ISSN : 1978-6603

JURNAL Ilmiah SAINTIKOM Sains dan Komputer

PERANCANGAN PERANGKAT LUNAK PROSES KRIPTOGRAFI METODE ULTRA 1.03

Widiarti Rista Maya[#1], Saiful Nur Arif[#2] Mukhlis Ramadhan[#3]
[#1]Program Studi Teknik Komputer, STMIKTriguna Dharma
[#1]Email :widya_rmaya87@yahoo.com

ABSTRAK

Kesulitan dari pembelajaran kriptografi adalah bagaimana menjelaskan tahapan-tahapan proses yang terjadi pada pembentukan kunci, enkripsi dan dekripsi. Tahapan-tahapan yang dimiliki oleh setiap metoda kriptografi berbeda-beda dan memiliki tingkat kesulitannya berbeda-beda pula. Dengan kesulitan yang ada maka diperlukan suatu perangkat lunak pembelajaran yang menunjang kemudahan dalam memahami metoda-metoda kriptografi, dalam hal ini khususnya metoda ULTRA 1.03. Metoda ULTRA 1.03 merupakan algoritma *stream cipher* yang memiliki 4 proses utama yaitu proses pembentukan kunci, proses pembentukan *dummy*, enkripsi dan dekripsi. Proses enkripsi dan dekripsi metoda ini memiliki 3 tahapan utama yaitu kompresi data Huffman, kriptografi dengan *Triple Transposition Key* dan pengkodean basis 64. Perangkat lunak pembelajaran kriptografi metoda ULTRA 1.03 mampu menampilkan tahapan penyelesaian untuk proses pembentukan kunci, pembentukan *dummy*, enkripsi dan dekripsi secara langkah demi langkah. Perangkat lunak pembelajaran ini juga menampilkan teori dan algoritma dari metoda ULTRA 1.03. Perangkat lunak pembelajaran kriptografi metoda ULTRA 1.03 dirancang sedemikian rupa untuk membantu pemahaman terhadap cara kerja algoritma dari metoda ULTRA 1.03.

Kata Kunci: Algoritma, Enkripsi, Dekripsi,Metode Ultra 1.03

ABSTRACT

The difficulty of learning cryptography is how to explain the stages of the stages of the process that occurs in the formation of keys, encryption and decryption. Stages stages of every different cryptographic methods and have different levels of difficulty as well. With the difficulties that exist, we need a learning software that support the ease in understanding the methods of cryptography, in this particular method of ULTRA 1:03. Method 1.03 ULTRA is a stream cipher algorithm that has four main processes key establishment process, the process of forming dummy, encryption and decryption. Encryption and decryption of this method has three main stages, namely Huffman data compression, cryptography with Triple transposition Key and base 64 encoding.Learning software ULTRA 1:03 cryptographic method capable of displaying the stage of completion for key establishment process, the formation of the dummy, the encryption and decryption step by step. Learning software also displays the theory and algorithms of methods ULTRA 1.03. Learning software ULTRA 1:03 cryptographic methods designed to help the understanding of the workings of the algorithm of the method ULTRA 1:03.

Keywords: Algorithm, Encryption, Decryption, 1:03 Ultra Methods

A. PENDAHULUAN

Keamanan data merupakan hal yang sangat penting dalam menjaga kerahasiaan informasi terutama yang berisi informasi sensitif yang hanya boleh diketahui isinya oleh pihak yang berhak saja, apalagi jika pengirimannya dilakukan melalui jaringan publik, apabila data tersebut tidak diamankan terlebih dahulu, akan sangat mudah disadap dan diketahui isi informasinya oleh pihak-pihak yang tidak berhak. Salah satu cara yang digunakan untuk pengamanan data adalah menggunakan sistem kriptografi yaitu dengan menyandikan isi informasi (*plaintext*) tersebut menjadi isi yang tidak dipahami melalui proses enkripsi dan untuk memperoleh kembali informasi yang asli, dilakukan proses dekripsi, disertai dengan menggunakan kunci yang benar. Namun, sejalan dengan perkembangan ilmu penyandian atau kriptografi, usaha-usaha untuk memperoleh kunci tersebut dapat dilakukan oleh siapa saja, termasuk pihak yang tidak sah untuk memiliki informasi tersebut. Oleh karena itu, penelitian tentang kriptografi akan selalu berkembang untuk memperoleh algoritma kriptografi yang makin kuat, sehingga usaha-usaha untuk memecah kode kriptografi secara tidak sah menjadi lebih sulit.

Metode Ultra 1.03 merupakan *symmetric stream cipher* dengan *varible length key* yang dikembangkan oleh seseorang yang bernama Dirk Rijmenants. Metode ini dikembangkannya dengan mengkombinasikan algoritma kompresi data Huffman dengan algoritma kriptografi menggunakan *tripple transposition key* untuk menambah keamanan algoritma kriptografi ini. Algoritma yang dikembangkan pada metode Ultra 1.03 ini dapat menerima 2 jenis kunci yaitu kunci utama dan *Private Crypto Code*. Kunci kedua adalah optional yang berfungsi sebagai kunci eksklusif untuk menentukan jenis user group. Algoritma ini akan membentuk 3 *transposition key* yang berukuran 463, 251, 181 yang saling berhubungan karena adanya feedback output dari pemrosesan key. Dengan metode Ultra 1.03 akan menghasilkan output dengan kecepatan yang tinggi serta mempunyai *Private Crypto Code*.

1. Uraian Teoritis

1.1 Definisi *Cryptography*

Kriptografi atau *Cryptography* berasal dari kata kryptos yang artinya tersembunyi dan grafia yang artinya sesuatu yang tertulis (bahasa Yunani) sehingga kriptografi dapat juga disebut sebagai sesuatu yang tertulis secara rahasia(tersembunyi).

Cryptography adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek-aspek pada keamanan informasi misalnya kerahasiaan, integritas data, otentikasi pengirim/penerima data, dan otentikasi data. Dengan pengembangan bidang *cryptography*, pembagian antara apa yang termasuk *cryptography* danapa yang tidak telah menjadi kabur. Dewasa ini, *cryptography* dapat dianggap sebagai perpaduan antara studi teknik dan aplikasi yang tergantung kepada keberadaan masalah – masalah sulit.

