Sistem Autentifikasi Transaksi Kartu Kredit dengan *Chip* EMV

Ramda Yanurzha - 13506011
Jurusan Teknik Informatika Institut Teknologi Bandung
Jalan Ganesha 10 Bandung 40132
email: if16011@students.if.itb.ac.id

Abstraksi *– Makalah ini membahas sistem autentifikasi dalam transaksi dengan menggunakan kartu kredit. Proses yang akan dibahas mencakup pembacaan data dari terminal pembayaran sampai otorisasi sebuah transaksi dari pihak bank. Proses tersebut dibutuhkan agar sebuah transaksi bersifat valid dan tidak disalahgunakan oleh pihak yang tidak bertanggung jawab. Sistem keamanan yang tercakup meliputi validasi nomor kartu kredit, enkripsi nomor PIN, dan server pemroses transaksi. Dalam makalah ini, kartu kredit yang dibahas adalah kartu kredit yang memiliki chip sebagai pelengkap pita magnetik.*

Kata Kunci*: kartu kredit, enkripsi, PIN, algoritma Luhn, EMV, DES, 3DES, CSC, PoS, ISO 7810*

1. PENDAHULUAN

Sepanjang sejarah, masalah transaksi keuangan adalah hal utama yang mendasari industri perbankan. Di masa sekarang ini, transaksi umumnya tidak dilakukan secara tunai, namun berupa transaksi elektronis (cashless). Dengan transaksi elektronis, proses yang terjadi adalah pemindahbukuan yang hanya menghitung angka. Bentuk fisik dari transaksi tersebut tidak ada kecuali jika nasabah melakukan penarikan uang untuk pembayaran.

Dalam perkembangannya, mata rantai berupa penarikan uang tersebut mempunyai alternatif yang lebih praktis, yaitu kartu kredit. Kartu kredit adalah sebuah kartu berisi identitas nasabah dan dapat digunakan untuk melakukan pembayaran secara elektronis. Tabungan nasabah akan didebet oleh pihak bank secara otomatis begitu transaksi dilakukan, lalu pihak bank yang nantinya akan membayar penjual (*merchant*).

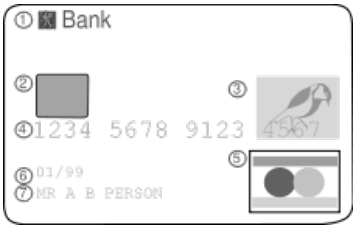
Kartu kredit di masa sekarang ini penggunannya sudah umum dan merupakan cara pembayaran yang paling praktis dan aman dalam kehidupan masyarakat modern. Dengan penggunaan komputer dan jaringan, kesalahan manusia dan penipuan dapat diminimalisasi. Dalam makalah ini, penulis akan membahas segi keamanan kartu kredit dan proses terjadinya transaksi.

2. PEMBAHASAN

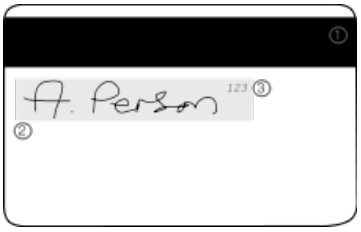
2.1 Kartu Kredit

Kartu kredit pertama kali mulai digunakan sekitar tahun 1920 di Amerika Serikat, namun konsep penggunaaan kartu kredit yang dikenal saat ini baru diciptakan pada tahun 1950. Sistem yang banyak digunakan sekarang ini adalah VISA dan MasterCard yang merupakan gabungan dari berbagai institusi yang bekerja sama dalam membuat standar kartu kredit.

Kartu kredit adalah sebuah kartu yang terbuat dari plastik. Sesuai standar ISO 7810 ID-1, kartu kredit berukuran 85,60 x 53,98 mm (3,370 × 2,125 in) dengan ketebalan 0,76 mm dan sudut membulat dengan radius 3,18 mm. Kartu kredit memiliki beberapa penanda fisik pada kedua sisinya. Di sisi depan terdapat nama bank penerbit, nama nasabah, merek kartu, logo hologram, nomor kartu, tanggal berlaku, dan chip EMV. Di bagian belakang terdapat pita magnetik, tanda tangan nasabah, dan *card security code* (CSC). Data nasabah disimpan dalam pita magnetik sedangkan chip EMV digunakan dalam proses verifikasi.



