e-ISBN 978-623-93343-4-5
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 15 September 2021

Keamanan Data Absensi Berbasis Android dengan Menggunakan Algoritma RSA Pada PT. Arkonin

Farhana Nabila[1], Henki Bayu Seta, S.Kom, MTI[2], Noor Falih, S.Kom., M.T[3]
Informatika / Fakultas Ilmu Komputer
Universitas Pembangunan Nasional Veteran Jakarta
Jl. Rs. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia
farhananabila@upnvj.ac.id[1], henkiseta@upnvj.ac.id[2], falih@upnvj.ac.id[3]

Abstrak. Kehadiran merupakan penilaian kinerja yang penting. Untuk mendorong bantuan proyek lapangan, diperlukan bantuan yang dapat mendukung pendataan proyek lapangan. Saat ini sudah ada sistem absensi *fingerprint*, namun ada beberapa kendala absensi yaitu karyawan yang mengerjakan proyek di lapangan harus mengirimkan data absensi ke perusahaan melalui email. Tujuan dari penelitian ini adalah untuk melindungi proses pengiriman data absensi karyawan di proyek lapangan ke perusahaan dengan menggunakan teknologi absensi *fingerprint* di android menggunakan metode algoritma RSA. Dalam penelitian ini pengamanan data absensi adalah pada proses pengiriman data dalam bentuk enkripsi pada aplikasi absensi, data secara otomatis terkirim ke *database* dalam bentuk data enkripsi dan dekripsi. Pengkodean sistem dilakukan dengan menggunakan JAVA, PHP dan MYSQL. Hasil dari penelitian ini adalah aplikasi pengamanan data absensi berbasis android, untuk mempermudah absensi yang bekerja di proyek lapangan sehingga data aman dan cepat hingga sampai kepada pihak yang terkait. Dari uraian di atas diharapkan dapat diperoleh data kehadiran yang asli.

Kata kunci: Absensi berbasis android, Algoritma RSA, Enkripsi dan Dekripsi.

1 Pendahuluan

Di era perkembangan, khususnya di bidang ilmu pengetahuan dan teknologi informasi, teknologi informasi semakin maju pesat, sehingga masyarakat dan dunia usaha perlu menuntut pengetahuannya untuk meningkatkan prestasi kerjanya. Sistem pelayanan waktu merupakan salah satu faktor yang dapat meningkatkan kinerja dan kualitas karyawan dengan cara meningkatkan kualitas dan efisiensi karyawan perusahaan. Namun, di era modern ini, ponsel dapat digunakan untuk sistem pengaturan waktu perusahaan. Sebelum memasuki suatu proyek, seorang karyawan harus hadir. Oleh karena itu, langkah-langkah keamanan yang tepat harus diambil terhadap ketidakhadiran karyawan. Untuk itu keamanan sistem manajemen kehadiran seluler karyawan harus aman. Keamanan komputer merupakan masalah penting di era teknologi saat ini, dan ada banyak masalah keamanan yang sering diabaikan oleh perusahaan. Dalam hal keamanan, perusahaan di bidang jasa menempatkannya di urutan kedua, jika bukan terakhir, dalam daftar hal-hal yang mereka anggap penting. Beberapa perusahaan hanya memikirkan layanan yang mereka tawarkan untuk berpartisipasi dalam layanan, tetapi mereka tidak terlalu memikirkan dalam hal keamanan.

Dengan menerapkan metode RSA pada pengembangan dari aplikasi absensi mobile diharapkan dapat diterapkan sebagai *alternative* absensi dilakukan di sebuah proyek lapangan. Dengan mempertimbangkan masalah ini, penulis berencana untuk menerapkan *fingerprint* atau sidik jari, lokasi, dan waktu untuk membuat aplikasi absensi di perangkat seluler karyawan guna meminimalkan gangguan kehadiran karyawan. Menurut peneliti aplikasi ini akan memudahkan karyawan dalam melaksanakan absensi di proyek lapangan. Sehingga, karyawan tidak perlu lagi mencari tempat atau lokasi untuk melakukan *fingerprint*/sidik jari di laptop. Karyawan dapat melakukan pekerjaannya dengan baik dan tidak mengganggu efesiensi dalam bekerja.

Proses yang melibatkan karyawan menggunakan *smartphone* untuk memverifikasi kehadiran dalam bentuk sidik jari enkripsi dan hasilnya dikirim langsung ke *database* perusahaan. Sebelum memasuki proyek di tempat, karyawan harus terlebih dahulu memeriksa keberadaan ponsel cerdas mereka. Jika karyawan telah memverifikasi kehadiran dan memenuhi persyaratan, data akan muncul di *database* perusahaan sebagai dekripsi (teks yang jelas) dan data akan dikelola oleh admin perusahaan untuk menghindari penipuan data. Proses mobilisasi sistem yang sedang dilakukan pada sistem ini akan lebih efisien dan praktis.

