NASKAH PUBLIKASI

PERANCANGAN APLIKASI PESAN SINGKAT TERENKRIPSI DENGAN ALGORITMA ADVANCED ENCRYPTION STANDARD (AES) BERBASIS WEB

Program Studi Teknik Informatika

Disusun oleh:

Reza Ahmad Kurniawan

5150411049

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO
UNIVERSITAS TEKNOLOGI YOGYAKARTA

2019

NASKAH PUBLIKASI

PERANCANGAN APLIKASI PESAN SINGKAT TERENKRIPSI DENGAN ALGORITMA ADVANCED ENCRYPTION STANDARD (AES) BERBASIS WEB

Disusun oleh:
Reza Ahmad Kurniawan
5150411049

Pembimbing,

Donny Avianto, S.T., M.T.

Tanggal: 26/8/2019

PERANCANGAN APLIKASI PESAN SINGKAT TERENKRIPSI DENGAN ALGORITMA ADVANCED ENCRYPTION STANDARD (AES) BERBASIS WEB

Reza Ahmad Kurniawan[1], Donny Avianto[2]
[1]*Program Studi Teknik Informatika, Fakultas Teknologi Informasi & Elektro*
[2]*Program Studi Teknik Informatika, Fakultas Teknologi Informasi & Elektro*

Universitas Teknologi Yogyakarta
Jl. Ringroad Utara Jombor Sleman Yogyakrta
Email: rezaahmadk@gmail.com

ABSTRAK

Komunikasi merupakan suatu kegiatan yang tidak dapat dihindari pada kehidupan manusia di seluruh dunia. Manusia adalah makhluk sosial yang tidak akan bisa melakukan segala aktifitas tanpa bantuan yang lain. Untuk mendapatkan suatu bantuan dalam segala bentuk harus dimulai dengan sebuah percakapan. Percakapan dilakukan antara satu orang dengan perseorangan atau kelompok untuk mendapatkan informasi mengenai kebutuhan yang diinginkan. Suatu percakapan dikatakan berhasil jika antar pembicara terjadi kepahaman yang sama mengenai topik pembicaraan. Terdapat banyak cara yang dilakukan untuk menyampaikan sebuah percakapan. Cara yang dilakukan ada dua jenis yaitu secara langsung dan tidak langsung menggunakan suatu media perantara untuk menyampaikan pesan komunikasi. Sejalan dengan perkembangan teknologi, tuntutan untuk dapat melakukan komunikasi tanpa dibatasi oleh tempat dan waktu sangat diperlukan. Penggunaan media perantara pengiriman pesan dengan memanfaatkan teknologi yang dapat diakses dan digunakan oleh banyak orang yaitu jaringan internet. Internet dapat digunakan sebagai sarana komunikasi karena digunakan hampir di seluruh dunia karena kemudahan dalam menggunakannya. Komunikasi melalui internet dapat dilakukan dengan menggunakan perangkat lunak berbasis web. Untuk melakukan sebuah percakapan menggunakan teknologi internet harus memperhatikan faktor ketepatan, kecepatan, kemudahan dan keamanan dalam mengaksesnya. Keamanan merupakan salah satu faktor terpenting, karena suatu perangkat lunak yang melakukan pengiriman pesan berisi informasi dapat diakses dan dibaca oleh pihak lain jika tidak memiliki sistem keamanan yang baik. Menggunakan algoritma enkripsi Advanced Encryption Standard (AES) untuk mengubah informasi pada pesan menjadi suatu susunan kode tertentu sehingga tidak mudah untuk dibaca. Alasan penggunaan algoritma AES karena memiliki aspek penting yaitu fleksibilitas, kesederhanaan dan kesesuaian algoritma untuk keragaman implementasi pada perangkat lunak.

Kata kunci: *Komunikasi, Internet, Keamanan, Enkripsi*

---

1. PENDAHULUAN

1.1 Latar Belakang

Proses komunikasi dapat diartikan sebagai transfer informasi atau pesan (*message*) dari pengirim pesan sebagai komunikator dan kepada penerima sebagai komunikan. Tujuan dari komunikasi adalah untuk saling pengertian (*mutual understanding*) antara kedua pihak yang terlibat dalam proses komunikasi [1]. Proses komunikasi seiring dengan perkembangan teknologi informasi menjadi semakin mudah untuk dilakukan. Teknologi informasi dapat melakukan kegiatan pengelolaan, penyimpanan, penyebaran dan pemanfaatan informasi dengan menggunakan gabungan antara perangkat keras (*hardware*) serta perangkat lunak (*software*). Perangkat keras dan perangkat lunak digunakan untuk meningkatkan kinerja serta memungkinkan berbagai kegiatan dapat

dilaksanakan dengan cepat, tepat serta akurat sehingga mampu meningkatkan produktivitas kerja.

Pada perkembangan teknologi informasi, proses penyampaian informasi semakin dipermudah dengan menggunakan teknologi internet. Internet (*Interconnected Network*) merupakan rangkaian jaringan komputer yang saling terhubung menggunakan TCP/IP (*Transmission Control Protocol/Internet Protocol*). TCP/IP digunakan untuk memberikan alamat dan identitas yang unik pada setiap komputer di seluruh dunia agar terhindar dari kesalahan pengiriman data [2].

Proses pengiriman data menggunakan jaringan internet memiliki beberapa masalah ketika terdapat informasi penting didalamnya. Pada saat informasi dikirimkan tentu memerlukan suatu sistem keamanan yang akan melindungi dari penyalahgunaan pihak tidak bertanggungjawab atau biasa disebut dengan peretas (*hacker*). Peretas saat melakukan aksi peretasan dibagi menjadi tiga jenis yaitu terhadap individu (*against person*), terhadap hak milik (*against property*) dan terhadap pemerintah (*against goverment*).

Cara yang dilakukan untuk mendapatkan informasi tersebut melalui jaringan internet antara lain melalui teknik *session hijacking*, DNS *spoffing* dan *packet sniffing*. Teknik *session hijacking* adalah upaya untuk menyalip sesi yang sudah aktif antara dua *host*. Menggunakan teknik ini, peretas dapat mengambil alih host yang sudah dilakukan autentikasi saat berkomunikasi dengan target sehingga tidak perlu membuang waktu untuk memecahkan kata sandi. Tidak masalah seberapa aman proses autentikasi karena sebagian besar sistem mengirim teks komunikasi yang jelas setelah dikonfirmasi dan menyebabkan sebagian besar komputer rentan terhadap serangan jenis ini [3].