Gambar 2.1.1 Bagian depan kartu kredit



Gambar 2.1.2 Bagian belakang kartu kredit

Standar kartu lebih lanjut dideskripsikan dalam standar ISO sebagai berikut:

1. ISO 7813 menjabarkan karakteristik tambahan dari kartu ID-1
2. ISO 7811 menjabarkan teknik umum penyimpanan data ke dalam kartu dengan media magnetik
3. ISO 7816 menjabarkan kartu ID-1 yang mempunyai chip EMV, misalnya tentang spesifikasi daya dan sinyal

4. ISO 14443 menjabarkan kartu ID-1 dengan chip tanpa kontak dan antena magnetik yang beroperasi di frekuensi 13,56 MHz (RFID) dan cara penyimpanan data biometris

2.2 Pita Magnetik

Sesuai ISO 7813, data disimpan dalam pita magnetik pada bagian belakang kartu. Pita magnetik terbuat dari partikel magnetik yang orientasinya merepresentasikan bit (1 atau 0) dari data yang disimpan. Partikel tersebut disusun sehingga membentuk pita lalu dilapisi film. Pita magentik terletak 5,66 mm dari ujung kartu dan mempunyai lebar 9,52 mm. Dalam pita tersebut terdapat 3 track data, masing-masing mempunyai lebar 2,79 mm.

Dalam dunia perbankan, yang dipakai adalah track pertama (A) dan kedua (B). Kedua track tersebut mempunyai densitas data 210 bit per inci. Setiap track dapat memuat karakter alfanumerik 7-bit atau karakter numerik 5-bit. Track pertama biasanya memuat karakter alfanumerik yang berisi sebagai berikut:

1. Start sentinel — 1 karakter (biasanya adalah '%')
2. Format code="B" — 1 karakter (alfabet)
3. Primary account number — hingga 19 karakter
4. Field Separator — 1 karakter (biasanya '^')
5. Name — nama nasabah, 2 sampai 26 karakter
6. Field Separator — 1 karakter (biasanya '^')
7. Expiration date — 4 karakter
8. Service code — 3 karakter
9. Discretionary data — biasanya berupa CSC, 3 karakter
10. End sentinel — 1 karakter (biasanya adalah '%')
11. Longitudinal redundancy check (LRC) — 1 karakter

Data tersebut dibaca oleh mesin Point-of-Sale (PoS) terminal dan bersama data transaksi serta kode validasi dikirimkan ke bank penerbit kartu.

Gambar 2.2.1 Terminal PoS

2.3 Chip EMV

Chip EMV adalah IC yang dimasukkan ke dalam kartu. Chip ini digunakan sebagai penyimpanan maupun pemroses data. Walaupun chip ini berukuran kecil (sekitar 3x5 mm) namun tersambung dengan bagian kontak logam yang berada di sisi luar kartu. Sejak tahun 1993, kartu yang menggunakan chip ini mengikuti standar EMV (Europay MasterCard Visa) dan dapat digunakan dalam kartu kredit maupun debit.

Gambar 2.3.1 Kartu kredit dan chip EMV

Chip ini tidak digunakan untuk menyimpan data pribadi nasabah, namun menyimpan prosesor kriptografik yang berguna untuk validasi sebuah transaksi. Jika proses autentifikasi dengan server bank penerbit gagal, maka transaksi tidak dapat dilakukan. Spesifikasi chip diatur dalam standar ISO 7816 yang mencakup bentuk fisik, posisi kontak, fungsionalitas, dan protokol komunikasi. Chip ini tidak mempunyai sumber listrik sendiri karena arus disediakan oleh terminal PoS.

2.4 Nomor Kartu

Setiap kartu kredit memiliki nomor yang unik sepanjang 10 digit. Nomor ini berguna sebagai identifikasi kartu saat transaksi dilakukan dan mewakili satu akun nasabah. Saat pembayaran, nomor kartu akan divalidasi dengan algoritma Luhn. Algoritma ini berguna untuk mencegah input nomor kartu yang salah. Cara kerja algoritma Luhn secara sederhana adalah sebagai berikut:

1. Dalam sebuah string berisikan angka, program mulai dari digit terakhir (paling kanan) lalu melakukan traversal ke digit pertama (paling kiri)
2. Untuk setiap digit kedua, program mengalikan digit tersebut dengan 2
3. Setelah sampai ke digit pertama, program menjumlahkan setiap digit yang sudah ditraversal dan menyimpan jumlahnya
4. Jika jumlah akhir dapat dibagi 10 dan sisanya adalah 0, maka string angka tersebut valid

