PROGRAM STUDI SARJANA SISTEM INFORMASI

PROPOSAL PENELITIAN TUGAS AKHIR 1

Implementation Document Signing and Verification System

with Digital Signatures

TUGAS AKHIR 1

12S18003	**Citra Hutajulu**
12S18013	**Yudika Purba**
12S18020	**Dita L. Sastri Sihombing**

FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO

PROGRAM STUDI SARJANA SISTEM INFORMASI

INSTITUT TEKNOLOGI DEL

LAGUBOTI

MARET 2022

Daftar Isi

DAFTAR TABEL

DAFTAR GAMBAR

BAB 1 PENDAHULUAN

Bab ini menjelaskan latar belakang, pertanyaan penelitian, tujuan penelitian, serta ruang lingkup dari penelitian

1.1 Latar Belakang

Tanda Tanda tangan basah (tanda tangan yang ditulis secara langsung) adalah nama tertulis atau tanda hukum seseorang, ditulis tangan oleh orang tersebut dan dibuat atau diadopsi dengan maksud, untuk mengesahkan suatu tulisan dalam bentuk permanen [1]. Tanda tangan tulisan tangan umumnya diperlukan untuk persetujuan, kepemilikan yang sah, pembuktian transaksi, dan terutama yang terkait dengan hasil uji klinis. Tanda tangan masih sering ditemukan dalam kehidupan sehari hari.

Tanda tangan merupakan (metode) yang mengandung sifat biometrik[1]. Sifat biometrik ini membahas terkait kemampuan yang membangun identitas seseorang dan dapat dilihat pada sifat fisik (*physical traits*) ataupun pola tingkah laku (*behavioral traits*). Contoh fisik adalah sidik jari sedangkan contoh tingkah laku adalah tanda tangan basah. Dengan mengandung biometrik ini membuat keamanan lebih meningkat dibandingkan dengan pembuatan sandi dari karakter perpaduan huruf dan angka. contoh; Xw8uyuZ

Tanda tangan merupakan salah satu atribut personal yang diterima secara luas [2]. Penandatanganan akan dilakukan terhadap dokumen yang menyangkut identitas seseorang, dan dokumen lalu, dapat diverifikasi oleh sipenerima. Pengirim pesan menghitung *Message Digest (*MD*)* dari pesan, MD diperoleh dengan mentransformasikan pesan M dengan fungsi hash satu arah. Kemudian, dienkripsi dengan algoritma kriptografi kunci privat salah satu misalnya algoritma RSA. Hasil enkripsi inilah yang disebut digital signature, ini diletakkan pada pesan M dan dikirim melalui saluran komunikasi, pesan M telah ditandatangani S.

Verification adalah serangkaian proses digital untuk mengkonfirmasikan keakuratan dari yang bersangkutan dalam memberikan tanda tangannya. Untuk mengembangkan tanda tangan secara digital tersebut seharusnya melibatkan skema *Cryptography* dalam melakukan *decryption* dan *encryption* pada proses *Signing* dan *Verification*

Klasifikasi tanda tangan berdasarkan bentuknya yaitu tanda tangan basah dan tanda tangan secara digital yang di sebut *digital signature*. Untuk klasifikasi tanda tangan digital berdasarkan keabsahannya yaitu, pertama, tanda tangan digital tersertifikasi yang harus memenuhi persyaratan: memenuhi keabsahan kekuatan hukum dan akibat hukum tanda tangan digital; menggunakan sertifikat digital yang dibuat oleh jasa penyelenggara sertifikasi digital suatu negara atau badan resmi yang diakui secara nasional maupun internasional dan dibuat dengan menggunakan perangkat pembuat tanda tangan digital tersertifikasi. Kedua, tanda tangan digital tidak tersertifikasi, yang dibuat tanpa menggunakan jasa penyelenggara sertifikasi elektronik suatu negara atau badan resmi yang diakui secara nasional maupun internasional

Fungsi *hash* merupakan nilai kode yang didapatkan dari hasil enkripsi, memiliki fungsi untuk memetakan suatu pesan (*bit-bit string*) dengan panjang sembarang ke sebuah *string* dengan panjang tertentu. Karakteristik fungsi hash adalah pemetaan dari *many-to-one*. Hal ini memungkinkan terjadinya pasangan masukan berbeda memiliki hasil pemetaan yang sama. Dimana nilai *hash* berlaku sebagai representasi sederhana dari suatu masukan *string*, dan hanya dapat digunakan jika nilai *hash* tersebut dapat diidentifikasi secara unik dengan masukan *string* tersebut. Artinya, untuk melakukan otentikasi dua buah dokumen, setiap dokumen perlu dipetakan terlebih dahulu dengan suatu fungsi *hash.* Apabila kedua dokumen menghasilkan nilai *hash* yang sama, berarti kedua dokumen diasumsikan masih sama.

Confidentiality, *Integrity* dan *Authentication* menjadi salah kriteria dari sebuah dokumen dapat dikatakan legal. Hal ini dapat dilakukan dengan menerapkan *hash*. Fungsi *hash* ini memiliki kemampuan dalam menjamin originalitas data yang dapat menghasilkan ciri khas *signature* yang masih terjamin keasliannya [3]. Kriptografi *public key* atau *asymetric key* banyak digunakan dalam implementasi keamanan data pada sistem informasi dan komunikasi. Algoritma yang digunakan seperti *Rivest, Shamir, dan Adleman* (*RSA*) dengan dua fungsi utama yaitu proses enkripsi dan dekripsi berbeda pada *Digital Signature Algorithm* (*DSA*) dua fungsi utama yaitu membuat tanda tangan digital dan memeriksa validitas tanda tangan

Tanda tangan digital sangat dibutuhkan dalam seluruh instansi seperti instansi pemerintahan, perusahaan (negeri/swasta) ataupun instansi pendidikan. Pada instansi pendidikan Institut Teknologi Del merupakan salah satu perguruan tinggi yang membutuhkan tanda tangan digital (*digital signature*). Dalam lingkungan kampus IT Del aktivitas membubuhi tanda tangan pada dokumen resmi seperti ijazah mahasiswa masih dilakukan secara manual, tentunya dibutuhkan banyak waktu untuk melakukan pekerjaan tersebut. Selain itu juga, dapat memberikan peluang kesalahan yang cukup besar dalam proses penandatanganan secara manual apabila terdapat jumlah dokumen yang sangat banyak.

