Vol.2 No.2 Juli-Desember 2016
JESIK
JURNAL ELEKRONIK SISTEM INFORMASI DAN KOMPUTER
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER (STMIK) BINA MULIA

PENERAPAN TEKNIK KRIPTOGRAFI PADA *DATABASE* MENGGUNAKAN ALGORITMA *ONE TIME PAD*

Hasrul Hasrul[1)], Lamro Herianto Siregar

STMIK Bina Mulia Palu
Website: stmik-binamulia.ac.id

ABSTRAK

Salah satu fungsi penting *database* adalah menjaga kerahasiaan data dan infomasi yang ada didalamnya. Masalah keamanan *database* ini telah menjadi salah satu topik pembicaraan penting karena banyaknya jumlah kejahatan *ciber* yang memanfaatkan celah keamanan *database* untuk memanipulasi data dan informasi yang ada. Pada dasarnya terdapat dua teknik untuk mengamankan *database*, yaitu steganografi dan kriptografi. Saat ini teknik kriptografi yang banyak digunakan karena teknik ini mengaburkan/memanipulasi data dan informasi yang dianggap penting sehingga sulit untuk dipahami oleh pihak-pihak yang tidak berhak. Karena itu penelitian ini akan membangun suatu aplikasi yang dapat mengenkripsi data dan informasi yang ada didalam *database* sehingga menghalangi pihak yang tidak bertanggungjawab untuk mengubah, mengambil, atau menyalahgunakan data dan informasi tersebut. Penelitian ini merupakan penelitian kualitatif dengan pendekatan eksperimen. Dengan metode algoritma *One Time Pad* (OTP), penelitian ini merancang sistem enkripsi-dekripsi OTP untuk mengamankan data dan informasi yang ada didalam *database* STMIK Bina Mulia Palu. Hasil penelitian ini menunjukkan bahwa teknik kriptografi menggunakan algoritma OTP dapat diimplementasikan dengan bahasa pemograman *Microsoft Visual Basic* 6.0 sehingga dapat mengamankan data-data yang ada didalam *database*. Untuk itu, penelitian kedepan perlu dikembangkan agar lebih baik lagi dan dapat dimanfaatkan serta diterapkan pada bidang-bidang kehidupan yang lebih kompleks lagi.

Kata Kunci: Kriptografi*, One Time Pad, Visual Basic, Database*.

1. Latar Belakang

Database mempunyai hubungan yang sangat erat terhadap perangkat teknologi komputer. Hal ini karena *database* merupakan aspek utama yang ada didalam sistem operasi komputer yang kini telah menjadi suatu alat bantu utama dalam kehidupan manusia sehari-hari.

Salah satu fungsi penting *database* adalah menjaga kerahasiaan data dan infomasi yang ada didalamnya, dan karena itu masalah keamanan *database* menjadi salah satu topik pembicaraan penting di bidang teknologi informasi. Hal ini nampak dari banyaknya jumlah kejahatan *ciber* yang dibicarakan dalam berbagai media masa. Para pelaku kejahatan *ciber* ini memanfaatkan celah keamanan dalam *database* untuk dimasuki dan memanipulasi data-data yang ada didalamnya.

Hal ini sangat penting karena informasi hasil pengolahan data seringkali sangat berharga bagi penggunanya sehingga tidak boleh diketahui oleh pihak yang tidak berkepentingan. Kerahasiaan data dan informasi merupakan hal yang sangat penting bagi beberapa kalangan, seperti militer, perbankan, pemerintahan, perusahaan, lembaga pendidikan, organisasi, dan lain sebagainya.

Bahkan karena pentingnya kerahasiaan data dan informasi yang berbasis teknologi informasi, Pemerintah telah menyusun berbagai peraturan perundang-undangan khusus di bidang teknologi informasi (IT). Diantaranya adalah peraturan *Helath Insurance Portability and Accountability Act* (HIPAA) yang menstandarkan keamanan data-data medis dan data-data individual lainnya.

Penyalahgunaan hak atas sebuah informasi merupakan hal yang sangat dihindari oleh orang-orang yang membangun *database* sehingga sangat dibutuhkan ketelitian dalam mengamankan *database* dari pihak yang tidak bertanggungjawab.

Pada dasarnya terdapat dua teknik untuk mengamankan *database*, yaitu steganografi dan kriptografi. Saat ini teknik kriptografi yang paling banyak digunakan karena teknik ini mengaburkan/ memanipulasi data dan informasi yang penting sehingga akan sulit untuk dipahami oleh pihak-pihak yang tidak berhak atas informasi tersebut.

Dari uraian diatas, penelitian ini akan membangun aplikasi untuk mengenkripsi data dan informasi yang ada didalam *database* sehingga menghalangi pihak yang tidak bertanggungjawab mengubah, mengambil, atau menyalahgunakan data dan informasi tersebut.

[1)] *Dosen STMIK Bina Mulia Palu*

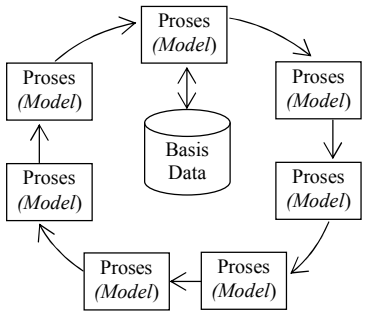
2. Tinjauan Pustaka

2.1 Data dan Informasi

Data adalah fakta mengenai objek, orang, dan lain-lain. Data dinyatakan dengan nilai angka, deretan karakter, atau simbol [1]. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata. Kejadian (*event*) tersebut adalah sesuatu yang terjadi pada saat tertentu [2]. Data adalah deskripsi dari sesuatu dan kejadian yang kita hadapi (*the discription of things and events that we face*) [3].

Adapun informasi adalah suatu data yang telah diolah sehingga lebih berarti dan berguna bagi penerimanya. Jadi, sumber suatu informasi adalah data [2]. Informasi merupakan suatu hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya, yang menggambarkan suatu kejadian (*event*) yang nyata (*fact*), yang dapat digunakan untuk pengambilan keputusan [4]. Informasi adalah sistem didalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi, serta menyediakan informasi kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [5].

Jadi, hubungan data dan informasi yaitu data adalah bentuk yang belum dapat bercerita banyak sehingga perlu diolah dengan suatu model tertentu agar dihasilkan informasi. Informasi ini diterima oleh pengguna yang membuat keputusan tertentu. Keputusan ini menimbulkan suatu tindakan yang menghasilkan sejumlah data baru yang ditangkap kembali sebagai *input*, diproses kembali melalui model tertentu, dan seterusnya sehingga membentuk suatu siklus yang disebut siklus informasi yang digambarkan sebagai berikut [5]:



Gambar 1 Siklus Informasi

2.2 *Database*

2.2.1 Pengertian *Database*

Banyak pakar yang telah mendefinisikan *database*, antara lain sebagai berikut [6]:

a. Himpunan kelompok data (arsip) yang saling berhubungan, yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
b. Kumpulan data yang saling berhubungan, yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
c. Kumpulan *file*/tabel/arsip yang berhubungan, yang disimpan di media penyimpanan elektronis.

Database juga didefinisikan sebagai suatu susunan/kumpulan data operasional yang lengkap dari suatu organisasi/perusahaan yang diorganisir/ dikelola dan disimpan secara terintegrasi dengan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan oleh pemakainya [7].

Database diartikan sebagai kumpulan item data yang berhubungan satu dengan yang lainnya, yang diorganisasikan berdasarkan sebuah skema/ struktur tertentu, tersimpan di *hardware* komputer dan dengan menggunakan suatu *software* tertentu untuk melakukan manipulasi untuk memperoleh kegunaan tertentu [8].

