REPOSITOR, Vol. 3, No. 3, Mei 2021, Pp. 331-342
ISSN : 2714-7975
E-ISSN : 2716-1382

Perbandingan Kinerja Algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) Menggunakan Fungsi *Hash Secure Hash Algorithm* (SHA-1) dan Keccak pada Tanda Tangan Digital

Nur Annisa Fitriani*[1], Aminudin[2], Sofyan Arifianto[3]
[1,2,3]Universitas Muhammadiyah Malang
nurannisafitriani@webmail.umm.ac.id[*1], aminudin2008@umm.ac.id[2],
sofyan.arifianto@gmail.com[3]

Abstrak

Sebuah data atau dokumen yang dikirimkan melalui internet sangat rentan terhadap serangan atau modifikasi serta sangat sulit untuk membuktikan keaslian data atau dokumen, maka dengan perkembangan sistem keamanan terbentuklah sebuah mekanisme kriptografi yang digunakan untuk memverifikasi keaslian dan kebenaran dari sebuah data yang disebut dengan tanda tangan digital. Tanda tangan digital seringkali dipadukan dengan fungsi hash untuk membuat tanda tangan pada suatu data. Algoritma tanda tangan yang sering digunakan adalah Diffle-Helman Digital Signature Algorithm dan lebih dikenal sebagai Digital Signature Algorithm (DSA). Algoritma Digital Signature Algorithm (DSA) dikembangkan menjadi algoritma Elliptic Curve Digital Signature Algorithm (ECDSA) yang menggunakan elliptic curve. Penelitian ini Melakukan perbandingan kinerja algoritma tanda tangan digital Elliptic Curve Digital Signature Algorithm (ECDSA) menggunakan fungsi hash yang berbeda dan menganalisis performa waktu proses dari awal hingga akhir antara algoritma Elliptic Curve Digital Signature Algorithm (ECDSA) menggunakan fungsi hash SHA-1 dan algoritma Elliptic Curve Digital Signature Algorithm (ECDSA) mengunakan fungsi hash Keccak. Parameter pengujian yang dilakukan membandingkan algoritma Elliptic Curve Digital Signature Algorithm (ECDSA) fungsi hash SHA-1 dan Keccak pada saat proses pembangkitan kunci (key generation), tahap penandatanganan (signature generation), dan tahap verifikasi tanda tangan digital (verifying).

Kata Kunci: *Tanda Tangan Digital, Algoritma ECDSA, SHA-1, Keccak (SHA-3)*

Abstract

Data or document sent over the internet is very vulnerable to attack or modification and it is very difficult to prove the authenticity of data or documents, so with the development of a security system a cryptography is formed which is used to verify the authenticity and truth of a data called a digital signature. Digital signatures are combined with hash functions to create signatures on data. The signature algorithm that is often used is the Diffle-Helman Digital Signature Algorithm and is better known as the Digital Signature Algorithm (DSA). The Digital Signature Algorithm (DSA) algorithm was developed into an Elliptic Curve Digital Signature Algorithm (ECDSA) algorithm that uses an elliptic curve. This study compares the great performance of the digital signature Elliptic Curve Digital Signature Algorithm (ECDSA) using different hash functions and the performance of analyzing processing time from start to finish between ECDSA algorithm using SHA-1 hash function and ECDSA algorithm using Keccak hash function. Parameter testing is carried out comparing the ECDSA algorithm of SHA-1 and Keccak hash functions during the key generation process (key generation), the signing stage (signature generation), and the digital signature leveraging stage (verification).

Keywords: *Digital Signature, Elliptic Curve Digital Signature Algorithm (ECDSA), SHA-1, Keccak (SHA-3)*

1. Pendahuluan

Perkembangan teknologi yang sangat pesat, memerlukan keamanan data yang lebih baik dan terjamin keamanannya, tentu sangat dibutuhkan untuk menjaga sebuah data. Kecepatan akses internet dan banyaknya sebuah data yang meningkat akan menimbulkan ancaman bagi pengguna internet, adanya ancaman tersebut bidang keamanan akan semakin berkembang. Kerahasiaan dan integritas data yang akan dikirim, merupakan tujuan yang paling penting dalam

sebuah sistem *security*, dimana pada proses tersebut merupakan proses terpenting dalam pertukaran sebuah data.

Sebuah data atau dokumen yang dikirimkan melalui internet sangat rentan terhadap serangan atau modifikasi serta sangat sulit untuk membuktikan keaslian data atau dokumen tersebut, maka dengan perkembangan sistem keamanan terbentuklah sebuah mekanisme kriptografi yang digunakan untuk memverifikasi keaslian dan kebenaran dari sebuah data yang disebut dengan tanda tangan digital. Tanda tangan digital, dipadukan dengan fungsi hash untuk membuat sebuah tanda tangan pada suatu data.Metode ini digunakan untuk menjaga keamanan dan konsistensi data adalah dengan menggunakan tanda tangan digital yang ditempelken pada *file*, data, *message* yang ingin di lindungi.

