JISKa, Vol. 6, No. 3, SEPTEMBER, 2021, Pp. 171 – 177
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Implementasi Algoritma RC4 pada Sistem Pengamanan Dokumen Digital Soal Ujian

Fauziyah Suwarsita Febriyani [(1)*], Arief Arfriandi [(2)]
Teknik Elektro, Fakultas Teknik, Universitas Negeri Semarang, Semarang
e-mail : fauziyahsuwarsita@gmail.com, arfriandi@mail.unnes.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 24 Mei 2021, direvisi 18 September 2021, diterima 19 September 2021, dan dipublikasikan 22 September 2021.

Abstract

*The development of science and technology has led to changes in the use of documents in life to become digital data. However, this can cause problems, namely regarding data security and confidentiality. To increase security and confidentiality can be done with cryptographic algorithm RC4. The research method uses the Waterfall method. The result of this research is a website that can secure document files with * doc extension using the RC4 algorithm. The test was carried out using the blackbox test and the CrackStation test for encryption testing. The results of the test show that the website can run well and successfully implements the RC4 algorithm.*

Keywords: Digital Data, Cryptography, RC4 Algorithm, Waterfall, Website

Abstrak

Dokumen dalam kehidupan sehari-hari telah berganti dari yang manual menjadi dokumen digital. Namun hal tersebut dapat menimbulkan masalah yaitu mengenai keamanan dan kerahasiaan data. Untuk meningkatkan keamanan dan kerahasiaan dapat dilakukan dengan kriptografi algoritma RC4. Metode penelitian menggunakan model *waterfall*. Hasil dari penelitian yaitu sebuah *website* yang dapat mengamankan file dokumen berekstensi **doc* menggunakan algoritma RC4. Pengujian dilakukan dengan menggunakan uji *blackbox* dan uji CrackStation untuk pengujian enkripsi. Hasil dari pengujian menunjukkan sistem dapat berjalan dengan baik dan berhasil mengimplementasikan algoritma RC4.

Kata Kunci: Data Digital, Kriptografi, Algoritma RC4, *Waterfall*, *Website*

1. PENDAHULUAN

Penggunaan dokumen digital saat ini telah banyak digunakan, salah satunya yaitu pada pembuatan soal ujian sekarang telah berkembang menjadi data digital. Sehingga menjadi lebih mudah diakses baik oleh guru maupun siswa.

Kemudahan pengaksesan informasi, tentunya berdampak pada munculnya resiko dan ancaman keamanan dan integritas data (Sumarno, 2018). Menurut DataLossDB (2016), pada tahun 2014 kebocoran data sekitar 50% terjadi pada sektor bisnis, sekitar 20% terjadi pada sektor pemerintahan dan sekitar 30% terjadi pada sektor kesehatan dan pendidikan. Kebocoran data juga terjadi pada pengguna pribadi, tetapi sulit untuk mengetahui jumlah persis dan tingkat keparahan kebocoran pada data pribadi (Alneyadi et al., 2016).

Untuk menjaga keamanan data dan menghindari kebocoran data dapat dilakukan berbagai cara salah satunya yaitu dengan menggunakan kriptografi. Kriptografi merupakan ilmu yang menggunakan teknik-teknik matematika yang berkaitan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, autentikasi entitas, dan autentikasi asal data (Menezes et al., 2001). Dalam ilmu kriptografi terdapat berbagai macam algoritma salah satunya yaitu RC4.

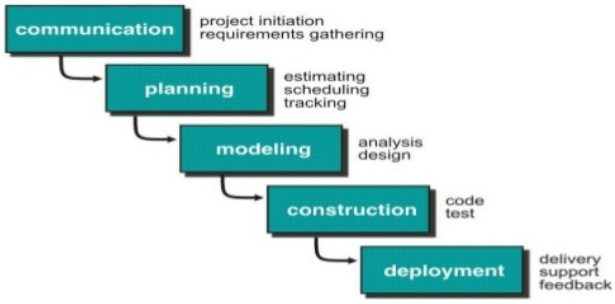
Berdasarkan cara kerjanya, algorima kriptografi RC4 merupakan jenis *stream cipher*. Kelebihanya adalah dengan menggunakan cara ini enkripsi algoritma RC4 dapat dilakukan pada data dengan panjang yang beragam. Algoritma RC4 dinilai sangat cepat dalam prosesnya, kurang lebih 10 kali lebih cepat dari DES (Hakim et al., 2014). Algoritma dari metode RC4 *stream cipher* ini terdiri dari dua bagian, yaitu: *key setup* dan *stream generation.* Pada *key setup* terdapat

tiga tahapan proses di dalamnya, yaitu Inisialisasi S-Box, enyimpan *key* dalam *key byte array*, permutasi pada S-Box. Pada *stream generation* akan menghasilkan nilai *pseudorandom* yang akan dikenakan operasi XOR untuk menghasilkan *ciphertext* ataupun sebaliknya yaitu menghasilkan *plaintext* (Saragi et al., 2020).

