PENERAPAN ALGORITMA *BRUTE FORCE* PADA PENEBAKAN DATA YANG DIENKRIPSI DENGAN METODE *MD5*

(Skripsi)

Oleh:

KIKI DIAH WULANDARI

S1 ILMU KOMPUTER
JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG

2019

APPLICATION OF *BRUTE FORCE* ALGORITHM IN ENCRYPTION DATA GUESSING USING *MD5* METHOD

BY

KIKI DIAH WULANDARI

Data security and confidentiality are important aspects in information systems and digital data. Data that contains important information must be kept safe and confidential. Password is one of the important keys for the security information. The password is a determinant for the security data that used to verify its identity in using certain services. Encryption Message Digest5 (MD5) is the process of randomizing passwords. The original messages from the data cannot be read by others. Brute Force Algorithm is an algorithm that can be used to guess passwords which encrypted with MD5. The guessing process is done by encrypting thousands of strings to MD5 so that it gets the same string. Strings are the result of guessed data. This research successfully applied the Brute Force algorithm to encrypt the encrypted data with MD5 method. The result is a utility that can guess passwords number of 1-8 digits and lowercase letters by 1-16 digits.

Keywords*: Algorithme Brute Force, Message Digest 5, MD5, Password, Encryption.*

ABSTRAK

PENERAPAN ALGORITMA *BRUTE FORCE* PADA PENEBAKAN DATA YANG DIENKRIPSI DENGAN METODE *MD5*

OLEH

KIKI DIAH WULANDARI

Keamanan dan kerahasiaan data merupakan aspek yang penting dalam sebuah sistem informasi dan data digital. Data yang berisi informasi penting harus dijaga keamanan dan kerahasiaannya. *Password* merupakan salah satu kunci penting dalam keamaan suatu informasi. *Password* menjadi penentu sebuah kemaanan data yang digunakan untuk memverifikasi identitasnya dalam menggunakan layanan tertentu. Enkripsi *Message Digest*5 (*MD5*) adalah proses pengacakan *password* agar pesan asli dari data tersebut tidak dapat dibaca oleh orang lain. Algoritma *Brute Force* merupakan algoritma yang dapat digunakan untuk menebak *password* yang dienkripsi dengan *MD5*. Penebakan yang dilakukan menggunakan algoritma ini adalah dengan cara mengenkripsi ribuan string kedalam *MD5* sehingga didapatkan string yang sama. String tersebut merupakan hasil dari penebakan data. Penelitian ini berhasil menerapkan algoritma *Brute Force* pada penebakan data yang dienkripsi dengan metode *MD5*. Hasilnya adalah sebuah utility yang dapat menebak *password* berupa angka sebanyak 1-8 digit dan huruf kecil sebanyak 1-16 digit.

Kata Kunci: Algoritma *Brute Force*, *Message Digest*5, *MD5*, *Password*, Enkripsi.

PENERAPAN ALGORITMA *BRUTE FORCE* PADA PENEBAKAN DATA YANG DIENKRIPSI DENGAN METODE *MD5*

Oleh

Kiki Diah Wulandari

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar
SARJANA KOMPUTER

Pada

Jurusan Ilmu Komputer
Fakultas Matematika Dan Ilmu Pengetahuan Alam

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG

2019

Judul Skripsi	:	**PENERAPAN ALGORITMA *BRUTE FORCE* PADA PENEBAKAN DATA YANG DIENKRIPSI DENGAN METODE *MD5***
Nama Mahasiswa	:	Kiki Diah Wulandari
No. Pokok Mahasiswa	:	1517051149
Jurusan	:	Ilmu Komputer
Fakultas	:	Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

1. Komisi Pembimbing

Dwi Sakethi, S.Si., M.Kom.	**Rizky Prabowo, M.Kom.**
NIP. 19680611 199802 1 001	NIP. 19880807 201903 1 011

2. Mengetahui
Ketua Jurusan Ilmu Komputer
FMIPA Universitas Lampung

Dr. Ir. Kurnia Muludi, M.S.Sc.
NIP. 19640616 198902 1 001

MENGESAHKAN

1. Tim Penguji

   Ketua : **Dwi Sakethi, S.Si., M.Kom.** .....

   Sekretaris : **Rizky Prabowo, M.Kom.** .....

   Penguji
   Bukan Pembimbing : **Didik Kurniawan, S.Si., M.T.** .....

Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Dr. Suratman, M.Sc.
NIP. 19640604 199003 1 002

Tanggal Lulus Ujian Skripsi : **28 Oktober 2019**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul "**Penerapan Algoritma *Brute Force* Pada Penebakan Data Yang Dienkripsi Dengan Metode *MD5***" merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang di skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 28 Oktober 2019

Kiki Diah Wulandari
NPM. 1517051149

RIWAYAT HIDUP

Kiki Diah Wulandari, Dilahirkan pada tanggal 5 Desember 1997. Anak tunggal dari pasangan yang bernama Bapak Sukir dan Ibu Maryati.

Penulis menyelesaikan pendidikan Taman Kanak-kanak (TK) Dharma Wanita tahun 2003, menyelesaikan Sekolah Dasar (SD) di SDN Negeri 1 Negara Jaya Kab Way Kanan pada tahun 2009, menyelesaikan Sekolah Menengah Pertama (SMP) di SMPN 01 Negeri Besar Kab Way Kanan pada tahun 2012, kemudian menyelesaikan sekolah di jenjang Sekolah Menengah Atas (SMA) SMA Muhammadiyah 2 Bandar Lampung dan lulus pada tahun 2015.

Pada tahun 2015, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer FMIPA Unila melalui jalur SBMPTN. Selama menjadi mahasiswa, penulis melakukan Praktik Kerja Lapangan pada bulan Januari 2018 di Kantor Televisi Republik Indonesia (TVRI) Lampung. Pada Bulan Juli 2018, penulis melaksanakan Kuliah Kerja Nyata di Desa Gunung Raya Kecamatan Marga Sekampung Kabupaten Lampung Timur.

MOTTO

“Barang siapa yang bersungguh sungguh, sesungguhnya kesungguhan tersebut untuk kebaikan dirinya sendiri"
(Qs. Al-ankabut:6)

“Bersyukur adalah cara paling mudah untuk bahagia”

“Bekerja keras dan bersikap baiklah. Hal luar biasa akan terjadi.”

(Conan O’Brien)

PERSEMBAHAN

Sujud syukurku kusembahkan kepada Allah SWT, Tuhan Yang Maha Agung dan Maha Tinggi. Atas takdirmu saya bisa menjadi pribadi yang berpikir, berilmu, beriman dan bersabar sehingga dapat menyelesaikan skripsi ini. Semoga keberhasilan ini menjadi satu langkah awal untuk masa depan saya dalam meraih cita-cita. Aamiin.

Skripsi ini ku persembahkan kepada orang tua yang luar biasa dalam hidupku. Ayah dan Ibu, terima kasih atas segala doa, kasih sayang, pengorbanan, usaha, dan motivasi yang tiada henti hingga saat ini. Keluarga besar yang selalu menyemangati dan menanyakan kapan skripsi ini akan selesai.

