SKEMA *BLIND SIGNATURE* BERBASIS *ELLIPTIC CURVE DISCRETE LOGARITHM PROBLEM*

Is Esti Firmanesa

Lembaga Sandi Negara, Jakarta
isesti.firmanesa@lemsaneg.go.id

ABSTRACT

Some blind signature schemes proposed previously are based on the Integer Faktorization Problem (IFP), such as RSA and Discrete Logarithm Problem (DLP), such as ElGamal, but both these schemes have not met the two properties as stated by Chaum, that every blind signature protocol should hold two fundamental properties, namely blindness and intractableness. For this reason, researchers made a blind signature scheme based on Elliptic Curve Cryptography (ECC) that its security depends on the Elliptic Curve Discrete Logarithm Problem (ECDLP). This scheme utilizes the inherent advantages in terms of the ECC key size is smaller than RSA and ElGamal with the same level of security. This scheme has proved to be robust and very difficult to trace. This scheme can be proposed for use in various applications such as e-voting, digital cash, and others.

Keywords: Blind Signature, ECC, ECDLP

PENDAHULUAN

Pada tahun 1985, ECC telah diperkenalkan pertama kali oleh Neal Koblitz dan Victor Miller. ECC adalah kriptosistem kunci publik yang aman dan efisien. Dalam [1], Vanstone telah menyimpulkan bahwa ECC menyediakan keefisiensian sekitar 10 kali lebih besar dari sistem faktorisasi integer maupun sistem logaritma diskrit, dalam hal biaya komputasi, ukuran kunci dan *bandwidth*. Selain itu, kelebihan dari ECC adalah pada sistem keamanannya dimana ECC mengandalkan kesulitan memecahkan ECDLP. Dari penelitian diperoleh informasi bahwa belum ada algoritma yg efisien untuk memecahkan ECDLP jika dibandingkan dengan IFP dan DLP yang sudah bisa dipecahkan dengan algoritma sub-eksponensial. Dalam hal ini, Vanstone menyatakan, "ECDLP diyakini lebih sulit dari IFP maupun DLP modulo *p*". Dengan begitu, keamanan yang tinggi dapat dicapai dengan ECC atas modulo bilangan prima *p*. Perbandingan panjang kunci dan waktu dalam memecahkan kunci ECC dan RSA/DSA dapat dilihat pada tabel 1.

Tabel 1. Perbandingan waktu untuk memecahkan kunci

Jumlah Bit		Waktu Untuk Memecahkan Kunci (MIPS Tahun)	Umur Proteksi
ECC	RSA/ DSA		
160	1024	1.00E + 12	Sampai 2010
224	2048	1.00E + 24	Sampai 2030

Tabel 1 memperlihatkan semakin besar panjang kunci, maka selisih lebar kunci yang digunakan semakin besar. Sebagai contoh untuk nilai keamanan 10^{12} MIPS (*Million Instructions Per Second*) tahun, ECC hanya membutuhkan 160 bit sedangkan RSA/DSA membutuhkan 1024 bit. Dengan kata lain RSA/DSA membutuhkan 6,4 kali jumlah bit kunci ECC untuk nilai 10^{12} MIPS tahun. MIPS tahun adalah satuan waktu yang menunjukkan kemampuan suatu algoritma untuk melakukan berjuta-juta instruksi per detik dalam satu tahun. Hal ini tentunya ECC dapat menjadi pilihan yang baik untuk membangun sistem kriptografi yang memiliki tingkat keamanan yang tinggi (Nana Juhana, 2005). Dari pertimbangan-pertimbangan tentang ECC di atas, paper

ini akan menyajikan skema *blind signature* berdasarkan ECDLP sebagai kekuatan keamanan ECC untuk mencapai syarat-syarat *digital signature* dan *blind signature* yang efisien.

Skema tanda tangan digital memungkinkan seseorang untuk menandatangani dokumen sedemikian rupa sehingga setiap orang dapat memverifikasi keabsahan tanda tangan (*signature*) otentik, tapi tidak ada seorang pun yang bisa memalsukan tanda tangan tersebut.

Skema tanda tangan buta (*blind signature*) adalah salah satu dari skema tanda tangan digital yang diusulkan oleh Chaum. Dalam mekanisme ini, dokumen yang akan ditandatangani akan "buta (*blind*)" bagi seorang penandatangan (*signer*) yang tidak mengetahui isi dokumen tersebut sebelum penandatanganan. Pertama kali skema *blind signature* diusulkan oleh Chaum pada tahun 1982. Chaum mengonstruksi *blind signature* sebagai alat kunci untuk mengembangkan sistem *e-cash*. Dia menunjukkan bahwa keragaman dan pertumbuhan layanan elektronik dapat berdampak pada privasi konsumen dan tingkat penggunaan pidana sehingga kriptosistem *blind signature* sangat disarankan diaplikasikan untuk keperluan keamanan. Kriptosistem *blind signature* benar-benar dirancang untuk melindungi privasi pelanggan dalam penggunaan sistem pembayaran elektronik yang aman dengan menjunjung tinggi privasi pelanggan. Pada dasarnya, skema *blind signature* adalah protokol untuk sekelompok pemohon (requester) dan penandatangan (*signer*).

