

CS 798

Privacy in Computation and Communication

Module 1

What is Privacy?

Spring 2024

Ian Goldberg

- iang@uwaterloo.ca
- <https://cs.uwaterloo.ca/~iang/>
- Online office hours: Tuesdays 11:00 am–noon (or by appointment)
- Office hours link on LEARN

Course personnel

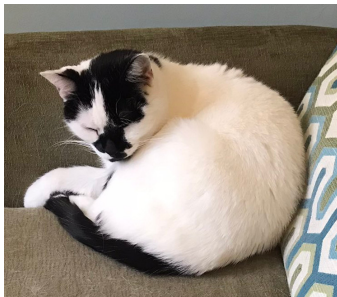
TA:

- Vecna (they/them)

Office hours: Thursdays 1–2 pm
(Same office hours link)

Special course personnel:

- Luna



(Come to Prof. Goldberg's office hours)

Recording of this course

- We will be recording the class sessions
- Full slides/audio/video/chat will be available to people taking the course this term (except when recording is paused; see below)
- Slides/audio/video of me (not video of students or any chat) are intended to be (lightly edited and) published online later
- If you want to ask a question, participate in discussion, etc., but not be recorded in the public version, either:
 - Participate via chat
 - In chat, indicate you'd like us to pause the recording

Pilot offering

- This is the very first offering of this course
- Things may not go perfectly smoothly
- We'll do our best, but please be understanding :-)
- And feedback on what is and is not working will be very helpful!
 - Throughout the term, not just in the course perception surveys at the end

Course mechanics

- LEARN: course info, recordings, assignments, grades, etc.
- Piazza: Q&A, general discussions
- BigBlueButton (BBB): course lectures, office hours (no need to make an account!)
- Course website: syllabus, slides, public materials
- uWaterloo GitLab: assignment submission

All links to the above are on LEARN.

Communication channels

- Important course announcements will be made on Piazza
 - Please keep up with the information there
- Use discussion forums in Piazza for all communication
 - Use a private question for questions not of general interest
- Use email only as a last resort, and then it must be from your uwaterloo.ca email address.
- Some communication might be sent *to* your uWaterloo email address
 - Check it regularly

Grades

- There will be four graded assignments in this course, each worth 25% of your course grade
- Three due during the term, the fourth during the final exam period
- Deadlines are already posted to LEARN and the course website
- Assignments are to be done individually
 - No sharing code or text
 - General rule: discussion with your classmates is fine, but don't write anything down during it
 - That will help avoid academic integrity issues

Assignment 0

- There is an additional *ungraded* assignment (available now!) that will help walk you through getting set up with git and docker (the tools you will use to submit all assignments)
- Collaboration on Assignment 0 is *encouraged*; indeed, please do so on Piazza so that everyone can benefit!

Assignments

- To do the assignments, you will minimally need:
 - A desktop or laptop on which you can install and run git and docker
 - The ability to write programs in at least one of Rust, C++, Python 3
 - A tool to produce written documents as PDFs (pdflatex preferred, but really anything is fine)
- You will submit your assignments as git repositories you will create (as part of Assignment 0) on uWaterloo's GitLab
<https://git.uwaterloo.ca>.

Other readings

- From time to time, there will be additional assigned readings
- Links will be provided from the class schedule page in LEARN
- They will mostly be supplementary material, but some may be important to completing the assignments

- ① What is privacy?
- ② Background
- ③ Privacy in computation: distributed trust
- ④ Privacy in computation: trusted hardware
- ⑤ Privacy in computation: homomorphic encryption
- ⑥ Privacy in communication: protecting metadata
- ⑦ Privacy in communication: censorship resistance

