Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)
Jakarta – Indonesia, 06 September 2022

IMPLEMENTASI KRIPTOGRAFI MENGGUNAKAN METODE ALGORITMA RSA PADA APLIKASI PENGAMANAN DATA BERBASIS JAVA DESKTOP UNTUK UD TIRTA SOEPER TELOER

Muhammad Zainal Solihin[1*], Krisna Adiyarta M.[2]

[1,2]Fakultas Teknologi Informasi, Teknik Informatika, Universitas Budi Luhur, Jakarta Selatan, Indonesia

Email: [1*]zainal0399@gmail.com, [2]krisna.adiyarta@budiluhur.ac.id
(* : corresponding author)

Abstrak-Kriptografi merupakan suatu teknik menyembunyikan dokumen dimana dokumen tersebut hanya dapat diketahui oleh orang tertentu dimana dokumen itu sering tersebut dengan enkripsi. Saat ini enkripsi sudah banyak dikembangkan salah satunya adalah metode *RSA* (Rivest Shamir Adleman) yang menggunakan dua kunci *public* dan kunci pribadi, dimana kunci tersebut dapat diatur dimana semakin kuat untuk dipecahkan karena sulitnya memfaktorkan dua bilangan yang sangat besar dan itu dianggap aman atau tidak, maka dari itu dalam penelitian ini akan dibuat keamanan yang lebih baik lagi dengan memadukannya dengan (*plaintext*) dimana pada kunci publik yang diatur diubah terlebih dahulu atau enkripsi dengan (*ciphertext)* baru bisa dipecahkan kembali dengan algoritma *RSA* (*Rivest Shamir Adleman*). Keamanan data merupakan aspek yang sangat penting dalam dunia informasi sekarang ini. Terlebih jika informasi yang disimpan dan dikirim dalam dokumen bersifat sangat rahasia, dimana pihak luar yang tidak diperbolehkan untuk mengetahuinya isi data tersebut, karena jika data itu penting ini sehingga bocor kepihak yang tidak bertanggung jawab maka akan menimbulkan kerugian bagi pihak perusahaan yang datanya dicuri maupun manipulasi data tersebut. Beberapa data yang sangat penting bagi perusahaan tersebut, data perusahaan, absensi karyawan dan data masuk penjualan, dan lain-lain yang dianggap penting dan perlu diamankan oleh pihak UD Tirta Soeper Teloer. Salah satu cara yang dapat digunakan yaitu dengan mengenkripsikan data tersebut melalui aplikasi kriptografi untuk menangani permasalahan yang ada pada perusahaan. Maka dari itu diperlukan aplikasi pengamanan data yang dapat dimiliki pada perusahaan tersebut. Dengan menerapkan algoritma kriptografi *RSA* (*Rivest Shamir Adleman)*, aplikasi ini dibangun dengan bahasa pemrograman *Java* berbasis *desktop*. Metode *RSA* digunakan pada aplikasi ini untuk mengenkripsi pesan rahasia yang berupa dokumen agar kerahasiaan data tersebut semakin kuat dan aman. Berdasarkan implementasi dan uji coba program aplikasi enkripsi dan dekripsi yang menggunakan algoritma *RSA* tersebut menjadi *plaintext* agar dapat dibaca kembali, mampu mengamankan dokumen atau data dengan baik.

Kata Kunci: *kriptografi, RSA, asimetris, enkripsi, deskripsi*

IMPLEMENTATION OF CRYPTOGRAPHY USING RSA ALGORITHM METHOD IN DATA SECURITY APPLICATION BASED ON JAVA DESKTOP FOR UD TIRTA SOEPER TELOER