Algoritma tanda tangan digital yang sering digunakan adalah *Diffle-Helman Digital Signature Algorithm* dan lebih dikenal sebagai *Digital Signature Algorithm* (DSA). Algortima ini menggunakan *prime finite field* dengan bilangan prima yang cukup besar sehingga menyulitkan pihak yang ingin menduplikasi tanda tangan yang dihasilkan[1]. Algoritma *Digital Signature Algorithm* (DSA) dikembangkan menjadi algoritma *Elliptic Curve Digital Signature Algoritmh* (ECDSA) yang menggunakan *elliptic curve* atas suatu *prime finite field* sehingga lebih mudah digunakan dalam proses menentukan kunci. Berdasarkan penelitian sebelumnya yang dilakukan oleh Dicky Wizanajani (2018), telah melakukan penelitian untuk melihat perbedaan dari tiga algoritma tanda tangan digital yang sering digunakan yaitu algoritma *Digital Signature Algorithm* (DSA), *Rivest Shamir Adleman* (RSA) dan *ECC-based* algoritma, dari ketiga algoritma tersebut dibandingkan berdasarkan kecepatan pada memverifikasi tanda tangan[2]. Rezania Agramanisti A. (2016) melakukan penelitian tanda tangan digital menggunakan algoritma RSA dan fungsi hash keccak dengan melakukan pengujian dari proses tanda tangan, verifikasi dan pembangkitan kunci pada algoritma RSA dan melakukan pengujian terhadap aspek keamanan tanda tangan tersebut[3].

Perbedaan penelitian ini dengan penelitian sebelumnya yaitu menggunakan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) dengan dua fungsi hash yang berbeda disetiap implementasinya lalu membandingkan kedua hasil fungsi hash tersebut. Penelitian ini melakukan perbandingan antara dua fungsi hash yaitu fungsi hash SHA-1 dan Keccak yang diimplementasikan kedalam algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) dimana pada penelitian yang dilakukan oleh Hanifah Azhar (2013) melakukan perbandingan fungsi hash MD5 dengan SHA-1 tanpa menggunakana algoritma tanda tangan digital dalam pengimplementasiannya dan mendapatkan hasil bahwa SHA-1 lebih aman digunakan untuk pengamanan kata sandi dan message digest yang dihasilkan SHA-1 lebih panjang sehingga saat dilakukan nya serangan maka membutuhkan waktu yang lebih lama[4]. Penelitian ini dilakukan untuk diimplentasikan kedalam tanda tangan digital menggunakan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA). Berkembangnya teknologi fungsi hash mengalami pembaharuan berdasarkan kompetisi terbuka yang dilakukan oleh NIST yang diberi nama Keccak (SHA-3) algoritma fungsi hash satu arah yang mengadopsi sponge structure untuk melakukan permutasi terhadap state dengan panjang yang tetap. Penelitian oleh Soleh (2011) membahas tentang studi dan implementasi algortima fungsi hash keccak yang diimplementasikan dalam sebuah software pada penggunaaan *SIMD (single instruction multiple data)* yang digunakan untuk menangani jumlah data yang sangat banyak dalam sebuah paralel secara efisien[5].

Penelitian yang dilakukan oleh Kusuma dan Darji (2014) melakukan implementasi algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) untuk membuat suatu tanda tangan digital pengujian perfoma pada file dokumen bereksistensi txt[6]. Pemanfaatan tanda tangan digital menggunakan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) dan Keccak pada penelitian D.A. Rahman (2019) yang menerapkan tanda tangan digital pada teks editor dimana algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) memiliki keungulan yaitu memiliki kunci yang lebih pendek[7]. Kedua penelitian tersebut menerapkan algoritma kriptografi kunci publik yang digunakan pada tanda tangan digital. Tanda tangan dibuat menggunakan nilai hash dari data atau sebuah *file*. Nilai hash tersebut yang akan menjaga konsistensi data dankeamanan data.Penggunaan fungsi hash ditetap kan pada data atau *file* secara utuh kemudian nilai hash dipakai untuk menandatangi *file* atau data tersebut[8].

