The Signal in the noise

HEART Computer Security and Privacy for the Modern World

💙 `EN.600.111(33)`

October 14, 2021

Tushar Jois

The Signal in the noise

HEART Computer Security and Privacy for the Modern World

🧡 `EN.600.111(26)`

October 20, 2021

Tushar Jois

Recap

- You can describe privacy as contextual integrity over information flows
- Onion routing in Tor builds circuits to provide censorship resistance
- Steganography protects both the content and presence of a message

Exit ticket from last time

- Trusting Tor
  - Tor is another system we have to trust
  - Funded by lots of people (incl. US) but mostly written by volunteers (open source)
- Virtual Private Networks and Tor
  - VPNs are similar to Tor (having another computer request traffic for you)
  - No guarantees that a VPN will not read/store/log your actions
    - VPNs claim terms of use, audits, etc but no formal promises
    - Tor has cryptographic guarantees (encrypted traffic)
- Exit nodes on Tor
  - Exit node needs to see your data to perform a web request
  - Can potentially break your privacy, but also can use TLS
- Generative models
  - Tokens are generated based off the model's prior training
    - Usually on a corpus of text
  - Both the sender and receiver need to have the same exact model for this to work
    - Online ML repositories (ModelZoo, etc)

Reminder about presentations

- End-of-class presentation on a security topic of your choosing (groups of 3)
- Ignite format
  - Every group member has 20 slides
  - Slides that automatically advance every 15 seconds
  - 5 minutes per person → 15 minutes total
  - Google Slides template to follow
- Important dates (email me to submit)
  - Topic/group selection
    - 🧡 **Wed Oct 20,** 💙 **Thu Oct 14**
  - Presentation abstract (1 paragraph)
    - 🧡 Wed Nov 3, 💙 Thu Oct 28
  - Presentations (with slides)
    - 🧡 Wed Nov 17, 💙 Thu Nov 11

Topics:

- ~~Speculative execution (Spectre) attacks~~
- Perceptual hashing (NeuralHash, PhotoDNA)
- ~~Privacy in machine learning~~
- ~~Supply chain vulnerabilities (SolarWinds) and solutions~~
- ~~Medical device security~~
- ~~Web security: SQL injection and XSS~~
- Research topic/student choice (must confirm)

First come first served!

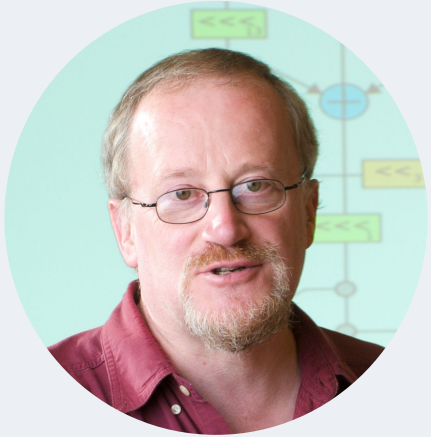
Recap

- You can describe privacy as contextual integrity over information flows
- Onion routing in Tor builds circuits to provide censorship resistance
- Steganography protects both the content and presence of a message

Lesson objectives

- Apply the security engineering framework to systems design
- Work through, step by step, the operation of the Signal protocol
- Understand the political and societal ramifications of secure messaging

