

02 – Ragam Cipher Klasik

(Bagian 1)

Oleh: Rinaldi Munir

Pendahuluan

- Kriptografi klasik merupakan kriptografi yang sudah tua, sudah ada sejak ribuan tahun yang lalu sampai ditemukan computer digital
- Cipher klasik (*classical cipher*) hanya memproses pesan berbasis huruf alfabet
- Menggunakan alat tulis pena dan kertas saja
- Termasuk ke dalam jenis kriptografi kunci-simetri

- Tiga alasan mempelajari kriptografi klasik:
 1. Memahami konsep dasar kriptografi.
 2. Sebagai dasar algoritma kriptografi modern.
 3. Untuk memahami kelemahan sistem *cipher*.

- *Cipher* di dalam kriptografi klasik disusun oleh dua teknik dasar:

1. Teknik substitusi: mengganti huruf plainteks dengan huruf cipherteks.

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

Contoh: Plainteks: MENGANTUK

Cipherteks: CQBSIBONW

2. Teknik transposisi: mengubah susunan atau posisi huruf plainteks menjadi susunan huruf cipherteks.

Disebut juga teknik *scrambling*, permutasi, atau pengacakan

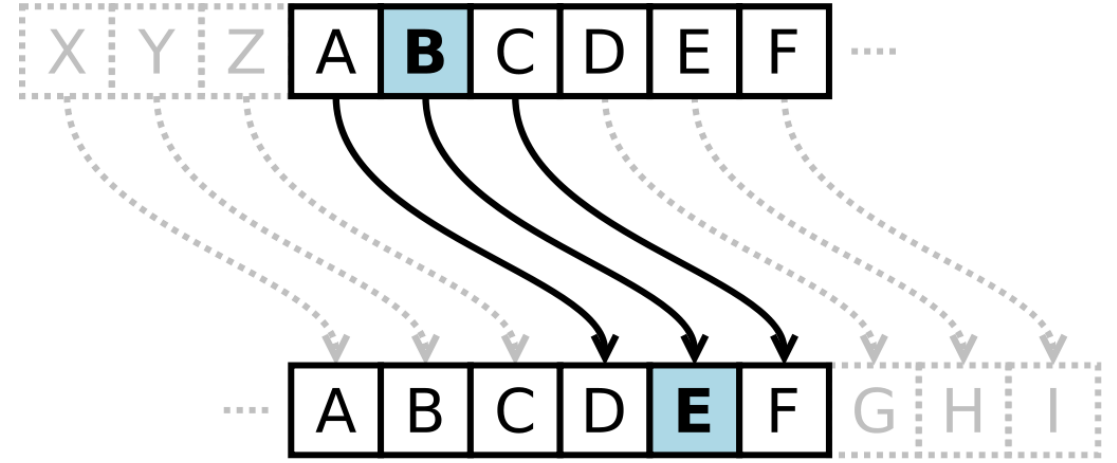
Contoh: Plainteks: MENGANTUK

Cipherteks: TNEAKMNGU

- Oleh karena itu, dikenal dua macam *cipher* di dalam kriptografi klasik:
 1. *Cipher* Substitusi (*substitution Cipher*)
 - metode enkripsi dan dekripsi menggunakan teknik substitusi
 2. *Cipher* Transposisi (*transposition Cipher*)
 - metode enkripsi dan dekripsi menggunakan teknik transposisi
- Kombinasi kedua teknik tersebut membentuk *product cipher* atau *super enkripsi*
$$\text{product cipher} = \text{cipher substitusi} + \text{cipher transposisi}$$

Cipher Substitusi

- Contoh yang terkenal: *Caesar Cipher*
- Tiap huruf alfabet digeser 3 huruf ke kanan



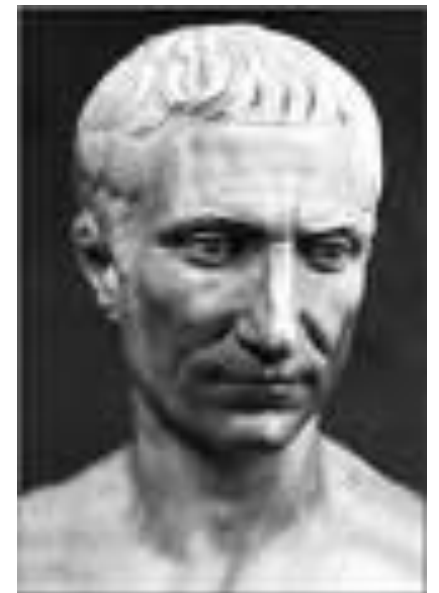
Plainteks : A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

