PrivateDrop: Practical Privacy-Preserving Authentication for Apple AirDrop

TECHNISCHE UNIVERSITÄT DARMSTADT

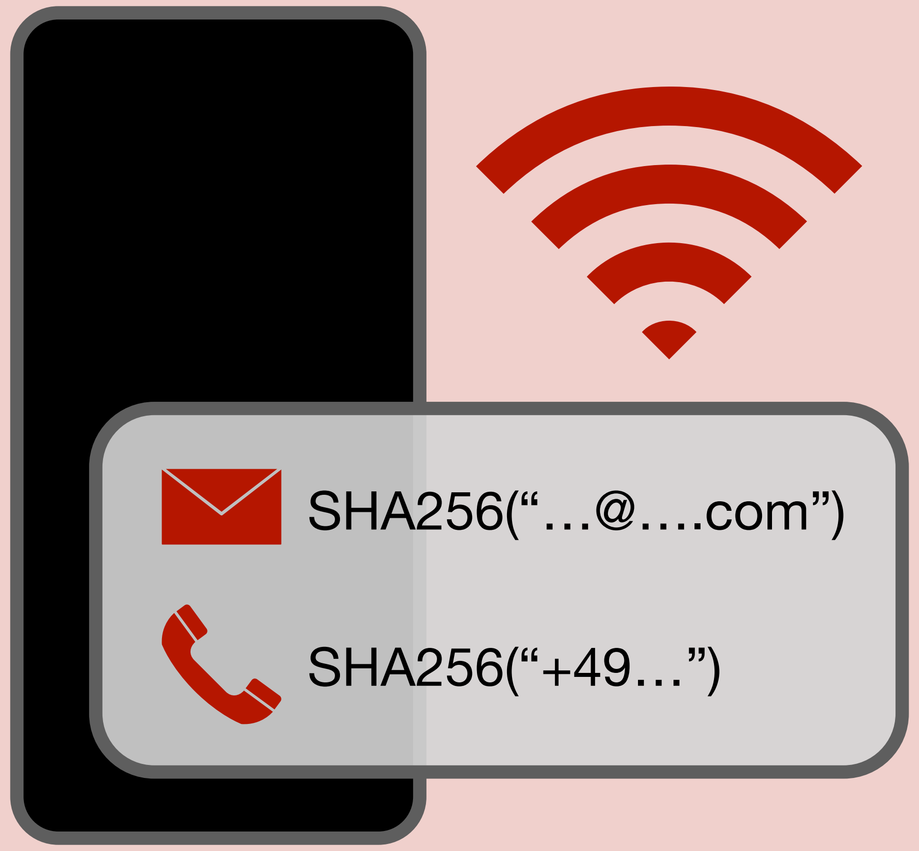
Alexander Heinrich, Matthias Hollick, Thomas Schneider,
Milan Stute, and Christian Weinert

ENCRYPTO
CRYPTOGRAPHY AND PRIVACY ENGINEERING

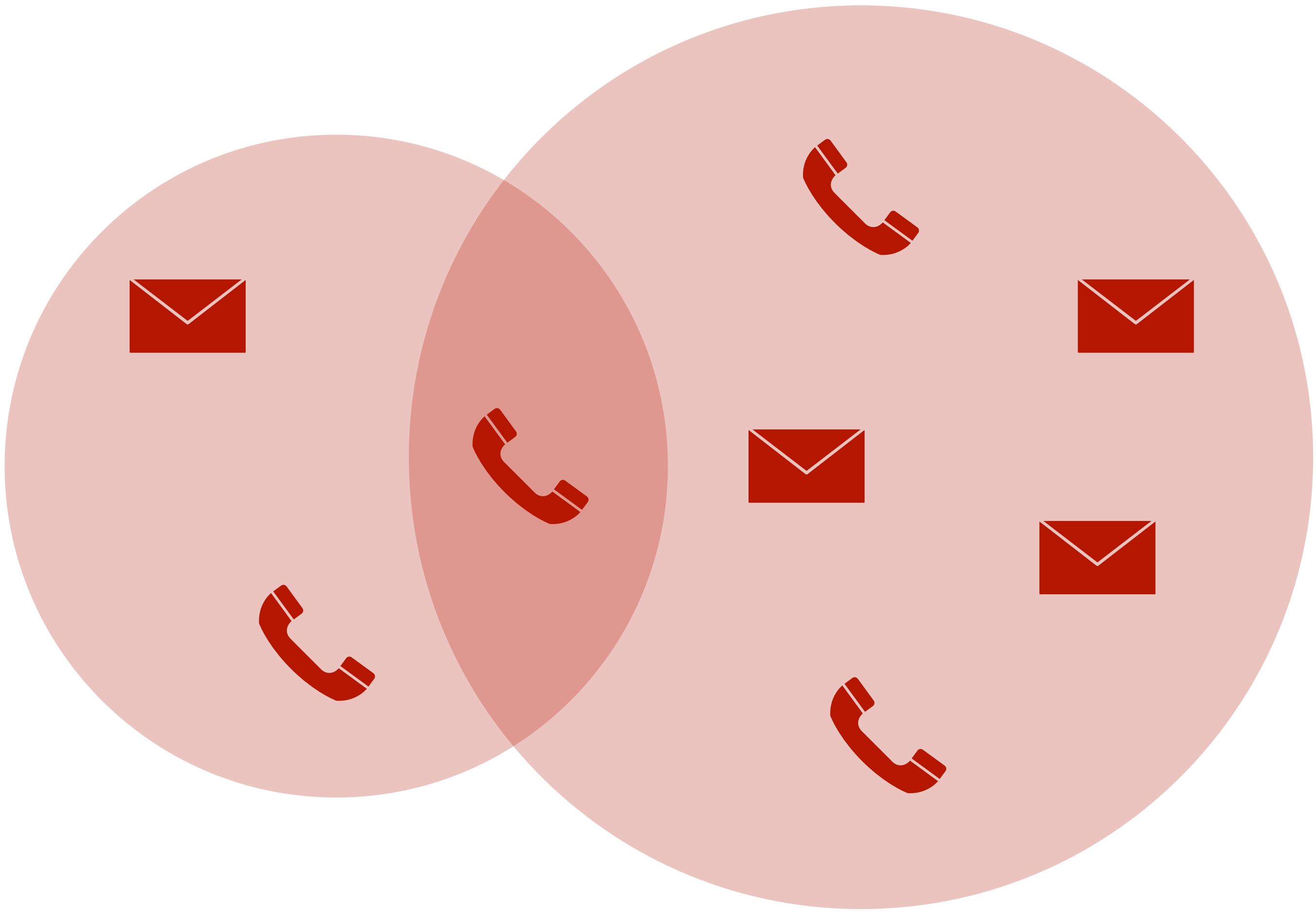
SEEMOO
SECURE MOBILE NETWORKING

privatedrop.github.io

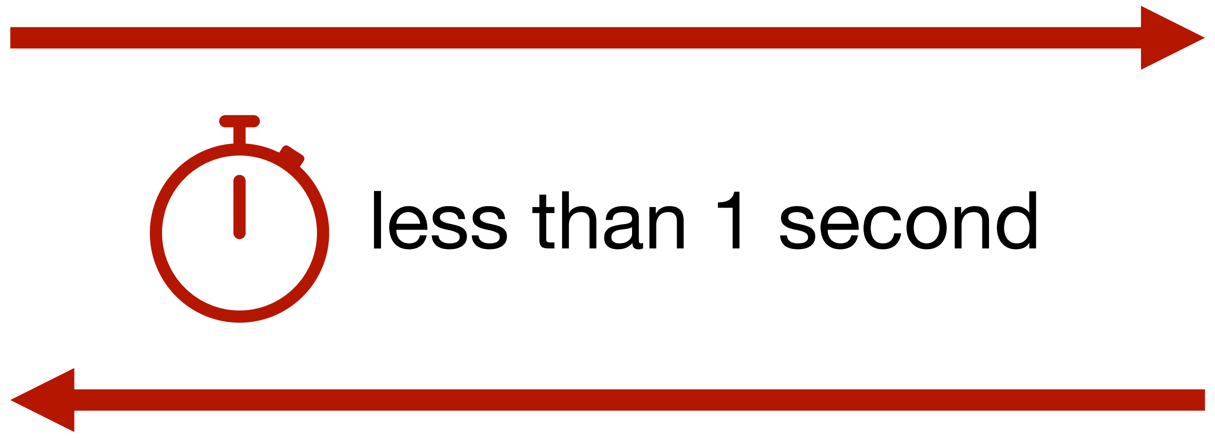
Roadmap to PrivateDrop



Discover
Contact Identifier Leakage
by Apple AirDrop



Design
Privacy-Preserving Authentication
via Private Set Intersection



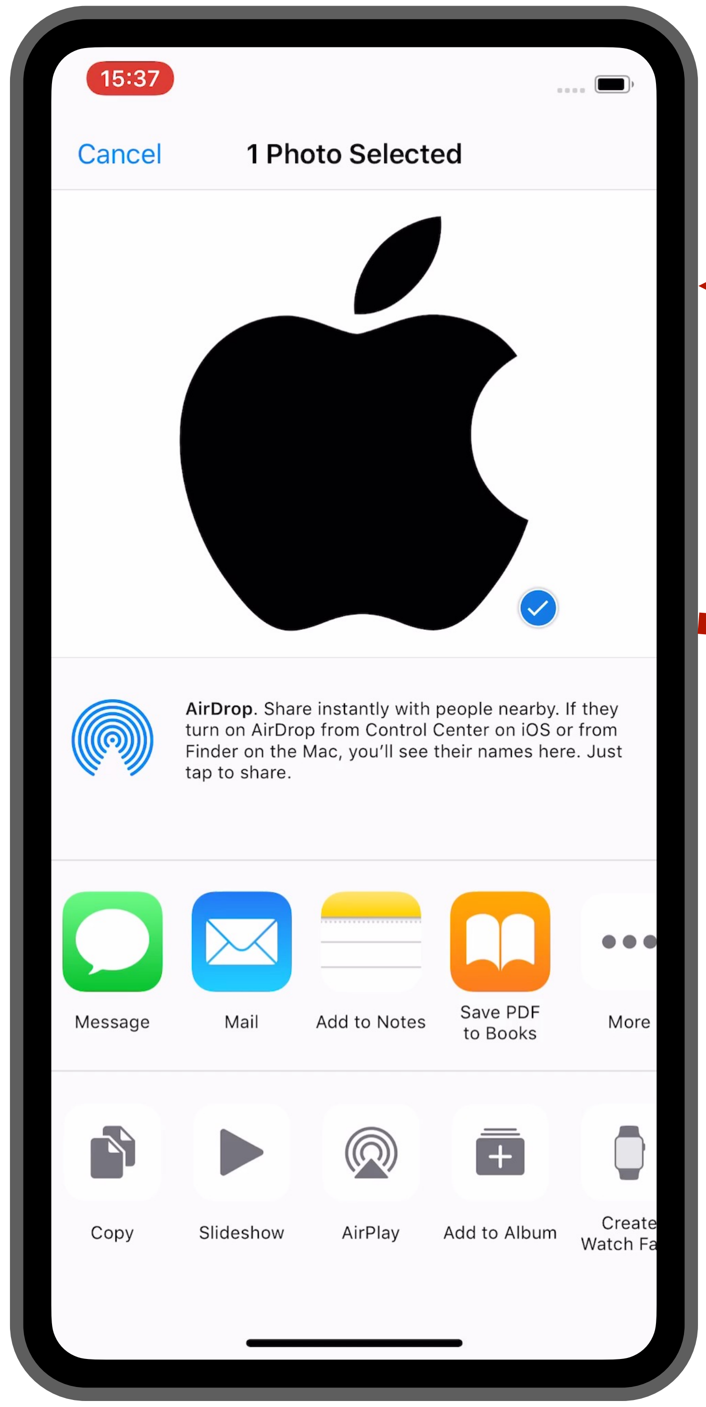
Demonstrate
Native Prototype
with Excellent User Experience

AirDrop Authentication

[SNMHKNH19]

Find out
whether we are
mutual contacts

via Wi-Fi/AWDL
[SKH18]



AirDrop. Share instantly with people nearby. If they turn on AirDrop from Control Center on iOS or from Finder on the Mac, you'll see their names here. Just tap to share.

