

Analisis Penerapan Aljabar Boolean Dalam Kriptografi Pada Masa Perang Dunia II

Bryan Bernigen - 13520034¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13520034@std.stei.itb.ac.id

Abstrak—Komunikasi merupakan kunci dari memenangkan perang. Begitu pula dengan informasi yang memiliki perang penting dalam memenangkan perang. Sehingga untuk dapat mengirimkan pesan kepada sekutu tanpa takut memberikan informasi kepada musuh jika pesan tersebut di sadap, diperlukan yang namanya kriptografi. Kriptografi sudah digunakan sejak jaman dahulu kala dalam perang, namun dalam perang dunia II diperlukan suatu kriptografi yang dapat dikirimkan melalui sinyal radio. Sinyal radio yang dikirimkan melalui transmiter sederhana hanya dapat bernilai 1 atau 0 sehingga cocok dengan sifat aljabar boolean yang hanya berfokus pada nilai 1 atau 0 saja. Kriptografi dengan aljabar boolean memiliki tingkat keamanan yang cukup tinggi jika kunci dari sebuah pesan di-generate secara random. Tingkat kekompleksannya dapat mencapai $2^x * n$ dengan x banyaknya bit per karakter dan n panjang pesan. Namun selama perang dunia II, kunci yang di-generate tidak sepenuhnya random sehingga tingkat kekompleksannya menjadi konstan yakni banyaknya cara menyusun mesin peng-generate kunci tersebut. Walaupun konstan, namun jika mesin tersebut cukup kompleks dan dalam pengkodeannya menggunakan tambahan keamanan seperti *double encryption*, pengkodean bit sendiri, mengirim pesan dalam bahasa lain, mengirim pesan tanpa format, dan menggunakan mesin peng-generate kunci yang dapat diatur-atur jumlah mata gir dan urutannya, maka kriptografi tersebut bisa dibilang “tidak terpecahkan”.

Kata Kunci—Aljabar Boolean, Kriptografi, Lorenz Cipher, perang dunia II.

I. PENDAHULUAN

Aljabar boolean adalah salah satu cabang matematika yang berfokus hanya pada dua nilai yakni 1/0 atau ya/tidak. Aljabar boolean biasanya digunakan dalam rangkaian elektronika seperti IC karena sifatnya yang sesuai dengan sifat listrik yakni ada listrik (1) atau tidak ada listrik (0). Bukan hanya itu saja, aljabar boolean juga dapat digunakan untuk mengodekan alfabet yang biasa kita gunakan sehingga kita tidak memerlukan 26 tombol untuk menuliskan seluruh alfabet, melainkan 1 tombol saja. Salah satu contoh penerapan aljabar boolean dalam mengodekan alfabet adalah dalam ASCII *encoding* yang mengodekan alfabet menjadi 8 buah bit yang setiap bitnya bernilai antara 0 atau 1. *Encoding* alfabet menjadi sebuah aljabar boolean membuat pengiriman pesan melalui transmiter sinyal radio sederhana menjadi memungkinkan karena komunikasi melalui sinyal radio sederhana pada dasarnya semudah menyambungkan dan melepaskan koneksi rangkaian listrik

sehingga cocok dengan sifat aljabar boolean yang hanya bernilai 1 atau 0 saja.

Komunikasi menggunakan radio merupakan sebuah kebutuhan dalam perang dunia II karena garis peperangan saat itu merentang sepanjang puluhan ribuan kilometer sehingga pesan penting atau instruksi dari atasan sulit untuk dikirimkan menggunakan cara konvensional seperti surat. Komunikasi menggunakan radio menjadi pilihan utama karena kecepatan pesan yang dikirimkan menggunakan sinyal radio adalah 300.000 kilometer per detik, atau lebih dikenal dengan kecepatan cahaya. Dengan kecepatan tersebut, pesan penting mengenai keadaan musuh atau instruksi dari atasan dapat tiba dalam kurang dari satu detik. Oleh karena itu, hampir seluruh komunikasi jarak jauh yang dilakukan pada perang dunia II dilakukan menggunakan sinyal radio.

Komunikasi menggunakan sinyal radio memang memiliki kelebihan dalam hal kecepatan pengiriman pesan. Namun komunikasi dengan sinyal radio memiliki satu kelemahan yaitu sinyal radio mudah untuk di sadap. Hanya dengan mengetahui frekuensi radio saat pesan tersebut dikirim, seseorang dapat menyadap isi pesan tersebut. Oleh karena itu, kriptografi sangat dibutuhkan untuk pengiriman pesan melalui sinyal radio terutama pada jaman perang ketika informasi merupakan salah satu senjata.