Penelitian ini dilakukan untuk menganalisis kinerja proses waktu dari awal hingga akhir antara algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) menggunakan fungsi hash SHA-1 dan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) mengunakan

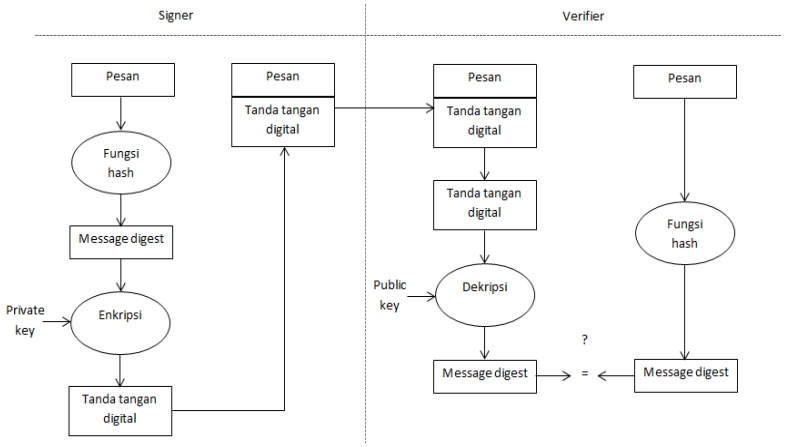
fungsi hash Keccak. Melakukan perbandingan untuk mengetahui kinerja algoritma pada tanda tangan digital dan apakah kedua fungsi hash tersebut dapat diimplementasikan menggunakan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA).

2. Metode Penelitian

2.1 Tanda Tangan Digital (Digital Signature)

Pada era digital, semua pesan sudah berbentuk data elektronik, sehingga pemberian tanda tangan pada dokumen cetak jarang dilakukan. Fungsi tanda tangan pada dokumen cetak tetap diterapkan untuk otentifikasi pada pesan digital, baik pesan yang ditransmisikan melalui saluran komunikasi maupun dokumen elektronik yang disimpan didalam komputer. Tanda tangan pada data digital dinamakan tanda tangan digital (digital signature), yang dimaksud dengan tanda tangan digital bukan lah tanda tangan yang di digitasi dengan alat scanner atau yang dikenal sebagai digitized signature atau tanda tangan yang dibuat menggunakan pena elektronik, tetapi yang dimaksud tanda tangan digital dalam kriptografi adalah suatu nilai kriptografi yang bergantung pada isi pesan dan pengirim pesan. Pesan yang isinya berbeda, meskipun dari pengirim yang sama, akan memiliki tanda tangan digital yang berbeda.

Tanda tangan digital yang valid yaitu yang memenuhi syarat-syarat dan memberikan penerima pesan dengan alasan yang sangat kuat untuk percaya bahwa pesan itu dibuat oleh pengirim yang dikenal (autentikasi), dan pesan itu tidak diubah dalam transit (integritas). Sistem tanda tangan digital menggunakan algoritma kunci publik dan juga fungsi hash untuk menjamin authentic city, integrity dan non-repudiation pesan yang ditandatangani. Tanda tangan digital juga dapat memastikan aspek non-repudiation sehingga penulis pesan tidak bisa menyangkal dikemudian hari bahwa pesan tersebut bukan ditulis olehnya [9]. Berikut adalah skema dari proses tanda tangan digital menggunakan kombinasi kunci public dan fungsi hash [8].



Gambar 1. Proses Tanda Tangan Digital

2.2 Elliptic Curve Digital Signature Algorithm (ECDSA)

Elliptic Curve Digital Signature Algorithm (ECDSA) merupakan sebuah algoritma tanda tangan digital menggunakan pasangan kunci berdasarkan algoritma kriptografi kunci publik Elliptic Curve Cryptographya (ECC). Elliptic Curve Cryptography (ECC) adalah kunci publik kriptografi yang menggunakan Elliptic Curve Discrete Logarithma Problem (ECDLP) sebagai dasar matematikanya. Algortima Elliptic Curve Digital Signature Algorithma (ECDSA) merupakan algoritma hasil pengembangan dari algortima Digital Signature Algorithm (DSA). Keamanan terletak pada kesulitan pemecahan persamaan kurva eliptik[10]. Tingkat keamanan algortima Elliptic Curve Digital Signature Algorithm (ECDSA) dinilai cukup baik digunakan dalam sebuah jaringan. Elliptic Curve Digital Signature Algorithm (ECDSA) membutuhkan tempat yang lebih sedikit dibandingkan algaritma RSA. Untuk tingkat keamanan yang sama, Elliptic Curve Digital Signature Algorithm (ECDSA) memiliki ukuran tanda tangan dan kunci publik yang lebih kecil, sehingga apabila ditambahkan pada packet header, tidak akan terlalu membebani dengan menambah ukuran paket yang akan dikirim.

