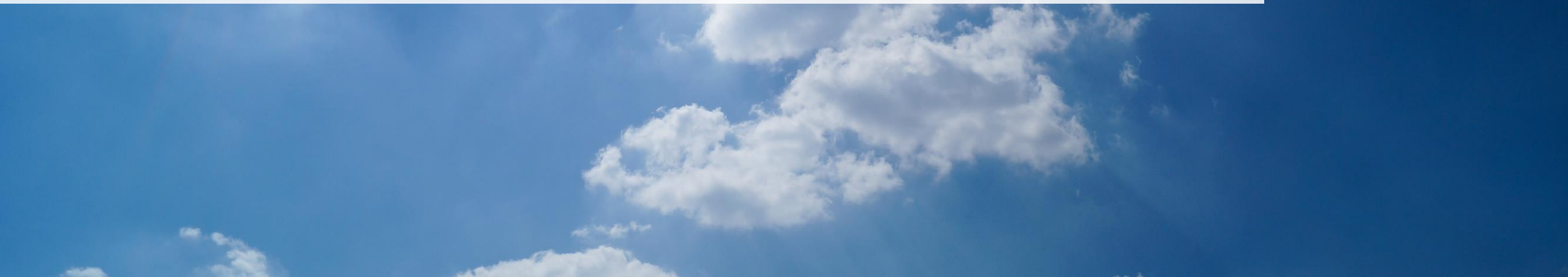
Strongly Anonymous Ratcheted Key Exchange



12/8/2022

FAU Erlangen-Nürnberg, Germany

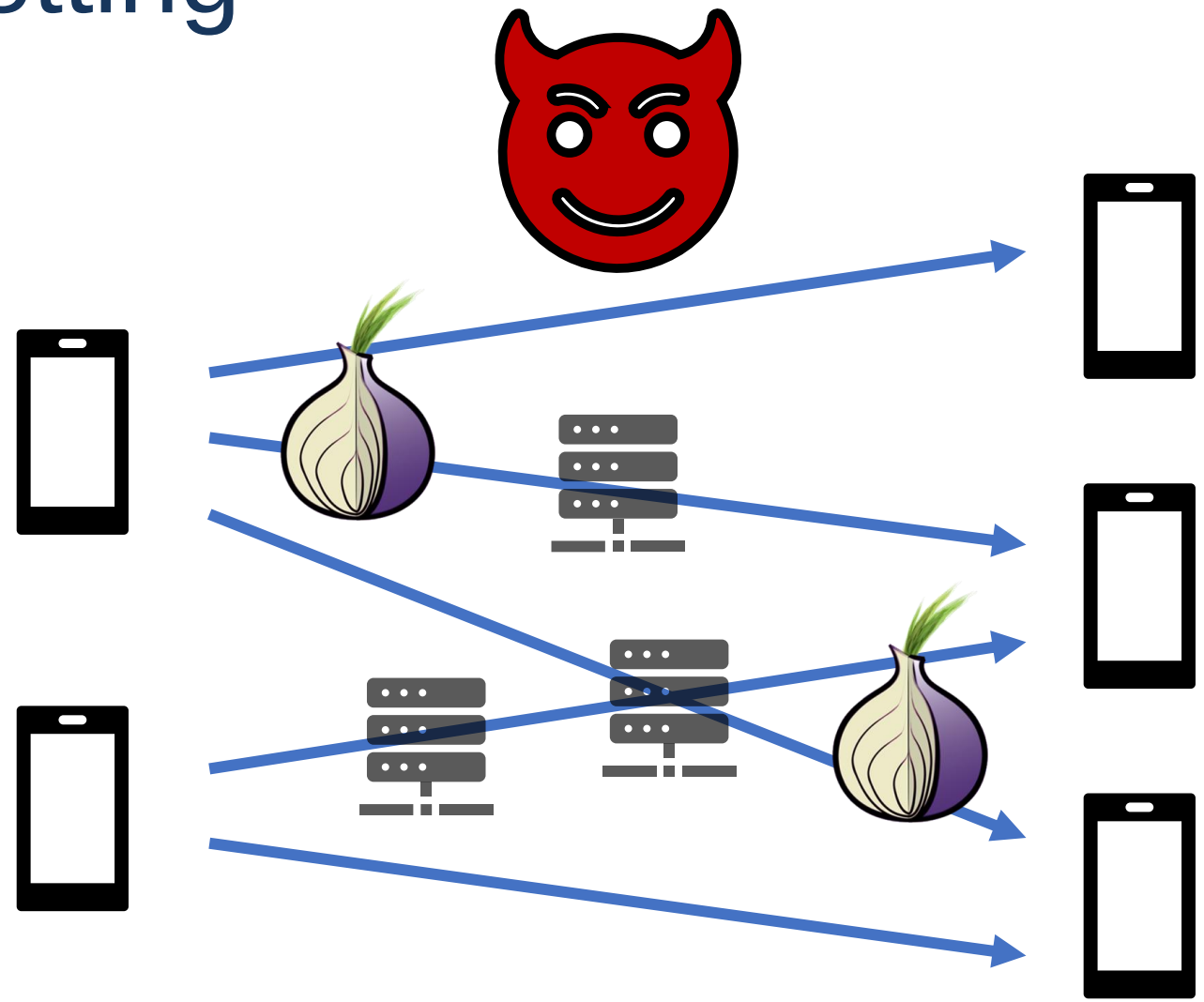
Benjamin Dowling, Eduard Hauck (get well soon!), Doreen Riepel, and Paul Rösler

