Analisis Pengamanan dan Serangan Terhadap Transaksi ATM

Samuel Simon – NIM: 13506032
Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganeca 10, Bandung
E-mail: if16032@students.if.itb.ac.id

Abstraksi
Saat ini kehidupan masyarakat di kota besar tidak lepas dengan dunia perbankan. Salah satu layanan yang diberikan bank bagi nasabahnya adalah Anjungan Tunai Mandiri (ATM). Dengan menggunakan satu kartu ATM, pemegang kartu dapat menarik uang tunai di tempat-tempat tertentu. Dengan demikian, nasabah tidak perlu mengalami kerepotan mencari bank terdekat dan mengantri di sana. Tidak hanya itu, penggunaan kartu ATM semakin berkembang hingga diterapkan pada tempat-tempat belanja. Pengguna dapat menggunakan kartu tersebut untuk membayar belanjaan mereka. Kartu ATM semakin penting karena telah menjadi pusat transaksi sebagian masyarakat. Oleh sebab itu, perlu dibuat suatu metode untuk mengamankan transaksi yang menggunakan kartu tersebut. Hal ini sangat penting, mengingat semakin maraknya serangan yang dilakukan untuk mencuri data-data yang ada dalam kartu tersebut. Makalah ini akan membahas tentang cara pengamanan transaksi perbankan melalui ATM. Beberapa metode yang telah menjadi standar di dunia perbankan internasional akan dijelaskan. Selain itu, akan dipaparkan pula algoritma-algoritma yang digunakan, serangan terhadap transaksi ATM beserta penanganannya, beberapa pengembangan yang dapat dan telah diterapkan, dan juga tak lupa beberapa teknik optimasi yang dapat diimplementasikan berdasarkan analisa penulis.

Kata kunci: *DES, ATM, Bond-Zelinski, PIN, pengamanan ATM, serangan ATM, pencurian PIN*

Pendahuluan
Di tahun 1960an, industri perbankan mulai mencari cara untuk memberikan layanan perbankan elektronik untuk mengatasi kurangnya sumber daya yang dapat dijadikan pegawai bank. Akhirnya, diciptakanlah sebuah terminal untuk masalah tersebut. Saat ini, terminal tersebut dikenal dengan nama anjungan tunai mandiri (ATM).

Sejak pertama kali digunakan, ATM banyak memberikan keuntungan kepada tidak hanya industri perbankan, namun juga para konsumer atau pengguna jasa layanan perbankan. Keuntungan-keuntungan tersebut di antaranya adalah:

- Konsumer dapat melakukan berbagai transaksi perbankan (penyimpanan, penarikan, pengecekan akun, transfer antar rekening) pada waktu apapun.
- Transaksi elektronik tidak membutuhkan adanya manusia untuk mengawasi, sehingga dapat mengurangi pegawai yang dapat digunakan.
- Dengan berkurangnya pegawai yang ada, bank juga dapat menghemat pengeluaran yang cukup besar.

Pada awalnya, ATM dapat diakses dengan menggunakan kartu kredit. Namun dengan penggunaan kartu kredit sebagai kartu ATM, bank mulai menghadapi masalah baru, yaitu keamanan. Salah satu masalah ini adalah jika ada seseorang yang mencoba untuk mengambil seluruh uang di ATM, kemudian melarikan diri. Selain itu, terdapat juga masalah lain seperti penggunaan kartu kredit palsu dan kehilangan atau pencurian kartu kredit.

Dengan banyaknya resiko tersebut, bank membuat cara lain untuk mensahkan transaksi di ATM. Cara tersebut adalah dengan memberikan sebuah kartu baru kepada konsumen yang memiliki beberapa fitur:

- Penyimpanan nomor akun pribadi konsumen pada strip hitam kartu.
- Penyimpanan nomor identifikasi personal (PIN), PIN ini juga dapat digantikan hal lain seperti sidik jari.

Dengan demikian, beberapa masalah dapat diatasi. Jumlah uang yang diambil bisa diawasi karena adanya nomor akun yang harus dicek. Selain itu, sesaat setelah kehilangan kartu, konsumen dapat langung melaporkan hal tersebut kepada bank. Kartu akan sulit digunakan oleh orang lain dengan adanya PIN, dan bank dengan mudah dapa melakukan pemblokiran pada kartu dengan adanya nomor khusus pada setiap kartu.