Abstract-*Cryptography is a technique of hiding documents where the document can only be known by certain people where the document is often encrypted. Currently, encryption has been widely developed, one of which is the RSA (Rivest Shamir Adleman) method which uses two public keys and a private key, where the key can be set which is getting stronger to solve because of the difficulty of factoring two very large numbers and it is considered safe or not, Therefore, in this study, better security will be made by combining it with (plaintext) where the public key that is set is changed first or encryption with (ciphertext) can only be cracked again with the RSA algorithm (Rivest Shamir Adleman). Data security is a very important aspect in today's information world. Especially if the information stored and sent in the document is very confidential, where outside parties are not allowed to know the contents of the data, because if the data is important so that it leaks to irresponsible parties it will cause losses for the company whose data is stolen or manipulated. the data. Some data that is very important for the company, company data, employee attendance and sales entry data, and others that are considered important and need to be secured by UD Tirta Soeper Teloer. One way that can be used is to encrypt the data through a cryptography application to deal with problems that exist in the company. Therefore, it is necessary to have a data security application that can be owned by the company. By implementing the RSA (Rivest Shamir Adleman) cryptographic algorithm, this application is built with the desktop-based Java programming language. The RSA method is used in this application to encrypt secret messages in the form of documents so that the confidentiality of the data is stronger and more secure. Based on the implementation and trial of the encryption and decryption application program that uses the RSA algorithm it becomes plaintext so that it can be read again, able to secure documents or data properly.*

Keywords: *Cryptography, RSA, asimetris, encryption, decryption*

1. PENDAHULUAN

Salah satu dampak negatif dalam perkembangan serta kemajuan teknologi informasi pada saat ini ialah pencurian data [1]. Dari kurangnya kesadaran dan kewaspadaan akan diamankan data dapat menimbulkan celah-celah dalam pencurian data melalui media elektronik atau media sosial sehingga sangat rentan sekali untuk penyalahgunakan oleh pihak yang tidak bertanggung jawab [2]. Dengan adanya pencurian data tersebut, maka aspek keamanan data dalam melakukan aktifitas pertukaran maupun penyimpanan data informasi sangatlah penting agar keaslian pada data tersebut akan tetap terjaga dengan baik [3]. Pada umumnya, perusahaan melakukan komputerisasi dengan menggunakan aplikasi Microsoft Office, pada pengolahan kata akan disimpan menggunakan aplikasi Microsoft Word maupun Microsoft Power Point, dan pada pengolahan angka akan disimpan menggunakan aplikasi Microsoft Excel, dan juga beberapa contoh dokumen yang perlu diamankan antara lain *.doc, *.xls, *.txt.

UD TIRTA SOEPER TELOER merupakan salah satu perusahaan yang bergerak di bidang penjualan telur ayam pada distributor telur manapun pembeli enceran. Penjualan telur pada umumnya sering kali mengalami perubahan setiap waktu seiring dengan adanya kondisi ekonomi dan bisnis. Para pemimpin suatu perusahaan atau para pelaku bisnis harus menemukan cara untuk terus mengikuti dan mengimbangi semua perubahan yang dapat mempengaruhi jalannya bisnis. Pada suatu perusahaan kita mendapati beberapa berkas pada setiap harinya. Berkas yang di maksud antara lain adalah berupa data absen karyawan, data harga maupun data data penjualan yang sedang digunakan oleh perusahaan. Maka, dapat dipastikan bahwa perusahaan tersebut memiliki dokumen-dokumen yang bersifat penting dan rahasia seperti data karyawan, data harga dan beberapa informasi mengenai perusahaan seperti harga naik maupun turun yang sedang di gunakan sekarang. Sehingga tidak boleh disebarluaskan kepada pihak luar yang tidak bertanggung jawab. Dokumen-dokumis yang bersifat penting atau rahasia yang dimiliki oleh perusahaan yang merupakan suatu intansi sangatlah rawan akan terjadinya pencurian data, sehingga dokumen-dokumen yang semestinya diamankan yakni dokumen yang bersifat sangat penting yang tidak boleh diketahui oleh pihak luar perusahaan yang dapat mengakibatkan resiko yang cukup fatal bagi perusahaan maupun karyawan. Maka diperlukan sebuah sistem yang disebut keamanan data guna mengantisipasi dari pencurian dan menyalahgunakan dokumen- dokumen tersebut dari pihak luar yang tidak bertanggung jawab. Pengamanan dengan menyadikan atau mengubah makna pesan tidak terbaca dengan menggunakan berbagai perhitungan, ilmu itu di sebut kriptografi [4].

