CS 290G (Fall 2014) – Introduction to Cryptography Nov 20th, 2014

Lecture 12: More on Trapdoor Functions and PKE's

Instructor: Rachel Lin *Scribe: Markus Karl Torfason*

1 Recap

Last class:

1. Theory of PRF from PRG
2. PKE definition. SMS
3. RSA

Today:

1. Formalize Trapdoor functions (permutations)
2. Construction of PKE
   (a) Single Message Security (SMS)
   (b) Multi Message Security (MMS)

2 Trapdoor Functions

Definition - Collection of Trapdoor Functions:
$F = \{f_i : D_i \to R_i\}_{i \in I}$
1) Easy to sample a function. I.e. $\exists$ PPT Gen(1^n) $\to$ (i,t), i$\in$I
2) Easy to sample from a given domain D_i
3) Easy to evaluate f_i
4) Hard to invert. I.e. $\forall$nu PPT A, $\exists$ negligible ϵ s.t.
$Pr[(i,t) \leftarrow Gen(1^n),\ x \leftarrow D_i,\ y = f_i(x),\ x' \leftarrow A(i,y) :\ f_i(x') = y] \leq \epsilon(n)$.
(*) 5) $\exists$ PPT inverter E s.t.
$Pr[(i,t) \leftarrow\ Gen(1^n),\ x \leftarrow D_i,\ y = f_i(x),\ x' \leftarrow E(i,t,y) :\ f(x') = y] = 1$

We notice that except for (*) this is the same as the definition of a collection of OWF's.

Theorem - RSA collection is a collection of TDP's:

A. We know they are permutations

1) $Gen(1^n):\ N = p \times q$ random n-bit primes $e \leftarrow \mathbb{Z}^*_{\phi(N)}$

$i = (N, e)$

$t = d = e^{-1} mod(\phi(N))$, output (i, t)

2) - 4) Same as before when showing RSA is a collection of OWF's

5) $E(i = (N, e),\ t,\ y = x^e mod(N)) = y^t mod(N) = x$

3 Construction of PKE

Attempt Construction of PKE from a collection of TDP $F = \{f_i\}_{i \in I}$

$Gen(1^n)$: Sample $PK = (N, e)$ as in RSA collection

$sk = (d, N)$

$((N, e, d)$ sampled as in RSA collection)

$Enc(pk, x) = x^e mod(N)$

$Dec(sk, c) = c^d mod(N)$

When we assume $D_i = R_i = \{0,1\}^n, Gen_F, E_F$ our construction attempt becomes:

$Gen(1^n)$: Sample $Gen_F \rightarrow (i, t)$

$pk = i$

$sk = (t, i)$

$Enc(pk, m):\ c = f_i(m)$

$Dec(sk = (i, t), c)$: Compute $m = f_l^{-1}(c)$ by $E_F(i, t, c)$

Step 1: Constructing a bit Encryption Scheme π

$Enc(pk, b) : r \leftarrow U_n$

$z = f_i(r)$

$y = h(r) \oplus b$

c = z, y

$Dec(sk = t, c)$

$c = z,\ y$

$r = f_i^{-1}(z)$

$b = y \oplus h(r)$

Definition: We say a predicate h is a hard-core predicate (HCP) for a collection of TDP's F

if:

1) h is efficient to evaluate

2) Hard to Predict $\forall nu\ PPT\ A\ \exists\, negligible\ \epsilon\ s.t.$

$Pr[i,\ t) \leftarrow Gen(1^n),\ x \leftarrow U_n,\ b \leftarrow A(f_i(x)):\ b = h(r)] \leq \epsilon(n)$

Theorem 1: π is SMS
Proof:
Let $G_i(r) = f_i(r)||h(r)$
$\{(pk, sk) \leftarrow Gen(1^n) : \ pk, \ Enc(pk, 0)\}$
$\approx$
$(pk, sk) \leftarrow Gen(1^n) : \ pk, \ Enc(pk, 1)\}$
$\Leftrightarrow$
$\{(i, t) \leftarrow Gen(1^n), \ r \leftarrow U_n : \ i, \ G_i(r) \oplus (0^n||0)\}$
$\approx$
$\{(i, t) \leftarrow Gen(1^n), \ r \leftarrow U_n : \ i, \ f_i(r), \ h(r) \oplus 1\}$

Claim 1:
$\{(i, t) \leftarrow Gen_F(1^n) : \ i, \ G_i(U_n)\}$
$\approx$
$\{(i, t) \leftarrow Gen_F(1^n) : \ i, \ U_{n+1}\}$
Proof:
Same as the proof for PRG using 1-bit expansion.

