

Aplikasi Teori Bilangan pada Kriptografi Kustomisasi

M Algah Fattah Illahi 13517122
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganessa 10 Bandung 40132, Indonesia
fattah_illahi@students.itb.ac.id

Abstrak—Pada era teknologi seperti sekarang, kita sudah tidak asing lagi dengan istilah kriptografi. Kini kita dapat mengirim informasi ke tempat yang begitu jauh jaraknya dengan begitu mudah. Namun kemudahan ini juga berlaku untuk pihak tidak bertanggung jawab yang ingin mencuri informasi yang anda miliki. Untuk memastikan informasi anda tidak jatuh ke tangan yang tidak diinginkan, kriptografi adalah salah satu solusinya. Sekarang ada begitu banyak macam algoritma kriptografi yang digunakan untuk melakukan pengamanan informasi. Tentunya kita dapat membuat algoritma kriptografi milik kita sendiri. Algoritma kriptografi kustomisasi sering kita temui pada kompetisi *cyber security—capture the flag*. Biasanya peserta akan diberikan kode sumber dan sebuah *flag* hasil enkripsi dan diminta untuk memecahkannya. Dengan begini peserta tidak bisa dengan mudah mencari metode dekripsi penyandian tersebut di internet dan harus memecahkannya sendiri. Pada makalah ini, penulis akan membahas aplikasi teori bilangan pada kriptografi kustomisasi dari sebuah soal *capture the flag*.

Kata Kunci—Algoritma, Dekripsi, Enkripsi, Kriptografi

I. PENDAHULUAN

Dewasa ini, perkembangan teknologi semakin pesat. Kita dapat mengirim pesan ke belahan dunia lain dalam beberapa milidetik saja. Bayangkan pada zaman dahulu orang harus menunggu berhari-hari bahkan bertahun-tahun untuk mendapatkan kabar dari orang terdekatnya, biaya pengirimannya pun tidak murah. Sebuah nikmat yang sepatutnya kita syukuri untuk dapat menikmati kemudahan ini.

Dibalik kemudahan tersebut, ternyata terdapat kecemasan yang menghantui pengirim dan penerima pesan, yaitu rasa cemas akan kerahasiaan pesan yang mereka kirim atau terima. Begitu banyak informasi yang lalu-lalang di internet, mulai dari informasi yang bersifat sepele yang disampaikan dua orang teman hingga informasi yang mengandung rahasia suatu negara atau organisasi.

Internet bisa diakses oleh siapa saja, sehingga pada dasarnya bisa saja seseorang melancarkan serangan *man in the middle* dan mencuri informasi penting milik anda. Disinilah peranan kriptografi untuk mengamankan informasi. Dengan kriptografi meskipun pesan tadi jatuh ke tangan yang salah, isi dari pesan tak akan bisa dibaca jika mereka tidak mampu mendekripsikan pesan tadi.

Saat ini ada berbagai macam kriptografi, mulai dari kriptografi sederhana hingga kriptografi dengan kompleksitas yang sangat tinggi.

Pada makalah ini saya akan membahas sebuah kriptografi

kustomisasi yang mengaplikasikan teori bilangan pada algoritmanya.

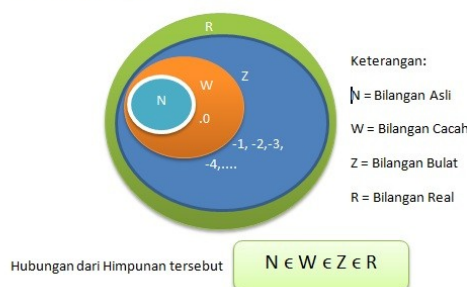
II. TEORI DASAR

A. Bilangan Bulat

Bilangan terbagi menjadi beberapa macam yaitu bilangan riil, bilangan rasional, bilangan bulat, bilangan cacah, dan bilangan asli. Klasifikasi bilangan tersebut dapat dilihat pada diagram venn berikut ini.

