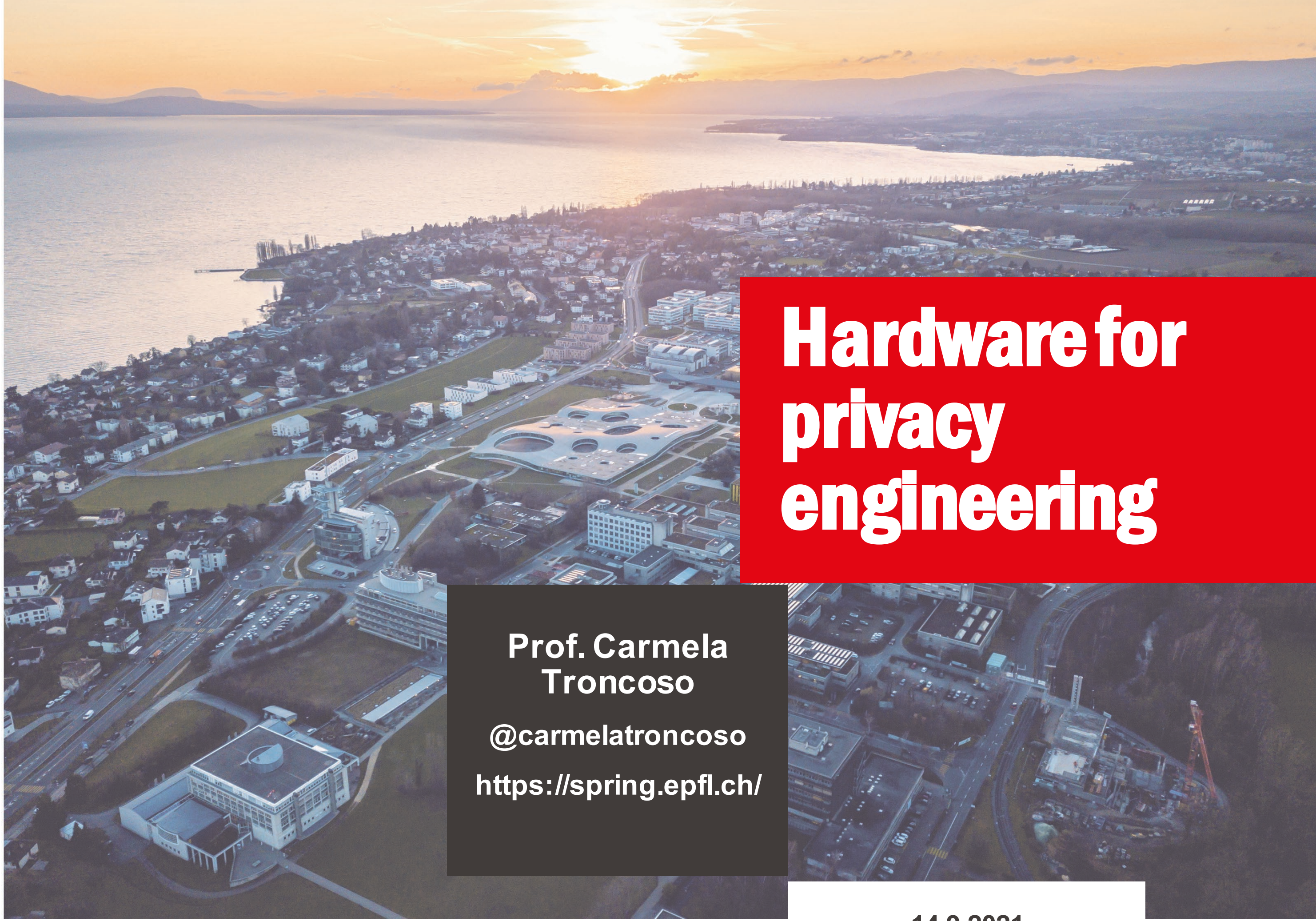


Hardware for privacy engineering

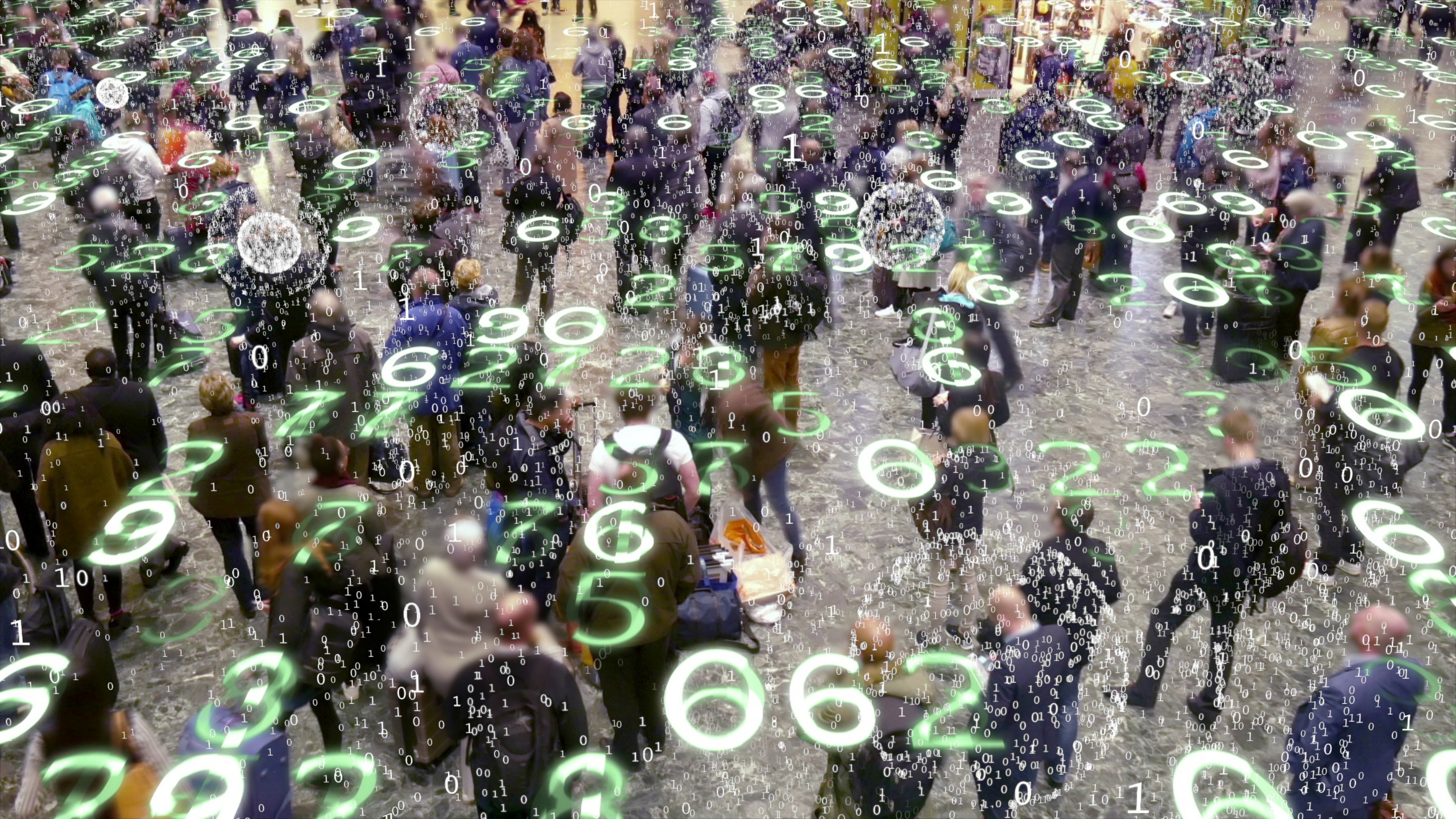
Prof. Carmela Troncoso

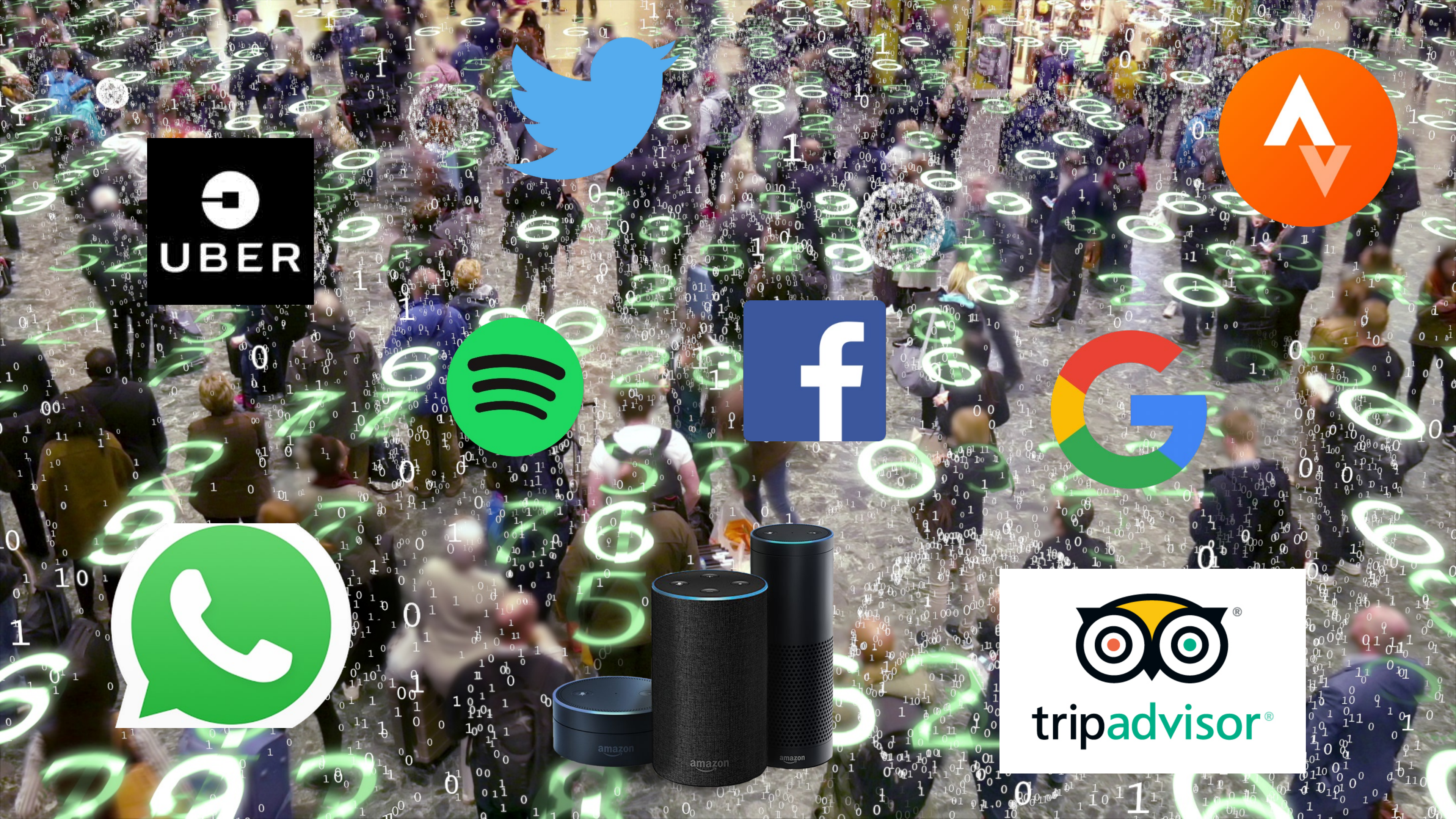
@carmelatroncoso

https://spring.epfl.ch/

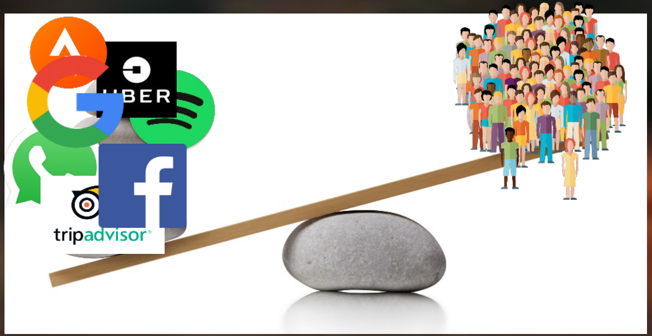
14.9.2021

École
polytechnique
fédérale
de Lausanne




UBER
tripadvisor®
amazon
amazon
amazon

UBER
f
tripadvisor

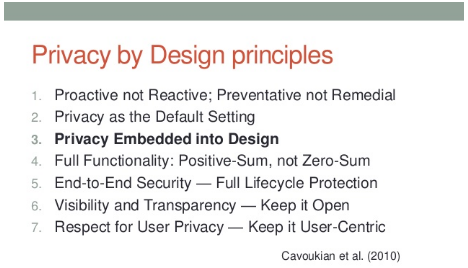


“Part of what makes a society a good place in which to live is the extent to which it
allows people freedom from the intrusiveness of others.
A society without privacy protection would be suffocation”

Solove (2007)

The goal: privacy by design

Privacy by Design



Privacy by Design principles

1. Proactive not Reactive; Preventative not Remedial
2. Privacy as the Default Setting