JURUSANKU TERCINTA ILMU KOMPUTER

UNIVERSITAS LAMPUNG

SANWACANA

Puji Syukur kehadirat Allah SWT yang telah melimpahkan rahmat, hidayah serta inayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul "Penerapan Algoritma *Brute Force* Pada Penebakan Data Yang Dienkripsi Dengan Metode *MD5*". Tidak lupa salam kepada Nabi Muhammad SAW, semoga memberikan syafaat kepada umat-Nya di hari kiamat nanti.

Penulis mengucapkan terima kasih kepada semua pihak yang telah membantu dan memiliki peran besar dalam penyusunan skripsi ini, yaitu:

1. Kedua Orang tua, Bapak Sukir dan Ibu Maryati yang telah memberikan doa, semangat, kasih sayang, dan motivasi yang tak terhingga.
2. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc., selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
3. Bapak Didik Kurniawan, S.Si., MT., selaku Sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
4. Bapak Dwi Sakethi S.Si., M.Kom., sebagai pembimbing utama, yang telah memberikan kritik, saran, dan masukan selama masa perkuliahan dan penyusunan skripsi sehingga penulis bisa sampai di tahap ini.
5. Bapak Rizky Prabowo S.Kom., M.Kom., sebagai Pembimbing II Ilmu Komputer yang telah memberikan kritik dan saran yang bermanfaat untuk perbaikan selama pembuatan skripsi ini.

6. Bapak Didik Kurniawan, S.Si., MT., sebagai Pembahas yang telah memberikan kritik dan saran yang bermanfaat untuk perbaikan selama pembuatan skripsi ini.
7. Bapak Drs. Suratman, M.Sc. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
8. Bapak Dwi Sakethi S.Si., M.Kom. sebagai pembimbing akademik yang telah membimbing, memotivasi, memberikan ide, kritik dan saran selama masa perkuliahan.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer yang telah memberikan ilmu yang bermanfaat dan pengalaman hidup kepada penulis selama menjadi mahasiswa.
10. Ibu Ade Nora Maela yang telah membantu segala administrasi penulis di Jurusan Ilmu Komputer.
11. Mas Ardi Nofalian yang selalu memberikan izin tempat untuk melaksanakan seminar dan mengerjakan skripsi.
12. Sahabat terbaik saya sedari bangku perkuliahan, Dwi Tata Mustika, Novella Daria Utami, Kenny Claudie Fandau yang telah menjadi teman canda, tawa, dan duka selama masa perkuliahan.
13. Sahabat terbaik saya sejak Sekolah Dasar, Nilawati dan Rela Ana Anggoro Kasih, Asep Ramansyah yang telah memberikan doa, dukungan serta menjadi tempat berbagi kesedihan dan kebahagiaan di hidup penulis.
14. Sahabat SMA saya, Mia, Uti, Acil, Diah, Meta, Refo, Akbar dan Ridho yang telah memberikan motivasi kepada saya
15. Danu Tri Hartono, yang telah memberikan dukungan serta bantuan dalam mengerjakan skripsi..

16. Teman-teman seperjuangan bimbingan skripsi yang telah berbagi cerita dan ilmu selama proses bimbingan skripsi.
17. Teman-teman ICS Squad yang telah menjadi tempat berbagi bahagia dan keluh kesah selama perkuliahan didalam kelas.
18. Teman-teman CECAN alias Cewe Cantik Ilkomp kelas C yang telah berbagi pengalaman, canda tawa, suka dan duka, serta perjuangan untuk menyelesaikan perkuliahan bersama-sama.
19. Keluarga besar Ilmu Komputer 2015 yang telah memberikan kenangan selama masa perkuliahan.
20. Almamater tercinta, Universitas Lampung yang telah memberikan kesempatan kepada penulis untuk menempuh pendidikan selama perkuliahan jenjang S1 dengan baik.

DAFTAR ISI

Halaman

DAFTAR TABEL

Tabel Halaman

DAFTAR GAMBAR

Gambar Halaman

I. PENDAHULUAN

A. Latar Belakang

Keamanan dan kerahasiaan data merupakan salah satu aspek penting dari suatu sistem informasi dan data digital. Pengamanan sistem informasi atau data digital saat ini sebagian besar menggunakan *username* dan *password*. Pengguna yang ingin mengakses sistem informasi akademik terlebih dahulu harus melakukan *login* sistem dengan cara memasukkan *username* dan *password* untuk keamanan sistem. Data digital yang berisi informasi penting juga biasanya diberikan *password* demi menjaga keamanan data. *Password* adalah kunci yang sangat penting bagi pengguna yang sering melakukan aktifitas di dunia maya, maka *password* tersebut harus diamankan agar hanya orang-orang yang berhak mengaksesnya yang dapat mengetahui maupun menggunakan *password* tersebut.

Enkripsi adalah proses penyandian pesan asli atau *plaintext* menjadi *ciphertext* (teks tersandi) (Hakim & Utami, 2014). Enkripsi adalah proses mengacak keadaan sebuah informasi agar tidak dapat dibaca oleh orang yang tidak memiliki kunci dekripsi dari enkripsi tersebut. Enkripsi dilakukan pada akun ataupun data rahasia (Munir, 2011).

Algoritma enkripsi yang populer saat ini antara lain: AES, *MD5*, SHA, RC4 dan lain-lain. Perbedaan dari enkripsi tersebut adalah dari logika yang digunakan dan jenis data yang diamankan. Algoritma *Message-Digest* 5 atau *MD5* adalah adalah fungsi *hash* kriptografik yang digunakan secara luas dengan *hash value* 128-bit. *MD5* merupakan salah satu perlindungan kepada *user* dalam menggunakan fasilitas internet di dunia maya, terutama yang berhubungan dengan *password*. Keamanan dari sebagian informasi yang hanya dapat diakses oleh orang-orang atau pihak tertentu, namun *user* sering lupa dengan *password* yang sudah dibuat karena sebagian *user* memiliki banyak akun dengan *password* yang berbeda-beda. Penebakan *password* dapat dilakukan secara sistematis dengan teknik *dictionary* yaitu dengan cara mencoba menebak koleksi kata-kata yang umum dipakai, atau yang memiliki relasi dengan *user* yang ditebak (tanggal lahir, nama anak, dsb). Cara lain untuk menebak *password* dapat dilakukan dengan menggunakan algoritma *Brute-force*.

Brute Force adalah sebuah pendekatan yang lempang (*straight forward*) untuk memecahkan suatu masalah, biasanya didasarkan pada pernyataan masalah (*problem statement*) dan definisi konsep yang dilibatkan. Algoritma *Brute Force* memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas. Kekuatan algoritma *Brute Force* terletak pada kemampuannya untuk menemukan semua pemecahan masalah yang mungkin (Munir, 2011). Algoritma *Brute Force* membutuhkan langkah yang sangat banyak karena menelusuri semua kemungkinan penyelesaian masalah, sehingga cenderung menjadi tidak mangkus jika digunakanuntuk memecahkan masalah dengan masukan yang sangat besar.Sebagai contoh, Algoritma *Brute Force* akan

memakan waktu lama jika di dalam sebuah *password* memiliki kombinasi karakter yang banyak, misalnya terdapat kombinasi angka huruf dan simbol karakter tertentu dalam sebuah *password* maka algoritma *Brute Force* membutuhkan waktu yang lama untuk mendapatkan *password*. Namun, jika ingin menggunakan algoritma yang pasti untuk menebak *password* algoritma ini adalah yang paling tepat. Maka pada penelitian ini, akan dicoba implementasi metode *Brute Force* pada proses penebakan *password* yang dienkripsi dengan *MD5*.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah bagaimana menerapkan algoritma *Brute Force* untuk menebak *password* pada data yang telah dienkripsi dengan metode *MD5*.

