IMPLEMENTASI PROTOKOL SECRET SPLITTING DENGAN FUNGSI HASH BERBASIS LATTICE PADA NOTARIS DIGITAL

Wahyu Indah Rahmawati[1)], Sandromedo Christa Nugroho[2)]
[1.2)] Lembaga Sandi Negara
e-mail : wahyu.indah@lemsaneg.go.id[1)], sandromedo.christa@lemsaneg.go.id[2)]

Abstrak
Perkembangan ilmu pengetahuan dan teknologi (IPTEK) di era globalisasi mendorong perubahan pada aspek-aspek kehidupan dan cara hidup manusia. Salah satunya adalah Notaris selaku pejabat umum yang berwenang untuk membuat akta otentik. Akta dan surat yang dibuat oleh Notaris nantinya dalam bentuk dokumen elektronik sebagai dokumen resmi bersifat otentik memerlukan pengamanan baik terhadap akta itu sendiri maupun terhadap isinya untuk mencegah penyalahgunaan secara tidak bertanggung jawab. Salah satu solusi untuk mengamankan informasi rahasia tersebut adalah dengan menggunakan protokol kriptografi. Pada penelitian ini akan mengimplementasikan Protokol Secret Splitting yang telah dimodifikasi pada notaris digital yang dikhususkan untuk mengatasi permasalahan warisan. Modifikasi yang dilakukan yaitu dengan menggunakan fungsi hash LASH, dimana algoritma ini merupakan salah satu jenis algoritma fungsi hash berbasis lattice. Adapun simulasi yang dibuat dengan menggunakan bahasa pemrograman MATLAB. Protokol yang dirancang memiliki beberapa aspek-aspek keamanan, diantaranya yaitu surat wasiat hanya dapat dibacakan jika telah dihadiri oleh seluruh Ahli Waris dan Notaris dapat memastikan keotentikan setiap Ahli Waris. Oleh karena itu, Protokol Secret Splitting dapat dimanfaatkan pada jasa pelayanan notaris digital dalam hal menjamin setiap entitas (yang sah) memiliki kontribusi dalam merekonstruksi pesan asli.

***Kata Kunci** : Ilmu Pengetahuan dan Teknologi, Notaris Digital, Informasi Rahasia, Protokol Secret Splitting, Algoritma Fungsi Hash LASH.*

1. PENDAHULUAN

Informasi merupakan hal yang sangat penting saat ini, dimana tanpa informasi kita akan serba ketinggalan. Pada jaman modern sekarang ini, masyarakat banyak memperoleh informasi melalui pemanfaatan teknologi informasi dan komunikasi yang terkoneksi dengan internet. Perkembangan ilmu pengetahuan dan teknologi (IPTEK) yang memanfaatkan internet, akan berpengaruh terhadap pelaksanaan tugas dan wewenang bagi beberapa praktisi hukum. Salah satunya yaitu Notaris selaku pejabat umum yang berwenang untuk membuat akta otentik, yang pada awalnya menggunakan cara konvesional dalam pembuatan akta otentik menuju arah jasa pelayanan notaris secara elektronik melalui internet sebagai media utama dalam kinerjanya untuk membuat suatu akta notaris dan mengarah kepada bentuk akta yang awalnya sah apabila tertuang dalam kertas menuju akta secara elektronik atau dalam bentuk dokumen elektronik. Akta dan surat yang dibuat oleh Notaris sebagai dokumen resmi bersifat otentik memerlukan pengamanan baik terhadap akta itu sendiri maupun terhadap isinya untuk mencegah penyalahgunaan secara tidak bertanggung jawab. Salah satu solusi untuk mengamankan informasi rahasia tersebut adalah dengan menggunakan protokol kriptografi. Penggunaan protokol kriptografi dapat memberikan jaminan keamanan informasi yang dipertukarkan.

Protokol kriptografi yang dapat diterapkan pada notaris digital yaitu Protokol *Secret Splitting*. Pada penelitian ini akan mengimplementasikan Protokol *Secret Splitting* yang telah dimodifikasi pada notaris digital yang dikhususkan untuk mengatasi permasalahan warisan. Modifikasi yang dilakukan yaitu dengan menggunakan fungsi hash. Algoritma fungsi hash yang digunakan yaitu Algoritma Fungsi Hash LASH, dimana algoritma ini merupakan salah satu jenis algoritma fungsi hash berbasis *lattice*. Adapun simulasi yang dibuat dengan menggunakan bahasa pemrograman MATLAB.

