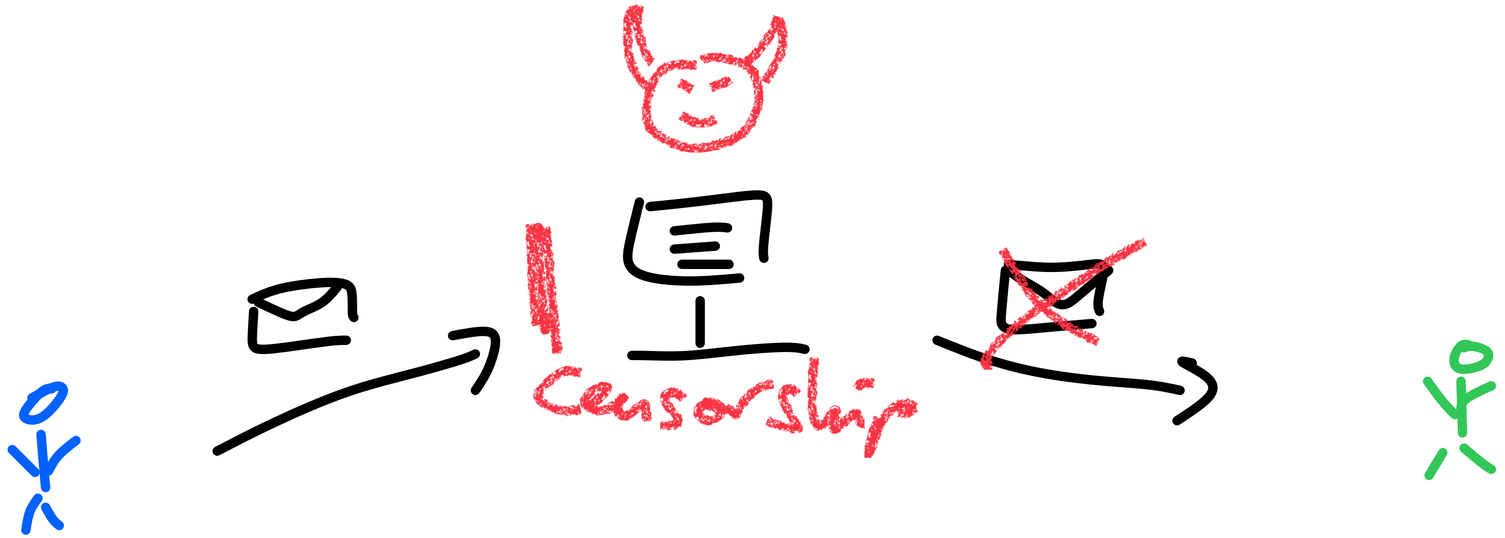
ASMesh: Anonymous and Secure Messaging in Mesh Networks Using Stronger, Anonymous Double Ratchet

