Penerapan Keamanan Basis Data Dengan Menggunakan Kombinasi Teknik Enkripsi SHA Dan Knapsack

Nurmi Hidayasari[1], Kasmawi[2], Mansur[3]
Jurusan Teknik Informatika Politeknik Negeri Bengkalis[1, 2, 3]
nurmihidayasari@polbeng.ac.id[1], kasmawi@polbeng.ac.id [2], mansur@polbeng.ac.id [3]

Abstract

The security of data storage in a database is an important thing that must be considered and managed by an agency and a certain way is needed so that data is protected from various kinds of crimes. No exception from the people who deal directly with the database. Cryptography is a technique that can be used to secure data by encoding (encrypting) data in a certain way. The goal is to protect the confidentiality of the data, making it more secure. Cryptography consists of two processes, namely the encryption process and the decryption process. Encryption is done to make data unreadable by converting data into random codes, while decryption is the opposite, which is the process of changing data in the form of random codes into data that can be read by certain parties. Cryptography provides many algorithms that can be used for the encryption process. This research combines the SHA 1 and Knapsack encryption algorithms in securing or encrypting the database, especially on user data. The goal is to improve the security of application systems and databases from various crimes. Including crimes that the database administrator may have committed himself. By doing this research can contribute in improving the security of database systems.

Keywords : Cryptography, Encryption, Knapsack, SHA

1. PENDAHULUAN

Seiring berkembangnya teknologi komputer, mendorong setiap instansi untuk terus meningkatkan kemampuan mereka dalam mengelola data dan informasi agar lebih aman, akurat dan efisien. Terutama keamanan penyimpanan data pada sebuah basis data, yang mana dibutuhkan cara tertentu agar data terhindar dari berbagai macam tindak kejahatan. Tidak terkecuali dari orang-orang yang berhubungan langsung dengan basis data. Salah satu teknologi atau metode yang bisa digunakan untuk mengamankan data adalah kriptografi. Kriptografi merupakan sebuah metode yang melakukan pengkodean (enkripsi) pada sebuah data dengan Teknik tertentu. Tujuannya agar daya tersebut terlindungi kerahasiaannya. Pada dasarnya kriptofrafi terdiri dari dua proses, yaitu proses enkripsi dan proses dekrispi.

Purwanto dalam penelitiannya menggunakan algoritma kriptografi simetris dan bersifat *stream cipher* sehingga data hasil enkripsi (*ciphertext*) mempunyai ukuran yang sama dengan data asli (*plaintext*). Teknik kriptografi simetris dipilih karena diharapkan dengan algoritma ini proses enkripsi dan dekripsi data dapat dilakukan dengan waktu yang lebih cepat dibandingkan dengan algoritma kriptografi kunci publik (asimetris) (Purwanto, 2014).

Penelitian lain tentang implementasi algoritma asimetri dan simetri untuk mengamankan data, yaitu menggunakan metode Knapsack dan Spritz. Kesimpulan dari penelitian ini adalah algoritma knapsack memiliki waktu yang lebih cepat dalam memproses enkripsi dan proses dekripsi dari pada algoritma spritz, karena algoritma spritz memiliki proses yang panjang dalam enkripsi dan dekripsi dari pada algoritma knapsack (Mayuni, 2021).

Fungsi hash biasanya digunakan untuk otentikasi atau integritas sebuah data/file, sedangkan pada kriptografi fungsi hash memiliki tambahan kriteria keamanan. Sehingga fungsi hash juga dapat digunakan untuk mengamankan data. Banyak penelitian yang menggunakan fungsi hash untuk melidungi data. Salah satunya adalah penelitian yang dilakukan oleh (Saputra dan Susandri, 2016), yang merancang aplikasi untuk melindungi suatu folder komputer dari

pengguna yang tidak diizinkan dengan metode algoritma genetika untuk mengacak kunci yang digunakan dan metode SHA 1 untuk mengenkripsi kunci tersebut. Penelitian ini telah berhasil menghasilkan aplikasi yang dapat memberikan keamanan terhadap data dalam sutu folder dan telah diuji melalui pengujian *black box*.

Penelitian ini mencoba menggabungkan kombinasi SHA 1 dan algoritma knapsack dalam mengamankan atau melakukan enkripsi basis data, khususnya pada data pengguna. Tujuannya agar dapat meningkatkan keamanan sistem aplikasi dan basis data dari berbagai tindak kejahatan. Termasuk kejahatan yang mungkin dilakukan oleh administrator basis data sendiri. Dengan dilakukannya penelitian ini diharapkan dapat berkontribusi dalam meningkatkan keamanan sistem basis data.