2 Landasan Teori

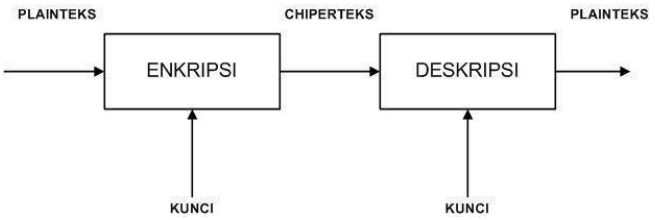
2.1 Kriptografi

Kriptografi berasal dari bahasa yunani *cryptos* yang berarti rahasia atau tersembunyi, dan *graphene* yang berarti tulisan [1]. Kriptografi oleh karena itu dapat disebut tulisan rahasia. Untuk Menezes dan Ochott,Vantone enkripsi adalah metode penelitian matematis yang terkait dengan keamanan data seperti kerahasiaan, otentikasi entitas, otentikasi informasi, dan otentikasi integritas informasi. Kriptografi tidak hanya menjamin keamanan data, tetapi juga menyediakan seperangkat teknologi.

Prinsip dasar kriptografi adalah:
1. Kerahasiaan adalah layanan dimana isi pesan yang dikirim selalu dirahasiakan dan tidak diketahui oleh pihak lain (kecuali pengirim, penerima, dll) atau pihak yang berwenang).
2. Integritas adalah layanan yang membantu Anda mengidentifikasi atau mendeteksi aktivitas ilegal.
3. Otentikasi adalah layanan yang terkait dengan identitas.
4. *Non*-repudiation (anti-repudiation) adalah pelayanan yang dapat mencegah pihak-pihak menolak upaya pekerjaan [2].

Kriptografi klasik merupakan metode enkripsi dimana kunci dekripsi yang digunakan untuk enkripsi simetris sama dengan kunci enkripsi. Untuk kriptografi kunci publik, kriptografi asimetris diperlukan jika kunci dekripsi tidak sama dengan kunci enkripsi [3]. Kriptografi asimetris menggunakan begitu banyak angka sehingga enkripsi, dekripsi, dan pembuatan kunci dengan kriptografi asimetris membutuhkan lebih banyak komputasi dari pada cipher simetris.

Istilah-istilah berikut digunakan dalam kriptografi:
1. Teks biasa (M) adalah pesan yang Anda kirim (termasuk informasi asli).
2. *Encrypted Text* (C) adalah pesan enkripsi (*encrypted*) yang merupakan hasil enkripsi.
3. Enkripsi (E) adalah proses pengubahan teks biasa menjadi teks sandi.
4. Dekripsi (D) adalah kebalikan dari enkripsi, yang mengubah *ciphertext* menjadi teks yang jelas sehingga terlihat seperti informasi yang benar/asli yang dibagikan.
5. Kunci adalah nomor tersembunyi yang digunakan selama enkripsi dan dekripsi.



Gambar 1. Proses Enkripsi Dan Dekripsi Sederhana.

Sederhananya, enkripsi terdiri dari dua proses utama, enkripsi dan dekripsi. Enkripsi adalah proses beralih dari kode yang dapat dipahami ke kode yang sulit dipahami (tidak dapat dibaca). Dekripsi adalah proses memulihkan data enkripsi dalam bentuk aslinya menggunakan algoritma yang sama. Gambar 1 menunjukkan proses enkripsi dan dekripsi sederhana.

2.2 Algoritma RSA

Kriptografi RSA adalah algoritma kriptografi kunci publik (asimetris). Ron Rivest, Adi Shamir dan Len Adlema pertama kali bertemu pada tahun 1977. Nama RSA sendiri berasal dari ketiga pendiri tersebut. Sebagai algoritma kunci publik, RSA memiliki dua kunci, kunci publik dan kunci pribadi. Proses enkripsi dan dekripsi RSA

e-ISBN 978-623-93343-4-5
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 15 September 2021

didasarkan pada konsep bilangan prima dan operasi modulus [4]. Kunci enkripsi dan kunci dekripsi adalah bilangan bulat. Kunci enkripsi tidak disembunyikan tetapi kunci publik (maka kunci publik), tetapi kunci dekripsi adalah rahasia (kunci pribadi).

