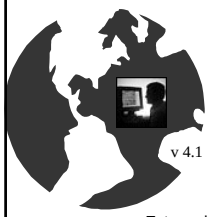
# Capítulo 19
# Protocolos y Esquemas Criptográficos

## Seguridad Informática y Criptografía




Material Docente de
Libre Distribución

Ultima actualización del archivo: 01/03/06
Este archivo tiene: 73 diapositivas

Dr. Jorge Ramió Aguirre
Universidad Politécnica de Madrid



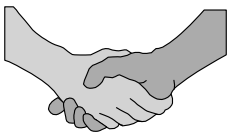




## Definición de protocolo criptográfico

- Protocolo: es el conjunto de acciones coordinadas que realizan dos o más partes o entidades con el objeto de llevar a cabo un intercambio de datos o información.
- Protocolos criptográficos serán aquellos que cumplen esta función usando para ello algoritmos y métodos criptográficos.
- Permiten dar una solución a distintos problemas de la vida real, especialmente en aquellos en donde puede existir un grado de desconfianza entre las partes.

¿Qué es un protocolo?



*Veamos 10 ejemplos*

http://www.criptored.upm.es/guiateoria/gt_m023c.htm

# Ejemplos de protocolos criptográficos (1)

1.- El problema de la identificación del usuario

¿Cómo permitir que un usuario se identifique y autentique ante una máquina -y viceversa- con una clave, password o passphrase y no pueda ser suplantado por un tercero?

2.- El problema del lanzamiento de la moneda

¿Cómo permitir que dos usuarios realicen una prueba con probabilidad ½ -como es el lanzamiento de una moneda- si éstos no se encuentran físicamente frente a frente y, a la vez, asegurar que ninguno de los dos hace trampa?





# Ejemplos de protocolos criptográficos (2)

3.- El problema de la firma de contratos

¿Cómo permitir que dos o más usuarios que se encuentran físicamente alejados puedan realizar la firma de un contrato, asegurando que ninguno de los firmantes va a modificar las condiciones ni negarse a última hora a dicha firma?

4.- El problema del descubrimiento mínimo de un secreto

¿Cómo poder demostrar y convencer a otra persona o a un sistema que uno está en posesión de un secreto, sin por ello tener que desvelarlo ni a ella ni a un tercero?

## Ejemplos de protocolos criptográficos (3)

5.- El problema del juego de póker mental o por teléfono

¿Cómo permitir que dos o más usuarios puedan jugar a través de la red un juego de póker -o cualquier otro- si no están físicamente en una misma mesa de juego y asegurando, al mismo tiempo, que ninguno de ellos va a hacer trampa?

6.- El problema de la división de un secreto o del umbral

Si tenemos un secreto único y por tanto muy vulnerable, ¿cómo permitir que ese secreto se divida en n partes, de forma que juntando k < n partes sea posible reconstruirlo y, en cambio, con k-1 partes imposible su reconstrucción?





## Ejemplos de protocolos criptográficos (4)

7.- El problema del esquema electoral o voto electrónico

¿Cómo realizar unas elecciones a través de una red, de forma que pueda asegurarse que el voto es único y secreto, que los votantes y mesas estén autenticados, y se pueda comprobar que el voto se contabiliza de adecuadamente en el cómputo?

8.- El problema de la transmisión por canales subliminales

Dos usuarios desean intercambiar información a través de un tercero del cual desconfían. ¿Cómo pueden hacerlo sin cifrar la información de forma que este tercero sólo vea un mensaje con texto en claro aparentemente inocente?

# Ejemplos de protocolos criptográficos (5)

9.- El problema del millonario

Dos usuarios desean conocer cuál de los dos tiene más dinero en su cuenta corriente. ¿Cómo pueden hacerlo de forma que, una vez terminado el protocolo, ambos sepan quién de los dos es más rico sin por ello desvelar el dinero que tiene el otro?

10.- El problema del correo electrónico con acuse de recibo

¿Cómo hacer que una vez recibido un correo electrónico, éste sólo pueda ser leído (abierto) si el receptor envía, con anterioridad al emisor, un acuse de recibo como sucede de forma similar con el correo ordinario certificado?





# El protocolo de firma ciega

Supongamos que Adela desea que Benito le firme algo pero sin que Benito se entere de qué es lo que está firmando. En este caso Benito actúa como un ministro de fe, autenticando a Adela.

Protocolo:

- ✉ Adela pone un documento dentro de un sobre.
- ✉ Adela cierra el sobre y se lo envía a Benito.
- ✉ Benito firma el sobre autenticando a Adela y se lo devuelve.
- ✉ Adela abre el sobre y demuestra que Benito al firmar en el sobre cerrado también ha firmado el documento que estaba en su interior.

En el anterior algoritmo, si Benito necesita una comprobación de la identidad de Adela, ésta sencillamente incluye una firma digital suya en el sobre que le permita a Benito comprobar su autenticidad.

http://www.di.ens.fr/~pointche/Documents/Papers/2003_joc.pdf

# Algoritmo de firma ciega RSA (Chaum)

Adela desea que Benito le firme un documento M

- Adela (A) conoce las claves públicas de Benito (B: $n_B$, $e_B$)
- A elige un coeficiente de ceguera k de forma que se cumpla mcd (k, $n_B$) = 1, calcula $k^{-1}$ = inv (k, $n_B$) y luego enmascara su mensaje mediante la siguiente operación:
  - $t_A = M * k^{eB} \bmod n_B$ $\rightarrow$ y lo envía a B
- B firma el valor: $t_B = t_A^{dB} \bmod n_B$ $\rightarrow$ y lo envía a A
- A quita la máscara haciendo $s = t_B * inv(k, n_B) \bmod n_B$
- El resultado es que A tiene $M^{dB} \bmod n_B$, es decir la firma de B del documento M, una firma ciega sobre M.

Comprobación: $t_B = (M * k^{eB})^{dB} \bmod n_B = M^{dB} * k \bmod n_B$
Luego: $[M^{dB} * k * inv(k, n_B)] \bmod n_B = M^{dB} \bmod n_B$





# Ejemplo de algoritmo de firma ciega

- Adela (A) desea que Benito (B) le firme el mensaje M = 65
- Claves públicas de B: $n_B = 299$, $e_B = 7$
- Clave privada y datos de B: $p_B = 13$; $q_B = 23$; $\phi(n_B) = 264$, $d_B = 151$
- A elige k / mcd (k, $n_B$), por ejemplo k = 60. Luego inv (k, $n_B$) = 5
- A enmascara el mensaje: $t_A = M * k^{eB} \bmod n_B = 65 * 60^7 \bmod 299$
- A envía a B: $t_A = 65*226 \bmod 299 = 39$
- B firma $t_A$ con clave privada: $t_B = t_A^{eB} \bmod n_B = 39^{151} \bmod 299 = 104$
- A quita la máscara: $s = t_B * inv(k, n_B) = 104 * 5 \bmod 299 = 221$
- Este valor (221) es el mismo que se obtendría si B firmase su con clave privada el mensaje M, es decir $65^{151} \bmod 299 = 221$

## Transferencia inconsciente o trascordada

Algoritmo de TI propuesto por Michael Rabin en 1981:

- Un usuario A transfiere a un usuario B un dato o secreto con un cifrado probabilístico del 50%.
- El usuario B recibe el dato y tiene una probabilidad del 50% de descubrir el secreto. Una vez que ha recibido el dato, B sabe si éste es el secreto o no.
- No obstante, el usuario A no tiene forma de saber si el usuario B ha recibido el secreto o no.