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

- Contoh:

Plainteks: awasi asterix dan temannya obelix

Cipherteks: DZDVL DVWHULA GDQ WHPDQQBA REHOLA



- Supaya lebih aman, cipherteks dikelompokkan ke dalam kelompok n-huruf, misalnya kelompok 4-huruf:

Semula: DZDVL DVWHULA GDQ WHPDQQBA REHOLA

Menjadi: DZDV LDVW HULA GDQW HPDQ QBAR EHOL A

- Atau membuang semua spasi:

DZDVL DVWHULAGDQWHPDQQBAREHOLA

- Tujuannya agar proses kriptanalisis menjadi lebih sulit dilakukan



Caesar wheel untuk membentuk tabel substitusi huruf alfabet

- Misalkan setiap huruf alfabet dikodekan ke dalam integer dari 0 sampai 25 sebagai berikut:

$$A = 0,$$

$$B = 1,$$

$$C = 2,$$

...

$$Z = 25$$

maka, secara matematis Caesar Cipher dirumuskan sebagai:

$$\text{Enkripsi: } c = E(p) = (p + 3) \bmod 26$$

$$\text{Dekripsi: } p = D(c) = (c - 3) \bmod 26$$

Ket: p = plainteks; c = cipherteks

ENKRIPSI:

Plainteks: awasi asterix dan temannya obelix

- $p_1 = 'a' = 0 \rightarrow c_1 = E(0) = (0 + 3) \bmod 26 = 3 = 'D'$
- $p_2 = 'w' = 22 \rightarrow c_2 = E(22) = (22 + 3) \bmod 26 = 25 = 'Z'$
- $p_3 = 'a' = 0 \rightarrow c_3 = E(0) = (0 + 3) \bmod 26 = 3 = 'D'$
- $p_4 = 's' = 18 \rightarrow c_4 = E(18) = (18 + 3) \bmod 26 = 21 = 'V'$
- $p_5 = 'i' = 8 \rightarrow c_4 = E(8) = (8 + 3) \bmod 26 = 11 = 'L'$
- dst...

Cipherteks: DZDVL DVWHULA GDQ WHPDQQBA REHOLA

DEKRIPSI:

Cipherteks: DZDVL DVWHULA GDQ WHPDQQBA REHOLA

- $c_1 = 'D' = 3 \rightarrow p_1 = D(3) = (3 - 3) \bmod 26 = 0 = 'a'$
- $c_2 = 'Z' = 25 \rightarrow p_2 = D(25) = (25 - 3) \bmod 26 = 22 = 'w'$
- $c_3 = 'D' = 3 \rightarrow p_3 = D(3) = (3 - 3) \bmod 26 = 0 = 'a'$
- ...
- $c_{12} = 'A' = 0 \rightarrow p_{12} = D(0) = (0 - 3) \bmod 26 = -3 \bmod 26 = 23 = 'x'$

Keterangan: $-3 \bmod 26$ dihitung dengan cara berikut:

$|-3| \bmod 26 = 3$, lalu $26 - 3 = 23$

atau dengan cara: $(26 + 0 - 3) \bmod 26 = 23 \bmod 26 = 23$

- Plainteks ditemukan kembali: awasi asterix dan temannya obelix

- Jika pergeseran huruf sejauh k , maka:

Enkripsi: $c = E(p) = (p + k) \bmod 26$

Dekripsi: $p = D(c) = (c - k) \bmod 26$

$k =$ kunci rahasia

- Untuk 256 karakter ASCII, maka:

Enkripsi: $c = E(p) = (p + k) \bmod 256$

Dekripsi: $p = D(c_i) = (c - k) \bmod 256$

$k =$ kunci rahasia

```

/ Program enkripsi dan dekripsi pesan dengan Caesar Cipher
#include <iostream>
#include <string.h>
using namespace std;

void enkripsi()
{
    string plainteks, cipherteks;
    int i, k;
    char c;

    cout << "Ketikkan pesan:";
    cin.ignore(); getline (cin, plainteks);
    cout << "Masukkan jumlah pergeseran (0-25): "; cin >> k;
    cipherteks = ""; // inisialisasi cipherteks dengan null string

    for (i=0; i < plainteks.length(); i++) {
        c = plainteks[i];
        if (isalpha(c)) { //hanya memproses huruf alfabet saja
            c = toupper(c); // ubah menjadi huruf kapital
            c = c - 65; // kodekan huruf ke angka 0 s/d 25
            c = (c + k) % 26; // enkripsi, geser sejauh k ke kanan
            c = c + 65; // kodekan kembali ke huruf semula
        }
        cipherteks = cipherteks + c; // sambungkan ke cipherteks
    }
    cout << "Cipherteks: "<<cipherteks<< endl; // cetak cipherteks
}

```

```

void dekripsi()
{
    string plainteks, cipherteks;
    int i, k;
    char c;

    cout << "Ketikkan cipherteks: ";
    cin.ignore();getline (cin, cipherteks);
    cout << "Masukkan jumlah pergeseran (0-25): ";
    cin >> k;
    plainteks = ""; // inisialisasi plainteks dengan null string

    for (i=0; i < cipherteks.length(); i++) {
        c = cipherteks[i];
        if (isalpha(c)) { //hanya memproses alfabet
            c = toupper(c); // ubah karakter ke huruf besar
            c = c - 65; // kodekan huruf ke angka 0 s/d 25
            if (c - k < 0) // kasus pembagian bilangan negatif
                c = 26 + (c - k);
            else
                c = (c - k) % 26;
            c = c + 65; // kodekan kembali ke huruf semula
            c = tolower(c); // plainteks dinyatakan sebagai huruf kecil
        }
        plainteks = plainteks + c; // sambungkan ke plainteks
    }
    cout << "Plainteks: " << plainteks << endl; // cetak plainteks
}

```

```
main()
{
    int pil; bool stop;
    stop = false;

    while (!stop) {
        cout << "Menu: " << endl;
        cout << "1. Enkripsi " << endl;
        cout << "2. Dekripsi " << endl;
        cout << "3. Exit      " << endl;
        cout << "Pilih menu: "; cin >> pil;
        switch (pil) {
            case 1 : enkripsi(); break;
            case 2 : dekripsi(); break;
            case 3 : stop = true; break;
        }
    }
}
```

```
Command Prompt

C:\data\Dataku\Buku\Buku Kriptografi\Edisi kedua>caesar
Menu:
1. Enkripsi
2. Dekripsi
3. Exit
Pilih menu: 1
Ketikkan pesan: the quick brown fox jumps over the lazy dog
Masukkan jumlah pergesaran (0-25): 18
Cipherteks: LZW IMAUC TJGOF XGP BMEHK GNWJ LZW DSRQ UGY
Menu:
1. Enkripsi
2. Dekripsi
3. Exit
Pilih menu: 2
Ketikkan cipherteks: LZW IMAUC TJGOF XGP BMEHK GNWJ LZW DSRQ UGY
Masukkan jumlah pergesaran (0-25): 18
Plainteks: the quick brown fox jumps over the lazy dog
Menu:
1. Enkripsi
2. Dekripsi
3. Exit
Pilih menu: 3

C:\data\Dataku\Buku\Buku Kriptografi\Edisi kedua>
```

Kriptanalisis Caesar Cipher

- *Caesar cipher* mudah dipecahkan dengan *exhaustive key search (brute force)* karena jumlah kuncinya sangat sedikit (hanya ada 26 kunci).
- Coba lakukan dekripsi dengan berbagai nilai k dari 0 sampai 25, lalu periksa apakah hasil dekripsi merupakan kata atau kalimat yang bermakna. Jika ya, maka diduga k adalah kuncinya.
- Untuk memastikan k adalah kunci yang benar, maka cobakan k untuk potongan kriptogram lainnya.