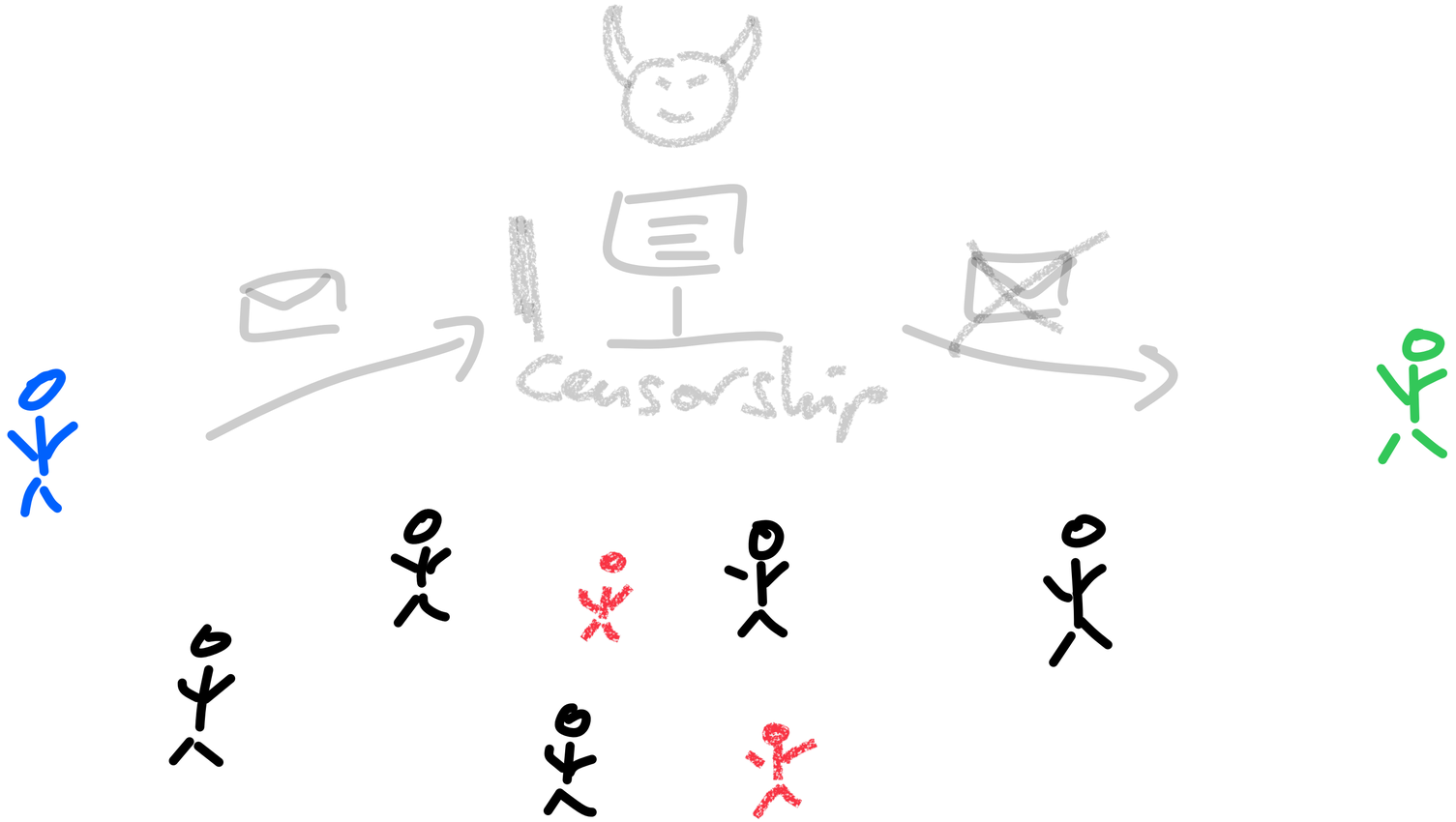
Mathematics and CS Seminar

Paul Rösler 26.09.2023

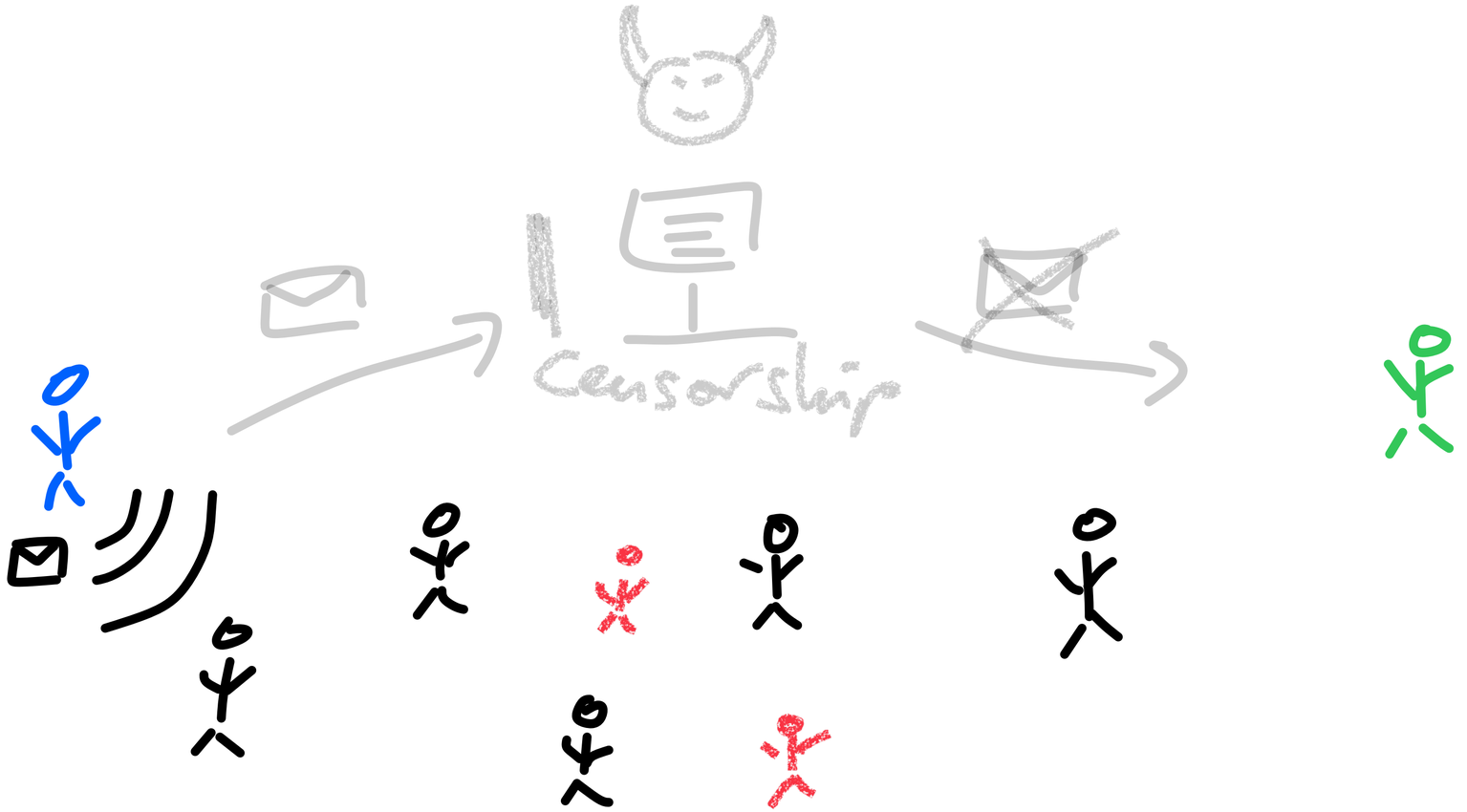
Mesh Networks



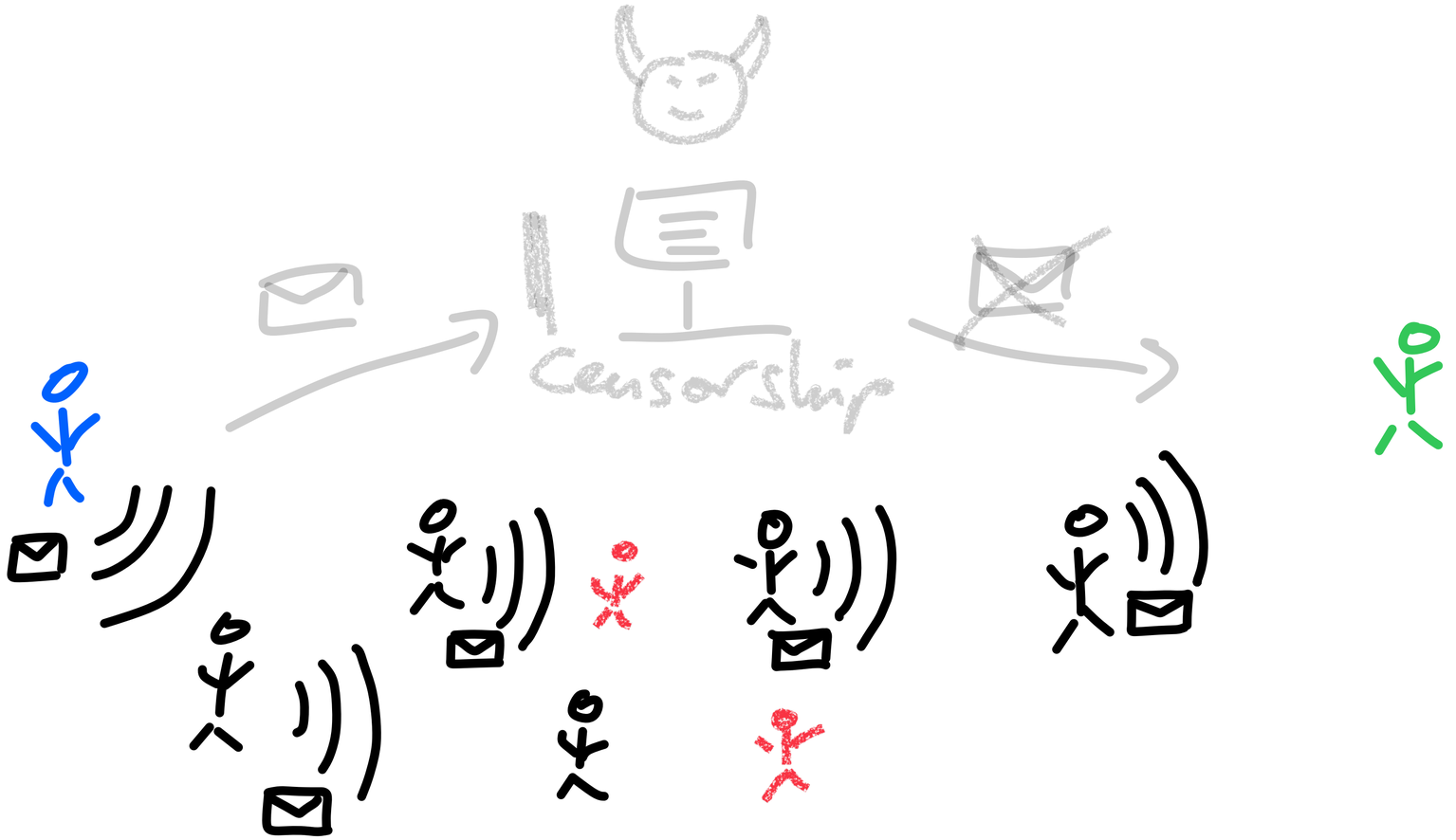
Mesh Networks



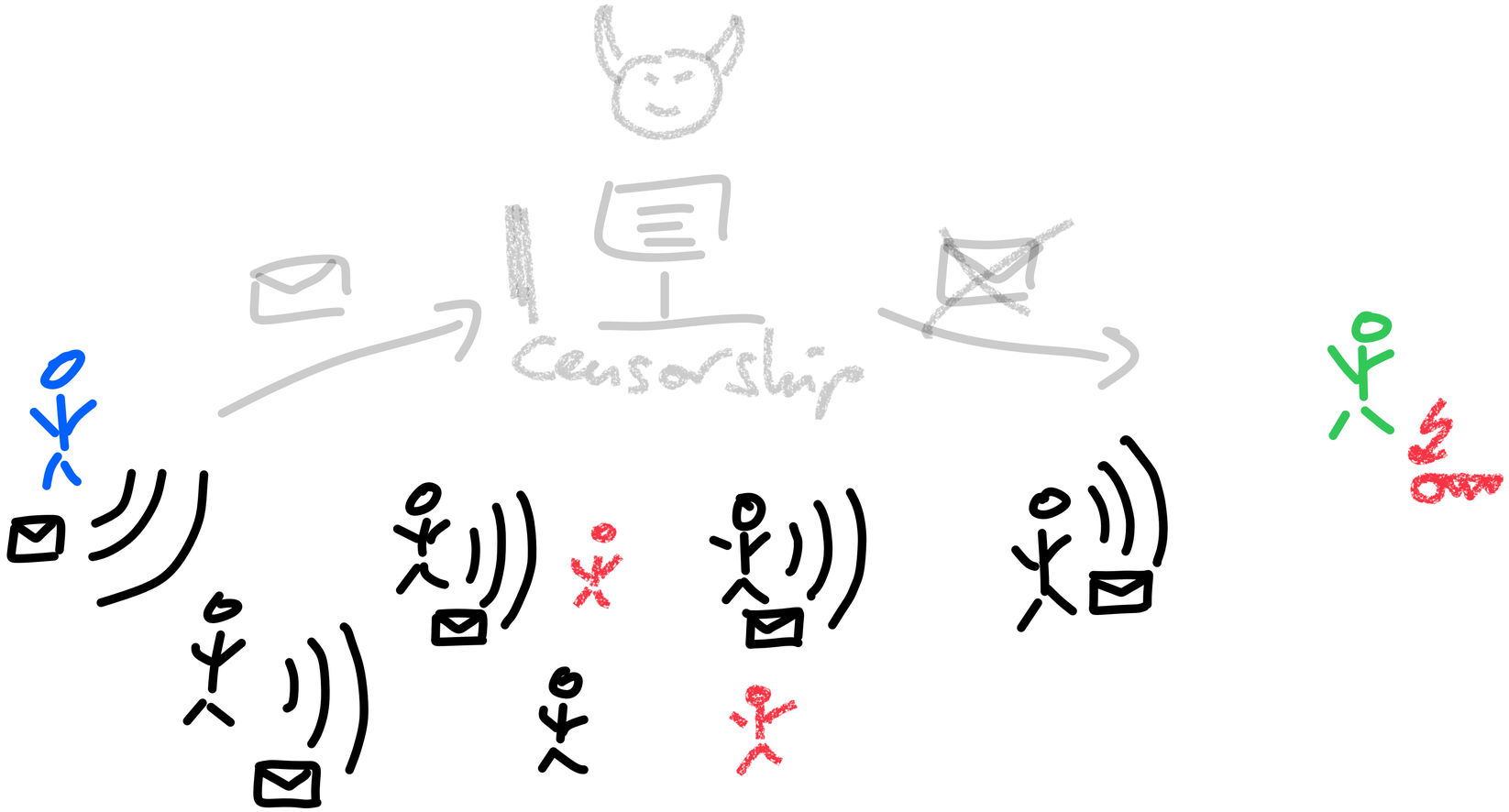
Mesh Networks



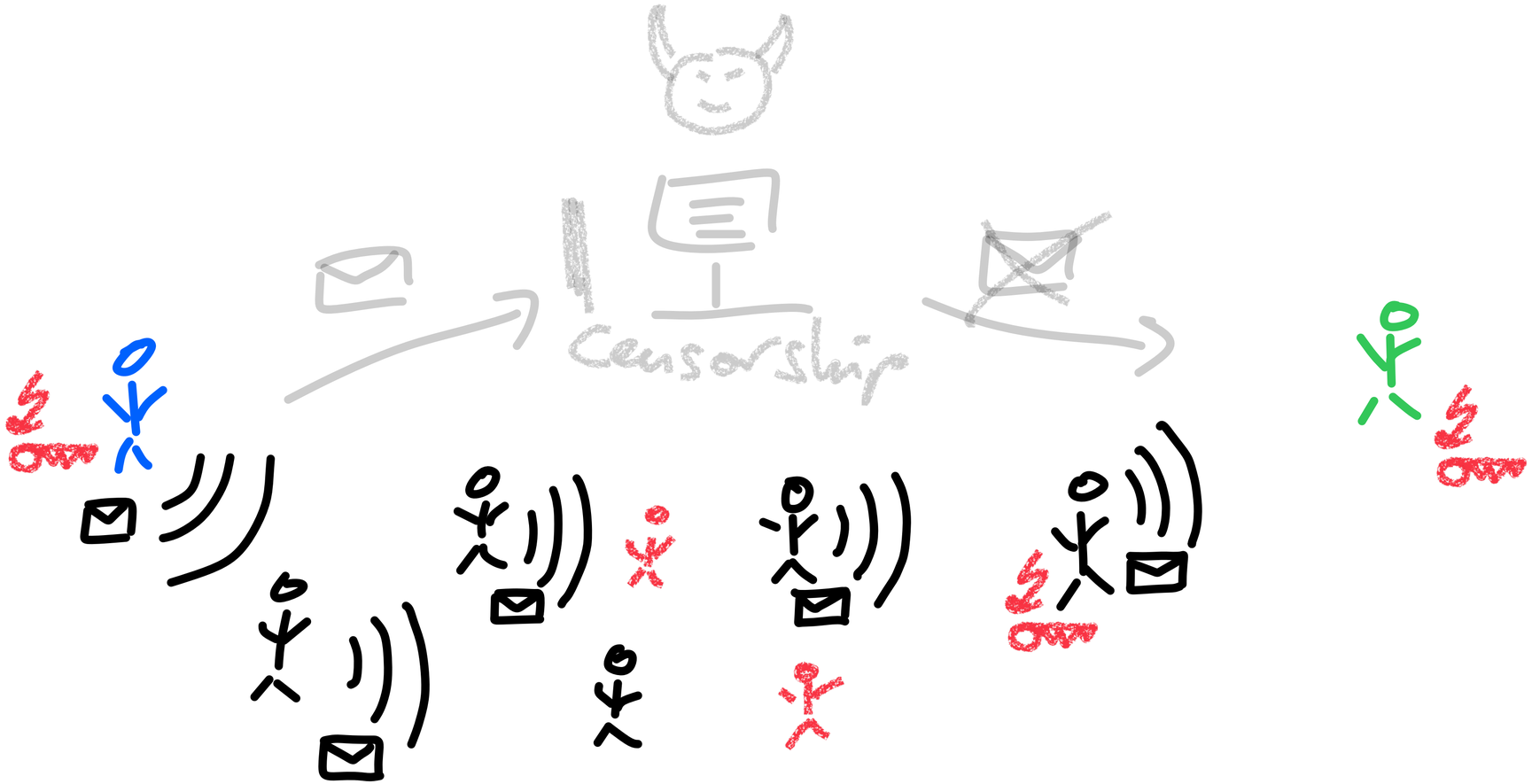
Mesh Networks



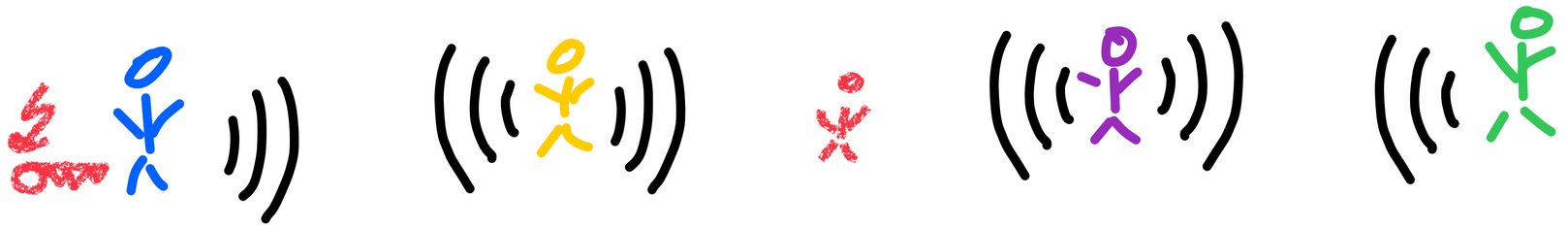
Mesh Networks



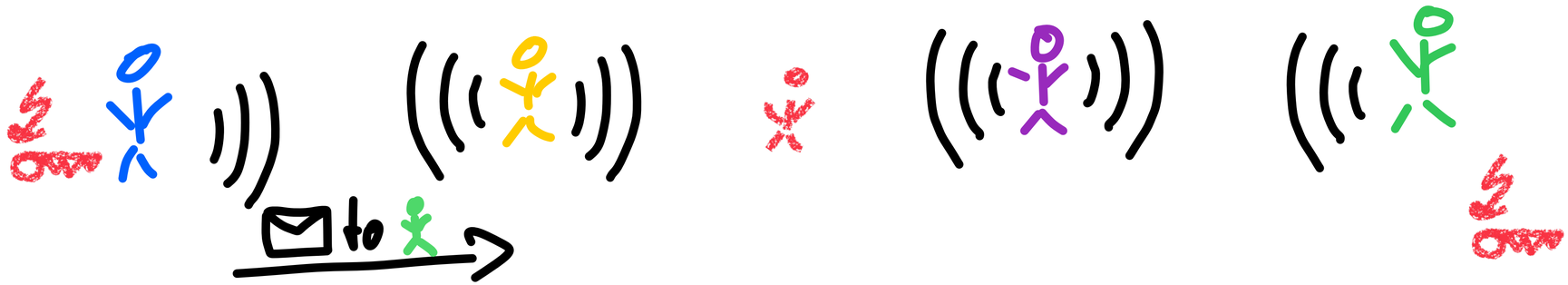
Mesh Networks



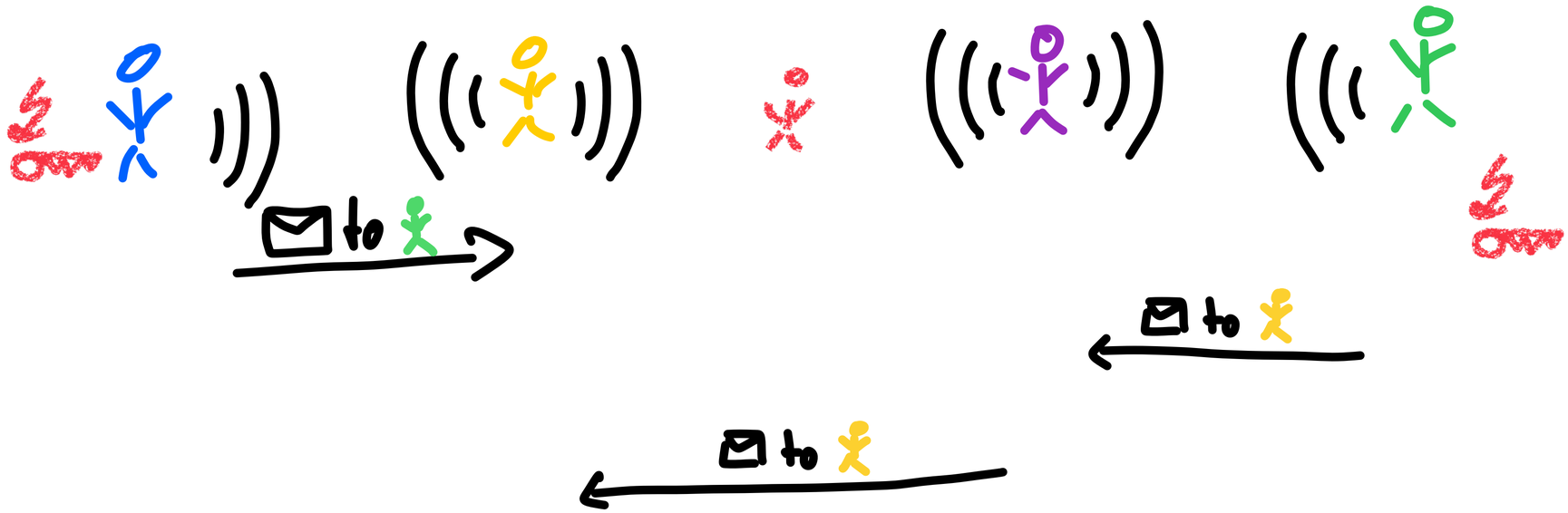
Mesh Networks: Security



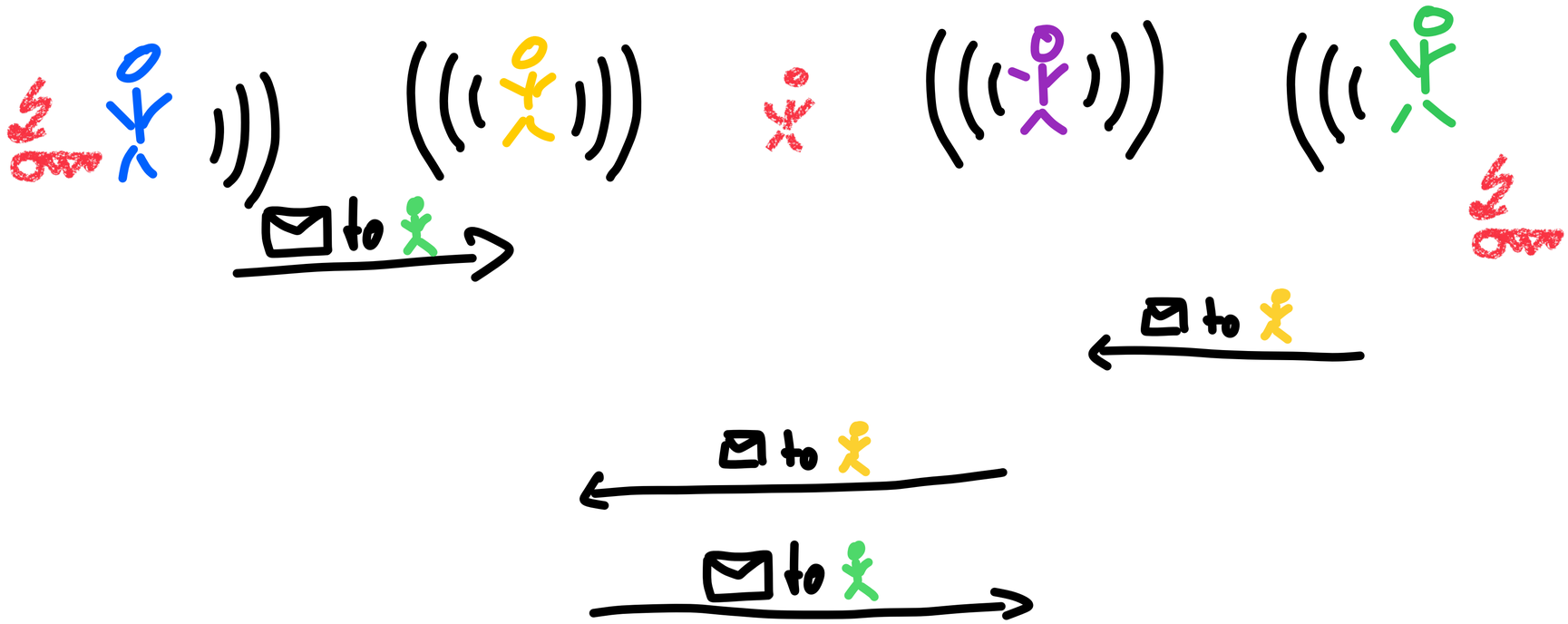
Mesh Networks: Security



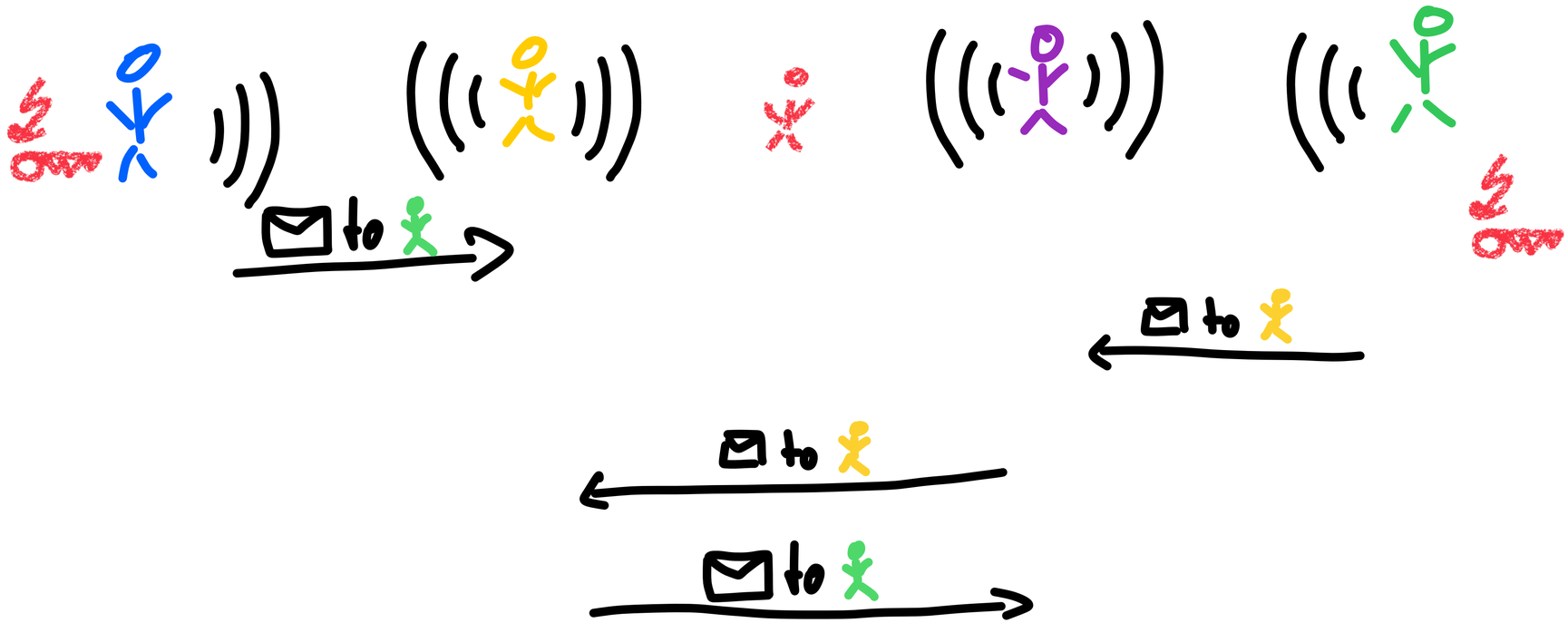
Mesh Networks: Security



Mesh Networks: Security

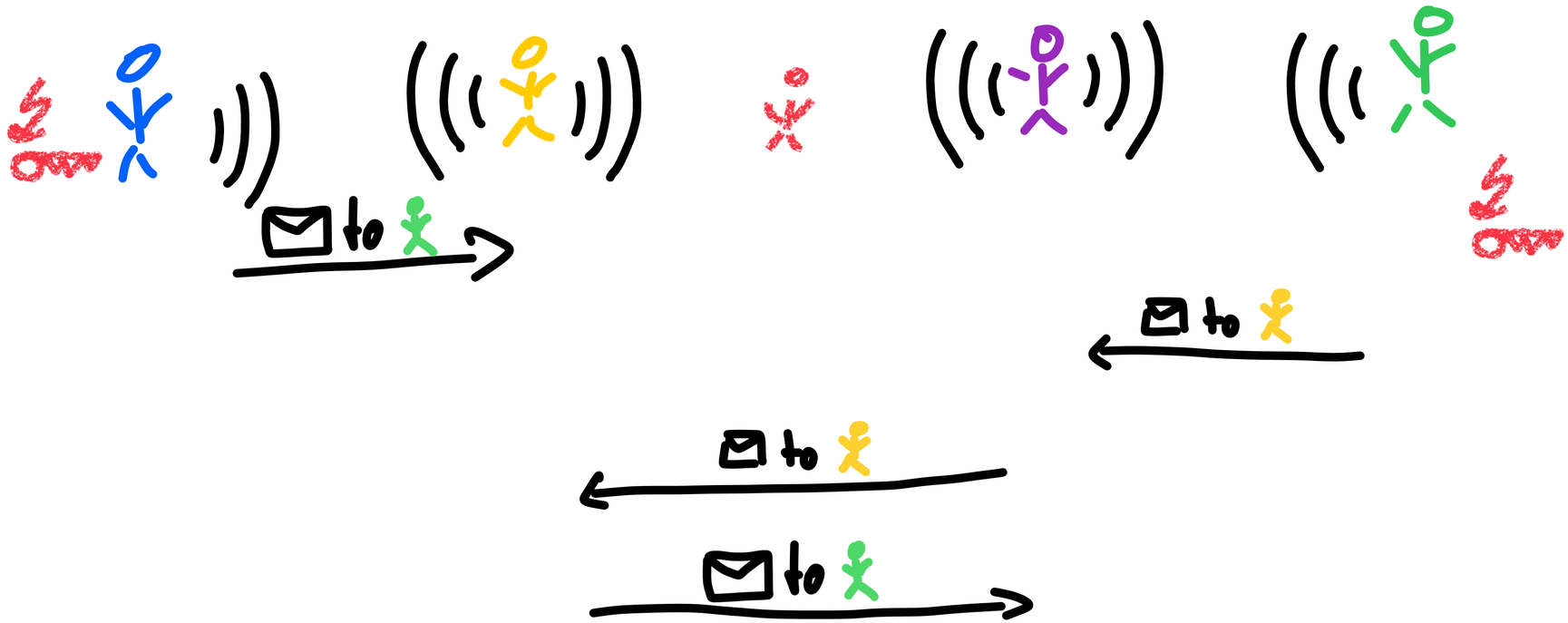


Mesh Networks: Security



