



Kriptografi

02. Implementasi Cipher Klasik

Kodrat Mahatma



Universitas Teknologi Digital

Tujuan Pembelajaran

- Memahami prinsip kerja cipher klasik.
- Mengimplementasikan enkripsi dan dekripsi sederhana.
- Menguji keamanan dasar cipher klasik.

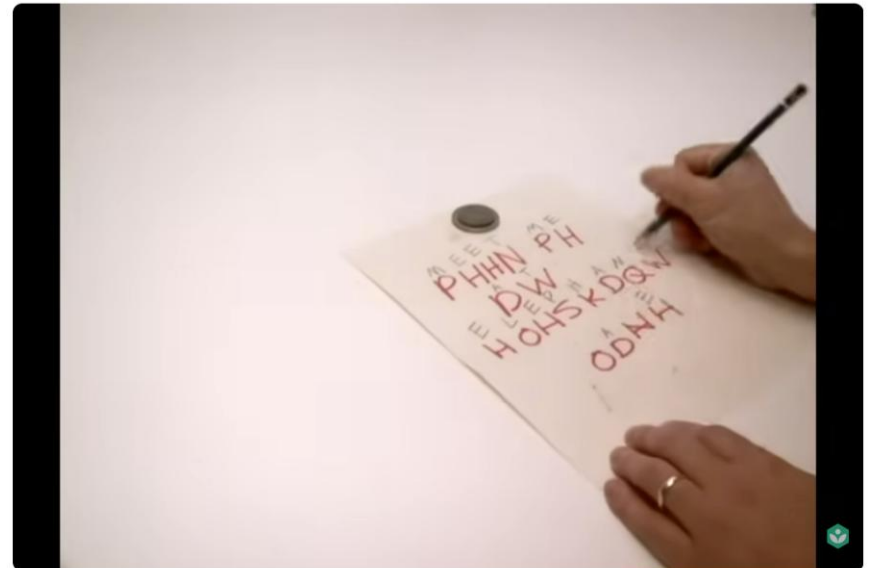
Konsep Dasar Cipher Klasik

Cipher klasik dibagi menjadi dua kategori utama:

1. Substitusi: mengganti simbol dengan simbol lain.
2. Transposisi: mengubah urutan huruf dalam pesan.

Caesar Cipher – Konsep

- Setiap huruf digeser sejauh n posisi.
- Contoh: shift = 3, A \rightarrow D, B \rightarrow E.
- Plaintext: HELLO \rightarrow Ciphertext: KHOOR



The Caesar cipher | Journey into cryptography | Computer Science | Khan Academy

<https://youtu.be/sMOZf4GN3oc?si=7PoYRYoPF5WdZF87>

Caesar Cipher – Python Implementation

```
def caesar_encrypt(text, shift):  
    result = ""  
    for char in text:  
        if char.isalpha():  
            base = ord('A') if char.isupper() else ord('a')  
            result += chr((ord(char) - base + shift) % 26 + base)  
        else:  
            result += char  
    return result  
  
print(caesar_encrypt('HELLO', 3))
```

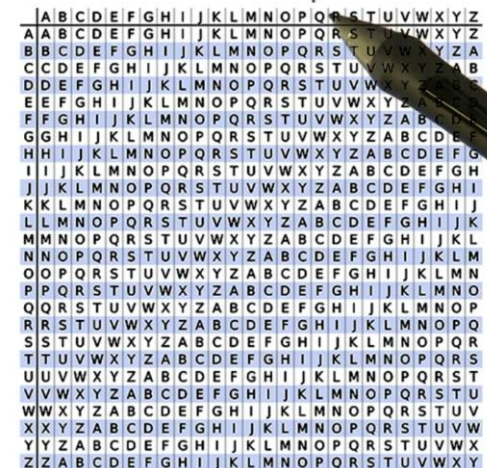
Vigenère Cipher – Konsep

- Menggunakan kata kunci untuk menentukan pergeseran setiap huruf.
- Contoh: Kunci = LEMON
- Plaintext = ATTACKATDAWN
- Ciphertext = LXFOPVEFRNHR

Vigenere Cipher

- Plaintext:
ATTACKATDAWN
- Key:
LEMON
- Keystream:
LEMONLEMONLE
- Ciphertext:
LXFOPVEFRNHR

Play K



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

Vigenere Cipher



Udacity
642K subscribers

Subscribe

6K



Share



Download

Clip

<https://youtu.be/SkJcmCaHqSo?si=A3JfwsoEQJJVhD4Z>

Vigenère Cipher – Python Implementation

```
def vigenere_encrypt(plain, key):  
    key = key.upper()  
    result = ""  
    for i, char in enumerate(plain.upper()):  
        if char.isalpha():  
            shift = ord(key[i % len(key)]) - 65  
            result += chr((ord(char) - 65 + shift) % 26 + 65)  
        else:  
            result += char  
    return result  
  
print(vigenere_encrypt('ATTACKATDAWN', 'LEMON'))
```

