Implementasi Tanda Tangan Digital Pada Teks Menggunakan Pemulihan Pesan

DSA dan Elliptic Curve DSA

Ubaidillah Ariq Prathama - 13520085
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: 13520085@std.stei.itb.ac.id

***Abstrak*—Tanda tangan digital adalah salah satu metode kriptografi yang digunakan secara luas untuk memastikan integritas dan otentikasi pesan digital. Namun, dalam situasi di mana pesan terenkripsi atau ditandatangani hilang atau rusak, pemulihan pesan menjadi tantangan yang signifikan. Makalah ini mempresentasikan pendekatan dalam tanda tangan digital dengan message recovery, yang memungkinkan pemulihan pesan yang hilang atau rusak dengan memanfaatkan informasi tambahan yang terkandung dalam tanda tangan digital.**

Kata Kunci—kunci publik; kunci privat; tanda tangan digital; kurva elliptic; digital signature algorithm;

I. PENDAHULUAN

Pendekatan yang diusulkan menggabungkan algoritma kriptografi kunci publik dan teknik pemulihan pesan untuk mencapai tanda tangan digital dengan kemampuan pemulihan pesan. Pertama, pesan asli dienkripsi dengan menggunakan kunci publik penerima, dan kemudian ditandatangani oleh pengirim menggunakan kunci pribadi pengirim. Tanda tangan digital yang dihasilkan menyimpan informasi tambahan yang memungkinkan pemulihan pesan asli jika terjadi kehilangan atau kerusakan pada pesan terenkripsi.

Metode pemulihan pesan dilakukan dengan memanfaatkan teknik rekonstruksi pesan. Ketika pesan terenkripsi hilang atau rusak, penerima dapat menggunakan informasi tambahan dalam tanda tangan digital untuk memulihkan pesan asli secara akurat. Selain itu, metode ini juga menyediakan mekanisme verifikasi autentikasi yang kuat, sehingga penerima dapat memverifikasi bahwa pesan asli belum diubah dan ditandatangani oleh pengirim yang sah.

Melalui analisis keamanan dan uji coba komputasional, paper ini menunjukkan bahwa pendekatan tanda tangan digital dengan pemulihan pesan yang diusulkan dapat memberikan tingkat keandalan dan keamanan yang cukup tinggi. Walaupun keamanannya tidak setinggi tanda tangan digital tanap pemulihan. Pendekatan ini juga memiliki efisiensi komputasional yang cukup baik dan lebih hemat memori, memungkinkan implementasi praktis dalam aplikasi keamanan komunikasi yang memerlukan pemulihan pesan.

Dengan demikian, tanda tangan digital dengan pemulihan pesan ini menawarkan solusi yang inovatif dan efektif untuk memastikan integritas dan otentikasi pesan digital dalam situasi di mana pesan terenkripsi hilang atau rusak. Pendekatan ini memiliki potensi untuk digunakan dalam berbagai aplikasi, termasuk komunikasi aman dan penyimpanan pesan digital yang andal.

II. DASAR TEORI

A. Kriptografi Kunci Publik

Kriptografi kunci publik, juga dikenal sebagai kriptografi asimetris, adalah cabang dari kriptografi yang melibatkan penggunaan sepasang kunci yang terdiri dari kunci publik dan kunci pribadi. Konsep kunci publik dan kunci pribadi ini diperkenalkan oleh Whitfield Diffie dan Martin Hellman pada tahun 1976, dan telah menjadi dasar dari banyak protokol keamanan modern.

1. Kunci Publik dan Kunci Privat

   Kriptografi kunci publik melibatkan penggunaan dua kunci yang berbeda, kunci publik dan kunci privat. Kunci publik adalah kunci yang dapat dipublikasikan dan digunakan oleh siapa saja untuk melakukan enkripsi pesan. Kunci privat, di sisi lain, harus dijaga kerahasiaannya dan hanya diketahui oleh pemiliknya. Kunci privat digunakan untuk dekripsi pesan yang dienkripsi menggunakan kunci publik yang sesuai.

