# Secure Interoperable Instant Messaging

Technical Report for Bundesnetzagentur

**23-05-03**

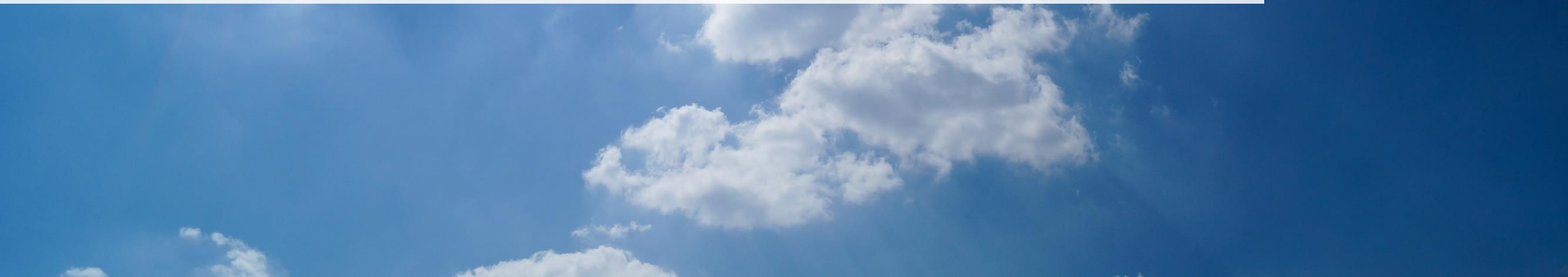

**Real-World Cryptography Group**
**FAU Erlangen-Nürnberg, Germany**

Prof. Dr. Paul Rösler

**Chair for Network and Data Security**
**Ruhr University Bochum, Germany**

Prof. Dr. Jörg Schwenk

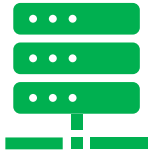
**Hackmanit GmbH**
**Bochum, Germany**

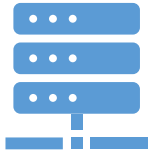



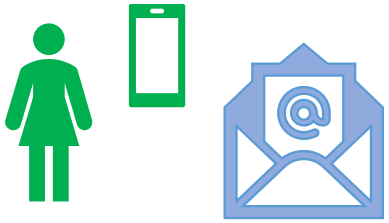
# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# End-to-End-Encryption

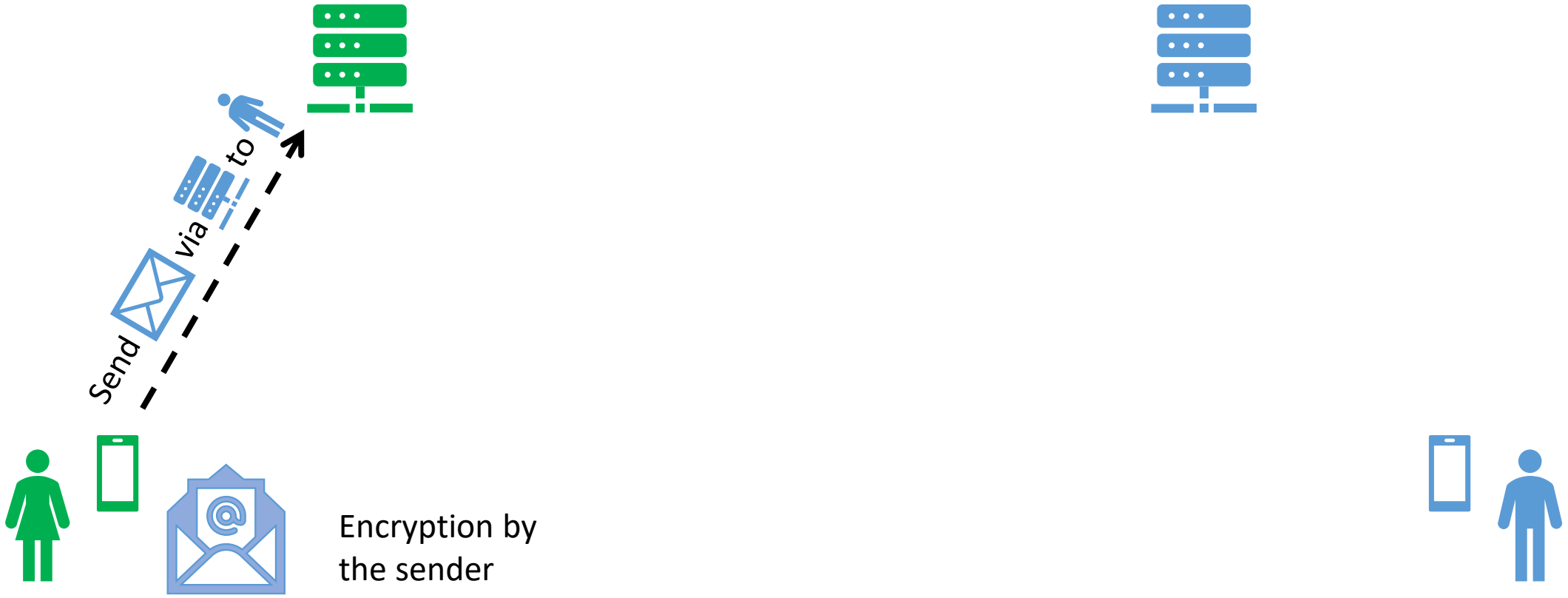




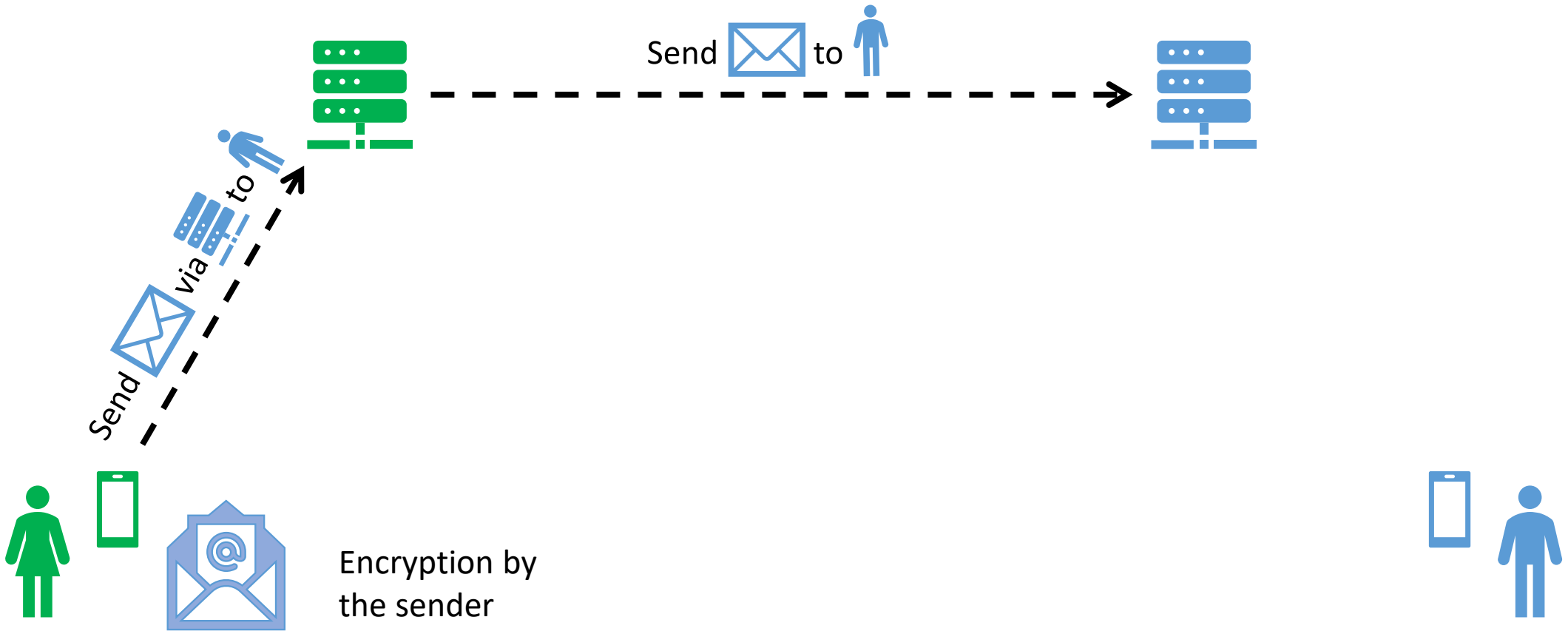


Encryption by
the sender

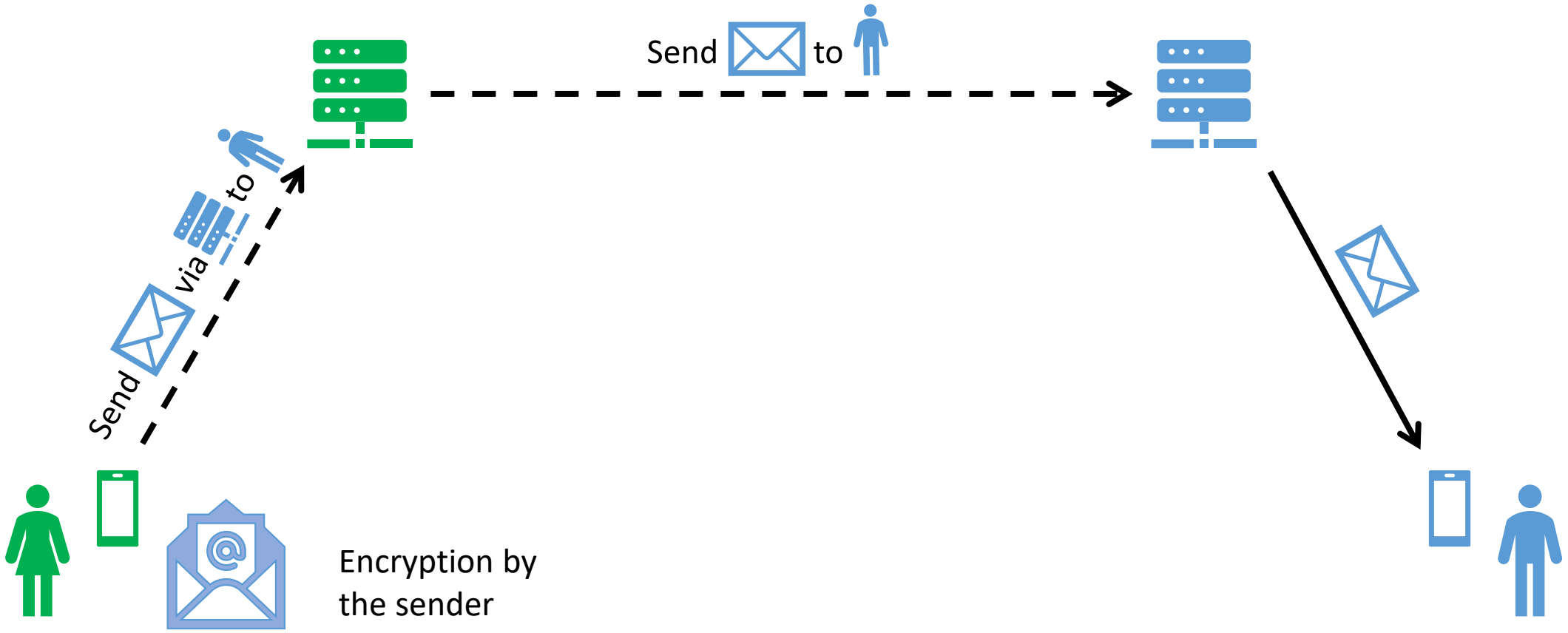
# End-to-End-Encryption



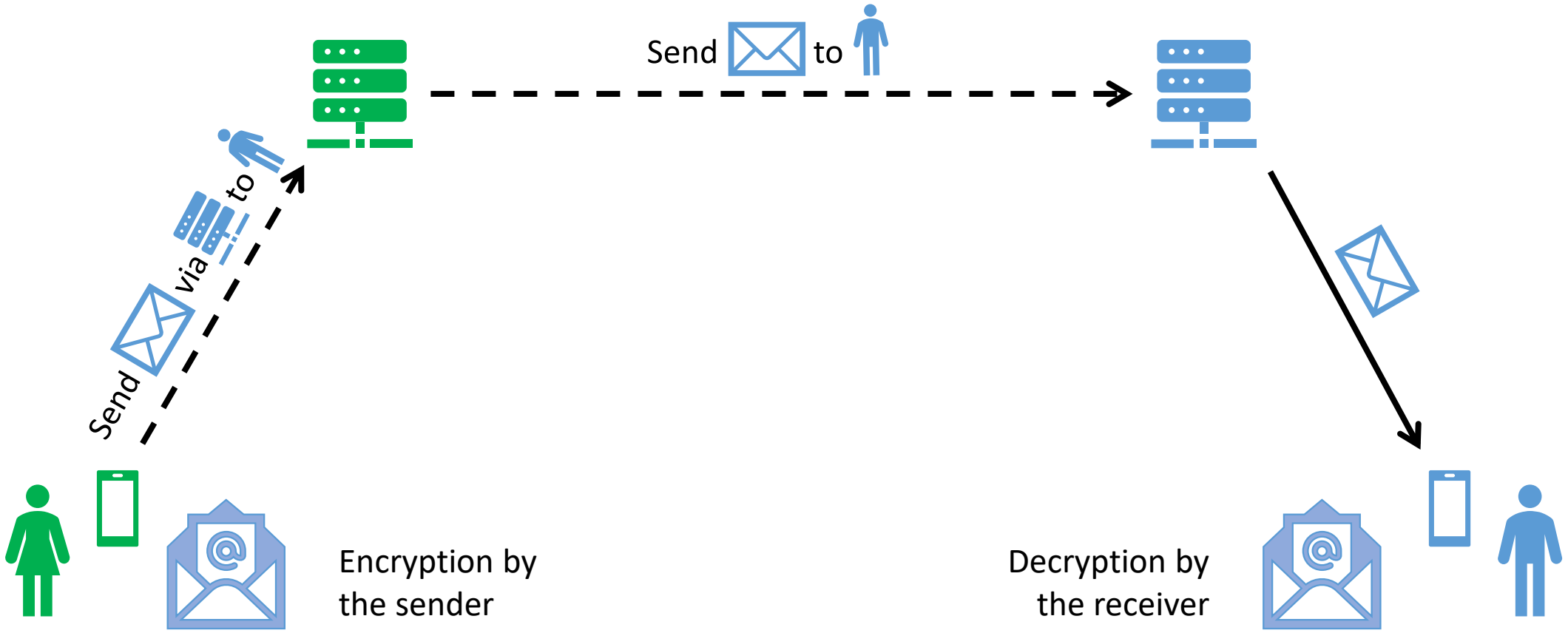
# End-to-End-Encryption



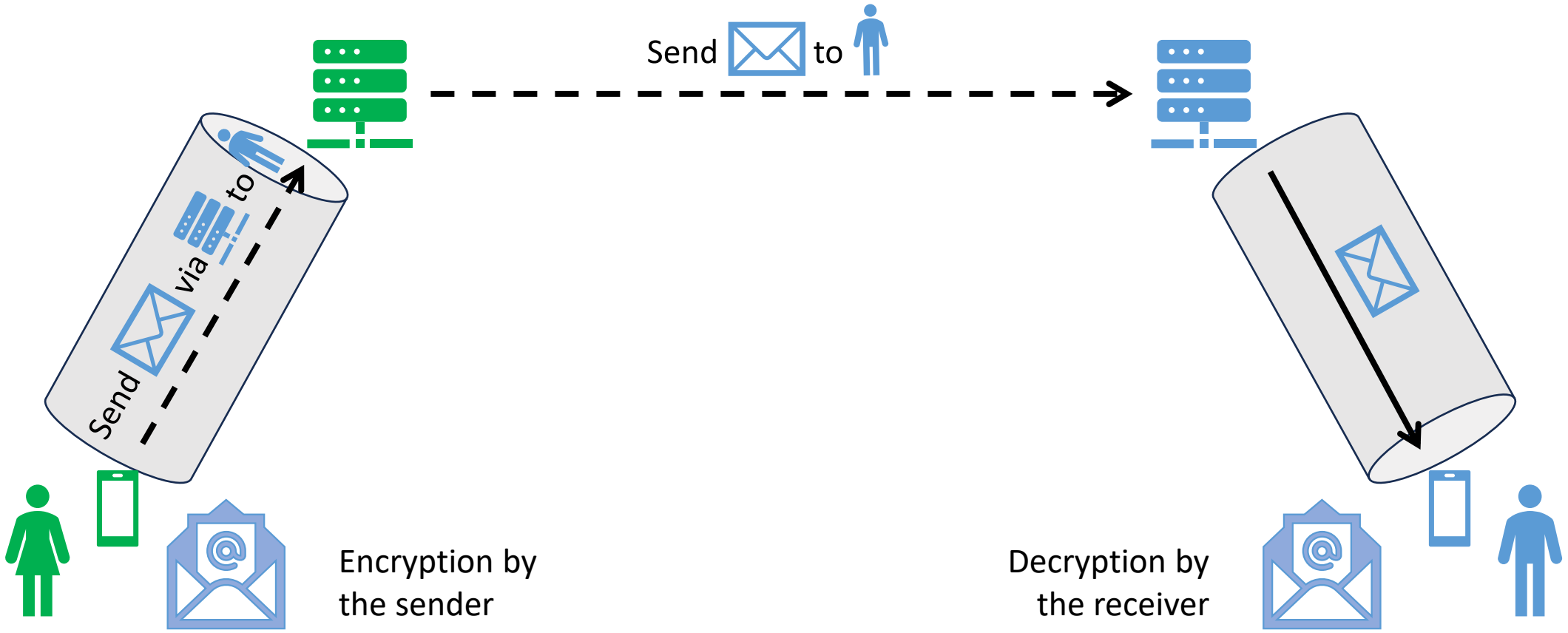
# End-to-End-Encryption



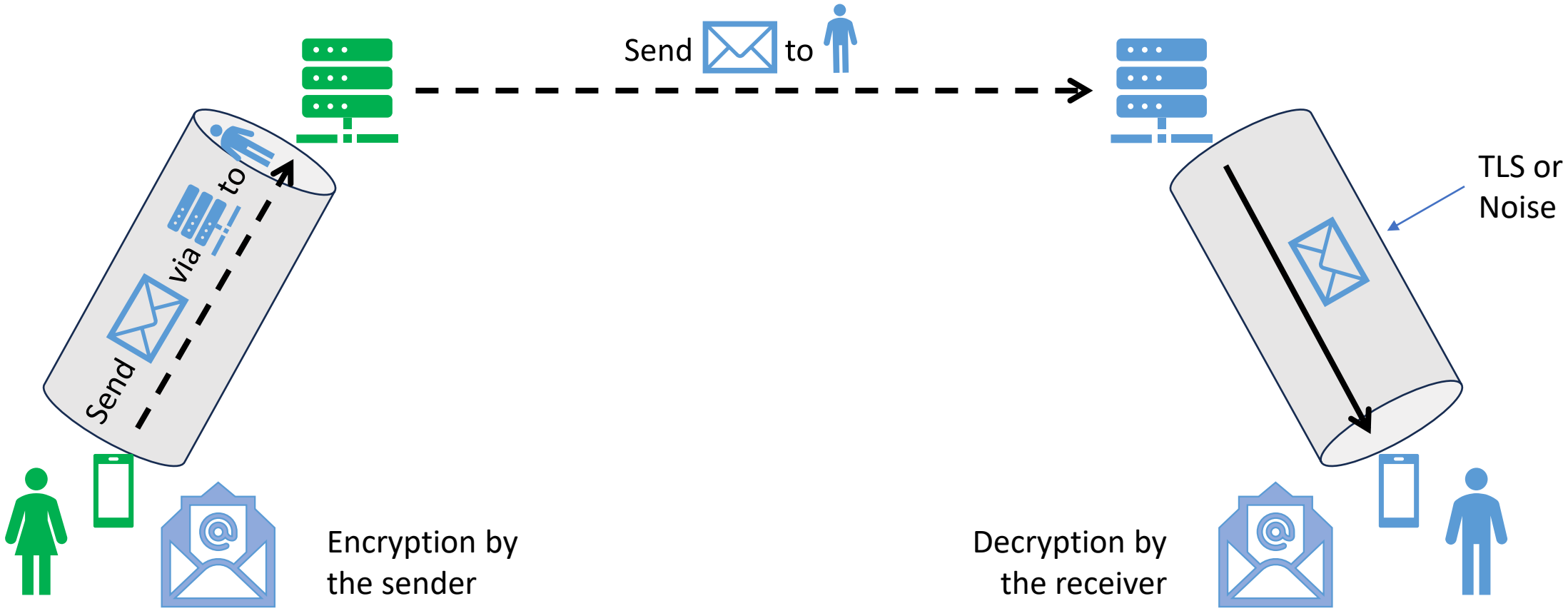
# End-to-End-Encryption



# End-to-End-Encryption



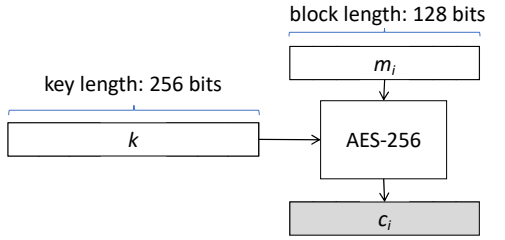
# End-to-End-Encryption

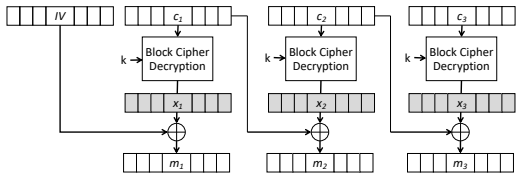


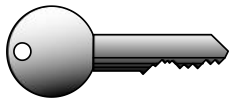
# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# For E2EE we need:



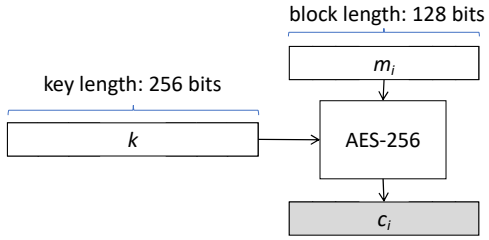



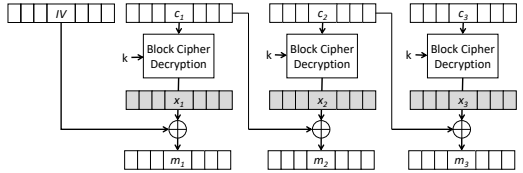
encryption algorithms
(AES, ChaCha20)

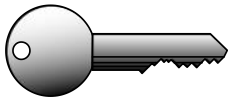
encryption modes
(authenticated encryption)

symmetric keys

via
Authenticated Key Agreement
Ratcheting
(MLS)



# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

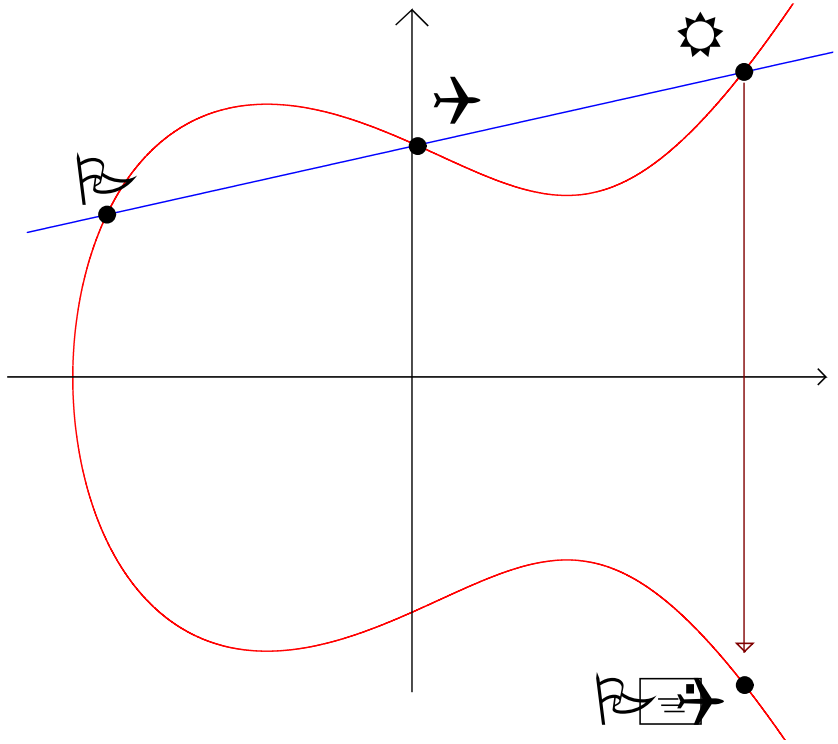
# Diffie-Hellman Key Exchange

Public parameters: $(EC(a, b), P, q)$

| *Alice* | | *Bob* |
|---|---|---|
| $a \xleftarrow{\$} \mathbb{Z}_q$ | | |
| $\alpha \leftarrow a \cdot P$ | | |
| | $\xrightarrow{\alpha}$ | |
| | | $b \xleftarrow{\$} \mathbb{Z}_q$ |
| | | $\beta \leftarrow b \cdot P$ |
| | $\xleftarrow{\beta}$ | |
| $k \leftarrow a \cdot \beta = ab \cdot P$ | | $k \leftarrow b \cdot \alpha = ba \cdot P$ |

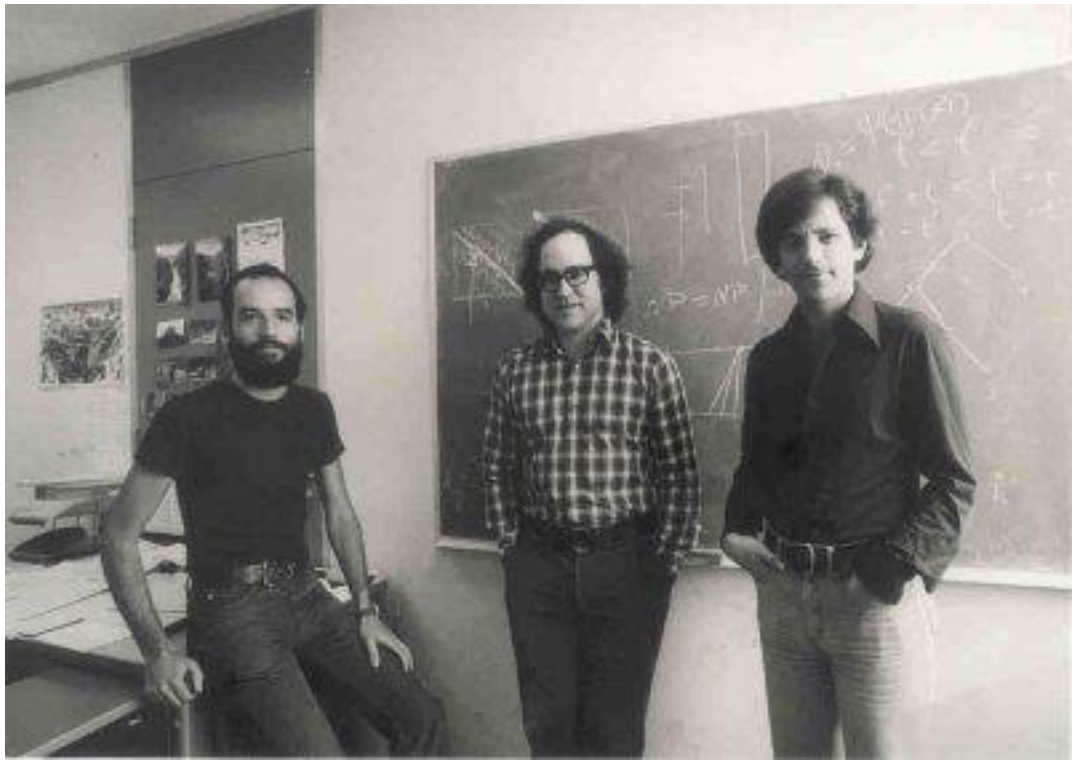
# Diffie-Hellman Key Exchange

- DHKE important building block in modern cryptography
- Mostly used over Elliptic Curves today
- Insecure against active attackers
  - cyber criminals
  - foreign intelligence agencies



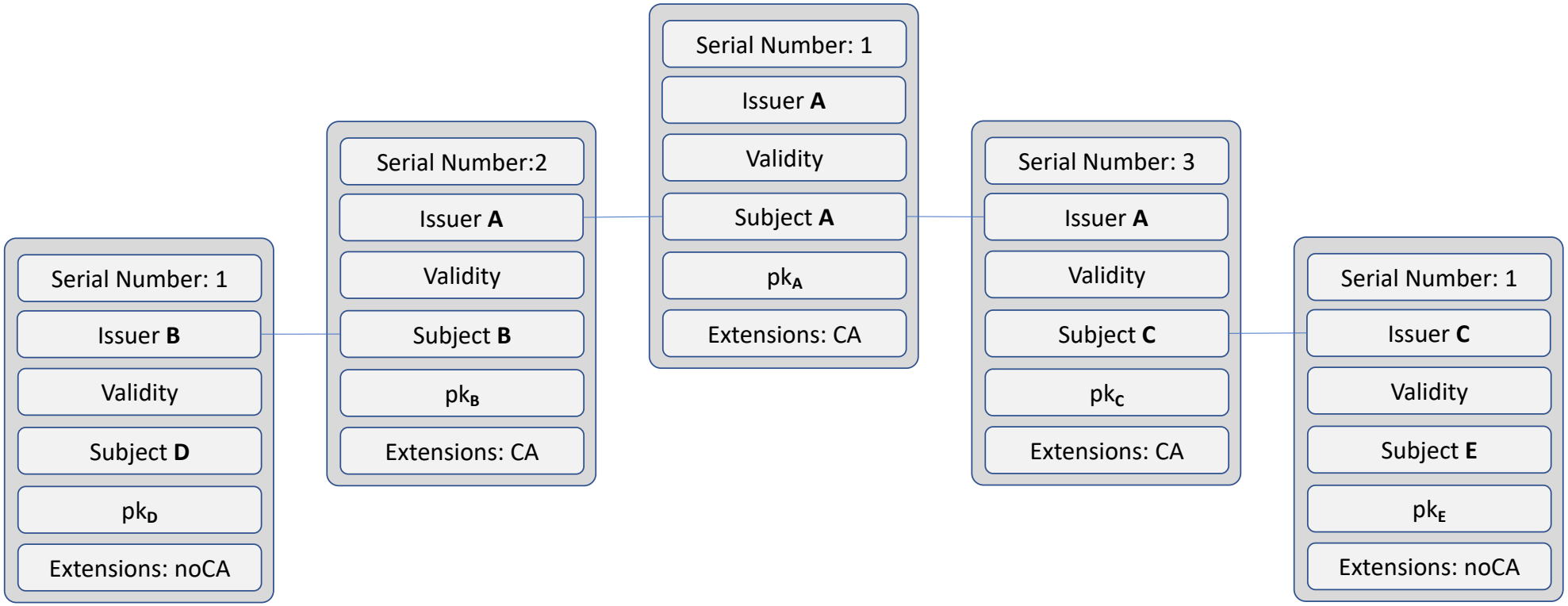
# Digital Signatures

- Rivest-Shamir-Adleman (RSA) Signatures
- Digital Signature Algorithm (DSA) Signatures
  - modulo prime numbers
  - over elliptic curves



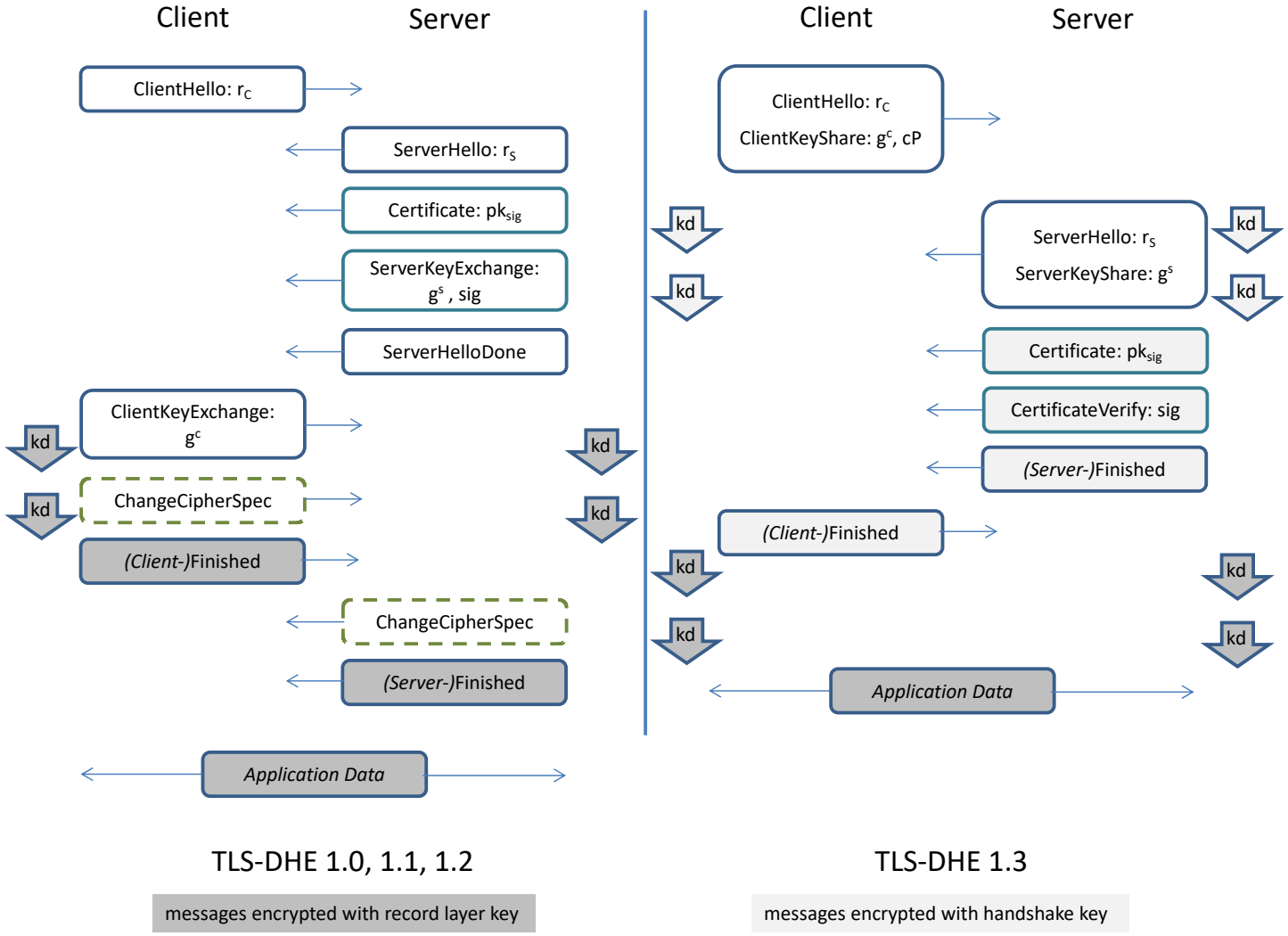
Ron Rivest, Adi Shamir, and Leonard Adleman in 1978

# Public-Key Infrastructures



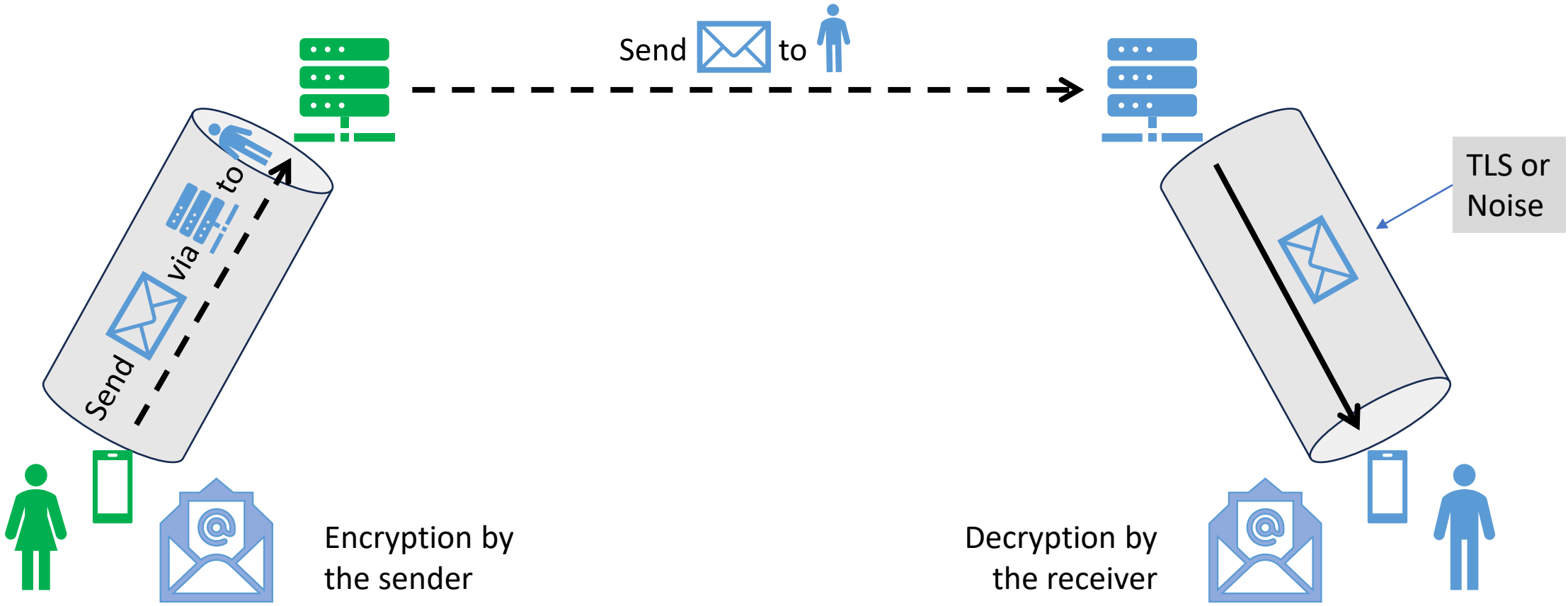

*Digital signatures allow for inheritance of trust*

# Transport Layer Security




- TLS-DHE uses DHKE and digital signatures
- Noise and X3DH authenticate differently

# End-to-End-Encryption



# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

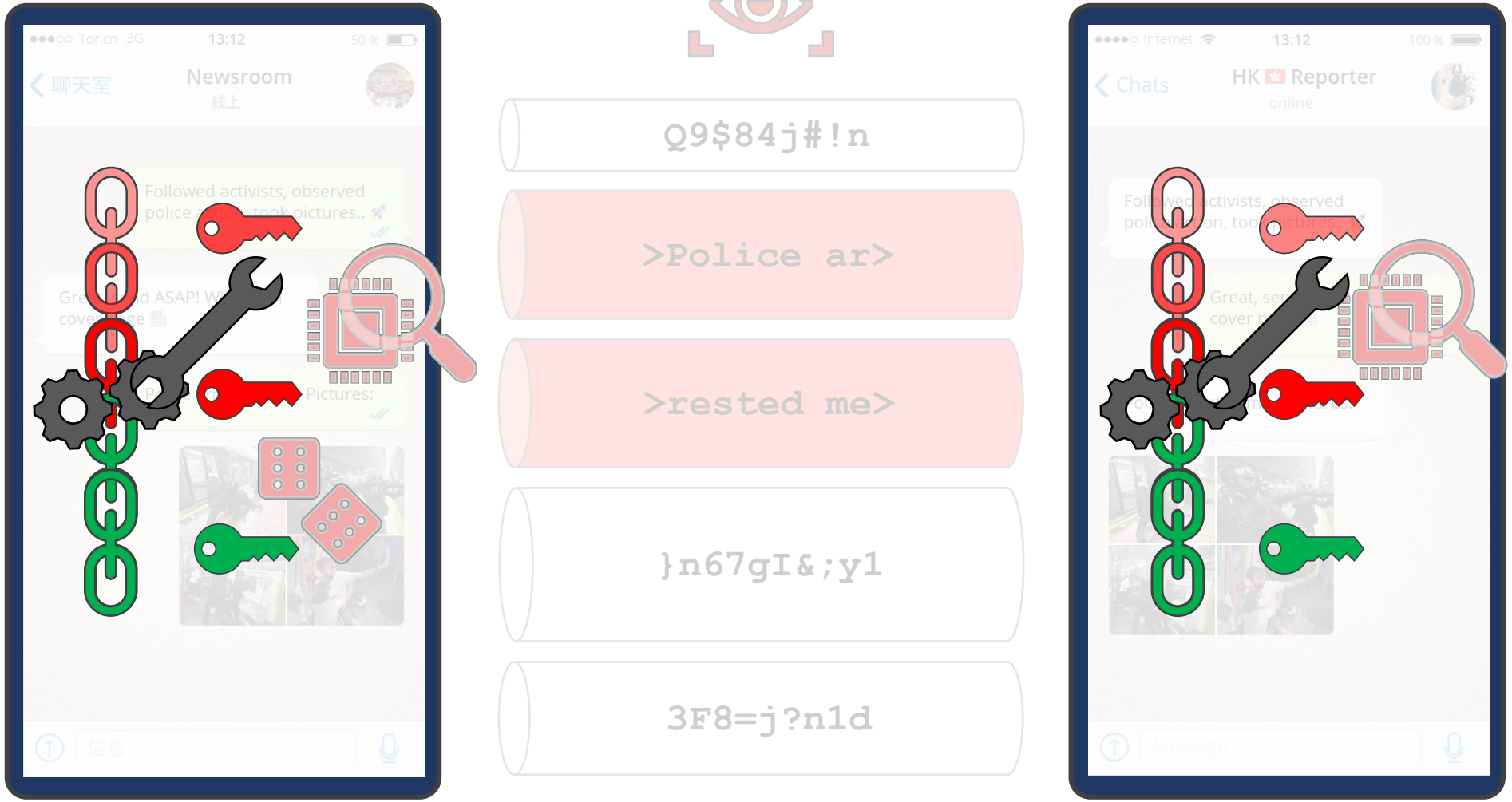
# Overview of Messengers

| IM Protocol | Two-Party | Group | Real Time |
|---|---|---|---|
| **Signal** | Double Ratchet (**DR**) | **DR** | WebRTC |
| **WhatsApp** | **DR** | Sender Key (**SK**) | SRTP |
| **Facebook Messenger** | **DR** with Message Franking | **SK** with Message Franking | Undocumented |
| **Wire** | Proteus (≈**DR**; diff. AE) | Proteus (≈**DR**) | SRTP |
| **Matrix** | Olm (≈**DR**; diff. KDF) | Megolm (≈**SK**) | WebRTC |
| **iMessage** | Public-key encryption | Public-key encryption | SRTP |
| **Telegram** | MTProto | Unencrypted | MTProto |