C. Tujuan Penelitian

Tujuan dari penelitian ini adalah mengembangkan suatu utiliti untuk menebak *password* yang dienkripsi dengan *MD5*.

D. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Menerapkan algoritma *Brute Force* untuk menebak data yang telah dienkripsi dengan *MD5* di mana pada tahapan selanjutnya dapat diterapkan pada berkas yang dipasang *password*.
2. Menambah bahan pustaka bagi peneliti lain yang ingin melakukan penelitian sejenis.

E. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut:

1. Penebakan dilakukan pada data yang telah dienkripsi dengan metode *MD5*.
2. Data yang ditebak pada program hanya berupa angka 0-99999999 dan abjad berupa huruf kecil sebanyak16 digit.
3. Metode penebakan *password* menggunakan algoritma *Brute Force*.

II. TINJAUAN PUSTAKA

A. Algoritma *Brute Force*

Brute Force merupakan algoritma dengan pendekatan yang lempang (*straight forward)* untuk memecahkan suatu persoalan. Algoritma *Brute Force* memecahkan masalah dengan sangat sederhana, langsung, dan jelas (*obvious way*) (Munir, 2014).

Algoritma *Brute Force* adalah algoritma untuk mencocokkan *pattern* dengan semua teks antara 0 dan n-m untuk menemukan keberadaan *pattern* dalam teks (Aan, 2017). Di dalam pencocokkan *string*, terdapat istilah teks dan *pattern*. Teks merupakan kata yang dicari dan dicocokkan dengan *pattern*. Sedangkan *pattern* merupakan kata yang dimasukkan untuk dicocokkan. Secara rinci, langkah–langkah yang dilakukan algoritma ini saat mencocokkan *string* adalah:

1. Algoritma *Brute Force* mulai mencocokkan *pattern* dari awal teks.
2. Dari kiri ke kanan, algoritma ini akan mencocokkan karakter per karakter *pattern* dengan karakter pada teks yang bersesuaian, sampai salah satu kondisi berikut terpenuhi:
   a. Karakter di *pattern* dan di teks yang dibandingkan tidak cocok.
   b. Semua karakter di *pattern* cocok. Kemudian algoritma akan memberitahukan penemuan di posisi ini.

3. Algoritma kemudian terus menggeser *pattern* sebesar satu ke kanan, dan mengulangi langkah ke-2 sampai *pattern* berada di ujung teks.

Serangan brute-force bergantung pada probabilitas. Semakin panjang password, semakin banyak password yang ada untuk memeriksa. Hal ini bergantung pada teori permutasi, yang merupakan susunan angka dalam urutan tertentu. Jadi pikirkan password sebagai anagram. Jika diberi huruf a, b, dan c, berapa banyak susunan memerintahkan berbeda bisa Anda buat? Dengan hanya tiga huruf, dapat dibuat satu set enam permutasi dari himpunan {a, b, c}, yaitu [a, b, c], [a, c, b], [b, a, c], [b, c, a], [c, a, b], dan [c, b, a] (Pramaditya, 2016).

Namun pada kemungkinan password sederhana. Pengulangan diperbolehkan, sehingga rumus untuk jumlah kemungkinan password p untuk ditebak adalah **p** = $\mathbf{x^n}$ dimana **x** adalahjumlah karakter mungkin. Dan **n** adalah panjang password. Jadi pada perhitungan jumlah kemungkinan password yang ada untuk karakter abjad adalah:

Tabel 2.1. Contoh Kombinasi Abjad

Jenis Kombinasi Abjad	Jumlah Kemungkinan Password		
	2 Karakter	4 Karakter	6 Karaker
Abjad Kecil	676	456976	308915776
Abjad Kecil dan Besar	2704	7311616	19770609664
Abjad Kecil, Besar dan Angka	3844	14776336	56800235584
Seluruh Karakter ASCII	8836	78074896	689869781056

Algoritma *Brute Force* juga memiliki kelebihan dan kelemahan. Adapun kelebihan algoritma *Brute Force* adalah sebagai berikut: (Munir, 2014).

1. Metode *Brute Force*dapat digunakan untuk memecahkan hampir sebagian besar masalah (*wide applicability*).
2. Metode *Brute Force* sederhana dan mudah dimengerti
3. Metode *Brute Force* menghasilkan algoritma yang layak untuk beberapa masalah penting seperti pencarian, pengurutan, pencocokan *string*, perkalian matriks.
4. Metode *Brute Force* menghasilkan algoritma baku (standard) untuk tugas-tugas komputasi seperti penjumlahan/perkalian nbuah bilangan, menentukan elemen minimum atau maksimum di dalam Tabel (list).

Adapun kelemahan algoritma *Brute Force* adalah sebagai berikut:

1. Metode *Brute Force* jarang menghasilkan algoritma yang mangkus.
2. Beberapa algoritma *Brute Force* lambat sehingga tidak dapat diterima.
3. Tidak sekontruktif/sekreatif teknik pemecahan masalah lainnya.

Algoritma *Brute Force* memecahkan masalah dengan sangat sederhana, langsung, dan dengan cara yang jelas/lempang. Penyelesaian permasalahan *password cracking* dengan menggunakan algoritma *Brute Force* akan menempatkan dan mencari semua kemungkinan *password* dengan masukan karakter dan panjang *password* tertentu tentunya dengan banyak sekali kombinasi *password*. Karena kesederhanaannya, pengguna hanya tinggal mendefinisikan karakter set yang diinginkan dan berapa ukuran dari *password*-nya. Tiap kemungkinan *password* akan dicoba oleh algoritma ini.

Brute Force merupakan sebuah teknik penyerangan dengan melakukan penebakan, hal ini sering terjadi pada penyerangan *password*. Sebuah *password* yang sudah dienkripsi dengan menggunakan *MD5* juga dapat diretas pada algoritma ini. Tujuan dari *Brute Force*, bukan mencoba mendekripsi *hash MD5*, tetapi untuk mengenkripsi ribuan kata sampai kita mendapatkan *string* yang sama. Misalkan terdapat sebuah *string* dengan panjang 4 karakter yang yang telah di enkripsi *MD5*, maka algoritma ini akan mencoba membandingkan satu per satu kombinasi *string* mulai dari a, b, c untuk kata yang terdiri dari satu digit, jika kata terdiri dari dua digit maka algoritma ini akan mencoba membandingkan kata dari aa, ab, ac dan seterusnya. Apabila salah satu daripada hasil itu memiliki kombinasi yang sama persis dengan *MD5* yang dipecahkan, maka bentuk huruf daripada *MD5* itulah jawabanya.

B. Kriptografi

Kriptografi adalah suatu ilmu yang mempelajari bagaimana cara menjaga agar data atau pesan tetap aman saat dikirimkan, dari pengirim ke penerima tanpa mengalami gangguan dari pihak ketiga. Kriptografi adalah suatu ilmu yang mempelajari bagaimana cara menjaga agar data atau pesan tetap aman saat dikirimkan, dari pengirim ke penerima tanpa mengalami gangguan dari pihak ketiga (Rismayani & Layuk, 2016).