Kriptografi dalam pengiriman pesan sudah dipakai dari jaman dahulu kala seperti *Caesar cipher* yang digunakan pada jaman Romawi kuno. Namun *Caesar cipher* tidak dapat digunakan pada zaman perang dunia II karena terlalu simple dan mudah untuk dipecahkan. Pada jaman perang dunia II, perang bukan hanya perang tank atau perang pesawat, namun ada juga perang informasi sehingga *cipher* pada jaman tersebut berkembang sangat pesat. Pada zaman tersebut diperlukan sebuah *cipher* yang sulit untuk dipecahkan oleh musuh, namun harus tetap mudah dan cepat untuk di *decrypt* oleh sekutu. Oleh karena itu ditemukanlah beberapa mesin *cypher* terkenal seperti mesin enigma. Mesin enigma merupakan salah satu mesin cypher paling terkenal yang digunakan oleh Nazi Jerman untuk mengodekan pesannya. Namun ternyata pemimpin Nazi Jerman sendiri tidak menggunakan mesin enigma untuk mengodekan pesannya, melainkan menggunakan mesin Lorenz.

Berbeda dengan mesin enigma prinsip kerjanya bergantung pada mekanisme kelistrikan untuk mengodekan pesannya, prinsip kerja mesin Lorenz berfokus pada teori XOR aljabar boolean. Sedangkan mekanisme yang ada pada mesin hanya

bertugas untuk meng-generate sebuah key untuk mengenkripsi dan mendekripsi pesan. Lalu pertanyaannya adalah seberapa aman enkripsi menggunakan prinsip XOR aljabar boolean? Dan apakah enkripsi tersebut bisa dipecahkan? Jika bisa mengapa? Untuk menjawab pertanyaan-pertanyaan tersebut, maka kita harus memahami beberapa istilah-istilah penting yang telah penulis jabarkan pada bab selanjutnya.

II. TEORI DASAR

1. Aljabar Boolean

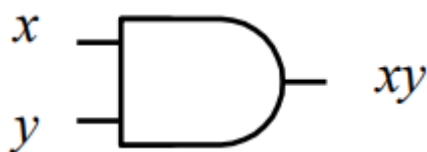
Aljabar Boolean adalah matematika yang digunakan untuk menganalisis dan menyederhanakan gerbang logika pada rangkaian-rangkaian digital elektronika. Pada dasarnya, aljabar boolean hanya bernilai benar dan salah atau dalam kelistrikan bernilai tinggi dan rendah. Biasanya aljabar boolean dilambangkan dengan 1 untuk tinggi atau benar dan 0 untuk rendah atau salah. Aljabar boolean sendiri ditemukan oleh matematikawan berkebangsaan Inggris pada 1854, yakni George Boole sehingga disebut boolean dari nama belakangnya Boole[1].

Terdapat beberapa hukum dasar aljabar boolean, yakni:

1. Hukum komutatif
2. Hukum Asosiatif
3. Hukum Distributif
4. Hukum AND
5. Hukum OR
6. Hukum Inversi

Dari keenam hukum tersebut, penulis hanya akan membahas Hukum AND, Hukum OR dan Hukum Inversi karena hukum lainnya merupakan hukum yang sudah sering digunakan pada matematika dasar.

Hukum AND merupakan hukum yang membandingkan dua buah masukan dan mengeluarkan nilai tinggi jika kedua masukan tinggi. Hukum AND direpresentasikan dengan gerbang logika AND.[2]



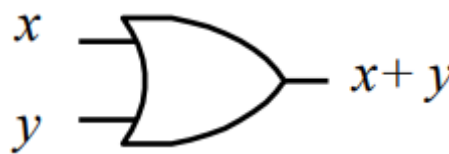
Gambar 1. Gerbang Logika AND

Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Aljabar-Boolean-\(2020\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Aljabar-Boolean-(2020)-bagian1.pdf)

Input		Output
A	B	
0	0	0
0	1	0
1	0	0
1	1	1

Tabel 1. Hasil Gerbang Logika AND

Hukum OR merupakan hukum yang menggabungkan dua buah masukan dan mengeluarkan nilai tinggi jika salah satu atau kedua masukan tinggi. Hukum OR direpresentasikan dengan gerbang logika OR.[2]



Gambar 2. Gerbang Logika OR

Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Aljabar-Boolean-\(2020\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Aljabar-Boolean-(2020)-bagian1.pdf)

Input		Output
A	B	
0	0	0
0	1	0
1	0	0
1	1	1

Tabel 2. Hasil Gerbang Logika OR

Hukum Inversi merupakan hukum yang membalik nilai masukan. Jika masukan tinggi maka keluaran akan rendah dan sebaliknya. Hukum inversi direpresentasikan dengan gerbang logika NOT.[2]



Gambar 3. Gerbang Logika NOT

Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Aljabar-Boolean-\(2020\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Aljabar-Boolean-(2020)-bagian1.pdf)

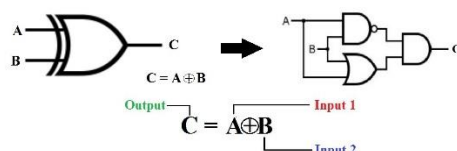
Input	Output
0	1
1	0

Tabel 3. Hasil Gerbang Logika NOT

Seperti yang sudah dijelaskan, terdapat representasi gerbang logika untuk setiap hukum-hukum dasar. Dari ketiga gerbang dasar tersebut, dapat diturunkan beberapa gerbang logika lainnya seperti gerbang logika NAND, gerbang logika NOR, gerbang logika XOR, dan gerbang logika XNOR. Gerbang yang akan penulis jabarkan lebih lanjut hanyalah gerbang XOR.

