ISSN 2685 - 4902 (Media Online)
Vol 2, No.01 Mei 2020

JURNAL ILMIAH
INTECH

Information Technology Journal of UMUS

EISSN : 2685-4902
Vol.2, No.01, Mei 2020

Jurnal Ilmiah

INTECH

Information Technology Journal of UMUS

Vol.2, No.01 Mei 2020
Jurnal Ilmiah INTECH

Terbit dua kali dalam setahun, yaitu pada bulan Mei dan November. Jurnal ini berisi artikel hasil pemikiran di bidang pendidikan dasar dan isu-isu pembelajaran pada sekolah dasar.

EDITOR IN CHIEF
Otong Saeful Bachri, S.Kom., M.Kom

MANAGING EDITOR
Harliana, ST., M.Cs

PRINCIPAL CONTACT
Nike Setiati, A.Md.Kom

SUPPORT CONTACT
Arif Wicaksono, S.A.P

MITRA BESTARI (STAFF AHLI)
Dr. Hamdani, ST., M.Cs (Universitas Mulawarman – Kalimantan Timur)
Dr. Heru Ismanto, S.Si., M.Cs (Universitas Merauke – Merauke Papua)
Dr. Agus Qomaruddin Munir, S.T., M.Cs (Universitas Respati - Yogyakarta)
Hartatik, ST., M.Cs (Universitas AMIKOM Yogyakarya – Yogyakarta)
Sri Ngundi Wahyuni, M.Kom (Universitas AMIKOM Yogyakarta)
Andri Syafrianto, M.Cs (STMIK El Rahma – Yogyakarta)
Meri Azmi, M.Cs (Politeknik Negeri Padang – Sumatera Barat)
Robiyanto, M.Kom (STMIK Bina Nusantara Jaya Lubuk Linggau – Sumatera Selatan)
Achmad Fitro, M.Kom (Politeknik NSC Surabaya- Jawa Timur)

PENANGGUNGJAWAB :
Rektor Universitas Muhadi Setiabudi Brebes: Dr. Robby Setiadi, S.Kom., M.M

ALAMAT PENYUNTING:
Program Studi Teknik Informatika, Universitas Muhadi Setiabudi Brebes.
Jalan Pangeran Diponogoro KM 2 Wanasari Brebes – Jawa Tengah 52252. Telp (0283) 6199000

Jurnal Ilmiah
INTECH
Information Technology Journal of UMUS

KATA PENGANTAR

Assalamualaikum Wr, Wb

Puji syukur kehadirat Allah SWT atas anugrahnya sehingga jurnal edisi kali ini dapat terbit. Sebelumnya kami ingin mengucapkan terimakasih banyak kepada dosen/peneliti/profesi yang telah mengirimkan artikelnya kepada dewan redaksi untuk dapat dipublish pada jurnal yang kami kelola. Semua artikel yang masuk kepada dewan redaksi telah melalui proses review oleh mitra bestari dan tim dewan redaksi, segala proses revisi dan redaksional juga telah dilakukan oleh penulis sebelum jurnal ini diterbitkan. Segala bentuk kritik dan saran yang membangun dari pembaca / peneliti yang dikirimkan sangat kami harapkan demi melakukan pembenahan jurnal yang kami kelola. Akhir kata kami menghaturkan terimakasih banyak kepada semua pihak yang sudah terlibat dalam proses penerbitan jurnal ini.

Wassalamualaikum wr wb.

Ketua Dewan Redaksi

DAFTAR ISI

KOMBINASI KRIPTOGRAFI DIFFIE-HELLMAN, MESSAGE-DIGEST 5 DAN RIVEST CHIPER 4

Sandi Fajar Rodiyansyah*[1], Tantri Wahyuni[2], Deden Sukmana[3]
[1,2,3]Program Studi Informatika, Fakultas Teknik Universitas Majalengka-Jawa Barat Indonesia
e-mail : ***[1]rodiyansyah@unma.ac.id**, [2]tantri_wahyuni80@yahoo.co.id

Abstrak

Dalam berkomunikasi atau bertukar informasi didalam jaringan, tidak semua informasi bersifat publik. Tentunya dalam berkomunikasi bisa saja mengandung informasi yang bersifat pribadi atau rahasia sehingga tidak semua orang boleh mengetahuinya. Penelitian ini akan menggunakan tiga algoritma kriptografi, yaitu Diffie-Hellman, Message-Digest 5 dan Rivest Chiper 4. Dimana algoritma DH bertugas untuk menghasilkan kunci rahasia dari P, kunci publik dan kunci privat. Kemudian kunci rahasia yang dihasilkan algoritma DH tersebut dienkripsi menggunakan algoritma MD5, setelah kunci rahasia dienkripsi algoritma MD5, kunci tersebut digunakan sebagai kunci untuk mengenkripsi data teks didalam dokumen menggunakan algoritma RC4. Berdasarkan hasil penelitian didapatkan bahwa waktu yang dibutuhkan untuk dekripsi relatif sedikit lebih lama dibanding proses enkripsi selain itu salah satu faktor yang mempengaruhi waktu pemrosesan adalah ukuran dokumen itu sendiri.
Kata kunci*— kriptografi, Diffie-Hellman, Message-Digest 5 dan Rivest Chiper 4*