2. TINJAUAN PUSTAKA

2.1 Keamanan Basis Data

Basis data merupakan sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data. Masalah keamanan data sangatlah komplek. Seringkali masalah keamanan data dapat melibatkan aspek hukum, sosial atau etika, kebijakan yang berhubungan dengan pelaksanaan atau terkait dengan pengendalian peralatan secara fisik. Keamanan basis data berkaitan dengan perlindungan terhadap basis data dari ancaman yang disengaja atau pun tidak disengaja, baik dengan menggunakan elemen kontrol peralatan komputasi atau yang tidak.

Analisis untuk keamanan basis data tidak hanya cukup pada layanan yang disediakan oleh DBMS (*Data Base Management System*), tetapi juga mencakup masalah yang terkait dengan basis data serta lingkungannya. Pertimbangan keamanan tidak hanya berlaku untuk data yang terdapat dalam basis data saja. Karena kesenjangan keamanan pada bagian lain dapat mempengaruhi sistem yang pada waktunya dapat mempengaruhi keamanan basis data. Sehingga, semua bagian dari sistem harus tetap diperhatikan keamanannya, seperti sistem operasi, jaringan komputer, bangunan di mana basis data berada secara fisik serta orang-orang yang memiliki kesempatan untuk mengakses sistem (Susilo, 2016).

2.2 Kriptografi

Kriptografi berasal dari bahasa Yunani, "*cryptos*" yang artinya "*secret*" (rahasia). Sedangkan "*graphein*" berarti "*writing*" (tulisan). Jadi secara harfiah kriptografi berarti "tulisan rahasia" (Munir, 2019)**.**

Dalam kriptografi akan sering ditemukan beberapa istilah. Beberapa istilah tersebut dapat juga disebut terminologi kriptografi, yaitu (Munir, 2019) :

1. *Plaintext*, merupakan pesan atau data berupa teks biasa yang dapat dibaca.
2. *Chipertext*, merupakan pesan atau data yang sudah berubah menjadi kode-kode yang tidak bisa dibaca begitu saja.
3. Enkripsi, merupakan proses merubah p*laintext* menjadi *chipertext*.

Dekripsi, merupakan kebalikan dari enkripsi yaitu mengembalikan *chipertext* menjadi *plaintext*.

Secara umum proses kriptografi sangat sederhana. Sebuah *plaintext* (m) akan dilewatkan pada proses enkripsi (E) sehingga menghasilkan suatu *ciphertext* (c). Kemudian untuk memperoleh kembali *plaintext*, maka *ciphertext* (c) melalui proses dekripsi (D) yang akan menghasilkan kembali *plaintext* (m) (Purwanto, 2014). Berdasarkan waktu penemuannya, kriptografi terbagi menjadi dua, yaitu kriptografi klasik dan kriptografi modern. Sedangkan berdasarkan teknik dasarnya terbagi dua, yaitu simetris dan asimetris.

2.3 Algoritma Knapsack

Algoritma knapsack mendasarkan keamanan dalam sulitnya memecahkan persoalan knapsack (*knapsack problem*). Knapsack berarti karung atau kantong yang memiliki kapasitas

terbatas. Barang-barang dimasukkan di dalam karung hanya bisa sampai maksimum karung saja. Jika kapasitas barang lebih besar maka harus memilih hanya sebagian dari seluruh barang untuk dimasukkan di dalam karung (Munir, 2019).

Knapsack problem dalam matematika ditulis sebagai berikut: Diketahui bobot knapsack adalah m, diketahui n adalah objek yang masing-masing bobot adalah w_1, w_2, ... , *wn*. Tentukan nilai b_i maka $M = b_1w_1$ *h* b_2w_2 hh *bnwn*, maka b_i bernilai 0 atau 1, Jika $b_i = 1$, maka objek *i* akan dimasukkan ke dalam knapsack, sebaliknya jika b_i bernilai 0, objek *i* tidak dimasukkan. Dalam teori algoritma knapsack termasukke dalam kelompok NP-complete, persoalan yang termasuk NP-complete tidak dapat dipecahkan dalam orde waktu polinominal (Rachmawati dan Candra, 2013).

Algoritma knapsack sederhana adalah mengkodekan pesan sebagaijalan keluar dari persoalan knapsack (Munir, 2019). Algoritma normal knapasack atau algoritma non-superincreasing merupakan bagian algoritma yang sulit darisegi komputasi, karena membutuhkan waktu dalam orde eksponensial untuk memecahkannya. Superincreasing knapsack dapat dirubah menjadi non- superincreasing knapsack dengan menggunakan kunci publik untuk proses enkripsi dan kunci privatdigunakan untuk proses dekripsi. Ide dasar dibalik rancangan enkripsi Merkle-Hellman merupakan subset masalah yang bisa dipecahkan dengan mudah. Masalah yang mudah dipecahkan tersebut disembunyikan dengan cara menyembunyikan barisan suprincreasing dengan melakukan perkalian modulo dan permutasi (Mayuni, 2021).