Database dapat dianggap sebagai tempat/ lokasi untuk sekumpulan berkas data yang sudah terkomputerisasi untuk memelihara dan memuat informasi, terutama bila informasi tersebut sedang dibutuhkan. Karena itu *database* harus melalui proses komputasi dalam pengelolaannya [9].

Dalam *database*, integritas data merupakan jaminan konsistensi data terhadap semua status konstrain yang diberlakukan pada data tersebut, sehingga memberikan jaminan keabsahan pada data itu sendiri. Beberapa integritas data meliputi integritas entitas, integritas referensial, konstrain domain, dan *enterprise constraint* [10].

Keamanan *database* adalah pemberian perlindungan pada *database* terhadap berbagai bentuk ancaman dan gangguan, baik yang bersifat teknis maupun administrasi. Hal ini penting karena seringkali terdapat gangguan yang sangat bervariasi terhadap *database*, dapat meliputi *hardware*, *software*, manusia, dan data. Secara keseluruhan, gangguan terhadap *database*, baik fisik maupun nonfisik, meliputi pencurian data, hilangnya kerahasiaan data, kehilangan integritas data, dan kehilangan kemampuan data [10].

Untuk memberi perlindungan keamanan pada *database* dapat dilakukan beberapa cara, antara lain pemberian otoritas pada pengguna untuk mengakses objek-objek dalam *database*.

Aspek-aspek layanan keamanan data yang ada dalam suatu *database* adalah:

a. Kerahasiaan (*confidentiality*). Layanan untuk menjaga agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak.
b. Integritas data (*data integrity*). Layanan yang menjamin bahwa pesan masih asli/utuh, atau belum pernah dimanipulasi selama pengiriman.
c. Otentikasi (*authentication*). Layanan yang berhubungan dengan identifikasi, baik

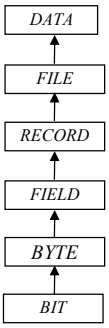
mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication*) maupun mengidentifikasi kebenaran sumber pesan (*data origin authentication*).

d. Penyangkalan (*non-repudation*). Layanan untuk mencegah entitas yang berkomunikasi untuk melakukan penyangkalan, yaitu pengirim pesan menyangkal telah melakukan pengiriman pesan, atau penerima pesan menyangkal telah menerima pesan yang dikirimkan.

2.2.2 Hirarki Data Dalam *Database*

Data-data didalam *database* memiliki suatu susunan tertentu yang berbentuk hirarki. Dalam hirarki ini, *bit* merupakan bagian terkecil dari seluruh data, berupa karakter *American Standar Code Form Information Intercharge* (ASCII). Sekumpulan *bit* akan membentuk *byte/attribute* dari *field* yang berupa huruf, yang membentuk nilai sebuah *field* yang merupakan tingkatan dasar dari *database*. Jadi, *field* mewakili atribut data, seperti kode anggota, nama anggota, alamat anggota, dan sebagainya. Kumpulan *field* tersebut akan membentuk sebuah *record*, sedangkan kumpulan *record* akan membentuk *file* (tabel), dan sekumpulan *file* akan membentuk *database.*

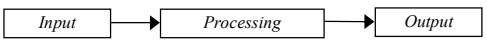
Urutan data dalam hirarki tersebut dapat dilihat pada gambar sebagai berikut [11]:



Gambar 2 Hirarki Data Dalam *Database*

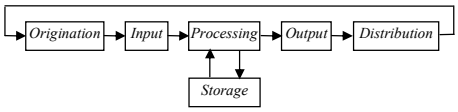
2.2.2 Sistem Pengolahan Data Dalam *Database*

Proses pengolahan data dalam *database* terdiri dari tiga tahapan dasar yang disebut siklus pengolahan data (*data processing cycle*), yaitu *input*, *processing*, dan *output* sebagai berikut [11]:



Gambar 3 *Data Processing Cycle*

Ketiga tahap dasar tersebut dapat dikembangkan lebih lanjut sehingga disebut siklus pengolahan data yang dikembangkan (*expanded data processing cycle*). Dalam pengembangan siklus pengolahan data ini akan ditambahkan lagi tiga tahapan, yaitu *origination*, *storage*, dan *distribution* sebagai berikut [11]:



Gambar 4 *Expanded Data Processing Cycle*

Penjelasan tahap-tahap dalam siklus pengolahan data yang dikembangkan yaitu [11]:

a. Pengorganisasian (*Origination*), berhubungan dengan pengumpulan data yang merupakan proses pencatatan (*recording*) data kedalam dokumen dasar.
b. Masukan (*Input*), merupakan proses untuk memasukkan data-data yang telah terkumpul kedalam proses komputer dengan *input device*.
c. Pemrosesan (*Processing*), merupakan proses pengolahan data yang dilakukan *processing device*, berupa proses hitung, banding, klasifikasi, urutan, kendali, atau cari didalam penyimpanan (*storage*).
d. Keluaran (*Output*), merupakan proses menghasilkan *output* dari hasil pengolahan data kedalam *output device*, yaitu berupa informasi.
e. Penyaluran/Pengiriman (*Distribution*), yaitu proses penyaluran/pengiriman *output* yang dihasilkan kepada pihak-pihak yang berhak dan membutuhkan informasi tersebut.
f. Penyimpanan (*Storage*), merupakan proses perekaman hasil pengolahan kedalam simpanan luar (*storage*). Hasil pengolahan yang disimpan dalam *storage* tersebut dapat dipergunakan sebagai bahan *input* untuk proses selanjutnya.

2.3 Kriptografi

2.3.1 Sejarah Kriptografi

Kriptografi dimulai pertama kali dengan metode pertukaran posisi untuk mengenkripsi pesan tertentu. Dalam sejarah dikatakan bahwa Julius Caesar selalu mengacak isi pesannya sebelum diserahkan pada para kurir. Karena itu ada yang mengatakan bahwa teknik Julius Caesar dianggap sebagai awal penggunaan kriptografi. Sesungguhnya, kriptografi telah digunakan oleh bangsa Mesir pada 4.000 tahun lalu dan masih terus digunakan hingga saat ini.

Saat ini kriptografi masih dibincangkan secara luas karena dapat digunakan sebagai suatu alat untuk melindungi kerahasiaan dan strategi-strategi yang digunakan suatu negara. Sebagian besar sejarah kriptografi adalah sejarah kriptografi klasik, yaitu metode enkripsi yang menggunakan kertas dan pensil atau menggunakan bantuan alat mekanik yang sangat sederhana.

Secara umum algoritma kriptografi klasik dikelompokkan menjadi dua kategori, yaitu [12]:

a. Algoritma *Cipher* Transposisi, dilakukan dengan cara mengubah susunan huruf-huruf yang ada didalam suatu pesan.

b. Algoritma *Cipher* Substitusi, dilakukan dengan cara mengganti setiap huruf atau kelompok huruf yang ada didalam suatu pesan dengan sebuah huruf atau kelompok huruf lain.

Kedua pengelompokan algoritma kriptografi ini karena sejarah kriptografi klasik mencatat penggunaan algoritma *cipher* transposisi oleh tentara Sparta di Yunani pada awal tahun 400 SM saat menggunakan suatu alat yang disebut *scytale*. Alat ini terdiri dari sebuah kertas panjang seperti pita dari daun *papyrus* yang dililitkan pada sebuah tabung selinder yang diameter tertentu. Diameter tabung selinder ini yang menyatakan kunci penyandian pesan. Pesan yang dikirimkan ditulis secara horizontal, baris per baris. Bila pita kertas ini dilepas dari tabung selinder, maka huruf-huruf dalam pita kertas telah tersusun acak membentuk pesan rahasia. Untuk membaca pesan, si penerima harus melilitkan kembali pita kertas tersebut pada tabung selinder yang berdiameter sama dengan diameter tabung selinder si pengirim.