2. Enkripsi dan Dekripsi

   Dalam kriptografi kunci publik, pesan asli dienkripsi menggunakan kunci publik penerima, sementara dekripsi dilakukan menggunakan kunci pribadi penerima. Proses enkripsi melibatkan transformasi pesan asli menjadi format yang tidak dapat dibaca atau dipecahkan oleh pihak yang tidak berwenang. Dekripsi, di sisi lain, adalah proses mengembalikan pesan ke bentuk aslinya menggunakan kunci pribadi yang sesuai.

3. Keamanan Kriptografi Kunci Publik

   Keamanan kriptografi kunci publik didasarkan pada masalah matematis yang sulit dipecahkan, seperti

faktorisasi bilangan besar atau permasalahan logaritma diskret. Kunci publik dapat dipublikasikan secara luas tanpa mengorbankan keamanan, karena kunci pribadi yang sesuai harus tetap dirahasiakan. Dengan menggunakan algoritma kriptografi yang kuat dan kunci yang cukup panjang, komunikasi aman dapat dicapai dengan menggunakan kriptografi kunci publik.

4. Aplikasi Kriptografi Kunci Publik

   Kriptografi kunci publik memiliki banyak aplikasi dalam keamanan komunikasi dan keamanan informasi. Beberapa contoh termasuk pertukaran kunci aman, tanda tangan digital, enkripsi email, otentikasi pengguna, dan banyak lagi. Protokol kriptografi kunci publik yang populer termasuk RSA (Rivest-Shamir-Adleman), Diffie-Hellman, ElGamal, dan ECC (Elliptic Curve Cryptography).

Dalam paper ini, kriptografi kunci publik digunakan sebagai dasar untuk membangun tanda tangan digital dengan kemampuan message recovery. Melalui penggunaan kunci publik dan kunci pribadi, pesan dapat dienkripsi, ditandatangani, dan pemulihan pesan dapat dilakukan dengan memanfaatkan informasi tambahan dalam tanda tangan digital.

B. Tanda Tangan Digital

Tanda tangan digital adalah metode kriptografi yang digunakan untuk mengotentikasi dan memverifikasi integritas pesan digital. Dalam konteks tanda tangan digital, pengirim pesan menggunakan kunci pribadi mereka untuk menghasilkan tanda tangan digital yang terkait dengan pesan tersebut. Penerima pesan kemudian menggunakan kunci publik pengirim untuk memverifikasi tanda tangan dan memastikan bahwa pesan tidak diubah selama pengiriman. Berikut adalah dasar-dasar teori yang relevan untuk pemahaman tanda tangan digital:

1. Fungsi Hash

   Fungsi hash adalah fungsi matematis yang mengonversi pesan dengan panjang variabel menjadi nilai hash tetap. Fungsi hash harus menghasilkan nilai hash yang unik untuk setiap pesan yang berbeda dan harus memperlihatkan sifat kebalikan yang kuat. Dalam konteks tanda tangan digital, fungsi hash digunakan untuk menghasilkan ringkasan pesan yang kemudian akan ditandatangani.

2. Kunci Pribadi dan Kunci Publik

   Tanda tangan digital menggunakan kriptografi kunci publik, yang melibatkan penggunaan sepasang kunci yang berbeda: kunci pribadi dan kunci publik. Kunci pribadi hanya diketahui oleh pemiliknya dan digunakan untuk membuat tanda tangan digital, sementara kunci publik dapat dipublikasikan dan digunakan untuk memverifikasi tanda tangan digital. Hubungan matematis yang rumit memastikan bahwa tanda tangan digital yang dihasilkan hanya dapat diproduksi oleh pemilik kunci pribadi yang sesuai.

3. Algoritma Kriptografi

   Berbagai algoritma kriptografi digunakan dalam implementasi tanda tangan digital. Salah satu algoritma yang umum digunakan adalah RSA (Rivest-Shamir-Adleman). RSA memanfaatkan operasi matematika pada angka besar, seperti operasi perkalian modular dan eksponensial modular, untuk menghasilkan tanda tangan digital yang unik.

4. Validasi dan Verifikasi

   Untuk memverifikasi tanda tangan digital, penerima menggunakan kunci publik pengirim untuk mendekripsi tanda tangan digital dan memperoleh nilai hash pesan asli. Penerima kemudian menggunakan fungsi hash yang sama untuk menghasilkan nilai hash dari pesan yang diterima. Jika nilai hash yang dihasilkan sesuai dengan nilai hash yang didekripsi dari tanda tangan digital, maka tanda tangan digital dianggap valid dan pesan dianggap tidak diubah.