Confidential: 's update w/ protocol to "Contact Cooperation"

Mesh Networks: Security



Confidential: 's update w/ protocol to "Contact Cooperation"

Anonymous: 's ID hidden (all sender IDs)

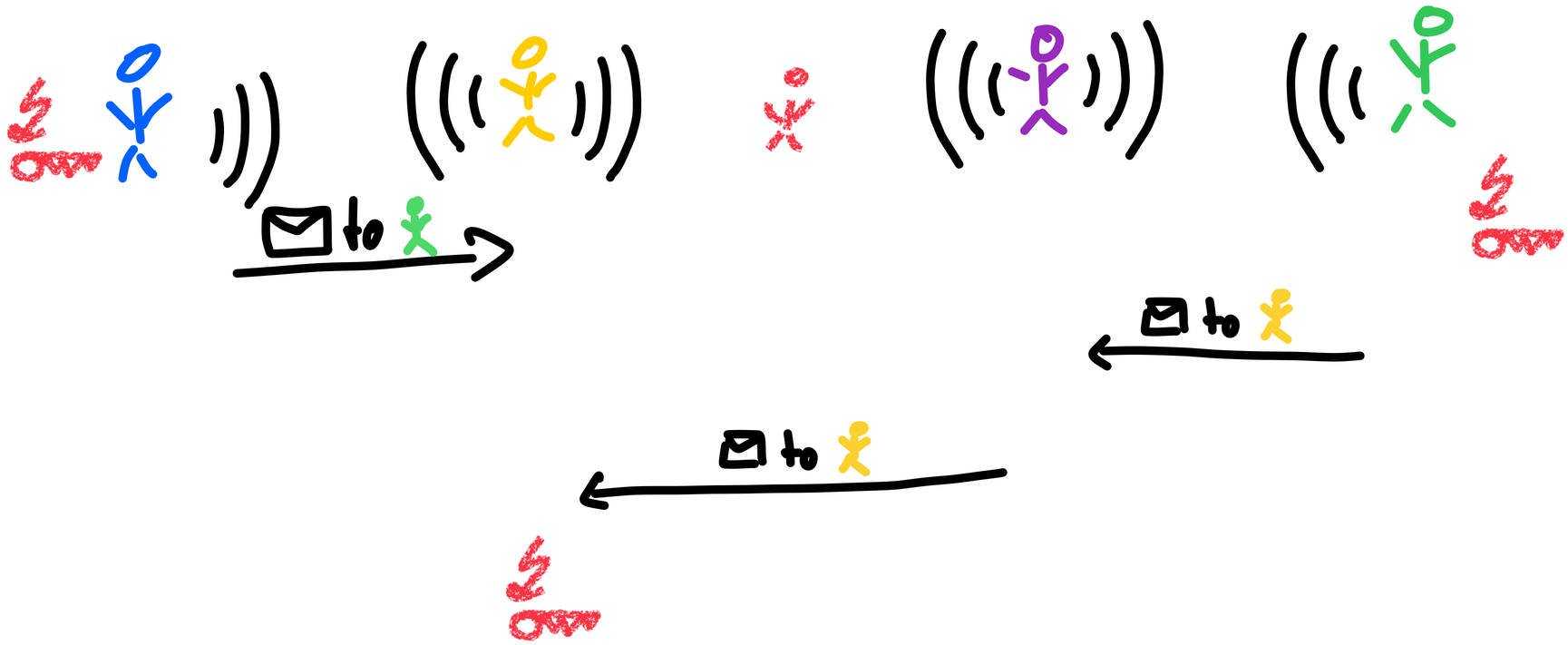
Mesh Networks: Security

Confidential: 's update w/ protects to "Contact Cooperation"

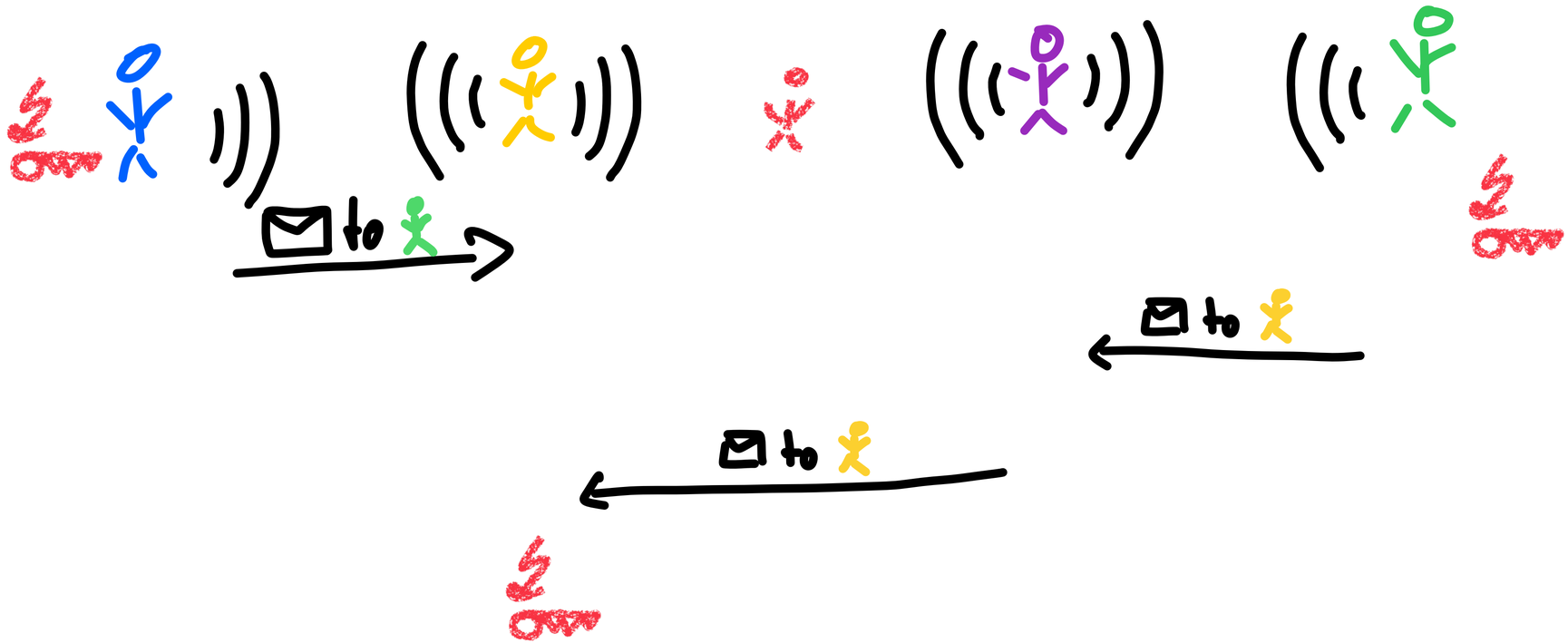
Anonymous: 's ID hidden (all sender IDs)

Alternative: Full Anonymity w/o Contact Cooperation

Mesh Networks: Security Model



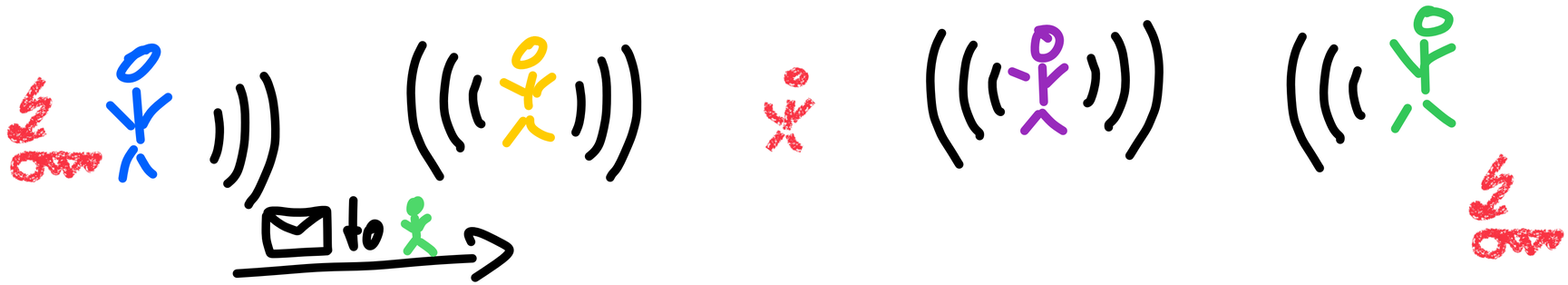
Mesh Networks: Security Model



Exposed State contains "Anonymous Challenge Ciphertext"

↳ Embedded construction dependent

Mesh Networks: Security Model



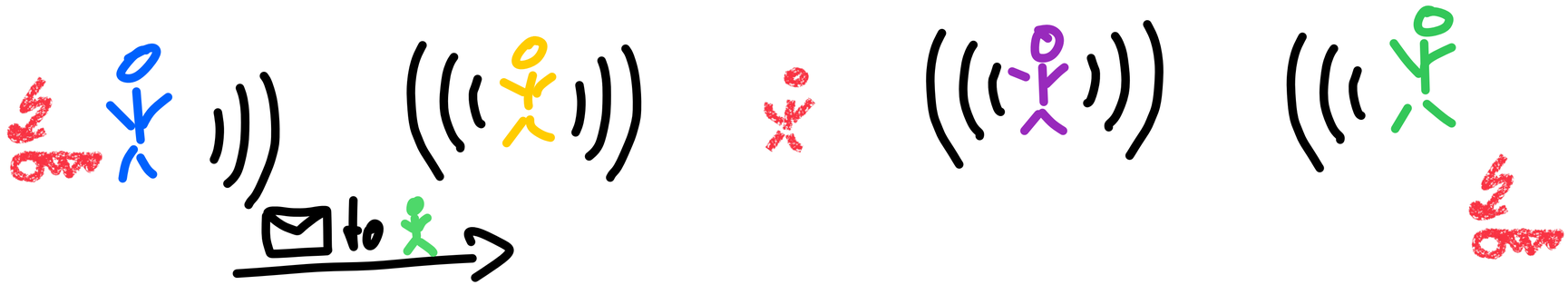
Exposed State contains "Anonymous Challenge Ciphertext"

