Disain Autentikasi Kunci Publik Menggunakan Teori Matriks
(*Authentication Design of Public Key Using Matrices Theory*)

Nama : Bayu Adi Persada
NIM : 13505043

Program Studi Teknik Informatika
Institut Teknologi Bandung
Jalan Ganesha No 10
Bandung

E-mail : *if15043@students.if.itb.ac.id*

ABSTRAK

Karya ini dibuat untuk mengembangkan skema autentikasi dalam kunci publik. Ide pengembangan dari skema autentikasi didasarkan kepada teori dari kode linear dengan teknik invers matriks. Berdasarkan skema enkripsi dan deksripsi, disain autentikasi sangat efesien dalam penggunaan untuk mengunci ruang dan tentunya mempunyai kompleksitas yang rendah, yaitu anggota dari O (n^2).

Penjelasan sebelumnya mengungkapkan bahwa disain autentikasi cukup cepat untuk mengenkripsi dan mendekripsi sebuah pesan. Selain itu, disain autentikasi cukup aman dari serangan penyusup yang tidak bertanggungjawab. Kelemahan dari disain ini adalah terdapat ekspansi dari pesan itu sendiri.

Kata Kunci *:* matriks, autentikasi, enkripsi, dan dekripsi.

A. PENDAHULUAN

Sejak diperkenalkan teknik baru di bidang kriptografi oleh Diffe dan Hellman pada tahun 1976, telah banyak cabang matematika yang digunakan untuk mengembangkan bidang kriptografi. Beberapa cabang matematika yang telah digunakan untuk mengembangkan kriptografi dapat disebutkan di antaranya adalah aljabar, teori bilangan, dan teori koding. Shao (1998) telah mengembangkan konsep autentikasi kunci publik berdasarkan faktorisasi dan logaritma diskrit. Jenis kunci publik yang lain adalah RSA yang menggunakan teori bilangan, ElGamal menggunakan logaritma diskrit, Elliptic Curve System yang dikembangkan berdasarkan teori grup, The Merkle-Ellman Knapsack System yang berdasar pada subset sum, dan McEliece public key system yang menggunakan teori koding (Stinson, 1995; Patterson,1987). Sementara itu, matriks invers tergeneralisasi (MIT) juga dapat digunakan untuk mengembangkan sistem kunci publik (Wu and Dawson, 1998).

Tujuan utama melakukan enkripsi data atau pesan adalah untuk menjamin keamanan (security) dari data yang akan dikirimkan. Bentuk pelayanan keamanan dalam kriptografi adalah autentikasi dan kerahasiaan pesan. Untuk itu Stalling (1995 : 114) menyarankan bahwa dalam mendisain kunci publik hendaknya memenuhi kedua jenis layanan keamanan tersebut; hal ini mengingat bahwa umumnya kunci publik banyak yang didisain hanya memenuhi layanan kerahasiaan data. Sedangkan Simmons (1993:232) menyarankan bahwa dalam mendisain kunci publik, selain harus menjamin keamanan data, sistem kunci publik juga harus efisien, khususnya dalam hal penggunaan ruang kunci, kompleksitas enkripsi dan dekripsi, dan ekspansi pesan.

Shao (1998:33-36) telah membuat skema autentikasi kunci publik yang dikembangkan berdasarkan faktorisasi dan logaritma diskrit. Dalam penelitiannya tersebut Shao menyimpulkan bahwa disain yang dibangun berdasarkan faktorisasi dan logaritma diskrit telah mampu memberikan layanan autentikasi dan kerahasiaan data. Artinya data yang ditransmisikan tidak dapat diterobos (unbreakable) oleh penyusup (intruder) jika problem faktorisasi dan logaritma diskrit secara

simultan tidak dapat diselesaikan. Tetapi sebagaimana telah ditunjukkan oleh Lee, dalam penelitian Shao tersebut tersirat adanya kelemahan, yaitu jika problem faktorisasi dapat diselesaikan, maka disain autentikasi kunci publik akan dapat diterobos (Lee, 1999:119). Ini berarti bahwa skema autentikasi kunci publik berdasarkan logaritma diskrit dan faktorisasi belum dapat memberikan layanan kerahasiaan data yang ditransmisikan secara maksimal. Ini berarti perlu dikembangkan skema kunci publik yang lain yang mampu memberikan jaminan autentikasi dan juga kerahasiaan data sekaligus.

Wu dan Dawson menawarkan disain kunci publik yang menggunakan MIT. Ide dasar pengembangan kunci publik tersebut menggunakan koreksi kesalahan kode dengan teknik MIT (Wu and Dawson, 1998). Dari disain telah dapat ditunjukkan bahwa MIT dapat digunakan untuk melakukan enkripsi dan dekripsi data, serta mampu menjamin kerahasiaan data. Tetapi disain yang diajukan Wu dan Dawson ini masih memiliki problem autentikasi antara pengirim dan penerima pesan. Artinya, antara pengirim dan penerima pesan (message) tidak dapat menjamin 'keaslian' pasangannya. Sebab dengan kunci (key) yang bersifat publik, siapa saja dapat mengirimkan pesan atau data menggunakan kunci tersebut. Ini berakibat penerima pesan tidak dapat mengetahui dengan pasti bahwa yang mengirimkan pesan memang orang yang diharapkan. Untuk itulah perlu dikembangkan skema autentikasi kunci publik berdasarkan matriks invers tergeneralisasi (Murtiyasa, 2001). Dengan skema autentikasi, diharapkan problem keamanan pengiriman data antara pengirim dan penerima pesan dapat di atasi.

