VOL. 16 NO. 3 SEPTEMBER 2015

ISSN : 1411-3201

Jurnal Ilmiah DASI

DATA MANAJEMEN DAN TEKNOLOGI INFORMASI

STMIK AMIKOM
YOGYAKARTA

VOL. 16 NO. 3 SEPTEMBER 2015
JURNAL ILMIAH
Data Manajemen Dan Teknologi Informasi

Terbit empat kali setahun pada bulan Maret, Juni, September dan Desember berisi artikel hasil penelitian dan kajian analitis kritis di dalam bidang manajemen informatika dan teknologi informatika. ISSN 1411-3201, diterbitkan pertama kali pada tahun 2000.

PENANGGUNG JAWAB :
Ketua STMIK AMIKOM Yogyakarta, Prof. Dr. M. Suyanto, M.M.

ALAMAT PENYUNTING & TATA USAHA
STMIK AMIKOM Yogyakarta, Jl. Ring Road Utara Condong Catur Yogyakarta, Telp. (0274) 884201 Fax. (0274) 884208, Email : jurnal@amikom.ac.id

BERLANGGANAN
Langganan dapat dilakukan dengan pemesanan untuk minimal 4 edisi (1 tahun) pulau jawa Rp. 50.000 x 4 = Rp. 200.000,00 untuk luar jawa ditambah ongkos kirim.

DAFTAR ISI

ACCESS CONTROL SMART CARD UNTUK PENGELOLAAN SISTEM DATABASE ABSENSI PADA STMIK AMIKOM YOGYAKARTA

Ali Mustopa

Teknik Informatika STMIK AMIKOM Yogyakarta
email: ali.m@amikom.ac.id

Abstract

Integrity and consistency management of a database system will be achieved, if any relation database for every transaction made is atomic and can recover any failures. Every transaction to access the data must have the authority / personal identification as a tool for validation of the transaction.
With the advancement of technology chipcard / Smart Card for the system is on-line Authonzation then any endorsement of transactions connected with the central database management system, and form validation is an electronic transaction in accordance with the value of the PIN (Personal Identification Number) specific. Applications chipcard / Smart Card in the system On-Line Authorization for access control quadrant.

Data Processing Facilities and Office Entry, developed by STMIK AMIKOM YOGYAKARTA in making Attendance System. Use of Smart Card Data attendance system aims to monitor the history and status of personnel attendance data. Use of Smart Cards On-Line to the database will facilitate the design of control systems and also allows control of the entry and exit of employees in a corporate environment.

***Keywords*:**
Information Systems, SmartCard, PIN, Database Systems

Pendahuluan

Salah satu aplikasi kemajuan teknologi *microcontroller* adalah *teknologi chipcard*, yaitu merealisasikan ditanam ke data kartu plastik yang digunakan sebagai transaksi keuangan. Keamanan, kehandalan dan kederhanaan dalam menyelesaikan permasalahan manipulasi dan penyimpanan data merupakan arti penting dari jenis kartu ini sehingga sering disebut kartu pintar *(smart card).* Pengembangan aplikasi card lain digunakan untuk pembayaran telepon kartu, akses televisi dengan satelit, penyimpanan data kesehatan, daftar simpan personil militer, akses kontrol identitas personil dan lain-lain.

Pada transaksi elektronik menggunakan kartu magnetik maupun yang dilengkapi PIN *(Personal Identification Number),* misalnya chipcard, unsur terpentingnya adalah bagaimana mengetahui keabsahan kartu atau keabsahan pemakai kartu sendiri. Sistem yang mengesahkan kartu pada transaksi elektronik disebut *authorization system.* Sistem pengesahan dibedakan menjadi dua macam yaitu *On-Line Authorization System* dan *Off-Line Anthorization System.*

1. On-Line Authorization System Pada sistem pengesahan *on-line* semua tempat transaksi *(point of sale)* dihubungkan dengan sistem pengendali pusat, yang berisi semua data *user*. Saat user menggunakan kartu, pembaca kartu *(card reader)* menghubungi pengendali pusat untuk mengecek keabsahan kartu tersebut dan mengetahui kondisi *account* pelanggan.

Dua persoalan sistem on-line adalah komunikasi data dan basis data. Untuk menghubungkan semua tempat transaksi ke pengendali pusat diperlukan jaringan data yang andal aman dan tepat. Diperlukan suatu basis data besar untuk menampung semua data Pelanggan. Sistem basis data yang digunakan dapat berupa basis data terpusat (*centralized database*) dimana setiap tran-saksi baik dekat maupun jauh harus berhubungan dengan basis data pusat, atau basis data terdistribusi *(Distributed database)* yang digunakan pada kota-kota besar tertentu, dan jalur komunikasi andal antar basis data untuk meng-*update* informasi transaksi yang ada.

2. Off-Line Authorization System Pada sistem pengesahan *off-line*, otorisasi transaksi dilakukan secara terpisah, yaitu di tempat transaksi. Sistem *off-line* setiap kartu sudah memiliki kode-kode yang dapat digunakan untuk mengetahui keabsahan kartu tersebut, termasuk kode PIN-nya dan *unit value* yang berfungsi sebagai alat tukar dalam transaksi tersebut.

Sistem off-line authorization memiliki beberapa keuntungan antara lain :

a. Tidak diperlukan jaringan data
b. Tidak diperlukan basis data
c. Biaya pengesahan nol, tidak ada biaya komunikasi

Kelemahan sistem *off-line* adalah ketergantung-an sistem pada keandalan kartu, karena semua data terdapat dalam kartu.

Pengembangan teknologi *smart card* di STMIK AMIKOM Yogyakarta yang diterapkan meliputi desain dan pembuatan *smart card*, termasuk *card reader* dan perangkat pendukung, seperti sistem *microprosesor*, perangkat lunak aplikasi *smart card*, dan perangkat lunak untuk suatu aplikasi sistem tertentu.