5. Keamanan

   Keamanan tanda tangan digital bergantung pada keamanan kriptografi kunci publik yang digunakan. Kunci pribadi harus tetap rahasia dan dilindungi dengan baik agar hanya pemiliknya yang dapat menghasilkan tanda tangan digital yang valid. Keamanan juga terkait dengan kekuatan fungsi hash yang digunakan, sehingga memastikan bahwa sulit bagi pihak yang tidak berwenang untuk menghasilkan tanda tangan digital palsu.

C. Digital Signature Algorithm (NIST)

Algoritma DSA adalah sebuah algoritma kriptografi kunci publik yang dapat digunakan untuk melakukan tanda tangan digital. Algoritma ini didasarkan pada kesulitan memecahkan permasalahan logaritma diskret dalam bidang bilangan bulat modulo.

Proses penandatanganan menggunakan bilangan acak k untuk memperoleh keunikan setiap kali tanda tangan digital dihasilkan. Selain itu, algoritma ini juga dapat memberikan anonimitas, karena tidak membeberkan kunci pribadi saat melakukan penandatanganan. Algoritma DSA memberikan alternatif yang kuat dan aman untuk tanda tangan digital. Namun, implementasinya memerlukan perhatian khusus untuk menghindari serangan terhadap bilangan acak K yang digunakan dan untuk memastikan kekuatan keamanan algoritma.

Berikut adalah penjelasan tentang algoritma tanda tangan digital menggunakan algoritma DSA:

Key Generation

1. Pilih bilangan prima p dan q sehingga $(p - 1)\ mod\ q = 0$

2. Menghitung $g = h^{\frac{p-1}{q}} \, mod \, p$, yang dalam hal ini $1 < h < p-1$ dan $h^{\frac{p-1}{q}} \, mod \, p > 1$
3. Tentukan kunci privat x, yang dalam hal ini $x < q$
4. Hitung kunci public $y = g^x \, mod \, p$
5. Kunci privat = x, kunci publik = (p, q, g, y)

Signing

1. Hitung message digest pesan m dengan fungsi hash SHA-1, $H(m)$.
2. Tentukan bilangan acak $k < q$.
3. Tanda tangan dari pesan m adalah bilangan r dan s. Hitung r dan s sebagai berikut:

$$r = (g^k \, mod \, p) \, mod \, q$$

$$s = \left(k^{-1}(H(m) + x \cdot r)\right) mod \, q$$

4. Kirim pesan m beserta tanda tangan r dan s

Verifying

1. Hitung message digest pesan m dengan fungsi hash SHA-1, $H(m)$.
2. Menghitung $w = s^{-1} \, mod \, q$
3. Menghitung $u_1 = (H(m) \cdot w) \, mod \, q$
4. Menghitung $u_2 = r \cdot w \, mod \, q$
5. Menghitung $v = \left((g^{u_1} \cdot y^{u_2}) \, mod \, p\right) mod \, q)$
6. Menerima tanda tangan jika dan hanya jika v = r

D. Elliptic Curve Digital Signature Algorithm

ECC adalah sistem kriptografi kunci-publik. Setiap pengguna memiliki kunci publik dan kunci privat. Kunci publik digunakan untuk enkripsi atau untuk verifikasi tanda tangan digital. Sedangkan, kunci privat digunakan untuk dekripsi atau untuk menghasilkan tanda tangan digital. ECDSA merupakan salah satu perluasan implementasi dari ECC. Bagian inti dari sistem kriptografi kunci-publik yang melibatkan kurva eliptik adalah grup eliptik (himpunan titik-titik pada kurva eliptik dan sebuah operasi biner +). Keamanan algoritma kurva elliptic didasari permasalahan logaritma diskrit pada kurva elliptic.

Dalam protokol ECDSA, pihak yang akan melakukan tanda tangan digital, mempunyai parameter domain kurva eliptik berupa Curve = $\{p, a, b, G, n\}$ dan pasangan kunci kunci rahasia d_A dan kunci publik Q_A. Kemudian pihak yang akan melakukan verifikasi terhadap tanda tangan, memiliki salinan dokumen m yang otentik dan kunci publik Q_A. Proses-proses yang terjadi adalah sebagai berikut :

Key Generation

1. Memilih sebuah bilangan bulat random d_A, yang nilainya diantara $[1, n-1]$
2. Menghitung $Q_A = d_A \cdot G = (x_A, y_A)$
3. Kunci rahasia = d_A, dan kunci publik = Q_A.