Algoritma membangkitkan kunci publik dan kunci privat dilakukan dengan cara (Mayuni, 2021):

1. Menentukan barisan superincreasing
2. Dikalikan setiap bagian di dalam barisan tersebut dengan *n* modulo *m*. Modulos *m* merupakan angka yang lebih besar dari pada jumlah semua bagian di dalam barisan, sedangkan penggali *n* seharusnya tidak mempunyai faktor persekutuan pada *m*.
3. Hasil dari jumlah perkalian menjadi kunci publik dan barisan superincreasing awal menjadi kunci privat.

Contoh:

Misalkan barisan superincreasing adalah {4, 7, 14, 30, 60, 117 }, m = 235 dan n = 24 Barisan non-superincreasing (normal) dapat dihitung sebagai berikut:

$4 \times 24 \ mod \ 235 = 96$

$7 \times 24 \ mod \ 235 = 168$

$14 \times 24 \ mod \ 235 = 101$

$30 \times 24 \ mod \ 235 = 15$

$60 \times 24 \ mod \ 235 = 30$

$117 \times 24 \ mod \ 235 = 223$

Jadi, kunci publik adalah {96, 168, 101, 15, 30, 223}, sedangkan kunci privat adalah {4, 7, 14, 30, 60, 117}.

2.4 Fungsi Hash untuk Kriptografi

Dalam kriptografi juga dikenal istilah fungsi hash, yakni sebuah fungsi yang masukannya adalah sebuah pesan dan keluaran sebuah sidik pesan (*message fingerprint*). Sidik pesan juga sering disebut *Message Digest* (MD). Fungsi hash dapat digunakan untuk mewujudkan layanan keutuhan data (Sadikin, 2012).

*Secure Hash Algorithm*256 (SHA-256), dirancang oleh *The National Institute of Standards and Technology* (NIST) pada tahun 2002. SHA-256 menghasilkan *message digest*

dengan panjang 256 bit. SHA-256 merupakan salah satu fungsi *hash* satu arah, karena tidak mungkin menemukan pesan dari *message digest* yang dihasilkan (Sembiring et al., 2019).

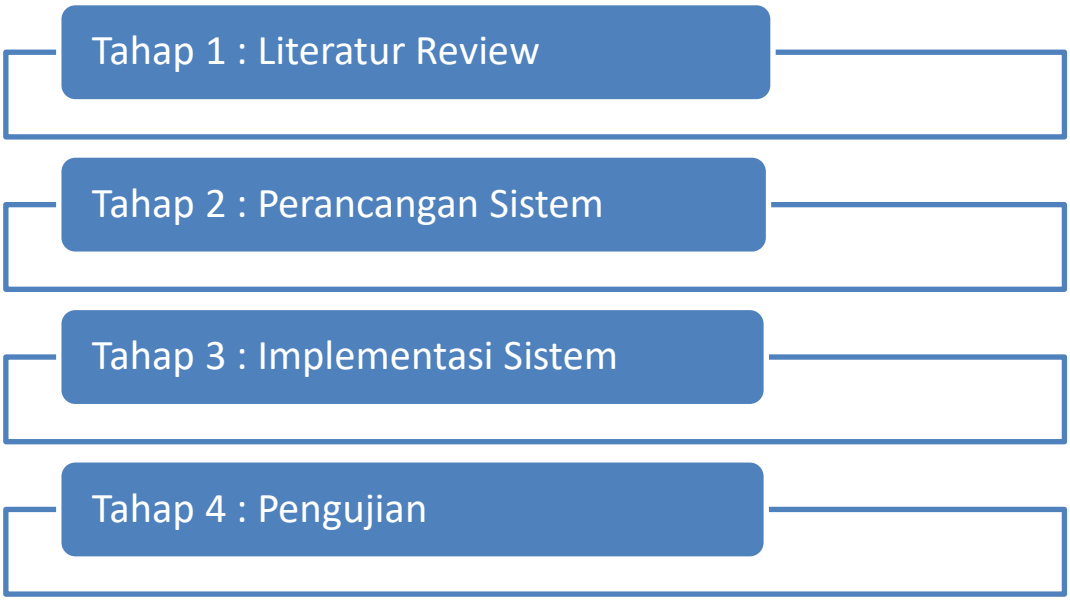
Langkah-langkah pembuatan *message digiset* dengan SHA-256 secara garis besar sebagai berikut :

1. Penambahan bit-bit pengganjalan (*padding bits*).
2. Penambahan nilai panjang pesan semula.
3. Insialisasi nilai *hash* awal.
4. Pengelohan pesan dalam blok berukuran 512 bit.