Aplikasi *smart card* di lingkungm perusahaan dilakukan oleh Bagian Rekayasa dari Divisi Jaringan, dengan mengintegrasikan pemakaian *Personal Computer* (PC), *card reader* dan *smart card* untuk menangani dan mengendalikan masuk/ keluarnya karyawan dalam lingkungan perusahaan, dan dikenal sebagai 'Sistem Absensi'. Langkah awalnya adalah mengganti Sistem Absensi lama yang menggunakan kartu magnetik dengan *chip-chard/smart card*. Keunggulan *smart card* untuk Sistem Absensi adalah keandalannya yang tinggi (tidak mudah rusak), dapat dimuati informasi dengan kapasitas lebih besar, dan mempunyai kea-manan tinggi karena adanya kode PIN *(Personal Identificaton Number)* bagi pemegangnya.

Jenis kartu yang digunakan Sistem Absensi adalah *'Protected Memory Card'* dengan tingkat keamanan cukup memadai. Jenis kartu ini mempunyai keunggulan karakteristik yaitu perubahan dan penghapusan data kontrol dilakukan melalui password. Selain berisi nomor identitas (NIP/NIK), kartu ini juga berisi data identifikasi untuk pengesahan kartu, yaitu kode pabrik kartu, versi chip, tipe chip, nomer urut, kode PIN,dan lain-lain. Kode PIN adalah kode unik yang hanya diketahui oleh pemilik kartu.

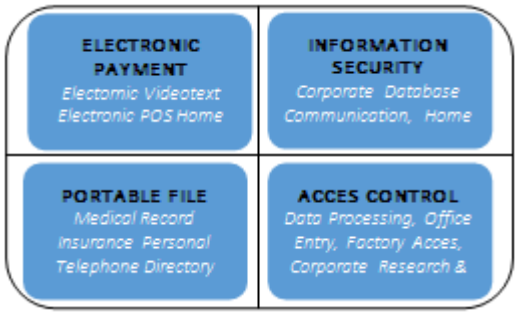
Tinjauan Pustaka

Aplikasi *Smart Card*

Smart card dapat digunakan dalam beberapa aplikasi, karena kemampuan *smart card* dapat diprogram berdasarkan ketersediaan kapasitas memori. Aplikasi *smart card* secara umum dapat dibagi menjadi 3 kategori yaitu untuk *data carrier* (pembawa data), *identification* (identifikasi) dan *financial* (keuangan).[1]

1. *Data carrier*
   Kartu yang digunakan harus mudah dalam pemakaiannya, sederhana dan aman untuk menyimpan informasi, misalnya data pemerikasaan kesehatan.
2. *Identification*
   Pemakaian kartu digunakan untuk pengaman yaitu mengidentifikasi pemegang ketika mengakses suatu hal khusus/pribadi *(privacy)*. Contoh akses komputer atau ijin masuk suatu pertandingan.
3. *Financial*
   Kartu digunakan untuk melakukan transaksi keuangan, contoh alat pertukaran universal pengganti keuangan *(cheque)*, ATK dan lain-lain.

Dari aplikasi umum dari *smart card* diatas dapat dipetakan dalam bentuk 4 kuadran yaitu *Electronic payment, Information system security, Access control dan Portable file.* Peta aplikasi smart card telah melingkupi aplikasi kartu magnetik (seperti *Prepaid services)*, dan juga mencakup beberapa aplikasi media penyimpanan lainnya seperti *encryption key*. Peta aplikasi *smart card* terlihat pada gambar 1 seperti pada gambar dibawah ini .



Gambar 1. Kuadran peta aplikasi smart card

Teknik Keamanan Transaksi pada Smart Card

Kerahasiaan dan autentifikasi data yang di-transmisikan dalam sistem komunikasi antara card reader dan komputer adalah kunci keamanan dan proteksi dari transaksi *smart card*. Bentuk proteksi bisa dilakukan secara *hardware* dan *software*. Proteksi *software* dilakukan berdasar akses kontrol data dan teknik *cryptography* (kriptografi) yang digunakan. Proteksi *hardware* menggunakan IC mikrokontroller selama pembuatannya.[3]

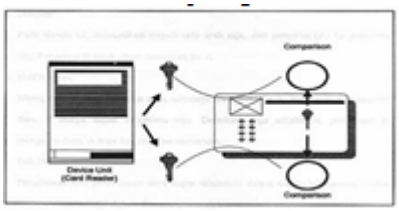
Tahapan sistem kriptografi atau *crypstosystem* yang mengubah pesan tak terlindungi atau *plaintext* menjadi pesan yang mengalami transfomasi atau *ciphertext* disebut *enkripsi*, dan proses sebaliknya untuk mendapatkan *plaintext* disebut *dekripsi.*

Dalam sistem kriptografi perubahan *plaintext* menjadi *cyphertext* diatur oleh *secret words* (kata sandi) atau *key* . Ada beberapa teknik dalam sistem kriptografi dan ini tergantung dari penentuan jenis key yang digunakan dalam proses enkripsi dan dekripsi yaitu :

1. Sistem kriptografi *symmetric*
   Digunakan satu kunci yang sama untuk meng-enkripsi dan mendekripsi suatu pesan, kunci tersebut harus dijaga kerahasiaannya dan sering disebut kunci rahasia. Contohnya adalah *Data Encryption Standar (DES), Fast Data Enkripsiment Algorithm (FEAL),* dan lain-lain.
2. Sistem kriptografi *asymmetric*
   Digunakan dua kunci yang berbeda. Satu kunci disebut *public key* (kunci umum), dan kunci lain disebut *private key* (kunci pribadi). Kunci umum digunakan untuk enkripsi, tidak di-rahasiakan dan kunci pribadi untuk dekripsi, di-rahasiakan. Contohnya adalah sekema Markle-Hellman, skema ElGamal, skema RSA (Rivest-Shamir-Edleman), dan lain-lain.

Sistem kriptografi DES/FEAL dikembangkan untuk *telecommand, communications* atau *data handling*. Proses enkripsi/dekripsi dalam *smart card* banyak menggunakan model DES yang *built-in* di kartu atau *card reader*. *Key* K digunakan untuk transaksi internal dan external kartu seperti mengeksekusi *command* atau akses data area, dan

dalam komunikasi sebagai autentifikasi penerimaan *message* (pesan) dari kartu ke *inteface device*. *Device* mengenkripsi dengan *key* K, dan mengirimkan *message* yang dienkripsi ke kartu. Kartu mendekripsi *message* dengan *key* K dan membandingkan dengan string tertentu. Jika keduanya sama maka transaksi selanjutnya bisa dilaksanakan. Alternatif lain kartu yang mengenkripsi *message*, dan *device* yang mendekripsi dan hasilnya dikembangkan ke kartu. Proses autentifikasi terlihat pada gambar 2.



