Penerapan Teori Bilangan pada Keamanan Pengiriman Suara Melalui *Bluetooth* dengan Algoritma AES

Suthasoma Mahardhika Munthe - 13522098[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13522098@std.stei.itb.ac.id

***Abstract*—*Bluetooth* adalah teknologi yang menyediakan koneksi nirkabel yang praktis dan mudah digunakan. Teknologi ini sangat berpengaruh terhadap industri perangkat audio. Kehadiran *Bluetooth* menyebabkan industri perangkat audio cenderung berfokus pada pengembangan perangkat yang dapat terhubung melalui koneksi *Bluetooth*. Pengiriman data suara seperti seperti musik melalui koneksi ini dapat menjadi rentan terhadap pembajakan. Oleh karena itu, sistem keamanan dalam proses transfer data melalui *Bluetooth* menjadi aspek yang harus dipenuhi. Salah satu sistem keamanan yang disediakan Bluetooth adalah proses enkripsi data sebelum dikirim melalui media gelombang radio. Model enkripsi yang sering digunakan adalah algoritma AES (*Advanced Encryption Standard*).**

***Keywords*—Bluetooth, Keamanan, Enkripsi, Advanced Encryption Standard.**

I. PENDAHULUAN

Bluetooth merupakan salah satu teknologi nirkabel yang memanfaatkan gelombang radio sebagai media pengiriman data. Teknologi ini menjadi sangat populer dan luas digunakan dalam berbagai perangkat, mulai dari *smartphone* hingga perangkat audio dan periferal komputer. *Bluetooth* memiliki kelebihan utama, yaitu kemampuannya untuk membuat koneksi nirkabel yang praktis dan mudah digunakan.

Kemunculan teknologi ini menyebabkan perubahan dalam dunia musik. Pada awalnya, musik seringkali terkait dengan pengalaman duduk dan mendengar lewat perangkat audio yang terhubung dengan kabel. Perangkat audio seperti earphone, headset, dan speaker yang awalnya dihubungkan dengan kabel, telah digantikan dengan teknologi nirkabel ini. Dewasa ini, kehadiran *Bluetooth* telah membawa revolusi besar dalam cara berinteraksi dengan teknologi audio. Hal ini menyebabkan industri teknologi audio saat ini cenderung berfokus pada pengembangan perangkat yang dapat terhubung melalui *Bluetooth*.

Namun, seiring dengan perkembangan teknologi, aspek keamanan menjadi hal yang krusial sehingga diperlukan sistem keamanan dalam proses pengiriman data lewat *Bluetooth*. Pengiriman data yang menggunakan media gelombang radio rentan terhadap kemungkinan pembajakan data. Perangkat *Bluetooth* dapat menjadi rentan terhadap serangan seperti *spoofing* (pemalsuan identitas), *sniffing* (pencurian data), dan serangan berbasis keamanan koneksi. Oleh karena itu, implementasi sistem keamanan yang baik sangat diperlukan untuk melindungi informasi yang dikirim dan diterima melalui *Bluetooth*.

Sistem keamanan yang umum adalah kriptografi. Dalam implementasinya banyak sekali algoritma enkripsi telah tersedia untuk menjaga keamanan data saat dikirim melalui saluran komunikasi. Algoritma enkripsi yang digunakan disesuaikan dengan kebutuhan sehingga proses pengiriman data tetap efektif dan efisien. Contoh algoritma enkripsi saat ini antara lain, AES (*Advanced Encryption Standard*), RSA (Rivest-Shamir-Adleman), DES (*Data Encryption Standard*), 3DES (Triple DES), Blowfish, Twofish, ChaCha20, Camellia, dan lain-lain. Algoritma enkripsi ini dapat berubah seiring waktu dengan adanya perkembangan teknologi. Perkembangan ini dapat menyebabkan sebuah algoritma dapat dipecahkan atau dikembangankan menjadi lebih aman.

