Introduction to network anonymity and mixnets

Claudia Diaz

KU Leuven

Outline

- Traffic analysis and network anonymity
- Mixes
- Mix networks, anonymous routing, dummy traffic
- Attacks on mixnets
- Take away points

What is Traffic Analysis

- Making use of (merely) the traffic data of a communication to extract information.
  - As opposed to ‘interception’ or ‘cryptanalysis’.

- What are *traffic data* or *network metadata*?
  - Identities or call signs of communicating parties.
  - Time, duration or length of transmissions.
  - Location of emitter or receiver.
  - No content – it may be encrypted.

“Just Metadata”

- Diffie & Landau – ‘Privacy on the line’:
  - *“Traffic analysis, not cryptanalysis, is the backbone of communications intelligence.”*
- NSA General Counsel Stewart Baker:
  - *“Metadata absolutely tells you everything about somebody’s life. If you have enough metadata, you don’t really need content.”*
- General Michael Hayden, former director of the NSA and the CIA:
  - *“We kill people based on metadata.”*

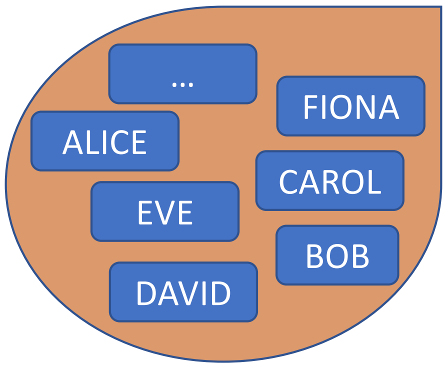
How easy is it to collect and exploit metadata?

- Exposed by default in core internet protocols:
  - TCP/IP, HTTP, UDP, FTP, TLS, DNS, …
- Available to a large number of intermediaries
  - Local LAN or WiFi router
  - Internet Service Provider (ISP), Mobile network operator
  - BGP routers, Autonomous Systems, Internet Exchanges
  - Internet backbone cables
- Metadata has lower legal protection than data content
- Metadata is machine-readable, lower volume than content and much easier to interpret automatically than content
- Metadata is difficult and expensive to protect

Anonymity

Anonymity definition (Pfitzmann and Hansen)

Anonymity is the state of being not identifiable within a set of subjects, the ***anonymity set***