Bagi kebanyakan orang, *cryptography* lebih diutamakan dalam menjaga komunikasi agar tetap rahasia. Seperti yang telah diketahui dan disetujui bahwa perlindungan (proteksi) terhadap komunikasi yang sensitif telah menjadi penekanan kriptografi selama ini. Akan tetapi hal tersebut hanyalah sebagian dari penerapan kriptografi dewasa ini.

Terdapat dua proses penting di dalam kriptografi yang berperan dalam merahasiakan suatu informasi yakni enkripsi (*Encryption*) dan dekripsi

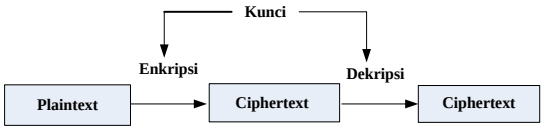
(*Decryption*).Enkripsi ialah transformasi data (*Plaintext*) ke dalam bentuk yang hampir tidak dapat dibaca (*Ciphertext*) tanpa pengetahuan yang cukup. Tujuan dari enkripsi ialah untuk menjamin kerahasiaan dengan menjaga informasi tersembunyi dari siapapun yang bukan pemilik atau yang berkepentingan dengan informasi tersebut, bahkan bagi orang yang memiliki akses terhadap data yang telah dienkripsi. Sedangkan dekripsi ialah kebalikan dari enkripsi, yakni transformasi dari data yang telah dienkripsi (C*iphertext*) kembali ke bentuk semula (*Plaintext*). Proses enkripsi dan dekripsi pada umumnya membutuhkan penggunaan sejumlah informasi yang rahasia, yang sering disebut kunci (*key*).

1.2 Teknik dalam *Cryptography*

Berdasarkan jumlah kunci yang digunakan, ada dua jenis sistem *cryptography* yaitu sistem *cryptography* simetris dan sistem *cryptography* asimetris.

1.2.1 Sistem C*ryptography* Simetris

Enkripsi simetris sering juga disebut sebagai enkripsi konvensional atau enkripsi kunci-tunggal (*single key*). Pada model enkripsi simetris ini digunakan algoritma yang sama untuk proses enkripsi/dekripsi dengan memakai satu kunci yang sama.



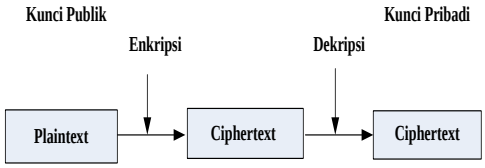
Gambar 1.1 Model sederhana Sistem C*ryptography* Simetris

Keamanan dari enkripsi simetris bergantung pada beberapa faktor. Pertama, algoritma enkripsi harus cukup kuat sehingga tidaklah praktis untuk mendekripsi suatu pesan hanya dengan memiliki *ciphertext* saja. Di samping itu, keamanan dari enkripsi simetris adalah bergantung pada kerahasiaan kunci, bukan kerahasiaan dari algoritma enkripsi itu sendiri. Semakin panjang kunci yang dipakai maka semakin sulit untuk menebak kunci dengan menggunakan metoda *brute force attacks* (mencoba semua kemungkinan kunci).

Algoritma enkripsi simetris yang popular dewasa ini adalah DES (*Data Encryption Standard*) dengan panjang kunci 56-bit, IDEA (128-bit), Twofish (sampai dengan 256-bit), Rijndael (sampai dengan 256-bit) dan lain – lain.

1.2.2 *Sistem Cryptography Asimetris*

Sistem *cryptography* asimetris biasanya lebih dikenal Dengan *cryptography* kunci-publik (public-key *cryptography)*.ide *cryptography* asimetris ini pertama kali dimunculkan oleh whitfielddiffiedan martin hellman pada tahun 1976. Diffie dan hellman mempostulatkan sistem ini tanpa menunjukkan algoritmanya.



Gambar 1.2. Model Sederhana Sistem *Cryptography* Asimetris.

1.2.3 *Block Cipher*dan Mode Operasi.

Sebuah *block ciphe r*adalah sebuahfungsi yang memetakan n-bit blok *plaintext* menjadi n-bit *ciphertext* (Menezes, 1996). Fungsi tersebut terdiri dari sebuah algoritma dan sebuah kunci. Hasil pemetaan dari *plaintext* ke *ciphertex t*akan berbeda-beda tergantung pada kunci yang digunakan. Baik *cryptography* simetris maupun *cryptography* asimetris bisa merupakan *block cipher*.

Untuk *plaintext* yang panjangnya lebih besar dari n-bit perlu dipilih mode

operasi untuk menentukan cara enkripsi / dekripsi *plaintext* tersebut. Ada beberapa pilihan mode operasi yang bisa diterapkan antara lain *Electronic CodeBook* (ECB), *Cipher Block Chaining* (CBC), *Cipher FeedBack* (CFB), *OutputFeedBack* (OFB). Keempat mode operasi ini memiliki kelebihan dan kekurangan masing – masing.

1.3 Aplikasidari*Cryptography*

1.3.1 Privacy

Privacy (kerahasiaan) mungkin merupakan aplikasi paling nyatadari c*ryptography*. *Cryptography* dapat digunakan untuk mengimplementasikan *privacy* hanya dengan mengenkrip informasi yang diinginkan untuk tetap private. Agar seseorang dapat membaca data private ini dia harus mendekrip terlebih dahulu. Kadang-kadang informasi tertentu bukan untuk diakses oleh siapapun juga, dan dalam hal ini informasi dapat disimpan sedemikian rupa sehingga membalik proses merupakan sesuatu yang secara virtual tidakmungkin. Misalnya, dalam sistem multi-user, tidak ada satu orangpun dimungkinkan untuk mengetahui daftar *passwords* dari masing-masing user dalam sistem. Biasanya nilai hash dari *password* yang disimpan bukan *password* itu sendiri. Hal ini memungkinkan user dari sistem yakin betul tentang informasi pribadi disimpan betul-betul aman dari gangguan orang lain karena dengan memasukkan *password* harus diverifikasi terlebih dahulu (dengan menghitung fungsi hashnya dan membandingkan dengan nilai hash yang tersimpan)

1.3.2 Digital Signature dan Authentication

Authentication adalah suatu proses untuk membuktikan dan memverifikasi informasi tertentu. Kadang-kadang seseorang ingin memverifikasi asal dokumen, indentitas pengirim, waktu dan tanggal penandatanganan dan/atau pengiriman, identitas komputer atau user dan lain-lain. Suatu *digital signature* adalah cara *cryptography* dimana dengan cara tersebut beberapa hal di atas dapat diverifikasi. Tanda tangan digital dari suatu dokumen adalah potongan informasi yang didasarkan kepada dokumen dan kunci rahasia penanda-tangan.