Adapun penggunaan awal dari algoritma *cipher* substitusi dan paling sederhana adalah *Caesar Cipher* yang digunakan oleh Julius Caesar. Caranya dengan mengganti setiap karakter pesan yang ditulis dengan karakter yang terletak pada tiga posisi berikutnya dalam susunan alfabet untuk keamanan pesan yang dikirim.

Kalangan Gereja juga menggunakan teknik kriptografi di awal berkembangnya agama Kristen untuk menjaga tulisan religius yang ada dari gangguan otoritas politik atau budaya yang dominan berkuasa saat itu. Saat itu metode yang terkenal adalah Angka si Buruk Rupa (*Number of the Beast*) yang ada didalam Kitab Perjanjian Baru, yaitu angka "666". Angka ini menyatakan cara kriptografi menyembunyikan pesan yang dianggap berbahaya, dan para ahli percaya bahwa pesan ini mengacu pada Kerajaan Romawi [12].

Kriptografi juga digunakan di India oleh para pecinta untuk berkomunikasi tanpa diketahui orang lain. Metode ini banyak digunakan oleh masyarakat yang dibuktikan dengan ditemukan kriptografi dalam buku Kama Sutra yang merekomendasikan agar kaum wanita seharusnya mempelajari seni dengan memahami *cipher.*

Pada abad ke-15 Leon Battista Albertini menemukan metode kriptografi yang disebut Kode Roda (*Wheel Cipher*) yang terdiri dari dua buah potongan silendris, yaitu silendris dalam dan silendris luar, yang dikenal dengan sebutan *cipher disk*. Setiap silendris memiliki seluruh lebel alfabet dengan susunan yang tidak berurutan dan sama. Silendris luar merupakan alfabet untuk teks-kode dengan metode *monoalphabetic substitution cipher alphabet,* yaitu metode dimana satu karakter didalam teks asli diganti dengan satu karakter bersesuaian atau fungsi satu ke satu.

Metode ini dikembangkan oleh Thomas Jefferson dan diberi nama Roda Kode Jefferson (*Jefferson's Wheel Cipher*) yang selanjutnya dikembangkan lagi oleh Bazeries hingga diberi nama *Silinder Bazerries*. Metode terakhir ini lebih fleksibel dari metode sebelumnya karena memungkinkan untuk dikembangkan secara terus menerus untuk menghindari *code breaking*. *Silinder Bazerries* kemudian dipecahkan oleh Deviaris pada tahun 1893 tetapi metode ini tetap dikembangkan dan masih dianggap aman untuk digunakan pada kasus-kasus tertentu hingga kini.

Pada abad ke-17, kriptografi menyebabkan Ratu Skotlandia, yaitu Queen Mary, menjadi salah satu korban hukuman mati dengan dipancung. Hukuman ini ditetapkan setelah ditemukan sebuah surat rahasia milik sang Ratu di balik penjara yang berhasil dipecahkan oleh seorang ahli pemecah kode. Surat rahasia Queen Mary ini merupakan sebuah surat terenkripsi dan memuat tentang rencana pembunuhan yang akan dilakukan terhadap Ratu Inggris, yaitu Ratu Elizabeth I.

Pada abad ke-20 kriptografi lebih banyak digunakan oleh kalangan militer pada perang dunia II, dimana pihak Nazi Jerman membuat mesin enkripsi yang dinamakan *Enigma.* Mesin ini menggunakan beberapa *rotor* (roda berputar) dan melakukan proses enkripsi yang sangat rumit. Pihak Nazi Jerman percaya bahwa pesan-pesan dikirim dengan *Enigma* tidak akan terpecahkan. Anggapan ini ternyata salah karena setelah mempelajari *Enigma* selama bertahun-tahun, pihak Sekutu berhasil memecahkan kode tersebut. Saat pihak Nazi Jerman mengetahui bahwa kode rahasia mereka telah terpecahkan, mereka melakukan beberapa kali perubahan pada *Enigma*.

Mesin enkripsi *Enigma* yang digunakan oleh Nazi Jerman termasuk dalam kriptografi berbasis *rotor* dan dapat mengenkripsi satu pesan dengan 15 milyar kemungkinan. Mesin berbasis *rotor* sebenarnya telah dibangun dan dipatenkan oleh beberapa Penemu dari beberapa negara yang berbeda sejak tahun 1917 hingga 1921. Antara lain adalah Edward Hug Hebern (Amerika), Arthur Scherbius (Jerman), Alexander Koch (Belanda), dan Arvid Gerhard Damm (Swedia).

Mesin rotor yang dibangun dan dipatenkan Koch kemudian dikembangkan untuk versi militer oleh Arthur Scherbius dan dipatenkan dengan nama *Enigma*. Diperkirakan mesin *Enigma* yang digunakan selama periode tahun 1935 hingga 1945 berjumlah 100.000 mesin.

Perkembangan paling pesat dan berpengaruh dalam sejarah kriptografi terjadi pada tahun 1976 saat Whitfield Diffie dan Martin Hellman mempublikasikan tesis berjudul *New Direction in Cryptography*. Tesis ini diperkenalkan konsep kunci publik kriptografi yang revolusioner dan

metode baru dalam pertukaran kunci, yaitu keamanan berdasarkan algoritma diskrit.

Di tahun 1978, Rivest, Shamir, dan Adleman menemukan enkripsi kunci publik yang pertama, yang dikenal sebagai RSA (Rivest, Shamir, *and* Adleman). Skema RSA berdasarkan permasalahan matematika rumit yang terdiri dari pemfaktoran bilangan-bilangan yang besar nilainya. Salah satu sumbangan terpenting dari kriptografi kunci publik ini adalah tanda tangan digital. Pada tahun 1991, standar internasional pertama untuk tanda tangan digital adalah berdasarkan pada skema kunci publik RSA [13].

Kriptografi memiliki sejarah yang panjang sejak munculnya teknik kriptografi klasik hingga berkembangnya algoritma-algoritma baru sebagai kriptografi modern. Saat ini berbagai standar dan infrastruktur yang terkait teknik kriptografi terus dikembangkan dan dibangun untuk memenuhi kebutuhan akan keamanan data dan informasi.

2.3.2 Pengertian dan Tujuan Kriptografi

Saat ini kriptografi telah menjadi suatu bagian penting di bidang teknologi informasi karena hampir semua penerapannya menggunakan kriptografi sebagai alat yang menjamin keamanan serta kerahasiaan data dan informasi. Karena itu kriptografi menjadi salah satu bidang ilmu yang berkembang pesat dan dalam waktu singkat banyak muncul algoritma-algoritma baru yang lebih unggul dari algoritma pendahulunya.

Kata kriptografi (*cryptography*) berasal dari bahasa Yunani, yaitu *cryptós* yang berarti rahasia, dan *gráphein* yang berarti tulisan [13]. Karena itu secara umum kriptografi diartikan sebagai tulisan rahasia. Namun beberapa definisi kriptografi yang dikemukakan para pakar, definisi pada tahun 80-an menyatakan bahwa *cryptography is the art and science of keeping messages secure* yang berarti kriptografi merupakan suatu ilmu sekaligus seni untuk menjaga keamanan suatu pesan [12]. Penggunaan kata seni ini berasal dari fakta bahwa di awal sejarah kriptografi, setiap orang memiliki cara yang unik untuk merahasiakan suatu pesan.