Gerbang XOR adalah gerbang logika yang membandingkan dua buah masukan dan akan mengeluarkan nilai tinggi jika hanya salah satu masukan yang tinggi. Gerbang XOR dapat dibuat dengan beberapa kombinasi gerbang AND, OR, dan NOT. Operasi XOR sendiri umumnya diberi simbol \oplus .

XOR Gate



Gambar 4. Gerbang Logika XOR

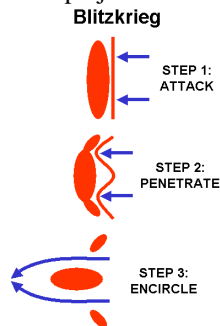
Sumber: <https://project123.com/2019/05/26/introduction-to-xor-gate/>

Input		Output
A	B	
0	0	0
0	1	1
1	0	1
1	1	0

Tabel 4. Hasil Gerbang Logika XOR

2. Komunikasi Menggunakan Sinyal Radio

Serangan Kilat atau yang lebih dikenal sebagai *Blitzkrieg* merupakan salah satu taktik perang Nazi Jerman yang paling terkenal karena keberhasilannya untuk menaklukkan Perancis hanya dalam kurun waktu 6 minggu saja. Kunci dari keberhasilan tersebut terletak pada mobilitas tank Jerman yang baik dan komunikasi antar personel yang baik. Komunikasi yang dimaksud di sini bukanlah komunikasi jarak dekat seperti mengobrol, melainkan komunikasi jarak jauh (>ratusan kilometer) karena taktik utama Nazi Jerman adalah mengirim dua pasukan tank terkuatnya jauh ke dalam wilayah musuh untuk mengepung sejumlah prajurit musuh.



Gambar 5. Cara Kerja *Blitzkrieg*

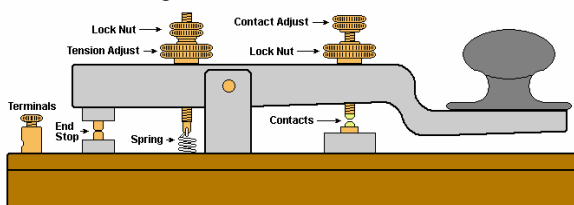
Sumber: <http://www.longwood.edu/staff/hardinds/blitzkrieg.html>

Taktik tersebut memiliki risiko yang sangat tinggi jika komunikasi antar divisi tidak cepat karena jika salah satu divisi (panah atas/bawah pada gambar 5) gagal untuk menembusi pertahanan musuh, maka divisi lainnya akan terkepung balik oleh musuh. Jadi diperlukan komunikasi yang baik untuk memastikan lokasi kedua divisi sudah tepat[3].

Untuk menyanggupi kebutuhan komunikasi tersebut, maka pada jaman perang dunia II, komunikasi radio merupakan standar komunikasi jarak jauh. Hampir setiap *platoon* (tim beranggotakan 16-40 orang) pada jaman itu memiliki sebuah radio untuk berkomunikasi dengan *platoon* lainnya. Komunikasi radio pada jaman masih menggunakan teknologi FM (*frequency modulation*). [4]

3. Transmitter Sederhana

Komunikasi menggunakan sinyal radio pada jaman itu cukup sederhana. Pengirim dan penerima hanya perlu menyamakan frekuensi radio saat pesan itu dikirim dan diterima. Lalu pengirim hanya perlu mengirim pesan melalui transmitter sederhana. Berikut gambar sebuah transmitter sederhana



Gambar 6. Transmitter Sederhana

Cara kerja transmitter sederhana adalah ketika tombol ditekan, maka bagian *contacts* akan terhubung dengan listrik sehingga menghasilkan suara. Jika tombol tidak ditekan, maka tidak akan muncul suara sama sekali. Oleh karena itu, transmitter hanya dapat digunakan untuk mengirim pesan dalam bahasa yang

sudah dikodekan menjadi sebuah aljabar boolean karena hanya ada dua keadaan yang bisa dikirim oleh transmiter, yakni ada dan tidak ada suara.[5] Contoh-contoh bahasa yang bisa dikirim melalui transmiter sederhana adalah sandi morse, ASCII *encoding*, dan lain-lain.