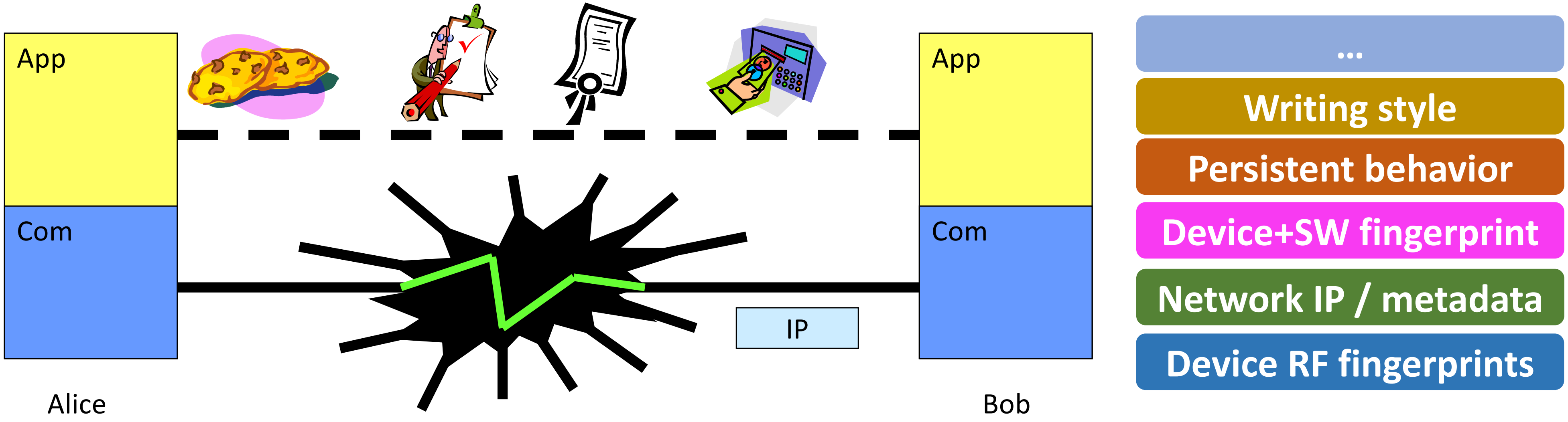


You CANNOT be anonymous on your own
You need a crowd of other (diverse) people

You are MORE anonymous when:

1. **The anonymity set contains more people**
2. **You do not stand out within that set**

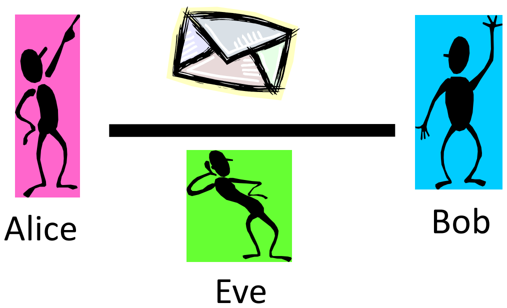
Note on Anonymity: Layers



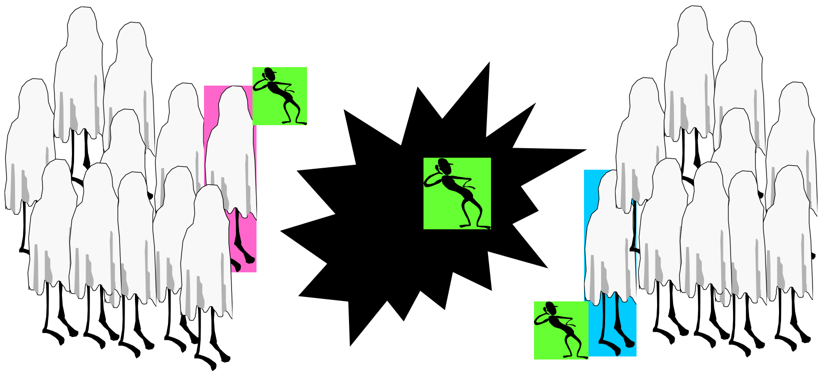
Leakage that enables deanonymisation can occur at multiple layers !