Protokol *Secret Splitting* pada notaris digital dalam penelitian ini merupakan pelayanan jasa notaris menyangkut masalah wasiat secara elektronik. Jika dianalogikan dengan kehidupan nyata, A, seseorang yang merasa umurnya tidak panjang lagi, , membuat surat wasiat yang isinya telah disahkan oleh Notaris. Setelah A meninggal, Ahli Waris akan mendapatkan isi surat wasiat dengan melibatkan Notaris. Namun, persyaratan yang harus dipenuhi di sini adalah semua Ahli Waris harus hadir di tempat saat pembacaan surat wasiat. Jika tidak, surat wasiat tersebut tidak dapat dibaca.

2. TINJAUAN PUSTAKA

Protokol *Secret Splitting*

Secret Splitting adalah sebuah metode untuk membagi angka-angka (*numbers*), teks, atau data komputer ke dalam dua atau banyak bagian. Suatu informasi diperoleh dari *secret splitting* terpisah yang tidak menyatakan beberapa informasi atau bagian dari informasi asli, dan tidak membantu berbagai cara untuk mendapatkan kembali informasi asli tersebut. Oleh karena itu, keamanan *secret splitting* secara matematika bergantung sepenuhnya selama *splitting* (*share*) tersebut terpisah.

Secret Splitting berguna dalam situasi dimana informasi rahasia haruslah disimpan oleh dua orang atau lebih, tanpa memperlihatkan informasi rahasianya pada seseorang. Semua orang dengan *share* pada informasi rahasianya harus setuju untuk menggabungkan semua *share* untuk mendapatkan kembali informasi asli, dan tidak ada seorangpun yang memperoleh informasi tanpa autorisasi dan bantuan dari semua orang yang memegang sisa dari *share* tersebut. Semakin banyak orang yang memiliki *share* berarti akan semakin aman, karena dengan semakin banyak orang yang setuju untuk meletakkan *share* secara bersama-sama.
Berikut adalah protokol dimana Trent dapat membagi pesan antara Alice dan Bob:
(1) Trent membangkitkan bit string acak, *R*; panjangnya sama dengan panjang pesan, *M*.
(2) Trent meng-XOR *M* dengan *R* untuk membangkitkan *S*.

$$(M \oplus R = S) \qquad (1)$$

(3) Trent memberikan *R* ke Alice dan *S* ke Bob.
(4) Alice dan Bob Meng-XOR *R* dan *S* untuk mendapatkan pesan, *M*.

$$(R \oplus S = M) \qquad (2)$$

Secret Splitting ini jika dilakukan dengan benar maka akan dapat menjamin keamanan secara mutlak. Teknik yang dilakukan berbasis One Time Pad yang merupakan teknik yang tidak dapat dipecahkan oleh komputasi apapun dengan syarat nilai dari *R* benar-benar acak dan terjaga kerahasiaannya.
Apabila, jumlah pihak yang terlibat lebih dari 2 (dua) orang (misal 4 orang), maka mekanismenya sebagai berikut:
(1) Trent membangkitkan bit string acak *R*, *S*, dan *T*; panjangnya sama dengan pesan, *M*.
(2) Trent meng-XOR *M* dengan ketiga string tersebut untuk membangkitkan *U*.

$$(M \oplus R \oplus S \oplus T = U) \qquad (3)$$

(3) Trent memberikan *R* ke Alice, *S* ke Bob, *T* ke Carol, dan *U* ke Dave.
(4) Untuk merekonstruksi pesan, *M*, Alice, Bob, Carol dan Dave harus menghitung:

$$(R \oplus S \oplus T \oplus U = M) \qquad (4)$$

Fungsi Hash LASH

Algoritma fungsi hash LASH merupakan salah satu keluarga fungsi hash berjenis MDC (algoritma fungsi hash tanpa menggunakan kunci). Hal yang menarik dari algoritma fungsi hash LASH adalah fungsi/algoritma kompresinya melibatkan tranformasi matrik modular, sehingga cukup relevan jika tumpuan keamanannya dapat dikaitkan dengan permasalahan komputasi *Lattice*. Jadi, algoritma fungsi hash LASH dapat digolongkan sebagai algoritma fungsi hash yang *provably secure*.