Ada empat tujuan mendasar dari ilmu kriptografi menurut Rismayani (2016) yang juga merupakan aspek keamanan informasi yaitu:

1. Kerahasiaan, adalah layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki otoritas atau kunci rahasia untuk membuka/mengupas informasi yang telah disandi.
2. Integritas data, adalah berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain kedalam data yang sebenarnya.
3. Autentikasi, adalah berhubungan dengan identifikasi/pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keaslian, isi datanya, waktu pengiriman, dan lain-lain.
4. Non-repudiasi., atau nir penyangkalan adalah usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman/terciptanya suatu informasi oleh yang mengirimkan/membuat.

Suatu sistem kriptografi (kriptosistem) bekerja dengan cara menyandikan suatu pesan menjadi suatu kode rahasia yang dimengerti oleh pelaku sistem informasi saja. Pada dasarnya mekanisme kerja semacam ini telah dikenal sejak zaman dahulu. Dalam era teknologi informasi sekarang ini, mekanisme yang sama masih digunakan tetapi tentunya implementasi sistemnya berbeda. Berikut ini

beberapa istilah yang umum digunakan dalam pembahasan kriptografi (Munir, 2011).

1. *Plaintext (message)* merupakan pesan asli yang ingin dikirimkan dan dijaga keamanannya.
2. *Chipertext* merupakan pesan yang telah dikodekan atau disandikan sehingga siap untuk dikirimkan.
3. *Cipher* merupakan algoritma matematis yang digunakan untuk proses penyandian *plaintext* menjadi *chipertext.*
4. Enkripsi *(encryption)* merupakan proses yang dilakukan untuk menyandikan *plaintext* sehingga menjadi *chipertext.*
5. Dekripsi *(decryption)* merupakan proses yang dilakukan untuk memperoleh kembali *plaintext* dari *chipertext.*
6. Kriptosistem merupakan sistem yang dirancang untuk mengamankan suatu sistem informasi dengan memanfaatkan kriptografi.

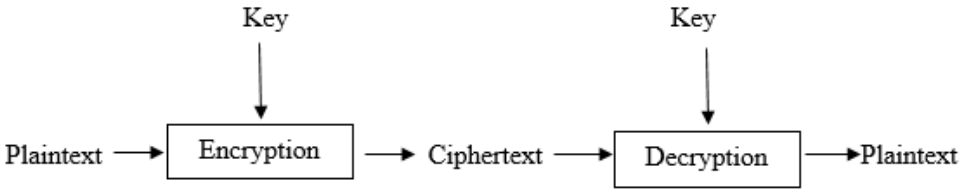
Kriptografi simetri disebut juga sebagai kriptografi konvensional adalah algoritma kriptografi yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Kriptografi simetri sering disebut sebagai algoritma kunci rahasia, algoritma kunci. Kelebihan kriptografi simetri dari kriptografi asimetri adalah lebih cepat (Nugroho, 2018).

1. Enkripsi

Proses utama dalam suatu algoritma kriptografi adalah enkripsi dan dekripsi. Enkripsi merubah sebuah *plaintext* ke dalam bentuk *chipertext*. Pada mode ECB (*Elekctronic Codebook*), sebuah blok pada *plaintext* dienkripsi ke dalam sebuah blok *chipertext* dengan panjang blok yang sama. Blok *cipher* memiliki

sifat bahwa setiap blok harus memiliki panjang yang sama (misalnya 128 bit). Namun apabila pesan yang dienkripsi memiliki panjang blok terakhir tidak tepat 128 bit, maka diperlukan mekanisme *padding*, yaitu penambahan bit-bit *dummies* untuk menggenapi menjadi panjang blok yang sesuai; biasanya *padding* dilakukan pada blok terakhir *plaintext*. *Padding* pada blok terakhir bisa dilakukan dengan berbagai macam cara, misalnya dengan penambahan bit-bit tertentu. Salah satu contoh penerapan *padding* dengan cara menambahkan jumlah total *padding* sebagai byte terakhir pada blok terakhir *plaintext*. Misalnya panjang blok adalah 128 bit (16 byte) dan pada blok terakhir terdiri dari 88 bit (11 byte) sehingga jumlah *padding* yang diperlukan adalah 5 byte, yaitu dengan menambahkan angka nol sebanyak 4 byte, kemudian menambahkan angka 5 sebanyak satu byte. Cara lain dapat juga menggunakan penambahan karakter *end-of-file* pada byte terakhir lalu diberi *padding* setelahnya.

(Bahri, Diana, & Dian, 2012).



Gambar 1. Enkripsi dan Dekripsi (Rismayani & Layuk, 2016).

2. Enkripsi Klasik

Algoritma kriptografi yang digunakan pada zaman sebelum komputer ada disebut algoritma klasik yang berbasiskan karakter. Proses persandian dilakukan pada setiap karakter pesan. Semua algoritma klasik termasuk ke dalam sistem kriptografi simetris dan digunakan jauh sebelum sistem dan digunakan jauh sebelum sistem kriptografi publik ditemukan.Kriptografi klasik dibagi menjadi dua yaitu *Substitution Ciphers* (Cipher Substitusi) dan *Transposition Ciphers* (Transposisi cipher) (Sadikin, 2012).

a. *Substitution Ciphers* (Cipher Substitusi)

   Sistem kriptografi yang menggunakan operasi substitusi disebut dengan sistem substitusi. Prinsip utama cipher substitusi yaitu mengganti munculnya sebual simbol dengan simbol lain. Sistem kriptografi yang berbasis substitusi diantaranya adalah *Shift Cipher* (Caisar Cipher), *Vigenère Cipher* dan *Hill Cipher* (Sadikin, 2012).

b. *Transposition Ciphers* (Cipher Transposisi).

   Algoritma ini melakukan transpose terhadap rangkaian karakter di dalam teks. Pada Cipher transposisi huruf-huruf pada *plaintext* tetap sama hanya urutannya yang diubah. Cipher transposisi dikenal dengan metode permutasi atau pengacakan (*scrambling*) karena transpose setiap karakter di dalam teks sama dengan mempermutasikan karakter-karakter tersebut. Cipher Transposisi dapat dikelompokkan ke dalam dua jenis yaitu sandi transposisi *columnar* dan sandi permutasi (Sadikin, 2012).

3. Enkripsi Modern

Enkripsi modern sangat berbeda dengan enkripsi konvensional, perbedaan ini di awali dengan perangkat yang digunakan yakni dengan menggunakan teknologi komputer dalam pengoprasiannya. Enkripsi modern mengamankan data baik yang ditranfer menggunakan layanan jaringan maupun yang tidak. Hal tersebut sangat berguna melindungi privasi, integritas data, *authentication* (pengesahan) dan *nonrepudation*((Mukhtar, 2018).