Signing

1. Memilih sebuah bilangan bulat random k, yang nilainya diantara $[1, n-1]$.
2. Menghitung $Q_A = k \cdot G = (x_1, y_1)$
3. Menghitung $r = x_1 \, mod \, n$, jika $r = 0$, maka kembali ke langkah 1.
4. Menghitung $k - 1 \, mod \, n$
5. Menghitung $e = H(m)$
6. Menghitung $s = (k-1)\,(e + d_A \cdot r) \, mod \, n$ tanda tangan Alice untuk message m adalah (r, s)

Verifying

1. Memverifikasi bahwa r dan s adalah bilangan bulat yang antara $[1, n-1]$
2. Menghitung $e = H(m)$
3. Menghitung $w = s - 1 \, mod \, n$
4. Menghitung $u_1 = e \cdot w \, mod \, n$ dan $u_2 = r \cdot w \, mod \, n$
5. Menghitung $u_1 \cdot G + u_2 \cdot Q_A = (x_1, y_1)$
6. Menghitung $v = x_1 \, mod \, n$
7. Menerima tanda tangan jika dan hanya jika $v = r$

III. Modifikasi Algoritma Tanda Tangan Digital

Terdapat beberapa algoritma tanda tangan digital yang umum digunakan. Algoritma tersebut antara lain DSA, RSA, ElGamal, dan Elliptic Curve. Dari berbagai algoritma tersebut, RSA memiliki keunikan yaitu dapat digunakan dengan text hashing ataupun message recovery. Sedangkan, algoritma lainnya hanya memungkinkan untuk tanda tangan dengan menggunakan text hashing. Oleh karena itu, akan dibuat tanda tangan digital dengan algoritma DSA dan ECDSA dengan message recovery. Kelebihan dari metode ini adalah tidak memerlukan fungsi hash dan memerlukan bandwidth yang lebih kecil. Akan tetapi, algoritma ini tidak dapat digunakan untuk enkripsi karena dilakukan pertukaran role dari kunci privat dan kunci publik.

A. DSA Dengan Pemulihan Pesan

Pada modifikasi ini, terdapat beberapa perubahan. Perubahan yang paling signifikan adalah tidak digunakannya hash function, digantikan dengan pesan yang akan dikirim. Algoritma ini memanfaatkan invers untuk melakukan recovery.

Key Generation

1. Pilih bilangan prima p dan q sehingga $(p-1) \, mod \, q = 0$

2. Menghitung $g = h^{\frac{p-1}{q}} \bmod p$, yang dalam hal ini $1 < h < p-1$ dan $h^{\frac{p-1}{q}} \bmod p > 1$
3. Tentukan kunci privat x, yang dalam hal ini $x < q$
4. Hitung kunci public $y = g^x \bmod p$
5. Kunci privat = x, kunci publik = (p, q, g, y)

Signing

1. Tentukan bilangan acak $k < q$.
2. Tanda tangan dari pesan m adalah bilangan r dan s. Hitung r dan s sebagai berikut:

$$r = m^{-1}(g^k \bmod p) \bmod q$$

$$s = \left(k^{-1}(1 + x \cdot r)\right) \bmod q$$

3. Kirim pesan m beserta tanda tangan r dan s

Recovery

1. Menghitung $u_1 = s^{-1} \bmod q$
2. Menghitung $u_2 = r \cdot u_1 \bmod q$
3. Menghitung $m = r^{-1}\left((g^{u_1} \cdot y^{u_2}) \bmod p\right) \bmod q$

Perhatikan bahwa banyak operasi invers yang dilakukan, agar komputasi lebih sedikit bentuk di atas ekuivalen dengan:

Signing

1. Tentukan bilangan acak $k < q$.
2. Tanda tangan dari pesan m adalah bilangan r dan s. Hitung r dan s sebagai berikut:

$$r = m(g^k \bmod p) \bmod q$$

$$s = (k - r \cdot x) \bmod q$$

3. Kirim pesan m beserta tanda tangan r dan s

Recovery

1. Menghitung $m = \left((g^s \cdot y^r) \bmod p\right)^{-1} r \bmod q$

B. ECDSA Dengan Pemulihan Pesan

Pada algoritma ini terdapat beberapa perbedaan dari algoritma yang asli. Perbedaan pertama adalah fungsi hash diganti dengan sebuah fungsi enkripsi. Selain itu, untuk melakukan tanda tangan diperlukan kunci publik dari penerima pesan.