3. **Privacy Embedded into Design**
4. Full Functionality: Positive-Sum, not Zero-Sum
5. End-to-End Security — Full Lifecycle Protection
6. Visibility and Transparency — Keep it Open
7. Respect for User Privacy — Keep it User-Centric

Cavoukian et al. (2010)

Privacy Embedded into Design
"Privacy by design is embedded into the design and architecture of IT systems [...]. It is not bolted as an addon, after the fact. The result is that privacy becomes an essential component of the core functionality being delivered. Privacy is integral to the system without diminishing functionality".

GDPR
EU General Data Protection Regulation

"the controller shall [...] implement appropriate technical and organisational measures […] which are designed to implement data-protection principles[...] in order to meet the requirements of this Regulation and protect the rights of data subjects."

Companies should promote consumer privacy throughout their organizations and at every stage of the development of their products and services. Companies should incorporate substantive privacy protections into their practices, such as data security, reasonable collection limits, sound retention practices, and data accuracy. Companies should maintain comprehensive data management procedures throughout the life cycle of their products and services.

The goal: privacy by design

PbD
Privacy by Design

GDPR
EU General Data Protection Regulation

Privacy by Design principles

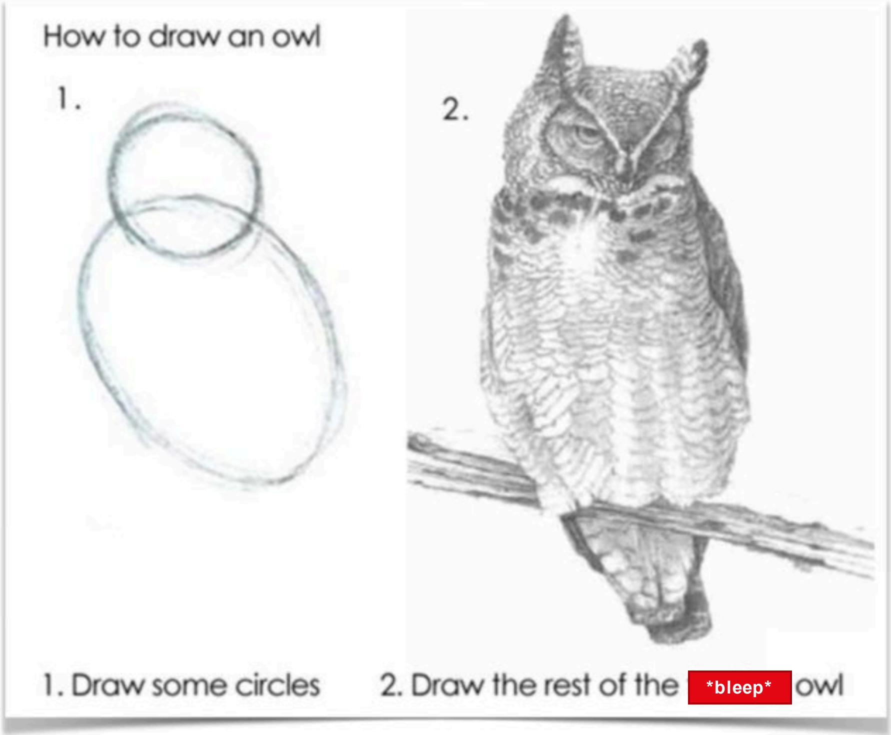
1. Proactive no
2. Privacy as th
3. **Privacy Eml**
4. Full Function
5. End-to-End S
6. Visibility and
7. Respect for U

