El Gamal Mix-Nets and Implementation of a Verifier

Erik Larsson and Carl Svensson

Introduction

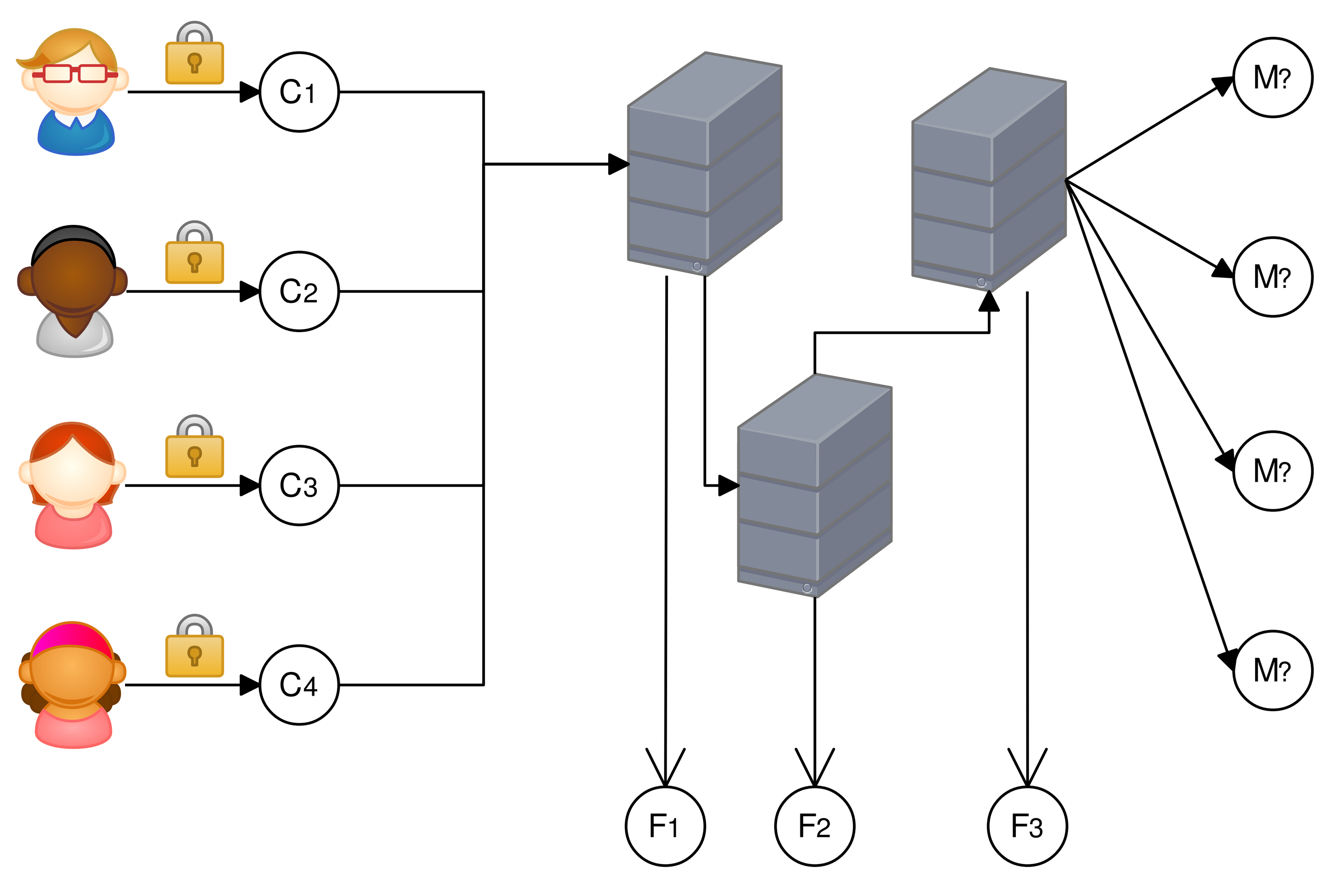
blah

Mix Networks

One purpose of mix networks is to provide untraceability to its users. A mix-net takes as input a list of encrypted messages. Verificatum is a reencryption mix-net. Such a mix-net consists of a number of servers, mix servers, which sequentially process the messages and reencrypts the list of messages and outputs them in a randomized order. After passing through all servers, the list of ciphertexts is decrypted and the result is a list of the messages in random order.

