

# XOR-based Vigenere Cipher

| Nama        | Ahmad Mu'min Faisal   |
|-------------|---|
| NIM         | 1203210101  |
| Kelas       | IF-01-02  |
| Link Github | <a href="https://github.com/fzl-22/xor-based-vigenere-cipher">https://github.com/fzl-22/xor-based-vigenere-cipher</a> |

## 1 Deskripsi

Project ini merupakan project Vigenere Cipher berbasis operasi XOR yang dibuat menggunakan bahasa C dan CMake build system generator. Cipher ini memanfaatkan operasi XOR dari bit-bit setiap karakter dari sebuah plain text dengan bit-bit setiap karakter dari key-nya.

## 2 Cara Menjalankan Program

Pastikan CMake sudah terinstall di sistem operasi. Kemudian, clone repositori project.

```
git clone git@github.com:fzl-22/xor-based-vigenere-cipher.git
```

Kemudian, masuk ke direktori project.

```
cd xor-based-vigenere-cipher
```

Buat direktori bernama `build` dan navigasi ke folder tersebut.

```
mkdir build && cd build
```

Setelah itu, buat Makefile dari file `CMakeLists.txt` dengan perintah berikut.

```
cmake -G "Unix Makefiles" ..
```

Setelah selesai, lakukan kompilasi.

```
make
```

Terakhir jalankan program dengan format perintah

```
./examples/vigenere <input_string> <key_string>
```

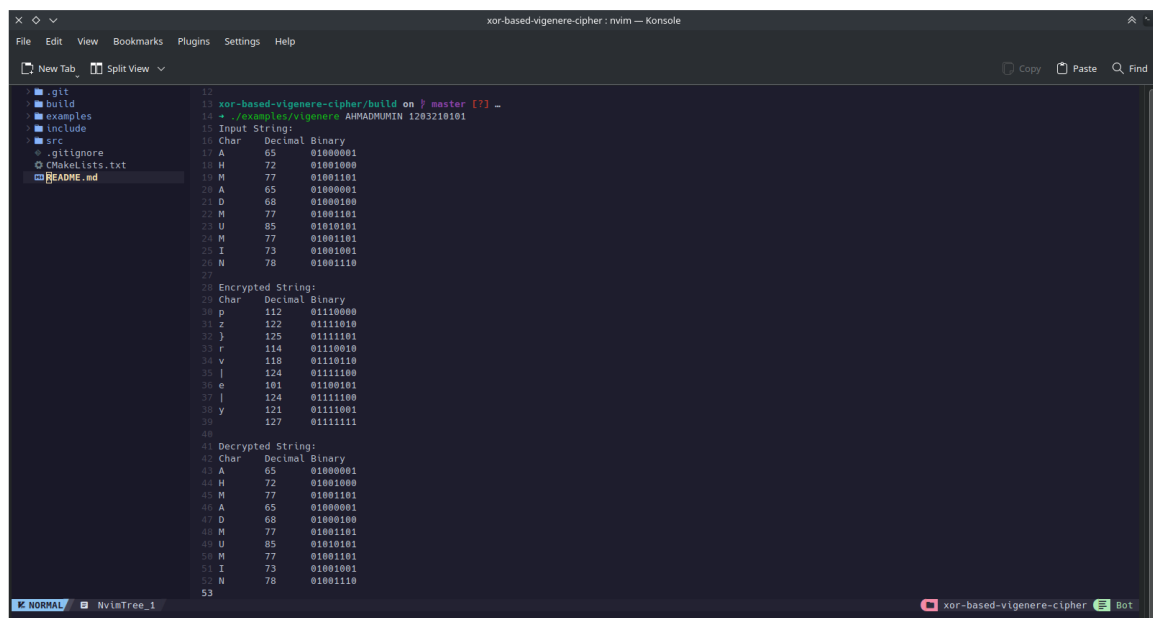
## 3 Percobaan

Sebagai percobaan, jalankan program dengan input dan key berikut:

```
INPUT : AHMADMUMIN  
KEY   : kijasik
```

Karena panjang string `kijasik` harus sama dengan panjang string `AHMADMUMIN`, maka key akan menjadi `kijasikkij`.