Privacy Embedded into Design

“P ... the design and architecture of IT ... ddon, after the fact. The result is ... omponent of the core functionality ... the system without diminishing

“the ... orga ... data- ... of thi

Com ... and at every ... ould incor ... secu ... uracy. Com ... throughout the life cycle of their products and services.



Privacy as data minimization

Build systems without data!

The least data in the system, the more privacy-preserving it is

Clearly related to a regulation principle

but, **it's not "data" that is minimized** (in the system as a whole)

data is kept in user devices

sent encrypted to a server (only client has the key)

distributed over multiple servers

…

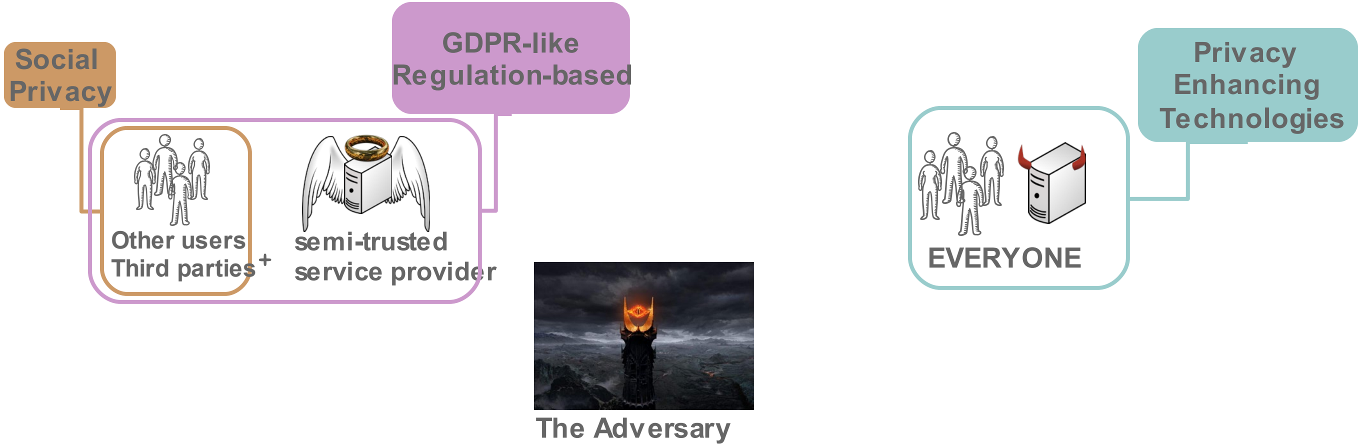
"data minimization"
on its own
is a ***BAD*** *metaphor*
for privacy-preserving
designs

Seda Gurses, Carmela Troncoso, Claudia Diaz. Engineering Privacy by Design.Computers, Privacy & Data Protection. 2011

Privacy as trust minimization

Overarching goal

Minimizing privacy **risks** and
trust assumptions placed on other entities



Social Privacy

GDPR-like Regulation-based

Other users
Third parties +

semi-trusted service provider

The Adversary

Privacy Enhancing Technologies

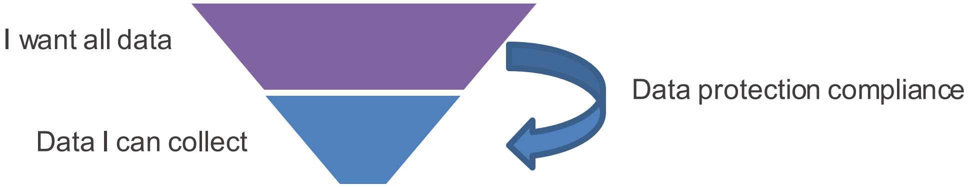
EVERYONE

Seda Gurses, Carmela Troncoso, Claudia Diaz. Engineering Privacy by Design Reloaded. Amsterdam Privacy Conference. 2015

- Seda Gurses and Claudia Diaz. "Two tales of privacy in online social networks." IEEE Security & Privacy Magazine. 2013