Messaging Setting

- Multiple Users
- Multiple Sessions/User

- Unidirectional Communication

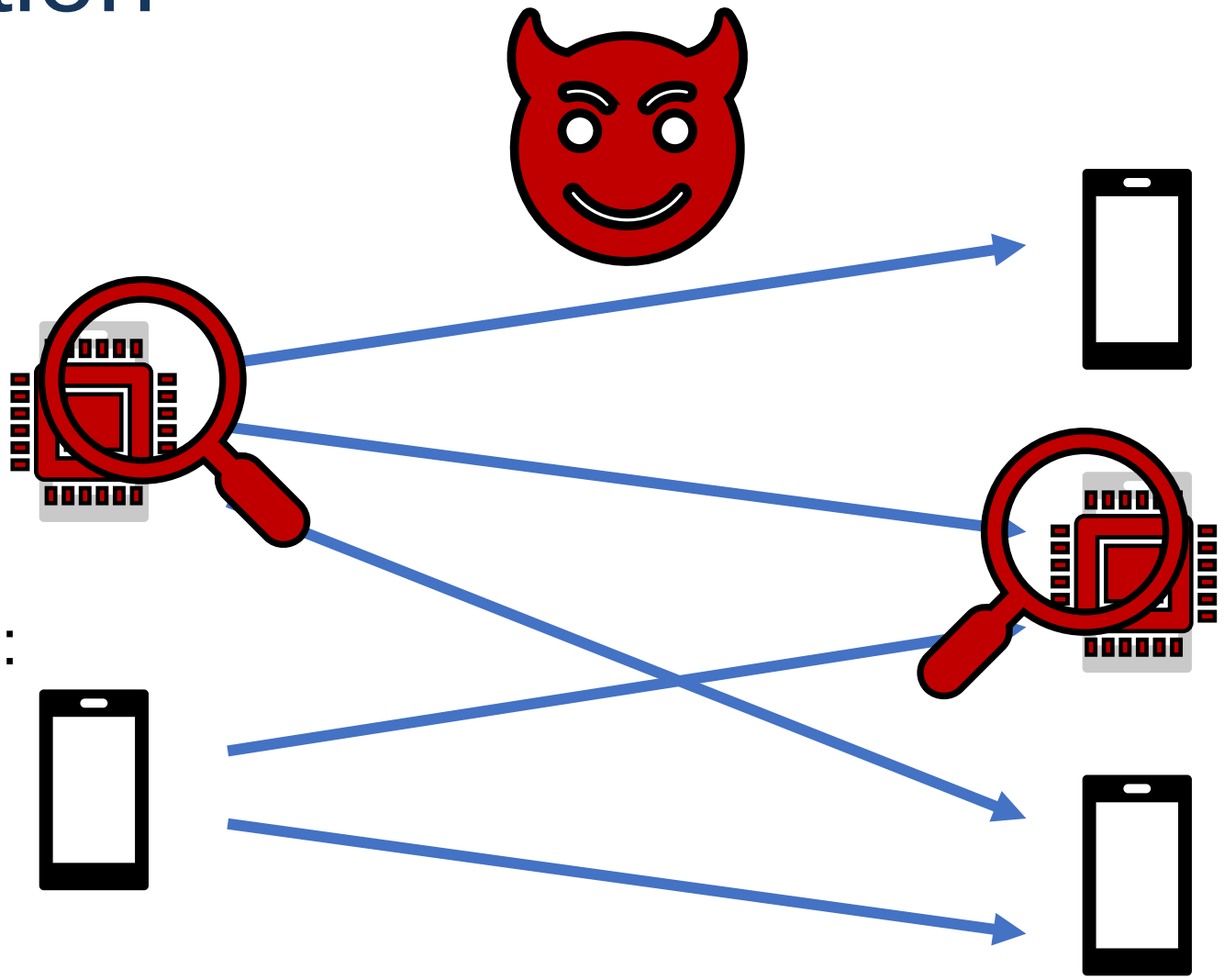
- Focus: Client-to-Client
  - Central Server
  - Federated Servers
  - Peer-to-Peer
  - …?!



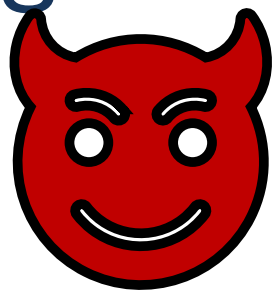
State Corruption

- Forward-Secrecy (FS):
  Past communication remains secure

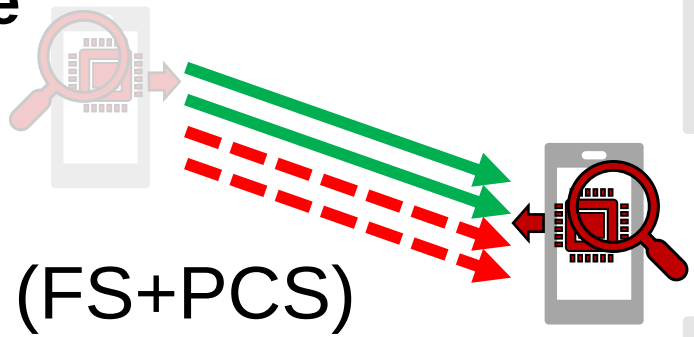
- Post-Compromise Security (PCS):
  Future communication recovers