4. Kriptografi dalam perang dunia II

Kriptografi adalah ilmu yang mempelajari mengenai cara mengamankan pesan dari sadapan menggunakan kode rahasia dan juga ilmu yang mempelajari cara memecahkan kode rahasia untuk mendapatkan pesan asli. Kriptografi memiliki peran penting dalam perang dunia II. Dengan kriptografi, perang dunia II dapat dipersingkat dengan cukup banyak.[6]

Selama perang dunia II, setiap negara memiliki metode kriptografi mereka masing-masing. Namun salah satu metode kriptografi yang paling terkenal adalah kriptografi Nazi Jerman yang menggunakan mesin enigma. Prinsip dasar kriptografi dengan mesin enigma adalah rangkaian kelistrikan. Rangkaian kelistrikan kompleks tersebutlah yang menyebabkan pesan dapat dikodekan. Enigma sendiri telah berhasil dipecahkan oleh Alan Turing. Kisah pemecahan enigma tersebut cukup terkenal sampai diangkat ke layar kaca dengan judul “The Immitation Game”. Namun tahukah kalian bahwa pemimpin tertinggi Nazi Jerman, Adolf Hitler tidak menggunakan mesin enigma untuk mengodekan pesannya. Aldof Hitler sendiri menggunakan mesin lain yang bernama mesin lorenz untuk mengodekan pesannya. Mesin Lorenz tidak mengandalkan kelistrikan untuk mengodekan pesannya melainkan hanya sesimpel aljabar boolean.

5. Lorenz Cipher

Mesin Lorenz adalah mesin yang digunakan petinggi Nazi Jerman untuk mengodekan pesannya pada masa perang dunia II. Mesin Lorenz dinilai sangat aman karena terdapat kurang lebih 1.0×10^{170} kemungkinan kombinasi dari mesin tersebut.[7]

Walaupun mesin tersebut terdengar kompleks karenadapat menghasilkan 1.0×10^{170} kombinasi, namun prinsip kerjanya cukup sederhana yakni menggunakan prinsip aljabar boolean. Cara kerjanya adalah pertama-tama sebuah pesan dalam alfanumerik yang akan dikirim dikodekan ke dalam kode Baudot. Kode Baudot sendiri adalah pengodean 26 alfabet, angka, dan beberapa simbol lainnya menjadi rangkaian 5 bit untuk sebuah karakter. Berikut merupakan kode baudot yang digunakan Nazi Jerman selama perang dunia II.

LETTERS	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	FIGURES	0	1	2	3	4	5	6	7	8	9	SPACES	SPACE	STOP			
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Gambar 7. Kode Baudot

Sumber: <https://www.codesandciphers.org.uk/lorenz/fish.htm>

Kode baudot sendiri bukanlah kode eksklusif Nazi Jerman. Kode baudot merupakan kode teleprinter internasional yang digunakan secara global.[9] Setelah pesan tersebut dikodekan dalam kode baudot, maka pesan tersebut akan di XOR kan dengan sebuah kunci yang dihasilkan dengan sebuah kunci yang di-generate oleh mesin Lorenz. Berikut adalah contoh penggunaan kode tersebut jika titik hitam dianggap sebagai 1

sedangkan kosong dianggap 0

Pesan		Kunci		Hasil
M	00111	K	11110	11001
A	11000	E	10000	01000
T	00001	R	01010	01011
D	10010	E	10000	00010
I	01100	N	10011	11111
S	10100	S	10100	00000

Tabel 5. Contoh Aplikasi Kriptografi Lorenz

Jika hasil dari pesan tersebut kita XOR kan kembali dengan kuncinya, maka akan diperoleh

Hasil	Kunci		Pesan	
00111	K	11110	M	11001
11000	E	10000	A	01000
00001	R	01010	T	01011
10010	E	10000	D	00010
01100	N	10011	I	11111
10100	S	10100	S	00000

Tabel 5. Contoh Aplikasi Kriptografi Lorenz

Dengan menggunakan kunci yang sama antara pengirim dan penerima, dapat diperoleh pesan awal yang dikirim oleh pengirim. Jika kita lihat, maka prinsip tersebut cukup sederhana untuk diterapkan. Lantas dari mana angka 1.0×10^{170} tersebut muncul? Jawabannya adalah dari proses peng-generate-an kunci menggunakan mesin Lorenz.

Mesin Lorenz adalah mesin yang digunakan oleh petinggi Nazi Jerman untuk meng-generate kunci untuk mengodekan pesan dalam *lorenz cipher*. Mesin Lorenz memiliki 12 gir dan 501 pin yang dapat diatur sedemikian rupa.[8] Setiap gir memiliki lebih dari 30 posisi mulai sehingga hasil kombinasi kemungkinan posisi mulai yang benar adalah $1,6 \times 10^{19}$ macam kemungkinan. Lalu terdapat 501 pin yang dapat diatur nyala matinya sehingga terdapat 2^{501} macam kemungkinan atau sekitar 6.5×10^{150} macam kemungkinan. Sehingga total kemungkinan mendapat posisi mulai ke-12 gir yang tepat dan 501 konfigurasi pin yang tepat adalah sekitar 1 banding 1.0×10^{170} . Oleh karena itu, petinggi Nazi Jerman merasa yakin bahwa kriptografi mereka aman dan tidak akan pernah bisa terpecahkan.[7]