Teknik peretasan berikutnya yaitu DNS *spoofing* yang terjadi ketika peretas mengubah pemetaan nama *domain* ke alamat IP dalam sistem DNS untuk mengarahkan lalu lintas ke sistem jahat atau untuk sekadar melakukan penolakan layanan terhadap suatu sistem [4]. Cara mendapatkan informasi secara ilegal oleh peretas selanjutnya adalah menggunakan teknik *packet sniffing*. *Packet sniffing* adalah teknologi yang digunakan oleh peretas untuk mencegat dan mendekripsi paket data yang mengalir pada jaringan komputer. Seperti yang diketahui, data bergerak dalam bentuk paket di jaringan. Paket-paket ini disebut sebagai data-gram yang memiliki berbagai ukuran tergantung pada *bandwidth* jaringan serta jumlah data yang dibawa dalam paket dalam ukuran *byte*. Setiap paket memiliki label identifikasi yang juga disebut *header*. *Header* membawa informasi dari sumber, tujuan, protokol, ukuran paket, jumlah paket secara berurutan dan jumlah unik paket [5].

Demi mendapatkan informasi rahasia yang lebih banyak, kemungkinan munculnya cara baru dalam peretasan pada internet sangat besar. Untuk melindungi informasi rahasia dari pihak tidak bertanggungjawab, diperlukan sebuah pengamanan terhadap informasi yang dikirimkan. Pengamanan yang dapat dilakukan antara lain menggunakan internet *firewall*, *Secure Socket Layer* (SSL) dan kriptografi. Internet *firewall* merupakan perangkat keras, perangkat lunak atau kombinasi keduanya yang memantau dan melakukan *filter* pada lalu lintas paket yang berusaha masuk atau keluar dari jaringan pribadi yang dilindungi. [6]. Cara pengamanan selanjutnya dengan menggunakan SSL yaitu protokol keamanan standar yang digunakan untuk mengamankan komunikasi antara *server web* dan *browser* dengan mengenkripsi data. SSL akan memastikan data yang dikirimkan dari *browser* ke *server web* telah dienkripsi. Untuk membuat koneksi SSL, terlebih dahulu harus membuat sertifikat SSL dan mengkonfigurasi *server web* untuk melayani di bawah layer SSL [7]. Kemudian cara yang terakhir menggunakan kriptografi untuk mengamankan informasi. Kriptografi adalah ilmu dan seni yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, dan autentikasi [8].

Terdapat dua metode untuk mengamankan informasi menggunakan kriptografi. Metode pertama adalah enkripsi yang dapat mengubah informasi pada data menjadi bentuk kode-kode tertentu sehingga menjadi sulit dipahami. Kemudian pada saat kode-kode diterima oleh penerima maka diperlukan metode yang dapat mengubah kembali informasi ke bentuk asalnya yaitu dengan menggunakan dekripsi [9]. Kriptografi memiliki berbagai macam algoritma untuk melakukan proses enkripsi dan dekripsi yang berbeda-beda. Salah satu algoritma untuk proses enkripsi dan dekripsi adalah *Advanced Encryption Standard* (AES). AES merupakan enkripsi simetris yang diadopsi oleh pemerintah Amerika Serikat. AES terdiri dari tiga blok *chipper*, yaitu AES-128, AES-192, AES-256 yang masing-masing memiliki ukuran 128-bit.

Berdasarkan penelitian judul *Advanced Encryption Standard* (AES) *Algorithm to Encrypt*

and Decrypt Data. Penelitian tersebut membahas mengenai beberapa kriteria yang digunakan *National Institute of Standards and Technology* (NIST) di Amerika Serikat untuk melakukan proses evaluasi Algoritma AES yaitu diantaranya keamanan (*security*), biaya (*cost*) dan implementasi algoritma [10]. Mempertimbangkan dari hasil evaluasi pada algoritma AES untuk mengamankan data yang berisi informasi penting sangat dibutuhkan. Menggunakan algoritma AES pada pesan saat proses pengiriman melalui media internet akan menjadi lebih aman. Pesan yang terkirim akan berbentuk kode hasil enkripsi yang akan disimpan pada *database*, kemudian ditampilkan pada penerima pesan berbentuk informasi yang telah didekripsi sehingga kemungkinan pencurian informasi sangat kecil.

1.2 Batasan Masalah

Adapun batasan masalah dalam penelitian ini meliputi:

a. Perancangan aplikasi mengambil data dari berbagai referensi yang ada.
b. Perancangan aplikasi dilakukan dengan menggunakan bahasa pemrograman PHP, JavaScript, HTML dan CSS.
c. Menghasilkan aplikasi yang dapat digunakan untuk melakukan pengiriman pesan singkat terenkripsi dengan aman menggunakan algoritma enkripsi AES.

1.3 Tujuan penelitian

Tujuan dari diadakannya penelitian, perancangan dan pembuatan aplikasi pesan singkat terenkripsi dengan algoritma *Advanced Encryption Standard* (AES) berbasis *web* ini:

a. Untuk mendapatkan sebuah aplikasi pesan singkat yang dapat melakukan enkripsi dan dekripsi pada pesan dengan menggunakan algoritma AES.
b. Untuk mendapatkan sebuah aplikasi pesan singkat yang aman ketika mengirimkan informasi dari pengirim kepada penerima dengan menggunakan algoritma enkripsi AES.
c. Untuk mendapatkan sebuah aplikasi pesan singkat yang teruji tingkat keamanannya dengan menggunakan algoritma enkripsi AES.

2. TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Beberapa hasil penelitian yang pernah dilakukan oleh peneliti sebelumnya yang memiliki bidang dan tema yang sama dengan penelitian yang akan dilakukan.

[11] melakukan penelitian dengan judul Pengembangan Aplikasi Chat Messenger Dengan Metode Advanced Encryption Standard (AES) Pada Smartphone. Membahas tentang bagaimana cara melakukan implementasi algoritma kriptografi AES pada aplikasi chat messenger sehingga pengguna dapat berkomunikasi secara aman tanpa khawatir akan terjadi penyadapan terhadap pesan yang dikirim.

[12] melakukan penelitian dengan judul Perlindungan Web Pada Login Sistem Menggunakan Algoritma Rijndael. Membahas tentang cara mengamankan sistem dengan algoritma Rijndael untuk mengenkripsi data karena telah dinobatkan sebagai algoritma AES.

[13] melakukan penelitian dengan judul Aplikasi Keamanan E-mail Menggunakan Algoritma AES (Advanced Encryption Standard) Berbasis Android. Membahas cara mengamankan informasi yang terdapat pada email menggunakan ilmu kriptografi dengan algoritma AES karena tingkat keamanannya yang kuat dan mudah diimplementasikan.

2.2 Pesan

Pesan adalah materi pernyataan yang disampaikan komunikator kepada komunikan secara lisan maupun berupa tulisan yang dapat dipahami oleh kedua pihak [14]. Pesan merupakan lambang, suara, gambar dan lainnya yang disampaikan dari suatu sumber ke sasaran (*audience*) dengan menggunakan media tertentu [15].

2.3 Data

Data adalah bahan mentah bagi informasi, dirumuskan sebagai kelompok lambang-lambang tidak acak menunjukkan jumlah-jumlah, tindakan-tindakan, hal-hal dan sebagainya. Data dikumpulkan dengan metode pengamatan secara langsung, wawancara, perkiraan korespondensi dan daftar pertanyaan [16].