- Bilangan
- Jenis-jenis Bilangan
1. Bilangan Asli / Natural Number = {1, 2, 3, 4,}
 2. Bilangan Cacah = {0, 1, 2, 3, 4,}
 3. Bilangan Bulat / Integers Number = {-4, -3, -2, -1, 0, 1, 2, 3, 4,}
 4. Bilangan Real / Rasional Number = $\{\frac{p}{q}, \text{dimana } p, q \in \text{Bilangan Bulat, dan } q \neq 0\}$
Pecahan = $\{\frac{p}{q}, \text{dimana } p, q \in \text{Bilangan Bulat, dan } q \neq 0, q \text{ bukan kelipatan } p\}$

Sistem Bilangan dinyatakan dalam Diagram Venn



Gambar 1. Diagram Venn Klasifikasi Bilangan
Sumber: <https://mathsku.blogspot.com/2016/11/jenis-bilangan-matematika.html>

Bilangan bulat merupakan bilangan yang tidak memiliki pecahan desimal. Contoh bilangan bulat antara lain 1337, -20, 0, 900, -2. Sedangkan bilangan yang bukan bilangan bulat antara lain 0.6, 2.3, 100.1.

Bilangan bulat memiliki sifat pembagian yang dituliskan dengan notasi ' $a | b$ ' yang memiliki arti a habis membagi b atau b merupakan kelipatan dari a. definisi formalnya adalah sebagai berikut.

Definisi 1

Misalkan a dan b adalah dua bilangan bulat dimana $a \neq 0$. Kita menyatakan bahwa a habis membagi b jika terdapat bilangan bulat c yang memenuhi persamaan $a = bc$

Sebagai contoh,
 $3 \mid 12$, dengan mengambil $c = 4$, didapat $12 = 3 \times 4$, maka 3 habis membagi 12 atau 12 merupakan kelipatan dari 3.

Secara umum bilangan bulat jika dibagi dengan bilangan bulat lainnya akan menghasilkan suatu hasil bagi bilangan bulat dan sisa yang juga merupakan bilangan bulat. Sifat ini dituliskan secara formal dalam teorema berikut.

Teorema 1

Misalkan m dan n adalah dua buah bilangan bulat dimana $n > 0$. Jika m dibagi dengan n maka terdapat dua buah bilangan bulat unik q (hasil) dan r (sisa), sedemikian rupa sehingga

$$m = nq + r$$

dengan $0 \leq r < n$.

Teorema tersebut ditemukan oleh matematikawan Yunani, Euclid pada tahun 350 SM. Teorema itu bernama teorema Euclidean. Notasi lain untuk menuliskan hasil dan sisa adalah dengan operator *modulus* (mod) dan *division* (div). Contoh diatas juga dapat ditulis dalam notasi modulo seperti berikut ini.

$$q = m \text{ div } n$$

$$r = m \text{ mod } n$$

Sebagai contoh 30 dibagi dengan 4 akan memberikan hasil 7 dan sisa 2.

$$30 = 4 \cdot 7 + 2$$

$$30 \text{ mod } 4 = 2$$

$$30 \text{ div } 4 = 7$$

Contoh lain, -3 dibagi dengan 4.

ingatlah bahwa syarat $0 \leq r < n$ harus berlaku. Sehingga sisa dari pembagian harus selalu positif.

$$-3 = 4 \cdot (-1) + 1$$

$$-3 \text{ mod } 4 = 1$$

$$-3 \text{ div } 4 = -1$$

Operator *modulus* atau mod memiliki banyak kegunaan. Satu diantaranya adalah untuk memberikan batas atas dan bawah pada bilangan bulat. Misalkan $a \text{ mod } b = c$, maka berlaku $0 \leq c < b$.

dalam aritmetika modulo juga dikenal istilah kongruen. Dua bilangan bulat a dan b dikatakan kongruen dalam modulus m jika dan hanya jika a dan b memberikan sisa yang sama ketika dibagi dengan m . Ekspresi tersebut dapat dituliskan sebagai berikut.

$$a \equiv b \pmod{m}$$

Dalam aritmetika modulo terdapat beberapa sifat aljabar yang berlaku sehingga dapat memudahkan perhitungan aritmetika modulo. Sifat-sifat tersebut adalah

