Modul 5

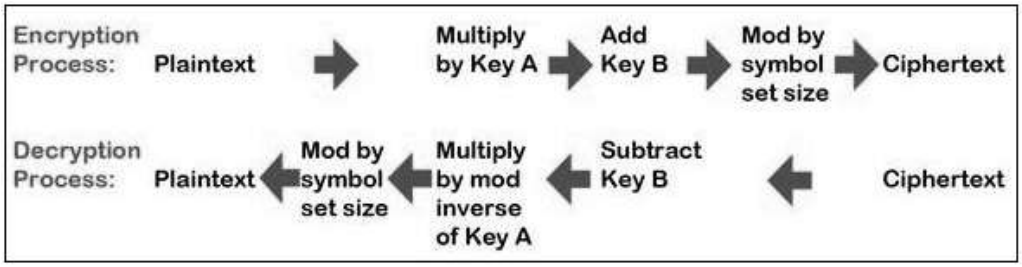
A. Tujuan

1. Mampu menjelaskan definisi dan konsep Affine Cipher
2. Mampu memahami konsep dasar Affine Cipher

B. Dasar teori

1. Affine Cipher

Cipher affine adalah jenis cipher substitusi monoalphabetic. Setiap karakter dipetakan ke ekuivalen numeriknya, dienkripsi dengan fungsi matematika dan kemudian dikonversi ke huruf yang berkaitan dengan nilai numerik barunya. Meskipun semua cipher monoalphabetic lemah, cipher affine jauh lebih kuat daripada cipher lainnya, karena memiliki lebih banyak kunci. Ilustrasi proses enkripsi dan dekripsi dari algoritma affine cipher.



Nilai pada kunci A memiliki batasan karena harus berkoprima dengan m (banyak karakter yang digunakan). Koprima adalah hubungan antara dua bilangan bulat yang memiliki FPB/GCD 1 atau saling prima. Kemungkinan nilai a adalah 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, dan 25 jika kita menggunakan karakter sebanyak 26. Jika kita menggunakan kunci A yang tidak saling prima dengan m, maka proses dekripsi tidak dapat dilakukan. Berikut adalah beberapa cara untuk mencari FPB atau GCD (Greatest Common Divisor)

- GCD in EXCEL : =GCD(kunci A; jumlah karakter m)
  Contoh : =GCD(3,26) akan menghasilkan nilai 1
- GCD in PHP : gmp_gcd(GMP|int|string $num1, GMP|int|string $num2): GMP
  Fungsi diatas terdiri dari 2 paramter yaitu num1 dan num2. Fungsi tersebut selalu menghasilkan nilai positif. Contoh kode :

Contoh 1. PHP GCD menggunakan while

```php
<?php
$p = $x = 3;
$q = $y = 26;

while ($x != $y) {
  if ($x > $y)
    $x = $x - $y;
  else
    $y = $y - $x;
}

echo "GCD dari $p dan $q adalah: $x";
//hasil = GCD dari 3 dan 26 adalah 1
?>
```

GCD dari 3 dan 26 adalah: 1

Contoh 2. PHP GCD menggunakan for

```php
<?php
$x = 3;
$y = 26;

if ($x > $y) {
  $temp = $x;
  $x = $y;
  $y = $temp;
}

for($i = 1; $i < ($x+1); $i++) {
  if ($x%$i == 0 and $y%$i == 0)
    $gcd = $i;
}

echo "GCD dari $p dan $q adalah: $x";
//hasil = GCD dari 3 dan 26 adalah 1
?>
```

GCD dari 3 dan 26 adalah: 1

Contoh 3. PHP GCD menggunakan rekursi

```php
<?php
function gcd($x, $y) {
  if ($y == 0)
    return $x;
  return gcd($y, $x%$y);
}

$x = 3;
$y = 26;

echo "GCD dari $p dan $q adalah: ".gcd($x,$y);
//hasil = GCD dari 3 dan 26 adalah 1
?>
```

GCD dari 3 dan 26 adalah: 1

- GCD in Python, menggunakan fungsi math.gcd(num1, num2). Contoh :

```
#Import math Library
import math

print (math.gcd(3, 26))
```

Hasil nya adalah 1

Nilai kunci B dapat berubah-ubah selama A tidak sama dengan 1 karena affine cipher termasuk algoritma kriptografi pergeseran. Nilai B harus ekuivalen dengan m (banyak karakter).