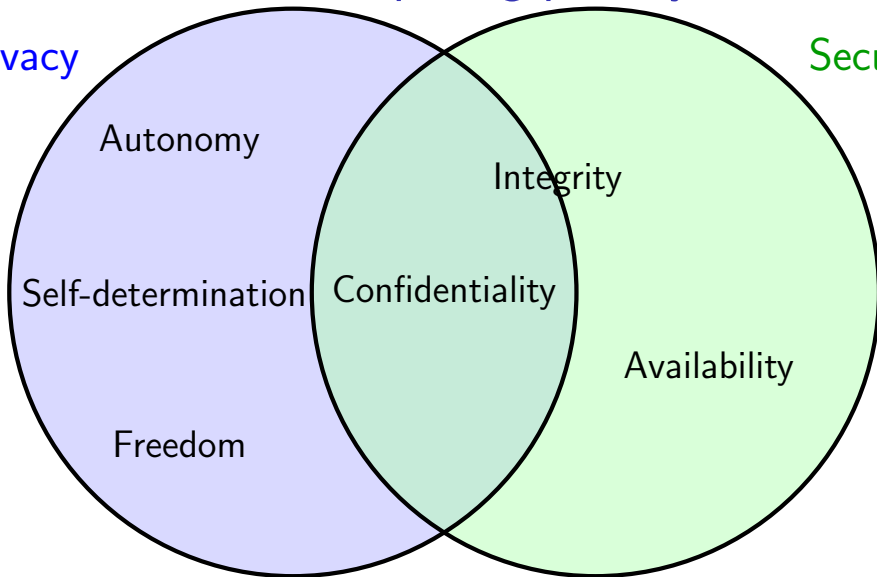
What is privacy?

- Surprisingly hard to concisely define
- Hiding personal information from others?
- Controlling who can do what with information about you?
- Being able to make decisions without undue external influence?

Comparing privacy with security

Privacy

Security



The importance of privacy

- Treating people with dignity as individuals
- When you lose privacy:
 - Surveillance and censorship
 - Coercion and social sorting
 - Subversion of democracy
- Privacy is a social good, not just an individual good
- Synergy between computer science and social science
 - CS 858 / SOC 701 “Surveillance and Privacy”

Privacy in computation

- Data breaches are exceptionally common

Senior citizens struck by data breach

Updated on: April 19, 2024 12:25 PM 

AT&T: Data breach affects 73 million or 51 million customers. No, we won't explain.

When the data was published in 2021, the company said it didn't belong to its customers.

DAN GOODIN - 4/10/2024, 6:28 PM

Nova Scotia Health contacting 2,690 patients after privacy breach

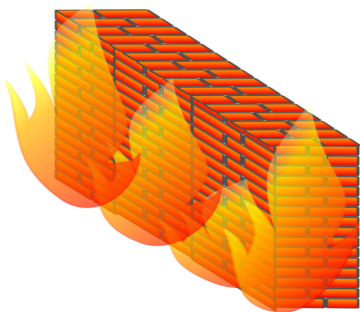
Investigation into breach began in September 2023 at Saint Martha's Regional Hospital in Antigonish, N.S.



Anjali Patil · CBC News · Posted: Apr 19, 2024 3:22 PM EDT | Last Updated: 1 hour ago

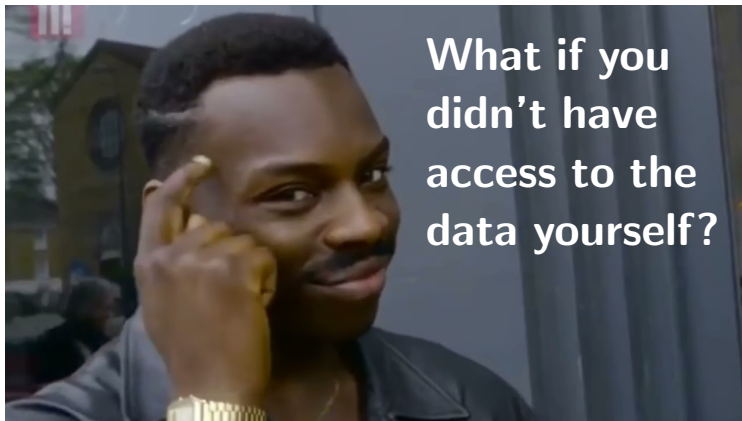
Why is computer security failing?