Berdasarkan uraian tersebut di atas, tulisan ini secara umum betujuan untuk mengkaji pembuatan disain autentikasi kunci publik berdasarkan teori matriks. Secara rinci akan mengkaji :
(1)metode enkripsi untuk mendapatkan ciphertext c dari suatu pesan m,
(2)metode dekripsi untuk mendapatkan kembali pesan m dari ciphertext c,
(3)karakteristik disain autentikasi kunci publik, dan
(4)resiko-resiko serangan terhadap disain autentikasi kunci publik.

B. METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah studi pustaka dengan dukungan implementasi program komputasi. Dengan studi pustaka diharapkan diperoleh berbagai informasi yang berhubungan dengan enkripsi dan dekripsi data, kriptografi kunci publik, dan teori - teori matriks invers yang mendukung sistem kunci publik.

Penggunaan program komputasi dimaksudkan untuk membantu menunjukkan kebenaran hasil-hasil teorema atau proposisi yang berhubungan dengan disain autentikasi kriptosistem kunci publik berdasarkan matriks invers. Ide dasar pengembangan mengunakan teori koding linear, yaitu $\mathbf{c} = \mathbf{mG}$. Dalam hubungan tersebut **c** adalah katakode (*codeword*) dari pesan **m** yang disandikan menggunakan matriks generator **G**, lebih detail lihat (Vanstone dan Oorschot, 1989).

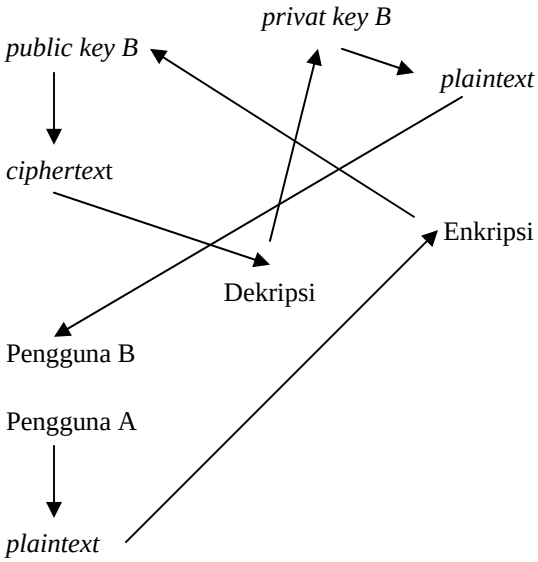
Kaitannya dengan pembuatan disain autentikasi kunci publik, untuk mendapatkan katakode **c**, penelitian ini akan melakukan rekonstruksi terhadap matriks generator **G** dengan teknik matriks invers.

C. HASIL DAN PEMBAHASAN

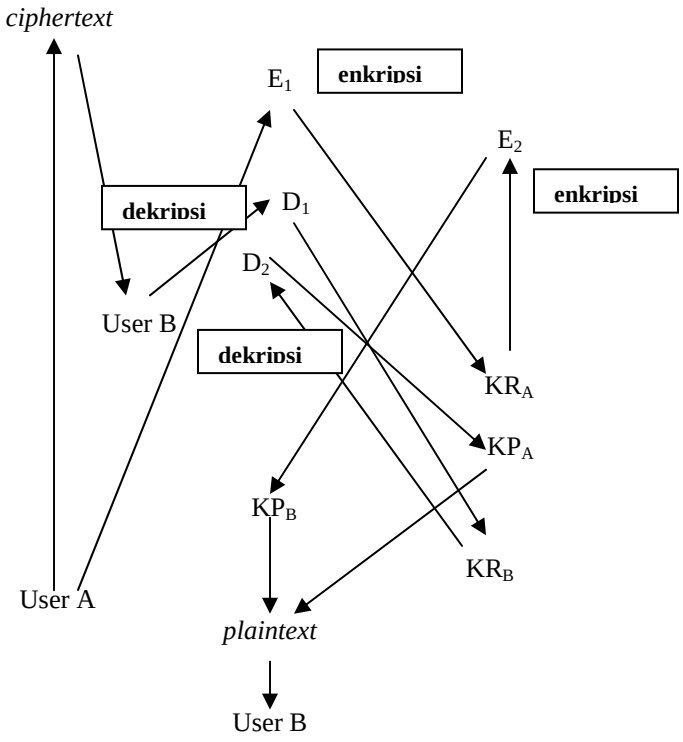
Kunci Publik

Skema enkripsi dan dekripsi dari sistem kunci publik bisa dijelaskan sebagai berikut : data (pesan) dari pengguna (*user*) A, yang biasa disebut dengan *plaintext*, dienkripsikan (disandikan) dengan kunci milik pengguna B, dalam hal ini berupa kunci publik (*public key*). Setelah *ciphertext* (data yang sudah disandikan) diterima pengguna B, *ciphertext* ini kemudian didekripsikan dengan menggunakan kunci kedua milik pengguna B, dalam hal ini disebut kunci pribadi (*privat key*); sehingga pengguna B bisa membaca *plaintext* yang dikirim pengguna A. Proses sederhana ini bisa digambarkan sebagaimana ditunjukkan Gambar 1.