Abstract

In communicating or exchanging information in a network, not all information is public. Of course, in communicating it may contain information that is personal or confidential so that not everyone can find out. This research will use three cryptographic algorithms, namely Diffie-Hellman, Message-Digest 5 and Rivest Chiper 4. Where the DH algorithm is tasked to generate the secret key from P, public key and private key. Then the secret key generated by the DH algorithm is encrypted using the MD5 algorithm, after the secret key is encrypted by the MD5 algorithm, the key is used as a key to encrypt text data in the document using the RC4 algorithm. Based on the results of the study it was found that the time required for decryption is relatively slightly longer than the encryption process besides that one factor that affects the processing time is the size of the document itself.
Keywords*— cryptographic, , Diffie-Hellman, Message-Digest 5 dan Rivest Chiper 4*

1. PENDAHULUAN

Dengan adanya Internet berbagai macam layanan komunikasi disajikan seperti web, e-mail, milis, newsgroups dan sebagainya. Dengan semakin maraknya pemanfaatan layanan komunikasi internet tersebut, maka permasalahan baru kian bermunculan apalagi ditambah dengan adanya *hacker* dan *cracker* yang justru menggangu kenyamanan pengguna internet. Banyak pengguna yang merasa terganggu oleh orang-orang yang tidak bertanggung jawab tersebut dengan menyiasatai pengamanan komunikasi yang dikomunikasikannnya, atau menyiasati dengan cara menjaga keaslian dari informasi yang diterimanya[1].

Dalam berkomunikasi atau bertukar informasi didalam jaringan, tidak semua informasi bersifat publik. Tentunya dalam berkomunikasi tersebut bisa saja mengandung informasi yang bersifat pribadi atau rahasia sehingga tidak semua orang boleh mengetahuinya. Namun tidak dapat dipungkiri bahwasannya internet merupakan *Public Network* sehingga memiliki risiko

Submitted: 19 Desember 2019, **Accepted:** 25 April 2020, **Published:** Mei 2020
ISSN: 2685-4902 (online), Website: http://jurnal.umus.ac.id/index.php/intech

yang amat besar untuk diakses oleh orang yang tidak berhak. Dalam hal ini muncul berbagai bidang ilmu pengkodean dalam pengiriman pesan diantaranya Kriptografi, Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan yang dikirim dari satu tempat ke tempat lain. Bidang ilmu kriptografi saat ini mulai digemari oleh banyak orang karena dengan menguasai teknik kriptografi seseorang akan dapat menjaga komputernya sendiri tanpa harus mengeluarkan modal besar jika dibandingkan menyewa atau membeli software khusus keamanan data [1].

Algoritma kriptografi dibagi menjadi tiga bagian berdasarkan kunci yang dipakainya. Diantaranya Algoritma Simetri, algoritma simetri ini menggunakan satu kunci untuk enkripsi dan dekripsinya. Salah satu contoh algoritma yang menggunakan kunci simetri adalah RC4 (*Rivest Chiper* 4). Kemudian Algoritma Asimetri atau yang biasa disebut algoritma kunci publik, merupakan algoritma yang menggunakan kunci yang berbeda untuk enkripsi dan dekripsinya salah satu contoh dari algoritma kunci asimetri adalah DH (*Diffie-Hellman*). Selain algoritma simetri dan asimetri yang selanjutnya adalah Hash Function (fungsi hash) yang sering disebut dengan fungsi hash satu arah (*one-way-function*), *message digest*, *fingerprint*, fungsi kompresi dan *message authentication code* (MAC), merupakan suatu fungsi matematika yang mengambil masukan panjang variabel dan mengubahnya ke dalam urutan biner dengan panjang yang tetap. Salah satu contoh dari hash function yaitu MD5 (Message-Digest 5)[2].

Algoritma RC4 (*Rivest Chiper* 4) merupakan salah satu algoritma kunci simetris berbentuk *stream chiper* yang memproses unit atau input data, pesan atau informasi pada satu saat. Unit atau data pada umumnya sebuah *byte* atau bahkan bit (*byte* dalam hal RC4). Algoritma ini tidak harus menunggu sejumlah input data, pesan atau informasi tertentu sebelum diproses atau menambahkan byte tambahan untuk mengenkrip[3].

Algortima Kriptografi Asimetri atau biasa juga disebut Algoritma kunci publik diterbitkan pertama kali dalam makalah *Diffie* dan *Hellman* yang didefinisikan sebagai kriptografi kunci publik dan biasanya disebut sebagai *Diffie-Hellman Key Exchange* (pertukaran kunci) atau protokol *Diffie-Hellman*. Tujuan dari algoritma ini adalah untuk memungkinkan dua pengguna saling bertukar kunci secara aman, kemudian dapat digunakan untuk enkripsi dan dekripsi pesan. Algoritma ini tidak berdasarkan pada proses enkripsi dan dekripsi, melain lebih kepada proses matematika yang dilakukan untuk menghasilkan kunci rahasia yang dapat disebarkan secara bebas tanpa harus khawatir karena kunci rahasia terebut hanya dapat didekripsi hanya oleh pengirim dan penerima pesan[4].

