Contents

this is an effort to break down the building blocks of crypsinous blockchain

1 Crypsinous blockchain

Crypsinous Blockchain is built on top of Zerocash sapling scheme, and Ouroboros Genesis blockchain. Each part U_p stores it's own local view of the Blockchain $C_{loc}^{U_p}$. C_{loc} is a sequence of blocks B_i (i>0), where each $B \in C_{loc}$

$$B = (tx_{lead}, st)$$

$$tx_{lead} = (LEAD, st\overrightarrow{x}_{ref}, stx_{proof})$$

$st\overrightarrow{x}_{ref}$ it's a vector of tx_{lead} that aren't yet in C_{loc}. $stx_{proof} = (cm_{\prime c}, sn_c, ep, sl, \rho, h, ptr, \pi)$ the Blocks' st is the block data, and h is the hash of that data. the commitment of the newly created coin is: $(cm_{c_2}, r_{c_2}) = COMM(pk^{COIN}||\tau||v_c||\rho_{c_2})$, τ is the clock current time. sn_c is the coin's serial number revealed to spend the coin.

$$sn_c = PRF^{sn}_{root_{sk}^{COIN}}(\rho_c)$$

$$\rho = \eta^{sk_{sl}^{COIN}}$$

η is is from random oracle evaluated at $(Nonce||\eta_{ep}||sl)$, ρ is the following epoch's seed. ptr is the hash of the previous block, π is the NIZK proof of the LEAD statement.

1.1 st transactions

the blockchain view is a chain of blocks, each block $B_j = (tx_{lead}, st)$, while st being the merkle tree structure of the validated transactions received through the network, that include transfer, and public transactions.

1.2 LEAD statement

for $x = (cm_{c_2}, sn_{c_1}, \eta, sl, \rho, h, ptr, \mu_\rho, \mu_y, root)$, and $w = (path, root_{sk^{COIN}}, path_{sk^{COIN}}, \tau_c, \rho_c, r_{c_1}, v, r_{c_2})$ for tuple $(x, w) \in L_{lead}$ iff:

- $pk^{COIN} = PRF^{pk}_{root_{sk}^{COIN}}(\tau_c)$.
- $\rho_{c_2} = PRF^{evl}_{root_{sk_{c_1}}^{COIN}}(\rho_{c_1})$. note here the nonce of the new coin is deterministically driven from the nonce of the old coin, this works as resistance mechanism to allow the same coin to be eligible for leadership more than once in the same epoch.
- $\forall i \in \{1, 2\} : DeComm(cm_{c_i}, pk^{COIN}||v||\rho_{c_i}, r_{c_i}) = T$.
- $path$ is a valid Merkle tree path to cm_{c_1} in the tree with the root $root$.
- $path_{sk^{COIN}}$ is a valid path to a leaf at position $sl - \tau_c$ in a tree with a root $root_{sk}^{COIN}$.
- $sn_{c_1} = PRF^{sn}_{root_{sk}^{COIN}}(\rho_{c_1})$
- $y = \mu_y^{root_{sk_{c_1}}^{COIN}||\rho_c}$

- $\rho = \mu_\rho^{root_{sk_{c_1}}^{COIN}||\rho_c}$
- $y < ord(G)\phi_f(v)$ note that this process involves renewing the old coin c_1 who's serial number gets revealed(proof of spending), becoming an input, to c_2 of the same value,

1.3 transfer transaction tx_{xfer}

transfer transaction of the pouring mechanism of input: old coin, and public coin, with output: new return change coin, and further recipient coin. such that input total value $v_1^{old} + v_{pub} = v_3^{new} + v_4^{new}$

$$tx_{xfer} = (TRANSFER, stx_{proof}, c_r)$$

$$stx_{proof} = (\{cm_{c_3}), cm_{c_4}\}), (\{sn_{c_2}, sn_{c_1}\}), \tau, root, \pi)$$

c_r is forward secure encryption of $stx_{rcpt} = (\rho_{c_3}, r_{c_3}, v_{c_3})$ to pk_r. the commitment of the new coins c_3, c_4 is:

$$(cm_{c_3}, r_{c_3}) = Comm(pk_{pk_s}^{COIN}||\tau||v_{c_3}||\rho_{c_3})$$

$$(cm_{c_4}, r_{c_4}) = Comm(pk_{pk_r}^{COIN}||\tau||v_{c_4}||\rho_{c_4})$$

1.3.1 spend proof

the spend proofs of the old coins sn_{c_1}, sn_{c_2} are revealed.