Cara enkripsi ini mempunyai banyak kelebihan, salah satunya adalah tiap orang hanya perlu memiliki satu set kunci, tanpa perduli berapa banyak orang yang akan diajak berkomunikasi. Jadi jika ada n orang berkomunikasi dengan cara ini, hanya dibutuhkan n set kunci saja. Selain itu cara enkripsi ini tidak membutuhkan saluran aman untuk pengiriman kunci, sebab kunci yang dikirim ini memang harus diketahui oleh publik. Cara enkripsi ini sangat praktis sehingga masyarakat umum dapat menggunakannya dengan mudah (Mukhtar, 2018).

Ada beberapa algoritma yang terkenal dari cara enkripsi ini misalnya: sistem *diffie hellman,* RSA,PGP (*Pretty Good Privacy*), dan *MD5* (*Message Digest*).

C. *Message Digest*5 (*MD5*)

Algoritma *MD5* adalah algoritma fungsi *hash* kelima yangdikembangkan oleh Ronald L. Rivest pada tahun 1992.Algoritma *MD5* dapat melakukan *hash*ing pada pesandengan panjang sembarang (*arbitrary*) menjadi *Message Digest*dengan panjang tetap (fix) sepanjang 128 bit.Algoritma tersebut memiliki

lisensi RSA dan banyakdigunakan oleh masyarakat umum secara luas padaberbagai aplikasi/layanan kriptografi berdasarkan padadokumen RFC 1321.

(Message-Digest algortihm 5) adalah fungsi *hash*kriptografik yang digunakan secara luas dengan *hash* value 128-bit. Pada standart Internet (RFC 1321), *MD5* telah dimanfaatkan secara bermacam-macam pada aplikasi keamanan, dan *MD5* juga umum digunakan untuk melakukan pengujian integritas sebuah berkas(Rismayani & Layuk, 2016).

Langkah-langkah dalam pembuatan *Message Digest*secara garis besar adalah sebagai berikut (Mukhtar, 2018):

1. Penambahan bit-bit pengganjal (*padding bits*).

   Pesan ditambah dengan sejumlah bit pengganjal sedemikian hingga panjang pesan (dalam satuan bit) kongruen dengan 448 modulo 512. Jika panjang pesan 448 bit, maka pesan tersebut ditambah dengan 512 bit menjadi 960 bit. Jadi, panjang bit-bit pengganjal adalah antara 1 sampai 512. Bit-bit pengganjal terdiri dari sebuah bit 1diikuti dengan sisanya bit 0.

2. Penambahan nilai panjang pesan semula.

   Pesan yang sudah diberi bit-bit pengganjal selanjutnya ditambah lagi dengan 64 bit yang menyatakan panjang pesan semula. Jika panjang pesan lebih dari 264 maka yang diambil adalah panjangnya dalam modulo 264. Dengan kata lain jika panjang pesan semula adalah K bit, maka 64 bit yag ditambahkan menyatakan modulo 264. Sesudah ditambah dengan 64 bit sekarang panjang pesan menjadi kelipatan 512 bit.

3. Inisialisasi penyangga (*buffer*) MD.

   MD5 membutuhkan 4 buah penyangga yang masing-masing panjangnya 32 bit. Total panjang penyangga adalah 4*32= 128 bit. Keempat penyangga ini diberi nama A, B, C dan D. Setiap penyangga diinisialisasi dengan nilai-nilai (dalam notasi HEX) sebagai berikut:

   A: 01234567

   B: 89ABCDEF

   C: FEDCBA98

   D: 76543210

4. Pengolahan pesan dalam blok berukuran 512 bit.

   Pesan dibagi menjadi L buah blok yang masing-masing panjangnya 512 bit (Y0 sampai YL -1). Setiap blok 512 bit diproses bersama dengan penyangga MD menjadi keluaran 128 bit, dan ini disebut proses H*MD5*.

   Fungsi *hash* satu-arah (One-way *Hash*) adalah fungsi *hash* yang bekerja dalam satu arah, pesan yang sudah diubah menjadi *Message Digest*tidak dapat dikembalikan lagi menjadi pesan semula. Dua pesan yang berbeda akan selalu menghasilkan nilai *hash* yang berbeda. Sifat-sifat fungsi *hash* satu-arah adalah sebagai berikut :

   a. Fungsi H dapat diterapkan pada blok data berukuran berapa saja.
   b. H menghasilkan nilai (h) dengan panjang tetap (*fixed-length output*).
   c. H(x) mudah dihitung untuk setiap nilai x yang diberikan.
   d. Untuk setiap h yang diberikan, tidak mungkin menemukan x sedemikian sehingga H(x)=h.

e. Untuk setiap x yang diberikan, tidak mungkin mencari y≠x sedemikian sehingga H(y)= H(x).
f. Tidak mungkin (secara komputasi) mencari pasangan x dan y sedemikian sehingga H(x)= H(y).

D. *Password*

Password adalah kumpulan karakter atau *string* yang digunakan oleh pengguna untuk memverifikasi/mengotentikasi identitasnya dalam menggunakan aplikasi atau layanan tertentu, dimana untuk menjaga kerahasiaan dan kemungkinan penyalahgunaan oleh pihak lain, maka *password* harus diganti secara periodik. Definisi lain dari *password* adalah kata kunci yang dirahasiakan nilainya, tujuannya adalah untuk mengamankan data atau akses pengguna terhadap program dari orang-orang yang tidak berhak (Nugroho, 2018).

Password adalah suatu bentuk dari data autentikasi rahasia yang digunakan untuk mengontrol akses kesuatu sumber informasi. *Password* dirahasiakan dari mereka yang tidak diizinkan untuk mengakses. Tujuan pembuatan *password* adalah demi keamanan. *password* yang kuat akan membuat orang yang tidak berhak sulit untuk membuka *password*. Karena itu, *password* digunakan untuk mengakses dokumen *file,* halaman web, *account* yang dilindungi (Enterprise, 2010).

Secara umum *password* merupakan garis pertahanan pertama untuk melawan akses dari pengguna yang tidak berkepentingan, oleh karena itu keamanan *password* harus senantiasa dijaga dengan baik, salah satu cara untuk menjaga keamanan *password* adalah dengan menerapkan kebijakan manajemen *password* yang baik (Nugroho, 2018) antara lain:

1. Mengganti *username* dan *passworddefault* yang dibangkitkan oleh sistem secara otomatis pada saat *login* pertama kali kedalam aplikasi, menggunakan *password* yang terdiri dari kombinasi huruf besar, huruf kecil, angka, tanda baca dan karakter khusus, contohnya “P4$$w0rD*!” untuk mempersulit *Brute Forceattack.*
2. Menghindari penggunaan *password* yang berisikan kata-kata umum atau yang terdapat dalam kamus untuk mempersulit *dictionary attack*, menghindari penggunaan *password* yang mengandung karakteristik atau pengenal pribadi seperti nama panggilan, tanggal lahir, nama keluarga dan lain-lain. Namun sebaiknya *password* memiliki sifat mudah diingat, contohnya 13uD!m4N yang mudah diingat dengan menggunakan metode *mnemonic* dari nama Budiman.
3. Menggunakan *password* yang berbeda atau unik pada setiap aplikasi dan/atau layanan jaringan komunikasi, melakukan penyimpanan *password* pengguna pada aplikasi dan/atau tempat yang aman, contohnya aplikasi *passwordsafe,keyPass,password* manager dan aplikasi lainnya, sehingga tidak perlu diingat, tidak mudah dicuri dan tidak dapat disalahgunakan oleh pihak yang tidak berkepentingan.