In the context of electroinic voting a reencryption mix-nets may work as follows.

1. The mix servers prepare the mix-net by generating public and secret keys.
2. Each voter encrypts his vote and appends it to a public list of encrypted votes.
3. In sequential order each mix server takes as input the list of encrypted votes, reencrypts and outputs them in a randomized order, replacing the previous list of encrypted votes.
4. After all mix servers have processed the list, each vote is jointly decrypted and posted on a bulletin board making the outcome of the election universally available without revealing how anyone voted.

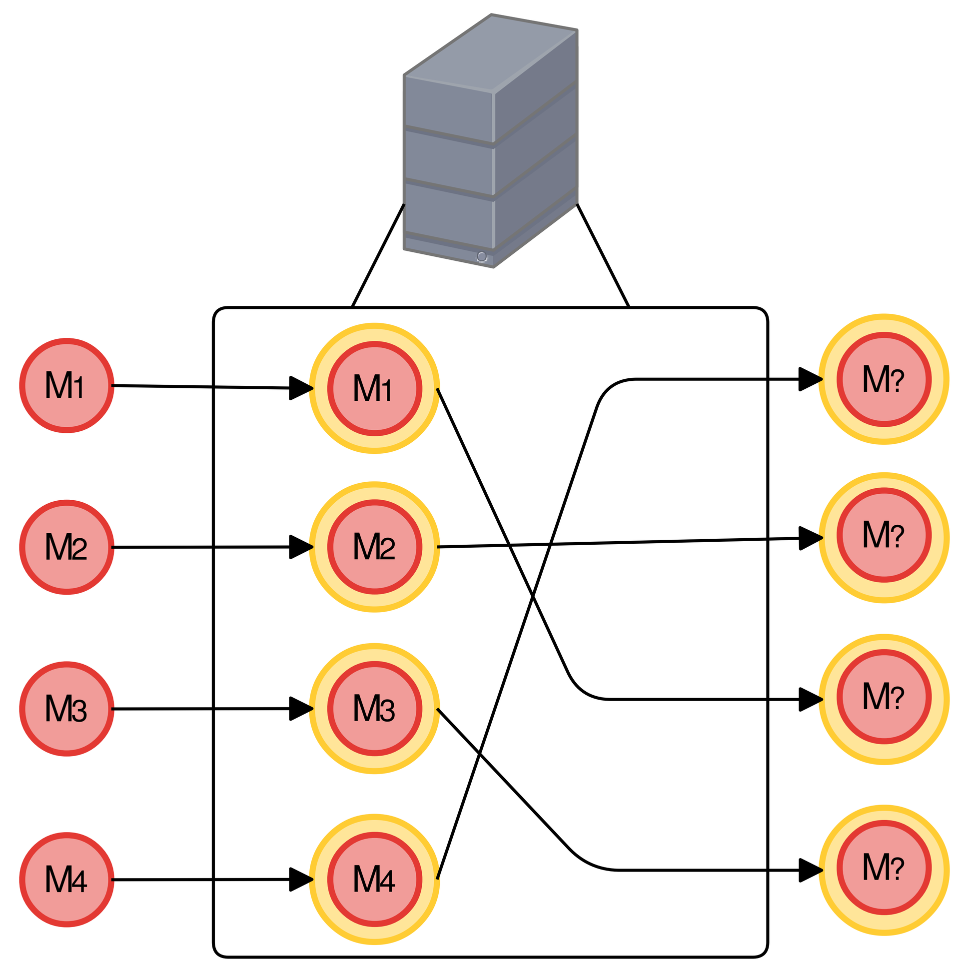


Mix Server

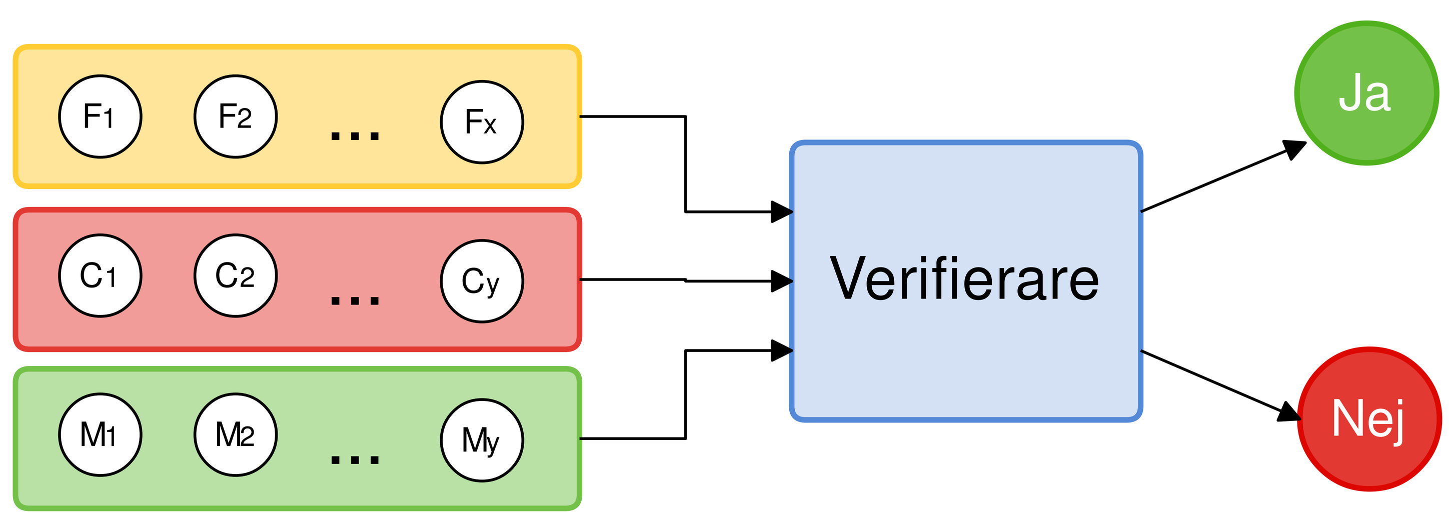
Each server in the mix-net perform the same actions during execution.

1. It is given the list of ciphertexts from the previous server.
2. It reencrypts these ciphertexts with its private key.
3. It randomly shuffles the list of ciphertexts.
4. It passes the reencrypted and shuffled list of ciphertexts to the next server.

During these steps a zero-knowledge proof is produced which can prove that the server indeed did output a list of shuffled and reencrypted ciphertexts without tampering with the ciphertexts.



Verifier



El Gamal Cryptography