Affine Cipher – Konsep

- Rumus: $C = (aP + b) \bmod 26$
- Dengan a dan 26 harus relatif prima.
- Contoh: $a=5, b=8 \rightarrow P='A' \rightarrow C='I'$

Affine Cipher

To encrypt: $(ax + b) \bmod 26$

- a and b are both between 0 and 25
- a is coprime with 26 ($a \neq \text{even}$ and $\neq 13$)
- x represents A-Z mapped to the integers 0-25

To decrypt: $a^{-1}(x - b) \bmod 26$

- a^{-1} is the modular multiplicative inverse of $a \bmod 26$

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Play K

Cryptography: The Affine Cipher



Zach's Math Zone
1.25K subscribers

Subscribe

144



Share

Download

Clip



<https://youtu.be/iyESl17lqFQ?si=kAWgmmnaC7soQRvi>

https://youtu.be/sroLDJl98sY?si=T5YjehbeO9H_Q_F3

Affine Cipher – Python Implementation

```
def affine_encrypt(text, a, b):  
    result = ""  
    for char in text.upper():  
        if char.isalpha():  
            result += chr(((a * (ord(char) - 65) + b) % 26) + 65)  
        else:  
            result += char  
    return result  
  
print(affine_encrypt('HELLO', 5, 8))
```

Playfair Cipher – Konsep

- Menggunakan matriks 5x5 dari kata kunci.
- Langkah: bentuk pasangan huruf, substitusi sesuai posisi di tabel.
- Contoh: KEYWORD → tabel kunci, kemudian enkripsi per pasangan.

key = TREAT plaintext: running mate

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

How to Encrypt Text Using the Playfair Cipher

Zach's Math Zone
1.25K subscribers

Subscribe

25 Share Download Clip ...

<https://youtu.be/PrpwPjG3jt4?si=5XpjEvVSSHiNJWhP>

Playfair Cipher – Python (Skema Dasar)

```
def generate_table(key):  
    alphabet = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'  
    table = ""  
    for c in key.upper() + alphabet:  
        if c not in table:  
            table += c  
    return [table[i:i+5] for i in range(0,25,5)]
```

```
table = generate_table('KEYWORD')  
for row in table: print(row)
```

Hill Cipher – Konsep

- Berdasarkan aljabar linear.
- Gunakan matriks kunci ($n \times n$) dan operasi mod 26.
- $C = K \times P \pmod{26}$

How to encrypt text using the Hill Cipher:

- 1) Map the plaintext to corresponding numbers 0 - 25.
- 2) Create a $n \times n$ invertible "key" matrix that contains numbers 0 - 25.
- 3) Multiply the key matrix by the plaintext vector.
- 4) Perform modulus 26 on resulting vector.

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Use the Hill Cipher to encrypt the following plaintext: car

How to Encrypt Text Using the Hill Cipher



Zach's Math Zone
1.25K subscribers

Subscribe

17



Share

Download

Clip



<https://youtu.be/uRRYf2f7kYA?si=Fpl1gjAbuUXhLVR>

Hill Cipher – Python (2x2)

```
import numpy as np
```

```
def hill_encrypt(text, key):  
    text = text.upper().replace(' ', '')  
    n = int(len(key)**0.5)  
    key = np.array(key).reshape(n, n)  
    result = ''  
    for i in range(0, len(text), n):  
        block = [ord(c) - 65 for c in text[i:i+n]]  
        cipher = np.dot(key, block) % 26  
        result += ''.join(chr(c + 65) for c in cipher)  
    return result
```

```
print(hill_encrypt('TEST', [3,3,2,5]))
```

Perbandingan Cipher Klasik

- Caesar: mudah dipecahkan dengan brute force.
- Vigenère: lebih kuat, tapi rentan terhadap analisis frekuensi.
- Hill: bergantung pada matriks kunci invertibel.
- Affine: kombinasi linear sederhana.