Contoh: kriptogram XMZVH

Tabel 1. Contoh *exhaustive key search* terhadap cipherteks XMZVH

Kunci (k) <i>ciphering</i>	'Pesan' hasil dekripsi	Kunci (k) <i>ciphering</i>	'Pesan' hasil dekripsi	Kunci (k) <i>ciphering</i>	'Pesan' hasil dekripsi
0	XMZVH	17	GVIEQ	8	PERNZ
25	YNAWI	16	HWJFR	7	QFSOA
24	ZOBXJ	15	IXKGS	6	RGTPB
23	APCYK	14	JYLHT	5	SHUQC
22	BQDZL	13	KZMIU	4	TIVRD
21	CREAM	12	LANJV	3	UJWSE
20	DSFBN	11	MBOKW	2	VKXTF
19	ETGCO	10	NCPLX	1	WLYUG
18	FUHDP	9	ODQMY		

Plainteks yang potensial adalah CREAM dengan $k = 21$.

Kunci ini digunakan untuk mendekripsikan potongan cipherteks lainnya.

Contoh lain:

Cipherteks: PHHW PH DIWHU WKH WRJD SDUWB

```
PHHW PH DIWHU WKH WRJD SDUWB
k
0 phhw ph diwhu wkh wrjd sduwb
1 oggv og chvgt vjg vqic rctva
2 nffu nf bgufs uif uphb qbsuz
3 meet me after the toga party
4 ldds ld zesdq sgd snfz ozqsx
5 kccr kc ydrpc rfc rmey nyprw
6 ...
21 ummb um inbmz bpm bwoi xizbg
22 tlla tl hmaly aol avnh whyaf
23 skkz sk glzkx znk zumg vgxze
24 rjjy rj fkyjw ymj ytlf ufwyd
25 qiix qi ejxiv xli xske tevx
```

(Sumber: William Stallings)

Cipherteks: VIVBQ SQBI SMBMUC LQ ICTI

<i>k</i>	Hasil dekripsi
0	vivbq sqbi smb muc lq icti
1	uhuap rpah rlaltb kp hbsh
2	tgtzo qozg qkzksa jo garg
3	sfsyn pnyf pjyjrz in fzqf
4	rerxm omxe oixiqy hm eype
5	qdqwl nlwd nhwhpx gl dxod
6	pcpuk mkvc mgvgow fk cwnc
7	obouj ljub lfufnu ej bvmb
8	nanti kita ketemu di aula
9	mzmsh jhsz jdsdlt ch ztkz
10	lylrg igry icrcks bg ysjy
11	kxkqf hfqx hbqbjr af xrix
12	jwjpe gepw gapaiq ze wqhw
13	iviod fdov fzozhp yd vpgv
14	huhnc ecnu eynygo xc uofu
15	gtgmb dbmt dxmxfn wb tnet
16	fsfla calsc cwlwem va smds
17	erekz bzkr bvkvdL uz rlcR
18	dqdjy ayjq aujuck ty qkbq
19	cpcix zxip ztitbj sx pjap
20	bobhw ywho yshsai rw oizo
21	anagv xvgn xrfqyg pu mgxm
22	xmzfu wufm wqfqyg pu mgxm
23	ylyet vtel vpepxf ot lfwl
24	xkxds usdk uodowe ns kevk
25	wjwcr trcj tncnvd mr jduj

- Bagaimana jika terdapat dua atau lebih nilai k yang menghasilkan pesan-pesan bermakna?

Contoh: Misalkan kriptogram `HSPPW` menghasilkan dua kemungkinan kunci yang potensial, yaitu:

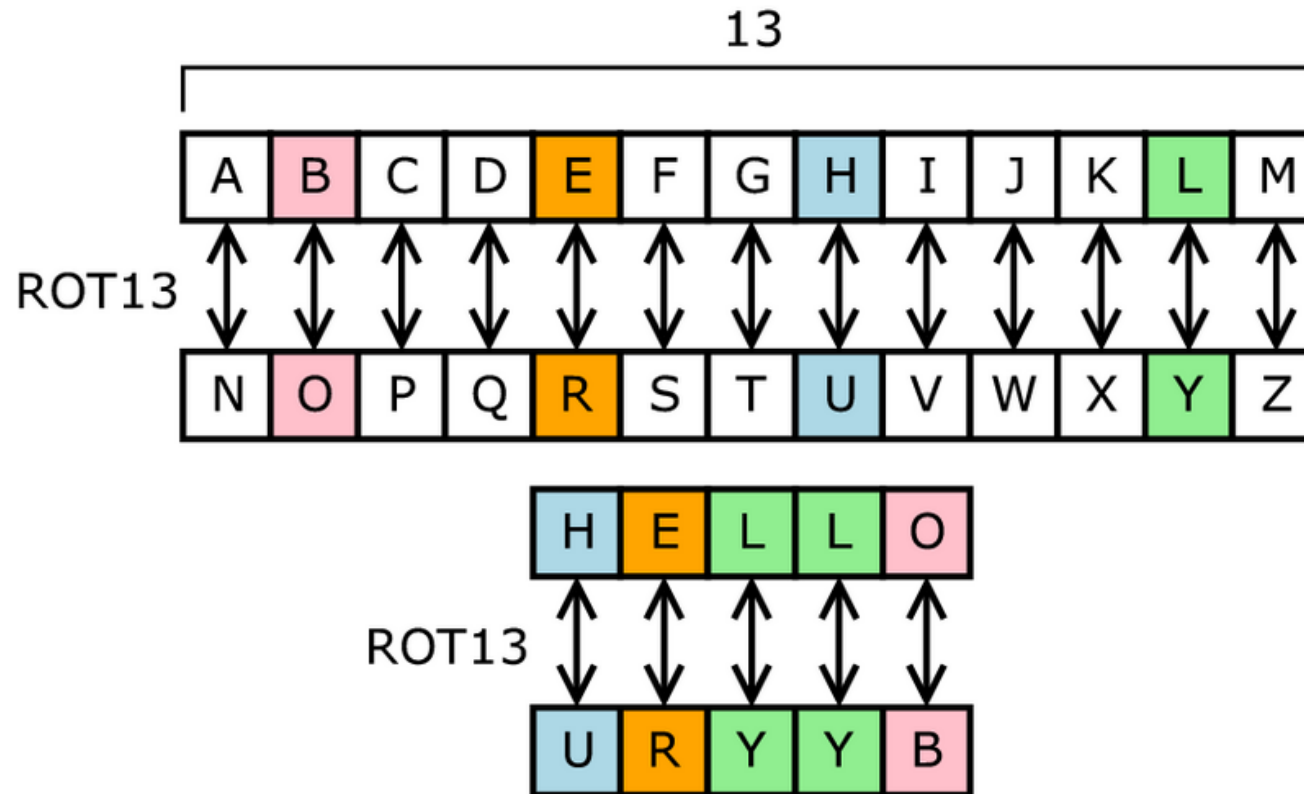
$k = 4$ menghasilkan pesan `dolls` (boneka)

$k = 11$ menghasilkan `wheel` (roda) .

Nilai k mana yang benar?

Jika kasusnya demikian, maka lakukan dekripsi terhadap potongan cipherteks lain tetapi cukup menggunakan $k = 4$ dan $k = 11$ agar dapat disimpulkan kunci mana yang benar.

- Di dalam sistem operasi Unix, ROT13 adalah fungsi menggunakan *Caesar cipher* dengan pergeseran $k = 13$



Sumber gambar: Wikipedia

- Contoh: ROT13 (ROTATE) = EBGNGR
- Nama “ROT13” berasal dari *net.jokes*
(<http://groups.google.com/group/net.jokes>) (tahun 1980)
- ROT13 biasanya digunakan di dalam forum *online* untuk menyandikan jawaban teka-teki, kuis, canda, dsb
- Enkripsi arsip dua kali dengan ROT13 menghasilkan pesan semula:

$$P = \text{ROT13}(\text{ROT13}(P))$$
 sebab $\text{ROT}_{13}(\text{ROT}_{13}(x)) = \text{ROT}_{26}(x) = x$
- Jadi dekripsi cukup dilakukan dengan mengenkripsi cipherteks kembali dengan ROT13

Cipher Transposisi

- Cipherteks diperoleh dengan mengubah posisi huruf di dalam plainteks.
- Dengan kata lain, algoritma ini melakukan *transpose* terhadap rangkaian huruf di dalam plainteks.
- Nama lain untuk metode ini adalah *cipher* **permutasi**, karena *transpose* setiap karakter di dalam teks sama dengan mempermutasikan karakter-karakter tersebut.

