Le Métayer gives us an overview of PETs from a particular perspective, focused on one key aspect of these technologies, namely, the kind of trust they can provide. Weŕe going to talk about that perspective, and then go through the overview.

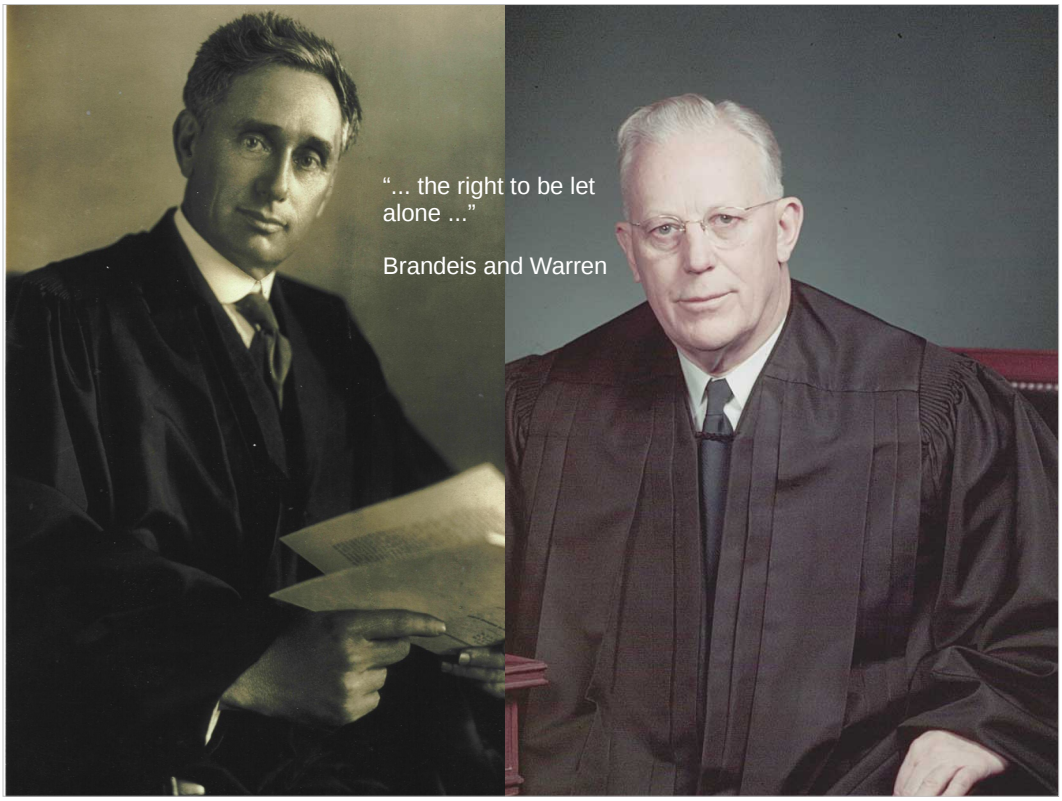
Small mention of interesting things

- No encrypted mail, yet
- Assignment 4: “Trust me”
- Janez Janša
- mozilla.dev.security.policy

The U.S. Supreme Court has ruled that federal administrative agencies can invoke the All Writs Act to preserve the status quo when a party within the agency's jurisdiction is about to take action that will prevent or impair the agency from carrying out its functions.

Two views organizing research on privacy

- Privacy as control
- **Privacy as confidentiality**



The idea is captured in the famous understanding of privacy as a right "to be left alone." Freedom from intrusion is central to privacy as "confidentiality" as Gürses and Diaz point out

According to this vision, everyone might be untrustworthy, so trust should be minimized.