The El Gamal cryptosystem is a public key cryptosystem based on the computational difficulty of computing discrete logarithms in cyclic groups. The cryptosystem is defined over a group G_q of order q with generator g. The secret key x is chosen randomly in $\mathbb{Z}_q$ and the public key y is created as follows

$$y = g^x$$

Encryption of a plaintext $m \in G_q$ is done by choosing a random $s \in \mathbb{Z}_q$ and computing

$$\mathrm{Enc}(m, s) = (u, v) = (g^s, y^s m) \in G_q \times G_q$$

Decryption of a ciphertext $(u, v) \in G_q \times G_q$ is achieved by using the private key x to compute

$$\mathrm{Dec}(u, v) = u^{-x} v = (g^s)^{-x} y^s m = (g^x)^{-s} y^s m = y^{-s} y^s m = m$$

The El Gamal Cryptosystem possesses a homomorphic property. This means that for two messages $m_1, m_2 \in G_q$ and random numbers $s_1, s_2 \in \mathbb{Z}_q$

$$\mathrm{Enc}(m_1, s_1) \cdot \mathrm{Enc}(m_2, s_2) = \mathrm{Enc}(m_1 m_2, s_1 + s_2)$$

By choosing one of the messages to be the identity element, $m_2 = 1$, and letting the other one being any message $m_1 = m$, one has obtained the ability to reencrypt a particular ciphertext without knowing the original plaintext nor the randomness.

$$\mathrm{Enc}(m, s_1) \cdot \mathrm{Enc}(1, s_2) = \mathrm{Enc}(m_1, s_1 + s_2)$$

This property of a cryptosystem is necessary if it should be used in a reencryption mix-net.

Conclusion

blah blah blah blah