Esta incertidumbre mutua forzará a los protagonistas a que terminen el protocolo sin hacer trampas.





## Algoritmo de TI de Rabin (1)

| | |
|---|---|
| Paso 1º | A elige dos primos (p y q), calcula n = p∗q y envía el valor n a B. |
| Paso 2º | B elige un número aleatorio x del CCR(n) de forma que que mcd (x,n) = 1, y devuelve a A el valor $K = x^2 \bmod n$. |
| Paso 3º | A calcula las cuatro raíces de $x^2 \bmod n$ y envía a B una de ellas. Las raíces de $x^2 \bmod n$ serán: x, n-x, y, n-y. Sólo A puede hacerlo porque conoce los valores de p y q. |
| Paso 4º | B intenta descubrir el valor de p o q. |

## Conclusión del algoritmo de TI de Rabin

Si B recibe x o n-x no será capaz de encontrar p o q.

No tiene más información que la que tenía porque:

☹ x y n-x son valores que conoce (B ha elegido x).

Si B recibe y o n-y, podrá encontrar p o q.

En este caso, como $x^2 \bmod n = y^2 \bmod n$, entonces:

☺ $(x^2 - y^2) \bmod n = (x+y)(x-y) \bmod n = 0$

Luego $(x+y)(x-y) = k*n$ y se cumplirá que:

$p = mcd\ (x+y, n)$ y

$q = mcd\ (x-y, n)$

Para entenderlo mejor ... veamos un ejemplo





## Ejemplo de algoritmo de TI de Rabin (1)

A Adela tiene como números secretos p y q, valores que corresponden a la factorización del valor n.

B Benito conoce el valor n y deberá descubrir, a partir del protocolo de transferencia inconsciente, p o q.

Ejemplo con valores:

Sea p = 7; q = 13. Luego, $n = p*q = 7*13 = 91$.

1.- A envía a B el valor n = 91.

2.- B elige al azar del CCR(91) el valor x = 15 y calcula $K = 15^2 \bmod 91 = 225 \bmod 91 = 43$. Se lo envía a A.

3.- A recibe K = 43 y calcula las 4 raíces de $x^2 \bmod n$.

# Cálculo de raíces de la TI de Rabin

A calcula las dos raíces de $x^2 \bmod n = K$ de en p y q:

$x_1^2 = K \bmod p = 43 \bmod 7 = 1 \quad \Rightarrow x_1 = 1$

$x_2^2 = K \bmod q = 43 \bmod 13 = 4 \quad \Rightarrow x_2 = 2$

Con estos valores usa ahora el Teorema del Resto Chino

No siempre será tan fácil el cálculo de estas raíces como se verá más adelante

Si no recuerda el Teorema del Resto Chino, repase el archivo de matemáticas. Teníamos que: $x_1 = 1$ y $x_2 = 2$.

Aplicando entonces la ecuación del TRC:





# Aplicación del TRC en la TI de Rabin

$y_1 = \text{inv}(n/p, p) = \text{inv}(91/7, 7) = \text{inv}(13,7) \quad \Rightarrow y_1 = 6$

$y_2 = \text{inv}(n/q, q) = \text{inv}(91/13, 13) = \text{inv}(7,13) \quad \Rightarrow y_2 = 2$

$$x = [(n/p)*y_1*x_1 + (n/q)*y_2*x_2] \bmod n$$

$$\therefore \quad x = (13*6*x_1 + 7*2*x_2) \bmod 91$$

Luego para todas las combinaciones $x_i$, p y q se tiene:

| | | |
|---|---|---|
| $\{x_1, x_2\}$ | $\Rightarrow$ [1,2] | $\Rightarrow$ x = 15 |
| $\{x_1, q-x_2\}$ | $\Rightarrow$ [1,13-2] = [1,11] | $\Rightarrow$ x = 50 |
| $\{p-x_1, x_2\}$ | $\Rightarrow$ [7-1,2] = [6,2] | $\Rightarrow$ x = 41 |
| $\{p-x_1, q-x_2\}$ | $\Rightarrow$ [7-1,13-2] = [6,11] | $\Rightarrow$ x = 76 |

☺ $15^2 \bmod 91 = 50^2 \bmod 91 = 41^2 \bmod 91 = 76^2 \bmod 91 = 43$.

☺ Además se cumple que 15 + 76 = 91 = n y 50 + 41 = 91 = n.

# Conclusión del algoritmo de TI de Rabin

A envía a B cualquiera de estos cuatro valores:15, 50, 41, 76.

- Si B recibe el número 15 (el valor que había enviado a A) o bien n-15 = 91-15 = 76 (que llamaremos valores *x*) no tiene más datos que los que tenía al comienzo del protocolo y no podrá factorizar n.
- Si B recibe cualquiera de los otros dos valores enviados por A (50 ó 41) valores que llamaremos *y*, podrá factorizar n usando la expresión mcd (x+*y*, n) con x, precisamente el valor elegido por B al comienzo del protocolo, es decir 15.
- Si $y = 50 \Rightarrow$ mcd (50+15, 91) = mcd (65, 91) = 13 q = 13
- Si $y = 41 \Rightarrow$ mcd (41+15, 91) = mcd (56, 91) = 7 p = 7





# Elección de p y q en algoritmo de Rabin

Para facilitar el cálculo de las raíces de $x^2 \bmod p$ y $x^2 \bmod q$, el usuario A elegirá los valores de p y q de forma que cumplan:

- ✓ El valor (p+1) sea divisible por 4.
- ✓ El valor (q+1) sea divisible por 4.

Si $x^2 \bmod p = a \bmod p \Rightarrow$ dos soluciones: $x_1$ y $(p - x_1)$

Si $x^2 \bmod q = a \bmod q \Rightarrow$ dos soluciones: $x_2$ y $(q - x_2)$

Estas soluciones se obtienen aplicando el TRC, no obstante si (p+1) es divisible por 4 entonces para este primo p si $x = a^{(p+1)/4}$ se cumple:

$(a^{(p+1)/4})^2 \bmod p = a^{(p+1)/2} \bmod p = a(a^{(p-1)/2}) \bmod p = a$

Esto es válido porque : $a^{(p-1)/2} \bmod p = 1$. Lo mismo sucede con q.

Luego: $x_1 = a^{(p+1)/4} \bmod p$ y $x_2 = a^{(q+1)/4} \bmod q$

# Problema lanzamiento de la moneda (1)

Algoritmo propuesto por Mario Blum en 1982.

Se trata de resolver una apuesta entre dos personas A y B distantes entre sí mediante el lanzamiento de una moneda (cara o cruz).

Situaciones si A lanza la moneda al aire:

Caso 1

1º A lanza la moneda.
2º B hace su apuesta y se lo dice a A.
3º A le dice a B que ha salido "justo lo contrario"
... independientemente de lo que haya salido.
En este caso el usuario A hace trampa ...






# Problema lanzamiento de la moneda (2)

Caso 2

1º A lanza la moneda.
2º B hace su apuesta y se lo dice a A.
3º No sale lo apostado por B y A se lo notifica.
4º B se desmiente y dice que "esa era su apuesta".
Ahora es el usuario B quien hace trampa ...


