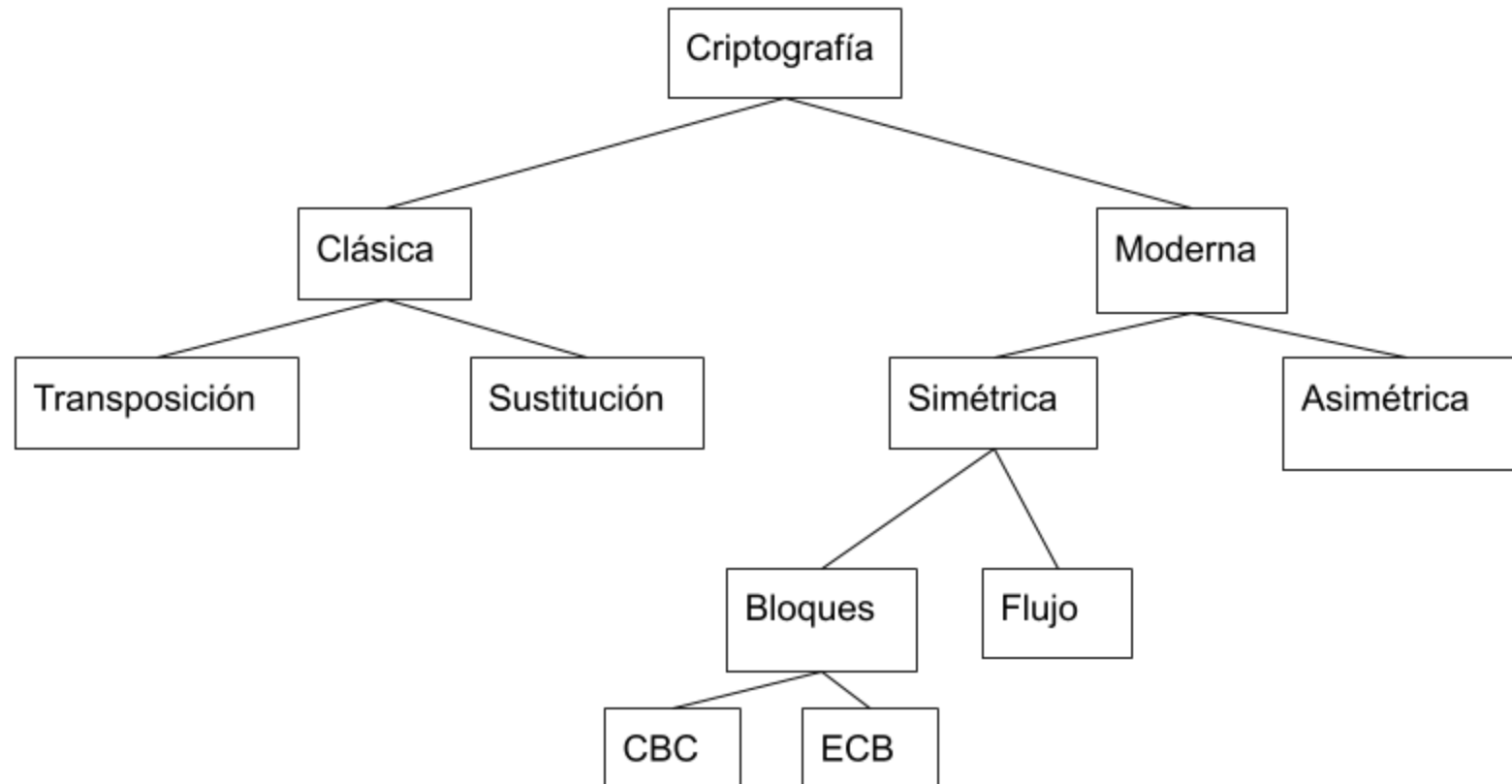


IC

Criptografía - Parte 2

Resumen Criptografía



Sistemas de criptografía

- Existen dos tipos básicos de criptosistemas:
 - **Sistemas de cifrado simétrico** (también conocidos como sistemas de clave secreta o clave privada)
 - **Sistemas de cifrado asimétrico** (también conocidos como sistemas de clave pública)

Criptografía simétrica

Emisor:

- Genera la clave compartida
- Distribuye la clave compartida
- El mensaje original es cifrado usando la clave compartida
- Se obtiene como resultado un mensaje cifrado
- Envía el mensaje cifrado al destinatario



Criptografía simétrica

Receptor:

- El receptor descifra utilizando el mismo sistema de cifrado y la clave compartida
- Se obtiene como resultado el mensaje original.



