

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Perkembangan teknologi khususnya teknologi komputer telah mengalami kemajuan yang sangat pesat. Perkembangan teknologi tersebut tidak lepas dari peran manusia yang setiap saat harus memperbaiki dan mencari inovasi baru agar teknologi tersebut dapat digunakan untuk membantu pekerjaan manusia.

Kita tahu bahwa teknologi komputer telah banyak digunakan diberbagai organisasi, baik organisasi besar maupun kecil. Teknologi komputer dimanfaatkan sebagai alat bantu untuk mempermudah pekerjaan dari perusahaan atau organisasi tersebut. Namun pada kenyataannya masih banyak perusahaan atau organisasi yang belum menggunakan teknologi komputer sebagai alat bantu pekerjaan, seperti sistem pengarsipan yang penulis jadikan kasus dalam pembuatan laporan skripsi ini yang masih menggunakan cara manual pada kantor Desa Buntu Bedimbar. Kearsipan merupakan bagian pekerjaan dari suatu institusi yang sangat penting. Informasi tertulis yang tepat harus tersedia apabila diperlukan agar suatu institusi dapat memberikan pelayanan yang efektif.

Kantor Desa Buntu Bedimbar merupakan salah satu instansi pemerintah yang berada di kecamatan Tanjung Morawa Kabupaten Deli Serdang. Dalam proses pelayanan administrasi Kantor Desa Buntu Bedimbar tidak lepas dari sebuah surat. Pengarsipan surat masuk dan surat keluar di Desa Buntu Bedimbar masih dilakukan secara manual. Surat dan lembar dokumen tersebut disimpan di box file berdasarkan tahun pembuatannya. Sistem manual seperti itu menyulitkan petugas

ketika akan mencari surat yang di inginkan karena harus mencari data satu persatu, sehingga hal tersebut memerlukan waktu yang cukup lama.

Untuk mengatasi kondisi ini maka penulis tertarik untuk membuat sebuah aplikasi Elektronik Arsip (E-Arsip) Surat berbasis web untuk memudahkan perangkat desa dalam pelayanan administrasi masyarakat menjadi lebih efisien dan efektif. Penulis menggunakan Algoritma *RIVEST SHAMIR ADLEMAN* (RSA) yaitu algoritma yang menggunakan sistem kriptografi kunci publik yang dapat memberikan jaminan keamanan pada jalur transmisi distribusi kunci.

Berdasarkan Keadaan Desa Buntu Bedimbar yang telah diuraikan di atas maka peneliti memutuskan untuk menyelesaikan permasalahan yang ada dengan cara membangun sebuah aplikasi berbasis web yang berjudul “**PENGAMANAN ELEKTRONIK ARSIP (E-ARSIP) PADA KANTOR DESA BUNTU BEDIMBAR MENGGUNAKAN ALGORITMA RIVEST SHAMIR ADLEMAN (RSA)**”. Dengan adanya sistem ini diharapkan dapat membantu petugas dalam pengarsipan surat pada Kantor Desa Buntu Bedimbar.

## **1.2. Rumusan Masalah**

Berdasarkan pembahasan pada latar belakang yang telah dijabarkan, maka dapat diambil suatu rumusan masalah sebagai berikut :

1. Bagaimana pengamanan Aplikasi E-Arsip surat pada Desa Buntu Bedimbar?
2. Bagaimana menerapkan Algoritma *Rivest Shamir Adleman* (RSA) pada aplikasi E-Arsip surat pada Desa Buntu Bedimbar?

### 1.3. Batasan Masalah

pembahasan pada latar belakang yang telah dijabarkan, maka dapat diambil suatu rumusan masalah sebagai berikut:

1. Membuat pengamanan sistem arsip surat pada kantor Desa Buntu Bedimbar dengan menggunakan Algoritma *Rivest Shamir Adleman*.
2. Penelitian lebih di fokuskan pada bagian sistem pengamanan surat dengan menggunakan Algoritma *Rivest Shamir Adleman* agar mempermudah pegawai dalam pengarsipan data surat pada sistem.
3. Perancangan aplikasi ini dibangun dengan menggunakan bahasa pemrograman PHP dan database MySQL.
4. Aplikasi yang dirancang berbasis web.
5. Format data surat pada sistem berjenis PDF.