Anonymous communication model

Classical secure communication model



Anonymous communication model



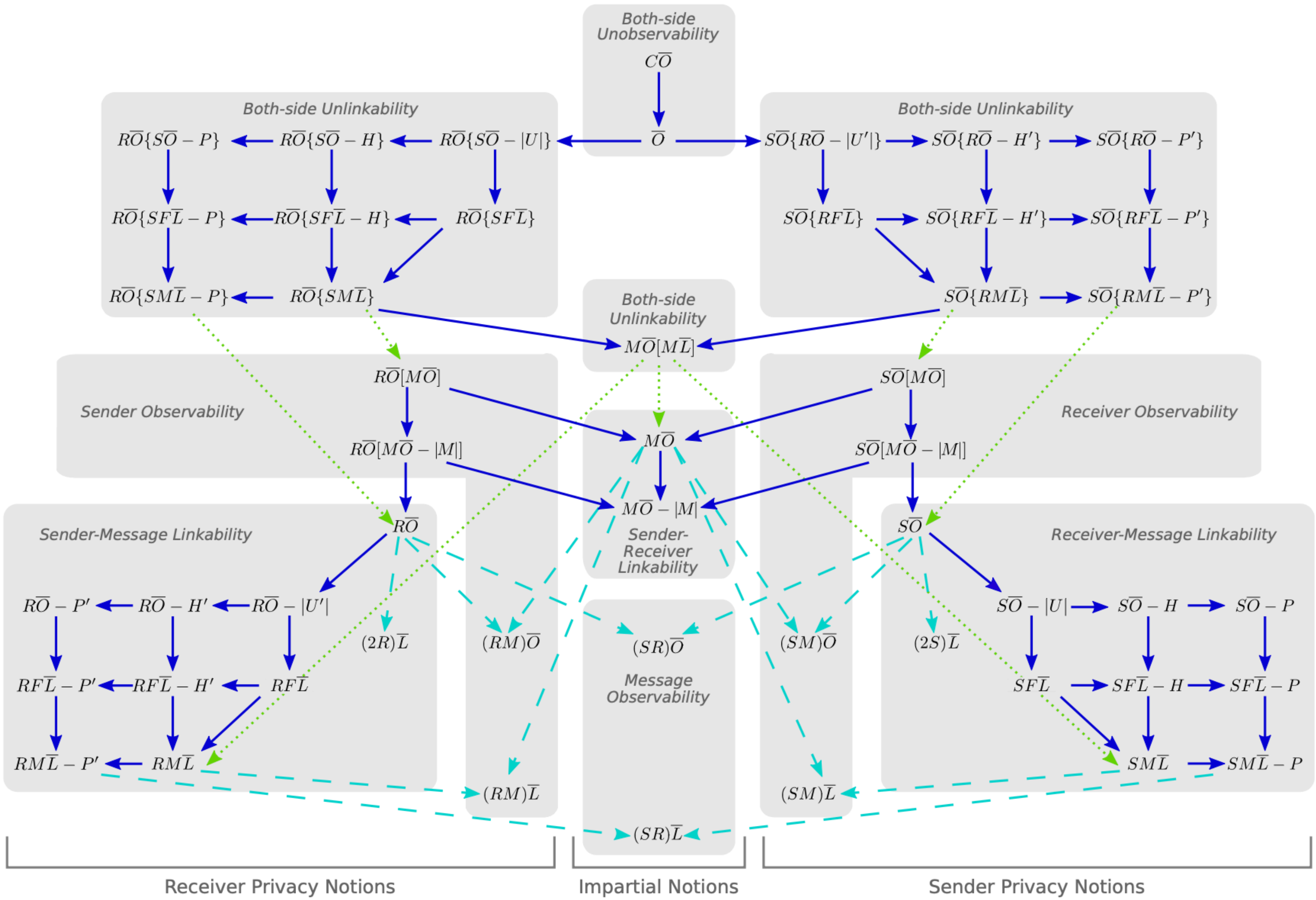
What would a “perfectly private” communication network offer?

- The possibility for Alice to communicate while preventing adversaries from learning:
  - **What** she is saying
  - **Who** she is communicating with (sending or receiving messages)
  - **When** she is communicating
  - **How long** she is communicating
  - **From where** she is communicating
  - The **amount of data** she is sending or receiving
  - Any **patterns** in her communications
  - **Whether** she is communicating at all

Privacy properties at the network layer

- **Confidentiality** of content
- **Anonymity**
  - Sender anonymity: receiver doesn't know who sent the message
  - Receiver anonymity: entity can be reached, or replied to, anonymously
  - Anonymity towards third parties: sender and receiver identify each other, but no other party can tell they are communicating with each other
- **Unlinkability**: impossible to determine that 2 (or more) messages, actions or pieces of data relate to the same user
- **Unobservability**: concealing the timing and volume of communications
- **Undetectability**: concealing participation in the network

- **Distribution of trust**: avoid central points of failure, resilience to partial compromise
- **Forward security**: limit the impact of participant compromise



C. Kuhn, M. Beck, S. Schiffner, E. Jorswieck, Thorsten Strufe. "On Privacy Notions in Anonymous Communication". PoPETs 2019