Si A y B están distantes y no hay un testigo de fe, ¿cómo puede desarrollarse el algoritmo para que ninguno de los dos pueda hacer trampa y, si lo hace, el otro lo detecte?

Esquema de Blum ⟶

## El problema de la moneda según Blum

Soluciones al problema del lanzamiento de la moneda:

- Usar el protocolo de la transferencia inconsciente de Rabin con probabilidad del 50% ya visto, o bien...
- Usar el Esquema General de Blum:

1º A partir de un conjunto de números que la mitad son pares y la otra impares y una función unidireccional f : $x \rightarrow y$, el usuario A elige un valor $x$, calcula $y$ = f ($x$) y lo envía a B.
2º El usuario B apuesta por la paridad de $x$.
3º A le muestra a B el verdadero valor de $x$ y su paridad.





## Condiciones del esquema general de Blum

- B tendrá igual probabilidad de recibir un número par o impar.
- A deberá tener una probabilidad igual (50%) de recibir una apuesta par o impar por parte B.
- Ninguno de los dos podrá hacer trampa.

¿Búsqueda de esa función f?

Antes deberemos explicar qué se entiende por restos cuadráticos y enteros de Blum 

# Restos cuadráticos de Blum

Buscamos una función unidireccional con trampa que cumpla las características del protocolo anterior.

El valor *a* es un resto cuadrático de Blum $R_2$ mod n si:

$$x^2 \bmod n = a$$

siendo mcd (*a*,n) = 1 $\xrightarrow{\text{solución}}$

¿Algún problema? Sí $\Rightarrow$ No sigue la paridad deseada.

Por ejemplo, el resto cuadrático $R_2 = 4$ en mod 11 se obtiene para x = 2 (par) y x = 9 (impar) ya que:

$2^2 \bmod 11 = 4 \bmod 11 = 4$ y $9^2 \bmod 11 = 81 \bmod 11 = 4$





# Enteros de Blum

Un entero de Blum es un número resultado del producto de dos primos p y q, ambos congruentes con 3 módulo 4.

En este caso se cumplirá que:

$y = x^2 \bmod n$ mantendrá la paridad con $z = y^2 \bmod n \ \forall \ x \in Z_n$

Ejemplo: sea n = 11∗19 = 209 y el valor x = 24

11 mod 4 = 3; 19 mod 4 = 3 (cumplen congruencia 3 mod 4 👍)

$y = x^2 \bmod n = 24^2 \bmod 209 = 576 \bmod 209 = 158$

$z = y^2 \bmod n = 158^2 \bmod 209 = 24.964 \bmod 209 = 93$

Como se observa, en este caso y es par y z es impar.

Luego, para todos los restos principales de *y* = 158 (par) que se obtengan con valores de *x* diferentes, el resto cuadrático $z_2$ será siempre el valor 93 (impar).

# Paridad en enteros de Blum

Es importante recalcar que:

- Existirá igual número de soluciones y (pares o impares) que de soluciones z (pares o impares).
- Esto no sucederá con enteros que no sean de Blum.
- Por lo tanto, esta igualdad de paridad en los valores de los restos de z y de y, hará que desde el punto de vista del usuario B que recibe como dato el valor z o resto $R_2$ enviado por A, exista una equiprobabilidad.

  El siguiente cuadro indica la paridad de $R_2$ para algunos módulos enteros y no enteros de Blum.





# Ejemplo de paridad en enteros de Blum

Paridad de elementos de $R_2$ para módulos enteros de Blum

| n | p | q | *y* (pares) | *y* (impares) | *z* (pares) | *z* (impares) |
|---|---|---|---|---|---|---|
| 21 | 3 | 7 | 10 | 10 | 10 | 10 |
| 33 | 3 | 11 | 12 | 20 | 12 | 20 |
| 57 | 3 | 19 | (24) | (32) | (24) | (32) |
| 69 | 3 | 23 | 36 | 32 | 36 | 32 |
| 77 | 7 | 11 | 36 | 40 | 36 | 40 |

Observe que se obtiene igual cantidad de valores y pares que de z pares. De la misma forma, se obtiene igual cantidad de valores y impares que de z impares.

# Ejemplo de paridad en no enteros de Blum

Paridad de elementos de $R_2$ para módulos no enteros de Blum

| n | p | q | y (pares) | y (impares) | z (pares) | z (impares) |
|---|---|---|---|---|---|---|
| 15 | 3 | 5 | 8 | 6 | 6 | 8 |
| 35 | 5 | 7 | 14 | 20 | 8 | 26 |
| 39 | 3 | 13 | 22 | 16 | 16 | 22 |

En este caso no se obtienen cantidades iguales de valores y, z.

Como ejercicio, compruebe que los números 21, 33, 57, 69 y 77 del ejemplo anterior son enteros de Blum y que, por el contrario, 15, 35 y 39 no lo son.





# El algoritmo de Blum

1) A elige dos primos p y q de forma que n = p*q es un entero de Blum (p y q son congruentes con 3 mod 4)
2) A elige un elemento x de $Z_n$ y calcula $y = x^2 \bmod n$. Luego calcula $z = y^2 \bmod n$, valor que envía a B.
3) B recibe z y apuesta por la paridad del valor y.
4) A le informa a B si ha acertado o no en su apuesta. Le muestra también el valor x elegido y el valor de y. Además le comprueba que n es un entero de Blum.
5) B comprueba que $y = x^2 \bmod n$ y que $z = y^2 \bmod n$.
6) A y B han actuado con una probabilidad del 50% en los pasos 2 y 3, respectivamente.

http://zoo.cs.yale.edu/classes/cs467/2005f/course/lectures/ln20.pdf

# Ejemplo del algoritmo de Blum

Sean los primos p = 7 y q = 19
Luego, n = p∗q = 7∗19 = 133
Comprobación de que n = 133 es un entero de Blum:
7 mod 4 = 3; 19 mod 4 = 3

- A elige el valor x = 41 y calcula:
  - $y = x^2 \bmod n$
    - $y = 41^2 \bmod 133 = 1.681 \bmod 133 = 85$
  - $z = y^2 \bmod n$
    - $z = 85^2 \bmod 133 = 7.225 \bmod 133 = 43$
- A envía a B el valor z = 43.
- B debe apostar por la paridad de y.





# Conclusión del ejemplo de Blum

Situación 1 (B acierta)

- Si B acierta y dice que y es impar, A no puede negarle que ha ganado. A debe mostrarle a B los valores x e y. Además debe demostrarle a B que n era un entero de Blum.

Situación 2 (B no acierta)

- Si B no acierta y dice que y es par, A le dice a B que ha perdido, le demuestra que n era un entero de Blum y le muestra el valor x elegido así como el valor y.

Compruebe que a iguales valores de resto principal y resto cuadrático se llega para x = 22, x = 92 y x = 111. Es decir, si se recibe z = 43 (impar) la única posibilidad es que el valor de y sea 85 (impar) y que A haya elegido como valor x alguno de éstos: 22, 41, 92 ó 111.

# La firma de contratos

Dos personas desean firmar un contrato sin un ministro de fe.