Sedangkan definisi dalam beberapa buku terbaru menyatakan bahwa kriptografi adalah suatu studi teknik matematika yang berhubungan dengan keamanan informasi seperti kerahasiaan, integritas data, otentikasi entitas, dan otentikasi keaslian data [13]. Kriptografi yaitu ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan keamanan informasi seperti kerahasiaan, integritas data, serta otentikasi [12].

Jadi, kriptografi adalah teknik pengamanan informasi, dimana informasi diubah dengan kunci tertentu melalui proses enkripsi sehingga menjadi bentuk informasi baru yang tidak dapat dipahami oleh orang yang tidak berhak. Informasi baru ini hanya dapat diubah menjadi pesan aslinya oleh orang yang berhak melalui proses dekripsi.

Selain untuk menjaga kerahasiaan pesan, kriptografi juga menangani permasalahan dalam keamanan data/informasi yang mencakup [13]:

a. Keabsahan pengirim (*user authentication*). Hal ini berkaitan dengan keaslian pengirim.
b. Keaslian pesan (*message authentication*). Hal ini berkaitan dengan keutuhan pesan.
c. Anti-penyangkalan (*non-repudiation*). Hal ini berkaitan dengan pengirim pesan tidak dapat menyangkal bahwa dia yang mengirim pesan.

2.3.3 Terminologi Dalam Kriptografi

Dalam kriptologi terdapat beberapa terminologi sebagai berikut [12]:

a. Pesan; Plainteks (*Plaintext*), dan Ciperteks (*Ciphertext*).
   Pesan adalah data/informasi yang dapat dibaca dan dipahami maknanya. Dalam kriptologi, nama lain pesan adalah *plaintext*. Pesan ini berupa data/informasi yang dikirim atau yang disimpan dalam suatu media perekaman. Agar pesan tidak dapat dipahami oleh pihak yang tidak berkepentingan, pesan ini disandikan dengan suatu kunci tertentu sehingga menjadi bentuk lain. Bentuk pesan yang tersandikan disebut *ciphertext*. Agar isi *ciphertext* dapat dipahami, harus ditransformasikan kembali menjadi *plaintext* dengan kunci yang sesuai.
b. Enkripsi (*encryption/enchipering*) dan dekripsi (*decryption/deciphering*)
   Proses menyandikan *plaintext* menjadi *ciphertext* disebut enkripsi, sedangkan proses mengembalikan *ciphertext* menjadi *plaintext* disebut dekripsi sesuai standar nama menurut ISO 7498-2. Dikatakan juga, enkripsi adalah proses yang melakukan perubahan sebuah kode dimengerti menjadi sebuah kode yang tidak bisa dimengerti/tidak terbaca.
   Dalam kriptografi, enkripsi adalah proses mengamankan informasi dengan membuat informasi tersebut tidak dapat dibaca tanpa bantuan pengetahuan khusus. Enkripsi dapat digunakan untuk keamanan, tetapi teknik lain masih diperlukan untuk membuat komunikasi yang aman, terutama untuk memastikan integritas dan autentikasi pesan.
c. *Cipher* dan Kunci (*Key*)
   Algoritma kriptografi disebut juga *cipher*, yaitu aturan untuk enkripsi dan deskripsi atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Kriptografi modern mengatasi masalah keamanan algoritma kriptografi dengan penggunaan *key* yang merupakan parameter untuk transformasi enkripsi dan dekripsi. *Key* yang biasa digunakan berupa *string* atau deretan bilangan.

d. Sistem kriptografi (*Cryptosystem*)
   Sistem kriptografi adalah terdiri dari algoritma kriptografi, semua *plaintext* dan *ciphertext* yang mungkin, serta *key*.
e. Penyadap (*Eavesdropper*)
   Penyadap adalah orang/pihak yang mencoba untuk menangkap pesan saat ditransmisikan. Tujuannya adalah mendapatkan informasi sebanyak-banyaknya tentang kriptografi yang digunakan dengan maksud untuk memecahkan *ciphertext*. Nama lain penyadap adalah *enemy*, *adversary*, *intruder*, *interceptor*, dan *bad guy*.
f. Kriptanalisis (*Cryptanalysis*) dan Kriptology (*Cryptology*)
   Kriptanalisis adalah ilmu dan seni untuk memecahkan *ciphertext* menjadi *plaintext* tanpa harus mengetahui *key* yang diberikan. Pelakunya disebut kriptanalis. Sedangkan kriptologi adalah studi mengenai kriptografi dan kriptanalisis.

2.3.4 Konsep Matematika Kriptografi

Konsep matematis yang mendasari algoritma kriptografi adalah relasi antara dua himpunan, yaitu himpunan elemen-elemen *plaintext* dan himpunan *ciphertext*. Adapun enkripsi dan dekripsi merupakan fungsi yang memetakan elemen-elemen diantara kedua himpunan tersebut [13].

Misalkan P menyatakan *plaintext* dan C menyatakan *ciphertext*, maka fungsi enkripsi E memetakan P ke C adalah E(P) = C, dan fungsi dekripsi D memetakan C ke P adalah D(C) = P.

Karena proses enkripsi menyandikan pesan dan dekripsi mengembalikan pesan tersandi ke bentuk pesan asal maka persamaannya adalah D(E(P)) = P.

Dengan menggunakan *key* (K), maka fungsi enkripsi menjadi EK(P) = C, sedangkan fungsi dekripsi menjadi DK(C) = P, sehingga kedua fungsi tersebut memenuhi persamaan DK(EK(P)) = P.

Untuk jelasnya, skema enkripsi dan dekripsi dengan kunci tersebut sebagai berikut [13]:

Gambar 5 Skema Enkripsi dan Deskripsi

2.4 Algoritma

2.4.1 Pengertian Algoritma

Kata algoritma memiliki sejarah yang aneh bila ditinjau dari asal usul katanya. Para ahli menemukan bahwa kata *algorism* berarti proses menghitung dengan angka Arab dan seseorang disebut sebagai *algorist* bila ia menggunakannya.

Para ahli bahasa berusaha menemukan asal usul kata ini namun hasilnya kurang memuaskan. Akhirnya ahli sejarah matematika menemukan bahwa asal kata ini berasal dari buku seorang penulis terkenal berbangsa Arab, yaitu Abu Ja'far Muhammad Ibnu Musa Al-Khuwarizmi yang hidup pada tahun 770 - 840 M. Ia menulis sebuah buku berjudul Kitab Aljabar Walmuqabala yang berarti Buku Pemugaran dan Pengurangan (*The Book off Restoration and Reduction*). Dari judul buku ini memperoleh akar kata Aljabar (*Algebro*).

Karena kata *algorism* sering dikelirukan dengan kata *arithmetic* maka dilakukan perubahan dari *algorism* menjadi *algorithm*, dimana akhiran *sm* berubah menjadi *thm*. Selanjutnya karena perhitungan dengan angka Arab dapat digunakan, lambat laun kata *algorithm* berangsur-angsur digunakan sebagai suatu metode perhitungan (komputasi) secara umum sehingga kehilangan makna kata aslinya. Dalam bahasa Indonesia, kata *algorithm* diserap menjadi kata algoritma.

Jadi, algoritma adalah urutan langkah-langkah penyelesaian masalah yang disusun secara sistematik dan logis [12]. Kata logis merupakan kata kunci dalam sistem algoritma karena langkah-langkah dalam algoritma harus logis dan harus dapat ditentukan apakah bernilai salah atau benar.

