Enkripsi Aturan Dinamis pada Aplikasi Pembayaran Gerak

Emir Husni[1], Danang Triantoro Murdiansyah[2]

***Abstract*—The trend of financial transactions using mobile phone or mobile payment is increasing. Using the mobile payment service, user can save money on the mobile phone and separate the money from prepaid bills. In order to protect the user, provider must equip the mobile payment service with transaction security, such as secure mobile payment application. This paper offers a safety feature which is used for mobile payment application based on Android operating system. A new method, Dynamic Rule Encryption (DRE), is created. In DRE, the encryption is ruled dynamically. By encrypting data with dynamic rules, DRE can protect the data. DRE also has a function as a token for authentication. DRE Token is generated using dynamic time-based rules. The time reference used in DRE is based on the order of the day in the year (day of the year). Performance measurement in this research is based on the Hamming distance. The test results show that the system performs quite good. The average distance among outputs is about 258 bits of 512 bits input. The results also show that the DRE program's execution time is quite fast, i.e. not more than 24 ms.**

***Intisari*—Tren transaksi keuangan dengan menggunakan telepon seluler atau pembayaran gerak semakin meningkat. Dengan menggunakan layanan pembayaran gerak, pengguna dapat menyimpan uangnya di telepon seluler dan terpisah dari pulsa telepon. Untuk melidungi pengguna layanan pembayaran gerak, penyedia layanan harus melengkapi layanan pembayaran gerak dengan keamanan transaksi yang baik. Salah satu cara untuk memberikan keamanan transaksi adalah dengan menggunakan aplikasi pembayaran gerak yang aman. Makalah ini menawarkan suatu fitur keamanan yang digunakan untuk aplikasi pembayaran gerak berbasis sistem operasi Android. Dibuat sebuah metode baru, yaitu enkripsi yang menggunakan aturan dinamis yang diberi nama Enkripsi Aturan Dinamis atau *Dynamic Rule Encryption* (DRE). DRE memiliki kemampuan untuk melindungi data dengan cara mengenkripsi data dengan aturan dinamis, dan DRE juga memiliki fungsi sebagai koin untuk autentikasi. Koin DRE dibangkitkan dengan menggunakan aturan dinamis berdasarkan waktu. Acuan waktu yang digunakan adalah berdasarkan urutan hari dalam tahun (*day of year*). Pengukuran kinerja pada makalah ini adalah berdasarkan jarak Hamming. Hasil pengujian menunjukkan jarak rata-rata antara keluaran yang satu dengan keluaran yang lain, sekitar 258 bit dari masukan 512 bit, sehingga dapat dikatakan baik. Pengujian waktu menunjukkan bahwa program DRE memiliki waktu eksekusi yang cukup cepat, yaitu tidak lebih dari 24 ms.**

***Kata Kunci*— pembayaran gerak, enkripsi, aturan dinamis, data sensitif, keamanan.**

[1,2]*Sekolah Teknik Elektro & Informatika, Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132 Indonesia (e-mail: ehusni@lskk.ee.itb.ac.id)*

I. PENDAHULUAN

Penetrasi telepon seluler di Indonesia, menurut Roy Morgan Research, telah mencapai 84% pada bulan Maret 2013. Meskipun angka ini masih didominasi oleh telepon seluler konvensional, penetrasi telepon pintar adalah dua kali lipat dari tahun sebelumnya, dari 12% menjadi 24% [1].

Tren saat ini adalah bahwa jumlah telepon seluler menurun perlahan-lahan sementara jumlah telepon pintar tumbuh pesat. Selain itu, sumber daya komputasi dalam peningkatan telepon pintar setiap tahun dengan aplikasi yang lebih dapat diintegrasikan ke dalamnya. Karena berbagai aplikasi, telepon pintar menjadi perangkat elektronik yang paling umum digunakan, menurut survei yang dilakukan oleh Forrester [2]. Berbagai aplikasi telepon pintar telah dihasilkan oleh pengembang di Indonesia, sebagai contoh: Dewi dan Pramono telah berhasil mengembangkan aplikasi pencatatan servis mobil berbasis Android [3], Sunarya, Halomoan, dan Ruswanda berhasil menghasilkan aplikasi rekam medis berbasis Android dan menggunakan teknologi RFID [4].

Tren transaksi keuangan dengan menggunakan telepon seluler atau pembayaran gerak semakin meningkat. Jumlah operator yang menyediakan layanan tersebut juga bertambah. Dengan menggunakan layanan pembayaran gerak, pelanggan dapat menyimpan uangnya di telepon seluler dan terpisah dari pulsa telepon. Jadi, telepon seluler berfungsi sebagai dompet elektronik. Menurut seorang pengamat telekomunikasi dari Masyarakat Telematika Indonesia (Mastel), peluang pertumbuhan pembayaran gerak di Indonesia sangat besar, karena penetrasi telepon seluler di Indonesia sudah mencapai 90% sementara penetrasi perbankan masih kecil [1].

Saat ini, sebagian besar perangkat Android terbaru dari berbagai produsen mendukung *Near Field Communication* (NFC). NFC telah diimplementasikan dalam aplikasi pembayaran gerak [5].