- Deben cumplirse dos condiciones:

- Que los firmantes queden obligados a culminar la firma sólo a partir de un punto del protocolo. Esto se conoce como compromiso de los contratantes.
- Que la firma no pueda falsificarse y que, además, pueda ser comprobada por la otra parte.

Un posible algoritmo ⇒

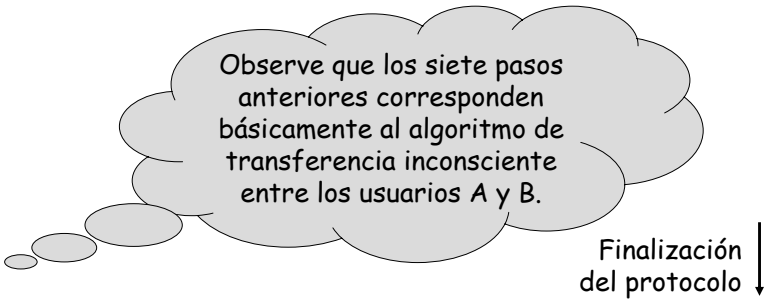




# Algoritmo básico de firma de contratos (1)

1. El usuario A elige dos claves $i_A$ y $j_A$ en un sistema de clave pública y calcula sus claves privadas $i_A^{-1}$ y $j_A^{-1}$.
2. El usuario B elige una clave secreta $K_B$.
3. A envía a B sus dos claves públicas $i_A$ y $j_A$.
4. B elige al azar una de las dos claves recibidas y con ella cifra su clave $K_B$, enviando el resultado al usuario A.
5. A elige al azar una de sus dos claves privadas $i_A^{-1}$ y $j_A^{-1}$ y descifra con dicha clave el valor recibido en el punto 4.
6. A cifra el primer bloque del mensaje de firma usando el valor elegido en el punto 5 como clave y lo envía a B.
7. B descifrará con la clave recibida el bloque de firma.

# Algoritmo básico de firma de contratos (2)




8. A repite la operación de los pasos 5 y 6 para cada uno de los bloques de su firma y B el paso 7.
9. Terminados los bloques de su firma, A repite el paso 6 utilizando ahora su otra clave privada y B el paso 7.





# Algoritmo básico de firma de contratos (3)

- Si A y B han elegido al azar la misma clave con una probabilidad del 50% para cada uno, B descifrará un mensaje con sentido en la primera vuelta. En caso contrario, B recibe un texto sin sentido y deberá esperar hasta recibir el último bloque de la segunda vuelta para obtener el texto en claro.
- Sin embargo, A no tiene cómo saber en cuál de los dos pasos (en la primera o la segunda vuelta) ha logrado B descifrar el criptograma y obtener un texto con sentido lo que fuerza a ambas partes a terminar el algoritmo.

# Firma de contratos: algoritmo de Even (1)

En el año 1985 Even, Goldreinch y Lempel proponen el uso de sistemas de cifra simétricos para la firma de contratos.

1. A elige un conjunto de 2n claves en un sistema simétrico: $C_1$, $C_2$, ..., $C_n$, $C_{n+1}$, ..., $C_{2n}$. Las claves se tomarán como parejas, esto es ($C_1$, $C_{n+1}$), ($C_2$, $C_{n+2}$), ..., ($C_n$,$C_{2n}$) aunque no tengan ninguna relación entre sí.
2. A cifra un mensaje estándar $M_A$ conocido por B con 2n claves $E_{C1}(M_A)$, $E_{C2}(M_A)$, ..., $E_{C2n}(M_A)$ y le envía a B ordenados los 2n criptogramas.
3. A se comprometerá más adelante a la firma del contrato si B puede presentarle para algún i el par ($C_i$, $C_{n+i}$).





# Firma de contratos: algoritmo de Even (2)

4. B elige también un conjunto de 2n claves de un sistema simétrico: $D_1$, $D_2$,..., $D_n$, $D_{n+1}$, ..., $D_{2n}$ y las claves se tomarán como parejas ($D_1$, $D_{n+1}$), ($D_2$, $D_{n+2}$), ..., ($D_n$,$D_{2n}$). B cifra un mensaje estándar $M_B$ conocido por A con las 2n claves $E_{D1}(M_B)$, $E_{D2}(M_B)$, ..., $E_{D2n}(M_B)$ y envía a A 2n criptogramas ordenados. B se comprometerá a la firma en los mismos términos que lo hizo A en el punto anterior.
5. A envía a B cada par ($C_i$, $C_{n+i}$) ordenados mediante una transferencia inconsciente; es decir enviando $C_i$ o $C_{n+i}$ con igual probabilidad. Lo mismo hace B enviando a A ordenadamente uno de los dos valores del par ($D_i$, $D_{n+i}$).

   En este punto A y B tienen la mitad de las claves del otro.

# Firma de contratos: algoritmo de Even (3)

6. Si la longitud de cada clave $C_i$ o $D_i$ es de L bits, A y B realizan el siguiente bucle con $1 \leq i \leq 2n$ para la clave $C_i$ y $D_i$ que no han usado en los pasos anteriores:

   for $1 \leq j \leq L$
   begin
   A envía a B el bit jésimo de todas esas claves $C_i$
   B envía a A el bit jésimo de todas esas claves $D_i$
   end (Esto se conoce como compromiso bit a bit)

7. Al realizar el bucle completo, A y B tienen las 2n claves del otro y se supone firmado el contrato.

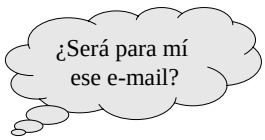
A y B pueden generar mensajes del tipo "Esta es mi mitad izquierda i de mi firma" para cifrar con la clave $C_i$ y $D_i$ y "Esta es mi mitad derecha i de mi firma" para cifrar con la clave $C_{n+i}$ y $D_{n+i}$





# El correo electrónico certificado

¿Cómo podemos estar seguros que un mensaje enviado por correo electrónico ha sido abierto y su contenido conocido sólo por su destinatario autorizado?




Para evitar estas situaciones podemos usar el protocolo del correo certificado

Los sistemas actuales de e-mail permiten emitir desde el cliente de correo del receptor un acuse de recibo.

No obstante, esto sólo significa que "alguien" en extremo receptor desde el buzón de entrada pincha sobre un mensaje nuevo y a la pregunta ¿enviar acuse recibo al emisor? "pisa" Enter eligiendo la opción Sí.

# El correo electrónico certificado

- El usuario A desea enviar un mensaje electrónico como correo certificado al usuario B.
- El usuario A le descubre el mensaje (le envía la clave) sólo después de que el usuario B le envíe el acuse de recibo correspondiente. De la misma manera que actuamos ante un correo certificado: nos entregan "la multa" ✉ si primero firmamos.
- El algoritmo será muy similar al anterior de firma de contratos propuesto por Even.

