Hello,

I've been struggling to find a justification for the size of the private key in BLS12-381

. So far, my best hit is within thischia network document, who gives some insight on the public key and signature, but just mentions what is going on with the private key.

private key (32 bytes):

Big endian integer.

pubkey (48 bytes):

381 bit affine x coordinate, encoded into 48 big-endian bytes. Since we have 3 bits left over in the beginning, the first bit is set to 1 iff y coordinate is the lexicographically largest of the two valid ys. The public key fingerprint is the first 4 bytes of hash256(serialize(pubkey)).

signature (96 bytes):

Two 381 bit integers (affine x coordinate), encoded into two 48 big-endian byte arrays. Since we have 3 bits left over in the beginning, the first bit is set to 1 iff the y coordinate is the lexicographically largest of the two valid ys. (The term with the i is compared first, i.e 3i + 1 > 2i + 7). The second bit is set to 1 iff the signature was generated using the prepend method, and should be verified using the prepend method.

Are these 32 bytes related to the parameter r

defined here? Or is just an un-educated guess?

As is common, we target a subfamily of these curves that has optimal extension field towers and simple twisting isomorphisms. In order to ensure Montgomery reductions and other approximation algorithms are space-efficient, we target r≈2^255

so that the most significant bit of rr (and qq) are unset with 64-bit limbs.

Any pointer will be appreciated.

Herman