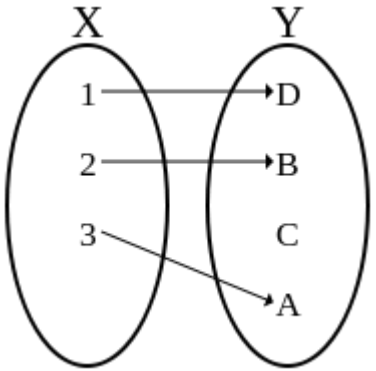
2- Métodos clásicos de cifrado y criptoanálisis

Definición de Criptosistema

- Recordar el concepto de **criptosistema:**
  - Podemos definir un criptosistema como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
    - Un conjunto finito de textos planos asignado a P
    - Un conjunto finito de textos cifrados asignado a C
    - Un conjunto finito de claves a K
    - Una función de cifrado (algoritmo de encriptación): E_k
    - Una función de descifrado (algoritmo de desencriptación): D_k
    - Para todo elemento x perteneciente a P se verifica que
      - **$D_k(E_k(x))=x$ (Inyectividad del criptosistema)**
  - Esta definición es para un criptosistema simétrico por bloques, para cualquier otro criptosistema la definición es análoga teniendo en cuenta siempre la condición indispensable de inyectividad o función uno a uno:

Imagen extraída de http://commons.wikimedia.org/wiki/File:Injection.svg



Cifrado por desplazamiento

- Algunas pequeñas definiciones previas.
- Este cifrado está basado en la aritmética modular.
  - Ej.: 9 mod 3=0, a mod m = r $\rightarrow$ a=qm+r, q$\in$Z, donde Z es el conjunto de números enteros.
- Que significa a $\equiv$ b mod m, se dice que a es congruente con b en aritmética modular m, sii m divide a la diferencia (a-b) (se expresa como m|(a-b)).
- Se puede ver que a mod m = b mod m entonces a $\equiv$ b mod m, de tal forma que a = b + km, k$\in$Z, donde Z es el conjunto de números enteros.

Cifrado por desplazamiento

- El algoritmo trabaja en siguiente conjunto finito en aritmética modular m:
  - $Z_m=\{0, 1, 2, \ldots, m-1\}$
- En este conjunto finito podemos definir dos operaciones "+" y "*" como:
  - Estas operaciones funcionan igual, salvo que el resultado final se aplica aritmética modular m (se reduce por m):
    - Ej. 11 * 13 =15 en Z_{16}, ya que 11 * 13 =143 mod 16 =15 (143 = 8 * 16 +15).
- El nuevo conjunto Z_m, con la operaciones "+" y "*" cumple una serie de propiedades.

Cifrado por desplazamiento

- Respecto la operación "+" cumple:
  - Cierre: $\forall a, b \in Z_m \rightarrow a + b \in Z_m$
  - Asociativa: $\forall a, b, c \in Z_m \rightarrow a + (b + c) = (a + b) + c$
  - Identidad: $\forall a \in Z_m \rightarrow a + 0 = a$
  - Inverso (simétrico): $\forall a \in Z_m \rightarrow a + (-a) = 0$ (-a es el elemento simétrico de a)
    - $-a = m - a$, ya que $a + m - a \bmod m = 0$ (la operación resta está definida: a–b en Z_m es $a + m - b \bmod m$)
  - Conmutativa: $\forall a, b \in Z_m \rightarrow a + b = b + a$
- Con la cuatro primeras propiedades, Z_m forma una estructura de grupo, y añadiendo la quinta propiedad es un **grupo conmutativo o abeliano**.
- Demostrar estas 5 propiedades para casa.

Cifrado por desplazamiento

- Respecto la operación "*" cumple:
  - Cierre: $\forall a, b \in Z_m \rightarrow a * b \in Z_m$
  - Conmutativa: $\forall a, b \in Z_m \rightarrow a * b = b * a$
  - Asociativa: $\forall a, b, c \in Z_m \rightarrow a * (b * c) = (a * b) * c$
  - Identidad: $\forall a \in Z_m \rightarrow a * 1 = a$ (1 es elemento identidad)
  - Distributiva: $\forall a, b, c \in Z_m \rightarrow (a + b) * c = (a * c) + (a * c)$, o $a * (b + c) = (a * b) + (a * c)$
  - **NO tiene Inverso** y por lo tanto no es un grupo respecto a esta operación.
    - Ej. En Z_{27} $4^{-1}=7$ porque $4*7=28 \bmod 27= 1$, pero y 3^{-1}???
    - Ej. En Z_{26} $3^{-1}=9$ porque $9*3=27 \bmod 26= 1$, pero y 2^{-1}???
- Con todas las propiedades de las dos operaciones, Z_m forma una estructura de **anillo conmutativo finito**.
- Demostrar estas 5 propiedades para casa.

Cifrado por desplazamiento

- Ahora ya podemos definir el cifrado por desplazamiento:
  - Podemos definir un criptosistema desplazamiento como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
    - Un conjunto finito de textos planos asignado a $P=Z_m$, tal que $x \in P$
    - Un conjunto finito de textos cifrados asignado a $C=Z_m$, tal que $y \in C$
    - Un conjunto finito de claves a $K=Z_m$, tal que $k \in K$
    - Una función de cifrado (algoritmo de encriptación): $E_k(x)= x + k \bmod m$
    - Una función de descifrado (algoritmo de desencriptación): $D_k(y)= y - k \bmod m$
- Es un criptosistema si: $D_k(E_k(x))=x$
  - $E_k(x)= x + k \bmod m$, $D_k(x + k \bmod m) = x + k - k \bmod m = x$

Cifrado por desplazamiento

- Un ejemplo de este cifrado es el cifrado del cesar para k = 3, que era la clave que utilizaba el cesar.
- Cual es el espacio de claves y por tanto su fortaleza:
  - $|K| = |Z_m| = m$
- Cuando más grande sea el tamaño de claves más difícil es atacar por la fuerza bruta.
- Se pueden hacer dos observaciones genéricas:
  - Para que un criptosistema sea eficiente E_k y D_k tienen que ser rápidos y óptimos en computación.
  - Además si un observador malo ve $y \in C$ no tiene que ser capaz de determinar $x \in P$ (en este criptosistema solo hace falta saber k).

Cifrado de sustitución

- Podemos definir un criptosistema sustitución como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
  - Un conjunto finito de textos planos asignado a P=Z_m, tal que x ∈ P
  - Un conjunto finito de textos cifrados asignado a C=Z_m, tal que y ∈ C
  - Un conjunto finito de claves de permutaciones, tal que k ∈ K (k una permutación del conjunto K que es el conjunto de todas las posibles permutaciones de m símbolos, π). La fortaleza es |K| = m!
  - Una función de cifrado (algoritmo de encriptación): $E_k(x)= \pi (x)$
  - Una función de descifrado (algoritmo de desencriptación): $D_k(y)= \pi^{-1} (x)$
- Ej. de una permutación aleatoria π : así $E_k(a)$ = X, o $E_k(b)$ = N, o $D_k(Q)$ = j, etc.
  - a b c d e f g h i j k l m n o p q r s t u v w x y z
  - X N Y A H P O G Z Q W B T S F L R C V M U E K J D I

Cifrado afín

- Podemos definir un criptosistema afín como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
  - Un conjunto finito de textos planos asignado a P=Z_m, tal que x ∈ P
  - Un conjunto finito de textos cifrados asignado a C=Z_m, tal que y ∈ C
  - Un conjunto finito de claves asignado a K= Z_m x Z_m^*, tal que k ∈ K
  - Una función de cifrado (algoritmo de encriptación):
    - $E_k(x)$= a*x + b mod m, con a ∈ Z_m^*, y b ∈ Z_m
  - Una función de descifrado (algoritmo de desencriptación): $D_k(y)$= (y – b) * a^{-1} mod m.
- Nuestro objetivo ahora es calcular a^{-1} mod m (inverso multiplicativo de a en aritmética modular m).
- No todos los elementos de Z_m tienen inverso multiplicativo (recordar que no existe la propiedad de elemento inverso respecto la operación "*" en Z_m).
- La operación de inverso multiplicativo, me define un nuevo conjunto Z_m^*
- Que está compuesto por todos aquellos elementos Z_m que tienen inverso multiplicativo.