AES (*Advanced Encryption Standard*) adalah salah satu algoritma enkripsi simetris. Model enkripsi ini memiliki kunci enkripsi dan dekripsi yang sama. AES sering digunakan dalam sistem keamanan pengiriman data melalui *Bluetooth*. Algoritma ini memanfaat berbagai macam bidang keilmuan. Salah satunya adalah teori bilangan.

II. LANDASAN TEORI

A. Teori Bilangan

Teori bilangan adalah cabang matematika murni yang ditujukan untuk mempelajari bilangan bulat (*integer*) atau fungsi bernilai bilangan bulat. Bilangan bulat (*integer*) adalah bilangan yang tidak mengandung pecahan desimal, misalnya 20, 3, 2004, -1, dan sebagainya[1].

Misalkan a dan b bilangan bulat dengan $a \neq 0$. a habis membagi b jika terdapat bilangan bulat c sedemikian sehingga $b = ac$. Sifat habis membagi ini dinotasikan dengan

$$a \mid b \text{ jika } b = ac,\ c \in \mathbb{Z} \text{ dan } a \neq 0.$$

Misalkan m dan n adalah bilangan bulat, n > 0. Jika m dibagi dengan n maka hasil pembagiannya adalah q (*quotient*) dan sisanya r (*remainder*), sedemikian sehingga

$$m = nq + r, 0 \leq r < n.$$

Pembagi bersama terbesar (PBB – greatest common divisor atau *gcd*) dari *a* dan *b* adalah bilangan bulat terbesar *d* sedemikian sehingga *d* | *a* dan *d* | *b*. Dalam hal ini dinyatakan sebagai PBB(*a*, *b*) = *d*.

Misalkan *m* dan *n* bilangan bulat, dengan syarat n > 0 sedemikian sehingga

$$m = nq + r, 0 \leq r < n$$

maka PBB(*m*, *n*) = PBB(*n*, *r)*.

Algoritma Euclidean dapat digunakan dalam menentukan PBB dari dua buah bilangan bulat. Misalkan *m* dan *n* adalah bilangan bulat tak negatif dengan $m \geq n$. Misalkan $r_0 = m$ dan $r_1 = n$. Lakukan secara berturut-turut pembagian untuk memperoleh

$$\begin{aligned} r_0 &= r_1q_1 + r_2 && 0 \leq r_2 < r_1, \\ r_1 &= r_2q_2 + r_3 && 0 \leq r_3 < r_2, \\ &\vdots \\ R_{n-2} &= r_{n-1}q_{n-1} + r_n && 0 \leq r_n < r_{n-1}, \\ R_{n-1} &= r_nq_n + 0 \end{aligned}$$

PBB(*m*, *n*) = PBB(r_0, r_i) = PBB(r_1, r_2) = … = PBB(r_{n-1}, r_n) = PBB(r_n, 0) = r_n.

Jadi, PBB dari *m* dan *n* adalah sisa terakhir yang tidak nol dari runtutan pembagian tersebut.

Dua buah bilangan bulat *a* dan *b* dikatan relatif prima jika PBB(*a*, *b*) = 1. Dari definisi tersebut dapat dibentuk sebuah kombinasi linier dari bilangan bulat m dan n sedemikian sehingga

$$ma + nb = 1$$

Misalkan *a* dan *m* bilangan bulat (*m* > 0). Operasi

$$a \bmod m$$

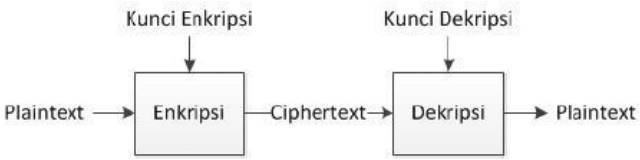
memberikan sisa jika *a* dibagi dengan *m*. Notasi *a* mod *m* = *r* sedemikian sehingga *a* = *mq* + *r*, dengan 0 ≤ *r* < *m*. *m* disebut sebagai modulus atau modulo, dan hasil aritmetika modulo m terletak dalam himpunan {0, 1, 2, 3, .., $\underline{m-1}$}.

