Vector Commitments with Efficient Proofs

Markulf Kohlweiss
Microsoft Research Cambridge, UK
markulf@microsoft.com

Alfredo Rial
IBBT and K.U.Leuven, ESAT-COSIC, Belgium
alfredo.rial@esat.kuleuven.be

Motivation. In privacy-preserving smart metering [RD10], users receive signed consumption readings from their meters and a signed tariff policy from the service provider. Each user calculates the fee to be paid and reveals the fee to the service provider, along with a zero-knowledge proof of correctness of the fee calculation.

Existing zero-knowledge proofs are inefficient, e.g., when the tariff policy consists of several functions signed by different entities. As a concrete example, consider a setting with a meter $\mathcal{M}$, a provider $\mathcal{P}$, a government agency $\mathcal{A}$ and a user $\mathcal{U}$. $\mathcal{M}$ measures user's consumption of, e.g., electricity or water, and sends $\mathcal{U}$ signed readings (Table 1). Each table row contains a consumption value, a consumption time interval and a signature on both the consumption and the time.

$\mathcal{P}$ sends $\mathcal{U}$ a signed tariff policy (Table 1) which details the price per unit of consumption for each time interval. $\mathcal{A}$ sends $\mathcal{U}$ another signed tariff policy (Table 1). $\mathcal{A}$'s tariff policy varies for each user and consequently $\mathcal{U}$'s identity is also signed.

To compute the fee, $\mathcal{U}$ chooses, for each time interval, the policy that offers the lowest rate. $\mathcal{U}$ multiplies the consumption at each time interval by this rate, and next sums up the costs of all the time intervals.

The zero-knowledge proof of fee correctness must keep secret which tariff policy $\mathcal{U}$ employed in each time interval. This proof involves an OR statement where $\mathcal{U}$ proves that, for each interval, the employed rate was signed either by $\mathcal{P}$ or by $\mathcal{A}$. The proof size grows with the number of tariff policies involved.

Our Contribution. We propose a method to reduce the proof size based on a novel primitive called Vector Commitments (VC). A VC scheme allows to commit to a vector $(x_1, \ldots, x_n)$ and open it to x_i $(i \in [1, n])$ with (unlike traditional commitments) cost independent of n. We present three VC constructions secure under the SDH, DHE and CDH assumptions respectively, and equip them with efficient zero-knowledge proofs of correct commitment generation and of an opening.

Before calculating the fee, $\mathcal{U}$ computes an intermediate table with the lowest rates (Table 1). $\mathcal{U}$ commits to the vector of lower rates contained in the intermediate table and proves in zero-

Meter Readings		
Cons.	Time	Sig.
1534	00:00	$\sigma_m(\mathrm{r}_1)$
1300	00:15	$\sigma_m(\mathrm{r}_2)$
1213	00:30	$\sigma_m(\mathrm{r}_3)$
⋮	⋮	⋮

Provider Policy		
Time	Rate	Sig.
00:00	10	$\sigma_p(\mathrm{r}_1)$
00:15	9	$\sigma_p(\mathrm{r}_2)$
00:30	8	$\sigma_p(\mathrm{r}_3)$
⋮	⋮	⋮

Agency Policy		
Time	Rate	Sig.
00:00	11	$\sigma_a(\mathrm{r}_1,\mathcal{U})$
00:15	8	$\sigma_a(\mathrm{r}_2,\mathcal{U})$
00:30	7	$\sigma_a(\mathrm{r}_3,\mathcal{U})$
⋮	⋮	⋮

Int. Table	
Time	Rate
00:00	10
00:15	8
00:30	7
⋮	⋮

Table 1: Meter readings, provider tariff policy, agency tariff policy and intermediate table

knowledge its correctness, i.e., for each vector component, $\mathcal{U}$ proves that it was signed either by $\mathcal{P}$ or by $\mathcal{A}$.