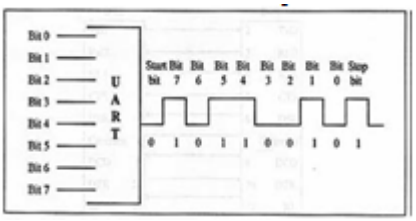
Gambar 2. Proses Autentifikasi Data Sistem Kriptografi Des/Feal Dengan Direct Key Yang Sama

Pengiriman Data Asinkron pada RS-232

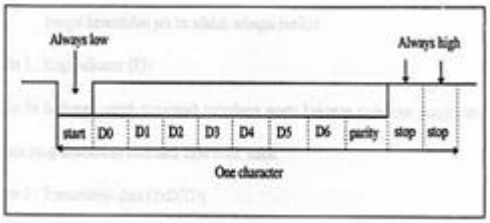
Sistem Pengiriman asinkron dan sinkron. data dalam komunikasi serial ada dua macam yaitu Pengiriman secara sinkron lebih kompleks dan sangat sulit untuk percobaan sederhana, karena kedua titik komunikasi harus selalu dibuat sinkron. Sinkronisasi pada penerimaan data RS-232 dilakukan oleh IC 8250/16450 UART.

(Universal Asyncronous Receiver/Transmitter). 8250 UART merupakan standar setiap port untuk IBM PC, sedangkan 16450 adalah standar untuk EBM PC/AT. UART kompatibel untuk tingkat hardware dan software.[2] Pada UART 8250 prinsipnya adalah mengambil 8 bit data secara parallel dalam data byte yang dikonversikan menjadi suatu aliran 8 bit tunggal, dengan ditambah start dan stop bit seperti pada gambar 2.4 UART juga mengontrol status lines yang didefinisikan oleh standar RS-232. Programmer dalam mengatur UART 8250 melalui I/O *port serial*, maka tiap-tiap port harus dispesifikasikan sebagai *offset* alamat mutlak dari *port.*

Bit-bit serial asinkron terdiri atas 1 *start* bit (selalu *low*), 6 sampai dengan 8 bit data, 1 bit paritas, 1 atau 2 stop bit (selalu *high*). Pada saat tidak ada data dikirim, kondisi saluran transmisi selalu high. Kondisi bit paritas ditentukan oleh sistem paritas yang digunakan (ganjil atau genap). Agar tidak terjadi kesalahan interpretasi antara pengirim dan penerima maka hendaknya sistem paritas yang digunakan disetujui bersama, paritas genap atau ganjil. Faktor lain yang perlu diper-hatikan dalam pengiriman data serial asinkron adalah kecepatan pengiriman. Besaran kecepatan penerimaan data serial adalah *bit per seconds* (bps) dan biasa disebut *baud rate* atau *character/seconds* (cps).



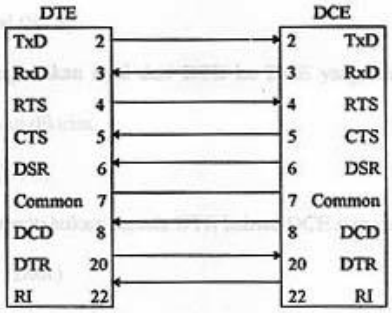
Gambar 3. processing uart bentuk format data pada komunikasi asinkron adalah seperti pada gambar 4.



Gambar 4. Format Pengiriman Data Serial Asinkron

RS-232 dan Fungsi Pin-nya

Standar RS-232 mendefinisikaan 9 jalur *control lines* digunakan sebagai koneksi antara DTE *(Data Terminal Equipment)* dan DCE *(Data Communications Equipment).* Jumlah pin yang digunakan pada konektor 25 pin cukup 9 pin seperti pada gambar 5 tetapi yang sifatnya sederhana cukup 3 pin yaitu pengirim, penerima dan *ground.*



Gambar 5. Koneksi 9 Pin Dari Interface Rs-232 Pada Konektor 25 Pin

Fungsi kesembilan pin ini adalah sebagai berikut:

1. Pin 1 : *Ring indicator* (RI)
   Pin ini berfungsi untuk mencegah terjadinya suatu kejuatan pada saat pengiriman data yang disebabkan oleh catu daya tidak stabil.
2. Pin 2: *Transmitted data* (TxD/TD)
   Sinyal dalam rangkaian ini dibangkitkan oleh DTE untuk diberikan ke DCE lalu disalurkan melaui saluran komunikasi ke tujuan (remote station). Pin ini merupakan jalur pengiriman data dari DTE ke DCE.
3. Pin 3: *Received data* (RxD/RD)
   Sinnal ini data yang dibangkitkan oleh DCE dalam sinyal yang diterima dari remote station,

lalu sinyal disalurkan ke DTE. Pin ini merupakan jalur pengiriman dari DCE ke DTE.

4. Pin 4: *Request to send* (RTS)
   Pin ini berfungsi menngirimkan sinyal dan DTE ke DCE yang menyatakan bahwa akan ada data yang akan dikirim.
5. Pin 5: *Clear to send* (CTS)
   Pin ini berfungsi memberitahukan kepada DTE bahwa DCE siap menerima data.
6. Pin 6: *Data set ready* (DSR)
   Pin ini menyatakan status modem (data set) lokal yang tersambung pada DCE.
7. Pin 7: *Signal ground* (Common)
   Berguna sebagai semua tegangan yang berinterferensi.
8. 8. Pin 8: *Data carrier detect* (DCD)
   Berfungsi untuk mendeteksi apakah DTE boleh atau tidak untuk menerima data
9. Pin 9: *Data terminal ready* (DTR)

Berguna memberitahukan kepada DCE bahwa DTE telah siap. Secara ringkas *procedure* yang harus dilewati oleh DTE adalah :

Sinyal DSR diterima yang menunjukkan bahwa modem telah mendapatkan daya yang dan siap untuk beroperasi.