Penggunaan skema *blind signature* selain ditemukan pada kegiatan atau transaksi, misalnya skema *cash digital* (*e-cash*), penerapan skema ini dapat digunakan dalam sistem pemilu kriptografi (*e-voting*). Pada *e-voting*, salah satu sistemnya adalah syarat-syarat dari pemungutan suara telah disetujui oleh pihak yang berwenang, misalnya pusat tabulasi atau panitia pemilihan sebelum pemungutan suara dihitung. Hal ini memungkinkan pihak berwenang untuk memeriksa pemilih dan memastikan pemilih bisa memilih, dan mereka tidak memberikan lebih dari satu suara. Namun, panitia pusat tabulasi/pemilu tidak akan mengetahui pilihan pemilih. Sehingga sistem *e-voting* diperlukan mekanisme *blind signature*.

Mekanisme global *blind signature* dapat digambarkan berikut ini. Setiap *requester* mengirimkan pesan terenkripsi untuk *signer* dan memperoleh tanda tangan (*signature*) yang valid dari *signer*. Perhatikan bahwa *signer* hanya menandatangani pesan dan tidak mendekripsi pesan tersebut. Kemudian, *signer* dapat memverifikasi keaslian *signature* setiap kali dia menerima pasangan pesan-*signature*. Namun, *signer* tidak dapat menghubungkan pasangan pesan-*signature* ke fase tertentu dari protokol penandatanganan yang menghasilkan pasangan ini.

METODA PENELITIAN

Metode penelitian yang digunakan dalam paper ini adalah Metode Penelitian dan Pengembangan, yaitu membahas ECC sebagai kriptografi kunci publik berbasis grup kurva eliptik yang tingkat keamanannya tergantung pada ECDLP diterapkan pada skema *blind signature*. Tujuan pembuatan skema *blind signature* berbasis ECDLP ini adalah untuk mencapai keefisiensian yang lebih baik dengan tingkat keamanan yang lebih tinggi daripada skema *blind signature* yang diterapkan pada RSA maupun algoritma ElGamal.

HASIL DAN PEMBAHASAN

Skema yang dibuat dalam paper ini adalah skema *blind signature* analog dengan ECC yang berbasis keamanan ECDLP. Sebelum membahas skema *blind signature* tersebut akan dibahas landasan teori yang mendukung skema tersebut.

DIGITAL SIGNATURE ..(1)

Tanda tangan digital (*digital signature*) digunakan proses pengiriman dan penerimaan pesan melalui jalur pribadi berdasarkan kesepakatan antara semua *user* terkait. Pesan ini dienkripsi dan didekripsi menggunakan sistem kriptografi tertentu untuk menjamin kerahasiaan dan integritas pesan. Peran *digital signature*

dalam proses pengiriman/penerimaan pesan ini adalah untuk menjamin integritas, otentikasi (pesan dan *user*), dan non-repudiasi. Konsep *digital signature* awalnya berasal dari kriptografi yang didefinisikan sebagai metode untuk pesan pengirim yang dienkripsi atau didekripsi dengan melibatkan fungsi *hash* untuk menjamin kerahasiaan pesan ketika ditransmisikan. Ketika fungsi *hash* digunakan untuk pesan, maka *digital signature* yang dihasilkan disebut pesan *digest*. Fungsi *hash* adalah algoritma matematika yang membuat pesan dari input dengan panjang sembarang sebagai masukan dan menghasilkan output dengan panjang tetap. Karena syarat yang satu arah, maka bagi pihak ketiga tidak mungkin untuk mendekripsi pesan yang terenkripsi.

Dua tahapan proses tanda tangan digital dijelaskan di bawah ini.

1. Fase *Signing*
   Proses dari penandaan (*signing*) memerlukan pentransformasian beberapa pesan rahasia yang dibawa oleh *user* ke dalam suatu tanda pengenal yang disebut tanda tangan (*signature*). Pertma kali pengirim membuat pesan/data sebagai input dari fungsi *hash* kemudian menghasilkan pesan *digest* sebagai *output*-nya. Yang ke-2, pesan *digest* ini dienkripsi dengan kunci rahasia pengirim dan *digital signature* pesan ini sudah dilakukan. Akhirnya, pengirim mengirim pesan/datanya dengan *digital signature*-nya ke penerima.
2. Fase Verifikasi
   Pertama kali penerima memperoleh pesan dengan *digital signature*-nya, dia mengulang proses yang sama dengan yang dilakukan oleh pengirim, yaitu menggunakan pesan tersebut sebagai input ke fungsi *hash* untuk memperoleh pesan *digest* pertama sebagai *output*. Kemudian mendekripsi *digital signature* dengan menggunakan kunci publik pengirim untuk memperoleh pesan *digest* ke-2. Terakhir, memverifikasi apakah dua pesan *digest* ini sama atau tidak.