Penyedia layanan pembayaran gerak harus melengkapi layanan pembayaran gerak dengan keamanan transaksi yang baik. Salah satu cara untuk memberikan keamanan transaksi adalah dengan menggunakan aplikasi pembayaran gerak yang aman. Aplikasi pembayaran gerak harus memiliki kemanan yang baik, karena pada aplikasi tersebut akan terdapat data sensitif, seperti akun, kata sandi, PIN, uang elektronik, dan data transaksi. Makalah ini menawarkan suatu fitur keamanan yang digunakan untuk aplikasi pembayaran gerak berbasis sistem operasi Android. Fitur keamanan ini adalah enkripsi aturan dinamis yang diberi nama *Dynamic Rule Encryption* (DRE), yang dibuat oleh penulis karena belum ada metode semacam ini yang dapat digunakan oleh penulis untuk fitur keamanan aplikasi pembayaran gerak. DRE memiliki kemampuan untuk melindungi data dengan cara mengenkripsi data dengan aturan dinamis. DRE juga memiliki fungsi

sebagai koin untuk autentikasi. Koin dibangkitkan DRE dengan menggunakan aturan dinamis atau berubah-ubah berdasarkan waktu. Acuan waktu yang digunakan adalah berdasarkan urutan hari dalam tahun (*day of the year*). Banyak aturan yang ada pada DRE terbatas sampai waktu tertentu. Hal ini menyebabkan perangkat harus terkoneksi ke peladen untuk memperbaharui aturan DRE. Dengan terhubung ke peladen, peladen dapat melakukan proses validasi terhadap data yang ada pada aplikasi pembayaran gerak, sehingga jika ada data tidak valid, data tersebut akan terdeteksi. Selain itu, pada koin DRE terdapat informasi tentang perangkat, waktu, akun pembayaran gerak pengguna, dan informasi lainnya yang digunakan untuk autentikasi.

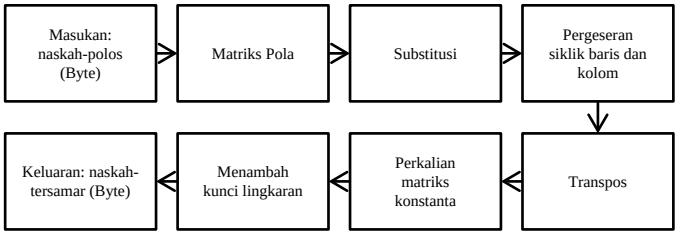
II. ENKRIPSI ATURAN DINAMIS

Enkripsi Aturan Dinamis (DRE) merupakan sandi kotak dengan kunci simetris. Ukuran sandi kotak DRE adalah 512 bit, sedangkan panjang kuncinya adalah 128 bit. Pada DRE terdapat enam tahap yang dilakukan pada naskah-polos, yaitu:

1. pembentukan matriks pola,
2. substitusi,
3. pergeseran siklik elemen berdasarkan baris dan kolom,
4. transpos,
5. perkalian dengan matriks konstanta, dan
6. menambah kunci lingkaran.

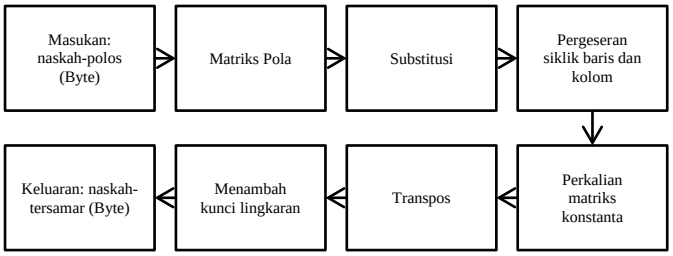
Tahap substitusi, pergeseran siklik, dan transpos berfungsi sebagai penyamaran dan difusi. Penyamaran berfungsi membuat korelasi antara naskah-tersamar dan kunci hilang atau tidak terlihat. Penyamaran merupakan fungsi yang non-linear. Difusi membuat korelasi antara naskah-polos dan naskah-tersamar hilang atau tidak terlihat. Dengan menggunakan difusi, kode akan sulit dipecahkan dengan menggunakan metode statistik.

Pada DRE, urutan tahapan dapat berubah berdasarkan waktu. Gbr. 1 adalah salah satu contoh perubahan tahapan tersebut.



Gbr. 1 Tahapan algoritme.

Pada waktu yang lain, urutan tahapan berubah menjadi seperti pada Gbr. 2.



Gbr. 2 Tahapan algoritme dengan pola berbeda.

Perubahan tersebut hanya sebagai salah satu contoh saja. DRE memiliki berbagai perubahan pola urutan. Selain itu, DRE memiliki elemen ataupun variabel yang dapat berubah berdasarkan waktu, yaitu:

1. pola penempatan masukan pada matriks,
2. elemen matriks yang distitusi,
3. baris dan kolom yang akan digeser dan besar pergeserannya,
4. matriks konstanta, dan
5. kunci utama untuk tahap menambah kunci lingkaran.

A. *Proses Enkripsi*

Berikut adalah penjelasan proses yang ada pada enkripsi.

1) Pembentukan matriks pola: Pada tahap ini, masukan DRE akan dibagi menjadi beberapa bagian dan bagian tersebut akan dibentuk berdasarkan suatu pola ke dalam matriks. Pola penempatan masukan pada matriks berubah berdasarkan waktu. Gbr. 3 memberikan contoh perubahannya, di mana elemen matriks dengan warna berbeda menunjukkan bagian yang berbeda.

byte0	byte1	byte2	byte3	byte4	byte5	byte6	byte7
byte8	byte9	byte10	byte11	byte12	byte13	byte14	byte15
byte16	byte17	byte18	byte19	byte20	byte21	byte22	byte23
byte24	byte25	byte26	byte27	byte28	byte29	byte30	byte31
byte32	byte33	byte34	byte35	byte36	byte37	byte38	byte39
byte40	byte41	byte42	byte43	byte44	byte45	byte46	byte47
byte48	byte49	byte50	byte51	byte52	byte53	byte54	byte55
byte56	byte57	byte58	byte59	byte60	byte61	byte62	byte63

Keterangan: Biru = SIM , hijau = IMEI, kuning = stempel-waktu, abu = nomor akun, orange = besar transaksi, putih = garam.

