ALGORITMA CAESAR CIPHER DAN HILL CIPHER UNTUK MENINGKATKAN KEAMANAN DATABASE

Indri Neforawati, S.T., M.T., Jahuda Dolf Bacas
Teknik Informatika dan Komputer
Politeknik Negeri Jakarta
Depok, Jawa Barat, Indonesia
jahuda.dolfbacas.tik17@mhsw.pnj.ac.id

ABSTRAK

Database atau basis data menurut Stephens dan Plew dalam buku Simarmata & Paryudi adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Perkembangan teknologi saat ini membuat pengaksesan terhadap informasi semakin mudah dan memberikan pengaruh besar terhadap keamanan informasi yang menggunakan media penyimpanan. Salah satu masalah dari keamanan informasi adalah kebocoran data yang dapat disebabkan oleh pihak pihak tertentu. Maka dari itu diperlukan sebuah metode untuk menjaga *database* atau basis data tersebut agar tetap aman. Dalam penelitian ini, metode yang digunakan untuk mengamankan *database* adalah enkripsi menggunakan algoritma *caesar cipher* dan *hill cipher*. Algoritma caesar cipher mengenkripsi data dengan cara menggeser posisi *plaintext* sebanyak jumlah kunci dan algoritma *hill cipher* mengenkripsi dengan cara mengalikan hasil dari enkripsi *caesar cipher* dengan kunci matriks. Hasil akhir dari penerapan algoritma ini diperoleh pengamanan data pada *database* lebih terjamin keamanannya.

Kata kunci : Algoritma Hill Cipher, Algoritma Caesar Cipher, Kriptografi, Keamanan Basis Data

BAB I PENDAHULUAN

Database atau basis data menurut Stephens dan Plew dalam buku Simarmata & Paryudi adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah semua hal yang digunakan oleh setiap orang sehari-hari untuk berbagai alasan.

Seiring perkembangan zaman, teknologi semakin berkembang dengan pesat. Perkembangan teknologi ini membuat pengaksesan terhadap informasi semakin mudah dan memberikan pengaruh besar terhadap keamanan informasi yang menggunakan media penyimpanan. Hal ini mengakibatkan banyaknya pihak pihak tertentu yang ingin mengambil atau merusak data-data tersebut secara sengaja. Data-data yang diambil tersebut nantinya akan disalah gunakan untul hal lain, seperti diperjualbelikan kepada pihak lain. Keterbatasan pengguna atau petugas sistem untuk terus memantau perkembangan sistem menjadi sebuah celah bagi pihak yang tidak berkenan untuk menyalahgunakan informasi tersebut. Selama data tersebut diproses, dikirimkan dan sampai pada tujuan atau sebaliknya, data informasi tersebut haruslah bersifat rahasia, terjaga keasliannya atau tidak ada perubahan sama sekali. Maka dari itu diperlukan sebuah metode untuk menjaga *database* atau basis data tersebut agar tetap aman. Salah satu metode yang bisa digunakan adalah dengan menggunakan algoritma kriptografi.

Dalam ilmu matematika terdapat konsep yang disebut kriptografi. Kriptografi berasal dari bahasa Yunani, menurut bahasa dibagi menjadi dua *kripto* dan *graphia*, *kripto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Berdasarkan terminologinya, kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dipahami lagi maknanya.

Menurut [1] kriptografi memiliki dua jenis yaitu kriptografi klasik dan kriptografi modern. Dalam kriptografi klasik, terdapat beberapa teknik diantaranya substitusi dan transposisi. Di dalam teknik substitusi juga dibagi beberapa macam cara yaitu *Caesar Cipher, Playfair Cipher, Shift Cipher, Hill Cipher,* dan *Vinegere Cipher*. Kriptografi klasik merupakan awal dari kriptografi modern, ada tiga alasan mengapa perlu memahami konsep algoritma kriptografi klasik diantaranya untuk memberikan pemahaman terhadap konsep dasar kriptografi, sebagai dasar dari algoritma kriptografi modern dan agar dapat memahami potensi kelemahan pada *cipher*.