Teorema 2

1. Jika $a \equiv b \pmod{m}$ dan c adalah sembarang bilangan bulat, maka

- a) $(a + c) \equiv (b + c) \pmod{m}$
- b) $ac \equiv bc \pmod{m}$
- c) $ac \equiv bc \pmod{m}$, dengan syarat $c \geq 0$

2. Jika $a \equiv b \pmod{m}$ dan $c \equiv d \pmod{m}$, maka

- a) $(a + c) \equiv (b + d) \pmod{m}$
- b) $ac \equiv bd \pmod{m}$

B. Kriptografi

Kriptografi merupakan seni penyandian pesan sehingga pesan tersebut tidak dapat dipahami artinya selain oleh pengirim dan penerima pesan tersebut. Pada zaman sekarang kriptografi sangat penting mengingat banyaknya informasi yang bersifat rahasia yang dikirimkan lewat internet.

Pesan yang akan dikirim dan belum dienkripsi disebut sebagai *plaintext*. Sedangkan pesan yang sudah dienkripsi dan siap untuk dikirimkan disebut *ciphertext*. *Ciphertext* tidak bisa dipahami maknanya kecuali orang yang memiliki *ciphertext* tersebut tahu cara untuk melakukan dekripsi terhadap *ciphertext* tersebut.

Kriptografi pertama kali digunakan oleh tentara Yunani pada sekitar tahun 400 SM. Penyandian ini disebut *scytale*. Metode penyandian yang digunakan adalah dengan menuliskan *plaintext* pada pita yang dililitkan pada sebuah batang dengan diameter tertentu. Dengan melepaskan lilitan pita dari batang pesan yang tadi bisa dibaca dengan jelas akan menjadi susunan huruf acak pada seutas pita. Untuk mengembalikan pesan menjadi *plaintext*, pita harus dililitkan kembali pada batang yang memiliki diameter yang sama.

Ilmu dan seni untuk memecahkan *ciphertext* menjadi *plaintext* tanpa mengetahui kuncinya disebut kriptanalisis. Tujuan dari kriptanalisis adalah untuk menemukan kelemahan dan ketidakamanan dalam skema penyandian sehingga memungkinkan peningkatan dan perbaikan.

Kriptografi pada zaman sekarang ada banyak sekali macamnya. Kriptografi dapat dibagi menjadi 3 berdasarkan jenis kuncinya, yaitu.

1. Kriptografi Simetris

Merupakan jenis kriptografi yang menggunakan kunci yang sama dalam enkripsi dan dekripsi pesannya. Kunci kesuksesan dari kriptografi ini adalah kerahasiaan kuncinya.

Contoh kriptografi yang merupakan kriptografi simetris adalah *caesar cipher*, *hill cipher*, *vigenere cipher*, *substitution cipher*, *AES*, *RC4* dsb. Kriptografi tadi menggunakan kunci yang sama dalam enkripsi dan dekripsi pesannya sehingga kerahasiaan kunci harus dijaga.

2. Kriptografi Asimetris

Kriptografi asimetris merupakan sebuah teknik penyandian dengan kunci enkripsi dan dekripsi yang berbeda. Kunci yang digunakan dalam enkripsi disebut kunci public (*public key*) dan

kunci yang digunakan untuk dekripsi *ciphertext* disebut kunci privat (*private key*). Dengan begitu kunci publik tidak perlu dijaga kerahasiaannya, sedangkan kunci privat harus tetap dijaga kerahasiaannya. Keuntungan dari penggunaan kriptografi ini adalah memberi jaminan keamanan kepada siapapun yang melakukan pergantian informasi walaupun di antara mereka tidak pernah ada perjanjian tentang keamanan pesan terlebih dahulu atau tidak saling mengetahui satu sama lain.

Contoh kriptografi asimetris adalah penyandian RSA dan DSA. Pada penyandian RSA, seseorang harus memiliki dua bilangan yang relatif prima yang didapat dari pembangkit bilangan acak. Kemudian menghitung nilai kunci publik dan privat, lalu kunci publik dapat disebarkan sementara kunci privat tetap dijaga kerahasiaannya. Dengan begitu siapapun yang memiliki kunci publik dapat mengirimkan pesan secara terenkripsi kepada pemilik kunci privat, tetapi tidak bisa membaca pesan terenkripsi dari orang lain yang menggunakan kunci publik yang sama.