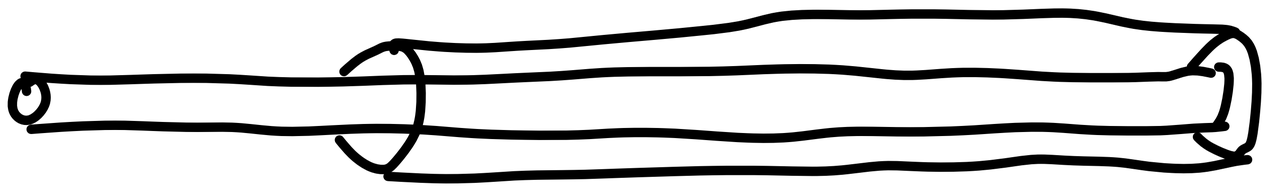
↳ Embedded construction dependent

↳ Unnatural for frame-based Model

↳ Simulation-Based

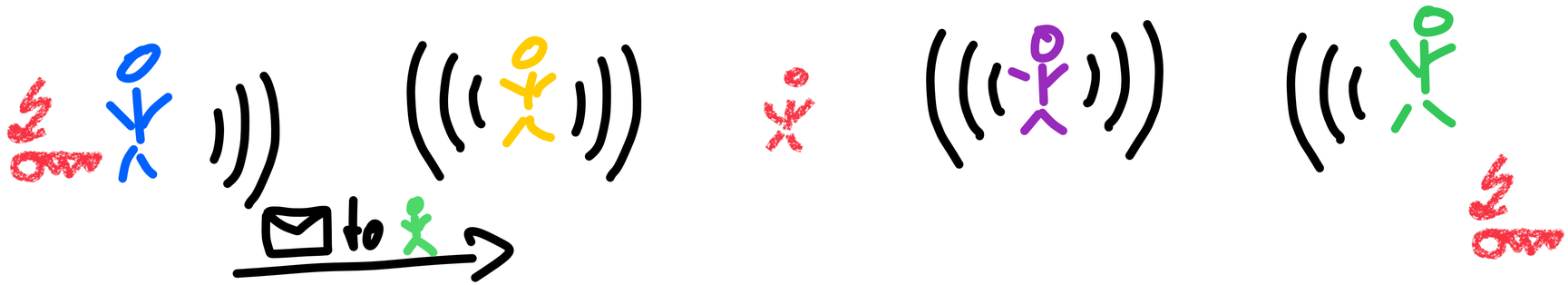
Mesh Networks: Construction

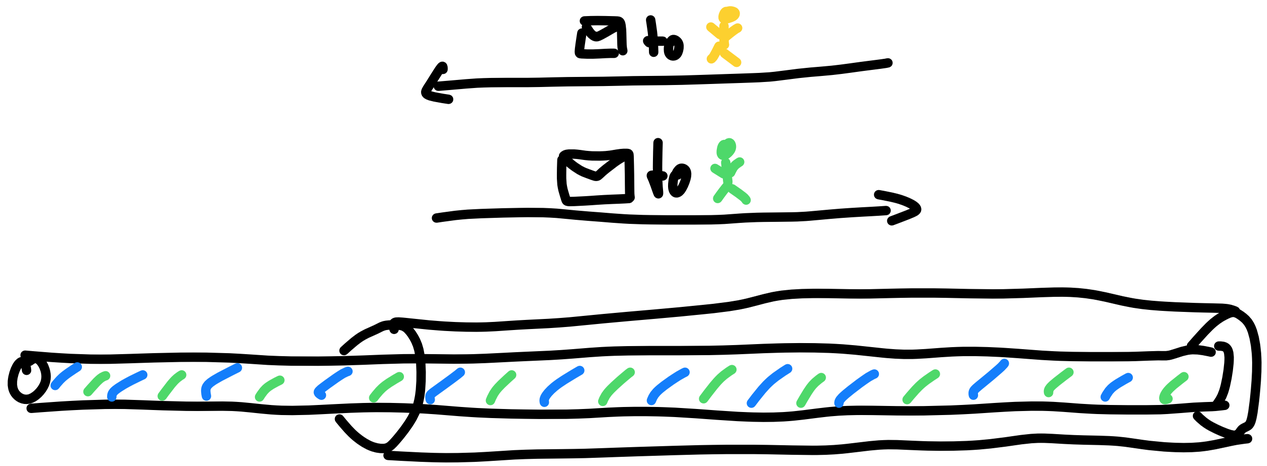




Nested Double Ratchet

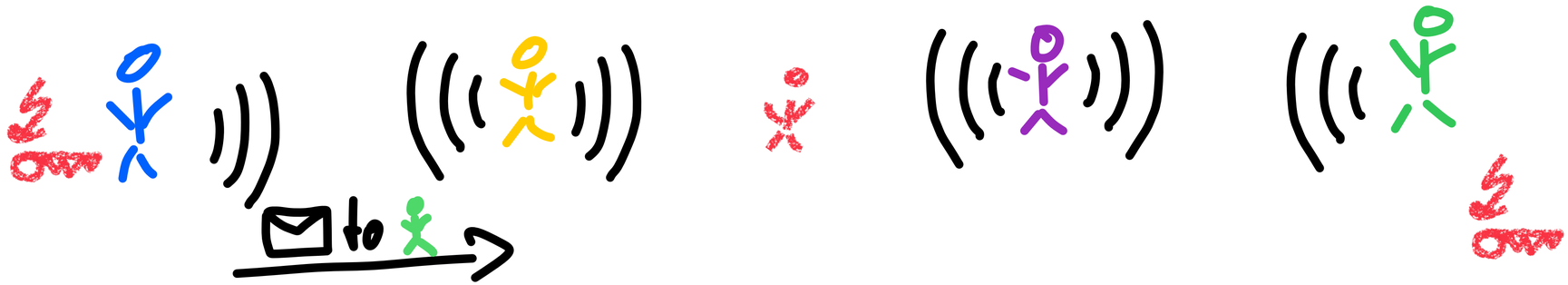
Mesh Networks: Construction

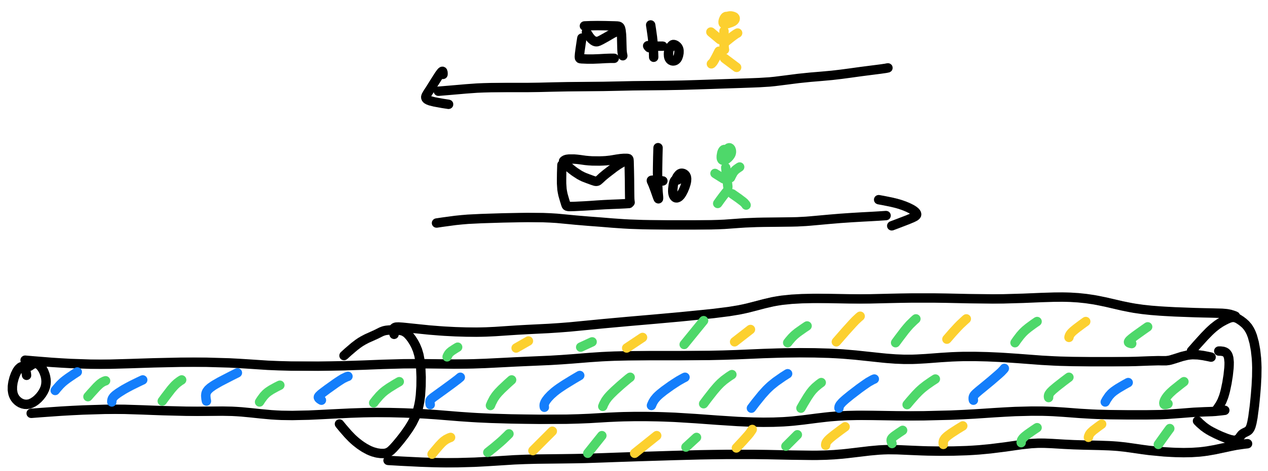




Nested Double Ratchet

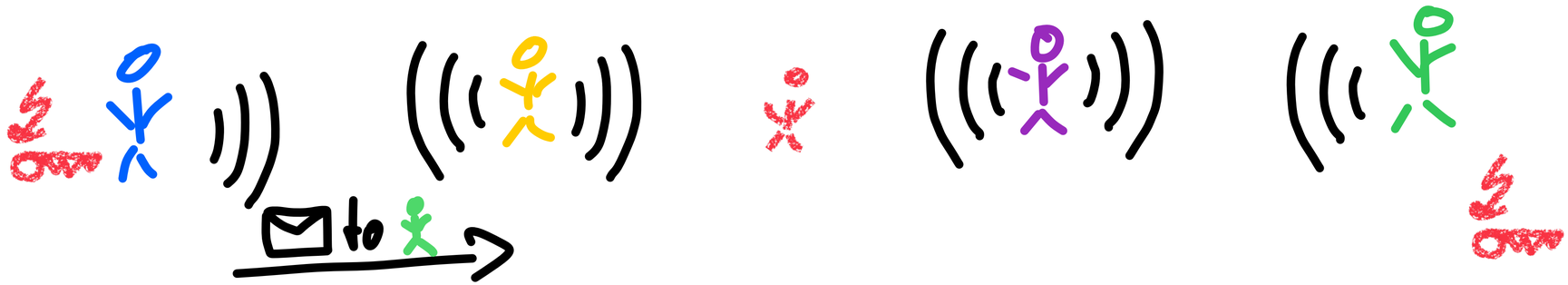
Mesh Networks: Construction

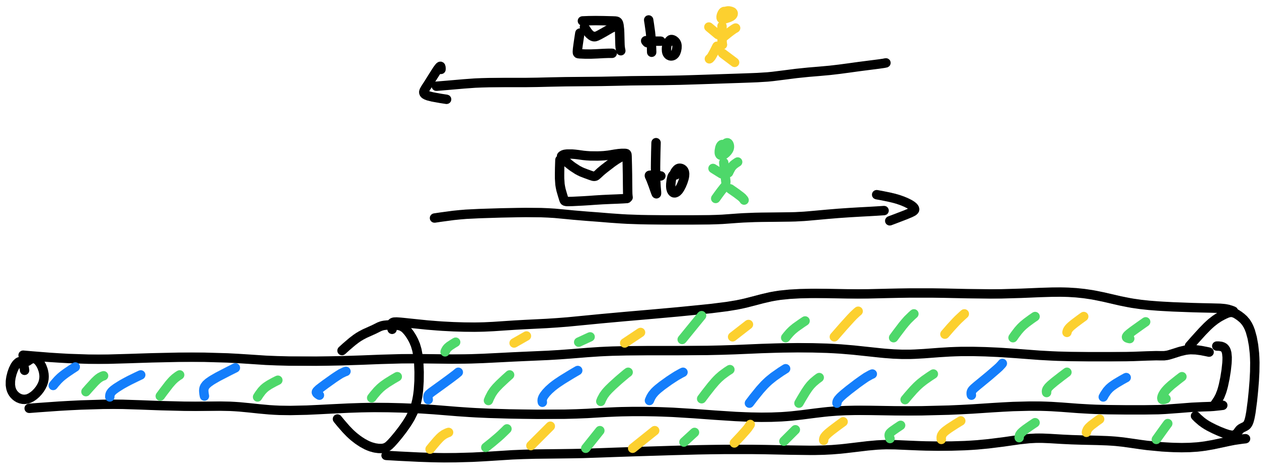




Nested Double Ratchet

Mesh Networks: Construction

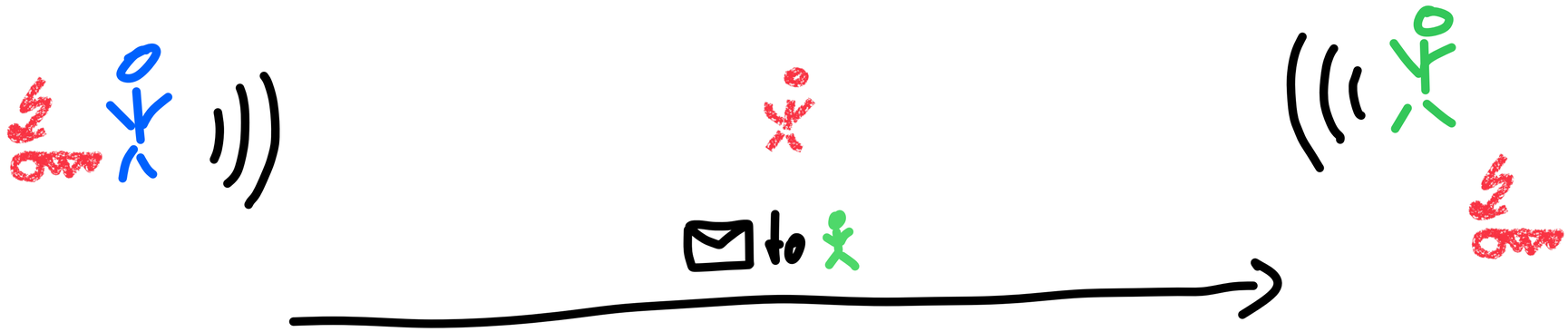




Nested Double Ratchet

+ PKE for every message
+ New pk DH every message
+ ...
⇒ Strong (ish) FS & PCS

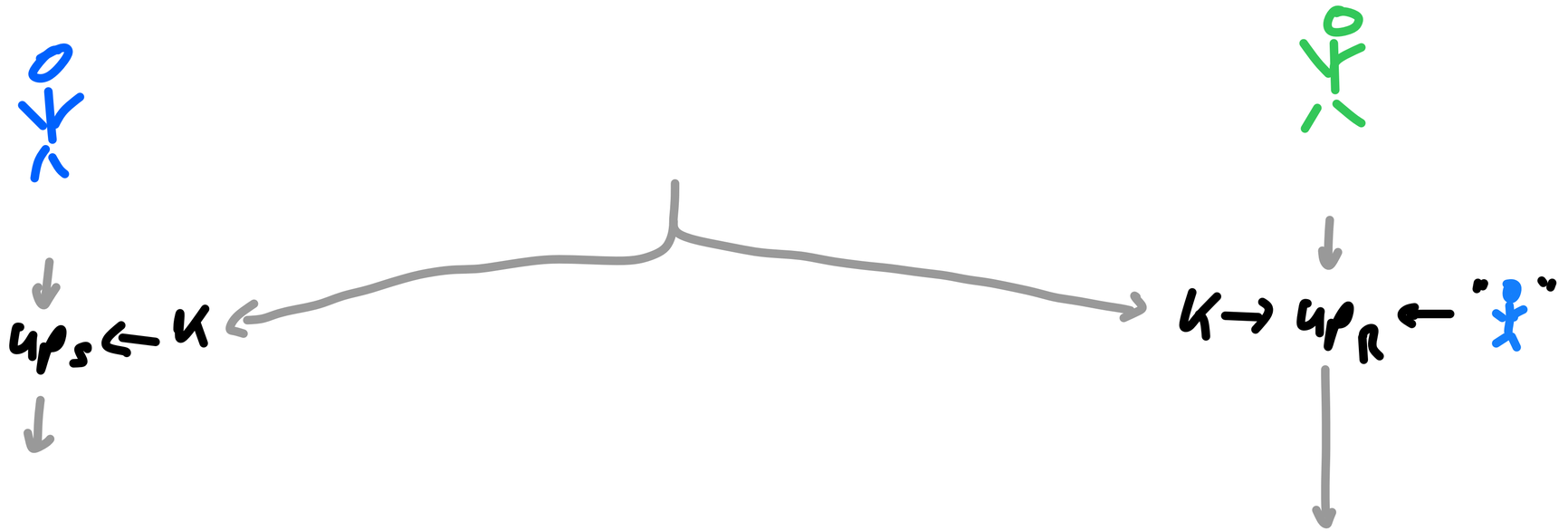
Anonymity: Related Work



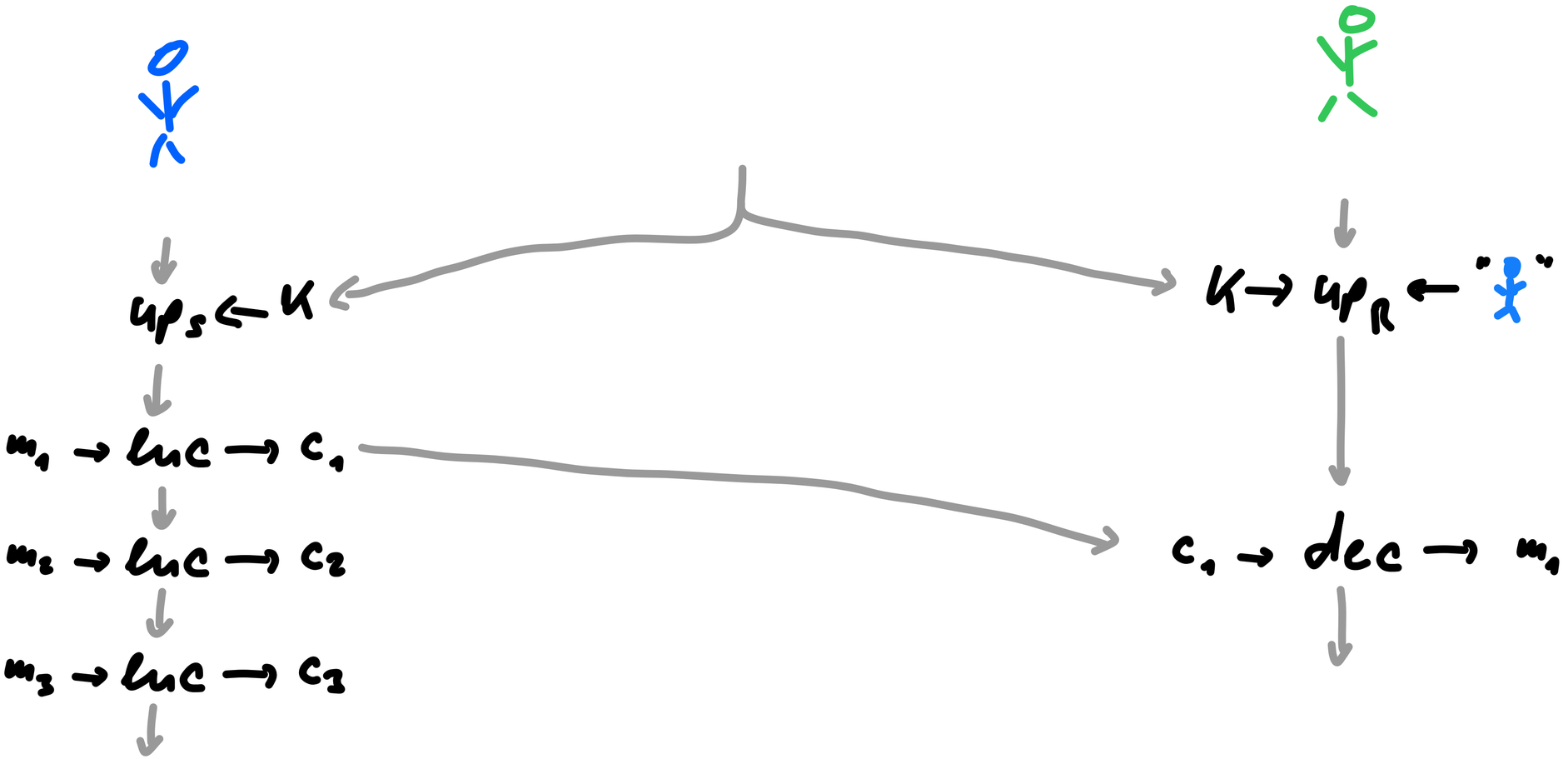
- Double Ratchet: Re-send g^a + Counters ⇒ No Anon
- Sealed Sender: PKE to static Key ⇒ No FS / PCS
- DHRR'22: Unidirectional reliable communication ⇒ unsuitable for mesh networks