Outputnya menjadi seperti berikut:



```
xor-based-vigenere-cipher: nvim — Konsole
File Edit View Bookmarks Plugins Settings Help
New Tab Split View Copy Paste Find
12
13 xor-based-vigenere-cipher/build on } master [?] ~
14 + ./examples/vigenere AHMADMUMIN 1283210101
15 Input String:
16 Char Decimal Binary
17 A 65 01000001
18 H 72 01001000
19 M 77 01001101
20 A 65 01000001
21 D 68 01000100
22 M 77 01001101
23 U 85 01010101
24 M 77 01001101
25 I 73 01001001
26 N 78 01001110
27
28 Encrypted String:
29 Char Decimal Binary
30 p 112 01110000
31 z 122 01111010
32 j 125 01111101
33 r 114 01110010
34 v 118 01110110
35 i 124 01111100
36 e 101 01100101
37 y 124 01111100
38 y 121 01111001
39
40 127 01111111
41
42 Decrypted String:
43 Char Decimal Binary
44 A 65 01000001
45 H 72 01001000
46 M 77 01001101
47 A 65 01000001
48 D 68 01000100
49 M 77 01001101
50 U 85 01010101
51 M 77 01001101
52 I 73 01001001
53 N 78 01001110
54
```

## 4 Penjelasan

Di dalam program, output karakter dari string yang ditampilkan terbagi menjadi 3 format. Yaitu `Char`, `Decimal`, dan `Binary`. Berikut adalah penjelasan lebih detailnya:

Char : karakter dari string yang diolah oleh program  
Decimal : representasi desimal dari karakter terkait di ASCII table.  
Binary : representasi biner dari bilangan desimal terkait

Berikut adalah kode ASCII dari desimal 0 hingga 127. Untuk lebih lengkapnya, kunjungi <https://www.ascii-code.com/>.

```
114
115 xor-based-vigenere-cipher/build on ? master [?] ...
116 → ascii -d
117  0 NUL    16 DLE    32      48 0     64 @     80 P     96 `    112 p
118  1 SOH    17 DC1    33 !     49 1     65 A     81 Q     97 a    113 q
119  2 STX    18 DC2    34 "     50 2     66 B     82 R     98 b    114 r
120  3 ETX    19 DC3    35 #     51 3     67 C     83 S     99 c    115 s
121  4 EOT    20 DC4    36 $     52 4     68 D     84 T    100 d    116 t
122  5 ENQ    21 NAK    37 %     53 5     69 E     85 U    101 e    117 u
123  6 ACK    22 SYN    38 &     54 6     70 F     86 V    102 f    118 v
124  7 BEL    23 ETB    39 '     55 7     71 G     87 W    103 g    119 w
125  8 BS     24 CAN    40 (     56 8     72 H     88 X    104 h    120 x
126  9 HT     25 EM     41 )     57 9     73 I     89 Y    105 i    121 y
127 10 LF     26 SUB    42 *     58 :     74 J     90 Z    106 j    122 z
128 11 VT     27 ESC    43 +     59 ;     75 K     91 [    107 k    123 {
129 12 FF     28 FS     44 ,     60 <     76 L     92 \    108 l    124 |
130 13 CR     29 GS     45 -     61 =     77 M     93 ]    109 m    125 }
131 14 SO     30 RS     46 .     62 >     78 N     94 ^    110 n    126 ~
132 15 SI     31 US     47 /     63 ?     79 O     95 _    111 o    127 DEL
133
```

Sehingga, input string `AHMADMUMIN` akan dicetak seperti berikut:

Plain text: "AHMADMUMIN"

Input String:

| Char | Decimal | Binary                                |
|------|---------|---------------------------------------|
| A    | 65      | 01000001 # representasi biner dari 65 |
| H    | 72      | 01001000 # representasi biner dari 72 |
| M    | 77      | 01001101 # dan seterusnya ...         |
| A    | 65      | 01000001                              |
| D    | 68      | 01000100                              |
| M    | 77      | 01001101                              |
| U    | 85      | 01010101                              |
| M    | 77      | 01001101                              |
| I    | 73      | 01001001                              |
| N    | 78      | 01001110                              |

Karena key-nya adalah `kijasikkij`, maka akan menjadi seperti ini (tidak dicetak oleh program):

Key: "kijasikkij"

