Implementasi Tanda Tangan Digital Menggunakan ECDSA dan Keccak pada Layanan *Electronic Health*

Christzen Leonardy - 13517125[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13517125@std.stei.itb.ac.id

Abstraksi—Pada era digital saat ini, banyak sekali informasi yang dibagikan menggunakan internet. Informasi yang sebelumnya disimpan menggunakan media kertas mulai beralih ke media elektronik. Salah satu informasi yang dibagikan menggunakan internet yaitu jejak rekam medis atau riwayat medis. Informasi tersebut sangat mudah dihadang oleh penyerang yang kemudian mengubahnya atau memanfaatkannya untuk mendapatkan informasi lainnya. Hal tersebut sangat tidak diinginkan, terutama dalam informasi riwayat medis. Untuk menghindari hal tersebut, perlu diimplementasi sebuah mekanisme untuk menjaga *authenticity*, *integrity*, dan *access* terhadap data tersebut. ECDSA dapat memastikan *authenticity* dan *integrity* dari riwayat medis. Secara tidak langsung, kedua aspek tersebut juga dapat memastikan pemberian *access* sesuai dengan keinginan pasien terkait. Sehingga dengan diterapkannya ECDSA dalam layanan e-health, jejak rekam media pasien akan lebih terjaga keamanannya.

Kata kunci—kriptografi, tanda tangan digital, ecdsa, e-health

I. Pendahuluan

Pada era digital saat ini, internet sangat luas digunakan untuk membagikan informasi. Media kertas yang sering digunakan pun mulai ditinggalkan dan informasi mulai disimpan dalam media elektronik. Hal tersebut dilakukan karena efisiensi penyimpanan pada media elektronik lebih tinggi daripada penyimpanan pada media kertas.

Electronic health adalah layanan yang memungkinkan penyedia layanan kesehatan dan pasien membagi informasi riwayat medis di pusat layanan kesehatan seperti di rumah sakit, klinik, dan lain-lain. Institusi kesehatan dan peneliti mengembangkan layanan ini untuk meningkatkan kualitas layanan kesehatan yang dapat diberikan. Tetapi, layanan ini memiliki masalah pada data privasi pasien yang sering menjadi sasaran penyerang.

Keamanan riwayat medis pasien perlu dipertahankan agar tidak disalahgunakan. Hal tersebut dikarenakan informasi tersebut adalah informasi yang sensitif. Pemilik informasi tersebut juga seharusnya memiliki kuasa untuk memberikan akses kepada orang lain.

Tanda tangan digital adalah salah satu skema yang dapat memastikan otentikasi, keaslian, dan anti-penyangkalan pesan. Hal tersebut membuat tanda tangan digital cocok digunakan pada layanan e-health. Skema ini dapat menyelesaikan masalah-masalah yang ada di dalam layanan e-health.

II. Dasar Teori

A. Tanda Tangan Digital

Tanda tangan digital adalah skema matematika yang digunakan untuk menunjukkan keaslian sebuah pesan. Tanda tangan digital adalah sidik jari digital yang unik bagi setiap orang yang digunakan untuk mengidentifikasi pengirimnya dan melindungi pesan saat proses pengiriman. Tanda tangan digital bekerja menggunakan kriptograf kunci publik yang memiliki sebuah pasangan kunci yaitu kunci privat dan kunci publik. Kunci privat mengenkripsi pesan dan hanya tersedia oleh pengirim atau *signer*. Kunci publik digunakan untuk mendekripsi pesan dan diberikan kepada penerima.

Kriptografi dapat memberikan 4 aspek keamanan yaitu kerahasiaan pesan (*confidentiality/secrecy*), otentikasi (*authentication*), keaslian pesan (*data integrity*), dan anti-penyangkalan (*non-repudiation*). Tanda tangan digital dapat memberikan 3 aspek keamanan yang dapat diberikan oleh kriptografi yaitu:

1) Otentikasi (*authentication*)

Ketika penerima pesan melakukan verifikasi terhadap tanda tangan digital menggunakan kunci publik pengirim, pengirim yakin bahwa tanda tangan digital tersebut hanya dapat diberikan oleh pengirim yang bersangkutan dengan kunci privat yang dimilikinya.

