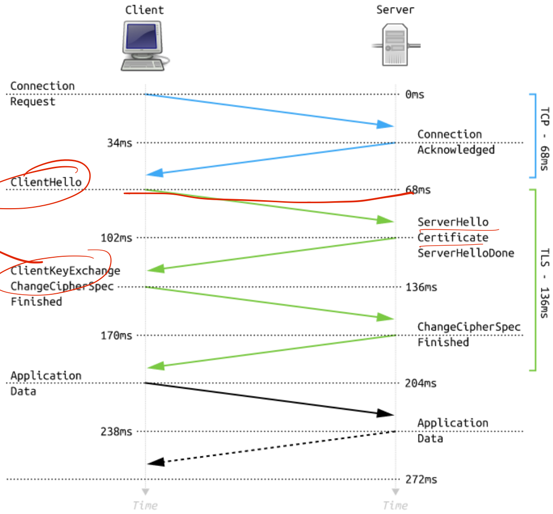
2D Code Verifying

Value Sets/Static JSON Schemas

Transport Layer Security (TLS)

CA

- ▶ Used whenever we access a https-webpage
- ▶ Requires a “certificate authority” (CA)



Client

Server

Connection Request

0ms

Connection Acknowledged

34ms

ClientHello

68ms

ServerHello
Certificate
ServerHelloDone

102ms

ClientKeyExchange
ChangeCipherSpec
Finished

136ms

ChangeCipherSpec
Finished

170ms

Application Data

204ms

Application Data

238ms

272ms

Time

Time

TCP - 68ms

TLS - 136ms

Authentication

- How does a system verify a humans identity?
- Single-factor vs. Multi-factor
- Three general ways – all need preparation!
  - Knowledge: Password, PIN, ...
  - Ownership: Hardware token, cell phone, id card, ...
  - Inherence: Biometric properties

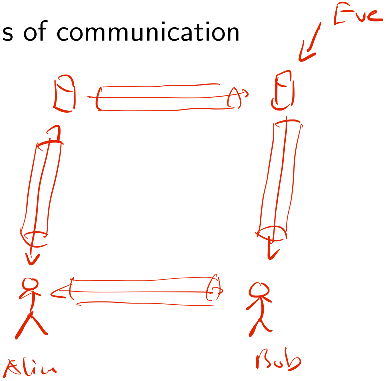
Online-Ausweis

- Personalausweise enthalten seit 2010 einen RFID-Chip
- Erlaubt maschinelles Auslesen der gespeicherten Daten
- Unterschiede für hoheitliche und nicht-hoheitliche Funktionen
- Bund signiert Stellen, die z.B. die eID verwenden wollen (PKI etc.)

personalausweisportal.de

End-To-End-Encryption

- Fully encrypted channel between both proper ends of communication
- Server is not necessarily the 'end'



End-To-End-Encryption

- Fully encrypted channel between both proper ends of communication
- Server is not necessarily the ‘end’
- Communication of people
  - Private devices of communication partners are ends
  - Not the mail server of one of them
- E2EE ensures that all parties in between the two only see encrypted stuff

Meta Data

- E-mail content can be encrypted (S/MIME, PGP)
- But meta data is still unencrypted
- That includes
  - From, to, cc fields
  - Subject
- Meta data needed for technical reason, but also leaks information

Questions?

References I

- Diffie, W./M. Hellman (1976). “New directions in cryptography”. In: *IEEE Transactions on Information Theory* 22.6, pp. 644–654. DOI: 10.1109/TIT.1976.1055638.
- Merkle, Ralph C. (1978). “Secure Communications over Insecure Channels”. In: *Commun. ACM* 21.4, pp. 294–299. ISSN: 0001-0782. DOI: 10.1145/359460.359473. URL: https://doi.org/10.1145/359460.359473.