

Kriptografi Klasik

Presented by

Dr. R. Rizal Isnanto, S.T., M.M., M.T.

Kriptografi klasik

- Ada 5 algoritma kriptografi klasik yang dipelajari di mata kuliah ini.
- Kelima algoritma tersebut adalah:
 - Caesar cipher
 - Vigenere cipher
 - Matrix encryption
 - Playfair
 - Vernam cipher

1. Caesar Cipher

- Ditemukan oleh Raja Romawi, Julius Caesar.
- Sistem ini mengharuskan kita menghitung pengganti sebuah huruf dengan menggeser posisi urutannya sesuai dengan kunci.
- Yang dipakai adalah huruf alfabet.

Langsung ke contohnya

- Misal :

kita mempunyai plaintext "GADJAH"

kuncinya = 6.

- Bagaimana penyelesaiannya?

sebelum membuat tabel penyelesaian, kita harus mengetahui konsepnya terlebih dahulu, inti dari caesar cipher adalah pergeseran posisi yang berurutan sesuai dengan jumlah nilai kuncinya.

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
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T

Jawabannya....

Setelah diurutkan, jawabannya adalah
"AUXDUB"

Vigenere cipher

- Merupakan perkembangan ide dari Caesar Cipher.
- Cipher ini menggunakan sebuah tabel yang berbentuk matriks alfabet, yang berisikan baris-baris alfabet yang telah digeser dari 1-25.
- Setiap baris dan kolom mendapat indeks sebuah huruf alfabet.
- Proses enkripsi dilakukan dengan cara mengambil karakter pertama kunci sebagai indeks baris dan karakter pertama plaintext sebagai indeks kolom.
- Elemen matriks yang ditunjuk oleh baris kolom tersebut merupakan karakter pada ciphertext.
- Proses diulangi sampai plaintext habis, jika kunci lebih pendek dari plaintext maka kunci boleh diulang.

Contoh soal:

- Misal : ada kunci = "gadjah"

dan plaintext-nya = "helloelephant"

*karena kunci lebih pendek dari plaintext maka dirangkakan beberapa kali.....

Plaintext	h	e	l	l	o	e	l	e	p	h	a	n	t
Key	g	a	d	j	a	h	g	a	d	j	a	h	g
ciphertext	n	e	o	u	o	l	r	e	s	q	a	u	z

Plaintext

Ciphertext

The image shows a Vigenere cipher square. The columns are labeled with the key 'g', 'a', 'd', 'j', 'a', 'h', 'g', 'a', 'd', 'j', 'a', 'h', 'g'. The rows are labeled with the plaintext 'h', 'e', 'l', 'i', 'o', 'e', 'l', 'e', 'p', 'h', 'a', 'n', 't'. The intersection of the key and plaintext letters gives the ciphertext 'neouolresqauz'. The square itself is a 26x26 grid of letters. A large red arrow points from the word 'Plaintext' to the square, and another large red arrow points from the square to the word 'Ciphertext'. A third red arrow points from the word 'Kunci' to the key row of the square.

Kunci

Plaintext	h	e	l	i	o	e	l	e	p	h	a	n	t
Key	g	a	d	j	a	h	g	a	d	j	a	h	g
ciphertext	n	e	o	u	o	l	r	e	s	q	a	u	z

Matrix encryption

- Ide : plaintext diletakkan pada matriks bujursangkar.
- Contoh : UNIVERSITAS DIPONEGORO
 - * spasi pada plaintext diabaikan.
- Jumlah karakter contoh plaintext di atas adalah 21 karakter, maka matriks yang dipakai adalah $25 = 5^2$.