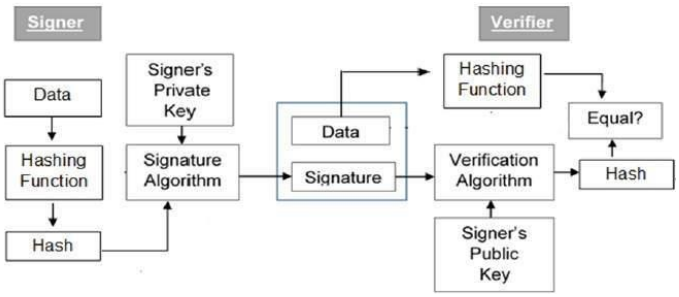
2) Keaslian pesan (*data integrity*)

Jika pesan diubah saat proses pengiriman ke penerima, maka saat pesan tersebut tidak akan terverifikasi pada sisi penerima. Hal tersebut dikarenakan hash dari pesan tersebut tidak sesuai dengan hasil algoritma verifikasi. Sehingga penerima dapat mengabaikan pesan tersebut karena sudah tidak sesuai dengan yang seharusnya.

3) Anti-penyangkalan (*non-repudiation*)

Kunci privat hanya dimiliki oleh pengirim. Oleh karena itu, hanya pengirim tersebut yang dapat memberikan tanda tangan digital tersebut pada sebuah pesan. Penerima dapat menunjukkan tanda tangan digital yang diberikan sebagai bukti jika pengirim menyangkal pernah memberikan pesan tersebut.

Model dari skema tanda tangan digital dapat dilihat pada gambar 1.



Gambar 1 Model Skema Tanda Tangan Digital
https://www.tutorialspoint.com/cryptography/images/model_digital_signature.jpg

B. Kriptografi Kunci Publik

Kriptografi kunci publik atau sering disebut juga sebagai kriptografi kunci nirsimetri adalah skema enkripsi yang menggunakan dua kunci yang berbeda untuk enkripsi dan dekripsi. Walaupun kedua kunci berbeda, keduanya memiliki relasi secara matematis. Berbeda dengan skema enkripsi pada umumnya yang menggunakan sebuah kunci untuk melakukan enkripsi dan dekripsi, setiap kunci pada kriptografi kunci publik memiliki satu fungsi saja. Kunci privat digunakan untuk enkripsi dan kunci publik digunakan untuk dekripsi.

C. Elliptic Curve Digital Signature Algorithm

ECDSA adalah skema tanda tangan digital kriptografi yang aman berdasarkan *Elliptic-Curve Cryptography* (ECC). ECDSA memanfaatkan operasi matematika pada grup siklik dari kurva eliptik dan kesulitan permasalahan *Elliptic-Curve Discrete Logarithm Problem* (ECDLP). Algoritma *sign* dan *verify* pada ECDSA bergantung pada operasi perkalian titik pada kurva eliptik. Kunci dan tanda tangan yang dihasilkan ECDSA lebih pendek daripada yang dihasilkan RSA untuk mendapatkan tingkat keamanan yang sama. Persamaan kurva eliptik adalah $y^2 = x^3 + ax + b \, mod \, p$. Sehingga parameter yang digunakan pada kurva eliptik yaitu a, b, p, n, dan titik basis (G). Ada tiga algoritma yang kami implementasikan untuk ECDSA yaitu pembangkitan kunci, *sign*, dan *verify sign*.

1) Pembangkitan Kunci

Pasangan kunci pada ECDSA terdiri dari sebuah kunci privat (bilangan bulat) dan kunci publik (titik pada kurva eliptik). Pembangkitan kunci privat (*privKey*) dilakukan dengan membangkitkan bilangan bulat sembarang di antara 0 sampai dengan n - 1. Kunci privat tersebut kemudian digunakan untuk membangkitkan kunci publik (*pubKey*) yang memanfaatkan operasi perkalian titik kurva eliptik dengan mengalikan kunci privat dengan titik basis atau $pubKey = privKey \times G$.

2) *Sign*

Algoritma tanda tangan digital ECDSA menerima sebuah pesan (*msg*) dan sebuah kunci privat untuk menghasilkan sebuah tanda tangan digital yang terdiri dari sepasang bilangan bulat $\{r, s\}$. Langkah-langkah yang dilakukan pada algoritma ini adalah sebagai berikut.