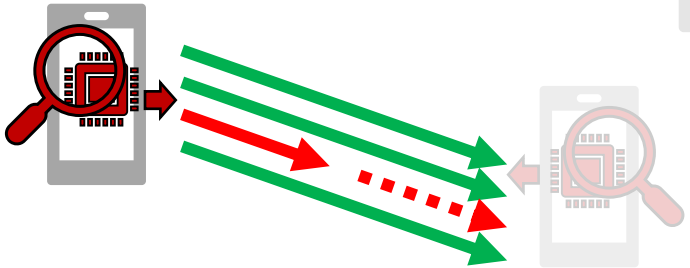
Old Security Guarantees

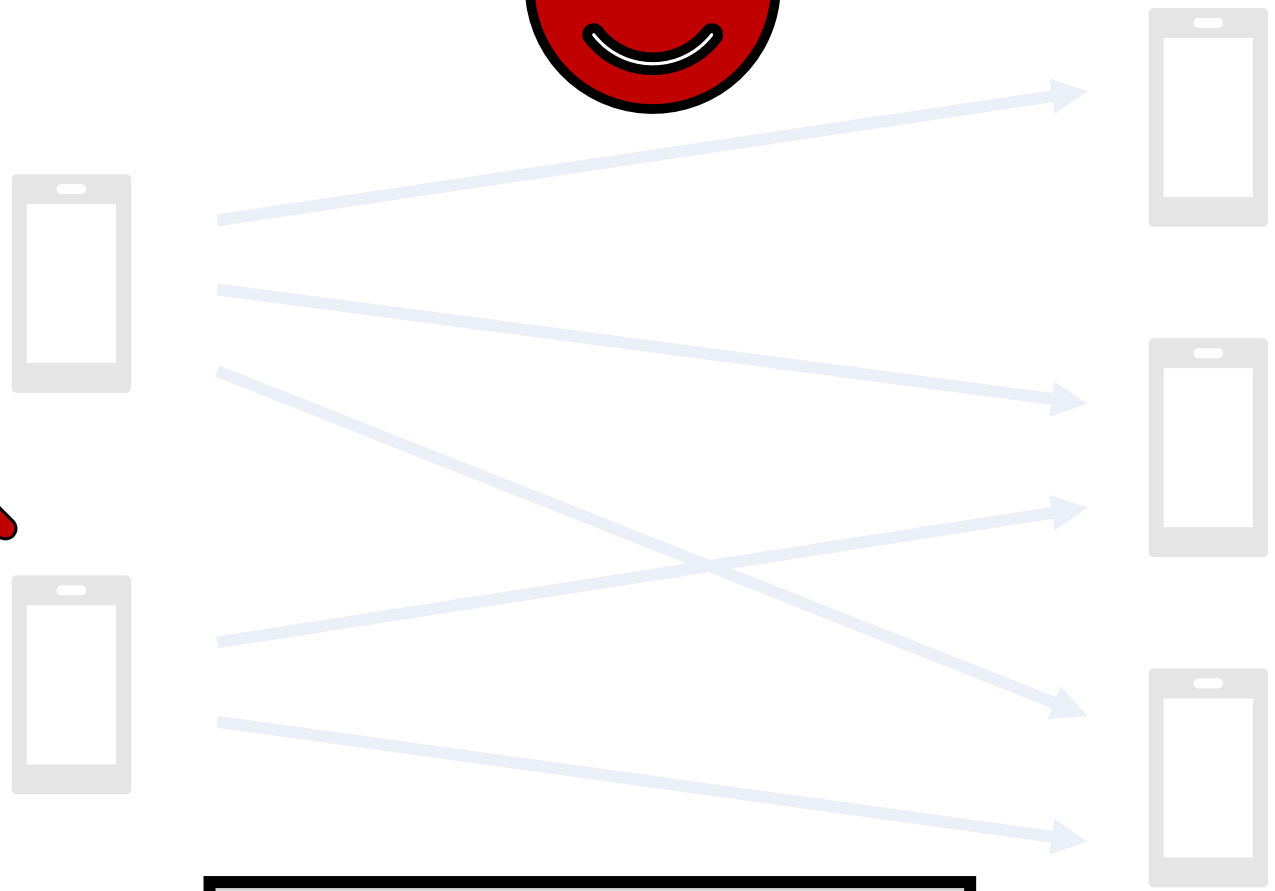


- Confidentiality (FS+PCS)
  - **Sender** corruption is **harmless**
  - **Receiver** corruption **breaks** only **future**



- Authenticity (FS+PCS)
  - Impersonation **only** immediately **after** state **corruption**