"A different family of privacy technologies considers however that placing such high levels of trust in organizations should be avoided whenever possible, as they leave individuals vulnerable to incompetent or malicious organizations." (Diaz et al. 2, our next reading)

-

Aim is to avoid trusting the untrustworthy---inverse problem of not trusting the trustworthy is overlooked

UCL

Two families of privacy technologies

Soft Privacy Technologies

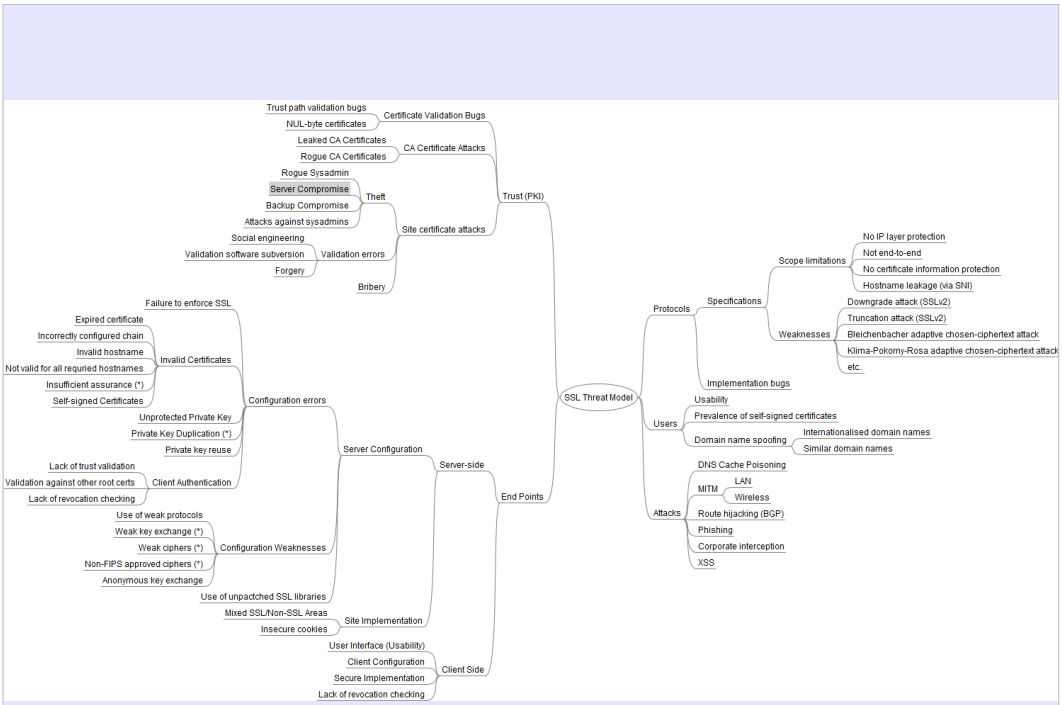
- Focus on compliance.
- Focus on "internal controls".
- **Assumption: a third party is entrusted with the user data.**
- Threat model: third party is trusted to process user data according to user wishes.
- Examples technologies:
  - Access control, tunnel encryption (SSL/TLS)
- "Keeping honest services safe from insiders / employees".

Hard Privacy Technologies

- Stronger focus on data minimization.
- **Assumption: there exists no single third party that may be trusted with user data.**
- Threat model: a service is in the hands of the adversary; may be coerced; may be hacked.
- Common assumption: k-out-of-n honest third parties.
- May relay on service integrity if auditing is possible.
- **Challenge: achieve functionality without revealing data!**