1. Menghitung *hash* dari pesan dengan menggunakan fungsi hash kemudian diubah menjadi bilangan bulat
2. Membangkitkan sebuah bilangan bulat sembarang *k* di antara 0 sampai dengan n - 1
3. Menghitung titik sembarang $R = k \times G$ dan mengambil koordinat x-nya $r = R.x$
4. Menghitung bukti tanda tangan dengan rumus $s = k^{-1} \times (h + r \times privKey) \, mod \, n$
5. Kembalikan tanda tangan digital$\{r, s\}$

3) *Verify Sign*

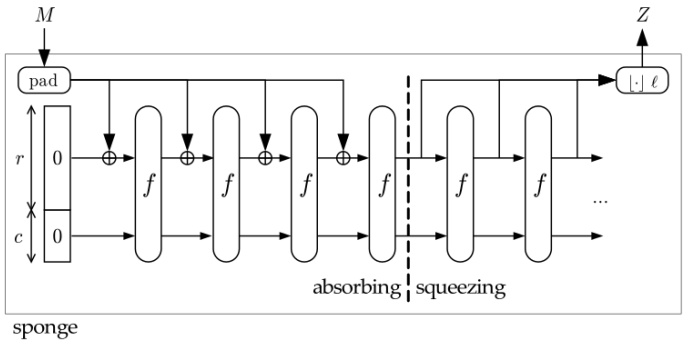
Algoritma untuk memverifikasi tanda tangan digital ECDSA membutuhkan man berupa pesan (*msg*) dan tanda tangan digital$\{r, s\}$yang dihasilkan oleh algoritma *sign* serta kunci publik (*pubKey*). Langkah-langkah untuk memverifikasi tanda tangan digital ECDSA adalah sebagai berikut.

1. Menghitung *hash* dari pesan dengan menggunakan fungsi hash yang sama saat melakukan *sign* dan diubah menjadi bilangan bulat
2. Menghitung invers modulo bukti tanda tangan $s1 = s^{-1} \, mod \, n$
3. Menghitung kembali titik sembarang $R' = (h \times s1) \times G + (r \times s1) \times pubKey$dan ambil koordinat x-nya $r' = R'.x$
4. Validasi tanda tangan dapat diperoleh dengan membandingkan apakah *r* sama dengan *r'*.

D. Fungsi Hash

Fungsi hash adalah sebuah fungsi matematika yang dapat mengubah sebuah angka menjadi sebuah angka lainnya. Pada umumnya, input sebuah fungsi hash memiliki panjang sembarang sedangkan output sebuah fungsi hash memiliki panjang yang tetap. Fungsi hash sering digunakan untuk menyimpan kata sandi dan pengecekan integritas data. Beberapa fungsi hash yang sering digunakan yaitu MD5, SHA, Keccak, RIPEMD, Whirlpool, dan lain-lain.

Fungsi hash Keccak adalah pemenang dari kompetisi SHA-3. Keccak sangat fleksibel karena konstruksi *sponge*-nya. Kontruksi *sponge* pada Keccak berbasis pada fungsi permutasi *random* yang memungkinkan masukan data sebesar apapun dan keluaran data sebesar apapun. Kedua fase tersebut juga sering dikatakan sebagai fase *absorbing* dan *squeezing*.



Gambar 2 Konstrukti *sponge* pada Keccak
https://keccak.team/images/Sponge-150.png

E. Electronic Health

Layanan yang disediakan oleh e-health memungkinkan penyedia layanan kesehatan dan pasien untuk membagi riwayat medis pasien untuk meningkatkan layanan kesehatan yang dapat diberikan. Hal tersebut dapat terjadi karena data yang disimpan secara elektronik lebih efisien daripada data yang tertulis di kertas. Layanan terkait sudah banyak dikembangkan oleh pihak yang terkait. Masalah utama yang masih banyak dihadapi layanan tersebut adalah mengenai privasi pasien. Hal tersebut dikarenakan banyaknya serangan yang dilakukan untuk mencuri data riwayat medis tersebut.

III. RANCANGAN SOLUSI

Setiap pengguna memiliki pasangan kunci privat dan kunci publik. Kunci publik disimpan pada sebuah server yang dapat diakses oleh publik. Fitur tanda tangan digital akan digunakan untuk menjaga aspek *authenticity*, *integrity*, dan *non-repudiation* pada saat pengguna memberikan data atau memberikan *authorization*.