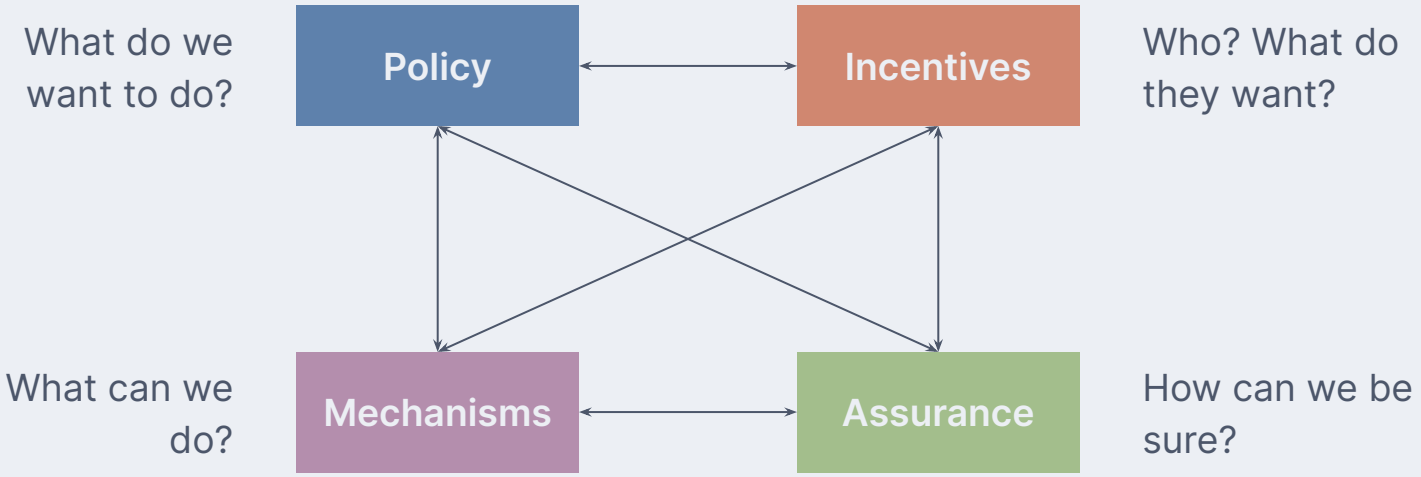
Security engineering



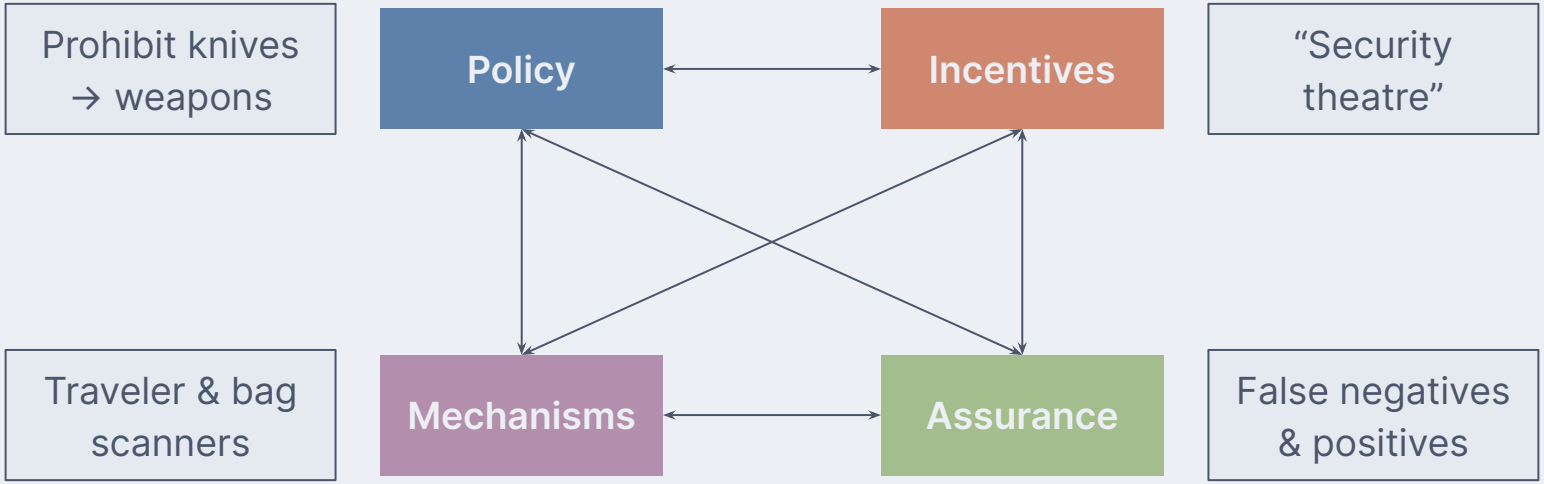
Ross Anderson

Professor, University of Cambridge

Security engineering



Airport security



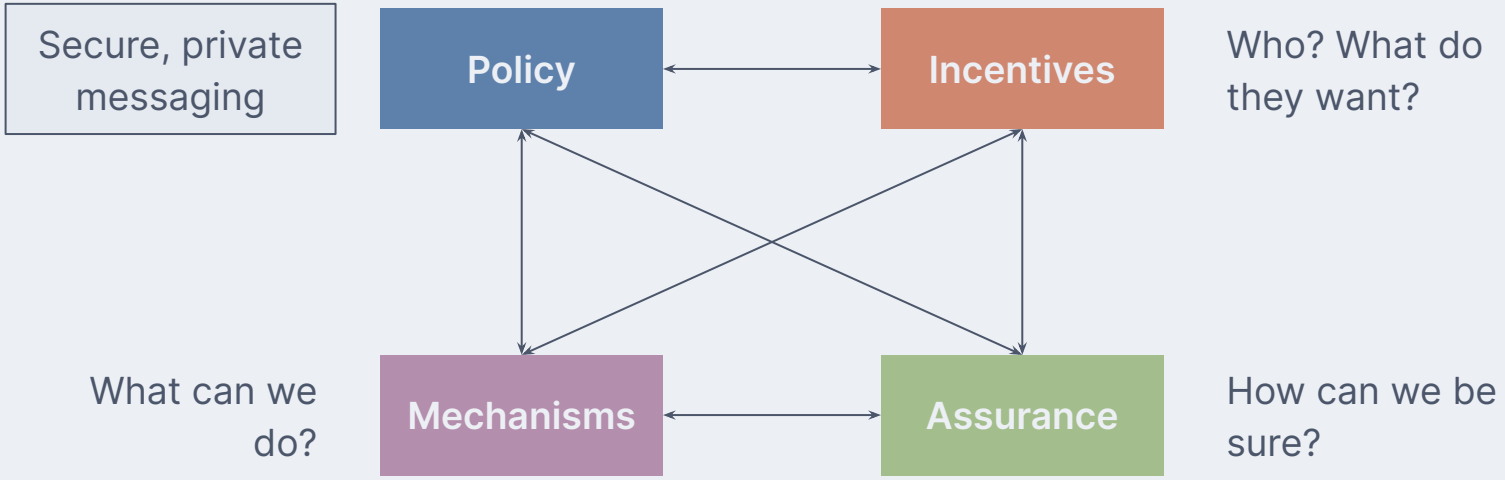
The importance of secure messaging

- Facebook Messenger, Instagram are not “end-to-end”
  - Facebook reads the messages, delivers ads about them
  - Governments can subpoena Facebook for your messages, reconstruct your digital life
- “Surveillance capitalism”
  - The person is the product
  - “Free” services provided by Big Tech powered by the selling of your data
- Data sharing agreements
  - Seen ads for things you’ve talked about on Amazon?

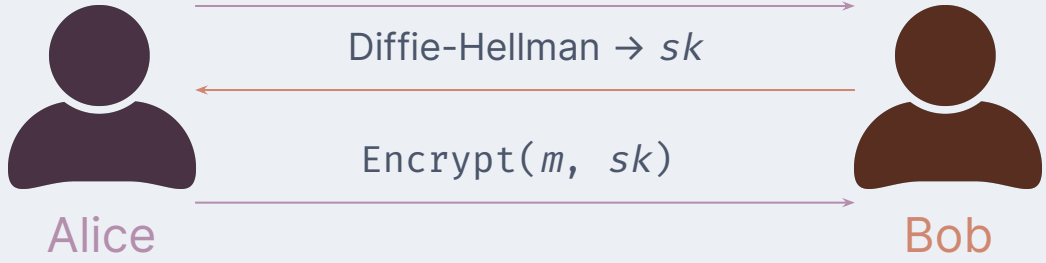
“But I have nothing to hide!”