Veamos una implementación del algoritmo ⇨





# Un algoritmo de correo certificado (1)

- A elige de forma aleatoria n+1 claves ($a_0$, $a_1$, $a_2$, ... $a_n$) de un sistema de cifra simétrico. Las claves $a_i$ no están relacionadas.
- Con la clave $a_0$ A cifrará el documento o carta, $C_0 = E_{a0}(M)$ y se lo envía a B.
- Las claves ($a_1$, $a_2$, ... $a_n$) serán la parte izquierda de la clave $KI_{Ai}$.
- A calcula $a_{n+i} = a_0 \oplus a_i$ para $1 \leq i \leq n$, obteniendo así la parte derecha de la clave ($a_{n+1}$, $a_{n+2}$, ... $a_{2n}$) es decir $KD_{Ai}$.
- A y B se ponen de acuerdo en un mensaje estándar de validación, por ejemplo V = "Mensaje de Validación".
- A cifra el mensaje de validación V con las 2n claves secretas, es decir n claves $KI_{Ai}$ y n claves $KD_{Ai}$.
- Cifrado de validación de la parte izquierda: $VI_{Ai} = E_{KIAi}(V)$.
- Cifrado de validación de la parte derecha: $VD_{Ai} = E_{KDAi}(V)$.
- A envía a B los pares ordenados ($VI_{Ai}$, $VD_{Ai}$) para i = 1, 2, ... n.

# Un algoritmo de correo certificado (2)

- B genera de forma similar n parejas de claves $KI_{Bi}$ y $KD_{Bi}$, 2n claves.
- B genera n parejas de mensajes "Acuse de Recibo de la parte i Izquierda" ($RI_i$) y "Acuse de Recibo de la parte i Derecha" ($RD_i$).
- B cifra las parejas ($RI_i$, $RD_i$) con un sistema simétrico usando las claves $KI_{Bi}$ y $KD_{Bi}$.
- B envía a A las parejas ordenadas ($IB_i$, $DB_i$) = [$E_{KIBi}$(RIi), $E_{KDBi}$(RDi)].
- Mediante una trasferencia trascordada A envía a B una de las dos claves secretas ($K_{IA1}$ o $K_{DA1}$) y lo mismo hace B que envía a A ($K_{IB1}$ o $K_{DB1}$).
- Este proceso se repite hasta que se envían los n valores de claves.
- B usa las claves enviadas por A en el paso anterior para comprobar que al descifrar $D_{KIAi}(V_{Ai})$ o $D_{KDAi}(V_{Ai})$ obtiene el Mensaje de Validación.
- A usa las claves enviadas por B en el paso anterior para comprobar que al descifrar $D_{KIBi}(I_{Bi})$ o $D_{KDBi}(I_{Bi})$ obtiene siempre $RI_i$ o $RD_i$.





# Un algoritmo de correo certificado (3)

- No pueden hacer trampa. A y B ya tienen información suficiente para demostrar ante un ministro de fe que el otro no ha seguido el protocolo.
- A y B se intercambian ahora bit a bit todos los bits de las claves de forma alterna. El primer bit de $KI_{A1}$, el primer bit de $KI_{B1}$, el primer bit de $KI_{A2}$, el primer bit de $KI_{B2}$, ... el primer bit de $KD_{A1}$, etc.
- Este paso se conoce como compromiso bit a bit entre A y B.
- A obtiene todas las claves de B y comprueba todos los Acuse de Recibo pareados, la parte i Izquierda y su correspondiente parte i Derecha.
- B obtiene todas las claves de A y comprueba que todos los envíos de A contienen el Mensaje de Validación. A deberá mostrar todas sus claves a B para que B compruebe que A ha usado la función $a_{n+1} = a_0 \oplus a_i$.
- Como B tiene todas las claves de A calcula ahora $a_0 = KI_{Ai} \oplus KD_{Ai}$. Para ello cualquiera de las parejas de Acuse de Recibo son válidas.
- B descifra el criptograma $Da_0(C_0) = M$ y recupera el mensaje ✉.

# El protocolo del póquer mental (1)

♣ ♦ ♠ ♥ Se trata de encontrar un protocolo que permita el juego del póquer a través de una red de computadores. Debe asegurarse el juego limpio, sin trampas. Aunque el número de jugadores debería ser 4, veremos un ejemplo sólo para dos jugadores. La condición necesaria es que el sistema de cifra sea conmutativo, es decir que permita $D_{KB}\{E_{KA}[E_{KB}(x)]\} = E_{KA}(x)$. Un sistema podría ser Poligh y Hellman con un único cuerpo p de cifra o Vigenère numérico.

1. A y B usan un sistema de cifra simétrica que tenga propiedades conmutativas, usando claves $K_A$ y $K_B$ respectivamente.
2. B cifra -acción mezcla- las 52 cartas (codificadas con un número aleatorio $c_i$) con su clave secreta $K_B$: $E_{KB}(c_i)$ y las envía a A.
3. A elige al azar 5 valores y envía a B: $E_{KB}(c_{B1}, c_{B2}, c_{B3}, c_{B4}, c_{B5})$.





# El protocolo del póquer mental (2)

4. B recibe estos valores y los descifra con su clave secreta $K_B$. Así obtiene: $D_{KB}[E_{KB}(c_{B1}, c_{B2}, c_{B3}, c_{B4}, c_{B5})] = c_{B1}, c_{B2}, c_{B3}, c_{B4}, c_{B5}$. Estas cinco cartas $c_{Bi}$ corresponden a la mano de B.
5. A elige otras cinco cartas de las 47 restantes, las cifra con su clave secreta $K_A$ y envía a B: $E_{KA}[E_{KB}(c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5})]$.
6. B descifra con su clave secreta $K_B$ la cifra anterior y envía a A el resultado: $E_{KA}(c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5})$.
7. A descifra lo anterior con su clave secreta $K_A$ y obtiene su mano $c_{Ai}$: $D_{KA}[E_{KA}(c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5})] = c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5}$.
8. Las restantes 42 cartas permanecerán en poder de A que es quien las reparte. Estas cartas siguen cifradas con la clave de B.
9. Si los jugadores desean cambiar algunas cartas, siguen el mismo procedimiento anterior.

# Protocolo de póquer mental con RSA (1)

En este caso se usará un sistema RSA en el que el módulo de trabajo n será compartido y el par de claves asimétricas de cada jugador, e y d, serán ambas secretas. Veamos un ejemplo para 4 jugadores.

1. El jugador A que repartirá las cartas, todas ellas codificadas con un número aleatorio $c_i$, las mezclará cifrándolas con su clave pública $e_A$: $E_{eA}[c_1, c_2, c_3, ... c_{50}, c_{51}, c_{52}]$ y las envía a B.
2. B elige cinco cartas, las cifra con su clave pública $e_B$ y devuelve a A : $E_{eB}\{E_{eA}[c_{B1}, c_{B2}, c_{B3}, c_{B4}, c_{B5}]\}$.
3. A descifra lo recibido con su clave privada $d_A$ y se lo envía a B: $E_{eB}[c_{B1}, c_{B2}, c_{B3}, c_{B4}, c_{B5}]$.
4. B descifra ahora con su clave privada $d_B$ lo recibido y se queda con su mano $c_{Bi} = c_{B1}, c_{B2}, c_{B3}, c_{B4}, c_{B5}$.





# Protocolo de póquer mental con RSA (2)

5. El jugador B pasa las restantes 47 cartas al jugador C y se repiten los pasos 2 al 4 anteriores entre C y A, usando ahora las claves $e_C$, $d_A$ y $d_C$.
6. Terminado el paso 5, el jugador C tendrá entonces como mano $c_{Ci} = c_{C1}, c_{C2}, c_{C3}, c_{C4}, c_{C5}$.
7. El jugador C pasa las restantes 42 cartas al jugador D y se repiten los pasos 2 al 4 entre D y A, usando ahora las claves $e_D$, $d_A$ y $d_D$.
8. Terminado el paso 7, el jugador D tendrá entonces como mano $c_{Di} = c_{D1}, c_{D2}, c_{D3}, c_{D4}, c_{D5}$.
9. El jugador D devuelve las 37 cartas que quedan y que están cifradas con su clave pública: $E_{eD}\{E_{eA}[c_1, c_2, c_3, ... c_{36}, c_{37}]\}$ al jugador A.