III. METODOLOGI

A. Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung. Waktu penelitian dilakukan pada tahun 2019.

Tabel 3.1. Jadwal Kegiatan Penelitian.

No	Kegiatan	Apri		Mei				Juni				Juli				Agustus				September				Oktober			
		3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Requirement																										
2	Analisis																										
3	Perancangan																										
4	Penulisan Kode Program																										
5	Seminar Usul																										
6	Pengujian																										
7	Seminar Hasil																										
8	Ujian Komprehensif																										

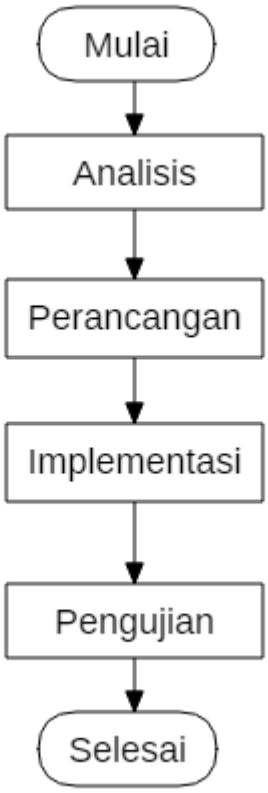
B. Spesifikasi *Hardware* dan *Software*yang Digunakan

1. Perangkat keras *(Hardware)*, yang digunakan pada penelitian ini memiliki spesifikasi sebagai berikut:
   a. Processor Intel Core i3
   b. RAM 4 GB
   c. Hardisk 500 GB

   Perangkat lunak (*Software*) yang dipakai memiliki spesifikasi sebagai berikut:
   a. Sistem Operasi : Windows 10
   b. Bahasa Pemrograman : C++
   c. IDE : Dev C++

C. Tahapan Penelitian

Tahapan penelitian ini dilakukan dengan beberapa langkah yaitu Analisis, Perancangan, Implementasi dan Pengujian. Diagram alir perancangan pada penelitian ini dapat dilihat pada Gambar.31.



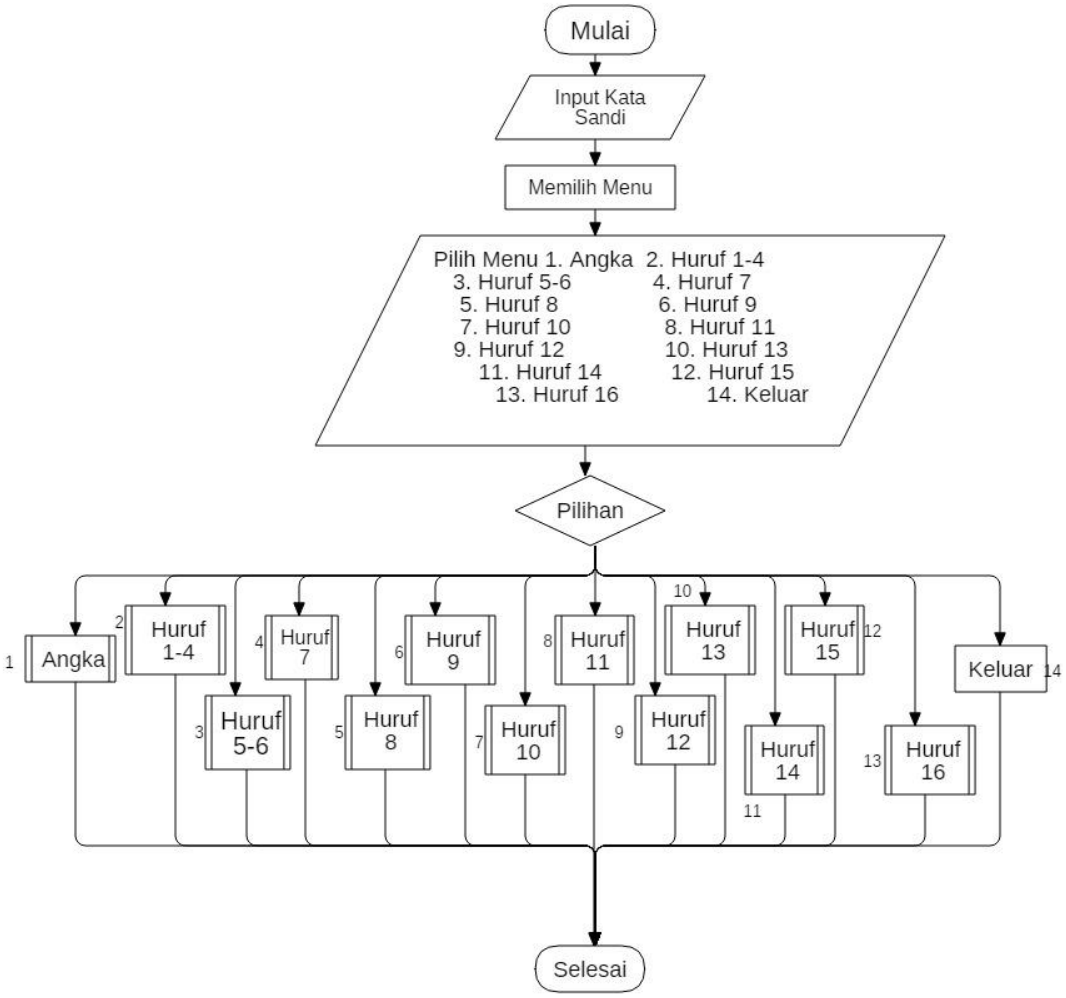
Gambar 3.1. Diagram Alir Penelitian.