# Ratcheting: Continuous Key Exchange

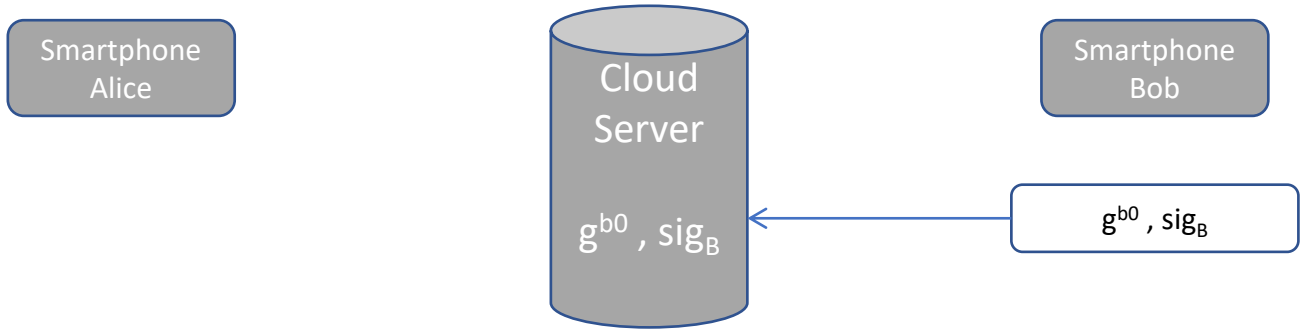
Alice

Bob

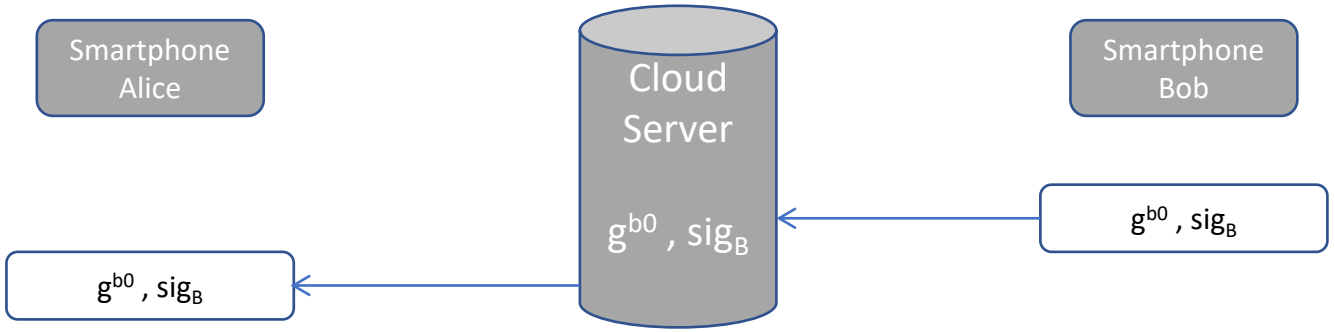



**Ratchet:**
Regularly replace key material in local states

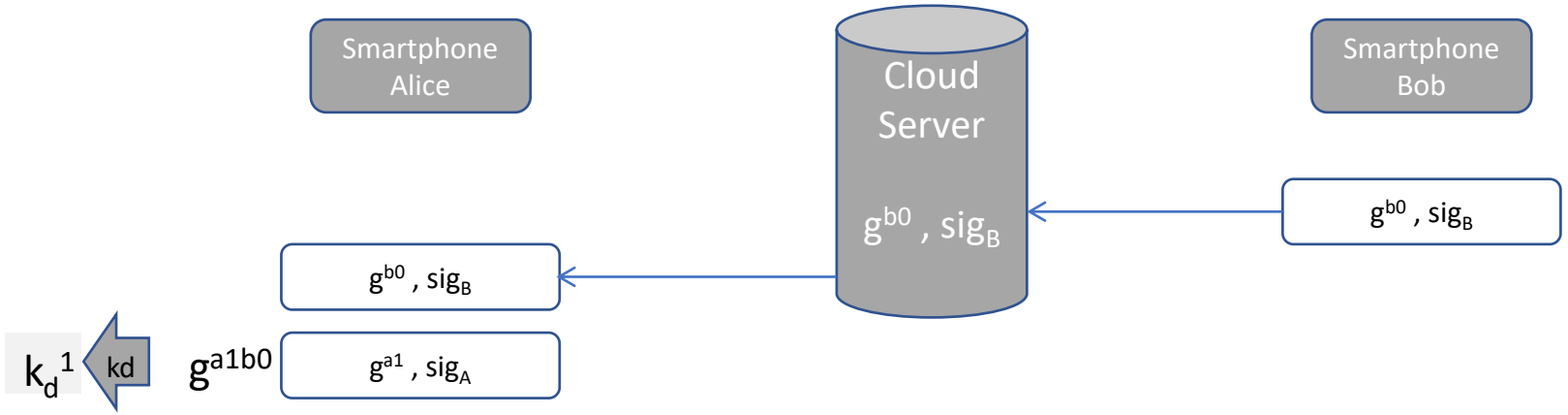
# Double Ratchet – continuous DHKE



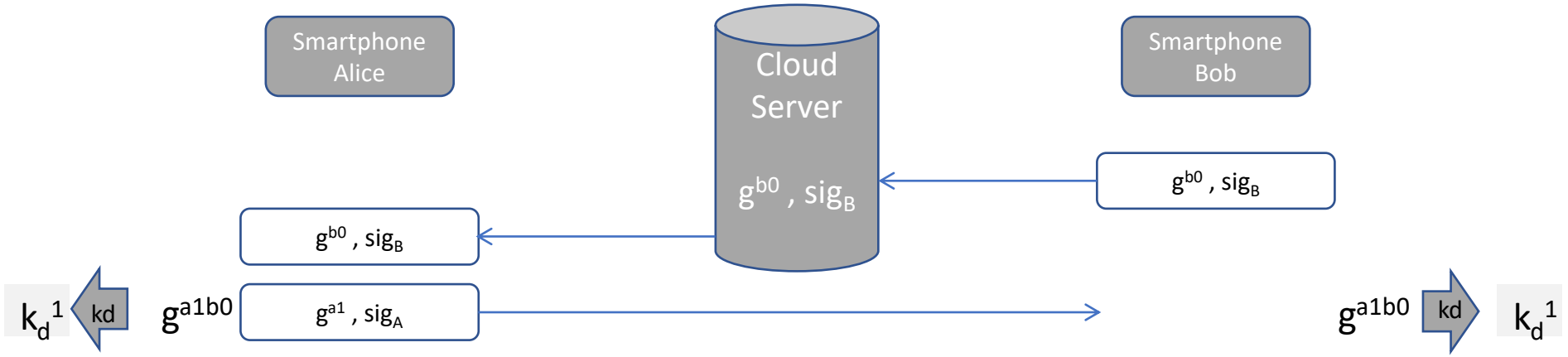
# Double Ratchet – continuous DHKE



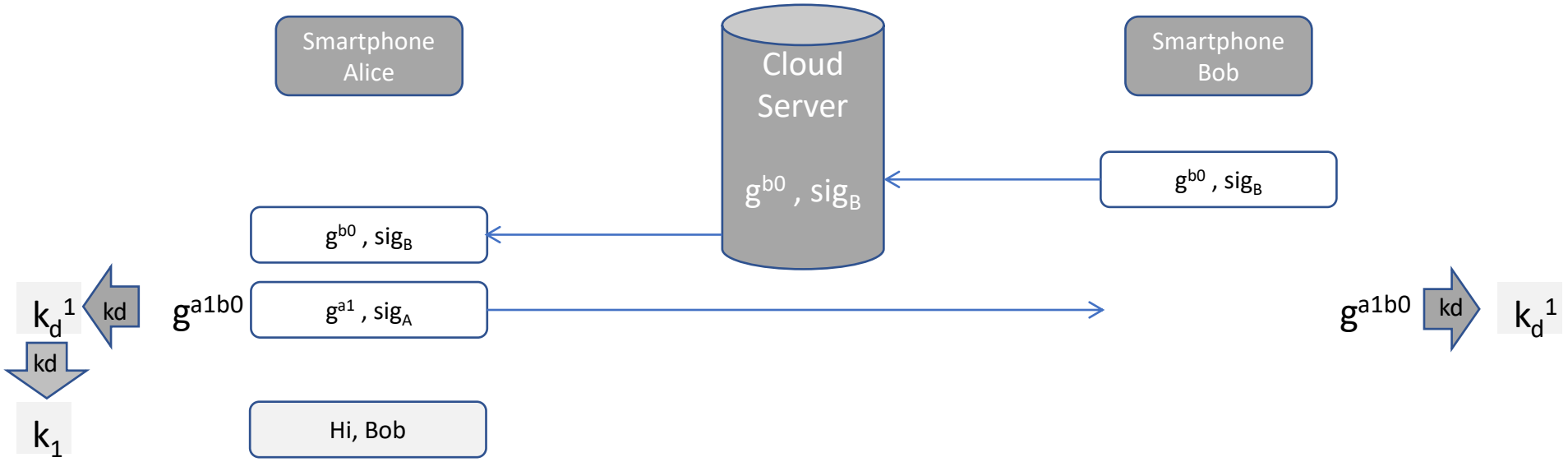
# Double Ratchet – continuous DHKE



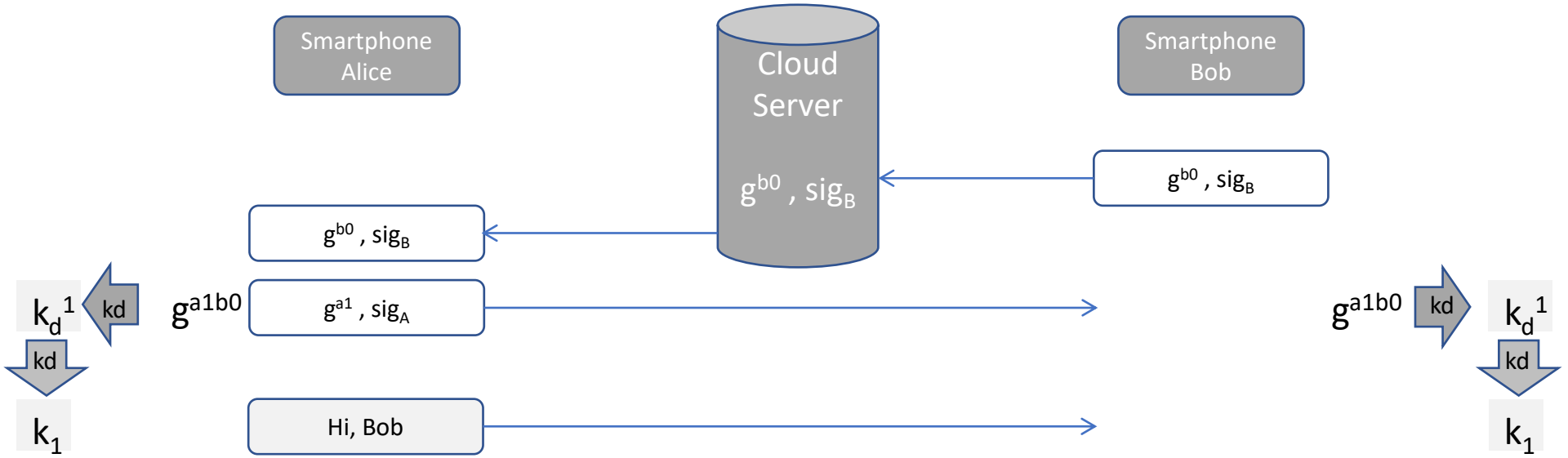
# Double Ratchet – continuous DHKE



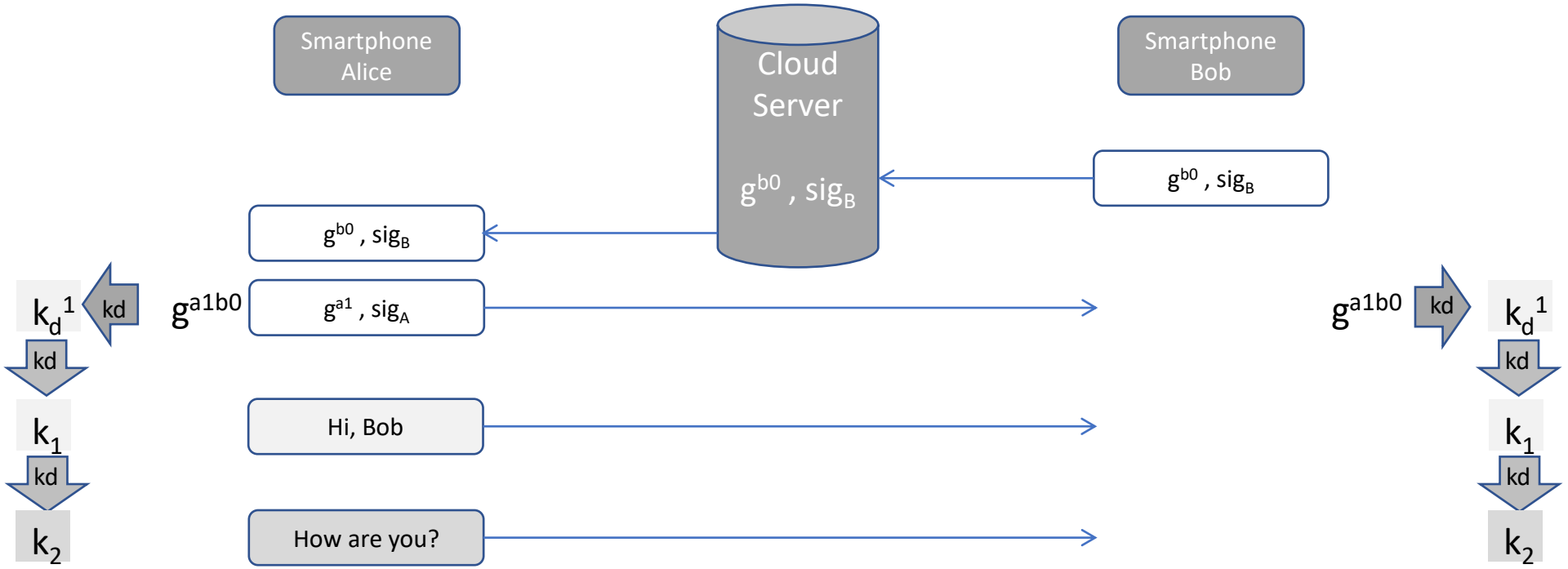