Gbr. 3 Elemen matriks 1.

Dengan adanya perubahan waktu, matriks pada Gbr. 3 dapat berubah menjadi matriks seperti pada Gbr. 4.

byte0	byte1	byte2	byte3	byte4	byte5	byte6	byte7
byte8	byte9	byte10	byte11	byte12	byte13	byte14	byte15
byte16	byte17	byte18	byte19	byte20	byte21	byte22	byte23
byte24	byte25	byte26	byte27	byte28	byte29	byte30	byte31
byte32	byte33	byte34	byte35	byte36	byte37	byte38	byte39
byte40	byte41	byte42	byte43	byte44	byte45	byte46	byte47
byte48	byte49	byte50	byte51	byte52	byte53	byte54	byte55
byte56	byte57	byte58	byte59	byte60	byte61	byte62	byte63

Keterangan: Biru = SIM , hijau = IMEI, kuning = stempel-waktu, abu = nomor akun, orange = besar transaksi, putih = garam.

Gbr. 4 Elemen matriks 2.

Perubahan tersebut hanya sebagai contoh saja. Terdapat berbagai macam perubahan pola yang dapat terjadi.

2) Substitusi: Pada tahap ini elemen *byte* pada matriks disubstitusi dengan nilai lain berdasarkan tabel pencocokan. Banyaknya elemen pada matriks yang akan disubstitusi adalah sebanyak 48 *byte* elemen. Substitusi elemen *byte* matriks

ditentukan oleh pola bit. Misal bentuk bit pada elemen *byte* matriks adalah b7 b6 b5 b4 b3 b2 b1 b0, di mana b0 adalah LSB dan b7 adalah MSB. Pasangan substitusi ditentukan dengan mengubah urutan bit pada *byte* mengikuti pola bit b7 b3 b5 b1 b6 b2 b4 b0. Pergantian *byte* elemen matriks dilakukan per *nibble* pada *byte* tersebut, yaitu b7 b3 b5 b1 disebut *nibble* 1 dan b6 b2 b4 b0 disebut *nibble* 2. Pergantian dilakukan dengan melihat tabel pencocokan. Tabel I adalah tabel pencocokan untuk *nibble* 1 (kiri) dan untuk *nibble* 2 (kanan).

TABEL I
PENCOCOKAN SUBSTITUSI

Input	Output	Input	Output
0000	0111	1000	0011
0001	1110	1001	1010
0010	0100	1010	1000
0011	1101	1011	0110
0100	0001	1100	0101
0101	0010	1101	1100
0110	1111	1110	1001
0111	1011	1111	0000

Input	Output	Input	Output
0000	1101	1000	0111
0001	1111	1001	0101
0010	1100	1010	1011
0011	1000	1011	0011
0100	0010	1100	1110
0101	0100	1101	1010
0110	1001	1110	0000
0111	0001	1111	0110

Berikutnya pada Tabel II adalah contoh substitusi elemen matriks.

TABEL II
CONTOH SUBSTITUSI ELEMEN MATRIKS

Masukan	Pola		Keluaran
	Nibble2 (b7 b3 b5 b1)	Nibble1 (b6 b2 b4 b0)	
1000 1001	1100	0001	1110 1110
0110 0010	0011	1000	1000 0011
0110 0101	0010	1101	1100 1100
1000 0011	1001	0001	0101 1110

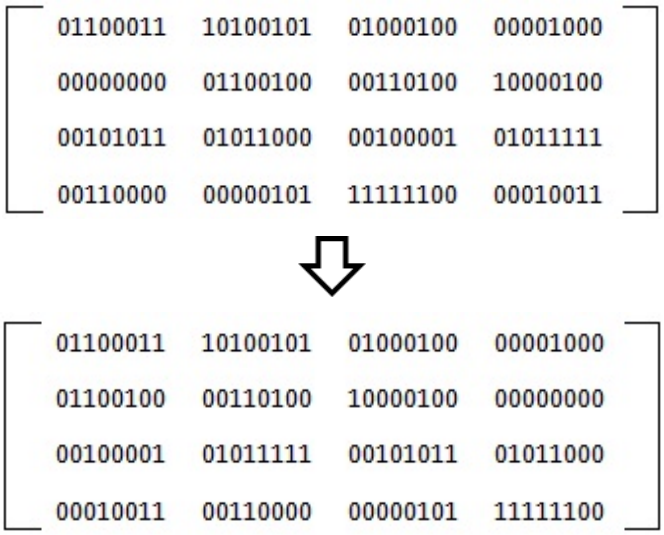
3) Pergeseran siklik elemen berdasarkan baris dan kolom: Pada tahap ini, elemen *byte* pada matriks akan digeser berdasarkan baris ataupun kolom sebesar yang telah ditentukan. Baris ke berapa dan kolom ke berapa yang digeser, serta besar pergeserannya dapat berubah-ubah berdasarkan waktu. Pergeseran ini dapat juga disebut rotasi. Berikut diberikan contoh pergeseran siklik pada matriks berukuran 4 x 4. Gbr. 5 adalah contoh pergeseran siklik baris dengan besaran pergeseran baris pada Tabel III. Sedangkan Gbr. 6 adalah contoh pergeseran siklik kolom dengan besaran pergeseran kolom pada Tabel IV.