1.3.3 Key Agreement Protocol

Suatu *key agreement protoco*l, juga dikenal dengan *key exchange protoco*l, adalah sebarisan langkah yang dilakukan bila dua atau lebih pihak perlu sepakat atas suatu kunci yang digunakan untuk suatu *secret-key cryptosystem*. Protokol ini memungkinkan orang menggunakan kunci secara bersama dengan bebas dan aman melalui suatu medium yang tidak aman, tanpa perlu terlebih dahulu ada pembentukan kunci rahasia bersama.

1.3.4 Digital Envelope

Dalam penggunaan *secret-key cryptosystems*, pertama *user* harus setuju padakuncisesi, yakni, kunci rahasia yang digunakan selama pengiriman satu pesan. Dalam melengkapi tugas tersebut ada resiko bahwa kunci disadap orang lain sewaktu transmisi. Inilah salah satu bagian dari masalah manajemen kunci (*key management problem*). *Public-key cryptography* menawarkan solusi yang menarik terhadap persoalan ini dalam satukerangka yang disebut dengan *digital envelope*.

1.3.5 Identification (Identifikasi)

Identification (identifikasi) adalah suatu proses melalui mana seseorang yakin tentang identitas orang lain atau entitas tertentu. Dalam kehidupan kita sehari-hari kita mengidentifikasikan anggota keluarga kita, kawan, dan teman sejawat dengan karakteristik fisik mereka, seperti suara, muka atau karakteristik lainnya.

Karakteristik ini disebut *biometrics*, yang hanya dapat digunakan pada jaringan dengan perangkat khusus.

Entitas dalam sebuah jaringan dapat juga mengidentifikasikan entitas lain dengan menggunakan metoda *cryptography*.

Otentikasi dan identifikasi adalah dua hal yang berbeda. Identifikasi mengharuskan *verifier* (pelaku verifikasi) membandingkan informasi yang diberikan terhadap semua entitas yang diketahuinya, sedangkan otentikasi membutuhkan pengecekan informasi tentang entitas tunggal yang diberikan dan diidenti-fikasikan sebelumnya. Selanjutnya, identifikasi harus mengidentifikasikan entitas secara unik, sementara autentikasi tidak mengharuskan keunikan. Sebagai contoh, seseorang yang sedang *log into* rekening bersama tidak diidentifikasikan secara unik, tetapi dengan mengetahui *password* bersama, mereka diotentikasikan sebagai salah satu pemilik/pengguna rekening tersebut. Kemudian, identifikasi tidak perlu mengotentikasikan pengguna dalam maksud tertentu.

B. METODOLOGI PENELITIAN

Tujuan dari penelitian ini, dapat membantu pemahaman cara kerja/algoritma kriptografi khususnya metoda ULTRA 1.03.

Pada Metoda ULTRA 1.03 terdapat 4 proses yaitu:

1. Algoritma Proses Pembentukan Kunci.
2. Algoritma Proses Pembentukan *Dummy*.
3. Algoritma Proses Enkripsi.
4. Algoritma Proses Dekripsi.

1. Variabel Yang Diamati

Variabel yang diamati berbentuk pesanteks. Ada beberapa macam bentuk yang akan dilakukan uji coba dalam penelitian diantaranya:

1. terdapat Kunci, *Plaintext* dan *ciphertext* yang berfungsi sebagai data *input* bertipe data *string* (*text*).
2. Panjang PCC (*Private Crypto Code*) dibatasi maksimal 22 karakter.

2. MetodeUltra 1.03

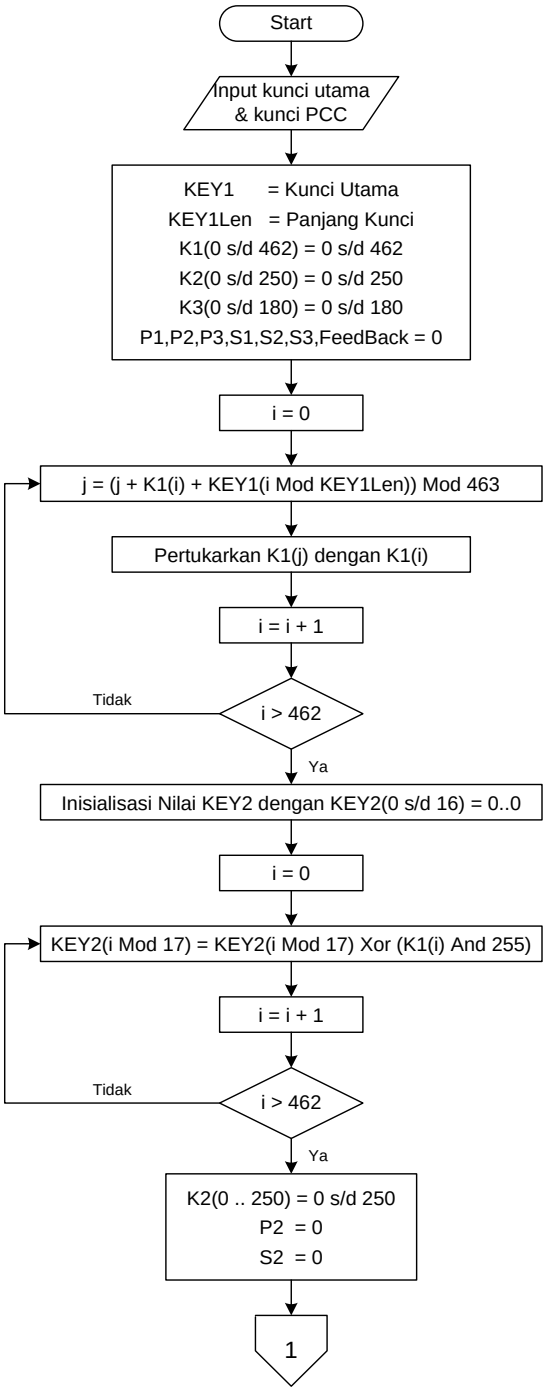
IV.1.1 Algoritma Proses PembentukanKunci