Criptografía simétrica - Ejemplo cifrado xor

Recipe		Input
XOR		IC{Introducción a la ciberseguridad}
Key 11		
Scheme Standard		
<input type="checkbox"/> Null preserving		
To Hex		
Delimiter Space		
Bytes per line 0		
		Output
		58 52 6a 58 7f 65 63 7e 75 64 72 72 78 e2 7f

Propiedades del XOR

- $a \text{ xor } b = c$
- $c \text{ xor } a = b$

Recipe

XOR

Key
11

HEX

Scheme
Standard

☐ Null preserving

XOR

Key
IC{

LATIN1

Scheme
Standard

☐ Null preserving

To Hex

Delimiter
Space

Bytes per line
0

Input

IC{Introducción a la ciberseguridad}

Output

11 11 11 11 3c 1e 2a 3d 0e 2d 31 09 31 a1 04 78

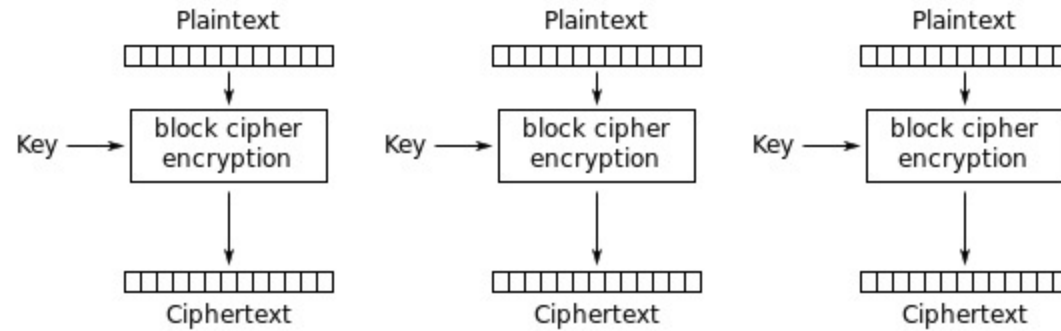
Criptografía simétrica

- Los sistemas simétricos o de clave privada utilizan la misma clave para encriptar y desencriptar
- Existen dos modos de operación básicos:
 - Cifrado en bloques
 - Cifrado de flujo

Cifrado en bloque

- El mensaje en texto claro se divide en bloques de longitud fija (8,16, ... bytes) y luego se aplica el algoritmo de cifrado a cada bloque utilizando una clave secreta. Ejemplos: DES, AES.
- Existen distintos modos de operación dependiendo de cómo se mezcla la clave con el texto claro:
 - ECB: Electronic Codebook
 - CBC: Cipher Block Chaining
 - CFB: Cipher FeedBack
 - OFB: Output FeedBack

Simétrica por bloques: AES ECB



Electronic Codebook (ECB) mode encryption

El mensaje se divide en bloques, y cada uno de ellos es cifrado con la misma clave

¿Desventaja?