1. Sinyal DSR diberikan untuk menghidupkan transmitter modem atau menghentikannya.
2. Sinyal CTS yang diterima menunjukkan bahwa modem siap menerima data.
3. Kesiapan modem lawan beroperasi dan kwalitasnya memenuhi syarat ditunjukkan oleh Carrier Detect.
4. Data yang akan dikirimkan melalui saluran komunikasi diberikan pada rangkaian Transmit Data.
5. Data yang diterima melalui saluran komunikasi dapat diambil dari rangkaian Received data.

Metode Penelitian

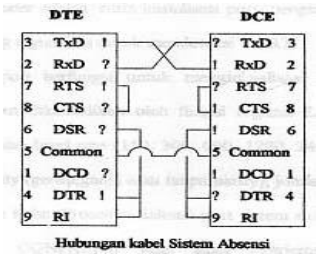
Perancangan Interface RS-232 untuk Komunikasi dengan Smart Card lewat WRU

Sistem Pengkabelan

Spesifikasi kabel dalam komunikasi sistem absensi tidak menggunakan logika standar RS-232 secara lengkap, tetapi hanya menggunakan tiga kaki yaitu TxD *(Transmit Data)*, RxD *(Receive Data)* dan *Signal Ground (common)*. Perancangan spe-sifikasi kabel dengan 3 kaki untuk menyeder-hanakan pengendali *interface* RS-232 dengan UART yang digunakan oleh komputer mikro.

Pada DTE tidak menyertakan tiga *input* terbuka aktif CTS, DSR dan DCD, tetapi mereka dililitkan dengan pasangan handshakingnya. Su-sunan *interface* DTE ini mengakibatkan selama ada tegangan input aktif CTS, DSR dan DCD akan diinterpretasikan sudah dalam logika aktif selama pengiriman dan penerimaan data. Pada beberapa aplikasi logika pengendali tidak dipentingkan, masalah input aktif terbuka diselesaikan dengan hanya menyusun tiga kawat (TxD, RxD, dan signal ground) dengan pelompat *pull-up* pada salah satu pin (contoh melilitkan pin CTS dan RTS) pada DTE. *Pull-up* adalah akal untuk menjamin keadaan suatu logika sehingga pemakaian jalur bisa menjaga *operasi* dengan lancar pada *interface*.

Spesifikasi untuk 9 kabel yang digunakan pada sistem absensi terlihat pada gambar 6.



Gambar 6. Hubungan kabel Sistem Absensi konekor 9 pin

Spesifikasi Sistem Absensi memeperlihatkan jalur data (pin 2 dan 3) dan *signal ground* yang menghubungkan ke *interface* lainnya. Signal *hand-shag* (RTS, CTS, DSR, DCD, dan RI) dalam sistem ini tidak digunakan, sehingga tidak perlu pull-up atau *push-down* karena signal-signal ini bisa diabaikan *(dont It care).* Selama *software* tidak melakukan pengecekan terhadap signal-signal ini maka signal-signal tersebut tidak perlu . Selain itu pada sistem absensi menggunakan mode komunikasi *asynchronous* jadi tidak diperlukan L* sinkronisasasi mengingat area aplikasi sistem absensi cukup luas yaitu meliputi seluruh bagian-bagian di PT. LEN INDUSTRI maka diperlukan sasi dalam sistem pengkabelan *(wiring).*

Modul Header

Untuk berfungsinya lapis jalinan data maka dipedukan suatu modul header yang terdiri dari rutin-rutin yang siap dipanggil oleh lapisan jalinan data. Modul header ini berisi konstanta-konstanta yang mewaldii nilai-niw tertentu. Rutin-rutin yang terdapat pada modul header adalah rutin inisialisasi port, pengiriman dan penerimaan data, dan parameter yang digunakan untuk mendeteksi UART. inisialissisi port berfungsi untuk menginisialisasi UART pada format tertentu. Format tersebut dikendalikan oleh fungsi register Line Control Register dengan menyatakan jumlah baud rate (150, 300, 600, 1200, 2400, 4800, atau 9600) bit per second, jenis parity (genap,ganjil atau tanpa parity, jumlah stop bh (1 atau 2), dan panjang data (7 atau 8) bit. Proses inisialisasi port sistern absensi absensi dilakukan secara ekstemal melalui file CONFIG.INI yang akan diterjemahkan dalam file ENKRFEAL.TPU sebagai inisialisasi UART, melalui fungsi interrupt BIOS 14 hexa (serial I/0).

Selain itu dengan diaktifkannya *port Line Control Register* pada bit 7 DLAB *(Divisor Latch Aems Bit)* maka akan digunakan oleh *Transmit/Receive Buffer Register* dan I*nterrupt Enable Register* (menjadi pasangm MSB dan LSB) untuk menentukan kecepatan transmisi.. Rutin pengirim pada modul *header* akan mengirim per karakter atau per byte secara berurutan. Pengiriman dila-kukan melalui *port Transmit Buffer Register* (TBR) ke *Card Reader* (WRU), karena bentuk Pengiriman dan penerimaan data secara *polling* maka sebelum mengirim data terlebih dahulu dicek bit 1 di *Line Status Register* apakah aktif atau tidak (port[LSR]=$0l). Untuk rutin permintaan data Melalui *port Receive Buffer Register* (RBR), maka dicek terlebih dahulu apakah RBR sudah kosong atau belum dengan melihat bit ke 6 dan 5 di Line Status Repster apakah aktif atau tidak. Secara praktis sebelum data RBR di reset dengan memberin nilai port [LSR]=$60.