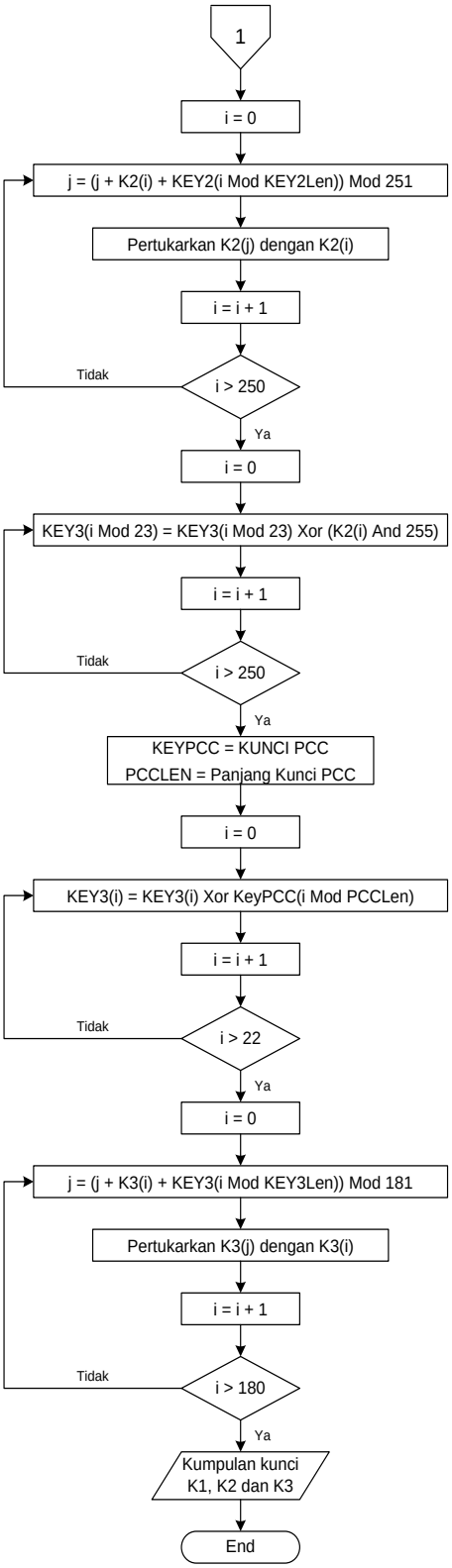
Algoritma ini digunakan dalam proses enkripsi dan dekripsi, sehingga penulis menuliskan algoritma ini terlebih dahulu. Berikut merupakan algoritma proses pembentukan kunci :

1. Input Kunci Utama & Kunci PCC
2. KEY1 = Kunci Utama
KEY1Len = Panjang Kunci
K1(0 s/d 462) = 0 s/d 462
K2(0 s/d 250) = 0 s/d 250
K3(0 s/d 180) = 0 s/d 180
P1,P2,P3,S1,S2,S3,FeedBack = 0
3. i=0
4. j = (j + K1(i) + KEY1(i Mod KEY1Len)) Mod463
5. Pertukarkan K1(j) dengan K1(i)
6. i=i+1
7. Ulang Proses 4 s/d 6 hingga i > 462
8. InisialisasiNilai KEY2 dengan
KEY2(0 s/d 16) = 0..0
9. i=0
10. KEY2(i Mod 17) = KEY2(i Mod 17) Xor (K1(i) And 255)
11. i=i+1
12. Ulangi Proses 10 dan 11 hingga i > 462
13. K2(0 .. 250) = 0 s/d 250
P2 = 0
S2 = 0
14. i=0
15. j = (j + K2(i) + KEY2(i Mod KEY2Len)) Mod 251
16. Pertukarkan K2(j) dengan K2(i)
17. i=i+1
18. Ulangi proses 15 s/d 17 hingga i > 250
19. i=0

20.KEY3(i Mod 23) = KEY3(i Mod 23) Xor (K2(i) And 255)
21.i=i+1
22.Ulangi Proses 20 dan 21 hingga i > 250
23.KEYPCC = KUNCI PCC
PCCLEN = Panjang Kunci PCC
24.i=0
25.KEY3(i) = KEY3(i) Xor KeyPCC(i Mod PCCLen)
26.i=i+1
27.Ulangi Proses 25 dan 26 hingga i > 22
28.i=0
29.j = (j + K3(i) + KEY3(i Mod KEY3Len)) Mod 181
30.Pertukarkan K3(j) dengan K3(i)
31.i=i+1
32.Ulangi Proses 29 s/d 31 hingga i > 180
33.Proses Penbentukan Kunci Selesai

Berikut digambarkan *flowchart* dari proses pembentukan kunci:



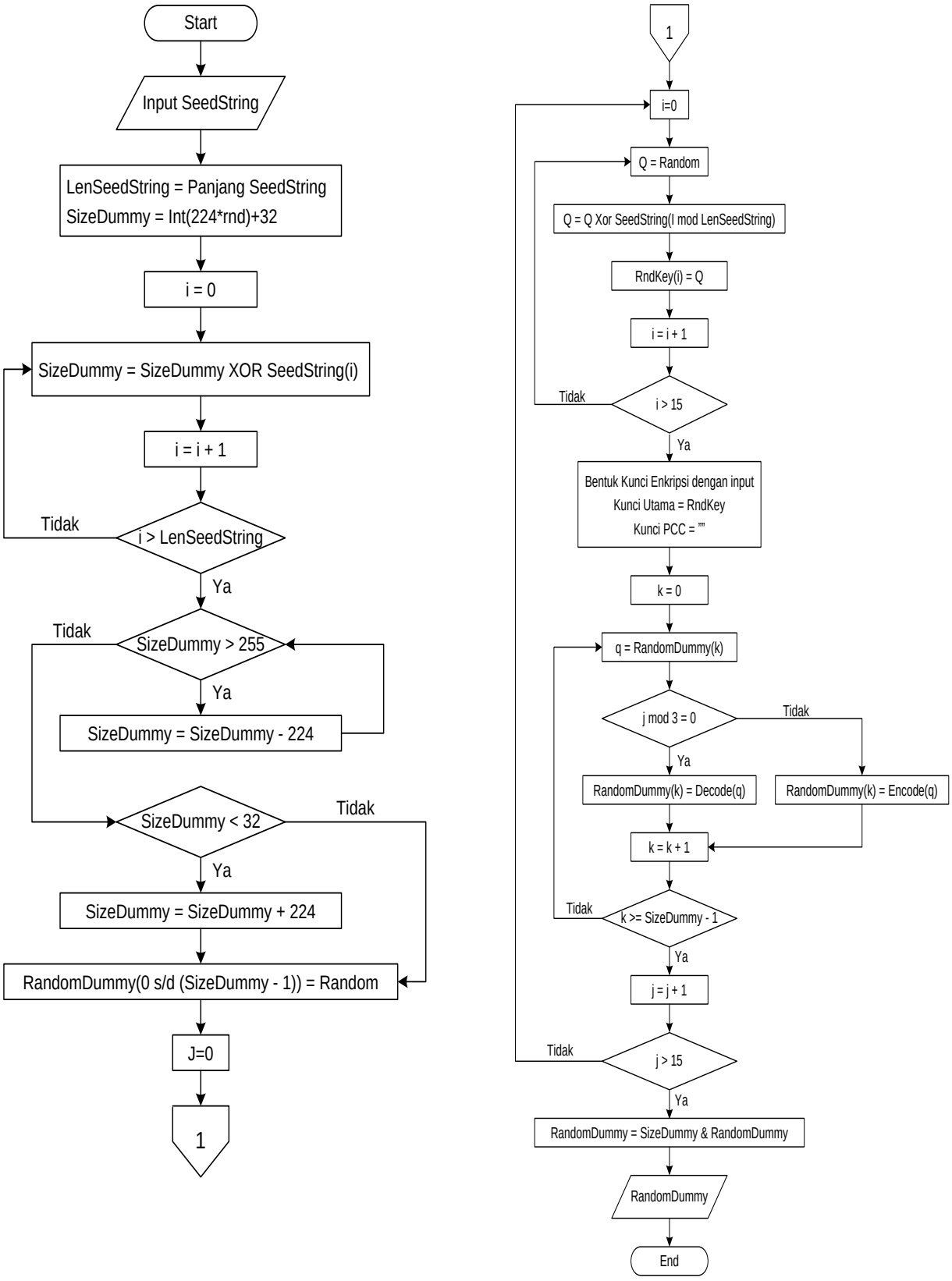


IV.1.2 Proses Pembentukan *Dummy*

Pembentukan *Dummy* hanya dilakukan pada proses Enkripsi. Pembentukan *Dummy* menghasilkan *text* yang panjang maupun isinya random untuk ditambahkan ke dalam *plaintext.* Algoritma Pembentukan *Dummy* antara lain:

1. Input Seed String
Len Seed String = Panjang Seed String
Size Dummy = Int(224*rnd)+32
2. i=0
3. Size Dummy = Size Dummy Xor Seed String(i)
4. i=i+1
5. Ulangi proses 3 dan 4 hingga I > LenSeed String
6. Jika Size Dummy > 255 maka Size Dummy = Size Dummy - 224
7. Ulangi Proses 6 Jika Size Dummy> 255
8. Jika Size Dummy< 32 maka Size Dummy = Size Dummy + 224
9. Random Dummy(0 s/d (Size Dummy - 1)) = Random
10. J=0
11. i=0
12. Q = Random
13. Q = Q Xor Seed String(I mod Len Seed String)
14. Rnd Key(i)=Q
15. i=i+1
16. Ulangi Proses 12 s/d 15 hingga i > 15
17. Bentuk Kunci Enkripsi dengan
input Kunci Utama = RndKey
Kunci PCC=””
18. k=0
19. q = Random Dummy(k)
20. Jika j mod 3 = 0 maka lakukan proses Random Dummy(k) = Decode (q) selain itu Random Dummy(k) = Encode(q)
21. k=k+1
22. Ulangi Proses 19 s/d 21 hingga k >= Size Dummy - 1
23. j=j+1
24. Ulangi Proses 11 s/d 23 hingga j > 15
25. Random Dummy = Size Dummy & Random Dummy
26. Proses Pembentukan Dummy Selesai

Berikut digambarkan *flowchart* dari proses pembentukan *dummy*:

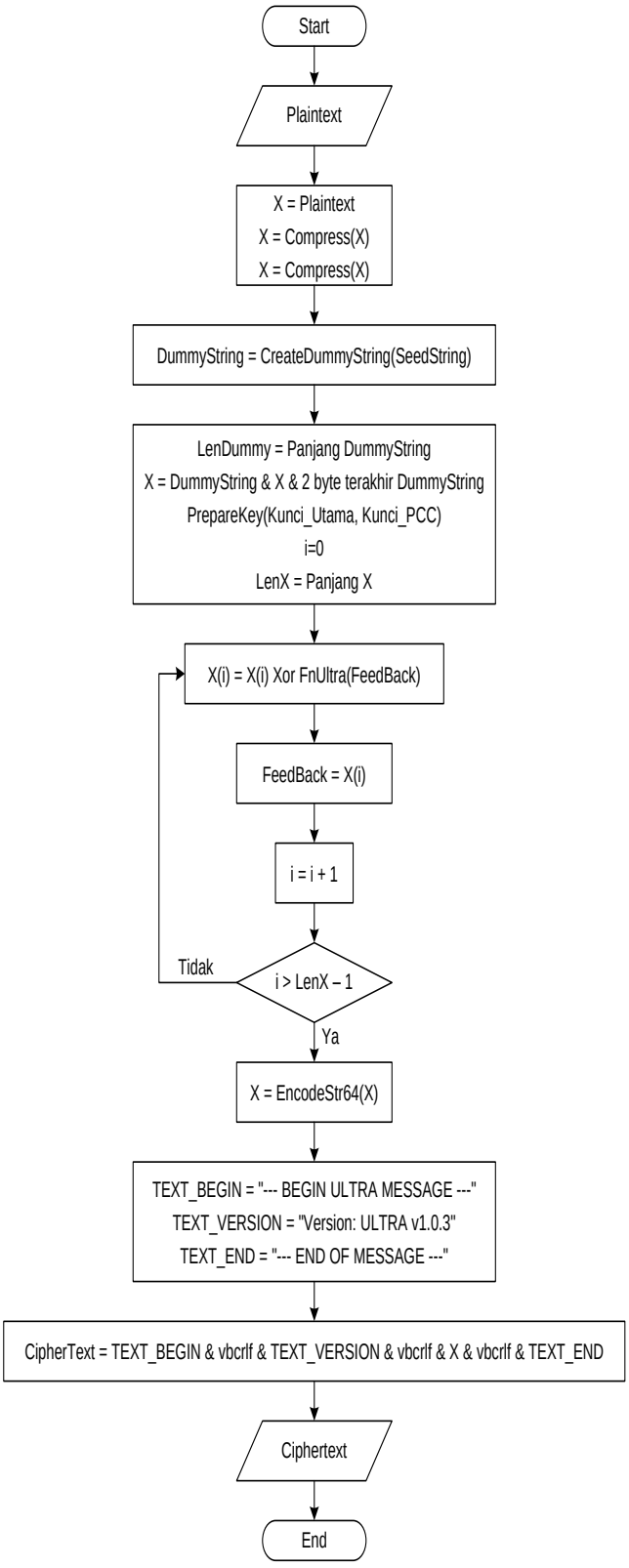

Start
Input SeedString
LenSeedString = Panjang SeedString
SizeDummy = Int(224*rnd)+32
i = 0
SizeDummy = SizeDummy XOR SeedString(i)
i = i + 1
Tidak
i > LenSeedString
Ya
Tidak
SizeDummy > 255
Ya
SizeDummy = SizeDummy - 224
SizeDummy < 32
Tidak
Ya
SizeDummy = SizeDummy + 224
RandomDummy(0 s/d (SizeDummy - 1)) = Random
J=0
1
1
i=0
Q = Random
Q = Q Xor SeedString(I mod LenSeedString)
RndKey(i) = Q
i = i + 1
Tidak
i > 15
Ya
Bentuk Kunci Enkripsi dengan input
Kunci Utama = RndKey
Kunci PCC = ""
k = 0
q = RandomDummy(k)
j mod 3 = 0
Tidak
Ya
RandomDummy(k) = Decode(q)
RandomDummy(k) = Encode(q)
k = k + 1
Tidak
k >= SizeDummy - 1
Ya
j = j + 1
Tidak
j > 15
Ya
RandomDummy = SizeDummy & RandomDummy
RandomDummy
End

IV.1.3 Proses Enkripsi

Proses Enkripsi pada metoda ULTRA 1.03 menerima input berupa *plaintext*, kunci utama dan kunci PCC dan memprosesnya menjadi *ciphertext*. Berikut merupakan algoritma proses enkripsi:

1. X = Plaintext
2. X = Compress(X)
3. X = Compress(X)
4. Dummy String = Create Dummy String(Seed String)
5. Len Dummy = Panjang Dummy String
6. X = Dummy String & X & 2 byte terakhir Dummy String
7. PrepareKey (Kunci_Utama, Kunci_PCC)
8. i=0,LenX = Panjang X
9. X(i) = X(i) Xor Fn Ultra (Feed Back)
10. FeedBack = X(i)
11. i=i+1
12. Ulangi Proses 9 s/d 11 hingga I >Len X – 1
13. X = Encode Str64(X)
14. TEXT_BEGIN = "--- BEGIN ULTRA MESSAGE ---"
15. TEXT_VERSION = "Version: ULTRA v1.0.3"
16. TEXT_END = "--- END OF MESSAGE ---"
17. CipherText = TEXT_BEGIN &vbcrlf& TEXT_VERSION &vbcrlf & X &vbcrlf& TEXT_END
18. Proses Enkripsi Selesai

Berikut digambarkan *flowchart* dari proses enkripsi:



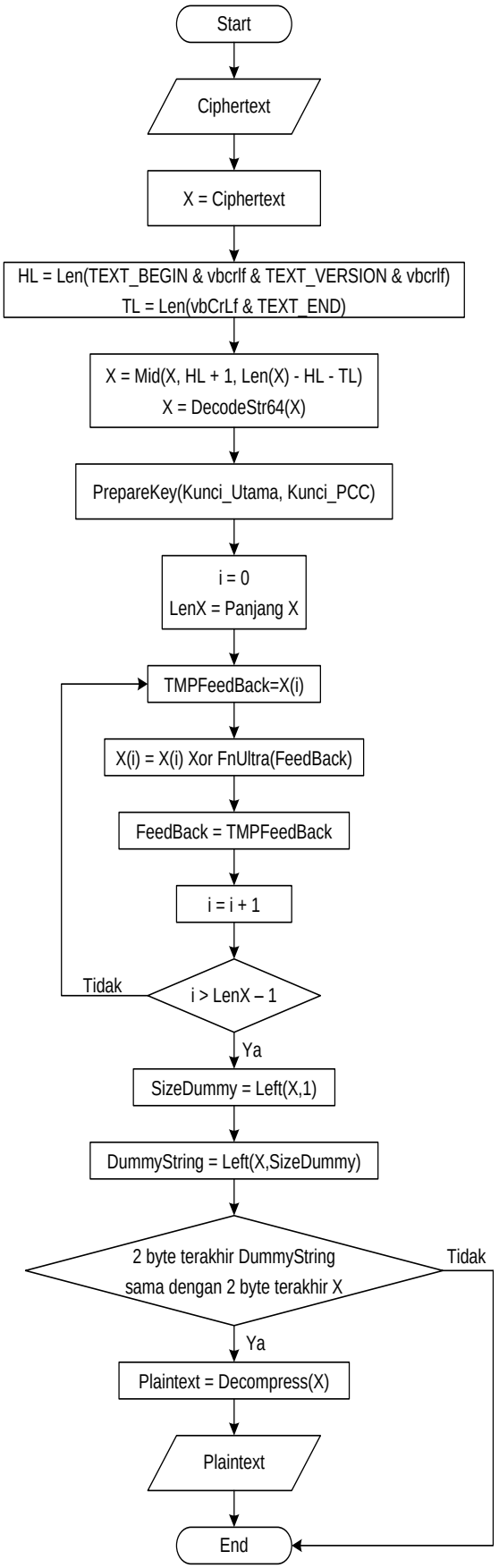
IV.1.4 Algoritma Proses Dekripsi

Proses Dekripsi pada metoda ULTRA 1.03 merupakan kebalikan dari proses Enkripsinya. Berikut merupakan algoritma proses dekripsi:

1. X = CipherText
2. HL = Len(TEXT_BEGIN &vbcrlf& TEXT_VERSION &vbcrlf)
3. TL = Len(vbCrLf& TEXT_END)
4. X = Mid(X, HL + 1, Len(X) - HL - TL)
5. X = DecodeStr64(X)
6. PrepareKey(Kunci_Utama, Kunci_PCC)
7. i=0
8. LenX = Panjang X
9. TMPFeedBack=X(i)
10. X(i) = X(i) XorFnUltra(FeedBack)
11. FeedBack = TMPFeedBack
12. i=i+1
13. Ulangi Proses 9 s/d 12 hingga I >LenX – 1
14. Size Dummy = Left(X,1)
15. Dummy String = Left(X, Size Dummy)
16. Jika 2 byte terakhir Dummy String tidak sama dengan

    2 byte terakhir X maka hentikan proses
17. Plaintext = Decompress(X)

Berikut digambarkan *flowchart* dari proses dekripsi:



C. HASIL DAN PEMBAHASAN

Proses penyelesaian dari kriptografi metoda ULTRA 1.03 dapat dibagi menjadi tiga bagian, yaitu:

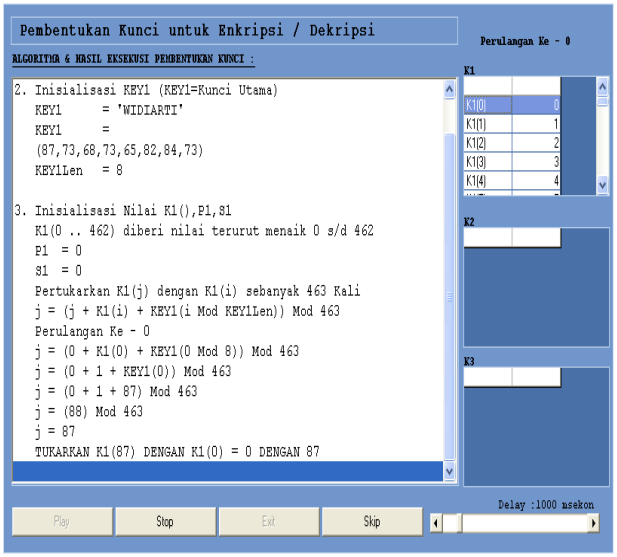
1. Proses PembentukanKunci.
2. Proses Pembentukan *Dummy String*
3. Proses EnkripsidanDekripsi.

Perangkat lunak pembelajaran kriptografi metoda ULTRA 1.03 ini dapat dijalankan dengan cara sebagai berikut :

1. Untuk proses pembentukan kunci, makalakukan proses berikut ini :
   a. Klik menu 'Penerapan', dan sub menu 'Proses Pembentukan Kunci'.

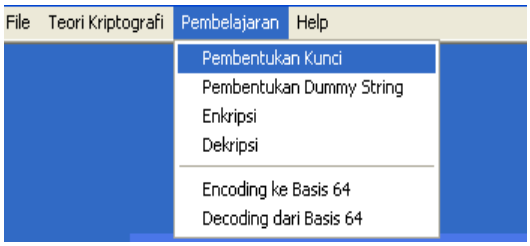
Gambar 4.1 Klik menu 'Penerapan' >> 'Proses Pembentukan Kunci'

   b. Muncul *form* 'Input Kunci Enkripsi / Dekripsi'.



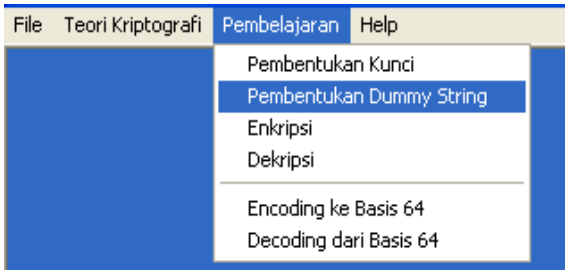
Gambar 4.2 Form Input Kunci Enkripsi / Dekripsi

   c. *Input* kunci utama pada *textbox* 'Kunci Utama' serta Kunci PCC pada *textbox* 'PCC – *Private Crypto Code*', klik tombol 'Ok' untuk memunculkan *form* 'Proses Pembentukan Kunci'. Klik tombol '*Play*' untuk memulai proses pembentukan kunci.



Gambar 4.3 Form Proses PembentukanKunci

2. Untuk proses pembentukan *dummy*, makalakukan proses berikutini :
   a. Klik menu 'Penerapan', dan sub menu 'Proses Pembentukan *Dummy*'.

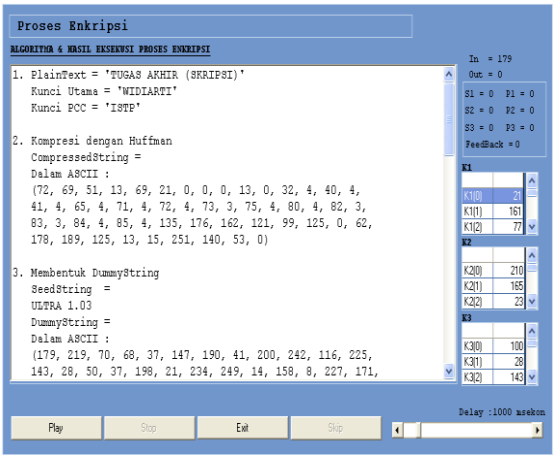


Gambar 4.4 Klik menu 'Penerapan' >> 'Proses Pembentukan Dummy'

   b. Muncul *form* 'Input untuk Pembentukan *Dummy*'.

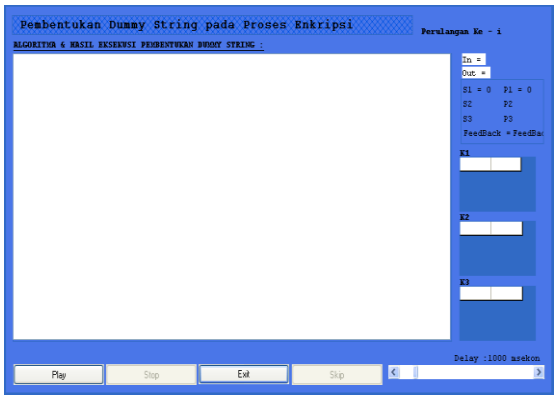
Gambar 4.5 Form Input Data untuk Proses Enkripsi

c. *Input Seedstring*, klik tombol 'Ok' untuk melanjutkan dan memunculkan *form* 'Pembentukan *Dummy*'. Klik tombol 'Play' untuk memulai proses pembentukan *dummy*.