TABEL III
CONTOH PERGESERAN SIKLIK BARIS

Baris	1	2	3	4
Besar geser putaran ke kiri	0	1	2	3

4) Transpos: Tahap ini adalah tahap operasi transpos pada matriks. Operasi transpos adalah operasi pada matriks di mana elemen baris menjadi elemen kolom dan elemen kolom menjadi elemen baris. Gbr. 7 adalah contoh transpos pada matriks 4 x 4.

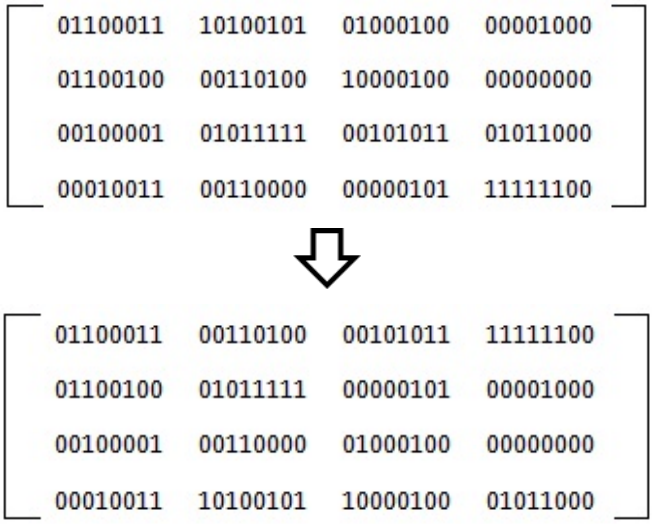
5) Perkalian dengan matriks konstanta: Tahap ini adalah tahap elemen matriks dikalikan dengan konstanta pada matriks. Elemen matriks dikalikan dalam bentuk *nibble*. Matriks konstanta yang menjadi pengali dapat berubah-ubah berdasarkan waktu. Perkalian antara matriks dengan matriks konstanta ini dilakukan dalam medan Galois (Galois *Field* atau disingkat GF), yaitu GF(2^4). Gbr. 8 adalah contoh salah satu matriks konstanta dalam GF(2^4).



Gbr. 5 Contoh pergeseran siklik baris.

TABEL IV
CONTOH PERGESERAN SIKLIK KOLOM

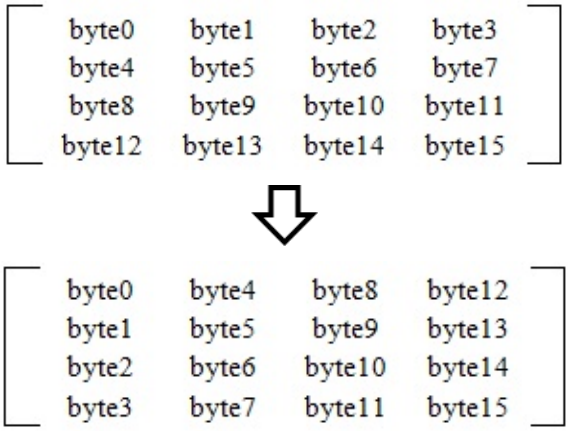
Kolom	1	2	3	4
Besar geser putaran ke atas	0	1	2	3



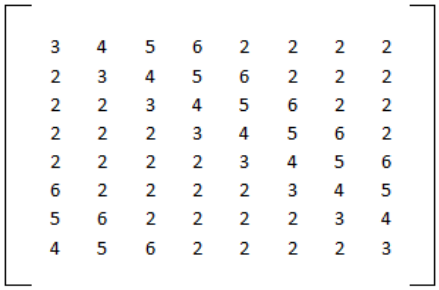
Gbr. 6 Contoh pergeseran siklik kolom.

6) Menambah kunci lingkaran: Pada tahap menambah kunci lingkaran, pada elemen matriks akan dilakukan operasi lingkaran *(*operasi *xor).* Lingkaran menggunakan kunci utama 128 bit dan kunci utama akan diekspansi sehingga menghasilkan subkunci 512 bit. Saat membangun subkunci,

proses ini melibatkan matriks pembantu M yang berisi kunci utama dan iterasinya dengan modulus 255. Gbr. 9 adalah matriks M yang dihasilkan dari kunci utama *4C 69 66 65 27 73 20 62 65 61 75 74 69 66 75 6C.* di mana Gbr. 10 adalah algoritme untuk mengekspansi kunci utama 128 bit menjadi subkunci 512 bit.



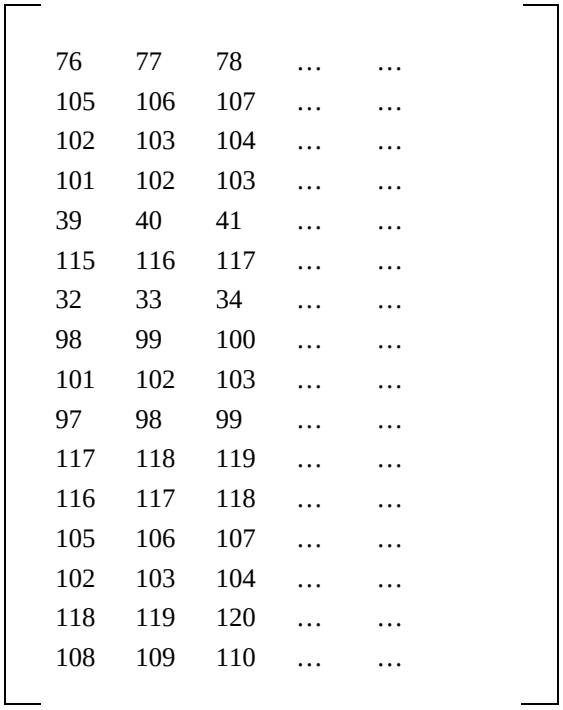
Gbr. 7 Contoh transpos pada matriks 4 x 4.