Cifrado afín: Algoritmo de Euclides

- Como definimos Z_m^*: $Z_m^* = \{\forall a \in Z_m \mid \text{mcd}(a, m) = 1\}$ (demostrarlo para casa, hoja de problemas).
- Este cifrado existe como tal solamente sii $a \in Z_m^*$, para que se pueda realizar la desencriptación: $D_k(y) = (y - b) * a^{-1} \bmod m$.
- Por tanto necesitamos ahora calcular dos cosas:
  - Como se calcula de manera eficiente el mcd: mediante el **A. Euclides**.
  - Como se calcula de manera eficiente el a^{-1} en Z_m: mediante el **A. Euclides extendido**.
- El algoritmo de Euclides se basa en la propiedad de que
  - $d = \text{mcd}(a, b) = \text{mcd}(b, a \bmod b)$ (demostrarlo para casa, hoja de problemas).

Cifrado afín: : Algoritmo de Euclides

- El algoritmo de Euclides se basa en la propiedad de que
  - d = mcd (a, b) = mcd (b, a mod b) (demostrarlo para casa, hoja de problemas).
  - d=mcd (a, b), a>b, a y b ≠ 0

$r_0 = a,\ r_1 = b$ $\qquad 0 < r_1 < r_0$

$r_0 = q_1 r_1 + r_2$ $\qquad 0 < r_2 < r_1$

$r_1 = q_2 r_2 + r_3$ $\qquad 0 < r_3 < r_{2^\circ}$

$r_2 = q_3 r_3 + r_4$ $\qquad 0 < r_4 < r_3$

……

$r_{n-2} = q_{n-1} r_{n-1} + r_n$ $\qquad 0 < r_n < r_{n-1}$

$r_{n-1} = q_n r_n + 0$ $\qquad d = r_n$

r_0 mod r_1

r_1 mod r_2

  - d = mcd (a, b) = mcd (r_0, r_1) = mcd (r_1, r_2) = … = mcd (r_{n-2}, r_{n-1}) = mcd (r_{n-1}, r_n) = mcd $(r_n, 0)$ = r_n

El cero es divisible por cualquier número que no sea cero (no queda resto al dividirlo por otro número), i.e. el resultado de dividir el cero entre cualquier otro número siempre es cero y la división siempre es exacta.

Cifrado afín: : Algoritmo de Euclides

- Ejemplo: mcd (26, 15) =d
- 26 = 1 * 15 + 11
  15 = 1 * 11 + 4
  11 = 2 * 4 + 3
  4 = 1 * 3 + 1
  3 = 3 * 1 + 0

26/15 (26 entre 15 a 1 y me sobran 11)
15/11
11/4
4/3
3/1

Divisiones enteras

mcd (26, 15) =3

Cifrado afín: Algoritmo de Euclides

- Una forma sencilla de implementación para el mcd (a, b) :

$r_0 \leftarrow a$
$r_1 \leftarrow b$
$n \leftarrow 1$
while $r_n \neq 0$

do
- $q_n \leftarrow [r_{n-1} / r_n]$
- $r_{n+1} \leftarrow r_{n-1} - q_n r_n$
- $n \leftarrow n+1$

$n \leftarrow n-1$
return $(q_1, q_2, \ldots, q_n, r_n)$

- Donde el símbolo "[ ]" significa redondear al entero más cercano por debajo, en la división entera.

Cifrado afín: Algoritmo extendido de Euclides

- El algoritmo extendido de Euclides calcula el inverso multiplicativo de un número a en aritmética modular m.
- Supongamos que mcd (m, a) = 1, en el cifrado afín:
  - $y = ax + b \bmod m$, siendo así un criptosistema.
- En el algoritmo de Euclides podemos poner todos los restos r_i en función solo de r_0 y r_1 , de manera recursiva obteniendo así la expresión del tipo para el último resto r_n, que es mcd (m, a)=1:
  - $1 = r_n = m\, u_n + a\, v_n$
- Por tanto con esta expresión ya tenemos el inverso multiplicativo de a en aritmética modular m, por la propia definición de congruencia:
  - $m\, u_n + a\, v_n = 1 \rightarrow a\, v_n \equiv 1 \bmod m \rightarrow v_n = a^{-1} \bmod m$

Cifrado afín: Algoritmo extendido de Euclides

- Como sacamos los restos para el Algoritmo extendido de Euclides en función solo de r_0 y r_1 , hasta llegar al resto r_n de manera recursiva:
- Utilizamos obviamente el algoritmo de Euclides
- d=mcd (a, b), a>b, a y b ≠ 0

  $r_0=a$, $r_1=b$

  $r_0=q_1r_1+r_2$ (1)

  $r_1=q_2r_2+r_3$ (2)

  $r_2=q_3r_3+r_4$ (3)

  .....

  $r_{n-2}=q_{n-1}r_{n-1}+r_n$

  $r_{n-1}=q_nr_n+0$ $d=r_n$

Cifrado afín: Algoritmo extendido de Euclides

- De (1) despejamos r_2:
  - $r_2 = r_0 - q_1 r_1$ (ya en función de r_0 y r_1)
- De (2) despejamos r_3, y sustituimos el valor de r_2 para dejarlo otra vez en función de r_0 y r_1:
  - $r_3 = r_1 - q_2 r_2 = r_1 - q_2(r_0 - q_1 r_1) = \ldots = (1+q_2 q_1)\, r_1 - q_2 r_0$
- De (3) despejamos r_4, y sustituimos el valor de r_2 y r_3 para dejarlo otra vez en función de r_0 y r_1:
  - $r_4 = r_2 - q_3 r_3 = \ldots = (1+q_3 q_2) r_0 - (q_1 + q_3 (1+q_2 q_1)) r_1$
- Así podemos continuar hasta dejar r_n en función de r_0 y r_1.
- De esta forma $r_0 u_n + r_1 v_n = 1 \rightarrow m u_n + a v_n = 1 \rightarrow$ **$v_n = a^{-1} \bmod m$**

ya en función de r_0 y r_1

Cifrado afín: Algoritmo extendido de Euclides

- Así tenemos resumiendo:
  - $r_0 = r_0 +$
    $r_1 = + r_1$
    $r_2 = r_0 - q_1 r_1$
    $r_3 = -q_2 \quad r_0 + (1+q_2q_1) r_1$
    $r_4 = (1+q_3q_2) \; r_0 - (q_1+q_3(1+q_2q_1)) \; r_1$
    .....
- Los coeficientes que acompañan a r_0 y r_1, son los u_i y v_i , que nos permiten calcular el inverso multiplicativo y cumple las siguientes reglas de recurrencias:

Para u_i, tenemos que:

- $u_0 = 1$
- $u_1 = 0$
- $u_i = u_{i-2} - q_{i-1}u_{i-1}$ para $i>1$

Para v_i, tenemos que:

- $v_0 = 0$
- $v_1 = 1$
- $v_i = v_{i-2} - q_{i-1}v_{i-1}$ para $i>1$

Cifrado afín: Algoritmo extendido de Euclides

- Así aplicando las reglas recurrencia obtenidas comprobamos que sacamos de nuevo los u_i y v_i:

Para u_i, tenemos que:

- $u_0 = 1$
- $u_1 = 0$
- $u_2 = 1 - q_1 u_1 = 1$
- $u_3 = 0 - q_2 1 = -q_2$
- $u_4 = 1 - q_3 (-q_2) = 1 + q_3 q_2$
- ……

Para v_i, tenemos que:

- $v_0 = 0$
- $v_1 = 1$
- $v_2 = 0 - q_1 1 = -q_1$
- $v_3 = 1 - q_2 (-q_1) = 1 + q_2 q_1$
- $v_4 = -q_1 - q_3 (1 + q_2 q_1)$
- ……

Cifrado afín: : Algoritmo de Euclides

- Ejemplo: mcd (26, 15) =d, y calcular el inverso multiplicativo de 15 en mod 26:
- $26 = 1 * 15 + 11 \quad \rightarrow 11 = 26 - 1 * 15 \quad (a)$
  $15 = 1 * 11 + 4 \quad \rightarrow 4 = 15 - 1 * 11 \quad (b)$
  $11 = 2 * 4 + 3 \quad \rightarrow 3 = 11 - 2 * 4 \quad (c)$
  $4 = 1 * 3 + 1 \quad \rightarrow 1 = 4 - 1 * 3 \quad (d)$
  $3 = 3 * 1 + 0$
- $1 = 4 - 1 * 3 = 4 - 1 (11 - 2 * 4) = 4 - 11 + 2 * 4 =$
  $= -11 + 3 * 4 = -11 + 3 * (15 - 1 * 11) =$
  $= -11 + 3 * 15 - 3 * 11 = 3 * 15 - 4 * 11 =$
  $= 3 * 15 - 4 * (26 - 1 * 15) = 3 * 15 - 4 * 26 + 4 * 15$
  $= 7 * 15 - 4 * 26 = 1$
- $7 * 15 - 4 * 26 = 1 \rightarrow 7 * 15 \equiv 1 \bmod 26 \rightarrow 15^{-1} = 7 \bmod 26$

Cifrado afín: fortaleza

- Podemos definir un criptosistema afín como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
  - Un conjunto finito de textos planos asignado a $P=Z_m$, tal que $x \in P$
  - Un conjunto finito de textos cifrados asignado a $C=Z_m$, tal que $y \in C$
  - Un conjunto finito de claves asignado a $K= Z_m \times Z_m^*$, tal que $k \in K$
  - Una función de cifrado (algoritmo de encriptación):
    - $E_k(x)= a*x + b \bmod m$, con $a \in Z_m^*$, y $b \in Z_m$
  - Una función de descifrado (algoritmo de desencriptación): $D_k(y)= (y - b) * a^{-1} \bmod m$.
  - **Ahora ya sabemos calcular a^{-1} mod m mediante Euclides y Euclides extendido.**
- Fortaleza del cifrado para romper por fuerza bruta $|K| = |Z_m| \times |Z_m^*|$
  - Sabemos $|Z_m|$, pero no sabemos $|Z_m^*|$:
  - Precisamente se puede demostrar que $|Z_m^*|$ es la famosa función de Euler.