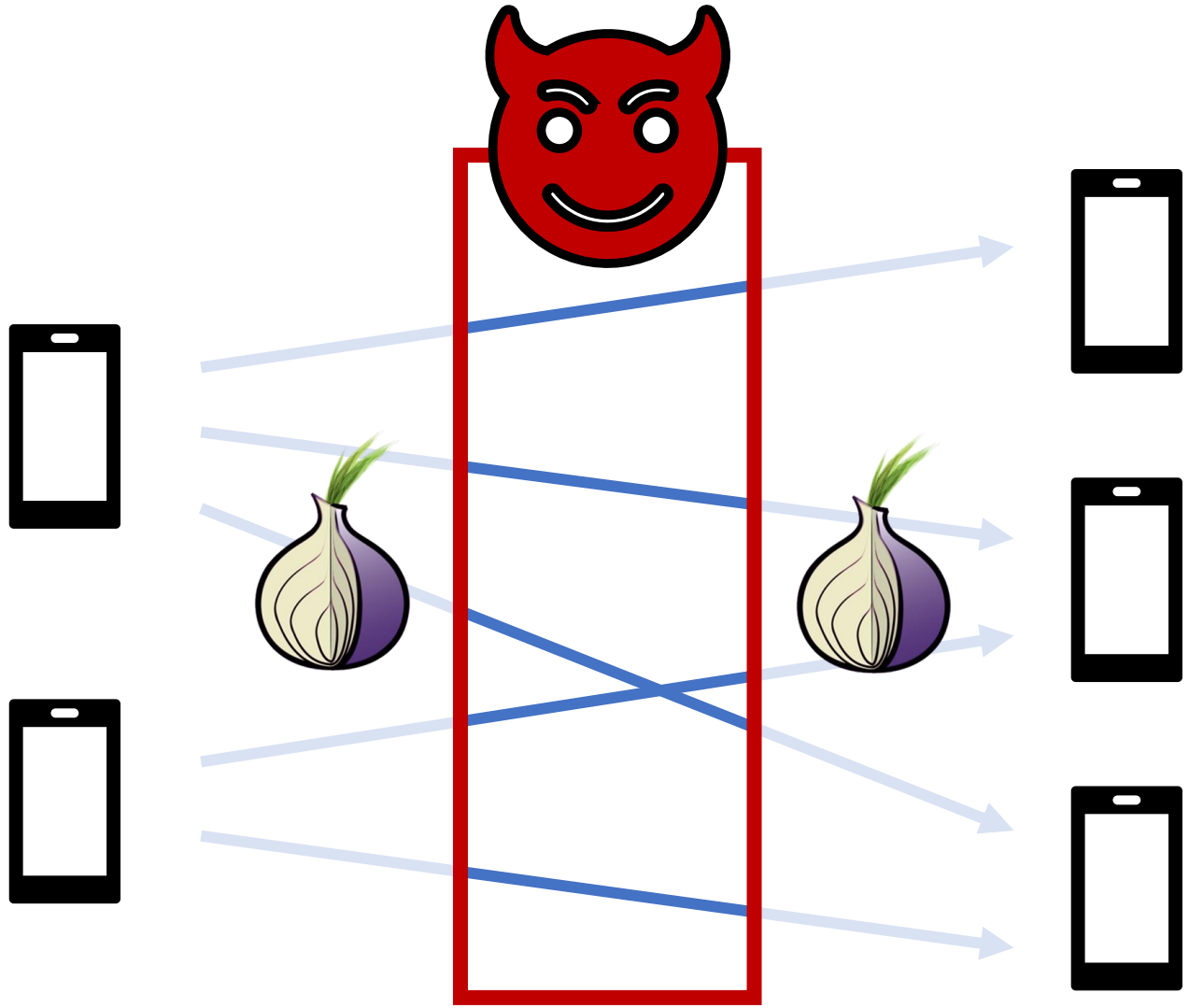




Efficient Construction:
PKE and SIG

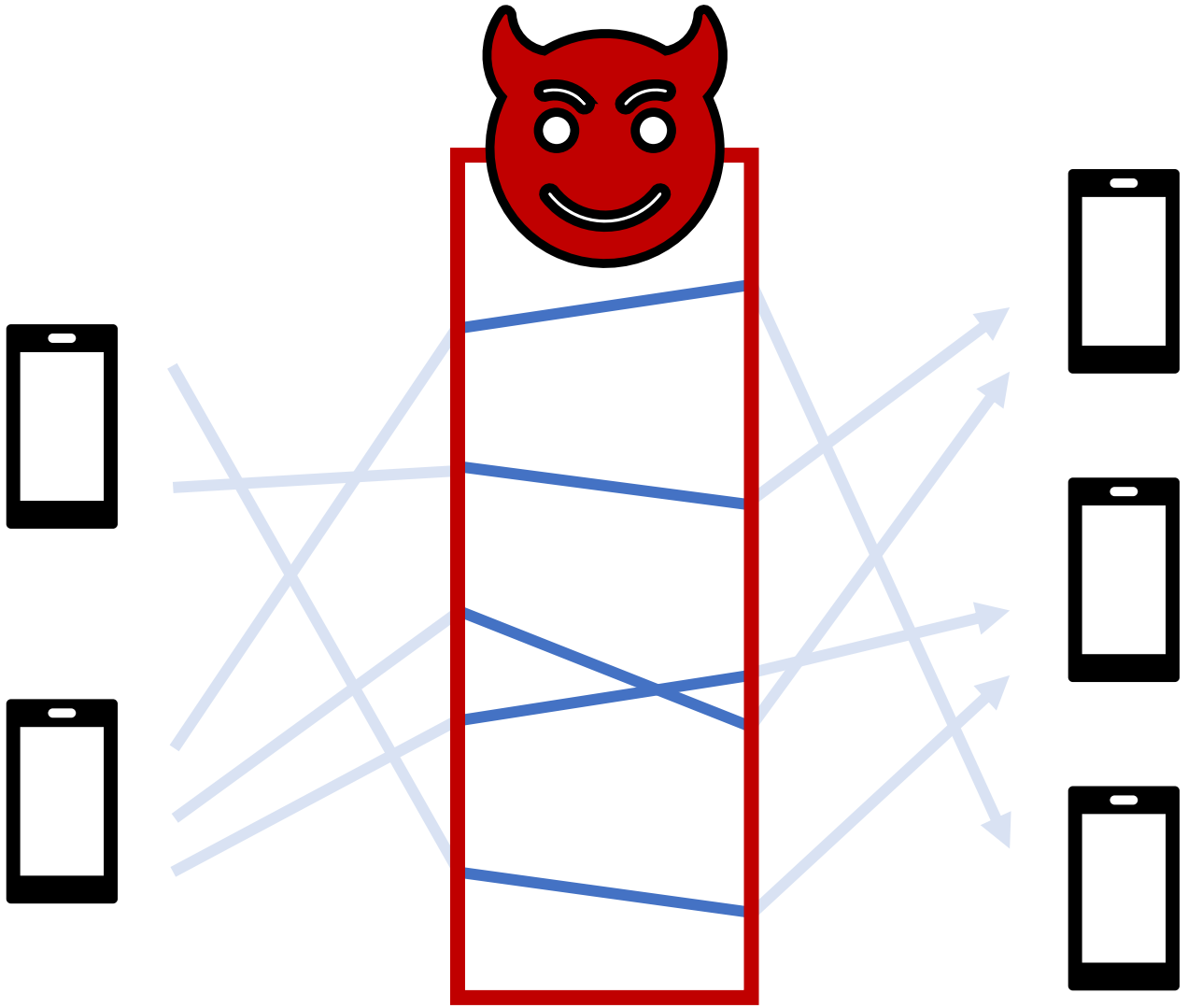
Anonymity

- Adversary …
  - Sees traffic