2. Enkripsi dan Dekripsi

   Fungsi Matematika:

   - **Fungsi Enkripsi**

     $E(x) = (ax + b) \bmod m$

     Keterangan :

     - Dimana x adalah indeks huruf dari panjang alfabet 0-25
     - m adalah panjang alfabet yang digunakan.
     - a dan b adalah kunci

   - **Fungsi Dekripsi**

     $D(y) = a^{-1} (y-b) \bmod m$

     Ketarangan :

     - Dimana y adalah nilai numerik dari teks yang dienkripsi
     - a^{-1} merupakan Modular Multiplicative Inverse (MMI) atau Inverse perkalian modular dari a mod m. Inverse perkalian modular hanya bisa dihitung jika a dan m merupakan bilangan koprima (Teorema Bezout). Berikut adalah rumus mencari nilai MMI nilai a:

     $a(n) \bmod m = 1$, dimana n adalah nilai MMI dari a mod m

     Contoh menghitung MMI :

     9 mod 26 = 9, sesuaikan dengan rumus MMI diatas, sehingga menjadi 9*(n) mod 26 = 1, maka

     9 * 3 mod 26 == 27 mod 26 = 1,

     Maka nilai MMI dari 9 mod 26 adalah 3

Berikut adalah Tabel Modular Multiplicative Inverse

A = 1	A' = 1
A = 3	A' = 9
A = 5	A' = 21
A = 7	A' = 15
A = 9	A' = 3
A = 11	A' = 19
A = 15	A' = 7
A = 17	A' = 23
A = 19	A' = 11
A = 21	A' = 5
A = 23	A' = 17
A = 25	A' = 25

3. Contoh

- **Fungsi Enkripsi**

  Pengirim dan penerima memutuskan sebuah kunci yang dipakai. Kita buat kunci yang dipakai adalah 3 dan 5. Kemudian pengirim ingin mengenkripsi pesan, yaitu cryptography'. Maka, akan diatur plaintext dan indeks alfabet sebagai berikut :

C	R	Y	P	T	O	G	R	A	P	H	Y
2	17	24	15	19	14	6	17	0	15	7	24

  Kemudian masukkan kunci dan indeks alfabet ke dalam rumus (fungsi) enkripsi. Diketahui a = 3, b = 5, maka :

  E(C) = (3*2 + 5) mod 26 = 11, maka hasil enkripsi C = L

  E(R) = (3*17 + 5) mod 26 = 4, maka hasil enkripsi R = E

  E(Y) = (3*24 + 5) mod 26 = 25, maka hasil enkrips Y = Z

  Dst...

Plaintext	C	R	Y	P	T	O	G	R	A	P	H	Y
x	2	17	24	15	19	14	6	17	0	15	7	24
(3x + 5)	11	56	77	50	62	47	23	56	5	50	26	77
(3x + 5) mod 26	11	4	25	24	10	21	23	4	5	24	0	25
Ciphertext	L	E	Z	Y	K	V	X	E	F	Y	A	Z

  Ciphertext hasil enkripsi kata cryptography adalah “LEZYKVXEFYAZ”

- **Fungsi Dekripsi**

  Untuk dekripsi, penerima menggunakan kunci yang sama dengan yang dipakai ketika proses enkripsi. Dekripsi menggunakan fungsi dekripsi diatas untuk mendapatkan plaintext.

