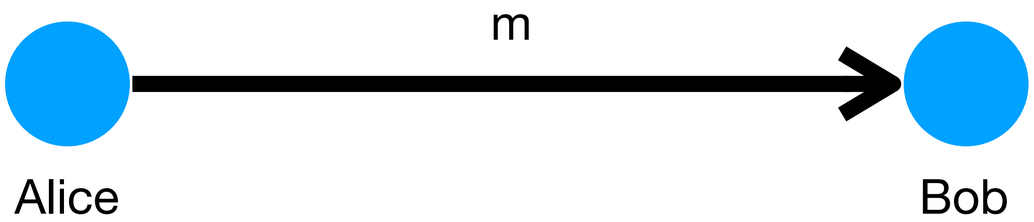
Poly Onions: Achieving Anonymity in the Presence of Churn

Megumi Ando (MITRE)
Miranda Christ (Columbia University)
Anna Lysyanskaya (Brown University)
Tal Malkin (Columbia University)

Our Problem

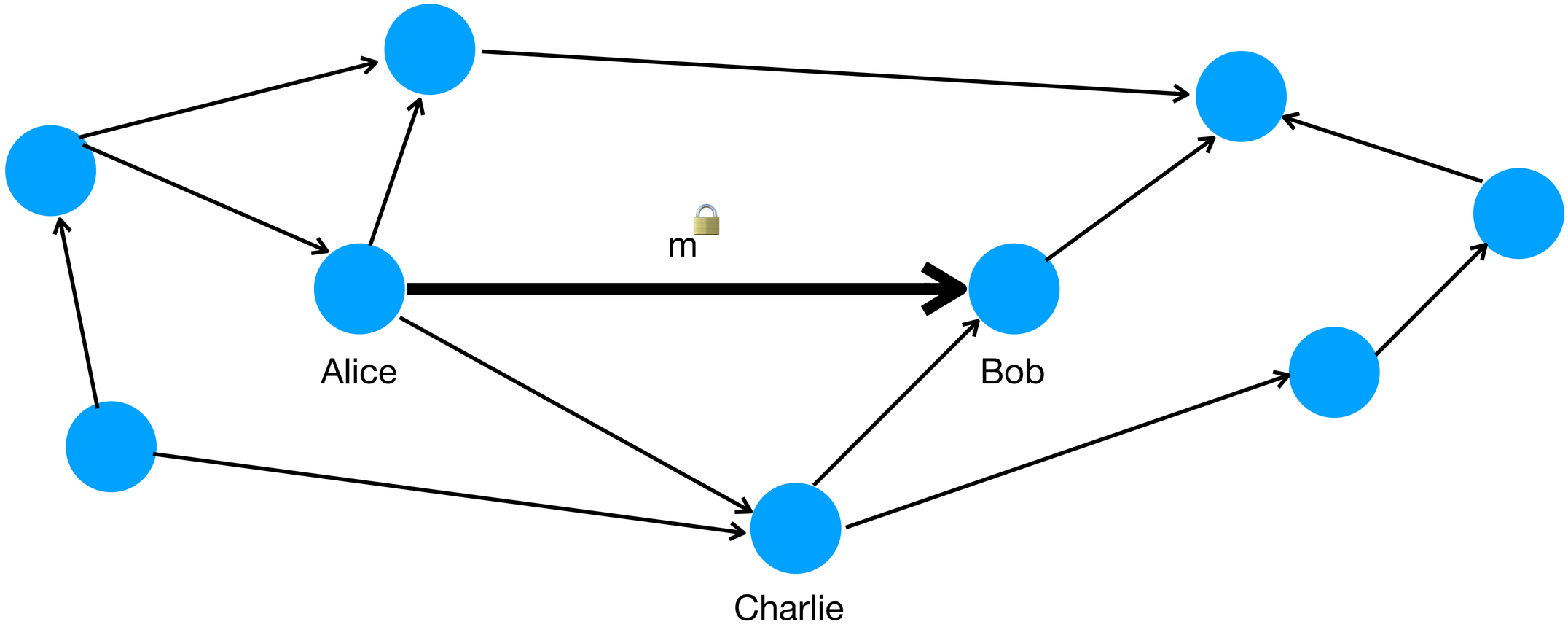
Alice wants to **anonymously** send a message to Bob



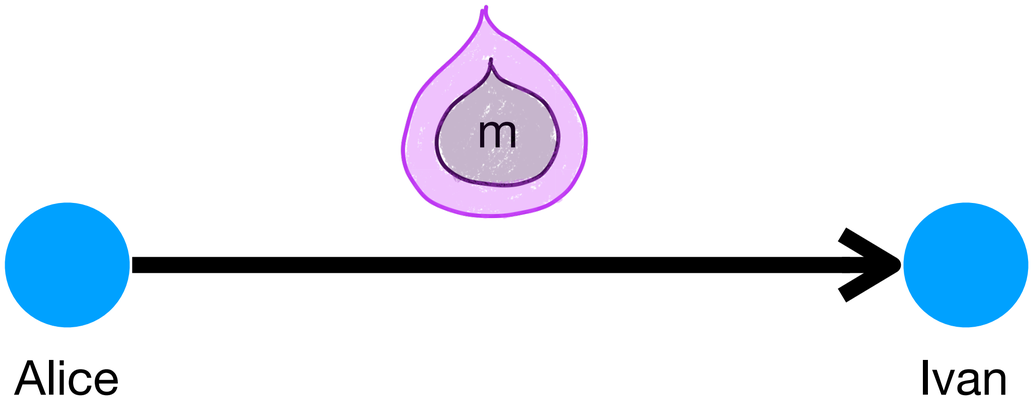
What do we mean by anonymous?

Adversary can’t tell who Alice is talking to

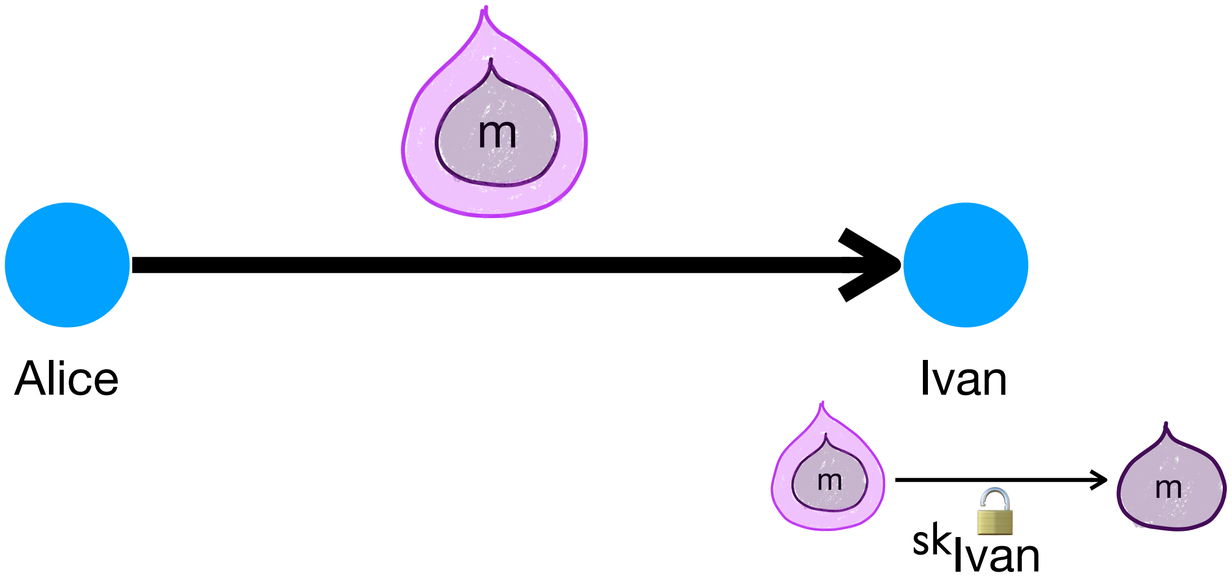
Even if it can **observe all network traffic** & **corrupt** constant fraction of nodes