Algortitma Elliptic Curve Digital Signature Algorithm (ECDSA) terdiri dari 3 tahap yaitu tahap Pembangkitan Kunci, pemberian tanda tangan dan tahap verifikasi yaitu sebagai berikut[11]:

a. *Key Generation* (Pembangkitan Kunci)

Key Generation adalah proses pertama dalam algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA)yang dilakukan untuk membangkitkan kunci pada pesan atau dokumen yang akan dikirim. Proses ini dilakukan untuk mendapatkan kunci publik dan kunci private.

b. *Signing* (Pemberian Tanda Tangan)

Signing adalah proses membangkitkan tanda tangan dari sebuah pesan, dan alur pemberian tanda tangan pada algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA). Dimana pada tahap ini disisipkan fungsi hash yang berfungsi sebagai *message digest*. Tanda tangan digital dibentuk menggunakan hasil hash dari pesan yang telah di hash ditambah dengan kunci private yang telah dibangkitkan.

c. *Verifying* (Verifikasi Tanda Tangan Digital)

Verifying adalah proses melakukan verifikasi tanda tangan digital dari pesan dan dibutuhkan kunci publik sebagai penanda tangan.

2.3 Fungsi Hash

Fungsi hash dalam kriptografi adalah fungsi hash yang berupa sebuah algoritma yang mengambil sejumlah blok data dan mengembalikan bit string berukuran tetap. String yang dihasilkan merupakan hash value. Fungsi hash memiliki banyak kegunaan, terutama untuk menjaga konsistensi dan keamanan suatu data. Fungsi hash dalam kriptografi adalah fungsi hash yang berupa sebuah algoritma yang mengambil sejumlah block data dan mengembalikan bit string berukuran tetap. String yang dihasilkan tersebut merupakan nilai hash. Perubahan yang terjadi pada data walaupun sangat kecil, akan menyebabkan perubahan yang sangat banyak pada hasil nilai hash. Data yang di hash disebut pesan dan nilai hash disebut message digest[4].

2.3.1 SHA-1

Algoritma SHA-1 digunakan untuk menghitung nilai message digest dari sebuah pesan, dimana pesan tersebut memiliki panjang maksimal 264 bit. Algoritma ini juga menggunakan message schedule yang terdiri dari 80 elemen 32-bit word, lima buah variabel 32-bit, dan variabel penyimpanan nilai hash 5 buah word 32-bit. Hasila khir algoritma SHA-1 adalah sebuah message digest sepanjanga160-bit. Langkah-langkah pembuatan message digest dengan SHA-1, yaitu sebagai berikut [8]:

1. Penambahan bit-bit pengganjal (*padding bits)*
   Pesan ditambahkan dengan sejumlah bit pengganjal sedemikian sehingga panjang pesan (dalam satuan bit) $\equiv$ dengan 448 modulo 512.
2. Penambahan nilai panjang pesan semula
   Pesan yang telah diberi bit-bit pengganjal selanjutnya ditambah lagi dengan 64 bit yang menyatakan panjang pesan semula.
3. Inisialisasi penyangga *Message Digest*
   SHA membutuhkan 5 buag penyangga *(buffer)* yang masing-masing panjangnya 32 bit.
4. Pengolahan pesan dalam blok berukuran 512 bit
   Pesan dibagi menjadi L blok buah blok yang masing-masing panjangnya 512 bit.

2.3.2. Keccak (SHA-3)

Keccak (SHA-3) merupakan algoritma fungsi hash satu arah yang mengadopsi sponge structure untuk melakukan permutasi terhadap state dengan Panjang yang tetap. Fungsi hash Keccak merupakan pemenang dari kompetisi terbuka untuk SHA-3 lainnya oleh NIST. Keccak menggunakan inner state selama proses hashing berlangsung.

A. Konstruksi Spons

Fungsi spons pada keccak berbasis pada kontruksi spon yang melakukan permutasi terhadap *state* dengan panjang tetap (b). Fungsi spon yang digunakan terdiri dari *padding*, *absorbing* dan *squeezing.* Setiap state memiliki panjang sesuai dengan panjang permutasi, yaitu b[3].

B. Fungsi Permutasi *Keccak-f*

Fungsi permutasi keccak-f merupakan fungsi utama dalam keccak. Panjang tetap dari string yang akan dipermutasi disebut dengan lebar permutasi, dinotasikan dengan (b). banyakny aiterasi dari transformasi internal disebut dengan round yang dinotasikan dengan

(n). Fungsi ni menggambil *state* sebagai masukkan, dan melakukan sejumlah operasi permutasi yang terdiri dari 5 tahapan operasi yaitu: *non-linearity* (chi), *diffusion* (theta), *inter-slince dispersion* (rho), *disturbing horizontal/vertical aligntment* (pi), dan *break symetri* (iota)[3].