# Protocolo de póquer mental con RSA (3)

10. El jugador A elige 5 cartas entre las 37 y devuelve al jugador D: $E_{eD}\{E_{eA}[c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5}]\}$.
11. El jugador D descifra con su clave privada $d_D$ lo recibido y envía a A: $E_{eA}[c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5}]$.
12. El jugador A descifra con su clave privada $d_A$ lo recibido y se queda con su mano $c_{Ai} = c_{A1}, c_{A2}, c_{A3}, c_{A4}, c_{A5}$.
13. Todos tienen su mano de juego. Las restantes 32 cartas quedan en poder de A cifradas por D y A: $E_{eD}\{E_{eA}[c_1, c_2, ... c_{31}, c_{32}]\}$.
14. Si un jugador X desea descarte, pide las cartas a A, elige las que desea, las cifra con su clave pública $e_X$ y se las devuelve a A, quien las envía a D para que descifre con su clave privada $d_D$. D las devuelve a A para que descifre con su clave privada $d_A$ y A envía a X: $E_{eX}$[cartas elegidas en su descarte].





# El canal subliminal

- Como ejemplo de canal subliminal, en un supermercado podrían incluir en la música ambiental una información no audible y que sólo nuestro subconsciente sea capaz de interpretar. No se extrañe de ello, este tipo de experimentos se han probado hace muchos años atrás.
- El concepto de canal subliminal fue propuesto por Gustavus Simmons en 1983. Se conoce también como el problema de los prisioneros.

- Dos prisioneros cómplices de un delito son encarcelados en celdas separadas. Si entre ellos pueden intercambiarse mensajes a través de un carcelero que los puede leer, ¿cómo hacen para que esos mensajes en principio inocentes, lleven de forma subliminal un mensaje cifrado y que el carcelero sea incapaz de dilucidar ese secreto?

  La técnica denominada esteganografía, hoy en día de moda, realiza una operación similar, normalmente ocultando un texto bajo una fotografía.

# El problema de los prisioneros

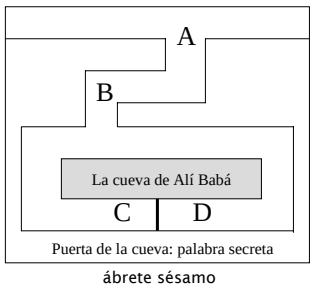
- El prisionero A genera un mensaje inocente M que desea enviar al prisionero B a través del guardia.
- Utilizando una clave secreta K acordada con anterioridad, el prisionero A "firma" el mensaje de forma que en esa firma se esconda el mensaje subliminal.
- El guardia recibe el mensaje "firmado" por A y como no observa nada anormal se lo entrega al prisionero B.
- El prisionero B comprueba la firma de su compañero A, autentica el mensaje y lee la información subliminal en M.

Existen varios esquemas de uso del canal subliminal para proteger la información, entre ellos el propio esquema de Simmons basado en la factorización de un número grande n compuesto por tres primos p, q y r. Habrá por tanto $2^3 = 8$ raíces de las cuales sólo algunas se usarán como valores válidos y otras no. Hace uso del Teorema del Resto Chino.





# Transferencia con conocimiento nulo TCN




Este modelo fue presentado por J. Quisquater y L. Guillou en Crypto '89 para explicar el protocolo de transferencia con conocimiento cero o nulo.

Algoritmo:

1. Mortadelo y Filemón se acercan a la cueva en el punto A.
2. Mortadelo se adentra en la cueva hasta llegar al punto C o D.
3. Filemón se acerca al punto B de la cueva y le pide a Mortadelo que salga por la ladera derecha o izquierda, según desee.
4. Mortadelo satisface la petición de Filemón y sale por la ladera que éste le ha solicitado, usando si es menester la palabra secreta para abrir la puerta.
5. Se repite el proceso desde el comienzo hasta que Filemón se convence que Mortadelo conoce la palabra secreta.

# Esquema de TCN de Koyama

- A desea demostrar a B que conoce la clave secreta RSA de un tercer usuario C, es decir $d_C$. Como es lógico también conocerá $p_C$, $q_C$ y $\phi(n_C)$. Las claves públicas de C son $n_C$ y $e_C$ que conocen tanto A como B.
- A y B se ponen de acuerdo y eligen dos valores aleatorios k y m con la condición de que $k*m = e_C \bmod \phi(n_C)$.
- Como A debe mantener en secreto el valor de $\phi(n_C)$ le propone a B que en cada ejecución del algoritmo elija un número m primo por lo que A calcula $k = [\{inv (m, \phi(n_C)\}*e_C] \bmod \phi(n_C)$.
- A propone a B un texto aleatorio M o bien A y B generan este texto usando, por ejemplo, un algoritmo de transferencia trascordada.
- Usando la clave privada $d_C$ de C, ahora A calcula $C = M^{dC} \bmod n_C$. Luego calcula $X = C^k \bmod n_C$ y envía el valor X a B.
- B recibe X y comprueba si $X^m \bmod n_C$ es igual al texto M. Si es así, quiere decir que A ha usado $d_C$, la clave privada de C.
- Se repite el proceso las veces que haga falta hasta que B acepte que A conoce clave privada de C.





# ¿Por qué funciona el esquema de Koyama?

Por simplicidad supondremos que los datos de C no tienen subíndice:

1. A conoce n, e, d, p, q, $\phi(n)$ y el texto M; B conoce n, e y el texto M.
2. B elige un primo m y se lo envía envía a A.
3. A calcula $k = [\{inv (m, \phi(n)\}*e] \bmod \phi(n)$.
4. A calcula $C = M^d \bmod n$ y $X = C^k \bmod n = M^{dk} \bmod n$ y envía este valor X a B.
5. B recibe X y calcula $X^m \bmod n = M^{(dk)m} \bmod n = M^{km*d} \bmod n$, pero como $k*m = e \bmod \phi(n)$ entonces $M^{km*d} \bmod n = M^{e*d} \bmod n = M$.
6. La única posibilidad para que B recupere el texto M en el paso 5, es que A haya usado en la cifra del paso 4 la clave privada d.
7. Si B no se convence en el primer intento, ambos repiten el algoritmo con valores primos m distintos en cada iteración, hasta que se cumpla un umbral ante el que B acepte que A está en posesión de ese secreto.