Anonymity

- Adversary …
  - Sees traffic
  - Looks random

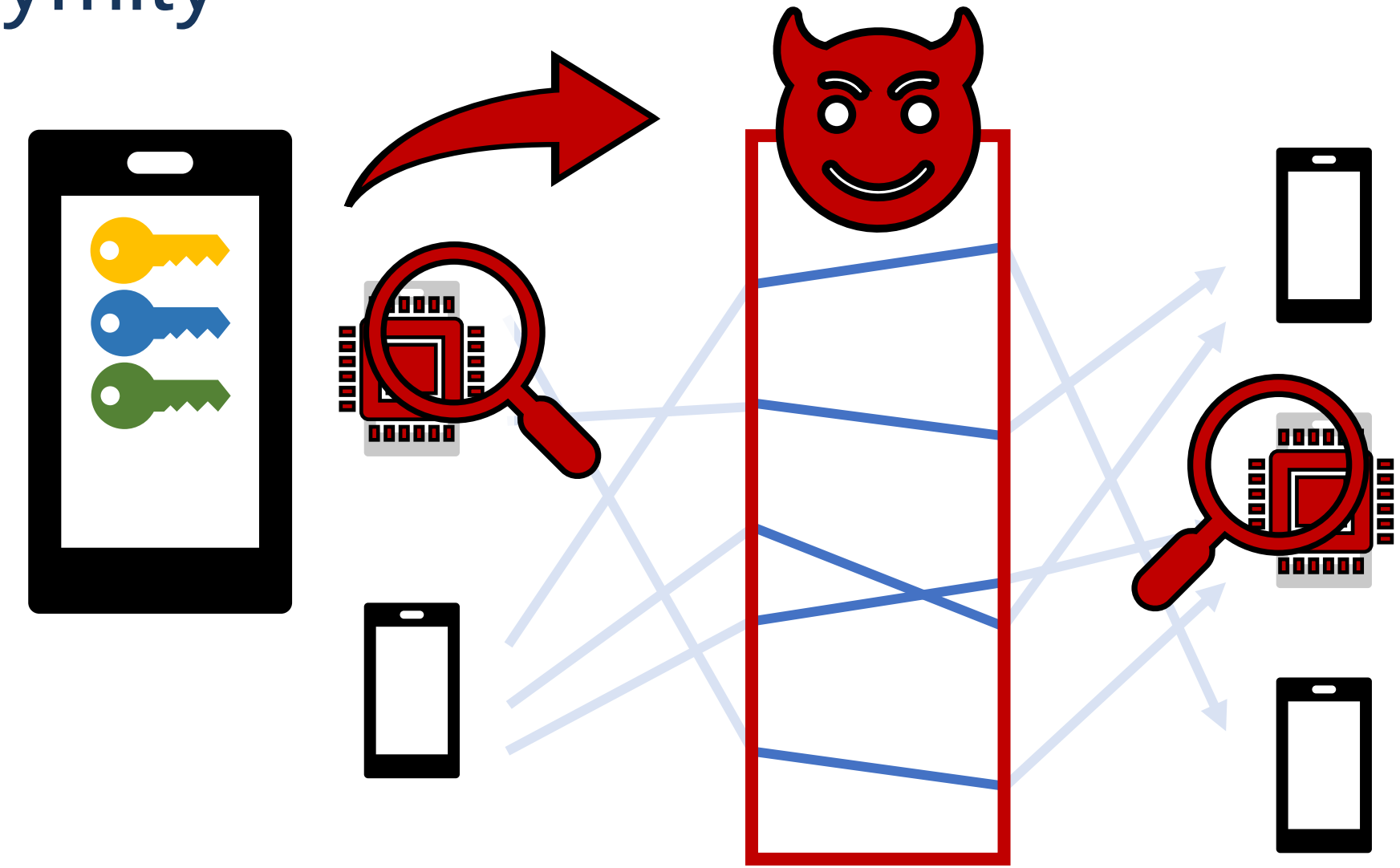


Anonymity

- Adversary …
  - Sees traffic
  - Looks random

!