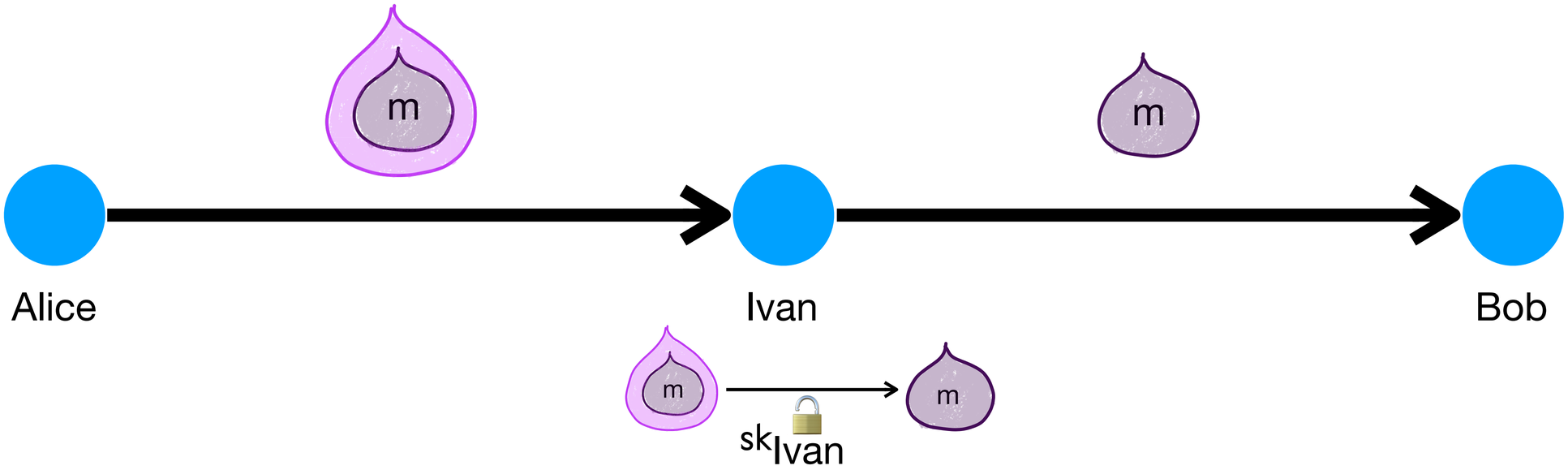
Solution: Onion Routing



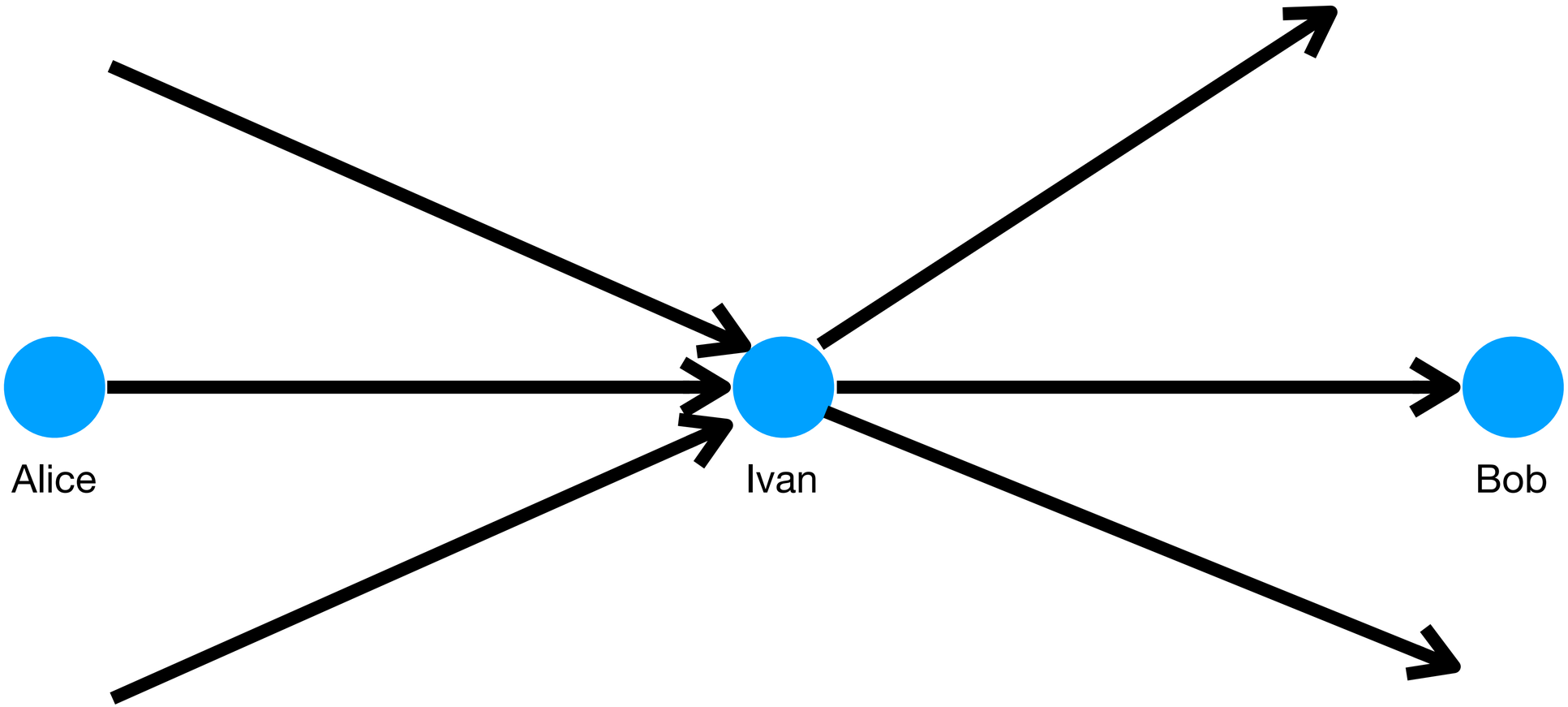
Solution: Onion Routing



Solution: Onion Routing



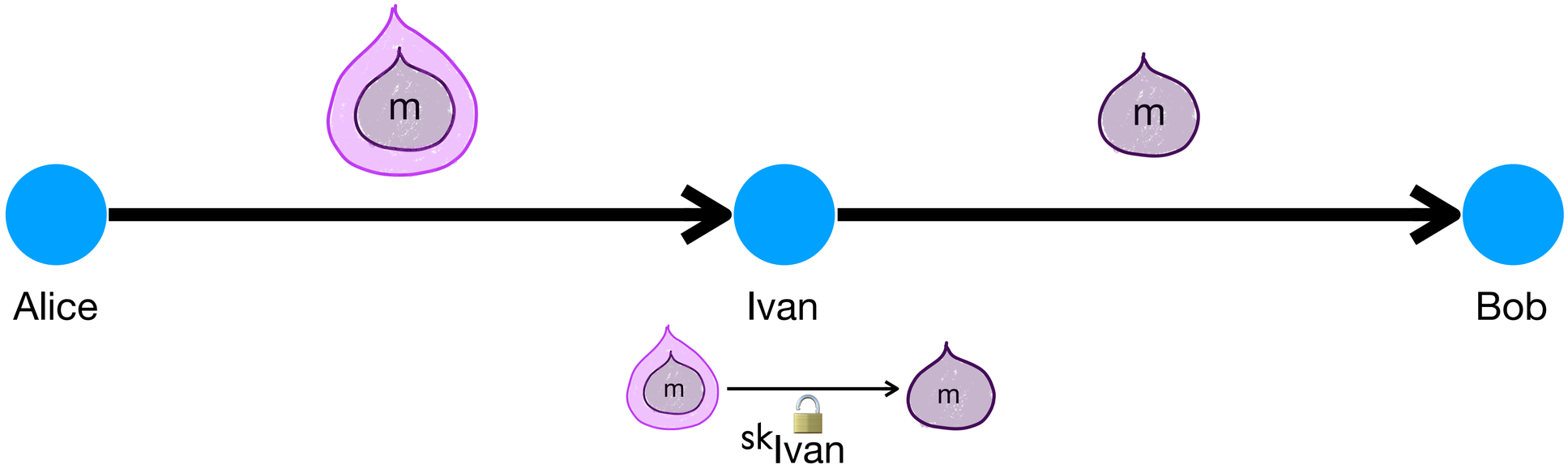
Solution: Onion Routing



Onions **mix** at honest intermediaries

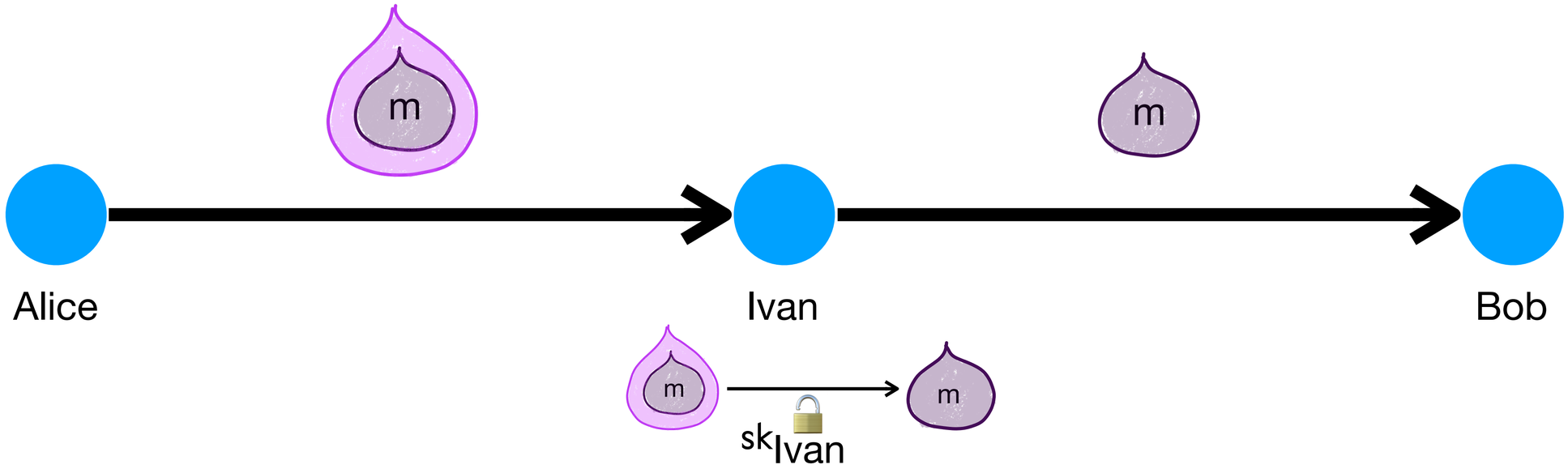
Solution: Onion Routing

There exists an onion routing protocol that is anonymous with $\text{polylog}(\lambda)$ rounds [Ando, Lysyanskaya, Upfal ‘18]