3. Kriptografi Hibrid

Permasalahan yang muncul dalam kriptografi adalah adanya *trade off* antara kecepatan dan kenyamanan. Semakin aman maka semakin tidak nyaman, begitu pula sebaliknya. Pada kriptografi simetris kecepatan enkripsi menjadi keunggulannya, namun ada kecemasan saat melakukan transfer kunci. Pada kriptografi asimetris, tidak ada keresahan ketika melakukan pengiriman kunci namun kecepatan pemrosesan yang relatif lambat mengurangi kenyamanan pengguna. Untuk menangani masalah tadi, kriptografi hibrid muncul. Dengan memanfaatkan kecepatan pemrosesan data oleh kriptografi simetris dan keringanan transfer kunci dari kriptografi asimetris, kriptografi hibrid meningkatkan kenyamanan tanpa mengurangi keamanan serta kenyamanan. Aplikasi kriptografi hibrida ini biasanya ditujukan pada pengguna komputer.

Kriptografi ini menggunakan *session key* yang dikirimkan kepada penerima dengan menggunakan kriptografi asimetris untuk mengamankannya terlebih dahulu. Kemudian setelah *session key* diterima dan didekripsi oleh penerima, pesan dikirim dengan menggunakan kriptografi simetris dengan *session key* yang tadi sudah dikirimkan. Dengan begitu penerima bisa mendekripsi pesan dalam waktu singkat dan keamanan *session key* tetap terjamin.

C. Pembangkit Bilangan Acak Semu

Pembangkit bilangan acak semu merupakan sebuah algoritma yang digunakan untuk membangkitkan bilangan yang bersifat acak atau tidak dapat ditebak. Hingga saat ini tidak ada komputasi yang bisa menghasilkan deret bilangan acak yang benar-benar sempurna. Bilangan acak yang dihasilkan hanyalah bilangan acak semu dari rumus-rumus matematika, karena pembangkitannya dapat diulang kembali.

Beberapa algoritma yang digunakan untuk membangkitkan bilangan acak semu adalah sebagai berikut.

1. Linear Congruential Generator (LCG)

merupakan pembangkit bilangan acak semu dengan bentuk

$$X_n = (aX_{n-1} + b) \bmod m$$

X_n = bilangan acak ke-n dari deretnya

a = faktor pengali

b = *increment*

m = *modulus*

Kunci pembangkit adalah X_0 yang disebut umpan (*seed*).

LCG mempunyai periode tidak lebih besar dari m , dan pada kebanyakan kasus periodenya kurang dari itu.

LCG memiliki periode penuh jika

1. b relatif prima terhadap m
2. $a - 1$ dapat dibagi dengan semua faktor prima dari m
3. $a - 1$ adalah kelipatan 4 jika m adalah kelipatan 4
4. $m > \max(a, b, x_0)$
5. $a > 0, b > 0$

Keunggulan LCG terdapat pada kecepatan prosesnya, karena hanya membutuhkan sedikit operasi bit. LCG tidak dapat digunakan untuk kriptografi karena urutan kemunculan bilangannya dapat diprediksi.

2. Blum Blum Shub

Merupakan algoritma pembangkit bilangan acak yang aman untuk kriptografi. Dibuat oleh Lenore Blum, Manuel Blum, dan Michael Shub pada tahun 1986.