Penelitian terkait implementasi algoritma RC4 pernah dilakukan oleh peneliti terdahulu. Seperti penelitian yang dilakukan oleh Ridho & Jemakmun (2020), pada penelitian tersebut sistem bertujuan untuk mengamankan *database* pegawai. Waluyo & Kanahebi (2021) dan Purba et al. (2020) telah melakukan penelitian dengan mengimplementasikan algoritma untuk mengamankan file teks. Zebua & Ndruru (2017) juga melakukan penelitian untuk menguraikan pengamanan citra digital berdasarkan modifikasi algoritma RC4. Penelitian lainnya yaitu penelitian yang dilakukan oleh Subhan et al. (2017) dengan mengimplementasikan algoritma RC4 berbasis android.

2. METODE PENELITIAN

Desain atau metode penelitian yang digunakan pada penelitian ini adalah model *waterfall*. Menurut Pressman (2003) model *waterlfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*. Terdapat lima tahapan pada metode penelitian *waterfall*, antara lain komunikasi, perencanaan, pemodelan, konstruksi, dan *deployment*.



Gambar 1. Tahapan Model *Waterfall* (Pressman, 2003).

2.1. Komunikasi

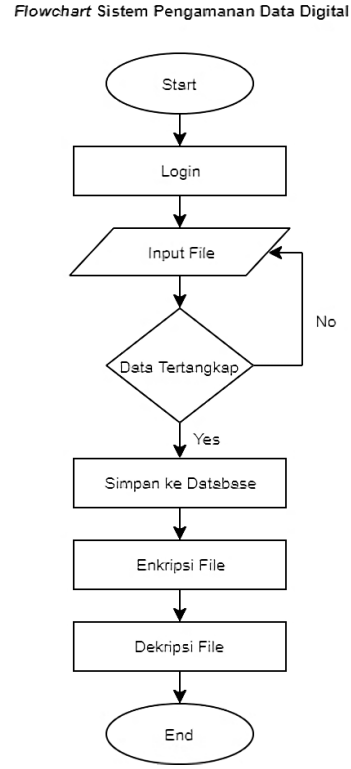
Langkah awal dalam membangun sistem yaitu menganalisis dan mengkomunikasikan kebutuhan-kebutuhan sistem berdasarkan hasil pengumpulan data. Dari tahap komunikasi diperoleh kesimpulan dibutuhkan adanya sistem untuk mengamankan dokumen ujian SMK Negeri 1 Slawi.

2.2. Perencanaan

Tahap perencanaan menjabarkan tentenag tugas-tugas tenik yang harus dilakukan mulai dari pengumpulan data sampai pengujian sistem, resiko yang mungkin terjadi pada saat menjalankan tugas-tugas tersebut, dan hasil yang ingin diperoleh yaitu terciptanya sebuah sistem pengamanan dokumen digital soal ujian dengan menggunakan algoritma RC4.

2.3. Pemodelan

Tahap pemodelan berfokus pada perancangan skema sistem pengamanan dokumen digital soal ujian kemudian merancang *database* menggunakan *MySQL* sebagai tempat penyimpanan data sistem. Selanjutnya merancang antarmuka sistem pengamanan dokumen digital soal ujian. *Flowcart* Sistem dapat dilihat pada Gambar 2.



Gambar 2. Flowchart Sistem

Sistem dimulai dengan *login* yaitu *user* memasukkan *username* dan *password*. Kemudian, *user* dapat menginputkan file ke dalam sistem, yang selanjutnya sistem akan menangkap file tersebut dan disimpan ke dalam *database.* Selanjutnya file tersebut dapat dienkripsi dan kemudian didekripsikan kembali jika diperlukan oleh *user*.