Cifrado afín: fortaleza

- La función de Euler se define como el número de inversos multiplicativos que existen en el conjunto Z_m :
  - $\varphi(m) = |Z_m^*|$
  - TFA (Teorema Fundamental de la Aritmética): Todo entero positivo m > 1 puede ser representado exactamente de una única manera como un producto de potencias de números primos, salvo la multiplicidad de los mismos:
    - $m = \prod_{i=1}^{n} p_i^{e_{i_k}}$, con p_i primos distintos.
      - Por ejemplo $1000 = 2^3 \times 5^3$, con $p_1=2$, $p_2=5$, con multiplicidades $e_1=3$ y $e_2=3$.
  - Así mediante el TFA se puede definir:
    - $\varphi(m) = \prod_{i=1}^{n} p_i^{(e_i-1)}(p_i-1) = \prod_{i=1}^{n} (p_i^{e_i} - p_i^{e_i-1})$

Cifrado afín: fortaleza

- Ejemplos de fortaleza del cifrado afín:
  - Número de claves en afín para Z_{27} (español)
    - $|K| = |Z_m| \times |Z_m^*|$
    - $|Z_m^*| = \varphi(m) = \prod_{i=1}^{n} p_i^{(e_i-1)}(p_i-1) = \prod_{i=1}^{n} (p_i^{e_i} - p_i^{e_i-1})$.
    - $27 = 3^3$; $|Z_{27}^*| = \varphi(27) = 3^3 - 3^2 = 18$
    - $|K| = |Z_{27}| \times |Z_{27}^*| = 27 * 18 = 486$
  - Número de claves en afín para Z_{26} (inglés)
    - $|K| = |Z_m| \times |Z_m^*|$
    - $|Z_m^*| = \varphi(m) = \prod_{i=1}^{n} p_i^{(e_i-1)}(p_i-1) = \prod_{i=1}^{n} (p_i^{e_i} - p_i^{e_i-1})$.
    - $26 = 13 * 2$; $|Z_{26}^*| = \varphi(27) = (2-1) * (13-1) = 12$
    - $|K| = |Z_{26}| \times |Z_{26}^*| = 26 * 12 = 312$

Cifrado afín: fortaleza

- Observaciones generales que luego en el AES y RSA utilizaremos:
  - Si m = p primo
    - $|Z_p^*| = \varphi(p) = p-1$
    - En Z_p todos los elementos menos el cero tienen inverso ya que todos son coprimos con p, es decir el mcd de cada elemento de Z_p con p es 1 (excepto el cero).
    - $Z_p^* \cup 0 = Z_p$
  - $(Z_m, +) \rightarrow$ Grupo conmutativo abeliano
    $(Z_m, +, *) \rightarrow$ Anillo
    Si m = p primo $\rightarrow Z_p^* \cup 0 = Z_p$
    $(Z_p, +, *) \rightarrow$ Campo finito

Cifrado por bloques

- Hasta el momento en todos los criptosistemas anteriores se cifra carácter a carácter, es decir símbolo a símbolo:
  - $M_p = a_1\ a_2 \ldots a_n$, con $a_i \in Z_m$
  - El cifrado $E_k(a_i)$
  - $M_c = c_1\ c_2 \ldots c_n$, con $c_i \in Z_m$
- Sin embargo en los cifrados polialfabéticos puede que un determinado símbolo del alfabeto se transforme en diferentes símbolos.
- Para hacer esto una metodología habitual es cifrar los textos planos en bloques de información.

Cifrado por bloques

- Así en un cifrado por bloque tenemos:
  - $M_p = a_1\ a_2 \ldots a_n$, con $a_i \in Z_m$
  - El primer bloque cifrado $E_k(a_1\ a_2 \ldots a_i) = c_1\ c_2 \ldots c_i$
    El segundo bloque cifrado $E_k(a_{i+1}\ a_{i+2} \ldots a_{2i}) = c_{i+1}\ c_{i+2} \ldots c_{2i}$
    .....
  - $M_c = (c_1\ c_2 \ldots c_i)(c_{i+1}\ c_{i+2} \ldots c_{2i})\ldots\ldots$ con $c_i \in Z_m$

- Así es los bloques de cifrado se generará un cifrado polialfabético.
- Los siguientes cifrados cumplen esto que se comenta: Vigenère, Hill, etc….

Cifrado Vigenère

- Podemos definir un criptosistema afín como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
  - Un conjunto finito de textos planos asignado a P=Z_mx…x Z_m= (Z_m)^n tal que x ∈ P
  - Un conjunto finito de textos cifrados asignado a C= (Z_m)^n, tal que y ∈ C
  - Un conjunto finito de claves asignado a K= (Z_m)^n, tal que k ∈ K
  - Una función de cifrado (algoritmo de encriptación):
    - $E_{\bar{k}}(\bar{x})$= (x_1+k_1, …, x_n+k_n) mod m
    - Una función de descifrado (algoritmo de desencriptación): $D_{\bar{k}}(\bar{y})$= (y_1- k_1, …, y_n- k_n) mod m
- Este ya es un cifrado por bloques y polialfabético.
- La fortaleza de este cifrado por la fuerza bruta es: |K|= |(Z_m)^n|= m^n.

Cifrado de Hill

- Podemos definir un criptosistema afín como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
  - Un conjunto finito de textos planos asignado a P= Z_m tal que x ∈ P
  - Un conjunto finito de textos cifrados asignado a C= Z_m , tal que y ∈ C
  - Un conjunto finito de matrices nxn asignado a K, tal que k ∈ K, de tal forma que det(k) ∈ Z_m , y tengamos que mcd(det(k), m)=1 para que exista el descifrado.
  - Una función de cifrado (algoritmo de encriptación):

$$E_k(\bar{x}) = (x_1, \ldots, x_n) \begin{pmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{pmatrix} \bmod m$$

  - Una función de descifrado (algoritmo de desencriptación):

$$E_k(\bar{y}) = (y_1, \ldots, y_n) \begin{pmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{pmatrix}^{-1} \bmod m$$

Cifrado de Hill

- Ej. Supongamos n=2, para un tamaño de alfabeto m:

$$k = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \text{mod m}$$

$$k^{-1} = \frac{1}{\det(k)} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix} \text{mod m}$$

Existe inyección si mcd(det(k), m)= mcd($a_{11}a_{22}$ - $a_{21}a_{12}$, m)= 1

Cifrado de Hill

- Ej. Supongamos n=2, para un tamaño de alfabeto m=26:

$k = \begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$mod 26, det(k)=(77-24) mod 26=(77+2) mod 26 =1 mod 26, entonces mcd(det(k), 26)=mcd(1, 26)=1.

$k^{-1} = \frac{1}{\det(k)} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}$mod m $= \frac{1}{1} \begin{pmatrix} 7 & -8 \\ -3 & 11 \end{pmatrix}$mod m

$= \begin{pmatrix} 7 & -8 \\ -3 & 11 \end{pmatrix}$ mod m $= \begin{pmatrix} 7 & 18 \\ 23 & 11 \end{pmatrix}$ mod m.

Comprobar que esta es la matriz inversa en Z_{26}.

Cifrado de Permutación

- Podemos definir un criptosistema afín como un conjunto de 5 elementos (quintupla) { P, C, K, E_k, D_k}:
  - Un conjunto finito de textos planos asignado a $P=Z_m x \ldots x\ Z_m = (Z_m)^n$ tal que $x \in P$
  - Un conjunto finito de textos cifrados asignado a $C= (Z_m)^n$, tal que $y \in C$
  - Un conjunto finito de claves asignado a K que es un conjunto de vectores de permutación de longitud n, tal que $\pi_n \in K$
  - Una función de cifrado (algoritmo de encriptación):
    - $E_{\pi_n}(\bar{x})$, con $(x_1, \ldots, x_n)$, $x_i \in P$
    - $E_{\pi_n}{}^{-1}(\bar{y})$, con $(y_1, \ldots, y_n)$, $y_i \in C$
- Este ya es un cifrado por bloques y polialfabético.
- La fortaleza de este cifrado por la fuerza bruta es: $|K| = n!$.