Kelompok TASI-2122-202 mendapatkan ide penelitian ini dari Rektor Institut Teknologi Del (IT Del) dimana ketika alumni mengunjungi kampus untuk keperluan legalisir dokumen ijazah, pihak dari kampus tidak dapat melakukan pengecekan terhadap dokumen untuk mengetahui dan memastikan keasliannya karena mungkin saja akan terdapat perubahan yang dilakukan oleh pihak yang berkepentingan (Alumni) terhadap dokumen yang ingin dilegalisir. Sehingga, dengan adanya Implementasi *Signing* dan *Verification* pada sistem dengan menggunakan *Digital Singnatures* dapat dijadian sebagai solusi untuk meningkatkan data yang akurat oleh kampus IT Del.

Pada *Implementation Signing and Verification system with Digital Signatures* akan menjadi sistem yang akan bermanfaat untuk kampus IT Del karena sistem yang akan diimplementasikan akan tersedia secara gratis (*unpaid*) dan dapat dimodifikasi untuk menghasilkan fitur – fitur baru yang akan meningkatkan kualitas sistem (*open source*).

1.2 Pertanyaan penelitian

Berdasarkan ide yang dijelaskan pada latar belakang, rumusan masalah dari penelitian yang dilakukan adalah:

1. Bagaimana implementasi sistem yang efektif untuk proses *signing & Verification* menggunakan *Digital signature*?

1.3 Tujuan Penelitian

Adapun tujuan dari pelaksanaan penelitian ini adalah untuk mengimplementasikan sistem yang efektif untuk *Signing and Verification with Digital Signature* serta merealisasikan sistem tersebut pada dokumen resmi di Institut Teknologi Del.

1.4 Ruang Lingkup

Pada penelitian ini dibahas mengenai sistem pemeriksaan dan penerapan *digital signature* pada dokumen formal akademik sebagai perangkat lunak dalam *signing & verification* pekerjaan dalam dokumen resmi IT Del. Berikut batasan yang terdapat pada penelitian ini:

1. Dokumen *softcopy* yang akan dikelola untuk penelitian ini adalah dokumen resmi yang dikeluarkan IT Del yaitu Ijazah Mahasiswa dan legalisirnya
2. Data yang digunakan untuk melengkapi informasi pada Ijazah Mahasiswa bersumber dari *Campus Information System* (CIS) IT Del

1.5 Sistematika Penyajian

Sistematika penulisan dari pelaksanaan Tugas Akhir dengan topik *Digital Signature* adalah sebagai berikut:

1. Bab 1 Pendahuluan. Pada Bab ini dijabarkan tentang latar belakang penelitian rumusan masalah, tujuan penelitian, ruang lingkup, dan sistematika penulisan penelitian.
2. Bab 2 Landasan Teori. Pada Bab ini berisi informasi dan teori para ahli yang mendukung penelitian sebelumnya terkait pemanfaatan teknologi dalam melakukan penandatangan secara digital dan dengan source berhubungan lainnya.
3. Bab 3 Analisis. Bab ini menjelaskan tentang metode atau langkah langkah yang dilakukan dalam penelitian Tugas Akhir. Selain itu, pada bab ini juga akan dijelaskan tentang analisis landasan teori sebagaimana telah di jelaskan pada bab 2.

BAB 2 LANDASAN TEORI

Bab ini menjelaskan tentang landasan teori atau tinjauan pustaka yang berisi antara lain: teori, metode, teknik, proses dan perangkat (*tools)* yang terkait dengan tema penelitian. Selain itu juga terdapat kajian mengenai hasil penelitian yang telah dilakukan sebelumnya yang berkaitan dengan topik penelitian.

2.1 Dokumen Formal Akademik

Menurut Kamus Besar Bahasa Indonesia (KBBI) Dokumen dapat didefinisikan sebagai surat yang tertulis atau tercetak yang dapat digunakan sebagai bukti sebuah keterangan (*evidence*). Formal dapat di defenisikan legal atau sesuai dengan peraturan yang sah, Akademik di defenisikan suatu teori dan bersifat Ilmu Pengetahuan sehingga, Dokumen Formal Akademik dapat diartikan sebagai berkas yang dihasilkan dari proses memperoleh ilmu pengetahuan yang berisi teks secara legal.

Oleh karena itu, perlu diketahui elemen apa saja yang tercantum dalam sebuah dokumen sehingga dapat dikatakan dokumen legal.

Elemen	Deskripsi
1. Kepala surat	Berisi tentang nama perusahaan, jenis perusahaan dan alamat lengkap juga symbol dari perusahaan
2. Data penanggalan surat	Berisi tentang data dokumen seperti hari/tanggal/tahun
3. Nomor surat	Surat yang diterbitkan dengan nomor yang sesuai
4. Lampiran	Lembar tambahan yang bisa dilampirkan
5. Hal/perihal	Menunjukan isi atau inti dari surat secara singkat
6. Alamat tujuan	Alamat ditujukannya sebuah surat

7. Nama, Jabatan dan Tanda tangan	Keterangan identitas pembuat surat
8. Tembusan	Untuk mengetahui kepada pihak mana saja surat dikirimkan

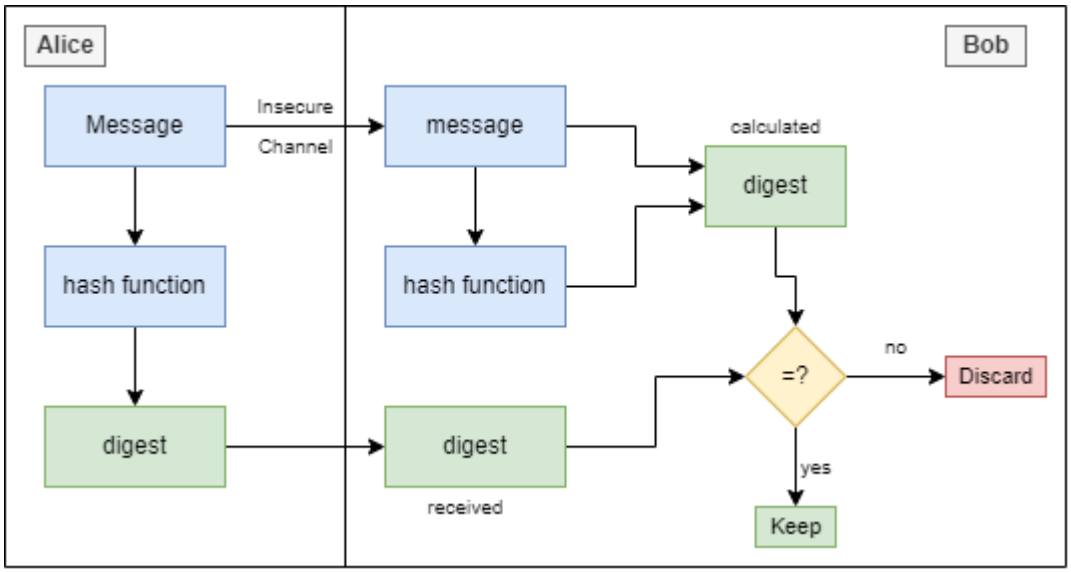
Tabel 1. Elemen Dokumen

2.2 Message Digest (MD)

Message Digest (Intisari pesan) merupakan fungsi hash kriptografi yang memiliki serangkaian angka yang terbentuk berdasarkan rumus *hashing* searah. MD dirancang untuk melindungi keutuhan (integritas) suatu data atau media untuk digunakan sebagai pendeteksi letak dan perubahan yang terjadi pada pesan. MD juga sejenis kriptografi yang memanfaatkan nilai hash untuk memberi peringatan kepada pemilik pesan dari setiap modifikasi yang diterapkan pada dokumen digital.