2.4 Database

Basis data adalah adalah sebuah sistem yang berfungsi untuk menyimpan dan mengolah

sekumpulan data. Setiap database mempunyai API (*Application Programming Interface*) tertentu untuk membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalamnya sehingga bisa dimanfaatkan oleh aplikasi lainnya. [17].

2.5 Kriptografi

Kriptografi berasal dari bahasa Yunani yaitu *crypto* (rahasia) dan *graphia* (tulisan) yang menurut terminologinya merupakan ilmu dan seni untuk menjaga keamanan pesan ketika dikirim dari suatu tempat ke tempat lain. Kriptografi terdiri dari tiga fungsi dasar yaitu enkripsi, dekripsi dan kunci. Berdasarkan penggunaan kunci pada algoritma kriptografi dibagi menjadi simetri dan asimetri. Algoritma simetri menggunakan satu kunci untuk proses enkripsi dan dekripsinya. Contoh algoritma simetri adalah *Data Encryption Standard* (DES), *Advanced Encryption Standard* (AES), RC2, RC4, RC5, RC6, *International Data Encryption Algoritm* (IDEA), *One Time Pad* (OTP) dan A5. Sedangkan algoritma asimetri menggunakan kunci yang berbeda untuk proses enkripsi dan dekripsinya. Kunci pada algoritma asimetri terbagi menjadi dua yaitu kunci umum (*public key*) yang semua orang boleh tau dan kunci rahasia (*private key*) yang dirahasiakan dari orang lain. Contoh algoritma asimetri adalah *Digital Signature Algoritm* (DSA), RSA, Diffie-Hellman (DH), *Elliptic Curve Cryptography* (ECC) dan Quantum [18].

2.6 AES (Advanced Encryption Standard)

Pada bulan Januari tahun 1997, NIST (*National Institute of Standards and Technology*) memprakarsai penelitian untuk mendapatkan pengganti dari DES (*Data Encryption Standard*). Persyaratan untuk standar baru, yang disebut *Advanced Encryption Standard* (AES) harus memenuhi beberapa diantarannya (1) blok cipher 128-bit dengan pilihan tiga ukuran kunci 128, 192 dan 256 bit, (2) memiliki desain secara umum (*public*) dan fleksibel, (3) mempunyai tingkat keamanan setidaknya sama dengan two-key triple-DES, (4) terbebas dari biaya royalti di seluruh dunia.

Pada akhir dari standarisasi yang telah bertahun-tahun dilakukan oleh kriptanalitik dan keahlian implementasi yang disediakan dari seluruh dunia, algoritma Rijndael yang dikembangkan oleh Joan Daemen dan Vincent Rijmen adalah pilihan populer untuk menjadi AES. Pada bulan November tahun 2001, AES sampai pada kesimpulannya dengan dipublikasikan pada *Federal Information Processing Standard* (FIPS) dan sampai saat ini dengan cepat menjadi komponen vital dari infrastruktur digital [19].

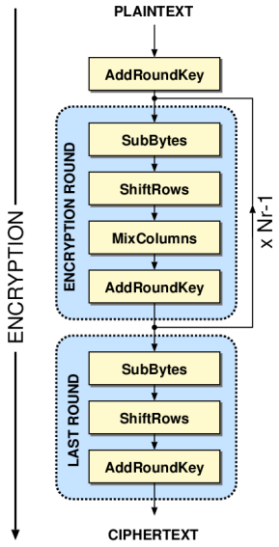
2.7 Algoritma AES

AES secara garis besar beroperasi pada blok 128-bit dengan kunci 128-bit sebagai berikut, (1) AddRoundKey yaitu melakukan operasi XOR antara state awal (*plaintext*) dengan *cipher key*. Tahap ini juga disebut sebagai *initial round*, (2) putaran sebanyak Nr – 1 kali. Proses yang dilakukan setiap putaran adalah SubBytes, ShiftRows, MixColumns dan AddRoundKey, (3) *final round* atau proses untuk putaran terakhir yaitu SubBytes, ShiftRows dan AddRoundKey.

Berdasarkan ukuran blok yang tetap, AES bekerja pada matriks berukuran 4 x 4 dimana setiap selnya terdiri atas 1 *byte* (8 bit). Setiap *plaintext* akan diubah terlebih dahulu ke dalam blok-blok tersebut dengan bentuk heksadesimal. Setelah proses pengubahan selesai, blok akan diproses dengan metode yang dijelaskan.

2.8 Enkripsi AES

Proses enkripsi pada algoritma AES terdiri dari 4 jenis transformasi byte yaitu SubBytes, ShiftRows, MixColumns dan AddRoundKey. Pada awal proses enkripsi, input yang telah disalin ke dalam *state* akan mengalami transformasi *byte* AddRoundKey. Setelah itu, *state* akan mengalami transformasi SubBytes, ShiftRows, MixColumns dan AddRoundKey secara berulang-ulang sebanyak Nr. Jumlah perulangan Nr tergantung pada kunci yang digunakan pada saat awal proses enkripsi. Untuk kunci 128-bit akan melakukan 10 putaran, 192-bit melakukan 12 putaran dan 256-bit melakukan 14 putaran. Proses perulangan sebanyak Nr dalam algoritma AES disebut sebagai *round function*. *Round* yang terakhir (*last round*) berbeda dengan sebelumnya dimana *state* tidak mengalami transformasi MixColumns). Keseluruhan proses enkripsi dengan algoritma AES dapat dilihat pada gambar 2.1.



Gambar 2.1. Proses Enkripsi AES

2.9 AddRoundKey

Pada proses ini *subkey* digabungkan dengan *state*. Proses penggabungan menggunakan operasi XOR untuk setiap *byte* dari *subkey* dengan *byte* yang bersangkutan dari *state*. Untuk setiap tahap, *subkey* dimunculkan dari kunci utama dengan menggunakan *processkey schedule*. Setiap *subkey* berukuran sama dengan *state* yang bersangkutan.

Misalkan dengan *plaintext* "Tugas Akhir 2019" dan dengan *key* "Two One Nine Two" maka dapat dibuat sebuah *state*. Bentuk awal dari *plaintext* dan *key* adalah dalam karakter ASCII (*American Standard Code for Information Interchange*). Sebelum menjadi sebuah *state*, terlebih dahulu *plaintext* diubah menjadi bentuk bilangan heksadesimal. Bilangan heksadesimal merupakan bilangan berbasis 16 yang terdiri dari angka 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 dan huruf A, B, C, D, E, F. Hasil dari pengubahan *plaintext* dan *key* menjadi heksadesimal yaitu 54 75 67 61 73 20 41 6B 68 69 72 20 32 30 31 39 serta 54 77 6F 20 4F 6E 65 20 4E 69 6E 65 20 54 77 6F dapat dijadikan *state* yang dapat dilihat pada tabel 2.1 dan 2.2.