Pada fase verifikasi, apabila diketahui bahwa kedua pesan *digest* tersebut tidak sama berarti ada gangguan terhadap keontetikasi dan integritas pesan/data dari pihak lain yang tidak berwenang.

***BLIND SIGNATURE* ..(2)**

Signer menandatangani pesan *requester* dan tidak mengetahui apapun tentang ini, selain itu tidak seorang pun mengetahui tentang hubungan antara pasangan pesan-*signature* kecuali *requester*. Proses *blind signature* adalah sebagai berikut:

1. Fase *Blinding*
   Awalnya pengirim memilih bilangan secara acak yang disebut faktor *blind* untuk menyamarkan pesannya sehingga *signer* akan membutakan pesan tersebut.
2. Fase *Signing*
   Ketika *signer* memperoleh pesan *blinded*, dia langsung mengenkripsi pesan *blinded* dengan kunci rahasianya dan kemudian mengirimkan *blind signature* kembali ke pengirim.
3. Fase *Unblinding*
   Pengirim menggunakan faktor *blind*-nya yang ditentukan dalam (1) untuk menemukan kembali *digital signature*-nya *signer* dari *signature* yang dibutakan (*blinded signature*).
4. Fase Verifikasi *Signature*
   Siapapun dapat menggunakan kunci publik *signer* untuk memverifikasi apakah *signature* tersebut adalah asli.

Protokol *blind signature* berbasis pada kriptosistem RSA pertama kali diusulkan oleh Chaum. Kemudian penelitian selanjutnya adalah membuat protokol *blind signature* berbasis kriptosistem ElGamal.
Pada tahun 1994, Camenisch memberikan definisi lebih jauh tentang "*blindness*" untuk sebuah skema *signature*. Skema *signature* dikatakan *blind* sesuai definisi Camenisch adalah bahwa ada faktor *blinding* yang unik untuk sepasang pesan-*signature* (*m*, *s*(*m*)) sehingga dua pesan yang sama ditandai akan menghasilkan dua *signature* yang berbeda.

***ELLIPTIC CURVE CRYPTOGRAPHY* ..(3)**

Masalah Komputasi Berbasis Keamanan

1. Misalkan diberikan n, p, $q \in \mathbb{Z}$. IFP adalah menghitung p dan q sehingga $n = pq$.
2. Misalkan diberikan suatu prima p, sebuah generator α dari $\mathbb{Z}_p^*$ dan elemen $\beta \in \mathbb{Z}_p^*$. DLP adalah menghitung x, $0 \le x \le p\text{-}2$ sehingga $\alpha^x \equiv \beta \pmod p$.
3. Misalkan $E(\mathbb{Z}_p)$ adalah kurva eliptik dan P, Q adalah titik-titik pada $E(\mathbb{Z}_p)$, diberikan $P \in E(\mathbb{Z}_p)$ berorder n (jumlah titik hasil perkalian titik P), $Q \in \langle P \rangle$, dan $Q=aP$, $a \in [0, n\text{-}1]$.
   ECDLP adalah menghitung a sehingga $Q=aP$.

Kurva Eliptik atas $\mathbb{Z}_p$**.** Misalkan $p > 3$ adalah bilangan prima dan $4a^3 + 27b^2 \neq 0$, dimana $a, b \in$ R. Kurva eliptik $E_p(a, b) = \{(x, y) \in R^2 / y^2 = x^3 + ax + b, 4a^3 + 27b^2 \neq 0, x, y, a, b \in R\} \cup \{\mathcal{O}$ (*titik infinity*)$\}$. $(E(\mathbb{Z}_p), +)$ membentuk grup abelian dengan $\mathcal{O}$ sebagai identitasnya.
Kurva eliptik E adalah himpunan titik (x, y) dengan x, $y \in E$ yang memenuhi $y^2=x^3+ax+b$ disertai dengan sebuah elemen tunggal, yaitu $\mathcal{O}$.

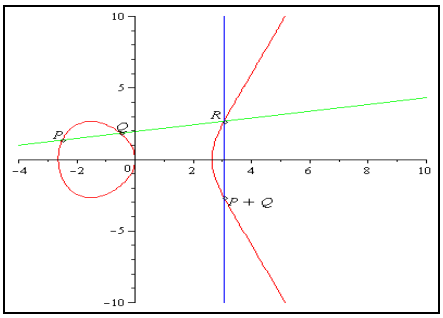
Order kurva eliptik $E(\mathbb{Z}_p)$ adalah jumlah maksimun titik yang terdapat pada suatu kurva eliptik $E(\mathbb{Z}_p)$ yang dinotasikan $\#E(\mathbb{Z}_p)$ berada pada interval $p + 1 - 2\sqrt{p} \le \#E(\mathbb{Z}_p) \le p + 1 + 2\sqrt{p}$.
Misalkan: a=1, b=6 dan p=11, persamaan kurva eliptik $y^2 = x^3 + x + 6$ (mod11), sehingga $4a^3+27b^2 = 4.1^3+27.6^2 = 976 \pmod{11} = 8 \neq \pmod{11}$. Selanjutnya dicari elemen-elemen grup eliptik $E_{11}(1,6)$ atas Z_{11}, dengan $Z_{11}=\{0,1,2,3,4,5,6,7,8,9,10\}$. Sebelum menentukan elemen-elemen $E_{11}(1,6)$, terlebih dahulu mencari residu kuadrat modulo 11 untuk mendapatkan nilai pasangan (x,y). Setelah melalui semua perhitungan tersebut diperoleh semua titik pada kurva eliptik $E_{11}(1, 6)$ adalah {(2, 4), (2, 7), (3, 5), (3, 6), (5, 2), (5, 9), (7, 2), (2, 9), (8, 3), (8, 8), (10, 2), (10, 9), $\mathcal{O}$ }.