Message Digest mengandung nomor kepemilikan (number hash) yang mewakili dokumen tertentu untuk melindungi pesan ynag ingin disampaikan agar terhindar dari pihak ketiga. Dapat dilihat pada gambar 1. Cara kerja MD sangat kompleks, 1MD ditugaskan untuk konten data tertentu juka memiliki perubahan yang sengaja dibuat atau tidak sengaja akan meminta pengirim (sender) untuk mengidentifikasi, modifikasi pihak yang melakukan perubaha.

Message Digest akan berubah jika file berubah. *Message Digest* dapat membantu menemukan file duplikat. *Message Digest* dapat diproduksi pada sistem dengan perintah MD. MD disimpan dengan aman di sistem dan dapat mengungkapkan pengguna yang tidak sah telah mengakses file. Namun, MD5 tidak dapat diandalkan untuk permasalahan pengiriman pesan yang berkaitan dengan pesan yang tabrakan (di mana 2 kunci untuk data yang berbeda adalah sama) dan tidak lagi digunakan.



Gambar 1. Message Digest

Message Digest memanfaatkan program berbagi file peer-to-peer (P2P) untuk memberikan pesan peringatan (*warning*) kepada pengguna saat mengunduh file yang identik. Itu, juga dapat menunjukkan dengan tepat asal unduhan duplikat.

Intisari pesan dienkripsi dengan kunci pribadi yang membuat tanda tangan digital. Ini menghasilkan jenis validasi yang memastikan bahwa pengguna yang tepat mengakses informasi yang di lindungi. Intisari pesan melindungi algoritma hash satu arah yang mengambil data acak dan mentransmisikan nilai hash panjang yang ditetapkan.

Untuk memulai proses, intisari pesan diinisialisasi. Kemudian data tersebut diolah melalui message digest dengan menggunakan update. Operasi terakhir termasuk *padding*, di mana *message digest* menyelesaikan perhitungan *hash* dan melakukan reset terhadap data . *Message digest* juga dapat diatur ulang kapan saja selama proses pengiriman data terjadi.

2.3 Message Digest Algorithm 5 (MDA5)

Message Digest Algorithm 5 (MDA5) merupakan salah satu dari banyak algoritma kriptografi (merupakan salah satu fungsi Hash). MDA5 sering digunakan untuk menyimpan kunci dan sering digunakan pada *digital signature*. Ciri yang spesifik dari MDA5 adalah RFC (*request for comment*). Dalam hal kerja AMD5 memiliki besar blok sebanyak 512 bit sedangkan ukuran *digest* adalah 128 bit. Karena *word size* ditentukan sebesar 32 bit, satu blok terdiri

dari 16 word sedangkan digest terdiri dari 4 word. Indeks untuk bit dimulai dari 0. Proses awal dimulai dengan padding sebagai berikut :

1. Bit dengan nilai 1 ditambahkan setelah akhir naskah
2. Deretan bit dengan nilai 0 ditambahkan setelah itu sehingga besar dari naskah mencapai nilai 448 /mod 512 (sedikitnya 0 dan sebanyaknya 511 bit dengan nilai 0 ditambahkan sehingga tersisa 64 bit untuk diisi agar besar naskah menjadi kelipatan 512).
3. 64 bit yang tersisa diisi dengan besar naskah asli dalam bit. Jika besar naskah asli lebih dari 2^{64} bit maka hanya 64 lower order bit yang dimasukkan. Lower order word untuk besar naskah asli dimasukkan sebelum high order word.

Setelah *padding*, naskah terdiri dari n-word M [0 . . . n -1], dimana n adalah kelipatan 16. Langkah berikutnya dalam *preprocessing* adalah menyiapkan MD5 *buffer* sebesar 4 word, yaitu:

(A, B, C, D)

Dimana A merupakan lower order word. *Buffer* diberi nilai awal sebagai berikut (nilai dalam hexadecimal dimulai dengan lower order byte).

A : 01 23 45 67

B : 89 ab cd ef

C : fe dc ba 98

D : 76 54 32 10

Proses *hashing* dilakukan perblok, dimana setiap blok melalui 4 putaran. Proses *hashing* menggunakan 4 fungsi yaitu F, G, H, dan I yang masing – masing mempunyai input 3 word dan output 1 word :

$F(X,Y,Z) = (X \wedge Y) \vee (\neg X \wedge Z)$

$G(X,Y,Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$

$$H(X,Y,Z) = X \oplus Y \oplus Z$$

$$I(X,Y,Z) = Y \oplus (X \vee \neg Z)$$

Dimana $\wedge$ adalah *bitwise and*, $\vee$ adalah *bitwise or*, $\oplus$ adalah bitwise exclusive or dan $\neg$ adalah *bitwise* not.

Kriptografi dirancang untuk menjadi instrument untuk mempelajari prinsip dan teknik dalam menyembunyikan informasi kedalam bentuk *ciphertext* yang kemudian diberikan kepada orang yang sah dengan kunci yang dimilikinya. Teknik tersebut dilakukan agar orang yang tidak berkepentingan dalam informasi tersebut tidak dapat membuka dan merubah informasi tersebut.

Ada beberapa bentuk serangan yang dapat dilakukan pada algoritma *hash* antara lain:

Solanki & Agarwal (2012) menguraikan antara lain:

1. *Pre-image Attacks*

   Dengan teknik serangan pre-image attacks kripanalis dapat menemukan input hash dari hasil keluaran yang sudah ditentukan.

2. *Second Pre-image Attacks*

   Serangan kedua *pre-image* ini sama dengan serangan *pre-image* yang pertama tetapi disini ditujukan untuk menemukan masukan kedua yang memiliki ouput informasi yang sama dengan input tertentu.