Tabel 2.1. State Plaintext

54	73	68	32
75	20	69	30
67	41	72	31
61	6B	20	39

Tabel 2.2. State Key

54	4F	4E	20
77	63	69	54
6F	65	6E	77
20	20	65	6F

2.10 KeyExpansion

Proses inisialisasi perubahan kunci yang dilakukan dengan mengambil input dari *state* kunci yang dibagi menjadi 4 *byte*. Pada *byte* akan dilakukan pergeseran ke kiri sebanyak satu kali, pergantian *byte* dengan S-Box seperti pada proses SubBytes dan melakukan operasi XOR berdasarkan pada tabel Round Constant seperti pada tabel 2.3.

Tabel 2.3. Round Constant

Putaran (128-bit)	Heksadesimal
1	01
2	02
3	04
4	08
5	10
6	20
7	40
8	80
9	1B
10	36

2.11 SubBytes

Pada proses SubBytes, dilakukan operasi substitusi tidak linear dengan cara mengganti setiap *byte* pada *state* AddRoundKey dengan *byte* pada sebuah tabel yang bernama S-Box. Berikut merupakan bentuk dari tabel S-Box yang terdiri dari 16 baris dan 16 kolom yang masing-masing berukuran 1 *byte*. Tabel S-Box dapat dilihat pada gambar 2.2.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Gambar 2.2. S-Box

Byte yang berada di dalam *state* AddRoundKey akan dicocokkan dengan kolom dan baris pada tabel S-Box. Setiap *byte* terdiri dari dua kombinasi antara angka-angka, angka-huruf, huruf-angka dan huruf-huruf. Untuk karakter pada kombinasi sebelah kiri dicocokkan dengan kolom tabel S-Box yang terletak disebelah kiri. Sedangkan kombinasi sebelah kanan dicocokkan dengan baris tabel S-Box dibagian atas. Berdasarkan hasil dari pergantian *byte* maka akan didapatkan *state* seperti pada tabel 2.4.

Tabel 2.4. State SubBytes

63	EB	F7	C9
77	1A	63	43
30	36	9C	5A
83	B3	6E	B1

2.12 ShiftRows

ShiftRows adalah proses melakukan *shift* atau pergeseran pada setiap elemen tabel yang dilakukan per barisnya. ShiftRows akan beroperasi pada setiap baris dari tabel *state*. Proses ini bekerja dengan cara memutar *byte* pada tiga baris terakhir (baris 1, 2 dan 3) dengan jumlah putaran yang berbeda-beda. Baris 1 diputar sebanyak 1 kali, baris 2 diputar sebanyak 2 kali dan baris 3 diputar sebanyak 3 kali. Sedangkan baris 0 tidak akan diputar. Hasil dari proses ShiftRows dapat dilihat pada tabel 2.5.

Tabel 2.5. State ShiftRows

63	EB	F7	C9
1A	63	43	77
9C	5A	30	36
B1	83	B3	6E

2.13 MixColumns

Proses yang terjadi pada MixColumns adalah mengalikan setiap elemen dari *state* ShiftRows dengan matriks Galois *field*. Operasi ini menggabungkan 4 *byte* dari setiap kolom tabel *state* ShiftRows dan menggunakan transformasi *linear*. Operasi MixColumns memperlakukan setiap kolom polinominal 4 suku dalam Galois *field.* MixColumns juga bisa disebut dengan perkalian matriks yaitu mengalikan 4 bilangan pada Galois *field* seperti pada tabel 2.6.

Tabel 2.6. Galois Field

02	03	01	01
01	02	03	01
01	01	02	03
03	01	01	02

Perhitungan MixColumns pada baris pertama dari Galois field dengan kolom pertama state ShiftRows sebagai berikut.

$= (02 \cdot 63) \oplus (03 \cdot 1A) \oplus (01 \cdot 9C) \oplus (01 \cdot B1)$

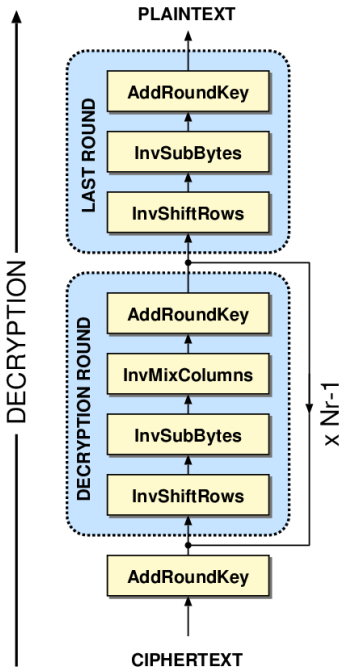
$= (10 \cdot 1100011) \oplus ((02 \cdot 1A) \oplus 1A) \oplus (1 \cdot 10011100) \oplus (1 \cdot 10110001)$

$= (11000110) \oplus (110100 \oplus 11010) \oplus (101101)$

$(11000110) \oplus (101110) \oplus (101101) = C5$

2.14 Dekripsi AES

Transformasi *cipher* dapat dibalikkan dan diimplementasikan dalam arah yang berlawanan untuk menghasilkan *inverse cipher* yang mudah dipahami pada algoritma AES. Transformasi *byte* yang digunakan untuk *inverse cipher* adalah InvShiftRows, InvSubBytes, InvMixColumns dan AddRoundKey. Proses dekripsi diawali dengan transformasi *bytes* AddRoundKey yang berasal dari *ciphertext*. Setelah itu, *state* akan mengalami transformasi InvShiftRows, InvSubBytes, InvMixColumns dan AddRoundKey secara berulang-ulang sebanyak Nr. Jumlah perulangan Nr tergantung pada kunci yang digunakan pada saat proses enkripsi. Panjang kunci pada saat proses enkripsi dan dekripsi harus sama. Proses perulangan sebanyak Nr dalam algoritma AES disebut sebagai *round function*. *Round* yang terakhir (*last round*) berbeda dengan sebelumnya dimana *state* tidak mengalami transformasi InvMixColumns. Berikut merupakan proses dari keseluruhan dekripsi dengan algoritma AES seperti yang terlihat pada gambar 2.3.



Gambar 2.3. Proses Dekripsi AES

3. METODE PENELITIAN

3.1 Metode Penelitian

Metode penelitian yang digunakan adalah model *waterfall* atau yang sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan. Pengembangan pada metode waterfall memiliki beberapa tahapan yang berurut yaitu analisis kebutuhan *requirement* (analisis kebutuhan), *design system* (desain sistem), *coding* (pengkodean) dan *testing* (pengujian) [10].

3.1 Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam melakukan penerapan algoritma enkripsi *Advanced Encryption Standard* (AES) pada perancangan aplikasi pesan singkat terenkripsi berbasis *web* yaitu:

a. Sistem operasi menggunakan Windows 10 Pro 64-bit.
b. Pengolah database menggunakan MySQL.
c. Perancangan sistem menggunakan Microsoft Visio 2007.
d. Pengolah bahasa pemrograman menggunakan Sublime Text Editor.
e. Penulisan laporan menggunakan Office 365.
f. Pengujian hasil enkripsi menggunakan Cryptool.

