Criptografía y Seguridad Computacional **2016-01**

Tarea 5: 18/05/2016

Entrega: 1/06/2016

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`, y con copia a `jsnavar1@uc.cl`, con el asunto `[cripto] Entrega Tarea5` a más tardar el día Miercoles 1 de Junio de 2016 a las 23:59. Cada estudiante sólo puede asumir que su tarea ha sido correctamente entregada si recibe confirmación del profesor o del ayudante. Usted debe entregar un archivo formato PDF generado en LaTeX. Para su conveniencia puede usar el template en la página web del curso.

Si utiliza referencias, es obligación agregarlas al final de su solución.

Problema 1 (10pts)

Demuestre como falsificar firmas en el esquena de firmas de Shnorr si eliminamos el string aleatorio $r = g^k$ usado en el hash sobre el mensaje.

Problema 2 (15pts)

Considere la siguiente variación del esquema de firmas RSA plano ($\mathsf{Sign}_{\langle d,N\rangle}(m) = m^d \mod N$) visto en clases: El algoritmo generador de llaves es idéntico al RSA plano ($pk = N, e$, $sk = N, d$, en donde $e \cdot d = 1 \mod \phi(N)$

Sea $n = ||N||$ (largo en bits del módulo $N = pq$). El esquema esta definido para mensajes de largo $9n/10 - 1$. Antes de firmar, el mensaje es cambiado a $\hat{m} = (0||m||0^{n/10})^d \mod N$, y la firma esta definida como $\sigma = \hat{m}^d \mod N$.

Luego verificamos comparando $\sigma^e \mod N$ con $(0||m||0^{n/10}) \mod N$.

1. Demuestre que el ataque visto en clases para RSA plano (elegir primero una firma arbitraria σ y luego computar el mensaje como $m = \sigma^e$) no funciona en este nuevo esquema.

2. Demuestre que el esquema sigue siendo inseguro.

Problema 3 (15pts)

Sea $\Pi = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ un esquema de firmas digitales seguro según la definición vista en clases (es decir, el adversario es incapaz de generar una firma válida para un mensaje, elegido por este, que no haya sido consultado a su oráculo de firmas). Considere el esquema de firmas digitales $\Pi_y = (\mathsf{Gen}, \mathsf{Sign}_y, \mathsf{Vrfy})$ tal que si $H(0) \neq y$ entonces $\mathsf{Sign}_{y_{sk}}(m) = \mathsf{Sign}_{sk}(m)$. En cambio, si $H(0) = y$, entonces $\mathsf{Sign}_{y_{sk}}(m) = sk||\mathsf{Sign}_{sk}(m)$.

- Demuestre que si H es un oráculo aleatorio, entonces Π_y es seguro para *cualquier* y.
- Demuestre que existe un y tal que Π_y no es seguro si H es instanciado con una función de hash fija (sin llave), como por ejemplo SHA-1.

Problema 4 (20pts, OBLIGATORIO)

1. Describa el conjunto y la operación binaria que definen un grupo de curva elíptica.
2. ¿Cómo se computa la operación sobre 2 elementos iguales?
3. ¿Cuál es el elemento neutro para la operación anterior?
4. ¿Nombre al menos 2 ventajas y una desventaja de utilizar grupos en curvas elipticas en comparación con los grupos vistos en clases, por ejemplo, el grupo de residuos cuadrados módulo $p = 2q + 1$ (p y q primos)?
5. ¿Cuál es la razón de utilizar una función de hash (SHA-1) para generar los parámetros de la curva P-256?
6. ¿Por qué el uso de una función de hash para generar los parámetros no es suficiente para convencer a la comunidad criptográfica de que la curva P-256 no fue generada maliciosamente?