BAB II METODE

Untuk metode yang digunakan dalam penelitian ini terdiri atas perancangan program dan perancangan algoritma. Berikut penjelasan mengenai metode yang digunakan:

2.1 Perancangan program

1) Proses enkripsi tulisan atau teks

a. User berencana menginput teks atau ke dalam database
b. User memilih berapa banyak pergeseran huruf yang akan dipakai.
c. User memilih matriks key sesuai keinginannya.
d. Teks kemudian dienkripsi menggunakan pergeseran huruf dan matriks key yang dipilih.
e. Teks yang terenkripsi masuk kedalam database.

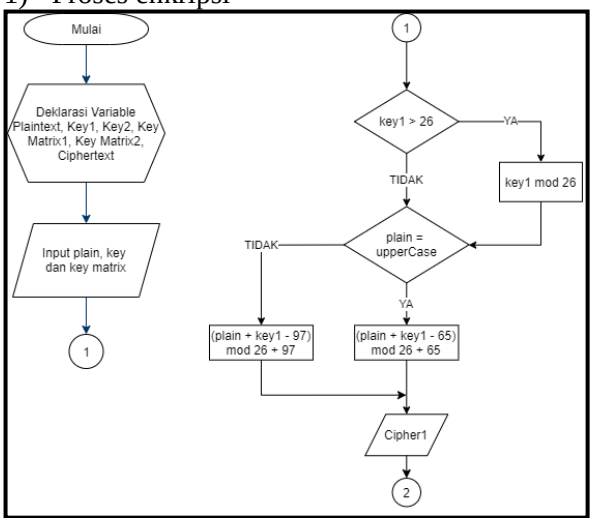
2) Proses dekripsi teks

a. Matriks key yang sebelumnya diinput dibuat inversnya.
b. Ciphertext yang ada pada database dicocokan dengan matriks key.
c. Kemudian dicocokan kembali dengan pergeseran huruf yang dipilih.
d. Plaintext akan muncul.

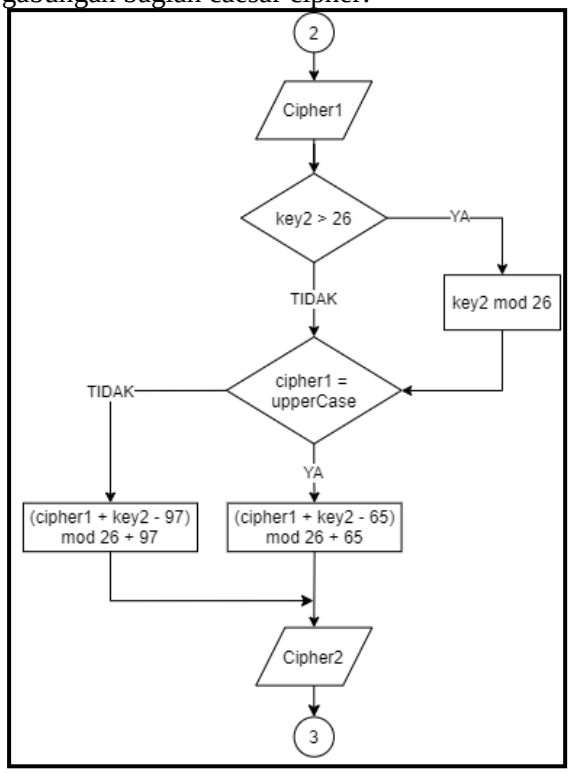
2.2 Perancangan Algoritma

Dalam penelitian ini, *Caesar Cipher* dan *Hill Cipher* akan digunakan dalam metode penggabungan algoritma. Sebelum diimplementasikan kedalam Bahasa *Java*, algoritma dirancang terlebih dahulu menggunakan *matlab*. Masing masing algoritma dibuat prosesnya setelah itu dilakukan uji coba terhadap algoritma yang sudah dibuat sebelum akhirnya diubah kedalam bahasa *java*. Teks yang nantinya diinput kedalam *database* akan dienkripsi terlebih dahulu menggunakan kedua algoritma tersebut sesuai dengan matriks *key* dan pergeseran huruf yang dipilih oleh *user*[2].