- Solidarity with those who do
  - Snowden/whistleblowers, but also “feeling of being watched” people
- You might not realize how much data is out there
  - “We kill people based on metadata”
- Data lasts forever, and you might have to someday
  - Data lasts *forever* -- and companies/banks/governments are looking

Activity

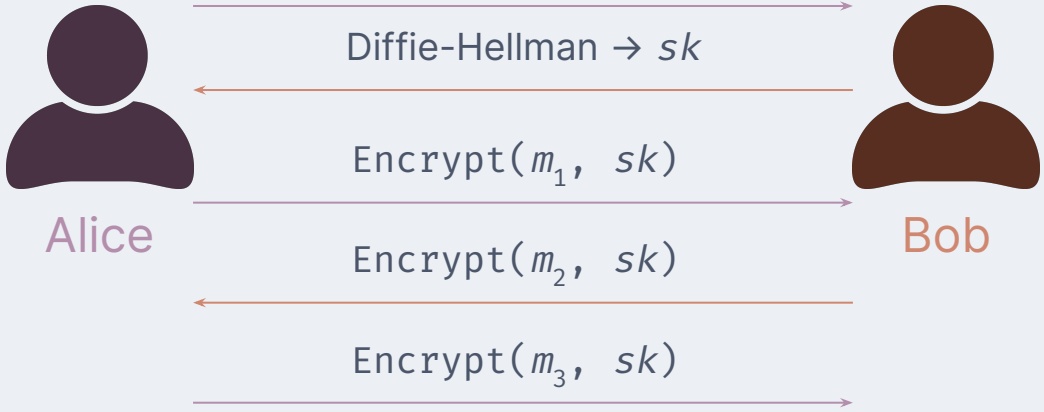

Secure, private messaging
Policy
Incentives
Who? What do they want?
What can we do?
Mechanisms
Assurance
How can we be sure?

Attempt 0



More than one message?

Attempt 1



Key compromise?

- If Alice loses *sk*, the entire message history is disclosed
  - Phone loss
  - Forensic extraction
- Can we do better?