B. Kriptografi

Kriptografi (*cryptography*) berasal dari bahasa Yunani, yaitu dari asal kata *cryptos* + *graphein*, di mana *cryptos* berarti rahasia dan *graphein* berarti gambar atau tulisan. Jadi, secara etimologi kriptografi berarti tulisan rahasia[2].

Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara mengubah pesan ke suatu bentuk yang tidak dapat dimengerti maknanya sehingga pesan tersebut menjadi aman dan tidak dapat dibaca orang lain yang tidak berkepentingan[3].

Pesan adalah data atau informasi yang dapat diketahui maknanya. Pesan ini dinamakan plainteks (*plaintext*). Pesan dapat berupa data atau informasi yang dikirim (salah satunya melalui saluran komunikasi) atau disimpan di dalam media perekaman (kertas, *storage*, dsb). Agar pesan tidak dapat diketahui maknanya oleh pihak lain, maka pesan disandikan ke bentuk lain[2]. Bentuk pesan yang disandikan dinamakan cipherteks (*chipertext*) atau kriptogram (*cryptogram*)[4]. Kriptografi memiliki peran dalam menyandikan sebuah plainteks ke dalam bentuk cipherteks maupun sebaliknya dengan sebuah kriptografer.

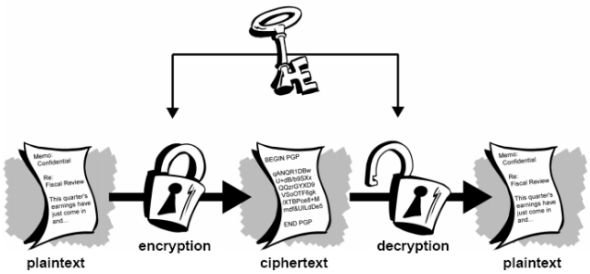


Gambar 1. Proses di dalam kriptografi.
Sumber: https://www.researchgate.net

Secara umum, proses di dalam kriptografi melibatkan proses enkripsi dan dekripsi. Proses enkripsi bertugas mengubah sebuah plainteks ke dalam bentuk sandinya, yaitu cipherteks. Sebaliknya proses dekripsi bertugas mengembalikan sebuah cipherteks ke dalam bentuk aslinya, yaitu plainteks[4].

Sistem kriptografi mencakup sebuah algoritma, semua plainteks yang mungkin, cipherteks, dan kunci-kunci[2]. Secara umum sistem kriptografi digolongkan menjadi dua buah, yaitu:

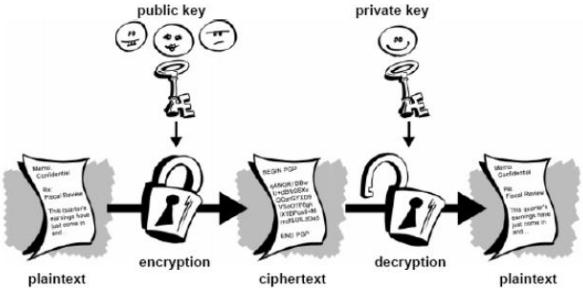
1. Sistem Kriptografi Simetri
   Dalam sistem ini, kunci yang sama digunakan untuk melakukan enkripsi dan dekripsi. Contoh dari sistem ini adalah RC4 (*ARCFOUR*), Data Encryption Standard (*DES*), *Blowfish*, *IDEA*, dan Advanced Encryption Standard (*AES*). Sistem kriptografi yang akan dibahas adalah Advanced Encryption Standard (*AES*). Kunci yang digunakan harus dirahasiakan karena para penyerang dapat mendekripsi cipherteks yang didapat. Panjang kunci yang digunakan juga menjadi sangat penting. Semakin panjang kunci yang digunakan maka penyerang akan semakin sulit untuk melakukan percobaan terhadap semua kombinasi (*brute-force attack*). Proses sistem kriptografi simetri dapat dilihat pada Gambar 2.