Contoh: Misalkan plainteks adalah

departemen teknik informatika itb

Enkripsi:

depart

emente

knikin

format

ikaitb

Cipherteks: (baca secara vertikal)

DEKFIEMNOKPEIRAANKMIRTIATTENTB (tanpa spasi)

DEKF IEMN OKPE IRAA NKMI RTIA TTEN TB (4 huruf)

Cipherteks: DEKFIEMNOKPEIRAANKMIRTIATTENTB

Dekripsi: Bagi panjang cipherteks dengan kunci.

(Pada contoh ini, $30 / 6 = 5$)

DEKFI

EMNOK

PEIRA

ANKMI

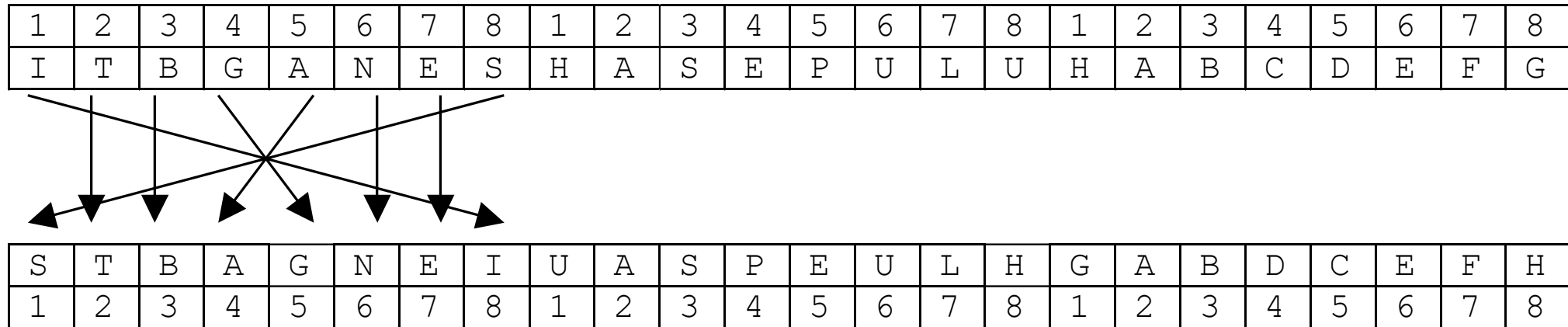
RTIAT

TENTB

Plainteks: (baca secara vertikal)

departemen teknik informatika itb

- Contoh lain: Plainteks: ITB GANESHA SEPULUH
- Bagi menjadi blok-blok 8-huruf. Jika < 8 , tambahkan huruf *dummy*.



- Cipherteks: **STBAGNEIUASPEULHGABDCEFH**

Contoh lain. Misalkan plainteks adalah

CRYPTOGRAPHY AND DATA SECURITY

Plainteks disusun menjadi 3 baris ($k = 3$) seperti di bawah ini:

C		T		A		A		A		E		I
R	P	O	R	P	Y	N	D	T	S	C	R	T
	Y		G		H		D		A		U	Y

maka cipherteksnya adalah

CTAAAEIRPORPYNDTSCR TYGHDAUY

Super-enkripsi

- Menggabungkan *cipher* substitusi dengan *cipher* transposisi.
- Disebut juga *product cipher*
- Mula-mula pesan dienkripsi dengan *cipher* substitusi, selanjutnya hasilnya dienkripsi dengan *cipher* transposisi (atau sebaliknya).

Contoh. Plainteks `hello world`

- dienkripsi dengan *caesar cipher* menjadi `KHOOR ZRUOG`
- kemudian hasil ini dienkripsi lagi dengan *cipher* transposisi ($k = 4$):

`KHOO`

`RZRU`

`OGZZ`

→ Cipherteks akhir adalah: **KROHZGORZOUZ**

- *Cipher* modern menggunakan konsep kombinasi *cipher* substitusi dan *cipher* transposisi, namun operasinya dibuat sekompleks mungkin

Bersambung