Teks biasa yang dienkripsi menggunakan enkripsi RSA adalah angka, sedangkan pesan yang dikirim biasanya berupa teks atau kata-kata. Jadi kita membutuhkan kode universal untuk mengubah pesan teks menjadi teks biasa digital. ASCII (*American Standard Code for Information Interchange*) adalah standar internasional untuk kode karakter dan simbol seperti heksadesimal dan Unicode, tetapi ASCII lebih umum, seperti 1234 untuk "|" karakter. Komputer dan metode komunikasi lainnya masih menggunakan ASCII untuk merepresentasikan teks [5].

Tabel 1. ASCII.

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	`
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(	72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29	)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-_	63	3F	?	95	5F	_	127	7F	DEL

2.2.1. Algoritma Pembangkitan Kunci

Proses enkripsi menggunakan algoritma RSA dapat disamakan dengan proses pengiriman data kehadiran untuk pengguna A (karyawan) dan pengguna B (admin). Sebaliknya, jika pesan di dekripsi menggunakan algoritma RSA, hal yang sama berlaku jika pengguna B menerima *plaintext* dari pengguna A. semua hasil *chipertext* diambil untuk enkripsi. Data diubah menjadi teks biasa dan dimasukkan langsung ke *database* perusahaan. User B kemudian melakukaan pengecekan data absensi dan menyatakan kehadiran karyawan tersebut. Untuk menghasilkan kunci enkripsi dan dekripsi di RSA :

1. Pilih dua bilangan prima sembarang p dan q
2. Hitung $n = p*q$, dengan $p \neq q$
3. Hitung $m = (p-1)*(q-1)$
4. Pembangkitan kunci publik *e* yang relative prima terhadap *m*.
   Bangkitkan kunci privat $d * e \equiv 1 \pmod{m}$ ekivalen dengan $d * e \equiv 1 + \text{mod}\ m$, sehingga *d* dapat dihitung dengan :

$$\underline{d =} \quad \underline{1} \; \underline{+} \; \underline{m} \; \underline{o} \; \underline{d} \; \underline{\frac{m}{e}}$$

Hasil dari algoritma di atas adalah :

a. Kunci Publik (n,e)
b. Kunci Privat (d,n)

2.2.2. Rumus Kunci Publik (Enkripsi)

Kunci publik adalah kunci yang didapat dari hasil pesan yang dikodekan menjadi kode ASCII dengan menggunakan rumus :

$$Ci = Pi^e \bmod n \quad (1)$$

Keterangan :
C : *Cipherteks*
P : *Plainteks*
e : Kunci Enkripsi

2.2.3. Rumus Kunci Privat (Dekripsi)

Setelah mendapatkan nilai dari kunci publik/kunci enkripsi, maka dapat ditentukan nilai kunci privat/kunci dekripsi. Pesan enkripsi dipecah menjadi array dengan batasan menggunakan rumus:

$$pi = Ci^d \bmod n \quad (2)$$

Keterangan :
C : *Cipherteks*
P : *Plainteks*
d : Kunci Dekripsi [6]

2.3 PHP

PHP dapat dipahami sebagai *preprocessor hypertext*. Ini adalah bahasa yang hanya dapat dijalankan di server yang dapat menampilkan hasilnya ke klien. Penerjemah PHP yang mengeksekusi kode PHP di sisi server disebut sisi server, sebagai lawan dari mesin virtual JAVA yang mengeksekusi program di sisi klien [7].

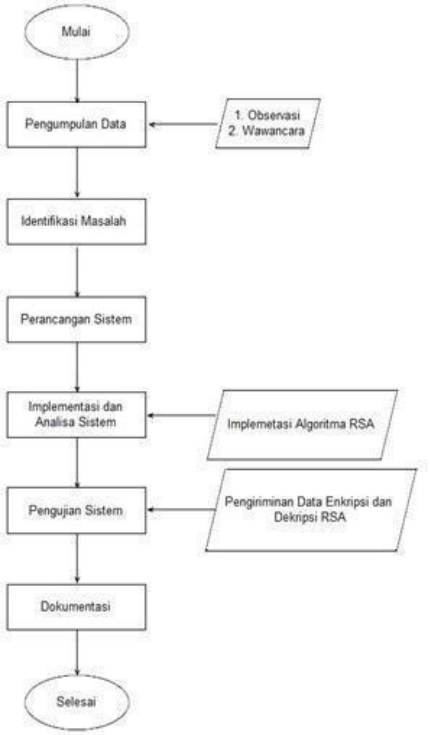
2.4 MySQL

MySQL disebut SQL dan merupakan singkatan dari *Structured Query Language*. SQL adalah bahasa terstruktur yang digunakan khusus untuk menganalisis *database*. SQL pertama kali didefinisikan oleh American National Standards Institute (ANSI) pada tahun 1986. MySQL adalah sistem manajemen *database open source*. MySQL adalah sistem manajemen basis data relasional [8]. Artinya informasi yang dikelola oleh *database* dapat ditempatkan pada beberapa tabel yang terpisah, sehingga informasi tersebut dapat diproses lebih cepat. MySQL dapat digunakan untuk mengelola *database* dengan jumlah data yang kecil atau bahkan sangat besar.