- Large, complex systems
- Large, complex **trusted computing bases** (TCB)
- “Tootsie pop” security



A modest proposal

- If you are a company and don't want to have your customer / user / data subject information stolen from you...



Privacy in computation

- Goal: be able to process information without having access to the information itself
- Obviously impossible?
- Perhaps not...

Privacy in computation

- Three approaches:
 - Distributed trust
 - Trusted hardware
 - Homomorphic encryption

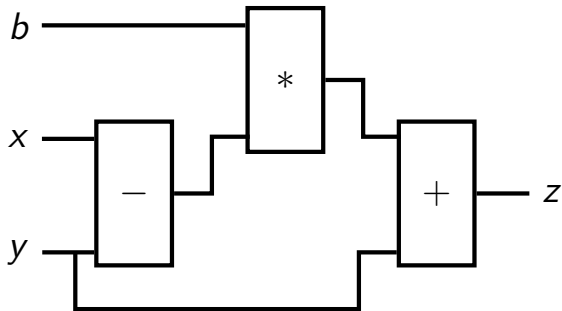
Distributed trust

- Key idea: split the information across multiple servers
- Not that each server has a subset of the data
- For example, you could have two servers, where server 1 holds R and server 2 holds S such that $R + S =$ the original data
 - Each of R and S is completely random, and contains *no information* about the original data
 - In a very strong mathematical sense
- So if either server is compromised, *no information* is leaked to the attacker
 - But if *both* are, you're out of luck!

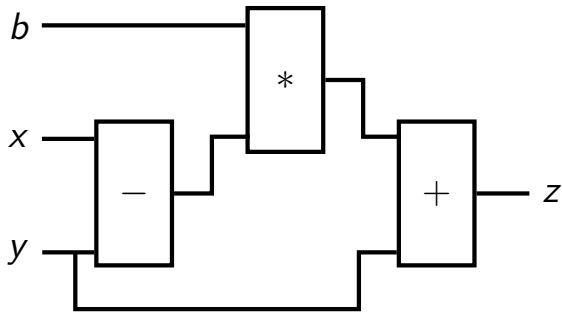
Distributed trust

- Compile the program P you want to run on the data into a “circuit” consisting of binary or arithmetic gates
- Note that you cannot have conditional execution like “if statements”, since the servers *cannot see the data!*
 - Oblivious algorithms

Oblivious computation



Oblivious computation



$$z = \begin{cases} y & \text{if } b = 0 \\ x & \text{if } b = 1 \end{cases}$$

Secure multiparty computation

- Perform **Secure Multiparty Computation (MPC)** to evaluate the circuit
- In MPC, multiple parties each have an input to a computation, and they execute some protocol to compute the output of some program over all of their inputs, without any party learning any other party's input
- Typically, “linear” operations (like addition of two variables or multiplying a variable by a constant) are extremely easy *local* operations, but more complex operations (like multiplying two variables) require parties to communicate
- Reading or writing to a secret-shared memory *at a secret-shared location* is one of the most complex operations

Trusted hardware

- Trusted Execution Environments (TEEs)
 - Intel SGX, AMD SEV, ARM TrustZone, etc.
 - What does “Trusted” mean here?
- Encrypt the data to a key inside the CPU
- The CPU can decrypt and process the data
- The rest of the computer, including the operating system, **cannot** see the data
 - But watch out for **side channels**
 - Also requires *oblivious algorithms* to maintain privacy

Trusted hardware

- On Intel SGX for example, each program can have its own *enclave*, and data encrypted for one enclave cannot even be decrypted by any other enclave.
- There is also a facility so that other programs running on the same machine (“local attestation”) or a different machine (“remote attestation”) can be assured they are communicating with, or encrypting data to, the enclave running a particular program, or one signed/authorized by a particular entity.