Command dan Protokol untuk Reader

Protokol merupakan digunakan sebagai ketetapan yang harus disepakati Dalam komunikasi antar terminal komputer dan card reader,untuk menjamin bahwa data yang dikirim di inter-prestasikan dengan benar. Protokol pengontrol jalinan data mendefinisikan:

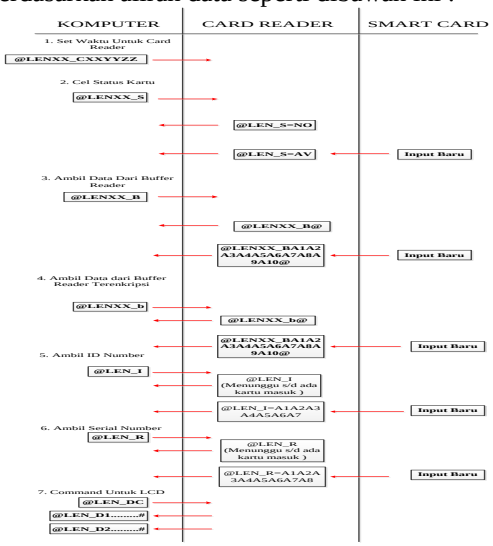
1. Format data yang dipertukarkan, jumlah bit/byte, dan skema pengkodean yang digunakan
2. Jenis dan urutan pesan agar dicapai keutuhan infomasi bak dari kesalahan dan dupkai

Protocol type selection (PTS) yang dipergunakan dalam sistem absensi menggunakan tipe protocol yang berupa urutan commad-commad yang telah didefenisikan terlebih dahulu. command-comand yang dikirim melalui serial port komunikasi berupa karakter string. Setiap command *acknowl-edgement* berupa karakter yang baru dikirim pada saat itu juga. Definisi protocol untuk *Smart Card* sistein abwm pada tabel 1 dibawah ini.

Tabel 1. Protocol untuk smart card Sistem Absensi

Command untuk mendapatkan ID Number	**"@LEN_I"**
Respons berupa ID Number	"@Len_I=A1A2A3A4A5A6A7" (A1 … A7 adalah 7 angka ID (0-9,A-F))
Command untuk mendapatkan serial number	"@LEN_R"
Respon berupa serial number	"@LEN_R=A1A2A3A4A5A6A7A8" (A1 … A8 adalah 8 angka numerik)
Command untuk memeriksa status kartu	"@LEN_S"
Rspon berupa status kartu	"@LEN_S=AV" Kartu ada (Available) "@LEN_S=NO" Kartu tak ada (not.av)
Command untuk baca keypad	"@LEN_K4" (Contoh 4 adalah jumlah input angka)
Command untuk men-set waktu mulai sistem beroperasi	"@LENXX_CXXYYZZ" XX=Jam, YY=Menit, ZZ=Detik" (XX adalah 2 angka numerik untuk ID setiap Card Reader
Command untuk mengambil data dari buffer Card Reader	"@LENXX_B" (XX adalah 2 angka numerik untuk ID setiap Card Reader"
Respon berupa data yang ada di buffer card Reader	"@LENXX_BA1A2A3A4A5A6A7A8A9A10" (A1 … A10 adalah 10 angka numerik)
Command untuk mengambil data dari Card Reader keadaan terenkripsi	"@LENXX_b" (XX adalah 2 angka numerik untuk ID setiap Card Reader"
Respon berupa data yang ada di buffer Card Reader Keadaan Terenkripsi	"@LENXX_BA1A2A3A4A5A6A7A8A9A10" (A1 … A10 adalah 10 angka numerik)
Command Untuk LCD :	"@LEN_DC"
Clear Display	"@LEN_D1 ----------#"
Tulis baris 1 (max, 16 karakter)	(…. Adalah karakter yang akan ditampilkan, dikahiri karakter "#" sebagai penutup)
Tulis baris 2(max, 16 karakter)	"@LEN_D2----------#"

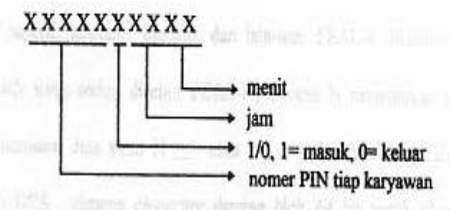
Instruksi yang digunakn pada sistem absensi berdasarkan *Command-commnad* diatas, ada beberapa instruksi yang dilakukan dari terminal komputer ke *card reader* atau dari *card reader* ke kartu berdasarkan aliran data seperti dibawah ini :



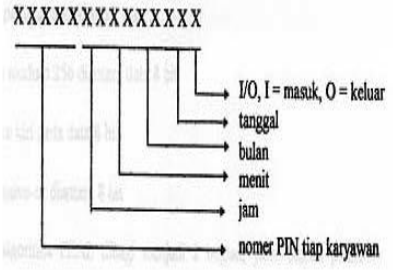
Dari sejumlah *command* diatas tidak semuanya harus dipakai , dalam Sistem Absensi, sebab terminal komputer akan selalu mengirim N command @LENXX_B ke sejumlah N *card reader* pada setiap selang 1 menit. Tiap *card reder* akan mendeteksi ID-nya berdasar nilai XX, bila command diterima oleh suatu card reader dan saat itu telah ada sejumah kartu telah dimasukan maka card reader mengirimkan seluruh isi buffer berupa @LENXX_BA11..Al 10A21..A3 I..A3 10 .... Anl..AnIO dimana n jumlah kartu yang telah dmusukkan ke *card reader*. Bentuk enkripsi dilakukan saat terminal komputer mengirim command @LENXX_B ke card redaer dan card rteader akan mengirimkan seluruh isi buffer berupa @LEXX_BA11..A110A21..A21OA31..A310 .... An1..An10 dimana data Ail..Ai8, i=l..n, terenkripsi dengan metode enkripsi FEAL. Data hasil enkripsi ini yang harus didekripsi oleh terminal komputer untuk operasi-operasi berikutnya.