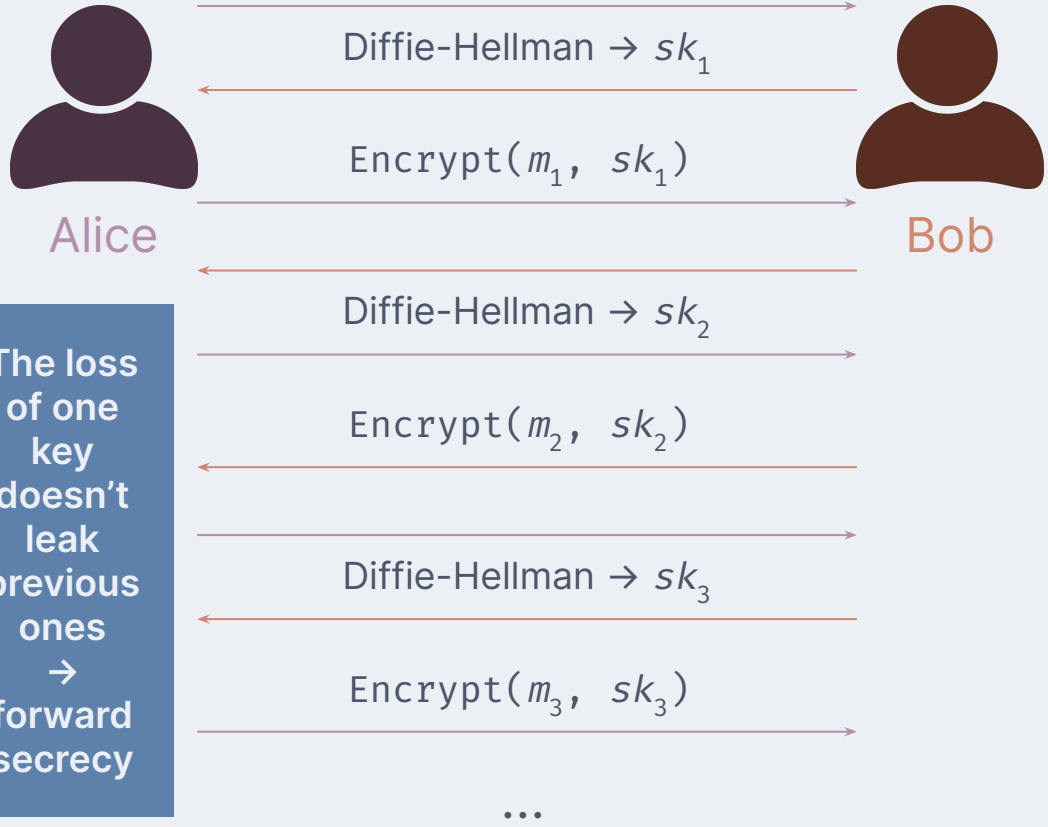
minimizing trust

Forward secrecy

“By giving certain private keys a short cryptoperiod and erasing them after they expire, it is possible to overcome the risk of recovery of derived keys due to the compromise of parties’ cryptographic states.”

IEEE 1363-2000, Standard Specifications for Public-Key Cryptography

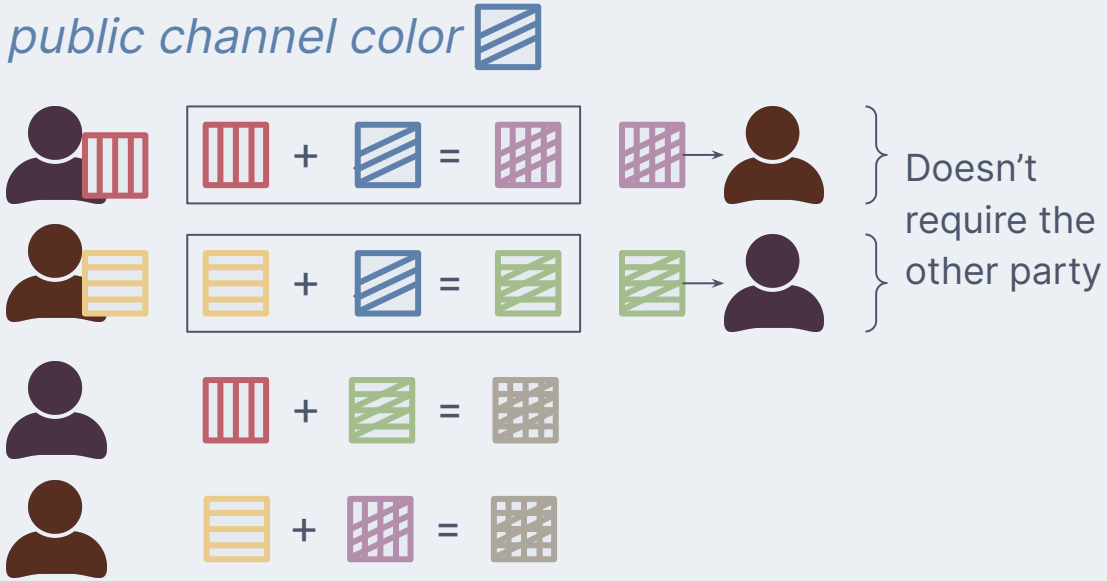
Attempt 2



The loss of one key doesn't leak previous ones → forward secrecy

What if a party is offline?

Requires interaction


public channel color
Doesn’t require the other party

Attempt 3

Alice

½ of Diffie-Hellman → $preA_1$

½ of Diffie-Hellman → $preA_2$

...

$preB_1$ → Diffie-Hellman → sk_1

Encrypt(m_1, sk_1) → c_1

(signed)

Server

½ of Diffie-Hellman → $preB_1$

½ of Diffie-Hellman → $preB_2$

...

c_1

Rest of Diffie-Hellman → sk_1

Bob

Alice

½ of Diffie-Hellman → $preA_1$

½ of Diffie-Hellman → $preA_2$

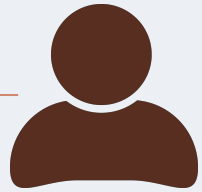
...

Server

½ of Diffie-Hellman → $preB_1$

½ of Diffie-Hellman → $preB_2$

...