Untuk suatu titik P pada kurva eliptik, $(\mathcal{O}+P+PP+\ldots)$ adalah *grup siklik*. Notasinya adalah $\langle P \rangle = \{0P=\mathcal{O}, 1P=P, 2P=P+P, 3P=P+P+P, \ldots\}$. Operasi penjumlahan ini disebut juga perkalian skalar, yaitu kP menunjukkan penjumlahan P sebanyak k kali, yaitu $kP=P+P+\ldots+P$.

Order dari suatu titik $P \in E(\mathbb{Z}_p)$ adalah bilangan bulat positif terkecil k, sehingga $kP = \mathcal{O}$ dan titik P disebut titik hingga (*finite point*). Jika tidak ada integer positif yang memenuhinya, maka order titik tersebut dikatakan tidak berhingga (*infinity*).

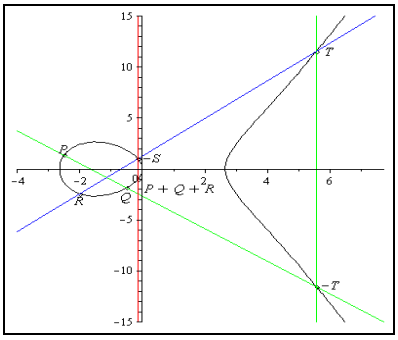
Struktur Grup Kurva Eliptik:
Diberikan sebarang himpunan tidak kosong $E(\mathbb{Z}_p)$ dan operasi penjumlahan titik kurva eliptik, maka $(E(\mathbb{Z}_p), +)$ disebut grup jika dipenuhi hal-hal sebagai berikut:
Misalkan P, Q, dan R adalah titik-titik pada kurva eliptik, maka:

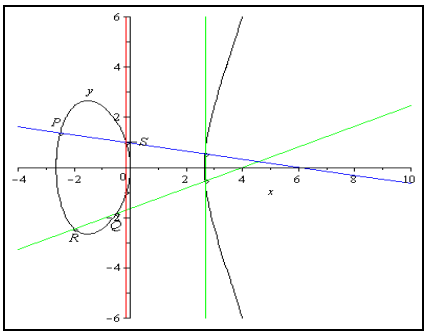
a. memenuhi sifat tertutup terhadap penjumlahan dalam $E(\mathbb{Z}_p)$: $P+Q=R \in E(\mathbb{Z}_p)$;



Gambar 1. Penjumlahan dua titik P dan Q yang berbeda

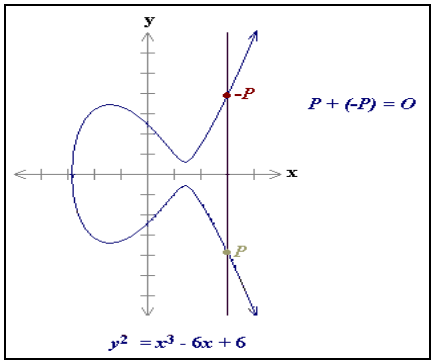
b. memenuhi sifat asosiatif: $(P+Q)+R=P+(Q+R)$;





Gambar 2. Penjumahan tiga titik memenuhi sifat asosiatif $(P+Q)+R=P+(Q+R)$

c. mempunyai identitas, yaitu titik *infinity* $\mathcal{O}$;



Gambar 3. Operasi penjumlahan Identitas

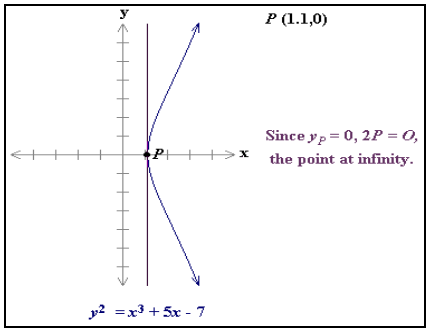
d. mempunyai invers, misal $-P$ dari P (lihat gambar 3).