Cifrados de flujo

- Por el momento hemos estudiado que el cifrado de sucesivos textos originales siempre se hace con una la misma clave:
  - $y=y_1, y_2, \ldots = e_k(y_1), e_k(y_2), \ldots$, donde cada cifrado se puede hacer símbolo a símbolo, o bloque a bloque (un bloque es más de un símbolo).
  - Pero lo interesante es la clave es la misma.
- En el cifrado de flujo se cambia el enfoque:
  - Tenemos un flujo de claves (denominada secuencia cifrante): $z_1, z_2, \ldots$
  - Esta secuencia cifrante nos permite cifrar el texto original: $x_1, x_2, \ldots$
  - Mediante la transformación: $y=y_1, y_2, \ldots = e_{z_1}(x_1), e_{z_2}(x_2), \ldots$
- El cifrado de flujo más sencillo es en el en el flujo de claves se codifica a partir de una clave que es independiente del texto original, conociéndose como el cifrado síncrono de flujo.
- En contraposición está el cifrado asíncrono, en el cual ala secuencia cifrante depende del texto a cifrar.

Cifrados de flujo

- Ej. El cifrado de Vigenère, puede ser redefinido con un cifrado de flujo:
  - Un conjunto finito de claves asignado a $K = (Z_m)^n$, tal que $k \in K$
  - $P = C = Z_m$ (Recordar que esto esta redefinido ya que antes era $(Z_m)^n$).
  - Así definimos nuestra función de cifrado como: $e_z(x) = (x+z) \bmod m$.
  - Así definimos nuestra función de descifrado como: $d_z(y) = (y-z) \bmod m$.
  - Definimos el flujo de llaves como:
  - $z_i = \begin{cases} k_i, & si \quad 1 \leq i < n \\ z_{i-n}, & si \quad i \geq n+1 \end{cases}$, donde $k = (k_1, \ldots, k_n)$.
  - Así la llave inicial es $k = (k_1, \ldots, k_n)$, y con la regla definida se genera el siguiente flujo de claves: $k_1, k_2, \ldots, k_n, k_1, k_2, \ldots, k_n, k_1, k_2, \ldots, k_n \ldots$
  - El flujo de claves es periódico de periodo n.

Cifrados de flujo

- Observación: podemos entender el cifrado de bloques como un caso particular del cifrado de flujo en el cual la secuencia cifrante es constante (como hemos visto con Vigenère).
- Se puede definir un cifrado de flujo periódico de periodo d si $z_{i+d} = z_i$.
- Como hemos dicho el cifrado de Vigenère se puede ver como un cifrado de flujo de periodo n.
- Muy habitualmente hoy en día los cifrados en flujo en términos de alfabetos binarios, 0 y 1.
- Es decir $P=C=Z_2$. En esta situación las operaciones se realizan en aritmética modular 2.

Cifrados de flujo

- Así estas operaciones son:
  - $e_z(x)=(x+z) \bmod 2$, con $x, z \in Z_2$.
  - $d_z(y)=(y+z) \bmod 2$, con $y, z \in Z_2$.
- En muchas ocasiones estos cifrados en flujo se implementan mediante registros de desplazamiento:
  - LFSR: *linear feedback shift register*.
  - NLFSR: *non linear feedback shift register*.
- Un registro de desplazamiento son varias celdas de memoria conectadas entre sí, donde cada celdas almacena un bit:
  - El valor de los bits de todas estas celdas determinan un estado del registro.
  - El estado del registro se puede cambiar mediante generalmente una señal del reloj (se puede cambiar al ritmo de un reloj).
  - Cuando se cambia el estado del registro el nuevo estado del registro, se hace desplazando los bits de cada celda a la celda de su derecha.
    - El bit de más a la derecha sale del registro, y un nuevo bit entra en la celda de más a la izquierda.
  - De esta manera se va generando una secuencia cifrante, para ser utilizada en un cifrado de flujo.

Cifrados de flujo

- Vamos a ver un ejemplo de como generar una secuencia cifrante en este contexto, mediante las llamadas reglas de recurrencia de orden n.
- Los registros de desplazamiento permiten generar secuencias cifrantes de periodo muy grande con buenas propiedades estadísticas de aleatoriedad.
- Además son muy eficientes ya que suelen estar implementados a nivel de *hardware*.

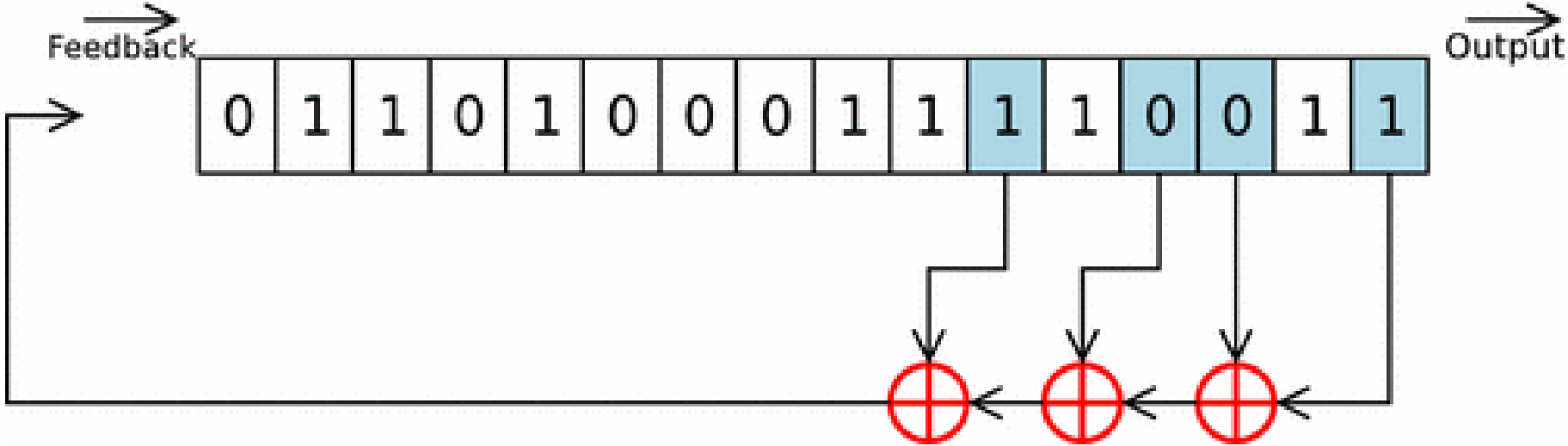


Imagen extraída https://es.wikipedia.org/wiki/LFSR

Cifrados de flujo

- Un ejemplo de la generación de una secuencia cifrante con una recurrencia lineal de grado n:
  - Inicializamos el registro de desplazamiento: k = (k_1, …, k_n), así inicializamos cada celda por $z_i = k_i$.
  - Supongamos la siguiente regla de recurrencia lineal de grado n:

$$z_i = \sum_{j=0}^{n-1} c_i z_{i+j} \ mod\ 2, i \geq 1, y\ c = (c_0, \ldots, c_{n-1})$$

  - Cada termino depende en general de los n términos anteriores.
  - Realmente la clave de inicialización en la secuencia cifrante es

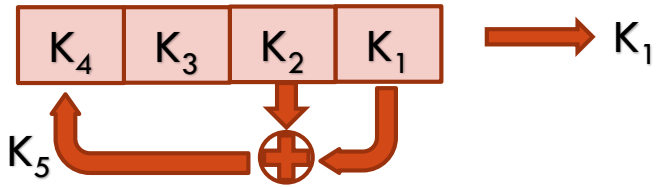
    K= (k_1, …, k_n, c_0, …, c_{n-1}).

Cifrados de flujo

- Si suponemos K = (k_1, …, k_n, c_0, …, c_{n-1}) = (1, 0, 0, 0, 1, 1, 0, 0), y aplicando la regla de recurrencia:

$$z_i = \sum_{j=0}^{n-1} c_i z_{i+j} \ mod\ 2, i \geq 1, y\ c = (c_0, \ldots, cn_{-1})$$

- Obtenemos $z_{i+4} = z_i + z_{i+1} \text{mod } 2$, que genera la siguiente secuencia cifrante:
  - 100010011010……
  - Para casa ver cuál es el periodo de la secuencia cifrante (es pequeño).