III. ANALISIS PENERAPAN ALJABAR BOOLEAN DALAM KRIPTOGRAFI

A. Aljabar Boolean dan Kriptografi

Seperti yang sudah dibahas pada bagian teori dasar *lorenz cipher*, letak keamanan kriptografi dengan *lorenz cipher* berada pada kekompleksan mesin *lorenz* dalam meng-generate kunci untuk mengodekan sebuah pesan. Namun disini penulis berusaha untuk mencari tahu seberapa aman sebuah kriptografi dengan prinsip XOR. Oleh karena itu, penulis mencoba untuk membuat sebuah program sederhana dalam bahasa Python untuk mempermudah perhitungan. Berikut merupakan kode yang digunakan penulis untuk mendapatkan data-data yang dibutuhkan.

```
'''
Kode baudot merupakan kode dengan panjang 5
bit sehingga
```

```
Penulis gantikan dengan integer 0-31 yakni
00000 - 11111
'''
# inisialisasi
pesan = [0 for i in range(32)]
kunci = 0
hasil = [0 for i in range(32)]

# membuat pesan menjadi 0-31
for i in range(31):
    pesan[i+1] += i+1

# melakukan XOR dengan key 0-31
for i in range(32):
    for j in range(32):
        hasil[pesan[j] ^ kunci] += 1
    kunci += 1

print("hasil:")
for i in range(4):
    for j in range(8):
        print(hasil[j+8*i], end=" ")
    print("")
```

Kode tersebut berusaha untuk mengodekan seluruh kode baudot dan meng-XOR-kannya dengan segala kemungkinan. Hasil dari kode tersebut adalah sebagai berikut.

```
hasil:
32 32 32 32 32 32 32 32
32 32 32 32 32 32 32 32
32 32 32 32 32 32 32 32
32 32 32 32 32 32 32 32
```

Dapat dilihat bahwa distribusi hasil XOR sangat merata. Seluruh bagian mendapat bagian yang sama yakni 32 kemungkinan. Sehingga untuk suatu karakter 5 bit, terdapat 32 kemungkinan kombinasi antara pesan dan kunci yang memungkinkan untuk menghasilkan 5 bit tersebut. Sehingga jika kita mencoba untuk melakukan *bruteforce* terhadap kode tersebut, maka kecepatan algoritma kita adalah $O(2^n)$ dalam notasi big O. Algoritma tersebut sangat buruk karena merupakan algoritma eksponensial. Sebagai contoh sebuah pesan dengan panjang 10 karakter akan memiliki lebih dari 1.0×10^{15} kemungkinan. Jika kita menggunakan CPU tercepat pada 2021 dengan *clock speed* sekitar 8.5 GHz dan asumsi 1 buah komputasi per Hz, maka dibutuhkan waktu kurang lebih 132458 detik atau kurang lebih 36,8 jam. Jika pesan tersebut berisi lebih dari 100 kata, maka akan dibutuhkan waktu berabad-abad untuk menghasilkan seluruh kemungkinan yang ada.

B. Aljabar Boolean vs Mesin Lorenz dalam Kriptografi

Jika kita bandingkan antara kriptografi menggunakan aljabar boolean dengan kriptografi menggunakan mesin *lorenz*, kita akan mendapatkan kompleksitas yang berbeda. Kriptografi menggunakan aljabar boolean memiliki kompleksitas $O(2^n)$

dalam notasi O besar atau secara lengkapnya adalah 32^n dengan n adalah jumlah karakter pesan. Sedangkan kriptografi dengan mesin lorenz memiliki kompleksitas yang tetap yakni 1.0×10^{170} untuk berapa pun panjang karakter pesan. Hal tersebut terjadi karena kompleksitas yang dihitung dalam kriptografi menggunakan mesin lorenz adalah kompleksitas mekanisme pengaturan peng-*generate* kunci. Ketika sudah ditemukan pengaturan yang tepat, maka akan diperoleh kunci yang benar untuk karakter ke-1 sampai karakter ke-n pesan. Sedangkan dalam kriptografi dengan aljabar boolean murni, kunci dianggap di-*generate* secara acak sehingga didapat 32 kemungkinan untuk setiap karakter dalam pesan.

Lalu mana di antara keduanya yang lebih aman? Jika hanya melihat banyaknya kemungkinan saja, maka untuk pesan dengan panjang karakter kurang dari 113, akan lebih aman untuk menggunakan mesin lorenz sedangkan untuk pesan dengan panjang lebih besar sama dengan 113, akan lebih aman menggunakan aljabar boolean saja. Namun pada praktiknya, akan jauh lebih aman untuk menggunakan aljabar boolean untuk panjang berapapun karena faktor ke-random-an kunci yang dihasilkan. Faktor ke-*random*-an memiliki peran penting dalam kriptografi karena sebuah pola dalam kriptografi akan berakibat fatal dalam keamanan kode tersebut. Penulis akan mengambil mesin lorenz dalam menjelaskan seberapa penting ke-*random*-an dalam kriptografi sekaligus seberapa bahaya pola dalam keamanan kriptografi.