SHA digunakan untuk menghitung *message diggest* dari pesan atau file data yang disediakan sebagai input. Pesan atau file dianggap sebagai kumpulan bit-bit. Panjang dari pesan adalah banyaknya bit didalam pesan (Pesan kosong memiliki panjang 0). Jika banyaknya bit di dalam pesan merupakan kelipatan 8, untuk memudahkan pembacaan dapat ditampilkan dalam format hexadecimal. Tujuan dari message padding adalah membuat panjang total dari isi pesan menjadi kelipatan 512 bit. SHA secara sekuensial memproses blok 512 bit ketika menghitung message diggest.

Pada message padding, tambahkan satu buah “1” , diikuti oleh m buah “0” diikuti oleh 64 bit integer pada akhir pesan untuk menghasilkan pesan dengan panjang 512 * n. 64 bit integer tersebut adalah panjang dari pesan asli sebelum message padding. Keamanan pada citra menggunakan algoritma *Secure Hash Algorithm* (SHA) berguna untuk menjaga kerahasiaan citra pada saat dienkripsi dengan kunci yang diproses menggunakan algoritma SHA256.

3. METODE PENELITIAN



Gambar 1. Metode Penelitian

Tahap pertama pada penelitian ini, yaitu literatur review untuk mendapatkan informasi serta referensi yang bersumber dari dokumen, buku, makalah, jurnal atau bahan tertulis lainnya. berupa teori, laporan penelitian, atau penemuan sebelumnya, yang bersumber dari media daring maupun luring. Semua informasi yang dikumpulkan merupakan informasi yang relevan dengan penelitian, yaitu terkait keamanan basis data, teknik enkripsi: fungsi hash dan knapsack.

Kedua, perancangan sistem, merupakan tahap merancang sistem keamanan basis data khususnya halaman pengguna. Tahap ini dilakukan beberapa proses perancangan, yaitu : Perancangan Basis Data, Perancangan Halaman Register dan Perancangan Program Enkripsi.

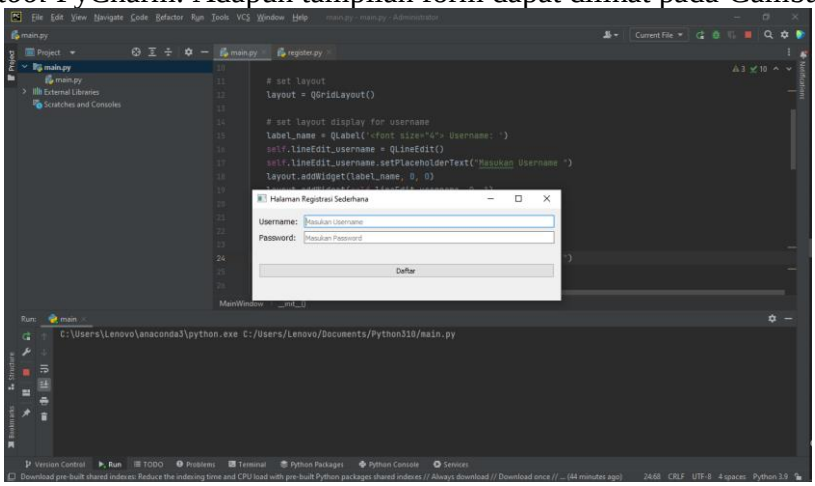
Selanjutnya akan dilakukan tahap implementasi sistem, dengan melakukan percobaan pengamanan basis data halaman pengguna yang telah dirancang pada tahap sebelumnya. Enkripsi akan dilakukan pada data *password* pengguna, sehingga data pengguna yang

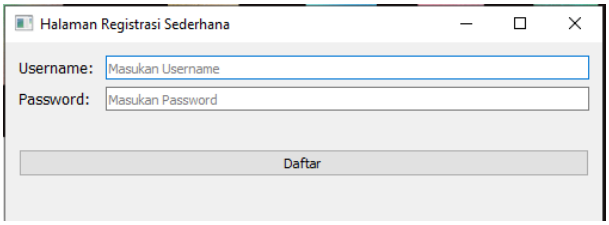
tersimpan di basis data lebih aman dan hanya yang berhak saja yang dapat pengguna pada halaman. Terakhir akan dilakukan pengujian sistem dengan menggunakan *tools* CrypTool versi 2.1 dapat diunduh pada situs *https://www.cryptool.org*.