3. *Collision Attacks*

   Serangan *collision* adalah usaha untuk menemukan dua pesan M1 dan M2 yang memiliki nilai hash yang sama. Kripanalis masuk pada satu pesan tetapi mencoba juga masuk pada pesan yang berikutnya.

4. *Birthday Attack terhadap fungsi hash satu arah*

   Dalam serangan *birthday* atau sering disebut *brute force* terhadap fungsi *hash* satu arah terdapat dua bentuk serangan. Yang pertama adalah mengingat *hash* dari pesan H(M), kemudian kripanalis membuat pesan lain dengan nama (M'), sehingga didapat H(M) = H(M'). Kedua adalah bentuk serangan yang lebih halus yaitu kripanalis menemukan dua

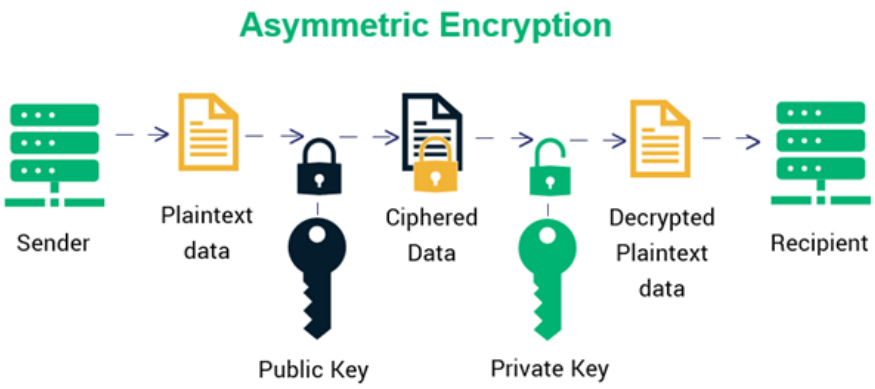
pesan acak M dan M’ sehingga H(M) = H(M’). ini yang disebut dengan tabrakan, dan serangan ini jauh lebih mudah dari yang pertama. Serangan kedua ini, umumnya dikenal sebagai *birthday attack.*

2.4 Asimetric Encryption

Enkripsi asimetris bekerja menggunakan dua kunci terpisah tetapi memiliki keterkaitan secara matematis untuk melakukan enkripsi dan mendekripsi data. Kunci publik melakukan dekripsi pada data sementara kunci pribadi melakukan enkripsi data, hal ini menjadi dikenal sebagai enkripsi kunci publik, kriptografi kunci publik, dan enkripsi kunci asimetris.

Kunci publik dapat diakses oleh semua pihak yang terlibat dalam pengiriman pesan dan dapat diakses serta mengenkripsi data dengan menggunakan kunci publik. Kunci pribadi digunakan untuk melakukan proses enkripsi pesan, setelah pesan dienkripsi data hanya dapat dibuka dengan menggunakan kunci pribadi yang sesuai. Kunci pribadi harus dirahasiakan agar tidak disalahgunakan. Hanya orang, server, mesin, atau instrumen yang berwenang yang memiliki akses ke kunci pribadi (*private key*).

Enkripsi asimetris dapat digunakan untuk memverifikasi pihak ketiga yang belum pernah ditemui melalui saluran publik yang tidak aman. Tidak seperti metode enkripsi tradisional (simetris), yang mengandalkan satu kunci untuk mengenkripsi dan mendekripsi data, enkripsi kunci asimetris menggunakan dua kunci terpisah untuk menjalankan enkripsi dan dekripsi.



Gambar 2. Asymetric Encryption

Metode enkripsi asimetris digunakan untuk : melakukan autentikasi, verifikasi integritas data, serta pertukaran kunci simetris. Enkripsi simetris gunakan untuk menangani sebagian besar enkripsi data. Terdapat 4 karakteristik Utama Enkripsi Asimetris, antara lain:

1. Enkripsi Asimetris Dirancang untuk Mengamankan Data & Pertukaran Kunci di Saluran Publik.

   Enkripsi kunci asimetris bertujuan untuk mengenkripsi data dengan aman di saluran publik sambil juga, menawarkan otentikasi dan integritas data. Karena tidak memerlukan pertukaran kunci, tidak ada masalah distribusi kunci yang anda miliki dengan enkripsi simetris.
2. Kunci dengan enkripsi asimetris.

   Kunci publik dan pribadi asimetris merupakan angka yang diacak secara unik dan besar. Setiap kunci harus acak dan tidak dapat diprediksi sehingga super komputer modern membutuhkan waktu yang lama untuk menebaknya
3. Algoritma enkripsi kunci publik yang kuat.

   Algoritma yang sangat populer untuk enkripsi asimetris dan pertukaran kunci adalah RSA, ElGamal, dan DSA. Meskipun bukan aturan yang ketat, sebagian besar waktu, enkripsi asimetris menggunakan kunci sepanjang 1024 bit, 2048 bit, atau lebih. Semakin panjang ukuran kunci, semakin aman enkripsi. Ketika kunci dibuat dengan enkripsi 2048-bit, ada 22048 kemungkinan kombinasi. Super komputer modern membutuhkan waktu ribuan tahun untuk melewati begitu banyak kombinasi untuk dapat menemukan kunci pribadi yang sesuai dari kunci publik.
4. Enkripsi Asimetris merupakan Proses Pembobotan Sumber Daya .

   Kunci besar itu membutuhkan banyak sumber daya dan itu berarti enkripsi membutuhkan waktu lebih lama untuk diselesaikan. Dengan kata lain, karena ukuran kunci lebih besar dan dua kunci terpisah terlibat, proses enkripsi dan dekripsi menjadi lebih lambat.

   Dengan demikian, enkripsi asimetris paling cocok untuk mengenkripsi potongan kecil data karena latensi dan persyaratan pemrosesannya. Jika anda menggunakannya untuk blok data yang besar, itu akan memberi lebih banyak beban ke server Anda. Terkadang, ini digunakan untuk awalnya membangun saluran komunikasi yang aman, yang dapat digunakan untuk memfasilitasi enkripsi simetris untuk bertukar