2.5 JAVA

JAVA adalah bahasa pemrograman yang sangat populer yang banyak digunakan dalam pengembangan semua jenis fitur perangkat lunak aplikasi dan aplikasi web. Bahasa ini awalnya dibuat oleh James Gosling ketika dia masih bekerja di Sun Microsystems, yang sekarang menjadi bagian dari Oracle, dan dirilis pada tahun 1995. Bahasa ini mengadopsi banyak sintaks yang ditemukan di C dan C. ++ tetapi dengan sintaks model objek yang lebih sederhana dan mendukung proses yang mendasarinya minimum. Aplikasi berbasis JAVA biasanya dikompilasi dalam *pcode* (*bytecode*) dan dapat dijalankan di berbagai Java Virtual Machines (JVM) [9].

3 Metodologi Penelitian



Gambar 2. Kerangka Pikir.

3.1 Pengumpulan Data

Beberapa metode yang digunakan dalam penelitian ini untuk mengumpulkan data, antara lain :

a. Observasi Metode observasi dilakukan dengan cara melakukan pengamatan terhadap data absensi pegawai pada bagian proyek lapangan perusahaan arkonin.

b. Wawancara Metode wawancara dilakukan dengan mewawancarai kepala divisi HRD dan staff IT untuk mengetahui informasi yang lebih mendalam mengenai permasalahan yang berhubungan dengan data absensi dan keamanan data.

3.2 Identifikasi Masalah

Mengidentifikasi masalah-masalah yang terkait di dalam keamanan data absensi proyek lapangan. Data yang dikumpulkan berupa teori yang berhubungan dengan pendataan absensi pegawai di perusahaan arkonin.

3.3 Perancangan Sistem

Perancangan sistem yang akan dibuat proses perhitungan enkripsi dan dekripsi yang dilakukan secara manual menggunakan algoritma RSA agar data yang di enkripsi tidak dapat dirubah oleh pegawai atau karyawan.

3.4 Implementasi dan Analisa Sistem

Pada tahap ini data yang di dapat dari pengumpulan data yang dilakukan di metode perancangan sistem yaitu dilakukan uji coba dengan menggunakan perhitungan di dalam program aplikasi.

3.5 Pengujian Sistem

Pengujian dilakukan dengan menggunakan *Black Box Testing* berfokus terhadap keamanan data dan pengujian proses waktu enkripsi menjadi dekripsi. Tujuannya adalah meminimalisir terjadinya kecurangan pada data saat melakukan absensi di sebuah proyek lapangan sehingga data aman dan cepat hingga sampai kepada pihak yang terkait sistem yang digunakan nantinya akan membantu user saat melakukan absensi

4 Hasil dan Pembahasan

4.1 Analisa Masalah

Untuk menyelesaikan masalah absensi karyawan adalah absensi mobile dengan sistem *fingerprint*. *Fingerprint* akan menghasilkan data yang akan menjadi data enkripsi berdasarkan status, jam, dan lokasi yang akan tercantum di dalam sistem *fingerprint* mobile. Selanjutnya data tersebut akan ter-otomatisasi menghasilkan proses data enkripsi menjadi data dekripsi sebelum masuk ke dalam *database* perusahaan.

4.2 Perancangan Sistem

Perancangan sistem untuk pengguna sistem berdasarkan perancangan dan identifikasi yang dilakukan. Perancangan ini di butuhkannya informasi secara cepat, tepat, akurat, dan keamanan terjamin kepada pihak yang membutuhkan agar data tidak dapat diubah. Algoritma yang digunakan adalah algoritma RSA, yaitu algoritma asimetris yang menggunakan dua kunci yang berbeda untuk enkripsi dan dekripsi. Pertama, *ciphertext* dikodekan menggunakan kode ASCII, dan kemudian dikodekan karakter. Pesan enkripsi dibagi menjadi tabel dengan batasan yang akan menghasilkan dekripsi.