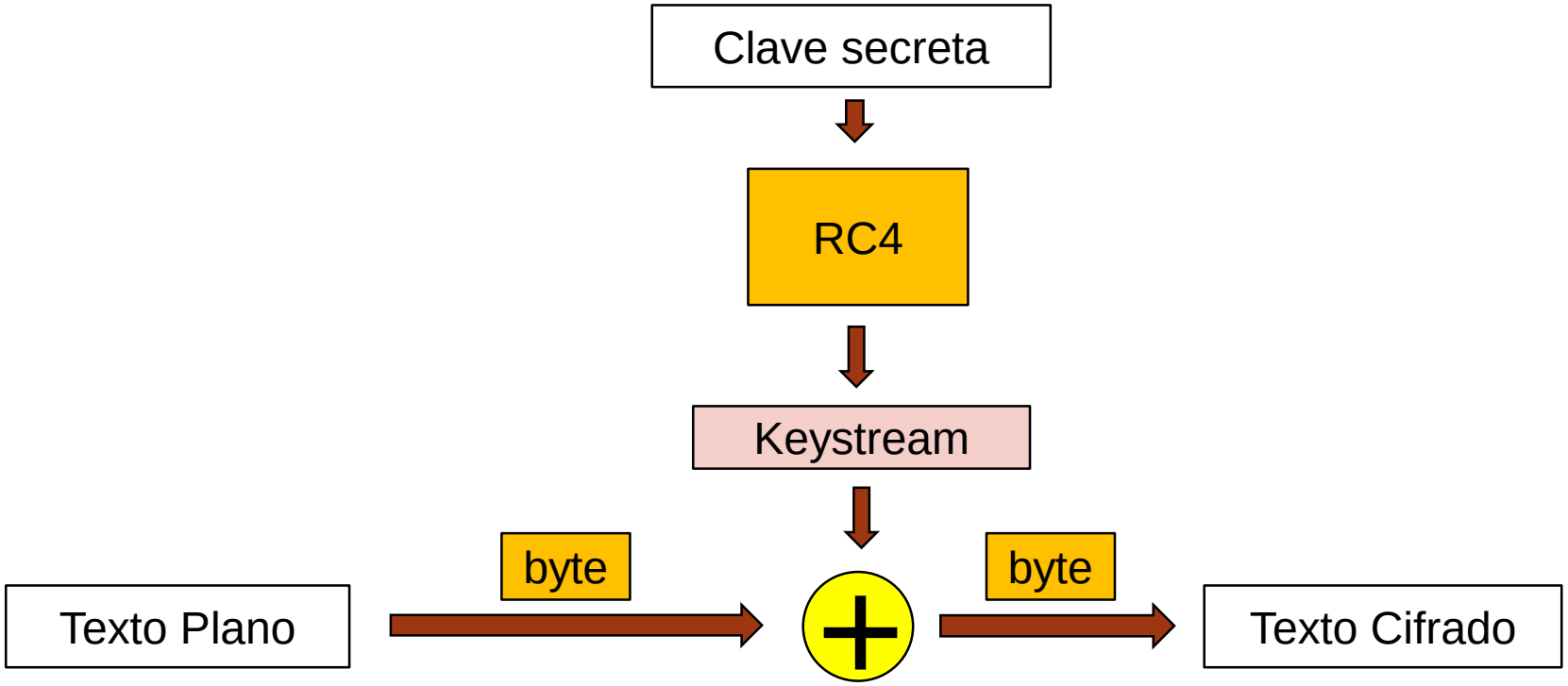
Cifrados de flujo

- Esto es un ejemplo sencillo, pero los registros de desplazamiento se pueden combinar de manera compleja:
  - **ACHTERBAHN-128/80** , porque admite las longitudes de clave de 80 bits y 128 bits.
  - Su significado es montaña rusa
  - Implementado con 2188 puertas NAND (ACHTERBAHN-80).
  - Es un cifrado de flujo síncrono.
  - Se basa en la filosofía LFSR, pero de manera no lineal es decir en NLFSR.
  - Este tipo de cifrados se suelen utilizar en el área de telecomunicaciones.
  - La unidad principal de información es el bit (algoritmo orientado a operaciones con bits).

Cifrados de flujo

- Existen otros muchos tipos de cifrados de flujos, orientados por ejemplo a las operaciones a nivel byte.
- Un ejemplo el RC4, que es el ejemplo típico de cifrado por flujo.
- Diseñado en 1987 por Ron Rivest para RSA Security (división de seguridad de EMC Corporation).
- ARCFOUR, ARC4 o Alleged-RC4 en su implementación no oficial, ya que RSA Security nunca ha liberado el algoritmo de su RC4.
  - Inicialmente el algoritmo era un secreto registrado, pero en septiembre de 1994 una descripción del algoritmo fue posteada anónimamente en una lista de correo de Cypherpunks.
- Clave variable con operaciones orientados a bytes.
- Basado en el uso de permutaciones aleatorias.
- Ocho a dieciséis operaciones de código máquina por byte de salida.
- Hoy en día en desuso, principalmente a partir del 2015, cuando dejo de ser un estándar en los navegadores en el Secure Sockets Layer / Transport Layer Security (SSL /TLS).

Cifrados de flujo: RC4



Producto de criptosistemas: incremento de fortaleza???

- Existen diversas metodologías para aumentar la fortaleza de un criptosistema, y una de ellas es combinar criptosistemas (producto de criptosistemas), que fue propuesto por Shannon.
- Se define el producto de criptosistemas:
  $S_1 \times S_2 = \{P, C, K_1, E_{k1}, D_{k1}\} \times \{P, C, K_2, E_{k2}, D_{k2}\} =$
  $\{P, C, K_1 \times K_2, E_{k2}(E_{k1}(x)), D_{k1}(D_{k2}(y))\}$.
- Ej. 1 el producto de dos criptosistemas desplazamiento:
  - $y = x + k_1 \bmod m$
    $y = x + k_2 \bmod m$
  - Si los componemos los dos criptosistemas:
  - $E_{k2}(E_{k1}(x)) = E_{k2}(x + k_1) = x + (k_1 + k_2) \bmod m$
  - Pero k_1 y $k_2 \in Z_m$. Por lo tanto $(k_1 + k_2) \in Z_m$. Así $|k_1 + k_2| = m$.
  - El producto no aumenta la fortaleza del criptosistema resultante.
    - A este tipo de criptosistemas se les suele llamar idempotentes: $S^2=S$.

Producto de criptosistemas: incremento de fortaleza???

- Ej. 2 el producto de dos criptosistemas desplazamiento y multiplicativo:
  - $y = x + b \bmod m$
    $y = ax \bmod m$
  - Si los componemos los dos criptosistemas:
  - $E_{k2}(E_{k1}(x)) = E_{k2}(x + b) = ax + ab \bmod m$
  - Pero a y b $\in Z_m$. Por lo tanto (ab) = c $\in Z_m$.
  - El producto aumenta la fortaleza del criptosistema resultante
    $y = ax + c \bmod m$, que es el criptosistema afín que tiene más fortaleza que cada uno de los individuales.
- Ej. 3 producto de sustitución y permutación que da origen a la criptografía moderna: FEISTEL, DES, AES, etc.

Criptoanálisis

- Aquí vamos a partir de la base que los criptoanalistas conocen el algoritmo de cifrado.
- Si no se conoce es más difícil, pero vamos a partir de las reglas de Kerckhoffs:
  - El algoritmo de cifrado es público.
  - La fortaleza reside en el secreto de la llave.
- Vamos a distinguir entre varios tipos de ataques en función del conocimiento que tengamos sobre
  - los textos planos
  - Textos cifrados
  - Parejas testo plano-cifrado
  - Etc.

Criptoanálisis

- Así los tipos de ataque pueden ser:
  - A. Ataque con sólo texto cifrado:
    - Algoritmo de cifrado.
    - Solo textos cifrados a descifrar.
  - B. Ataque de texto claro conocido:
    - Algoritmo de cifrado.
    - Textos cifrados a descifrar.
    - Una o más parejas de texto claro – texto cifrado con la clave secreta.
  - C. Ataque con texto claro elegido:
    - Algoritmo de cifrado.
    - Textos cifrados a descifrar.
    - Una o más parejas de texto claro (pero que se ha elegido de manera adecuada) – texto cifrado. Esta pareja ha sido cifrada con la clave secreta de los textos a criptoanalizar.
  - D. Ataque con texto cifrado elegido:
    - Algoritmo de cifrado.
    - Textos cifrados a descifrar.
    - Una o más parejas de texto claro – texto cifrado (pero que se ha elegido de manera adecuada). Esta pareja ha sido cifrada con la clave secreta de los textos a criptoanalizar.
  - E. Ataque con texto elegido:
    - Algoritmo de cifrado.
    - Textos cifrados a descifrar.
    - Una o más parejas de texto claro – texto cifrado (pero que se ha elegido de manera adecuada). Esta pareja ha sido cifrada con la clave secreta de los textos a criptoanalizar.
    - Una o más parejas de texto claro – texto cifrado (pero que se ha elegido de manera adecuada). Esta pareja ha sido cifrada con la clave secreta de los textos a criptoanalizar.