Algoritma MD5 (*Message-Digest* 5) adalah salah satu dari serangkaian algoritma message-digest yang dirancang oleh Profesor Ronald Rivest dari Massachusetss Institute of Technology (MIT). Ketika kerja analitis menunjukan bahwa pendahulu MD5 yaitu MD4 mulai tidak aman, maka MD5 kemudian dirancang pada tahun 1991 sebagai pengganti dari MD4. Hash MD5 sepanjang 128-bit (16 byte), yang dikenal sebagai intisari pesan, *message digest* secara tipikal ditampilkan dalam bilangan hexadesimal 32-digit. MD5 telah dimanfaatkan secara bermacam-macam pada aplikasi keamanan dan MD5 juga umum digunakan untuk melakukan pengujian ingeritas data[5]. Dengan semakin banyaknya teknik kriptografi yang ada sehingga pada penelitian ini akan dilakukan ujicoba mengkombinasikan algoritma kriptografi *Diffie-Hellman*, MD5 dan RC4 untuk mengamankan atau mengenkripsi informasiyang berupa data teks dan melakukan ujicoba untuk mengetahui berapa lama waktu yang dibutuhkan untuk mengenkripsi data teks.

Penelitian [6] melakukan penelitian dengan hasil bahwa MD5 merupakan sebuah algoritma kriptografi *hashfunction* yang banyak digunakan sebagai *digital signature* dari sebuah file atau sebagai enkripsi password dalam database. Salah satu teknik kriptanalisis yang bisa diterapkan untuk menembus enkripsi MD5 adalah exhaustive key search. Kebutuhan performa komputasi tingkat tinggi dari teknik ini akan diatasi dengan penggunaan dua buah GPU kelas high-end (NVIDIA & AMD), dengan kriptanalisis yang diimplementasikan secara paralel dengan menggunakan bahasa CUDA dan OpenCL. Pengujian dilakukan dengan menggunakan 1 s/d 9 digit random string yang berdasar dari 65 macam karakter. Hasil pengujian menunjukkan sebuah high-end GPU memiliki batas kemampuan kriptanalisis hingga 8 s/d 9 digit random string, dengan waktu kriptanalisis terlama mencapai lebih dari 1 minggu. Sedangkan untuk

perbandingan performansi, OpenCL pada GPU AMD menghasilkan performa terbaik jika dibandingkan dengan CUDA & OpenCL pada GPU NVIDIA.

Sementrara peneliti [7] menghasilkan penelitian bahwa perkembangan metode penyimpanan digital sekarang ini semakin beragam, salah satunya adalah metode penyimpanan berbasis Cloud yang memberikan akses kepada penggunannya untuk menyimpan data di dalam Internet dengan kapasitas penyimpanan yang dapat disesuaikan dengan keinginan penggunanya. Namun, metode Cloud ini memiliki kekurangan yaitu berkaitan dengan masalah keamanan data . Pada penelitian ini akan akan dibahas mengenai keamanan data pada Cloud dengan menggunakan kombinasi algoritma kriptografi Triple DES Algorithm (3DES) dan pertukaran kunci Diffie-Hellman. Sistem yang dibangun adalah aplikasi berbasis desktop yang menyediakan konten untuk mengunggah dan mengunduh file dokumen dari user, yang didalam nya sudah terdapat proses enkripsi dan dekripsi dengan algoritma kriptografi TripleDESAlgorithm (3DES) serta pertukaran kunci user dengan Diffie-Hellman. Penelitian ini bertujuan untuk menganalisa performansi dari algoritma kriptografi Triple DES Algorithm (3DES) pada keamanan file dokumen saat ada proses enkripsi dan dekripsi, avalanche effect, pemakaian daya dan Diffie-Hellman.