2.4. Konstruksi

Konstuksi merupakan tahap pembuatan kode, sehingga tercipta sistem yang telah dirancang sebelumnya. Bahasa pemrograman yang digunakan untuk membangun sistem adalah PHP dan menggunakan *database MySQL* sebagai penyimpanan datanya. Algoritma RC4 digunakan untuk mengamankan data yang dimasukkan ke dalam *database*. Dan kemudian dapat dikembalikan lagi seperti semula saat diperlukan.

Setelah proses pengkodean selesai, maka akan dilakukan pengujian terhadap sistem yang telah dibuat. Pengujian sistem dilakukan menggunakan *blackbox testing*. Pengujian *blackbox* dilakukan untuk mengetahui apakah fungsi-fungsi pada program dapat berjalan dengan baik, mulai dari menerima input, memproses, dan memberikan *output*. Pengujian selanjutnya menggunakan *software* penyerang yang berfokus terhadap hasil implementasi enkripsi algoritma RC4 pada isi file dokumen. *Software* yang digunakan adalah CrackStation.

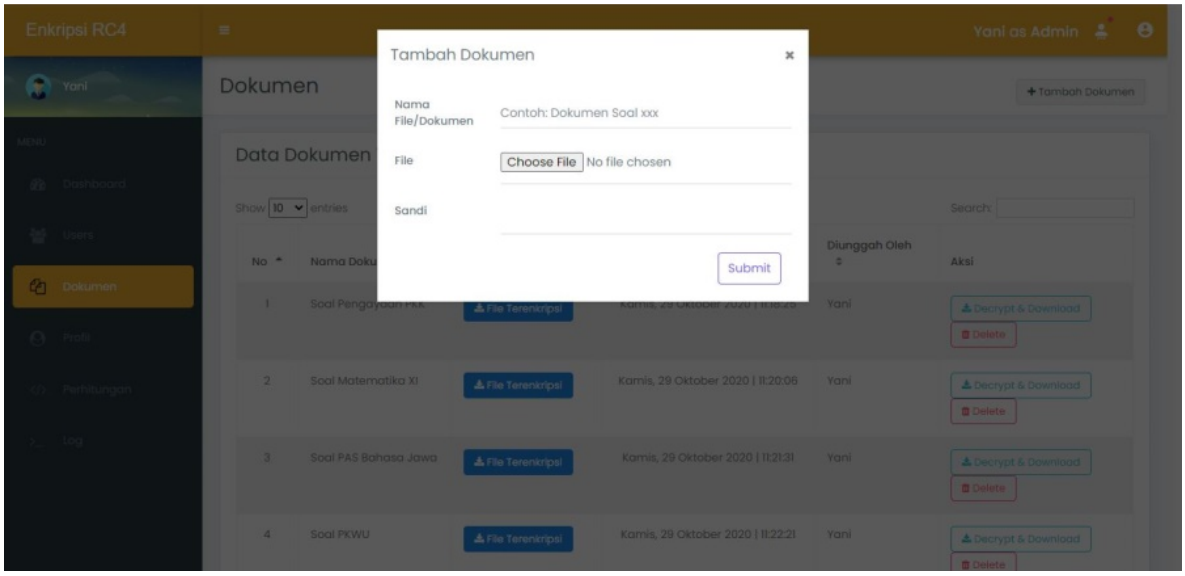
2.5. *Deployment*

Tahap *deployment* merupakan tahapan dimana sistem telah siap digunakan oleh pengguna. Kemudian untuk menjaga sistem tetap berjalan dengan baik maka perlu dilakukan pemeliharaan secara berkala sesuai dengan kebutuhan.

