**TUGAS PERTEMUAN 10**

**Kriptografi**

****

Prodi Teknik Informatika

Universitas Pelita Bangsa

Matkul Kriptografi

Muhammad Najamuddin Dwi Miharja, S.Kom, M.kom

Yusuf Putra Bintang Satria ~ 312110317

TI.21.A2

**MEMBUAT PROGRAM DENGAN ONE TIME PAD DIIMPLEMENTASIKAN KE STEGANO**

• **Pembahasan**:

1. **One Time Pad (OTP**

* One Time Pad (OTP) merupakan sebuah algoritma kriptografi kunci simetris yang menggunakan kunci sepanjang pesan yang sama, dan panjangnya setara atau lebih panjang dari pesan itu sendiri.
* OTP dianggap sebagai metode kriptografi yang sangat aman jika kunci dipertahankan dengan baik dan hanya digunakan satu kali.

• **Kelebihan**:

* Keamanan Tinggi: Jika digunakan dengan benar, OTP dapat memberikan tingkat keamanan yang sangat tinggi karena kunci yang digunakan hanya digunakan satu kali.
* Kunci yang Berkualitas: Kunci yang dihasilkan oleh OTP adalah benar-benar acak dan setara atau lebih panjang dari pesan, membuatnya sulit untuk dipecahkan dengan serangan kriptoanalisis.

• **Kekurangan:**

* Pengelolaan Kunci Sulit: Kunci harus dihasilkan dan dibagikan dengan aman sebelum komunikasi dimulai. Pengelolaan kunci ini bisa menjadi tantangan, terutama jika pembagian kunci tidak aman.
* Penggunaan yang Canggih: Penerapan OTP yang benar membutuhkan disiplin yang tinggi, terutama dalam pengelolaan kunci dan proses enkripsi/dekripsi, yang dapat menjadi rumit untuk digunakan dalam situasi kehidupan nyata.

• **Struktur File:**

* Struktur file pada OTP sebagian besar bergantung pada implementasinya. Namun, umumnya terdapat dua bagian utama: pesan terenkripsi dan kunci.
* File pesan terenkripsi berisi hasil enkripsi pesan dengan menggunakan OTP, sedangkan file kunci berisi kunci yang digunakan untuk enkripsi tersebut.

• **Catatan Tambahan:**

* Pemakaian yang Efektif: OTP umumnya lebih efektif pada situasi di mana pesan yang dikirimkan singkat dan kunci dapat dengan aman dibagikan sebelumnya.
* Tidak Praktis untuk Semua Penggunaan: Meskipun sangat aman, OTP tidak selalu praktis untuk semua penggunaan sehari-hari karena kompleksitas pengelolaan kunci dan ukuran kunci yang besar.

Penting untuk dicatat bahwa keamanan dari OTP sangat tergantung pada implementasi dan manajemen kunci yang dilakukan. Kesalahan dalam manajemen kunci atau penggunaan kunci yang tidak benar dapat mengurangi keamanan metode ini. - Kedua program menggunakan struktur yang serupa, dengan perbedaan sintaksis antara PHP dan Python.

* Fungsi-fungsi utama melakukan operasi konversi dan enkripsi secara berurutan.
* Input dari pengguna digunakan untuk menyediakan plainteks dan kunci.
* Output mencakup hasil enkripsi dan dekripsi yang ditampilkan kepada pengguna.

1. Steganografi:

* Pembahasan:

Steganografi adalah seni atau ilmu menyembunyikan pesan atau data sensitif dalam suatu media yang tampaknya biasa atau umumnya tidak memicu kecurigaan, sehingga pihak ketiga yang tidak berkepentingan tidak menyadari keberadaan pesan tersebut. Dalam konteks kriptografi, teknik ini sering disebut sebagai "steganografi kriptografis" atau "stegano".