2.4.2 Algoritma *One Time Pad*

Algoritma *One Time Pad* (OTP) ditemukan pada tahun 1917 oleh Major Yoseph Mouborgne dan Gilbert Vernam pada perang dunia II. Metode ini telah diklaim sebagai satu-satunya algoritma kriptografi sempurna dan tidak dapat dipecahkan. Suatu algoritma dikatakan aman bila tidak ada cara untuk menemukan *plaintext*-nya. Hingga saat ini hanya algoritma OTP yang dinyatakan tidak dapat dipecahkan meskipun menggunakan sumber daya yang tidak terbatas.

Algoritma OTP adalah salah satu algoritma konvensional dan merupakan algoritma untuk mengengkripsi berbagai informasi seperti gambar, tulisan, dan sebagainya. Penggunaan algoritma OTP dalam kriptografi adalah sebagai dasar untuk mengaburkan suatu informasi yang ingin dirahasiakan dengan cara mengacak informasi tersebut sehingga menjadi suatu informasi yang tidak dapat dipahami oleh orang lain [14].

Algoritma OTP adalah algoritma berjenis *symmetric key*, artinya kunci yang digunakan untuk melakukan enkripsi dan dekripsi merupakan kunci yang sama. Dalam proses enkripsi, algoritma OTP menggunakan cara *stream cipher* dimana *cipher* berasal dari hasil XOR antara *bit plaintext* dan *bit key* [15].

Prinsip enkripsi pada algoritma OTP adalah dengan mengkombinasikan setiap karakter *plaintext* dengan satu karakter *key*. Karena itu, panjang *key* harus sama dengan panjang *plaintext*.

Secara teoritis, tidak mungkin untuk mendekripsi *ciphertext* tanpa kuncinya karena bila *key* yang digunakan adalah *key* yang salah maka yang diperoleh bukan *plaintext* yang seharusnya. Setiap *key* hanya boleh digunakan untuk sekali pesan, pengambilan dilakukan secara acak agar tidak dapat diterka, dan jumlah karakter *key* harus sebanyak jumlah karakter pesan.

Algoritma OTP sering digunakan dalam enkripsi karena prosesnya yang relatif mudah. Fungsi untuk mengenkripsi hanya dengan cara meng-XOR-kan *plaintext* dengan *key* yang telah disiapkan untuk menghasilkan *ciphertext*, yaitu c = p XOR k. Sedangkan fungsi untuk mendekripsi hanya meng-XOR-kan *ciphertext* dengan *key* yang disepakati, yaitu p = c XOR k.

2.5 Sistem Bilangan

2.5.1 Bilangan *Binner*

Sistem bilangan *binner* (sistem bilangan basis dua) adalah sebuah sistem penulisan angka dengan menggunakan dua simbol, yaitu 0 dan 1. Sistem bilangan *binner* modern ditemukan oleh Gottfried Wilhelm Leibniz pada abad ke-17.

Perbedaan mendasar metode bilangan *binner* dan bilangan desimal adalah berkenaan dengan basis. Bilangan desimal berbasis 10 (X10) dengan berpangkatkan 10x, adapun bilangan *binner* berbasiskan 2 (X2) dengan perpangkatan 2x. Komputer memproses data/program berupa bilangan *binner* yang dinyatakan dalam angka 1 dan 0. Untuk mengkonversi bilangan *binner* ke bilangan desimal adalah mengalikan 2 dengan pangkat N (suku ke-N).

2.5.2 Bilangan *Hexa* Desimal

Bilangan *heksa* desimal yang seringkali disingkat menjadi *heks* adalah bilangan berbasis enam belas (X16). Simbol yang digunakan dalam metode *heksa* desimal adalah 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A(=10), B(=11), C(=12), D(=13), E(=14), dan F(=15).

2.5.3 Aritmatika *Modulo*

Aritmatika *modulo* berbasiskan bilangan genap. Dalam aritmatika *modulo*, misalkan *a* adalah bilangan bulat dan *m* adalah bilangan bulat > 0, maka operasi *a mod m* memberikan sisa jika *a* dibagi dengan *m*. Bilangan *m* disebut *modulus* atau *modulo*, dan hasil aritmatika *modulo m* terletak didalam himpunan {0, 1, 2, …, *m*-1}.

Notasi dari *modulo* adalah *a mod m* = *r* sedemikian rupa sehingga *a* = *mq* + *r*, dengan 0 < *r* < *m*. Beberapa contoh operasi dengan operator *modulo* sebagai berikut:

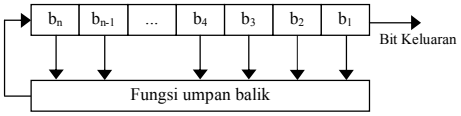
(i) 23 mod 5 = 3 (23 = 5.4+3)
(ii) 27 mod 3 = 0 (27= 3.9+0)
(iii) 6 mod 8 = 6 (6 = 8.0 + 6)
(iv) 0 mod 12 = 0 (0 = 12.0+6)

2.6 *Linear Feedback Shift Register*

Pembangkit aliran kunci yang sering digunakan dalam kriptografi adalah *Feedback Shift Register* (FSR) yang disebut register geser dengan umpan balik. Kriptografi berbasis register geser telah digunakan pihak militer sejak awal penggunaan alat elektronik. FSR terdiri dari [16]:

a. Register geser, yaitu barisan *bit-bit* (b_n, b_n-1, ..., b_4, b_3, b_2, b_1) yang panjangnya n (disebut register geser n-*bit*).
b. Fungsi umpan balik, yaitu fungsi yang menerima masukan dari register geser dan mengembalikan nilai fungsi ke register geser.

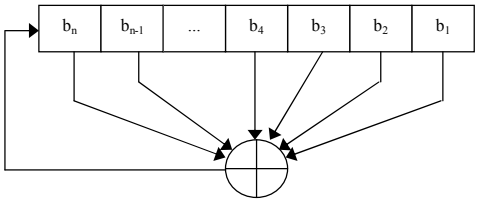
Bagian-bagian FSR digambarkan sebagai berikut [16]:



Gambar 6 Bagian-Bagian FSR

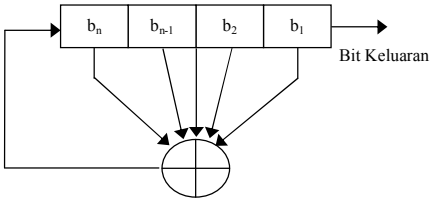
Setiap kali sebuah *bit* dibutuhkan, semua *bit* didalam register digeser 1 *bit* ke kanan. *Bit* yang paling kiri (bn) dihitung sebagai fungsi *bit-bit* lain yang ada dalam register. Keluaran dari register geser adalah 1 *bit*, yaitu *bit* b1 yang tergeser. *Bit* keluaran ini yang menjadi kunci enkripsi.

