Jurnal Teknik: Universitas Muhammadiyah Tangerang,
Vol. 7, No. 2, Juli – Desember, Tahun 2018: hlm. 1-12
P-ISSN: 2302-8734
E-ISSN: 2581-0006

PENGAMANAN TEKS MENGGUNAKAN METODE ALGORITMA RSA DENGAN VERIFIKASI REALTIME BIOMETRIK MENGGUNAKAN OPENCV

ANGGA ADITYA PERMANA[1], & RACHMAT DESTRIANA[2]
Program Studi Teknik Informatika - Fakultas Teknik
Universitas Muhammadiyah Tangerang
Jl. Perintis Kemerdekaan I/33, Cikokol Kota Tangerang
Email: *anggaumt@gmail.com*[1], *rachmat.destriana@gmail.com*[2]

ABSTRAK

Penelitian tentang model pengamanan teks yang dapat digunakan sebagai salah satu instrumen sistem pengamanan teks. Adapun prinsip pengamanan dokumen khusunya teks ini adalah bagaimana sistem dapat mengamankan proses penyimpanan dan pengirimannya. Mula-mula teks dalam bentuk teks dienkripsi. Sehingga teks tersebut tidak dapat dibaca oleh siapapun. Karena teks telah berubah menjadi susunan huruf yang teracak. Dokumen yang susunan hurufnya telah teracak tersebut jika ingin dibaca oleh pemilik dokumen, maka dokumen tersebut harus dibuka dengan dekripsi. Dalam penelitian ini, metode yang digunakan adalah metode RSA, dimana metode tersebut menggunakan perhitungan matematika yang rumit dan disertai dengan kunci pengaman awal (*private key* atau *public key*) sehingga amat sulit untuk ditembus oleh orang yang tidak berkepentingan, adapun tanda tangan digital yang digunakan sebagai kunci alternatif disamping kunci private dan kunci publik. Dalam penelitian ini juga menggunakan metode *login* dan metode verifikasi wajah realtime biometrik pengenal wajah sebagai bentuk validasi keamanan lanjutan menggunakan *OpenCV library*. Sistem ini dibangun menggunakan perangkat lunak Android Studio 3.0 dengan bahasa pemprograman Java. Hasil pengujian ini menunjukkan bahwa sistem dapat meliki fitur-fitur yang telah disebutkan sebagai dinding keamanan berlapis. Mengenkripsi dan mendekripsi dengan verifikasi biometrik (wajah) yang telah direkam dan disimpan oleh aplikasi secara realtime untuk membuka aktifitas RSA guna melakukan proses enkripsi dan dekripsi.

***Kata Kunci**: Algoritma, RSA, Enkripsi, Dekripsi, OpenCV.*

1. PENDAHULUAN

Dewasa ini, Semakin pesatnya perkembangan teknologi informasi (TI) tidak akan pernah lepas dari permasalahan keamanan computer (*Computer Security*). Keamanan komputer sebagai isu yang tidak habis dibicarakan para pelaku bidang TI selalu menuntut adanya pembaharuan setiap saat dan berkala. Namun hal yang tidak kalah penting dari permasalahan keamanan komputer dalam hal keabsahan penyimpanan maupun pengiriman data atau File Dokumen.

Seperti pada kasus pengamanan *file* yang sering ditemui, pengguna yang tidak berhak mengakses file dapat mengakses file yang bukan haknnya. Maka, cara pengama-nan *file* paling praktis di era teknologi in-formasi dewasa ini ialah mengembangkan algoritma baru menggunakan kriptografi yang telah dikembangkan oleh banyak pe-ngembang serta menggabungkannya dengan metode verifikasi Computer Vision yang baru-baru ini banyak dikembangkan dan implementasinya. Karena bisa diakses dengan mudah, maka aspek-aspek keamanan dalam filter pengaksesan *user* sangatlah penting. Pengamanan berbasis *password* belum terlalu ampuh untuk mengamankan sebuah *file* pada perusahaan atau instansi, keamanan tersebut tetap saja belum mampu untuk

P-ISSN: 2302-8734
E-ISSN: 2581-0006

menghentikan para pengakses yang iseng untuk melakukan teknik brute force attack guna membobol *file*. Teknik *brute force attack* yaitu suatu teknik untuk mengakses data dengan cara melakukan pembobolan (*input password*) langsung secara acak pada file yang dituju. Bila teknik ini berhasil dilakukan, maka sudah bisa dipastikan bahwa data akan jatuh ke tangan orang yang tidak berkepentingan dan dengan mudah dapat dibaca.

Dalam dunia TI integritas suatu data yang disimpan terkadang juga menjadi pertanyaan, apakah data tersebut benar-benar aman oleh orang yang bersangkutan atau tidak, dan apakah isi dari data benar-benar otentik tanpa pengamanan. Hal ini merupakan masalah serius karena bisa saja sesorang mengakses dan mengubah data yang sebenarnya.