### 1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Untuk mengamankan Aplikasi E-arsip surat pada Kantor Desa Buntu Bedimbar.
2. Untuk menerapkan Algoritma *Rivest Shamer Adleman* pada Aplikasi E-Arsip Surat pada Desa Buntu Bedimbar.
3. Membuat sistem pengamanan yang mudah di akses oleh pegawai pada Kantor Desa Buntu Bedimbar.

### 1.5. Manfaat Penelitian

Penulis mengharapkan dari penelitian yang dilakukan dapat memberikan efek yang positif dan memberikan manfaat. Berikut manfaat yang ingin dicapai dalam penelitian ini :

1. Memberikan kemudahan bagi petugas dalam pengarsipan Surat berbasis web.
2. Mengenalkan kepada warga Desa akan kemajuan dan perkembangan teknologi.
3. Menghindari hilangnya data penduduk yang masih tersimpan secara manual.
4. Sebagai tempat atau media yang dapat mempermudah perangkat desa Buntu bedimbar dalam melakukan pelayanan masyarakat seperti surat menyurat secara efektif dan efisien.
5. Dapat mempermudah pegawai dalam mencari data kependudukan di desa Buntu Bedimbar.

#### **1.6. Sistematika penulisan**

Untuk mempermudah dalam penyusunan dan memahami skripsi maka penulis menyajikan sistematika penulisan sebagai berikut :

#### **BAB I : PENDAHULUAN**

Pada bab ini akan dijelaskan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, sistematika penulisan.

#### **BAB II : TINJAUAN PUSTAKA**

Pada bab ini memuat tentang materi-materi pendukung dalam penyusunan skripsi, mulai dari teori-teori yang digunakan, konsep-konsep yang akan diterapkan dalam menyelesaikan permasalahan yang penulis teliti dalam penelitian ini.

**BAB III : METODE PENELITIAN**

Pada bab ini memuat mengenai metode yang penulis gunakan dalam menyelesaikan rumusan masalah, tahap-tahap mengenai teknik pengolahan data, perancangan aplikasi, dan pembuatan aplikasi.

**BAB IV : HASIL DAN PEMBAHASAN**

Pada bab ini memuat hasil-hasil yang didapat dari penelitian serta melakukan pembahasan atas hasil yang diperoleh. Kesulitan yang ditemukan saat perancangan dan pembuatan aplikasi.

**BAB V : KESIMPULAN DAN SARAN**

Pada bab ini membuat kesimpulan dan saran penulis atas penelitian yang dilakukan.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Pengertian Sistem**

Sistem adalah merupakan satu kesatuan data yang terhubung dan terorganisir secara prosedural. (Achmad Fikri dan Indra: 2020).

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersamasama untuk melakukan suatu kegiatan tertentu untuk mencapai tujuan tertentu.

Berdasarkan pengertian diatas dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai dalam melaksanakan suatu kegiatan pokok perusahaan.

#### **2.2 Aplikasi**

Aplikasi adalah suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia.

a. Menurut Kamus Besar Bahasa Indonesia Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu.

b. Menurut Rachmad Hakim S, adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur *windows* dan permainan dan sebagainya.

c. Menurut Harip Santoso, adalah suatu kelompok file (*form, class, report*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi *payroll*, aplikasi *fixed asset*, dan lain-lain.

Dari uraian di atas maka dapat disimpulkan bahwa aplikasi adalah perangkat lunak yang diciptakan dengan berbagai komponen atribut yang sesuai dengan pengguna agar dapat membantu pengguna dalam mengolah setiap data agar menghasilkan input dan output.

### **2.3. Arsip**

Arsip merupakan salah satu sumber informasi penting yang dapat menunjang proses kegiatan administrasi maupun birokrasi. Sebagai rekaman informasi dari seluruh aktivitas organisasi, arsip berfungsi sebagai pusat ingatan, alat bantu pengambilan keputusan, bukti eksistensi organisasi dan untuk kepentingan organisasi yang lain. (fathurrahman, 2018)

Arsip juga merupakan naskah tertulis yang didalamnya memuat keterangan-keterangan penting. (Shella Ayurindah, 2022)

Berdasarkan pengertian diatas dapat disimpulkan bahwa Arsip adalah kumpulan dokumen, rekaman, atau benda lain yang memiliki nilai historis, administratif, atau legal. Arsip digunakan untuk menyimpan dan melestarikan informasi yang penting untuk referensi di masa depan. Pengelolaan arsip yang baik

memastikan bahwa informasi tersebut tetap dapat diakses, dicari, dan dipelajari oleh generasi yang akan datang.