4.2.1. Perhitungan Manual Menggunakan Algoritma RSA

Pembuatan kunci enkripsi dan dekripsi RSA :

1. Mencari bilangan random 1000 – 2000
2. Kemudian mencari bilangan prima dari bilangan random
3. Pilih dua buah bilangan prima, *p* dan *q*
   p = 1847
   q = 1423
4. Mencari nilai *n*
   n = *p* * *q*
   n = 1847 * 1423
   n = 2628281
5. Mencari nilai *m*
   m = (*p* -1) * (*q* – 1)
   m = (1847 – 1) * (1423 -1)
   m = (1846) * (1422)
   m = 2625012
6. Mencari nilai *e* yang relative prima terhadap *m*, Rumus algoritma *Euclidean* :

$$r_0 = q_1 r_1 + r_2, dimana, 0 < r_2 < r_1$$
$$r_1 = q_2 r_2 + r_3, dimana, 0 < r_3 < r_2 r_{n=0} \quad (3)$$

e-ISBN 978-623-93343-4-5
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 15 September 2021

Percobaan 1 : nilai e = 2, m = 2625012 gcd ($e.m$) = 1
gcd (2, 2625012) = 1
r_0 = 2, r_1 = 2625012
$r_0 = q_1.r_1+r_2$
2 = 0.2625012+2
Keterangan : nilai awal q = 0,

$$r_2 = r_0 - (q_1.r_1)$$
$$r_2 = 2 - (0.2625012)$$
$$r_2 = 2$$

Karena nilai r belom 0 maka dilanjutkan :

$r_1 = q_2 . r_2 + r_3$
2625012 = 1312506 . 2 + 0
Keterangan : $q_2 = r_1 : r_2$

$$q_2 = 2625012 : 2$$
$$q_2 = 1312506$$

Dari perhitungan di atas, kita dapat melihat bahwa nilai r sebelum 0 adalah 2. Ini berarti e yang kita coba bukanlah gcd(e,m)=1. Itu artinya kita harus mencoba nilai lain, misalnya kita mencoba e = 5 sebagai berikut : percobaan 2 : nilai e = 2, m = 2625012
gcd ($e.m$) = 1
gcd (2, 2625012) = 1
r_0 = 5, r_1 = 2625012
$r_0 = q_1 . r_1 + r_2$
5 = 0 . 2625012 + 5
Keterangan : nilai awal q = 0,

$$r_2 = r_0 – (q_1 . r_1)$$
$$r_2 = 5 – (0 . 2625012)$$
$$r_2 = 5$$

Karena nilai r belom 0 maka dilanjutkan : $r_1 = q_2 . r_2 + r_3$
2625012 = 5250024 . 5 + 0
Keterangan : $q_2 = r_1 : r_2$

$$q_2 = 2625012 : 5$$
$$q_2 = 5250024$$
$$r_3 = r_1 – (q_2 . r_2)$$
$$r_3 = 2625012 – (5250024 . 5)$$
$$r_3 = 2625012 – 2625012$$
$$r_3 = 0$$

Dari hasil di atas di dapatkanlah nilai e, yaitu 5.

7. Mencari nilai d dimana d = (e * d) mod m
= 1
e = 5
m = 2625012

$$e * d \bmod m = 1$$

$$5 * 1050005 \bmod 2625012 = 1$$

Keterangan : nilai d adalah nilai yang kita tebak dan coba cocokkan karena 5 * 1050005 = 5250025 mod 2625012 maka hasilnya 1 maka 1050005 adalah angka yang tepat untuk nilai d.

8. Dengan di dapat nilai d yaitu 1050005, maka pasangan kunci adalah : Kunci publik (e,n) yaitu (5, 2628281)
Kunci privat (d,n) yaitu (1050005, 2628281)
n = 2628281
e (Kunci Enkripsi) = 5
d (Kunci Dekripsi) = 1050005

4.2.2. Proses Enkripsi

Untuk proses enkripsi yang di lakukan android untuk mengirim data ke *database* perusahaan hanya mengamankan data jam masuk, jam keluar, status keluar, status masuk, dan lokasi. Dengan bit input yang berbeda menghasilkan *chipertext* yang berbeda dengan jumlah bit yang sama, tetapi *chipertext* yang berbeda. Karena nilai generator kunci adalah p dan q diambil secara acak. Contoh yang dilakukan pada jam masuk :

1. *Plaintext* : 08:03:04
   Kode ASCII dari *plaintext* jam masuk

Tabel 2. Plaintext Jam Masuk 08:03:04

plaintext	0	8	:	0	3	:	0	4
ASCII	48	56	58	48	51	58	48	52

$0 = 48^5 \bmod 2628281 = 2488992$
$8 = 56^5 \bmod 2628281 = 1421047$
$: = 58^5 \bmod 2628281 = 1914799$
$0 = 48^5 \bmod 2628281 = 2488992$
$3 = 51^5 \bmod 2628281 = 720440$
$: = 58^5 \bmod 2628281 = 1914799$
$0 = 48^5 \bmod 2628281 = 2488992$
$4 = 52^5 \bmod 2628281 = 1731568$

Jadilah *chipertext* adalah :
2488992.1421047.1914799.2488992.720440.1914799.
2488992.1731568

4.2.3. Proses Dekripsi

Untuk proses dekripsi hanya dapat di lihat oleh admin di dalam *database*, data yang ter dekripsi secara otomatis melalui proses enkripsi yaitu data jam masuk, jam keluar, status masuk, status keluar, dan lokasi. Untuk contoh proses dekripsi disini penulis mengambil contoh dari data absensi jam masuk yang sudah otomatis ter-enkripsi.