Key String:

| Char | Decimal | Binary                                 |
|------|---------|--|
| k    | 107     | 01101011 # representasi biner dari 107 |
| i    | 105     | 01101001 # representasi biner dari 105 |
| j    | 106     | 01101010 # dan seterusnya ...          |
| a    | 97      | 01100001                               |
| s    | 115     | 01110011                               |
| i    | 105     | 01101001                               |
| k    | 107     | 01101011                               |
| k    | 107     | 01101011                               |
| i    | 105     | 01101001                               |
| j    | 106     | 01101010                               |

Algoritma enkripsi Vigenere Cipher berbasis XOR (dipanggil dengan fungsi `vigenere_cipher`) akan melakukan operasi XOR dari setiap karakter input string dan key string secara bitwise. Sebagai contoh, ambil karakter ke-6 dari input string (konsekuensinya, akan dilakukan operasi XOR secara bitwise dengan karakter ke-6 dari key string). Sehingga, akan dilakukan operasi bitwise XOR antara karakter **M** dengan **i**.

```
M ^ i = 77 ^ 105
      = 01001101 ^ 01101001
      = 00100100 # desimal 36, karakter $
```

Bilangan biner **00100100** merupakan representasi dari bilangan desimal **36**. Dimana dalam ASCII table, nilai desimal **36** merepresentasikan karakter **\$**. Proses ini terus berlanjut untuk semua karakter, sehingga akan menampilkan hasil enkripsinya adalah:

Hasil enkripsi: `"*!' 7$>& $"`

Encrypted String:

| Char | Decimal | Binary   |
|------|---------|----------|
| *    | 42      | 00101010 |
| !    | 33      | 00100001 |
| '    | 39      | 00100111 |
|      | 32      | 00100000 |
| 7    | 55      | 00110111 |
| \$   | 36      | 00100100 |
| >    | 62      | 00111110 |
| &    | 38      | 00100110 |
|      | 32      | 00100000 |
| \$   | 36      | 00100100 |

Terdapat hasil yang unik ketika bilangan binernya adalah 00100000 (desimal 32 ). Dalam ASCII table, desimal 32 mewakili simbol SP atau spasi. Apabila hasil enkripsi ini didekripsi ulangan dengan cara memanggil fungsi `vigenere_cipher` lagi, maka hasil dekripsinya adalah seperti di bawah ini.

Hasil dekripsi: "AHMADMUMIN"

Decrypted String:

| Char | Decimal | Binary   |
|------|---------|----------|
| A    | 65      | 01000001 |
| H    | 72      | 01001000 |
| M    | 77      | 01001101 |
| A    | 65      | 01000001 |
| D    | 68      | 01000100 |
| M    | 77      | 01001101 |
| U    | 85      | 01010101 |
| M    | 77      | 01001101 |
| I    | 73      | 01001001 |
| N    | 78      | 01001110 |

Proses dekripsi akan melalui cara yang sama dengan proses enkripsi, yaitu melakukan operasi bitwise XOR pada setiap karakter dari string hasil enkripsi dengan key-nya. Sebagai contoh, ambil karakter ke-6 dari string hasil enkripsi dan key, maka prosesnya adalah sebagai berikut:

```
$ ^ i = 36 ^ 105
      = 00100100 ^ 01101001
      = 01001101 # desimal 77, karakter M
```

Sehingga akan didapatkan karakter ke-6 dari string hasil enkripsi kembali menjadi M . Proses ini berlanjut untuk semua karakter pada string hasil enkripsi.

Informasi Tambahan: apabila karakter hasil enkripsi berada pada desimal di luar range 32-127, maka karakter yang berkaitan tidak dapat ditampilkan karena tidak termasuk dalam *ASCII printable characters*. Misalnya, apabila hasil enkripsinya memiliki representasi desimal 9, maka tidak akan ditampilkan karakter (melainkan sebuah spasi horizontal yang cukup lebar). Hal ini dikarenakan desimal 9 merupakan representasi desimal dari HT, yaitu *Horizontal Tab*.

## 5 Kesimpulan

---

Dengan ini, dapat disimpulkan bahwa Vigenere Cipher berbasis XOR merupakan *symmetric encryption* karena proses enkripsi dan dekripsi tetap menggunakan key yang sama. Modifikasi algoritma enkripsi klasik menjadi enkripsi modern berbasis bit tetap sama secara esensi.