2. TINJAUAN PUSTAKA

Ada beberapa penelitian yang mendukung sebagai referensi penelitian ini:

1. Penelitian yang dilakukan oleh Leo Benny "Analisis dan Perancangan Aplikasi Kriptografi Keamanan File Berbasis Teks Dengan Menggunakan Metode RSA" yang diterbitkan oleh *Riset dan E-Jurnal Manajemen Informatika Komputer* Vol. 1 Nomor 2, April 2017 dengan nomor e-ISSN: 2541-1330 p-ISSN: 2541-1322. Yang telah meneliti pengirim file atau dokumen dengan metode pengamanan kriptografi RSA. Bahwa dengan menggunakan algoritma RSA yang menggunakan perhitungan matematika yang rumit maka pengguna yang tidak berhak tidak akan mampu membobol apa yang telah diamankan oleh algoritma RSA ini.
2. Penelitian yang di lakukan oleh Nugraha, Ary Reza dengan judul Penyembunyian pesan rahasia yang terenkripsi menggunakan algoritma RSA pada media kompresi. Jurnal Teknik POMITS Vol2, No.1, (2013), ISSN: 2337-3539 (2301-9271). Saat ini banyak sekali penerimaan dan pengiriman pesan yang beredar tetapi bias di lacak oleh orang yang tidak berkepentingan, Untuk itu diperlukan suatu cara untuk mengamankan pesan rahasia tadi agar tidak diketahui oleh orang yang tidak berkepentingan. Penyembunyian pesan rahasia yang berupa file dalam arsip ZIP dapat menjadi salah satu solusi untuk keamanan data yang bersifat rahasia jika data tersebut ingin dikirimkan. Arsip ZIP merupakan kumpulan dari beberapa *file* yang terkompresi dimana ukuran dari *file-file* tersebut beragam. Biasanya orang tidak memperhatikan ukuran *file* dari arsip ZIP karena ukuran dari *file-file* di dalam arsip ZIP tersebut terkompresi. Dari ha-sil uji coba yang dilakukan, file yang berisi pesan rahasia berhasil disembu-nyikan pada arsip ZIP serta tidak akan terbaca pada aplikasi pembaca arsip ZIP.
3. Penelitian yang di lakukan oleh Aditya Permana dengan judul "Kriptografi pada file Dokumen *Microsoft office* menggunakan metode RSA". Jurnal Komputer Program Studi Ilmu Komputer Universitas Brawijaya Malang, 2005. Dalam penelitian ini algoritma yang digunakan dalam proses enkripsi dan dekripsi adalah algoritma RSA dimana algoritma ini termasuk algorit-ma asimetris atau penggunaan dua kunci dalam proses dekripsi dan enkripsinya.

3. LANDASAN TEORI

1. Kriptografi

Kriptografi berasal dari akar kata Yunani *kryptos* dan *gráphō*, yang mempunyai arti "tulisan tersembunyi". Kriptografi adalah ilmu yang mempelajari bagaimana membuat suatu pesan yang dikirim pengirim dapat disampaikan kepada penerima dengan aman. Kriptografi dapat memenuhi kebutuhan umum suatu transaksi, yaitu:

a. Kerahasiaan (*confidentiality*) dijamin dengan melakukan enkripsi (penyandian).
b. Keutuhan (*integrity*) atas data dilakukan dengan fungsi *hash* satu arah.
c. Jaminan atas identitas dan keabsahan (*authenticity*) pihak-pihak yang melakukan transaksi dilakukan dengan menggunakan *password* atau sertifikat digital. Sedangkan keotentikan data transaksi dapat dilakukan dengan tanda-tangan digital.

P-ISSN: 2302-8734
E-ISSN: 2581-0006

d. Transaksi dapat dijadikan barang bukti yang tidak bisa disangkal (*non- repudiation*) dengan memanfaatkan tandatangan digital dan sertifikat digital.

RSA

Pada algoritma RSA terdapat tiga proses yaitu, pembangkitan kunci, proses enkripsi dan proses dekripsi. Letak kesulitan algoritma ini adalah bagaimana menemukan dua faktor bilangan prima yang besar yang akan digunakan sebagai kunci publik dan kunci privat. Dua bilangan prima besar tersebut p dan q dimana $p \neq q$.

Konsep utama keamanan dari RSA adalah susah pemfaktoran bilangan-bilangan besar menjadi faktor- faktor primanya. Terdapat besaran-besaran yang penting di algoritma RSA yakni:

1. p dan q bilangan prima (rahasia);
2. n = p.q (tidak rahasia);
3. φ (n) = (p - 1) (q - 1) (rahasia);
4. e (kunci enkripsi) (tidak rahasia);
5. d (kunci dekripsi) (rahasia);
6. m (plainteks) (rahasia); dan
7. chiperteks (tidak rahasia).

Teknik operasi pembangkitan kunci pada RSA adalah sebagai berikut

1. Memilih dua bilangan prima berbeda p dan q.
   - Untuk alasan keamanan, bilangan bulat p dan q dipilih secara ran-dom.
2. Compute n = p q. Hitung n = p.q
   - N digunakan sebagai modulus dari kunci publik dan kunci privat.
3. Hitung $\varphi(n) = (p-1)(q-1)$, di mana φ is fungsi Euler tosien.
4. Pilih sebuah bilangan bulat e sehingga $1 < e < \varphi(n)$ dan faktor pembagi terbesar dari $(e, \varphi(n)) = 1$; i.e., e dan φ(n) adalah relatif prima.
   - E digunakan sebagai eksponen kunci publik.
   - E mempunyai panjang bit yang pendek dan berat Hamming yang ringan menghasilkan hasil yang lebih efisien dalam enkripsi- umumnya 0x10001 = 65,537. Namun demikian, semakin kecil nilai e (such as 3) semakin kecil pula tingkat keamanan di hal-hal tertentu.
5. Berdasar technet.microsoft.com, penerapan RSA di dalam pertukaran kunci adalah dengan cara mengenkripsi kunci privat dari pesan dengan menggunakan kunci publik hasil pembang-kitan dari RSA dan pesan berisi kunci itu dapat dibuka hanya dengan kunci privat hasil pembangkitan RSA yang dimiliki oleh penerima pesan.

2. Verifikasi dan Validasi

Pengertian Verifikasi dan Validasi, Verifikasi merupakan proses pemeriksaan kesesuaian model logika operasional dengan logika diagram alur atau dapat disederhanakan dengan "apakah terdapat kesalahan dalam program?" (Hoover dan Perry, 1989). Sedangkan menurut (Law dan Kelton 1991) verifikasi merupakan suatu proses untuk memeriksa kesesuaian jalannya program komputer simulasi dengan yang diinginkan dengan cara melakukan pemeriksaan program komputer, selain itu verifikasi dapat diartikan sebagai proses penerjemahan model simulasi konseptual kedalam bahasa pemrograman secara benar.