Autentifikasi Pengguna
Pada saat akan melakukan transaksi ATM, pengguna harus melakukan hal berikut:

1. Memasukkan kartu ATM kemudian nomor akun yang disimpan pada kartu akan dibaca oleh mesin.

2. Memasukkan PIN pada papan kunci ATM.

Nomor akun disimpan sebagai nilai Q_1 dan PIN yang dimasukkan disimpan sebagai nilai Q_2. Aturan yang biasa digunakan pada mesin ATM adalah dengan melihat pada tabel autentifikasi yang dimiliki bank. Namun terdapat beberapa ancaman pada aturan ini, yaitu:

1. Isi dari tabel yang akan dilihat oleh ATM, dapat diubah datanya oleh orang-orang yang tidak bertanggung jawab.
2. Jalur komunikasi yang digunakan ATM untuk mengirimkan nilai Q_1 dan Q_2 dapat disusupi sehingga nilai Q_2 dapat diketahui.

Cara untuk mengatasi permasalahan ini adalah dengan:

- Melakukan enkripsi pada tabel dan membuat data yang ada tidak dapat ditulis. Hal ini akan membuat orang-orang yang tidak bertanggung jawab akan kesulitan untuk membaca dan mengubah data yang ada.
- Melakukan enkripsi pada data yang akan dikirim dari ATM, sehingga nilai-nilai yang dikirim sulit dibaca oleh pihak lain.

Tidak ada satupun dari solusi tersebut yang dapat menjadi solusi sebenarnya. Hal ini disebabkan, pihak lain masih dapat mencuri dan mengubah data yang ada. Namun, dengan cara penanganan tersebut, maka ancaman-ancaman yang akan dilakukan tersebut menjadi lebih sulit. Tujuan akhir dari metode di atas bukan untuk membuat ancaman-ancaman yang akan dilakukan menjadi tidak mungkin untuk dilakukan, namun untuk membuat ancaman-ancaman tersebut sulit dilakukan dan menjadi tidak efektif. Namun, pengamanan data-data tersebut selalu menjadi masalah tersendiri bagi bank.

Operasi *Online*/*Offline*

Kemampuan sistem komputer dan kebutuhan perawatan komputer secara periodik di tahun 1960an memaksa pihak bank untuk memberikan dua pilihan operasi pada ATM, yaitu:

- Online: identifikasi pengguna dilakukan dengan memverifikasi langsung ke komputer pusat.
- Offline: identifikasi pengguna dilakukan hanya dengan memverifikasi pada lokal ATM saja

Bank harus dapat memastikan kedua pilihan tersebut bekerja. Pada saat normal, operasi dilakukan secara *online*, namun pada saat komputer pusat mengalami proses perawatan, maka mode yang dipilih adalah mode *offline*.

Keterbatasan kapabilitas ATM dan kemungkinan pengguna ATM yang dapat berkembang menjadi semakin besar mengakibatkan ukuran tabel untuk menyimpan data-data tersebut akan menjadi semakin besar dan tidak dapat disimpan secara lokal di ATM.

Terdapat pula masalah lain yang harus menjadi pertimbangan, yaitu daftar konsumen dapat berubah setiap saat dengan adanya penambahan dan penghapusan konsumen dari tabel setiap harinya. Jika saja terdapat 100 ATM yang harus diperbarui datanya setiap hari, maka keuntungan penggunaan ATM dari sisi biaya akan menghilang. Kemungkinan yang dapat dilakukan adalah dengan menghubungkan ATM dengan jaringan. Hal tersebut dapat membuat perubahan data dilakukan dengan cepat dan secara bersamaan. Di sisi lain, hal ini mengakibatkan sistem ATM akan menjadi semakin rentan untuk disusupi.

Enkripsi Sebagai PengamanTransaksi ATM

Solusi permasalahan tersebut adalah dengan membuat Q_1 dan Q_2 berhubungan:

$$\mathbf{Q_2} = f(\mathbf{Q_1})$$

Proses enkripsi dilakukan untuk mengatasi permasalahan ini. Misalkan (Q_1) $Q_2 = E_K\{Q_1\}$. Jika algoritma $E_K\{...\}$ cukup kuat, maka informasi akan pasangan (Q_1,Q_2) dan untuk ($\{Q_1^{(i)}\},\{Q_2^{(i)}\} \mid 1 \leq i \leq N$) membuat pengguna ataupun orang yang mencuri data tidak dapat dengan mudah mendeduksi nilai K.

Untuk mengautentifikasi pengguna, mesin ATM akan memastikan apakah relasi $Q_2 = E_K(Q_1)$ terpenuhi. Ini berarti nilai K harus tersimpan pada setiap ATM. Hal ini membuat timbulnya resiko kunci tersebut diambil karena tidak adanya penjagaan di ATM. Oleh sebab itu, setiap ATM memiliki sebuah modul yang bernama *high-security module* (HSM), sebuah pemroses tambahan yang anti kerusakan dan pencurian dan berfungsi untuk memvalidasi PIN. Kunci K tersebut juga tersimpan dengan aman pada HSM.