Algoritma fungsi hash LASH adalah algoritma yang mengompres pesan *M* dengan panjang arbitrary/sembarang, menjadi nilai *hash N* berukuran 8m bit (m *byte*) yang fix/tetap. Algoritma fungsi hash LASH memiliki beberapa parameter input, antara lain *s* dan *m*. Dimana s adalah pesan *M* dengan panjang *arbitrary*/sembarang dan m adalah variasi kelipatan blok pada algoritma fungsi hash LASH. Terdapat beberapa variasi nilai *m* yang distandarkan dan dapat dipilih pada algoritma fungsi hash LASH, yaitu $m = 20$ (algoritma fungsi hash LASH-160 bit), $m = 32$ (algoritma fungsi hash LASH-256 bit), $m = 48$ (algoritma fungsi hash LASH-384 bit), dan $m = 64$ (algoritma fungsi hash LASH-512 bit).

Algoritma fungsi hash LASH memiliki mekanisme penambahan bit secara otomatis (*padding*) untuk memenuhi jumlah kelipatan panjang blok *m* yang telah ditentukan sebelumnya. Dimana, jika

panjang pesan *M* kurang dari panjang kelipatan blok *m* yang telah ditentukan, maka akan ditambahkan *padding* agar panjang pesan *M* menjadi kelipatan blok *m*. *Padding* yang digunakan pada algoritma fungsi hash LASH diawali dengan bit "1" dan selanjutnya adalah bit "000…" sampai dengan panjang kelipatan (blok m–1), sedangkan 1 (satu) byte lagi berisikan informasi mengenai panjang pesan *M* yang dihashkan.

Pada implementasinya, 1 (satu) byte terakhir pada proses padding merupakan informasi mengenai panjang pesan yang di-hash , padding panjang pesan tersebut merupakan padding optional, yang dapat diimplementasikan maupun tidak, tergantung dari kesepakatan implementasi, dan pengguna algoritma fungsi hash LASH. Gambar 1 menunjukkan proses *padding* pada algoritma fungsi hash LASH.

Pesan	100 … 000	X	Panjang Pesan

Gambar 1. Proses *Padding* pada Algoritma Fungsi Hash LASH

Secara umum proses kompresi pada algoritma fungsi hash LASH adalah:

$$f(R, M) = (R \oplus M) + H[R||M] \qquad (5)$$

Untuk memudahkan pembahasan proses kompresi pada algoritma fungsi hash LASH, maka akan dilakukan pembahasan proses kompresi bagian kiri dan bagian kanan.

a. Proses kompresi bagian kiri pada algoritma fungsi hash LASH, ($R \oplus M$)

Proses awalnya menggunakan iterasi inisial vektor R_0 = IV, yaitu matriks yang berisi vektor-vektor 0 = [0, 0, 0, … , s/d m] elemen $\mathbb{Z}_{256}^{m}$. Contohnya, jika m yang dipilih adalah 20 *byte* (160 bit), maka R_0 = [0, 0, 0, … , s/d 20] elemen $\mathbb{Z}_{256}^{20}$. Selanjutnya matriks *R* akan di-*xor binary*-kan dengan pesan *M* yang masing-masing adalah vektor dalam ruang vektor $\mathbb{Z}_{256}^{m}$ dan menghasilkan matriks XL berukuran 1 x 20 dalam ruang vektor $\mathbb{Z}_{256}$.

b. Proses kompresi bagian kanan pada algoritma fungsi hash LASH, $H[R||M]$

Proses awalnya adalah membangkitkan matriks *H*, matriks *H* berukuran mxn elemen $\mathbb{Z}_{256}$, dimana m adalah ukuran baris, dan *n* ukuran kolom pada matriks *H*. Matriks *H* dikonstruksi dari elemen *a*, yaitu :

(1) $y_0 = 54321$ elemen $\mathbb{Z}$

(2) $y_i = y_{(i-1)}2 + 2 \pmod{2^{31}-1}$, dengan $i = 1, \dots, n-1$

(3) $a_i = y_i \pmod{2^8}$, dengan $i = 1, \dots, n-1$

sehingga didapatkan nilai $a = a_0, a_{n-1}, \dots a_2, a_1$ yang kemudian diputar sampai dengan m kali, hingga membentuk matriks

$$H = \begin{bmatrix} a0 & \cdots & a1 \\ \vdots & \ddots & \vdots \\ am-1 & \cdots & am \end{bmatrix} \text{ elemen } \mathbb{Z}_{256}^{mxn}.$$

Setelah didapatkan matriks *H*, kemudian dilakukan pengabungan antara matriks *R*, dan matriks *M*. Matriks *R* elemen $\mathbb{Z}_{256}^{m}$, dibinerkan, dalam hal ini matriks *R* berisi *m* entri yang masing-masing panjangnya 8 bit.