# Ejemplo del esquema de TCN de Koyama

- Supongamos que A desea demostrar a B que conoce la clave privada de C. Los valores públicos de C son n = 77, e = 13.
- El mensaje M acordado por A y B es la palabra PADRINO con la codificación que se muestra a continuación:

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | Ñ | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 |

- Supongamos que B elige como valor aleatorio m = 29.
- A calcula k según el algoritmo de Koyama y para cada valor $M_i$ del mensaje (P = 18, A = 2, D = 5, etc.) calcula primero $C = M_i^d \bmod n$ y luego $X = C^k \bmod n$ = 30, 39, 31, 27, 54, 36, 68 que envía a B.
- B calcula $30^{29} \bmod 77$, $39^{29} \bmod 77$, $31^{29} \bmod 77$, $27^{29} \bmod 77$, $54^{29} \bmod 77$, $36^{29} \bmod 77$, $68^{29} \bmod 77$ y obtiene la cadena de caracteres PADRINO.
- El protocolo puede repetirse para otros valores primos m que elija B y siempre se obtendrá como resultado el mismo mensaje M. ✌





# Solución del ejemplo de TCN de Koyama

- Como n = 77, es obvio que p = 7, q = 11, $\phi(n) = 60$. Por lo tanto, puesto que e = 13 entonces d = inv {e, $\phi(n)$} = inv (13, 60) = 37.
- $M_1 = 18$; $M_2 = 2$; $M_3 = 5$; $M_4 = 20$; $M_5 = 10$; $M_6 = 15$; $M_7 = 17$.
- $C_1 = 18^{37} \bmod 77 = 39$; $C_2 = 2^{37} \bmod 77 = 51$; $C_3 = 5^{37} \bmod 77 = 47$; $C_4 = 20^{37} \bmod 77 = 48$; $C_5 = 10^{37} \bmod 77 = 10$; $C_6 = 15^{37} \bmod 77 = 71$; $C_7 = 17^{37} \bmod 77 = 52$.
- $k = [\{\text{inv}(m, \phi(n)\} * e] \bmod \phi(n) = \text{inv}(29, 60) * 13 \bmod 60 = 17$.
- $X_1 = 39^{17} \bmod 77 = 30$; $X_2 = 51^{17} \bmod 77 = 39$; $X_3 = 47^{17} \bmod 77 = 31$; $X_4 = 48^{17} \bmod 77 = 27$; $X_5 = 10^{17} \bmod 77 = 54$; $X_6 = 71^{17} \bmod 77 = 36$; $X_7 = 52^{17} \bmod 77 = 68$. Luego X = 30, 39, 31, 27, 54, 36, 68.
- $30^{29} \bmod 77 = 18 = P$; $39^{29} \bmod 77 = 2 = A$; $31^{29} \bmod 77 = 5 = D$ $27^{29} \bmod 77 = 20 = R$; $54^{29} \bmod 77 = 10 = I$; $36^{29} \bmod 77 = 15 = N$ $68^{29} \bmod 77 = 17 = O$.

En este ejemplo el valor de n es muy pequeño y resulta muy fácil romper la clave privada simplemente factorizando el módulo ☺.

## El voto electrónico o por ordenador

- Todos tenemos de una u otra forma una idea intuitiva, aunque quizás no completa, sobre cómo se desarrolla un proceso electoral.
- La pregunta es si es posible realizar este tipo de eventos desde Internet, lo que se conoce como esquema electoral.
- La respuesta es sí con la ayuda de técnicas y protocolos criptográficos aunque no se trata sólo de un problema de implementación técnica; es menester tener en cuenta otros factores importantes, a saber:
  - Socio-políticos, económicos, jurídicos, legislativos...





## Definición de esquema electoral

"Un esquema de votación electrónica es una aplicación distribuida y constituida por un conjunto de mecanismos criptográficos y protocolos que, de forma conjunta, permiten que se realicen elecciones en una red de computadores, de forma segura, incluso suponiendo que los electores legítimos pueden tener un comportamiento malicioso."

*Andreu Riera*
*Tesis Doctoral, Universidad Autónoma de Barcelona, España, 1999*

## Requisitos de un esquema electoral (1)

Requisitos de un esquema electoral:

- Sólo pueden votar quienes estén censados.
- El voto debe ser secreto.
- El voto debe ser único por cada votante.
- Se contabilizarán todos los votos válidos.
- El recuento parcial no debe afectar a votos que se emitan con posterioridad.

sigue





## Requisitos de un esquema electoral (2)

Requisitos de un esquema electoral:

- Cada votante podrá comprobar que su voto ha sido tenido en cuenta en el escrutinio.

Esto último es muy importante ✌

Y, además:

- Se debe proteger el proceso contra ataques en red.
- El proceso debe ser factible, práctico y dentro de lo posible de uso universal.

## Primera aproximación del voto electrónico

MCV = Mesa Central de Votación

- El votante cifra su voto con la clave pública de MCV.
- El votante envía su voto a la MCV.
- La MCV descifra el voto y lo contabiliza.
- La MCV hace público el resultado.

¿Qué problemas presenta este esquema? TODOS... ☹

La MCV no sabe de dónde vienen los votos, si éstos son válidos o no y si alguien vota más de una vez. Además puede conocer la identidad del votante por lo que se vulnera el secreto del voto. Lo único que aquí se protege es el secreto del voto ante terceros.





## Segunda aproximación del voto electrónico

MCV = Mesa Central de Votación

- El votante firma su voto con su clave privada y lo cifra luego con la clave pública de MCV.
- El votante envía su voto a la MCV.
- La MCV descifra el voto, lo contabiliza y hace público el resultado.

¿Qué problema tenemos ahora?

En este nuevo esquema se satisface que cada votante autorizado vote una sola vez, no obstante seguimos vulnerando el secreto del voto ante la MCV.

## Tercera aproximación del voto electrónico

El tercer esquema contempla dos mesas:

- MCV = Mesa Central de Votación
- MCL = Mesa Central de Legitimación

✍ Evita que la MCV conozca a quién ha votado el votante, mediante un protocolo entre ambas, y además gestionan una lista de votantes censados.

☞ MCV y MCL deben ser órganos independientes

Veamos cómo funciona este esquema





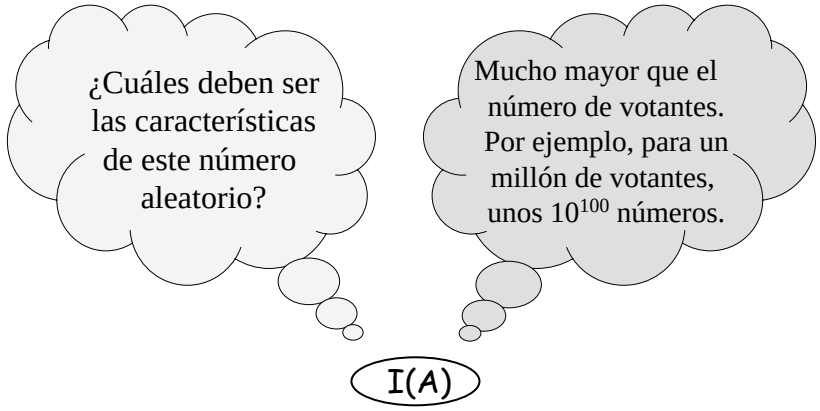
## Un protocolo de voto electrónico (1/5)

1. El votante A envía a la MCL el mensaje:
   Buenos días, soy A y vengo a votar.
2. La MCL verifica si A está censado. Si no es un votante legítimo rechaza la solicitud. Si es legítimo, le envía un número aleatorio de identificación único i(A) y le borra de la lista para impedir que vuelva a votar.