# Double Ratchet – continuous DHKE



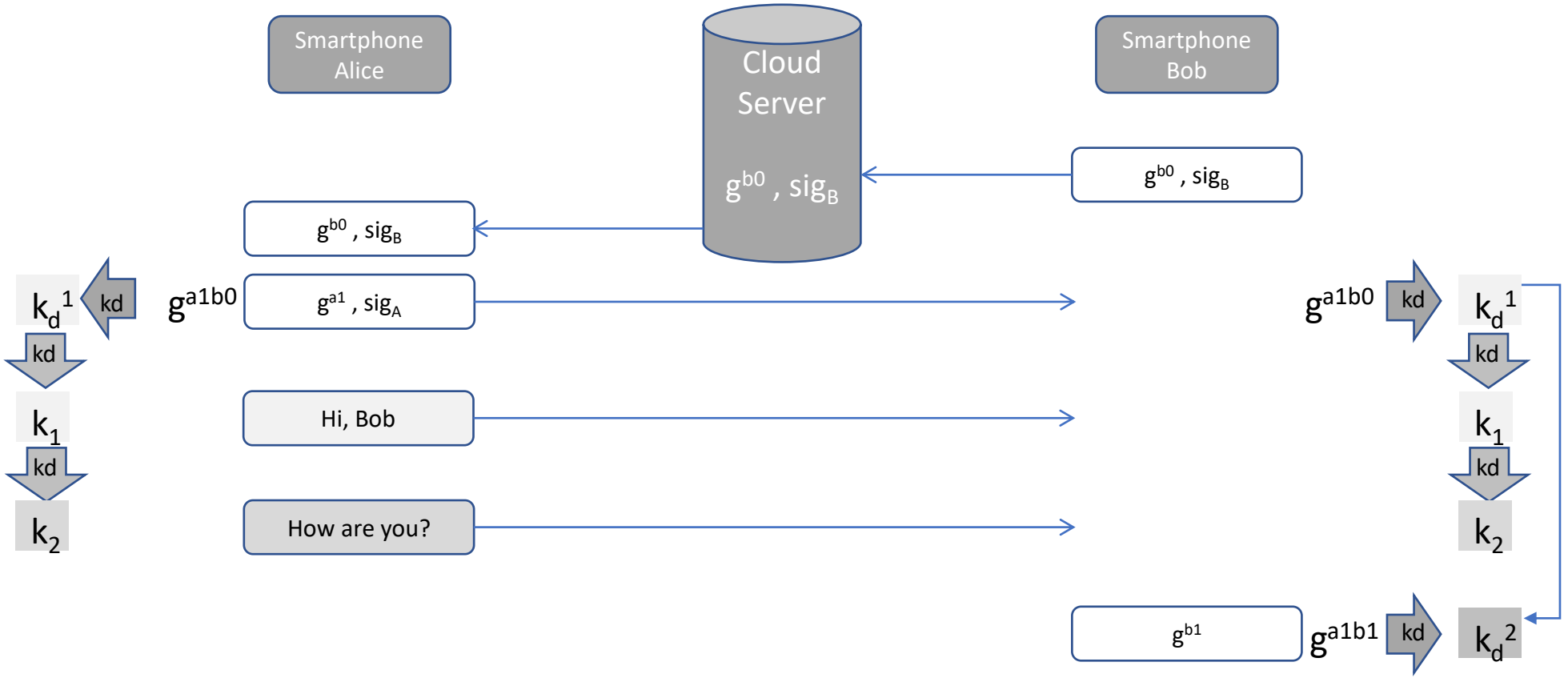
# Double Ratchet – continuous DHKE



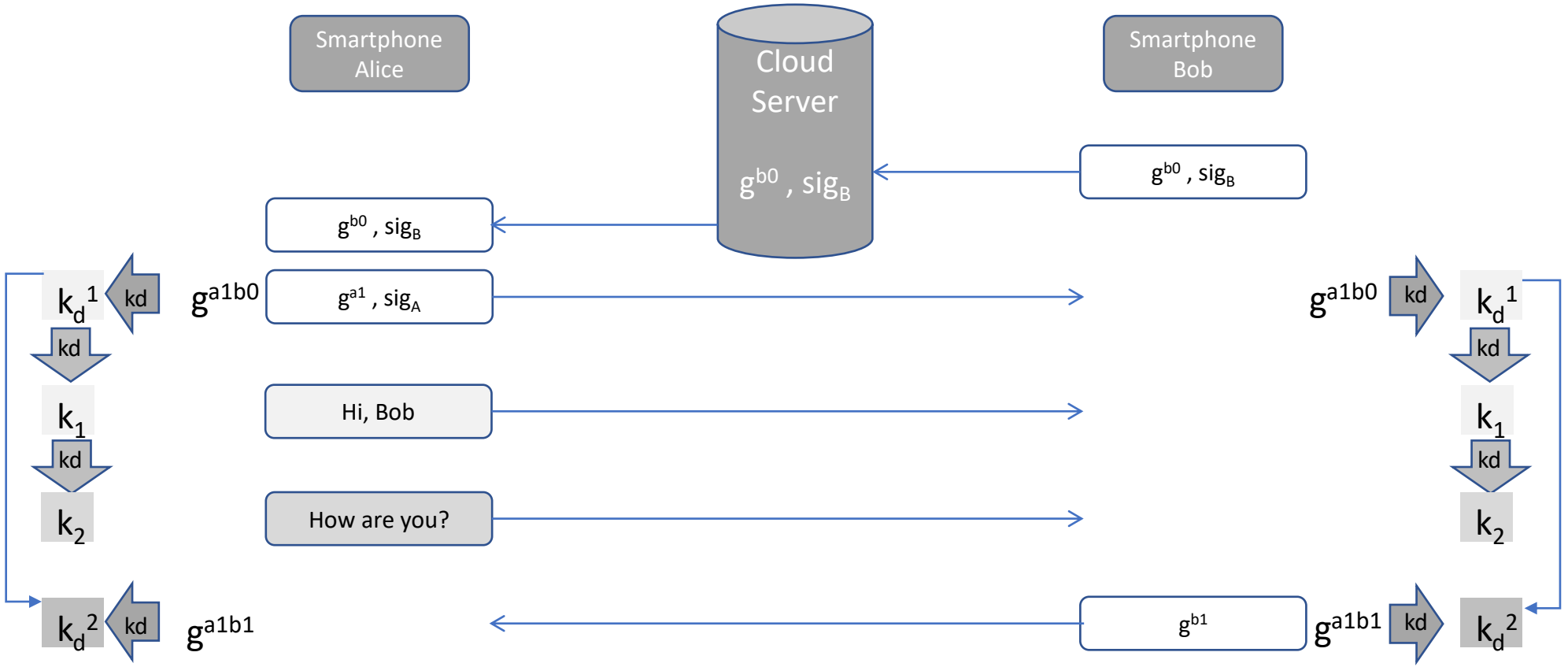
# Double Ratchet – continuous DHKE



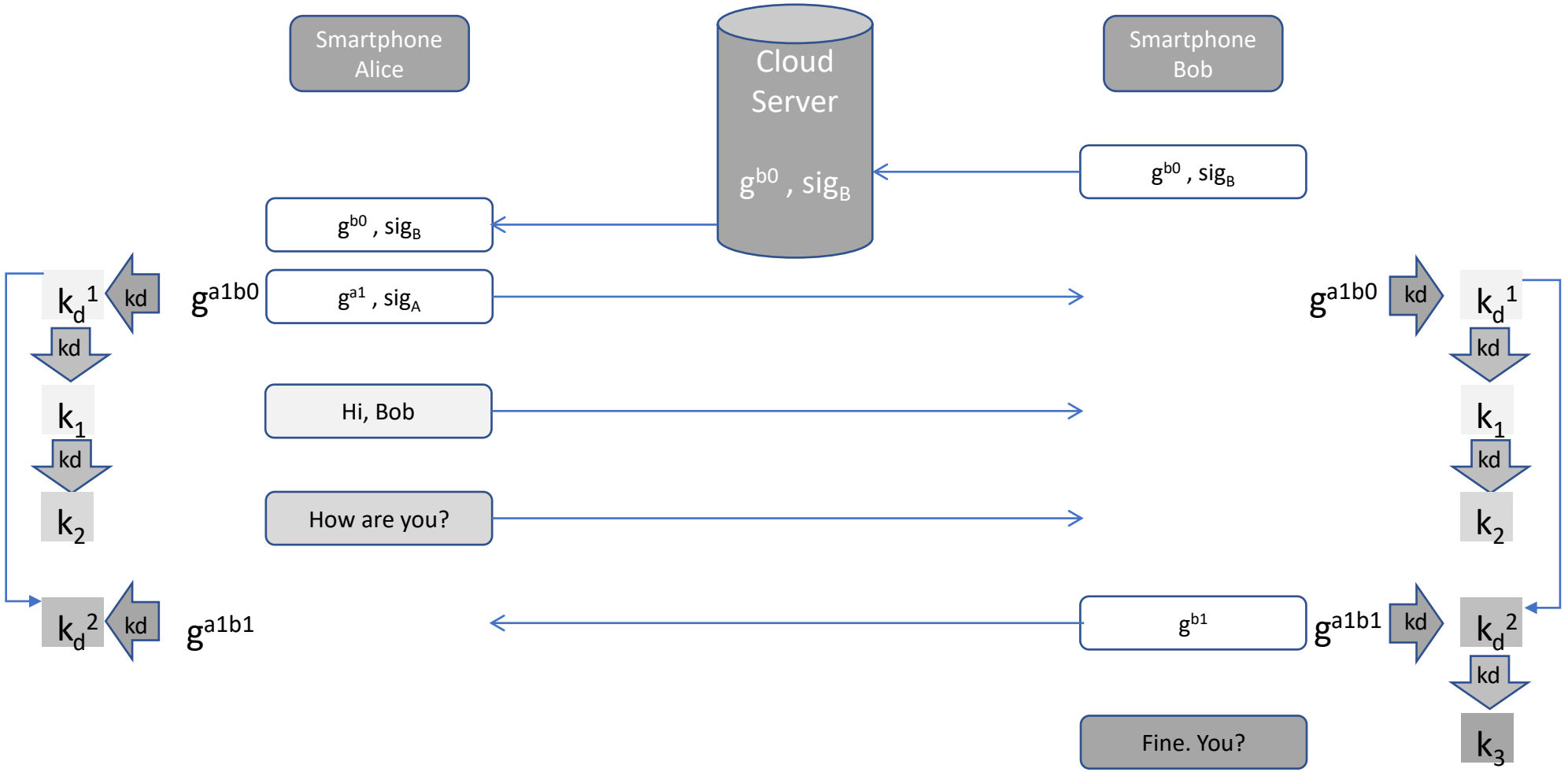
# Double Ratchet – continuous DHKE



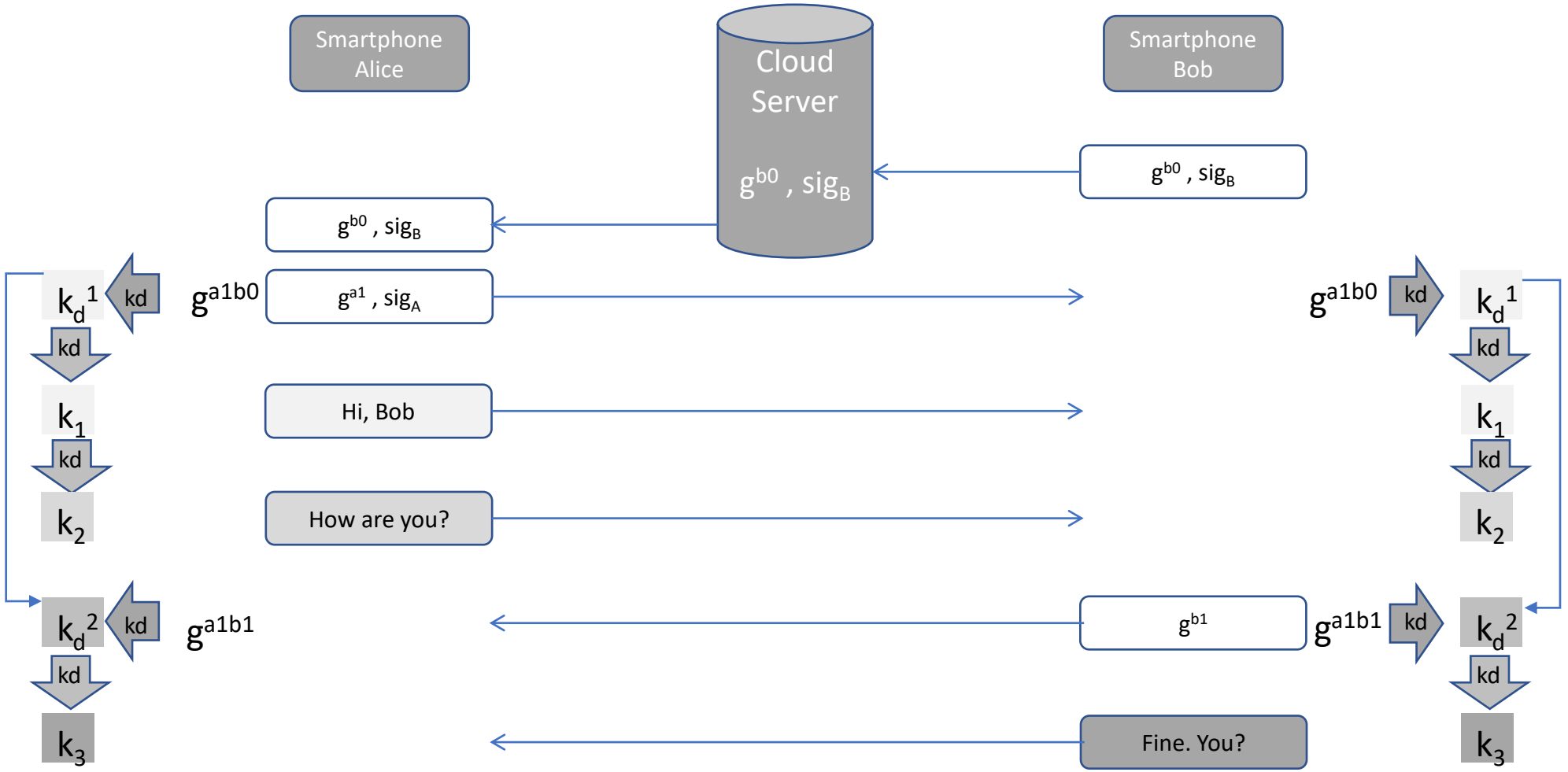
# Double Ratchet – continuous DHKE



# Double Ratchet – continuous DHKE



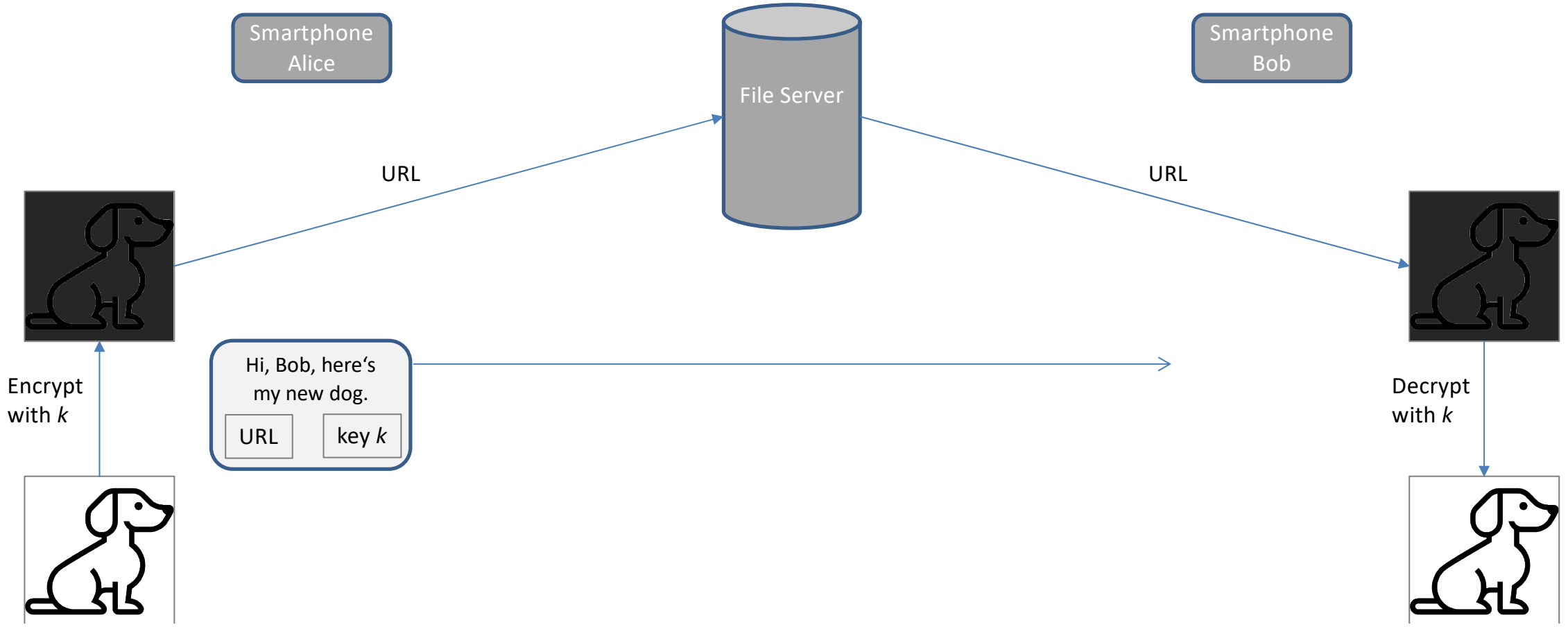
# Double Ratchet – continuous DHKE



# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# Key and URL via Text Message