Validasi merupakan proses penentuan apakah model konseptual simulasi benar-benar merupakan representasi akurat dari sistem nyata yang dimodelkan. Validasi model dapat pula dikatakan sebagai langkah dalam memvalidasi atau menguji apakah model yang telah disusun dapat merepresentasikan sistem nyata dengan benar. Suatu model dapat dikatakan valid ketika tidak memiliki perbedaan yang signifikan dengan sistem nyata yang diamati baik dari karakteristiknya maupun dari perilakunya. Validasi dapat dilakukan dengan menggunakan alat uji statistik yang meliputi uji keseragaman data output, uji kesamaan dua rata-rata, uji kesamaan dua variansi dan uji kecocokan distribusi (Law and Kelton,1991).

Validasi diartikan sebagai suatu tindakan pembuktian dengan cara yang sesuai bahwa tiap bahan, proses, prosedur, kegiatan, sistem, perlengkapan atau mekanisme yang digunakan dalam produksi dan pengawasan akan senantiasa mencapai hasil yang diinginkan.

Dari definisi-definisi tersebut tersebut di atas membawa pengertian, bahwa:

P-ISSN: 2302-8734
E-ISSN: 2581-0006

a) Validasi adalah suatu tindakan pembuktian, artinya validasi merupakan suatu pekerjaan "dokumentasi".
b) Tata cara atau metode pembuktian tersebut harus dengan "cara yang sesuai", artinya proses pembuktian tersebut ada tata cara atau metodenya, sesuai dengan prosedur yang tercantum dalam CPOB.
c) "Obyek" pembuktian adalah tiap-tiap bahan, proses, prosedur, kegiatan, sistem, perlengkapan atau mekanisme yang digunakan dalam produksi dan pengawasan mutu (ruang lingkup).
d) Sasaran/target dari pelaksanaan validasi ini adalah bahwa seluruh obyek pengujian tersebut akan senantiasa mencapai hasil yang diinginkan secara terus menerus (konsisten).

3. Biometrik

Biometrik berasal dari bahasa Yunani yaitu, *bios* yang berarti hidup dan *metron* berarti ukuran. Biometrik adalah suatu metode untuk mengenali manusia berdasar pada satu atau lebih ciri-ciri fisik atau tingkah laku yang unik. Alasan mengguna-kan Biometrik yaitu karena keterbatasan manusia memverifikasi segala hal hanya dari sisi:

a) Verifikasi berdasarkan kebendaan, semua data-data yang dibutuhkan berada pada suatu benda (seperti dokumen atau kartu kredit). Apabila hilang maka orang lain dapat memalsukannya atau menyalahgunakannya.
b) Verifikasi berdasarkan pengetahuan: biasanya menggunakan *password*, bahkan jika menggunakan algoritma enkripsi terbaikpun, tetap terdapat kunci yang bisa membukanya.

Tanda Tangan biometric didefinisikan sebagai proses menurunkan kunci privat dari sampel biometrik dan menggunakan kunci privat tersebut untuk menandatangani e-dokumen. Kunci privat yang unik dapat di-bangkitkan secara dinamis dari salah satu sampel biometrik tanpa memerlukan penyimpanan. Ini mengeliminir permasalahan tempat penyimpanan kunci privat dalam isu manajemen kunci privat. Pembangkitan kunci privat ini memudahkan penanda-tanganan dokumen seperti penandatanganan dokumen kapanpun dan dimanapun tanpa membawa fisik disk atau *smart card.*

4. OpenCV

OpenCV (*Open Source Computer Vision Library*) adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis secara *real-time*, yang pengembangannya diawali oleh Intel, dan sekarang didukung oleh Willow Garage dan Itseez. OpenCV dirilis dibawah lisensi permisif BSD yang lebih bebas dari pada GPL, dan memberikan kebebasan sepenuhnya untuk dimanfaatkan secara komersil tanpa perlu mengungkapkan kode sumbernya. Ia juga memiliki antar muka yang mendukung bahasa pemrograman C++, C, Python dan Java, termasuk untuk sistem operasi Windows, Linux, Mac OS, iOS dan Android. OpenCV didisain untuk efisiensi dalam komputasi dan difokuskan pada aplikasi *real-time*.

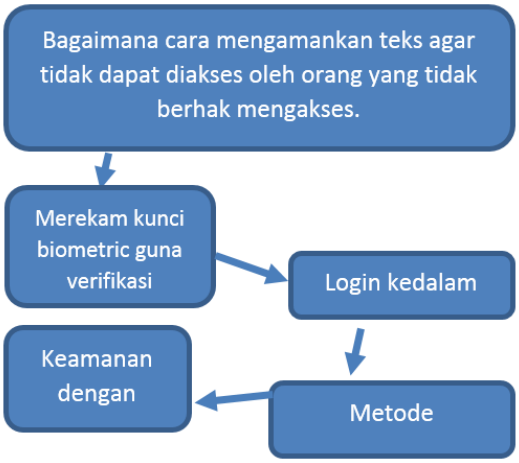
Ditulis dalam C++ yang dioptimalkan, dengan perpustakaan dapat memanfaatkan pengolahan pada prosesor inti majemuk (*multi-core processing*). Mendukung OpenCL, sekaligus memberikan keuntungan dari akselerasi *hardware* yang dimiliki *platform* komputasi heterogen. OpenCV telah diadopsi di seluruh dunia, memiliki lebih dari 47 ribu komunitas pengguna dengan estimasi *download* melebihi 7 juta. Penggunaannya openCV disebutkan, mulai dari seni interaktif, meliputi inspeksi penambangan (*mines inspection*), *stitching maps* di web, sampai dengan robotika maju (*advanced*).

