UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

Matemática Discreta
Y
Criptografía

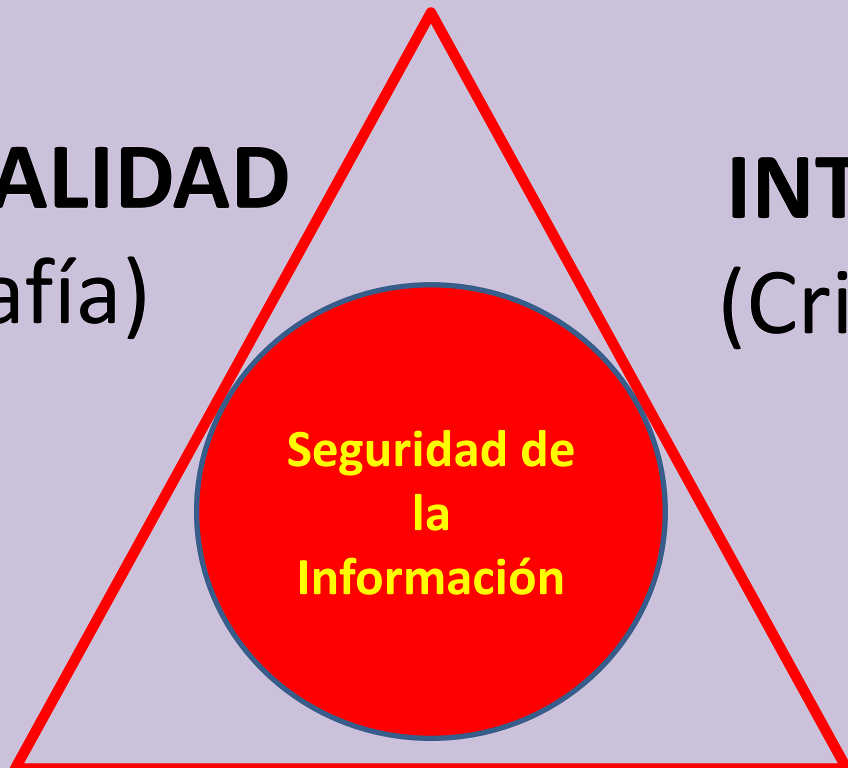
CRIPTOSISTEMA
R.S.A.

Marcelo Cipriano

Familia de estándares ISO 2700 y anteriores

CONFIDENCIALIDAD
(Criptografía)

INTEGRIDAD
(Criptografía)



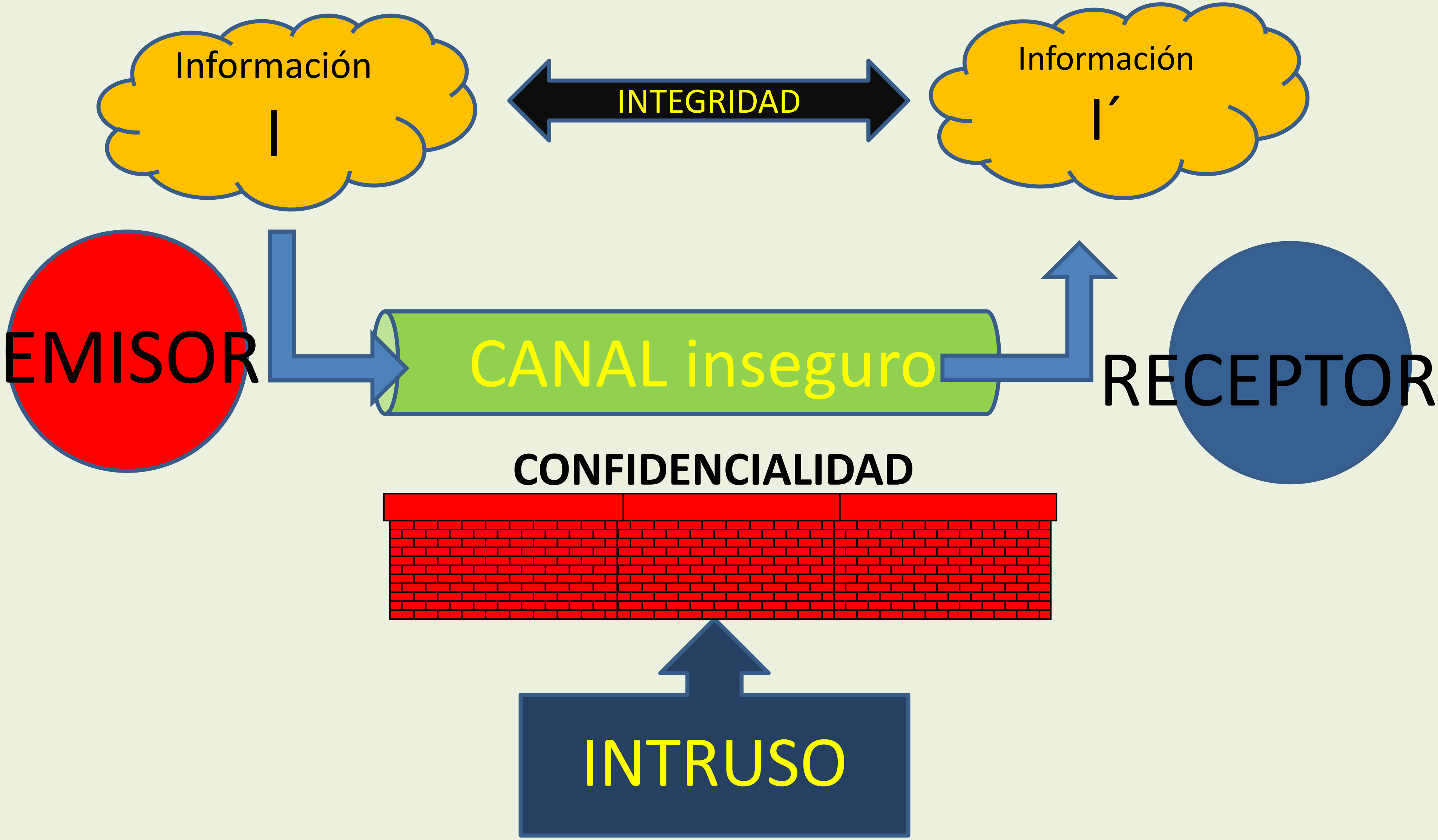
DISPONIBILIDAD
(técnicas administrativas e informáticas)