Powerful network adversaries

- **Capabilities**
  - Can monitor all links in the network
  - Can compromise entities in the network by injecting corrupt nodes (Sybil attack) or through coercion (importance of forward security and deniability)
  - Active adversary: can read, inject, delete, modify messages

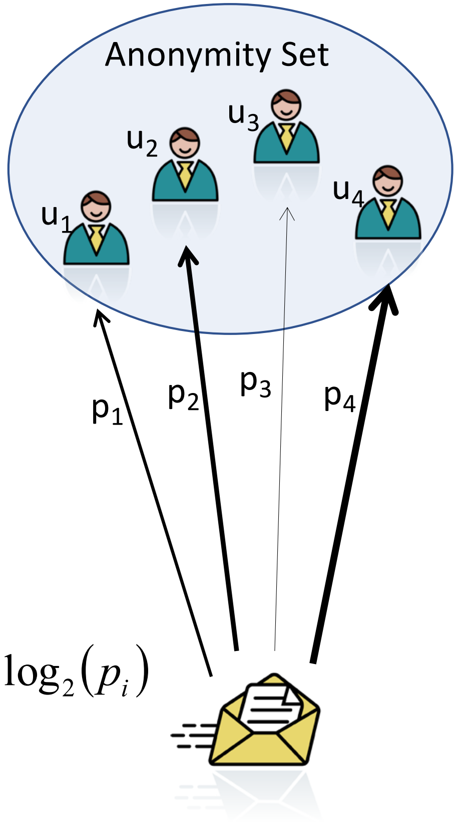
- **Main objective**: determine who communicates with whom

- **Limitations**: cannot break crypto primitives or see inside nodes it does not control

- Attack **method**: analysis of metadata

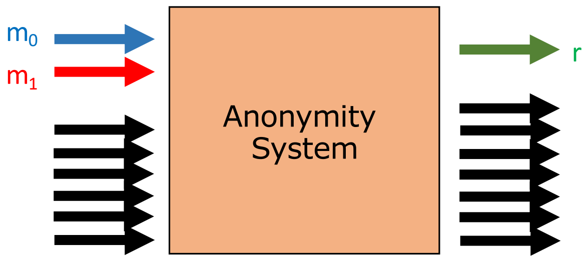
Anonymity metrics: evaluate adversarial success

- Approaches:
  - Possibilistic metrics
  - Probabilistic / entropy metrics
    - Capture scalability
  - Indistinguishability / differential privacy metrics
    - Capture how close to perfect



p_0 = Prob (m_0 =r)
p_1 = Prob (m_1 =r)

$$\varepsilon = |\ln(p_0/p_1)|$$

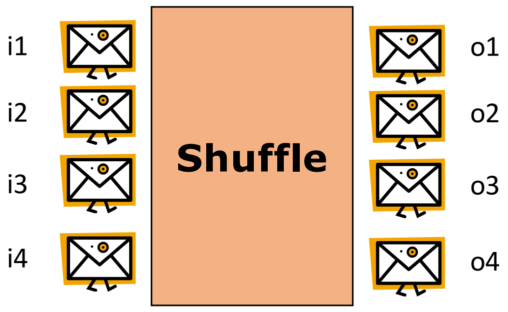


$$H = -\sum_{i=1}^{N} p_i \cdot \log_2(p_i)$$

Mixes

Chaumian mix

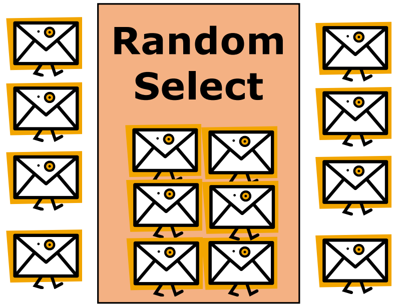
- Mix: Proxy for anonymous email
- Goal: an adversary observing the input and output of the mix is not able to relate input messages to output messages
  - Bitwise unlinkability
    - The mix performs a crypto operation on input messages
    - Input/output of the mix cannot be correlated based on content or size
  - Prevent traffic analysis based on message I/O order and timing
    - Achieved by batching and shuffling messages