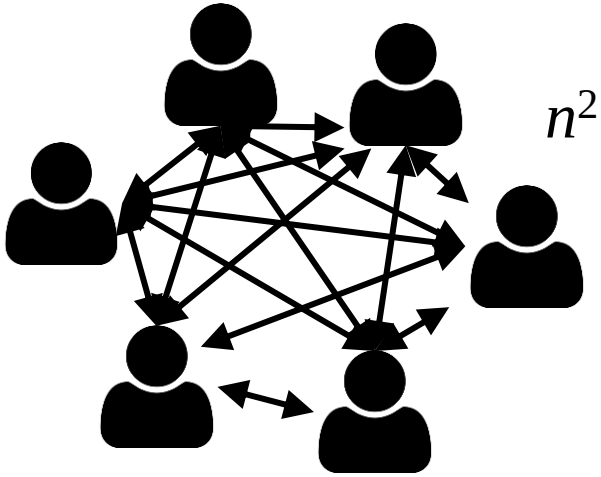


# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

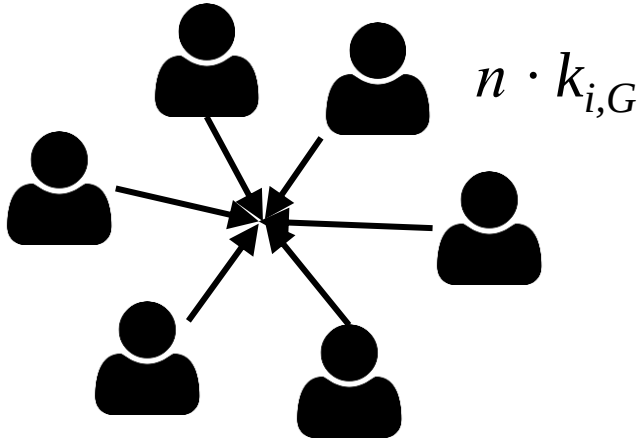
# Group Communication

- Pairwise Channels



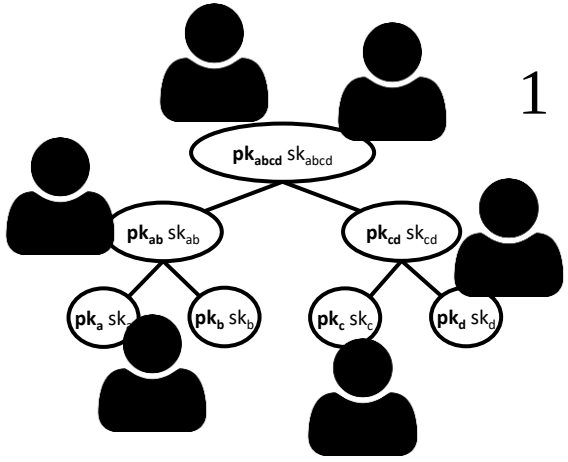
$n^2 \cdot k_{i,j}$

- Sender Key



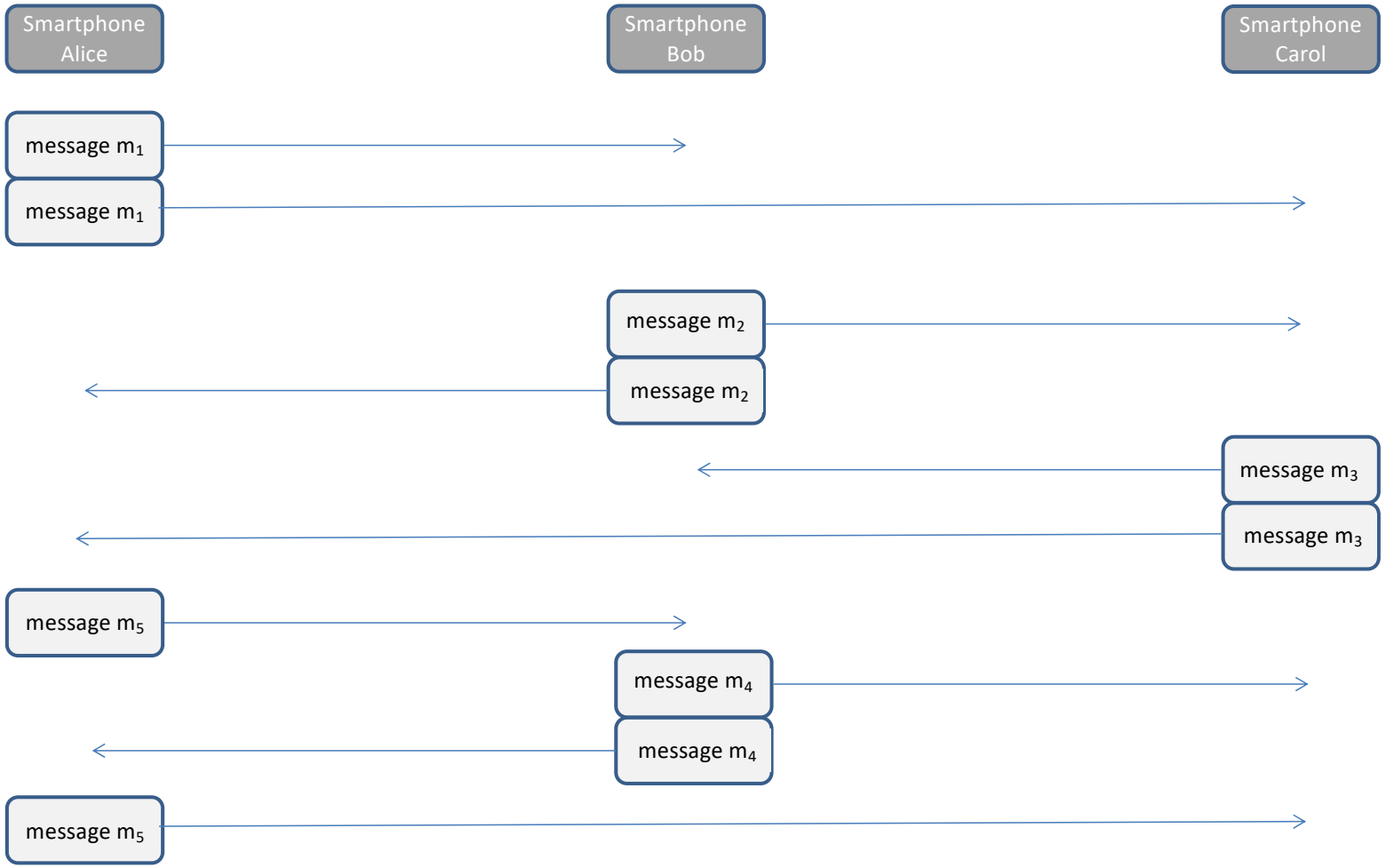
$n \cdot k_{i,G}$

- Group Key

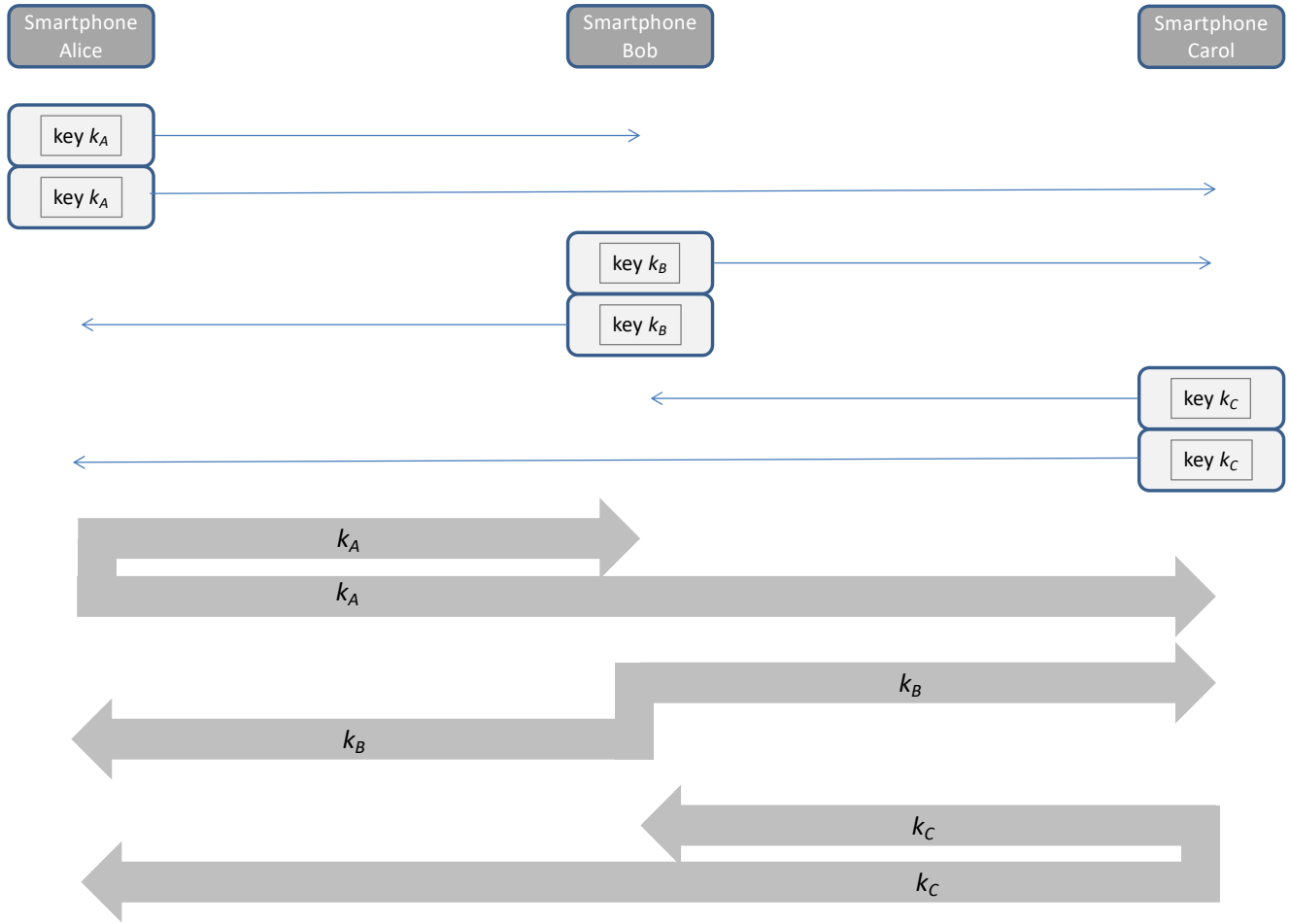



$1 \cdot grk$

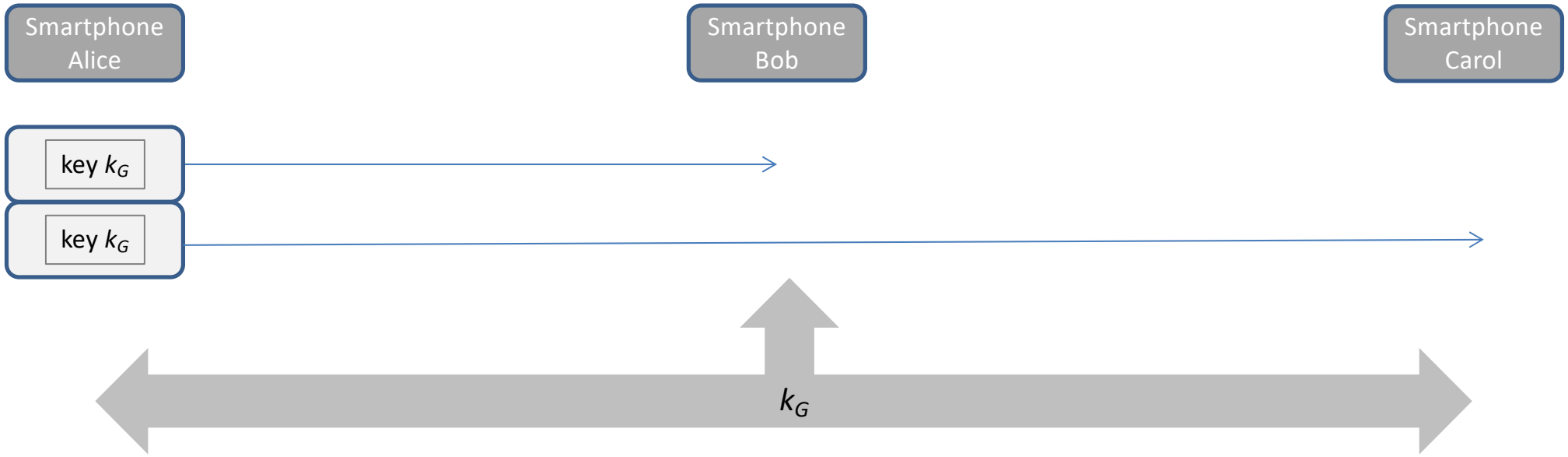
# Group Communication



# Group Communication

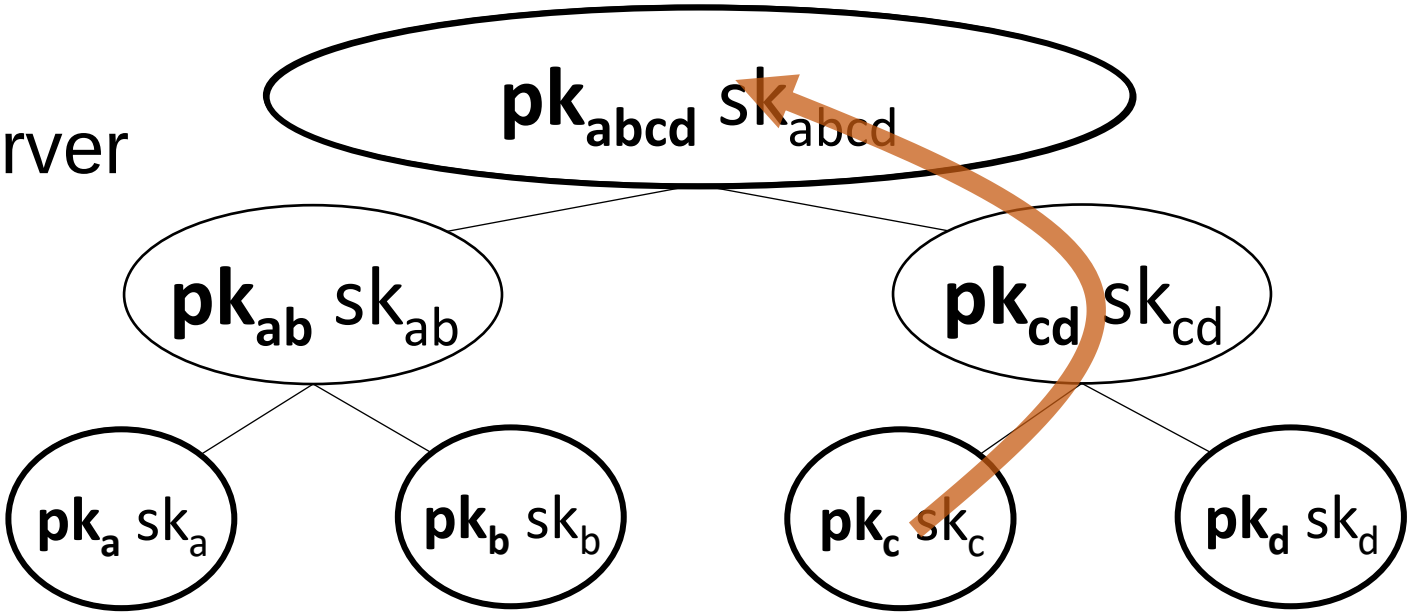


# Group Communication



# IETF MLS

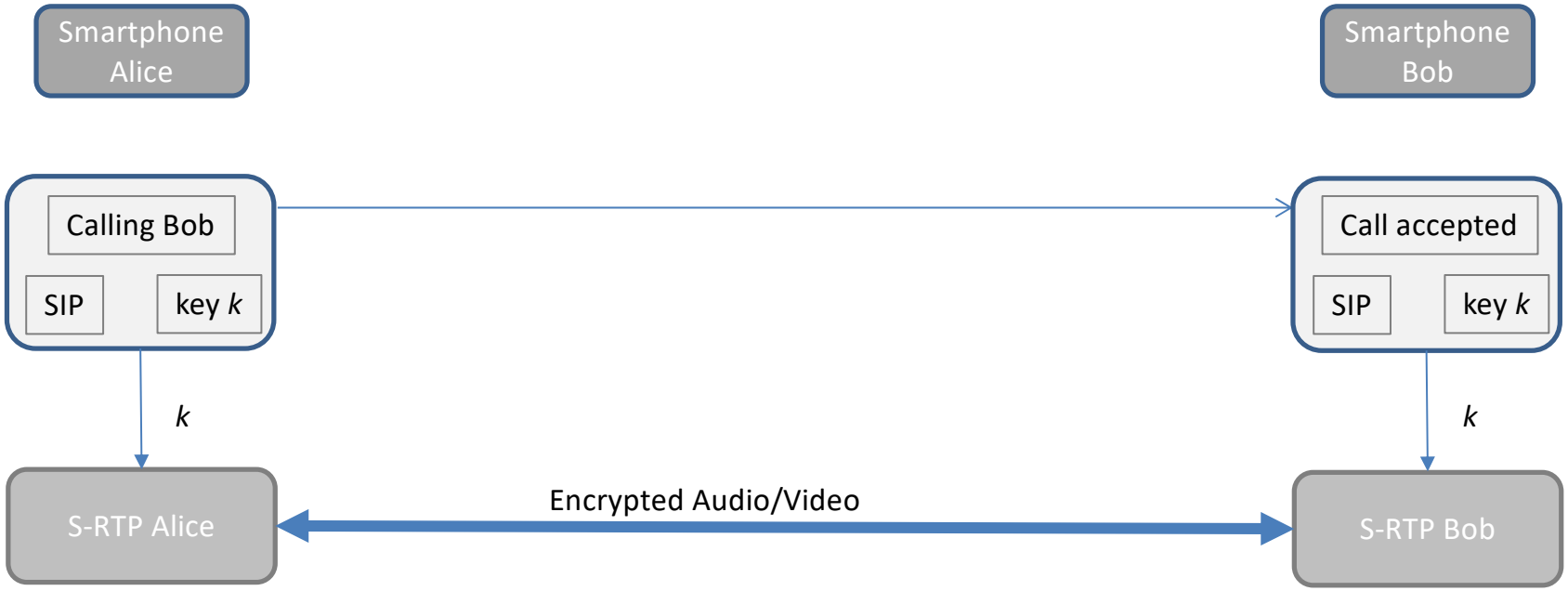
- IETF Standard
- Security similar to Double Ratchet
- More efficient
- Needs synchronization server



# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# Real-Time Communication



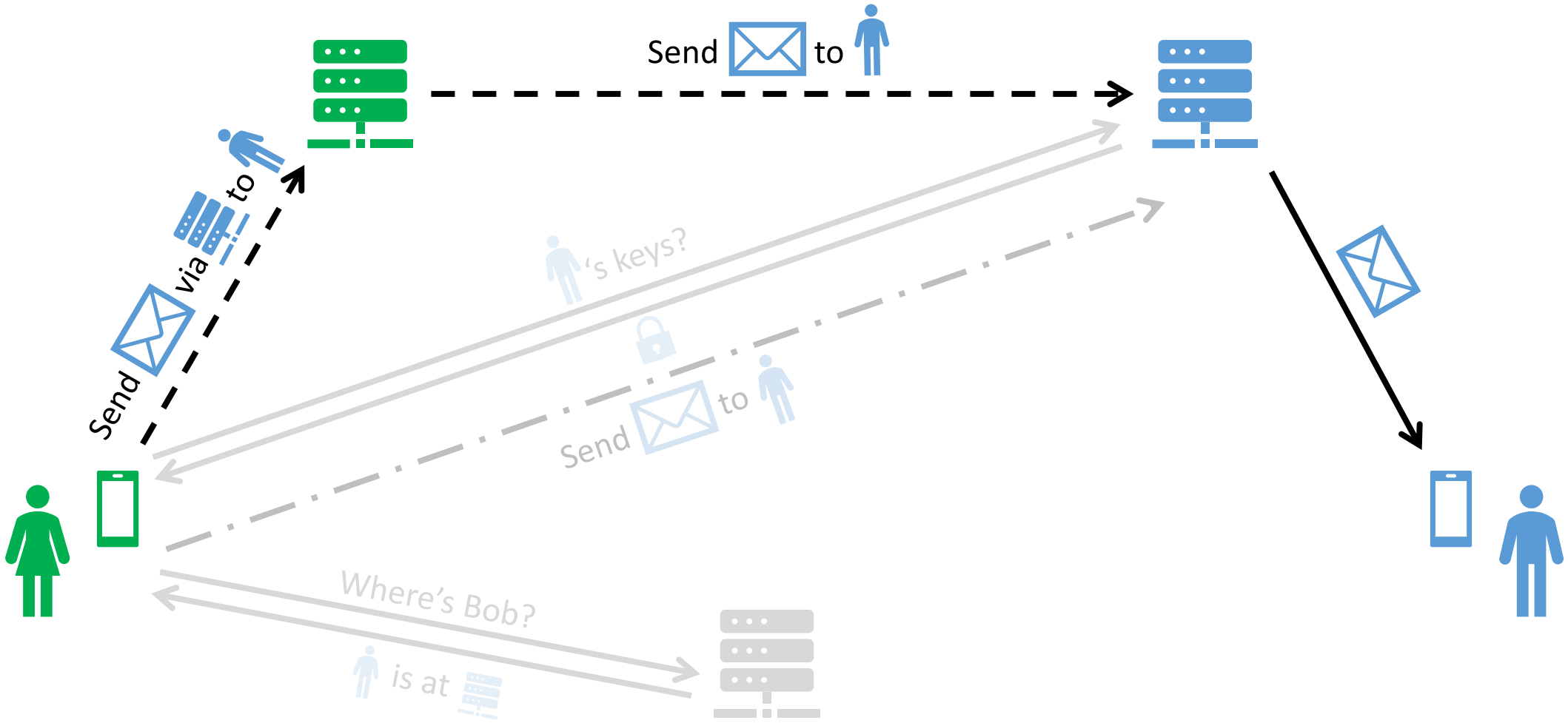
# Overview of Messengers

| IM Protocol | Two-Party | Group | Real Time |
|---|---|---|---|
| **Signal** | Double Ratchet (**DR**) | **DR** | WebRTC |
| **WhatsApp** | **DR** | Sender Key (**SK**) | SRTP |
| **Facebook Messenger** | **DR** with Message Franking | **SK** with Message Franking | Undocumented |
| **Wire** | Proteus (≈**DR**; diff. AE) | Proteus (≈**DR**) | SRTP |
| **Matrix** | Olm (≈**DR**; diff. KDF) | Megolm (≈**SK**) | WebRTC |
| **iMessage** | Public-key encryption | Public-key encryption | SRTP |
| **Telegram** | MTProto | Unencrypted | MTProto |

# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# Overview of Setting



# Requirements

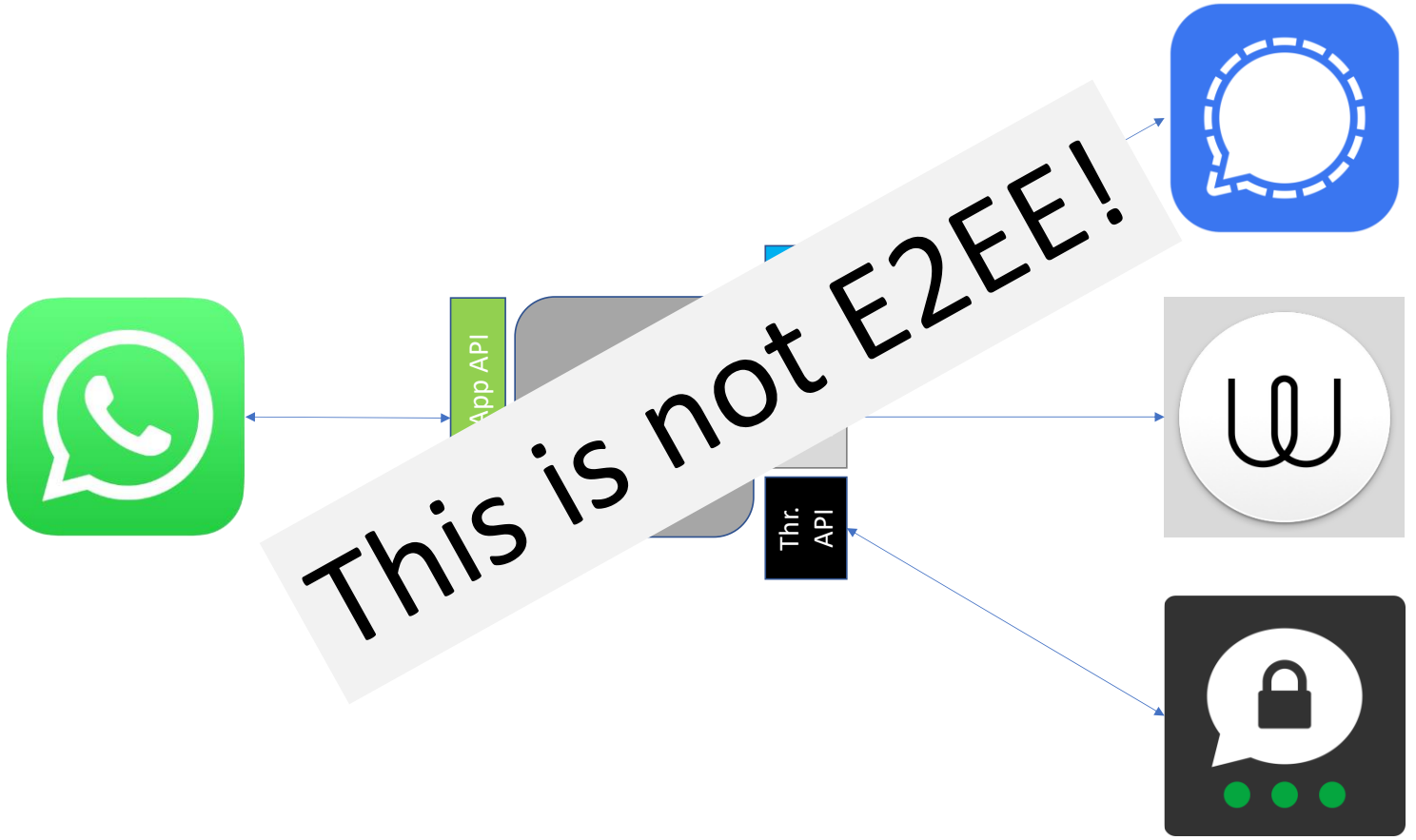
**<u>Functional Interop:</u> Two-Party, Groups, Real Time**

**<u>End-to-End Confidentiality:</u>**
"*The **level of security**,*
*including the **end-to-end encryption**, [...]*
***shall be preserved** across the interoperable services.*" §3

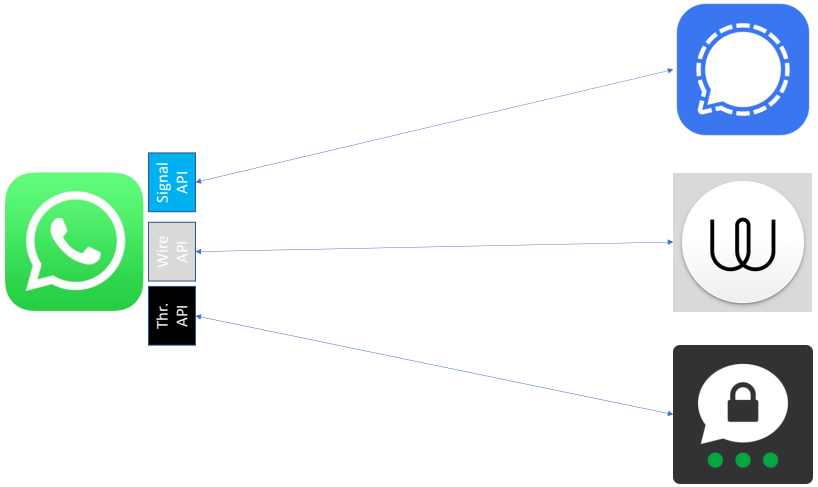
**<u>Metadata Protection:</u>**
"*The gatekeeper shall **collect and exchange** [...]*
*only the **personal data** of end users*
*that is **strictly necessary** [...].*" §8

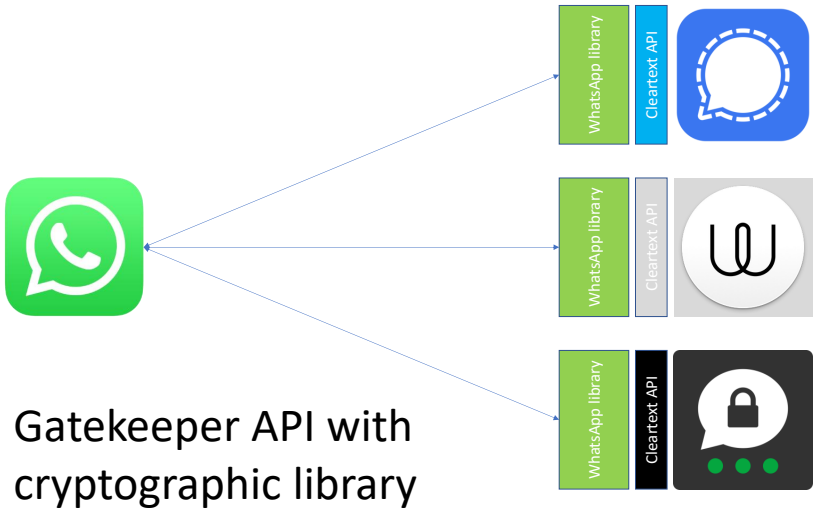
**<u>Abuse Prevention:</u>**
"*The gatekeeper [should be able to **take**] **measures***
*to [...] **not endanger***
*the **integrity, security and privacy** of its services [...].*" §9