#### **2.4. Website**

*Website* merupakan sebuah kumpulan halaman-halaman web beserta file-file pendukungnya, seperti file gambar, video, dan file digital lainnya yang disimpan pada sebuah web server yang umumnya dapat diakses melalui internet. (Yudin W: 2020).

*Website* merupakan salah satu sumber daya teknologi yang berkembang pesat. Saat ini, informasi web didistribusikan lebih dekat dan mudah, yang memungkinkan suatu teks, gambar ataupun objek yang lain menjadi acuan dasar untuk membuka halaman halaman web yang lain. (A. Nurkholis: 2022).

#### **2.5. Pengertian Database**

*Database* adalah kumpulan data yang terorganisir yang disimpan dan diakses secara elektronik dari sistem komputer. *Database* sangat kompleks bahkan semakin hari semakin dikembangkan. Semakin banyak data pada database maka semakin banyak informasi yang harus diamankan. (Tomy dan Ahmad Taufik: 2020).

#### **2.6. Teknologi Yang Digunakan**

##### **2.6.1. HTML (*Hypertext Markup Language*)**

HTML merupakan protokol yang digunakan untuk mentransfer data atau dokumen dari web server ke browser (microsoft Edge, Mozilla firefox, Google Chrome, dll.) (Husni Tahmrin: 2021).



### **2.6.2. PHP (*Hypertext Preprocessor*)**

PHP adalah komponen PHP adalah komponen dari PHP Hypertext Preprocessor. PHP adalah salah satu jenis bahasa scripting yang digunakan untuk membangun aplikasi untuk web dan menghubungkannya ke server. PHP adalah bahasa yang menggunakan add on HTML untuk membangun aplikasi yang menggunakan data dan data secara maksimal. Sebagian data yang dikirim keluar akan diproses sendiri oleh server, dan ada juga data yang akan dikirimkan ke browser. Skrip khusus bahasa diinstal di server dan dijalankan di klien. Klien, antarmuka pengguna browser, akan mereplikasi masalah tersebut. PHP adalah bahasa pemrograman yang berkoordinasi dengan HTML, dieksekusi di server, dan dapat digunakan untuk membangun berbagai situs web, termasuk Dynamic Server Pages (ASP) dan Java Server Pages (JSP). (Ira Murni, Atika, DKK: 2023)

### **2.6.3. XAMPP**

XAMPP adalah perangkat lunak *open source*, yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri beberapa program antara lain : Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. (Ertie Nur Hartiwati: 2022).

### **2.7. *Unifed Modelling Language (UML)***




UML (*Unified Modeling Language*) merupakan bahasa pemodelan perangkat lunak yang telah distandarisasi sebagai media penulisan untuk cetak biru (*blueprints*) perangkat lunak. UML dapat digunakan untuk visualisasi, spesifikasi,

konstruksi dan beberapa dokumentasi sistem yang ada dalam perangkat lunak. UML digunakan untuk membantu *programmer* atau *developer* dalam membuat dan membangun *software* atau perangkat lunak (Sumiati et al., 2021). UML dapat dijelaskan sebagai suatu bahasa yang dipergunakan untuk menjelaskan persyaratan yang ada, membuat Analisa dan desain membuat gambaran arsitektur dalam *Object Oriented Programming* UML mempunyai banyak diagram diantaranya sebagai berikut:

#### A. *Use Case Diagram*

*Use Case Diagram* adalah sebuah diagram yang wajib dikerjakan pertama kali pada saat pemodelan perangkat lunak berorientasi objek dibuat. *Use Case* menggambarkan suatu hubungan timbal balik antara satu actor atau lebih dari sistem yang akan dibangun. *Use Case Diagram* juga berfungsi untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use Case Diagram* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut adalah simbol-simbol yang ada pada dalam *use case diagram*:



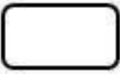

Tabel 2. 1 Simbol-Simbol Use Case Diagram

| Simbol  | Nama            | Keterangan   |
|---|-----------------|--|
|  | Aktor           | Adalah Penggunaan dari sistem.<br>Nama aktor diisi dengan kata benda.  |
|  | <i>Use Case</i> | Adalah pekerjaan yang dikerjakan aktor. Nama use case menggunakan kata kerja.  |
|  | Asosiasi        | Keterkaitan antar aktor dan use case   |
| <<use>>   | <i>Include</i>  | Keterkaitan antara use case dan use case, include menjelaskan jika sebelum suatu pekerjaan dikerjakan maka wajib melakukan pekerjaan lain terlebih dahulu.                 |
| <<extends>>   | <i>Extends</i>  | Keterkaitan antara use case dan use case, extends menjelaskan jika pekerjaan yang dikerjakan tidak sesuai atau memiliki suatu kondisi khusus, maka kerjakan pekerjaan itu. |

## B. Activity Diagram

*Activity Diagram* mendeskripsikan aliran kerja (*workflow*) dengan cara menggambarkan kegiatan dari sebuah sistem. *Activity diagram* mendeskripsikan aliran kerja dengan penggambaran atau kegiatan dari suatu menu yang terdapat pada software. *Activity diagram* mengilustrasikan kegiatan kerja sistem bukan kegiatan yang dapat dilakukan oleh aktor, jadi kesimpulannya *activity diagram* adalah kegiatan yang bisa dikerjakan oleh sistem. Simbol-simbol dari *activity diagram* adalah sebagai berikut:

Tabel 2. 2 Simbol-Simbol Activity Diagram



| <b>Nama</b>           | <b>Simbol</b>   | <b>Keterangan</b>  |
|-----------------------|---|--|
| <i>Innitial State</i> |   | Menggambarkan awal suatu aktivitas   |
| <i>Final State</i>    |  | Menggambarkan berakhirnya aktivitas  |
| <i>Activity</i>       |  | Menggambarkan kegiatan yang dikerjakan oleh sistem                                 |
| <i>Decision</i>       |  | Menggambarkan pilihan serta mengambil keputusan dari pilihan yang telah disediakan |
| <i>Control Flow</i>   |   | Menggambarkan arah dari kegiatan sebuah sistem.                                    |



### C. ERD (*Entity Relationship Diagram*)

*ERD (entity Relationship Diagram)* merupakan sebuah model Teknik pendekatan dimana berfungsi untuk menjelaskan atau mendeskripsikan dengan cara memberi gambaran keterkaitan sebuah model. Gambaran keterkaitan itu dapat dikatakan bagian yang paling utama dari ERD yaitu memperlihatkan objek data (*Entity*) serta keterkaitan (*Relationship*) yang terdapat didalam objek data berikutnya.

ERD merupakan satu alat pemodelan data dan akan membantu mengatur data dalam sebuah sistem ke dalam entitas-entitas dan menetapkan keterkaitan antar entitas. Proses ini tentunya memungkinkan untuk menganalisa serta menciptakan struktur *database* agar bisa ditempatkan di dalam penyimpanan serta dapat diambil secara efisien. Simbol-simbol yang terdapat di ERD adalah sebagai berikut:


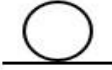




Tabel 2. 3 Simbol-Simbol ERD (*Entity Relationship Diagram*)

| Nama           | Keterangan  | Simbol  |
|----------------|---|---|
| <i>Entitas</i> | Sesuatu yang nyata atau abstrak dan memiliki ciri khas sebagai tempat untuk penyimpanan data. |  |
| <i>Relasi</i>  | Keterkaitan alami yang dapat tercipta antara satu entitas atau lebih.                         |  |

|                |  |   |
|----------------|--|---|
| <i>Atribut</i> | Memiliki ciri umum seluruh atau sebagian besar instansi pada entitas tertentu.                               |  |
| <i>Garis</i>   | Merupakan sebuah garis yang menghubungkan atribut dan himpunan entitas serta himpunan entitas dengan relasi. |  |