4. HASIL PENELITIAN DAN PEMBAHASAN

4.1 Hasil Perancangan Sistem

Penelitian ini menggunakan basis data sederhana yang dibuat pada PHPMyAdmin. Isi basis data terdiri dari table pengguna/*user*. Setelah itu, merancang antarmuka sederhana dalam bentuk form pendaftarn (*register*). Pada form diminta untuk memasukkan nama (*username*) dan *password* yang akan didaftarkan. Antarmuka dibangun menggunakan bahasa Python dan dijalankan dengan *tool* PyCharm. Adapun tampilan form dapat dilihat pada Gambar 2.





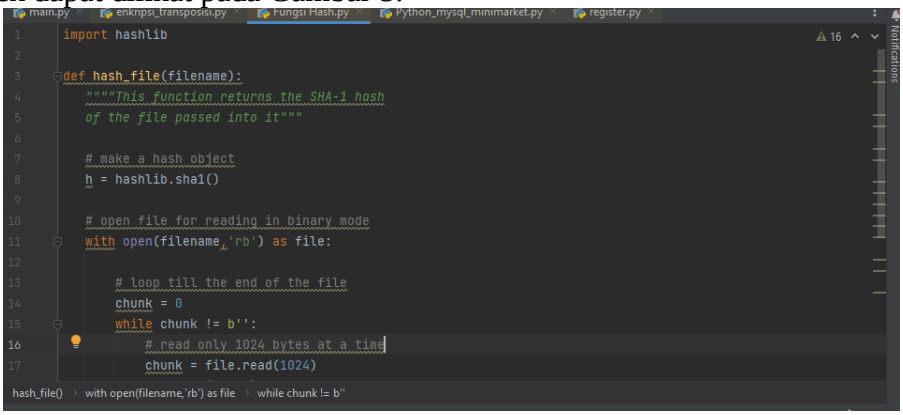
Gambar 2. Tampilan Form Registrasi

Selanjutnya melakukan koneksi basis data dengan *form register* menggunakan modul **MySQLdb** yang ada pada Python. Modul dapat di-*install* dengan perintah **pip install MySQLdb**. Setelah berhasil di-*install*, import modul pada file Python. Kemudian koneksi dapat dilakukan dengan perintah **mysql.connector.connect**. Untuk kode lengkapnya dapat dilihat pada Gambar 3.

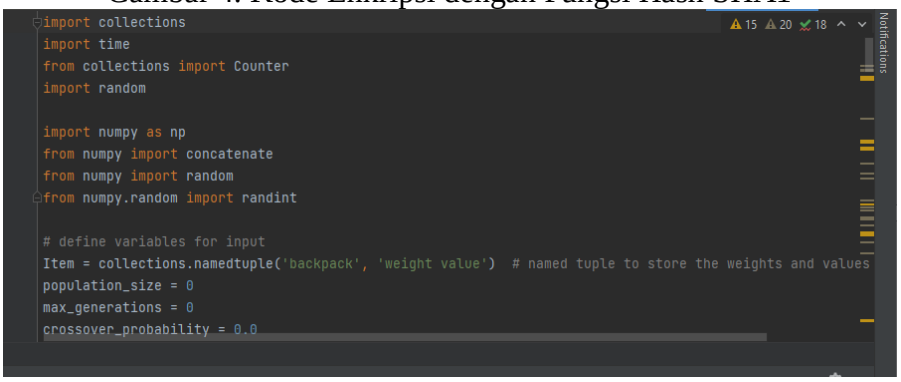
Gambar 3. Koneksi ke Basis Data

4.2 Hasil Implementasi Algoritma

Selanjutnya penerapan dua teknik enkripsi, yaitu SHA dan Knapsack pada form *register* yang telah dibuat sebelumnya. Modul Python yang akan digunakan adalah **hashlib**. Sama seperti cara sebelumnya, modul harus di-*install* terlebih dahulu dengan menggunakan perintah **pip install hashlib**. Kode yang digunakan dapat dilihat pada Gambar 4. Untuk kode enkrispi dengan Knapsack dapat dilihat pada Gambar 5.