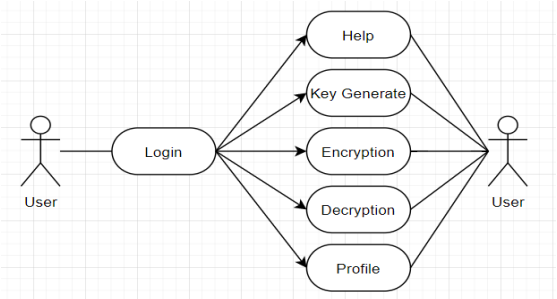
Kriptografi adalah pengaman pesan dimana pesan disandikan dengan menggunakan berbagai penghitung oesan sehingga pesan tersebut tidak dimengerti maknanya oleh orang lain. Kriptografi pertama kali ditemukan oleh bangsa Mesir pada tahun 3000 SM. Kriptografi berasal dari bahasa yunani yaitu kriptos dan graphia yang artinya “tulisan yang disembuyikan”. Beberapa hal yang membedakan antara kriptografi yaitu metode persandiannya. Semaki rumit metode yang digunakan maka pengamanan pesan akan lebih susah dipecahkan oleh pihak lain. Penelitian ini menggunakan algoritma RSA dalam mengamankan pesan [5].

Sistem keamanan data yang akan dibuat dengan menggunakan metode kriptografi dan algoritma RSA sebagai proses enkripsi dan dekripsi file. Metode kriptografi merupakan metode untuk mengamankan sebuah data, baik data tersebut berupa bentuk teks, maupun angkah. Dan untuk mengaplikasikan rancangan pengamanan dokumen menggunakan RSA, kami akan membuat aplikasi berbasis desktop [6].

2. METODE PENELITIAN

2.1 *Use Case Diagram* Penggunakan Aplikasi

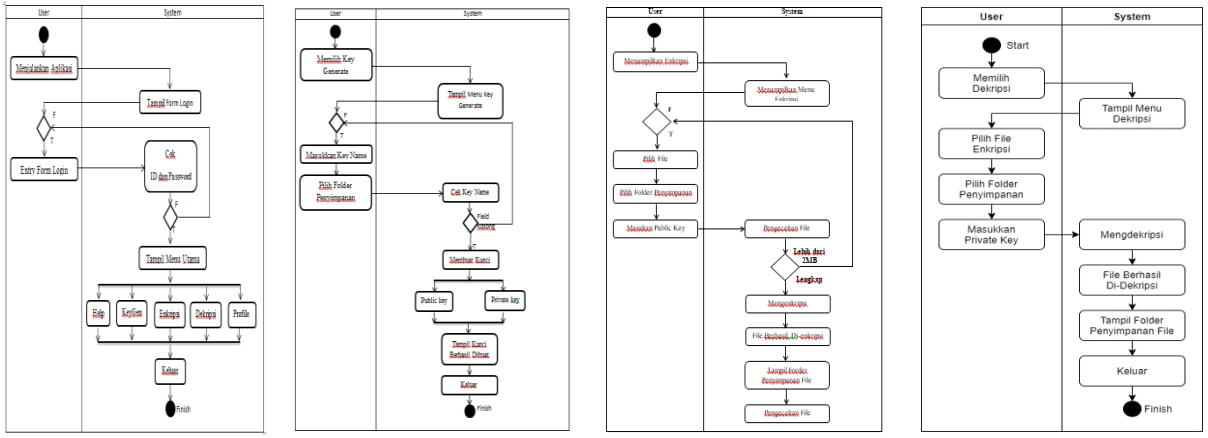
Use case diagram merupakan diagram yang menggambarkan hubungan antara User dengan sistem. *Use case* diagram juga bisa digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem dan bisa juga mempresentasikan sebuah interaksi *User* dengan sistem.