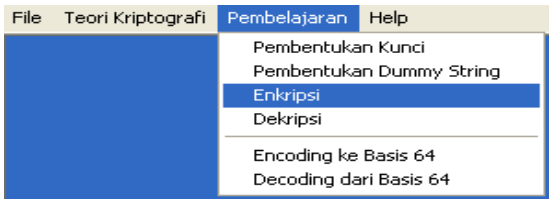


Gambar 4.6 Form Pembentukan Dummy

3. Untuk proses enkripsi, maka lakukan proses berikut ini :
   a. Klik menu 'Penerapan', dan sub menu 'Enkripsi'.



Gambar 4.7 Klik menu 'Penerapan' >> 'Enkripsi'



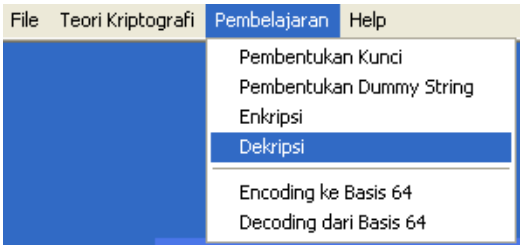
b. Muncul*form* 'Input untukEnkripsi'.

Gambar 4.8 Form Input untukEnkripsi

c. *Input plaintext* dan kedua kunci, klik tombol 'Ok' untuk melanjutkan dan memunculkan *form* 'Proses Enkripsi'. Klik tombol 'Play' untuk memulai proses enkripsi.

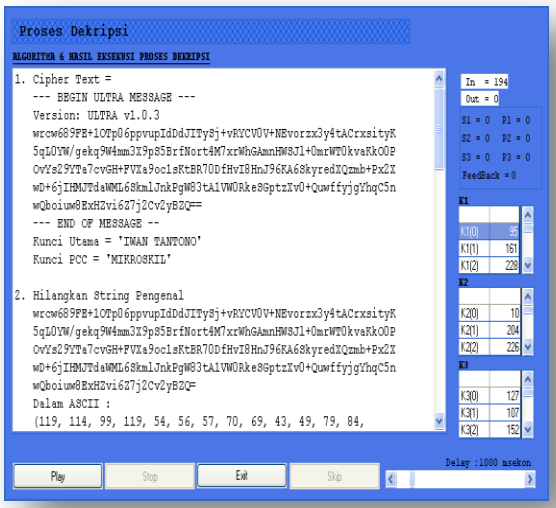
Gambar 4.9 Form Proses Enkripsi

4. Untuk proses dekripsi, maka lakukan proses berikut ini :
   a. Klik menu 'Penerapan', dan sub menu 'Dekripsi'.



Gambar 4.10 Klik menu 'Penerapan' >> 'Dekripsi'

b. Muncul *form* 'Input untukDekripsi'.



Gambar 4.11 Form Input Data untuk Proses Dekripsi

c. *Input ciphertext* dan kedua kunci, klik tombol 'Ok' untuk memunculkan *form* 'Proses Dekripsi'. Klik tombol 'Play' untuk memulai proses dekripsi.

Gambar 4.12 Form Proses Dekripsi

D. SIMPULAN

Berdasarkan pembahasan dan evaluasi dari bab terdahulu, maka dapat ditarik kesimpulan yaitu:

1. Hasil penelitian menunjukkan bahwa Proses Pembentukan Kunci pada Metoda ULTRA 1.03 cukup sederhana walaupun jumlah perulangan yang dilakukan sangat banyak.
2. Metoda ULTRA 1.03 dalam proses enkripsi, dekripsi dan pembentukan *dummy* menggunakan bantuan 3 buah *Transposition Key* dalam prosesnya.
3. Perangkat lunak ini dapat membantu pemahaman cara kerja/algoritma kriptografi khususnya metoda ULTRA 1.03.

Adapun beberapa saran yang mungkin dapat membantu dalam pengembangan perangkat lunak pembelajaran metoda kriptografi yaitu:

1. Dapat dipertimbangkan untuk menambahkan tutorial yang lebih menarik dan lebih sederhana agar lebih mudah dimengerti.
2. Perangkat lunak pembelajaran ini dapat dikembangkan untuk menampilkan proses enkripsi dan dekripsi untuk *file*.
3. Perangkat lunak pembelajaran ini dapat ditambahkan narasi/suara untuk memperjelas dan mempermudah dalam proses pembelajaran.

E. DAFTAR PUSTAKA

Childs, Lindsay N. 2000. *A Concrete Introduction to Higher Algebra. Undergraduate Texts in Mathematics.* New York: Springer-Verlaag.

Kurniawan, Yusuf. 2004. *Kriptografi keamanan internet dan jaringan komunikasi Informatika.* Bandung: Bandung ofset.

Nurnawati, E.K. 2008. *Analisis kriptografi menggunakan algoritma Vigenere cipher dengan mode operasi Cipher Block Chaining (CBC).* Tesis tidak

dipublikasikan. Yogyakarta: IST AKPRIND.

Schneier B., 1996. *Applied Crytography*, Second Edition, John Wiley & Sons, Inc.

Supriyanto, Aji. 2009. *Pemakaian kriptografi kunci publik untuk proses enkripsi dan tandatangan digital pada dokumen e-mail*.*Jurnal Dinamika Informatika***1**(1): 14 - 19.

Wahyuni, Ana. 2011. *Aplikasi kriptografi untuk pengamanan E-dokumen dengan metode hybrid: Biometrik tanda tangan dan DSA (Digital Signature Algorithm).* Tesis tidak dipublikasikan. Semarang: Universitas Diponegoro.

Yuliana, D.K. 2009. *Modul pembelajaran enkripsi dengan menggunakan algoritma DES (Data Enkripsi Standart) melalu visualisasi*. Tesis tidak dipublikasikan. Jakarta: Universitas Guna Darma.