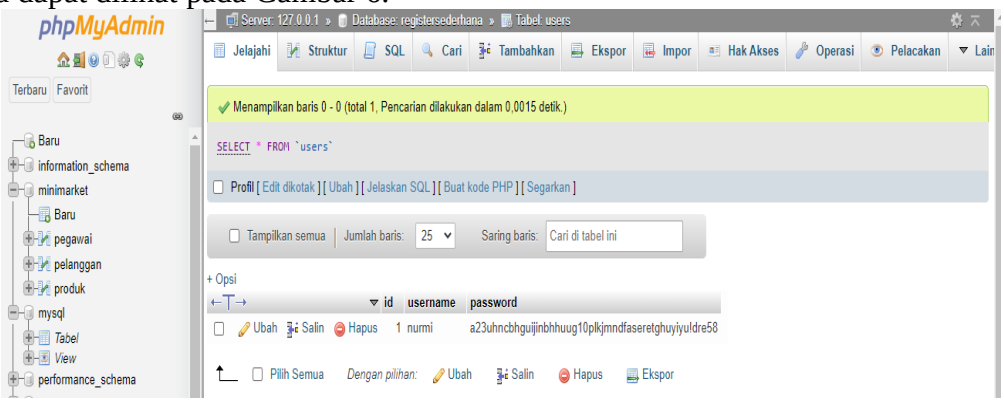


Gambar 4. Kode Enkripsi dengan Fungsi Hash SHA1



Gambar 5. Kode Enkripsi dengan Knapsack

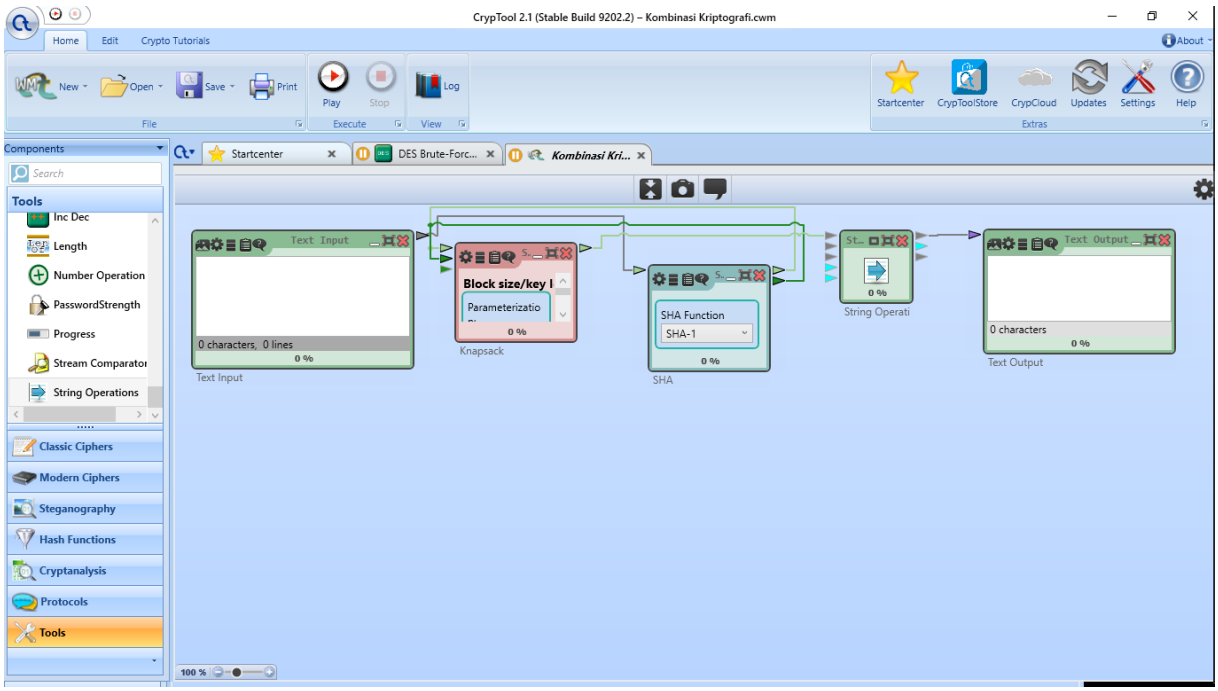
Hasil dari kombinasi dua teknik enkripsi SHA dan Knapsack dapat dilihat pada basis data yang telah dibuat sebelumnya. Data password yang dimasukkan pada form *register* berhasil disimpan pada basis data dalam bentuk kode acak yang merupakan hasil enkrispi. Hasilnya dapat dilihat pada Gambar 6.