Gambar 1. Rancangan *Use Case* Aplikasi

2.2 *Activity* Diagram

Activity Diagram merupakan bagian dari penggambaran sistem secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasi oleh kode program. *Activity* Diagram memodelkan event- event yang terjadi didalam suatu *Use Case* dan digunakan untuk pemodelan aspek dinamis dari sistem.



Gambar 2. Rancangan Activity Diagram

3. HASIL DAN PEMBAHASAN

Algoritma program sangat penting saat pembuat program. Karena algortima dapat mempermudah dalam menjalankan suatu program. Dalam aplikasi ini terdapat dari beberapa algoritma yang digunakan untuk menjalankan proses proses ini.

3.1 Algoritma *Form* Menu Utama

```
1.  Tampilan semua isi form login
2.  If action = login
3.      Tampil form Menu Utama
4.      Pilih action
5.      If action Generate Key
6.          Tampil semua isi form Generate Key
7.      Else if action = Encryption
8.          Tampil semua isi form Generate Key
9.  Else if action = Decryption
10. Tampil semua isi form Decryption
11.     Else if action = Profile
12.         Tampil semua isi form Profile
13.     Else if Action = Help
14.         Tampil semua isi form Help
15.     Else if action = exit
16.         Tampil Popup konfirmasi keluar
17.     Else
18.         Tampil Form Menu Utama
19.     Else if action = yes
20.         Exit program
21.     Else
22.         Tampil form Menu Utama
23.     End If
```

Algoritma Generate Key merupakan fungsi yang digunakan untuk melakukan kunci public dan kunci *private*:

Gambar 3. Algoritma Form Menu Utama

3.2 Algoritma *Form Generate Key*

Algoritma di bawah ini merupakan bagaimana proses pembuatan kunci terjadi. Dijelaskan jika user memilih *form* yang ada, dan dapat mengetaui apa yang terjadi jika user memilih tombol *save to*, *generate key*, *clear*, dan *close*:

```
1.  Tampilkan form generate key
2.  Masukan key name
3.  If action = save to
4.  Pilih directory penyimpanan key
5.  Else
6.          Kembali ke form generate key
7.  If action = save to then
8.  Simpan kunci
9.  Else
10.         Kembai ke form generate key
11. Else If action = generate key then
12. Buat kunci
13. If form = terisi then
14. Progress bar berjalan
15. Tampilakan pesan berhasil dan waktu pembuatan kunci
16. If action = ok then
17. Kembali ke form generaye key
18. Else
19.         Kembali ke proses pembuatan kunci
20. End If
21. Else
22.       Tampilkan peringatan kolom kosong
23. If action = ok then
24. Kembali ke tahap awal
25. Else
26.       Kembali ke tahap awal
27.       End If
28. Else If action = clear then
29.       Bersihkan data yang sedang dimasukan
30.       Kembali ke proses pembuatan kunci
31. Else If action = close then
32.       Kembali ke form MU
33. End If
```