  - Can corrupt states



Anonymity

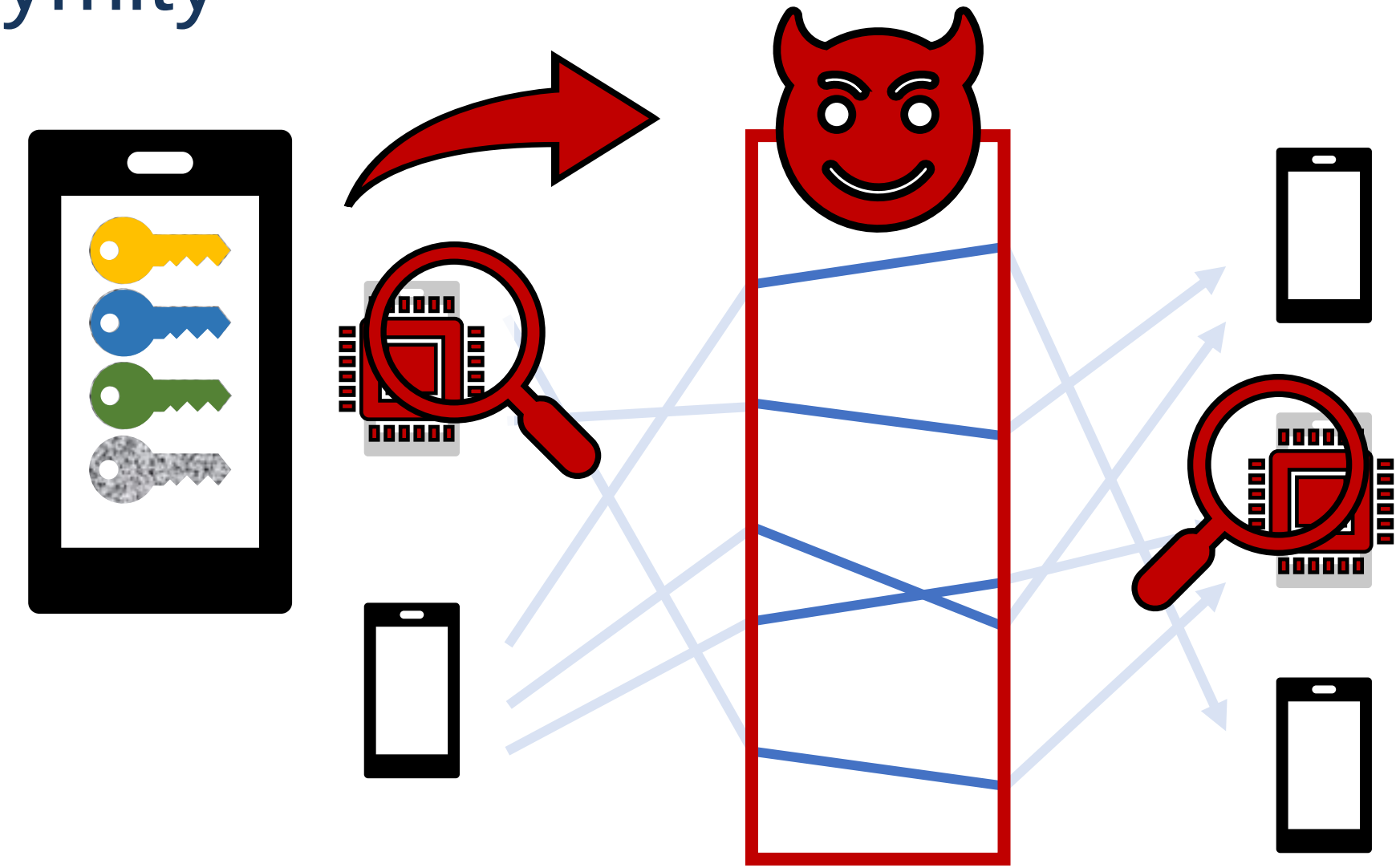
- Adversary …
  - Sees traffic
  - Looks random

!

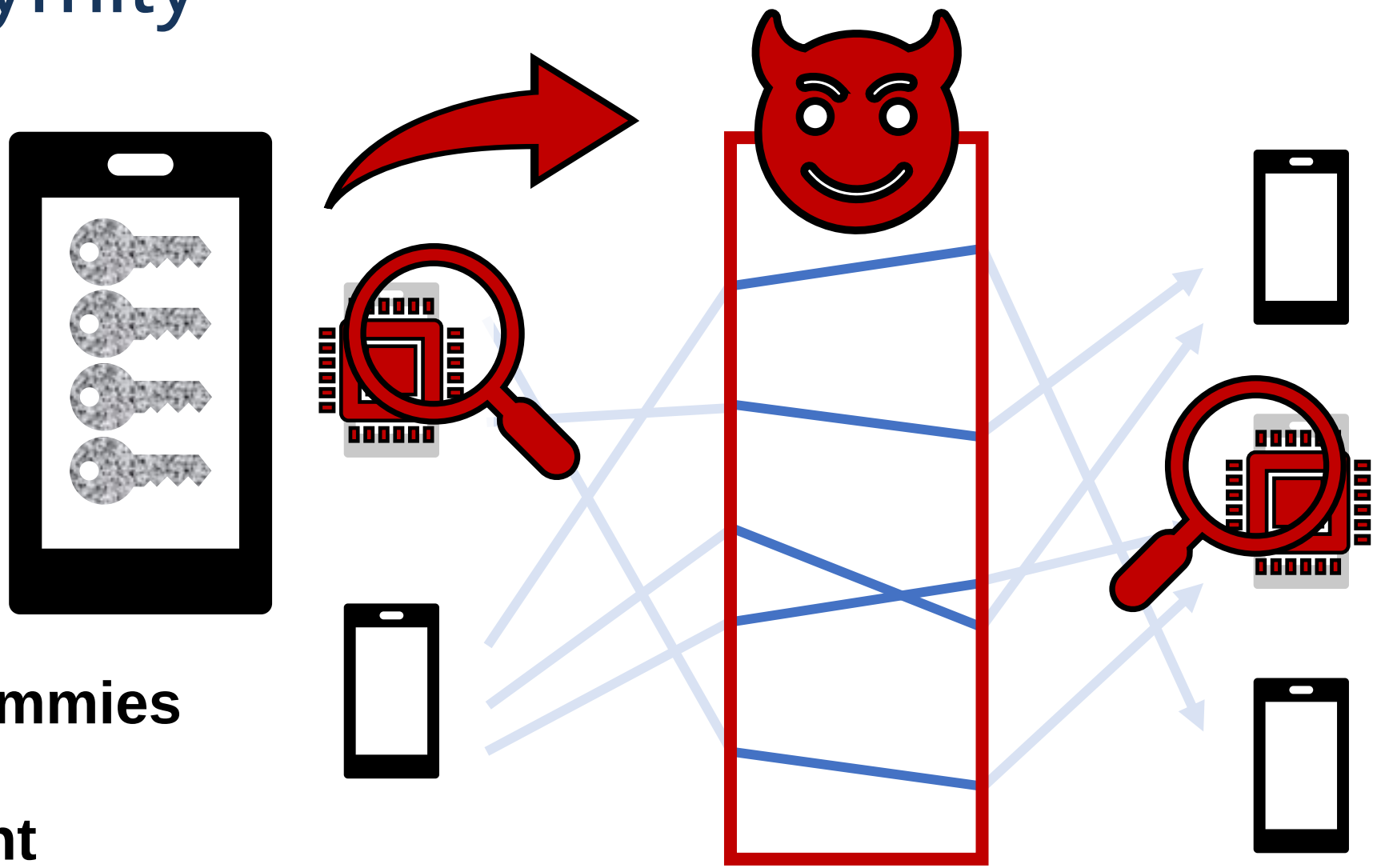
  - Can corrupt states
  - Look random
  - Like dummy states
  - Updatable