Gambar 6. Hasil Enkripsi

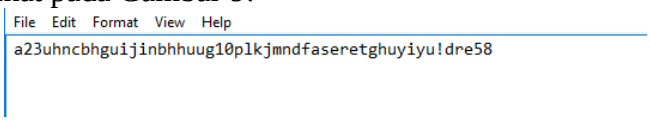
4.3 Pengujian

Berikut merupakan proses pengujian menggunakan *tool* CrypTool 2.1 dan hasil yang didapat. Rancangannya dapat dilihat pada Gambar 7.

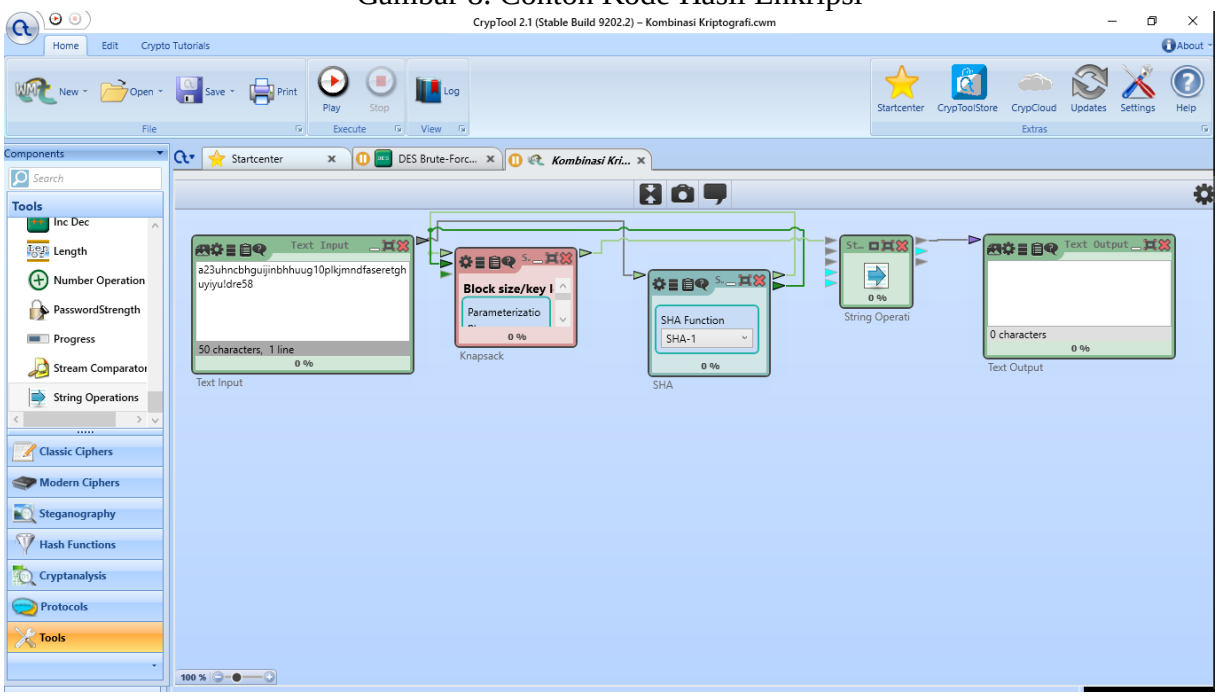


Gambar 7. Rancangan Simulasi Deskripsi dengan Kombinasi SHA1 dan Knapsack pada CrypTool

Pada rancangan simulasi deskripsi di atas terdapat *text input* yang digunakan sebagai tempat untuk memasukkan atau mengetik teks kode hasil enkripsi sebelumnya. Untuk percobaan digunakan hasil enkripsi yang dapat dilihat pada Gambar 8. Dan kode setelah dimasukkan dapat dilihat pada Gambar 9.