# Where to ‘translate’ E2EE?

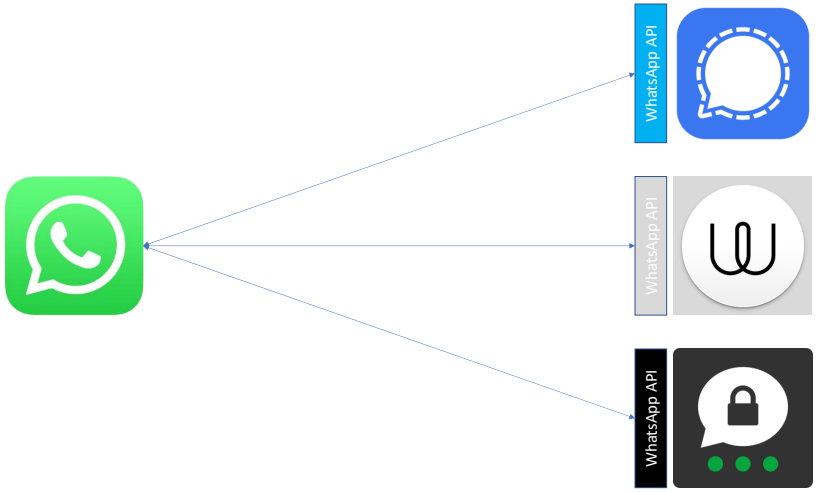


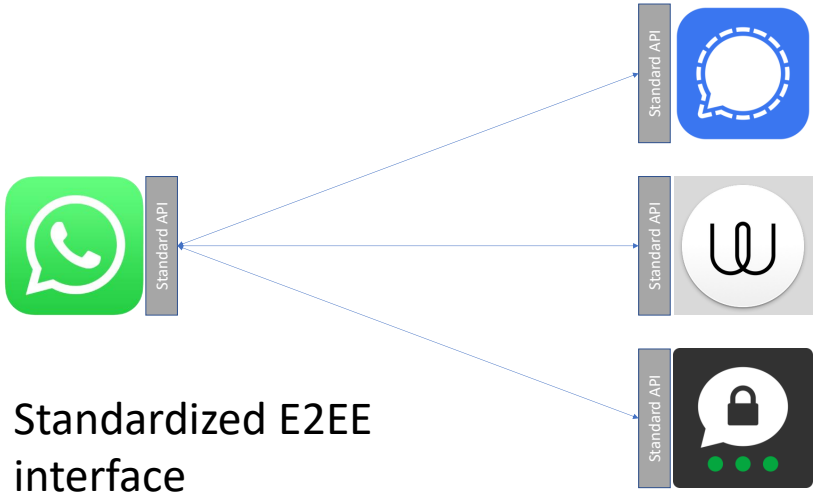
# Where to ‘translate’ E2EE?




Gatekeeper API with cryptographic library



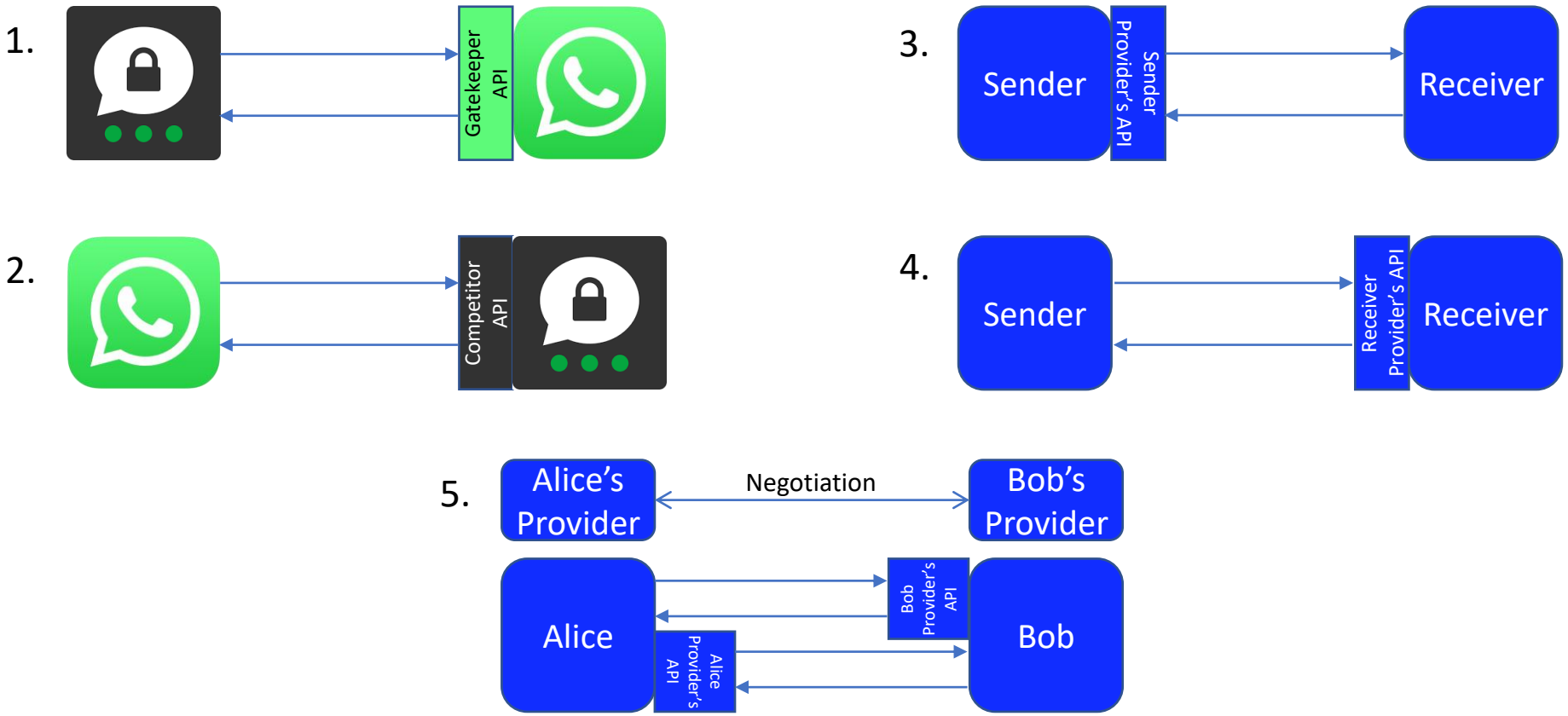




Standardized E2EE interface

# Agenda

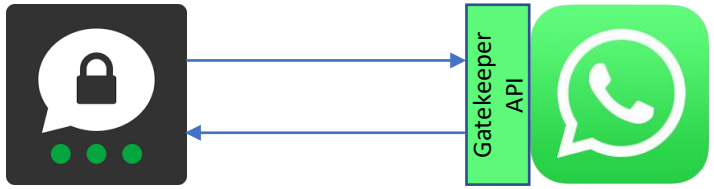
- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

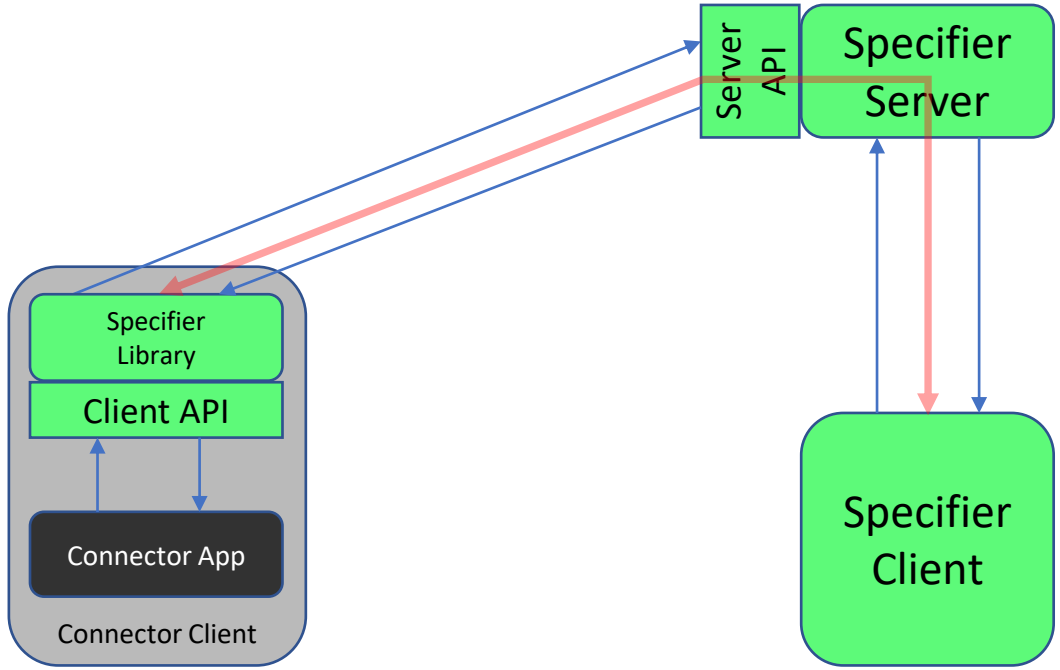
# Location of API



# Location of API: Client & Server

1.



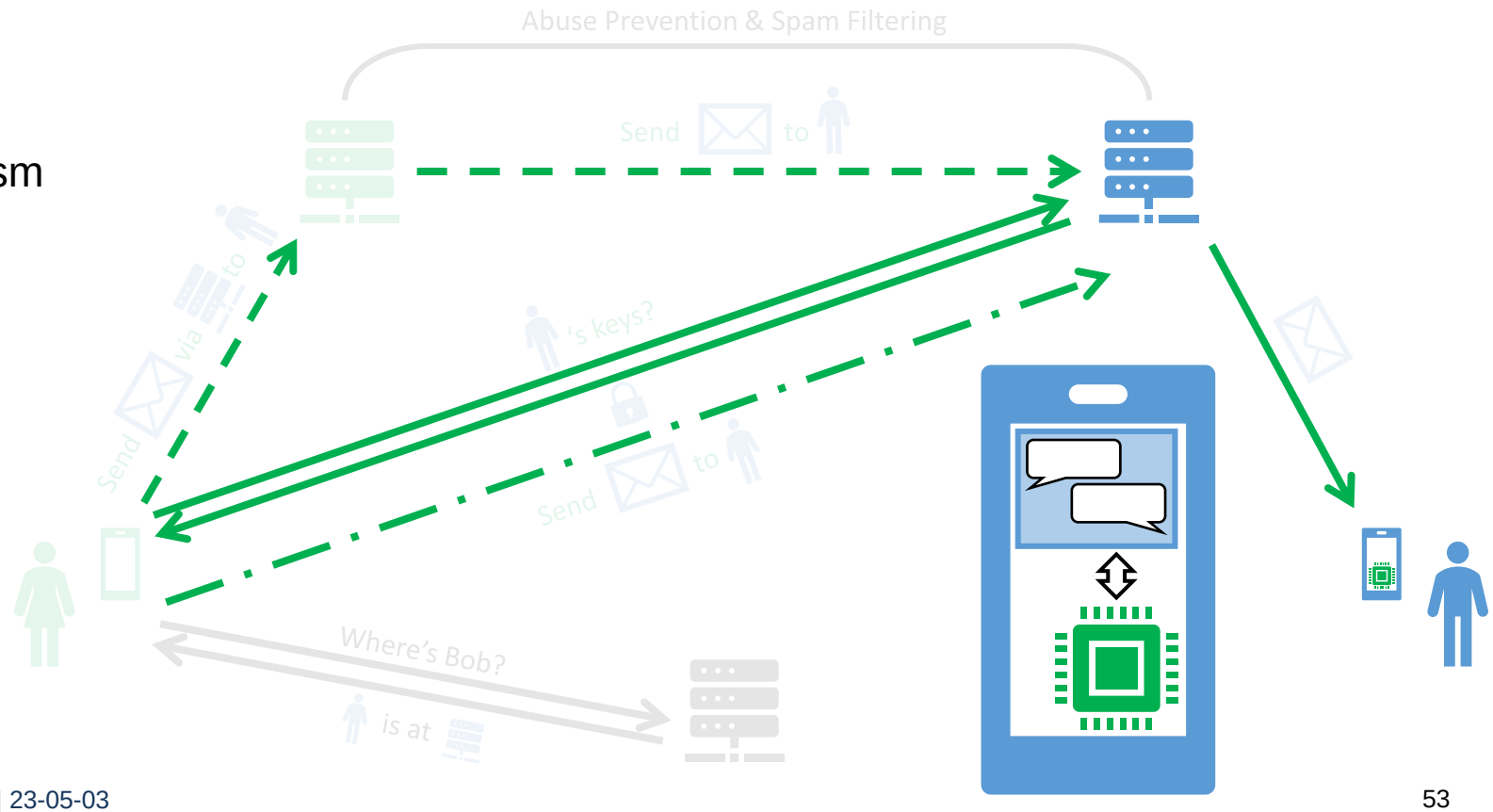
# Communication Protocol

1. Gatekeeper‘s Core Protocol + X

2. Standardization of Protocol

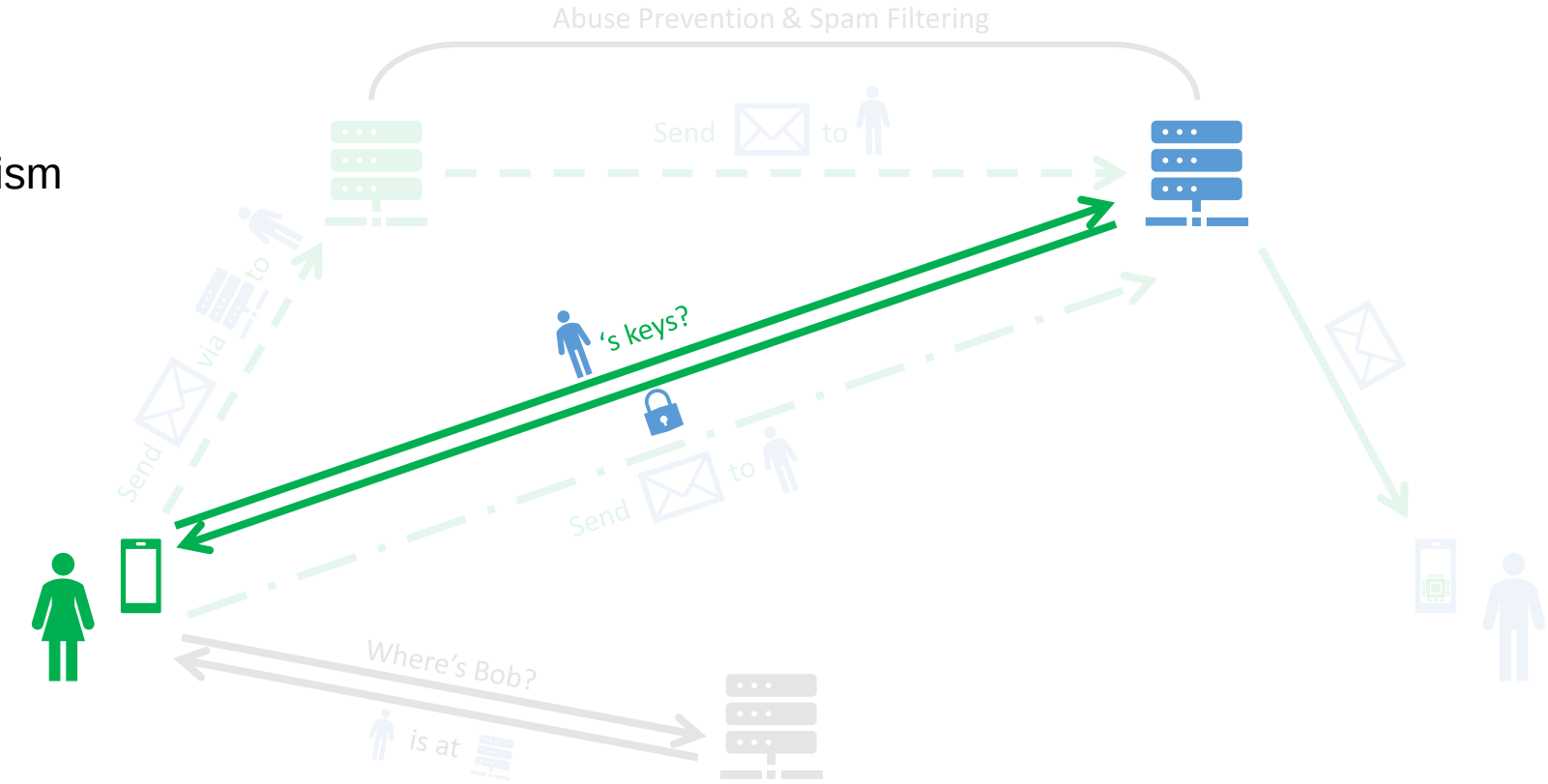
# Communication Protocol

1. Gatekeeper‘s Core Protocol + X
   - API + Documentation
   - Client: Library?
   - Server:
     - Replicate key distribution
     - Forwarding service
     - Abuse prevention (interactively)
     - Reporting and blocking mechanism
2. Standardization of Protocol



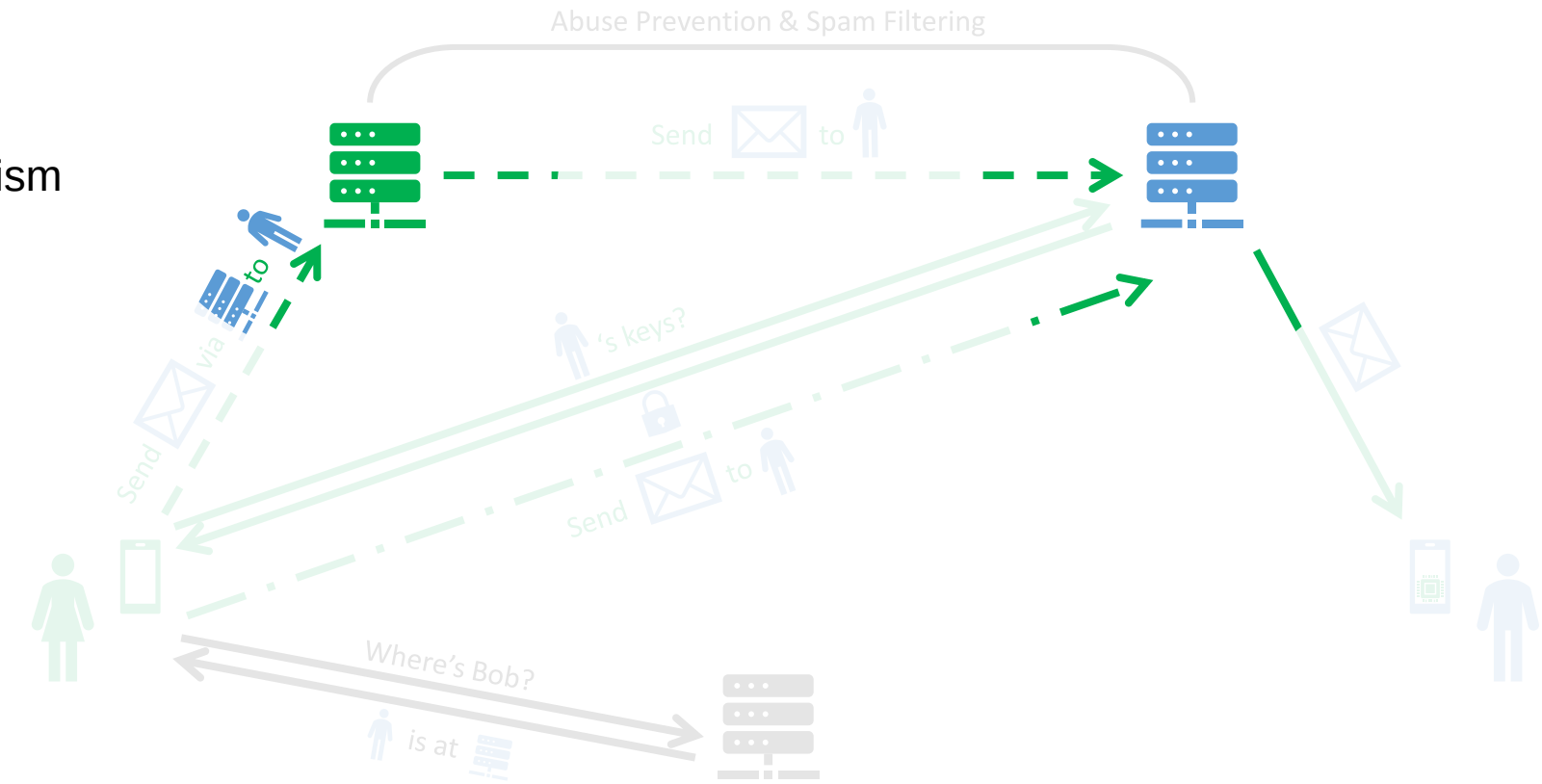
# Communication Protocol

1. Gatekeeper‘s Core Protocol + X
   - API + Documentation
   - Client: Library?
   - Server:
     - Replicate key distribution
     - Forwarding service
     - Abuse prevention (interactively)
     - Reporting and blocking mechanism
2. Standardization of Protocol