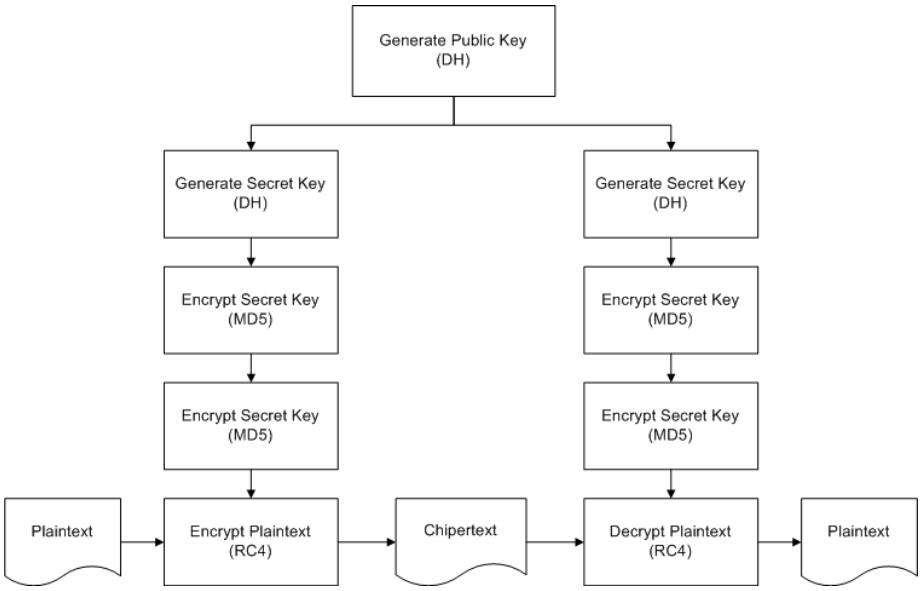
Sementara itu penelitian yang dilakukan peneliti [8] menyebutkan bahwa Kriptografi mempunyai kemampuan untuk mengamankan sebuah pesan. Kriptografi mempunyai aspek keamanan berupa keabsahan pengirim, keaslian pesan dan anti penyangkalan. Aspek keamanan ini dapat diselesaikan dengan teknik autentifikasi yang salah satu caranya dengan menggunakan tanda tangan digital. Tanda tangan digital dapat dilakukan dengan menggunakan fungsi hash. Algoritma message digest 5 adalah salah satu fungsi hash yang digunakan untuk sistem tanda tangan digital. Dalam paper ini, aplikasi tanda tangan digital menggunakan algoritma *message digest* 5 dibangun dengan menggunakan bahasa pemrograman Visual Basic 6.0. Pesan yang dibubuhi tanda tangan digital antara lain surat pemberitahuan dan surat penagihan. Perancangan dilakukan dengan beberapa tahap diantaranya adalah membaca isi dokumen, menyimpan isi dokumen digital, mencari nilai hash dengan algoritma message digest 5, kemudian menyimpan message digest pada dokumen digital. Berdasarkan pengujian yang telah dilakukan, diketahui bahwa sistem dapat membandingkan antara nilai hash yang telah dibuat pada dokumen digital dengan nilai hash dari dokumen digital tersebut. Dari proses perbandingan nilai hash yang merupakan tanda tangan digital untuk dokumen digital dapat diketahui apakah sebuah dokumen digital telah mengalami perubahan atau tidak. Sedangkan dari peneliti [9] menyebutkan bahwa pengujian RC4 dapat dilakukan dengan penjadwalan kunci melalui larik byte yang berukuran 256.

Dari penelitian-penelitian yang disebutkan diatas ada beberapa persamaan antara penelitian terdahulu dengan penelitian ini. Selain ada persamaan yang telah disebutkan diatas tentunya ada juga perbedaannya. Penelitian ini dilakukan dengan mengkombinasikan antara algoritma kriptografi kriptografi simteri (menggunakan kunci yang sama untuk enkripsi dan dekripsi), algoritma yang penulis pilih yaitu algoritma RC4 (*Rivest Chiper* 4). Kemudian algoritma kriptografi asimetri (menggunakan kunci yang berbeda untuk enkripsi dan dekripsinya), penulis memilih algoritma *Diffie-Hellman Key Exchange*. Dan algoritma kriptografi fungsi hash (*one-way-function*), algoritma yang penulis pilih yaitu algoritma Message-Digest 5 (MD5). Dari hasil kombinasi ketiga algoritma tersebut dilakukan uji coba untuk mengenkripsi dan dekripsi data teks plaintext.

2. METODE PENELITIAN

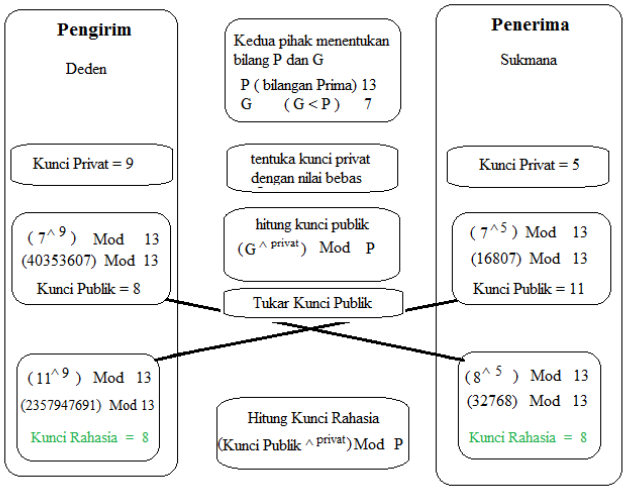
Algoritma kriptografi Diffie-Hellman(DH) merupakan algoritma kriptografi asimetri atau biasa disebut algoritma kunci publik yang mengunakan kunci berbeda untuk enkripsi dan dekripsinya. Namun, algoritma Diffie-Hellman ini tidak berdasarkan proses enkripsi dan dekripsi, melainkan lebih kepada proses matematika yang menghasilkan kunci rahasia simetri (sama). Kemudian, kunci rahasia untuk enkripsi dan dekripsi tersebut penulis enkripsi lagi

menggunakan fungsi hash satu arah yaitu algoritma Message-Digest 5 (MD5). Hasil enkripsi dari MD5 ini tidak bisa didekripsi menjadi plaintext kembali. Selanjutnya kunci rahasia yang telah dienkripsi oleh algoritma MD5 tersebut penulis jadikan sebagai kunci untuk mengenkripsi dan dekripsiplaintext/chipertext menggunakan algoritma simetri yaitu Rivest Chiper 4 (RC4). Kerangka kerja penelitian ini dapat dilihat pada **Gambar 1**.