Matriksnya menjadi:

1	2	3	4	5
U	N	I	V	E
R	S	I	T	A
S	D	I	P	O
N	E	G	O	R
O	X	X	X	X

•Misal : kunci = 41325 (berdasar kolom)

•Jadi chipertext =

VTPOX URSNO IIIGX NSDEX EAORX

dan digabung menjadi :

VTPOXURSNOIIIGXNSDEXEAORX

- Dekripsi :
 - Bagi 1 blok untuk 5 huruf
 - Buat matriks berdasar kolom (atas ke bawah)
 - Urutkan dalam posisi 12345
 - Baca per baris (UNIVERSITASDIPONEGORO)

PLAYFAIR

- Ide : buat matriks 5 x 5 untuk diisi 26 karakter,
*ada 1 sel pada kolom tersebut diisi oleh 2 karakter
: I/J
- Contoh : kunci = GADJAH

1	2	3	4	5
G	A	D	J/I	H
B	C	E	F	K
L	M	N	O	P
Q	R	S	T	U
V	W	X	Y	Z

- Jika plaintext = KRIPTOGRAFI

- KR IP TO GR AF IX

* bila jumlah karakter pada plaintext tersebut ganjil seperti contoh di atas maka ditambah karakter X untuk menggenapinya.

- Buat kunci, misal : GADJAH

- Buat sel matriks 5 x 5 (=25)

G	A	D	J/I	H
B	C	E	F	K
L	M	N	O	P
Q	R	S	T	U
V	W	X	Y	Z

G	A	D	J/I	H
B	C	E	F	K
L	M	N	O	P
Q	R	S	T	U
V	W	X	Y	Z

m	K	R	I	P	T	O	G	R	A	F	I	X
c	U	C	O	H	Y	T	Q	A	C	J	Y	D

Contoh:

- Plaintext : KRIPTOGRAFI → KR IP TO GR AF IX
- Kunci: GADJAH
dienkripsi menjadi
- Ciphertext : UC OH YT QA CJ YD → UCOHYTQACJYD
kemudian dideskripsikan lagi menjadi
KR JP TO GR AF JX → KRJP TO GR AF JX → KRJPTOGRAFJX

Vernam cipher

- Memanfaatkan konsep bit dan logika XOR untuk memecahkan kodenya.
- Contoh :
 - Plaintext : KAMU
 - Kunci : AKU
 - Bagaimana mencari ciphertextnya?
 - Untuk mencarinya kita ubah dulu plaintext dan kuncinya ke ASCII
 - $K = (\text{ASCII } 75) = 01001011$
 - $A = (\text{ASCII } 65) = 01000001$
 - $M = (\text{ASCII } 77) = 01001101$
 - $U = (\text{ASCII } 85) = 01010101$

- Maka plaintextnya menjadi

- 01001011010000010100110101010101

Dan kuncinya menjadi

- 010000010100101101010101

oleh karena kunci lebih pendek dari plaintext maka kunci diulang agar panjangnya sesuai dengan plaintext-nya

- 01000001010010110101010101000001

- Kemudian plaintext dan kunci di-XOR-kan
- P : 01001011010000010100110101010101
- K : 01000001010010110101010101000001 XOR
- C : 00001010000010100001100000010100
- ASCII → bisa dicari (jika angkanya kecil, biasanya merupakan fungsi kontrol yang tidak bisa dicetak)
 - 00001010 = 10d → ?
 - 00001010 = 10d → ?
 - 00011000 = 24d → ?
 - 00010100 = 20d → ?
- Untuk proses deskripsi pesan juga melakukan operasi yang sama yaitu XOR antara cipher dan kuncinya maka didapat plaintext-nya.