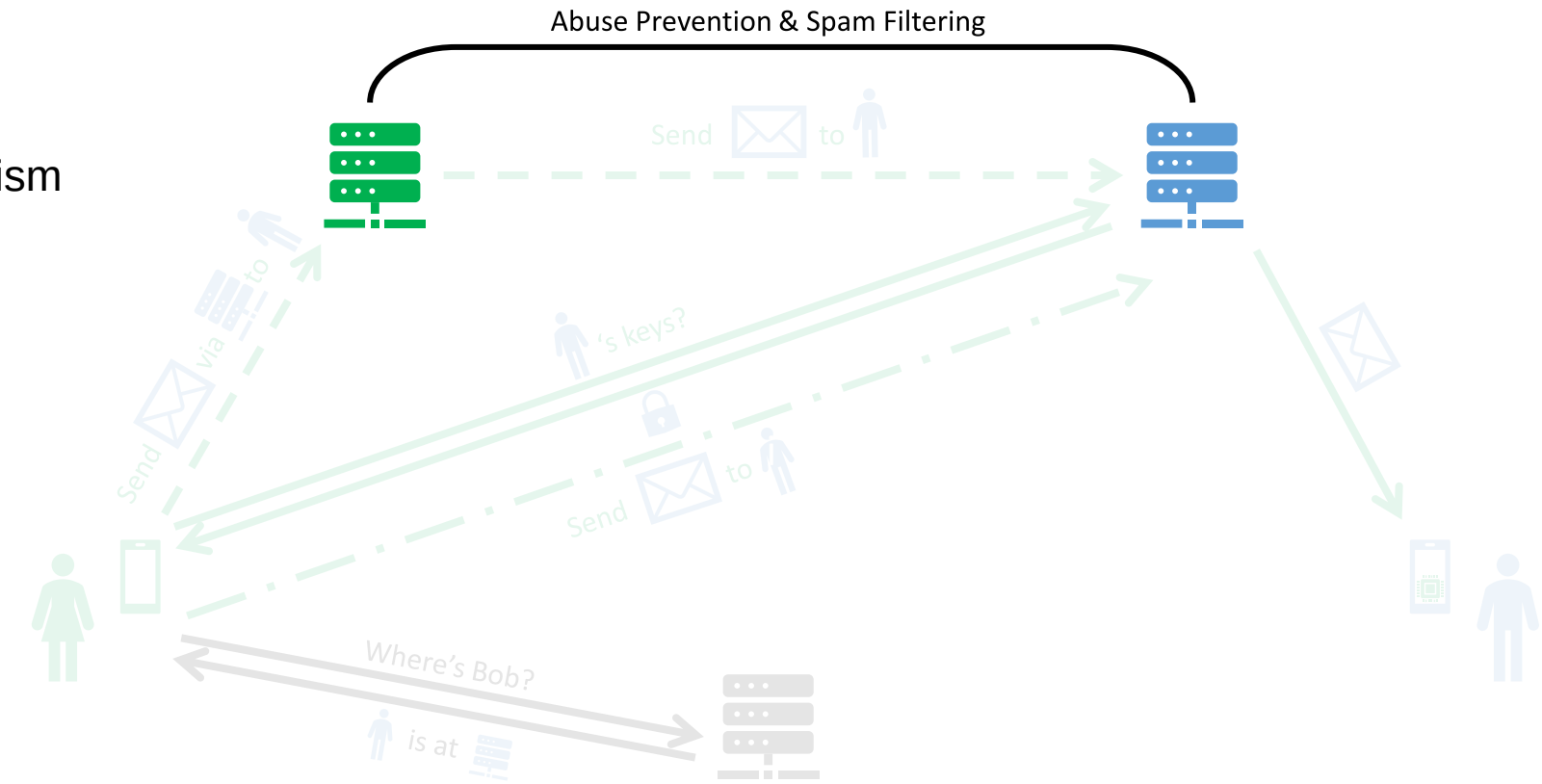
# Communication Protocol

1. Gatekeeper‘s Core Protocol + X
   - API + Documentation
   - Client: Library?
   - Server:
     - Replicate key distribution
     - Forwarding service
     - Abuse prevention (interactively)
     - Reporting and blocking mechanism
2. Standardization of Protocol



# Communication Protocol

1. Gatekeeper‘s Core Protocol + X
   - API + Documentation
   - Client: Library?
   - Server:
     - Replicate key distribution
     - Forwarding service
     - Abuse prevention (interactively)
     - Reporting and blocking mechanism
2. Standardization of Protocol

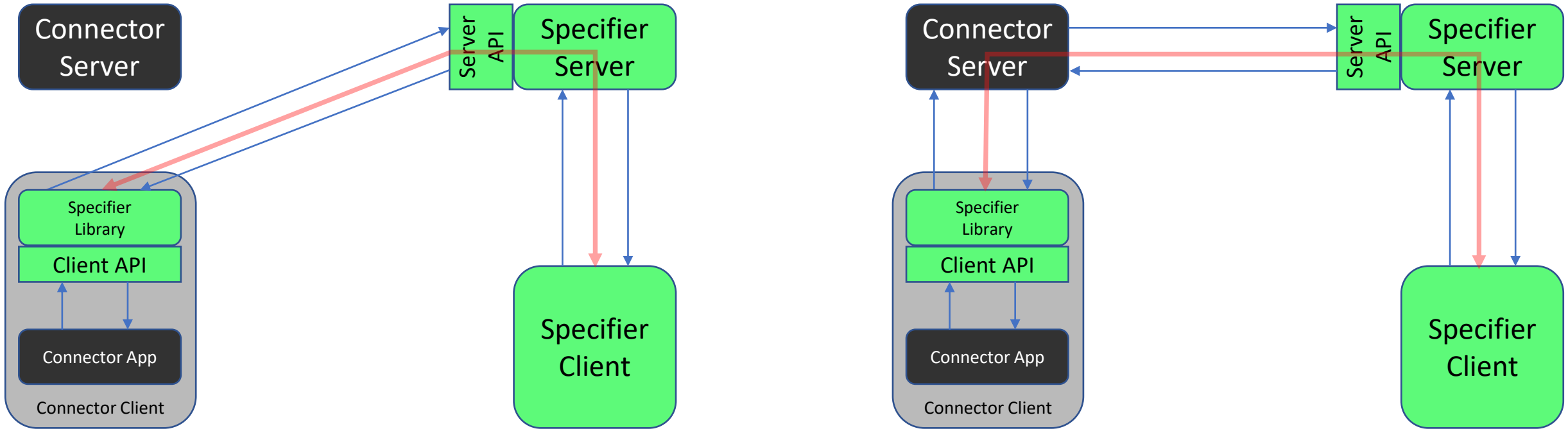


# Communication Protocol

1. Gatekeeper‘s Core Protocol + X
   - API + Documentation
   - Client: Library?
   - Server:
     - Replicate key distribution
     - Forwarding service
     - Abuse prevention (interactively)
     - Reporting and blocking mechanism
2. Standardization of Protocol
   - New protocol
   - ≥ Best gatekeeper
   - Simple standard update
   - Individual abuse prevention

   → Don‘t start with solution

# Transmission Path

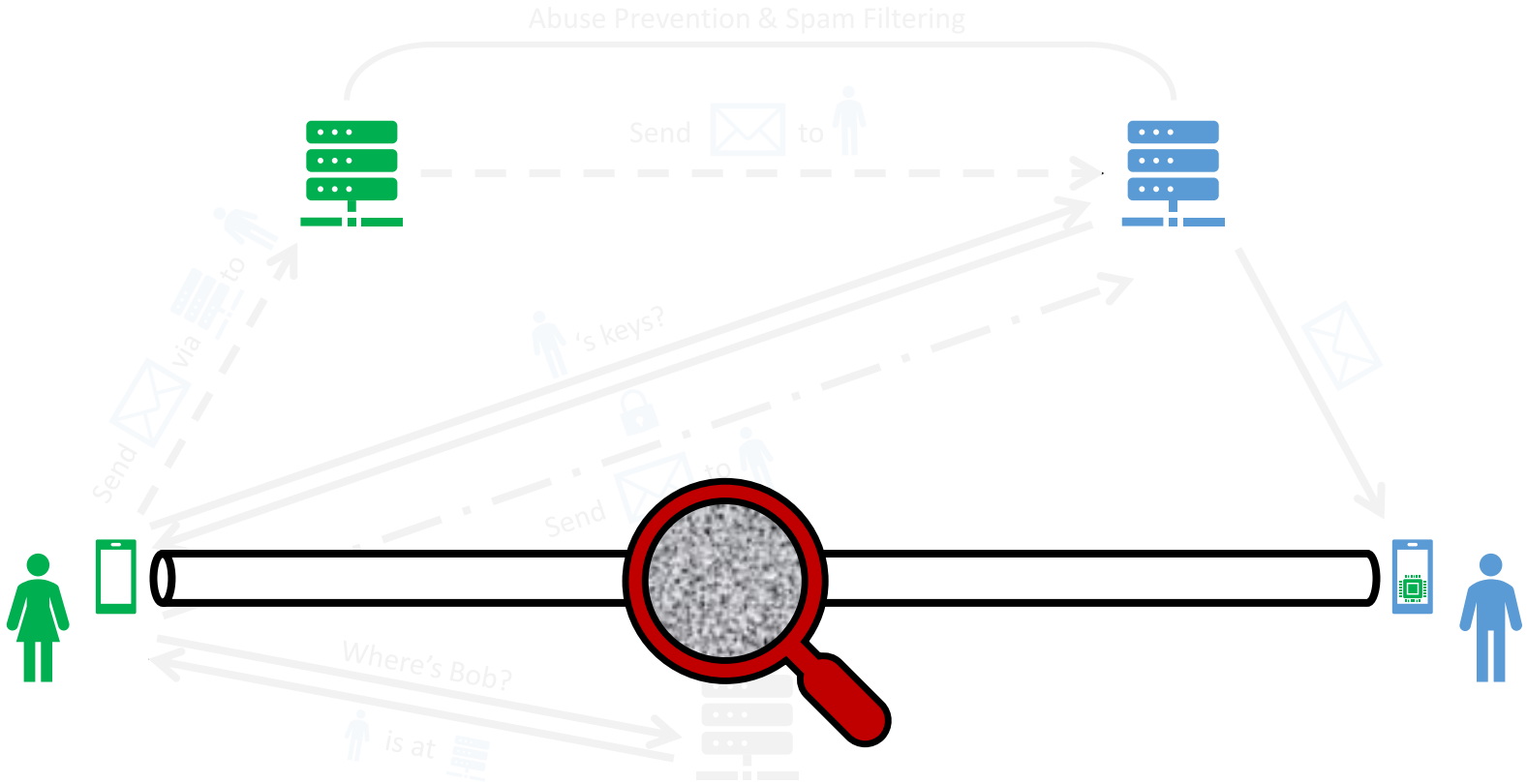


# Confidentiality, Privacy & Abuse Prevention

Aka. “+ X”

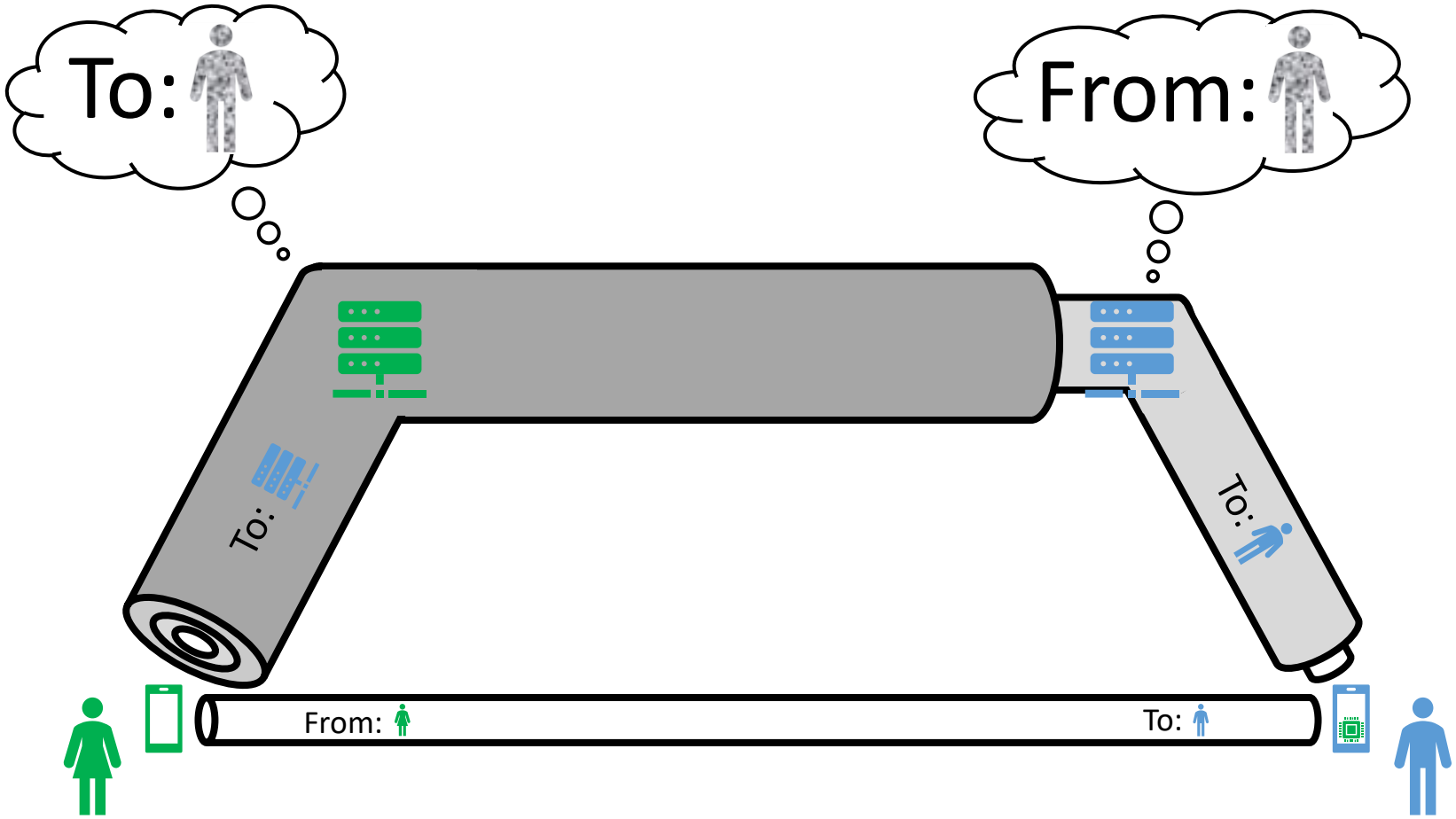
# Confidentiality, Privacy & Abuse Prevention

- Message Delivery
  - Confidential + Y



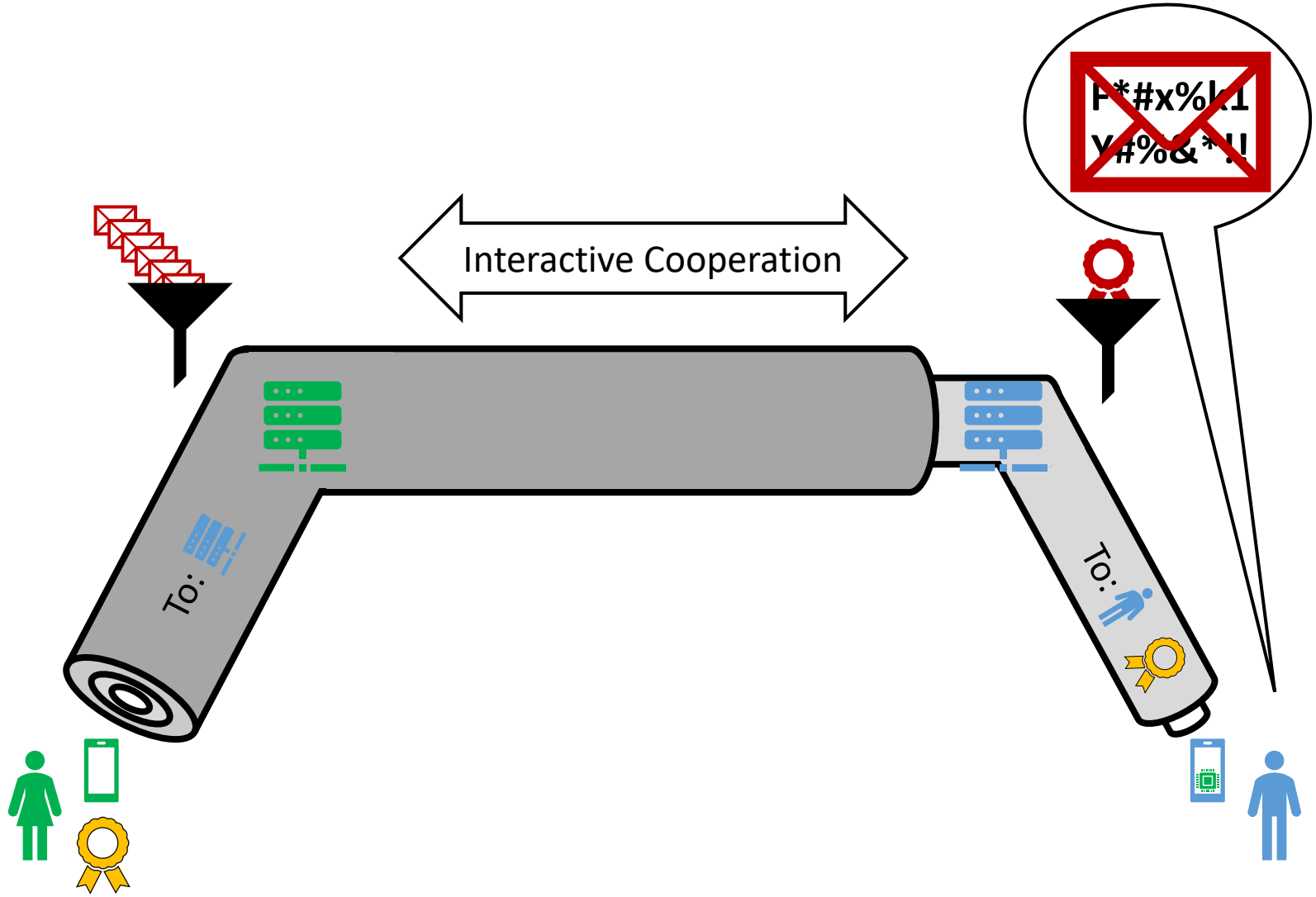
# Confidentiality, Privacy & Abuse Prevention

- Message Delivery
  - Confidential + Y
  - Private



# Confidentiality, Privacy & Abuse Prevention

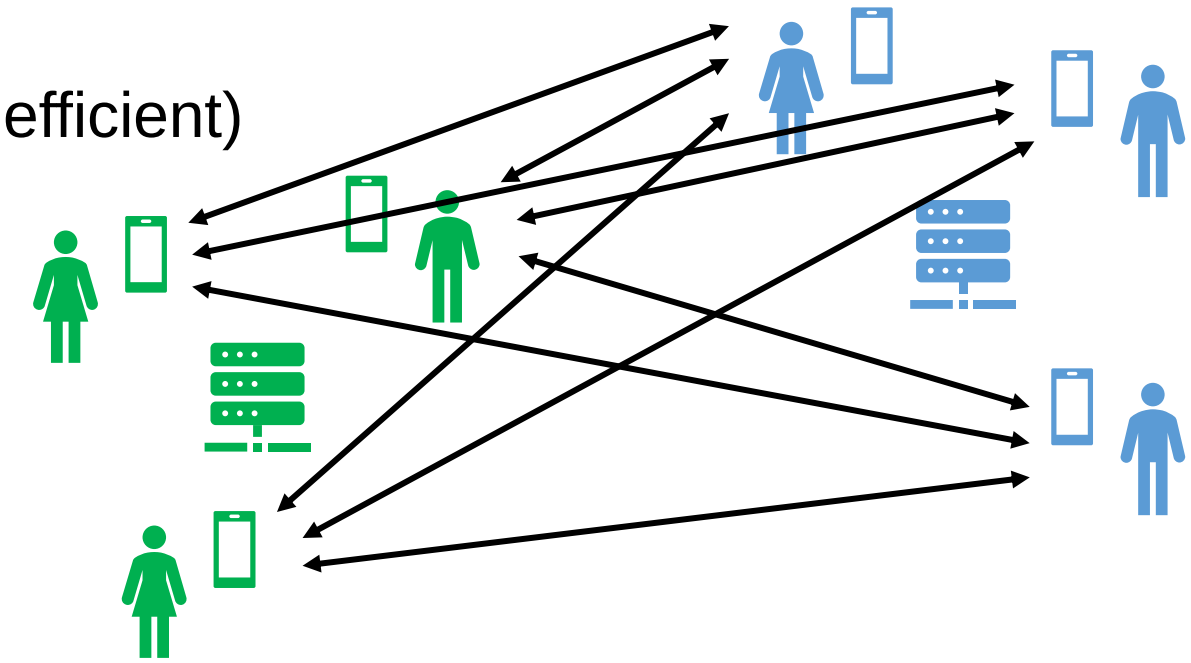
- Message Delivery
  - Confidential + Y
  - Private
- Abuse Reporting
- Effective User Blocking
  - For Individuals
  - From Platform
- Spam Filtering