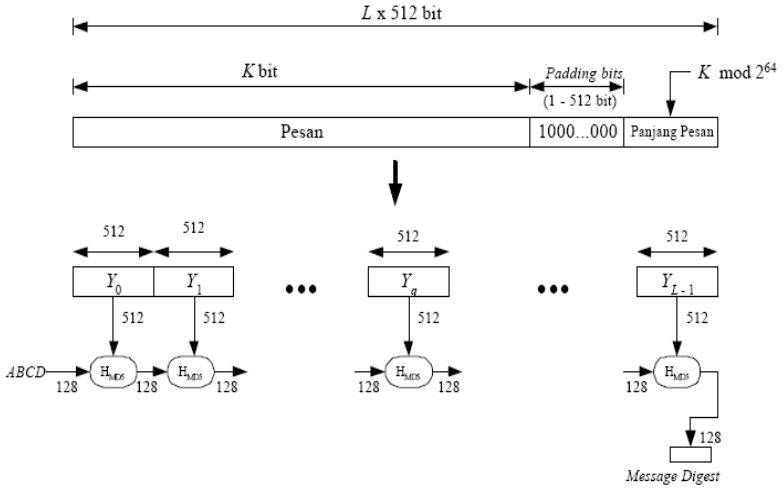
Gambar 1 Kerangka kerja penelitian.

Algoritma *Diffie-Hellman* termasuk kedalam algoritma kriptografi asimetri atau biasa disebut algoritma kunci publik. Algoritma ini tidak berdasarkan proses enkripsi dan dekripsi, melainkan lebih kepada proses matematika yang menghasilkan kunci rahasia, ilustrasi pertukaran kunci *Diffie-Hellman* bisa dilihat pada **Gambar 2**.



Gambar 2 Ilustrasi Pertukaran Kunci Diffie-Hellman

MD5 adalah fungsi hash satu arah yang dibuat oleh Ronald Rivest pada tahun 1991. MD5 merupakan perbaikan dari MD4, setelah MD4 berhasil diserang *cryptanalyst*. Algoritma MD5 menerima masukan berupa pesan dengan ukuran sembarang dan menghasilkan *message digest* yang panjangnya 128bit[10]. Gambaran pembuatan *message digest* dengan algoritma MD5 diperlihatkan pada Gambar 3.

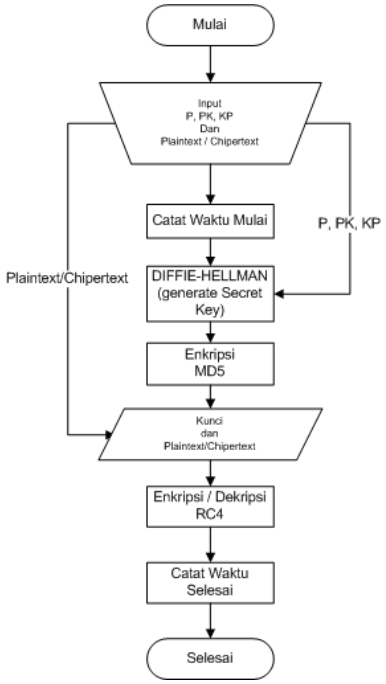


Gambar 3. Pembuatan Message Digest[10]

Seperti yang telah dijelaskan dalam kerangka penelitian, kunci rahasia yang dihasilkan dari algoritma Diffie-Hellman kemudian enkripsi kembali menggunakan algoritma MD5. Pada penelitian ini proses enkripsi MD5 menggunakan fungsi yang disediakan bahasa pemrograman PHP untuk mengenkripsi kunci rahasia yang dihasilkan algoritma *Diffie-Hellman* dengan MD5, yaitu menggunakan fungsi php md5().

Setelah mendapatkan kunci dari algoritma *Diffie-Hellman* dan telah di *enkripsi* menggunakan algoritma *Message-Digest* 5, kemudian digunakan sebagai kunci untuk *enkripsi* dan *dekripsi plaintext* menggunakan algoritma *Rivest Chiper* 4 yang merupakan algoritma kriptografi simetri.

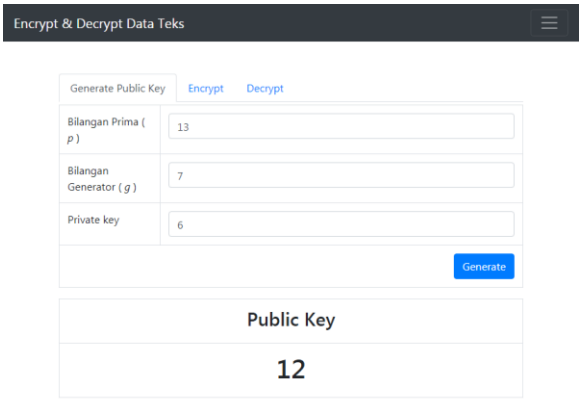
Untuk mengetahui berapa lama waktu yang dibutuhkan tiga algoritma tersebut untuk menghasilkan kunci rahasia (*secret key*), *enkripsi* dan *dekripsi*. Mula-mula user menginput P (*Prime number*) yang telah disetujui kedua pihak sebelumya, PK (*Private Key*) yang di tentukan oleh pengguna sendiri, dan KP (Kunci Publik) yang sebelumnya telah dihitung menggunakan algoritma DH, *Plaintext* atau *Chipertext* dalam bentuk dokumen (.txt) yang diunggah kedalam dalam aplikasi. Setelah input P, PK, KP dan *plaintext/chipertext* berhasil (digunggah). Kemudian dicatat waktu mulai, setelah itu proses pembuatan kunci rahasia (*secret key*) dengan algoritma DH dilakukan, setalah mendapat kunci rahasia (*secret key*) tahap selanjutnya yaitu mengenkripsi kunci rahasia tersebut dengan algoritma MD5. Setelah kunci rahasia tersebut di *enkripsi* menggunakan algoritma MD5 penulis menggunakan kunci tersebut untuk mengenkripsi *plaintext* atau *dekripsi chipertext* menggunakan algoritma RC4, jika proses *enkripsi/dekripsi* selesai maka catat waktu selesai. Ilustrasi proses enkripsi dan dekripsi serta perhitungan lama waktu proses dapat dilihat pada Gambar 4.