6. Android

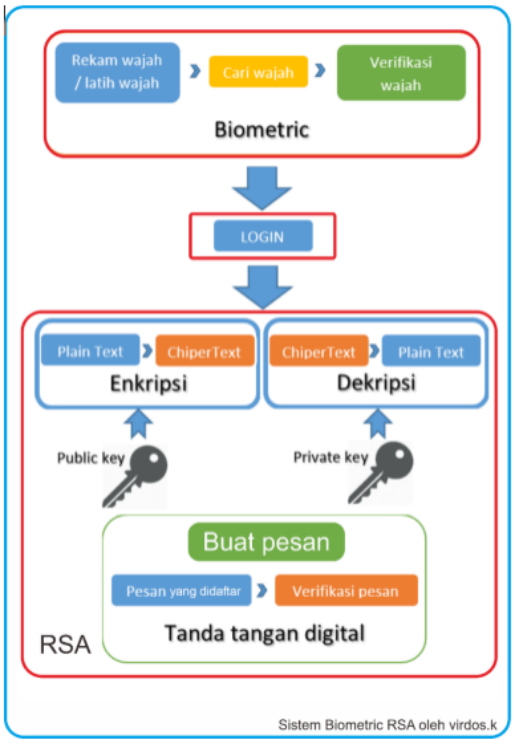
Android adalah sistem operasi (*Operating System*) yang umumnya digunakan pada perangkat dengan navigasi *full touch screen* yang biasa dimiliki oleh *smartphone* dan komputer tablet. Android sudah diambil alih oleh perusahaan Google Inc yang telah membelinya pada tahun 2005 dari Android Inc. Google menyediakan *software/tools* yang dikembangkan khusus untuk dijadikan alat pengembang aplikasi android yang di-beri nama "Android Studio". Android Studio dikembangkan dengan menggunakan bahasa Java dengan menambahkan *library-library* khusus yang diperuntukan untuk membuat aplikasi android. Android studio menggunakan metode *native code* yang memisahkan antara *view* dan *controller* (Permana, 2016).

P-ISSN: 2302-8734
E-ISSN: 2581-0006

6. Kerangka Pemikiran



Bagan 1 dekripsi kerangka pemikiran dari penelitian pada metode yang akan digunakan.



Bagan 2 Dekripsi sistem yang akan digunakan dalam penelitian algoritma RSA dengan pengenal wajah biometrik.

Pengenal wajah biometrik dibutuhkan sebagai keamanan berlapis sebelum memasuki class login dan RSA.

4. METODOLOGI PENELITIAN

Objek yang diteliti pada penelitian ini ialah cara mengamankan teks ataupun sebuah *class* RSA dengan menggunakan berbagai pendekatan demi mengembangkan bentuk keamanan yang lebih mumpuni, diterapkannya metode RSA dengan kunci private dan kunci publik yang dihasilkan secara acak oleh modul pada IDE Android Studio bahasa pemprograman Java, serta keamanan pengenal wajah sebelum akses *login* sebagai lapis keamanan ganda.

Yang biasa digunakan pada sistem keamanan saat ini adalah 1024 bit panjang modulus bit eksponen pada RSA. Sedangkan untuk perekaman kunci biometriknya menggunakan aplikasi yang dikembangkan menggunakan library OpenCV dengan cara pembelajaran atau pengenalan *node* objek yang akan direkam.

Ada beberapa metologi penelitian yang digunakan pada penelitian ini diantaranya:

a) Penelitian eksperimental, penelitian yang bertujuan untuk menyelidiki sebab akibat tertentu dengan memberikan perlakuan tertentu atau kondisi yang berbeda.
b) Observasi, Merupakan metode pengumpulan data dengan cara melakukan pengamatan secara langsung pada obyek yang diteliti yaitu biometric pengenal wajah dan tanda tangan digital serta RSA. Tentunya pada kasus ini menggunakan bahasa pemprograman *java*
c) Studi Pustaka, Merupakan metode pengumpulan data dengan cara mengumpulkan data-data dari berbagai sumber yang mendukung penelitian baik itu dari buku, jurnal ilmiah, makalah prosiding maupun artikel lainnya yang mendukung penelitian.

Alat penelitian yang digunakan dalam proses penelitian ini sebagai berikut:

1. Dell latitude E5430 spesifikasi Windows 10 Pro, 4 Gb RAM, Intel Core i5 2.4 Ghz.
2. Android Studio 3.0
3. OpenCV library 2140
4. Xperia Z1 C custom rom lollipop 5.1.2

P-ISSN: 2302-8734
E-ISSN: 2581-0006

1. Perancangan User Interface

Gambar 1 perancangan User Interface pengenal wajah dengan OpenCV

Mengadaptasi dari metode yang telah digambarkan maka rancangan *user interface* dapat dipetakan seperti gambar di atas.

