COM S 6830 – Cryptography Nov, 11, 2011

Lecture 19: Zero Knowledge Proofs - Part 4

Instructor: Rafael Pass *Scribe: Shentong Wang*

Review: Definition for Zero Knowledge

As a review of the previous lecture, it was defined that $(\mathcal{P}, \mathcal{V})$ is a Zero Knowledge proof for L with respect to R_L if the following holds:

1. Soundness: $\forall x \in L, \exists y \in \{0,1\}^* s.t. Pr[Out_v[P(x,y) \leftrightarrow V(x)] = 1] = 1$
2. Completeness: $\forall P^*, \forall x \notin L, \forall y \in \{0,1\}^* s.t. Pr[Out_v[P^*(x,y) \leftrightarrow V(x)] = 1] \leq \epsilon(n)$
3. For all PPT V^*, there exists an expected PPT S such that for all $x \in L$, $w \in R_L(x)$, and $z \in \{0,1\}^*$, the following distributions are identical:
   (a) $View_{V^*}[P(x,w) \Leftrightarrow V^*(x,z)]$
   (b) $S(x,z)$

Alternative Definition

The view of the Verifier in a Zero Knowledge Proof is hard to define in some cases. Thus, we can alter from the View to the Output of the Verifer. Thus, following two definitions are equivalent:

1. $View_{V^*}[P(x,w) \Leftrightarrow V^*(x,z)]$
2. $Out_{V^*}[P(x,w) \Leftrightarrow V^*(x,z)]$

These two definitions are equivalent for the following reasons:

1. Knowing the view for V^* implies Knowing the output for V^*, because with the view for V^*, we can run V^* and provide its view to get its output. Since V^* is a PPT, these operations can be completed in polynomial time.
2. Knowing the output for V^* implies Knowing view for V^*, because for every V^*, we can construct another V^{**}, which outputs all its view. This new V^{**} must also run in polynomial time, and therefore, knowing this V^{**}'s output implies knowing V^*'s view.

It is important to note that both these definitions are not closed under parallel execution. The main intuition behind this statement is that the black box simulator constructed earlier cannot guess many opposing verifiers if they can collaborate in parallel under polynomial constraints.

Witness Indistinguishability

Thus, in finding that closure under parallel execution is hard to prove for the definition Zero Knowledge, it would be wise to turn to an alternative weaker definition which has a subset of the desired qualities in Zero Knowledge. Thus, define Witness Indistinguishability, which roughly states that if there are two different witnesses for x, then a verifier should not be able to distinguish between the two witnesses after talking to the prover.

Formalization

Let $(\mathcal{P}, \mathcal{V})$ be an Interactive Proof with efficient prover for L with witness relation R_L . $(\mathcal{P}, \mathcal{V})$ is Witness Indistinguishable with respect to R_L if for all PPT V^*, $x \in L$, $w_1, w_2 \in R_L(x)$, and $z \in \{0,1\}^*$, the following distributions are indistinguishable:

1. $Out_{V^*}[P^*(x, w_1) \Leftrightarrow V^*(x, z)]$
2. $Out_{V^*}[P^*(x, w_2) \Leftrightarrow V^*(x, z)]$

Weakness of Witness Indistinguishability

The shrewd observer will notice that this definition of Witness Indistinguishability bears an uncommon resemblance to Message Indistinguishability in encryption. Thus, from this insight, it is easy to see that Zero Knowledge Proofs implies Witness Indistinguishability, beause if there exists a simulator is indistinguishable from a proover both witnesses, then neither will be distinguishable from each other from an application of the Hybrid Lemma. On the other hand, if the same technique for proving Message Indistinguishability implies Zero Knowledge Encryption is used for proving Witness Indistinguishability implies Zero Knowledge Proofs, one runs into the problem of not being able to find a polynomial time computable generic witness for the Simulator to use in all cases as in the Message Indistinguishability case, where a Simulator can encrypt a known message. (namely $\{0\}^n$) Since the main application of Zero Knowledge Proofs is in NP-Hard witnesses, this poses as a serious problem for constructing a simulator. Witness Indistinguishability has even more problems when there exists a cannonical witness for all x's which can be reduced to from all other witnesses for x efficiently or even worse, when there exists only one witness for all x, in which case, Witness Indistinguishability becomes vacuously true for any interactive proof. Thus, it can be concluded that Witness Indistinguishability is strictly weaker than Zero Knowledge.

Witness Indistinguishability as a Relaxation of Zero Knowledge Simulators

The main reason Witness Indistinguishability does not imply Zero Knowledge is the restriction of efficient simulators, because some witness might not be able to be found with efficient simulators. Thus, if this restriction can be relaxed, to where simulators are bounded exponentially instead of polynomially, it can be shown that Witness Indistinguishability implies Zero Knowledge. This is because with exponential bounded simulators, a witness can be found by brute force. Therefore, afterwards, the proof continues similar to the encryption case. Thus, we can represent the spectrum of simulator bounds as follows:

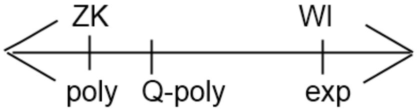

Figure 1: **Rasterization Increments**

Thus, the reasonable follow-up to this analysis would be whether there are any definitions between Zero Knowledge and Witness Indistinguishable which hold importance. There in fact is. Another relaxation is to restrict simulators to quasi-poly bounded simulators. (Simulators which run in $O(2^{(\log(n))^c})$) This family of Interactive Proofs are also significant in cryptography. (This will be explored in the homework)