3. Hasil Penelitian dan Pembahasan

Pada tahapan ini melakukan pengujian sistem yang di implementasikan berbasis web. Pengujian dilakukan yaitu sebagai berikut:

3.1 Pengujian Sistem

Pengujian terbagi menjadi menjadi 2 yaitu pengujian pada algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) menggunakan fungsi hash SHA-1 dan pengujian algoritma ECDSA menggunakan fungsi hash Keccak. Pengujian dilakukan sebanyak 5 kali dengan inputan text yang berbeda.

3.1.1 Pengujian Algoritma ECDSA menggunakan Fungsi Hash SHA-1

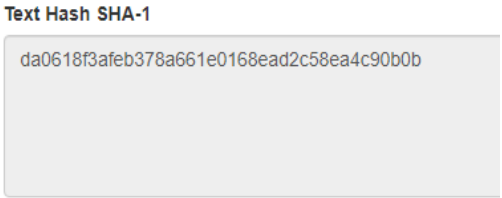
Inputan Text 1 : Rapat akan diselenggarakan hari Senin, jam 9 pagi.

Gambar 2. Hasil inputan text 1 dihash menggunakan fungsi hash SHA-1

Tabel 1. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash SHA-1 inputan text 1

Percobaan ke	Waktu				
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.2565	0.0009	0.0009	Valid
2.	233	0.1718	0.0007	0.0015	Valid
3.	283	0.1694	0.0019	0.0019	Valid
4.	409	0.1453	0.0025	0.0033	Valid
5.	571	0.1884	0.0037	0.0079	Valid
Rata - rata		0,9314	0,0097	0,0155	

Inputan Text 2 : Universitas Muhammadiyah Malang



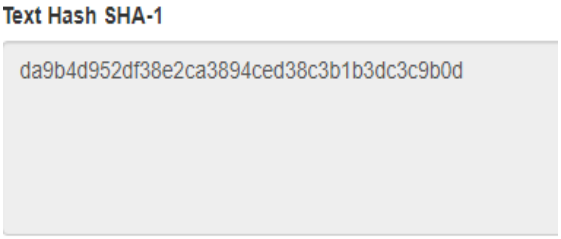
Gambar 3. Hasil inputan text 2 dihash menggunakan fungsi hash SHA-1

Tabel 2. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash SHA-1 inputan text 2

Percobaan ke	Waktu				
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.4691	0.0009	0.0023	Valid
2.	233	0.3686	0.0011	0.0036	Valid

3.	283	0.2551	0.0011	0.0044	Valid
4.	409	0.3438	0.0029	0.0072	Valid
5.	571	0.5357	0.0042	0.0197	Valid
Rata - rata		1,9723	0,0102	0,0372	

Inputan Text 3: Dalam rangka peringatan hari Sumpah Pemuda, akan diadakan beberapa kegiatan. Berbagai acara yang akan diadakan mencerminkan ketangguhan pemuda Indonesia dalam menghadapi tangtangan zaman. Datang dan jadikanlah sejarah dengan mengunjungi Gedung Pemuda pada Kamis 28 oktober 2021.



Gambar 4. Hasil inputan text 3 dihash menggunakan fungsi hash SHA-1

Tabel 3. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash SHA-1 inputan text 3

Percobaan ke		Waktu			
(1)	Panjang kunci kurva Ecc (second) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.3501	0.0007	0.0015	Valid
2.	233	0.2599	0.0009	0.0016	Valid
3.	283	0.2478	0.0010	0.0018	Valid
4.	409	0.2501	0.0019	0.0038	Valid
5.	571	0.1926	0.0049	0.0068	Valid
Rata - rata		1,3005	0,0094	0,0155	

Inputan Text 4 : Perkenalkan nama saya iqbaal.
Umur 22 tahun.
Asal Jakarta.
Jurusan Teknik Informatika.

