Cifrado Vernam Y RSA

Ronald Mas,
Angel Ramirez

20 de agosto de 2020

Contenido

Código ASCII

El Código Estándar Americano para el Intercambio de Información (ASCII) es usado en la mayoría de las computadoras y otros equipos electrónicos. Este código emplea siete bits, cuatro dígitos de zona y los otro cuatro restantes corresponden al BCD 8421.
Los dígitos de zona utilizados son los siguientes:

Caso	**Código**
Letras mayúsculas	100 101
Letras minúsculas	110 111
Números	011
Signos de puntuación	010

American Standard Code for Information Interchange (ASCII)

	$B_7B_6B_5$						
$B_4B_3B_2B_1$	000	001	010	011	100	101	110
0000	NULL	DLE	SP	0	@	P	`
0001	SOH	DC1	!	1	A	Q	a
0010	STX	DC2	"	2	B	R	b
0011	ETX	DC3	#	3	C	S	c
0100	EOT	DC4	$	4	D	T	d
0101	ENQ	NAK	%	5	E	U	e
0110	ACK	SYN	&	6	F	V	f
0111	BEL	ETB	'	7	G	W	g
1000	BS	CAN	(	8	H	X	h
1001	HT	EM	)	9	I	Y	i
1010	LF	SUB	*	:	J	Z	j
1011	VT	ESC	+	;	K	[	k
1100	FF	FS	,	<	L	\	l
1101	CR	GS	-	=	M	]	m
1110	SO	RS	.	>	N	^	n
1111	SI	US	/	?	O	_	o

Ejemplo 1:

Decodifique el mensaje:

100 0101	110 1100	010 0000	110 0011	110 1111	111 0010	111 0010	110 0101
110 1111	010 0000	110 0100	110 0101	110 1100	010 0000	111 0000	111 0010
110 1111	110 0110	110 0101	111 0011	110 1111	111 0010	010 0000	110 0101
111 0011	010 0000	100 1101	100 0001	101 1000	011 1000	011 0010	100 0000
110 0111	110 1101	110 0001	110 1001	110 1100	010 1110	110 0011	110 1111
110 1101							

Observaciones:

- Decodificar un mensaje en ASCII sin una agrupación de 7 en 7 no es tan sencillo debido a que podría causar una confunsión con los códigos BCD 8421 (4 bits) o Unicode (16 bits).

Solución del Ejemplo 1:

100 0101	110 1100	010 0000	110 0011	110 1111	111 0010	111 0010	110 0101
E	l		c	o	r	r	e
110 1111	010 0000	110 0100	110 0101	110 1100	010 0000	111 0000	111 0010
o		d	e	l		p	r
110 1111	110 0110	110 0101	111 0011	110 1111	111 0010	010 0000	110 0101
o	f	e	s	o	r		e
111 0011	010 0000	100 1101	100 0001	101 1000	011 1000	011 0010	100 0000
s		M	A	X	8	2	@
110 0111	110 1101	110 0001	110 1001	110 1100	010 1110	110 0011	110 1111
g	m	a	i	l	.	c	o
110 1101							
m							

Cifrado Vernam

El cifrado Vernam es un cifrado de flujo que cifra de la siguiente manera. Dada una cadena de bits

$$m_1 m_2 \cdots m_n \in M$$

y una clave

$$k_1 k_2 \cdots k_n \in K$$

la transformación de cifrado está dada por:

$$E_{k_j}(m_j) = m_j + k_j = c_j \in C$$

y la transformación de desciframiento viene dada por:

$$D_{k_j}(m_j) = c_j + k_j = m_j \in C$$

donde $+$ es la suma módulo 2.

Operación XOR

El operador XOR tiene como salida un 1 siempre que las entradas no coincidan, lo cual ocurre cuando una de las dos entradas es exclusivamente verdadera. Esto coincide con la suma módulo 2.