Gambar 2. Proses enkripsi dan dekripsi kriptografi simetri.
Sumber: [2]

2. Sistem Kriptografi Asimetri
   Ide dasar dari sistem kriptografi kunci publik adalah bahwa kunci kriptografi dibuat sepasang, satu kunci untuk enkripsi dan satu kunci untuk dekripsi. Kunci untuk enkripsi bersifat publik (*public key*) bersifat public (tidak rahasia), sedangkan kunci dekripsi bersifat rahasia sehingga dinamakan kunci rahasia (*private key* atau *secret key*). Kunci ini dipilih sehingga, secara praktik, tidak mungkin menurunkan kunci rahasia dari kunci publik[4]. Proses enkripsi dan dekripsi pada sistem

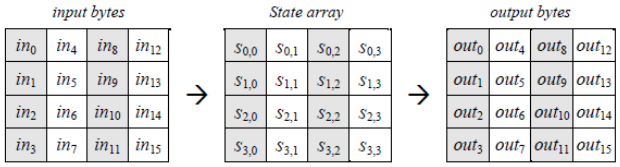
kriptografi ini dapat dilihat pada Gambar 3.



Gambar 3. Proses enkripsi dan dekripsi kriptografi kunci-publik
Sumber: [2]

C. *Advanced Encryption Standard (AES)*

Setiap data yang akan dienkripsi akan dibagi sesuai dengan panjang *Chiper Key* yang digunakan. *Chiper Key* yang digunakan pada algoritma *AES* adalah 128, 192, dan 256 bit[5]. Data yang akan dienkripsi akan dibagi menjadi blok-blok sepanjang *Cipher Key* yang dipilih. Setelah dibagi menjadi beberapa blok, setiap blok akan dibagi menjadi *array of byte* sedemikian sehingga tersusun menjadi sebuah matriks. Matriks terdiri dari 4 baris, setiap baris berisi ***Nb*** byte, dengan ***Nb*** adalah panjang blok dibagi 32. Setiap byte akan mengandung informasi posisinya pada matriks pada baris ke-*r* dan kolom ke-*c*, dengan $0 \le r < 4$ dan $0 \le c < \boldsymbol{Nb}$. Susunan matriks dapat dilihat melalui Gambar 4 berikut.



Gambar 4. Matriks input dan output
Sumber: [5]

Sistem penjumlahan dan perkalian yang digunakan dalam proses enkripsi dan dekripsi pada algoritma ini berbeda. Sistem penjumlahan dilambangkan dengan $\oplus$ dan perkalian dilambangkan dengan ●. Byte diinterpretasikan dalam bentuk polinomial. Sebagai contoh, {01100011} (dalam desimal 99) diidentifikasi dengan representasi $x^6 + x^5 + x + 1$.

Proses penjumlahan dengan notasi $\oplus$ dengan operasi XOR, atau dengan modulo 2, $1 \oplus 1 = 0$, $1 \oplus 0 = 1$, dan $0 \oplus 0 = 0$. Oleh karena itu, penjumlahan byte dengan byte lain dideskripsikan dengan operasi modulo 2 untuk setiap bit yang bersesuaian. Untuk dua byte, misalnya $\{a_1a_2a_3a_4a_5a_6a_7a_8\}$ dan $\{b_1b_2b_3b_4b_5b_6b_7b_8\}$, hasil penjumlahannya adalah $\{c_1c_2c_3c_4c_5c_6c_7c_8\}$, dengan $c_i = a_i \oplus b_i$, $1 \le i \le 8$.

Sebagai contoh, ekspresi di bawah ini ekuivalen satu dengan yang lain:

$(x^6 + x^4 + x^2 + x + 1) \oplus (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$
(notasi polinom);
{01010111} $\oplus$ {10000011} = {11010100}
(notasi biner);
{57} $\oplus$ {83} = {d4}
(notasi heksadesimal).