Closure Under Parallel Execution of Witness Indistinguishability

One of the strengths of Witness Indistinguishability is that Witness Indistinguishability is closed under parallel execution. Thus, if $(\mathcal{P}, \mathcal{V})$ is Witness Indistinguishable, then $(\mathcal{P}^n, \mathcal{V}^n)$ is also Witness Indistinguishable. (In this context, $(\mathcal{P}^n, \mathcal{V}^n)$ is running verifier V and P in parallel n times) In fact, different verifiers can be run out of sync in parallel and Witness Indistinguishability is still held. The proof is simple and involves a single application of the hybrid lemma over the different combinations of w_1 and w_2 provers. Note that this property holds only if all provers are efficient. Otherwise, an efficient distinguisher for $(\mathcal{P}, \mathcal{V})$ can not be constructed from $(\mathcal{P}^n, \mathcal{V}^n)$ through closure under efficient operations. This proof does not extend to the Black Box Zero Knowledge case, because it is hard to isolate the different verifiers running in parallel to match paralell simulators to. Recall that with a Black Box Zero Knowledge simulator is privied to all of the verifier's randomness and outside influences. This becomes hard to construct if each of the verifiers communicate with each other during the parallel execution. There are some definitions for Zero Knowledge involving isolated independent parallel runs of Verifiers, but these definitions disregard some of the most relevant attacks on Zero Knowledge Proofs.

Witness Hiding

Another relaxation of Zero Knowledge is Witness Hiding, which can be defined as Interactive Proofs which cannot be used to expose the witness they are using to prove a certain x.

Hard Ensembles

An Ensemble $\{D_n\}_{n\in\mathbb{N}}$ is hard for R_L if for all nuPPT $\mathcal{A}$, there exists negligible function ϵ, where for all n in $\mathbb{N}$, $Pr[x \leftarrow D_n : A(x) \in R_L(x)] \leq \epsilon(n)$. Thus, $\{D_n\}_{n\in\mathbb{N}}$ is an ensemble of languages with hard to compute witnesses.

Formalization of Witness Hiding

With the definition of Hard Ensembles, $(\mathcal{P}, \mathcal{V})$ can be defined as Witness Hiding for $\{D_n\}_{n\in\mathbb{N}}$ if for every function w such that $w(x) \in R(x)$, PPT V^*, $z \in \{0,1\}^*$, and $n \in \mathbb{N}$

$$Pr[x \in D_n : Out_{V^*}[P(x,w) \Leftrightarrow V^*(x,z)]w(x)] \leq \epsilon(n)$$

$(\mathcal{P}, \mathcal{V})$ is Witness Hiding for R_L if it is Witness Hiding withr espect to all hard ensemblses for R_L . Basically, this definition states that conversation with the prover preserves the hardness of the witness it is trying to protect.

Comparison of Witness Indistinguishability and Witness Hiding

One might wonder the strength of the definition of Witness Indistinguishability in comparison with Witness Hiding. In the end, Witness Indistinguishability and Witness Hiding are two incomparable definitions. This can be seen as follows:

1. Witness Indistinguishability is more strict than Witness Hiding, because even though a Witness Hiding proof is not able to reveal its witness, it is allowed to reveal a significant portion of its witness which would violate Witness Indistinguishability.

2. Witness Hiding is more strict than Witness Indistinguishability in the case where every x has only a single witness. In this case, all Interactive Proofs are vacuously Witness Indistinguishable, but a Witness Hiding proof must still hide its witness.

Witness Hiding has applications in public key security, where the generator of the public key would like to convince the sender that as the receiver of the message, he/she has a secret key to decrypt the message. Here, the secret key would serve as the witness which must be proven by the receiver, but hidden from the sender and everyone in between.

A Witness Indistinguishable Proof which is also Witness Hiding

Define ensemble $\{D_n\}_{n\in\mathbb{N}}$ as follows, where f is a OWF:

$$D_n = \{x_1, x_2 \leftarrow \{0,1\}^n : f(x_1), f(x_2)\}$$

$$R_L(y_1, y_2) = \{x | f(x) = y_1 \vee f(x) = y_2\}$$

Basically, $\{D_n\}_{n\in\mathbb{N}}$ is defined as an ensemble, where its witness relation is defined as the inverse of one of the two outputs for f. For this relation, a Witness Indistinguishable Proof is also Witness Hiding. Assume for contradiction that there exists a Verifier, $\mathcal{A}$ which breaks Witness Hiding for this ensemble with non-negligible probability. One can construct a machien $\mathcal{A}$', which inverts f with non-negligible probability.

1. Given an input, y, to invert, $\mathcal{A}$' calculates a $y' = f(r)$ for a randomly generated r.
2. Afterwards, it conducts an Interactive Proof with $\mathcal{A}$ using $x = (y_0, y_1)$, where $y = y_b$ and $y' = y_{1-b}$ for some randomly generated bit b, and $w = (r)$
3. $\mathcal{A}$' then outputs whatever $\mathcal{A}$ outputs at the end of the conversation.

This proof works mainly because $\mathcal{A}$ is not able to distinguish between which spot $\mathcal{A}$' put $f(r)$ into due to Witness Indistinguishability. Thus, it will always invert y with $\frac{1}{2}$ chance.