Gambar 4 Proses Penghitungan Lama Proses Enkripsi/Dekripsi

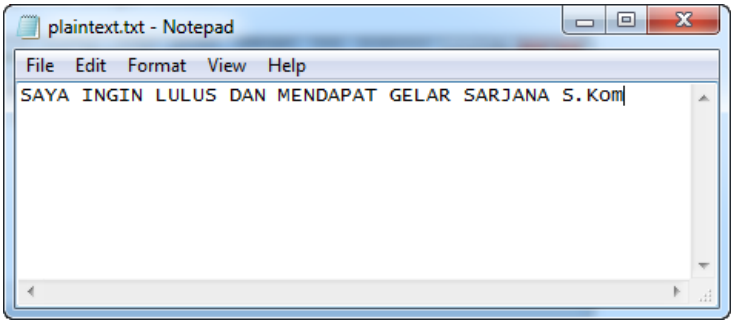
3. HASIL DAN PEMBAHASAN

Langkah pertama yaitu pembuatan kunci publik dengan algoritma *Diffie-Hellman*. Kedua pihak (pengirim dan penerima) menghitung kunci publik dengan menginputkan P (*Prime number*), G (*generator*) yang telah disetujui sebelumnya oleh keduanya dan kunci privat (*private key*) dari masing-masing pihak (pengirim dan penerima).Keduanya telah setuju bahwa P = 13 dan G = 7, kemudian pengirim memilih kunci privat 6, dan penerima memilih kunci privat 4, kemudian keduanya menghitung kunci publik masing-masing, kunci publik yang dihasilkan bisa dilihat pada Gambar 5.



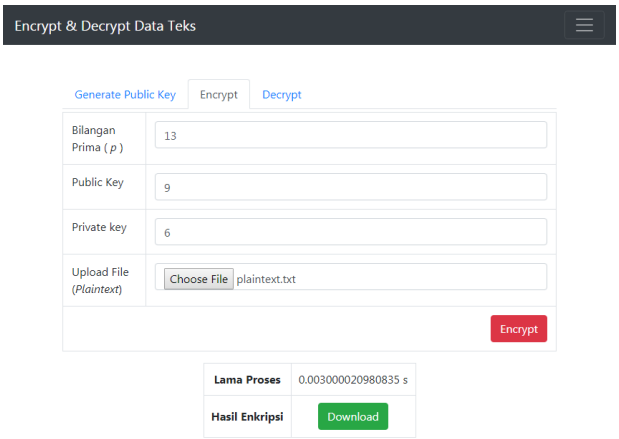
Gambar 5 Proses Menghasilkan Kunci Publik

Setelah kunci publik ditukar maka proses enkripsi bisa dilakukan dengan menginput P (*prime number*), Kunci publik (pengirim), kunci privat (pengirim) dan data teks (*plaintext*) dalam bentuk dokumen dengan ekstensi (.txt). Gambar 6 merupakan contoh dari *plaintext* yang akan dilakukan *enkripsi*.



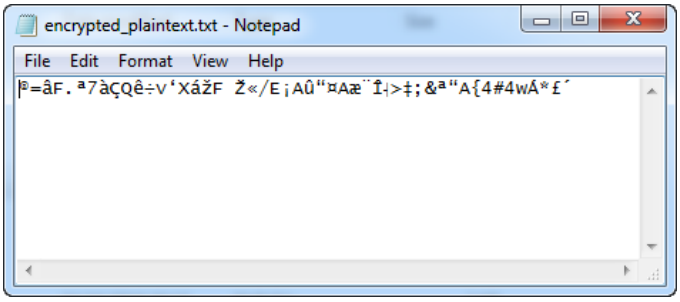
Gambar 6 Isi Dokumen Belum Dienkripsi

Selanjutnya dengan menggunakan P = 13, Kunci publik pengirim = 9, kunci *privat* pengirim = 6, dan pengirim menyisipkan dokumen yang akan dienkripsi. Proses *enkripsi* dapat dilihat pada Gambar 7.