Gbr. 8 Contoh salah satu matriks konstanta dalam GF(2^4).

Prosedur ekspansi kunci utama adalah sebagai berikut.

1. Kunci utama atau kunci rahasia, K, ditranspos untuk mendapatkan K1.
2. K2 didapatkan dengan melakukan operasi xor antara K dengan K1.
3. Berdasarkan kolom, K2 dibagi dua, yaitu bagian kanan dan bagian kiri. Kedua bagian tersebut dilakukan operasi xor untuk mendapatkan K3.
4. Bagian kanan K2 dan bagian kiri K2 ditranspos. Kedua bagian hasil transpos tersebut kemudian operasi xor untuk mendapatkan K4.
5. K3 dan K4 dilakukan konkatenasi (disatukan) untuk mendapatkan K5.
6. Nilai bilangan bulat dari *byte* K5 dijumlahkan untuk mendapatkan L.
7. Kse1 dihitung dengan rumus Kse1 = L mod C1, dengan C1 = 23.
8. Kse2 dihitung dengan rumus Kse2 = L mod C2, dengan C2 = 15.
9. Kemudian, subkunci Ks2 didapatkan dari penurunan dari matriks pembantu M sebagai berikut.
   Ks1[row][column] = M[row][Kse1+Kse2+column]
   Ks2[row][column] = M[row][Ks1[row][column]]
10. Subkunci Ks1 didapatkan dari penurunan dari matriks pembantu M sebagai berikut.
    Ks1[row][column] = M[row][Ks2[row][column]]
11. Kolom ke-i dari matriks Ks1 dirotasi secara vertikal ke arah bawah sejauh (((int(K[i]) mod 12) + Kse1) mod 15).
12. Kolom ke-i dari matriks Ks2 dirotasi secara vertikal ke arah bawah sejauh (((int(K[i]) mod 10) + Kse1) mod 15).



Gbr. 9 Matriks M.

Algoritme di atas dapat menghasilkan banyak perubahan bit pada subkunci ketika sedikit perubahan terjadi pada kunci utama [6]. Ks1 dan Ks2 masing-masing akan menghasilkan subkunci 256 bit, sehingga total subkunci yang dihasilkan adalah 512 bit. Gbr. 11 adalah contoh beberapa subkunci yang dihasilkan dari kunci utama 4C 69 66 65 27 73 20 62 65 61 75 74 69 66 75 6C.

Setelah naskah-polos melalui enam tahapan di atas, naskah-polos menjadi naskah-tersamar yang akan dikirim dari aplikasi bayar saat transaksi berlangsung.

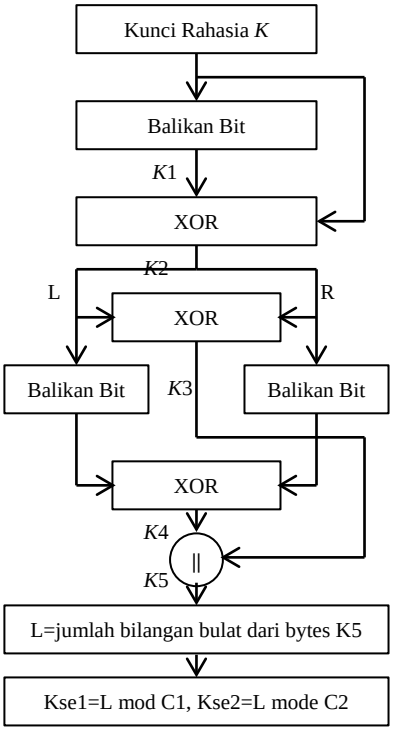
B. Proses Dekripsi

Berikut adalah penjelasan proses yang ada pada dekripsi.

1) Invers substitusi: Tahap ini merupakan kebalikan dari substitusi. Invers substitusi dilakukan dengan menggunakan tabel invers substitusi. Aturan untuk menentukan pasangan invers substitusi sama dengan aturan yang ada pada tahap substitusi enkripsi. Substitusi elemen *byte* matriks dilakukan dengan melihat tabel pencocokan invers substitusi seperti pada Tabel V.

2) Invers pergeseran siklik elemen berdasarkan baris dan kolom: Tahap ini kebalikan dari pergeseran siklik elemen

berdasarkan baris dan kolom pada proses enkripsi**.** Besar dan arah pergeseran merupakan kebalikan pergeseran yang ada pada proses enkripsi.



Gbr. 10 Algoritme ekspansi kunci utama.