2.5 Kriptografi

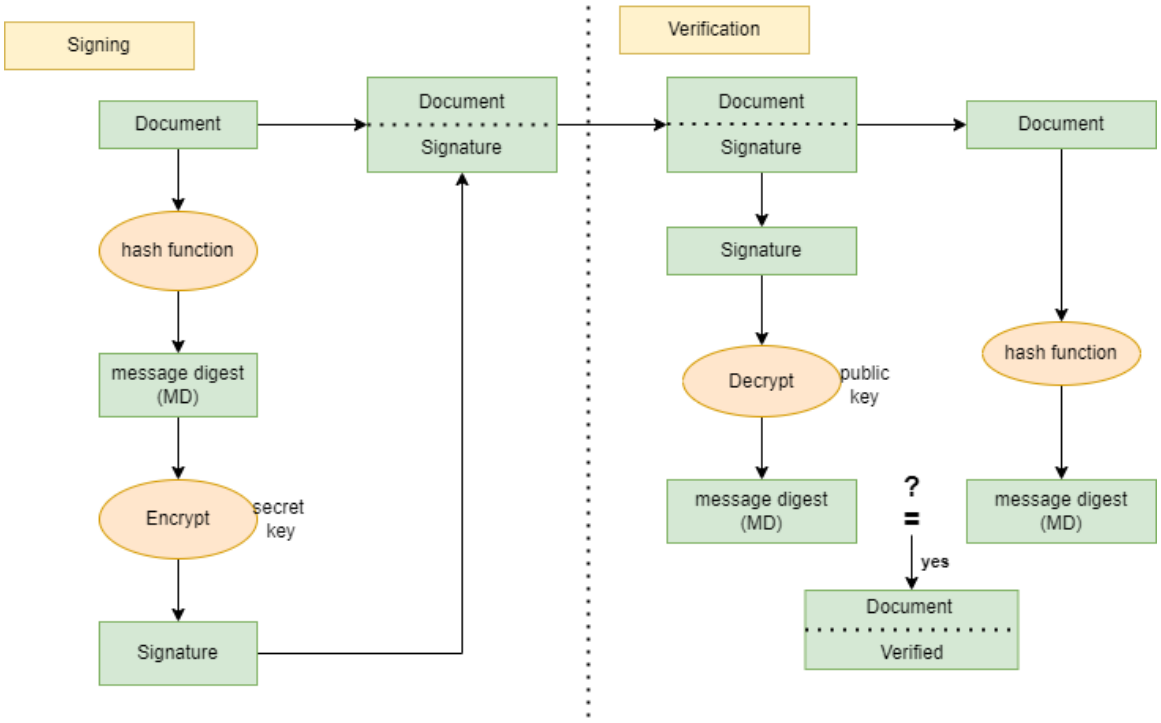
Kriptografi merupakan sebuah teknik untuk mengamankan sebuah pesan.Tujuan digunakannya kriptografi adalah suatu seni dan ilmu pengetahuan untuk menjaga keamanan suatu pesan [1].Kriptografi adalah suatu studi tentang teknik-teknik matematis yang berhubungan dengan aspek-aspek keamanan informasi seperti kerahasiaan *(confidentiality),* keutuhan data *(data integrity)* dan otentikasi *(authentication)* [2]:

a) Confidentiality berkaitan dengan pelayanan yang ditujukan untuk melakukan penjagaan terhadap pesan sehingga tidak publik dibaca oleh pihak yang tidak berwenang
b) Data Integrity berkaitan dengan pelayanan yang menjamin jika sebuah pesan adalah asli dan tidak pernah dilakukan manipulasi dalam proses pengiriman ke penerima pesan
c) Authentication lebih terkait dengan pelayanan dengan adanya identifikasi baik mengidentifikasikan kebenaran pihak pihak yang berkomunikasi (*user authentication atau identity authentication*) atau mengidentifikasi kebenaran sumber pesan tersebut (*data origin authentication*)
d) Non-repudiation berkaitan dengan bentuk pelayanan dalam mencegah entitas yang melakukan komunikasi penyangkalan, yakni pengirim pesan menyangkal melakukan pengiriman pesan atau penerima menyangkal telah menerima pesan.

2.6 Asimetric Encryption

Digital Signature merupakan tanda tangan dengan melibatkan proses enkripsi dan dekripsi. Sistem dengan Digital Signature digunakan untuk meningkatkan kualitas pelayanan penerbitan dokumen. Data yang ditambahkan ke,atau informasi kriptografi dari unit data yang memungkinkan penerima data untuk membuktikan sumber dan Integritas data [4] ini dapat melindungi terhadap perilaku atau tindakan pemalsuan. Digital Signature berfungsi sebagai penanda untuk memastikan bahwa data berkas dokumen terintegrasi dan identitas *verify* digital signature tersebut dilakukan dengan fungsi *hash* yang memastikan penandatangan oleh pihak yang berkepentingan. Secara umum, enkripsi dimaksudkan untuk mengamankan data asli dengan mengubah keseluruhan informasi sehingga tidak ada orang yang tidak

berkepentingan memiliki hak akses. Proses ini menghasilkan kunci yang digunakan untuk Dekripsi dan mendapatkan data dalam bentuk aslinya. Terdapat 2 hal yang diperlukan dalam pembuatan *Digital Signature*.



Gambar 3. Signing & Verification

a. *Signing*

Dalam penandatanganan pada dokumen formal akdemik, proses penandatanganan dilakukan oleh user dengan menggunakan kunci pribadi (*private key*) yang disimpan dengan aman oleh si pemberi tandatangan. Algoritma matematika bertindak seperti sandi yang membuat data yang cocok dengan dokumen yang ditandatangani yaitu fungsi *hash* dan mengenkripsi data tersebut. (Gambar 2)

b. Verification

Proses penandatangan digital mensyaratkan bahwa tanda tangan dihasilkan oleh pesan yang di enkripsi menggunakan *private key* (signing) yang kemudian dapat di dekripsi dengan *public key* menggunakan algoritma kriptografi. Tanda tangan pengguna tidak dapat di replikasi tanpa adanya akses dari *private key*.