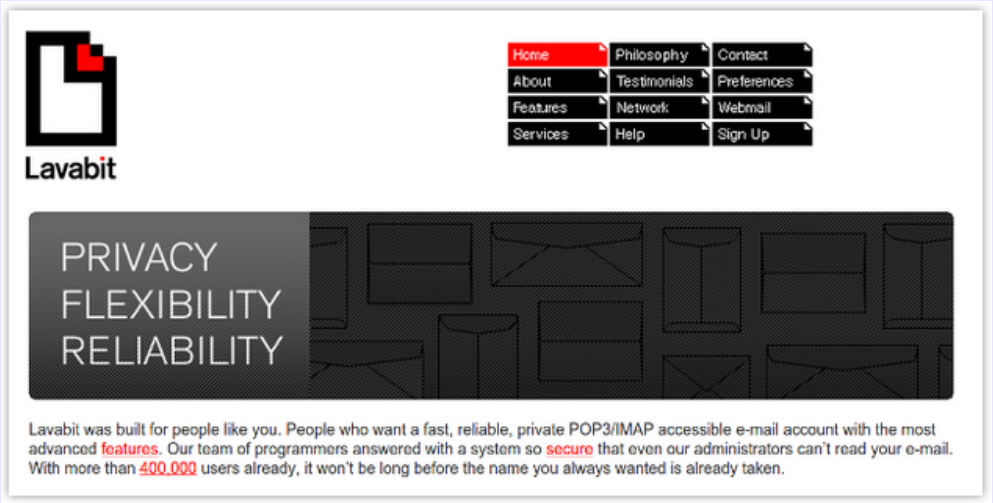
Slide credit: George Danezis

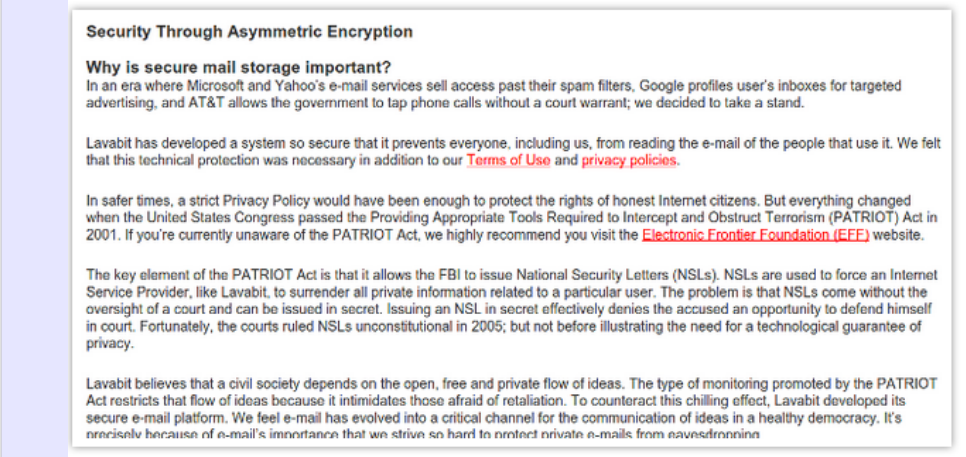
walk through (notice bad examples of soft privacy technology)



https://www.ssllabs.com/downloads/SSL_Threat_Model.png

walk through


Lavabit
Home
Philosophy
Contact
About
Testimonials
Preferences
Features
Network
Webmail
Services
Help
Sign Up
PRIVACY
FLEXIBILITY
RELIABILITY
Lavabit was built for people like you. People who want a fast, reliable, private POP3/IMAP accessible e-mail account with the most advanced features. Our team of programmers answered with a system so secure that even our administrators can't read your e-mail. With more than 400,000 users already, it won't be long before the name you always wanted is already taken.



Security Through Asymmetric Encryption

Why is secure mail storage important?
In an era where Microsoft and Yahoo's e-mail services sell access past their spam filters, Google profiles user's inboxes for targeted advertising, and AT&T allows the government to tap phone calls without a court warrant; we decided to take a stand.

Lavabit has developed a system so secure that it prevents everyone, including us, from reading the e-mail of the people that use it. We felt that this technical protection was necessary in addition to our Terms of Use and privacy policies.

In safer times, a strict Privacy Policy would have been enough to protect the rights of honest Internet citizens. But everything changed when the United States Congress passed the Providing Appropriate Tools Required to Intercept and Obstruct Terrorism (PATRIOT) Act in 2001. If you're currently unaware of the PATRIOT Act, we highly recommend you visit the Electronic Frontier Foundation (EFF) website.