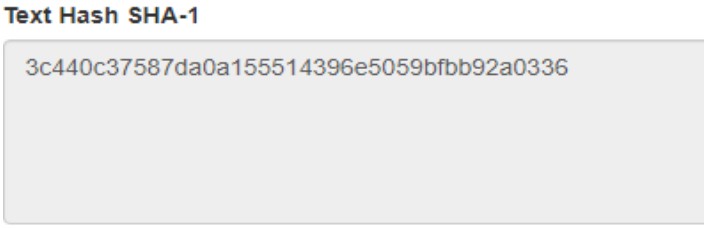
Gambar 5. Hasil inputan text 4 dihash menggunakan fungsi hash SHA-1

Tabel 4. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash SHA-1 inputan text 4

Percobaan ke		Waktu			
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	1.1710	0.0005	0.0008	Valid
2.	233	0.8396	0.0007	0.0012	Valid
3.	283	0.3020	0.0015	0.0036	Valid

4.	409	0.2654	0.0026	0.0047	Valid
5.	571	2.9117	0.0054	0.0129	Valid
Rata - rata		5,4897	0,0107	0,0232	

Inputan Text 5 : Rinaldi Munir
Dosen Program Studi Teknik Informatika Bandung
Buku Kriptografi
Edisi Kedua
Penerbit Informatika
Tahun 2019



Gambar 6. Hasil inputan text 5 dihash menggunakan fungsi hash SHA-1

Tabel 5. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash SHA-1 inputn text 4

Percobaan ke			Waktu		
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.3412	0.0007	0.0042	Valid
2.	233	0.2186	0.0010	0.0016	Valid
3.	283	0.2252	0.0016	0.0019	Valid
4.	409	0.2303	0.0017	0.0072	Valid
5.	571	0.2154	0.0035	0.0094	Valid
Rata-rata		1,2307	0,0085	0,0243	

3.1.2 Pengujian Algoritma ECDSA menggunakan Fungsi Hash Keccak

Inputan Text 1 : Rapat akan diselenggarakan hari Senin
jam 9 pagi.

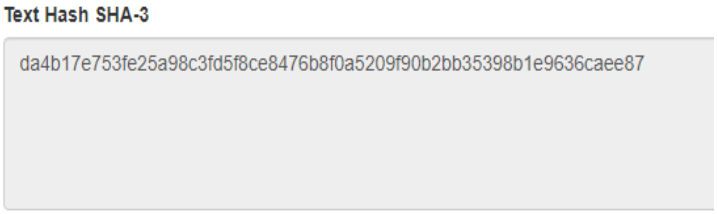
Gambar 7. Hasil inputan text 1 dihash menggunakan fungsi hash Keccak

Tabel 6. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash Keccak inputan text 1

Percobaan ke			Waktu		
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.1677	0.0006	0.0009	Valid
2.	233	0.1612	0.0007	0.0013	Valid
3.	283	0.1750	0.0010	0.0021	Valid

4.	409	0.1527	0.0016	0.0031	Valid
5.	571	0.1633	0.0036	0.0119	Valid
Rata-rata		0,8199	0,0075	0,0193	

Inputan Text 2 : Universitas Muhammadiyah Malang

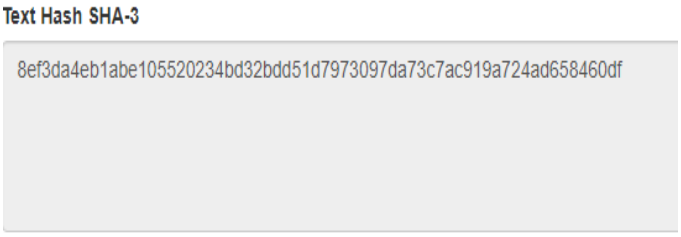


Gambar 8. Hasil inputan text 2 dihash menggunakan fungsi hash Keccak

Tabel 7. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash Keccak inputan text 2

Percobaan ke	Waktu				
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.2665	0.0005	0.0008	Valid
2.	233	0.2724	0.0007	0.0016	Valid
3.	283	0.2549	0.0010	0.0019	Valid
4.	409	0.3939	0.0026	0.0062	Valid
5.	571	0.4345	0.0046	0.0106	Valid
Rata - rata		1,6222	0,0094	0,0211	

Inputan Text 3 : Dalam rangka peringatan hari Sumpah Pemuda, akan diadakan beberapa kegiatan. Berbagai acara yang akan diadakan mencerminkan ketangguhan pemuda Indonesia dalam menghadapi tangtangan zaman. Datang dan jadikanlah sejarah dengan mengunjungi Gedung Pemuda pada Kamis 28 oktober 2021.



Gambar 9. Hasil inputan text 3 dihash menggunakan fungsi hash Keccak

Tabel 8. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash Keccak inputan text 3

Percobaan ke	Waktu				
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.2228	0.0007	0.0012	Valid
2.	233	0.2141	0.0009	0.0013	Valid
3.	283	0.1982	0.0010	0.0020	Valid
4.	409	0.2144	0.0021	0.0045	Valid
5.	571	0.2534	0.0037	0.0084	Valid
Rata-rata		1,1029	0,0084	0,0174	

Inputan Text 4 : Perkenalkan nama saya iqbaal.
Umur 22 tahun.
Asal Jakarta.
Jurusan Teknik Informatika.