Trust minimization through purpose limitation

The Usual approach

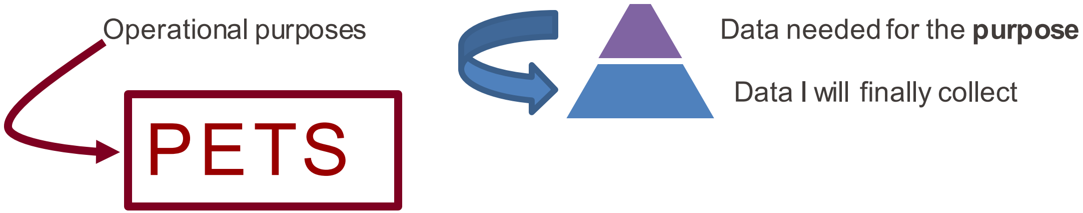


I want all data

Data I can collect

Data protection compliance

The Privacy engineering approach



Operational purposes

PETS

Data needed for the **purpose**

Data I will finally collect

technological

“purpose limitation”

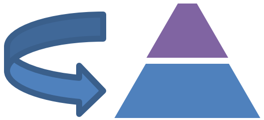
limits the need for trust

*It is a **GOOD** metaphor for privacy-preserving designs*

Privacy by design as purpose limitation

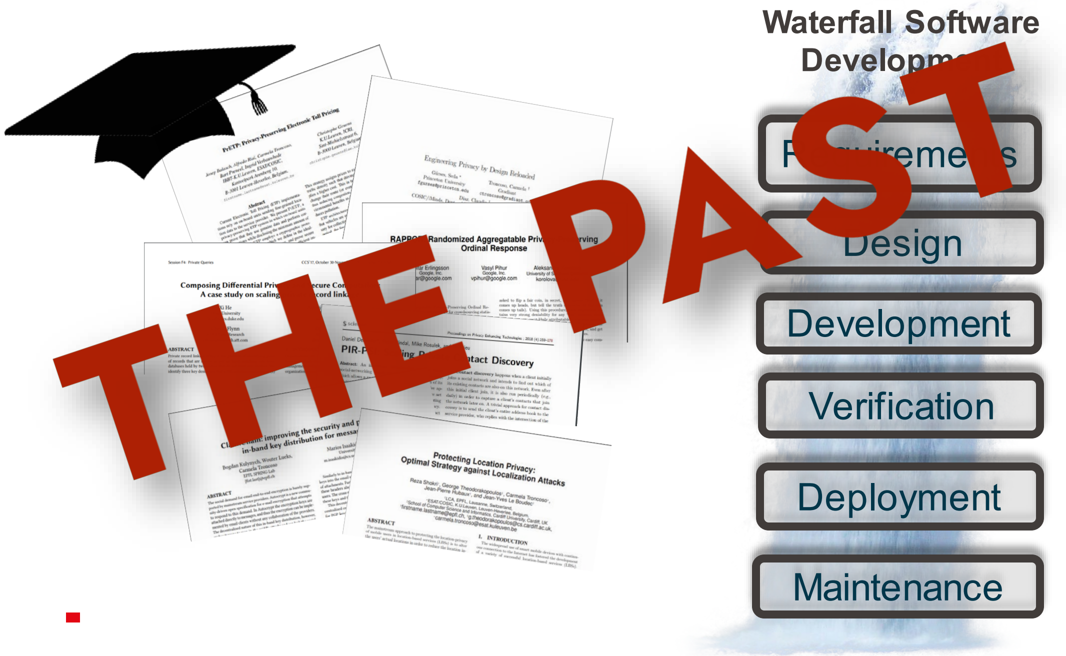
The Privacy engineering approach

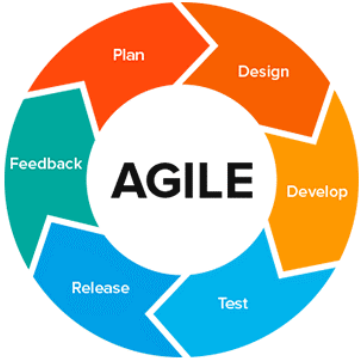
Operational purposes
PETS

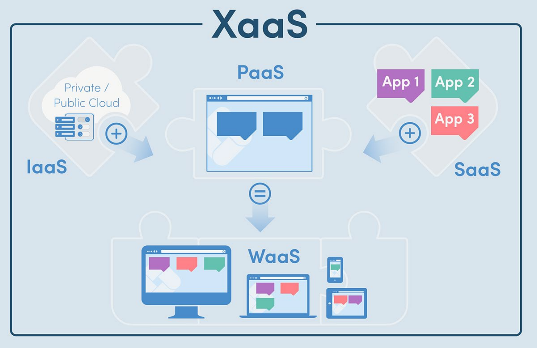


Data needed for the **purpose**

Data I will finally collect


Waterfall Software Developm
Requirements
Design
Development
Verification
Deployment
Maintenance
THE PAST
PrETP: Privacy-Preserving Electronic Toll Pricing
Engineering Privacy by Design Reloaded
RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response
Composing Differential Privacy and Secure Computation: A case study on scaling private record linkage
PIR-PSI
Contact Discovery
Claimchain: improving the security and privacy of in-band key distribution for messaging
Protecting Location Privacy: Optimal Strategy against Localization Attacks