Dengan beberapa *button*, *togglebutton* dan *java surface view* agar dapat mendeteksi wajah secara *realtime*. Penjelasan untuk beberapa *tool* yang ada pada form perancangan user interface diatas adalah sebagai berikut:

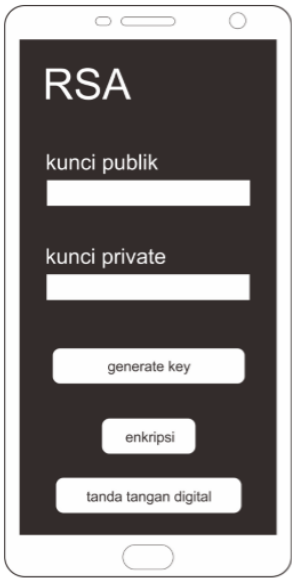
a) Java *surface view* (*Surface camera view*) guna menampilkan rekaman ka-mera yang membaca secara langsung wajah dengan menempatkan kotak hijau sebagai *landmark* wajah pada titik yang diidentifikasi sebagai wajah.
b) ImageView ikon menampilkan warna ikon (merah, kuning, hijau) merah mewakili tingkang kecocokan wajah kurang dari 50%, kuning mewakili tingkat kecocokan wajah 80%, hijau mewakili tingkat kecocokkan wajah >81%
c) *TextView* Status menampilkan kondisi atau modus dalam aktifitas ini (modus diam, melatih wajah atau mencari wajah)
d) *TextView* nama wajah menampilkan hasil prediksi wajah yang diidentifikasi secara *realtime*.
e) Button lihat wajah menampilkan *preview* wajah yang telah dilatih.
f) *EditText* guna menginput nama wajah yang sedang dilatih.
g) *Button Login* berfungsi melemparkan class dari aktifitas pengenal wajah ke *form login.*
h) *ImageView camera preview* berfungsi menampilkan gambar hitam putih yang telah diproses, diprediksi dan diidentifikasi secara *realtime*.
i) *ToggleButton* latihan berfungsi masuk ke mode latihan (*text* akan menjadi "berhenti latihan" setelah masuk ke mode latihan).
j) *ToggleButton* rekam berfungsi merkam wajah dan menyimpannya (*text* akan menjadi *"stop"* setelah ditekan).
k) *ToggleButton* cari berfungsi masuk ke modus pencarian wajah (*text* akan menjadi "berhenti mencari" setelah masuk ke mode mencari wajah).

Gambar 2 perancangan User Interface form login.

Pada *activity login* terdapat form dimana berfungsi untuk menginput username dan password, berikut penjelasan setiap tools yang ada pada *form login*.

a) *TextView login* menampilkan judul *activity login*.
b) *TextView Username* berfungsi menampilkan keterangan *EditText username*.
c) *EditText username* guna memasukkan *input username*.
d) *TextView password* berfungsi menampilkan keterangan *EditText password*.
e) *EditText password* berfungsi menginput data *password*.
f) *Button login* berfungsi melempar *class* ke *class MainActivity*.

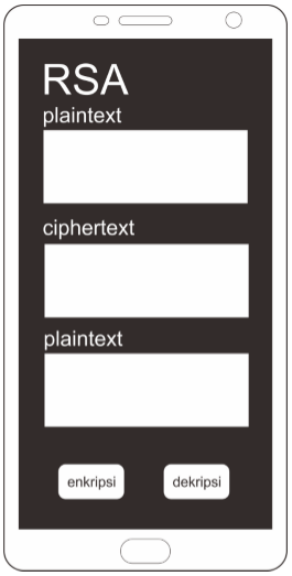
P-ISSN: 2302-8734
E-ISSN: 2581-0006



Gambar 3 perancangan User Interface Main Activity.

Pada *class MainActivity* terdapat beberapa *tools* dimana fungsi pengamanan teks RSA dan tanda tangan digital berjalan, berikut beberapa penjelasan dari *tools* yang ada di dalam *class MainActivity*.

a) *TextView RSA* menampilkan judul *MainActivity*.
b) *TextView* kunci public berfungsi menampilkan keterangan *EditText* kunci publik.
c) *EditText* kunci publik guna menampilkan kunci publik yang telah dihasilkan
d) *TextView* kunci private berfungsi menampilkan keterangan *EditText* kunci private.
e) EditText kunci private berfungsi menampilkan kunci private yang telah dihasilkan.
f) *Button generate key* berfungsi menghasilkan kunci publik dan kunci private.
g) *Button* enkripsi berguna melempar *class MainActivity* menuju ke *class RSA* (enkripsi dan dekripsi).
h) *Button* tanda tangan digital berfungsi melempar *class* menuju ke *class SignatureActivity*.



Gambar 4 perancangan User Interface RSA (enkripsi dan dekripsi).

Pada gambar 4 dijelaskan gambaran form RSA (enkripsi dan dekripsi) sebagai keamanan teks dengan metode RSA dengan beberapa *TextView, EditText* dan *Button*. Berikut penjelasannya.

a) *TextView RSA* menampilkan judul *form activity RSA.*
b) *TextView plaintext* berfungsi sebagai keterengan *EditText plaintext* dibawahnya.
c) *EditText plaintext* berfungsi menginput teks yang akan di enkripsi.
d) *TextView ciphertext* berfungsi sebagai keterangan *EditText chipertext* dibawahnya.
e) *EditText ciphertext* berfungsi menampilkan hasil plaintext yang telah dienkripsi (*ciphertext*).
f) *TextView plaintext* berfungsi sebagai keterangan *EditText plaintext* dibawahnya.
g) *EditText plaintext* (terkahir) berfungsi menampilkan hasil dekripsi *ciphertext* dari *EditText ciphertext*.
h) *Button* enkripsi berfungsi mengenkripsi teks dari *EditText plaintext* (pertama) dan menampilkannya di *EditText ciphertext*
i) *Button* dekripsi berfungsi mendeskripsikan teks *dari EditText chipertext* ke dalam *EditText plaintext* (terakhir).