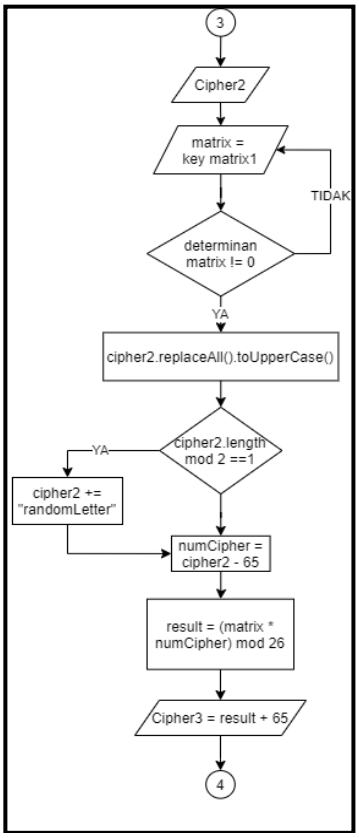
1) Proses enkripsi



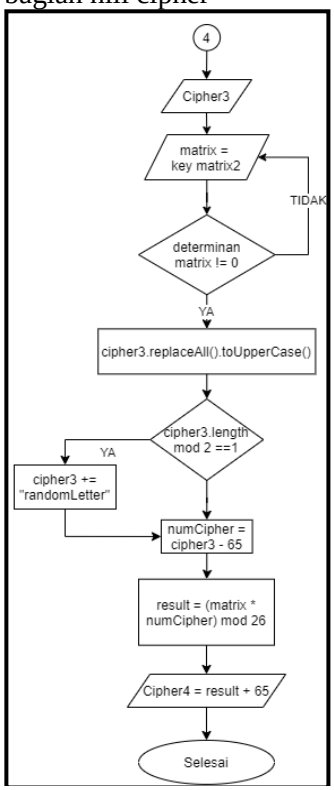
Gambar 1. Flowchart Enkripsi Algoritma gabungan bagian caesar cipher.