Cipher	Karakteristik	Kelebihan	Kelemahan	Penjelasan Singkat
Caesar Cipher	Pergeseran huruf tetap (misal +3)	Sangat mudah dipahami dan diimplementasikan	Mudah dipecahkan dengan brute force karena hanya 25 kemungkinan kunci	Setiap huruf digeser dengan jumlah tetap. Contoh: A→D, B→E.
Vigenère Cipher	Menggunakan kunci berupa kata, dengan pergeseran berbeda per huruf	Lebih kuat dari Caesar karena pergeseran bergantung pada kunci	Rentan terhadap analisis frekuensi jika kuncinya pendek atau berulang	Polialfabetik cipher; tiap huruf plaintext dikombinasikan dengan huruf kunci.
Hill Cipher	Berdasarkan operasi matriks mod 26	Sulit dipecahkan tanpa mengetahui matriks kunci	Hanya dapat digunakan jika matriks kunci invertibel (memiliki invers)	Mengubah huruf menjadi vektor dan mengenkripsi dengan perkalian matriks.
Affine Cipher	Kombinasi linear dari Caesar Cipher ($E(x) = ax + b \text{ mod } 26$)	Sedikit lebih kuat dari Caesar	Masih mudah dipecahkan karena bersifat linear sederhana	Kombinasi substitusi linier dengan dua parameter a dan b yang memenuhi $\text{gcd}(a, 26) = 1$.

Aktivitas Praktikum

1. Implementasikan dua cipher klasik menggunakan Python.
2. Buat input/output file teks.
3. Bandingkan hasilnya dengan CrypTool atau CyberChef.

Diskusi

- Apa kelemahan utama cipher klasik di era modern?
- Bagaimana cara menambah keamanan tanpa mengganti algoritma sepenuhnya?

Transisi ke Cipher Modern

- Cipher modern (AES, RSA) menyelesaikan kelemahan klasik dengan matematika modular kompleks, panjang kunci besar, dan operasi bit-level.

Tugas Mini

Buat program sederhana yang:

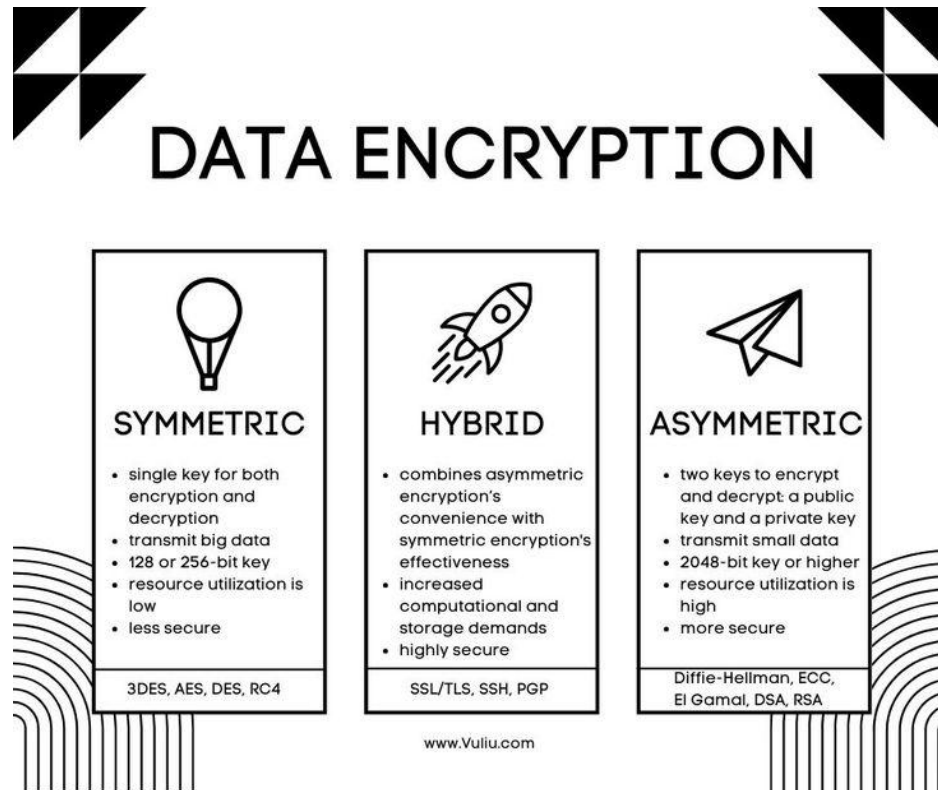
- Menerapkan Caesar & Vigenère Cipher.
- Mampu enkripsi dan dekripsi otomatis.
- Menyimpan hasil ke file .txt.

Sumber Belajar

- <https://www.cryptool.org/en/>
- **Cryptography for Everybody**
<https://www.youtube.com/c/nilsretrohobbyroom>
- AI
- Dan lain-lain

Penutup

- Cipher klasik adalah pondasi untuk memahami kriptografi modern.
- Selanjutnya (minggu depan): Implementasi DES dan AES.





Selamat belajar !

'Cryptography is the
mathematics of trust.'