The key element of the PATRIOT Act is that it allows the FBI to issue National Security Letters (NSLs). NSLs are used to force an Internet Service Provider, like Lavabit, to surrender all private information related to a particular user. The problem is that NSLs come without the oversight of a court and can be issued in secret. Issuing an NSL in secret effectively denies the accused an opportunity to defend himself in court. Fortunately, the courts ruled NSLs unconstitutional in 2005; but not before illustrating the need for a technological guarantee of privacy.

Lavabit believes that a civil society depends on the open, free and private flow of ideas. The type of monitoring promoted by the PATRIOT Act restricts that flow of ideas because it intimidates those afraid of retaliation. To counteract this chilling effect, Lavabit developed its secure e-mail platform. We feel e-mail has evolved into a critical channel for the communication of ideas in a healthy democracy. It's precisely because of e-mail's importance that we strive so hard to protect private e-mails from eavesdropping.

- Unlike the design of most secure servers, which are ciphertext in and ciphertext out, this is the inverse: plaintext in and plaintext out. The server stores your password for authentication, uses that same password for an encryption key, and promises not to look at either the incoming plaintext, the password itself, or the outgoing plaintext.
-
- The ciphertext, key, and password are all stored on the server using a mechanism that is solely within the server's control and which the client has no ability to verify. There is no way to ever prove or disprove whether any encryption was ever happening at all, and whether it was or not makes little difference."


“Even though they advertised that they ‘can’t’ read your email, what they meant was that they would choose not to.”
Moxie Marlinspike
https://moxie.org/blog/lavabit-critique/

End-to-End Encryption

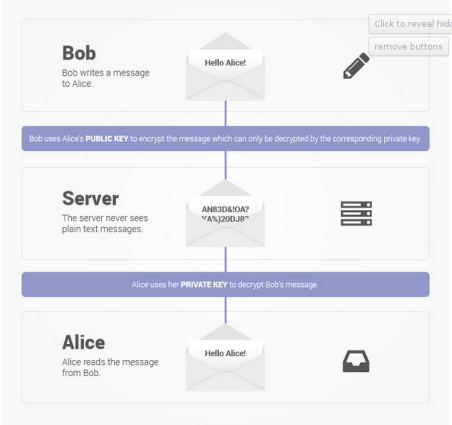
Messages are encrypted at all times

Messages are stored on ProtonMail servers in encrypted format. They are also transmitted in encrypted format between our servers and user devices. Messages between ProtonMail users are also transmitted in encrypted form within our secure server network. Because data is encrypted at all steps, the risk of message interception is largely eliminated.