Criptoanálisis del cifrado por desplazamiento

- Se rompe inmediatamente con B, C, D, y E.
- Con A necesitamos hacer un análisis de frecuencias, ya que no todos los símbolos del lenguaje se usan con la misma frecuencia.
- En general el análisis de frecuencias se puede utilizar para todo tipo de ataque A (la letra más frecuente en el cifrado corresponde a la más frecuente del original y así sucesivamente).

	Castellano	Inglés
A	11.96	8.04
B	0.92	1.54
C	2.92	3.06
D	6.87	3.99
E	16.78	12.51
F	0.52	2.30
G	0.73	1.96
H	0.89	5.49
I	4.15	7.26
J	0.30	0.16
K	0.0	0.67
L	8.37	4.14
M	2.12	2.53
N	7.01	7.09
O	8.69	7.60
P	2.77	2.00
Q	1.53	0.11
R	4,94	6.12
S	7.88	6.54
T	3.31	9.25
U	4.80	2.71
V	0.39	0.99
W	0.0	1.92
X	0.06	0.19
Y	1.54	1.73
Z	0.15	0.19

Criptoanálisis del cifrado afín

- Se rompe inmediatamente con B, C, D, y E.
- Para ello se resuelve el sistema siguiente de dos ecuaciones con dos incógnitas, que son precisamente la clave, para una pareja de texto claro- cifrado conocido (x_1, y_1), (x_2, y_2):
  - $y_1 = a\ x_1 + b \bmod m$
  - $y_2 = a\ x_2 + b \bmod m$
  - Recordar que para resolverlo la división es multiplicar por el inverso multiplicativo.
- Con A necesitamos hacer un análisis de frecuencias, ya que no todos los símbolos del lenguaje se usan con la misma frecuencia.

Criptoanálisis del cifrado afín

- Ej. En el caso A. Supongamos que se conoce que se ha cifrado con un afín el siguiente texto en inglés:

  FMXVEDKAPHFERBNDKRXRSREFMORU

  DSDKDVSHVUFEDKAPRKDLYEVLRHHRH

- Calculamos la frecuencia de los símbolos:
- R(8), D(7), E(5), H(5), H(5), F S y V (4) …….
- Lanzamos hipótesis:
  - La E se trasforma en R y la T en D:
    - 17 =a 4 + b mod m
    - 3 =a 19 + b mod m
    - *Se resuelve y sale a* = 6, *b* = 19, pero no valen porque el mcd(6,26)=2.

Criptoanálisis del cifrado afín

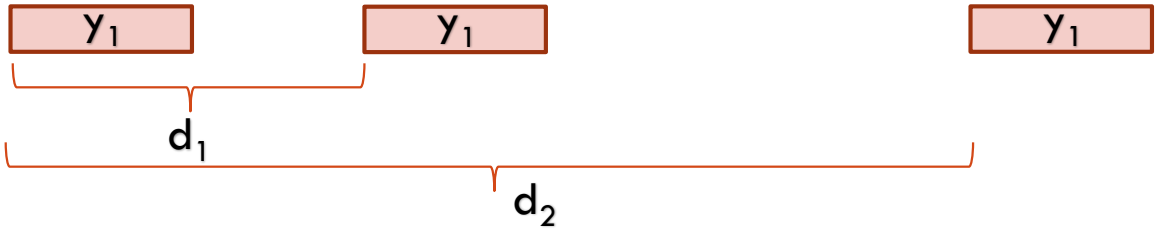
- R(8), D(7), E(5), H(5), H(5), F S y V (4) …….
- Lanzamos otra hipótesis:
  - La E se trasforma en R y la T en E:
    - *Se resuelve y sale a* = 13, pero no vale de nuevo porque el mcd (13,26)=2.
- Lanzamos otra hipótesis:
  - La E se trasforma en R y la T en K:
    - *Se resuelve y sale a* = 3, y b = 5, que es válido ya que el mcd(3,26)=1.
    - Ahora probamos a descifrar con esa claves y observar el resultado:
    - Así suponemos y =3 x + 5 mod 26 y para descifrar la transformación inversa es x= (y-5) 3^{-1}. Es decir x=9y-19 mod 26, que nos da: algorithmsarequitegeneraldefinitionsofarithmeticprocesses

Criptoanálisis del cifrado Vigenère

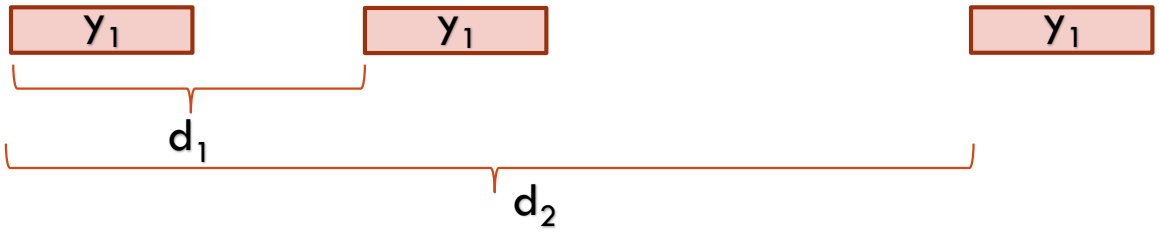
- Pasamos ya al criptoanálisis de los cifrados polialfabéticos que es más complicado (los anteriores eran cifrados mono alfabéticos).
- Se realiza en dos pasos:
  - **Primero** se detecta cual es la longitud de la clave, es decir n (existen dos métodos):
    - Test de Kasiski.
    - Índice de coincidencia.
  - **Segundo** se detecta cada una de las componentes de la clave mediante un método basado en índice de coincidencia.
  - Así vamos a describir cada uno de los métodos con detalle a continuación.

Criptoanálisis del cifrado Vigenère (Test de Kasiski)

- Descrito por Friderich Kasiski en 1863, aunque fue descubierto previamente por Charles Baggage en 1854.
- Se basa en una idea muy sencilla:
  - Dos segmentos de texto originales idénticos podrían ser cifrados obteniéndose el mismo texto cifrado si coincidieran con la misma parte de la clave.
- Así supongamos que tenemos la cadena cifrada como y_1, que ha sido cifrada con la misma parte de la clave:



Criptoanálisis del cifrado Vigenère (Test de Kasiski)



- Así se cumple que:
  - $d_1 = q_1 n$
    $d_2 = q_2 n$
    .....
- La distancias son múltiplos de la longitud de la clave utilizada, o lo que es lo mismo que:

  n| d_1, d_2, ..., para coger el mínimo n entonces
  n| mcd (d_1, d_2, ...)

Criptoanálisis del cifrado Vigenère (Test de Kasiski)

- Ej. Kasiski: Supongamos el siguiente texto que viene del inglés cifrado con Vigenère. Vamos a averiguar, cual es el tamaño de la calve utilizado:

CHREEVOAHMAERATBIAXXWTNXBEEOPHBSBQMQEQERBW
RVXUOAKXAOSXXWEAHBWGJMMQMNKGRFVGXWTRZXWIAK
LXFPSKAUTEMNDCMGTSXMXBTUIADNGMGPSRELXNJELX
VRVPRTULHDNQWTWDTYGBPHXTFALJHASVBFXNGLL**CHR**
ZBWELEKMSJIKNBHWRJGNMGJSGLXFEYPHAGNRBIEQJT
AMRVLCRREMNDGLXRRIMGNSNRW**CHR**QHAEYEVTAQEBBI
PEEWEVKAKOEWADREMXMTBHH**CHR**TKDNVRZ**CHR**CLQOHP
WQAIIWXNRMGWOIIFKEE

- La cadena de texto cifrado CHR aparece en cinco lugares del texto cifrado, comenzando en las posiciones 1, 166, 236, 276 y 286. Las distancias desde la primera aparición hasta la otras ocurrencias son (respectivamente) 165, 235, 275 y 285. El mcd de estos cuatro enteros es 5, por lo que es muy probablemente la longitud de la palabra clave sea cinco caracteres.