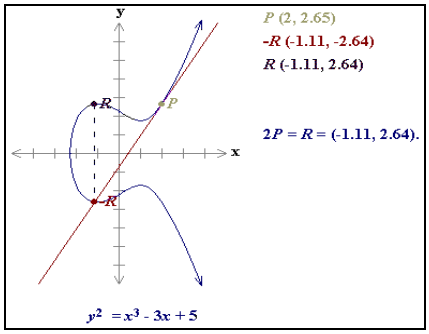
Operasi aritmetika penjumlahan pada $E(\mathbb{Z}_p)$ adalah sebagai berikut:

1. $P+\mathcal{O}=\mathcal{O}+P=P$, untuk semua $P\in E(\mathbb{Z}_p)$ (lihat gambar 3);
2. Jika $P(x,y)\in E(\mathbb{Z}_p)$, maka $(x,y)+(x,-y)=\mathcal{O}$ ($(x,-y)\in E(\mathbb{Z}_p)$ dinotasikan $-P$ dan disebut negatif P) (lihat gambar 3);
3. Penjumlahan dua titik yang berbeda. Misal $P=(x_1,y_1)\in E(\mathbb{Z}_p)$, $Q=(x_2,y_2)\in E(\mathbb{Z}_p)$, $P \neq \pm Q$, maka $P + Q = (x_3, y_3)$, dimana $x_3 = \lambda^2 - x_1 - x_2$; $y_3 = \lambda (x_1 - x_3) - y_1$; dan $\lambda = (y_2-y_1)/(x_2-x_1)$ (lihat gambar 1);

   Contoh:
   Misalkan titik $P(3,10)$ dan $Q(9,7)$ dalam $E_{23}(1,1)$, maka $P+Q=R(x_R,y_R)$, dengan x_R dan y_R diperoleh dengan menghitung nilai λ terlebih dahulu.
   $\lambda = (y_2-y_1)/(x_2-x_1) = (7-10)/(9-3) = -3/6$
   $= -1/2 = -2^{-1} \pmod{23} = 11$,
   kemudian dapat dihitung nilai x_3 dan y_3 , yaitu:
   $x_3 = \lambda^2 - x_1 - x_2 = 11^2\text{-}3\text{-}9 \pmod{23} = 17$
   $y_3 = \lambda (x_1 - x_3) - y_1 = 11(3\text{-}17)\text{-}10 \pmod{23} = 20$
   Jadi $P+Q=(17,20)$.
4. Penjumlahan dua titik yang sama.
   Misal $P=(x_1,y_1)\in E(\mathbb{Z}_p)$, $P+P=2P=(x_3,y_3)$, dimana $x_3 = \lambda^2 - 2x_1$; $y_3 = \lambda (x_1-x_3)-y_1$; dan $\lambda = (3x_1^2 + a) / 2y_1$.
   Operasi ini disebut *doubling* suatu titik.

   Contoh:
   Misalkan titk $P(x_1,y_1)=P(3,10)$ dalam $E_{23}(1,1)$, sehingga perkalian skalar $2P=P+P$ dihitung dengan cara berikut ini.
   $\lambda = (3x_1^2 + a) / 2y_1$
   $= (3.3^2+1) / 2.10 \pmod{23} = 6$
   $x_3 = \lambda^2 - 2x_1 = 36\text{-}6 \pmod{23} = 7$
   $y_3 = \lambda (x_1-x_3)-y_1 = 6(3\text{-}7)\text{-}10 = 12$
   Jadi, $2P = (x_3, y_3) = (7, 12)$.





Gambar 4. Operasi *doubling* jika $yP = 0$ dan $yP \neq 0$

SKEMA *BLIND SIGNATURE BERBASIS* ECC ..(4)

Pada proses enkripsi dan dekripsi kurva eliptik, *plaintext* m diwakili menjadi titik $P(x,y)$ pada bidang koordinat x, y. Titik yang akan dienkripsi sebagai sebuah *ciphertext* dan kemudian didekripsi.

Stallings [1] menyatakan, "Ada teknik relatif mudah yang dapat digunakan untuk mempresentasikan pesan sebagai titik pada kurva eliptik".

Metode kurva eliptik yang diusulkan oleh Koblitz dijelaskan berikut ini. Misalkan $Ep(a,b)$ adalah kurva eliptik atas $\mathbb{Z}_p$ dengan persamaan $y^2=x^3+ax+b \pmod p$, dimana $4a^3 + 27b^2 \neq 0 \pmod p$. Dan kemudian titik dasar $G = (x, y)$ pada $E_p(a, b)$ ditentukan yang mempunyai order yang sangat besar n sehingga $n \times G = \mathcal{O}$. Dua kelompok yaitu *requester* $\{R_i \mid 1 \leq i \leq n, n \in N\}$, dan seorang *signer,* semuanya adalah peserta dari skema *blind signature*. *Requester* R_i memilih kunci privat $n_i \in \mathbb{Z}_p$ dan membangkitkan kunci publik yang berkaitan $P_i \equiv n_i \times G \pmod p$. Demikian juga *signer* memilih kunci privat secara acak $n_s \in \mathbb{Z}_p$ dan membangkitkan kunci publik yang berkaitan $P_s \equiv n_s \times G \pmod p$.