Zero Access to User Data

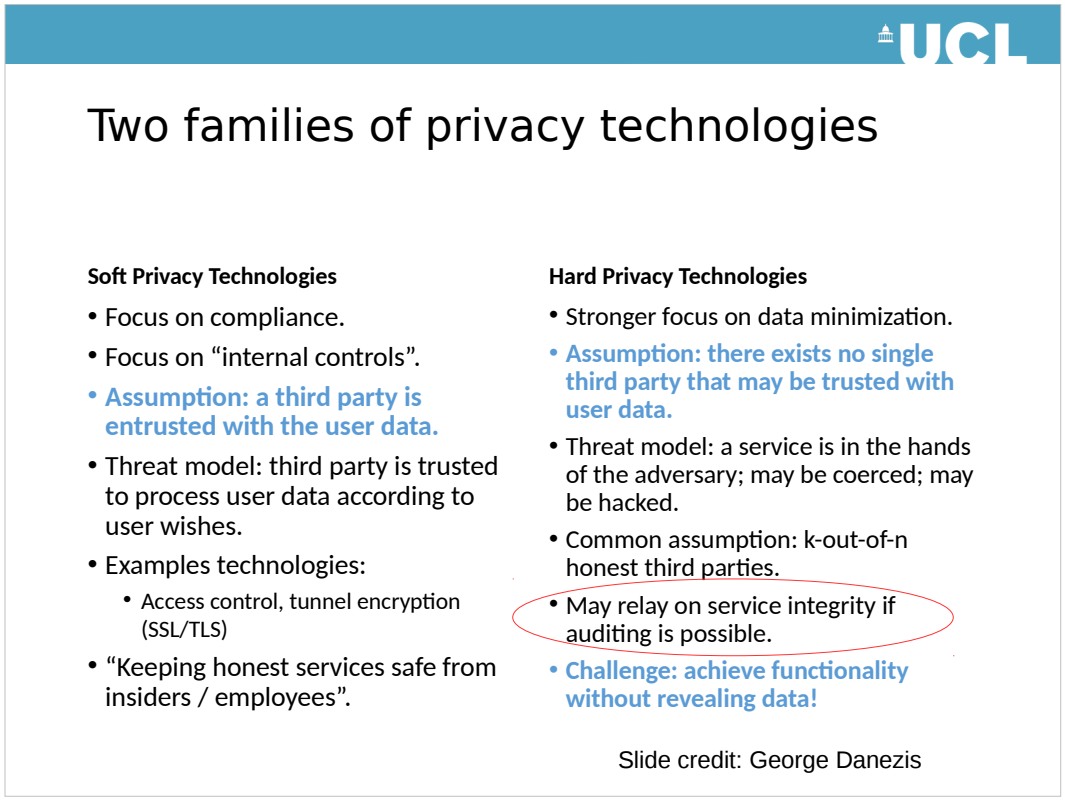
Your encrypted data is not accessible to us

ProtonMail's zero access architecture means that your data is encrypted in a way that makes it inaccessible to us. Data is encrypted on the client side using an encryption key that we do not have access to. This means we don't have the technical ability to decrypt your messages, and as a result, we are unable to hand your data over to third parties. With ProtonMail, privacy isn't just a promise, it is mathematically ensured. For this reason, we are also unable to do data recovery. If you forget your password, we cannot recover your data.

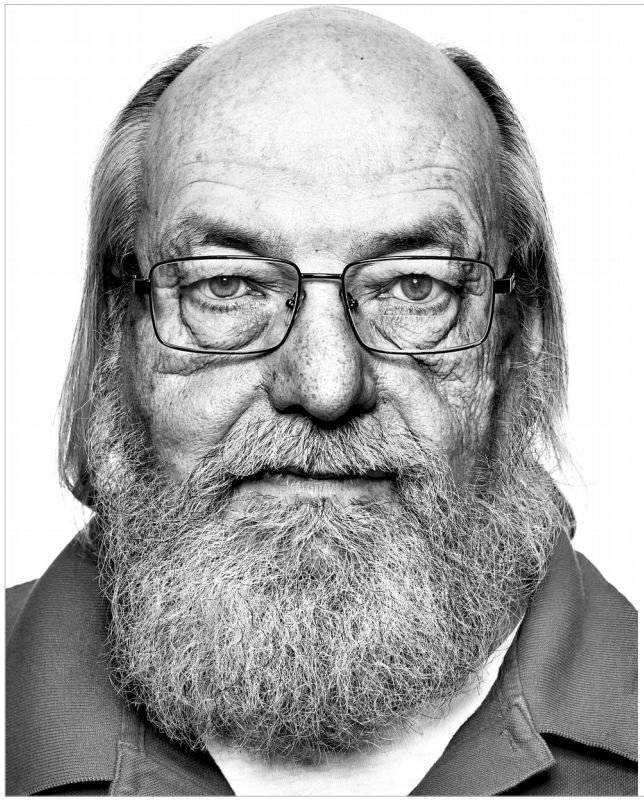


End-to-end encryption means that no one but the intended recipient can read the message

By contrast, protonmail does it’s cryptographic operations client-side



- Trust as field-verifiability (recall Ross Anderson on a UK military view of trust). Good if you can get it...
- Also, note that technology depends on lots of things. And the lesson of "Reflections on Trusting Trust" was that audit is not a panacea
- There is a role for both of these technologies, it's not just that Soft technologies paper over bad design in a way that hard privacy technologies don't



The moral is obvious. You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me). No amount of source-level verification or scrutiny will protect you from using untrusted code.