Seguridad de la Información

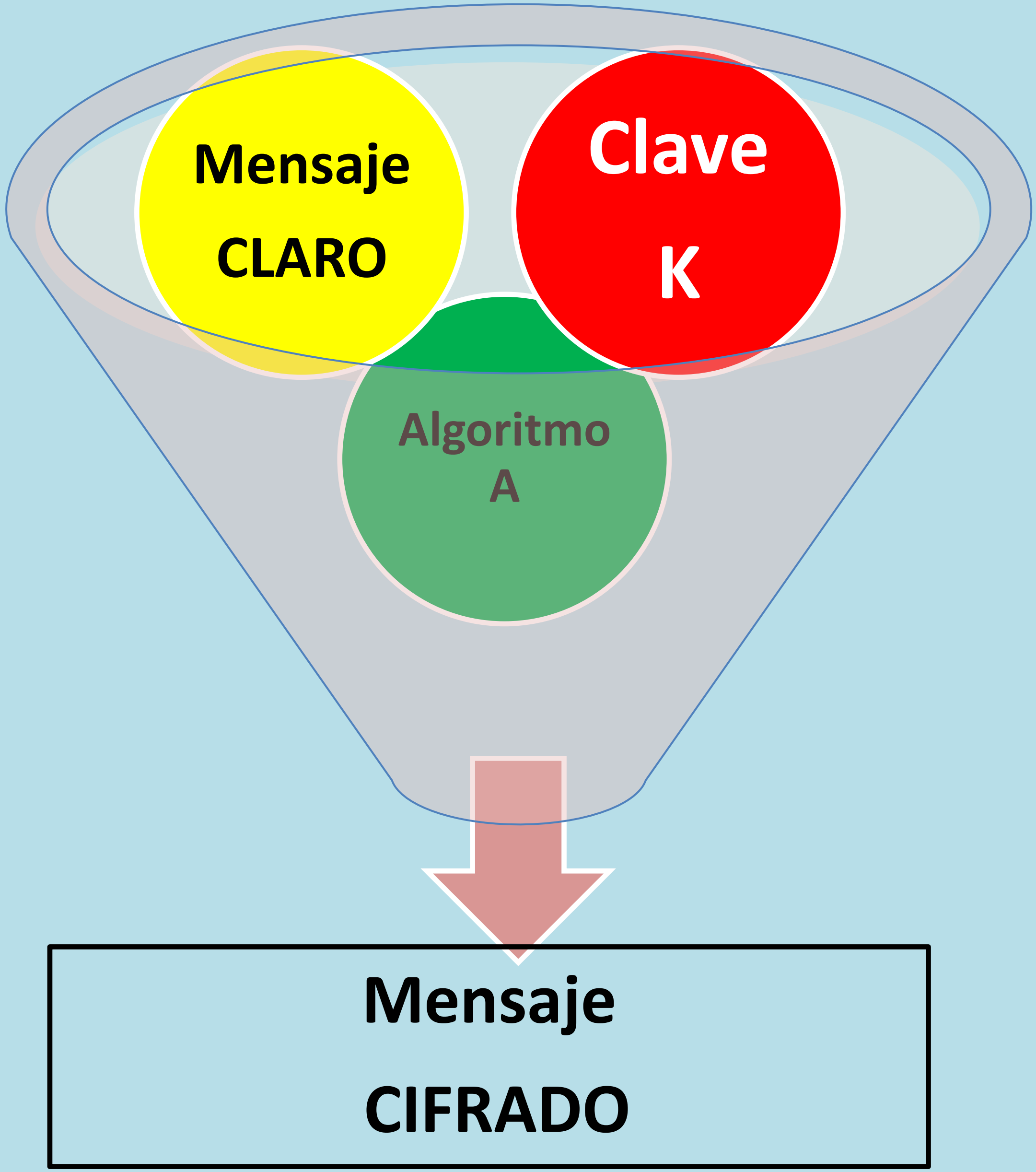
CRIPTOGRAFIA MODERNA

Teoría de la Información y Teoría de Códigos	Teoría de Números y Galois (Campos de Galois, – Teoría de Grupos, etc.)	Teoría de la Complejidad Computacional

Teoría Matemática de la Comunicación



CRIPTOSISTEMA



CONFIDENCIALIDAD
Tipos de Criptografía Moderna

Clave Privada
Simétricos

- La clave debe ser compartida por el Emisor y el Receptor.
- Se usa la misma clave para CIFRAR Y DESCIFRAR

Clave Pública
Asimétricos

- El Emisor tiene una clave para CIFRAR (pública).
- El Receptor tiene OTRA clave para DESCIFRAR (secreta)

Criptografía de Clave Pública

Ron Rivest

Adi Shamir

Lenard Adleman

RSA

CRIPTOSISTEMA RSA

- Algoritmo: exponenciación modular
  - Clave Pública (e, n)
  - Clave Privada (d, n)

$$CIFRADO_{(e;N)}(m) = m^{e} \equiv c(\mathrm{mod}\, N)$$

$$DESCIFRADO_{(d;N)}(c) = c^{d} \equiv m(\mathrm{mod}\, N)$$

Matemática del RSA

$$N = p * q$$

Cálculo del módulo N

- ***p* y *q* son primos positivos de al menos 512 bits c/u.**

Cálculo de *e* y *d*: inversos multiplicativos

$$e * d \equiv 1 (\mathrm{mod}\, \varphi(N))$$

Fortaleza del RSA (computación electrónica)

- **Dificultad para hallar los factores primos de N:**
- **Tamaño de los números primos.**