Dalam implementasi, skema enkripsi dan dekripsi sebagaimana diilustrasikan pada Gambar 1 di samping memang mampu menjamin kerahasiaan pesan yang diterima user B, sebab hanya user B yang memiliki kunci pribadi untuk mendekripsi pesan yang dikirimkan user A. Tetapi skema tersebut belum mampu memberikan layanan autentikasi (authentication) pesan yang dikirim, sebab berdasarkan skema tersebut dimungkinan setiap orang (selain user A) dapat mengirimkan pesan ke user B dengan menggunakan kunci publik user B yang dipublikasikan. Skema enkripsi dan dekripsi yang menjamin kerahasiaan dan autentikasi pesan dapat diilustrasikan sebagaimana ditunjukkan pada Gambar 2.



Gambar 1: Skema enkripsi-dekripsi sistem kunci publik plaintext



Gambar 2: Skema autentikasi kunci publik

Dari Gambar 2 tersebut dapat dijelaskan bahwa jika user A ingin mengirimkan *plaintext* ke user B, pertama-tama user A melakukan enkripsi E_1 menggunakan kunci pribadi (KR_A) milik user A, kemudian melakukan enkripsi E_2 menggunakan kunci publik (KP_B) milik user B. Selanjutnya user A mengirimkan *ciphertext* ke user B. Pada saat *ciphertext* diterima user B, pertama-tama user B mendekripsi *ciphertext* tersebut menggunakan kunci pribadi KR_B milik user B, kemudian didekripsi lagi menggunakan kunci publik KP_A milik user A. Selanjutnya user B akan mendapatkan *plaintext* yang dikirimkan user A.

Matriks Invers

Telah diketahui bahwa teori matriks banyak diaplikasikan untuk menyelesaikan persoalan-persoalan yang dapat dibawa kedalam bentuk model persamaan linear. Salah satu di antaranya adalah penerapannya pada permasalahan teori penyandian data atau teori koding, khususnya pada kode linear. Dalam prakteknya, metode penyandian data dengan matriks dapat memanfaatkan operasi penjumlahan matriks dan/atau perkalian matriks. Misalnya pada field Z_2 (bilangan bulat modulo 2), diketahui pesan **m** = (1 0 1) dengan bantuan matriks **b** = (1 1 0) diperoleh katakode **c** = **m** + **b** = (0 1 1). Dengan memanfaatkan operasi perkalian matriks, misalnya pesan **m** = (1 0 1) dan matriks

G = , diperoleh katakode **c** = **mG** = (1 0 1)
= (0 1 1 0 1).

Ilustrasi ini menunjukkan bahwa teori matriks sangat dimungkinkan untuk dikembangkan dalam teori penyandian data.

Dalam teori matriks telah diketahui bahwa untuk matriks persegi **A** berdimensi *n*x*n* yang nonsingular, matriks **A** tersebut mempunyai invers, yang dinotasikan dengan $\mathbf{A}^{-1}$, sedemikian hingga berlaku $\mathbf{A}\,\mathbf{A}^{-1} = \mathbf{A}^{-1}\,\mathbf{A} = \mathbf{I}_n$. Metode untuk mendapatkan matriks $\mathbf{A}^{-1}$ di antaranya dapat dilakukan dengan jalan (i) operasi baris elementer, (ii) menggunakan matriks adjoint dan determinant.

Ayres (1982) secara khusus menunjukkan bahwa jika matriks **A**(*m*,*n*) mempunyai rank *m*, maka matriks **A** tersebut mempunyai invers kanan (*right inverse*) matriks $\mathbf{A}^k$ (*n*,*m*), sedemikian hingga $\mathbf{A}\,\mathbf{A}^k = \mathrm{I}_m$, dengan notasi $\mathbf{A}^k$ menyatakan invers kanan dari **A**. Salah satu metode untuk mencari invers kanan dari matriks **A** tersebut dapat dijelaskan sebagai berikut ini.

1. Pilih submatriks dari **A**, katakanlah **S**, yang berdimensi *m*x*m* dan non singular.
2. Cari invers dari **S**, yaitu $\mathbf{S}^{-1}$.
3. Invers kanan dari **A** adalah $\mathbf{A}^k$ yang berdimensi *n*x*m* di mana : jika matriks **S** diperoleh dengan jalan menghilangkan kolom ke-*i* dari matriks **A**, maka baris ke-*i* dari matriks $\mathbf{A}^k$ adalah baris nol, sedangkan baris-baris sisanya berasal dari baris-baris $\mathbf{S}^{-1}$ yang bersesuaian dengan kolom-kolom dari matriks **A** untuk memperoleh matriks **S** (Ayres Jr, 1982).

Memperhatikan cara mendapatkan invers kanan tersebut, jelas bahwa banyaknya invers kanan dari suatu matriks adalah tidak tunggal.

