# Criptografía

Segundo Cuatrimestre de 2007

Departamento de Computación Facultad de Ciencias Exactas y Naturales Universidad de Buenos Aires

# Trabajo Práctico

Truco Mental

Integrante	LU	Correo electrónico
Albanesi, Matías	??/??	@
Carbajo, Pablo	717/04	pcarbajo@dc.uba.ar
Freijo, Diego	4/05	giga.freijo@gmail.com
Venturini, Maura	??/??	@

#### Abstracto

En el presente trabajo se diseño e implementó un protocolo para jugar al truco mediante el cual se asegura el cumplimiento de las reglas de juego.

### Palabras Clave

Encriptación, algoritmos de clave pública/privada y simétricos, RSA, AES, Mental Truco

# ${\rm \acute{I}ndice}$

1.	Logi	ca de juego	3				
2.	El j	ego del truco	4				
	2.1.	Reglas de juego	4				
	2.2.	Logica de juego	4				
3. El protocolo							
	3.1. El reparto						
		3.1.1. Protocolo de reparto de cartas	5				
		3.1.2. Pre-inicio del juego	7				
	3.2.	Transcurso del juego	8				
4.	Con	ideraciones	9				

# 1. Logica de juego

- 2. El juego del truco
- 2.1. Reglas de juego
- 2.2. Logica de juego

# 3. El protocolo

## 3.1. El reparto

#### 3.1.1. Protocolo de reparto de cartas

El siguiente es el protocolo que da como resultado tres cartas para cada jugador elegidas de forma azaroza.

1. B le pide conexión a A

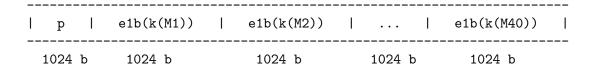
2. A genera una clave simétrica k (por ejemplo, con AES), encripta las 40 cartas  $M_1...M_40$  con k y las envía a B (k sirve para asegurar que el mazo enviado por A en este paso es válido).

3. B genera un p primo grande y genera  $e_b^1$ ,  $d_b^1$  (utilizadas para asegurar una repartición justa) y  $e_b^2$ ,  $d_b^2$  (utilizadas para asegurar que se jueguen las cartas tocadas) tal que

$$e_b^1 * d_b^1 = 1[mod \ p - 1] = e_b^2 * d_b^2$$

Envía a A el p y las cartas ya encriptadas con k encriptadas a su vez con  $e_b^1$  (usando RSA)

$$e_b^1(k(M_i)) = k(M_i)^{e_b^1}$$



4. A usa p para generarse sus propias claves

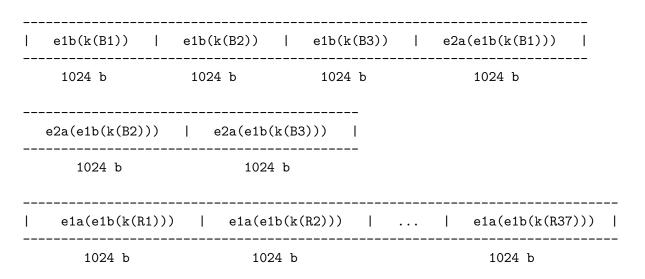
$$e_a^1*d_a^1 = 1[mod\ p-1] = e_a^2*d_a^2$$

Elige al azar 3 cartas de las enviadas por B  $(B_1, B_2, B_3)$  y las firma con  $e_a^2$ . Envía cada carta como una tupla

$$< e_b^1(k(Bi)), e_a^2(e_b^1(k(Bi))) > = < k(Bi)^{e_b^1}[mod\ p], k(Bi)^{(}e_b^1*e_a^2)[mod\ p] >$$

A su vez, repite el paso anterior realizado por B enviandole a este el resto de las cartas (Ri) encriptadas con su clave:

$$e_a^1(e_b^1(k(Ri))) = k(Ri)^{(e_b^1 * e_a^1)}[mod \ p]$$



5. B<br/> recibe sus cartas (las tuplas) y les aplica la desencripcion de <br/>  $e_b^1$  con  $d_b^1\colon$ 

$$< d_b^1(k(B_i)^{e_b^1}[mod\ p]), d_b^1(k(B_i)^{e_b^1*e_a^2}[mod\ p]) > =$$

$$< k(B_i)^{e_b^1*d_b^1}[mod\ p], k(B_i)^{e_b^1*e_a^2*d_b^1}[mod\ p] > =$$

$$< k(B_i), k(B_i)^{e_a^2}[mod\ p] >$$

A su vez, elige 3 cartas al azar  $(A_1, A_2, A_3)$  del resto (las  $R_i$ ) y les aplica tambien  $d_b^1$ :

$$d_b^1(k(A_i)^{e_b^1*e_a^1}[mod\ p]) = k(A_i)^{e_b^1*e_a^1*d_b^1}[mod\ p] = k(A_i)^{e_a^1}[mod\ p]$$

