

Kelompok :

1. Andrey Hartawan Suwardi
2. Ivone Liwang
3. Franklin Jaya
4. Michael Sawitto

sistem kerja dari kelas `Shiapcrypto` berdasarkan kode Python, dengan fokus pada proses enkripsi dan dekripsi menggunakan kunci rahasia `AFVM`.

Sistem Kerja Kelas `Shiapcrypto`

1. Inisialisasi Kunci Rahasia

Ketika kelas `Shiapcrypto` diinisialisasi dengan kunci rahasia `AFVM`, metode `generate_keys` akan memproses setiap karakter dari kunci ini dan mengubahnya menjadi nilai ASCII:

- `A` → `65`
- `F` → `70`
- `V` → `86`
- `M` → `77`

Nilai ini disimpan dalam variabel:

- `B1 = 65`
- `B2 = 70`
- `B3 = 86`
- `B4 = 77`

2. Proses Enkripsi

Misalkan teks yang ingin dienkripsi adalah `HELLO`.

Langkah-langkah Enkripsi:

Untuk setiap karakter dari `plaintext`, dilakukan langkah berikut:

- Karakter H :
 - Nilai ASCII: `ord('H') = 72`
 - Proses XOR bertahap:
``plaintext
(((72 ^ 65) ^ 70) ^ 86) ^ 77
72 ^ 65 = 9
9 ^ 70 = 79
79 ^ 86 = 25
25 ^ 77 = 84

...

- Hasil akhir: `\chr(84) = 'T'`

- Karakter E :

- Nilai ASCII: `\ord('E') = 69`

- Proses XOR bertahap:

```plaintext`

$((69 \wedge 65) \wedge 70) \wedge 86 \wedge 77$

$69 \wedge 65 = 4$

$4 \wedge 70 = 66$

$66 \wedge 86 = 20$

$20 \wedge 77 = 89$

...

- Hasil akhir: `\chr(89) = 'Y'`

- Karakter L (pertama):

- Nilai ASCII: `\ord('L') = 76`

- Proses XOR bertahap:

```plaintext`

$((76 \wedge 65) \wedge 70) \wedge 86 \wedge 77$

$76 \wedge 65 = 13$

$13 \wedge 70 = 75$

$75 \wedge 86 = 29$

$29 \wedge 77 = 80$

...

- Hasil akhir: `\chr(80) = 'P'`

- Karakter L (kedua, sama dengan yang pertama):

- Proses XOR yang sama menghasilkan: `'P'`

- Karakter O :

- Nilai ASCII: `\ord('O') = 79`

- Proses XOR bertahap:

```plaintext`

$((79 \wedge 65) \wedge 70) \wedge 86 \wedge 77$

$79 \wedge 65 = 14$

$14 \wedge 70 = 72$

$72 \wedge 86 = 30$

$30 \wedge 77 = 83$

...

- Hasil akhir: `\chr(83) = 'S'`

Jadi, teks terenkripsi dari `"HELLO"` adalah `"TYPPS"`.

### 3. Proses Dekripsi

Setelah dienkripsi menjadi "TYPPE", proses dekripsi bekerja dengan cara yang sama, karena XOR bersifat simetris. Ini berarti, kita cukup menggunakan nilai ASCII dari karakter terenkripsi dan menjalankan proses XOR dengan kunci-kunci dalam urutan terbalik ('B4', 'B3', 'B2', 'B1') untuk mendapatkan kembali teks asli.

# Langkah-langkah Dekripsi:

- Karakter T :
  - Nilai ASCII: ``ord('T') = 84``
  - Proses XOR bertahap:

```
```plaintext
(((84 ^ 77) ^ 86) ^ 70) ^ 65
84 ^ 77 = 25
25 ^ 86 = 79
79 ^ 70 = 9
9 ^ 65 = 72
```
```
  - Hasil akhir: ``chr(72) = 'H'``
- Karakter Y :
  - Nilai ASCII: ``ord('Y') = 89``
  - Proses XOR bertahap:

```
```plaintext
(((89 ^ 77) ^ 86) ^ 70) ^ 65
89 ^ 77 = 20
20 ^ 86 = 66
66 ^ 70 = 4
4 ^ 65 = 69
```
```
  - Hasil akhir: ``chr(69) = 'E'``
- Karakter P (pertama):
  - Nilai ASCII: ``ord('P') = 80``
  - Proses XOR bertahap:

```
```plaintext
(((80 ^ 77) ^ 86) ^ 70) ^ 65
80 ^ 77 = 29
29 ^ 86 = 75
75 ^ 70 = 13
13 ^ 65 = 76
```
```
  - Hasil akhir: ``chr(76) = 'L'``

- Karakter P (kedua, hasil yang sama): 'L'

- Karakter S :

- Nilai ASCII: ``ord('S') = 83``

- Proses XOR bertahap:

```plaintext`

`(((83 ^ 77) ^ 86) ^ 70) ^ 65`

`83 ^ 77 = 30`

`30 ^ 86 = 72`

`72 ^ 70 = 14`

`14 ^ 65 = 79`

````

- Hasil akhir: ``chr(79) = 'O'``

Jadi, teks terenkripsi "TYPPE" berhasil didekripsi kembali menjadi "HELLO".

#### Kesimpulan Sistem Kerja

- Proses enkripsi dan dekripsi menggunakan operasi XOR pada nilai ASCII dari karakter.

- Setiap karakter diubah menjadi nilai ASCII menggunakan ``ord()``, di-XOR secara bertahap dengan empat kunci ('B1', 'B2', 'B3', 'B4'), lalu diubah kembali menjadi karakter menggunakan ``chr()``.

- Algoritma yang sama digunakan untuk enkripsi dan dekripsi. (Kata kunci : XOR Binary untuk mendapatkan hasil yang di xor)

Link Collab : [Projectes.ipynb](https://colab.research.google.com/drive/1_aFdhZDpo-3DeK1nXRMqakrEddN3pZDk?usp=sharing)

[https://colab.research.google.com/drive/1\\_aFdhZDpo-3DeK1nXRMqakrEddN3pZDk?usp=sharing](https://colab.research.google.com/drive/1_aFdhZDpo-3DeK1nXRMqakrEddN3pZDk?usp=sharing)