Sedangkan pada operasi perkalian dua buah byte dikalikan dan dimodulo dengan polinom tidak tereduksi derajat 8, yaitu

$$m(x) = x^8 + x^3 + x^4 + x + 1,$$

atau {01}{1b} dalam representasi heksadesimal

Sebagai contoh, {57} ● {83} = {c1}, karena

$(x^6 + x^4 + x^2 + x + 1)(x^7 + x + 1) = x^{13} + x^{11} + x^9 + x^8 + x^7 + x^7 + x^5 + x^3 + x^2 + x + x^6 + x^4 + x^2 + x + 1$

$= x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1$

dan

$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1$ modulo $(x^8 + x^3 + x^4 + x + 1) = x^7 + x^6 + 1$.

Reduksi dengan *m(x)* memastikan hasil perkalian dua buah byte selalu dalam bentuk polinom dengan derajat di bawah 8.

Seperti yang sudah dijelaskan sebelumnya, panjang *Chiper Key* yang digunakan adalah 128, 192, dan 256 bit. Panjang kunci direpresentasikan dengan ***Nk*** = 4, 6, atau 8 (panjang kunci dalam bit dibagi 32). Pada algoritma *AES* angka *round* yang digunakan saat eksekusi bergantung pada panjang kunci yang digunakan. Angka *round* dilambangkan dengan ***Nr*** dengan ***Nr*** = 10 saat ***Nk*** = 4, ***Nr*** = 12 saat ***Nk*** = 6, dan ***Nr*** = 14 saat ***Nk*** = 8. Kombinasi Key-Block-Round yang ditetapkan dalam algoritma ini dapat dilihat pada Gambar 5 berikut.

	Key Length *(Nk words)*	Block Size *(Nb words)*	Number of Rounds *(Nr)*
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

Gambar 5. Kombinasi Key-Block-Round.
Sumber: [5]

Proses enkripsi melibatkan prosedur *round* sebanyak ***Nr*** yang sesuai dengan panjang kunci yang dipilih, yaitu 10, 12, atau 14[5]. Setiap kali pengulangan akan dilakukan operasi `SubBytes()`, `ShiftRows()`, `MixColumns()`, dan `AddRoundKey()`. Algoritma dapat dilihat melalui Gambar 6.

```
Cipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
   byte  state[4,Nb]

   state = in

   AddRoundKey(state, w[0, Nb-1])

   for round = 1 step 1 to Nr-1
      SubBytes(state)
      ShiftRows(state)
      MixColumns(state)
      AddRoundKey(state, w[round*Nb, (round+1)*Nb-1])
   end for

   SubBytes(state)
   ShiftRows(state)
   AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1])

   out = state
end
```

Gambar 6. *Pseudocode* untuk proses enkripsi.
Sumber: [5]

Fungsi `SubBytes()` digunakan untuk melakukan transformasi dengan menggunakan sebuah tabel substitusi (S-box). S-box (Gambar 8), yang mana *invertible*, dibentuk dengan dua transformasi:

1. Hitung invers perkalian byte, elemen {00} dipetakan pada dirinya sendiri.

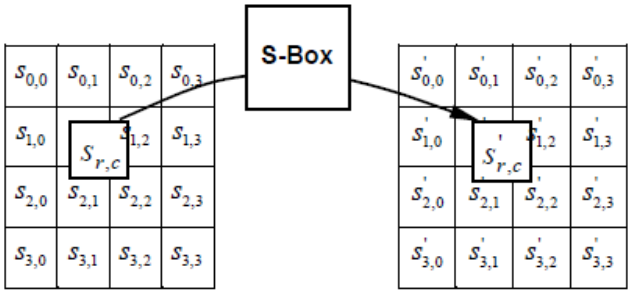
2. Gunakan transformasi affine dibawah ini:

$b_i = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i$

untuk $0 \leq i < 8$, dengan b_i adalah bit ke-i pada byte, dan c_i adalah bit ke-i dari byte c dengan nilai {63} atau {011000011}. Proses tersebut dapat diekspresikan menggunakan matriks di bawah ini.

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}.$$

Gambar 7 mengilustrasikan efek dari proses `SubBytes()` transformasi pada *state* matriks plainteks yang akan dienkripsi.



