Binding Signature

Consistency of balance with the claim description and the output description which would be minted in the MASP pool is enforced by the *Airdrop Binding Signature*. Similarly to a standard MASP or Sapling transaction, this signature has a dual role:

- To prove that the total value claimed in NAM within the output descriptions respects the allowed conversion set with the *zatoshi* allocated in the claim description.
- To prove that signer knew the randomness used for the Claim, Output and Convert *value commitments*, in order to prevent reply attacks.

Let $\mathbb{J}^{(r)}$ and $\mathbb{J}^{(r)*}$ be defined as in §5.4.9.3 'Jubjub' of the *Zcash Protocol Specification*. Following §5.4.8.3 'Homomorphic Pedersen commitments (Sapling and Orchard)' from the former specs we have:

$$\mathsf{ValueCommit}^{\mathsf{x}} : \mathsf{ValueCommit}^{\mathsf{Sapling}}.\mathsf{Trapdoor} \times \{-\frac{r_{\mathbb{J}-1}}{2} .. \frac{r_{\mathbb{J}-1}}{2}\} \to \mathsf{ValueCommitment}^{\mathsf{Sapling}}.\mathsf{Output}$$

$\mathsf{ValueCommit}^{\mathsf{Sapling}}$ is the above function with:

- $\mathcal{R}^{\mathsf{Sapling}} = \mathsf{FindGroupHash}^{\mathbb{J}^{(r)*}}(''\text{Zcash\_cv}'', ''\text{r}'')$
- $\mathcal{V}^{\mathsf{Sapling}} = \mathsf{FindGroupHash}^{\mathbb{J}^{(r)*}}(''\text{Zcash\_cv}'', ''\text{v}'')$

$\mathsf{ValueCommit}^{\mathsf{MASP}}$ is the above function with:

- $\mathcal{R}^{\mathsf{MASP}} = \mathsf{FindGroupHash}^{\mathbb{J}^{(r)*}}(''\text{MASP\_r\_\_}'', ''\text{r}'')$
- $\mathcal{V}^{\mathsf{MASP}} \equiv \mathsf{vb}_{\mathsf{NAM}} = \mathsf{abst}_{\mathbb{J}}(\mathsf{PRF}^{\mathsf{vcgMASP}}(\mathsf{t}_{\mathsf{NAM}}))$

Where $\mathsf{t}_{\mathsf{NAM}}$ is the bytestring representing the NAM asset type.

$\mathsf{ValueCommit}^{\mathsf{mint}}$ is the above function with:

- $\mathcal{R}^{\mathsf{MASP}}$
- $\mathcal{V}^{\mathsf{mint}} = [\mathsf{V}_{\mathsf{MASP}}]\mathcal{V}^{\mathsf{MASP}} + [\mathsf{V}_{\mathsf{Sapling}}]\mathcal{V}^{\mathsf{Sapling}}$

where $\mathsf{V}_{\mathsf{Sapling}}$ and $\mathsf{V}_{\mathsf{MASP}}$ are defined in a *allowed conversion* $\{(\mathsf{A}_{\mathsf{Sapling}}, \mathsf{V}_{\mathsf{Sapling}}), (\mathsf{A}_{\mathsf{MASP}}, \mathsf{V}_{\mathsf{MASP}})\}$, as described in the *Multi Asset Shielded Pool Specification* § 0.12.4 'Convert'.

Since the value commitments in Sapling and MASP use different random base, introduce the *Randomness Renormalization Factor* $\mathcal{N} : \mathbb{J}^{(r)}$

$$\mathcal{N} = [\mathsf{rcv}^{\mathsf{Sapling}}]\mathcal{R}^{\mathsf{MASP}} - [\mathsf{rcv}^{\mathsf{Sapling}}]\mathcal{R}^{\mathsf{Sapling}}$$

Suppose the transaction has:

- A *Claim description* with value commitment $\mathsf{cv}^{\mathsf{Sapling}}$, committing to value $\mathsf{v}^{\mathsf{Sapling}}$ with randomness $\mathsf{rcv}^{\mathsf{Sapling}}$ using $\mathsf{ValueCommit}^{\mathsf{Sapling}}$
- An *Output description* with value commitment $\mathsf{cv}^{\mathsf{MASP}}$, committing to value $\mathsf{v}^{\mathsf{MASP}}$ with randomness $\mathsf{rcv}^{\mathsf{MASP}}$ using $\mathsf{ValueCommit}^{\mathsf{MASP}}$
- A *Convert description* with value commitment $\mathsf{cv}^{\mathsf{mint}}$, committing to value $\mathsf{v}^{\mathsf{mint}}$ with randomness $\mathsf{rcv}^{\mathsf{mint}}$ using $\mathsf{ValueCommit}^{\mathsf{mint}}$

Validators calculate the *airdrop transaction validating key* as:

$$\mathsf{bvk}^{\mathsf{Airdrop}} = \mathsf{cv}^{\mathsf{Sapling}} + \mathsf{cv}^{\mathsf{mint}} - \mathsf{cv}^{\mathsf{MASP}} + \mathcal{N}$$

The signer calculates the *airdrop transaction signing key* as:

$$\mathsf{bsk}^{\mathsf{Airdrop}} = \mathsf{rcv}^{\mathsf{Sapling}} + \mathsf{rcv}^{\mathsf{mint}} - \mathsf{rcv}^{\mathsf{MASP}}$$

In order to check for implementation faults, the signer SHOULD also check that

$$\mathsf{bvk}^{\mathsf{Airdrop}} = \mathsf{BindingSig}^{\mathsf{Airdrop}}.\mathsf{DerivePublic}(\mathsf{bsk})$$

Let $\mathsf{SigHash}$ be the *SIGHASH transaction hash* as defined in [ZIP-243], not associated with an input, using the *SIGHASH type* SIGHASH_ALL.

A validator checks balance by validating that

$$\mathsf{BindingSig}^{\mathsf{Sapling}}.\mathsf{Validate}_{\mathsf{bvk}^{\mathsf{Airdrop}}}(\mathsf{SigHash}, \mathsf{bindingSigAirdrop}) = 1.$$