Criptoanálisis del cifrado Vigenère (Índice de coincidencia)

- El índice de coincidencia fue escrito por Wolfe Friedman en 1920.
- Dado el vector de caracteres, $\bar{x}= (x_1, \ldots, x_l)$, se define el índice de coincidencia, $I_c(\bar{x})$, como la probabilidad de que dos elementos escogidos al azar dentro de la cadena de l caracteres coincidan.
- Vamos a estimar esta medida de probabilidad.
- Supongamos que denotamos a las frecuencia de los caracteres A, B, C, …, por $f_0, f_1, f_2, \ldots, f_{m-1}$.
- Para cada $i \in [0, m-1]$ hay $\binom{f_i}{2}$ formas de elegir dos elementos iguales a i.
- Recordemos lo significa el número combinatorio $\binom{n}{r} = \frac{n!}{r!(n-r)!}$
  - De cuantas formas se pueden elegir un comité de 5 personas de un total de 9 personas sin atender a la ordenación de los mismo: $\binom{9}{5} = \frac{9!}{5!(9-5)!} = 126.$

Criptoanálisis del cifrado Vigenère (Índice de coincidencia)

- Si contemplamos todos los caracteres tenemos

$$\sum_{i=0}^{m-1} \binom{f_i}{2}$$

- Que representa el número de casos posibles en los que podemos elegir dos caracteres iguales entre un total de m caracteres del alfabeto.
- ¿Cuál sería el caso en el que siempre tenemos probabilidad de escoger 2 caracteres iguales?
  - Cuando el vector de caracteres, $\bar{x}= (x_1, \ldots, x_l)$, tiene l caracteres iguales ese es el caso en el que el índice de coincidencia es máximo.
  - Por lo tanto el número de formas que tengo de seleccionar parejas de dos caracteres iguales en ese vector de caracteres iguales es $\binom{l}{2}$.

Criptoanálisis del cifrado Vigenère (Índice de coincidencia)

- Asi podemos concluir que el índice de coincidencia, $I_c(\bar{x})$ es:

$$I_c(\bar{x}) = \frac{\sum_{i=0}^{m-1}\binom{f_i}{2}}{\binom{l}{2}} = \frac{\sum_{i=0}^{m-1}\frac{f_i!}{2!(f_i-2)!}}{\frac{l!}{2!(l-2)!}} = \frac{\sum_{i=0}^{m-1}\frac{f_i(f_i-1)(f_i-2)!}{2!(f_i-2)!}}{\frac{l(l-1)(l-2)!}{2!(l-2)!}} = \frac{\sum_{i=0}^{m-1} f_i(f_i-1)}{l(l-1)} =$$

$$\sum_{i=0}^{m-1} P_i \frac{f_i-1}{l-1} \cong \sum_{i=0}^{m-1} {P_i}^2$$

- Asi podemos concluir que el índice de coincidencia, $I_c(\bar{x})$, cuando $\lim_{l\to\infty}$, es decir cuando $f_i - 1 \approx f_i$ y $l - 1 \approx l$, se convierte en:

$$\sum_{i=0}^{m-1} {P_i}^2$$

Criptoanálisis del cifrado Vigenère (Índice de coincidencia)

- ¿Para que vale esta métrica en el criptoanálisis?.
- Primero vamos a ver diferencia en esta medida para un lenguaje con estructura y un lenguaje aleatorio.
  - Para el inglés P_0, P_1, P_2, …, P_{25} tenemos que $\lim_{l\to\infty} I_c(\bar{x}) \cong \sum_{i=0}^{m-1} {P_i}^2 = 0{,}065$.
  - Calcular $I_c(\bar{x})$ para el español.
  - Para un lenguaje aleatorio de m símbolos tenemos $P_0=P_1=P_2= \ldots =P_{m-1} =\frac{1}{m}$ tenemos que $\lim_{l\to\infty} I_c(\bar{x}) \cong \sum_{i=0}^{m-1} \left(\frac{1}{m}\right)^2 = 0{,}038$ (para m=26).
- Entonces vemos que hay una clara diferencia para esta medida entre un leguaje estructurado y un lenguaje aleatorio.
- Este principio lo vamos a utilizar para calcular el tamaño de la clave en el criptoanálisis de Vigenère.

Criptoanálisis del cifrado Vigenère (Índice de coincidencia)

- Supongamos que tenemos el texto cifrado por Vigenère:
  - $M_c = y_1\ y_2 \dots y_l$, con $y_i \in Z_m$
- Supongamos que ese texto se ha cifrado con un clave de tamaño n:
  - $M_c = y_1\ y_2 \dots y_n\ y_{n+1}\ y_{n+2} \dots y_{2n}\ y_{2n+1}\ y_{2n+2} \dots y_{3n} \dots y_l$, con $y_i \in Z_m$
- Podemos organizarlo de la siguiente forma:
  - $y_1 \quad y_2 \quad \dots\ y_n$
    $y_{n+1} \quad y_{n+2} \quad \dots\ y_{2n}$
    $y_{2n+1}\ y_{2n+2} \quad \dots\ y_{3n}$
    .....
- Podemos definir los siguientes vectores con estructura de lenguaje, con el índice de coincidencia es diferente al de un lenguaje aleatorio.

Criptoanálisis del cifrado Vigenère (Índice de coincidencia)

- Estos vectores son los siguientes:
  - $\bar{x}_1 = (y_1, y_{n+1}, y_{2n+1}, \ldots)$
    $\bar{x}_2 = (y_2, y_{n+2}, y_{2n+2}, \ldots)$
    ……
    $\bar{x}_n = (y_n, y_{2n}, y_{3n}, \ldots)$
- Si n es el tamaño de bloque con el que se ha cifrado, todos estos vectores tienen estructura de lenguaje y por tanto su coincidencia es diferente al de un lenguaje aleatorio.
- Si por ejemplo el lenguaje original es el Inglés entonces:
  - $I_c(\bar{x}_1) \cong I_c(\bar{x}_2) \cong I_c(\bar{x}_3) \cong \ldots \cong I_c(\bar{x}_n) \cong 0{,}065$.
- Esto solo pasa si n es la longitud de la clave usada, si no es así es un lenguaje aleatorio, y por tanto su valor se aproxima a:
  - $I_c(\bar{x}_1) \cong I_c(\bar{x}_2) \cong I_c(\bar{x}_3) \cong \ldots \cong I_c(\bar{x}_n) \cong 0{,}038$.

Criptoanálisis del cifrado Vigenère (Test de Kasiski)

- Ej. Índice de coincidencia: Supongamos el siguiente texto que viene del inglés cifrado con Vigenère. Vamos a averiguar, cual es el tamaño de la calve utilizado:

CHREEVOAHMAERATBIAXXWTNXBEEOPHBSBQMQEQERBW
RVXUOAKXAOSXXWEAHBWGJMMQMNKGRFVGXWTRZXWIAK
LXFPSKAUTEMNDCMGTSXMXBTUIADNGMGPSRELXNJELX
VRVPRTULHDNQWTWDTYGBPHXTFALJHASVBFXNGLLCHR
ZBWELEKMSJIKNBHWRJGNMGJSGLXFEYPHAGNRBIEQJT
AMRVLCRREMNDGLXRRIMGNSNRWCHRQHAEYEVTAQEBBI
PEEWEVKAKOEWADREMXMTBHHCHRTKDNVRZCHRCLQOHP
WQAIIWXNRMGWOIIFKEE

- Veamos si el cálculo de índices de coincidencia da la misma conclusión que el ejemplo anterior con Kasiski. Con la hipótesis de tamaño de bloque n = 1, el único índice de coincidencia sale 0.045. Con n = 2, los dos índices son 0,046 y 0,041. Con n = 3, obtenemos 0.043, 0.050, 0.047. Con n = 4, tenemos índices 0.042, 0.039, 0.046, 0.040. Finalmente, con n = 5, obtenemos los valores **0,063**, **0,068**, **0,069**, **0,061** y **0,072** ≅ **0,065**. Esto también proporciona una fuerte evidencia de que la longitud de la palabra clave es cinco.

Criptoanálisis del cifrado Vigenère (Cálculo de subclaves)

- Una vez que ya sabemos la longitud de la clave, ya sea bien por el test de Kasiski o índice de coincidencia, vamos a estimar ahora la propia clave de cifrado: $\bar{k} = (k_1, \ldots, k_n)$.
- Para ello nos vamos a basar en el índice de coincidencia.
- Supongamos que denotamos a las frecuencia de los caracteres de nuestro texto cifrado por A, B, C, …, por $f_0, f_1, f_2, \ldots, f_{m-1}$ como ya habíamos comentado.
- Por otro lado sabemos que los símbolos de nuestro lenguaje tiene tienen una probabilidad asociada: $P_0, P_1, P_2, \ldots, P_{m-1}$, que es la probabilidad del lenguaje.

