Aleo on-chain MPC Beaver Triple Generation

Pierre-André LONG

July 17, 2024

Let $(v_i)_{1 \leqslant i \leqslant n}$ be the n MPC share holders, called validators. For $1 \leqslant i, j \leqslant n$:
AHE_i: additively homomorphic encryption with public key of v_i.
AHD_i: decryption with private key of v_i.
k is a unique base field element identifying the beaver triple we are generating.
$k_A = commit_{bhp256}(k, 1), k_B = commit_{bhp256}(k, 2), k_{AB} = commit_{bhp256}(k, 3)$, indexes for storing each value of the triple in the MPC.

Validator i: Alice		**Validator j: Bob**
- Generates random a_i. - Calls `add_private`(k_A, a_i). - Mints n records containing $c_{A_{i,j}} = AHE_i(a_i)$ to v_j.	$\xrightarrow{c_{A_{i,j}}}$	
		- Generates random $b_j, r_{i,j}$. - Calls `add_private`(k_B, b_i). - Calls `add_private`$(k_{AB}, -r_{i,j})$. - Computes: $c_{B_{i,j}} = b_j \cdot c_{A_{i,j}} + AHE_i(r_{i,j})$ - Mints n records containing $c_{B_{i,j}}$ to v_i.
	$\xleftarrow{c_{B_{i,j}}}$	
- Computes: $S_{i,j} = AHD_i(c_{B_{i,j}}) = a_i \cdot b_j + r_{i,j}$ - Calls `add_private`$(k_{AB}, S_{i,j})$.		

Hence:

$$[k_{AB}] = \left[\sum_{i=1}^{n}\sum_{j=1}^{n}(S_{i,j} - r_{i,j})\right] = \left[\sum_{i=1}^{n}\sum_{j=1}^{n} a_i \cdot b_j\right] = [a \cdot b]\,,$$

$$[k_A] = [a] \text{ and } [k_B] = [b]\,.$$