LAPORAN IMPLEMENTASI VIGNERE CHIPER

ANALISIS FREKUENSI

Sebagai bentuk penugasan di Mata Kuliah Kriptografi

Dosen Pengampu: Kodrat Mahatma

Disusun oleh: Kelompok 5

Siti Fatimah (20123062)

Jesi Rosyanti (20123053)

PROGRAM STUDI INFOMATIKA S1

UNIVERSITAS TEKNOLOGI DIGITAL

2025

A. Pendahuluan

Salah satu algoritma kriptografi klasik yang cukup dikenal adalah Vigenère Cipher. Cipher ini metode enkripsi yang menggunakan kata kunci (key) untuk menentukan jumlah pergeseran tiap huruf pada plaintext. Setiap huruf pada kunci akan menentukan nilai geseran huruf pada teks asli, sehingga pergeserannya tidak tetap. Pada tugas ini dilakukan implementasi Vigenère Cipher menggunakan bahasa pemrograman Python, serta analisis frekuensi pada ciphertext yang dihasilkan. Analisis frekuensi ini membantu memahami pola huruf yang muncul dan menunjukkan tingkat keamanan cipher.

B. Landasan Teori

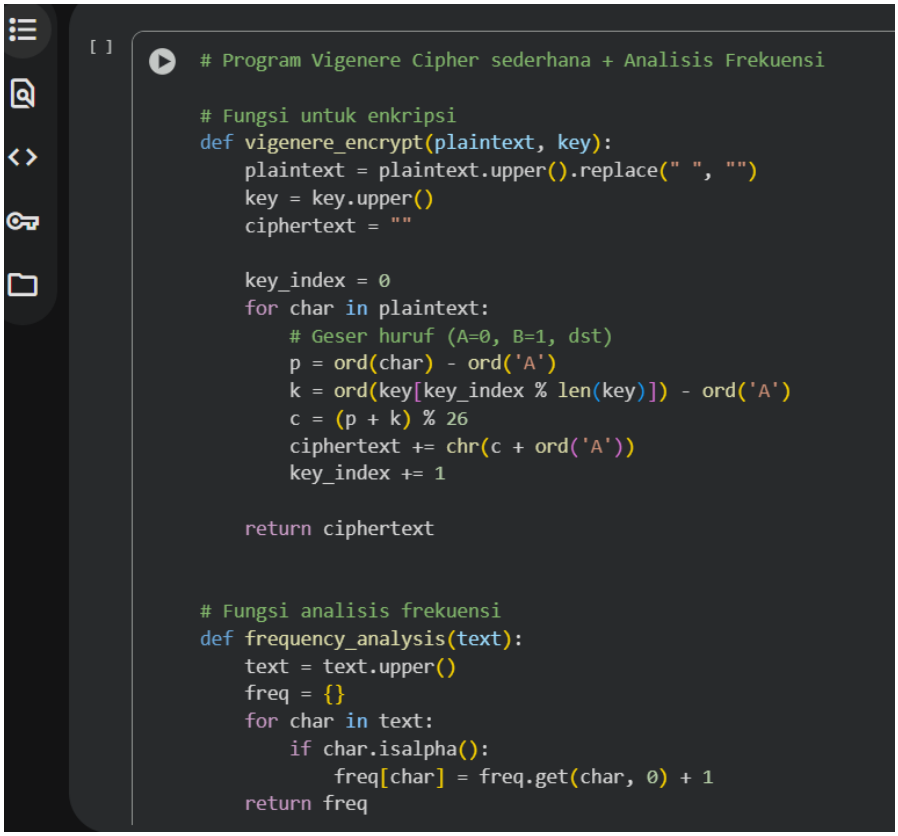
Vignere Cipher termasuk ke dalam polyalphabetic subatitution cipher, yaitu proses enkripsinya dengan cara:

- Mengubah huruf plaintext dan key menjadi indeks 0–25 (A=0, B=1, dst).
- Menambahkan indeks plaintext dan key secara modulo 26.