- Several mixes can be chained to distribute trust:
  - Sender → Mix_1 : $\{Mix_2, \{Rec, msg\}_{K_{Mix_2}}\}_{K_{Mix_1}}$

Other mix designs based on batching

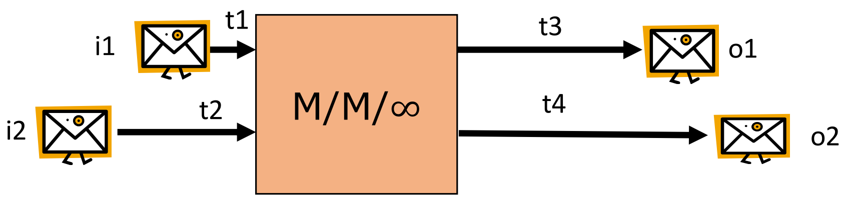
- Timed mixes:
  - Flush periodically, every T time units, regardless of how many messages have arrived
  - Optional flushing conditions: flush only if a minimum number of messages has been received

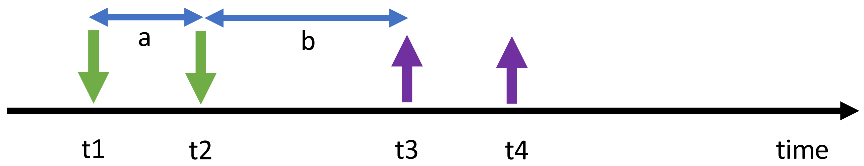


- Pool mixes (Mixmaster):
  - Flush only a subset of (randomly selected) messages and keep the rest for the next round, to be mixed with new arrivals
  - Long-tail anonymity sets
  - Increased variance of latency

Continuous-time mixes

- Stop-And-Go / Poisson mixes:
  - Delay each message individually with the amount of time drawn from an exponential distribution
  - Anonymity similar to a pool mix because of the memoryless property of exponential distributions
  - Delays picked by the sender: can predict delivery time





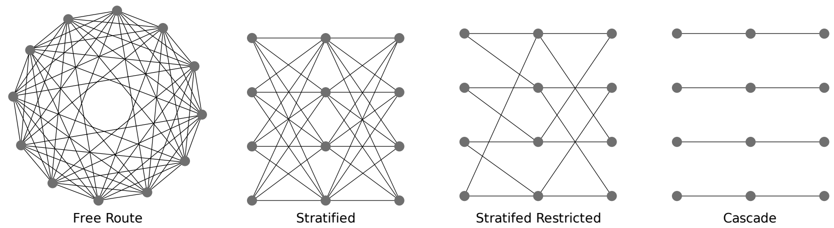
For an exponential random variable X it holds that:

Pr[X > a+b | X > a] = Pr[X > b]

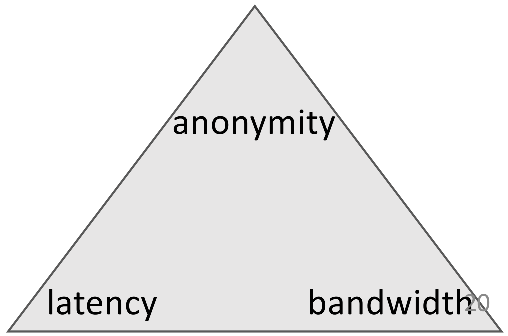
Mix networks and anonymous routing

Mix networks

- Distribute trust to avoid single points of failure:
  - Route messages through multiple mixes to provide anonymity even if some mixes are compromised



- Network topology?
- Who selects routes?
- Latency / Anonymity / Bandwidth tradeoffs?