!

  - With FS+PCS

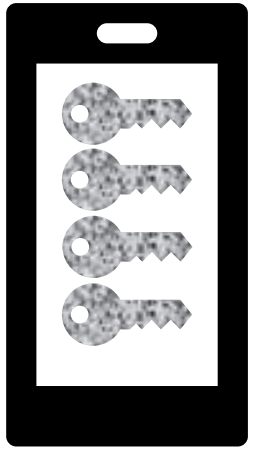


Anonymity



- → **Traffic** random
- → **Secrets** random
- → Randomizable **dummies**
- → Secrets and traffic **independent**

Construction Idea: PKE



- Updatable encryption key
  → Sender state re-randomized

- Keys:

$$ek \leftarrow (g^r, g^{xr})$$
$$dk \leftarrow x$$

- Key update:

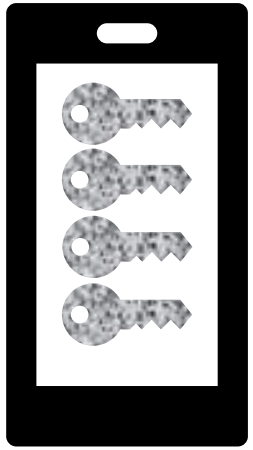
$$ek = (ek_0, ek_1)$$
$$ek' \leftarrow (ek_0^{r'}, ek_1^{r'}) = (g^R, g^{xR})$$

- En- & Decryption:

$$c \leftarrow (ek_0^s, \mathrm{H}(ek_0^s, ek_1^s) \oplus m) = (c_0, c_1)$$
$$m \leftarrow \mathrm{H}(c_0, c_0^{dk}) \oplus c_1$$
$$= \mathrm{H}(g^R, (g^R)^x) \oplus (\mathrm{H}(g^R, g^{xR}) \oplus m)$$

Construction Ideas: SIG

- Updatable signing key
  → Sender state re-randomized



$$sk^*_{i,b} \leftarrow_\$ \mathcal{R}, \quad i \in [l], b \in \{0,1\}$$

- Keys (Lamport):

$$sk^* \leftarrow \begin{pmatrix} sk^*_{0,0}, & \cdots & , sk^*_{l-1,0} \\ sk^*_{0,1}, & \cdots & , sk^*_{l-1,1} \end{pmatrix}$$

$$vk^* \leftarrow \begin{pmatrix} \mathrm{f}(sk^*_{0,0}), & \cdots & , \mathrm{f}(sk^*_{l-1,0}) \\ \mathrm{f}(sk^*_{0,1}), & \cdots & , \mathrm{f}(sk^*_{l-1,1}) \end{pmatrix}$$

- Keys (Encrypted Lamport):

$$c_{i,b} \leftarrow \mathrm{enc}(ek, sk^*_{i,b})$$

$$sk \leftarrow \begin{pmatrix} c_{0,0}, & \cdots & , c_{l-1,0} \\ c_{0,1}, & \cdots & , c_{l-1,1} \end{pmatrix}$$

$$vk \leftarrow (vk^*, dk)$$

Construction Ideas: SIG

- Keys:

$$c_{i,b} = \mathrm{enc}(ek, sk^*_{i,b}), \quad i \in [l], b \in \{0,1\}$$

$$sk = \begin{pmatrix} c_{0,0}, & \cdots & , c_{l-1,0} \\ c_{0,1}, & \cdots & , c_{l-1,1} \end{pmatrix}$$

$$vk = \left(\begin{pmatrix} vk^*_{0,0} & = \mathrm{f}(sk^*_{0,0}), & \cdots & , vk^*_{l-1,0} \\ vk^*_{0,1} & , & \cdots & , vk^*_{l-1,1} \end{pmatrix}, dk \right)$$