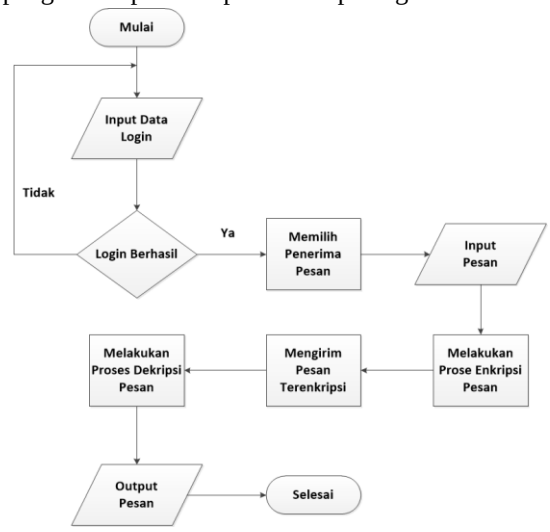
3.2 Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam melakukan penerapan algoritma enkripsi *Advanced Encryption Standard* (AES) pada perancangan aplikasi pesan singkat terenkripsi berbasis *web* yaitu:

a. Laptop Acer Aspire E1-471.
b. Processor Intel(R) Core(TM) i3-2348M CPU @ 2.30 GHz.
c. RAM 6,00 GB (5,84 usable).
d. Hardisk 500 GB.

3.3 Perancangan Sistem

Pada tahapan ini, spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari untuk menentukan desain sistem. Spesifikasi yang dipelajari adalah mengenai kebutuhan calon pengguna dan proses implementasi program. Hal ini dilakukan untuk memenuhi kebutuhan pengguna sehingga akan mempermudah dalam menentukan langkah-langkah yang harus dilakukan untuk menyelesaikan desain sistem. Berikut adalah *flowchart* untuk proses pengiriman pesan dapat dilihat pada gambar 3.1.



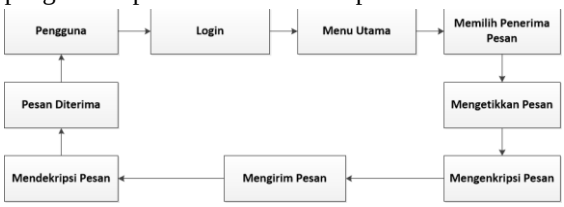
Gambar 3.1. Flowchart Proses Pertukaran Pesan

4. ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem

Sistem pada proses perancangan aplikasi pesan singkat terenkripsi dengan algoritma AES akan

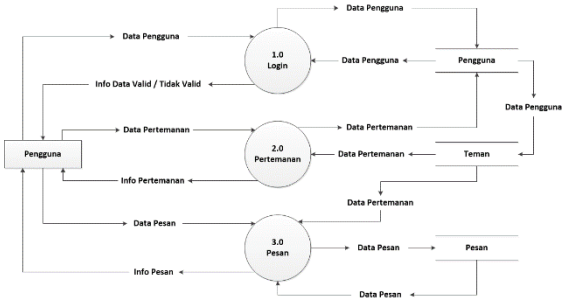
dirancang untuk meningkatkan keamanan pada proses pengiriman pesan. Alur sistem dapat dilihat pada gambar 4.1 yang menggambarkan proses pengiriman pesan secara terenkripsi.



Gambar 4.1. Alur Pengiriman Pesan

4.2 Rancangan Sistem

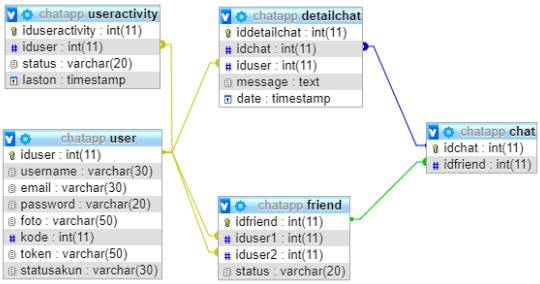
Pada proses perancangan sistem, spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari untuk menentukan desain sistem. Spesifikasi yang dipelajari adalah mengenai kebutuhan calon pengguna dan proses implementasi program untuk memenuhi kebutuhan pengguna sehingga akan mempermudah dalam menentukan langkah-langkah yang harus dilakukan untuk menyelesaikan desain sistem. Langkah yang harus dilakukan adalah pendefinisian kebutuhan informasi, kriteria kinerja sistem, identifikasi jenis input dan output yang diinginkan pengguna. Kemudian proses perancangan desain sistem akan menggunakan Data Flow Diagram (DFD) seperti pada gambar 4.2.



Gambar 4.2. DFD

4.3 Rancangan Database

Perancangan database untuk aplikasi pesan singkat terenkripsi dengan algoritma AES seperti terlihat pada gambar 4.3. Entitas useractivity memiliki relasi dengan entitas user melalui atribut iduser. Kemudian entitas user memiliki relasi dengan entitas detailchat dan entitas friend melalui atribut iduser. Selanjutnya, entitas detailchat memiliki relasi dengan entitas chat melalui atribut idchat. Relasi entitas terakhir yaitu antara entitas friend dengan entitas chat melalui atribut idfriend.

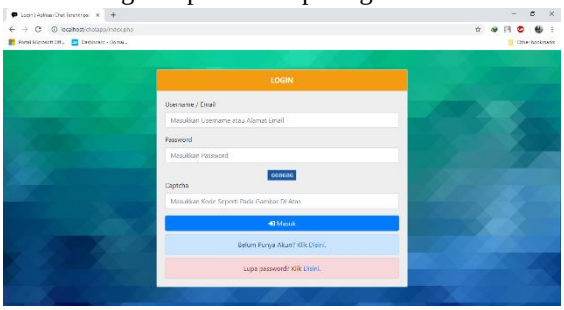


Gambar 4.3. Relasi Antar Tabel

5. IMPLEMENTASI DAN PENGUJIAN

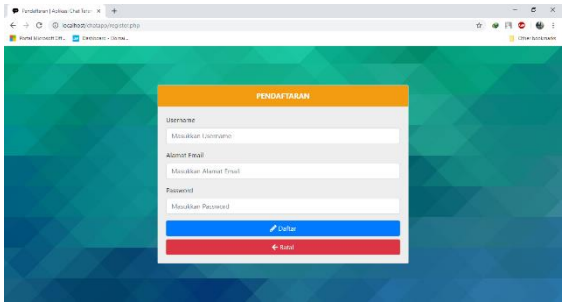
5.1 Tampilan Aplikasi

Aplikasi yang dibangun diimplementasikan berdasarkan rancangan yang telah dibuat dalam bentuk flowchart dan DFD. Berikut merupakan tampilan aplikasi dari hasil implementasi rancangan-rancangan tersebut beserta penjelasannya. Tampilan halaman login merupakan merupakan menu yang digunakan pengguna untuk melakukan proses masuk ke dalam sistem dengan masukkan data *username/email*, *password* dan *captcha*. Tampilan halaman login dapat dilihat pada gambar 5.1.