Gambar 2. Flowchart Enkripsi lanjutan algoritma gabungan bagian Caesar cipher

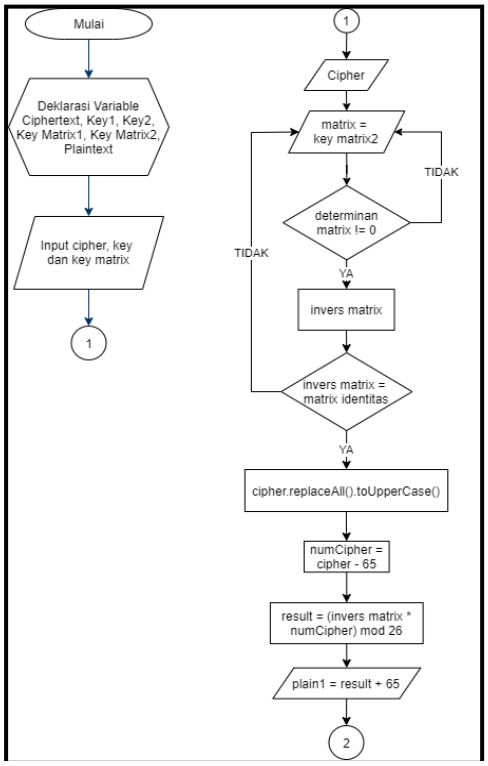


Gambar 3. Flowchart lanjutan enkripsi algoritma gabungan bagian hill cipher

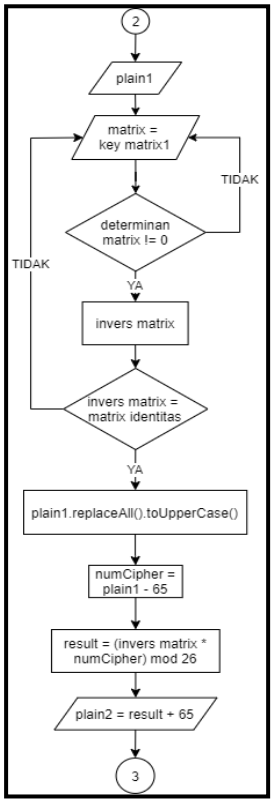


Gambar 4. Flowchart lanjutan enkripsi algoritma gabungan bagian hill cipher

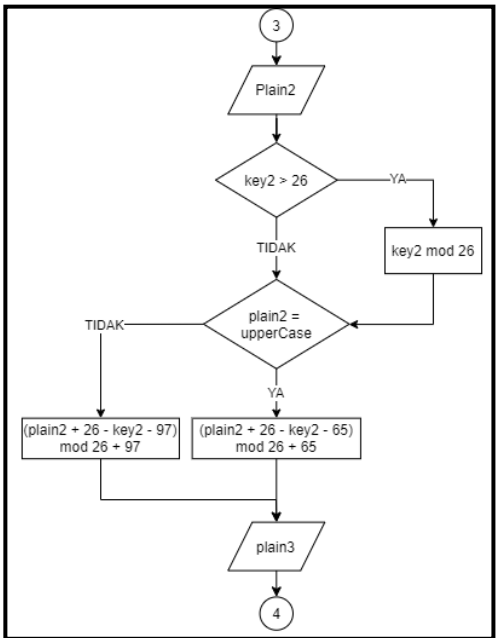
2) Proses dekripsi



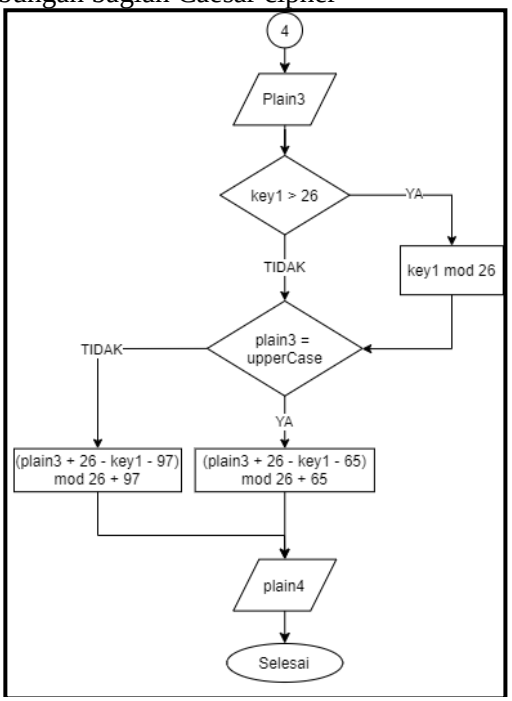
Gambar 5. Flowchart dekripsi algoritma gabungan



Gambar 6. Flowchart dekripsi lanjutan algoritma gabungan bagian hill cipher



Gambar 7. flowchart lanjutan dekripsi alogritma gabungan bagian Caesar cipher



Gambar 8. Flowchart lanjutan dekripsi algoritma gabungan bagian akhir

Pengujian dilakukan dengan cara menginput kata dengan berbagai jumlah karakter, mulai dari kata yang berjumlah ganjil hingga genap, serta kata yang menggunakan huruf besar maupun kecil. Kata yang di-input akan menjalani proses enkripsi dan dekripsi sesuai dengan algoritma yang telah digabung. Data dari hasil pengujian berupa waktu dalam satuan detik. Untuk pengujian keamanan dari hasil enkripsi menggunakan website online dan Cryptool 2. Dan pengujian terhadap hasil enkripsi apakah masuk kedalam database atau tidak.