Notaris

Notaris adalah pejabat umum yang berwenang untuk membuat akta otentik dan kewenangan lainnya sebagaimana maksud dalam Undang Undang no 30 tahun 2004 tentang Jabatan Notaris (UUJN). Suatu akta otentik ialah suatu akta didalam bentuk yang ditentukan oleh undang-undang, yang dibuat oleh atau dihadapan pegawai-pegawai umum yang berkuasa untuk itu ditempat dimana akta dibuatnya.

Notaris berwenang membuat akta otentik mengenai semua perbuatan, perjanjian, dan ketetapan yang diharuskan oleh peraturan perundang-undangan dan/ atau yang dikehendaki oleh yang berkepentingan untuk menyimpan akta, memberikan grosse, salinan dan kutipan akta, semuanya itu sepanjang pembuatan akta-akta itu tidak juga ditugaskan atau dikecualikan kepada pejabat lain atau orang lain yang ditetapkan oleh undang-undang. Dalam menjalankan jabatannya, salah satu kewajiban yang harus dilakukan oleh Notaris yaitu membacakan akta di hadapan penghadap dengan dihadiri oleh paling sedikit 2 (dua) orang saksi dan ditandatangani pada saat itu juga oleh penghadap, saksi, dan Notaris (pasal 16 ayat (1) butir l UUJN).

Sebagai alat bukti tertulis yang terkuat dan terpenuh, apa yang dinyatakan dalam akta notaris harus diterima, kecuali pihak yang berkepentingan dapat membuktikan hal yang sebaliknya secara memuaskan di hadapan persidangan pengadilan. Fungsi Notaris di luar pembuatan akta otentik diatur untuk pertama kalinya secara komprehensif dalam UUJN. Demikian pula ketentuan tentang pengawasan terhadap pelaksanaan jabatan Notaris dilakukan dengan mengikutsertakan pihak ahli/akademisi, di samping Departemen yang tugas dan tanggung jawabnya di bidang kenotariatan serta Organisasi Notaris.

3. METODE PENELITIAN

Metode penelitian yang digunakan adalah kajian kepustakaan. Metode penelitian kepustakaan adalah penelitian yang datanya diambil terutama atau seluruhnya dari kepustakaan. Pada penelitian ini metode kajian kepustakaan bertujuan untuk menelaah teori yang berkaitan mengenai prinsip perancangan protokol dari berbagai referensi seperti buku, *electronic book*, tesis, dan kepustakaan lainnya.

Tahapan proses penelitian ini adalah sebagai berikut:

(1) Studi Literatur
Melakukan studi literatur dari beberapa buku atau referensi lain mengenai Protokol *Secret Splitting*, algoritma fungsi hash LASH, dan profesi notaris.
(2) Mendesain Protokol
Melakukan perancangan Protokol *Secret Splitting* sesuai kebutuhan.
(3) Simulasi
Melakukan simulasi terhadap Protokol *Secret Splitting* yang dirancang.
(4) Analisis
Melakukan analisis aspek keamanan pada Protokol *Secret Splitting* yang dirancang.
(5) Pengambilan Kesimpulan
Pengambilan kesimpulan dari hasil penelitian.

4. HASIL DAN PEMBAHASAN

4.1. DESKRIPSI PROTOKOL *SECRET SPLITTING* DENGAN FUNGSI HASH BERBASIS *LATTICE* PADA NOTARIS DIGITAL

Dalam memberikan pelayanan jasanya, notaris digital/elektronik menerapkan protokol *secret splitting* yang dimodifikasi. Modifikasi yang dilakukan yaitu dengan menggunakan Algoritma Fungsi Hash LASH, dimana algoritma ini merupakan salah satu jenis algoritma fungsi hash berbasis *lattice*.

Deskripsi protokol *secret splitting* dengan fungsi hash berbasis *lattice* pada notaris digital dijelaskan sebagai berikut :

a. Fase Sebelum Pewaris Meninggal

(1) Pewaris membuat surat wasiat/pesan, M, dan membangkitkan bilangan acak x, dan y
(2) Pewaris menghitung h(x), h(y) , dan C, dimana persamaan dari C adalah

$$C = x \oplus y \oplus N \oplus M \qquad (6)$$

(3) Pewaris mengirimkan h(x), h(y), dan N ke Notaris
(4) Pewaris mengirimkan x ke Ahli Waris 1 dan y ke Ahli Waris 2