Anonymity: Message Anonymizer Syntax

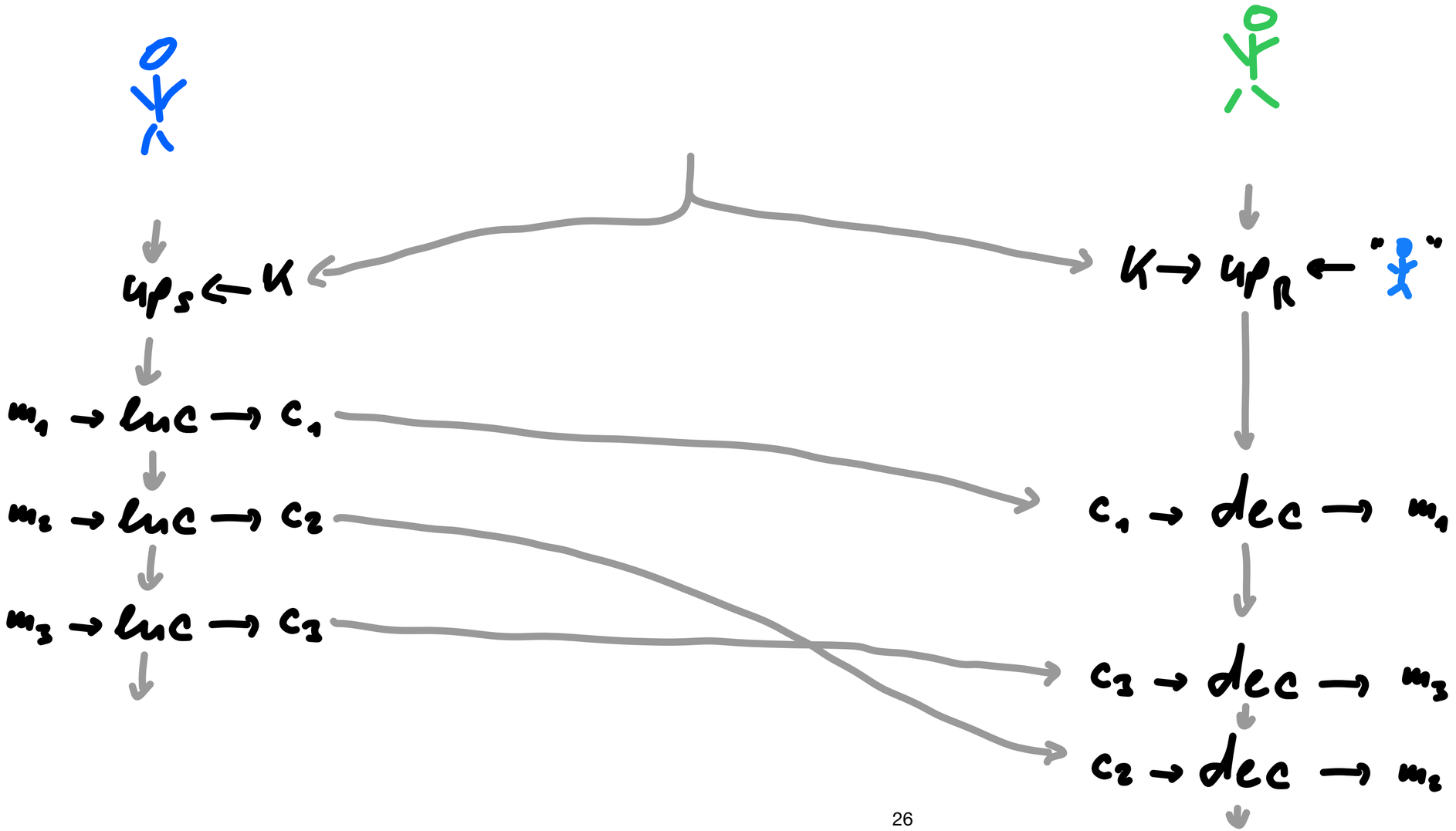
Anonymity: Message Anonymizer



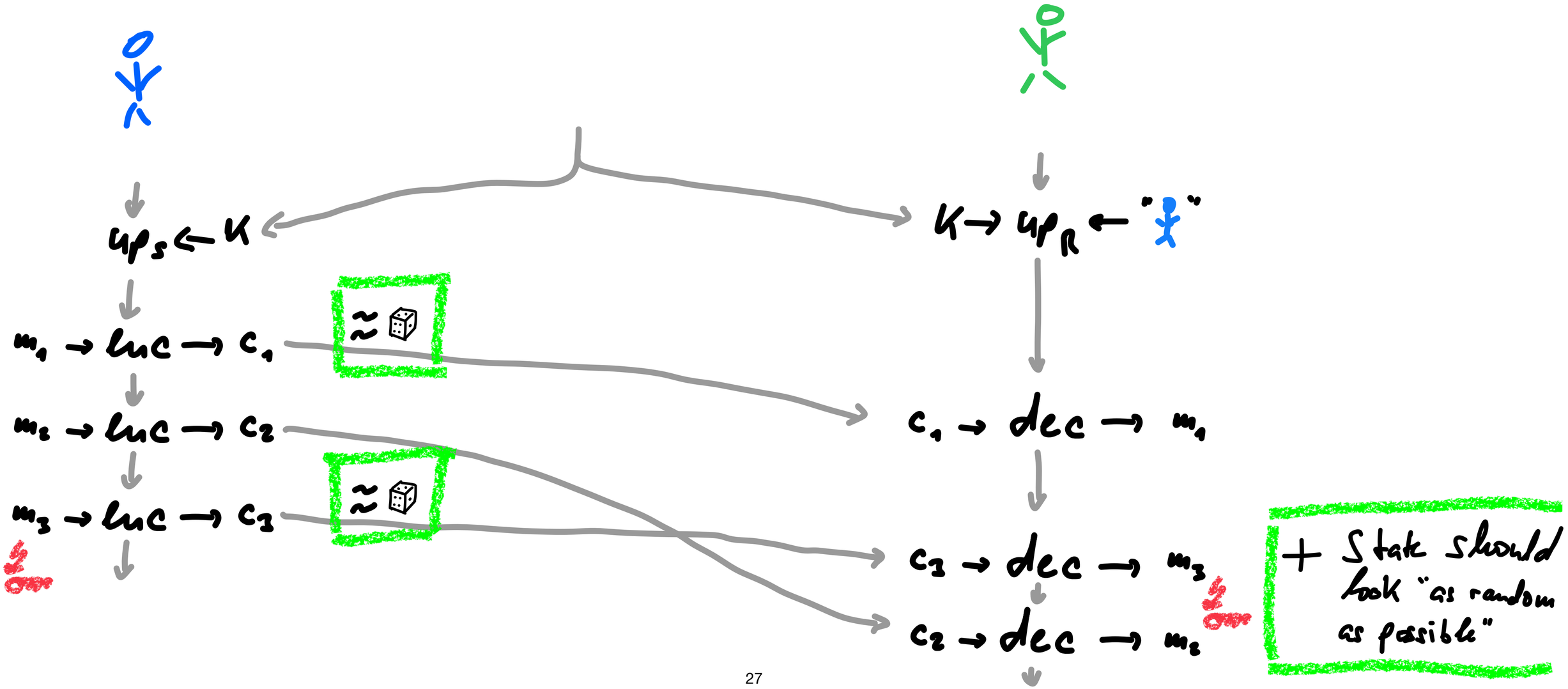
Anonymity: Message Anonymizer



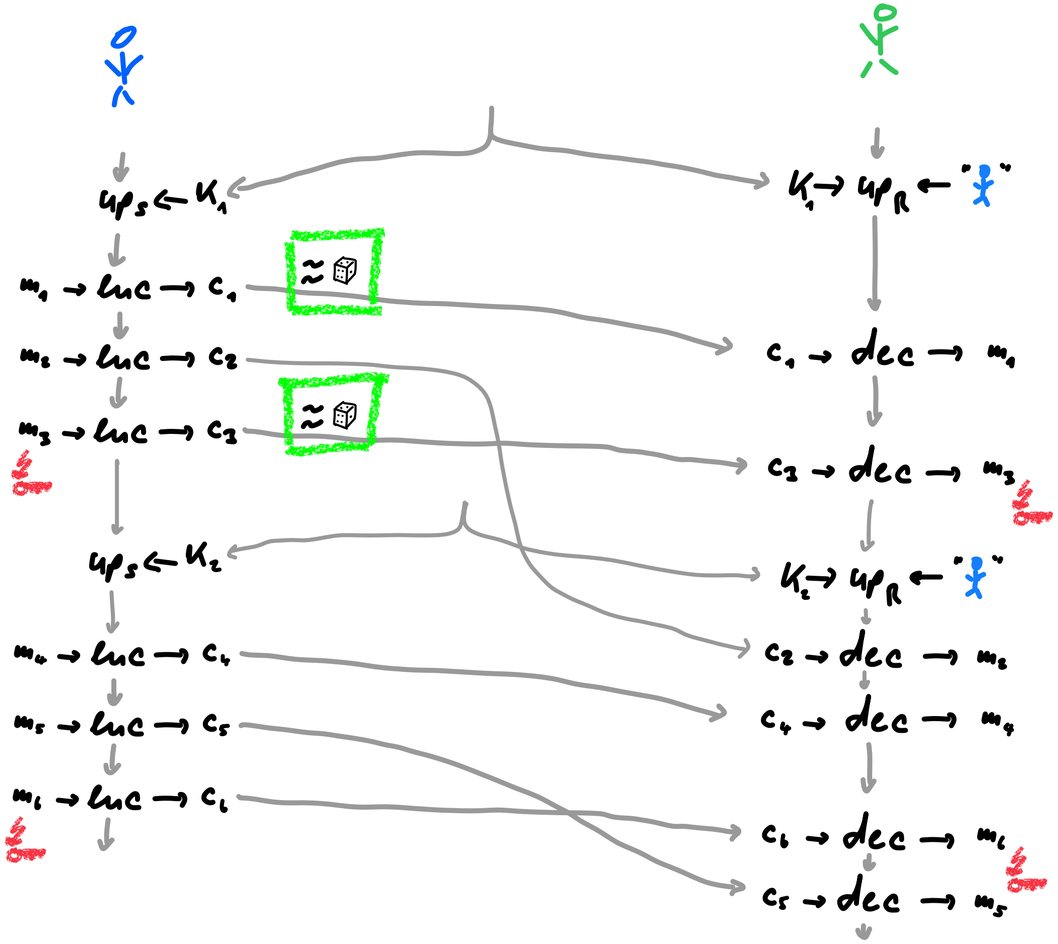
Anonymity: Message Anonymizer



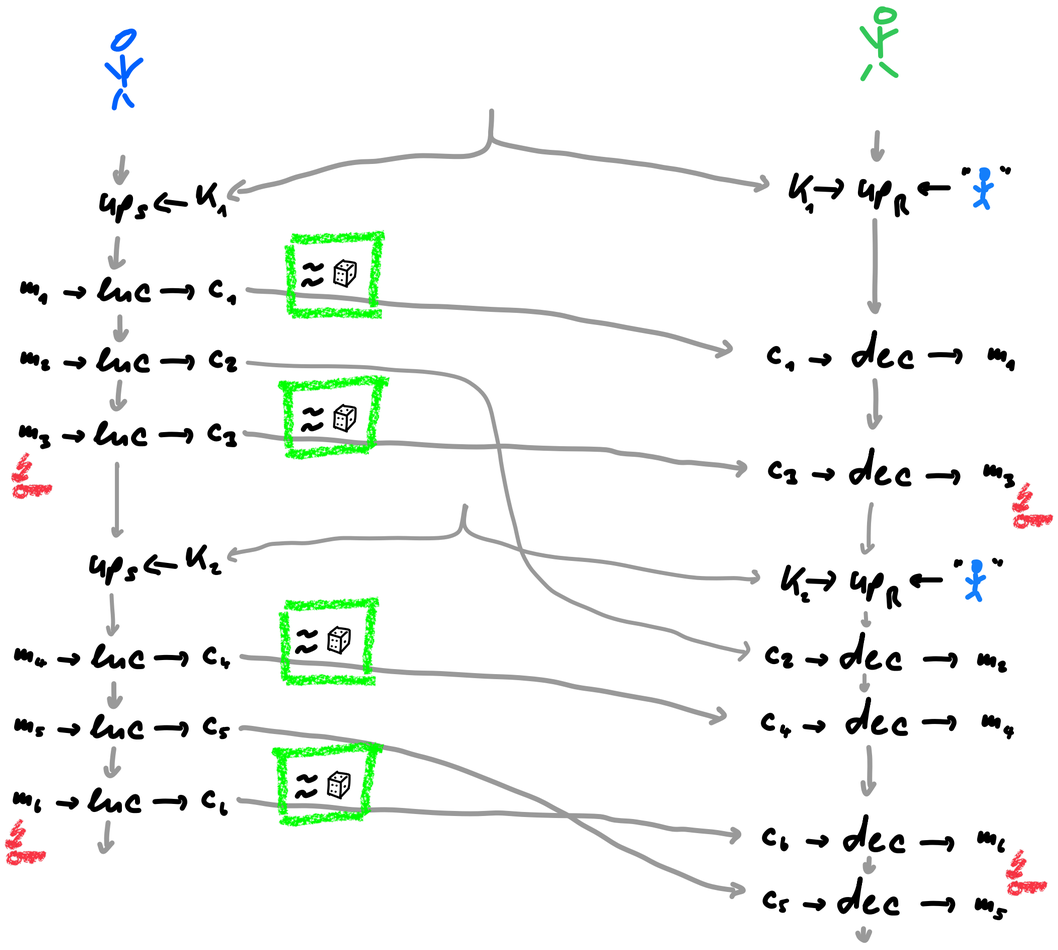
Anonymity: Message Anonymizer Security



Anonymity: Message Anonymizer

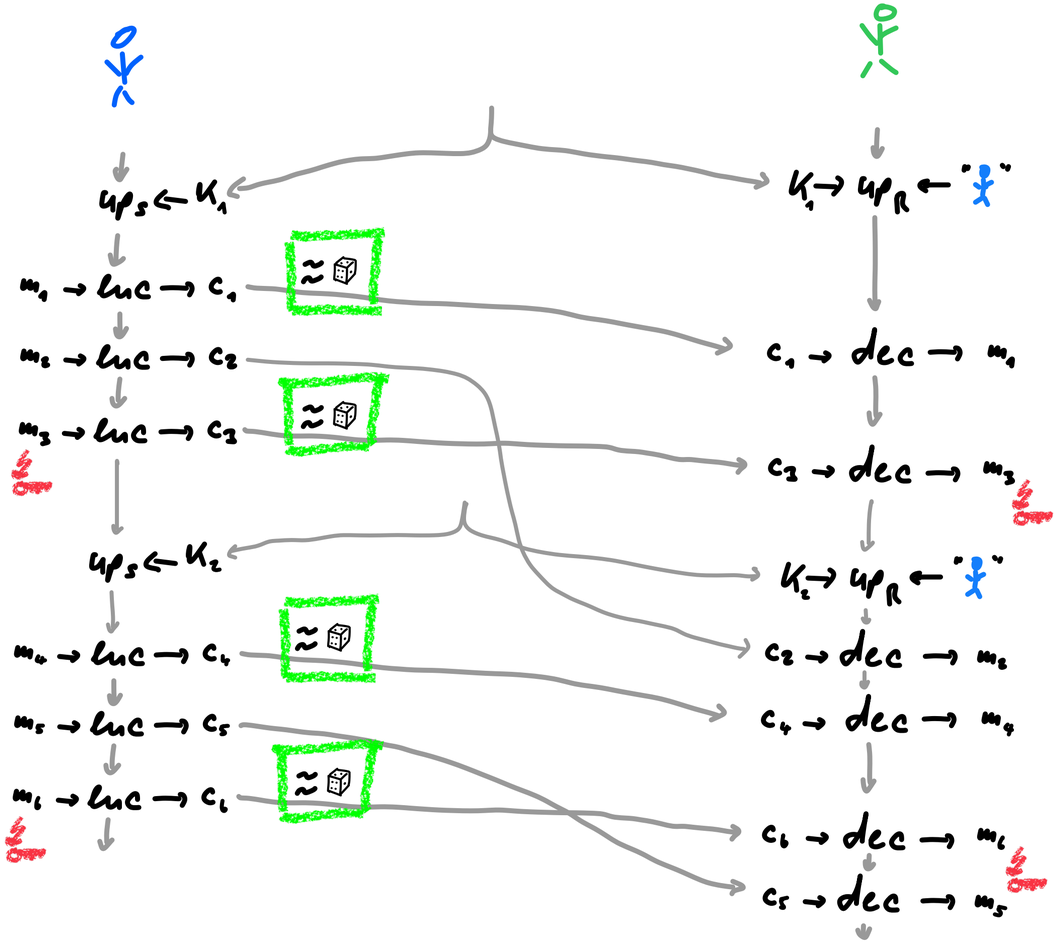


Anonymity: Message Anonymizer

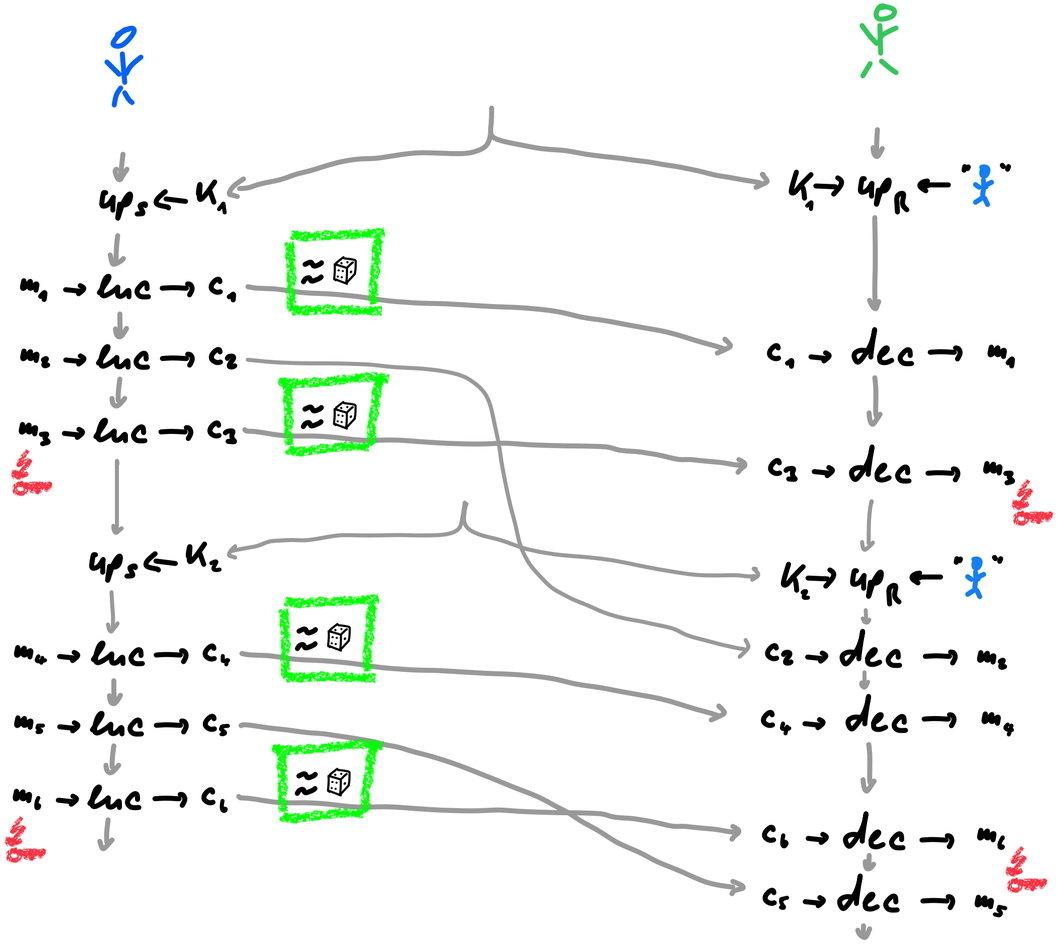


Anonymity: Message Anonymizer

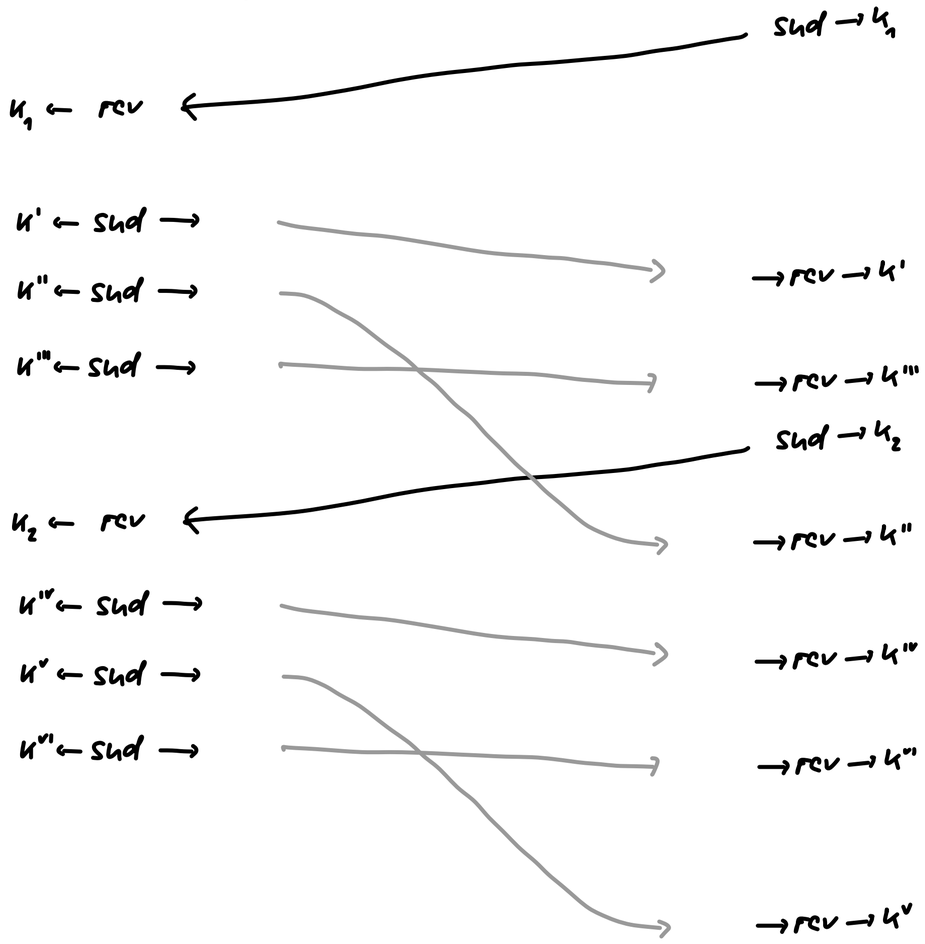
Composition w/ DR



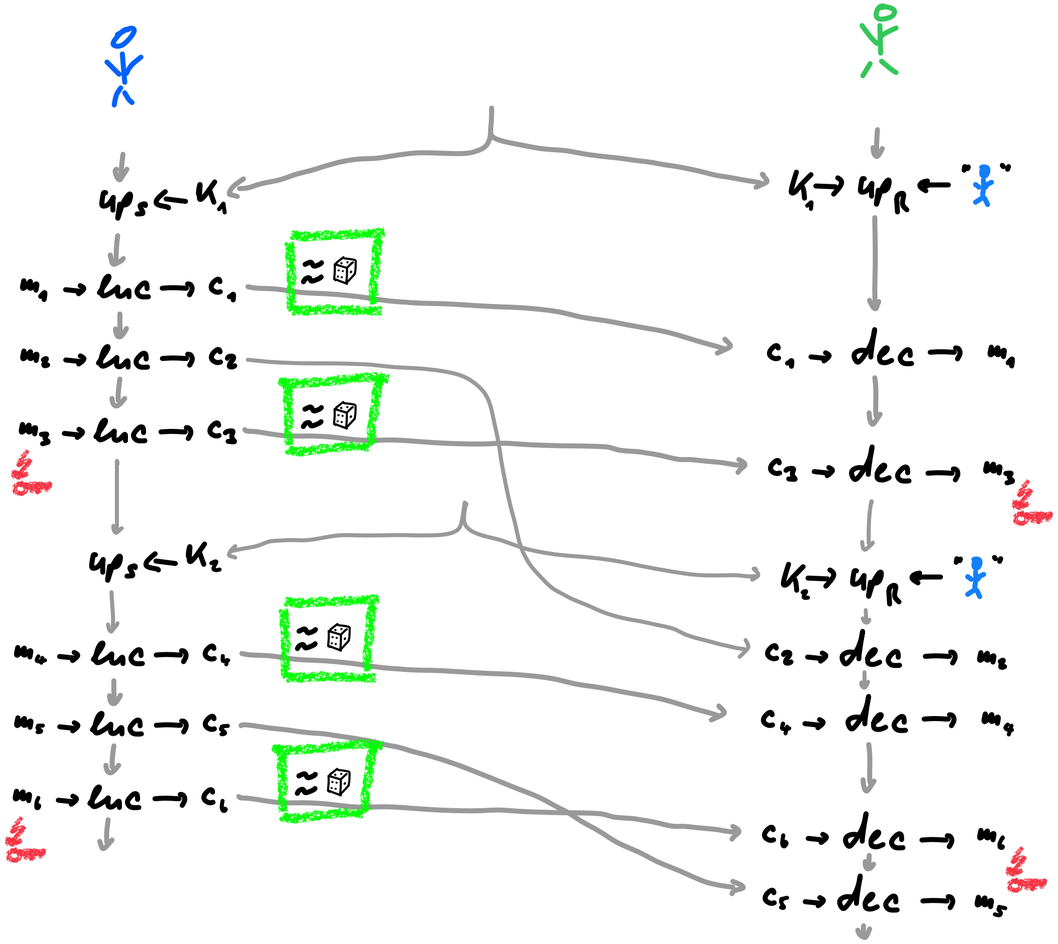
Anonymity: Message Anonymizer



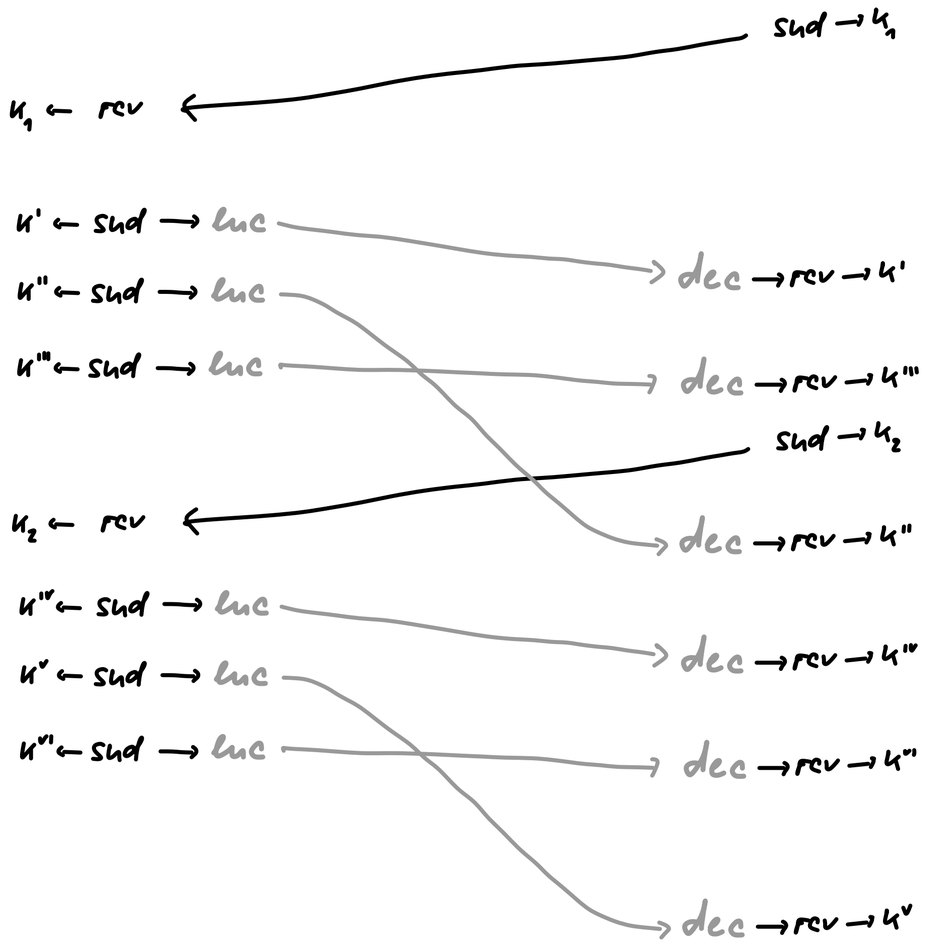