Berikut ini diberikan algoritma Luhn dalam bahasa C:

```
static int
isValidNumber(const char *number)
{
    int n, i, alternate, sum;

    if (!number)
        return 0;

    n = strlen(number);

    if (n < 13 || n > 19)
        return 0;

    for (alternate = 0, sum = 0,
i = n - 1; i > -1; --i)
{        if (!isdigit(number[i]))
            return 0;

        n = number[i] - '0';

        if (alternate) {
            n *= 2;
            if (n > 9)
                n = (n % 10) + 1;
        }
        alternate = !alternate;

        sum += n;
    }

    return (sum % 10 == 0);
}
```

Algoritma Luhn tidak mengautentifikasi nomor kartu, melainkan hanya memvalidasinya. Jika algoritma ini menghasilkan kondisi benar, maka transaksi dilanjutkan untuk proses autentifikasi lebih lanjut.

2.5 Enkripsi EMV

Seperti yang sudah dibahas, chip EMV dalam kartu kredit bertujuan sebagai prosesor kriptografi. Standar kriptografi yang umum digunakan adalah Triple DES (Data Encryption Standard) yang merupakan pengembangan dari DES berbasis *block cipher* biasa. Algoritma enkripsi DES sendiri dianggap tidak aman, karena dengan metode kriptoanalisis linear enkripsi tersebut dapat dipecahkan dalam waktu singkat. Walaupun begitu, dengan membuat algoritma tersebut menjadi rangkap tiga, kerumitan dalam pemecahannya dianggap cukup untuk keamanan transaksi. DES mempunyai ukuran blok sebesar 64 bit, namun hanya dipakai 56 bit dan menyisakan 8 bit lainnya untuk parity check. Algoritma DES secara global berjalan sebagai berikut:

1. Blok plaintext dipermutasi dengan matrik permutasi awal (*initial permutation atau IP*). Plaintext merupakan data (teks) yang akan dienkripsi. *Plaintext* ini direpresentasikan sebagai bit, misalnya huruf P direpresentasikan sebagai 01010000. Keseluruhan *plaintext* dibagi ke dalam blok-blok. Setiap blok terdiri atas 64 bit.
2. Hasil permutasi awal kemudian dienkripsikan dengan fungsi Feistel sebanyak 16 kali (16 putaran). Setiap putaran menggunakan kunci internal yang berbeda.
3. Hasil enkripsi kemudian dipermutasikan dengan matriks balikan (*invers initial permutation atau IP-1*) menjadi blok *ciphertext*.

Proses utama dalam DES adalah fungsi Feistel (Fungsi-F) yang langkah kerjanya adalah sebagai berikut:

1. Blok dipecah menjadi 2 (64 bit menjadi 32 bit)
2. Blok yang sudah dipecah tersebut diekspansi menjadi 48 bit dengan menggandakan bit yang sudah ada
3. Hasilnya kemudian dikombinasikan dengan subkunci menggunakan operasi XOR. Subkunci isinya berbeda-beda untuk setiap pengulangan fungsi dan didapat dari kunci utama dari user melalui proses *key schedule*.
4. Blok yang sudah dikombinasikan dibagi menjadi 8 bagian yang masing-masing berukuran 6 bit dan masing-masing masuk ke proses substitusi yang disebut S-box. Dalam proses ini, setiap bagian digantikan dengan 4 bit output dengan menggunakan transformasi nonlinear yang disediakan oleh lookup table.
5. Setelah melalui proses substitusi, ke 32 bagian output disusun dengan proses permutasi yang pada akhirnya memberikan blok yang sudah terenkripsi

S_1															
14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S_2															
15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S_3															
10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S_4															
7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S_5															
2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S_6															
12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S_7															
4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S_8															
13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

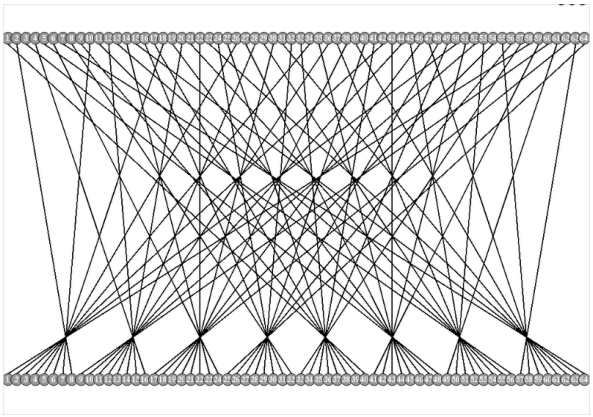
Tabel 2.5.1 Lookup table S-box

Tahap S-box adalah tahap terpenting dalam algoritma enkripsi DES. Misalnya diketahui sebuah karakter mempunyai bentuk biner 010011. S-box akan mengambil 2 bit paling luar (01) dan 4 digit di tengah (1001). Dengan begitu, pada S1 karakter tersebut akan diubah menjadi karakter pada baris ke-2 (01) dan kolom ke-9 (1001), yaitu 10 (1010). Proses tersebut