6D 6A 49 40 6D 52 F2 49 40 3D 83 67 6E 34 3D 31
68 6F 35 3E 32 6E 6B 4A 41 6E 53 F3 4A 41 3E 84
3F 33 6F 6C 4B 42 6F 54 F4 4B 42 3F 85 69 70 36
34 70 6D 4C 43 70 55 F5 4C 43 40 86 6A 71 37 40
6E 4D 44 71 56 F6 4D 44 41 87 6B 72 38 41 35 71
39 42 36 72 6F 4E 45 72 57 F7 4E 45 42 88 6C 73
74 3A 43 37 73 70 4F 46 73 58 F8 4F 46 43 69 6D
50 47 74 59 F9 50 47 44 8A 6E 75 3B 44 38 74 71
39 75 72 51 48 75 5A FA 51 48 45 8B 6F 76 3C 45
49 76 5B FB 52 49 46 8C 70 77 3D 46 3A 76 73 52

Gbr. 11 Contoh beberapa subkunci yang dihasilkan dari kunci utama.

3) Invers perkalian matriks konstanta: Tahap ini adalah kebalikan dari perkalian matriks konstanta pada proses enkripsi. Perkalian antara matriks dengan invers matriks konstanta ini dilakukan dalam GF(2^4).

4) Invers transpos: Invers transpos sama dengan transpos. Untuk mengembalikan matriks ke keadaan semula, transpos dilakukan sekali lagi.

TABEL V
PENCOCOKAN INVERS SUBSTITUSI

Input	Output	Input	Output
0000	1111	1000	1010
0001	0100	1001	1110
0010	0101	1010	1001
0011	1000	1011	0111
0100	0010	1100	1101
0101	1100	1101	0011
0110	1011	1110	0001
0111	0000	1111	0110

Input	Output	Input	Output
0000	1110	1000	0011
0001	0111	1001	0110
0010	0100	1010	1101
0011	1011	1011	1010
0100	0101	1100	0010
0101	1001	1101	0000
0110	1111	1110	1100
0111	1000	1111	0001

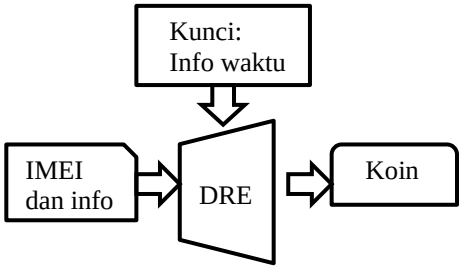
5) Invers menambah kunci lingkaran: Tahap ini merupakan kebalikan dari proses menambah kunci lingkaran yang ada pada tahap enkripsi. Pada elemen matriks akan dilakukan operasi xor sama seperti yang ada pada tahap enkripsi tetapi dengan urutan jadwal subkunci yang terbalik. Proses mendapatkan subkunci dari kunci utama sama dengan yang ada pada tahap enkripsi.

Setelah naskah-tersamar melalui lima tahapan dekripsi di atas, naskah-tersamar menjadi naskah-polos dan data dapat digunakan oleh aplikasi pembayaran gerak. Tahap dekripsi lebih sedikit dari tahap enkripsi. Hal ini karena pada tahap dekripsi tidak ada tahap pembentukan pola pada matriks. Setelah dekripsi dilakukan, aplikasi pembayaran gerak dapat mengambil data yang diinginkan pada lokasi data tersebut disimpan pada matriks pola.

C. DRE Sebagai Koin

Tujuan DRE dibuat adalah untuk mengamankan data yang dipertukarkan saat transaksi menggunakan aplikasi pembayaran gerak dilakukan. Pada aplikasi pembayaran gerak, DRE selain berfungsi sebagai algoritme enkripsi yang mengenkripsi data, juga berfungsi sebagai koin. Koin akan memastikan transaksi adalah valid mulai dari awal hingga akhir transaksi. Dengan adanya fungsi koin, DRE sangat cocok digunakan pada aplikasi pembayaran gerak yang memiliki mode transaksi non-jaringan, karena pada transaksi non-jaringan sangat rentan terhadap penipuan (transaksi dengan perangkat palsu/tidak terdaftar), pengalihan lawan transaksi (serangan orang di tengah), ataupun serangan lainnya.

Pada koin yang dibangkitkan terdapat informasi tentang perangkat pengguna, misalnya IMEI dan IMSI. Selain informasi tentang perangkat pengguna, pada koin juga terdapat informasi waktu, akun pengguna, data transaksi, dan informasi lainnya jika diperlukan. Aturan yang digunakan DRE untuk membangkitkan koin berubah-ubah berdasarkan waktu. Aturan tersebut disimpan dalam basis data DRE. Pada penelitian ini, acuan waktu yang digunakan untuk membangkitkan koin adalah berdasarkan urutan hari dalam tahun (*day of the year*). Gbr. 12 mengilustrasikan diagram blok DRE sebagai koin.



Gbr. 12 Pembangkit koin.

Proses pengamanan ataupun autentikasi yang dilakukan DRE pada transaksi menggunakan aplikasi pembayaran gerak adalah sebagai berikut.

1. Saat transaksi dimulai, dua perangkat yang akan melakukan transaksi membangkitkan koin.
2. Koin kemudian dipertukarkan oleh pembeli (pembayar) dan penjual (penerima bayar). Koin pembeli dikirim ke penjual dan koin penjual dikirim ke pembeli. Pada saat pertukaran data, selain koin yang dikirim, informasi perangkat seperti IMEI dan IMSI juga dikirim secara tidak terenkripsi.
3. Kedua perangkat akan memeriksa koin dengan cara melakukan dekripsi menggunakan aturan yang ada pada basis data DRE di aplikasi pembayaran gerak. Koin penjual akan didekripsi di perangkat pembeli sehingga diperoleh IMEI dan IMSI perangkat penjual, serta data lainnya. Koin pembeli akan didekripsi di perangkat penjual sehingga diperoleh IMEI dan IMSI perangkat pembeli, serta data lainnya.
4. IMEI dan IMSI hasil dekripsi koin akan dibandingkan dengan IMEI dan IMSI yang dikirim dalam format tidak terenkripsi (naskah-polos). Jika IMEI dan IMSI hasil dekripsi sama dengan IMEI dan IMSI yang terkirim dalam naskah-polos maka perangkat lawan transaksi adalah valid untuk melakukan transaksi.