Criptoanálisis del cifrado Vigenère (Cálculo de subclaves)

- Así si k_i, fuese la componente i de la clave de cifrado, entonces la siguiente cantidad será muy cercana a la probabilidad del lenguaje:
- Suponiendo que $\frac{l}{n}$ es la longitud de las cadenas de cualquiera de las n cadenas $\bar{x}_i$ que habíamos calculado antes:
  - $\bar{x}_1 = (y_1, y_{n+1}, y_{2n+1}, \ldots)$
    $\bar{x}_2 = (y_2, y_{n+2}, y_{2n+2}, \ldots)$
    ……
    $\bar{x}_n = (y_n, y_{2n}, y_{3n}, \ldots)$
- Que provenían del texto cifrado
  - $M_c = y_1\ y_2 \cdots y_n\ y_{n+1}\ y_{n+2} \cdots y_{2n}\ y_{2n+1}\ y_{2n+2} \cdots y_{3n} \cdots y_l$, con $y_i \in Z_m$
- Así podemos calcular la probabilidad de los caracteres en el vector $\bar{x}_i$ como:
- $\frac{f_0}{\frac{l}{n}}, \frac{f_1}{\frac{l}{n}}, \frac{f_2}{\frac{l}{n}}, \ldots, \frac{f_{m-1}}{\frac{l}{n}}$.

Criptoanálisis del cifrado Vigenère (Cálculo de subclaves)

- Pero recordar que este $\bar{x}_i$ es un cifrado de desplazamiento ya que está formado por todos los caracteres que se han cifrado con la subclave k_i.
- Por tanto la probabilidad desplazada:
  - $\frac{f_{0-k_i}}{\frac{l}{n}}, \frac{f_{1-k_i}}{\frac{l}{n}}, \frac{f_{2-k_i}}{\frac{l}{n}}, \ldots, \frac{f_{(m-1)-k_i}}{\frac{l}{n}},$
- Será muy cercana a las probabilidades reales de los símbolos del lenguaje: $P_0, P_1, P_2, \ldots, P_{m-1}$., que figuran en la transparencia 45.
- Por tanto podemos definir la siguiente cantidad $M(k_i)$ para calcular cada una de las subclaves:
- $M(k_i) = \sum_{j=0}^{m-1} P_j \frac{f_{j-k_i}}{\frac{l}{n}}$, cuando probabilidad desplazada coincida con las subclave k_i utilizada entonces: $M(k_i) = \sum_{i=0}^{m-1} {P_i}^2 \cong 0{,}065$ (para el inglés).

P_j

Criptoanálisis del cifrado Vigenère (Cálculo de subclaves)

- Se puede demostrar que $\sum_{i=0}^{m-1} P_i P_{i-j} = \sum_{i=0}^{m-1} P_i P_{i+j}$, por tanto la cantidad anterior se puede expresar como:
  - $M(k_i) = \sum_{j=0}^{m-1} P_j \frac{f_{j+k_i}}{\frac{l}{n}}$.

- Demostrar para casa que $\sum_{i=0}^{m-1} P_i P_{i-j} = \sum_{i=0}^{m-1} P_i P_{i+j}$.
- Cuando la probabilidad desplazada no sea la adecuada, es decir el k_i considerado no es una posible subclave con la que se ha cifrado el texto plano, entonces el valor de esa cantidad $M(k_i)$ tenderá el índice de coincidencia de un lenguaje aleatorio.

Criptoanálisis del cifrado Vigenère (Test de Kasiski)

- Ej. Cálculo de subclaves: Supongamos el siguiente texto que viene del inglés cifrado con Vigenère. Vamos a averiguar, cual son la subclaves sabiendo que n=5:

CHREEVOAHMAERATBIAXXWTNXBEEOPHBSBQMQEQERBW
RVXUOAKXAOSXXWEAHBWGJMMQMNKGRFVGXWTRZXWIAK
LXFPSKAUTEMNDCMGTSXMXBTUIADNGMGPSRELXNJELX
VRVPRTULHDNQWTWDTYGBPHXTFALJHASVBFXNGLLCHR
ZBWELEKMSJIKNBHWRJGNMGJSGLXFEYPHAGNRBIEQJT
AMRVLCRREMNDGLXRRIMGNSNRWCHRQHAEYEVTAQEBBI
PEEWEVKAKOEWADREMXMTBHHCHRTKDNVRZCHRCLQOHP
WQAIIWXNRMGWOIIFKEE

- Para ello vamos a utilizar la cantidad estudiada anteriormente:
  - $M(k_i) = \sum_{j=0}^{m-1} P_j \frac{f_{j+k_i}}{\frac{l}{n}}$.

Criptoanálisis del cifrado Vigenère (Test de Kasiski)

i	$M(k_i) = \sum_{j=0}^{m-1} P_j \frac{f_{j+k_i}}{\frac{l}{n}}$
1	.035 .031 .036 .037 .035 .039 .028 .028 .048 **.061** .039 .032 .040 .038 .038 .045 .036 .030 .042 .043 .036 .033 .049 .043 .042 .036
2	**.069** .044 .032 .035 .044 .034 .036 .033 .029 .031 .042 .045 .040 .045 .046 .042 .037 .032 .034 .037 .032 .034 .043 .032 .026 .047
3	.048 .029 .042 .043 .044 .034 .038 .035 .032 .049 .035 .031 .035 **.066** .035 .038 .036 .045 .027 .035 .034 .034 .036 .035 .046 .040
4	.045 .032 .033 .038 **.060** .034 .034 .034 .050 .033 .033 .043 .040 .033 .029 .036 .040 .044 .037 .050 .034 .034 .039 .044 .038 .035
5	.034 .031 .035 .044 .047 .037 .043 .038 .042 .037 .033 .032 .036 .037 .036 .045 .032 .029 .044 **.072** .037 .027 .031 .048 .036 .037

Criptoanálisis del cifrado Vigenère (Test de Kasiski)

- Por lo tanto de esta tabla podemos deducir que es probable que la clave sea $\bar{k} = (9, 0, 13, 4, 19)$ y, por lo tanto, la palabra clave probablemente sea JANET.
- Así con esta clave obtenemos para el descifrado:

thealmondtreewasintentativeblossomthedayswerelonger oftenendingwithmagnificenteveningsofcorrugatedpinkskiesthe huntingseasonwasoverwithhoundsandgunsputawayforsix monthsthevineyardswerebusyagainasthewellorganizedfarmers treatedtheirvinesandthemorelackadaisicalneighborshurriedto dothepruningtheyshouldhavedoneinnovember

Criptoanálisis del cifrado Hill

- Se rompe inmediatamente con B, C, D, y E, siempre que tengamos n tuplas de parejas de texto planos y cifrados, siendo n la dimensión de la matriz clave.
- De esta forma:
  - $M_p = x_1\ x_2 \dots x_n\ x_{n+1}\ x_{n+2} \dots x_{2n} \dots x_{(n-1)n+1}\ x_{(n-1)n+2} \dots x_{nn}$, con $x_i \in Z_m$
  - El primer bloque cifrado $E_k(x_1\ x_2 \dots x_n) = (y_1\ y_2 \dots y_n)$
    El segundo bloque cifrado $E_k(x_{i+1}\ x_{i+2} \dots x_{2n}) = (y_{n+1}\ y_{n+2} \dots y_{2n})$

    .....

  - $M_c = y_1\ y_2 \dots y_n\ y_{n+1}\ y_{n+2} \dots y_{2n} \dots y_{(n-1)n+1}\ y_{(n-1)n+2} \dots y_{nn}$, con $y_i \in Z_m$

Criptoanálisis del cifrado Hill

- Así tenemos:

$$\begin{pmatrix} y_1 & \cdots & y \\ \vdots & \ddots & \vdots \\ y_{(n-1)n+1} & \cdots & y_{nn} \end{pmatrix} = \begin{pmatrix} x_1 & \cdots & x_n \\ \vdots & \ddots & \vdots \\ x_{(n-1)n+1} & \cdots & x_{nn} \end{pmatrix} \begin{pmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{pmatrix} \bmod m$$

Por tanto:

$$\begin{pmatrix} k_{11} & \cdots & k_{1n} \\ \vdots & \ddots & \vdots \\ k_{n1} & \cdots & k_{nn} \end{pmatrix} = \begin{pmatrix} x_1 & \cdots & x_n \\ \vdots & \ddots & \vdots \\ x_{(n-1)n+1} & \cdots & x_{nn} \end{pmatrix}^{-1} \begin{pmatrix} y_1 & \cdots & y \\ \vdots & \ddots & \vdots \\ y_{(n-1)n+1} & \cdots & y_{nn} \end{pmatrix} \bmod m$$

POSIBLE Tarea de Evaluación Continúa: Criptoanálisis de los LFSR y NLFSR.