# Group Messaging

Alternatives:

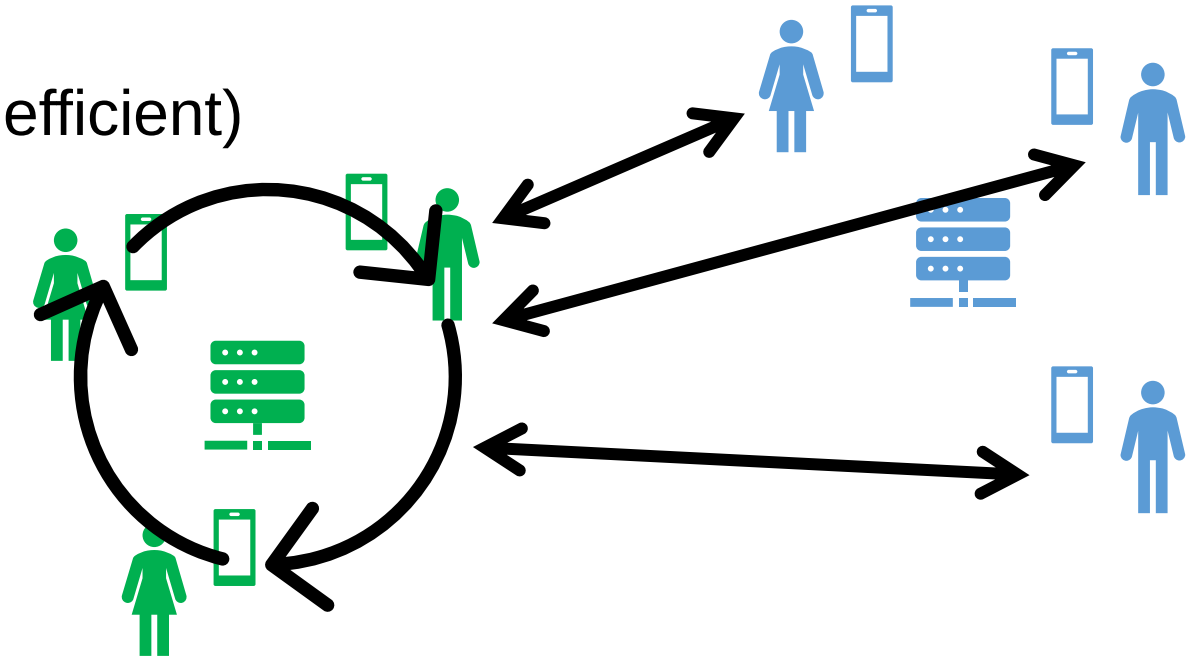
1. Via Pair-Wise Channels (Simple but Inefficient)
2. Gatekeeper's Core Protocol
3. Connect Providers' Subgroups
4. Standardize Group Protocol



# Group Messaging

Alternatives:

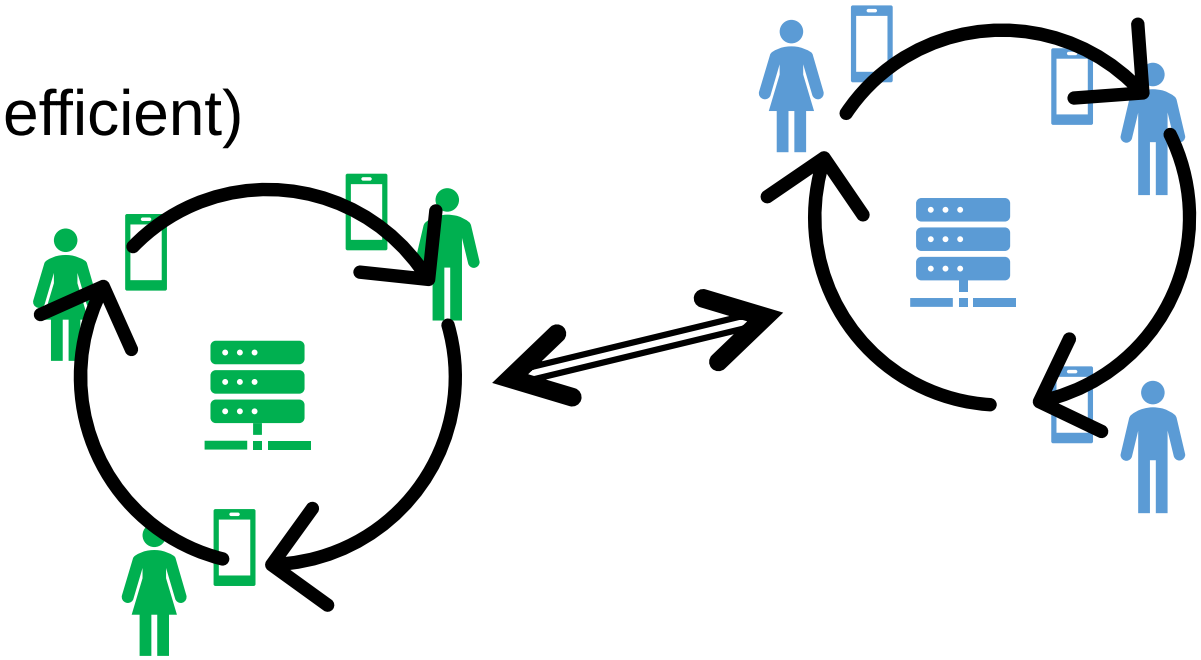
1. Via Pair-Wise Channels (Simple but Inefficient)
2. Gatekeeper's Core Protocol
3. Connect Providers' Subgroups
4. Standardize Group Protocol



# Group Messaging

Alternatives:

1. Via Pair-Wise Channels (Simple but Inefficient)
2. Gatekeeper's Core Protocol
3. Connect Providers' Subgroups
4. Standardize Group Protocol



# Group Messaging

Alternatives:

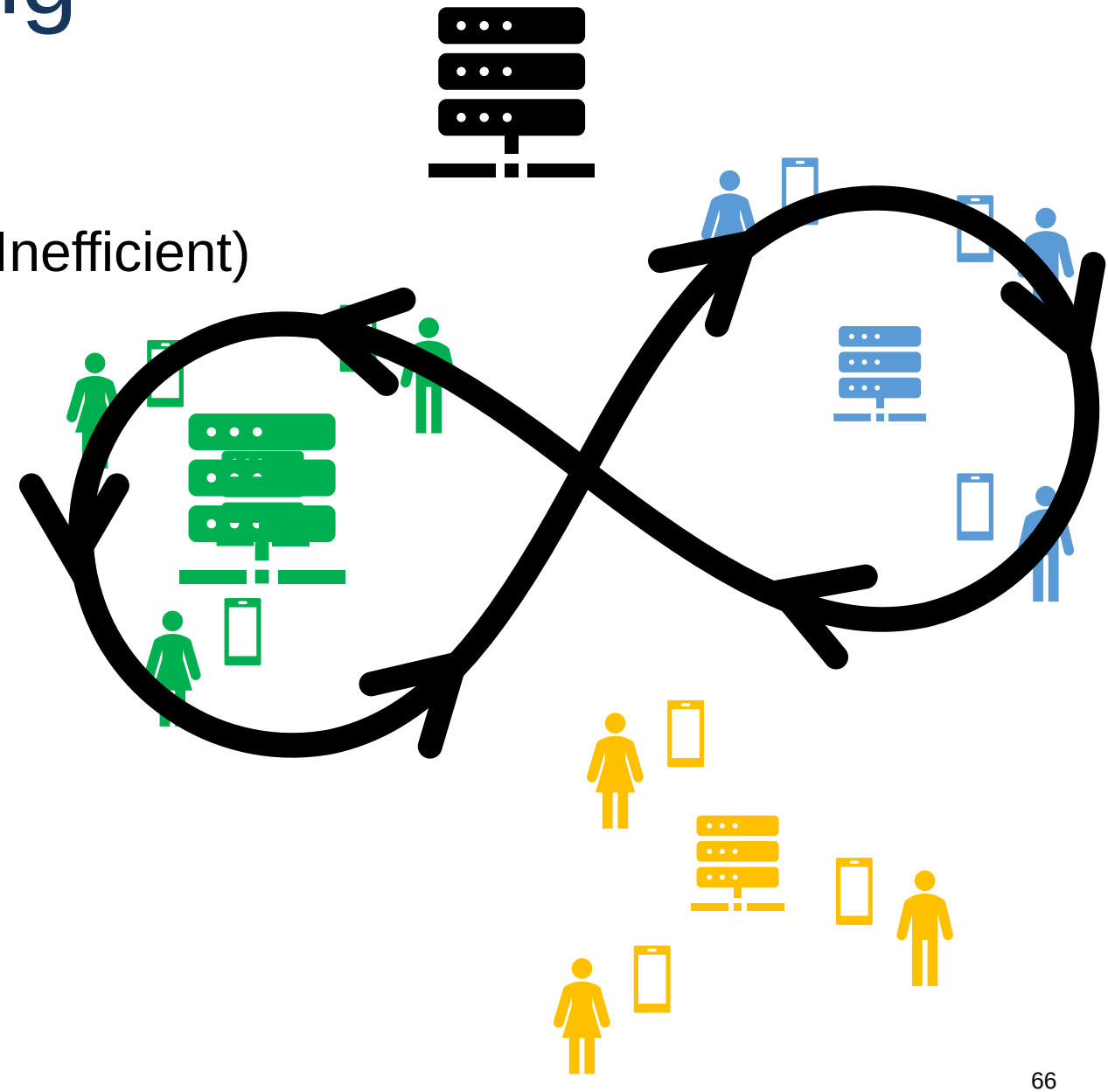
1. Via Pair-Wise Channels (Simple but Inefficient)
2. Gatekeeper's Core Protocol
3. Connect Providers' Subgroups
4. Standardize Group Protocol

Problems and Solutions:

- Consistent Group Management:

→ Centralized?
→ Where?

- Multiple Gatekeepers & Non-Gatekeepers



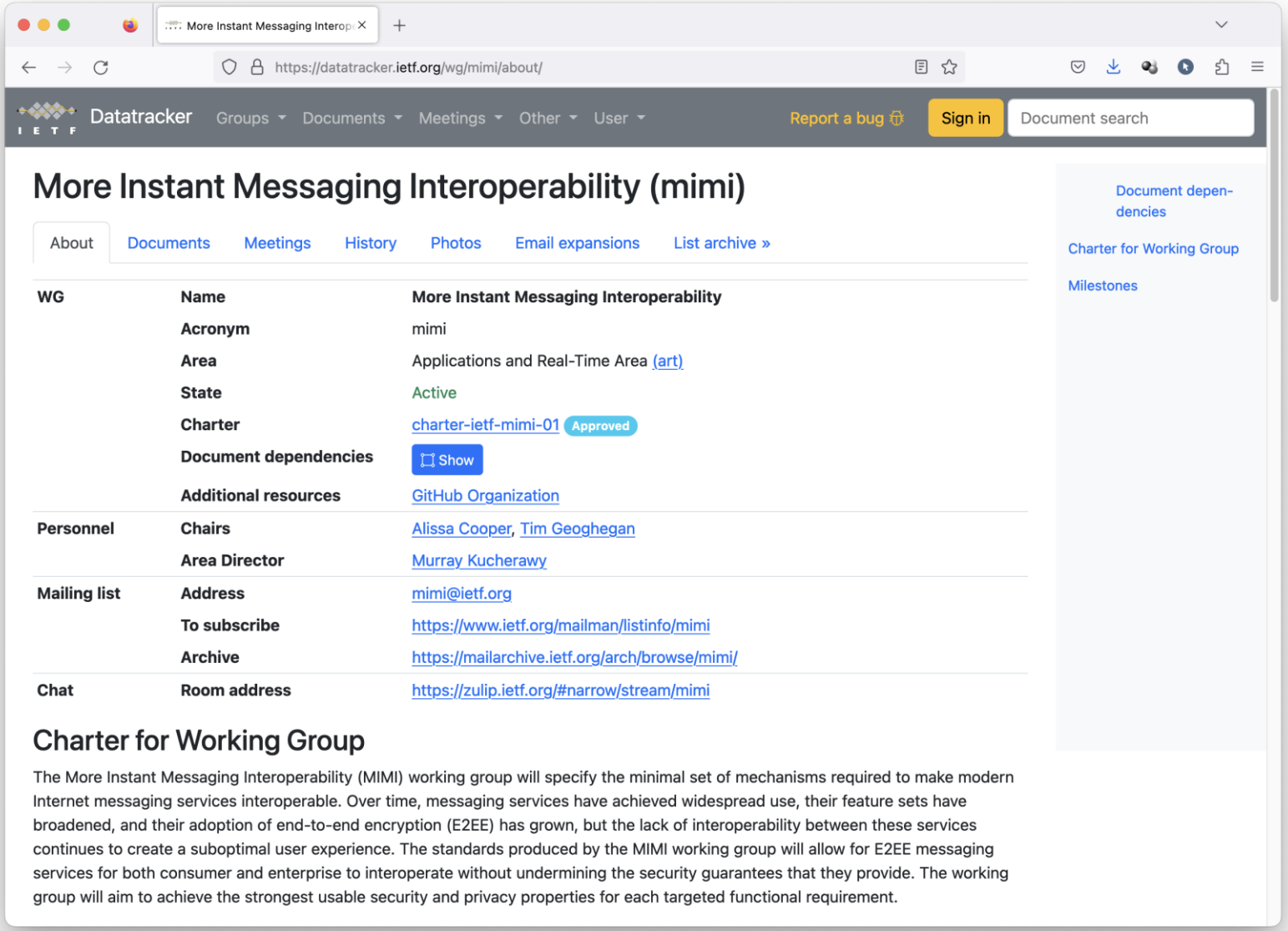
# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# IETF MIMI

*„The More Instant Messaging Interoperability (MIMI) working group will specify the minimal set of mechanisms required to make modern Internet messaging services interoperable.“*

*„The working group will aim to achieve the strongest usable security and privacy properties for each targeted functional requirement.“*



# Standardization Goal

*Re-use existing standards whenever possible*

# Identities, Key Distribution, Trust

- Standardize naming scheme similar to e-mail:

# Identities, Key Distribution, Trust

- Standardize initial authentication:
  - IM providers use long-lived signature or X3DH keys to authenticate users
  - Signature key can be adapted to any authentication scheme
    - If signatures are used to authenticate users, use them directly
    - If X3DH is used, sign a long-lived X3DH key of the user

- Standardize signature schemes for key distribution and Trust establishment

# Text messaging

| IM Protocol | Two-Party | Group | Real Time |
|---|---|---|---|
| **Signal** | Double Ratchet (**DR**) | **DR** | WebRTC |
| **WhatsApp** | **DR** | Sender Key (**SK**) | SRTP |
| **Facebook Messenger** | **DR** with Message Franking | **SK** with Message Franking | Undocumented |
| **Wire** | Proteus (≈**DR**; diff. AE) | Proteus (≈**DR**) | SRTP |
| **Matrix** | Olm (≈**DR**; diff. KDF) | Megolm (≈**SK**) | WebRTC |
| **iMessage** | Public-key encryption | Public-key encryption | SRTP |
| **Telegram** | MTProto | Unencrypted | MTProto |

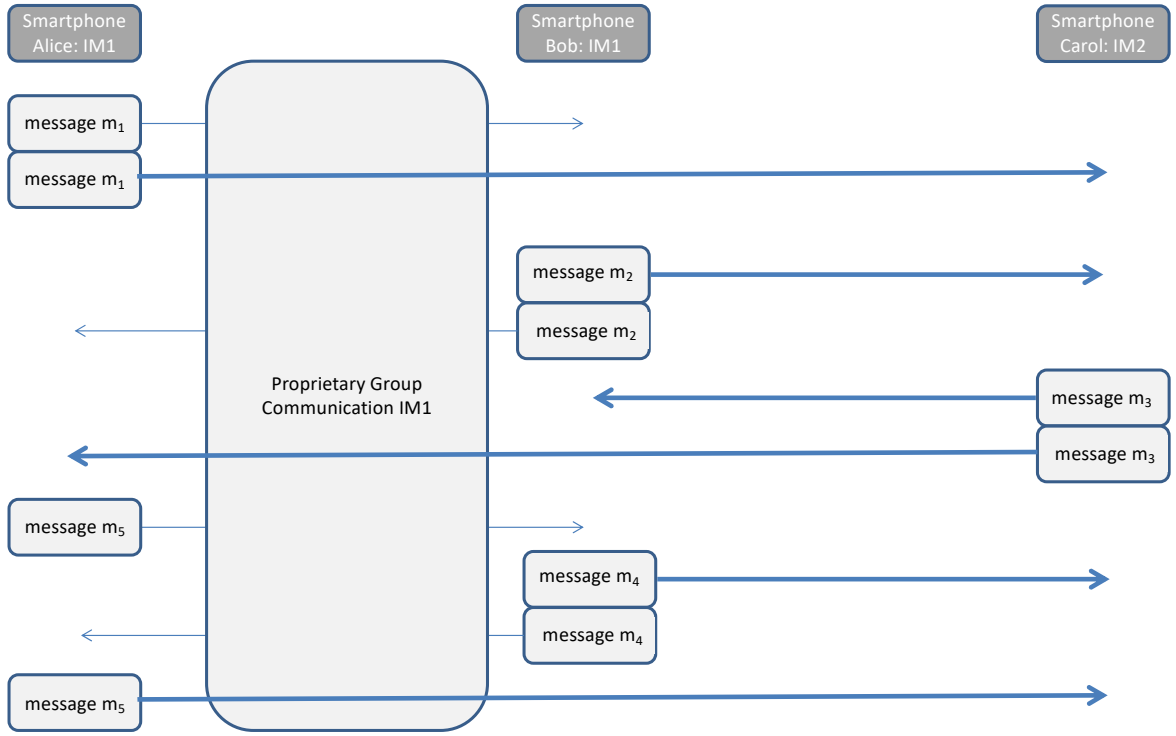
- Double Ratchet de facto standard for key management
  - many details differ and need to be standardized
- MLS may become an alternative once it is deployed

# File transfer

- General mechanism similar across IM applications
- Need for standardization:
  - Data format of encrypted files, e.g., JSON Web Encryption
  - Data format for download URL and decryption key in text message

# Group Messaging

- MLS, once it is successfully deployed
- Individual text message channels for competitor members in gatekeeper groups



# Real-Time Communication

- IETF WebRTC to enable browser-to-browser audio and video calls
  - Signaling channel over Web Application
  - Key exchange via DTLS
  - Encrypted communication via SRTP
- Adaption to IM (already done by IM providers)
  - Signaling and key exchange can be done via text chats
  - SRTP seems a good choice for encrypted communication

# Summary of Standardization

- Recommended standardization (short term)
  - Naming scheme for IM clients
  - Initial authentication via digital signatures
- Optional standardization (long term)
  - E2EE for text messages (double ratchet or MLS)
  - File transfer (data formats)
  - Group messaging (double ratchet or MLS)
  - Real-time communications (SRTP plus standardization of signaling and key exchange)

# Agenda

- Instant Messaging: Shared Cryptographic Mechanisms
  - Authenticated Encryption
  - Key Agreement
- Instant Messaging: Novel Cryptographic Mechanisms
  - Text messaging
  - File transfer
  - Group communication
  - Real-time communication
- Interoperable Instant Messaging
  - API Approach
  - Standardization Approach
- Summary

# Summary

- E2EE Confidentiality & Privacy:
  - Required by DMA!
  - Practically achievable through APIs and/or standardization ✓
- Gatekeeper API vs. IM Standard
  - Standardization:
    - Slow
    - Equal overhead for all parties
  - Gatekeeper API:
    - Fast, agile
    - Cryptographic library provided by the gatekeeper
  - Approach
    - Start with API, standardize only basic functionality

# Summary: Approach

- Minimal initial standardization: Plaintext formats, ID scheme
- Gatekeeper libraries
- Clear documentation
- Gradual additional standardization
- Agile, flexible standard & space for individual protocols

Missing aspects:

- Data migration
- Update mechanisms
- Adversarial providers (e.g., complicate interop)