C. Mesin Lorenz dan Kelemahannya

Seperti yang sudah dijabarkan pada penjelasan-penjelasan sebelumnya, mesin lorenz memiliki kurang lebih 1.0×10^{170} macam kombinasi yang memungkinkan. Angka tersebut sebenarnya sangat besar sehingga siapa pun bisa dengan tenang mengatakan bahwa mesin lorenz tidak mungkin bisa dipecahkan terutama oleh orang yang belum pernah melihat mesin tersebut. Pernyataan tersebut sebenarnya tidak sepenuhnya salah, namun tidak sepenuhnya benar juga karena selama perang dunia II, *lorenz cipher* telah berhasil dipecahkan oleh inteligen Inggris. Kode tersebut berhasil dipecahkan oleh John Tilman bersama timnya yang bahkan belum pernah melihat mesin lorenz.[7] Lantas bagaimana Tilman bersama timnya berhasil memecahkan sandi tersebut tanpa melihat mesinnya secara langsung? Jawabannya adalah karena kesalahan seorang tentara Jerman yang bertugas untuk mengirimkan pesan. Tentara tersebut mengirim sebuah pesan kepada rekannya. Namun rekannya tersebut gagal mendapat pesan tersebut sehingga rekannya meminta tentara tersebut untuk mengirimkan ulang pesan tersebut. Pada saat itulah tentara tersebut melakukan kesalahan fatal yakni karena ia mengirim ulang pesan tersebut menggunakan konfigurasi awal yang sama persis dan ia menyingkat beberapa kata dari pesan awalnya. Sebenarnya jika ia hanya melakukan salah satu dari kesalahan tersebut, kriptografi dengan mesin lorenz tidak akan terpecahkan. Namun karena kedua kesalahan tersebut, Tilman dan timnya dapat menduga-duga bentuk dan cara kerja kriptografi lorenz dengan sempurna. Tilman dan timnya berhasil menemukan jumlah gir, banyaknya mata setiap gir, dan seluruh konfigurasi lainnya.[7]

Dengan mengetahui banyaknya mata dalam setiap gir, lokasi setiap gir, dan kapan gir tersebut berputar, kita dapat

memprediksi sebuah pola berdasarkan informasi tersebut. Hal tersebutlah yang membuat Alan Turing dapat menemukan lokasi awal setiap gir dengan metode turingery. Prinsip kerja metode tersebut cukup sederhana yakni membandingkan 3 buah kata bersebelahan. Jika 1 dibandingkan dengan 1 atau 0 dibandingkan dengan 0, maka hasilnya 0. Sengkan jika 1 dibandingkan dengan 0 atau 0 dibandingkan dengan 1, maka hasilnya 0[11]. Berikut contoh prinsip turingery

M	N	T	M+N	N+T
1	1	1	1	1
1	1	1	1	1
0	0	1	1	0
0	0	1	1	0
0	1	0	0	0

Tabel 6. Contoh Prinsip Turingery

Prinsip turingery inilah yang digunakan Alan Turing dalam mengodekan colossus, sebuah komputer yang dirancang khusus untuk keperluan pemecahan kriptografi. Dengan metode turingery, komputer colossus, dan pengetahuan akan banyaknya mata gir dalam setiap gir dan periode putaran gir, maka *lorenz cipher* yang pada dasarnya memiliki 1.0×10^{170} kemungkinan dapat terpecahkan oleh orang-orang yang bahkan belum pernah melihat mesin tersebut.

Faktor utama dalam pemecahan *lorenz cipher* adalah pola. Kesalahan seorang tentara dalam mengirimkan dua buah pesan yang memiliki isi yang hampir sama dengan konfigurasi yang sama membuat para *code breaker* memiliki dua buah kode yang memiliki pola yang sama pada awal pesan. Hal tersebut yang membuat mekanisme mesin lorenz dapat diketahui dan membuat sandi tersebut berhasil dipecahkan.

D. Kriptografi yang Aman

Seperti yang sudah dibahas sebelumnya, kelemahan mesin lorenz terdapat pada adanya pola pada saat peng-*generate*-an kunci. Pola tersebut dihasilkan karena peng-*generate*-an kunci pada mesin lorenz tidak dihasilkan secara acak melainkan menggunakan mekanisme gir dan pin. Hal tersebut membuat kunci yang dihasilkan tidak sepenuhnya *random*. Conothnya adalah gir pertama akan berputar sebanyak 41 kali sebelum memutar gir kedua sehingga pada kode lorenz, akan terdapat pengulangan pola sebanyak 41 kali. Kepastian dalam jumlah mata gir itulah yang membuat kunci yang dihasilkan mesin lorenz tidak *random*. Kunci yang dihasilkan oleh mesin lorenz adalah kunci yang *pseudo-random* [8] yaitu kunci yang terlihat *random*, namun sebenarnya dapat dihasilkan menggunakan suatu prosedur matematis. Dengan demikian jika persamaan matematisnya ditemukan, maka kunci tersebut sudah tidak lagi *random*.