Sender

TLS connection with

client and server certificates

HTTP POST **/Discover**

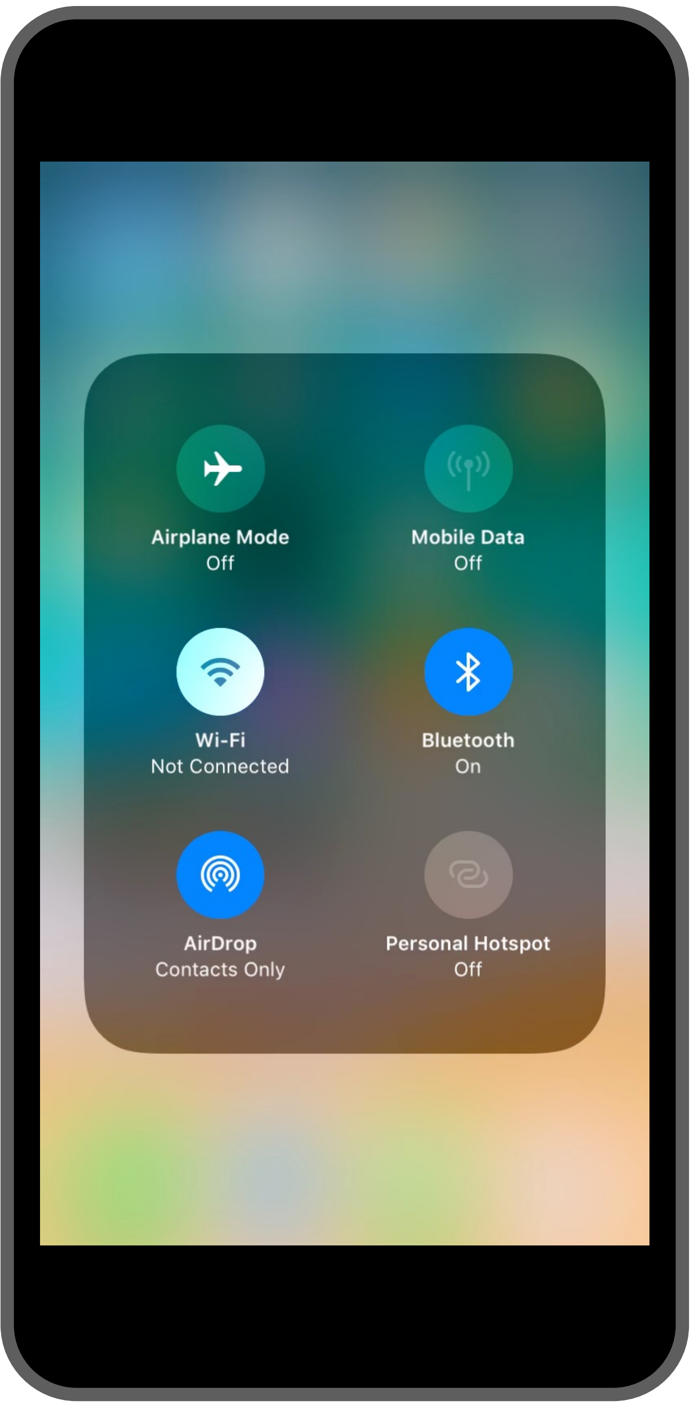
with sender's validation record*

I want to find other people, so I tell them who I am

* Apple-signed cert including

$H_i = SHA256(+49\ 123\ \ldots)$

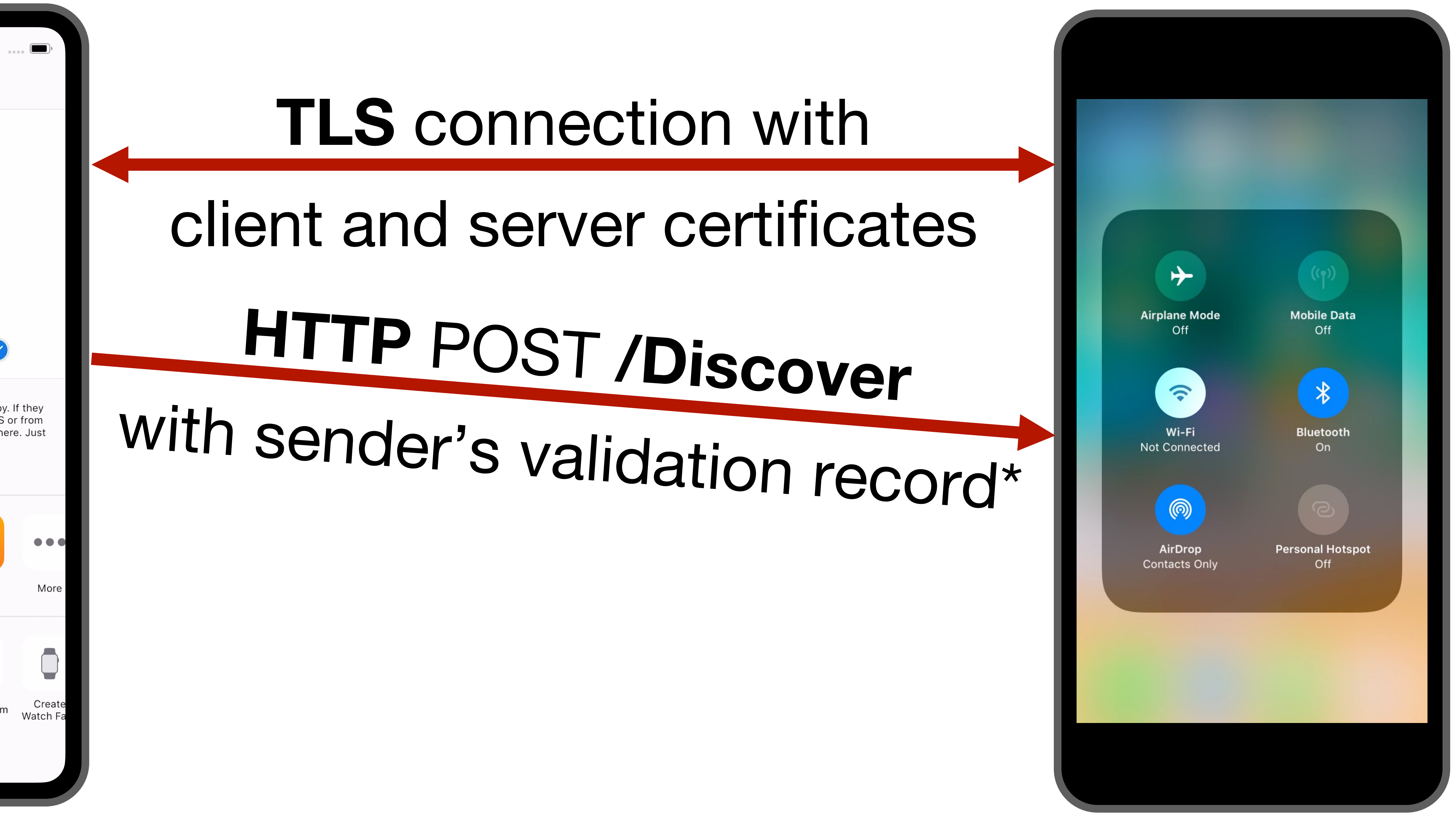
$H_j = SHA256(\ldots @icloud.com)$



Receiver

AirDrop Authentication

[SNMHKNH19]



TLS connection with client and server certificates

HTTP POST **/Discover** with sender's validation record*

Receiver

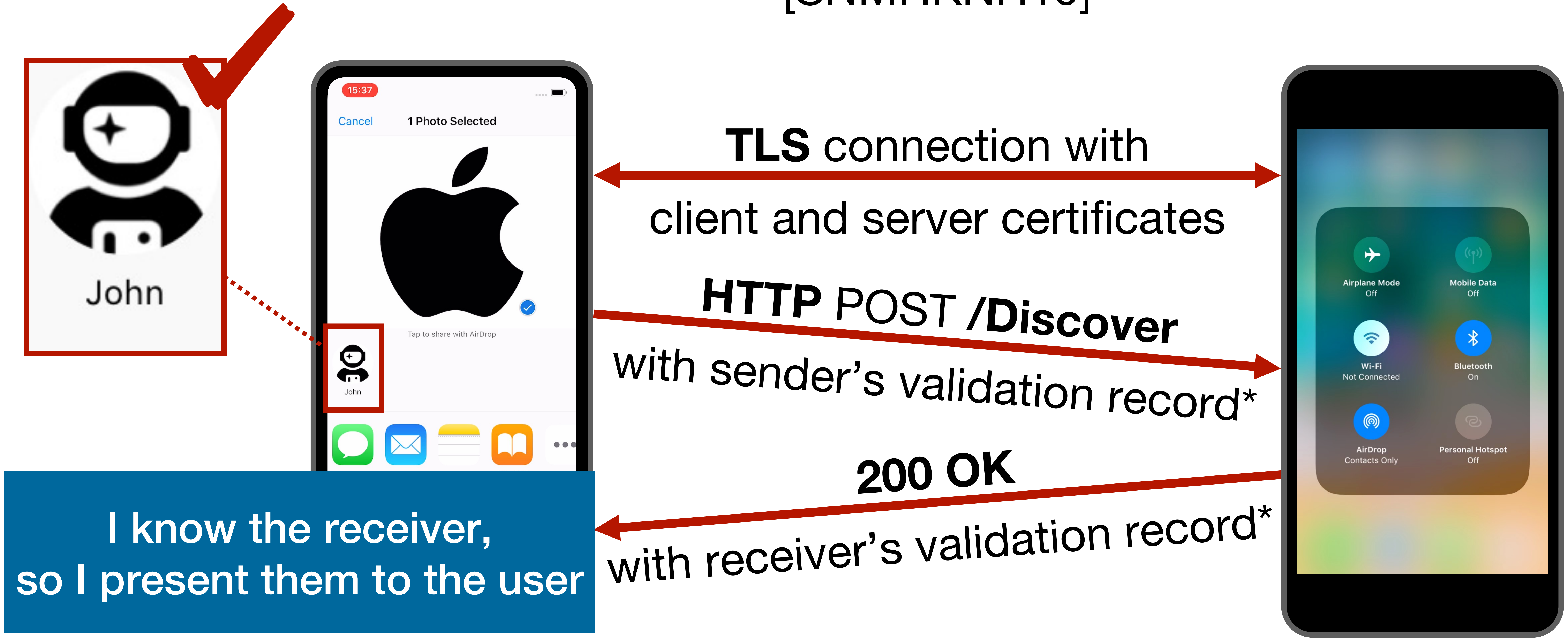
$\exists H_i \in VR: H_i \in \mathit{address\ book}$ ✓
(+ check validation record
 + check TLS certificate)

I know the sender,
so I tell them who I am

* Apple-signed cert including
$H_i = SHA256(+49\ 123\ \ldots)$
$H_j = SHA256(\ldots @\mathrm{icloud.com})$

AirDrop Authentication

[SNMHKNH19]