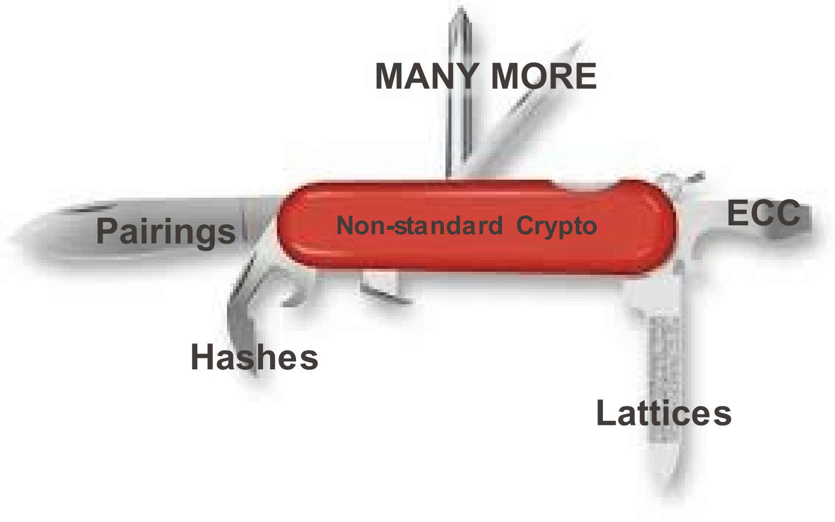
Decentralized Search Engine for Investigative Journalists

- Blank slate for design
- “Total” control of the stack
- Space to deploy any crypto
- Space to design whole system

COVID-19 Contact Tracing Apps

- Design based on existing components
- Reliance on services (Mobile/Cloud)
- Little space for own crypto
- End-to-end design space limited by others

A privacy-preserving Hardware Crypto Swiss-knife



Cryptographic tools for building privacy-preserving systems

- **Private Set Intersection**
  - private contact discovery, private search (Apple’s CSAM detection, Datashare), intrusion detection…
  - Based on
    - Modular exponentiation or ECC
    - Hashing
    - Multiparty computation (mostly 2PC)
    - Oblivious PRF
    - …
  - Use of **cuckoo filters** for efficiency of communication and speed
  - Requires **one operation per element in a dataset**

- De Cristofaro, Emiliano, and Gene Tsudik. "Practical private set intersection protocols with linear complexity." *Financial Crypto* (2010).
  Weinert, Christian. Practical Private Set Intersection Protocols for Privacy-Preserving Applications. PhD Thesis. TU Darmstadt (2021)

Cryptographic tools for building privacy-preserving systems

- **Private Information Retrieval**
  - private messaging, certificate transparency, private media browsing,…
  - Computational (one server) vs. Information theoretical (several servers)
  - Based (mainly) on Homomorphic Encryption (mostly lattices)
  - Use of **packing and batching** or **offline pre-processing** for efficiency of communication and speed
  - Requires **many operations** on the server side, proportional to dataset size

B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, “Private Information Retrieval,” FOCS (1995)

- R. Henry. Tutorial; Private Information Retrieval. CCS (2017)

Cryptographic tools for building privacy-preserving systems

- **Attribute-based credentials**
  - Anonymous authentication [Can’t build end-to-end without authentication!!]
  - Zero-knowledge proofs and/or blind signatures
  - Based on
    - Modular exponentiation or ECC
    - Hash functions
    - Bilinear pairings
  - Many extensions to achieve multiple properties (revocation, limited usage,…)
  - Requires **one operation** on the client/server side, but expensive

David Pointcheval, Olivier Sanders. Short Randomizable Signatures. CT-RSA (2016)
Camenisch, J., Hohenberger, S., Kohlweiss, M., Lysyanskaya, A., & Meyerovich, M. How to win the clonewars: efficient periodic n-times anonymous authentication. CCS (2006)

Cryptographic tools for building privacy-preserving systems

- **Many more…**
  - Underlying building blocks are (mostly) common
  - Bottlenecks are common

- **Take a look at privacy-preserving systems proposed in**
  - **Proceedings of Privacy Enhancing Technologies Symposium**
  - **Usenix Security**
  - **ACM Computer and Communication Security**

-

We have the protocols… so why is there no privacy?

- Hard to deploy in practice
  - Based on non-standard –use– of cryptographic primitives (mainly academic)
  - SW libraries need to build from scratch, no existing hardware support
    - Few libraries exist, pseudo academic, suboptimal support
  - They are **very slow**, no existing hardware acceleration (focus on AES, RSA)
  - They are **very energy consuming**, no efficient implementations available

How can you help?

- Build hardware support for non-standard cryptographic primitives

  - Build accelerators for privacy-preserving cryptographic primitives
    - ECC and modular exponentiation are used beyond their typical contexts!

  - Build hardware for commodity devices **and** independent

  - Build energy-efficient implementations
    - Some fast
    - Some small and very efficient (IoT or offline phases)
    - Some with both

  - Ensure you talk with developers when deciding the abstraction offered to the exterior
    - If we cannot program it, it does not exist

We done?

No, but now at least we can start

We done?