Operasi File

Sebelum melakukan operasi file, bila data yang dikirim oleh masing-masing card reader dalam keadaan terenkripsi maka terminal komputer harus mendeskripsi terlebih dahulu sehingga didapatkan deretan 10 karakter numerik, contoh 3452100815,format dari 10 karakter numerik itu seperti berikut :



Format data 10 karakter numerik diatas sebelum dimasukan ke dalam file maka harus disesuaikan terlebih dahulu dengan /skema database sistem absensi. Bentuk format file adalah tiap baris berisi urutan 14 karakter alpha-numerik yang menyatakan status keluar-masuk setiap karyawan tiap hari, contoh 34521081527120, format dari 14 karakter alpha-numerik itu seperti berikut :



Gambar 7. Ilustrasi Format 14 Karakter

Enkripsi Data FEAL

Sistern kriptografi dengan metode FEAL digunakan untuk pengolahan data kecepatan tinggi, pada awalnya dikembangkan oleh *Nippon Telegraph and Telephone* (NTT) Jepang tahun 1987. Pada proses pengembangannya FEAL ada beberapa tingkat, yaitu FEAL-4 tingkat 4, FEAL-8 tingkat 8. FEAL-8 lebih banyak diimplementasikan seperti *facsimile*, modem, dan lain-lain. FEAL-8 dikembangkan menjad,i *FEAL-family* yang sering disebut FEAL-N, d N menentukan round *number* untuk randomisasi data yaitu N > 4 atau 2x , x > 2. Algoritma FEAL-N Kompatibel dengan DES, dimana *chipering* dengan blok 64 bit untuk *plaintext, ciphertext*, dan kunci rahasia.

Ada beberapa pendapat dalam perancangan FEAL, yaitusebagai berikut :

1. Algotitma FEAL dirancang denganmemenuhi standar algoritma DES
2. Algoritma FEAL dirancangdengan mempermudah kriteria untuk evaluasi kemapuan randomisasi data dalam proses enchiper.
3. Ada beberapa operasi enchiper dalam algoritma FEAL untuk mengurangi langkah-langkah dalam pemrogaman dengan cara :
   a. Penambahan modulo 256 diantara data 8 bit
   b. Rotasi 2 bit ke kiri pada data 8 bit
   c. Operasi exclusiveor diantara 8 bit

Pada algoritma FEAL dibagi Menjadi 2 bagian, yaitubagian pemroseskunci dan randomisasi data. Pemrosesan kunci digunakan untuk men-campur masing-masing bit pada kunci 64 bit dan menghasilkan 256 bit kunci pengembangan *(extended key)*. Randomisasi data digunakan untuk mencampur masukan plaintext dengan kunci pengembangan dan menghasilkan *chipertext*. Proses Pada randomisasi data ada 3 bagian :

1. Proses pendahuluan : p (D $\oplus$ K_1 ® X

2. Proses internal : mencampur X denganbagian dari kunci pengembangan
3. Proses penutup: X. ⊕ K2 ® C
   dimana P = *plaintext*, X = data internal C = *chipertext* ,K_1 dan K_2 = kunci pengembangan.

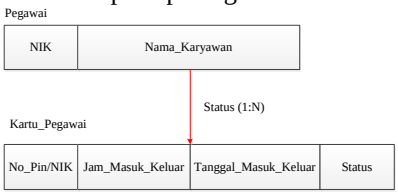
Ada beberapa algoritma yang digunakan dalam perancangan metode enkripsi FEAL :

1. Algoritma Enchiper
   Plaintext P dibagi menjadi Lo dan Ro. dengan panjang yang sama (32 bit)
   P = (L_o, R_o.)
   Pertama (L_o , R_o) = (L_o, R_o) ⊕ (K_N,K_{N-1},K_{N+2},K_{N+3})
   Berikutnya (L_o, R_o) = (L_o, R_o) ⊕ (D (F , L_o)
   Selanjutnya dihitung persamaan berikut untuk r = 1 sampai r = N dengan iterasi
   $R_r = L_{T-1} \oplus f(R_{r-1},K_{r-1})$
   $L_T = R_{r-1}$
   Hasil iterasi adalah (L_T , R_r). Keluaran dari akhir dari proses iterasi adalah (L_N,R_N) diubah menjadi (R_N L_N). Selanjutnya dihitung (R_N, L_N).= (R_N ,L_N) ⊕ (K_{N+4},K_{N+5},K_{N+6},K_{N+7}) Chipertext = (R_N, L_N)
2. Algoritma Dechiper
   Chipertext (R_N,L_N) dipisah menjadi R_N dan L_N dengan panjang sama. Peertama (R_N, L_N) = (R_N , L_N) ⊕ (K_{N+4},K_{N+5},K_{N+6},K_{N+7}). Beikutnya (R_N,L_N) = (R_N, L_N) ⊕ (F , R_N). Selanjutnya dihitung Persamaan berikut untuk r = 1 sampai r = N dengan iterasi
   $L_{T-1} = Rr \oplus f(L_t,K_{r-1})$
   $R_{r-1} = L_T$
   Hasil iterasi adalah (L_o, R_o). Berikutnya dihitung (L_o, R_o) = (L_o, R_o) ⊕ (F , L_o), Selanjutnya dihitung (L_o, R_o) = (L_o, R_o) ⊕ (K_N,K_{N+1},K_{N+2},K_{N+3}) Chipertext = (L_o, R_o)
3. Penjualan kunci (Extended key) K FEAL-N
   Panjang kunci K 64 bit dibagi menjadi 2 bagian yaitu A_o dan B_o masing-masing panjangnya 32 bit, K = (A_o,B_o). D_o = F . Panjadualan key K dihitung sebanyak K_I (I = 0 sampai N+7), N adalah round number, dengan banyak iterasi dari r =1 sampai (N/2 + 4), (N ≥4, N genap).
   $D_r = A_{r-1}$
   $A_r = B_{r-l}$
   $B_r=f_K$ **(**a **,**b)=$f_K(B_{r-1} \oplus D_{r-1}l$,
   $K_{2(r-1)}$ =(Bro,Br1)
   $K_{2(r-1)+1}$ =(B_{r2},B_{r3})
   A_r, B_r , dan Dr adalah variabel pemabantu, Br = (B_{r0},B_{r1},B_{r2},B_{r3}), panjang B_{r0} B_{r3} masing-masing 8 bit.
   $K_{2(r-1)}$: 16 bit dari Br disebelah kiri $K_{2(r-1)+1}$: 16 bit dari B, sebelah kanan
4. Fungsi f
   Fungsi f (a , b) adalah notasi untuk fungsi dengan variabel a dan b dengan panjang 8 bit. a = (a_0, a_1, a_2, a_3), b = (b_0, b_1, b_2, b_3**).** Fungsi f = (f_o, f_l, f_2 ,f_3) dihitung dengn rumusan sebagai berikut :
   $f_1' = a_1 \oplus b_0$ $f_2' = a_2 \oplus b_1$
   $f_1 = f_1' \oplus a_0$ $f_2' = f_2 \oplus a_3$
   $f_l = S_1 (f_l'', f_2'')$ $f_2 = S_2 (f_2'', f_1'')$
   $f_0 = S_o (a_o, f_1)$ $f_3 = S_1 (a_{o3} f_2)$
   $Y=S_o(X_1, X_2)=R_{ot2}((X_1+X_2) \bmod 2.56)$, $Y=S_1(X_1, X_2)=R_{ot2}((X_1+X_{2+1}) \bmod$
   Y: output 8 bit, X_1,X_2: input 8 bit
   Rot2(Y) : rotasi 2 bit ke kiri pada data Y
5. Fungsi fK
   Masukan untuk fungsi f_K , a dan b , dipisah mejadi 4 buah blok 8 bit f_K (a , b) Nilai f_K= (f_{K0}, f_{K1},f_{K2}, f_{K3}) dirumukan sebagai berikut :
   $F_{K1}' = a_1 \oplus a_0$ $F_{K2}' = a_2 \oplus a_3$
   $f_{Kl} = S_1 (f_{K1}', (f_{K2}' \oplus b_0))$ $f_{Kl2} = S_{10}(f_{K12}, (f_{K21} \oplus b_{01})$
   $f_{KO} = S_O (a_0,(f_{K1} \oplus b_2))$ $f_{KO3} = S_1 (a_3,(f_{K2} \oplus b_3))$