TLS connection with client and server certificates

HTTP POST **/Discover** with sender's validation record*

200 OK with receiver's validation record*

I know the receiver, so I present them to the user

Sender

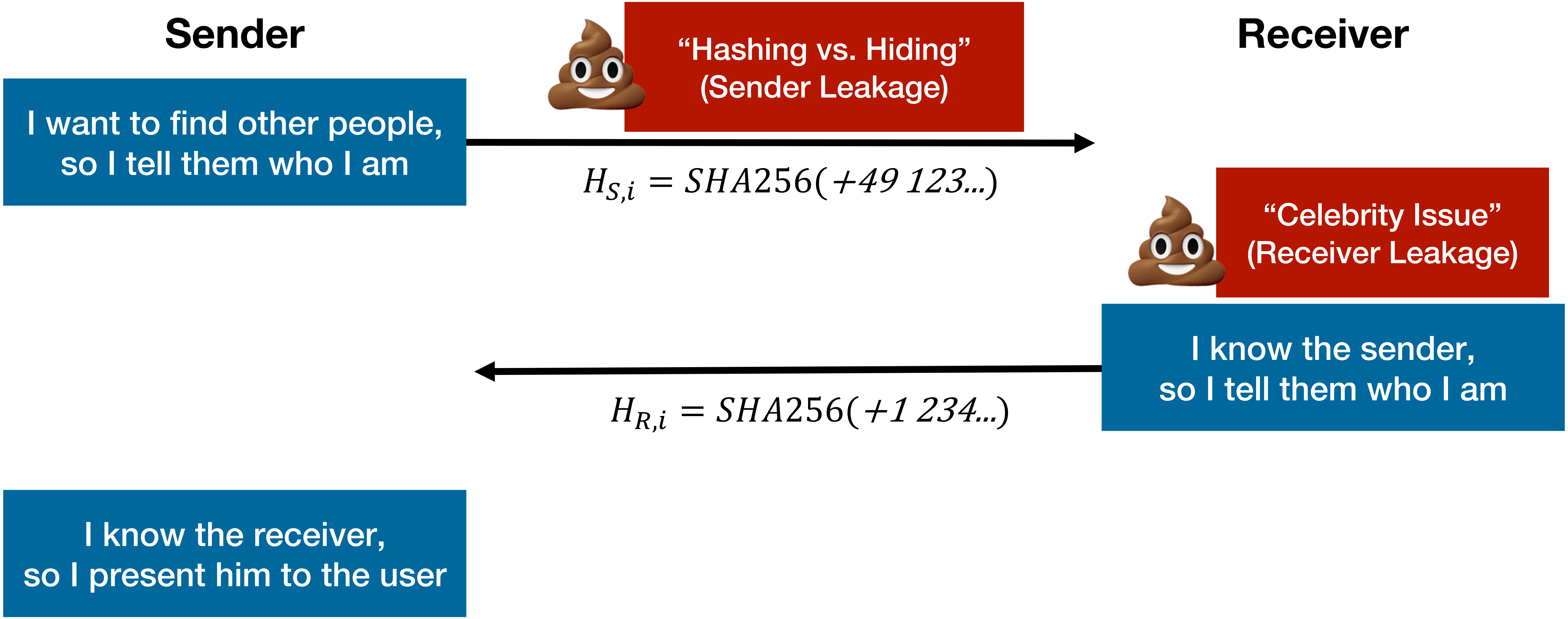
Receiver

* Apple-signed cert including

$$H_i = SHA256(+49\ 123\ ...)$$

$$H_j = SHA256(...@icloud.com)$$

AirDrop Authentication: What can go wrong here?



Sender

Receiver

I want to find other people, so I tell them who I am

“Hashing vs. Hiding” (Sender Leakage)

$H_{S,i} = SHA256(+49\ 123...)$

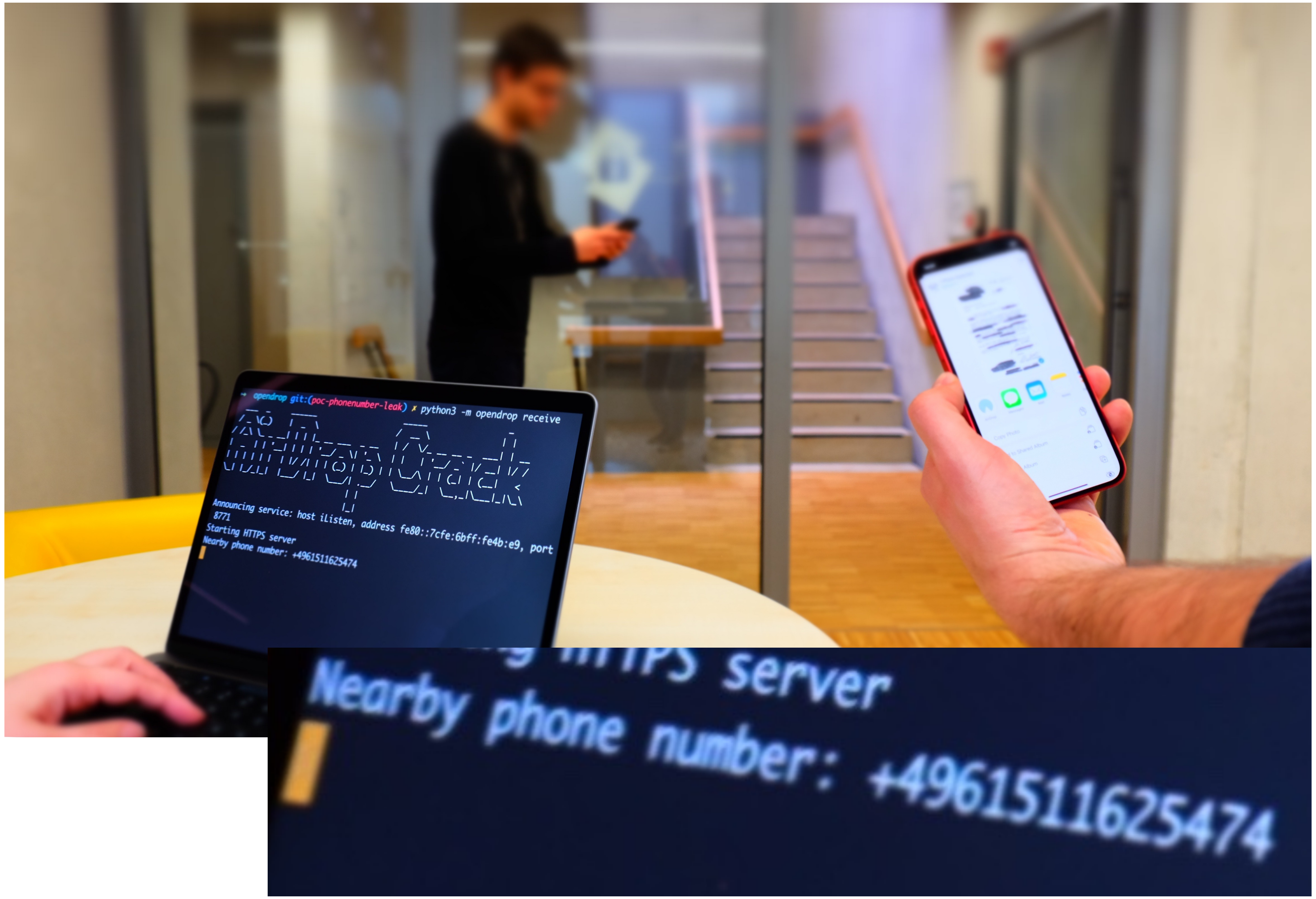
“Celebrity Issue” (Receiver Leakage)

I know the sender, so I tell them who I am

$H_{R,i} = SHA256(+1\ 234...)$

I know the receiver, so I present him to the user

Exploiting the Vulnerabilities in Practice



```
→ opendrop git:(poc-phonenumber-leak) ✗ python3 -m opendrop receive
Announcing service: host iListen, address fe80::7cfe:6bff:fe4b:e9, port
8771
Starting HTTPS server
Nearby phone number: +4961511625474
```

```
...g HTTPS server
Nearby phone number: +4961511625474
```

Requirements

- Physical proximity to target
- Wi-Fi-capable device

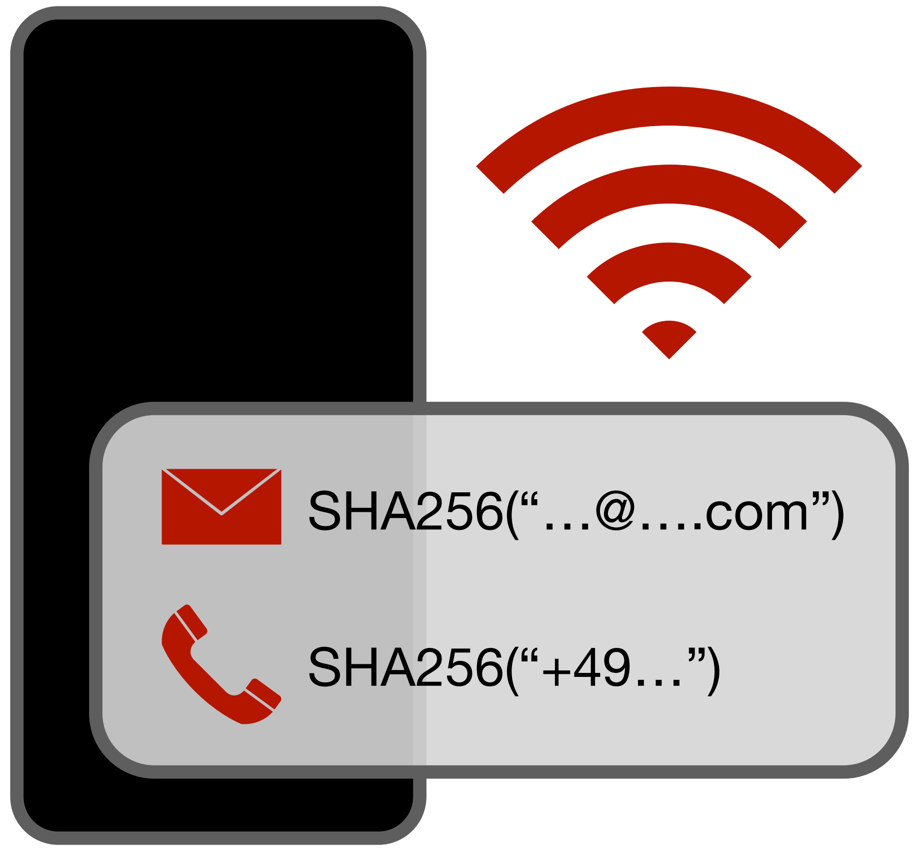
Proof-of-concept

- "AirCollect"
  https://github.com/seemoo-lab/opendrop/blob/poc-phonenumber-leak/README.PoC.md
- Makes use of optimized rainbow tables [HWSDS21]