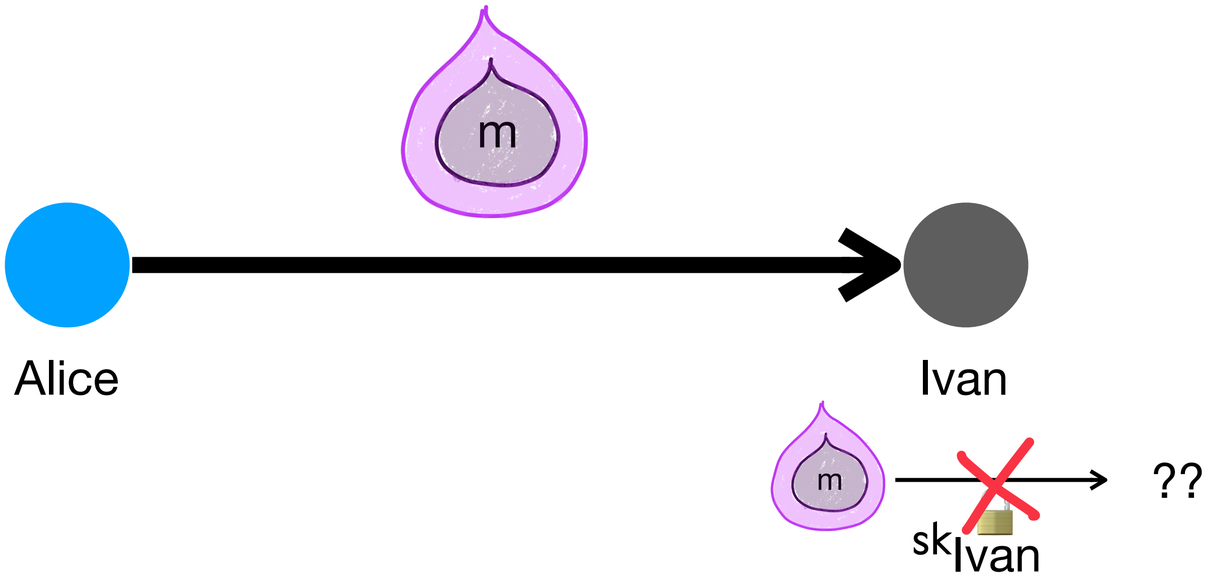


Solution: Onion Routing

There exists an onion routing protocol that is anonymous with $\text{polylog}(\lambda)$ rounds [Ando, Lysyanskaya, Upfal ‘18] **in the single-run setting without churn**



What if Ivan is offline?



Our contributions

- Introduce anonymity definitions for the multi-run setting with network churn

Our contributions

- Introduce anonymity definitions for the multi-run setting with network churn
- Show that for a natural class of onion routing protocols, single-run anonymity implies multi-run anonymity

Our contributions

- Introduce anonymity definitions for the multi-run setting with network churn
- Show that for a natural class of onion routing protocols, single-run anonymity implies multi-run anonymity
- Define poly onion encryption (I/O, security defs)

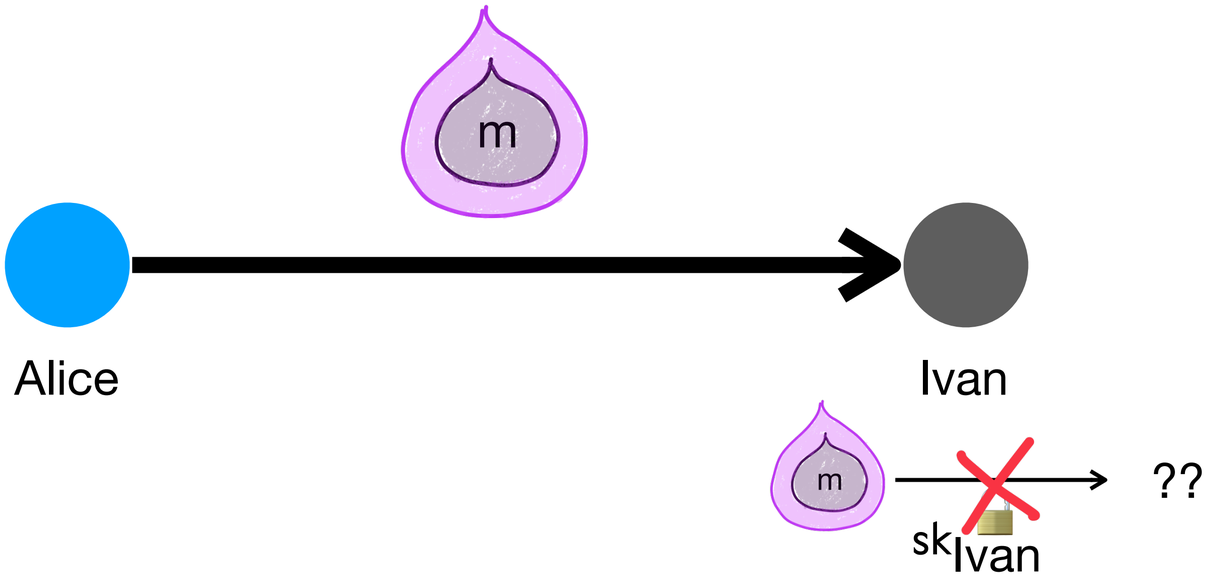
Our contributions

- Introduce anonymity definitions for the multi-run setting with network churn
- Show that for a natural class of onion routing protocols, single-run anonymity implies multi-run anonymity
- Define poly onion encryption (I/O, security defs)
- Construct a poly onion encryption scheme

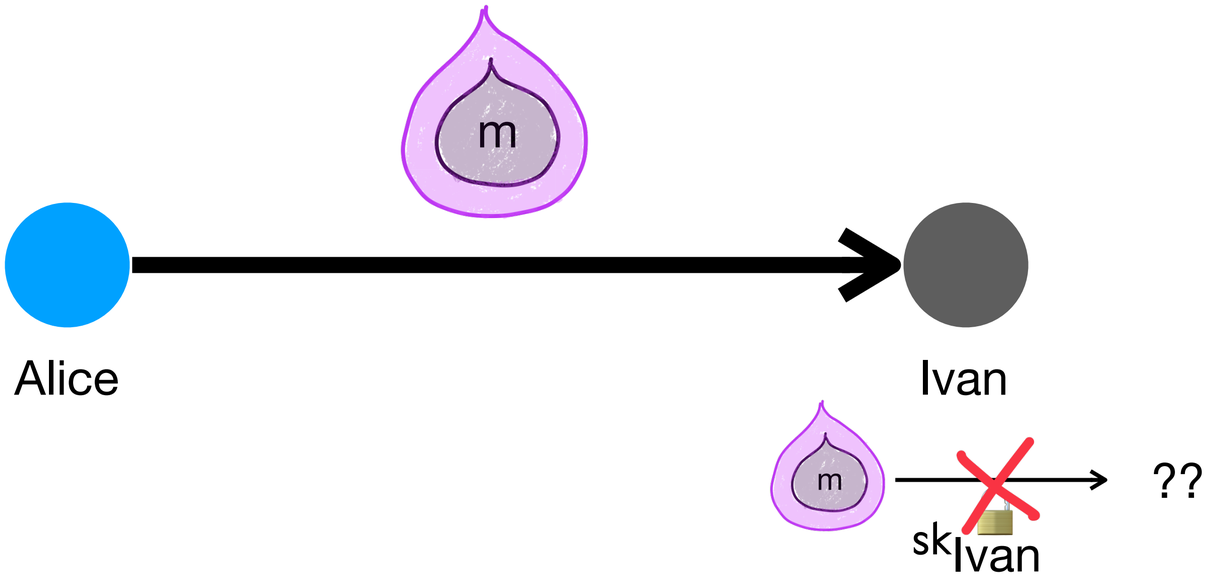
Our contributions

- Introduce anonymity definitions for the multi-run setting with network churn
- Show that for a natural class of onion routing protocols, single-run anonymity implies multi-run anonymity
- Define poly onion encryption (I/O, security defs)
- Construct a poly onion encryption scheme
- Apply Poly Onion Encryption to a known onion routing protocol to obtain a protocol that is anonymous against a passive adversary, with churn

What if Ivan is offline?



What if Ivan is offline?