No, but now at least we can start… to build systems

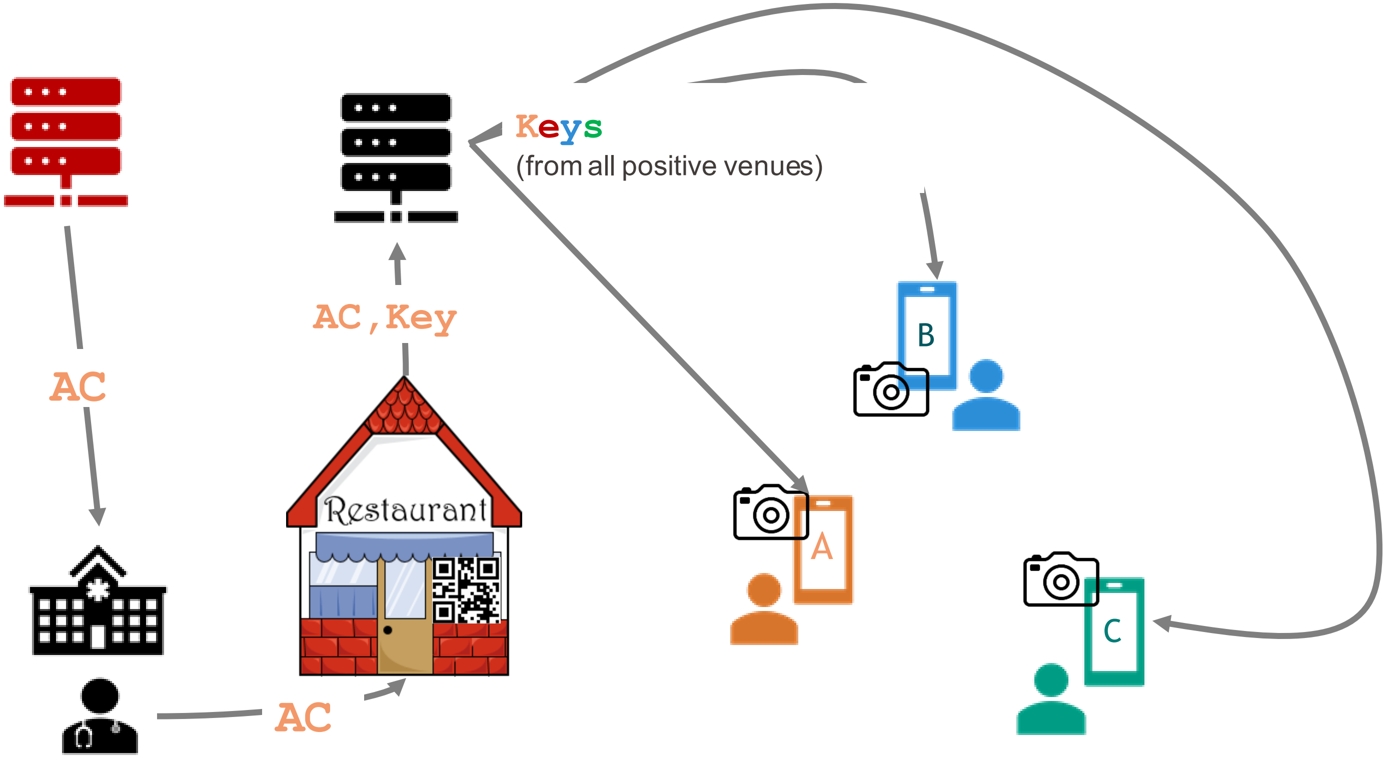
Batch operations

- It’s private… so many times we don’t know which record we are interested in, or we don’t want to show which record we are interested in
  - Private Set Intersection: needs one operation per element
  - Attribute Based Credentials: some blacklists require one operation per revocation
  - PIR: homomorphic operations needed on all records

Very used batch-friendly trick: trial-decrypt

- It’s private… so many times we don’t know what key to use
  - So we try all of them

Very used batch-friendly trick: trial-decrypt



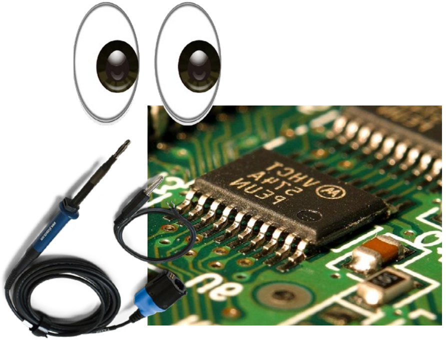
Very used batch-friendly trick: trial-decrypt

- It’s private… so many times we don’t know what key to use
  - So we try all of them

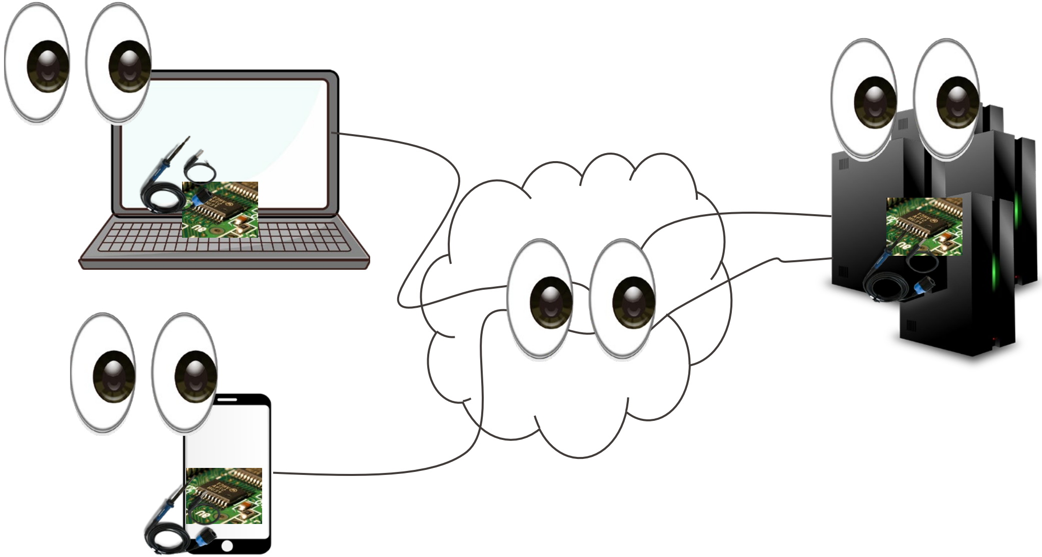
- Not only batch of ciphertexts, also keys! (quadratic)