Agar kriptografi menjadi aman, maka diperlukan kunci yang digenerate secara *random*. Namun selama perang dunia II belum ada metode untuk meng-*generate* kunci secara acak, sehingga kriptografi yang sangat aman belum dapat di temukan. Namun ada beberapa cara untuk meningkatkan keamanan kriptografi pesan selama perang dunia II, yakni:

1. Double encryption

Double encryption adalah salah satu cara untuk menambah keamanan pada enkripsi sebuah kriptografi karena terdapat dua

kode yang harus dipecahkan untuk memperoleh kode aslinya. Pada kasus Nazi Jerman pada perang dunia II, *double encryption* dapat dipeloreh dengan menggunakan dua mesin yakni mesin enigma dan mesin lorenz. Mesin enigma akan mengacak pesan menggunakan kelistrikan dan mesin lorenz akan mengacak kembali pesan menggunakan aljabar boolean sehingga untuk memecahkan sandi tersebut, diperlukan dua kali pembobolan. Namun kelemahan dari *double encryption* adalah waktu. Diperlukan waktu ekstra baik untuk mengkodekan pesan maupun membaca pesan.

2. Menggunakan pengkodean aljabar boolean sendiri

Salah satu kelemahan fatal dari kriptografi Nazi Jerman adalah penggunaan kode baudot yang pada dasarnya merupakan kode internasional. Sehingga walaupun pesan yang dikirim sudah dikodekan, orang yang berusaha untuk meng-*intercept* pesan tersebut tetap tahu karakter demi karakter yang dikirim. Namun jika Nazi Jerman mengkodekan alfabet dengan aljabar boolean mereka sendiri, maka sekutu tidak akan bisa membobol pesan rahasia mereka tanpa tahu bagaimana bahasa tersebut dikodekan menggunakan aljabar boolean. Ini merupakan salah satu cara terbaik untuk mengecoh pembobol apalagi jika pengkodean menggunakan aljabar boolean tersebut dibuat semirip mungkin dengan kode internasional seperti kode baudot namun dengan konfigurasi yang berbeda. Hal tersebut dapat membuat *misleading* kepada para pembobol sehingga pesan tersebut menjadi lebih aman.

3. Menggunakan bahasa lain

Salah satu kelemahan lain dalam enkripsi kode Nazi Jerman terletak pada penggunaan bahasa Jerman dalam kode mereka. Sebagai contoh, militer Amerika pada perang dunia II menggunakan bahasa suku-suku pedalaman dalam mengirimkan pesan berkode. Dengan demikian walaupun sistem enkripsi pesan tersebut terbongkar, musuh mungkin tidak tahu bahwa cara pembongkaran yang mereka lakukan sudah tepat karena hasil pembongkaran tersebut bukan dalam bahasa yang mereka kenal sehingga mereka menganggap bahwa cara pembongkaran tersebut tidak tepat. Cara ini merupakan salah satu cara yang paling efektif karena cara ini tidak mengorbankan waktu demi keamanan.

4. Pesan tidak berformat

Salah satu kesalahan lain Nazi Jerman dalam mengenkripsi pesan mereka adalah karena mereka menggunakan format dalam pengiriman pesan. Sebagai contoh, setiap hari pesan rahasia yang dikirimkan akan dibuka dengan tujuan pengiriman dan dilanjut dengan laporan cuaca. Hal tersebut membuat musuh dapat menebak-nebak konfigurasi enkripsi karena mereka tahu bahwa pada awal pesan pasti berisi tujuan dan laporan cuaca. Kelemahan inilah yang mendasari pemecahan enigma oleh Alan Turing selain fakta bahwa sebuah karakter yang dikodekan dengan enigma tidak mungkin menjadi karakter itu sendiri (contoh: a tidak mungkin jadi a dan b tidak mungkin jadi b). Kedua kelemahan tersebutlah yang membuat pesan yang dikodekan dengan enigma berhasil dibobol. Namun jika pesan tersebut tidak memiliki format, pesan tersebut tidak akan dapat dipecahkan dengan semudah itu. Salah satu cara menghancurkan format dalam laporan adalah menambah karakter atau kata atau kalimat tanpa makna selama pengiriman pesan. Namun kelemahan hal tersebut adalah bertambahnya

waktu untuk mengenkripsi dan mendekripsi pesan.

5. Lokasi dan ukuran gir yang dapat diubah-ubah

Salah satu cara untuk menambah keamanan pesan adalah mengubah cara kunci tersebut dibuat. Salah satu kelemahan mesin lorenz adalah ke-12 gir yang ada tidak dapat diubah baik urutan maupun ukurannya. Hal tersebut membuat pola yang dihasilkan akan selalu sama yakni sesuai ukuran gir dan periode gir berputar. Namun jika ukuran dan atau urutan gir dapat ditukar-tukar, maka kunci tersebut akan menjadi lebih *random*.

