OP5: Mixnet anonymity vs performance

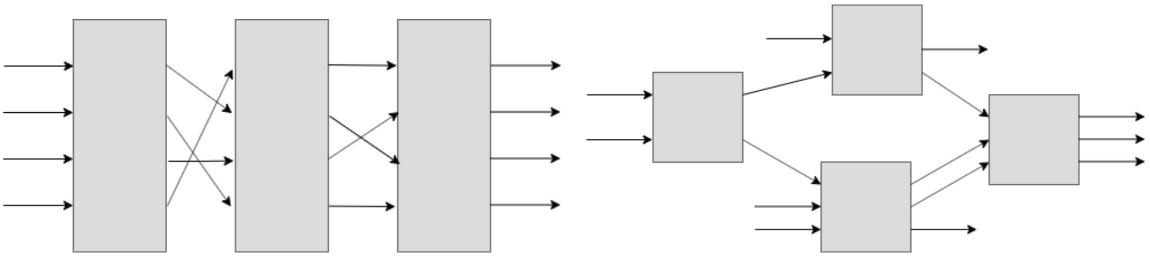
When designing systems using Mixnets for anonymity [1], the choice of Mixnet type mainly depends on the number of users, the message distribution, the required anonymity level and time performance and the need for decentralisation. The discussed threat model is a Global Passive Adversary (GPA) performing traffic analysis attacks to link encrypted messages going out of a Mixnet with those going in. We will use *k-anonymity* to quantify anonymity. A set of messages is considered *k-anonymous* if one cannot distinguish an encrypted message from at least $k-1$ other messages.

The main trade-off in Mixnets design, is anonymity against performance. A high level of anonymity in a Mixnet usually implies a high latency. We assume exchanged packets are padded and split into equal sized messages, and a constant number of messages per second m. To provide anonymity, Mixnet nodes wait for k messages to arrive before forwarding them by batch, thus providing *k-anonymity*. Latency is defined as the maximum time a message has to wait at a node before the batch is sent, represented as $t = \frac{k}{m}$ seconds.

Starting with a small number of users u_1, the optimized Mixnet for both anonymity and performance is a single centralised Cascade Mixnet (1a). The centralised Cascade Mixnet has exactly one entry and exit node. It can be composed of a single node, or of a chain of non-colluding nodes shuffling messages before passing them to each other, users only need to trust one node in the chain. As all messages follow the same path in the Mixnet, it is very difficult for a GPA to link ingoing and outgoing messages. Having a single Mixnet for a large network may induce an increased latency delay for remote clients. u_1 is defined as the number of users that constantly generate $m_1 = \frac{k_1}{t_1}$ messages per second, for chosen k_1*-anonymity* and t_1 latency. Latency and anonymity are directly linked.

There are multiple possible improvements with a larger number of users $u_2 > u_1$, generating $m_2 > m_1$ messages per second. One can keep a centralised Cascade Mixnet, improving the anonymity level while keeping the same latency t_1 by taking $k_2 = m_2 \times t_1 > k_1$, or decreasing latency for the same anonymity level k_1 such that $t_2 = \frac{k_1}{m_2} < t_1$. Otherwise, one can use a Free Route Mixnet (1b) having multiple entrance and exit nodes, providing the same anonymity level k_1 and latency t_1 as long as there are still k_1 messages (or none) at each node of the Mixnet at any step. It is also possible to make trade-offs between anonymity, latency and decentralisation improvements, depending on the needs of the system.

Different systems have different Mixnets needs. Some e-voting protocols use Mixnets to shuffle votes, making it impossible to link a vote to a person. They can be centralised, need a strong anonymity guarantee and are not time critical. Tor [2] aims to offer a low-latency, decentralised network to its users, at the price of a lower privacy level compared with e-voting systems. There is no absolute best mix between anonymity, performance and decentralisation, but mixes adapted to each system.



(a) A Cascade Mixnet

(b) A Free Route Mixnet

Figure 1: Examples of Mixnets

References

[1] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, February 1981. https://www.chaum.com/publications/chaum-mix.pdf.

[2] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," *Paul Syverson*, vol. 13, June 2004. https://svn-archive.torproject.org/svn/projects/design-paper/tor-design.pdf.