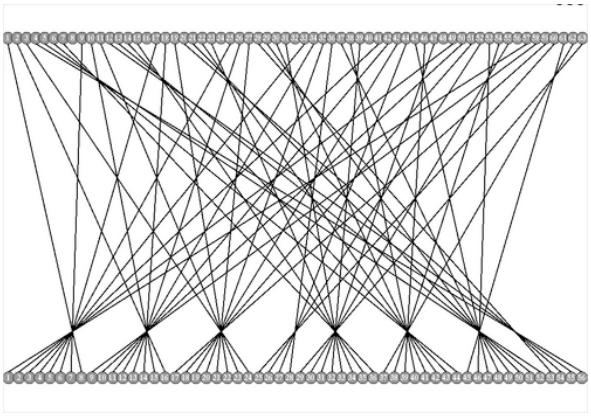
dilakukan sampai terbentuk 16 variasi kunci.

Proses *key schedule* juga dibutuhkan untuk menghasilkan subkunci unik sebanyak 16 buah yang digunakan dalam fungsi Feisel. Cara kerja proses tersebut adalah sebagai berikut:

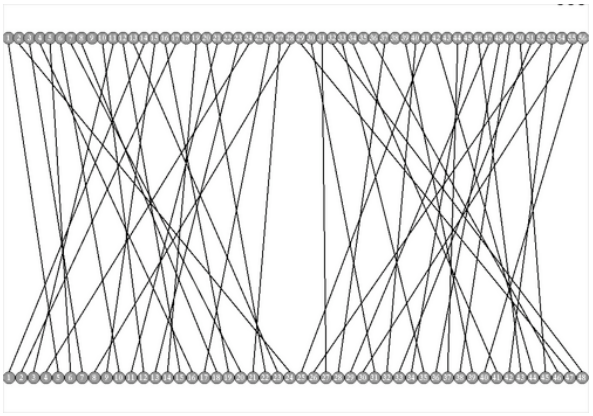
1. Mula-mula, dari 64 bit masukan dipilih 56 bit melalui PC-1 (*Permutated Choice-1*). Sisanya dibuang atau dijadikan parity check.
2. Blok 56 bit tersebut dibagi dua menjadi 28 bit dan diproses sendiri.
3. Pada tahap berikutnya, kedua bagian dirotasi ke arah kiri sebanyak 1 atau 2 bit
4. Lalu, 24 bit dari masing-masing bagian dipilih berdasarkan PC-2 (*Permutated Choice-2*), sehingga setiap bit digunakan dalam 14 subkunci.