Ken Thompson, ACM Turing Award Speech, “Reflections on Trusting Trust”

This is a tricky problem. It eludes in principle the approach to auditing the security or privacy properties of software by examining it’s source code.

- Question for class: What are the **really** hard, “adamantine” technologies that address the case where even the creator of the technology cannot be trusted?

- Laptop Lens Covers
- Direct Introspection Device
- Decoy based encryption
- ...
- Key shared property: “directly field verifiable”
- none of these is forever, of course...

Edward Snowden and bunnie Huang

Figure 1: Top Secret slides extracted from the Snowden Archive illustrating one intelligence agency's perspective on metadata and location services offered by a major US brand [9]

Let's check out the introspection engine: video at 38:39 - 41:53

introspection engine

Technologies of “confidentiality”

- Diaz and Gürses terminology is a little awkward here
- Anonymous authentication protocols
- Anonymous communication networks
- Private Information Retrieval
- ... all require judicious use of modern cryptography

authentication: selective disclosure credentials, functional encryption more generally

resistance to traffic analysis is central to anonymous communication networks

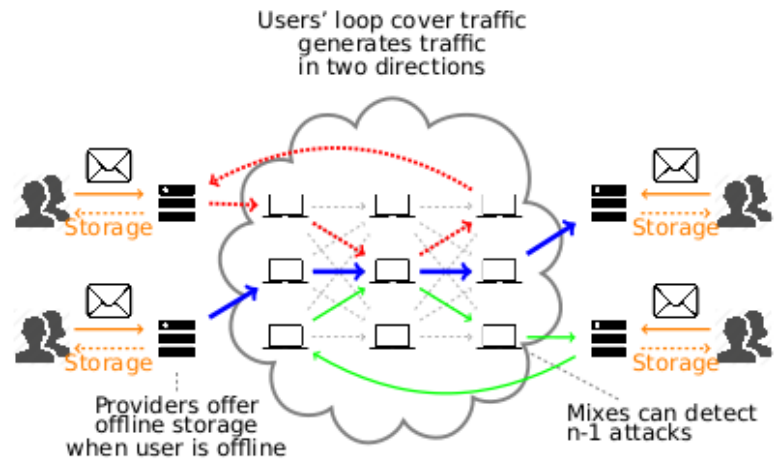
PIR is not about anonymity (necessarily), but “other details of the transaction”

Anonymous authentication protocols

- Selective disclosure credentials

  “The new certificates function in much the same way as cash, stamps, cinema tickets, subway tokens, and so on: anyone can establish the validity of these certificates and the data they overtly specify, but no more than just that. A “demographic” certificate, for instance, can specify its holder’s age, income, marital status, and residence, all digitally tied together in an unforgeable manner.” (Brand,xix)

In anonymous authentication protocols [4,6], the user first obtains a credential from an issuer (e.g., the government) certifying a set of attributes. Later, the user is able to selectively prove properties on these attributes to a verifying party (e.g., a vendor). The main property of these protocols is that a statement on the attributes can be proven without revealing any additional information besides the statement itself.



Anonymous communication networks, including mixnets, and low-latency networks like tor.

This is a new mixnet called Loopix, which a student is studying with me at the moment

Private Information Retrieval (PIR)

- Access database records, but don’t reveal to the database server which ones
- Simplest case?