Gambar 10. Hasil inputan text 4 dihash menggunakan fungsi hash Keccak

Tabel 9. Hasil scenario pengujian algoritma ECDSAmenggunakan fungsi hash Keccak inputan text 4

Percobaan ke	Waktu				
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.3657	0.0005	0.0009	Valid
2.	233	0.4022	0.0029	0.0023	Valid
3.	283	0.3343	0.0011	0.0030	Valid
4.	409	0.8775	0.0045	0.0036	Valid
5.	571	0.4713	0.0053	0.0188	Valid
Rata - rata		2,4510	0,0143	0,0286	

Inputan Text 5 : Rinaldi Munir
Dosen Program Studi Teknik Informatika Bandung
Buku Kriptografi
Edisi Kedua
Penerbit Informatika
Tahun 2019

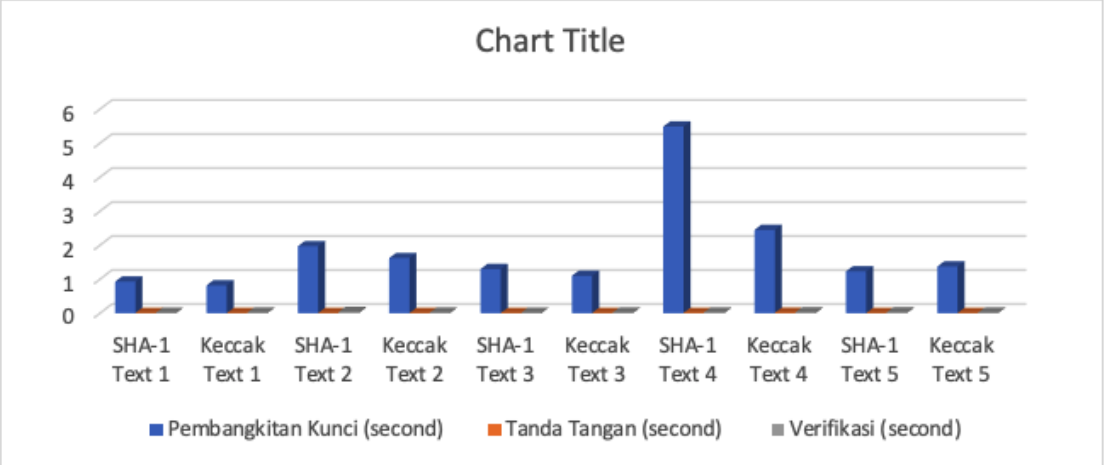
Gambar 11. Hasil inputan text 4 dihash menggunakan fungsi hash Keccak

Tabel 10. Hasil scenario pengujian algoritma ECDSA menggunakan fungsi hash Keccak inputan text 5

Percobaan ke	Waktu				
(1)	Panjang kunci kurva Ecc (bit) (2)	Pembangkitan Kunci (second) (3)	Tanda Tangan (second) (4)	Verifikasi (second) (5)	Hasil Verfikasi (6)
1.	163	0.2661	0.0005	0.0009	Valid
2.	233	0.2830	0.0012	0.0023	Valid
3.	283	0.2711	0.0011	0.0034	Valid
4.	409	0.2673	0.0019	0.0037	Valid
5.	571	0.2868	0.0032	0.0094	Valid
Rata- rata		1,3743	0,0079	0,0197	

3.2 Hasil Analisa Perbandingan ECDSA menggunakan Fungsi Hash SHA-1 dan ECDSA menggunakan Fungsi Hash Keccak

Hasil yang diperoleh setelah melakukan 5 kali percobaan dengan menginputkan data berupa text yang berbeda pada setiap percobaan, mendapatkan hasil bahwa algoritma tanda tangan digital *Elliptic Curve Digital Signature Algorithm* (ECDSA) dapat diterapkan menggunakan fungsi hash SHA-1 atau fungsi hash Keccak. Berikut pada Gambar 12. merupakan hasil grafik perbandingan waktu pembangkitan kunci, penandatanganan dan verifikasi terhadap text 1 menggunakan algoritma ECDSA fungsi hash SHA-1 keccak.



Gambar 12. Grafik Perbandingan Hasil Waktu Pembangkitan Kunci, Tanda Tangan, dan Verifikasi Algoritma ECDSA menggunakan fungsi hash SHA-1 dan Keccak pada Text 1, Text 2, Text 3, Text 4, dan Text 5

Menurut grafik diatas, dapat disimpulkan bahwa implementasi algoritma ECDSA menggunakan fungsi hash Keccak mendapatkan waktu yang sedikit lebih cepat pada pembangkitan kunci text 1 dengan waktu 0,8199 *milli second*, untuk waktu penandatangan algoritma ECDSA menggunakan fungsi hash Keccak juga mendapatkan waktu yang lebih cepat pada text 1 dengan waktu 0,0075 *milli second*, dan untuk waktu verifikasi tanda tangan algoritma ECDSA menggunakan fungsi hash SHA-1 mendapatkan waktu sedikit yang cepat pada text 1 dan text 3 dengan waktu 0,0155 *milli second*.