Bob

$preB_1$ → Diffie-Hellman → sk_1

Encrypt(m_1, sk_1) → c_1

c_1

Rest of Diffie-Hellman → sk_1

$preA_1$ → Diffie-Hellman → sk_2

Encrypt(m_2, sk_2) → c_2

c_2

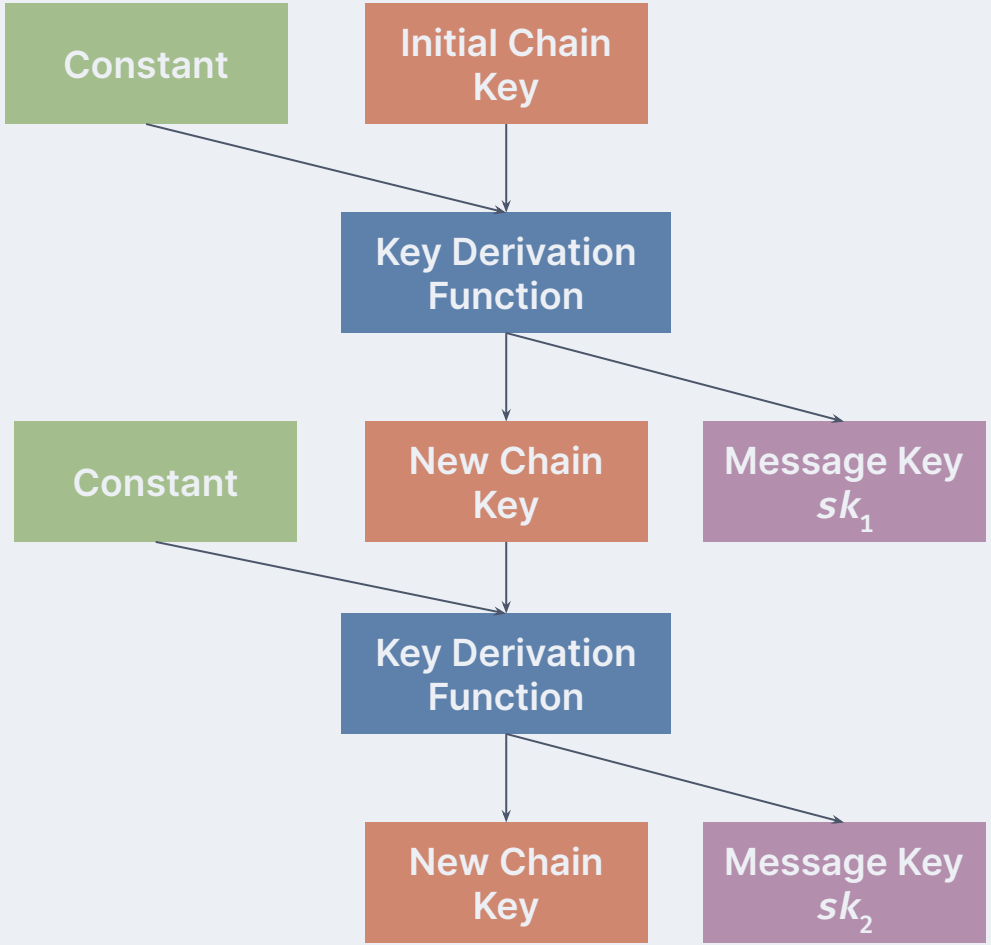
Rest of Diffie-Hellman → sk_2

What if they want to talk a lot or with other people?

Run out of pre-keys quickly

KDF chain

- Special cryptographic construct that generates new keys from old keys
  - We can use the new keys for subsequent messages
  - Requires both parties to be in the same “state” of the ratchet
- Send a message, Alice encrypts with a key, and then “ratchets it forward”
  - Bob receives the message, decrypts it, and then “ratchets it forward”
  - Forward secrecy without significant interaction
  - Both have to keep in sync



minimizing trust

Symmetric ratcheting

“The parties derive new keys for every Double Ratchet message so that earlier keys cannot be calculated from later ones… [giving] some protection to earlier or later encrypted messages in case of a compromise of a party's keys.”