BAB III HASIL DAN PEMBAHASAN

3.1 Analisis *Encryption Time*

Hasil pengujian perhitungan *Encryption Time* yang dilakukan pada *plaintext* yang memiliki jumlah karakter ganjil dan genap dapat dilihat pada tabel berikut ini:

Tabel 1. Hasil pengujian pada *Encryption Time*

No	Jumlah Karakter	*Encryption Time*
1	Ganjil	0.0187596 Detik
2	Genap	0.0103287 Detik

Berdasarkan Tabel 1, hasil *Encryption time* dipengaruhi oleh jumlah karakter pada *plaintext*. Karena pada pengujian jumlah karakter pada *plaintext* yang bersifat ganjil lebih banyak daripada *plaintext* yang bersifat genap, maka waktu yang diperlukan oleh *plaintext* ganjil sedikit lebih lama.

Encryption Time merupakan perhitungan waktu saat algoritma gabungan berjalan mengenkripsi *plaintext* yang diisi. *Encryption Time* pada *plaintext* yang bersifat ganjil (0.0187596 detik) lebih lambat jika dibandingkan dengan *plaintext* yang bersifat genap (0.0103287 detik).

3.2 Analisis *Decryption Time*

Hasil pengujian perhitungan *Decryption Time* yang dilakukan pada *plaintext* yang memiliki jumlah karakter ganjil dan genap dapat dilihat pada tabel berikut ini:

Tabel 2. Hasil pengujian pada *Decryption Time*

No	Jumlah Karakter	*Decryption Time*
1	Ganjil	(error)
2	Genap	0.0010884 Detik

Berdasarkan Tabel 2, hasil *Decrytion time* dipengaruhi oleh jumlah karakter pada *ciphertext*. Karena pada pengujian *ciphertext* yang bersifat ganjil akan error sehingga waktu

tidak akan muncul, sedangkan *ciphertext* yang bersifat genap akan tetap muncul waktunya.
Decryption Time merupakan perhitungan waktu saat algoritma gabungan berjalan mendekripsi *ciphertext* yang diisi ataupun dipilih. *Decryption Time* pada *ciphertext* yang bersifat ganjil akan error jika dibandingkan dengan *ciphertext* yang bersifat genap (0.0010884 detik).

3.3 Analisis Keamanan Hasil Enkripsi
Pengujian pemeriksaan keamanan dari hasil enkripsi yang dilakukan pada *plaintext* yang bersifat ganjil dan genap menggunakan *online website* dan *Cryptool* dapat dilihat pada tabel berikut ini

Tabel 3. Hasil uji keamanan pada hasil enkripsi

Jumlah Karakter	***Cipher text***	***Web (Caesar)***	***Web (Hill)***	***Cryptool***
Ganjil	QVQZET 19SWP PXTYW XPTM	Tidak berhasil dipecahkan	Tidak berhasil dipecahkan	Tidak berhasil dipecahkan
Genap	GPIFPHN WOZ:<BG	Tidak berhasil dipecahkan	Tidak berhasil dipecahkan	Tidak berhasil dipecahkan

Berdasarkan Tabel 3, *plaintext* yang dienkripsi oleh algoritma gabungan yang memiliki jumlah karakter ganjil maupun genap tidak berhasil dipecahkan atau di serang menggunakan metode *bruteforce*, baik itu menggunakan metode *Caesar cipher bruteforce, hill cipher bruteforce*, maupun menggunakan *software cryptool*. Hal ini membuktikan bahwa keamanan dalam program ini berfungsi dengan baik dan benar. Ini juga membuktikan bahwa untuk meningkatkan efisiensi dari algoritma *Caesar cipher* dapat dilakukan dengan menggabungkannya dengan algoritma lain.