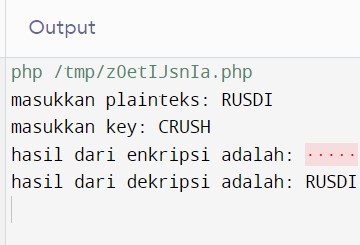
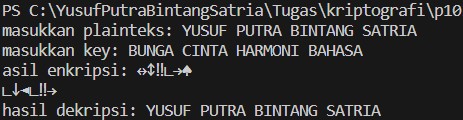
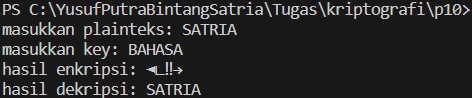
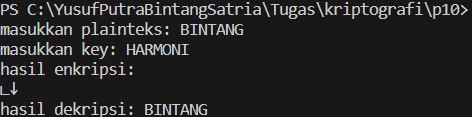
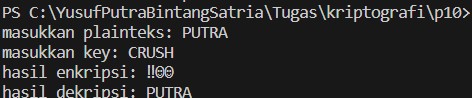
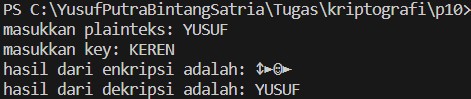
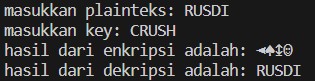
* Kelebihan:
  + Kerahasiaan yang Tinggi: Steganografi memberikan tingkat kerahasiaan yang tinggi karena pesan tersembunyi tidak mudah terdeteksi.
  + Ketahanan terhadap Pemecahan: Dibandingkan dengan enkripsi konvensional, steganografi bisa lebih tahan terhadap upaya pemecahan karena pihak ketiga mungkin tidak tahu di mana mencari pesan tersembunyi.
  + Ketidakberlanjutan: Dalam beberapa kasus, jika pesan tersembunyi berhasil diidentifikasi, itu tidak selalu merusak integritas pesan.
* Kekurangan:
* Kapasitas Terbatas: Media penyimpanan yang digunakan untuk menyembunyikan pesan memiliki batas kapasitas. Steganografi sering kali tidak cocok untuk menyembunyikan data yang sangat besar.
* Sensitivitas terhadap Modifikasi: Beberapa teknik steganografi dapat menjadi rentan terhadap modifikasi pada media yang menyebabkan kehilangan pesan atau pengungkapan pesan tersembunyi.
* Kesulitan Deteksi: Kekurangan dari sudut pandang keamanan; pesan tersembunyi mungkin sulit untuk dideteksi oleh metode konvensional.
* Struktur File:

Struktur file dalam konteks steganografi tergantung pada media yang digunakan. Secara umum, steganografi dapat diimplementasikan pada berbagai jenis file, termasuk gambar, audio, video, atau dokumen teks. Contoh struktur file untuk gambar mungkin melibatkan penyisipan data tambahan ke dalam bit atau byte yang kurang signifikan, tanpa secara signifikan merusak kualitas visual gambar tersebut.

Sebagai contoh, pada gambar, setiap pixel mungkin digunakan untuk menyimpan beberapa bit tambahan yang tidak terlihat oleh mata manusia. Struktur file tersebut sangat bervariasi tergantung pada metode steganografi yang digunakan dan media yang dipilih untuk menyembunyikan pesan.

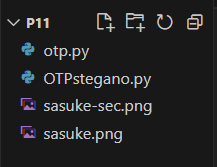
Implementasi steganografi dapat menggunakan berbagai algoritma, seperti LSB (Least Significant Bit) untuk gambar, atau modifikasi pada nilai-nilai frekuensi dalam domain frekuensi untuk audio atau video. Setiap metode memiliki struktur file yang khas sesuai dengan karakteristik media penyimpanan tersebut.