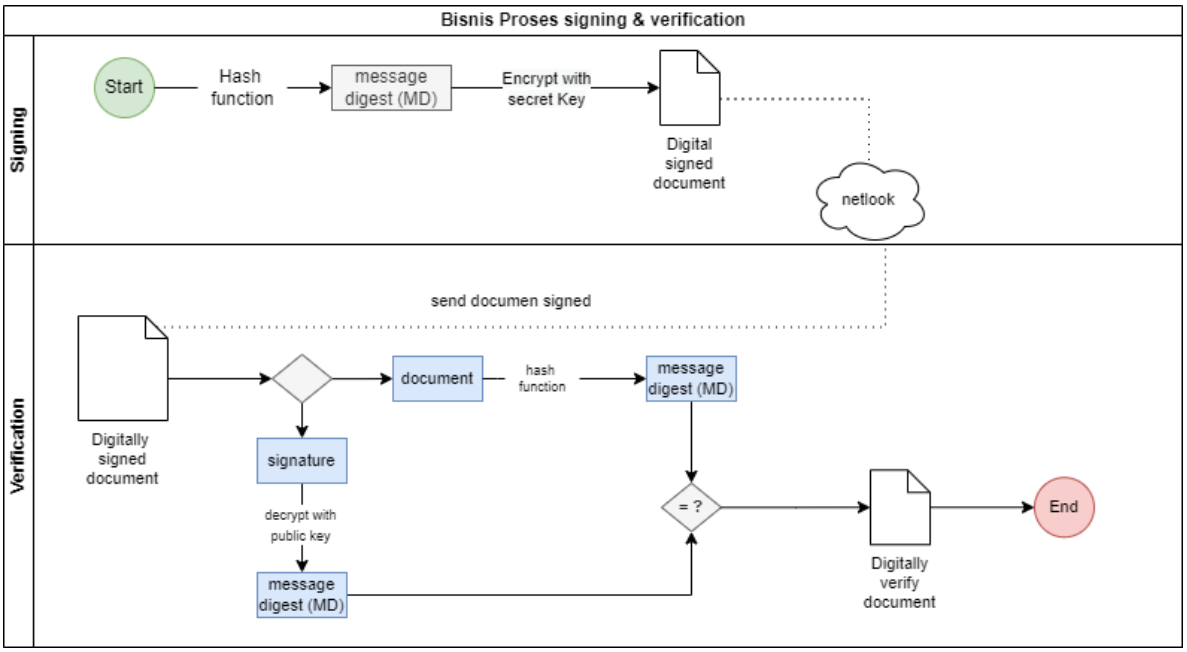
Dalam pemanfaatan Teknologi di era saat ini telah banyak variasi penelitian penggunaan tanda tangan tulisan tangan menggunakan perangkat yang dipakai di pergelangan tangan menerapkan tanda tangan elektronik berkualitas.[5] Cloud yang diusulkan untuk google document dan google sheets menawarkan fungsionalitas sebagai *Software as a service* untuk menjamin keaslian dan integritas data elektronik dan untuk keperluan menyimpan dan memasukkan bermacam versi dokumen digital yang dibuat oleh suatu pihak dalam aplikasi[6] [7].

Kriptografi kunci publik yang paling umum adalah RSA dengan kesulitannya terletak pada masalah dekomposisi bilangan bulat besar dan RSA paling sering digunakan dalam beberapa tahun terakhir (Gambar 2)

2.6.1 Bisnis Proses Signing & Verification

Pada gambar 2.3 dalam melakukan proses *signing & verification*. User harus login terhadap sistem, kemudian akan mengirimkan kunci menggunakan algoritma *hash* kepada penerima,(*receiver*) selanjutnya pengirim (*sender*) akan melakukan enkripsi menggunakan kunci pribadi (*private key)* kemudian kunci pribadi tersebut akan dibuat kedalam dokumen digital dan akan dikirimkan kepada penerima (*receiver*).

Pada tahap selanjutnya penerima (*receiver*) akan mendekripsi kunci public yang diberikan oleh pengirim (*sender)* menggunakan algoritma *hash* yang dikirim oleh pengirim (*sender)* jika hasilnya sama maka dokumen akan diverifikasi.



Gambar 4. Bisnis Proses Signing & Verification

2.6.1.1 Procedure

Adapun prosedur dalam proses *signing* dan *verification* yang harus dipenuhi sebagai berikut:

1. Prosedur penandatanganan dokumen dilakukan oleh pengirim (*sender*) kemudian akan mengirim kemudian akan mengirim algoritma *hash* berupa *bit value* kepada penerima, selanjutnya *bit value* akan di enkripsi menggunakan kunci pribadi kepada dokumen yang ditanda tangani kemudian dikirim kepada penerima dokumen.
2. Prosedur verifikasi dilakukan oleh penerima (*receiver*) yang menerima *bit value* dari pengirim (*sender*) kemudian akan di dekripsi menggunakan kunci public , jika hasilnya sama dengan yang dikirimkan oleh pengirim maka dokumen tersebut dapat di verifikasi.

2.6.1.2 Service Time

Dalam menyelesaikan *signing* dan *verification* pengiriman dilakukan secara *peer-to-peer* (P2P). Waktu yang dibutuhkan sekitar 2 menit untuk *signing* dan 2 menit untuk *verification*. Pada proses *signing* menggunakan aspek keamanan privasi/kerahasiaan yang menjamin bahwa kerahasiaan informasi yang terdapat pada dokumen yang di tanda tangani terkirim secara aman. Integritas yang menjamin dokumen yang telah ditanda

tangani tidak mengalami perubahan. Dokumen yang terkirim secara aman tanpa ada perubahan dari pihak manapun.

Pada proses *verification* menggunakan aspek keamanan identifikasi yang memeriksa identitas yang terdapat pada dokumen yang telah diterima sesuai dengan identitas yang mengirim dokumen (*sender*). Tanda terima yang memberikan informasi kepada pengirim bahwa dokumen yang dikirimkan telah diterima dengan baik.

2.6.2 Digital Signature Scheme

Tanda tangan digital mengacu pada *requirements* dan skema tanda tangan digital yang melibatkan hanya pihak yang bersangkutan yaitu pengirim pesan (*sender*) dan penerima pesan (*receipt*) yang dapat berkomunikasi

Terdapat beberapa kebutuhan Digital signature berdasarkan properti dan serangan yang terjadi pada dokumen digital antara lain sebagai berikut:

1. Tanda tangan harus berupa pola bit yang bergantung pada pesan yang dikirim tertanda.
2. Tanda tangan harus menggunakan beberapa informasi unik untuk pengirim untuk mencegah baik pemalsuan maupun penyangkalan.
3. Harus relatif mudah untuk menghasilkan tanda tangan digital.
4. Harus relatif mudah untuk mengenali dan memverifikasi tanda tangan digital.
5. Secara komputasi tidak mungkin untuk memalsukan tanda tangan digital, baik dengan membangun pesan baru untuk tanda tangan digital yang ada atau dengan membangun tanda tangan digital palsu untuk pesan tertentu.
6. Harus praktis untuk menyimpan salinan tanda tangan digital di penyimpanan

Kerahasiaan diberikan dengan melakukan enkripsi simetris yaitu keseluruhan pesan diikuti tanda tangan oleh private key untuk melakukan fungsi tanda tangan dan fungsi kerahasiaan Di dalam kasus perselisihan dengan adanya pihak ketiga maka harus dapat melihat pesan yang disampaikan dan tanda tangannya Jika tanda tangan dihitung pada pesan terenkripsi, maka pihak ketiga juga membutuhkan akses ke kunci dekripsi untuk membaca pesan asli Validitas skema tergantung pada keamanan *private key* jika pengirim kemudian ingin menolak mengirim pesan tertentu maka pengirim dapat mengklaim.