Koin akan selalu dipertukarkan saat pengiriman data antara dua perangkat. Dengan demikian, penipuan ataupun pengalihan lawan transaksi oleh perangkat pihak ketiga saat transaksi berlangsung dapat dicegah.

III. PENGUJIAN

Pengujian yang dilakukan untuk menguji DRE adalah pengujian jarak (jarak Hamming) dan pengujian waktu baik untuk enkripsi maupun dekripsi. Pengujian jarak dapat mencerminkan kekuatan sebuah algoritme sandi, dan pengujian waktu dapat mencerminkan berat atau tidaknya suatu algoritme sandi dijalankan pada suatu perangkat atau sistem operasi.

A. Pengujian Jarak

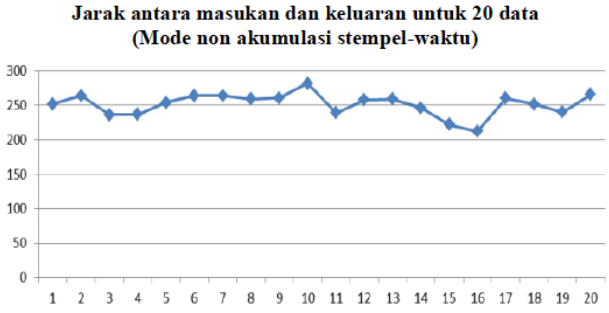
Jarak (jarak Hamming) adalah banyaknya posisi bit berbeda dari dua kata-sandi. Besar jarak yang paling baik dari masukan kata-sandi dan keluaran kata-sandi dari sandi adalah setengah dari total panjang kata-sandi.

Pengujian DRE pada makalah ini dilakukan dengan melakukan 500 kali pengujian. Pada setiap pengujian, data yang menjadi masukan berbeda-beda. Salah satu data yang menjadi masukan adalah informasi waktu, yaitu stempel-waktu. Stempel-waktu yang diambil sebagai masukan adalah stempel-waktu dengan perbedaan setiap 30 detik. Stempel-waktu akumulasi adalah stempel-waktu akumulasi setiap 30 detik. Stempel-waktu non-akumulasi adalah perbedaan stempel-waktu dengan 30 detik sebelumnya. Tabel VI adalah hasil pengujian jarak dari DRE.

TABEL VI
HASIL PENGUJIAN JARAK

No	Pengujian	Rata-rata jarak	
		jarak dengan masukan	jarak dengan keluaran sebelum-nya
1	Menggunakan seluruh tahap (per 30 detik)	253,016	244,84
2	Menggunakan Seluruh Tahap (Akumulasi Penambahan Per 30 Detik)	253,016	257,55
3	Tahap substitusi	262,064	175,73
4	Tahap pergeseran siklik baris	96,108	120,014
5	Tahap pergeseran siklik kolom	179,196	169,136
6	Tahap transpos	191,454	123,558
7	Tahap perkalian dalam GF(2^4)	164,148	143,326
8	Tahap menambah kunci lingkaran	227,656	156,298

Berikutnya adalah hasil pengujian jarak dalam grafik dengan jumlah data yang ditampilkan adalah 20 data. Gbr. 13 menunjukkan hasil pengujian jarak untuk non-akumulasi stempel-waktu dan Gbr. 14 menunjukkan hasil pengujian jarak untuk akumulasi stempel-waktu.

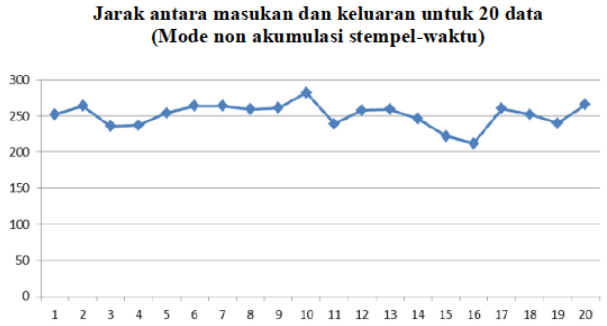


Gbr. 13 Hasil pengujian jarak untuk non-akumulasi stempel-waktu.

Masukan DRE pada percobaan ini adalah data (kata-sandi) dengan ukuran 512 bit (termasuk garam), maka jarak yang paling baik antara kata-sandi masukan dan keluaran adalah 256 bit. Dengan melihat hasil pengujian di atas, dapat disimpulkan DRE memiliki tingkat keamanan yang baik. Dengan menggunakan seluruh operasi atau tahap, DRE dapat menghasilkan jarak rata-rata antara masukan dan keluaran sekitar 256 bit.

Jarak antara keluaran yang satu dengan keluaran yang lainnya, dengan memberikan sedikit perbedaan pada

masukannya, juga dapat mencerminkan tingkat kekuatan sebuah sandi. Jika jarak antara keluaran yang satu dengan keluaran yang lain mendekati setengah panjang kata-sandi, maka hasil enkripsi kata-sandi yang satu dengan yang lain memiliki perbedaan susunan kata-sandi yang semakin baik pula. Dengan menggunakan seluruh operasi, DRE dapat menghasilkan jarak rata-rata antara keluaran yang satu dengan keluaran yang lain sekitar 258 bit. Jarak Hamming yang didapat dengan menggunakan DRE jauh melebihi jarak Hamming jika menggunakan enkripsi yang sudah tersedia (contohnya: AES).