Database Absensi

Database absensi, digunakan dan ditempatkan dalam salah satu server yang ada di PT. LEN INDUSTRI. Pada Penelitian ini, sistem basis data yang telah ada di untuk menyusun sernua bentuk transaksi yang berupa keluaran file text dari salah satu workstation.

Perancangan proses dari relasi berdasarkan record yang digunakan untuk basis data Sistem Absensi adalah seperti pada gambar 8.



Gambar 8. Relasi Record-Record Di Sistem Absensi

Klasifikasi *relationship* antara tabel Pegawai dan tabel Kartu Pegawai adalah one to many (I:N), sebab seorang karyawan dalam setiap hari bisa masuk keluar kantor lebih dari satu kaii. Struktur relasi diatas menujukkan bahwa Pegawai sebagai *owner* dan Kartu Pegawai sebagai *members*, sehingga Pegawai akan menurunkan kepada Kartu_Pegawai.

Skema database pada sistern absensi seperti pada tabel 2 dan tabel 3.

Tabel 2. Skema Pegawai

No	Nama Field	Tipe Data	Panjang Data (byte)	Keterangan
1	NIK	Numerik	5	Field NIK Menyatakan Nomor Induk Karyawan, dan digunakan sebagai kunci dari record Pegawai
2	Nama	Karakter	25	Field Nama Meyatakan nama setiap karyawan di

STMIK AMIKOM Yogyakarta

Tabel 3. Skema Kartu_Pegawai

No	Nama Field	Tipe Data	Panjang Data (byte)	Keterangan
1	No_PIN	Numerik	5	Field No_Pin Menyatakan nomor PIN dari Setiap kartu, dan digunakan sebagai kunci dari Record Kartu_Pegawai
2	Jam_Masuk_Keluar	Numerik	4	Field Jam_Masuk_Keluar menyatakan kondisi jam masuk/keluar karywan dari Lingkungan Perusahaan
3	Tanggal_Masuk_Keluar	Numerik	4	Field Tanggal_Masuk_Keluar meyatakan kondisi tanggal masuk/keluar karyawan dari lingkungan Perusahaan
4	Satus	Karakter	1	Field status meyatukan status masuk/keluar karyawan, dimana I = masuk, dan O = Keluar

Hasil dan Pembahasan

Spesifikasi Proses

Modifikasi yang dilakukan terjadi pada proses otentikasi, sehingga dalam Penelitian ini spesifikasi proses hanya dibuat untuk proses yang dimodifikasi. Pada pihak *Client* proses otentikasi dilakukan oleh proses 1.1.3. otentikasi server, sedangkan pada bagian server pada proses 2.3.3. otetitikasi Client.

Sedangkan proses yang lain tidak dituliskan karena merupakan fasilitas yang disediakan *compiler*.

```
Proses Otentikasi Server
{Mengirimkan pesan no 3}
{Menerima pesan no 2}
Kamus:
        pesan: string
        no_pesan : integer
        kuncisesi : sting
        Bongkar_pesan: Function(pesan:string) ??string
        {menghilangkan Confounder, menata data}
        uji : Function (pesan:string) ??boolean
        {membaca pesan dan mengambil nilai nonce1)
        Confound:Procedure (pesan: string)
        {Menyamarkan data dengan menambahkan confounder pada pesan}
        Krip : Function(pesan, kunci : string) ??string
        {mengacak pesan dengan memakai kunci}
        Kirim: Function (pesan, server :string) ??boolean
        {Mengirim pesan kepada server}
Algoritma:
        If (no_pesan = 2)
        Bongkar_pesan(pesan);
        If (uji(pesan) == notOK)
        Exit; {server tidak terpercaya}
        Else
        Pesan ??gettime() + getcmd() + rand(seed,no) + Authenticator;
        Confound(pesan);
        Krip(pesan,kuncisesi)
        Kirim(pesan, Server);
        Endif
        Endif
Proses Otentikasi Server
        {Mengirimkan pesan no 2}
        {Menerima pesan no 1,3}
Kamus
        pesan: string
        no_pesan: integer
        kuncisesi : string
        Bongkar_pesan : Function(pesan:string) ??string
        {menghilangkan Confounder, menata data }
        uji : Function (pesan:string) ??boolean
        {membaca. pesan dan mengambil nilai nonce}
        Confound: Procedure (pesan: string)
        {Menyamarkan data dengan menambahkan confounder pada pesan}
        Krip : Function(pesan,kunci : string) ??string
        {mengacak pesan dengan memakai kunci}
        Kirim: Function (pesan, server :string) ??boolean
        {Mengirim pesan kepada server}
Algoritma:
        If (no_pesan = 1)
        Pesan ??gettime() + getcmd() + rand(seed,no) + Authenticator;
        Confound(pesan);
        Krip(pesan,kuncisesi)
        Kirim(pesan, Server);
        Endif
        If (no_pesan = 3)
        Bongkar_pesan(pesan);
        If (uji(Authenticator) == notOK)
        Exit; {Client tidak, terpercaya}
        Else
        Exec(getcmd());
        Endif
        endif
        Endif
        Endif
Proses Otentikasi Server
        {Mengirimkan pesan no 2}
        {Menerima pesan no 1,3}
Kamus:
        pesan: string
        no_pesan: integer
        kuncisesi : string
        Bongkar_pesan : Function(pesan:string) ??string
        {menghilangkan Confounder, menata data }
        uji : Function (pesan:string) ??boolean
        {membaca. pesan dan mengambil nilai nonce}
        Confound: Procedure (pesan: string)
        {Menyamarkan data dengan menambahkan confounder pada pesan}
        Krip : Function(pesan,kunci : string) ??string
        {mengacak pesan dengan memakai kunci}
        Kirim: Function (pesan, server :string) ??boolean
        {Mengirim pesan kepada server}
Algoritma:
        If (no_pesan = 1)
        Pesan ??gettime() + getcmd() + rand(seed,no) + Authenticator;
        Confound(pesan);
        Krip(pesan,kuncisesi)
        Kirim(pesan, Server);
        Endif
        If (no_pesan = 3)
        Bongkar_pesan(pesan);
        If (uji(Authenticator) == notOK)
        Exit; {Client tidak, terpercaya}
        Else
        Exec(getcmd());
        Endif
        endif
```