Kode Linear

Suatu *k* blok data atau pesan **m** = $m_1m_2m_3\ldots m_k$ akan dikodekan ke dalam katakode (*codeword*) ***x*** = $x_1x_2x_3\ldots x_n$, dimana *n* *k*. Katakode – katakode tersebut membentuk kode (*code*). Melalui prosedur tersebut, bagian pertama dari katakode memuat data itu sendiri, yaitu :
$x_1 = m_1$, $x_2 = m_2$, $x_3 = m_3$, …, $x_k = m_k$; sedangkan bagian berikutnya memuat *n*-*k* simbol kontrol (*check symbols*), yaitu x_{k+1}, x_{k+2}, …, x_n. Baik data maupun katakode berada dalam sistem biner, karenanya operasi hitung yang berhubungan dengan katakode mengikuti operasi hitung dalam modulo 2.

Pada kode linear **C**-(n,k) yang mempunyai dimensi *k*, berarti **C** dapat dibangun oleh *k* buah vektor yang bebas linear. Dengan demikian, jika satu basis dari **C** dipilih, maka ada korespondensi satu-satu antara ruang pesan (data) berdimensi-*k* dengan kode **C**. Suatu kode linear juga bias dibangun oleh suatu matriks tertentu. Matriks yang digunakan untuk membangun kode linear ini dinamakan matriks generator. Matriks generator **G** untuk kode linear **C**-(n,k) adalah suatu matriks berdimensi *k*x*n* yang mempunyai baris-baris berupa vektor basis dari **C** (Vanstone dan Oorschot, 1989).

Untuk pesan **m** dan matriks generator **G**, maka **c** = **mG** adalah katakode. Karena basis suatu ruang vektor yang berdimensi *k* adalah tidak tunggal, maka matriks generator **G** untuk kode linear **C** pun tidak tunggal. Tetapi satu matriks generator tertentu mungkin lebih berguna (menguntungkan) daripada matriks generator yang lain. Dalam hal matriks generator **G**

berbentuk **G** = [**I** $_k$ **A**], dengan **I** $_k$ adalah matriks identitas berdimensi *k*x*k* dan **A** adalah matriks berdimensi *k*x*(n-k)*, matriks **G** tersebut dikatakan dalam bentuk standard (*standard form*). Suatu matriks generator yang belum dalam bentuk standard dapat dibawa ke dalam bentuk standard dengan jalan melakukan sederetan operasi baris elementer.

Skema Autentikasi

Dalam teori koding linear telah disebutkan bahwa untuk suatu pesan m dengan matriks generator **G**, maka c = m**G** adalah katakode. Untuk menjamin keamanan data atau pesan m, dari disain c = m**G** dapat dilakukan rekonstruksi terhadap matriks **G**. Model rekonstruksi untuk matriks generator **G** tersebut dapat dijelaskan sebagai berikut :

1.Pilih matriks generator **G** = [I_k A], dengan sembarang matriks A berdimensi *k*x(*n*-*k*).
2.Sembarang matriks non singular **S** berdimensi *k*x*k*.
3.Pilih sembarang matriks permutasi **P** berdimensi *n*x*n*.
4. G' = **SGP**

Selanjutnya matriks **G**' ini digunakan untuk melakukan enkripsi pesan m untuk memperoleh *ciphertext* c, jadi c = m**G**'. Jadi di sini matriks **G**' berfungsi sebagai kunci publik untuk melakukan enkripsi pesan m. Sedangkan untuk melakukan dekripsi terhadap *ciphertext* c untuk mendapatkan pesan m dapat dilakukan komputasi sebagai berikut :

$$c(\mathbf{P}^{-1}\mathbf{G}^k\mathbf{S}^{-1}) = m\mathbf{G}'(\mathbf{P}^{-1}\mathbf{G}^k\mathbf{S}^{-1}) = m\mathbf{SGP}(\mathbf{P}^{-1}\mathbf{G}^k\mathbf{S}^{-1}) = m.$$

Jadi di sini matriks **R** = $\mathbf{P}^{-1}\mathbf{G}^k\mathbf{S}^{-1}$ berfungsi sebagai kunci pribadi untuk melakukan dekripsi terhadap *ciphertext* c sehingga mendapatkan pesan m. Secara singkat, sistem enkripsi dan dekripsi dari disain yang baru tersebut dapat disajikan dalam Tabel 1 berikut ini.

Tabel 1. Sistem enkrispi-dekripsi

Publik	**G**'
Pribadi	**R**
Enkripsi	c = m**G**'
Dekripsi	m = cR

Prosedur untuk mendapatkan kunci publik **G**' dan kunci pribadi R dapat dijelaskan sebagai berikut :

1.Pilih matriks generator **G** = [I_k A], dengan sembarang matriks **A** berdimensi *k*x*(n-k)*.
2.Pilih sembarang matriks nonsingular **S** berdimensi *k*x*k*.
3.Pilih sembarang matriks permutasi **P** berdimensi *n*x*n*.
4. G' = **SGP.**
5. Cari invers kanan dari **G**, yaitu $\mathbf{G}^k$.
6. Cari invers dari **S**, yaitu $\mathbf{S}^{-1}$.
7. Cari invers dari **P**, yaitu $\mathbf{P}^{-1}$.
8. R = $\mathbf{P}^{-1}\mathbf{G}^k\mathbf{S}^{-1}$.