- **“Baja” eficiencia de los algoritmos de Factorización.**

¡IRROMPIBLE!

Fortaleza de RSA (computación Cuántica)

- **Facilidad para hallar los factores primos de N**
- **No importa el tamaño de los números primos.**
- **“Super” eficiencia en los algoritmos de factorIzación.**

¡ROMPIBLE!

Hagamos un esquema RSA (1)

1. **Elegir 2 números primos de tamaño determinado:** ***p= 11, q=47***

2. **Se multiplican**
   N= 11*47 = 517 (1)

3. **Calcular** φ***(N) = (p-1)*(q-1) = 10*46 = 460 (2)***

Hagamos un esquema RSA (2)

4. Elegir *e* tal que sea coprimo con φ*(N)*
MCD(e,φ(N))=1
usando el Algoritmo Extendido de Euclides)
MCD(7 , 460) = 1
7 (-197) + 460 (3)=1 (3) (Teorema de Bezaut)

5. Si se aplica en ambos miembros mód (φ(N):
7 (-197) + 460 (3) mód (φ(N) $\equiv$ 1 mód (φ(N) (4)

Hagamos un esquema RSA (3)

6. resolviendo:

$$7^* (-197+460) + 0^*(3) \text{ mód } (\varphi(N) \equiv 1 \text{ mód } (\varphi(N) \; (5)$$

$$7 ^*263 \equiv 1 \text{ mód } (\varphi(N) \; (6)$$

CLAVE PÚBLICA
(7; 517)

CLAVE PRIVADA
(263; 517)

Cifrando

Sea un mensaje m (numérico) y coprimo con N
Por ejemplo m=3

Luego Cifrar $_{(7,517)}(3) = 3^7 \bmod(517) \equiv 119 \pmod{517}$

SE ENVÍA 119

Descifrando

Se recibe el mensaje cifrado 119 mod (517)