Pada makalah ini, data yang diberikan oleh pengguna diasumsi dalam bentuk json. Sehingga yang disebut pesan adalah keseluruhan objek json tersebut. Objek json tersebut akan diubah menjadi string. Kemudian string tersebut akan di-hash menggunakan algoritma fungsi hash Keccak. Hasil hash dalam bilangan bulat akan dienkripsi menggunakan algoritma kriptografi kunci publik ECDSA dengan kunci privatnya untuk menghasilkan tanda tangan digitalnya. Tanda tangan digital tersebut kemudian akan ditambahkan pada belakang string json tersebut.

Dari sisi penerima, pesan tersebut akan di-hash dengan menggunakan algoritma fungsi hash yang sama pada tahap *sign* yaitu algoritma fungsi hash Keccak. Kemudian tanda tangan digital yang ada di belakang pesan akan didekripsi menggunakan algoritma kriptografi kunci publik ECDSA untuk dengan kunci publiknya. Hasil dari dekripsi tersebut akan dibandingkan dengan hasil hash pesan. Jika hasilnya sama, maka pesan tersebut terverfikasi dari pengirim. Jika tidak, maka pesan tersebut bisa saja bukan dari pengirim atau telah diubah dalam proses pengiriman.

IV. IMPLEMENTASI

A. Implementasi ECDSA

Implementasi algoritma ECDSA pada solusi ini akan memiliki beberapa operasi utama yaitu penjumlahan, penggandaan, dan perkalian titik, serta pembangkitan kunci, *sign*, dan *verify sign*. Untuk mempermudah operasi titik, implementasi akan dipisah menjadi 2 kelas yaitu kelas Point dan kelas Ecc.

Kelas Point terdiri atas dua atribut yaitu x dan y yang keduanya bertipe integer. Kelas Ecc yang dibangun menggunakan standar kurva eliptik *brainpoolP256r1*. Atribut pada kelas Ecc dapat dilihat pada Tabel 1.

Atribut	Tipe	Nilai
a	integer	56698187605326110043627228396178346077120614539475214109386828188763884139993
b	integer	17577232497321838841075697789794520262950426058923084567046852300633325438902
p	integer	76884956397045344220809746629001649093037950200943055203735601445031516197751
n	integer	76884956397045344220809746629001649092737531784414529538755519063063536359079
G	Point	(6324372974956233335529224355031297033477817557105472658709538162362714411478 6, 3821861509375352389312227796403081038758540553977260258155783188748571799795 75)

Operasi matematika untuk titik eliptik dibangun sesuai dengan aturan yang terdapat pada kurva eliptik. Berikut ini penjelasan bagaimana melakukan operasi matematika untuk titik eliptik.

1) Penjumlahan titik ($P + Q = R$)
   i. Menghitung gradien garis dengan rumus
   $$m = \frac{y_p - y_q}{x_p - x_q}$$
   ii. Menghitung koordinat titik hasil
   $$x_r = m^2 - x_p - x_q$$
   $$y_r = m(x_p - x_r) - y_p$$

2) Penggandaan titik ($P + P = R$)
   i. Menghitung gradien garis dengan rumus
   $$m = \frac{3x_p^2 - a}{2y_p}$$
   ii. Menghitung koordinat titik hasil
   $$x_r = m^2 - 2x_p$$
   $$y_r = m(x_p - x_r) - y_p$$

3) Perkalian titik ($kP = R$)
   i. Jika $k > 1$ dan k genap, keluarkan 2 kali dari k dan ulangi
   $$2\left(\frac{k}{2}P\right) = R$$
   ii. Jika $k > 1$ dan k ganjil, keluarkan sebuah k kemudian 2 kali dari k dan ulangi
   $$2\left(\frac{k-1}{2}P\right) + P = R$$
   iii. Jika $k = 1$
   $$P = R$$

1) Pembangkitan kunci

Pada tahap pembangkitan kunci, pengguna dapat membangkitkan pasangan kunci privat dan kunci publik yang akan digunakan untuk menandatangani pesan dan memverifikasi pesan. Kunci privat akan disimpan oleh pengguna dan tidak dapat diakses oleh pengguna lain sedangkan kunci publik akan disimpan di dalam sebuah server dapat diakses oleh pengguna lain untuk kebutuhan verifikasi. Berikut contoh hasil pasangan kunci privat dan kunci publik yang dihasilkan.