Composition w/ DR



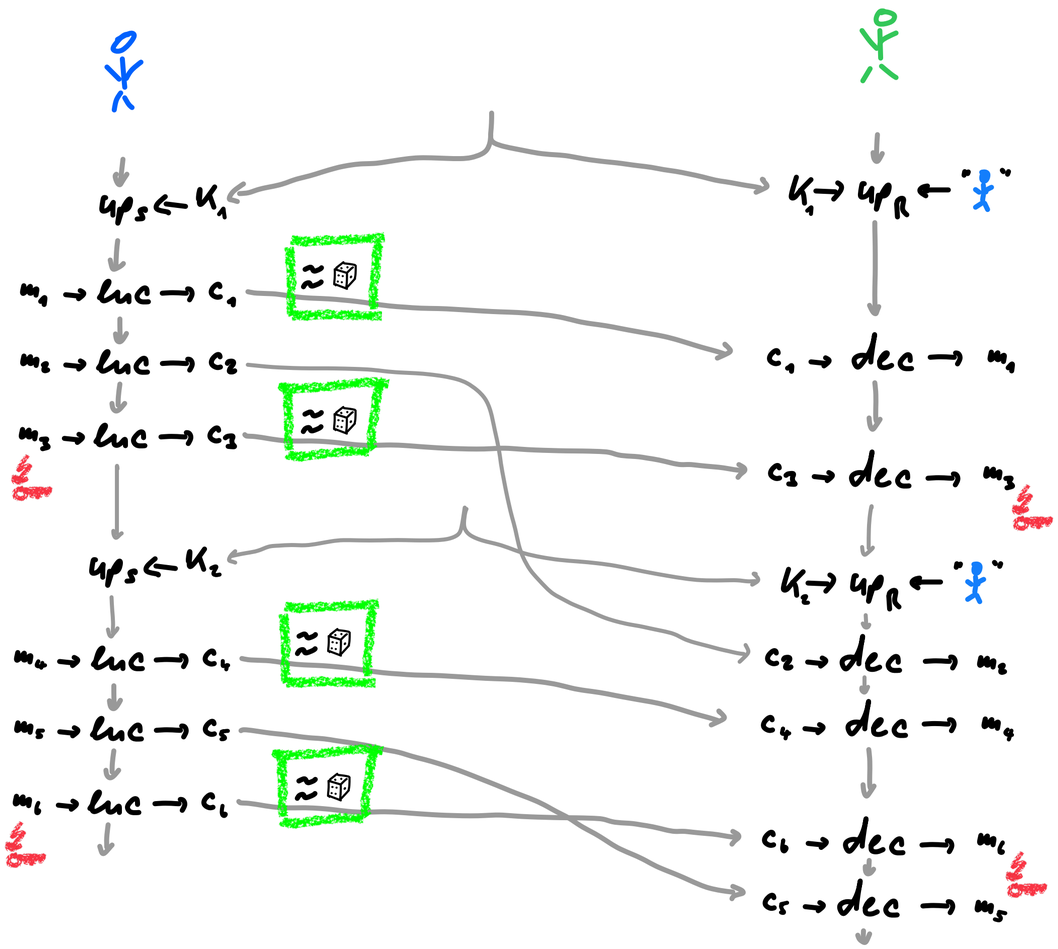
Anonymity: Message Anonymizer



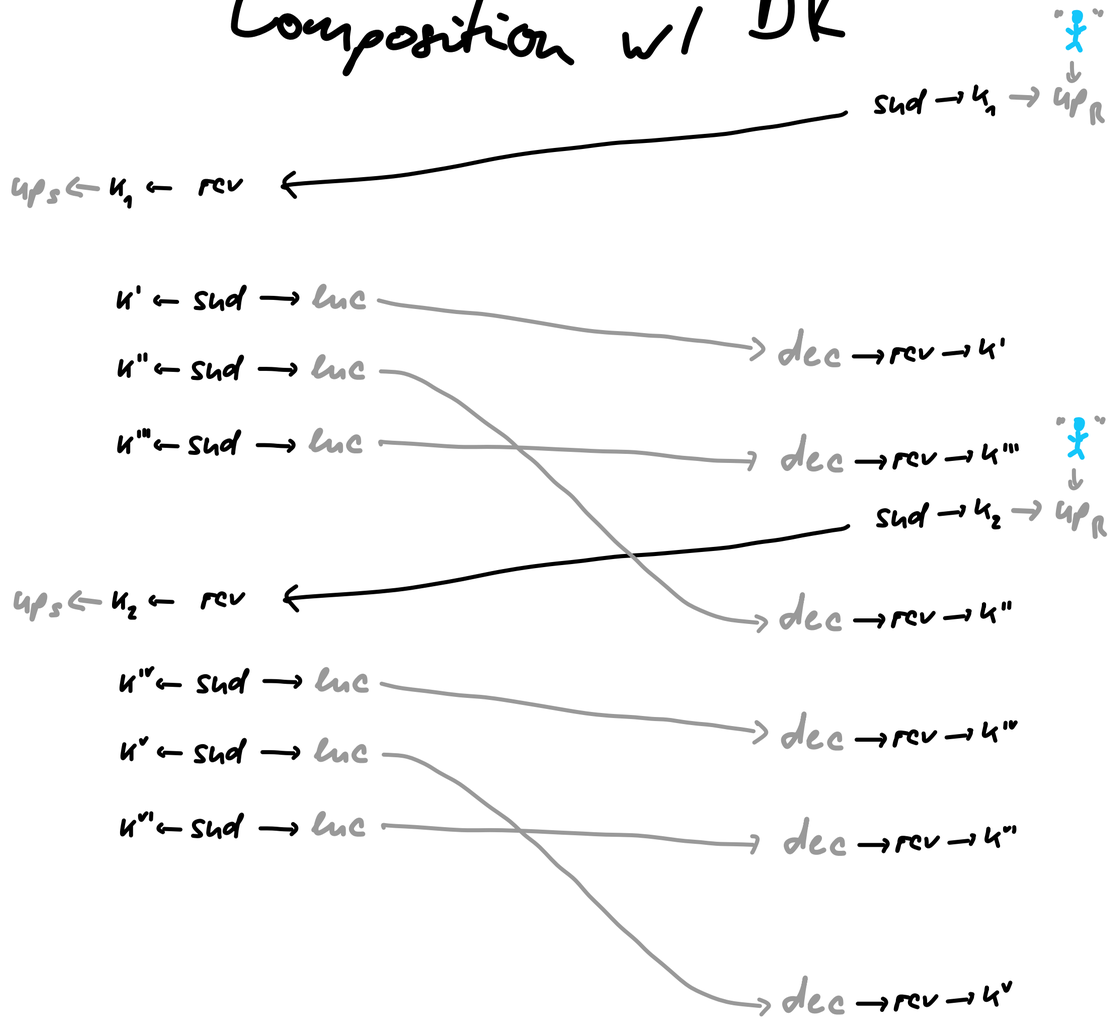
Composition w/ DR



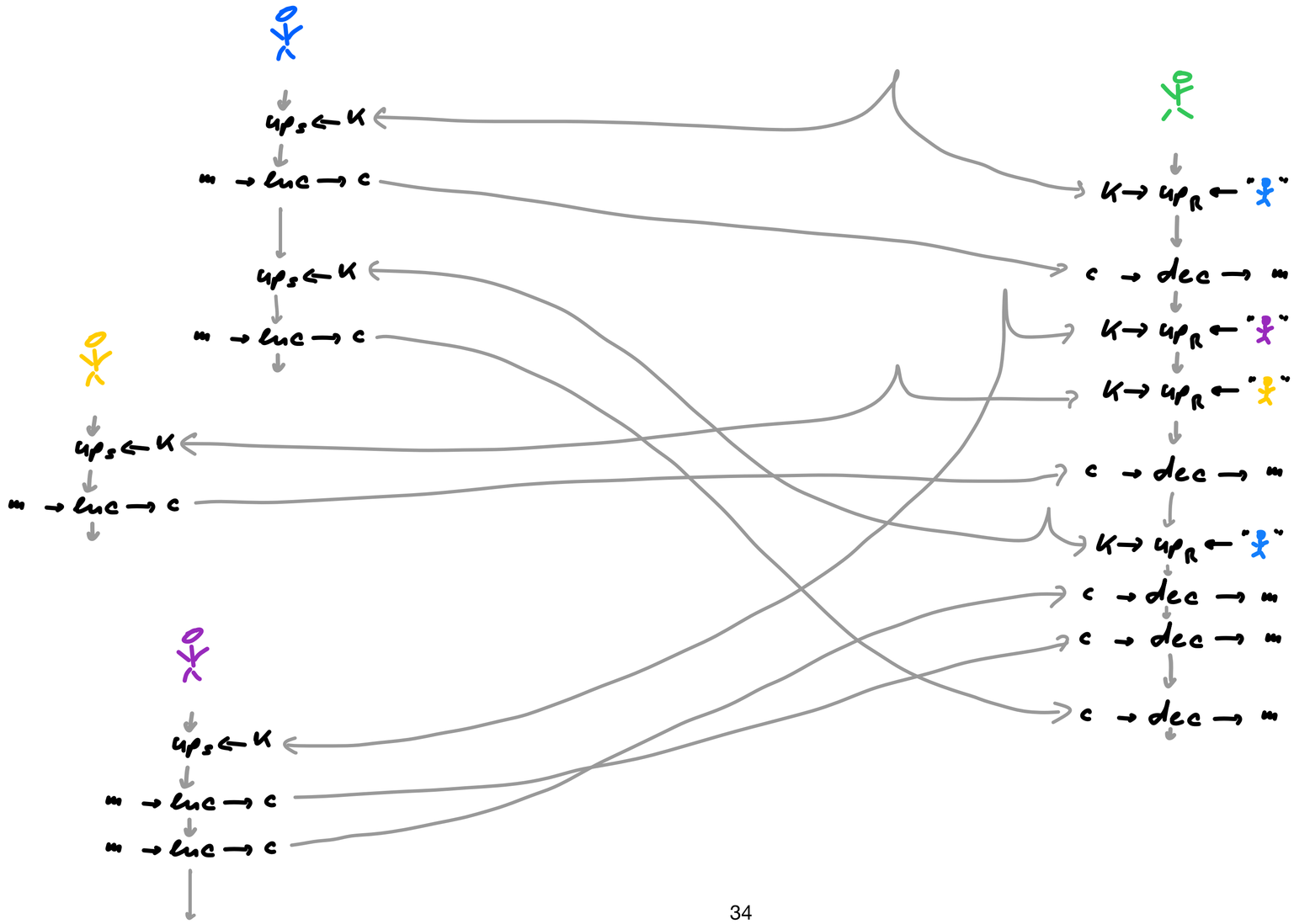
Anonymity: Message Anonymizer



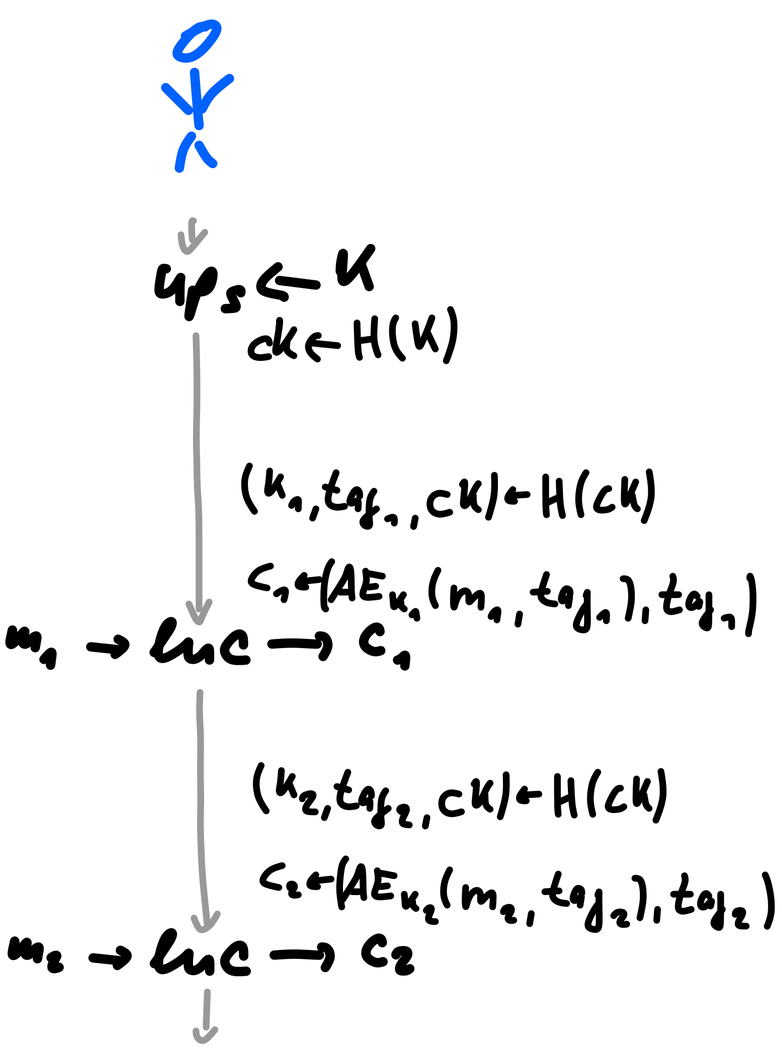
Composition w/ DR

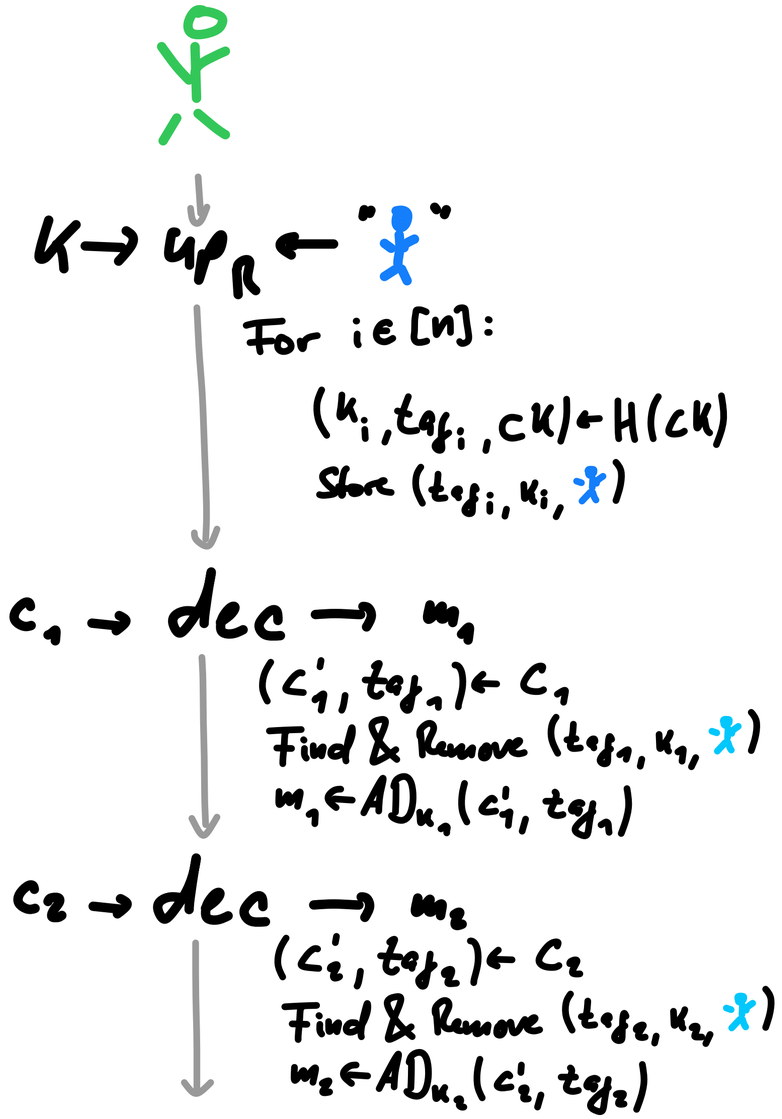


Anonymity: Message Anonymizer Multi Sender



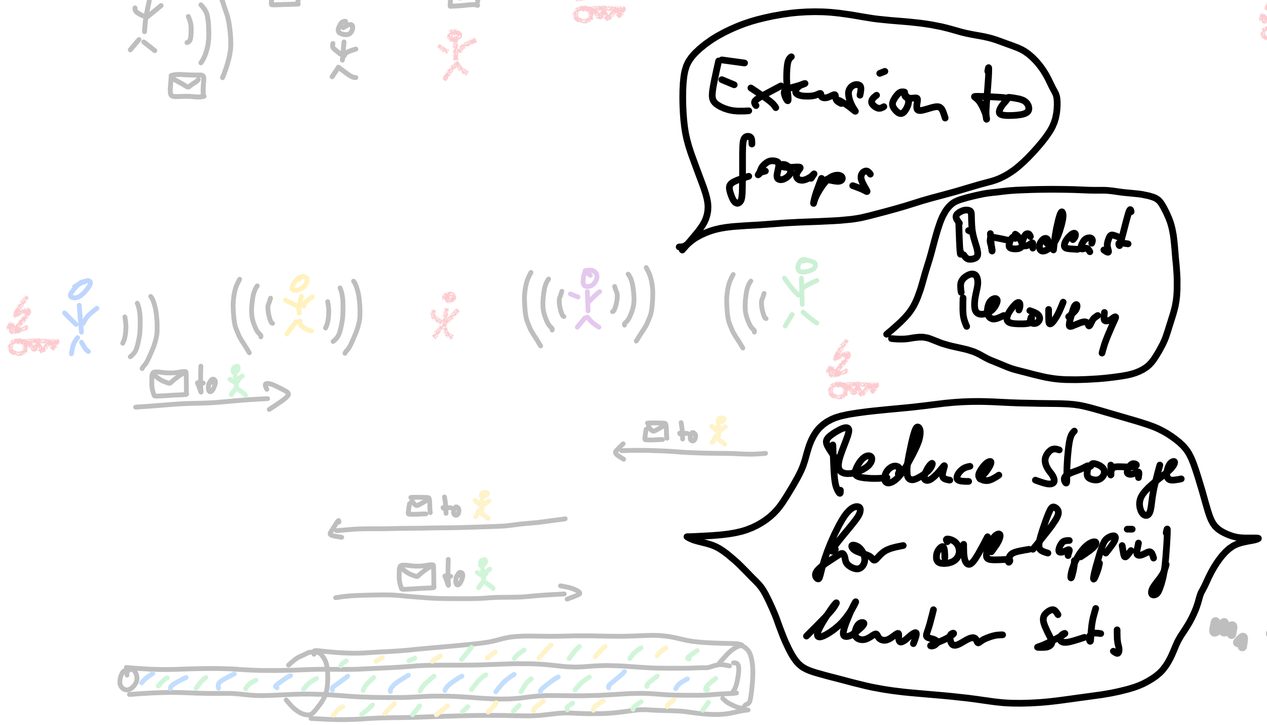
Anonymity: Message Anonymizer Construction

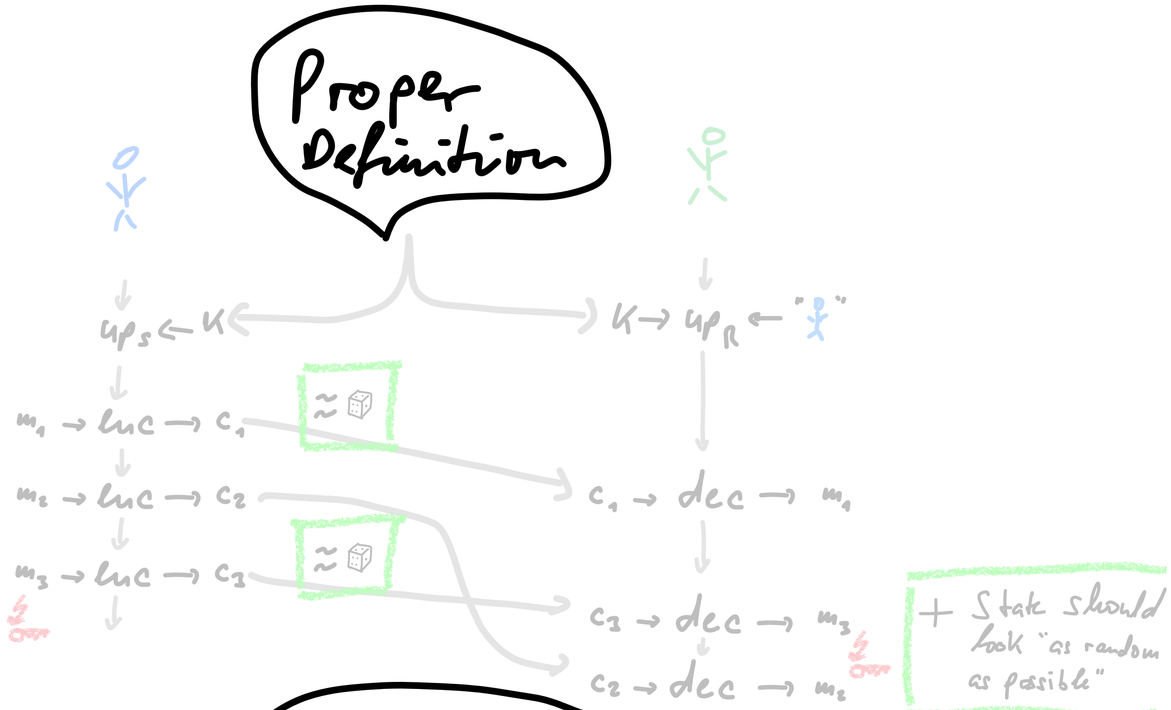


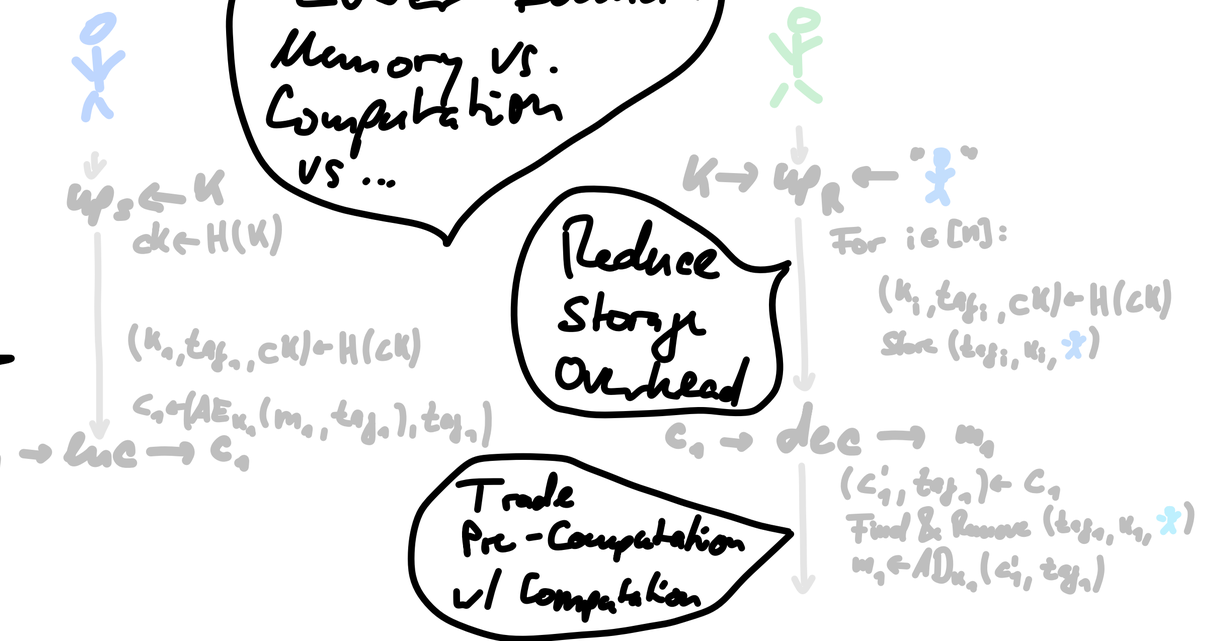


Summary


Censorship

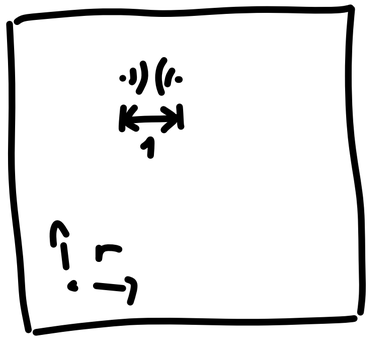

Extension to groups
Broadcast Recovery