* **Dan berikut adalah hasil dari programnya :**
* Otp :

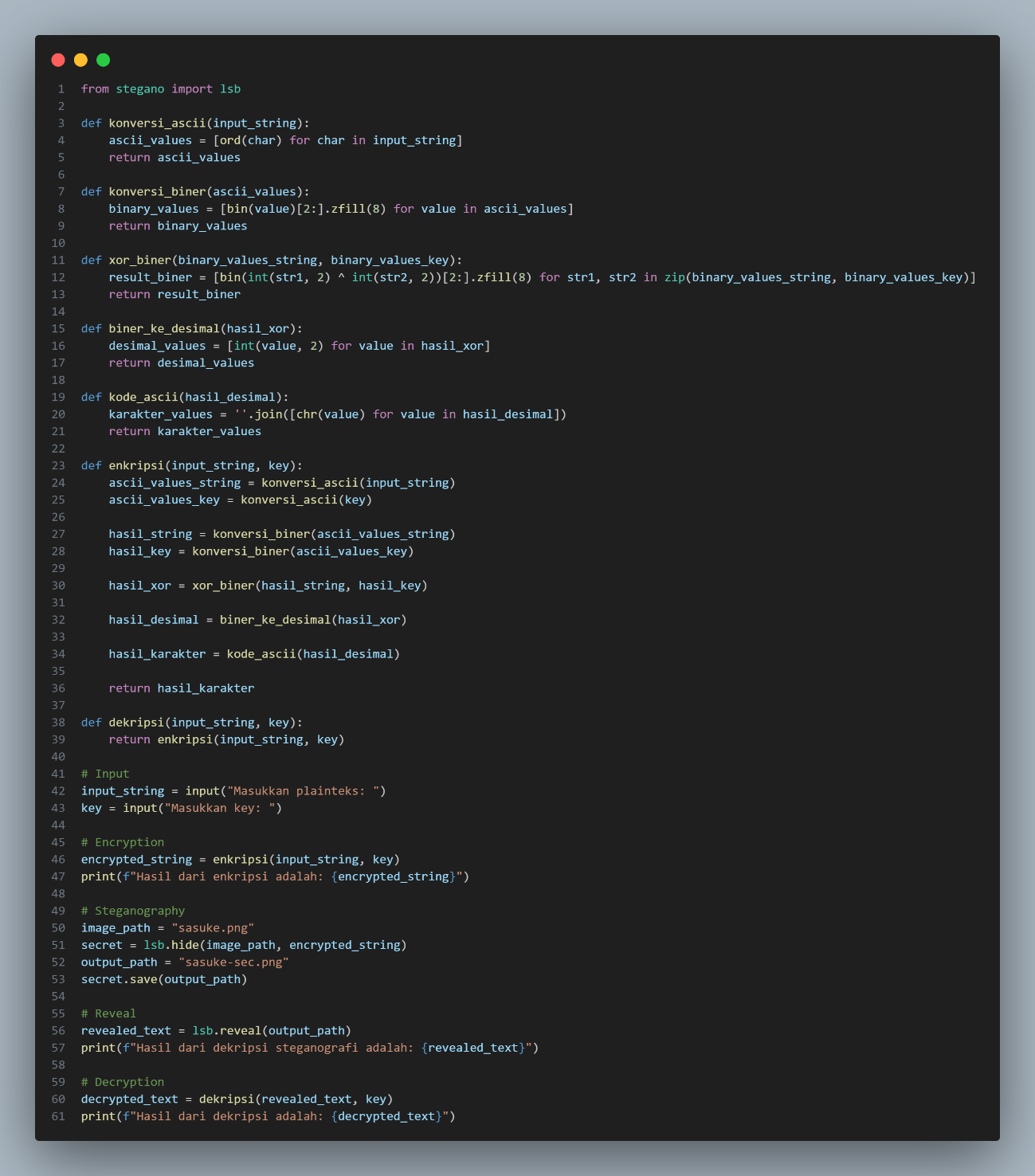


* Otpstegano :