Terdapat 2 skema *Digital Signatures*

a) ELGamal Digital Signature Scheme

Sebelum memeriksa standar Tanda Tangan Digital NIST, sangat membantu untuk memahami skema tanda tangan ElGamal dan Schnorr. Skema enkripsi ElGamal dirancang untuk mengaktifkan enkripsi dengan kunci publik pengguna dengan dekripsi oleh kunci pribadi pengguna. Skema tanda tangan ElGamal melibatkan penggunaan kunci privat untuk enkripsi dan kunci publik untuk dekripsi (ELGA 84, ELGA85).

Algoritma ElGamal merupakan algoritma enkripsi kunci asimetris untuk kriptografi kunci publik yang didasari kesepakatan kunci Diffie-Hellman. Hal ini diusulkan oleh Taher Elgamal pada tahun 1984. Algoritma ElGamal digunakan pada perangkat lunak GNU Privacy Guard, yang merupakan versi dari PGP, dan kriptosistem lainnya. Digital Signature Algorithm(DSA) merupakan varian dari skema tanda tangan digital ElGamal, dan tidak sama dengan algoritma ElGamal. ElGamal merupakan skema tanda tangan digital berbasis logaritma diskrit. ElGamal terdiri dari tiga komponen, yaitu pembangkit kunci (key generator), algoritma enkripsi, dan algoritma dekripsi.

$$c_2(c_1^x)^{-1} = \frac{m \cdot h^y}{g^{xy}} = \frac{m \cdot g^{xy}}{g^{xy}} = m$$

Jika panjang pesan lebih besar dari pada ukuran G, maka pesan tersebut dapat dipecah ke beberapa bagian dan setiap bagian akan dienkripsi masing-masing.

Aspek Keamanan ElGamal:

ElGamal merupakan contoh sederhana dari sebuah algoritma enkripsi kunci asimetris. Algoritma ini merupakan algoritma yang bergantung pada probabilitas, yang maksudnya adalah sebuah pesan dapat dienkripsi menjadi berbagai kemungkinan *ciphertext*, dengan konsekuensi enkripsi ElGamal biasa menghasilkan 2:1 ekspansi ukuran *plainteks* dan *ciphertext*.

Keamanan ElGamal terletak pada kesulitan dalam memecahkan masalah logaritma diskrit dalam *G*. Akan tetapi, jika masalah logaritma diskrit ini terpecahkan, maka skema ElGamal pun dapat dipecahkan. Kekuatan ElGamal terletak pada asumsi yang disebut Decisional Diffie-Hellman(DDH). Asumsi ini sering kali lebih kuat daripada asumsi logaritma diskrit.

Enkripsi dengan menggunakan skema ElGamal membutuhkan dua eksponensiasi, bagaimanapun kedua eksponensiasi ini bergantung pada pesan dan dapat dihitung terlebih dahulu jika memang diperlukan. Panjang *ciphertext* dua kali panjang plainteks, dan merupakan kekurangan algoritma ini dibandingkan algoritma lainnya. Dekripsi hanya membutuhkan satu eksponensiasi, tidak seperti pada RSA. Dekripsi pada ElGamal tidak dapat dipercepat dengan menggunakan *Chinese Remainder Theorem*.

b) Schnorr Digital Signature Scheme

Skema Schnorr meminimalkan jumlah komputasi yang bergantung pada pesan yang diperlukan untuk menghasilkan tanda tangan.Untuk pembuatan tanda tangan tidak tergantung pada pesannya dan dapat dilakukan selama waktu idle prosesor. Bagian yang bergantung pada pesan dari generasi tanda tangan membutuhkan mengalikan bilangan bulat -bit dengan -bit bilangan bulat.

Koblitz menggunakan perkalian skalar tipikal. Hasil dari penelitian ini adalah kedua sistem bekerja dengan benar. Namun, skema tanda tangan Schnorr pada kurva Koblitz menggunakan *Double-and Add* berkinerja lebih baik dalam efisiensi waktu daripada skema tanda tangan Schnorr pada kurva Koblitz menggunakan perkalian skalar tipikal.

Skema tanda tangan Schnorr pada kurva Koblitz menggunakan *Double-and Add* berkinerja lebih baik dalam efisiensi waktu daripada skema tanda tangan Schnorr pada kurva Koblitz menggunakan perkalian skalar tipikal. Pemilihan parameter pada schnor terletak pada setiap pengguna skema tanda tangan menyetujui *group G* dengan pembangkit *g* dari bilangan prima *q* di mana masalah logaritma diskritnya sulit dipecahkan.

Setiap pengguna lalu menyetujui fungsi *hash* H. Pembangkitan kunci pada schnor dapat dilihat pada Langkah berikut ini :

1. Pilih kunci privat x di mana $0 < x < q$.
2. Kunci publik adalah y di mana $y = g^{-x}$
3. Pilih bilangan prima p sepanjang L bit, di mana $p = qz + 1$, dengan z adalah *integer* dan $512 \leq L \leq 1024$ dan L dapat dibagi dengan 64.
4. Plih h di mana $1 < h < p-1$ dan $g = h^2 \bmod p > 1$.
5. Pilih x mengunakan metode acak tertentu, di mana $0 < x < q$.
6. Hitung $y = g^x \bmod p$
7. Kunci publik adalah $\{p, q, g, y\}$, dan kunci privatnya adalah x.

   Bila diperlukan $\{p, q, g\}$ dapat digunakan bersama oleh pengguna yang berbeda dalam suatu sistem. Proses pemberian tanda tangan pada Schnor adalah sebagai berikut : Untuk menandatangani pesan m, dilakukan :

1. Pilih bilangan k secara acak dengan memenuhi $0 < k < q$
2. Diberi persamaan $r = g^k$
3. Diberi persamaan $e = H(M \,||\, r)$
4. Diberi persamaan $s = (k + xe) \bmod q$

Tanda tangan digital merupakan pasangan (e, s), dengan $0 \leq e < q$ dan $0 \leq s < q$, jika *group* Schnorr digunakan dan $q < 2^{160}$, yang berarti tanda tangan ini cukup dalam 40 *byte*.

2.6.3 RSA Algorithm (Rivest-Shamir-Adleman Algorithm)

RSA (Rivest-Shamir-Adleman) merupakan *block cipher* dimana *plaintext* dan *ciphertext* bilangan bulat antara 0 dan n - 1 untuk beberapa n. Ukuran tipikal untuk n adalah 1024 bit, atau 309 digit desimal. Artinya, n kurang dari 2^{1024}. Kami memeriksa RSA di bagian ini di beberapa detail, dimulai dengan penjelasan tentang algoritma. Kemudian kami memeriksa beberapa dari implikasi komputasi dan cryptanalytical RSA.