Misalkan pesan m yang akan ditandatangani (*signed*), pertama R_i membangkitkan faktor *blinding* $(n_i \times P_i)$ kemudian pesan *blinded* α dikirimkan ke *signer*, diketahui $\alpha \equiv m \times (n_i \times P_i) \pmod p$. Faktor ini disebut faktor *blinding* karena *signer* tidak mengetahui apapun tentang pesan m sampai perhitungan $\alpha \equiv m \times (n_i \times P_i) \pmod p$.Untuk setiap α yang akan ditandatangani, *signer* memilih $n_v \in \mathbb{Z}_p$ secara acak sebagai faktor *blinding* ke-2 dan membangkitkan pasangan *blind signature* (r,s), yaitu:
$r \equiv n_v \times \alpha \pmod p$ dan $s \equiv (n_v + n_s) \times \alpha \pmod p$.

Catat bahwa n_s adalah kunci privat *signer*. Kemudian pasangan pesan dan *signature* $(\alpha,(r,s))$ dikirimkan ke R_i. Disarankan bahwa *signer* harus menyimpan catatan (α,n_v) *database* untuk menghindari *collision*. Meskipun kemungkinannya kecil untuk dua pesan yang sama menggunakan faktor *blinding* n_v yang sama. Cara untuk memecahkan situasi tersebut adalah menjaga catatan tersebut untuk menentukan faktor-faktor *blinding* yang berbeda untuk pesan yang sama.

Ketika *requester* R_i menerima pasangan pesan-*signature* $(\alpha,(r,s))$ maka R_i mengupas *signature* (r,s) dengan menggunakan kunci privatnya n_i bersama kunci publik P_s milik *signer* untuk menghasilkan *signature* yang dikupas s', dimana:
$s' \equiv s - m \times n_i \times P_s \pmod p$. Kemudian R_i menghitung: $m' \equiv n_i (n_i - 1) \cdot m$.

Terakhir, *requester* R_i menerbitkan (m', s', r) dan siapapun dapat menggunakan kunci publik *signer* P_s untuk memverifikasi keotentikan *signature* yang dikupas s' dengan memeriksa apakah memenuhi formula $r \equiv s' - m' \times P_s \pmod p$.

Ilustrasi protokol *blind signature* adalah sebagai berikut:

1. *Requester* R_i membawa pesan m kemudian membuat $\alpha \equiv m \times (n_i \times P_i) \pmod p$ dan dikirimkan ke *signer.*
2. *Signer* menandatangani α dengan memilih secara acak n_v dan memeriksa apakah (α,n_v) ada di dalam *database*-nya.
   Jika ya, *signer* memilih n_v yang berbeda untuk pesan *blinded* yang sama berikutnya. Kemudian *signer* menghitung $r \equiv n_v \times \alpha \pmod p$ dan $s \equiv (n_v + n_s) \times \alpha \pmod p$ kemudian mengembalikan pasangan pesan-*signature* $(\alpha,(r,s))$ ke *requester* R_i.
3. Dalam kasus ini, *signer* harus bisa menjaga (α,n_v) di dalam *database*-nya.
4. *Requester* R_i menggunakan s pada (r,s) dengan menggunakan kunci rahasianya sendiri, yaitu n_i dan kunci publik *signer* P_s untuk menghasilkan $s' \equiv s - m \times n_i \times P_s \pmod p$. Kemudian R_i menghitung $m' \equiv n_i (n_i - 1) \cdot m$.
5. Siapapun dapat mengakses kunci publik *signer* P_s untuk memverifikasi keontetikan *signature* (m', s', r) dengan memeriksa apakah memenuhi $r \equiv s' - m' \times P_s \pmod p$.

Teorema 1

Pasangan tiga (m', s', r) adalah *signature* dari pesan m yang merupakan protokol dari skema *blind signature.*

Bukti:

Pembuktian bahwa (m', s', r) adalah *signature* yang valid dari pesan m untuk protokol di atas. Validitas dari *signature*

(m', s', r) dapat ditunjukkan sebagai berikut:
(Buktikan $r \equiv s' - m' \times P_s$ adalah sama dengan $r \equiv n_v \times \alpha$)
Karena $s' \equiv s - m \times n_i \times P_s \pmod p$ dan $m' \equiv n_i (n_i - 1) \cdot m$ maka

$$
\begin{aligned}
r &\equiv s' - m' \times P_s \\
&\equiv s - m \times n_i \times P_s - n_i (n_i - 1) \cdot m \times P_s \\
&\equiv s - m \times n_i \times P_s - n_i \times n_i \times m \times P_s + (n_i \times m \times P_s) \\
&\equiv s - n_i \times n_i \times m \times P_s \equiv (n_v + n_s) \times \alpha - n_i \times n_i \times m \times P_s \\
&\equiv (n_v + n_s) \times m \times n_i \times P_i - n_i \times n_i \times m \times P_s \\
&\equiv n_v \times m \times n_i \times P_i + n_s \times m \times n_i \times P_i - n_i \times n_i \times m \times P_s \\
&\equiv n_v \times m \times n_i \times P_i + n_s \times m \times n_i \times n_i \times G - n_i \times n_i \times m \times n_s \times G \\
&\equiv n_v \times \alpha \Rightarrow r
\end{aligned}
$$

Terbukti bahwa $r \equiv s' - m' \times P_s$ pada (m', s', r) adalah sama dengan $r \equiv n_v \times \alpha$.