Gambar 7. `SubBytes()` melakukan transformasi untuk setiap byte pada *state* matriks.
Sumber: [5]

S-box yang digunakan pada proses transformasi fungsi `SubBytes()` ditunjukkan pada gambar 8. Sebagai contoh, jika $s_{1,1}$ = {53}, maka nilai byte yang akan disimpan pada $s'_{1,1}$ adalah nilai pada S-box baris ke-'5' dan kolom ke-'3', yaitu $s'_{1,1}$ = {ed}.

x \ y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Gambar 8. S-box: nilai substitusi (representasi heksadesimal)
Sumber: [5]

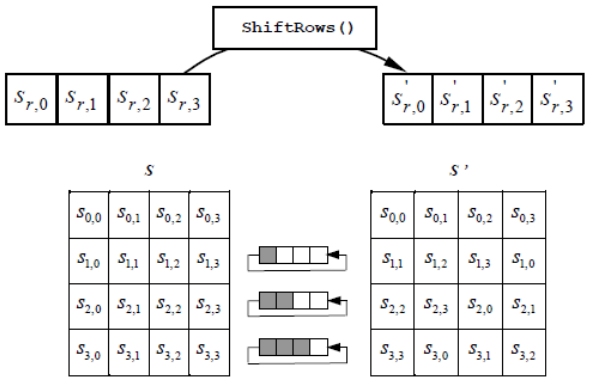
Selanjutnya proses transformasi menggunakan fungsi `ShiftRows()`. Pada fungsi ini, byte pada 3 baris terakhir di rotasikan secara sirkular dengan menjadikan byte ke-*i* pada baris ke-*I* menjadi elemen pertama.

$s'_{r,c} = s_{r,(c + shift(r, Nb)) \bmod Nb}$ for $0 < r < 4$ dan $0 <= c < \boldsymbol{Nb}$,

dengan nilai dari *shift(r, **Nb**)* bergantung pada urutan baris.

shift(1,4) = 1; *shift*(2,4) = 4; *shift*(3,4) = 4.

Transformasi ini berefek pada pemindahan byte ke posisi terendah sesuai dengan urutan barisnya. Ilustrasi pergeseran dari byte pada *state* matriks dapat dilihat melalui Gambar 9.



Gambar 9. Prosedur `ShiftRows()` secara sirkular menggeser posisi byte pada *state* matriks.
Sumber: [5]

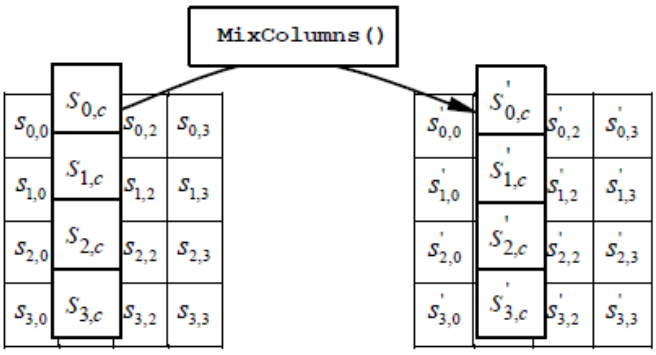
Proses selanjutnya adalah . Transformasi yang dilakukan procedure ini adalah melakukan perkalian modulo $x_4 + 1$ dengan sebuah polinom $a(x)$ yang sudah ditetapkan.

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}.$$

Transformasi perkalian dapat dilihat melalui perkalian matriks di bawah ini.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \quad \text{for } 0 \leq c < \boldsymbol{Nb}.$$

Ilustrasi transformasi prosedur ini dapat dilihat pada Gambar 10 berikut.



Gambar 10. `MixColumns()` beroperasi pada setiap kolom.
Sumber: [5]

Proses terakhir adalah `AddRoundKey()`. Prosedur ini melakukan operasi XOR antara setiap byte pada setiap kolom dan *Round Key* yang berkoresponden dengan byte pada kolom tersebut.