P-ISSN: 2302-8734
E-ISSN: 2581-0006

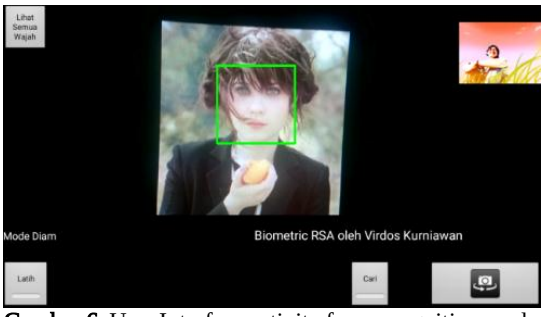
Gambar 5. perancangan User Interface tanda tangan digital

Pada form tanda tangan digital terdapat beberapa *tools* yang digunakan untuk menjalankan *activity*-nya, adapun penjelasan adalah sebagai berikut:

a) *TextView* tanda tangan digital menampilkan judul *activity*.
b) *TextView* pesan yang akan didaftar menjelaskan informasi *EditText* pesan yang akan didaftar dibawahnya.
c) *EditText* pesan yang akan didaftar berfungsi menginput pesan atau *text* yang akan didaftar.
d) *TextView* pesan yang telah didaftar menjelaskan EditText pesan yang telah didaftar dibawahnya.
e) *EditText* pesan yang telah didaftar menampilkan pesan yang akan didaftar setelah di encode dari *EditText* pesan yang akan didaftar.
f) *TextView* verifikasi pesan menjelaskan *EditText* verifikasi pesan dibawahnya.
g) *EditText* verifikasi pesan menampilkan menampilkan status sukses atau tidaknya verifikasi.

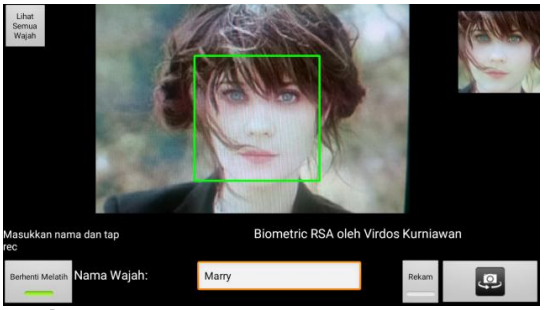
2. Eksekusi dan Analisa

Pada tahap eksekusi ini dilakukan dengan aplikasi hasil dari penelitian eksperimental yang telah dikembangkan juga menggunakan observasi pengumpulan data dan studi pustaka. Menghasilkan aplikasi hasil *debug* dari Android Studio 3.0 yang di debug dengan menggunakan Smartphone sony xperia Z1 compact custom rom android 5.1.2.



Gambar 6. User Interface activity face recognition mode diam menggunakan kamera belakang.

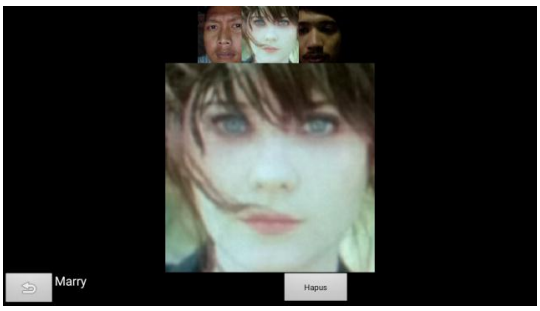
Pada *activity face recognition* ini ada dua dua pilihan mode yaitu mode latih wajah dan cari wajah. *Landmark* hijau yang telah sesuai mempersegi di area wajah menan-dakan kamera sudah siap membaca dan memproses gambar.



Gambar7 User Interface activity face recognition mode latih

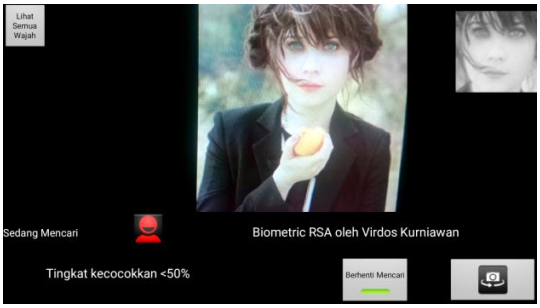
Mode latihan merubah *text* dari *ToggleButton* "latih" menjadi "berhenti melatih" dengan tombol menyala lalu menampilkan *EditText* penamaan wajah dan memunculkan *ToggleButton* rekam dalam keadaan mati, untuk menyimpan wajah maka tekan tombol rekam dan seketika landmark akan mencari posisi wajah yang akan disimpan setelahnya ditampilkan di *ImageView* sebelah pojok kanan atas.

P-ISSN: 2302-8734
E-ISSN: 2581-0006



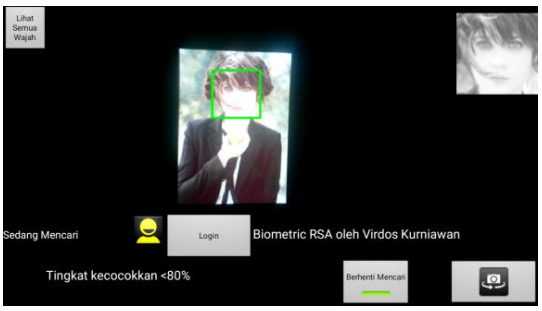
Gambar 7 User Interface activity face recognition wajah yang sudah direkam.

Pada *activity* ini wajah yang telah di-simpan dan diberi nama akan terlihat sebagai galeri wajah. Dengan *preview* di bagian tengah, pilihan wajah di bagian atas dan nama pojok kiri bawah bersama tombol kembali serta tombol hapus dibagian tengah.



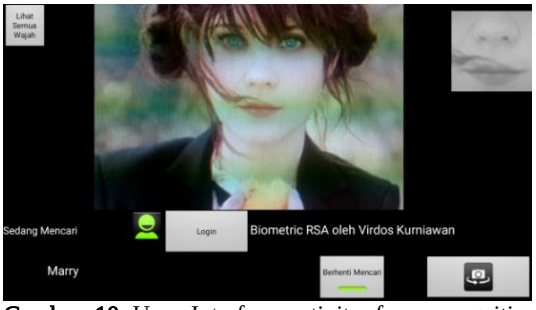
Gambar 8 User Interface activity face recognition mode cari <50%.