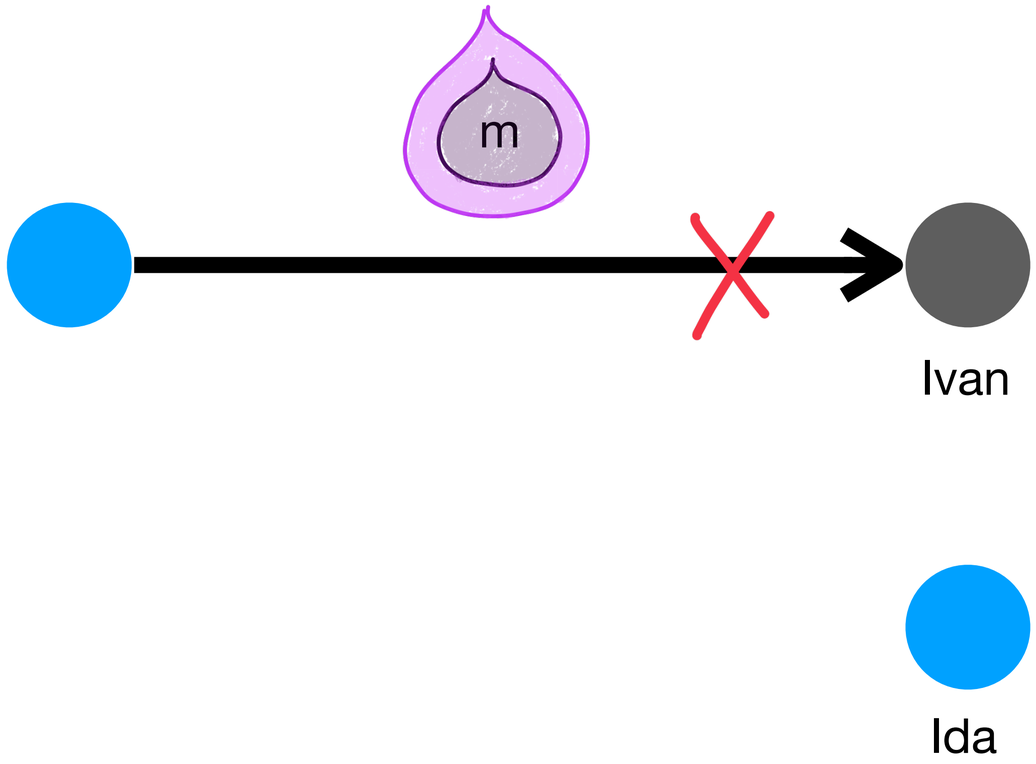


Onion is *dropped*

First attempt: duo onions

[Iwanik, Klonowski, Kutyłowski ‘05]

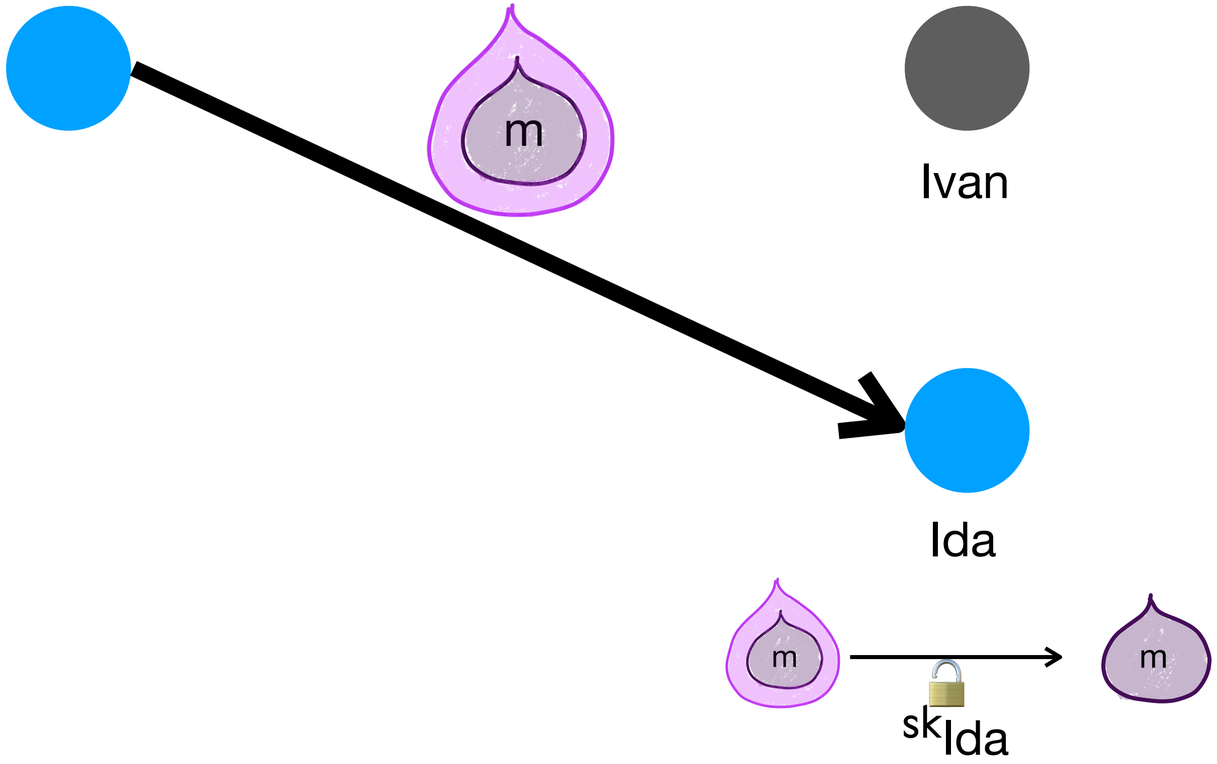
Idea: onion is peelable by Ivan *and* Ida



First attempt: duo onions

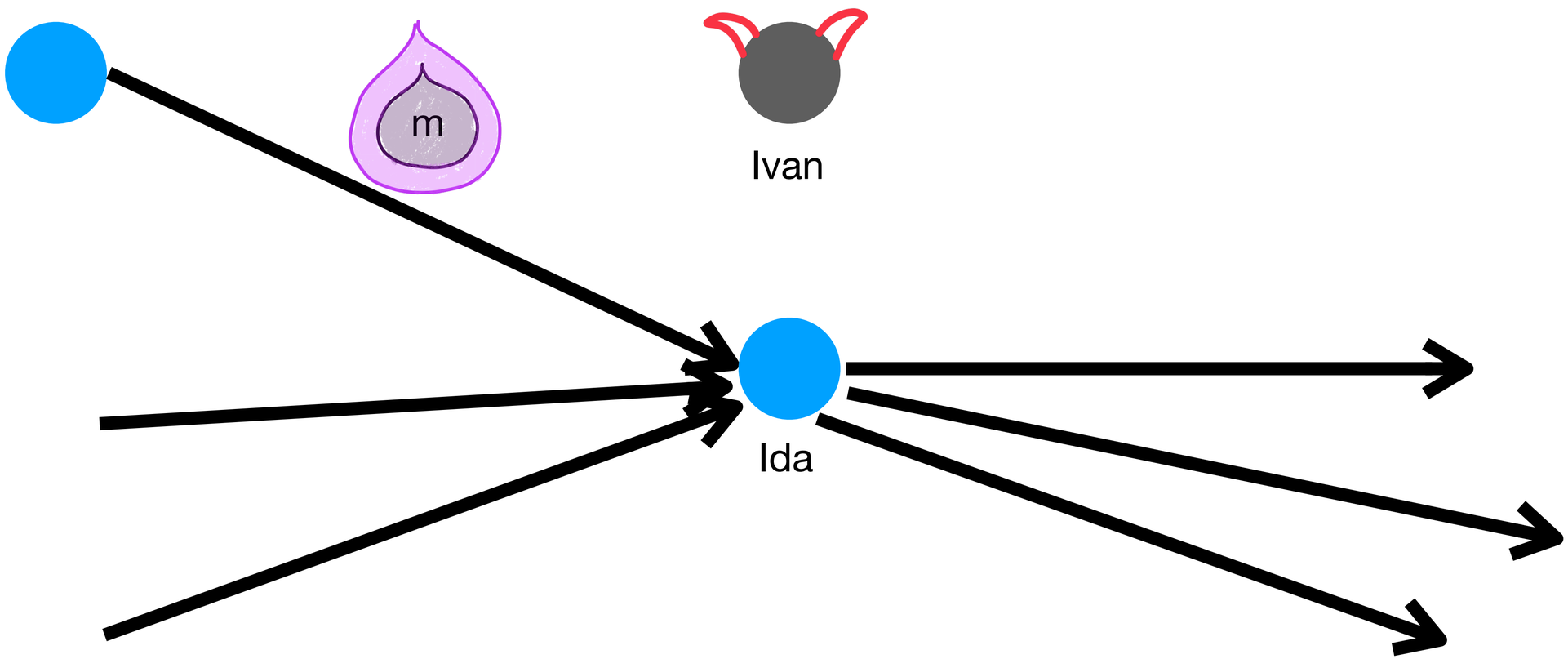
[Iwanik, Klonowski, Kutyłowski ‘05]