Adapun periode register geser adalah panjang barisan keluaran sebelum register geser berulang kembali. Contoh dari FSR adalah *Linear Feedback Shift Register* (LFSR). Fungsi umpan baliknya adalah peng-XOR-an *bit-bit* tertentu dalam register sebagai berikut [16]:



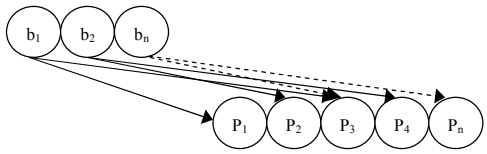
Gambar 7 LFSR Sederhana

(LFSR n-bit mempunyai 2n-1 status internal (keadaan isi register). Secara teoritis, LFSR dapat membangkitkan 2n-1 barisan *bit* acak semu sebelum perulangan. Jadi periode maksimal LFSR adalah 2n-1 sebagai berikut [16]:



Gambar 8 LFSR 2-*Bit*

Gambar atas adalah contoh LFSR 2-*bit* yang fungsi umpan balik meng-XOR-kan b_1 dan b_2, yaitu $b = f(b_1, b_n)$ xor .. Sebagai contoh, bila register diinisialisasi dengan 21 (akan dikonversi ke bentuk *binner*), maka isi register geser akan menggeser dari kiri ke arah kanan dan kembali diulangi sampai ke P_n sebagai berikut [16]:

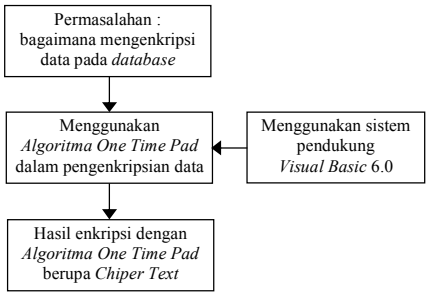


Gambar 9 Inisialisasi Kunci Masukan

Dari gambar diatas dapat dilihat bahwa priode kunci yang digunakan akan berulang setelah kunci mencapai angka terakhir n dan kembali ke awal mengikuti panjang *plaintext*.

2.7 Kerangka Pikir Penelitian

Berdasarkan uraian diatas maka kerangka pikir penelitian ini sebagai berikut :



Gambar 10 Kerangka Pikir Penelitian

3. Metode Penelitian

Penelitian ini menggunakan jenis penelitian kuantitatif, yaitu penelitian untuk memperoleh penjelasan dari suatu teori dan hukum-hukum realitas yang dikembangkan dengan model-model matematis berdasarkan pendapat yaitu penelitian yang memperoleh data yang berbentuk angka atau kualitatif yang diangkakan [17].

Penelitian ini juga dikategorikan dalam tipe penelitian eksperimen kriptografi yang tujuannya adalah merancang sebuah aplikasi yang dapat mengenkripsi data pada *database* sesuai pendapat bahwa penelitian eksperimen adalah penelitian yang sistematis, logis dan teliti dalam melakukan kontrol terhadap suatu kondisi [18].

Teknik pengumpulan data yang digunakan adalah:

a. Observasi, yaitu melakukan pengamatan terhadap *database* STMIK Bina Mulia Palu, khususnya data mahasiswa Jurusan TI Angkatan 2012 (Kelas B) yang akan digunakan dalam ujicoba pengenkripsian data.
b. Wawancara (*Interview*), melakukan tanyajawab dengan Kepala BAAK dan Pengelola PDPT STMIK Bina Mulia Palu.
c. Dokumentasi, yaitu mempelajari data-data yang menjadi sampel dalam penelitian ini.
d. Studi Pustaka, yaitu mempelajari berbagai literatur yang terkait dengan penelitian ini.

Tahap pengembangan kriptografi dalam penelitian ini sebagai berikut:

a. Pengumpulan bahan-bahan yang terkait dengan sistem pengamanan *database* menggunakan algoritma OTP yang diperoleh dari berbagai literatur dan artikel, serta penelitian terdahulu yang mendukung penelitian ini.
b. Penyusunan semua bahan yang telah diperoleh sesuai dengan prosedur penggunaan algoritma OTP dalam pengamanan *database* dengan bahasa pemprograman *Visual Basic* 6.0.
c. Pengumpulan dan perancangan alat bantu perancangan *software*, yaitu *Flowchart*, *Data Flow Diagram* (DFD), *Acces Database*, dan *Visual Basic* 6.0.

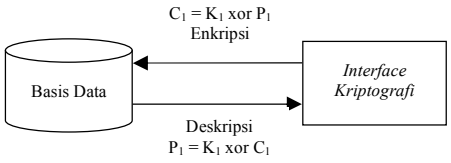
Metode analisis data dalam penelitian ini terdiri dari beberapa langkah sebagai berikut:

a. Reduksi Data. Dari data-data yang terkumpul, perlu pemuatan rangkuman data inti, yaitu data yang diperlukan sehingga tetap dalam data. Proses ini memerlukan pembuangan data yang tidak diperlukan dalam analisis selanjutnya.
b. Penyusunan Data. Rangkuman data yang diperoleh disusun berdasarkan indikator, pemilihan data-data sensitif yang memerlukan perlindungan data.
c. Pembuatan *Database*. Data-data yang telah dirangkum disusun dalam suatu *database* yang telah diberi pengamanan data.

4. Hasil Penelitian

4.1 Analisis Sistem Yang Diusulkan

Penerapan teknik kriptografi menggunakan algoritma OTP diusulkan rancangan sistem yang menggunakan bahasa pemograman *Visual Basic* 6.0 dan *microsoft office acces* 2007 sebagai media penyimpanan dengan desain sebagai berikut:



Gambar 11 Desain Sistem yang Diusulkan

4.1.1 Proses Enkripsi-Dekripsi OTP

Enkripsi dalam penelitian ini dinyatakan sebagai penjumlahan *modulo* 26 dari satu karakter *plaintext* dengan satu karakter *key* algoritma OTP sebagai berikut:

$$C_i = (P_i + K_i) \; mod \; 26$$

P_i adalah *plaintext* ke-i dan C_i adalah huruf *ciphertext* ke-i. Panjang *key* sama dengan panjang *plaintext*, sehingga tidak ada pengulangan penggunaan *key* selama proses enkripsi.

Angka 26 muncul karena sistem ini menggunakan abjad, artinya hanya abjad A – Z yang dapat dikodekan dengan sistem ini. Bila diinginkan pengkodean sembarang data (teks, gambar, suara, atau video), sistem ini diperluas dengan penggunaan sistem bilangan *binner* karena semua tipe data dapat dianggap data *binner* dan karena bilangan *binner* hanya mengenal 0 dan 1, maka basis 26 diubah menjadi basis 2.

Penjumlahan *modulo* 2 dinyatakan dengan XOR yang sering digunakan dalam sistem digital. *Ciphertext* dari penjumlahan *modulo* 2 adalah satu *bit plaintext* dengan satu *bit key*, yaitu:

$$Ci = (Pi + Ki) \; mod \; 2$$

P_i adalah *bit plaintext*, K_i adalah *bit key*, dan C_i adalah *bit ciphertext*.

Sedangkan *plaintext* dari penjumlahan *modulo* 2 adalah satu *bit ciphertext* dengan satu *bit key*, yaitu:

$$P_i = (C_i + K_i) \; mod \; 2$$

Mengingat operasi penjumlahan *modulo* 2 adalah identik dengan operasi *bit* yang menggunakan operator XOR, maka persamaan enkripsi ditulis sebagai berikut:

$$C_i = P_i \text{ xor } K_i$$

sedangkan dekripsi ditulis sebagai berikut:

$$P_i = C_i \text{ xor } K_i$$

Pada proses enkripsi, *bit* hanya mempunyai dua nilai sehingga proses enkripsi hanya menyebabkan dua keadaan pada *bit*, yaitu berubah atau tidak berubah. Kedua keadaan ini ditentukan oleh *key* enkripsi yang disebut aliran kunci (*keystream*) dari sebuah pembangkit pembangkit aliran kunci (*keystream generator*).