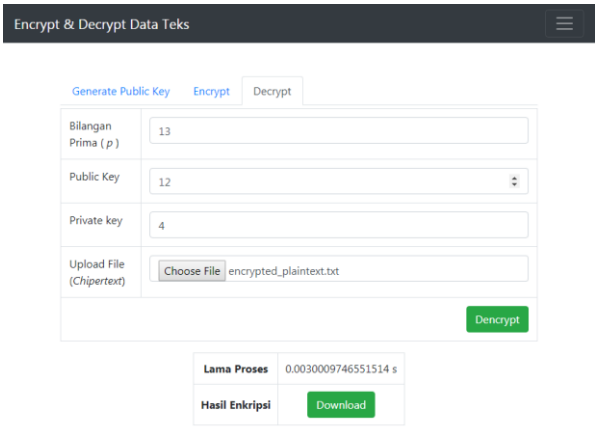
Gambar 7 Proses Enkripsi

Di dalam proses *enkripsi* tersebut menggunakan tiga algoritma kriptografi, yaitu *Diffie-Hellman*, *Message-Digest* 5 dan *Rivest Chiper* 4. Algoritma DH bertugas untuk menghasilkan kunci rahasia dari P, kunci publik dan kunci privat. Kemudian kunci rahasia yang dihasilkan algoritma DH tersebut dienkripsi menggunakan algoritma MD5, setelah kunci rahasia dienkripsi algoritma MD5, kunci tersebut digunakan sebagai kunci untuk mengenkripsi data teks didalam dokumen menggunakan algoritma RC4. Dari ketiga algoritma tersebut menghasilkan sebuah dokumen yang isinya adalah *chipertext* dengan lama proses enkripsi yaitu 0.003000020980835 detik. Adapun isi dokumen yang telah dienkripsi bisa dilihat pada Gambar 8.



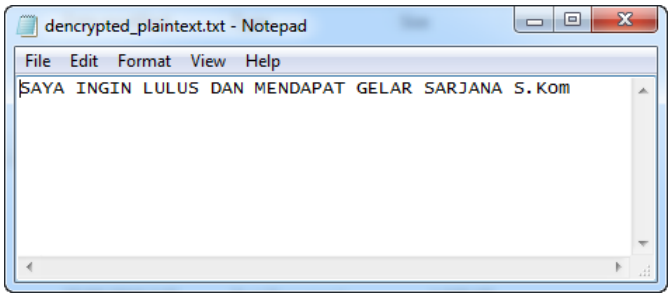
Gambar 8 Isi Dokumen Setelah Dienkripsi

Selanjutnya adalah proses dekripsi yang dilakukan penerima dengan memasukkan P = 13, kunci publik penerima = 12, kunci privat penerima = 4 dan sebuah dokumen yang telah dienkripsi (chipertext). Proses deksipsi dapat dilihat pada Gambar 9.



Gambar 9 Proses Dekripsi

Di dalam proses dekripsi tersebut sama-sama menggunakan tiga algoritma kriptografi, yaitu *Diffie-Hellman*, *Message-Digest* 5 dan *Rivest Chiper* 4. Untuk prosesnya sama dengan pada saat proses enkripsi. Maka dari ketiga algoritma tersebut menghasilkan sebuah dokumen yang isinya adalah plaintext dengan lama proses dekripsi yaitu 0.0030009746551514 detik. Isi dokumen yang telah didekripsi bisa dilihat pada Gambar 10.



Gambar 10 Isi Dokumen Hasil Dekripsi

Dari hasil penelitian dapat diketahui bahwa salah satu variasi kombinasi algoritma kriptografi *Diffie-Hellman*, *Message-Digest* 5 dan *Rivest Chiper* 4 yang peneliti rancang pada kerangka penelitian dapat diterapkan. Namun perlu dicatat bahwa variasi kombinasi ini belum diuji dan dikaji lebih lanjut apakah variasi ini merupakan variasi kombinasi yang efektif dan baik atau bukan. Selanjutnya peneliti telah melakukan beberapa percobaan untuk mengetahui berapa lama waktu yang dibutuhkan untuk enkripsi dan dekripsi data teks menggunakan algoritma kriptografi DH, MD5 dan RC4. Peneliti hanya mencoba mengenkripsi dan dekripsi 5 (lima) dokumen dengan ukuran dokumen berbeda-beda ukuran dokumen, waktu pemrosesan dapat dilihat pada Tabel 1.

Tabel 1 Waktu Enkripsi dan Dekripsi

Ukuran Dokumen	Waktu Enkripsi (DH, RC4, MD5)	Waktu Dekripsi (DH, RC4, MD5)
204.893 bytes	2.3581349849701 detik	2.6181499958038 detik
409.788 bytes	4.2432420253754 detik	4.2852449417114 detik
614.681 bytes	6.1413509845734 detik	6.3953659534454 detik
819.572 bytes	8.0914630889893detik	8.2414720058441 detik
1.051.865 bytes	10.547604084015 detik	11.200640916824 detik

Pada proses enkripsi peneliti menggunakan kunci yang sama yaitu P = 13, kunci publik = 10, dan kunci privat = 1. Pada proses dekripsi peneliti juga menggunakan kunci yang sama

yaitu P = 13, kunci publik = 6, dan kunci privat = 2. Dalam percobaan ini penulis hanya melakukan satu kali proses enkripsi dan dekripsi pada setiap dokumen dan didapatkan data didalam tabel di atas. Bisa dilihat selisih waktu enkripsi maupun dekripsi untuk setiap ukuran dokumen menunjukkan bahwa ukuran dokumen berpengaruh terhadap waktu pemrosesan enkripsi maupun dekripsi, kemudian proses dekripsi relatif sedikit lebih lama dibanding proses enkripsi.