IV. SIMPULAN

Kriptografi menggunakan prinsip aljabar geometri merupakan salah satu kriptografi yang sederhana namun memiliki keamanan yang beragam tergantung dengan bagaimana kunci tersebut dibuat. Jika menggunakan kunci yang sepenuhnya *random*, maka kriptografi menggunakan aljabar geometri menjadi sangat aman karena memiliki kompleksitas $2^x * n$ dengan x banyaknya bit per karakter dan n panjang pesan. Namun jika kunci tersebut di-*generate* secara *pseudo-random* atau dengan kata lain tidak sepenuhnya *random*, maka kompleksitasnya akan menjadi tetap yakni jumlah cara menyusun mekanisme peng-*generate* kunci. Satu hal yang perlu diingat, jika kunci tersebut memiliki jumlah kombinasi yang sangat banyak, namun memiliki suatu pola tertentu secara berkala, maka pemecahannya tidak akan serumit suatu kunci dengan kombinasi yang jauh lebih sedikit namun 100% *random*. Jika kita memilih untuk meng-*generate* kunci secara *pseudo-random*, maka ada beberapa cara untuk meningkatkan keamanan pesan kita yakni menggunakan dua kali enkripsi, menggunakan metode pengodean alfabet dalam aljabar boolean kita sendiri, mengubah pesan menjadi suatu bahasa lain, mengirim pesan tanpa format yang baku, memperbanyak kombinasi cara mesin meng-*generate* kunci sehingga tidak terlihat pola sama sekali.

VII. UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena dengan rahmat dan berkat-Nya, penulis dapat menyelesaikan makalah ini dengan baik. Penulis ingin mengucapkan terima kasih kepada kedua orang tua penulis karena sudah mendukung penulis selama masa penyusunan makalah ini. Penulis juga ingin mengucapkan kepada Dr. Ir. Rinaldi Munir, M.T selaku dosen K1 mata kuliah IF2120 Matematika Diskrit 2021 yang sudah mengajari penulis mengenai dasar dari aljabar boolean dan kriptografi sehingga penulis memiliki ide mengenai makalah ini. Penulis juga ingin berterima kasih kepada algoritma referensi google dan youtube yang sudah memberikan referensi mengenai kriptografi selama perang dunia II pada halaman utama penulis sehingga penulis berkesempatan untuk mempelajari mengenai kriptografi selama perang dunia II dengan lebih mendalam.

REFERENSI

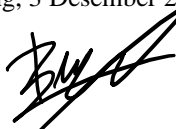
- [1] Dickson Kho, 2020, "Pengertian Aljabar Boolean dan Hukumnya", <https://teknikelektronika.com/pengertian-aljabar-boolean-hukum-aljabar-boolean/>, diakses 3 Desember 2021

- [2] Ulfa Faudiah, 2020, "Logika Digital Komputer", <https://medium.com/@ulfafaudiah99/logika-digital-komputer-21150ff77308>, diakses 3 Desember 2021
- [3] History.com Editors, 2019, "Blitzkrieg", <https://www.history.com/topics/world-war-ii/blitzkrieg#:~:text=Blitzkrieg%20is%20a%20term%20used,loss%20of%20soldiers%20and%20artillery.>, diakses 4 Desember 2021
- [4] George I. Back, "Military Communication", <https://www.britannica.com/technology/military-communication>, diakses 4 Desember 2021
- [5] Britannica Editors, "Morse Code", <https://www.britannica.com/topic/Morse-Code>, diakses 4 Desember 2021
- [6] Dr. Dough Lantry, "War of Secrets: Cryptology in WWII", <https://www.nationalmuseum.af.mil/Visit/Museum-Exhibits/Fact-Sheets/Display/Article/196193/war-of-secrets-cryptology-in-wwii/>, diakses 4 Desember 2021
- [7] Singingbanana, 2014, "Lorenz: Hitler's 'Unbreakable' Cipher Machine", <https://www.youtube.com/watch?v=GBsfWSQVtYA>, diakses 5 Desember 2021
- [8] Tony Sale, "The Lorenz Cipher and How Bletchley Park broke it", <https://www.codesandciphers.org.uk/lorenz/fish.htm>, diakses 5 Desember 2021
- [9] <https://cs.stanford.edu/people/eroberts/courses/soco/projects/2008-09/colossus/audot.html>, diakses 5 Desember 2021
- [10] <https://www.independent.co.uk/news/obituaries/captain-jerry-roberts-bletchley-park-codebreaker-who-helped-crack-the-tunny-code-hitler-used-to-communicate-with-generals-9219984.html>, diakses 5 Desember 2021
- [11] B Jack Copeland, "Colossus: Breaking the German 'Tunny' Code at Bletchley Park. An Illustrated History", <http://www.rutherfordjournal.org/article030109.html>, diakses 5 Desember 2021
- [12] W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

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, 3 Desember 2020



Bryan Bernigen (13520034)