COM S 6830 – Cryptography Nov 3, 2009

Lecture 19: Witness-Hiding Protocols and MACs

Instructor: Rafael Pass *Scribe: Gabriel Bender*

1 Witness Indistinguishability

Definition 1 *Let (P, V) be an interactive proof with an efficient prover for language $\mathcal{L}$ and witness relation $\mathcal{R}_{\mathcal{L}}$. Then (P, V) is* **witness-indistinguishable** *if for every PPT V^* and nuPPT P there exists a negligible function ϵ s.t. for all $x \in \mathcal{L}, w_1, w_2 \in \mathcal{R}_{\mathcal{L}}(x)$ and $z \in \{0,1\}^*$, $\mathcal{D}$ distinguishes the following distributions with probability at most $\epsilon(|x|)$:*

$$\{Output_{V^*}[P(x, w_1) \leftrightarrow V^*(x, z)]\}$$
$$\{Output_{V^*}[P(x, w_2) \leftrightarrow V^*(x, z)]\}$$

We observe that this is closed under parallel composition, whereas zero knowledge is closed under sequential but not parallel composition. In fact, witness indistinguishability is equivalent to zero knowledge if the simulator in the definition of zero knowledge is allowed to run in exponential time.

Here is an example where witness indistinguishability is not useful: Suppose f is a one-way permutation, and we set $\mathcal{L}(x) = \{y : (\exists x)(y = f(x)\}$ and $\mathcal{R}_{\mathcal{L}}(y) = \{x : f(y) = f(x)\}$. Consider a prover which, on input $f(x) \in \mathcal{L}$, outputs the witness x. Intuitively, the prover doesn't "hide" anything, but witness-indistinguishability is satisfied vacuously because there is a unique witness for each element of our language.

Definition 2 *Suppose (P, V) is an interactive protocol with an efficient prover for $\mathcal{L}$ and $\mathcal{R}_{\mathcal{L}}$. Then (P, V) is* **witness hiding** *for $\{D_n\}_{n \in \mathbb{N}}$ if for every function $w : \mathcal{L} \to \mathcal{L}$ s.t. $(\forall x)(w(x) \in \mathcal{R}_{\mathcal{L}})$, and every PPT V^*, $n \in \mathbb{N}$, and $z \in \{0,1\}^*$ we have*

$$Pr[x \leftarrow \mathcal{D}_n : Output_{V^*}[P(x, w(x)) \leftrightarrow V^*(x, z)] \in \mathcal{R}_{\mathcal{L}}(x)] \leq \epsilon(n)$$

Definition 3 *An ensemble $\mathcal{D}_n$ is* **hard** *for $\mathcal{R}_{\mathcal{L}}$ if for every nuPPT $\mathcal{A}$ there exists a negligible function ϵ s.t. for all n,*

$$Pr[x \leftarrow \mathcal{D}_n : \mathcal{A}(x) \in \mathcal{R}_{\mathcal{L}}(x)] \leq \epsilon(n)$$

Let $\mathcal{D}_n = \{x \leftarrow \{0,1\}^n : f(x)\}$ and $\mathcal{R}_{\mathcal{L}}(x) = \{w : f(w) = x\}$, with f a one-way function. Then $\mathcal{D}$ is a hard ensemble.

It is an open question whether we can find a protocol that is witness hiding but not zero knowledge.

Exercise: Zero-knowledge implies witness hiding.

Proposition 1 *Let f be a one-way function, $\mathcal{D}_n = \{x_1 \leftarrow \{0,1\}^n, x_2 \leftarrow \{0,1\}^n : f(x_1), f(x_2)\}$ and $\mathcal{R}_\mathcal{L}(y_1, y_2) = \{x : f(x) = y_1 \vee f(x) = y_2\}$. Then $\mathcal{D}_n$ is hard.*

Proof. Suppose not. Then there exists a nuPPT $\mathcal{A}$ s.t. for every negligible function ϵ there exist infinitely many n s.t.

$$Pr[(y_1, y_2) \leftarrow \mathcal{D}_n; x \leftarrow \mathcal{A}(D_n) : f(x) = y_1 \vee f(x) = y_2] > \epsilon(n)$$

We claim that we can invert f with non-negligible probability. To do so, we construct a machine $\mathcal{A}'$, which, on input $y \in \{0,1\}^n$, computes

$$\mathcal{A}'(y) = x' \leftarrow \{0,1\}^n; b \leftarrow \{0,1\} : \begin{cases} \mathcal{A}(y, f(x')), & \text{if } b = 0 \\ \mathcal{A}(f(x'), y), & \text{if } b = 1 \end{cases}$$