Anonymous routing

Feature Name			Description	Instantiation and Symbols
Network Structure	Network topology		Degree of node connectivity in the network	⊠ (fully) □ (mostly) ⊏ (partially)
	Connection type	Direction	Data flow in connections	→ (unidirectional) ↔ (bidirectional)
		Synchronization	Timing model for connection establishment and data sending	≠ (asynchronous) ≅ (synchronous)
	Symmetry	Roles	Users operating as relays	•·•·• (peer-to-peer) •·• (client-server) •·⊙·• (hybrid)
		Topology	Node topology for routing	··· (flat) ✤ (hierarchical)
		Decentralization	Degree of decentralization for non-routing services	⊙ (semi decentralized) ○ (fully decentralized)
Routing Info	Network view		Network view necessary for making routing decisions	● (complete) ◐ (partial)
	Updating		Triggers for routing information updates	◷ (periodic) ϟ (event-based)
Communication Model	Routing type		Node selection per route	•·· (source-routed) ··•·· (hop-by-hop)
	Scheduling		Prioritization of traffic	≡ (fair) ◇ (prioritized)
	Node selection	Determinism	Determinism of node selection	✓ (deterministic) ✗ (non-deterministic)
		Selection set	Permissible set of nodes per route	Ⓐ (all) ⬢ (restricted, security) 🌍 (restricted, network) ☺ (user-based)
		Selection probability	Node selection probability per route	⊛ (uniform) ◎ (weighted, static) ❋ (weighted, dynamic)
Performance, Deployability	Latency		Protocol latency	L (low-latency) H (high-latency) M (mid-latency)
	Communication mode		Longevity of connections	•–• (connection-based) ✉ (message-based)
	Implementation		Implemented	✓ (yes) ✗ (no)
	Code availability		Open source	✓ (yes) ✗ (no)

How are mixnets similar/different from Tor?

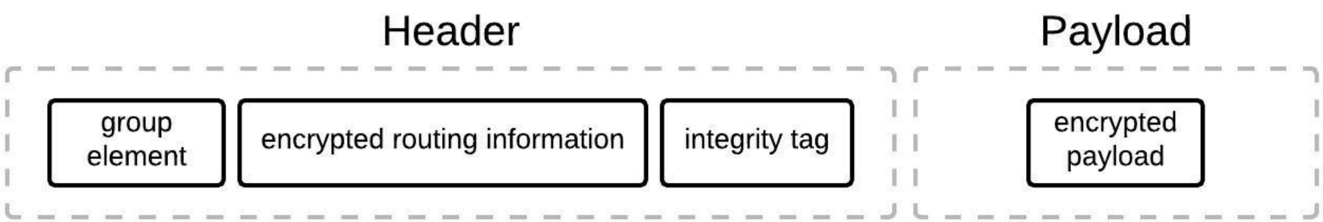
- Similar
  - **Source routed** with nested encryption (though voting mixnets use cascades and re-randomizable crypto)
  - Packets traverse an **overlay network** with **multiple hops**

- Different:
  - Tor is **connection-based** vs Mixnets that are **packet-based** (routing info in each packet)
  - Tor does not add **latency** vs latency added in Mixnets
    - Vulnerable to end-to-end confirmation vs (possibly) vulnerable to long-term intersection attacks
    - Designed to resist local adversaries vs global adversaries

- Additionally (possible in both systems):
  - **Dummy traffic** strategies to strengthen anonymity and enable unobservability

Sphinx packet format

- Compact and secure packet format for nested encryption
- Like Onion Routing, each mix in the path “peels off” a layer
- Unlike Onion Routing, there is no interactive circuit/session establishment with shared ephemeral keys
  - Keys must be derived from the packet itself: combination of group element and private key of the mix
- Per-hop bitwise unlinkability
- Tagging attack detection
- Replay attack detection



Single Use Reply Blocks (SURBs)