- Mismo texto plano mismo texto cifrado

AES en Python

```
pip3 install pycryptodome
```

```
from Crypto.Cipher import AES
clave = b'abcdefghijklmnop'
texto_plano = b'1234567890123456-.-.-.-.-.-.-.-.-.-.1234567890123456'
cifrador = AES.new(clave, AES.MODE_ECB)
texto_cifrado = cifrador.encrypt(texto_plano)
texto_descifrado = cifrador.decrypt(texto_cifrado)
```

```
print(texto_cifrado)
b'3z\xb7s\xceK\x19\xd1#\xf60"z\x9f\xc9\xd7\xd1;}\xf1\xeb\x07\x160\xb1\xd9W\xa5\xb8\xd7\xc2\xc3z\xb7s\xceK\x19\xd1#\xf60"z\x9f\xc9\xd7'

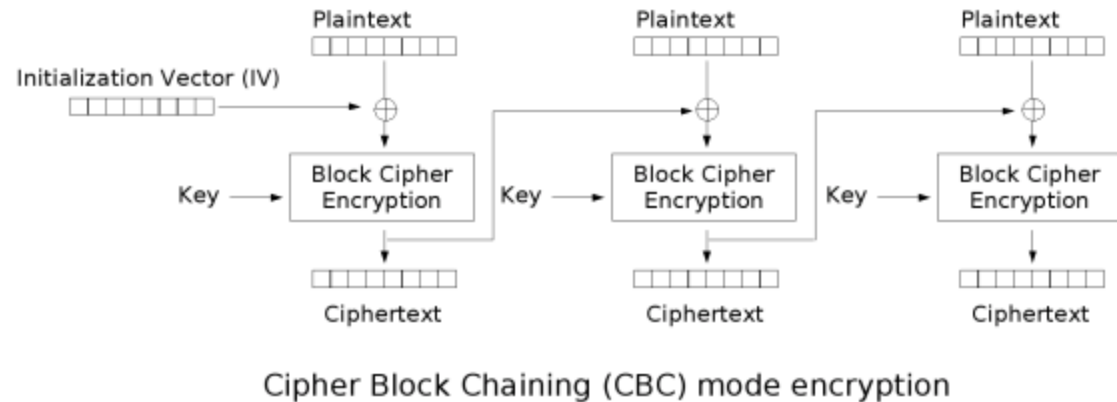
# Primer bloque cifrado
texto_cifrado[:16]
b'3z\xb7s\xceK\x19\xd1#\xf60"z\x9f\xc9\xd7'

# Segundo bloque cifrado
texto_cifrado[17:-16]
b';}\xf1\xeb\x07\x160\xb1\xd9W\xa5\xb8\xd7\xc2\xc3'

# Tercer bloque cifrado
texto_cifrado[-16:]
b'3z\xb7s\xceK\x19\xd1#\xf60"z\x9f\xc9\xd7'

print(texto_descifrado)
b'1234567890123456-.-.-.-.-.-.-.-.-.-.1234567890123456'
```


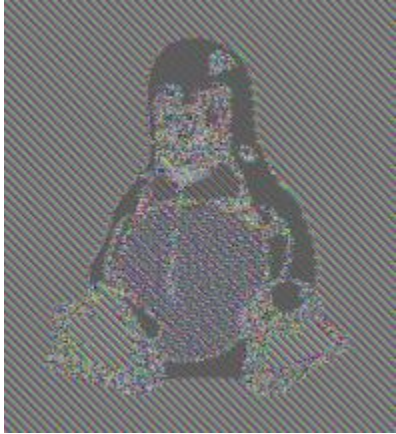
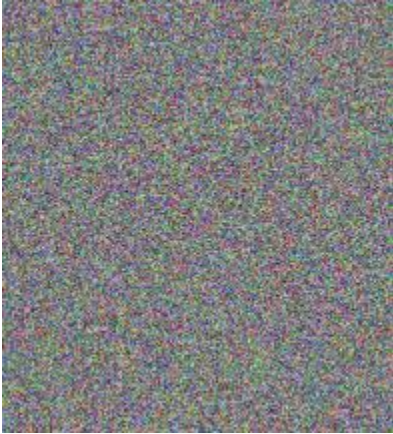
Simétrica por bloques: AES CBC



A cada bloque se le aplica XOR con el bloque anterior antes de ser cifrado, cada bloque depende de todo lo procesado anteriormente

Cifrados en bloque

- ECB puede revelar patrones en los datos cifrados

Original	ECB	CBC
		

Cifrado de flujo

- Para algunas aplicaciones, como el cifrado de conversaciones telefónicas, el cifrado en bloques es inapropiada porque los datos se producen en tiempo real en pequeños fragmentos. Las muestras de datos pueden ser tan pequeñas como 8 bits o incluso de 1 bit.
- El algoritmo genera una secuencia pseudoaleatoria (secuencia cifrante o keystream en inglés) de bits que se emplea como clave. El cifrado se realiza combinando la secuencia cifrante con el texto claro. Ejemplo: RC4.



Cifrado simétrico

- Ventajas
 - Gran velocidad de cifrado y descifrado de datos
 - No aumenta el tamaño del mensaje al cifrar datos
- Desventajas
 - La seguridad depende de un secreto compartido entre el emisor y el receptor
 - Problemas en el uso de una misma clave en un grupos de personas
 - La administración de claves no es “escalable”. Se necesita un medio seguro para el intercambio de la clave

Algunos ejemplos de algoritmos: 3DES, RC5, IDEA, AES, Blowfish

Criptografía asimétrica

- Modo encriptación
- Modo autenticación (usado en firma digital)
- Ejemplo de uso con RSA

Claves asimétricas

- Los sistemas asimétricos utilizan dos claves:
 - La clave pública está disponible para todos
 - La clave privada es conocida sólo por el individuo dueño del par de claves
- Las claves están matemáticamente relacionadas entre sí. Lo que una hace, la otra lo deshace
- Ambas claves pueden ser usadas para encriptar y desencriptar, dependiendo del modo de operación utilizado (encripción o autenticación)

