Rust-miximus

I built an MVP version of Miximus in Rust using bellman and tools from Matter Labs. Happy to share it with the community as a means of providing more implementations of applications and circuits in Rust.

In particular this exposes WASM functions that can be used to interact with the zkSNARK though I have not written this part of the application and will most likely will not. Happy to discuss for anyone interested though! This was largely inspired by work done by Kobi Gurk at Qedit in this repo and of course @barryWhiteHat.

One thing that is particularly questionable that I am unsure of is using this utility to do proofs of membership in an incremental merkle tree. Since generating the zkSNARK proof parameters requires knowing the depth of the merkle tree/length of the authentication proof, when this updates is it any risk to continuously regenerate the parameters over new depths? Assuming we have access to tamperproof randomness, I presume not but I'm curious to hear thoughts if anyone has some to share. It certainly makes the case for using sparse merkle trees so that parameter generation only needs to happen once, over a depth 32 SMT.