CS 290G (Fall 2014) – Introduction to Cryptography December 4, 2014

Lecture 15: Zero-Knowledge Proof Systems

Instructor: Rachel Lin *Scribe: Tiawna Cayton*

1 Knowledge

We define "knowledge" as any information conveyed in a message or conversation. In a secure encryption, seeing the ciphertext reveals no information about the plaintext. We want to quantify what knowledge Bob gains through interaction with Alice.

From a behavioral view, we say that Bob learns no knowledge if whatever Bob can do after talking with Alice, he can also do before talking with Alice. We say Bob learns only S if whatever he can do after talking with Alice, he can do, given y_1 without talking to Alice.

Eg 1: Alice $\text{—}O^n \rightarrow$ Bob No knowledge. Bob can reproduce O^n for himself.
Eg 2:Alice $r \leftarrow U_l$ $\text{—}r \rightarrow$ Bob No knowledge. Bob can sample r with identical distribution.
Eg 3: Alice$p, g \leftarrow \Pi_n$ $\text{—}N = p \cdot q, p, q \rightarrow$ Bob No knowledge. Bob could have sampled $p, q \leftarrow \Pi_n$ and generated $N = p \cdot q, p, q$
Eg 4:Alice $\leftarrow N\text{—}$ Bob $N \leftarrow U_{2n}$
Alice $\text{—}p$, if $N = p\cdot, \perp o.w.q \rightarrow$ Bob
There is knowledge because Bob could not have presented a factor of N. He couldn't have computed this conversation on his own.

Intuitively, if Bob can "reproduce" a conversation with Alice "efficiently," then this conversation gives no knowledge to Bob.

We say "efficiently" means a uniform PPT computation.

To explain "reproduce" we use the following cases:

Case 1: Alice sends deterministic message to Bob. "Reproduce" means Bob can generate m on his own.

Case 2: Alice sends a randomized message to Bob. "Reproduce" means:

1) Bob can sample from the same distribution
2) Bob can sample from D' that is indistinguishable from D

Case 3: Alice has a conversation with Bob. "Reproduce" means Bob can replicate the conversation, for instance, if Alice is using a PRF, Bob would reproduce an indistinguishable conversation using a RF.

A knowledge based notion of secure encryption means that a ciphertext should convey no knowledge.

Definition Zero knowledge encryption
A private-key encryption scheme (Gen, Enc, Dec) is zero-knowledge if $\exists$ PPT "simulator" S such that $\forall m_n$ the following holds.

$$\{k \leftarrow Gen(1^n), Enc(k, m)\} \approx \{S(1^n)\}$$

Theorem: Any zero-knowledge encryption is a SMS encryption and vice versa.
($\Leftarrow$) $\forall m_0 m_1$,
$\{k \leftarrow Gen(1^n) : Enc(k, m_0)\} \approx \{S(1^n)\}$
$\approx$
$\{k \leftarrow Gen(1^n) : Enc(k, m_1)\} \approx \{S(1^n)\}$

$S(1^n)$:

1) $k \leftarrow Gen(1^n)$

2) $c' \leftarrow Enc(k, 0)$ outputs c

Thus it is zero-knowledge.
($\Rightarrow$) Using trasitivity, if it exists, (and by the arguments in the first half of the proof) it is SMS.

2 Proof System for a Language

Setting: Alice(x)-Prover, Bob(x)-Verifier.
x - math statement, x = N is a product of two primes, $x \in L$, $L = \{x\}$
$L = N : p, qprimes, N = p \cdot q$

NP languages: $L = \{x : \exists w, R_l(x, w) = 1\}$ R_L is efficiently checkable.
Alice wants to convince Bob that $x \in L$
Goal: Bob, through this conversation, learns no knowledge other than x, and the fact that $x \in L$.
Paradoxical task:

1) Prover wants to convince verifier.

2) Prover shows no knowledge to verifier.

3 Zero-Knowledge Proof System

Proof System
$< P, V >$ interactive PPT machine is a zero-knowledge proof system for NP language L.
$P(x, w)$ has conversation with $V(x)_{\to b}$ b-accepts or rejects

Completeness: $\forall\, x \in L, \forall\, w\ s.t.\ R_l(x, w) = 1$

$$Pr[e \leftarrow [P(x, w) \leftrightarrow V(x)] : out_v(e) = 1] = 1$$

Soundness: $\forall\, x \notin L, \forall\, n.u.PPT\ P^*, \exists \epsilon(n)$ neg. s.t.

$$Pr[e \leftarrow [P^* \leftrightarrow V(x)] : out_v(e) = 1] \leq \epsilon(n)$$

Zero-Knowledge
$forall x \in L, \forall n.u.PPTV^*, \exists$ simulator S s.t.

$$\{e \leftarrow [P(x, w) \leftrightarrow V^*(x)] : View_{V^*}(e)\} \approx \{S(x)\}$$

where $View_{V^*}(e)$:x,message from P, random coins of V^*