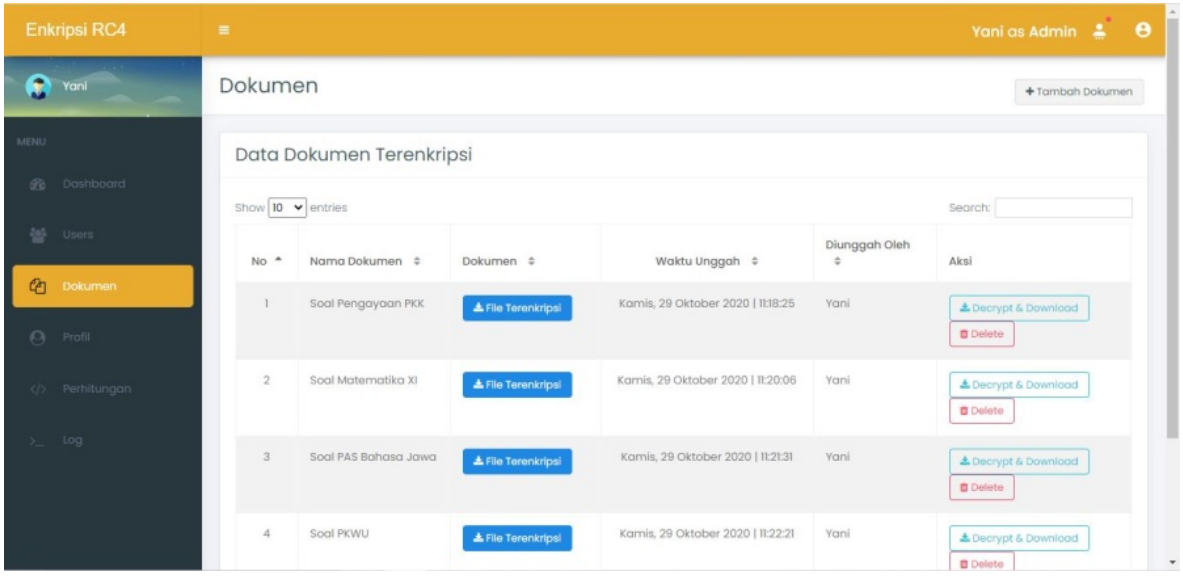
3. HASIL DAN PEMBAHASAN

3.1. Hasil Implementasi

Sistem dapat mengenkripsi dan mendekripsikan kembali file dokumen dengan format file *doc*. menggunakan algoritma RC4 sehingga dapat mengamankan dokumen soal ujian. Untuk mengamankan dokumen, *user* harus menginputkan dokumen beserta kuncinya seperti yang dapat dilihat pada Gambar 3. File yang berhasil diamankan akan masuk ke dalam *database* dan ditampilkan pada menu dokumen sperti pada Gambar 4. Kemudian jika ingin mendekripsikannya klik tombol *decrypt & download* kemudian *user* harus menginputkan kunci yang sama seperti saat mengamankan dokumen.



Gambar 3. *Form* Tambah Dokumen



Gambar 4. Halaman Daftar Dokumen Terenkripsi

3.2. Hasil Uji *Blackbox*

Hasil pengujian *blackbox* menunjukan bahwa sistem dapat berjalan dengan baik sesuai dengan input yang diberikan. Hasil pengujian *blackbox* dapat dilihat pada Tabel 1.

Tabel 1. Hasil Uji *Blackbox*.

No.	Test Case	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian
1.	Memasukkan *username* dan *password*.	Mengisi *form login* pada halaman *login* kemudian pilih tombol *login*.	Saat data yang sudah terisi sesuai dengan data pada *database* maka akan masuk menuju halaman *dashboard*. Apabila tidak sesuai atau kosong maka akan muncul peringatan.	Sesuai
2.	Melihat data *users*.	Pilih menu *users*.	Sistem menampilkan daftar semua *user* yang ada dalam *database*.	Sesuai
3.	Menambah *user*.	Klik tombol tambah *user*.	Ketika klik tombol tambah *user* maka muncul *form* "Add New User" data yang harus diisi. Ketika form sudah diisi, data akan tersimpan dan muncul pesan berhasil.	Sesuai
4.	Mengubah dan menghapus data *user*.	Memfungsikan tombol "Edit" dan "Delete" pada daftar *user*.	Ketika klik "Edit" sistem menampilkan *form edit* data *user* yang dipilih. Ketika klik "Delete" sistem menghapus data *user* yang dipilih.	Sesuai
5.	Mencari data *user*.	Mengetik nama *user* yang dicari pada kolom pencarian.	Sistem menampilkan hasil pencarian sesuai dengan nama *user* yang dimasukkan pada kolom pencarian.	Sesuai
6.	Melihat data data dokumen.	Pilih menu dokumen.	Sistem menampilkan daftar semua dokumen.	Sesuai
7.	Menambah dokumen (enkripsi dokumen).	- Klik tombol tambah dokumen. - Mengisi semua *field* termasuk sandi untuk enkripsi.	Ketika klik tombol tambah dokumen maka muncul *form* "Tambah Dokumen". Setelah semua field terisi data dokumen akan tersimpan dan terenkripsi dan muncul pesan sistem berhasil.	Sesuai
8.	Men-*download* dan mendekripsi dokumen.	Klik tombol "Decrypt & Download" pada daftar dokumen.	Ketika klik tombol "Decrypt & Download" dokumen akan ter-*download* dalam keadaan sudah terdekripsi.	Sesuai
9.	Melihat profil.	Pilih menu profil.	Sistem menampilkan detail data *user* dan riwayat dari user.	Sesuai
10.	Mengubah data profil *user*.	Pilih submenu "Edit Profil" pada menu profil.	Ketika klik "Edit Profil" maka sistem menampilkan *field* edit data profil *user*.	Sesuai
11.	Melakukan perhitungan	Pilih menu perhitungan kemudian mengisi *form* perhitungan.	Setelah *form* perhitungan terisi maka muncul alur perhitungan enkripsi algoritma RC4 sesuai dengan teks dan sandi yang dimasukkan.	Sesuai
12.	Melihat *log* aktivitas sistem	Pilih menu *log*	Sistem menampilkan *log* aktivitas yang terjadi pada sistem.	Sesuai

