MA-53G Intr. a la Criptografía	**Semestre Otoño, 1998**

Examen: 9 de Julio

Profesor: Marcos Kiwi *Auxiliar: Alejandro Hevia*

Tiempo: 3:30 horas

(i).- (0.5 pts) Compresión es usualmente utilizado para almacenar y transmitir datos. Suponga que se desea utilizar compresión junto con encripción. Que es preferible, comprimir y luego encriptar, o encriptar y luego comprimir.

Justifique dando al menos tres razones.

(ii).- (0.5 pts) Considere un sistema de firmas digitales para el cual se sabe que obtener la función de firma a partir de la función de verificación es computacionalmente equivalente a factorizar. ¿Es dicho sistema de firmas computacionalmente seguro? Justifique.

(iii).- (1.0 pts) La funciones hash también pueden ser utilizadas para crear *estampas digitales de tiempo*, i.e., para probar que un documento $D \in \Sigma^*$ existió un determinado día. Una alternativa es publicar D en el Mercurio, pero esto es caro. Mejor es publicar $H(D)$, donde $H\colon \Sigma^* \to \{0,1\}^l$ es una buena función hash. Sin embargo, si uno tiene un conjunto de documentos $D_1, \ldots, D_n \in \Sigma^*$ los costos de publicación suben linealmente en n. Alternativamente, uno puede publicar $H(H(D_1)\circ\cdots\circ H(D_n))$ lo que hace los costos de publicación independientes de n, pero la demostración de que D es uno de los documentos $D_1, \ldots, D_n$ requiere conocer $H(D_i)$, para todo $i \in \{1, \ldots, n\}$, lo que es engorroso especialmente si n es grande (digamos 10^6).

De un esquema en que el tamaño de la publicación es independiente de n, del tamaño de los documentos, y que la demostración de que D es uno de los documentos $D_1, \ldots, D_n$ crece de acuerdo a $\log n$.

(iv).- (1.0 pts) Demuestre que es una mala idea elegir primos de la forma $p = 2^k + 1$ para protocolos basados en la conjetura de intractabilidad del problema del logaritmo discreto, en particular, pruebe que para dicho tipo de primos, el problema del logaritmo discreto puede ser resuelto eficientemente. Especifique porque su argumento falla para el caso de p primo arbitrario.

<u>Indicación</u>: Sea g un generador de $\mathbb{Z}_p^*$. Pruebe que conocido $y = g^x \bmod p$ se puede calcular el bit menos significativo de $x \in \mathbb{Z}_{p-1}$ y que conocido dicho bit es posible calcular eficientemente el siguiente bit menos significativo de x. Generalize.

(v).- (1.0 pts) Considere el siguiente sistema de firmas digitales. Sea p un primo gande y g un generador de $\mathbb{Z}_p^*$. La clave privada de firmas es $x \in \{1, \ldots, p-1\}$ y la clave pública de verificación es p e $y = g^x \bmod p$. Sea además $H\colon \Sigma^* \to \mathbb{Z}_p^*$ una función hash. Para firmar el mensaje $m \in \Sigma^*$, se calcula $h \leftarrow H(m)$, y $z \leftarrow xh^{-1} \bmod p-1$ (si h no posee un inverso módulo $p-1$, entonces se calcula $h \leftarrow H(m \circ h)$ hasta que se obtiene un valor h que posee un inverso módulo $p-1$). La firma de m es $s \leftarrow g^z \bmod p$.

Proponga un mecanismo de verificación eficiente para el sistema de firmas descrito (justifique todas

sus afirmaciones).
Muestre que el sistema propuesto no es seguro (justifique su respuesta y especifique tanto el tipo de ataque como el tipo de falsificación ante el cual el sistema es vulnerable).

(vi).- (1.0 pts) Sean $\{E_k\}_{k\in\mathcal{K}}$ una familia de funciones de encripción donde cada E_k es una permutación de $\{0,1\}^{|k|}$. Para cada $k \in \mathcal{K}$ se define la función de encripción probabilista que a cada $M \in \{0,1\}^{|k|}$ le asocia

$$F_k(M) \quad \stackrel{\text{def}}{=} \quad (E_k(R), R \oplus M)\,,$$

donde R es una secuencia de bits aleatorios del mismo tamaño del mensaje M. Para cada nueva encripción de un mensaje se elije una nueva secuencia R de bits aleatorios.

Se dice que un ataque es *de reconstrucción de clave a texto puro elegido* si al adversario le es permitido fijar un $n \in \mathbb{N}$, generar una secuencia de mensajes $M_1, M_2, \ldots, M_n$, y para cada M_i obtener el correspondiente texto cifrado.

Se dice dice que un ataque es *de reconstrucción de clave a texto puro aleatorio* si al adversario le es permitido fijar un $n \in \mathbb{N}$, y obtener pares de texto puro/cifrado donde el texto puro es elegido al azar.

Pruebe que si el criptosistema basado en las funciones de encripción probabilista $\{F_k\}_{k\in\mathcal{K}}$ es vulnerable ante un ataque de reconstrucción de clave a texto puro elegido entonces también lo es ante un ataque de reconstrucción de clave a texto puro aleatorio.

(vii).- (1.0 pts) Suponga que dado p se puede encontrar un divisor q de $p-1$ tal que $q > \sqrt{p}$. Pruebe que entonces el siguiente algoritmo puede usarse para establecer la primalidad de p.

- Determinar un primo q divisor de $p-1$ tal que $q > \sqrt{p}$
- Escoger a al azar en $\mathbb{Z}_p^*$
- Si $1 < \operatorname{mcd}(a-1, p) < p$ entonces p es compuesto.
- Verificar que $a^q \equiv_p 1$.
- Recursivamente probar que q es primo.

Observe que basta probar el siguiente

Lema 1 *Si $q > \sqrt{p}$ es un primo y para algún $a \in \mathbb{Z}_p^*$, $\operatorname{mcd}(a-1,p) = 1$ y $a^q \equiv_p 1$, entonces p es primo.*

<u>Indicación</u>: Asuma que existe un $d \leq \sqrt{p}$ divisor de p y estudie el orden de a en $\mathbb{Z}_d^*$.