Selanjutnya *keystream* di-XOR-kan dengan aliran *bit-bit plaintext* P_1, P_2, ... P_i untuk menghasilkan aliran *bit-bit ciphertext*, yaitu:

$$C_i = P_i \text{ xor } K_i$$

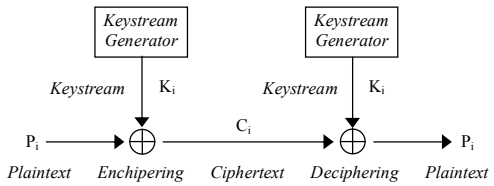
Di sisi penerima, *bit-bit ciphertext* kemudian di-XOR-kan dengan *keystream* yang sama untuk menghasilkan *bit-bit plaintext*, yaitu:

$$P_i = C_i \text{ xor } K_i$$

Karena proses enkripsi sebanyak dua kali berturut-turut maka akan menghasilkan kembali *plaintext* semula.

Dalam skema global algoritma OTP, *keystream* menghasilkan elemen *bit key* (K_i) yang kemudian di-XOR-kan dengan *bit plaintext* (P_i) sehingga menghasilkan *bit ciphertext* (C_i). Di sisi penerima, *keystream* yang sama di-XOR-kan dengan *bit ciphertext* (C_i) sehingga menghasilkan *bit plaintext* (P_i) semula.

Skema global algoritma OTP tersebut digambarkan sebagai berikut:



Gambar 12 Konsep Algoritma OTP

Secara teoritis, syarat yang harus dipenuhi untuk merancang suatu *unbreakable cipher*, yaitu *key* harus dipilih secara acak dimana setiap *key* memiliki peluang yang sama untuk terpilih, dan panjang *key* harus sama dengan panjang *plaintext* yang akan dienkripsi. Kedua syarat ini menyebabkan *plaintext* yang sama belum tentu dienkripsi menjadi *ciphertext* yang sama. Kriptanalisis akan menemukan bahwa *ciphertext* yang dideskripsikannya menghasilkan beberapa *plaintext* yang berbeda, sehingga membingungkan dalam menentukan *plaintext* mana yang benar.

Misalnya sebuah *plaintext*, yaitu LAMRO memiliki sebuah *key*, yaitu CRASH. Perlu diingat, panjang *key* harus sama dengan *plaintext* dan tidak ada karakter yang diulang.

Untuk melakukan enkripsi, pertama-tama harus dicari kode ASCII *plaintext* (LAMRO) dan diubah ke bentuk *binner* sebagai berikut:

Tabel 1 Kode ASCII dan Notasi *Binner Plaintext*

Karakter *Plaintext*	ASCII	Notasi *Binner*
L	76	0100 1100
A	65	0100 0001
M	77	0100 1101
R	82	0101 0010
O	79	0100 1111

Kemudian dicari kode ASCII *key* (CRASH) dan diubah ke bentuk *binner* sebagai berikut:

Tabel 2 Kode ASCII dan Notasi *Binner Key*

Karakter *Key*	ASCII	Notasi *Binner*
C	67	0100 0011
R	82	0101 0010
A	65	0100 0010
S	83	0101 0011
H	72	0100 1000

Notasi *binner key* kemudian di-XOR-kan dengan notasi *binner plaintext* sehingga menghasilkan *ciphertext* sebagai berikut:

Tabel 3 Hasil Proses XOR *Plaintext* dan *Key*

Ciphertext
0000 1111 = SI
0001 0011 = D3
0000 1100 = FF
0000 0001 = SOH
0000 0111= BEL

Adapun proses dekripsi pesan juga melakukan operasi yang sama, yaitu XOR antara *ciphertext* dengan *key* sebagai berikut:

Tabel 4 Kode ASCII dan Notasi *Biner* Deskripsi

Ciphertext	*Key*	*Plaintext*
0000 1111 = SI	C = 0100 0011	L =0100 1100
0001 0011 = D3	R = 0101 0010	A =0100 0001
0000 1100 = FF	A = 0100 0001	M =0100 1101
0000 0001 = SOH	S = 0101 0011	R =0101 0010
0000 0111= BEL	H = 0100 1000	O =0100 1111

Setelah pengirim mengenkripsi pesan dengan *key*, ia harus menghancurkan *key* tersebut. Penerima pesan menggunakan *key* yang sama untuk mendekripsikan *ciphertext* menjadi *plaintext* dengan persamaan sebagai berikut:

$$P_i = (C_i + K_i) \bmod 26$$

4.1.2 Algoritma OTP Untuk Sistem Pengaman *Access Database*

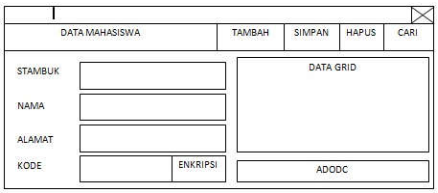
Semua yang bersifat mengamankan dengan metode tertentu merupakan inti kriptografi, yaitu menjamin kerahasiaan informasi menggunakan penyandian. Keutuhan *database* dilakukan dengan algoritma OTP, begitu pula jaminan atas identitas dan keabsahan pihak-pihak yang dapat mengakses *database* tersebut.

Enkripsi pada *level database* dilakukan pada saat data ditulis dan dibaca dari *database* tersebut. Enkripsi ini dilakukan pada kolom-kolom tabel *database*. Pemilihan *field* yang sensitif untuk diproteksi merupakan langkah pertama yang harus dilakukan dalam proses enkripsi dan dekripsi.

Pengenkripsian dengan algoritma OTP juga tergolong sangat baik karena data yang sama tidak akan berubah pengacakannya jika menggunakan *key* yang sama sehingga pengacakan pada tabel relasi tidak akan berbeda-beda bentuknya jika menggunakan *key* yang sama.

Penelitian ini menerapkan proses enkripsi dan deskripsi data mahasiswa Jurusan TI angkatan 2012 (Kelas B) pada STMIK Bina Mulia Palu sebagai sempel eksperimen. Bila data yang sangat sensitif dalam data-data tersebut adalah pada *field* stambuk, maka pengenkripsian dilakukan hanya pada *field* tersebut.

Desain antar muka (*interface*) sistem kriptografi dan media penyimpanan yang dibangun dalam penelitian ini sebagai berikut:



Gambar 13 Desain *Interface* Sistem Kriptografi

Sedangkan rancangan *database* yang digunakan dalam penelitian ini sebagai berikut:

Tabel 5 Rancangan *Database*

No	*Field*	Tipe Data	Lebar
1	STB	TEXT	255
2	NAMA	TEXT	255
3	ALAMAT	TEXT	255

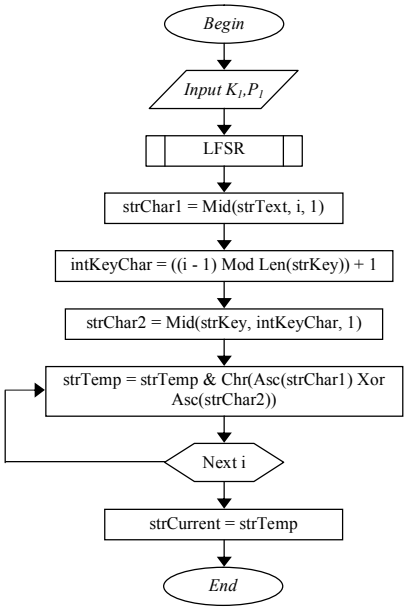
4.2 *Flowchart*

4.2.1 Desain Sistem

Sebelum membuat diagram blok tentang alur enkripsi data dengan algoritma OTP, terlebih dahulu disusun algoritma yang menunjang proses enkripsi data. Algoritma ini sangat penting dalam pengimplentasian pada bahasa pemograman yang digunakan. Langkah-langkah dalam menyusun algoritma OTP sebagai berikut:

a. Masukkan *key* enkripsi dan deskripsi data yang panjangnya tidak melebihi *plaintext*.
b. Untuk pembangkitan *key* acak, dilakukan LFSR sepanjang karakter *key* sebagai berikut:
   1) b_1 sapai b_n diisi oleh *bit-bit key*.
   2) b_1 digeser ke kanan sepanjang 1 *bit* hingga ke b_n.
   3) b_1 akan dijadikan *bit* keluaran.
   4) b_1 dimasukkan kembali menjadi b_n dan sebaliknya.
c. *Bit-bit* keluaran dijadikan *key* baru untuk enkripsi karakter berikutnya.
d. Lakukan *looping* sepanjang karakter *plaintext*.
e. Dalam *looping* terjadi beberapa proses berikut:
   1) Xor-kan panjang karakter *plaintext* dengan panjang *key*.
   2) *Modulo* hasil peng-xor-an tersebut dari karakter pertama hingga karakter terakhir sepanjang *key*.