1.3.2 NIZK proof π

for the circuit inputs, and witnesses

$x = (\{cm_{c_3}, cm_{c_4}\}, \{sn_{c_1}, sn_{c_2}\}, \tau, root)$
$w = (root_{sk_{c_1}^{COIN}}, path_{sk_{c_1}^{COIN}}, root_{sk_{c_2}^{COIN}}, path_{sk_{c_2}^{COIN}}, pk_{c_3}^{COIN}, pk_{c_4}^{COIN}, (\rho_{c_1}, r_{c_1}, v_1, path_1), (\rho_{c_2}, r_{c_2}, v_2, path_2),$
$(\rho_{c_1}, r_{c_1}, v_1, path_1))$

π is a proof for the following transfer statement using zerocash pouring mechanism.

$$\forall_i \in \{1,2\} : pk_{c_i}^{COIN} = PRF^{pk}_{root_{sk_{c_i}}^{COIN}}(1)$$

$$\forall_i \in \{1,\ldots,4\} : DeComm(cm_{c_i}, pk_{c_i}^{COIN}||v_i||\rho_{c_i}, r_{c_i}) = T$$

$$v_1 + v_2 = v_3 + v_4$$

$$path_1 \text{ is a valid path to } cm_{c_1} \text{ in a tree with the root } root$$

$$path_2 \text{ is a valid path to } cm_{c_2} \text{ in a tree with the root } root, sn_{c_2} = PRF^{zdrv}_{root_{sk_{c_1}^{COIN}}}(\rho_{c_1})$$

$$path_{sk_{c_i}^{COIN}} \text{ is a valid path to a leaf at position } \tau \text{ in } , root_{sk_{c_i}^{COIN}} i \in \{1,2\}$$

$$sn_{c_i} = PRF^{sn}_{root_{sk_{c_i}^{COIN}}}(\rho_{c_i}), \forall_i \in \{1,2\}$$

2 Performance

since Crypsinous is based of sapling scheme, the performance relative to zerocash sapling scheme is that number of constraints in the PRF is improved by replacing sha256 (83,712 constraints) by pederson commitment (2,542 constraints), but on the other hand the proving take twice that of the sapling.

3 Appendix

3.1 PRF

pseudo random function $f(x)$ is defined as elliptic curve encryption over the group $< g >$ of random output as *elligator* curves of poseidon hash H

3.1.1 PRF^{sn}:

$$PRF^{sn}_{root^{COIN}_{sk}}(x) = H(x||0b00)^{root^{COIN}_{sk}}$$

3.1.2 PRF^{pk}:

$$PRF^{pk}_{root^{COIN}_{sk}}(x) = H(x||0b01)^{root^{COIN}_{sk}}$$

3.1.3 PRF^{evl}:

$$PRF^{evl}_{root^{COIN}_{sk}}(x) = H(x||0b10)^{root^{COIN}_{sk}}$$

3.2 $root^{COIN}_{sk}(\tau)$

the root in the merkle tree of the current epoch's coins secret keys, at the onset of the epoch, the initial slot's coin's secret key at time τ is sampled at random $sk^{COIN}_{\tau} \overset{s}{\leftarrow} \{0,1\}^{l_{PRF}}$, and $sk^{COIN}_{i+1} \leftarrow PRF^{evl}_{sk_i}(1)$

3.3 Comm,DeComm

the equivocal commitment $(cm, r) \leftarrow Comm(m)$, while the de-commitment is $DeComm(cm, m, r) \rightarrow True$ if it verifies. the commitment can be implemented as blinded encryption of m, as follows

$$mG_1 + rG_2$$

for random groups G_1, G_2, or as $PRF^{comm}_{r}(m)$

$$PRF^{comm}_{r}(m) = H(m||0b11)^{r}$$

4 references

https://eprint.iacr.org/2018/1132.pdf