Luego Descifrar $_{(263,517)}$(119)= 119^{263} mod(517) ≡ 3 (mod (517)

SE DESCIFRÓ EL MENSAJE 3

OBSERVACIONES

-**Es “fácil” CIFRAR (exponenciacion “pequeña”)**
-**Es “difícil” DESCIFRAR (exponenciación “grande”)**
- **No es un algoritmo “liviano” (Lightweight Cryptography)**
-**No es eficiente para usar en el mundo real para cifrar archivos.**

-**USOS REALES CRIPTOGRÁFICOS:**
- **enviar claves secretas para usar con otros algoritmos de cifrado más eficientes. (Cobertura RSA)**
-**Algoritmo de Firma Digital para Autenticación.**

“Cifrando” con RSA

$$\boxed{CIFRADO_{(e;N)}(m_1) = m_1^{\,e} \equiv c_1 (\bmod N)}$$

Aunque no sirve para mensajes, igualmente se puede usar para “practicar” y armar la infraestructura matemática.
El Emisor se pone en contacto con el Receptor y éste le indica su Clave Pública (e=7; N=517) por un canal cualquiera.
Cada carácter se transforma número usando su ASCII .
Cada letra del mensaje se cifra por separado.

Mensaje “HOLA MUNDO”.

H =72 $\Rightarrow$ Cifrar$_{(7;517)}$(72)= 72^7 mod (517) $\equiv$ 74 mod(517)
O = 79 $\Rightarrow$ Cifrar$_{(7;517)}$(79)= 79^7mod (517) $\equiv$ 7 mod(517)
L = 76 $\Rightarrow$ Cifrar$_{(7;517)}$(76)= 76^7mod (517) $\equiv$ 417 mod(517)
A = 65 $\Rightarrow$ Cifrar$_{(7;517)}$(65)= 65^7mod (517) $\equiv$ 241 mod (517)
M = 77 $\Rightarrow$ Cifrar$_{(7;517)}$(77)= 77^7mod (517) $\equiv$ 44 mod (517)
U = 85 $\Rightarrow$ Cifrar$_{(7;517)}$(85)= 85^7mod (517) $\equiv$ 409 mod (517)
N = 78 $\Rightarrow$ Cifrar$_{(7;517)}$(78)= 78^7mod (517) $\equiv$ 485 mod (517)
D = 68 $\Rightarrow$ Cifrar$_{(7;517)}$ (68)= 68^7mod (517) $\equiv$ 84 mod (517)
O = 79 $\Rightarrow$ Cifrar$_{(7;517)}$ (79)= 79^7mod (517) $\equiv$ 7 mod (517)

Finalmente se envían los mensajes 74,7,417,241,44,409,485,84,7 mod (517)

“Descifrando” con RSA

$$\boxed{DESCIFRADO_{(d;N)}(c_1) = c_1^{\,d} \equiv m_1 (\mathrm{mod}\, N)}$$

El Receptor usará su Clave Privada (263;517) la cual está en su poder y permanece a resguardo.
Aplicará el Algoritmo de Descifrado y recuperará el código enviado.
Cada código ASCII descifrado es convertido en su caracter correspondiente .
Uniendo todas las letras… obtiene el mensaje completo.

Descifrar$_{(263;517)}$(74) = 74^{263} mod(517) ≡ 72 mod(517) ⇒H
Descifrar$_{(263;517)}$(7) = 7^{263} mod(517) ≡ 79 mod(517) ⇒ O
Descifrar$_{(263;517)}$(417)= 417^{263} mod(517) ≡ 76 mod(517) ⇒ L
Descifrar$_{(263;517)}$(241)= 241^{263} mod (517) ≡ 65 mod(517) ⇒ A
Descifrar$_{(263;517)}$(44) = 44^{263} mod (517) ≡ 77 mod(517) ⇒ M
Descifrar$_{(263;517)}$(409)= 409^{263} mod (517) ≡85 mod(517) ⇒ U
Descifrar$_{(263;517)}$(485)= 485^{263} mod (517) ≡ 78 mod(517) ⇒ N
Descifrar$_{(263;517)}$(84) = 84^{263} mod (517) ≡ 68 mod(517) ⇒ D
Descifrar$_{(263;517)}$(7) = 7^{263} mod (517) ≡ 79 mod(517) ⇒ O

Mensaje recibido “HOLA MUNDO”.

Bibliografía y Recursos

- Menezes;, A. van Oorschot, P..; Vanstone A. Handbook of Applied Cryptography. 1997
Disponible on-line

- Lucena López. Criptografía y Seguridad en Computadores. 2014. Disponible on-line.

- Lenguaje Python. www.python.org