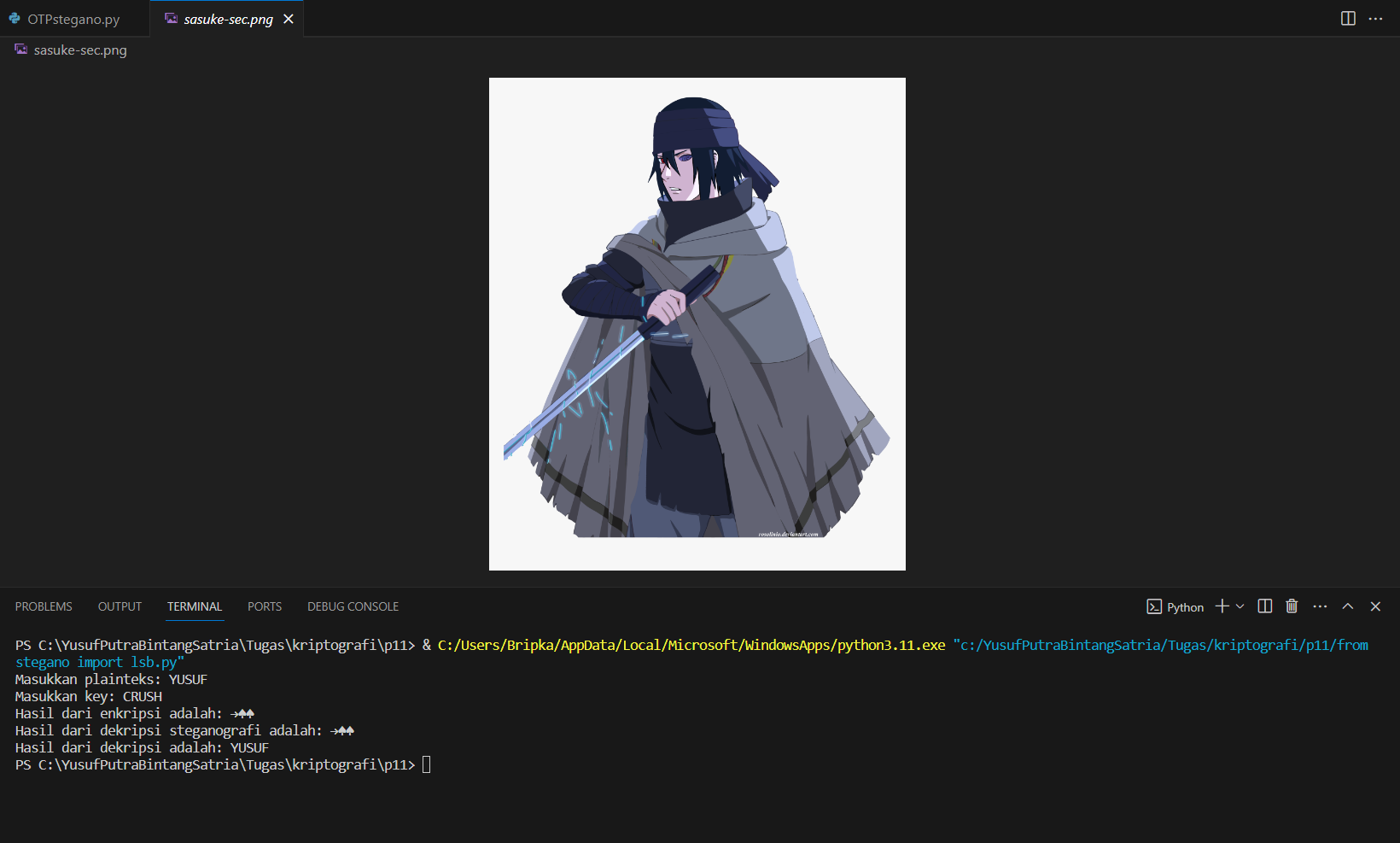
1. Berikut struktur filenya:



1. Kode programnya:



1. Hasil dari programnya:



* **Rangkuman Program One Time Pad (OTP):**

**1. OTP.PHP:**

* Program ini ditulis dalam bahasa PHP.
* Mempunyai fungsi-fungsi dasar, seperti konversi ASCII, konversi ke biner, operasi XOR biner, konversi biner ke desimal, dan konversi desimal ke karakter ASCII.
* Terdapat fungsi enkripsi dan dekripsi menggunakan OTP.
* Menggunakan input dari pengguna untuk plainteks dan kunci.
* Output program mencakup hasil enkripsi dan dekripsi.