Berdasarkan disain kriptosistem kunci publik dengan model enkripsi dan dekripsi seperti disajikan dalam Tabel 1 tersebut di atas, selanjutnya dibuat model autentikasi antara pengirim pesan dan penerima pesan. Andaikan user A akan mengirimkan pesan m dengan panjang *k* kepada user B. Pertama-tama user A harus membangun kunci publik **G**'(*k*,*n*) dan kunci pribadi R(*n*,*k*). Prosedur untuk membangun kunci publik **G**' dan kunci pribadi R sebagai berikut :

1.Pilih matriks generator **G** = [I_k **A**], dengan sembarang matriks **A** berdimensi *k*x*(n-k)*.
2.Pilih sembarang matriks nonsingular **S** berdimensi *k*x*k*.
3.Pilih sembarang matriks permutasi **P** berdimensi *n*x*n*.
4.G' = **SGP.**
5.Cari invers kanan dari **G**, yaitu $\mathbf{G}^k$.
6.Cari invers dari **S**, yaitu $\mathbf{S}^{-1}$.
7.Cari invers dari **P**, yaitu $\mathbf{P}^{-1}$.
8.R = $\mathbf{P}^{-1}\mathbf{G}^k\mathbf{S}^{-1}$.

Sedangkan bagi user B juga harus membangun kunci publik H(*n*,*w*) dan kunci pribadi F(*w*,*n*). Dalam hal ini nilai *n* yang dipakai user B ini harus sama dengan nilai *n* yang digunakan user A. Dengan demikian berlaku hubungan *knw*. Prosedur untuk membangun kunci publik H dan kunci pribadi F bagi user B dapat dijelaskan sebagai berikut :

1.Pilih matriks generator **U** = [I_n **B**], dengan sembarang matriks **B** berdimensi *n*x*(w-n)*.
2.Pilih sembarang matriks nonsingular **Q** berdimensi *n*x*n*.
3.Pilih sembarang matriks permutasi **V** berdimensi *w*x*w*.
4.H = **QUV.**
5.Cari invers kanan dari **U**, yaitu $\mathbf{U}^k$.
6.Cari invers dari **Q**, yaitu $\mathbf{Q}^{-1}$.
7.Cari invers dari **V**, yaitu $\mathbf{V}^{-1}$.
8.F = $\mathbf{V}^{-1}\mathbf{U}^k\mathbf{Q}^{-1}$.

Berdasarkan kunci-kunci yang dimiliki user A dan user B tersebut, selanjutnya user A dapat melakukan enkripsi pesan m untuk dikirim kepada user B. Enkripsi pesan yang dilakukan oleh user A ada dua tahap. Tahap pertama dienkripsi dulu menggunakan kunci pribadi milik user A, yaitu kunci R. Kemudian hasil enkripsi tahap pertama tersebut dienkripsi lagi menggunakan kunci public milik user B, yaitu H. Jadi dalam hal ini enkripsi yang dilakukan oleh user A adalah :

1) $m\mathbf{R}T = m\,(\mathbf{P}^{-1}\mathbf{G}^{k}\mathbf{S}^{-1})^{T}$
$= m\,(\mathbf{S}^{-1})^{T}(\mathbf{G}^{k})^{T}(\mathbf{P}^{-1})^{T}$
$= y$

2) $y\mathbf{H} = m\,(\mathbf{S}^{-1})^{T}(\mathbf{G}^{k})^{T}(\mathbf{P}^{-1})^{T}\,\mathbf{Q\,U\,V}$
$= c$

selanjutnya *ciphertext* c ini dikirimkan kepada user B.

Tabel 2. Sistem enkripsi-dekripsi dari skema autentikasi

	User A	*User* B
Publik	**G**	**H**
Pribadi	**R**	**F**
Enkripsi	**1)** $\mathbf{y} = \mathbf{mR}^{T}$ **2)** $\mathbf{c} = \mathbf{yH}$	-
Dekripsi	-	**1)** $\mathbf{y} = \mathbf{cF}$ **2)** $\mathbf{m} = \mathbf{y(G')}^{T}$

Pada saat user B menerima *ciphertext* **c**, untuk dapat membaca pesan **m**, maka user B juga perlu melakukan dekripsi dalam dua tahap. Dekripsi tahap pertama menggunakan kunci pribadi milik user B, yaitu **F**. Sedangkan dekripsi tahap kedua menggunakan kunci publik milik user A, yaitu **G'**. Jadi prosedur untuk melakukan dekripsi terhadap *ciphertext* c oleh user B adalah :

1) $\mathbf{cF} = \mathbf{m}\,(\mathbf{S}^{-1})^{T}(\mathbf{G}^{k})^{T}(\mathbf{P}^{-1})^{T}\mathbf{Q\,U\,V}\,\mathbf{V}^{-1}\mathbf{U}^{k}\mathbf{Q}^{-1}$
$= \mathbf{m}\,(\mathbf{S}^{-1})^{T}(\mathbf{G}^{k})^{T}(\mathbf{P}^{-1})^{T}$
$= \mathbf{y}.$

2) $\mathbf{y(G')}^{T} = \mathbf{m}(\mathbf{S}^{-1})^{T}(\mathbf{G}^{k})^{T}(\mathbf{P}^{-1})^{T}(\mathbf{SGP})^{T}$
$= \mathbf{m}(\mathbf{S}^{-1})^{T}(\mathbf{G}^{k})^{T}(\mathbf{P}^{-1})^{T}\mathbf{P}^{T}(\mathbf{G})^{T}\mathbf{S}^{T}$
$= \mathbf{m}.$

Sistem enkripsi dan dekripsi dari skema autentikasi tersebut dapat dirangkum seperti dalam Tabel 2.