3.3. Hasil Uji CrackStation

Hasil pengujian terhadap 10 sampel diketahui bahwa algoritma RC4 tidak berhasil dipecahkan. Hasil uji CrackStation dapat dilihat pada Gambar 5.

Hash	Type	Result
ÂþÌ□□□□Â□ßÜb%ob□·□iMò(□?ñ¿¡	Unknown	Unrecognized hash format.
□ÏëË□□□□Ú□□Ò□Y!o%2«□iP	Unknown	Unrecognized hash format.
□ÆäÌ□□□×ÿB(Ø□[2!V□£□{W·û3´□□¬	Unknown	Unrecognized hash format.
□ÏëÉß0□×ÿ□□ÝUa5rp□¯□	Unknown	Unrecognized hash format.
□ÆäÌ□□□×ÿB(Ø□[2!V□£□{W·û3°□□°	Unknown	Unrecognized hash format.
□ÌëÉß0□ßð□Ià□Y/md□	Unknown	Unrecognized hash format.
ÂþÄß5□Úð□□Á□\`Ld«□iW»□º	Unknown	Unrecognized hash format.
□éÅê¬ WâÙ'Iñ¹a□!D*□²Z□□=□;□	Unknown	Unrecognized hash format.
ÂþÄß5□Úð□□Á□\`Cd□¯□i□□□¼□£□□	Unknown	Unrecognized hash format.
□Êæì□ □Þ±□□ß□Z`rd»×bB¥□¹□¿	Unknown	Unrecognized hash format.

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Gambar 5. Hasil Uji CrackStation

3.4. Pembahasan

Penelitian ini menggunakan metode penelitian model *waterfall* dimana setiap tahap dilakukan dengan urut tanpa saling tumpang tindih sehingga meminimalisasi terjadinya kesalahan. Setiap tahapan pada model *waterfall* juga memiliki spesifikasinya sendiri, sehingga sebuah sistem dapat dikembangkan sesuai dengan apa yang dikehendaki. Software yang dikembangkan dengan metode ini biasanya menghasilkan kualitas yang baik (Widiyanto, 2018).

Penelitian ini mengimplementasikan algoritma RC4 pada pengamanan sistem soal ujian kerana berdasarkan cara kerjanya algoritma RC4 merupakan algoritma jenis *stream cipher* yang memproses input data pada satu waktu. Input data pada umumnya berbentuk *byte* atau bahkan bit (*byte* dalam hal RC4) sehingga dengan menggunakan algortima RC4 proses enkripsi dan dekripsi dapat diimplementasikan pada file dokumen soal ujian dengan panjang yang bervariasi. Algoritma RC4 juga tidak perlu menunggu sejumlah input data atau pesan tertentu sebelum diproses, atau menambahkan *byte* tambahan untuk melakukan enkripsi (Nugroho et al., 2016).

Algoritma RC4 dirancang supaya dapat diimplementasikan di *software* dengan sangat efisien. Sehingga membuat algoritma RC4 sangat popular untuk aplikasi internet, antara lain RC4 digunakan dalam standar TLS (*Transport Layer Security*), dan WEP (*Wireless Equivalent Privacy*) (Agung & Budiman, 2015).

Pada pengujian CrackStation menggunakan 10 sampel sudah dapat membuktikan bahwa algoritma RC4 dapat diimplementasikan untuk enkripsi, ditunjukkan dengan *ciphertext* hasil dari proses enkripsi tidak dapat dipecahkan.