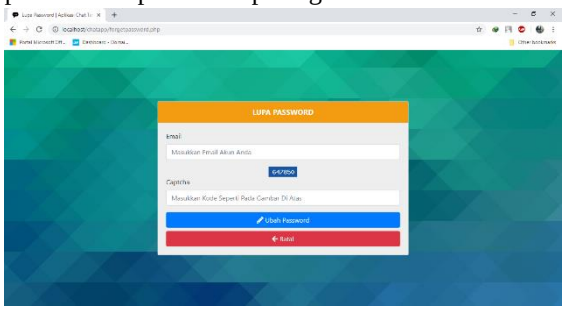
Gambar 5.1. Halaman Login

Halaman pendaftaran akun baru digunakan calon pengguna untuk melakukan proses pendaftaran ke sistem dengan memasukkan data ke dalam *form* input yang berupa *username*, *email* dan *password*. Kemudian calon pengguna akan menerima pesan setelah menekan tombol daftar untuk mengkonfirmasi pendaftaran akun melalui email yang didaftarkan sebelumnya. Kemudian calon pengguna akan menerima pesan setelah menekan tombol daftar untuk mengkonfirmasi pendaftaran akun melalui *email* yang didaftarkan sebelumnya. Halaman pendaftaran akun baru dapat dilihat pada gambar 5.2.



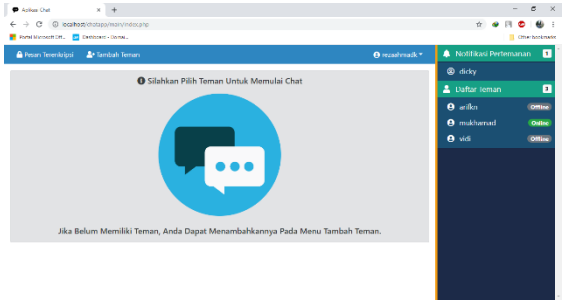
Gambar 5.2. Halaman Pendaftaran Akun Baru

Halaman lupa *password* merupakan halaman yang digunakan pengguna untuk melakukan pengaturan ulang *password* pada akun yang terdaftar pada sistem. Pengguna diminta memasukkan alamat *email* yang digunakan pada akun dan *captcha* sebagai keamanan tambahan. Halaman lupa password dapat dilihat pada gambar 5.3.



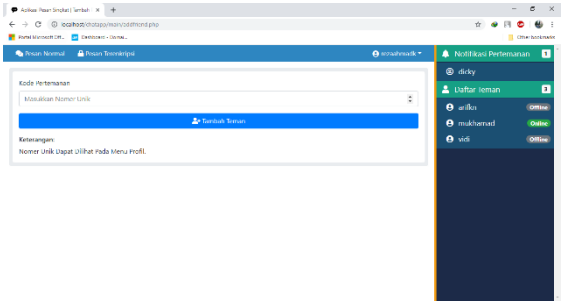
Gambar 5.3. Halaman Lupa Password

Halaman menu utama akan tampil pada saat pertama kali diakses oleh pengguna yang berhasil melakukan login dan akan menampilkan beberapa menu diantaranya pesan terenkripsi, tambah teman, profil, notifikasi pertemanan dan daftar teman. Halaman menu utama dapat dilihat pada gambar 5.4.



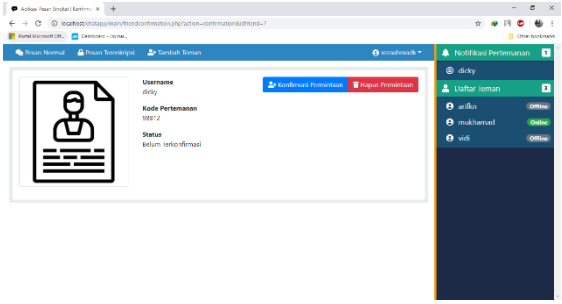
Gambar 5.4. Halaman Menu Utama

Halaman tambah pertemanan digunakan pengguna untuk menambah pertemanan yang dilakukan dengan memasukkan kode unik milik calon teman yang terdapat pada menu profil ke dalam form input kode pertemanan. Setelah kode dimasukkan, pengguna dapat menambahkan permintaan pertemanan dengan menekan tombol tambah teman. Halaman tambah pertemanan dapat dilihat pada gambar 5.5.



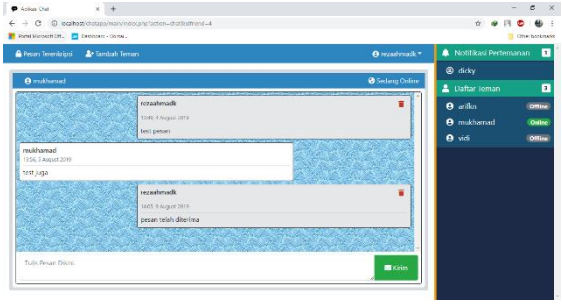
Gambar 5.5. Halaman Tambah Pertemanan

Halaman konfirmasi pertemanan akan ditampilkan pada saat membuka notifikasi pertemanan. Konfirmasi permintaan pertemanan dapat dilakukan dengan menekan tombol konfirmasi. Sedangkan jika ingin menolak permintaan pertemanan pengguna dapat menekan tombol hapus permintaan. Halaman konfirmasi pertemanan dapat dilihat pada gambar 5.6.



Gambar 5.6. Halaman Konfirmasi Pertemanan

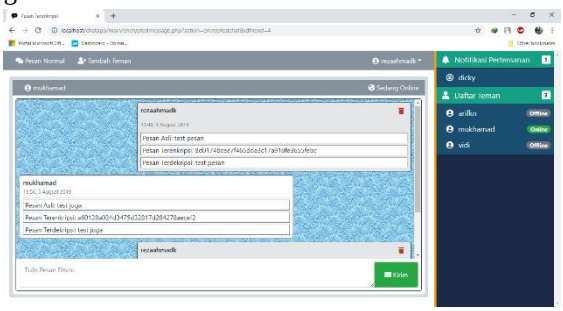
Halaman pertukaran pesan normal akan tampil pada saat membuka percakapan pada saat memilih penerima pesan dari daftar teman. Pada menu ini pengirim dan penerima dapat saling bertukar pesan. Halaman pertukaran pesan noraml dilihat pada gambar 5.7.



Gambar 5.7. Halaman Pertukaran Pesan Normal

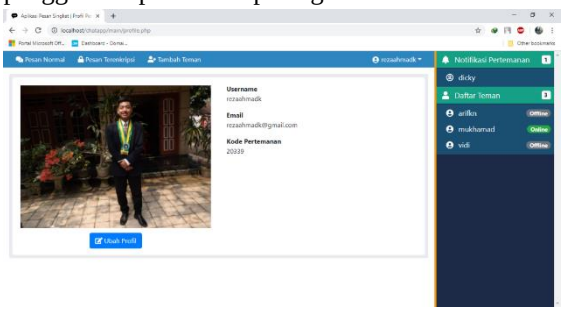
Halaman tambah pertukaran pesan terenkripsi akan menampilkan pesan asli, terenkripsi dan terdekripsi. Pengirim dan penerima dapat mengetikkan pesan pada *form* input dan menekan

tombol kirim untuk mengirimkan pesan. Halaman pertukaran pesan terenkripsi dapat dilihat pada gambar 5.8.