- Good news
  - Can be probabilistic
  - Can abort early (timing attacks?)
  - Can use some plaintext (trade-off with privacy)

The privacy adversary sees the whole system

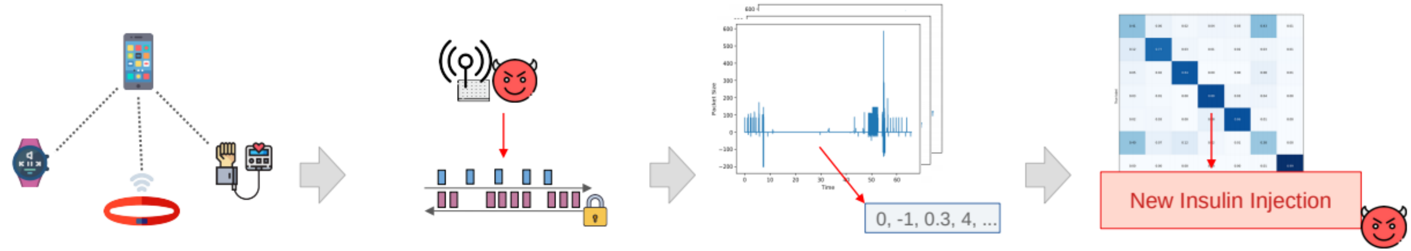


Hardware adversary



Privacy adversary

Traffic Analysis: encryption is only the first step to privacy

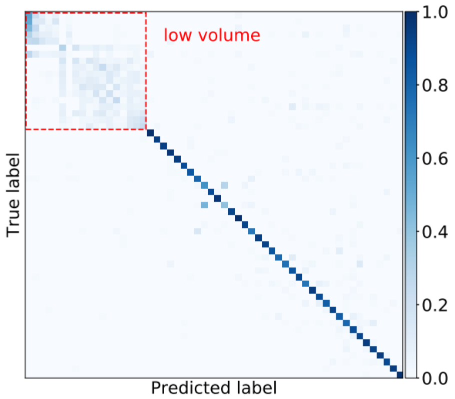


(1) Wearable devices communicate with a smartphone using Bluetooth

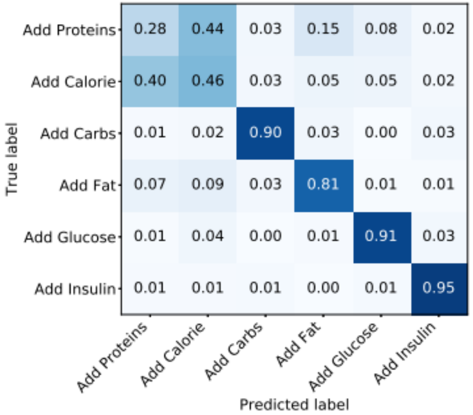
(2) An adversary eavesdrops upon their encrypted communications

(3) Feature extraction

(4) Traffic classification: identification of sensitive actions



Recognize actions
heartbeat measurement,
food intake,
send message,
receive message…



	Add Proteins	Add Calorie	Add Carbs	Add Fat	Add Glucose	Add Insulin
Add Proteins	0.28	0.44	0.03	0.15	0.08	0.02
Add Calorie	0.40	0.46	0.03	0.05	0.05	0.02
Add Carbs	0.01	0.02	0.90	0.03	0.00	0.03
Add Fat	0.07	0.09	0.03	0.81	0.01	0.01
Add Glucose	0.01	0.04	0.00	0.01	0.91	0.03
Add Insulin	0.01	0.01	0.01	0.00	0.01	0.95

L. Barman, A. Dumur, A. Pyrgelis, J-P. Hubaux. Every Byte Matters: Traffic-Analysis of Bluetooth Wearable Devices. Ubicomp (2021)

The solution: dummy traffic

- Also known as chaff, or fake traffic

- Goal: hide information from a network observer

- Designs do one of more of:
  - Add packets at random intervals – the distribution matters!
  - Add packets to mimic other applications/websites/actions behaviour

- High cost: computation and bandwidth (and energy) – rarely deployed

What can you do?

- Reduce the energy consumption
  - Can computation and network costs be combined? Co-design of crypto and network?
  - Can we reuse energy from sending packets on the network?
  - Can we create dummies on hardware?
    - Dummies just need to be random
    - Timing is independent of real traffic (that is the point)

  - Sometimes speed is not a requirement, can afford slow,efficient computation

Negative results are also ok!

- And if all the above cannot be done… research that shows the limits of protection is as important as solutions

- Informing the policy discussion about limitations shapes the space of deployments (otherwise they will believe that the holy grail will appear)

Key takeaways

- Privacy engineering is about minimizing trust – through purpose limitation

- Limiting purpose requires use of non-standard cryptographic and non-cryptographic solutions
    - Solutions are constrained by infrastructure **including hardware**

- Need better support for non-standard crypto primitives and other privacy-preserving building blocks!

- If you cannot, showing the limits is equally important