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dihasilkan sebuah sistem pengamanan dokumen digital dengan mengimplementasikan algoritma RC4. Sebagaimana pengujian yang telah dilakukan pada pengujian *blackbox* menunjukkan bahwa sistem dapat berjalan sesuai dengan inputan yang diberikan. Kemudian pada pengujian *software* CrackStation untuk menguji 10 sampel *ciphertext* menunjukkan bahwa *ciphertext* 100% tidak dapat dipecahkan.

DAFTAR PUSTAKA

Agung, H., & Budiman, B. (2015). IMPLEMENTASI AFFINE CHIPER DAN RC4 PADA ENKRIPSI FILE TUNGGAL. *Prosiding SNATIF*, *0*(0), 243–250.

Alneyadi, S., Sithirasenan, E., & Muthukkumarasamy, V. (2016). A survey on data leakage prevention systems. *Journal of Network and Computer Applications*, *62*(C), 137–152. https://doi.org/10.1016/j.jnca.2016.01.008

DataLossDB. (2016, February 11). *2015 Reported Data Breaches Surpasses All Previous Years*. https://datalossdbdotorg.wordpress.com/

Hakim, E. L., Khairil, K., & Utami, F. H. (2014). APLIKASI ENKRIPSI DAN DESKRIPSI DATA MENGGUNAKAN ALGORITMA RC4 DENGAN MENGGUNAKAN BAHASA PEMROGRAMAN PHP. *JURNAL MEDIA INFOTAMA*, *10*(1), 1–7. https://doi.org/10.37676/JMI.V10I1.226

Menezes, A. J., van Oorschot, P. C., & Vanstone, S. a. (2001). Handbook of Applied Cryptography. In *Handbook of Applied Cryptography* (Issue 9). CRC Press.

Nugroho, N. B., Azmi, Z., & Arif, S. N. (2016). APLIKASI KEAMANAN EMAIL MENGGUNAKAN ALGORITMA RC4. *Jurnal SAINTIKOM*, *15*(3), 81–88.

Pressman, R. S. (2003). *Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku II)*. Andi.

Purba, B., Gulo, F. A., Utami, N. I., & Sihotang, Y. A. (2020). Pengamanan File Teks Menggunakan Algoritma RC4. *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)*, 420–425.

Ridho, M., & Jemakmun. (2020). *IMPLEMENTASI KEAMANAN BASI DATA PADA DATA PEGAWAI DI PT TIRTA MUSI MENGGUNAKAN METODE RC4*. Universitas Bina Darma.

Saragi, D. R., Gultom, J. M., Tampubolon, J. A., & Gunawan, I. (2020). Pengamanan Data File Teks (Word) Menggunakan Algoritma RC4. *Jurnal Sistem Komputer Dan Informatika (JSON)*, *1*(2), 114–119. https://doi.org/10.30865/json.v1i2.1745

Subhan, S., Amini, S., & Ariyani, P. F. (2017). IMPLEMENTASI PENGAMANAN DATA ENKRIPSI SMS DENGAN ALGORITMA RC4 BERBASIS ANDROID. *Prosiding SENIATI*, A29.6.

Sumarno, S. (2018). Analisis Kinerja Kombinasi Algoritma Message-Digest Algortihm 5 (MD5), Rivest Shamir Adleman (RSA) dan Rivest Cipher 4 (RC4) Pada Keamanan E-Dokumen. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima (JUSIKOM PRIMA)*, *2*(1), 41–48.

Waluyo, S., & Kanahebi, D. V. (2021). Sistem Pengamanan File Menggunakan Algoritma Rc4 Berbasis Webbase Studi Kasus : Pt. Tjipta Jaya Bersama. *Semnas Ristek (Seminar Nasional Riset Dan Inovasi Teknologi)*, *5*(1), 803–808.

Widiyanto, W. W. (2018). ANALISA METODOLOGI PENGEMBANGAN SISTEM DENGAN PERBANDINGAN MODEL PERANGKAT LUNAK SISTEM INFORMASI KEPEGAWAIAN MENGGUNAKAN WATERFALL DEVELOPMENT MODEL, MODEL PROTOTYPE, DAN MODEL RAPID APPLICATION DEVELOPMENT (RAD). *Jurnal Informa : Jurnal Penelitian Dan Pengabdian Masyarakat*, *4*(1), 34–40. https://doi.org/10.46808/INFORMA.V4I1.34

Zebua, T., & Ndruru, E. (2017). Pengamanan Citra Digital Berdasarkan Modifikasi Algoritma RC4. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *4*(4), 275–282. https://doi.org/10.25126/jtiik.201744474