#### D. Sequence Diagram

Sequence diagram merupakan diagram yang menjelaskan interaksi objek berdasarkan urutan waktu. Sequence diagram dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu, seperti yang tertera pada Use Case Diagram.








| NO | GAMBAR  | NAMA  | KETERANGAN  |
|----|---|---|---|
| 1  |  | <i>Actor</i>                                | Menggambarkan orang yang sedang berinteraksi dengan sistem. |
| 2  |  | <i>Entity Class</i>                         | Menggambarkan hubungan yang akan dilakukan                  |
| 3  |  | <i>Boundary Class</i>                       | Menggambarkan sebuah gambaran dari foem                     |
| 4  |  | <i>Control Class</i>                        | Menggambarkan penghubung antara boundary dengan tabel       |
| 5  |  | <i>A focus of Control &amp; A Life Line</i> | Menggambarkan tempat mulai dan berakhirnya message          |
| 6  |  | <i>A message</i>                            | Menggambarkan Pengiriman Pesan                              |

Gambar 2. 1 Simbol-Simbol Sequence Diagram

### E. Class Diagram

Class diagram atau diagram kelas merupakan suatu diagram yang digunakan untuk menampilkan kelas-kelas berupa paket-paket untuk memenuhi salah satu kebutuhan paket yang akan digunakan nantinya. Namun, pada *class diagram* sesuai modelnya dibagi menjadi 2 bagian, *class diagram* yang pertama merupakan penjabaran dari domain model yang merupakan abstraksi dari basis data. *Class diagram* yang kedua merupakan bagian dari modul program MVC Pattern (Model View Controller), di mana terdapat *class boundary* sebagai *class interface*, *class control* sebagai tempat ditemukannya algoritma, dan *class entity* sebagai table dalam basis data dan *query* program.

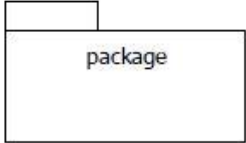
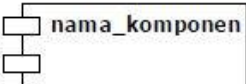

#### SIMBOL CLASS DIAGRAM

| NO | GAMBAR  | NAMA                    | KETERANGAN   |
|----|---|-------------------------|--|
| 1  |  | <i>Generalization</i>   | Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).                    |
| 2  |  | <i>Nary Association</i> | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.  |
| 3  |  | <i>Class</i>            | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.  |
| 4  |  | <i>Collaboration</i>    | <u>Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor</u>   |
| 5  |  | <i>Realization</i>      | Operasi yang benar-benar dilakukan oleh suatu objek.   |
| 6  |  | <i>Dependency</i>       | <u>Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri</u> |
| 7  |  | <i>Association</i>      | Apa yang menghubungkan antara objek satu dengan objek lainnya  |

Gambar 2. 2 Simbol-Simbol Class Diagram

## F. *Component Diagram*

Component diagram merupakan untuk menggambarkan software pada suatu sistem. *Component diagram* merupakan penerapan pada piranti lunak atau *software* dari satu *class* maupun lebih, dan biasanya berupa *file data*, *source code*, *exe*, *table*, *dokumen*, atau yang lainnya.

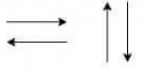








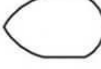
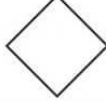
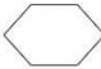
| Simbol  | Deskripsi   |
|---|---|
| Package<br>                              | package merupakan sebuah bungkusan dari satu atau lebih komponen  |
| Komponen<br>                           | Komponen sistem   |
| Kebergantungan / <i>dependency</i><br> | Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai                             |
| Antarmuka / <i>interface</i>  | sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen |

Gambar 2.3 Simbol-Simbol Component Diagram



## 2.8. Flowchart

Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program. Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah. Flowchart berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus. Fungsi utama dari flowchart adalah memberi gambaran jalannya sebuah program dari satu proses ke proses lainnya.