Key Generation

1. Memilih sebuah bilangan bulat random d_A, yang nilainya diantara $[1, n-1]$
2. Menghitung $Q_A = d_A \cdot G = (x_A, y_A)$
3. Kunci rahasia = d_A, dan kunci publik = Q_A.

Signing

1. Memilih sebuah bilangan bulat random k, yang nilainya diantara $[1, n-1]$.
2. Menghitung $Q = k^{-1} \cdot d_A \cdot (G + y_B) \bmod n = (x_1, y_1)$
3. Menghitung $r = m \cdot H^{-1}(x_1) \bmod n$, jika $r = 0$, maka kembali ke langkah 1.
4. Menghitung $s = (d_A \cdot k^{-1} - d_A^2 \cdot H(r)) \bmod n$
5. Mengirim tanda tangan untuk message m adalah (r, s)

Recovery

1. Memverifikasi bahwa r dan s adalah bilangan bulat yang antara $[1, n-1]$
2. Menghitung $Q = (s \cdot (G + y_B) + H(r) \cdot (d_B^2 + 1) \cdot y_A) \bmod n = (x_1, y_1)$
3. Menghitung $m = r \cdot H(x_1) \bmod n$
4. Menerima tanda tangan jika dan hanya jika format m benar.

IV. Analisis Dan Pembahasan

Pada bagian ini akan dilakukan analisis perbandingan algoritma tanda tangan digital tanpa pemulihan dan dengan pemulihan. Kedua algoritma ini memiliki kelebihan dan kekurangan masing-masing.

A. Pengujian Algoritma Modifikasi

1. DSA Dengan Pemulihan Pesan

   Plaintext:

   Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus hendrerit lacus quis lorem finibus mattis. Nam viverra, lectus ac porttitor rutrum, justo nisi elementum lorem, sodales cursus orci ligula ut ligula. Proin ultricies vel augue non vulputate. Fusce placerat nunc in neque dictum pretium. Aliquam nec nisl sed urna finibus facilisis et ut libero. Praesent ut vestibulum mi. Nulla facilisi. Fusce dapibus volutpat elit nec consectetur.

   Signature:

   HL+94pqezmeEbByFKjkPefa9v8agmz5kd7bgqpbYs6oNLVl+tziFrIbDA31fFbWLdnF5JZ/iVso2QpZMvZ2qIw==

   Recovery:

   Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus hendrerit lacus quis lorem finibus mattis. Nam viverra, lectus ac porttitor rutrum, justo nisi elementum lorem, sodales cursus orci ligula ut ligula. Proin ultricies vel augue non vulputate. Fusce placerat nunc in neque dictum pretium. Aliquam nec nisl sed urna finibus facilisis et ut libero. Praesent ut vestibulum mi. Nulla facilisi. Fusce dapibus volutpat elit nec consectetur.

2. ECDSA Dengan Pemulihan Pesan

Plaintext:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus hendrerit lacus quis lorem finibus mattis. Nam viverra, lectus ac porttitor rutrum, justo nisi elementum lorem, sodales cursus orci ligula ut ligula. Proin ultricies vel augue non vulputate. Fusce placerat nunc in neque dictum pretium. Aliquam nec nisl sed urna finibus facilisis et ut libero. Praesent ut vestibulum mi. Nulla facilisi. Fusce dapibus volutpat elit nec consectetur.

Signature:

4+hV4myMX5nc+8klfiz38eNDcX6epAkStDK79YNKOsvjkYvjbpjCgajMAXicUYo5

Recovery:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus hendrerit lacus quis lorem finibus mattis. Nam viverra, lectus ac porttitor rutrum, justo nisi elementum lorem, sodales cursus orci ligula ut ligula. Proin ultricies vel augue non vulputate. Fusce placerat nunc in neque dictum pretium. Aliquam nec nisl sed urna finibus facilisis et ut libero. Praesent ut vestibulum mi. Nulla facilisi. Fusce dapibus volutpat elit nec consectetur.

