Updatable proofs for hard languages
Draft, September 30, 2019

Michał Zając

Clearmatics

Definition 1 (Decisionally hard language). *We call language $\mathcal{L}$* decisionally hard *if for all λ and PPT algorithm $\mathcal{D}$*

$$\left|\Pr[\mathcal{D}(x) = 1 \mid x \in \mathcal{L}] - \Pr[\mathcal{D}(x) = 1 \mid x \notin \mathcal{L}]\right| \leq \mathsf{negl}(\lambda) \ .$$

Given NP decisional hard language $\mathcal{L}$ we define $\mathbf{R}$ its corresponding relation if $\mathsf{x} \in \mathcal{L}$ iff there exists w such that $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$.

We skip in this draft full formal definition of zero knowledge system, see e.g. [Gro06], and recall only on the updatable knowledge soundness property as defined in [**?**].

Definition 2 (Updatable knowledge soundness).

Definition 3 (Fully updatable proof system). *We call a zero-knowledge proof system* NIZK fully updatable *if it is updatable-sound and equipped with an efficiently computable function* $\mathsf{updateProof}$ *that given an instance π, proof π_0 valid for an SRS srs_0, updated SRS srs_n, and proofs of correct updates $\{\rho_i\}_{i=0}^{n}$ returns proof π_n, such that $\mathsf{V}(\mathsf{srs}_n, \{\rho_i\}_{i=0}^{n}, \mathsf{x}, \pi_n)$ accepts.*

Theorem 1. *Let $\mathcal{L}$ be a decisionally hard NP language and $\mathbf{R}$ its corresponding instance-witness relation. Then there exists no fully updatable zero-knowledge proof system for $\mathbf{R}$.*

Proof. We proceed by contradiction. Let NIZK be such a zk proof system, we build an algorithm $\mathcal{D}$ that on input x decides whether x belongs to $\mathcal{L}$ with non-negligible advantage.

First, $\mathcal{D}$ sets up SRS update oracle $\mathsf{U}\text{-}\mathcal{O}_{\mathsf{s}}$ and queries it on the setup intent getting SRS srs_0 and proof ρ of its correctness. Since $\mathcal{D}$ runs $\mathsf{U}\text{-}\mathcal{O}_{\mathsf{s}}$ internally, she learns a trapdoor td_0 corresponding to srs_0 and is able to produce a simulated proof π_0 for x. Second, $\mathcal{D}$ runs a PPT algorithm $\mathcal{A}$ that is given $\mathsf{srs}_0, \{\rho_0\}$ and access to $\mathsf{U}\text{-}\mathcal{O}_{\mathsf{s}}$. In the end, the SRS is srs_n, for some natural n, and the sequence of update correctness proofs $\{\rho_i\}_{i=0}^{n}$. Since NIZK is fully updatable, verifier $\mathsf{V}(\mathsf{srs}_n, \{\rho\}_{i=0}^{n}, \mathsf{x}, \mathsf{updateProof}(\mathsf{srs}_n, \{\rho\}_{i=0}^{n}, \pi_0))$ accepts for $\mathsf{x} \notin \mathcal{L}$ with negligible probability only.

References

Gro06. Jens Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 444–459. Springer, Heidelberg, December 2006. `doi:10.1007/11935230_29`. 1