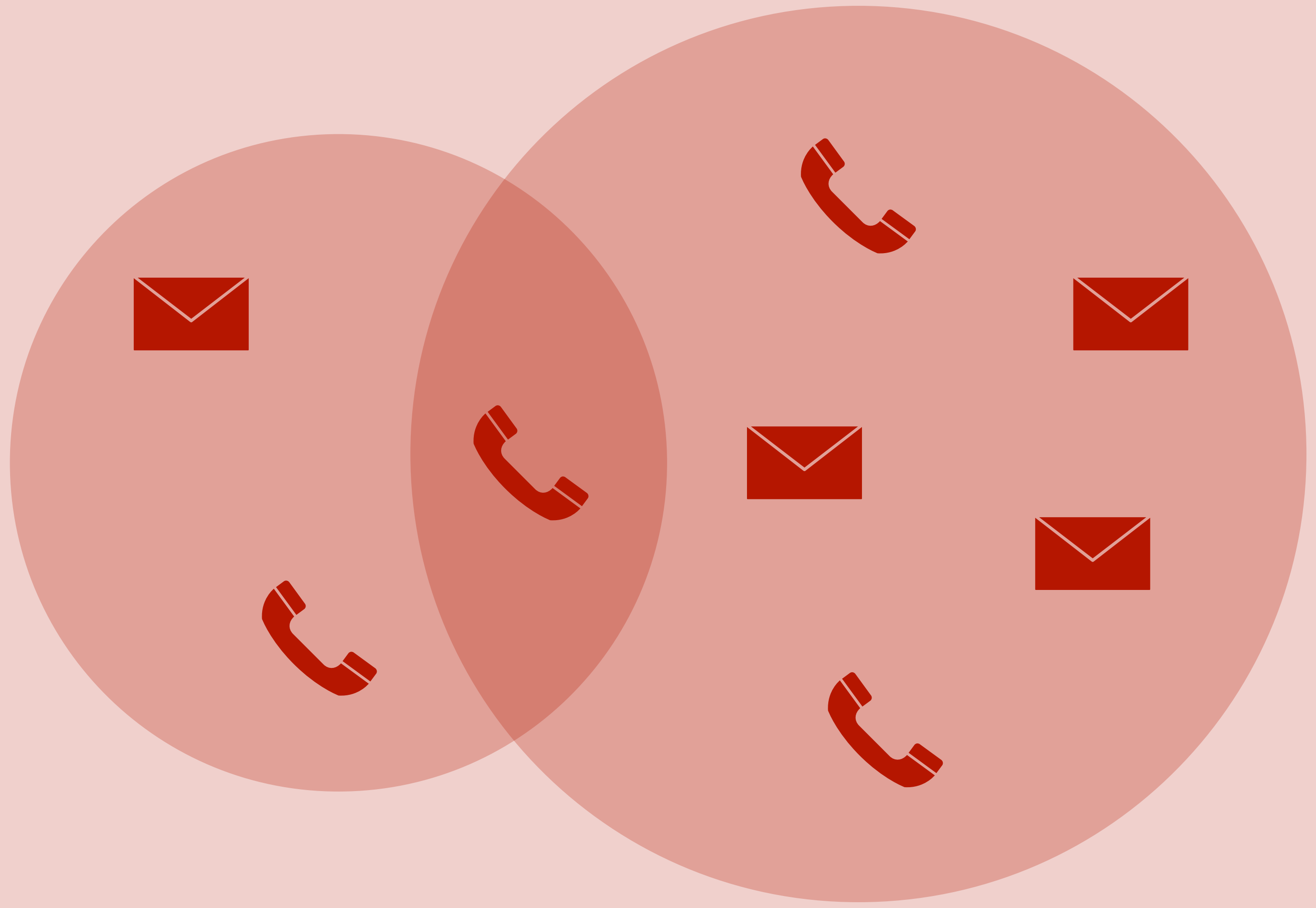
Impact

- Recover phone numbers of AirDrop users in real-time

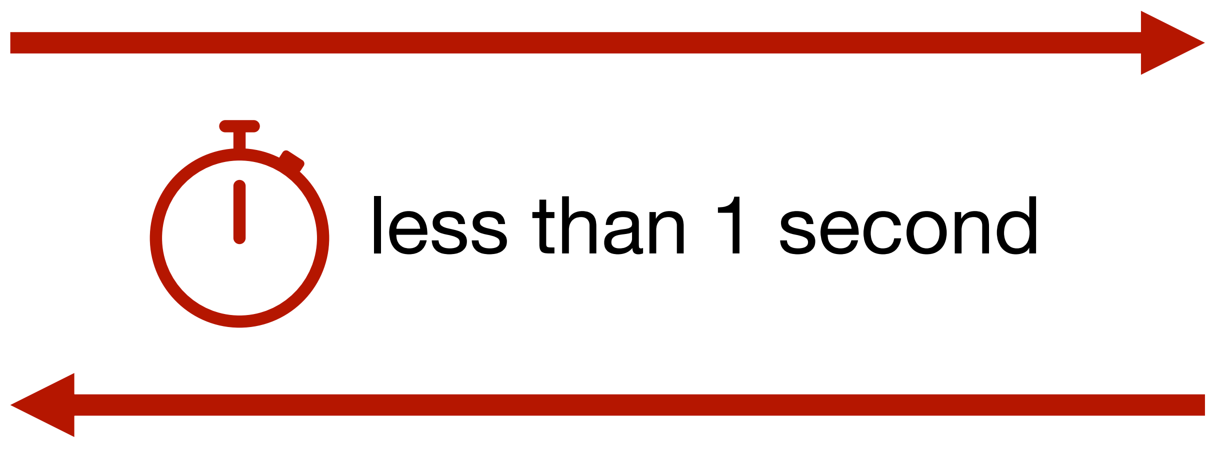
Roadmap to PrivateDrop



Discover
Contact Identifier Leakage
by Apple AirDrop



Design
Privacy-Preserving Authentication
via Private Set Intersection

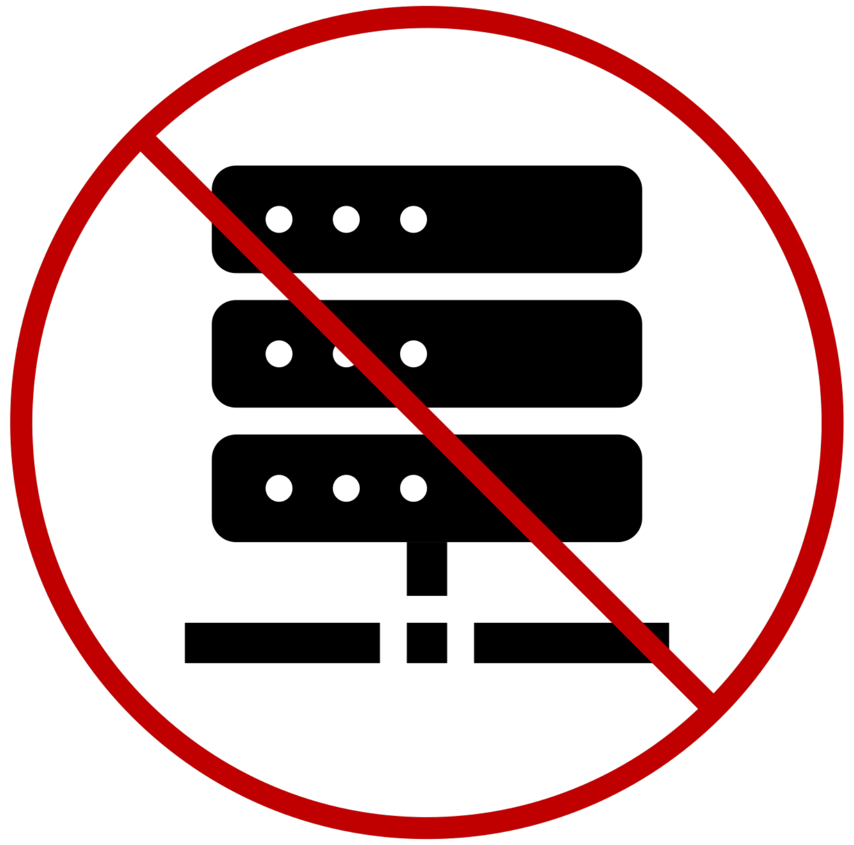


Demonstrate
Native Prototype
with Excellent User Experience

PrivateDrop Requirements

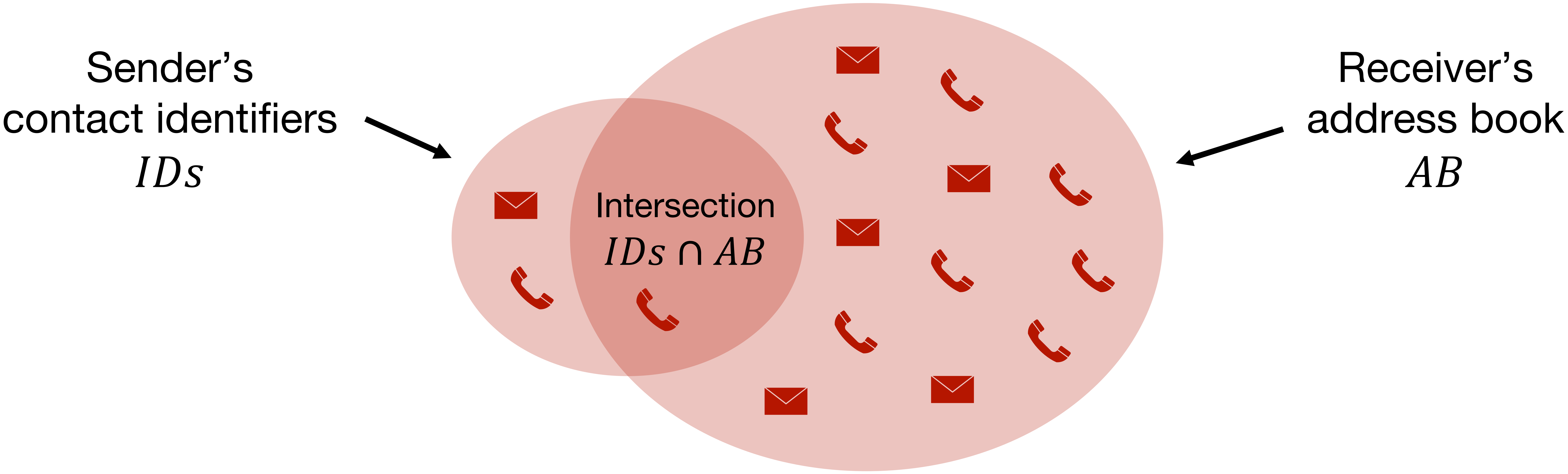
Privacy requirements:

1. Disclose contact identifiers only *if both parties are mutual contacts.*
2. Only disclose those *contact identifiers that the other party already knows.*



Apply private set intersection (PSI) to achieve private mutual authentication

Private Set Intersection (PSI)

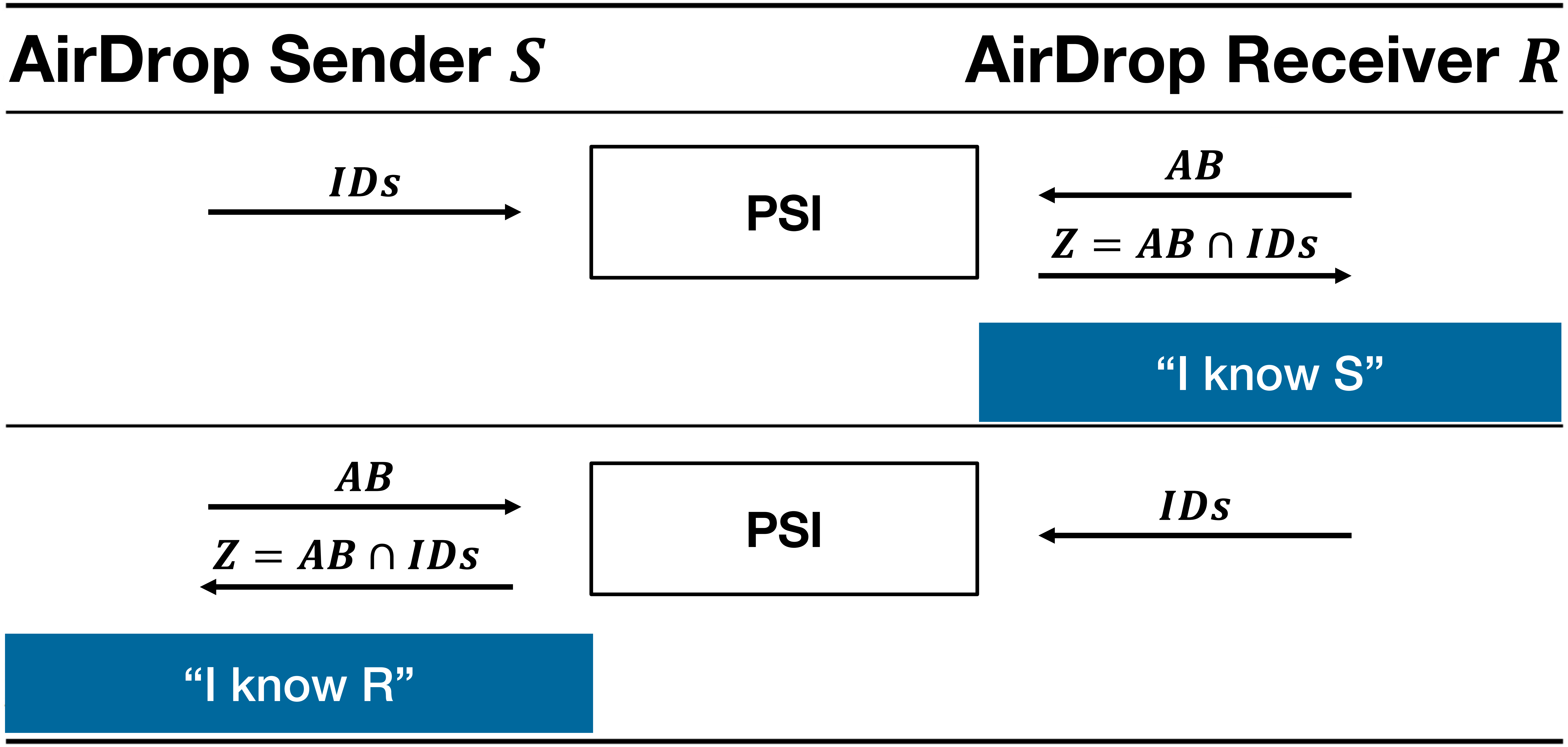


Sender's contact identifiers IDs

Receiver's address book AB

Intersection $IDs \cap AB$

AirDrop: Semantics



AirDrop Sender S | **AirDrop Receiver R**

IDs → PSI ← AB

PSI → $Z = AB \cap IDs$

"I know S"