Adapun rumusnya: **C = (P + K) mod 26**

C. Implementasi Program

Berikut kode yang digunakan pada Google Collab:



```
[ ]
# Program Vigenere Cipher sederhana + Analisis Frekuensi

# Fungsi untuk enkripsi
def vigenere_encrypt(plaintext, key):
    plaintext = plaintext.upper().replace(" ", "")
    key = key.upper()
    ciphertext = ""

    key_index = 0
    for char in plaintext:
        # Geser huruf (A=0, B=1, dst)
        p = ord(char) - ord('A')
        k = ord(key[key_index % len(key)]) - ord('A')
        c = (p + k) % 26
        ciphertext += chr(c + ord('A'))
        key_index += 1

    return ciphertext


# Fungsi analisis frekuensi
def frequency_analysis(text):
    text = text.upper()
    freq = {}
    for char in text:
        if char.isalpha():
            freq[char] = freq.get(char, 0) + 1
    return freq
```

```python
# ==== INPUT ====
plaintext = input("Masukkan plaintext: ")
key = input("Masukkan key: ")

# ==== PROSES ====
ciphertext = vigenere_encrypt(plaintext, key)
freq = frequency_analysis(ciphertext)

# ==== OUTPUT ====
print("\n=== HASIL ENKRIPSI ===")
print("Ciphertext :", ciphertext)

print("\n=== ANALISIS FREKUENSI ===")
for huruf, jumlah in sorted(freq.items()):
    print(f"{huruf} : {jumlah}")
```

D. Hasil dan Analisis

1. Hasil

```
Masukkan plaintext: BELAJAR KRIPTOGRAFI
Masukkan key: JESI

=== HASIL ENKRIPSI ===
Ciphertext : KIDISEJSAMHBXKJIOM

=== ANALISIS FREKUENSI ===
A : 1
B : 1
D : 1
E : 1
H : 1
I : 3
J : 2
K : 2
M : 2
O : 1
S : 2
X : 1
```

2. Analisis Frekuensi

Semisal kita ambil salah satu huruf sesuai di dalam tabel berikut:

Huruf	Frekuensi
K	5
Q	3
T	3
B	2
F	2
I	2

Interpretasi:

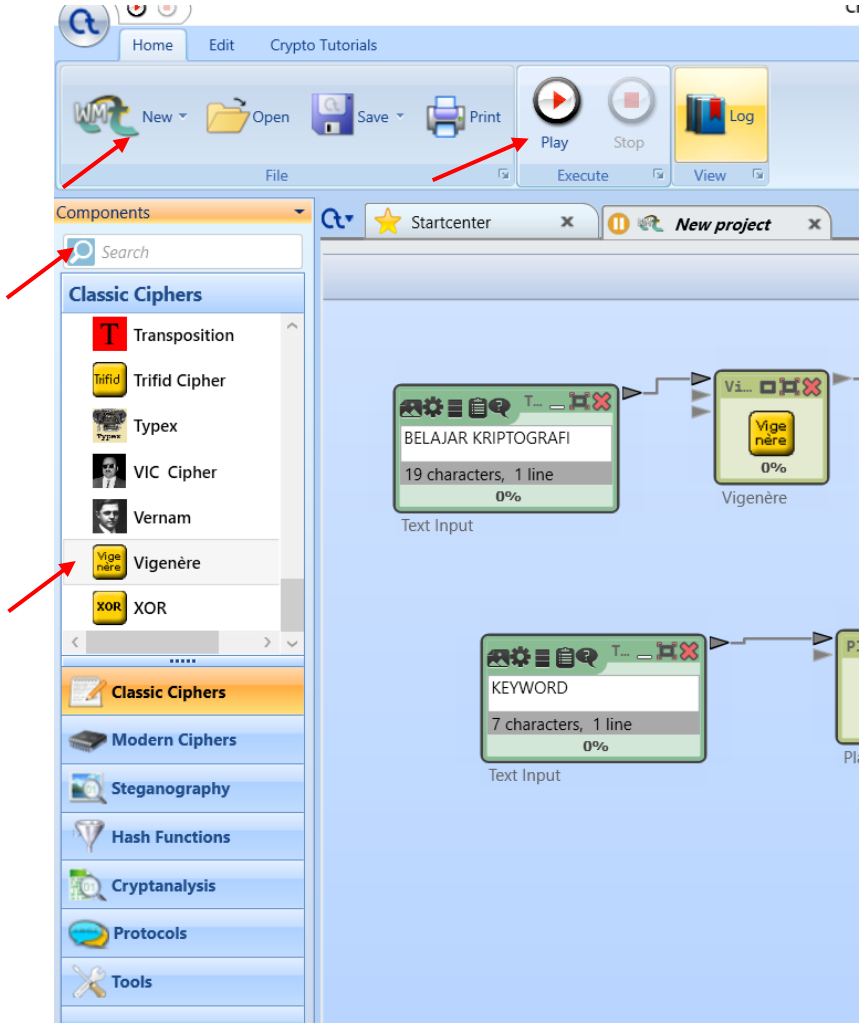
- Distribusi huruf pada ciphertext tidak merata, namun lebih acak dibandingkan cipher substitusi mono-alfabet.
- Hal ini menunjukkan Vigenère lebih kuat terhadap analisis frekuensi sederhana, tetapi jika panjang kunci diketahui, maka cipher dapat dipatahkan. Juga jika ciphertext sangat panjang, pola huruf dari key dapat dianalisis menggunakan Kasiski Examination.