Algoritma enkripsi yang digunakan pada transfer data ATM adalah sebuah protokol yang dibuat oleh IBM berbasiskan LUCIFER, sebuah teknik enkripsi berbasiskan algoritma DES.

Jika Q_1(ID_Konsumen) diberikan oleh pihak bank, dan PIN(ID_Konsumen) = $E_K\{ Q_1(\text{ID\_Konsumen}) \}$, dihasilkan berdasarkan masukan pengguna, hal ini juga mengakibatkan pengguna tidak dapat memilih PIN secara bebas. Hal ini diatasi dengan dibuatnya PINoffset, sebuah modul tambahan yang dibuat oleh Chubb Integrated Systems. PINoffset disimpan pada kartu ATM. Q_1(ID_Konsumen),

PIN(ID_Konsumen), PIN_Offset(ID_Konsumen), dan U - PIN(ID_Konsumen) oleh IBM didefinisikan sebagai:

U-PIN(ID_Konsumen) = Left$_{16}$[E_K{Q_1(ID_ Konsumen)}] + PINOffset(ID_Konsumen)

,dimana Left$_{16}$[...] menyatakan 16 bit paling kiri yang ada dalam nilai tersebut.

Proses pada transaksi ATM menjadi:
Pengguna memasukkan kartu ATM dan mesin akan membaca kartu tersebut,
Pengguna memasukkan U–PIN(ID_Konsumen), Q_1(ID_Konsumen) dan PINOffset(ID_Konsumen) dibaca dari kartu ATM,
Proses U-PIN(ID_Konsumen) = Left$_{16}$[E_K{Q_1(ID_ Konsumen)}] + PINOffset(ID_Konsumen) dilakukan di ATM dan validitas dari U-PIN(ID_Konsumen) =Left$_{16}$[E_K{Q_1(ID_Konsumen) }] + PINOffset(ID_Konsumen) dapat dipastikan.

Satu kelemahan yang ada pada skema tersebut adalah adanya 4 digit hexadesimal dari U-PIN(ID_Konsumen) yang dapat berisi nilai `0, 1, 2, ..., 9, A, B, ..., F`. Nilai tersebut, khususnya `A, B, .., F` tidak ada pada papan kunci ATM. Untuk mengatasi permasalahan ini, dibuatlah sebuah tabel pemetaan untuk melakukan desimalisasi nilai-nilai tersebut. Tabel tersebut berbentuk seperti ini:

0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7

8	9	A	B	C	D	E	F
8	9	0	1	2	3	4	5

Gambar 1. Contoh tabel desimalisasi

Verifikasi PIN dilakukan di modul HSM. Antarmuka untuk program aplikasi (API) untuk modul HSM telah dibuat IBM dengan sintaks `Encrypted_PIN_Verify`(...), yang akan mengembalikan nilai `YES/NO`. Sebagai tambahan pada Q_1, salah satu masukan adalah tabel desimalisasi.

Serangan Transaksi ATM

Pada tahun 2003, Bond dan Zelinski telah menemukan cara untuk menemukan U-PIN(ID_ Konsumen). Cara ini dapat dilakukan oleh pemrogram sistem perbankan yang memiliki akses ke sistem secara langsung. Pemrogram sistem yang memiliki akses langsung ke sistem dapat melakukan verifikasi PIN menggunakan HSM dan melakukan 60 percobaan dalam satu detik. Pemrogram sistem mencoba masukan (Q_1,PINOffset) dengan 10 U-PIN berbeda yang dipilih dengan aturan U-PIN$_j$ dipilih sehingga:

PIN_Konsumen$_j$+PINOffset = (j, j, j, j), 0 ≤j ≤ 9,

dan tabel desimalisasi dipilih sehinga baris kedua memiliki nilai satu untuk baris pertama yang memiliki nilai *j*. HSM akan menghasilkan keluaran YES jika PIN mengandung nilai *j* dan NO jika tidak. Dengan demikian, 10 percobaan akan menentukan nilai apa saja yang terdapat dalam PIN. Jika nilai berbeda yang dapat ada dalam PIN adalah *K*, maka dibutuhkan T_k tambahan percobaan dimana:

$$T_K = \begin{Bmatrix} 1, jika\ k = 1 \\ 14, jika\ k = 2 \\ 36, jika\ k = 3 \\ 240, jika\ k = 4 \end{Bmatrix}$$

Nilai rata-rata percobaan yang perlu dilakukan untuk mendapatkan PIN yang sebenarnya kurang lebih 240 kali percobaan. Jika bank mengizinkan penarikan uang Rp. 5.000.000,00 untuk satu kartu dalam satu hari, maka pada istirahat selama 30 menit, seorang pemrogram sistem yang tidak jujur dapat menemukan:

$$\frac{30\ x\ 60\ x\ 60}{240} = 450$$

PIN dan menarik Rp.2.250.000.000,00.