Menurut syarat *blindness* yang didefinisikan oleh Camenisch, ada dua pasangan pesan-*signature* (m, (r_1,s_1)) dan (m, (r_2,s_2)) yang dihasilkan dengan memberikan dua pesan yang sama karena pesan yang sama tersebut masing-masing mempunyai faktor *blinding* n_v. Untuk membuktikan protokol *blindness*, paper ini akan menunjukkan bahwa dua pesan-*signature* yang sama akan diaplikasikan dengan faktor-faktor *blinding* yang sama juga. Misalkan n_1 dan n_2 adalah faktor *blinding* berturut-turut untuk dua pesan yang sama, (r_1,s_1) dan (r_2,s_2) adalah *signature* yang sesuai dihasilkan dari protokol di atas.

Misalkan (r_1,s_1) dan (r_2,s_2) sama, maka:

$$
\begin{aligned}
r_1 &\equiv r_2 \pmod p \\
n_1 x \alpha &\equiv n_2 x \alpha \\
n_1 &\equiv n_2
\end{aligned}
$$

dan

$$
\begin{aligned}
s_1 &\equiv s_2 \pmod p \\
(n_1 + n_s) x \alpha &\equiv (n_2 + n_s) x \alpha \\
n_1 &\equiv n_2
\end{aligned}
$$

Oleh karena itu, n_1 dan n_2 adalah dua faktor *blinding* yang berbeda akan menghasilkan dua *signature* yang berbeda juga dan memenuhi protokol *blindness.*

Seperti *blindness* yang didefinisikan oleh Chaum, *signer* tidak mengetahui hubungan antara sesuatu yang ditandatangani *s* dan sesuatu yang ditandatangani yang dikupas *s'*. Hal ini jelas bahwa *signer* dalam protokol di atas tidak dapat melacak *s*' dari *s* karena *s*' yang dihasilkan dengan menerapkan kunci rahasia n_i dari *requester* R_i. Seperti yang sudah diketahui bahwa untuk mendapatkan kunci rahasia dari *requester* setara dengan memecahkan masalah logaritma diskret kurva eliptik. Dengan demikian, protokol *blind signature* berbasis ECC ini juga memegang properti *blindness* yang didefinisikan oleh Chaum sehingga memenuhi properti *blindness*.

Contoh:

1. Menentukan persamaan kurva eliptik dan titik dasar/generator kurva eliptik. Misalkan persamaan kurva eliptik $y^2 = x^3 + x + 6 \pmod{11}$ dan titik dasar G=(2,7).
2. *Signer* memilih n_s=3 secara acak sebagai kunci rahasianya dan kemudian membangkitkan kunci publiknya P_s, dimana $P_s \equiv 3 \times (2,7) \pmod{11}$.
3. *Requester* memilih n_i=7 secara acak sebagai kunci rahasianya dan kemudian membangkitkan kunci publiknya P_i dimana $P_i \equiv 7 \times (2,7) \pmod{11}$.
4. *Requester* menetapkan faktor *blinding*-nya ($7 \times P_i$) untuk mengubah pesan *m*-nya menjadi pesan *blinded* α, dimana $\alpha \equiv m \times (7 \times P_i) \pmod{11}$. Kemudian pesan *blinded* α tersebut dikirimkan ke *signer.*
5. *Signer* memilih n_v=5 secara acak sebagai faktor *blinding* ke-2. *Signer* harus memeriksa apakah (α, 5) ada di dalam *database*. Jika ya, *signer* memilih bilangan yang beda untuk pesan *blinded* yang sama berikutnya. Kemudian *signer* membangkitkan pasangan dari *blind signature* (*r*,*s*) dimana $r \equiv 5 \times \alpha \pmod{11}$ dan $s \equiv (5+3) \times \alpha \pmod{11}$. Yang terakhir, *signer* mengirimkan pasangan pesan-*signature* (α,(*r*,*s*)) kembali ke *requester*.
6. *Requester* mengupas *s* dari (*r*,*s*) dengan menggunakan kunci

rahasianya n_i=7 dan kunci publik *signer* P_s yang menghasilkan $s' \equiv s - m \times 7 \times P_s \pmod{11}$, kemudian *requester* juga menghitung $m' \equiv 7\ (7-1) \cdot m$. Terakhir *requester* mengumumkan pasangan tiga (m', s', r).

7. Proses *blind signature* pada contoh di atas kemudian diverifikasi untuk memeriksa apakah (m', s', r) memenuhi formula $r \equiv s' - m' \times P_s \pmod{11}$ dan akan dibuktikan pada Fase Verifikasi berikut ini.