E. Validasi dengan CrypTool

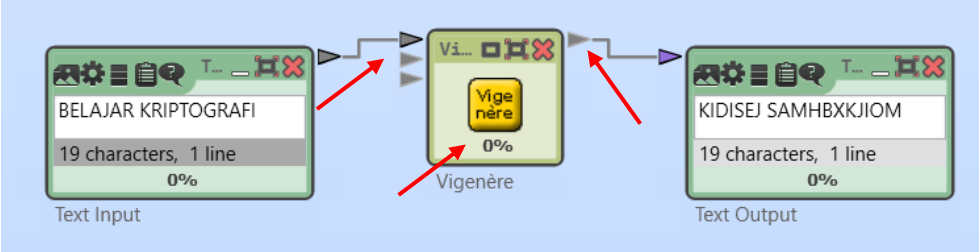
Tahapan singkat penggunaan CrypTool yaitu:

a. Buka aplikasi CrypTool 2

b. Klik ‘New’ di pojok kiri atas

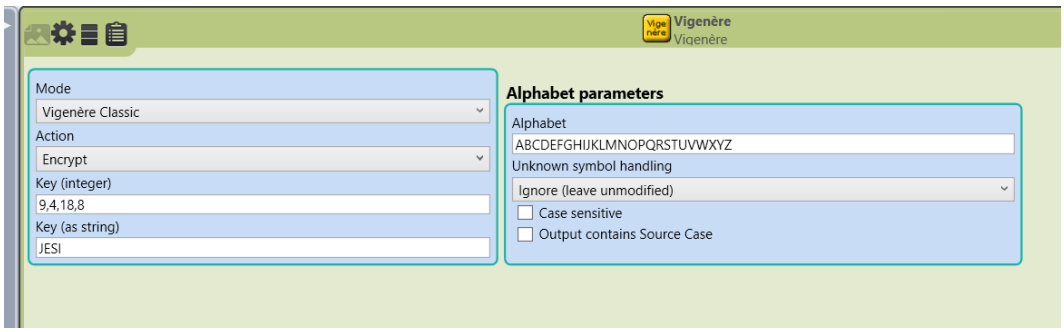


c. Di samping kiri ada tampilan pilihan menu Cipher dan pilih Vigenere atau bisa langsung di cari di fitur pencariannya.



d. Tarik simbol panah di sebelah kiri untuk memasukan teks inputannya dan sebelah kanan untuk hasil outputnya.

e. Untuk memasukkan kata kuncinya kita bisa klik 2x alat kerja Vigenere tersebut.



Di bagian setting ini kita cari '**Key (as string)**' dan masukkan kata kuncinya. Di sini kata kunci 'JESI' sudah dimasukkan.

f. Kembali ke halaman layar kerja dan klik tombol 'Play'dan 'Stop' di atas untuk mendapatkan hasilnya.

Berdasarkan hasil yang diperoleh, Vigenère Cipher berhasil diimplementasikan menggunakan Python dengan benar dan menunjukkan program telah bekerja sesuai teori. Namun, keamanan cipher ini masih dapat dilemahkan melalui analisis panjang kunci dan teknik kriptoanalisis seperti metode Kasiski.