Q&A

Nov 13th, 2019

Pedersen Commitment openings

- A Pedersen Commitment is a combination of 2 (or more) values

$$X = \underbrace{vG}_{\text{Secret value}} + \underbrace{\gamma H}_{\text{Blinding factor}}$$

- Even with infinite compute power and the possibility to infer value v from point vG is it impossible to get v from X

Pedersen Commitment openings

- A Pedersen Commitment is a combination of 2 (or more) values $X = \underbrace{vG} + \underbrace{\gamma H}$
- Opening up means “the value of v”
  - Opens to 0 : $X = \gamma H$
  - Opens to 0 or 1: It
    - either opens to 0 ($X = \gamma H$)
    - or it opens to 1 ($X = G + \gamma H$)

ZK-Proof opening to 0

- Given $X = aG + bH$.

- Why is (s, R) so that $sH = R + \mathscr{H}(G \,|\, R \,|\, X)X$ zero-knowledge proof that $a = 0$?

ZK-Proof opening to 0

Given $X = aG + bF$. (s, R) so that $sF = R + \mathscr{H}(G|R|X)X$

- Assume $a \neq 0$, then $\mathscr{H}(G|R|X)X = \mathscr{H}(G|R|X)qG + \mathscr{H}(G|R|X)bF$
- Then $R = -\mathscr{H}(G|R|X)qG + rF$ so that the G parts panel each other out

$$sF = \underbrace{\cancel{-\mathscr{H}(G|R|X)qG} + rF}_{R} + \underbrace{\cancel{\mathscr{H}(G|R|X)qG} + \mathscr{H}(G|R|X)bF}_{\mathscr{H}(G|R|X)X}$$

- $R = -\mathscr{H}(G|R|X)qG + rF$ is impossible, since the output to the hash is used as input (Hash depends on R)
- That only works if $a = q = 0$

What is the purpose of Σ-protocol for opening to 0 or 1?

What is the purpose of Σ-protocol for opening to 0 or 1?

- We use it as a building block
  - Not directly
  - We use the term $f = mx + r$ from the Σ-protocol.
  - x is the output of the hash and r is a secret value. If you know f, you cannot infer what m is, because r is secret. Using the Σ-protocol, however, you have a proof that m is either 0 or 1.

Example

- Assume a 1-in-3 proof
  - $X = uG + xH, Y = vG + yH, X = wG + zH$
- I know only $Y = vG + yH$
- I can give you $0X + 1Y + 0Z = Y$, but I want to hide the coefficients
  - Yet you need a proof that each coefficient is 0 or 1

Example

$$X = uG + xH,\ Y = vG + yH,\ X = wG + zH$$

- If we use the Σ-protocol, we get f_0, f_1, f_2

$$f_x X + f_y Y + f_z Z = (m_x x + r_x)X + (m_y x + r_y)Y + (m_z x + r_z)Z$$

- And that is

$$x(m_x X + m_y Y + m_z Z) + \underbrace{(r_0 X + r_1 Y + r_2 Z)}_{R}$$

- R is independent of x and can be computed beforehand

Example

$$X = uG + xH,\ Y = vG + yH,\ X = wG + zH$$

$$x(m_x X + m_y Y + m_z Z) + \underbrace{(r_0 X + r_1 Y + r_2 Z)}_{=R}$$

Example

$X = uG + xH,\ Y = vG + yH$

The protocol to open a coin is now

- I Give $R' = R + \alpha H, v$
- We do 3 parallel Σ-protocols, resulting in f_x, f_y, f_z
- I also provide signature (s, V)
- You compute $Q = f_1 X + f_2 Y + f_3 Z - R' - vG$
  - Thus: $Q = Y + \alpha H - vG = (\alpha + y)H$
- And check if it opens to 0 (via Schnorr-Signature):

$$sH \stackrel{?}{=} V + \mathcal{H}(Q \,|\, G \,|\, H \,|\, V)Q$$

Example

- The protocol to open a coin involves the plaintext value v
- For JoinSplit transactions, we have several input coins and several output coins, but the sum is 0
  - 1-in-N protocol for each input coin
    - $\underbrace{x(f_x X + f_Y Y + f_z Z) - R'}_{\text{input coin}} - \underbrace{(vG + \alpha H)}_{\text{output coin}}$ opens to 0
  - (Lelantus also has a serial number term $vG + sF + \alpha H$)