3.4 Analisis hasil enkripsi kedalam *database*
Pengujian ini dilakukan untuk membuktikan apakah penggunaan dari algoritma gabungan ini dapat dilakukan untuk data yang ingin disimpan kedalam *database*, hasil dari pengujian dapat dilihat pada tabel berikut:

Tabel 4. Hasil uji enkripsi kedalam *database*

No	**Hasil dekripsi sesuai dengan plaintext**	**Berhasil disimpan kedalam *database***
1	Sesuai	Berhasil

Berdasarkan Tabel 4, hasil dari proses enkripsi pada *plaintext* yang memiliki jumlah karakter ganjil dan genap dapat disimpan kedalam *database*, hal ini membuktikan bahwa implementasi algoritma gabungan dapat digunakan hanya untuk sebagai alat enkripsi dekripsi maupun sebagai alat untuk meningkatkan keamanan data yang akan dimasukan kedalam *database*.

BAB IV SIMPULAN DAN SARAN

Setelah melakukan penelitian dengan metode yang telah direncanakan untuk menganalisis mekanisme penggabungan algoritma yaitu algoritma *Caesar cipher* dan *hill cipher* dan implementasi dari penggabungan algoritma kedalam bahasa *java*, maka dapat disimpulkan bahwa gabungan dari kedua algoritma ini dapat digunakan untuk meningkatkan keamanan data yang berada dalam *database*. Kesimpulan yang dapat diambil berdasarkan perhitungan *Encryption time*, *decryption time*, pemeriksaan keamanan dari hasil enkripsi dan pengujian hasil enkripsi kedalam *database* adalah sebagai berikut:

a. Waktu yang dibutuhkan untuk enkripsi data tergantung dari jumlah karakter yang ada pada *plaintext*, semakin banyak karakter yang ada pada plaintext maka akan semakin lama waktu yang dibutuhkan untuk enkripsi data.
b. Setiap kali melakukan enkripsi dengan *plaintext* yang bersifat ganjil, maka akan ada proses penambahan 1 huruf dibelakang *ciphertext*, sehingga *output* dari hasil enkripsinya berupa *ciphertext* yang bersifat genap.
c. Dekripsi tidak bisa dilakukan pada *ciphertext* yang memiliki jumlah karakter yang bersifat ganjil.
d. Dekripsi hanya bisa dilakukan pada *ciphertext* yang memiliki jumlah karakter yang bersifat genap.
e. *Ciphertext* yang dihasilkan dari penggabungan algoritma *Caesar cipher* dan *hill cipher* tidak dapat dipecahkan oleh *tools online* maupun dengan *software Cryptool 2*.
f. Algoritma *Caesar cipher* lebih efisien digunakan jika digabungkan dengan

algoritma lain. Jika hanya menggunakan *Caesar cipher* saja maka *ciphertext* masih dapat dipecahkan.

g. Hasil dari proses enkripsi dapat diinput ke dalam database sehingga jika ingin mengimplementasikan algoritma ini pada program penyimpanan yang memiliki *database* sangatlah mungkin. Sebaliknya untuk proses dekripsi dapat langsung dilakukan menggunakan *ciphertext* yang tersimpan pada *database*.

SARAN

Berdasarkan pengerjaan dan pengujian yang telah dilakukan, terdapat saran untuk peneliti selanjutnya agar dapat mengembangkan dari kekurangan yang terdapat dalam penelitian ini. Saran tersebut adalah:

a. Enkripsi oleh algoritma gabungan ini sebaiknya dapat dilakukan pada gambar dan suara.
b. Dapat membuat GUI dengan bahasa java yang lebih lengkap.

DAFTAR PUSTAKA

[1] Munir, R. 2019. Kriptografi Edisi Kedua

[2] Fajri, G. R., *et al*. 2020. Keamanan Data Pada Pengarsipan Surat Menggunakan Metode Kriptografi Klasik Vigenere Cipher Dan Shift Cipher, Jurnal Zonasi, 61-72.