Usos posibles de la criptografía asimétrica

Modo encriptación (Cifrar):

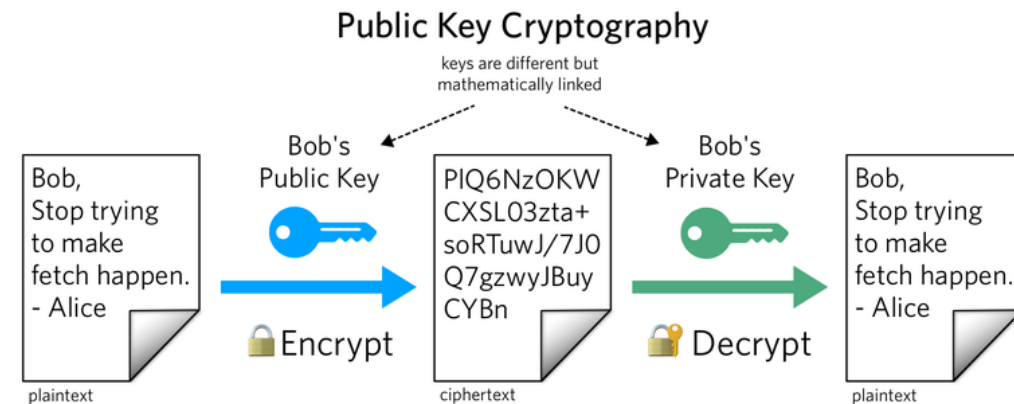
- Emisor encripta con la pública del receptor, el receptor desencripta con su privada.
- Garantiza confidencialidad

Modo autenticación (Firmar):

- Emisor encripta con su privada, el receptor desencripta con la pública del emisor.
- Garantiza autenticidad, integridad y no repudio.

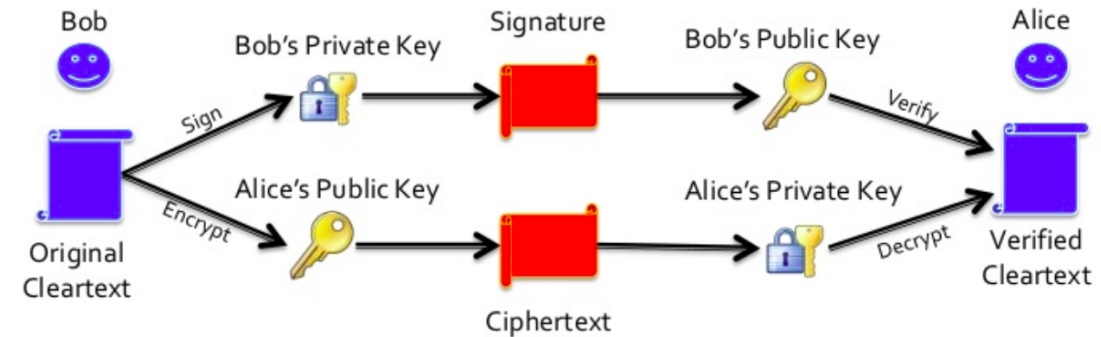
Criptografía asimétrica – Modo encriptación

- El mensaje original es encriptado usando la clave pública del receptor.
- Se obtiene como resultado un mensaje encriptado.
- El mensaje encriptado es enviado al destinatario.
- El mensaje se desencripta usando la clave privada del receptor
- Se obtiene como resultado el mensaje original



Criptografía asimétrica – Modo autenticación

- El mensaje original es encriptado usando la clave privada del emisor.
- Se obtiene como resultado un mensaje encriptado.
- El mensaje encriptado es enviado.
- El mismo puede ser enviado a más de un destinatario.
- El mensaje se descripta usando la clave pública del emisor.
- Se obtiene como resultado el mensaje original.



Cifrado asimétrico

Ventajas

- No es necesario efectuar ningún intercambio de claves secretas.
- A través de sus distintos modos de uso se cubre gran parte de los requisitos de seguridad de la información.

Desventajas

- Requiere mayor potencia de cómputo para cifrar y descifrar que el método simétrico.
- El mensaje cifrado es de mayor tamaño que el original.