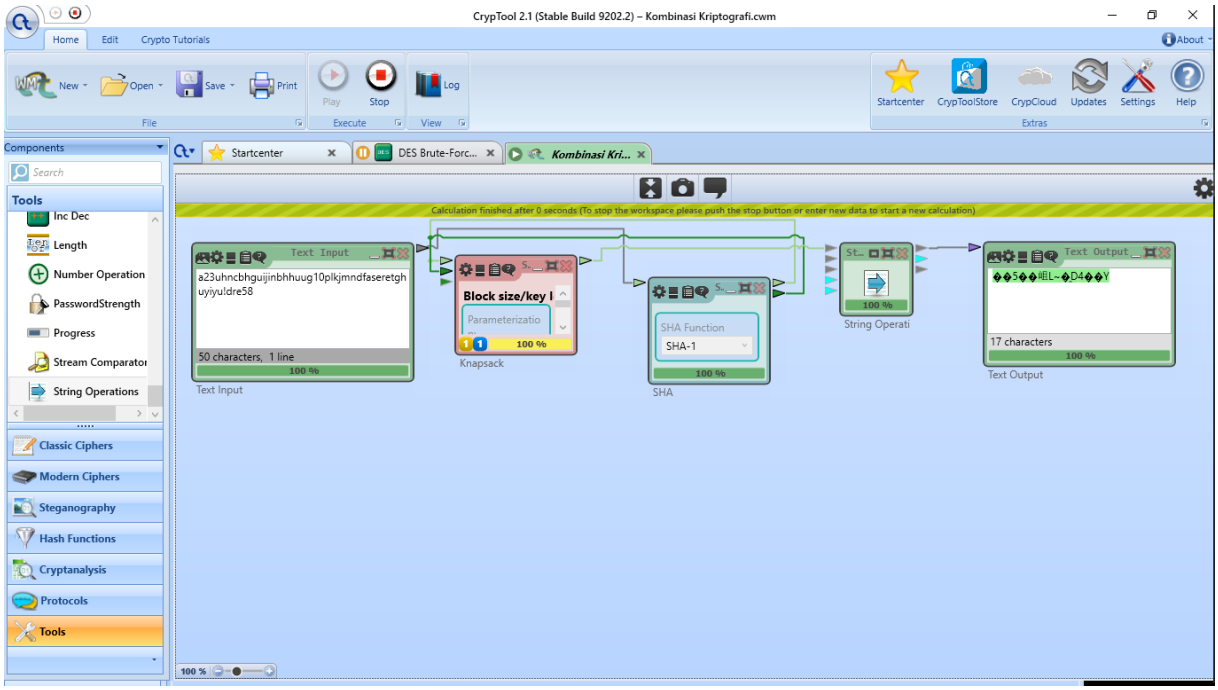


Gambar 8. Contoh Kode Hasil Enkripsi



Gambar 9. Kode Dimasukkan pada *Text Input*

Setelah kode tersebut dimasukkan, maka dapat menjalankan simulasi dengan meng-klik tombol “Play” yang tersedia. Pada Gambar 10. dapat dilihat hasil dari proses deskripsi atau *decode*.



Gambar 10. Hasil Simulasi Deskripsi dengan Kombinasi SHA1 dan Knapsack pada CrypTool

5. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian ini adalah:

1. Penelitian ini berhasil merancang keamanan basis data dengan antarmuka *register* sederhana.
2. Penelitian ini berhasil menerapkan kombinasi teknik enkripsi SHA dan Knapsack.
3. Dari hasil pengujian dengan menggunakan CrypTool 2.1 dapat diketahui bahwa Kombinasi SHA1 dan Knapsack tidak berhasil dideskripsi atau di-*decode* seperti teks asli.

Saran penulis terkait penelitian ini, antara lain:

1. Kedepannya teknik kombinasi ini dapat diterapkan pada sistem informasi dalam bentuk *real*.
2. Kedepannya dapat dilakukan implementasi kombinasi teknik enkripsi lainnya, baik yang klasik maupun modern. Guna meningkatkan dan mengembangkan keamanan sistem basis data.

6. DAFTAR PUSTAKA

Mayuni. (2021). *Implementasi Algoritma Knapsack dan Algoritma Spritz dalam Mengamankan File*.

Munir, R. (2019). *KRIPTOGRAFI (Edisi Kedua)*. Penerbit INFORMATIKA.

Purwanto, H. (2014). Penerapan Keamanan Basis Data Dengan Teknik Enkripsi. *Jurnal Sistem Informasi Universitas Suryadarma*, *1*(1), 12–25. https://doi.org/10.35968/jsi.v1i1.30

Rachmawati, D., & Candra, A. (2013). Implementasi algoritma greedy untuk menyelesaikan masalah knapsack problem. *Jurnal Ilmiah Saintikom*.

Sadikin, R. (2012). *Kriptografi untuk Keamanan Jaringan*. Penerbit ANDI.

Saputra, S., & Susandri. (2016). *Penerapan Algoritma Genetika dan Fungsi Hash SHA 1 pada Sistem Keamanan Dokumen dalam Folder*. *11*(1), 1–8.

Sembiring, H., Manik, F. Y., & Zaidah, T. (2019). Penerapan Algoritma Secure Hash Algorithm

(SHA) Keamanan Pada Citra. *MEANS (Media Informasi Analisa Dan Sistem)*, *4*(1), 33–36. https://doi.org/10.54367/means.v4i1.316

Susilo, G. (2016). KEAMANAN BASIS DATA PADA SISTEM INFORMASI DI ERA GLOBAL. *Jurnal TRANSFORMASI*, *12*(2), 78–87.