Criptografía y Seguridad Computacional **2016-01**

Tarea 2: 3/04/2016

Entrega: 15/03/2016

Reglas

Las tareas se pueden discutir de a 2 personas. Sin embargo, cada estudiante tiene la obligación de escribir su propia solución. La entrega de la tarea se realiza mediante correo electrónico a `fkrell@gmail.com` con el asunto `[cripto] Entrega Tarea2` a más tardar el día Viernes 15 de Abril de 2016 a las 6:00 pm. Cada estudiante sólo puede asumir que su tarea ha sido correctamente entregada si recibe un confirmación del profesor. Usted debe entregar un archivo formato PDF generado en LaTeX. Para su conveniencia puede usar el template en la página web del curso.

Problema 1 (10pts)

Implemente una función pseudo-aleatoria con llaves de largo n^2 bits, entrada de largo $\log n$ bits y output de largo n bits. Muestre que la ventaja de cualquier algoritmo distinguidor para su función es 0.

Problema 2 (10pts)

Sea F un cifrador de bloque (F es una permutación pseudo-aleatoria) con llaves de largo n bits (ejemplo $n = 64$) para bloques de largo n. Dado la actual velocidad de los procesadores y la capacidad de almacenamiento, la comunidad criptográfica ha recomendado utilizar cifradores de bloque con llaves de largo $2n$ bits. Un "experto" en seguridad a recomendado usar el siguiente cifrador de bloque F' basado en F: Obtener 2 llaves k_1, k_2 independientes para F, obteniendo una llave final $k = \langle k_1, k_2 \rangle$ de $2n$ bits y computar el cifrador como $F'_{k_1,k_2}(x) = F_{k_1}(F_{k_2}(x))$.

Explique por qué el esquema propuesto por el "experto" no provee la seguridad deseada. Es decir, muestre un algoritmo distinguidor (no necesariamente de tiempo polinomial) para F que requiera tiempo significativamente menor a 2^{2n}.

Problema 3: seguridad CPA (10pts)

Sea G un PRG y F una PRF. Determine si los siguientes esquemas de cifrado son CPA-seguros. Justifique su respuesta formalmente.

- (3pts) $\mathsf{Enc}(k, m)$: output $\langle r, G(r) \oplus m \rangle$, para r uniforme en $\{0,1\}^\lambda$.
- (4pts) $\mathsf{Enc}(k, m)$: output $m \oplus F_k(0^n)$.
- (4pts) $\mathsf{Enc}(k, m) : m \in \{0,1\}^{2n}$, $m = m_1 || m_2$, $|m_1| = |m_2| = n$. output $\langle r, F(r) \oplus m_1, F(r+1) \oplus m_2 \rangle$.

Problema 4: naive HMAC (10pts)

Sea $H^s : \{0,1\}^* \to \{0,1\}^n$ una función de hash resistente a colisiones.

1. a) Demuestre que el siguiente código de autenticación de mensajes NO es necesariamente seguro.

   - $\mathsf{Gen}(1^\lambda)$: Output $k \sim U_\lambda$.
   - $\mathsf{Mac}_k(m)$: Output $t = H^s(k||m)$.
   - $\mathsf{Vrfy}_k(m, t)$: Output 1 si y sólo si $t = H^s(k||m)$.

   Hint: Piense que H esta construida usando la transformación de Merkle-Damgärd sobre una función $h^s : \{0,1\}^{n'} \to \{0,1\}^n$ resistente a colisiones.

2. b) Un oráculo aleatorio $O(\cdot)$ es una función pública (todos los participantes, incluyendo al adversario tienen acceso a ella) que puede ser consultada como "caja negra" y que tiene la propiedad de que para cada consulta nueva x, el valor $O(x)$ es uniformemente aleatorio. Demuestre que si asumimos que H es un oráculo aleatorio, entonces la construcción anterior sí es un código de auntenticación de mensajes seguro.

Problema 5: Definiciones de Funciones Hash (10pts)

Demuestre que cualquier función de hash $H^s : \{0,1\}^m \to \{0,1\}^n$ resistente a colisiones, es resistente a segunda preimagen, y que cualquier función resistente a segunda preimagen es resiste a preimagen. Es decir, si para H^s es infactible encontrar $x_1 \neq x_2$ tales que $H^s(x_1) = H^s(x_2)$, entonces

dado x uniforme en $\{0,1\}^m$ es infactible también encontrar $x' \neq x$ tal que $H^s(x) = H^s(x')$. A la vez, si dado x uniforme en $\{0,1\}^m$ es infactible encontrar $x' \neq x$ tal que $H^s(x) = H^s(x')$, entonces si y es uniforme en $\{0,1\}^n$ también es infactible encontrar x tal que $H^s(x) = y$.

Problema 6: Árboles de Merkle (10pts)

1. Describa el escenario de interés visto en clases para árboles de Merkle.
2. De una definición formal de seguridad para este escenario.
3. Describa formalmente la construcción vista en clases.
4. Demuestre su seguridad.

Problema (Opcional)

Implemente el ataque del problema 2 sobre algún cifrador de bloque (reduciendo su espacio de llaves). ¿Hasta que largo de llaves pudo realizar su ataque?