Idea: onion is peelable by Ivan *and Ida*



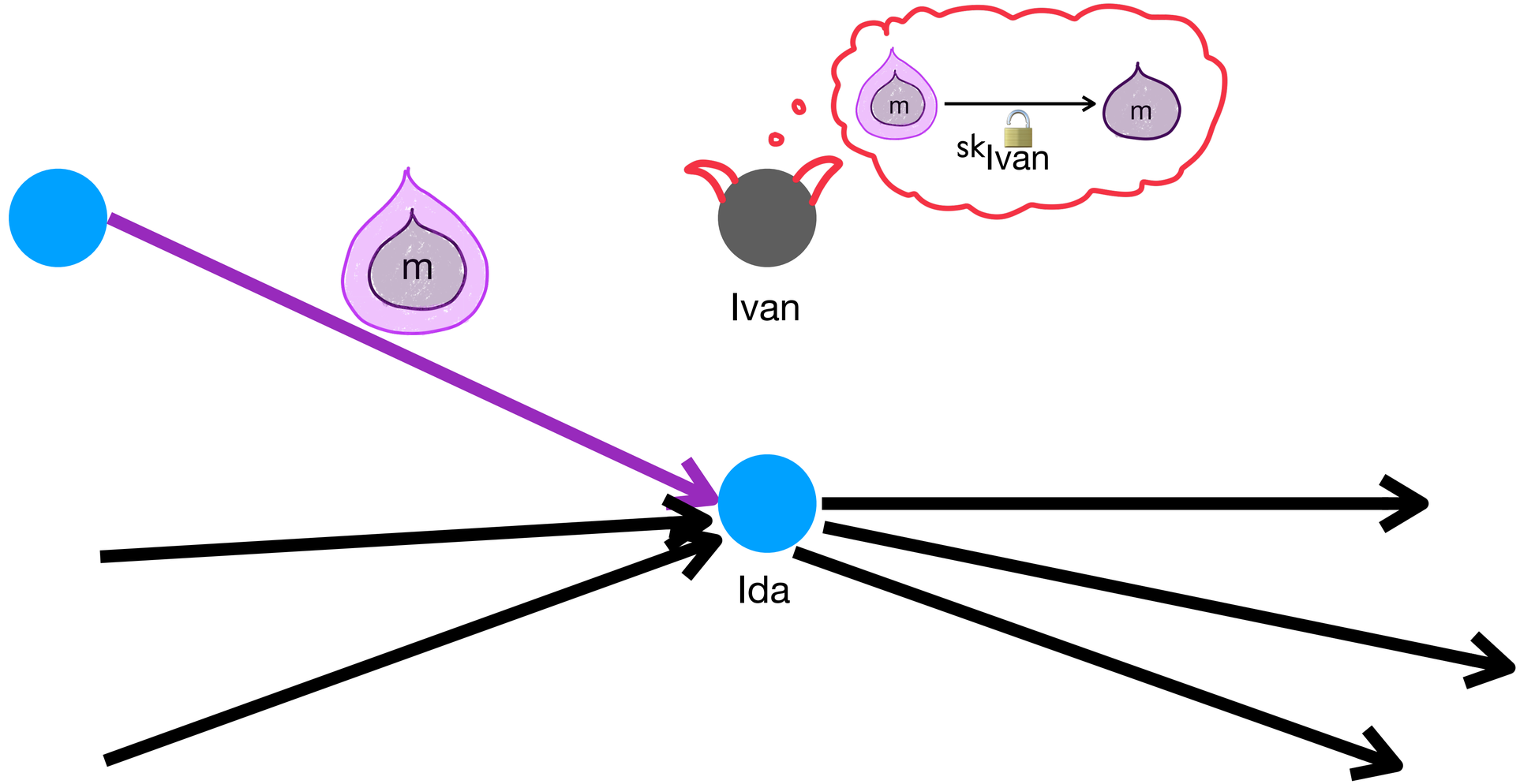
Problem: no mixing

If *either* Ivan or Ian is corrupted, the adversary can trace the onion



Problem: no mixing

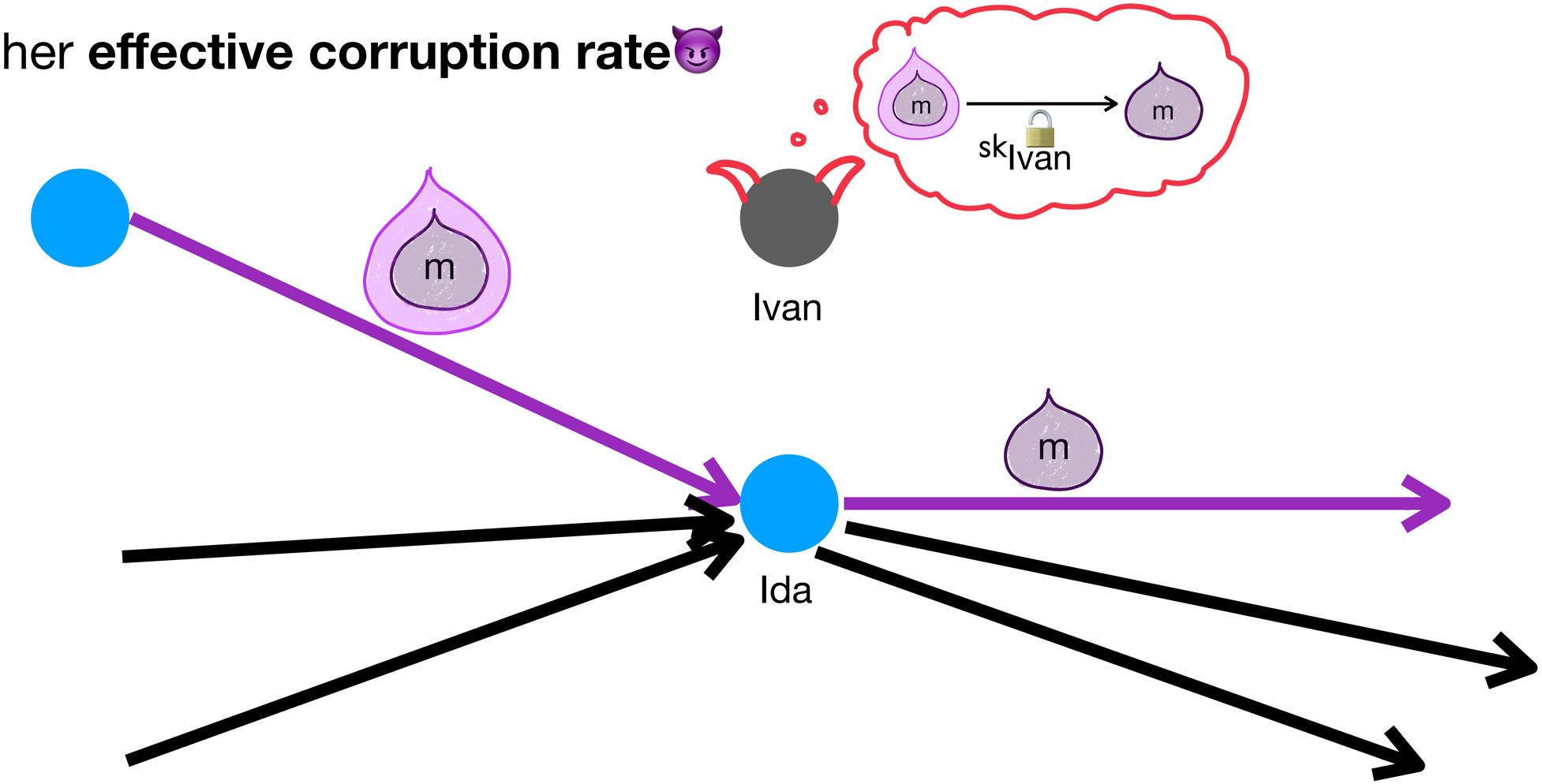
If *either* Ivan or Ida is corrupted, the adversary can trace the onion



