

Cryptographic Protocols

Notes 3

Scribe: Sandro Coretti (modified by Chen-Da Liu Zhang)

About the notes: These notes serve as written reference for the topics not covered by the papers that are handed out during the lecture. The material contained therein is thus a *strict* subset of what is relevant for the final exam.

This week, the notes discuss the definition of (perfect) zero-knowledge and a proof that the three-move protocols we have encountered so far (graph isomorphism, Fiat-Shamir, Guillou-Quisquater, Schnorr) are perfectly zero-knowledge [Mau09, Theorem 2].

3.1 Definition of Zero-Knowledge

Intuitively, an interactive proof (P, V) between a prover P and verifier V is zero-knowledge if after interacting with P , *any* verifier V' has no more information than before executing the protocol. This is captured by the notion of a *simulator* S that reproduces V' 's view in the proof without actually communicating with P .

More precisely, consider the following two random experiments:

1. Prover P interacts with V' ; let Z be the random variable corresponding to the resulting transcript and P_Z its distribution.
2. Simulator S interacts with V' and outputs a transcript; let Z' denote the corresponding random variable and $\hat{P}_{Z'}$ its distribution.

Definition 3.1. *An interactive proof (P, V) is (perfectly) zero-knowledge if for every efficient V' there exists an efficient simulator S (with access to V') producing a transcript Z' that is distributed identically to the transcript Z in the actual interaction between P and V' , i.e.,*

$$P_Z = \hat{P}_{Z'}.$$

The interactive proof is honest-verifier zero-knowledge (HVZK) if the simulator exists for (the honest) verifier V .

In this course, when proving the zero-knowledge property, there will always be a single simulator S that works for all verifiers V' . This is referred to as *black-box* simulation.

3.2 Honest-Verifier Zero-Knowledge and c -simulatability

The HVZK property is perhaps not very interesting per se, but it is a useful tool in proving (perfect) zero-knowledge. All three-move protocols in this course satisfy the even stronger notion of c -simulatability.

Definition 3.2. A three-move protocol round of an interactive proof (P, V) for a language L with challenge space \mathcal{C} is c -simulatable¹ if for any value c one can efficiently generate a triple (t, c, r) with the same distribution as occurring in the protocol (between P and the honest V) conditioned on the challenge being c .

In other words, there has to exist an efficient algorithm that given any $x \in L$ and $c \in \mathcal{C}$, produces values t and r with a distribution $\hat{P}_{TR|C}$ such that $\hat{P}_{TR|C}(t, r, c) = P_{TR|C}(t, r, c)$ for all t, c , and r , where $P_{TR|C}(t, r, c)$ is the distribution occurring in the actual protocol conditioned on the challenge being c .

It is easily seen that if the challenge is efficiently samplable, c -simulatability implies HVZK: the honest-verifier simulator simply chooses $c \in \mathcal{C}$ uniformly at random and generates t and r according to the c -simulatability.

It is also easy to see that HVZK (and ZK) compose sequentially. That is, an interactive proof (P, V) consisting of s independent perfect HVZK (resp. ZK) three-move rounds is also perfect HVZK (resp. ZK): The simulator simply appends the transcripts of the simulators in each round.

3.3 Proving the Zero-Knowledge Property

In this section we show that an interactive proof (P, V) consisting of independent perfectly HVZK three-move rounds is perfectly zero-knowledge if, additionally, the challenge space \mathcal{C} is not too large.

3.3.1 Perfect Zero-Knowledge

Lemma 3.1. An HVZK three-move protocol round of an interactive proof (P, V) where V chooses the challenge uniformly at random from a polynomially bounded challenge space \mathcal{C} is zero-knowledge.

Proof. Consider a potentially dishonest verifier V' . The simulator S has *black-box rewinding access* to V' . This means that S cannot see the code of V' (hence, it uses it as a black-box), but S may rewind V' at any point to an earlier state in its computation.

Simulator S creates a transcript as following:

1. Generate a triple (t, c, r) according to the HVZK simulation.
2. Pass t to V' and receive the challenge c' .
3. If $c = c'$, output the triple (t, c, r) . Otherwise, rewind V' to the first point and repeat the simulation attempt.

The expected number of trials is $|\mathcal{C}|$, which is polynomial by assumption (the HVZK simulator returns a uniformly random challenge c independent from c'). Also, the distribution of the transcript generated by the simulator S is the same as the transcript generated in the real protocol (P, V) . \square

Corollary 3.2. An interactive proof (P, V) consisting of s independent perfectly HVZK three-move rounds so that in every round V chooses the challenge uniformly at random from the same polynomially bounded challenge space \mathcal{C} is perfectly zero-knowledge.

¹This is also called *special HVZK* in the literature.

Note that Corollary 3.2 is a slightly more general than Theorem 2 in [Mau09] in that it works for any HVZK protocol and not only for c -simulatable ones.

Proof. With Lemma 3.1, we know that an HVZK three-move round with uniform challenge from a polynomially bounded \mathcal{C} is zero-knowledge. Also, the sequential composition of s independent ZK three-move rounds is ZK. \square

References

- [Mau09] Ueli M. Maurer. Unifying zero-knowledge proofs of knowledge. In *AFRICACRYPT*, pages 272–286, 2009.