- Signature:

$$m = (m_0, \cdots, m_{l-1}), \quad m_i \in \{0,1\}$$

$$\sigma \leftarrow (c_{0,m_i}, \cdots, c_{l-1,m_i})$$

$$= (\mathrm{enc}(ek, sk^*_{0,m_i}), \cdots, \mathrm{enc}(ek, sk^*_{l-1,m_i}))$$

- Verify:

$$\text{For all } i \in [l]\text{:}$$

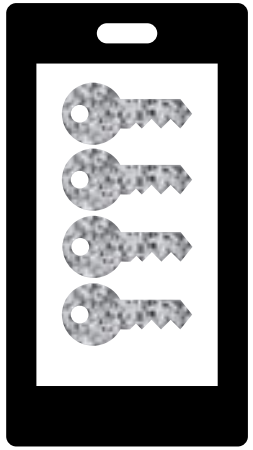
$$\mathrm{f}(\mathrm{dec}(dk, \sigma_i)) \stackrel{?}{=} vk^*_{i,m_i} = \mathrm{f}(sk^*_{i,m_i})$$

Construction Ideas: SIG

- Updatable signing key
  → Sender state re-randomized

$$c_{i,b} = \mathrm{enc}(ek, sk^*_{i,b}), \quad i \in [l], b \in \{0,1\}$$

$$sk = \begin{pmatrix} c_{0,0}, & \cdots & , c_{l-1,0} \\ c_{0,1}, & \cdots & , c_{l-1,1} \end{pmatrix}$$



- Key update:

$$(ek', c'_{0,0}, \cdots, c'_{l-1,0}, c'_{0,1}, \cdots, c'_{l-1,1}) \leftarrow \mathrm{rand}(ek, c_{0,0}, \cdots, c_{l-1,0}, c_{0,1}, \cdots, c_{l-1,1}))$$

$$sk' = \begin{pmatrix} c'_{0,0}, & \cdots & , c'_{l-1,0} \\ c'_{0,1}, & \cdots & , c'_{l-1,1} \end{pmatrix}$$

Construction Ideas: SIG

- Compact signature: $\sigma \leftarrow \prod_{i \in [l]} c_{i,m_i}$

$$= g_1^{\sum sk^*_{i,m_i}}$$

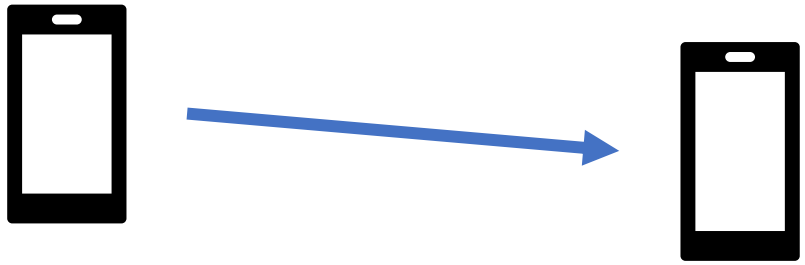
BLS Aggregation Trick

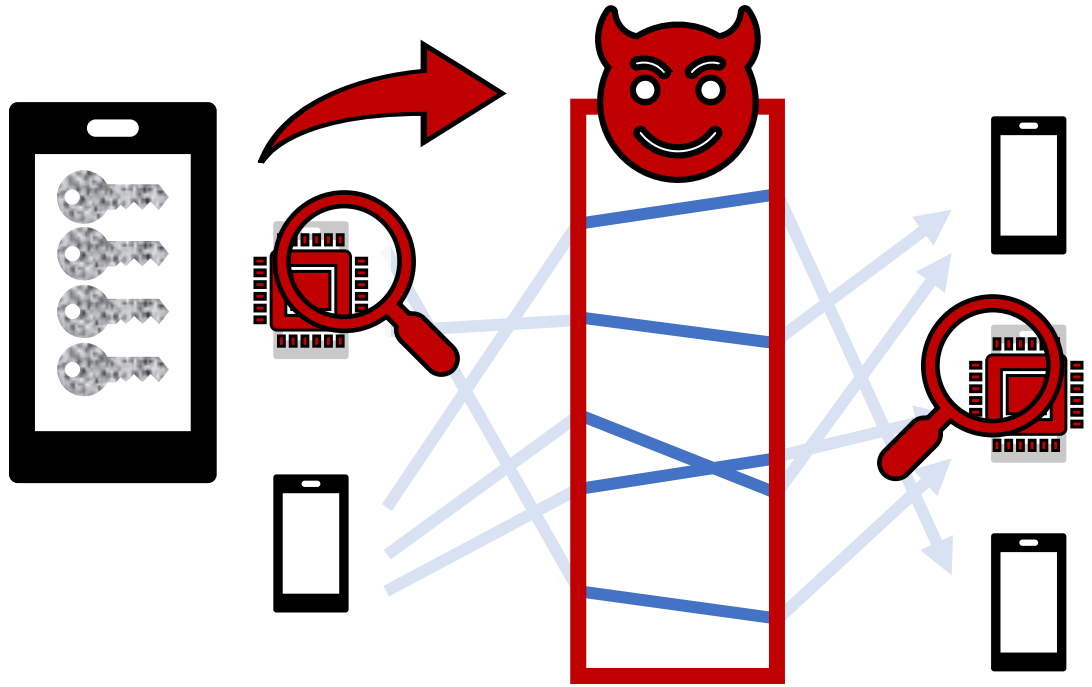
- Verification: $\mathrm{f}(\mathrm{dec}(dk, \sigma)) \stackrel{?}{=} \prod_{i \in [l]} vk_{i,m_i}$

$$e(\sigma, g_2) \stackrel{?}{=} e(g_1^{\sum sk^*_{i,m_i}}, g_2)$$

Performance

→ Dominated by SIG





- Sending
  - $4l$ group operations (signing)
  - $2l$ group elements (verification key)
- Receiving
  - $2l$ group operations (verifying)

ia.cr/2022/1187