Berdasarkan percobaan yang telah dilakukan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) menggunakan fungsi hash Keccak mendapatkan hasil waktu yang sedikit lebih singkat dibandingkan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) mengunakan fungsi hash SHA-1 pada pemproses-an waktu pembangkitan kunci, waktu penandatanganan dan waktu verifikasi tanda tangan digital pada text yang diinputkan. Fungsi hash SHA-1 dan fungsi hash Keccak dapat diterapkan untuk algoritma tanda tangan digital, namun untuk fungsi hash SHA-1 hanya dapat menghasilkan nilai hash 160 bit untuk panjang inputan text yang diolah. Pada keccak panjang nilai hash dapat berupa 224, 256, 384 atau 512 bit.

4. Kesimpulan

Dari tahapan penelitian diatas, maka dapat ditarik kesimpulan yaitu :

1. Melakukan perbandingan kinerja algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) menggunakan fungsi hash SHA-1 dan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) mengunakan fungsi hash Keccak.
2. Parameter pengujian yang dilakukan membandingkan algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) fungsi hash SHA-1 dan Keccak yang berfokus pada waktu proses penandatanganan.
3. Algoritma tanda tangan digital *Elliptic Curve Digital Signature Algorithm* (ECDSA) dapat diterapkan menggunakan fungsi hash SHA-1 atau fungsi hash Keccak.

4. Algoritma *Elliptic Curve Digital Signature Algorithm* (ECDSA) menggunakan fungsi hash Keccak mendapatkan hasil waktu yang sedikit lebih singkat saat pembangkitan kunci pada text 1 dengan waktu 0,8199 *milli second*, untuk waktu penandatangan yang lebih singkat, pada text 1 dengan waktu 0,0075 *milli second*, dan untuk waktu verifikasi tanda tangan algoritma ECDSA menggunakan fungsi hash SHA-1 mendapatkan waktu sedikit yang cepat pada text 1 dan text 3 dengan waktu 0,0155 *milli second*.

Referensi

[1] V. Kusuma, J. Matematika, dan F. Matematika, "Elliptic Curve dan Implementasinya pada Algoritma Tanda Tangan Digital," *J. Sains dan Seni ITS*, vol. 3, no. 2, hal. 3–6, Sep 2014.
[2] D. W. R, "Perbandingan Algoritma Berbasis Elliptic Curve Cryptography Dengan RSA dan DSA Pada Tanda Tangan Digital," *Inform. STEI ITB*, vol. 2, no. 1, hal. 1–7, 2007.
[3] R. A. Azdy, "Tanda tangan Digital Menggunakan Algoritme Keccak dan RSA," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 5, no. 3, hal. 184–191, 2016, doi: 10.22146/jnteti.v5i3.255.
[4] H. Azhar, "Perbandingan Algoritma Fungsi Hash MD5 dengan SHA-1," 2013.
[5] M. Y. Soleh dan S. Teknik, "Studi dan Implementasi Algoritma Keccak," 2011.
[6] V. Kusuma, J. Matematika, dan F. Matematika, "Elliptic Curve dan Implementasinya pada Algoritma Tanda Tangan Digital," vol. 3, no. 2, hal. 3–6, 2014.
[7] D. A. Rahman, "Pemanfaatan Tanda Tangan Digital Menggunakan ECDSA dan Keccak pada Teks Editor," 2018.
[8] K. Yauris, "Penggunaan Fungsi Hash dan Tanda Tangan Digital dalam Transmisi Data," *Makal. ke-2 IF4020 Kriptografi, Semester II Tahun 2015/2016*, 2016.
[9] R. Munir, "Kriptografi." Informatika Bandung, Bandung, hal. 880, 2019.
[10] T. N. Ovari, "Implementasi Elliptic Curve Digital Signature Algorithm (ECDSA) untuk Mengatasi Black Hole dan Worm Hole Attack pada Komunikasi V2V di Lingkungan VANETs," hal. 93, 2017.
[11] V. K. Pesan *et al.*, "Implementasi Algoritma Ecdsa Untuk Pengamanan E-Mail 2 . Tinjauan Pustaka 2 . 3 ECDSA (Elliptical Curve Digital Signature)."