Gambar 4. Algoritma *Form Generaye Key*

3.3 Algoritma *Form Encryption*

Algoritma di bawah ini melakukan bagaimana proses form encryption terjadi. Dijelaskan jika user memilih form yang ada, dan dapat mengetahui apa yang terjadi jika user memilih tombol *browse, save to, open, encryption, clear,* dan *close file*:

```
1.  Tampilkan semua isi form enkripsi
2.  Masukan file
3.  Pilih file
4.  If file ≤ 2Mb then
5.  Tampilan pesan peringatan ukuran file
6.          If action = ok then
7.  Kembali ke proses enkripsi
8.  Else
9.          Kembai ke proses enkripsi
10. End If
11. Else If action = save to then
12.         Pilih directory
13. Else
14.       Kembali ke form encryption
15. Else If
16.       Pilih kunci publik
17.       Pilih action
18. If action = open then
19. Kembali ke proses enkripsi
20. Else If action = enkrip then
21.       If form = terisi then
22.                 Proses enkripsi
23.                 Progress bar berjalan
24.         Tampilkan pesan enkrip berhasil
25. If action = ok then
26.                 Tampikan waktu enkrip
27.                 Tampilkan direktori tempat menyimpanan file
28.                 Kembali ke proses enkrip
29. Else
30                  Kembali ke proses enkrip
31. End If
32. Else
33              Tampilan pesan form kosong
34              If action = ok then
35                  Kembali ke proses enkripsi
36              Else
```

Gambar 5. Algortima *Form Encryption*

3.4 Algoritma *Form Decrption*

Algoritma dibawah ini merupakan bagaimana proses form decryption terjadi. Dijelaskan jika user memilih form yang ada, dan dapat mengetahui apa yang terjadi jika user memilih tombol browse,save to,open,decryipt,clear,dan close:

```
1.  Tampilkan form dekripsi
2.  Masukan file
3.  If action = browse
4.  Pilih file yang sudah di enkrip
5.  If action = save to then
6.  Pilih tempat menyimpan
7.  If action = save to then
8.  Kembali ke proses dekrip
9.  Else
10.         Kembali ke form decryption
11. If action open then
12. Pilih private key
13. Else
14.         Kembali ke form decryption
15. Else If action = decrypt then
16.   If form = terisi then
17.             Proses dekripsi
18.             Progress bar berjalan
19.             Tampilkan pesan dekrip berhasil
20.             If action = ok then
21.             Tampilan direktori tempat penyimpanan file
22.             Kembali ke proses dekripsi
23.         Else
24.           Tampilan pesan form kosong
25.           If action = ok then
26.                 Kembali ke proses dekripsi
27.             Else
28.                 Kembali ke proses dekripsi
29.             End If
30. Else If action = hapus then
31. Bersikan data yang sedang dimasukan
32. Kembali ke form enkripsi
33. Else If action = close then
34. Kembali ke menu utama
35. End If
```

Gambar 6. Algoritma *Form Profile*

3.5 Algoritma *Form Profile*

Algoritma di bawah ini merupakan bagaimana proses *form profile* terjadi. Dijelaskan jika *user* harus diklik tombol *help* dan *form* akan kembali ke menu utama jika *user* memilih tombol kembali :

```
1.  Tampilkan form profile
2.  If action = kembali then
3.      Kembali ke MU
4.  Else
5.      Tetap di form profile
6.  End If
```

Gambar 7. Algoritma Form Profile

3.6 Algoritma *Form Help*

Algoritma di bawah ini merupakan proses form bantuan terjadi. Dijelaskan apabila pengguna ingin mengetahui cara menggunakan fitur yang ada di program ini, maka pengguna harus klik tombol bantuan. Form akan tertutup dan kembali ke menu utama apabila pengguna klik tombol kembali:

```
Tampilkan form dekripsi
Masukan file
If action = browse
Pilih file yang sudah di enkrip
If action = save to then
Pilih tempat menyimpan
If action = save to then
Kembali ke proses dekrip
Else
        Kembali ke form decryption
If action open then
Pilih private key
Else
        Kembali ke form decryption
Else If action = decrypt then
    If form = terisi then
                Proses dekripsi
                Progress bar berjalan
                Tampilkan pesan dekrip berhasil
                If action = ok then
                Tampilan direktori tempat penyimpanan file
                Kembali ke proses dekripsi
            Else
                Tampilan pesan form kosong
                If action = ok then
                        Kembali ke proses dekripsi
                    Else
                        Kembali ke proses dekripsi
                    End If
    Else If action = hapus then
    Bersikan data yang sedang dimasukan
    Kembali ke form enkripsi
    Else If action = close then
    Kembali ke menu utama
    End If
```