Key Expansion diperlukan untuk menghasilkan kunci pada ronde pengulangan berikutnya. Hal ini digunakan agar setiap ronde, kunci yang digunakan untuk melakukan proses enkripsi berbeda. Hal ini meningkatkan keamanan data yang dienkripsi menggunakan sistem kriptografi ini. *Key Expansion* menghasilkan total ***Nb*** (***Nr***+1) word: pada awal diinisialisasi ***Nb*** word, dan setiap Nr ronde membutuhkan Nb word data kunci. Hasil *Key Expansion* berupa sebuah array berupa 4-byte word. Proses *Key Expansion* dapat dilihat melalui Gambar 11.

```
KeyExpansion(byte key[4*Nk], word w[Nb*(Nr+1)], Nk)
begin
   word  temp

   i = 0

   while (i < Nk)
      w[i] = word(key[4*i], key[4*i+1], key[4*i+2], key[4*i+3])
      i = i+1
   end while

   i = Nk

   while (i < Nb * (Nr+1)]
      temp = w[i-1]
      if (i mod Nk = 0)
         temp = SubWord(RotWord(temp)) xor Rcon[i/Nk]
      else if (Nk > 6 and i mod Nk = 4)
         temp = SubWord(temp)
      end if
      w[i] = w[i-Nk] xor temp
      i = i + 1
   end while
end

Note that Nk=4, 6, and 8 do not all have to be implemented;
they are all included in the conditional statement above for
conciseness.
```

Gambar 11. *Pseudocode* untuk *Key Expansion*.
Sumber: [5]

Proses dekripsi algoritma AES adalah dengan melakukan proses *reverse* pada susunan proses di atas. Oleh karena itu, pada prosedur dekripsi dimuat prosedur invers yaitu, `InvShiftRows()`, `InvSubBytes()`, `InvMixColum()`, dan `AddRoundKey()`.

D. Bluetooth

Teknologi *Bluetooth* merupakan salah satu teknologi nirkabel yang memanfaatkan gelombang radio sebagai media pengiriman data[2]. *Bluetooth* dikembangkan pertama kali oleh Ericson Mobile Communications pada tahun 1994 untuk mengganti kabel sebagai media penghubung antarproduk mereka[6]. Badan yang bertanggung jawab terhadap pengembangan teknologi *Bluetooth* adalah *Bluetooth SIG (Special Interest Group*. Sampai makalah ini ditulis, Bluetooth versi terbaru yang telah diciptakan adalah *Bluetooth* versi 5.4[7].

Proses penyandingan (*pairing*) adalah langkah awal dalam proses koneksi *Bluetooth*. Proses ini melibatkan pertukaran kunci atau kode keamanan antara perangkat untuk memastikan keamanan koneksi. Selanjutnya akan dilakukan proses konfigurasi dan koneksi. Kedua perangkat akan melakukan proses konfigurasi sesuai dengan layanan yang mereka dukung. Selain itu, pada proses ini akan dilakukan juga proses pengaturan parameter seperti frekuensi atau chanel yang digunakan untuk pertukaran data, model transmisi, dan lain-lain. Kemudian, koneksi telah aktif dan kedua perangkat dapat bertukar berkomunikasi dan bertukar data sesuai layanan yang telah diaktifkan.

III. Pengiriman Suara Melalui Bluetooth dengan Algoritma Advanced Encryption Standard (*AES*)

Pengiriman suara yang dilakukan menggunakan algoritma *AES* melalui *Bluetooth* melibatkan proses enkripsi dan dekripsi. Saat pertama kali terkoneksi dengan perangkat lain melalui *Bluetooth*, kedua perangkat telah melakukan pertukaran kunci. Kunci yang dihasilkan bergantung pada perangkat dan protokol yang digunakan.

Data suara seperti dalam format MP3, akan dienkripsi terlebih dahulu sebelum mengirimnya ke perangkat lain melalui *Bluetooth*. Proses ini melibatkan pembagian data menjadi blok-blok kecil yang siap dikirim melalui *Bluetooth*, enkripsi, pembagian data, pengiriman melalui *Bluetooth*, penerimaan dan dekripsi, rekonstruksi data suara, pemrosesan akhir.