Mode latih seperti gambar diatas, *handler class facerecognition* ini menerap-kan *mLikely* sebagai definisi gambaran yang sama dengan kecocokan *node* setelah wajah direkam. Pada mode cari wajah ini, *text ToggleButton* "cari" akan menjadi "berhenti mencari" dengan tombol menyala, artinya pada proses ini, aplikasi sedang mencari dan mecocokkan wajah yang ditangkap oleh kamera lalu dicocokkan dengan wajah yang sudah pernah disimpan. Pada gambar diatas dapat dilihat bahwa prediksi wajah dengan tingkat kecocokan kurang dari 50% akan memunculkan *ImageView* ikon berwarna merah dan *TextView b*ertuliskan "Tingkat kecocokkan <50%". Saat pembacaan wajah kurang akurat kita harus memposisikan pengambilan tataletak wajah seperti yang pernah disimpan. Jika tidak tepat maka hanya akan terbaca <50% atau <80%. Maka dari itu pengambilan *realtime* sangat ber-pengaruh pada aplikasi ini untuk memenuhi kebutuhan pengguna yang membutuhkan keamanan dan kecepatan. Pengambilan gam-bar realtime ini hanya 2 kali pengambilan perdetik.



Gambar 9 User Interface activity face recognition mode cari <80%.

Pada gambar diatas terlihat bahwa pengenal wajah hanya mengenali wajah dengan tingkat kecocokkan <80% dan memunculkan tombol *login* agar pengguna dapat mengakses *form login*. *ImageView* ikon pun di ubah menjadi berwarna kuning.



Gambar 10 User Interface activity face recognition mode cari >80%

Pada saat pengenal wajah mengenali wajah hingga lebih dari 80% tingkat kecocokkan maka *TextView* tingkat kecocokkan akan berubah menjadi *text* nama si gambar yang telah disimpan. Tombol login pun pastinya muncul dan *ImageView* ikon berubah warna menjadi hijau.

Tombol login yang fluktuatif nampak dan menghilang pada <50% dan <80% saat proses pencarian wajah dikarenakan pada *handler class facerecognition* ini disetting untuk memudahkan proses *debug* saat mengakses tombol *login*.

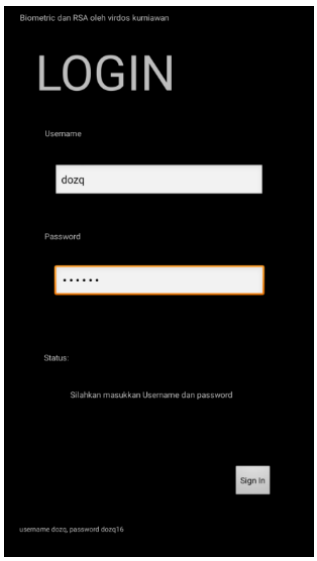
Pada implementasi tombol *login* sebenarnya menyulitkan pengguna yang membutuhkan untuk menekan layar pada saat pengguna mencocokkan wajah dengan aplikasi. Maka dari itu di dalam baris kode pada handler di bagian percabangan if nya dapat ditulis:

P-ISSN: 2302-8734
E-ISSN: 2581-0006

```
mHandler = new Handler() {
    @Override
    public void handleMessage(Message msg) {
        if (msg.obj=="IMG")
        {
         Canvas canvas = new Canvas();
          canvas.setBitmap(mBitmap);
          Iv.setImageBitmap(mBitmap);
          if (countImages>=MAXIMG-1)
          {          toggleButtonGrabar.setChecked(false);
            grabarOnclick();
          }
        }
        else
        {       textresult.setText(msg.obj.toString());          ivGreen.setVisibility(View.INVISIBLE);
ivYellow.setVisibility(View.INVISIBLE);          ivRed.setVisibility(View.INVISIBLE);
if (mLikely<0) {              ivRed.setVisibility(View.VISIBLE);              textresult.setText("Tingkat kecocokkan <0%");
            }
else if (mLikely<50) {              textresult.setText("Tingkat kecocokkan <50%");
ivRed.setVisibility(View.VISIBLE);
            }
else if (mLikely<80){              ivYellow.setVisibility(View.VISIBLE);
Intent x = new Intent(org.opencv.javacv.facerecognition.FdActivity.this,
org.opencv.javacv.facerecognition.Login.class);
startActivity(x);
            }
else {              ivGreen.setVisibility(View.VISIBLE);
Intent x = new Intent(org.opencv.javacv.facerecognition.FdActivity.this,
org.opencv.javacv.facerecognition.Login.class);
startActivity(x);

        }
    }
  }
};
```

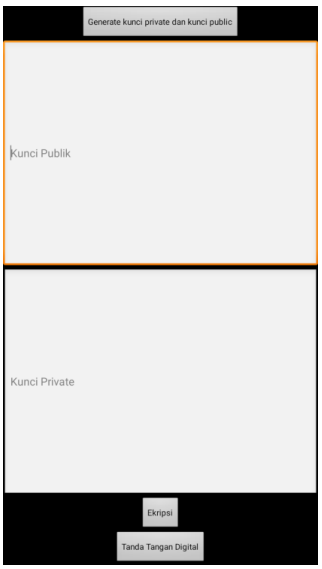
Code diatas guna melemparkan *user* menuju *activitiy login* setelah tingkat kecocokkan <80% dan >80% tanpa harus menekan tombol *login*.



Gambar 11 User Interface activity login.