Serangan Terhadap PIN serupa dengan sebuah pohon pencarian. Setelah dianalisa, penulis menemukan bahwa algoritma ini dapat dioptimasi sehingga dapat mengurangi jumlah percobaan yang perlu dilakukan untuk melakukan serangan.

Setelah dianalisa, algoritma tersebut dapat dioptimasi dengan menambahkan algoritma heuristik pada algoritma awal tersebut. Algoritma heuristik ini digunakan untuk menghilangkan beberapa kemungkinan yang sudah pasti tidak akan menjadi solusi.

Selain itu, dengan beberapa percobaan menggunakan contoh kasus pencarian pohon, ditemukan bahwa penggunaan algoritma yang berbeda untuk mencari kombinasi angka menghasilkan optimasi pada pencarian secara keseluruhan. Proses DFS secara umum memberikan performansi yang lebih baik. Hal ini dapat dijelaskan dengan mengetahui cara kerja DFS.

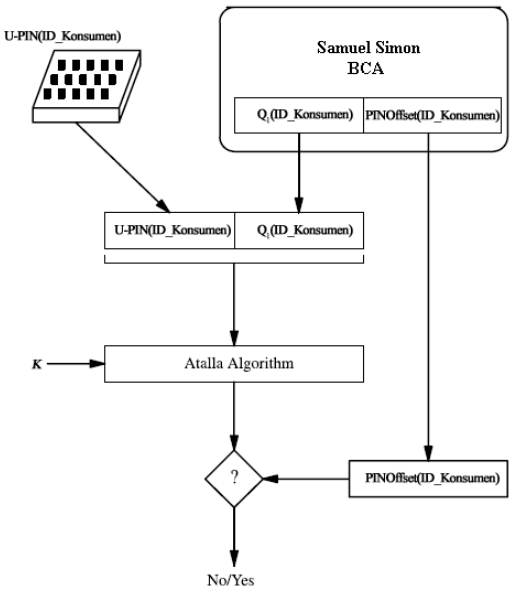
Dengan DFS, maka pencarian akan dilakukan hingga selesai suatu kemungkinan. Dengan demikian, maka pencarian kemungkinan tersebut akan berhenti saat suatu kemungkinan sudah

dipastikan sebagai solusi. Hal ini cukup meningkatkan performansi dari algoritma, karena algoritma tidak perlu mengecek semua kemungkin yang ada.

Solusi Serangan

Untuk mengatasi serangan ini, dapat diterapkan suatu cara sederhana, yaitu dengan menghilangkan tabel desimalisasi sebagai masukan.

Algoritma pengecekan PIN dapat dimodifikasi menjadi seperti berikut:



Gambar 2. Alternatif Proses Validasi PIN ATM

PINOffset(ID_Konsumen) dapat dihasilkan dengan mengenkripsi gabungan Q_1(ID_Konsumen) dan U-PIN(ID_Konsumen). Gabungan ini dapat dienkripsi dengan menggunakan DES. PINOffset hasil enkripsi tersebut disimpan secara magnetis di dalam kartu.

PINOffset(ID_Konsumen) = DES_K\{ Q_1(ID_ Konsumen)|| U-PIN(ID_Konsumen)\}

Dengan demikian, maka penggunaan tabel desimalisasi sudah tidak diperlukan lagi.

Autentifikasi secara *offline* tidak jauh berbeda dengan autentifikasi secara *online*. Proses tersebut hanya merupakan pengulangan proses pada ATM. Ada keuntungan tersendiri melakukan pengubahan teknik validasi PIN Pengguna dengan relasi U-PIN(ID_Konsumen) dan PINOffset(ID_Konsumen) seperti di atas. Keuntungan yang paling signifikan adalah PIN konsumen tidak disimpan di komputer pusat bank, melainkan tersimpan di dalam kartu ATM masing-masing pelanggan. Dengan demikian proses *offline* tidak lagi bergantung pada komputer pusat.

Sebuah perusahaan di Amerika, The National Cash Register Company, telah mengembangkan dan memasarkan produk ATM seperti ini. NCR mengembangkan suatu algoritma $E_k\{...\}$ khusus untuk melakukan komputasi pada produk ATMnya pada masa awal produksi. Namun, pada akhirnya NCR turut mengubah algoritma yang digunakannya menjadi algoritma DES.

Daftar Pustaka

- M. Bond dan P. Zelinski. “Decimilisation Table Attacks for PIN Cracking”. University of Cambridge, Computer Laboratory, Report 560. February 2003.
- G. Konhem, Alan. Computer Security and Cryptography. John Wiley. 2007.
- G. Purdy. “A High Security Log-In Procedure”. Communications of the ACM, 17, 442–445. 1984.