TUGAS 2 KEAMANAN KOMPUTER/KRIPTOGRAFI

TUGAS KELOMPOK

DISUSUN OLEH :

NAMA :

1. MUH. NURKHALIS | 222110

2. USWATUN KHASANAH | 222305

3. NUR AFIFAH NAJWA | 222308

KELAS : 5TKKO-F

JURUSAN : TEKNIK INFORMATIKA

UNIVERSITAS DIPA MAKASSAR

2024/2025

- SOURCE CODE

```
port base64
from tkinter import DISABLED, NORMAL, Button, Entry, Label, Tk, filedialog, messagebox

from Crypto.Cipher import AES
from Crypto.Util.Padding import pad, unpad


def encrypt_data(data, key):
    key = key.encode("utf-8")
    while len(key) < 32:
        key += b" "
    key = key[:32]
    cipher = AES.new(key, AES.MODE_CBC)
    ct_bytes = cipher.encrypt(pad(data, AES.block_size))
    iv = base64.b64encode(cipher.iv).decode("utf-8")
    ct = base64.b64encode(ct_bytes).decode("utf-8")
    return iv, ct


def decrypt_data(iv, ct, key):
    iv = base64.b64decode(iv)
    ct = base64.b64decode(ct)
    key = key.encode("utf-8")
    while len(key) < 32:
        key += b" "
    key = key[:32]
    cipher = AES.new(key, AES.MODE_CBC, iv)
    pt = unpad(cipher.decrypt(ct), AES.block_size)
    return pt


def choose_file():
    file_path = filedialog.askopenfilename()
    if file_path:
        file_entry.delete(0, "end")
        file_entry.insert(0, file_path)


def on_encrypt():
    key = key_entry.get()
    file_path = file_entry.get()
    text_data = text_entry.get()
    if not key or (not file_path and not text_data):
        messagebox.showerror("Error", "Kunci atau file/teks belum diisi!")
        return
    try:
        if text_data:
            iv, cipher_text = encrypt_data(text_data.encode("utf-8"), key)
            result_entry.delete(0, "end")
            result_entry.insert(0, cipher_text)
            iv_entry.delete(0, "end")
            iv_entry.insert(0, iv)
        elif file_path:
            with open(file_path, "rb") as file:
                data = file.read()
            iv, cipher_text = encrypt_data(data, key)
            result_entry.delete(0, "end")
            result_entry.insert(0, cipher_text)
            iv_entry.delete(0, "end")
            iv_entry.insert(0, iv)
        save_button.config(state=NORMAL)
    except Exception as e:
        messagebox.showerror("Error", f"Terjadi kesalahan: {str(e)}")


```

```python

def on_decrypt():
    key = key_entry.get()
    cipher_text = result_entry.get()
    iv = iv_entry.get()
    if not key or not cipher_text or not iv:
        messagebox.showerror("Error", "Kunci, IV, atau cipherteks tidak diisi!")
        return
    try:
        decrypted_data = decrypt_data(iv, cipher_text, key)
        try:
            decoded_data = decrypted_data.decode("utf-8")
            messagebox.showinfo("Hasil Dekripsi", f"Plainteks: {decoded_data}")
        except UnicodeDecodeError:
            save_path = filedialog.asksaveasfilename(defaultextension=".bin")
            if save_path:
                with open(save_path, "wb") as file:
                    file.write(decrypted_data)
                messagebox.showinfo(
                    "Berhasil", f"Data biner telah disimpan di {save_path}"
                )
    except Exception as e:
        messagebox.showerror("Error", f"Terjadi kesalahan: {str(e)}")


def save_encrypted():
    cipher_text = result_entry.get()
    iv = iv_entry.get()
    if not cipher_text or not iv:
        messagebox.showerror("Error", "Tidak ada cipherteks untuk disimpan!")
        return
    save_path = filedialog.asksaveasfilename(defaultextension=".txt")
    if save_path:
        try:
            with open(save_path, "w", encoding="utf-8") as file:
                file.write(f"IV: {iv}\nCiphertext: {cipher_text}")
            messagebox.showinfo("Berhasil", f"Cipherteks telah disimpan di {save_path}")
        except Exception as e:
            messagebox.showerror("Error", f"Terjadi kesalahan: {str(e)}")


def reset_fields():
    key_entry.delete(0, "end")
    text_entry.delete(0, "end")
    file_entry.delete(0, "end")
    result_entry.delete(0, "end")
    iv_entry.delete(0, "end")
    save_button.config(state=DISABLED)


root = Tk()
root.title("AES Enkripsi dan Dekripsi")
root.geometry("600x500")
Label(root, text="Masukkan Kunci:").pack(pady=5)
key_entry = Entry(root, width=50, show="*")
key_entry.pack(pady=5)
Label(root, text="Masukkan Teks untuk Enkripsi / Dekripsi:").pack(pady=5)
text_entry = Entry(root, width=50)
text_entry.pack(pady=5)
Label(root, text="Pilih File untuk Enkripsi / Dekripsi:").pack(pady=5)
file_entry = Entry(root, width=50)
file_entry.pack(pady=5)
choose_button = Button(root, text="Pilih File", command=choose_file)
choose_button.pack(pady=5)
Label(root, text="Ciphertext (Hasil Enkripsi):").pack(pady=5)
result_entry = Entry(root, width=50)
result_entry.pack(pady=5)
Label(root, text="IV (Initialization Vector):").pack(pady=5)
iv_entry = Entry(root, width=50)
iv_entry.pack(pady=5)
encrypt_button = Button(root, text="Enkripsi", command=on_encrypt)
encrypt_button.pack(pady=10)
decrypt_button = Button(root, text="Dekripsi", command=on_decrypt)
decrypt_button.pack(pady=10)
save_button = Button(
    root, text="Simpan Hasil Enkripsi", state=DISABLED, command=save_encrypted
)
save_button.pack(pady=10)
reset_button = Button(root, text="Reset", command=reset_fields)
reset_button.pack(pady=10)
root.mainloop()

```

- Tampilan GUI