|   |  |  |   |
|---|--|--|---|
|    | <p><b>Flow</b></p> <p>Simbol yang digunakan untuk menggabungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan Connecting Line.</p> |    | <p><b>Input/output</b></p> <p>Simbol yang menyatakan proses input atau output tanpa tergantung peralatan.</p>                             |
|  | <p><b>On-Page Reference</b></p> <p>Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang sama.</p>  |  | <p><b>Manual Operation</b></p> <p>Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.</p>                             |
|  | <p><b>Off-Page Reference</b></p> <p>Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang berbeda.</p>  |  | <p><b>Document</b></p> <p>Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak.</p> |
|  | <p><b>Terminator</b></p> <p>Simbol yang menyatakan awal atau akhir suatu program.</p>  |  | <p><b>Predefine Proses</b></p> <p>Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedur.</p>                                  |
|  | <p><b>Process</b></p> <p>Simbol yang menyatakan suatu proses yang dilakukan komputer.</p>  |  | <p><b>Display</b></p> <p>Simbol yang menyatakan peralatan output yang digunakan.</p>  |
|  | <p><b>Decision</b></p> <p>Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya dan tidak.</p>                           |  | <p><b>Preparation</b></p> <p>Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.</p>       |

Gambar 2.4 Simbol-Simbol *Flowchart*

## 2.9. Algoritma Rivest Shamir Adleman (RSA)

Algoritma RSA merupakan algoritma yang menggunakan sistem kriptografi kunci publik yang dapat memberikan jaminan keamanan pada jalur transmisi distribusi kunci serta mendukung tanda tangan digital yang memverifikasi pesan yang diterima merupakan pesan asli yang dikirim oleh pengirim pesan. (Ida Bagus Gede dan Ida Bagus Ary: 2022). RSA dikatakan aman, karena sulitnya memfaktorkan bilangan  $n$ , dimana  $n = p \cdot q$ ,  $p$  dan  $q$  adalah bilangan prima yang sangat besar. RSA membangkitkan kunci privat dan kunci publik-nya, dengan langkah-langkah sebagai berikut.

- a. Membangkitkan nilai  $p$  dan  $q$  secara sembarang, dimana  $p$  dan  $q$  ini adalah bilangan prima yang besar.
- b. Menghitung  $n = p \cdot q$ .
- c. Menghitung  $\varphi(n)$  yaitu  $\varphi(n) = (p - 1)(q - 1)$ .
- d. Memilih kunci public  $e$  yang relative prima terhadap  $\varphi(n)$ .  $GCD(\varphi(n), e) = 1$
- e. Membangkitkan kunci private  $d$  menggunakan rumus
 
$$e \cdot d = k \cdot \varphi(n) + 1 \quad (3)$$
- f. Dihasilkanlah pasangan kunci public  $(e, n)$  dan kunci private  $(d, n)$ .

Sedangkan untuk proses enkripsi dan dekripsi, RSA melakukan langkah-langkah seperti berikut:

- a) Enkripsi suatu pesan ( $m$ ) menggunakan kunci public  $(e, n)$   $c = m \cdot e \text{ mod } n$  (4)
- b) Dekripsi suatu cipherteks
- c) menggunakan kunci privat  $(d, n)$

$$m = c^d \text{ mod } n$$

### **2.9.1 Kelebihan Algoritma RSA**

Adapun kelebihanya yaitu:

1. RSA memiliki tingkat keamanan yang tinggi, terutama ketika digunakan dengan kunci yang cukup panjang.
2. RSA menggunakan pendekatan kriptografi kunci publik, dimana ada kunci publik untuk mengenkripsi data dan kunci private untuk mendeksripsi data.
3. RSA sangat fleksibilitas yang dapat digunakan untuk enkripsi data, tanda tangan digital, dan protokol keamanan lainnya.

### **2.9.2 Kekurangan Algoritma RSA**

Adapun kekurangannya yaitu :

1. RSA cenderung memerlukan komputasi yang intensif, terutama saat menggunakan kunci yang panjang. Proses enkripsi dan dekripsi RSA bisa menjadi lambat, terutama jika ukuran kunci yang digunakan sangat besar.
2. RSA memerlukan manajemen kunci yang hati-hati, terutama ketika digunakan dalam skenario di mana ada banyak pihak yang terlibat.
3. Untuk tingkat keamanan yang tinggi, RSA membutuhkan ukuran kunci yang besar. Kunci yang lebih besar berarti memerlukan lebih banyak memori untuk penyimpanan dan memerlukan lebih banyak waktu untuk menghitung operasi kriptografi.