Ejemplos de algoritmos

- Diffie-Hellman
- RSA
- DSA
- ElGamal
- CCE

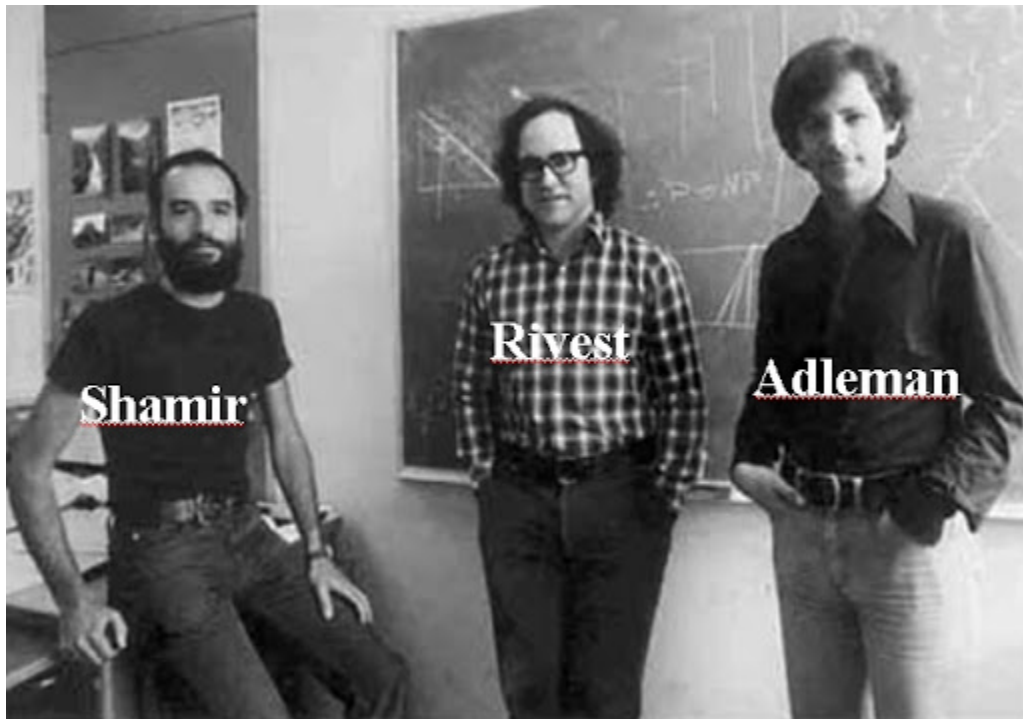
Usos prácticos / protocolos

- En esquemas de confianza centralizados como PKI. Brinda certificados digitales:
 - TLS para comunicaciones seguras (https, smtps, imaps, etc)
 - Certificados personales para encriptación y firma digital
- PGP (esquema de confianza descentralizado para encriptación y firma digital de personas)
- otros protocolos
 - SSH
 - IPsec
 - etc

RSA

Sistema criptográfico de clave pública desarrollado creado por investigadores del MIT en 1978. Es uno de los algoritmos de clave pública mas populares y es válido tanto para cifrar, intercambiar claves y firmar digitalmente.

Seguridad reside en el problema de la factorizacion entera -> Orden exponencial



RSA - Generación de claves

- Elijo dos números primos p y q
- Calculo $n = p * q$
- Calculo $\Phi(n) = (p-1)(q-1)$
- Elijo un valor e tal que $1 < e < \Phi(n)$ y que $\text{mcd}[e, \Phi(n)] = 1$
- Calculo d , inverso multiplicativo modular de e con módulo **modulo** $\Phi(n)$
- Clave publica = (e, n)
- Clave privada = (d, n)

Ejemplo generación de claves

- Elijo dos números primos: $p = 3$ / $q = 11$
- Calculo $n = p * q$
 - $n = 33$
- Calculo $\Phi(n) = (p-1)(q-1)$
 - $\Phi(n) = (3-1)(11-1)$
 $\Phi(n) = 20$
- Elijo un valor e tal que $1 < e < \Phi(n)$ y que $\text{mcd}[e, \Phi(n)] = 1$
 - $e = 7$ (en la practica se utiliza $e = 65537$)
- Calculo el inverso multiplicativo modular d para $e = 7$, $\Phi(n) = 20$
 - $d = 3$