Karakteristik Disain Autentikasi

Berdasarkan prosedur pembentukan kunci serta skema enkripsi-dekripsi dari disain autentikasi yang telah disajikan di atas, berikut ini akan dikaji beberapa sifat yang dimiliki oleh disain autentikasi dari sistem tersebut. Khususnya akan dibahas tentang ruang kunci (*key space*), kompleksitas enkripsi dan dekripsi (*encryption and decryption complexity*), dan ekspansi pesan (*message expansion*).

Ruang Kunci

Ruang kunci di sini adalah kunci yang dipakai untuk enkripsi maupun untuk dekripsi. Kunci-kunci untuk enkripsi adalah $\mathbf{R}(n,k)$ dari user A dan $\mathbf{H}(n,w)$ dari user B. Matriks kunci **R** memerlukan kn bit, sedangkan matriks kunci **H** memerlukan nw bit. Jadi ruang kunci untuk enkripsi data memerlukan $kn + nw = n\,(k + w)$ bit.

Sedangkan kunci-kunci untuk melakukan dekripsi adalah matriks $\mathbf{F}(w,n)$ dari user B dan $\mathbf{G'}(k,n)$ dari user A. Matriks kunci **F** memerlukan nw bit, sedangkan matriks kunci **G'** memerlukan kn bit. Secara keseluruhan ruang kunci untuk melakukan dekripsi adalah $nw + kn = n\,(k + w)$ bit. Jadi total ruang kunci yang diperlukan, baik untuk dekripsi maupun enkripsi adalah $n\,(k + w)$ bit + $n\,(k + w)$ bit = $2n\,(k + w)$ bit.

Kompleksitas Enkripsi dan Dekripsi

Kompleksitas enkripsi dan dekripsi dihitung berdasarkan banyaknya operasi hitung, baik perkalian maupun penjumlahan, yang diperlukan pada kedua proses tersebut. Pada proses enkripsi diperlukan dua tahapan pengerjaan. Pada tahap pertama adalah menghitung $\mathbf{y} = \mathbf{mR}^{T}$. Pesan **m** berdimensi $1x\!k$ dan matriks kunci $\mathbf{R}^{T}$ berdimensi $k\!x\!n$. Karenanya $\mathbf{mR}^{T}$ memerlukan $(2k - 1)n$ buah operasi. Operasi $\mathbf{mR}^{T}$ menghasilkan array **y** berdimensi $1x\!n$.

Enkripsi tahap kedua dengan melakukan perhitungan $\mathbf{c} = \mathbf{yH}$. Pada operasi **yH**, di mana **y** berdimensi $1x\!n$ dan **H** berdimensi $n\!x\!w$, memerlukan $(2n\text{-}1)w$ buah operasi.

Jadi kompleksitas enkripsi diperoleh dengan jalan menghitung banyaknya operasi, baik dari enkripsi pada tahap pertama maupun enkripsi tahap kedua, yaitu $(2k - 1)n + (2n\text{-}1)w = (k + w)2n - (n + w)$ operasi. Jadi kompleksitas enkripsi termasuk anggota $O(n^2)$.

Sementara itu kompleksitas dekripsi juga dicari melalui banyaknya perhitungan (baik penjumlahan maupun perkalian) dari dua tahapan dekripsi. Tahap pertama menghitung **cF**.

Array **c** berdimensi 1x*w*, sedangkan **F** berdimensi *wxn*, sehingga operasi **cF** memerlukan (2*w* – 1)*n* operasi. Operasi **cF** menghasilkan array ***y*** berdimensi 1x*n*.

Dekripsi tahap kedua, yaitu menghitung **yG'**T. Karena *y* berdimensi 1xn dan **G'**T berdimensi nxk, maka **yG'**T memerlukan (2*n*-1)*k* operasi. Dengan demikian, total operasi yang diperlukan pada dekripsi pesan adalah (2*w* – 1)*n* + (2*n*-1)*k* = (*k* + *w*) 2*n* – (*n* + *k*) buah operasi, yang berarti masuk anggota O (n^2).

Secara umum proses enkripsi dan dekripsi pesan atau data memerlukan operasi sebanyak (*k*+*w*)2*n* – (*n*+*w*) + (*k*+*w*) 2*n* – (*n*+*k*) = (*k* + *w*) 4*n* – (2*n* + *k* – *w*) buah operasi. Ini berarti juga bahwa kompleksitas enkripsi dan dekripsi data masuk anggota O(n2). Atau dengan kata lain, disain autentikasi mempunyai kompleksitas enkripsi dan dekripsi yang rendah. Dengan demikian disain ini dapat digunakan untuk melakukan enkripsi dan dekripsi dengan cepat.