Kunci privat	550403742024691860812806260647358097126804793938470123085657550260378866694987
Kunci publik	(27660976180963848645498449546691105422166274491700746245315476675725475534743, 863622901856077845789856680342740498484880758974880622230019917036390690795 6)

2) *Sign*

Pada tahap *sign*, pengguna akan menandatangani pesan yang akan dikirim. Kunci privat pengguna akan digunakan untuk menghasilkan tanda tangan digital atas pesan tersebut. Proses untuk menghasilkan tanda tangan digital mengikuti algoritma ECDSA pada umumnya yang sudah dijelaskan pada Bab II Dasar Teori. Tanda tangan digital yang dihasilkan akan diubah menjadi string base64 dan ditambahkan di belakang pesan. Berikut hasil *sign* yang diperoleh pada pesan contoh.

Pesan asli	{"Content":"Message for ECDSA Signing"}
Pesan dengan tanda tangan digital	{"Content":"Message for ECDSA Signing"} --- Begin of digital signature ---- KDYyMzEwNjY3NDI1NjkwMDU5ODY2MjM0ODg4MzQ4MTUyODU1MTMzMjI1MTYzODEzODA1OTk0ODA4ODIzMjcyNzEzOTM4MTE5NzA5NTY3LDkzMjI5NzczNDM0NjExMDE0MTE0MjQ3NzM3MjQ1NjE0OTIwNTg2Mjk3NjE4NTUwNjUzMDQzNjE1NTM4MjU3NjI0NjQ3MDMzNzI0NDMp --- End of digital signature ----

3) *Sign verify*

Pada tahap *sign verify*, penerima pesan dapat melakukan verifikasi pesan dengan menggunakan kunci publik pengirim yang dapat diakses oleh penerima pesan. Kunci publik tersebut akan digunakan untuk mendekripsi tanda tangan digital. Hasil tersebut kemudian akan dibandingkan dengan hash dari pesan untuk mendapatkan hasil verifikasi. Berikut ini adalah contoh hasil *sign verify* yang dilakukan terhadap pesan yang sudah diberikan tanda tangan digital pada tahap *sign*.

Hasil *sign verify*	*true*

B. Implementasi Layanan *Electronic Health*

Implementasi layanan *e-health* yang dilakukan akan terdiri atas beberapa operasi utama yaitu penyetujuan penyimpanan informasi pengguna dan pemberian *authorization* untuk mengakses informasi pengguna. Kedua operasi tersebut diimplementasi dalam bentuk layanan dan akan diberikan tanda tangan digital untuk setiap penggunaan layanan tersebut. Penggunaan tanda tangan digital di sini digunakan untuk memastikan aspek kriptografi mengenai *authentication, data integrity*, dan *non-repudiation* dapat terpenuhi.

Berikut ini contoh penggunaan tanda tangan digital pada data yang dikirim ketika memberikan *authorization* untuk mengakses informasi pengguna tersebut.

Pesan asli	{"userEmail":"patient@gmail.com","viewerEmail":"doctor@gmail.com"}
Pesan dengan tanda tangan digital	{"userEmail":"patient@gmail.com","viewerEmail":"doctor@gmail.com"} --- Begin of digital signature ---- KDU3ODYwODI4NDI2Njk0MjcxMzU3MzIxNTQ3MzI0MzA0Mjg5NDM4OTg4MDExODI0OTc2OTI0MzE1OTA5NDUwOTA3NzU3NzQ3NDIxNzYwLDc1NjMyNTcyOTEyNDQyMjYwNjQ0MjA3NTg2OTYzMjU1MTk4OTQyODU5NzYzMjY5MjEwNjM3OTcxNzI0NjM0MTU4NzY3ODQ2MjczODA3KQ --- End of digital signature ----

Hasil *sign verify*	*true*

Berikut ini contoh penggunaan tanda tangan digital pada data yang dikirim ketika menyetujui untuk melakukan penyimpanan informasi pengguna.