Fase Verifikasi

Validitas *signature* (m', s', r) dengan mudah dapat ditunjukkan sebagai berikut, karena $s' \equiv s - m \times 7 \times P_s \pmod{11}$ dan $m' \equiv 7\ (7-1) \cdot m$, maka:

$$
\begin{aligned}
& s' - m' \times P_s \\
= \ & s - m \times 7 \times P_s - 7\ (7-1) \cdot m \times P_s \\
= \ & s - m \times 7 \times P_s - 7 \times 7 \times m \times P_s + 7 \times m \times P_s \\
= \ & s - 7 \times 7 \times m \times P_s \\
= \ & (5+3) \times \alpha - 7 \times 7 \times m \times P_s \\
= \ & (5+3) \times m \times (7 \times P_i) - 7 \times 7 \times m \times P_s \\
= \ & 5 \times m \times 7 \times P_i + 3 \times m \times 7 \times P_i - 7 \times 7 \times m \times P_s \\
= \ & 5 \times m \times 7 \times 7 \times (2,7) + 3 \times m \times 7 \times 7 \times (2,7) - 7 \times 7 \times m \times 3 \times (2,7) \\
= \ & 5 \times m \times 7 \times P_i = 5 \times \alpha = n_v \times \alpha \\
\equiv \ & r \pmod{11}
\end{aligned}
$$

Terbukti bahwa (m', s', r) memenuhi formula $r \equiv s' - m' \times P_s \pmod{11}$.

KESIMPULAN

Penelitian ini menghasilkan kesimpulan sebagai berikut :

1. Ada tiga basis keamanan pada kriptografi asimetrik, yaitu kriptografi berbasis IFP (misalkan, RSA), DLP (misalkan DSA/ElGamal), dan ECDLP (yaitu ECC).
2. Panjang kunci dengan level keamanan yang sama, ECC lebih baik daripada RSA maupun algoritma ElGamal, contohnya RSA/ElGamal dengan panjang kunci 4096 bit memberikan level keamanan yang sama dengan panjang kunci 313 pada ECC. Dengan demikian ECC mempunyai keefisiensian yang lebih besar 10 kali lebih bila dibandingkan dengan RSA/ElGamal.
3. *Blind Signature* adalah kriptografi yang sangat penting saat ini untuk digunakan dalam protokol-protokol seperti *e-voting* dan *e-cash* yang aman untuk melindungi privasi pelanggan dengan menjunjung tinggi privasi pelanggan. Skema *blind signature* yang sudah diusulkan adalah berbasis pada IFP dan DLP. Namun skema tersebut tidak memenuhi syarat *digital signature* dan *blindness.*
4. Dengan pertimbangan keefisiensian dan keamanan, ECC lebih baik daripada kriptografi berbasis IFP dan DLP, serta belum terpenuhinya syarat *digital signature* dan *blindness* pada skema *blind signature* sebelumnya, maka diusulkan skema *blind signature* berbasis ECDLP pada ECC atas $\mathbb{Z}_p$.
5. Skema *blind signature* berbasis ECDLP memberikan keefisiensian lebih besar karena membutuhkan penyimpanan lebih hemat dan komputasional lebih sedikit jika dibandingkan dengan skema *blind signature* lainnya. Hal ini dikarenakan dengan level keamanan yang sama, panjang kunci ECC lebih pendek daripada kedua kriptografi berbasis IFP maupun DLP tersebut. Di samping itu ECC telah memenuhi syarat skema *blind signature.*

DAFTAR PUSTAKA

[1] Darrel Hankerson, Alfred Menezes, S. Vanstone. (2003). *Guide to Elliptic Curve Cryptography*. Springer-Verlag.

[2] Esti Rahmawati Agustina, Is Esti Firmanesa. (2013). *Implementation of RSA Blind Signature on Crypto-ON2 Protokol*. World Academy of Science, Engineering and Technology. London

[3] Fuh-Gwo Jeng, Tzer-Long Chen, Tzer-Shyong Chen. (2010). *An ECC-Based Blind Signature Scheme.* Journal Of Network. Academy Publisher.

[4] Is Esti Firmanesa. (2009). *Konstruksi Algoritma Penandaan Dijitel ElGamal Berbasis Grup Kurva Eliptik*. Tesis Program Magister Matematika Terapan, IPB.

[5] Nana Juhana. (2005). *Implementasi Elliptic Curve Cryptosistem (ECC) pada Proses Pertukaran Kunci Diffie-Hellman dan Skema Enkripsi ElGamal*. Tugas Akhir Kemanan Sistem informasi Lanjut, ITB.
[6] S. A. Vanstone. (1997). *Elliptic Curve Cryptosistem-The Answer to Strong, Fast Public-key Cryptography for Securing Constrained Environments*. Information Security Technical Report.
[7] W. Stallings. (2003). *The Book of Cryptography and Network Security-Principles and Practices*, 3rd ed. Prentice Hall.