Gambar 8. Algoritma *Form Help*

3.7 Algoritma Proses Pembuatan Kunci RSA

Di bawa ini merupakan proses pembuatan kunci dibuat:

```
Start
Cari bilangan prima p dan q secara acak
Hitung modulu n = p.q
Hitung m/phe = (p-1).(q-1)
Hitung nilai e
If e = If e = gcd(e,m) = 1 dengan 1 < e < m/phe
      Hitung nilai d
If d = = (d x e) mod n = 1
Dapat kunci publik dan kunci pribadi
Simpan kunci publik dan kunci pribadi
Else
      Hitung kembali nilai d
      End If
Else
      Hitung kembali nilai e
End If
```

Gambar 9. Algoritma Proses Pembuatan Kunci RSA

3.8 Algoritma Proses Enkripsi RSA

Di bawa ini merupakan bagaimana proses kunci enkirpsi *RSA* terjadi:

```
Start
Masukan file
Baca file dalam tipe data data byte
Ambil nilai byte array jadi string
      Dapat nilai ASCII
      Dapat plainteks
Baca kunci publik (e,n)
Enkripsi file c = p ^ e mod n
Dapat cipherteks
Ubah cipherteks menjadi string
Gabungkan cipherteks dengan spasi
Buat file enkripsi
Simpan file enkripsi ke dalam folder
End If
```

Gambar 10. Algoritma Proses Enkripsi RSA

3.9 Algoritma Proses Dekripsi RSA

Di bawa ini merupakan bagaimana proses kunci dekripsi terjadi:

1. Start
2. Masukan file
3. Baca file enkripsi
4. Hapus spasi dari file enkripsi
5. Dapat cipherteks
6. Baca kunci publik (d,n)
7. Dekripsi file p = c ^ d mod n
8. Dapat plainteks
9. Dapat file ASCII
10. Ubah ASCII ke dalam byte array
11. Buat file dekripsi ke byte array
12. Simpan file dekripsi ke dalam folder
13. End

Gambar 11. Algoritma Proses Dekripsi RSA

3.10 Pengujian Aplikasi

Pada bagian ini dilakukan langkah pengujian dari rancangan sistem yang sudah dibuat. Berikah langkah pengujian sistem aplikasi pengamanan data dengan algoritma RSA, berbasis Java Desktop :
Berikut ini adalah pengujian program yang bertujuan untuk mengetahui apakah program dapat berjalan dengan baik setelah kebutuhan software dan hardware sudah terpenuhi untuk diuji coba.

Gambar 12. Pengujian Aplikasi

3.11 Tabel Pengujian Data Proses Enkripsi dan Proses Dekripsi

Pada pengujian ini, akan membahas tentang perbandingan- perbandingan antara proses encryption dan proses decryption. File yang digunakan untuk pengujian yaitu *.doc, *.xls, dan *.txt.

Tabel 1. Pengujian Proses *Encruption*

Nama File Asli	Ukuran File Asli	Ukuran File Hasil Encryption	Waktu Encryption (Detik)
*Customer ID *.doc*	*160 KB*	*439 KB*	*1047*
*Customer ID 1 *.xls*	*28 KB*	*82 KB*	*0.288*
*Customer ID 2 *.txt*	*4 KB*	*14 KB*	*0.328*

Tabel 2. Pengujian Proses Decryption

Nama File Asli	Ukuran File Encryption	Ukuran File Hasil Decryption	Waktu Decryption (Detik)
en.Customer ID *.doc	439 KB	160 KB	109.051
en.Customer ID 1 *xls	82 KB	28 KB	37.093
en.Customer ID 2 *.txt	14 KB	4 KB	21.309