Enkripsi	L	E	Z	Y	K	V	X	E	F	Y	A	Z
y	11	4	25	24	10	21	23	4	5	24	0	25
9(y-5)	54	-9	180	171	45	144	162	-9	0	171	-45	180
9(y-5) mod 26	2	17	24	15	19	14	6	17	0	15	7	24
Dekripsi	C	R	Y	P	T	O	G	R	A	P	H	Y

Langkah pertama adalah menghitung MMI.

a(n) mod 26 = 1, maka 3*n mod 26 = 1, maka 3*9 mod 26 = 1, hasil akhir 27 mod 26 = 1

Pada perhitungan diatas, nilai n adalah Modular Multiplication Inverse. Sehingga kita telah mengetahui bahwa nilai MMI adalah 9. Selanjutnya masuk ke perhitungan algoritma dekripsi affine cipher.

D(L) = 9(11-5) mod 26 = 2, maka hasil dekripsi L = C,

D(E) = 9(4-5) mod 26 = 17, maka hasil dekripsi E = R,

D(Z) = 9(25-5) mod 26 = 24, maka hasil dekripsi Z = Y,

Dst...

Hasil akhir dekripsi adalah “cryptography”

4. Keamanan

Affine cipher termasuk algortima substitusi monoalphabetic sehingga mewarisi kelemahan dari kelas cipher tersebut. Algortima caesar cipher bisa disebut juga dengan affine cipher yang memiliki nilai a = 1 karena fungsi enkripsi hanya mengurangi ke pergeseran linier.

Untuk mendekripsi ciphertext secara paksa, dapat digunakan metode brute force untuk melakukan test pada setiap kemungkinan nilai A dan B. Jika kita menggunakan 26 alfabet, maka hanya terdapat 12 kemungkinan nilai A (koprima A dan M) yang dapat digunakan dan 26 kemungkinan nilai B (alfabet yang digunakan). Sehingga hanya terdapat 312 tes yang perlu di lakukan untuk mendekripsi ciphertext menggunakan metode brute force.

Kelemahan utama cipher berasal dari fakta bahwa jika cryptanalyst dapat menemukan (melalui analisis frekuensi, brute force, menebak atau sebaliknya) plaintext dari dua karakter ciphertext maka kuncinya dapat diperoleh dengan memecahkan persamaan simultan. Karena kita tahu bahwa a dan m relatif prima, ini dapat digunakan untuk membuang banyak kunci yang "salah" dengan cepat dalam sistem otomatis.

C. Program

Berikut adalah contoh hasil program Affine chiper

```
Plaintext :  CRYPTOGRAPHY
kunci A :  3
kunci B :  5
Encrypted Text: LEZYKVXEFYAZ
Decrypted Text: CRYPTOGRAPHY
```

D. TUGAS INDIVIDU

1. Membuat fungsi enkripsi dan dekripsi teks menggunakan affine chiper (desain bebas, minimal dapat menampilkan plaintext dan ciphertext sesuai contoh di modul).
    - Bebas menggunakan bahasa pemrograman apa pun silahkan (Python, Java, PHP, dll).
    - Kata yang akan di enkripsi adalah nama masing-masing. Contoh : Yusuf Fadlila Rachman
    - Kunci yang digunakan bebas. Contoh: kunci A = 9, B = 17
    - Tambahkan dengan **PENJELASAN PROGRAM** yang anda buat (terutama fungsi enkripsi dan dekripsinya) langsung di samping kode. Penjelasan ditulis pakai comment saja boleh, atau di bagian readme Github.
2. Membuat enkripsi dan dekripsi teks di excel. Kata yang dienkripsi sesuai dengan nama lengkap masing-masing. Kunci sesuaikan dengan yang dipakai program. Contoh ada di file excel.
3. Pengumpulan Tugas Praktikum
    - **Upload file excel dan kode program di akun github masing-masing.** Kemudian kumpulkan link github di SPADA. Contoh : https://github.com/fadilrahman46/IoT
    - **Untuk kelas TI E** paling lambat tanggal **27 September 2021 jam 23.59**
      Untuk kelas TI D paling lambat tanggal **28 September 2021 jam 23.59**
    - Format penamaan file SKD_namakelas_nim_nama