Pesan asli (sebelum menjadi string)	{
	"userEmail":"patient@gmail.com",
	"bloodPressure":"180/120",
	"sugarLevel":"120",
	"heartRate":"180",
	"age":"50",
	"note":"Sugar level checked after fasting"
	}

Pesan dengan tanda tangan digital	{"userEmail":"patient@gmail.com","bloodPressure":"180/120","sugarLevel":"120","heartRate":"180","age":"50","note":"Sugar level checked after fasting"} --- Begin of digital signature ---- KDMwNTY2OTc3NTY5NzAwNTg4Mjc2MDkxNTM0MDk2MzIzMjAxMjAyMjY5Mjg2OTc3NzU0ODc3NTIzNjE3NzY2MzY3MzIxMDcwNzE5OTk3LDM4MjUyODc5NzA5NzE1MTU5MTMzNDIyMTg3NzIxMTk1OTM3NzE0MjQ5NTI3NDM0MDY3NTgwNTg5MjQ5MzczODIyMjE3MzM0ODcwMjgzKQ --- End of digital signature ----

Hasil *sign verify*	*true*

V. EKSPERIMEN DAN ANALISIS

Dari ketiga aspek kriptografi yang dipenuhi oleh ECDSA, hanya *authentication* dan *data integrity* yang dapat diuji melalui eksperimen. Berikut adalah pesan atau data yang akan diberikan tanda tangan digital, pesan yang sudah diberikan tanda tangan digital, dan pasangan kunci privat dan publik pengguna.

Kunci privat	55040374202469186081280626064735809712680479393847012308565755026037886694987
Kunci publik	(27660976180963848645498449546691105422166274491700746245315476675725475534743, 86362290185607784578985668034274049848488075897488062223001991703639069079 56)

Pesan asli (sebelum menjadi string)	{
	"userEmail":"patient@gmail.com",
	"bloodPressure":"160/100",
	"sugarLevel":"110",
	"heartRate":"160",
	"age":"55",
	"tbc":"true"
	}

Pesan dengan tanda tangan digital	{"userEmail":"patient@gmail.com","bloodPressure":"160/100","sugarLevel":"110","heartRate":"160","age":"55","tbc":"true"} --- Begin of digital signature ---- KDI5NTQ5ODM1MDIyNDYzNDQ3MjgyMTg5MzQyOTIwNDgwODc0ODkyOTI2OTY2OTk4NzQxNjM2NTM4NDkyMjU4MjQ5NzY1MjgyNTE4NjA4LDUzNzgxMTU3NTcyNjMwMjAwNDI1OTMwMDg1MDA4MDg5NTAwMDIzNTcyNzA0MzE2NjI4MDQyMTUwNTkxOTkzOTgxMDY3ODgzNzY3OTMxxKQ --- End of digital signature ----

Pengujian akan dilakukan dengan beberapa cara yaitu membuat tanda tangan digital menggunakan kunci privat yang berbeda, memverifikasi pesan menggunakan kunci publik yang salah, mengubah isi pesan, dan mengubah tanda tangan digital.

1) Membuat tanda tangan digital menggunakan kunci privat yang berbeda

Kunci privat berbeda	**192303312092460724465818307153872409459252922692148542040782635424961501721 23**
Pesan dengan tanda tangan digital	{"userEmail":"patient@gmail.com","bloodPressure":"160/100","sugarLevel":"110","heartRate":"160","age":"55","tbc":"true"} --- Begin of digital signature ---- **KDM1NDA2MjM3NTYxNzc1MTYwMDA4MjEwNzU4MjA1MDE0NjQ4NzY3Mjk2NjK2NjQ2NTEzMzY4NjEyMDcyNjMyOTQyMDU1NTA5MzUyOTg0LDUyMDY4MjA3MjU4MDY4MzAwMTUyOTA4MjYxODQxMDMyMjUxMTY5NDI4OTY5OTUxNTU3NzU5NTc5ODMxNTkwMDkwMzUzNTM2NzA0MjAp** --- End of digital signature ----

Hasil *sign verify*	*false*

2) Memverifikasi pesan menggunakan kunci publik yang salah

Kunci publik salah	(27660976180963848645498449546691105422166274491700746245315476675725475534743, 86362290185607784578985668034274049848488075897488062223001991703639069079 5**7**)