b. Fase Setelah Pewaris Meninggal

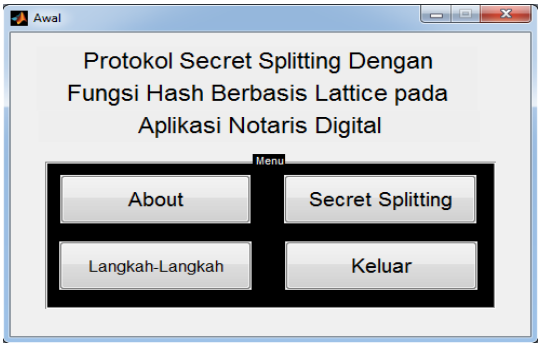
(5) Ahli Waris 1 dan Ahli Waris 2 menyerahkan parameter yang diterima dari Pewaris yaitu x, dan y kepada Notaris
(6) Notaris menghitung h(x) dan h(y), kemudian membandingkan dengan nilai h(x) dan h(y) dari Pewaris. Jika tidak sama kembali ke step sebelumnya
(7) Notaris me*recover* pesan M dengan persamaan

$$M = x \oplus y \oplus N \oplus C \qquad (7)$$

4.2. SIMULASI PROGRAM

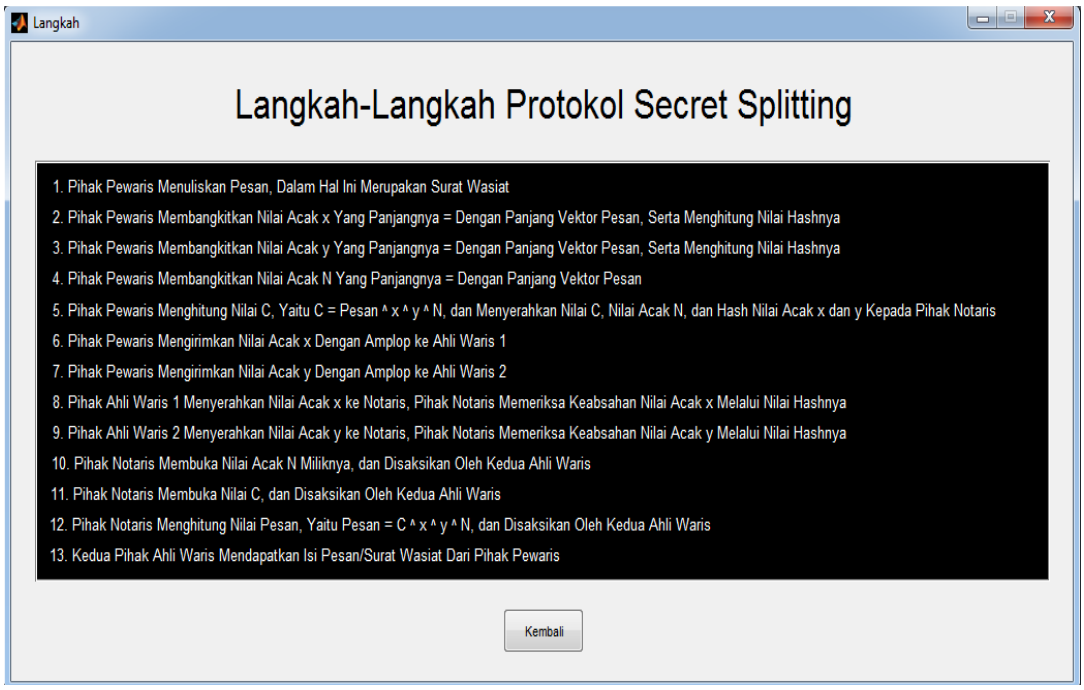
Simulasi Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice* pada notaris digital dibuat dengan menggunakan bahasa pemrograman MATLAB. Simulasi yang dibuat merepresentasikan 4 (empat) entitas yaitu Pewaris sebagai pihak pemberi warisan, Ahli Waris 1 sebagai pihak penerima

warisan 1, Ahli Waris 2 sebagai pihak penerima warisan 2, dan Notaris sebagai pihak ketiga terpercaya. Gambar dibawah menunjukkan *form* menu Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice* pada Notaris Digital.