Para completar la mano de A, debe completar las tuplas con su firma:

$$e_b^2(k(A_i)^{e_a^1}[mod \ p]) = k(A_i)^{e_a^1 * e_b^2}[mod \ p]$$

y se las envía a A:

$$< k(A_i)^{e_a^1} [mod \ p], k(A_i)^{e_a^1 * e_b^2} [mod \ p] >$$

_							
I	e1a(k(A1))	e1a(k(A2))	I	e1a(k(A3))	I	e2b(e1a(k(A1)))	I
_	1024 b	1024 b		1024 b		1024 b	
e2b(e1a(k(A2)))		   e2b(e1a	e2b(e1a(k(A3)))				
_	1024 b	102	 4 b				

6. A recibe sus cartas y les aplica la desencripción de  $e_a^1$  con  $d_a^1$ :

$$< d_a^1(k(A_i)^{e_a^1}[mod\ p]), d_a^1(k(A_i)^{e_a^1*e_b^2}[mod\ p])> =$$
 
$$< k(A_i)^{e_a^1*d_a^1}[mod\ p], k(A_i)^{e_a^1*e_b^2*d_a^1}[mod\ p]> =$$
 
$$< k(A_i), k(A_i)^{e_b^2}[mod\ p]>$$

A utiliza k para ver las cartas que le tocaron, su mano queda

$$< A_i, k(A_i)^{e_b^2} [mod \ p] >$$

Por ultimo, envia k a B

7. B recibe k, con lo que la utiliza para ver los  $M_i$  que le mando A en el paso 2 (poseia  $k(M_i)$ ) y verificar que le mandó un mazo valido. También desencripta su mano para ver las cartas que le tocaron:

$$\langle B_i, k(B_i)^{e_a^2} [mod \ p] \rangle$$

### 3.1.2. Pre-inicio del juego

Este es final del inicio del juego. Se setean datos para ser usados durante el resto de la mano.

1. Ahora que ambos tienen sus manos, B se genera un par de claves RSA comunes y corrientes

$$(e_b^3, n_b), (d_b^3, n_b)$$

y envia la publica  $(e_b^3, n_b)$  a A. Usará  $d_b^3$  sólo para firmar sus acciones.

A su vez, genera un paquete con el timestamp inicial decretando esta como la hora de inicio de juego. Lo envía firmado con  $d_b^3$ .

2. A lee con la clave pública de B lo firmado por él y verifica que el timestamp es válido. Se genera también sus pares de claves RSA

$$(e_a^3, n_a), (d_a^3, n_a)$$

y envía su aceptación (firmada con  $d_a^3$ ) de la hora de inicio.

3. Al recibir B la verificación, puede empezar a jugar (notar que es mano).

### 3.2. Transcurso del juego

Durante el transcurso del juego se deberán cumplir las siguientes reglas:

- Es mano el que pidió conexion.
- Supongo que B quiere jugar  $B_2$ . Entonces debe enviar la firma que posee:

$$k(B_2)^{e_a^2}[mod\ p]$$

Cuando A lo recibe, le aplica su desencripción:

$$d_a^2(k(B_2)^{e_a^2}[mod\ p]) = k(B_2)^{e_a^2 * d_a^2}[mod\ p] = k(B_2)$$

Y con aplicar k, obtiene  $B_2$ .

- Cada vez que se canta o se juega una carta, se debe adjuntar un timestamp del momento del canto/juego. Luego, el paquete debe ser firmado (con  $d_a^3 / d_b^3$  según corresponda) para evitar el repudio del emisor mas adelante ("...no no, yo no te cante, entendiste mal...") y la reutilización del canto como prueba falsa más adelante ("...si si, vos cantaste truco; hace 5 horas, pero cantaste, mira...").
- El que finalize el juego (el que se vaya al mazo, el que mate la ultima carta del contrincante, el que cante 33 en el falta envido) debe además enviar un el timestamp firmado, marcándolo como el final oficial del juego. El oponente, aunque esté caliente, debe confirmarle si la hora es válida.

# 4. Consideraciones