Toda la información irá cifrada y firmada

Características de i(A)

## Un protocolo de voto electrónico (2/5)








## Un protocolo de voto electrónico (3/5)

3. La MCL envía a la MCV la lista de números de validación.
4. El votante A escoge una identificación secreta s(A) y envía a la MCV el mensaje formado por el trío [i(A), v(A), s(A)] es decir:
   - su identificación i(A)
   - su voto v(A)
   - su número secreto s(A)

Puede generarlo internamente con su sistema de cifra. Será también un valor de muchos dígitos.

## Un protocolo de voto electrónico (4/5)

5. La MCV verifica que el número i(A) de identificación se encuentra en el conjunto N de los números censados y cruza los datos para evitar que se vote más de una vez. Quita i(A) del conjunto N y añade s(A) al conjunto de electores que han optado por la opción v(A).
6. La MCV contabiliza los votos y hace público el resultado, junto con la lista de números secretos s(A) que han votado a la opción v(A) ... luego →





## Un protocolo de voto electrónico (5/5)

☺ Cada elector puede comprobar si su voto ha sido contabilizado sin hacer pública su opción.

¿Qué pasa si MCV y MCL no son independientes?

Si las dos mesas, MCV y MCL, no tienen la idoneidad y la integridad que se presume, la solución está en el uso de una diversidad de esquemas más desarrollados que evitan esta anomalía mediante protocolos, entre ellos ANDOS (All-or-Nothing Disclosure Of Secrets) Distribución Anónima de Números de Validación, pero esto ya se escapa del objetivo de este libro.

## Otros esquemas de mesas electorales

Hay muchos otros esquemas con dos mesas, una única mesa e incluso ninguna, cada uno con sus características propias. Entre ellos tenemos:

- Modelo de Cohen y Fisher (1985)
- Modelo de Fujioka y otros (1992)
- Modelo de Park y otros (1993)
- Modelo de Sako y Killian (1995)
- Modelo de Borrel y Rifà (1996)

Observe que son modelos y esquemas muy recientes.





## Estado del arte en voto electrónico

- Existen diversos modelos y esquemas, algunos de ellos probados con éxito con un número reducido de electores.
- No está todavía bien solucionado el problema de la protección física y lógica de la una red como Internet ante ataques masivos, denegación de servicio, etc. Es uno de los problemas al que se enfrentan estos esquemas, su difícil escalabilidad. No obstante, sí se puede asegurar la factibilidad de un proceso de voto telemático práctico y seguro en cuanto a privacidad y autenticidad.
- Para mayor información sobre voto telemático:

http://vototelematico.diatel.upm.es/

Fin del capítulo

# Cuestiones y ejercicios (1 de 4)

1. ¿Qué diferencia hay entre un protocolo de red como por ejemplo TCP/IP con un protocolo criptográfico?
2. En una transferencia inconsciente de Rabin, A y B se intercambian lo siguiente. A envía a B el número compuesto n = 55, B elige el valor x = 9 y envía $x^2$ mod n a A. ¿Qué valores de los 4 que puede devolver A a B permiten a este último factorizar el cuerpo n?
3. ¿Qué sucede si en el ejemplo anterior B elige x = 10?
4. ¿En el ejemplo anterior, están bien elegidos por A los valores de p y q? Qué valores usaría si p y q fuesen números mayores que 10?
5. Presente una solución al problema del lanzamiento de la moneda a través del esquema de transferencia inconsciente de Rabin.
6. Calcule todos los valores de $x^2$ mod 13. Sea a = 2, 3, 4, 5, 6. ¿Cuáles son restos cuadráticos de Blum en el cuerpo n = 13?, ¿por qué?





# Cuestiones y ejercicios (2 de 4)

7. Para los restos cuadráticos encontrados en el ejercicio anterior, ¿se cumple la paridad en el valor de x? ¿Qué significa esto?
8. ¿Cuáles de los siguientes siete números compuestos son enteros de Blum: 69, 143, 161, 189, 319, 713, 1.333? ¿Justifíquelo?
9. Encuentre todos los restos de y, z para el entero de Blum n = 33.
10. En un protocolo con enteros de Blum, A trabaja en n = 77 y elige el valor x = 15. Calcula $y = x^2$ mod n y luego $z = y^2$ mod n. Envía el valor z a B. ¿Cuál es el escenario del protocolo y cómo trabaja?
11. ¿Qué sucede si en el esquema anterior de Blum el usuario B conoce el valor de los primos p y q? ¿Funciona así el protocolo?
12. En el algoritmo de firma de contratos con claves asimétricas y una clave simétrica, ¿cómo puede comprobar el usuario B que A está usando en cada vuelta una clave privada distinta y no hace trampa?

# Cuestiones y ejercicios (3 de 4)

13. ¿Cómo se entiende el compromiso de firma de A y B en el esquema de firma de contratos de Even?
14. En el esquema anterior de Even ¿qué relación tiene el compromiso bit a bit con el término correcto del protocolo? ¿Por qué están A y B obligados a terminar el protocolo hasta el último bit?
15. Se desea que el usuario B le firme de forma ciega al usuario A el mensaje M = 100. Si $n_B = 253$, $e_B = 19$ y el usuario A elige k = 25, realice y compruebe el protocolo de firma ciega.
16. ¿Para qué podría servir un protocolo como el de firma ciega?
17. ¿Por qué decimos que el actual acuse de recibo de los clientes de correo electrónico no corresponde a uno verdadero?
18. En el algoritmo de correo con acuse de recibo, compruebe que B obtiene la clave de descifrado del mensaje haciendo $KI_{Ai} \oplus KD_{Ai}$.





# Cuestiones y ejercicios (4 de 4)

19. Generalice el póker mental con cifra simétrica para 4 jugadores.
20. ¿Qué diferencia hay en cuanto a la elección de cartas de una mano entre el esquema de póker mental con cifra simétrica y el esquema con cifra asimétrica? ¿Es esto un inconveniente o no?
21. En el esquema de Quisquater y Guillou de conocimiento nulo, si Mortadelo y Filemón repiten el protocolo 20 veces, ¿cuál es la probabilidad de que el primero engañe al segundo?
22. Usando el software Simulación de Fortaleza de Cifrados, repita el ejercicio de TCN de Koyama con n = 465.256.980.233 y e = 4.171. B elige el valor m = 131, el mensaje M es el mismo y se recibe: $X_1 = 394.106.275.745$; $X_2 = 342.981.204.125$; $X_3 = 49.911.481.740$; $X_4 = 366.983.136.296$; $X_5 = 56.903.681.682$; $X_6 = 246.374.030.904$; $X_7 = 254.152.395.874$. ¿Qué valor tiene la clave privada d?

Use el portapapeles

# Prácticas del tema 19

Software Fortaleza: http://www.criptored.upm.es/software/sw_m001e.htm

1. Usando el software que se indica, compruebe los valores de los ejercicios presentados y resueltos en este capítulo.
2. Resuelva los ejercicios que se han propuesto en la sección anterior de este capítulo. Invéntese luego algunos ejemplos con números grandes.
3. Usando este software, compruebe que es posible realizar el protocolo del póquer mental mediante el algoritmo de Poligh-Hellman.

Software CripClas: http://www.criptored.upm.es/software/sw_m001c.htm

1. Usando el software CripClas compruebe que el sistema de Vigenère sirve para realizar el protocolo del póquer mental.

Nota: en este año 2006 ya estará disponible un software de prácticas específico para la resolución de algunos protocolos criptográficos.