Inverso multiplicativo modular

$$d \equiv 1/e \pmod{\Phi(n)}$$

$$d * e \pmod{\Phi(n)} = 1$$

$7 * 0 \bmod 20 = 0$	$7 * 10 \bmod 20 = 10$
$7 * 1 \bmod 20 = 7$	$7 * 11 \bmod 20 = 17$
$7 * 2 \bmod 20 = 14$	$7 * 12 \bmod 20 = 4$
$7 * 3 \bmod 20 = 1$	$7 * 13 \bmod 20 = 11$
$7 * 4 \bmod 20 = 8$	$7 * 14 \bmod 20 = 18$
$7 * 5 \bmod 20 = 15$	$7 * 15 \bmod 20 = 5$
$7 * 6 \bmod 20 = 2$	$7 * 16 \bmod 20 = 12$
$7 * 7 \bmod 20 = 9$	$7 * 17 \bmod 20 = 19$
$7 * 8 \bmod 20 = 16$	$7 * 18 \bmod 20 = 6$
$7 * 9 \bmod 20 = 3$	$7 * 19 \bmod 20 = 13$

Se calcula con el algoritmo extendido de euclides

d es guardado en secreto como el exponente privado

Ejemplo generación de claves

- Elijo dos números primos: $p = 3$ / $q = 11$
- Calculo $n = p * q$
 - $n = 33$
- Calculo $\Phi(n) = (p-1)(q-1)$
 - $\Phi(n) = 20$
- Elijo un valor e tal que $1 < e < \Phi(n)$ y que $\text{mcd}[e, \Phi(n)] = 1$
 - $e = 7$ (en la practica se utiliza $e = 65537$)
- Calculo el inverso multiplicativo modular d para $e = 7$, $\Phi(n) = 20$
 - $d = 3$

Clave publica = (7,33)

Clave privada = (3,33)

RSA - Cifrado/Descifrado

$$\text{mensaje}^e \bmod(n) = \text{cifrado}$$

$$\text{cifrado}^d \bmod(n) = \text{mensaje}$$

$m^e \bmod n = c$

$0^7 \bmod 33$	=	0	$17^7 \bmod 33$	=	8
$1^7 \bmod 33$	=	1	$18^7 \bmod 33$	=	6
$2^7 \bmod 33$	=	29	$19^7 \bmod 33$	=	13
$3^7 \bmod 33$	=	9	$20^7 \bmod 33$	=	26
$4^7 \bmod 33$	=	16	$21^7 \bmod 33$	=	21
$5^7 \bmod 33$	=	14	$22^7 \bmod 33$	=	22
$6^7 \bmod 33$	=	30	$23^7 \bmod 33$	=	23
$7^7 \bmod 33$	=	28	$24^7 \bmod 33$	=	18
$8^7 \bmod 33$	=	2	$25^7 \bmod 33$	=	31
$9^7 \bmod 33$	=	15	$26^7 \bmod 33$	=	5
$10^7 \bmod 33$	=	10	$27^7 \bmod 33$	=	3
$11^7 \bmod 33$	=	11	$28^7 \bmod 33$	=	19
$12^7 \bmod 33$	=	12	$29^7 \bmod 33$	=	17
$13^7 \bmod 33$	=	7	$30^7 \bmod 33$	=	24
$14^7 \bmod 33$	=	20	$31^7 \bmod 33$	=	4
$15^7 \bmod 33$	=	27	$32^7 \bmod 33$	=	32
$16^7 \bmod 33$	=	25	$33^7 \bmod 33$	=	0

$c^d \bmod n = m$

$0^3 \bmod 33$	=	0	$17^3 \bmod 33$	=	29
$1^3 \bmod 33$	=	1	$18^3 \bmod 33$	=	24
$2^3 \bmod 33$	=	8	$19^3 \bmod 33$	=	28
$3^3 \bmod 33$	=	27	$20^3 \bmod 33$	=	14
$4^3 \bmod 33$	=	31	$21^3 \bmod 33$	=	21
$5^3 \bmod 33$	=	26	$22^3 \bmod 33$	=	22
$6^3 \bmod 33$	=	18	$23^3 \bmod 33$	=	23
$7^3 \bmod 33$	=	13	$24^3 \bmod 33$	=	30
$8^3 \bmod 33$	=	17	$25^3 \bmod 33$	=	16
$9^3 \bmod 33$	=	3	$26^3 \bmod 33$	=	20
$10^3 \bmod 33$	=	10	$27^3 \bmod 33$	=	15
$11^3 \bmod 33$	=	11	$28^3 \bmod 33$	=	7
$12^3 \bmod 33$	=	12	$29^3 \bmod 33$	=	2
$13^3 \bmod 33$	=	19	$30^3 \bmod 33$	=	6
$14^3 \bmod 33$	=	5	$31^3 \bmod 33$	=	25
$15^3 \bmod 33$	=	9	$32^3 \bmod 33$	=	32
$16^3 \bmod 33$	=	4	$33^3 \bmod 33$	=	0