Algoritma :

1. Pilih dua buah bilangan prima rahasia, p dan q yang masing-masing kongruen dengan $3 \pmod{4}$.
2. Kalikan keduanya menjadi $n = pq$. Bilangan n disebut bilangan bulat Blum.
3. Pilih bilangan bulat acak lain, s , sebagai umpan sedemikian sehingga :
 - (i) $1 < s < n$
 - (ii) s dan n relatif prima
 kemudian hitung $x_0 = s^2 \bmod n$
4. Barisan bit acak dihasilkan dengan melakukan iterasi berikut sepanjang yang diinginkan:
 - (i) hitung $x_i = x_{i-1}^2 \bmod n$
 - (ii) z_i = bit LSB (*Least Significant Bit*) dari x_i
 barisan bit acak adalah z_1, z_2, z_3, \dots

III. ANALISIS KRIPTOGRAFI KUSTOMISASI

Kriptografi yang akan dianalisis pada makalah ini berasal

dari sebuah soal *capture the flag* yang diadakan oleh Komunitas Cyber IPB. Diberikan sebuah file yang berisi *ciphertext* dan *source code* penyandian yang digunakan dalam bahasa python.

Berikut *source code* lengkapnya.

```
import random
def func1(c):
    a=0
    b=1
    while(c):
        a=a+b
        b=a-b
        c-=1
    return a
def func2(f):
    a=int(len(f)/2)
    b=random.randint(int(a/2),a)
    c=random.randint(int(a/2),a)
    d='mz'*b
    e='mz'*c
    return d+f+e
def enc(v1):
    v2=list(func2(v1))
    v4=[ ]
    v3=1
    while(len(v2)!=0):
        if func1(v3)> len(v2):
            v3=1
        else:
            ind=func1(v3)-1
            v4.append(v2[ind])
            del v2[ind]
            v3+=1
    return v4
print('='*58,'V.2.0')
plain=input('INPUT : ')
if len(plain) > 128:
    exit()
encoded = ".join(enc(plain))
print('ENCRYPTED:',encoded)
print('='*64)
```

dan berikut *ciphertext* yang diberikan

“mzzmmmmzzzUmmmzmmzzzmibmmzzmzmzmz4_mzzmz
 zmmzz_0zmmzzzmmcmmmmmzmmrrAzmmzmzzayIizmzm
 zgzAr5zmmzan_ummmhksndammc_daIm5mm{Akamon
 _malU_kgczm_Alnflzzme4_bjmzzfla}mmznAzmmmmzzzmm
 zzzmzzz”

Pada potongan *source code* terdapat penggunaan pembangkit bilangan acak.

```
import random
.
.
.
def func2(f):
    a=int(len(f)/2)
    b=random.randint(int(a/2),a)
    c=random.randint(int(a/2),a)
    d='mz'*b
    e='mz'*c
    return d+f+e
.
.
```

Pada potongan kode diatas, f merupakan *plaintext*. Sehingga jelas bahwa fungsi func2 adalah fungsi yang mengembalikan sebuah *string* yang merupakan *plaintext* ditambah dengan ‘mz’ sebagai awalan dan imbuhan dengan jumlah pengulangan acak dalam *range* panjang *plaintext* dibagi 2 hingga panjang *plaintext*.

Berikutnya kita lihat fungsi func1.

```
.
.
.
def func1(c):
    a=0
    b=1
    while(c):
        a=a+b
        b=a-b
        c-=1
    return a
.
.
.
```

func1 mengembalikan nilai ke c dari barisan fibonacci.

Sekarang kita akan masuk pada bagian utama algoritma penyandian ini.

```
def enc(v1):
    v2=list(func2(v1))
    v4=[ ]
    v3=1
    while(len(v2)!=0):
        if func1(v3)> len(v2):
            v3=1
        else:
            ind=func1(v3)-1
            v4.append(v2[ind])
            del v2[ind]
            v3+=1
    return v4
```

v2 berisi daftar angka yang merupakan indeks relatif karakter pada *plaintext* pada posisi tersebut. Sehingga jelas bahwa v2 adalah kunci dari kriptografi ini. Kriptografi ini pada dasarnya hanya mengacak-acak huruf pada *plaintext* dengan aturan tertentu sehingga bisa dipulihkan menjadi *plaintext* kembali dari *ciphertext*.