Gbr. 14 Hasil pengujian jarak untuk akumulasi stempel-waktu.

Operasi atau tahap pada enkripsi yang memberikan perbedaan yang paling baik pada kata-sandi adalah operasi substitusi. Operasi substitusi mengasilkan jarak sekitar 262 bit.

B. Pengujian Waktu

Pengujian waktu dapat merepresentasikan berat atau tidaknya sandi dijalankan pada perangkat atau sistem operasi. Berikut adalah hasil pengujian waktu untuk enkripsi dan dekripsi. Pengujian dilakukan sebanyak 1000 kali dengan menggunakan telepon pintar Google Nexus S. Telepon pintar Sony Xperia SK17i juga digunakan dalam pengujian ini yang berfungsi sebagai pembanding. Tabel VII adalah hasil pengujian waktu.

TABEL VII
HASIL PENGUJIAN WAKTU

Perangkat	Rata-Rata Waktu (ms)	
	Enkripsi	Dekripsi
Google Nexus S	19,537	23,519
Xperia SK17 i	18,262	18,428

TABEL VIII
HASIL PENGUJIAN WAKTU RATA-RATA DRE DAN AES

Sandi	Rata-Rata Waktu (ms)	
	Enkripsi	Dekripsi
DRE	19,537	23,519
AES	0,62	1,147

Dilihat dari hasilnya, dapat disimpulkan bahwa DRE tidak berat atau tidak membebani jika dijalankan pada perangkat bergerak seperti telepon pintar. Hal ini diperkuat dengan hasil pengujian menggunakan perangkat pembanding Sony Xperia SK17i. Dengan menggunakan Sony Xperia SK17i, enkripsi dan dekripsi DRE hanya membutuhkan waktu sekitar 18 ms.

Hasil pengujian waktu DRE juga dibandingkan dengan hasil pengujian waktu enkripsi *Advanced Encryption Standard* (AES) [7]. Kedua pengujian menggunakan perangkat Google Nexus S dengan 1000 kali pengujian. Tabel VIII memberikan hasil perbandingan pengujian waktu antara DRE dan AES.

Jika dibandingkan dengan kecepatan AES, kecepatan DRE lebih lambat. Dengan menggunakan Google Nexus S, waktu yang dibutuhkan AES untuk melakukan enkripsi adalah 0,62 ms. Waktu yang dibutuhkan AES untuk melakukan dekripsi adalah 1,15 ms. Penyebab DRE tertinggal jauh dari AES dalam hal kecepatan adalah karena fungsi AES dirancang dalam bahasa pemrogaman native (C/C++), sehingga dapat melakukan akses memori, perhitungan *integer* dan perhitungan *floating-point* lebih cepat daripada DRE yang dirancang menggunakan bahasa pemrogaman Android [8].

IV. KESIMPULAN

Algoritme DRE dirancang sebagai salah satu elemen keamanan pada aplikasi pembayaran gerak yang dalam makalah ini berfungsi untuk mengamankan data sensitif dan juga sebagai koin yang melakukan autentikasi ketika bertransaksi.

Berdasarkan pengujian jarak yang dilakukan pada penelitian ini, DRE memiliki tingkat keamanan yang baik karena jarak rata-rata antara masukan dan keluaran adalah setengah dari panjang data yang menjadi masukan DRE. Dengan demikian, naskah-polos tidak akan mudah dilacak dari naskah-tersamar. Berdasarkan pengujian waktu yang dilakukan, DRE ringan digunakan pada perangkat bergerak karena memiliki waktu eksekusi yang cukup cepat, yaitu tidak lebih dari 24 ms.

REFERENSI

[1] R. Chadha. (2013) “Indonesia Online: A Digital Economy Emerges, Fueled by Cheap Gerak Handsets,” http://www.slideshare.net/slideshow/ embed_code/21155895, tanggal akses: 1 September 2015.

[2] G. Tsirulnik (2010), “Gerak Phone Ranked Most Used Electronic Device: Forrester,” http://www.gerakmarketer.com/cms/news/research/7473.html.

[3] C. Dewi, K.N. Pramono, “Pembuatan Aplikasi Pencatatan Servis Mobil di PT. Armada International Motor Berbasis Android,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, Vol. 4, No. 4, 2015.

[4] U. Sunarya, J. Halomoan, G.A.P. Ruswanda, “Perancangan Rekam Medis PPTM Berbasis Android dan Mikrokontroler Menggunakan Teknologi RFID,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, Vol. 4, No. 1, 2015.

[5] H. Ubaya, “Perancangan Prototipe Sistem Aplikasi Bayar dengan Teknologi Near Field Communication (NFC) Berbasis Android,” Tesis MT., Institut Teknologi Bandung, Bandung, 2011.

[6] A.J. Paul, P. Mythili, J.K. Paulus, “Matrix Based Key Generation to Enhance Key Avalanche in Advance Encryption Standard,” *Proceedings of International Conference on VLSI, Communication & Instrumentation*, 2011.

[7] C. Paar, J. Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*, Berlin Heidelberg: Springer-Verlag, 2010.

[8] J.Y. Lee, J.K. Lee, “Android Programming Techniques for Improving Performance,” *Proceedings of 3rd International Conference iCAST*, 2011.