Perrin and Marlinspike, “The Double Ratchet Algorithm” (2016)

Signal protocol*

*abridged

Alice

(signed)

Server

Bob

½ of Diffie-Hellman → $preA_1$

½ of Diffie-Hellman → $preA_2$

...

$preB_1$ → Diffie-Hellman → sk_1

Encrypt(m_1, sk_1) → c_1

½ of Diffie-Hellman → $preB_1$

½ of Diffie-Hellman → $preB_2$

...

c_1

Rest of Diffie-Hellman → sk_1

Ratchet forward sk_1 → sk_2

Ratchet forward sk_1 → sk_2

Alice

½ of Diffie-Hellman → $preA_1$

½ of Diffie-Hellman → $preA_2$

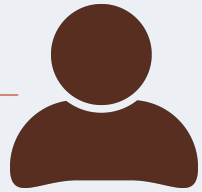
...

Server

½ of Diffie-Hellman → $preB_1$

½ of Diffie-Hellman → $preB_2$

...



Bob

$preB_1$ → Diffie-Hellman → sk_1

Encrypt(m_1, sk_1) → c_1

c_1

Rest of Diffie-Hellman → sk_1

Ratchet forward sk_1 → sk_2

Ratchet forward sk_1 → sk_2

c_2

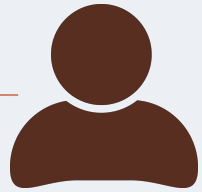
Encrypt(m_2, sk_2) → c_2

Ratchet forward sk_2 → sk_3

Ratchet forward sk_2 → sk_3

The loss of one key doesn't leak previous ones → forward secrecy

Server



Bob

f Diffie-Hellman → $preA_1$

½ of Diffie-Hellman → $preB_1$

f Diffie-Hellman → $preA_2$

½ of Diffie-Hellman → $preB_2$

...

...

B_1 → Diffie-Hellman → sk_1

ncrypt(m_1, sk_1) → c_1

c_1

Rest of Diffie-Hellman → sk_1

tchet forward sk_1 → sk_2

Ratchet forward sk_1 → sk_2

c_2

Encrypt(m_2, sk_2) → c_2

tchet forward sk_2 → sk_3

Ratchet forward sk_2 → sk_3

Sidebar

- Not a trivial protocol
- Complicated to provide forward secrecy, limited interaction, and efficiency
- Good example of security engineering in practice
- Needs to be usable in practice

Activity II

- Download and install the Signal app
  - https://signal.org/install
- Play around with the app
  - Send some messages!
- Add your group project members on Signal
  - Try to use Signal to coordinate for the project!
- Questions to think about
  - What features did you see in Signal?
  - How was using Signal different?
  - Was the cryptography involved evident?

Society

- Signal banned in several countries
  - “Censorship circumvention” -- but brittle
- “Going Dark”
  - FBI’s initi
    end-to-e
- EARN IT Act
  - Providers
    encrypte
    message
  - Defeats e
    name of detecting abuse
  - Horrible, abusive content -- but universal scanning might not be the answer
- The debate rages on

> “a transparent and deeply cynical effort by a few well-connected corporations and the Trump administration to use child sexual abuse to their political advantage, the impact to free speech and the security and privacy of every single American be damned”
>
> *Senator Ron Wyden (D-OR) on the EARN IT Act*

Looking ahead

- Your presentation topics and groups are due today! (email me)
  - *Next up:* presentation abstract (1 paragraph) 🧡 Wed Nov 3, 💙 Thu Oct 28
- We're going next use our security techniques to provide privacy
- Do the reading: "Data Security on Mobile Devices", Chapter 1
  - Only have to read Chapter 1 ("Introduction")
  - If you can, skim through the optional "SoK: Cryptographic Confidentiality of Data on Mobile Devices"
- Exit ticket before class is done

Lesson objectives

- Apply the security engineering framework to systems design
- Work through, step by step, the operation of the Signal protocol
- Understand the political and societal ramifications of secure messaging