B. Kelebihan dan Kekurangan Algoritma Modifikasi

Algoritma sebelumnya tidak mampu menyediakan pemulihan pesan. Pada skema seperti itu, tanda tangan ditambahkan pada pesan dan verifikasinya hanya mungkin jika pesan diketahui. Adapun kelebihan dari modifikasi tanda tangan digital dengan pemulihan pesan antara lain:

1. Pemulihan Pesan

   Kelebihan utama dari tanda tangan digital dengan pemulihan pesan adalah kemampuannya untuk memulihkan pesan asli yang hilang atau rusak. Dalam tanda tangan digital biasa, jika pesan terenkripsi hilang atau rusak, tidak mungkin untuk memulihkan pesan tersebut. Namun, dengan message recovery, informasi tambahan dalam tanda tangan digital memungkinkan pemulihan pesan yang akurat.

   Verifikasi pesan memerlukan langkah tambahan yang memberi tahu kita bahwa pesan yang dipulihkan adalah benar. Hal ini biasanya dicapai melalui penambahan redundansi pada pesan sebelum ditandatangani dan melalui pemeriksaan redundansi setelah pemulihan.

   Tentu saja, jika suatu skema tanda tangan menyediakan pemulihan pesan, selalu dapat digunakan dengan hashing juga. Pesan dapat dihash dan nilai hash ditandatangani. Pada pihak verifikasi, nilai hash dipulihkan (menggunakan fitur pemulihan pesan dari skema tanda tangan) dan otentisitas pesan diverifikasi melalui perbandingan nilai hash yang dipulihkan tersebut dengan nilai hash yang dihitung secara lokal dari pesan. Ini adalah proses yang biasanya kita terbiasa dengan pada aplikasi RSA.

2. Penghematan Ruang Penyimpanan

   Dengan tanda tangan digital biasa, pesan asli dan tanda tangan digital harus disimpan secara terpisah. Namun, dengan message recovery, pesan asli dapat dipulihkan dari tanda tangan digital, sehingga menghemat ruang penyimpanan yang diperlukan untuk menyimpan pesan asli.

Adapun kekurangan dari modifikasi algoritma ini antara lain:

1. Penambahan Kompleksitas

   Dalam tanda tangan digital biasa, tanda tangan itu sendiri sudah cukup untuk memverifikasi integritas dan otentikasi pesan. Namun, dalam message recovery, pemulihan pesan asli memerlukan informasi tambahan dalam tanda tangan digital. Oleh karena itu, jika informasi tambahan tersebut hilang atau rusak, tidak mungkin untuk memulihkan pesan asli.

   Implementasi tanda tangan digital dengan message recovery memerlukan tambahan langkah dan algoritma untuk memulihkan pesan asli. Hal ini dapat meningkatkan kompleksitas sistem dan mengharuskan penggunaan lebih banyak sumber daya komputasi.

2. Potensi Keamanan yang Menurun

   Dalam tanda tangan digital biasa, pesan asli dihash terlebih dahulu sebelum ditandatangani, sehingga menyediakan lapisan keamanan tambahan. Namun, dengan pemulihan pesan, pesan asli mungkin terungkap secara terbuka dalam tanda tangan digital. Hal ini berpotensi meningkatkan risiko keamanan jika pesan jatuh ke tangan yang salah.

   Semua algoritma modifikasi yang dibahas rentan terhadap serangan substitusi. Jika diberikan tanda tangan valid untuk suatu pesan, mudah untuk mengubah tanda tangan tersebut sedemikian rupa sehingga menjadi tanda tangan valid untuk pesan lain yang diketahui.

C. Keamanan Algoritma Modifikasi

Jika terdapat pihak ketiga yang ingin melakukan message forgery tanpa mengetahui kunci privat. Hal ini ekuivalen dengan mencari kombinasi semua variabel (g, y, m, r, s) sehingga memenuhi persamaan recovery. Hal ini akan sulit dilakukan jika pihak ketiga tidak memiliki pesan asli. Akan tetapi, jika pihak ketiga memiliki pesan asli dan tanda tangan digital, hal ini dapat dilakukan dengan memecahkan persamaan recovery. Kemudian melakukan serangan substitusi dengan pesan yang diinginkan oleh pihak ketiga. Pada umumnya, kelemahan ini ditangani dengan menambahkan redundansi pada pesan.

