MACI anonymization requires ElGamal cryptographic functions in zero knowledge. Support for anonymization in MACI will come sometime in the future, but I took a stab at implementing its required building blocks:

GitHub

weijiekoh/elgamal-babyjub

Contribute to weijiekoh/elgamal-babyjub development by creating an account on GitHub.

This library implements ElGamal encryption, decryption, and rerandomization in Typescript for the BabyJub elliptic curve. It also provides decryption and rerandomization circuits written in circom.

A quick note about how it encodes plaintext. We define a plaintext value as a BigInt in the BabyJub finite field. To encrypt it, we need to first convert it into a safe BabyJub elliptic curve point. Instead of using a map-to-curve function, the encodeToMessage

function generates a random value r

, computes g ^ {r}

, and outputs both g ^ {r}

and the x-increment e

that must be added to the plaintext to obtain the x-value of g ^ {r}

.

The ciphertext is therefore two elliptic curve points and one field element: (c_1, c_2, e)

.

After decrypting (c_1, c_2)

to obtain the elliptic curve point m

, we convert it to the original plaintext by computing m_x - e

where m_x

is the x-coordinate of m

.

I'd love any feedback and suggestions on how to improve it. Thanks to@kobigurk, @snjax, and others in the iden3 Telegram group for their help.