First, observe that the input distribution that $\mathcal{A}'$ gives to $\mathcal{A}$ is precisely the one used in the definition of hardness. Whenever $\mathcal{A}$ succeeds, it will output either x' or some x s.t. $f(x) = y$. Since b is chosen at random, the probability that $\mathcal{A}$ outputs x is precisely 1/2. Hence, there are infinitely many n for which $\mathcal{A}'$ inverts f is greater than $\epsilon(x)/2$ for any negligible function ϵ. This is equivalent to saying that f is not a one-way function; contradiction. ■

Proposition 2 *If (P, V) is witness indistinguishable for ${\mathcal{R}_\mathcal{L}}^{2w}$ then (P, V) is also witness hiding for $\{\mathcal{D}_n^{2w}\}_{n \in \mathbb{N}}$.*

Proof. Suppose we can break witness hiding. We want to show that witness indistinguishability can also be broken. We take as input a value y, and randomly sample $x' \leftarrow \{0,1\}^n$ and $b \leftarrow \{0,1\}$. We set $y_b = y$ and $y_{1-b} = f(x')$. We then run a machine machine that, with non-negligible probability, outputs a witness for the interaction

$$Output_{V^*}[P((y_0, y_1), x') \leftarrow V^*((y_0, y_1), z)]$$

for an appropriately chosen z. There are two witnesses for (y_0, y_1): x', and some value x s.t. $f(x) = y$. By witness-indistinguishability, the difference between the probability that the function's output is x and the probability that the output is x' must be negligible; otherwise, we could guess with non-negligible probability which witness the prover had access to. So we will output x with probability not much smaller than $r/2$ whenever our black box outputs x or x' with probability r. ■

Informally, we say that we have a proof of knowledge if there exists a knowledge extractor which, by simulating an interaction using rewinding, can determine the input given to the prover.

2 Authentication

Definition 4 *(Gen, Tag, Ver) is a* **MAC** *(Message Authentication Code) if*

- *Gen is a PPT: $k \leftarrow Gen(1^n)$*
- *Tag is a PPT: $\sigma \leftarrow Tag_k(m)$*
- *Ver is a deterministic polynomial-time algorithm: $Ver_k(m, \sigma) \in \{0, 1\}$*

Definition 5 (Correctness) *For all $m \in \{0,1\}^n$,*

$$Pr[k \leftarrow Gen(1^n) : Ver_k(m, Tag_k(m)) = 1] = 1$$

Definition 6 (Existential Unforgeability) *We shouldn't be able to sign any message unless we have the public key, even if we have access to message- signature pairs.*

It turns out that this definition is a little too strong, since the adversary can always provide a message he was provided with and the corresponding signature. We therefore use a slightly weaker definition of security:

Definition 7 *A MAC (Gen, Tag, Ver) is* **secure** *if for every nuPPT $\mathcal{A}$ there is a negligible function ϵ s.t. for all $n \in \mathcal{N}$,*

$$\begin{aligned} Pr[k \leftarrow Gen(1^n); m, \sigma \leftarrow \mathcal{A}^{Tag_k(\cdot), Ver_k(\cdot)} : \\ \mathcal{A} \text{ didn't query } Tag_k(m) \ \wedge \ Ver_k(m, \sigma) = 1] \ \leq \ \epsilon(n) \end{aligned}$$

Proposition 3 *Let $\{f_s\}$ be a family of pseudo-random functions. The the following is a MAC scheme:*

- *$Gen(1^n) \leftarrow \{0,1\}^n$*
- *$Tag_k(m) = f_k(m)$*
- *$Verify_k(m, \sigma) = \begin{cases} 1, & \text{if } f_k(m) = \sigma \\ 0, & \text{otherwise} \end{cases}$*

Proof. Correctness is an immediate consequence of the construction, since $f_k(m)$ is well-defined. If g was a truly random function then the probability that we could guess $g(m)$, even given g applied to an arbitrary number of other message, would be exactly $1/2^n$. If $\mathcal{A}$ could guess m with non-negligible probabity for $m_1, \ldots, m_n$ and $f_k(m_1), \ldots, f_k(m_r)$ then given an oracle $\mathcal{O}$ that either computed f_k or g, we could compute $\mathcal{O}(m_1), \ldots, \mathcal{O}(m_r)$ and run $\mathcal{A}$ to predict $\mathcal{O}(m)$. It would succeed with negligible probability for g and non-negligible probability for f_k, and so f_k couldn't be pseudorandom; contradiction. ■