1. Analisis

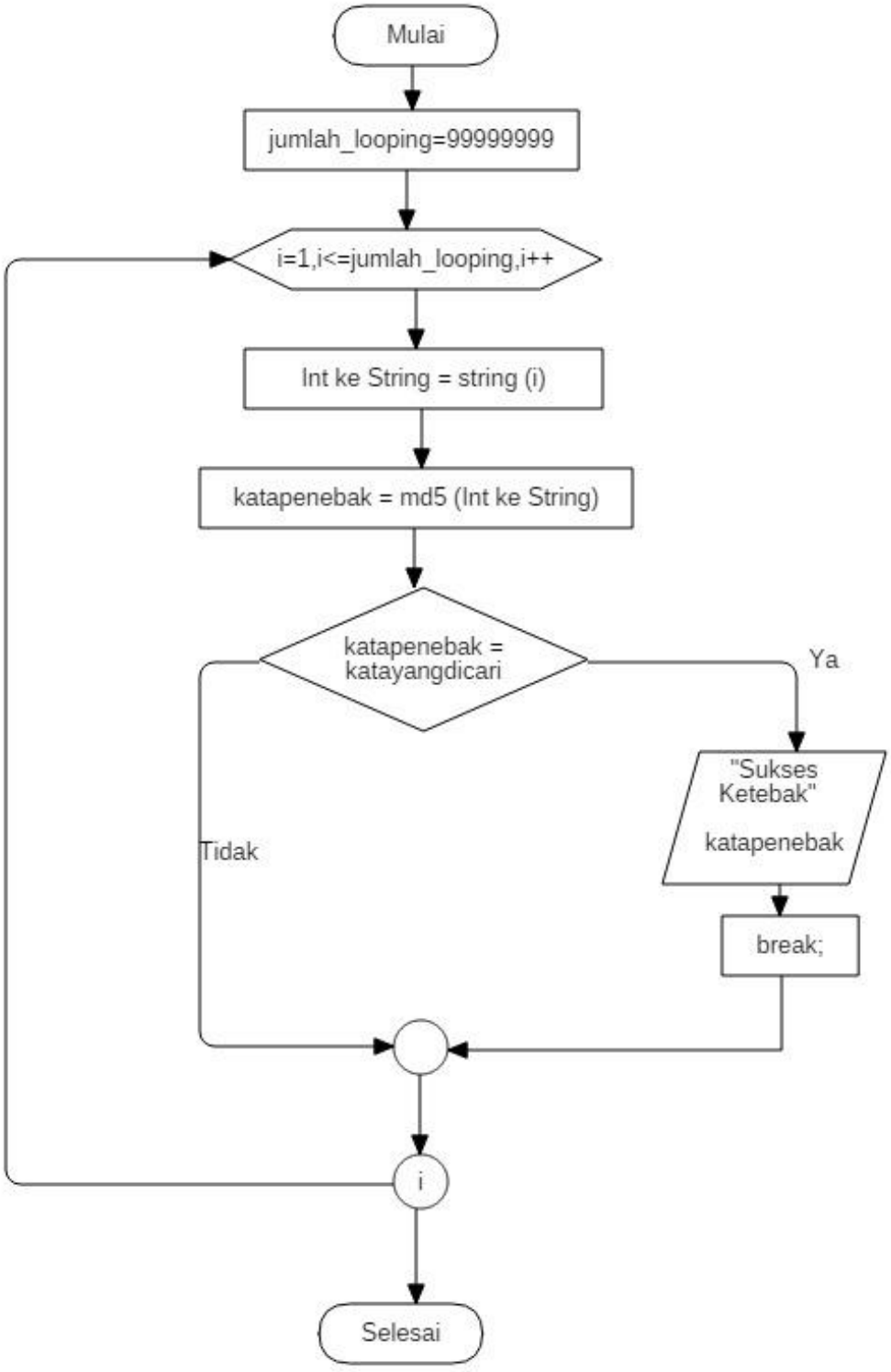
Pada tahap analisis kebutuhan dimulai dengan mengidentifikasi dan mengumpulkan studi literatur mengenai metode-enkripsi khususnya metode algoritma *MD5* dan mengenai penerapan algoritma *Brute Force*dalam kegiatan sehari-hari. Identifikasi masalah pada penelitian ini bertujuan untuk mencari faktor-faktor yang harus diperhatikan untuk menyelesaikan masalah penebakan data menggunakan algoritma *Brute Force*.

2. Perancangan

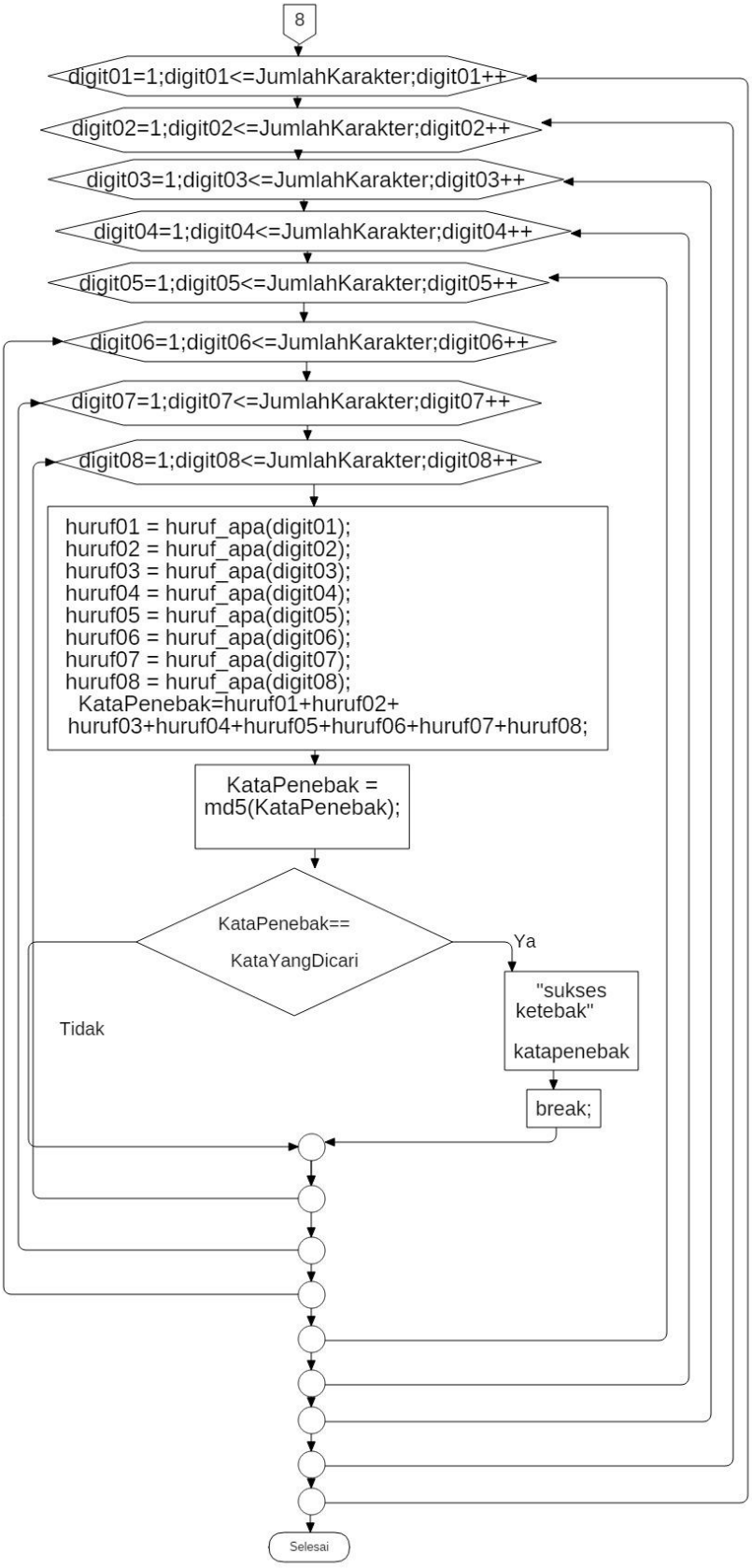
Tahapan kedua pada penelitian ini adalah pembuatan algoritma program untuk penyelesaian masalah yang dirumuskan. Algoritma yang dibuat adalah hasil pengembangan dari algoritma *Brute Force.* Berikut ini adalah *Flowchart* Algoritma *Brute Force* untuk menyelesaikan masalah penebakan data yang dienkripsi dengan *MD5*. *Flowchart* menu penebakan data ditampilkan pada Gambar 3.2.



Gambar 3.2 *Flowchart* Menu Program.



Gambar3.3. *Flowchart* Fungsi Menebak Angka.