**2. OTP.PY:**

* Program ini ditulis dalam bahasa Python.
* Fungsi-fungsi utama serupa dengan versi PHP, dengan konversi ASCII, konversi ke biner, operasi XOR biner, konversi biner ke desimal, dan konversi desimal ke karakter ASCII.
* Implementasi enkripsi dan dekripsi menggunakan OTP.
* Menggunakan input dari pengguna untuk plainteks dan kunci.
* Output program mencakup hasil enkripsi dan dekripsi.

**3. Catatan Tambahan:**

* Kedua program menggunakan struktur yang serupa, dengan perbedaan sintaksis antara PHP dan Python.
* Fungsi-fungsi utama melakukan operasi konversi dan enkripsi secara berurutan.
* Input dari pengguna digunakan untuk menyediakan plainteks dan kunci.
* Output mencakup hasil enkripsi dan dekripsi yang ditampilkan kepada pengguna.

**4. Kesimpulan:**

* Kedua program tersebut menyajikan implementasi sederhana dari algoritma OTP dalam bahasa PHP dan Python.
* Meskipun kedua program memiliki tujuan yang sama, perbedaan sintaksis dan struktur bahasa membuat mereka tampak berbeda, namun inti logika dari keduanya tetap serupa.
* Program ini mengimplementasikan enkripsi dan dekripsi dengan menggunakan One-Time Pad (OTP) serta menyisipkan pesan terenkripsi ke dalam gambar menggunakan steganografi Least Significant Bit (LSB). Berikut adalah rangkuman dari setiap bagian program:

1. Konversi ASCII:

- Fungsi `konversi\_ascii` mengambil string input dan mengonversinya menjadi daftar nilai ASCII.

2. Konversi Biner:

- Fungsi `konversi\_biner` mengambil daftar nilai ASCII dan mengonversinya ke dalam daftar nilai biner dengan panjang tetap 8 bit.

3. XOR Biner:

- Fungsi `xor\_biner` melakukan operasi XOR pada setiap bit dari nilai biner string input dengan nilai biner kunci.

4. Konversi Biner ke Desimal:

- Fungsi `biner\_ke\_desimal` mengonversi hasil XOR dari biner ke desimal.

5. Konversi ASCII ke Karakter:

- Fungsi `kode\_ascii` mengambil nilai desimal dan mengonversinya kembali menjadi string karakter.

6. Enkripsi:

- Fungsi `enkripsi` menggunakan OTP dengan melakukan konversi ASCII, XOR, dan konversi kembali untuk menghasilkan teks terenkripsi.

7. Dekripsi:

- Fungsi `dekripsi` menggunakan OTP dan merupakan operasi yang sama dengan enkripsi.

8. Input:

- Meminta pengguna untuk memasukkan plaintext dan kunci

9. Enkripsi:

- Melakukan enkripsi plaintext menggunakan OTP.

10. Steganografi (LSB):

- Menyembunyikan teks terenkripsi dalam gambar menggunakan LSB.

11. Reveal:

- Mengungkapkan teks tersembunyi dari gambar yang telah dimodifikasi

12. Dekripsi:

- Melakukan dekripsi pada teks tersembunyi menggunakan OTP.

Program ini mencampurkan kriptografi OTP dengan steganografi LSB, yang secara umum meningkatkan tingkat keamanan pesan. Namun, penggunaan OTP memerlukan kunci yang sama panjang dengan pesan, dan penggunaannya yang benar-benar acak dapat menjadi tantangan dalam implementasi praktis.

• **Berikut adalah code dari program diatas :**

https://github.com/YusufPutraBintangSatria/OTP\_Stegano\_Chipper.git