AB → PSI ← IDs

$Z = AB \cap IDs$ ← PSI

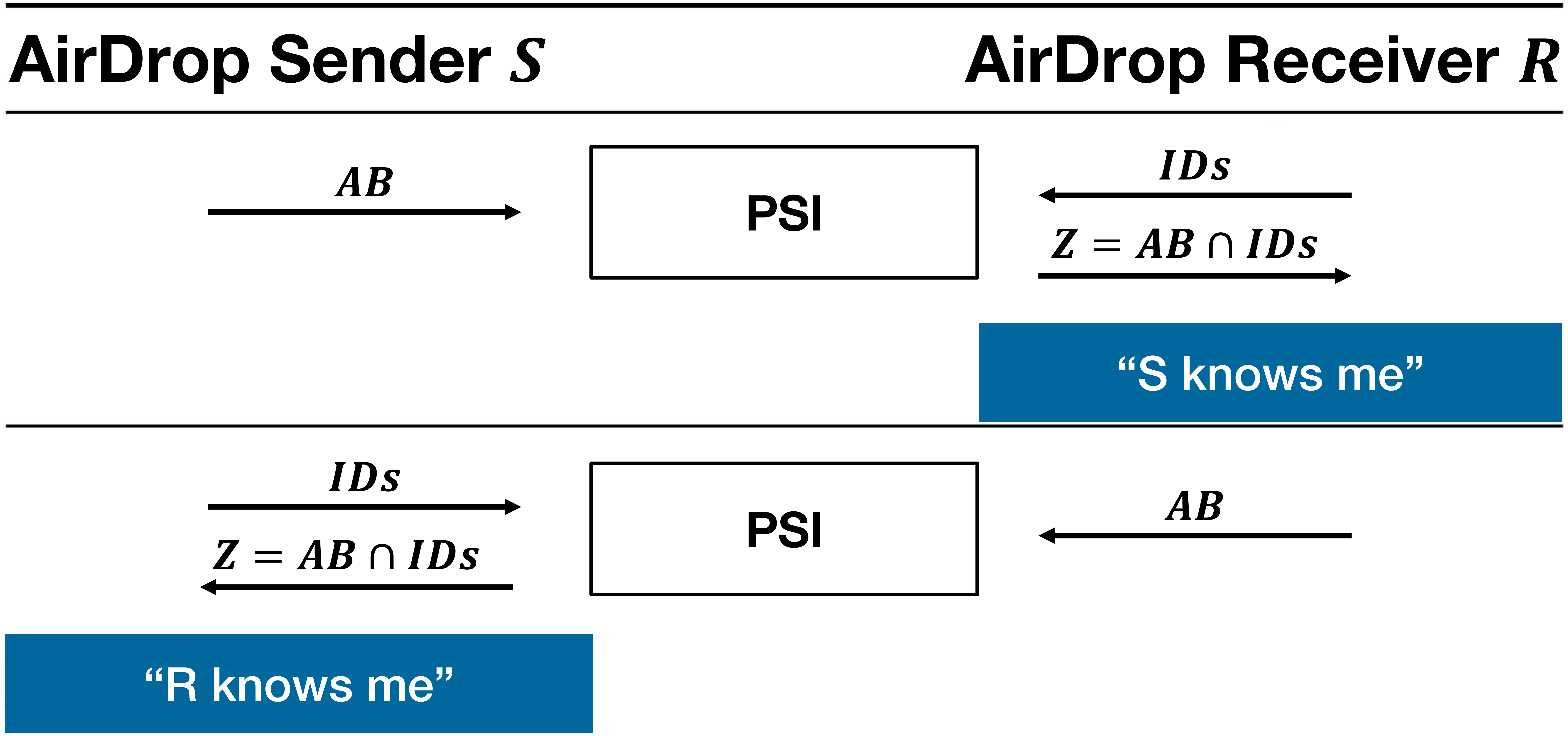
"I know R"

Problems:

- **Malicious receivers**
- **Online complexity depends on AB (large)**

AB: address book
IDs: contact identifiers

PrivateDrop: Semantics



Next: S and R can disclose their known identities, i.e., $IDs \cap AB$

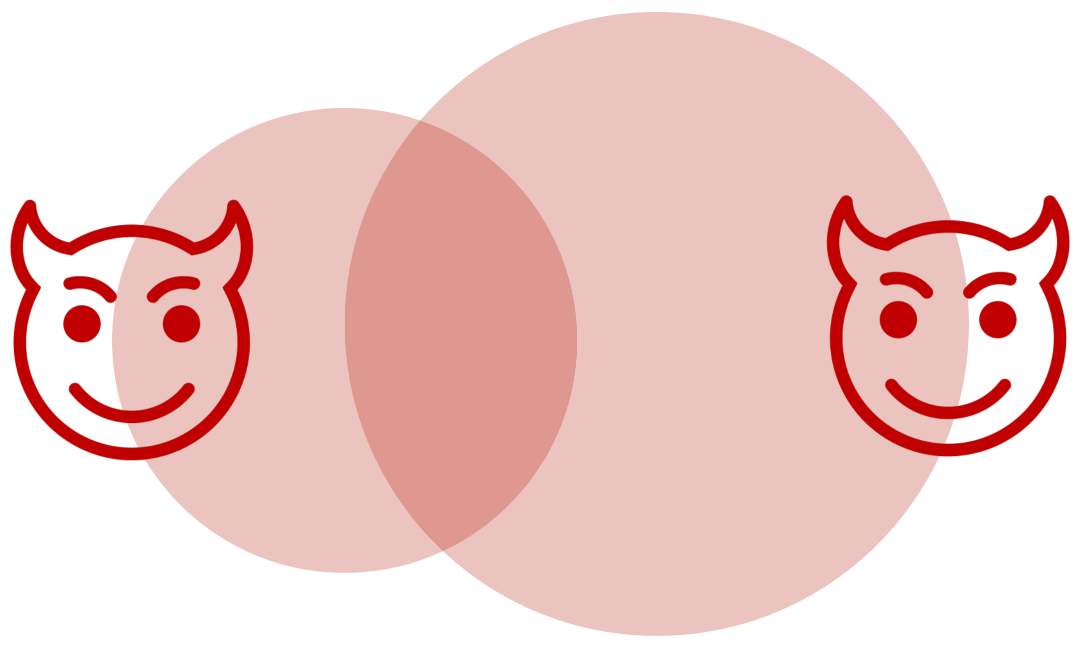
Changed Semantics:

- **Receivers in check**
- **Online complexity depends on IDs (small)**

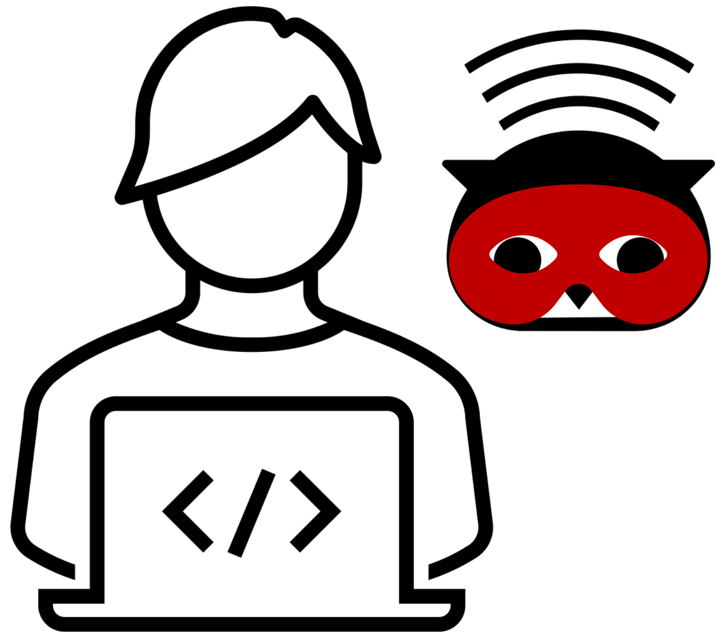
AB: address book
IDs: contact identifiers