Gambar 3.4. *Flowchart* Program Menebak Huruf 8 Digit

3. Implementasi

Pada Tahap ini *flowchart* yang telah dibuat kemudian di implementasi kedalam kode program menggunakan bahasa C++. Proses *coding*dilakukan sesuai dengan urutan *flowchart* yang telah dibuat pada tahap sebelumnya, kode program penebakan data dimulai dengan membuat kode program menebak angka yang kemudian dilanjutkan dengan pembuatan kode program untuk menebak huruf.

4. Pengujian

Tahapan terakhir adalah pengujian program. Tujuan dari pengujian ini adalah untuk memastikan apakah program sudah berfungsi sesuai dengan yang diharapkan. Apabila dari proses pengujian program didapati beberapa kekurangan pada program maka program direvisi ulang untuk memperbaiki program tersebut. Program yang sudah dibuat diuji coba dengan data yang ada. Data pada penelitian ini berasal dari data angka dan huruf yang telah dienkripsi menggunakan metode *MD5*. Pengujian terhadap penebakan data dilakukan dengan memproses apakah hasil penebakan data sesuai dengan data asli sebelum terenkripsi. Proses penebakan data diuji berdasarkan jumlah digit yang dipilih secara acak.

Tabel 3.2. Rencana Pengujian Angka

Enkripsi *MD5*	Angka Asli	Jumlah Digit	Hasil Program	Waktu	Keterangan
	1	1	1		
	3	1	3		
	5	1	5		
	7	1	7		
	9	1	9		
	111	3	111		
	333	3	333		
	555	3	555		
	777	3	777		
	999	3	999		
	11111	5	11111		
	33333	5	33333		
	55555	5	55555		
	77777	5	77777		
	99999	5	99999		
	1111111	7	1111111		
	3333333	7	3333333		
	5555555	7	5555555		
	7777777	7	7777777		
	9999999	7	9999999		
	11111111	8	11111111		
	33333333	8	33333333		
	55555555	8	55555555		
	77777777	8	77777777		
	99999999	8	99999999		

Tabel 3.3. Rencana Pengujian Huruf 1-4 Digit

Enkripsi *MD5*	Huruf Asli	Jumlah Digit	Hasil Program	Waktu	Keterangan
	a	1	a		
	b	1	b		
	c	1	c		
	k	1	k		
	l	1	l		
	m	1	m		
	x	1	x		
	y	1	y		
	z	1	z		

Tabel 3.3. Lanjutan

Enkripsi *MD5*	Huruf Asli	Jumlah Digit	Hasil Program	Waktu	Keterangan
	aa	2	aa		
	ab	2	ab		
	be	2	be		
	de	2	de		
	za	2	za		
	zz	2	zz		
	aaa	3	aaa		
	apa	3	apa		
	itu	3	itu		
	ini	3	ini		
	zzz	3	zzz		
	aaaa	4	aaaa		
	asas	4	asas		
	coba	4	coba		
	diam	4	diam		
	kiki	4	kiki		
	masa	4	masa		
	zzzz	4	zzzz		

Tabel 2.4 Rencana Pengujian Huruf 5-6 Digit.

Enkripsi *MD5*	**Huruf Asli**	**Jumlah Digit**	**Hasil Program**	**Waktu**	**Keterangan**
	aaaaa	5	aaaaa		
	about	5	about		
	brute	5	brute		
	zzzzz	5	zzzzz		

Tabel 3.5. Rencana Pengujian Huruf 7 Digit.

Enkripsi *MD5*	**Huruf Asli**	**Jumlah Digit**	**Hasil Program**	**Waktu**	**Keterangan**
	aaaaaaa	7	aaaaaaa		
	aaazzzz	7	aaazzzz		
	aabacac	7	aabacac		

Tabel 3.6. Rencana Pengujian Huruf 8 Digit.

Enkripsi *MD5*	**Huruf Asli**	**Jumlah Digit**	**Hasil Program**	**Waktu**	**Keterangan**
	aaaaaaaa	8	aaaaaaaa		
	aaaazzzz	8	aaaazzzz		
	aaaakiki	8	aaaakiki		

Tabel 3.7. Rencana Pengujian Huruf 9 Digit.

Enkripsi *MD5*	**Huruf Asli**	**Jumlah Digit**	**Hasil Program**	**Estimasi Waktu**	**Keterangan**
	aaaaaaaaa	9	aaaaaaaaa		
	aaaaazzzz	9	aaaaazzzz		
	aaaabzzzz	9	aaaabzzzz		

V. SIMPULAN DAN SARAN

Penelitian penerapan Algoritma *Brute Force* untuk menebak *password* yang dienkripsi dengan metode *MD5* telah selesai dilakukan. Berikut ini adalah beberapa kesimpulan dan saran yang dapat digunakan sebagai rujukan untuk penelitian selanjutnya.

A. Simpulan

Berikut adalah simpulan yang dapat digunakan oleh peneliti selanjutnya:

1. Program berhasil menebak *password* angka dari 1-8 digit.
2. Program dapat menebak *password* huruf dari 1-16 digit.
3. Semakin banyak jumlah digit pada *password* maka waktu yang dibutuhkan untuk menebak maka akan memakan waktu lebih lama.
4. Proses penebakan angka lebih cepat dibandingkan dengan proses penebakan huruf.
5. *Brute Force* berhasil diterapkan pada utility penebakan *password*.

B. Saran

Berikut adalah beberapa saran yang dapat digunakan oleh peneliti selanjutnya:

1. Program diperluas batasannya agar dapat menebak kombinasi angka dan huruf serta karakter khusus.
2. Program dapat dikembangkan agar dapat diterapkan pada sebuah file yang dipasangi *password* yang terenkripsi dengan *MD5*

DAFTAR PUSTAKA

Aan, M. (2017). Implementasi Algoritma Brute Force Dalam Pencarian Data Katalog Buku. *Informasi Dan Teknologi Ilmiah (Inti)*, *Iii*, 102–103.

Bahri, S., Diana, & Dian, S. P. (2012). Menggunakan Metode Enkripsi Md5 (Message-Digest Algorihm 5). *Jurnal Ilmiah*, *5*(5), 1–15.

Enterprise, J. (2010). *Trik Mengamankan Password*. Jakarta: Elex Media Komputindo.

Hakim, E. L., & Utami, F. H. (2014). *Aplikasi Enkripsi Dan Deskripsi Data Menggunakan Algoritma Rc4 Dengan Menggunakan Bahasa*. *10*(1), 1–7.

Mukhtar, H. (2018). *Kriptografi Untuk Keamanan Data*. Yogyakarta: Deepublish.

Munir, R. (2011). *Kriptografi*. Bandung: Informatika.

Munir, R. (2014). *Algoritma Brute Force Bagian 2*. Itb: Program Studi Informatika Sekolah Teknik Elektro Dan Informatika.

Nugroho, S. C. (2018). *Optimized Dictionary Attack On Md5 Algortihm*. *Issn: 2085*, 24–26.

Pramaditya, H. (2016). *Brute Force Password Cracking Dengan Menggunakan Graphic Processing Power*. *2*, 12.

Rismayani, & Layuk, N. S. (2016). *Pemanfaatan Enkripsi Md5 Pada Keamanan Login Sistem Informasi Manufaktur Pt . Maruki International Indonesia*. 2–3.

Sadikin, R. (2012). *Kriptografi Untuk Keamanan Jaringan.* Yogyakarta: Penerbit Andi.