Gambar 2.5.1 Diagram proses IP



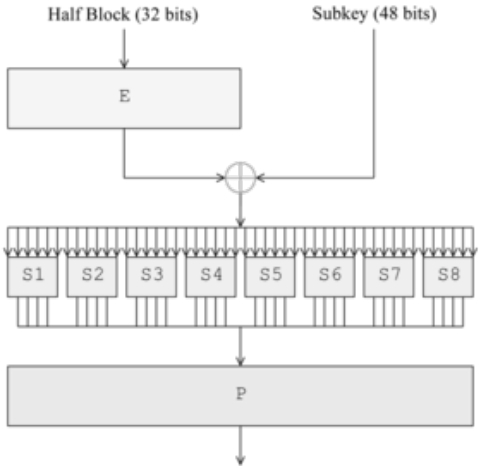
Gambar 2.5.2 Diagram proses PC-1 56-bit



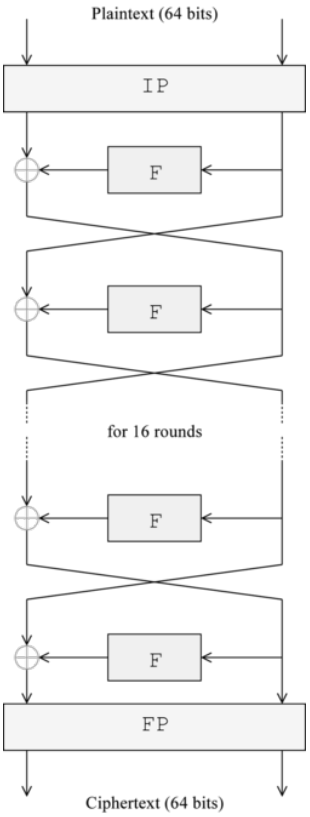
Gambar 2.5.3 Diagram proses PC-2

Proses enkripsi Triple DES (3DES) yang digunakan oleh EMV adalah penumpukan algoritma DES sebanyak tiga rangkap. Karena DES merupakan *block cipher* yang merupakan fungsi transformasi, maka keluaran proses tersebut dapat menjadi masukan proses berikutnya. Misal M adalah blok yang dienkripsi, maka blok tersebut akan menjalani proses $DES(k_3;DES(k_2;DES(k_1;M)))$. Proses 3DES memiliki key utama sepanjang 112 bit. Untuk memecahkannya, kompleksitas waktu algoritma yang dibutuhkan adalah 2^{32} plaintext yang diketaui, 2^{113} langkah, 2^{90} enkripsi DES satu tahap, and 2^{88} ruang pada memori. Dengan optimalisasi cara jumlah langkah (misal dengan *meet-in-the-middle attack*) dapat dikurangi menjadi 2^{90}. Untuk kemampuan prosesor saat ini, algoritma 3DES cukup aman sampai beberapa tahun ke depan.

Walaupun algoritma 3DES termasuk pelan dan membutuhkan kemampuan prosesor yang lebih besar dibandingkan algoritma yang baru seperti AES (*Advanced Encryption Standard*) 128-bit, namun EMV tetap memakai algoritma ini karena implementasi yang sudah luas dan matang. 3DES juga dipakai karena dengan sedikit modifikasi tetap kompatibel dengan kode DES satu tahap biasa.



Gambar 2.5.4 Diagram fungsi Feistel



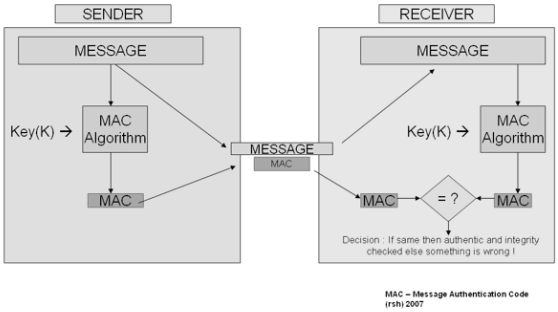
Gambar 2.5.5 Diagram algoritma DES

2.6 Autentifikasi

Karena dalam transaksi menggunakan kartu kredit nasabah tidak langsung melakukan pembayaran kepada *merchant*, perlu dilakukan proses autentifikasi yang bertujuan untuk mensahkan pembelian. Cara yang umum adalah dengan tanda tangan nasabah pada nota pembelian yang kemudian dicocokkan dengan tanda tangan pada bagian belakang kartu. Sayangnya, tanda tangan dapat ditiru dan memungkinkan terjadinya penipuan. Untuk mengatasi hal ini, penerbit kartu kredit menggunakan pengamanan tambahan, yaitu PIN (*Personal Identification Number*). PIN adalah kode angka (sepanjang 4 atau 6 digit) yang digunakan untuk mengecek apakah pengguna adalah pemilik sah kartu kredit tersebut. Umumnya dalam proses transaksi pengguna diberi kesempatan 3 kali untuk memasukkan PIN yang benar. Untuk 4 digit angka, kombinasi yang mungkin berjumlah 10.000 buah (0000-9999) sehingga peluang untuk menebak nomor yang benar cukup kecil (P(A) = 1/3333).

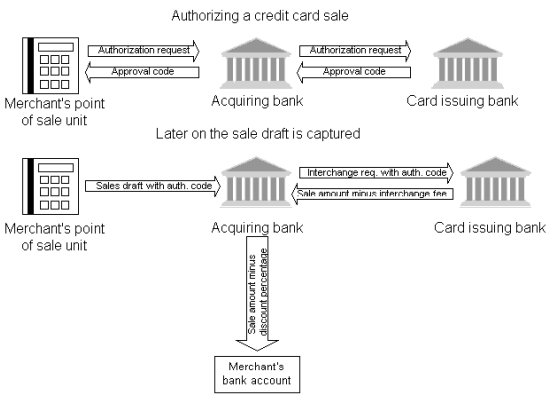
Walaupun begitu, nomor PIN tidak aman karena membutuhkan input manual dari pengguna. Penipu yang memiliki nomor kartu kredit dan data pribadi dapat melihat saat pemilik yang asli melakukan input PIN, sehingga kartu kredit dapat disalahgunakan. Kartu kredit yang hanya memiliki pita magnetik dapat dengan mudah digandakan dan dipalsukan.