Problem: no mixing

If *either* Ivan or Ida is corrupted, the adversary can trace the onion

➡️ higher **effective corruption rate**😈

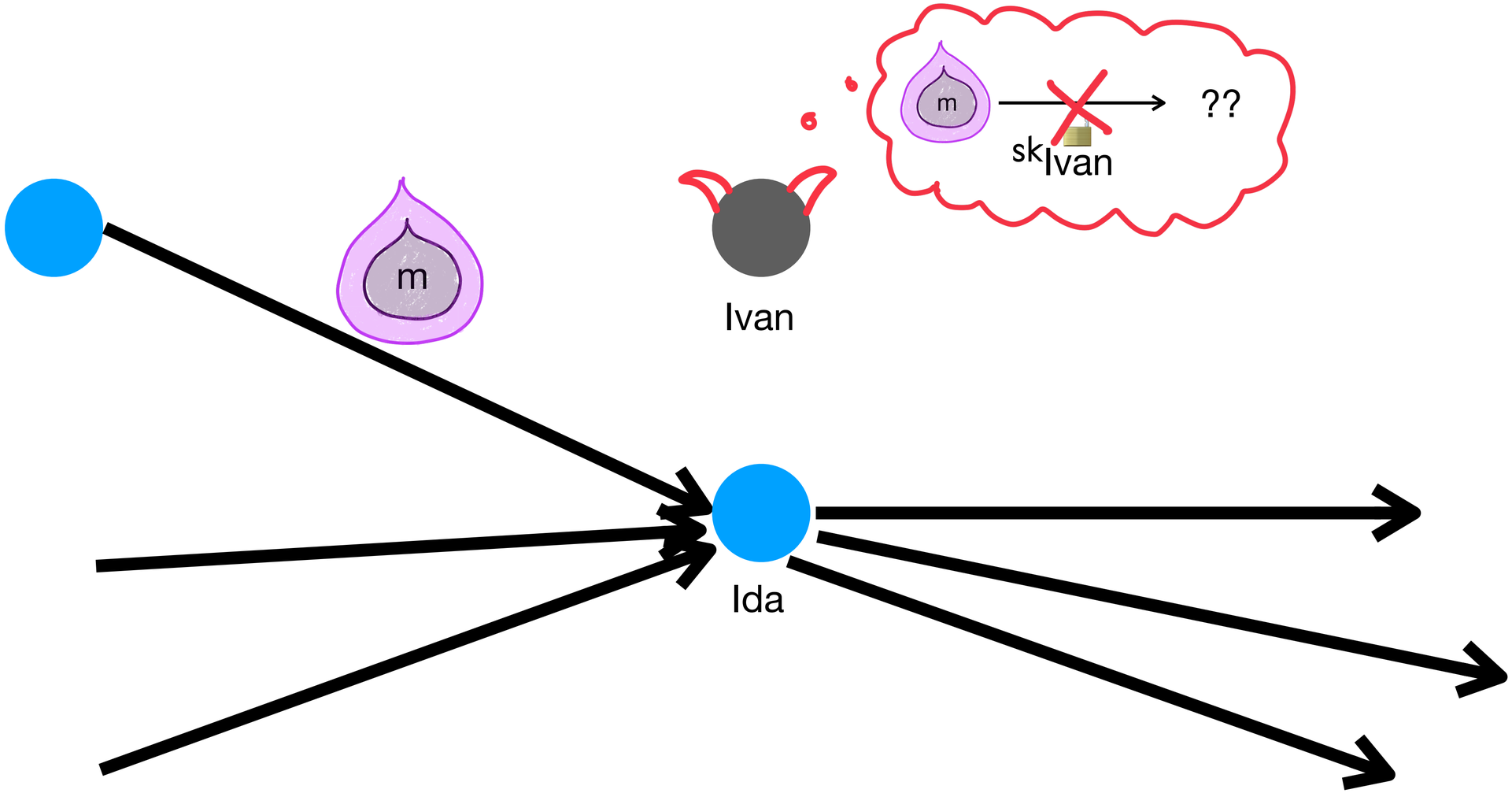


Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion

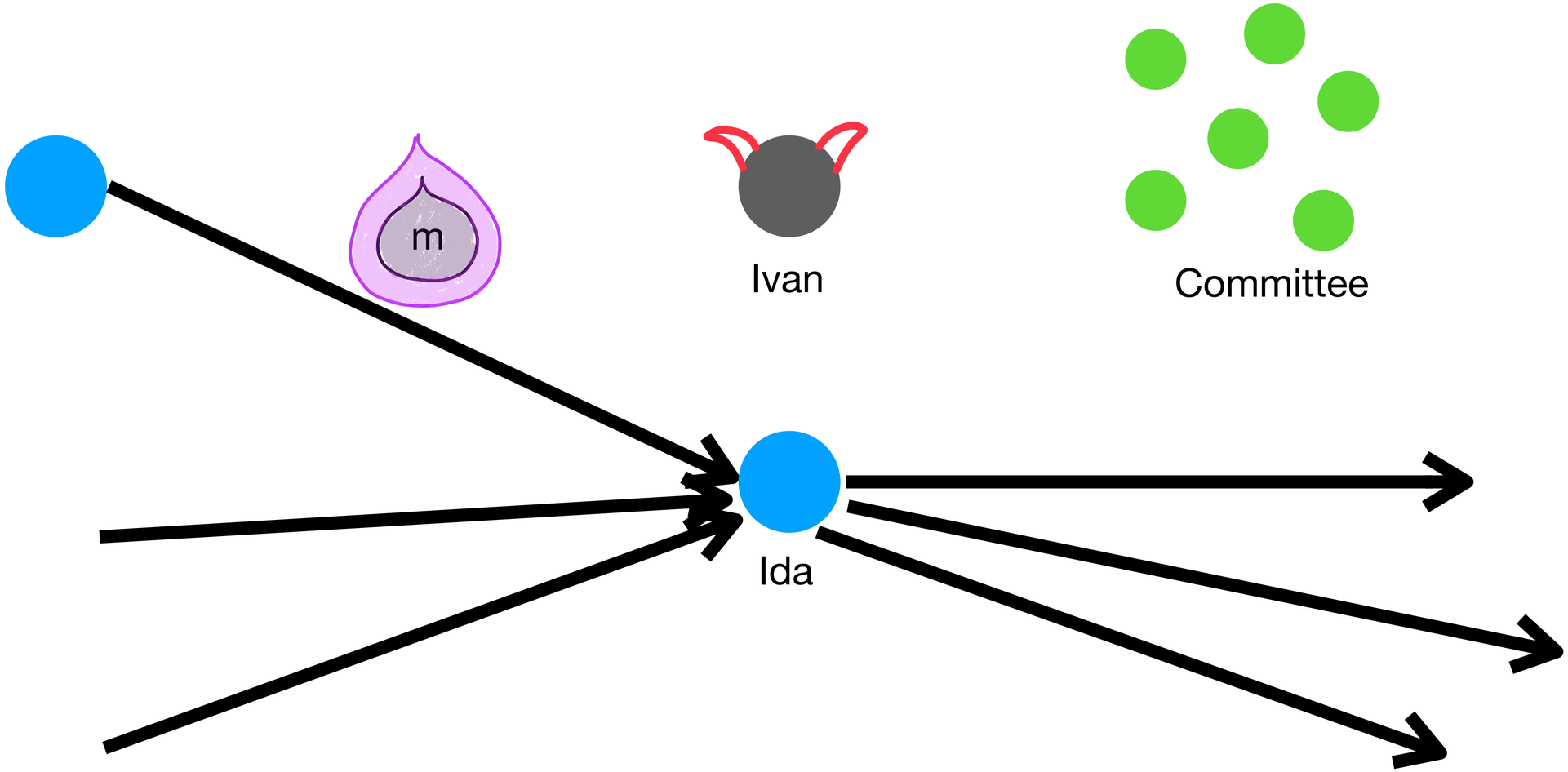
Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion



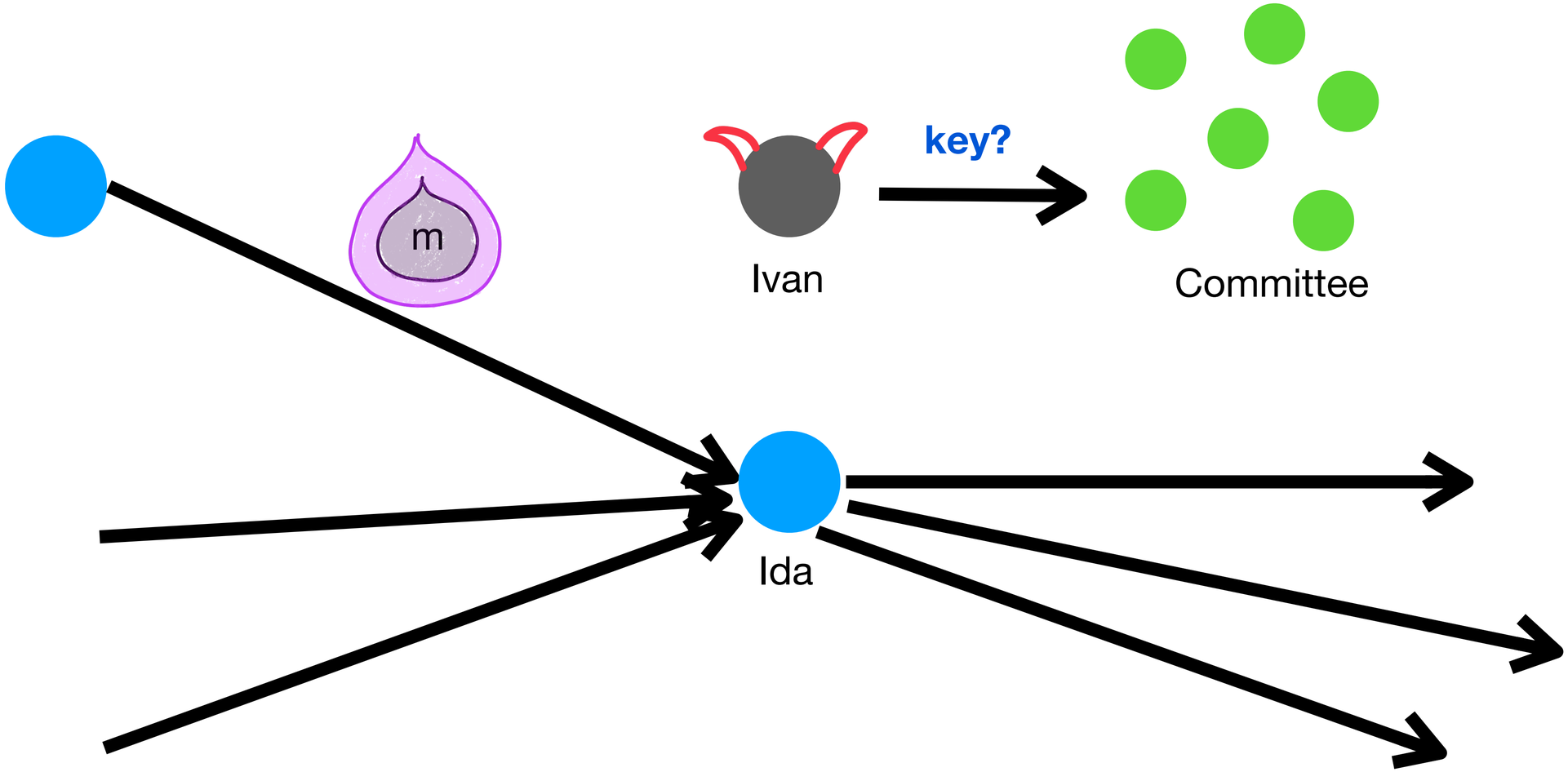
Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion



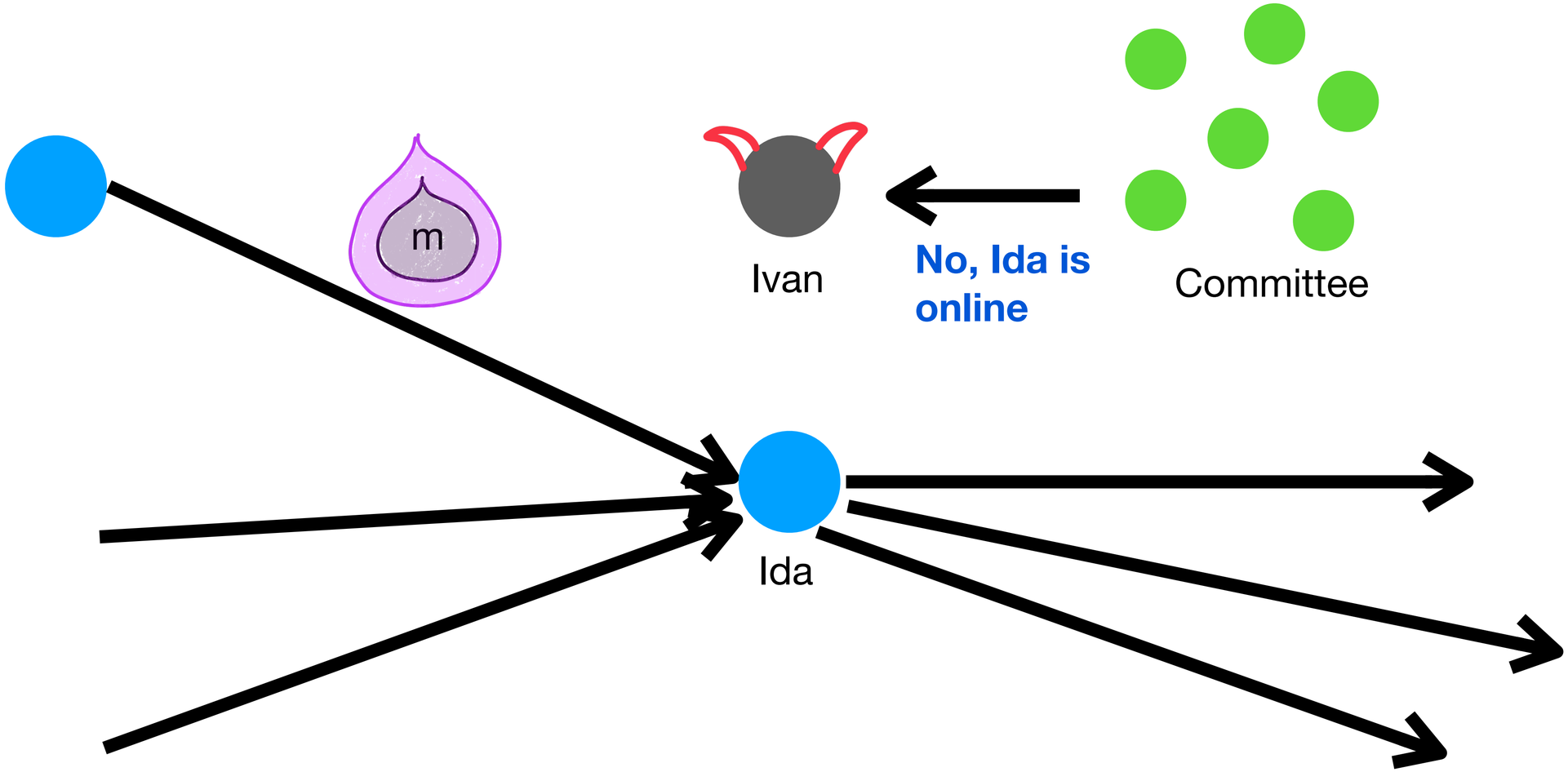
Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion



Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion



Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion
- Have mixing if primary candidate is **honest and online**

Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion
- Have mixing if primary candidate is **honest and online**
  - **Probability of this is independent of # candidates**

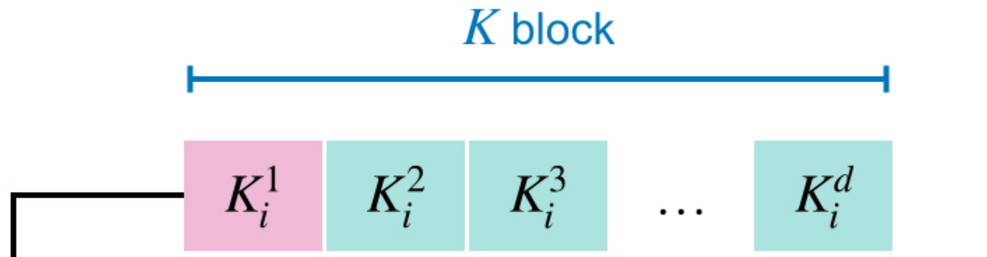
Our solution: poly onions

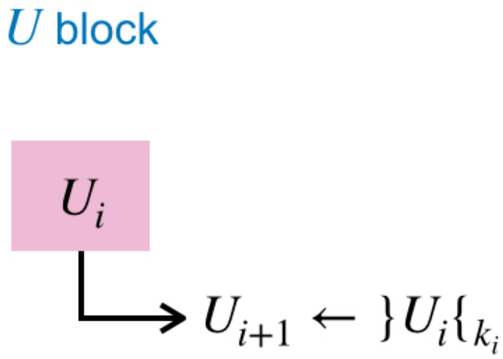
- An alternate candidate needs a key from a **committee** to peel the onion
- Have mixing if primary candidate is **honest and online**
  - **Probability of this is independent of # candidates**
- Can extend to any number of candidates

Our solution: poly onions

- An alternate candidate needs a key from a **committee** to peel the onion
- Have mixing if primary candidate is **honest and online**
  - **Probability of this is independent of # candidates**
- Can extend to any number of candidates
  - **Boosts probability that onion is not dropped**

Ida: first candidate
Ivan: second candidate



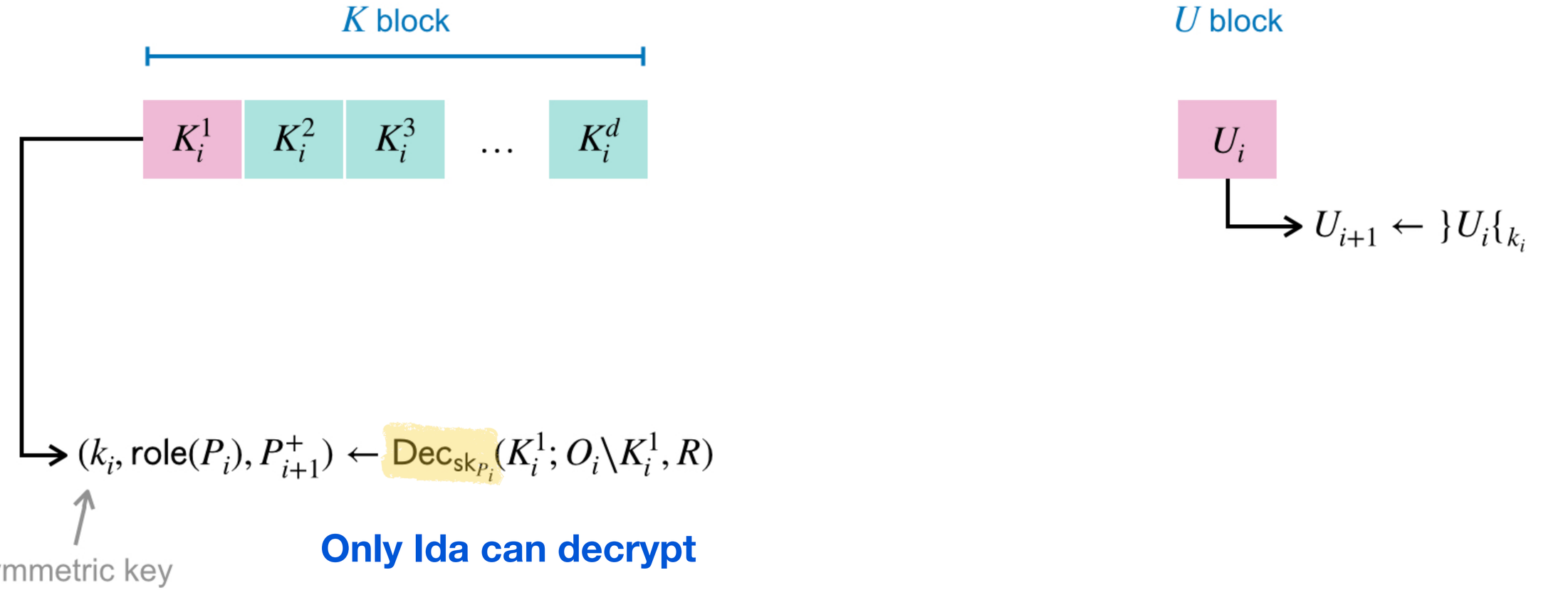


$(k_i, \text{role}(P_i), P_{i+1}^+) \leftarrow \text{Dec}_{\text{sk}_{P_i}}(K_i^1; O_i \backslash K_i^1, R)$

