Blind bid protocol v0.22

Dmitry Khovratovich

Dusk Foundation

July 2, 2019

1 Notation

Let E be the Bulletproof curve with prime order r.
Variables:

- Seed S – 32-byte string.
- Secret K – 32-byte string.
- Transaction hash X – 32-byte string.
- Bidding data M – 32-byte string.
- Merkle tree $\mathcal{T}$ of bids with root R_T.
- Coin amount d – integer between 0 and 2^{64}.
- Counter N – 32-byte string.

Functions:

- H – Poseidon, a Bulletproofs-friendly hash function. Maps strings of $\mathbb{F}_r$ to $\mathbb{F}_r$.
- $\mathcal{H}(X, \mathcal{O})$ Merkle root construction function. It assumes that $\mathcal{O}$ is a Merkle opening for X in a tree built using H, and outputs the tree root corresponding to the opening.
- $F(d, Y)$ – score function. Takes 64-bit input d and 256-bit input Y and operates as follows:
  - Truncate Y to left 128 bits and interpret the result as 128-bit integer Y'.
  - Output $f = (d \cdot 2^{128})/Y'$, where division is the integer division.

2 Circuit for blind bid computation

Let C be the following computation:

- **Public Input:** S.
- **Private Input:** K, d.
- **Flow**

1. $M = H(K)$;
2. $Z = H(S, K)$;
3. $C_d = g^d h^r$
4. $X = H(C_d, M, S)$;
5. $\mathcal{H}(X, \mathcal{O}) = R$.
6. $Y = H(S, X, K)$;
7. $Q = F(d, Y)$.

Public Output: Z, R, Q

Then Π is the Bulletproof proof of computational integrity of C.

3 Protocol

Procedure:

1. Seed S is computed and broadcasted.
2. Bidder selects secret K.
3. Bidder, at most once per seed, sends a bidding transaction with data $M = H(K)$ and proof of knowledge of K.
4. For every bidding transaction with d coins in the form of commitment C_d and data M the uniqueness of M is verified and entry $X = H(C_d, M, S)$ is added to $\mathcal{T}$.
5. Potential bidder computes $Y = H(S, X, K)$, score $Q = F(d, Y)$, and identifier $Z = H(S, K)$.
6. Bidder selects a bid root R_T and broadcasts (Z, R_T, Q, π) where
$$\pi = \Pi(Z, R_T, Q, S;\ K, d).$$
7. The proof with the highest Q wins.
8. The winner can use Z to identify himself during the block generation.

Remark 1. By definition $Y' < 2^{128}$ so $Q > d$. Thus a score for each bidder is higher than his bid. Therefore, the winning score Q_{max} is bigger than any bid as otherwise the owner of such bid would have a bigger score.

4 Security

Requirements:

1. A tuple (Z, R, Q, π) is a proof of knowledge of secret K such that $Z = H(S, K)$.
2. **Bid binding** For given Z it is infeasible to find two different bids that yield the same Z.
3. **Bid privacy** Based on the score, no bid can be ruled out as a winner. Close bids have close probability to win.

Proofs:

1. π is a proof of knowledge of K used in the computation of Z, according to the properties of the Bulletproofs proof system and to the description of computation C.

2. Assuming collision resistance of H it is infeasible to find distinct M, M' giving the same Z or distinct K, K' that ield the same M. Therefore for one Z can exist only one M and one X (by the uniqueness requirement of M), and thus the only possible bid.

3. We first show that for any pair $(d, Q), d < Q$, (see Remark 1) there exists 128-bit Y such that $F(d, Y) = Q$. We note that in the bid computation we get $Q > d$ Let us consider

$$y = (d \cdot 2^{128})/Q.$$

We obtain that $d \cdot 2^{128} = Qy + v$, where $v < Q$. This implies

$$(d \cdot 2^{128})/y = Q + v/y = Q.$$

The last equation holds since Q (around 64 bits) is much smaller than y (128 bits, as $d < Q$). Therefore each bid can be a winner and can not be ruled out.

Now consider two bids d and $d + \epsilon$ and compare the probabilities $P_y = y/2^{128}$ of getting hashes y and y' based on their size as integers. For given Q, we can compute candidates as

$$y = (d \cdot 2^{128})/Q; \quad y' = ((d + \epsilon) \cdot 2^{128})/Q$$

It follows that $y' \approx (1 + \epsilon/d)y$, so we get

$$P_y/P_{y'} = y/y' \approx 1 + \epsilon/d.$$

So we conclude that the quotient of probabilities is approximately equal to the quotient of bids.