Berikut adalah *script* python yang digunakan untuk memecahkan penyandian tadi.

```
def func1(c):#fibonaci
    a=0
    b=1
    while(c):
        a=a+b
        b=a-b
        c-=1
    return a

#inisiasi
raw = open('reborn.enc').read()[:-1]
salin = list(raw)
key = []
v3 = 1

#generate key list
while(len(salin) != 0):
    if(func1(v3) > len(salin)):
        v3 =1
    else:
        ind = func1(v3)-1
        key.append(ind)
        del salin[ind]
        v3 += 1

#penyusunan kembali
result = ['X' for i in range(len(raw))]
#print key

for i in range(len(raw)):
    j = key[i]
    k = 0

    while((j > 0) or(result[k] != 'X')):
        #print 'j = ',j
        if(result[k] == 'X'):
            j -= 1

        k += 1

    result[k] = raw[i]

flag = ''.join(result)
flag = flag.strip('mz')
print flag
```

Kode diatas mengembalikan *plaintext* dengan cara membangkitkan kunci enkripsi/dekripsi dengan menggunakan metode yang sama pada kode untuk enkripsi. Kemudian menyusun karakter-karakter pada *ciphertext* dengan menaruhnya kembali pada posisi aslinya yang dapat diketahui dari kunci yang sudah dibangkitkan.

Plaintext yang didapat setelah *script* tadi dijalankan

```
“agrihack{mzny4_sAdar_gA_kalaU_InI_cUmA_bermodAl
k4n_fung5i_f1b0nacI_5ajA}”
```

Jika algoritma dari kriptografi ini dirahasiakan dari pihak selain pengirim dan penerima pesan. Maka salah satu jalan untuk melakukan dekripsi terhadap *ciphertext* adalah dengan mencoba semua kemungkinan susunan dari karakter yang ada pada *ciphertext* hingga didapat pesan yang bisa dimengerti.

Dengan metode ini, jika terdapat *plaintext* dengan panjang n maka akan terdapat nⁿ kemungkinan yang harus dicoba dalam usaha mendapatkan *plaintext*.

Kekuatan dari kriptografi ini adalah kerahasiaan dari algoritmanya. Meski begitu tidak banyak peserta yang berhasil memecahkan kriptografi ini, meski algoritmanya sudah dibocorkan.

IV. KESIMPULAN

Teori bilangan memiliki peranan yang amat banyak dalam ilmu komputer, salah satunya di bidang kriptografi. Kriptografi pada saat ini merupakan hal yang sangat penting mengingat banyaknya informasi penting yang bersifat rahasia yang dikirimkan lewat internet. Siapapun dapat terhubung dengan internet dan mencuri informasi tersebut, jika hal ini terjadi disanalah kriptografi memainkan peranannya.

Kriptografi kustomisasi yang dibahas pada makalah ini memanfaatkan teori bilangan dalam aplikasinya. Kriptografi ini memang tidak digunakan dalam pengiriman pesan di dunia nyata, tetapi algoritmanya cukup sulit untuk dipecahkan dan merupakan sarana yang baik untuk melatih *skill problem solving*.

V. UCAPAN TERIMA KASIH

Pertama-tama, saya ucapkan syukur kepada Allah SWT karena berkat rahmat dan karunia nya, saya diberi kesehatan dan kemampuan untuk menyelesaikan makalah ini dengan baik. Kemudian saya juga mengucapkan terima kasih sebesar-besarnya kepada tim dosen pengampu mata kuliah IF 2120 Matematika diskrit yaitu kepada Dra. Harlili S., M.Sc., Dr. Ir.Rinaldi Munir, M.T., dan Dr. Judhi Santoso, M.Sc. yang telah membimbing dan menyampaikan materi terkait Matematika Diskrit. Semoga makalah ini dapat memberikan manfaat sebesar-besarnya kepada para pembaca.

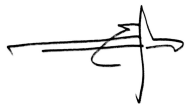
REFERENCES

- [1] <https://www.mastekno.com/id/pengertian-tujuan-dan-jenis-jenis-kriptografi-rumus-penyelesaian/> Diakses tanggal 9 Desember 2018.
- [2] [http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2017-2018/Pembangkit_Bilangan_Acak_\(2018\).pdf](http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2017-2018/Pembangkit_Bilangan_Acak_(2018).pdf) Diakses tanggal 9 Desember 2018

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 8 Desember 2018



M Algah Fattah Illahi 13517122