Trusted hardware

- TEEs decrypt and process the data (and typically re-encrypt the result)
 - But there's a lot of trust required in the hardware design
 - And the side channels to be managed, since the CPU *does* see the decrypted data
- What if the server could process the encrypted data *without decrypting it at all?*

Homomorphic encryption

- In 2009, a new kind of cryptography was discovered
 - Fully homomorphic encryption (FHE)
- Since then, there have been substantial improvements
 - Also implementations, libraries, etc.
- What does it do?
- I have some data x , you have a program P . I want you to compute $P(x)$ for me, but I don't want you to learn x (and often, you don't want me to learn P).
- With FHE, I send you $E(x)$, and with P and $E(x)$, you can compute $E(P(x))$ and send it back to me.

Homomorphic encryption

- General idea: be able to support a handful of simple operations on encrypted data
 - Minimally, be able to compute $E(x + y)$ and $E(x \cdot y)$ from $E(x)$ and $E(y)$
- Then (as with MPC) turn your program P into a circuit consisting of these simple operations
- Again, you can't have "if statements", since the server can't see the data
 - Oblivious algorithms come up again

Homomorphic encryption

- There are several current FHE protocols, with slightly different properties
- We will **not** be discussing *how* the FHE algorithms work (i.e., the math under the hood)
- We **will** be discussing what you can *do* with them

Privacy in communication

- The next set of modules will discuss privacy in communication
- Who can see what when people communicate online?
- What might you want to protect?
 - The *contents* of a message
 - The *metadata* of a message
 - The *existence* of a message
 - *Proof* of any of the above

Protect those things from whom?

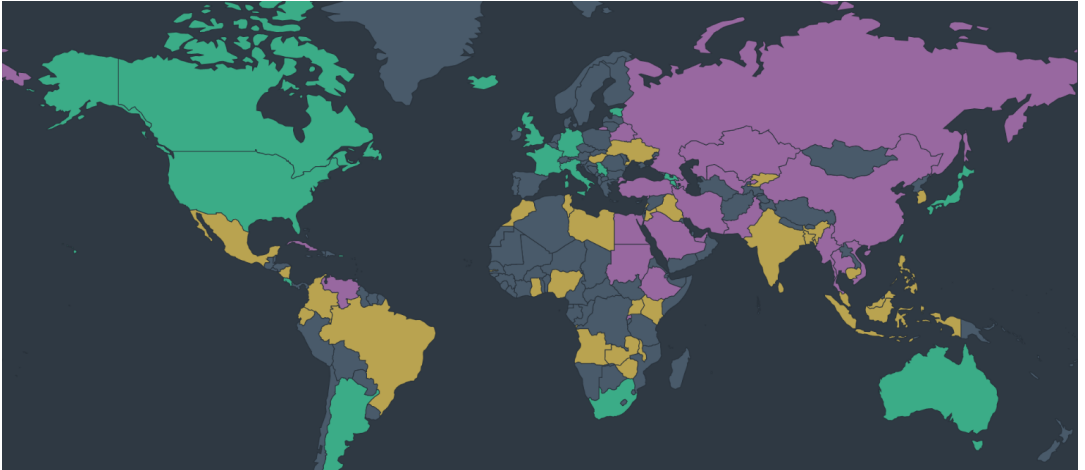
- Third parties
 - ISPs, eavesdroppers (local or global), governments
- Servers facilitating the message delivery
 - Apple, Signal, etc.
- The people you're communicating with
 - One-on-one or in a group
- Malware running on your own device
 - Tough to do?

Protecting metadata

- We'll look at various technologies to protect communication:
- Messaging apps (and their shortcomings)
- Tor (and its shortcomings)
- Metadata-Protecting Communication Systems (active research area)

Internet censorship

- Many countries censor the Internet



<https://freedomhouse.org/explore-the-map?type=fotn&year=2023>

Internet censorship

- What is it?
- Why does it happen?
- What technologies can people use to evade Internet censorship?
- Why is this a *privacy* issue?