Private Information Retrieval (PIR)

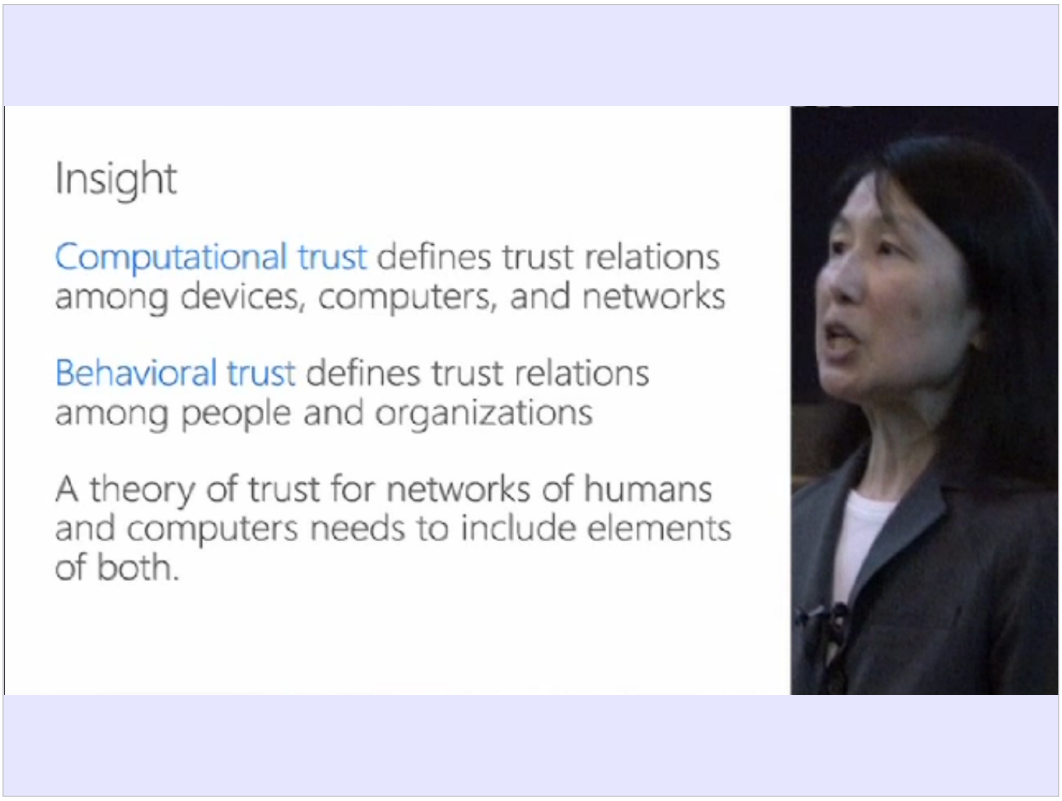
- Access database records, but don’t reveal to the database server which ones
- Simplest case?
  - (Hint: it requires transmitting a lot of data)

Private Information Retrieval (PIR)

- Access database records, but don’t reveal to the database server which ones
- Simplest case?
  - (Hint: it requires transmitting a lot of data)
  - (Hint: it requires transmitting the max possible amount of data for that database)

Recap

- Privacy as control: a matter of policy, which **controls data use**. Does not necessarily try to minimize trust in a third party; may try to provide evidence of trustworthiness of trusted systems
- Privacy as confidentiality: **minimizes disclosure**. Tries to minimize trust in third parties



- A whole ‘nother view: Diaz and Gürses on “privacy as practice”
- Be sure you trust the trustworthy as much as possible (note difference from security focus on avoiding trusting the untrustworthy)
- “socio-technical”
- transparency
- “as much as possible”: might mean mitigating the case where you’re wrong


D.C.'09

gpg

- Ways to generate keys: full generate
- expert mode
- edit-key
- `sign’

the openssl command

- One nice way to view certificate information
- man openssl
- man x509
- verifying the fingerprint for a self-signed TLS certificate

openssl x509 -fingerprint -in ./cert.pem -noout

you can get much more information about certificates with openssl x509 ... . Worth investigating a bit.

gnutls

- includes an interactive commandline client, gnutls-cli
- Can redirect stdin to use it one-shot
- Can redirect stdout to save

You can use this to view certificates, which are pem files