TABEL ASCII

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	00 <small>0000 0000</small>	01 <small>0000 0001</small>	02 <small>0000 0010</small>	03 <small>0000 0011</small>	04 <small>0000 0100</small>	05 <small>0000 0101</small>	06 <small>0000 0110</small>	07 <small>0000 0111</small>	08 <small>0000 1000</small>	09 <small>0000 1001</small>	10 <small>0000 1010</small>	11 <small>0000 1011</small>	12 <small>0000 1100</small>	13 <small>0000 1101</small>	14 <small>0000 1110</small>	15 <small>0000 1111</small>
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
	□	┌	└	┐	↯	⊗	✓	⤵	↶	➤	≡	∇	⇓	⇐	⊗	⊙
	16 <small>0001 0000</small>	17 <small>0001 0001</small>	18 <small>0001 0010</small>	19 <small>0001 0011</small>	20 <small>0001 0100</small>	21 <small>0001 0101</small>	22 <small>0001 0110</small>	23 <small>0001 0111</small>	24 <small>0001 1000</small>	25 <small>0001 1001</small>	26 <small>0001 1010</small>	27 <small>0001 1011</small>	28 <small>0001 1100</small>	29 <small>0001 1101</small>	30 <small>0001 1110</small>	31 <small>0001 1111</small>
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
	☐	⊖	⊕	⊗	⊘	✓	∩	⊣	⊗	†	‡	⊖	⊞	⊠	⊡	⊢
	32 <small>0010 0000</small>	33 <small>0010 0001</small>	34 <small>0010 0010</small>	35 <small>0010 0011</small>	36 <small>0010 0100</small>	37 <small>0010 0101</small>	38 <small>0010 0110</small>	39 <small>0010 0111</small>	40 <small>0010 1000</small>	41 <small>0010 1001</small>	42 <small>0010 1010</small>	43 <small>0010 1011</small>	44 <small>0010 1100</small>	45 <small>0010 1101</small>	46 <small>0010 1110</small>	47 <small>0010 1111</small>
2	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
	48 <small>0011 0000</small>	49 <small>0011 0001</small>	50 <small>0011 0010</small>	51 <small>0011 0011</small>	52 <small>0011 0100</small>	53 <small>0011 0101</small>	54 <small>0011 0110</small>	55 <small>0011 0111</small>	56 <small>0011 1000</small>	57 <small>0011 1001</small>	58 <small>0011 1010</small>	59 <small>0011 1011</small>	60 <small>0011 1100</small>	61 <small>0011 1101</small>	62 <small>0011 1110</small>	63 <small>0011 1111</small>
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
	64 <small>0100 0000</small>	65 <small>0100 0001</small>	66 <small>0100 0010</small>	67 <small>0100 0011</small>	68 <small>0100 0100</small>	69 <small>0100 0101</small>	70 <small>0100 0110</small>	71 <small>0100 0111</small>	72 <small>0100 1000</small>	73 <small>0100 1001</small>	74 <small>0100 1010</small>	75 <small>0100 1011</small>	76 <small>0100 1100</small>	77 <small>0100 1101</small>	78 <small>0100 1110</small>	79 <small>0100 1111</small>
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	80 <small>0101 0000</small>	81 <small>0101 0001</small>	82 <small>0101 0010</small>	83 <small>0101 0011</small>	84 <small>0101 0100</small>	85 <small>0101 0101</small>	86 <small>0101 0110</small>	87 <small>0101 0111</small>	88 <small>0101 1000</small>	89 <small>0101 1001</small>	90 <small>0101 1010</small>	91 <small>0101 1011</small>	92 <small>0101 1100</small>	93 <small>0101 1101</small>	94 <small>0101 1110</small>	95 <small>0101 1111</small>
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
	96 <small>0110 0000</small>	97 <small>0110 0001</small>	98 <small>0110 0010</small>	99 <small>0110 0011</small>	100 <small>0110 0100</small>	101 <small>0110 0101</small>	102 <small>0110 0110</small>	103 <small>0110 0111</small>	104 <small>0110 1000</small>	105 <small>0110 1001</small>	106 <small>0110 1010</small>	107 <small>0110 1011</small>	108 <small>0110 1100</small>	109 <small>0110 1101</small>	110 <small>0110 1110</small>	111 <small>0110 1111</small>
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
	112 <small>0111 0000</small>	113 <small>0111 0001</small>	114 <small>0111 0010</small>	115 <small>0111 0011</small>	116 <small>0111 0100</small>	117 <small>0111 0101</small>	118 <small>0111 0110</small>	119 <small>0111 0111</small>	120 <small>0111 1000</small>	121 <small>0111 1001</small>	122 <small>0111 1010</small>	123 <small>0111 1011</small>	124 <small>0111 1100</small>	125 <small>0111 1101</small>	126 <small>0111 1110</small>	127 <small>0111 1111</small>
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

- Terima kasih
- Ada pertanyaan?