Tabla XOR

1	1	0
1	0	1
0	1	1
0	0	0

Ejemplo 2

Al operar 1101 con 0110 usando XOR se tiene que:

$$1\,XOR\,0 = 1$$
$$0\,XOR\,1 = 1$$
$$1\,XOR\,1 = 0$$
$$1\,XOR\,0 = 1$$

Ejemplo 3:

Codificar el mensaje **Hola** usando el cifrado Vernam con clave **948/ma**:

1) Codificar usando la tabla Ascii el mensaje **Hola** se tiene:

$$100\,1000 \quad 110\,1111 \quad 110\,1100 \quad 110\,0001.$$

2) Codificar usando la tabla Ascii la clave **948/** (la longitud debe coincidir con el mensaje, por ello e suprime ma) se tiene:

$$011\,1001 \quad 011\,0100 \quad 011\,1000 \quad 010\,1111$$

AL operar usando XOR se tiene:

$$111\,0001 \quad 101\,1011 \quad 101\,0100 \quad 100\,1110$$

Por tanto el mensaje encriptado es:

$$q[TN$$

RSA

En criptografía, RSA (Rivest, Shamir y Adleman) es un sistema criptográfico de clave pública desarrollado en 1979. Es el primer y más utilizado algoritmo de este tipo y es válido tanto para cifrar como para firmar digitalmente. La seguridad de este algoritmo radica en el problema de la factorización de números enteros.

Algoritmo

1) Elegir dos números primos p y q relativamente grandes.
2) Calcular $n = p \cdot q$.
3) Hallar el indicador de Euler de n, el cuál es $\varphi(n) = (p-1)(q-1)$.
4) Elegir un $e \in \mathbb{N}$ tal que

$$1 < e < \varphi(n) \ \wedge \ MCD(e, \varphi(n)) = 1.$$

5) Hallar un $d \in \mathbb{N}$ tal que $de \equiv 1 \, (mod \, \varphi(n))$.

Continua Algoritmo RSA

Continua Algoritmo

6) Escribamos:
Clave Pública: (e,n)
Se puede entregar a cualquier persona.
Clave Privada: (d,n)
El propietario debe guardarla de modo que nadie tenga acceso a ella.

7) Resolver las siguientes congruencias para:
Encriptar $c = m^e \, (mod \, n)$.
Desencriptar $m = c^d \, (mod \, n)$.

Ejemplo

1) Elegimos los números primos $p = 29, q = 43$.
2) Calcular $n = 29 \times 43 = 1247$.
3) Hallar $\varphi(1247) = (28)(42) = 1176$.
4) Elegimos $e = 5$ que cumple

$$1 < e < 1176 \ \wedge \ MCD(e, \varphi(n)) = 1.$$

5) Hallar un $d \in \mathbb{N}$ tal que $de \equiv 1 \, (mod \, 1176)$. Al resolver se tiene que:

$$d = 941.$$

6) Escribamos:
   Clave Pública: (5,1247)
   Clave Privada: (941,1247)
7) Resolver las siguientes congruencias para:
   Encriptar $c = m^5 \, (mod \, 1247)$.
   Desencriptar $m = c^{941} \, (mod \, 1241)$.

Continua Ejemplo

EL receptor recibe la clave pública $(5, 1247)$ y desea enviar el mensaje:

send money

Codificamos en RSA, al aplicar la tabla de equivalencia numérica vista en clases anteriores se tiene:

$$\{m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, m_9\} = \{18, 04, 13, 03, 12, 14, 13, 04, 24\},$$

Al aplicar la clave pública se tiene:

$$\begin{array}{lll} m_1^5 \equiv 18^5 \equiv 363 \equiv c_1 \,(mod\ n) & , & m_2^5 \equiv 4^5 \equiv 1024 \equiv c_2 \,(mod\ n) \\ m_3^5 \equiv 13^5 \equiv 934 \equiv c_3 \,(mod\ n) & , & m_4^5 \equiv 3^5 \equiv 243 \equiv c_4 \,(mod\ n) \\ m_5^5 \equiv 12^5 \equiv 679 \equiv c_5 \,(mod\ n) & , & m_6^5 \equiv 14^5 \equiv 367 \equiv c_6 \,(mod\ n) \\ m_7^5 \equiv 13^5 \equiv 934 \equiv c_7 \,(mod\ n) & , & m_8^5 \equiv 4^5 \equiv 1024 \equiv c_8 \,(mod\ n) \\ m_9^5 \equiv 24^5 \equiv 529 \equiv c_9 \,(mod\ n) & , & \end{array}$$

Continua Ejemplo

Por tanto el código que envía es:

$\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9\} = \{363, 1024, 934, 243, 679, 367, 934, 1024, 529$

La persona original recibe el mensaje y usa su clave privada (941,1247).

$$c_1^d \equiv 363^{941} \equiv 18 \equiv m_1 \, (mod \, n) \quad , \quad c_2^d \equiv 1024^{941} \equiv 4 \equiv m_2 \, (mod \, n)$$

así sucesivamente hasta recuperar el mensaje original.