Pada proses pembagian data menjadi blok-blok, data dibagi menjadi blok-blok sesuai dengan panjang kunci yang digunakan pada algoritma *AES*, yaitu 128, 192 atau 256 bit. Kemudian, blok-blok data dienkripsi satu per satu menggunakan algoritma *AES*.

Data yang sudah dienkripsi kemudian dibagi menjadi paket-paket kecil yang siap dikirim melalui Bluetooth. Pembagian ini munkin melibatkan penambahan header yang berisi informasi tambahan seperti nomor urut paket, *checksum*, dan lain-lain. Data yang sudah melalui proses ini kemudian akan dikirim.

Proses pengiriman melibatkan protocol *Bluetooth* seperti RFCOMM atau L2CAP, yang mendukung transfer data. Setelah diterima oleh perangkat penerima, paket-paket data didekripsi menggunakan kunci yang sama pada perangkat pengirim. Setelah selesai didekripsi, seluruh paket data akan direkonstruksi dengan digabungkan kembali untuk mengembalikan data ke dalam format semula. Data suara yang telah didekripsi dapat diproses lebih lanjut, seperti dilakukan proses kompresi ulang ke dalam format lain sesuai dengan kebutuhan.

IV. Kesimpulan

Pengiriman data suara melalui Bluetooth yang terenkripsi dapat menggunakan algoritma *Advanced Encryption Standard* atau *AES*. Semakin besar panjang kunci yang digunakan, maka semakin sulit bagi penyerang untuk melakukan pembajakan pada data.

Algoritma AES adalah sebuah algoritma kriptografi simetri yang menggunakan kunci yang sama untuk proses enkripsi dan dekripsi. Algoritma ini menghasilkan kunci yang berbeda untuk proses enkripsi pada setiap ronde untuk menghindari output (chiperteks) yang sama pada blok data yang sama. Hal ini meningkatkan keamanan pada proses pengiriman data.

V. Ucapan Terima Kasih

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena atas rahmat dan berkat-Nya, penulis dapat menyelesaikan makalah ini dengan baik. Penulis juga mengucapkan terima kasih kepada kedua orang tua penulis karena sudah mendukung penulis selama masa penyusunan makalah ini. Penulis mengucapkan terima kasih kepada Bapak Dr. Ir. Rinaldi Munir, M.T., Ibu Fariska Zakhralativa Ruskanda S.T., M.T., dan Ibu Dr. Nur Ulfa Maulidevi S.T., M.Sc selaku dosen mata kuliah IF2120 Matematika Diskrit yang telah membimbing penulis dan memberikan banyak ilmu yang bermanfaat selama perkuliahan. Penulis juga ingin berterima kasih kepada Bapak Made Harta Dwijaksara, S.T., M.Sc., Ph.D.

yang telah memberikan referensi yang sangat bermanfaat dalam menyelesaikan makalah ini.

REFERENSI

[1] R. Munir, "Teori Bilangan (Bagian 1)." 2020. Diakses: 9 Desember 2023. [Daring]. Tersedia pada: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2023-2024/14-Teori-Bilangan-Bagian1-2023.pdf

[2] M. H. Dwijaksara, "Studi dan Implementasi Kriptografi Kunci-Publik untuk Otentikasi Perangkat dan Pengguna pada Komunikasi Bluetooth." Teknik Informatika, 2008.

[3] S. Bruce, "Description os a New Variable Length Key, 64-bit Block Cipher (Blowfish)." 1996.

[4] R. Munir, "Diktat Kuliah IF5045 Kriptografi." 2004.

[5] M. J. Dworkin, "Advanced Encryption Standard (AES)," National Institute of Standards and Technology, Gaithersburg, MD, NIST FIPS 197-upd1, 2023. doi: 10.6028/NIST.FIPS.197-upd1.

[6] A. N. Klingsheim, *J2ME Bluetooth Programming*. 2004.

[7] B. SIG, "Bluetooth Specification 5.4." Februari 2023.

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 2023

Suthasoma Mahardhika Munthe 135220908