RSA merupakan algoritma yang baik untuk digunakan dalam proses tanda tangan digital dan enkripsi. RSA digunakan secara luas dalam keamanan data sampai saat ini jika diberikan

kunci dengan panjang bit yang cukup besar. RSA melibatkan dua buah kunci, yaitu kunci publik (*public* key) dan kunci pribadi (*private key*). Kunci publik dapat diketahui siapapun dan berguna untuk mengenkripsi pesan. Pesan yang dienkripsi dengan menggunakan kunci publik tersebut hanya bisa didekripsi oleh pihak yang memiliki kunci pribadi (*private key*).

Kunci publik dan privat untuk RSA dapat dibangkitkan dengan cara sebagai berikut:

1. Pilih dua buah bilangan prima besar secara acak, p dan q
2. Hitung $n = pq$
3. Hitung $phi = (p - 1)(q - 1)$
4. Pilih bilangan bulat $e > 1$ yang relative prima terhadap phi
5. Hitung d dengan memenuhi persamaan $de \equiv 1 \bmod phi$ atau $de = 1 + k * phi$

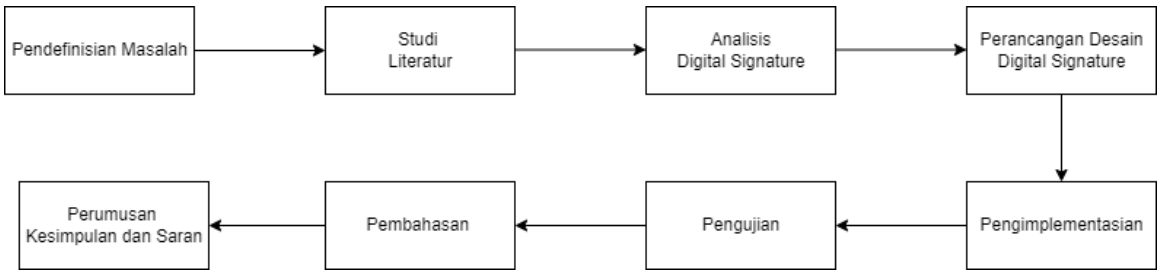
Kunci publik adalah pasangan (e, n) sedangkan kunci privat adalah pasangan (d, n).

BAB 3 ANALISIS

Bab ini akan memaparkan analisis metode yang digunakan oleh penulis dalam melakukan penelitian serta analisis fungsional dari *Digital Signatures* yang akan dibangun.

3.1 Metode Penelitian

Langkah-langkah yang dilakukan dalam metodologi penelitian pengerjaan Tugas Akhir ini dapat dilihat melalui Gambar 3.1 berikut :



Gambar 5. Tahapan Penelitian

Berikut merupakan penjelasan dari metodologi penelitian yang dilakukan pada Tugas Akhir dengan topik *Digital Signature*

1. Pendefenisian Masalah

   Pada tahap ini akan dilakukan pendefenisian masalah yang terdiri dari latar belakang, pertanyaan penelitian, tujuan penelitian, dan ruang lingkup yang akan dilakukan selama tahap penelitian
2. Studi Literatur

   Pada tahap ini akan dilakukan studi literatur berdasarkan landasan teori yang merupakan referensi dari berbagao sumber seperti buku, jurnal penelitian, atau beberapa studi kepustakaan lainnya yang berkaitan dengan penelitian
3. Analisis Digital Signature

   Pada tahap ini akan dilakukan analisis Digital Signature untuk memperoleh informasi mengenai penelitian yang akan dilakukan

4. Perancangan Desain Digital Signature

   Pada tahap ini dilakukan perancangan terhadap desain dokumen signing dan *verification* with *Digital Signature*

5. Pengimplementasian

   Pada tahap ini dilakukan implementasi dengan menghubungkan data mahasiswa yang terdapat pada *Campus Information System* (CIS) IT Del menggunakan API agar terhubung dengan sistem *Digital Signatures*

6. Pengujian

   Setelah tahap implementasi digital signature telah dikembangkan, maka akan dilanjutkan dengan tahap pengujian (Testing) terhadap sistem yang telah dikembangkan untuk menguji proses *Signing Verification* dokumen formal secara otomatis

7. Pembahasan

   Pada tahap ini akan dilakukan pembahasan terhadap hasil implementasi dan pengujian yang telah diperoleh

8. Perumusan kesimpulan dan saran

   Pada tahap ini dilakukan perumusan kesimpulan terkait hasil penelitian yang sudah dilakukan serta saran yang akan menjadi penelitian selanjutnya.

3.2 Jadwal Penelitian

Pada sub bab ini akan dijelaskan tentang jadwal penelitian yang akan dijelaskan pada tabel jadwal penelitian antara lain sebagai berikut :

Daftar kegiatan	**Lingkup (Bab)**	**Week**													
		1	**2**	**3**	**4**	**5**	**6**	**7**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**17**
Pengumpulan masalah																
Penentuan topik &																

diskusi topik																
Pengerjaan Proposal	Bab 1															
Pengumpulan proposal tahap 1	Bab 1															
Pengerjaan proposal	Bab 1, 2, 3															
Pengumpulan artefak	Bab 1, 2,3															
Pelaksanaan Seminar Proposal	Bab 1,2,3															

Tabel 2. Jadwal Penelitian

References

[1] B. Schneier, Applied Cryptography, Canada: John Wiley and Sons, 1996.

[2] Menezes, Handbook Applied Cryptography, Florida: CRC Press LLC, 1997.

[3] "Electronic Signatures and Infrastructures (ESI) Protocols for remote digital signature creation," *ETSI TS 119 432*, vol. 1.1.1, no. 2019-03, p. 10, 2019.

[4] B. Y. E. ALONA, "Handwritten Signature Verification Using Wrist-Worn Devices," *ACM DL*, vol. 2, 2018.

[5] M.-L. Pura, "Implementing Cloud Qualified Electronic Signatures for Documents using Available Cryptographic," *IEEE* , p. 1, 2020.

[6] I. Aciobanitei, "Qualified Electronic Signature SaaS Solution for Google Docs & Google Sheets Documents," *IEEE*, 2020.

[7] W. STALLINGS, CRYPTOGRAPHY AND NETWORK SECURITY, France, 2011.

[8] N. N. Xiong, "A Privacy Preserving Handwritten Signature Verification Method Using Combinational Features and Secure KNN," *IEEE Access*, vol. 6, no. 2018, p. 1, 2018.