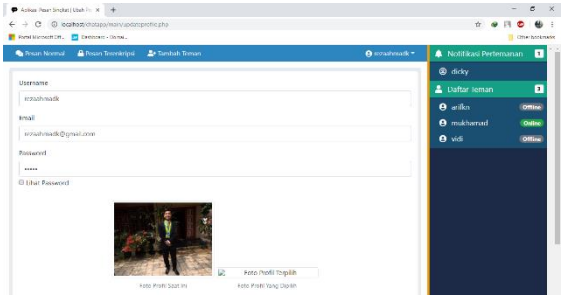
Gambar 5.8. Halaman Pertukaran Pesan Terenkripsi

Halaman profil pengguna akan menampilkan informasi akun yang digunakan pada saat berhasil melakukan proses login. Informasi yang dapat dilihat dan dibaca oleh pengguna yaitu *username*, *email* dan kode pertemanan. Halaman profil pengguna dapat dilihat pada gambar 5.9.



Gambar 5.9. Halaman Profil Pengguna

Halaman ubah profil penggguna akan digunakan oleh pengguna untuk mengubah informasi akun yang digunakan. Informasi akun sebelumnya akan ditampilkan pada *form* input sehingga dapat mempermudah pengguna dalam melakukan pengubahan. Halaman ubah profil pengguna dapat dilihat pada gambar 5.10.



Gambar 5.10. Halaman Ubah Profil Pengguna

5.2 Pengujian Program

Pengujian program pada proses enkripsi dan dekripsi berguna untuk mengetahui apakah data berupa pesan teks yang dikirim dan diterima bekerja dengan maksimal. Beberapa hal yang diuji pada program akan dijelaskan seperti berikut.

5.2.1 Pengujian Waktu Proses Enkripsi dan Dekripsi Pesan

Program akan diuji dengan menghitung berapa lama waktu yang diperlukan untuk melakukan proses enkripsi dan dekripsi pesan. Selain itu, perhitungan waktu ini juga mencakup keseluruhan elemen pada program ketika dibuka melalui *browser*. Pengujian dilakukan menggunakan fitur *browser* Google Chrome yaitu *Developer Tools* saat menghitung waktu halaman *web* menampilkan pesan terdekripsi. Proses penampilan pesan dihitung sejak pengguna mengirimkan pesan terenkripsi sampai pesan hasil dekripsi ditampilkan. Berikut merupakan hasil dari pengujian yang dapat dilihat pada tabel 5.1.

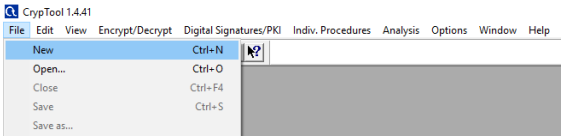
Tabel 5.1. Pengujian Waktu Proses Enkripsi dan Dekripsi Pesan

No.	Jumlah Pesan	Waktu (detik)
1.	100 pesan	500ms sampai 800ms
2.	200 pesan	1 detik sampai 2 detik
3.	500 pesan	3 detik sampai 4 detik
4.	1000 pesan	5 detik sampai 7 detik
5.	2000 pesan	10 detik sampai 14 detik

5.2.2 Pengujian Kunci Enkripsi dan Dekripsi

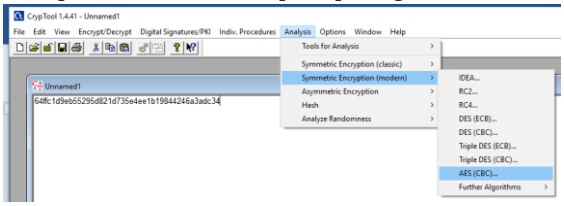
Pengujian berdasarkan kunci enkripsi dan dekripsi dilakukan untuk mengetahui berapa besar waktu yang dibutuhkan untuk mengetahui isi pesan sebenarnya jika terjadi kebocoran informasi. Kunci enkripsi dan dekripsi adalah sama karena algoritma AES termasuk ke dalam kriptografi simetris. Berikut merupakan hasil enkripsi dari aplikasi yang dirancang. Pesan hasil proses enkripsi yaitu 64ffc1d9eb55295d821d735e4ee1b19844246a3adc34. Pengujian dilakukan dengan aplikasi CrypTool untuk mendekripsikan pesan hasil enkripsi secara paksa (*brute force*). Hasil pengujian pesan terenkripsi menggunakan CrypTool akan dijelaskan sebagai berikut.

Pertama yaitu membuka aplikasi CrypTool file lalu buat file baru dengan membuka menu File kemudian pilih New seperti pada gambar 5.11. Setelah berhasil maka akan menampilkan jendela baru yang digunakan untuk memasukkan pesan terenkripsi.



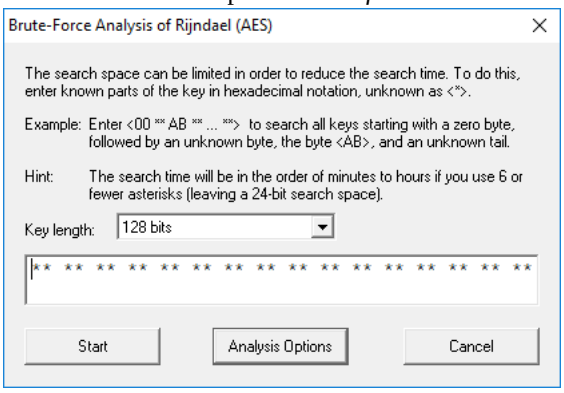
Gambar 5.11. Pembuatan File Baru Pada CrypTool

Kedua yaitu mengetikkan pesan terenkripsi pada jendela baru. Buka menu Analysis, lalu arahkan *cursor* pada bagian *Symmetric Encryption* (*modern*) maka akan muncul beberapa pilihan algoritma enkripsi termasuk AES seperti pada gambar 5.12.



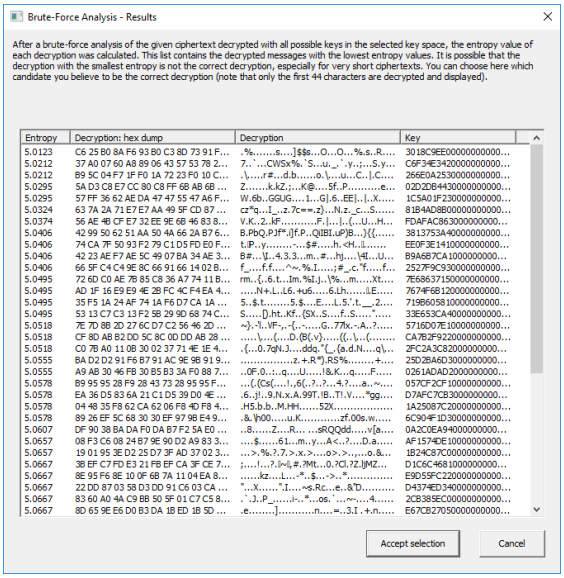
Gambar 5.12. Pemilihan Menu Analysis Pada CrypTool

Langkah ketiga yaitu pilih pada AES (CBC) maka akan muncul jendela baru untuk melakukan *brute force* seperti pada gambar 5.13. CBC (*Cipher Blocker Chaining*) merupakan bentuk lanjutan dari enkripsi blok *cipher*. Lalu atur panjang kunci pada 128-bits dan mengosongkan *form* input lalu pilih Start untuk memulai proses *brute force*.