1. *Chipertext* : 2488992.1421047.1914799.2488992.720440.1914799.2488992.1731568
   Kode ASCII dari *chipertext* jam masuk dengan *plaintext*
   08:03:04 $0 = 2488992^{1050005} \bmod 2628281 = 48 \rightarrow 0$
   $8 = 1421047^{1050005} \bmod 2628281 = 56 \rightarrow 8$
   $: = 1914799^{1050005} \bmod 2628281 = 58 \rightarrow :$
   $0 = 2488992^{1050005} \bmod 2628281 = 48 \rightarrow 0$
   $3 = 720440^{1050005} \bmod 2628281 = 51 \rightarrow 3$
   $: = 1914799^{1050005} \bmod 2628281 = 58 \rightarrow :$
   $0 = 2488992^{1050005} \bmod 2628281 = 48 \rightarrow 0$
   $4 = 1731568^{1050005} \bmod 2628281 = 52 \rightarrow 4$

4.3 Implementasi dan Analisa Sistem

Uji coba proses absensi menggunakan android, data akan langsung terkirim ke dalam *database*, dan perhitungan yang dilakukan dengan menggunakan program, pengujian ini dilakukan apakah proses enkripsi dan dekripsi terjamin keamanan datanya hingga sampai ke pihak yang terkait. Proses ini juga dilakukan secara manual untuk menguji apakah pehitungan dengan menggunakan aplikasi dan manual sama keakuratannya. Jumlah bit yang sama akan menghasilkan chipertext yang berbeda, tetapi input bit yang berbeda akan menghasilkan chipertext yang berbeda.

4.3.1. Perhitungan Enkripsi dengan Menggunakan Program Aplikasi

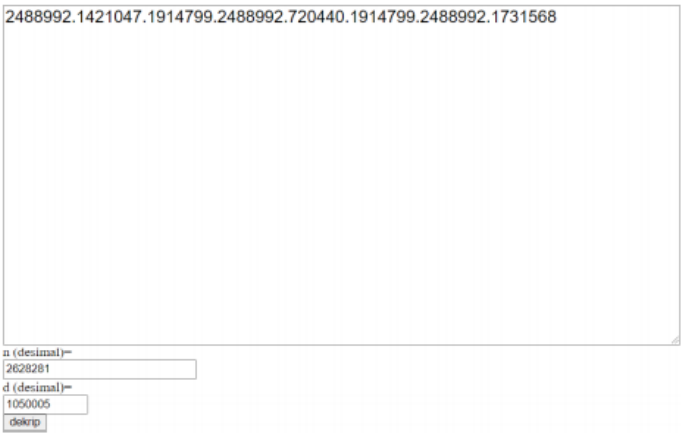
Dapat dilihat pada gambar 3 proses yang dilakukan dengan menggunakan aplikasi menggunakan nilai n =

2628281 dan e = 5. Hasil nilai tersebut dilakukan dengan menggunakan perhitungan secara manual menggambil nilai secara acak atau random.

Gambar 3. Perhitungan Enkripsi Menggunakan Program.

4.3.2. Perhitungan Dekripsi dengan Menggunakan Program Aplikasi

Pengujian proses dekripsi dilakukan setelah mendapatkan hasil pada proses enkripsi. Jika sudah mendapatkan hasil dari proses enkripsi maka dilakukanlah uji coba perhitungan menggunakan aplikasi dengan proses dekripsi, dengan nilai n = 2628281 dan d = 1050005. Nilai tersebut di dapat dari proses perhitungan manual.



Gambar 4. Perhitungan Dekripsi Menggunakan Program.