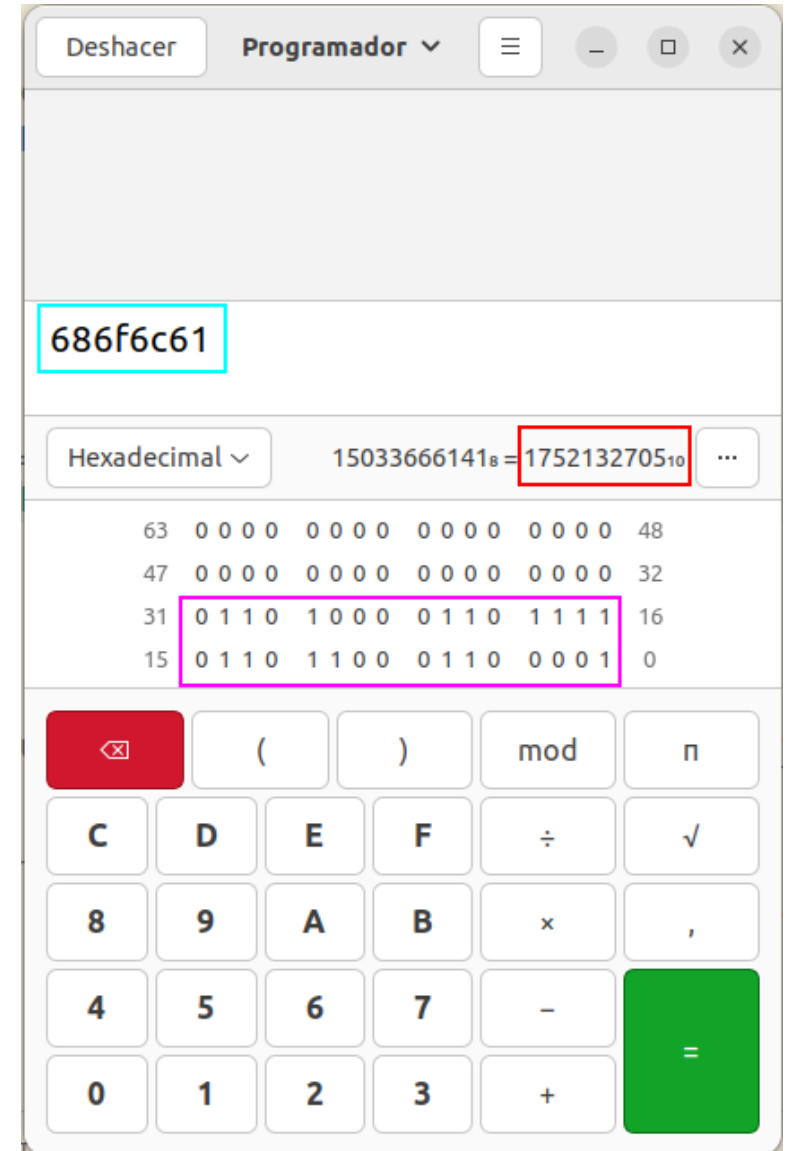
¿cuántos mensajes distintos se pueden cifrar? -> n

RSA - Cifrado/Descifrado

- En cifrado asimétrico, sólo se cifran números, por tanto, ciframos la representación decimal de un string

```
>>> m = 'hola'
>>> for c in m:
...     print(c, hex(ord(c)))
...
h 0x68
o 0x6f
l 0x6c
a 0x61
>>>
```

"hola" => \x68\x6f\x6c\x61 => 1752132705



Criptografía asimétrica en python

Usamos funciones de la libreria libnum: <https://github.com/hellman/libnum>

```
# generar primos
- generate_prime(size, k=25)

# calcular inverso modular
- invmod(a, n) - modulo inverse

# transformar mensaje a numero y numero a mensaje
- s2n(s) - packed string to number
- n2s(n) - number to packed string
```

Función para cifrar o descifrar

```
- pow(mensaje, e, n)
```

Firma con RSA:

Firma (cifrado con la privada):

```
firma = pow(m, d, n)
```

Chequeo de firma (descifrar con la pública):

```
m = pow(firma, e, n)
```

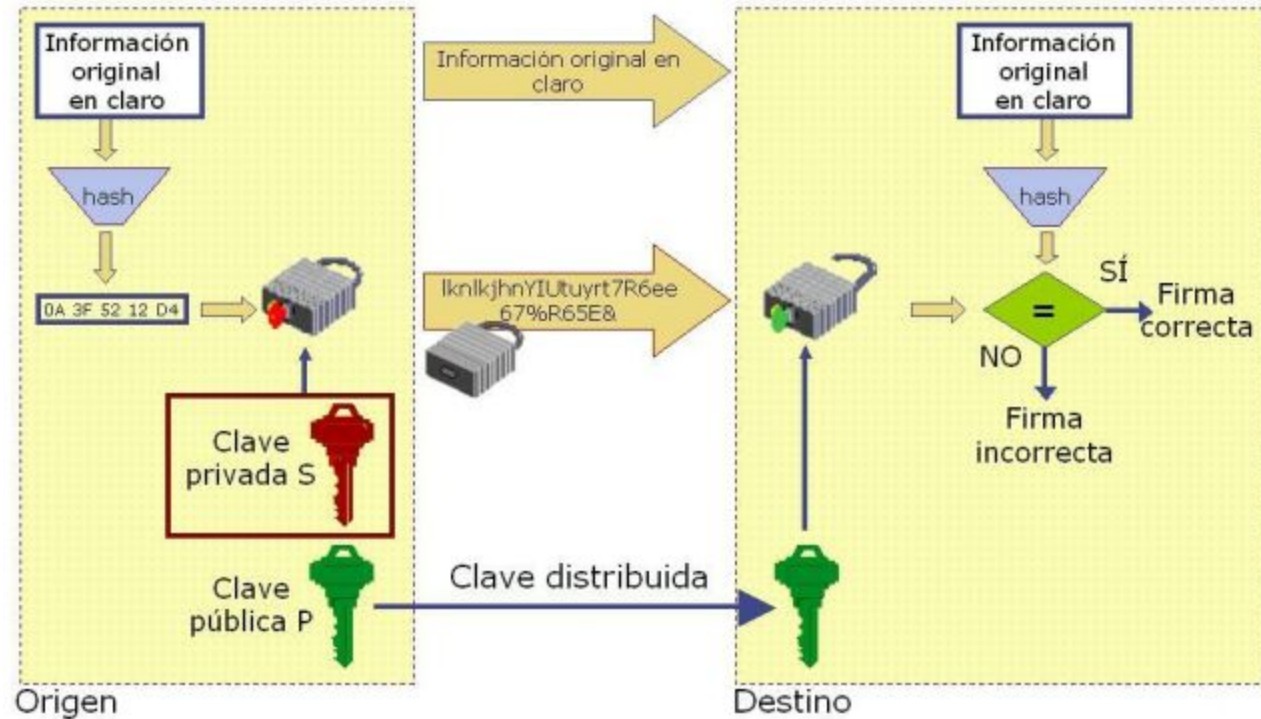
Dado que mi clave pública es pública, cualquiera puede descifrarlo y verificar la firma para reconocer la autoría del mismo.

RSA

- Fortaleza del sistema:
 - Conocidos n y e que son públicos, la "única" manera de conocer la clave privada d , es realizar la factorización de n , para obtener p y q
 - d , $\Phi(n)$, p y q se deben mantener en secreto
 - RSA basa su seguridad en la dificultad computacional de factorizar números compuestos muy grandes
 - Si se logra factorizar el módulo público (n), es posible recuperar la clave privada
 - <http://www.factordb.com>

Firma digital

Firma digital



La firma digital garantiza la integridad del mensaje y el no-repudio en origen.

Si se pudo verificar la firma utilizando la clave pública del emisor, entonces la misma sólo pudo haberse generado con la clave privada del mismo.

Criptografía simétrica

- Alta velocidad o tasa de cifra (MB/s)
- Eficiente para uso en grupos reducidos (necesito 1 sola clave)
- Infraestructura sencilla
- Claves pequeñas
- Es necesario compartir claves por medios no seguros
- Si se compromete la clave se compromete toda la comunicación
- No permite autenticación
- Elevado número de claves a recordar

Criptografía asimétrica

- Numero de claves reducido
- Seguridad computacional de la clave privada
- No es necesario transmitir la clave privada ente emisor y receptor
- Permite intercambio de claves seguro
- Permite autenticar usuarios
- Necesidad de infraestructura de clave publica para el manejo de la confianza