Untuk menangani masalah ini, digunakanlah PIN dan chip EMV. Saat transaksi di terminal PoS dilakukan, nasabah memasukkan nomor PIN dan *merchant* memasukkan nomor kartu kredit serta jumlah transaksi. Jika nomor kartu lolos validasi, mesin PoS akan mengirimkan data-data secara terenkripsi tersebut ke server pengautentikasi di bank penerbit. Untuk menambah keamanan, data tersebut dikirim bersama MAC (*Message Authentication Code*). MAC dapat berupa blok terenkripsi ataupun sebuah fungsi hash. Sesampainya di server, data akan didekripsi dan jika sesuai dengan database pada server proses transaksi akan dilakukan. Arsitektur yang umum digunakan adalah XFS (*Extention for Financial Services*) dan PKCS#11 (*Public-key Cryptographic Standard*) sebagai API (*Application Programming Interface*) untuk proses enkripsi-dekripsi.



Gambar 2.6.1 Diagram MAC

Dalam perkembangannya, transaksi keuangan dapat dilakukan melalui internet dan dapat menggunakan kartu kredit untuk sarana pembayarannya. Transaksi kartu kredit melalui internet kurang aman karena tidak diperlukan kartu secara fisik, melainkan hanya nomor kartu dan data pribadi nasabah. Untuk mengantisipasi terjadinya penyalahgunaan, penerbit kartu kredit mencetak CSC (*Card Security Code*) atau CVV (*Card Verification Code*) pada kartu. Ada 2 tipe kode CSC, satu termasuk dalam data pada pita magnetis (CVV1), sedangkan satunya lagi hanya dicetak di bagian luar kartu (CVV2). Saat transaksi melalui internet dilakukan, server akan meminta CVV2. Karena kode 3 digit tersebut hanya terdapat di bagian fisik kartu, penyalahgunaan menjadi dipersulit dan dapat dihindarkan.



Gambar 2.6.2 Diagram pemrosesan transaksi

3. KESIMPULAN

Dalam transaksi menggunakan kartu kredit, terdapat beberapa metode pengamanan transaksi dari penyalahgunaan, yaitu:

1. Penggunaan PIN sebagai pengganti tanda tangan
2. Validasi nomor kartu kredit dengan algoritma Luhn
3. Penggunaan chip EMV dengan enkripsi 3DES
4. Verifikasi CSC pada kartu untuk transaksi melalui internet

Dengan metode-metode tersebut, penggunaan kartu kredit sebagai cara pembayaran menjadi lebih aman. Seiring dengan berkembangnya bidang keamanan finansial, di masa yang akan datang cara pengamanan transaksi akan semakin praktis dan efisien.

DAFTAR REFERENSI

[1]. Deo et al. *Authentication System and Method for Smart Card Transaction, U.S. Patent 5,721,781*. Feb 24, 1988.

[2]. Ehrsam et al. *Product Block Cipher System for Data Security, U.S. Patent 3,962,539*. Feb. 24, 1975

[3]. Haymann, Frank V. *Preventing Unauthorized Use of a Credit Card, U.S. Patent 5,365,046*. Nov 15, 1994.

[4]. International Standard Organization (ISO). *ISO/IEC 7810:2003*. (http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=31432&ICS1=35&ICS2=240&ICS3=15). Tanggal akses: 1 Januari 2008.

[5]. Lucks, Stefan .*Attacking Triple Encryption.* (http://th.informatik.uni-mannheim.de/People/Lucks/papers/pdf/3des.pdf.gz). Tanggal akses: 1 Januari 2008.

[6]. National Institute of Standards and Technology (NIST). *Data Encryption Standard (DES)*. (http://www.itl.nist.gov/fipspubs/fip46-2.htm). Tanggal akses: 30 Desember 2007.

[7]. Reeder, Kenneth Rodney. *Point of Sale Method and System, U.S. Patent 6,014,636*. Jan 11, 2000.