Symmetric key

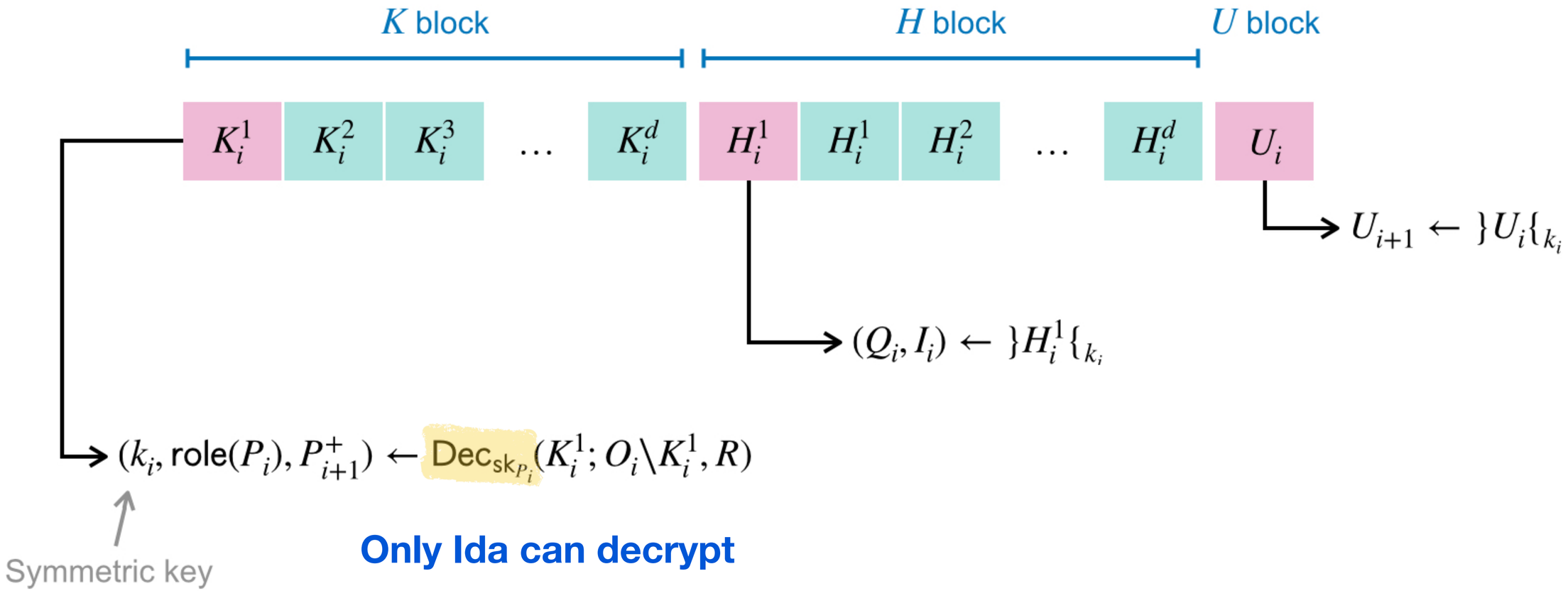
Needed to peel rest of onion

Ida: first candidate
Ivan: second candidate



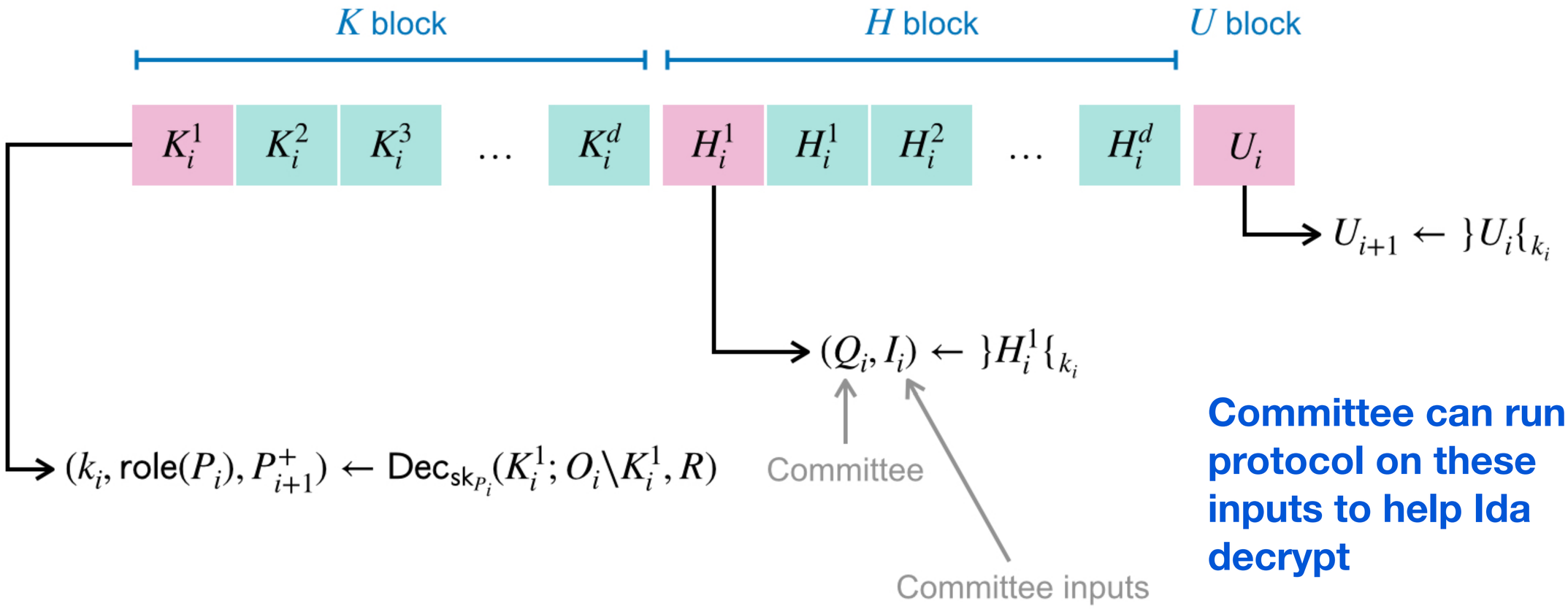
Only Ida can decrypt

Ida: first candidate
Ivan: second candidate



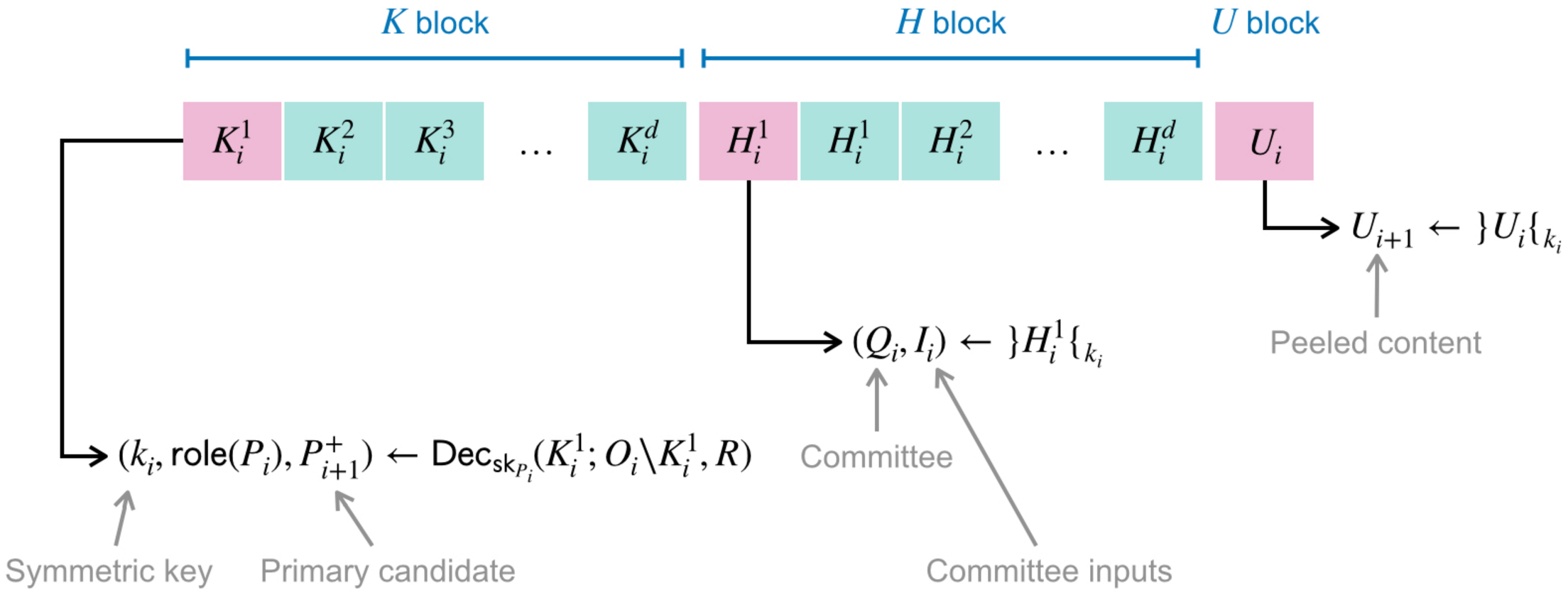
Only Ida can decrypt

Ida: first candidate
Ivan: second candidate



Committee can run protocol on these inputs to help Ida decrypt

Ida: first candidate
Ivan: second candidate



Summary: our contributions

- Introduce anonymity definitions for the multi-run setting with network churn
- Show that for a natural class of onion routing protocols, single-run anonymity implies multi-run anonymity
- Define poly onion encryption (I/O, security defs)
- Construct a poly onion encryption scheme
- Apply Poly Onion Encryption to a known onion routing protocol to obtain a protocol that is anonymous against a passive adversary, with churn

Thank you!

ePrint: 2022/392

Luna, Megumi’s daughter and onion routing enthusiast