4. KESIMPULAN

Berdasarkan perancangan, pembuatan, serangkaian uji coba dan analisa program aplikasi kriptografi ini, maka dapat diambil suatu kesimpulan diantara lain :

a. Dengan adanya aplikasi kriptografi, proses penyimpanan data-data perusahaan yang penting menjadi lebih aman dan nyaman dari orang orang yang tidak berkepentingan.
b. Aplikasi Implementasi Algoritma RSA untuk enkripsi dan dekripsi data menggunakan Java Desktop dapat digunakan untuk menjaga keaslian data (authentication) dan keutuhan data (integrity).
c. Waktu yang digunakan untuk melakukan proses enkripsi dan dekripsi berbanding lurus dengan ukuran file yang diproses (semakin kecil ukuran file yang diproses maka akan semakin cepat proses enkripsi dan dekripsi dilakukan, semakin besar ukuran file yang diproses maka akan semakin lama proses enkripsi dan dekripsi dilakukan).
d. Proses dekripsi dengan kunci yang sesuai akan mengembalikan file menjadi file semula tanpa mengalami perubahan sedikit pun.

DAFTAR PUSTAKA

[1] M. F. Wicaksono, "Implementasi Modul Wifi Nodemcu Esp8266 Untuk Smart Home," *Komputika: Jurnal Sistem Komputer*, vol. 6, no. 1, pp.1-6, 2017.
[2] A. Arief, and R. Saputra, "Implementasi Kriptografi Kunci Publik Dengan Algoritma RSA–CRT Pada Aplikasi Instant Messaging," *Scientific Journal of Informatics*, vol. 3, no. 1, 2016.
[3] Fatonah, and D. I. Mulyana, "Implementasi Metode Rivest Shamir Adleman Untuk Enkripsi dan Dekripsi Text," *JICOM: Jurnal Informatika dan Teknologi Komputer*, vol. 3, no. 1, pp. 32-39, 2022.
[4] R. Kurniawan, R., "Rancang Bangun Aplikasi Pengaman Isi File Dokumen Dengan Algoritma RSA," *ALGORITMA: Jurnal Ilmu Komputer dan Informatika*, vol. 01, no. 01, pp. 46-52, 2017.
[5] M. A. Fahreza, and A. H. Harbani, "Aplikasi Keamanan Data Gambar Menggunakan Algoritma RSA (Rivest Shamir Adleman) Berbasis Deskstop," *TeknoIS: Jurnal Ilmiah Teknologi Informatika dan Sains*, vol. 9, no. 1, pp. 1-9, 2019.
[6] S. Muliati, "Implementasi Kriptografi Dengan Menggunakan Metode DES Dan Kunci Publik RSA Untuk Mengamankan Data Pada Kandeoag Kota Jakarta Timur," *SKRIPSI Fakultas Sains dan Teknologi UIN Jakarta*. 2008.
[7] S. T. C. Kurniawan, Dedih and Supriyadi, "Implementasi Kriptografi Algoritma Rivest Shamir Adleman Dengan Playfair Ciper Pada Pesan Teks Berbasis Android," *Jurnal Online Informatika*. Vol. 2, no. 2, pp. 102-109, 2017.
[8] Sukaesi, and S. Wahyuni, "Penerapan Algoritma Kriptografi Rives Shamir Adleman (RSA) Pada Sebuah Images," *Jurnal Informatika Simantik*, vol.1, no. 1, pp. 32-39, 2016.
[9] S. Sutejo, "Implementasi Algoritma kriptografi RSA (Rivest Shamir Adleman) Untuk Keamanan Data Rekam Medis Pasien," *INTECOMS: Journal of Information Technology and Computer Science*, vol. 4, no. 1, pp. 104-114, 2021.

[10] M. K. Harahap and Rina, "Kombinasi Kriptografi RSA dengan Linear Congruential Generator," *SinkrOn: Jurnal & Penelitian Teknik Informatika*, vol. 3, no. 1, pp. 267–271, 2018.
[11] R. Harahap, "Implementasi Algoritma Skipjack Untuk Mengamankan Audio," *TIN: Terapan Informatika Nusantara*, vol. 2, no. 1, pp. 29–34. 2021.