PrivateDrop Design and Implementation

Maliciously Secure PSI Protocol

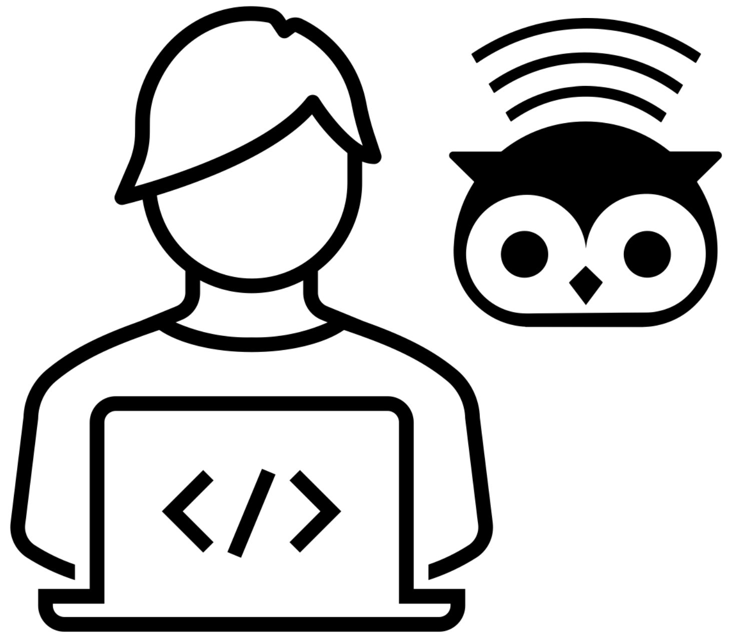


PrivateDrop Implementation

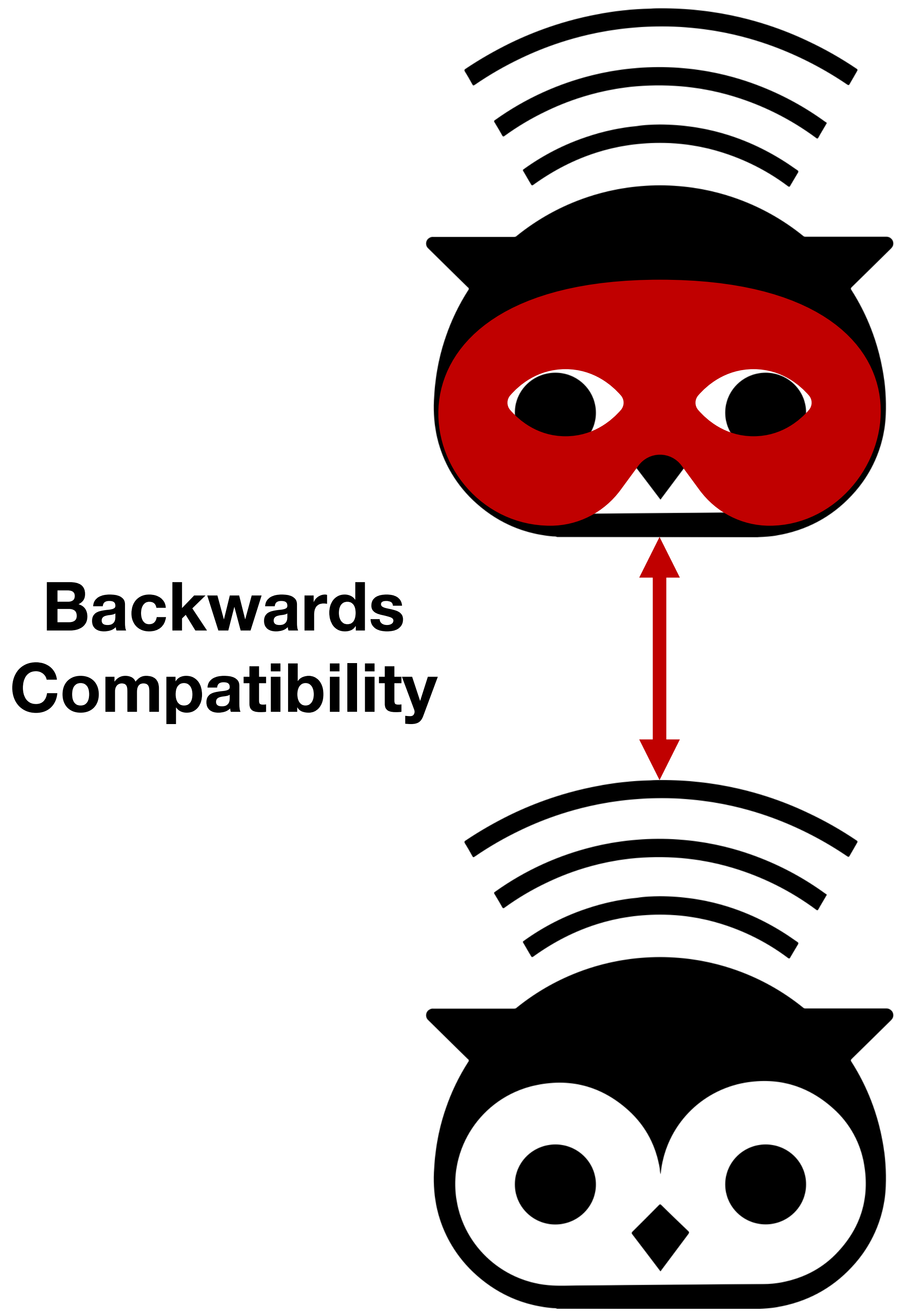


Protection against Malicious Inputs

AirDrop Implementation

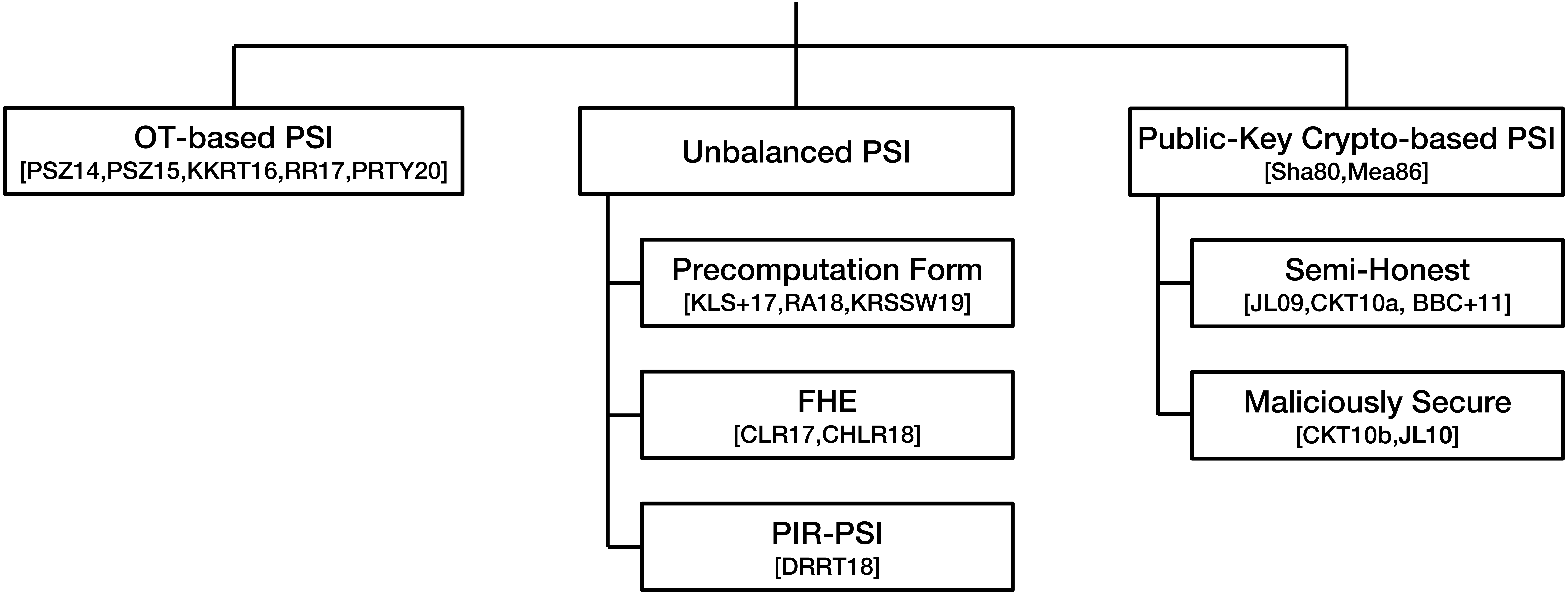


Integration of PSI into AirDrop



Backwards Compatibility

Choice of PSI Protocol



OT-based PSI
[PSZ14,PSZ15,KKRT16,RR17,PRTY20]

Unbalanced PSI

Precomputation Form
[KLS+17,RA18,KRSSW19]

FHE
[CLR17,CHLR18]

PIR-PSI
[DRRT18]

Public-Key Crypto-based PSI
[Sha80,Mea86]

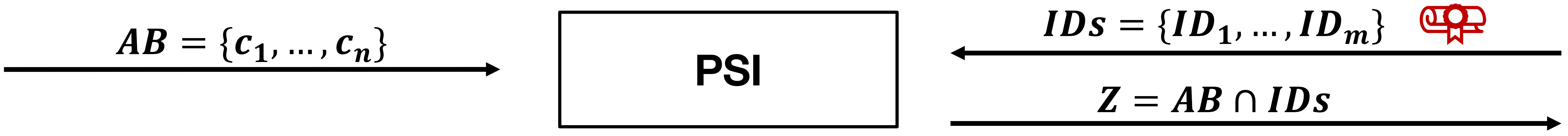
Semi-Honest
[JL09,CKT10a, BBC+11]

Maliciously Secure
[CKT10b,**JL10**]

Optimized PSI Protocol of [JL10]

AirDrop Sender S

AirDrop Receiver R



$\boldsymbol{AB = \{c_1, \dots, c_n\}}$ →

PSI

← $\boldsymbol{IDs = \{ID_1, \dots, ID_m\}}$ 📜

$\boldsymbol{Z = AB \cap IDs}$ →

Precomputation

$k \xleftarrow{\$} \mathbb{Z}_q$

For $j = 1$ to n:

$\quad u = H\left(H(c_j), H(c_j)^k\right)$

For $j = 1$ to m:

$\quad \alpha_i \xleftarrow{\$} \mathbb{Z}_q$

$\quad h_i = H(ID_i)$

$\quad y_i = (h_i)^{\alpha_i}$ Obtain 📜 for y_i

Online

$\xleftarrow{y_1, \dots, y_m \text{ 📜}}$

$\xrightarrow[\{u_1, \dots, u_n\}]{z_1, \dots, z_m}$

For $i = 1$ to m:

$\quad$ Verify 📜

$\quad z_i = y_i^k$

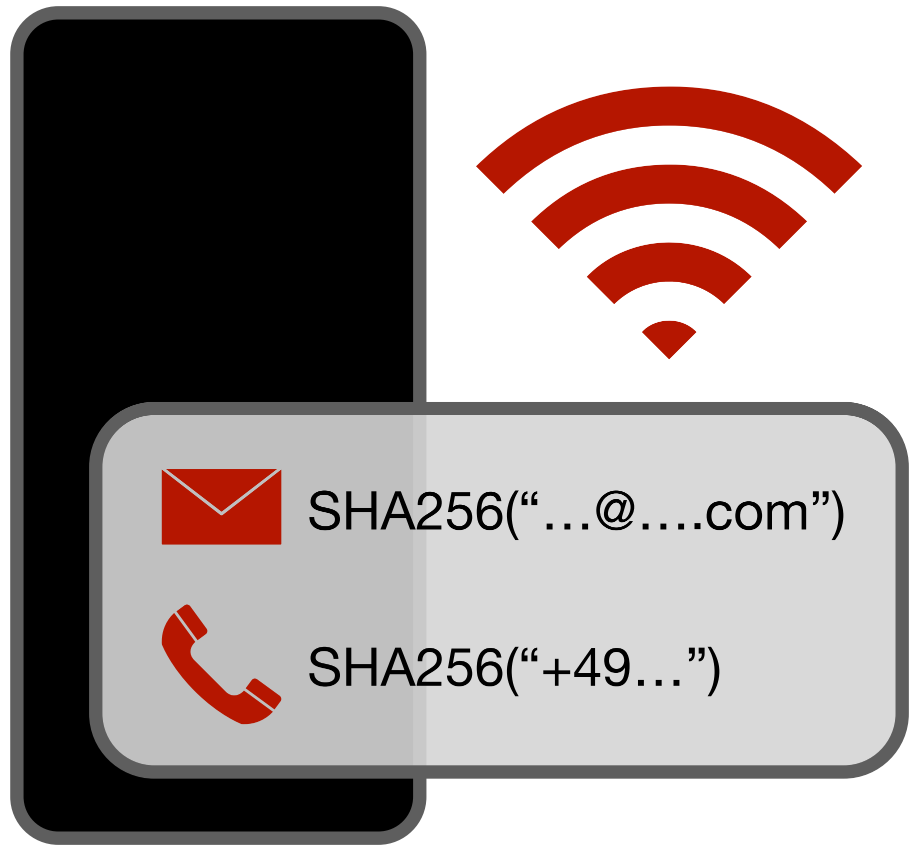
For $i = 1$ to m:

$\quad v_i = H\left(h_i, (z_i)^{1/\alpha_i}\right)$

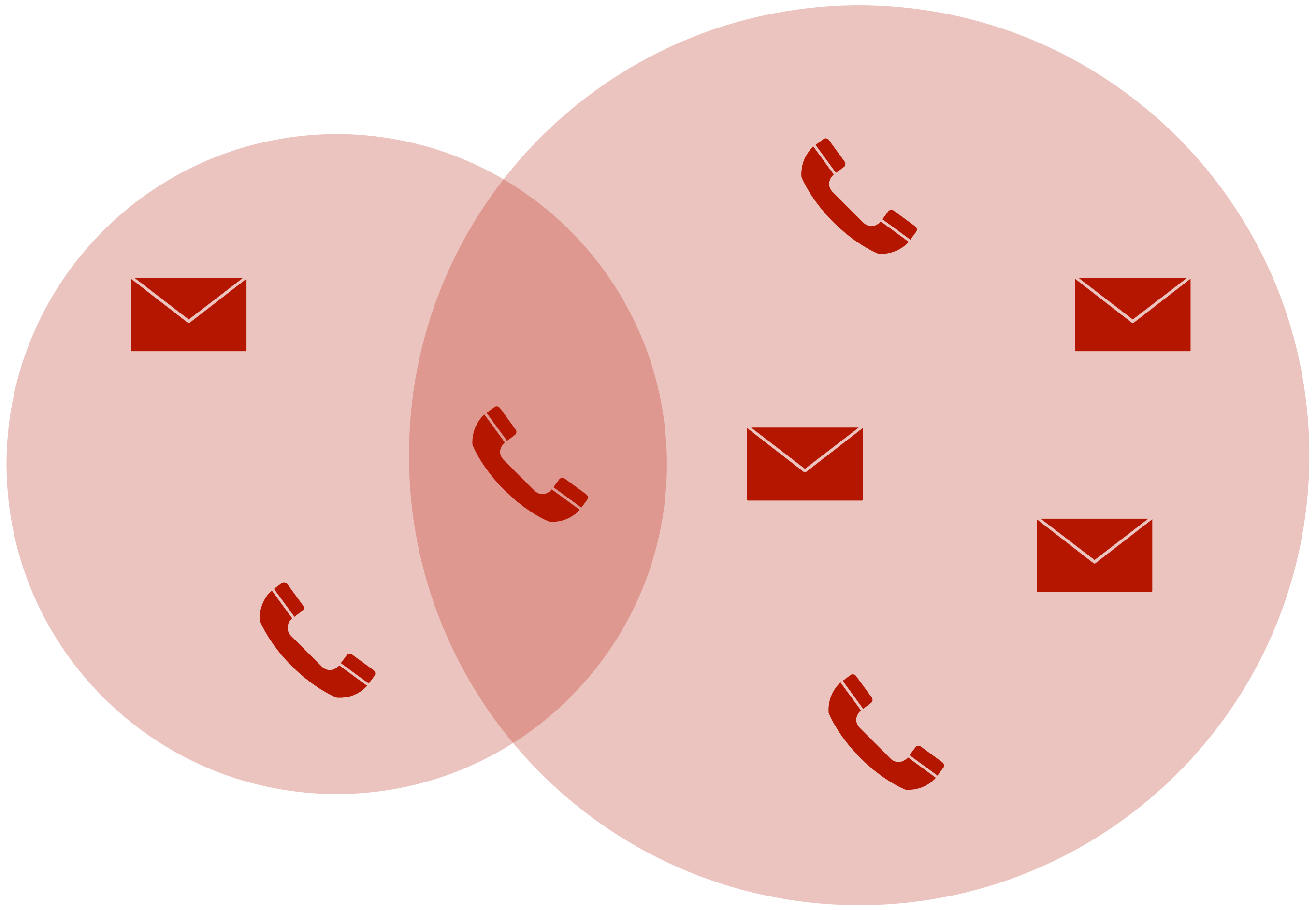
Output $\{ID_i \in IDs | \exists j : u_j = v_i\}$

(simplified version, omits ZK proofs for malicious security)

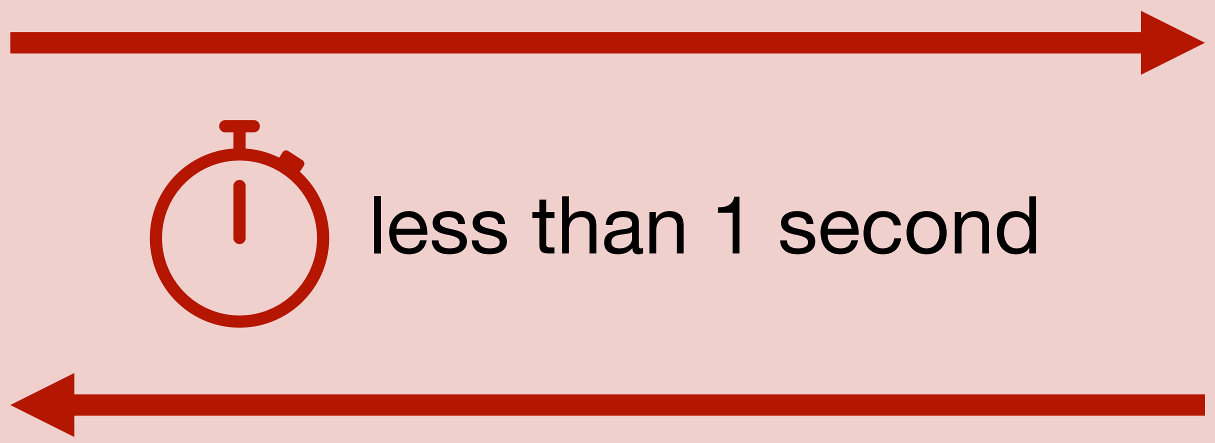
Roadmap to PrivateDrop



Discover
Contact Identifier Leakage
by Apple AirDrop

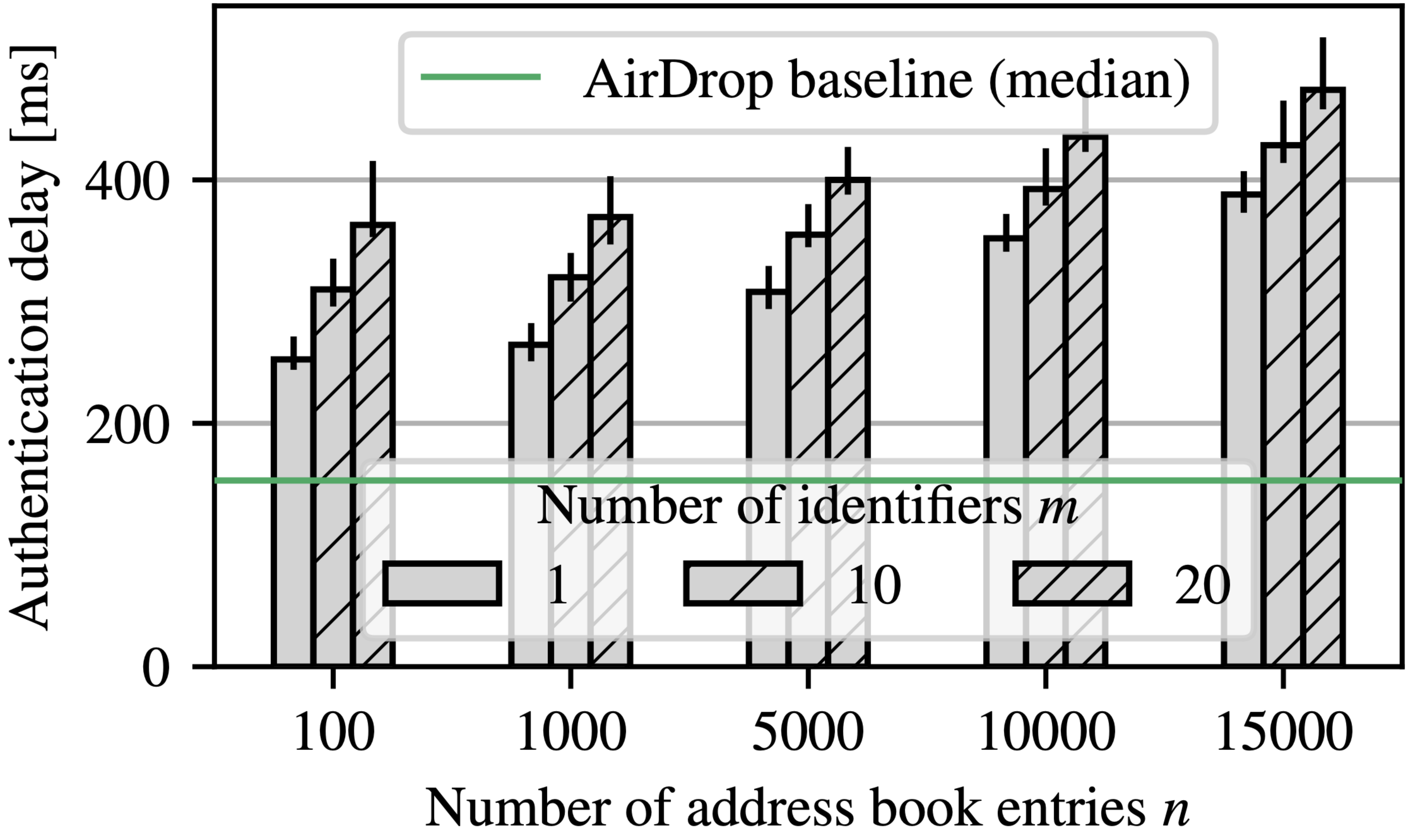


Design
Privacy-Preserving Authentication
via Private Set Intersection



Demonstrate
Native Prototype
with Excellent User Experience

PrivateDrop Results: Authentication Delay



- **Native** implementation for macOS and iOS

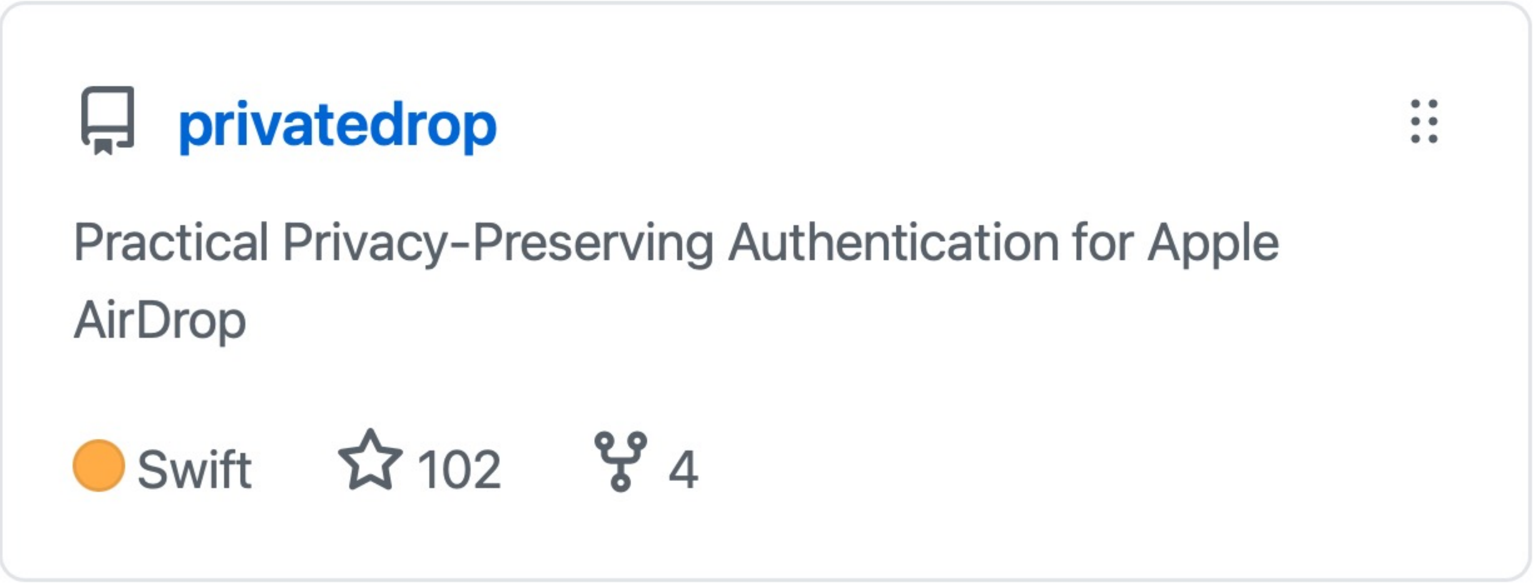
- There is **some (expected) overhead**
- But, authentication **delay is well below 1 second** ("immediate response")

Setup: MacBook and iPhone connected via USB cable (results for Wi-Fi connection with stronger variance in the paper)

PrivateDrop: Privacy-Preserving Mutual Authentication for Apple AirDrop

Open-Source Software

Native implementation for macOS and iOS as open-source software available at privatedrop.github.io