Ekspansi Pesan

Suatu kriptosistem kunci publik dikaakan mempunyai ekspansi pesan jika panjang *ciphertext* lebih panjang dari panjang *plaintext*. Dalam disain ini, *plaintext* **m** mempunyai panjang *k*, sedangkan *ciphertext* **c** mempunyai panjang *w*. Jadi rasio ekspansi pesan adalah *r* = *w*/*k*. Jika nilai *w* cukup besar dibandingkan dengan nilai *k*, maka disain autentikasi ini mempunyai ekspansi pesan yang cukup besar. Dengan kata lain, disain autentiksi ini kurang efisien dalam penggunaan ruang pesan.

Analisis Resiko

Analisis resiko ditekankan pada kemungkinan para penyusup (*intruder*) menemukan pesan **m** dari *ciphertext* **c** yang ditransmisikan oleh user A. Secara teknis, karena metode dekripsi tahap kedua menggunakan kunci publik dari user A, maka keamanan data atau *ciphertext* sangat tergantung dari keamanan kunci pribadi dari user B, yaitu **F**. Ini berarti penyusup dimungkinkan berusaha mencari matriks kunci **F** tersebut. Kemungkinan serangan terhadap sistem ini terutama berhubungan dengan pencarian matriks **F**. Jika matriks **F** diperoleh, maka penyusup akan dapat melakukan dekripsi tahap pertama. Selanjutnya dengan menggunakan kunci publik dari user A, penyusup dapat melakukan dekripsi tahap kedua untuk mendapatkan *plaintext* dari pesan yang dikirimkan.

Pencarian terhadap matriks **F** oleh para *intruder* dimungkinkan melalui :
(1)mencari semua kemungkinan matriks **F**, dan
(2)mencari matriks-matriks untuk membentuk matriks kunci **F**.

Mencari Semua Matriks F yang Mungkin

Matriks **F** berdimensi *n*x*w*. Dalam *field* terhingga Z_2 = {0, 1} banyaknya kemungkinan matriks **F** adalah sama dengan banyaknya cara untuk memilih anggota-anggota **F**. Karena tiap entri dari matriks **F** hanya dapat diisi bilangan 0 atau 1 (yang berarti ada dua kemungkinan), sedangkan matriks **F** berdimensi *n*x*w* berarti mempunyai *nw* anggota, maka banyaknya kemungkinan matriks **F** adalah 2^{nw}.

Suatu jumlah yang sangat besar untuk segera menemukan sebuah matriks kunci **F**. Dengan demikian, jika *n* dan *w* cukup besar, probabilitas untuk mendapatkan sebuah matriks kunci **F** adalah, suatu probabilitas yang sangat kecil untuk memperoleh sebuah matriks kunci **F**. Dengan kata lain, *intruder* akan kesulitan untuk menemukan matriks kunci **F**, yang berarti *ciphertext* **c** aman dari gangguan para penyusup.

Mencari Matriks F Berdasarkan Matriks-matriks Pembentuknya

Menurut prosedur pembentukan kunci,
$\mathbf{F} = \mathbf{V}^{-1}\mathbf{U}^{k}\mathbf{Q}^{-1}$.
Karena matriks **U** dalam bentuk standard, yaitu $\mathbf{U} = [\mathbf{I}_n\ \mathbf{B}]$, yang berarti **F** dapat diperoleh jika matriks **V** dan matriks **Q** diperoleh. Jika ***V*** dan **Q** diperoleh, maka $\mathbf{V}^{-1}$ dan $\mathbf{Q}^{-1}$ dapat ditemukan, yang pada gilirannya matriks kunci $\mathbf{F} = \mathbf{V}^{-1}\mathbf{U}^{k}\mathbf{Q}^{-1}$ ditemukan.

Matriks nonsingular **Q** berdimensi *n*x*n*. Pada Z_2 ={0,1}, banyaknya matriks nonsingular **Q** tersebut adalah :
$(2^n - 2^0)\ (2^n - 2^1)\ (2^n - 2^2)\ .\ .\ .\ (2^n - 2^{n-1})$
(Macwilliams and Sloane, 1993:231).

Sebagai gambaran jika **Q**(*5*,*5*), maka banyaknya matriks nonsingular **Q** yang mungkin adalah : (31)(30)(28)(24)(16) = 9999360 buah, suatu jumlah yang sangat besar. Jadi, jika *n* ini cukup besar, maka metode serangan untuk mencari matriks **Q** ini juga tidak praktis.

Demikian halnya matriks permutasi **V** berdimensi *w*x*w*. Matriks permutasi **V** ini berasal dari matriks identitas $\mathbf{I}_w$ dengan jalan melakukan permutasi terhadap baris-barisnya. Dalam hal ini banyaknya permutasi untuk memperoleh matriks **V** adalah $w(w\text{-}1)(w\text{-}2) \ldots 3 .2.1 = w!$. Jika *w* cukup besar, maka *w*! juga sangat besar, berarti sangat sulit untuk mendapatkan matriks permutasi **V**.