Hasil *sign verify*	*false*

3) Mengubah isi pesan

Pesan dengan tanda tangan digital	{"userEmail":"patient@gmail.com","bloodPressure":"160/100","sugarLevel":"110","heartRate":"160","age":"55","tbc":"true"**,"hiv":"true"**} --- Begin of digital signature ---- KDI5NTQ5ODM1MDIyNDYzNDQ3MjgyMTg5MzQyOTIwNDgwODc0ODkyOTI2OTY2OTk4NzQxNjM2NTM4NDkyMjU4MjQ5NzY1MjgyNTE4NjA4LDUzNzgxMTU3NTcyNjMwMjAwNDI1OTMwMDg1MDA4MDg5NTAwMDIzNTcyNzA0MzE2NjI4MDQyMTUwNTkxOTkzOTgxMDY3ODgzNzY3OTMxxKQ --- End of digital signature ----

Hasil *sign verify*	*false*

4) Mengubah tanda tangan digital

Pesan dengan tanda tangan digital	{"userEmail":"patient@gmail.com","bloodPressure":"160/100","sugarLevel":"110","heartRate":"160","age":"55","tbc":"true" --- Begin of digital signature ---- KDI5NTQ5ODM1MDIyNDYzNDQ3MjgyMTg5MzQyOTIwNDgwODc0ODkyOTI2OTY2OTk4NzQxNjM2NTM4NDkyMjU4MjQ5NzY1MjgyNTE4NjA4LDUzNzgxMTU3NTcyNjMwMjAwNDI1OTMwMDg1MDA4MDg5NTAwMDIzNTcyNzA0MzE2NjI4MDQyMTUwNTkxOTkzOTgxMDY3ODgzNzY3OTMxxK**R** --- End of digital signature ----

Hasil *sign verify*	*false*

Berdasarkan eksperimen yang telah dilakukan, implementasi ECDSA pada layanan e-health terimplementasi dengan baik. Hal tersebut dibuktikan oleh eksperimen 4 buah *test case* yang

merepresentasi aksi yang dapat dilakukan oleh penyerang dan sebuah *test case* yang sesuai harapan pada bab IV bagian B.

VI. KESIMPULAN DAN SARAN

Kesimpulannya adalah tanda tangan digital dengan menggunakan algoritma kriptografi kunci publik memiliki peran yang sangat penting pada era digital ini. Penggunaan tanda tangan digital dapat memastikan aspek *authenticity*, *integrity*, dan *non-repudiation* sebuah pesan atau dokumen yang ditandatanganinya. Salah satu penerapan tanda tangan digital adalah implementasi ECDSA pada layanan e-health. Implementasi ini terbukti dapat melindungi integritas dan akses pada data pengguna.

Untuk selanjutnya, selain menggunakan tanda tangan digital, disarankan juga untuk menggunakan enkripsi pada data yang disimpan. Hal tersebut perlu dilakukan untuk memenuhi seluruh aspek kriptografi. Walaupun dengan menggunakan tanda tangan digital untuk memberikan akses kepada pihak tertentu, tetapi penyerang bisa saja menyerang server dan mendapatkan semua informasi pengguna.

REFERENSI

[1] "What is Digital Signature- How it works, Benefits, Objectives, Concept". https://www.emptrust.com/blog/benefits-of-using-digital-signatures. Diakses pada tanggal 21 Desember 2020

[2] "Cryptography Digital signatures". https://www.tutorialspoint.com/cryptography/cryptography_digital_signatures.htm. Diakses pada tanggal 21 Desember 2020

[3] Munir, R. 2020. "Kriptografi Kunci Publik 2020". Slide terdapat pada link: http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2020-2021/Kriptografi-Kunci-Publik-2020.pdf

[4] "What is Public-key Cryptography?". https://www.globalsign.com/en/ssl-information-center/what-is-public-key-cryptography. Diakses pada tanggal 21 Desember 2020

[5] "Keccak". https://keccak.team/keccak.html. Diakses pada tanggal 21 Desember 2020

[6] Mishall Al-Zubaidie, Zhongwei Zhang, Ji Zhang. (2019). "Efficient and Secure ECDSA Algorithm and its Application". Australia: University of Southern Queensland.

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.

Medan, 21 Desember 2020

Christzen Leonardy - 13517125