Reduce storage for overlapping Membership Sets
to
to
to
to


Proper Definition
+ State should look "as random as possible"


Lower Bound: Memory vs. Computation vs...
Reduce Storage Overhead
Trade Pre-Computation w/ Computation

Simulation & Evaluation

$A \times A = 25^2$ Field



max Node Degree

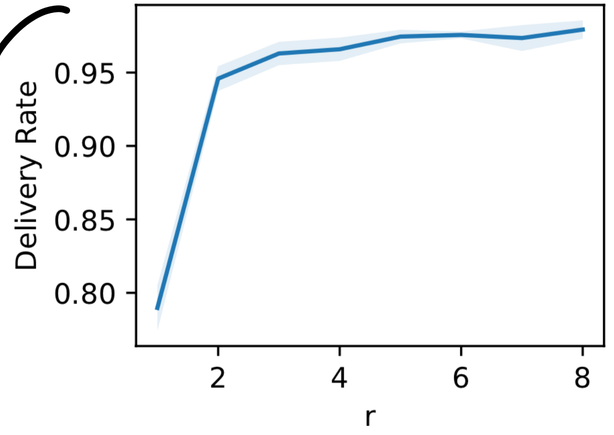
max Path Length

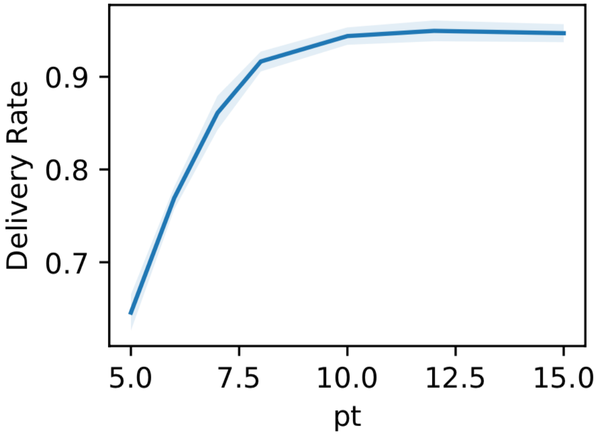
Distance/Step

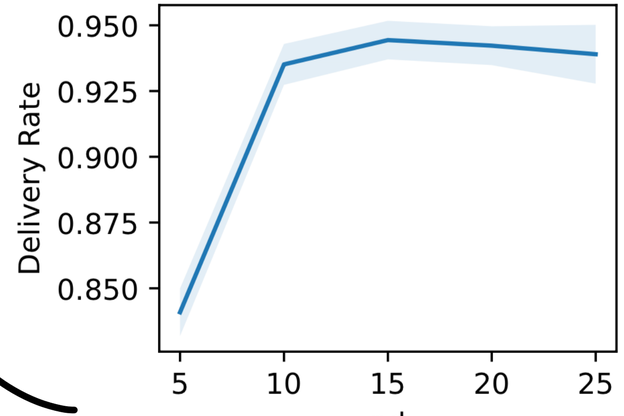
Users

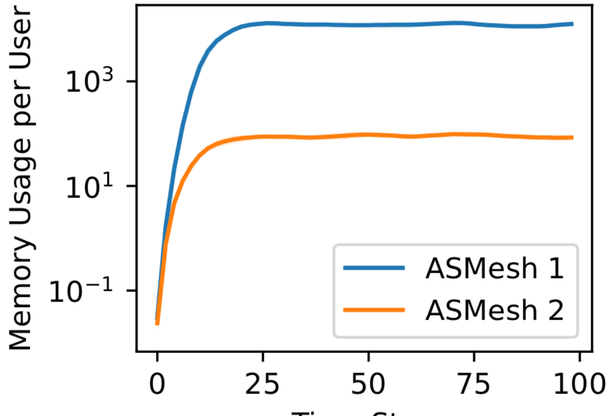
Profile	n	r	pt	nd	#Hops	Latency	#Re-enc
Standard	600	2	10	25	5.355	5.983	0.503
					5.396	6.043	n/a
Dense	3000	10	5	5	4.550	4.550	0.372
					4.545	4.545	n/a
Sparse	100	10	10	20	3.356	5.699	0.301