Kesimpulannya, karena matriks **V** dan matrik **Q** sulit ditemukan, *intruder* pun tidak dapat menemukan matriks kunci **F**. Dengan kata lain, *intruder* tetap tidak dapat membongkar *ciphertext* **c** untuk mendapatkan pesan **m**.

D. KESIMPULAN DAN SARAN

Tulisan ini telah menunjukkan bahwa teori matriks dapat digunakan untuk mengembangkan ilmu komputer, khususnya dalam bidang penyandian data pada sistem komunikasi data. Dari pembahasan di atas dapat disimpulkan bahwa :

1)Disain autentikasi pada pengiriman pesan **m** dari user A ke user B adalah :
Kunci Publik : **G', H**
Kunci Pribadi : **R, F**
Enkripsi : (1) $\mathbf{y = mR^T}$
(2) **c = yH**
Dekripsi : (1) **y = cF**
(2) $\mathbf{m = y\,(G')^T}$
Dalam disain tersebut **G'** dan **R** berturut-turut adalah kunci publik dan kunci pribadi milik user A. Sedangkan **H** dan **F** beturut-turut adalah kunci publik dan kunci pribadi milik user B.

2)Untuk menjamin autentikasi, user A melakukan dua tahapan enkripsi, yang pertama pesan **m** dienkripsi dengan kunci **G'**, hasilnya dienkripsi lagi menggunakan kunci **H**, sehingga diperoleh*ciphertext* **c**. Selanjutnya *ciphertext* **c** dikirimkan ke user B.

3)Untuk dapat membaca *ciphertext* **c**, user B melakukan dua tahapan dekripsi, yang pertama didekripsi menggunakan kunci **F**, selanjutnya didekripsi lagi menggunakan kunci **G'**, sehingga diperoleh pesan **m**.

4)Disain autentikasi yang diajukan juga cukup efisien, kaitannya dengan penggunaan ruang kunci dan kompleksitas enkripsi dan dekripsi, yaitu anggota O (n^2). Ini artinya, disain autentikasi ini cukup mudah dan cepat dalam melakukan enkripsi dan dekripsi data. Sementara kelemahan dari sistem ini adalah mempunyai ekspansi pesan untuk setiap data yang dikirimkannya.

5)Dengan mengambil dimensi matriks kunci **F** yang cukup besar, disain autentikasi ini dapat dikatakan aman dari kemungkinan serangan para *intruder*.

Penelitian lanjutan yang dapat disarankan dari tulisan ini adalah dalam bidang implementasi disain autentikasi untuk pengamanan data. Ini perlu dilakukan untuk mengetahui lebih jauh keandalan atau keamanan *ciphertext* dari kemungkinan serangan para penyusup (*intruder*). Penelitian lain yang juga dapat dikembangkan adalah dalam model *key agreement* atau persetujuan kunci antara pengirim dan penerima pesan.

E. DAFTAR PUSTAKA

Munir, Rinaldi. 2003. Diktat Kuliah IF2153 Matematika Diskrit *Edisi Keempat.* Agustus 2003. Bandung : Penerbit ITB.

Lee, N.Y., 1999. Security of Shao's Signature Schemes Based on Factoring and Discrete Logarithms *dalam IEE Proceedings Computer Digit. Tech.* **Vol. 146 No. 2**, March 1999, Pp:119-124.

MacWilliams, F.J., and Sloane, N.J.A., 1993. *The Theory of Error Correcting Codes*, Eight Impression, Amsterdam : North-Holland Mathematical Library.

Murtiyasa, B., 2001. Aplikasi Matriks Invers Tergeneralisasi pada Kriptografi *dalam Jurnal Penelitian Sain dan Teknologi* **Vol. 1 Nomor 2** Februari 2001. Hal. 82-90. Surakarta : Lembaga Penelitian UMS.
http://www.unej.ac.id/fakultas/mipa/
Tanggal Akses : 26 Desember 2006 pukul 18.30.

Patterson, W., 1987. *Mathematical Cryptology for Computer Scientists and Mathematicians,* New Jersey : Rowman & Littlefield Publisher.
http:// www.powells.com/cgi-bin
Tanggal Akses : 26 Desember pukul 19.00.

Shao, Z., 1998. Signature Schemes Based on Factoring and Descrete Logarithms *dalam IEE Proceedings Computer Digit. Tech.* **Vol. 145 No. 1**, January 1998, Pp:33-36.

Simmons, G.J.(ed.), 1993. *Contemporary Cryptology : The Science of Information Integrity*, New York : IEEE Press.
http:// portal.acm.org
Tanggal Akses : 27 Desember pukul 18.15

Stalling, W., 1995. *Network and Internetwork Security Principles and Practice*, New Jersey : Prentice Hall.
http://williamstallings.com
Tanggal Akses : 27 Desember pukul 18.00

Stinson, D. R., 1995. *Cryptography Theory and Practice*, Florida : CRC Press LLC.

Vanstone, S.A., and Oorscot, V.P.C., 1989. *An Introduction to Error Correcting Code with Applications*, Kluwer Academic Publisher.

Wu, C.K., and Dawson, E., 1998. Generalised Inverses in Public Key Cryptosystem Design *dalam IEE Proceedings Computer Digit. Tech.* **Vol. 145 No. 5**, September 1998, Pp:321-326.