4. KESIMPULAN

Salah satu cara untuk mengkombinasikan algoritma kriptografi DH, MD5 dan RC4 adalah dengan menggunakan algoritma DH untuk menghasilkan kunci rahasia untuk digunakan dalam proses *enkripsi* dan *dekripsi plaintext*. Namun, sebelum kunci rahasia tersebut digunakan untuk proses *enkripsi* dan *dekripsi plaintext/chipetext*, terlebih dahulu dienkripsi menggunakan algoritma MD5, setelah mendapat kunci rahasia dan telah dienkripsi dengan algoritma MD5, kunci tersebut dapat digunakan untuk enkripsi dan *dekripsi plaintext/chipertext* menggunakan algoritma RC4. Dari hasil penelitian, didapatkan waktu tercepat untuk enkripsi yaitu 2.3581349849701 detik dengan ukuran dokumen 204.893bytes, untuk waktu terlamanya adalah 10.547604084015 detik dengan ukuran dokumen sebesar 1.051.865 bytes. Sedangkan waktu untuk dekripsi didapatkan waktu tercepat yaitu 2.6181499958038 detik dengan ukuran dokumen 204.893bytes, untuk waktu terlamanya adalah 11.200640916824 detik dengan ukuran dokumennya 1.051.865 bytes. Dari hasil tersebut dapat disimpulkan bahwa salah satu faktor yang berpengaruh terhadap waktu enkripsi maupun dekripsi adalah ukuran dari dokumen itu sendiri.

DAFTAR PUSTAKA

[1] Mukhtar, H., 2018, *Kriptografi Untuk Keamanan Data*, DEEPUBLISH : Yogyakarta.

[2] Ariyus, D., 2008, *Pengantar Ilmu Kriptografi Teori Analisis dan Implementasi*, ANDI : Yogyakarta.

[3] Haji, W. H., & Mulyono, S., 2012, Implementasi RC4 Stream Chiper untuk Keamanan Basis Data, *Seminar Nasional Aplikasi Teknologi Informasi 2012 (SNATI 2012) Yogyakarta,* [online] available at https://journal.uii.ac.id/Snati/article/view/2954/2726

[4] Yalisa, N., Arhami, M., & Azhar., 2018, Algoritma Elgamal dengan Pertukaran Kunci Diffie Hellman pada Aplikasi Keamanan Citra Sidik Jari Berbasis Android, *Proceeding Seminar Nasional Politeknik Negeri Lhokseumawe,* No.1, Vol. 2, pp. A-1 - A-7, [online] available at http://e-jurnal.pnl.ac.id/index.php/semnaspnl/article/view/736/702

[5] Khairina, D.M., 2011, Analisis Keamanan Sistem Login, *Jurnal Informatika Mulawarman,* No. 2, Vol. 6, pp. 64-67, [online] available at http://e-journals.unmul.ac.id/index.php/JIM/article/view/74/pdf

[6] Alfiansyah, R., Fitriyani, & Ikhsan, N., 2015, Kriptanalisis MD5 dengan Menggunakan Komputasi Kinerja Tinggi, *e-Proceeding of Engineering*, No.2,Vol. 2, pp. 6802-6806, [online] available at https://libraryeproceeding.telkomuniversity.ac.id/index.php/engineering/article/view/3076/2920

[7] Parulian. R. B., Nasution, S. M., & Purboyo, T. W., 2015, Perancangan dan Implementasi Secure Cloud dengan Menggunakan Diffie-Hellman Key Exchange dan Triple DES Algorithm (3DES), *e-Proceedings of Engineering*, No.2,Vol. 2, pp.3808-3815, [online] available at https://libraryeproceeding.telkomuniversity.ac.id/index.php/engineering/article/view/1134/1087

[8] Phungky, D., & Izzuddin, A., 2015, Aplikasi Tanda Tangan Digital (Digital Signature) Menggunakan Algoritma Message-Digest 5 (MD5), *Jurnal ENERGY*, No.1, Vol. 5, pp. 14-19, [online] available at https://ejournal.upm.ac.id/index.php/energy/article/view/155

[9] Prasetyo, T.F., Hikmawan, A., 2016, Analisis Perbandingan dan Implementasi Sistem Keamanan Data dengan Metode Enkripsi RC4 SHA dan MD5, *Infotech Journal*, No.2, Vol. 2, pp. 42-46, [online] available at http://library.palcomtech.com/pdf/6638.pdf

[10] Maryano, B., 2008, Penggunaan Fungsi Hash Satu-Arah untuk Enkripsi Data, *Media Informatika*, No.3, Vol.7, pp. 138-146, [online] available at https://jurnal.likmi.ac.id/Jurnal/11_2008/Penggunaan_Fungsi_Hash_BM_.pdf