					3.367	5.490	n/a

Delivery Rate > 90%







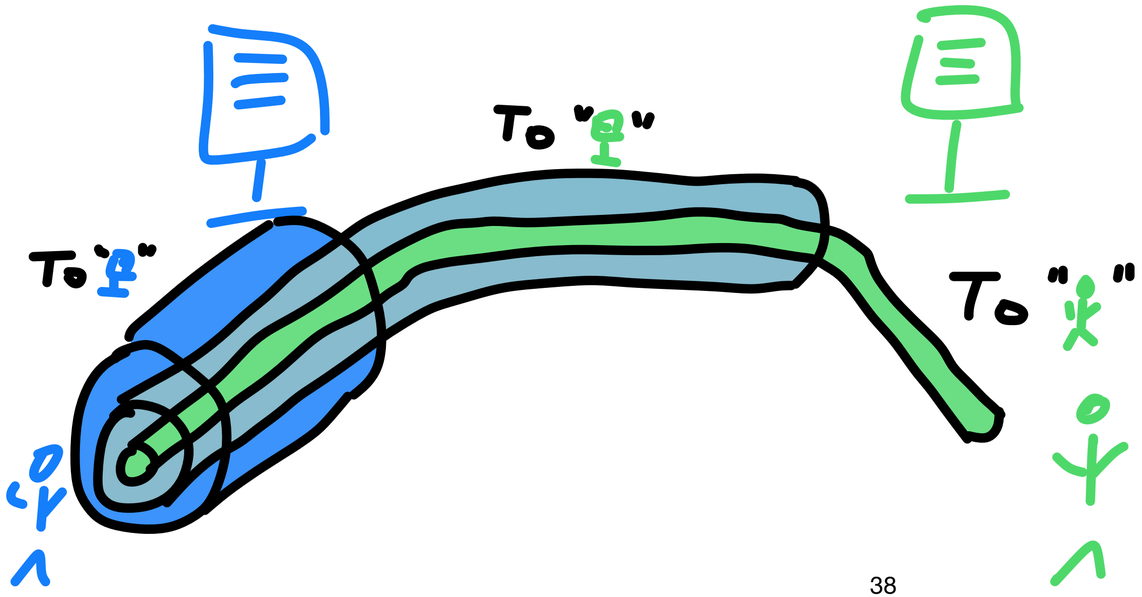


Anonymity : Broader Context

Hides Social Graph

↳ Used by Signal

↳ Relevant to Interoperable Messaging (GDPR + DMA)



Lower Bound

Simplest Syntax

$(sk_1, \ldots, sk_n, rk) \leftarrow gen$

$(sk, c) \leftarrow enc(sk, m)$

$(rk, m) \leftarrow dec(rk, c)$

1) if $|rk| = n \cdot m$, then 1

Simplest Execution Model

Snd(i, m, ch) -> c : c is random if ch=b=1

Rcv(c)

SCorrupt(i) -> sk_i // w/o: global symm. key

RCorrupt -> rk // w/o: global pub. key

↳ should "look random": hide # ctxts/sender

// w/o: puncturable symm. enc (tree)

// w/: if pre-computed symm. keys in rk