Gambar 5.13. Pengaturan Kunci Untuk Proses Brute Force Pada CrypTool

Langkah terakhir yaitu menunggu hasil dari proses *brute force* selesai 14. Berikut adalah hasil dari *brute force* setelah menunggu selama 14 jam dan pesan belum terpecahkan dengan menekan tombol *cancel*. Sebuah jendela baru dengan informasi hasil dari *brute force* seperti *entropy* yaitu ukuran pesan, *decryption* dalam heksadesimal, *decryption* dalam pesan teks dan *key* berupa kunci enkripsi seperti pada gambar 5.14.



Gambar 5.14. Tampilan Hasil Brute Force Pada Cryptool

6. PENUTUP

6.1 Kesimpulan

Perancangan Aplikasi Pesan Singkat Terenkripsi Dengan Algoritma Advanced Encryption Standard (AES) Berbasis Web dibangun dengan dasar untuk memberi keamanan pada proses pengiriman pesan singkat. Berdasarkan pada hasil analisis dan penelitian yang dilakukan maka didapat kesimpulan sebagai berikut.

a. Telah dibangun Aplikasi Pesan Singkat Terenkripsi Dengan Algoritma *Advanced Encryption Standard* (AES) Berbasis Web dengan menggunakan bahasa pemrograman HTML, CSS, Javascript, PHP dan SQL.
b. Aplikasi yang dibangun dapat melakukan proses enkripsi dan dekripsi pesan singkat dengan algoritma AES 128-bit.
c. Aplikasi yang dibangun dapat mengamankan pesan yang terenkripsi dengan algoritma AES 128-bit.

6.2 Saran

Berdasarkan kesimpulan diatas, maka terdapat beberapa saran yang dapat membangun Aplikasi Pesan Singkat Terenkripsi Dengan Algoritma Advanced Encryption Standard (AES) Berbasis Web menjadi lebih baik dari tulisan ini, yaitu sebagai berikut.

a. Aplikasi ini masih belum ada fitur untuk mengecek jika terdapat pesan baru dan notifikasi pertemanan secara realtime.

Kedepannya bisa dibuat untuk pemuatan pesan baru dan notifikasi pertemanan secara *realtime*.

b. Aplikasi ini masih belum dapat melakukan proses enkripsi dan dekripsi pesan berupa gambar, suara dan video. Kedepannya bisa dikembangkan untuk mengirimkan pesan terenkripsi dari berbagai sumber.
c. Aplikasi ini masih menggunakan pemrograman native dalam menerapkan algoritma enkripsi. Kedepannya dapat dikembangkan untuk pemrograman dengan *framework*.

UCAPAN PERSEMBAHAN

Naskah Publikasi ini dapat diselesaikan tidak lepas dari segala bantuan, bimbingan, dorongan dan doa dari berbagai pihak, yang pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Kepada Bapak Dr. Bambang Moertono Setiawan, MM., Akt., CA. selaku Rektor di Universitas Teknologi Yogyakarta.
2. Kepada Bapak Sutarman, Ph.D. selaku Dekan Fakultas Teknologi Informasi dan Elektro.
3. Kepada Ibu Dr. Enny Itje Sela, S.Si., M.Kom. selaku Ketua Program Studi Teknik Informatika.
4. Kepada Bapak Donny Avianto, S.T., M.T. selaku Dosen Pembimbing Proyek Tugas Akhir yang telah banyak memberikan petunjuk dalam penyusunan naskah publikasi ini.
5. Kepada kedua orang tua penulis, yang telah mendukung, dan selalu mendoa'kan sehingga bisa menyelesaikan program dan laporan tugas akhir dengan cepat.

DAFTAR PUSTAKA

[1]Suprapto, T. (2009), Pengantar Teori Dan Manajemen Komunikasi, Yogyakarta: MedPress.

[2]Maryono, N. dan Istiana, B.P. (2008), Teknologi Informasi dan Komunikasi 3, Bogor: Quadra.

[3]Whitaker, A. dan P. Newman, D. (2006), Penetration Testing and Network Defense, Indianapolis: Cisco Press.

[4]M. Stewart, J., Tittel, E. dan Chapple, M. (2008), CISSP: Certified Information Systems Security Professional Study Guide, Indianapolis: Wiley Publishing, Inc.

[5]Sharma, D. (2009), Foundation of IT, Punjab: Excel Books India.

[6]Migga K, J. (2009), Guide to Computer Network Security, Chattanooga: Springer Science & Business Media.

[7]Abdulhaque B, A. (2018), Building Serverless Python Web Services with Zappa, Birmingham: Packt Publishing.

[8][9]Mukhtar, H. (2018), Kriptografi Untuk Keamanan Data, Yogyakarta: Deepublish.

[10]Abdullah, A.M. (2017), Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data, Cryptography and Network Security, Departement of Applied Mathematics and Computer Science, Eastern Mediterranean University, Cyprus, (June).

[11]Suryanto, I., Suhery, C. dan Brianorman, Y. (2017), Pengembangan Aplikasi Chat Messenger Dengan Metode Advanced Encryption Standard (AES) Pada Smartphone, Jurnal Coding Sistem Komputer Untan, 05(2), 1–12.

[12]Adhitya Dharmawan, E., Yudaningtyas, E. dan Sarosa, M. (2013), Perlindungan Web pada Login Sistem Menggunakan Algoritma Rijndael, Jurnal EECCIS, 7(1), 77–84.

[13]Ayun Qolbu M, N., Sutardi dan Tajidun, L. (2016), Aplikasi Keamanan E-mail Menggunakan Algoritma AES (Advanced Encryption Standard) Berbasis Android, SemanTIK, 2(1), 321–330.

[14]Mutialela C.R., (2017), Konsep dan Aplikasi Ilmu Komunikasi, Yogyakarta: Andi.

[15]Suprapto, T. (2009), Pengantar Teori Dan Manajemen Komunikasi, Yogyakarta: MedPress.

[16] Hutahaean, J. (2014), Konsep Sistem Informasi, Yogyakarta: Deepublish.

[17]Enterprise, J. (2016), Belajar Java, Database, dan NetBeans dari Nol, Jakarta: Elex Media Komputindo.

[18]Ariyus, D. (2008), Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi, Yogyakarta: Andi.

[19] Dobberin, H., Rijmen, V. dan Sowa, A. (2015), Advanced Encryption Standard - AES, Berlin: Springer Science & Business Media.