- Sphinx headers that route back to the original sender
  - Can only be used once → prevent replay attacks
- Uses:
  - Indistinguishable replies
  - Reliable transport (ACKs)
  - Can function similarly to “onion addresses”
- Practical challenges
  - Limited validity (tradeoff with forward security)
  - Inefficient if downstream traffic much larger than upstream

Dummy traffic

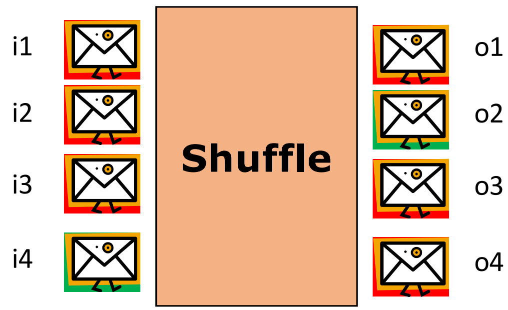
- Fake messages introduced to confuse the attacker
- Indistinguishable from real traffic
- Increase anonymity and enable unobservability

- Dummy traffic design
  - Generated by users and/or by mixes?
  - Destination? (self, mix or other user)
  - Frequency of generation? Deterministic or random? Dependent or independent of real traffic?
  - Higher order correlations? (e.g., replies to simulate “conversations”)
  - …

Two attacks on mixnets

Blending (or "N-1") attacks

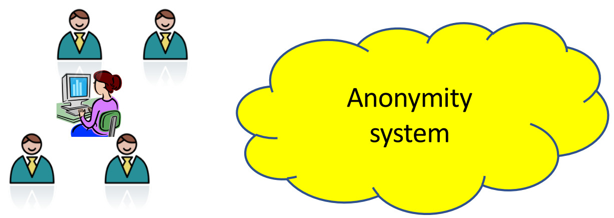
- Attack steps
  1. Empty the mix of legitimate messages
  2. Let the target message into the mix
  3. Fill the mix with attacker-generated messages, while preventing other legitimate messages from entering the mix
  4. At the output the adversary recognizes his own messages. The unknown message is the target

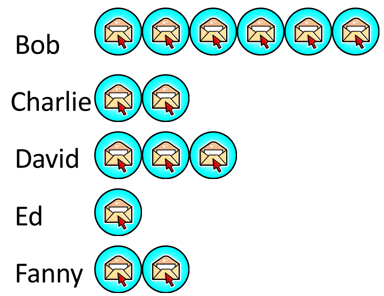


- Very simple attack for Chaumian mixes, more sophisticated variants also affect other types of mixes
- Attack is detectable with loops of dummy traffic

Long-term intersection attacks

- Assumptions:
  - Alice has persistent communication relationships (she communicates repeatedly with her friends)
  - There is a large population of senders and a different subset sends their messages with Alice’s in each round
- Method:
  - Combine many observations (looking at who receives when Alice sends)
- Intuition:
  - If we observe rounds in which Alice sends, her likely recipients will appear frequently
- Result:
  - We can create a vector that expresses Alice’s sending profile





Notes on long-term intersection attacks

- Hard to conceal persistent communications
  - *Any* practical anonymous communication channel will reveal long-term relationships
- The larger the ratio between user base and the mix threshold, the better the attack works
- Unobservability (dummy traffic) might help
  - BUT: expensive, and online/offline status may be hard to conceal
- Long-Term intersection Attacks take time:
  - Anonymity may be tactical
  - Evolution of user communication patterns over time

Take away points

- Anonymity needs to be protected at all layers: it is fragile
- You can’t be anonymous on your own: a crowd to blend in is needed
- Anonymous routing requires taking many features and tradeoffs into consideration
- Dummy traffic is needed for unobservability
- Mixnets are an alternative to onion routing that
  - are packet-based and higher-latency
  - can provide stronger anonymity towards global network adversaries