Diagram blok dari enkripsi data dengan algoritma OTP sebagai berikut:



Gambar 14 Diagram Blok Enkripsi

Untuk dekripsi, proses yang sama dilakukan kembali. *Plaintext* yang digunakan adalah *ciphertext* hasil enkripsi sebelumnya, sedangkan *key* untuk mendekripsi sama dengan *key* yang digunakan pada saat mengenkripsi.

4.3 Implementasi

Visual Basic (VB) 6.0 adalah bahasa pemrograman yang secara cepat dan mudah dapat digunakan untuk membuat aplikasi pada *Microsoft Windows* dan dapat memanfaatkan kemampuan *Microsoft Windows* secara optimal. Kesederhanaan VB 6.0 terletak pada kemudahan menulis bahasa pemrograman dan bentuk tampilan yang dikehendaki. VB 6.0 juga mampu menambah sendiri sebagian kode program secara otomatis kedalam program sehingga pekerjaan *programmer* semakin mudah. Selain itu, VB 6.0 tidak menyulitkan dalam membangun aplikasi.

Karena itu penelitian ini menggunakan VB 6.0 dalam membuat program yang mengamankan *database* dengan algoritma OTP, khususnya enkripsi untuk sistem pengamanan *database*. Program utama proses enkripsi dan dekripsi data dengan algoritma OTP yang telah dibahas diatas kemudian ditulis kembali menggunakan bahasa pemrograman VB 6.0.

4.4 Uji Coba Algoritma OTP

Setelah algoritma OTP diimplementasikan maka selanjutnya dilakukan uji coba pada sistem, apakah telah sesuai dengan yang diinginkan atau belum. Hasil uji coba enkripsi data dengan menggunakan algoritma OTP dengan bahasa pemrograman VB 6.0 sebagai berikut:

mahasiswa

STB	NAMA	ALAMAT
120302069	Nurlaela Singgani	Palu
120302093	Muh. Arianto	Palu
120302098	Monewati Ibrahim	Palu
120302119	Lamro Herianto Siregar	Palu
130302142	Siti mawar	Mamboro

Gambar 15 *Field* Nomor Stambuk Pada *Database* Sebelum Dienkripsi

mahasiswa

STB	NAMA	ALAMAT	Add New Field
]R]A_^PY@	Siti mawar	Mamboro	
]S]A_^P\K	Lamro Heriant	Palu	
]S]A_^Q[K	Nurlaela Singg	Palu	
]S]A_^QTA	Muh. Arianto	Palu	
]S]A_^QTJ	Monewati Ibra	Palu	

Gambar 16 *Field* Nomor Stambuk Pada *Database* Setelah Dienkripsi

Gambar 17 Data *Field* Nomor Stambuk Mahasiswa Sebelum Dienkripsi

Gambar 18 Data *Field* Nomor Stambuk Mahasiswa Setelah Dienkripsi

5. Kesimpulan

Hasil penelitian ini menunjukkan bahwa penerapan teknik kriptografi pada *database* menggunakan algoritma OTP dapat dilakukan untuk mengamankan data-data yang ada didalam *database*. Teknik kriptografi tersebut dapat diimplementasikan dengan menggunakan bahasa pemograman *Microsoft Visual Basic* 6.0 yang dapat mengenkripsi dan mendeskripsi data dengan memasukkan kode yang telah disepakati.

6. Penutup

Mekanisme dalam teknik kriptografi pada penelitian ini masih sangat sederhana, namun diharapkan dapat berguna sebagai langkah awal untuk masuk kedalam dunia kriptografi, khususnya untuk penerapan kriptografi pada *database* menggunakan algoritma OTP dan bahasa pemograman *Microsoft Visual Basic*. Diharapkan penelitian ini dapat dikembangkan agar lebih baik lagi dan dapat dimanfaatkan serta diterapkan pada bidang-bidang kehidupan yang lebih kompleks lagi.

Daftar Pustaka

[1] Kadir, Abdul. 2015. *Pengertian Informasi Menurut Para Ahli*.www.sarjanaku. com.

[2] Jogiyanto, H. M. 2010. *Analisis dan Desain Sistem Informasi; Pendekatan Terstruktur, Teori dan Praktek Aplikasi Bisnis.* Yogyakarta: Andi Offset.

[3] Ladjamudin, Al-Bahra Bin. 2013. *Analisis dan Desain Sistem Informasi.* Jakarta: Graha Ilmu.

[4] Irmansyah, Faried. 2013. *Dasar-Dasar Algoritma dan Pemrograman.* Yogyakarta: Andi Offset.
[5] Jogiyanto, H. M. 2011. *Sistem Teknologi Informasi; Pendekatan Terintegrasi, Konsep Dasar, Teknologi, Aplikasi, Pengembangan, dan Pengelolaan.* Yogyakarta: Andi Offset.
[6] Bejo, Orang. 2012. *Pengertian Data Base Menurut Para Ahli*. www.orangbejo.com.
[7] Marlinda, L. 2004. *Sistem Basis Data.* Yogyakarta: Andi Offset.
[8] Hariyanto, B. 2004. *Sistem Manajemen Basis Data*. Bandung: Informatika.
[9] Muiz. 2015. *Pengertian Sistem Basis Data Menurut Para Ahli*. www.dosenit.com.
[10] Simarmata, J. 2007. *Perancangan Basis Data.* Yogyakarta: Andi Offset.
[11] Jogiyanto, H. M. 2004. *Pengenalan Komputer.* Yogyakarta: Andi Offset.
[12] Munir, Rinaldi. 2006. *Diktat Kuliah IF5054 Kriptografi*. Progdi Teknik Informatika Sekolah Teknik Elektro dan Informatika.
[13] Manezes. 2012. *Pengertian dan Sejarah Kriptografi.* Info-dan-pengertian.-blogspot.
[14] Huda, Miftakul. 2015. *Kriptografi Gambar Menggunakan Algoritma One-Time-Pad dan Vigenere Chiper*. Semarang: Program Pascasarjana Universitas Dian Nuswantoro.
[15] Zein, Miftah. 2013. *One Time Pad*. www.zenshifu.com/one-time-pad/.
[16] New, Wave. 2010. *Linear Feed Back Shift Register*. www.new-wafe-instrumen.com.
[17] Sugiyono. 2011. *Metode Penelitian Kuantitatif, Kualitatif dan R&D*. Jakarta: Alfabeta.
[18] Damanic. 2016. *Pengertian Penelitian Eksperimen*. http://pengertian-pengertian-info.blogspot.co.id/2016/02/pengertian-penelitian-eksperimen.html.