To prove fee correctness, $\mathcal{U}$ proves that the employed rate was committed in the correct vector component. Since VC schemes allow for zero-knowledge proofs of commitment opening of size independent of the vector length, this proof is more efficient than the proof of the OR statement mentioned above. Consequently, when the use of the intermediate table surpasses a threshold, the cost of creating and proving correctness of the intermediate table is amortized.

Related Work

Accumulators. Accumulators [BdM93] allow to commit to sets $S = \{x_1, \ldots, x_n\}$ such that it is possible to open the commitment to $x \in S$ with cost independent of $|S|$. (Some accumulators also allow to prove that $x \notin S$.) The main difference with respect to VC schemes is that the committed $\{x_1, \ldots, x_n\}$ are not bound to a particular vector component, which is needed for our zero-knowledge proofs. It is possible to bind x_i to the position i by committing to $\{1||x_1, \ldots, n||x_n\}$. However, such a construction of VC schemes based on accumulators leads to rather inefficient zero-knowledge proofs of correct commitment generation and opening, since they would require a range proof to prove that each component consists of two fields $i||x_i$.

Polynomial Commitments. A polynomial commitment scheme [KZG10] allows to commit to a polynomial $\phi(x)$ and to open the commitment to polynomial evaluations $\phi(x)$. Polynomial commitments imply VC schemes, since one can compute a commitment to a vector $(x_1, \ldots, x_n)$ by interpolating the points $\{(1, x_1), \ldots, (n, x_n)\}$ to obtain a polynomial $\phi(x)$ and committing to $\phi(x)$. In fact, our construction secure under the SDH assumption is a simplification of the polynomial commitment scheme in [KZG10] that follows that methodology. We augment this construction with ZKPK's of correct commitment generation and of an opening, which allow this construction to be applicable to a wider variety of cryptographic protocols.

Vector Commitments for ZKS. Concise mercurial vector commitments [LY10] allow to commit to a vector $(x_1, \ldots, x_n)$ and to open it to x_i with cost independent of n. The mercurial property implies that there are two commitment procedures. A soft commitment procedure that generates a dummy value which later can be softly opened to any value, and a hard commitment procedure that generates commitments which can be (hardly or softly) opened to only one value. Concise mercurial vector commitments are useful to construct zero-knowledge sets (ZKS) [MRK03]. Our construction secure under the DHE assumption is a simplification of the construction in [LY10]. Again, we augment our construction with ZKPK's of correct commitment generation and of an opening.

Very recently, it has been shown that concise mercurial vector commitments can be obtained from mercurial commitments and a primitive also called vector commitments [CF11]. The functionality provided by vector commitments in [CF11] is the same as the one we propose, but the security properties are different. In particular, since they focus on the use of VC schemes to construct ZKS, they do not require vector commitments to be hiding and they propose a definition for the biding property stronger than ours. Our construction secure under the CDH assumption modifies the construction in [CF11] and equips it with efficient ZKPK.

References

[BdM93] Josh Cohen Benaloh and Michael de Mare. One-way accumulators: A decentralized alternative to digital sinatures (extended abstract). In *EUROCRYPT*, pages 274–285, 1993.

[CF11] Dario Catalano and Dario Fiore. Concise vector commitments and their applications to zero-knowledge elementary databases. Cryptology ePrint Archive, Report 2011/495, 2011. `http://eprint.iacr.org/`.

[KZG10] Aniket Kate, Gregory M. Zaverucha, and Ian Goldberg. Constant-size commitments to polynomials and their applications. In Masayuki Abe, editor, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 177–194. Springer, 2010.

[LY10] Benoît Libert and Moti Yung. Concise mercurial vector commitments and independent zero-knowledge sets with short proofs. In Daniele Micciancio, editor, *TCC*, volume 5978 of *Lecture Notes in Computer Science*, pages 499–517. Springer, 2010.

[MRK03] Silvio Micali, Michael O. Rabin, and Joe Kilian. Zero-knowledge sets. In *FOCS*, pages 80–91. IEEE Computer Society, 2003.

[RD10] Alfredo Rial and George Danezis. Privacy-preserving smart metering. Technical Report MSR-TR-2010-150, Microsoft Research, November 2010.