Kriteria Pengukuran

Tingkat kerahasiaan pada sistem keamanan ini diukur dari waktu yang diperlukan penyerang untuk membongkar sistem keamanan. Penyerang dalam hal ini memakai metode tebakan terhadap kunci dari pesan dan menguji kebenaran tebakannya dengan membandingkannya dengan pesan-pesan yang telah berhasil dicuri.

Waktu yang diperlukan sebelum sistem keamanan terbongkar harus memenuhi persyaratan penyerangan terhadap kunci pada pesan harus mampu ditunda hingga minimal 60 menit. Karena setiap satu jam program hasil implementasi akan

melakukan perubahan kunci sesi yang digunakan untuk berhubungan dengan server.

Pembahasan Pengukuran

Dengan memakai algoritma diatas usaha untuk mencari kunci dilakukan. Perulangan yang dila-kukan oleh algoritma tersebut untuk mencari kunci sangat tergantung perkiraan waktu terjadinya pertukaran pesan tersebut dituliskan oleh pengirim pesan. Dalam pengujian dilakukan penghitungan waktu yang diperlukan untuk menemukan kunci dengan anggapan tebakan waktu transaksi tidak lebih dari ±5 detik. Sehingga tebakan dilakukan dengan tebakan terhadap waktu pesan selama waktu 10 detik.

Algoritma yang telah kita sebutkan diatas melakukan tebakan untuk satu buah tebakan selama ± 0,1 detik. Dengan asumsi bahwa kita harus melakukan 100 macam tebakan untuk selang waktu 10 detik, maka program harus melakukan tebakan sebanyak 10 (detik) kali 100 (milidetik) kali 1(1 buah kandidat kunc1) buah tebakan hasilnya harus melakukan tebakan sebanyak 1.000 kali. Sehingga waktu yang diperlukan untuk menebak kunci dengan memakai seluruh kandidat kunci adalah adalah 100 000 kali 0,l detik (waktu yang diperlukan untuk satu tebakan kunci) dengan kata lain waktu yang diperlukan adalah 10 000 detik atau 2 jam 46 menit 40 detik. Lebih dari yang telah didefinisikan diatas.

Penutup

Kesimpulan

1. Pengendalian akses keluar/masuk pegawai tetap dapat dijaga konsistensinya terhadap transaksi basis data, walaupun digunakan smart card sebagai pengganti kartu magnetik pada sistem yang sedang atau sudah berjalan.
2. Pengendalian status kartu serta proses transfer data dan aliran data dapat dilakukan dengan mengeluarkan commnad berupa string yang ditetapkan antara terminal komputer ke card reader atau sebaliknya.
3. Implementasi kriptografi FEAL dengan jumlah round number 8, kunci 64 bit dan proses ciphering per blok 64 bit pada sistem komunikasi digunakan untuk menjaga keamanan data setiap kartu pegawai selama dijalur komunikasi dari card reader ke komputer.
4. Dari uji pengukuran terhadap pengiriman paket data baik data original atau non original, untuk kerja perangkat lunak yang diimplementasikan pada prosespengiriman dan penerimaan data akan didapat hasil terbaik, jika digambarkan secara grafik pertambahan jumlah paket data te waktu dihasilkan grafik linear dan overhead T1proses, adalah kecil

Saran

Ada beberapa saran yang diberikan oleh penulis untuk pengembangan lebih lanjut aplikasi smart card yaitu sebagai berikut :

1. Untuk mengintegrasikan sistem basis data dengan akses kontrol smart card akan menjadi lebih padu, bila perangkat lunak dibuat dalam satu paket under Windows sehingga bisa multitasking dengan program aplikasi lain atau workstation tidak bersifat dedicated lagi.
2. Pengembangan lebih lanjut aplikasi card tidak hanya untuk sistem absensi tetapi untuk mengintegrasikan seluruh transaksi data kepegawaian STMIK AMIKOM Yogyakarta, akses kontrol pintu,kornputer, machine fotocopy dan sebagainya.

Daftar Pustaka

[1] McCrindle, John. 1990, Smart Card IFS Pubhsing /Springer-Verlag: Berli Heidelberg New York London Paris Tokyo

[2] S.K Tunothy. 1992, Windows Programmer's Guide to serial Comunication, SAMS Publising:Indiana

[3] [BIRD93) R. Bird, L. Gopal, A. Herzberg, P. A. Janson, S.Kutten, RMolva, M.Yung, " Systematic Design of a Family of Attack Resistant Authentication Protocols",*IEEE Journal On Selected Areas in Communication,* Vol 11, No 5, Juni 1993. Hal 679 - 693.