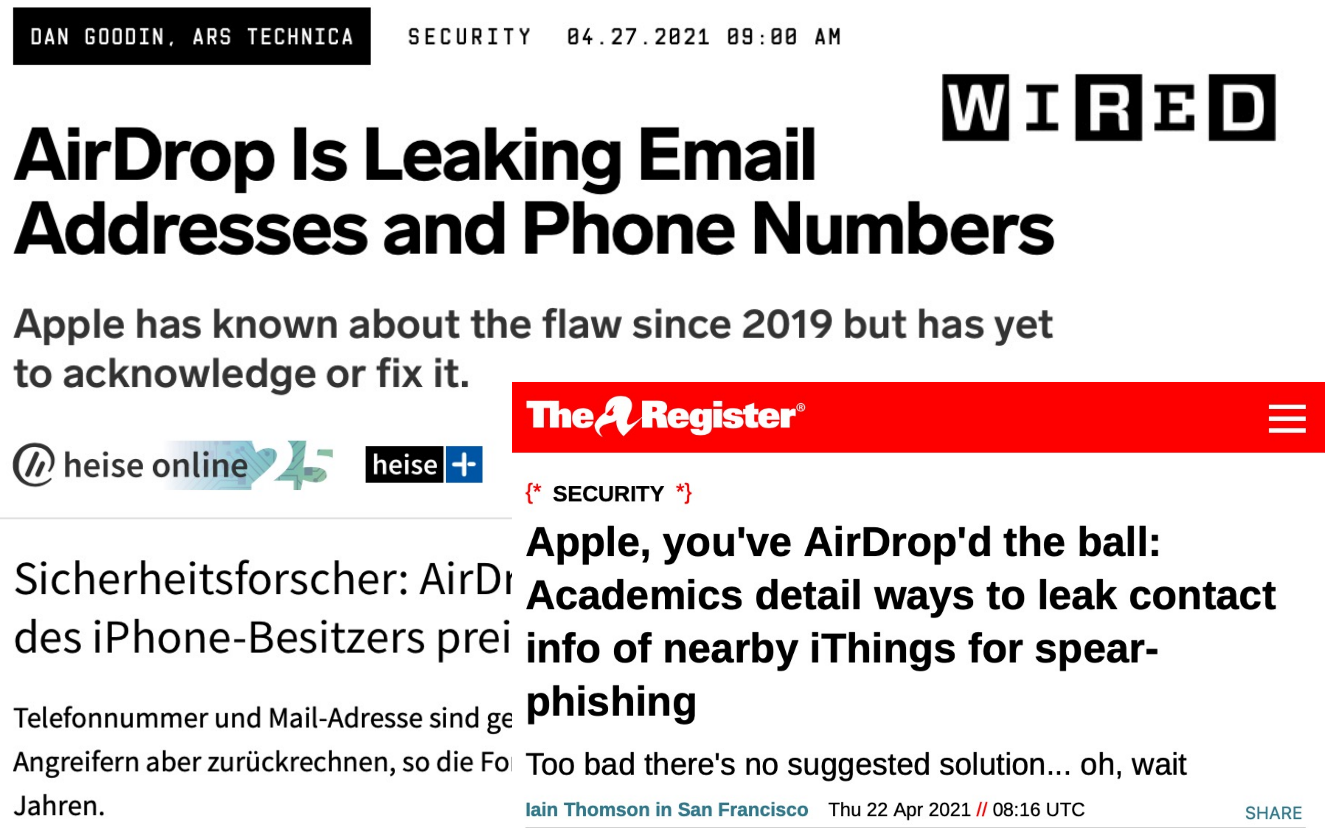
privatedrop

Practical Privacy-Preserving Authentication for Apple AirDrop

● Swift ☆ 102 4

Press and Media

International and national coverage



DAN GOODIN, ARS TECHNICA SECURITY 04.27.2021 09:00 AM

AirDrop Is Leaking Email Addresses and Phone Numbers

Apple has known about the flaw since 2019 but has yet to acknowledge or fix it.

Sicherheitsforscher: AirDr
des iPhone-Besitzers prei

Telefonnummer und Mail-Adresse sind ge
Angreifern aber zurückrechnen, so die Fo
Jahren.

{* SECURITY *}

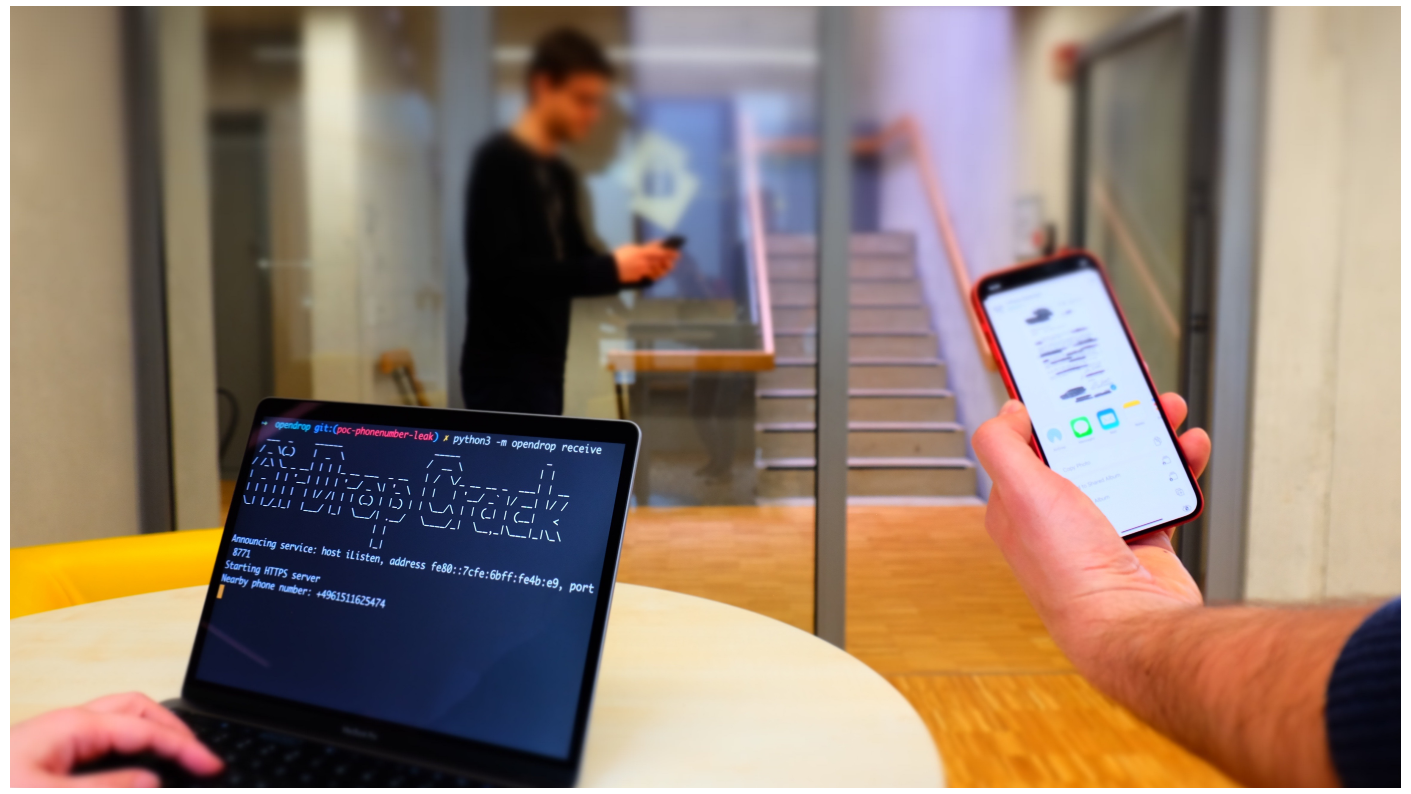
Apple, you've AirDrop'd the ball: Academics detail ways to leak contact info of nearby iThings for spear-phishing

Too bad there's no suggested solution... oh, wait

Iain Thomson in San Francisco Thu 22 Apr 2021 // 08:16 UTC SHARE

Responsible Disclosure

Apple users are still vulnerable to the discovered privacy leaks



References (1/3)

[BBC+11] Pierre Baldi, Roberta Baronio, Emiliano De Cristofaro, Paolo Gasti, Gene Tsudik. **Countering GATTACA: Efficient and Secure Testing of Fully-Sequenced Human Genomes.** *CCS*, 2011.

[CKT10a] Emiliano De Cristofaro, Gene Tsudik. **Practical Private Set Intersection Protocols with Linear Complexity.** *FC*, 2010.

[CKT10b] Emiliano De Cristofaro, Jihye Kim, Gene Tsudik. **Linear-Complexity Private Set Intersection Protocols Secure in Malicious Model.** *ASIACRYPT*, 2010.

[CLR17] Hao Chen, Kim Laine, Peter Rindal. **Fast Private Set Intersection from Homomorphic Encryption.** *CCS*, 2017.

[CHLR18] Hao Chen, Zhicong Huang, Kim Laine, Peter Rindal. **Labeled PSI from Fully Homomorphic Encryption with Malicious Security.** *CCS*, 2018.

[DRRT18] Daniel Demmler, Peter Rindal, Mike Rosulek, Ni Trieu. **PIR-PSI: Scaling Private Contact Discovery.** *PoPETS*, 2018.

[HHSSW21] Alexander Heinrich, Matthias Hollick, Thomas Schneider, Milan Stute, Christian Weinert. **PrivateDrop: Practical Privacy-Preserving Authentication for Apple AirDrop.** *USENIX Security*, 2021.

[HWSDS21] Christoph Hagen, Christian Weinert, Christoph Sendner, Alexandra Dmitrienko, Thomas Schneider. **All the Numbers are US: Large-scale Abuse of Contact Discovery in Mobile Messengers.** *NDSS*, 2021.

[JL09] Stanislaw Jarecki, Xiaomin Liu. **Efficient Oblivious Pseudorandom Function with Applications to Adaptive OT and Secure Computation of Set Intersection.** *TCC*, 2009.

References (2/3)

[JL10] Stanislaw Jarecki, Xiaomin Liu. **Fast Secure Computation of Set Intersection.** *SCN*, 2010.

[KKRT16] Vladimir Kolesnikov, Ranjit Kumaresan, Mike Rosulek, Ni Trieu. **Efficient Batched Oblivious PRF with Applications to Private Set Intersection.** *CCS*, 2016.

[KKRT16] Ágnes Kiss, Jian Liu, Thomas Schneider, N. Asokan, Benny Pinkas. **Private Set Intersection for Unequal Set Sizes with Mobile Applications.** *PoPETS*, 2017.

[KRSSW19] Daniel Kales, Christian Rechberger, Thomas Schneider, Matthias Senker, Christian Weinert. **Mobile Private Contact Discovery at Scale.** *USENIX Security*, 2019.

[Mea86] Catherine A. Meadows. **A More Efficient Cryptographic Matchmaking Protocol for Use in the Absence of a Continuously Available Third Party.** *S&P*, 1986.

[PSZ14] Benny Pinkas, Thomas Schneider, Michael Zohner. **Faster Private Set Intersection based on OT Extension.** *USENIX Security*, 2014.

[PSZ15] Benny Pinkas, Thomas Schneider, Michael Zohner. **Phasing: Private Set Intersection using Permutation-based Hashing.** *USENIX Security*, 2015.

[PRTY] Benny Pinkas, Mike Rosulek, Ni Trieu, Avishay Yanai. **PSI from PaXoS: Fast, Malicious Private Set Intersection.** *EUROCRYPT*, 2020.

[RA18] Amanda Cristina Davi Resende, Diego F. Aranha. **Faster Unbalanced Private Set Intersection.** *FC*, 2018.

References (3/3)

[RR17] Peter Rindal, Mike Rosulek. **Malicious-Secure Private Set Intersection via Dual Execution.** *CCS*, 2017.

[Sha80] Adi Shamir. **On the Power of Commutativity in Cryptography.** *ICALP*, 1980.

[SKH18] Milan Stute, David Kreitschmann, Matthias Hollick. **One Billion Apples' Secret Sauce: Recipe for the Apple Wireless Direct Link Ad hoc Protocol.** *MobiCom*, 2018.

[SNMHKNH19] Milan Stute, Sashank Narain, Alex Mariotto, Alexander Heinrich, David Kreitschmann, Guevara Noubir, Matthias Hollick. **A Billion Open Interfaces for Eve and Mallory: MitM, DoS, and Tracking Attacks on iOS and macOS Through Apple Wireless Direct Link.** *USENIX Security*, 2019.

Acknowledgements

- Icons from Font Awesome, Apple Keynote, and Microsoft PowerPoint
- Photo by Ann-Kathrin Braun and Daniela Fleckenstein