Applying Claim 1:
$\{D_1\} = \{(i, t) \leftarrow Gen_F, \ r' \leftarrow U_{n+1} : \ r' \oplus (0^n||0)\}$
$\approx$
$\{D_2\} = \{(i, t) \leftarrow Gen_F, \ r' \leftarrow U_{n+1} : \ r' \oplus (0^n||1)\}$

$\{D_1\} \approx \{D_2\}, \ \{D_1\} \approx \{H_1\}, \ \{D_2\} \approx \{H_2\}$
$\Rightarrow \{H_1\} \approx \{H_2\}$ by transitivity.

Step 2:

Bit encryption		String Enc
	$\Rightarrow$	
SMS		MMS

Definition of Multi Message Security (MMS) for bit encryption:

We say π is MMS if $\forall \, poly \, q, \ \forall \, \{\vec{m}_0\}, \ \{\vec{m}_1\}$
$\{(pk, sk) \leftarrow Gen(1^n) : pk, Enc(pk, m_0[1]) \ ... \ Enc(pk, m_0[q])\} = \{H_0\}$
$\approx$
$\{(pk, sk) \leftarrow Gen(1^n) : pk, Enc(pk, m_1[1]) \ ... \ Enc(pk, m_1[q])\} = \{H_q\}$

$\vec{m}_b = m_b[1]...m_b[q]$, where $\forall \, i, \ m_b[i]$ are single bits.

Theorem: Any π that is SMS is also MMS
Proof for bit Encryption:
Given π is SMS:
$\{D_1\} = \{(pk, sk) \leftarrow Gen: \ pk, \ Enc(pk, 0)\}$
$\approx$
$\{D_2\} = \{(pk, sk) \leftarrow Gen: \ pk, \ Enc(pk, 1)\}$

We want to prove the left
Fix $\{\vec{m}_0\}, \{\vec{m}_1\}$.

$\{H_1\} = \{(pk, sk) \leftarrow Gen: \ pk, \ Enc(pk, m_1[1]), \ Enc(pk, m_0[2] \ ... \ Enc(pk, m_0[q])\}$

Claim 1: $\{H_0\} = \{H_1\}$
Proof by SMS of π

$\{D_1\} = \{(pk, sk) \leftarrow Gen: \ pk, \ Enc(pk, m_0[1])\}$
$\approx$
$\{D_2\} = \{(pk, sk) \leftarrow Gen: \ pk, \ Enc(pk, m_1[1])\}$

Consider M(pk,c):
$c_2, ..., \ c_q = Enc(pk, m_0[2]) \ ... \ Enc(pk, m_0[q])$.
Output pk, $c_1 = c, \ c_2, \ ..., c_q$

$\{M(D_1)\} = \{H_0\}$
$\{M(D_2)\} = \{H_1\}$

By closure under efficient operation $\Rightarrow \{H_0\} \approx \{H_1\}$

$\{H_i\} = \{(pk, sk) \leftarrow Gen : pk, Enc(pk, m_1[1]...Enc(pk, m_1[i], Enc(pk, m_0[i{+}1]...Enc(pk, m_0[q])\}$

Claim 2: $\forall i \ \{H_i\} \approx \{H_{i+1}\}$

$H_i: \ pk, \ m_1[1] \quad m_1[i] \quad m_0[i+1] \quad ... \quad m_0[q]$

$H_{i+1}: \quad m_1[i+1] \quad m_0[i+2] \quad ... \quad m_0[q]$

$\{D_1\} = \{pk, \ Enc_{pk}(m_0[i+1])\}$
$\{D_2\} = \{pk, \ Enc_{pk}(m_1[i+1])\}$

Same proof as Claim 1
Intuition (False): By transitivity $\{H_0\} \approx \{H_q\}$
Formally prove by contraposition and use Strong Hybrid Lemma.