Penerapan rancangan user *interface login* ini agak berbeda karena di lengkapi dengan *TextView* Status dimana akan menampilkan user berhasil login atau gagal *login*.

TextView ini dapat digantikan dengan Toast dari java Android Studio pada percabangan if yang telah ditulis didalam code yang akan menampilkan sekilas informasi status didalamnya.



Gambar 12 User Interface activity RSA penghasil kunci.

Pada activity ini kita telah masuk kedalam aplikasi RSA dimana dibagian ini menunjukkan form penghasil kunci publik dan kunci private. Mulai dari sini semua *TextView* yang menerangkan tentang *EditText* dibawahnya seperti pada perancangan *user interface* digantikan dengan property *"hint"* pada *EditText*. Seperti yang terlihat didalam gambar pada bagian dalam *EditText* terdapat *"hint" text* kunci publik dan kunci private.

Pada activity ini kita mengeksekusi awal dengan menekan tombol "*generate* kunci private dan kunci publik" pada bagian paling atas.

Maka akan terlihat kunci private dan kunci publik yang telah dihasilkan seperti pada gambar dibawah ini.

Setelah menghasilkan kunci kita dapat memilih dua activity yang terdapat pada tombol enkripsi (RSA) dan tombol tanda tangan digital.

P-ISSN: 2302-8734
E-ISSN: 2581-0006

Gambar 13 User Interface activity RSA penghasil kunci (telah di generate).

Gambar 14 User Interface activity RSA enkripsi dan dekripsi

Gambar diatas merupakan *activity RSA* enkripsi dan dekripsi saat *user* belum meng-*input text* apapun didalamnya.

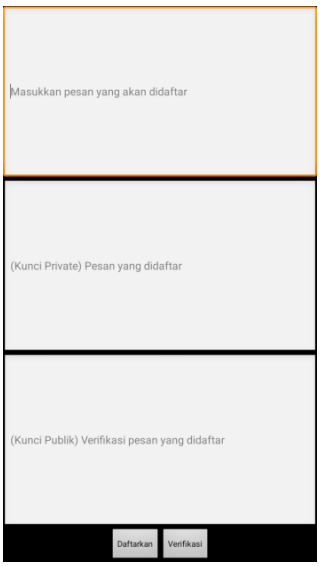
Gambar 15 User Interface activity RSA enkripsi dan dekripsi (setelah input).

Pada gambar diatas terlihat proses enkripsi dan dekripsi dengan 1024 bit panjang modulus bit eksponen RSA.

Enkripsi dan dekripsi ini menggunakan tools dari java yang kita import:

```
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;
```

Inti dari enkripsi dan dekripsi ini tidak menggunakan pengubahan modulus RSA yang terlalu mendalam. Disini hanya menggunakan default tools dari keamanan java sendiri.



Gambar 16 User Interface activity tanda tangan digital

P-ISSN: 2302-8734
E-ISSN: 2581-0006

Gambar 17 User Interface activity tanda tangan digital (setelah input).

Gambar di atas merupakan *activity* tanda tangan digital sebelum diinput dan sesudah diinput.

Pada daftarkan dan verifikasi pesan menggunakan kunci publik dan kunci private yang telah dihasilkan sebelumnya. Dapat mendaftarkan pesan dan memverifikasi

7. HASIL PENELITIAN

1. Kesimpulan

Dari penelitian yang telah dilakukan maka dapat ditarik beberapa kesimpulan yang menyangkut pendekatan dari penggu-naan metode RSA dan biometric pengenal wajah secara realtime sebagai berikut:

a) Data atau teks yang telah diamankan dengan metode RSA dan *biometric* pengenal wajah ini lebih aman karena mempunyai lapis keamanan (*layered security*) menggunakan *form login.* Tingkat efisiensinya bergantung pada implementasi OpenCV dan pembacaan node yang cepat dan ringkas.
b) Pengembangan aplikasi yang masih awal menyebabkan beberapa fitur yang tidak ada dan penggunaan user interface yang masih tergolong merepotkan. Jika kita bandingkan biometric pembaca sidik jari pada *smartphone* yang jauh le-bih cepat dan fleksible.

2. Saran

Dari segala pendekatan yang dilakukan, alur dari aplikasi ini sudah terlihat namun masih tersendat pada pengembangan aplikasi sendiri. Dengan mengandalkan kemampuan dari pengamanan kriptografi RSA dan *Computer Vision facerecognition* sebagai *biometric*. Pengembangan aplikasi diharapkan dapat berlanjut demi memajukan dunia keamanan komputer yang telah ramai dengan teknik pengenalan wajah ini.

DAFTAR PUSTAKA

Analisis dan Perancangan Aplikasi Kriptografi Keamanan File Berbasis Teks Dengan Menggunakan Metode RSA. Riset dan E-Jurnal Manajemen Informatika Komputer Vol. 1 Nomor 2, April 2017 dengan nomor e-ISSN: 2541-1330 p-ISSN: 2541-1322.

Penyembunyian pesan rahasia yang terenkripsi menggunakan algoritma RSA pada media kompresi. Jurnal Teknik POMITS Vol2, No.1, (2013), IISN: 2337-3539 (2301-9271).

Kriptografi pada file Dokumen Microsoft office menggunakan metode RSA. Jurnal computer Program Studi Ilmu Komputer Universitas Brawijaya Ma-lang, 2005.

https://github.com/ayuso2013/face-recognition (2018)

https://questdot.com/android-rsa-encrypt-decrypt-message-tutorial/ (2018)

Permana, A.A, 2016, *Model Layanan Informasi Lokasi Masjid di Wilayah Kota Tangerang Menggunakan Perangkat Bergerak (Mobile Device),* Jurnal Teknik, Vol 5, No 1 (2016), ISSN 2302-8734