D. Performa Algoritma Modifikasi

Melalui pengujian, didapatkan waktu eksekusi yang dibutuhkan untuk melakukan tanda tangan dan verifikasi sebagai berikut:

Algoritma	Waktu Sign	Waktu Verify	Recovery
DSA	10,23 ms	11,12 ms	No
ECDSA	32,76 ms	46,21 ms	No
Modifikasi DSA	9,91 ms	20,77 ms	Yes
Modifikasi ECDSA	40,48 ms	70,03 ms	Yes

V. KESIMPULAN DAN SARAN

Tanda tangan digital dengan pemulihan pesan adalah sebuah metode yang memungkinkan pemulihan pesan asli dari tanda tangan digital. Skema ini memiliki kelebihan dalam hal pemulihan pesan yang hilang atau rusak, serta dapat mengurangi penggunaan ruang penyimpanan yang diperlukan untuk menyimpan pesan asli. Namun, skema ini juga memiliki kelemahan dan risiko yang perlu diperhatikan.

Potensi message forgery menjadi salah satu perhatian utama dalam skema digital signature dengan message recovery. Pihak yang tidak sah dapat mencoba memanipulasi pesan asli atau bagian redundansi pesan untuk menghasilkan tanda tangan digital yang masih valid. Oleh karena itu, penting untuk menggunakan skema yang dirancang dengan baik dan menerapkan langkah-langkah keamanan yang tepat untuk mengurangi risiko message forgery.

ACKNOWLEDGMENT

Kami ingin menyampaikan rasa terima kasih yang tulus kepada semua pihak yang telah memberikan kontribusi pada penelitian ini. Pertama-tama, puji syukur ke hadirat Tuhan yang Maha Esa yang telah memberikan kesempatan kepada penulis untuk menyelesaikan makalah ini dengan baik. Kami ingin berterima kasih kepada dosen pengampu mata pelajaran IF4020 Kriptografi, Dr. Ir. Rinaldi Munir, MT. yang telah memberikan pengetahuan dan dukungan terhadap topik yang diangkat pada makalah ini. Penulis berharap agar makalah ini dapat mendukung pengembangan block cipher yang lebih aman dan menambah wawasan pembaca mengenai tanda tangan digital.

REFERENCES

[1] Rinaldi Munir, “Algoritma ElGamal”, Bandung: ITB, 2023.

[2] Rinaldi Munir, “Elliptic Curve Cryptography”, Bandung: ITB, 2023.

[3] Rinaldi Munir, “Tanda Tangan Digital”, Bandung: ITB, 2023.

[4] Rinaldi Munir, “Digital Signature Standard”, Bandung: ITB, 2023.

[5] Svitlana Kazmirchuk, Anna Ilyenko, Sergii Ilyenko, Olena Prokopenko, and Yana Mazur, “The Improvement of Digital Signature Algorithm Based on Elliptic Curve Cryptography”, National Aviation University, 2021

[6] Svitlana Kazmirchuk, Sergii Ilyenko, “Digital Signature Authentication Scheme with Message Recovery Based on The Use of Elliptic Curve”, National Aviation University, 2020

[7] Nedal Tahat, Rania Shaqboua, Emad E. Abdallah, Mohammad Bsoul, Wasfi Shatanawi, "A new digital signature scheme with message recovery using hybrid problems," International Journal of Electrical and Computer Engineering (IJECE), 2019.

[8] Y. M. Tseng, J. K. Jan and H. Y. Chien, "Digital signature with message recovery using self-certified public keys and its variants," Applied Mathematics and Computation, vol. 136, no. 2-3, pp. 203-214, 2003.

[9] K. Nyberg and R. A. Rueppel, "Message recovery for signature schemes based on discrete logarithm problem," Advances in Cryptology – EUROCRYPT'94, Springer, Berlin, 1994, pp. 175–190.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 20 Mei 2023

Ubaidillah Ariq Prathama