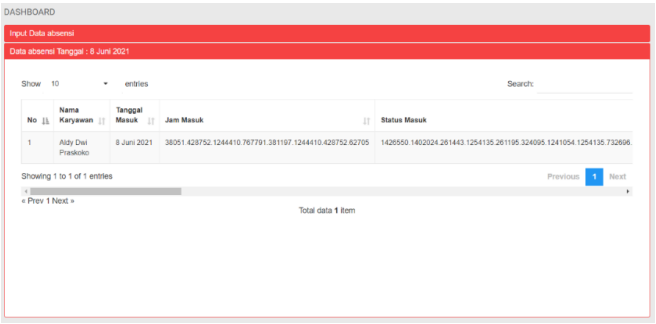
4.3.3. Enkripsi Pada Aplikasi Berbasis Android

Untuk meng-menenkripsi data yang tersimpan didalam aplikasi absensi berbasis android. Hasilnya ketika melakukan *fingerprint*/sidik jari data tersebut sudah ter-enkripsi di dalam aplikasi absensi, maka data yang sudah ter-enkripsi tidak akan bisa diubah oleh karyawan dan data yang terbaca berupa *ciphertext*. Untuk data yang ter-enkripsi secara otomatis di dalam aplikasi absensi berbasis android yaitu jam masuk, status masuk, jam keluar, status keluar, dan lokasi.

Gambar 5. Enkripsi Berbasis Android.

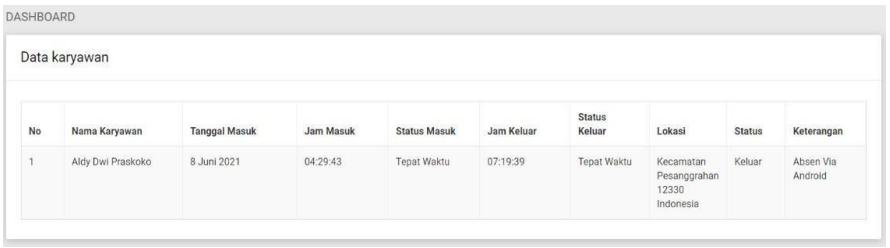
4.3.4. Enkripsi Dan Dekripsi Pada *Database*

Setelah proses enkripsi di dalam absensi berbasis android telah selesai, data tersebut akan ter-otomatisasi terkirim ke dalam *database* perusahaan yang dimana data tersebut terkirim menjadi dua data yaitu data enkripsi dan data dekripsi. Hanya admin yang dapat melihat data enkripsi dan dekripsi di dalam *database*. Proses pengamanan meng-enkripsi data dengan bit input yang berbeda dan menerima *chipertext* yang berbeda, meskipun menghasilkan *chipertext* yang berbeda dengan jumlah bit yang sama. Ini karena nilai pembangkit kunci kunci (*p* dan *q*) diambil secara acak.



Gambar 6. Enkripsi *Database*.

Proses yang berjalan selama penyandian *chipertext* dikodekan dalam kode ASCII kemudian dikodekan dalam karakter, yang membuat proses mengubah enkripsi menjadi dekripsi yang sangat lama. Pesan yang ter-enkripsi dibagi menjadi array dengan batasan yang akan menghasilkan dekripsi.



Gambar 7. Dekripsi *Database.*

4.4 Pengujian Sistem

Pengujian sistem adalah proses menjalankan perangkat lunak sistem untuk menentukan apakah sistem mengetahui bahwa program yang sedang berjalan dan proses keamanan tidak dapat diubah oleh pengguna. Pengujian sistem dilakukan termasuk pengujian *black box*. Pengujian *black box* adalah proses yang meminimalkan kesalahan pada setiap proses enkripsi dan dekripsi dengan menjalankan atau mengeksekusi setiap proses, dan mengamati apakah hasilnya memenuhi persyaratan. Tes berjalan secara mandiri, yaitu setiap karyawan harus menginstal aplikasi absensi di *smartphone* masing-masing. Selanjutnya admin akan membuat akun untuk login karyawan. Setelah pengguna login untuk absensi, data akan otomatis dikirim ke *database* perusahaan melalui algoritma kata sandi RSA.

4.4.1. Pengujian Black Box

Menjalankan tes sederhana menggunakan metode *black box* untuk memverifikasi bahwa sistem sesuai dengan yang diharapkan. Metode ini menguji kemampuan program untuk mengeksekusi perintah yang diharapkan berdasarkan *input*, proses, dan *output*. berikut menunjukkan hasil uji fungsional aplikasi keamanan absensi. Hasil pengujian menunjukkan berhasil atau tidaknya sistem dalam melakukan perintah login, enkripsi, dan dekripsi.