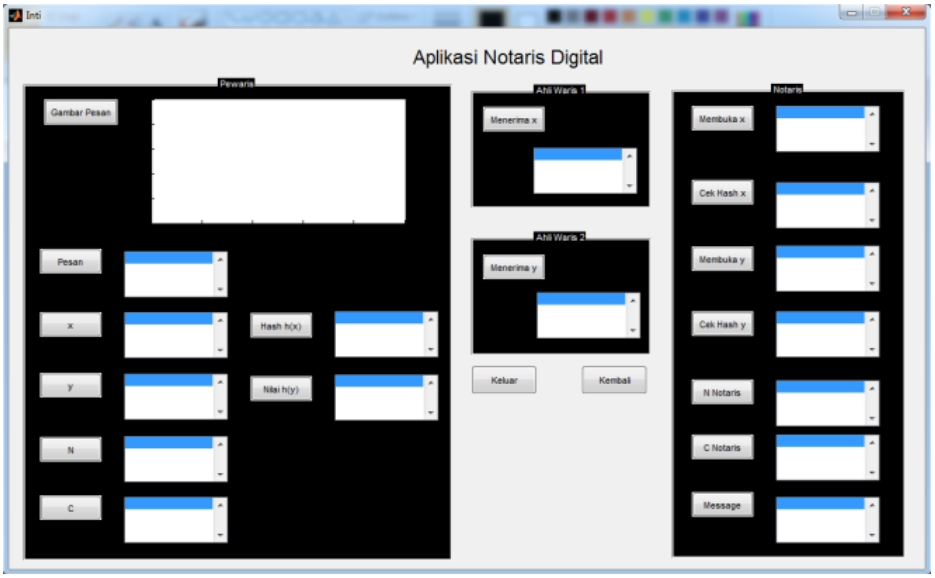
Gambar 2. *Form* Menu Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice*

Gambar dibawah menunjukkan *form* langkah-langkah pada Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice* pada Notaris Digital.



Gambar 3. *Form* Langkah-Langkah pada Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice*

Gambar dibawah menunjukkan *form* utama Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice* pada Notaris Digital



Gambar 4. *Form* Utama Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice*

5. ANALISIS

Sistem informasi yang baik adalah sistem informasi yang efektif dan efisien dalam mendata dan mengolah informasi, serta memiliki aspek-aspek keamanan informasi. Aspek keamanan pada Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice* pada Notaris Digital diantaranya sebagai berikut :

(1) Surat wasiat hanya dapat dibacakan jika telah dihadiri oleh seluruh ahli waris. Hal tersebut dikarenakan penerapan Protokol *Secret Splitting* yang dapat menjamin bahwa setiap entitas (yang sah) memiliki kontribusi dalam merekonstruksi pesan asli. Dimana setiap Ahli Waris dibekali *secret splitting* yang berbeda yang nantinya dikumpulkan untuk merekonstruksi surat wasiat.

(2) Notaris dapat memastikan keotentikan setiap Ahli Waris. Hal ini dapat dilihat ketika Notaris menghitung nilai hash dari *secret splitting* setiap Ahli Waris dan membandingkannya dengan nilai hash setiap Ahli Waris yang diterima dari A. Jika sama, berarti keotentikan Ahli Waris 1 dan Ahli Waris 2 terbukti, para Ahli Waris yang terlibat adalah pihak yang valid. Namun jika ternyata nilai hash yang dihitung tidak sama, maka Ahli Waris 1 dan Ahli Waris 2 diindikasikan sebagai pihak yang tidak sah untuk ikut merekonstruksi surat wasiat.

6. KESIMPULAN

Penelitian ini menghasilkan kesimpulan sebagai berikut :

(1) Protokol *Secret Splitting* dengan Fungsi Hash berbasis *Lattice* pada Notaris Digital merupakan program simulasi pelayanan jasa notaris menyangkut masalah wasiat secara *online*.

(2) Protokol *Secret Splitting* yang diterapkan merupakan protokol *Secret Splitting* yang telah dimodifikasi dengan menggunakan Algoritma Fungsi Hash LASH, yang merupakan salah satu jenis algoritma fungsi hash berbasis *lattice*.

(3) Protokol *Secret Splitting* dapat dimanfaatkan pada jasa pelayanan notaris digital dalam hal menjamin setiap entitas (yang sah) memiliki kontribusi dalam merekonstruksi pesan asli.

DAFTAR PUSTAKA

Desiani, Anita & Arhami, Muhammad. 2005. Pemrograman MATLAB : Andi. Yogyakarta.

Page.,D. Embedded Implementation of LASH. University of Bristol. United Kingdom.

Schneier, Bruce. 1996. Applied Cryptography : Protocol, Algorithms and Source Code in C, John Willey & Sons, Inc.

Undang-Undang Nomor 30 Tahun 2004 tentang Jabatan Notaris (UUJN)