a. Pengujian Login

Tabel 3. Pengujian Login

No	Data Masukkan	Hasil yang diharapkan	Hasil pengujian	keterangan
1.	User input *username* dan *password* secara benar	Setelah memasukkan data login dan mengklik tombol login, validasi data login dilakukan. Jika valid, user dapat mengakses halaman utama	User dapat login ke dalam aplikasi absensi dan mengakses menu utama	Berhasil
2.	User input *username* dan *password* secara salah	User tidak dapat melakukan login karena data tidak valid	Sistem tidak dapat menerima	Berhasil

b. Pengujian Enkripsi

Tabel 4. Pengujian Enkripsi

No	Data Masukkan	Hasil yang diharapkan	Hasil pengujian	keterangan
1.	User melakukan absensi *fingerprint*/sidik jari di dalam aplikasi tujuan pengirim masukan enkripsi (kunci publik nilai *n* dan *e*)	Sistem menyimpan data dalam database	Sistem menyimpan data absensi di database	Berhasil

c. Pengujian Dekripsi

Tabel 5. Pengujian Dekripsi

No	Data Masukkan	Hasil yang diharapkan	Hasil pengujian	keterangan
1.	Admin melakukan pengecekan data terhadap absensi masukkan dekripsi (kunci privat nilai *n* dan *d*)	Sistem akan menampilkan pesan yang sudah otomatis ter-dekripsi	Sistem menampilkan pesan yang sudah di dekripsi	Berhasil

5 Kesimpulan dan Saran

5.1 Kesimpulan

1. Dengan adanya sistem informasi absensi mobile karyawan ini dengan menerapkan algoritma RSA, proses penerimaan data absensi karyawan, proses pengiriman absensi dari proyek ke perusahaan yang sebelumnya tidak otomatis dan dilakukan manual, sekarang proses tersebut sudah menjadi lebih otomatisasi dan dapat mengurangi tingkat kecurangan data pada proses enkripsi dan dekripsi.
2. Aplikasi Android dan aplikasi web dirancang untuk saling terintegrasi, menunjukkan bahwa sistem dapat menyederhanakan proses pengiriman data dan entri data kehadiran karyawan untuk mengurangi dan mengoptimalkan .
3. Semakin tinggi angka yang digunakan, semakin sulit pesan atau kata sandi untuk dibaca atau ditebak oleh orang lain.

5.2 Saran

1. Keamanan tambahan disediakan oleh kebutuhan untuk studi lebih lanjut dari proses enkripsi dan dekripsi menggunakan algoritma kriptografi lainnya.
2. Diharapkan kedepannya aplikasi ini tidak hanya dapat digunakan di proyek lapangan saja, tetapi dapat digunakan oleh seluruh karyawan yang bekerja di perusahaan maupun di proyek lapangan PT.Arkonin. .
3. Aplikasi harus dikembangkan kembali untuk penggunaan smartphone untuk sistem operasi selain Android.

Referensi

[1] Manurung, J., Sirait, K., Panggabean, J. F., & Komputer, D. (2018). Penerapan Algoritma Rsa Untuk Pengamanan File. *Terakreditasi DIKTI*, *2*(2), 112–116.
[2] Hermansa, Umar, R., & Yudhana, A. (2019). Analisis Sistem Keamanan Teknik Kriptografi Dan Steganografi Pada Citra Digital (Bitmap). *Seminar Nasional Teknologi Fakultas Teknik Universitas Krisnadwipayana*, 1–9.
[3] Pratama, L., & Subandi, S. (2018). Pengamanan Table Database Menggunakan Kriptografi Algoritma Rsa. *Skanika*, *1*(3), 925–930.
[4] Sulaiman, R., & Vebu, M. (2018). Peningkatan Keamanan Pesan Berbasis Android Menggunakan Algoritma Kriptografi RSA. Jurnal Sisfokom (Sistem Informasi Dan Komputer), 7(2), 116. https://doi.org/10.32736/sisfokom.v7i2.574
[5] Enny Dwi Oktaviyani, Deddy Ronaldo, & Mustafa Arifin. (2019). Aplikasi Booking Kost Berbasis Android Di Kota Palangka Raya. *Jurnal Teknologi Informasi Jurnal Keilmuan Dan Aplikasi Bidang Teknik Informatika*, *13*(2), 1–11. https://doi.org/10.47111/jti.v13i2.250.
[6] Natsir, M. (2016). Pengembangan Prototype Sistem Kriptografi Untuk 61 Enkripsi Dan Dekripsi Data Office. Jurnal, 6, 2089–5615.
[7] Trimarsiah, Y., & Arafat, M. (2017). Analisis Dan Perancangan Website Sebagai Sarana. *Jurnal Ilmiah MATRIK*, *Vol. 19 No*, 1–10.
[8] Mitra, V., Sujaini, H., Negara, A. B. P. (2017). Rancang Bangun Aplikasi Web Scraping untuk Korpus Paralel Indonesia-Inggris dengan Metode HTML DOM. Jurnal Sistem dan Teknologi Informasi (JUSTIN), *5*(1), 36-41.
[9] Novendri, M. S., Saputra, A., & Firman, C. E. (2019). Aplikasi Inventaris Barang Pada Mts Nurul Islam Dumai Menggunakan Php Dan Mysql. *Lentera Dumai*, *10*(2), 46.